Radial Attention: $\mathcal{O}(n \log n)$ Sparse Attention with Energy Decay for Long Video Generation

Xingyang Li* **Muyang Li*** **Tianle Cai** **Haocheng Xi**
Shuo Yang **Yujun Lin** **Lvmin Zhang** **Songlin Yang** **Jinbo Hu**
Kelly Peng **Maneesh Agrawala** **Ion Stoica** **Kurt Keutzer** **Song Han**

MIT NVIDIA Princeton UC Berkeley Stanford First Intelligence
https://github.com/mit-han-lab/radial-attention

"Nothing spreads without loss; every signal, every influence, every attention — decays with distance." — *Inspired by thermodynamic principles*

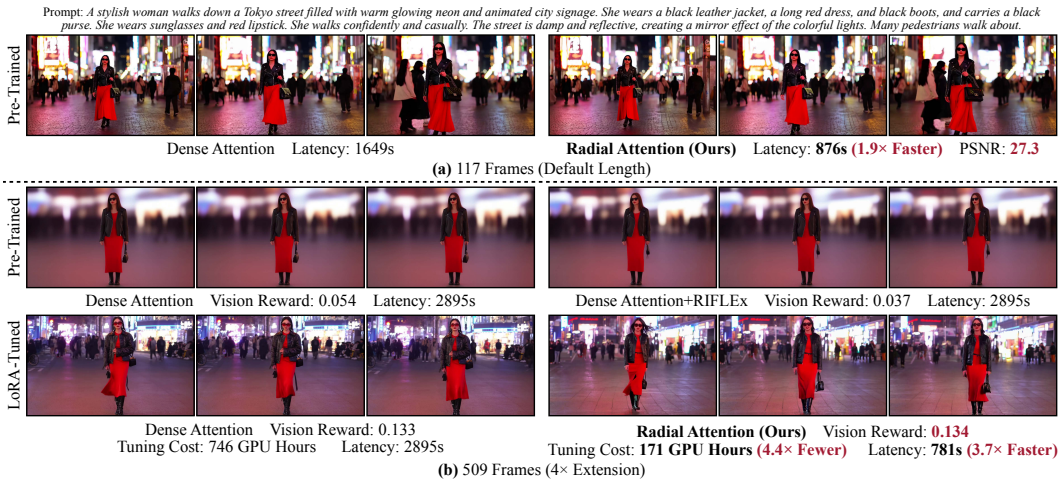


Figure 1: We present Radial Attention, a sparse attention mechanism with $\mathcal{O}(n \log n)$ computational complexity. Radial Attention accelerates pre-trained HunyuanVideo [1] by 1.9× at its default video length while maintaining comparable video quality. When generating 4× longer videos, it reduces tuning costs by up to 4.4× and speeds up inference by up to 3.7× versus dense attention.

Abstract

Recent advances in diffusion models have enabled high-quality video generation, but the additional temporal dimension significantly increases computational costs, making training and inference on long videos prohibitively expensive. In this paper, we identify a phenomenon we term *Spatiotemporal Energy Decay* in video diffusion models: post-softmax attention scores diminish as spatial and temporal distance between tokens increase, akin to the physical decay of signal or waves over space and time in nature. Motivated by this, we propose *Radial Attention*, a scalable sparse attention mechanism with $\mathcal{O}(n \log n)$ complexity that translates energy decay into exponentially decaying compute density, which is significantly more efficient than standard $\mathcal{O}(n^2)$ dense attention and more expressive than linear attention. Specifically, Radial Attention employs a simple, static attention mask where each token attends to spatially nearby tokens, with the attention window size

*indicates equal contributions.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

 https://doi.org/10.52202/085713-0568

shrinking with temporal distance. Moreover, it allows pre-trained video diffusion models to extend their generation length with efficient LoRA-based fine-tuning. Extensive experiments show that Radial Attention maintains video quality across Wan2.1-14B, HunyuanVideo, and Mochi 1, achieving up to a 1.9× speedup over the original dense attention. With minimal tuning, it enables video generation up to 4× longer while reducing training costs by up to 4.4× compared to direct fine-tuning and accelerating inference by up to 3.7× compared to dense attention inference. Code is released at https://github.com/mit-han-lab/radial-attention.

1 Introduction

Diffusion models have achieved remarkable success in generating high-quality images [2, 3]. Recent advances have extended their capabilities to video generation, producing visually compelling results [4, 5, 6, 7, 1].

However, such advances incur substantial computational costs. Unlike image synthesis, the temporal dimension in video synthesis greatly increases token counts compared to images, and the quadratic scaling of self-attention with context length renders training and inference on long videos computationally prohibitive, restricting model scalability.

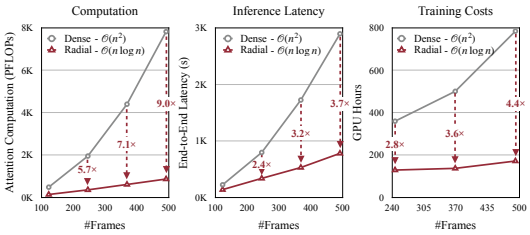


Figure 2: Radial Attention reduces the computational complexity of attention from $\mathcal{O}(n^2)$ to $\mathcal{O}(n \log n)$. When generating a 509-frame 720p video with HunyuanVideo, it reduces the attention computation by 9×, achieves 3.7× speedup, and saves 4.4× tuning costs.

Several prior works tried to mitigate this challenge using sparse attention. For instance, as illustrated in Figure 3(a), Sparse VideoGen (SVG) [8] employs an online profiling strategy that classifies each attention head as either spatial or temporal and then applies the corresponding sparse mask. While this can accelerate inference, it poses challenges during training, especially for longer videos. The profiling may misclassify attention heads on unseen data distributions, whose error can be reinforced during optimization, leading to degraded performance. Other approaches replace the softmax attention with linear alternatives [9, 10], but these often require substantial architectural changes, where modest fine-tuning is typically insufficient to recover the original video quality.

In physics, it is well known that signals and waves lose energy as they propagate through space and time. Inspired by this principle, we observe a similar phenomenon in attention: post-softmax attention scores between tokens diminish as their spatial or temporal distance increases (see Figure 4(b)). We term this phenomenon **Spatiotemporal Energy Decay** and model the decay as an exponential function of both spatial and temporal distances. Based on this model, we unify spatial and temporal attention heads in SVG [8] into **Radial Attention**, a scalable sparse attention mechanism with $\mathcal{O}(n \log n)$ computational complexity (see Figure 2). Radial Attention employs a static sparse attention mask to translate the concept of energy decay into computation density decay. The mask design is simple yet effective: each token attends to others at similar spatial locations, while the attention window shrinks exponentially with temporal distance, as illustrated in Figure 3(b).

Moreover, since Radial Attention only prunes unimportant token relations without modifying the underlying softmax attention mechanism, it enables efficient adaptation of pre-trained video diffusion models to longer sequences using lightweight fine-tuning, such as LoRA [11]. Compared to full-parameter fine-tuning with dense attention, it achieves better video quality, as LoRA focuses on updating parameters most critical for temporal coherence and visual fidelity. The length-extension LoRA is also compatible with existing style LoRAs (see Section 5.2).

When generating videos at the default length, Radial Attention accelerates leading video diffusion models of Wan2.1-14B [7], HunyuanVideo [1] by up to 1.9× speedup. When generating 4× longer videos, Radial Attention reduces tuning costs by up to 4.4× and accelerates inference by up to 3.7× without sacrificing quality. Some visual examples on HunyuanVideo can be found in Figure 1.

https://doi.org/10.52202/085713-0568

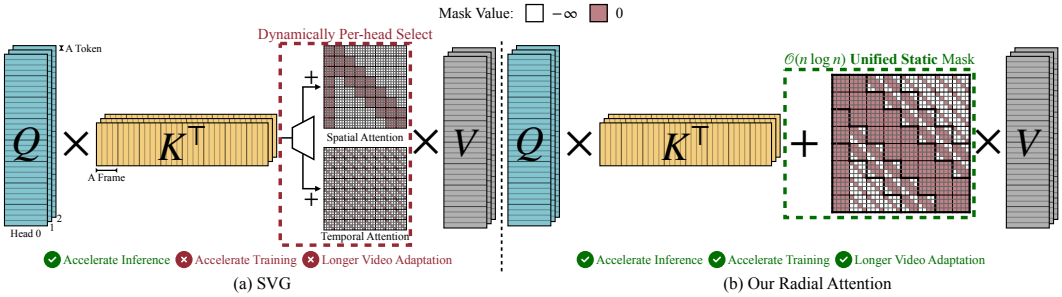


Figure 3: Attention pipelines of SVG [8] and our Radial Attention. Softmax is omitted for clarity. **(a)** SVG dynamically selects either a spatial or temporal attention for each head to speed up inference. However, it does not overcome the original model's length limitation and cannot be trained on unseen distributions like longer videos. **(b)** Our Radial Attention uses a static mask that unifies spatial and temporal attention with $\mathcal{O}(n \log n)$ computational complexity. This static design enables efficient longer-video adaptation.

2 Related Work

Video diffusion models. Diffusion models have achieved state-of-the-art (SOTA) results in image synthesis [2, 3, 12, 10]. Researchers further extend them to the video domain. Early approaches [13, 14, 15, 16] adapted 2D UNets [2, 17] to handle frame sequences by adding temporal modules. Ever since the advent of Sora [4], the community has largely shifted to use DiT [18] as the backbone. Latte [19] first proposed decoupled spatial and temporal attention for modeling video sequences. To better capture long-range dependencies and jointly model spatial-temporal dynamics, recent SOTA models have adopted 3D dense attention [20, 21, 5, 22, 1, 7, 6]. However, dense attention is computationally intensive due to the additional temporal dimension, and its cost scales quadratically with the number of frames, posing substantial challenges for both training and deployment.

Efficient video generation. Many techniques developed to accelerate image diffusion models—such as timestep distillation [23, 24], caching [25, 26], quantization [27, 28, 29], and distributed inference [30, 31, 32]—also apply to video diffusion. However, video models often rely on 3D dense attention, shifting the bottleneck from feedforward to attention layers. Recent works like SageAttention [33, 34, 35, 36] and FlashAttention-3 [37] show that quantizing attention can significantly speed up inference. In large language models (LLMs), sparse attention has been widely applied for acceleration [38, 39, 40, 41, 42, 43, 44, 45, 46]. For instance, Long LoRA [39] combines two local sparse attention patterns with shifting to achieve a global receptive field in video understanding. PowerAttention [45] restricts attention to power-of-two token distances, yielding $\mathcal{O}(n \log n)$ complexity. However, these methods ignore the inherent spatial and temporal structure in video data, making them suboptimal for video generation (see Section 5.2). To better exploit this structure, several video-specific sparse attention methods have been proposed [8, 47, 48, 49]. For example, STA [47] uses sliding 3D windows for local attention, and SVG [8] dynamically selects spatio-temporal patterns for each head. Both improve efficiency but struggle with long videos: STA's fixed receptive field limits long-range dependencies, while SVG's runtime profiling becomes unreliable for unseen long video distributions. In contrast, our Radial Attention employs a static $\mathcal{O}(n \log n)$ pattern across all heads, accelerating both training and inference and enabling efficient extension to longer videos.

Long video generation. Due to the quadratic cost of dense attention, training and inferernce on long videos remain highly expensive. RIFLEx [50] extends video length by modifying RoPE [51] frequencies to tackle temporal repetition and motion deceleration, allowing $2\times$ extrapolation with the pre-trained models. However, it still suffers from poor video quality (*e.g.*, blurring) when generating longer videos. Dalal *et al*. generate short video segments and stitch them together via test-time training layers [52]. Framepack [53] adopts an autoregressive strategy, generating short clips sequentially based on context frames that are encoded into a fixed number of tokens. Other approaches replace dense attention with linear attention [10, 9, 54, 55, 56, 57, 58, 59], offering faster computation and global receptive fields. However, linear attention struggles to capture local details [60], often degrading quality. Our Radial Attention strikes a middle ground between $\mathcal{O}(n^2)$ dense attention and $\mathcal{O}(n)$ linear attention, achieving $\mathcal{O}(n \log n)$ complexity while preserving the visual fidelity. Moreover, it can be efficiently fine-tuned from existing models using LoRA [11], enabling scalable longer-video generation with minimal overhead.

Attention with $\mathcal{O}(n \log n)$ complexity. Preliminary efforts in this direction include Reformer [61], which approximates dense attention via locality-sensitive hashing; H-Transformer [62], which

https://doi.org/10.52202/085713-0568

imposes a hierarchical structure on the attention matrix; Multi-resolution attention [63], which recursively refines high-attention regions; Fast Multipole Attention [64], which adapts the classical fast multipole method; and LogSparse Transformer [65] for time-series forecasting, which restricts each token to attend to $\mathcal{O}(\log n)$ positions per layer. However, these methods are often hardware-unfriendly and exhibit limited scalability. In contrast, our approach employs a simple, block-friendly static mask that scales efficiently while maintaining strong modeling capacity.

3 Preliminary

Diffusion models synthesize videos by sampling Gaussian noise $\boldsymbol{X}_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ in a latent space and progressively denoising it through a neural network to produce a clear latent $\boldsymbol{X}_0$, which is subsequently decoded into the final video using a pre-trained decoder. Compared to images, videos introduce an additional temporal dimension, significantly increasing the number of latent tokens. For instance, generating a 5-second 720p video in HunyuanVideo [1] requires approximately 110K tokens. Excessive latent compression degrades video quality [3], limiting token reduction.

To capture spatiotemporal correlation in video generation, recent models [7, 1, 22, 5] use 3D dense attention, which computes interactions between all token pairs. Given n tokens with embedding dimension d, attention is computed as:

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d}}\right)\boldsymbol{V}, \tag{1}$$

where $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{n\times d}$ are the query, key, and value matrices. The computation of $\boldsymbol{Q}\boldsymbol{K}^T$ incurs $\mathcal{O}(n^2)$ time and memory complexity. While FlashAttention [66, 67] series reduce memory overhead, the quadratic time complexity remains a bottleneck, especially for long or high-resolution videos. Thus, designing more efficient attention mechanisms is vital for scaling video diffusion models.

To mitigate this computational burden, sparse attention restricts interactions to a subset of token pairs. Formally, this is achieved by adding a sparsity mask $\boldsymbol{M} \in \{-\infty, 0\}^{n\times n}$ to the attention logits:

$$\text{SparseAttention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top + \boldsymbol{M}}{\sqrt{d}}\right)\boldsymbol{V}. \tag{2}$$

Entries set to $-\infty$ are ignored in the softmax computation. Various schemes have been proposed to construct the mask. Static methods, such as STA [47], use fixed sparsity patterns but are less expressive. In contrast, dynamic schemes like SVG [8] does dynamic sparse pattern based on input content to improve fidelity. However, dynamic masking introduces online mask decision overhead and does not apply to training. Can we design a static attention pattern that matches the expressiveness of dynamic methods and can also be used in training?

4 Method

The key insight of Radial Attention is that attention scores between tokens decay with increasing spatial and temporal distance. This motivates us to allocate computation based on the inherent spatiotemporal correlations. In Section 4.1, we characterize the spatiotemporal energy decay phenomenon in attention. In Section 4.2, we formally define Radial Attention, which translates energy decay into corresponding compute density reduction, enabling speedup. We also analyze its complexity and approximation error, showing that the complexity is $\mathcal{O}(n \log n)$ and the effectiveness of our mask. Finally, in Section 4.3, we show how to extend pre-trained models to longer videos using Radial Attention.

4.1 Spatiotemporal Energy Decay in Attention

In Figure 4(a), we show two post-softmax attention maps from HunyuanVideo [1]. Following the terminology in SVG [8], the left map is referred to as spatial attention, where each token primarily attends to nearby tokens within adjacent frames. The right map represents temporal attention, where each token focuses on tokens at the same spatial location across different frames. Figure 4(b) illustrates the attention score distributions of these two maps, along with a third curve of averaged attention scores over multiple heads and diffusion steps. In Figure 4(b1), we show the average attention score between tokens at the same spatial location but with increasing temporal distance. In Figure 4(b2),

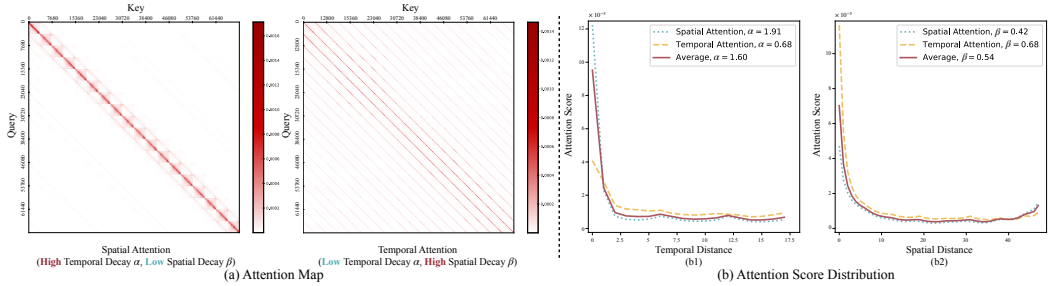


Figure 4: **(a)** Example spatial and temporal attention maps from HunyuanVideo (defined in Section 4.1). **(b)** Attention score distributions. (b1): Average score between tokens at the same spatial location decreases with temporal distance (b2): Average attention score within a frame decreases with spatial distance. *Spatial* and *Temporal Attention* refer to the distributions derived from the corresponding maps in (a). *Average* means averaging over multiple random maps and diffusion steps. The plots indicate that spatial attention shows a high temporal decay and relatively low spatial decay, while temporal attention exhibits the opposite.

we show the average score between tokens in the same frame as spatial distance increases. In both cases, attention scores exhibit a clear decay pattern with increasing distance between the query and key tokens. We refer to this phenomenon as Spatiotemporal Energy Decay. Moreover, regression analysis suggests that this decay closely follows an exponential distribution (see Section 5.3).

Specifically, following the notation from Section 3, assume the video latent consists of f frames, each containing s tokens (in total $n = fs$ tokens). Consider a query token located at the k_0-th spatial position of the i_0-th frame. The corresponding attention score after applying softmax, denoted by $\boldsymbol{p} \in [0,1]^n$, is given by $\boldsymbol{p} = \mathrm{softmax}(\boldsymbol{Q}_{i_0 s+k_0} \boldsymbol{K}^\top)$. Then there exist constants $\alpha, \beta > 0$ and $C_{\text{rel}} > 0$, for each key token at spatial position l in frame j satisfying

$$p_{js+l} \leq C_{\text{rel}} e^{-\alpha|j-i_0|-\beta|l-k_0|} p_{i_0 s+k_0}. \tag{3}$$

Parameters α and β control temporal and spatial decay, respectively. High β (strong spatial locality) and low α model temporal attention, while high α and low β capture spatial attention, as shown in the empirical plots in Figure 4(b). This motivates our unified sparsity pattern that leverages both spatial and temporal decay in a principled manner.

4.2 Radial Attention: Convert the Energy Decay to Compute Density Decay

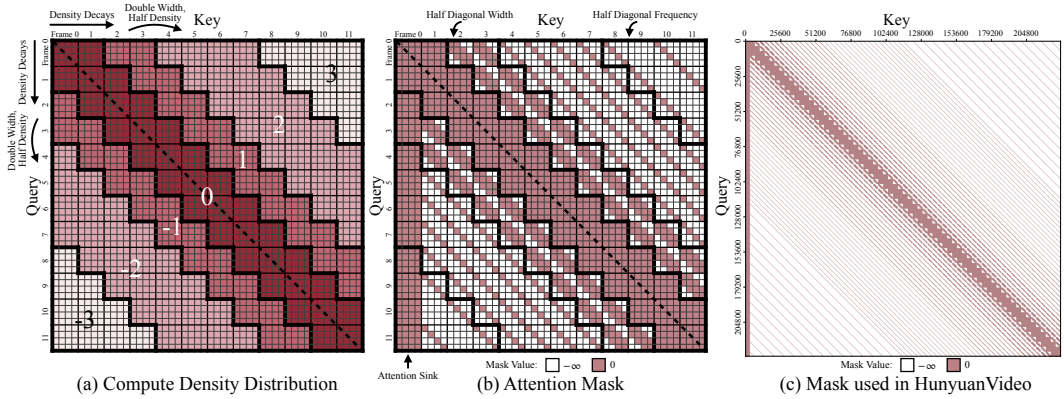
Radial Attention simulates energy decay through compute density decay to save computation.

Temporal density decay. Along the temporal dimension, Radial Attention applies an exponential decay rule: the compute density between tokens in frame i and frame j is $(\frac{1}{2})^{\lfloor \log_2(\max(|i-j|,1)) \rfloor}$. This forms a structured pattern as illustrated in Figure 5(a) with $2\lceil \log_2(\max(f,2)) \rceil - 1$ diagonal bands centered on the main diagonal (band 0). Bands above and below the diagonal are indexed as $1, 2, 3, \ldots$ and $-1, -2, -3, \ldots$, respectively. Each band's width doubles relative to the previous one, ensuring that the total computation per band remains bounded by a constant. The attention from tokens in frame i to frame j lies in band $\mathrm{sign}(j-i) \cdot \lfloor \log_2 \max(|i-j|,1) \rfloor$. The central band (band 0) retains 100% compute density, while each successive band moving outward has half the compute density of the preceding one – producing a radial decay effect with progressively lighter colors.

Spatial density decay. As observed in Figure 4 and formalized in Equation 3, most attention energy is concentrated on tokens at similar spatial locations across frames. We preserve these high-energy interactions, which yield diagonal-like structures within each frame-to-frame attention block. Due to temporal decay, the computed diagonal width of these blocks shrinks as the temporal distance between frames increases. Specifically, as shown in Figure 5(b), the diagonal width for attention between frame i and frame j is given by $\lfloor \frac{s}{2^{\lfloor \log_2 \max(|i-j|,1) \rfloor}} \rfloor$. If it falls below 1, instead of further narrowing the diagonal, we reduce the frequency of diagonals. Specifically, we only retain diagonals in those blocks where $|i-j| \bmod \lceil \frac{2^{\lfloor \log_2 \max(|i-j|,1) \rfloor}}{s} \rceil = 0$ to keep the same amortized attention density decay.

Formal definition. Here we formally define the Radial Attention mask. We construct a 4D attention mask $\tilde{\boldsymbol{M}} \in \{-\infty, 0\}^{f \times f \times s \times s}$, where each element $\tilde{M}_{i,j,k,l} = 0$ indicates that the token at spatial position k in frame i is permitted to attend to the token at position l in frame j. Conversely, $\tilde{M}_{i,j,k,l} =$

https://doi.org/10.52202/085713-0568



(a) Compute Density Distribution (b) Attention Mask (c) Mask used in HunyuanVideo

Figure 5: **(a)** The compute density pattern. The attention map is divided into $2\lceil \log_2(\max(f,2))\rceil - 1$ bands (here, the number of frames $f = 12$) based on the temporal distance between tokens. The central band has full compute density, while each successive outer band has half the density of the previous one. Except for band ± 1, each band also doubles the diagonal width of its predecessor. **(b)** The corresponding attention mask for (a). The compute density is reflected in the compute diagonal width of each frame-to-frame block. When the diagonal width drops below 1, we reduce the frequency of diagonals. We additionally add an attention sink. **(c)** An example mask used in HunyuanVideo, illustrating the final sparsity pattern in practice.

$-\infty$ denotes that attention between the token pair is suppressed. The mask is constructed according to:

$$\tilde{M}_{i,j,k,l} = \begin{cases} 0, & \text{if } 2^{\lfloor \log_2 \max(|i-j|,1)\rfloor} \le s \text{ and } |k-l|+1 \le \frac{s}{2^{\lfloor \log_2 \max(|i-j|,1)\rfloor}} \\ 0, & \text{if } |i-j| \bmod \lceil \frac{2^{\lfloor \log_2 \max(|i-j|,1)\rfloor}}{s} \rceil = 0 \text{ and } k = l \\ -\infty. & \text{otherwise} \end{cases} \tag{4}$$

The final attention mask $\boldsymbol{M} \in \{-\infty, 0\}^{n\times n}$ used in the attention operation of Equation 2 is obtained by flattening frame and spatial indices: $M_{is+k,js+l} = \tilde{M}_{i,j,k,l}$. For better quality, we incorporate an attention sink [38, 8] as the first frame's attention is crucial. Figure 5(c) shows a example mask we use in HunyuanVideo for generating a 253-frame 720p video. This strategy keeps spatial interactions with high temporal proximity while using sparse sampling for distant frames to maintain efficiency.

Relation to SVG. Radial Attention unifies spatial and temporal attention in SVG [8] using a single attention mask. Specifically, the central band (band 0 in Figure 5(a)) of our mask already captures dense spatial interactions, effectively subsuming the spatial attention in SVG. For temporal attention, SVG overlooks temporal decay, allocating unnecessary computation to distant frames with low relevance. In contrast, Radial Attention reduces attention to these regions and reallocates the budget toward tokens nearer in time, achieving both improved efficiency and enhanced modeling.

Complexity analysis. The computational cost of our method is proportional to the number of zeros in the attention mask $\tilde{\boldsymbol{M}}$. When the number of frames f is large, we derive the following upper bound:

$$\#\text{zeros in } \tilde{\boldsymbol{M}} \le \underbrace{4s^2 f}_{\text{central band and sink}} + \underbrace{\sum_{r=1}^{\lfloor \log_2 s\rfloor} 2^{r+1} f \frac{2s^2}{2^r}}_{\text{diagonal width}\ge 1} + \underbrace{\sum_{r=\lfloor \log_2 s\rfloor+1}^{\lceil \log_2 f\rceil - 1} 2^{\lfloor \log_2 s\rfloor+1} fs}_{\text{diagonal width}<1} \tag{5}$$

$$\le 4s^2 f \log_2 f = 4sn(\log_2 n - \log_2 s). \tag{6}$$

A detailed derivation of Equation 5 can be found in Appendix A.1. From Equation 6, we find that for long videos (i.e., large f) with fixed resolution s, the computational complexity scales as $\mathcal{O}(n \log n)$. Empirical results on HunyuanVideo, shown in Figure 2, confirm this trend. Notably, our Radial Attention reduces attention computation by 9× compared to dense attention for 4× longer videos.

Error analysis. Following Equation 3, we derive an error bound for the attention score corresponding to a query token at position k_0 in the i_0-th frame. $\tilde{\boldsymbol{p}} = \text{softmax}(\boldsymbol{Q}_{i_0 s+k_0} \boldsymbol{K}^\top + \tilde{\boldsymbol{M}}_{i_0 s+k_0})$ denotes the masked attention score. The ℓ_1 attention error of our approximated attention is bounded as follows:

$$\left\|\tilde{p} - p\right\|_1 \le C_{\text{rel}} \left[\frac{8\, e^{-\beta\left(\frac{s}{2}+1\right)}}{(1-e^{-\alpha})(1-e^{-\beta})} + 4\, \frac{1+e^{-\beta}}{1-e^{-\beta}}\, \frac{e^{-\alpha(s+1)}}{1-e^{-\alpha}} \right] = O(C_{\text{rel}} e^{-\min(\beta/2,\alpha)s}). \tag{7}$$

Table 1: Quantitative results at the default video length. Under the same computation budget, our method consistently outperforms STA and PA in PSNR, SSIM, and LPIPS, matches the video fidelity of SVG, and achieves 1.8× speedup on HunyuanVideo and Wan2.1-14B on a single H100 GPU.

Model	Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)	Vision Reward (↑)	PFLOPs	Latency (s)	Speedup
Hunyuan Video (117 frames)	Original	–	–	–	0.141	612	1649	–
	STA (FA3)	26.7	0.866	0.167	0.132	331	719	2.29×
	PA	22.1	0.764	0.256	0.140	339	1002	1.65×
	SVG	27.2	**0.895**	**0.114**	**0.144**	340	867	1.90×
	Ours	**27.3**	0.886	**0.114**	0.139	339	876	1.88×
Wan2.1-14B (69 frames)	Original	–	–	–	**0.136**	560	1630	–
	STA (FA3)	22.9	0.830	0.171	0.132	322	812	2.01×
	PA	22.4	0.790	0.176	0.126	324	978	1.67×
	SVG	23.2	0.825	0.202	0.114	324	949	1.71×
	Ours	**23.9**	**0.842**	**0.163**	0.128	323	917	1.77×

Proof details are provided in Appendix A.2. As Equation 7 shows, the error decreases exponentially with larger decay rates α and β. In Section 5.3, we further empirically compare this error bound to that of SVG, showing that Radial Attention achieves smaller errors, thereby validating its effectiveness.

Hardware-friendly block sparsity. To ensure efficient execution on modern hardware, attention is computed over 128×128 blocks rather than individual 1×1 tokens [68, 8, 40, 43, 44, 66].

4.3 Low-Rank Adaptation for Long Videos

Although we employ an efficient attention mechanism, the pre-trained model was originally trained on short videos. Recent works [50] have explored training-free methods for extending generation to longer videos, but their performance remains limited due to length distribution mismatch. Training directly on long videos, meanwhile, is computationally prohibitive. Radial Attention alleviates this challenge by reducing the training time complexity to $\mathcal{O}(n \log n)$. Importantly, it preserves critical inter-token relations in the softmax attention, allowing the original pre-trained weights to remain largely intact. Thus, only minimal fine-tuning is required. To further minimize training overhead, we incorporate low-rank adapters (LoRA) [11, 39] into the attention mechanism. Specifically, LoRA is applied to the query, key, value, and output projections of the attention layers, enabling efficient fine-tuning with significantly reduced memory and computational costs. Empirically, we find that LoRA fine-tuning with Radial Attention not only minimizes overhead but also improves video quality by refining only the most critical weights and attention more effectively. See Section 5.3 for detailed results.

5 Experiments

5.1 Setups

Models. We benchmark our method on three popular text-to-video diffusion models: Mochi 1 [22], HunyuanVideo [1], and Wan2.1 [7], which contain 10, 13, and 14 billion parameters, respectively. Mochi 1 can generate up to a 5-second video with 480p resolution and 162 frames. HunyuanVideo can generate up to a 5-second video with 720p resolution and 125 frames. Wan2.1-14B can generate up to a 5-second video with 720p and 81 frames.

Benchmarks. We use Vision Reward [69] (higher is better) to approximate the human rating of the generated videos. For pre-trained models evaluated at their default video lengths, we further report PSNR and SSIM to quantify numerical similarity, and LPIPS [70] to assess perceptual differences between the outputs of the original models and the benchmarked methods. For longer-video generation, we additionally use VBench-long[71] to evaluate our fine-tuned models. Specifically, we report metrics of subject consistency, aesthetic quality, and imaging quality, where the original models exhibit notable degradation.

Baselines. We compare Radial Attention against the following methods:

- **SVG [8]:** Accelerates video models with sparse attention by dynamically classifying attention heads as spatial or temporal and applying corresponding masks.
- **Spatial/Temporal:** The respective attention masks used in SVG's spatial and temporal heads, as described in Section 4.1.
- **STA [47]:** Applies sliding window attention to capture spatially and temporally local dependencies.

https://doi.org/10.52202/085713-0568

Table 2: Quantitative results at the extended (2× and 4×) video lengths. With minimal fine-tuning, our method maintains the quality regarding Vision Reward and multiple VBench dimensions (Subject Consistency, Aesthetic Quality, and Image Quality) when the length grows. It also achieves high sparsity, reducing training costs by up to 4.4× and delivering up to 3.7× inference speedup.

Model	#Frames	Method	Sparsity	Training Time (h)	Training Speedup	Inference Time (s)	Inference Speedup	Vision Reward (↑)	VBench		
									S.C.	A.Q.	I.Q.
Hunyuan Video	125 (1×)	Original	0.00%	–	–	225	–	0.119	0.959	0.643	0.672
	253 (2×)	Original	0.00%	–	–	797	1.00×	0.122	0.953	0.603	0.611
		RIFLEx	0.00%	–	–	797	1.00×	**0.128**	0.969	0.622	0.614
		Spatial	80.5%	16.0	2.81×	335	2.38×	0.054	**0.979**	0.607	0.670
		Temporal	80.7%	16.2	2.78×	338	2.36×	0.104	0.963	0.620	0.658
		Long LoRA	80.6%	16.6	2.71×	363	2.20×	0.112	0.958	0.620	**0.685**
		PA [45]	80.4%	16.7	2.69×	334	2.39×	0.109	0.967	0.608	0.653
		SANA	–	12.8	3.52×	285	2.80×	-0.205	0.907	0.300	0.442
		Full	0.00%	45.0	1.00×	797	1.00×	0.124	0.955	0.616	0.648
		Ours	80.8%	16.2	**2.78**×	339	**2.35**×	0.126	0.968	**0.623**	0.663
	509 (4×)	Original	0.00%	–	–	2895	1.00×	0.054	0.988	0.545	0.451
		RIFLEx	0.00%	–	–	2895	1.00×	0.037	**0.989**	0.539	0.456
		Spatial	88.3%	20.7	4.52×	755	3.83×	0.112	0.922	0.598	0.664
		Temporal	88.2%	21.1	4.44×	774	3.74×	0.083	0.972	0.597	0.646
		Long LoRA	88.4%	20.9	4.48×	803	3.61×	0.130	0.936	0.618	**0.689**
		PA [45]	88.2%	21.8	4.29×	766	3.78×	0.128	0.950	0.590	0.648
		Full	0.00%	93.6	1.00×	2895	1.00×	0.133	0.977	0.590	0.635
		Ours	88.3%	21.4	**4.37**×	781	**3.71**×	**0.134**	0.973	**0.623**	0.672
Mochi 1	163 (1×)	Original	0.00%	–	–	112	–	0.071	0.973	0.623	0.672
	331 (2×)	Original	0.00%	–	–	302	1.00×	0.040	0.937	0.551	0.466
		Spatial	76.1%	8.57	1.75×	186	1.62×	0.088	0.935	0.596	0.595
		Temporal	76.3%	8.54	1.76×	189	1.60×	0.075	0.936	0.591	0.593
		Long LoRA	76.0%	9.07	1.65×	210	1.44×	0.095	0.950	0.596	**0.630**
		PA [45]	77.8%	8.53	1.76×	183	1.65×	0.101	0.946	0.610	0.626
		SANA	–	8.22	1.82×	166	1.82×	-0.201	0.905	0.334	0.568
		Full	0.00%	15.0	1.00×	302	1.00×	0.095	0.923	0.610	0.594
		Ours	76.4%	8.43	**1.78**×	185	**1.63**×	**0.110**	**0.951**	**0.615**	0.602
	667 (4×)	Original	0.00%	–	–	992	1.00×	-0.091	0.916	0.383	0.322
		Spatial	85.2%	17.4	2.83×	382	2.60×	0.091	0.930	0.611	0.585
		Temporal	85.4%	17.6	2.80×	393	2.52×	0.028	0.931	0.556	0.536
		Long LoRA	86.0%	19.0	2.59×	426	2.33×	0.086	0.944	0.584	0.543
		PA [45]	86.5%	17.3	2.84×	381	2.60×	0.107	0.956	**0.633**	**0.650**
		Full	0.00%	49.2	1.00×	992	1.00×	0.099	0.934	0.613	0.613
		Ours	85.5%	17.4	**2.83**×	386	**2.57**×	**0.113**	**0.958**	0.618	0.638
Wan2.1 -14B	81 (1×)	Original	0.00%	–	–	1630	–	0.135	0.973	0.623	0.672
	161 (2×)	Original	0.00%	–	–	5735	1.00×	0.109	0.946	0.598	0.614
		Full	0.00%	28.0	1.00×	5735	1.00×	**0.150**	0.966	0.590	**0.689**
		Ours	73.6%	14.5	**1.93**×	2847	**2.01**×	0.145	**0.981**	**0.607**	0.677

- **PowerAttention (PA) [45]:** A sparse attention mechanism with $\mathcal{O}(n \log n)$ complexity for LLMs, attending only to tokens at power-of-two distances.
- **LongLoRA [39]:** Uses shifted local attention to efficiently extend the context window of LLMs.
- **SANA[10]:** An efficient diffusion model backbone with linear attention. We replace softmax attention with SANA's for adapting to longer videos.
- **RIFLEx[50]:** Training-free video length extrapolation by adjusting the frequency of RoPE [51].

Implementation details. In terms of Radial Attention implementation, we use FlashInfer [72] for inference and Block-Sparse-Attention [73] with FlashAttention-2 [67] backend during training. For default-length inference, we evaluate HunyuanVideo on 117 frames at 768p resolution (768×1280), and Wan2.1 on 69 frames at the same resolution. Following SVG, we apply dense attention during the first 12 steps as a warm-up phase for all models. Additionally, we keep dense attention in the first DiT block to maintain quality. We measure all the latencies with a single NVIDIA H100 GPU.

For longer-video generation, we fine-tune the model with videos that are 2∼4× longer than the default length from OpenVid-1M [74]. Specifically, we sample 2k top-scoring videos in aesthetic and motion scores for each extended length. We use 8 H100 GPUs for training, which takes around 16∼21 hours for HunyuanVideo, 8∼17 hours for Mochi 1, and 15 hours for Wan 2.1. Inference latency for Wan 2.1 is measured on a single H100, while HunyuanVideo and Mochi 1 are evaluated using 8 H100s. See Appendix B for more details.



Figure 6: Examples of videos generated by Radial Attention and the original Wan2.1-14B in the default video length. Radial Attention mirrors the video quality of the original model.

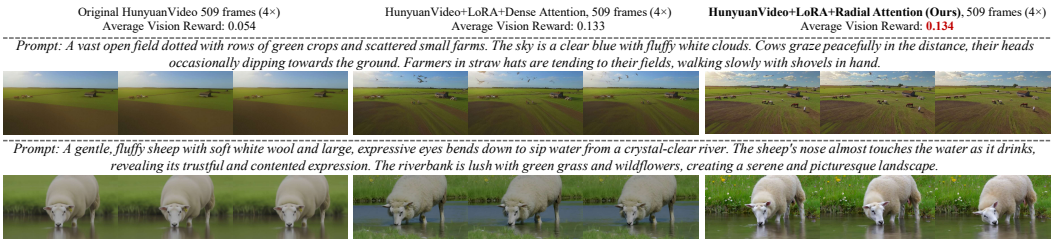


Figure 7: Visual comparison of HunyuanVideo with 4× length extension (509 frames). LoRA fine-tuned models using Radial Attention achieve higher vision rewards, outperforming dense attention baselines, while achieving a 3.7× speedup and reducing tuning costs by 4.4×.

5.2 Main Results

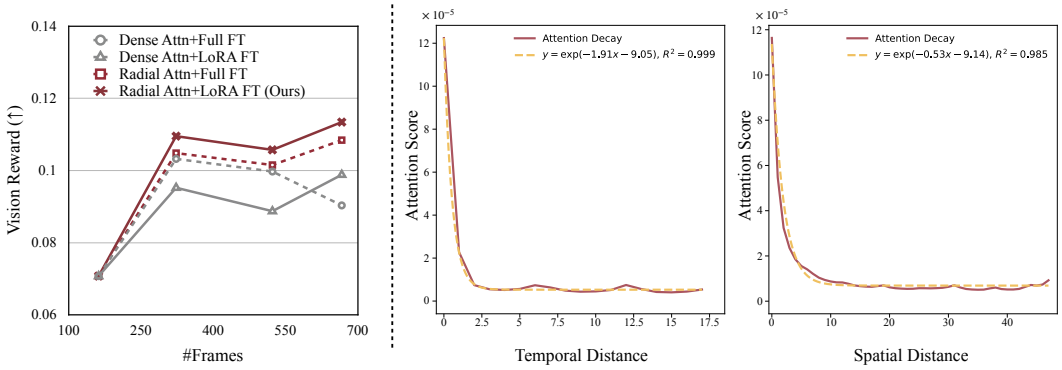
Training-free inference acceleration. Table 1 presents a quantitative comparison of Radial Attention against three strong sparse attention baselines on HunyuanVideo [1] and Wan2.1-14B [7] at their default generation lengths. Corresponding visual results are provided in Figure 6 and Appendix C.1. Under the same compute budget (measured in PFLOPs), Radial Attention preserves the video quality of dense attention while consistently outperforming STA and PA on similarity metrics (PSNR, SSIM, LPIPS), and matches the quality of SVG. While PA shares a similar $\mathcal{O}(n \log n)$ complexity with our design, it ignores the spatio-temporal locality inherent in video data, making it suboptimal in practice.

Regarding efficiency, we adopt the same system optimization used in SVG [8]. Specifically, on a single H100, our Radial Attention achieves 1.9× and 1.8× end-to-end speedups for HunyuanVideo and Wan 2.1, respectively, matching the theoretical compute budget savings (1.8× and 1.7× fewer PFLOPs). Although STA yields slightly higher speedup by using FlashAttention-3 (FA-3) [37], it suffers from noticeably degraded visual quality. Our current implementation uses FA2 [67]. Upgrading to FA3 is orthogonal to our algorithm and is left as future work.

Long video generation. Table 2 provides results for video generation at 2× and 4× the original lengths, with visualizations available in Figure 7 and Appendix C.2. For Wan2.1-14B, only 2× extrapolation is reported due to its significantly higher computational and memory costs. To ensure fairness, all sparse attention baselines use similar sparsity ratios.

When generating longer videos, the original models without further tuning exhibit significant quality degradation, especially for 4× video length extension. While RIFLEx improves performance at 2× length extrapolation, its quality deteriorates beyond that, indicating limited extension capability. Spatial and temporal sparse attentions suffer from limited reception fields; on the other hand, LongLoRA and PA, though with a global reception field, fail to capture spatial-temporal correlations, resulting in degraded quality. Interestingly, PA exhibits a large gain in Vision Reward after fine-tuning, suggesting that its original sparse pattern misaligns with the pre-trained attention distribution. Fine-tuning allows the model to adapt to the imposed attention sparsity, improving alignment and quality. SANA, which replaces softmax attention with linear attention, requires massive retraining and fails under fine-tuning-based video length extension. In contrast, Radial Attention achieves quality on par with LoRA fine-tuned dense attention models. Notably, it even slightly improves the Vision Reward over the pre-trained model at the default video length.

https://doi.org/10.52202/085713-0568



(a) LoRA Effectiveness Results (b) Regression Analysis

Figure 8: **(a)** Radial Attention outperforms dense attention in generation quality. When combined with LoRA, it further improve the quality while significantly reducing training costs. **(b)** We model the attention decay curves using the exponential function $y = \exp(-ax + b)$. It fits the data well, achieving $R^2 > 0.985$.

Thanks to $\mathcal{O}(n \log n)$ complexity, Radial Attention delivers substantial inference and training speedups over dense attention, as detailed in Table 2 and Figure 2. For instance, when generating 4× longer videos, we can save up to 4.4× training costs and get up to 3.7× inference speedup.

Compatibility with existing LoRAs. A key advantage of Radial Attention is its seamless compatibility with pre-trained task-specific LoRAs (*e.g.*, artistic style transfer). We observe that Radial Attention is compatible with existing style LoRAs in both the default length settings, and the longer video settings by directly merging the LoRA weights of Radial Attention in long videos and the style LoRAs. Visualizations and further analysis can be found in Appendix C.3.

5.3 Ablation Study & Analyses

Effectiveness of Low-Rank Adaptation. Figure 8(a) compares Vision Reward between full fine-tuning and LoRA as video length increases. For dense attention, LoRA fine-tuning lags behind full fine-tuning until 4× length extension. However, with our proposed Radial Attention, LoRA fine-tuning matches or even outperforms full fine-tuning, suggesting that Radial Attention not only scales better computationally, but also makes the model easier to adapt to longer-video generation.

Attention error. We evaluate the average attention output Mean Squared Error (MSE, lower is better) of Radial Attention on Wan2.1-14B, comparing it to SVG [8] and STA [47]. Radial Attention achieves an MSE of 3.9×10^{-3}, significantly lower than 4.4×10^{-3} for SVG and 1.5×10^{-2} for STA, demonstrating the effectiveness of our mask in preserving attention fidelity.

Regression results. We perform regression analysis using the model $y = \exp(-ax + b)$ to fit the average attention decay curves in Figure 4. As illustrated in Figure 8, the fitted curves achieve an R^2 value of over 0.985, indicating that the exponential functions can well model the decay.

More ablations on Radial Attention design choice. Please refer to Appendix D for more details.

6 Conclusion & Discussion

In this work, we propose Radial Attention, an $\mathcal{O}(n \log n)$ sparse attention for efficient long video generation. We observe *Spatiotemporal Energy Decay* in video diffusion models, which motivates a unified attention pattern with sub-quadratic complexity. At default video length, Radial Attention achieves up to a 1.9× speedup with high fidelity. For videos up to 4× longer, Radial Attention preserves quality and delivers up to 4.4× and 3.7× speedups in training and inference, respectively, with minimal LoRA fine-tuning. This work contributes toward scalable, high-quality video generation and offers a foundation for efficient long-range attention in broader sequence modeling tasks.

Limitations. The assumption of exponential decay for attention scores (Equation 3) simplifies the complex spatiotemporal dependencies in natural video data. While aiding theoretical analysis, future work could improve efficiency and performance by more deeply understanding and modeling the underlying data structure. As shown in Equation 6, our method still has quadratic complexity with respect to resolution. Future work should explore more efficient attention mechanisms and pre-training strategies, as in NSA [75] and MoBA [76], to better support long, high-resolution videos.

Acknowledgments

We thank MIT-IBM Watson AI Lab, National Science Foundation, Hyundai, and Amazon for supporting this research.

References

[1] 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, 2, 3, 4, 7, 9, 24, 26

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 2, 3

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3, 4

[4] OpenAI. Sora: Video generation models as world simulators, 2024. 2, 3

[5] 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. 2, 3, 4

[6] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024. 2, 3

[7] Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 2, 3, 4, 7, 9, 24, 26

[8] Haocheng Xi, Shuo Yang, Yilong Zhao, Chenfeng Xu, Muyang Li, Xiuyu Li, Yujun Lin, Han Cai, Jintao Zhang, Dacheng Li, et al. Sparse videogen: Accelerating video diffusion transformers with spatial-temporal sparsity. In *ICML*, 2025. 2, 3, 4, 6, 7, 9, 10, 25, 26

[9] Hongjie Wang, Chih-Yao Ma, Yen-Cheng Liu, Ji Hou, Tao Xu, Jialiang Wang, Felix Juefei-Xu, Yaqiao Luo, Peizhao Zhang, Tingbo Hou, et al. Lingen: Towards high-resolution minute-length text-to-video generation with linear computational complexity. In *CVPR*, 2025. 2, 3

[10] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, et al. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. *ICLR*, 2025. 2, 3, 8

[11] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022. 2, 3, 7

[12] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024. 3

[13] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *NeurIPS*, 2022. 3

[14] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 3

[15] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 3

 https://doi.org/10.52202/085713-0568

[16] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. In *ICLR*, 2024. 3

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 3

[18] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 3

[19] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 3

[20] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024. 3

[21] Xiangyu Peng, Zangwei Zheng, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, Wenjun Li, Yuhui Wang, Anbang Ye, Gang Ren, Qianran Ma, Wanying Liang, Xiang Lian, Xiwen Wu, Yuting Zhong, Zhuangyan Li, Chaoyu Gong, Guojun Lei, Leijun Cheng, Limin Zhang, Minghao Li, Ruijie Zhang, Silan Hu, Shijie Huang, Xiaokang Wang, Yuanheng Zhao, Yuqi Wang, Ziang Wei, and Yang You. Open-sora 2.0: Training a commercial-level video generation model in $200k. *arXiv preprint arXiv:2503.09642*, 2025. 3

[22] Genmo Team. Mochi 1. https://github.com/genmoai/models, 2024. 3, 4, 7, 24, 26

[23] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. *CVPR*, 2025. 3

[24] Jiachen Li, Weixi Feng, Tsu-Jui Fu, Xinyi Wang, S Basu, Wenhu Chen, and William Yang Wang. T2v-turbo: Breaking the quality bottleneck of video consistency model with mixed reward feedback. In *NeurIPS*, 2024. 3

[25] Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free. In *CVPR*, 2024. 3

[26] Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *CVPR*, 2025. 3

[27] Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-diffusion: Quantizing diffusion models. In *ICCV*, 2023. 3

[28] Tianchen Zhao, Tongcheng Fang, Enshu Liu, Wan Rui, Widyadewi Soedarmadji, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, Huazhong Yang, et al. Vidit-q: Efficient and accurate quantization of diffusion transformers for image and video generation. In *ICLR*, 2025. 3

[29] Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Junxian Guo, Xiuyu Li, Enze Xie, Chenlin Meng, Jun-Yan Zhu, and Song Han. Svdquant: Absorbing outliers by low-rank component for 4-bit diffusion models. In *ICLR*, 2025. 3

[30] Muyang Li, Tianle Cai, Jiaxin Cao, Qinsheng Zhang, Han Cai, Junjie Bai, Yangqing Jia, Ming-Yu Liu, Kai Li, and Song Han. Distrifusion: Distributed parallel inference for high-resolution diffusion models. In *CVPR*, 2024. 3

[31] Jiarui Fang, Jinzhe Pan, Jiannan Wang, Aoyu Li, and Xibo Sun. Pipefusion: Patch-level pipeline parallelism for diffusion transformers inference. *arXiv preprint arXiv:2405.14430*, 2024. 3

[32] Jiarui Fang, Jinzhe Pan, Xibo Sun, Aoyu Li, and Jiannan Wang. xdit: an inference engine for diffusion transformers (dits) with massive parallelism. *arXiv preprint arXiv:2411.01738*, 2024. 3

[33] Jintao Zhang, Jia Wei, Pengle Zhang, Jun Zhu, and Jianfei Chen. Sageattention: Accurate 8-bit attention for plug-and-play inference acceleration. In *ICLR*, 2025. 3

[34] Jintao Zhang, Haofeng Huang, Pengle Zhang, Jia Wei, Jun Zhu, and Jianfei Chen. Sageattention2: Efficient attention with thorough outlier smoothing and per-thread int4 quantization. In *ICML*, 2025. 3

[35] Jintao Zhang, Jia Wei, Pengle Zhang, Xiaoming Xu, Haofeng Huang, Haoxu Wang, Kai Jiang, Jun Zhu, and Jianfei Chen. Sageattention3: Microscaling fp4 attention for inference and an exploration of 8-bit training. *arXiv preprint arXiv:2505.11594*, 2025. 3

[36] Jintao Zhang, Xiaoming Xu, Jia Wei, Haofeng Huang, Pengle Zhang, Chendong Xiang, Jun Zhu, and Jianfei Chen. Sageattention2++: A more efficient implementation of sageattention2. *arXiv preprint arXiv:2505.21136*, 2025. 3

[37] Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *NeurIPS*, 2024. 3, 9

[38] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *ICLR*, 2024. 3, 6

[39] Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. Longlora: Efficient fine-tuning of long-context large language models. In *ICLR*, 2024. 3, 7, 8

[40] Huiqiang Jiang, Yucheng Li, Chengruidong Zhang, Qianhui Wu, Xufang Luo, Surin Ahn, Zhenhua Han, Amir Abdi, Dongsheng Li, Chin-Yew Lin, et al. Minference 1.0: Accelerating pre-filling for long-context llms via dynamic sparse attention. *NeurIPS*, 2024. 3, 7

[41] Jintao Zhang, Chendong Xiang, Haofeng Huang, Jia Wei, Haocheng Xi, Jun Zhu, and Jianfei Chen. Spargeattn: Accurate sparse attention accelerating any model inference. In *ICML*, 2025. 3

[42] Jiaming Tang, Yilong Zhao, Kan Zhu, Guangxuan Xiao, Baris Kasikci, and Song Han. Quest: Query-aware sparsity for efficient long-context llm inference. In *ICML*, 2024. 3

[43] Ruyi Xu, Guangxuan Xiao, Haofeng Huang, Junxian Guo, and Song Han. Xattention: Block sparse attention with antidiagonal scoring. In *ICML*, 2025. 3, 7

[44] Shang Yang, Junxian Guo, Haotian Tang, Qinghao Hu, Guangxuan Xiao, Jiaming Tang, Yujun Lin, Zhijian Liu, Yao Lu, and Song Han. Lserve: Efficient long-sequence llm serving with unified sparse attention. In *MLSys*, 2025. 3, 7

[45] Lida Chen, Dong Xu, Chenxin An, Xintao Wang, Yikai Zhang, Jiangjie Chen, Zujie Liang, Feng Wei, Jiaqing Liang, Yanghua Xiao, et al. Powerattention: Exponentially scaling of receptive fields for effective sparse attention. *arXiv preprint arXiv:2503.03588*, 2025. 3, 8

[46] Guangxuan Xiao, Jiaming Tang, Jingwei Zuo, Junxian Guo, Shang Yang, Haotian Tang, Yao Fu, and Song Han. Duoattention: Efficient long-context llm inference with retrieval and streaming heads. In *ICLR*, 2025. 3

[47] Peiyuan Zhang, Yongqi Chen, Runlong Su, Hangliang Ding, Ion Stoica, Zhenghong Liu, and Hao Zhang. Fast video generation with sliding tile attention. *arXiv preprint arXiv:2502.04507*, 2025. 3, 4, 7, 10, 24, 26

[48] Yifei Xia, Suhan Ling, Fangcheng Fu, Yujie Wang, Huixia Li, Xuefeng Xiao, and Bin Cui. Training-free and adaptive sparse attention for efficient long video generation. *arXiv preprint arXiv:2502.21079*, 2025. 3

[49] Xin Tan, Yuetao Chen, Yimin Jiang, Xing Chen, Kun Yan, Nan Duan, Yibo Zhu, Daxin Jiang, and Hong Xu. Dsv: Exploiting dynamic sparsity to accelerate large-scale video dit training. *arXiv preprint arXiv:2502.07590*, 2025. 3

[50] Min Zhao, Guande He, Yixiao Chen, Hongzhou Zhu, Chongxuan Li, and Jun Zhu. Riflex: A free lunch for length extrapolation in video diffusion transformers. In *ICML*, 2025. 3, 7, 8

[51] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024. 3, 8

[52] Karan Dalal, Daniel Koceja, Gashon Hussein, Jiarui Xu, Yue Zhao, Youjin Song, Shihao Han, Ka Chun Cheung, Jan Kautz, Carlos Guestrin, et al. One-minute video generation with test-time training. *arXiv preprint arXiv:2504.05298*, 2025. 3

[53] Lvmin Zhang and Maneesh Agrawala. Packing input frame context in next-frame prediction models for video generation. *arXiv preprint arXiv:2504.12626*, 2025. 3

[54] Songhua Liu, Weihao Yu, Zhenxiong Tan, and Xinchao Wang. Linfusion: 1 gpu, 1 minute, 16k image. *arXiv preprint arXiv:2409.02097*, 2024. 3

[55] Dongchen Han, Xuran Pan, Yizeng Han, Shiji Song, and Gao Huang. Flatten transformer: Vision transformer using focused linear attention. In *ICCV*, 2023. 3

 https://doi.org/10.52202/085713-0568

[56] Songhua Liu, Zhenxiong Tan, and Xinchao Wang. Clear: Conv-like linearization revs pre-trained diffusion transformers up. *arXiv preprint arXiv:2412.16112*, 2024. 3

[57] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 5156–5165. PMLR, 2020. 3

[58] Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. In *Forty-first International Conference on Machine Learning*, 2024. 3

[59] Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *Forty-first International Conference on Machine Learning*, 2024. 3

[60] Han Cai, Junyan Li, Muyan Hu, Chuang Gan, and Song Han. Efficientvit: Lightweight multi-scale attention for high-resolution dense prediction. In *ICCV*, 2023. 3

[61] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020. 3

[62] Zhenhai Zhu and Radu Soricut. H-transformer-1d: Fast one-dimensional hierarchical attention for sequences, 2021. 3

[63] Zhanpeng Zeng, Sourav Pal, Jeffery Kline, Glenn M Fung, and Vikas Singh. Multi resolution analysis (mra) for approximate self-attention, 2022. 4

[64] Yanming Kang, Giang Tran, and Hans De Sterck. Fast multipole attention: A divide-and-conquer attention mechanism for long sequences, 2024. 4

[65] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32, 2019. 4

[66] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *NeurIPS*, 2022. 4, 7

[67] Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *ICLR*, 2024. 4, 8, 9

[68] Muyang Li, Ji Lin, Chenlin Meng, Stefano Ermon, Song Han, and Jun-Yan Zhu. Efficient spatially sparse inference for conditional gans and diffusion models. In *NeurIPS*, 2022. 7

[69] Jiazheng Xu, Yu Huang, Jiale Cheng, Yuanming Yang, Jiajun Xu, Yuan Wang, Wenbo Duan, Shen Yang, Qunlin Jin, Shurun Li, et al. Visionreward: Fine-grained multi-dimensional human preference learning for image and video generation. *arXiv preprint arXiv:2412.21059*, 2024. 7, 25

[70] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 7

[71] Dian Zheng, Ziqi Huang, Hongbo Liu, Kai Zou, Yinan He, Fan Zhang, Yuanhan Zhang, Jingwen He, Wei-Shi Zheng, Yu Qiao, and Ziwei Liu. Vbench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness. *arXiv preprint arXiv:2311.17982*, 2025. 7

[72] Zihao Ye, Lequn Chen, Ruihang Lai, Wuwei Lin, Yineng Zhang, Stephanie Wang, Tianqi Chen, Baris Kasikci, Vinod Grover, Arvind Krishnamurthy, and Luis Ceze. Flashinfer: Efficient and customizable attention engine for llm inference serving. In *MLSys*, 2025. 8

[73] Junxian Guo, Haotian Tang, Shang Yang, Zhekai Zhang, Zhijian Liu, and Song Han. Block Sparse Attention. `https://github.com/mit-han-lab/Block-Sparse-Attention`, 2024. 8

[74] Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. In *ICLR*, 2025. 8

[75] Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, YX Wei, Lean Wang, Zhiping Xiao, et al. Native sparse attention: Hardware-aligned and natively trainable sparse attention. *arXiv preprint arXiv:2502.11089*, 2025. 10

[76] Enzhe Lu, Zhejun Jiang, Jingyuan Liu, Yulun Du, Tao Jiang, Chao Hong, Shaowei Liu, Weiran He, Enming Yuan, Yuzhi Wang, et al. Moba: Mixture of block attention for long-context llms. *arXiv preprint arXiv:2502.13189*, 2025. 10

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See abstract and introduction. We clearly state what we do in the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 6.

https://doi.org/10.52202/085713-0568

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: See Section 4 and Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Section 5.1 and Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: See Section 5.1 and Appendix B. Our code is released at https://github.com/mit-han-lab/radial-attention

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

https://doi.org/10.52202/085713-0568

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Section 5.1 and Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not report error bars as each experiment is resource-intensive. All reported results are from single runs, but reproducible.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section 5.1 and Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We read the NeurIPS Code of Ethics. Our paper does not have an ethics issue.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Appendix E.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We will explicitly specify the usage permission of our code and models with proper licenses.

    Guidelines:

https://doi.org/10.52202/085713-0568

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: See Appendix F.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: A comprehensive README will be provided when we release our code and models.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this paper does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

https://doi.org/10.52202/085713-0568

A Derivations and Proofs

A.1 Complexity

In this section, we provide the detailed complexity derivation in Equation 5, which scales as $\mathcal{O}(n \log n)$. Since the computational cost of masked attention is proportional to the number of zeros in the attention mask $\tilde{\boldsymbol{M}}$, we only need to derive an $\mathcal{O}(n \log n)$ upper bound for the latter.

Central band&attention sink. Firstly, recall from Figure 5(a) that we apply dense attention on these frame-to-frame attention blocks within the central band and attention sink. The attention sink refers to the pattern that every token attends to all tokens in the first frame. Using the same notation as in Section 4, where n is the total number of tokens, s is the number of tokens per frame, and f is the number of frames (so $n = fs$), we define the attention mask for this region as $\tilde{\boldsymbol{M}}^{(1)} \in \{-\infty, 0\}^{f \times f \times s \times s}$:

$$\tilde{M}^{(1)}_{i,j,k,l} = \begin{cases} 0, & \text{if } |i-j| \le 1 \text{ or } j = 0 \\ -\infty. & \text{otherwise} \end{cases} \tag{8}$$

Here, $\tilde{M}^{(1)}_{i,j,k,l}$ indicates whether the k-th token in frame i is allowed to attend to the l-th token in frame j, with 0 denoting allowed attention and $-\infty$ indicating disallowed attention. Since the attention sink spans f blocks and the central band includes at most $3f$ blocks, the total number of nonzero entries in this region is bounded by:

$$\#\text{zeros in } \tilde{\boldsymbol{M}}^{(1)} \le 4 \cdot f \cdot s^2 = 4s^2 f. \tag{9}$$

Bands with diagonal width ≥ 1. The second part is those bands with diagonal width ≥ 1, except the central band. The mask for this region can be defined as $\boldsymbol{M}^{(2)} \in \{-\infty, 0\}^{f \times f \times s \times s}$:

$$\tilde{M}^{(2)}_{i,j,k,l} = \begin{cases} 0, & \text{if } 2^{\lfloor \log_2 \max(|i-j|,1) \rfloor} \le s \text{ and } |k-l| + 1 \le \frac{s}{2^{\lfloor \log_2 \max(|i-j|,1) \rfloor}} \\ -\infty. & \text{otherwise} \end{cases} \tag{10}$$

Thus, since there are at most $\lfloor \log_2 s \rfloor$ bands in this region, the number of zeros in these bands is bounded by:

$$\#\text{zeros in } \tilde{\boldsymbol{M}}^{(2)} \le \sum_{r=1}^{\lfloor \log_2 s \rfloor} \underbrace{2^{r+1} sn}_{\text{area bound for band } \pm r} \cdot \underbrace{2/2^r}_{\text{compute density bound of band } \pm r} \tag{11}$$

$$\le \sum_{r=1}^{\lfloor \log_2 s \rfloor} \frac{2^{r+2} s^2 f}{2^r} \tag{12}$$

$$= 4s^2 f \cdot \lfloor \log_2 s \rfloor. \tag{13}$$

Bands with diagonal width < 1. The last part is those bands with $\frac{s}{2^{\lfloor \log_2 \max(|i-j|,1) \rfloor}} < 1$, where we reduce the frequency of diagonals. The mask for this region $\boldsymbol{M}^{(3)} \in \{-\infty, 0\}^{f \times f \times s \times s}$ is given by:

$$\tilde{M}^{(3)}_{i,j,k,l} = \begin{cases} 0, & \text{if } |i-j| \bmod \lceil \frac{2^{\lfloor \log_2 \max(|i-j|,1) \rfloor}}{s} \rceil = 0 \text{ and } k = l \\ -\infty. & \text{otherwise} \end{cases} \tag{14}$$

Since there are at most $(\lceil \log_2 f \rceil - 1) - (\lfloor \log_2 s \rfloor + 1)$ bands satisfying this condition, we have the number of zeros in these bands bounded by:

$$\#\text{zeros in } \tilde{\boldsymbol{M}}^{(3)} \le \sum_{r=\lfloor \log_2 s \rfloor + 1}^{\lceil \log_2 f \rceil - 1} \underbrace{2^{\lfloor \log_2 s \rfloor + 1}}_{\text{number of diagonals}} \cdot \underbrace{n}_{\text{area bound of each diagonal}} \tag{15}$$

$$\le (\lfloor \log_2 f \rfloor - \lfloor \log_2 s \rfloor) 4s^2 f. \tag{16}$$

Combining Equation 9, Equation 13, and Equation 16 together, we have the aggregate upper bound of the number of zeros in Radial Attention's mask:

$$\text{\# of zeros in } \tilde{\boldsymbol{M}} \leq 4s^2 f \cdot \lfloor \log_2 f \rfloor \leq 4s \cdot n(\log_2 n - \log_2 s), \tag{17}$$

which scales $\mathcal{O}(n \log n)$ with the number of frames f for long video generation.

A.2 Error Bound

The design of Radial Attention is inspired by the spatial-temporal structure in video. In this section, we formulate this intuition by theoretically bounding the asymptotic approximation error of Radial Attention with respect to the decay speed of the attention value in the spatial and temporal dimensions.

We focus on bounding the approximation error of a single row of the attention matrix. We fix a reference query token at position k_0 of frame i_0, and write the unnormalized row of the attention matrix as

$$a_{j,l} \;=\; \exp(\boldsymbol{Q}_{i_0 s+k_0} \boldsymbol{K}_{js+l}^{\top}).$$

where $\boldsymbol{Q}_{i_0 s+k_0}$ refers to the query vector at position k_0 in frame i_0, $\boldsymbol{K}_{js+l}$ refers to the key vector at position l in frame j, and s refers to the number of tokens per frame.

Assumptions

(A1) **Relative exponential decay.** To capture the intuition that the closer frames have a stronger correlation and each token typically attends to tokens in other frames at similar spatial positions, we assume there exist $C_{\text{rel}} > 0$ and $(\alpha, \beta) > 0$ such that

$$0 \;\leq\; a_{j,l} \;\leq\; C_{\text{rel}}\, e^{-\alpha|j-i_0|-\beta|l-k_0|}\, a_0, \quad a_0 := a_{i_0,k_0} > 0.$$

where α characterizes the temporal decay rate and β characterizes the spatial decay rate.

(A2) **Infinite temporal grid & finite spatial grid.** To conduct asymptotic analysis, we let $j \in \mathbb{Z}$ (temporal) but keep $l \in \{1, \ldots, s\}$ (spatial). Extending j to $\mathbb{Z}$ only enlarges the sums we bound.

Notation

$$Z := \sum_{j\in\mathbb{Z}} \sum_{l=1}^{s} a_{j,l}, \qquad Z_{\text{keep}} := \sum_{(j,l)\,:\,\tilde{\boldsymbol{M}}_{i_0,j,k_0,l}=0} a_{j,l}, \qquad Z_{\text{out}} := Z - Z_{\text{keep}}.$$

Exact and masked softmax rows: $p_{j,l} = a_{j,l}/Z$, $\tilde{p}_{j,l} = a_{j,l}\mathbf{1}_{\{\tilde{\boldsymbol{M}}=0\}}/Z_{\text{keep}}$. The total variation error can be calculated as follows by standard algebraic argument,

$$\left\|\tilde{p} - p\right\|_1 \;=\; 2\,\frac{Z_{\text{out}}}{Z}. \tag{1}$$

Because a_0 itself is in the sum, $Z \geq a_0$. Hence

$$\frac{Z_{\text{out}}}{Z} \;\leq\; \frac{Z_{\text{out}}}{a_0}. \tag{2}$$

Mask geometry For a temporal offset $\Delta t := |j - i_0| \geq 0$, define the bandwidth

$$w(\Delta t) \;:=\; \frac{s}{2^{\lfloor \log_2 \max(\Delta t, 1) \rfloor}} \;\in \{1, 2, 4, \ldots, s\}.$$

The mask keeps a spatial index l iff $|l - k_0| \leq w(\Delta t)$ and the frame is one of the sub-sampled frames; otherwise $\tilde{\boldsymbol{M}}_{i_0,j,k_0,l} = -\infty$.

Two kinds of approximation errors, therefore, appear:

(i) Spatial tails inside kept frames

https://doi.org/10.52202/085713-0568

For each Δt, the discarded spatial part satisfies

$$\sum_{d>w(\Delta t)} e^{-\beta d} \;\le\; \frac{e^{-\beta(w(\Delta t)+1)}}{1-e^{-\beta}}.$$

Because $w(\Delta t) \ge \frac{s}{2}$ when $\Delta t \le s$,

$$T_1 := 2C_{\text{rel}}a_0 \sum_{\Delta t\ge 0} e^{-\alpha\Delta t} \sum_{d>w(\Delta t)} e^{-\beta d} \;\le\; \frac{4C_{\text{rel}}a_0}{(1-e^{-\alpha})(1-e^{-\beta})}\, e^{-\beta\left(\frac{s}{2}+1\right)}. \tag{3}$$

(ii) Frames skipped by the subsampling rule

For $\Delta t > s$, only every $K(\Delta t) = \left\lceil 2^{\lfloor \log_2 \Delta t \rfloor}/s \right\rceil$ frame is kept; the remainder contributes

$$T_2 := 2C_{\text{rel}}a_0 \frac{1+e^{-\beta}}{1-e^{-\beta}} \sum_{\Delta t>s} e^{-\alpha\Delta t} \;\le\; \frac{2C_{\text{rel}}a_0\,(1+e^{-\beta})}{(1-e^{-\beta})(1-e^{-\alpha})}\, e^{-\alpha(s+1)}. \tag{4}$$

Total variation error Combine all equations above:

$$\boxed{\left\|\tilde{p}-p\right\|_1 \;\le\; C_{\text{rel}}\left[\frac{8\,e^{-\beta\left(\frac{s}{2}+1\right)}}{(1-e^{-\alpha})(1-e^{-\beta})} \;+\; 4\,\frac{1+e^{-\beta}}{1-e^{-\beta}}\,\frac{e^{-\alpha(s+1)}}{1-e^{-\alpha}}\right] = O(C_{\text{rel}}e^{-\min\{\beta/2,\alpha\}s})}.$$

This characterizes how the decay rates affect the approximation error.

B Additional Implementation Details

In terms of our LoRA fine-tuning for longer-video generation, we fine-tune HunyuanVideo [1] and Mochi 1 [22] at a global batch size of 1 with sequence parallelism, and train Wan 2.1 [7] with a global batch size of 8. All tuning experiments are conducted on 8 H100 GPUs. During training, we keep the first two DiT blocks with dense attention. Since there are 60, 48, and 40 blocks for HunyuanVideo, Mochi 1, and Wan2.1-14B, this only incurs negligible overhead. We train HunyuanVideo for 2× and 4× length video generation for 2400 and 1200 steps, respectively. We train Mochi 1 for 5000 steps for both 2× and 4× length video generation. We train Wan2.1-14B for 2500 steps for 2× length video generation. The LoRA rank is 128 for all training tasks.

C Visualization of the generated videos

In this section we compare Radial Attention against various baselines in video quality, and list our speedup in both training and inference.

C.1 Default Video Length

We provide a visual comparison between the original dense attention, STA [47], and our Radial Attention on HunyuanVideo [1] and Wan2.1-14B [7]. We conduct experiments under 768p, 117 frames settings for HunyuanVideo, and 768p, 69 frames settings for Wan2.1-14B. As shown in Figure A and Figure B, Radial Attention has higher PSNR compared to STA [47], effectively maintaining the high fidelity of the original videos.

C.2 Longer-video Length

We provide a visual comparison between the aforementioned baselines and Radial Attention on HunyuanVideo [1], Mochi 1 [22], and Wan2.1-14B [7]. We conduct experiments under the default resolution settings, which are 720p for HunyuanVideo and Wan2.1-14B, and 480p for Mochi 1. Moreover, we generate videos at 4× longer length for HunyuanVideo (21 seconds, 509 frames) and

Table A: We compare Radial Attention against another $\mathcal{O}(n \log n)$ attention baseline, Harmonic Series (HS). Radial Attention consistently outperforms it across all metrics.

Model	Method	PSNR (↑)	SSIM (↑)	LPIPS (↓)	VisionReward (↑)
HunyuanVideo (117 frames)	HS	27.0	0.881	0.119	0.136
	Ours	**27.3**	**0.886**	**0.114**	**0.139**

Table B: Ablation on the number of initial full-attention (warmup) steps for default-length video generation of Wan2.1-14B model. The 12-step warmup achieves the best performance across all metrics.

Model	#Warmup Steps	PSNR (↑)	SSIM (↑)	LPIPS (↓)
Wan2.1-14B (69 frames)	0	12.8	0.486	0.522
	4	18.5	0.693	0.267
	8	21.7	0.778	0.183
	11	23.2	0.813	0.151
	12 (Ours)	**23.6**	**0.823**	**0.146**
	13	23.5	0.819	0.150

Mochi 1(22 seconds, 667 frames), and 2× longer length for Wan2.1-14B (10 seconds, 161 frames). We use Vision Reward [69] to evaluate the generated videos. Figure C, Figure D, and Figure E demonstrate that Radial Attention achieves the highest average Vision Reward score compared to the baselines, well preserving the video quality even at longer-video settings.

C.3 LoRA Compatibility Visual Results

As illustrated in Figure F, combining our extended-length LoRA with existing style LoRAs preserves visual quality while enabling longer-video generation. We observe that the content style generated by the merged LoRA exhibits subtle differences from the original LoRAs. This discrepancy is primarily attributed to the relatively small dataset used for training the extended-length LoRA, which may introduce a slight style bias that interacts with the style LoRA. We expect that training the length-extension LoRA on a more comprehensive dataset would help mitigate this issue.

D Ablations on Initial Dense-Attention Layers and Steps

We provide additional ablation studies to justify our design choices, including how many dense-attention timesteps and blocks we use to deliver the best video quality with the same computation budget, as well as the design of our $\mathcal{O}(n \log n)$ attention mask.

D.1 Ablation on Initial Dense-Attention Steps

For default-length video generation, we follow SVG [8] to apply full attention to the first 25% of timesteps (12 steps) as a warmup for all methods. We ablate this setting in Table B. For fair comparison, we match the overall computation of all settings by adjusting the sparsity of our Radial Attention mask. The 12-step warmup achieves the best video quality across all metrics.

For 4× longer video generation, we apply full attention to the first 2 steps as a warmup during inference. The impact of different warmup steps on HunyuanVideo is shown in Table C. Computation is matched across all configurations. Using 2 warmup steps achieves the highest Vision Reward.

D.2 Ablation on Initial Dense-Attention Layers

To better capture global information, we keep the first two layers as full attention during training. Table D reports results on HunyuanVideo when using 0, 1, 2, or 3 dense layers, under the same computation budget. Our choice of using 2 full-attention layers delivers the best video quality.

https://doi.org/10.52202/085713-0568

Table C: Ablation on the number of warmup steps for 4× longer video generation. Two warmup steps yield the best Vision Reward.

Model	#Warmup Steps	Vision Reward (↑)
HunyuanVideo (117 frames)	0	0.154
	1	0.160
	2 (Ours)	**0.163**
	3	0.157

Table D: Ablation on the number of initial full-attention (dense) layers during training. Using two full-attention layers yields the highest Vision Reward.

Model	#Dense Layers	Vision Reward (↑)
HunyuanVideo (117 frames)	0	0.139
	1	0.156
	2 (Ours)	**0.163**
	3	0.157

D.3 Comparison with Other $\mathcal{O}(n \log n)$ Sparsity Patterns

We additionally conduct an ablation study to validate the effectiveness of the sparsity pattern in our proposed $\mathcal{O}(n \log n)$ attention mask. Specifically, we compare Radial Attention with the Harmonic Series Decay Attention (HS), which features a computed diagonal width inversely proportional to its distance from the main diagonal. Table A presents quantitative results comparing Radial Attention with HS, demonstrating the superiority of Radial Attention.

E Broader Impacts

Radial Attention significantly reduces computational costs for video diffusion models, enabling longer-video generation with minimal fine-tuning efforts while maintaining quality. This paves the way for high-quality video creation tools for education and creative arts. Since Radial Attention accelerates self-attention to $\mathcal{O}(n \log n)$ complexity, it can accelerate video diffusion models and decrease energy consumption, leading to greener AI applications. This also helps the popularization of generative models. However, malicious users can misuse our method to create deepfakes and spread misinformation. The technology may also exacerbate the digital divide between those with and without access to the minimal necessary computational resources. To address these concerns, we advocate for responsible deployment, adherence to ethical standards, and the development of effective detection methods. We encourage the research community to continue advancing both efficient generation techniques and safeguards to ensure these powerful tools benefit society while minimizing potential harms. We will explicitly specify the usage permission of our code and models with proper licenses.

F License

Here, we show all the licenses for our used assets. Wan 2.1 [7], Mochi 1 [22], Diffusers, and STA [47] are under Apache-2.0 license. The license of HunyuanVideo [1] is here. SVG [8] and OpenVid-1M do not have an explicit license.

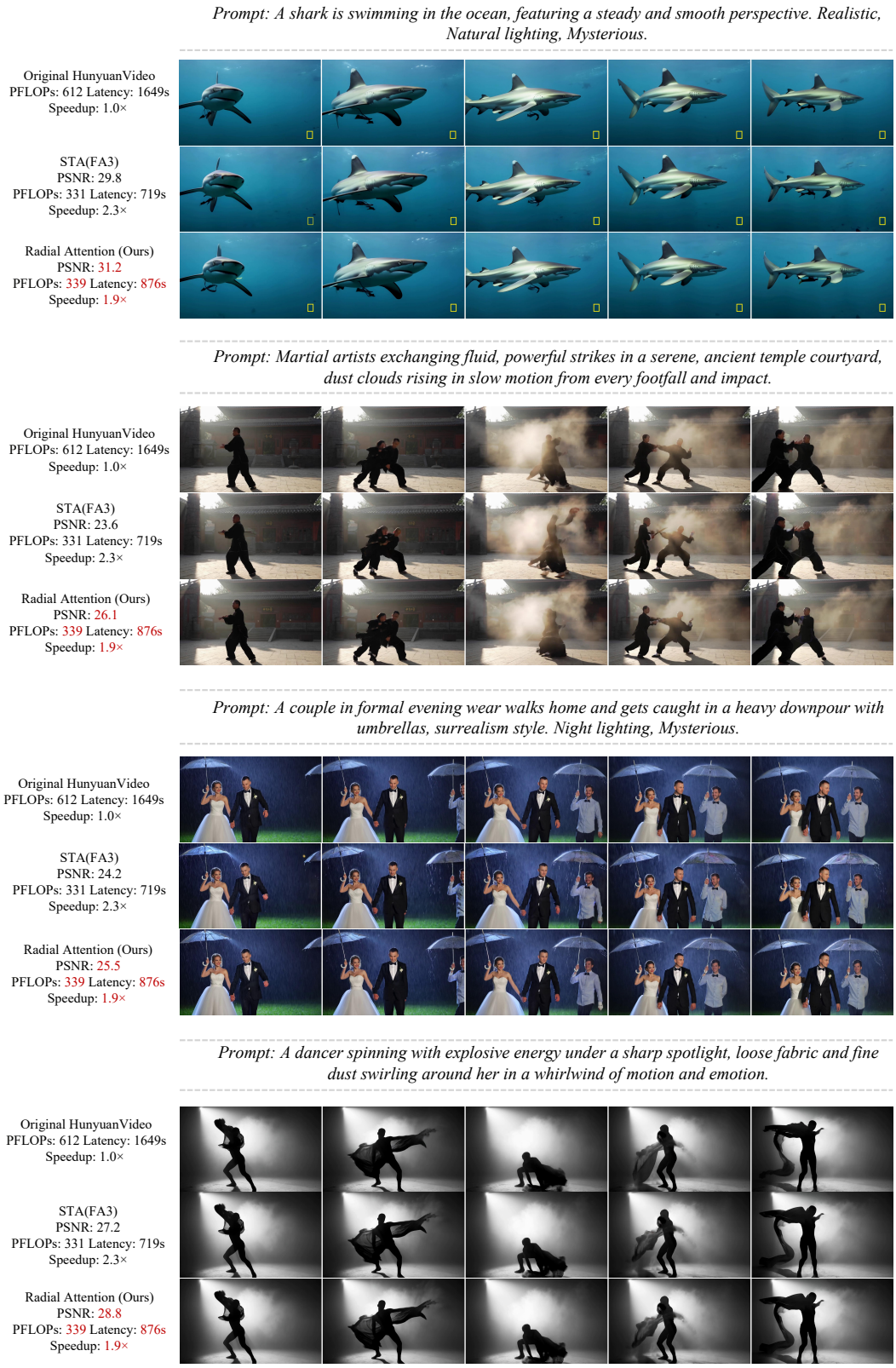


Figure A: Comparison of Dense Attention and Radial Attention on HunyuanVideo Text-to-Video generation at the default length (5 seconds, 117 frames, 768p).

https://doi.org/10.52202/085713-0568

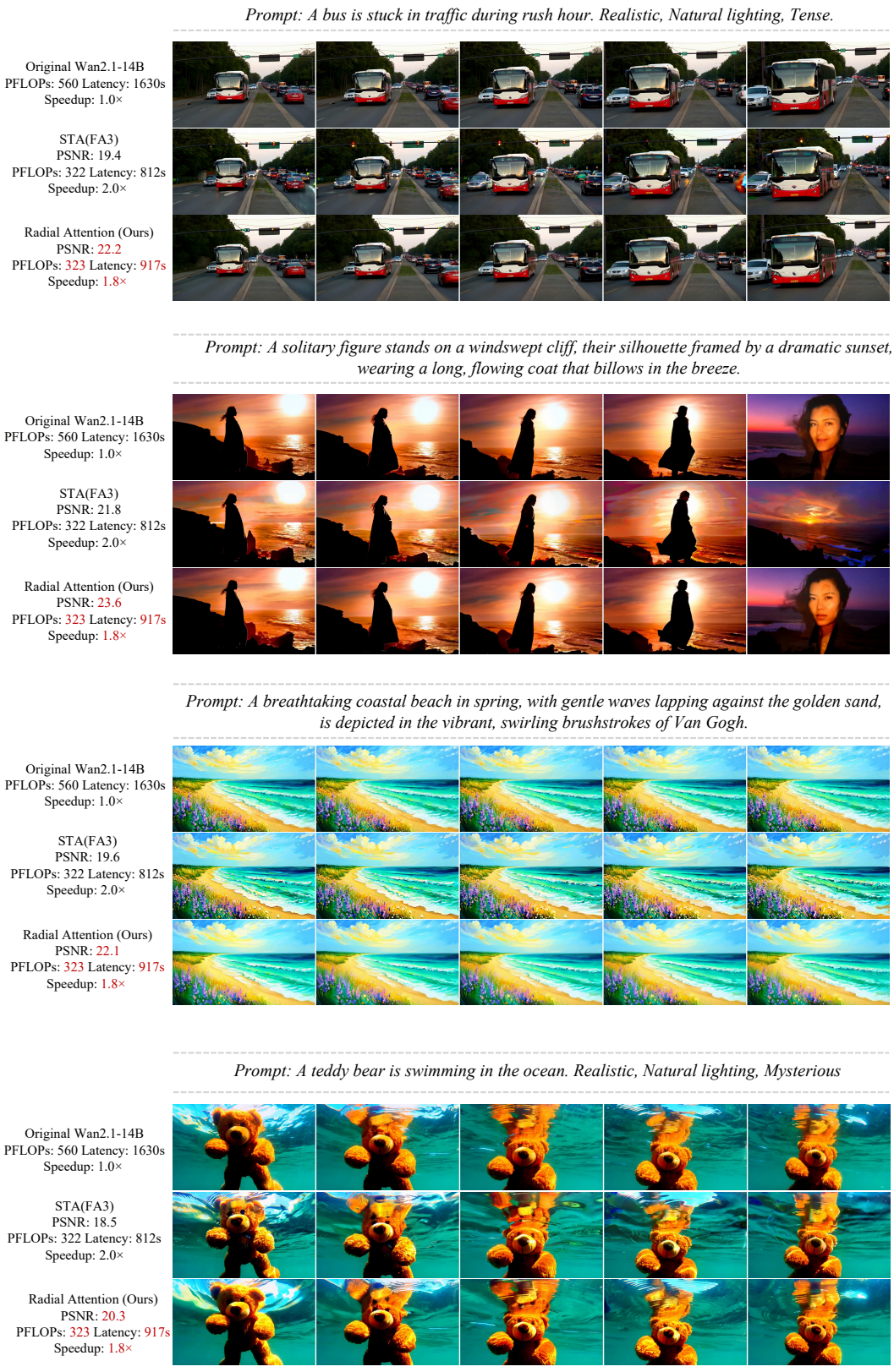


Figure B: Comparison of Dense Attention and Radial Attention on Wan2.1-14B Text-to-Video generation at the default length (4 seconds, 69 frames, 768p).

https://doi.org/10.52202/085713-0568

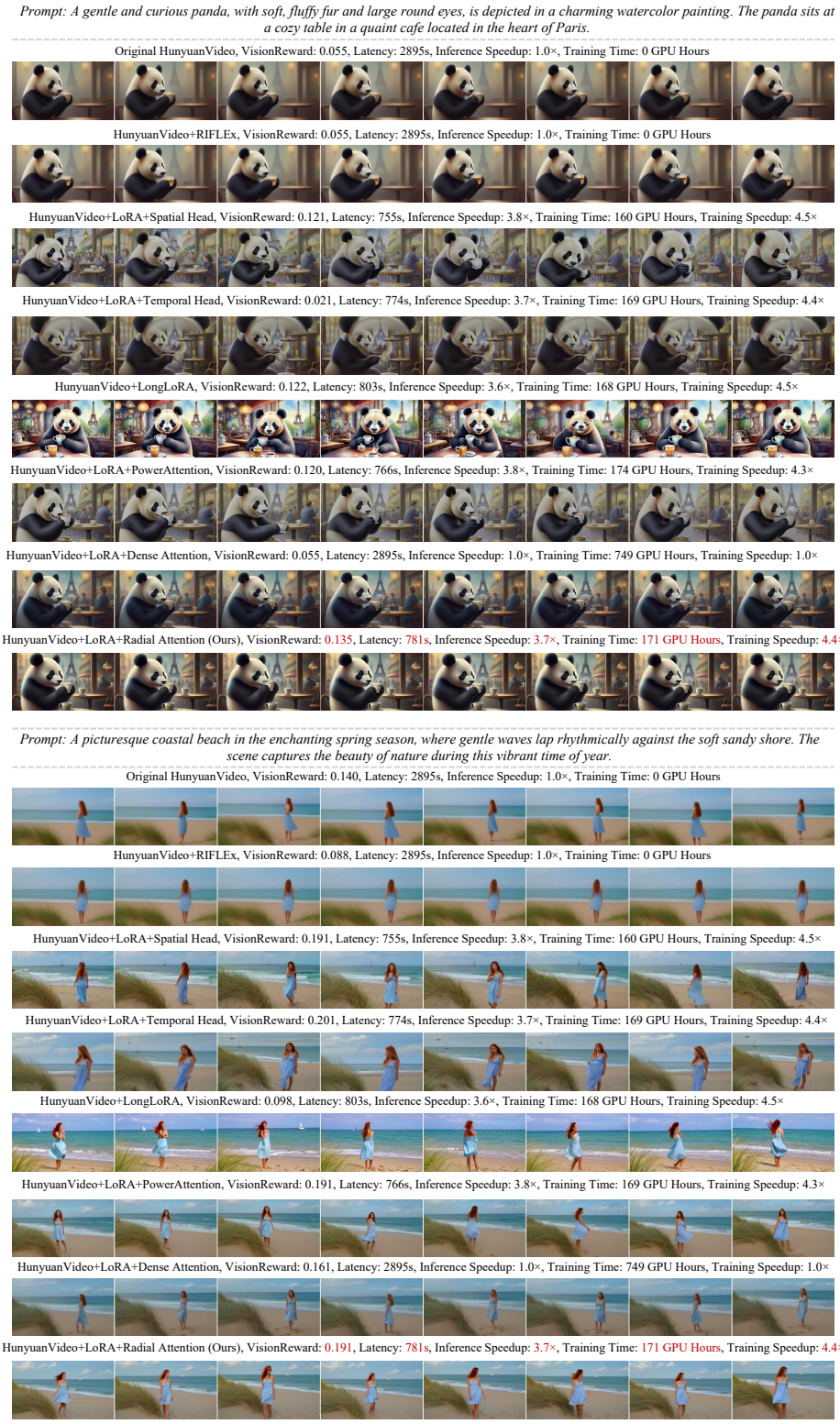


Figure C: Comparison of all baselines and Radial Attention at 4× default length (21 seconds, 509 frames) Text-to-Video video generation from HunyuanVideo. Radial Attention achieves the best Vision Reward score with good visual quality and consistency. In contrast, Original HunyuanVideo and RIFLEx generate blurred videos with poor visual quality. Temporal Head generates distorting figures. Spatial Head, Long LoRA, and PowerAttention generate temporally inconsistent video backgrounds. Dense Attention generates less dynamic videos.

https://doi.org/10.52202/085713-0568

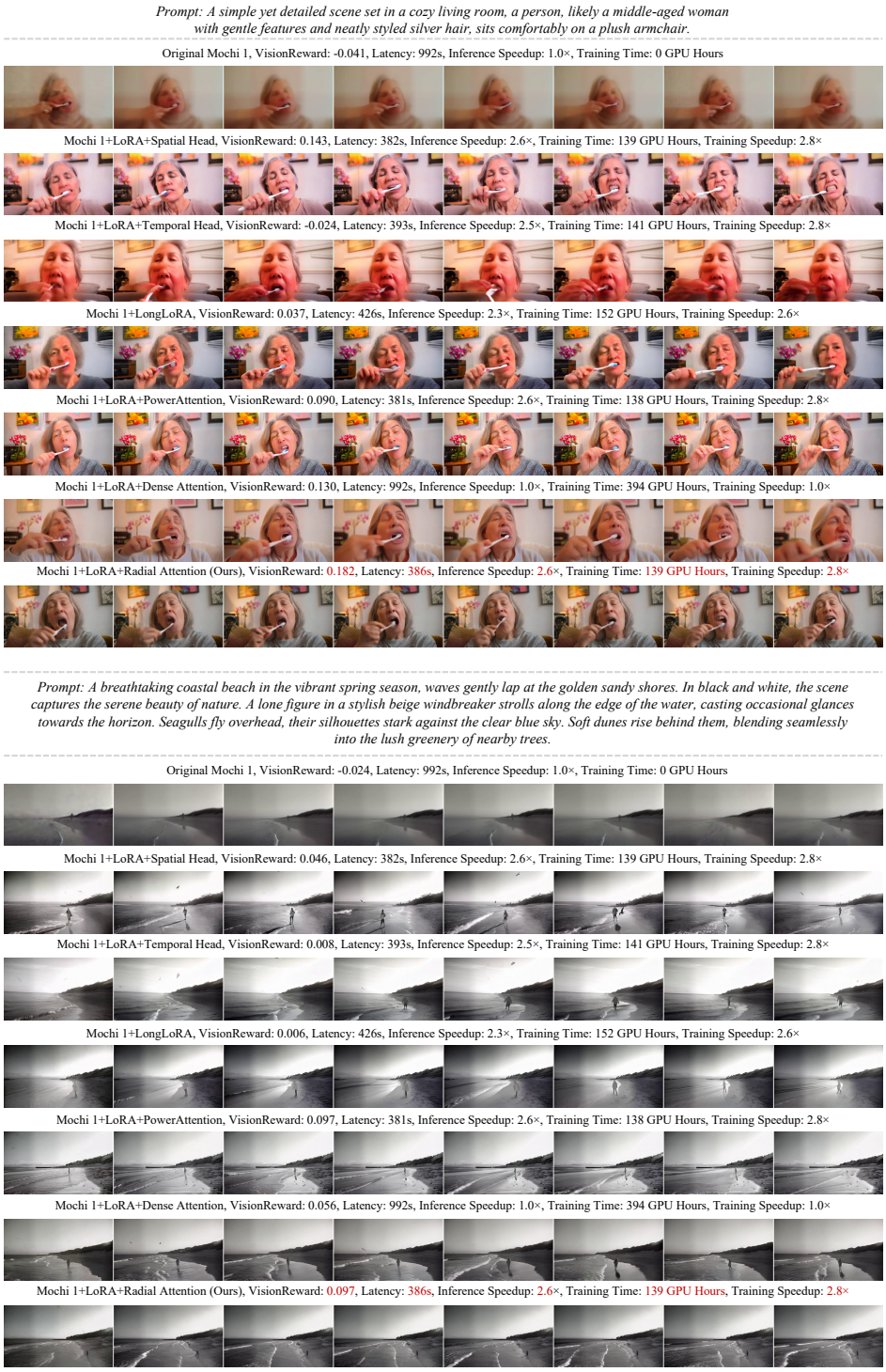


Figure D: Comparison of all baselines and Radial Attention at 4× default length (22 seconds, 667 frames) Text-to-Video video generation from Mochi 1. Radial Attention achieves the highest Vision Reward score because it has excellent visual quality and consistency. In contrast, Original Mochi 1 generates blurred videos with poor visual quality. Spatial Head, Temporal Head, Long LoRA, PowerAttention, and Dense Attention generate videos with either inconsistent backgrounds or inconsistent figures.

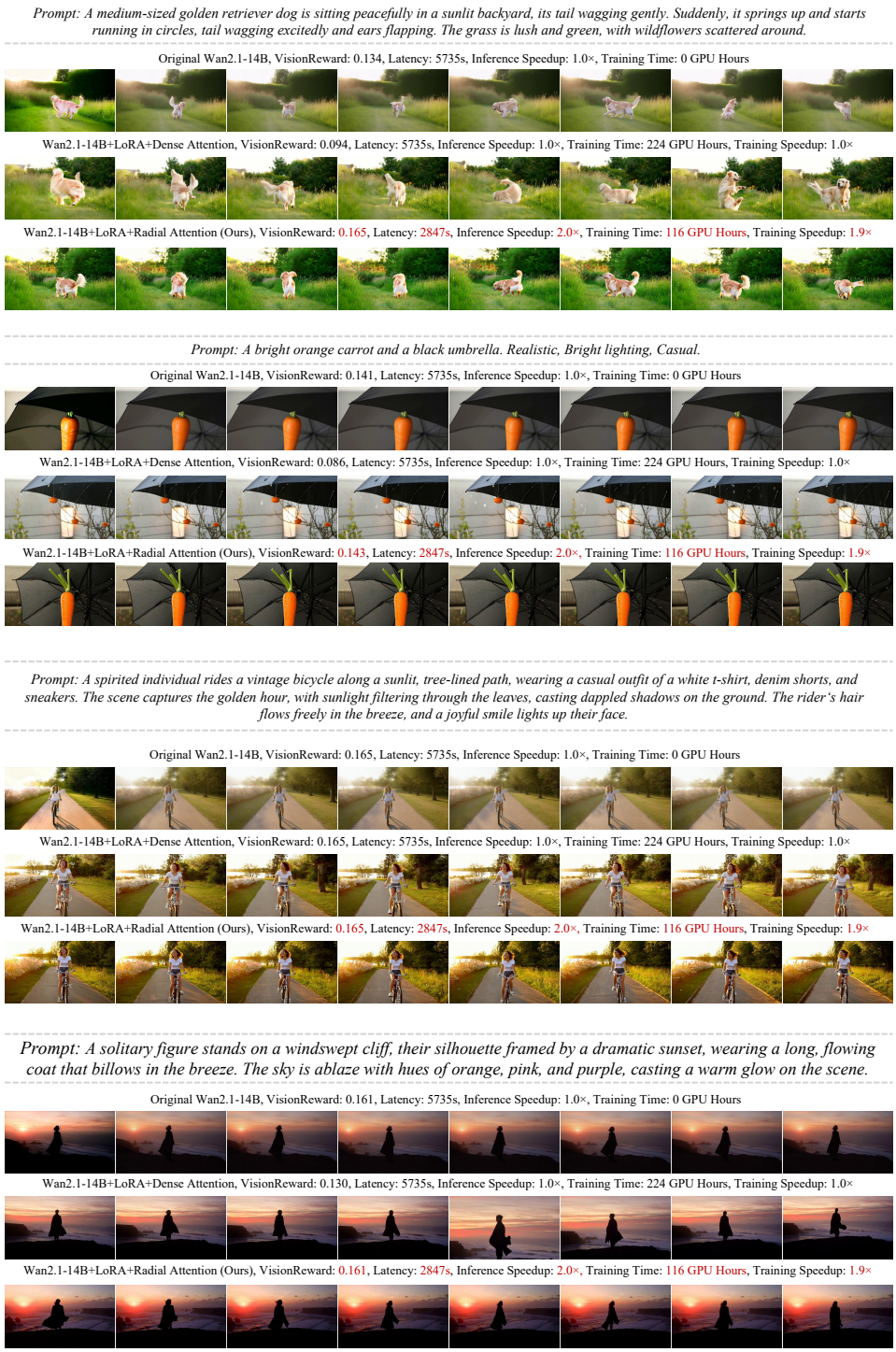


Figure E: Comparison of all baselines and Radial Attention at 2× default length (10 seconds, 161 frames) Text-to-Video video generation from Wan2.1-14B. Radial Attention achieves the highest Vision Reward score original Wan2.1-14B generates blurred videos and Dense Attention generates videos with inconsistent figures.



Figure F: Radial Attention LoRA is compatible with existing style LoRAs. On HunyuanVideo, it extends video length by 4× while maintaining a vision reward comparable to that of the original-length LoRA video.

https://doi.org/10.52202/085713-0568