PRINCETON
Electrical and Computer Engineering

Meta

# LinGen: Towards High-Resolution Minute-Length Text-to-Video Generation with Linear Computational Complexity

*Hongjie Wang, Chih-Yao Ma, Yen-Cheng Liu, Ji Hou, Tao Xu, Jialiang Wang, Felix Juefei-Xu, Yaqiao Luo, Peizhao Zhang, Tingbo Hou, Peter Vajda, Niraj K. Jha, Xiaoliang Dai*

The IEEE/CVF Conference on Computer Vision and Pattern Recognition 2025

CVPR Nashville JUNE 11-15, 2025

# High Quality Video Generation is at Huge Cost

- Length: 10-20 seconds (without extension)
- Resolution: 720-1080p (without super-resolution)
- Why not generate longer videos at higher resolutions? quadratic complexity of self-attention
  - a 2-minute 4K video costs 16,384 times more than a 15-second 1080p video

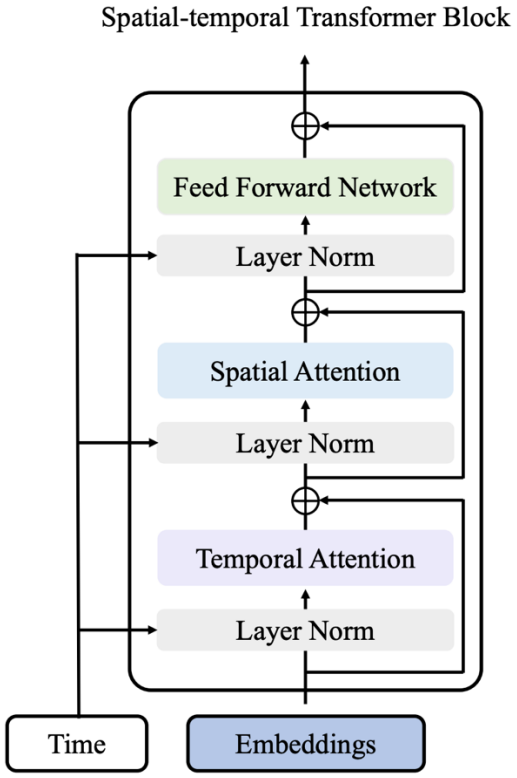
Google Veo2

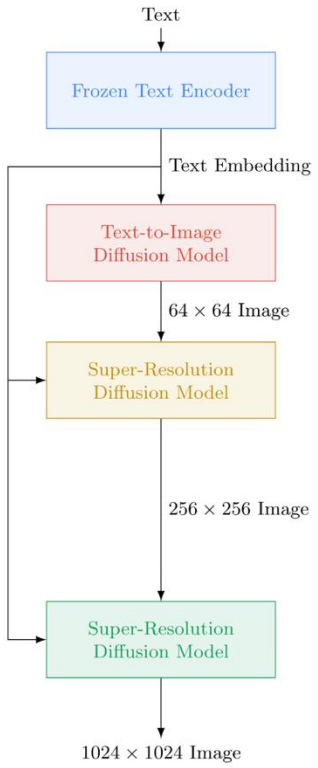
OpenAI Sora

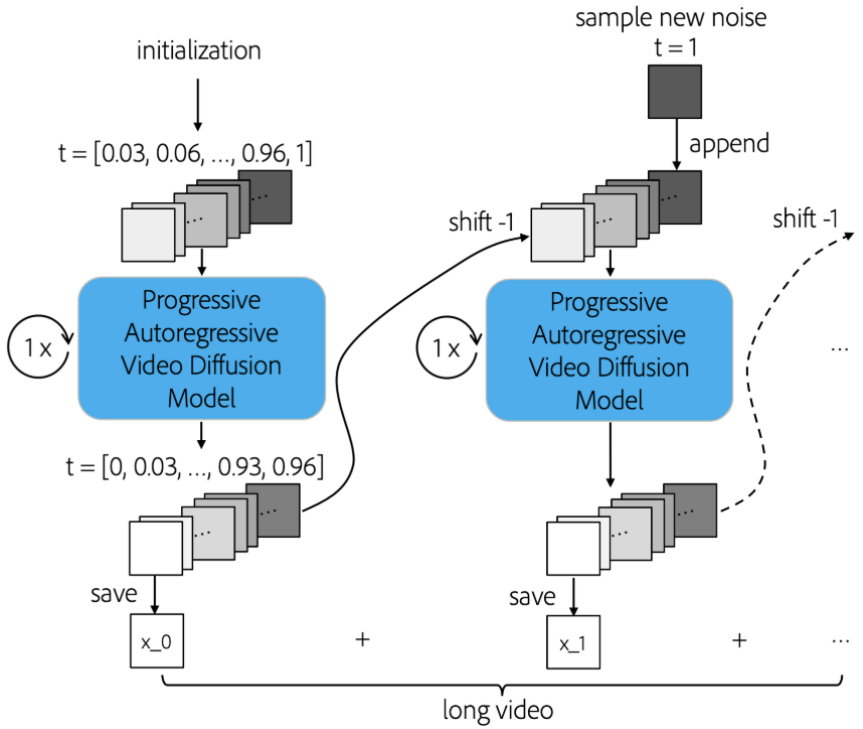
Meta MovieGen

# Existing Solutions

- Factorized Attention: Quadratic complexity and degraded quality
- Super-Resolution: Degraded quality and texture fidelity
- Video Extension:  Localized receptive field (fails to ensure long-term consistency)



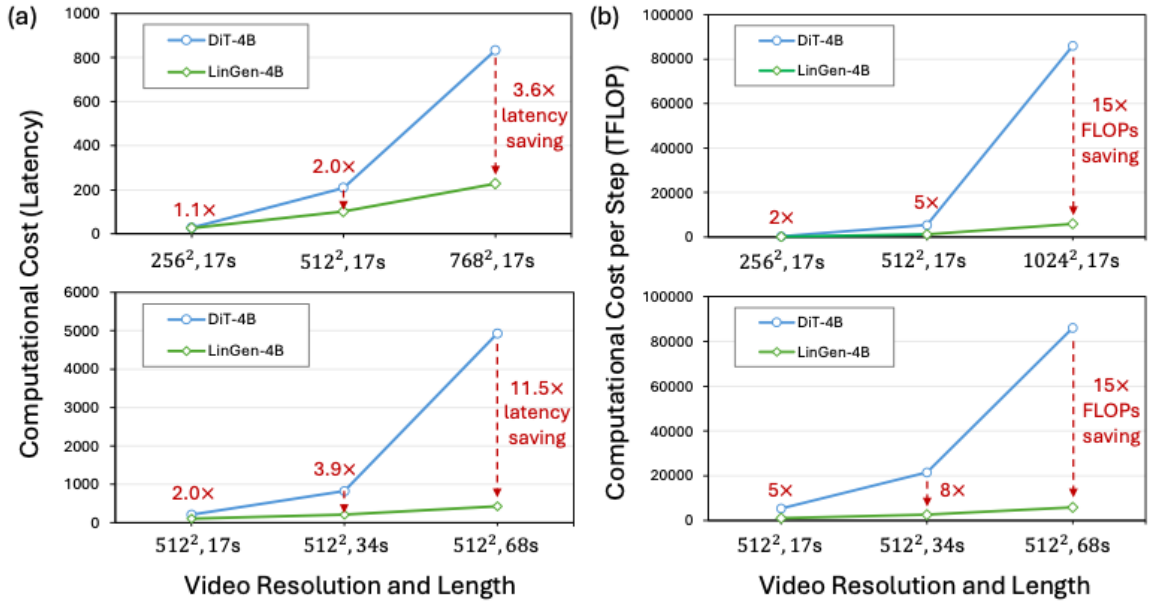





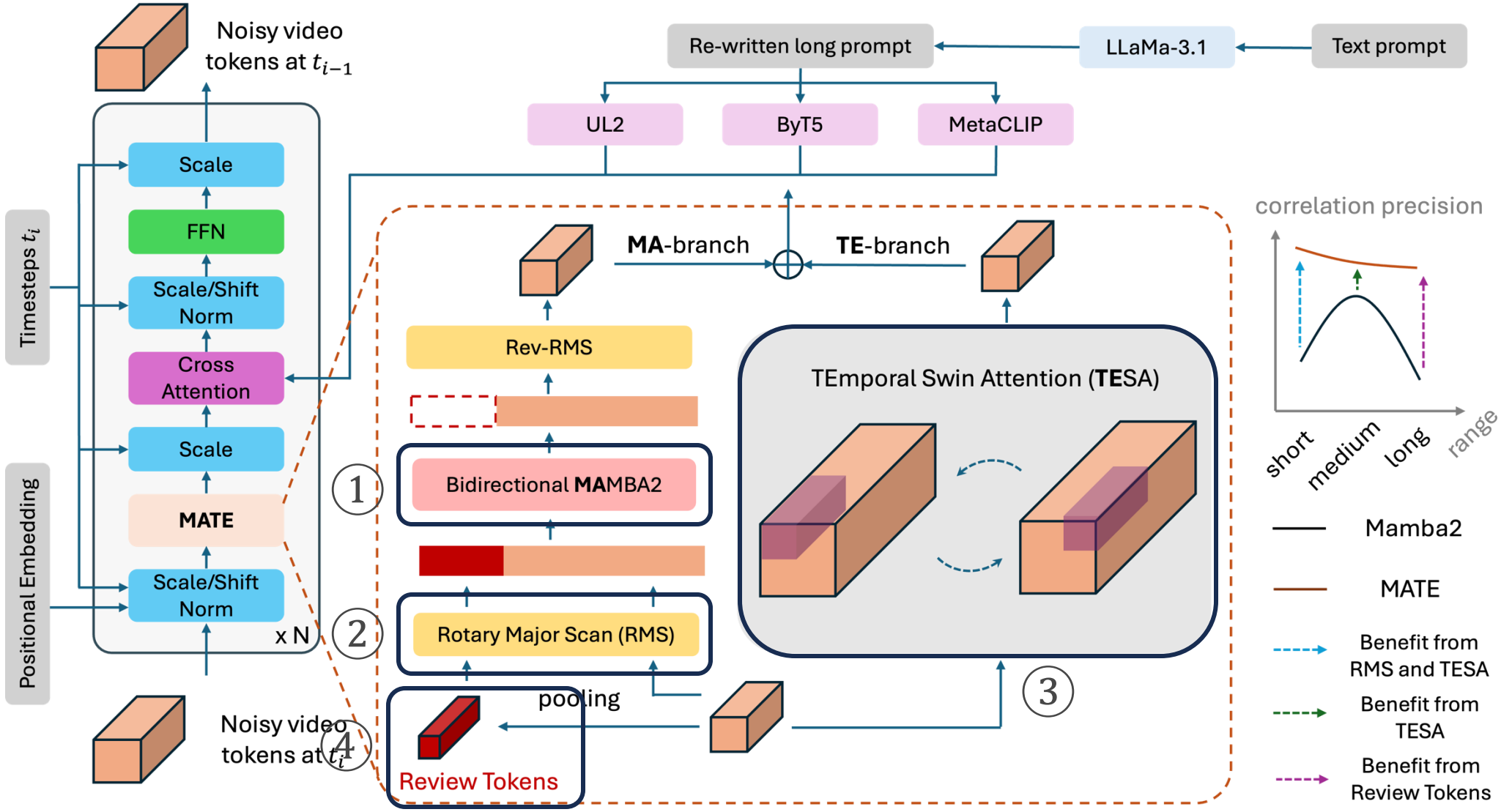
Lu, Haoyu, et al. "VDT: General-purpose video diffusion transformers via mask modeling." *arXiv preprint arXiv:2305.13311* (2023).
Saharia, Chitwan, et al. "Photorealistic text-to-image diffusion models with deep language understanding." *Advances in neural information processing systems* 35 (2022): 36479-36494.
Xie, Desai, et al. "Progressive autoregressive video diffusion models." *arXiv preprint arXiv:2410.08151* (2024).

# LinGen: Linear Complexity while Maintaining High Quality

- Linear computational complexity
- Maintain the same high quality as the self-attention-based DiT
- Global receptive field

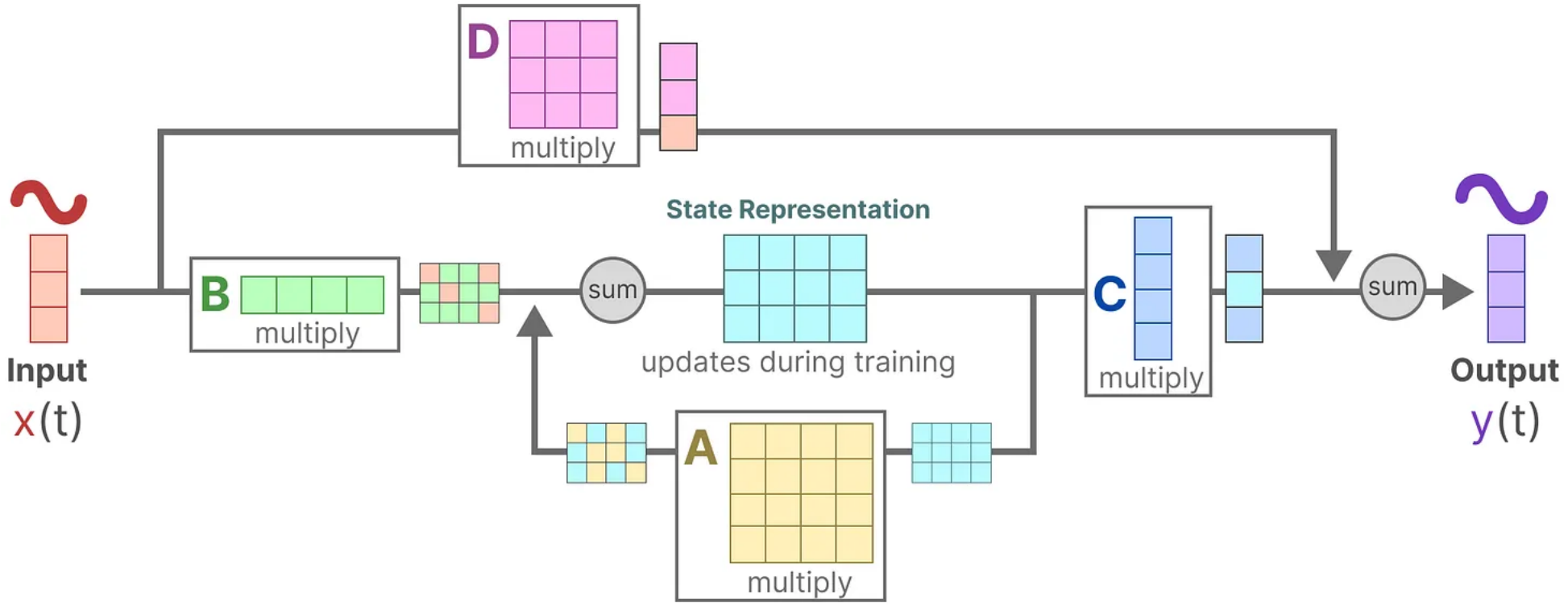



# LinGen: Linear Complexity while Maintaining High Quality



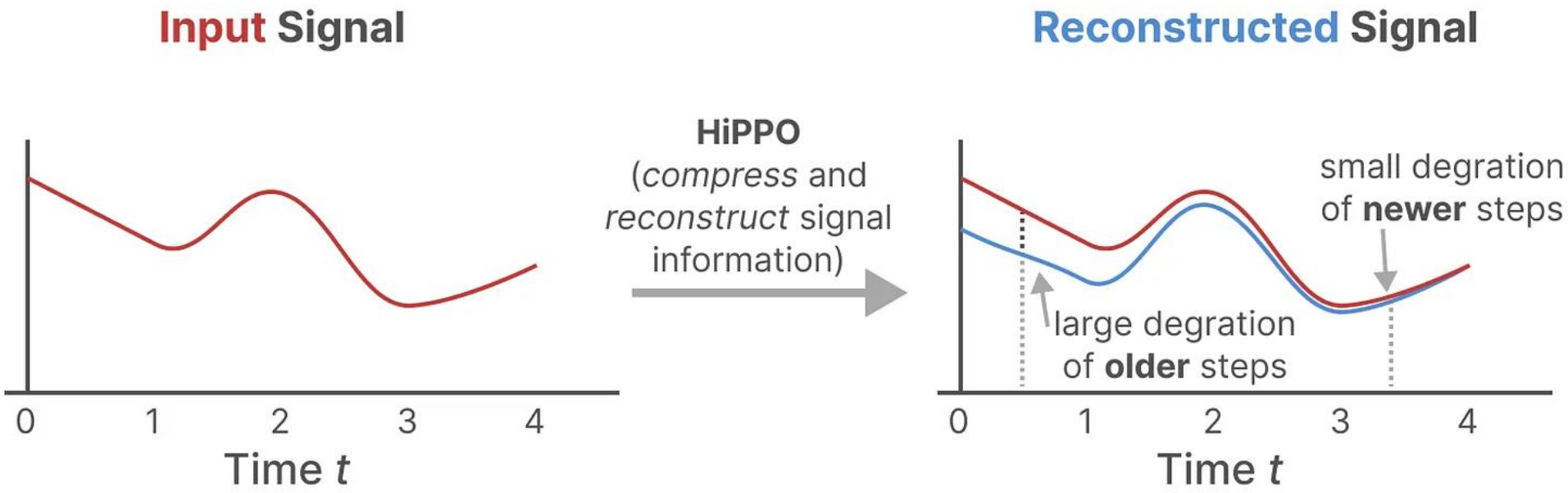
# State Space Model

- Sequence-to-sequence model with a hidden state memory
- The decay of the last state representation is controlled by A



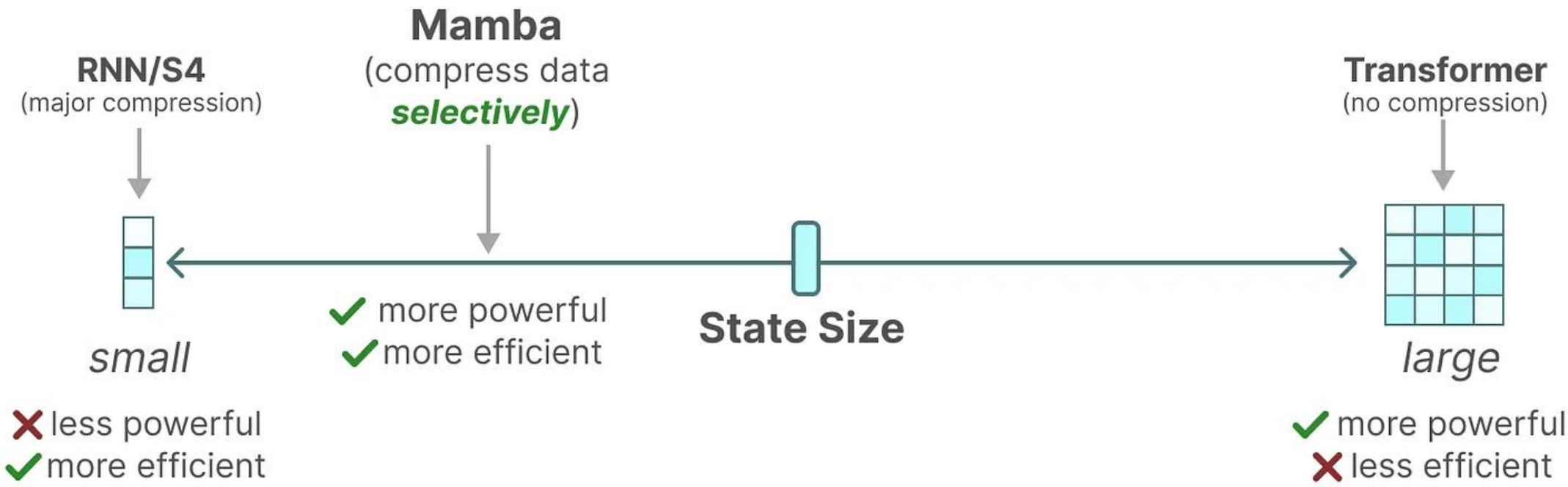
# State Space Model

- Sequence-to-sequence model with a hidden state memory
- The decay of the last state representation is controlled by A
- The precision of long-range correlations decays due to this



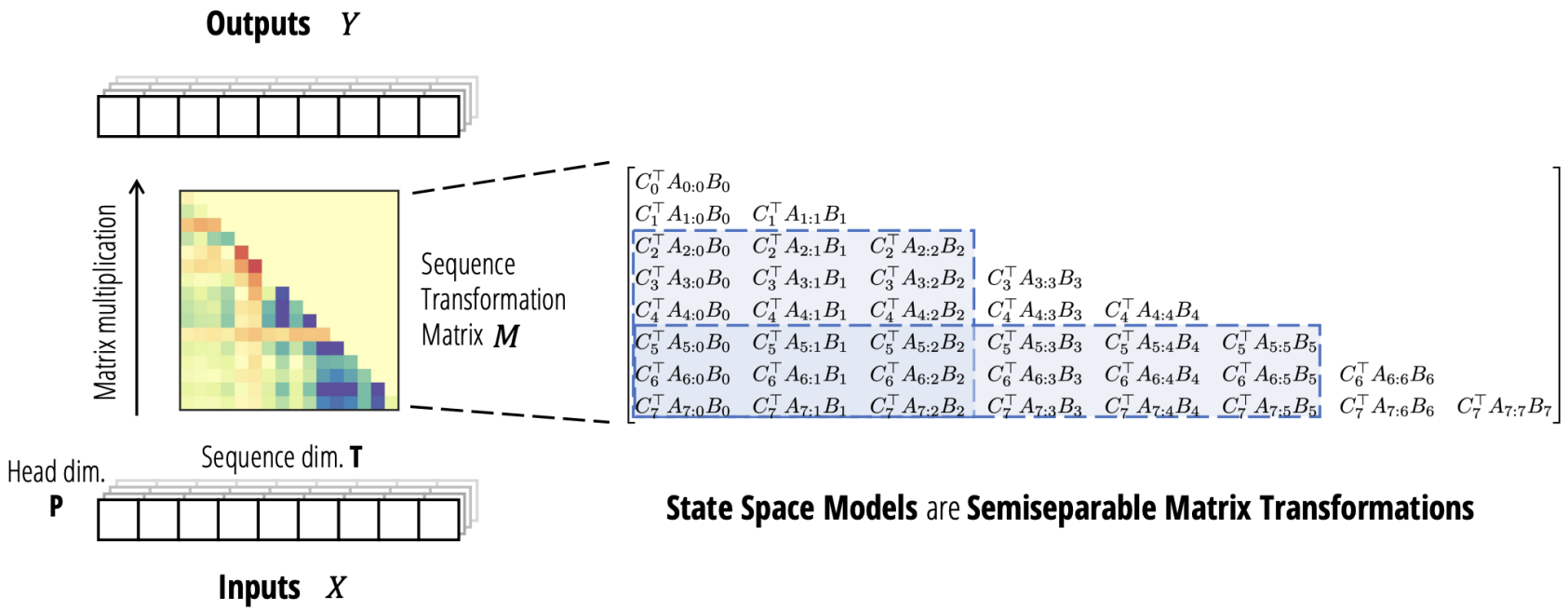
# Mamba: Selective Compression

- Mamba selectively compresses long-range correlation



# Mamba2: Attention-Format SSM

- Mamba2 can be written in attention format
  - It can leverage existing attention optimizations, such as xFormers, FlashAttention
- Mamba2 natively supports tensor parallelism and sequence parallelism
- Mamba2 supports much larger memory size

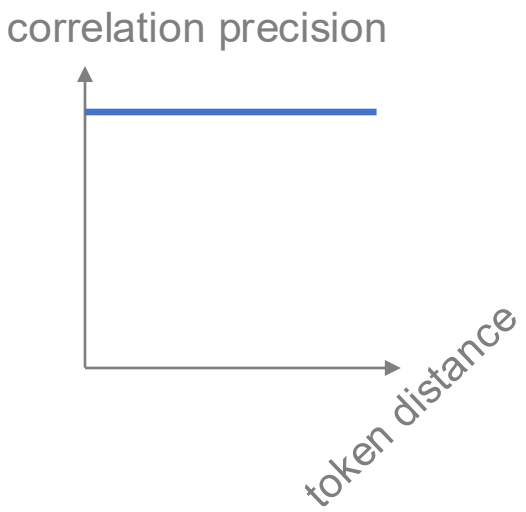



Dao, Tri, and Albert Gu. "Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality." *arXiv preprint arXiv:2405.21060* (2024).

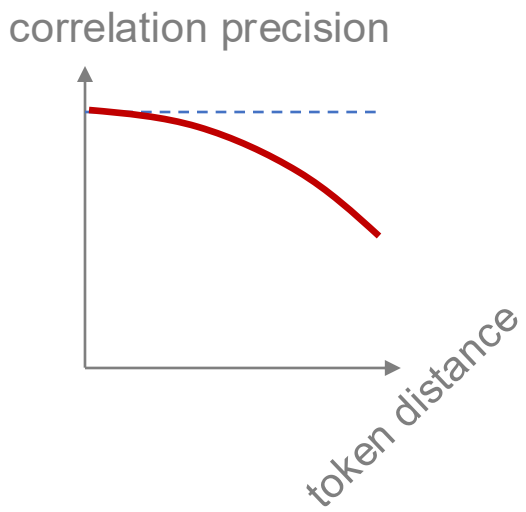
# Correlation Precision across Token Distances

*adjacency preservation issue*

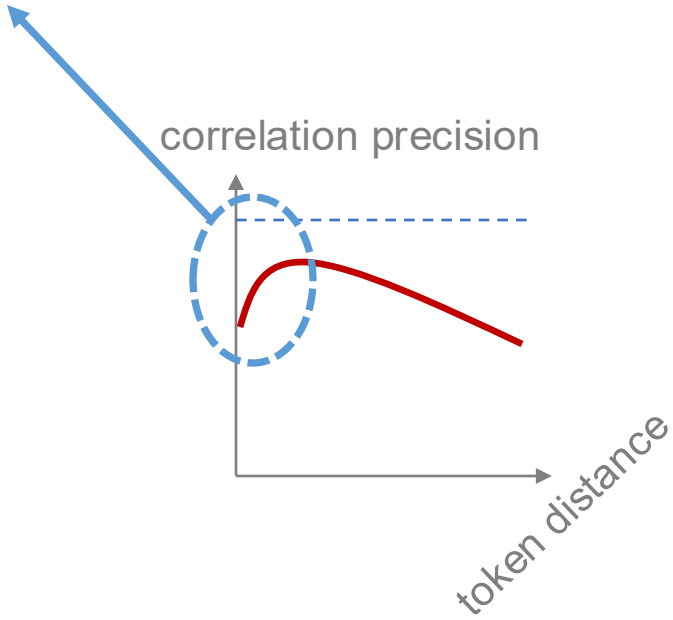
when turning a 2D/3D tensor to a sequence




Attention

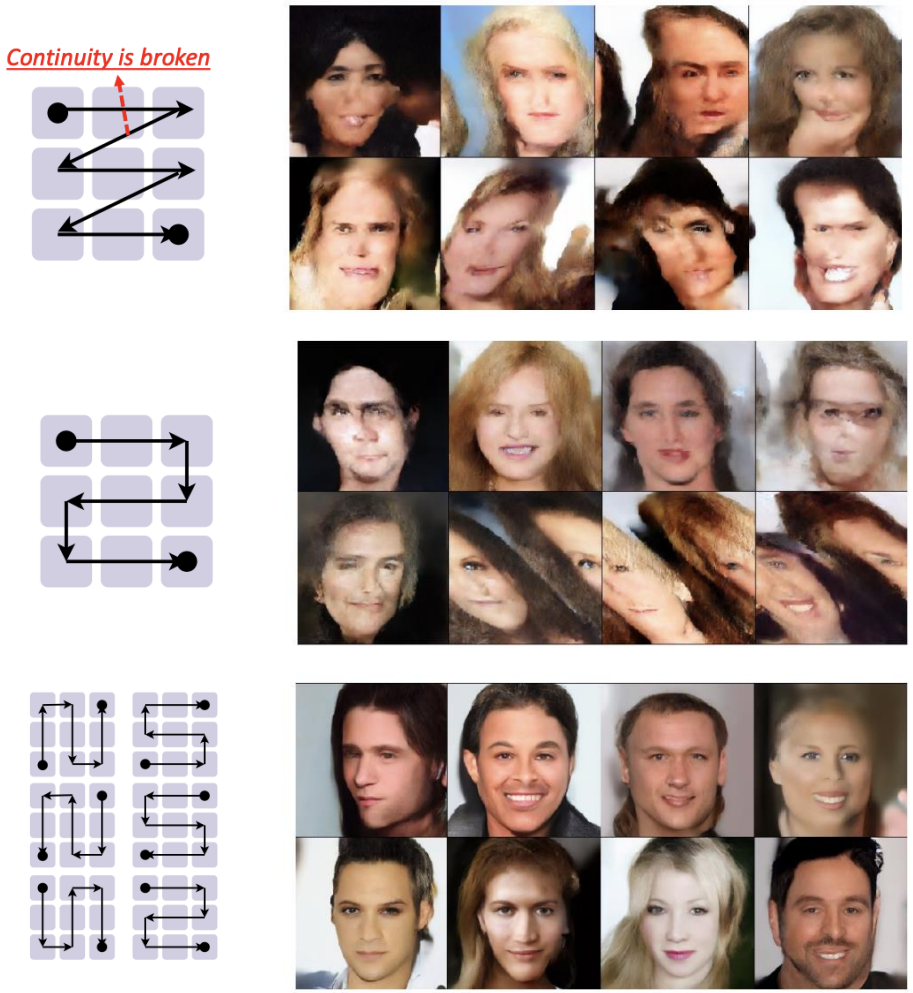


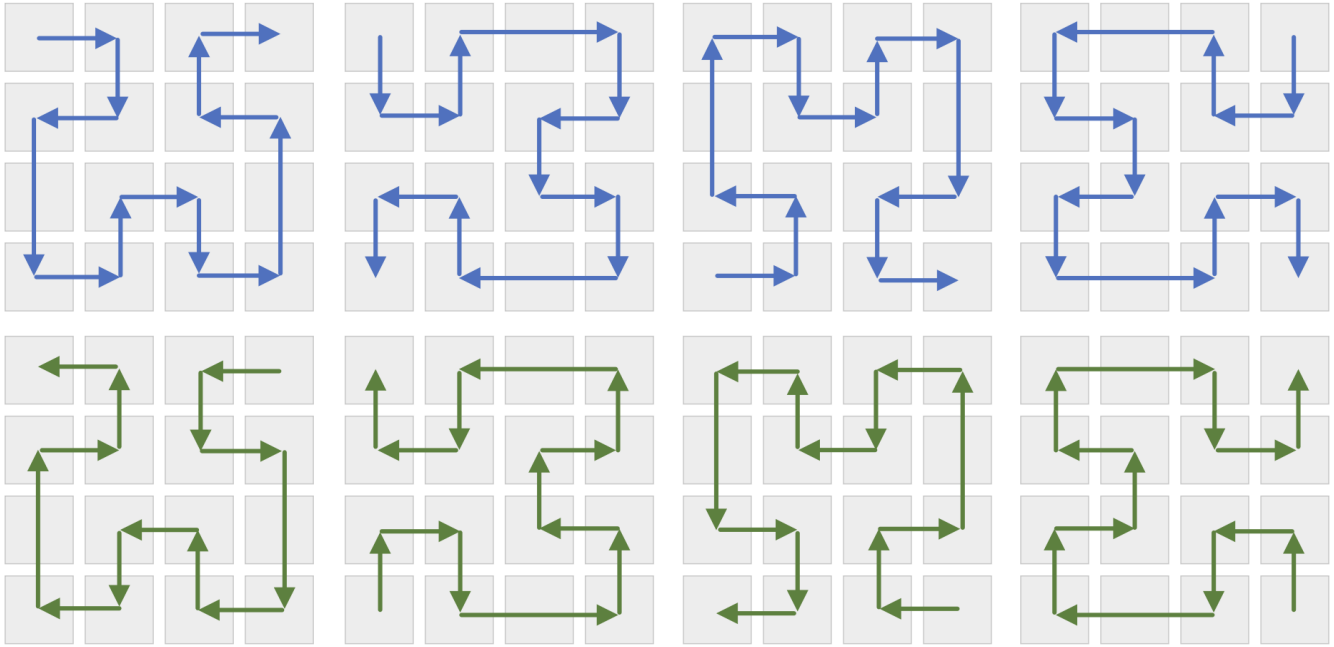

Mamba-text




Mamba-vision

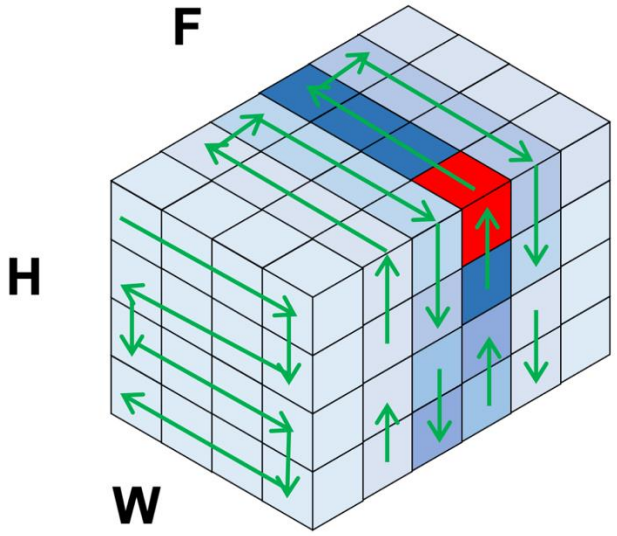
# Scan a 2D Tensor to a Sequence



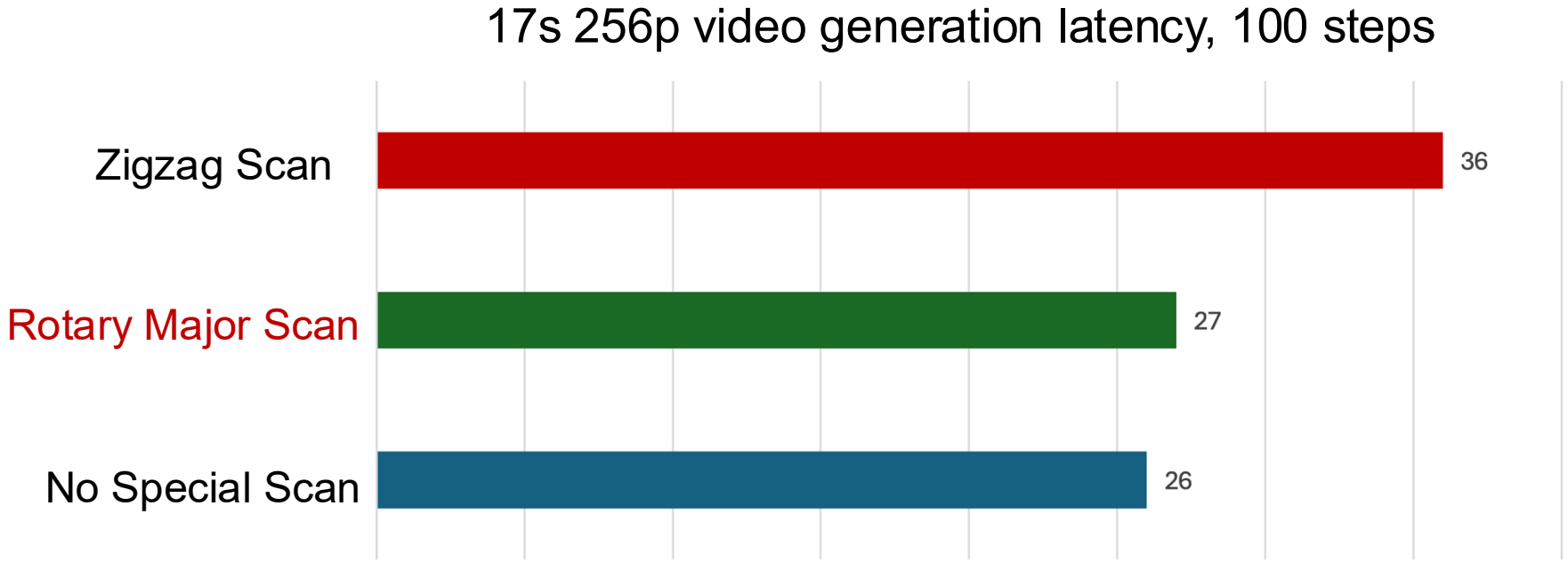



Hu, Vincent Tao, et al. "Zigma: Zigzag mamba diffusion model." *arXiv preprint arXiv:2403.13802* (2024).
Liu, Xiao, Chenxu Zhang, and Lei Zhang. "Vision mamba: A comprehensive survey and taxonomy." *arXiv preprint arXiv:2405.04404* (2024).

# How about a Huge 3D Video Token Tensor?




3D Zigzag Scan

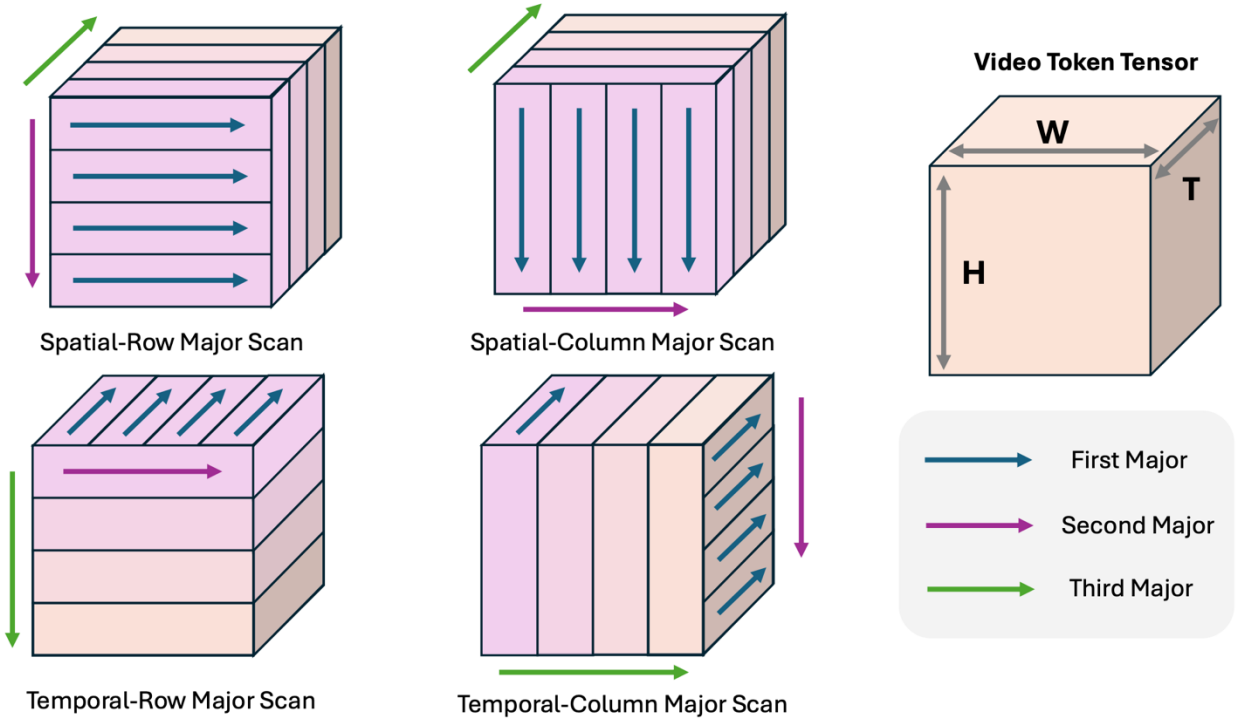


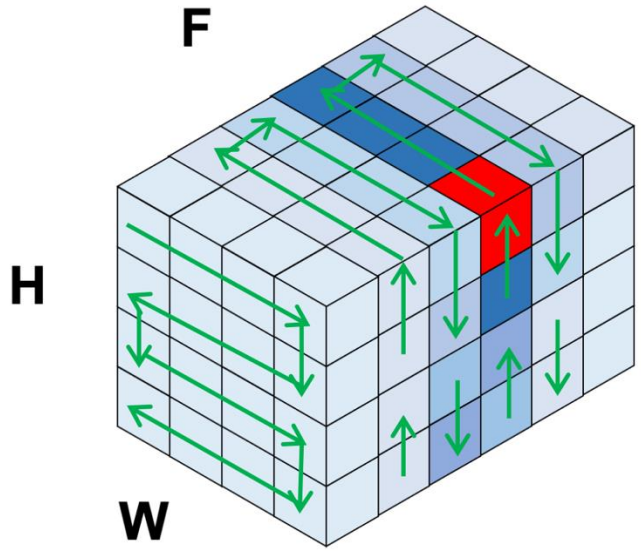

Existing special scan methods, such as Zigzag scan and Hilbert scan, incur significant extra cost when dealing with huge 3D tensors

Gao, Yu, et al. "Matten: Video Generation with Mamba-Attention." *arXiv preprint arXiv:2405.03025* (2024).

# Rotary Major Scan

- Minimize the average distance between adjacent tokens after scanning at nearly no cost

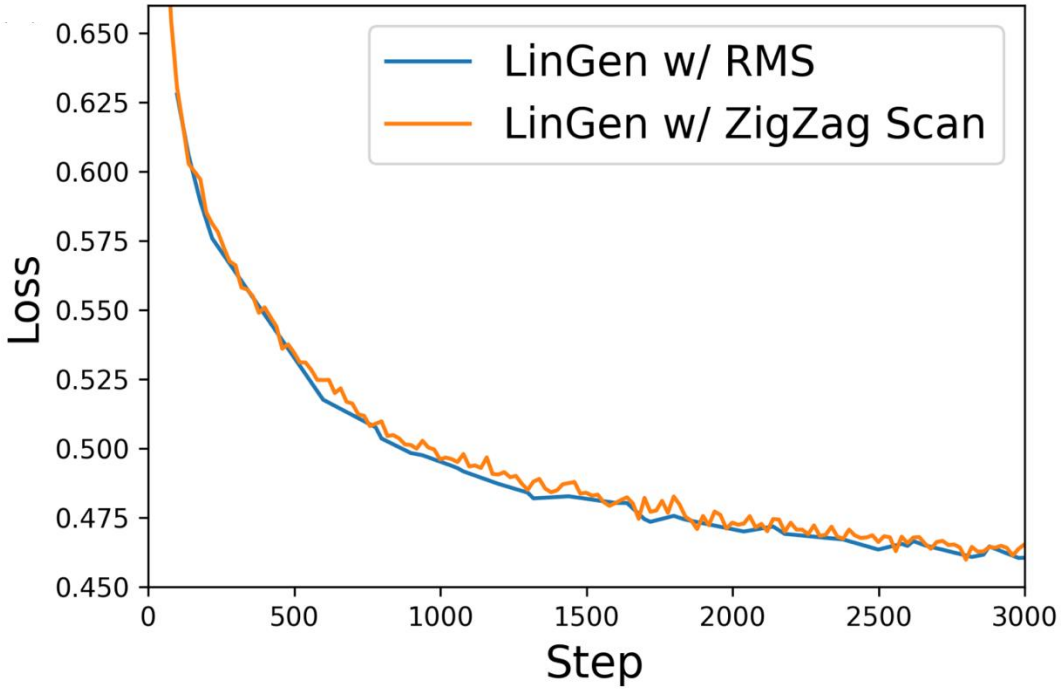


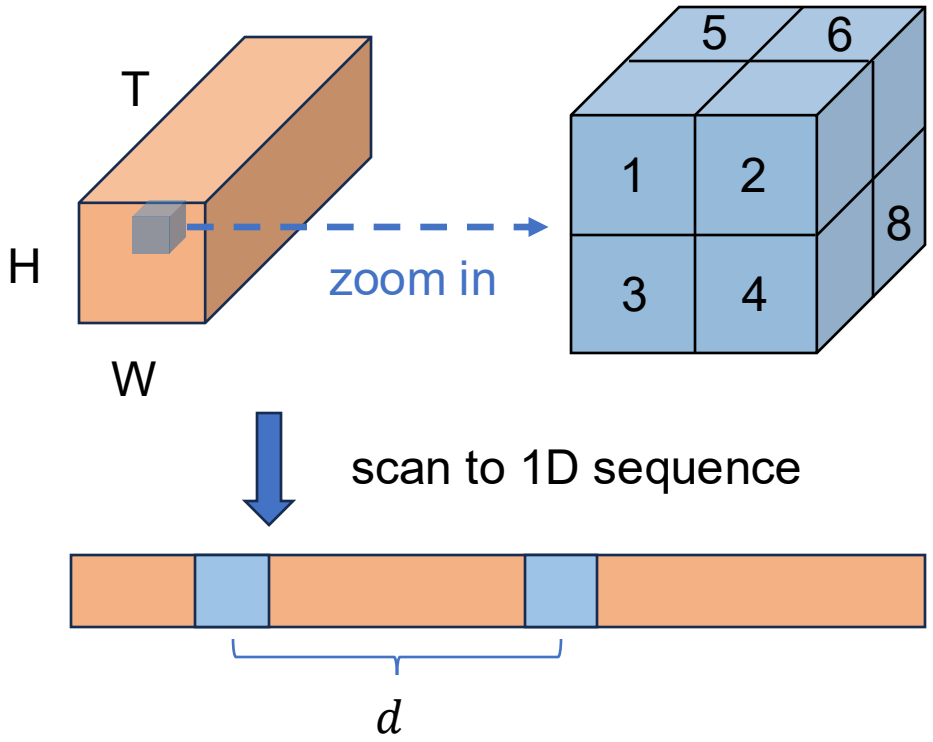
# Rotary Major Scan

- Minimize the average distance between adjacent tokens after scanning at nearly no cost
- It can be simply implemented in a few lines of code while outperforming Zigzag scan

```
elif self.scan_type == "rotary":  ### Rotary Major Scan
    if NF == 1 and self.layer_idx % 2 == 1:  ### text-to-image
        xz = rearrange(xz, "b d (h w) -> b d (w h)", h=H, w=W)
    elif NF > 1 and self.layer_idx % 4 > 0: ### text-to-video
        if self.layer_idx % 4 == 1:
            xz = rearrange(xz, "b d (f h w) -> b d (f w h)", f=NF, h=H, w=W)
        elif self.layer_idx % 4 == 2:
            xz = rearrange(xz, "b d (f h w) -> b d (h w f)", f=NF, h=H, w=W)
        elif self.layer_idx % 4 == 3:
            xz = rearrange(xz, "b d (f h w) -> b d (w h f)", f=NF, h=H, w=W)
```



# Issue: Spatial-Temporal Neighbors in the Token Tensor




None of the existing scan methods can place all the 8 tokens close to each other

Spatial-Row Major Scan

[ ... ( ... 1 2 ... ) ( ... 3 4 ... ) ... ] [ ... ( ... 5 6 ... ) ( ... 7 8 ... ) ... ]

[ ] Tokens in a frame

( ) Tokens in a row

Spatial-Column Major Scan

...

Temporal-Row Major Scan

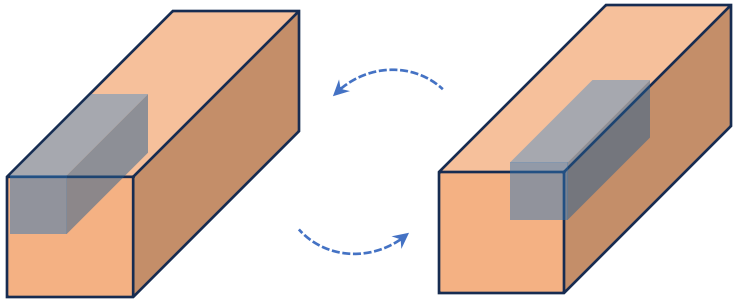
...

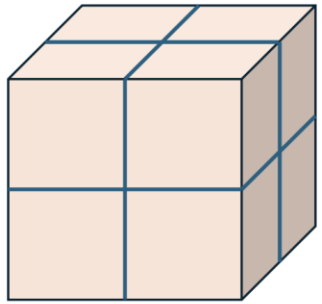
Temporal-Column Major Scan

...

# TEmporal Swin Attention (TESA)

- TESA is a 3D window attention with
  - a special fixed window size (small spatial range and medium temporal range)
  - alternately shifted window scopes

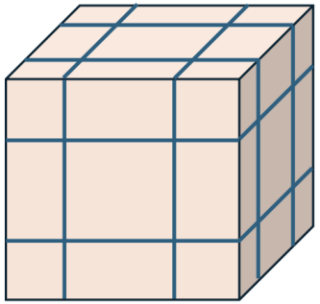




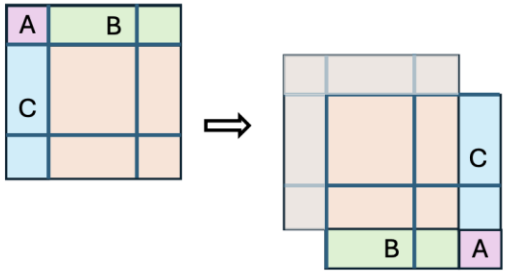
Normal TESA Windows

Normal Spatial Division

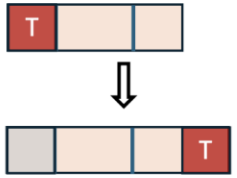
Normal Temporal Division



Shifted TESA Windows




Shifted Spatial Division

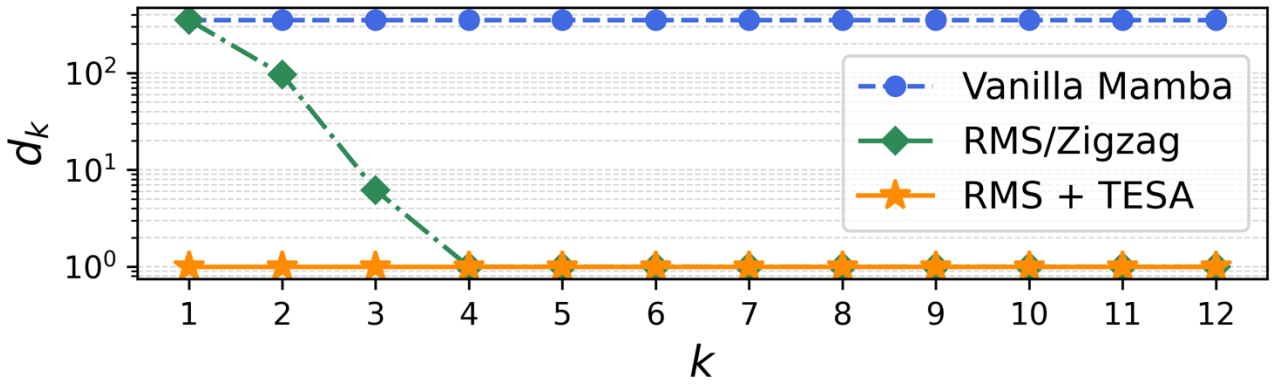



Shifted Temporal Division

# TEmporal Swin Attention (TESA)

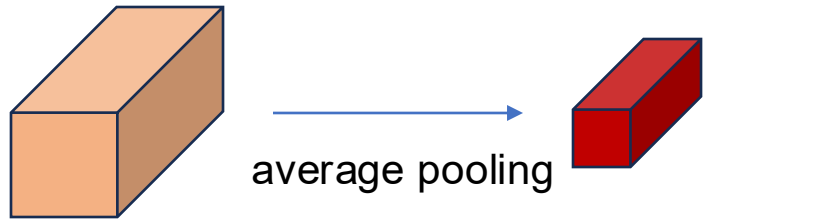
- TESA is a 3D window attention with
  - a special fixed window size (small spatial range and medium temporal range)
  - alternately shifted window scopes
- ✅ Small spatial range reduces the cost but still covers adjacent tokens
- ✅ Medium temporal range ensures temporal consistency across frames
- ✅ Fixed window size makes the complexity of TESA linear

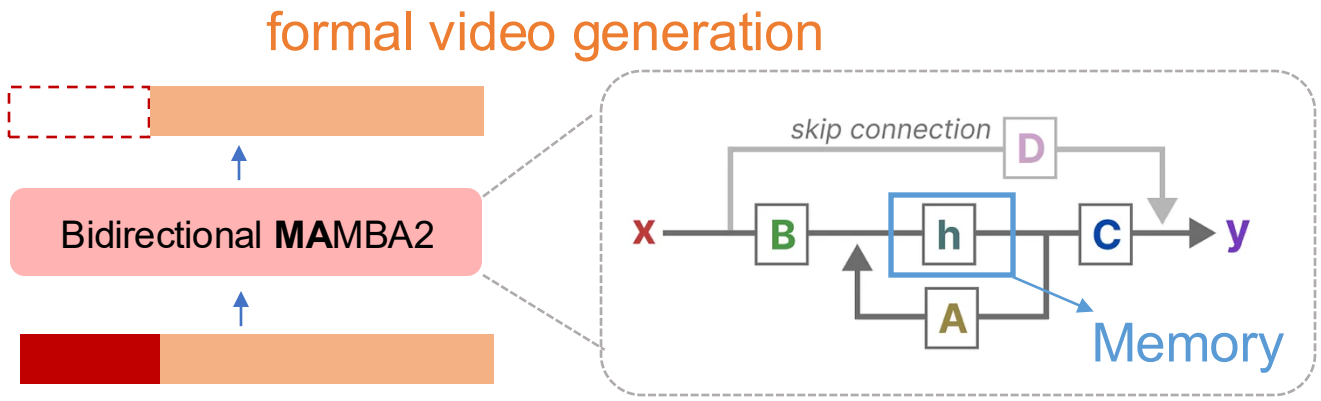
| Model | Latency/s |
|---|---|
| LinGen (default setting) | 102 |
| LinGen w/o TESA | 94 (-8) |



# Review Tokens: Enhancing Long-Range Correlations

- Average pooling to obtain the overview of the video that is being processed
- Overview appended to the beginning of sequence to write it into the memory of BiMamba
- Marginal extra cost due to the aggressive pooling ratio (8x4x4)



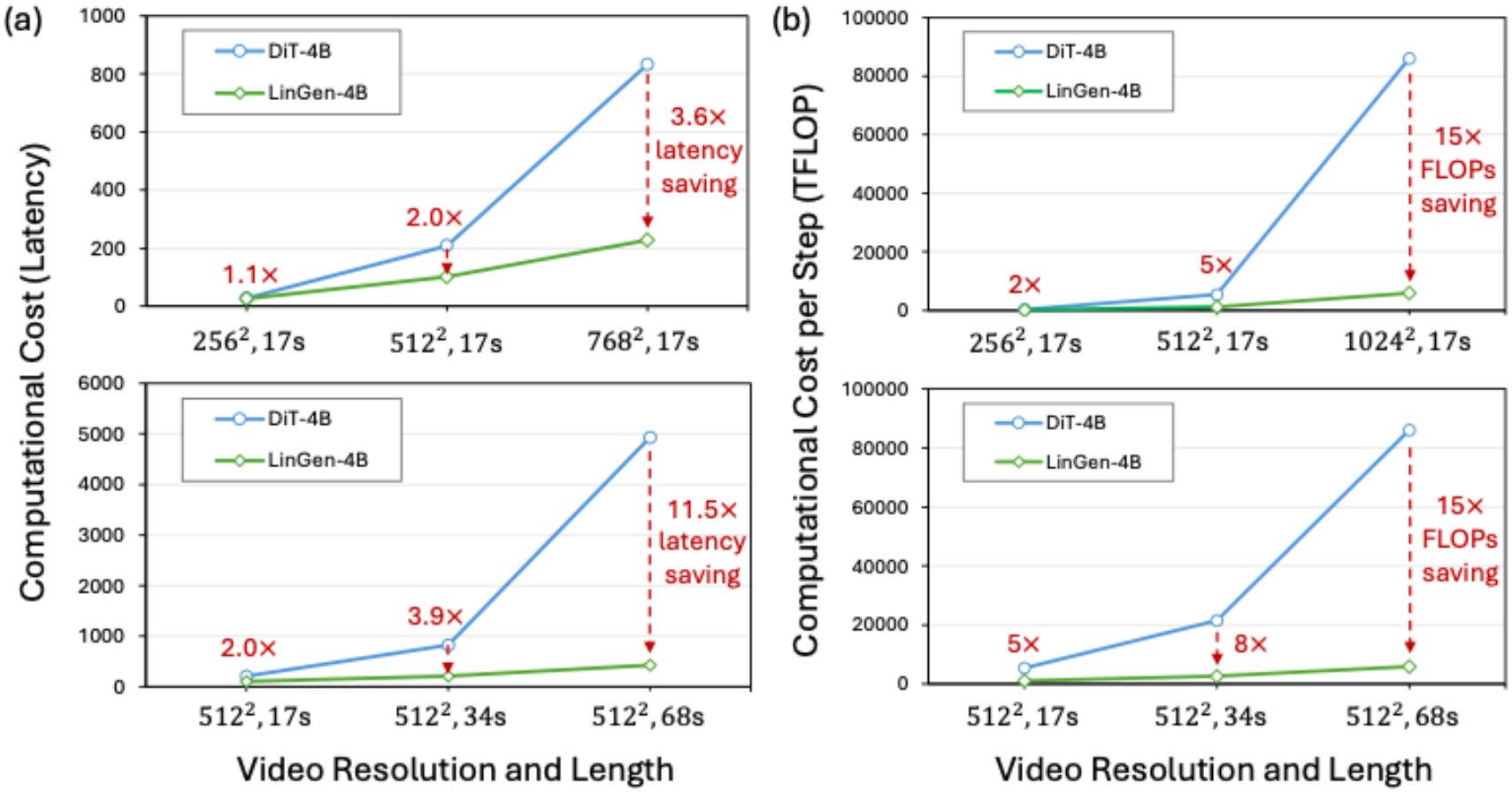

Writing the overview of the generated video
into the hidden memory of BiMamba2

# Training Recipe

- Progressive Training: gradually increase sequence length in the latent space
  - 256p text-to-image
  - 256p text-to-video, 17s
  - 512p text-to-video, 17s
  - 512p text-to-video, 34s
  - 512p text-to-video, 68s
- Hybrid Training
  - t2i : t2v = 1:50 to enhance the consistency of generated videos
- Quality Tuning
  - Fine-tuning on 3K videos with extremely high quality and good motion

# Efficiency: Linear Computational Complexity

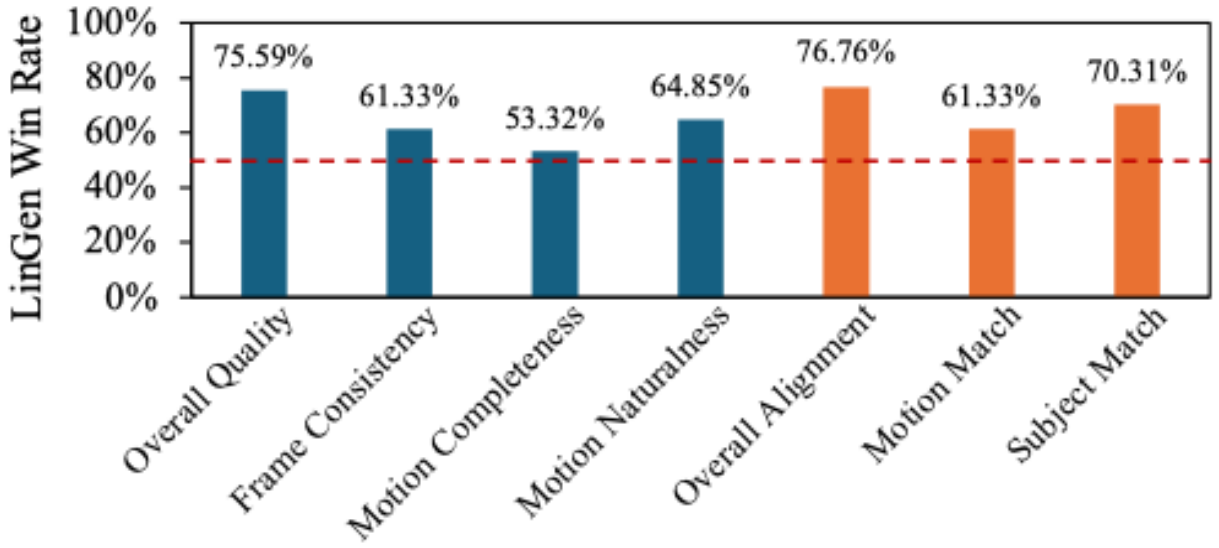
- LinGen generates 512p 68-second 16fps videos on a single GPU without super-resolution or video extension, achieving up to 12x speed-up without sampling distillation



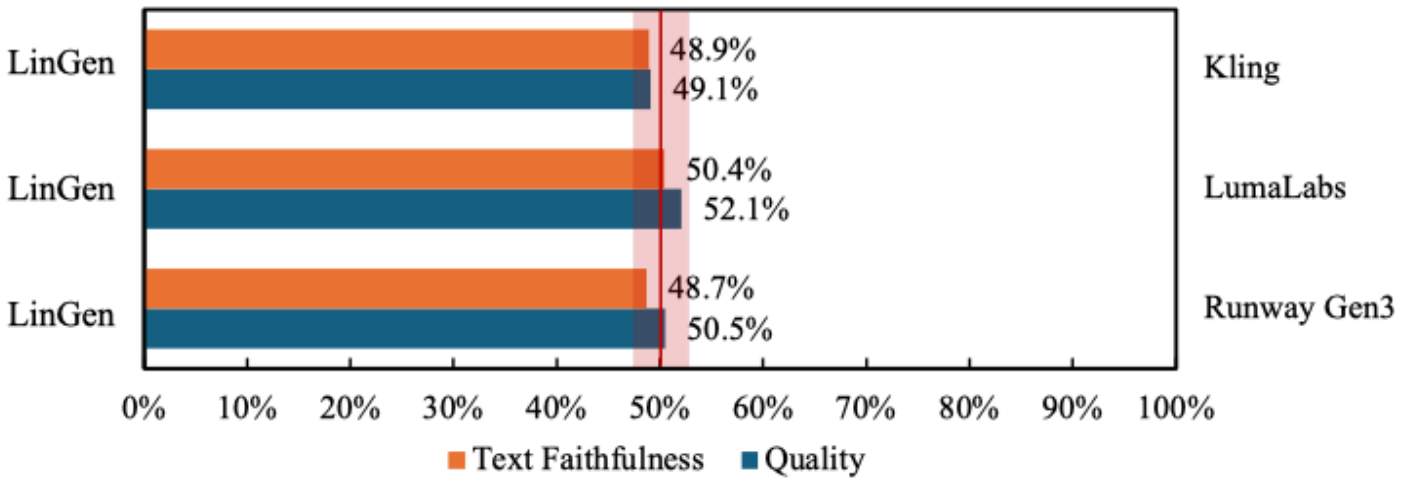
# Performance: Visual Examples

# Performance: Human Evaluation

LinGen vs. DiT
same size
same dataset
same training recipe



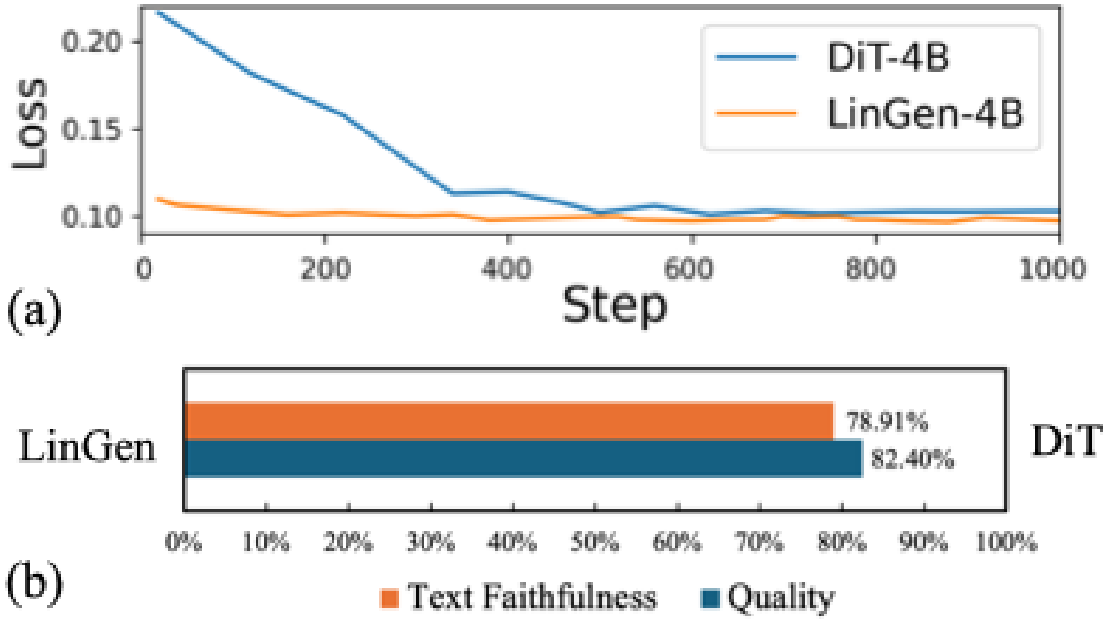

LinGen vs. SOTA models



# Faster Adaptation to Longer Token Sequences

Transferring from 256p text-to-video generation to 512p text-to-video generation

Human evaluation after 1k pre-training steps on 512p text-to-video generation



# Performance: Automatic Metrics

| Model | Subject Consist. | BG. Consis. | Temp. Flick. | Motion Smooth. | Aesthe. Quality | Imag. Quality | Dyna. Degree | **Quality Score** | **Total Score** | **Max. Raw Frames** |
|---|---|---|---|---|---|---|---|---|---|---|
| Runway Gen-3 [37] | 97.10% | 96.62% | 98.61% | 99.23% | 60.14% | **63.34%** | **66.82%** | **84.11%** | **82.32%** | 256 |
| Kling [18] | **98.33%** | 97.60% | 99.30% | **99.40%** | 46.94% | 61.21% | 65.62% | 83.39% | 81.85% | 313 |
| OpenSora V1.2 [59] | 96.75% | **97.61%** | **99.53%** | 98.50% | 42.39% | 56.85% | 63.34% | 81.35% | 79.76% | 408 |
| LinGen | 98.30% | 97.60% | 99.26% | 98.58% | **63.67%** | 60.55% | 63.36% | 83.77% | 81.76% | **1088** |

| Model | Object Class | Multiple Objects | Human Action | Color | Spatial Relatio. | Scene | Appear. Style | Temp. Style | Overall Consist. | **Semantic Score** |
|---|---|---|---|---|---|---|---|---|---|---|
| Runway Gen-3 [37] | 87.81% | 53.64% | 96.40% | 80.90% | 65.09% | **54.57%** | **24.31%** | **24.71%** | 26.69% | 75.17% |
| Kling [18] | 87.24% | **68.05%** | 93.40% | 89.90% | **73.03%** | 50.86% | 19.62% | 24.17% | 26.42% | **75.68%** |
| OpenSora V1.2 [59] | 82.22% | 51.83% | 91.20% | **90.08%** | 68.56% | 42.44% | 23.95% | 24.54% | **26.85%** | 73.39% |
| LinGen | **90.98%** | 55.15% | **97.50%** | 83.95% | 58.15% | 53.51% | 21.08% | 24.29% | 26.32% | 73.73% |

Table 1. Automatic evaluation of LinGen on VBench-Long. **Quality Score** measures the quality of generated videos and **Semantic Score** measures text-video alignment. **Total Score** is their weighted sum. Higher values indicate better performance for all these metrics.

LinGen has a similar score to Gen-3 and Kling, while achieving much longer video generation at much lower cost

# Conclusions

- It is not necessary to involve full attention to achieve good performance on video generation
- Deploying SSMs (or any other seq2seq model with long-range decay) on vision tasks encounters the adjacency preservation issue
- Our proposed Rotary-Major Scan and TEmporal Swin Attention address this issue at marginal cost
- Review tokens enhance the long-range correlations in generated videos
- Compared to the standard DiT, our linear-complexity LinGen achieves better video quality and text-video alignment at 12x lower cost