NIPS 2025

*Spotlight*

# FPSAttention: Training-Aware FP8 and Sparsity Co-Design for Fast Video Diffusion

Akide Liu*, Zeyu Zhang*, Zhexin Li†, Xuehai Bai†, Yuanjie Xing†, Yizeng Han, Jiasheng Tang‡, Jichao Wu, Mingyang Yang, Weihua Chen, Jiahao He, Yuanyu He, Fan Wang, Gholamreza Haffari, Bohan Zhuang‡

Monash University
DAMO Academy, Alibaba Group
ZIP Lab, Zhejiang University
Hupan Lab

# 1. Challenges

**Chapter 01**

## The Efficiency Challenge in Video Diffusion

- Video diffusion models achieve great quality but are computationally heavy.
- Hundreds of denoising steps → slow sampling.
- Spatio-temporal attention ≈ 70% of total inference time.
- Example: Wan 2.1-14B takes ~2.5 hours to generate a 5 s video.

## Quantization and Sparsity in Video Diffusion

- Quantization reduces precision (e.g., FP32 → FP8) to save memory and speed up computation.
- Sparsity skips less important tokens to reduce quadratic attention cost.
- Each method works independently but fails to optimize jointly.
- However, naively combining them leads to large performance drop.
- FP8 quantization and sparsity co-design is underexplored.

## Why Simple Combination Fails?

- Quantization errors distort high-magnitude attention scores.
- Sparsity selects those high-magnitude tokens → error amplification.
- No training-time adaptation ⇒ large training-inference gap.
- **Need a joint, training-aware optimization to balance efficiency and quality.**

## Our Motivation

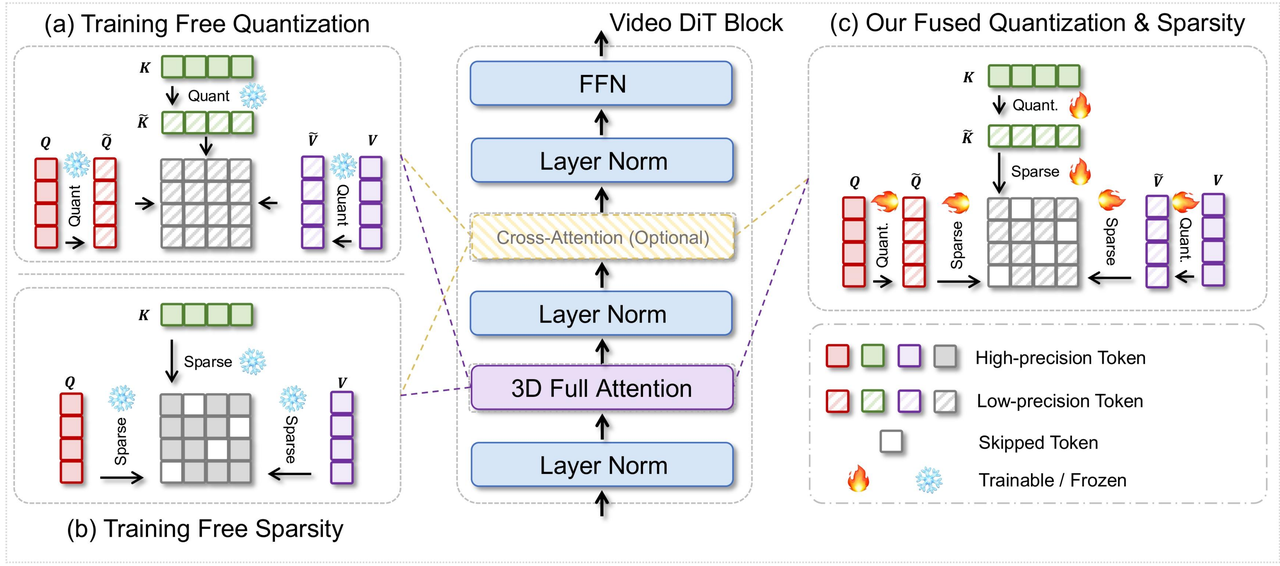
- Efficiency is crucial for real-world deployment.
- Quantization and sparsity offer complementary benefits — if optimized together.
- Solve the training-inference gap
- We aim to co-design a training-aware framework that achieves high speed without quality loss in video diffusion

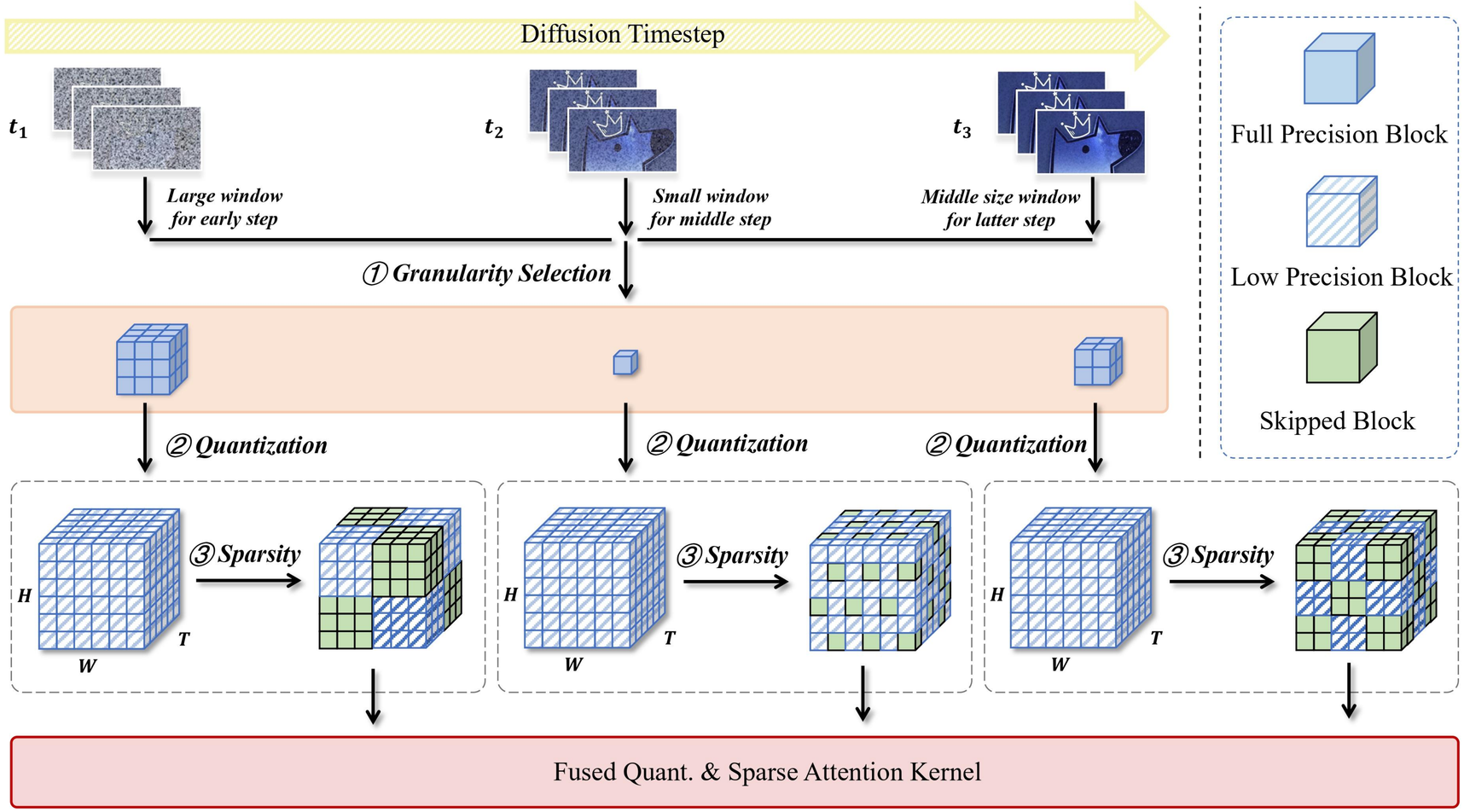
# 2. Method

**Chapter 02**

## Three core innovations:

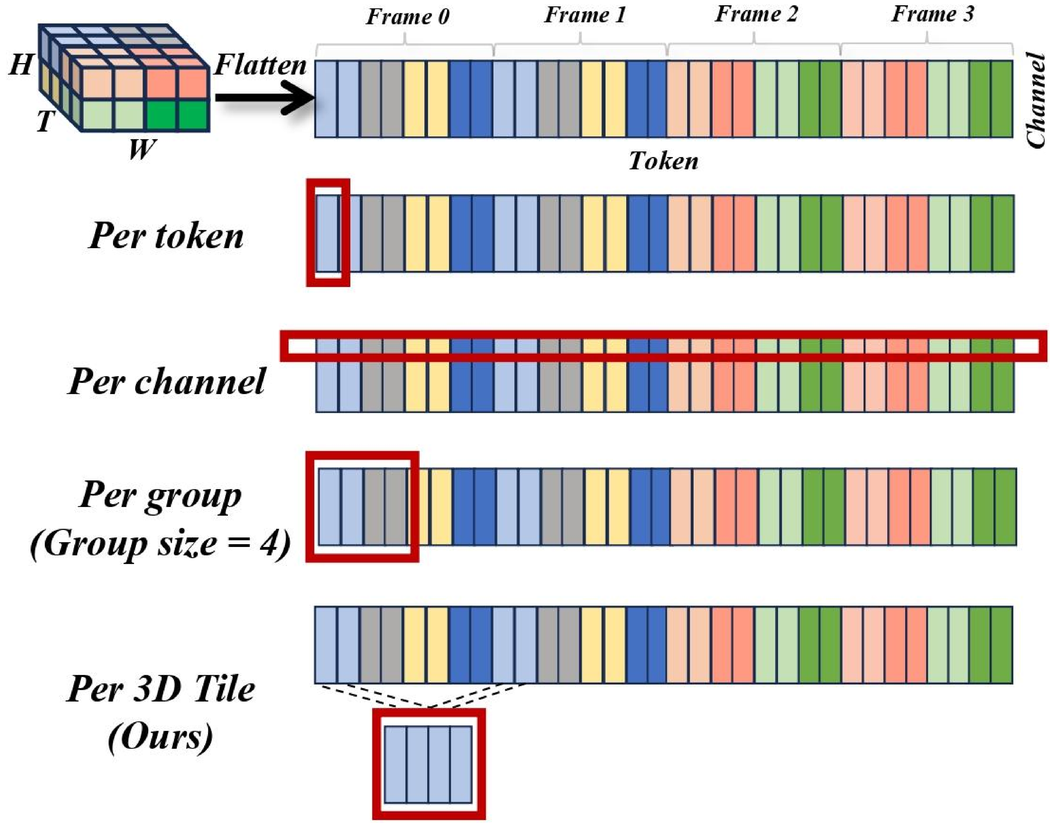
- Unified 3D tile-wise granularity for both quantization & sparsity.
- Denoising step-aware strategy adapting compression across diffusion timesteps.
- Hardware-optimized kernel leveraging FlashAttention & NVIDIA Hopper features.
- Achieves 7.09× kernel and 4.96× end-to-end speedup on Wan 2.1–14B without quality loss




Diffusion Timestep
$t_1$
$t_2$
$t_3$
Large window for early step
Small window for middle step
Middle size window for latter step
① Granularity Selection
② Quantization
② Quantization
② Quantization
③ Sparsity
③ Sparsity
③ Sparsity
H
W
T
Full Precision Block
Low Precision Block
Skipped Block
Fused Quant. & Sparse Attention Kernel

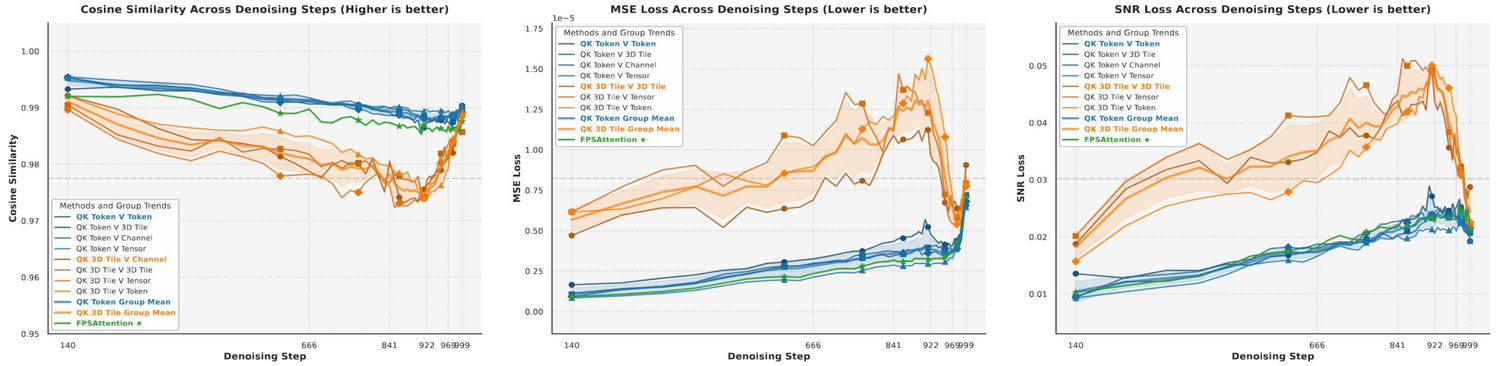
## Tile-wise FP8 Quantization:

- Divide Q/K/V into 3D tiles aligned with GPU compute units.
- Each tile quantized independently with local scaling factors.
- Preserves fine-grained activation dynamics, reduces memory cost 2×.
- Perfectly matches structured sparsity pattern and FlashAttention kernel



# Denoising Step-Aware Strategy

- Diffusion steps show varying sensitivity to compression errors.
- Early & late steps → tolerate coarse quantization & high sparsity.
- Mid steps → require fine precision & dense attention.
- Adaptive schedule aligns with diffusion dynamics




**(a)** Cosine similarity. **(b)** MSE loss. **(c)** SNR loss.

# 3. Experiments

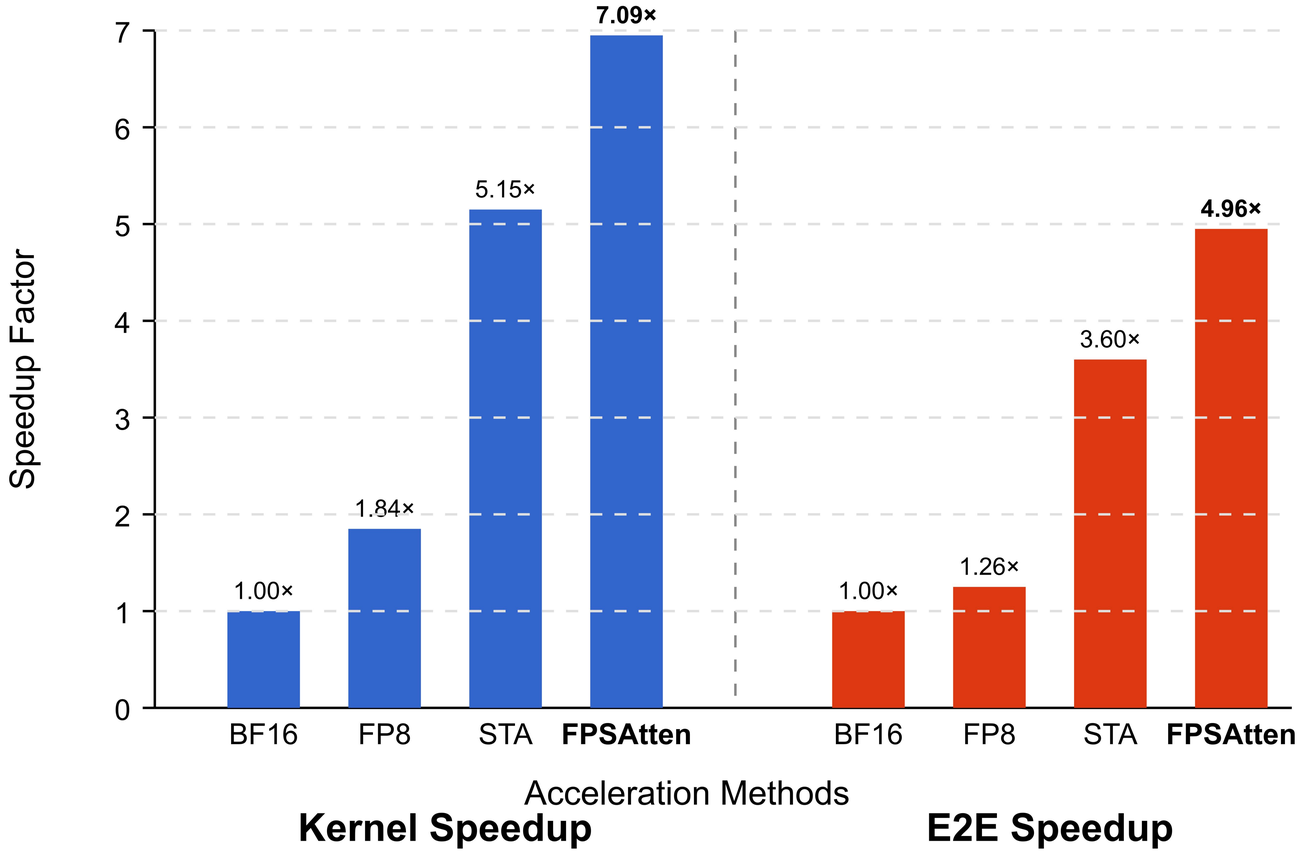
**Chapter 03**

# VBench Results

NIPS 2025

DAMO 开发者矩阵

| Method | Quality | | | | | Efficiency | | | |
|---|---|---|---|---|---|---|---|---|---|
| | PSNR ↑ | SSIM ↑ | LPIPS ↓ | ImageQual ↑ | SubConsist ↑ | FLOPS ↓ | Latency ↓ | Speedup ↑ | Speedup‡ ↑ |
| Wan2.1-1.3B† | - | - | - | 0.6708 | 0.9536 | 77.52 PFLOPS | 271s | 1.00x | - |
| SageAttention | 20.18990 | 0.78241 | 0.18811 | 0.6699 | 0.9453 | 37.61 PFLOPS | 141s | 1.91x | - |
| SpargeAtten | 17.72979 | 0.72628 | 0.26183 | 0.6541 | 0.8982 | 43.15 PFLOPS | 205s | 1.32x | - |
| SparseVideoGen | 19.51276 | 0.78891 | 0.20513 | 0.6729 | 0.9292 | 30.67 PFLOPS | 152s | 1.78x | - |
| STA | 18.78546 | 0.76335 | 0.23187 | 0.6626 | 0.8992 | 31.78 PFLOPS | 143s | 1.89x | - |
| Ours Quant | 20.99712 | 0.79820 | 0.15114 | 0.6798 | 0.9458 | 32.01 PFLOPS | 144s | 1.88x | - |
| Ours Quant + Sparse | 21.35417 | 0.80835 | 0.15398 | 0.7103 | 0.9338 | 32.01 PFLOPS | 110s | 2.45x | - |
| Wan2.1-14B† | - | - | - | 0.6715 | 0.9528 | 637.52 PFLOPS | 1301s | 1.00x | 1.00x |
| SageAttention | 24.33985 | 0.82283 | 0.15607 | 0.6724 | 0.9530 | 301.98 PFLOPS | 646s | 2.01x | 1.94x |
| SpargeAtten | 21.38291 | 0.81452 | 0.21723 | 0.6350 | 0.9173 | 339.30 PFLOPS | 734s | 1.77x | 2.12x |
| SparseVideoGen | 23.52881 | 0.80113 | 0.17032 | 0.6868 | 0.9489 | 259.79 PFLOPS | 613s | 2.12x | 3.13x |
| STA | 22.65635 | 0.82024 | 0.19283 | 0.6577 | 0.9530 | 264.34 PFLOPS | 548s | 2.37x | 3.60x |
| Ours Quant + Sparse | 25.74353 | 0.83171 | 0.07610 | 0.7103 | 0.9435 | 273.01 PFLOPS | 423s | 3.07x | 4.96x |


Speedup Factor
0
1
2
3
4
5
6
7
1.00×
1.84×
5.15×
7.09×
BF16
FP8
STA
FPSAtten
1.00×
1.26×
3.60×
4.96×
BF16
FP8
STA
FPSAtten
Acceleration Methods
Kernel Speedup
E2E Speedup

**Baseline** **Ours**

***Prompt:*** Snow rocky mountains peaks canyon. Snow blanketed rocky mountains surround and shadow deep canyons. The canyons twist and bend through the high elevated mountain peaks

***Prompt:*** Golden fish swimming in the ocean

**1 Training-aware co-design matters**

Jointly optimizing quantization and sparsity during training causes far less quality loss than post-hoc (training-free) compression — and unlocks much higher performance ceilings.

**2 Algorithm × Infrastructure synergy**

Real acceleration requires tight coupling between algorithm design and system/infrastructure implementation, rather than isolated model-level tricks.

**3 FP8 and sparse attention for future deployment**

FP8 quantization and structured sparse attention are key enablers for mobile inference and video world models, offering practical scalability to real applications.

Paper

Webpage


# THANKS

感谢聆听