Qualcomm

Attention Surgery: An Efficient Recipe to Linearize Your Video Diffusion Transformer


Mohsen Ghafoorian


Denis Korzhenkov


Amir Habibian

Qualcomm AI Research*

*Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

Introduction

You can turn linear causal attention into RNN!

Sadly linear attention is under-expressive! :(

Video Diffusion's Bottleneck

- DiTs dominate SOTA video diffusion models
- Quadratic self-attention (Softmax) complexity is the real bottleneck.

Self Attn. Cross Attn. FFN

$$\mathcal{O}(sb(\boxed{nd^2 + n^2} + \boxed{nn_k d + (n + n_k)d^2} + \boxed{nd^2}))$$

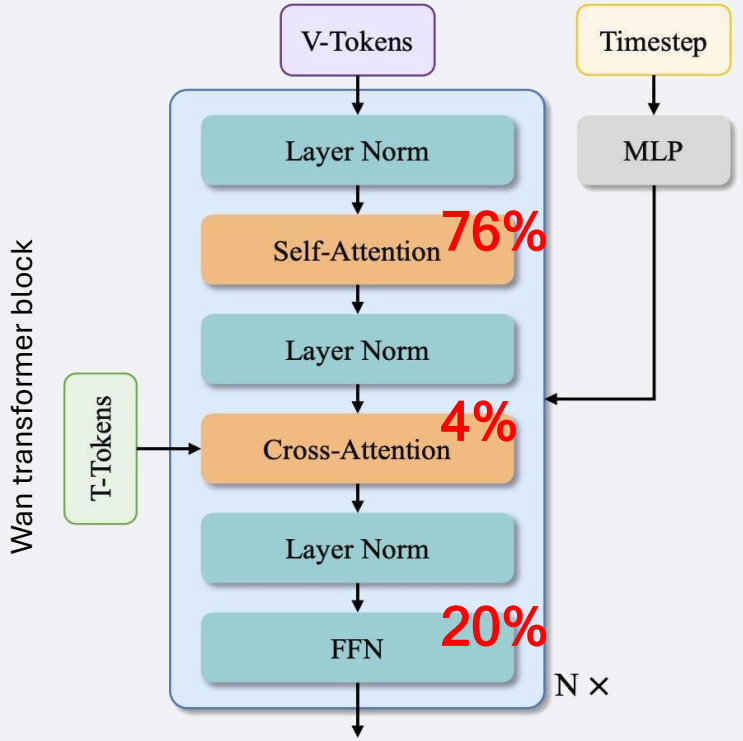
- n: #tokens, s: #denoising steps, d: feature dim, n_k: #context tokens, b: #blocks
- For Wan2.1 1.3B 480p n=~33k

- Linear attention[2]

$$V_i' = \frac{\sum_{j=1}^{N} \text{sim}(Q_i, K_j) V_j}{\sum_{j=1}^{N} \text{sim}(Q_i, K_j)}. \quad \text{sim}(q, k) = \exp\left(\frac{q^T k}{\sqrt{D}}\right)$$

$$V_i' = \frac{\sum_{j=1}^{N} \phi(Q_i)^T \phi(K_j) V_j}{\sum_{j=1}^{N} \phi(Q_i)^T \phi(K_j)}, \qquad V_i' = \frac{\phi(Q_i)^T \boxed{\sum_{j=1}^{N} \phi(K_j) V_j^T}}{\phi(Q_i)^T \boxed{\sum_{j=1}^{N} \phi(K_j)}}.$$

Terms that can be precomputed



Wan[1]2.1 1.3B 480p compute distribution

1. Team Wan et al. "Wan: Open and advanced large-scale video generative models." arXiv preprint arXiv:2503.20314 (2025).
2. Katharopoulos et al. "Transformers are RNNs: Fast autoregressive transformers with linear attention." ICML 2020.

Attention Surgery in a Glance

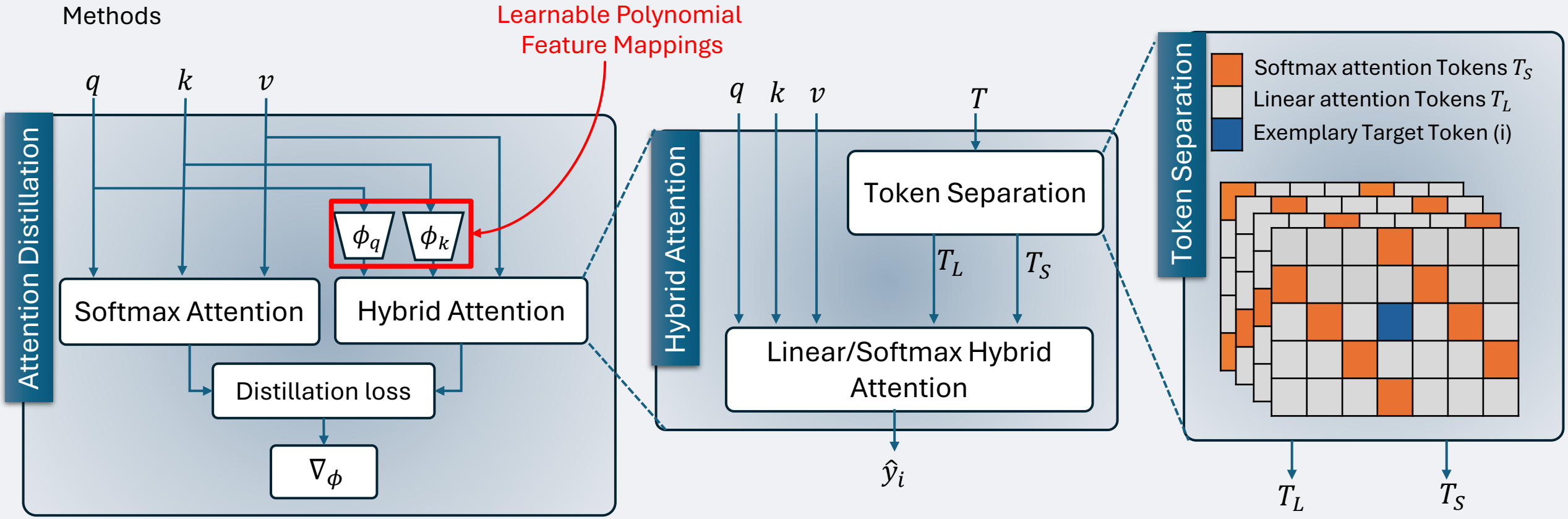
- What?
  - Combine:
    - Efficiency of the linear attention.
    - Expressiveness of Softmax.
  - Avoid the excessive cost of training models from scratch.
- How?
  - Leverage Hybrid attention, inspired by recent developments in LLMs[1]
  - Efficient tuning recipe to linearize/hybridize existing SOTA bidir. Softmax model (**Wan2.1** 1.3B[2])
    - Block-wise attention distillation.
    - Light-weight fine-tuning.
- Outcome?
  - Training within only **<400-gpu-hours**, 2 orders of magnitude faster than SANA-video.
  - Efficient hybrid model with negligible loss in quality

"An astronaut flying in space, Van Gogh style."

1. Zhang et al. "Lolcats: On low-rank linearizing of large language models." *ICLR* 2025.
2. Team Wan et al. "Wan: Open and advanced large-scale video generative models." arXiv preprint arXiv:2503.20314 (2025).

Attention Surgery

Methods



$$\mathcal{L}_{\text{ad}} = \log\left(1 + \left\| e^{q_i k_j^\top} - \phi_q(q_i)\phi_k(k_j)^\top \right\|_2^2\right)$$

Attention distillation

$$\mathcal{L}_{\text{vd}} = \left\| y - \hat{y} \right\|_1$$

Value distillation

Softmax Attention terms **Linear Attention terms**

$$\hat{y}_i = \frac{\sum_{j \in T_S} \exp\left(q_i k_j^\top / \sqrt{D} - c_i\right) v_j + \phi_q(q_i)\left(\sum_{j \in T_L} \phi_k(k_j)^\top v_j\right)}{\sum_{j \in T_S} \exp\left(q_i k_j^\top / \sqrt{D} - c_i\right) + \phi_q(q_i)\left(\sum_{j \in T_L} \phi_k(k_j)^\top\right)}$$

* Zhang et al. "Lolcats: On low-rank linearizing of large language models." *ICLR* 2025.

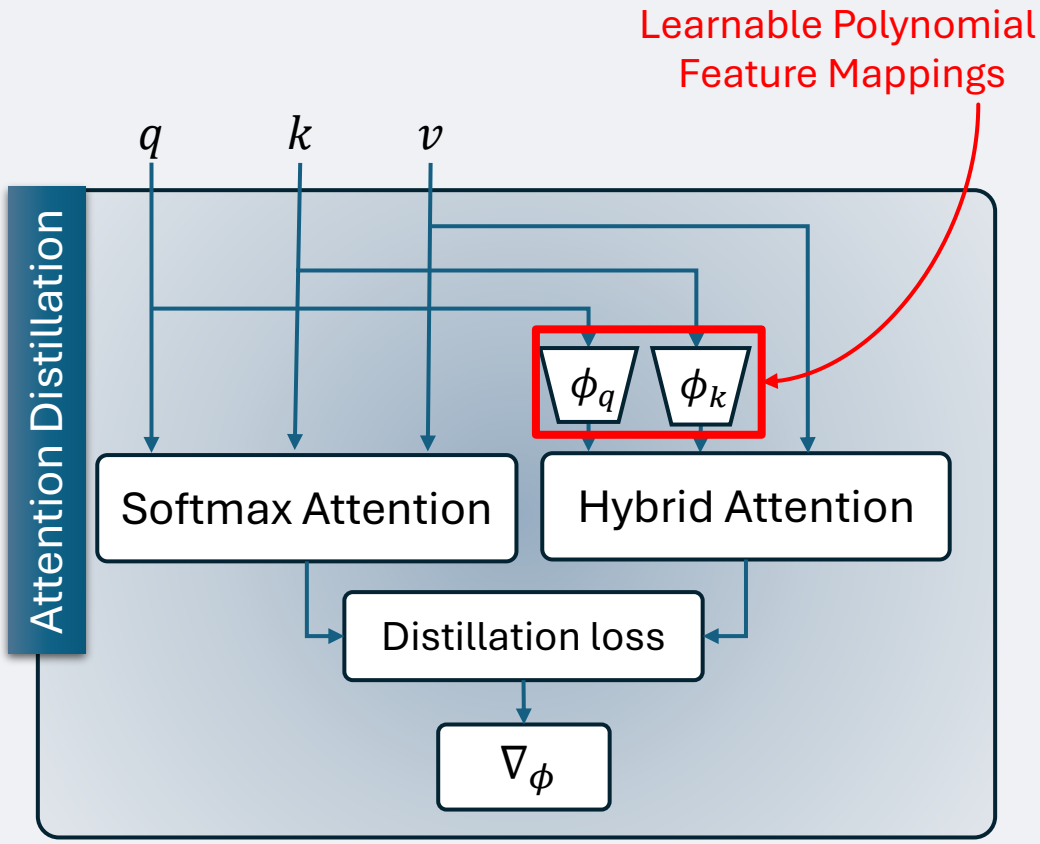
More on Polynomial Feature Mapping

- Design:
  - d-layer MLP, with degree-P polynomial:

$$\phi(x) = [\ (\psi_1(x))^1, (\psi_2(x))^2, \ldots, (\psi_P(x))^P\ ]^\top \in \mathbb{R}^{D'},$$

  - ReLU non-linearity on the last layer
    - To guarantee the non-negativity requirement of the kernel trick.
- Rationale
  - **Learnable**
    - Opposed to non-learnable elu(x)+1 in original attention
    - Increase the rank and expressiveness of linear branch
  - **Polynomial**:
    - Facilitate approximating the large dynamic range of exponential:

$$e^{q_i k_j^\top} \approx \phi(q_i)\phi(k_j)$$



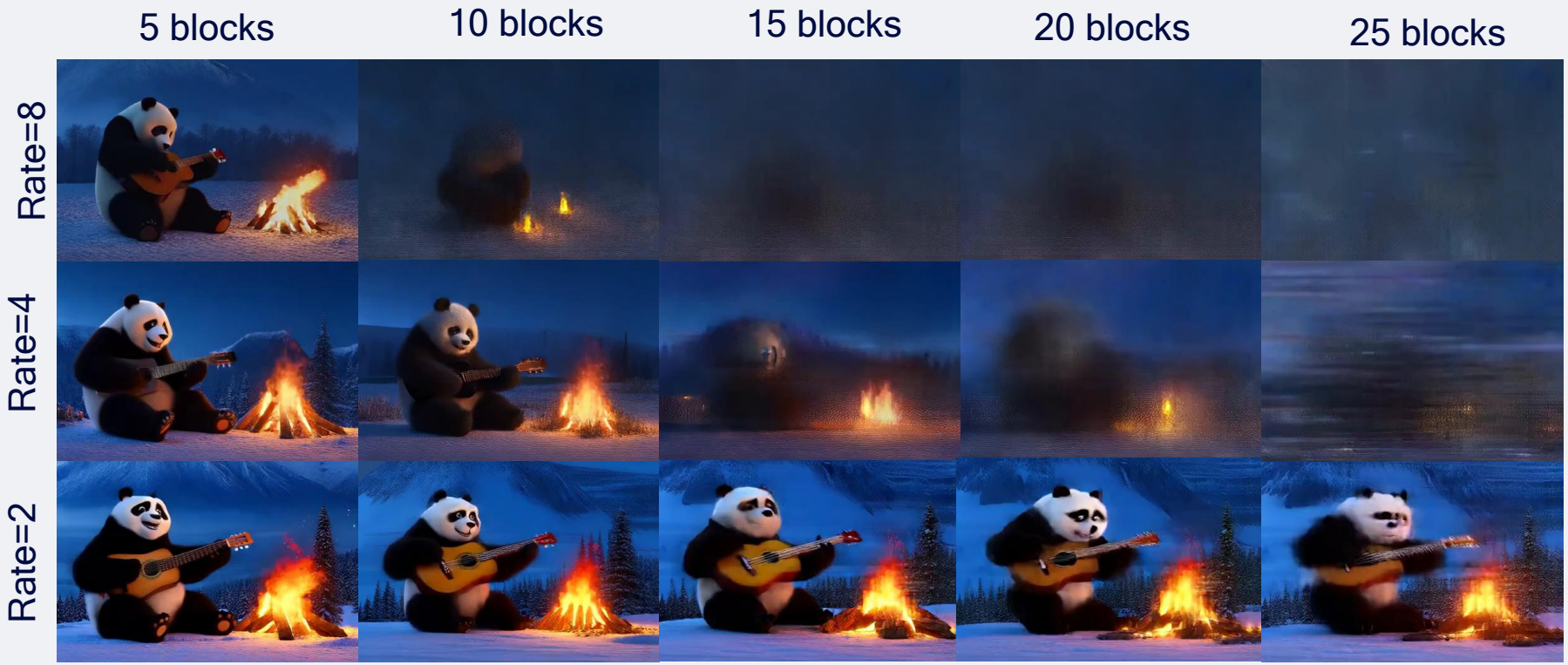
Algorithm 1 Attention Distillation for isolated attention layer l (Trainables $\boldsymbol{\phi} = (\phi_q, \phi_k)$)

Require: Teacher params Θ^{T}; Student params Θ_l^{S} of layer l with frozen weights except for $\boldsymbol{\phi} = (\phi_q, \phi_k)$; prompt distribution $\mathcal{P}$; noise distribution $\mathcal{N}$; sampling denoising steps set $\mathcal{T}$; batch size m; update repeats U; learning rate η;

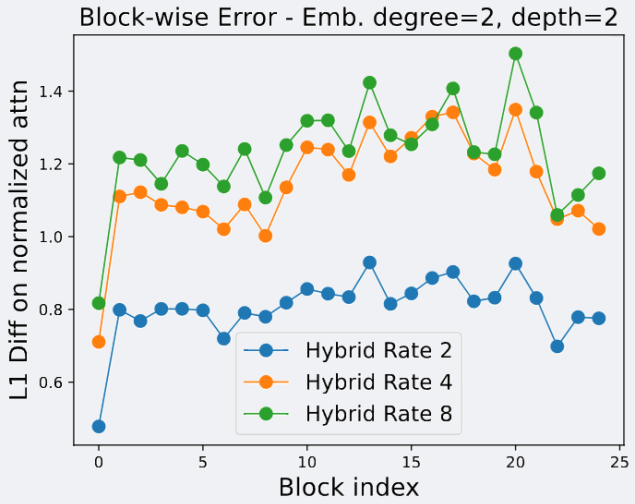
while not converged **do**
  Randomly sample prompts $\{p^{(n)}\}_{n=1}^{m} \sim \mathcal{P}$ and initial noises $\{\varepsilon_0^{(n)}\}_{n=1}^{m} \sim \mathcal{N}$
  ▷ **Cache teacher trajectories:**
  $\{\{(x_t^{(n)}, y_{t,l}^{(n)})\}_{t\in\mathcal{T}}\}_{n=1}^{m} \leftarrow \textsc{TeacherTrajectory}(\Theta^{\mathrm{T}}, \{(p^{(n)}, \varepsilon_0^{(n)})\}_{n=1}^{m}, \mathcal{T})$
  for $u = 1$ **to** U **do** ▷ Multiple student updates from the cached batch
    $\hat{y}_{t,l}^{(n)} \leftarrow \textsc{StudentAttn}\big(\Theta_l^{\mathrm{S}}, x_t^{(n)}\big)$ ▷ Using equation Eq. (5)
    $L \leftarrow \dfrac{1}{m\,|\mathcal{T}|} \sum_{n=1}^{m} \sum_{t\in\mathcal{T}} \big\| y_{t,l}^{(n)} - \hat{y}_{t,l}^{(n)} \big\|_1,$ ▷ Using Eq. (7) or Eq. (6)
    $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \eta \nabla_{\boldsymbol{\phi}} L$
  end for
end while
Return $\boldsymbol{\phi} = (\phi_q, \phi_k)$

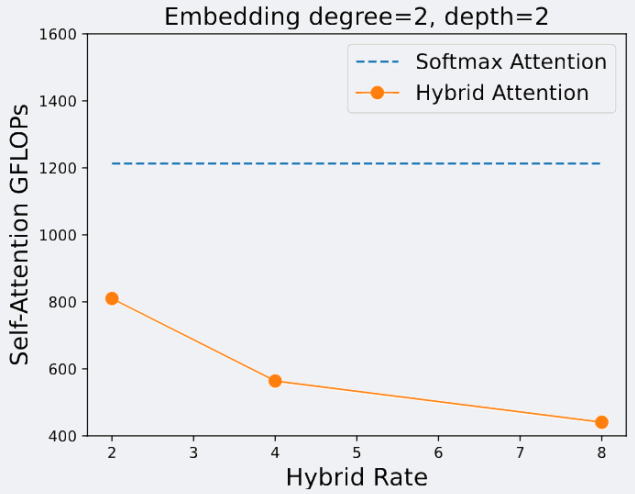
Attention distillation doesn't require video datasets; rather textual prompts only.

Qualitative Results After Pretraining Distillation



Heterogeneous Arch. & Overview





Residual Error of using rate r at i-th block

Compute Cost of using rate r at i-th block

$$\min_{\{z_{ir}\}} \sum_{i=1}^{B} \sum_{r \in \mathcal{R}} e_{ir} z_{ir} \quad \text{s.t.} \quad \sum_{i=1}^{B} \sum_{r \in \mathcal{R}} c_{ir} z_{ir} \leq \beta, \; \sum_{r \in \mathcal{R}} z_{ir} = 1 \; \forall i, \; z_{ir} \in \{0, 1\}.$$

Optimization var: Are we using rate r at i-th block?

Given Total Compute Budget

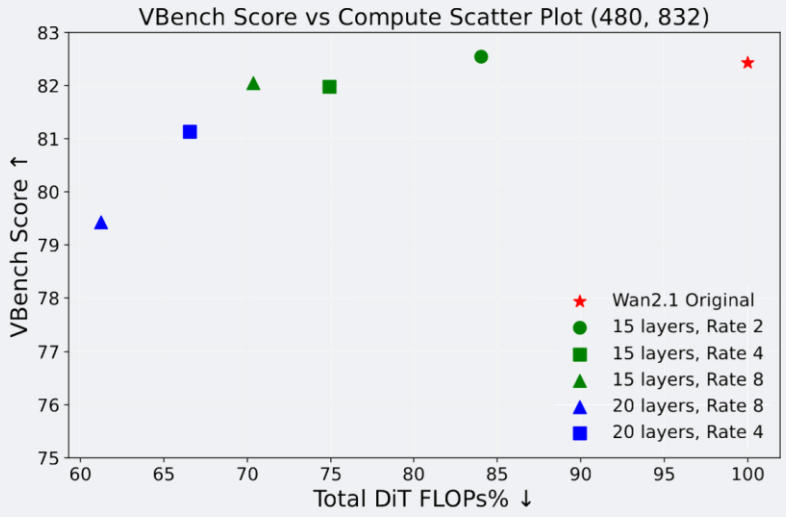
Multi-choice knapsack problem!

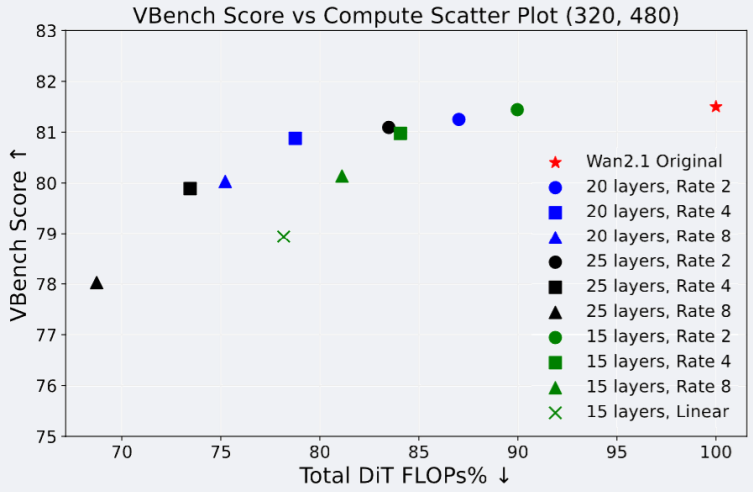
Overview

1. **Layer-wise Distillation** of the full Softmax attention activations or values to pretrain the hybrid attention Φ params
2. **Heterogenous Architecture Optimization**
3. **Light-weight Finetuning**

Quantitative Results

Quality/Compute Trade-off vs Wan2.1





Comparison to SOTA Efficient VDMs

Models with 2B–5B parameters	Total↑	Quality↑	Sem.↑
Open-Sora Plan V1.3 (Lin et al., 2024)	77.23	80.14	65.62
CogVideoX 5B (Yang et al., 2025)	81.91	83.05	77.33
CogVideoX1.5 5B (Yang et al., 2025)	82.01	82.72	79.17
Models up to 2B parameters			
Efficient VDiT Ding et al. (2025)	76.14	-	-
Open-Sora V1.2 Zheng et al. (2024)	79.76	81.35	73.39
LTX-Video (HaCohen et al., 2024)	80.00	82.30	70.79
SnapGenV (Wu et al., 2025b)	81.14	83.47	71.84
Hummingbird (Isobe et al., 2025)	81.35	83.73	71.84
MVDiT - Mobile (Wu et al., 2025a)	81.45	83.12	74.76
CogVideoX 2B (Yang et al., 2025)	81.55	82.48	77.81
PyramidalFlow (Jin et al., 2025)	81.72	84.74	69.62
M4V (Huang et al., 2025)	81.91	83.36	76.10
STA (Zhang et al., 2025d)	83.00	85.37	73.52
VSA (Zhang et al., 2025c)	82.77	83.60	79.47
SANA-Video (Chen et al., 2025)	83.71	84.35	81.35
Wan2.1 1.3B (Wan et al., 2025)	83.31	85.23	75.65
Wan2.1 1.3B* (Wan et al., 2025)	83.10	85.10	75.12
+ Attention Surgery (15×R2)	83.21	85.19	75.25

VBench

Model	VBench-2.0					
	Total↑	Hum.Fid.↑	Creativity↑	Control.↑	Com.sense↑	Physics↑
Wan2.1 1.3B	56.0	80.7	48.7	34.0	63.4	53.8
CogVideoX-1.5 5B	53.4	72.1	43.7	29.6	63.2	48.2
Attn. Surgery 15×R2	55.1	78.9	47.5	33.4	63.1	52.8

VBench-2.0

Human Preference Study

Prompt Dimension	Preference %		
	Ours	No pref.	Wan2.1
Appearance Style	51.8	19.6	28.6
Color	52.2	21.7	26.1
Human Action	16.2	54.1	29.7
Object Class	30.3	30.3	39.4
Overall Consistency	30.5	45.8	23.7
Scene	10.0	40.0	50.0
Spatial Relationship	43.9	33.3	22.8
Subject Consistency	35.7	21.4	42.9
Temporal Flickering	20.7	56.0	23.3
Temporal Style	28.3	39.1	32.6
Total	31.0	39.7	29.3

Overview & Ablations

Importance of Distillation Pret

Converted Blocks	Attention	Distillation	Total↑	Quality↑	Semantic↑
15	Linear	×	59.7	69.7	20.0
15	Linear	✓	**78.9**	**82.2**	**65.9**
20	Hybrid $R = 8$	×	77.3	80.2	65.9
20	Hybrid $R = 8$	✓	**80.0**	**81.7**	**73.2**

Impact of Heter. Architecture Opt.

	Block selection	Total↑	Quality↑	Semantic↑
15×R4	homogeneous	81.0	82.4	75.1
	heterogeneous	**81.9**	**83.2**	**76.4**
20×R4	homogeneous	80.9	81.9	**76.61**
	heterogeneous	**81.1**	**82.5**	75.2

	Block selection	Total↑	Quality↑	Semantic↑
20×R8	homogeneous	80.0	81.7	73.2
	heterogeneous	**80.9**	**82.3**	**75.4**
25×R4	homogeneous	79.9	81.1	**75.1**
	heterogeneous	**80.2**	**82.1**	72.5

Impact of Linear Kernel Arch.

ϕ Specification		VBench Total↑			Parameters ↓	FLOPs ↓
Poly. degree	MLP layers	10×R8	15×R8	20×R8	(M)	(G)
6	3	82.1	80.8	**80.3**	15.4	387
4	2	**82.3**	81.6	**80.3**	3.9	70
3	2	**82.3**	**81.9**	79.8	2.4	60
2	2	82.1	81.5	80.2	**1.2**	**30**

*"A boat sailing leisurely along the Seine River with the Eiffel Tower in background, **black and white**.."*

"A cute fluffy panda eating Chinese food in a restaurant"

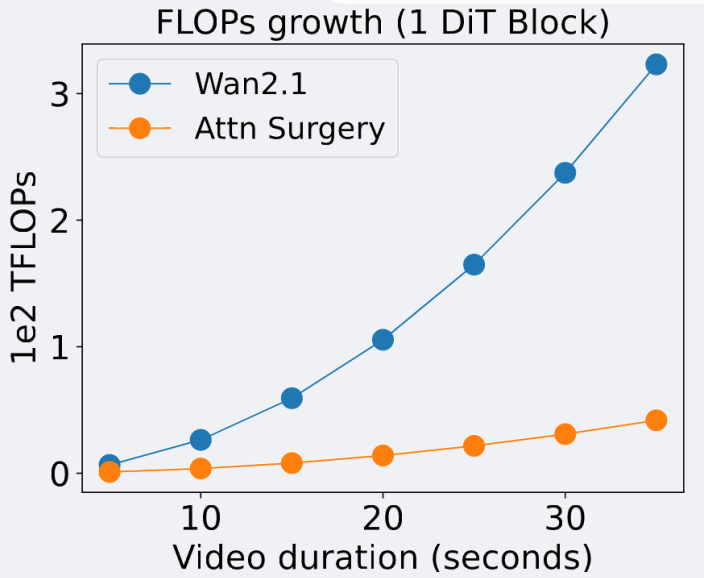
Without Distillation

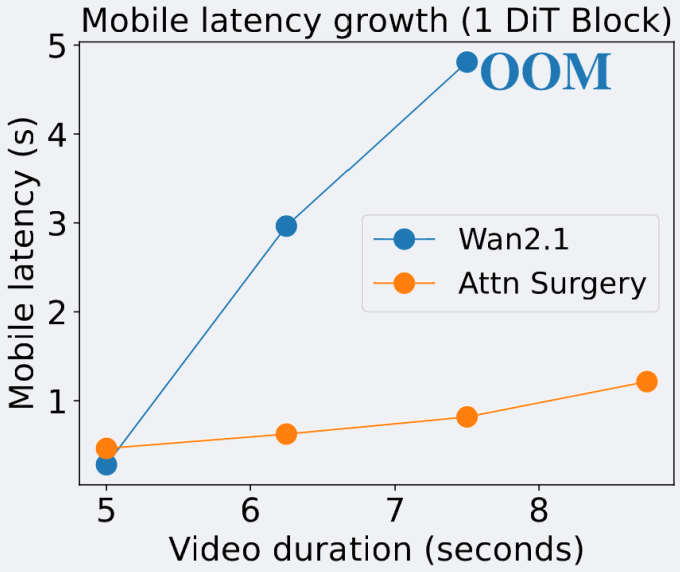
With Distillation

15xLinear

20xR8

Scaling Behavior and On-device Measurements





Attention Block	Number of frames - Memory Read/Write (GB)							
	81		101		121		141	
	W	R	W	R	W	R	W	R
Softmax Flash Attention	5.1	6.0	12.9	16.4	22.7	53.6	OOM	OOM
HedgeHog Linear Attention	5.7	8.1	7.0	10.1	6.9	11.3	8.0	13.2
Attention Surgery - R8	6.3	10.1	5.2	10.9	6.4	13.2	7.8	35.2

Total Memory Read/Write of 1 DiT Block

* Measurements on Snapdragon 8 Gen 4