Scaling Transformers: Architectures, Longer Contexts, Better Data

Juan Carlos Niebles

Scaling Transformers

Train new Architecture Designs

Inference for Long/Streaming Video

Dataset & Benchmarking Visual Generation

Scaling Transformers

Train new Architecture Designs

Inference for Long/Streaming Video

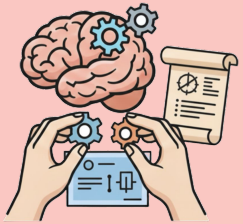
Dataset & Benchmarking Visual Generation

Pretraining is expensive!

Pretraining

Cost

Handcrafted Designs



AlexNet, ResNet, Transformers

Automated Methods

NASNet, EfficientNet, EVOLUTION, STAR

Pretraining is expensive!

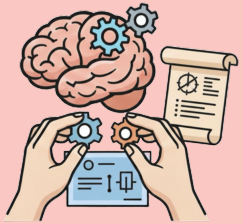
Pretraining

Synthetics/ Small-scale

Assumptions don't always hold

Zoology, MAD

Handcrafted Designs



AlexNet, ResNet, Transformers

Automated Methods

NASNet, EfficientNet, EVOLUTION, STAR

Cost

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Pretraining is expensive!

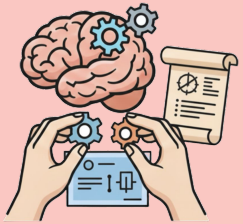
Pretraining

Synthetics/ Small-scale

Assumptions don't always hold

Zoology, MAD

Handcrafted Designs



AlexNet, ResNet, Transformers

Automated Methods

NASNet, EfficientNet, EVOLUTION, STAR

WHAT DO WE WANT?

New (large-scale) model architecture designs

Small compute budgets

Cost

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

~~Pretraining is expensive!~~ New model architectures Post-training?

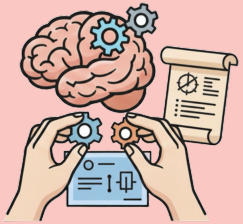
Pretraining

Synthetics/ Small-scale

Assumptions don't always hold

Zoology, MAD

Handcrafted Designs



AlexNet, ResNet, Transformers

Automated Methods

NASNet, EfficientNet, EVOLUTION, STAR

Post-training

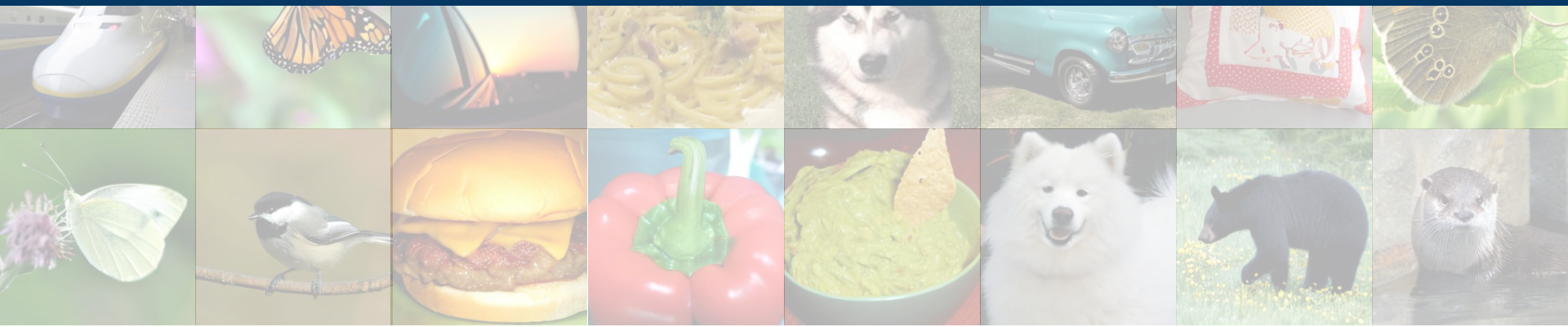
Can we <u>derive</u> new model architectures using pretrained models under small compute budgets?

Cost

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Samples generated using Hybrid architectures obtained via grafting (Chandrasegaran et al., 2025)
Chandrasegaran, Keshigeyan, et al. "Exploring Diffusion Transformer Designs via Grafting." *NeurIPS 2025 Oral*

Grafting = Model Architecture Editing



[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Diffusion x Transformers

Image

Video

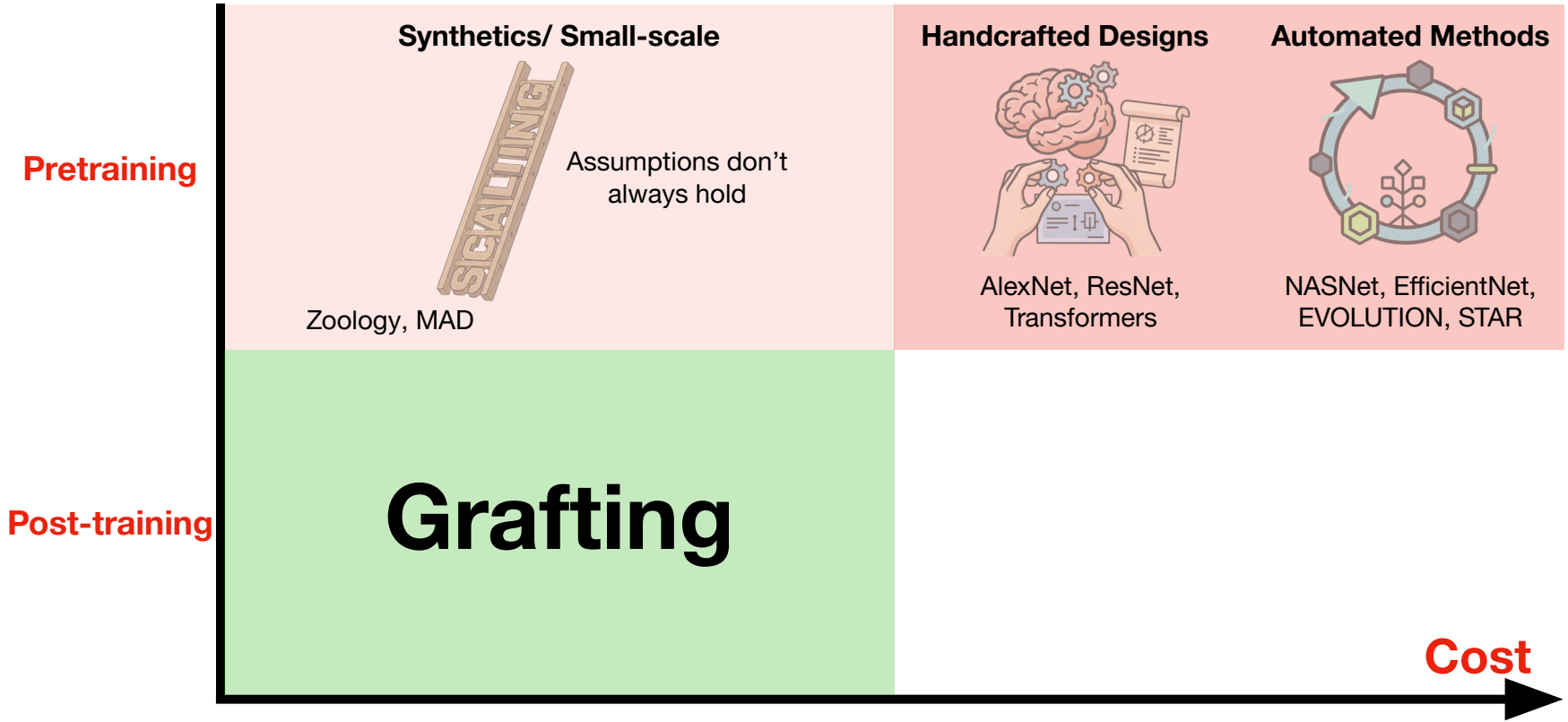
Boat fry bear
rock tree ABCs
thing sleep
boat tennis

Text

Peebles, William, and Saining Xie. "Scalable diffusion models with transformers." ICCV 2023
Brooks, Tim, et al. “Video Generation Models as World Simulators.” *OpenAI*
Lou, Aaron, Chenlin Meng, and Stefano Ermon. "Discrete Diffusion Modeling by Estimating the Ratios of the Data Distribution." *ICML 2024*

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

~~Pretraining is expensive!~~ New model architectures Post-training?

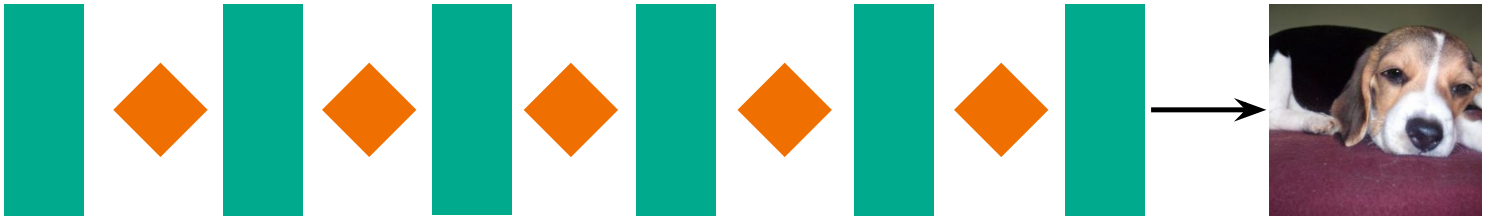


[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting: How to edit a computational graph?

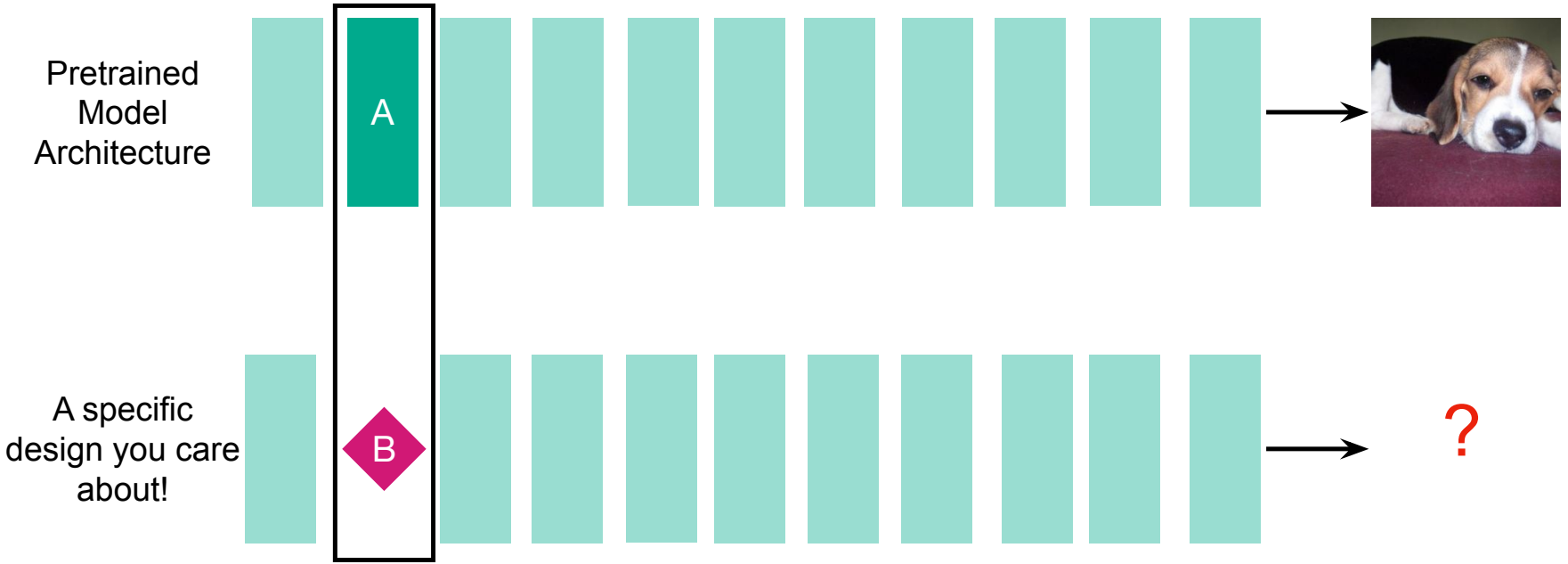
Pretrained Model Architecture

A specific design you care about!



[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Core problem: How to edit a computational graph?

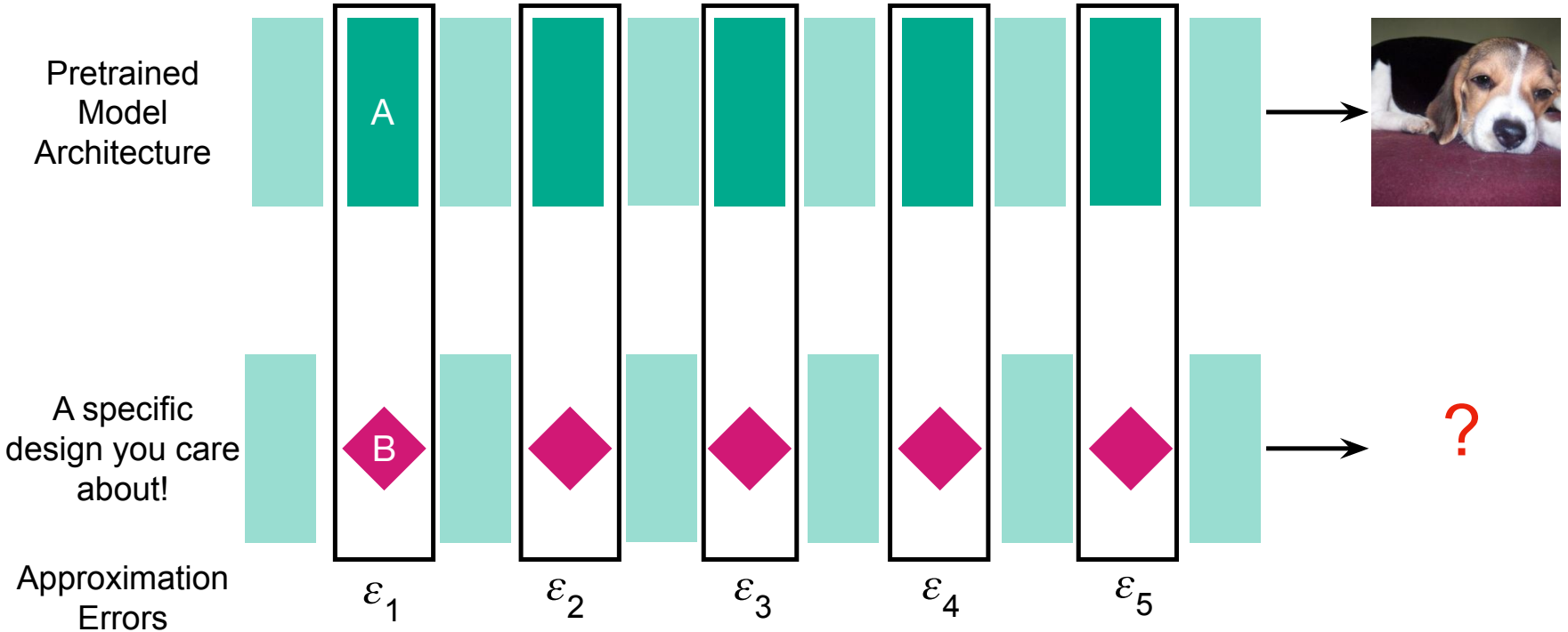


(Q1) Operator initialization

How to initialize B such that B ≅ A

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Core problem: How to edit a computational graph?



(Q2) Error propagation

How to mitigate $\varepsilon_1, \varepsilon_2, \ldots\ldots, \varepsilon_n$?

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting is a simple two-stage approach to architecture editing

(Q1) Operator initialization

How to initialize 

such that 

≅ 

(Q2) Error propagation

How to mitigate $\varepsilon_1, \varepsilon_2, \ldots\ldots, \varepsilon_n$?

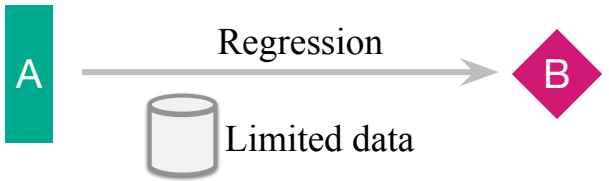
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting is a simple two-stage approach to architecture editing

(Q1) Operator initialization

How to initialize B such that B ≅ A

(Stage 1) Activation Distillation



(Q2) Error propagation

How to mitigate $\varepsilon_1, \varepsilon_2, \ldots\ldots, \varepsilon_n$?

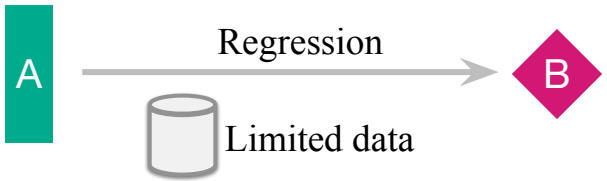
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting is a simple two-stage approach to architecture editing

(Q1) Operator initialization

How to initialize B such that B ≅ A

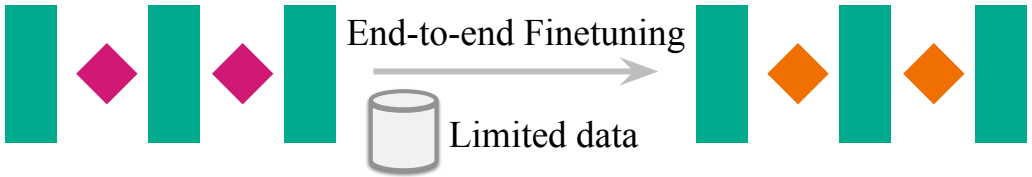
(Stage 1) Activation Distillation



(Q2) Error propagation

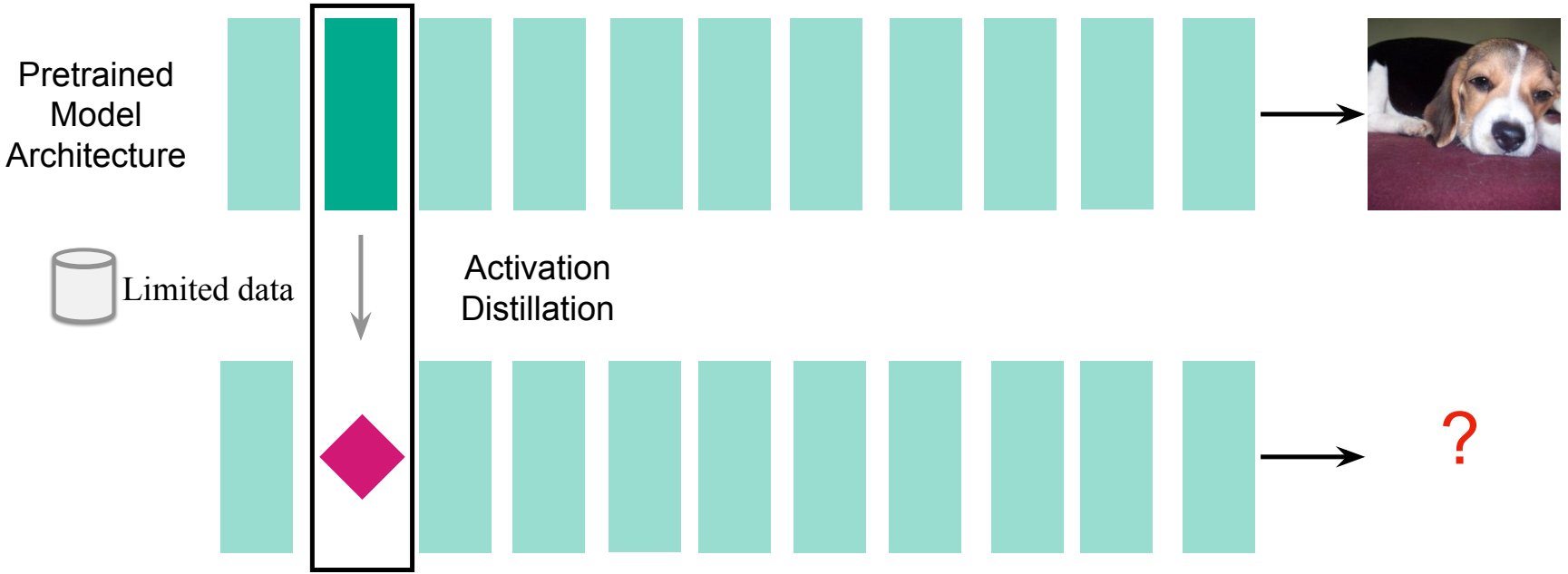
How to mitigate $\varepsilon_1, \varepsilon_2, \ldots\ldots, \varepsilon_n$?

(Stage 2) Lightweight Finetuning



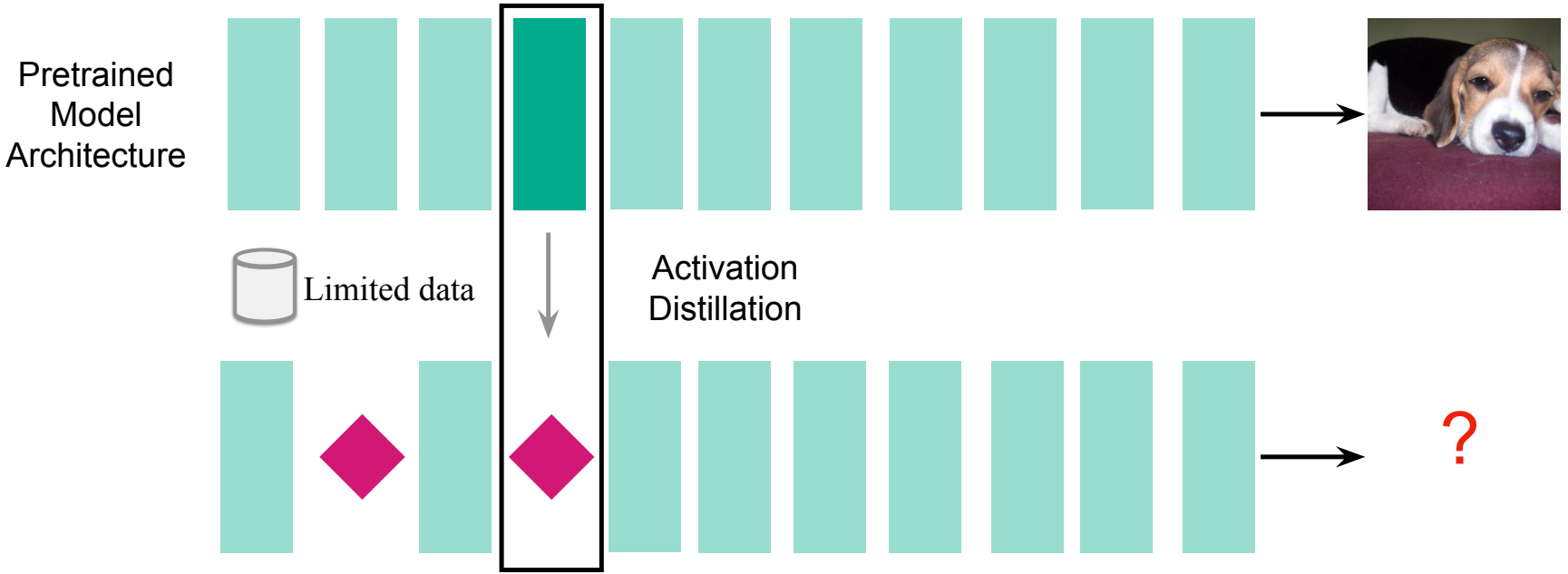
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting Stage 1 - Activation Distillation



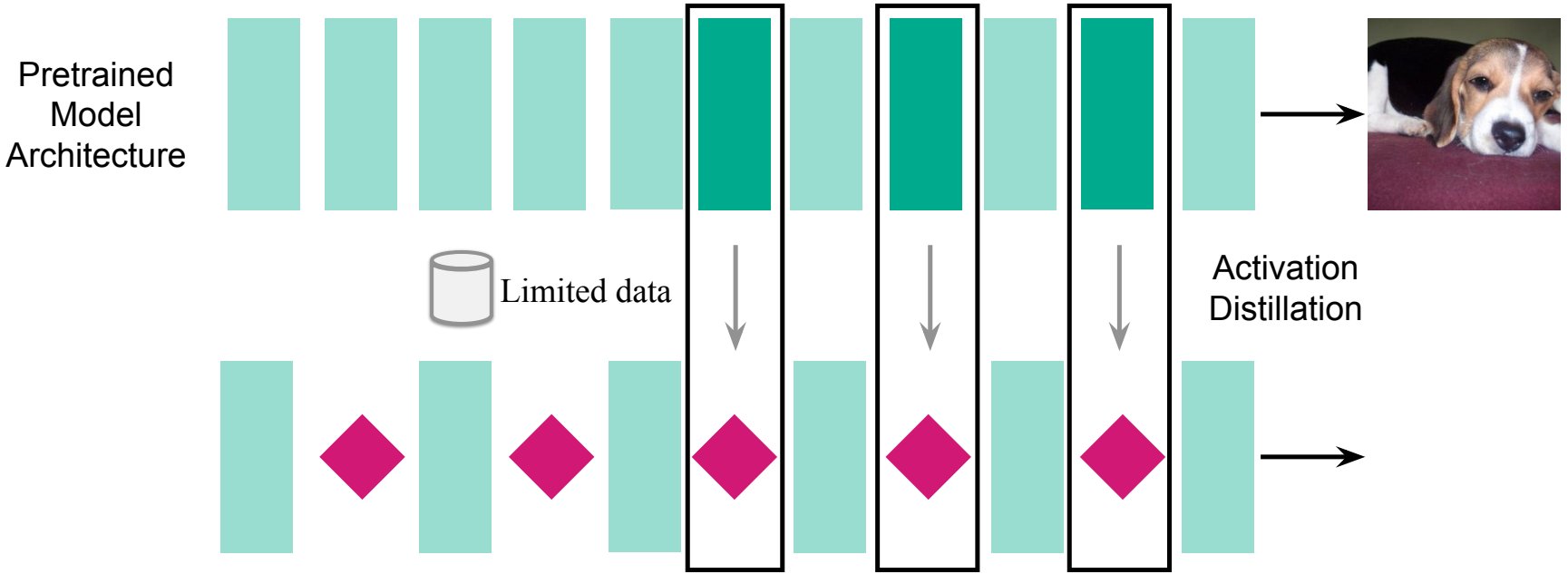
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting Stage 1 - Activation Distillation



[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

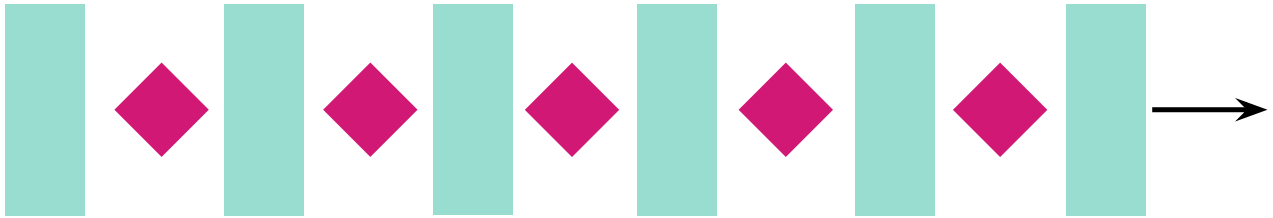
Grafting Stage 1 - Activation Distillation



Initialize in parallel!

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting Stage 2 - Lightweight finetuning



[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

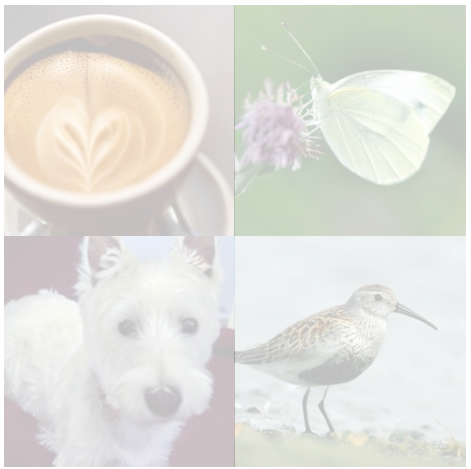
Grafting - Stage 2

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Results

Efficient Hybrid Architectures for Class-conditional Image Generation

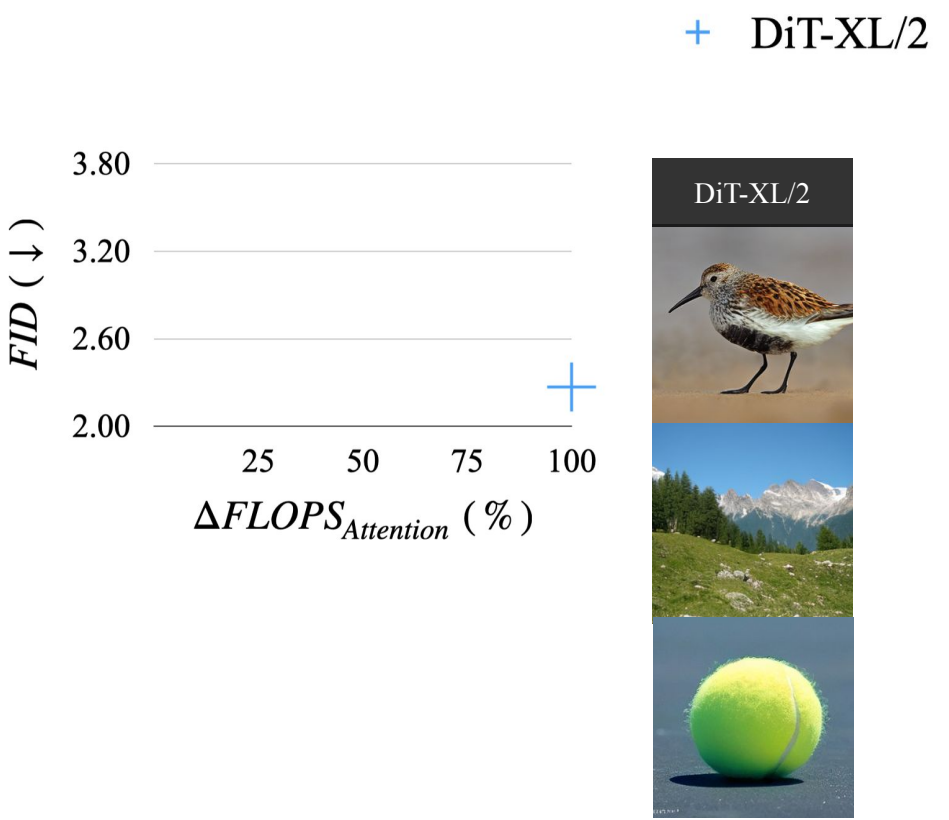
Faster Text-to-Image Generation



Restructure Model Architectures (e.g., Convert Model Depth to Width)

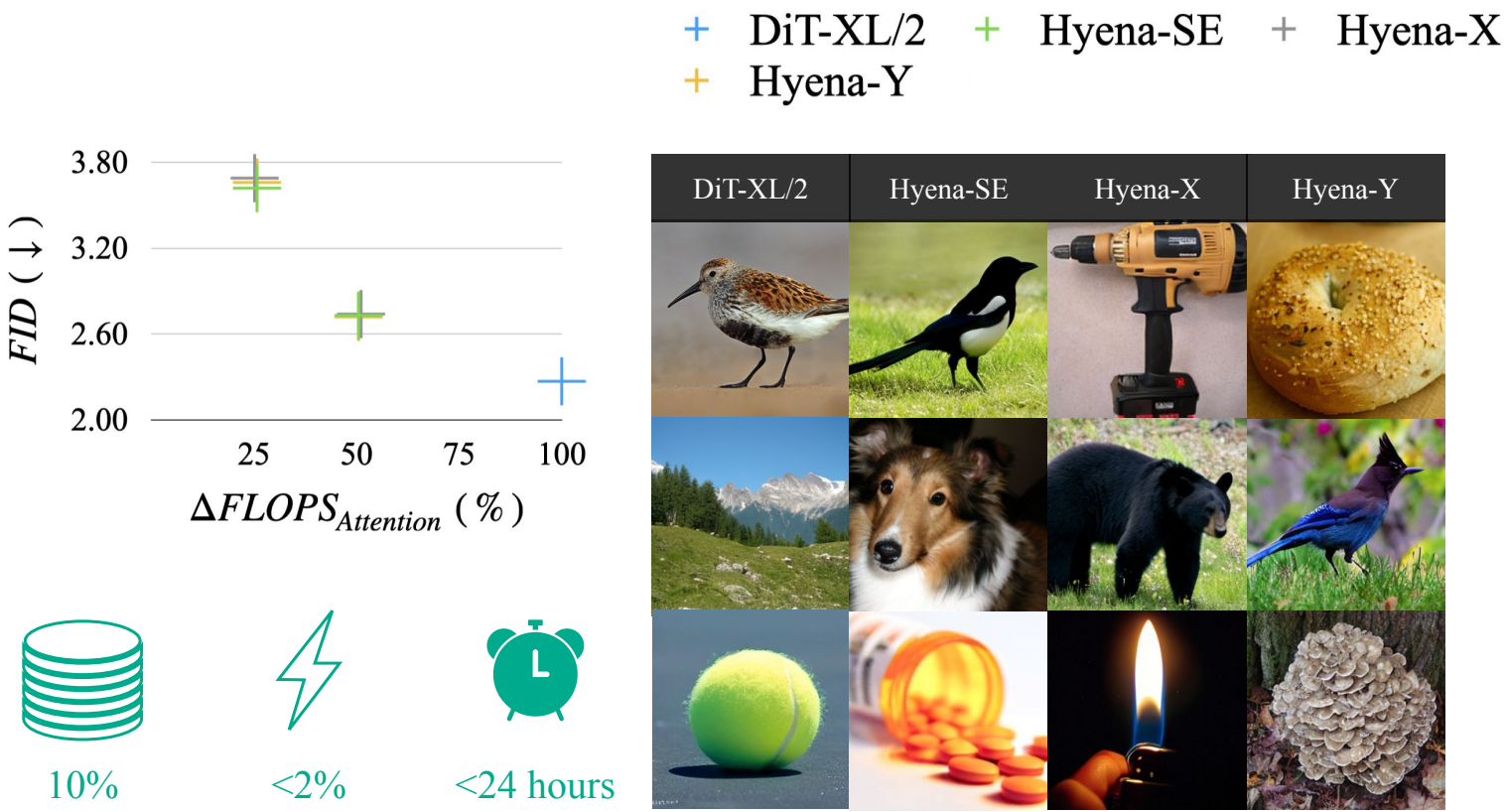
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

[MHA] Experiments I: Image Generation with Hybrid Architectures obtained via Grafting



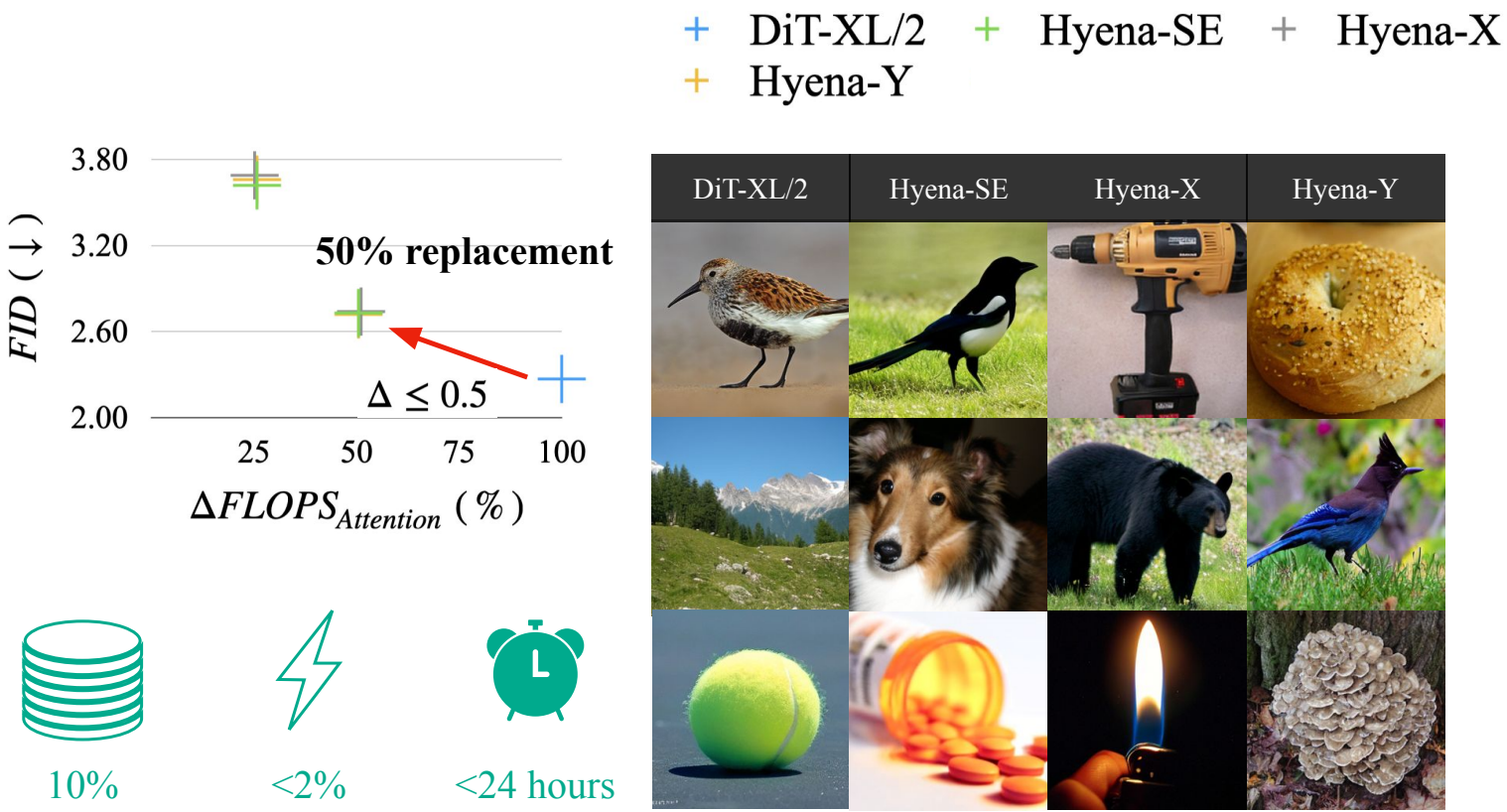
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

[MHA] Experiments I: Image Generation with Hybrid Architectures obtained via Grafting



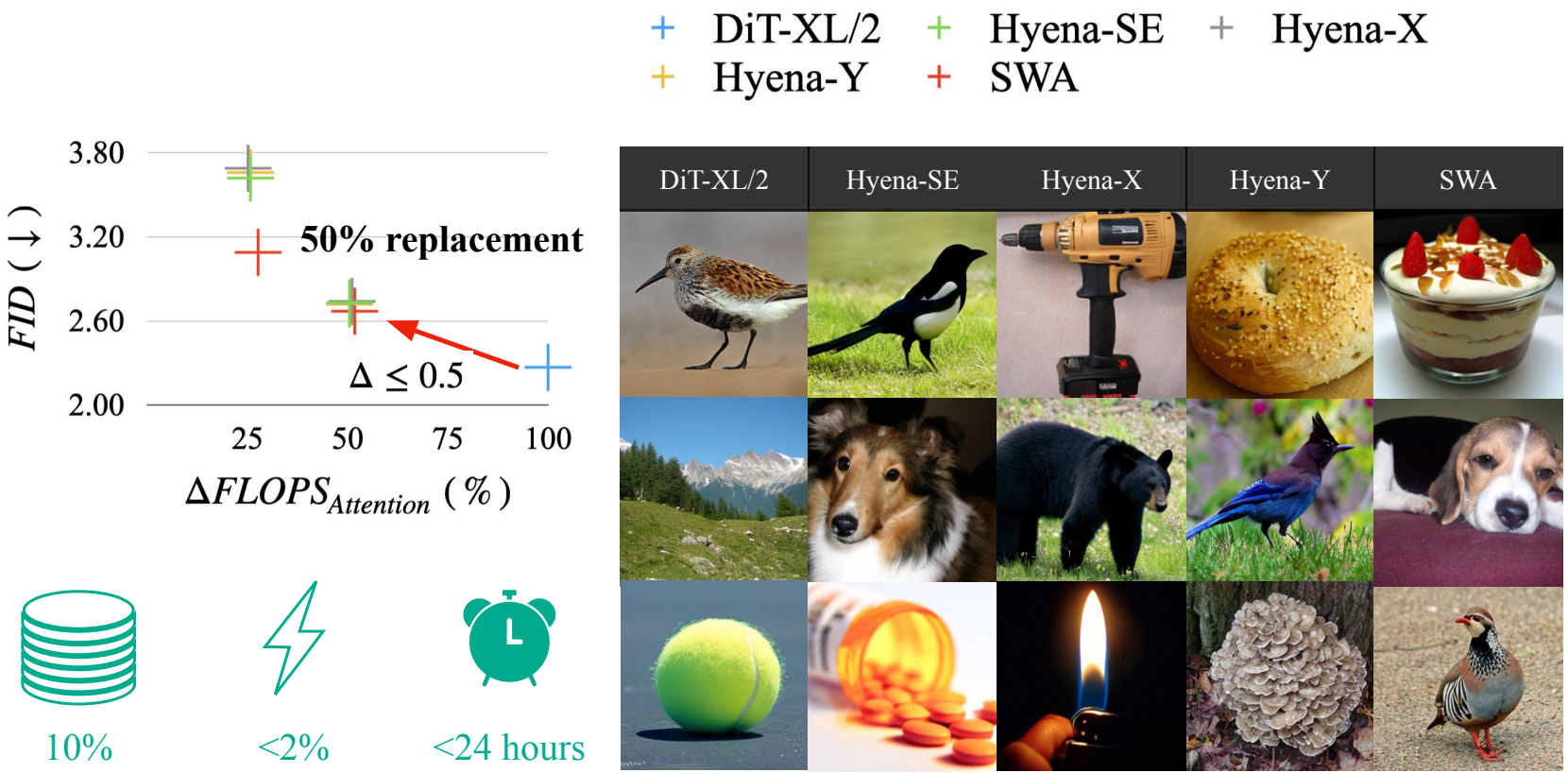
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

[MHA] Experiments I: Image Generation with Hybrid Architectures obtained via Grafting



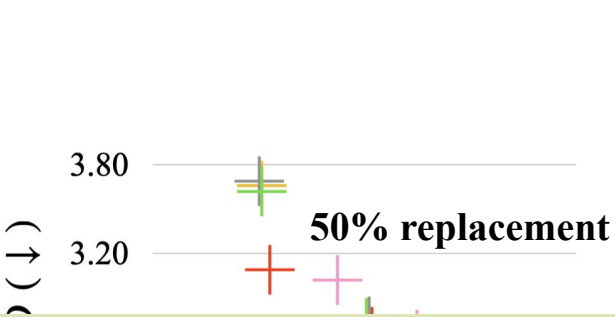
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

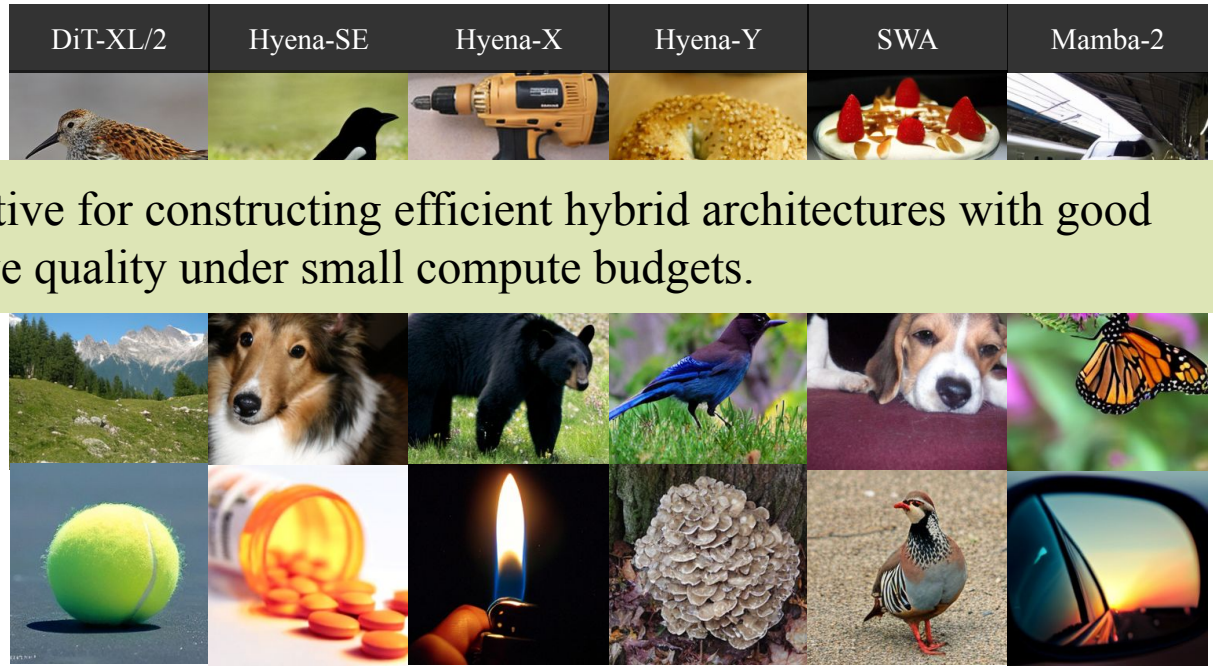
[MHA] Experiments I: Image Generation with Hybrid Architectures obtained via Grafting



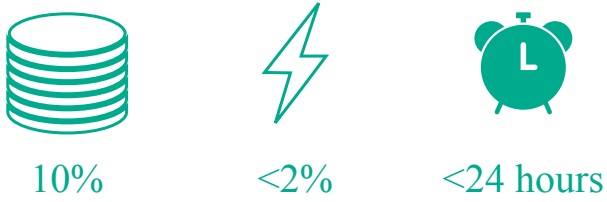
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

[MHA] Experiments I: Image Generation with Hybrid Architectures obtained via Grafting





Takeaway 1: Grafting is effective for constructing efficient hybrid architectures with good generative quality under small compute budgets.

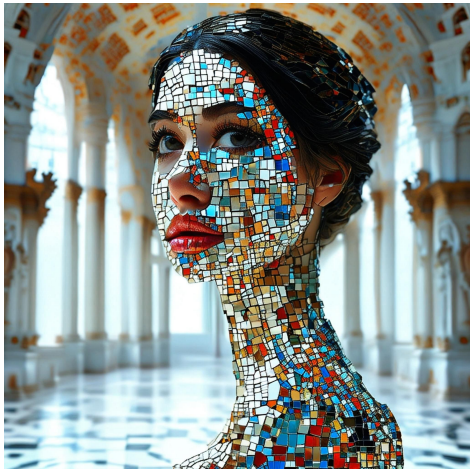


In practice, Mamba-2 is expensive due to projections (Sequence length=256 in this setup)

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Experiments

Efficient Hybrid Architectures for Class-conditional Image Generation

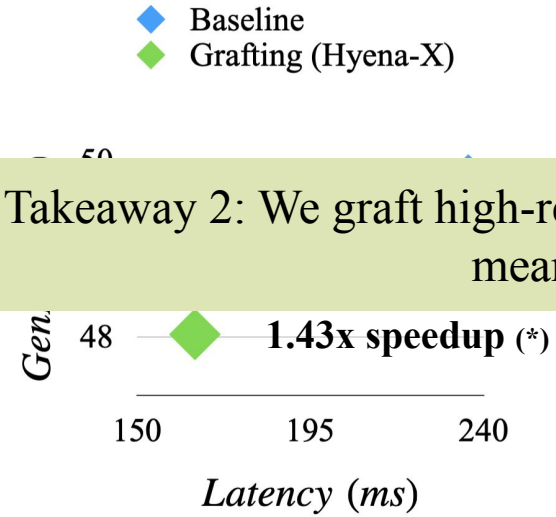


Faster Text-to-Image Generation

Restructure Model Architectures (e.g., Convert Model Depth to Width)

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Experiments II: Grafting Text-to-Image Diffusion Transformers



Takeaway 2: We graft high-resolution text-to-image DiTs, constructing hybrid architectures with meaningful speedups and minimal quality drop.

(*) Speedup measured @ Batch size=2 / H100

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Experiments

Efficient Hybrid Architectures for Class-conditional Image Generation

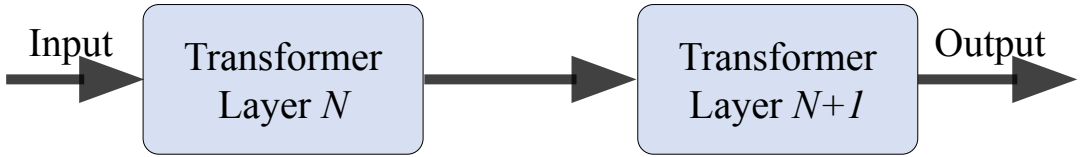
Faster Text-to-Image Generation

Restructure Model Architectures (e.g., Convert Model Depth to Width)

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Case study: Converting Model Depth to Width via Grafting

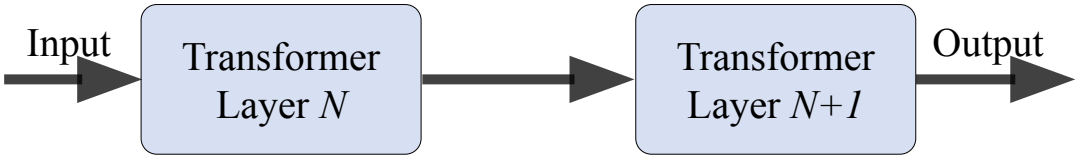
(a) Sequential Transformer Blocks



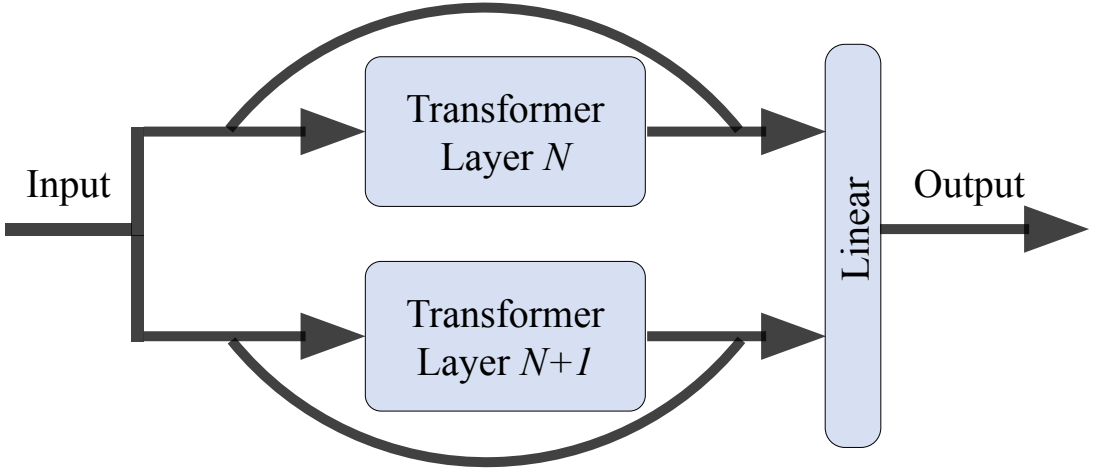
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Case study: Converting Model Depth to Width via Grafting

(a) Sequential Transformer Blocks

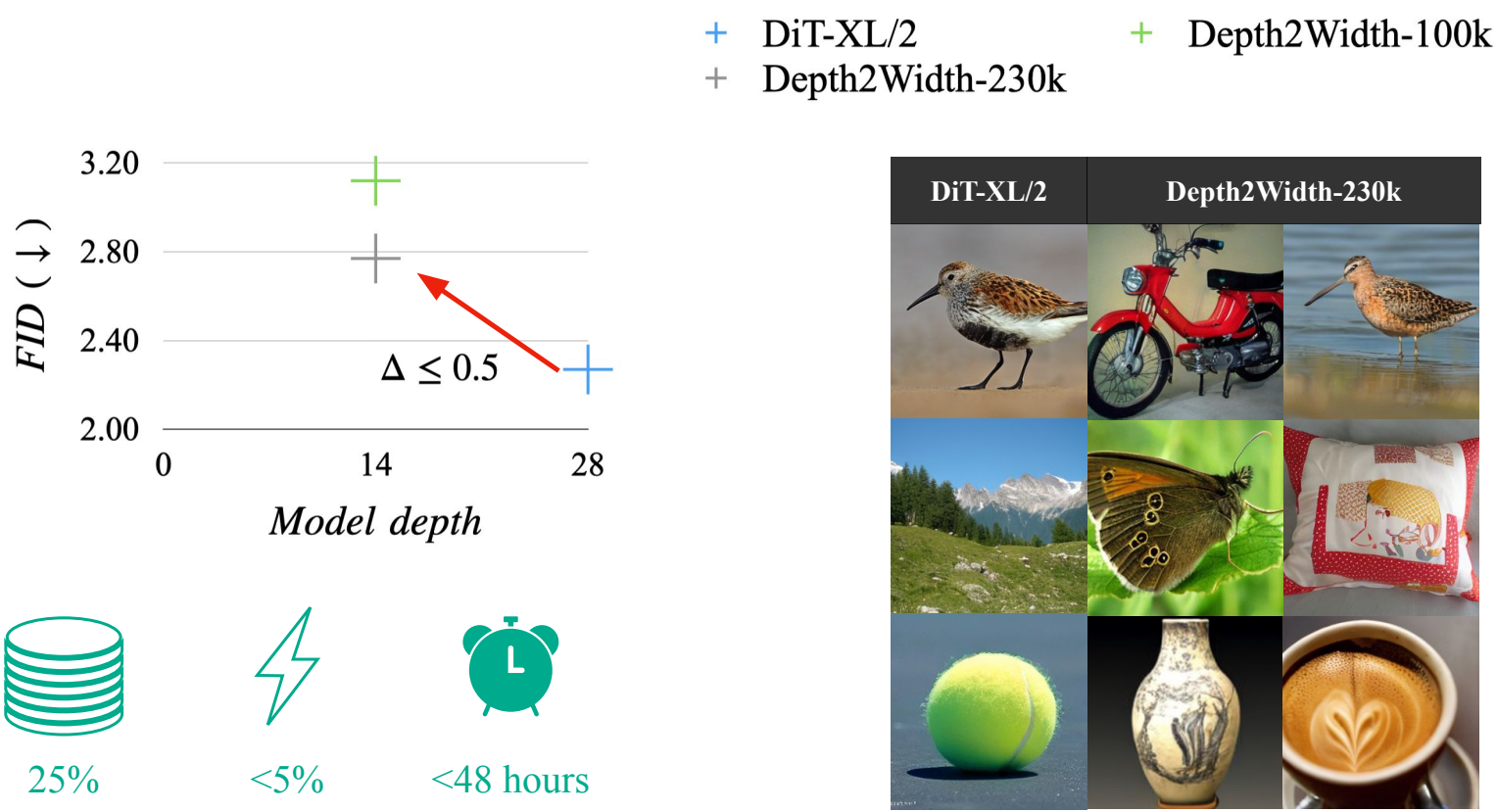


(b) Rewiring Transformer Blocks in Parallel via Grafting



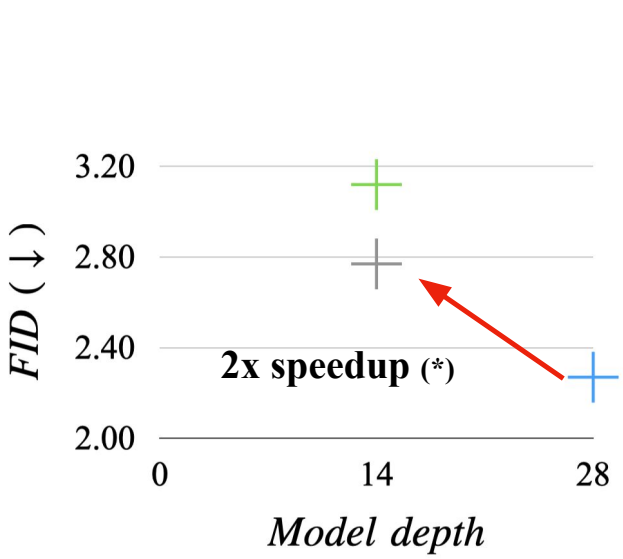
[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

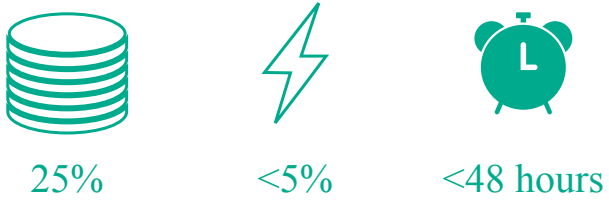
Case study: Converting Model Depth to Width via Grafting

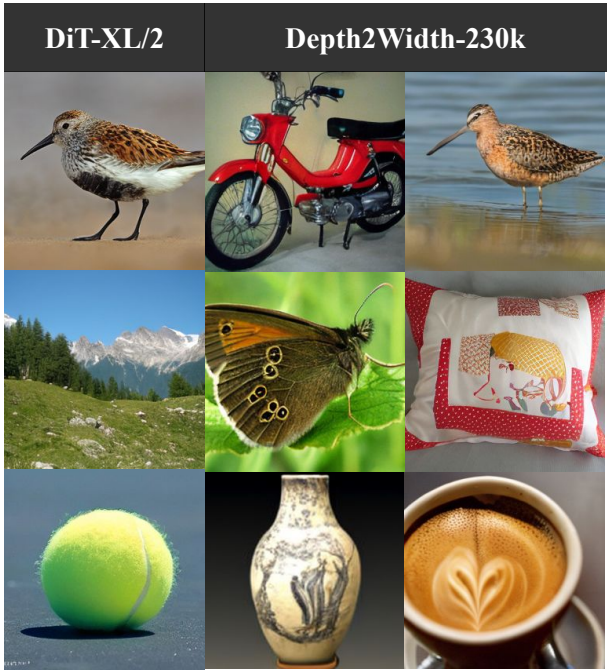


[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Case study: Converting Model Depth to Width via Grafting



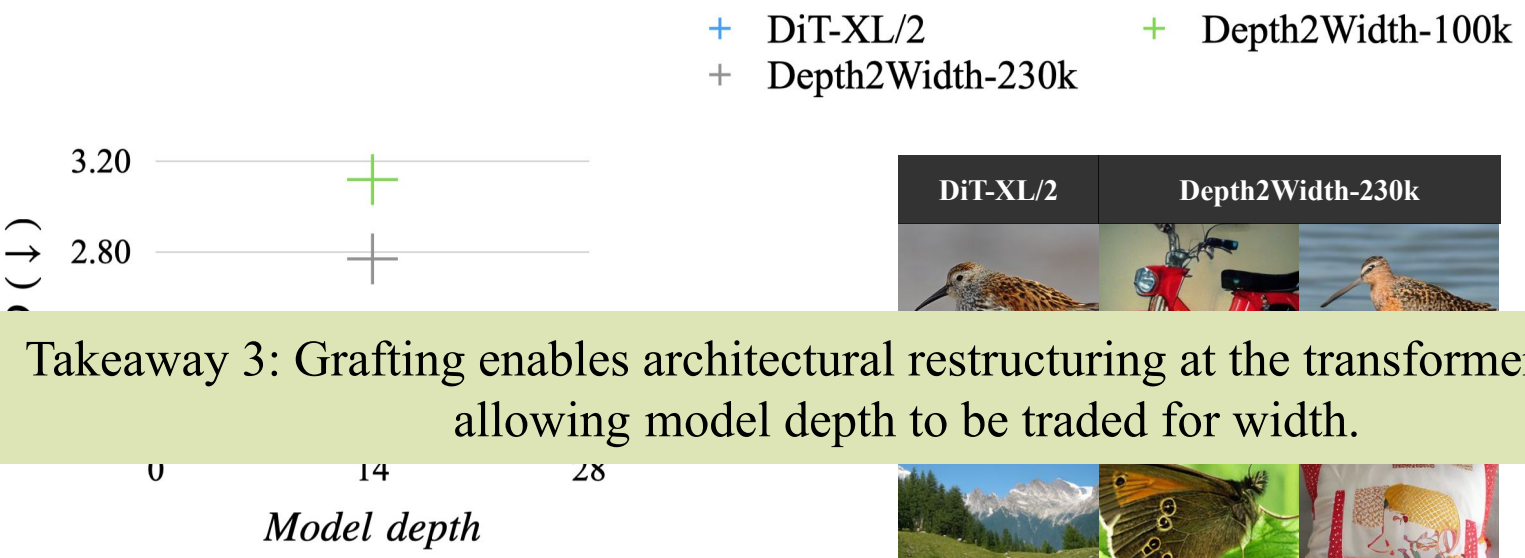




(*) Speedup measured @ Batch size=2 / H100

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Case study: Converting Model Depth to Width via Grafting



Takeaway 3: Grafting enables architectural restructuring at the transformer block level, allowing model depth to be traded for width.

25%

<5%

<48 hours

(*) Speedup measured @ Batch size=2 / H100

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

Grafting - Summary

- Grafting is a simple approach to post-training architecture editing.
- Results on:
  - Hybrid Class-conditioned image generation
  - Efficient Text-to-image Generation
  - Model restructuring

Models/ code/ demo/ samples
grafting.stanford.edu

[K. Chandrasegaran et al .Exploring Diffusion Transformer Designs via Grafting. NeurIPS 2025]

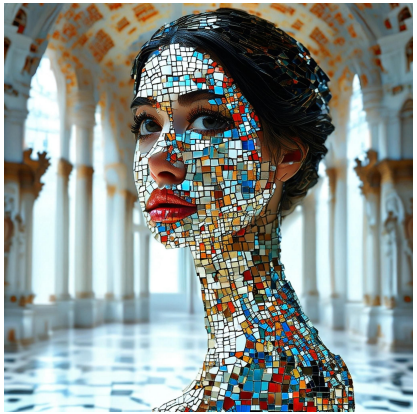
Scaling Transformers

Train new Architecture Designs

Inference for Long/Streaming Video

Dataset & Benchmarking Visual Generation

[Grafting, NeurIPS 25]



Scaling Transformers

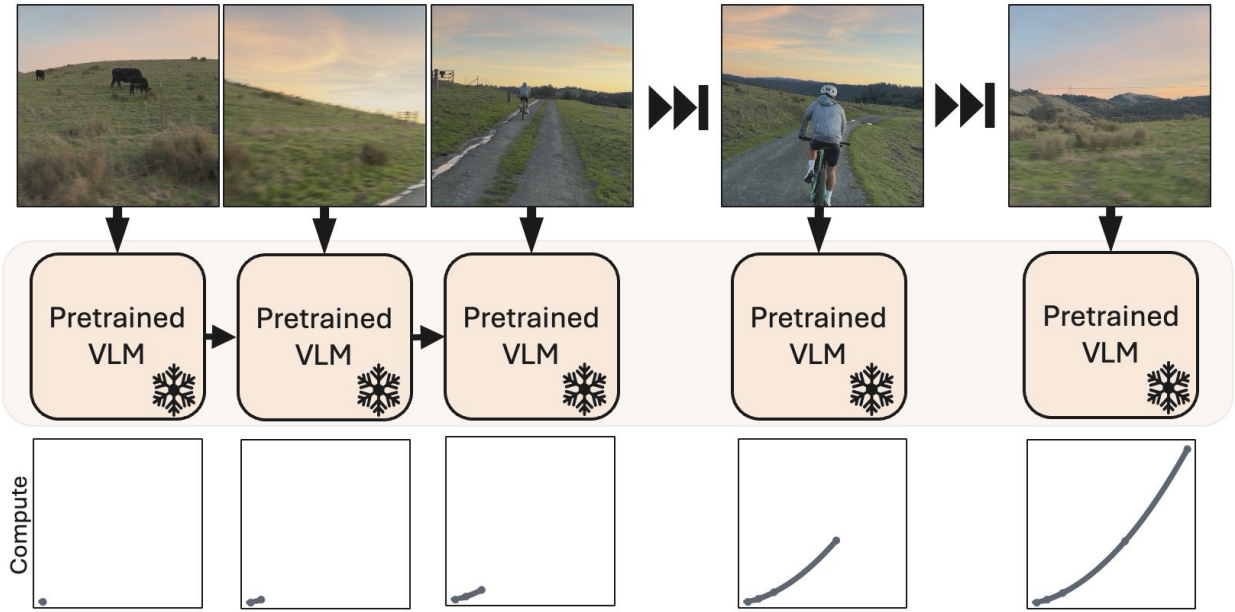
Train new Architecture Designs

Inference for Long/Streaming Video

Dataset & Benchmarking Visual Generation

[Grafting, NeurIPS 25]

VLMs for Video Understanding



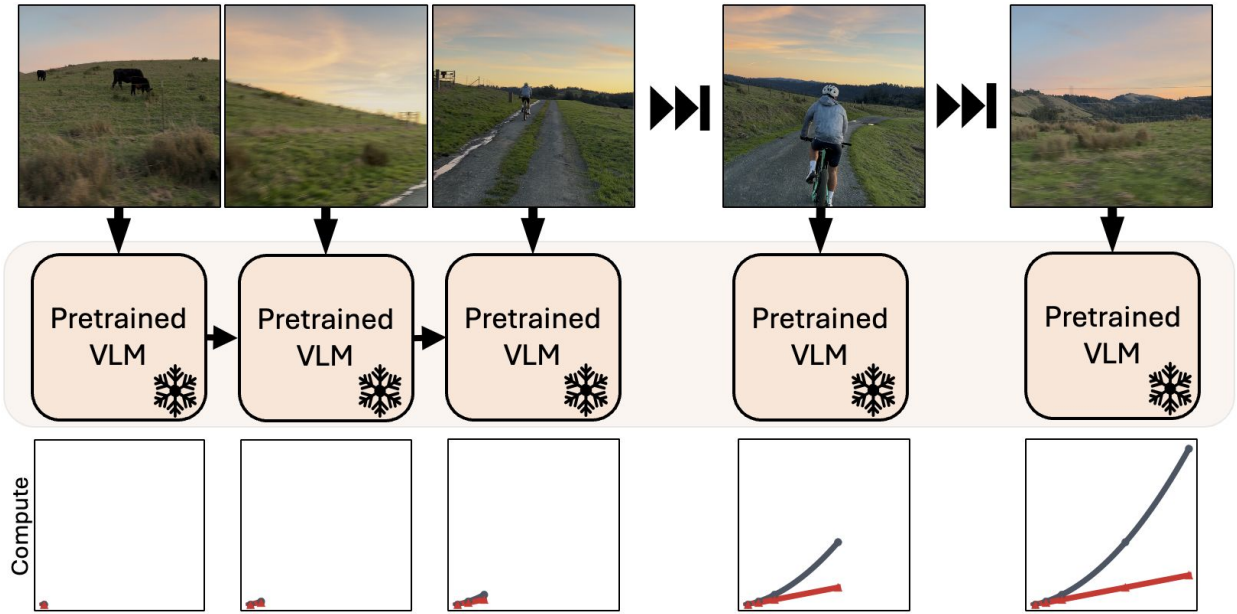
What we have

$O(N^2)$

dense self-attention over video tokens

[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

StateKV: linear scaling of pretrained VLMs



What we have

O(N²)

dense self-attention over video tokens

What we need

O(N)

constant cost per frame · linear total

[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

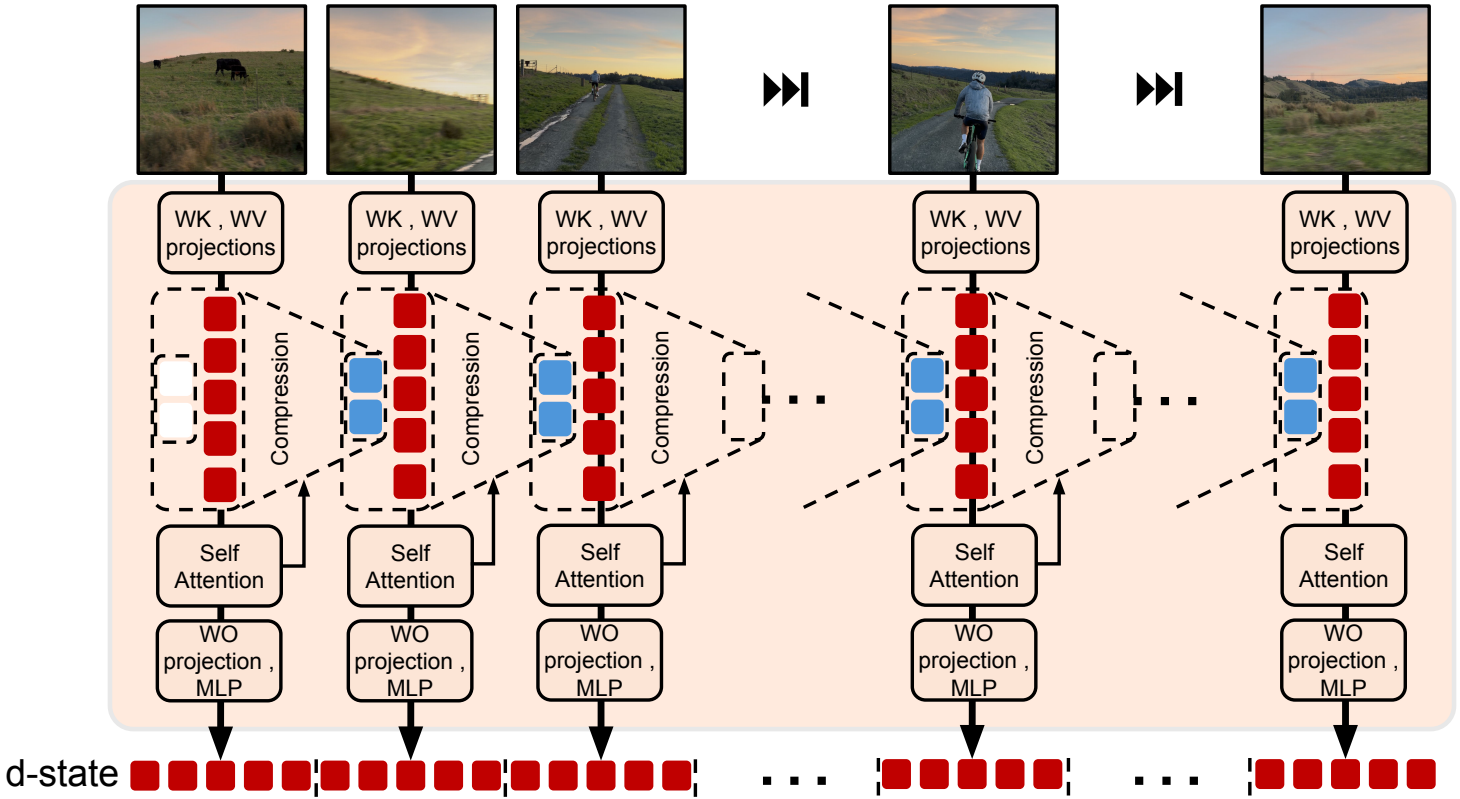
StateKV: linear scaling of pretrained VLMs

Two key assumptions that we verify empirically:

1. **Most of the past does not matter**
   a. most cross-frame attention is concentrated on a surprisingly small subset of tokens
2. **The important tokens change slowly**
   a. instead of recomputing an optimal memory from scratch, we can maintain and update a compact state as the video progresses

[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

StateKV algorithm: compressed memory - `cstate`



[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

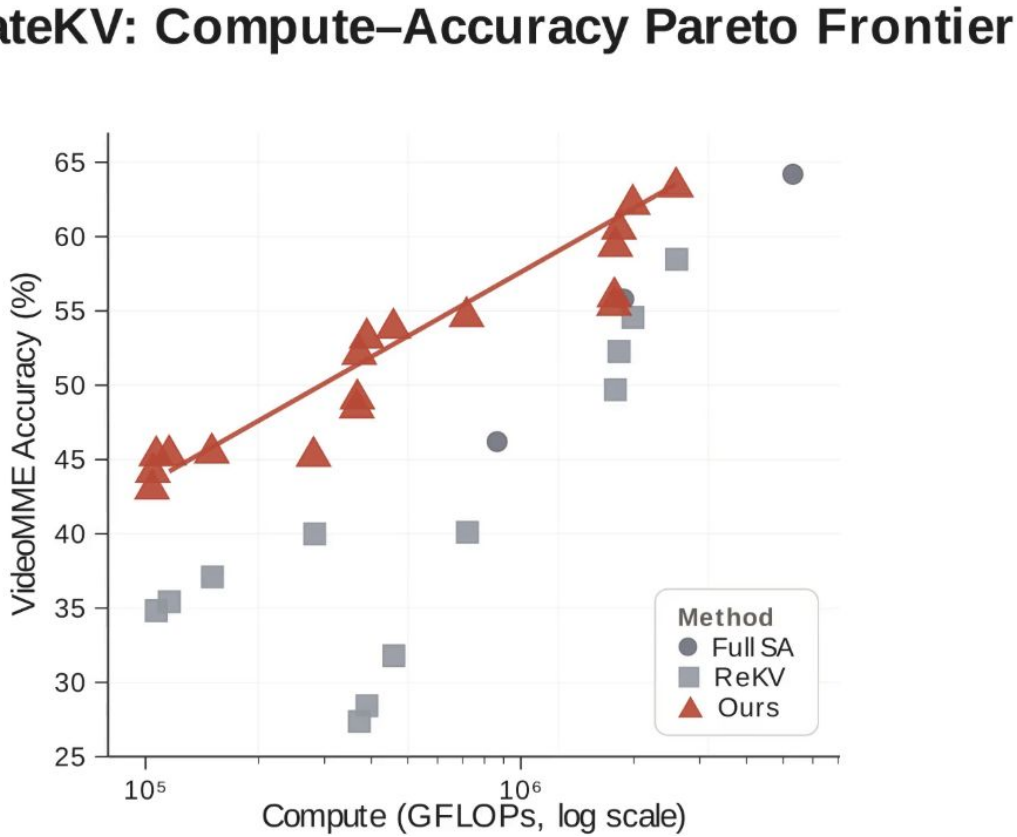
StateKV algorithm: update state recurrently

[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

StateKV algorithm: update state recurrently

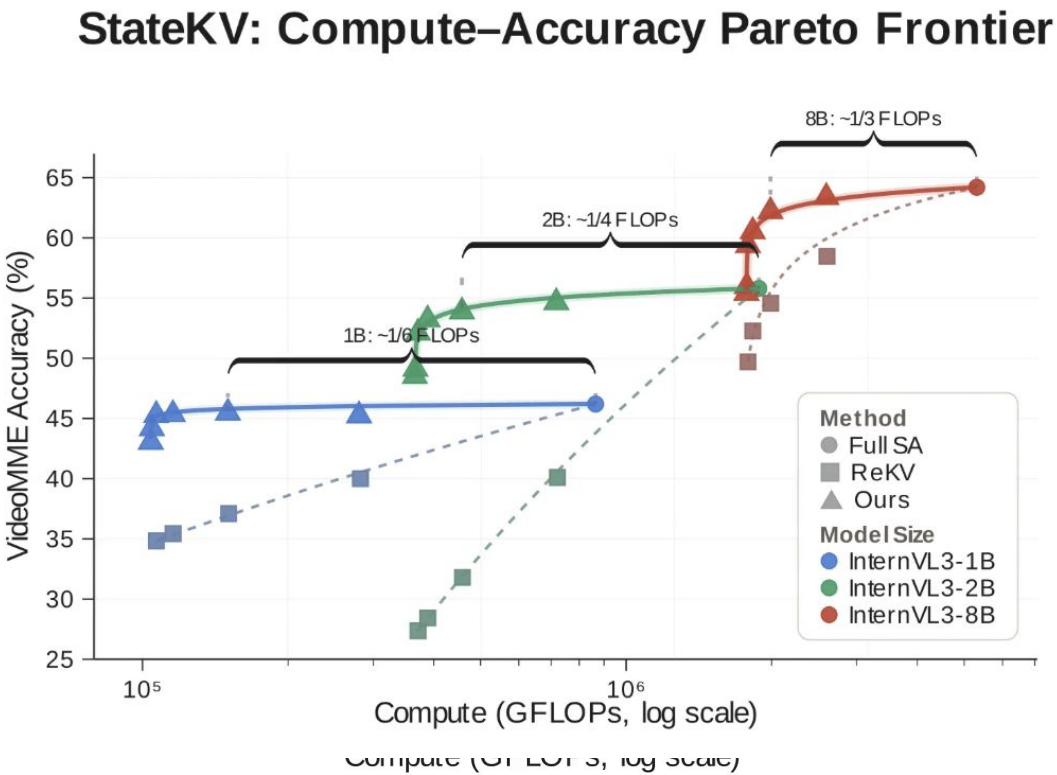
[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

Accuracy–compute tradeoff



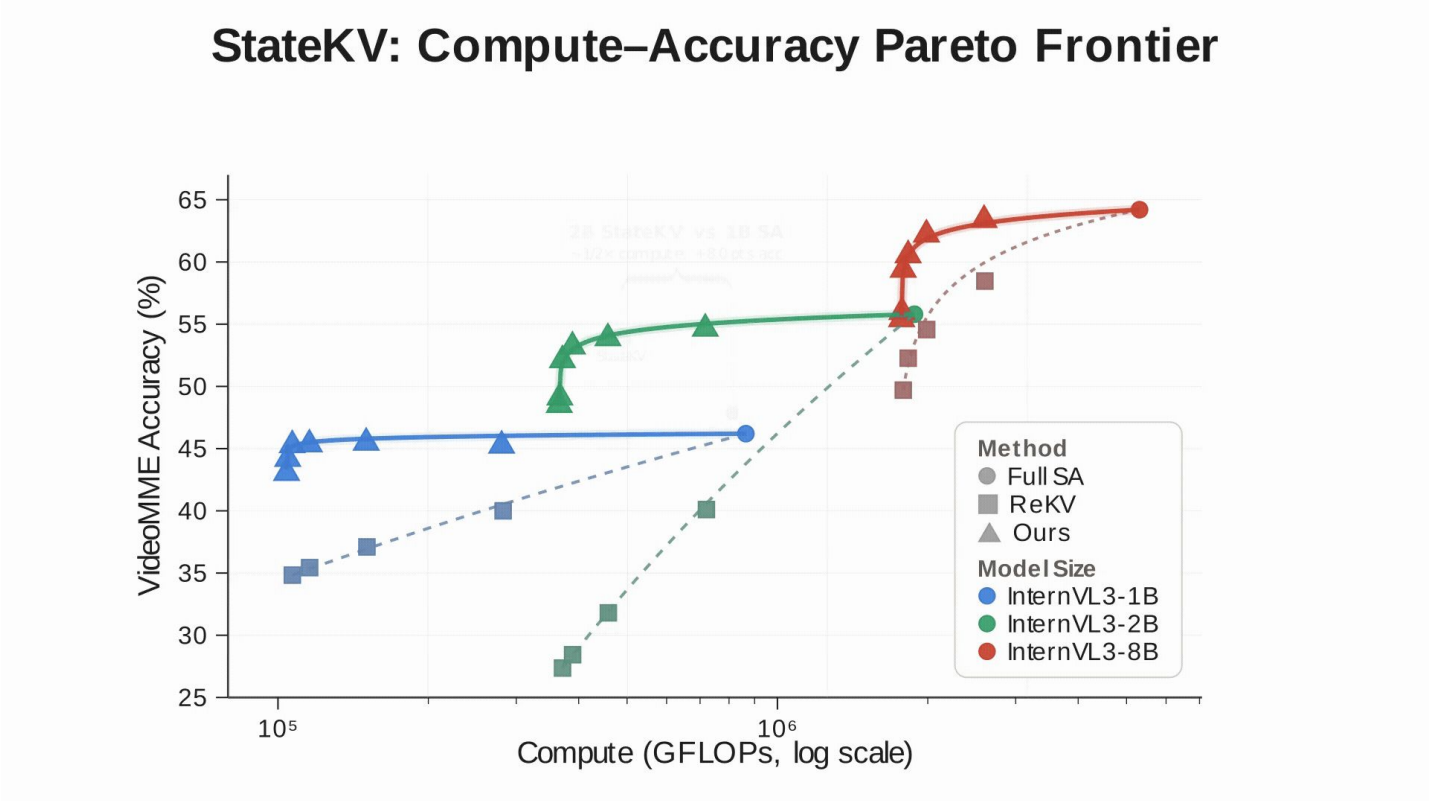
[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

Accuracy–compute tradeoff



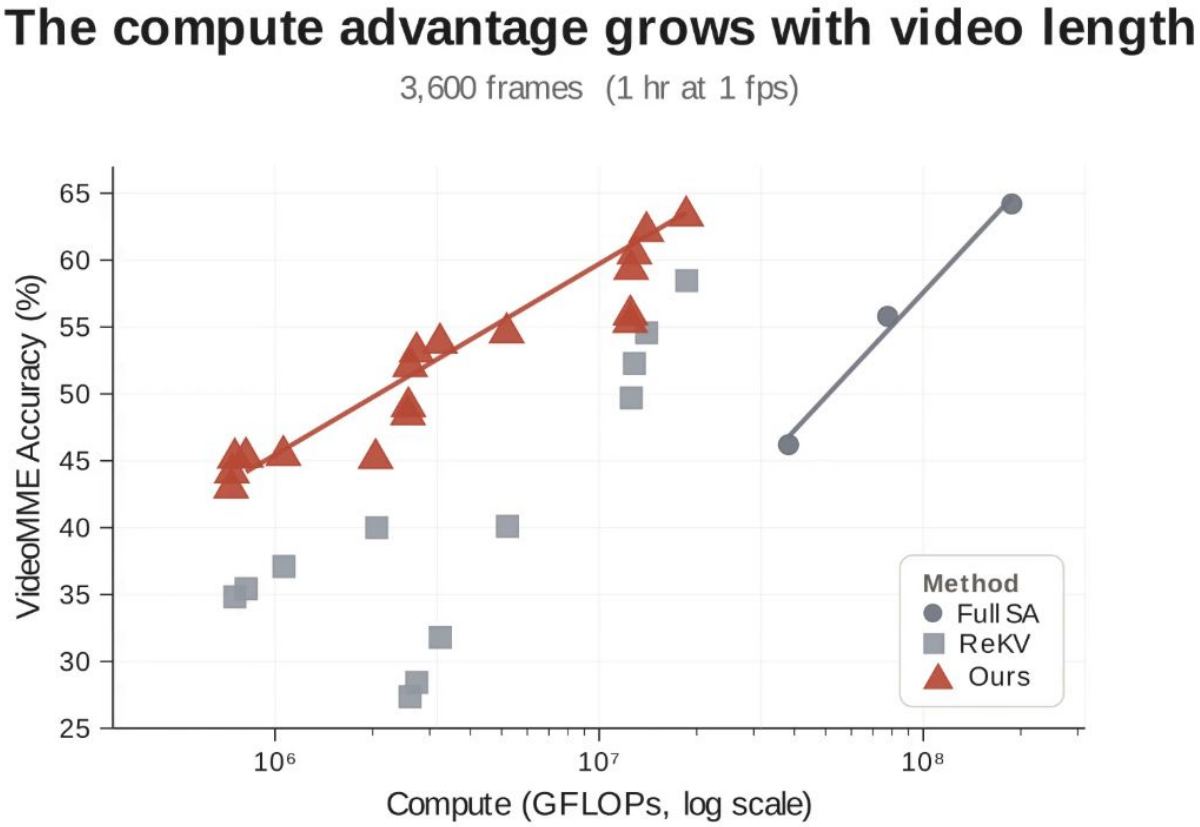
[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

Accuracy–larger models with same compute



[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

The long-video regime



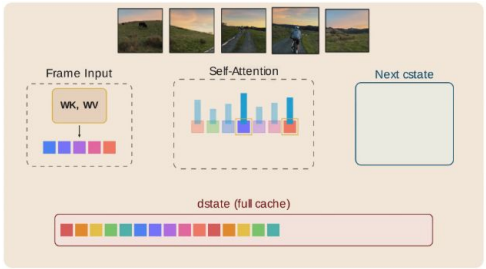
[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

StateKV - Summary

1. StateKV enables long-video VLM **inference to scale linearly**

2. It preserves most of full self-attention **performance**

3. Enables running larger models at a the cost of smaller models



Blog / paper/ code

ceyzaguirre4.github.io/StateKV

[C. Eyzaguirre et al. Linear Scaling Video VLMs for Long Video Understanding. arxiv. 2026.]

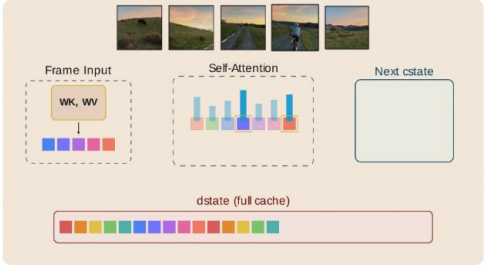
Scaling Transformers

Train new Architecture Designs

[Grafting, NeurIPS 25]

Inference for Long/Streaming Video

[StateKV, arxiv 26]



Dataset & Benchmarking Visual Generation

Scaling Transformers

Train new Architecture Designs

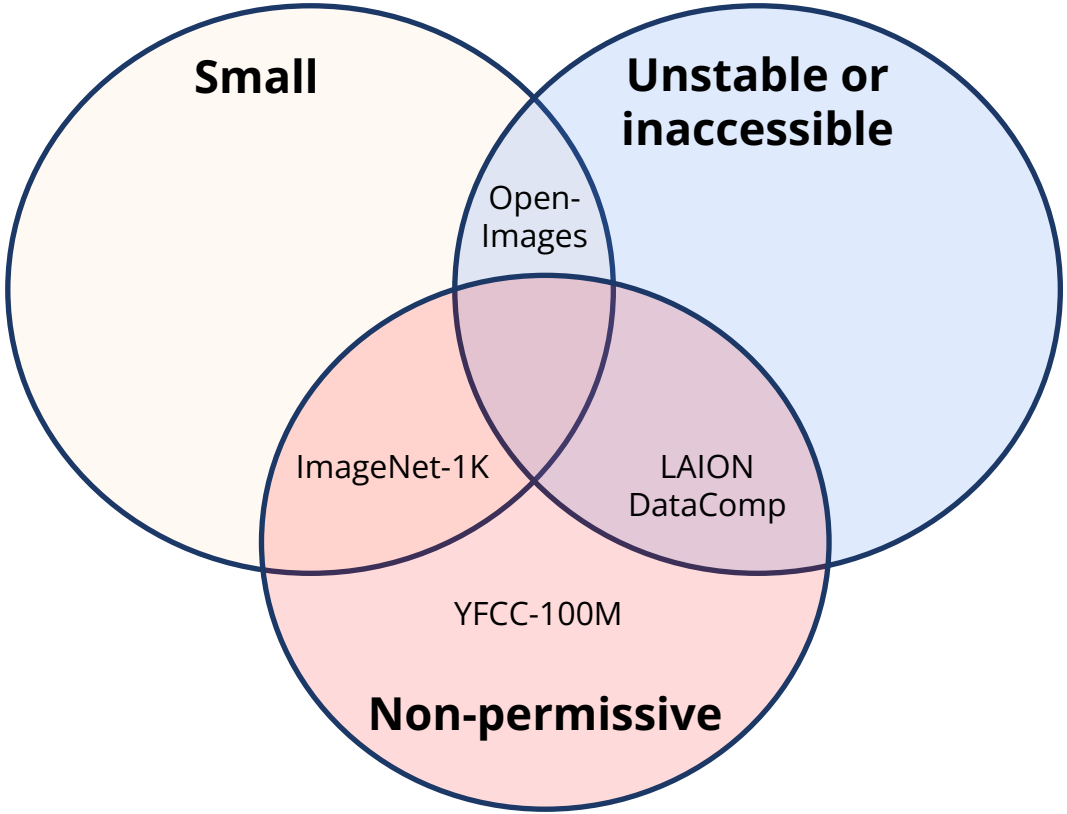
[Grafting, NeurIPS 25]

Inference for Long/Streaming Video

[StateKV, arxiv 26]

Dataset & Benchmarking Visual Generation

Current datasets for visual generation



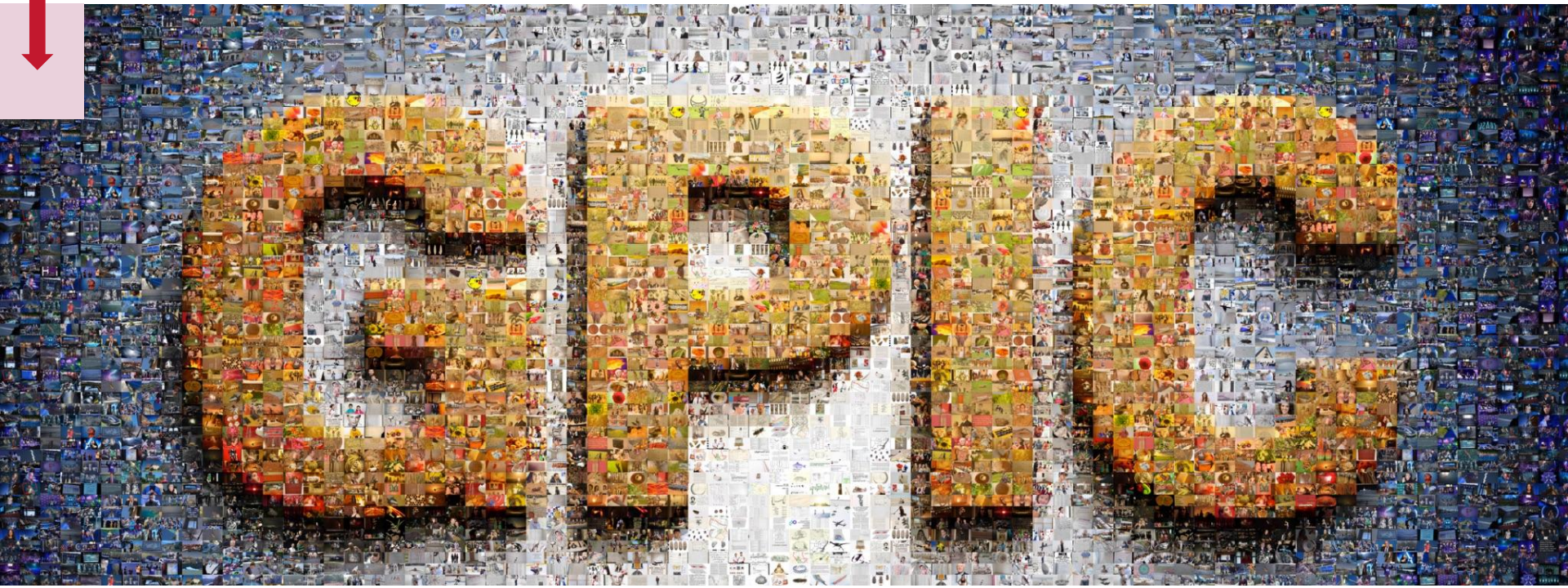
What properties should a new dataset have?

Small	→	100M images
Unstable or inaccessible	→	Host the raw data in sharded tarfiles in a stable repo
Non-permissive	→	All CC, CC-BY, or Public domain licensed.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

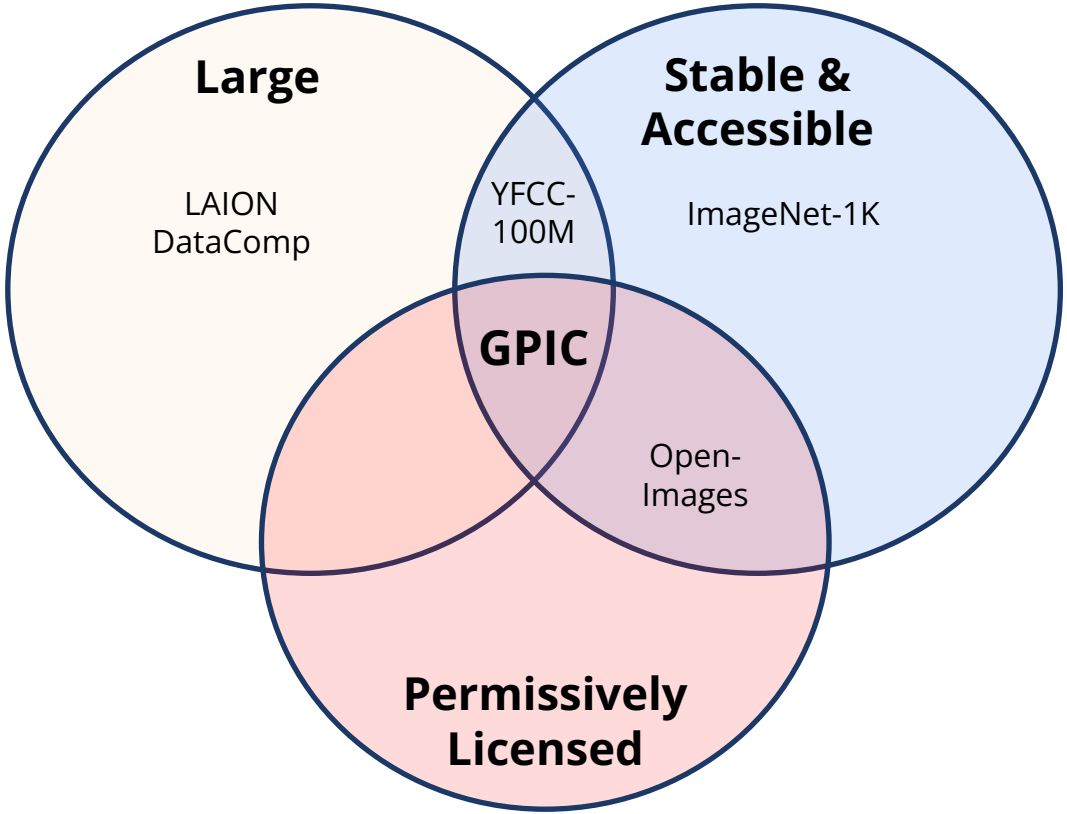
GPIC: A Giant Permissive Image Corpus for Visual Generation

ImageNet-1K



[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Current datasets for visual generation



[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

GPIC: A Giant Permissive Image Corpus for Visual Generation

Dataset Statistics

Image Resolution

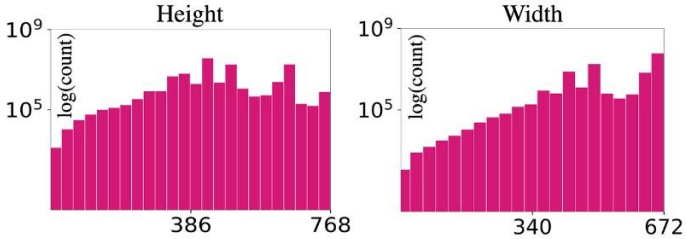
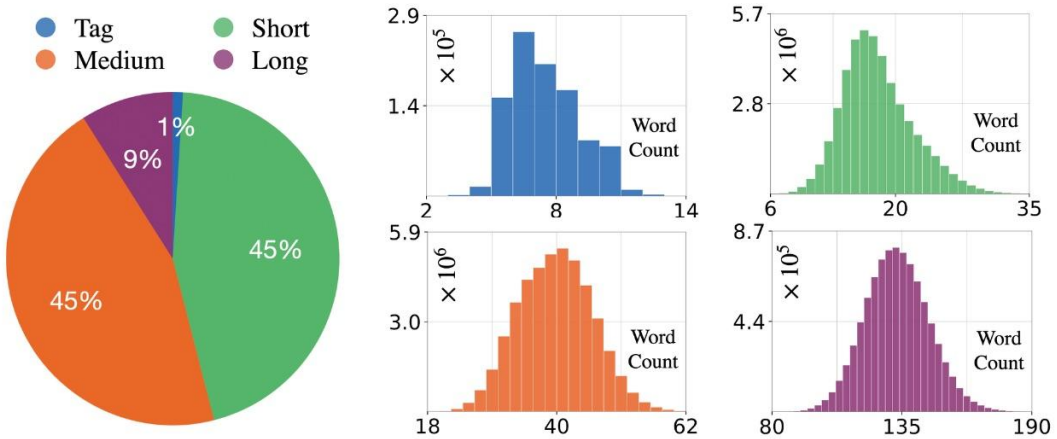


Image License

CC BY	80.92%
Public Domain	13.85%
CC0	4.42%
No restrictions	0.81%

Caption Statistics



GPIC Dataset Release

Centrally Hosted | 12.9TB / 8000 shards | MIT License

GPIC Dataset Splits

Train	Test	Validation
100M Image-Text Pairs	**1M** Image-Text Pairs	**200K** Image-Text Pairs

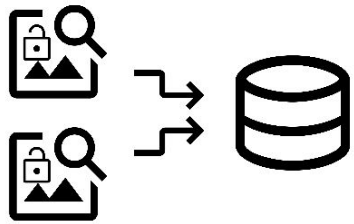
GPIC Benchmark Scales

GPIC-Full	GPIC-Lite	GPIC-Nano
100M Image-Text Pairs	**10M** Image-Text Pairs	**1M** Image-Text Pairs

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

GPIC Construction Pipeline

1



Source Pool/ Licensing: 86.4 million Flickr, 15.6 million Wikimedia. Licenses in {CC, CC-BY, Public Domain} only (commercial use ok, research use ok).

2

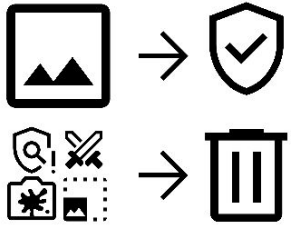
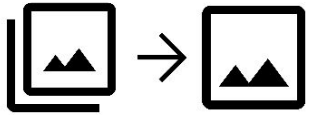


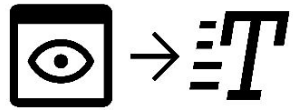
Image Filtering: Basic image quality**,** extreme aspect ratios/sizes and Safety + NSFW.

3



Deduplication: SSCD copy-detection to remove near-duplicates, and verified with SHA-256 hashing

4



Caption Generation: Qwen3-VL-4B-Inst across four formats: tag, short, medium, and long.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Image-Text Examples from GPIC

Medium

A motocross rider in full gear rides an orange dirt bike through muddy terrain. Another rider is visible in the background on a similar trail, with trees and open field surrounding the track.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Image-Text Examples from GPIC

Medium

The back of a red and white FDNY ambulance is parked on a street. It has “KEEP BACK” written in bold letters and emergency lights along the top, with trees and a building visible in the background.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Image-Text Examples from GPIC

Short

A gray fighter jet with "LN" and "301" markings sits on an airfield runway.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Image-Text Examples from GPIC

Short

A bird perches on a bare tree branch against a cloudy sky. Distant mountains are visible in the background.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Image-Text Examples from GPIC

Short

A person blows a large flame into the dark night while others watch from lounge chairs.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Image-Text Examples from GPIC

Long

A two-story house with a beige exterior and a dark green metal roof stands at the center of the scene, featuring white trim around its windows and porch. A wide front porch with white railings extends across the lower level, and a set of stairs leads up to the entrance. To the right of the house, a bright pink pop-up canopy is set up on a grassy lawn, with a few people standing nearby under it. The house is surrounded by lush green trees and manicured bushes, with some flowering plants in red and pink near the walkway. A paved path curves from the left side toward the house's front steps, and a trash can sits near the bushes. In the background, more trees and a faint string of lights are visible, suggesting an outdoor gathering or event.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

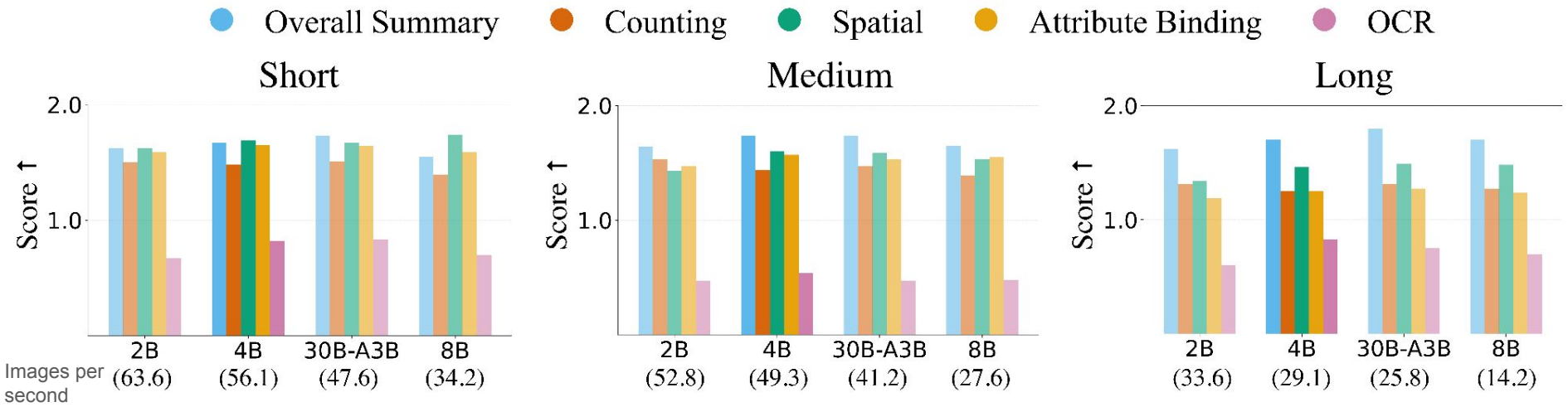
Image-Text Examples from GPIC

Tag

blue car, yellow car, showroom, people, vintage

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Caption Quality



Qwen3-VL at 2B, 4B, 8B, and 30B scales.
Throughput (images per second, 1xH100) is shown in parentheses for each model.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

GPIC Benchmarking Protocol for Image Generation

ImageNet-1K Benchmark

- Uses FID which is saturated.
- Many recent models now obtain lower FID than held-out real images.
- Reference statistics are computed over the ImageNet-1K training set.

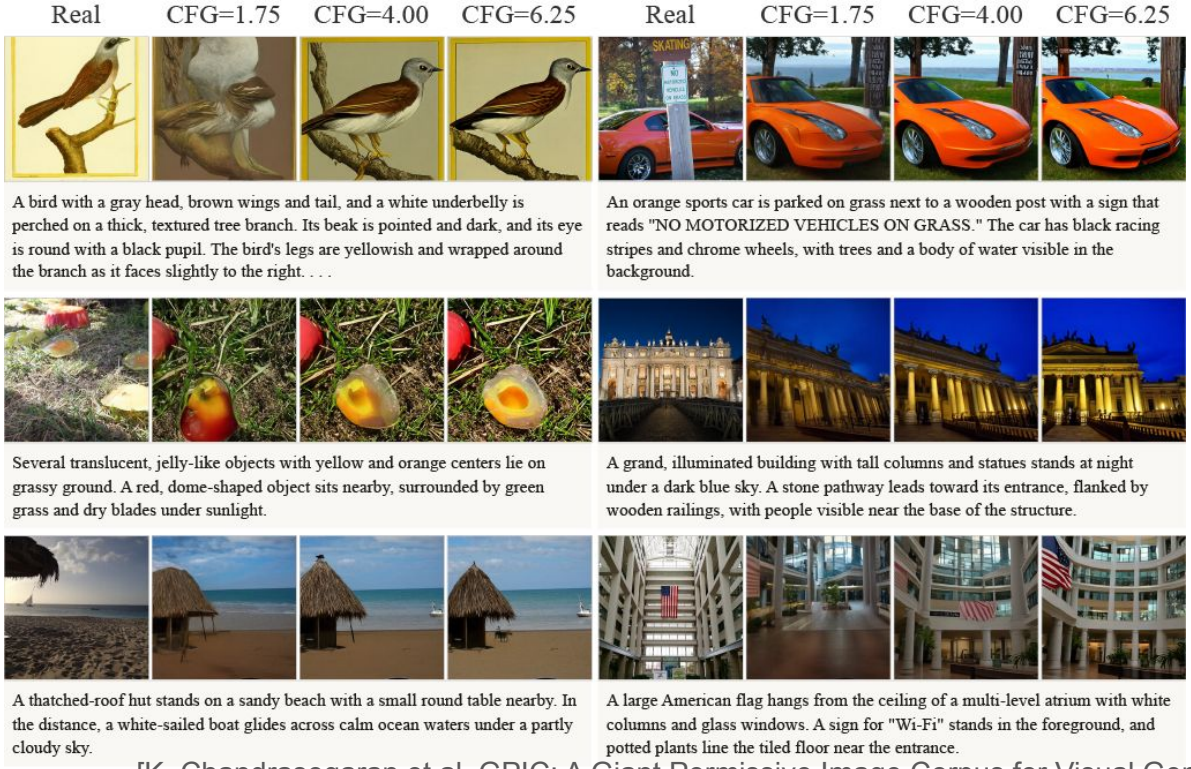
GPIC Benchmark

- Uses FD-DINOv2, which is unsaturated.
- FD-DINOv2 correlates better with human judgments than FID (Stein et. al).
- Reference statistics are computed over the 1M GPIC test set.

Stein et al. "Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models." *NeurIPS 2023*

Can we train a model on GPIC? Yes:

JiT-T2I (1.1B), pixel-space flow matching, trained for one epoch on GPIC-Full.



[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Can we train a model on GPIC? Yes:

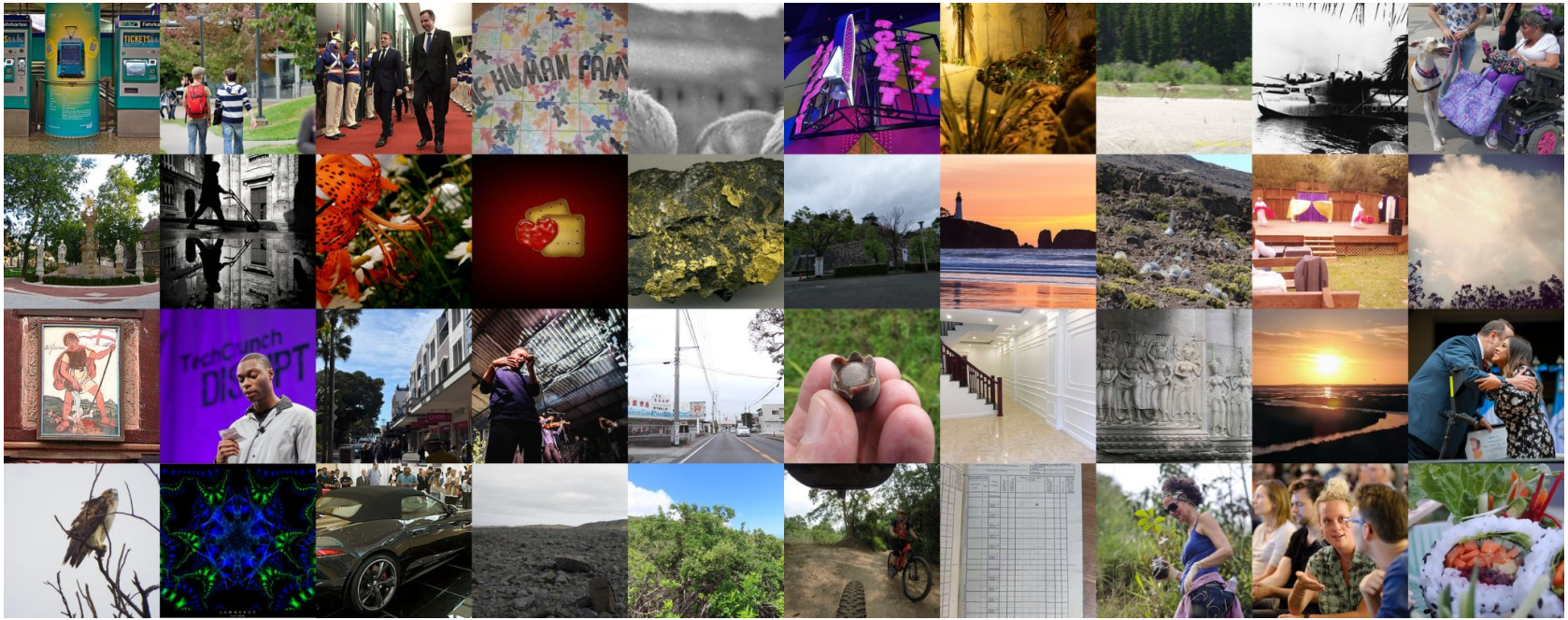
JiT-T2I (1.1B) [PixelGen-XXL/16, Ma et al arxiv 2026]
pixel-space flow matching, trained for one epoch on GPIC-Full.

CFG	FD ↓	Precision ↑	Recall ↑	Density ↑	Coverage ↑
1.75	204.01	0.917	0.530	1.034	0.806
4.00	87.80	0.933	0.765	1.012	0.906
6.25	76.25	0.942	0.792	1.014	0.908

Table 3: **JiT-T2I baseline results on GPIC-Full after training for one epoch.** We report FD, Precision, Recall, Density, and Coverage for three classifier-free guidance scales. We use 50-step Euler sampling for all generations. All metrics are computed against the 1M GPIC test set.

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

GPIC: A universe of 100M permissively licensed images



[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

GPIC: A Giant Permissive Image Corpus for Visual Generation

Dataset, Benchmark, Models

[K. Chandrasegaran et al. GPIC: A Giant Permissive Image Corpus for Visual Generation. arxiv. 2026.]

Scaling Transformers

Train new Architecture Designs

[Grafting, NeurIPS 25]

Inference for Long/Streaming Video

[StateKV, arxiv 26]

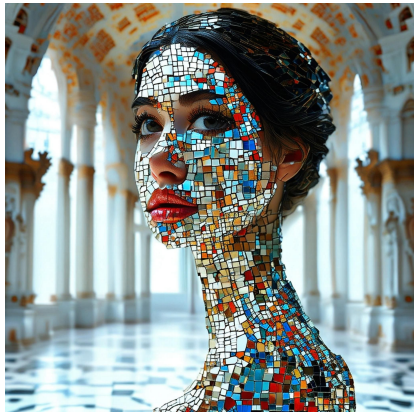
Dataset & Benchmarking Visual Generation

[GPIC, arxiv 26]

Scaling Transformers

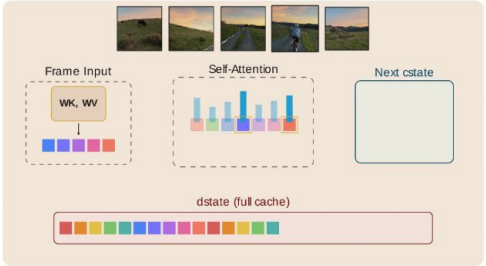
Train new Architecture Designs

[Grafting, NeurIPS 25]



Inference for Long/Streaming Video

[StateKV, arxiv 26]



Dataset & Benchmarking Visual Generation

[GPIC, arxiv 26]

Thank you!

- K. Chandrasegaran, M. Poli, D. Y Fu, D. Kim, L. M. Hadzic, M. Li, A. Gupta, S. Massaroli, A. Mirhoseini, J. C. Niebles, S. Ermon, L. Fei-Fei. “**Exploring Diffusion Transformer Designs via Grafting**”. NeurIPS 2025
- C. Eyzaguirre, J. Wu, and J. C. Niebles. **Linear Scaling Video VLMs for Long Video Understanding**. arxiv. May 2026
- K. Chandrasegaran, K. Sargent, S. Agarwal, M. Jang, M. Poli, J. C. Niebles, J. Johnson, J. Wu, and L. Fei-Fei. **GPIC: A Giant Permissive Image Corpus for Visual Generation**. arxiv. May 2026

🔗 slides/code/data/models/papers