Conditional Generation of Diffusion Models

Conditional generation

User input:

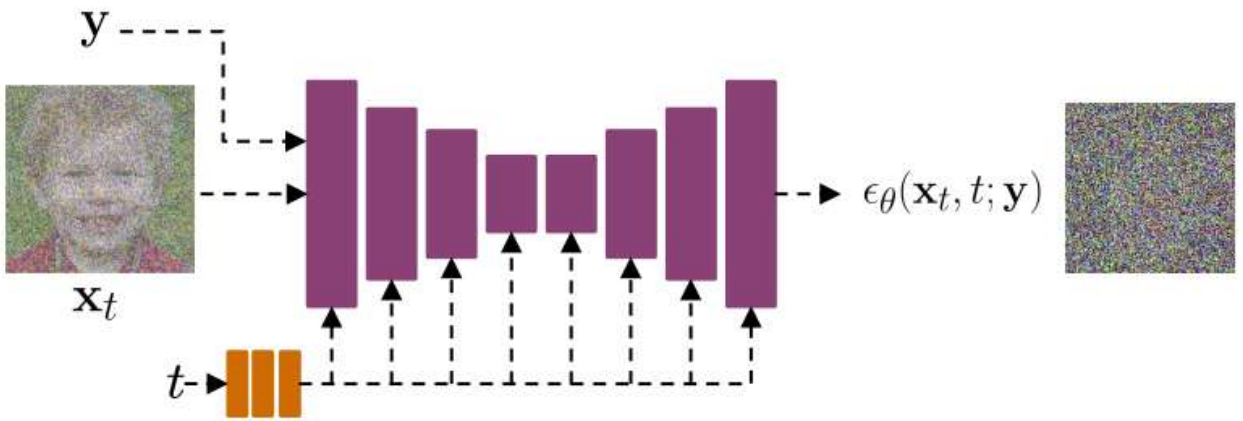
An astronaut riding a horse

Conditional generation

- Let (x,y) denote (image,caption) pairs
- Training a conditional generative model involves learning p(x | y)

- Train score model for the image x conditional on caption y

$$\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim p_{\text{data}}(\mathbf{x},\mathbf{y})}\mathbb{E}_{\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\mathbb{E}_{t\sim\mathcal{U}[0,T]}\ ||\epsilon_\theta(\mathbf{x}_t, t; \mathbf{y}) - \epsilon||_2^2$$



Control the generation process

$p(\mathbf{x})$

Forward model $p(\mathbf{y} \mid \mathbf{x})$

Control signal $\mathbf{y}$ = "dog"

$p(\mathbf{x} \mid \mathbf{y})$

Inverse distribution

Bayes' rule:

$$p(\mathbf{x} \mid \mathbf{y}) = \frac{p(\mathbf{x})\,p(\mathbf{y} \mid \mathbf{x})}{p(\mathbf{y})}$$

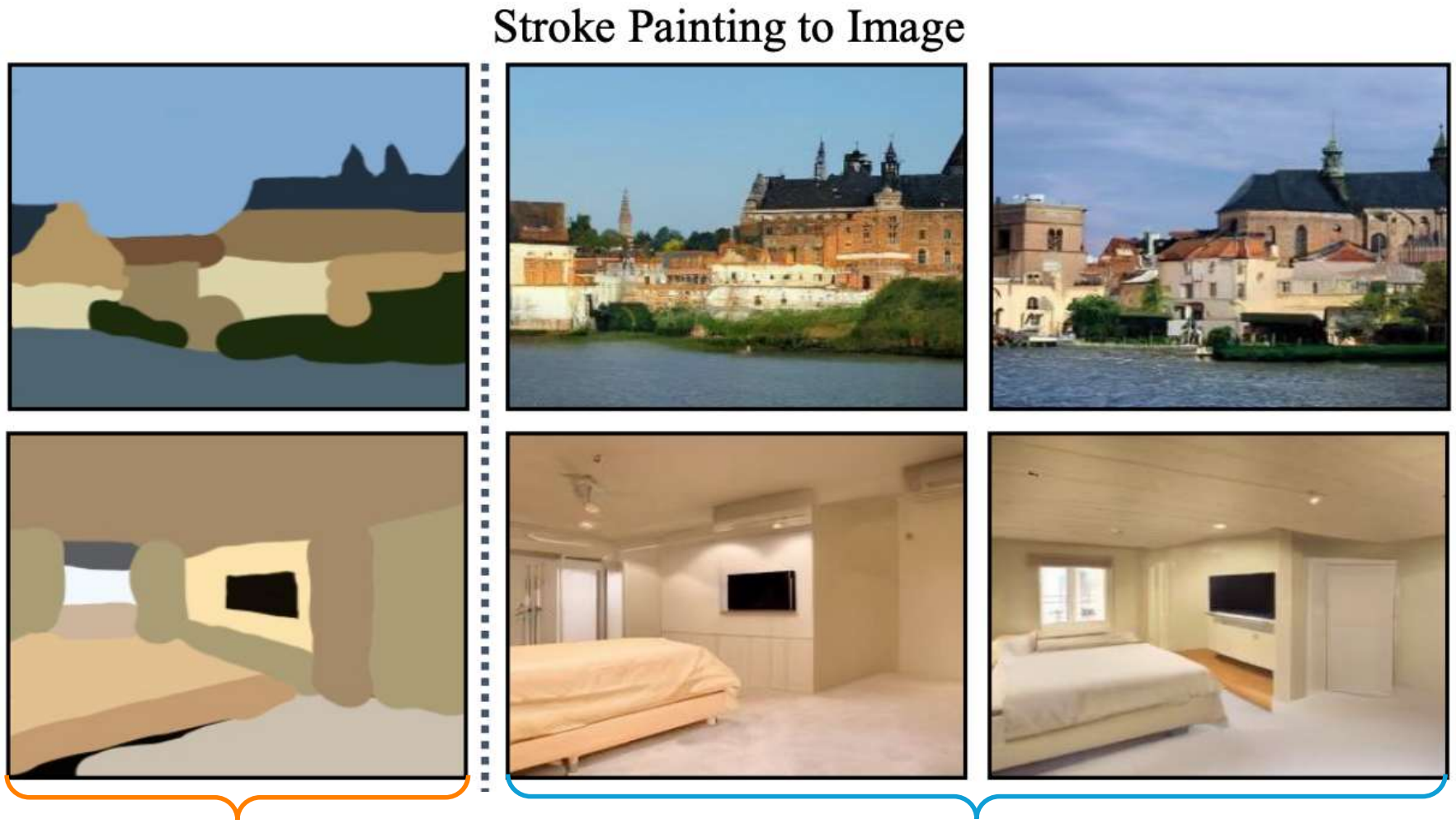
Bayes' rule for score functions:

$$\nabla_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y}) = \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \nabla_{\mathbf{x}} \log p(\mathbf{y} \mid \mathbf{x}) - \nabla_{\mathbf{x}} \log p(\mathbf{y})$$

$$= \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \nabla_{\mathbf{x}} \log p(\mathbf{y} \mid \mathbf{x})$$

Plug in different forward models for the same score model

Stroke to image synthesis



Forward model
$p(\mathbf{y} \mid \mathbf{x})$
can be specified.

Stroke paintings
$\mathbf{y}$

Sampled images
$\mathbf{x} \mid \mathbf{y}$

[Meng, He, Song, Song, Wu, Zhu, Ermon. ICLR 2022]

Language-guided image generation

$\mathbf{y}$ $\mathbf{x} \mid \mathbf{y}$

(Prompt)
Treehouse in the style of Studio Ghibli animation




Forward model

$p(\mathbf{y} \mid \mathbf{x})$

is an image captioning neural network.

[Work by @danielrussruss]

Classifier Guidance

Bayes' rule:

$$p(\mathbf{x} \mid \mathbf{y}) = \frac{p(\mathbf{x})\,p(\mathbf{y} \mid \mathbf{x})}{p(\mathbf{y})}$$

Bayes' rule for score functions:

$$\nabla_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y}) = \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \nabla_{\mathbf{x}} \log p(\mathbf{y} \mid \mathbf{x}) - \nabla_{\mathbf{x}} \log p(\mathbf{y})$$

$\nabla_{\mathbf{x}} \log p(\mathbf{y})$ → **0**

$$= \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \nabla_{\mathbf{x}} \log p(\mathbf{y} \mid \mathbf{x})$$

$\nabla_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y})$: Conditional score

$\nabla_{\mathbf{x}} \log p(\mathbf{x})$: Unconditional score

$\nabla_{\mathbf{x}} \log p(\mathbf{y} \mid \mathbf{x})$: Classifier obtained as the difference

Classifier-Guided Diffusion

To explicit incorporate class information into the diffusion process, trained a classifier $f_\phi(y|\mathbf{x}_t, t)$ on noisy image the use gradients to guide diffusion sampling process toward the condition

$$\begin{aligned}
\nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t, y) &= \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t) + \nabla_{\mathbf{x}_t} \log q(y|\mathbf{x}_t) \\
&\approx -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) + \nabla_{\mathbf{x}_t} \log f_\phi(y|\mathbf{x}_t) \\
&= -\frac{1}{\sqrt{1-\bar{\alpha}_t}} (\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) - \sqrt{1-\bar{\alpha}_t} \nabla_{\mathbf{x}_t} \log f_\phi(y|\mathbf{x}_t))
\end{aligned}$$

A new classifier-guided predictor $\bar{\epsilon}_\theta$ would take the form as following,

$$\bar{\boldsymbol{\epsilon}}_\theta(\mathbf{x}_t, t) = \boldsymbol{\epsilon}_\theta(x_t, t) - \sqrt{1-\bar{\alpha}_t} \nabla_{\mathbf{x}_t} \log f_\phi(y|\mathbf{x}_t)$$

To control the strength of the classifier guidance, we can add a weight w to the delta part,

$$\bar{\boldsymbol{\epsilon}}_\theta(\mathbf{x}_t, t) = \boldsymbol{\epsilon}_\theta(x_t, t) - \sqrt{1-\bar{\alpha}_t}\, w \nabla_{\mathbf{x}_t} \log f_\phi(y|\mathbf{x}_t)$$

Classifier-Guided Diffusion

Algorithm 1 Classifier guided diffusion sampling, given a diffusion model $(\mu_\theta(x_t), \Sigma_\theta(x_t))$, classifier $f_\phi(y|x_t)$, and gradient scale s.

Input: class label y, gradient scale s
$x_T \leftarrow$ sample from $\mathcal{N}(0, \mathbf{I})$
for all t from T to 1 **do**
  $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$
  $x_{t-1} \leftarrow$ sample from $\mathcal{N}(\mu + s\Sigma \nabla_{x_t} \log f_\phi(y|x_t), \Sigma)$
end for
return x_0

Algorithm 2 Classifier guided DDIM sampling, given a diffusion model $\epsilon_\theta(x_t)$, classifier $f_\phi(y|x_t)$, and gradient scale s.

Input: class label y, gradient scale s
$x_T \leftarrow$ sample from $\mathcal{N}(0, \mathbf{I})$
for all t from T to 1 **do**
  $\hat{\epsilon} \leftarrow \epsilon_\theta(x_t) - \sqrt{1-\bar{\alpha}_t}\, \nabla_{x_t} \log f_\phi(y|x_t)$
  $x_{t-1} \leftarrow \sqrt{\bar{\alpha}_{t-1}} \left(\frac{x_t - \sqrt{1-\bar{\alpha}_t}\hat{\epsilon}}{\sqrt{\bar{\alpha}_t}} \right) + \sqrt{1-\bar{\alpha}_{t-1}}\hat{\epsilon}$
end for
return x_0

https://arxiv.org/pdf/2105.05233

Classifier-Guided Diffusion

https://arxiv.org/pdf/2105.05233

Classifier-Free Guidance

- Train both a conditional and an unconditional score model (by randomly dropping the caption during training)

- Combine the two models as follows

$$(1+w)\nabla_{\mathbf{x}_t}\log p(\mathbf{y}|\mathbf{x}_t)+\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t)=(1+w)\big(\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t|\mathbf{y})-\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t)\big)+\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t)$$

$$=(1+w)\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t|\mathbf{y})-w\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t)$$

- w is the classifier-guidance strength

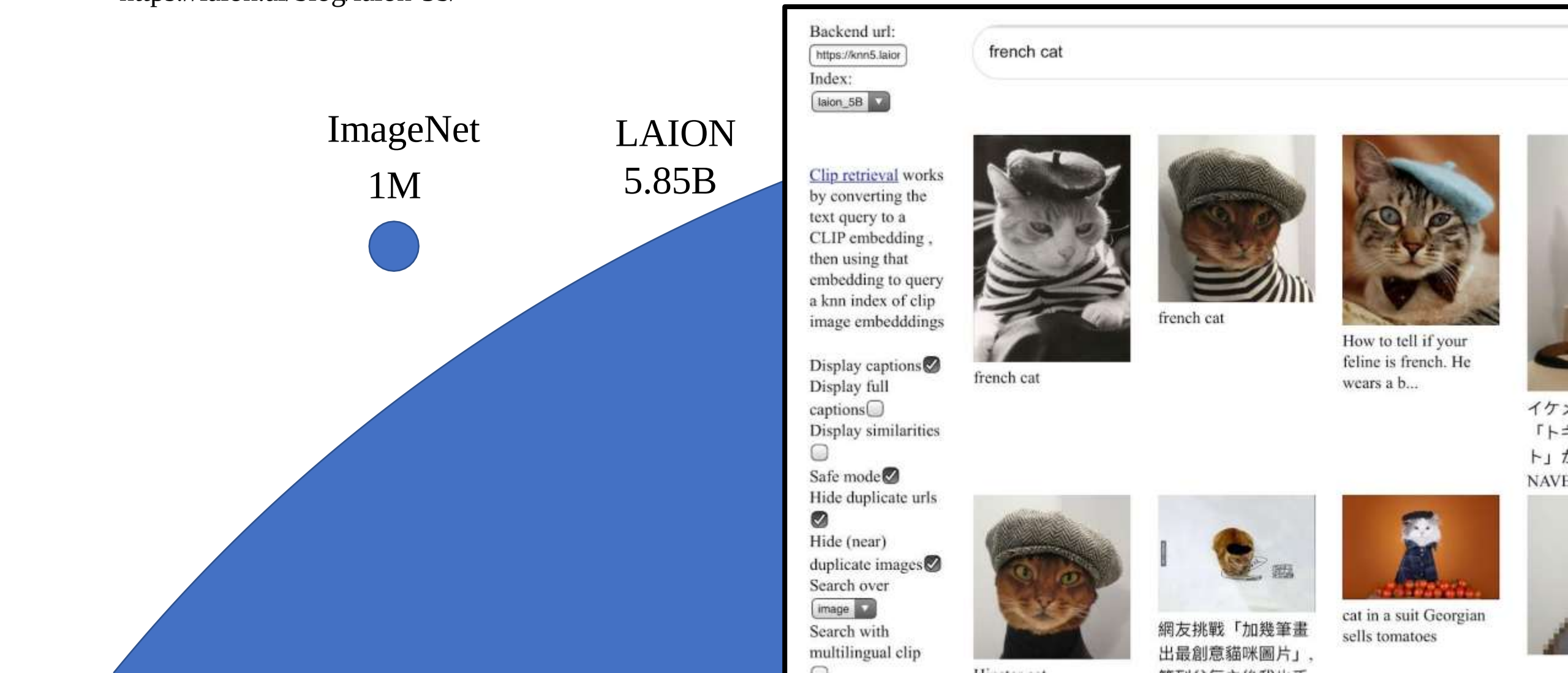
Effect of classifier guidance

Increased classifier guidance strength (w)

Text-to-Image

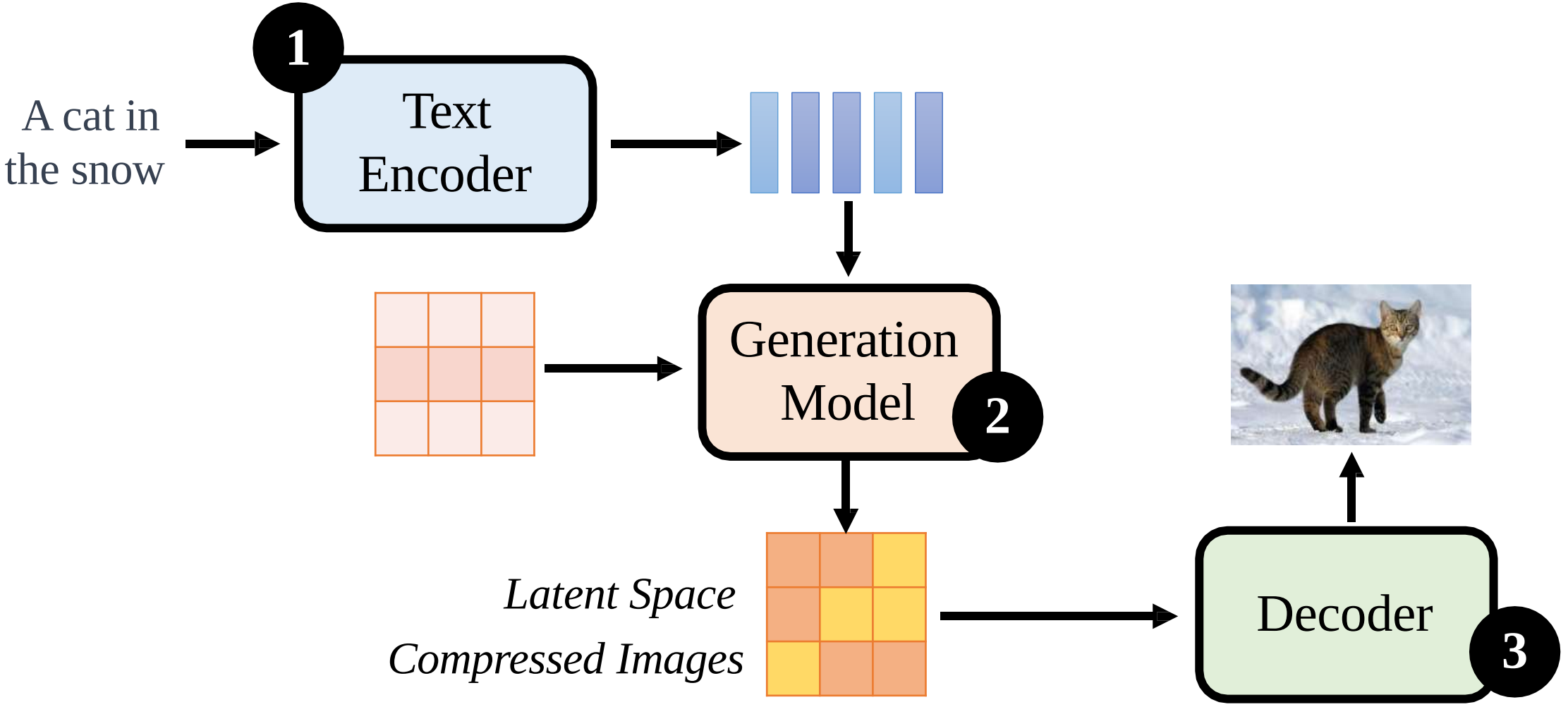
https://laion.ai/blog/laion-5b/

A cat in the snow → Text-to-image Generator → 



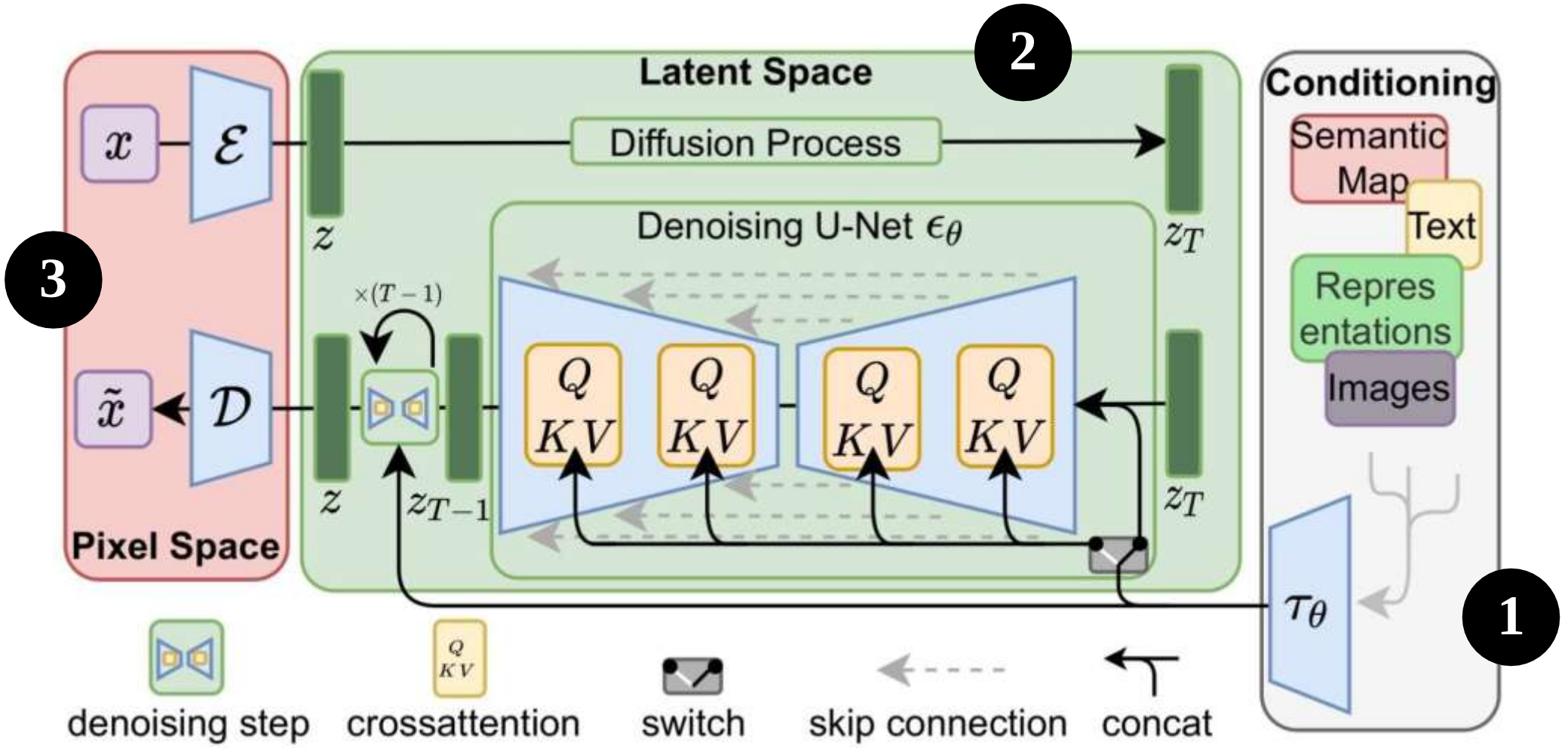
Framework

A cat in the snow → Text-to-image Generator → 



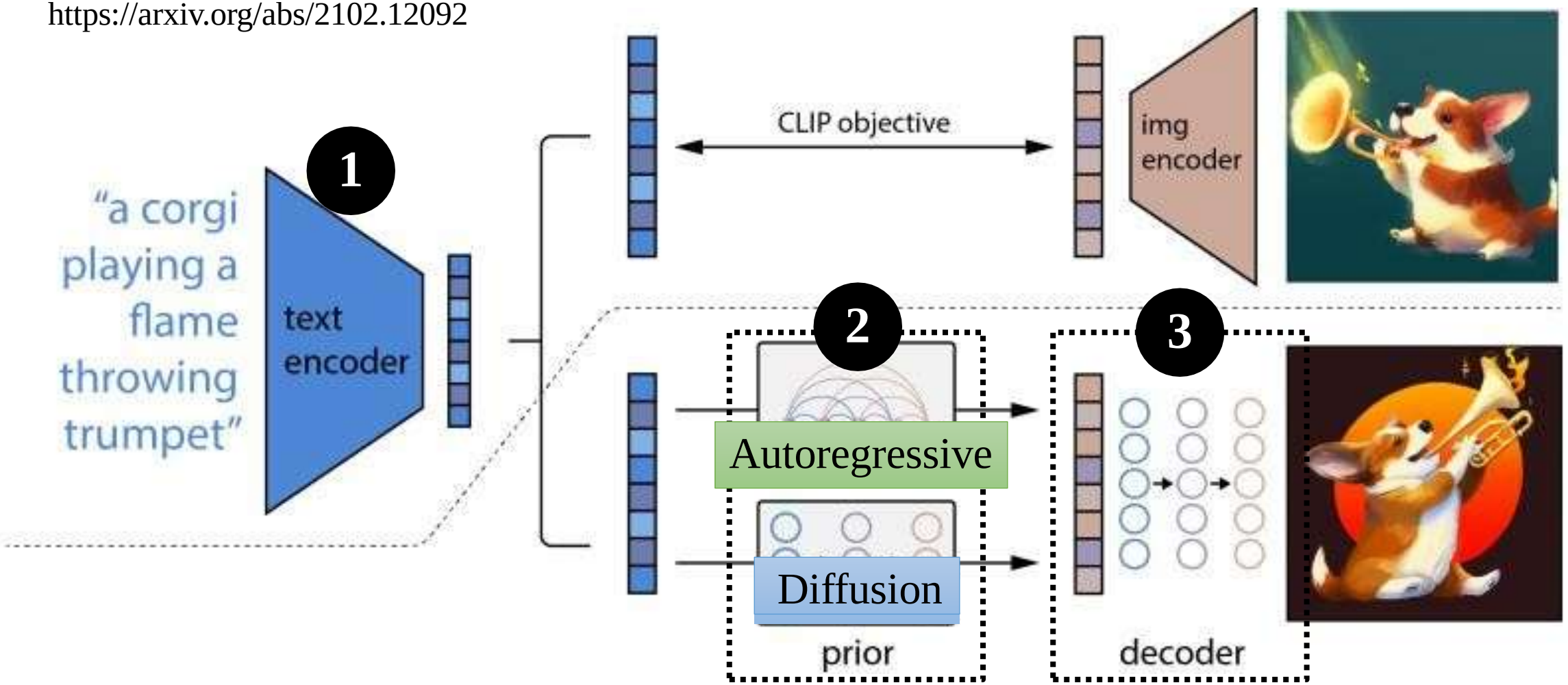
Stable Diffusion

https://arxiv.org/abs/2112.10752



DALL-E series

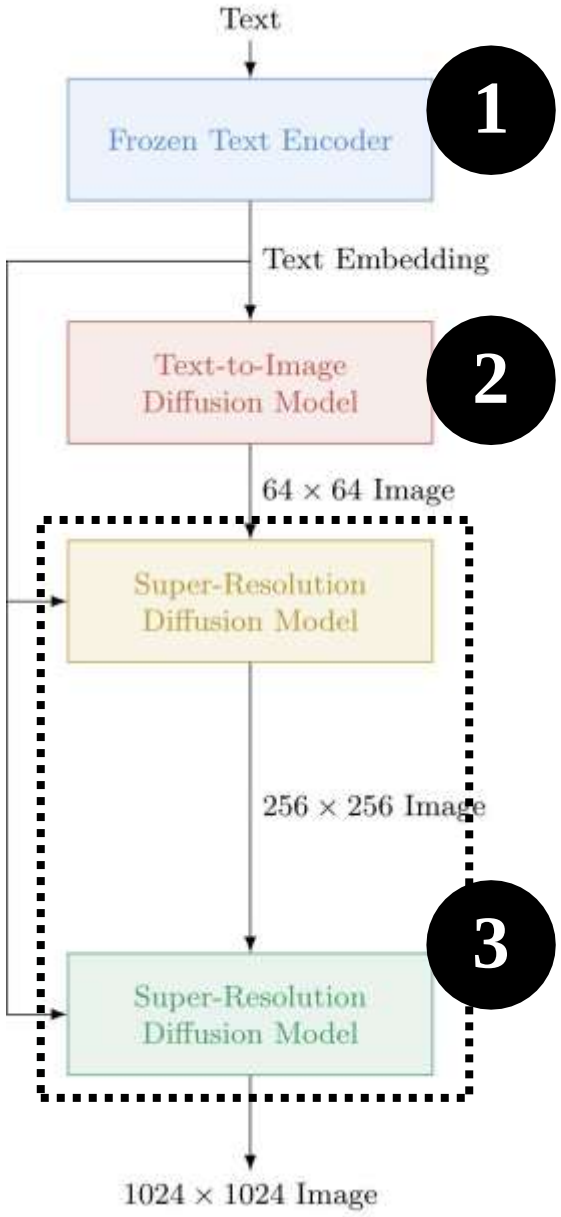
https://arxiv.org/abs/2204.06125
https://arxiv.org/abs/2102.12092

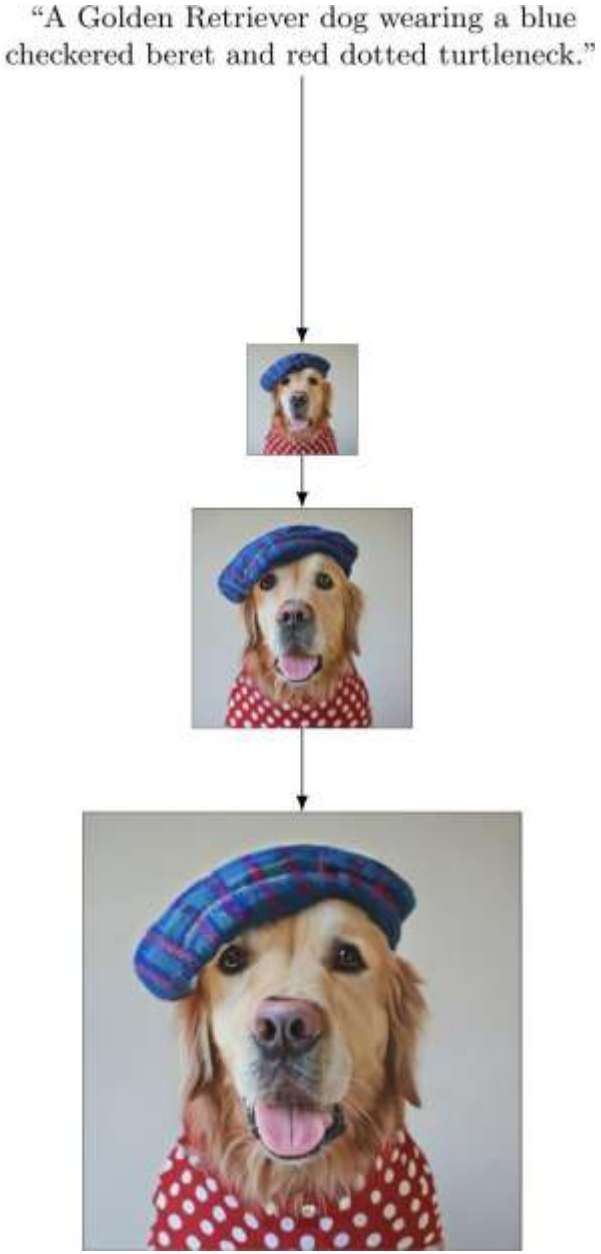


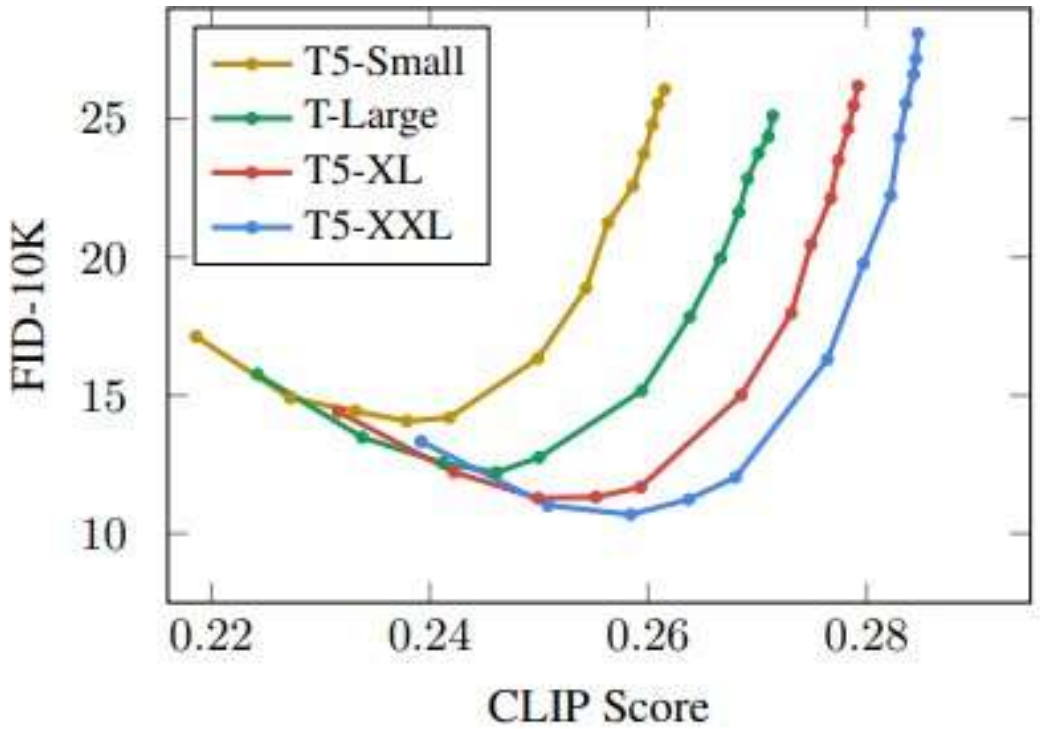
Imagen

https://imagen.research.google/

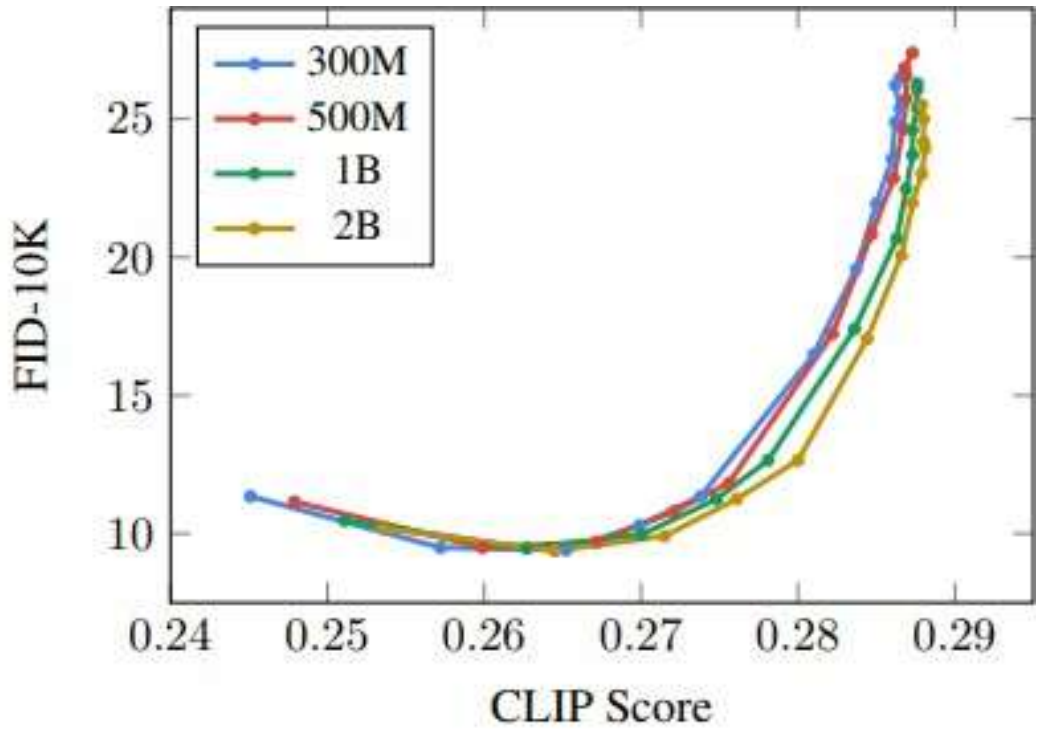
https://arxiv.org/abs/2205.11487







(a) Impact of encoder size.



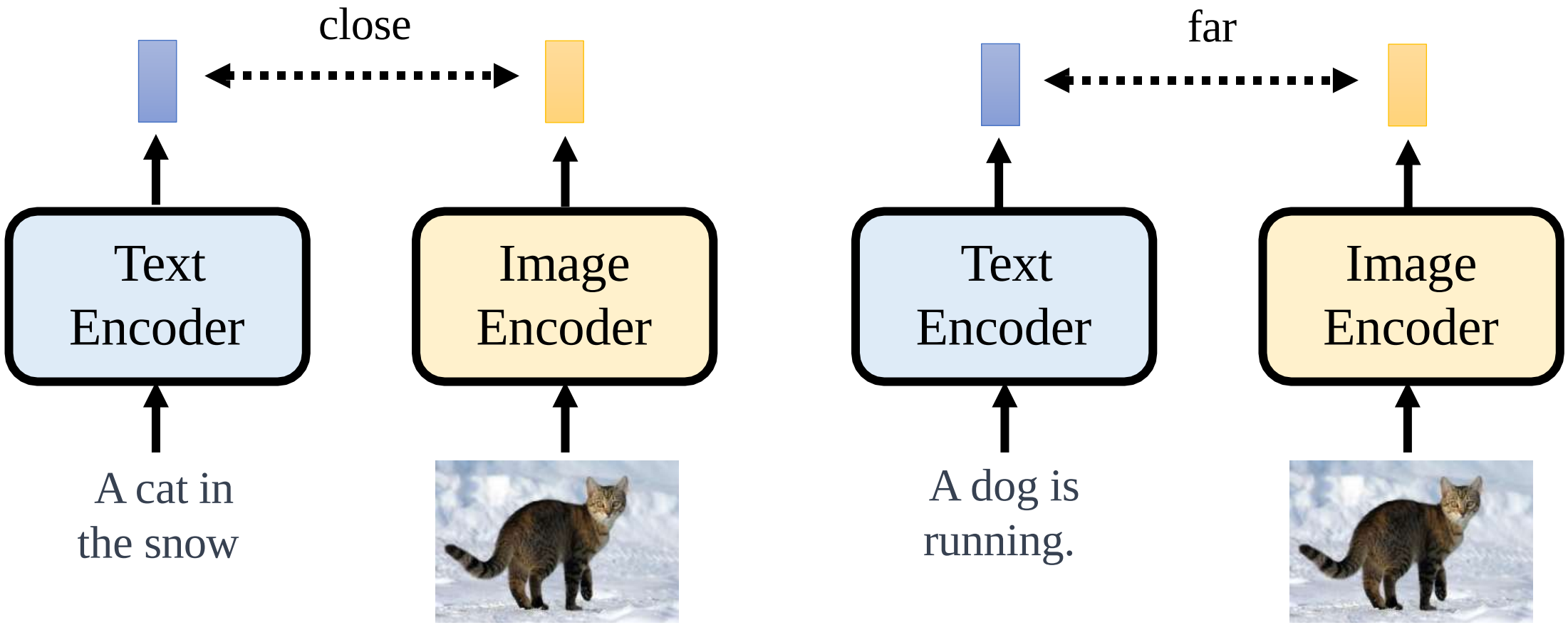
(b) Impact of U-Net size.

https://arxiv.org/abs/2205.11487

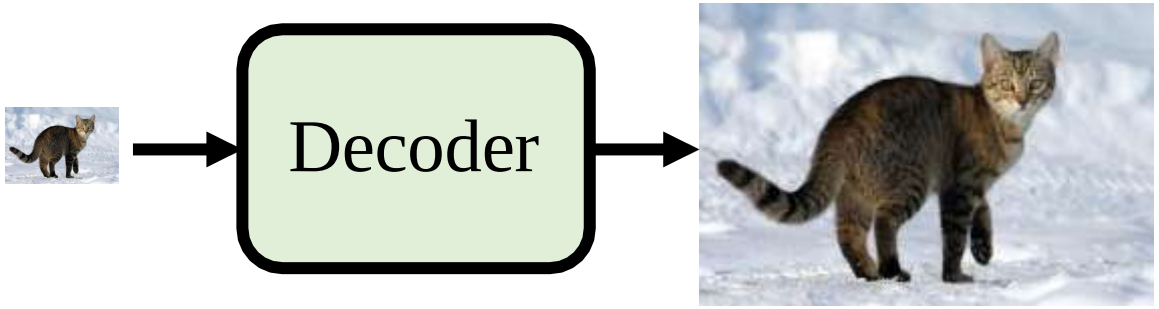
Contrastive Language-Image Pre-Training (CLIP)

https://arxiv.org/abs/2103.00020

400 million image-text pairs



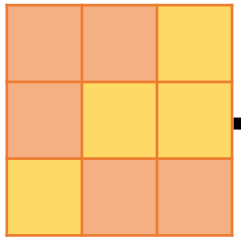
Progressive/Multi-Scale



(Images are generated by Midjourney)

Latent Diffusion

Latent Representation



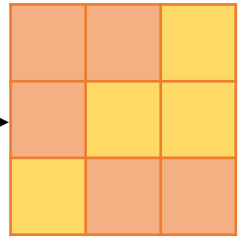
Decoder

Auto-encoder

$H \times W \times 3$

Encoder

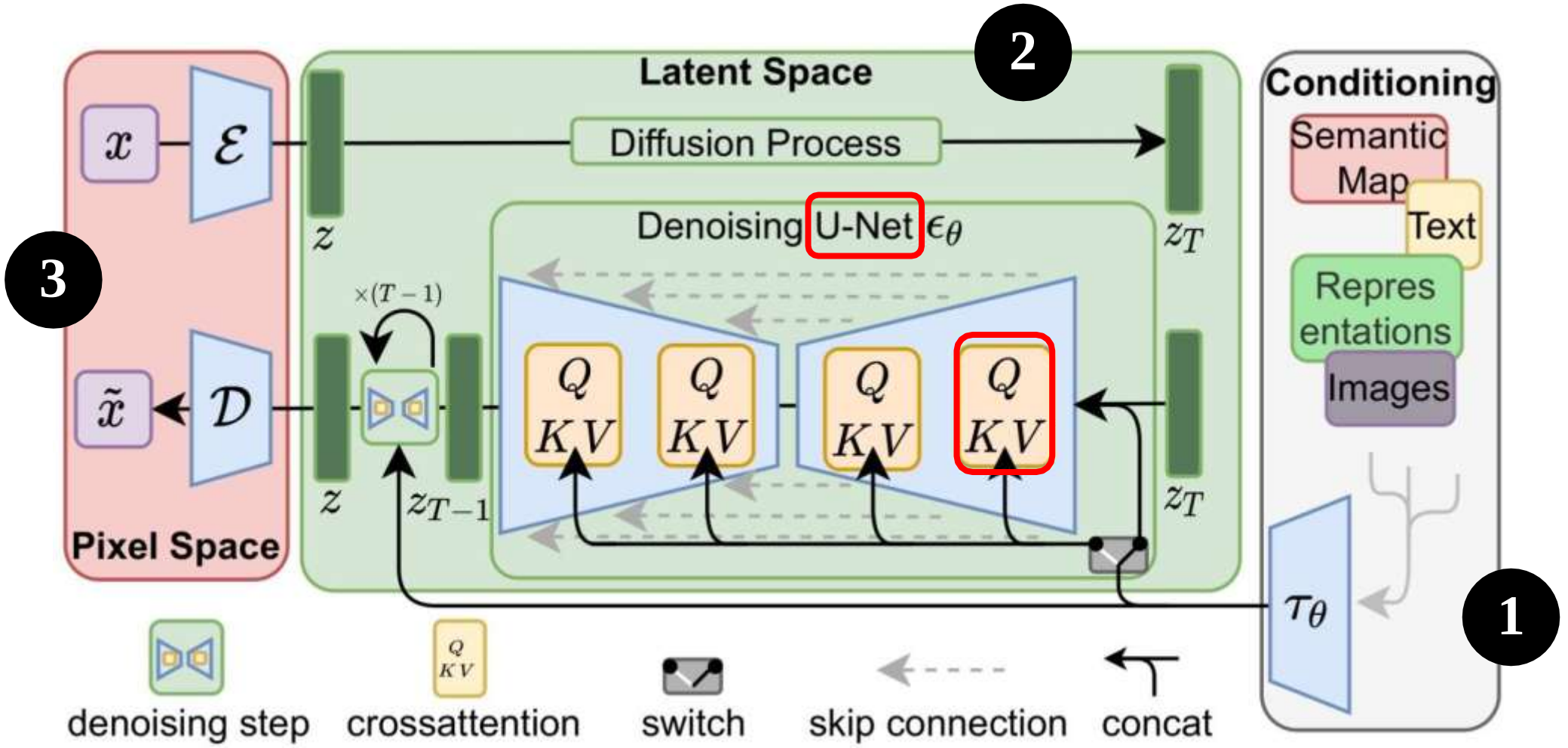
$h \times w \times c$



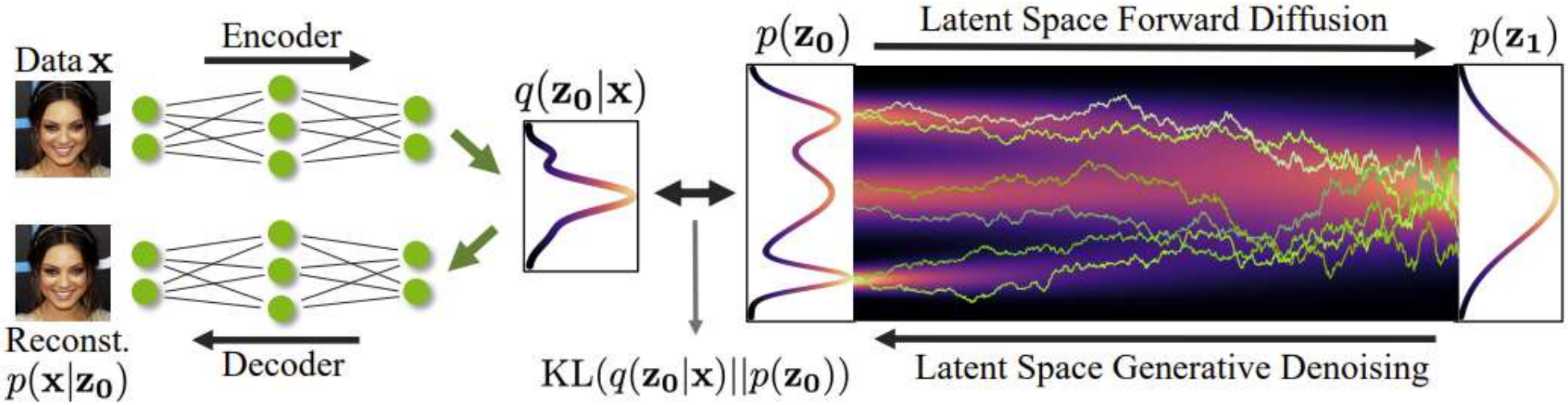
Decoder

Stable Diffusion

https://arxiv.org/abs/2112.10752



Latent diffusion model

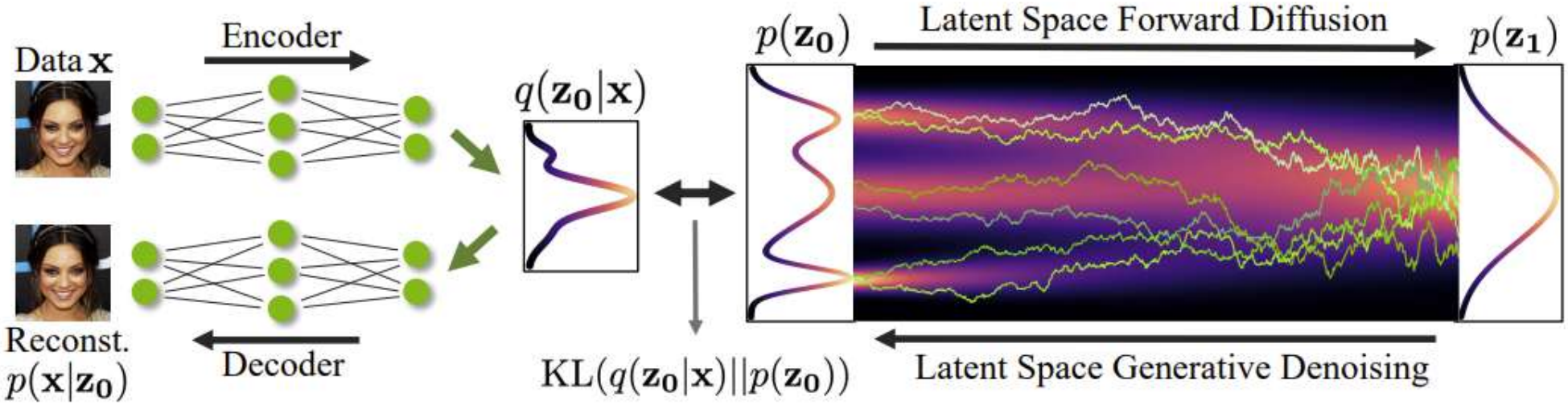


VAE mapping data to lower dimensional space

1. **Faster**
2. **Can be applied to any data type (e.g. discrete)**

Diffusion model prior over the latent space of the autoencoder

Stable diffusion text2image model



VAE mapping data to lower dimensional space

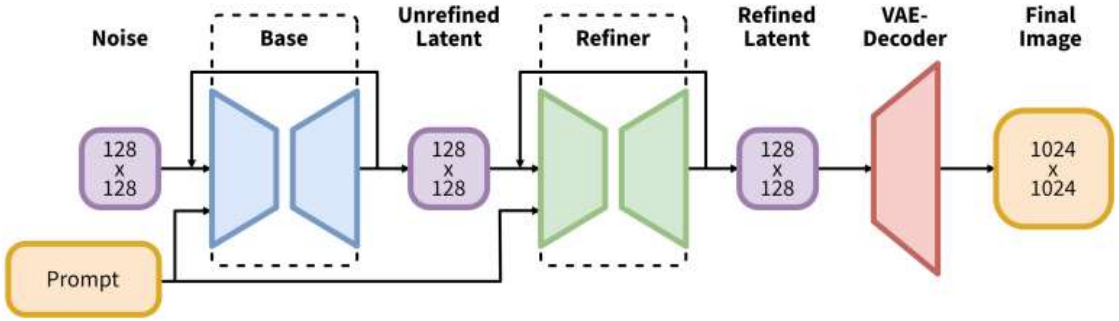
1. Pre-trained, focus on reconstruction (autoencoder)

Diffusion model prior over the latent space of the autoencoder

2. Trained in the second stage, keeping initial autoencoder fixed

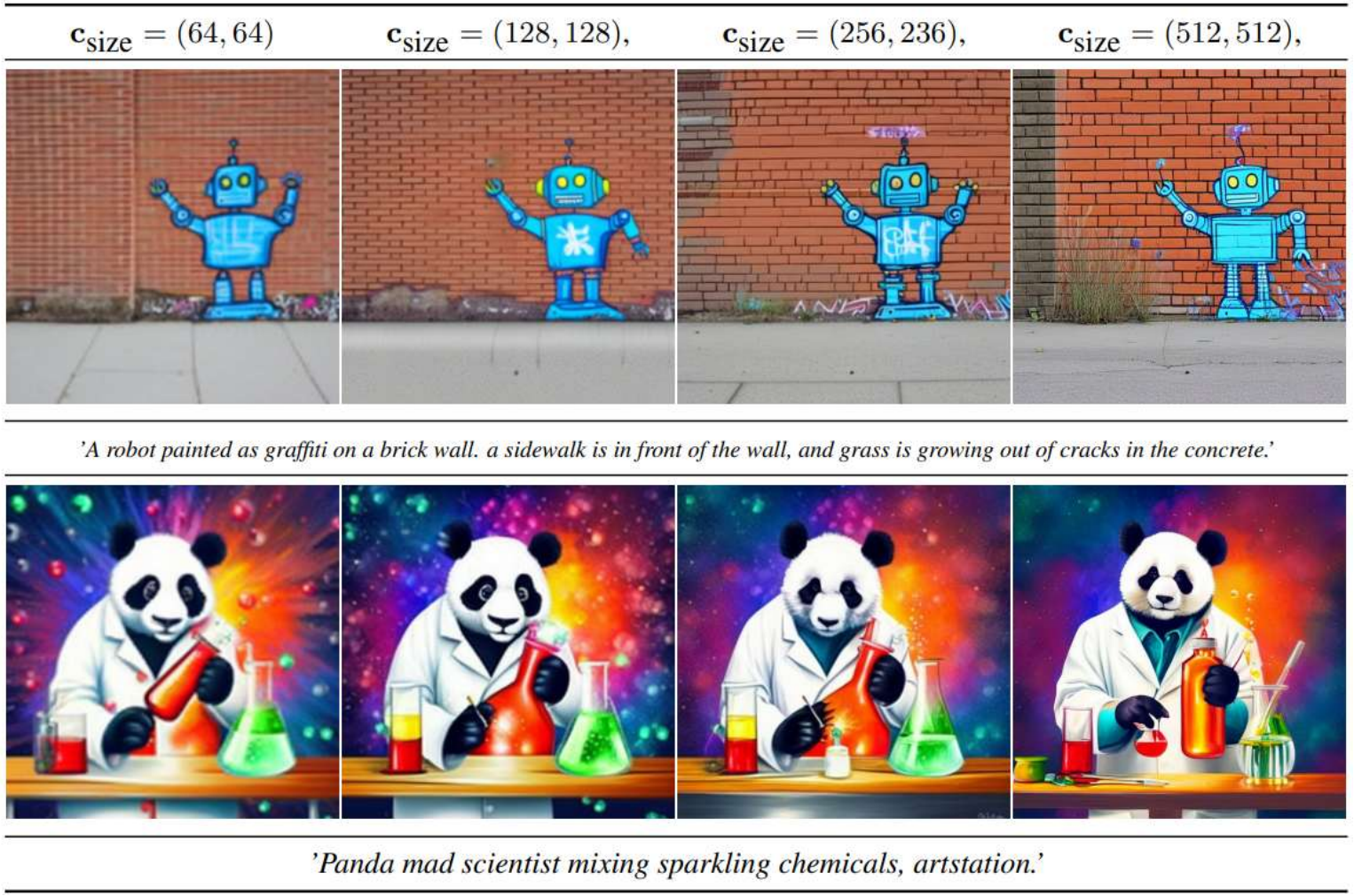
Large scale, open source model, widely adopted

SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis



https://arxiv.org/pdf/2307.01952

SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis



https://arxiv.org/pdf/2307.01952

SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis

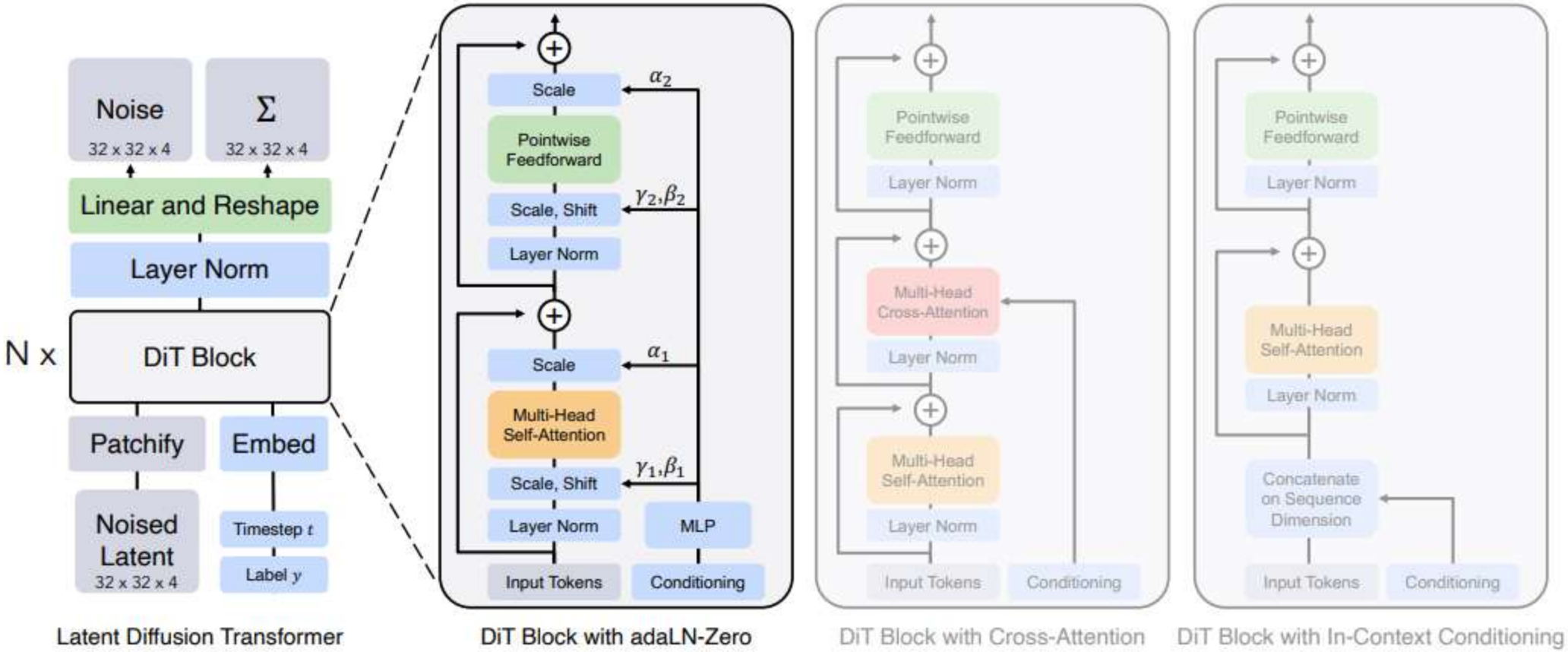
$c_{\text{crop}} = (0, 0)$	$c_{\text{crop}} = (0, 256)$,	$c_{\text{crop}} = (256, 0)$,	$c_{\text{crop}} = (512, 512)$,
			
'An astronaut riding a pig, highly realistic dslr photo, cinematic shot.'			
			
'A capybara made of lego sitting in a realistic, natural field.'			

https://arxiv.org/pdf/2307.01952

SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis

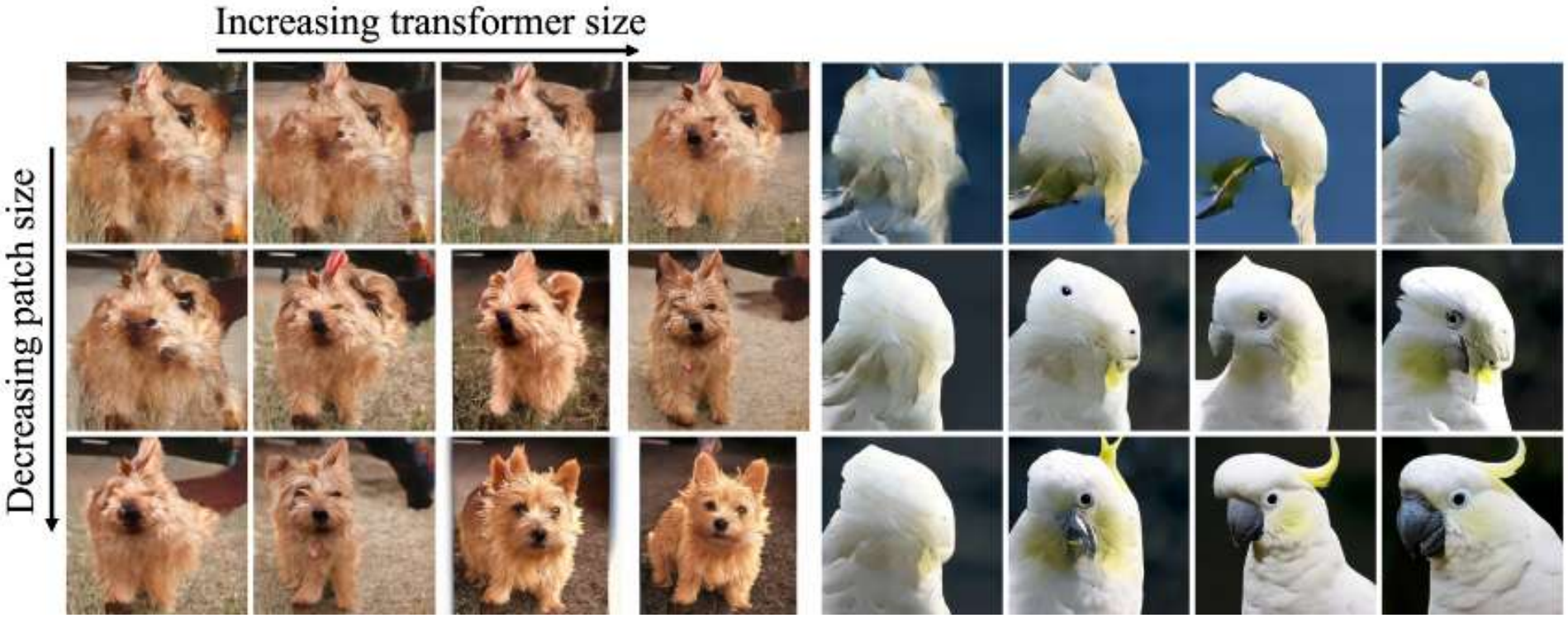
https://arxiv.org/pdf/2307.01952

DiT: Scalable Diffusion Models with Transformers



https://arxiv.org/pdf/2212.09748

DiT: Scalable Diffusion Models with Transformers



https://arxiv.org/pdf/2212.09748

DiT: Scalable Diffusion Models with Transformers

DiT “Dog-Ball”

BigGAN “Dog-Ball”

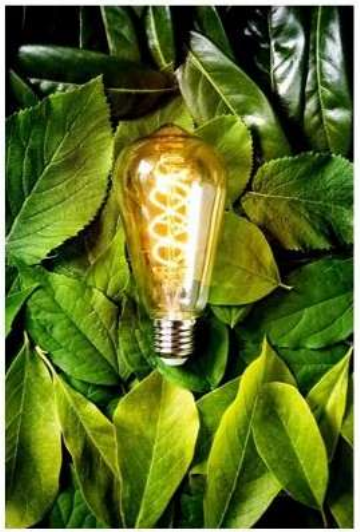
https://arxiv.org/pdf/2212.09748

DiT: Scalable Diffusion Models with Transformers

https://arxiv.org/pdf/2212.09748

CLIP is good but…

It cannot tell these apart!



(a) some plants surrounding a lightbulb

(b) a lightbulb surrounding some plants

Thrush et al. "Winoground: Probing vision and language models for visio-linguistic compositionality". CVPR 2022. https://arxiv.org/pdf/2204.03162.pdf

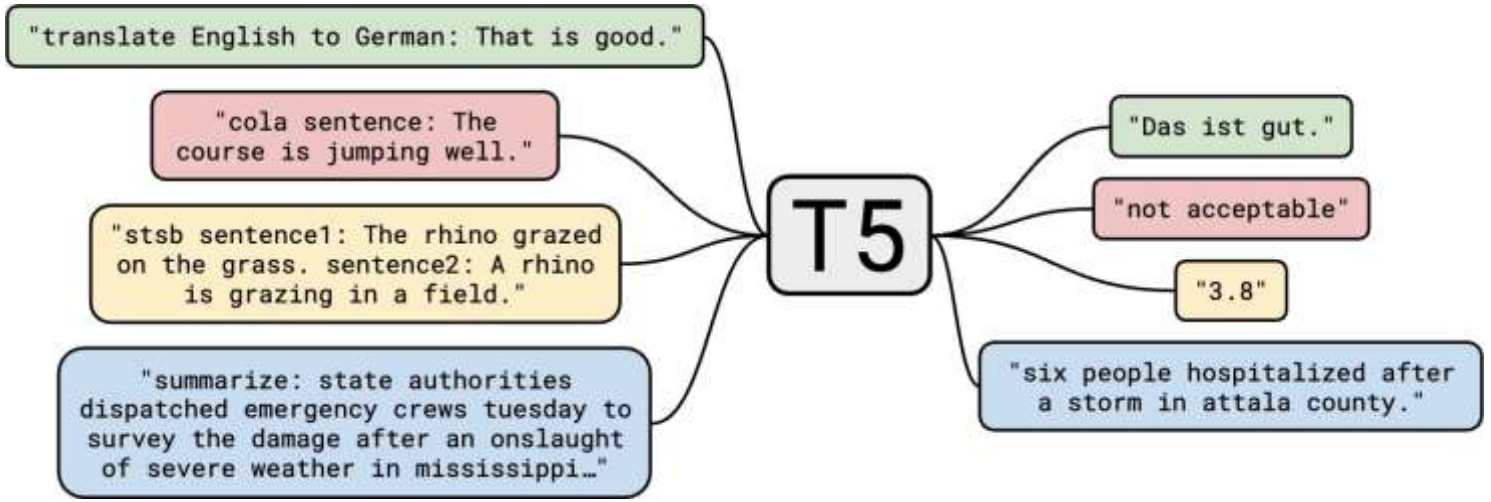
CLIP is good but…

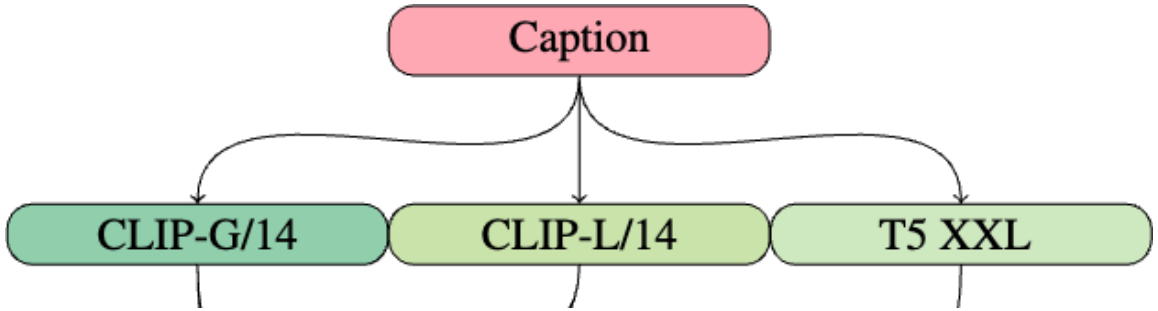
- It cannot handle spatial relation ships well
- It cannot handle negations well
- It cannot handle counts well
- The length limit is 77 tokens
- Sometimes ignores some details

CLIP is not a very good text encoder!

- What we want:
- Can capture semantics well ✗
- Can encode long sentences ✗
- Can attend to details ✗
- Can differentiate between different spatial relationships described in text ✗

Attempt : Add another text encoder on top of CLIP for better language understanding

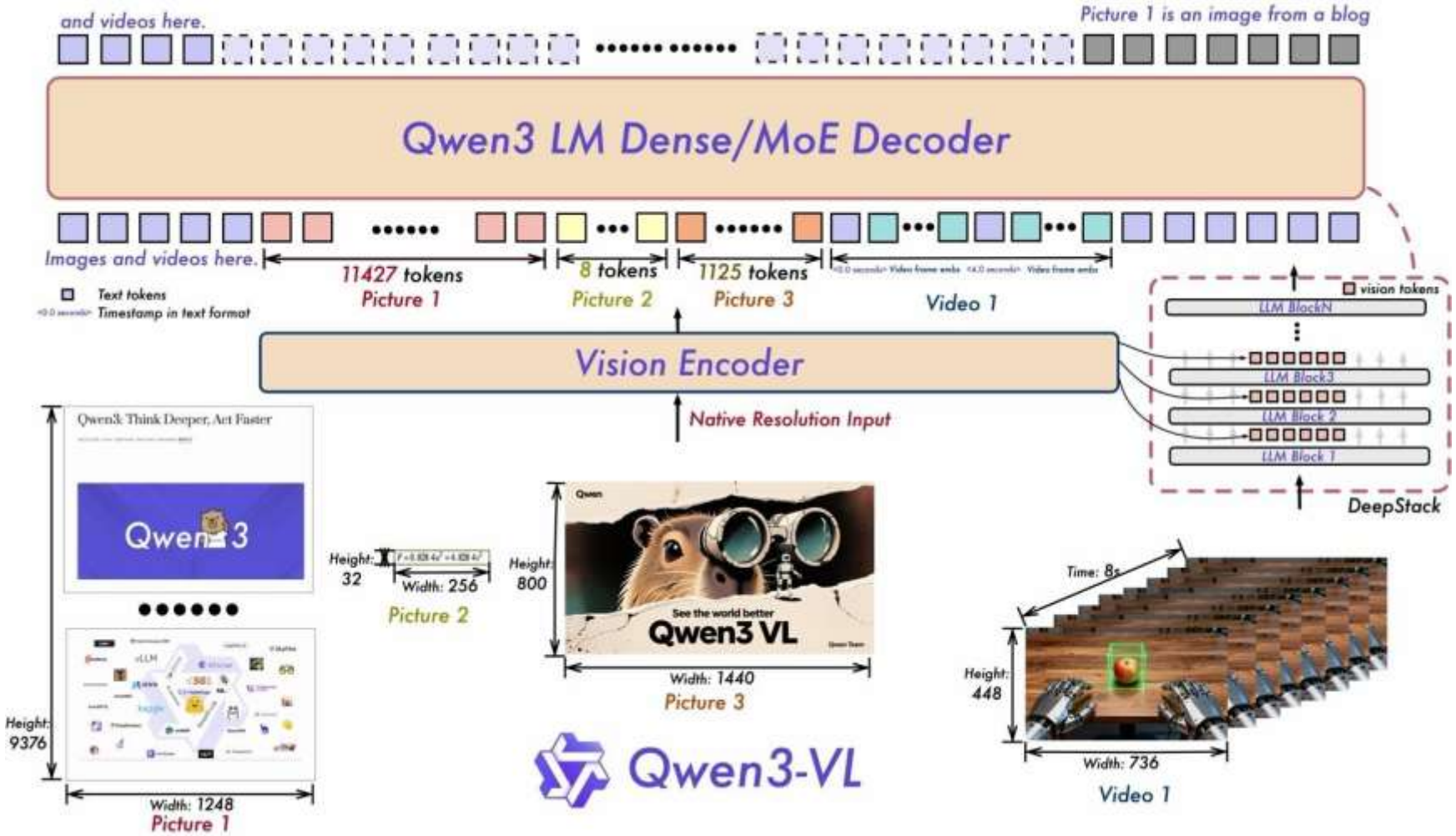




Raffel et al. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". JMLR 2020. https://arxiv.org/pdf/1910.10683

Esser et al. "Scaling Rectified Flow Transformers for High-Resolution Image Synthesis". https://arxiv.org/pdf/2403.03206

Attempt : Why not just use an LLM/VLM/MLLM

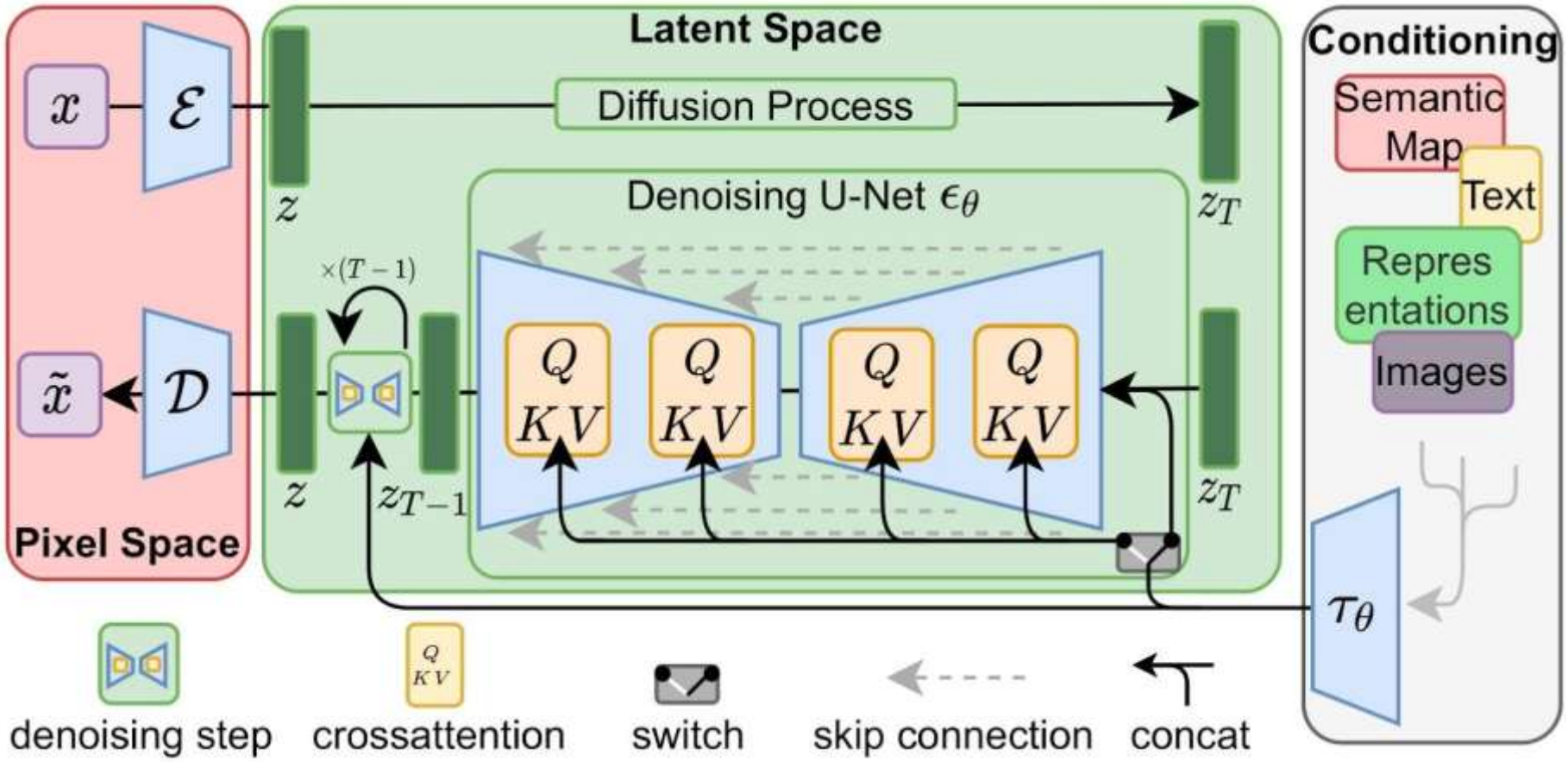


Qwen Team. “Qwen3-VL Technical Report”. https://arxiv.org/pdf/2511.21631

How to input text into an image generative model?

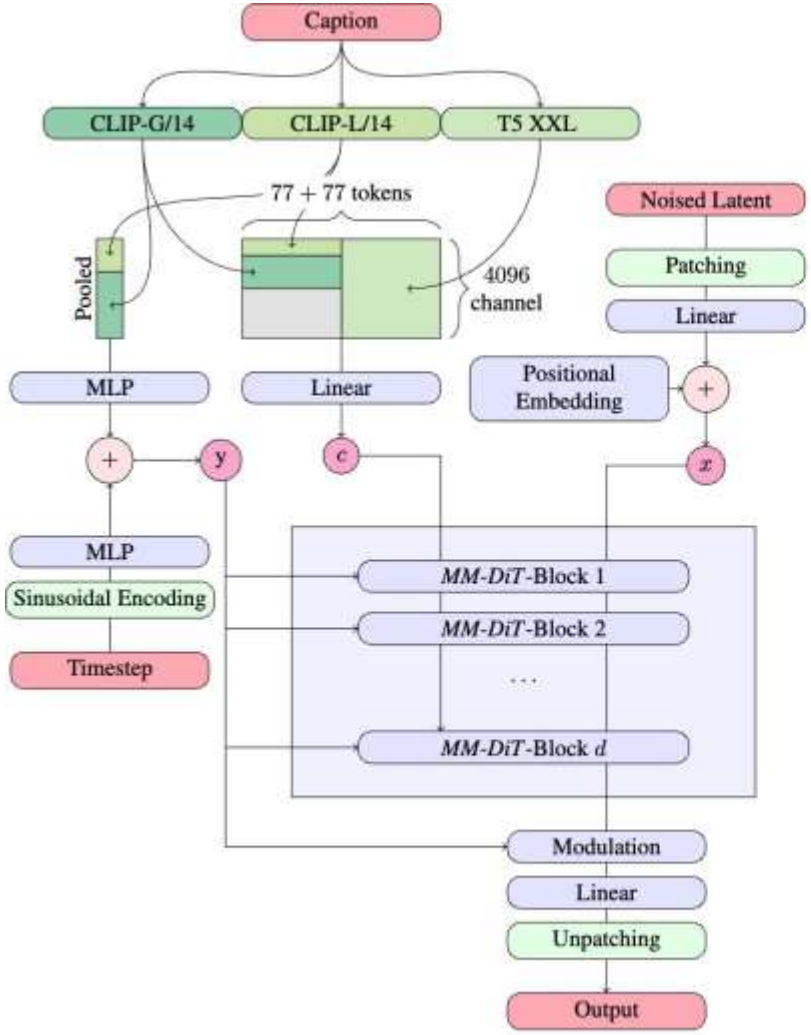
1. Somehow encode text into some feature vectors 
2. Somehow squish the encoded text features into the diffusion model 

Attempt 1: Cross Attention

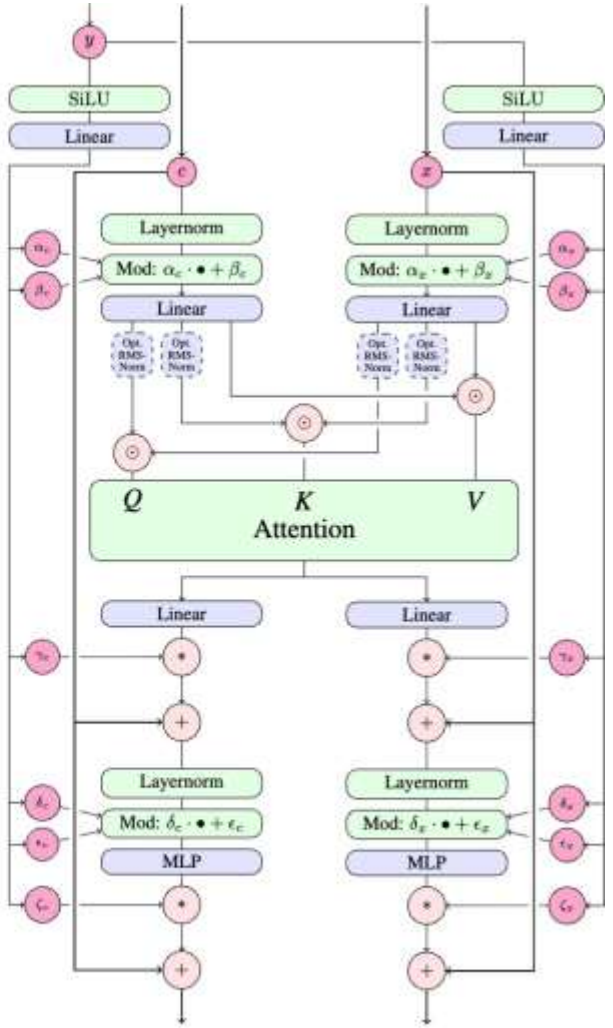


Rombach et al. "High-Resolution Image Synthesis with Latent Diffusion Models". CVPR 2022. https://arxiv.org/pdf/2112.10752

Attempt 2: Double Stream Multimodal-DiT



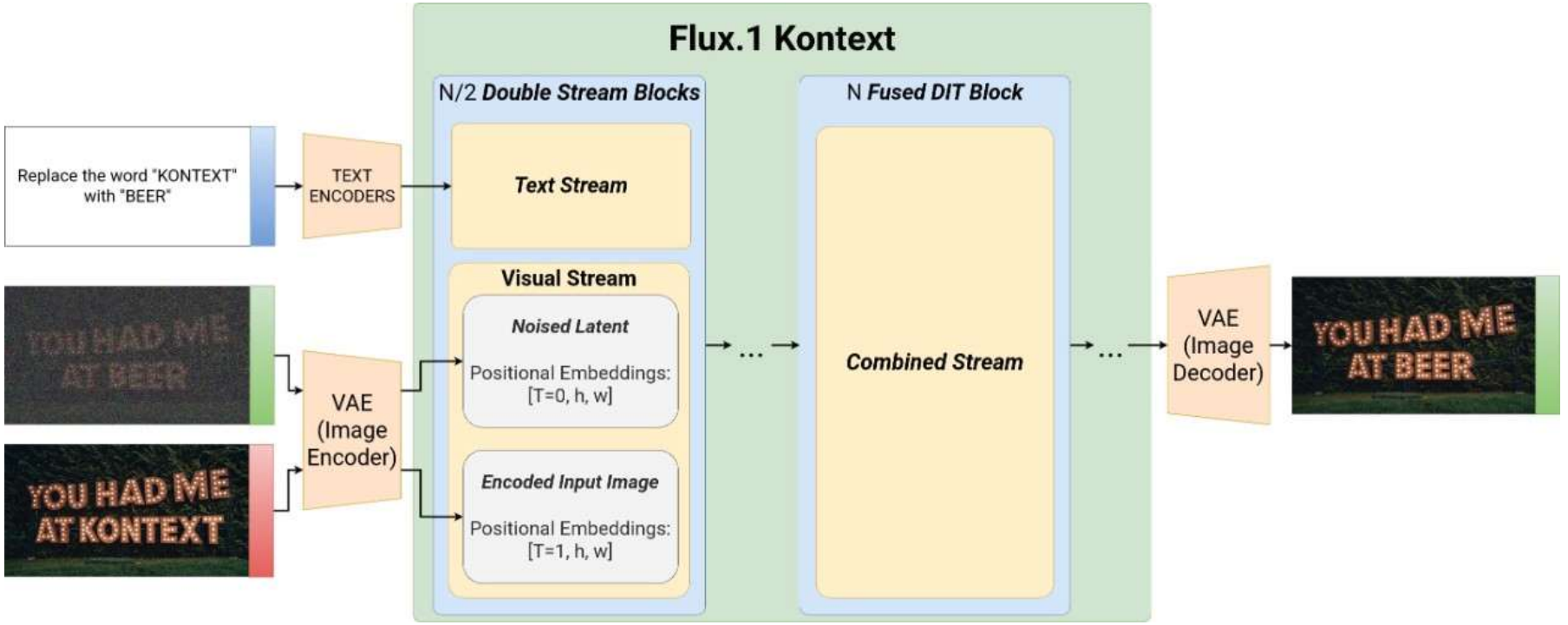
(a) Overview of all components.



(b) One *MM-DiT* block

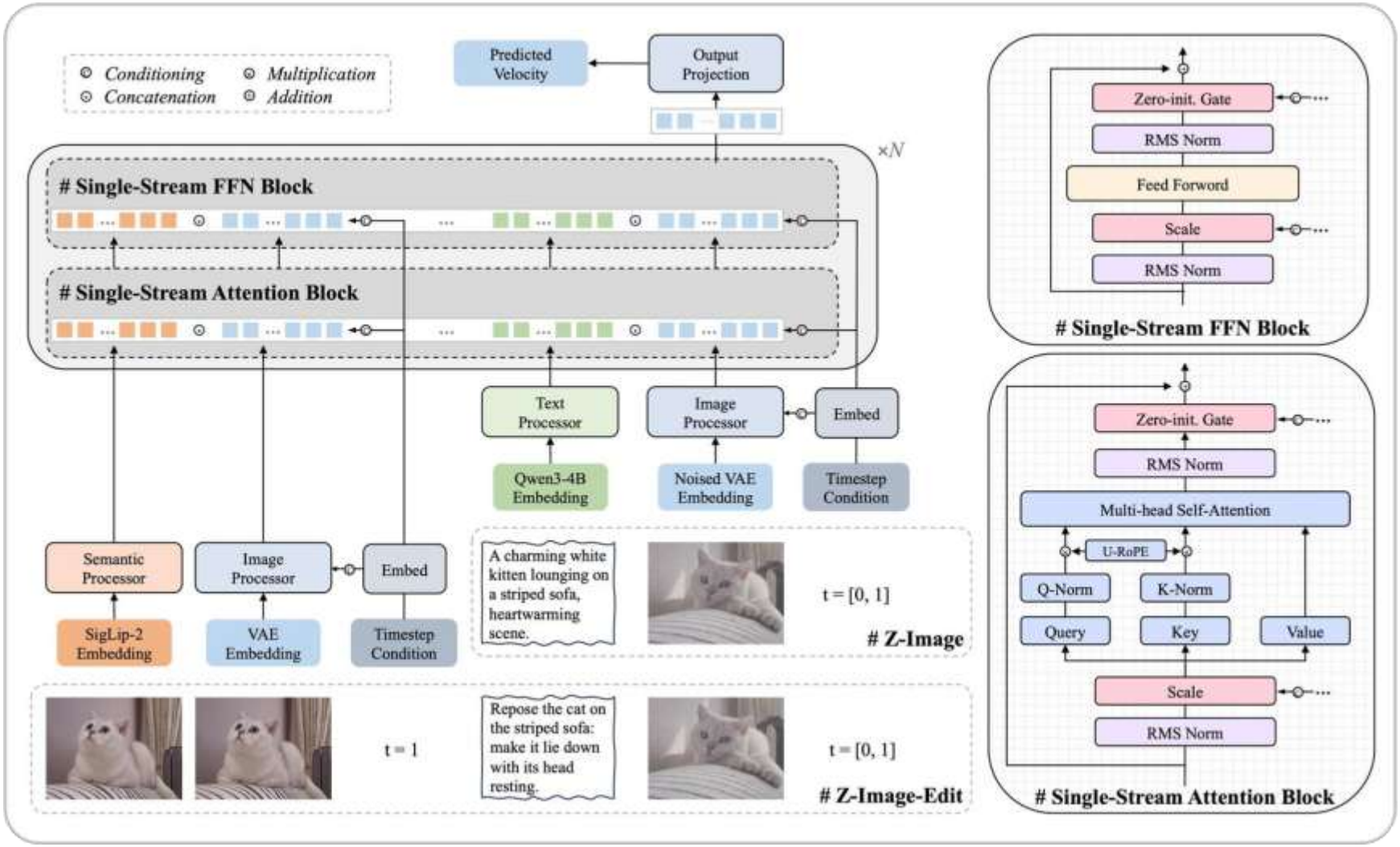
Esser et al. "Scaling Rectified Flow Transformers for High-Resolution Image Synthesis". https://arxiv.org/pdf/2403.03206

Attempt 2: Double stream -> merged stream MM-DiT



Black Forest Labs. "FLUX.1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space". https://arxiv.org/pdf/2506.15742

Attempt 3: Single stream MM-DiT



Z-Image Team, Alibaba Group. "Z-Image: An Efficient Image Generation Foundation Model with Single-Stream Diffusion Transformer". https://arxiv.org/pdf/2511.22699

Attempt 3.5: “Native multimodal model”

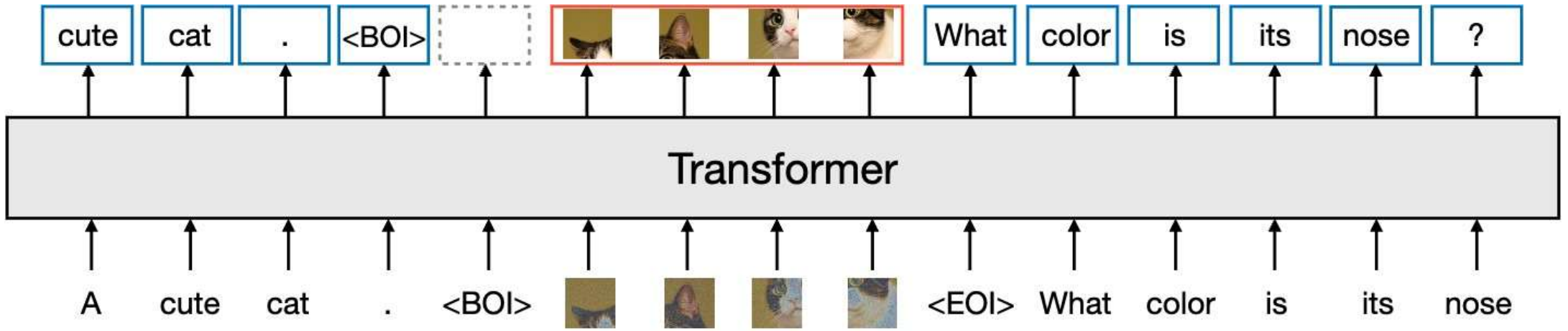
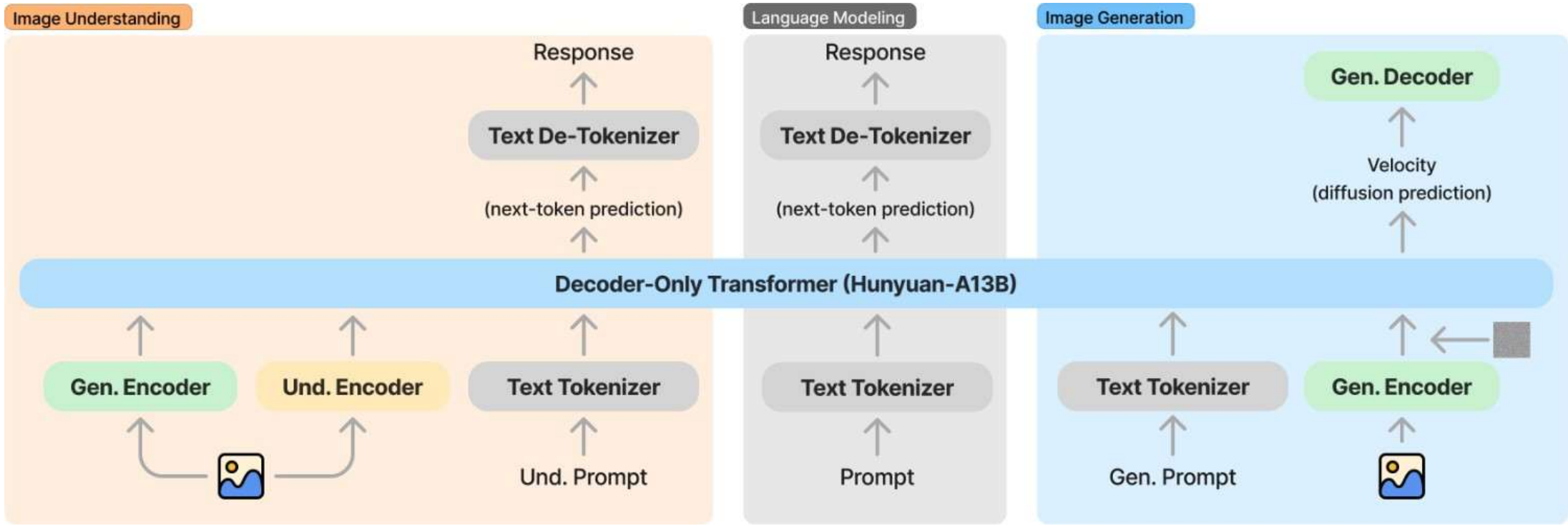


Figure 1: A high-level illustration of Transfusion. A single transformer perceives, processes, and produces data of every modality. Discrete (text) tokens are processed autoregressively and trained on the next token prediction objective. Continuous (image) vectors are processed together in parallel and trained on the diffusion objective. Marker BOI and EOI tokens separate the modalities.

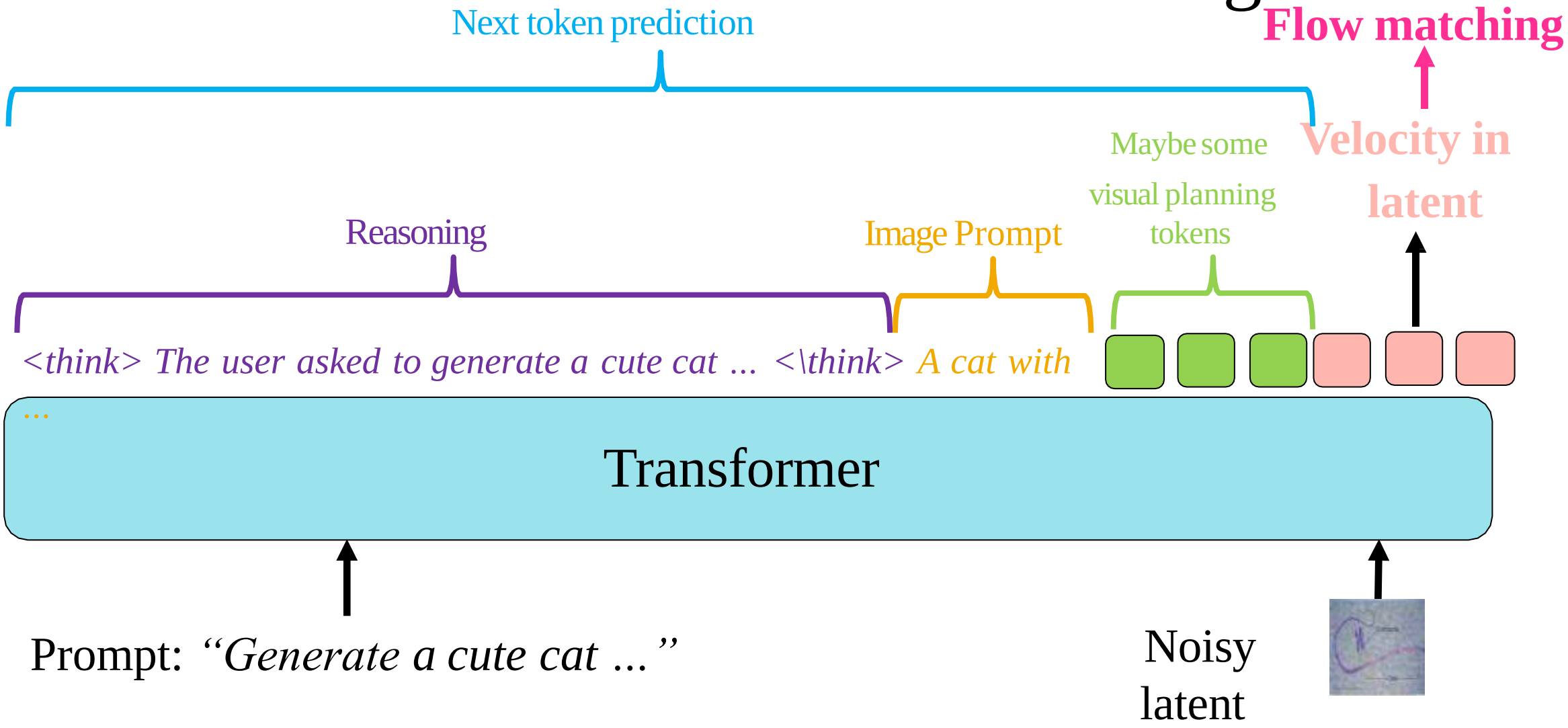
Zhou et al. “Transfusion: Predict the Next Token and Diffuse Images with One Multi-Modal Model”. https://arxiv.org/pdf/2408.11039

Attempt 3.5: “Native multimodal model”

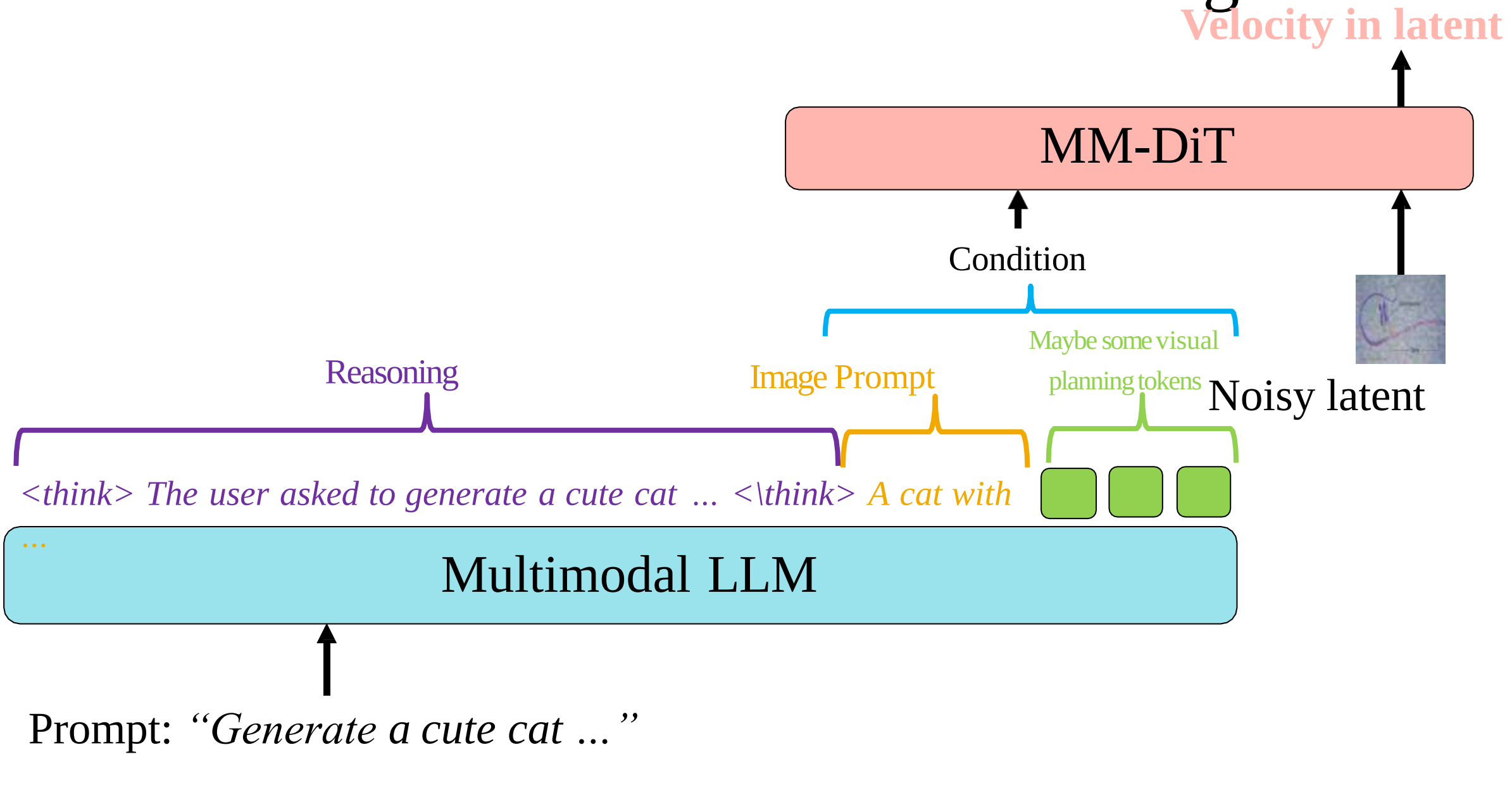


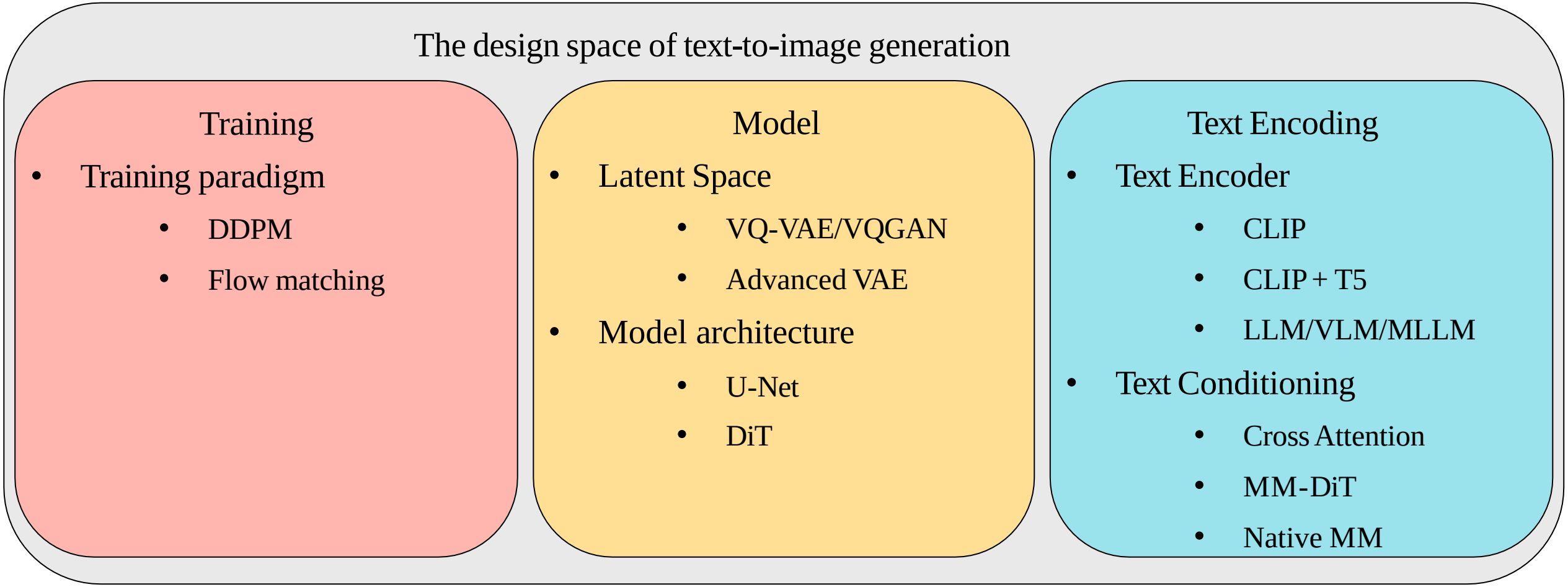
Tencent Hunyuan Foundation Model Team. “HunyuanImage 3.0 Technical Report”. https://arxiv.org/pdf/2408.11039

Guess on how Nano Banana & GPT-4o Image work



Guess on how Nano Banana & GPT-4o Image work




The design space of text-to-image generation
Training
Training paradigm
DDPM
Flow matching
Model
Latent Space
VQ-VAE/VQGAN
Advanced VAE
Model architecture
U-Net
DiT
Text Encoding
Text Encoder
CLIP
CLIP + T5
LLM/VLM/MLLM
Text Conditioning
Cross Attention
MM-DiT
Native MM

The design space of text-to-image generation

Training

- Training paradigm
  - DDPM
  - Flow matching

Model

- Latent Space
  - VQ-VAE/VQGAN
  - Advanced VAE
- Model architecture
  - U-Net
  - DiT

Text Encoding

- Text Encoder
  - CLIP
  - CLIP + T5
  - LLM/VLM/MLLM
- Text Conditioning
  - Cross Attention
  - MM-DiT
  - Native MM

This is Stable Diffusion 1 & 2!

The design space of text-to-image generation

Training

- Training paradigm
  - DDPM
  - Flow matching

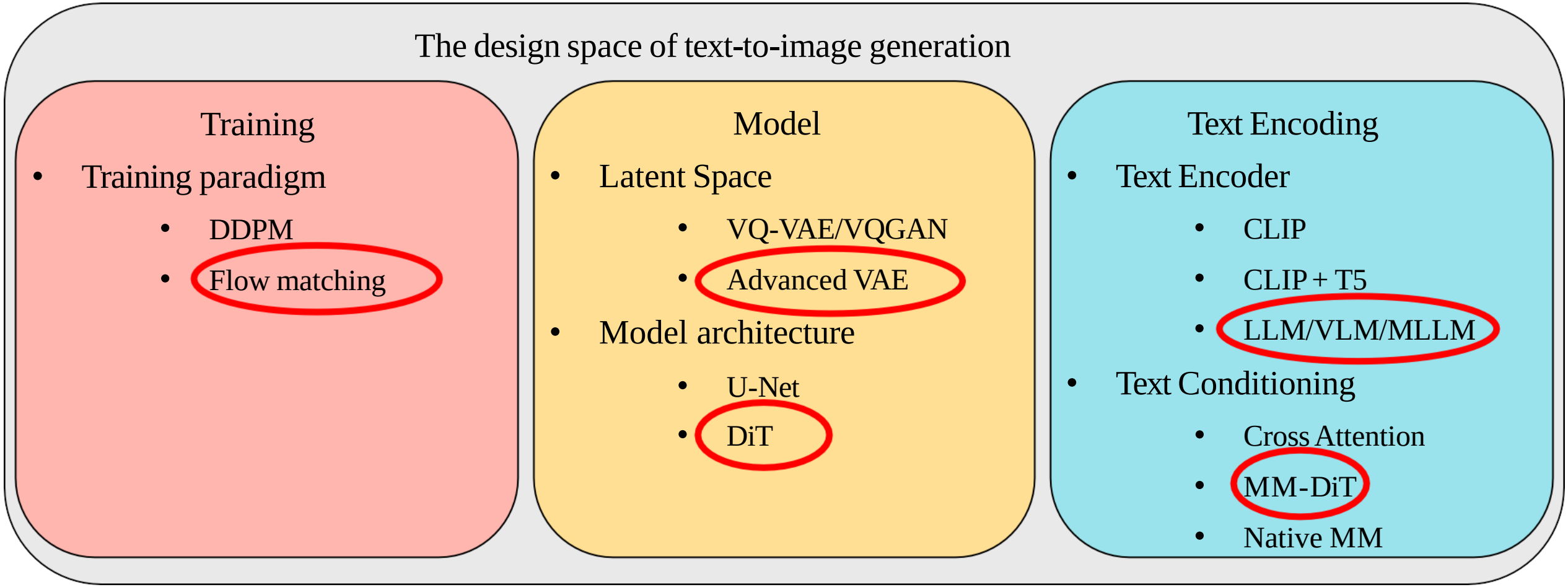
Model

- Latent Space
  - VQ-VAE/VQGAN
  - Advanced VAE
- Model architecture
  - U-Net
  - DiT

Text Encoding

- Text Encoder
  - CLIP
  - CLIP + T5
  - LLM/VLM/MLLM
- Text Conditioning
  - Cross Attention
  - MM-DiT
  - Native MM

This is Stable Diffusion 3 & Flux 1!



This is Flux 2, Z-Image, Qwen-Image, etc!

The design space of text-to-image generation

Training

- Training paradigm
  - DDPM
  - Flow matching

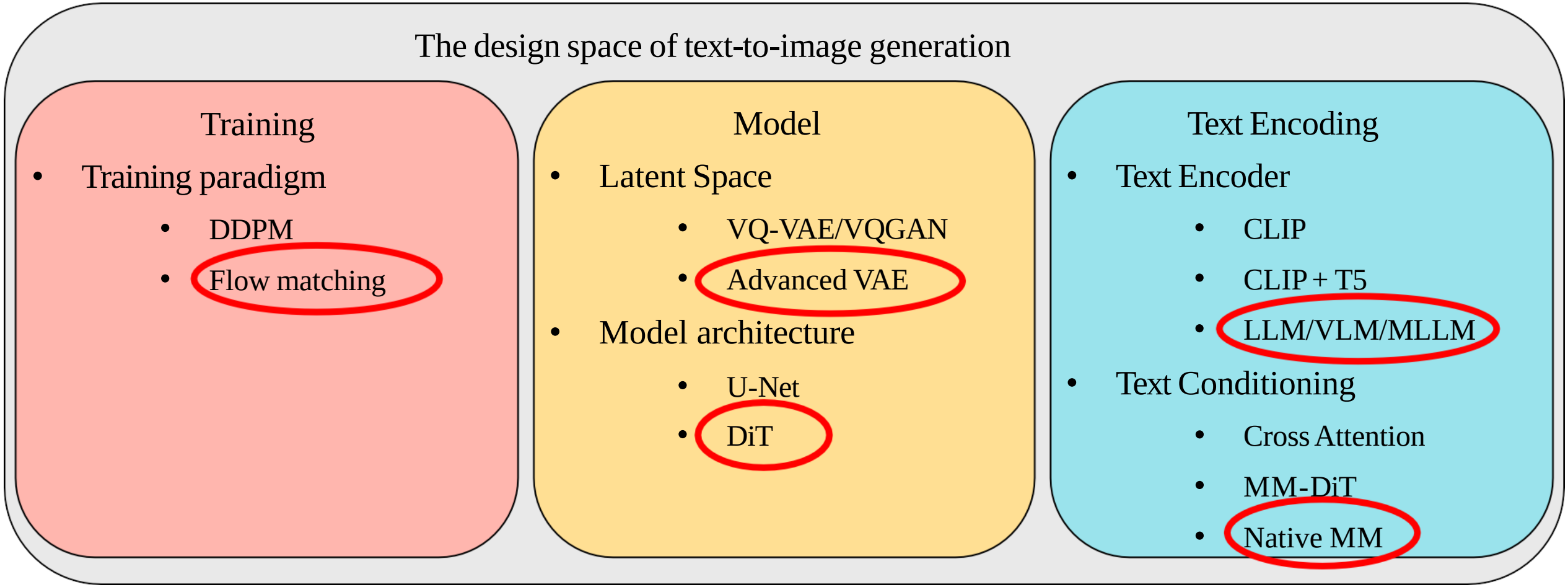
Model

- Latent Space
  - VQ-VAE/VQGAN
  - Advanced VAE
- Model architecture
  - U-Net
  - DiT

Text Encoding

- Text Encoder
  - CLIP
  - CLIP + T5
  - LLM/VLM/MLLM
- Text Conditioning
  - Cross Attention
  - MM-DiT
  - Native MM

This is Transfusion, Hunyuan 3.0, etc!



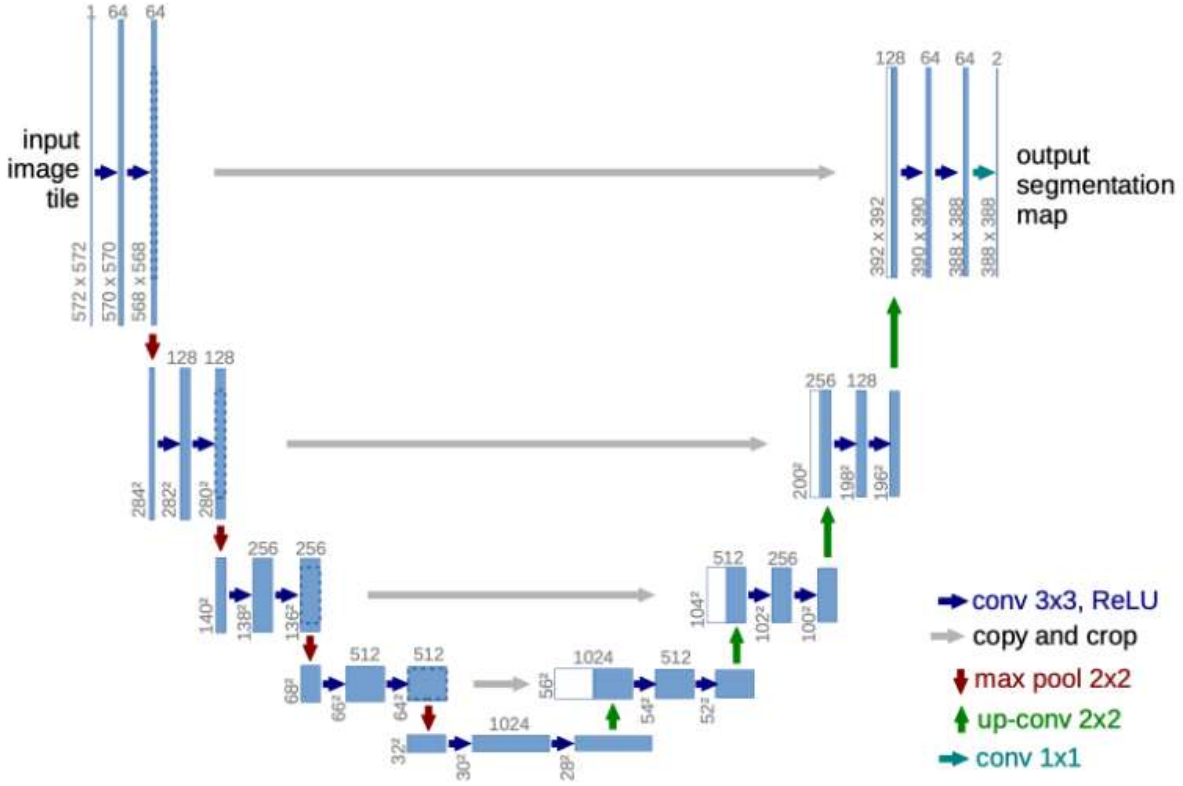
(Probably also Nano Banana & GPT-4o Image)

Video Generation

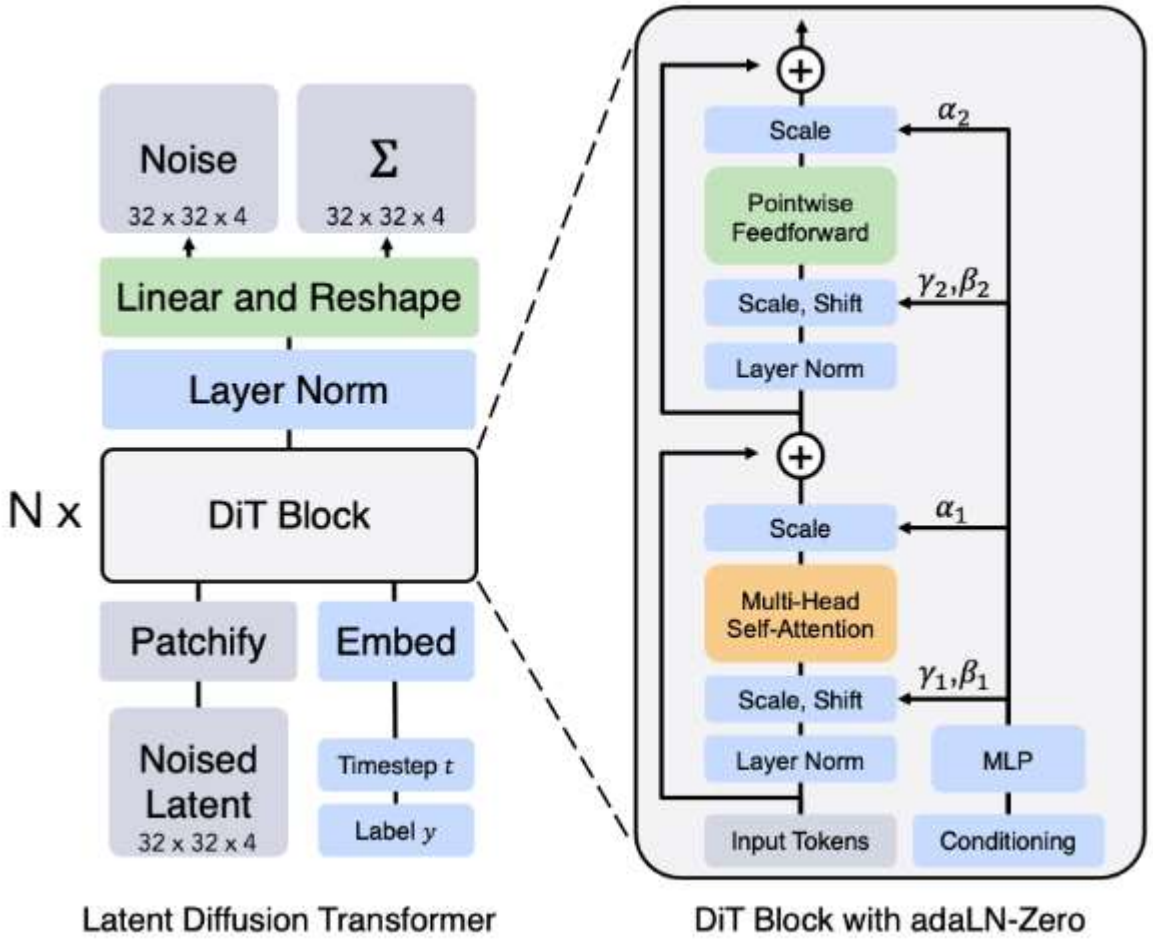
Sora

https://openai.com/index/sora/

Model Architecture

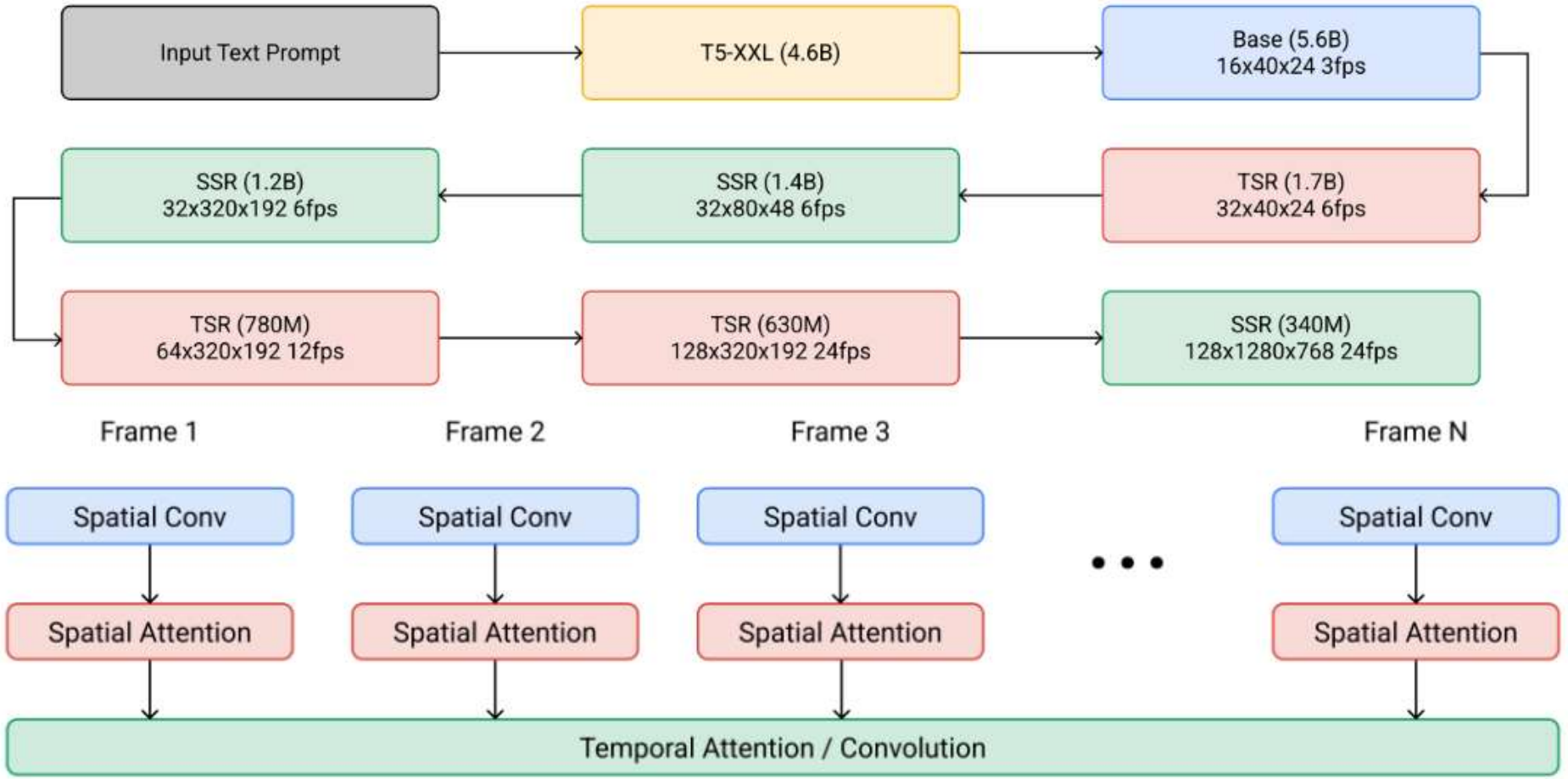


U-Net: https://arxiv.org/abs/1505.04597



DiT: https://arxiv.org/abs/2212.09748

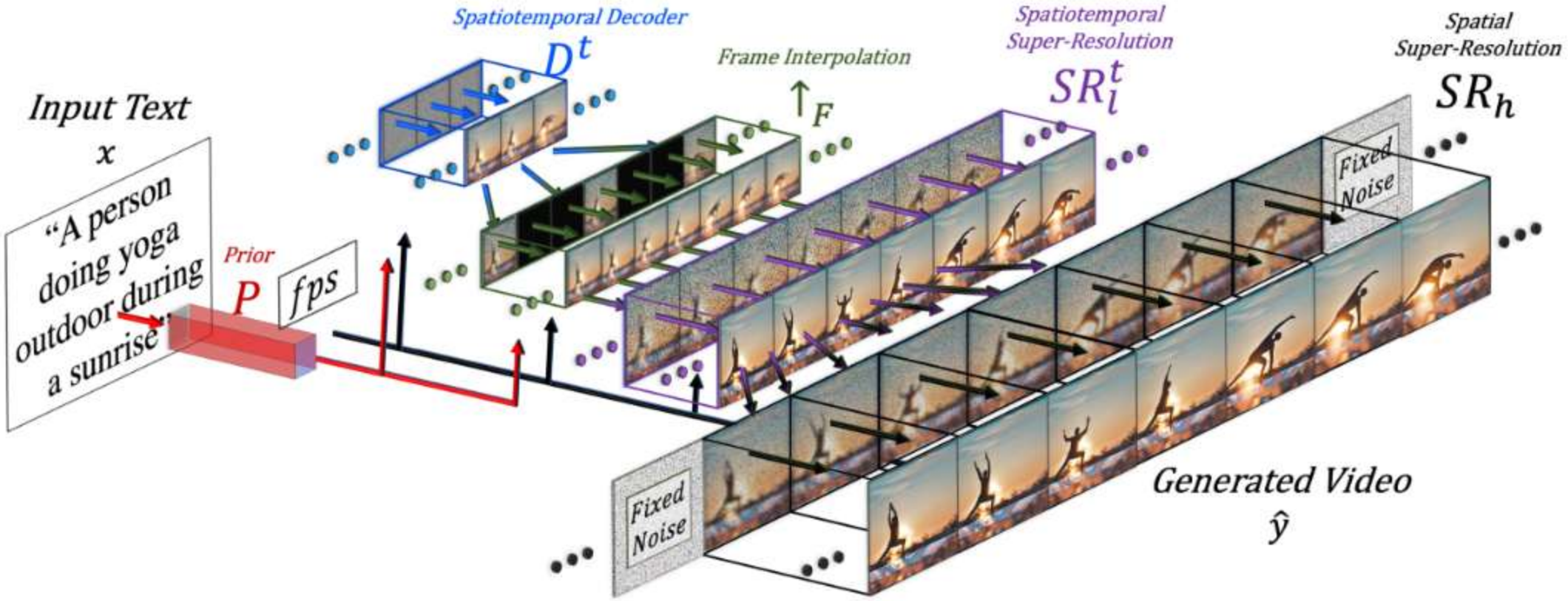
Imagen Video



Upper: The cascaded sampling pipeline in Imagen Video. In practice, the text embeddings are injected into all components, not just the base model.
Below: The architecture of one space-time separable block in the Imagen Video diffusion model.

https://arxiv.org/abs/2210.02303

Adapting Image Models to Generate Videos



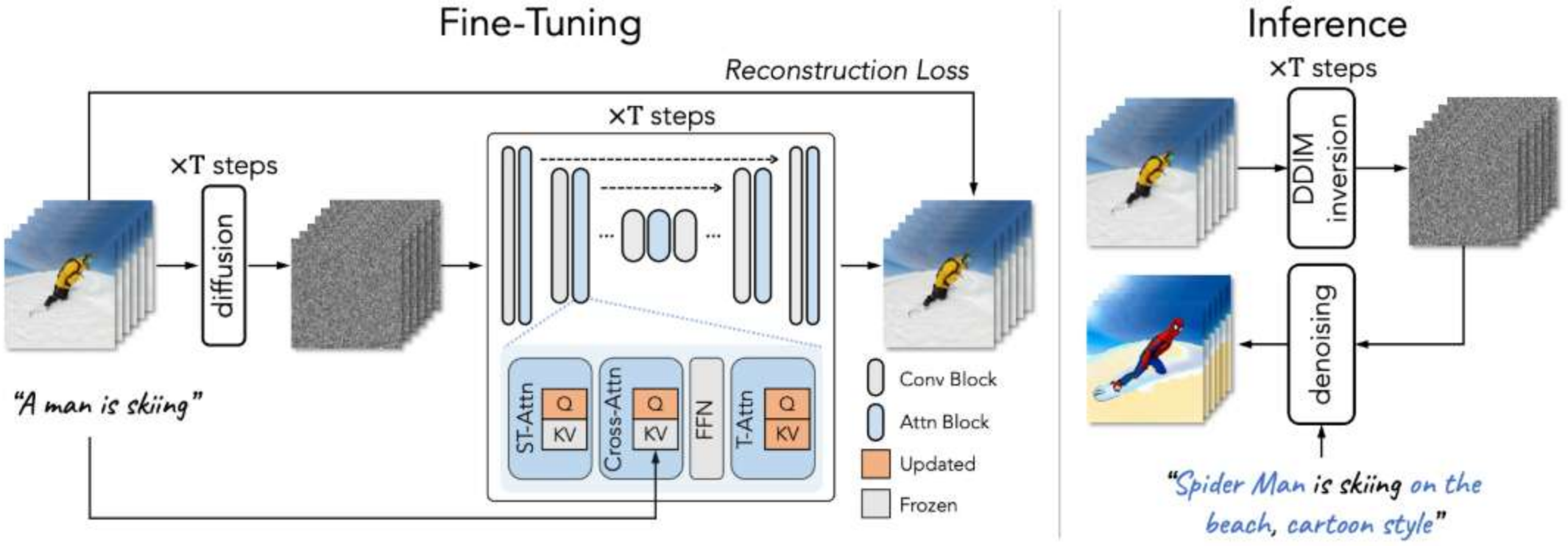
$$\hat{\mathbf{y}}_t = \mathrm{SR}_h \circ \mathrm{SR}_l^t \circ \uparrow_F \circ D^t \circ P \circ (\hat{\mathbf{x}}, \mathrm{CLIP}_{\mathrm{text}}(\mathbf{x}))$$

Make-A-Video https://arxiv.org/abs/2209.14792

Adapting Image Models to Generate Videos

https://ai.meta.com/blog/generative-ai-text-to-video/

Adapting Image Models to Generate Videos



$$\mathbf{Q} = \mathbf{W}^Q \mathbf{z}_{v_i}, \quad \mathbf{K} = \mathbf{W}^K [\mathbf{z}_{v_1}, \mathbf{z}_{v_{i-1}}], \quad \mathbf{V} = \mathbf{W}^V [\mathbf{z}_{v_1}, \mathbf{z}_{v_{i-1}}]$$

$$\mathbf{O} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right) \cdot \mathbf{V}$$

Tune-A-Video https://arxiv.org/abs/2212.11565

Adapting Image Models to Generate Videos

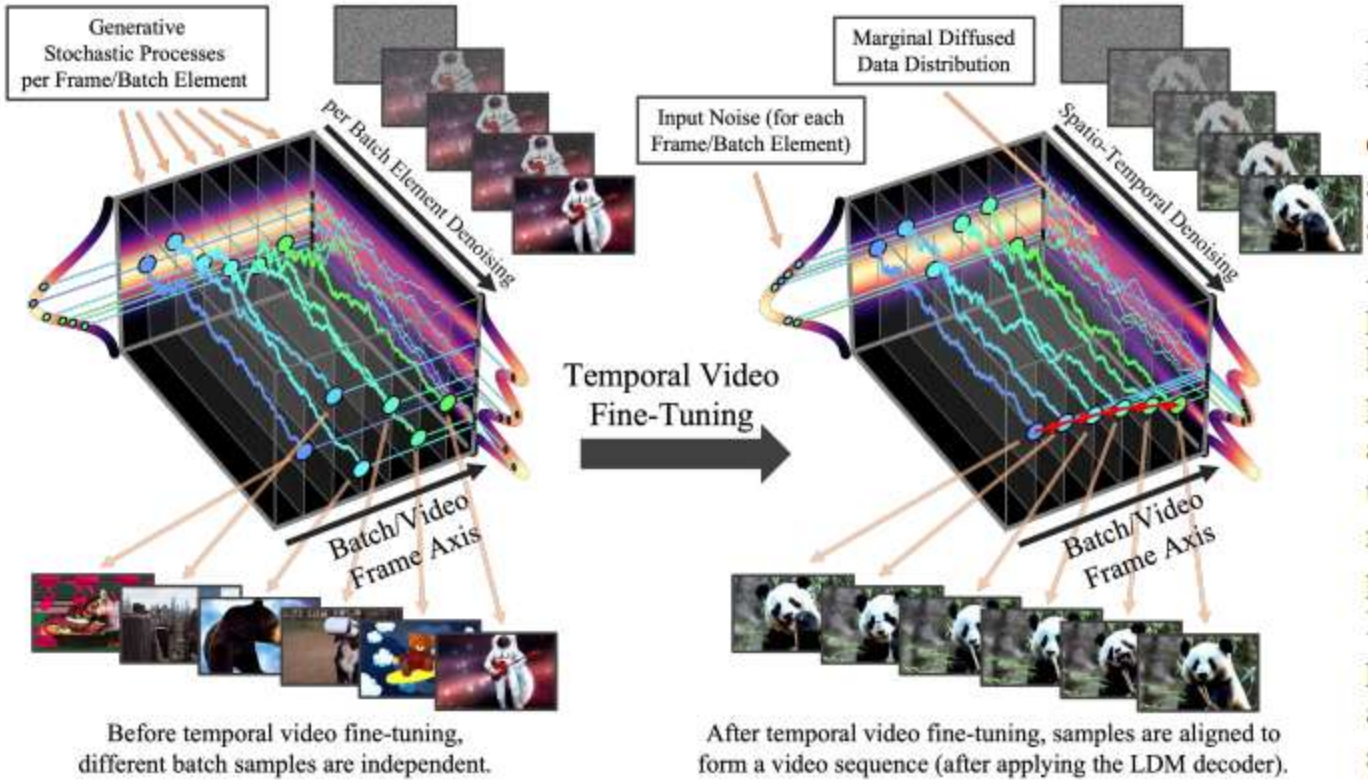
"A man is skiing"

"Spider Man is skiing on the beach, cartoon style”

"A man, wearing pink clothes, is skiing at sunset"

Tune-A-Video https://arxiv.org/abs/2212.11565

Adapting Image Models to Generate Videos



Before temporal video fine-tuning, different batch samples are independent.

After temporal video fine-tuning, samples are aligned to form a video sequence (after applying the LDM decoder).

Figure 2. **Temporal Video Fine-Tuning.** We turn pre-trained image diffusion models into temporally consistent video generators. Initially, different samples of a batch synthesized by the model are independent. After temporal video fine-tuning, the samples are temporally aligned and form coherent videos. The stochastic generation process before and after fine-tuning is visualised for a diffusion model of a one-dim. toy distribution. For clarity, the figure corresponds to alignment in pixel space. In practice, we perform alignment in LDM's latent space and obtain videos after applying LDM's decoder (see Fig. 3). We also video fine-tune diffusion model upsamplers in pixel or latent space (Sec. 3.4).

Video LDM https://arxiv.org/abs/2304.08818

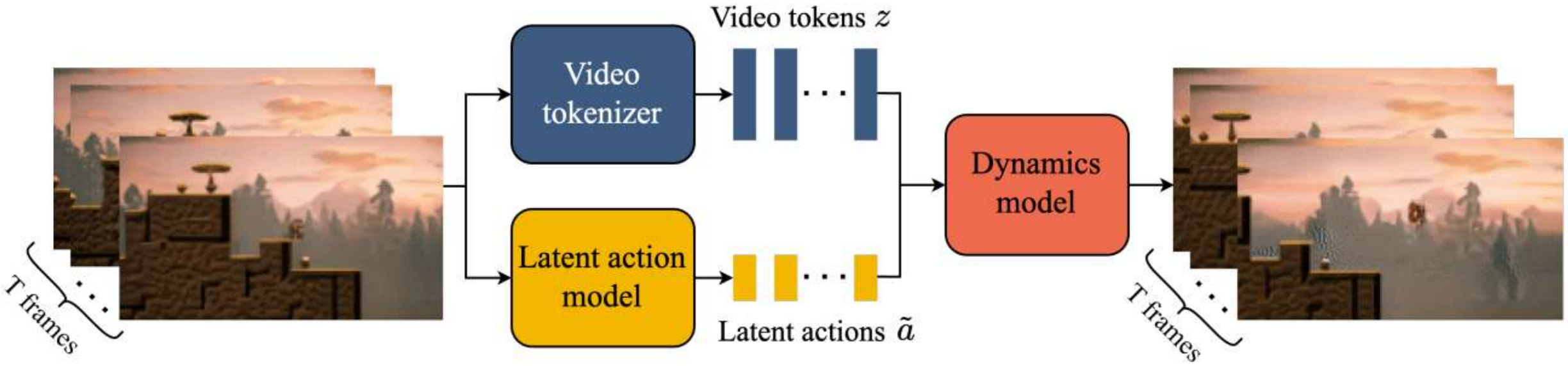
Adapting Image Models to Generate Videos

Video LDM https://arxiv.org/abs/2304.08818

Genie: Generative Interactive Environments

https://sites.google.com/view/genie-2024/home

Genie: Generative Interactive Environments



Genie takes in T frames of video as input, tokenizes them into discrete tokens $\boldsymbol{z}$ via the video tokenizer, and infers the latent actions $\boldsymbol{\tilde{a}}$ between each frame with the latent action model. Both are then passed to the dynamics model to generate predictions for the next frames in an iterative manner.

https://sites.google.com/view/genie-2024/home

Genie: Generative Interactive Environments

“Remarkably, Genie learns not only which parts of an observation are generally controllable, but also infers diverse latent actions that are consistent across the generated environments. Note here how the same latent actions yield similar behaviors across different prompt images. ”

https://sites.google.com/view/genie-2024/home

Genie: Generative Interactive Environments

The future of generative virtual worlds

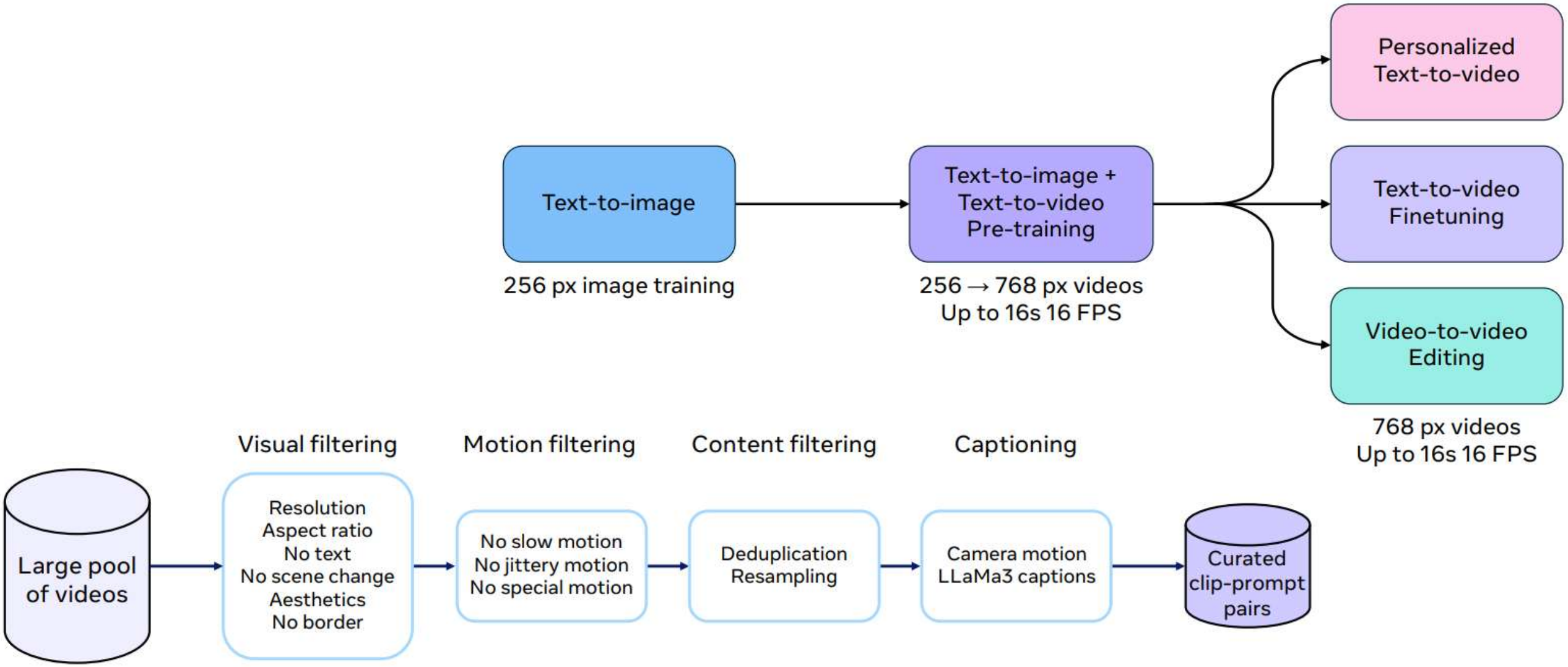
“Trajectories with the same latent action sequence typically display similar behaviors. ”

https://sites.google.com/view/genie-2024/home

Movie Gen: A Cast of Media Foundation Models

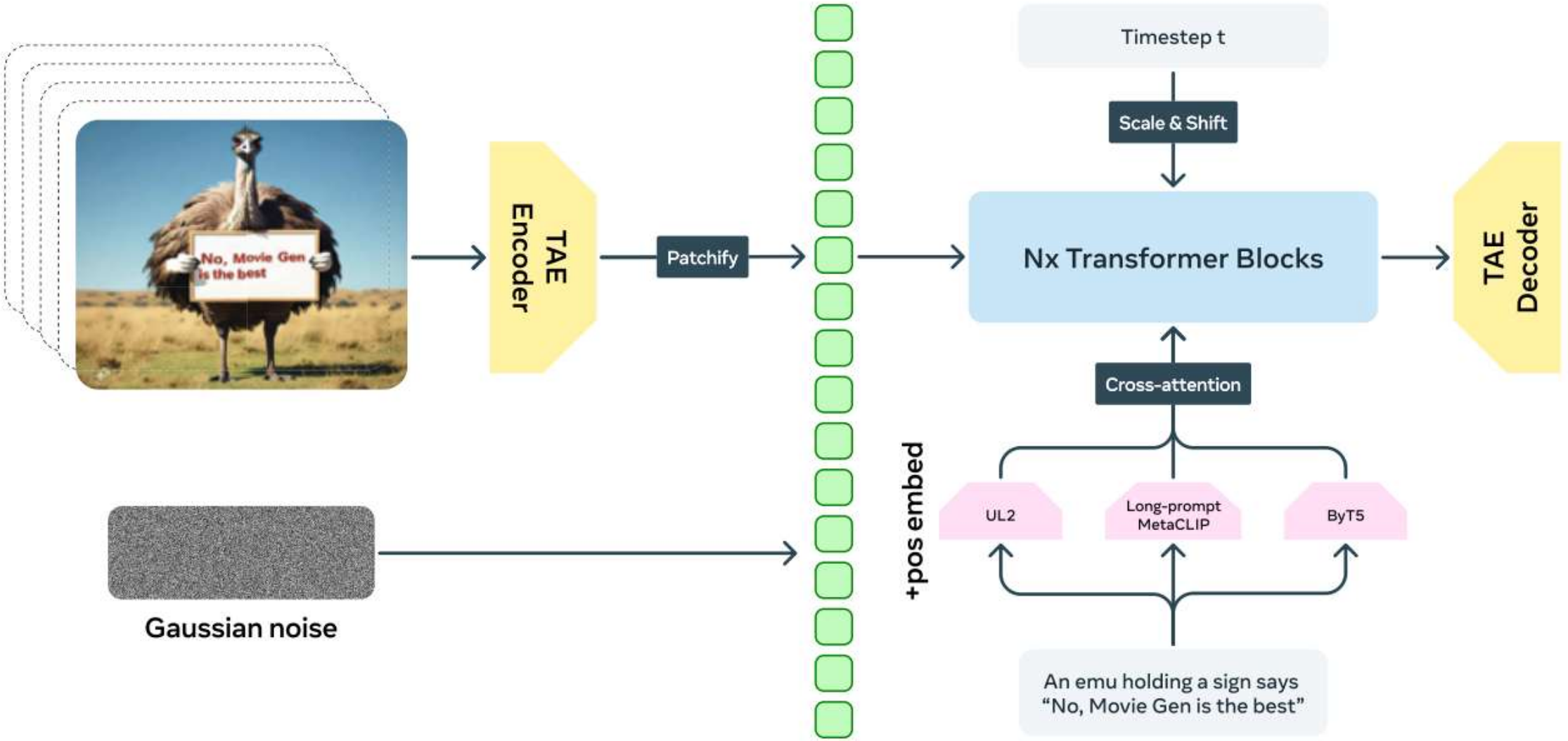
https://ai.meta.com/blog/movie-gen-media-foundation-models-generative-ai-video/

Movie Gen:
A Cast of Media Foundation Models



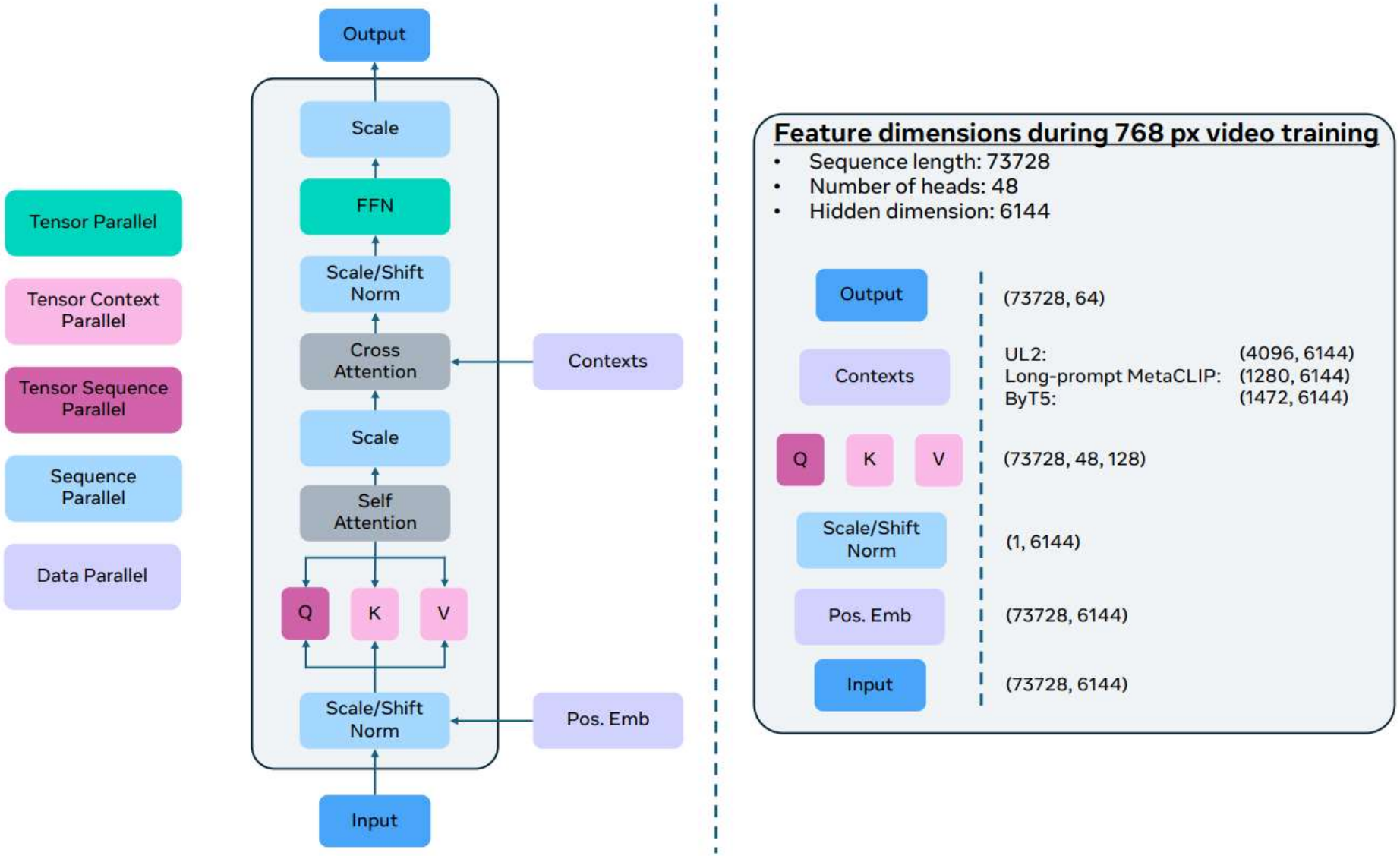
https://ai.meta.com/static-resource/movie-gen-research-paper

Movie Gen: A Cast of Media Foundation Models



https://ai.meta.com/static-resource/movie-gen-research-paper

Movie Gen: A Cast of Media Foundation Models



https://ai.meta.com/static-resource/movie-gen-research-paper

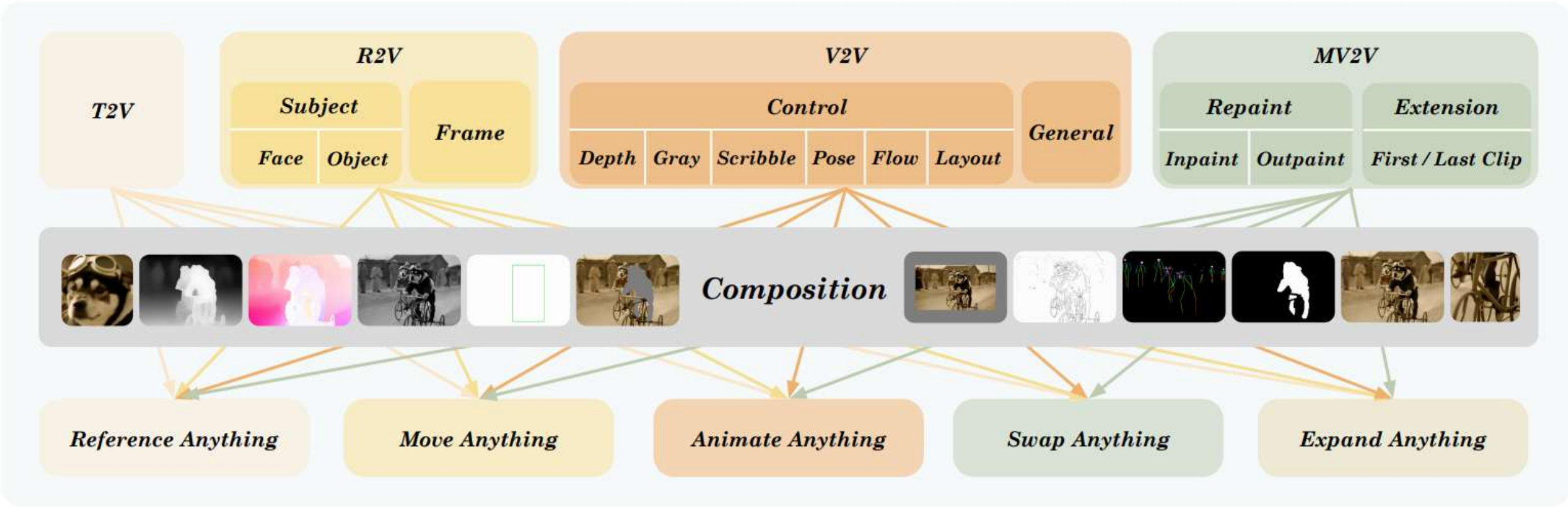
One Model to Workflow

*image taken from ChatGPT 4o

VACE/Wan

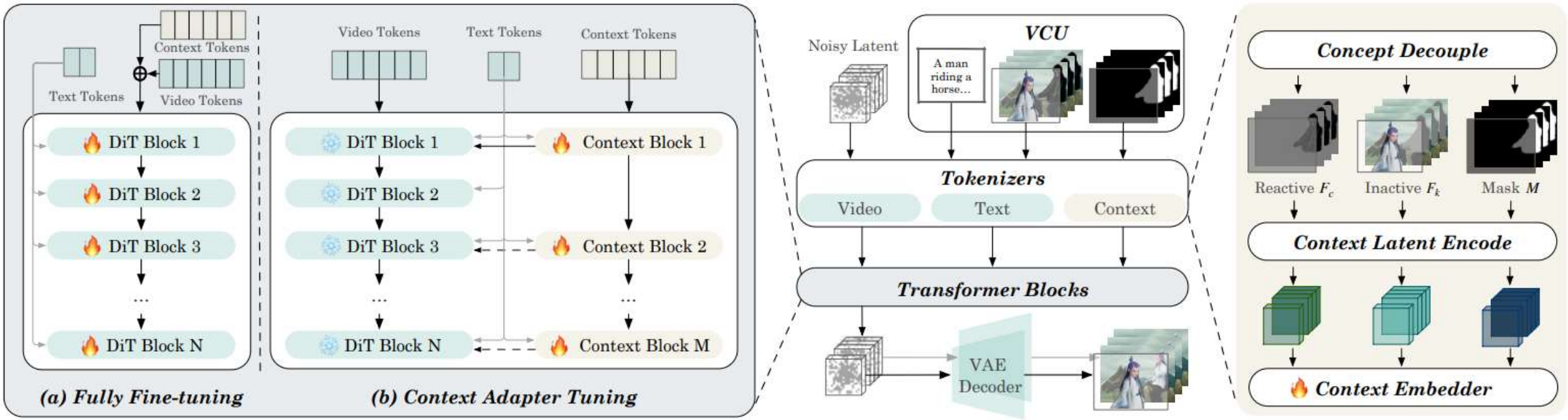
https://www.alibabacloud.com/en/press-room/alibaba-introduces-open-source-model-for-video

VACE



https://arxiv.org/pdf/2503.07598

VACE



https://arxiv.org/pdf/2503.07598

VACE

https://arxiv.org/pdf/2503.07598

Sora2 & Veo3.1: Omni Model / “WORLD MODEL”

From Silent Video to Native Audiovisual Generation

- The target is synchronized audiovisual events.
- Dialogue, ambience, and sound effects must cohere with motion.
- New bottlenecks: lip-sync, event-sound causality, and cross-shot continuity.

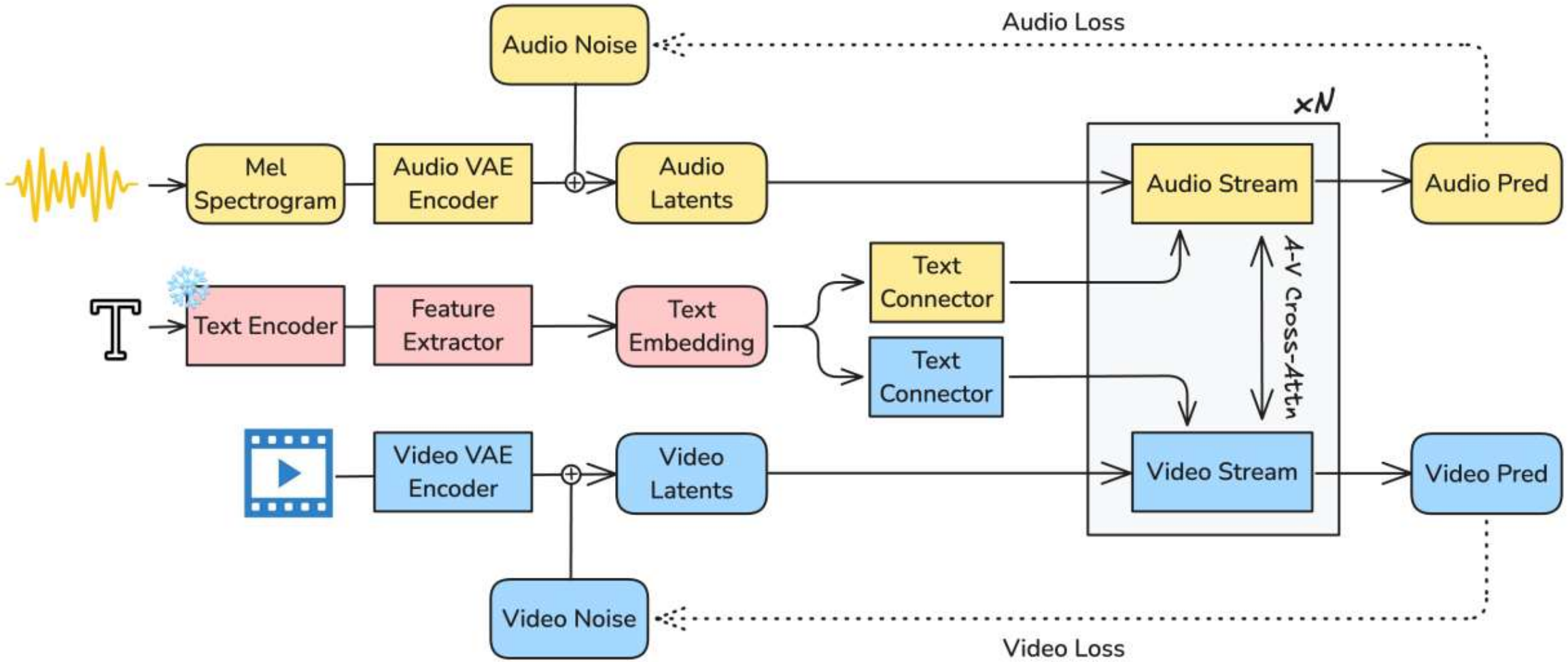
- Sora 2 improves physics, realism, synchronized audio, and steerability.
- Prompts now combine subject, camera, pacing, and audio intent.
- Storyboard, remix, stitch, extend, and characters define the workflow.

https://sora.chatgpt.com/explore

- Veo 3.1 adds richer native audio and stronger prompt adherence.
- Reference images, first/last frames, and extension expose creative control.
- A strong Veo prompt reads like a director's brief.

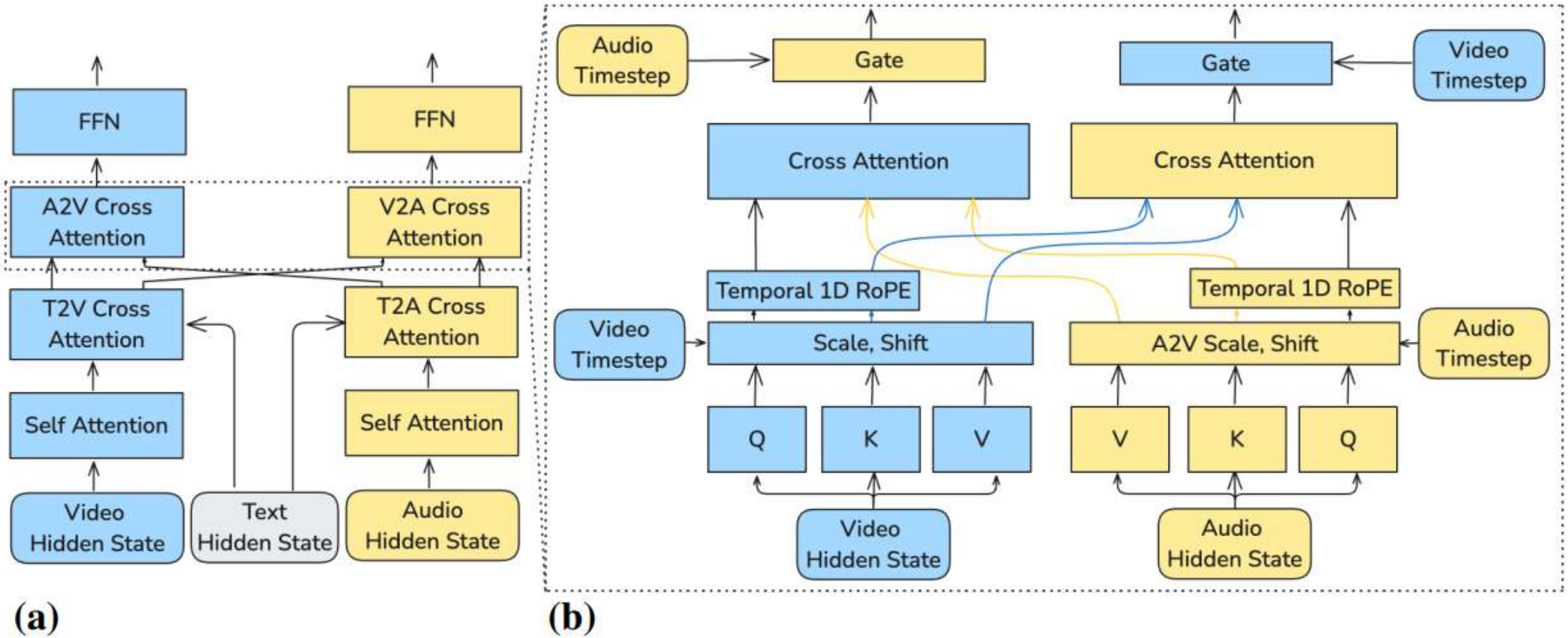
https://blog.google/innovation-and-ai/products/veo-updates-flow/?utm_source=chatgpt.com

LTX



https://arxiv.org/pdf/2601.03233

LTX



https://arxiv.org/pdf/2601.03233

LTX

Figure 3: **Visualization of AV cross-attention maps.** The maps are averaged across attention heads and the model layers; V2A and A2V maps correspond to the first and last 1/3 of inference steps, respectively. Red vertical lines on the audio waveform mark the timestamps of the displayed frames. The visualization demonstrates the model's ability to spatially track a moving vehicle, dynamically shift attention from one speaker to another and then to both simultaneously, and focus on the lip region during close-up speech.

https://arxiv.org/pdf/2601.03233

Seedance

https://seed.bytedance.com/zh/blog/official-launch-of-seedance-2-0