Carnegie Mellon University

Lecture 7: Text-to-Image Generation & SOTA Models

Yutong (Kelly) He

10-799 Diffusion & Flow Matching, Feb 5th, 2026

Modal aws

AWS Credits are finally here 🥳

Dhruv from AWS is here to show us some quick tutorials

Quiz time!

10 minutes

Closed-book

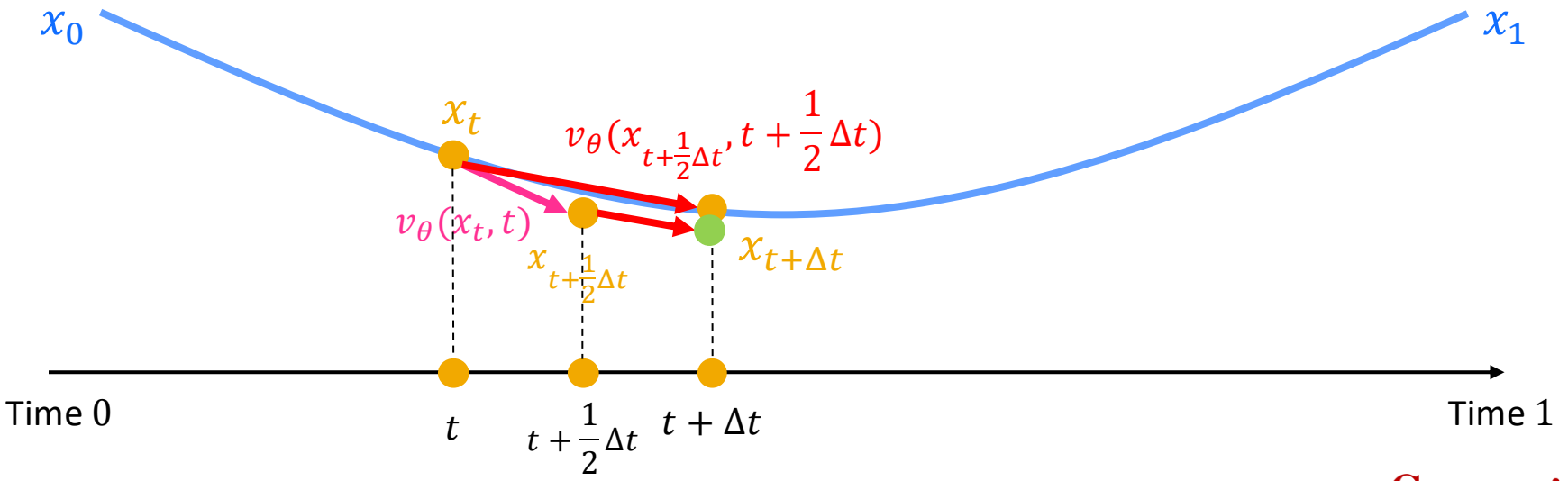
Pen & Paper

If you don’t want to stay for the lecture, feel free to leave after submitting your quiz!

Housekeeping Announcements

- Homework 3 is out! https://kellyyutonghe.github.io/10799S26/homework/
    - Due date: 2/15 Sun, Late Due date: 2/17 Mon
    - Start early and finish early if possible! This way you'll have more time for HW 4
    - Poster PDF submission 2/25 Wed
    - Poster Session 2/26 Thur
    - No class on 2/24 Tue
- Will have a poll about AWS credit on Discord soon

Correction – Midpoint solver



Previously we learned about the design space of diffusion models

The design space of diffusion models

Training

- Prefixed noise schedule
- Training noise sampling schedule
- Loss weighting w.r.t. time

Model

- Reparameterization
- Input/Output scaling
- How to do time conditioning

Sampling

- Solver
- Sampling time noise schedule
- Number of time steps

Last time we learned how to turn an unconditional diffusion into a conditional one!

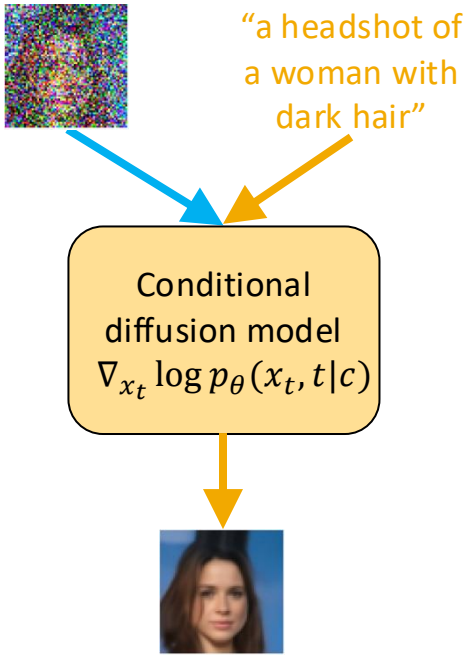
Training-based methods:

- Classifier guidance diffusion
- Classifier free guidance (if you don't already have a conditional model)

Training-free methods:

- DPS
- MPGD
- SDEdit
- Classifier free guidance (if you already have a conditional model)

Today we will be looking at this one specific type of condition – Text!

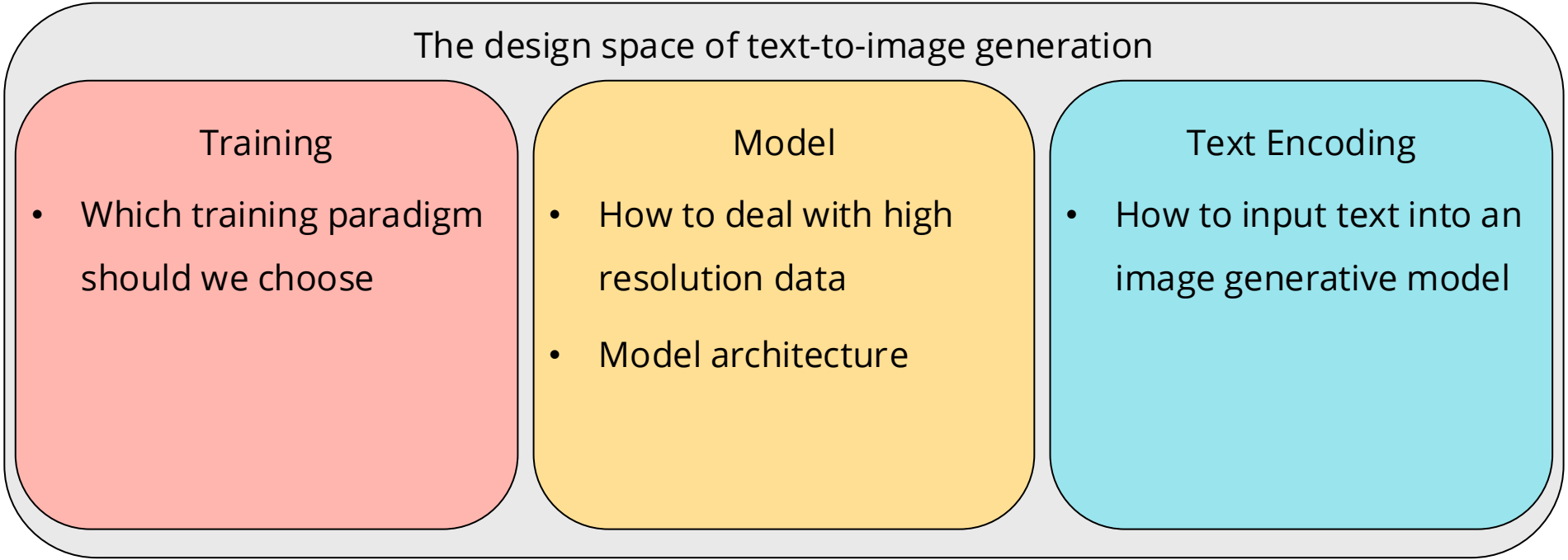


Carnegie Mellon University

The design space of text-to-image generation

The design space of text-to-image generation

The design space of text-to-image generation



Images are super super high dimensional data

In HW we are dealing with 64x64 images

- 64x64x3=12,288 => 18M model parameters

Now what if we need to train models for 256x256 images

- 256x256x3=196,608 => probably in the hundreds of M's parameters

What about 1080p resolution?

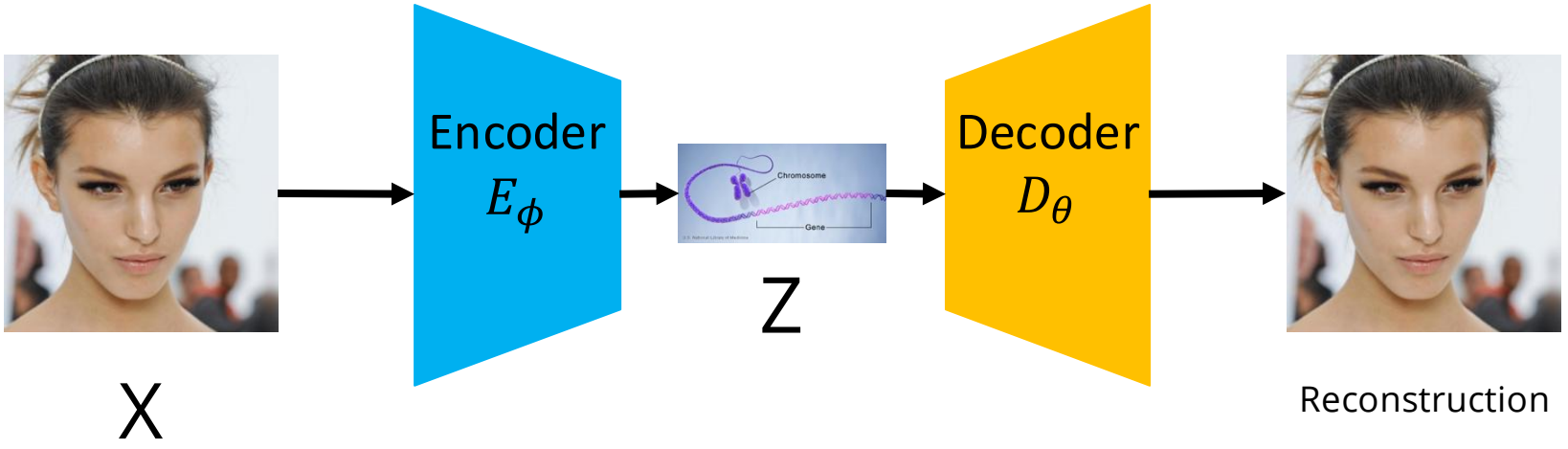
- 1920×1080x3=**6,220,800** => probably in the billions of parameters

And what about 4k Images?

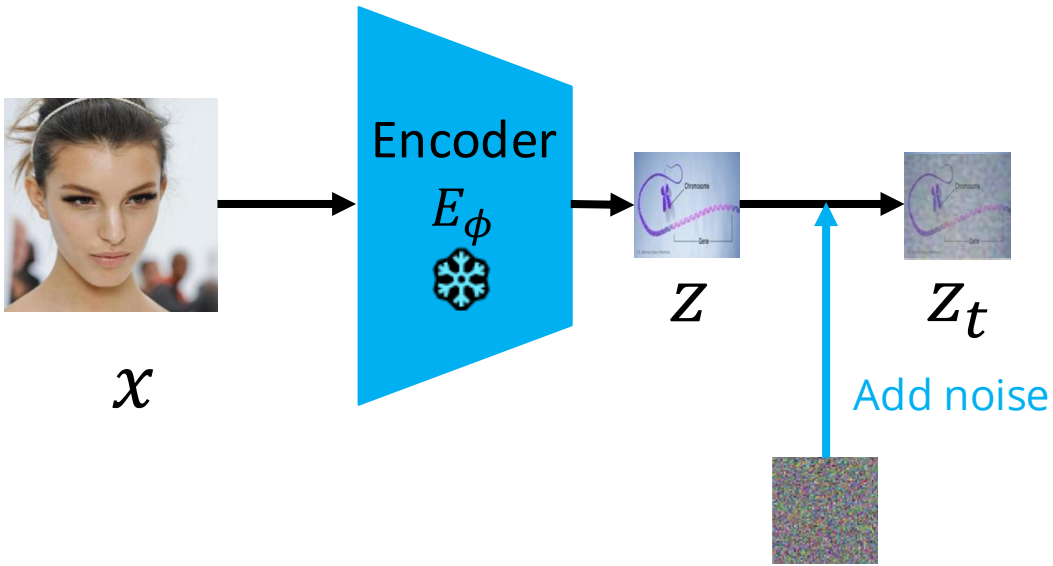
- 3840x2560x3=**29,491,200** => probably dozens of billions of parameters

How to deal with high resolution data?

Idea: Compression via an autoencoder



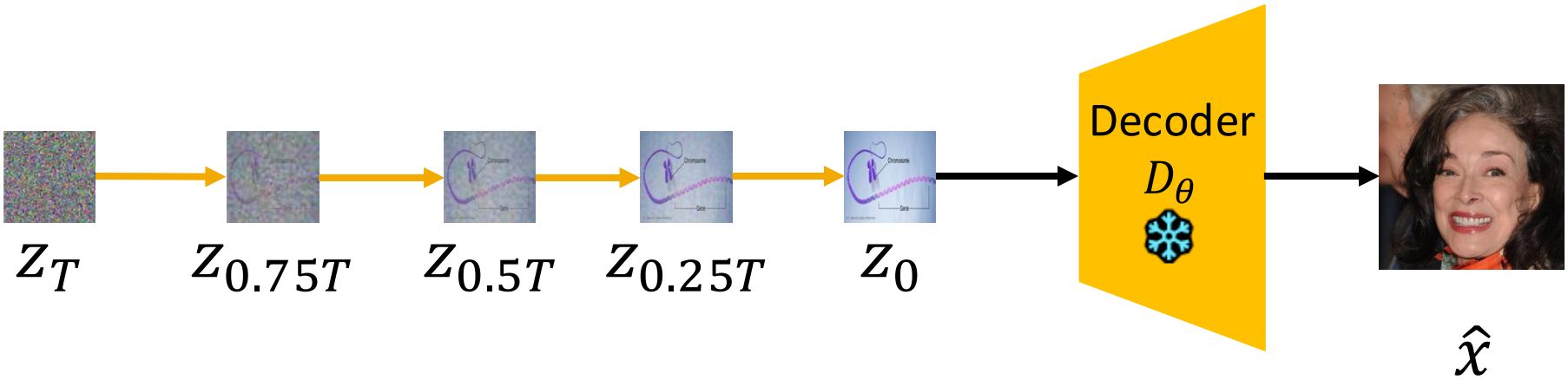
Latent diffusion models (forward process)



Carnegie Mellon University

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

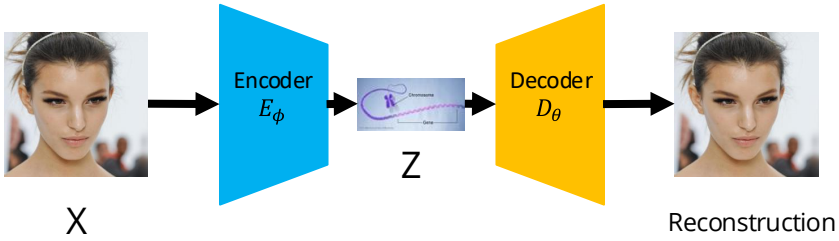
Latent diffusion models (reverse process)



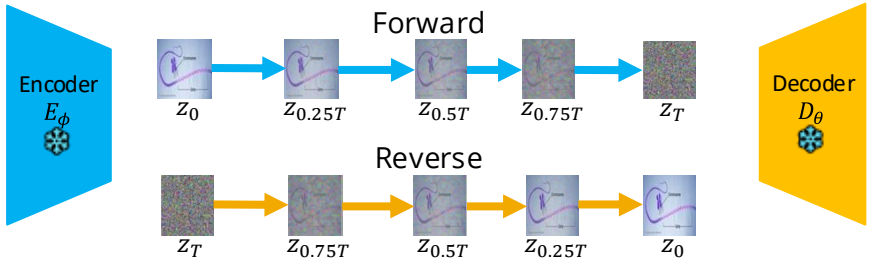
Rombach et al. "High-Resolution Image Synthesis with Latent Diffusion Models". CVPR 2022. https://arxiv.org/pdf/2112.10752

Latent diffusion models

1. Pre-train an autoencoder on your data

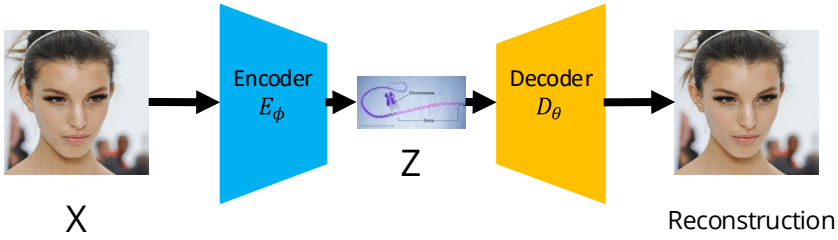


2. Freeze your pre-trained autoencoder, and then train a diffusion model in the latent space of your autoencoder

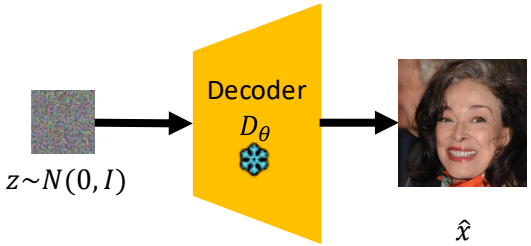


Carnegie Mellon University

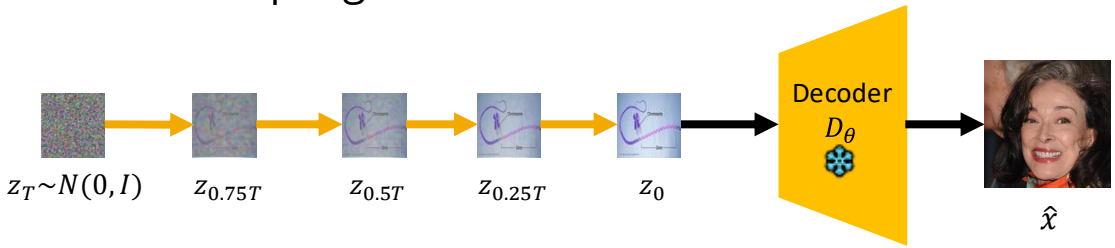
Vanilla VAE v.s. Latent Diffusion



- Vanilla VAE sampling:



- Latent diffusion sampling:



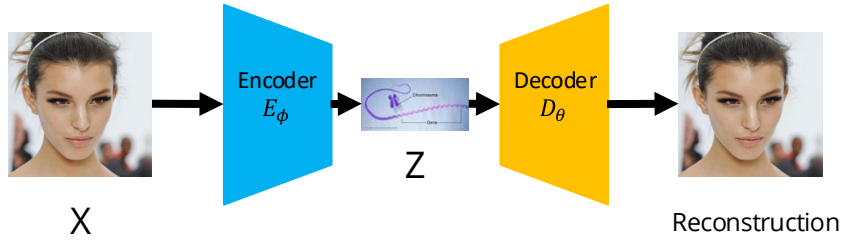
Carnegie Mellon University

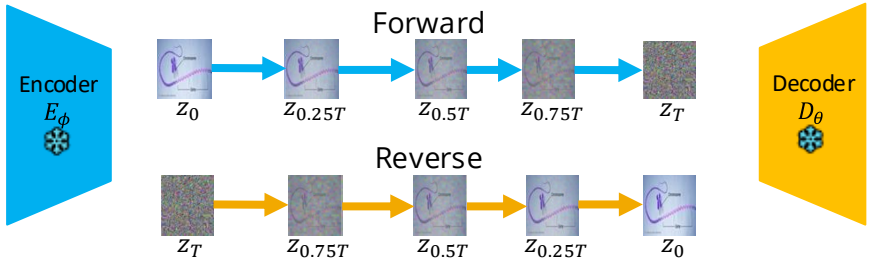
Latent diffusion models

1. Pre-train an autoencoder on your data

Which autoencoder?

2. Freeze your pre-trained autoencoder, and then train a diffusion model in the latent space of your autoencoder





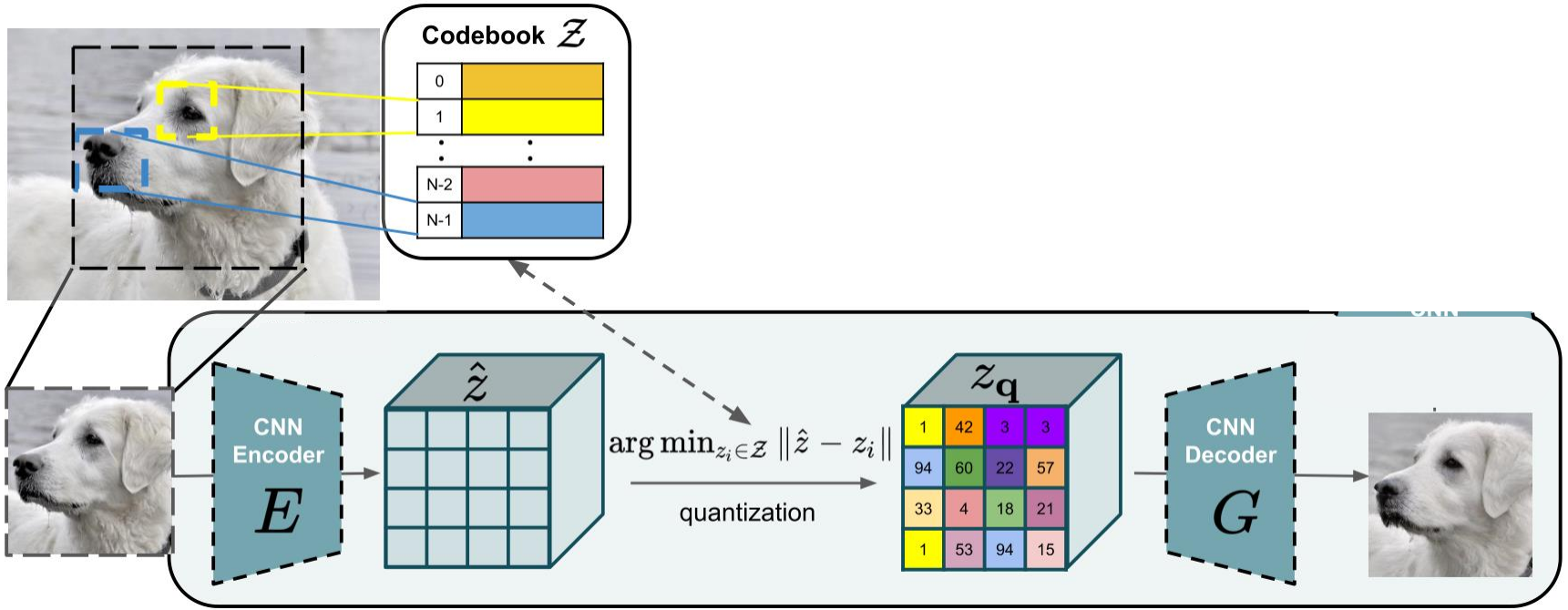
Carnegie Mellon University

Attempt 1: A normal VAE

Vanilla VAE has a lot of problems:

- The “semantics” are vague
- Can have the “average face” problem – generations are blurry and overly smoothed out
    - Sometimes there’s even a posterior collapse problem: the decoder just completely ignores the sampled latent and just do its own thing
- Difficult to be compatible with transformers

Attempt 2: Vector-Quantized VAE (VQ-VAE)

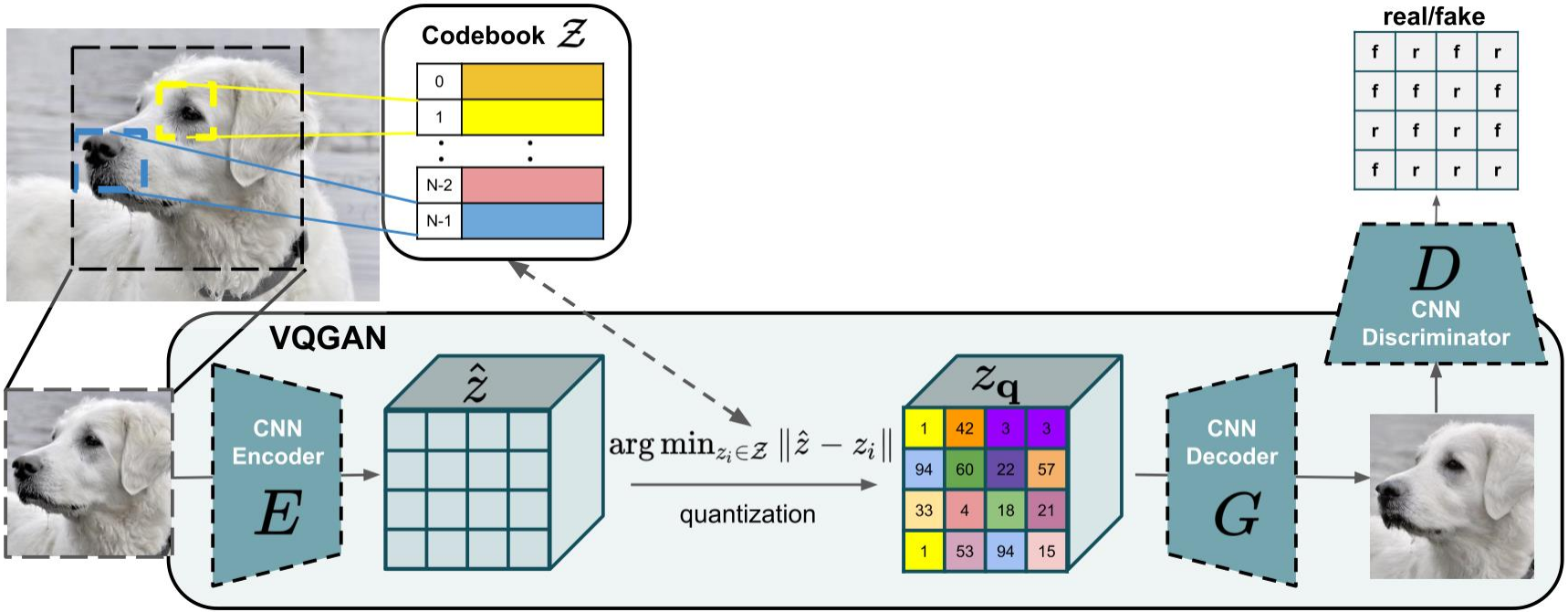


van den Oord et al. "Neural Discrete Representation Learning". NeurIPS 2017. https://arxiv.org/pdf/1711.00937

Esser et al. "Taming Transformers for High-Resolution Image Synthesis". CVPR 2021. https://arxiv.org/pdf/2012.09841

Carnegie Mellon University

Attempt 2.5: Vector-Quantized GAN (VQGAN)

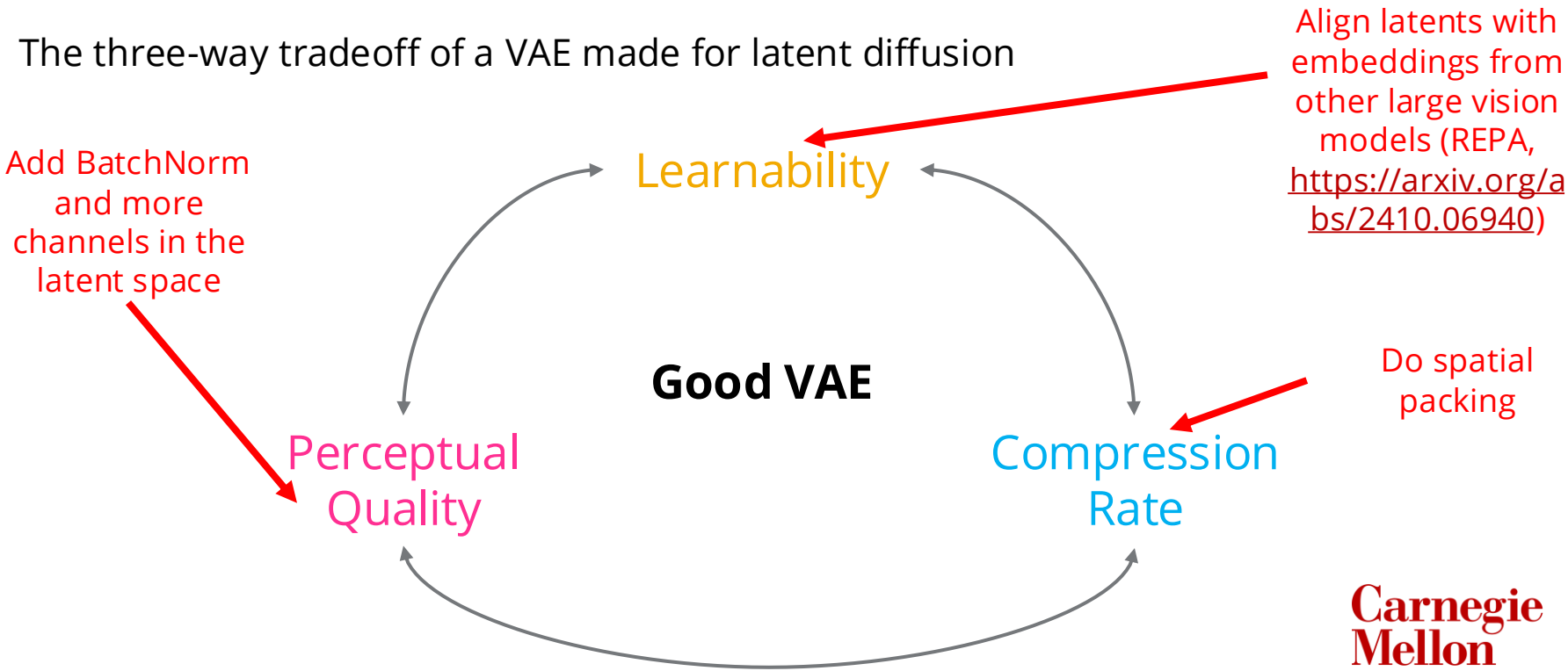


Carnegie Mellon University

Esser et al. "Taming Transformers for High-Resolution Image Synthesis". CVPR 2021. https://arxiv.org/pdf/2012.09841

Attempt 3: Flux 2, making continuous VAE work

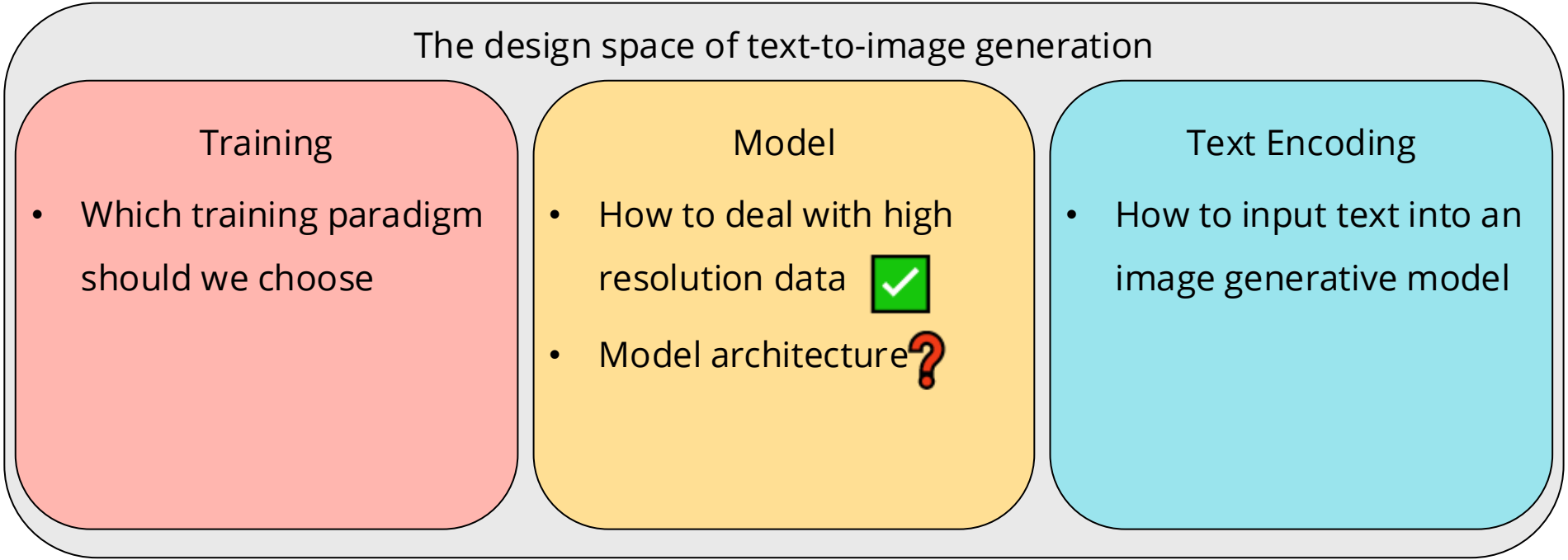
The three-way tradeoff of a VAE made for latent diffusion



Black Forest Labs. https://bfl.ai/research/representation-comparison

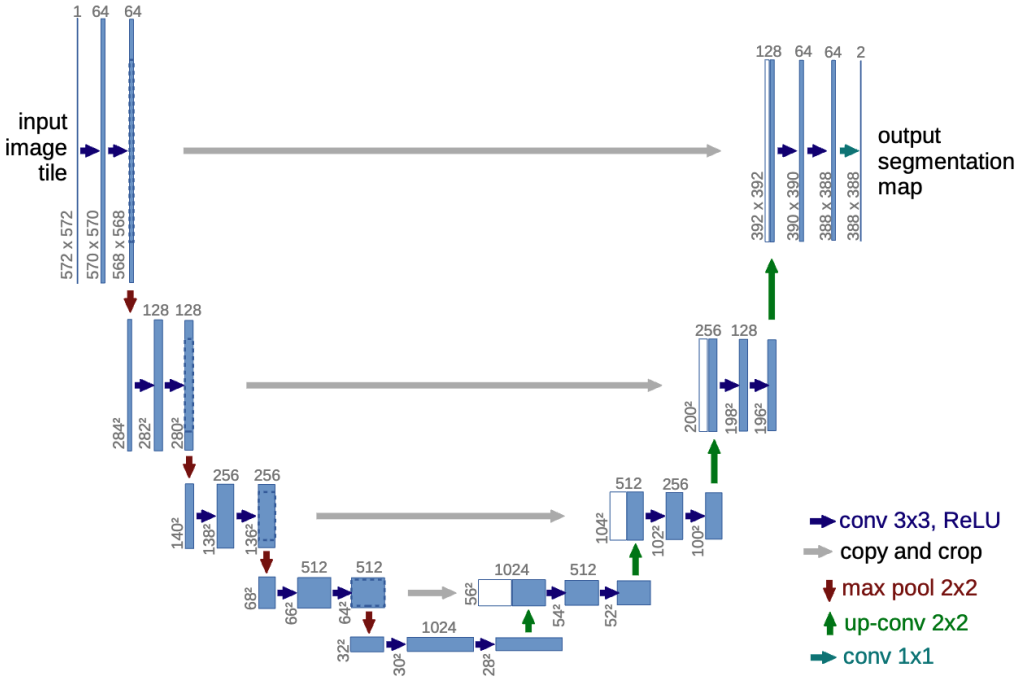
Carnegie Mellon University

The design space of text-to-image generation



What kind of model architecture should we use?

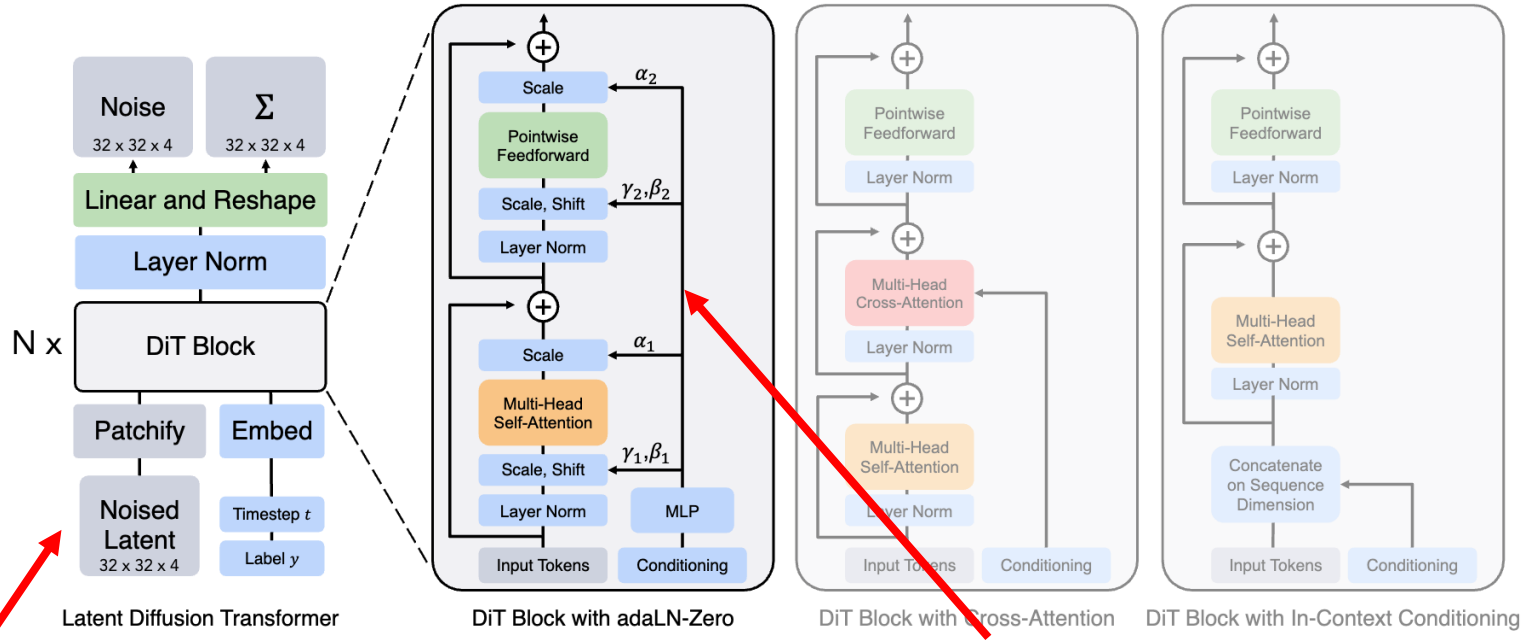
Attempt 1: Ofc we can use our good old



Net

Ronneberger et al. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2020. https://arxiv.org/abs/1505.04597

Attempt 2: Actually we can use a transformer (DiT) too



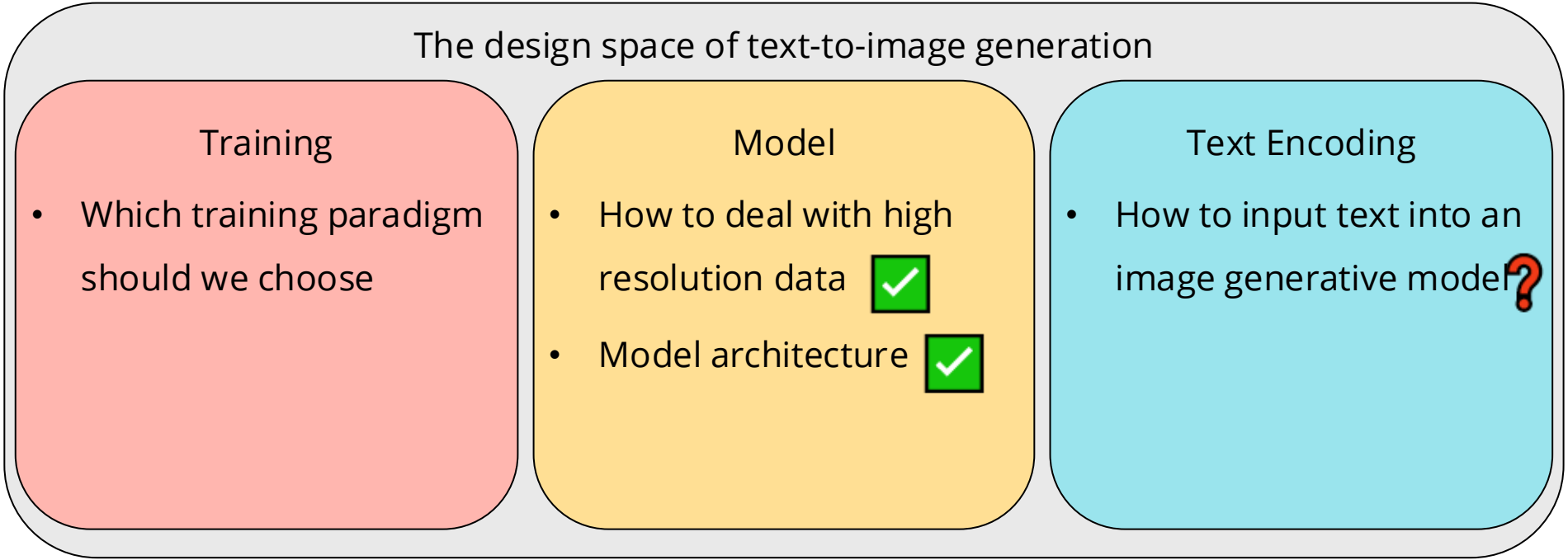
2D input => 2D positional embeddings

Time step conditioning injected into every layer

Carnegie Mellon University

Peebles & Xie. "Scalable Diffusion Models with Transformers". ICCV 2023. https://arxiv.org/pdf/2212.09748

The design space of text-to-image generation



How to input text into an image generative model?

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

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 1: Learn text features with images

Positive Examples







Negative Examples

Carnegie Mellon University

Positive examples

“dog”

“a cute dog”

“a Cavalier King Charles Spaniel”

Carnegie Mellon

Same thing, different “augmentations” => Positive examples

Negative Examples

Negative Examples

“dog”

Positive Examples

“cat”

“puppy plushie”

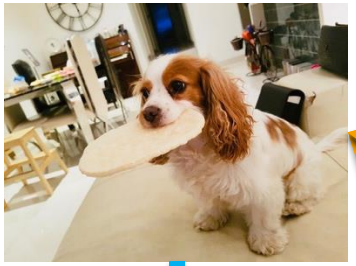
“a CMU PhD student”

“a CMU Prof”

Carnegie Mellon University

Negative Examples

Negative Examples



"dog"

"cat"

"puppy plushie"

"a CMU PhD student"

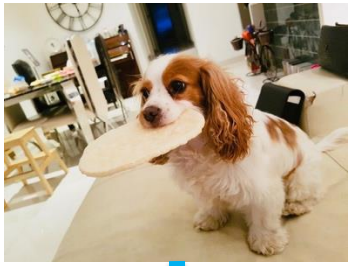
"a CMU Prof"

Positive Examples

Carnegie Mellon University

Negative Examples

Negative Examples



“dog”

Positive Examples

“cat”

“puppy plushie”

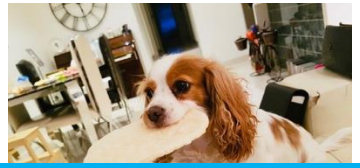
“a CMU PhD student”

“a CMU Prof”

Carnegie Mellon University

Negative Examples

Negative Examples



Do this for 400 Million image-text pairs

“dog”

Positive Examples

“cat”

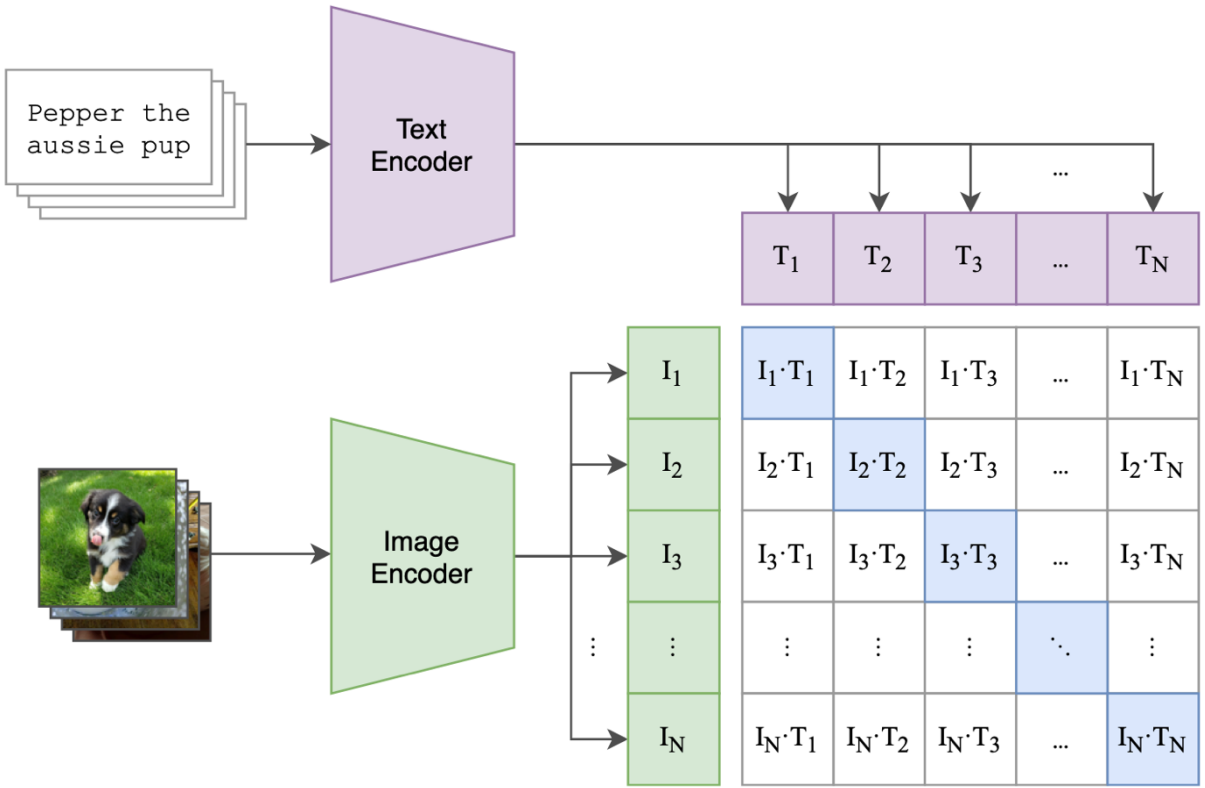
“puppy plushie”

“a CMU PhD student”

“a CMU Prof”

Carnegie Mellon University

Attempt 1: CLIP -- Contrastive Language-Image Pretraining



Carnegie Mellon University

Radford et al. "Learning Transferable Visual Models From Natural Language Supervision". https://arxiv.org/pdf/2103.00020

CLIP is good but…

It cannot tell these apart!

(a) some plants surrounding a lightbulb

(b) a lightbulb surrounding some plants

Carnegie Mellon University

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…

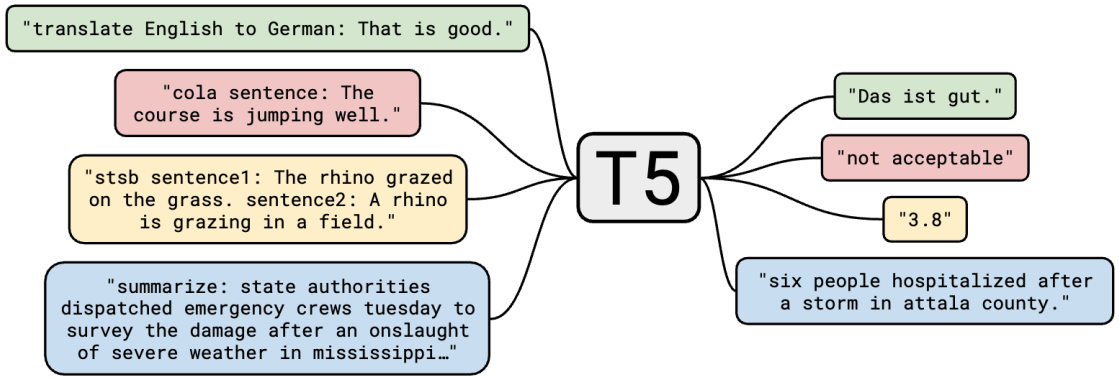
- It cannot handle spatial relation ships well
- It cannot handle negations well
- It cannot handle counts well
- The length limit is 77 tokens
- Sometimes ignores some details

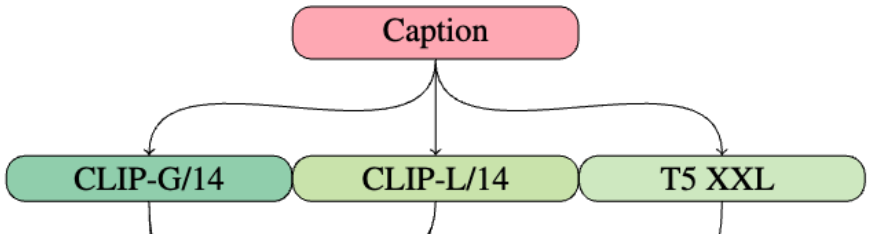
CLIP is not a very good text encoder!

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



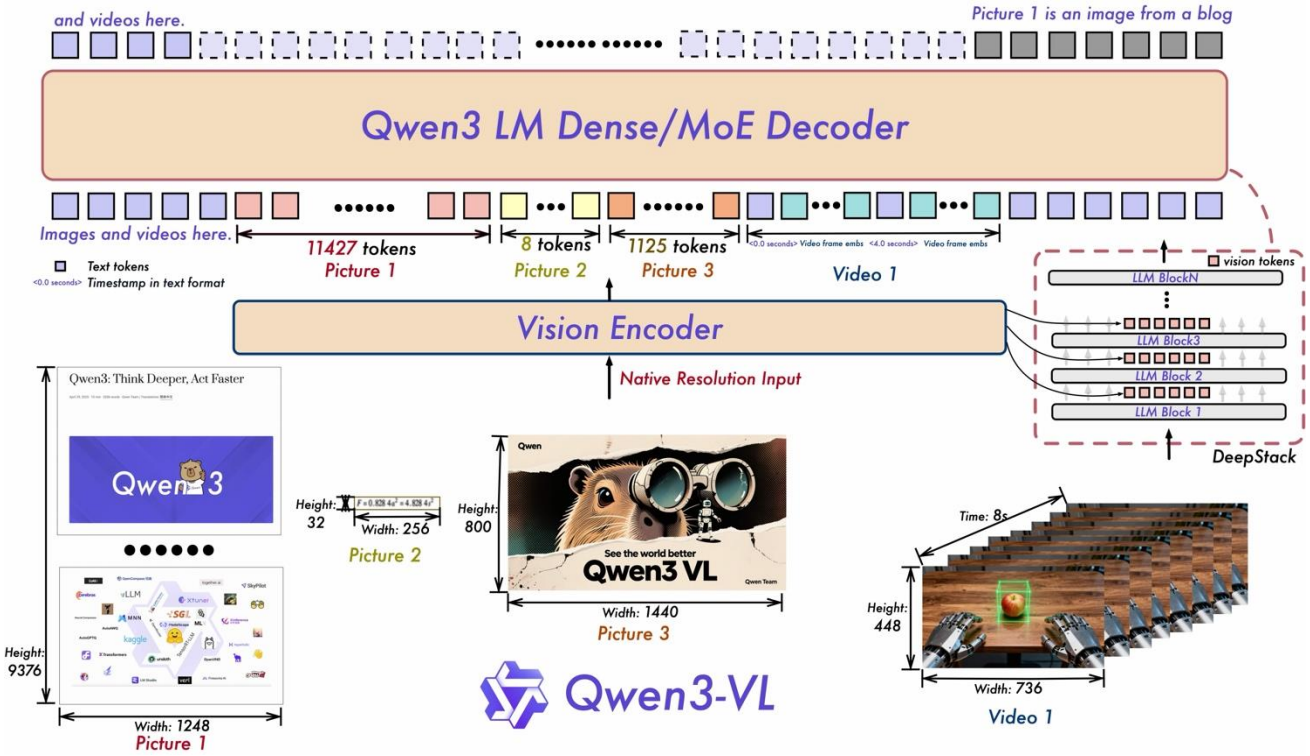


Raffel et al. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". JMLR 2020. https://arxiv.org/pdf/1910.10683

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

Carnegie Mellon University

Attempt 3: Why not just use an LLM/VLM/MLLM lol



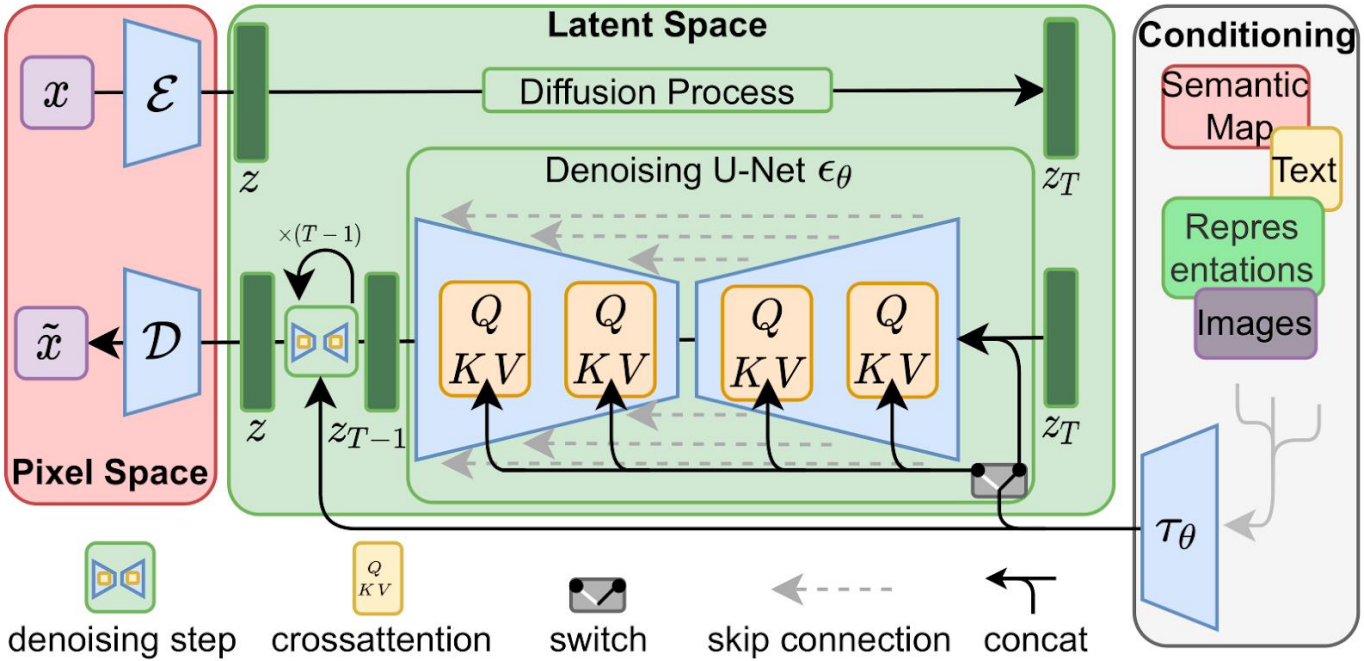
Qwen Team. "Qwen3-VL Technical Report". https://arxiv.org/pdf/2511.21631

Carnegie Mellon University

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 ❓

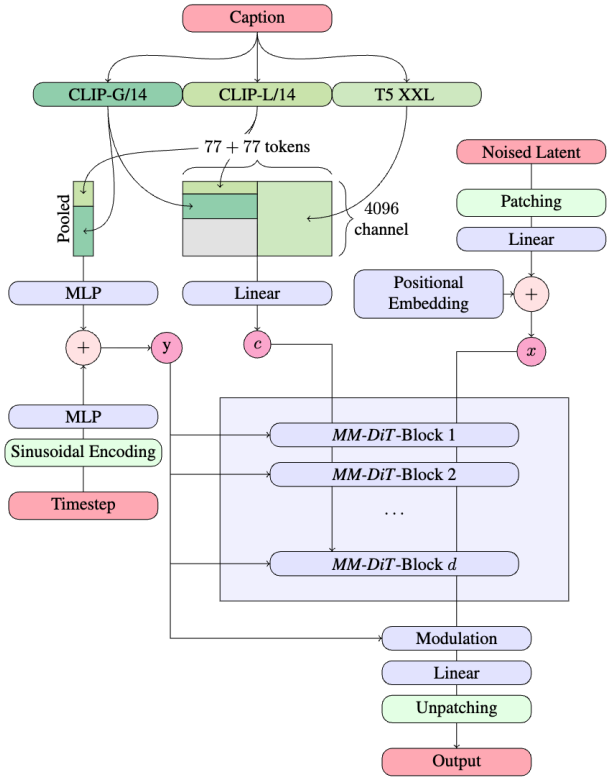
Attempt 1: Cross Attention



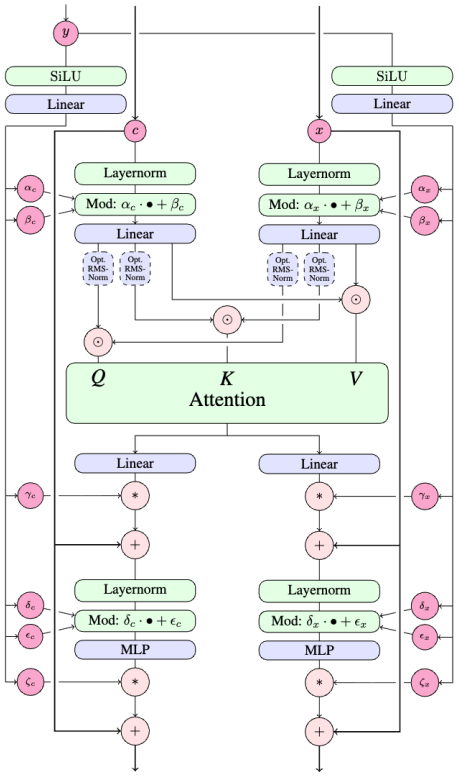
Carnegie Mellon University

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.

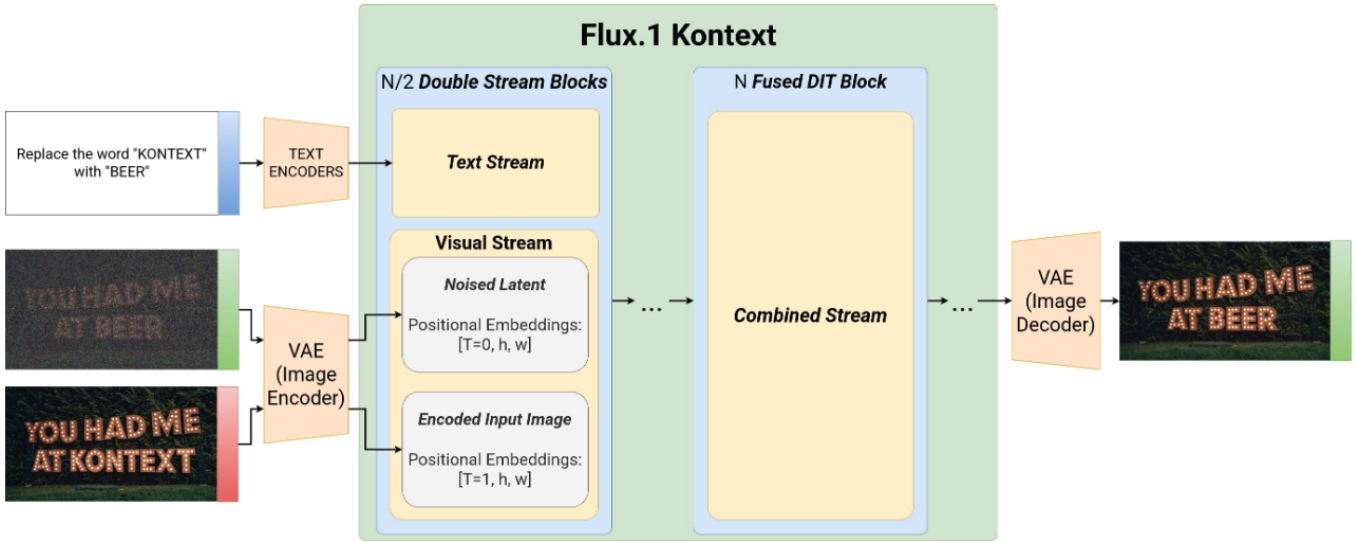


(b) One *MM-DiT* block

Carnegie Mellon University

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

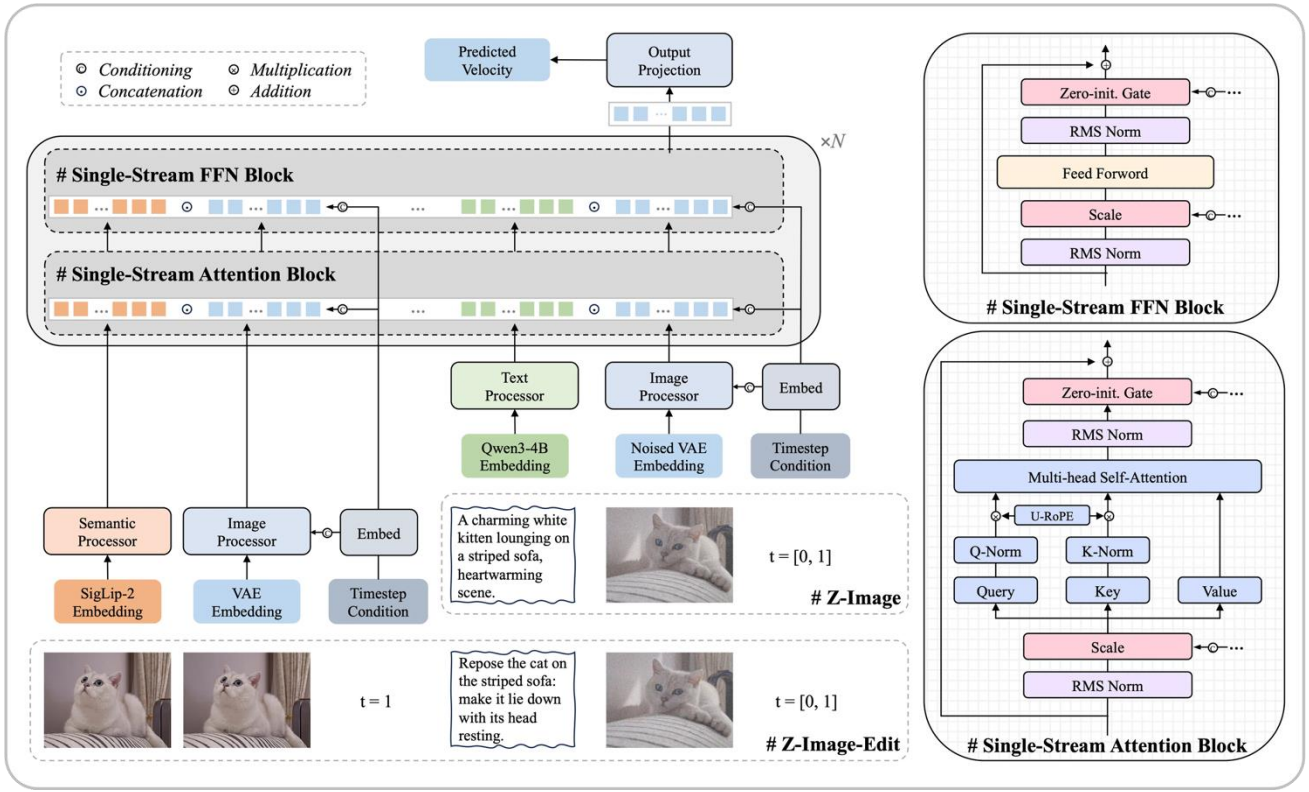
Attempt 2: Double stream -> merged stream MM-DiT



Carnegie Mellon University

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



Carnegie Mellon University

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”

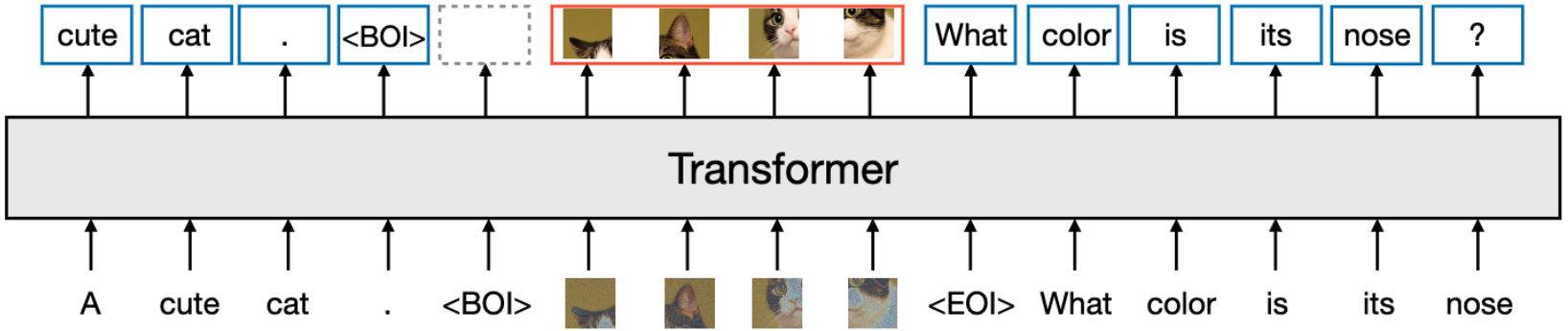
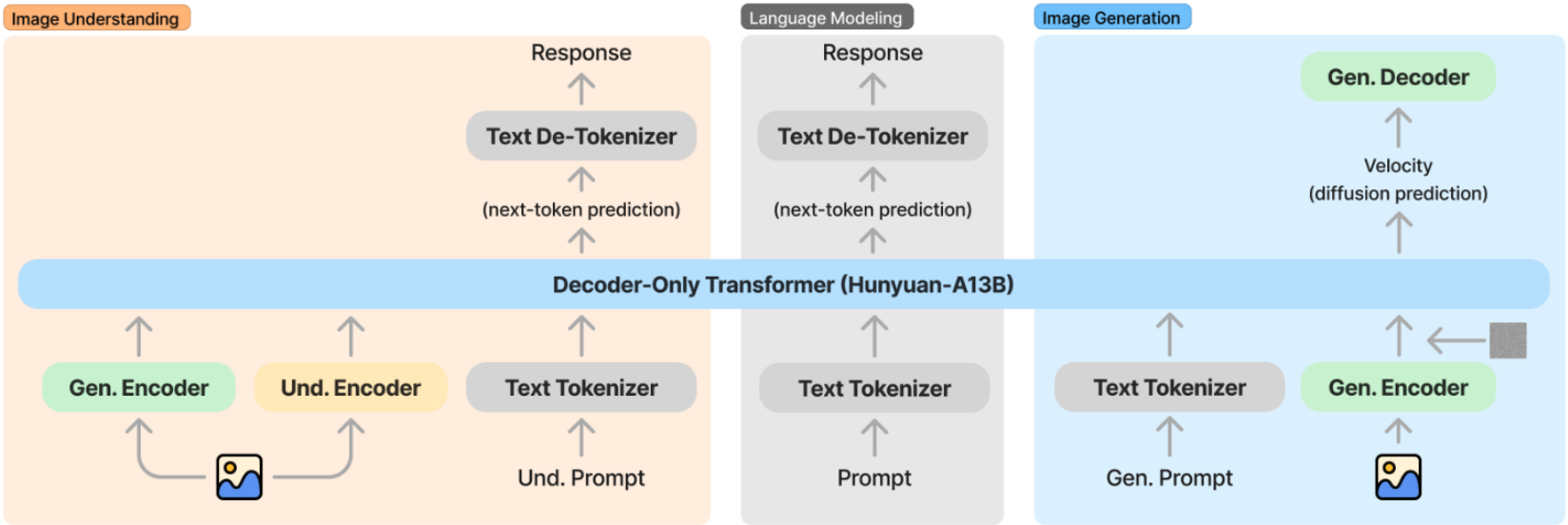


Figure 1: A high-level illustration of Transfusion. A single transformer perceives, processes, and produces data of every modality. Discrete (text) tokens are processed autoregressively and trained on the next token prediction objective. Continuous (image) vectors are processed together in parallel and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

Carnegie Mellon University

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

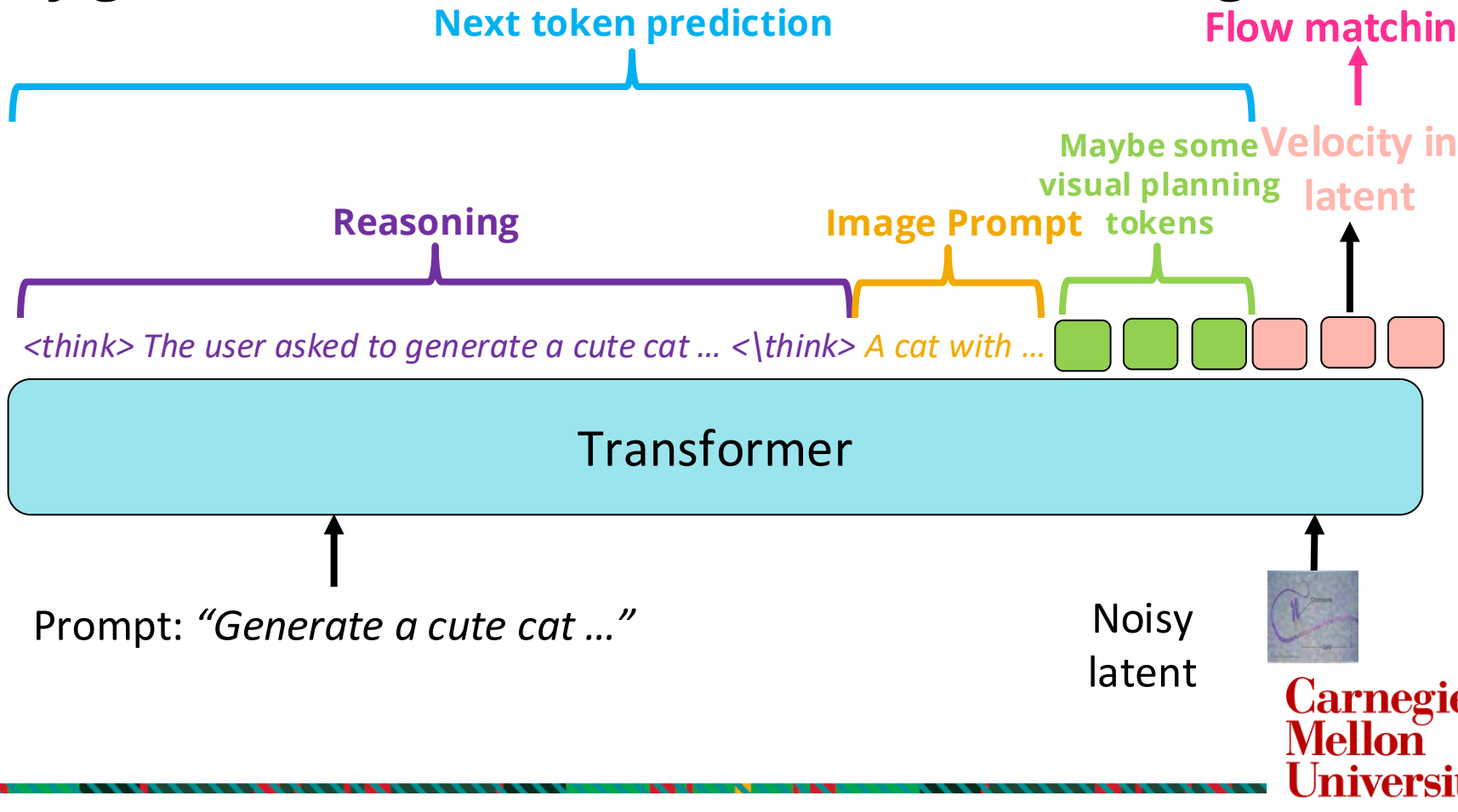
Attempt 3.5: “Native multimodal model”



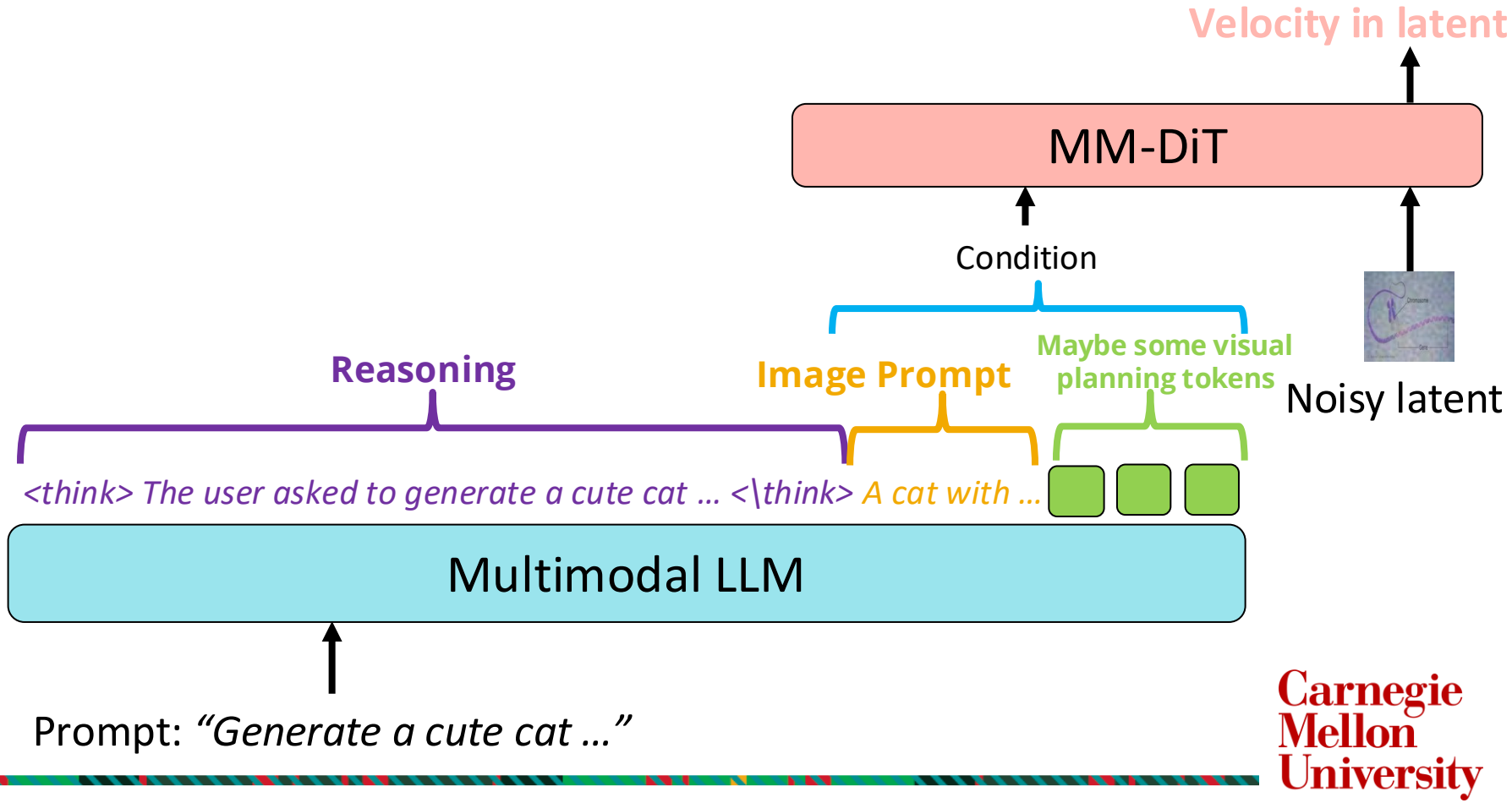
Carnegie Mellon University

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

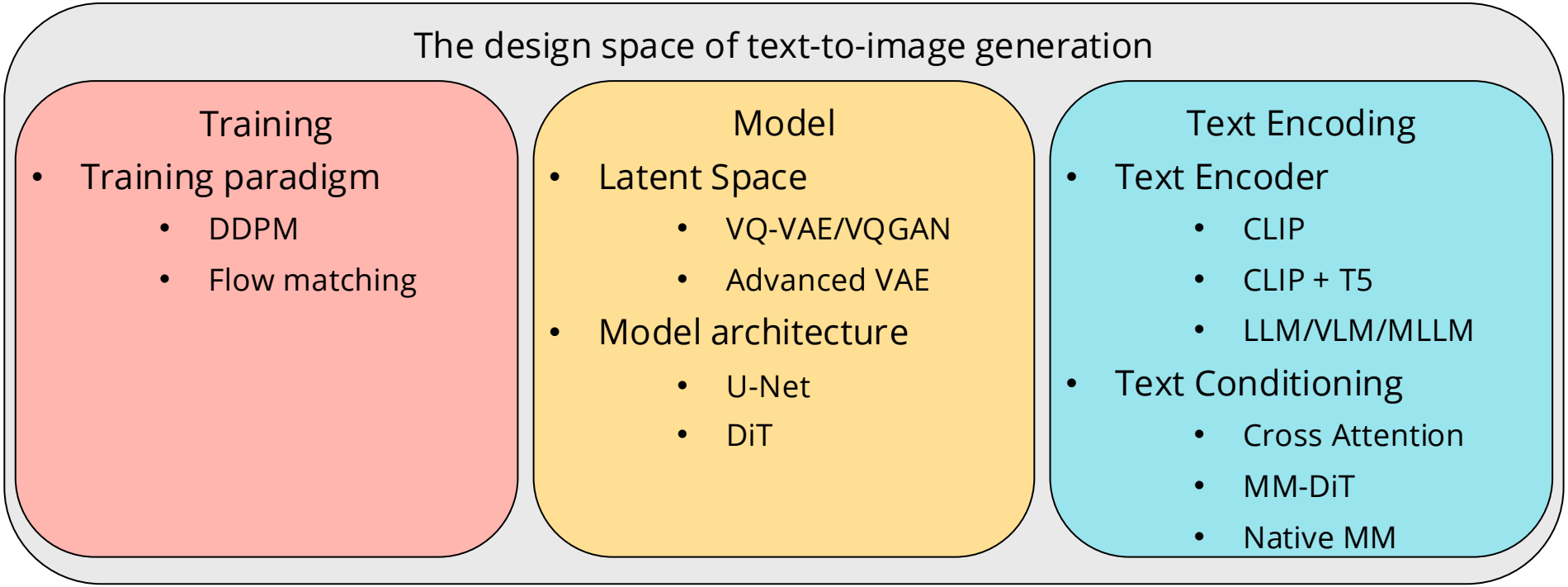
My guess on how Nano Banana & GPT-4o Image work



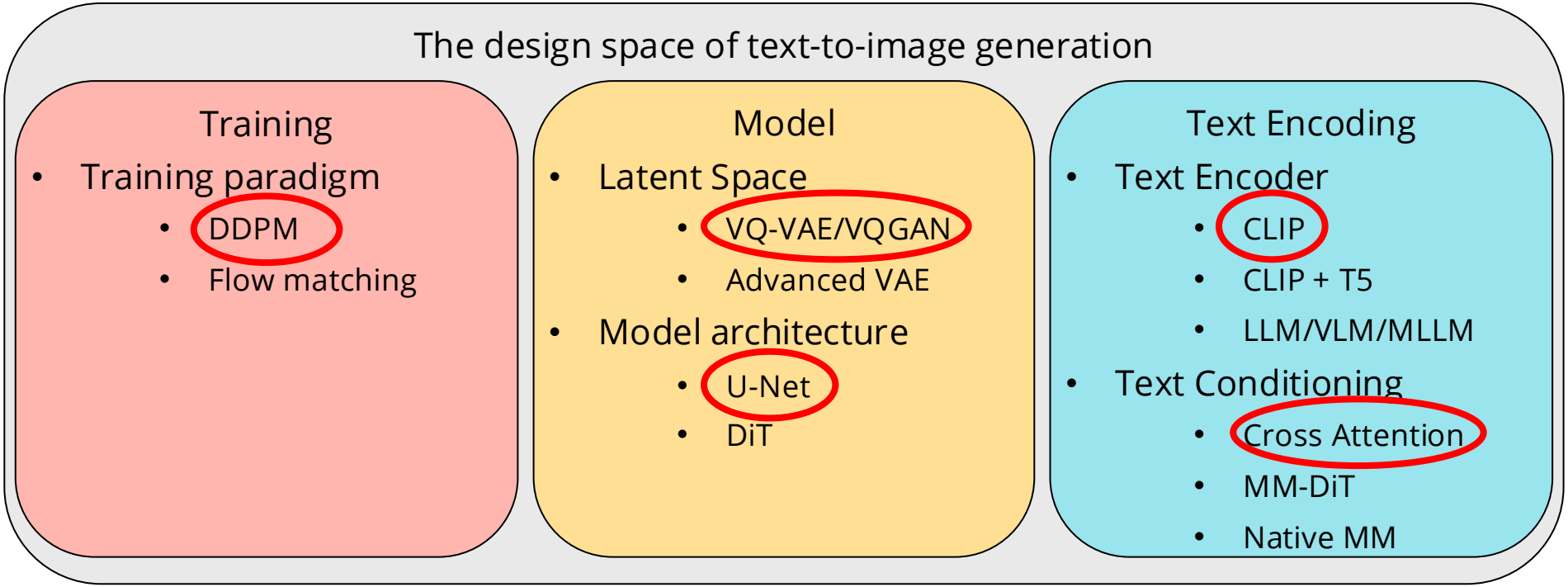
My guess on how Nano Banana & GPT-4o Image work



The design space of text-to-image generation

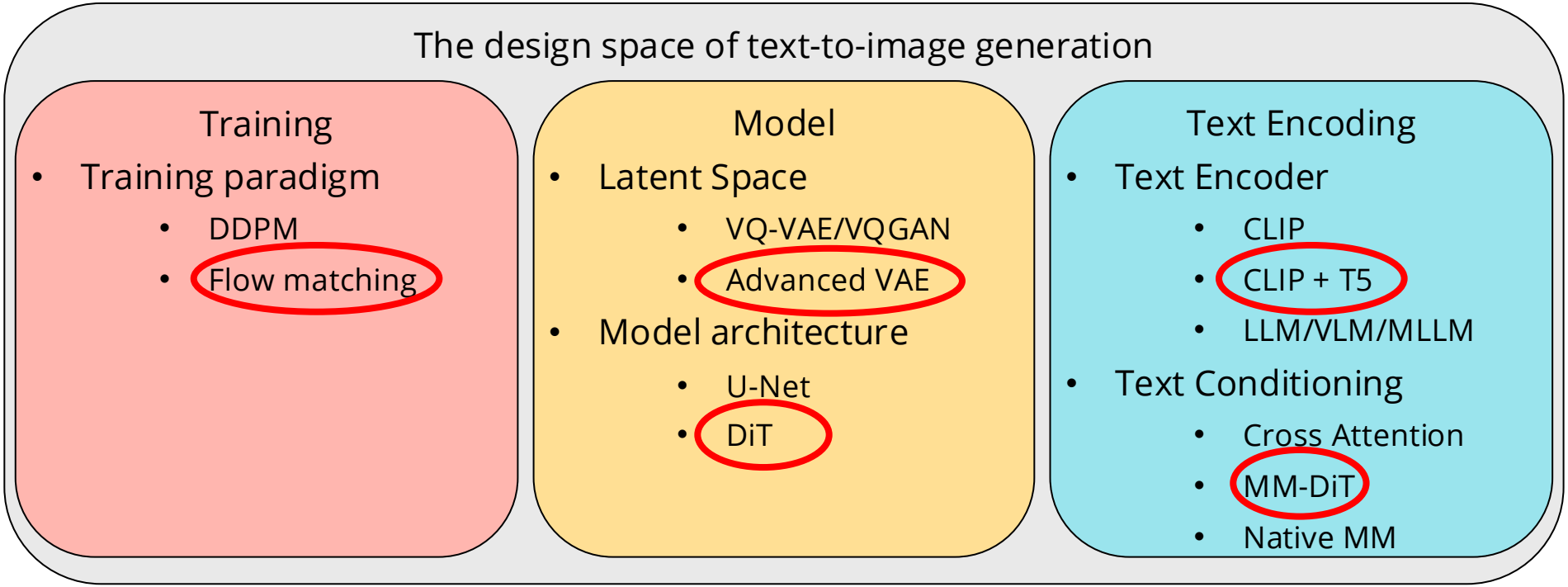


The design space of text-to-image generation



This is Stable Diffusion 1 & 2!

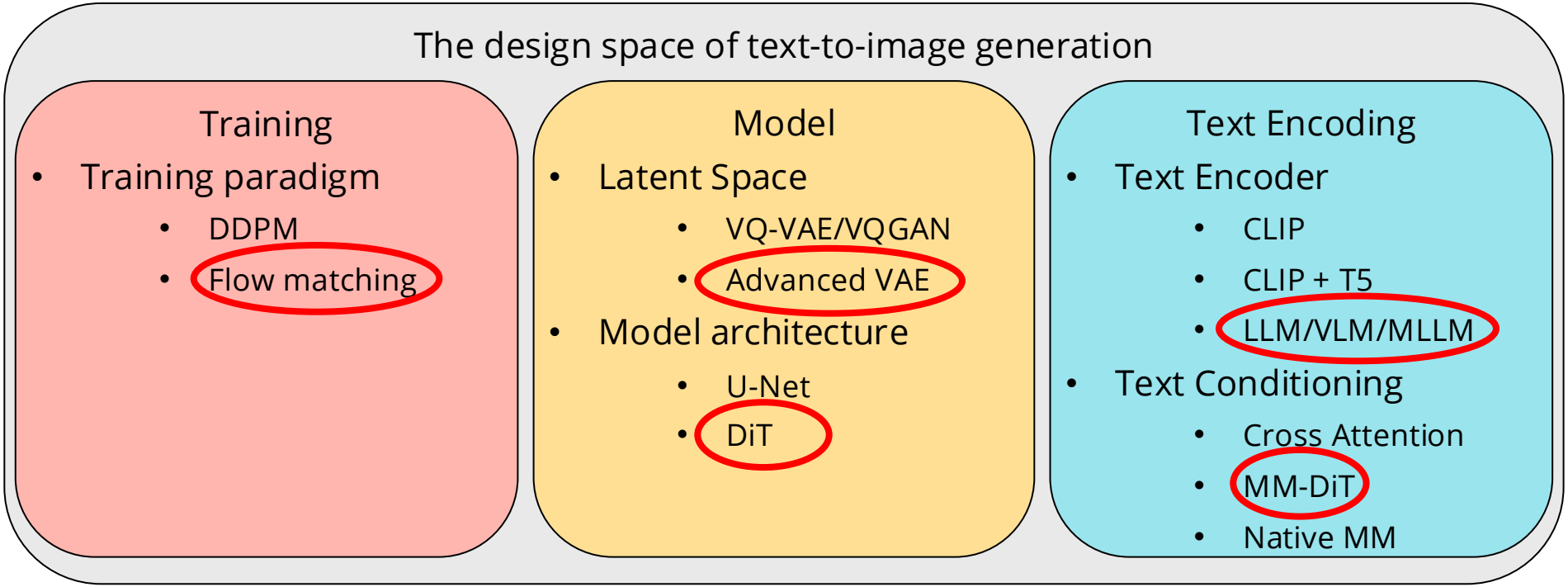
The design space of text-to-image generation



This is Stable Diffusion 3 & Flux 1!

Carnegie Mellon University

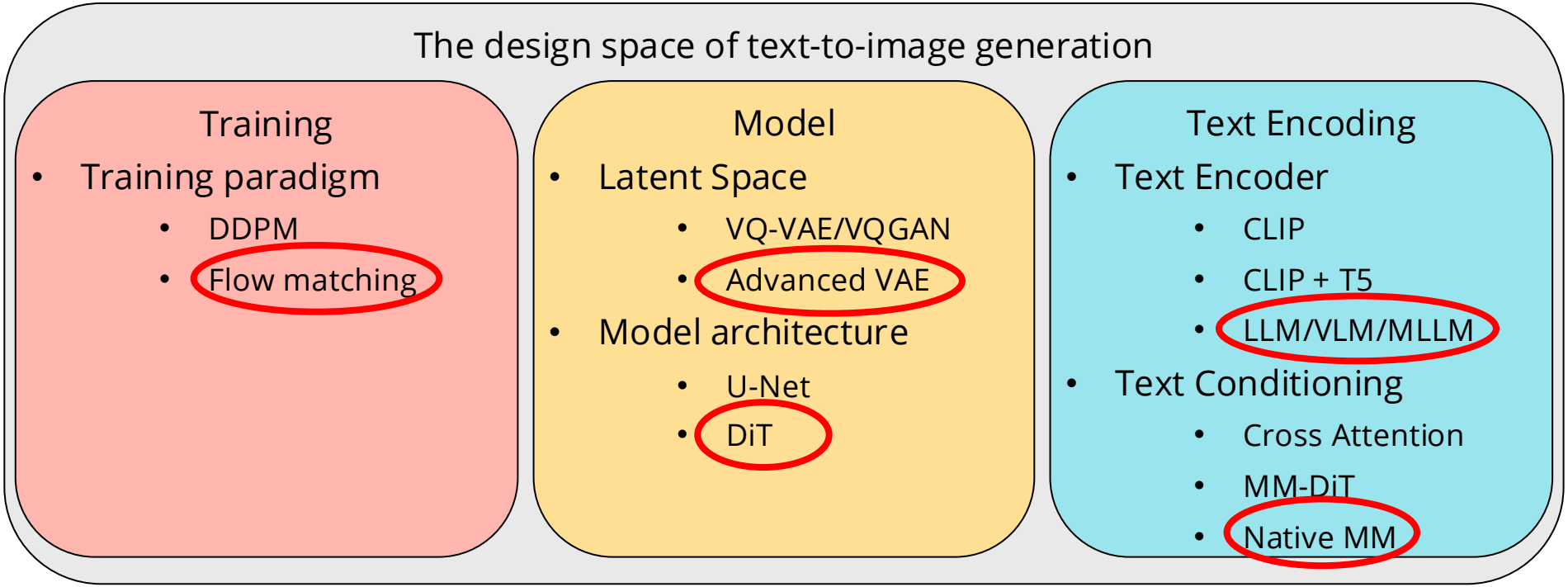
The design space of text-to-image generation



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

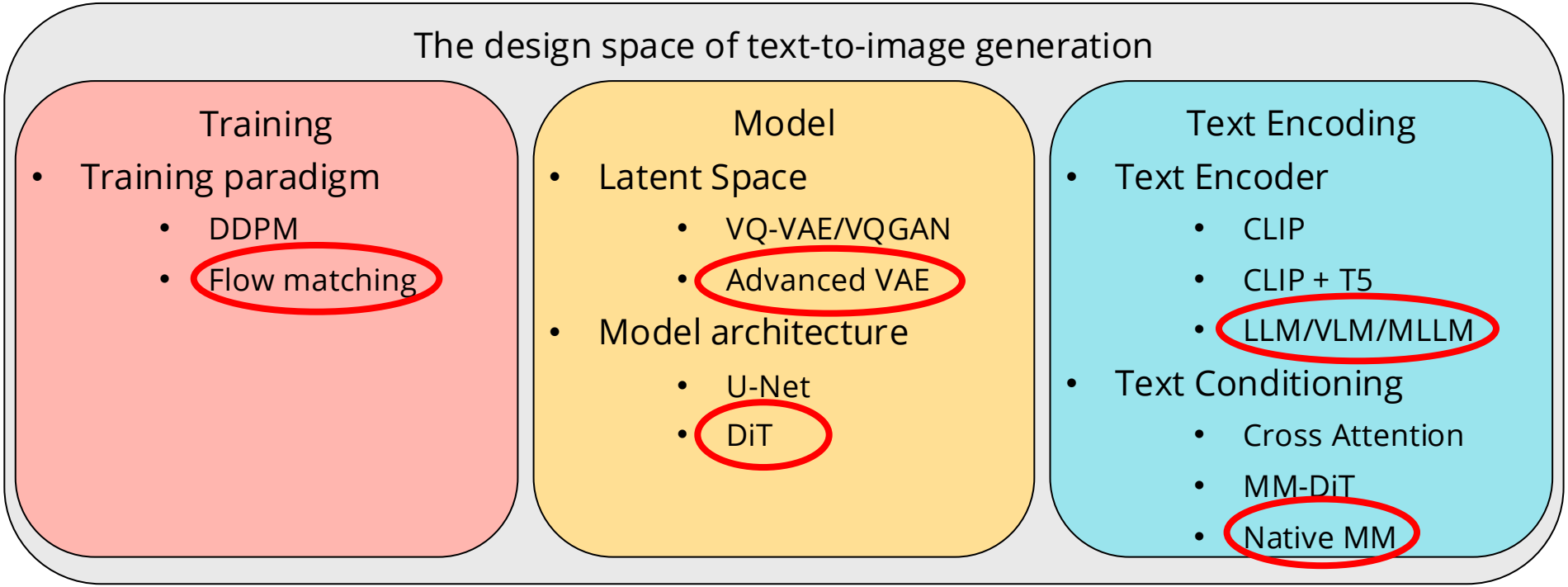
Carnegie Mellon University

The design space of text-to-image generation



This is Transfusion, Hunyuan 3.0, etc!

The design space of text-to-image generation



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

Carnegie Mellon University

We are now in the realm of the state-of-the-arts

- **2/3 (Tue):** How to use diffusion/flow models for robotics, control & decision making (**Max Simchowitz**, MLD Prof.) ✅
- **2/5 (Thur):** Text-to-image models and SOTA techniques ✅
- **2/10 (Tue):** How to sample from diffusion/flow models with a single step
    - Distillation
    - Consistency models
    - Flow maps
- **2/12 (Thur):** Real-time generation techniques for video (**Linqi (Alex) Zhou**, Luma AI)