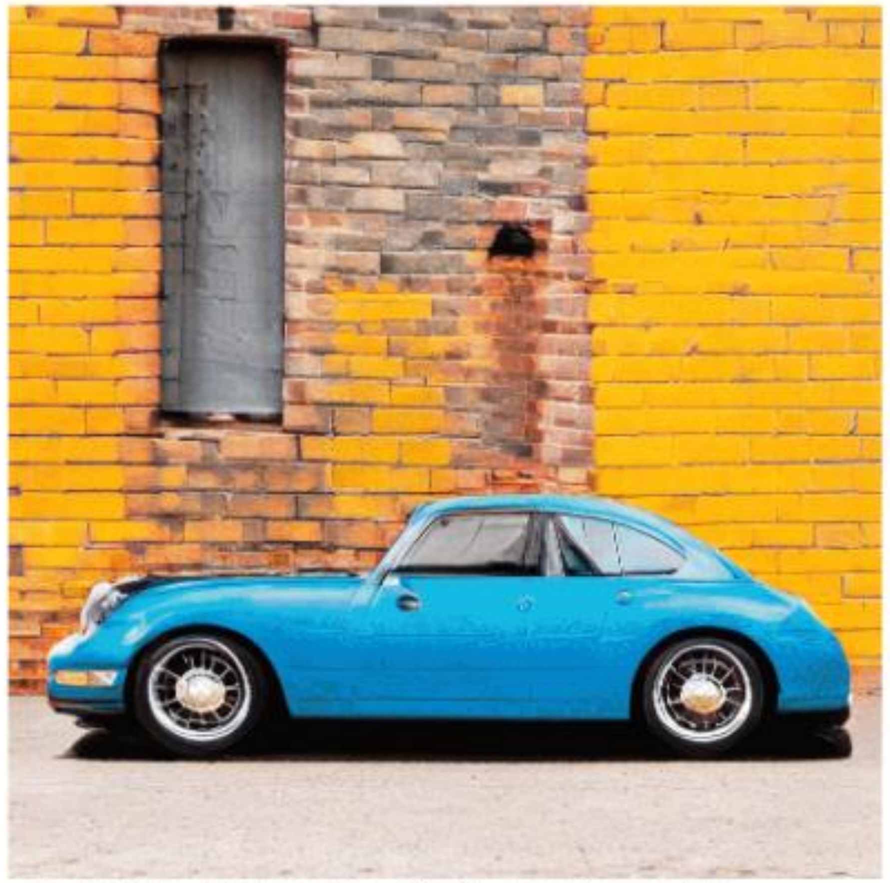
A living room with a fireplace at a wood cabin. Interior design.



a blue Porsche 356 parked in front of a yellow brick wall.

Eiffel Tower, landscape photography

A painting of a majestic royal tall ship in Age of Discovery.

Lecture 14: Text-to-Image Synthesis

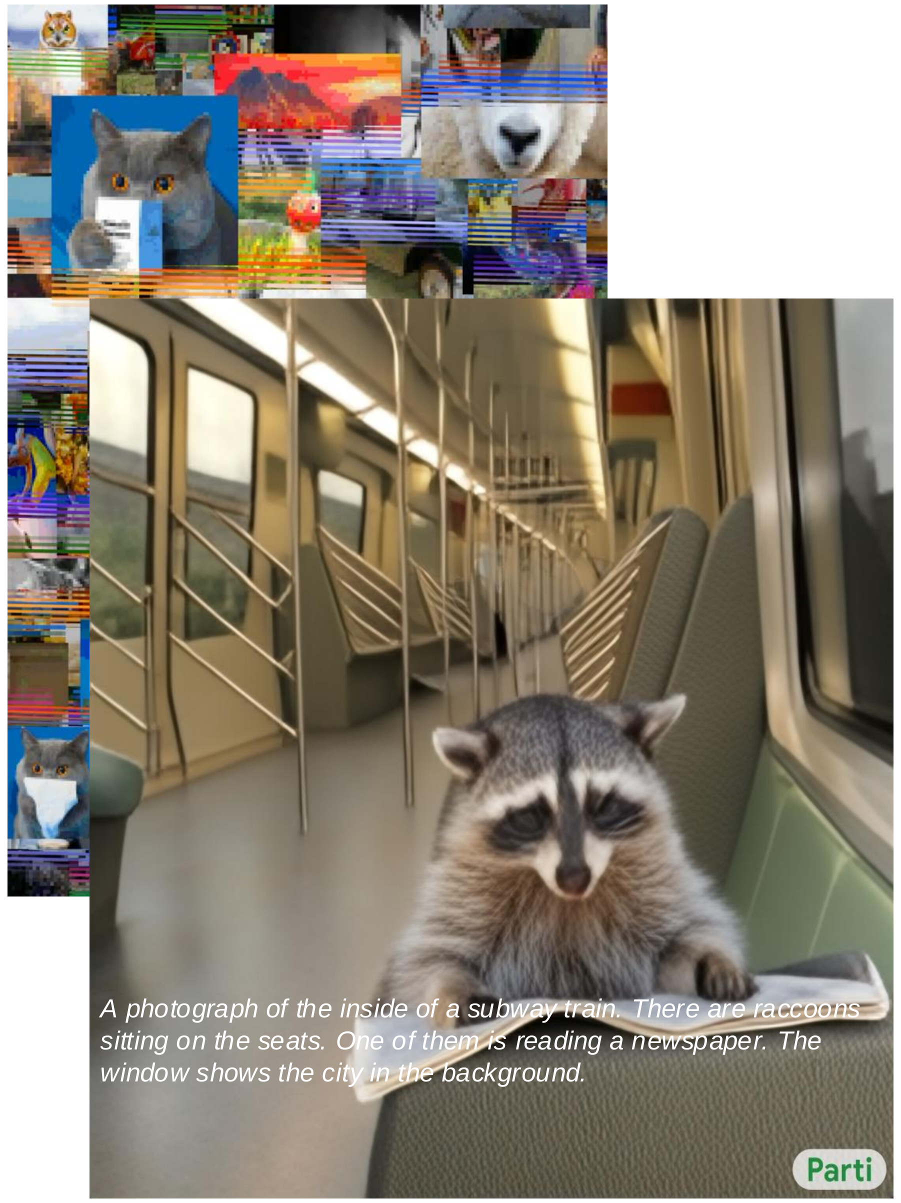
Jun-Yan Zhu

16-726 Spring 2025

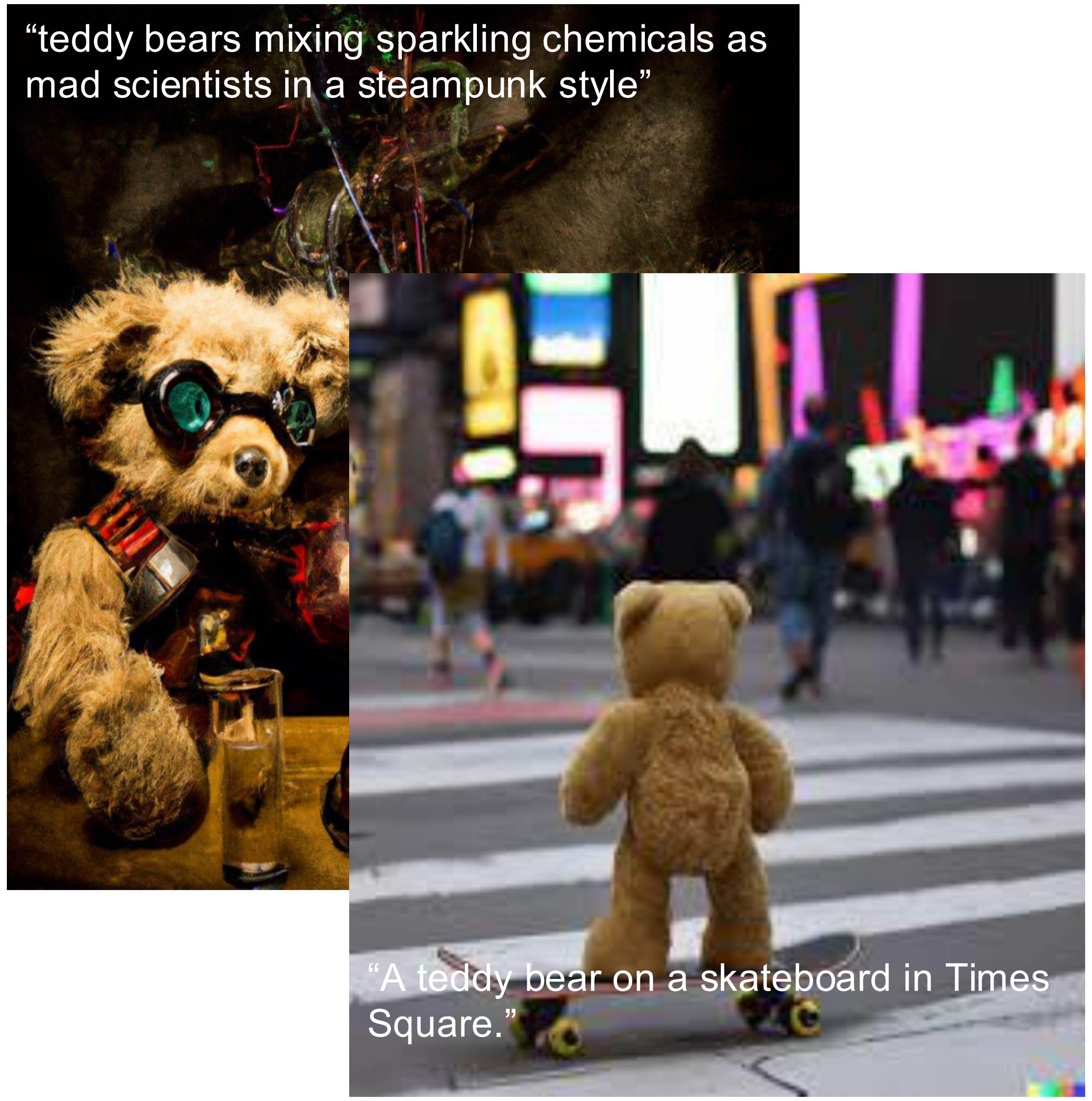
Slides credit: many slides are from Robin Rombach, Karsten Kreis, Ruiqi Gao, Arash Vahdat, etc.

Photo credit: Minguk Kang et al.

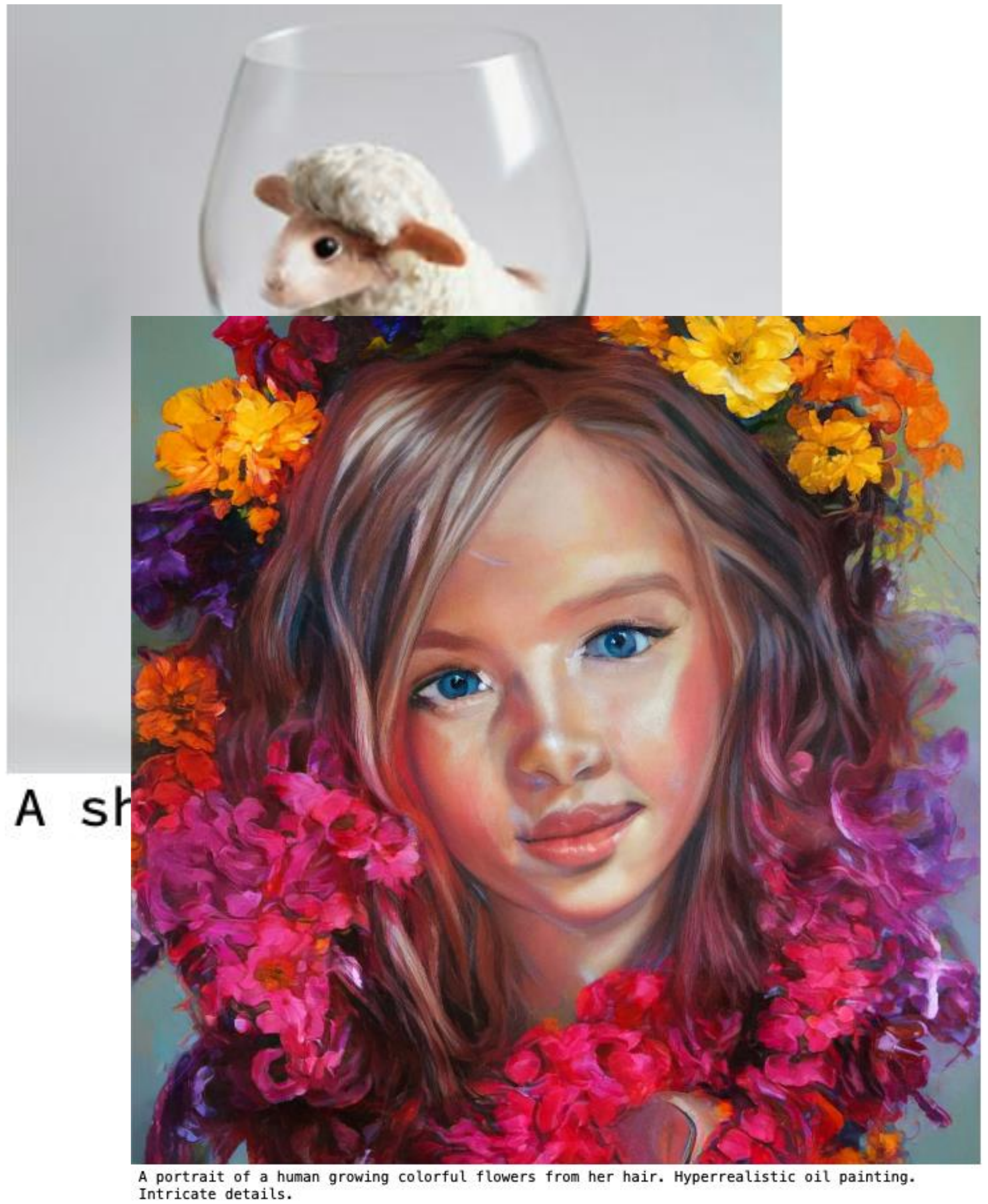
Text-to-Image Everywhere



Autoregressive models
(Image GPT, Parti)

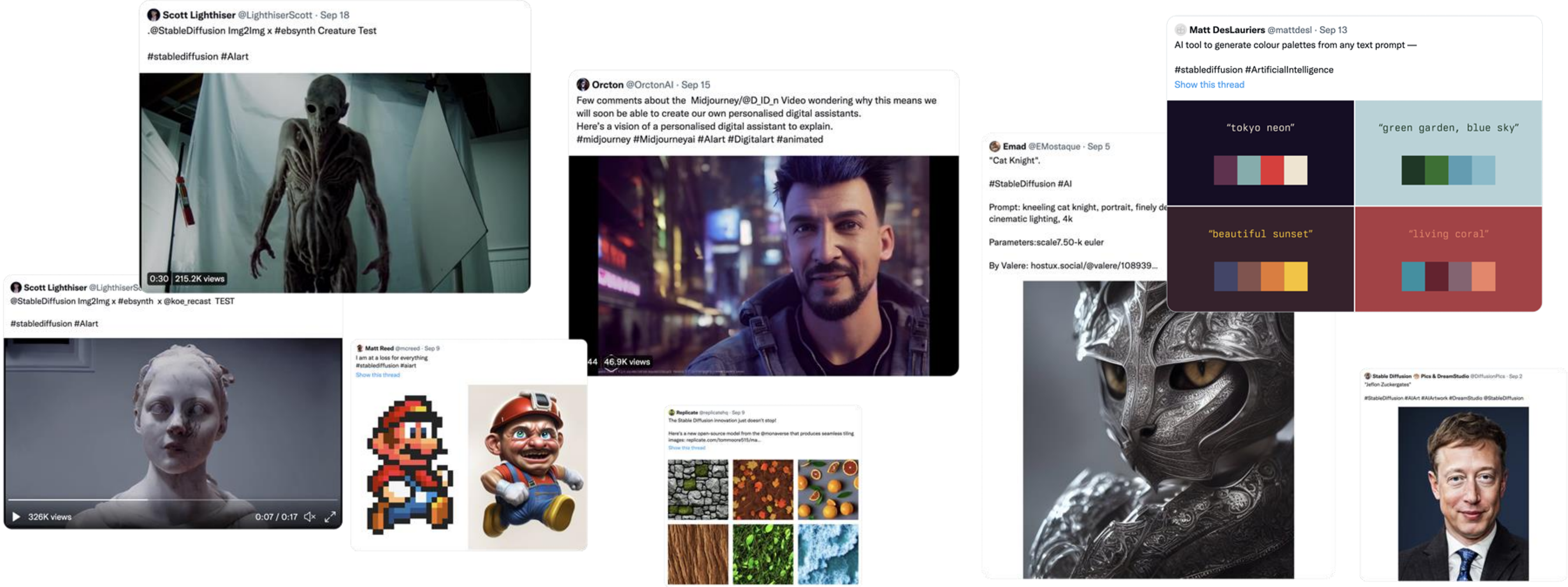


Diffusion models
(DALL-E 2, Imagen)



GANs, Masked GIT
(GigaGAN, MUSE)

Text-to-Image Everywhere



Slides credit: Robin Rombach

Where/when did it start?

First Text-to-Image System

Step 1: Image Selection.
Step 2: Layout Optimization (Minimum overlap, Centrality, Closeness)

A Text-to-Picture Synthesis System for Augmenting Communication
Xiaojin Zhu, Andrew Goldberg, Mohamed Eldawy, Charles Dyer, and Bradley Strock. AAAI 2007

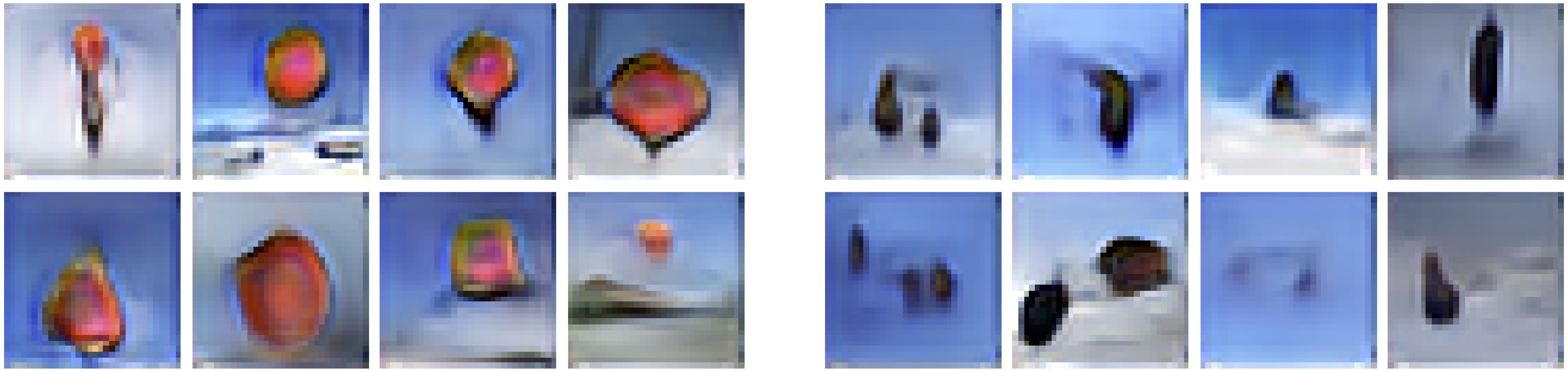
First Text-to-Image System

Therapy for people
with communicative disorders

Math learning and reading comprehension
for young children

A Text-to-Picture Synthesis System for Augmenting Communication
Xiaojin Zhu, Andrew Goldberg, Mohamed Eldawy, Charles Dyer, and Bradley Strock. AAAI 2007

First Deep Learning Work



A stop sign is flying in blue skies.

A herd of elephants flying in the blue skies.

Generating Images from Captions with Attention.
Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, Ruslan Salakhutdinov. ICLR 2016.

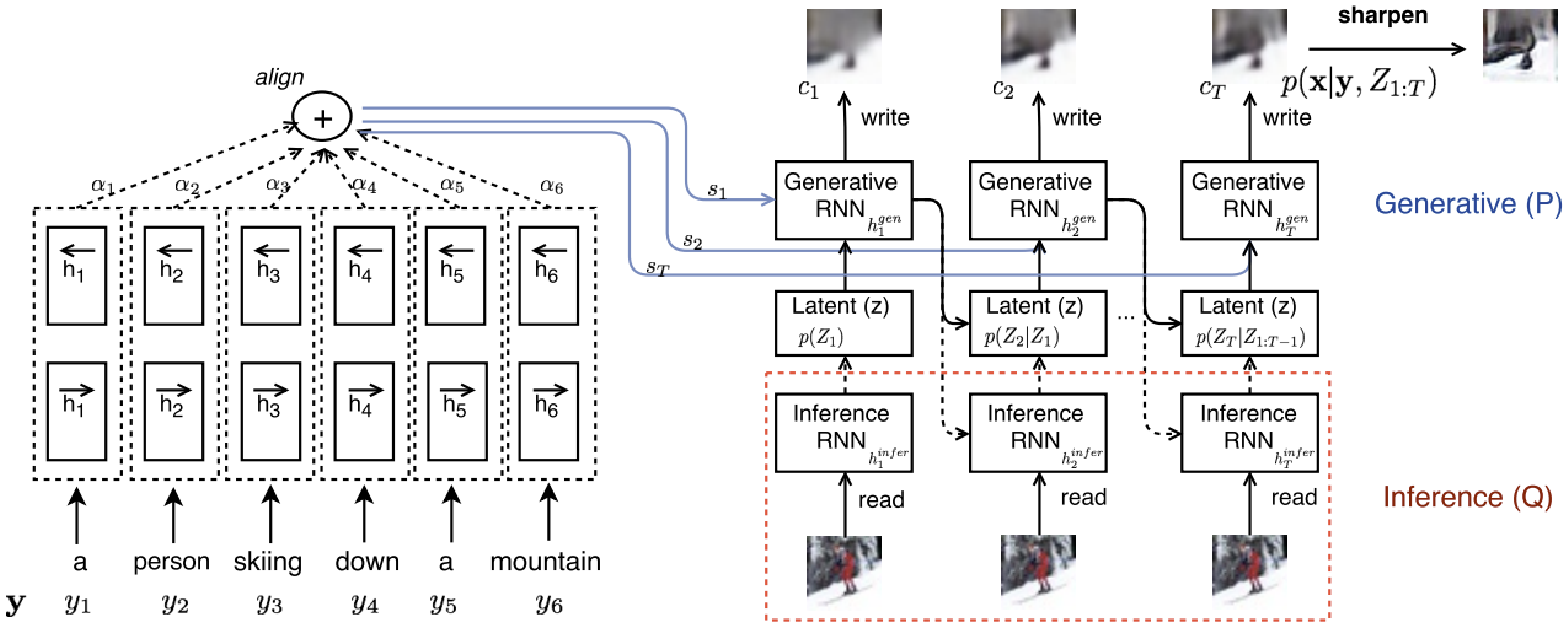
First Deep Learning Work

A toilet seat sits open in the grass field.

A person skiing on sand clad vast desert.

Generating Images from Captions with Attention.
Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, Ruslan Salakhutdinov. ICLR 2016.

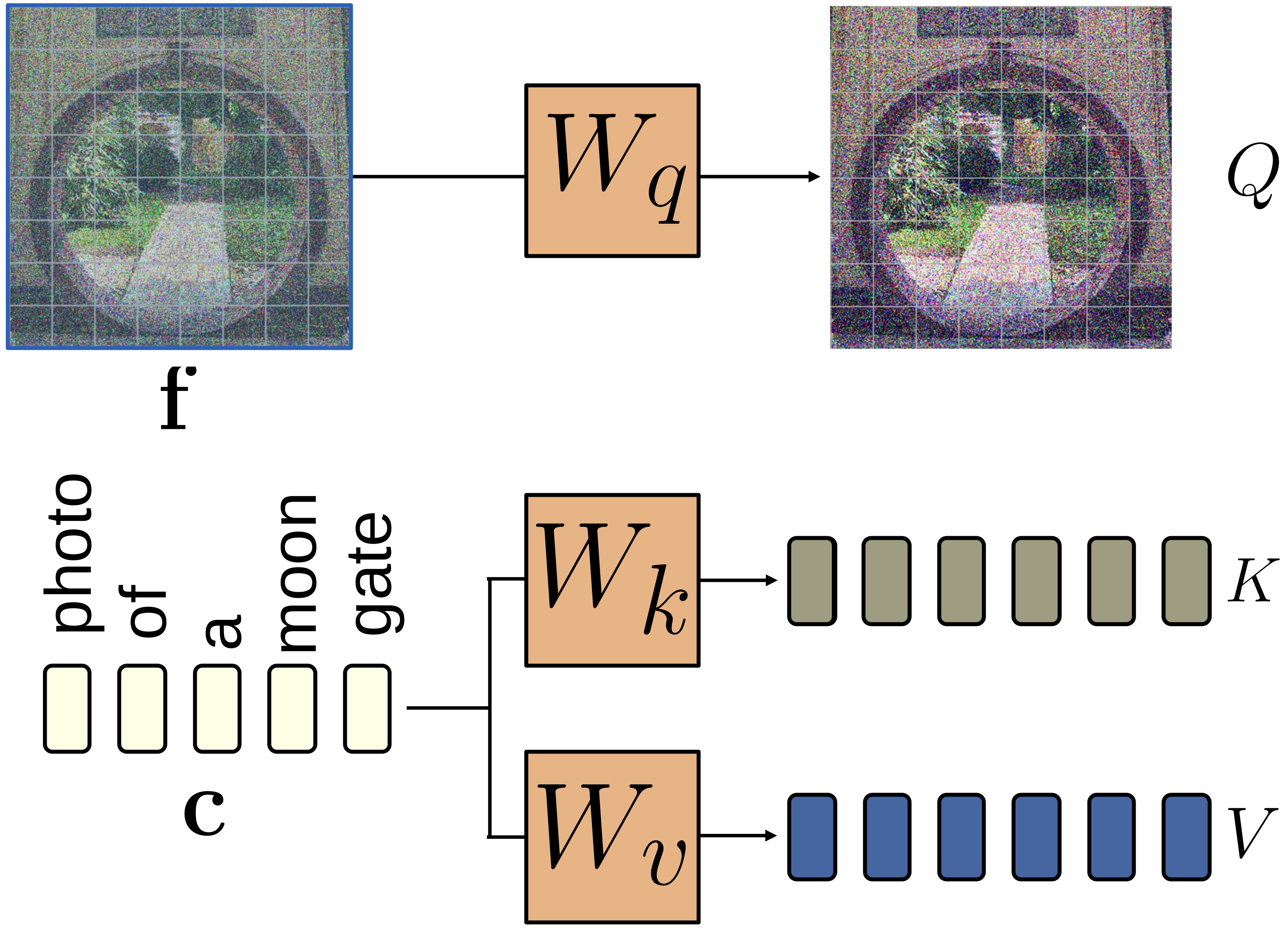
First Deep Learning Work

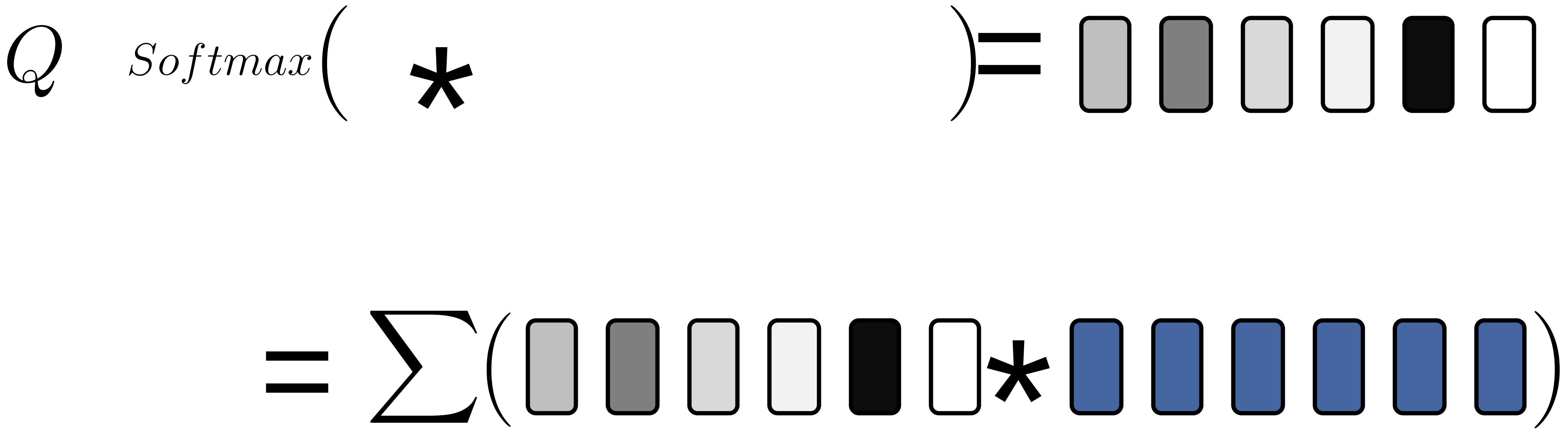


VAES + RNN+ cross-attention

Generating Images from Captions with Attention.
Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, Ruslan Salakhutdinov. ICLR 2016.

Text-Image Cross-Attention





i.e.

Output = $\mathrm{Softmax}\left(\frac{Q.K^T}{\sqrt{d'}}\right)V$

How could we improve it?

How could we improve it?

- Better generative modeling techniques.
- Better text encoders.
- Better generator architectures.
- Better ways to connect text and image.
- Bigger data + more GPU/TPU computing.
- Bigger model sizes.

GANs-based Text-to-Image

Generative Adversarial Text to Image Synthesis
Scott Reed et al., ICML 2016

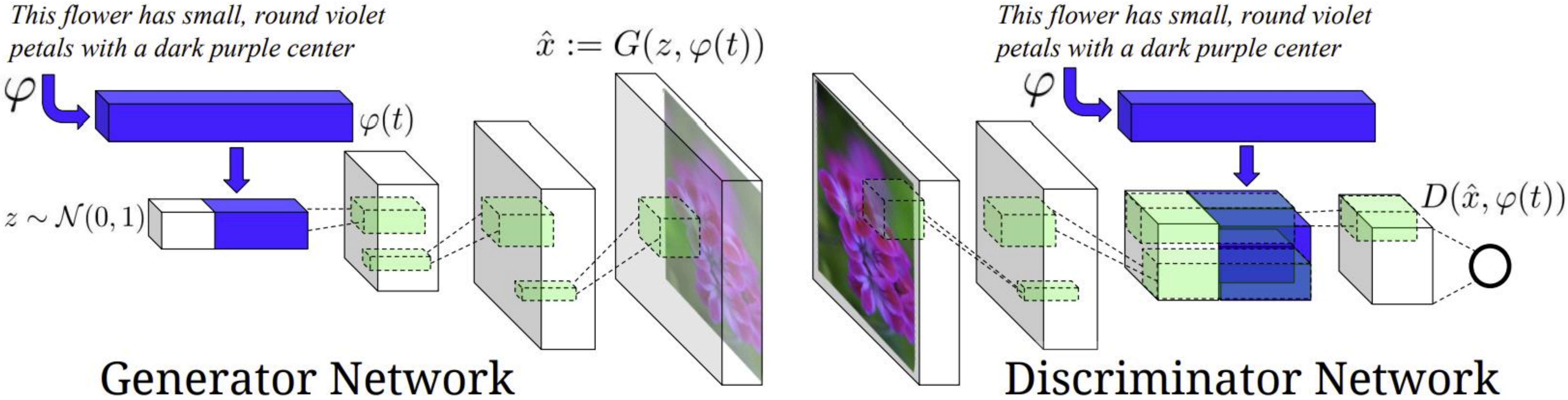
GANs-based Text-to-Image

the flower has petals that are bright pinkish purple with white stigma

this white and yellow flower have thin white petals and a round yellow stamen

Generative Adversarial Text to Image Synthesis
Scott Reed et al., ICML 2016

GANs-based Text-to-Image

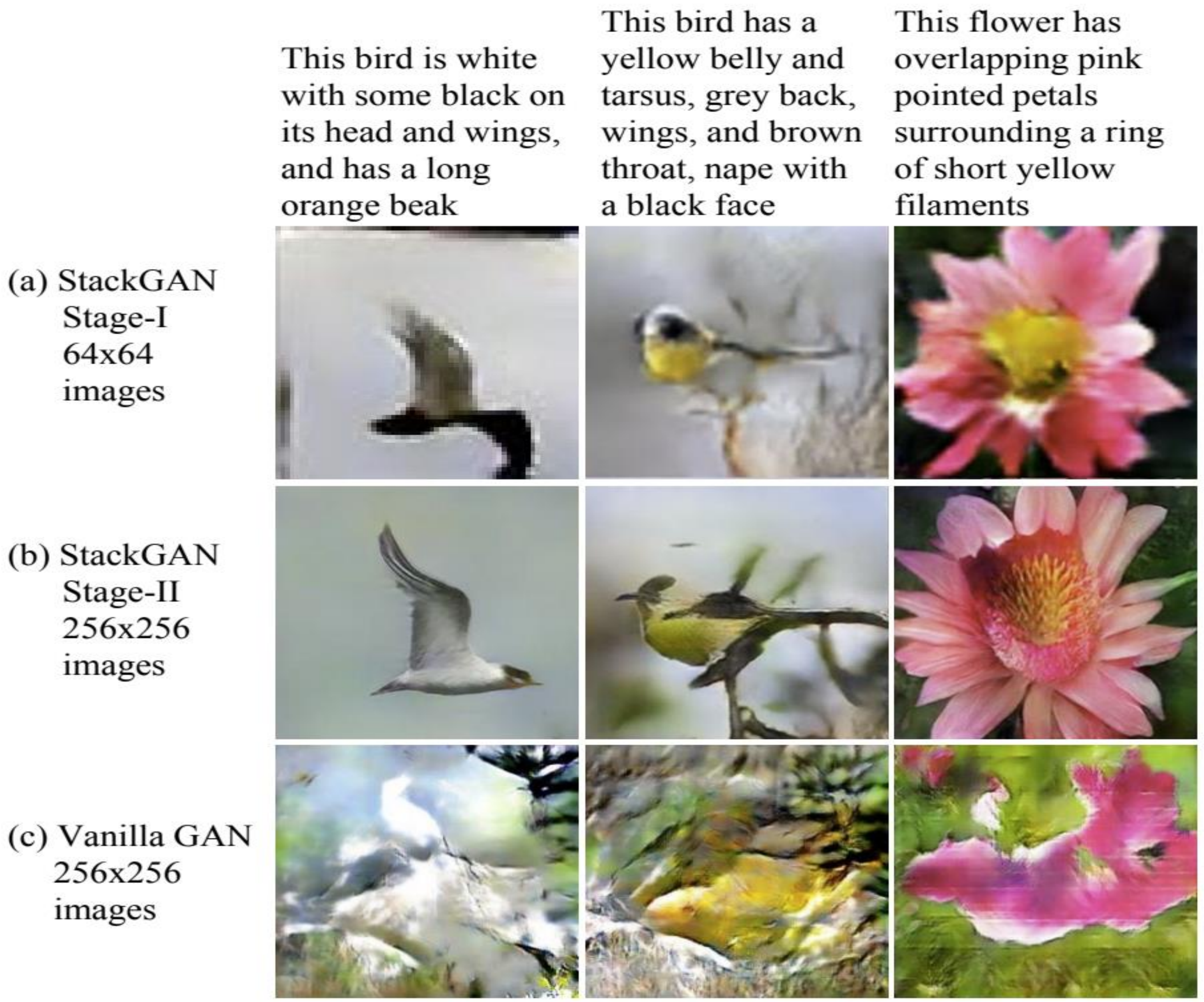


Conditional GAN + CNN + concatenation

Generative Adversarial Text to Image Synthesis
Scott Reed et al., ICML 2016

How to increase resolution?

+Two-stage Models

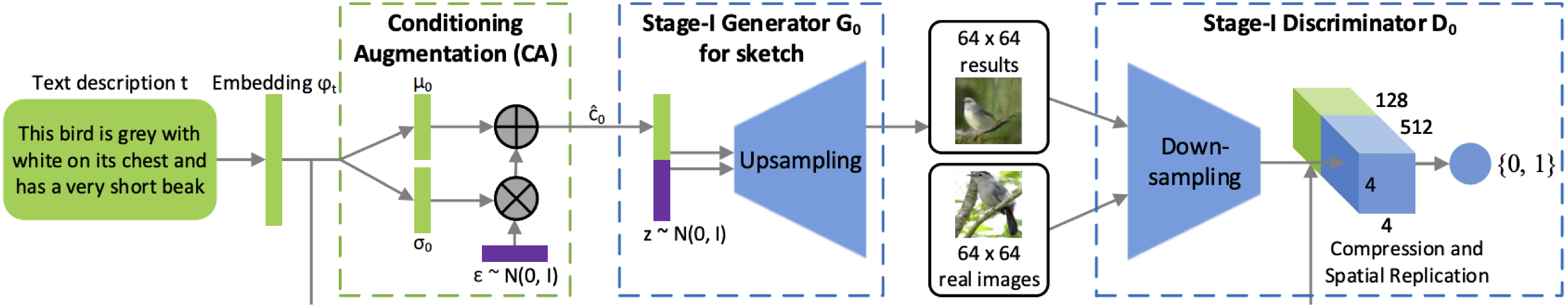


Two-stage Conditional GAN + CNN + concatenation

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks

Han Zhang et al., ICCV 2017

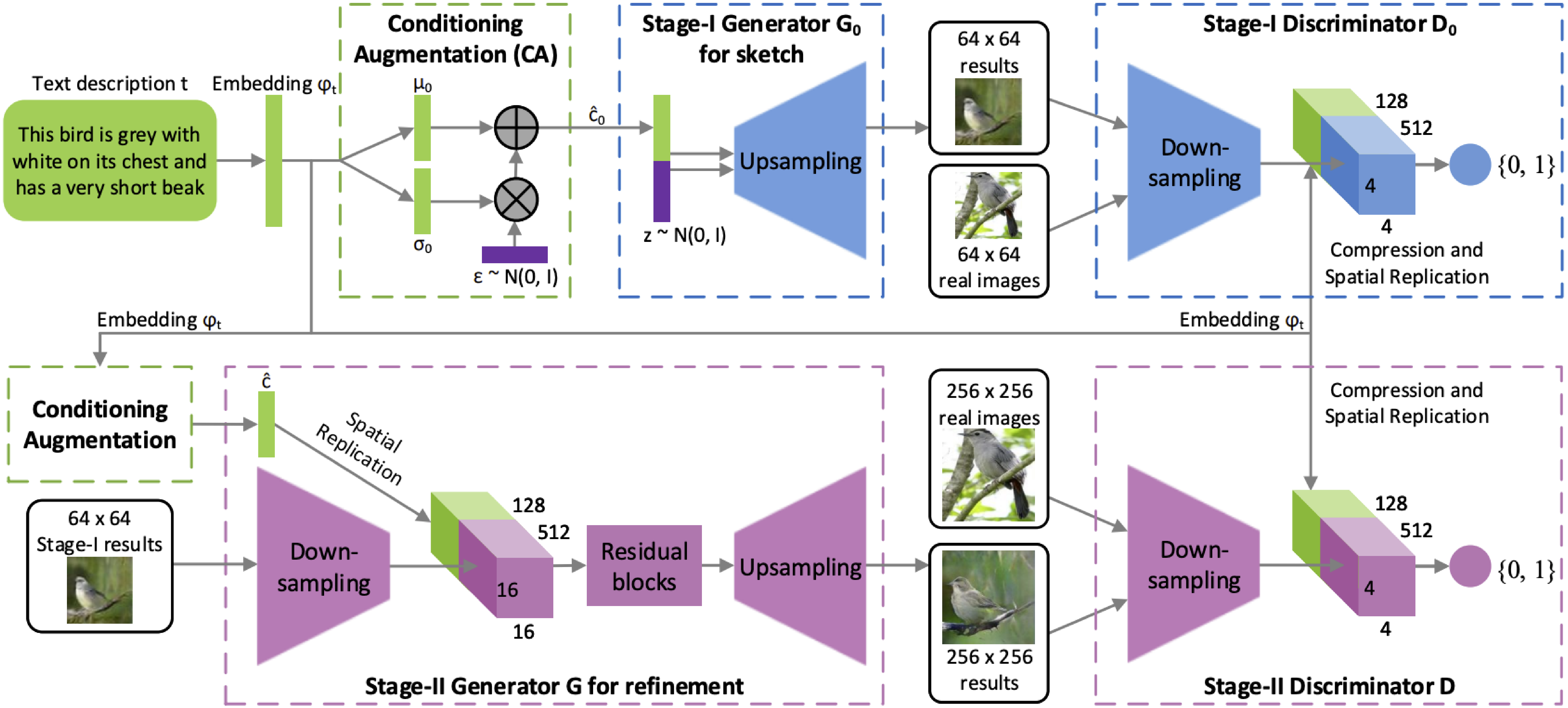
+Two-stage Models



Two-stage Conditional GAN + CNN + concatenation

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks
Han Zhang et al., ICCV 2017

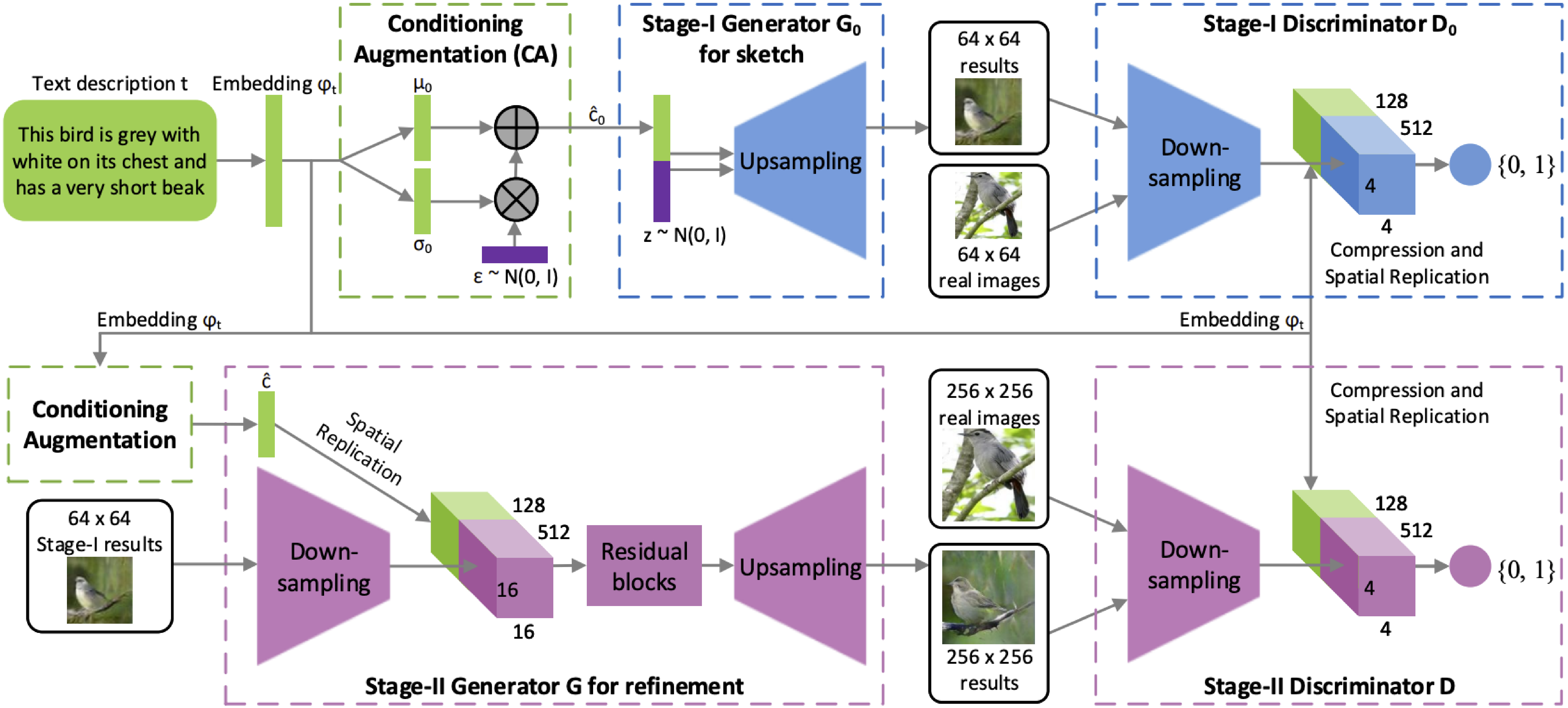
+Two-stage Models



Two-stage Conditional GAN + CNN + concatenation

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks
Han Zhang et al., ICCV 2017

+Two-stage Models



Two-stage Conditional GAN + CNN + concatenation

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks
Han Zhang et al., ICCV 2017

+Two-stage Models

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks
Han Zhang et al., ICCV 2017

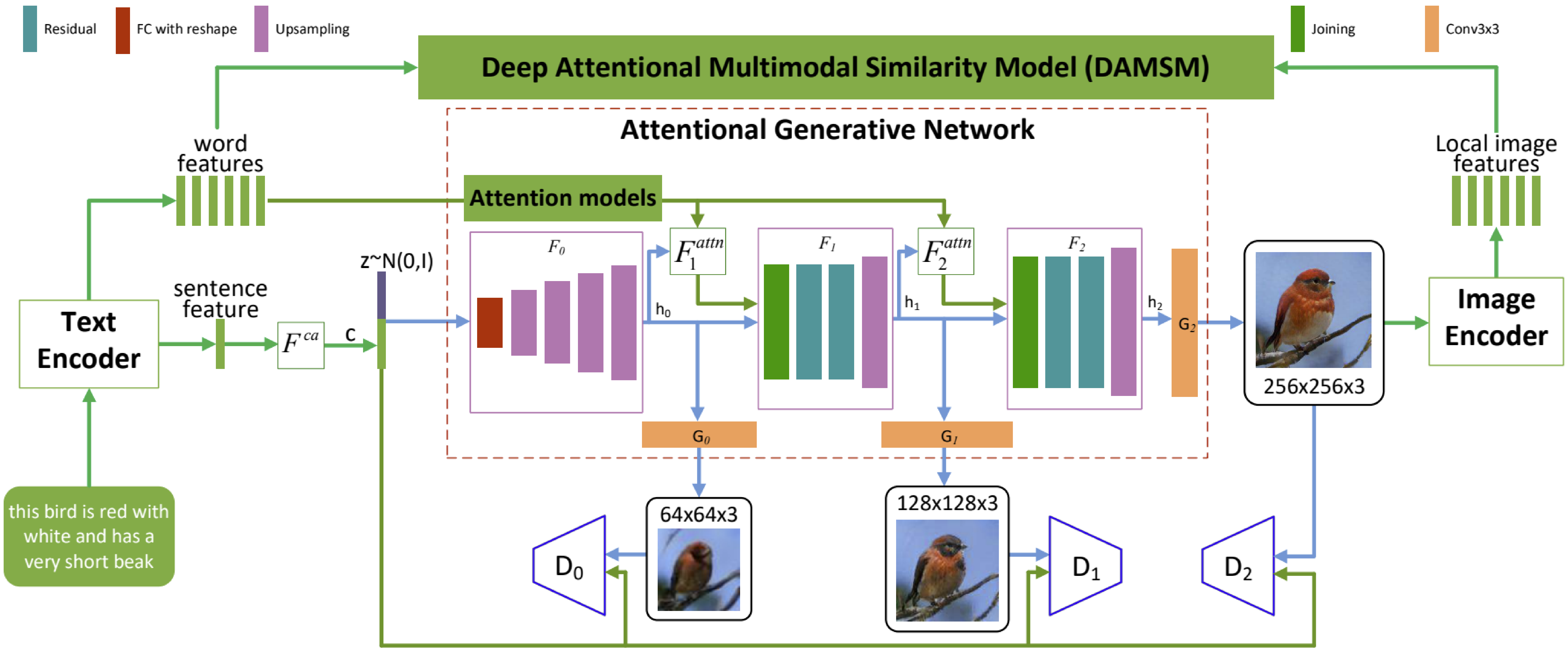
+Two-stage Models

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks
Han Zhang et al., ICCV 2017

+ Cross-attention to connect Text and Image

AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks
Tao Xu et al., CVPR 2018

+ Cross-attention to connect Text and Image



AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks
Tao Xu et al., CVPR 2018

Got Stuck in 2018-2020
(Birds, MS COCO)

Who shall we blame?

- **Better generative modeling techniques: VAEs, GANs?**
- **Better text encoders: LSTM/RNN?**
- **Better generator architectures: CNNs?**
- Better ways to connect text and image.
- Bigger data + more GPU/TPU computing.
- Bigger model sizes.

How could we synthesize images beyond single or a few categories

033 [cs.CV] 23 Jun 2021

Taming Transformers for High-Resolution Image Synthesis

Patrick Esser* Robin Rombach* Björn Ommer
Heidelberg Collaboratory for Image Processing, IWR, Heidelberg University, Germany
*Both authors contributed equally to this work

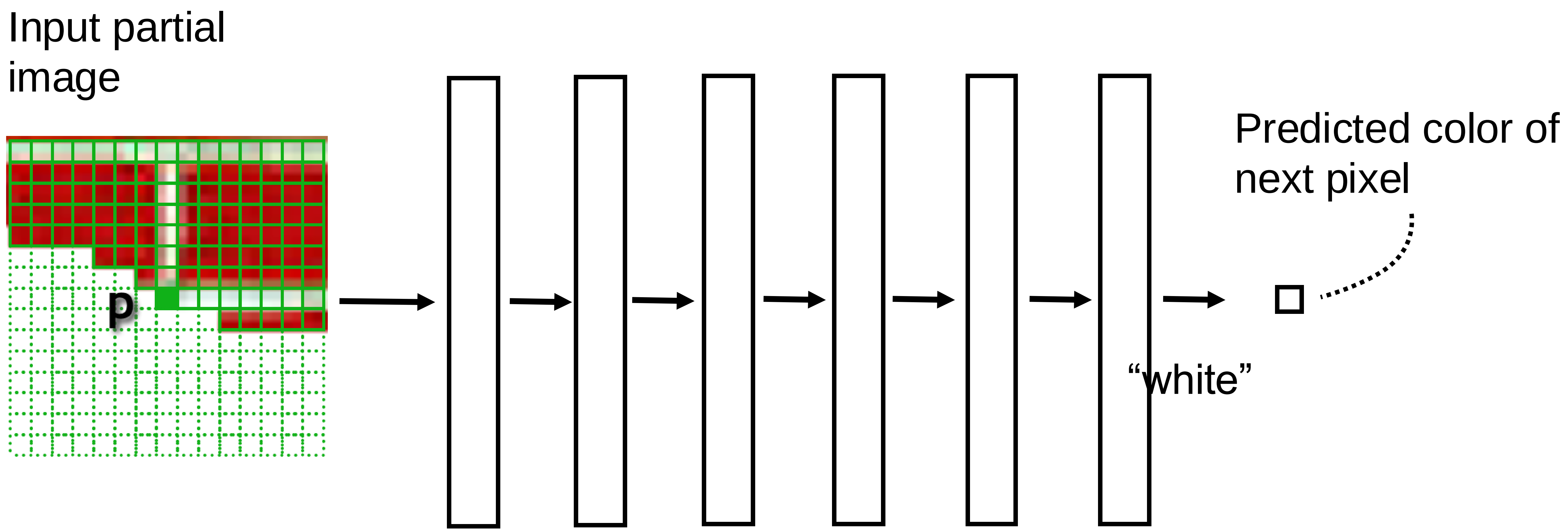
Figure 1. Our approach enables transformers to synthesize high-resolution images like this one, which contains 1280x460 pixels.

Abstract

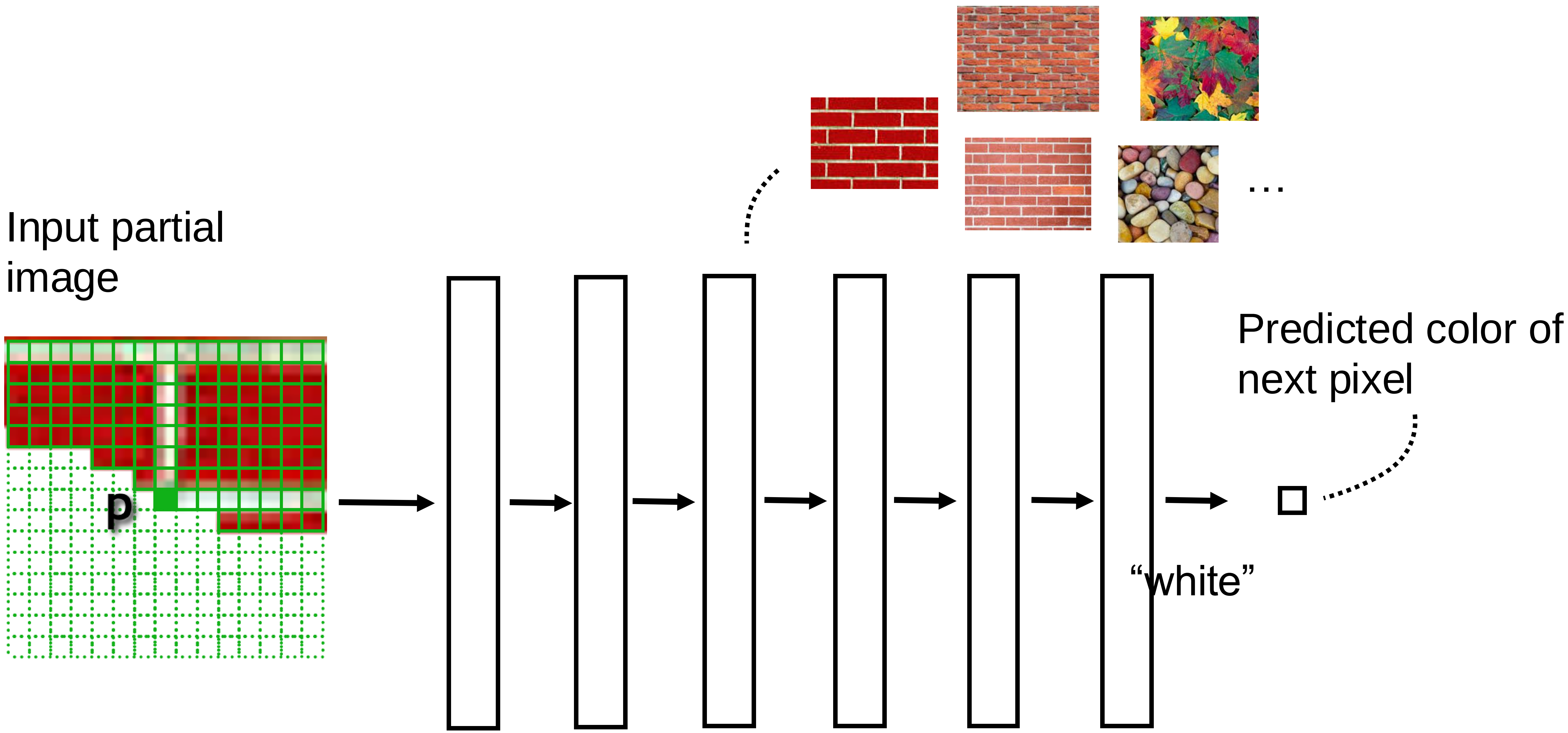
Designed to learn long-range interactions on sequential
... state-of-the-art results

and are increasingly adapted in other areas such as audio [12] and vision [8, 16]. In contrast to the predominant vision architecture, convolutional neural networks (CNNs), the transformer architecture contains no built-in inductive prior on the locality of interactions and is therefore free

Autoregressive (AR) image synthesis

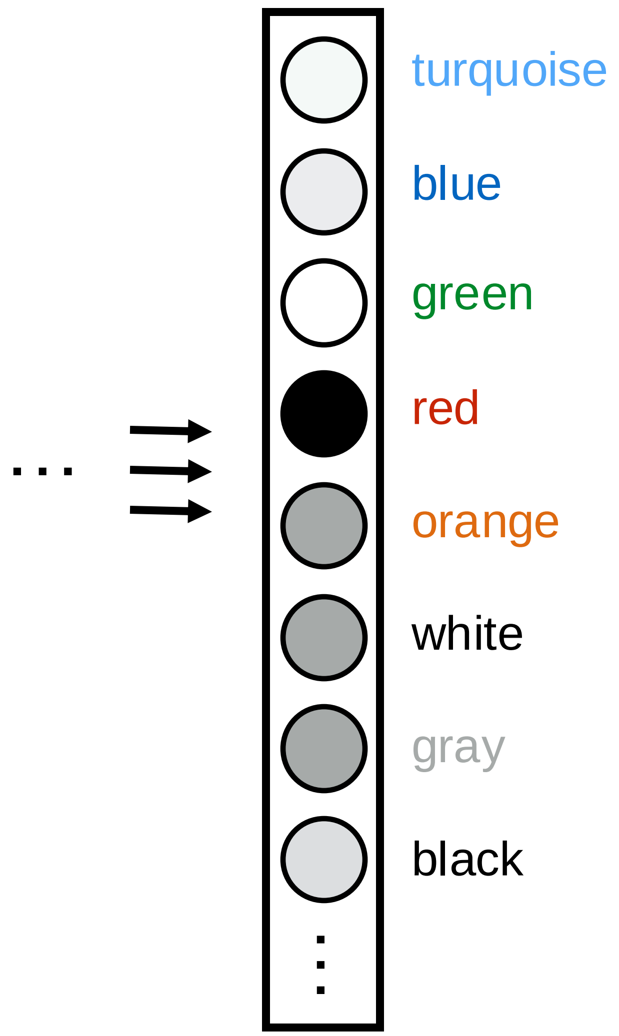


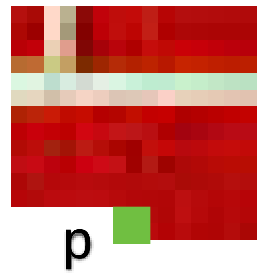
[PixelRNN, PixelCNN, van der Oord et al. 2016]



[PixelRNN, PixelCNN, van der Oord et al. 2016]

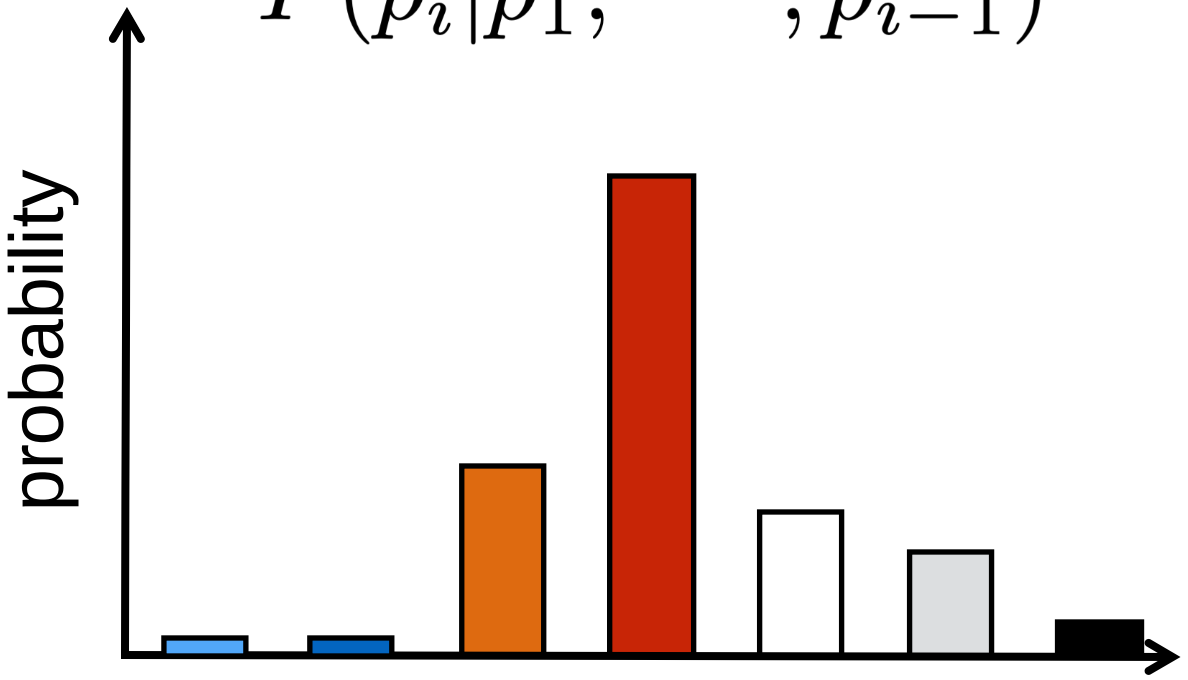
Network output

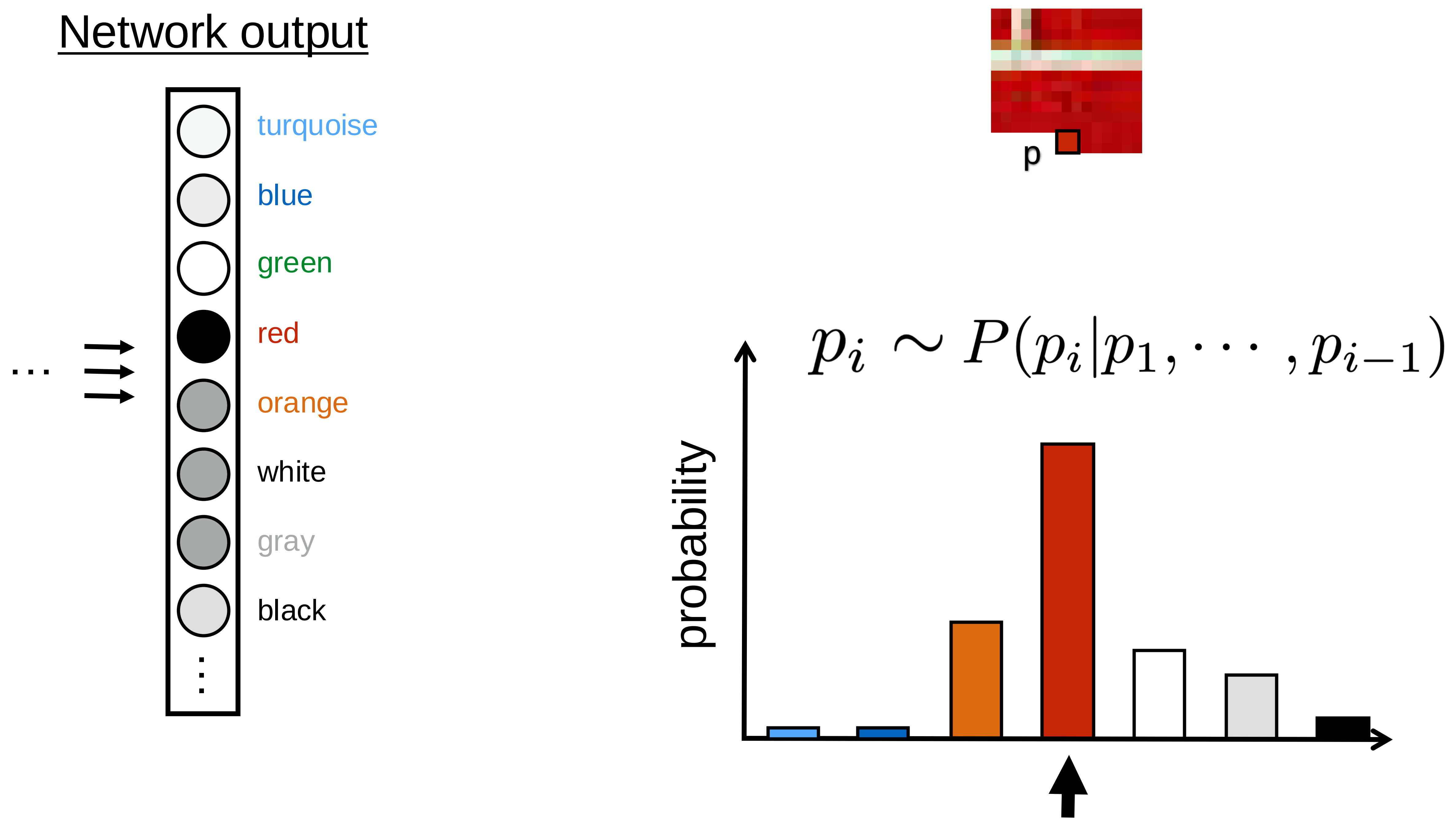




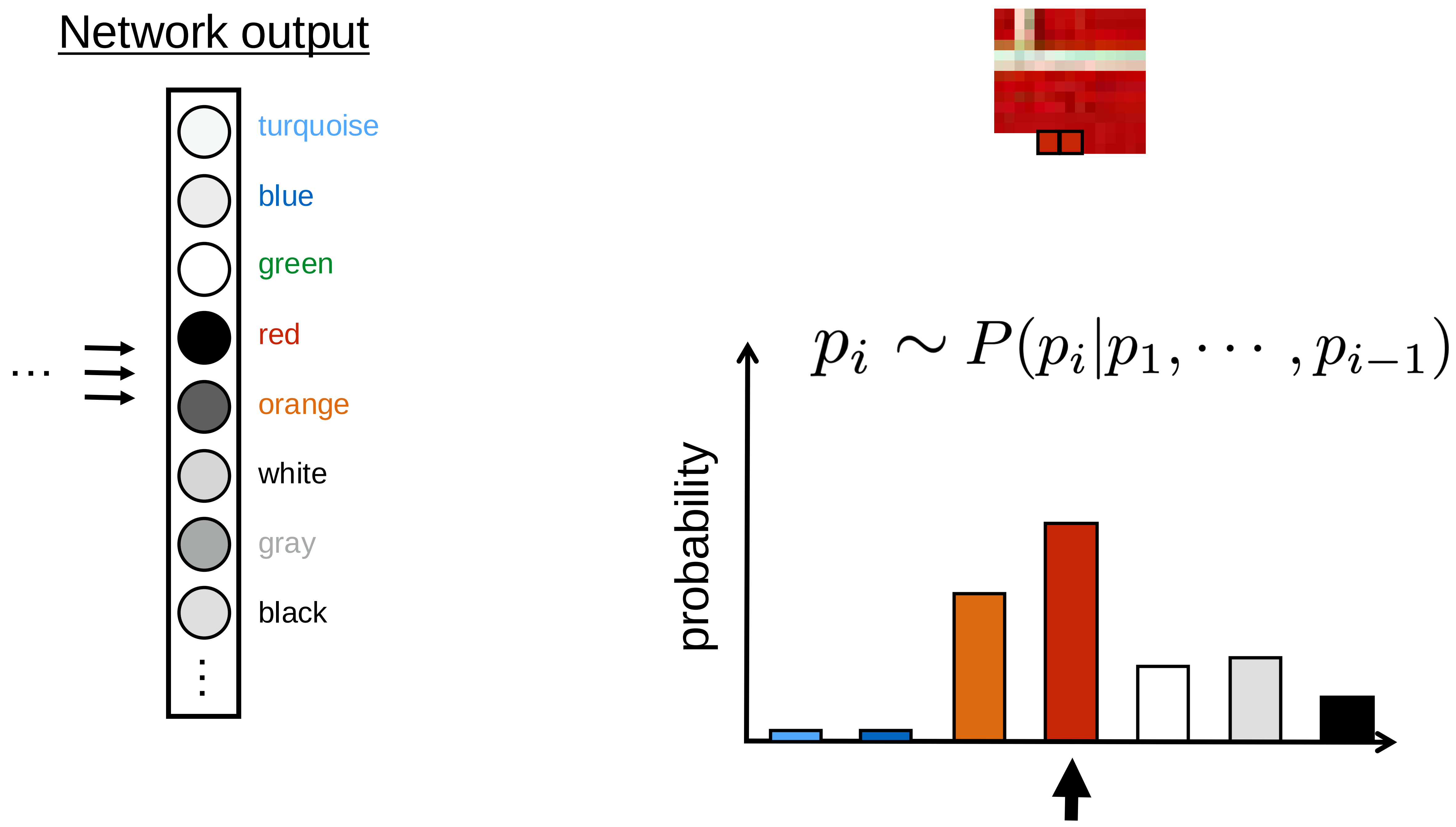
P(next pixel | previous pixels)

$$P(p_i|p_1, \cdots, p_{i-1})$$

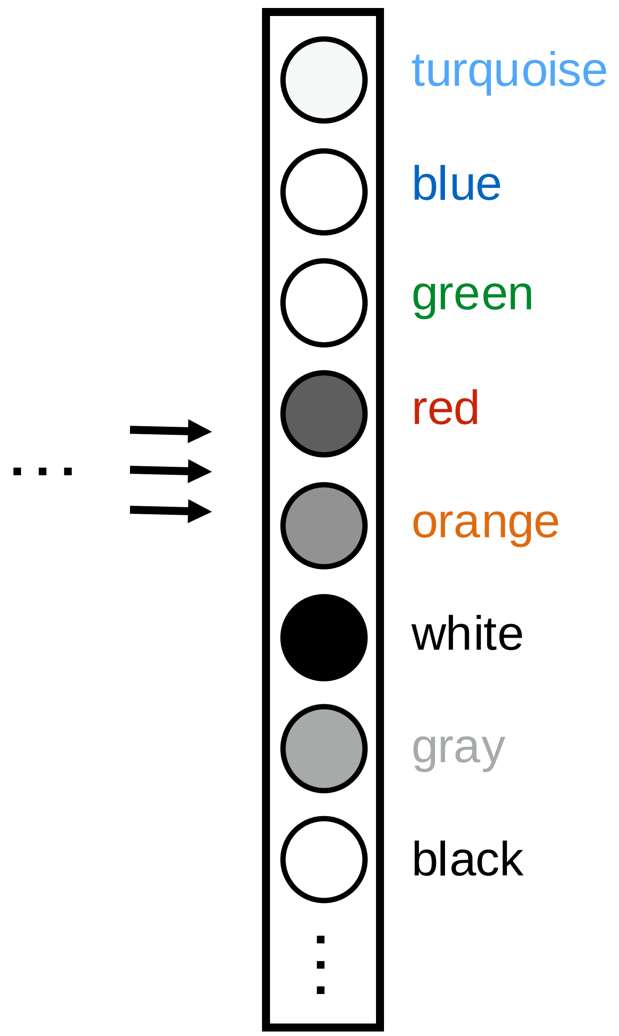


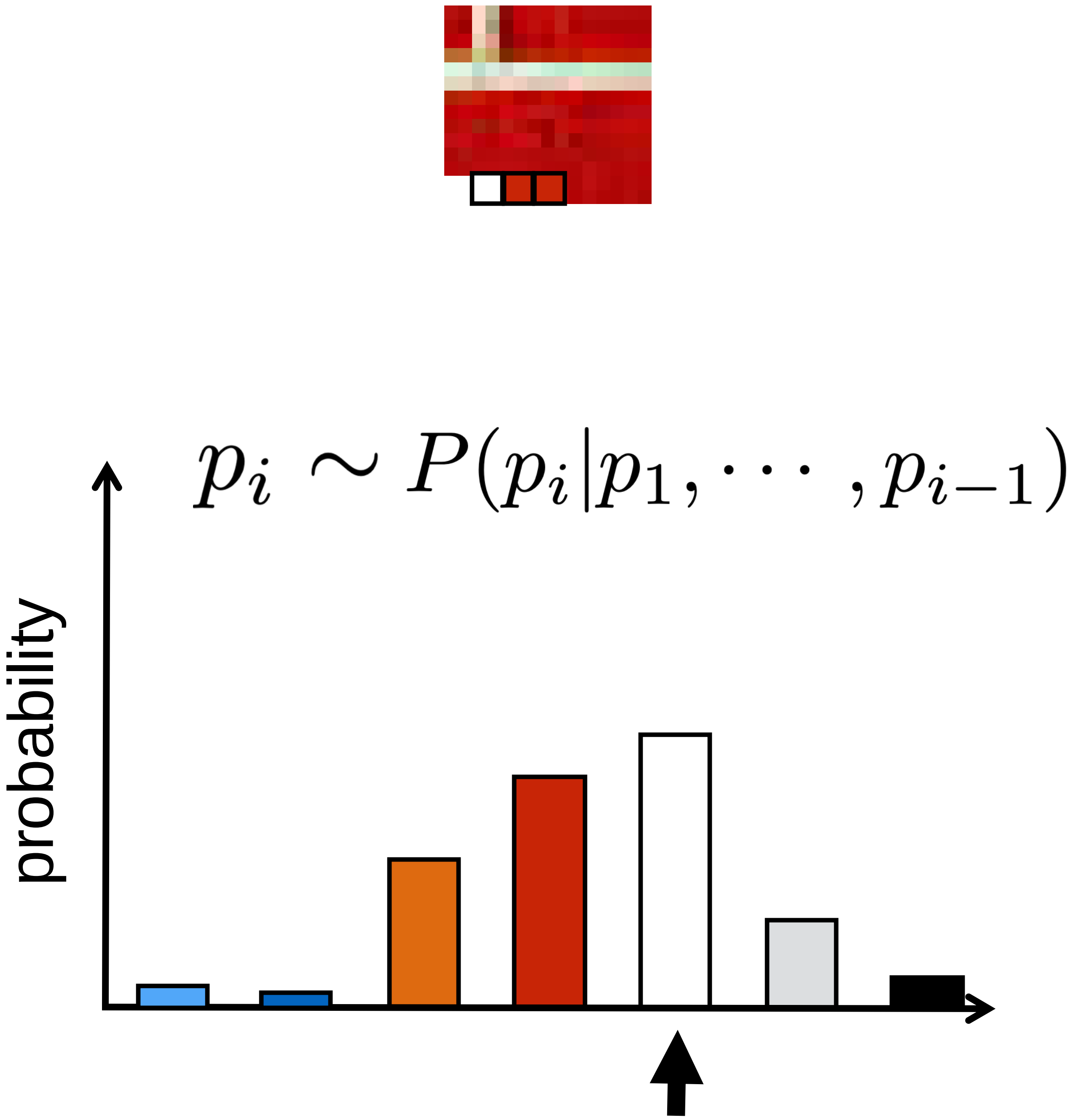


Network output
turquoise
blue
green
red
orange
white
gray
black
p
$p_i \sim P(p_i|p_1, \cdots, p_{i-1})$
probability

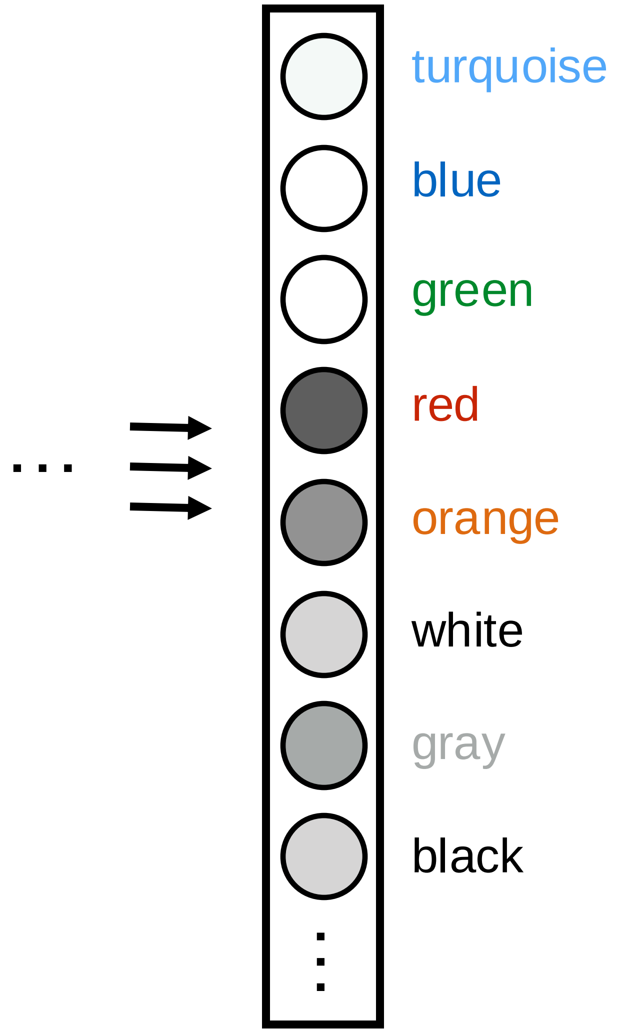

Network output
turquoise
blue
green
red
orange
white
gray
black
$p_i \sim P(p_i | p_1, \cdots, p_{i-1})$
probability

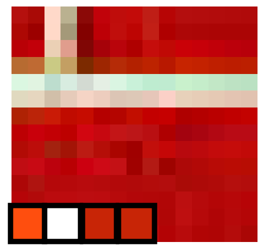
Network output



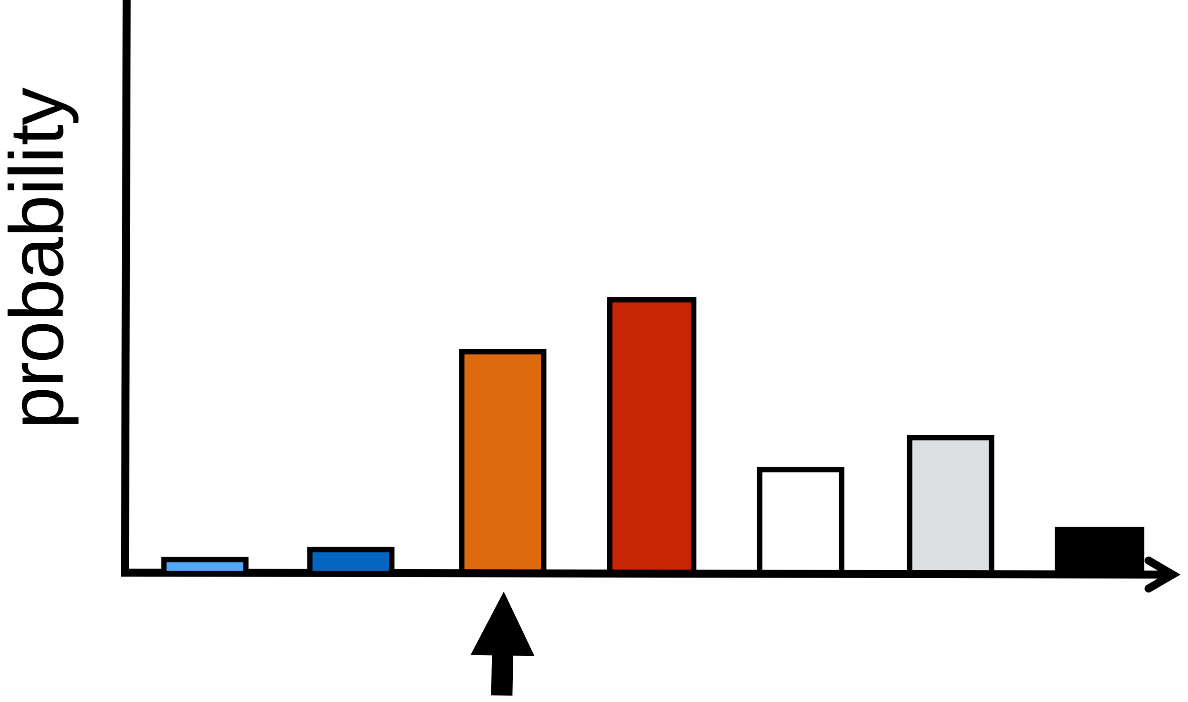


Network output



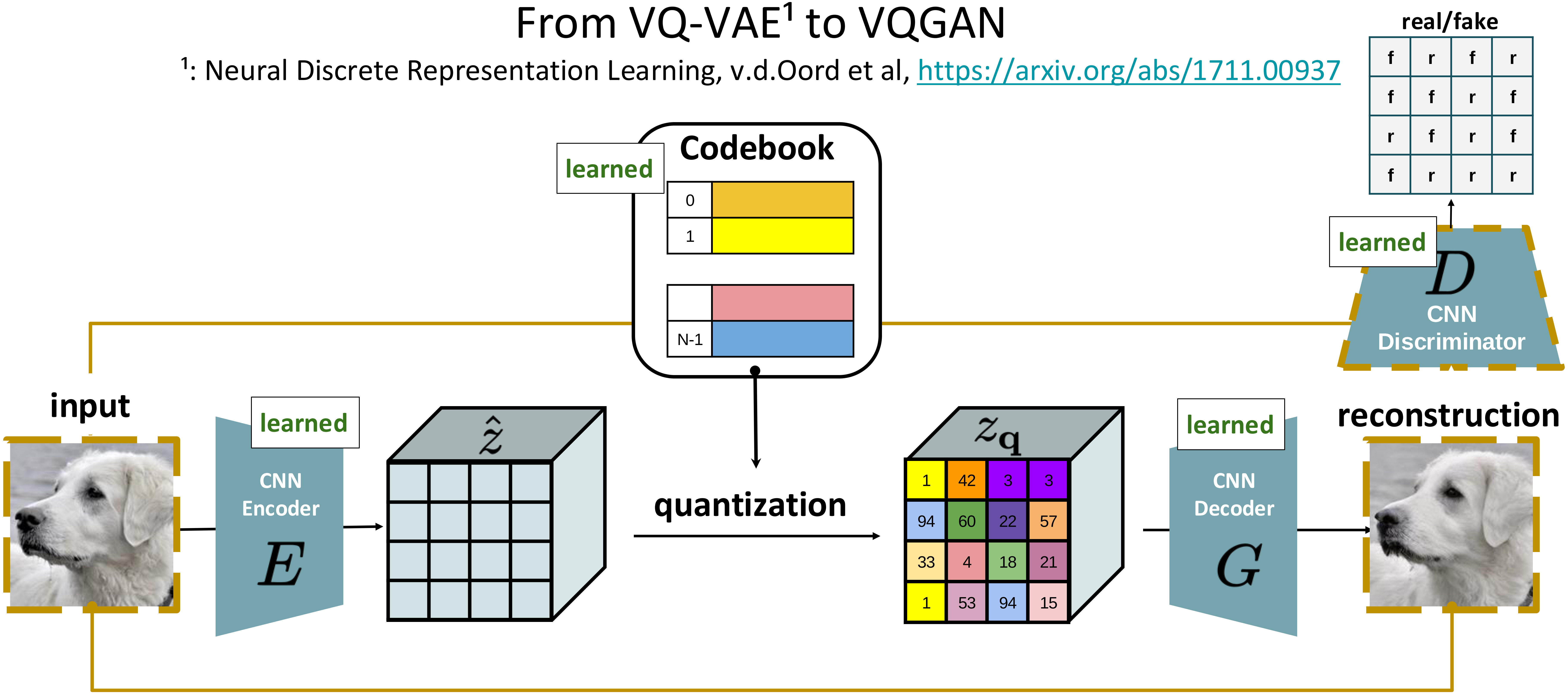


$$p_i \sim P(p_i | p_1, \cdots, p_{i-1})$$



From VQ-VAE[1] to VQGAN

[1]: Neural Discrete Representation Learning, v.d.Oord et al, https://arxiv.org/abs/1711.00937



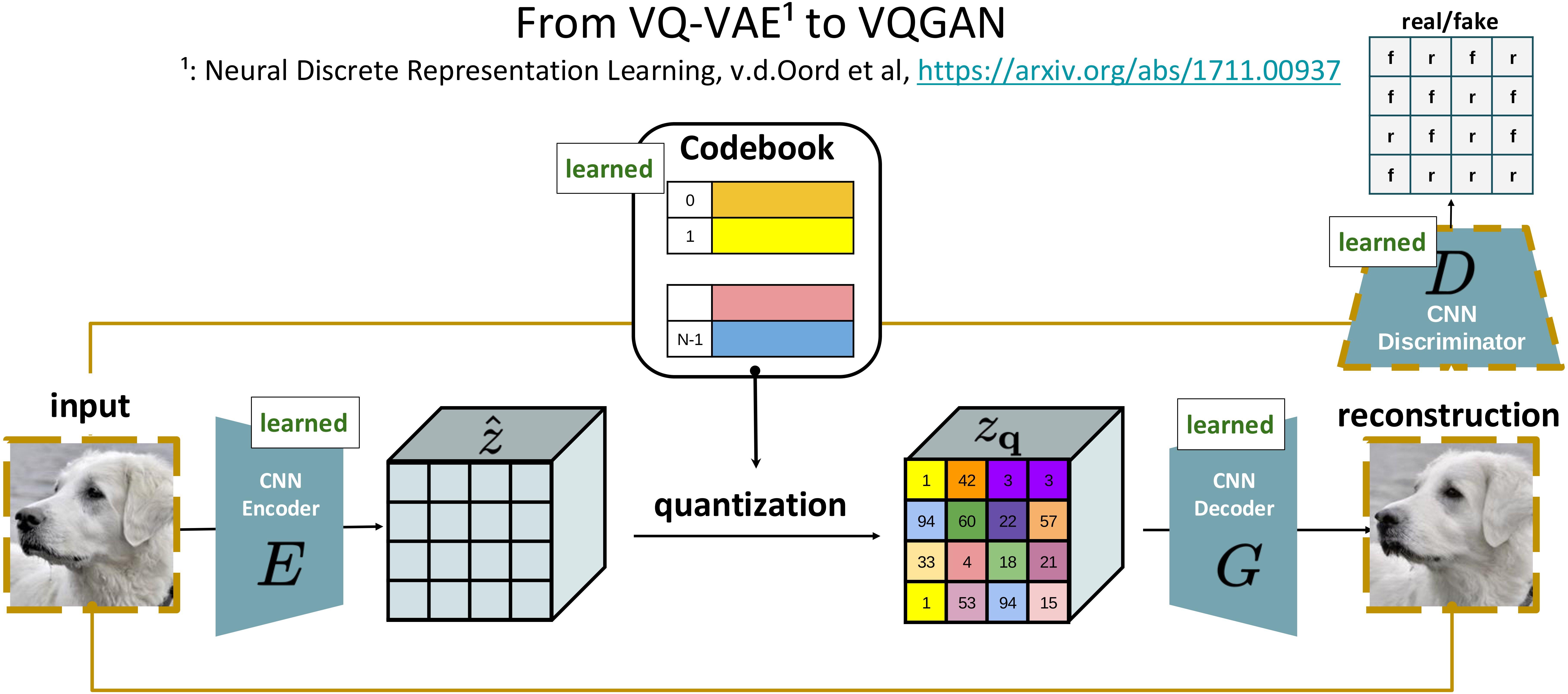
i) replace L2/L1 rec. loss with Perceptual loss (includes pixel-level)

ii) add (patch-wise) Discriminator to favor realism over perfect reconstruction

Slide credit: Robin Rombach

From VQ-VAE[1] to VQGAN

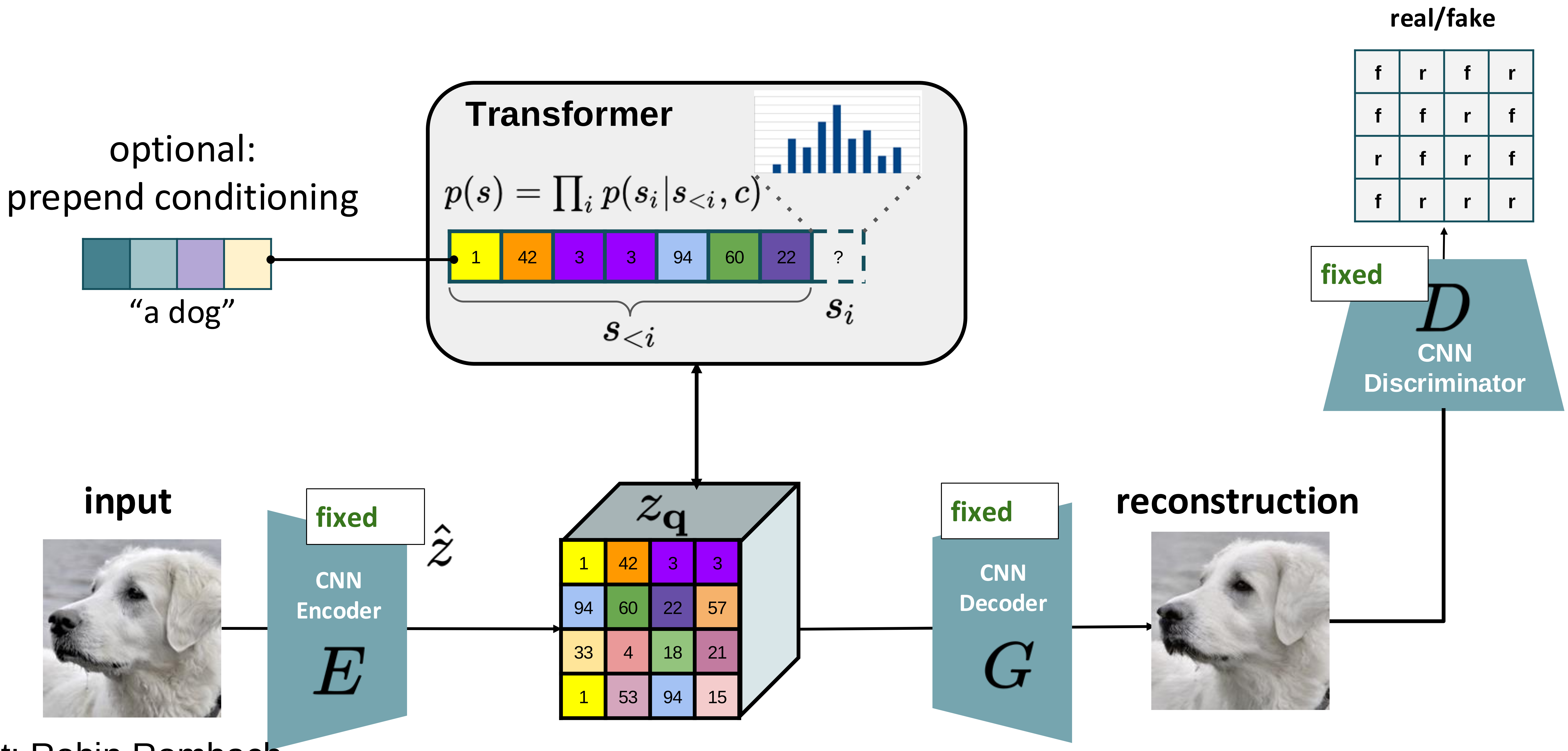
[1]: Neural Discrete Representation Learning, v.d.Oord et al, https://arxiv.org/abs/1711.00937



$$\mathcal{L}_{\text{total}} = \mathcal{L}_{rec} + \mathcal{L}_{VQ} + \lambda\mathcal{L}_{GAN} \text{ where } \lambda = \frac{\nabla_{G_L}[\mathcal{L}_{rec}]}{\nabla_{G_L}[\mathcal{L}_{GAN}]+\delta}$$

Slide credit: Robin Rombach

Transformer Training



Slide credit: Robin Rombach

Slide credit: Robin Rombach

Scaling VQGAN for Text-to-Image!

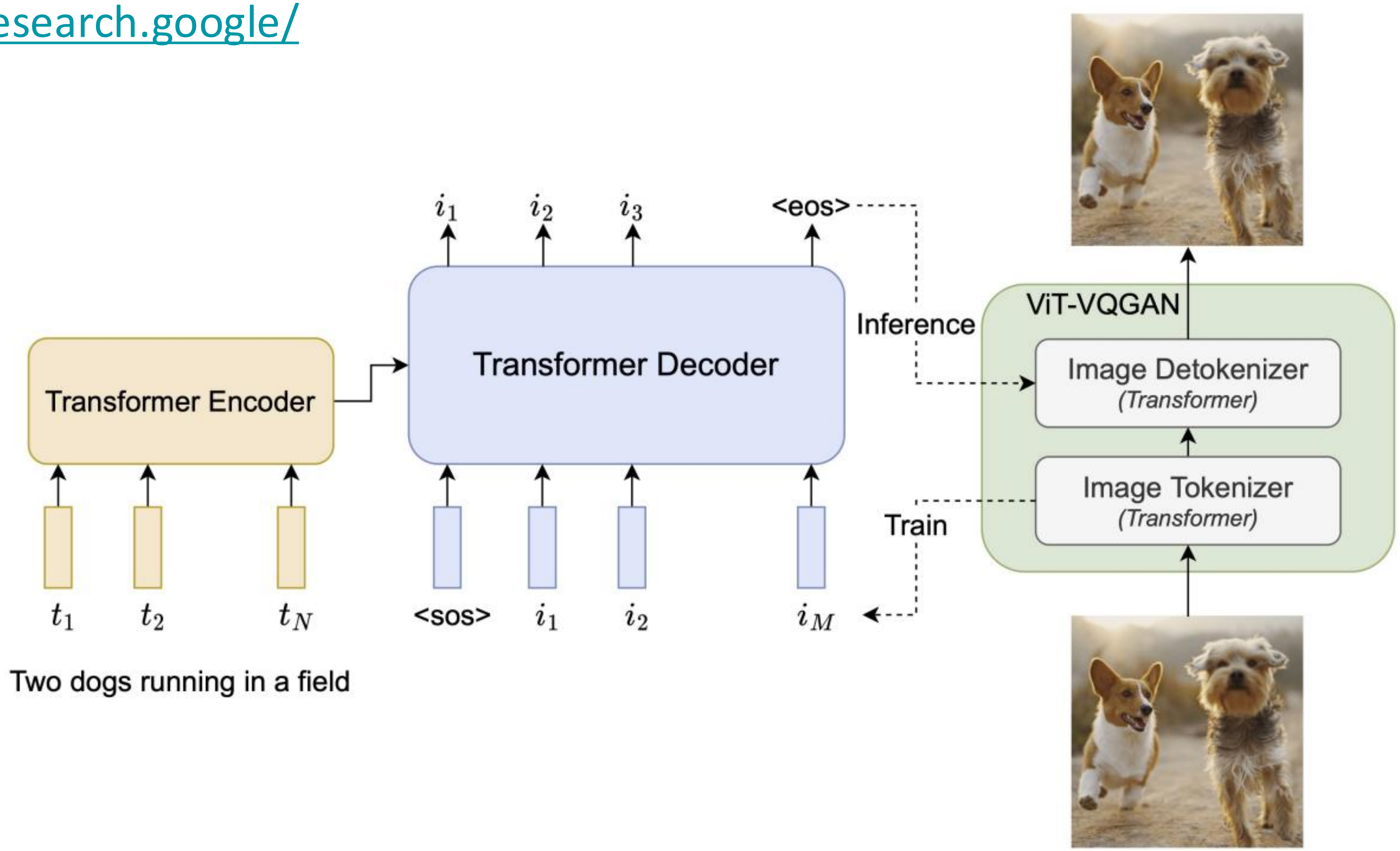
- see recently released “Parti” paper by Google (text-to-image model)
  - https://parti.research.google/

A portrait photo of a kangaroo wearing an orange hoodie and blue sunglasses standing on the grass in front of the Sydney Opera House holding a sign on the chest that says Welcome Friends!

Slide credit: Robin Rombach

Scaling VQGAN for Text-to-Image!

- see recently released “Parti” paper by Google (text-to-image model)
  - https://parti.research.google/



Transformer-based Encoder/Decoder + Transformer-based Autoregressive models

Another Approach: Diffusion Models!

great results for image synthesis

Denoising Diffusion Probabilistic Models

Jonathan Ho, Ajay Jain, et al

https://arxiv.org/abs/2006.11239

Diffusion Models beat GANs on Image Synthesis

Prafulla Dhariwal, Alex Nichol

https://arxiv.org/abs/2105.05233

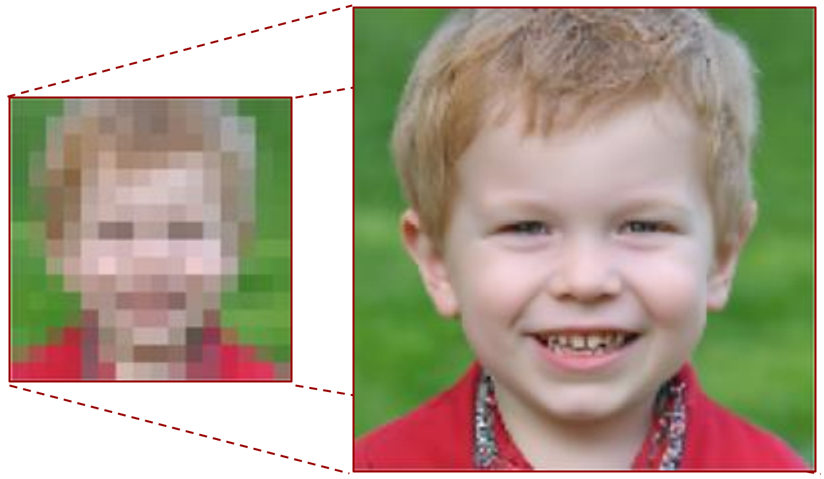


Image Super-Resolution via Iterative Refinement

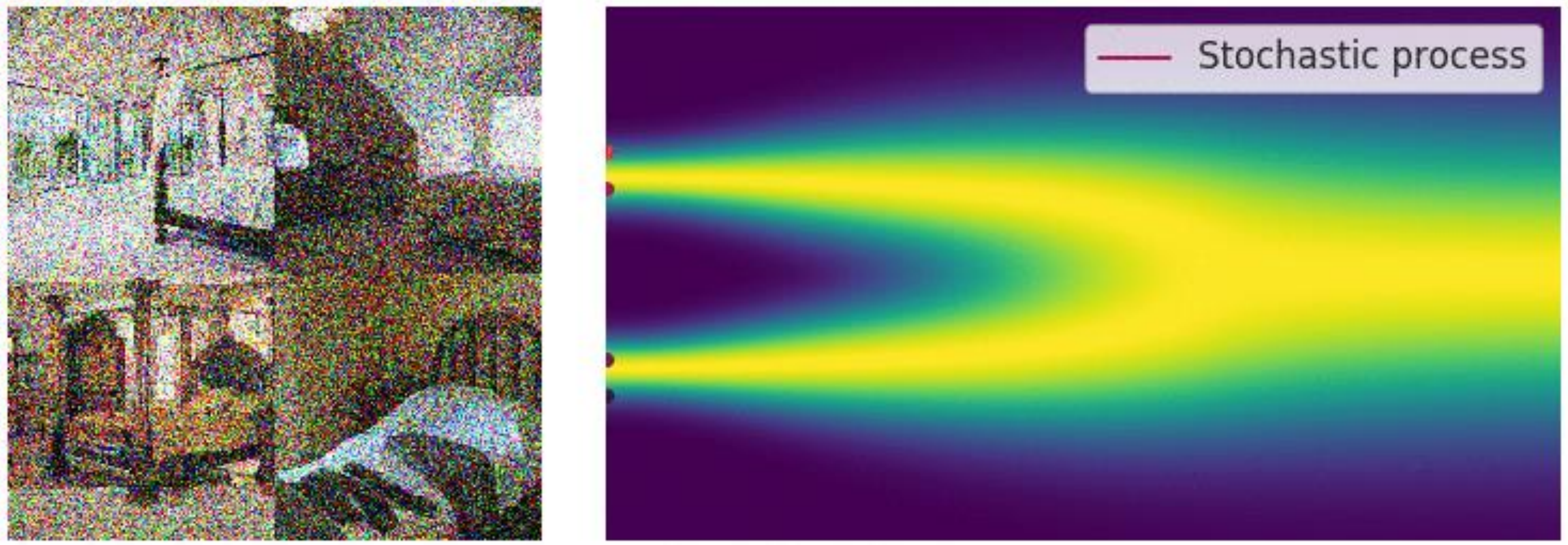
Chitwan Saharia, et al

https://arxiv.org/abs/2104.07636

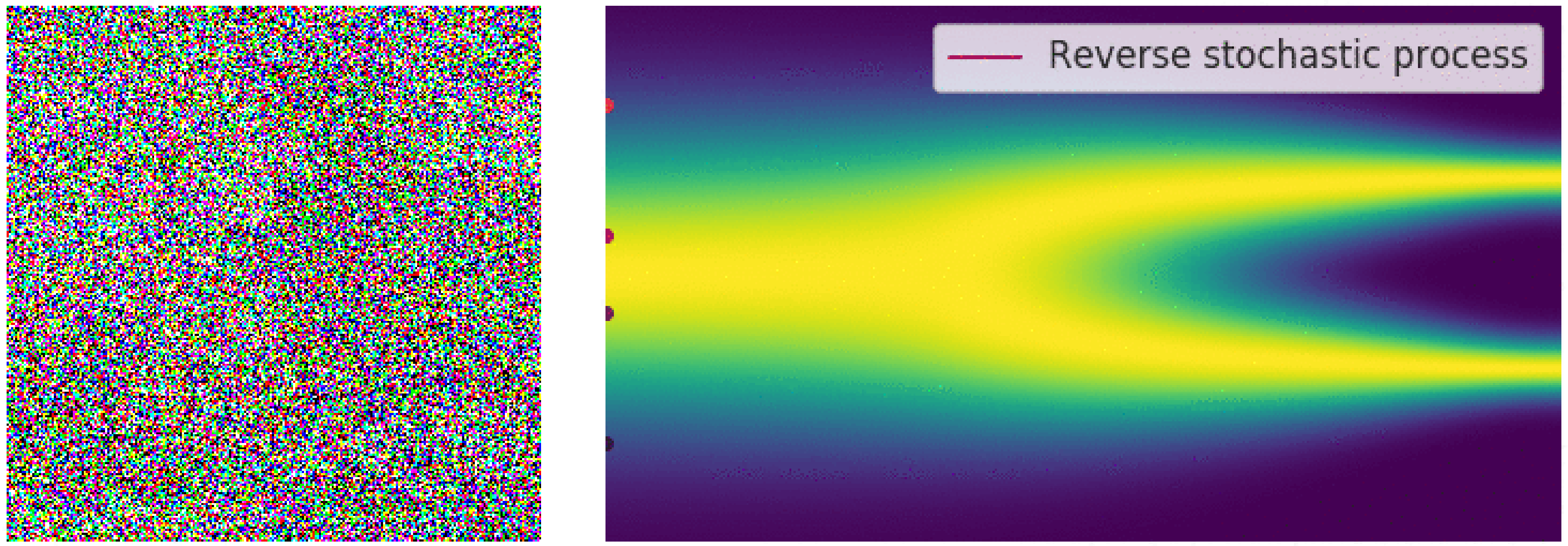
… but very expensive :(

Slide credit: Robin Rombach

Brief Overview of Diffusion Models



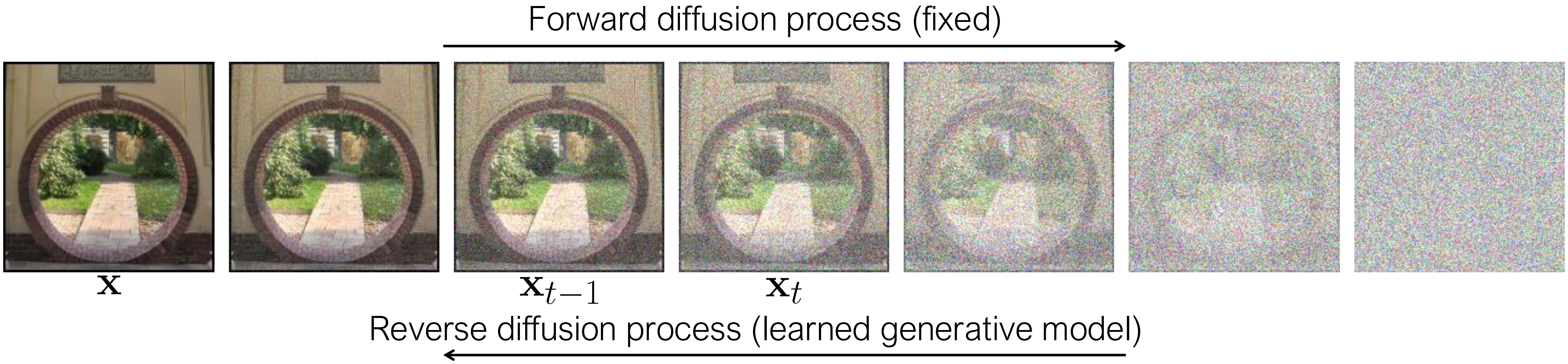
- “destroy” the data by gradually adding small amounts of gaussian noise



- “create” data by gradually denoising a noisy code from a stationary distribution

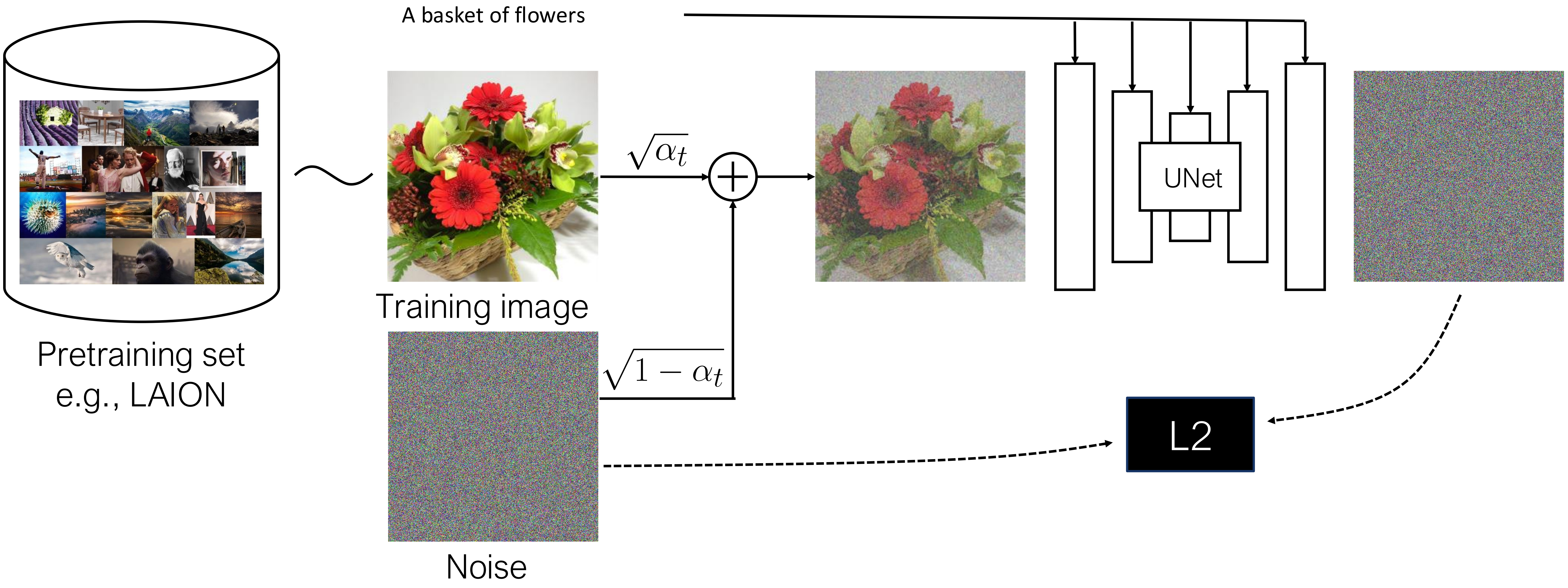
Animations from https://yang-song.github.io/blog/2021/score/

Diffusion model inference



*slides motivated from https://cvpr2022-tutorial-diffusion-models.github.io

Diffusion model training



*slides credit: from custom-diffusion

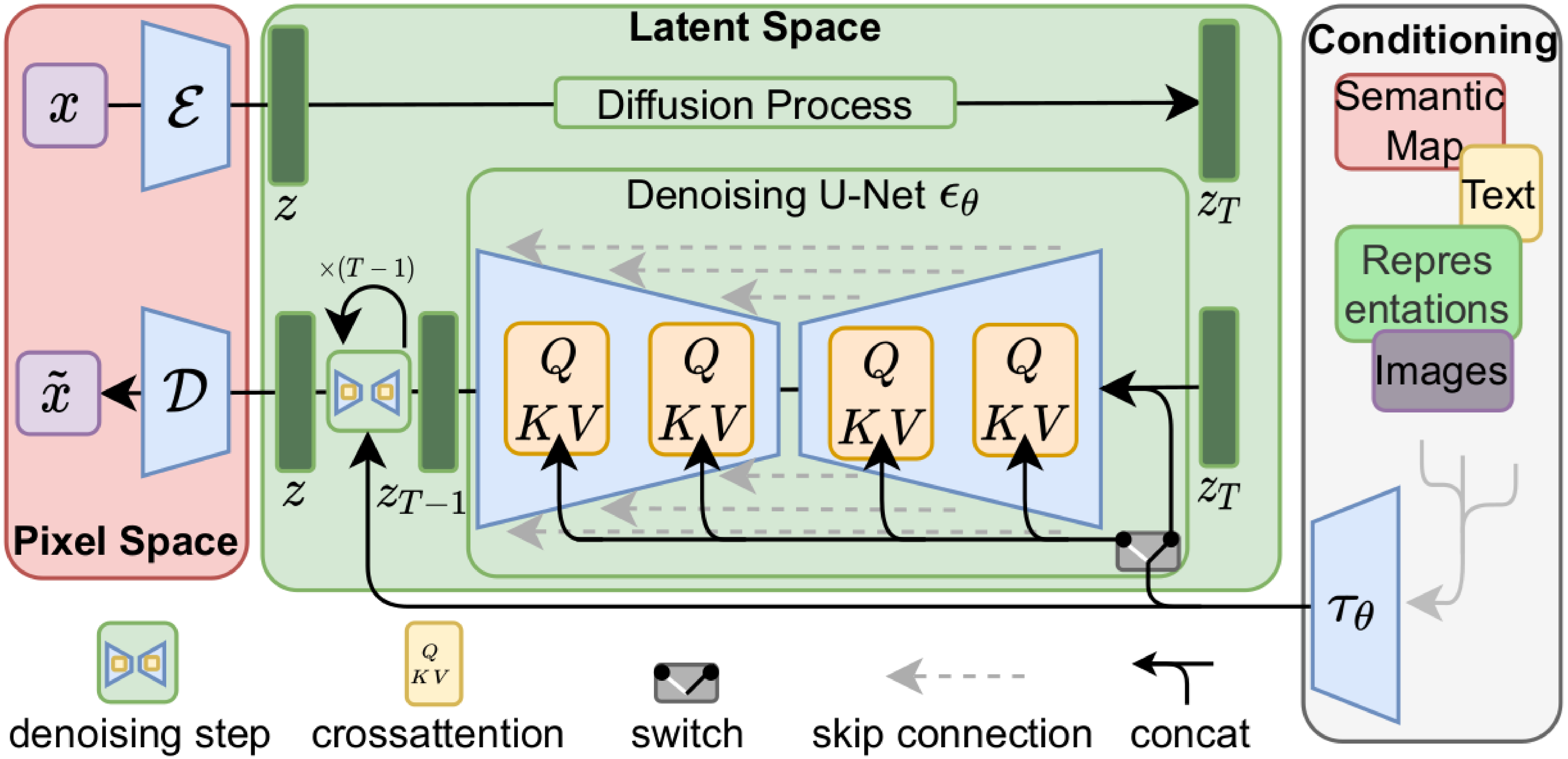
Latent Diffusion Modeling: Architecture

Autoencoder with KL or VQ regularization.

VQ-reg.: $\mathcal{L}_{\text{total}} = \mathcal{L}_{rec} + \mathcal{L}_{VQ} + \lambda \mathcal{L}_{GAN}$

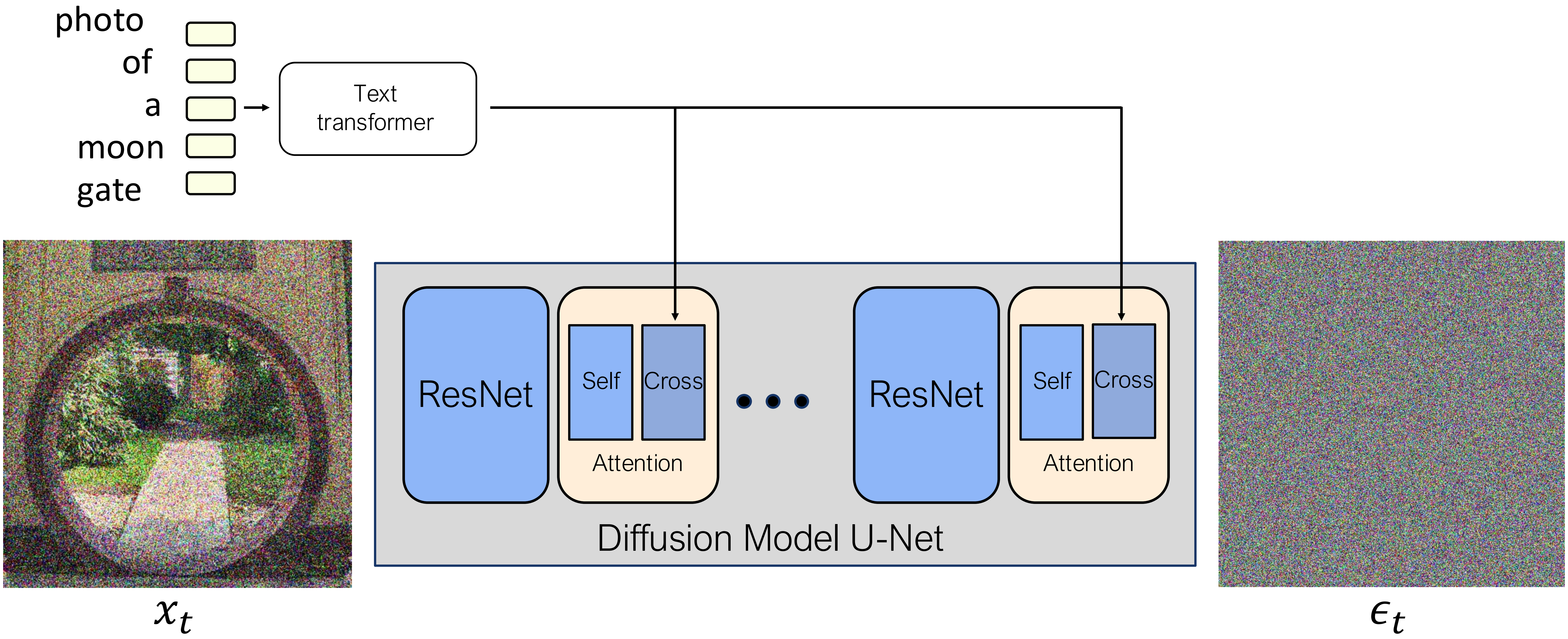
KL-reg.: $\mathcal{L}_{\text{total}} = \mathcal{L}_{rec} + \beta \mathcal{L}_{KL} + \lambda \mathcal{L}_{GAN}$

where $\lambda = \frac{\nabla_{G_L}[\mathcal{L}_{rec}]}{\nabla_{G_L}[\mathcal{L}_{GAN}]+\delta}$



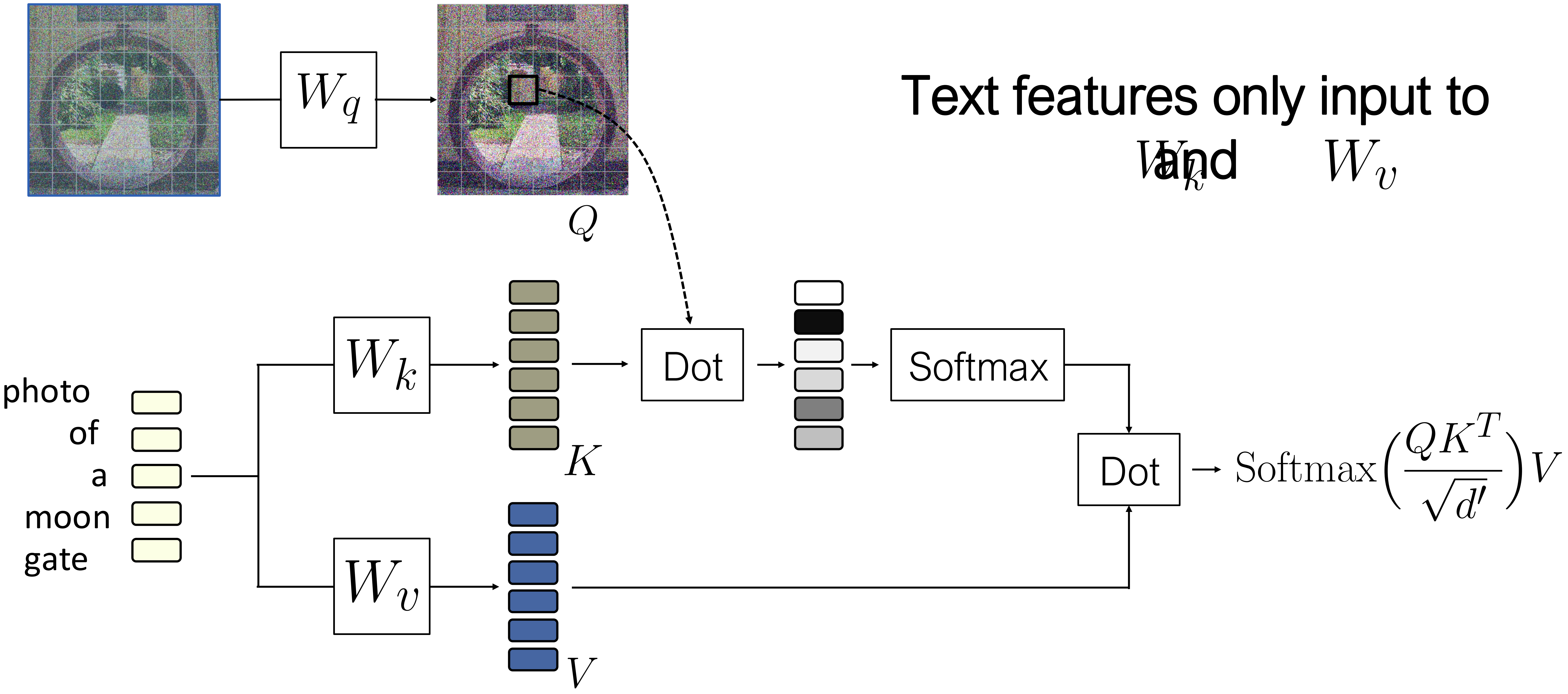
Slide credit: Robin Rombach

Diffusion Model Architecture



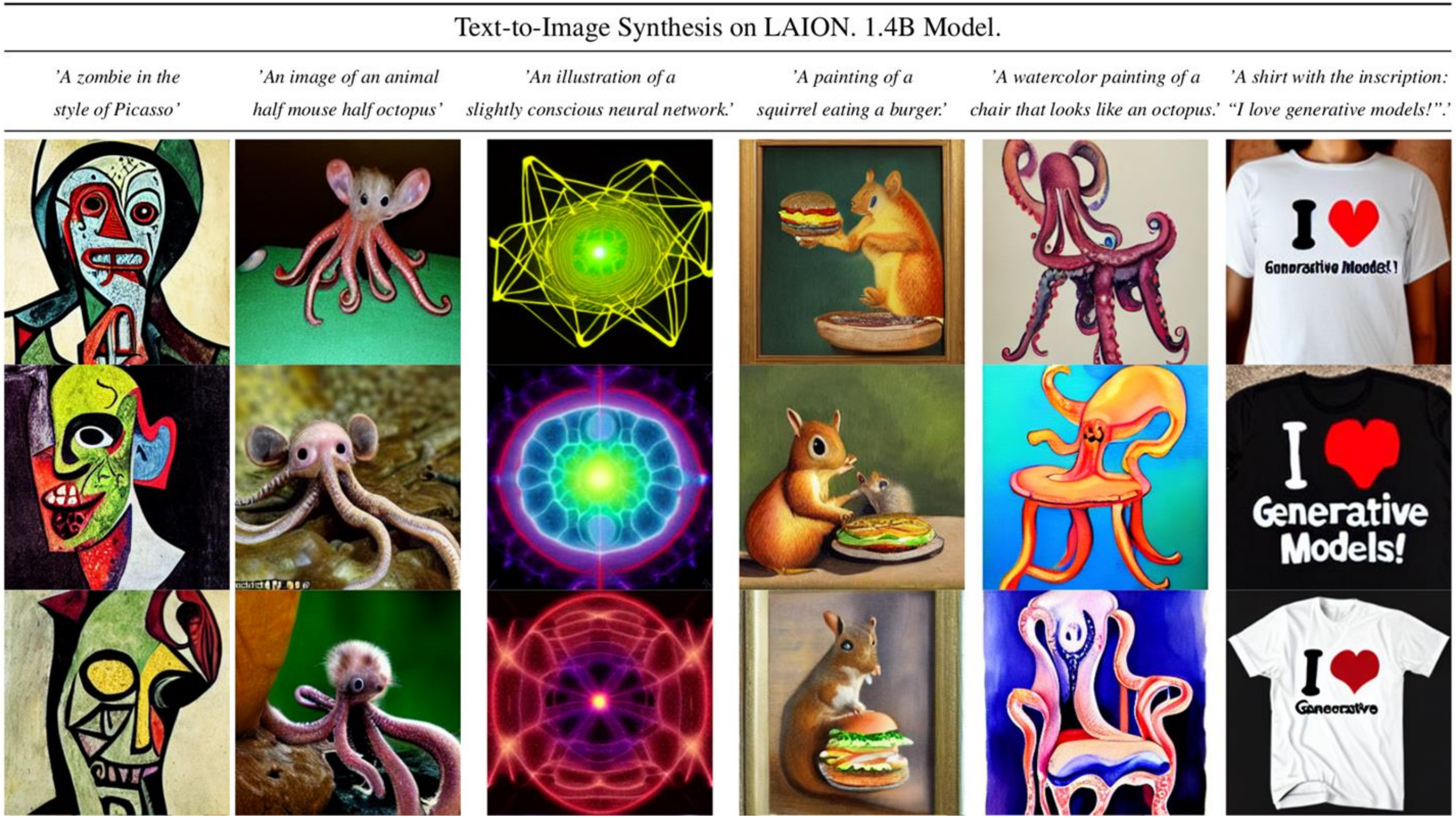
Text-to-Image Cross-Attention

Text features only input to W_k and W_v



LDMs for Text-to-Image Synthesis

- 32x32 cont. space
- 600M Transformer
- 800M UNet
- 400M Image/Text Pairs

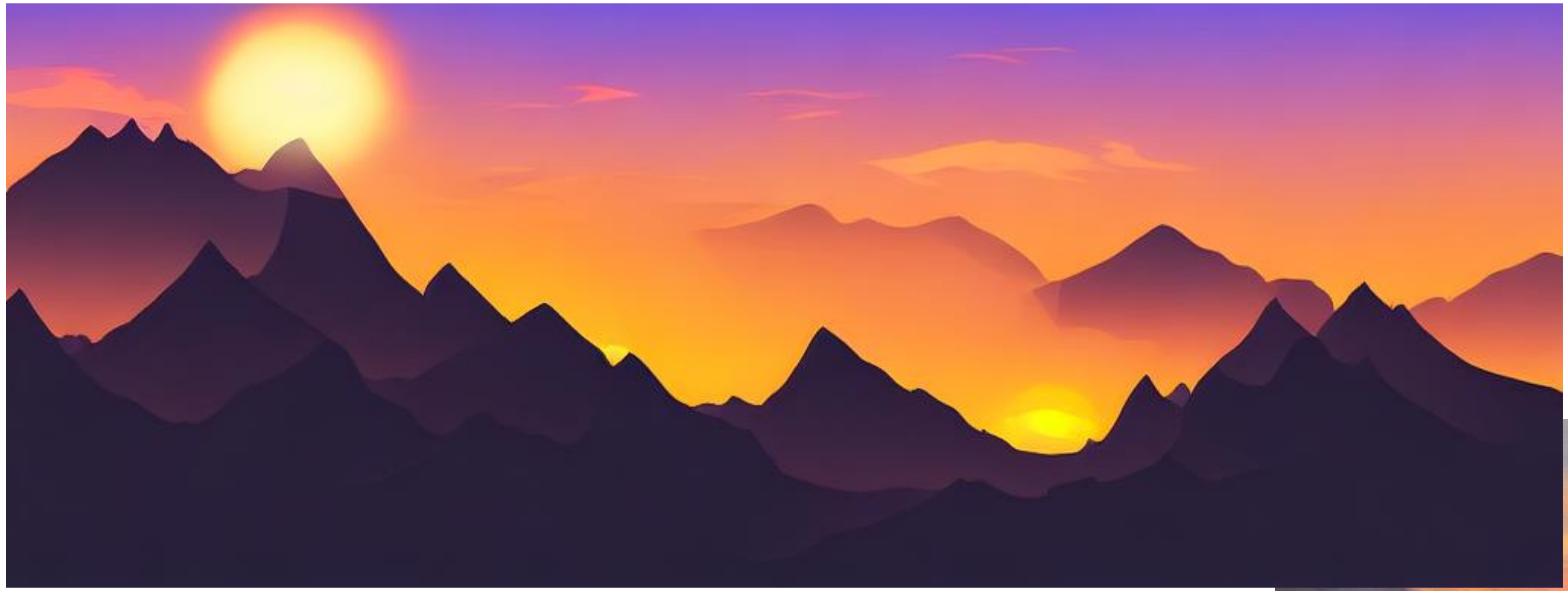


Slide credit: Robin Rombach

LDMs for Text-to-Image Synthesis

convolutional sampling (train on 256^2, generate on $>256^2$)

"A sunset over a mountain range, vector image"



"A sunset over a mountain range, oil on canvas"

Slide credit: Robin Rombach

“Cheat Code”: Classifier-Free Diffusion Guidance

Jonathan Ho, Tim Salimans

- see https://arxiv.org/abs/2207.12598
- works very well for conditional image generation:

Constant Embedding

$$\hat{\epsilon}_\theta(x_t; y, t) \leftarrow \epsilon_\theta(x_t; \emptyset, t) + s \cdot (\epsilon_\theta(x_t; y, t) - \epsilon_\theta(x_t; \emptyset, t)), \quad s \geq 1.0$$

Text Prompt

$s = 1.0$

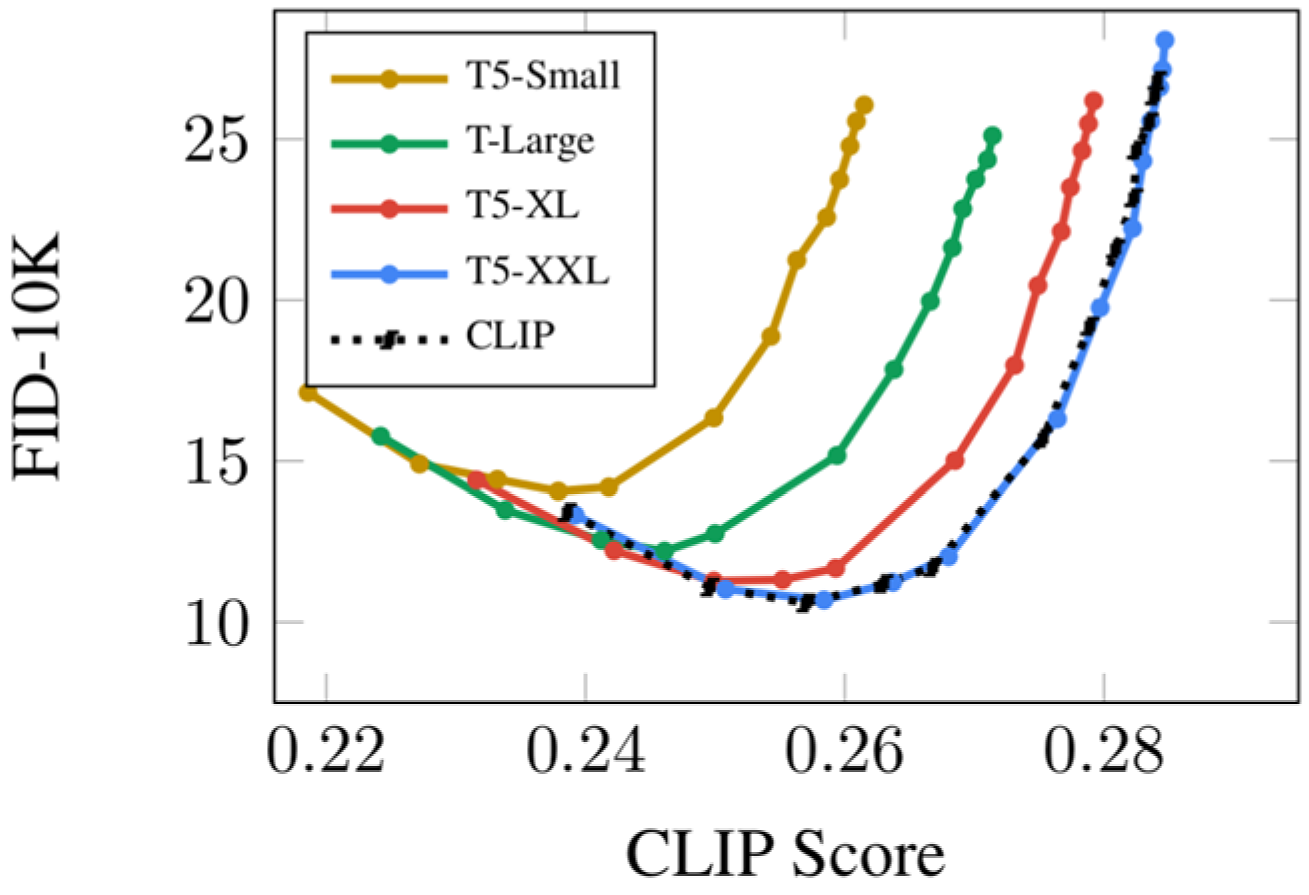
$s = 7.5$

Stable Diffusion

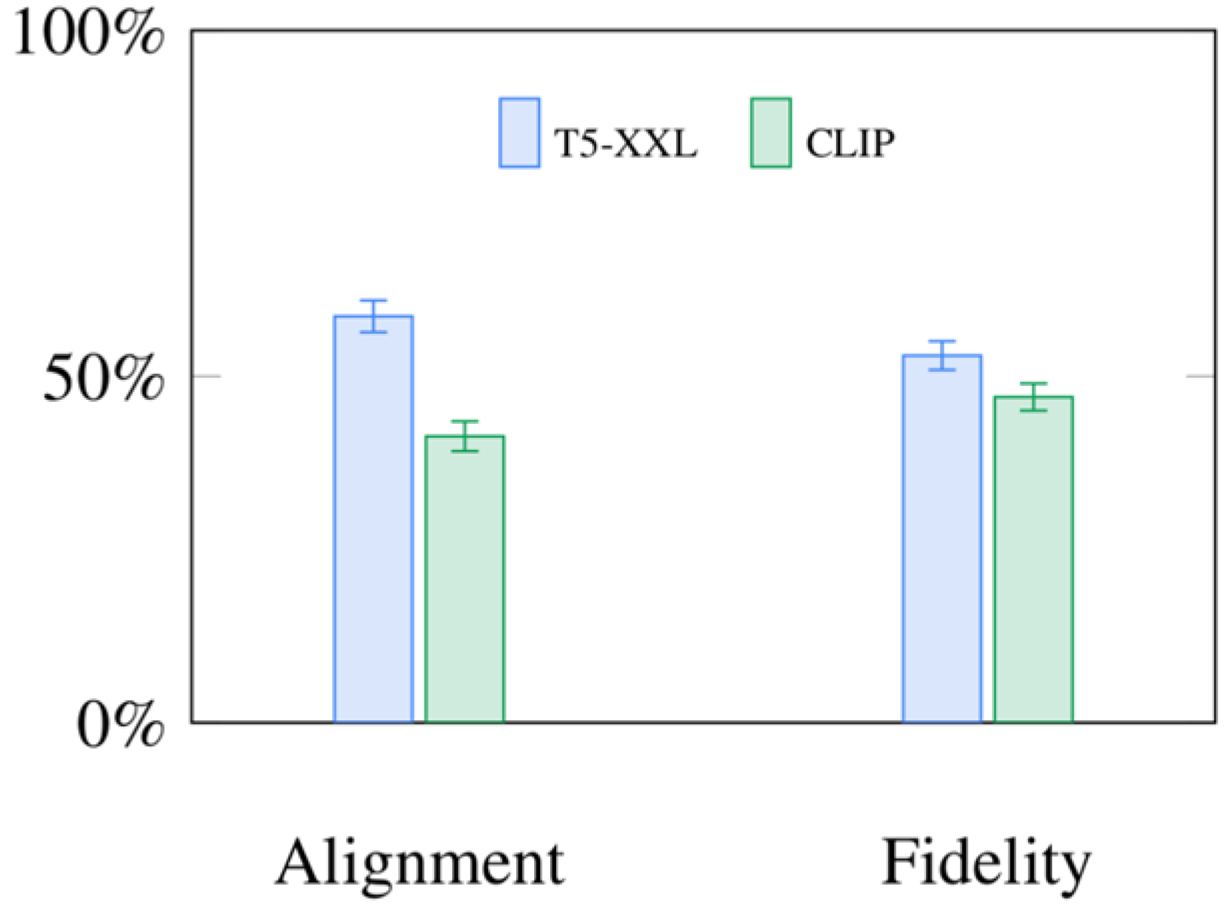
Latent Diffusion ++

From Latent to Stable Diffusion

- goal: achieve a small model that people can actually run locally on “small” GPUs (~10GB VRAM)
- progressive training: pretrain on 256x256, then continue on 512x512
- fix text encoder (as in Imagen)
- → choose CLIP (ViT-L/14) since performance/size tradeoff seems significant



(a) Pareto curves comparing various text encoders.



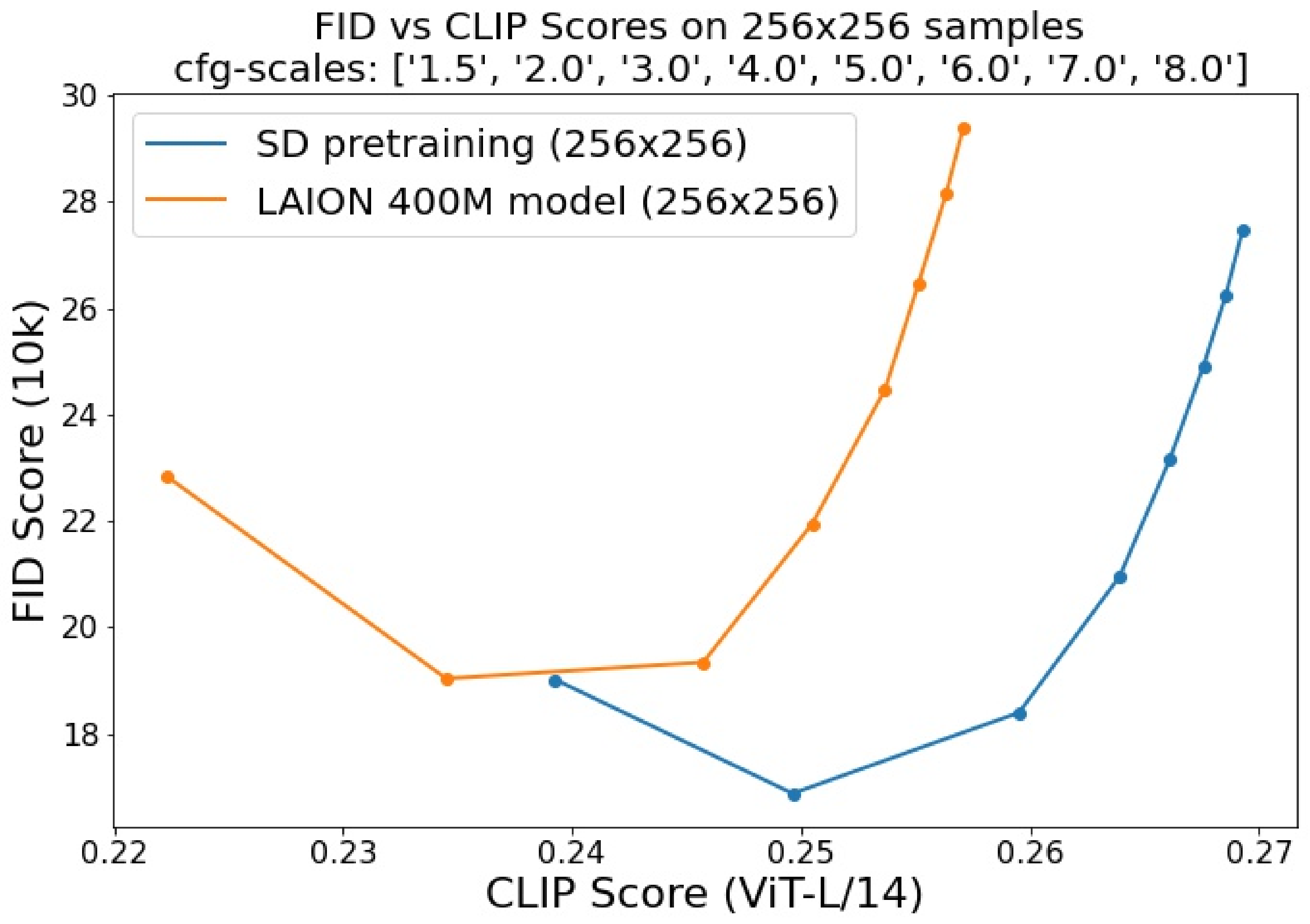
(b) Comparing T5-XXL and CLIP on DrawBench.

Figure from Imagen, https://arxiv.org/abs/2205.11487

From Latent Diffusion to Stable Diffusion

Stage 1: Pretraining @256x256

- 237k steps at resolution 256x256 on LAION 2B(en)
- batch-size = 2048
- ~ 64 A100 GPUs



10k random COCO val captions / 50 decoding steps

From Latent Diffusion to Stable Diffusion

Stage 2: Training @512x512. batch-size=2048, #gpus=256

part 1 (v1.1):

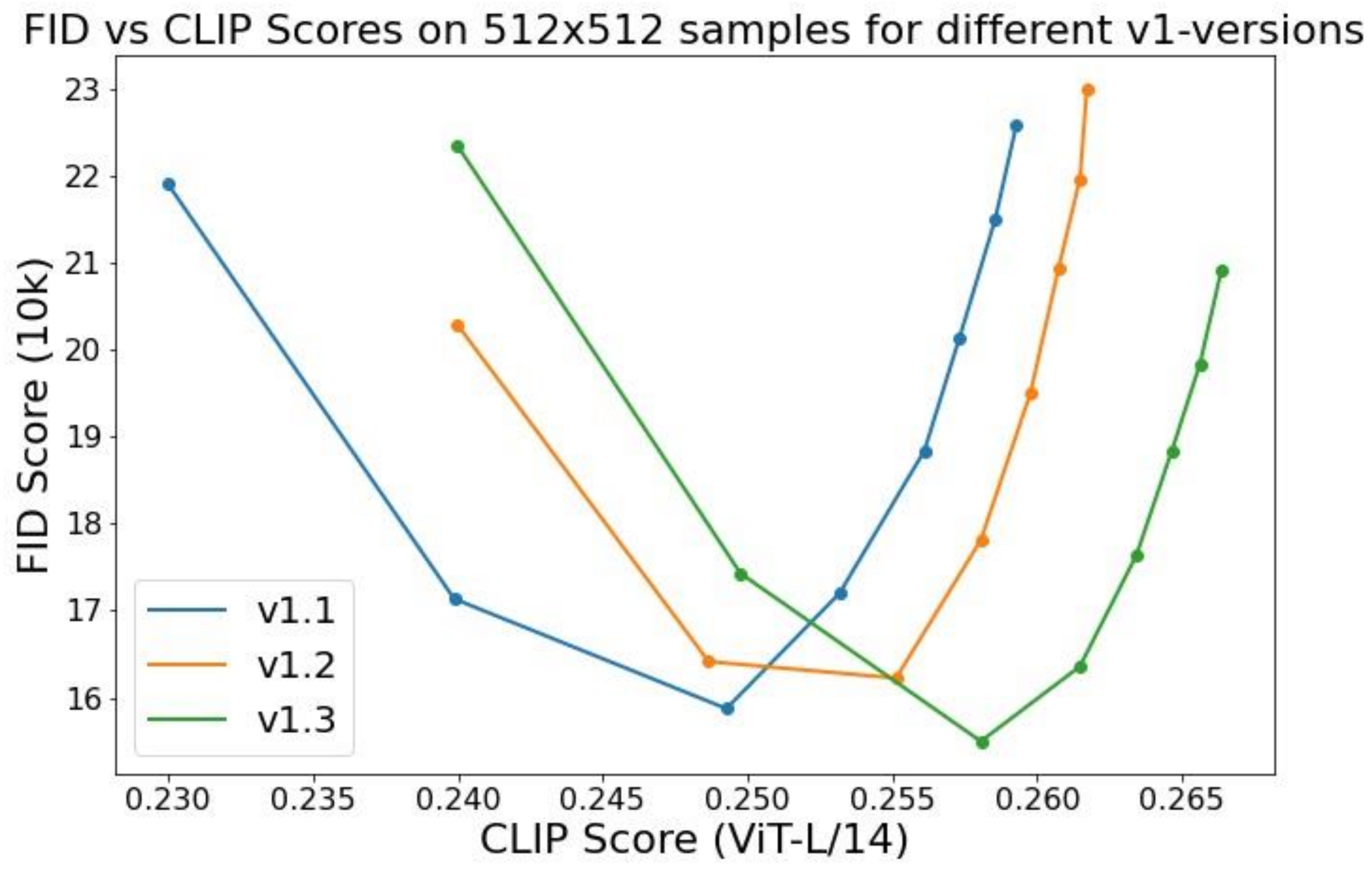
- 194k steps at resolution 512x512 on laion-high-resolution (170M examples from LAION-5B with resolution >= 1024x1024).

part 2 (v1.2):

- 515k steps at resolution 512x512 on "laion-improved-aesthetics" (a subset of laion2B-en, filtered to images with an original size >= 512x512, estimated aesthetics score > 5.0, and an estimated watermark probability < 0.5

part 3/4 (v1.3/v1.4):

- 195k/225k steps at resolution 512x512 on "laion-improved-aesthetics" and 10% dropping of the text-conditioning



10k random COCO val captions / 50 decoding steps

→ 4.2 GB checkpoint (EMA only, fp32)







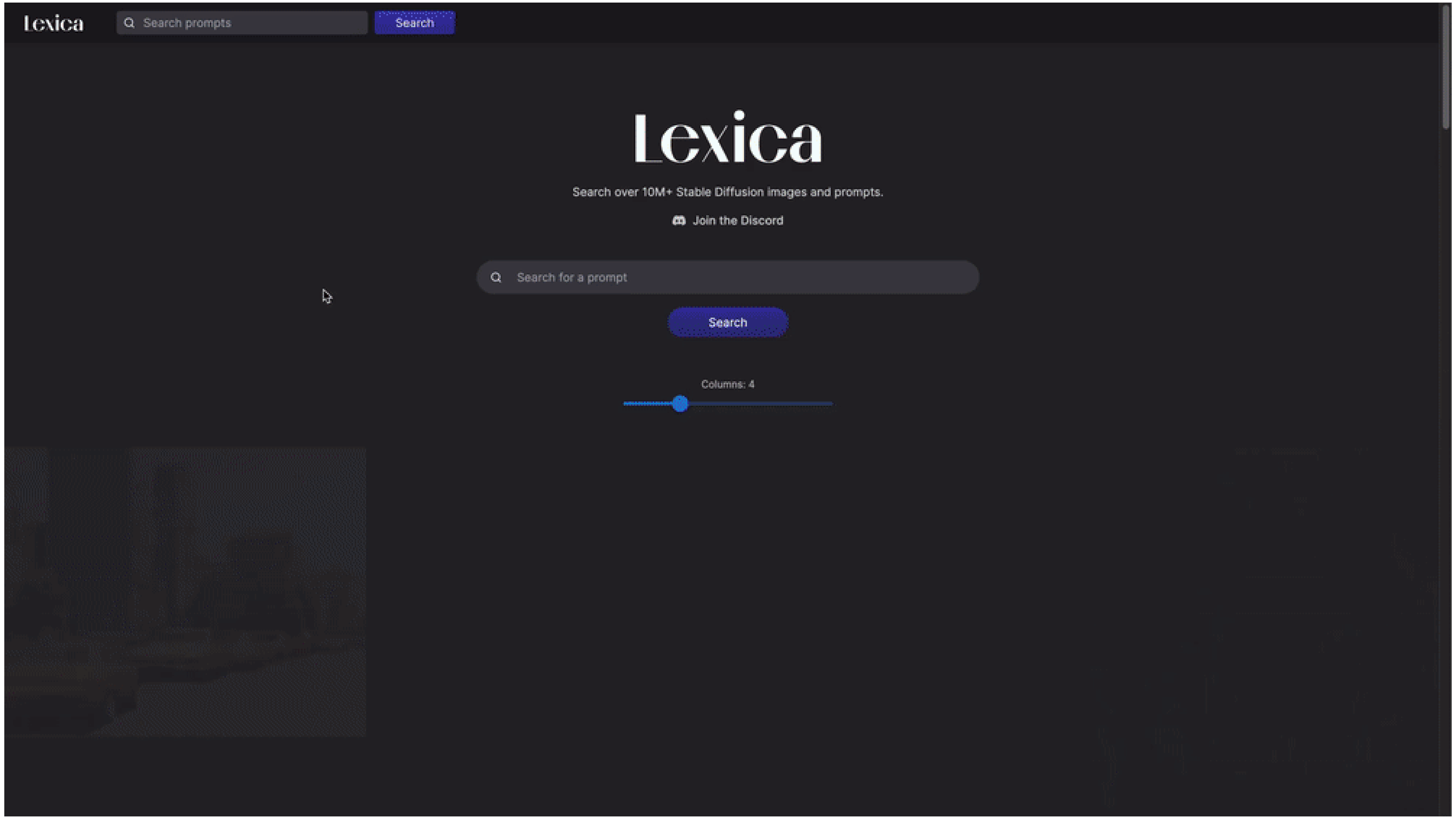

Slide credit: Robin Rombach

Video Synthesis

Stable Diffusion (img2img) + EBSynth by Scott Lightsier:
https://twitter.com/LighthiserScott/status/1567355079228887041?t=kXXCAVtuO5lJCGcro3Ma3A&s=19

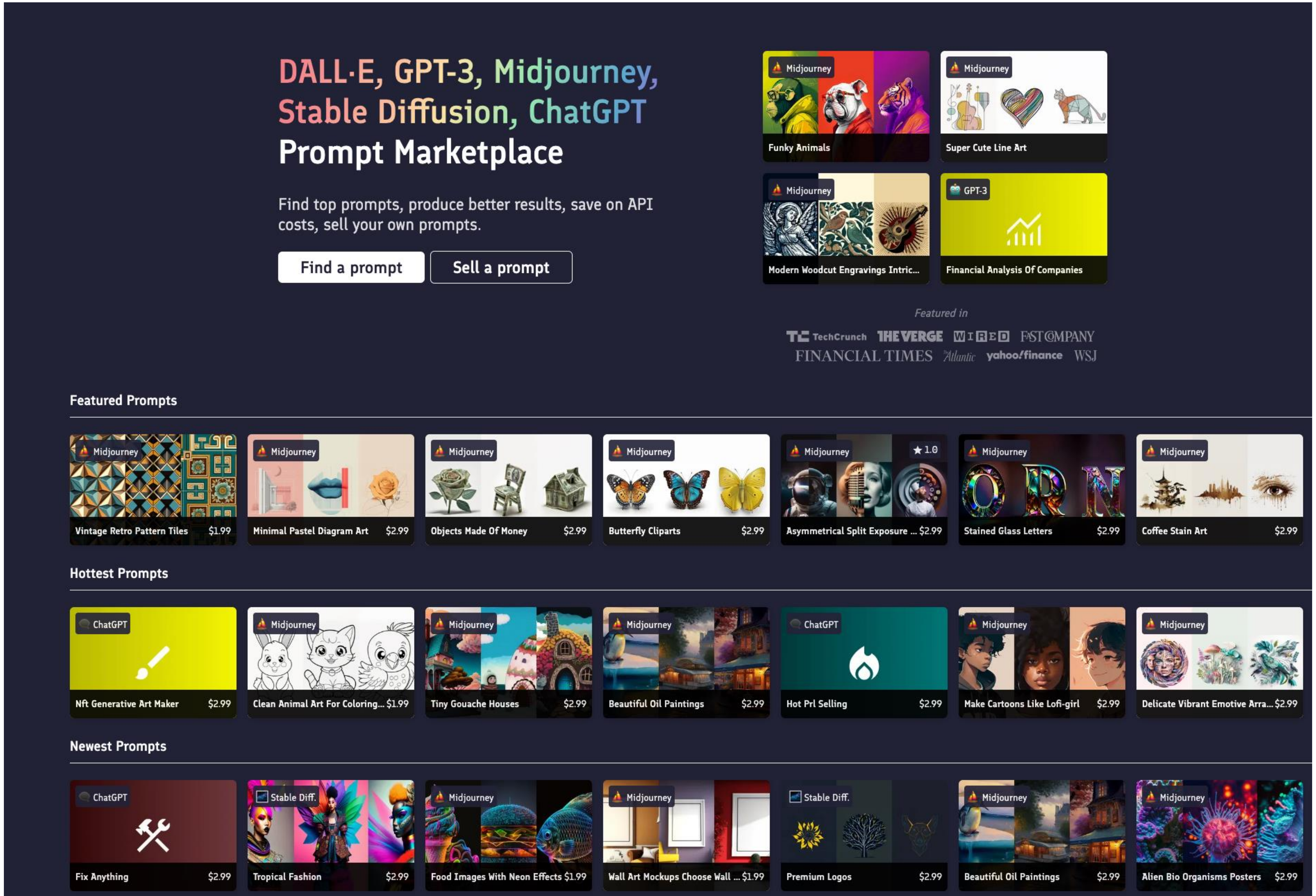
EBSynth: single-frame video stylization app: https://ebsynth.com/

Prompt Search Engine (lexica.art)

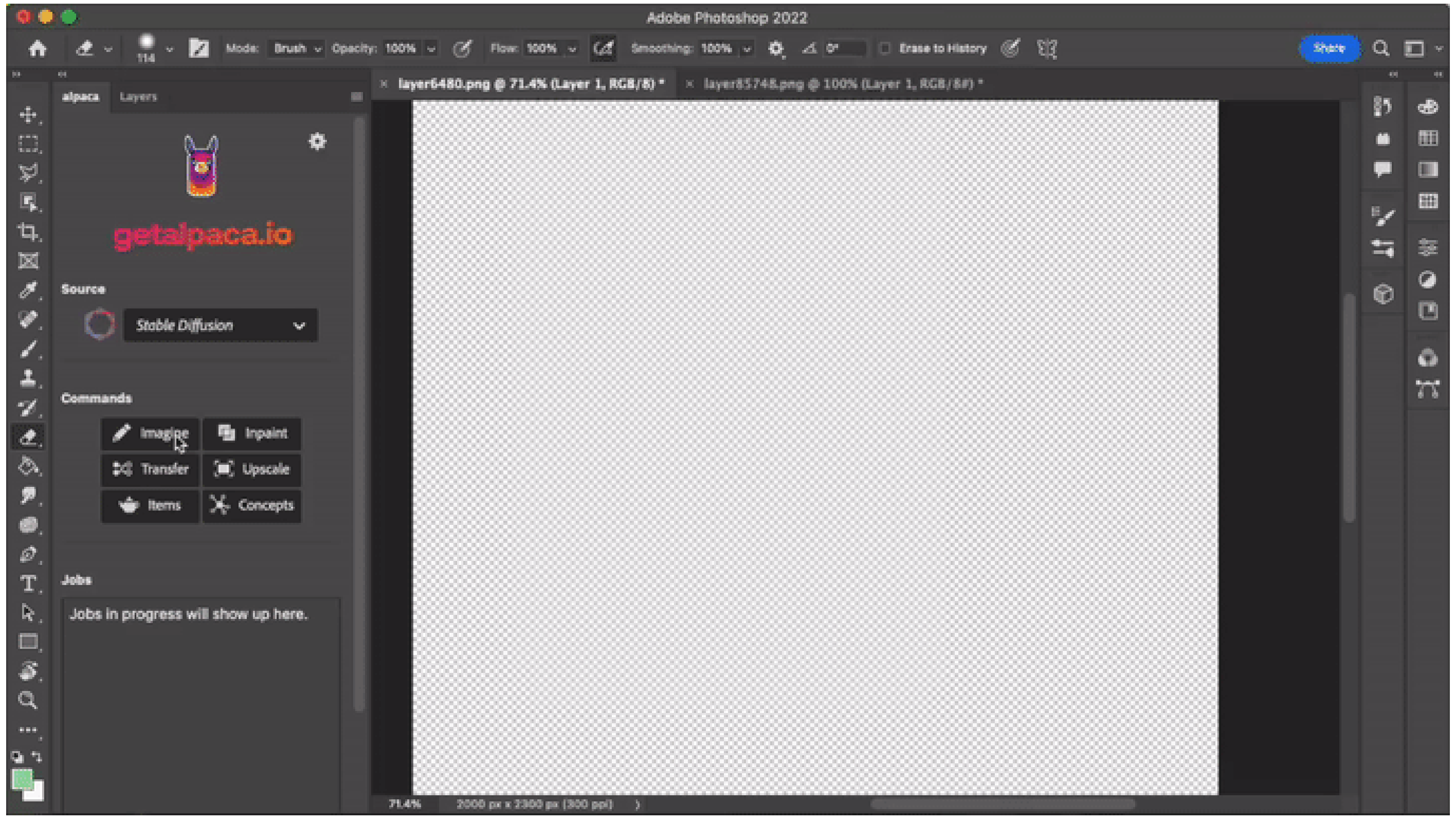


Slide credit: Robin Rombach

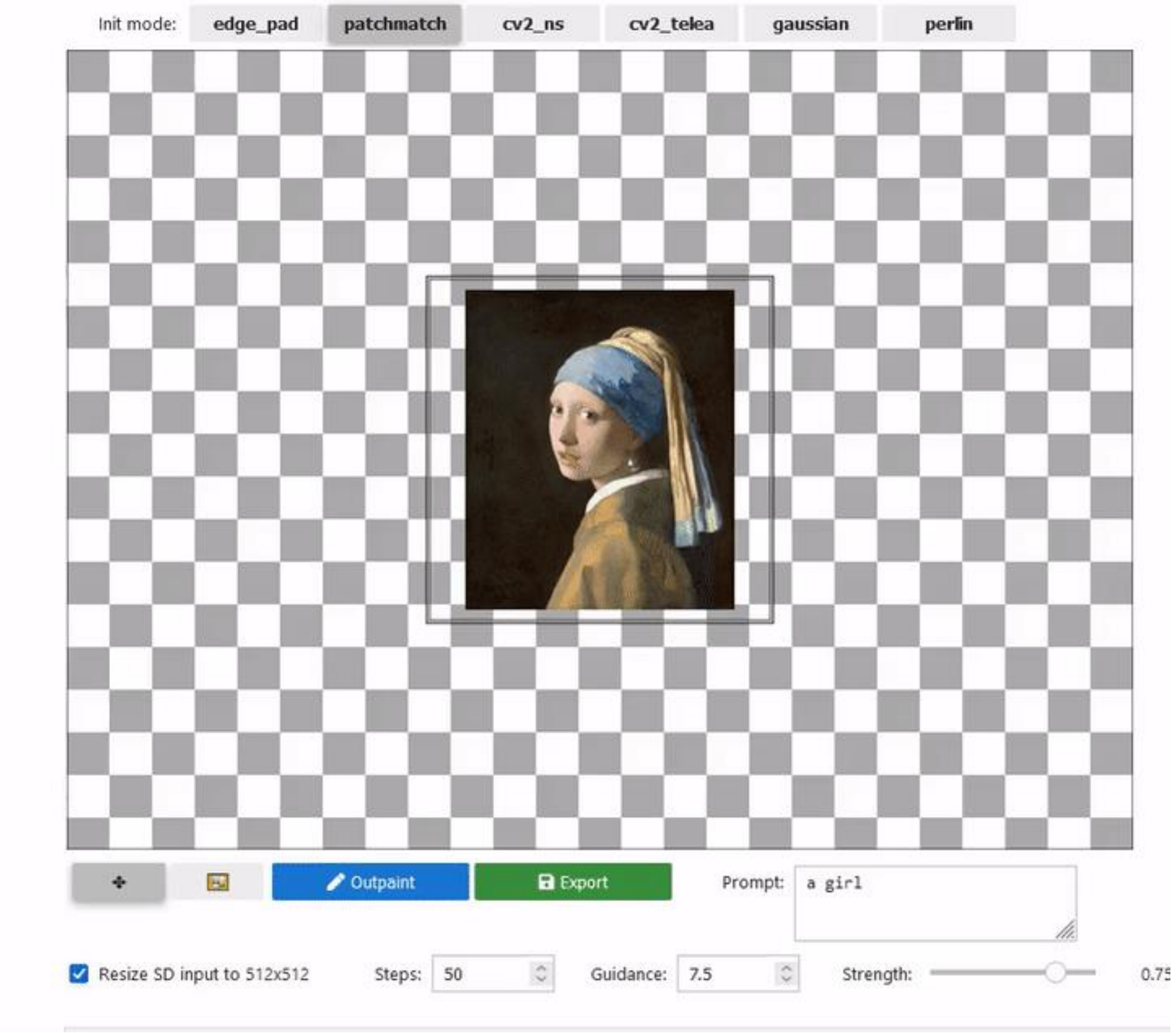
Prompt Marketplace (promptbase.com)



UIs / Plug-Ins for Photoshop, GIMP etc



https://twitter.com/wbuchw/status/1563162131024920576



https://github.com/lkwq007/stablediffusion-infinity

Slide credit: Robin Rombach

What if you have 1,000+ GPUs/TPUs

DALL·E 2, Imagen

vibrant portrait painting of Salvador Dalí with a robotic half face; a shiba inu wearing a beret and black turtleneck; a close up of a handpalm with leaves growing from it; an espresso machine that makes coffee from human souls, artstation; panda mad scientist mixing sparkling chemicals, artstation; a corgi's head depicted as an explosion of a nebula

Sprouts in the shape of text 'Imagen' coming out of a fairytale book. A photo of a Shiba Inu dog with a backpack riding a bike. It is wearing sunglasses and a beach hat. A high contrast portrait of a very happy fuzzy panda dressed as a chef in a high end kitchen making dough. There is a painting of flowers on the wall behind him.

Teddy bears swimming at the Olympics 400m Butterfly event. A cute corgi lives in a house made out of sushi. A cute sloth holding a small treasure chest. A bright golden glow is coming from the chest.

- Pixel-based Diffusion (No encoder-decoder)
- pre-trained text encoder (CLIP, t5)
- Diffusion model + classifier-free guidance
- Cascaded models: 64->128->512

https://cdn.openai.com/papers/dall-e-2.pdf
https://arxiv.org/abs/2205.11487

Diffusion vs. Autoregressive vs. GANs

GigaGAN: Scaling up GANs

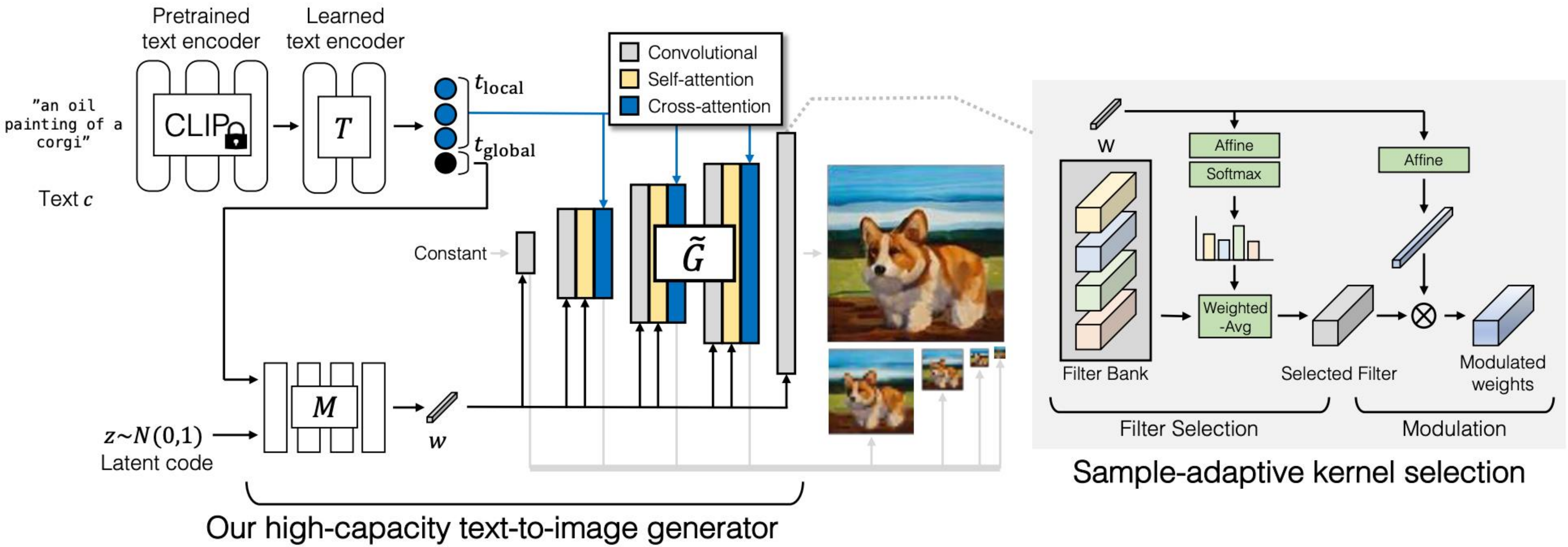
A portrait of a human growing colorful flowers from her hair. Hyperrealistic oil painting. Intricate details.

A golden luxury motorcycle parked at the King's palace. 35mm f/4.5.

a cute magical flying maltipoo at light speed, fantasy concept art, bokeh, wide sky

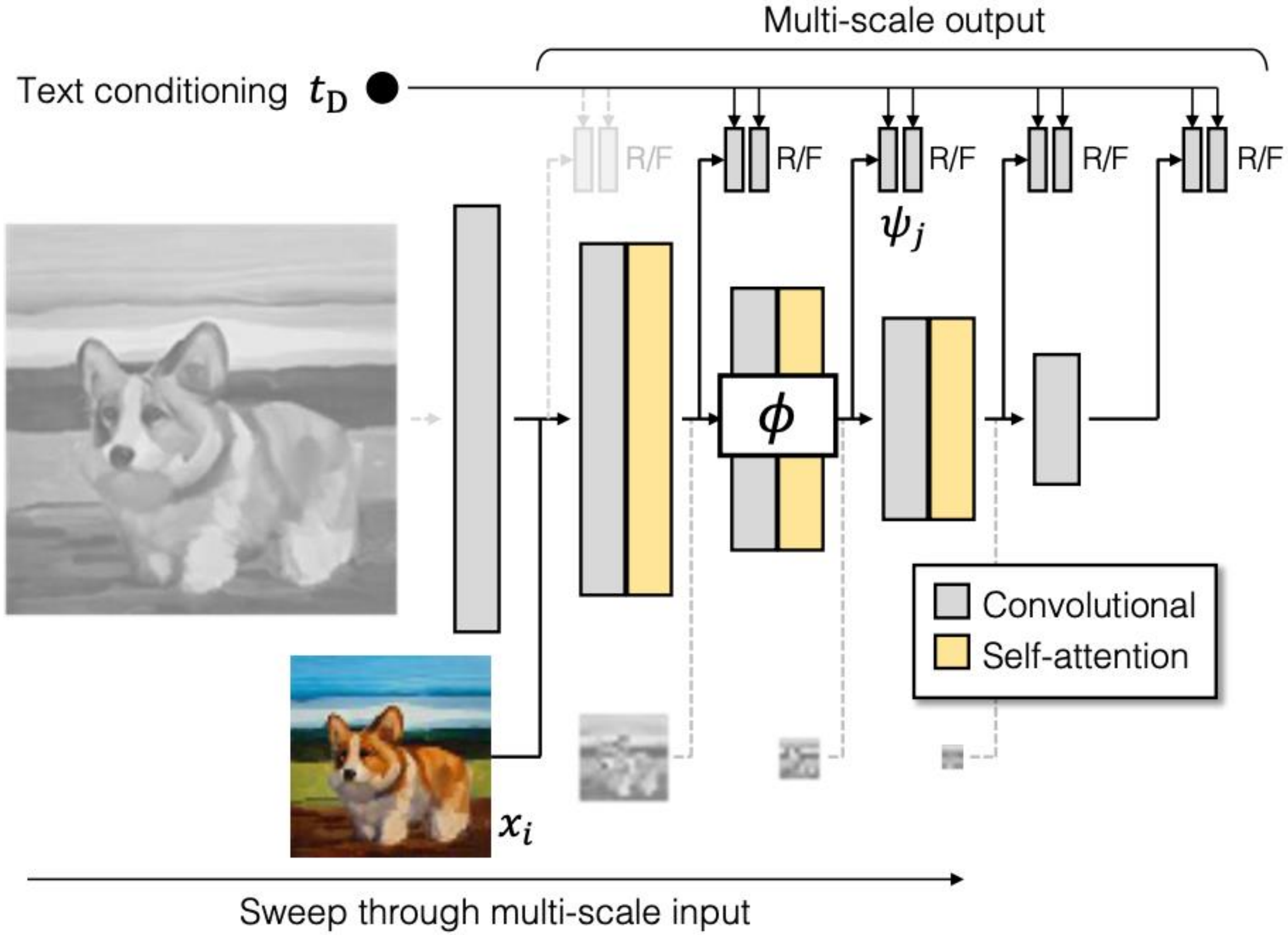
[Kang et al., CVPR 2023]

GigaGAN Generator



[Kang et al., CVPR 2023]

GigaGAN Discriminator

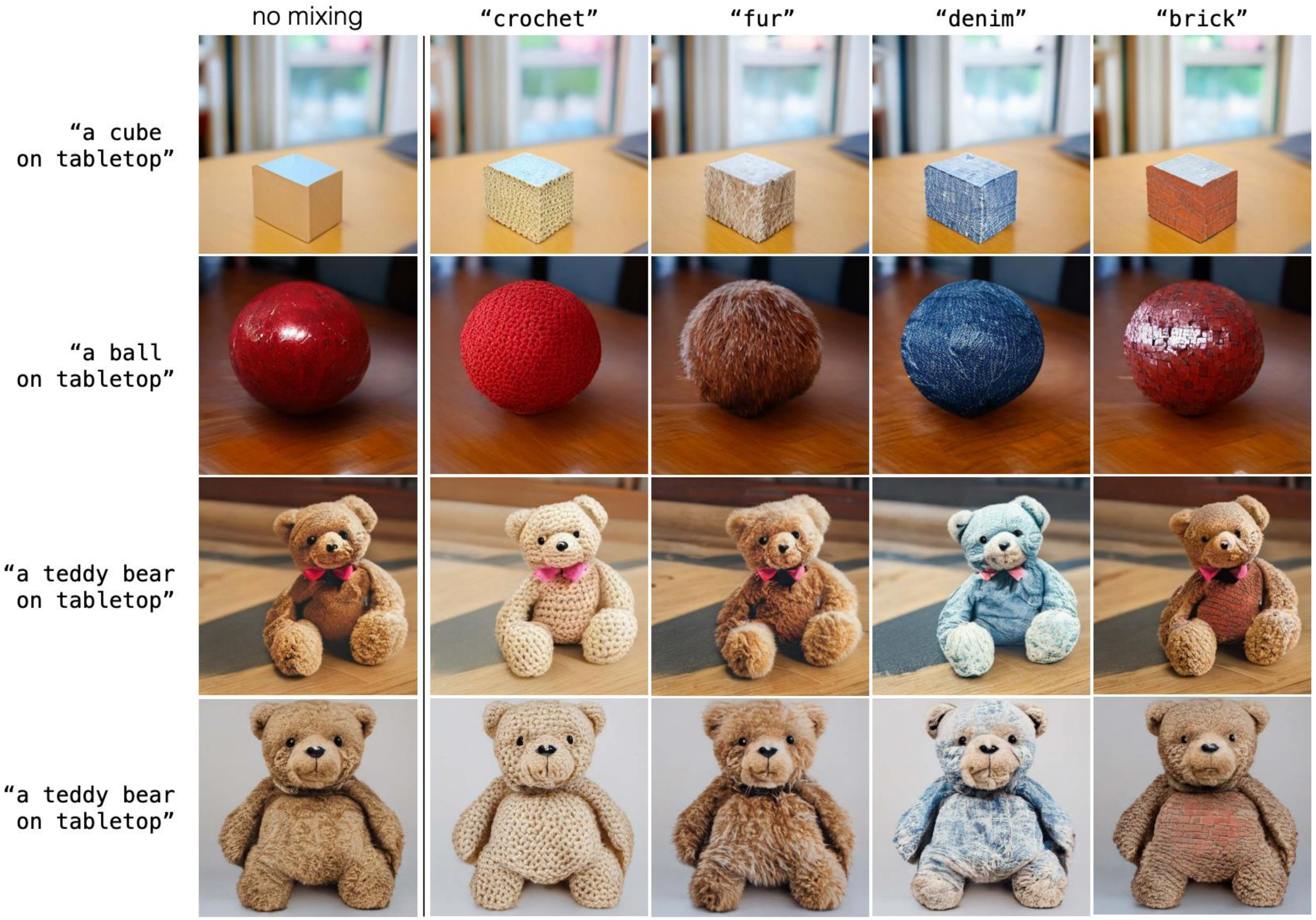


[Kang et al., CVPR 2023]

Style Mixing

[Kang et al., CVPR 2023]

Prompt Mixing



[Kang et al., CVPR 2023]

Input
Real-ESRGAN (1K)
SD Upscaler (1K)
GigaGAN Up (1K)
GigaGAN Up (4K)
GigaGAN Upsampler (4096px, 16Mpix, 3.66s)

Comparison between Different Models

Ours (512px, 0.14s / img, truncation $\psi = 0.8$)

LDM (256px, 9.4s / img, 250 steps, guidance=6.0)

Stable Diffusion v1.5 (512px, 2.9s / img, 50 steps, guidance=7.5)

DALL·E 2 (1024px)

Ours (512px, 0.14s / img, truncation $\psi = 0.8$)

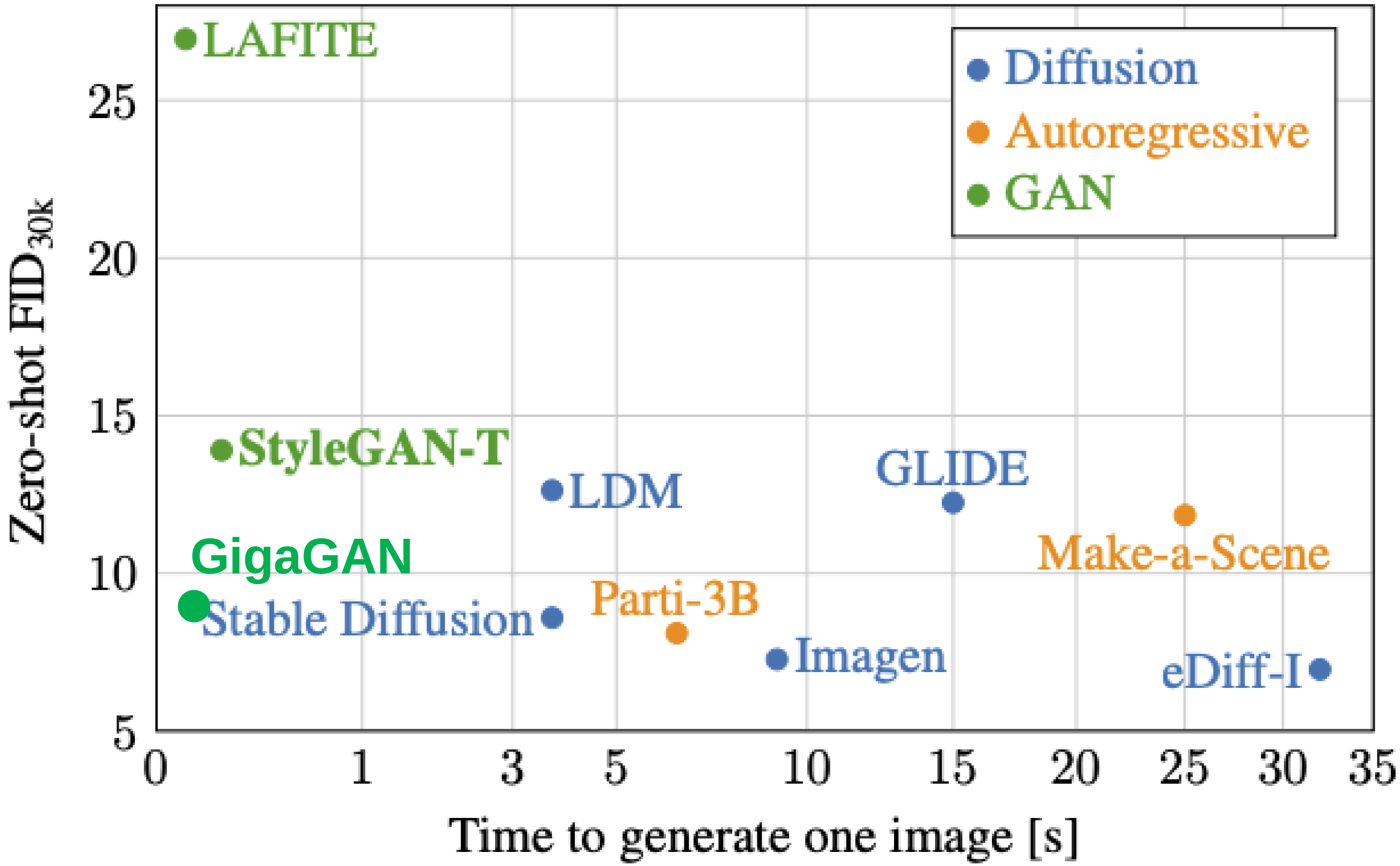
LDM (256px, 9.4s / img, 250 steps, guidance=6.0)

Stable Diffusion v1.5 (512px, 2.9s / img, 50 steps, guidance=7.5)

DALL·E 2 (1024px)

[Kang et al., CVPR 2023]

StyleGAN-T



[Sauer et al., ArXiv 2023]

How could we improve it?

- Better generative modeling techniques: VAEs, GANs, diffusion, AR, Hybrid
- Better text encoders: RNN/LSTM -> Transformers (CLIP, T5)
- Better generator architectures: RNN/LSTM -> CNN -> CNN + Transformer
- Better ways to connect text and image: concatenation -> AdaIN -> cross-attention
- More data + GPU/TPU computing: a few hundred A100.
- Bigger model sizes: 1B-20B.