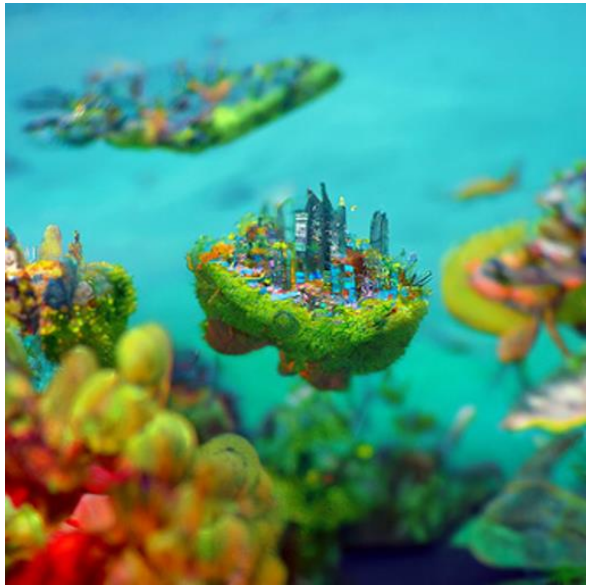
Neural Networks:
Tools and Principles Part 2

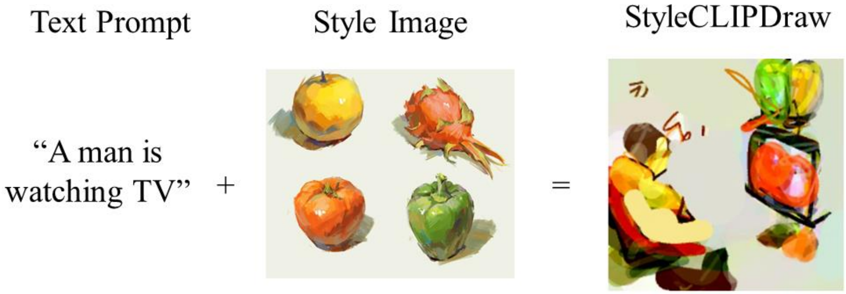
Image Generation Models

Intelligent Computational Media, Spring 2022

Nam Hee Gordon Kim



“Coral reef city by artists on ArtStation”



Announcements

- Nam Hee's Office Hours: 4pm-5pm Thursday, 1pm-2pm Friday (Zoom or T-Talo B133)
  - Come chat about assignments/project
- Please do your exercises!

Week 1 (Tue 01.02 - Fri 04.02.)

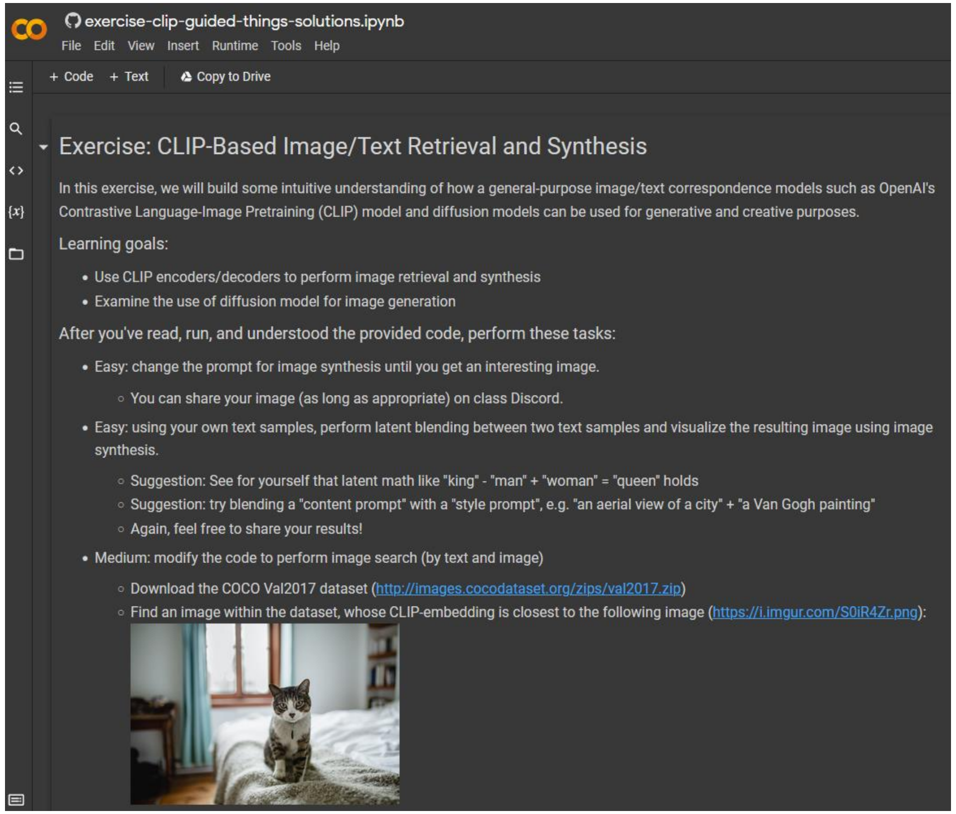
Day 1 / Tue (Overview and Motivation — Perttu Hämäläinen):

- Introduction: Each student adds a slide in a shared Google Slides document: What's your background, what do you want to learn? This will help students to find teammates and teachers to customize the course contents
- Lecture: Overview and motivation. Why one should rather co-create than compete with AI technology.
- Exercise: Introduction to tensors, numpy and matplotlib through processing images and audio. [Open in Colab], [Solutions]
- Related to the above, see also: https://numpy.org/devdocs/user/absolute_beginners.html
- Exercise: Continuing the Numpy introduction, now for a simple data science project. [Open in Colab], [Solutions].
- Exercise: Training a very simple neural network using a Kaggle dataset of human height and weight. [Open in Colab], [Solutions]

Day 2 / Wed (Neural Networks Basics — Nam Hee Gordon Kim):

- Lecture: Neural Networks: Tools and Principles, Part 1. Basics and standard techniques of neural networks. Introduction to convolutional neural networks, autoencoders, and transfer learning.
- Perttu's Old Slides, Part 1
- Demos:
  - Demo 1: Inference
  - Demo 2: Training
  - Demo 3: Denoising CNN Autoencoder
- Exercise: Image classification (optional). [Open in Colab], [Solutions]
- Exercise: Using a Pre-Trained Generative Adversarial Network (GAN) to generate images. [Open in Colab], [Solutions]

Today’s Exercise



You can start doing the exercises right now!

TL;DR

- Correspondence: match different media of data by concept (e.g. text description vs. image depiction)
  - Major advances in recent years: DALL-E, CLIP
- Generative adversarial networks (GANs): learn to generate e.g. "believable" images, and learn to fuse latents for style transfer
- Diffusion models: slow but high-fidelity generative models, becoming more popular

Learning Goals

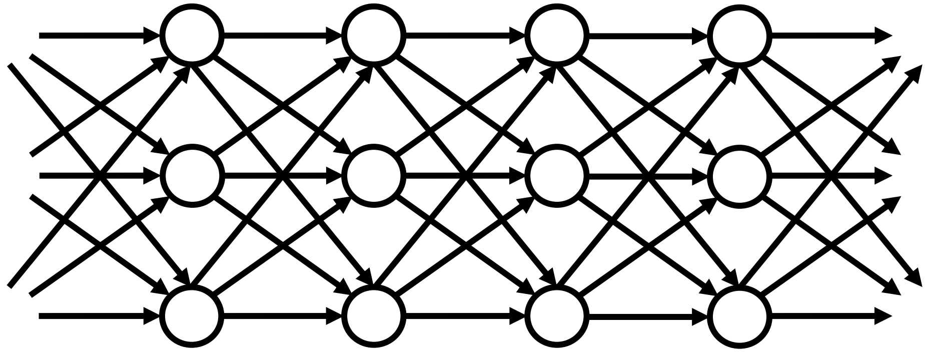
- Understand correspondence in an abstract level
- Glimpse at the high-level ideas of generative models
- Look at cool applications of generative models for visuals

Agenda

- Finishing Neural Networks Basics (10 minutes)
- Image-Text Correspondence and CLIP (5 minutes)
- GANs and Diffusion Models (10 minutes)
- Applications of GANs and Diffusion (20 minutes)

(Pause)

Last Time: Neural Networks



"Neural Network"

= Versatile data-driven input/output mappings

Generative Models Learn to Generate Data

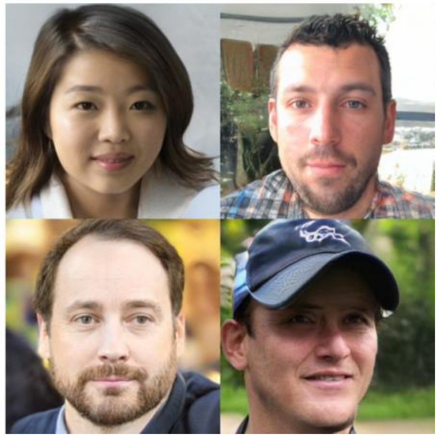
- “Generative models”: neural networks that map “noises” to images/audios/samples

$$\varepsilon \longrightarrow \hat{x}$$

noise sample

- For example, human face generator

$$\begin{matrix} \varepsilon_1 & \varepsilon_2 \\ \varepsilon_3 & \varepsilon_4 \end{matrix} \longrightarrow$$



BigGAN generating fake images of 1000 categories

[Submitted on 28 Sep 2018 (v1), last revised 25 Feb 2019 (this version, v2)]

Large Scale GAN Training for High Fidelity Natural Image Synthesis

Andrew Brock, Jeff Donahue, Karen Simonyan

Despite recent progress in generative image modeling, successfully generating high-resolution, diverse samples from complex datasets such as ImageNet remains an elusive goal. To this end, we train Generative Adversarial Networks at the largest scale yet attempted, and study the instabilities specific to such scale. We find that applying orthogonal regularization to the generator renders it amenable to a simple "truncation trick," allowing fine control over the trade-off between sample fidelity and variety by reducing the variance of the Generator's input. Our modifications lead to models which set the new state of the art in class-conditional image synthesis. When trained on ImageNet at 128x128 resolution, our models (BigGANs) achieve an Inception Score (IS) of 166.5 and Frechet Inception Distance (FID) of 7.4, improving over the previous best IS of 52.52 and FID of 18.6.

ARTBREEDER

Extend your imagination

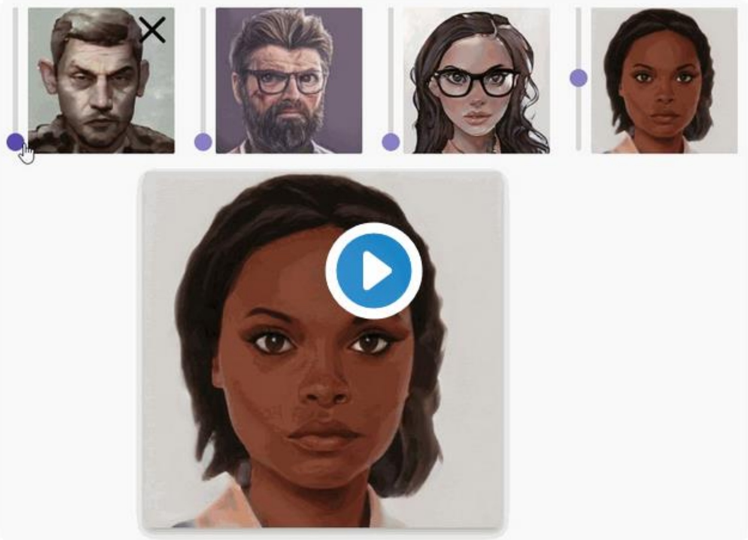
Thousands of users have collectively made 59946402 images

View Gallery | Start | Watch Intro

Bay Raitt
@bayraitt

Artbreeder is a nuclear powered pencil.

have a spin remixing some of mine here:artbreeder.com/bayraitt/starr…#artbreeder #ai #ganbreeder #conceptart #comics



♡ 8,991 2:29 AM - Sep 18, 2019

2,942 people are talking about this

Henry Lynch
@HenryLynch_Art

'New World Officer'

Character created with AI image breeding and paint-over. Fast and interesting. It is the future. @Artbreeder 👏#ConceptArt #CharacterDesign #SciFi #AI #ConceptArtist

♡ 58 5:18 AM - Oct 9, 2019

See Henry Lynch's other Tweets

TELTHONA
@telthona

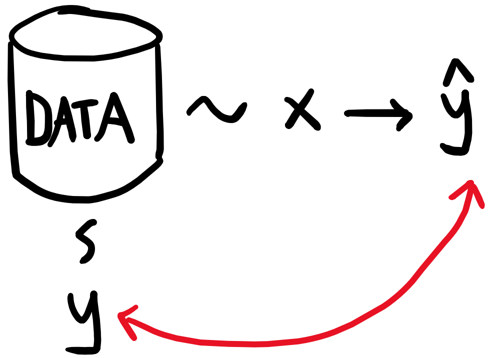
I generated bunch of concepts with new AI powered website - it's the best AI app that i tried so far! It's amaizng for creature exploration, mood thumbnailing and more ❤️ finally AI that i can truly use in the creative process! ganbreeder.app #Aimakesart #ganbreeder

♡ 69 8:41 PM - Jun 29, 2019

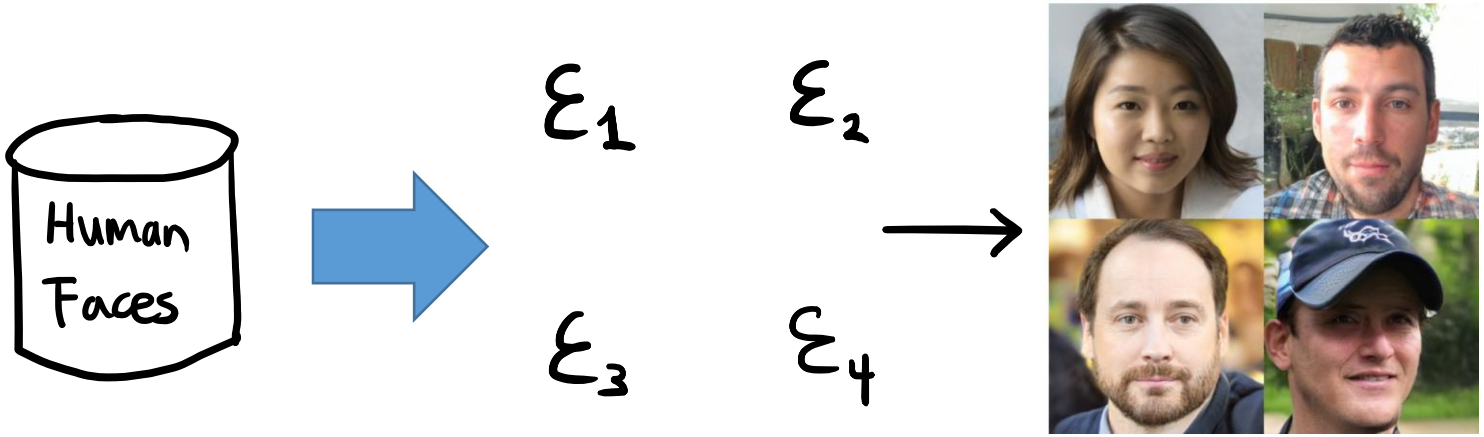
See TELTHONA's other Tweets

Fine Tuning and Transfer Learning

- Remember: **data** defines a NN's behaviour



- Suppose we have a human face generator

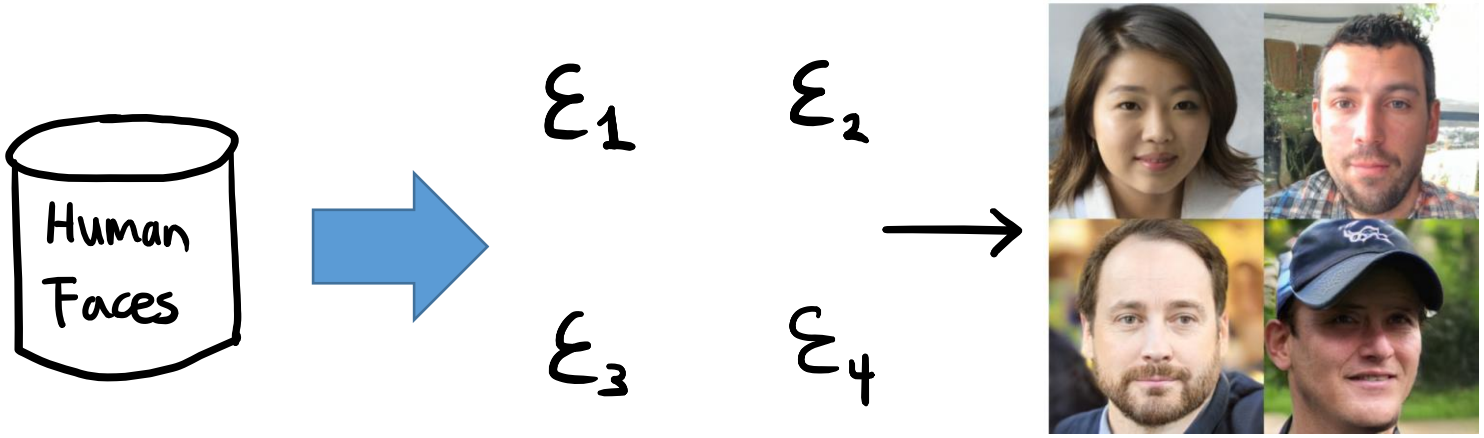


What if I told you...

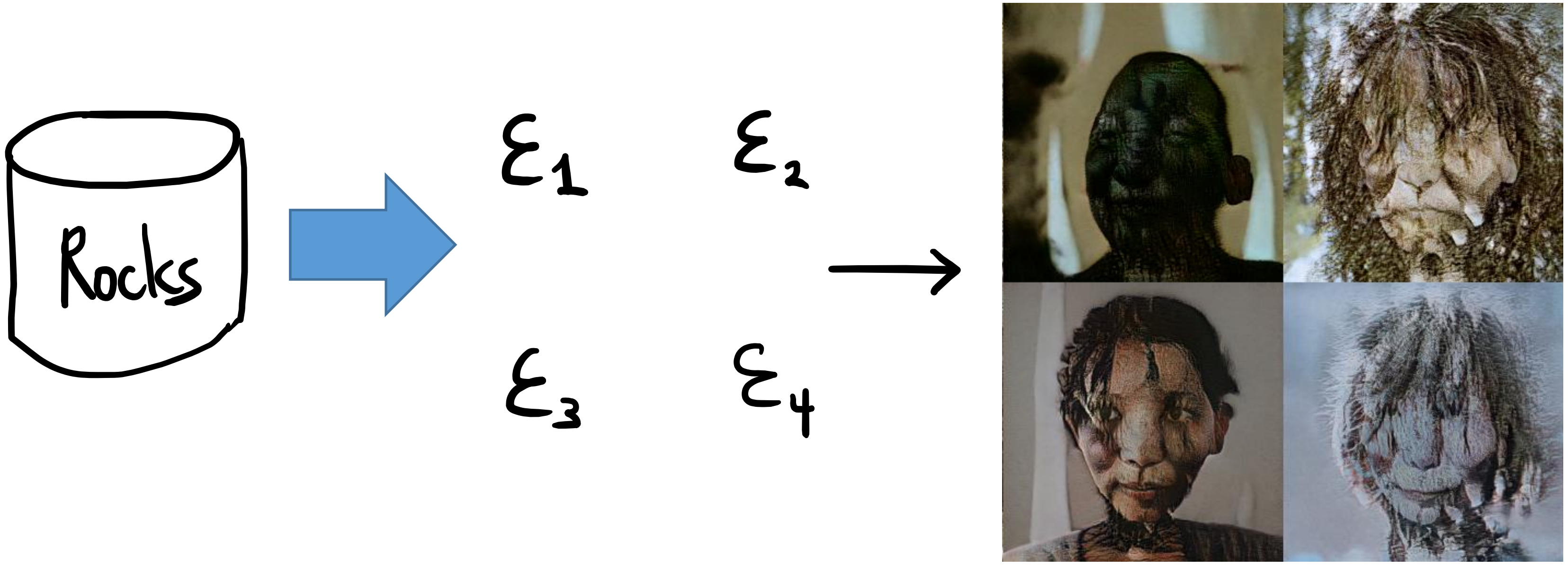
You can change the data

Fine Tuning and Transfer Learning

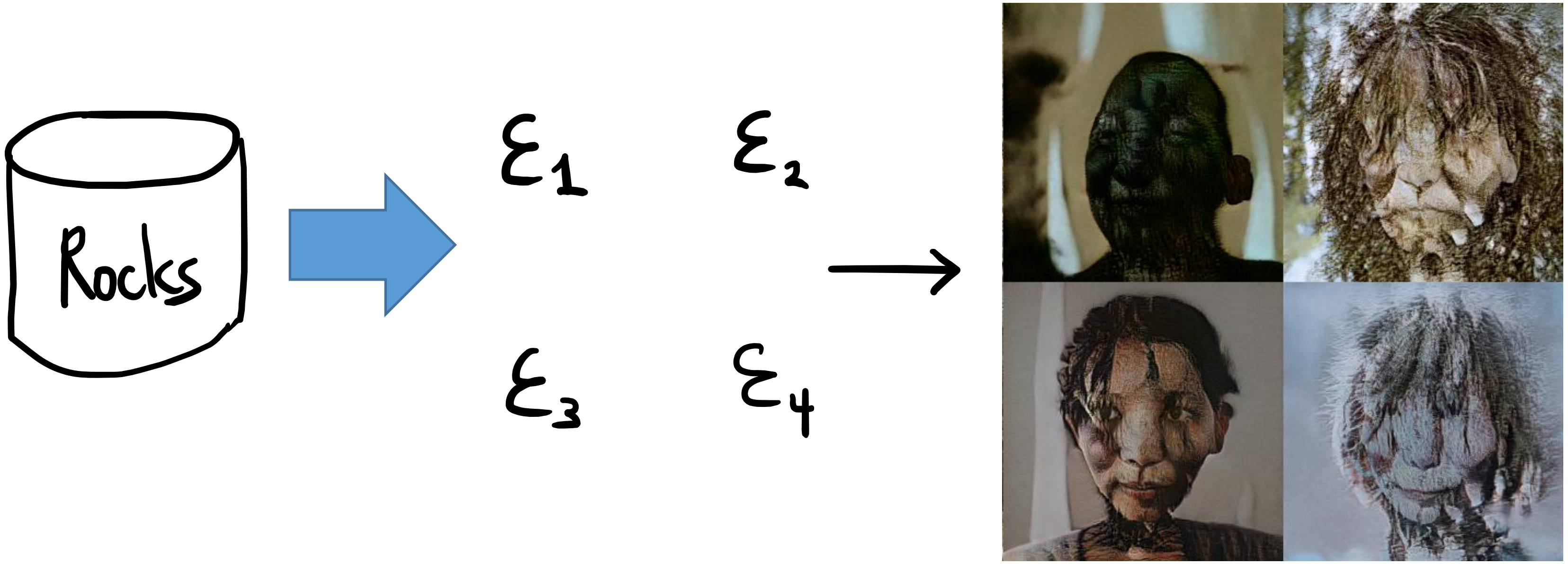
First, train an NN:



Then, with the same NN:



How Does This Work?

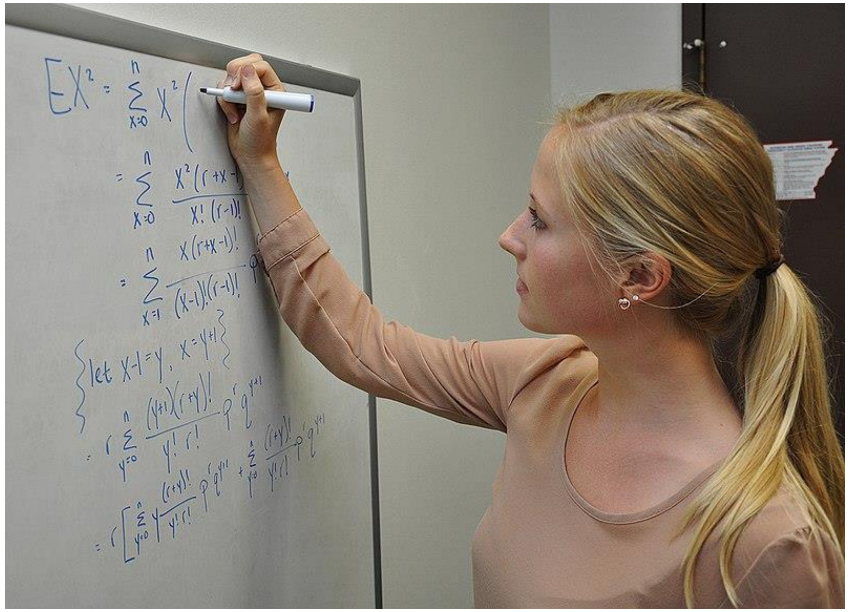


- My neural net was a human face “expert”
- It learned *a little bit* about rocks
- It generates rocks but relies on *existing knowledge* about human faces

Cold-Start and Warm-Start

Cold-Start: learn behaviours from scratch

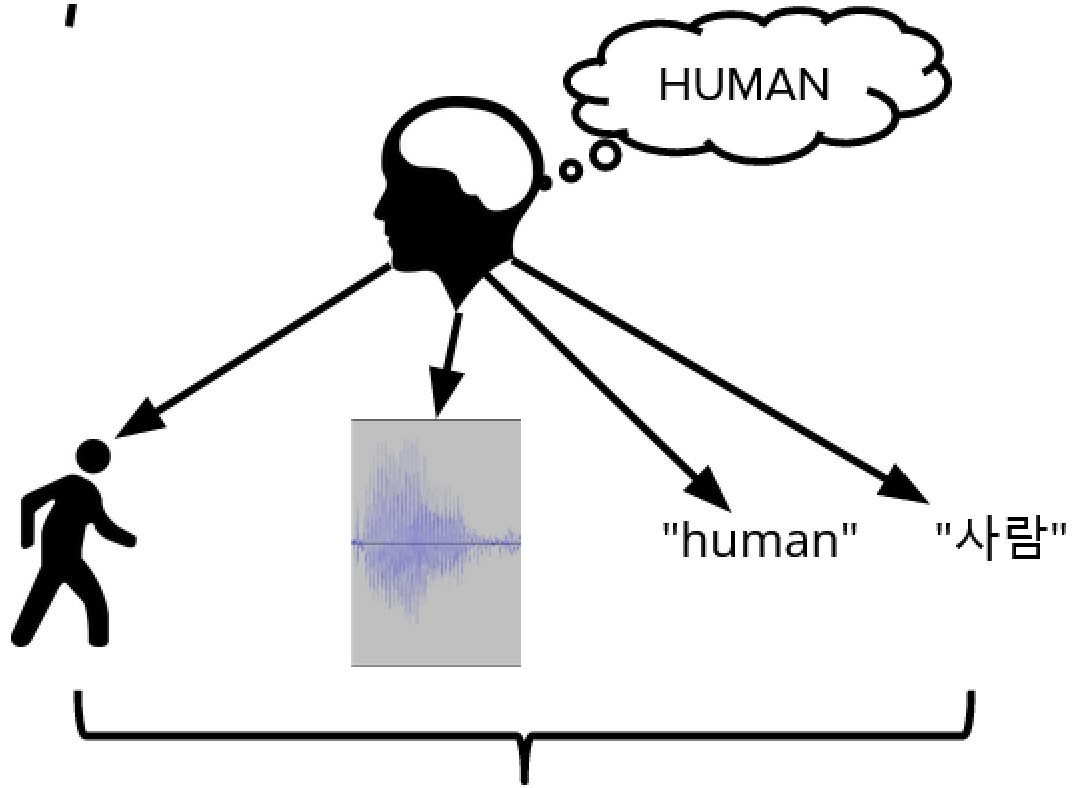
- e.g. initialize NN parameters with noise, use *lots* of data to train it until behaviour emerges



Warm-Start: use existing knowledge to learn new skills

- e.g. use existing NN parameters, use *some* data to slowly adapt the parameters

Learning is Correspondence



These observations are in *correspondence* with one another.
Q: What does it mean to learn the correspondence?

Correspondence is Latent Space Sharing

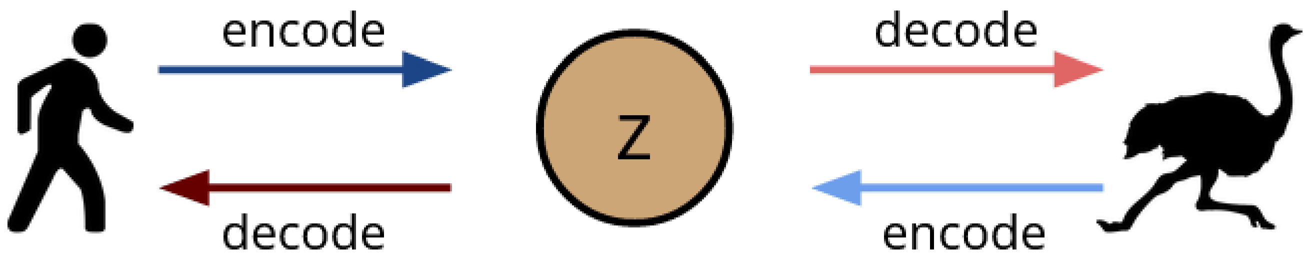


Image-Text Correspondence (Captioning)

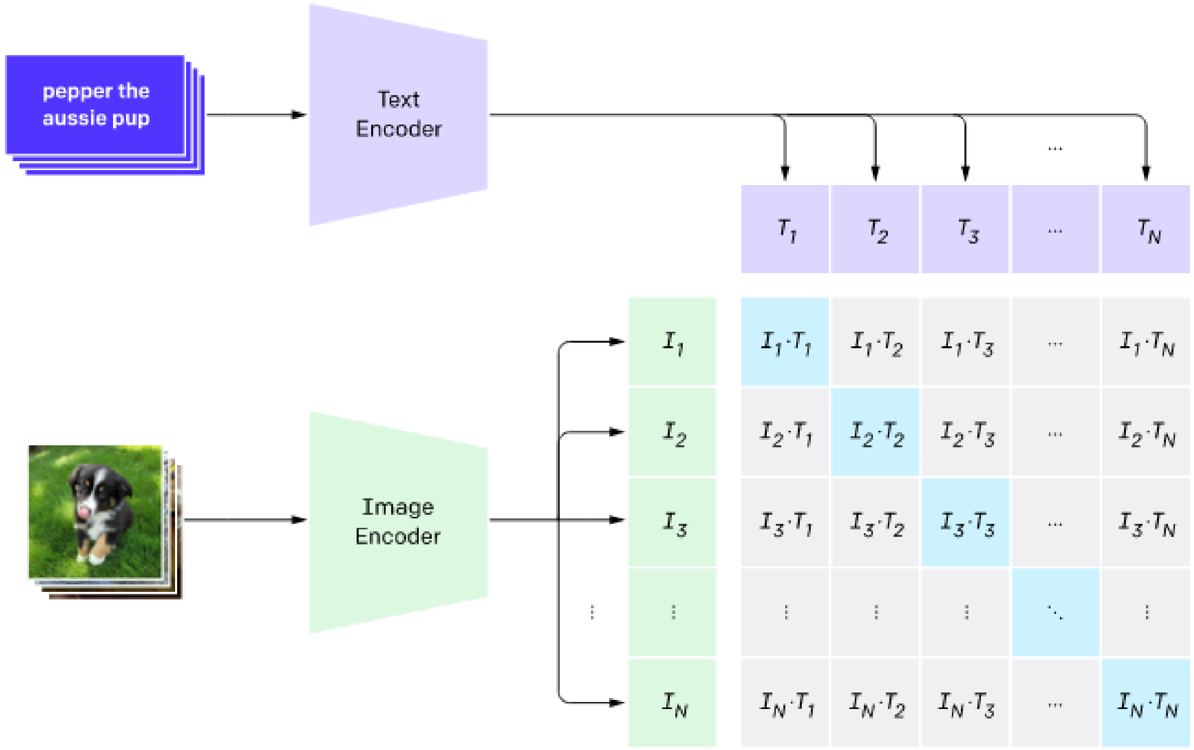
The man at bat readies to swing at the pitch while the umpire looks on.

A large bus sitting next to a very tall building.

A horse carrying a large load of hay and two people sitting on it.

Bunk bed with a narrow shelf sitting underneath it.

OpenAI CLIP (2021)



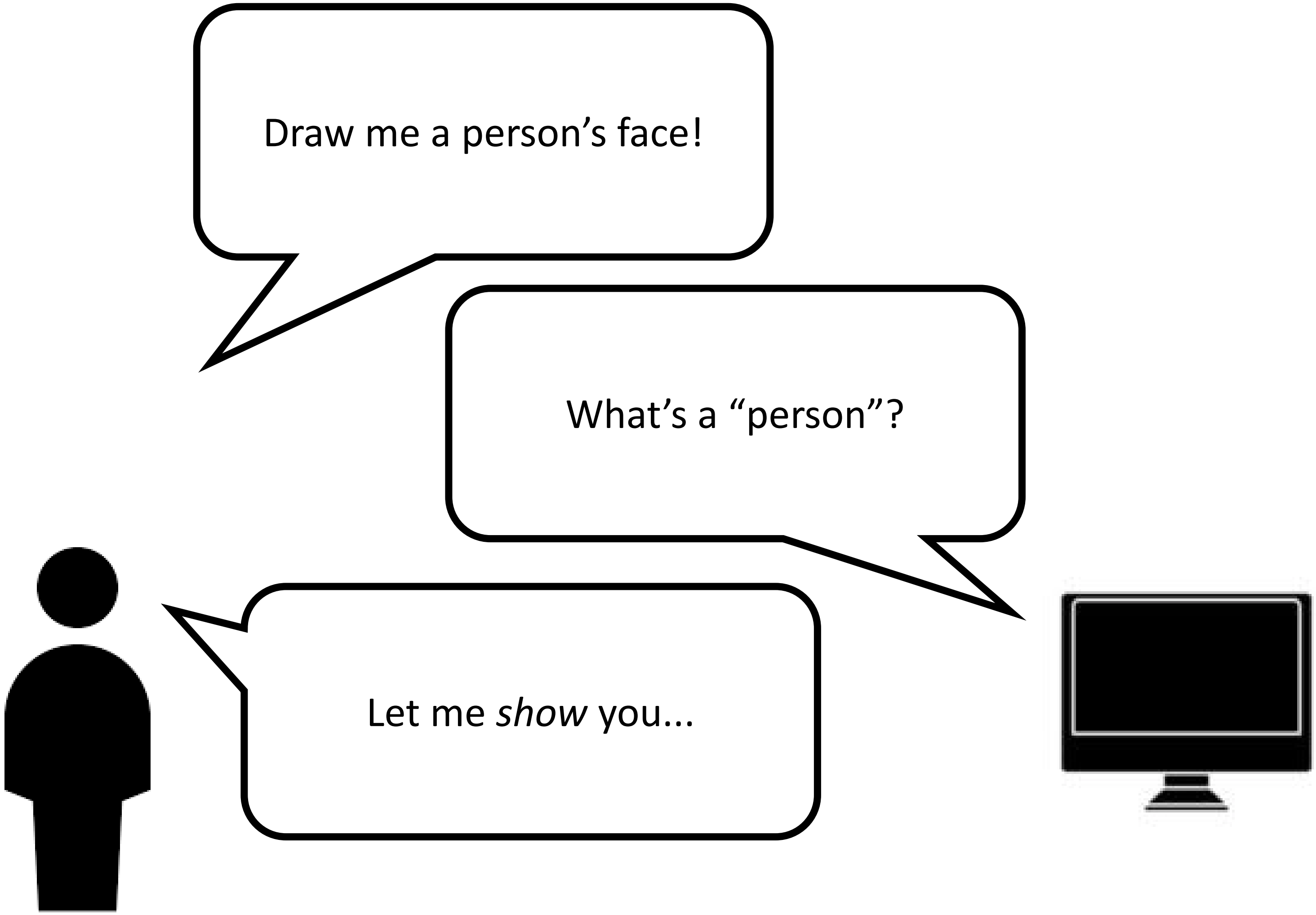
- "Contrastive Language-Image Pretraining" (CLIP): major breakthrough in image-text correspondence using contrastive learning and massive data

Recall: Generative Models

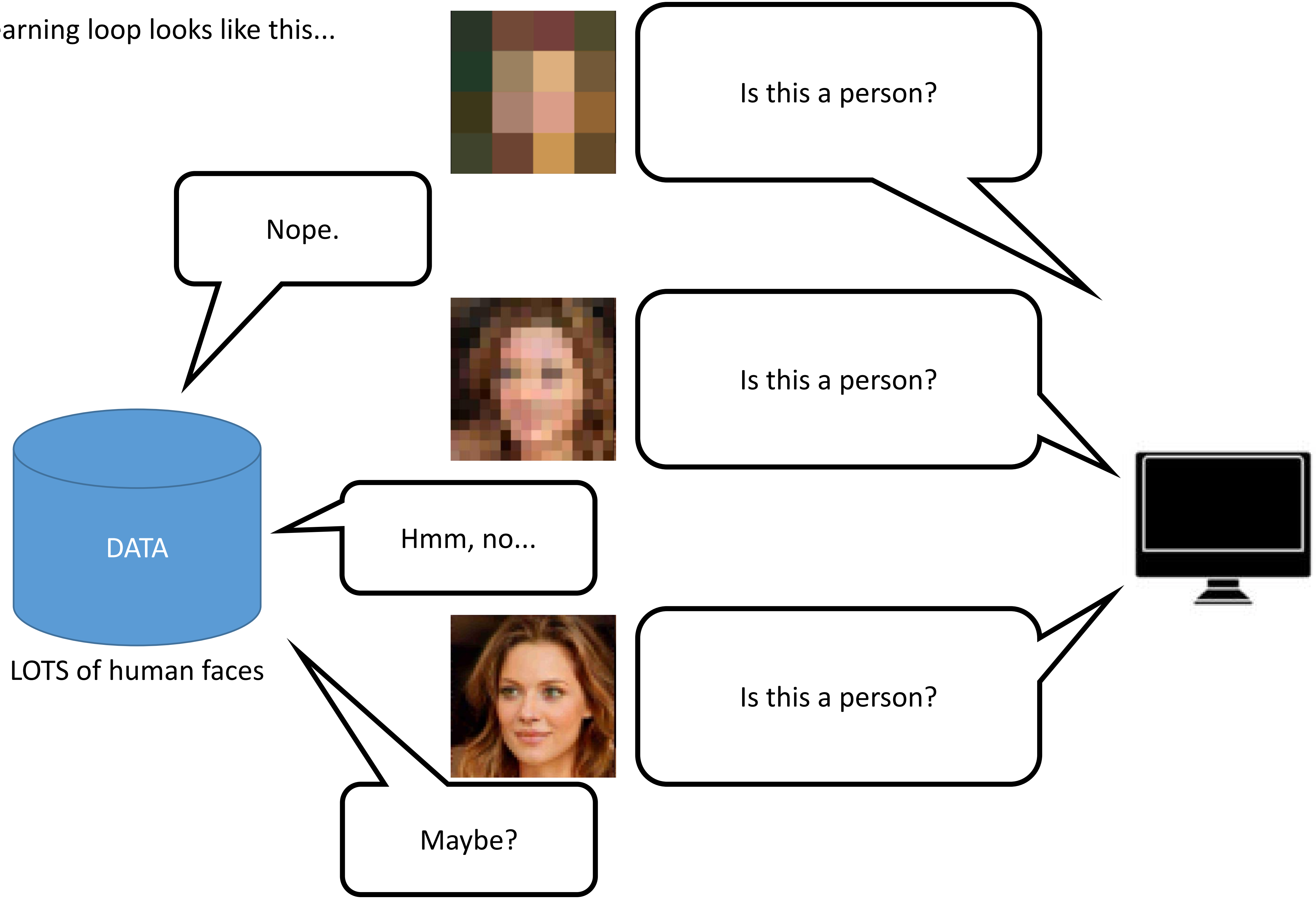
- "Generative models": neural networks that map "noises" to images/audios/samples

$$\underset{\text{noise}}{\epsilon} \longrightarrow \underset{\text{sample}}{\hat{x}}$$

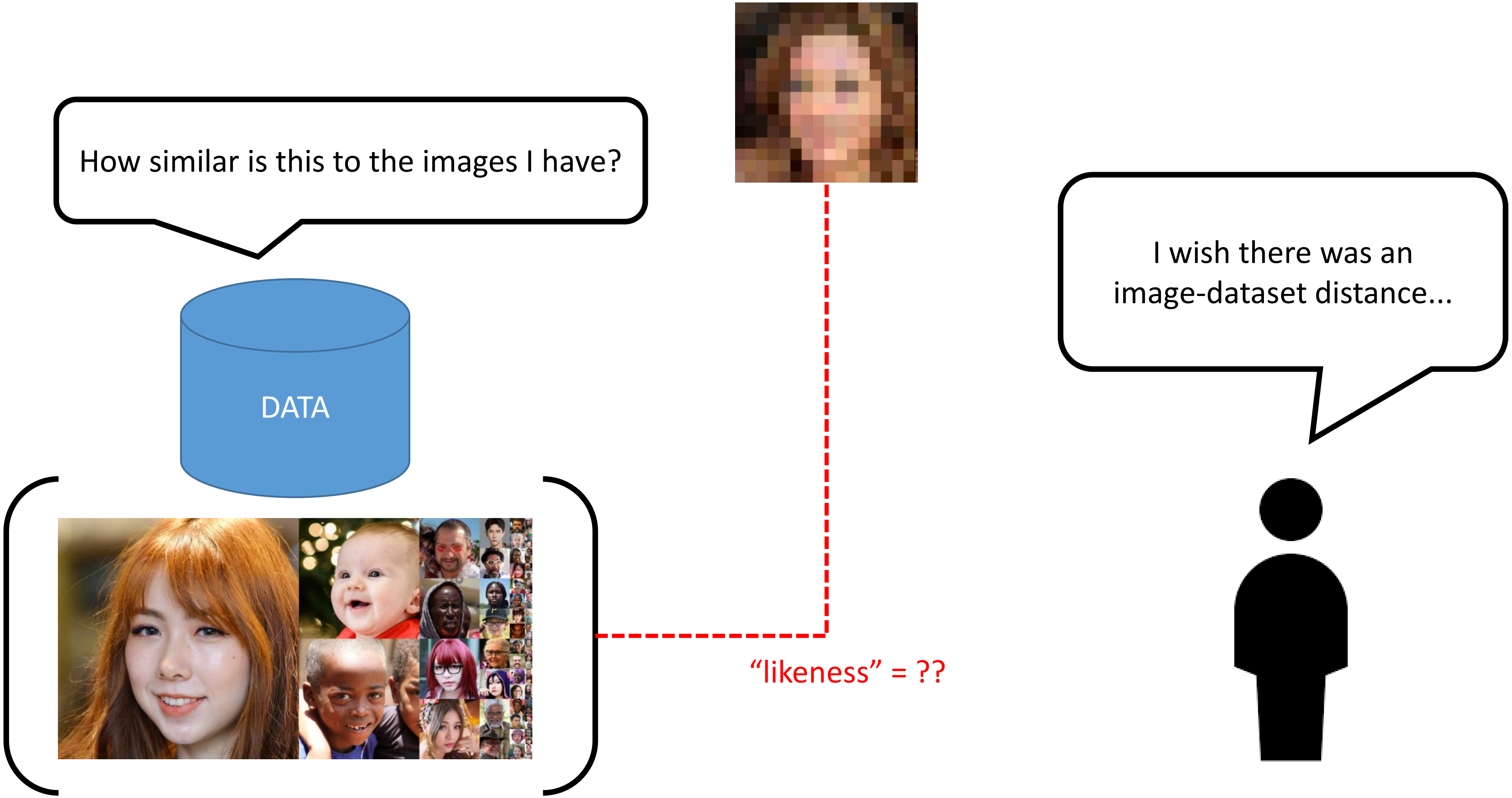
How Do We Use Data?



Every learning loop looks like this...



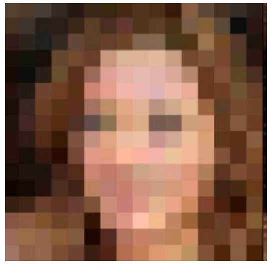
Quantifying “Likeness”



Generative Adversarial Networks (GANs)

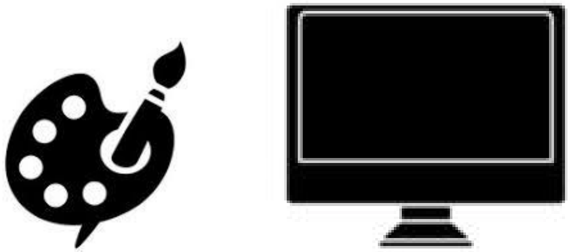
I will predict whether an image
is from the dataset!

“fake”
(low likeness)



My “confidence” can be used as the
generated image’s likeness .

Discriminator



Generator

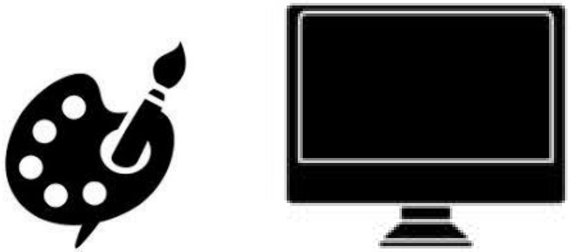
I will make images that can
fool the discriminator!

Generative Adversarial Networks (GANs)

“real”
(WRONG)

Generator is getting good at faking.
I’m going to catch up!

Discriminator

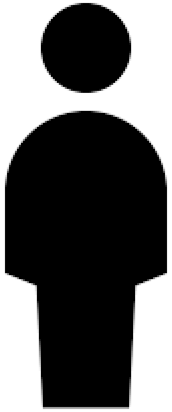


Discriminator is getting good at
catching fakes. I’m going to catch up!

Generator

GANs at Inference Time

Draw me a person's face!



Generator

Here you go! Even the smartest discriminator will have a hard time knowing it's fake.

Diffusion: Recent Alternative to GANs

- Slower at inference, but “more accurate” than GANs

Denoising Diffusion Probabilistic Models

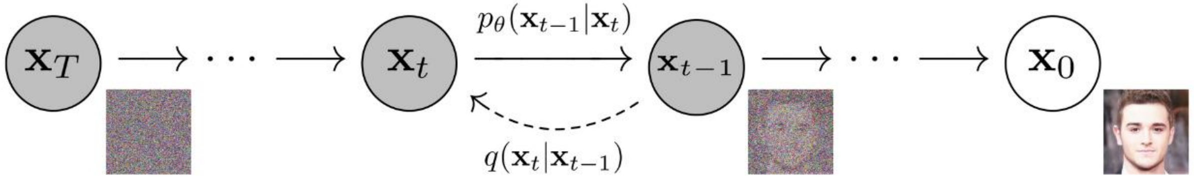
Jonathan Ho
UC Berkeley

Ajay Jain
UC Berkeley

Pieter Abbeel
UC Berkeley

Paper (high-res, 98 MB) | Paper (arXiv, 10 MB) | GitHub

Images generated unconditionally by our probabilistic model. These are not real people, places, animals or objects.



Diffusion probabilistic models are parameterized Markov chains trained to gradually denoise data. We estimate parameters of the generative process p.

Correspondence + Generative Model = Power

TEXT PROMPT an armchair in the shape of an avocado. . . .

AI-GENERATED IMAGES

Edit prompt or view more images↓

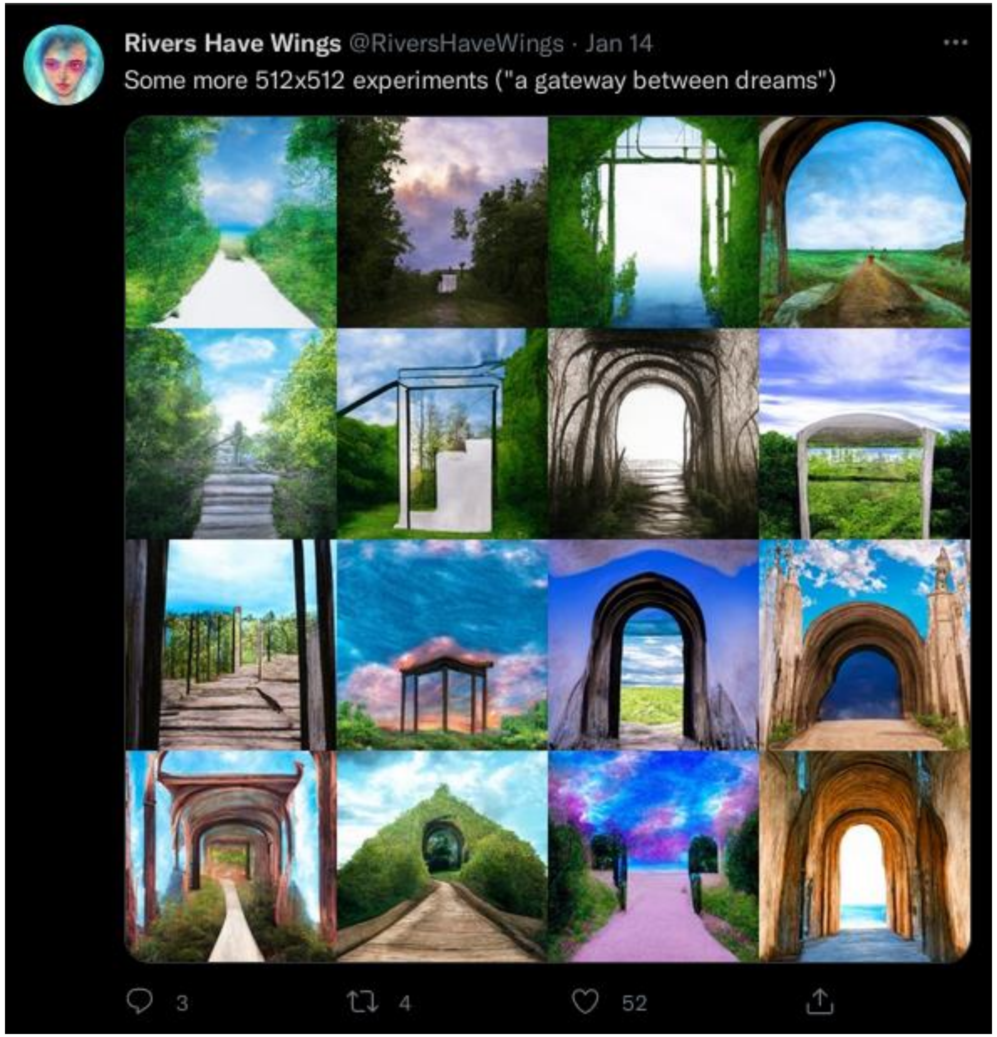
TEXT PROMPT a store front that has the word 'openai' written on it. . . .

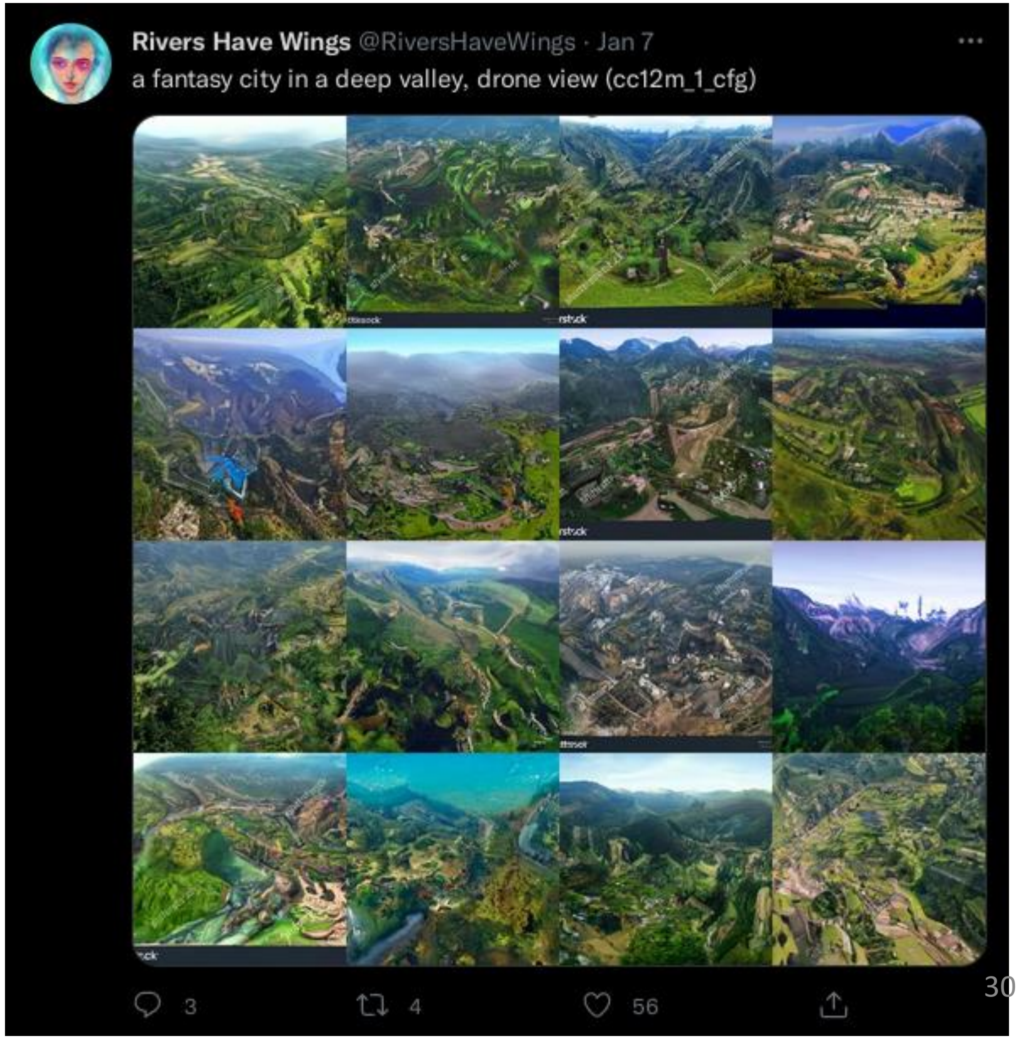
AI-GENERATED IMAGES

Edit prompt or view more images↓

OpenAI DALL-E (2021)

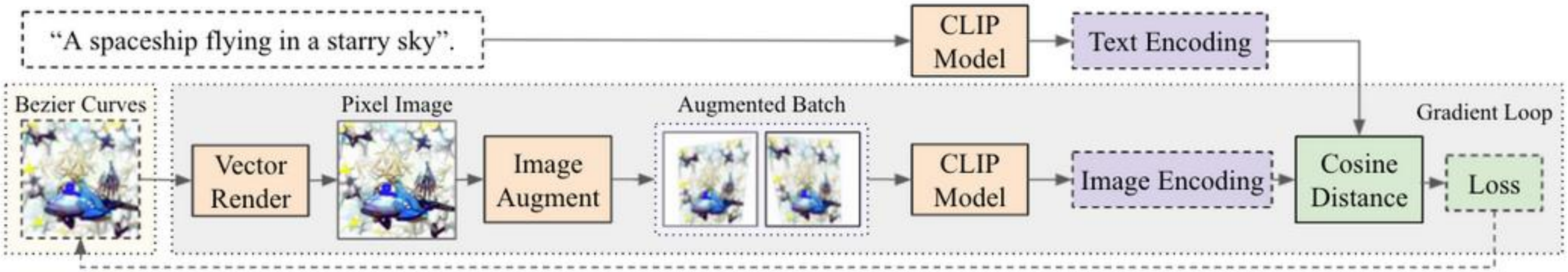
CLIP-Guided Diffusion



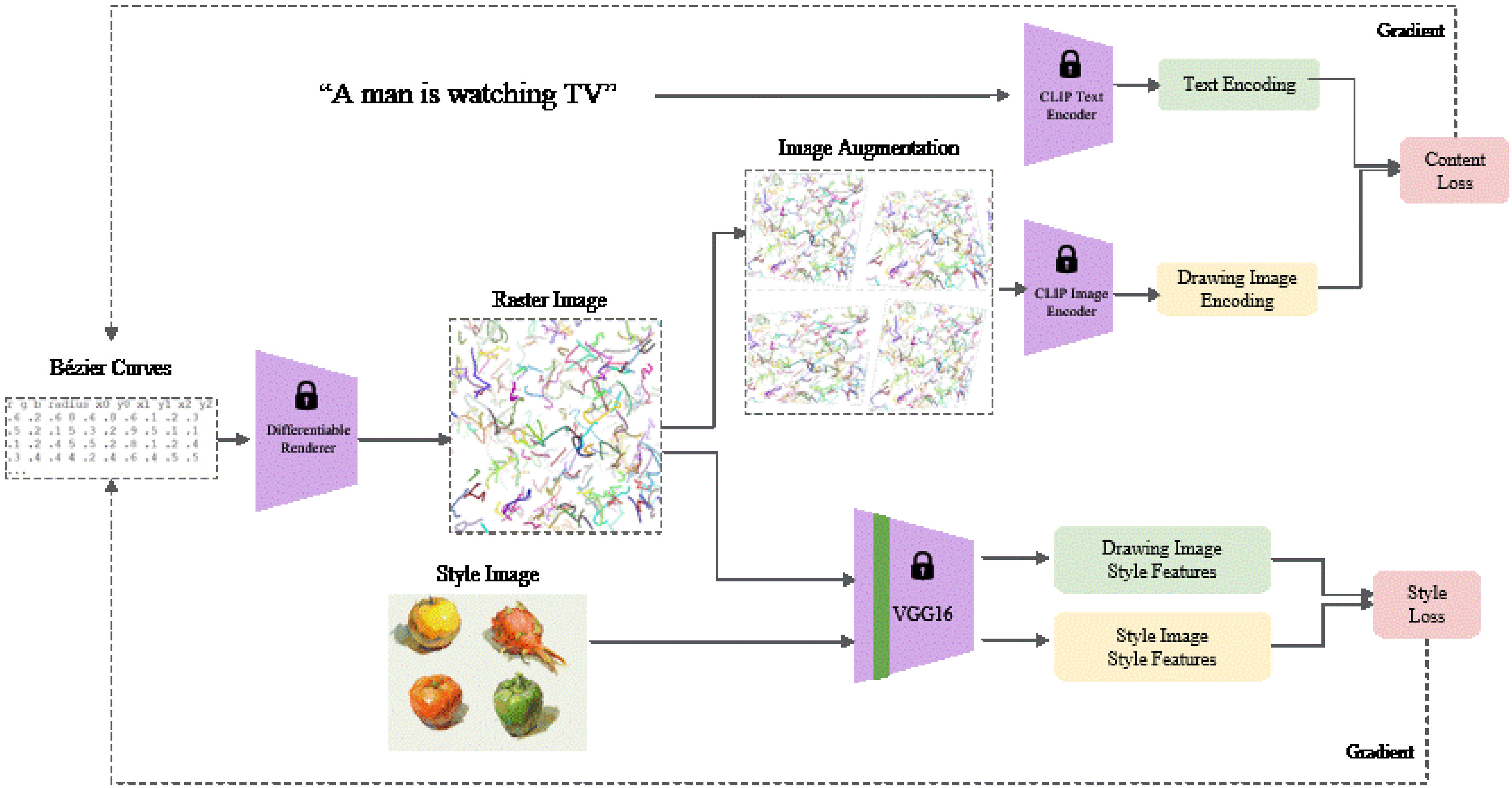


CLIPDraw (2021)

"A drawing of a cat". | "Horse eating a cupcake". | "A 3D rendering of a temple". | "Family vacation to Walt Disney World". | "Self".



StyleCLIPDraw (2021)



OpenAI GLIDE (2021)

“a hedgehog using a calculator”

“a corgi wearing a red bowtie and a purple party hat”

“robots meditating in a vipassana retreat”

“a fall landscape with a small cottage next to a lake”

“a surrealist dream-like oil painting by salvador dalí of a cat playing checkers”

“a professional photo of a sunset behind the grand canyon”

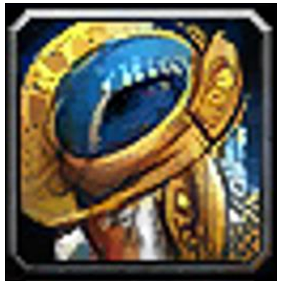
“a high-quality oil painting of a psychedelic hamster dragon”

“an illustration of albert einstein wearing a superhero costume”

Generating Game Assets

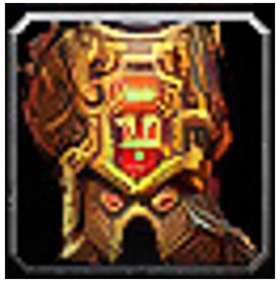
- These WoW icons don't exist

new race/pet type?

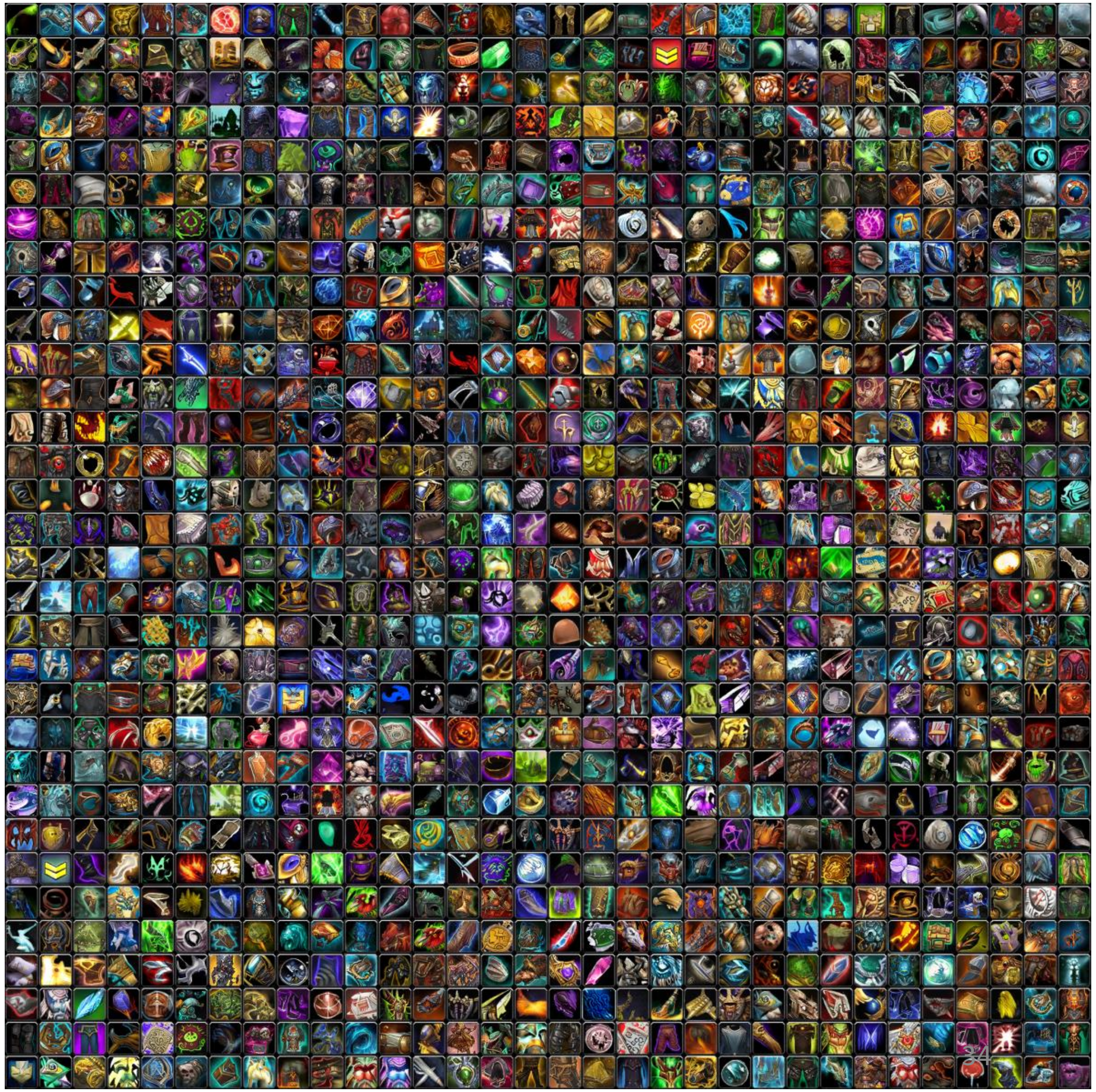


legendary headpiece

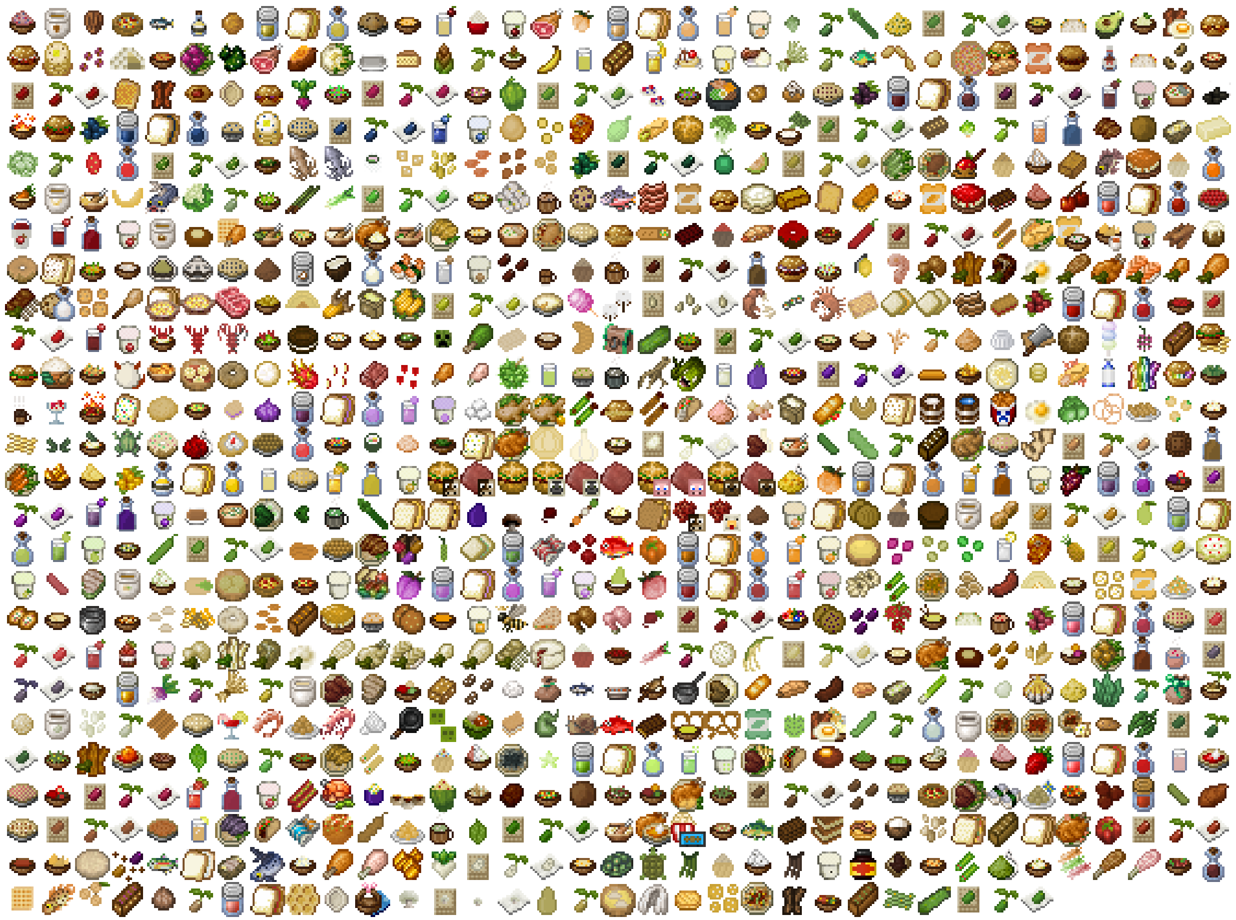
new spell



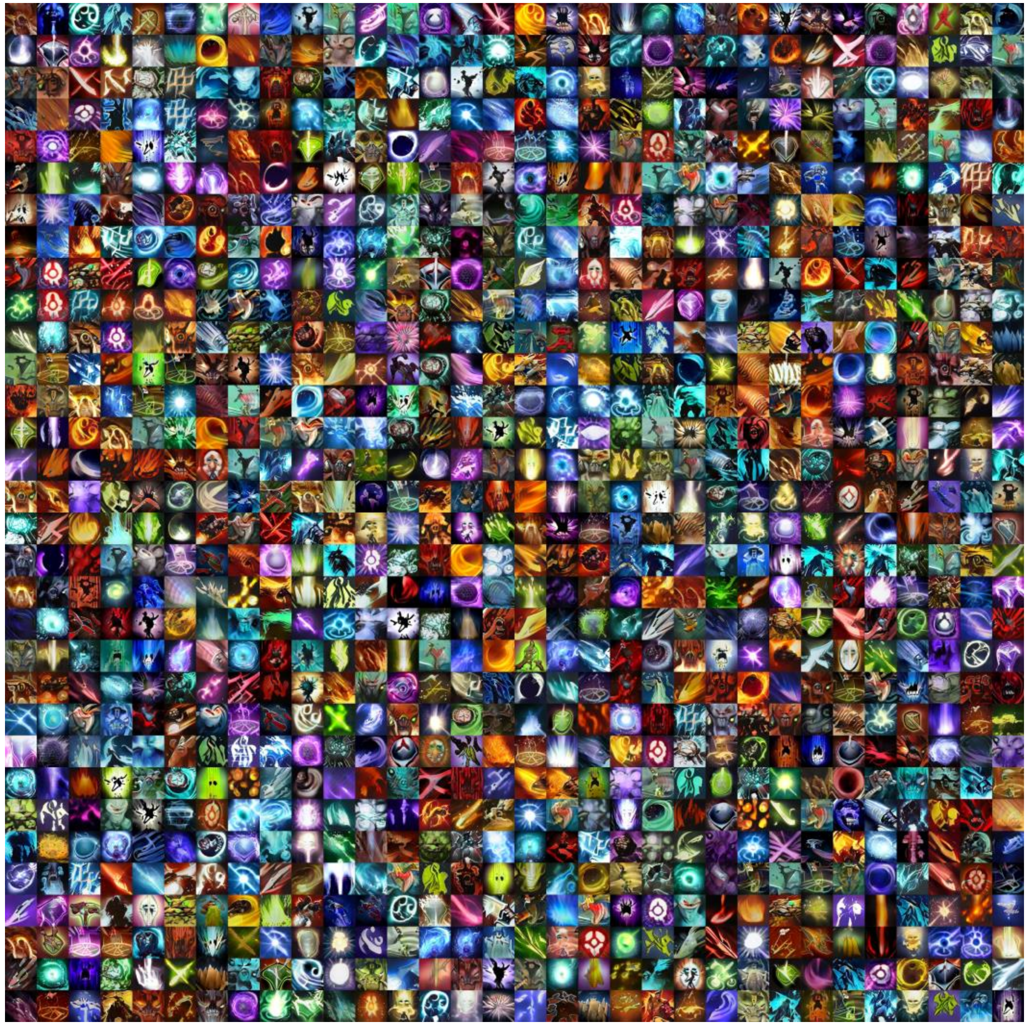
item for troll-themed quest



Generating Game Assets



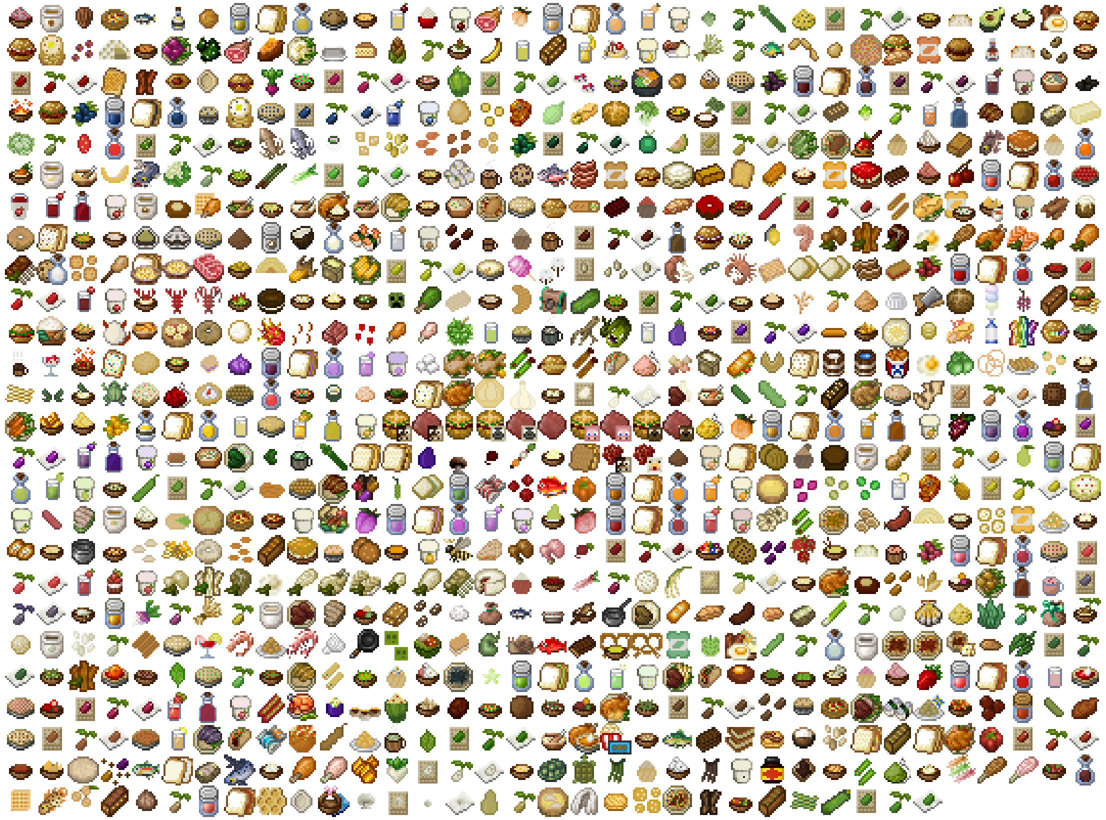
JRPG Food



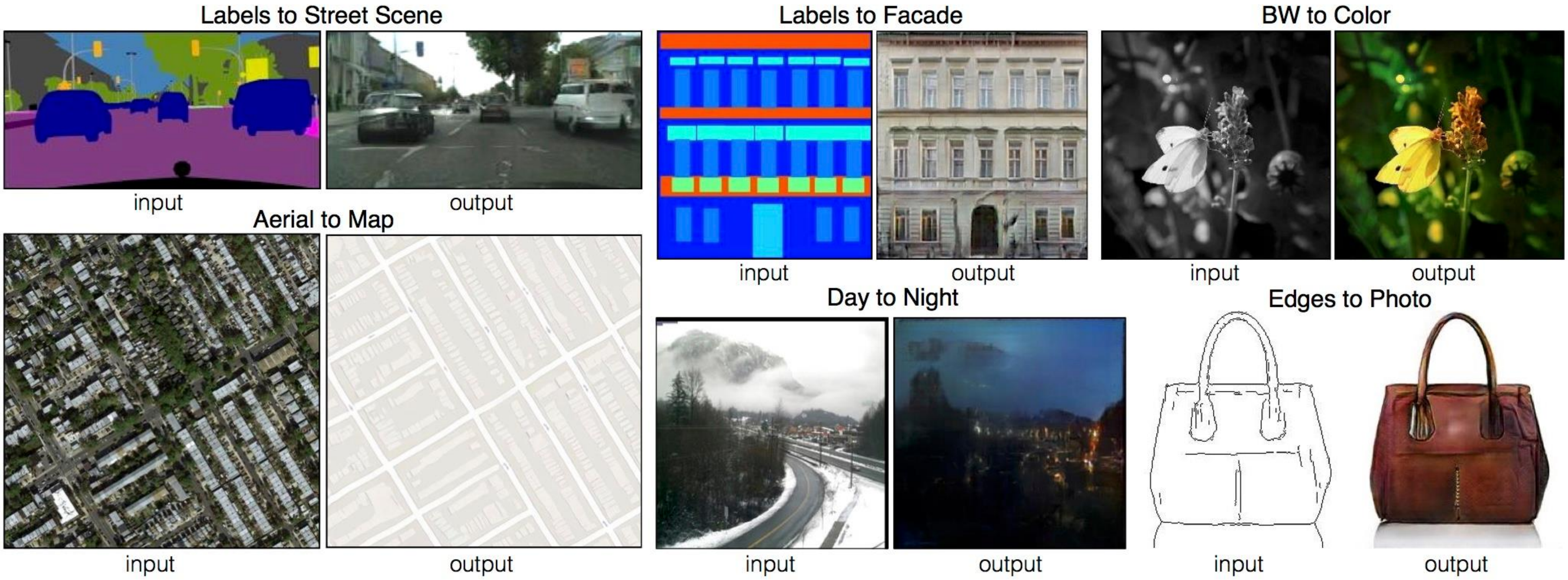
Dota2 Icons

Hints for Game Asset Generation

- https://www.pinterest.com/pin/335799715939349668/

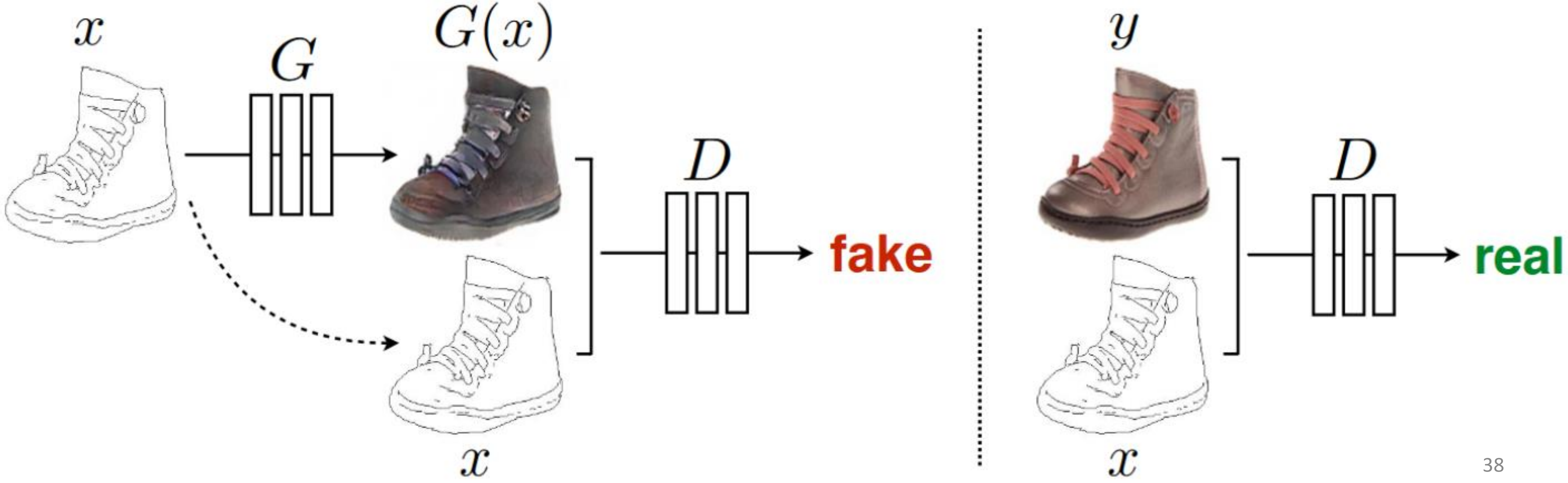


pix2pix (Isola et al. 2017)



pix2pix (Isola et al. 2017)

- pix2pix is an example of conditional GAN
- Generator input: edge map, noise
- Discriminator input: edge map, real or generated images



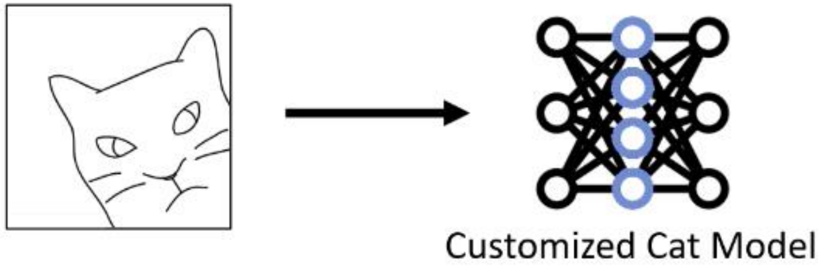
Sketch Your Own GAN (Wang et al. 2021)

Sketch Your Own GAN

Sheng-Yu Wang[1] David Bau[2] Jun-Yan Zhu[1]

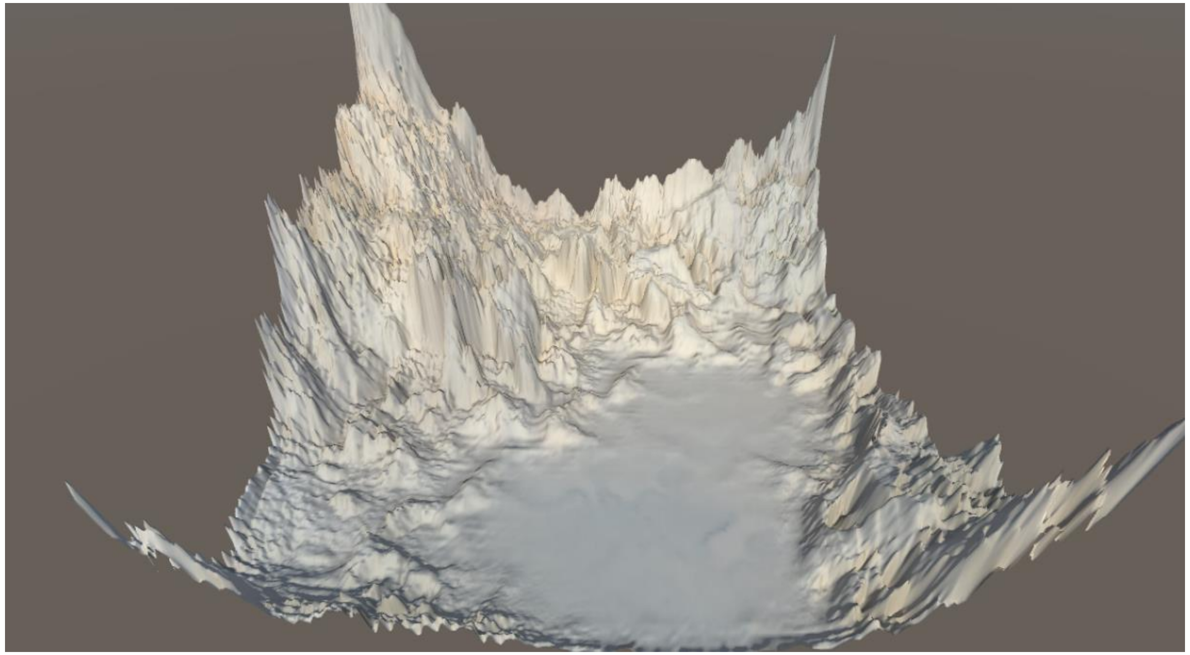
[1]CMU [2]MIT CSAIL

Code [GitHub] ICCV 2021 [Paper]



Generating terrain (Beckham 2017)

- Generate heightmaps with GAN
- Use pix2pix to generate textures from heightmaps



Mattull 2020

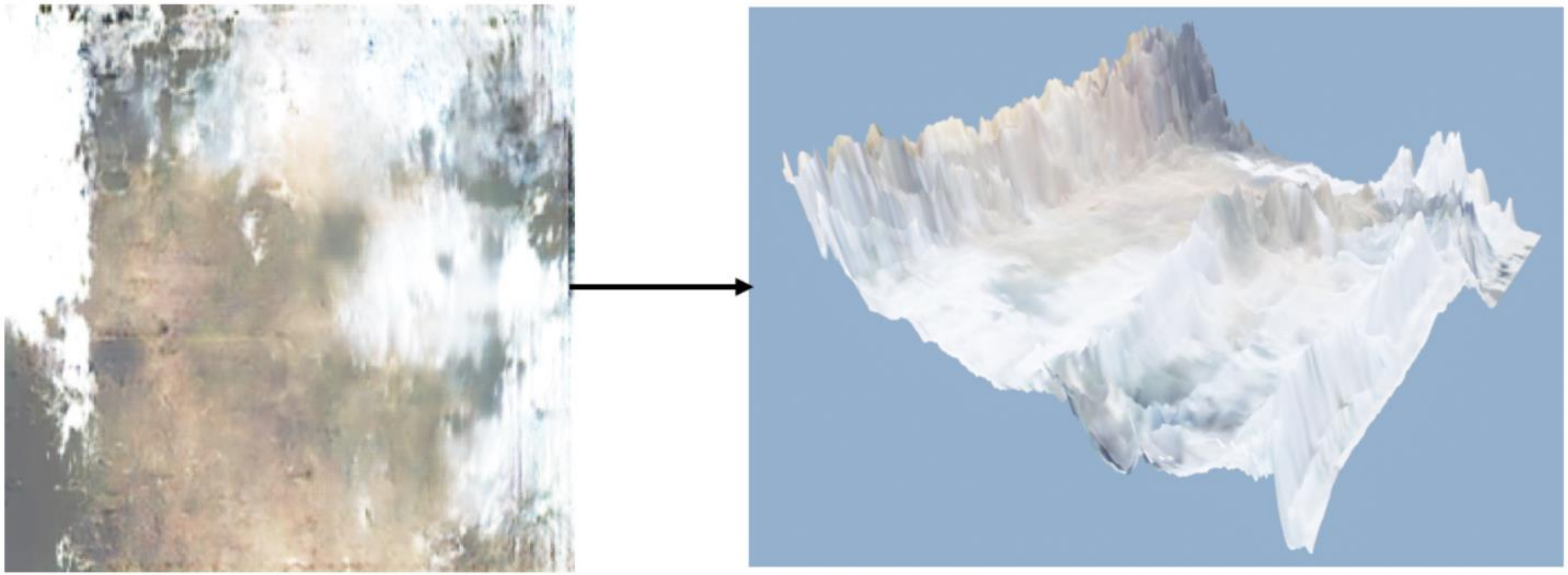
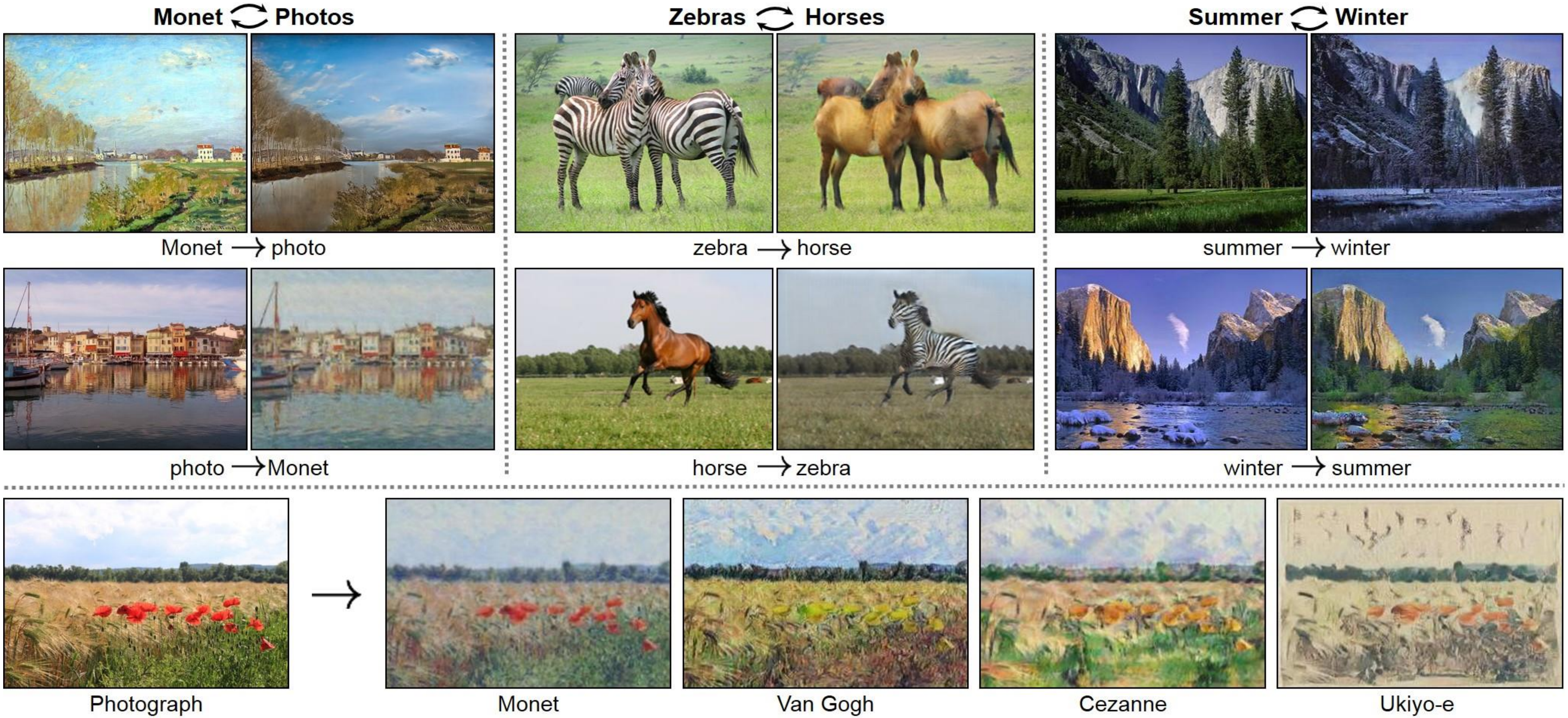
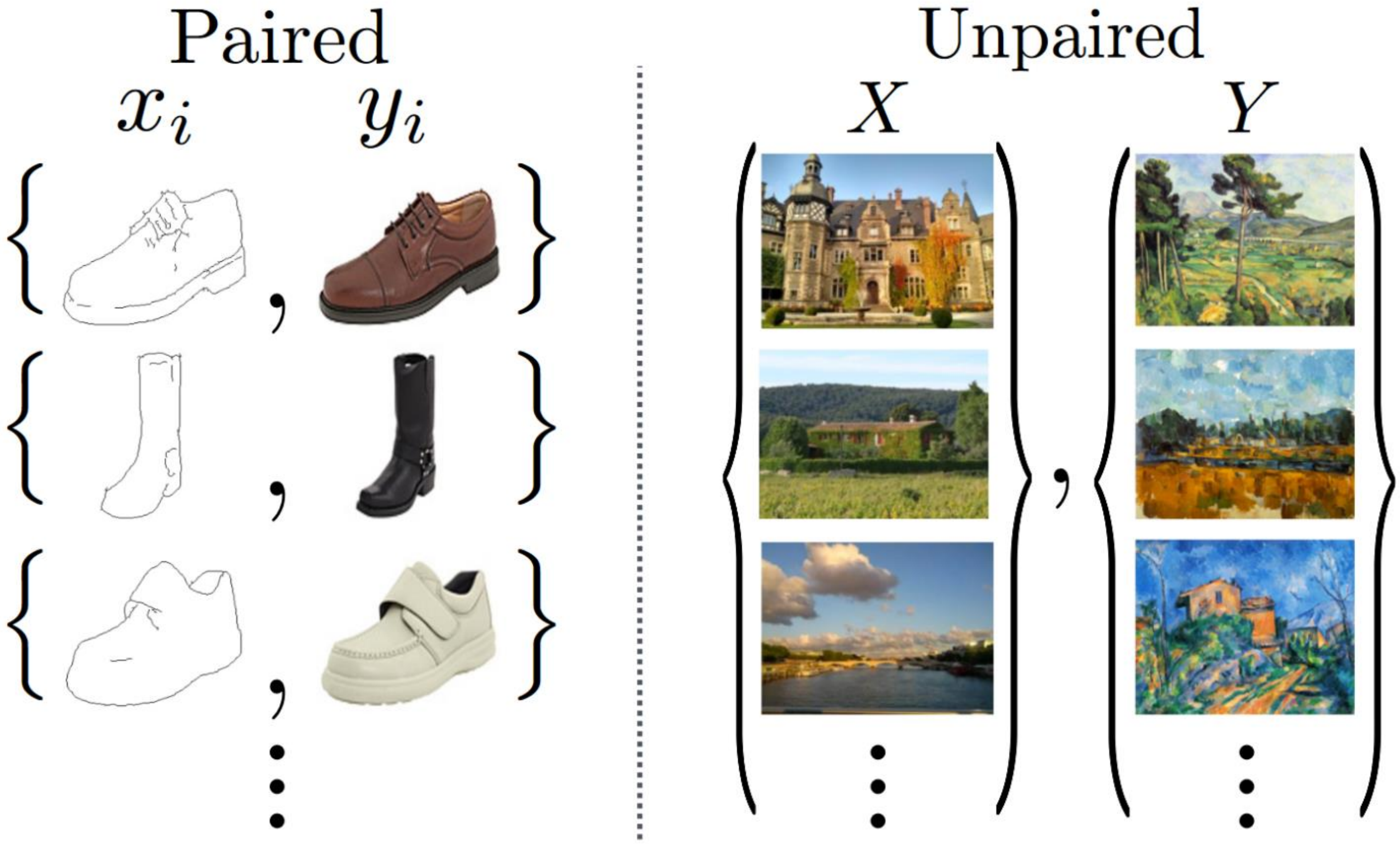


Figure 6.14: Using Amazon's Lumberyard terrain editor to render realistic terrain from a texture map generated by a pix2pix GAN.

CycleGAN (Jun-Yan Zhu et al. 2017)



CycleGAN: no training pairs needed!



CycleGAN: no training pairs needed!

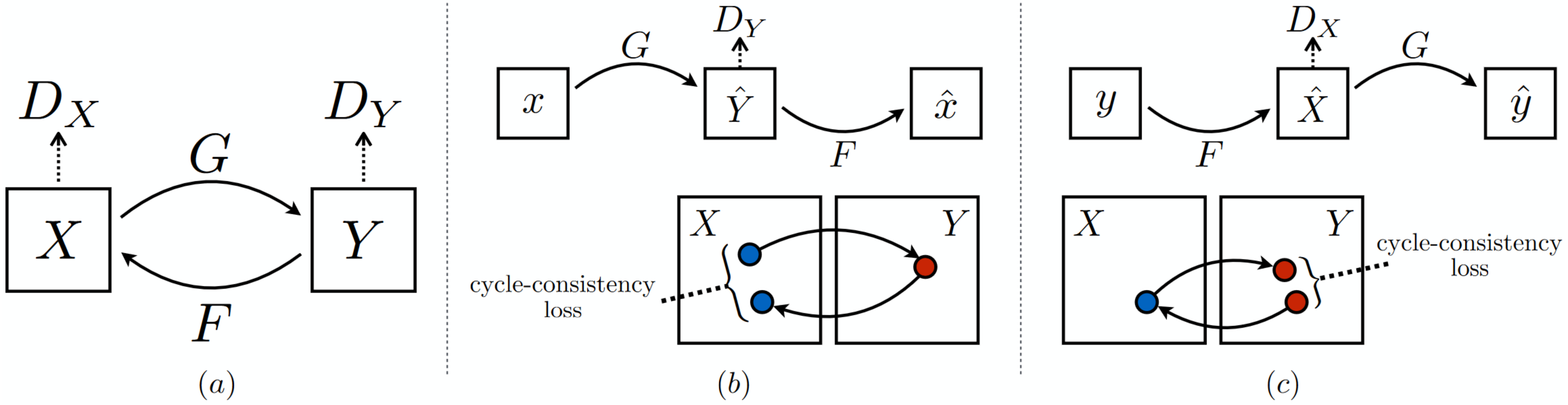
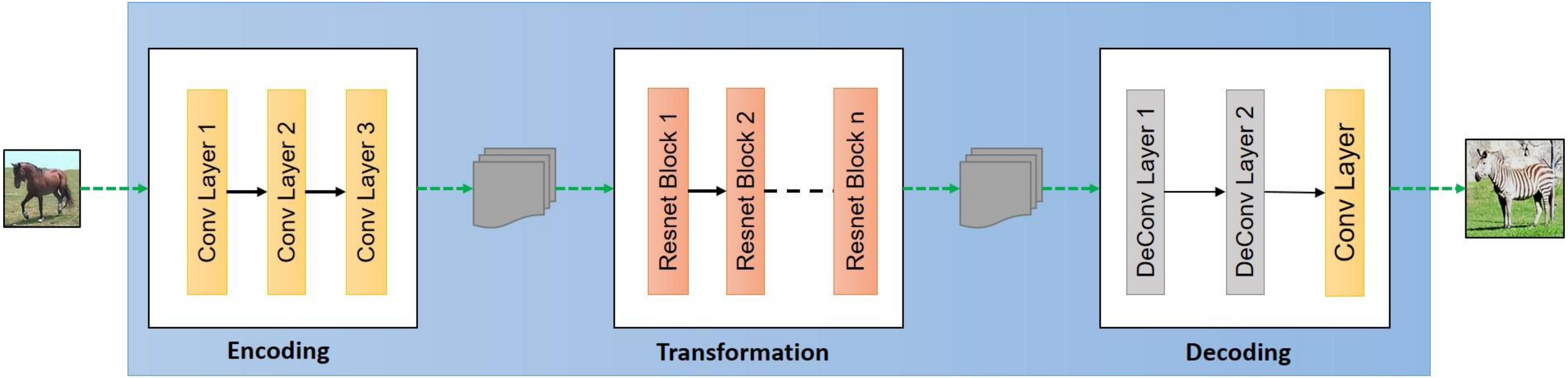


Figure 3: (a) Our model contains two mapping functions $G : X \to Y$ and $F : Y \to X$, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y, and vice versa for D_X and F. To further regularize the mappings, we introduce two *cycle consistency losses* that capture the intuition that if we translate from one domain to the other and back again we should arrive at where we started: (b) forward cycle-consistency loss: $x \to G(x) \to F(G(x)) \approx x$, and (c) backward cycle-consistency loss: $y \to F(y) \to G(F(y)) \approx y$

CycleGAN: no training pairs needed!



Painting with GANs (2019)

GauGAN: https://github.com/NVlabs/SPADE

GAN Dissection: https://gandissect.csail.mit.edu

StyleGAN 2 ”circuit bending”

Finetuning a pretrained Nvidia StyleGAN 2 network with only 250 images (google image search with ”dragons”)

https://twitter.com/Norod78/status/1218282356391530496?s=20

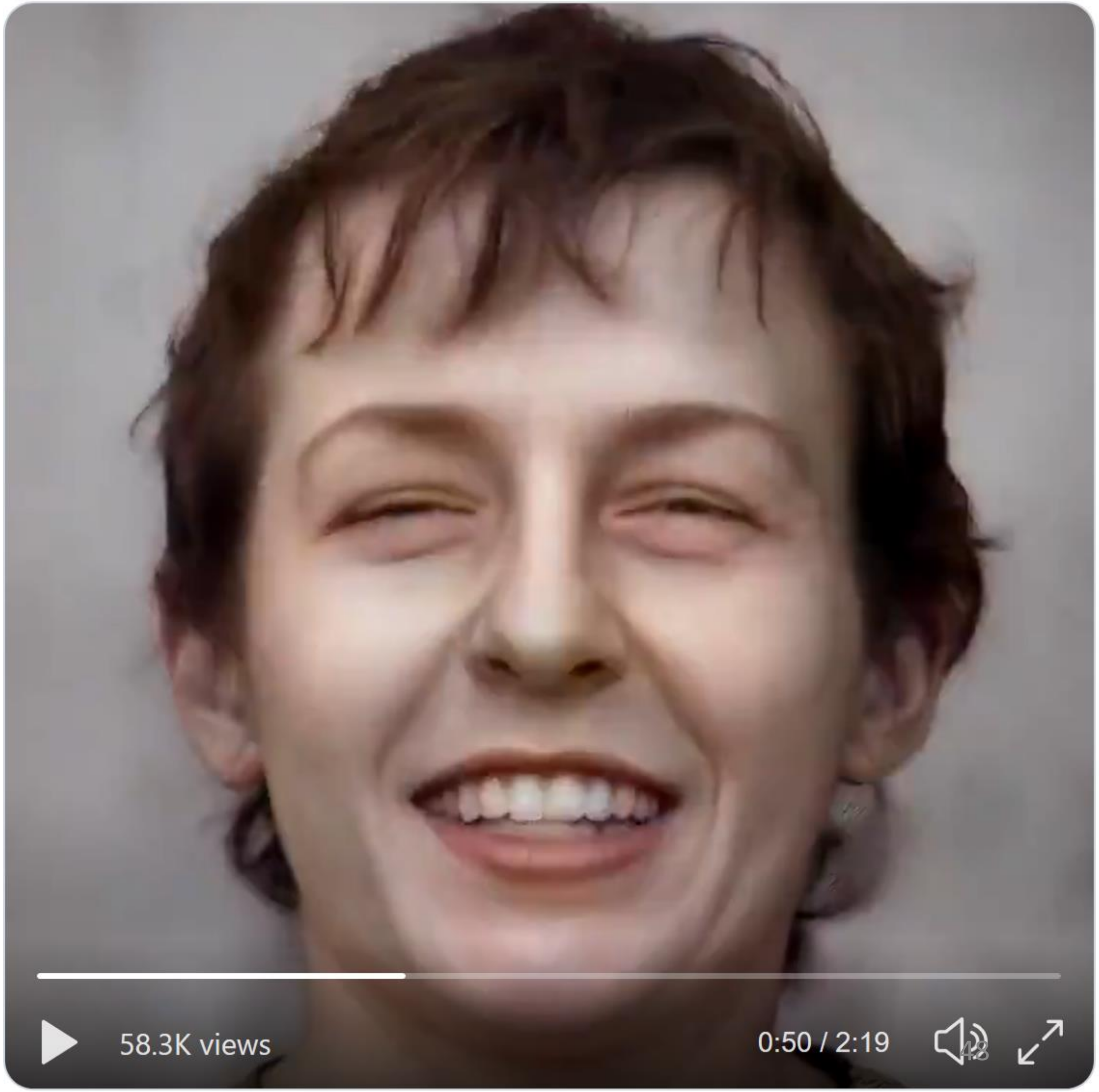
Music visualization using StyleGAN 2

https://twitter.com/quasimondo/status/1244562140217905153?s=20

Mario Klingemann @quasimondo · Mar 30

Current progress on mapping music to facial expression vectors. #StyleGAN2 #realtime

Song: "Triggernometry" by Kraftamt, 2014



StyleGAN font generation

http://www.machinelearningfont.com/#mlfabout

[Submitted on 29 May 2019 (v1), last revised 30 May 2019 (this version, v2)]

GlyphGAN: Style-Consistent Font Generation Based on Generative Adversarial Networks

Hideaki Hayashi, Kohtaro Abe, Seiichi Uchida

In this paper, we propose GlyphGAN: style-consistent font generation based on generative adversarial networks (GANs). GANs are a framework for learning a generative model using a system of two neural networks competing with each other. One network generates synthetic images from random input vectors, and the other discriminates between synthetic and real images. The motivation of this study is to create new fonts using the GAN framework while maintaining style consistency over all characters. In GlyphGAN, the input vector for the generator network consists of two vectors: character class vector and style vector. The former is a one-hot vector and is associated with the character class of each sample image during training. The latter is a uniform random vector without supervised information. In this way, GlyphGAN can generate an infinite variety of fonts with the character and style independently controlled. Experimental results showed that fonts generated by GlyphGAN have style consistency and diversity different from the training images without losing their legibility.

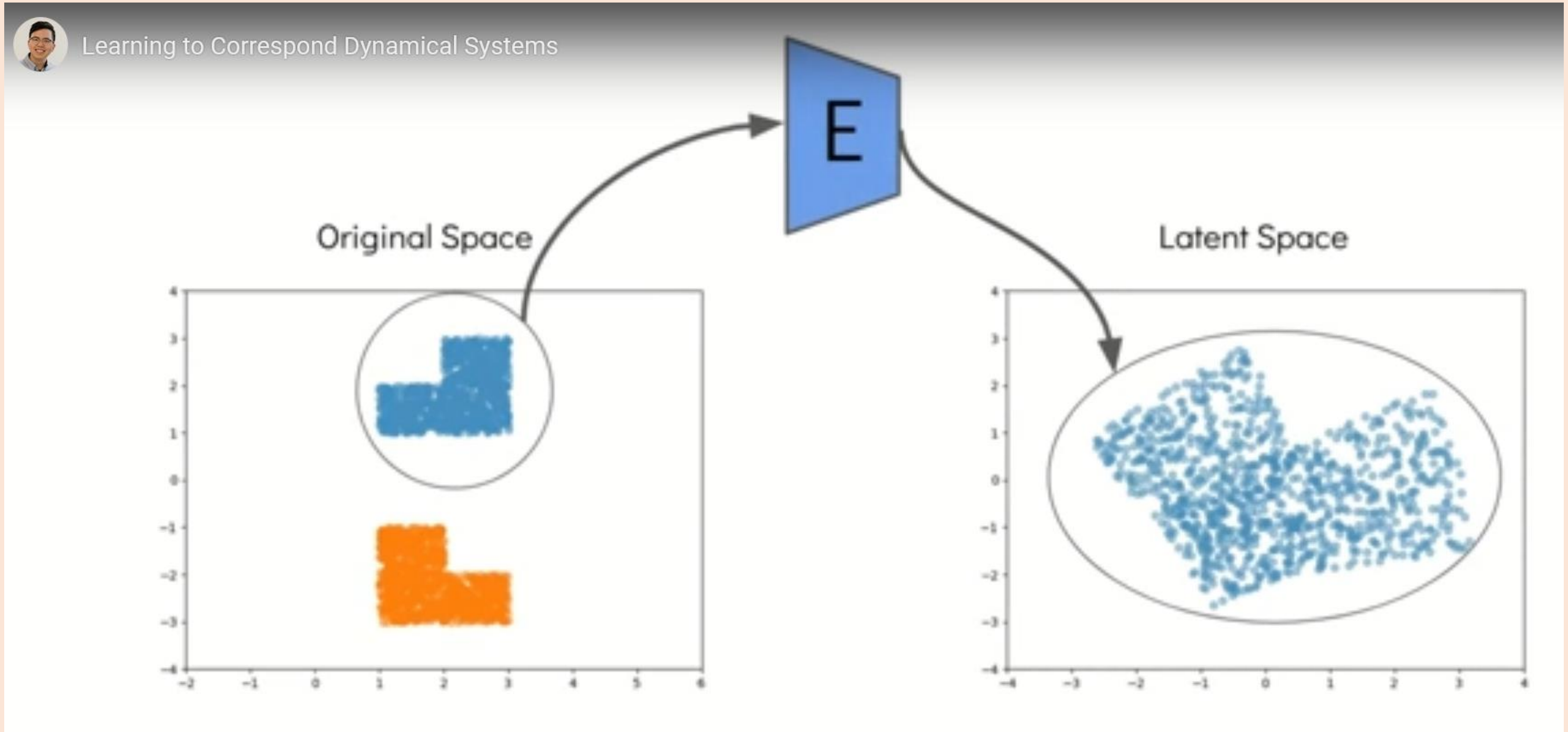
Controlling GANs (Härkönen et al. 2020)

- Applications using GANs
  - Font generation
  - Anime character generation
  - Interactive Image generation
  - Text2Image (text to image)
  - 3D Object generation
  - Image Editing
  - Face Aging
  - Human Pose Estimation
  - Domain-transfer (e.g. style-transfer, pix2pix, sketch2image)
  - Image Inpainting (hole filling)

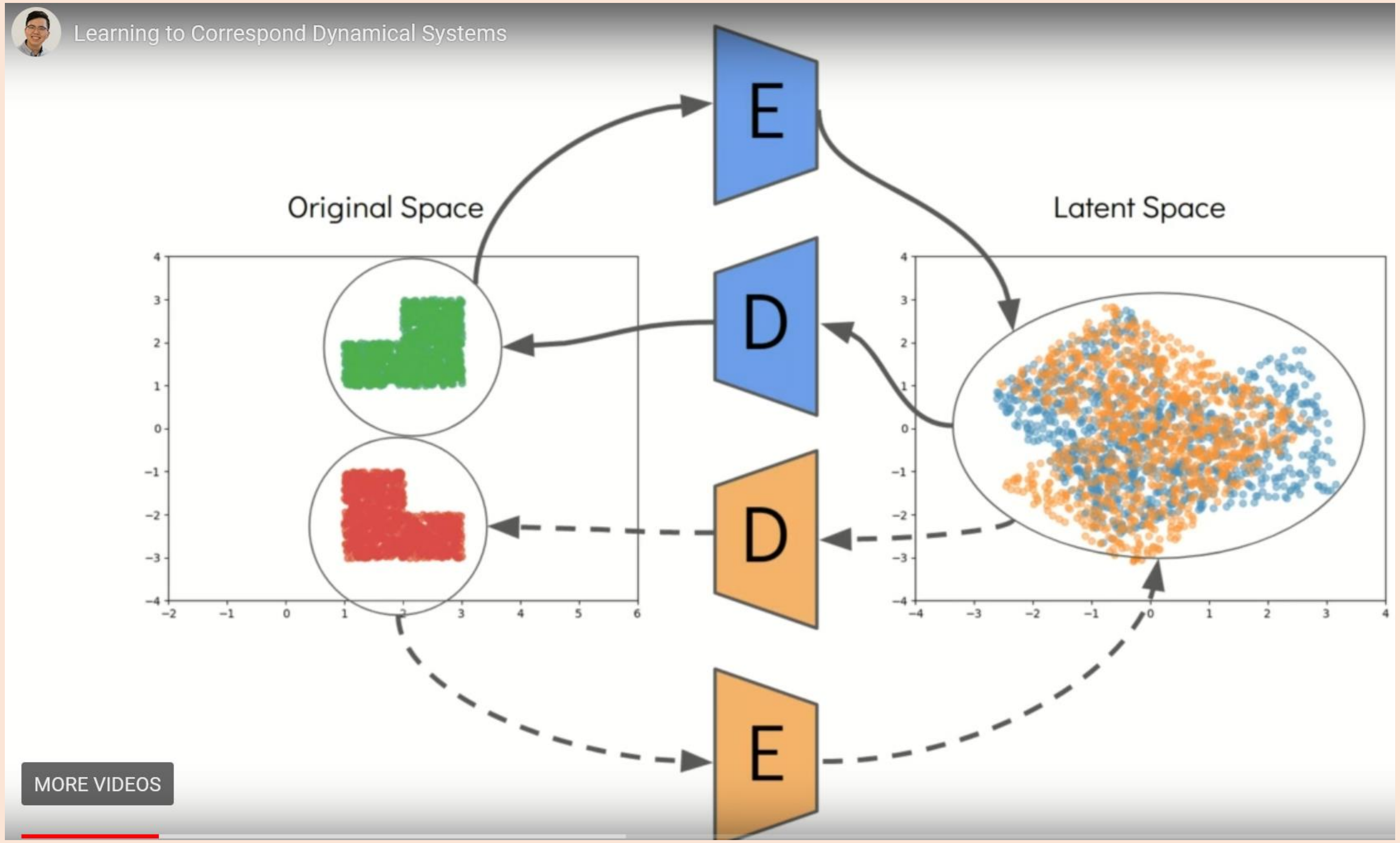
https://github.com/nashory/gans-awesome-applications

Questions?

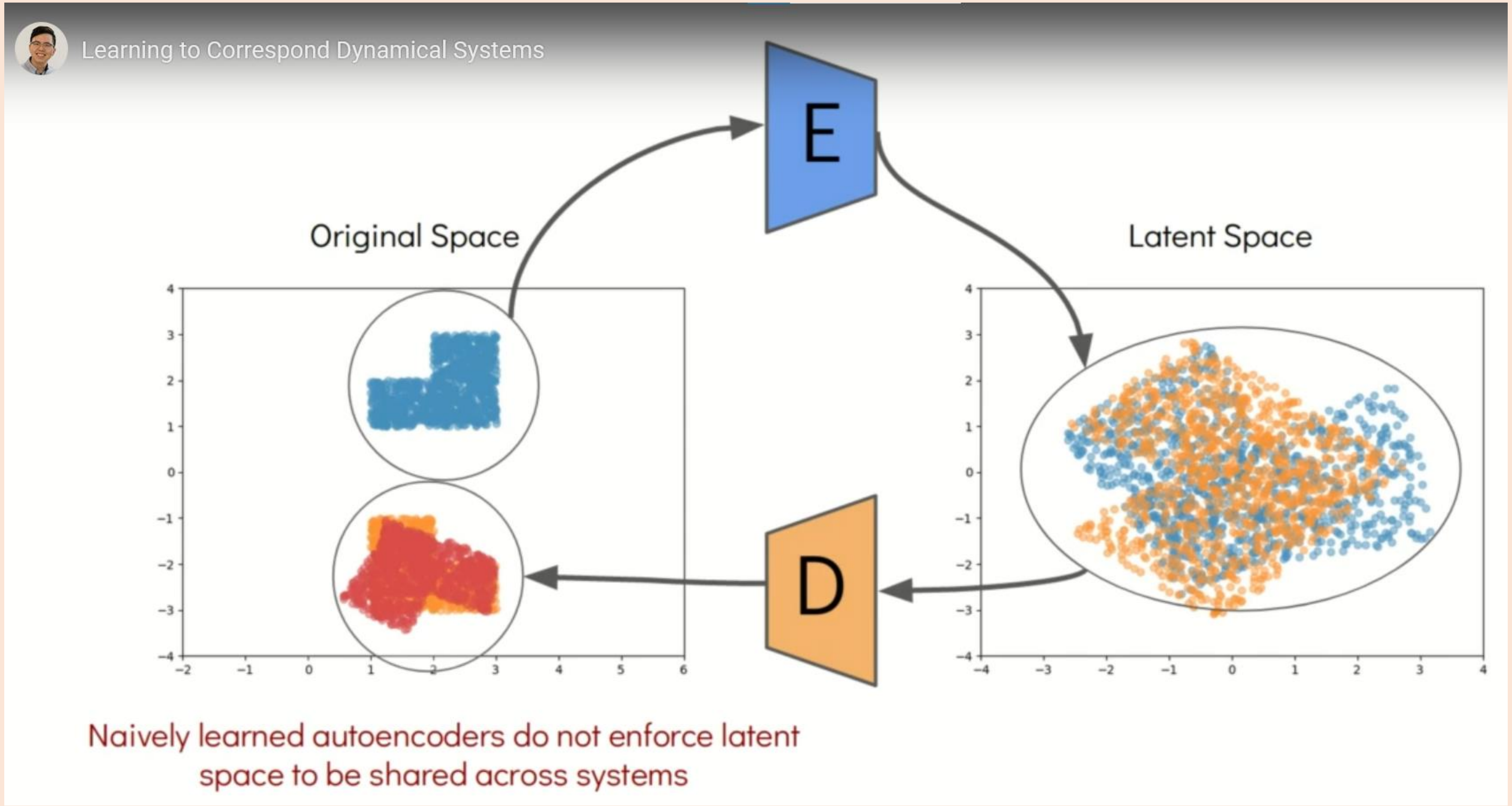
Style Transfer = Latent Space Sharing



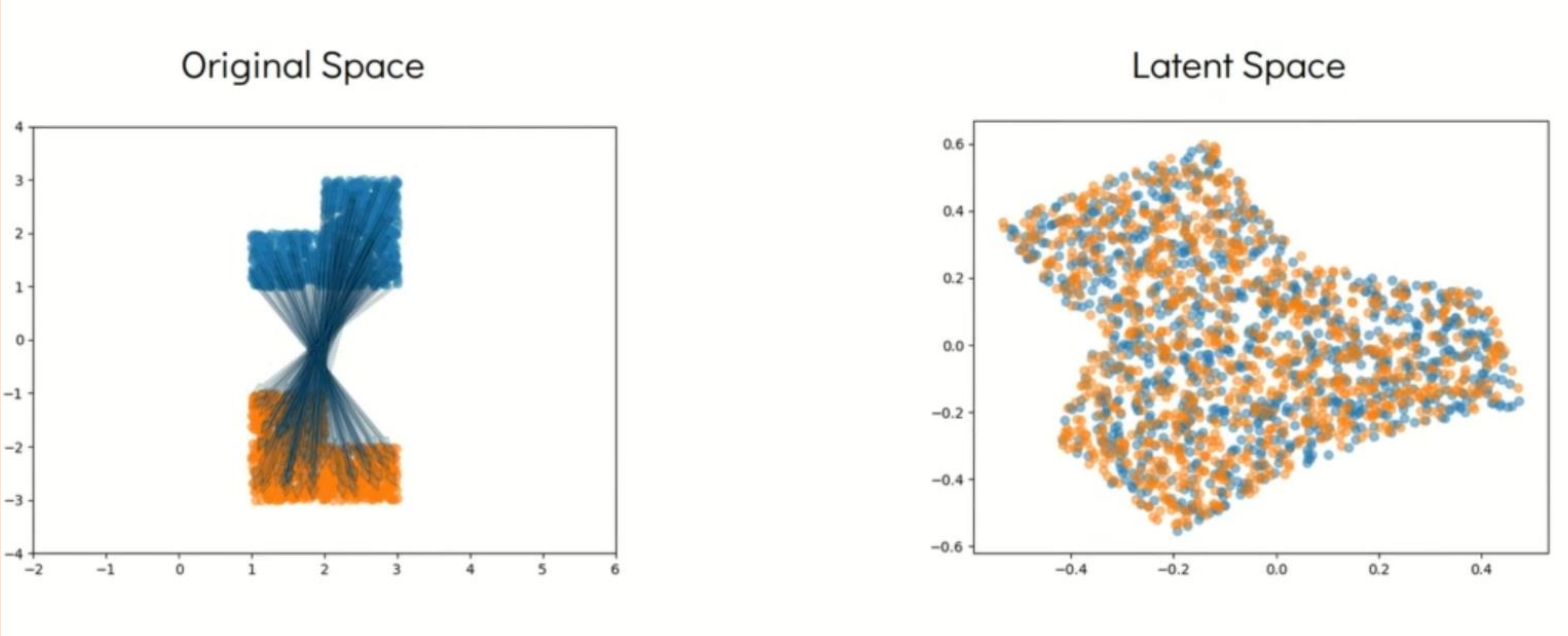
Style Transfer = Latent Space Sharing



Style Transfer = Latent Space Sharing



“Latent Space Fusion”
Enables Cross-Domain Correspondence



Practical Tips

- Top-tier papers from 2020 and onward usually have notebooks. Run the notebooks on Google Colab whenever possible!
- Use pre-trained NN whenever possible!
  - Warm-start is ALWAYS better than cold-start (unless it's bad model/data)
- Always start with reading documentation/code on input/output formats. Massage the data as efficiently as possible!
- Use progress bars!