cape
Campus Asia Plant Environment innovation

Yonsei University, Zhejiang University & Chiba University

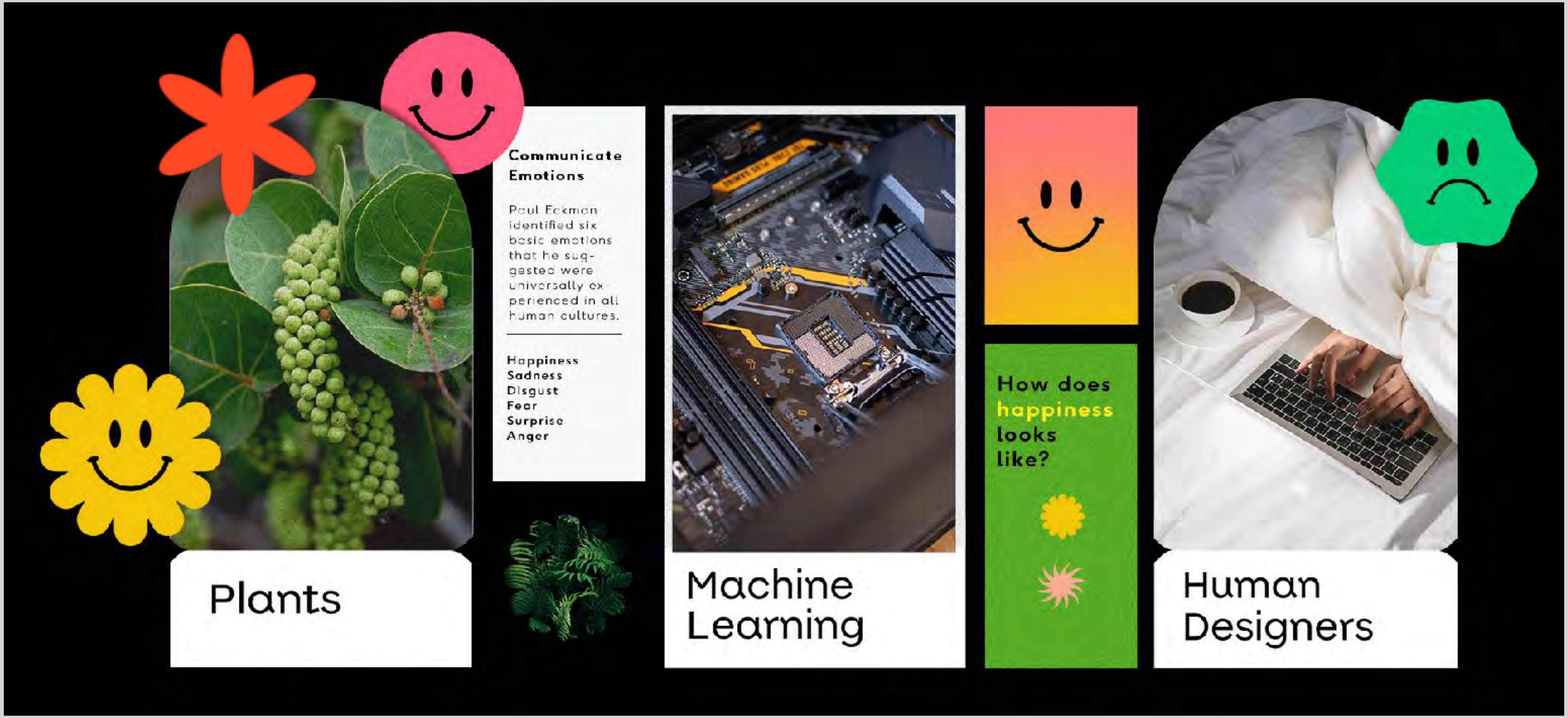
CAPE WS 2021 CAPE WS 2021

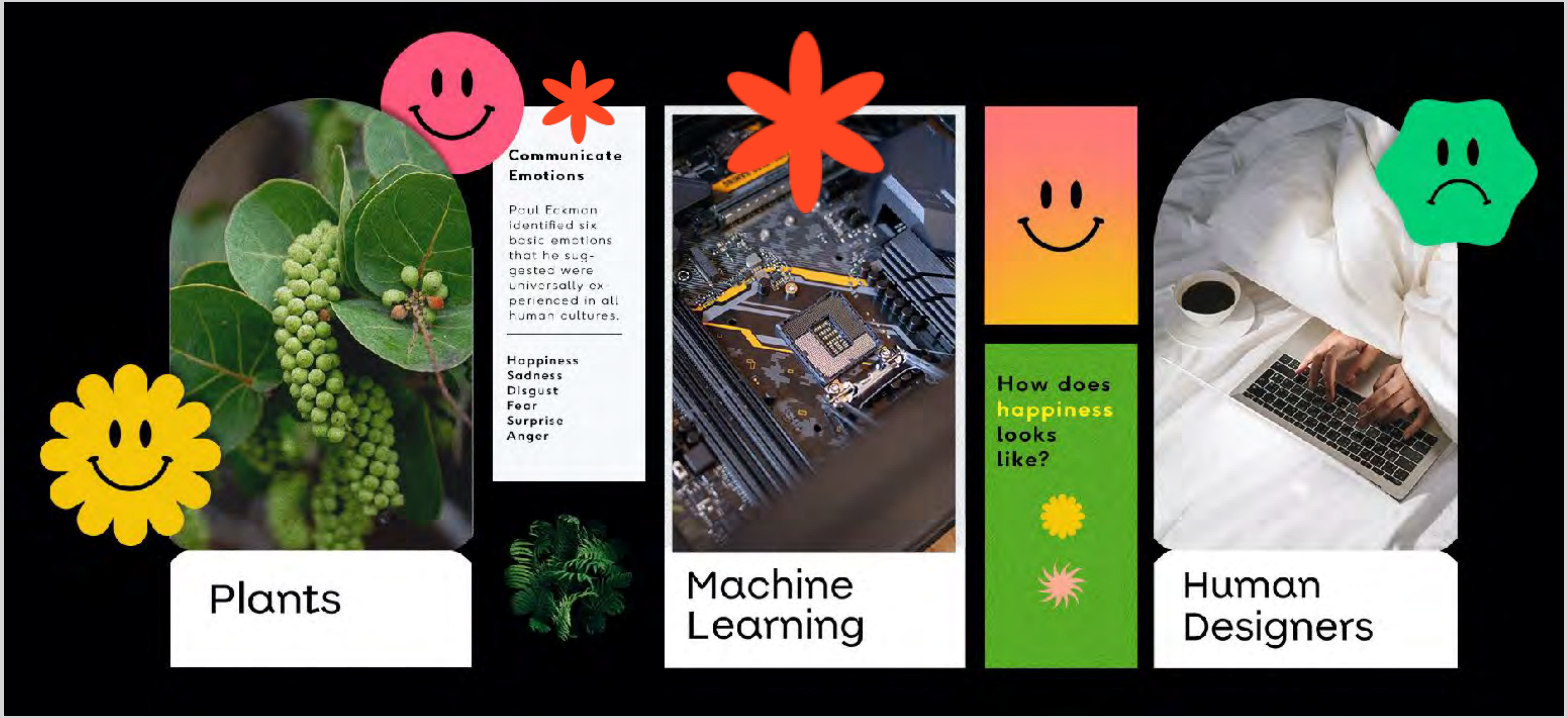
*

Generative Design

Generative Adversarial Networks

Cape Online Workshop 2021

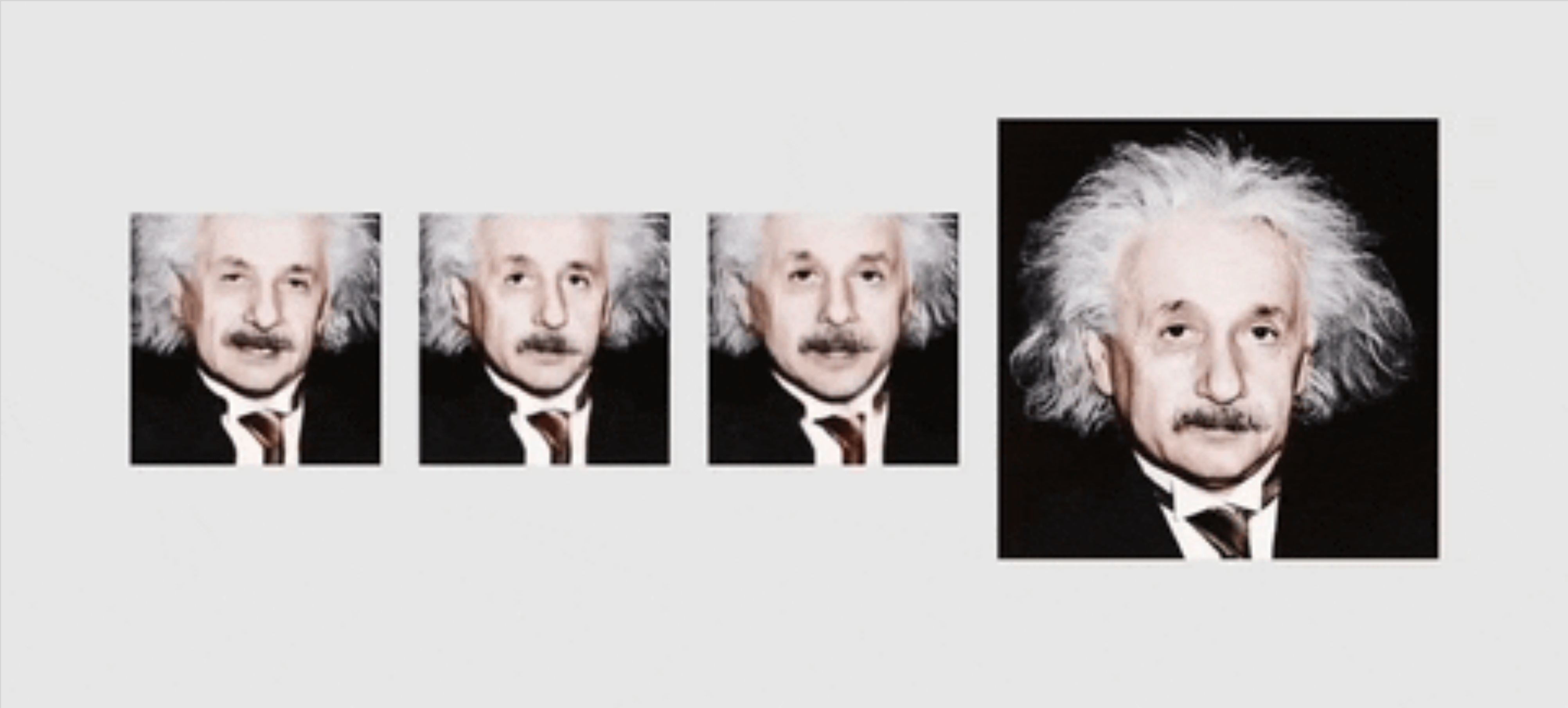

Communicate Emotions
Paul Ekman identified six basic emotions that he suggested were universally experienced in all human cultures.
Happiness
Sadness
Disgust
Fear
Surprise
Anger
Plants
Machine Learning
How does happiness looks like?
Human Designers

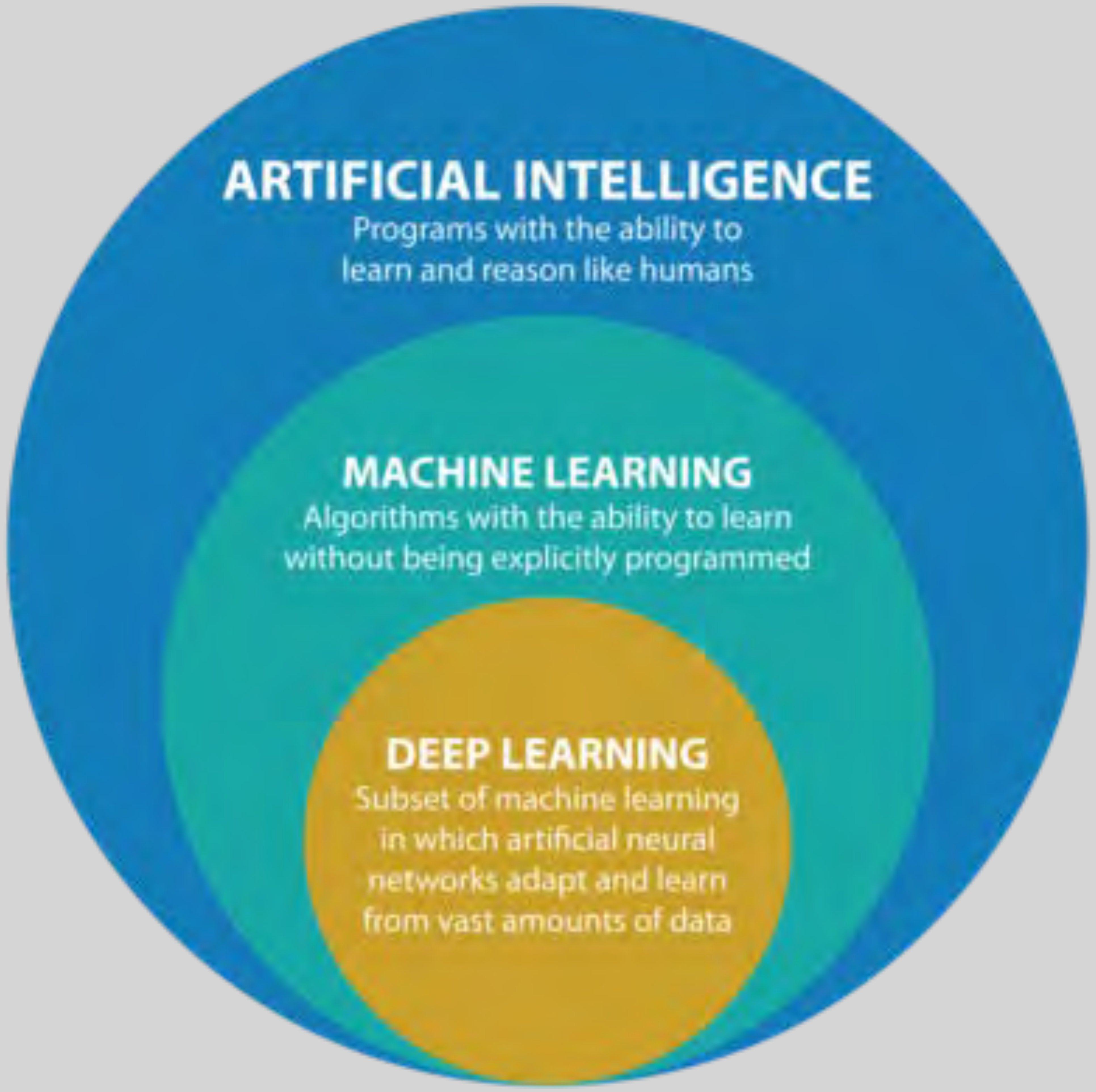

Communicate Emotions
Paul Ekman identified six basic emotions that he suggested were universally experienced in all human cultures.
Happiness
Sadness
Disgust
Fear
Surprise
Anger
Plants
Machine Learning
How does happiness looks like?
Human Designers

As smartphone or computer users we use

AI in a daily basis

We have all heard on seen what AI can do in terms of **image generation**




ARTIFICIAL INTELLIGENCE
Programs with the ability to learn and reason like humans
MACHINE LEARNING
Algorithms with the ability to learn without being explicitly programmed
DEEP LEARNING
Subset of machine learning in which artificial neural networks adapt and learn from vast amounts of data

During this workshop we want you use Machine Learning (GAN) to:

1

Understand the **possibilities** that AI presents for the Design process?

2

Learn not only the theory but also use and **try to generate something**

3

What kind of **opportunities** can this present applied in a services, product or experience?

Our Approach

To use AI as a design tool

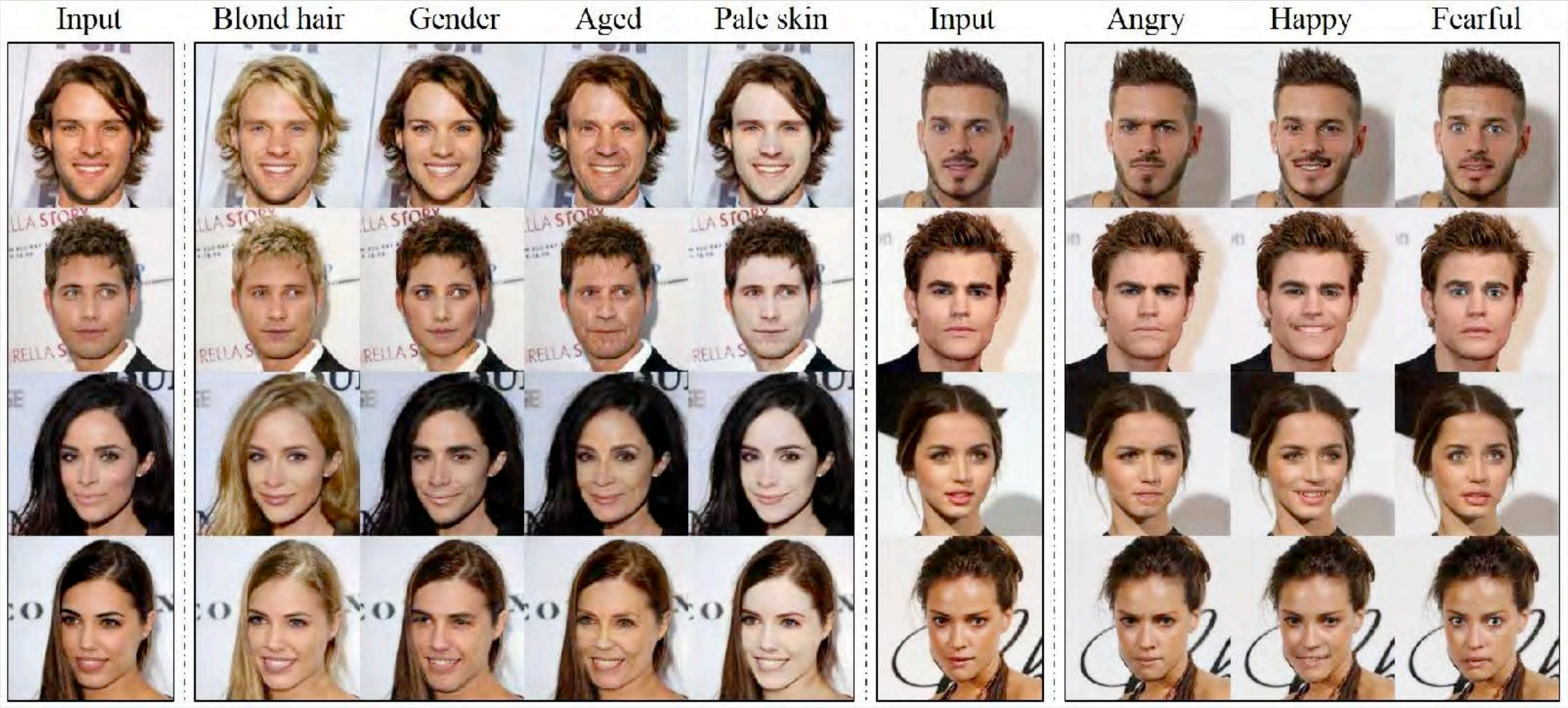
Specifically approaching this from a visual design perspective.

Generative algorithms are part of unsupervised learning techniques.

They underpin one of the **most innovative concepts in machine learning** in the past decade: **Generative Adversarial Networks (GANs).**

A generative model can learn to mimic any distribution of it. Their potential is huge as they can be taught to recreate similar models in any domain. Some of these domains include, but are not limited, to the following:

Images
Music
Speech
Text
Videos



Example of a Deepfake (Source: StarGAN GitHub)

Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis (Chuan Li, Michael Wand 2016)

Colorful Image Colorization (Zhang, Richard and Isola, Phillip and Efros, Alexei A 2016)

DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks (Kupyn, Orest and Budzan, Volodymyr and Mykhailych, Mykola and Mishkin, 2017)

Ai as an Assistant

The **designer** and the **machine** making **decisions together** to expand and improve the results.

Data-driven approach for design ideation, where Liuqing Chen and Pan Wan propose the use of different tools, including generative adversarial networks, to create design concepts by implementing Koesther's biosociation (Chen et al., 2019).

L. Chen et al./J. Vis. Commun. Image R. 61 (2019) 10–22

Fig. 2. Samples of training datasets (left: spoons; right: leaves).

Fig. 3. Randomly synthesized images by proposed GAN model.

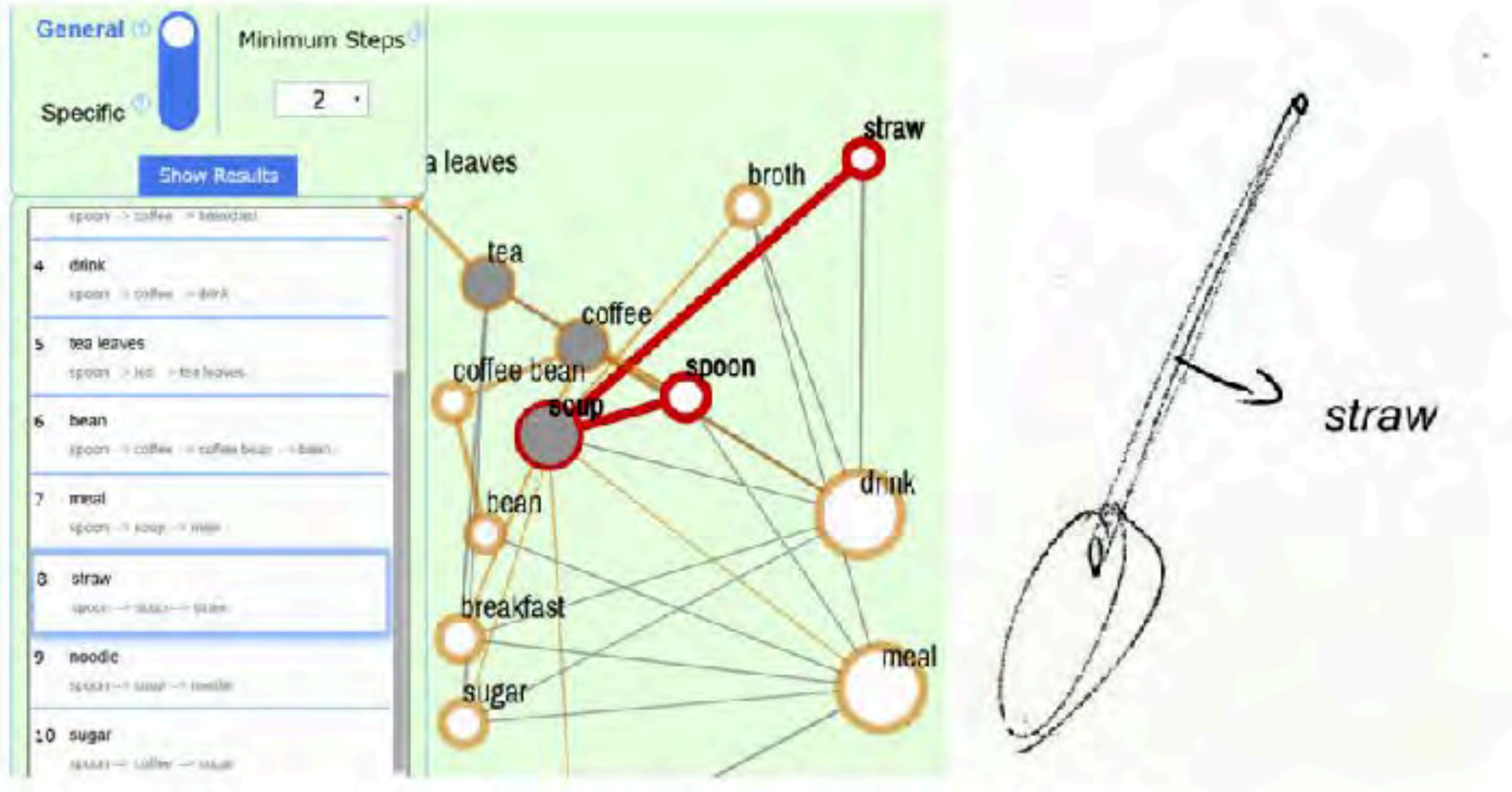


Fig. 7. "straw" was found in the network (left) and the idea was illustrated (right).

The DesING project, as a final example, explores the use of algorithms for creative inspiration in the area of fashion design (Sbai, 2019).

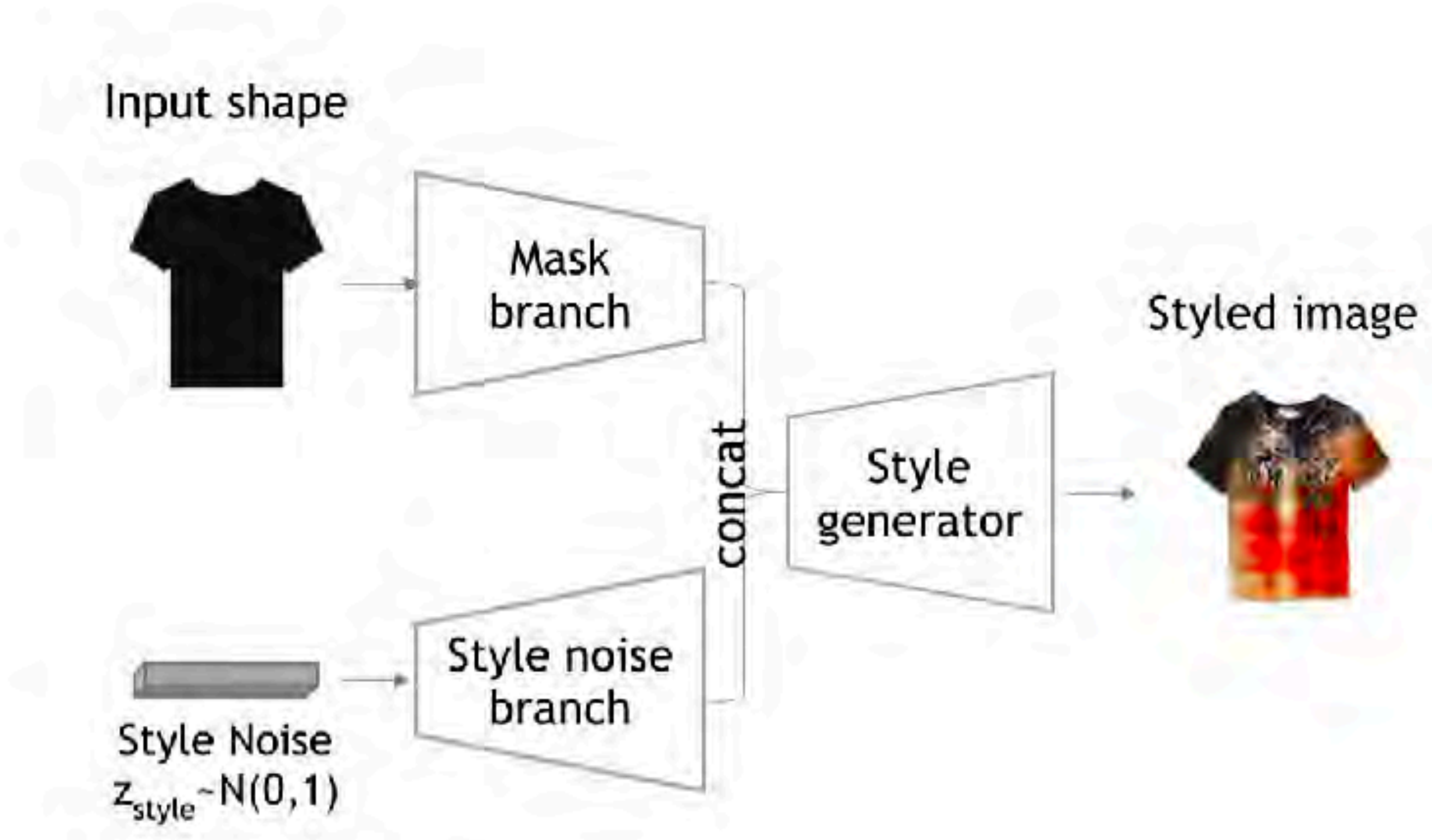


Figure 4: From the mask of a product, our StyleGAN model generates different styled image for each style noise.

The DesING project, as a final example, explores the use of algorithms for creative inspiration in the area of fashion design (Sbai, 2019).

Generated Dataset

The DesING project, as a final example, explores the use of algorithms for creative inspiration in the area of fashion design (Sbai, 2019).

ChAIr Project (Schmitt, 2018), which uses AI to generate chair shapes that designers can reinterpret and then build.

ChAIr Project (Schmitt, 2018), which uses AI to generate chair shapes that designers can reinterpret and then build.

ChAIr Project (Schmitt, 2018), which uses AI to generate chair shapes that designers can reinterpret and then build.

ChAIr Project (Schmitt, 2018), which uses AI to generate chair shapes that designers can reinterpret and then build.

ChAIr Project (Schmitt, 2018), which uses AI to generate chair shapes that designers can reinterpret and then build.

ChAIr Project (Schmitt, 2018), which uses AI to generate chair shapes that designers can reinterpret and then build.

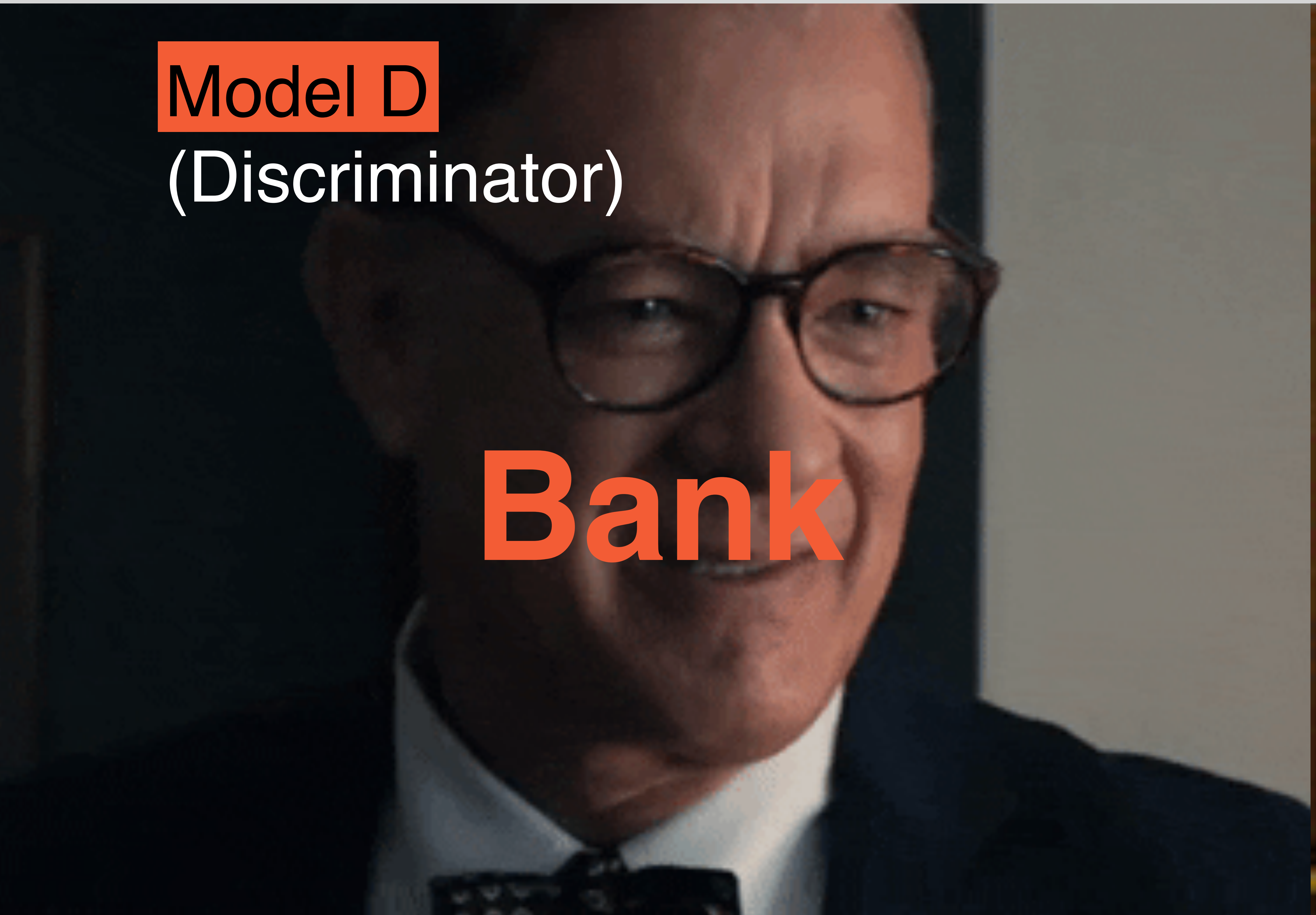
ChAIr Project (Schmitt, 2018), which uses AI to generate chair shapes that designers can reinterpret and then build.

Generative Adversarial Networks / GAN

Is a deep learning framework in which **two models**, a **generative** model G and a **discriminative** model D, are trained simultaneously.

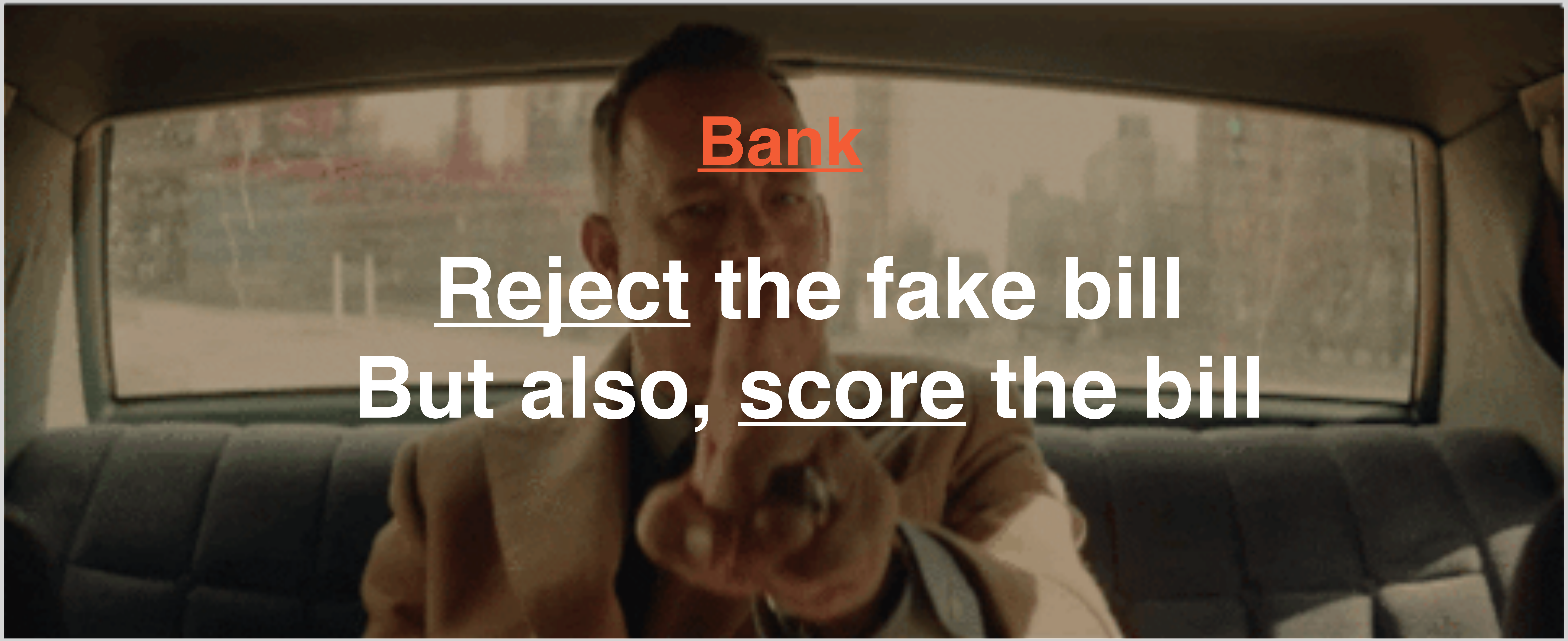

cape
Campus Asia Plant Environment innovation
Yonsei University, Zhejiang University & Chiba University
CAPE WS 2021 CAPE WS 2021
GAN STORY
"Catch me if you can"
Leonardo Dicaprio
As a the "Generator"
Tom Hanks
As a the "Discriminator"

The characters



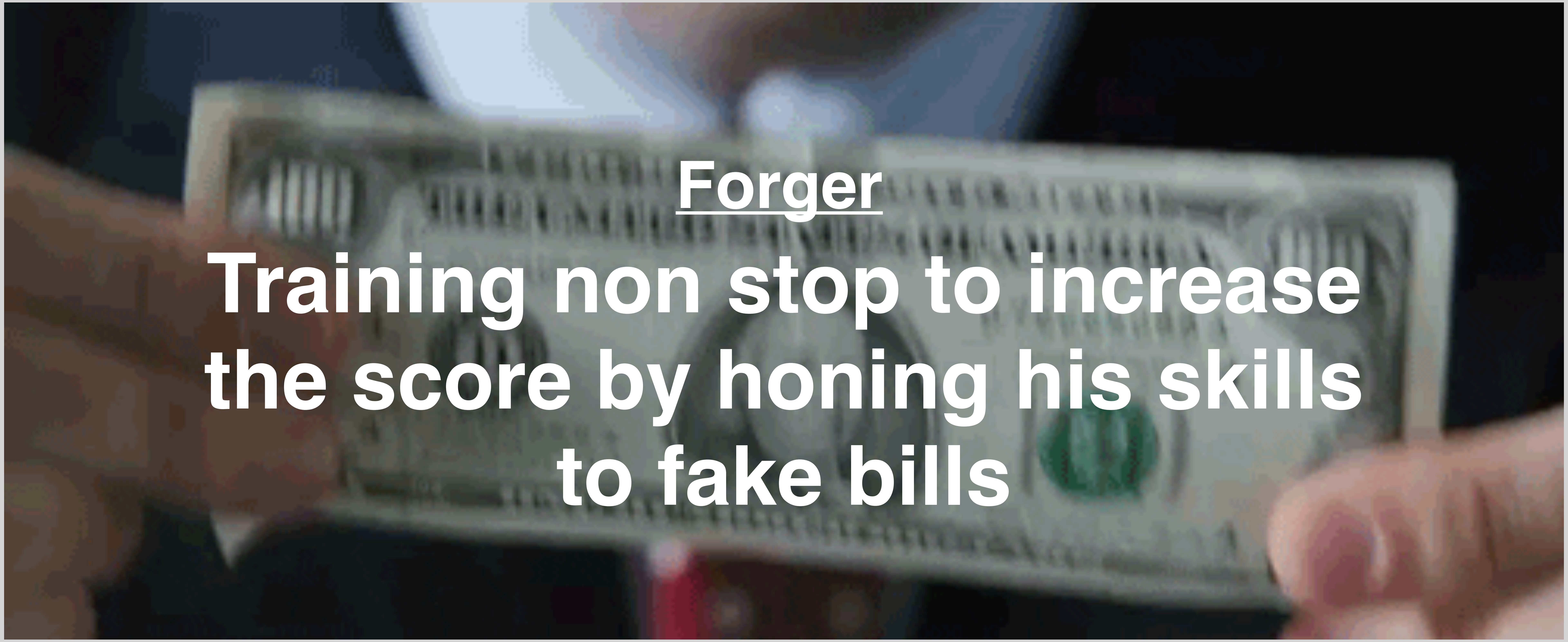
Model G
(Generator)

Model D
(Discriminator)



Model G
(Generator)

Model G
(Generator)



Model D
(Discriminator)


The end

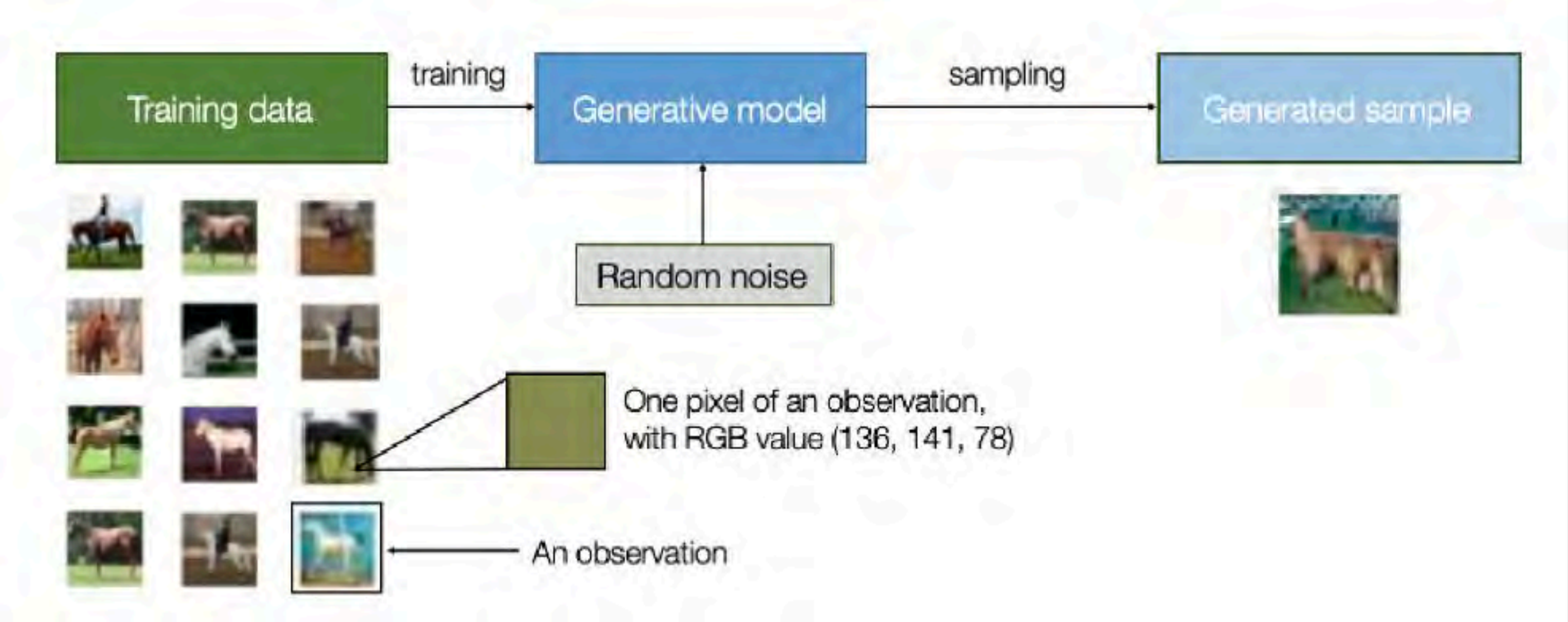
Horse/zebra image translation using a pre-trained DC-GAN

We shall first look at what it means to say that a model is **generative** and learn how it differs from the more widely studied **discriminative modeling.**

Generative Modeling

By sampling from this model, we are able to generate new data.

What is Generative Modeling?

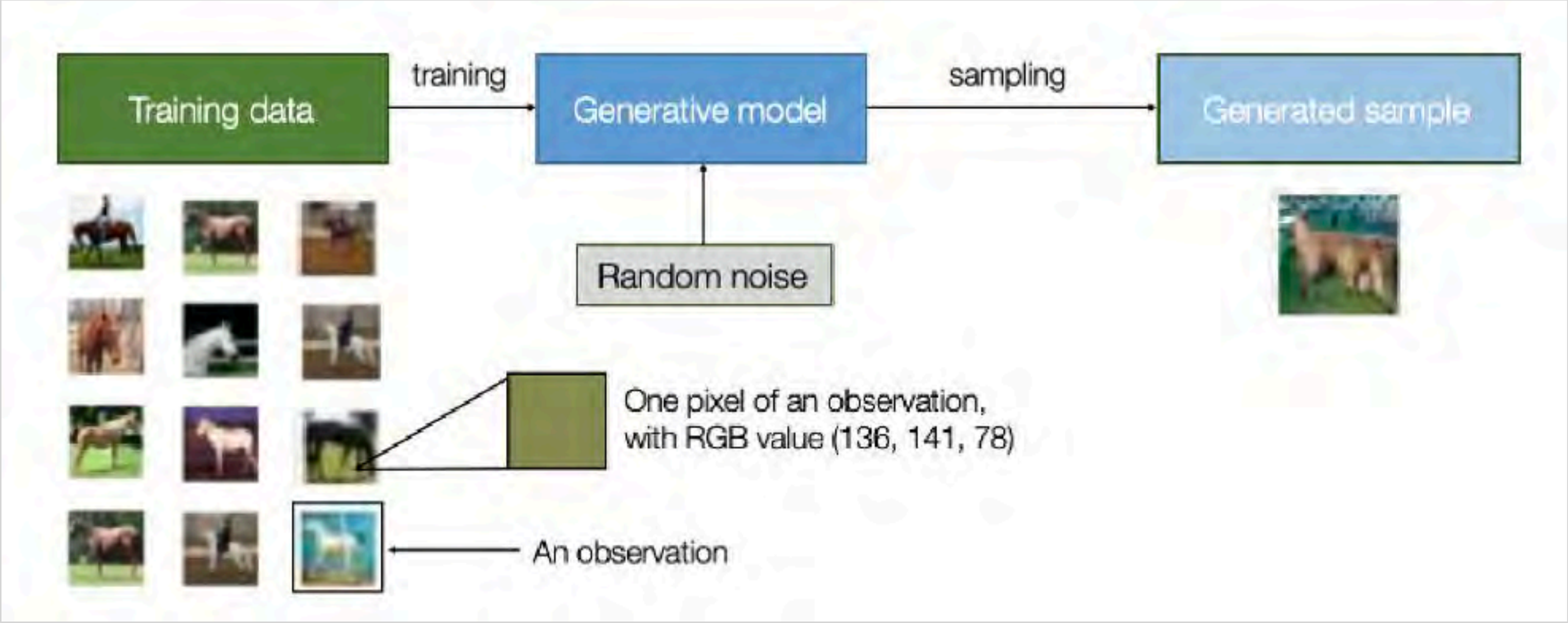


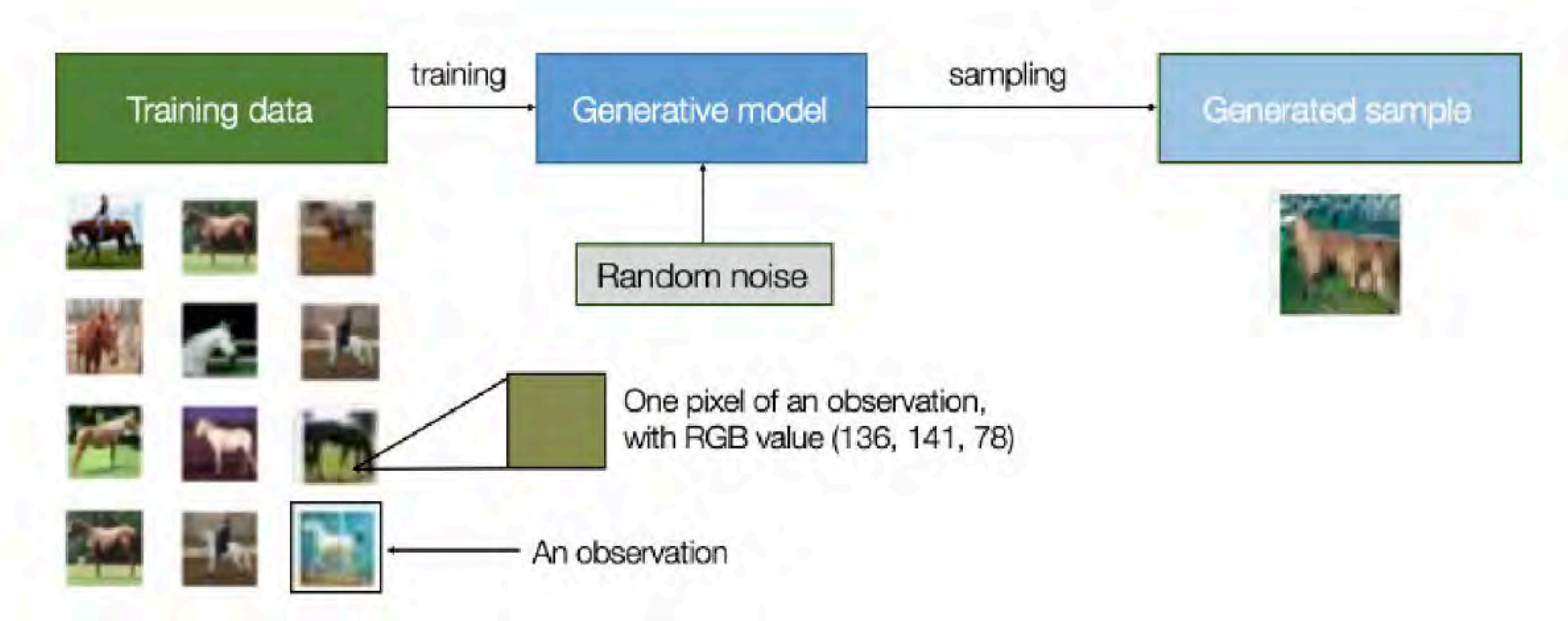
A generative model can be broadly defined as follows: A generative model describes how a dataset is generated, in terms of a probabilistic model.

Suppose we have a dataset containing images of horses.

We may wish to build a model that can generate a new image of a horse that has never existed but still looks real because the model has learned the general rules that govern the appearance of a horse.

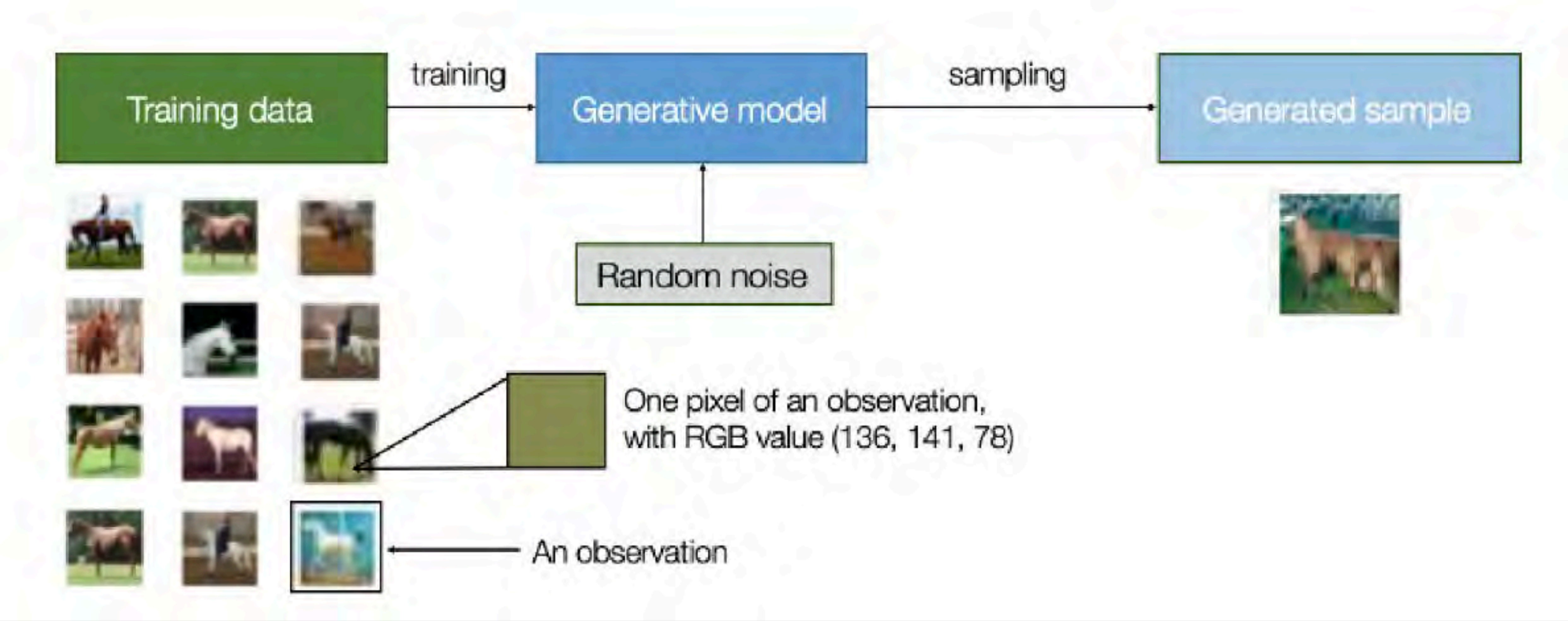
<u>This is the kind of problem that can be solved using generative modeling.</u>





First, we require a **dataset** consisting of many examples of the entity we are trying to generate.

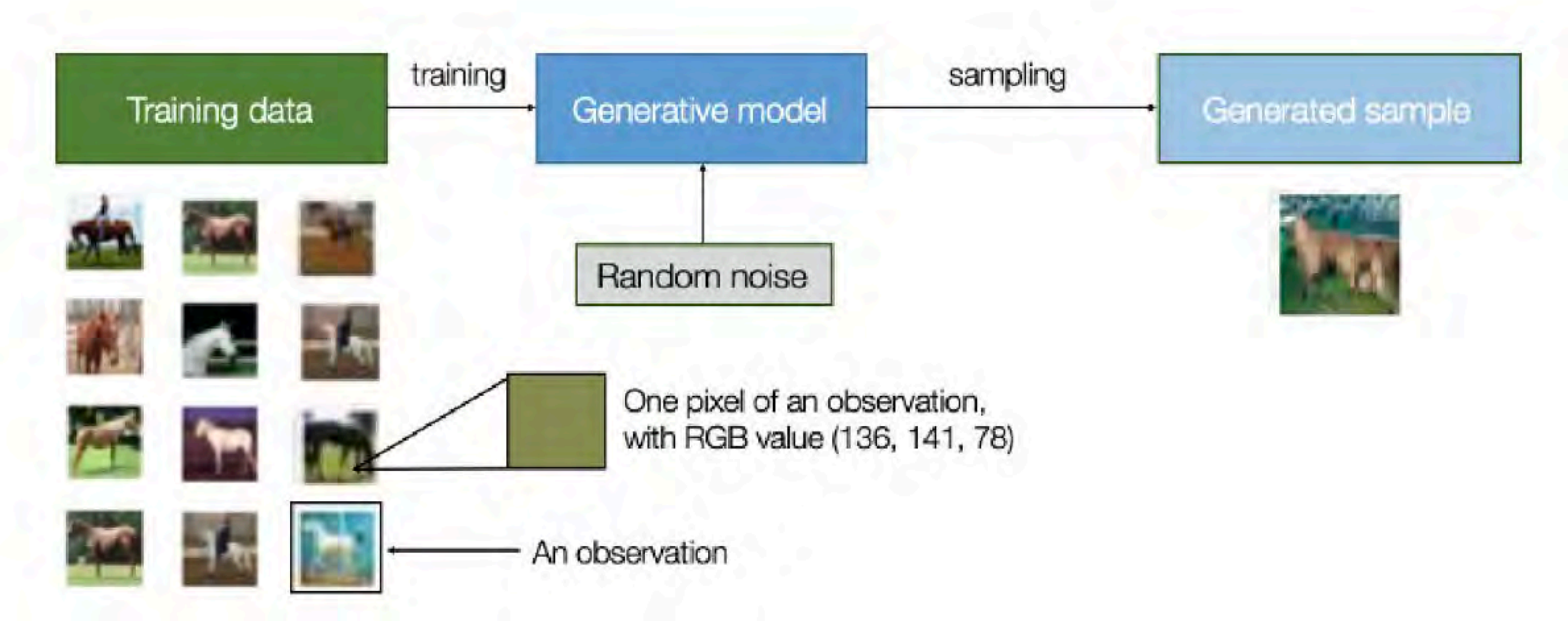
This is known as the **training data**, and one such data point is called an observation.



A generative model must also be probabilistic rather than deterministic.

If our model is merely a fixed calculation, such as taking the average value of each pixel in the dataset, **it is not generative** because the model produces the same output every time.

The model must include a stochastic (random) element that influences the individual samples generated by the model.



In other words, we can imagine that there is some unknown probabilistic distribution that explains why some images are likely to be found in the training dataset and other images are not.

It is our job to build a model that mimics this distribution as closely as possible and then sample from it to generate new images.

Generative vs Discriminative Modeling

Generative vs Discriminative Modeling

In order to comprehend generative algorithms, it can be helpful to contrast them with discriminative algorithms.

When input data is fed into a discriminative algorithm, it aims to predict the label to which the data belongs. The algorithm aims to map features to labels.

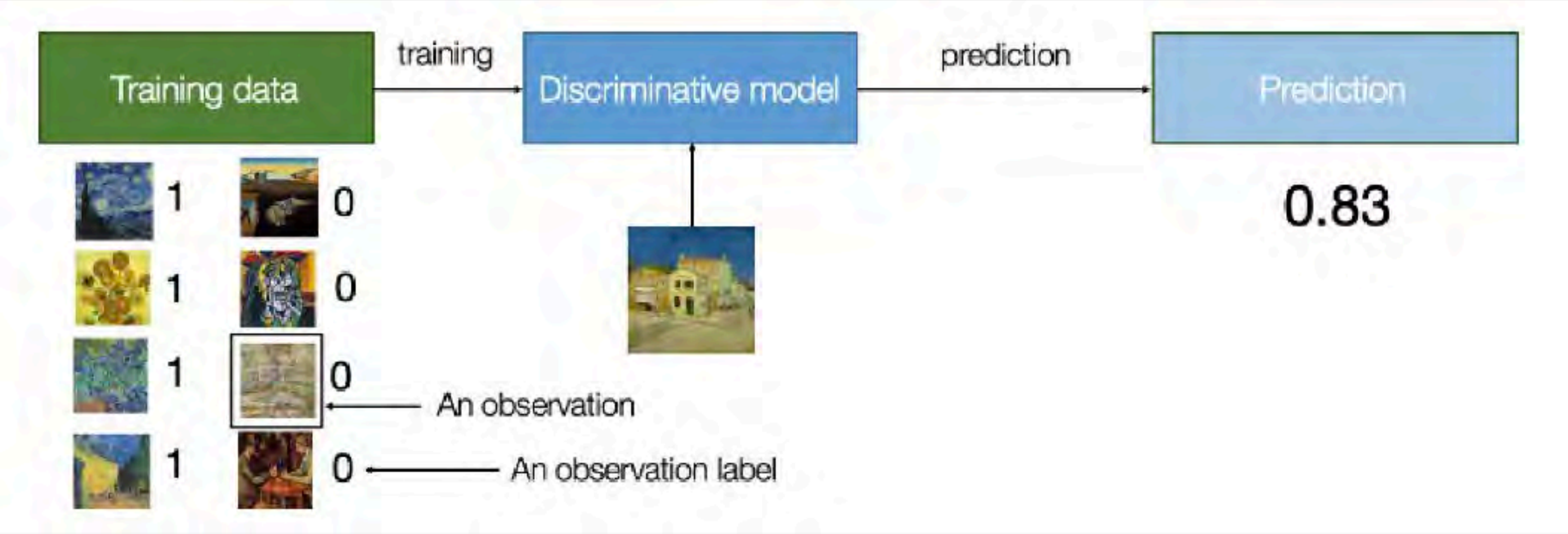
Generative algorithms, on the other hand, do the opposite; they aim to predict features given a certain label.

To truly understand what generative modeling aims to achieve and why this is important, it is useful to compare it to its counterpart, discriminative modeling.

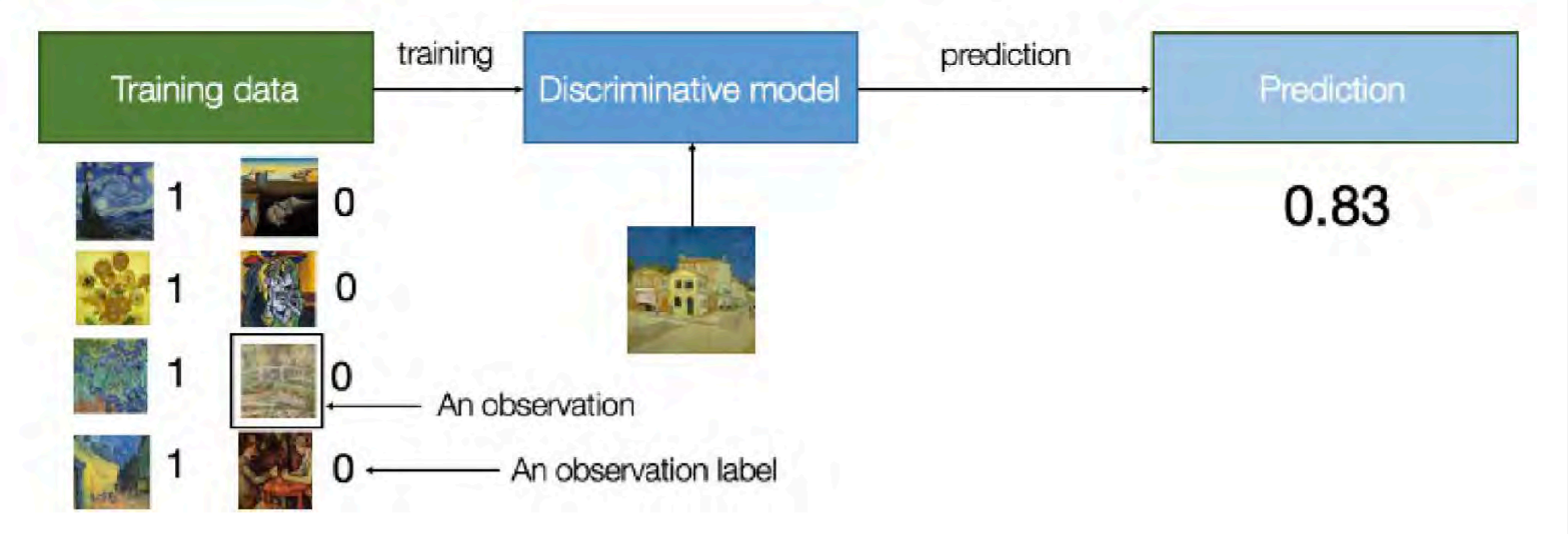
Discriminative modeling
Example:

Suppose we have a dataset of paintings, some painted by Van Gogh and some by other artists.

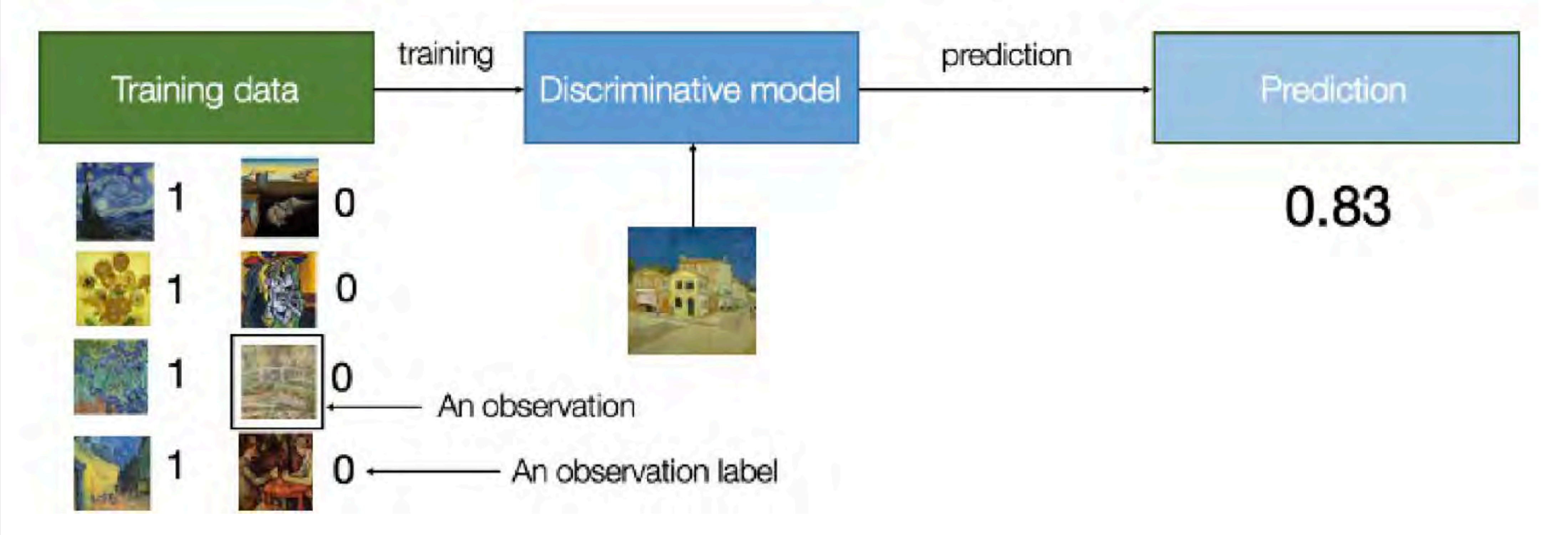
With enough data, we could train a **discriminative** model to predict if a given painting was painted by Van Gogh.



Our model would learn that certain colors, shapes, and textures are more likely to indicate that a painting is by Van Gogh, and for paintings with these features, the model would upweight its prediction accordingly.



Our model then learns how to discriminate between these two groups and outputs the probability that a new observation has label 1 that it was painted by Van Gogh.



For this reason, discriminative modeling is synonymous with **<u>supervised learning,</u>** or learning a function that maps an input to an output using a labeled dataset.

Generative modeling is usually performed with an **<u>unlabeled dataset</u>** (that is, as a form of unsupervised learning), though it can also be applied to a labeled dataset to learn how to generate observations from each distinct class.

Let's take a look at some mathematical notation to describe the difference between generative and discriminative modeling.

Discriminative modeling estimates $p(y|\mathbf{x})$—the probability of a label y given observation $\mathbf{x}$.

Generative modeling estimates $p(\mathbf{x})$—the probability of observing observation $\mathbf{x}$.

If the dataset is labeled, we can also build a generative model that estimates the distribution $p(\mathbf{x}|y)$.

In other words, **Discriminative Modeling** attempts to estimate the probability that an observation x belongs to category y.

Generative modeling doesn't care about labeling observations. Instead, it attempts to estimate the probability of seeing the observation at all.

Discriminative Modeling

The key point is that even if we were able to build a perfect discriminative model to identify Van Gogh paintings, it would still have no idea how to create a painting that looks like a Van Gogh.

Generative Modeling

We would instead need to train a generative model, which can output sets of pixels that have a high chance of belonging to the original training dataset.

Generative Adversarial Networks (GAN)

GANs are comprised of two neural networks: a generator and a discriminator.

They are able to generate new, synthetic data.

The generator outputs new instances of the data, while the discriminator determines whether each instance of the data belongs to the training dataset.

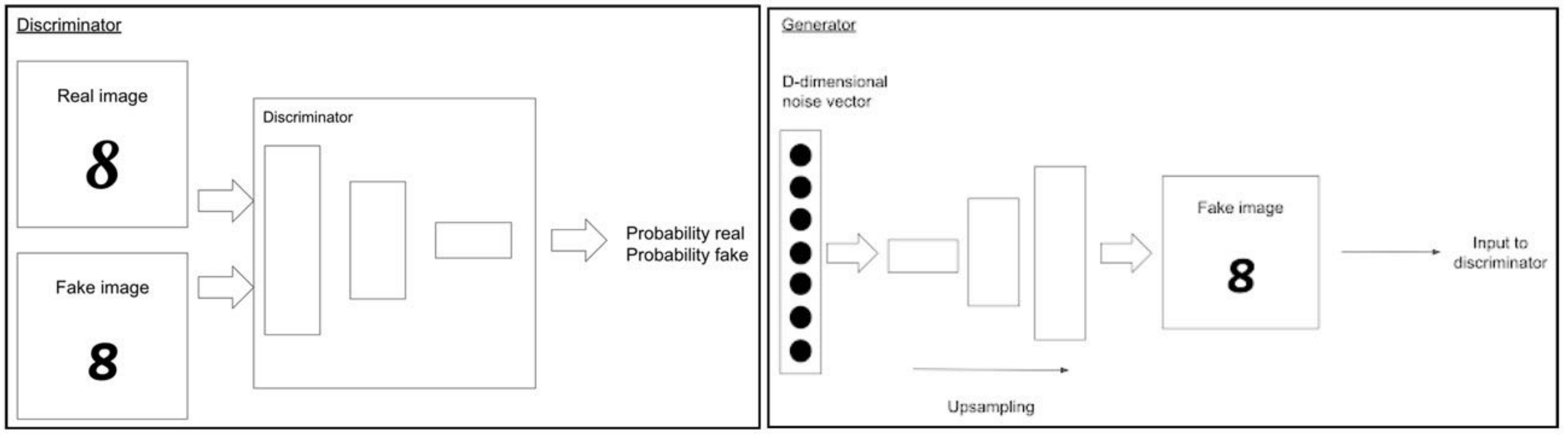
The following image gives an illustration of the output from a GAN datasets.The images on the far-right side of the grid are the true values and the others are generated by the model:

The following are the steps that a GAN takes:

1. Random numbers are fed into the generator and an image is generated

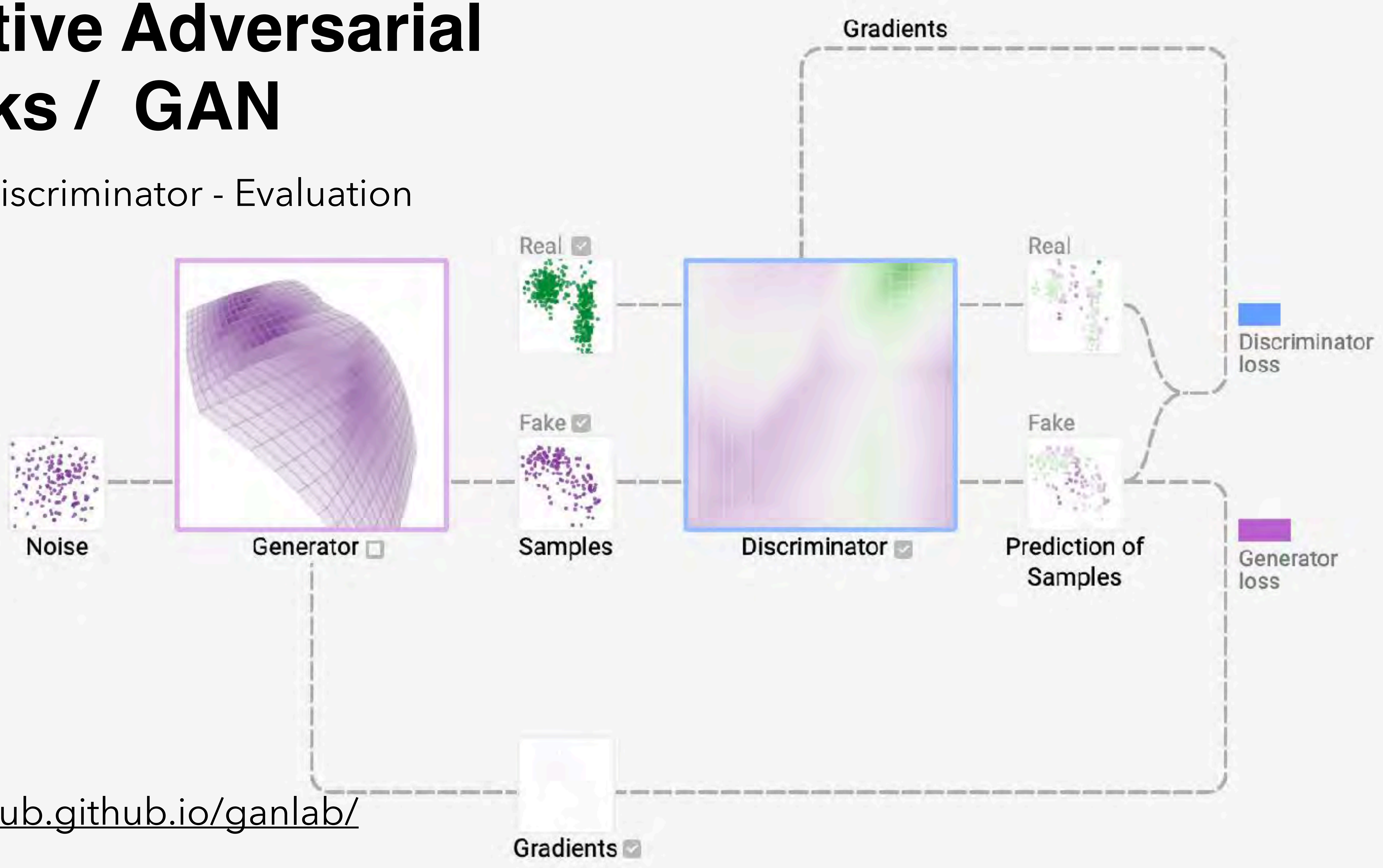
2. The generated image is fed into the discriminator along with other images taken from the real dataset

3.The discriminator considers all of the images fed into it and returns a probability as to whether it thinks the image is real or fake



Generative Adversarial Networks / GAN

Generator - Discriminator - Evaluation



https://poloclub.github.io/ganlab/

Training Generative Adversarial Networks (GANs)

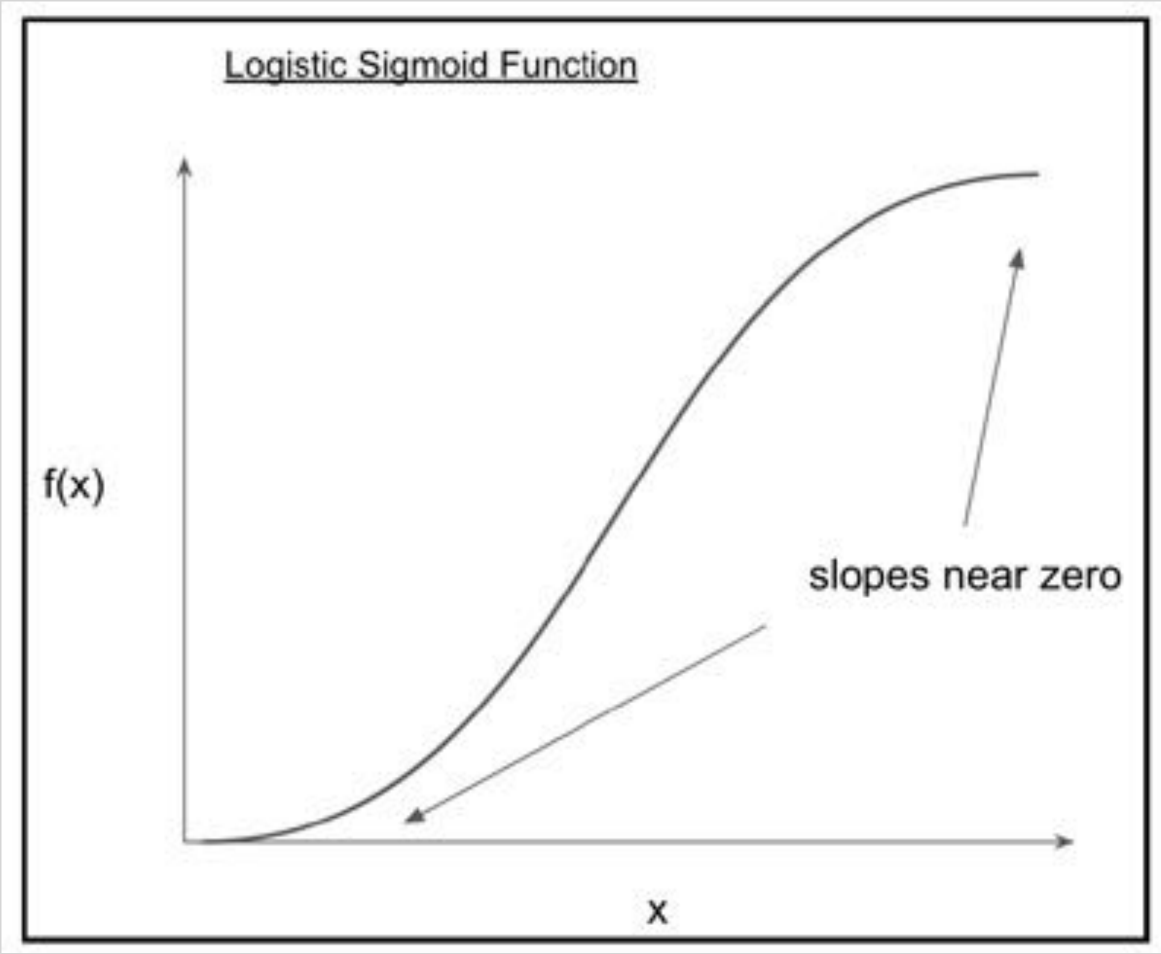
The training process of a GAN is as follows:
Training the discriminator: Generator values should be held constant
Training the generator: The discriminator should be pretrained against the original dataset

Either side of the GAN can overpower the other if one is significantly higher-performing than the other.

For example, if the discriminator is too high-performing, it will return a value that is very close to either 0 or 1, and the generator will have difficulties reading the gradient.

On the other hand, if the performance of the generator is too high, then it will continually exploit weaknesses in the discriminator that lead to false negatives.

Such a dilemma during training is illustrated in the following graph for the case of a binary classification problem. We want to stop one side of the GAN from winning so that both sides can continue to learn together for an extended period of time:



GAN challenges

GANs often face the following major challenges:

Mode collapse: When the generator collapses and produces limited varieties of samples

Diminished gradient: When the discriminator performs too well that the generator gradient vanishes and learns nothing

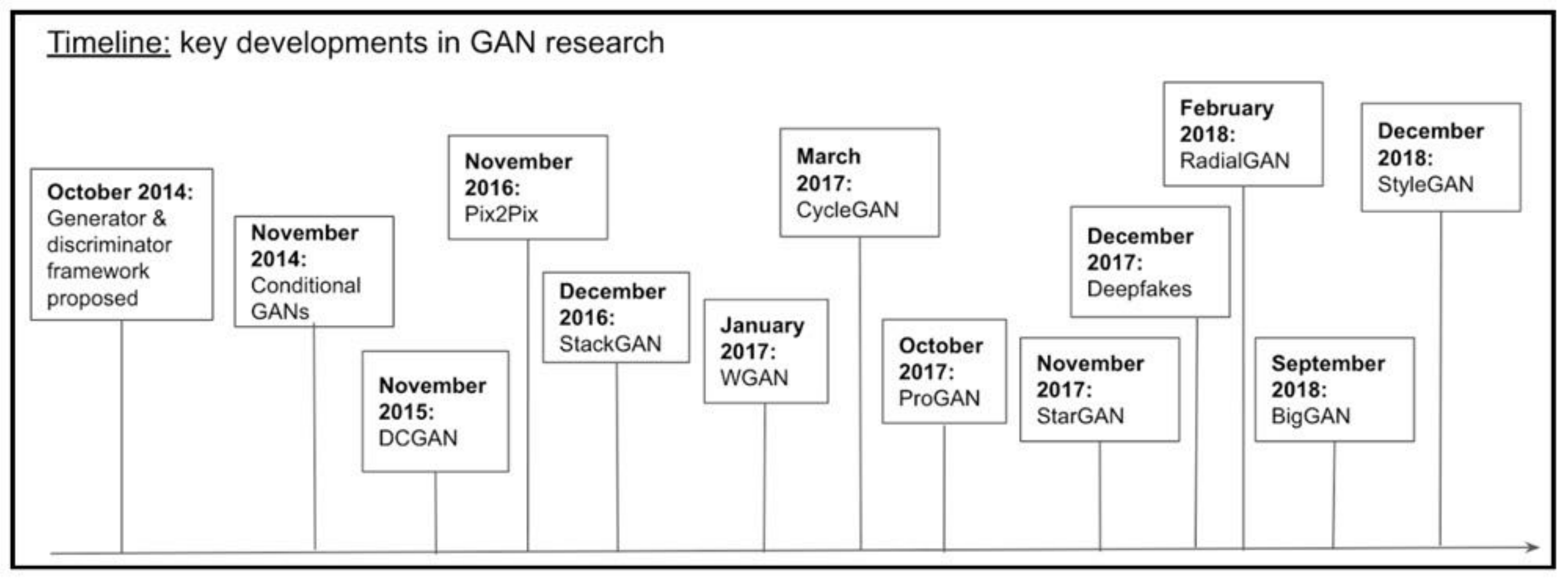
High sensitivity: They are highly sensitive to the hyperparameter selections

Overfitting: An imbalance between the generator and discriminator can cause overfitting

Long training time: If the model is trained on a GPU, it can take hours

Non-convergence: The models may never converge, as the parameters can oscillate and destabilize

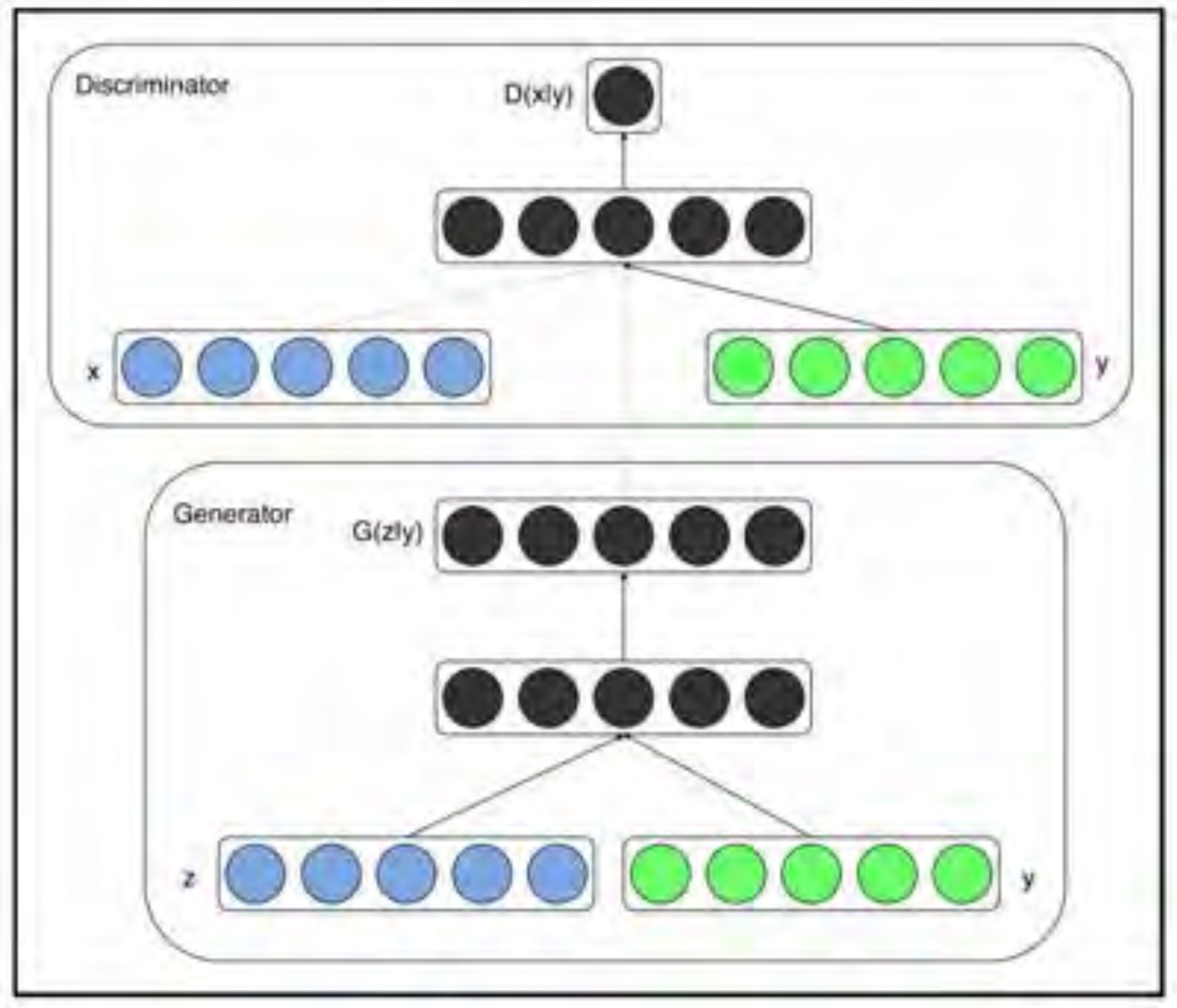
GAN variations and timelines



Conditional GANs

Conditional GANs are a central theme that form the building blocks of many state-of-theart GANs.
The paper submitted by Mirza and Osindero in 2014 shows how integrating the class labels of data yields greater stability in GAN train.

This idea of conditioning GANs with prior information is a common approach in future GAN research. It is particularly important for papers whose main focus is on image-to-image or text-to-image applications:



Conditional GANs Sample

In this sample we can create automated tagging of images, with multi-label predictions, using conditional adversarial nets to generate a (possibly multi-modal) distribution of tag-vectors conditional on image features. For the experiments they use MIR Flickr 25,000 dataset , and extract the image and tags features using the convolutional model and language model we described above.The first 150,000 examples were used as training set.

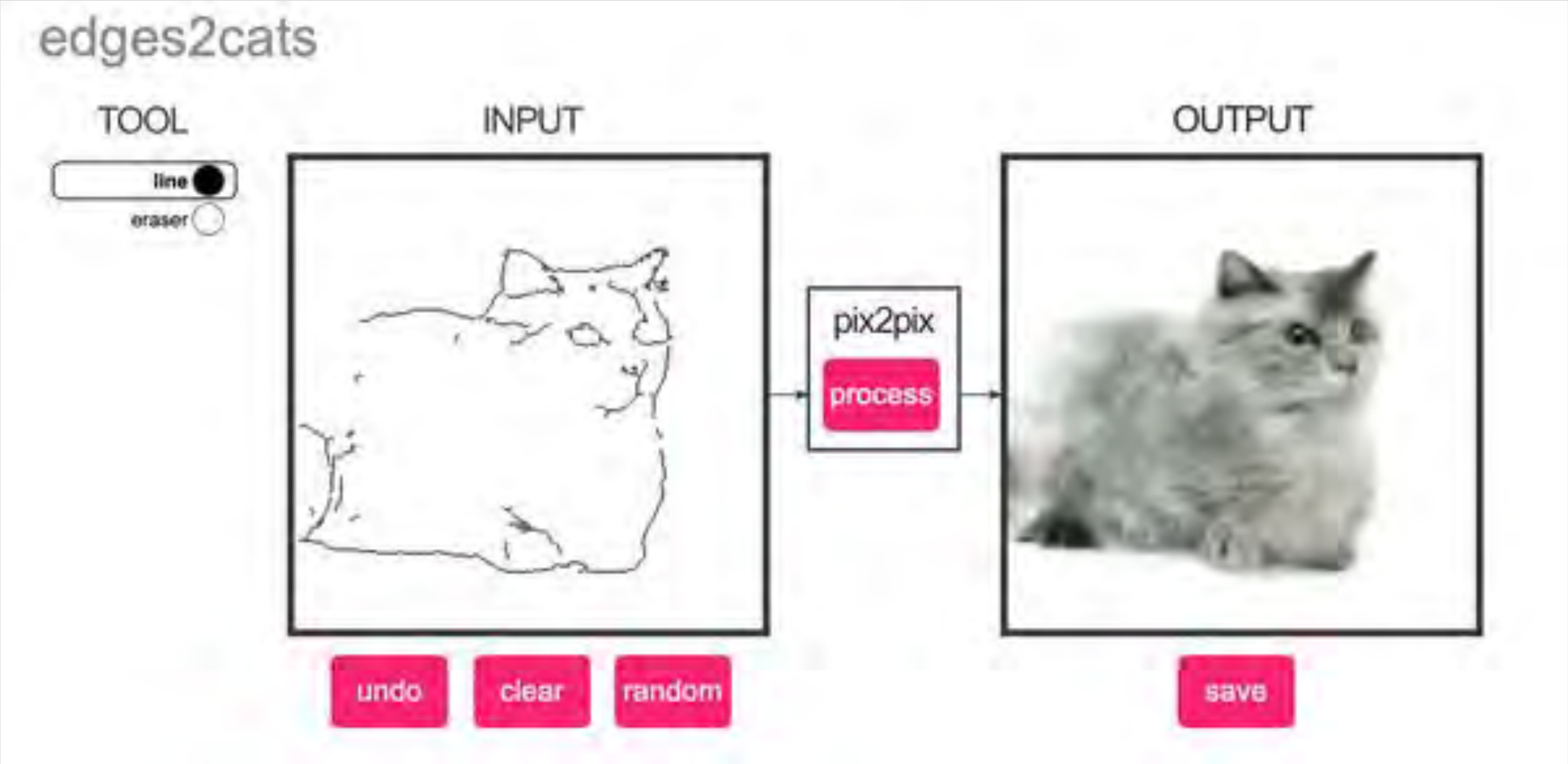
	User tags + annotations	Generated tags
	montanha, trem, inverno, frio, people, male, plant life, tree, structures, transport, car	taxi, passenger, line, transportation, railway station, passengers, railways, signals, rail, rails
	food, raspberry, delicious, homemade	chicken, fattening, cooked, peanut, cream, cookie, house made, bread, biscuit, bakes
	water, river	creek, lake, along, near, river, rocky, treeline, valley, woods, waters
	people, portrait, female, baby, indoor	love, people, posing, girl, young, strangers, pretty, women, happy, life

Pix2Pix translation GAN model

Another image-to-image translation GAN model is Pix2Pix. It has many applications such as edge maps to photo-realistic images, and black and white to colour. The architecture uses paired training samples and incorporates a PatchGAN. The PatchGAN looks at individual regions of the image in order to determine whether they are real or fake, versus when considering the whole image.

The following are some examples of images generated by implementing the Pix2Pix GAN:

Mapping the edges of a handbag to a full realistic image is as follows:

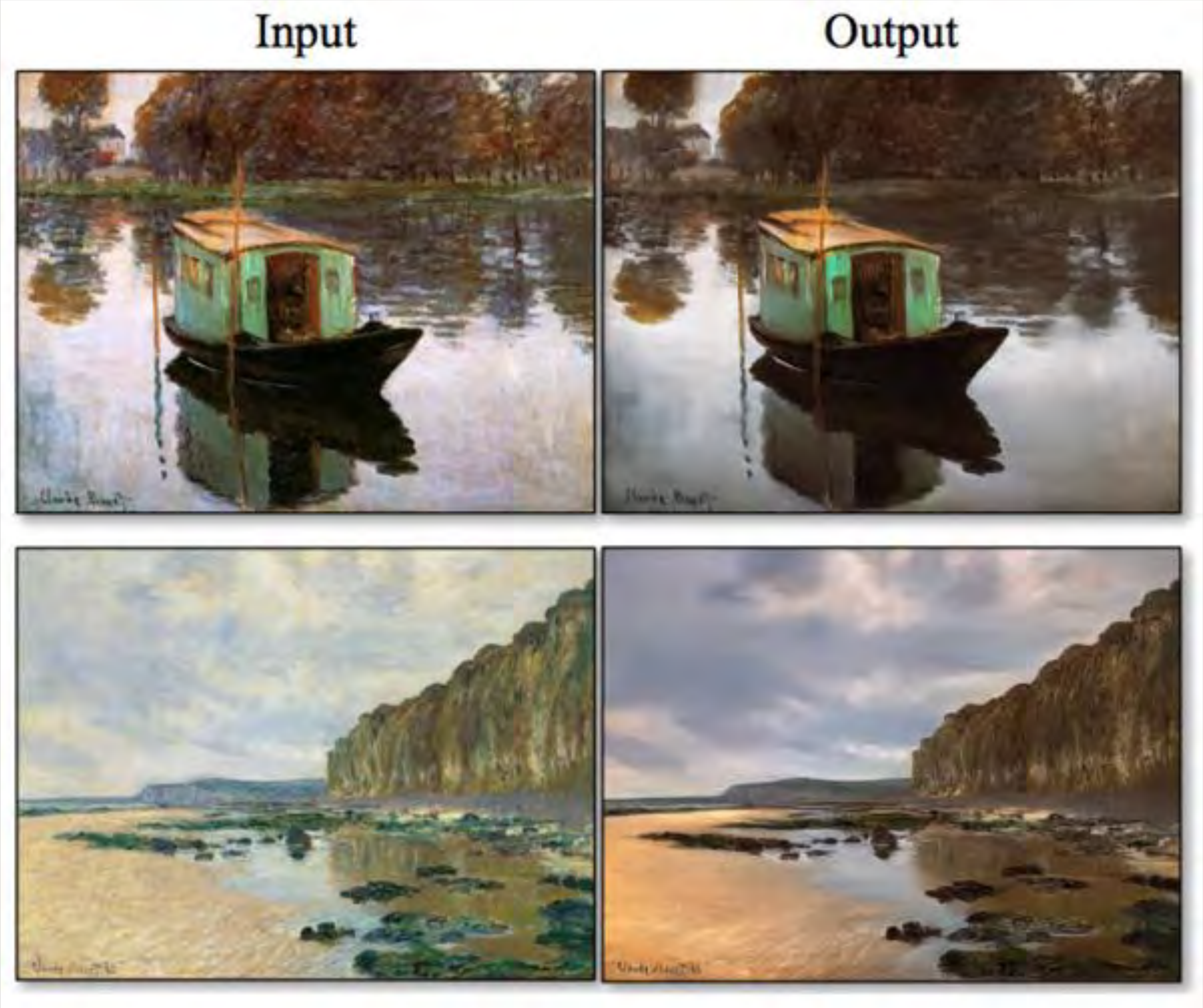


https://affinelayer.com/pixsrv/index.html

CycleGAN

There are many applications of CycleGAN, such as enhanced image resolution and style transfer. More specifically, it can be used for converting pictures to paintings (and vice versa), along with mapping one animal to another, as shown in the following example screenshots. The following are the examples cited in the CycleGAN paper on implementing CycleGAN:

Mapping Monet's paintings to a photographic style looks as follows:



Previous Work with GAN

Video & Motion Graphics Through Latent Walks

Different Projects Based our Students Interest.

Previous Projects

Logo-icon Generator

Font Generator

Icon-logo Project

Commission

Creation of logo for a Japanese tea shop (Branding design)

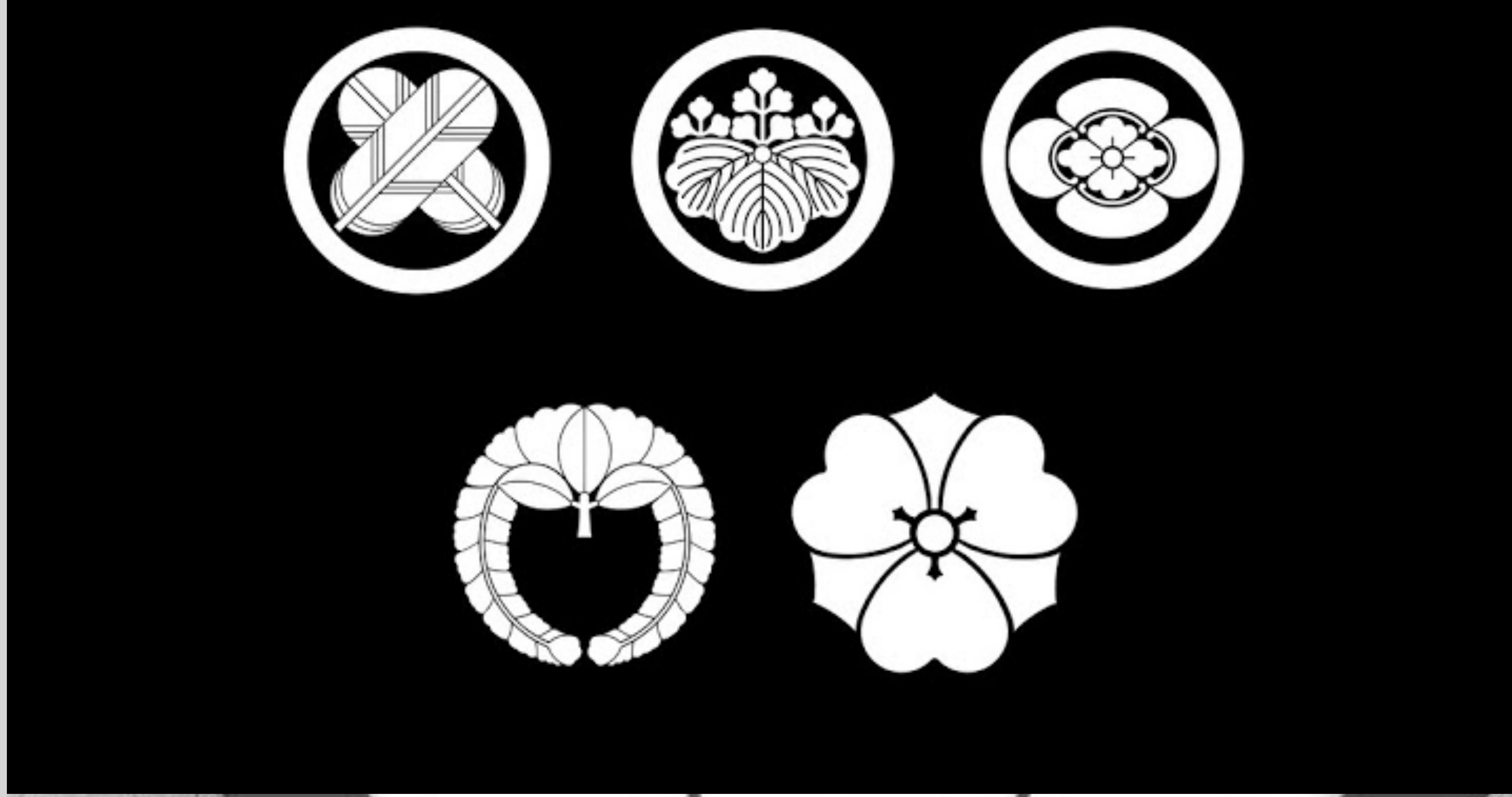
Objective

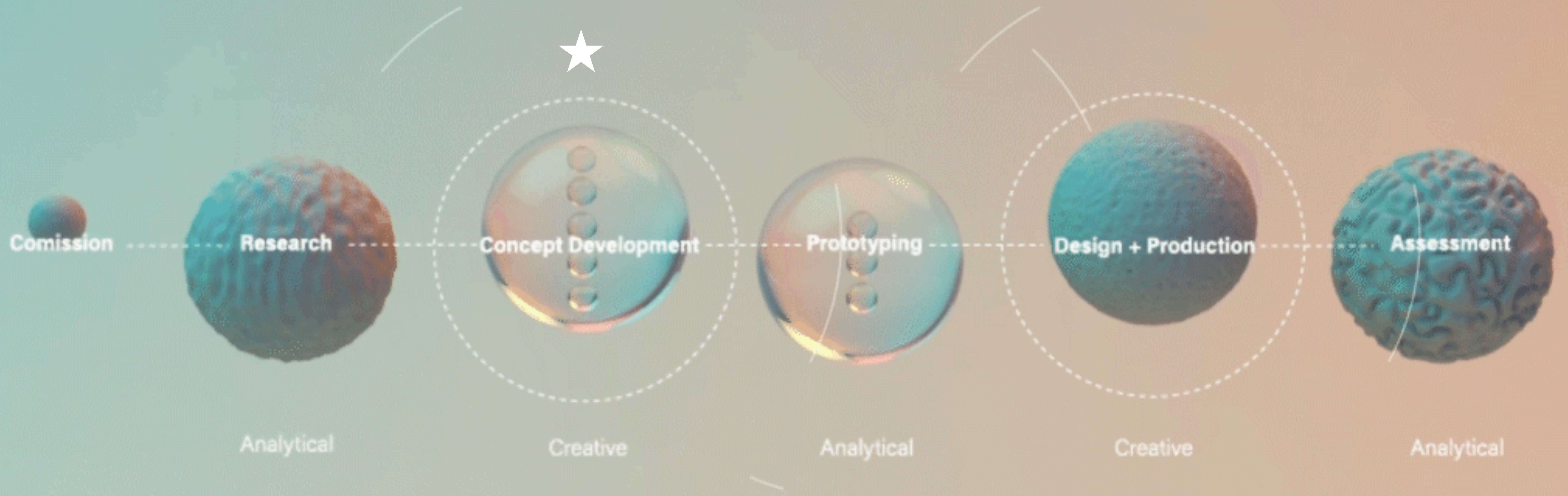
Creation of new Kamon
Using different elements related to the Japanese tea shop

Kamon

Japanese **family crest** or **emblems** are graphic **symbols associated with family clans** or establishments and usually are used to decorate different objects like flags, armor, curtains, and personal items such as clothing.

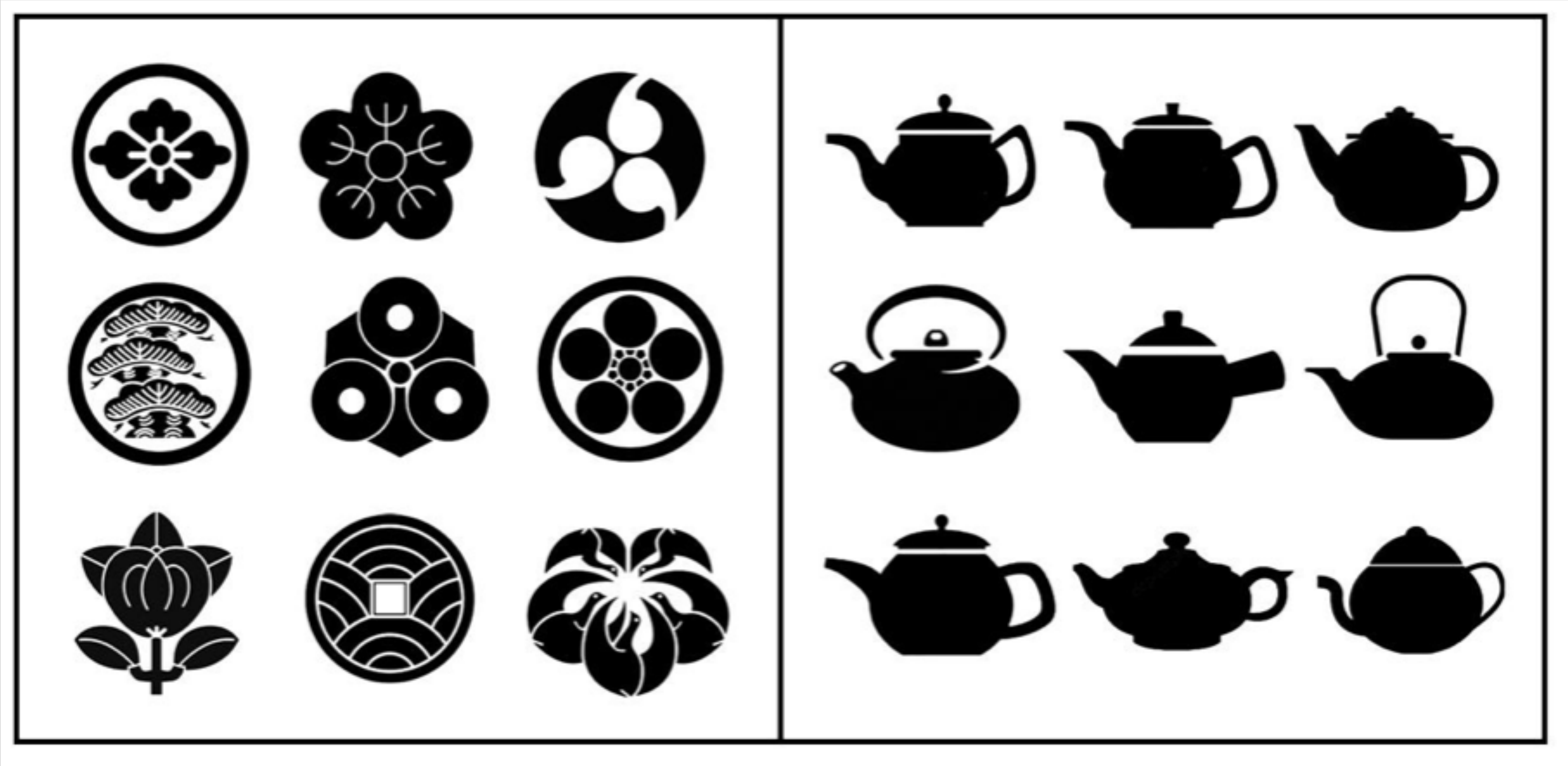
The role of these family crests has changed over time. They are now used in a **commercial approach** as company logos.





Ai as an Artificial co-worker

New Kamon



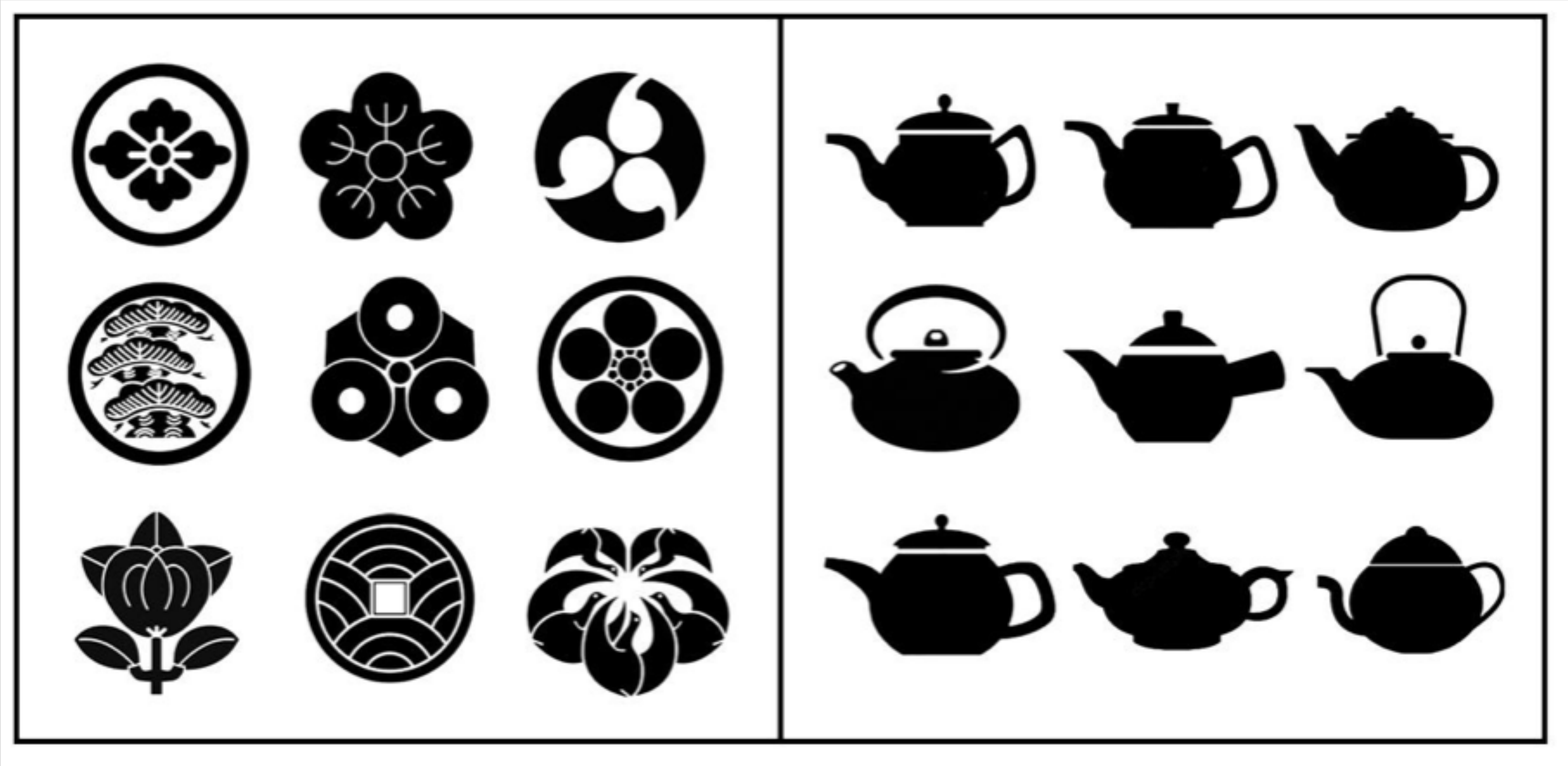
First test

Mixing two domains

Dataset was made of 200 standardized black and white graphic symbols (256 x 256 pixels) that belongs to two different domains: Japanese Kamons and Teapots.

Dataset

We can **establish parameters and patterns** in the dataset and force the ouput to follow and apply these parameters and patterns.



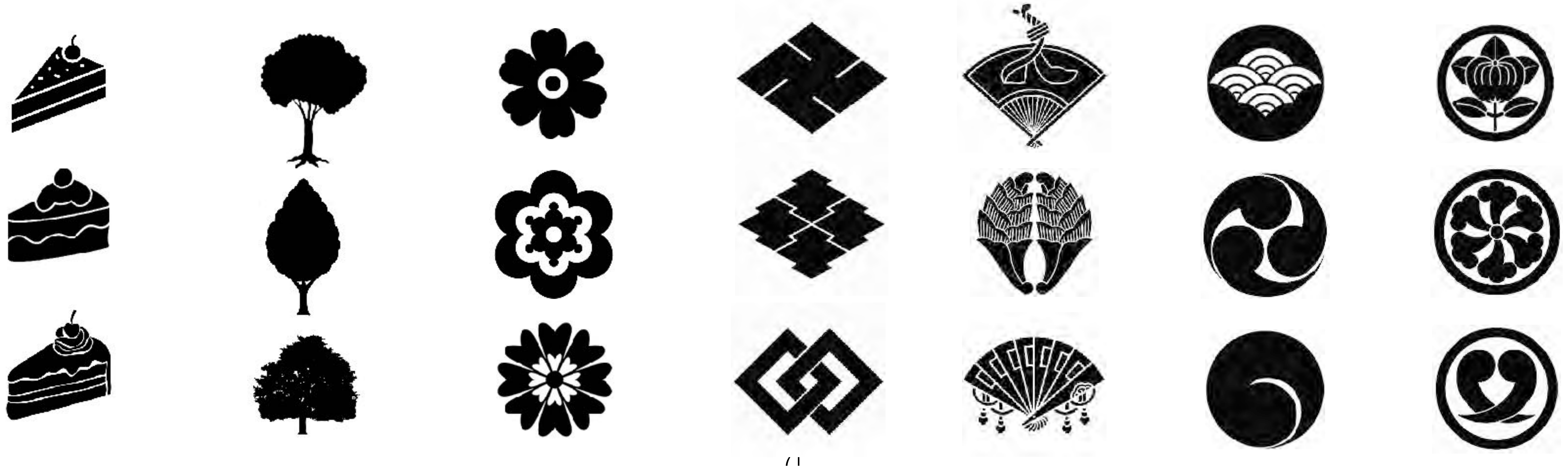
Dataset - 200 images

Results

Second test

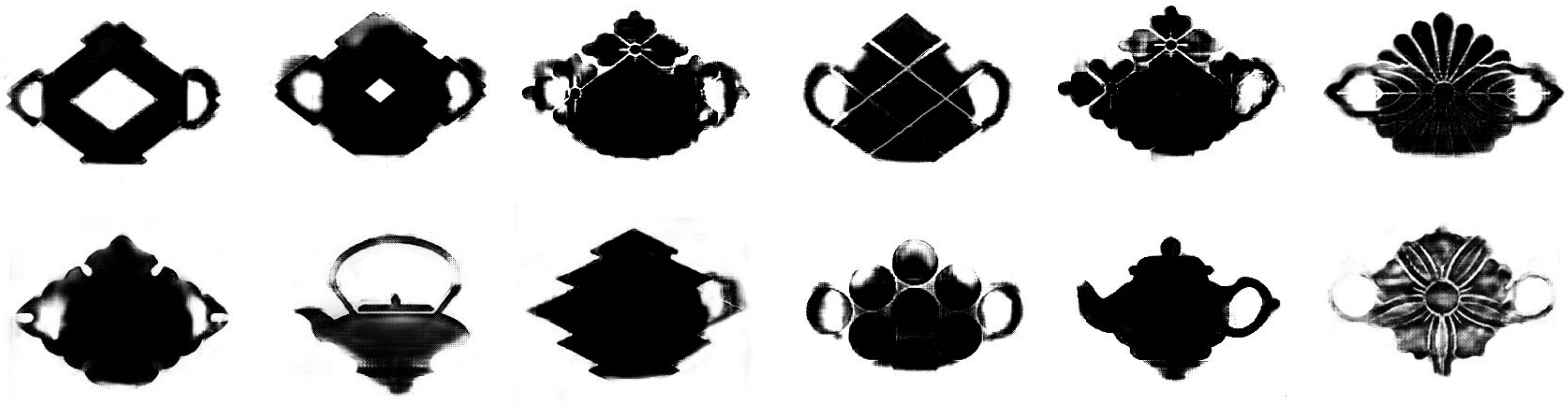
Data Set - 13 Domains (650 Images)

Cakes	Trees	Flowers	Kamon1	Kamon 2	Kamon 3	Kamon 4
50	50	50	50	50	50	50

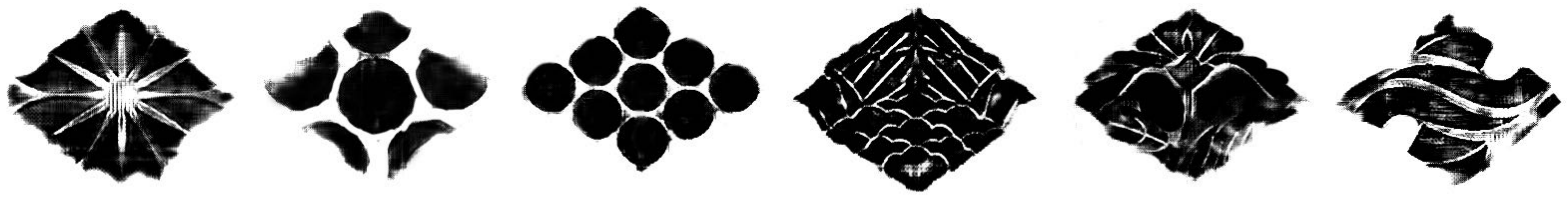


Results

Results



Results



Data set

We realized that by **combining different dimensions** in the dataset. (Different domains or categories) and <u>forcing them to output a mix</u> between this two domains you can generate new dimensions.

We can see this as the perfect approach to use GAN in a generative design approach.

Evaluation

Mr. Shoryu Hatoba / Monsho-Uwaeshi (artisan painter who paints family crests onto kimonos by hand).

What exactly constitutes a "**proper appearance** and feel of a crest."?

Hatoba:That….is extremely hard to explain.

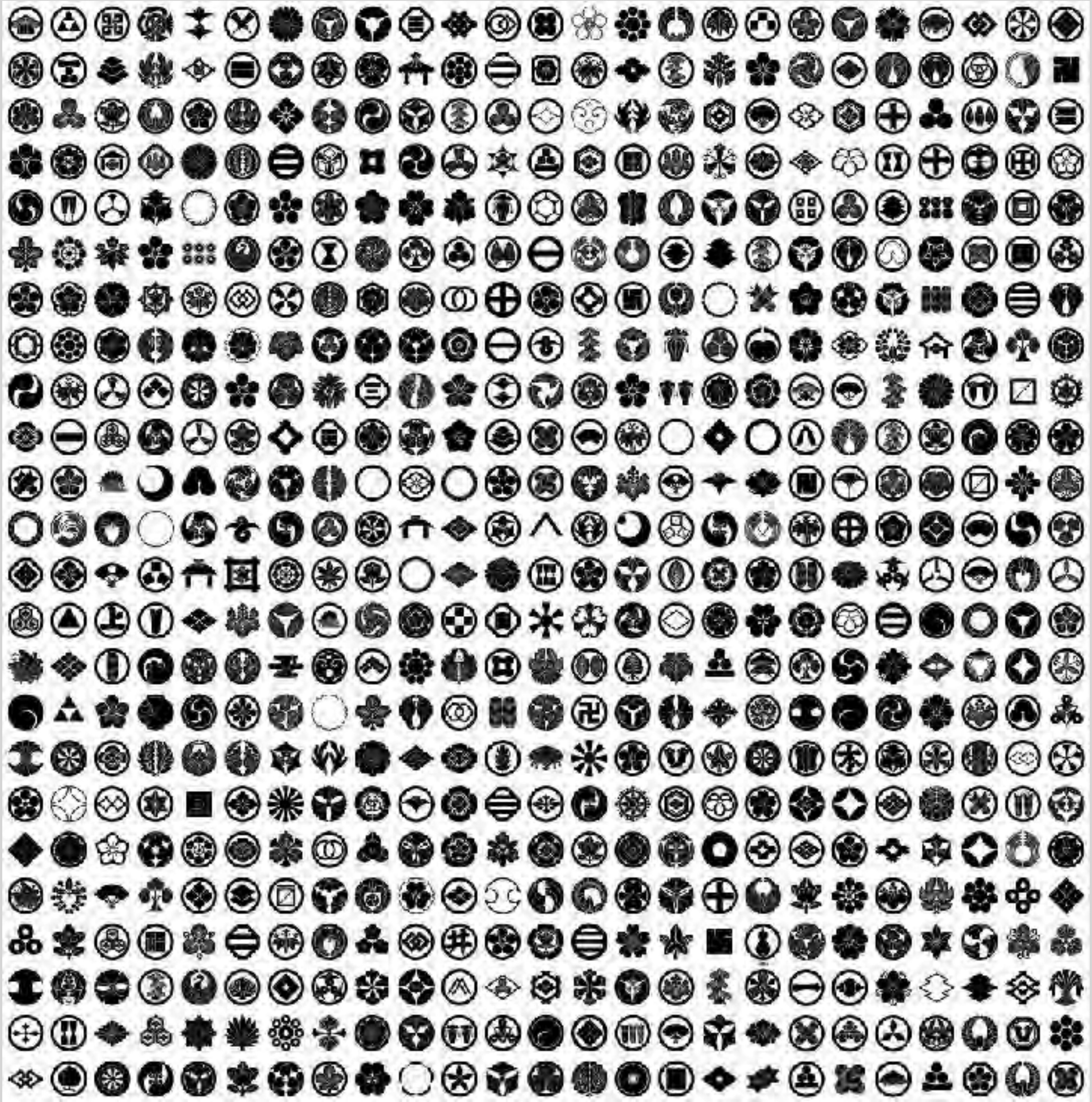
This will get into where we draw the line between what is and can't be considered as being "crest-like."

The difference **between a logo and a crest can be understood just by looking at the two**, but to explain this using words is pretty difficult.

Ask Ai vision

100 Kamon dataset samples:

30 labeled as **Emblems**
59 labeled as **Illustrations**
83 labeled as **Logos**
68 labeled as **Symbols**



Font Design Project

Goal

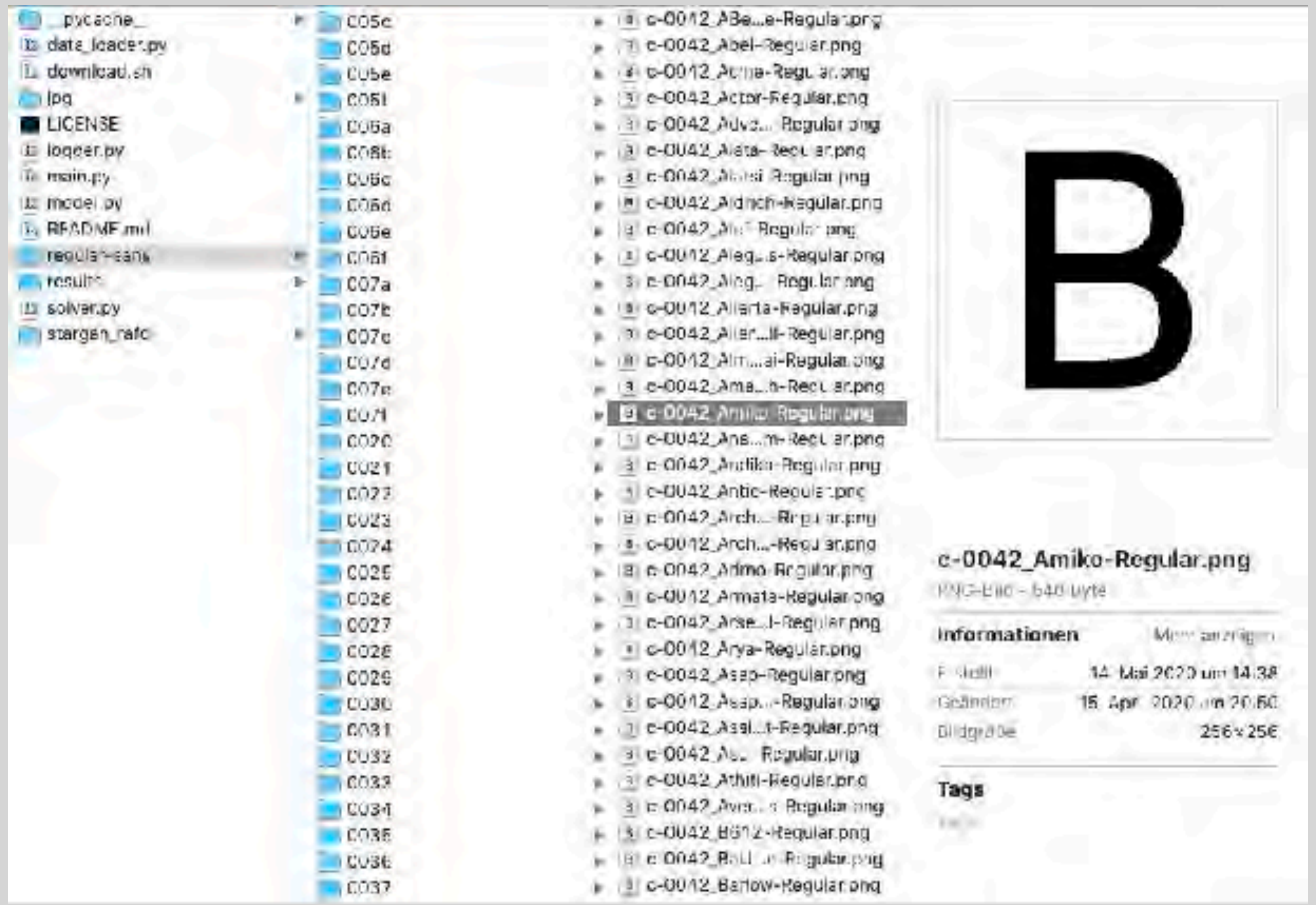
Create a Font that is legible and usable.

Objective

Develop a tool that creates a new character set for every input.

Dataset – 22,000 Items

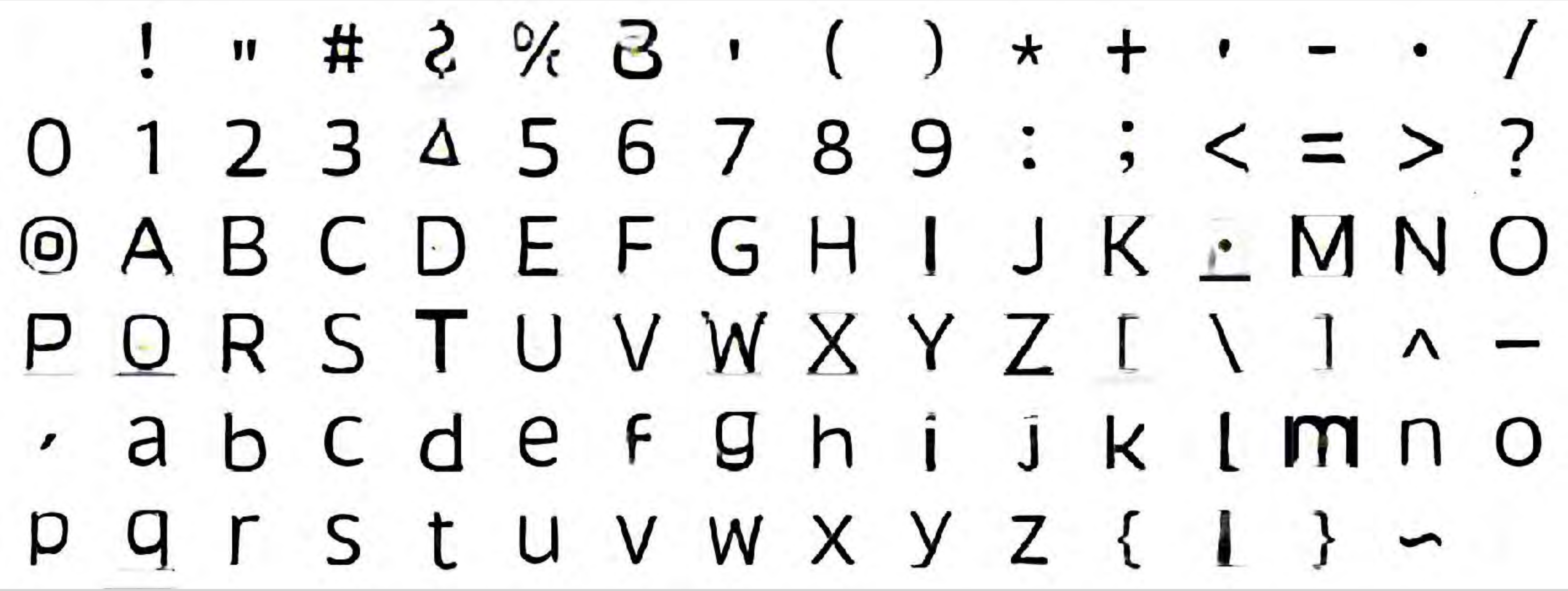
We got a bunch of Fonts (Google Master) and put them in a logical directory hierarchy.
Then we extracted all the glyphs and divided them by the UniCode.



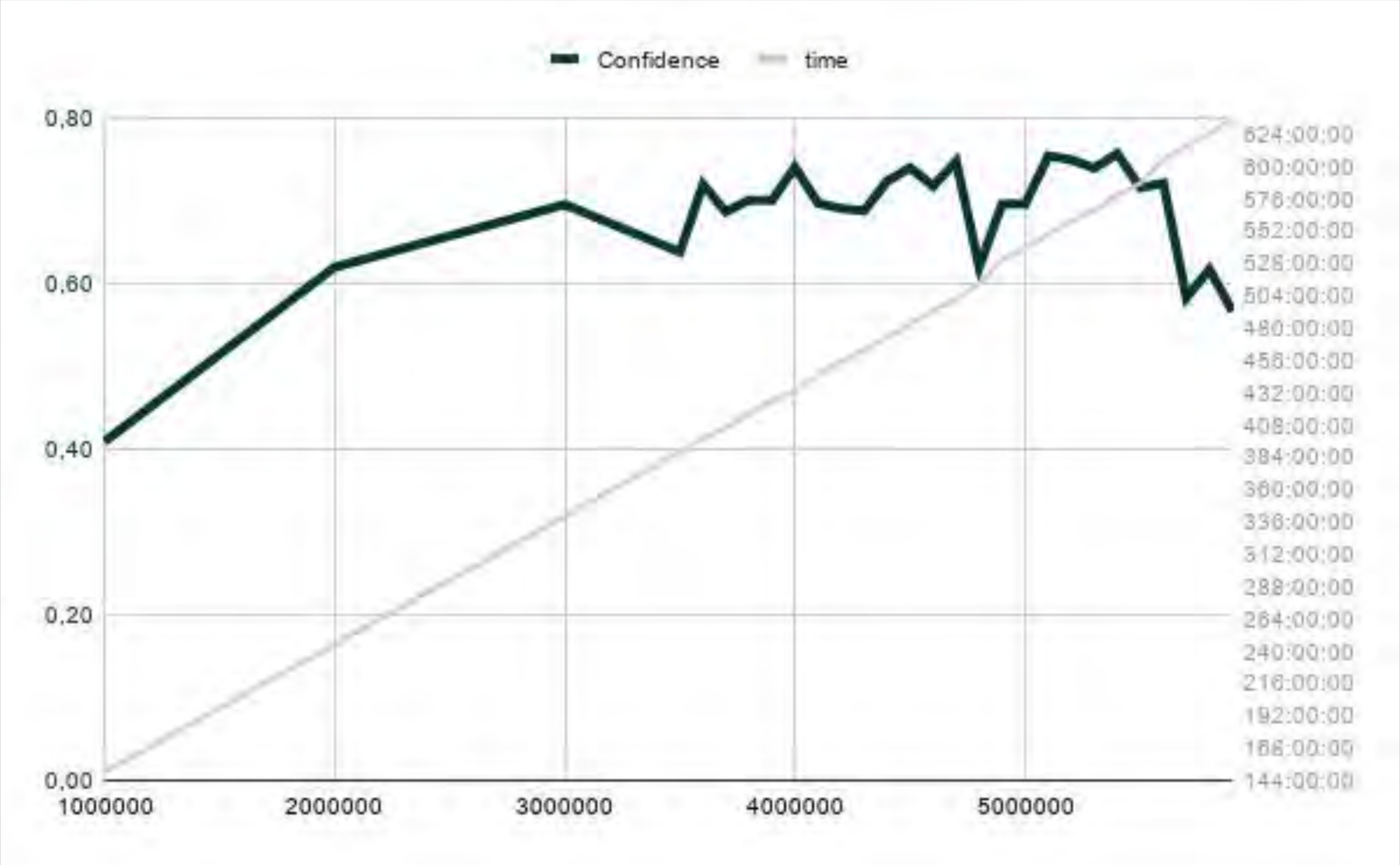
Training

Training

Evaluating



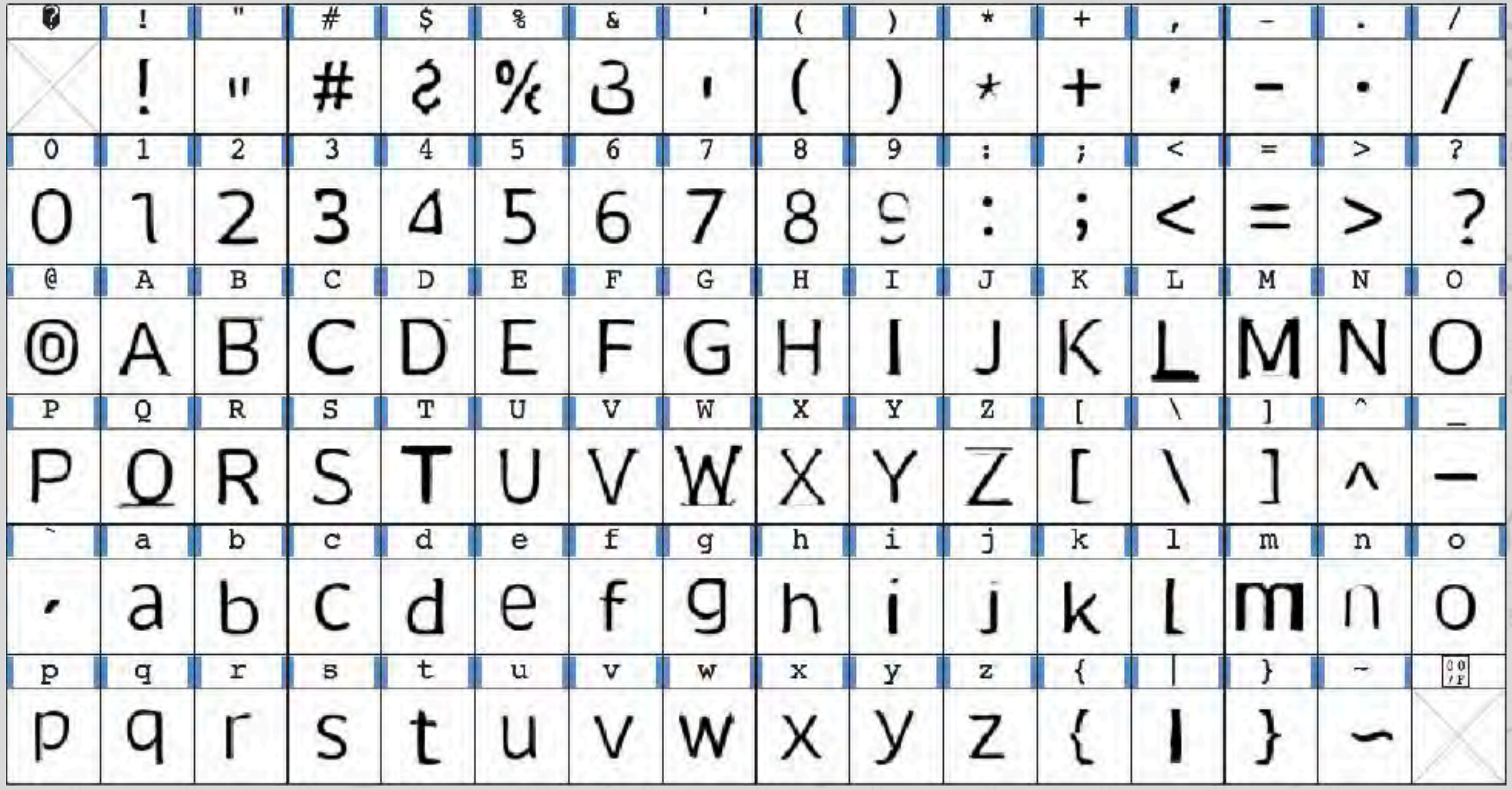
Ask Ai vision



Comparing the confidence of googles character recognition.

Google Cloud's Vision API offers powerful pre-trained machine learning models through REST and RPC APIs. Assign labels to images and quickly classify them into millions of predefined categories.

Creating the actual Font file



Possibilities

Q&A

Questions & Answers

Cape Workshop 2021

	Thursday	Friday	Saturday	Sunday	Monday	Tuesday	Wedsnesday	Thursday	Friday
Time JP	February 18	February 19	February 20	February 21	February 22	February 23	February 24	February 25	February 26
10:00-12:00			Trainning Day & Plant Tracking	Trainning Day & Plant Tracking		Students Team work	Students Team work	Students Team work	Students Team work
13:00-14:00	Introduction to the Workshop and Instructions. Students Presentation 5 to 10 Mins Presentation per School (School Introduction and team members)	Generative Design using GAN Lecture Part. 2			Showing Results of Trainning sessions and Tracking	Presentation of first Ideas	Generative Design Lecture Part.3	Preparation for Final Presentation	Preparation for Final Presentation
14:00-15:00	Presentation by Nagase Sensei & "Team formation"	Exercise using Generative Adversarial Networks			Conceptualization & preparation of first pitch presentation	Feedback Session	Design & Prototyping Team Work		
15:00-16:00	Introduction to Generative Design Lecture Part 1.				Team Work	Advise Session			Final Presentation
16:00-17:00		Closing Exercise 1							Feedback & Closure

* During the sessions marked in Yellow, Professors are Invited to Attend

Create Github Account

Next Session:

GAN Lecture 2
Exercise by teams

See you tomorrow!

Contact us!

Hisa Nimi

hisanimi@chiba-u.jp

Design Researcher
Chiba University

Juan Carlos Chacón

juancarlos@chiba-u.jp

Design Researcher
Chiba University