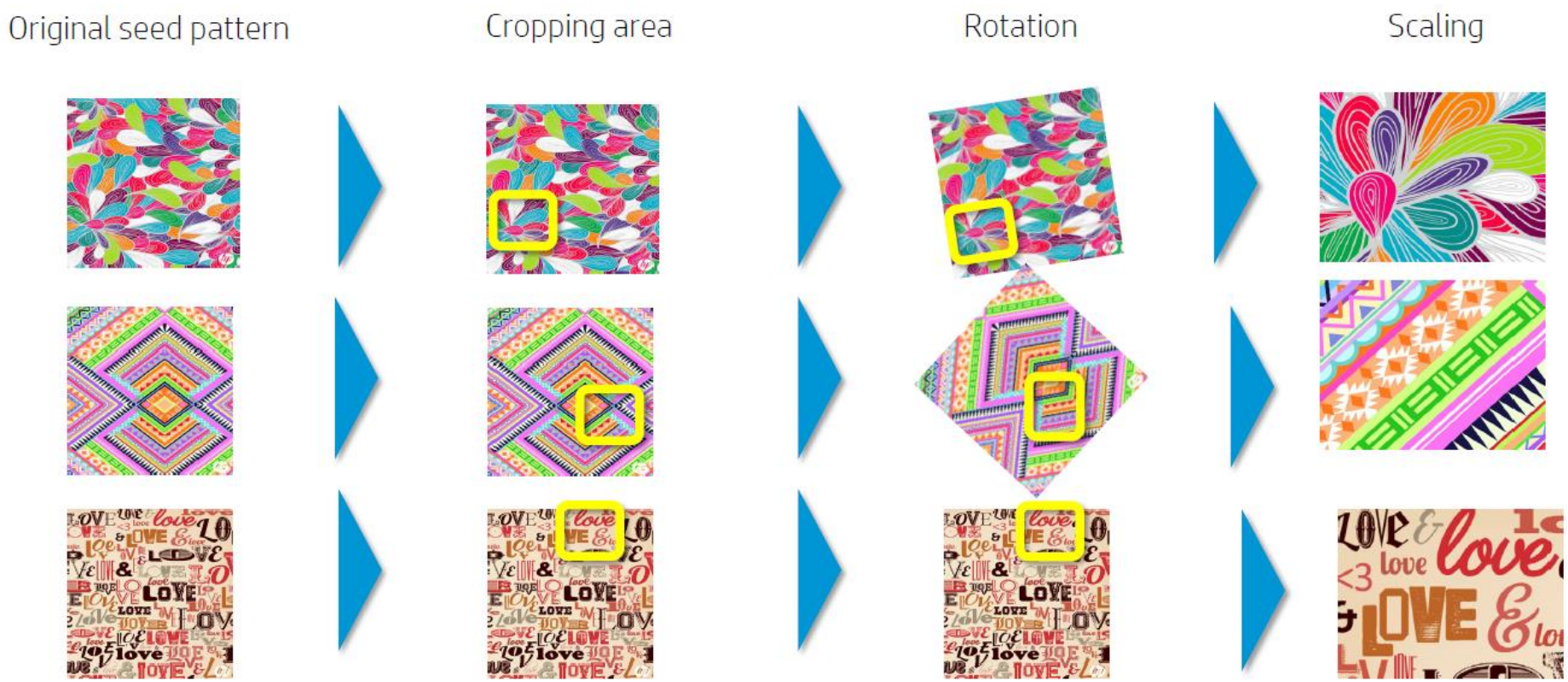
Generating HP Mosaic seeds using generative adversarial network



- HP Mosaic let you create automatic designs based on core patterns. It automatically generates a potentially unlimited number of unique graphic designs based on a fixed number of base patterns
- The heart of HP Mosaic are one or few base patterns which are vector-based graphics (SVG or PDF format). The graphics are varied by rotation, scaling and color change to always create new patterns

HP Mosaic campaigns

Problem description

Creating Mosaic seeds is a hard task. The design requires to be **complex as possible in vector format and fits the specific campaign/brand**.

- **Time consuming**
- **Weak design outcome**
- **Uniqueness**

GOAL

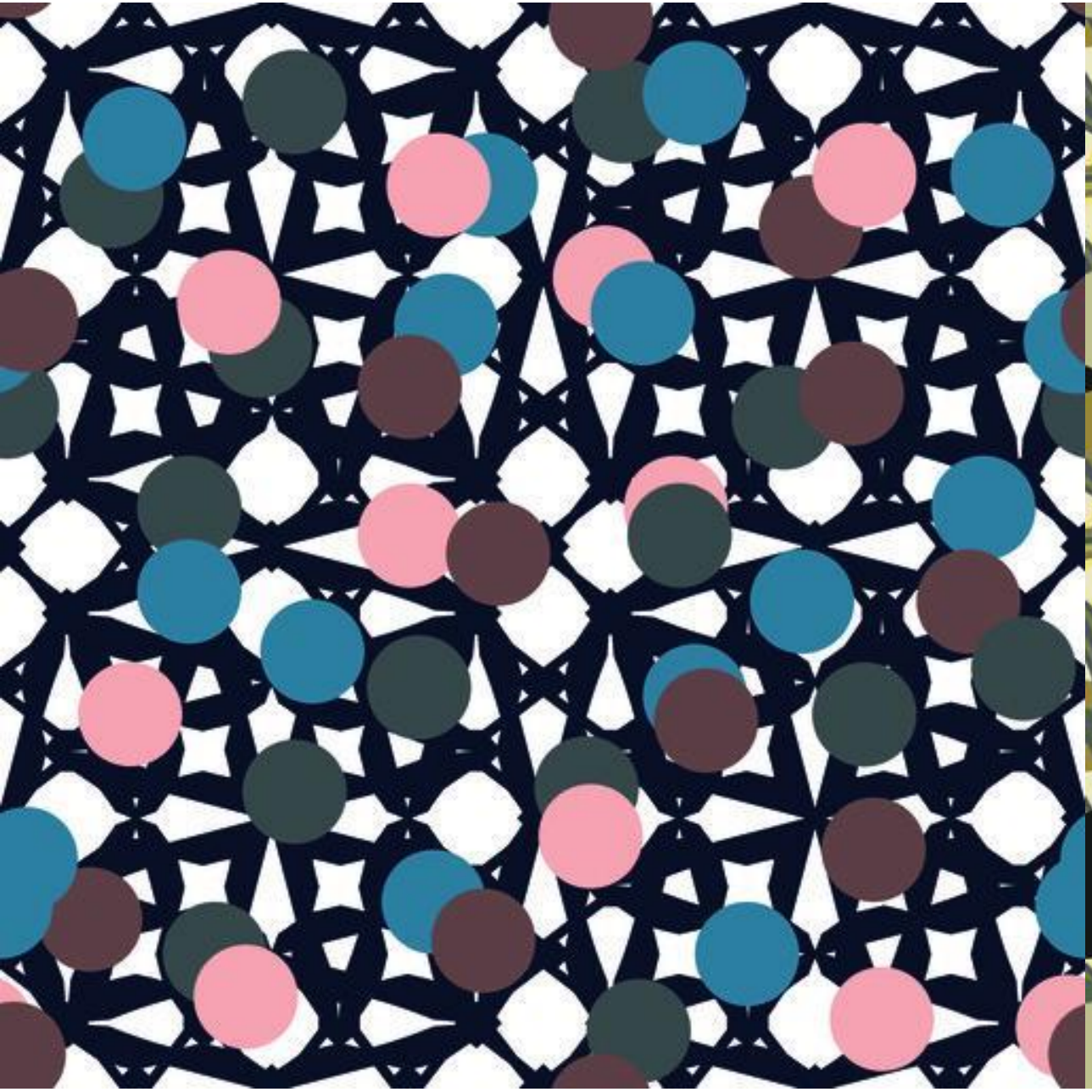
Help the designers to create seeds for Mosaic algorithm

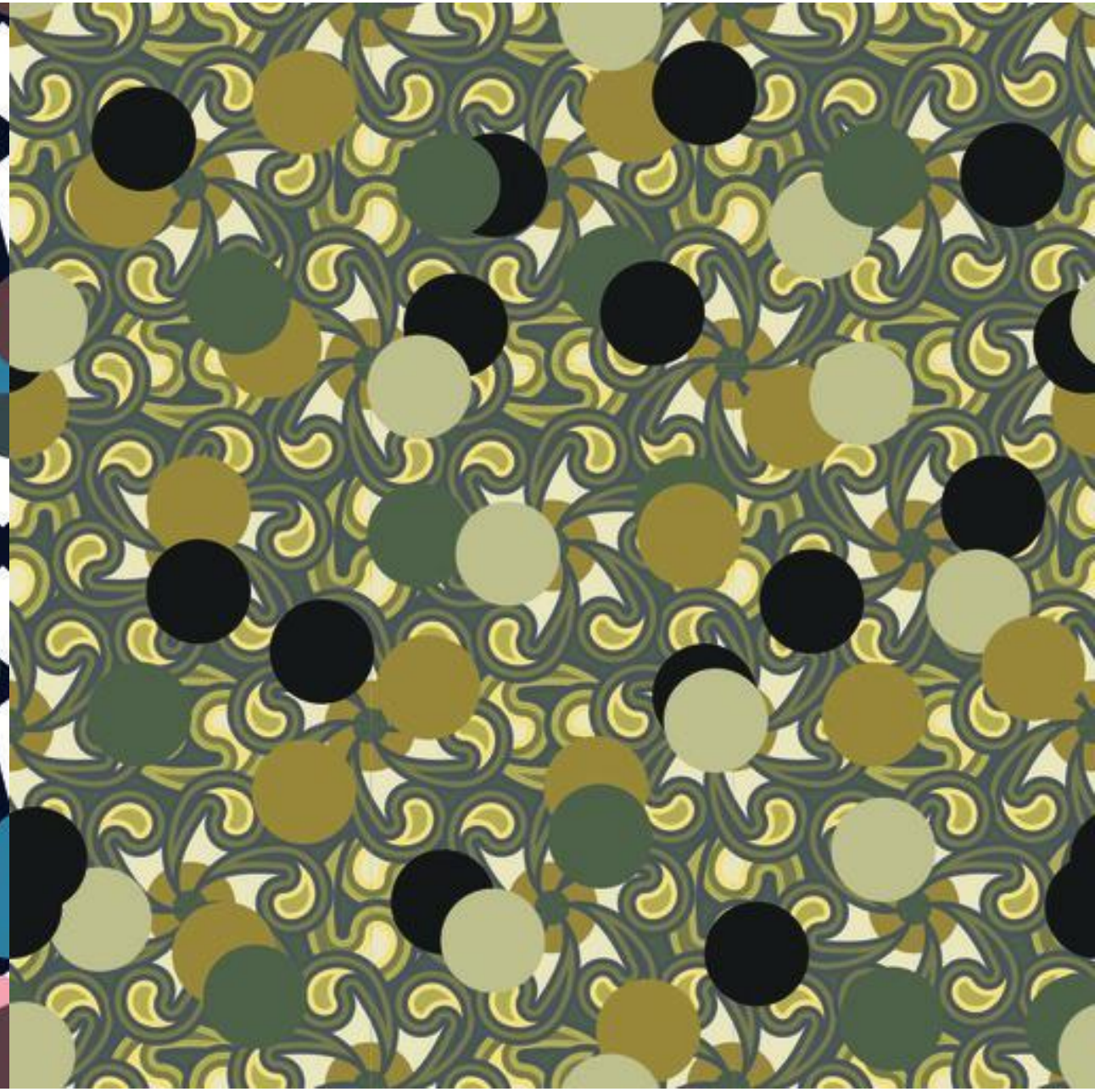
Proposed solution

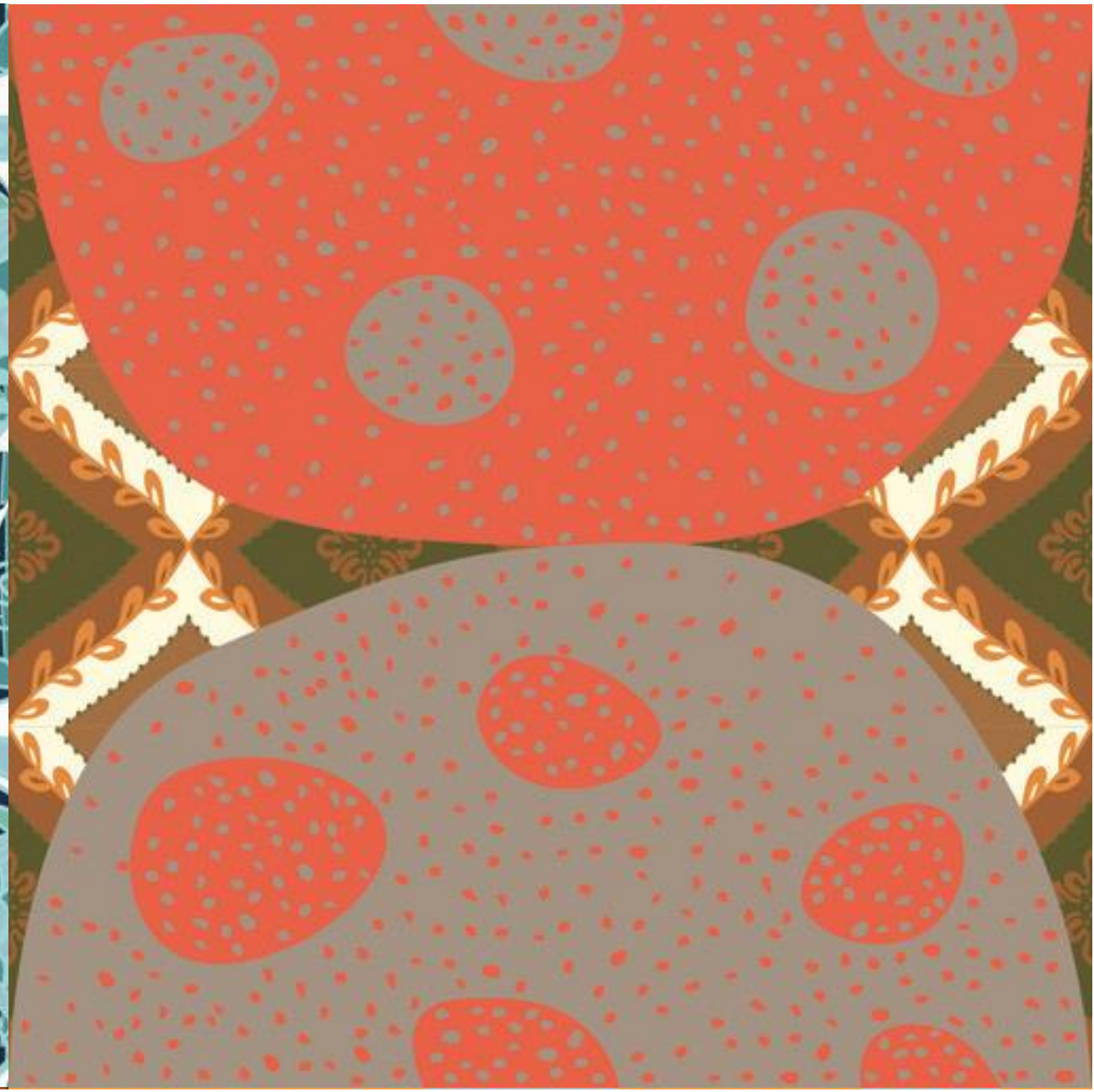
Using Generative adversarial networks (GAN) for creating Mosaic seeds based on a dataset of existing Mosaic seeds

The dataset

- For training the GAN properly and getting a variety of unique designs, we need to use large dataset as possible
- I’ve managed to collect just 400 mosaic base pattens (seeds) in vector format that fits the HP Mosaic algorithm requirements, which is a very small dataset for training a GAN and achieving suitable results
- In order to increase the dataset size I used data augmentation
- The dataset was augmented with color shuffling and combinations of the vector images
- The 400 dataset images augmented into 4000 images





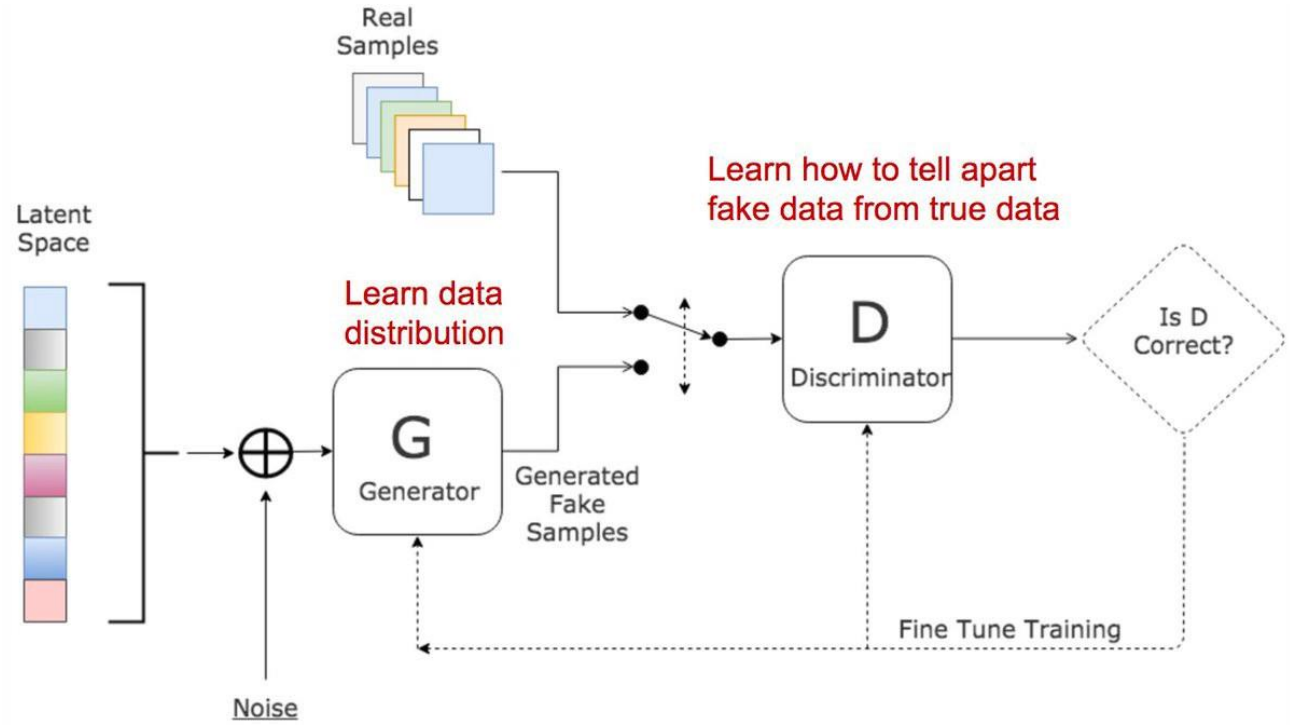


Training Images

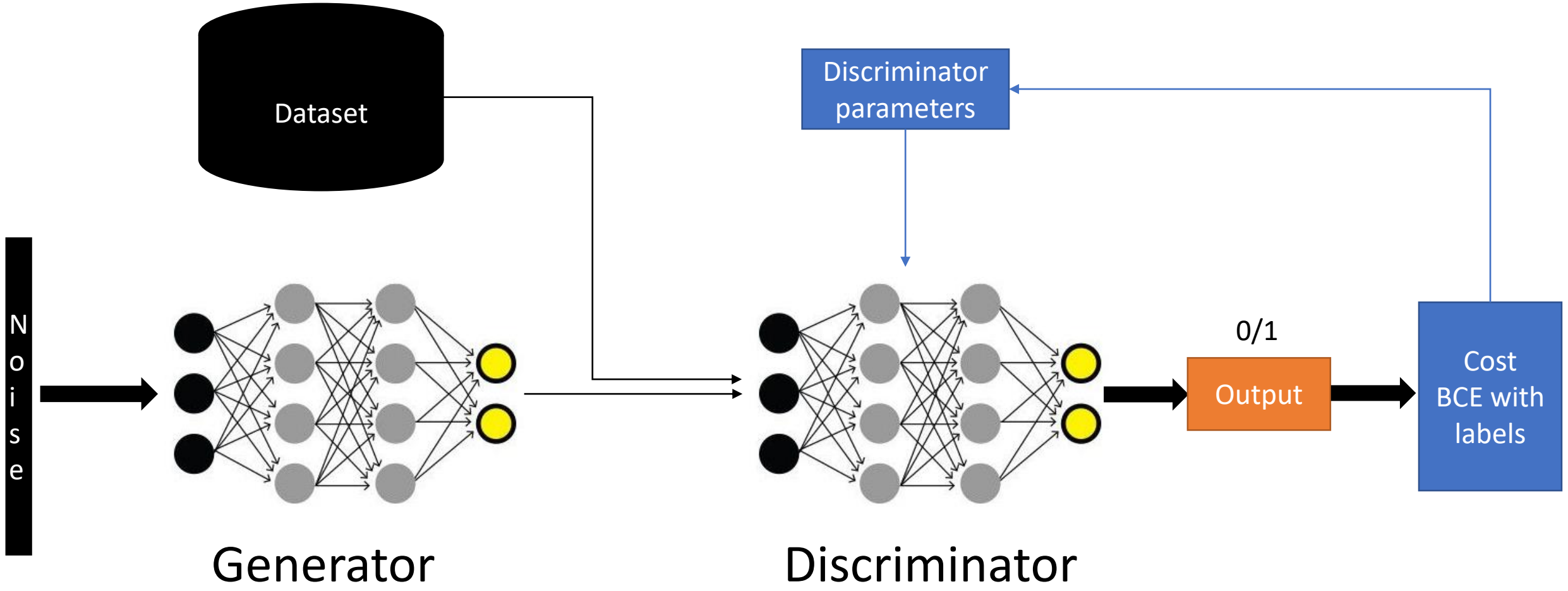
GAN Architecture

A generative adversarial network (GAN) has two parts:

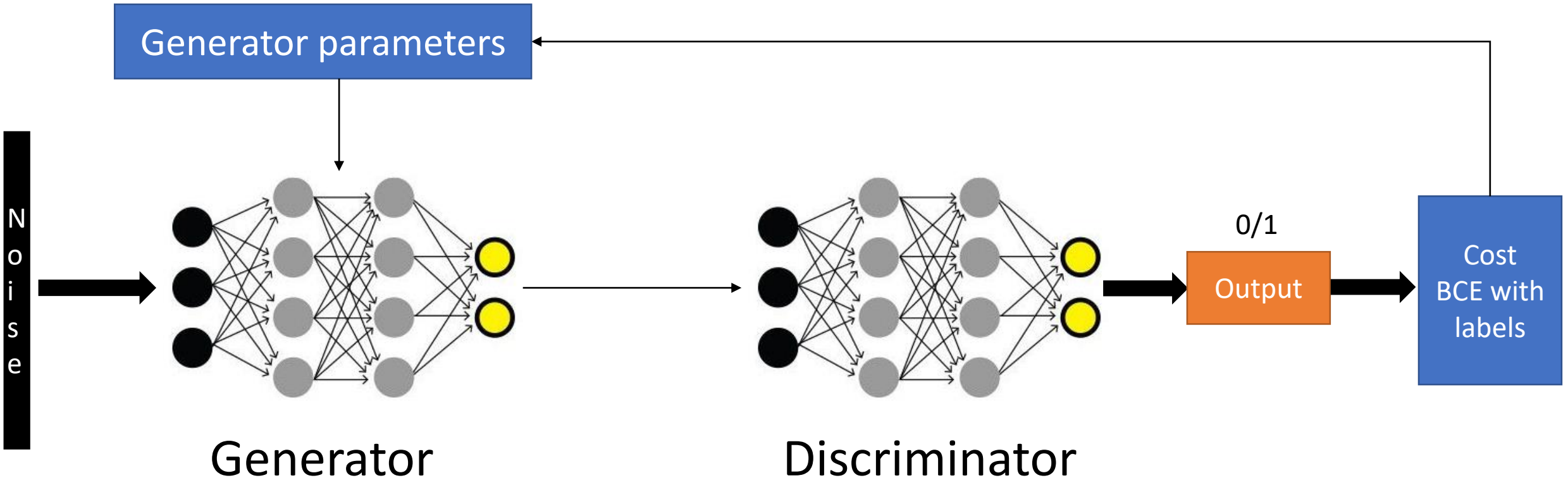
- The **generator** learns to generate plausible data. The generated instances become negative training examples for the discriminator.
- The **discriminator** learns to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results.



Discriminator weights update



Generator weights update



Loss function – Binary cross entropy

$$BCE = -\frac{1}{N}\sum_{i=0}^{N} y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i)$$

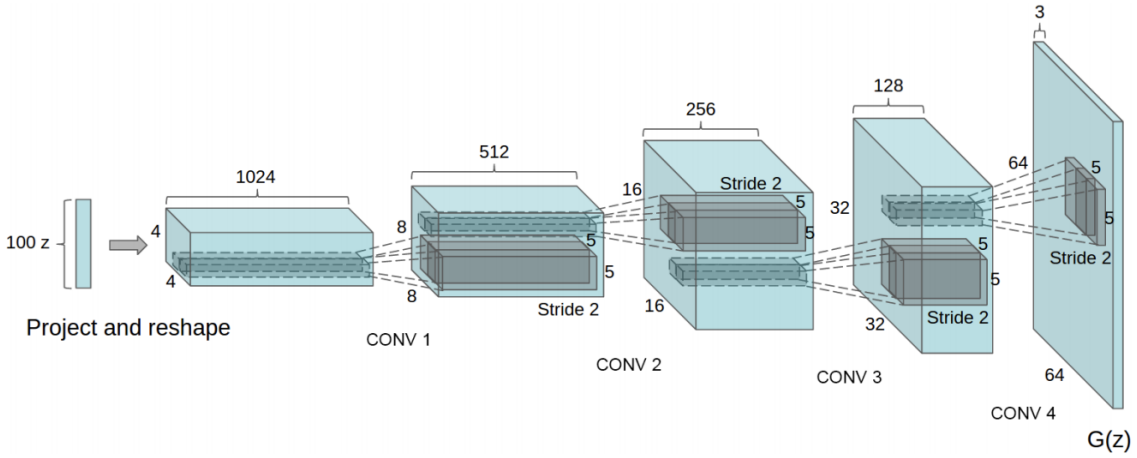
Y_i	$\hat{Y}_i$	$Y_i log(\hat{Y}_i)$
0	any	0
1	0.99	~0
1	~0	~-inf

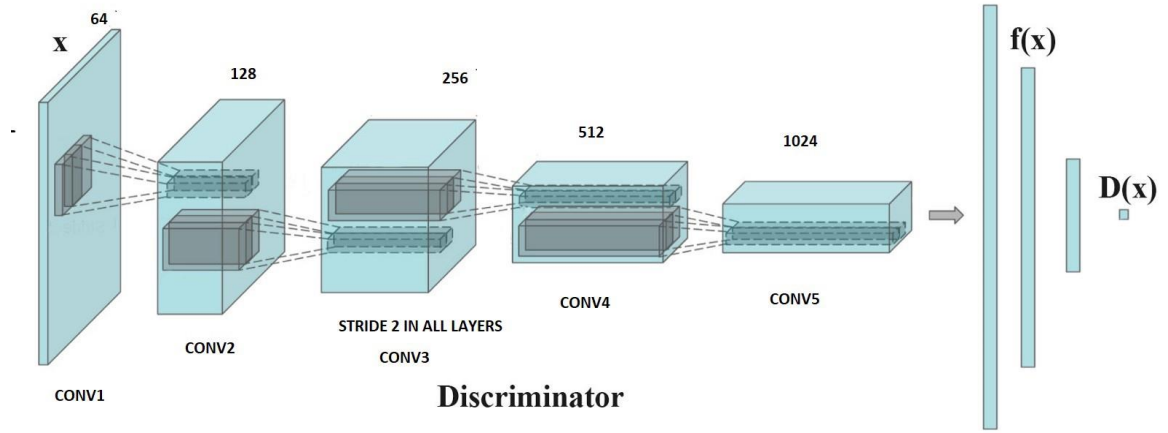
Y_i	$\hat{Y}_i$	$(1 - Y_i) log(1 - \hat{Y}_i)$
1	any	0
0	0.01	~0
0	~1	~-inf

GAN common problems

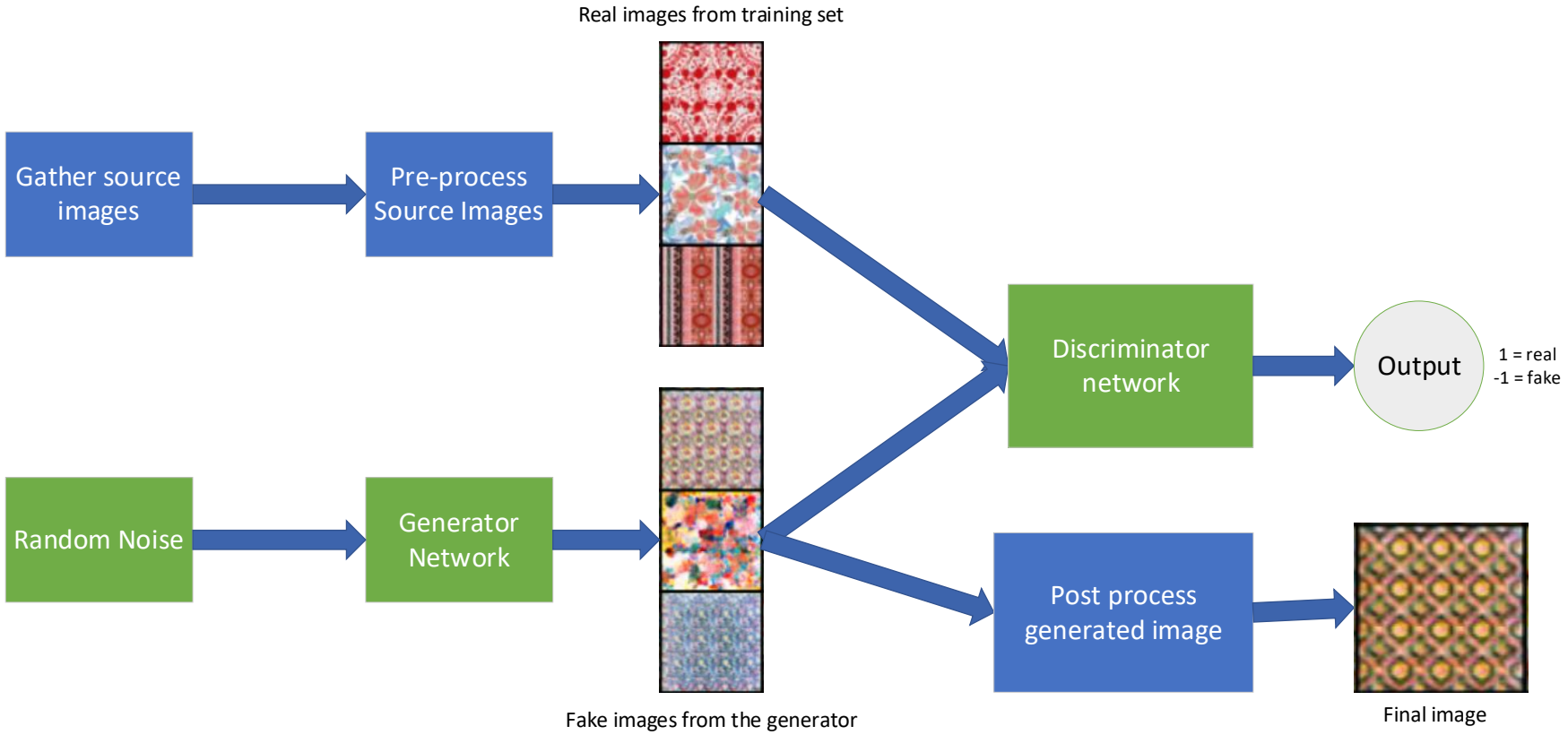
- **Vanishing Gradients**
- **Mode Collapse**
- **Failure to Converge**

DCGAN


100 z
Project and reshape
1024
4
4
512
8
8
Stride 2
5
5
CONV 1
256
16
16
Stride 2
5
5
CONV 2
128
32
32
Stride 2
5
5
CONV 3
3
64
64
Stride 2
5
5
CONV 4
G(z)

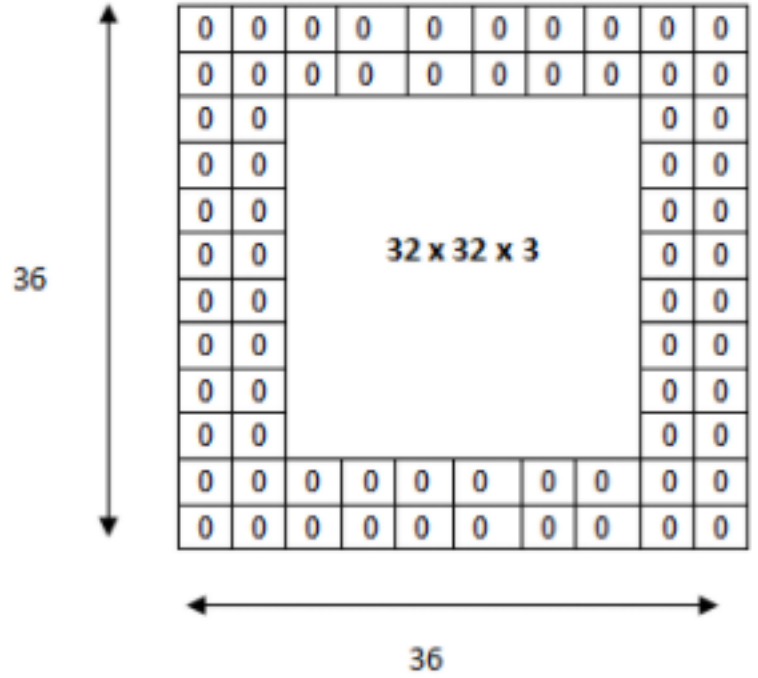

x
64
CONV1
128
CONV2
256
STRIDE 2 IN ALL LAYERS
CONV3
512
CONV4
1024
CONV5
Discriminator
f(x)
D(x)

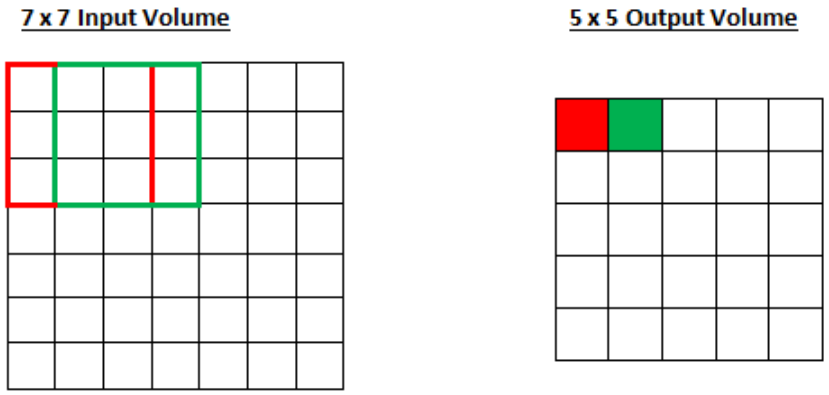
Training process



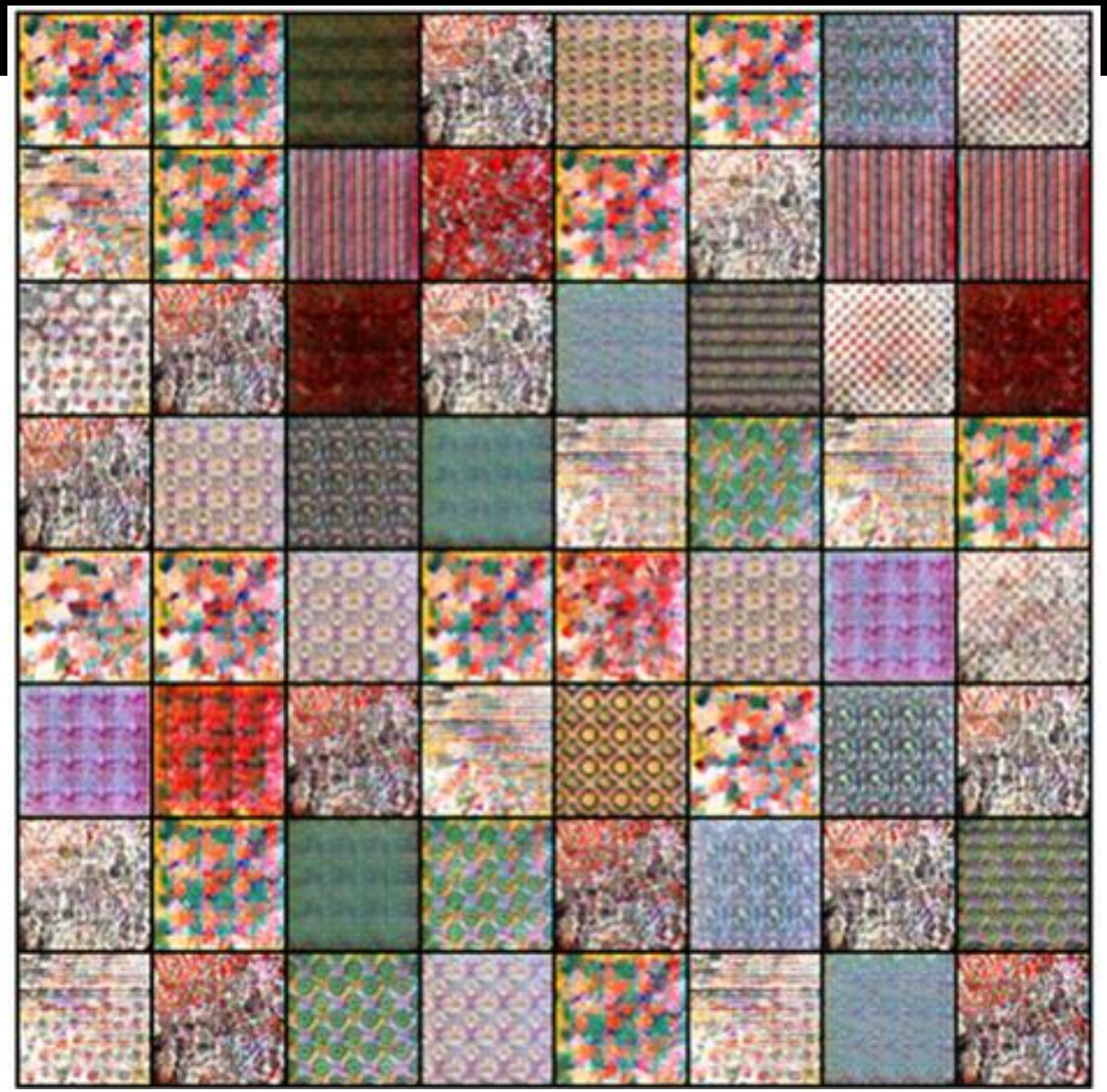
Experiments with DCGAN

- Kernel size
- Padding
- Stride
- Number of features
- Adding and changing layers

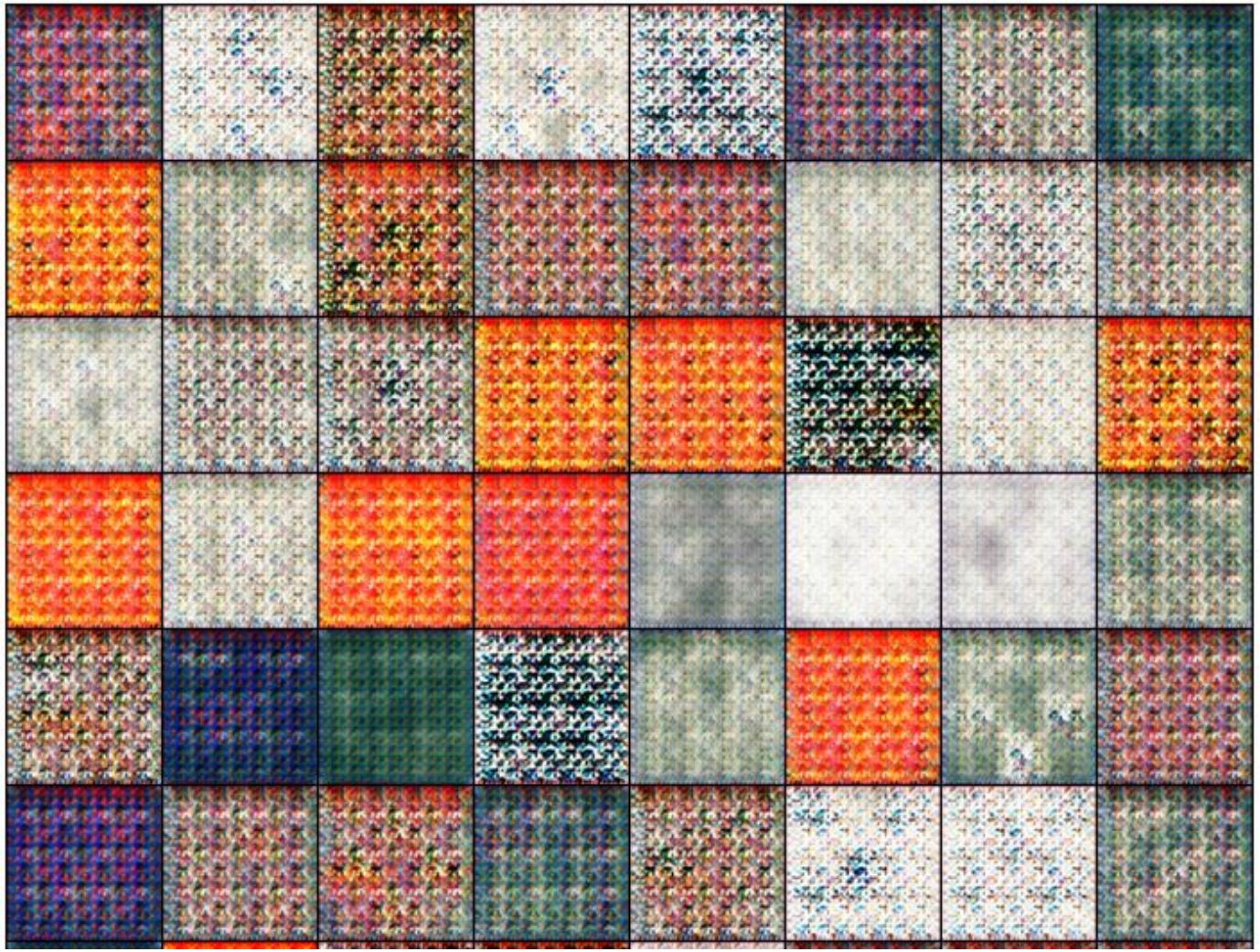


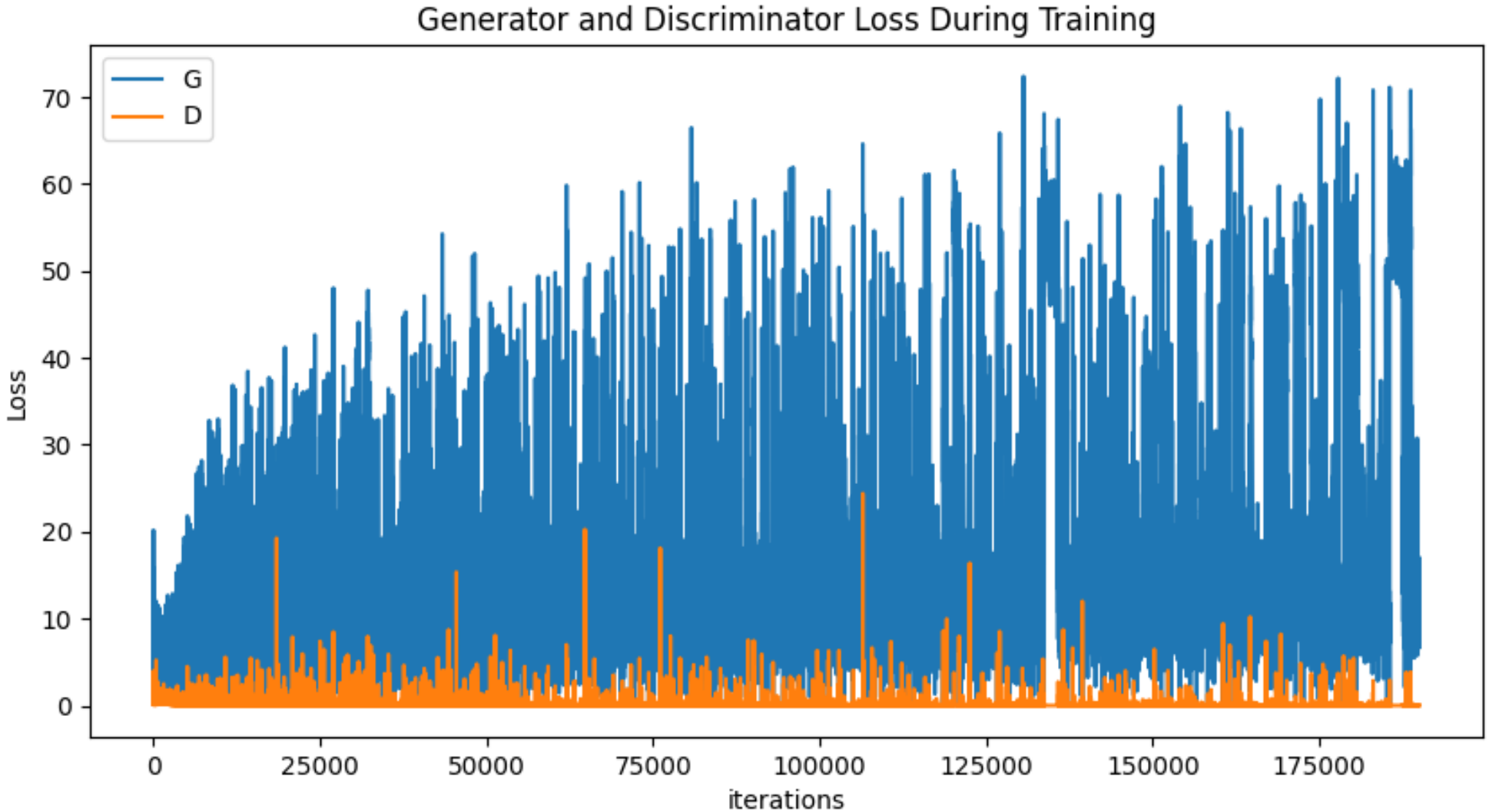


DCGAN Results

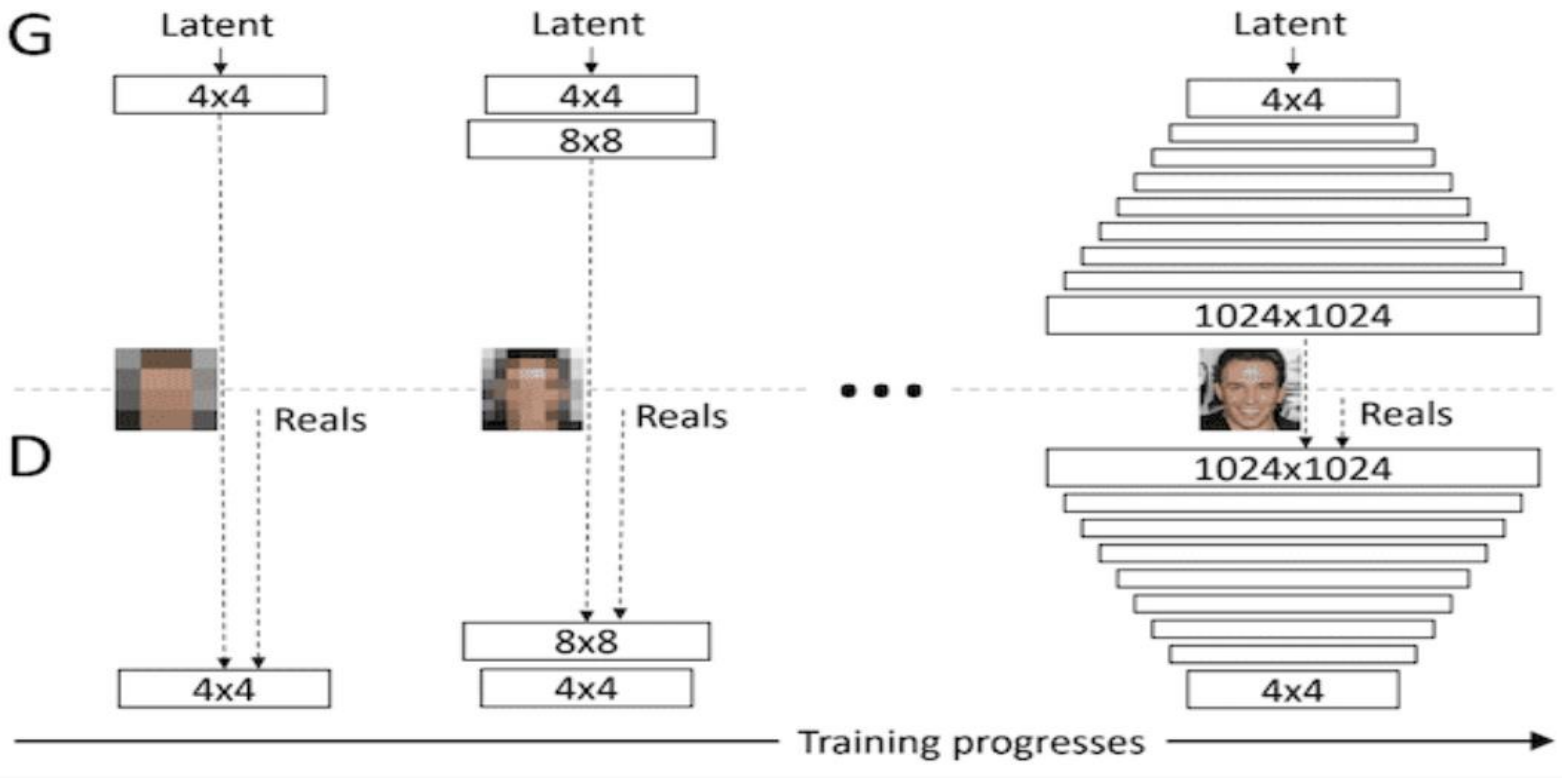


Attempt to generate larger images using more layers



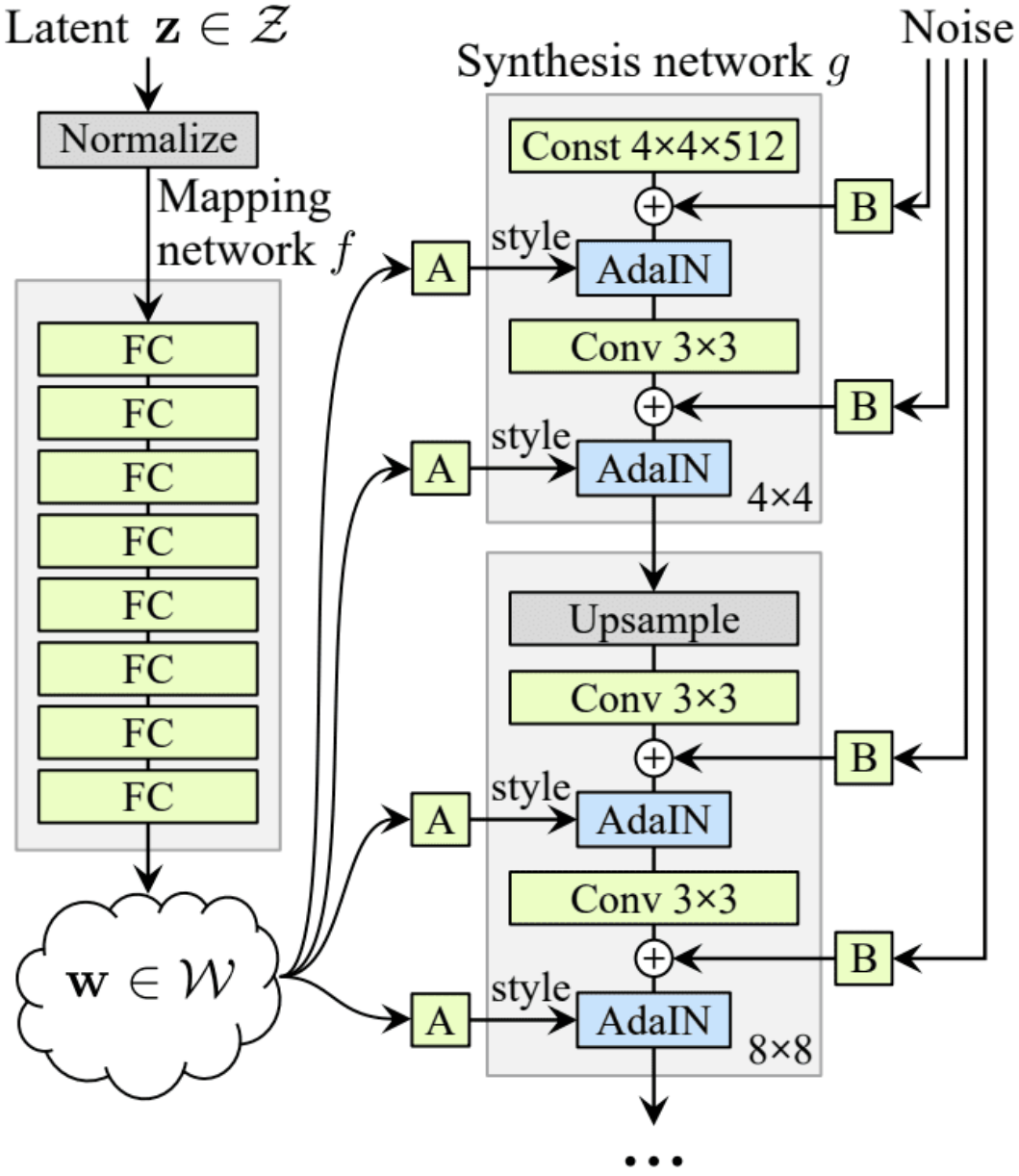

Generator and Discriminator Loss During Training
G
D
Loss
0
10
20
30
40
50
60
70
0
25000
50000
75000
100000
125000
150000
175000
iterations

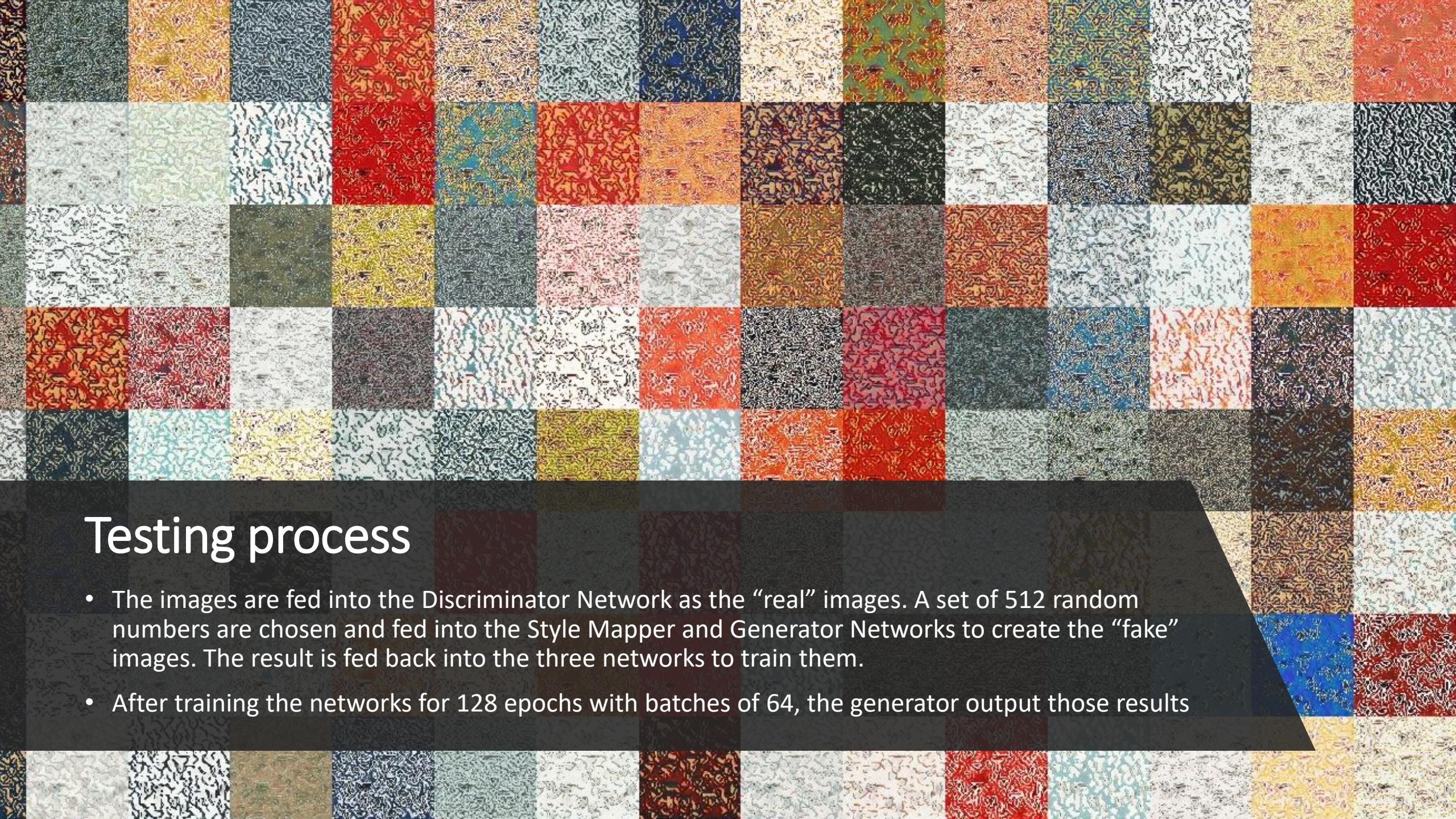
PrograssiveGAN



StyleGan

- Baseline Progressive GAN.
- bilinear upsampling.
- mapping network and adaptive instance normalization
- Removal of latent vector input to generator.
- noise to each block.





Testing process

- The images are fed into the Discriminator Network as the “real” images. A set of 512 random numbers are chosen and fed into the Style Mapper and Generator Networks to create the “fake” images. The result is fed back into the three networks to train them.
- After training the networks for 128 epochs with batches of 64, the generator output those results

StyleGAN with Adaptive Discriminator Augmentation

One of the major improvements in StyleGAN ADA is dynamically changing the amount of image augmentation during training.

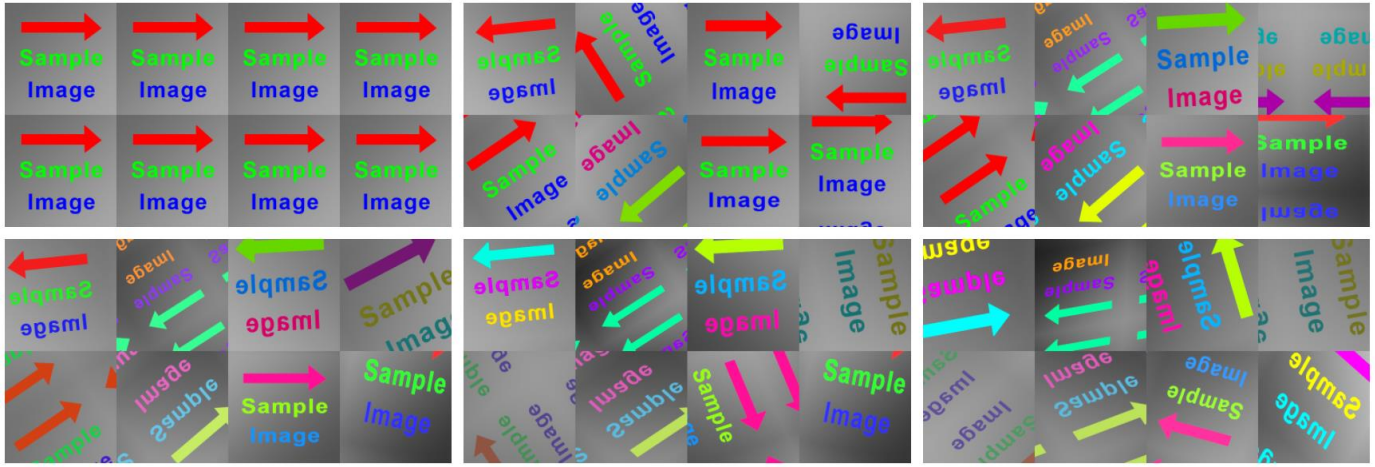
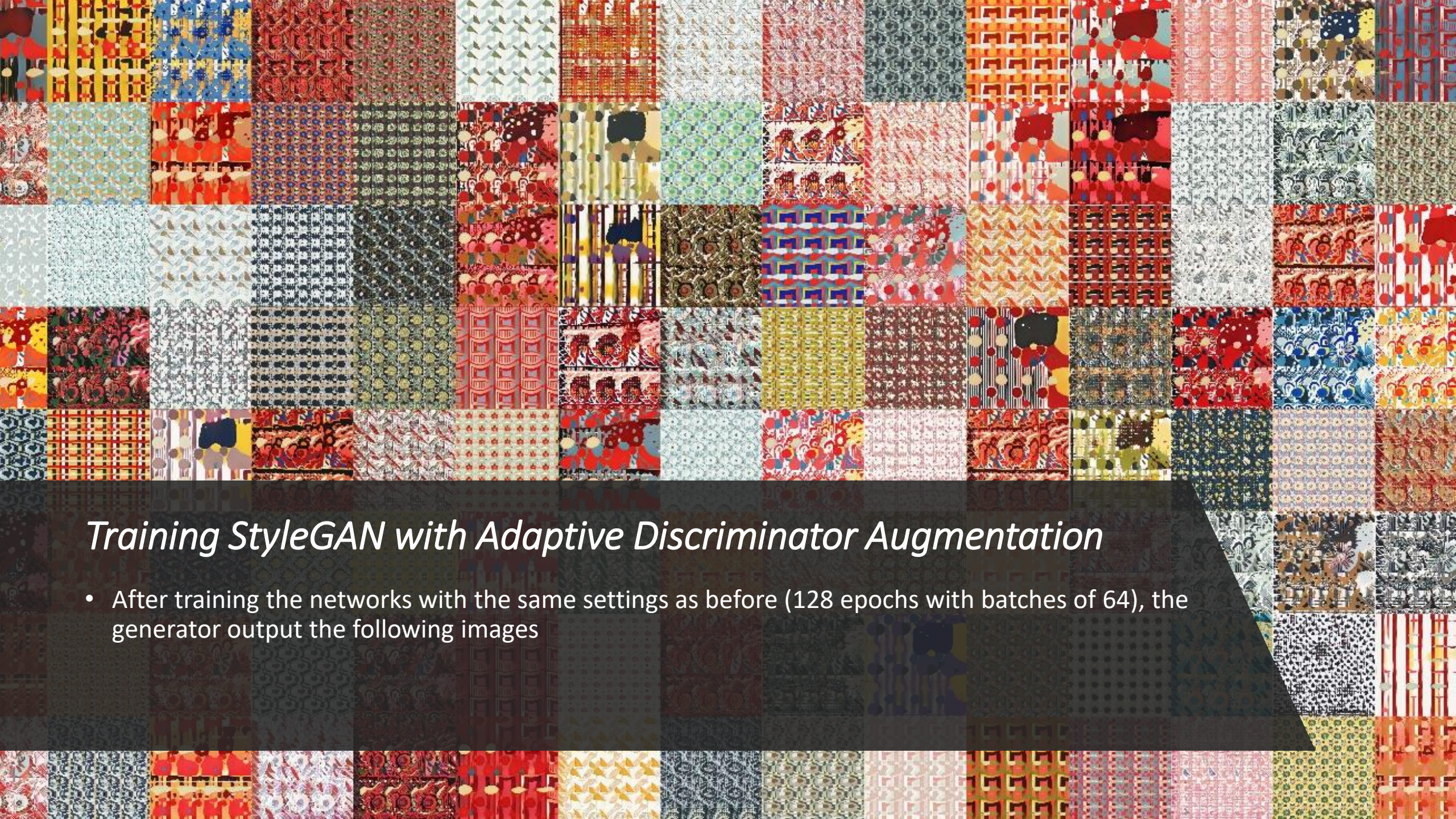
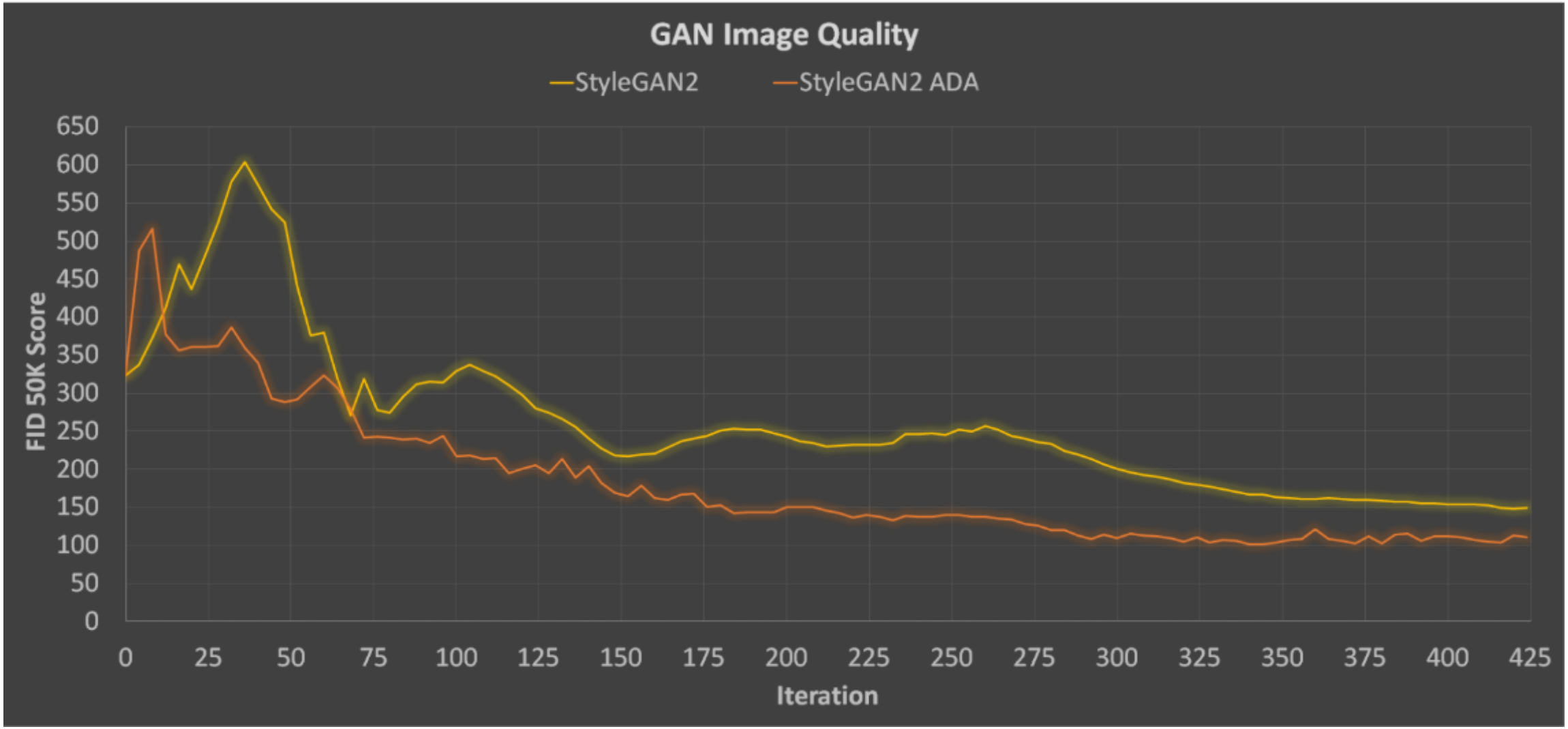


Image Augmentation with (top) p=0.0, 0.2, 0.4 (bottom) p=0.6, 0.8, 1.0, Images by Author



Training StyleGAN with Adaptive Discriminator Augmentation

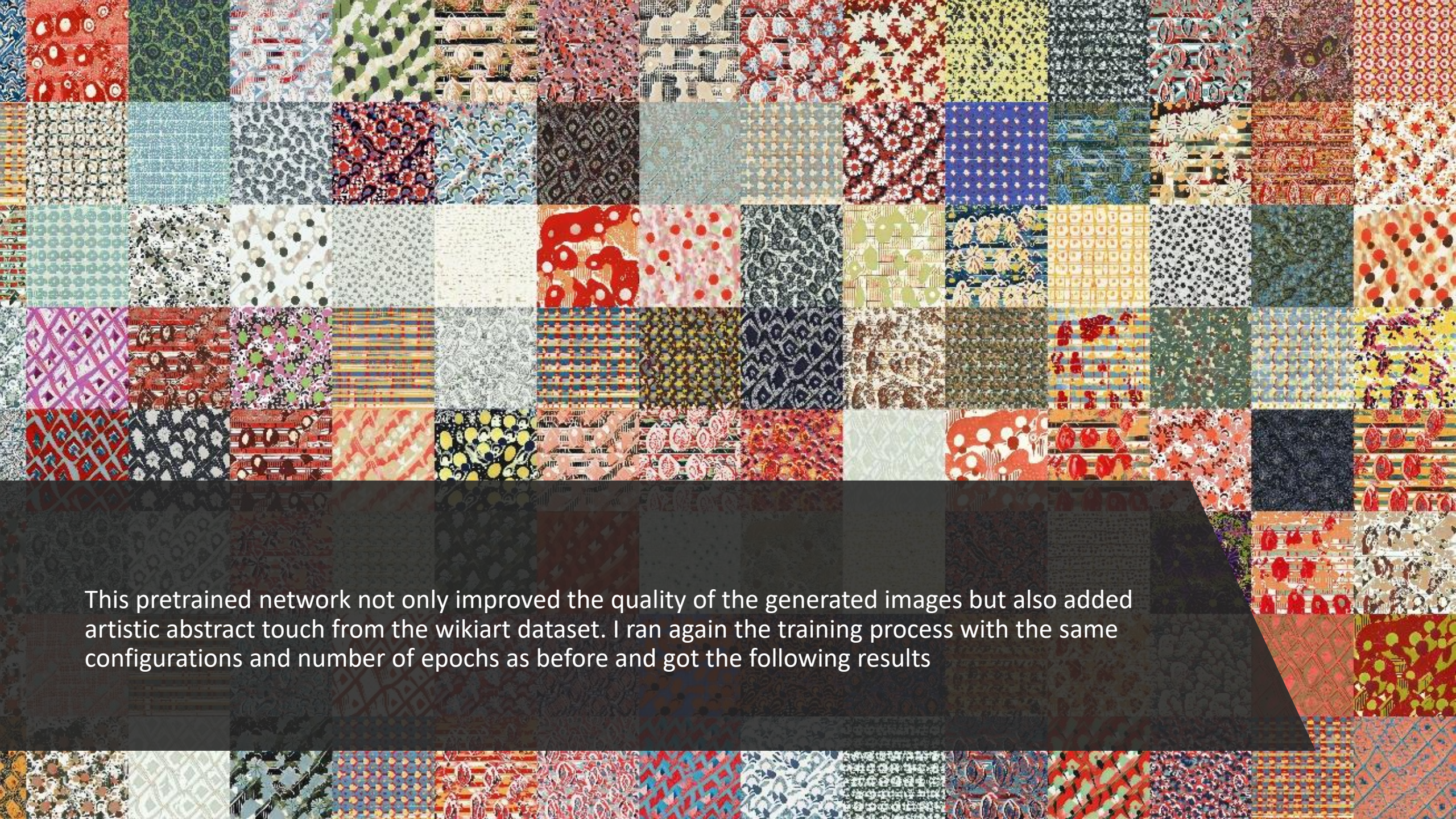
- After training the networks with the same settings as before (128 epochs with batches of 64), the generator output the following images


GAN Image Quality
StyleGAN2
StyleGAN2 ADA
FID 50K Score
650
600
550
500
450
400
350
300
250
200
150
100
50
0
0
25
50
75
100
125
150
175
200
225
250
275
300
325
350
375
400
425
Iteration

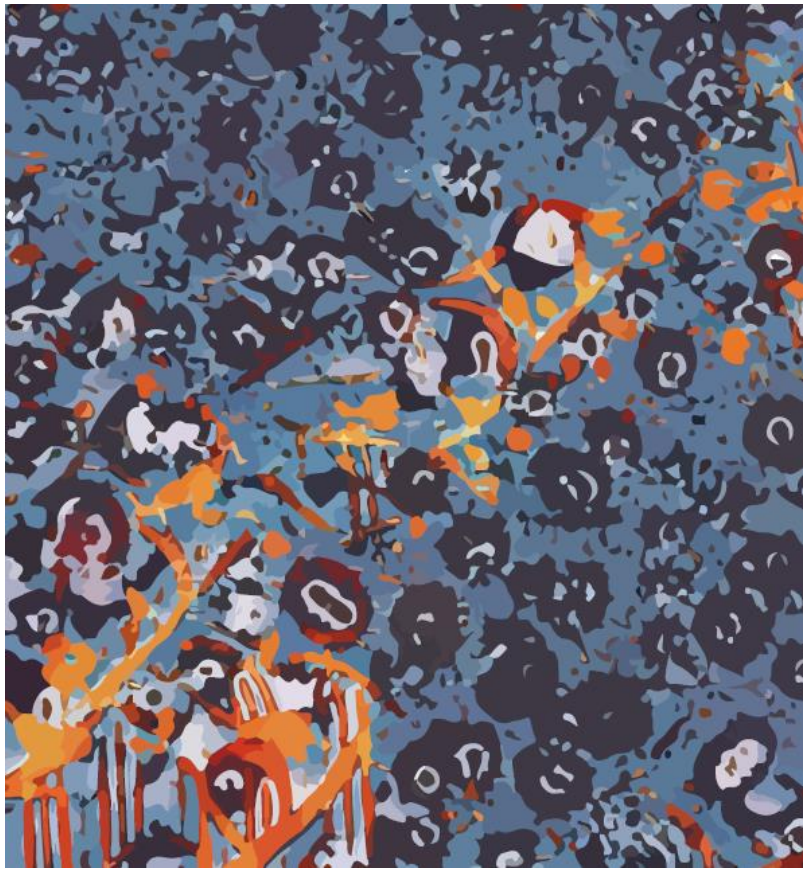
Transfer Learning

- It’s possible to improve the quality of the generated images by first training the GAN on a different, larger set of images, and then further train the model using the smaller dataset. This technique is called Transfer Learning. It was first described as a technique for training NNs in 1976.
- *“Transfer Learning is technique that uses a pre-trained neural network trained for Task 1 for achieving shorter training time in learning Task 2. — Stevo Bozinovski”*
- I found a pre-trained GAN network on a public domain from Wikiart, and I used it as my that had been trained with tens of thousands of artistic photos training starting point.

Samples from the art works used for the Wikiart pretrained network



This pretrained network not only improved the quality of the generated images but also added artistic abstract touch from the wikiart dataset. I ran again the training process with the same configurations and number of epochs as before and got the following results

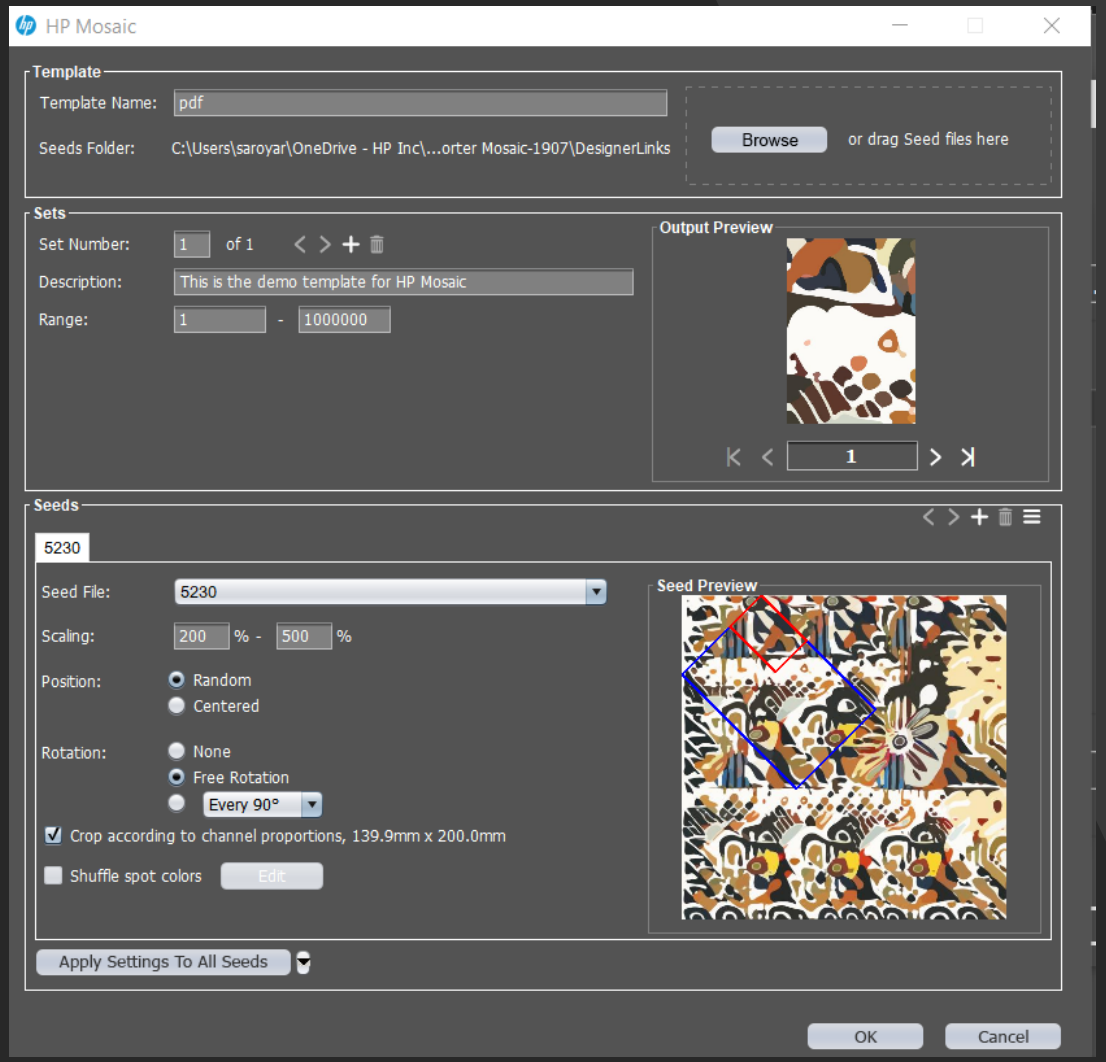





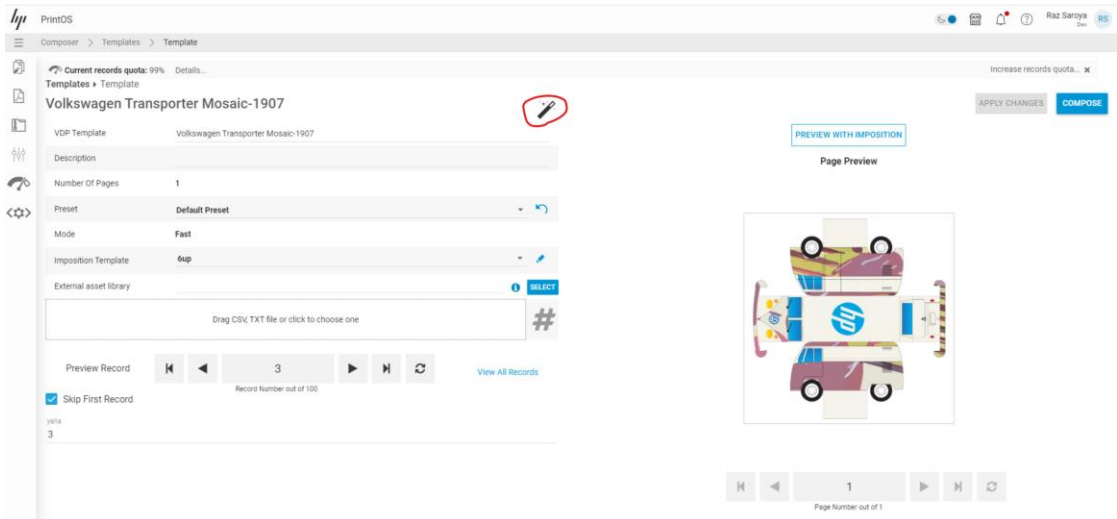
Large samples

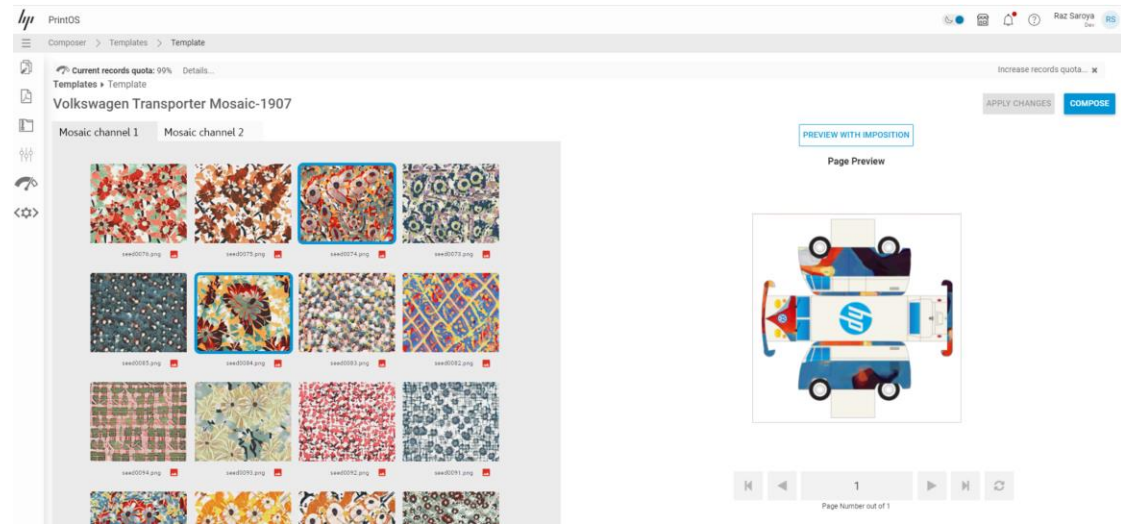
Mosaic algorithm with the generated seeds

The following image is HP Mosaic plugin for Adobe Illustrator, I selected the first sample seed from above and configured free rotation, random position and scaling from 200%-500%.



The output of the algorithm with custom template


PrintOS
Raz Saroya
Composer > Templates > Template
Current records quota: 99% Details...
Increase records quota...
Templates ▸ Template
Volkswagen Transporter Mosaic-1907
APPLY CHANGES
COMPOSE
VDP Template
Volkswagen Transporter Mosaic-1907
Description
Number Of Pages
1
Preset
Default Preset
Mode
Fast
Imposition Template
6up
External asset library
SELECT
Drag CSV, TXT file or click to choose one
Preview Record
3
Record Number out of 100
View All Records
Skip First Record
yalla
3
PREVIEW WITH IMPOSITION
Page Preview
1
Page Number out of 1


PrintOS
Raz Saroya
Composer > Templates > Template
Current records quota: 99% Details...
Increase records quota...
Templates ▸ Template
Volkswagen Transporter Mosaic-1907
APPLY CHANGES
COMPOSE
Mosaic channel 1
Mosaic channel 2
PREVIEW WITH IMPOSITION
Page Preview
1
Page Number out of 1

Projecting

- The StyleGAN generator model is basically a network of weights, we can try to represent an input image with those weights (projecting an input image into the latent space of the model). We can use this projection to combine the style of an input image with a given seed by learning the input image.