

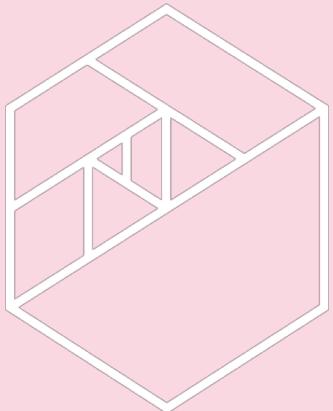
Making Faces: Conditional generation of faces using GANs via Keras+Tensorflow

SOPHIE SEARCY

Who am I?

Who am I?

- ▶ Sophie Searcy
- ▶ Curriculum development lead and Data Science Instructor at Metis
- ▶ Deep Learning and Data Science Ethics
- ▶ Write and lead free workshops with t4tech



METIS



4	Standard Error	180921.196
5	Median	2079.10532
6	Mode	163000
7	Standard Deviation	140000
8	Sample Variance	79442.5029
9	Kurtosis	6311111264
10	Skewness	6.53628186
11	Range	1.88287576
12	Minimum	720100
13	Maximum	34900
14	Sum	755000
15	Count	264144946
16	Confidence Level(95.0%)	4078.35485



Signposting

Who is this for

Someone who

- ▶ Understands and can explain the fundamentals of modern Deep Learning (there will be a review)
 - ▶ BackProp
 - ▶ Stochastic Gradient Descent
 - ▶ Common loss and activation functions
- ▶ Has built models using a recent Deep Learning package (PyTorch, Theano, Keras, etc.)

What we'll cover

Students should be able to:

- ▶ Understand and explain the important components of Generative Adversarial Networks
- ▶ Use provided boilerplate code and adapt it for new purposes
- ▶ State of The Art techniques in GANs:
 - ▶ Students will be exposed to a few important, recent developments.
 - ▶ Students will have the building blocks needed to independently explore new techniques.

Deep Learning Review

Essential parts: Differentiable functions

- ▶ $h_0 = f(W_0^T x + b_0)$
- ▶ $h_1 = f(W_1^T h_0 + b_1)$
- ▶ ...
- ▶ $y = f(W_n^T h_n + b_n)$
- ▶ DL models use these functions to process data in steps from input → output
 - ▶ Traditional application:
 - ▶ Tabular data → Regression/Classification
 - ▶ New (ish) applications
 - ▶ Image → Text
 - ▶ Image → Image

Essential parts: Stochastic Gradient Descent + BackProp

Gradient Descent

- ▶ Finds adjustment to function parameters that minimizes the loss function

Back Propagation

- ▶ Chain rule of calculus in algorithm form.
- ▶ Applies gradient descent over many layers of a network.

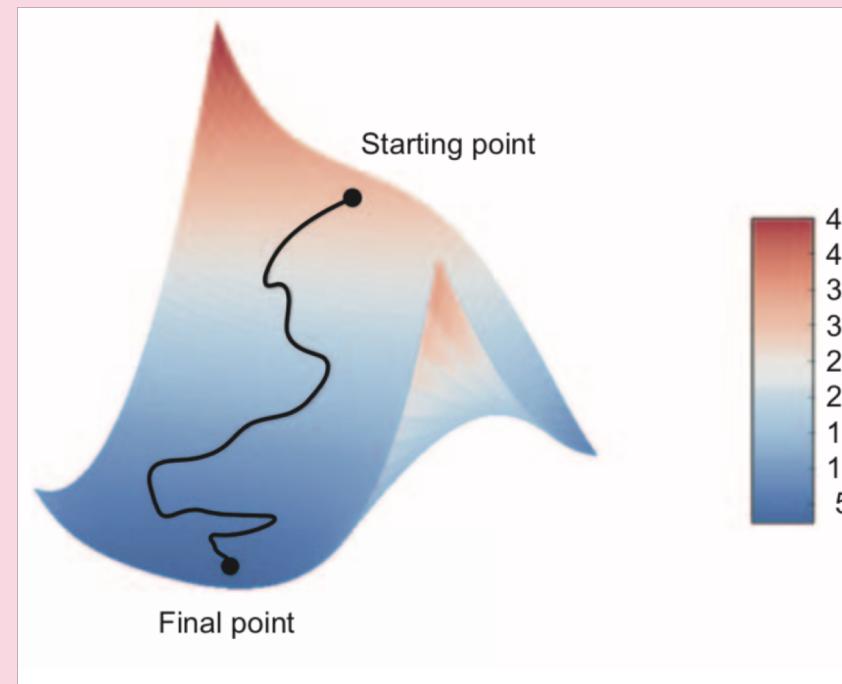
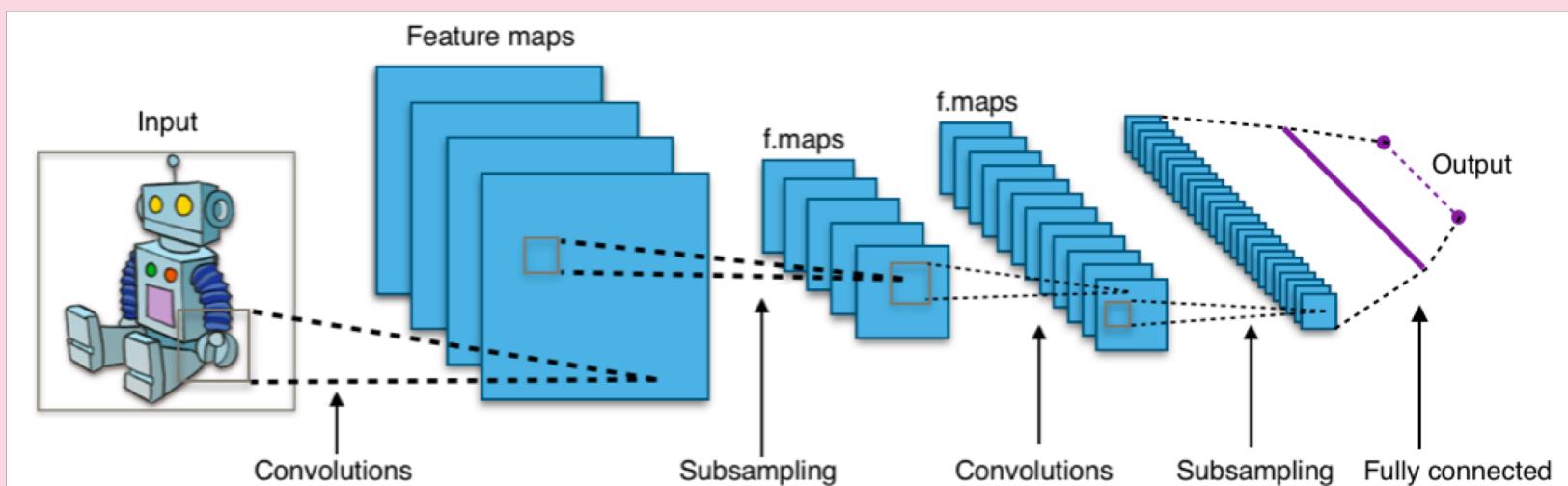


Figure 2.12 Gradient descent down a 2D loss surface (two learnable parameters)

GAN Overview

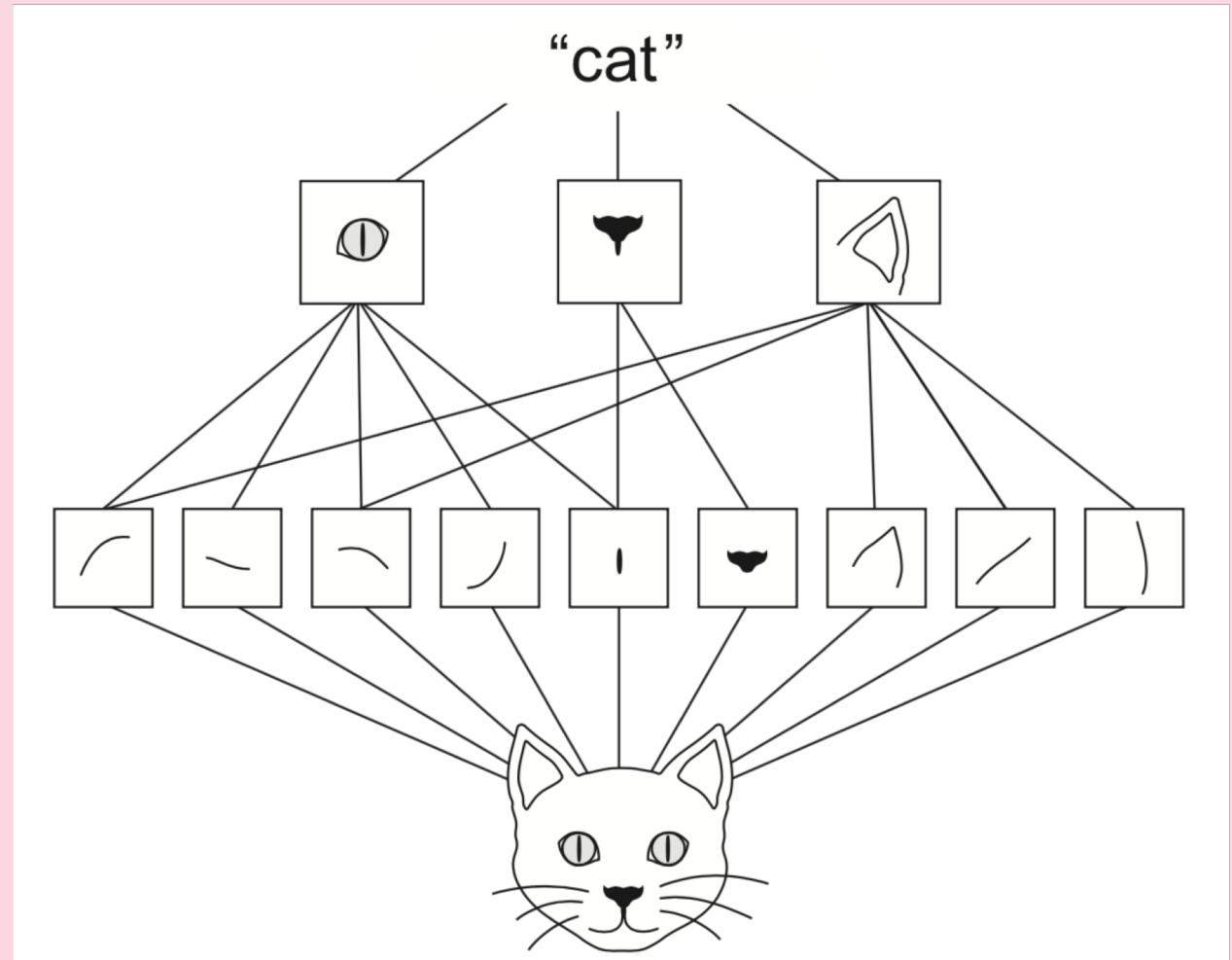
Convolutional Classifiers

- ▶ Convolutions learn feature maps
- ▶ Use sampling/pooling to summarize over height and width of image
- ▶ Output is some classification vector, e.g. probabilities

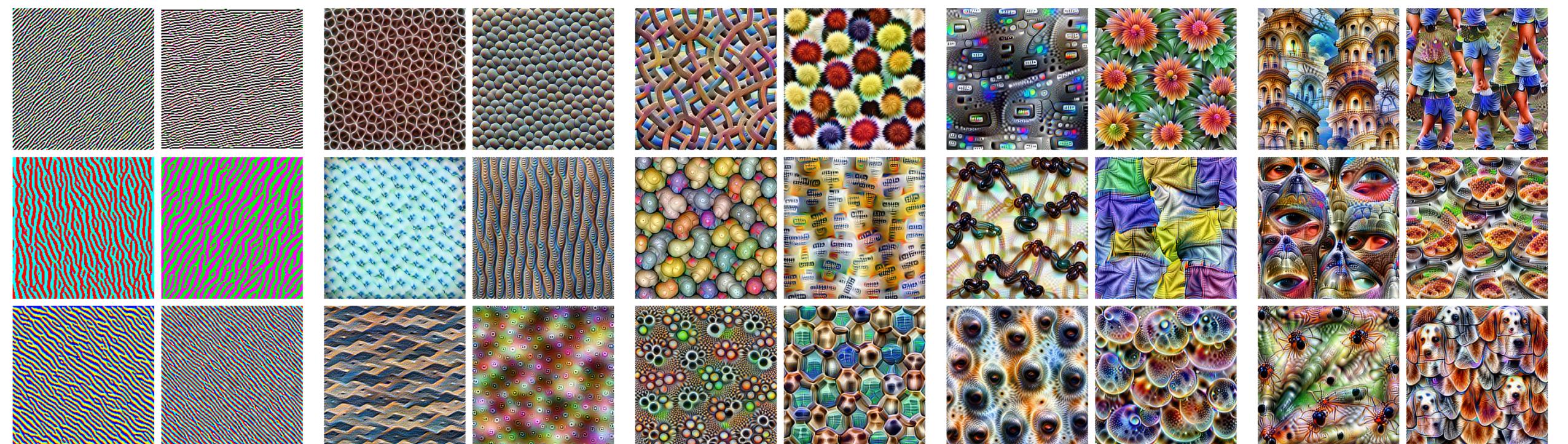


[source](#)

- ▶ Convolutional filters
- ▶ Pixels → subparts → parts → whole

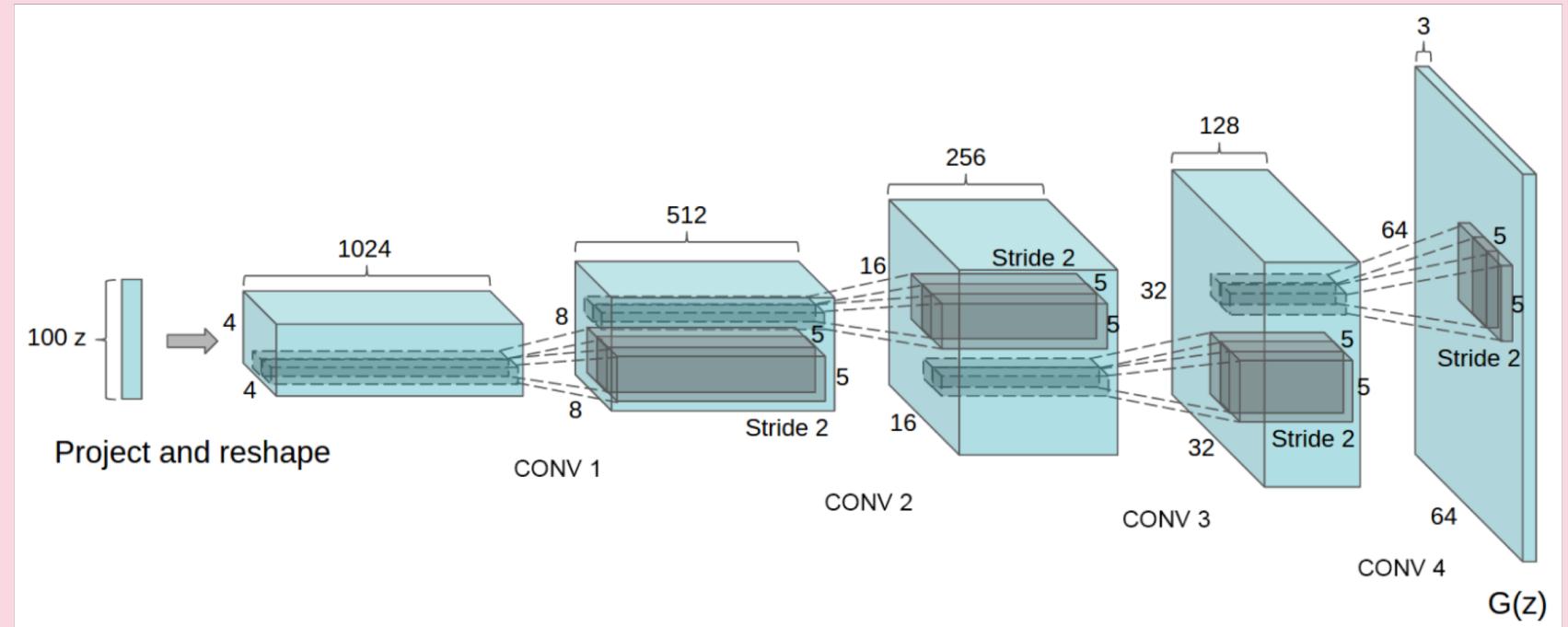


- ▶ Convolutional filters
- ▶ Pixels → subparts → parts → whole
- ▶ Visualize by finding input that maximizes activity at layer

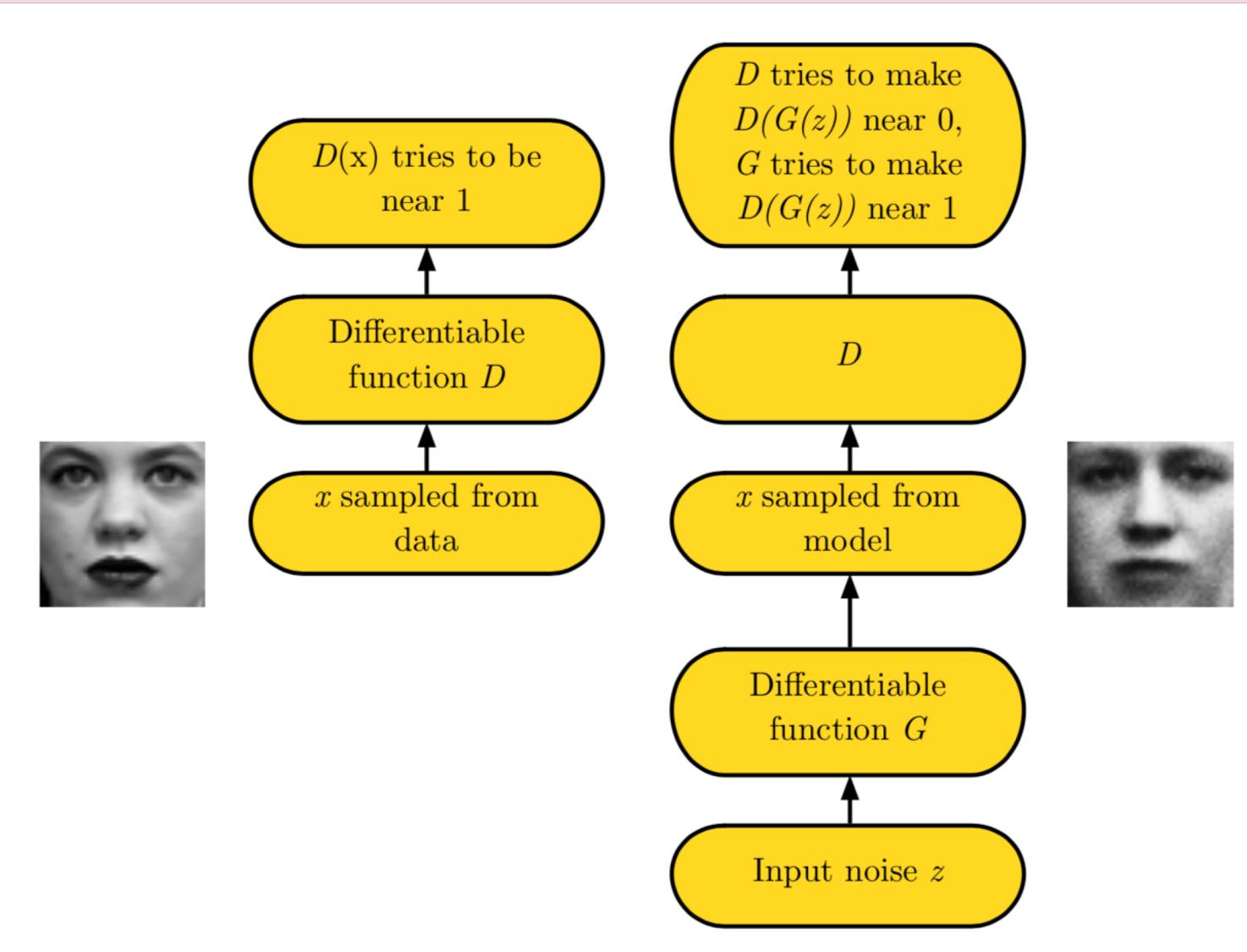


Convolutional Generation

- ▶ Convolutions learn feature maps
- ▶ Upsampling/DeConvolution progressively grow image



GAN Architecture



Generator task

A single CIFAR frog



Generator task

First 100 CIFAR frogs



Generator task

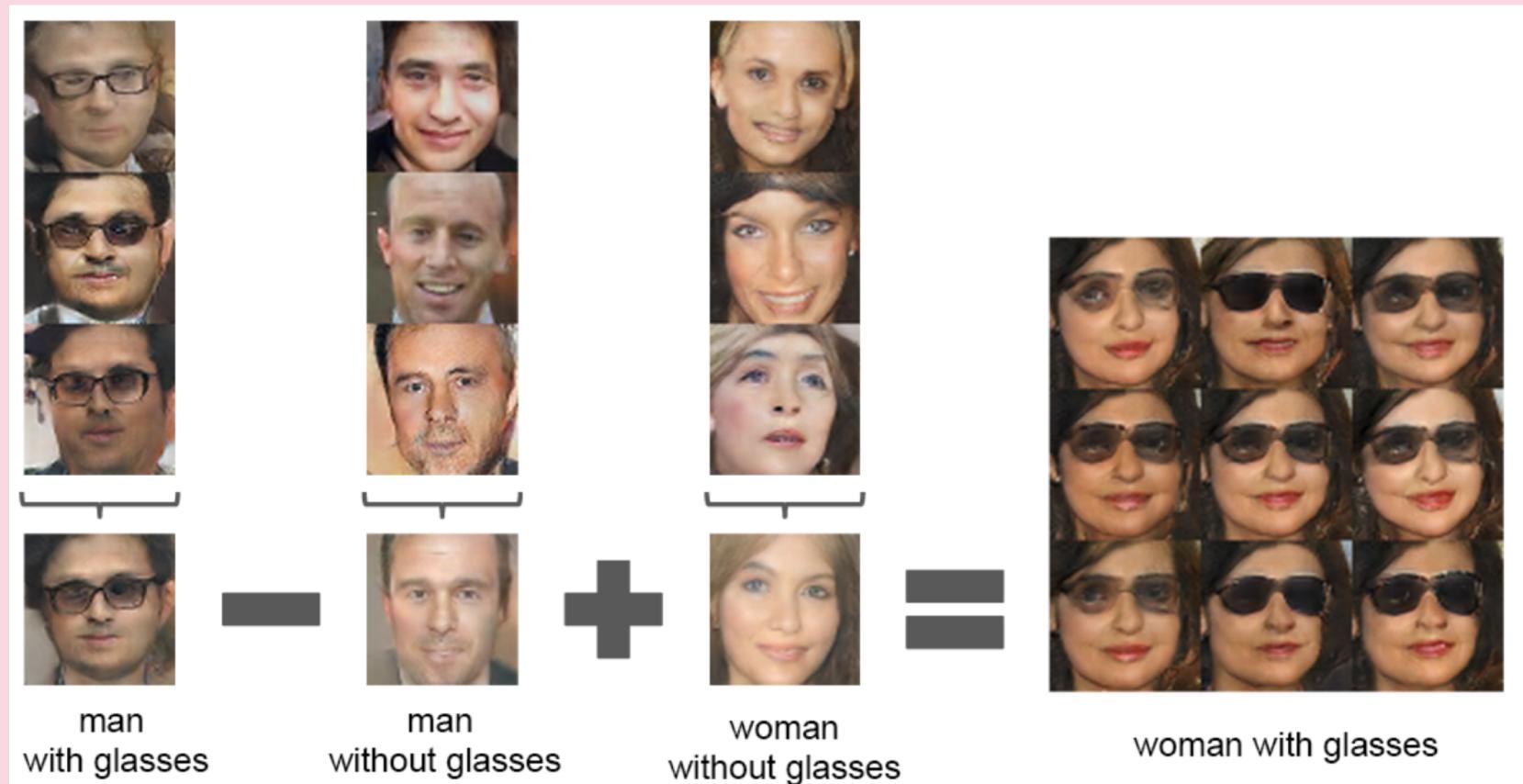
The generator learns the
distribution of the training data.



Distribution Learning

Generator learns *distribution* of training data

- ▶ Meaningful understanding of that training data



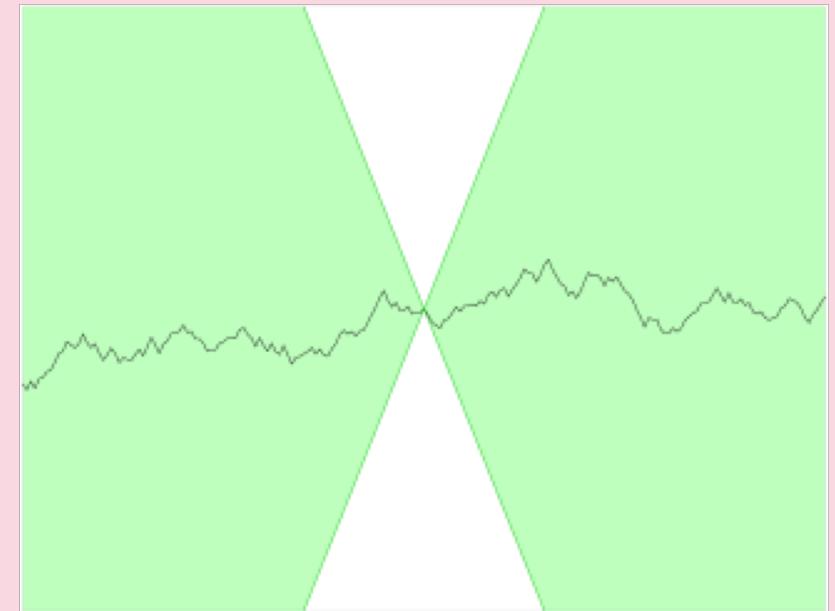
Advanced GAN Topics

Lipschitz Continuity

- ▶ Problem: in many cases, the discriminator can be essentially impossible for the generator to beat.
 - ▶ Impossible to win → zero gradient → no learning

Lipschitz constant: Maximum rate of change of a function

Spectral Normalization (Miyato et al 2018) constrains the Lipschitz constant of the discriminator, ensuring stable training of generator.



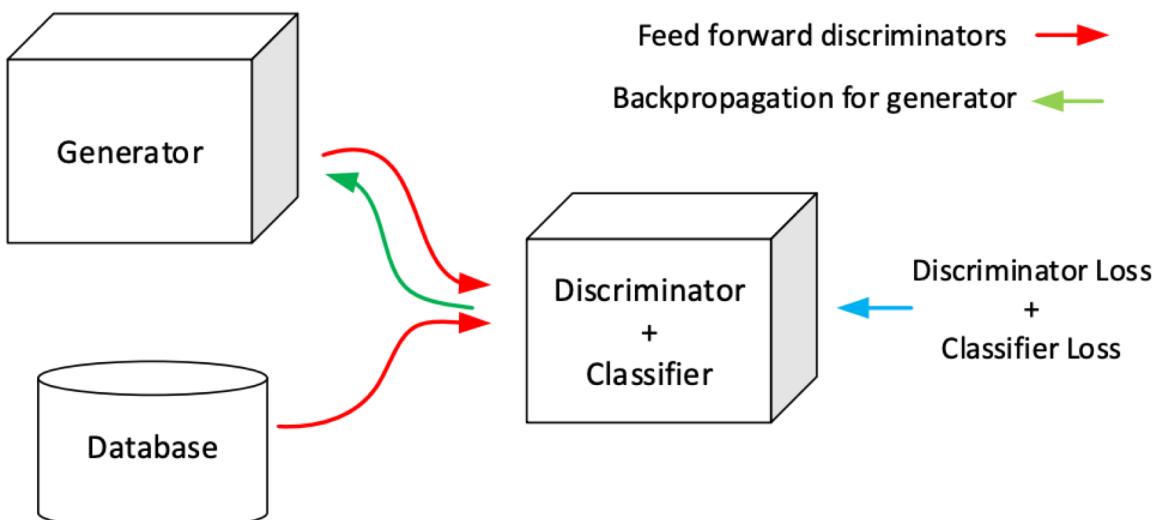
[source](#)

Multilabel Conditional GAN

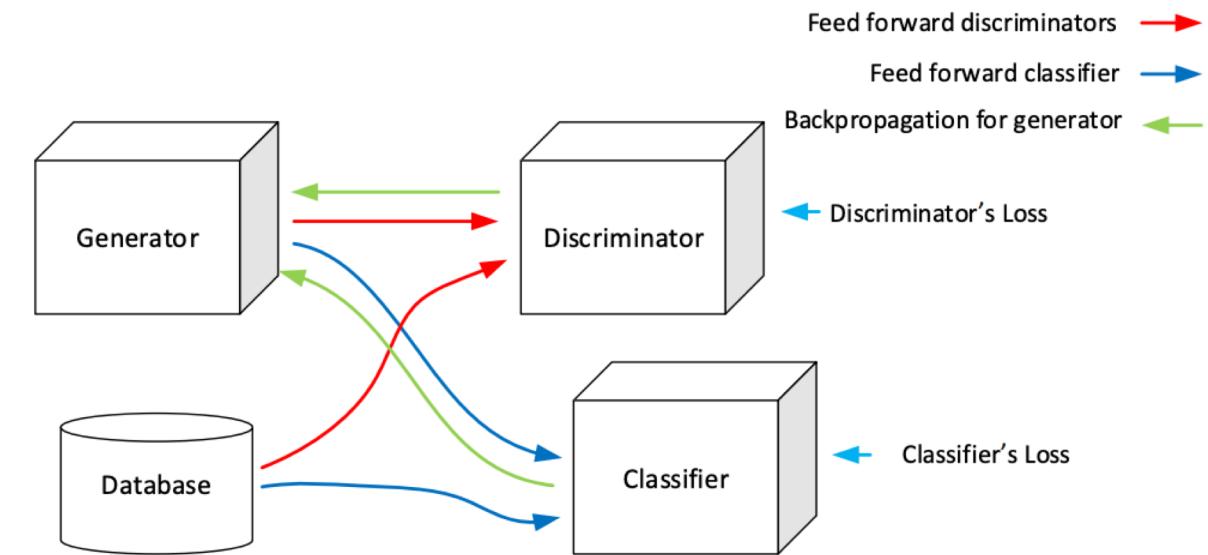
Classifier + GAN

- ▶ Classes provide additional signal
 - ▶ both generator and discriminator learn data distribution more quickly
 - ▶ Significantly quicker learning (wall clock)
- ▶ Allows direct manipulation of class feature in generator

VAC GAN



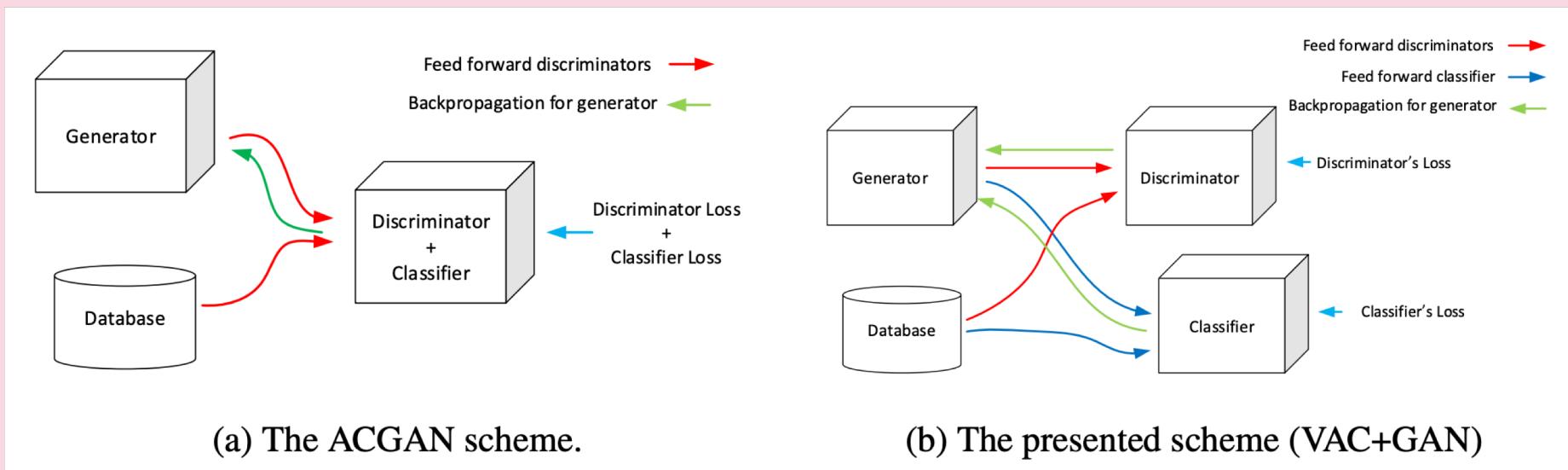
(a) The ACGAN scheme.



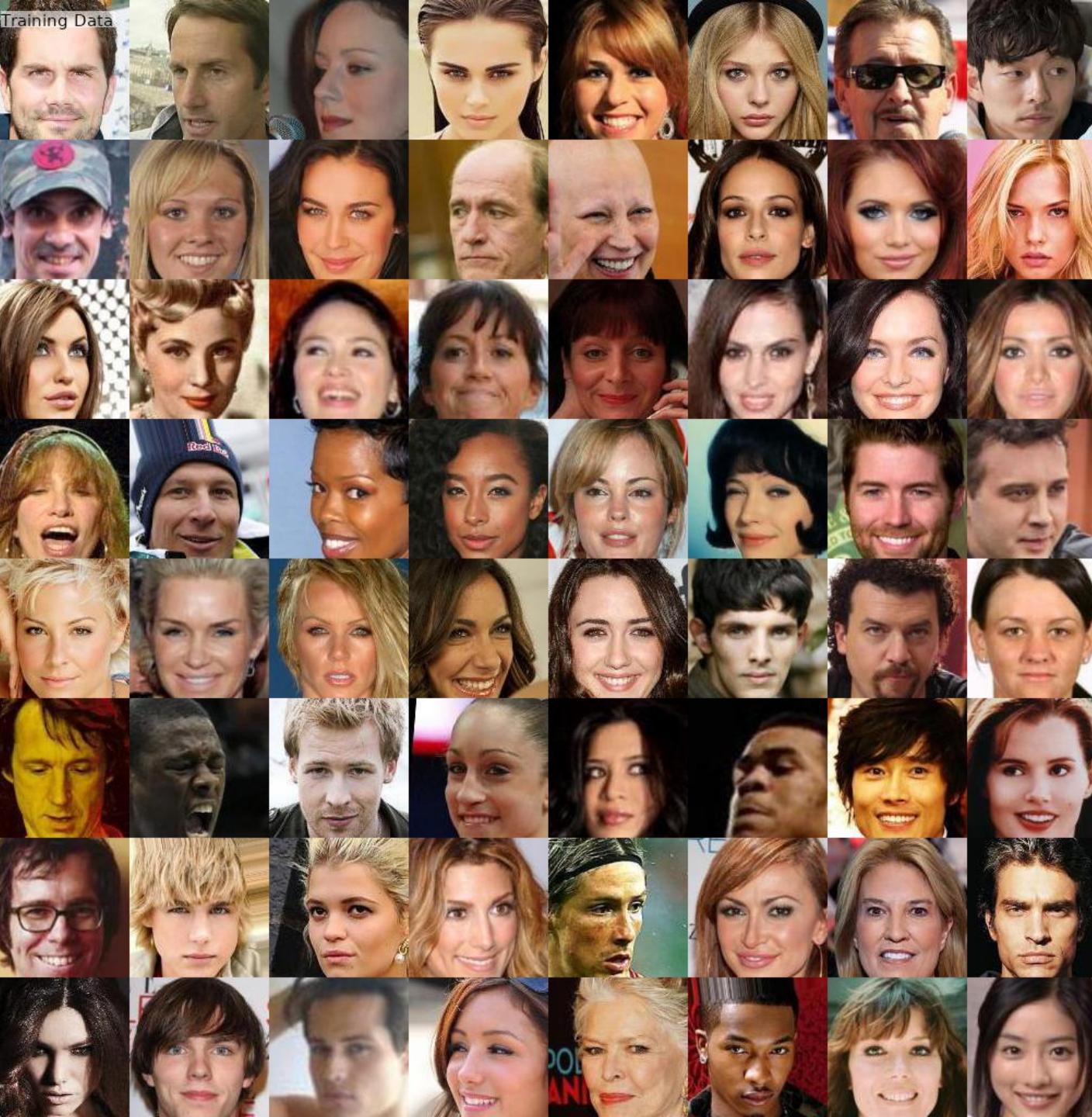
(b) The presented scheme (VAC+GAN)

VAC GAN

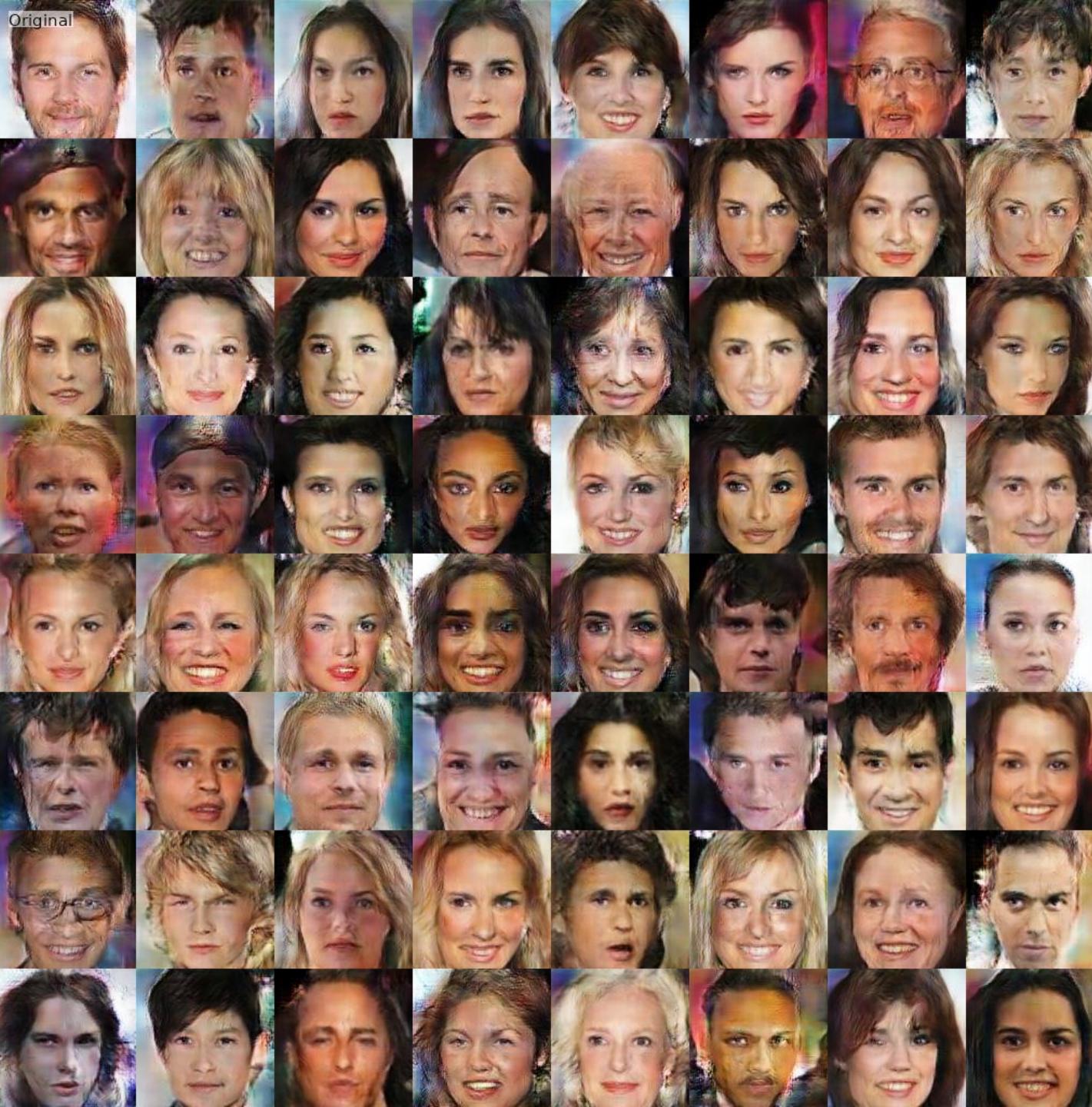
- ▶ VAC GAN
 - ▶ Good: Versatile classification with GAN
 - ▶ Bad: Requires a 3rd model
- ▶ Today's demo: VAC-GAN variant that combines Discriminator and Classifier



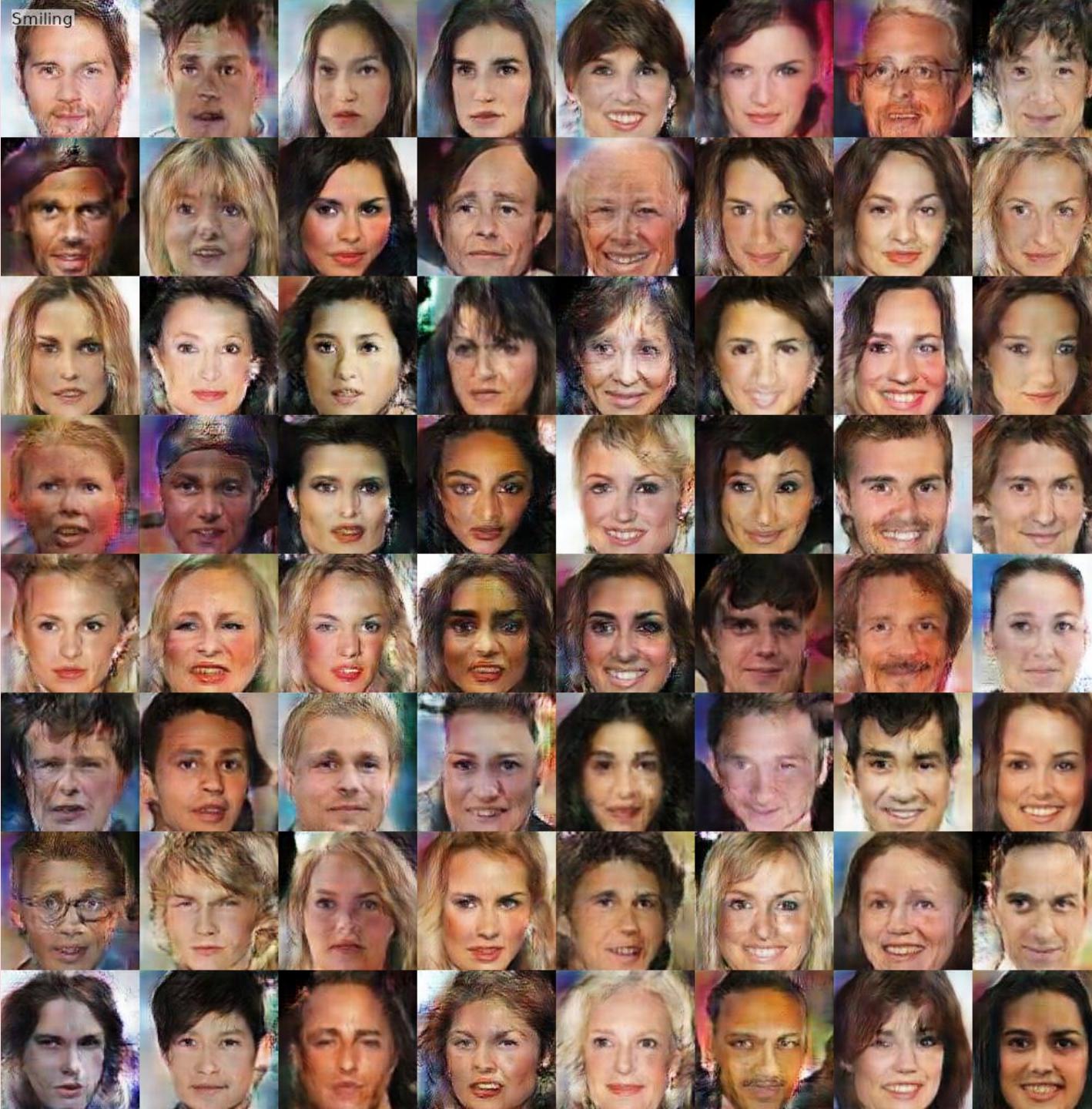
Results



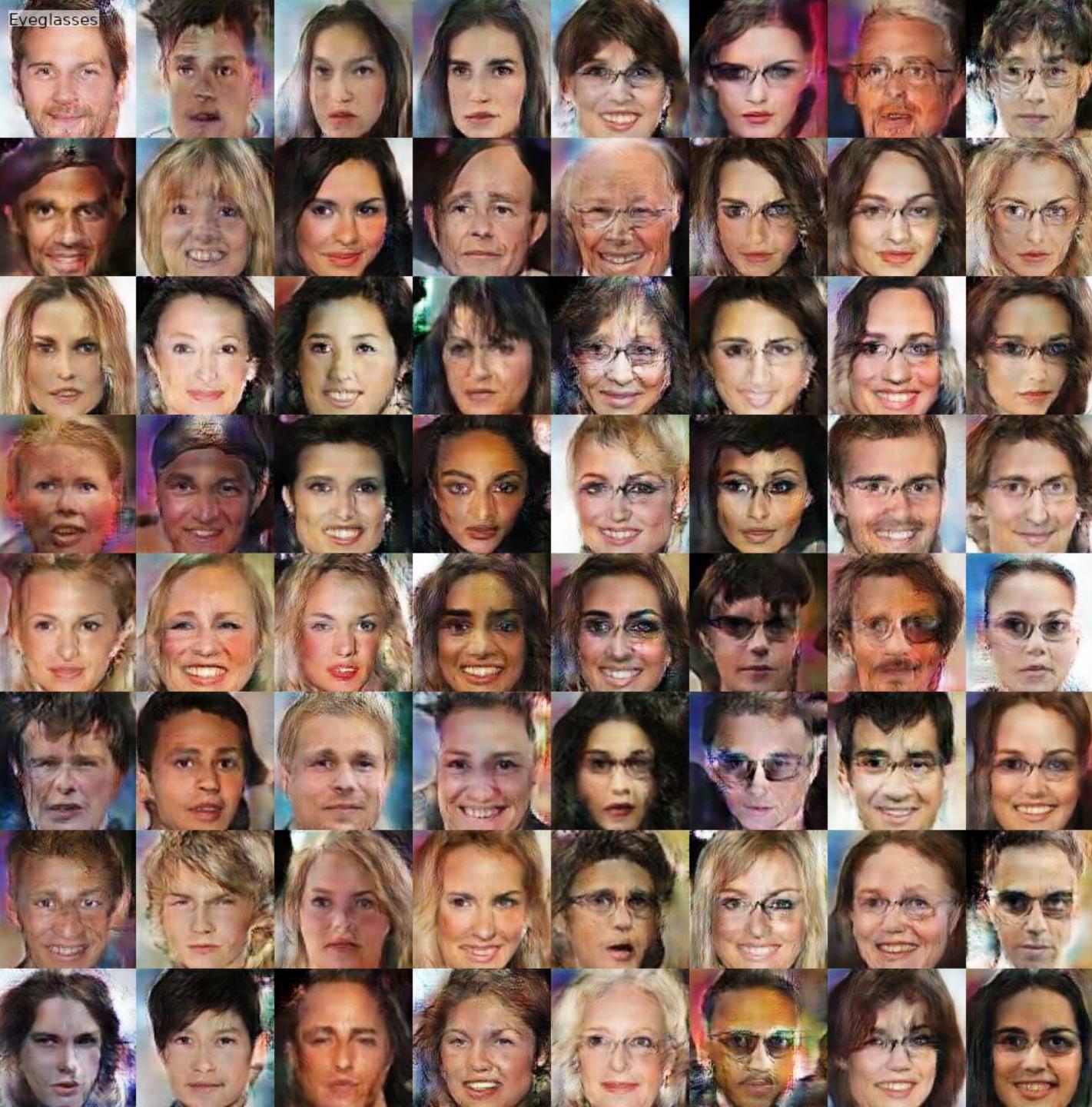
Results



Results



Results



Results

