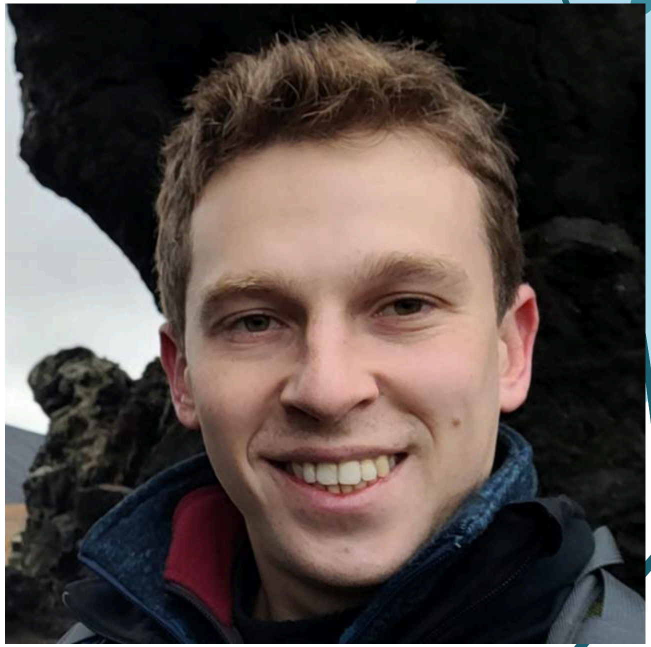
Neural Networks Based Graphics

Generative Models

About me

- Daniel Roich
- Masters student under Dani's supervision
- Fell in love with NN based graphics at the the 3rd year of B.S.C thanks to participating in the Sadna
- PTI – Pivotal Tuinng Inversion



Our plan

01 **Neural Networks**
Recap of Neural Networks

02 **VAE**
Variational Auto Encoders

03 **GAN**
Generative Adversarial Network

04 **GANs Results**
Real World Applications

05 **Editing**
Editing images using GANs

06 **The Future**

01

Neural Networks

Neural Networks recap

What is learning

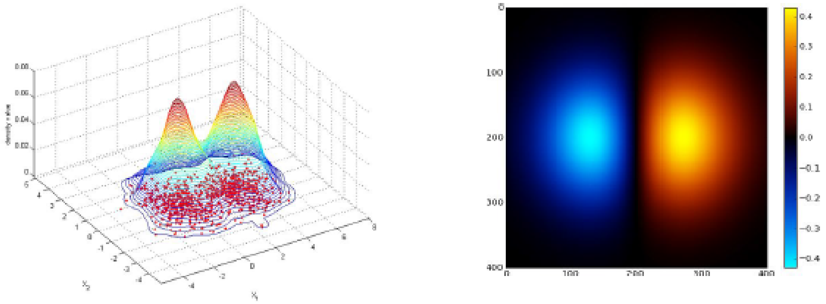
A cat sitting on a suitcase on the floor

- Data – (X, y)
- Goal – learn a function $f: X \rightarrow y$
- Examples – Classification, Regression, Object Detection and Image Captioning

Unsuperivsed Learning

Figure copyright Ian Goodfellow, 2016. Reproduced with permission.

1-d density estimation



2-d density estimation

- Data – X
- Just data, no labels

- Goal – learn some underlying hidden structure of the data

- Examples – Clustering, density estimation and feature learning

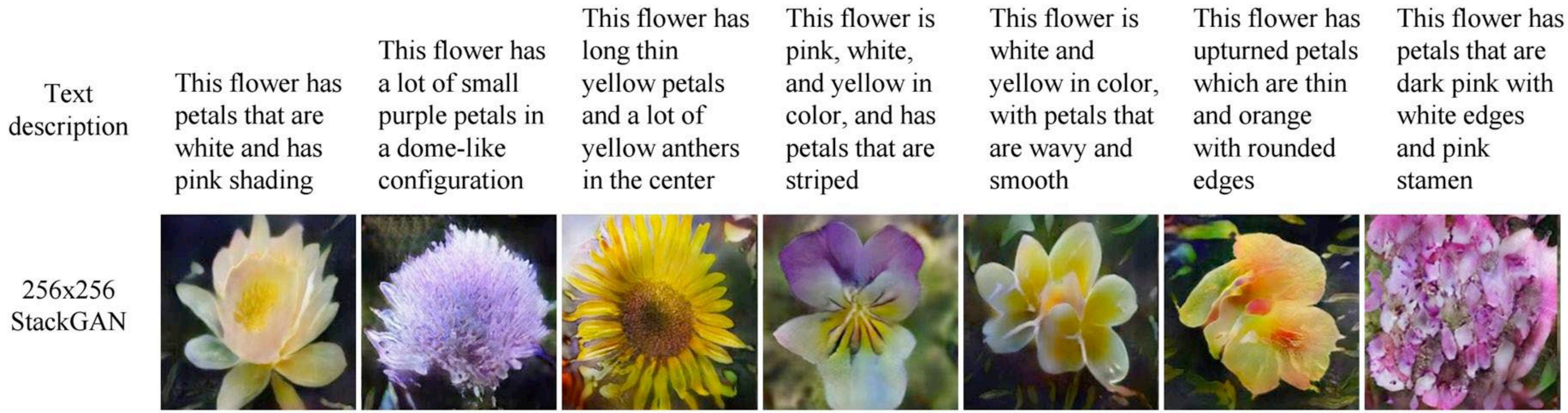
02

VAE

Variational Auto Encoders

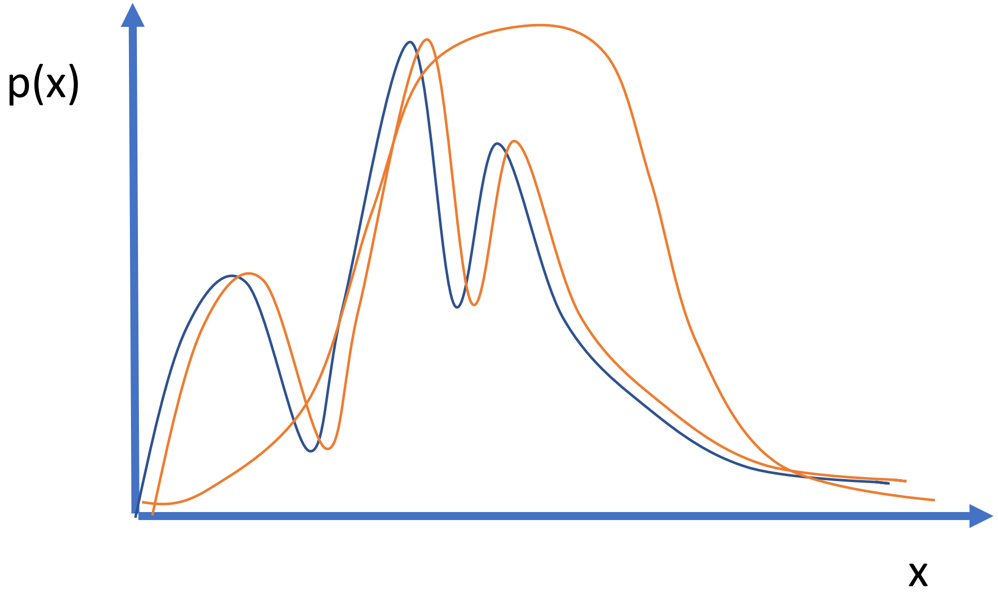
Generative Models

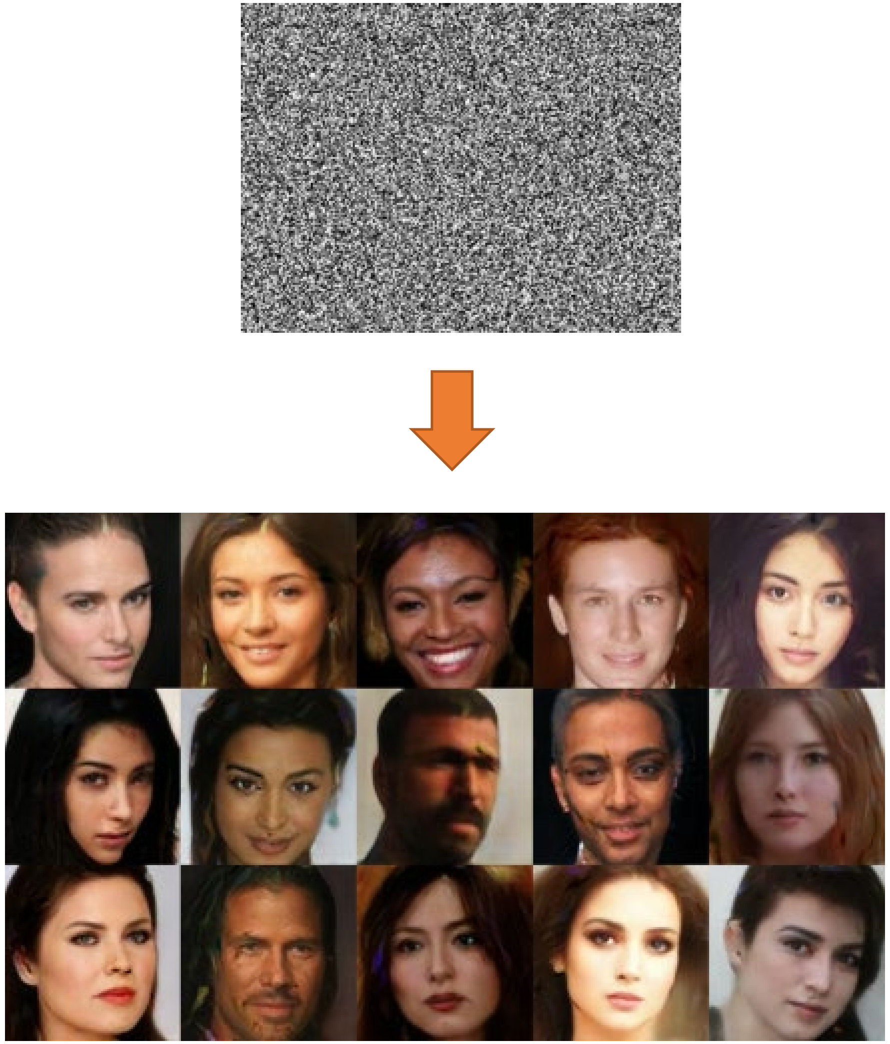
Why are deep generative models useful?
Why are deep generative models useful?
Why are deepgenerative models useful?



Generating Models

- Real data distribution:



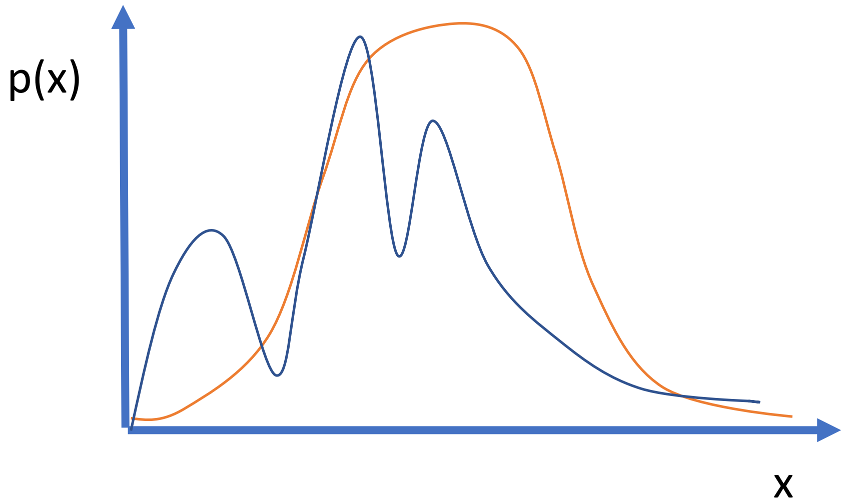


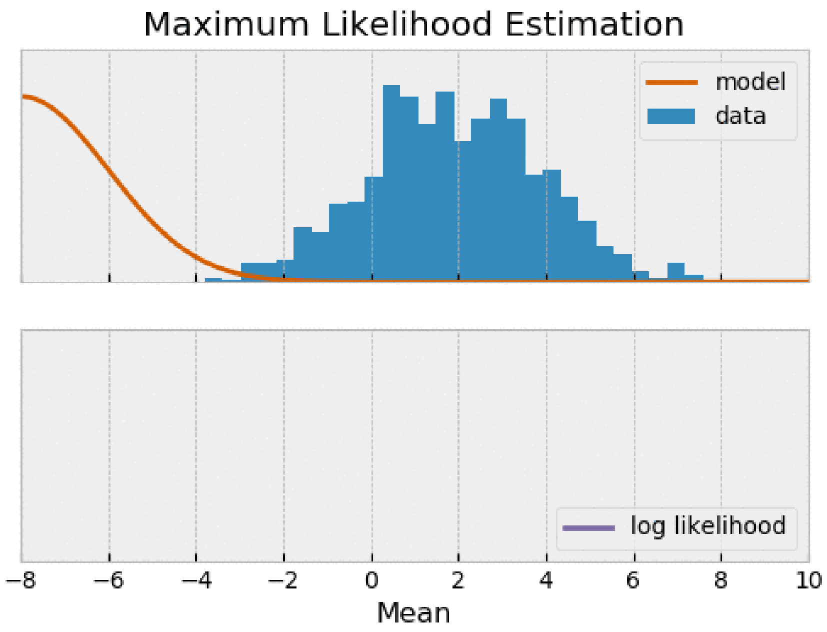
- Create new data drawn from the same distribution
- **Any distribution** in d dimensions can be generated by taking a set of d variables that are **normally distributed** and **mapping** them through a sufficiently complicated function

Maximum likelihood

- Maximize likelihood of training data:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{x \sim p_{\text{data}}} \log p_{\text{model}}(\boldsymbol{x} \mid \boldsymbol{\theta})$$





Example:
Data: generated from a Gaussian distribution with a variance of 4, and an **unknown** mean.
One may think of MLE as taking the Gaussian, sliding it over all possible means, and choosing the mean which causes the model to fit the data best.

https://towardsdatascience.com/maximum-likelihood-estimation-984af2dcfcac generative models

explicit density function

implicit density function

VAE

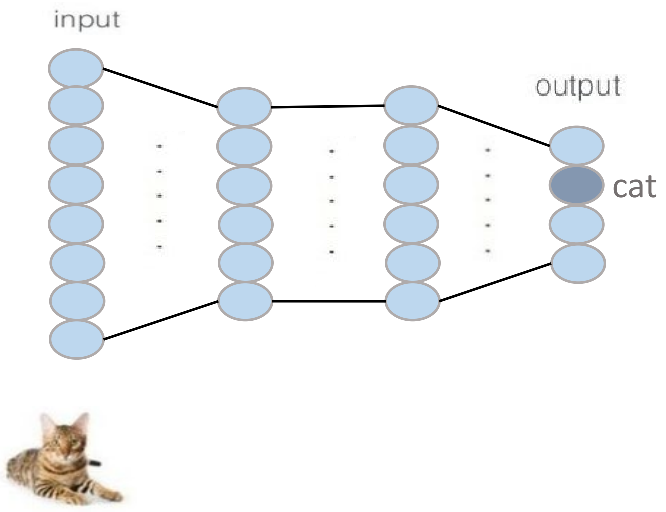
(Variational Auto Encoder)

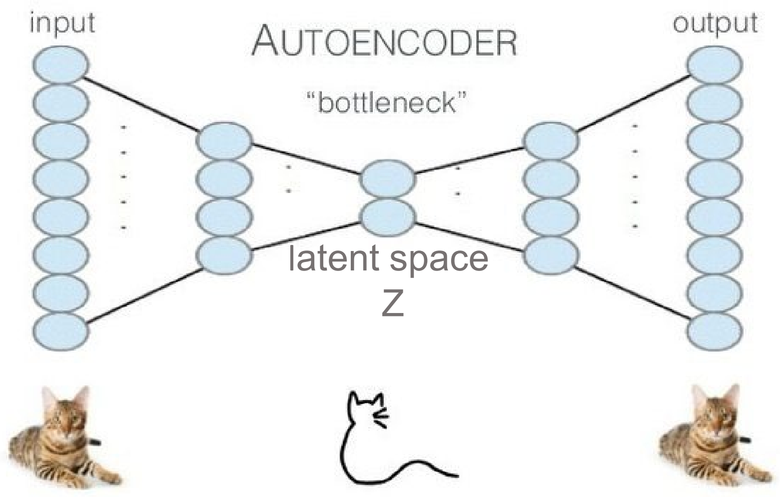
GAN

(Generative Adversarial Net)

Autoencoders

- Unsupervised learning – comparing output to input
- Finding a lower dimensional representation of the input

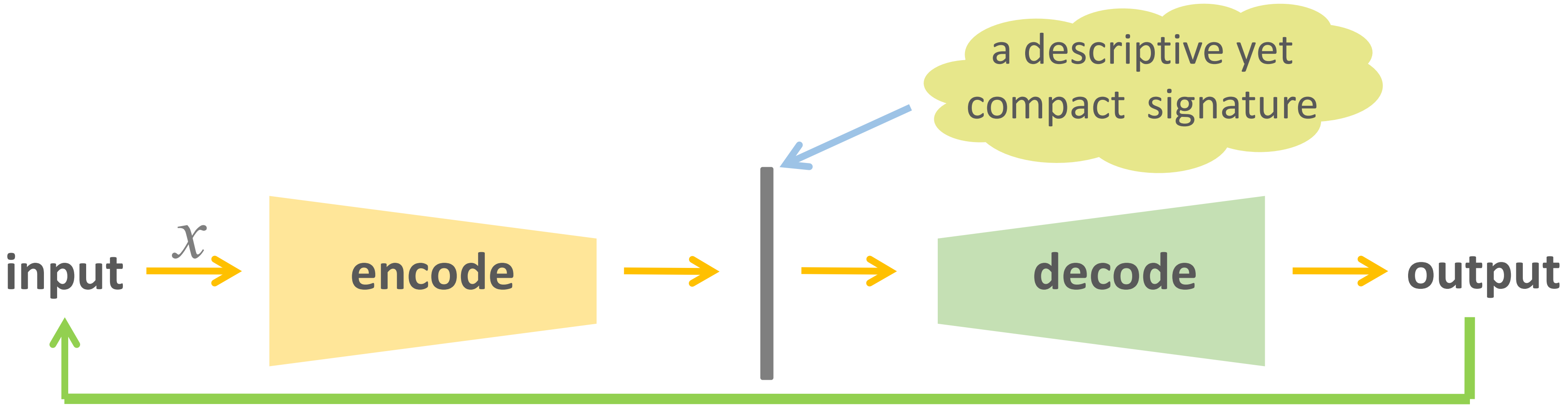




Why do we need a latent space?

- Original data contains unmeaningful representation
- Latent features capture meaningful data representation

Autoencoder Loss

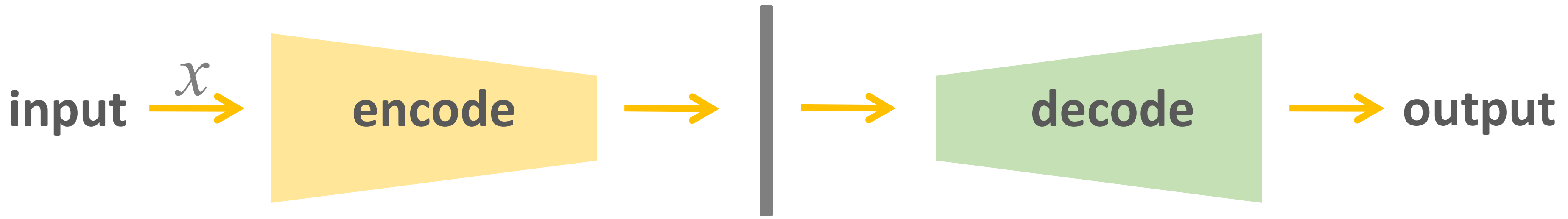


$$L = \| D(E(X)) - x \|_2$$

Reconstruction Loss

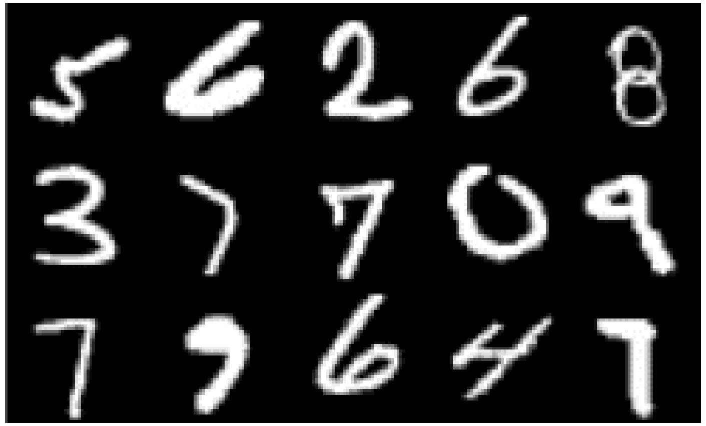
How can we use an autoencoder to generate new data?

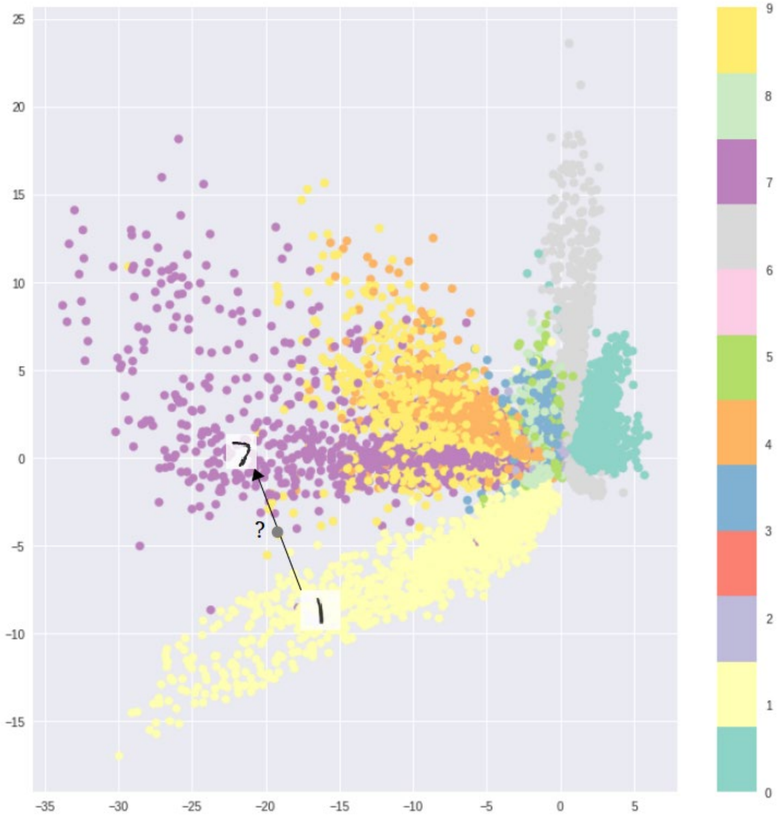
- Sample data from the latent space and reconstruct new images



How can we use an autoencoder to generate new data?

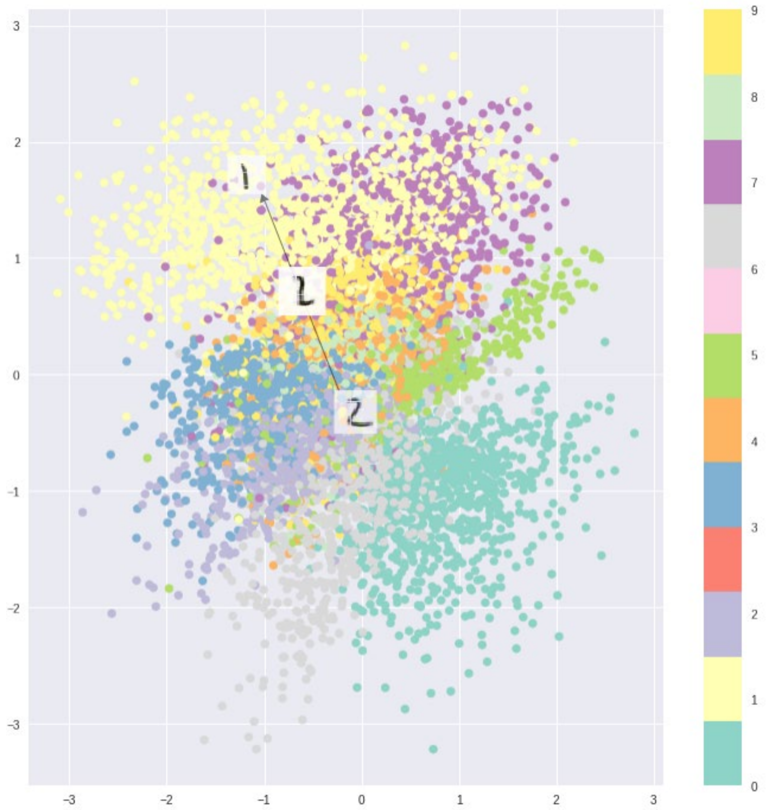
- Sample data from the latent space and reconstruct new images
- Can we do that with regular autoencoders?
    - Latent space not continuous
    - Hard to interpolate between points

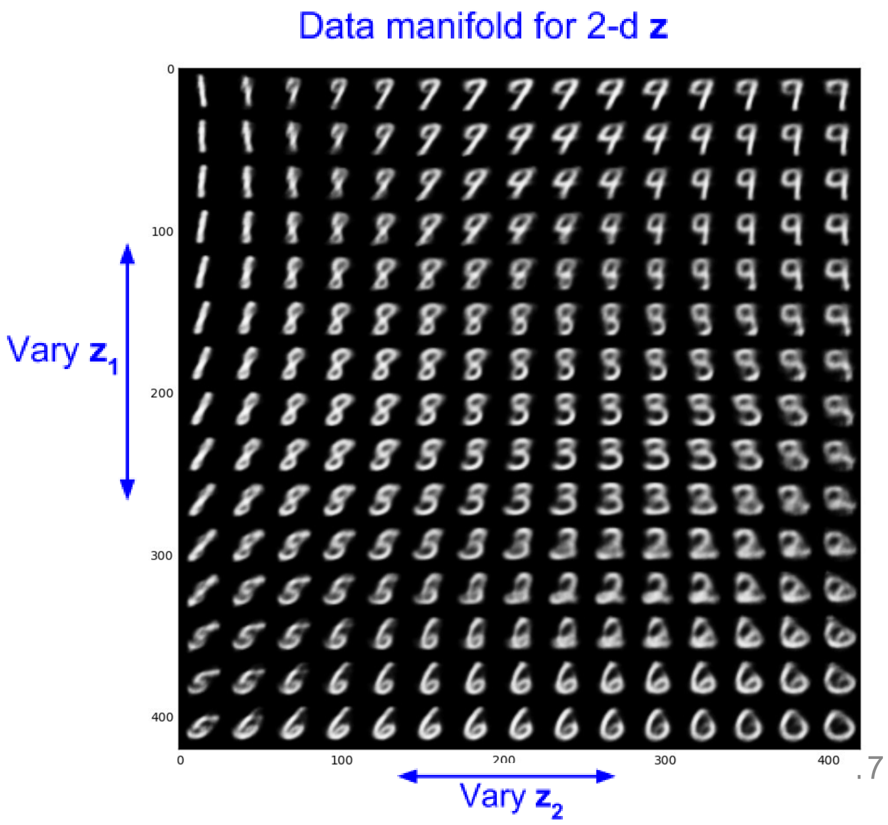




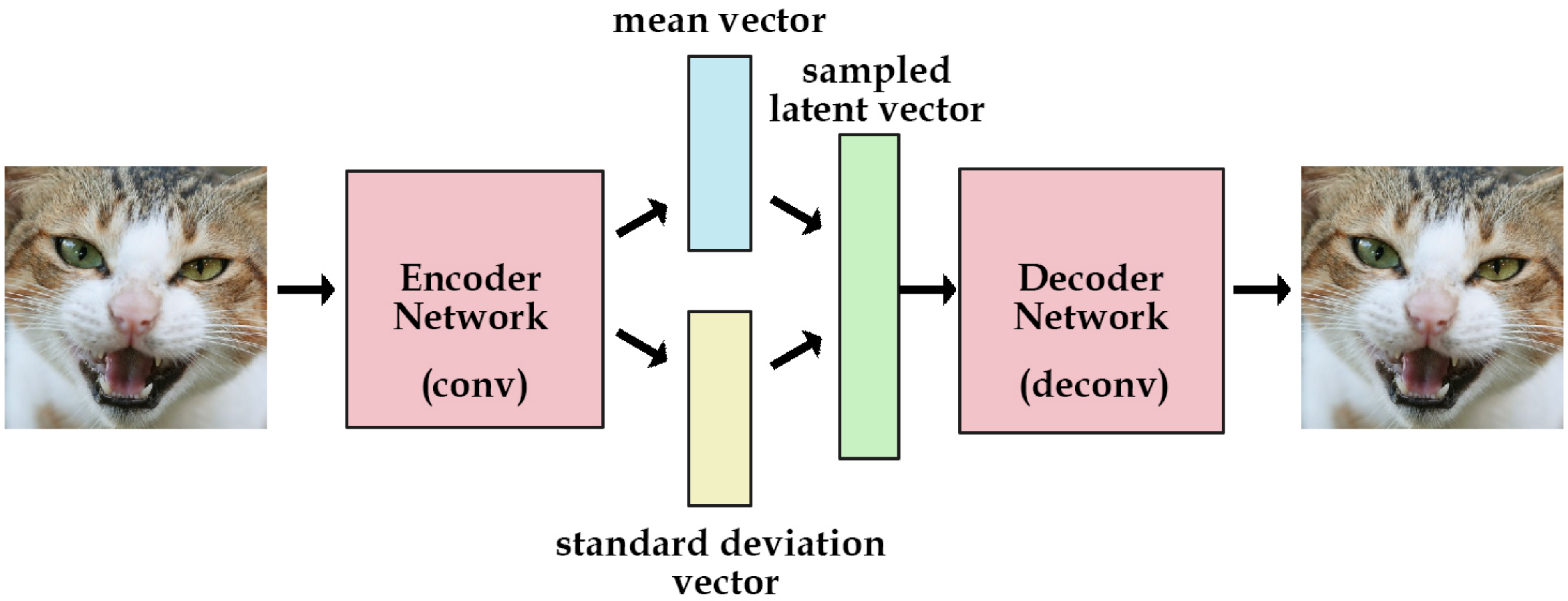
Solution – Variational Auto Encoders

- Make a continuous latent space
- Sample from the latent distribution to receive new data

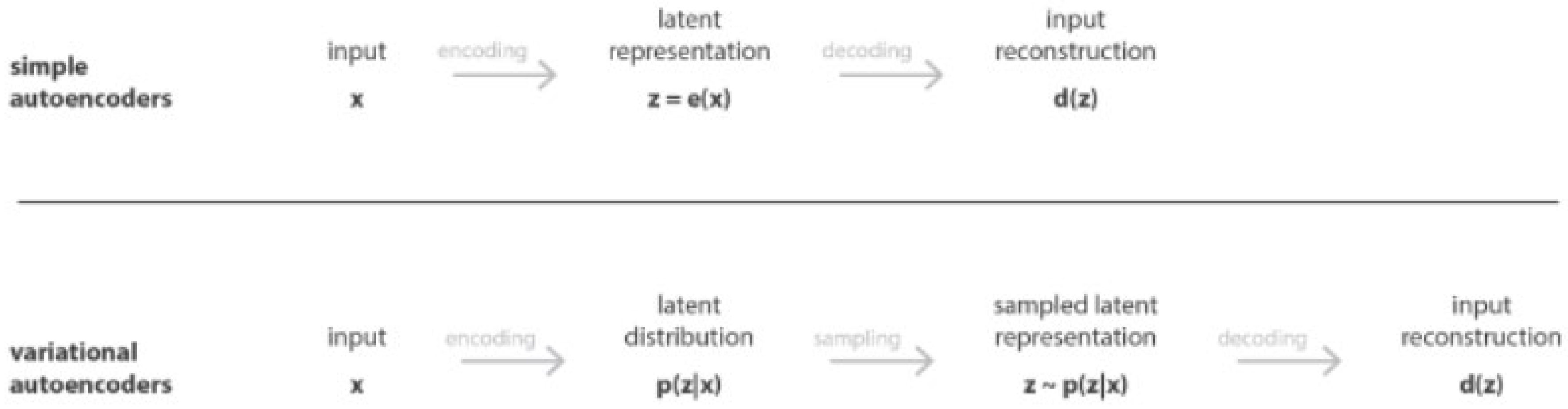




VAE
Kingma et al. - 2014



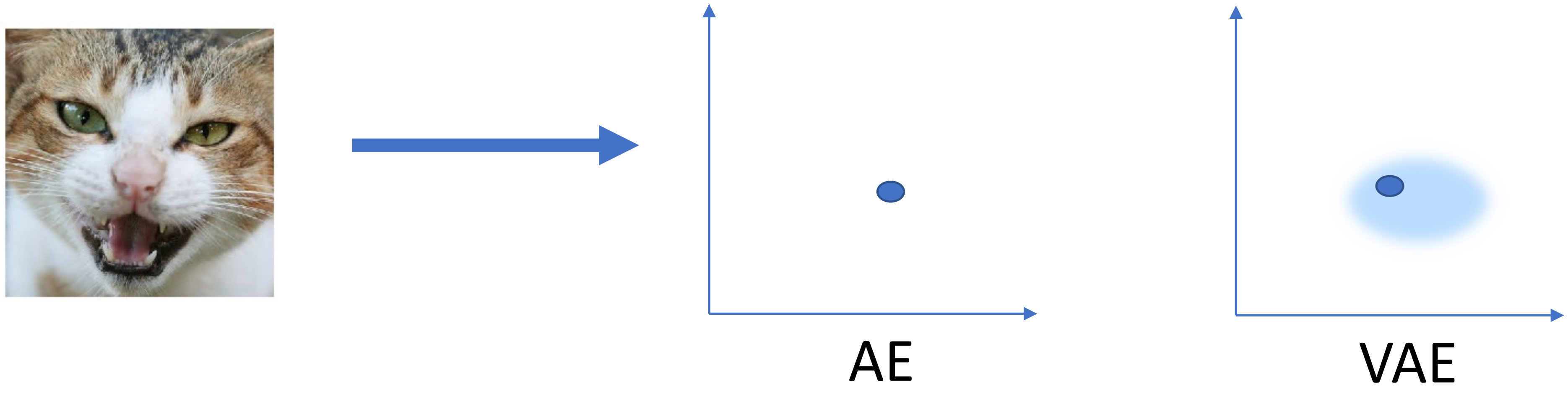
VAE Flow



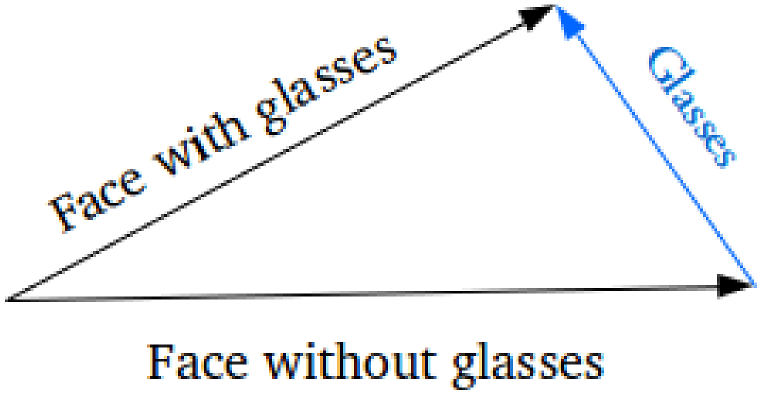
Difference between autoencoder (deterministic) and variational autoencoder (probabilistic).

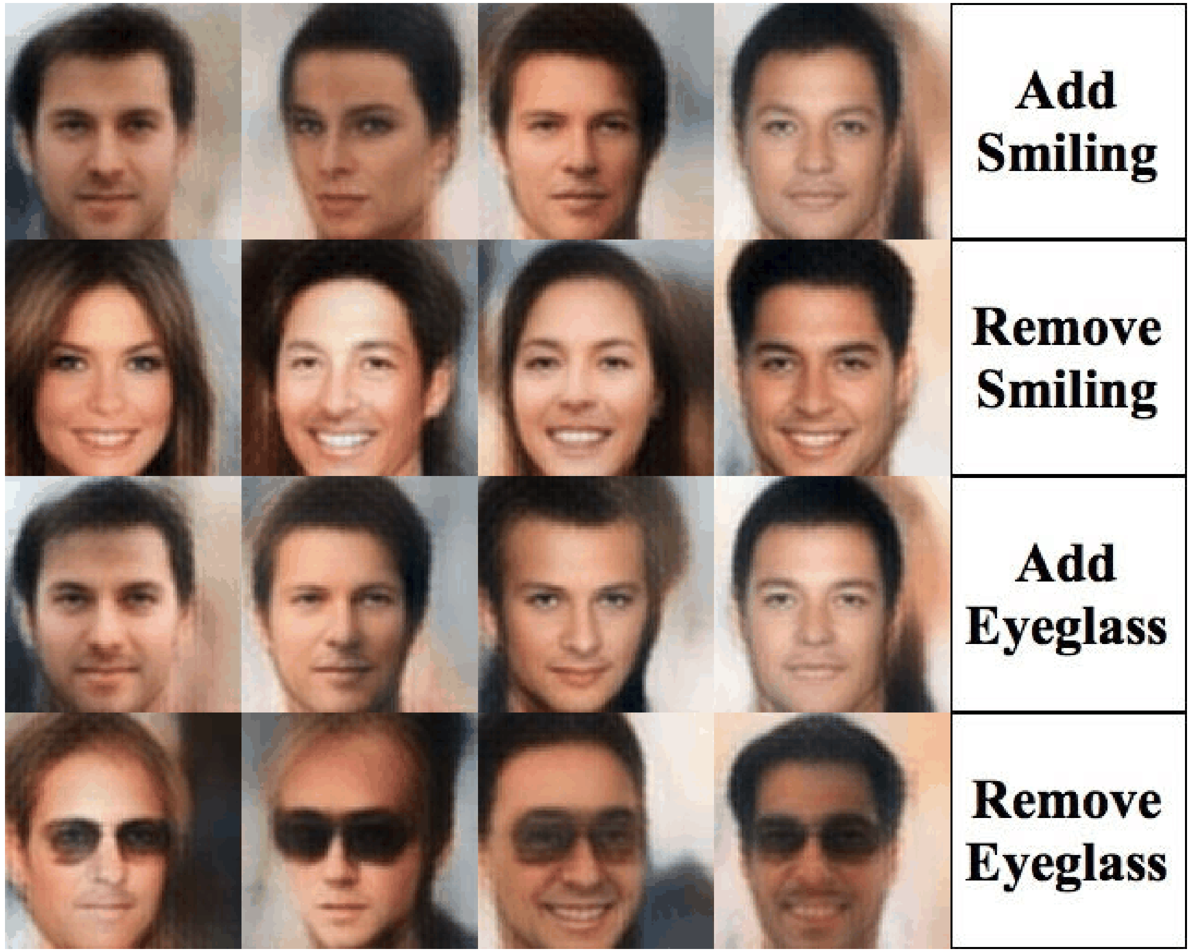
VAE

- Same image can be encoded in different ways
- Creates a continuous space
- Close points have similar meaning



Latent Space Arithmetic





https://houxianxu.github.io/assets/project/dfcvae

Summary-VAE

- Allows using autoencoder for data generation
- Explicitly models the density function
- Finds meaningful data features
- Disadvantage – blurry outcomes, both generative and reconstructive

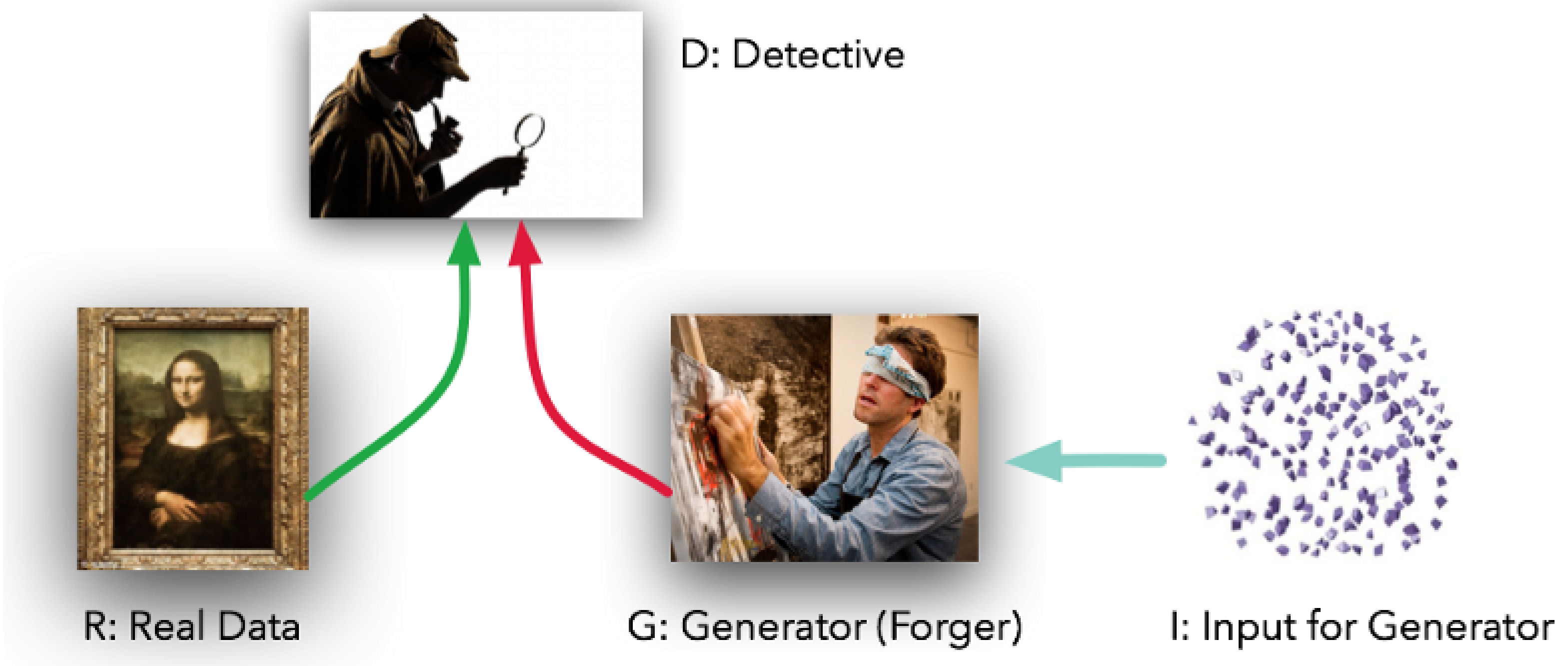
- What if we give up modeling the density function?

03

GAN

Generative Adversarial Network

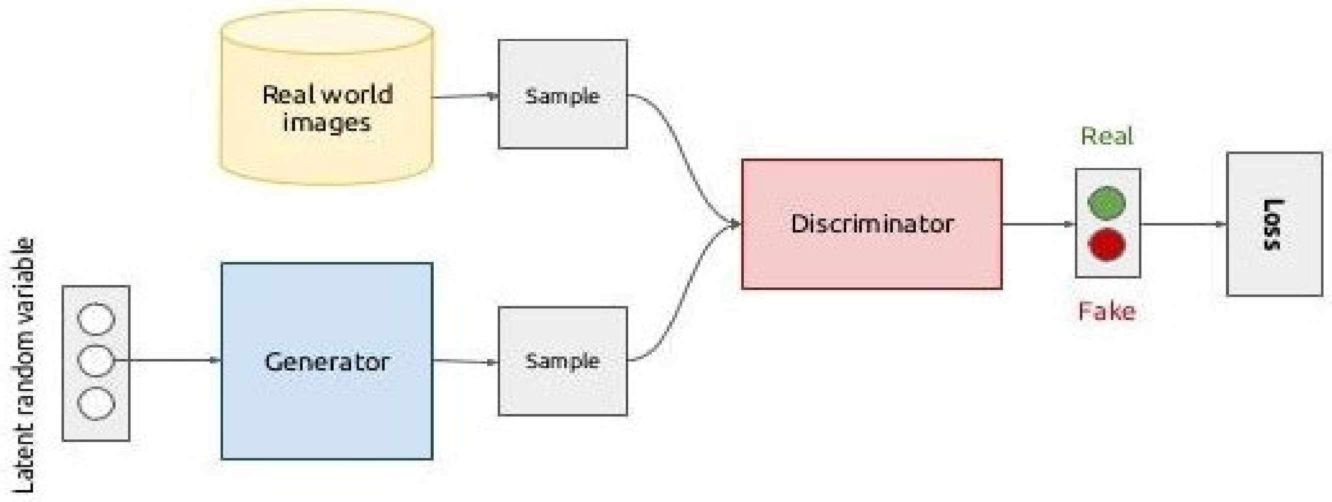
2 player game



Dive deeper

Discriminators Goal?

Generators Goal?



$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

How to Train?

Training Procedure

$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

Alternate between:

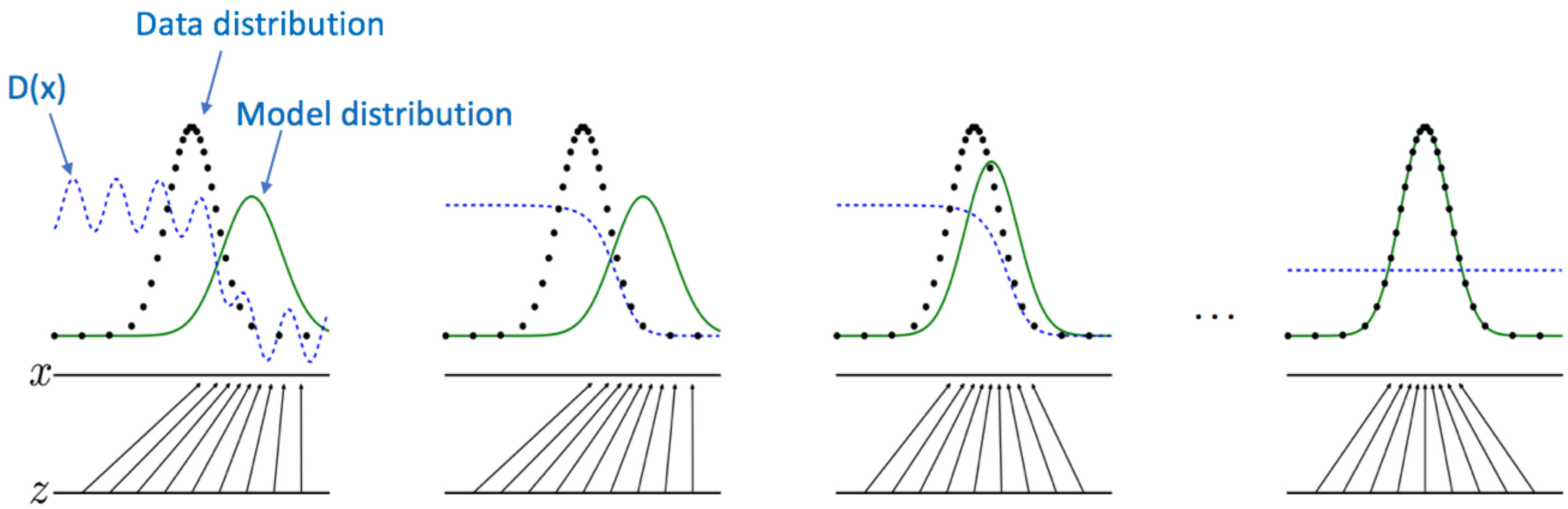
1. Gradient Acent on Discriminator

$$\max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

2. Gradient Decent on Generator

$$\min_{\theta_g} \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))$$

What its like to train a GAN



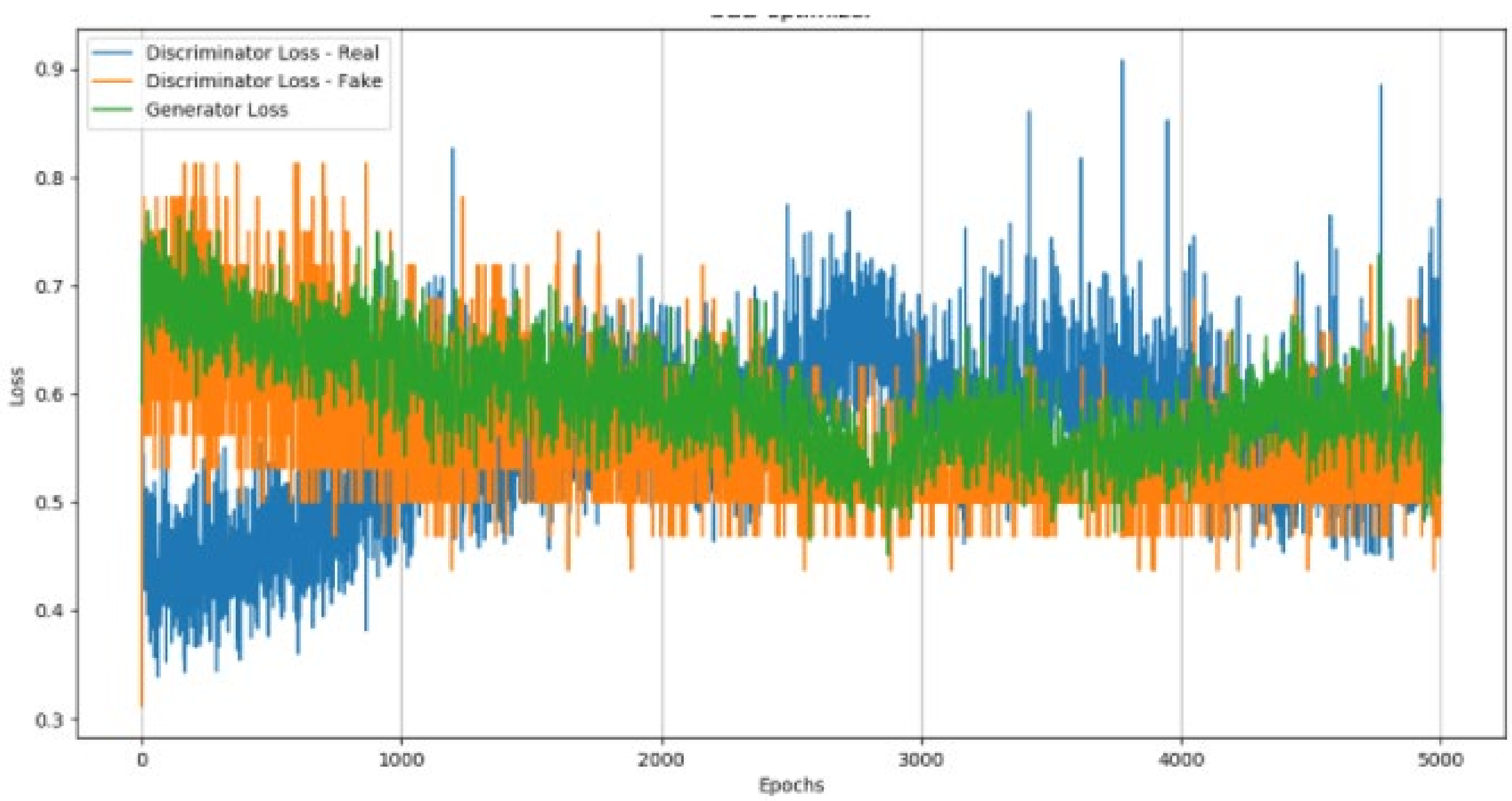
(a) **Poorly fit model**

(b) **After updating D**

(c) **After updating G**

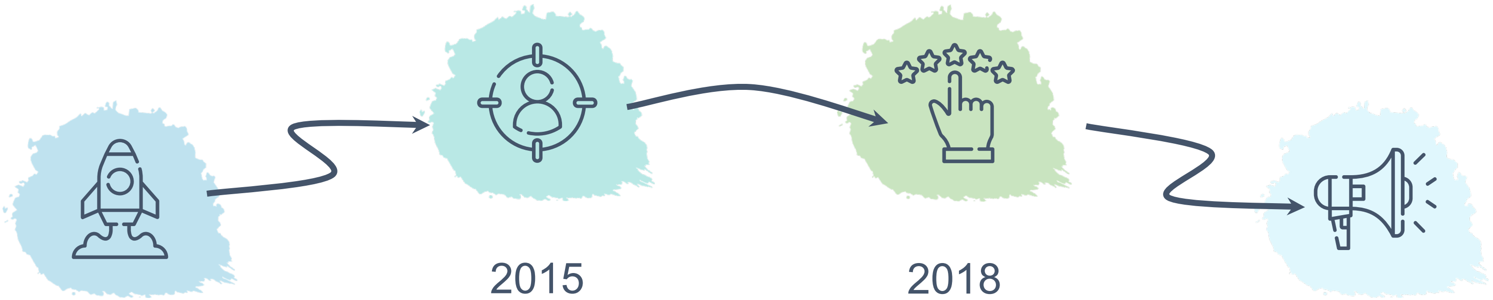
(d) **Mixed strategy equilibrium**

Training plot



My First Experience

GANs History



2014
GAN

Goodfellow, et al

2015
DCGAN

Radford, et al

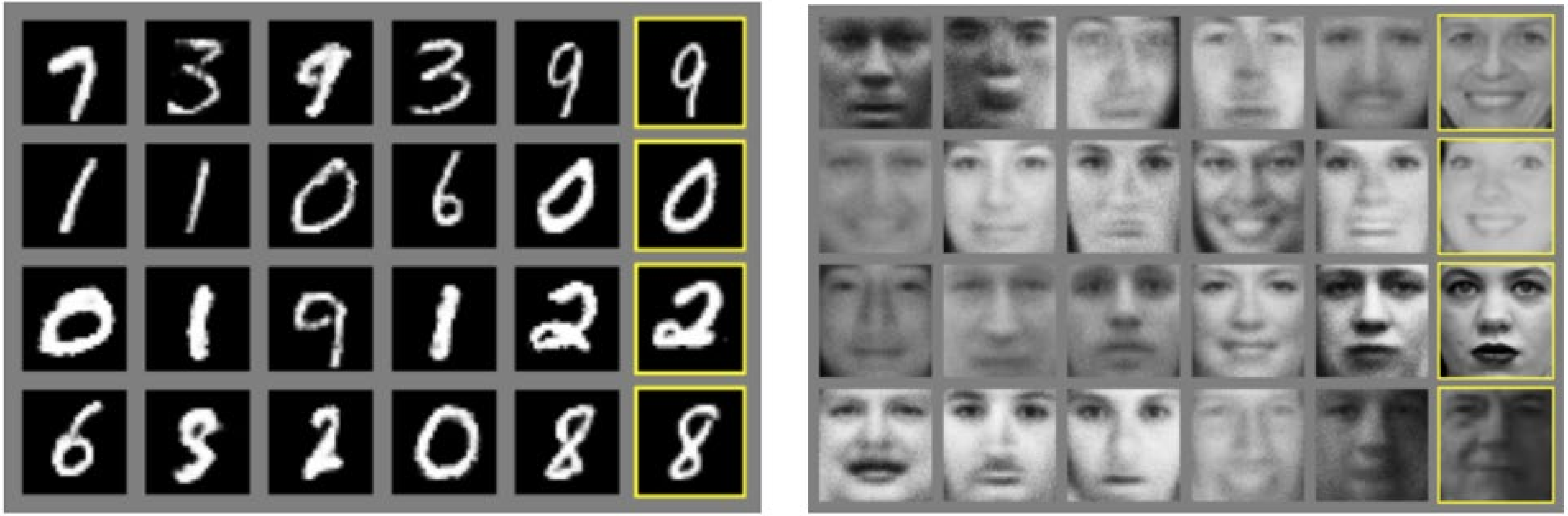
2018
BigGAN

Brock, et al

2018
StyleGAN

Karras, et al

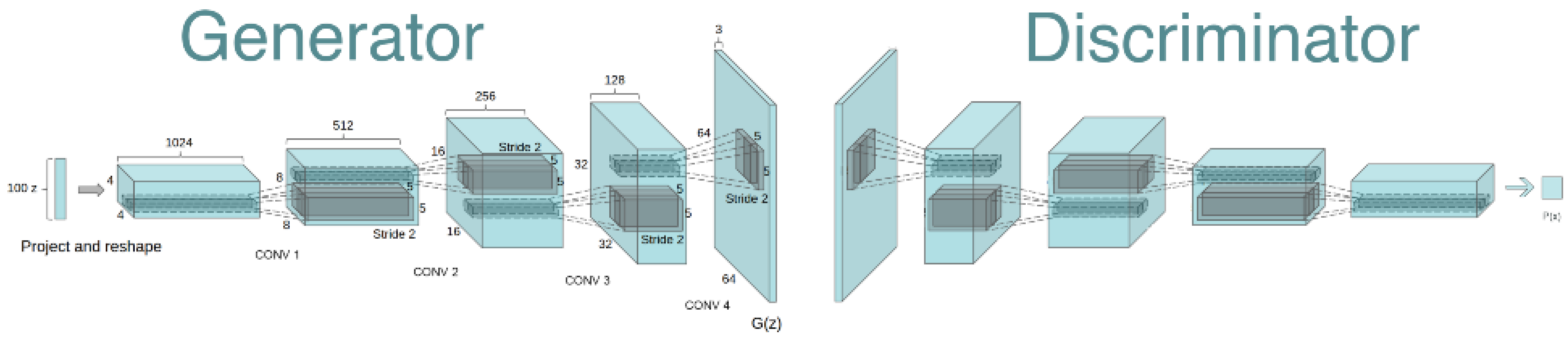
GAN



Yellow boxes are real data samples that are nearest matches to last column of fake images. This shows the generator didn't merely memorize training examples

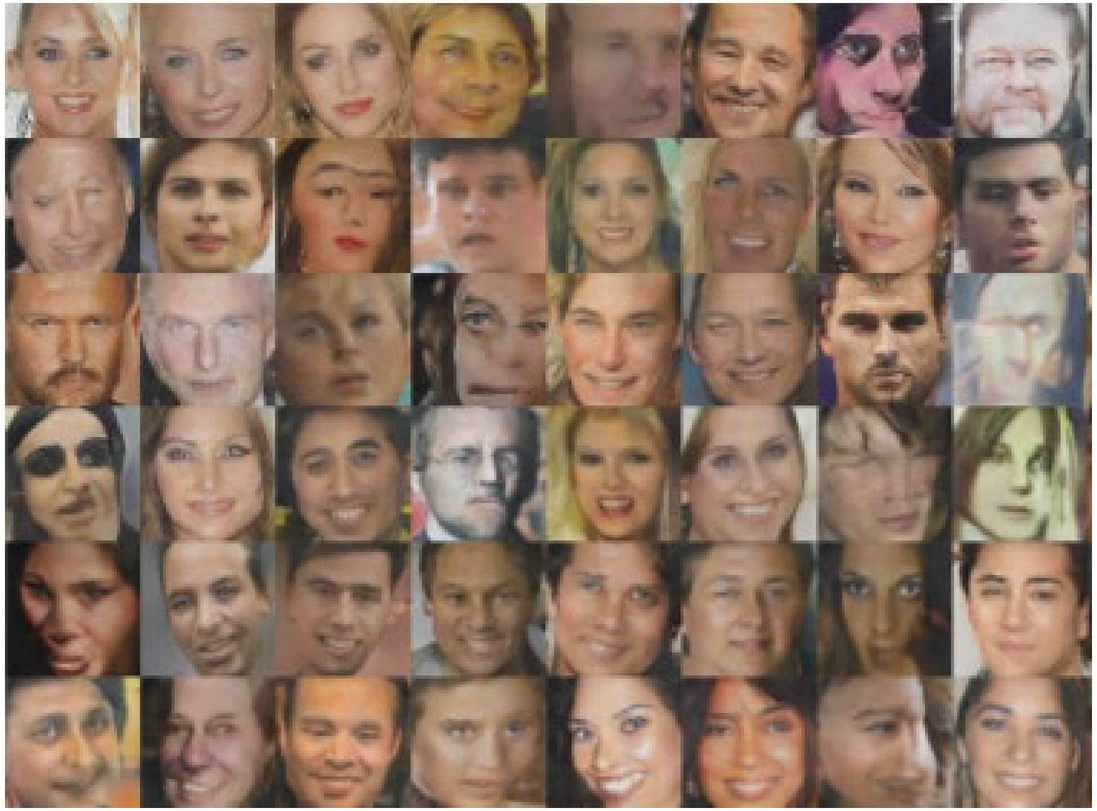
GAN

DCGAN



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

DCGAN



BigGAN

Figure 1: Class-conditional samples generated by our model.

BigGAN

StyleGAN

Super Resolution

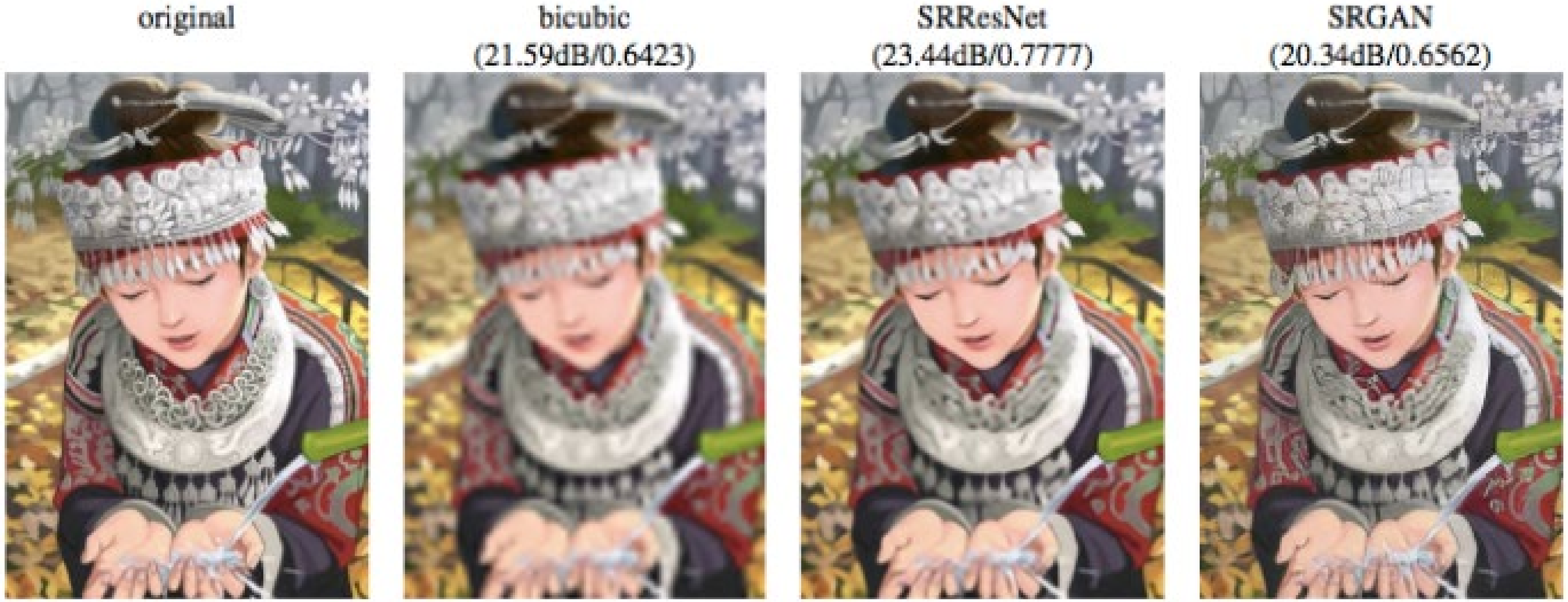
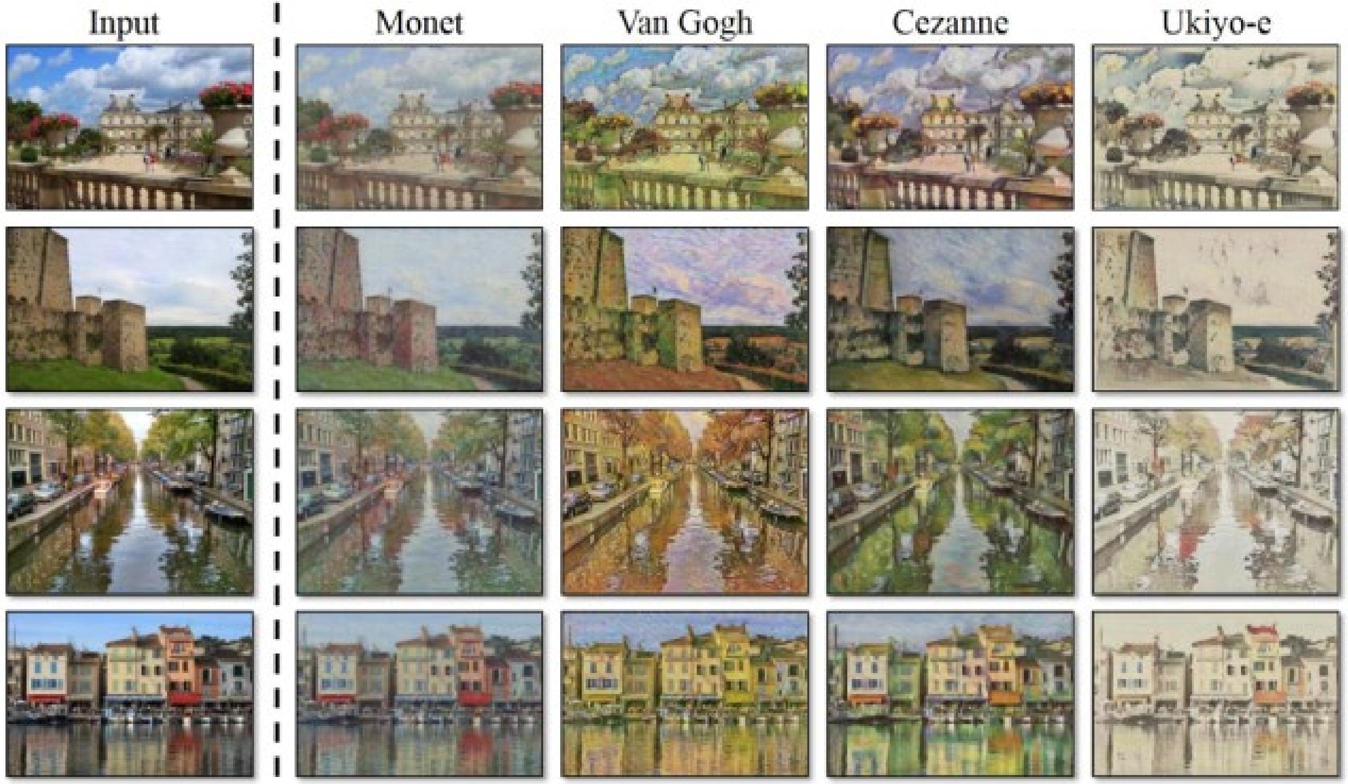


Image To Image Translation

More Image To Image(because it's amazing)



Applications Overview

Conditional Image Generation

SOTA results. Photorealistic image generation and many use-cases

Stability

Hard to train as the proccess is at times unstable and does not reach nash eqlibalerium

Creating Larger Datasets

Opens new opprotunities as data became critical resource in many open problems

Mode Collapse

Stuck in mode collapse – the Generator outputs the same image all the time. **Why?**

Auction

Edmond de Belamy Sold for $432,500

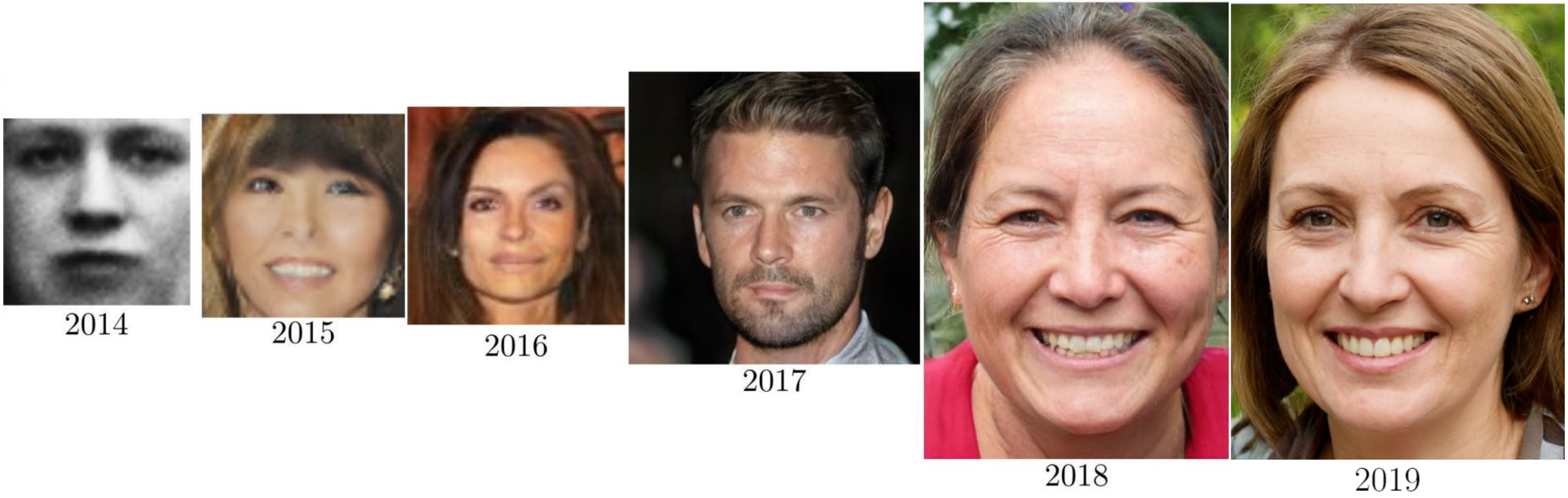
Why I chose to show you this?

05

Editing

Editing Images Using GANs

Generative Adversarial Networks (GANs)



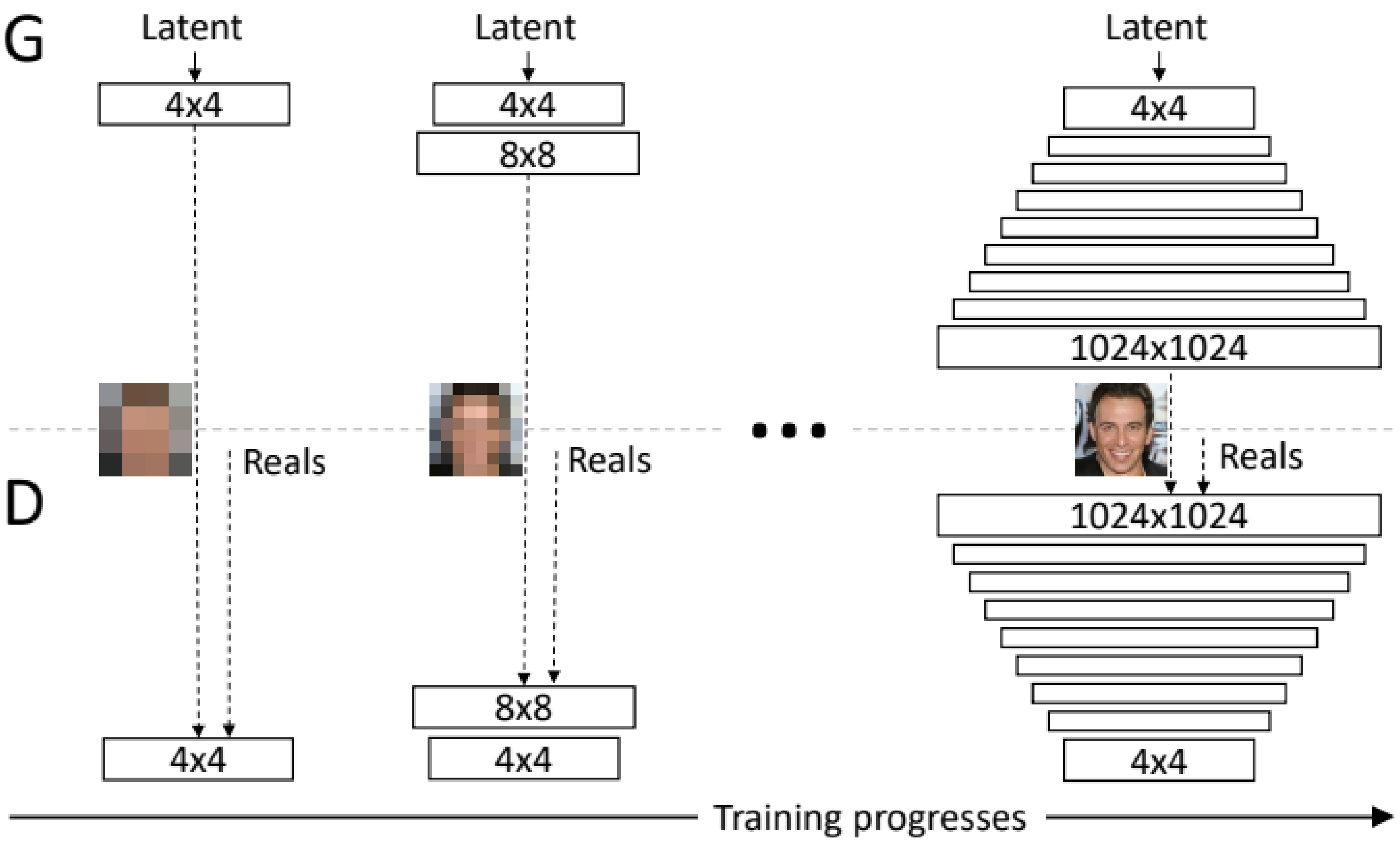
Random images

https://thispersondoesnotexist.com/

Random images

https://thesecatsdonotexist.com/

Baseline Progressive GAN



StyleGAN2-ADA (2020)

- ArXiv: https://arxiv.org/abs/2006.06676
- PyTorch implementation: https://github.com/NVlabs/stylegan2-ada-pytorch
- TensorFlow implementation: https://github.com/NVlabs/stylegan2-ada
- MetFaces dataset: https://github.com/NVlabs/metfaces-dataset

StyleGAN2 (2019)

- ArXiv: https://arxiv.org/abs/1912.04958
- Video: https://youtu.be/c-NJtV9Jvp0
- TensorFlow implementation: https://github.com/NVlabs/stylegan2

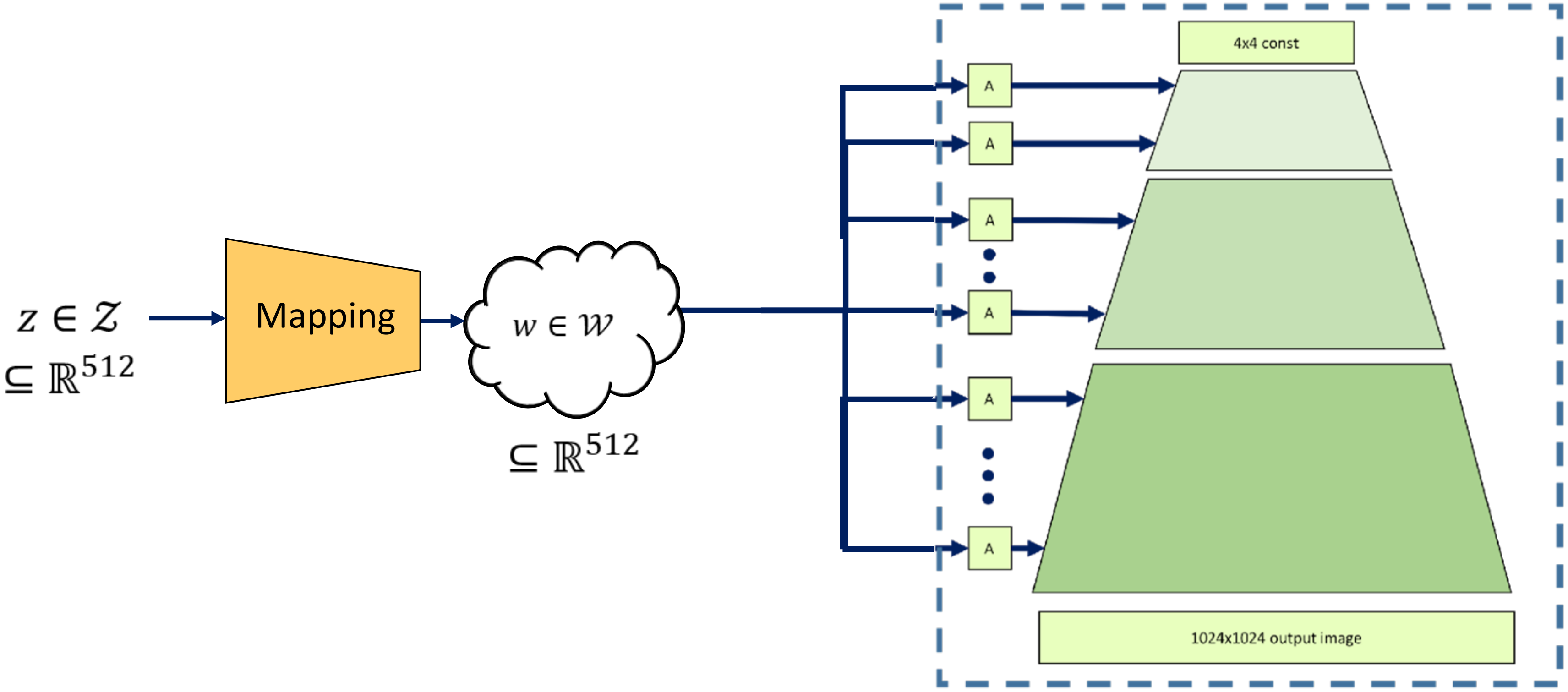
StyleGAN (2018)

- ArXiv: https://arxiv.org/abs/1812.04948
- Video: https://youtu.be/kSLJriaOumA
- TensorFlow implementation: https://github.com/NVlabs/stylegan
- FFHQ dataset: https://github.com/NVlabs/ffhq-dataset

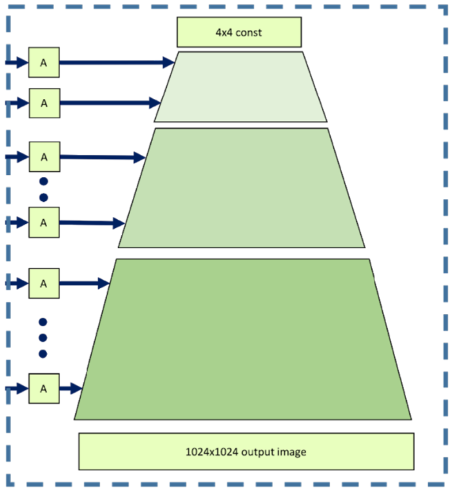
Progressive GAN (2017)

- ArXiv: https://arxiv.org/abs/1710.10196
- Video: https://youtu.be/G06dEcZ-QTg
- TensorFlow implementation: https://github.com/tkarras/progressive_growing_of_gans
- Theano implementation: https://github.com/tkarras/progressive_growing_of_gans/tree/original-theano-version
- CelebA-HQ dataset: https://github.com/tkarras/progressive_growing_of_gans#preparing-datasets-for-training

StyleGAN Architecture

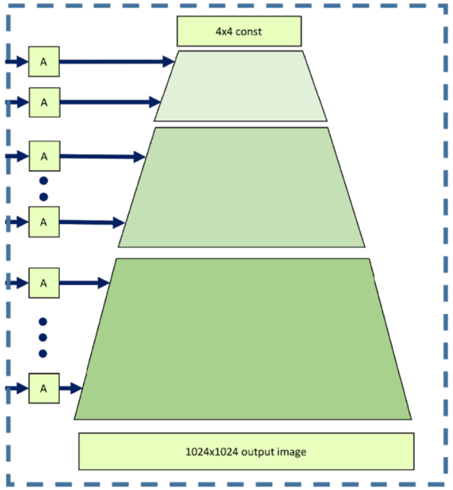


StyleGAN's style inputs correspond to different levels of details: coarse, medium, fine



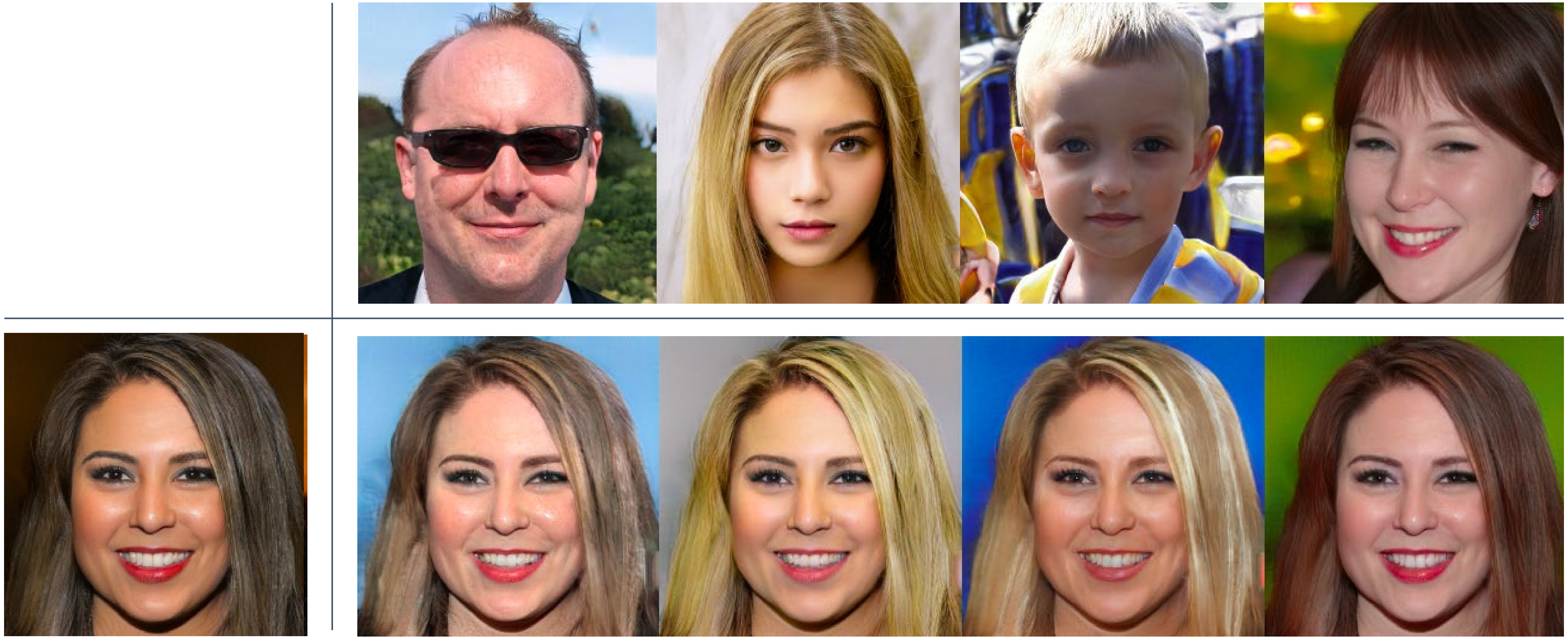
Control: pose, hair, face shape

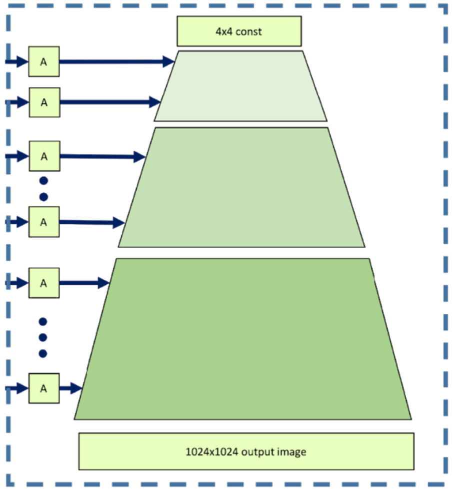
StyleGAN's style inputs correspond to different levels of details: coarse, medium, fine



Control: facial features, eyes

StyleGAN's style inputs correspond to different levels of details: coarse, medium, fine

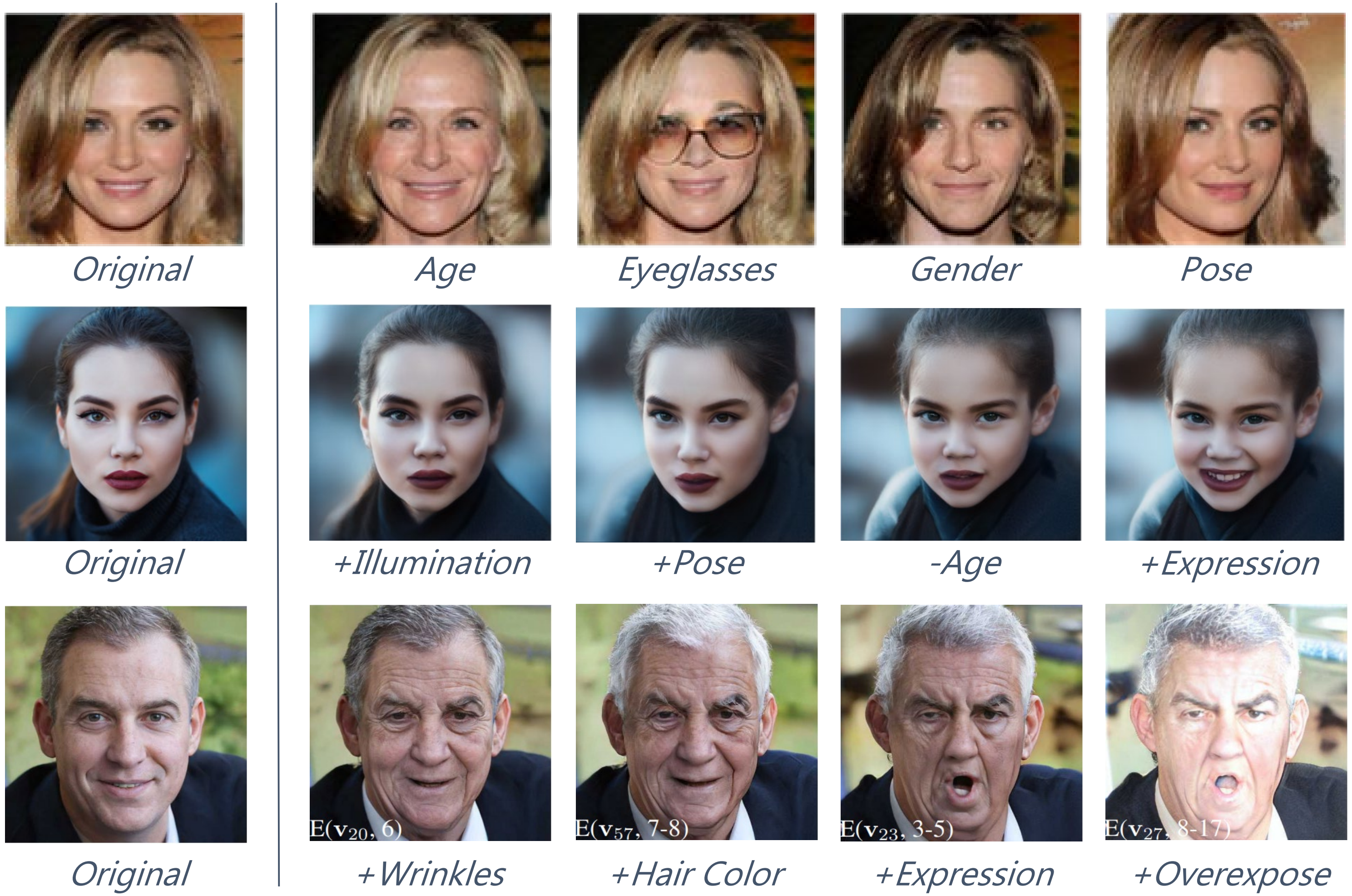




Control: lighting, color scheme

Not all is perfect

Why StyleGAN?



It has been show that the $\mathcal{W}$ latent space is <u>disentangled</u> and <u>offers editing capabilities</u>

InterFaceGAN (2019)
StyleFlow (2020)
GANSpace (2020)

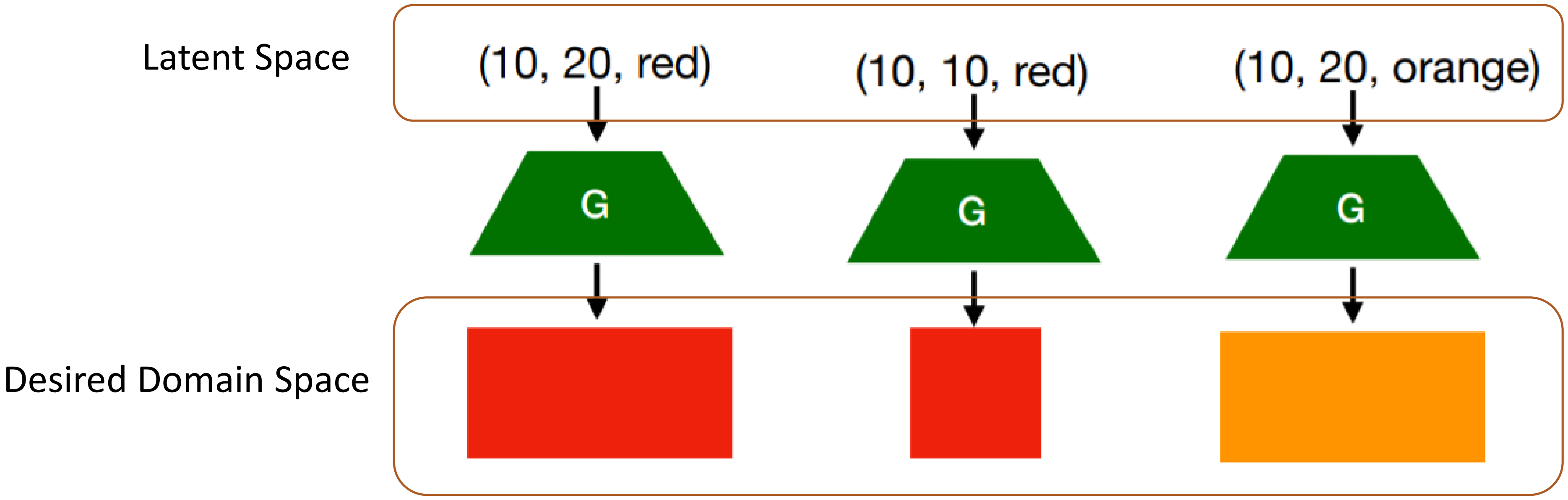
What is Disentanglement?

No formal definition which is widely accepted

Proposed definition: a change in a single underlying factor of variation in the sample x should lead to change in a single factor in the learned representation r(x)

Example - Rectangles

z = (height, width, color)

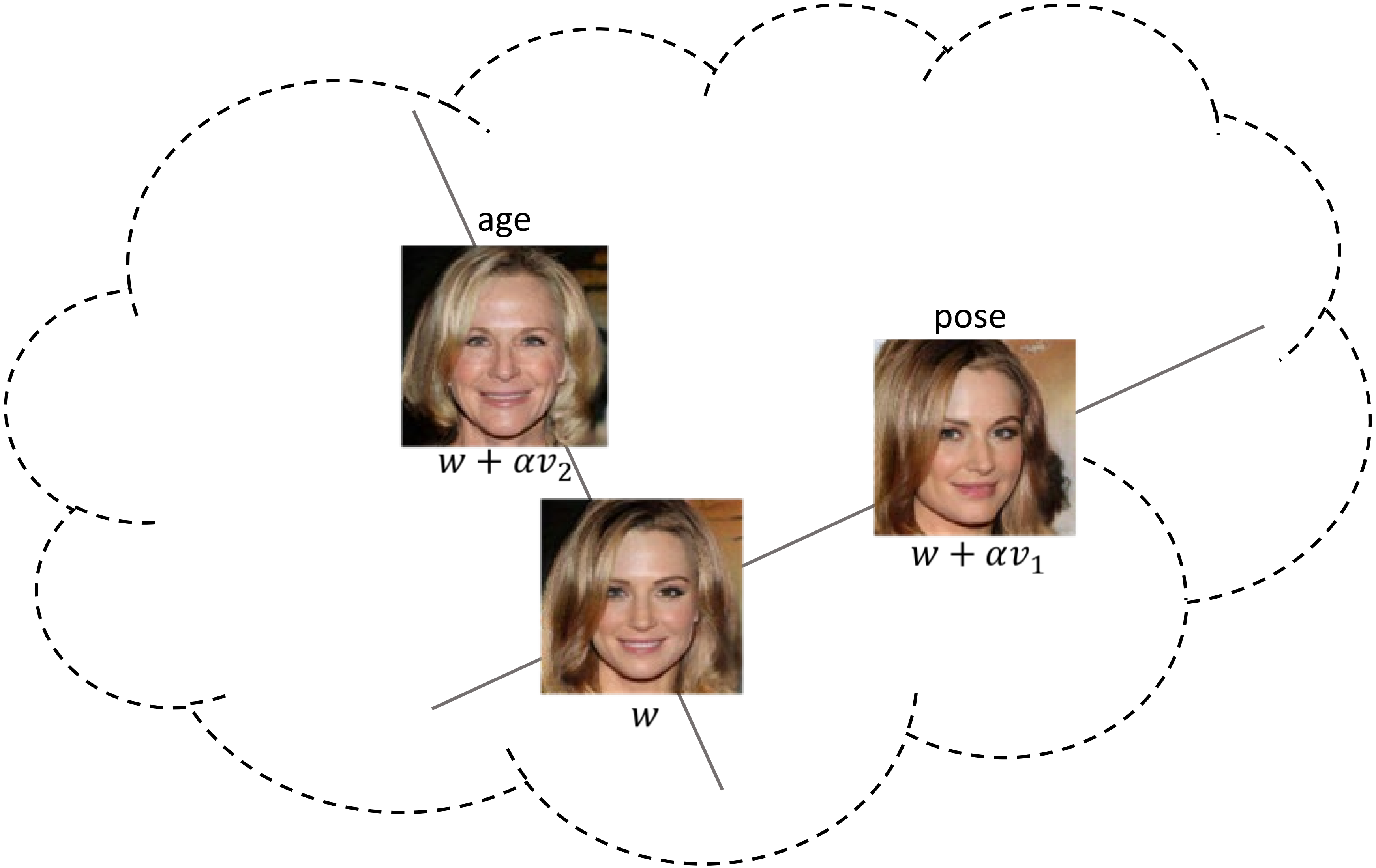



Photorealistic image generation
Disentangled space
High Quality Editing

Editing Applications

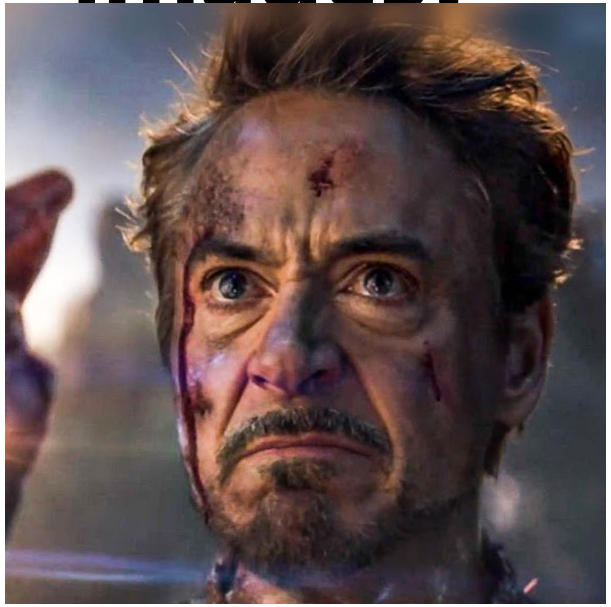
Creative content generation

Latent space editing



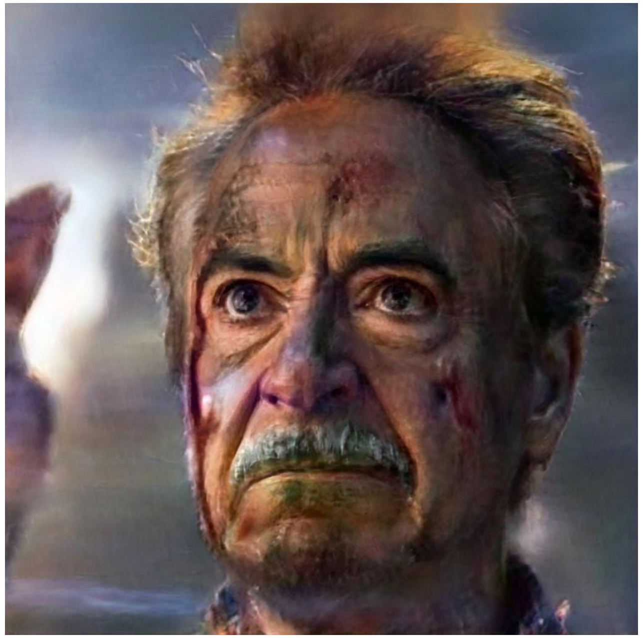
Can we perform <u>meaningful</u> <u>manipulations</u> on <u>Real</u> <u>images?</u>

Can we perform meaningful manipulations on Real images?



$I \in R^{3\times256\times256}$

End Goal



$I \in R^{3\times256\times256}$

$$I \xrightarrow{\text{Inversion}} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$$

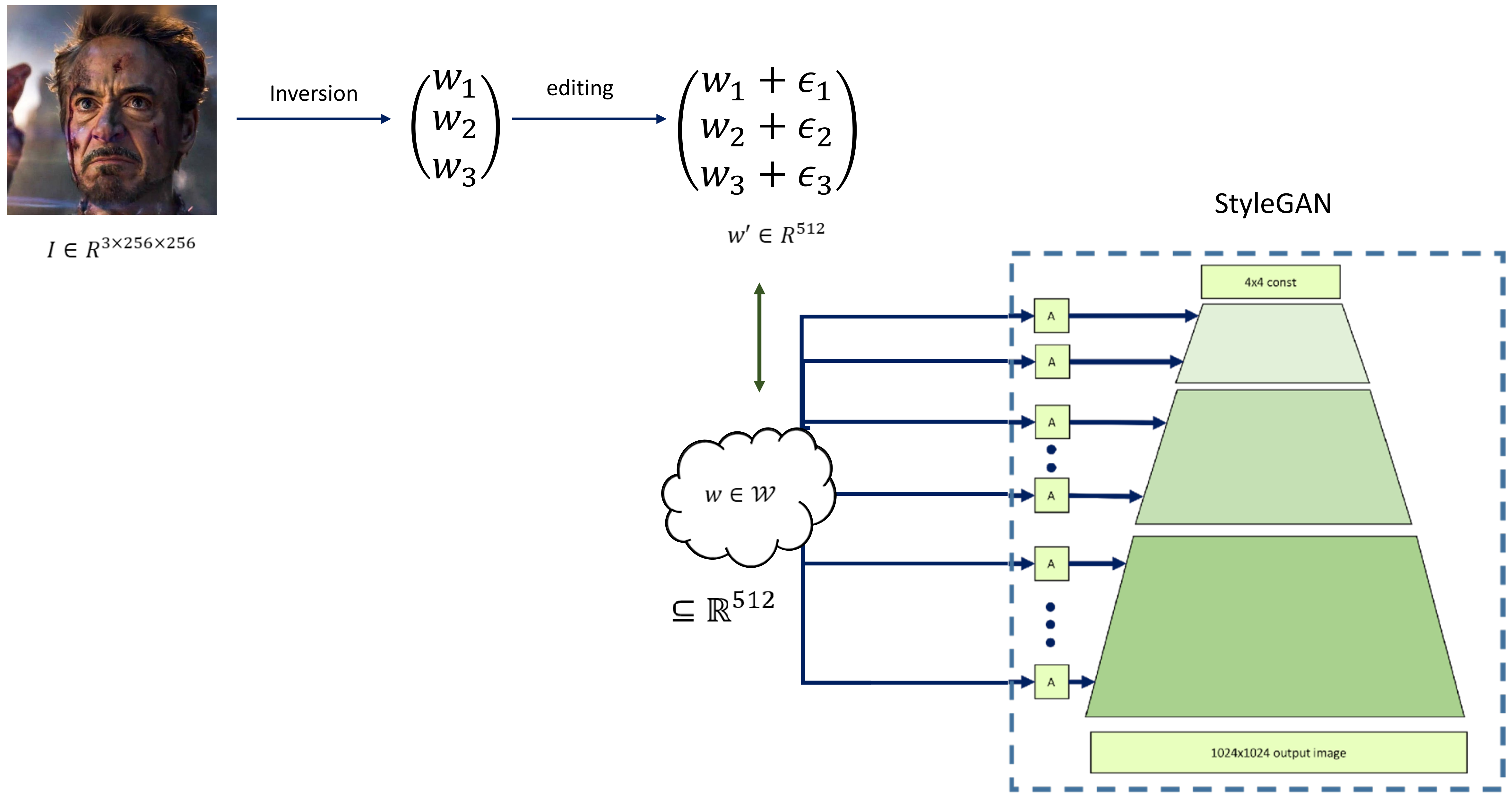
$I \in R^{3\times256\times256}$

$$\xrightarrow{\text{Inversion}} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} \xrightarrow{\text{editing}} \begin{pmatrix} w_1 + \epsilon_1 \\ w_2 + \epsilon_2 \\ w_3 + \epsilon_3 \end{pmatrix}$$

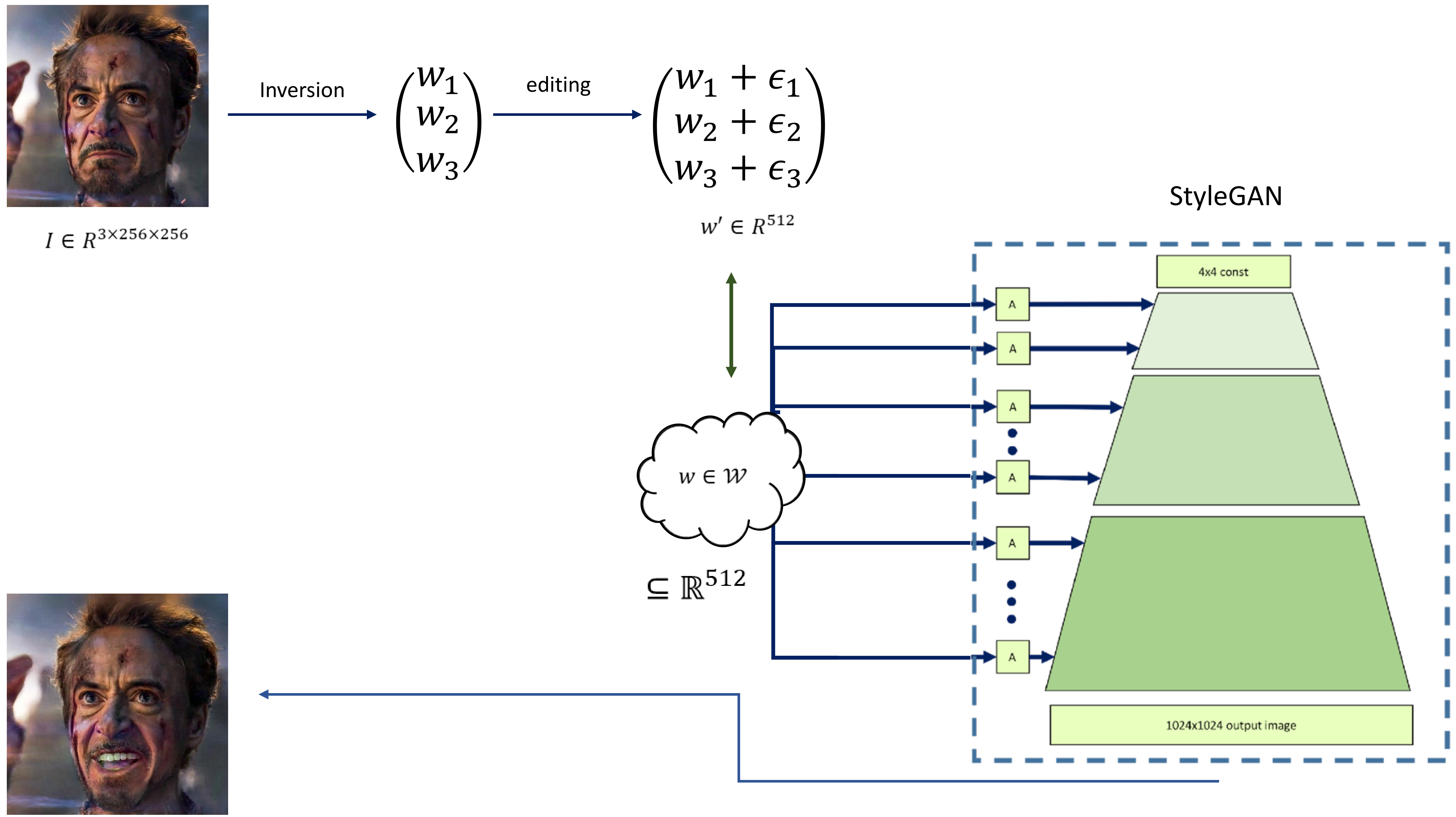
$w' \in R^{512}$

$$\begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{pmatrix}$$

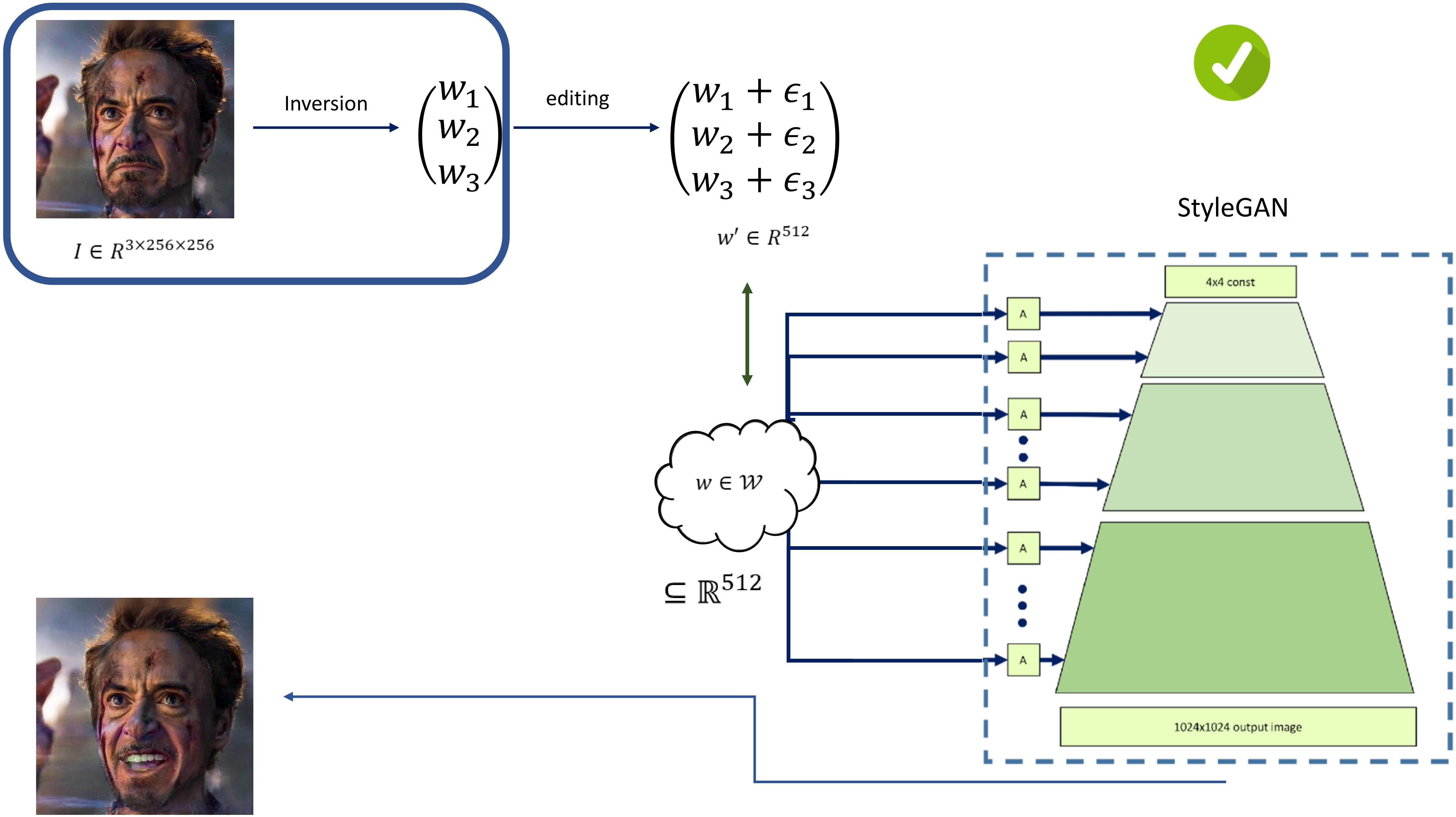
Learned smile direction



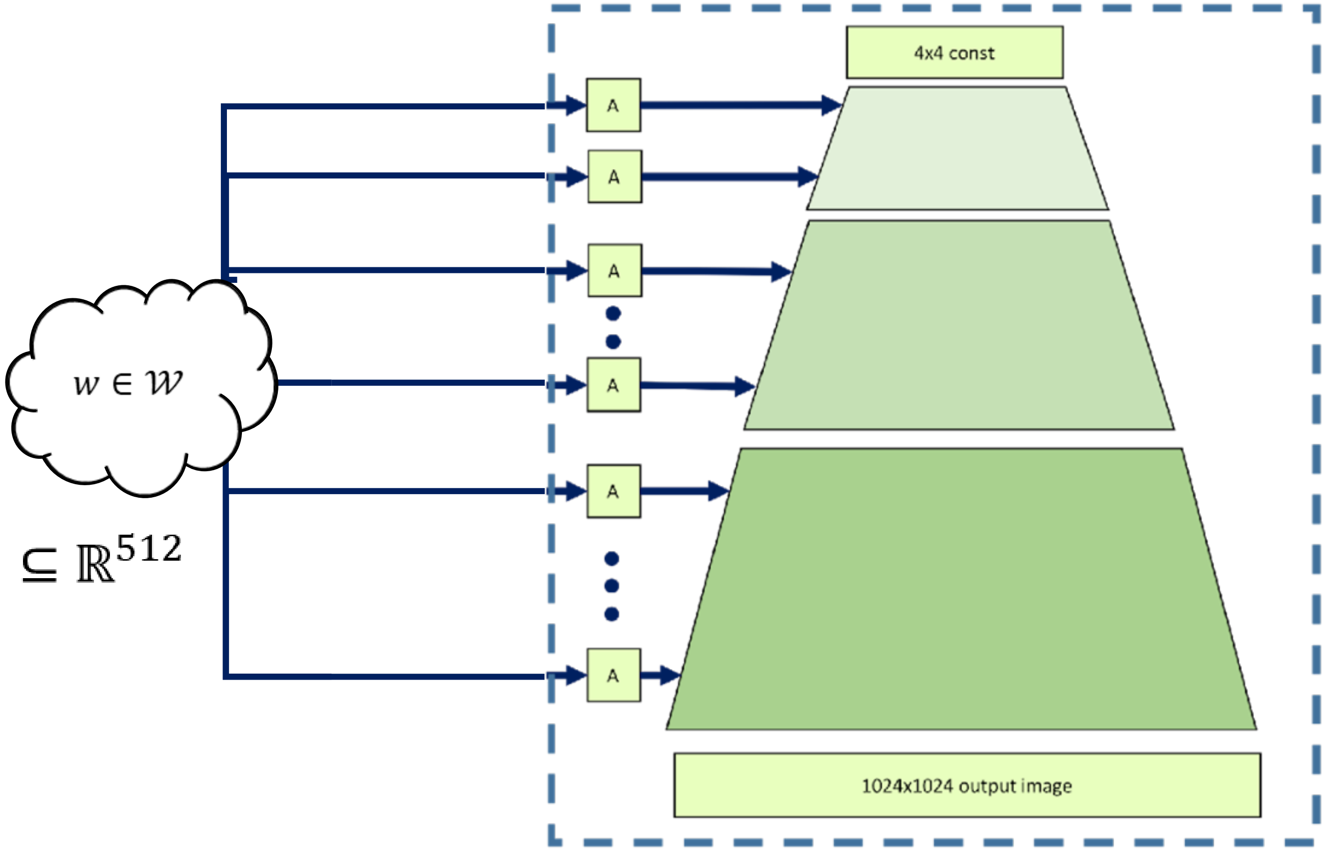
Inversion
editing
$I \in R^{3\times256\times256}$
$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$
$\begin{pmatrix} w_1 + \epsilon_1 \\ w_2 + \epsilon_2 \\ w_3 + \epsilon_3 \end{pmatrix}$
$w' \in R^{512}$
StyleGAN
4x4 const
A
$w \in \mathcal{W}$
$\subseteq \mathbb{R}^{512}$
1024x1024 output image



Inversion
$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$
editing
$\begin{pmatrix} w_1 + \epsilon_1 \\ w_2 + \epsilon_2 \\ w_3 + \epsilon_3 \end{pmatrix}$
$I \in R^{3\times256\times256}$
$w' \in R^{512}$
StyleGAN
4x4 const
A
$w \in \mathcal{W}$
$\subseteq \mathbb{R}^{512}$
1024x1024 output image


Inversion
$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$
editing
$\begin{pmatrix} w_1 + \epsilon_1 \\ w_2 + \epsilon_2 \\ w_3 + \epsilon_3 \end{pmatrix}$
$I \in R^{3\times256\times256}$
$w' \in R^{512}$
StyleGAN
4x4 const
A
A
A
A
A
A
1024x1024 output image
$w \in \mathcal{W}$
$\subseteq \mathbb{R}^{512}$

Current Latent Space



Extend The Latent Space

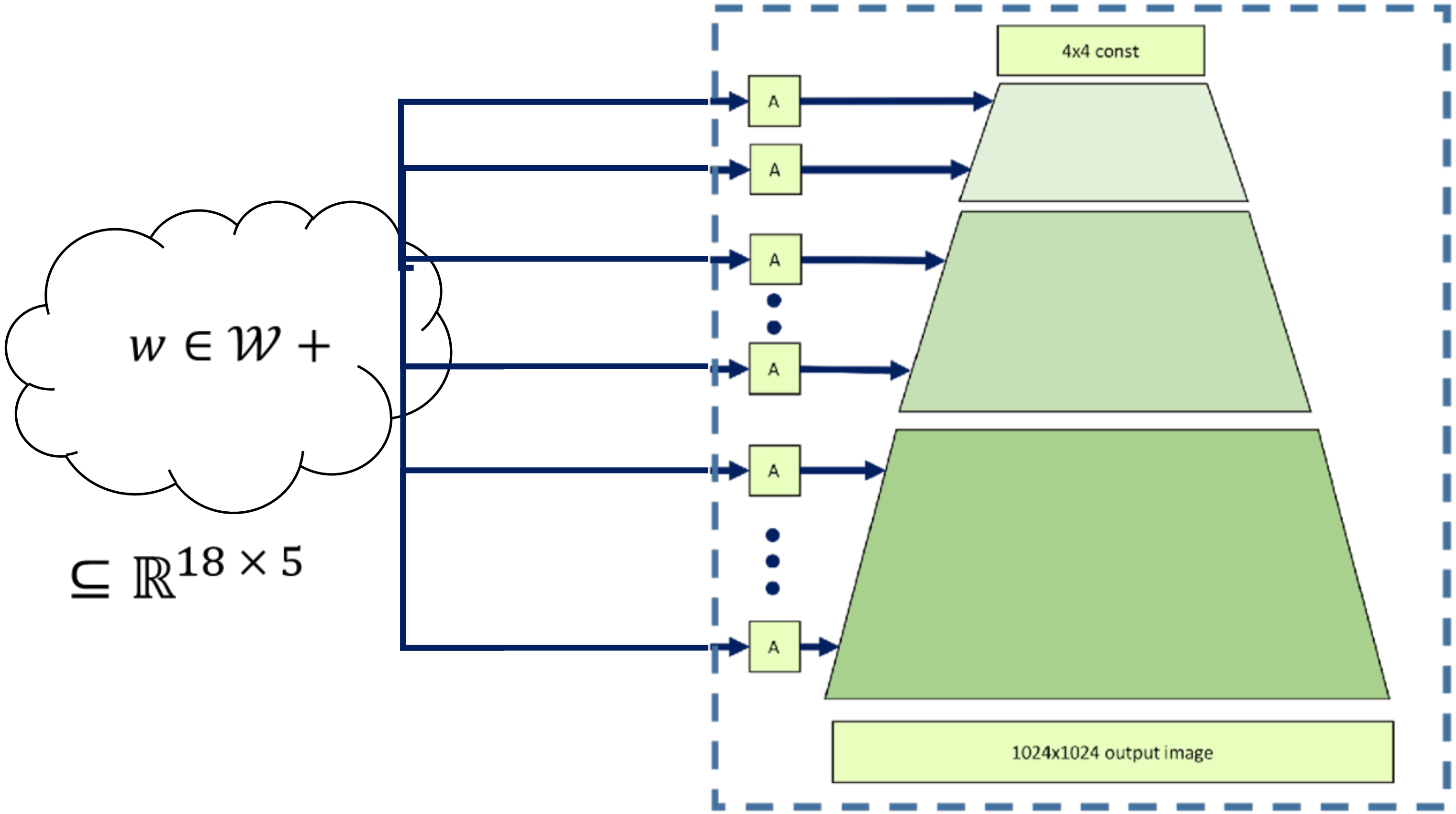


Image2StyleGAN (2020)

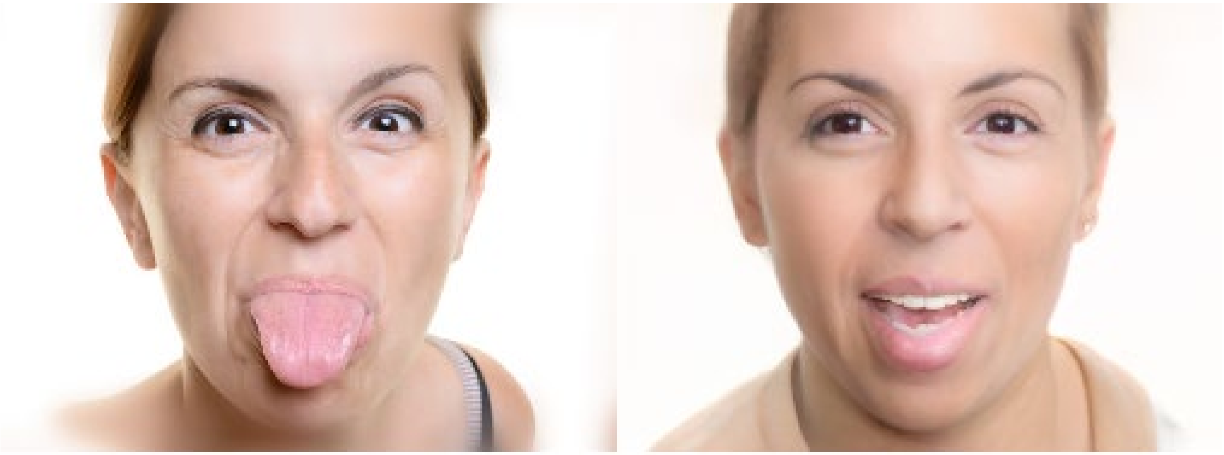
Extend The Latent Space

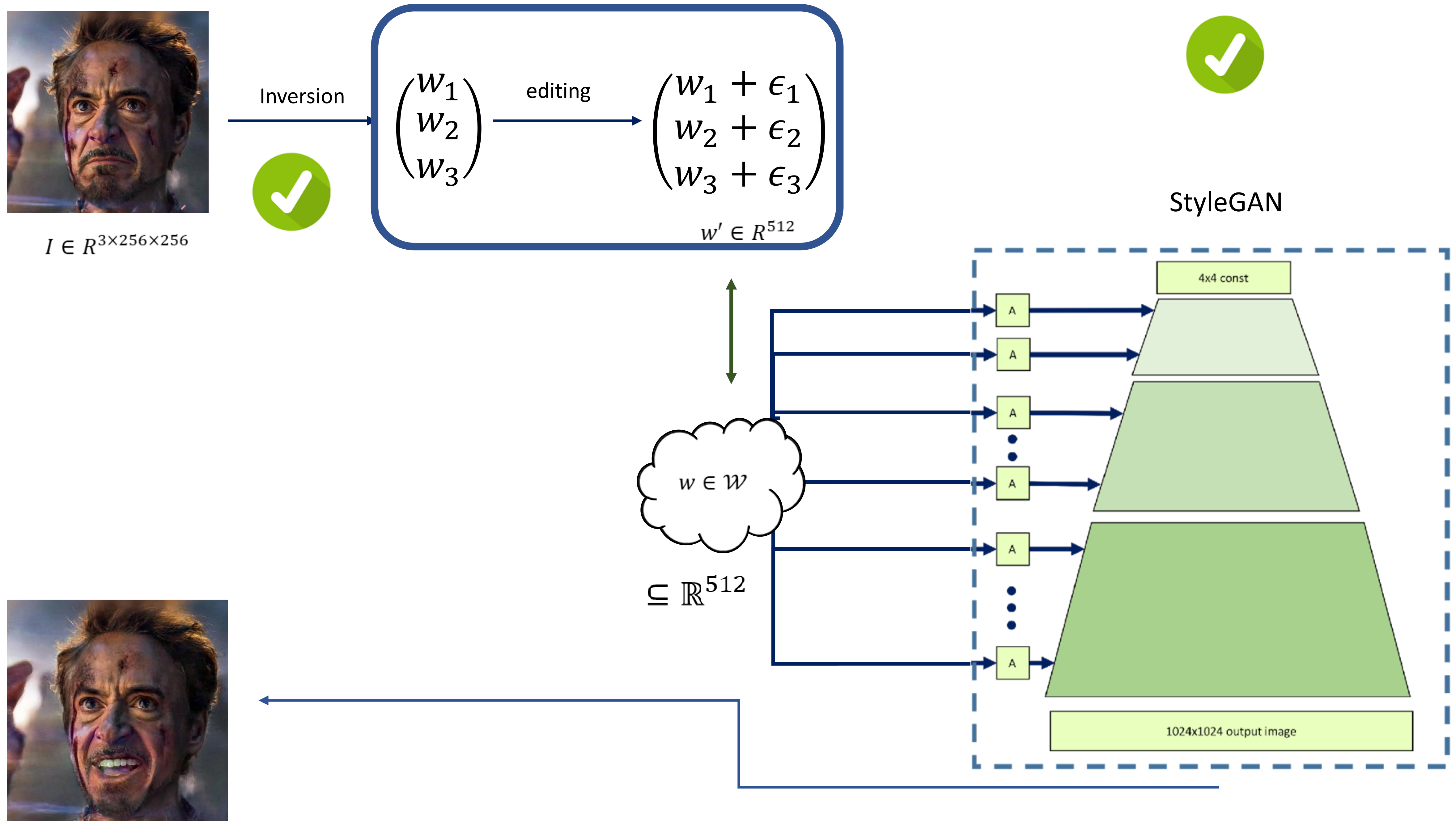
$w \in \mathcal{W}+$
(18×512)

$w \in \mathcal{W}+$ $w \in \mathcal{W}$ source

Image2StyleGAN (2020)

Inversion Drawbacks

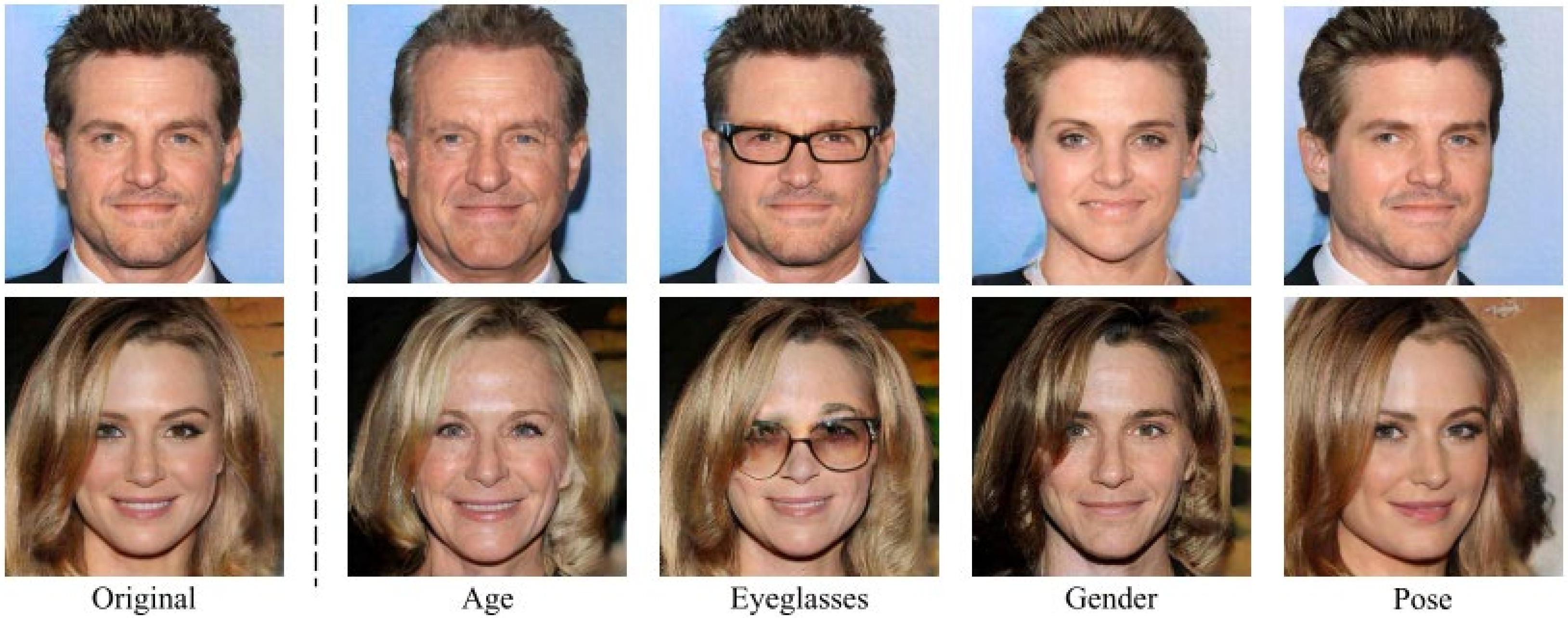




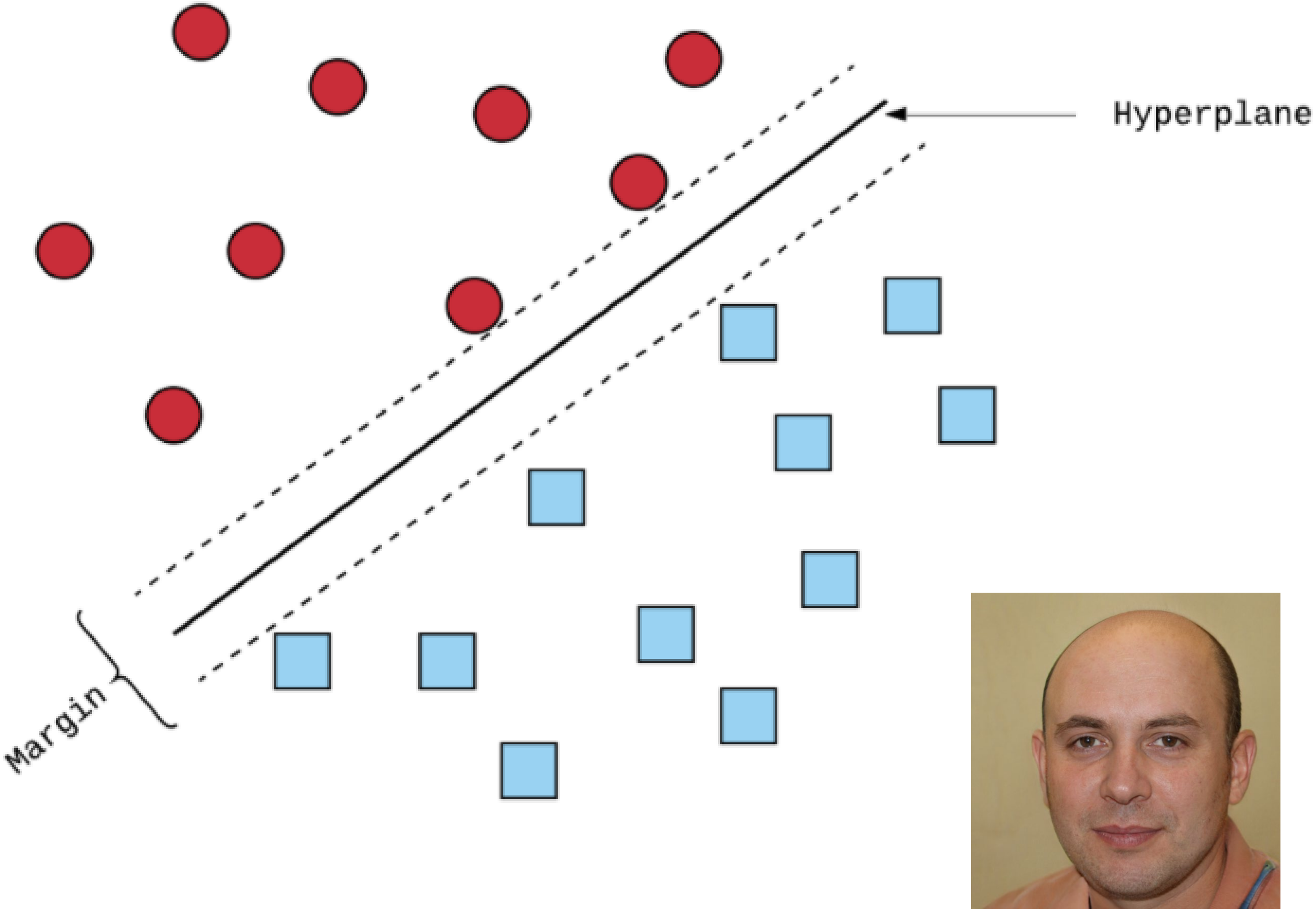
$I \in R^{3\times256\times256}$
Inversion
$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$
editing
$\begin{pmatrix} w_1 + \epsilon_1 \\ w_2 + \epsilon_2 \\ w_3 + \epsilon_3 \end{pmatrix}$
$w' \in R^{512}$
StyleGAN
4x4 const
A
A
A
A
A
A
$w \in \mathcal{W}$
$\subseteq \mathbb{R}^{512}$
1024x1024 output image

InterFaceGAN
Interpreting the Latent Space of GANs for Semantic Face Editing (2020)
Yujun Shen, et al

Demo

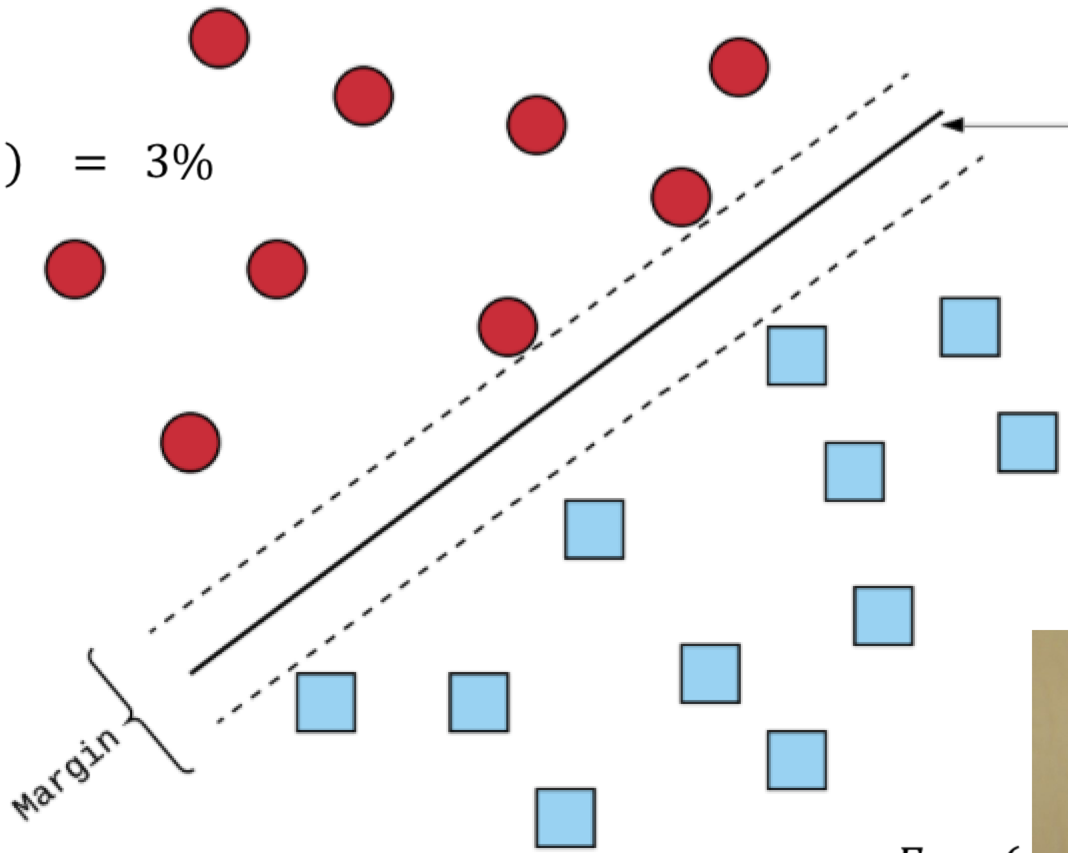


Assumptions



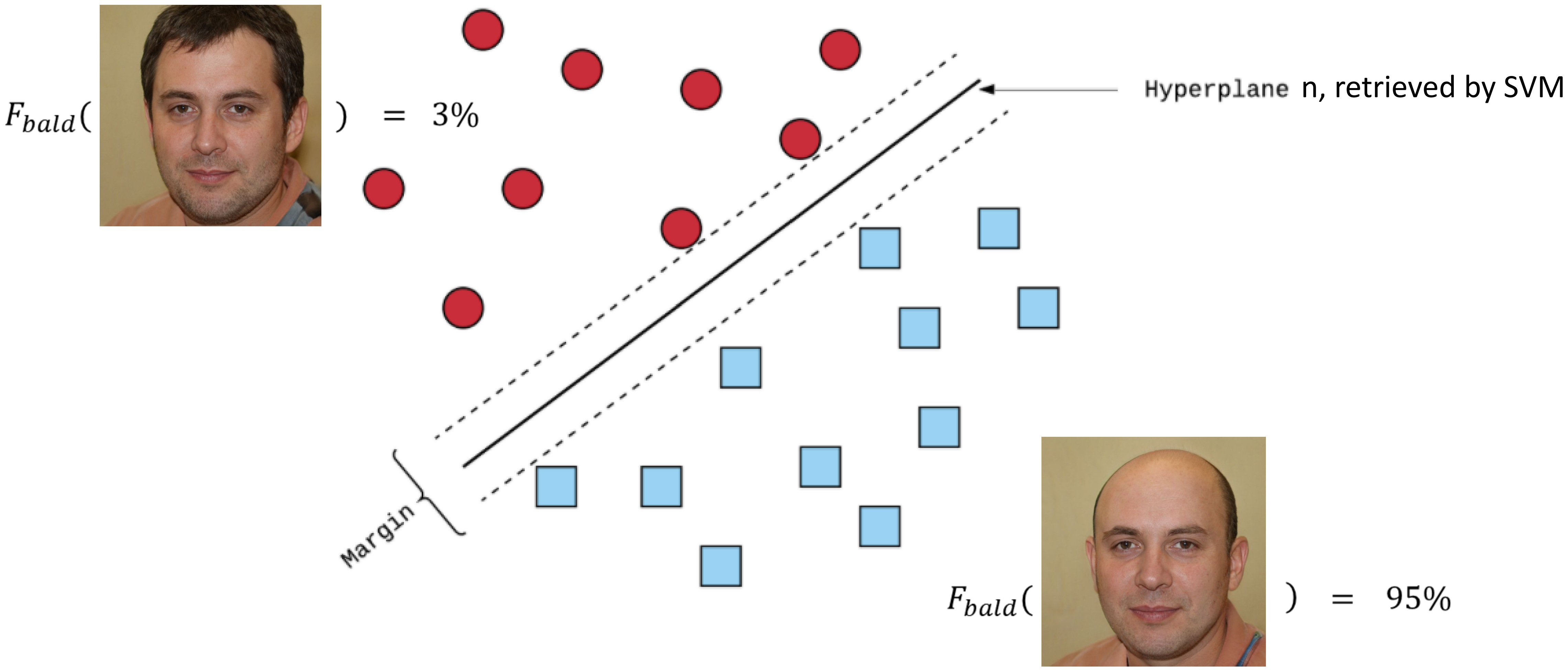
Method

$F_{bald}($  $) = 3\%$

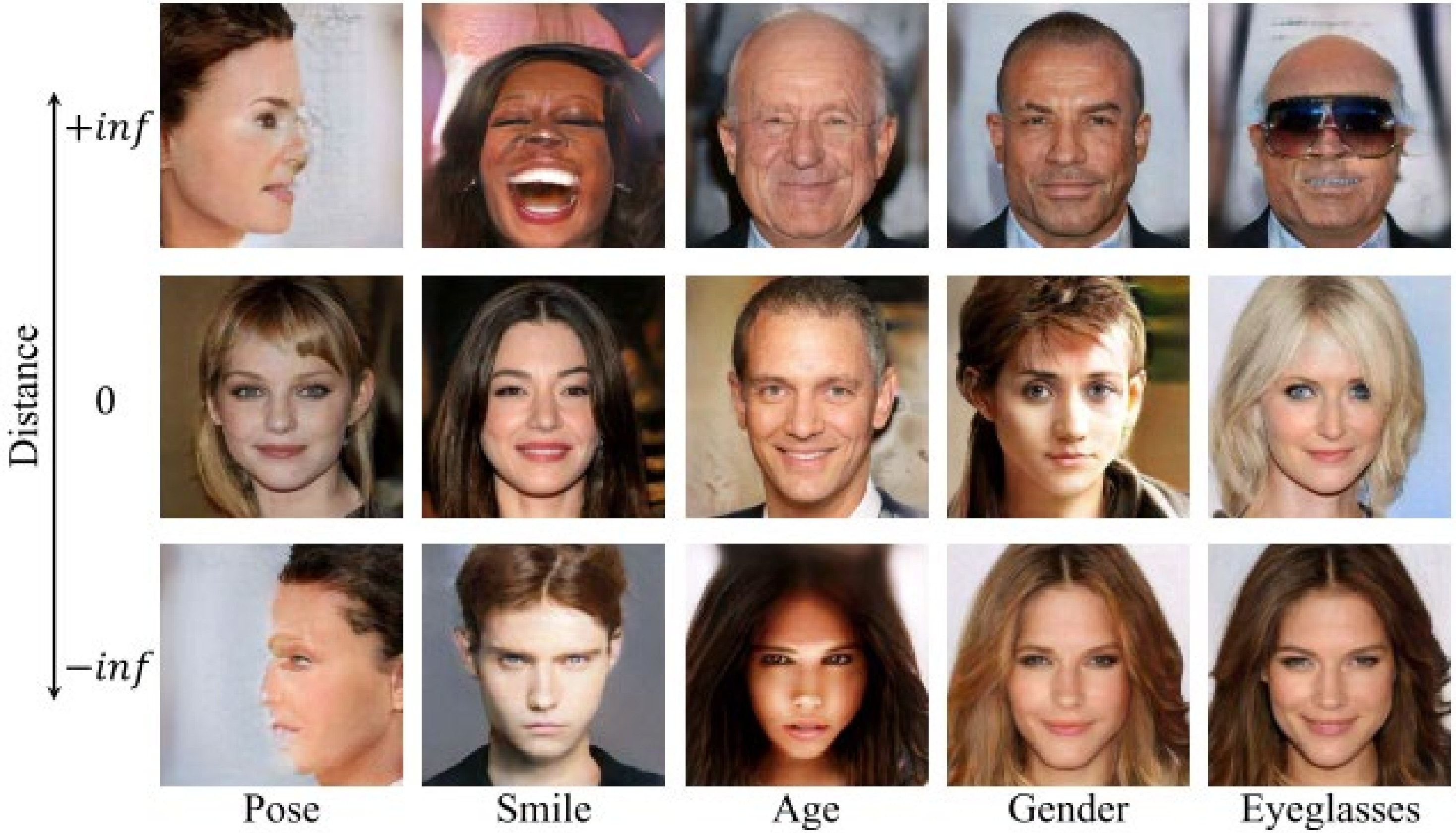


$F_{bald}($  $) = 95\%$

Method

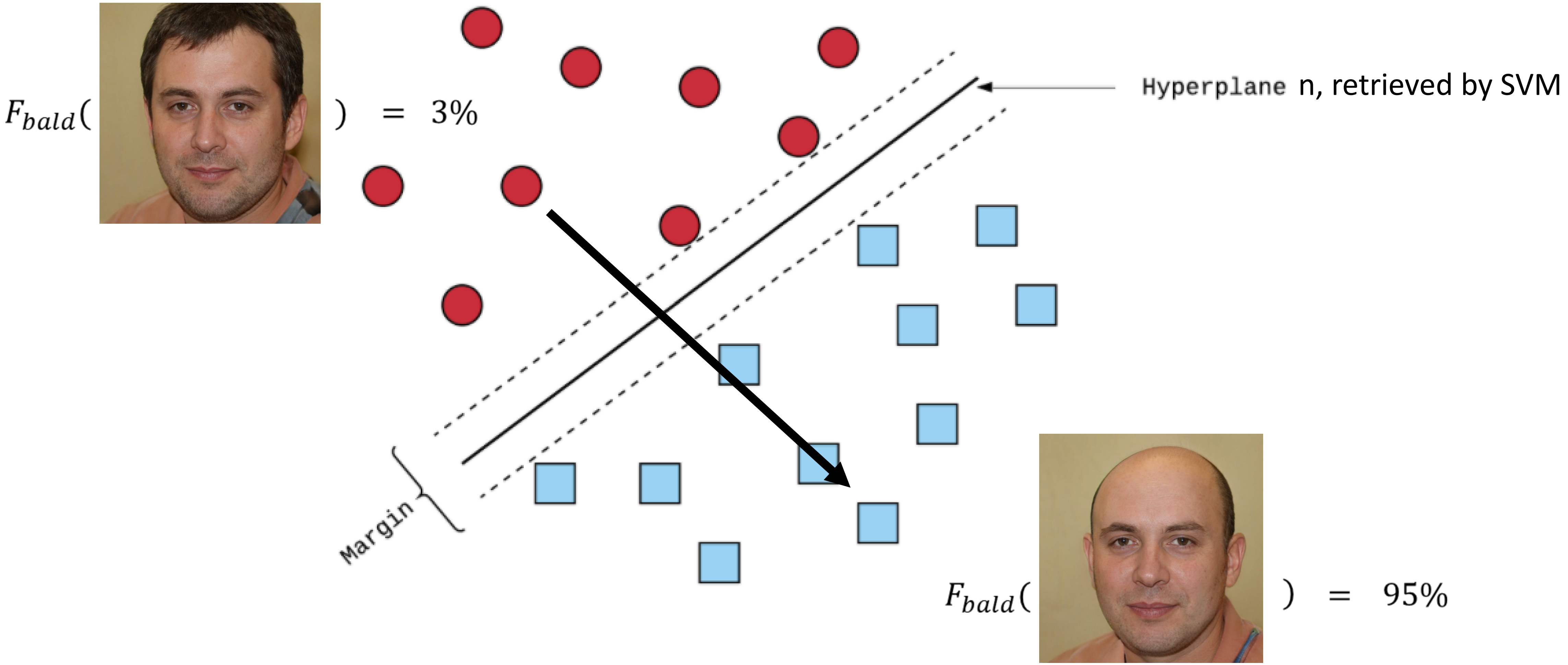


Assumptions Visualization



Method

$$w_1 \rightarrow w_1 + \alpha n = w_2$$

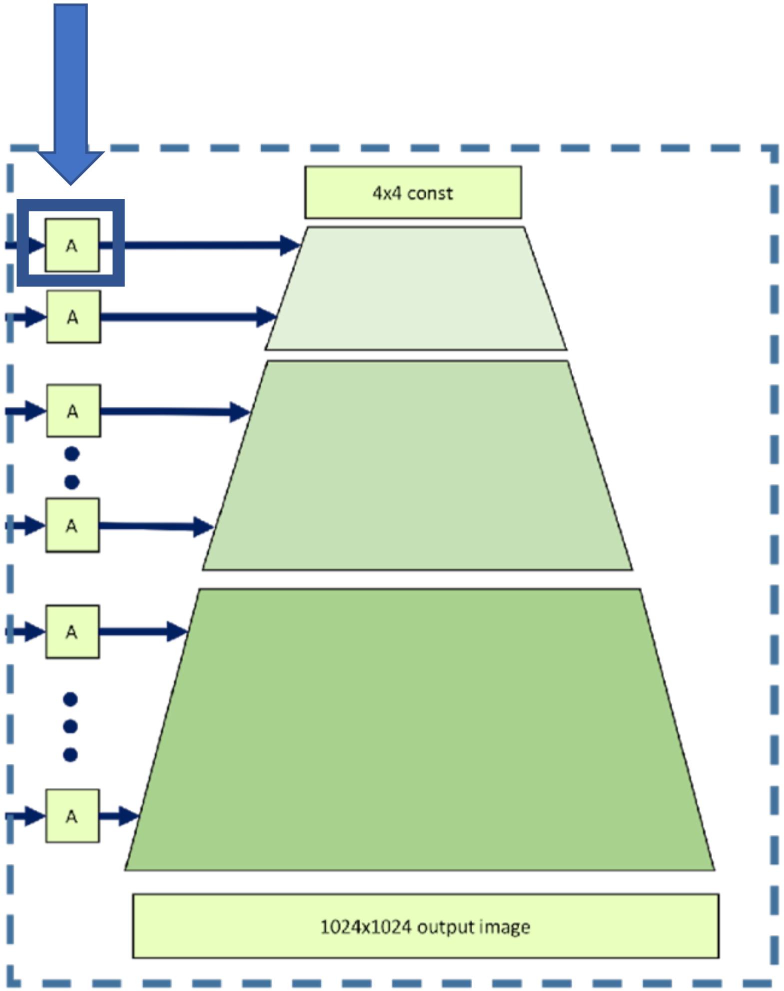


SeFa - Closed-Form Factorization of Latent Semantics in GANs (2020)

Yujun Shen, Bolei Zhou

Demo

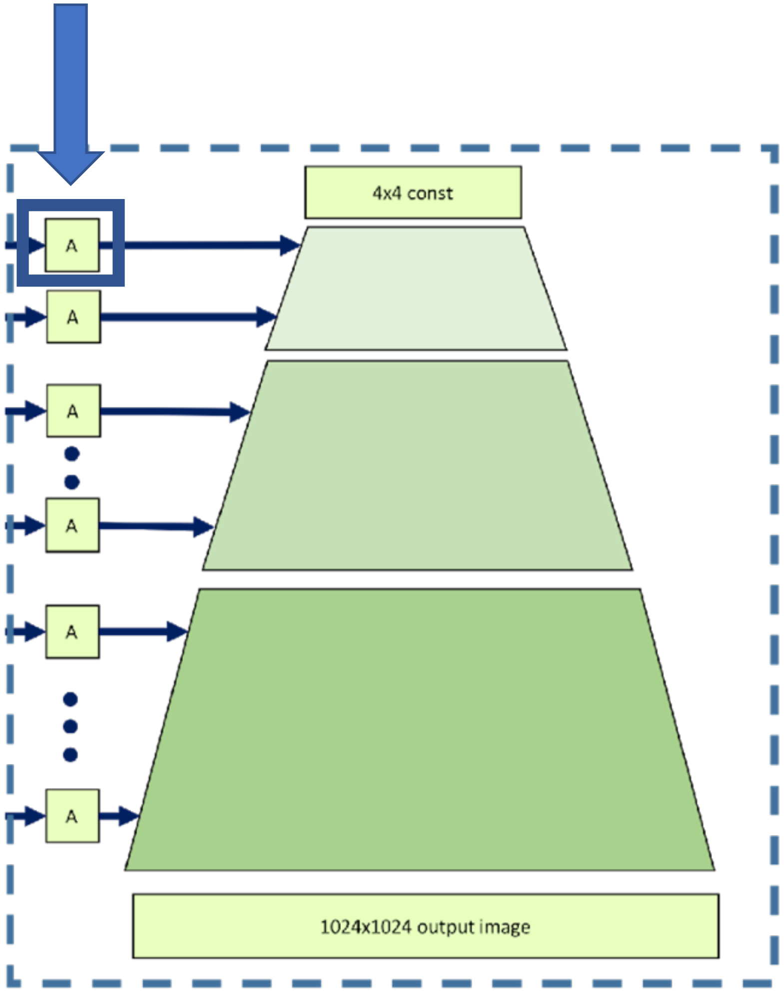
Method



$$A(w) = Aw + b$$

$$A(w') = A(w + an) = Aw + b + \alpha An$$

Method



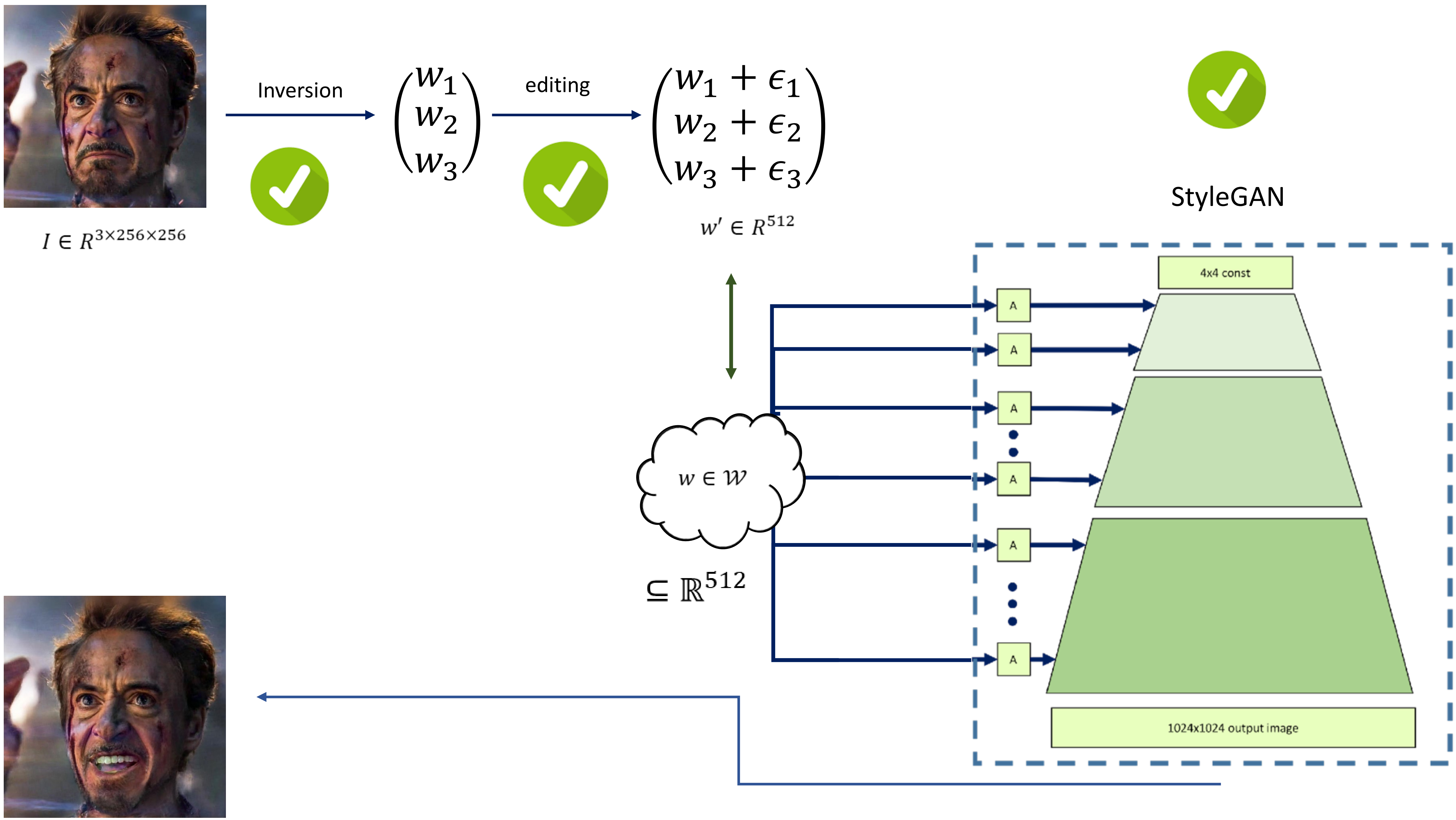
$$A(w) = Aw + b$$

$$A(w') = A(w + an) = Aw + b + \alpha An$$

$$n^* = argmax_{\{n \in \mathbb{R}^d : n^T n = 1\}} ||An||_2^2$$

SeFa advantage over InterFaceGAN

InterFaceGAN advantage over SeFa



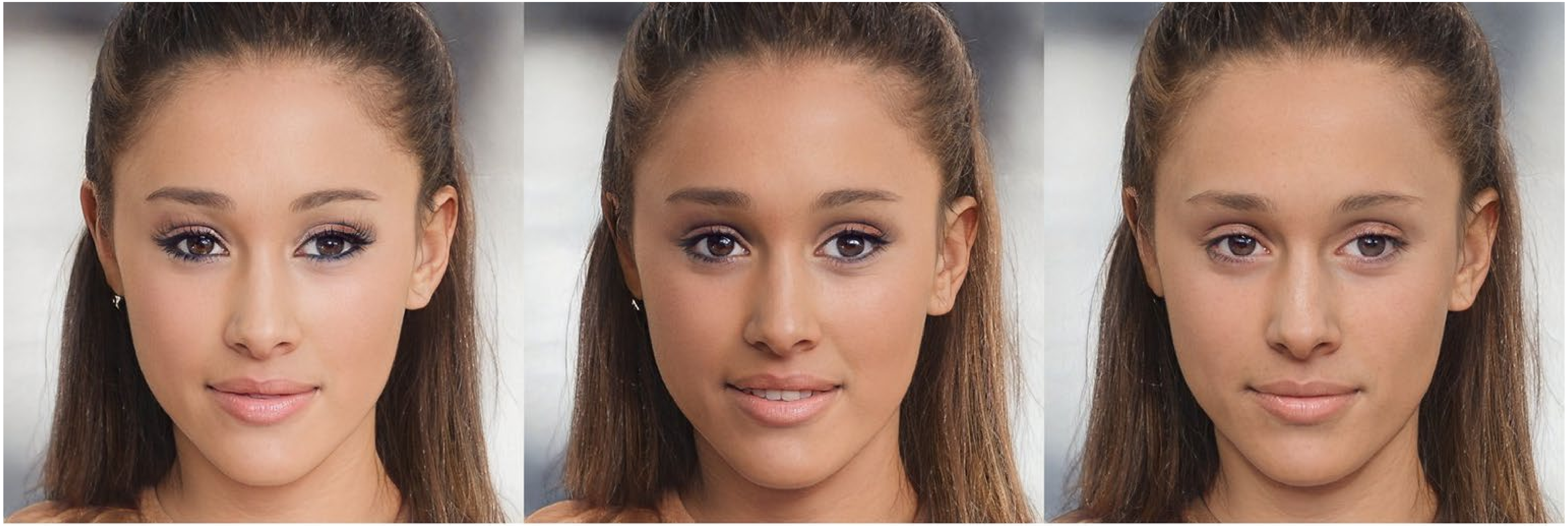
$I \in R^{3\times256\times256}$
Inversion
$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$
editing
$\begin{pmatrix} w_1 + \epsilon_1 \\ w_2 + \epsilon_2 \\ w_3 + \epsilon_3 \end{pmatrix}$
$w' \in R^{512}$
StyleGAN
4x4 const
A
$w \in \mathcal{W}$
$\subseteq \mathbb{R}^{512}$
1024x1024 output image

Play at home

https://github.com/danielroich/PTI

https://colab.research.google.com/github/danielroich/PTI/blob/main/notebooks/inference_playground.ipynb

Text driven image manipulation



Inversion | A tanned woman | A woman without makeup

Text driven image manipulation

Inversion | A man with a beard | A blonde man

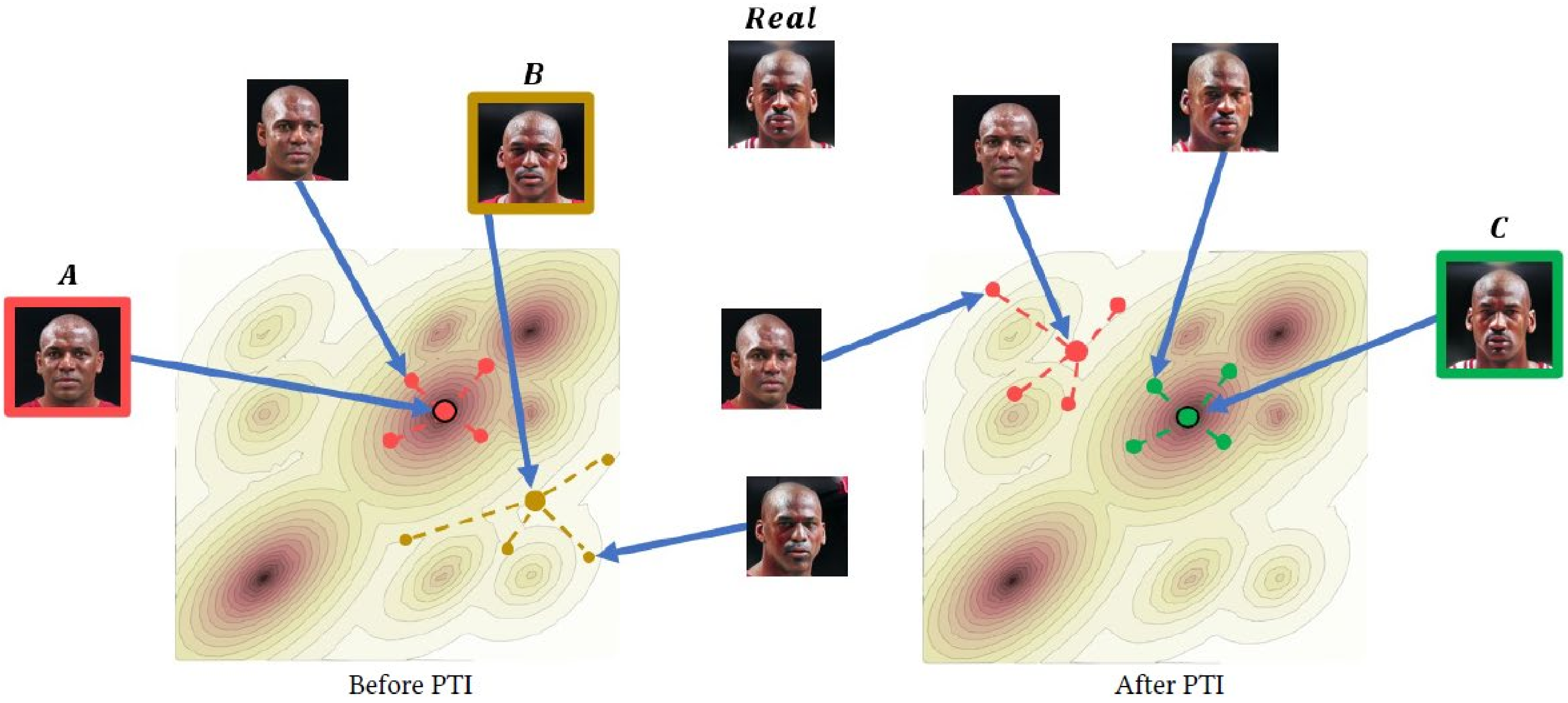
Editing

Editing

Editing

Pivotal Tuning for Latent-based editing of Real Images

PTI Idea



Results

Results

NN Based Graphics Future