

On Generative Models

CS189/289A: Introduction to Machine Learning

Stella Yu

UC Berkeley

30 November 2017

Outline

1. Why generative models
2. Autoencoder (AE)
3. Variational Autoencoder (VAE)
4. Generative Adversarial Networks (GAN)

Why Generative Models

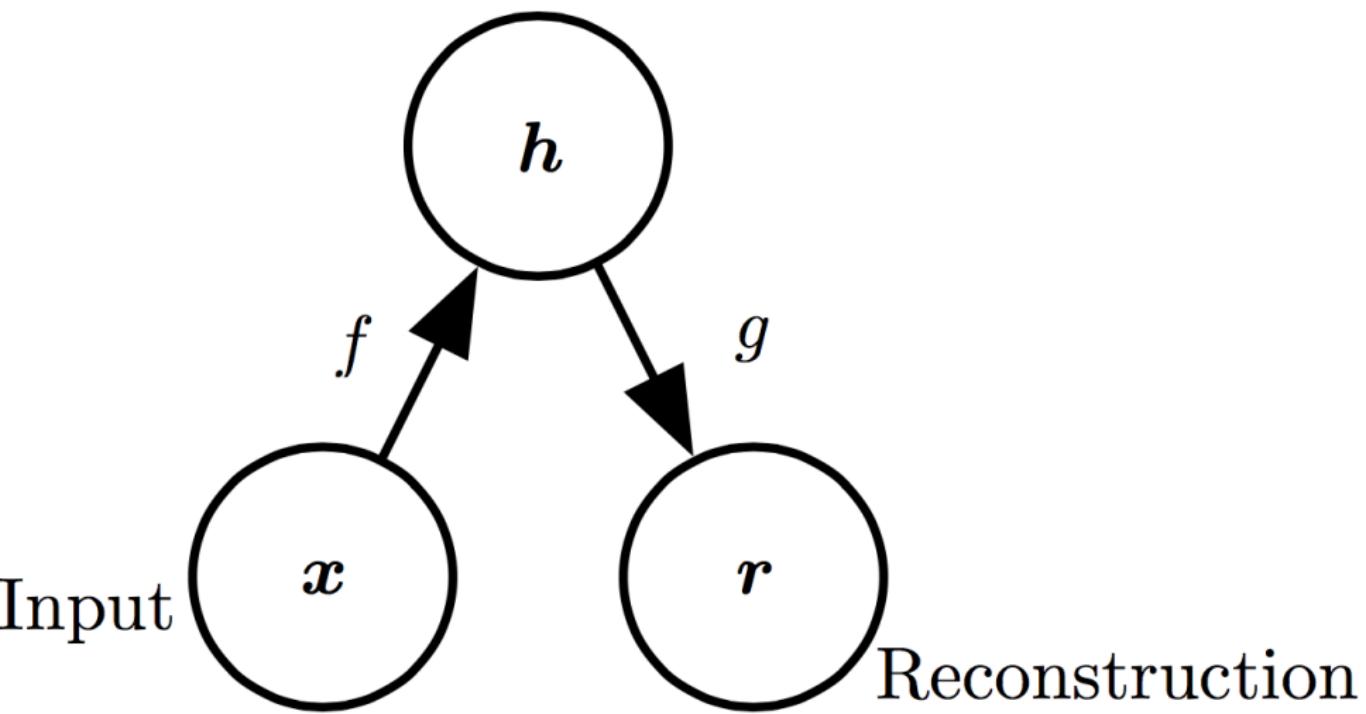
1. More of the kind:



2. *What I cannot create, I do not understand.* – Richard Feynman

Autoencoder (AE)

Hidden layer (code)



Avoiding Trivial Identity

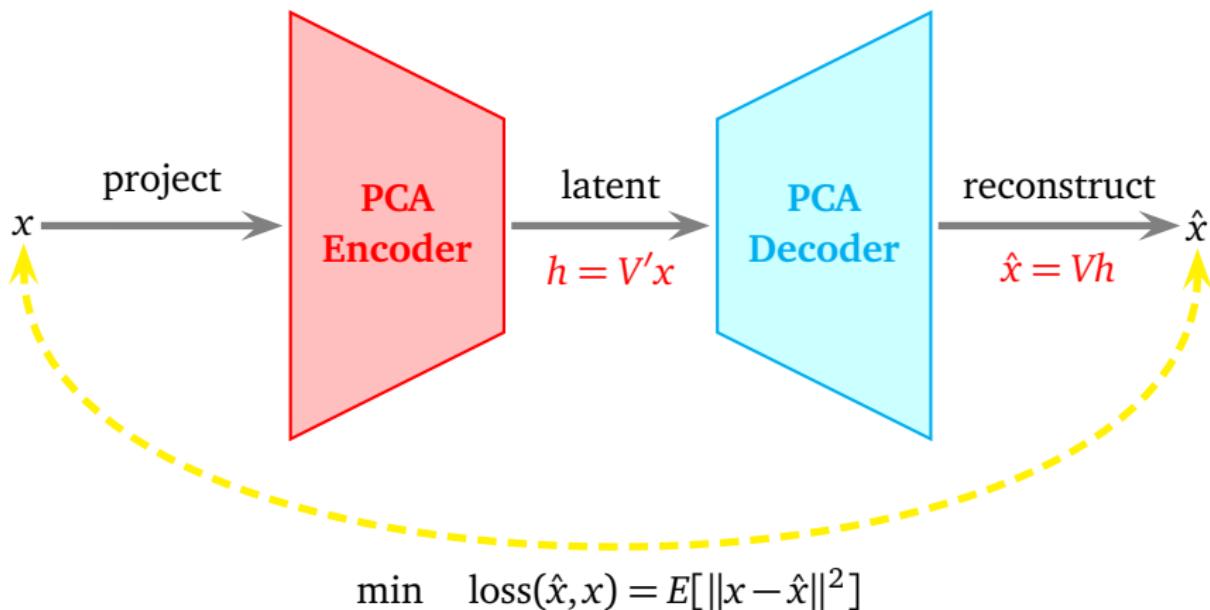
1. Undercomplete autoencoders

- ▶ h has lower dimension than x
- ▶ f or g has low capacity (e.g., linear g)
- ▶ Must discard some information in h

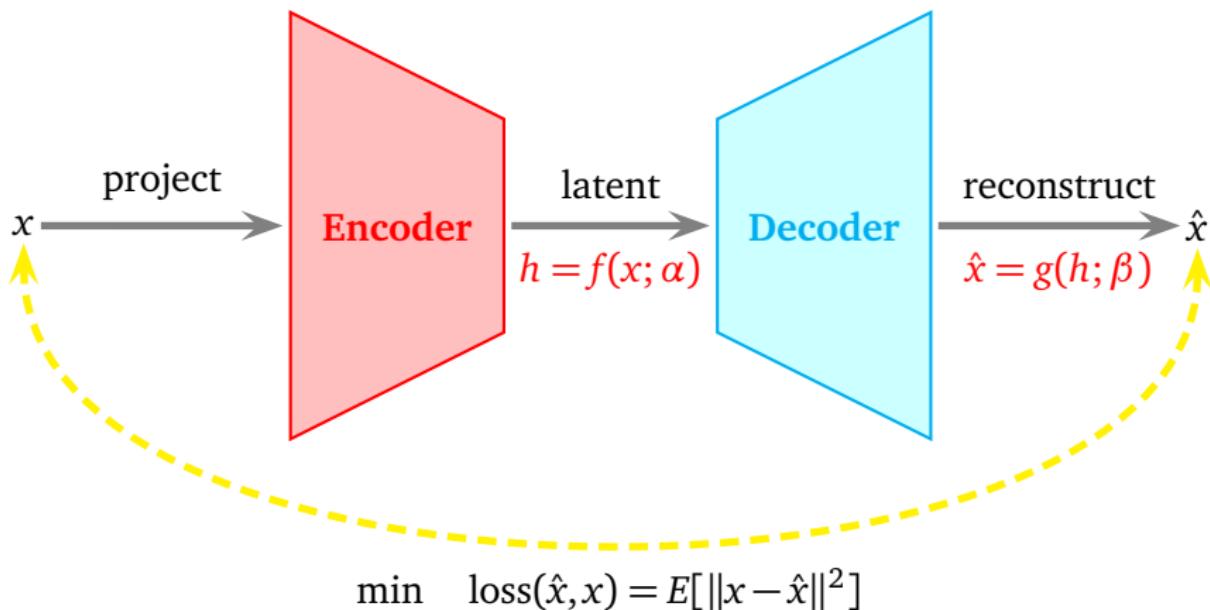
2. Overcomplete autoencoders

- ▶ h has higher dimension than x
- ▶ Must be regularized: e.g. sparsity

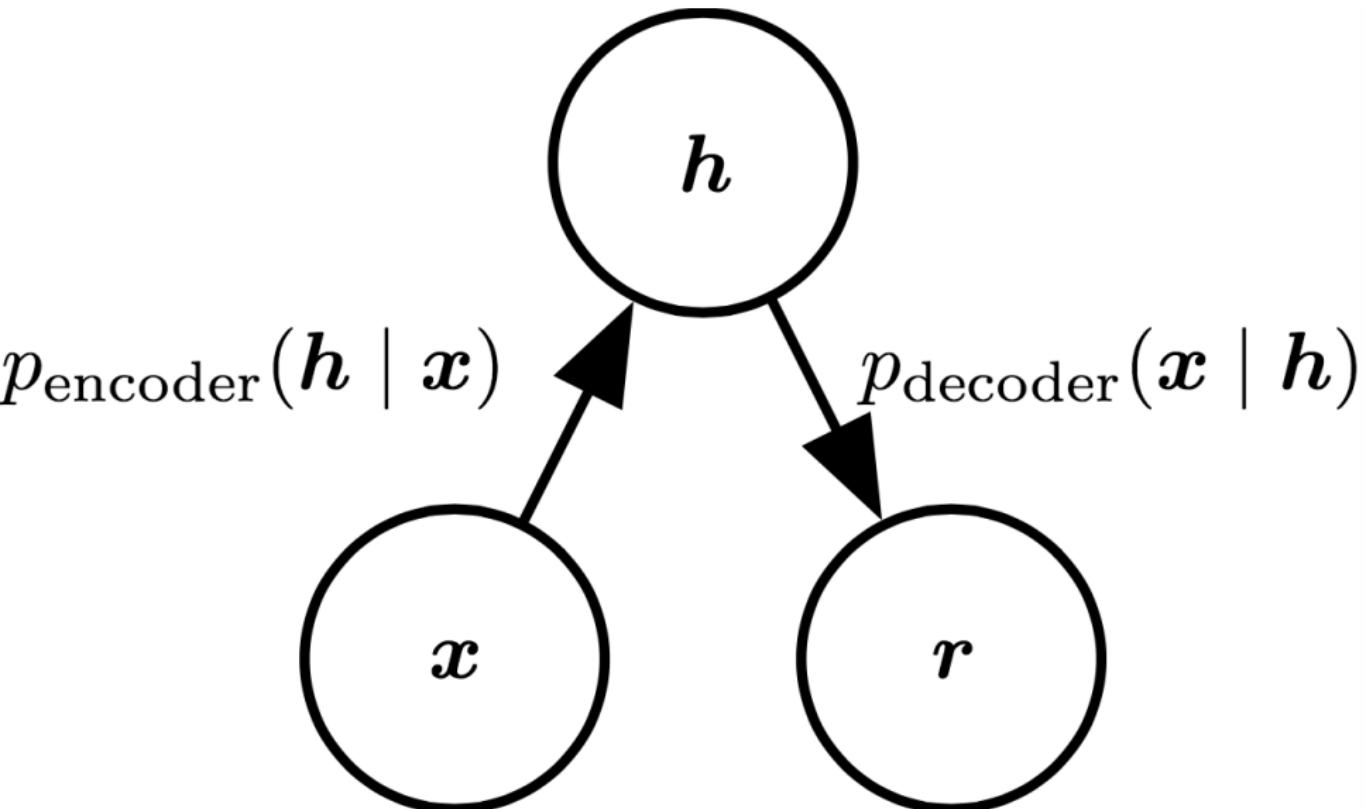
PCA as An Encoder-Decoder Pipeline



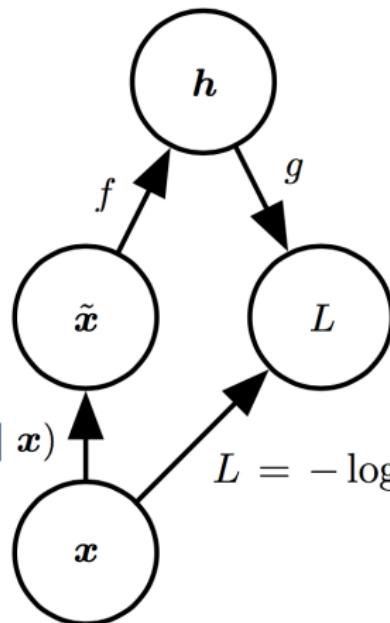
AE: A Nonlinear Encoder-Decoder Pipeline



Stochastic Autoencoder

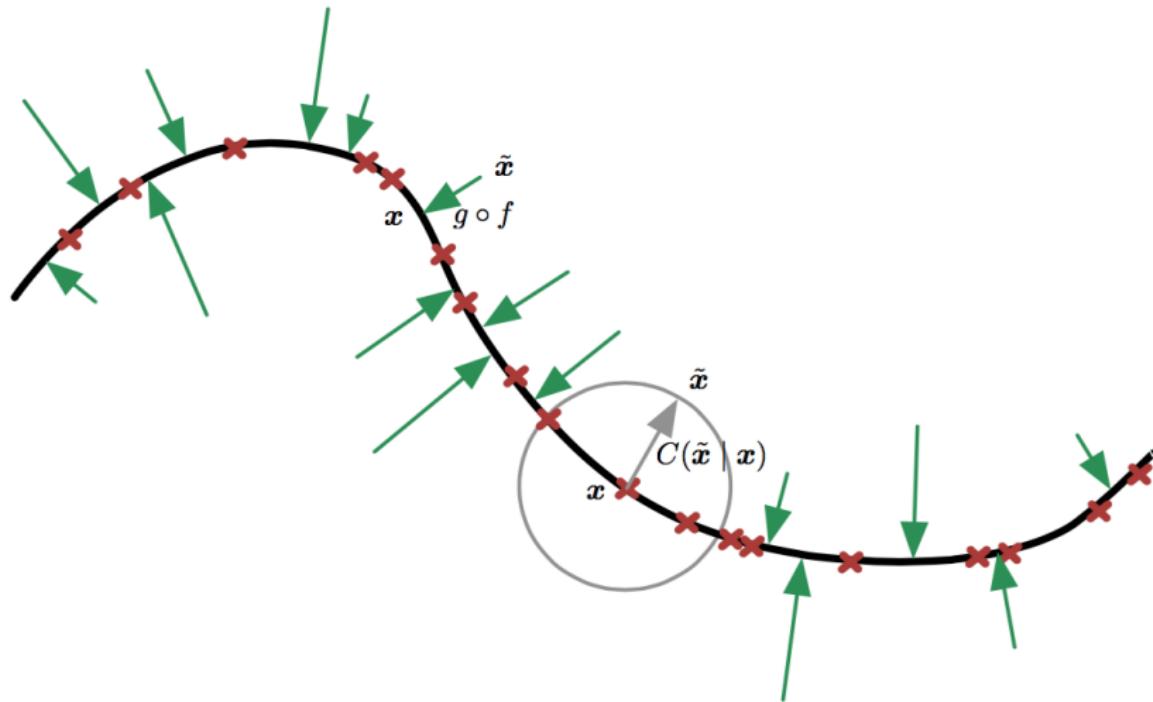


Denoising Autoencoder



C : corruption process
(introduce noise)

Denoising Autoencoder Learns A Manifold



Generative Model: Overall Approach

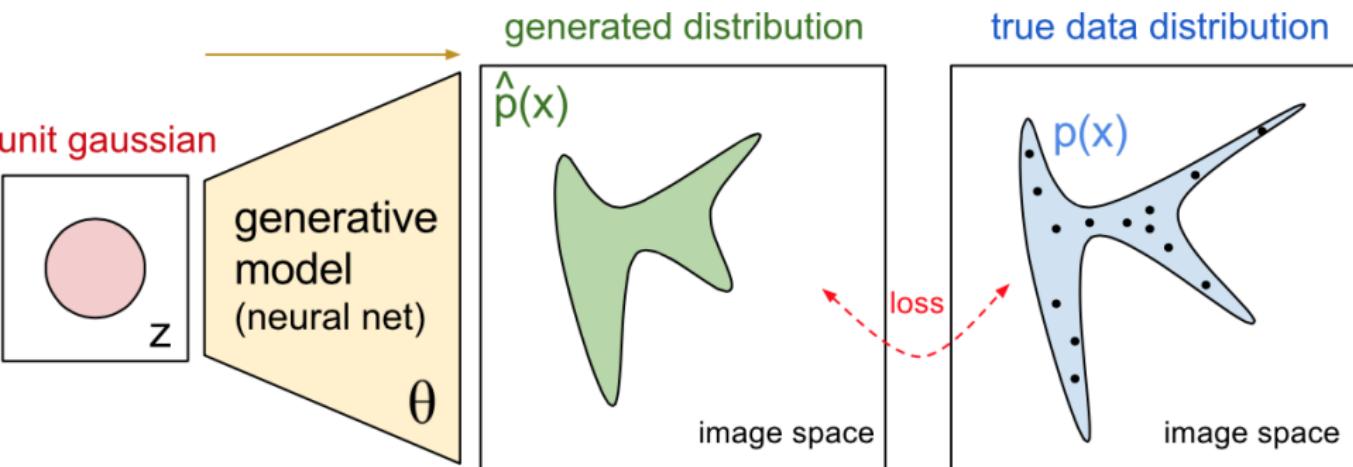
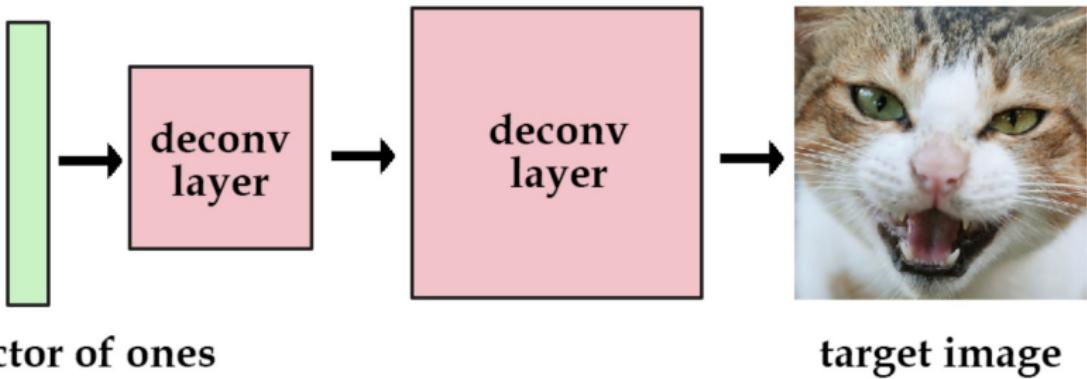
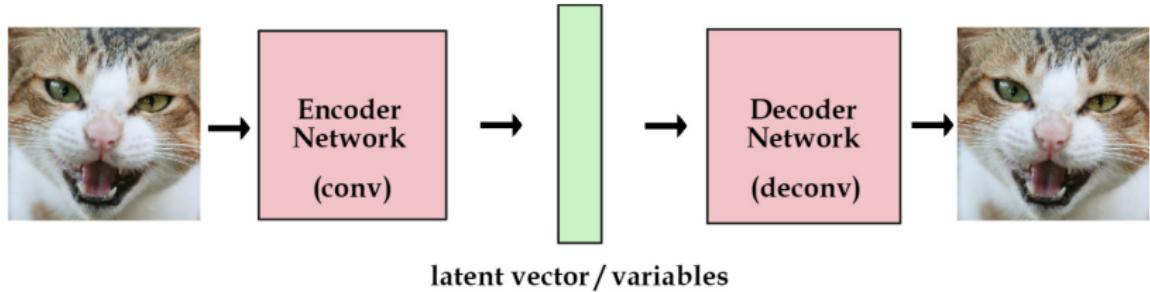


Image Generator Given A Code

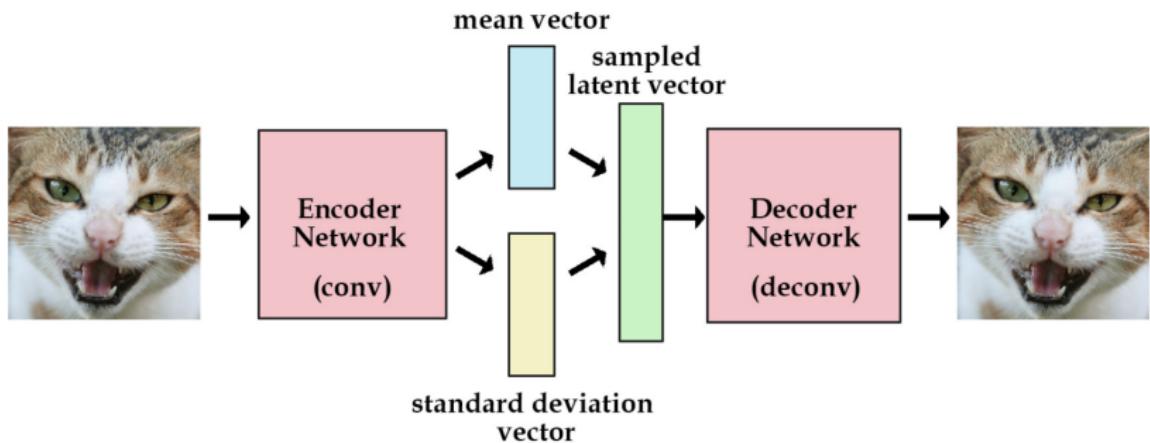


Variational Autoencoders Explained, Kevin Frans, 2016

An Autoencoder Learns Latent Code



VAE Captures and Models Code Distribution



VAE Sample Results



left: 1st epoch, middle: 9th epoch, right: original

PROGRESSIVE GROWING OF GANs FOR IMPROVED QUALITY, STABILITY, AND VARIATION

Tero Karras

NVIDIA

{tkarras,

Timo Aila

NVIDIA

,taila,

Samuli Laine

NVIDIA

,slaine,

Jaakko Lehtinen

NVIDIA and Aalto University

,jlehtinen}@nvidia.com

ABSTRACT

We describe a new training methodology for generative adversarial networks. The key idea is to grow both the generator and discriminator progressively: starting from a low resolution, we add new layers that model increasingly fine details as training progresses. This both speeds the training up and greatly stabilizes it, allowing us to produce images of unprecedented quality, e.g., CELEBA images at 1024^2 . We also propose a simple way to increase the variation in generated images, and achieve a record inception score of 8.80 in unsupervised CIFAR10. Additionally, we describe several implementation details that are important for discouraging unhealthy competition between the generator and discriminator. Finally, we suggest a new metric for evaluating GAN results, both in terms of image quality and variation. As an additional contribution, we construct a higher-quality version of the CELEBA dataset.

Generative Adversarial Networks (GAN)

GENERATIVE ADVERSARIAL NETS, 2014

[Goodfellow et al. 2014]

We propose a new framework for estimating the quality of generative models via a game. In this framework, we simultaneously train two models, a generative model G that captures the data distribution, and a discriminative model D that estimates the probability that a sample came from the training data rather than G . The training procedure for G is to maximize the probability of D making a mistake. This

Acknowledgements: The following set of slides are provided by Tero Karras and Jaakko Lehtinen.



Player 1



Player 2

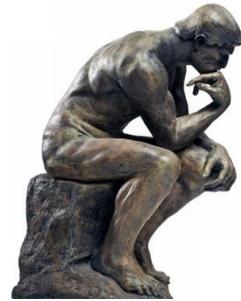




Player 1



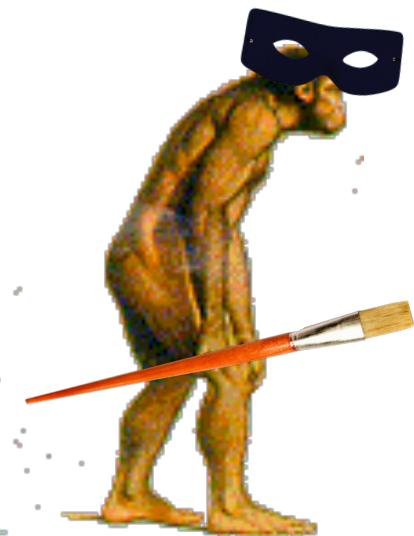
Player 2



Player 1



Player 2



Player 1

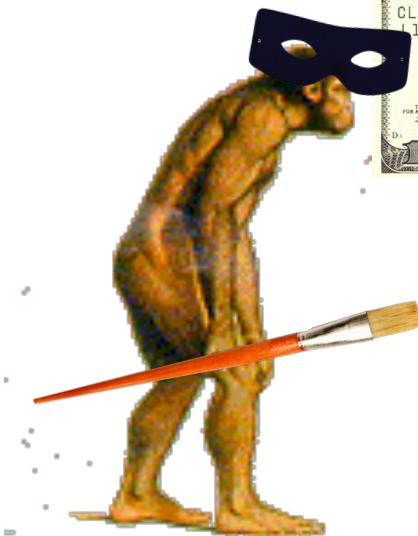


Player 2



Player 1

Player 2



Player 1



Player 2



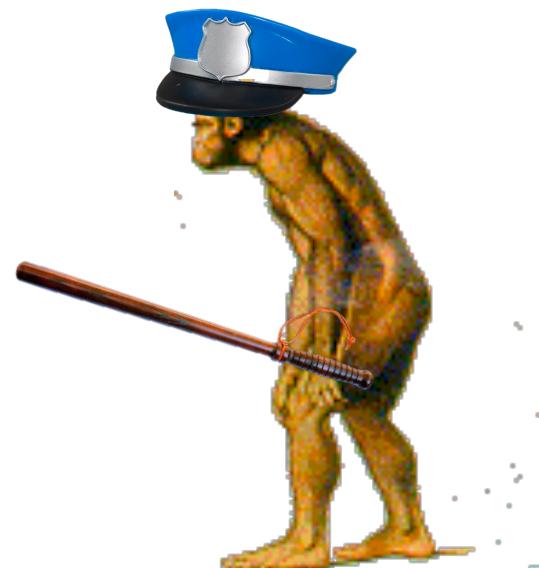
Player 1



Player 2



Player 1



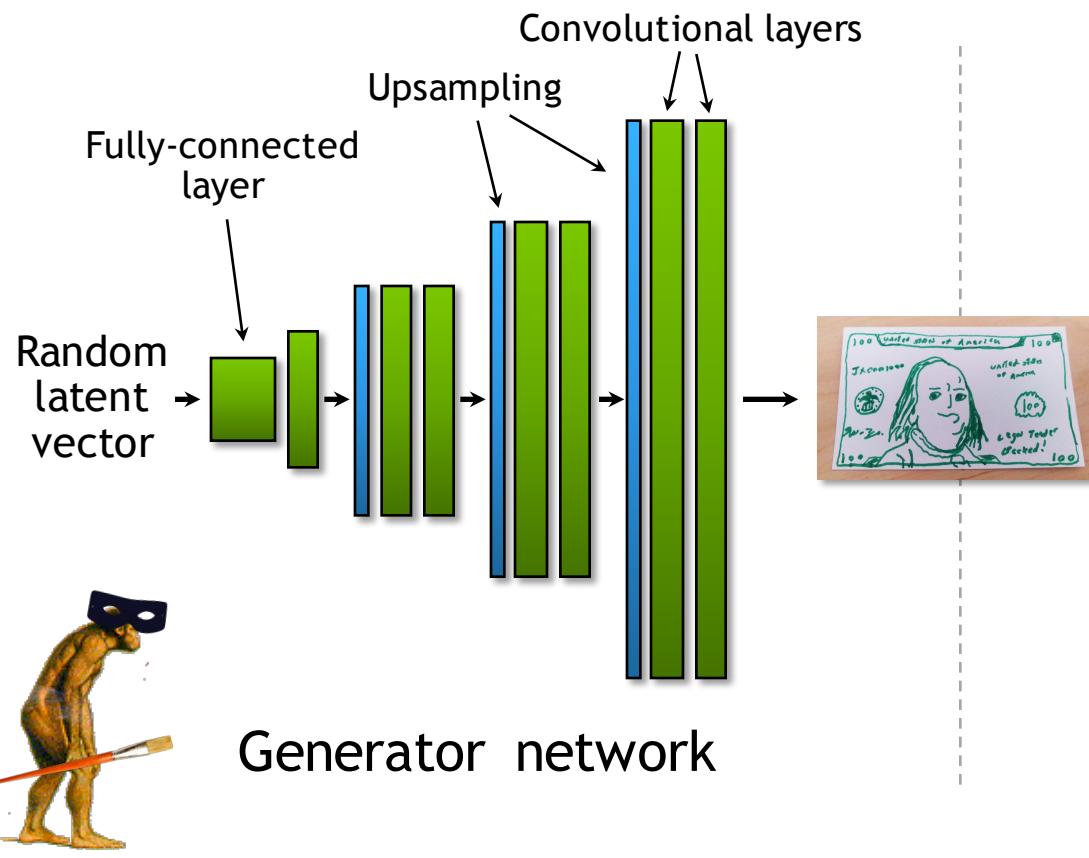
Player 2



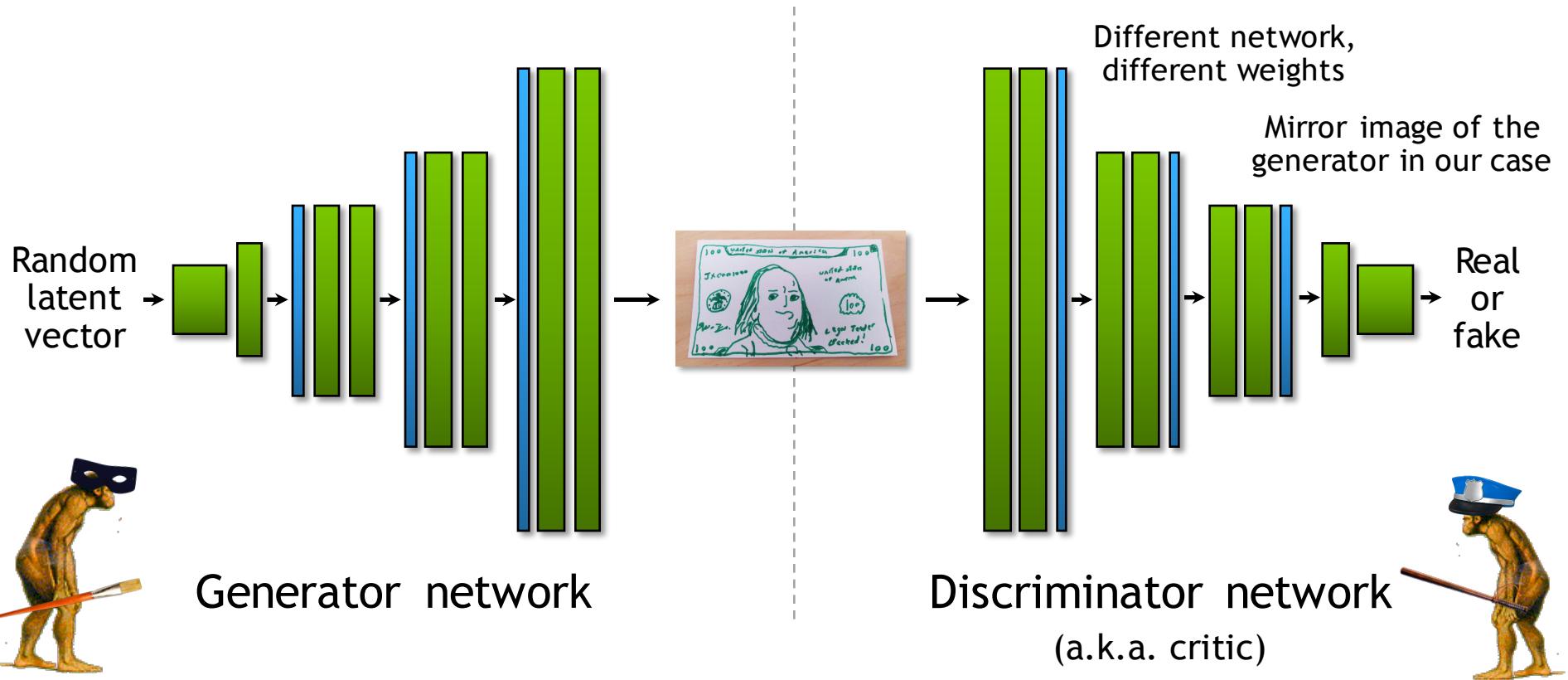
Player 1

Player 2

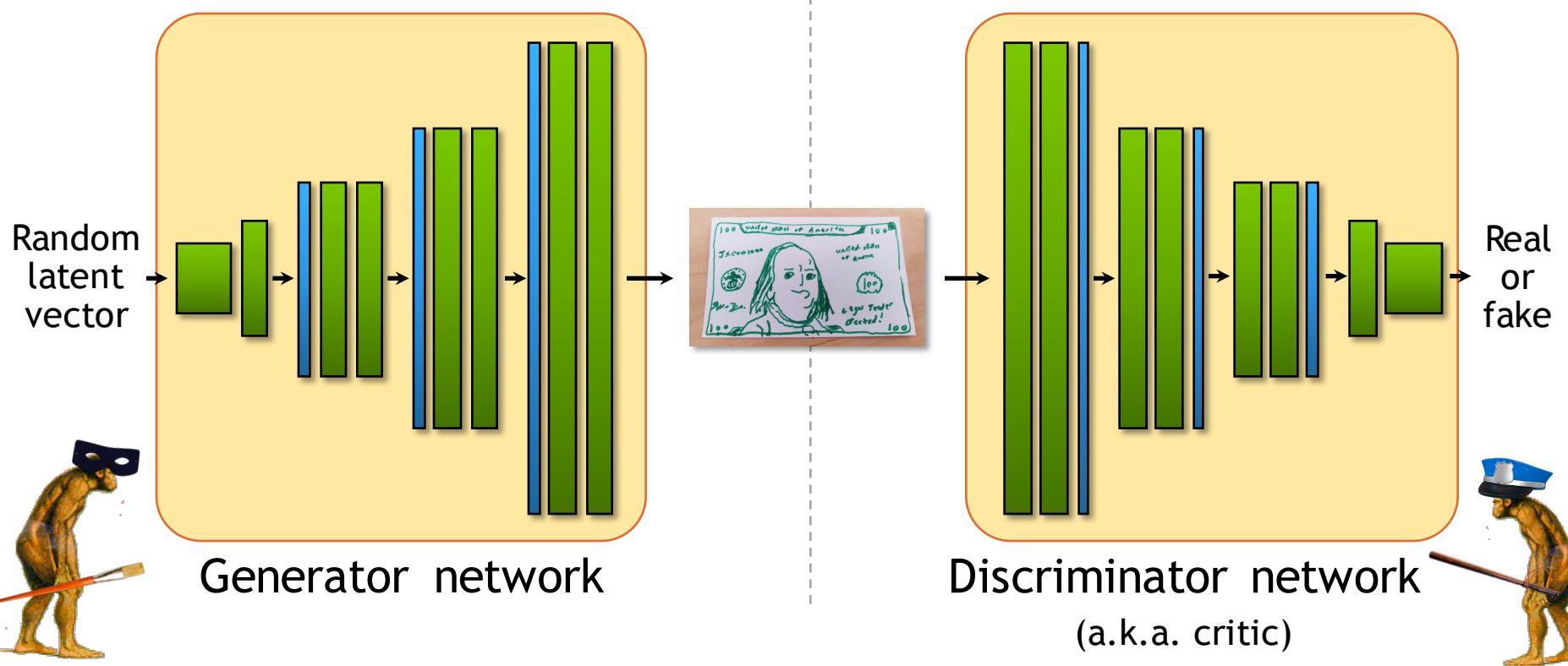
In practice



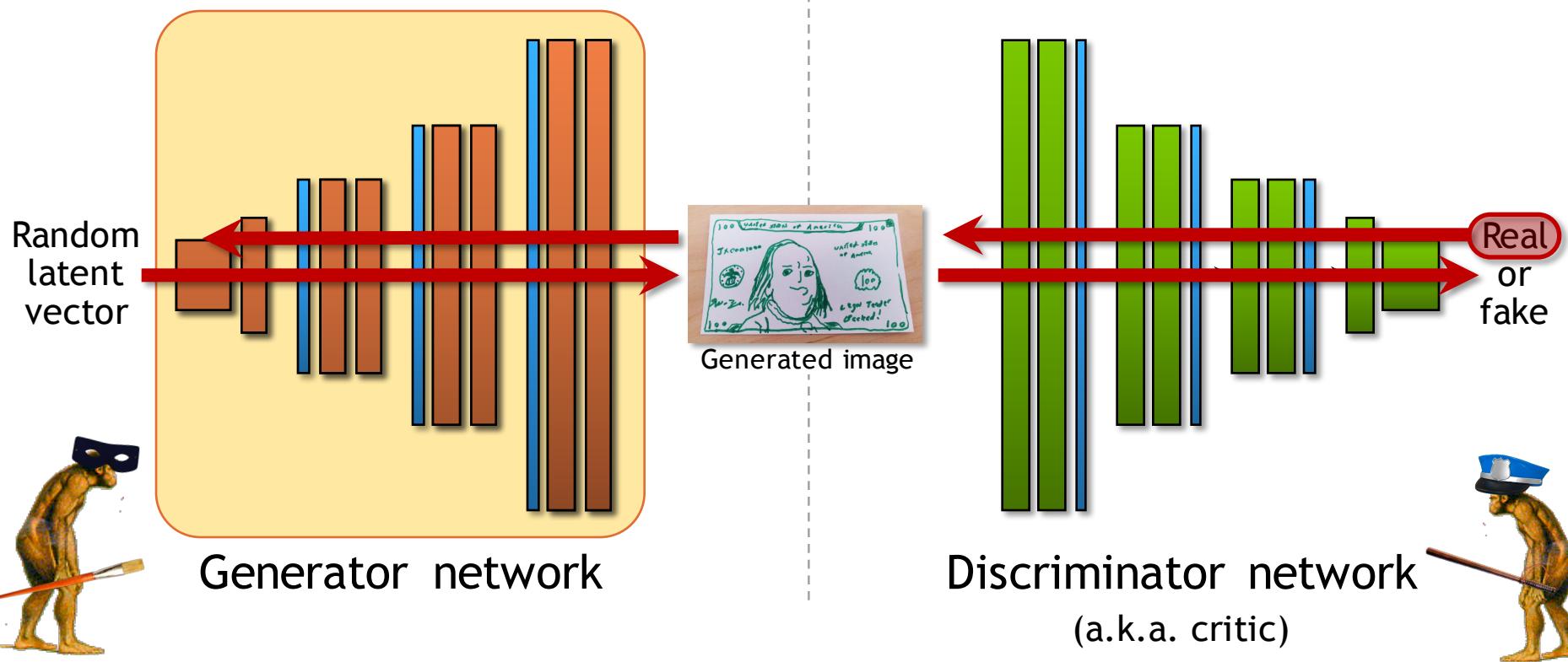
In practice



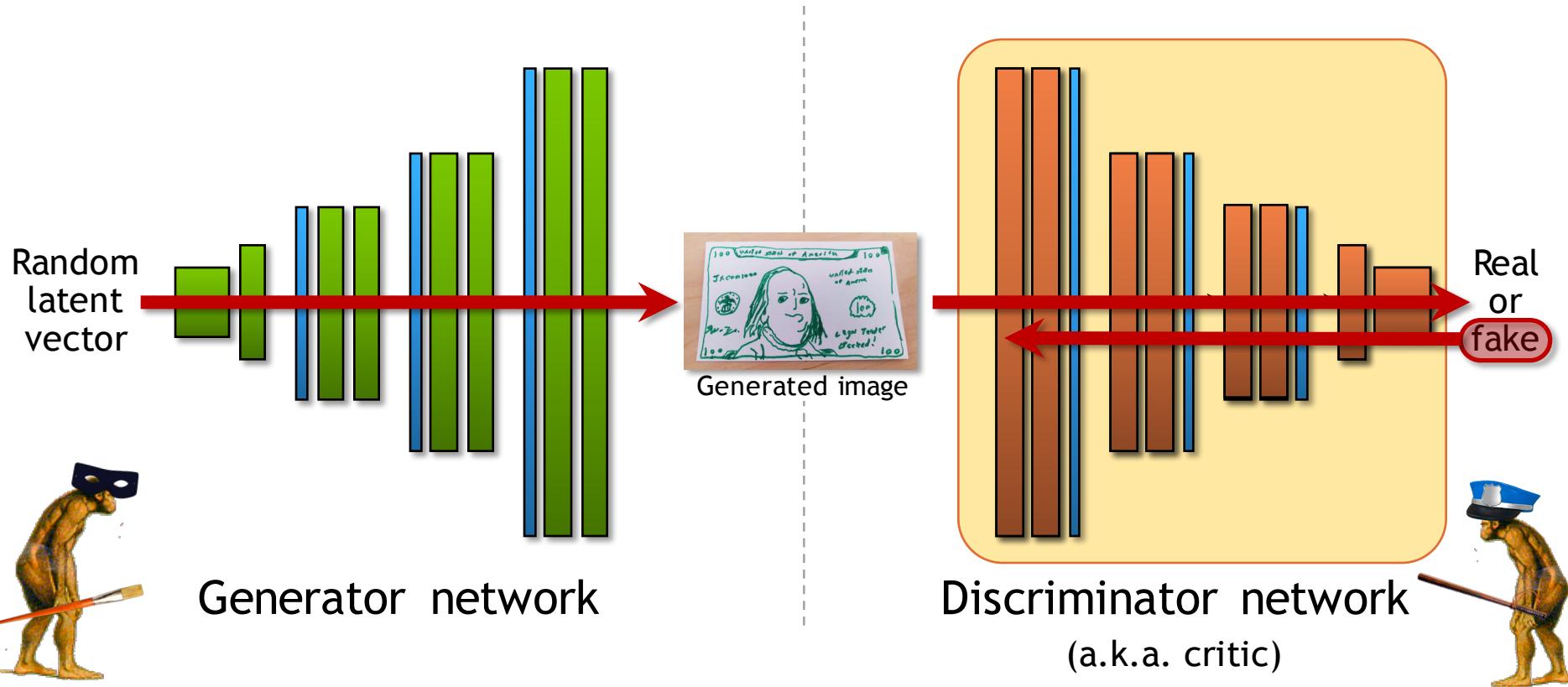
Training



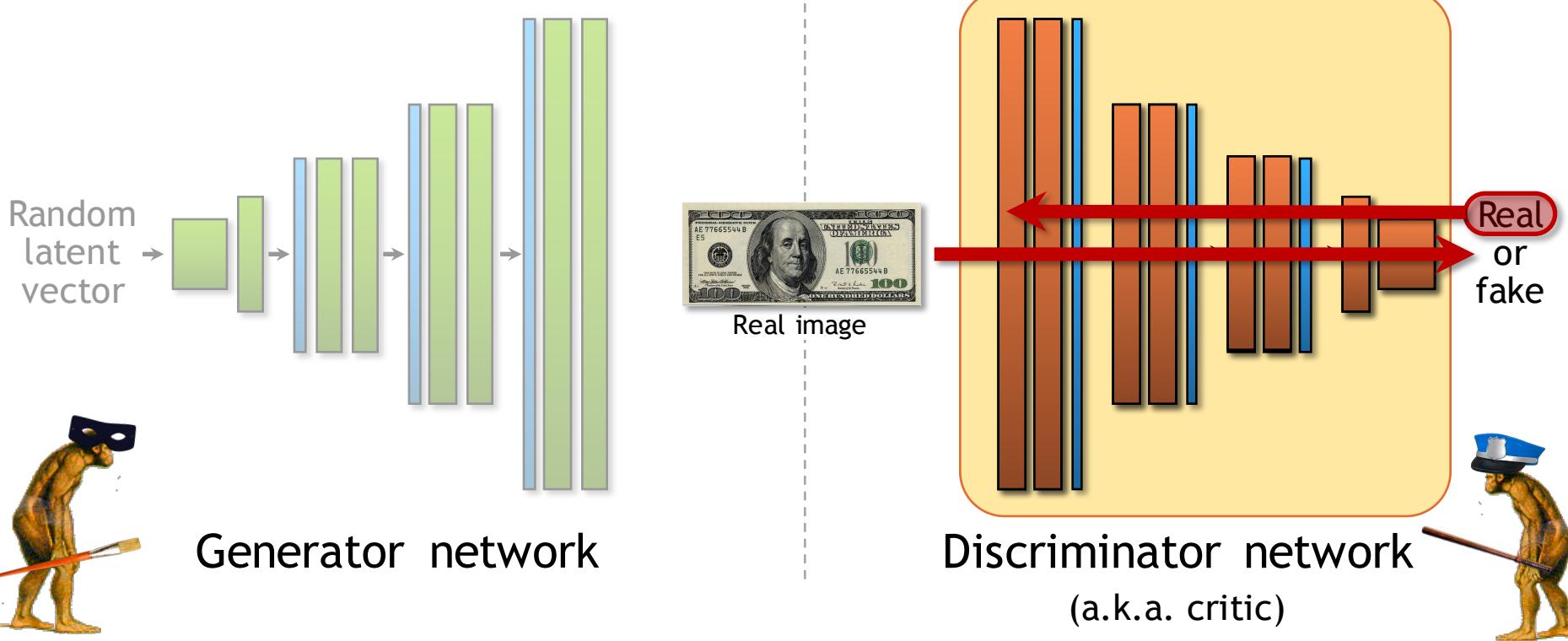
Training



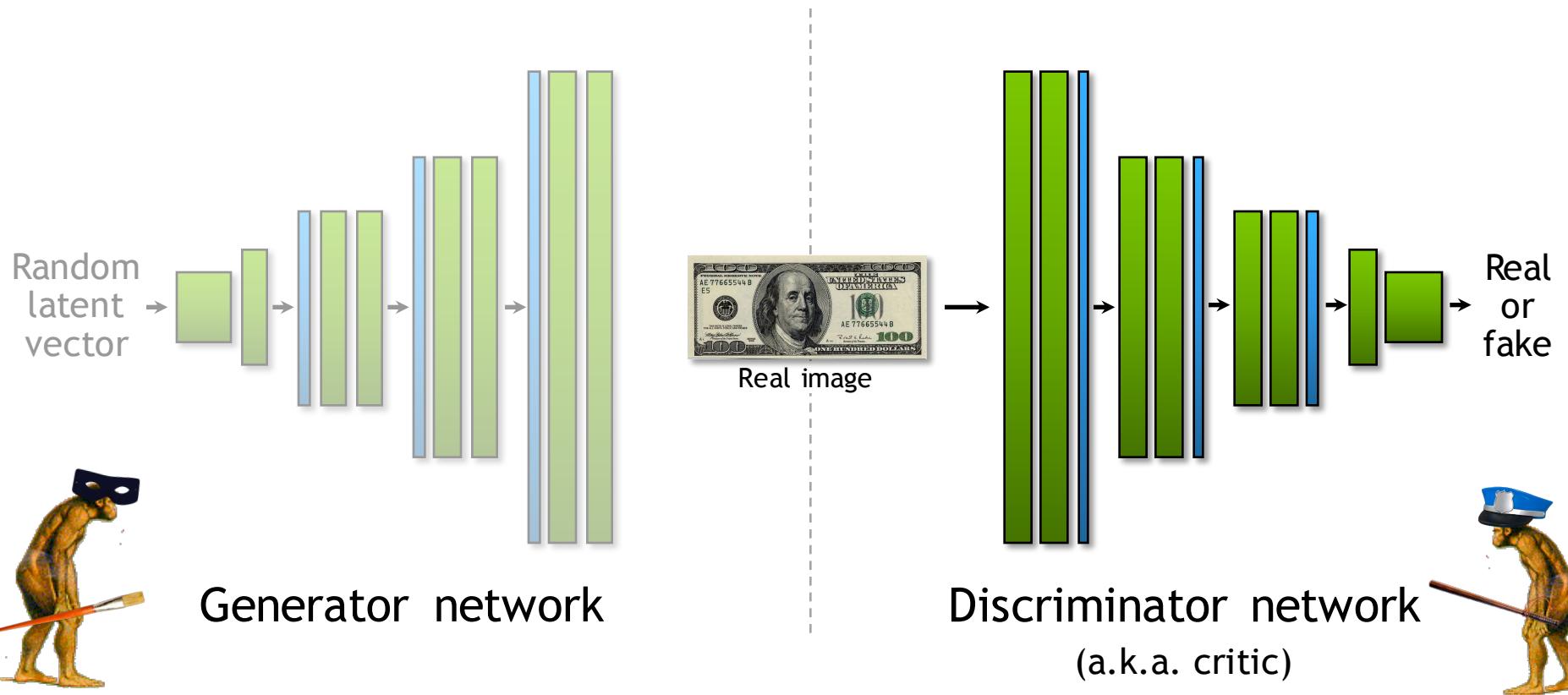
Training



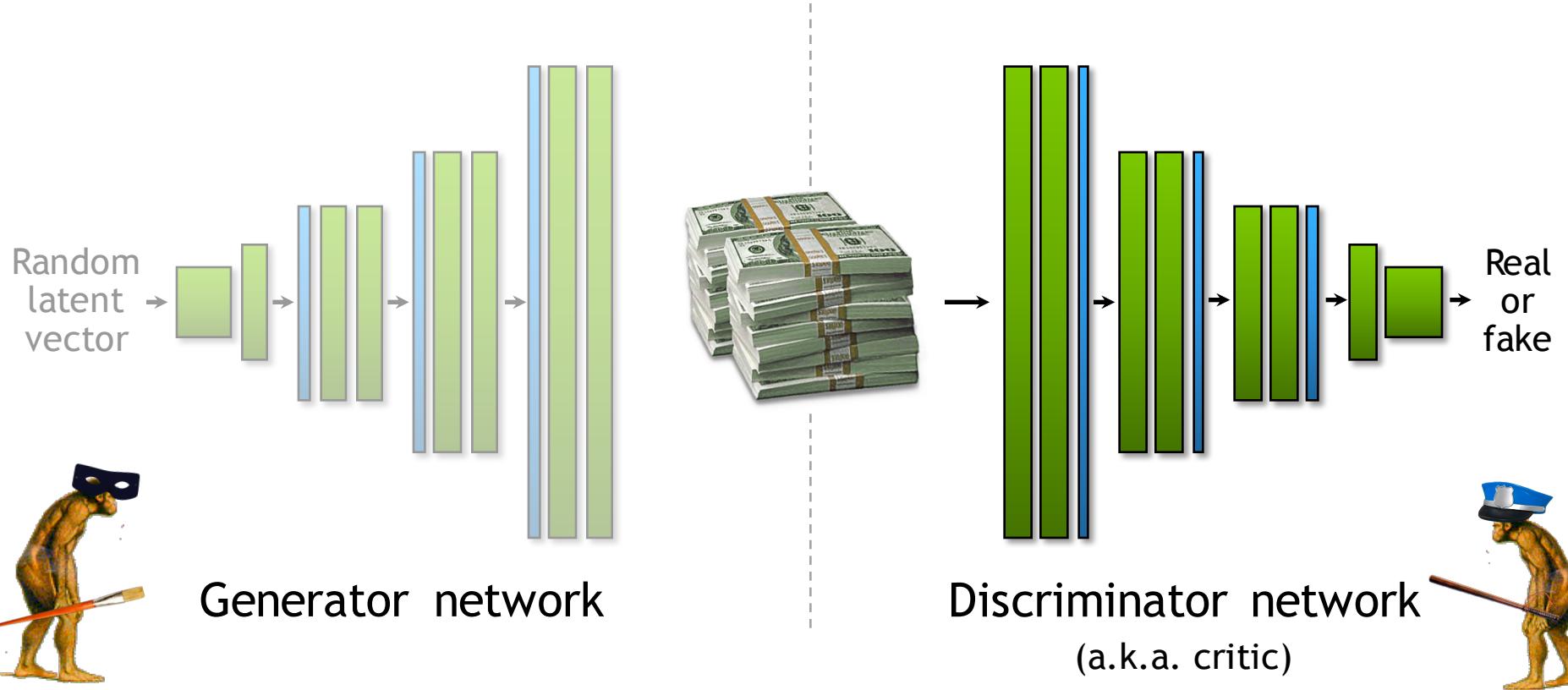
Training



Training



Training



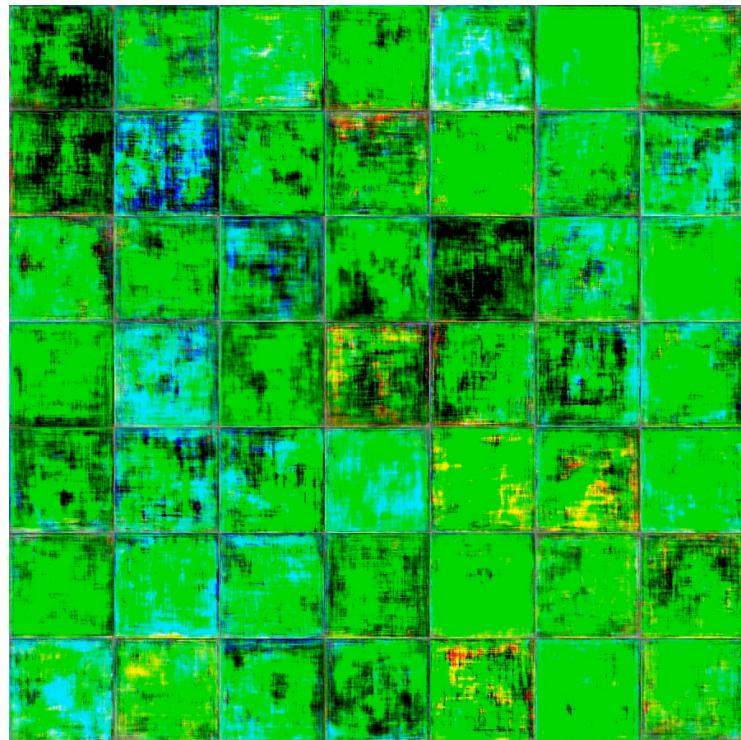
UNSUPERVISED REPRESENTATION LEARNING WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS, 2016

good image representations is by training Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), and later reusing parts of them as feature extractors or discriminator networks as feature extractors for supervised tasks. GANs provide a principled alternative to maximum likelihood techniques. One can additionally argue that the generative process and the lack of a heuristic cost function (such as pixel-wise independent mean-square error) are attractive to representation learning. GANs have

TOWARDS PRINCIPLED METHODS FOR TRAINING GENERATIVE ADVERSARIAL NETWORKS, 2017

Generative adversarial networks (GANs) (Goodfellow et al., 2014a) have achieved great success in generating realistic images. However, they are widely considered to be difficult to train. Important problems include stabilizing sequence learning methods for speech and language, (Goodfellow et al., 2015; Radford et al., 2015; Salimans et al., 2016; Lamb et al., 2016; Wu et al., 2016)

What could possibly go wrong?



What could possibly go wrong?



What could possibly go wrong?



UNSUPERVISED REPRESENTATION LEARNING WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS, 2016

Historical attempts to scale up GANs were unsuccessful. This motivated the authors of LAPGAN (Denton et al., 2017) to iteratively update the generator until it was able to generate images that were visually indistinguishable from the real data. However, this approach to iterative training was unable to scale up to large datasets.

TOWARDS PRINCIPLED METHODS FOR TRAINING GENERATIVE ADVERSARIAL NETWORKS, 2017

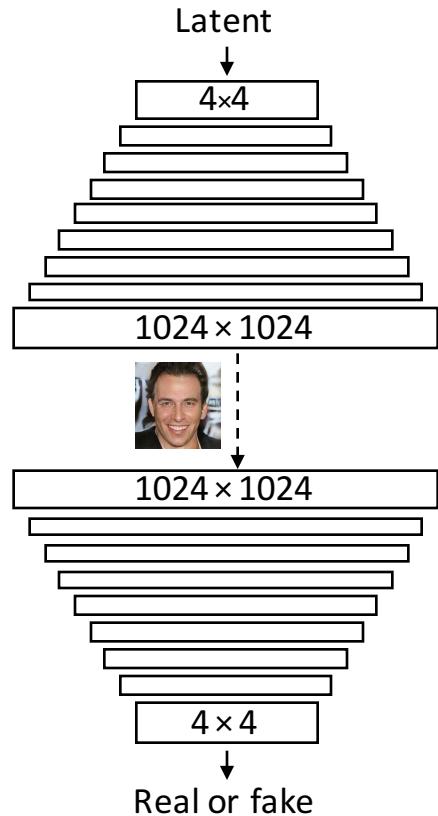
However, they still remain relatively finding stable architectures [Salimans et al., 2016]. Despite their success, there is little to no theory explaining why GANs still rely on heuristic modifications. The training process is often difficult to train and unstable.

sensitive to modifications

UNROLLED GENERATIVE ADVERSARIAL NETWORKS, 2017

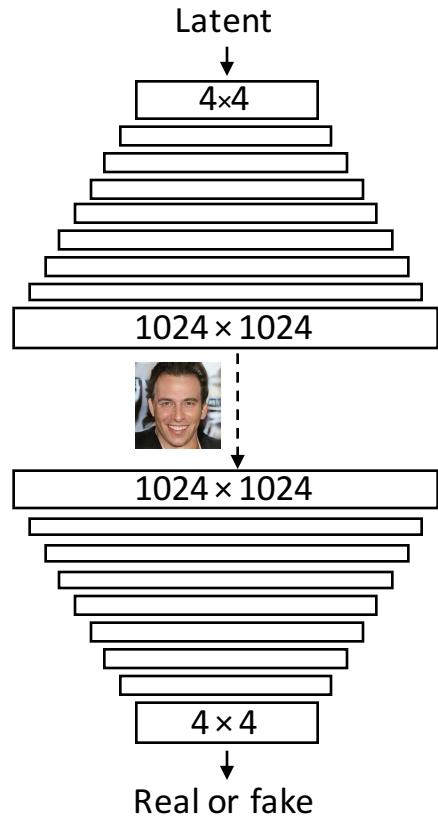
providing a solution. In practice, however, GANs suffer from several **failure mode**. One common failure mode is that the generator and discriminator both produce only a single sample or a small family of samples. Another involves the generator and discriminator oscillating during training rather than converging to a fixed point. In addition, if one agent is much more powerful than the other, the learning signal to the other agent becomes **useless**. To train GANs many tricks must be employed, such as careful selection of the mini-batch size (Radford et al., 2015), minibatch discrimination (Salimans et al., 2016), and noise injection (Salimans et al., 2016; Sonderby et al., 2016). Even with these tricks the set of hyperparameters for which training is successful is generally very small in practice.

Traditional approach

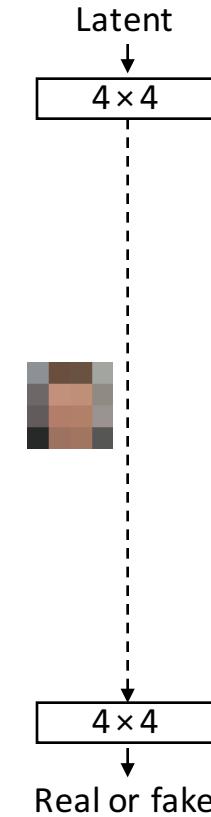


- High resolution \Rightarrow deep networks
- All layers are initialized to random weights
- Neither of the two networks has any idea what it's supposed to do!

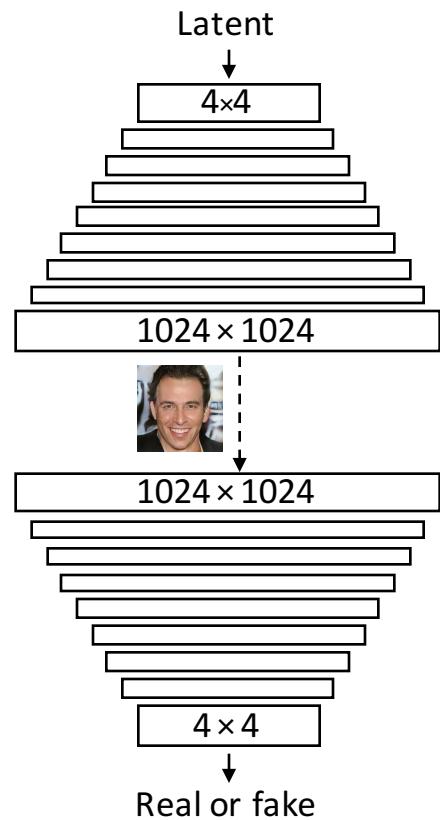
Traditional approach



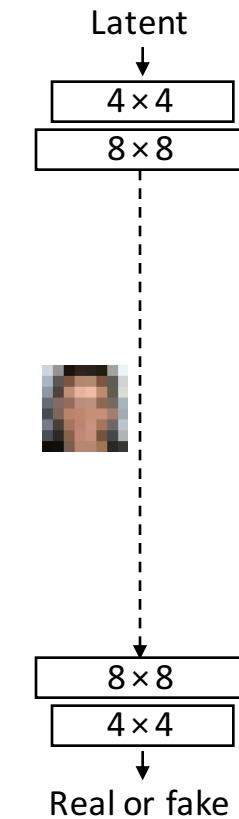
Progressive growing



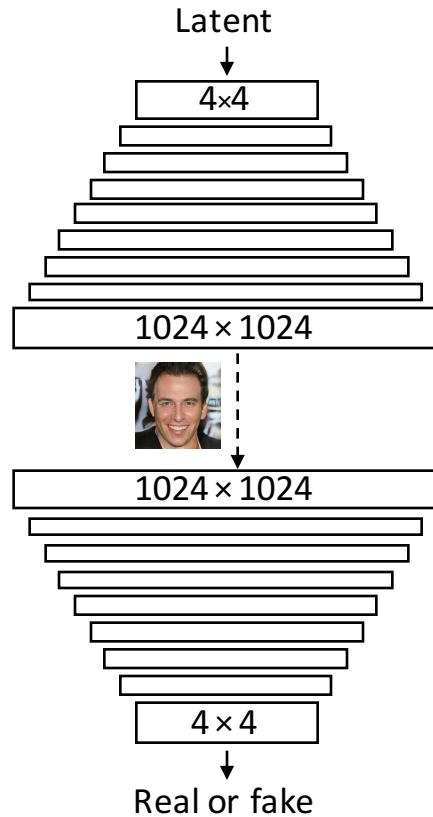
Traditional approach



Progressive growing



Traditional approach



Progressive growing

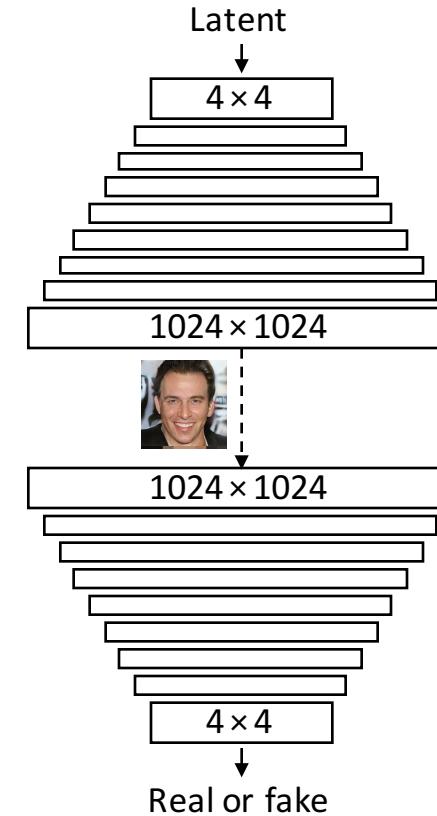




Figure 5: 1024×1024 images generated using the CELEBA-HQ dataset. See Appendix F for a larger set of results, and the accompanying video for latent space interpolations. On the right, two images from an earlier megapixel GAN by Marchesi (2017) show limited detail and variation.