

Generative Models

Links to have open

- [GANSpace](#)
- [GAN Lab](#)
- [GANpaint](#)
- <https://j.mp/cam2head>





Which was real?

- First slide: FFHQ dataset (<https://github.com/NVlabs/ffhq-dataset>)
- Second slide: mid-training snapshot of fine-tuning StyleGAN2

Today: Generation

- One approach: LMs.
- Today: alternatives
- Applications examples: [HAI-GEN workshop](#)
 - Platform game level generation
 - 3D animal shapes
 - Controlling an instrument with your voice g.co/tonetransfer
 - Generating documentation for Jupyter notebooks

HAI-GEN Workshop at IUI 2021

2nd Workshop on Human-AI Co-Creation with Generative Models

This Artwork Does Not Exist

Using generative adversarial networks (GAN), we can learn how to create realistic-looking fake versions of almost anything, as shown by this collection of sites that have sprung up in the past month. Learn [how it works](#).



This Person Does Not Exist

The site that started it all, with the name that says it all. Created using a style-based generative adversarial network (StyleGAN), this website had the tech community buzzing with excitement and intrigue and inspired many more sites.

Created by Phillip Wang.



This Cat Does Not Exist

These purr-fect GAN-made cats will freshen your feeline-gs and make you wish you could reach through your screen and cuddle them. Once in a while the cats have visual deformities due to imperfections in the model – beware, they can cause nightmares.

Created by Ryan Hoover.



This Rental Does Not Exist

Why bother trying to look for the perfect home when you can create one instead? Just find a listing you like, buy some land, build it, and then enjoy the rest of your life.

Created by Christopher Schmidt.

How might an AI *generate* an image?

Goal: Why can generative models work?

- For details on *how*, see the readings for this week
- What about the world makes this work?

Main Ideas Today

- Real things occupy a tiny portion of the space of all possible things.
 - Images, sentences, sounds, etc.
 - We can describe that tiny space as a *distribution*
- We can compose learned functions in various ways to model that distribution

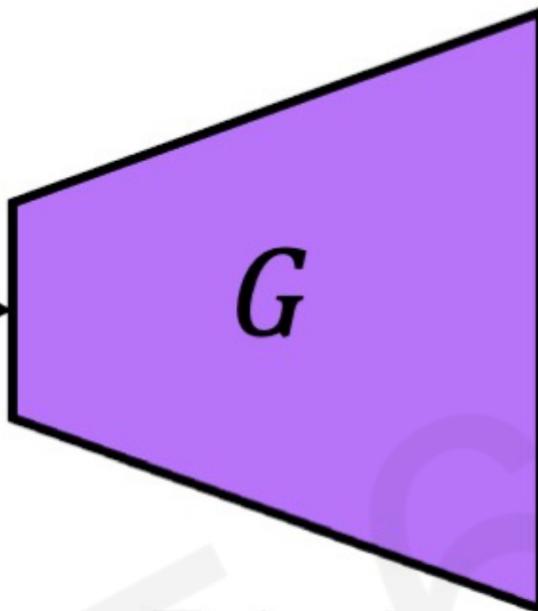
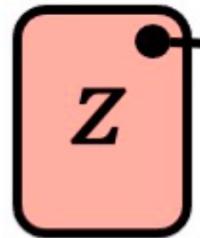
Real things occupy a tiny portion of the space of all possible things.

- Natural images are a small subset of all possible images

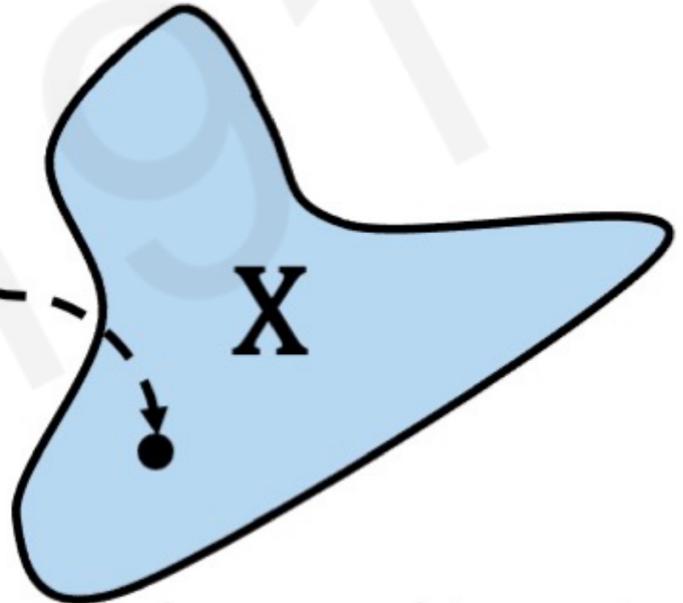
Generator Network

Gaussian noise

$$z \sim N(0,1)$$



Trained
generator



Learned target
data distribution

Demo: <https://observablehq.com/@stwind/latent-flowers-ganden>

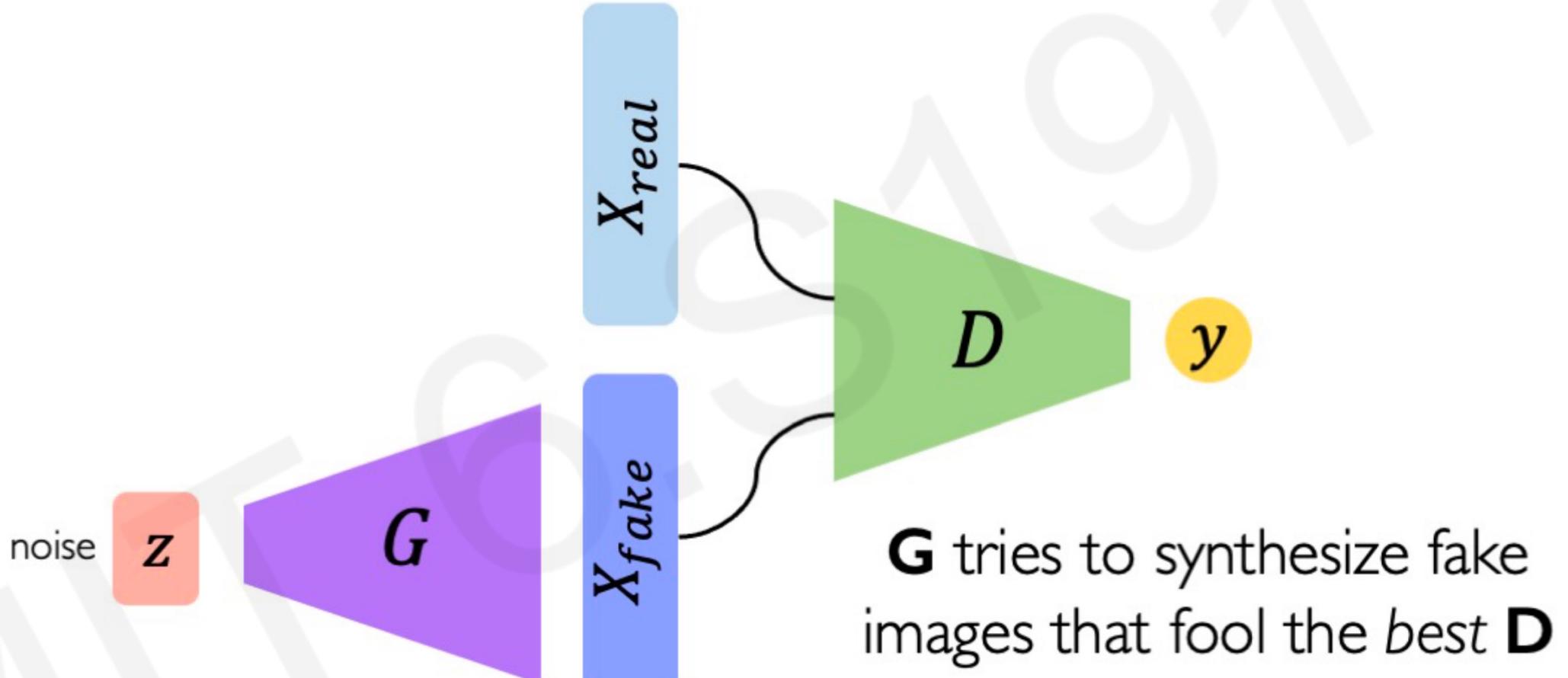
How to compose functions to model distributions?

And how do we train those functions??

Ways of modeling distributions

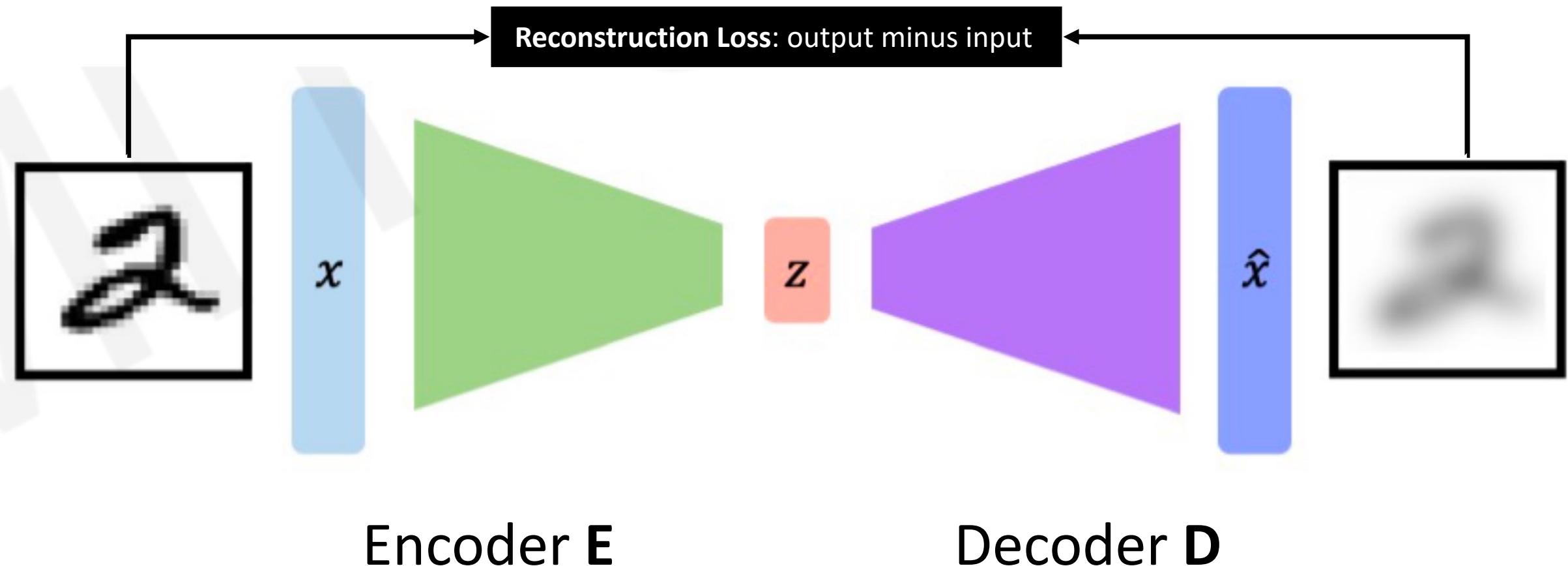
- Single decisions (softmax classifier)
- Sequential decisions (autoregressive)
- Latent variables
- Diffusion-based

Generative Adversarial Network

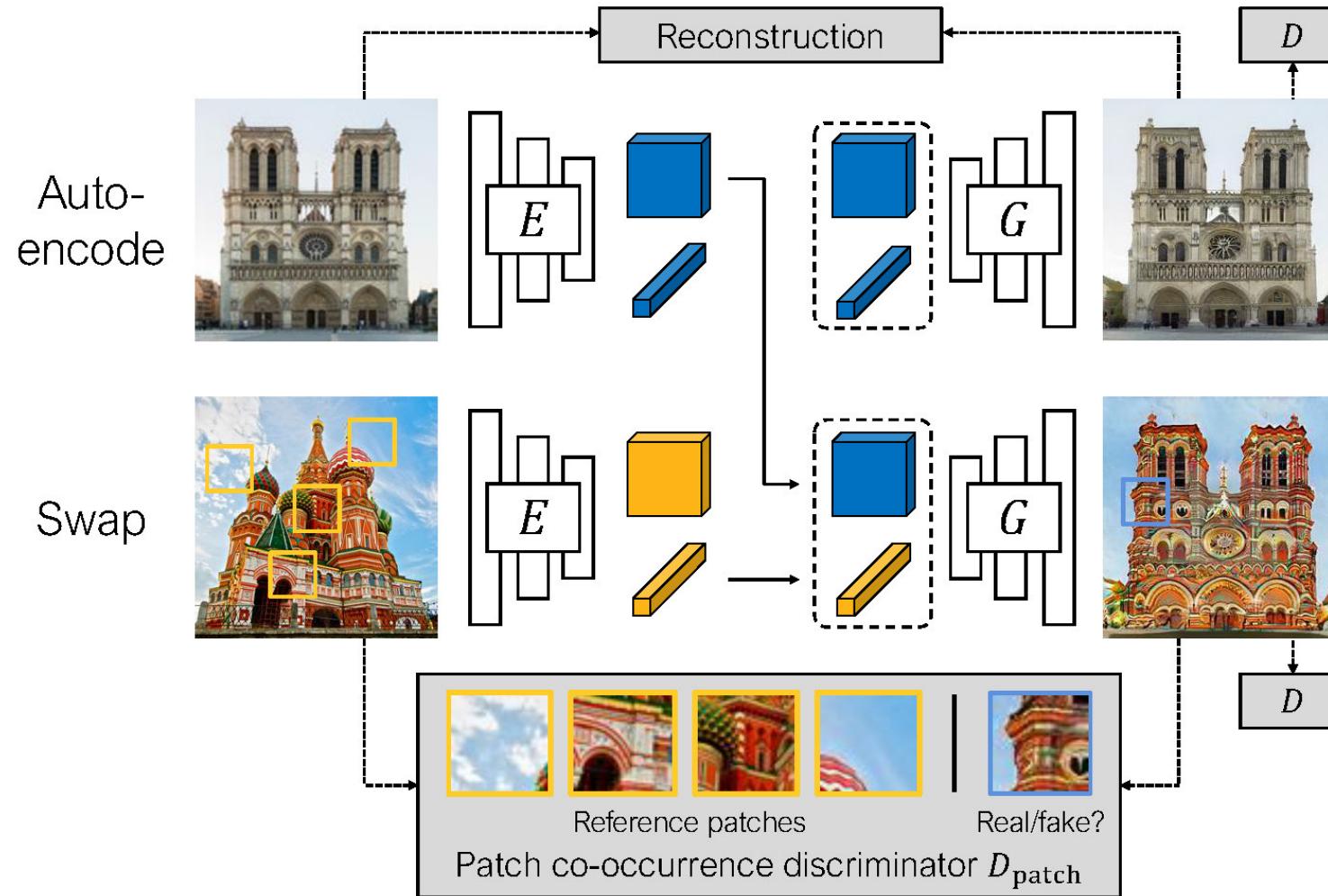


Try it at [GANLab](#)

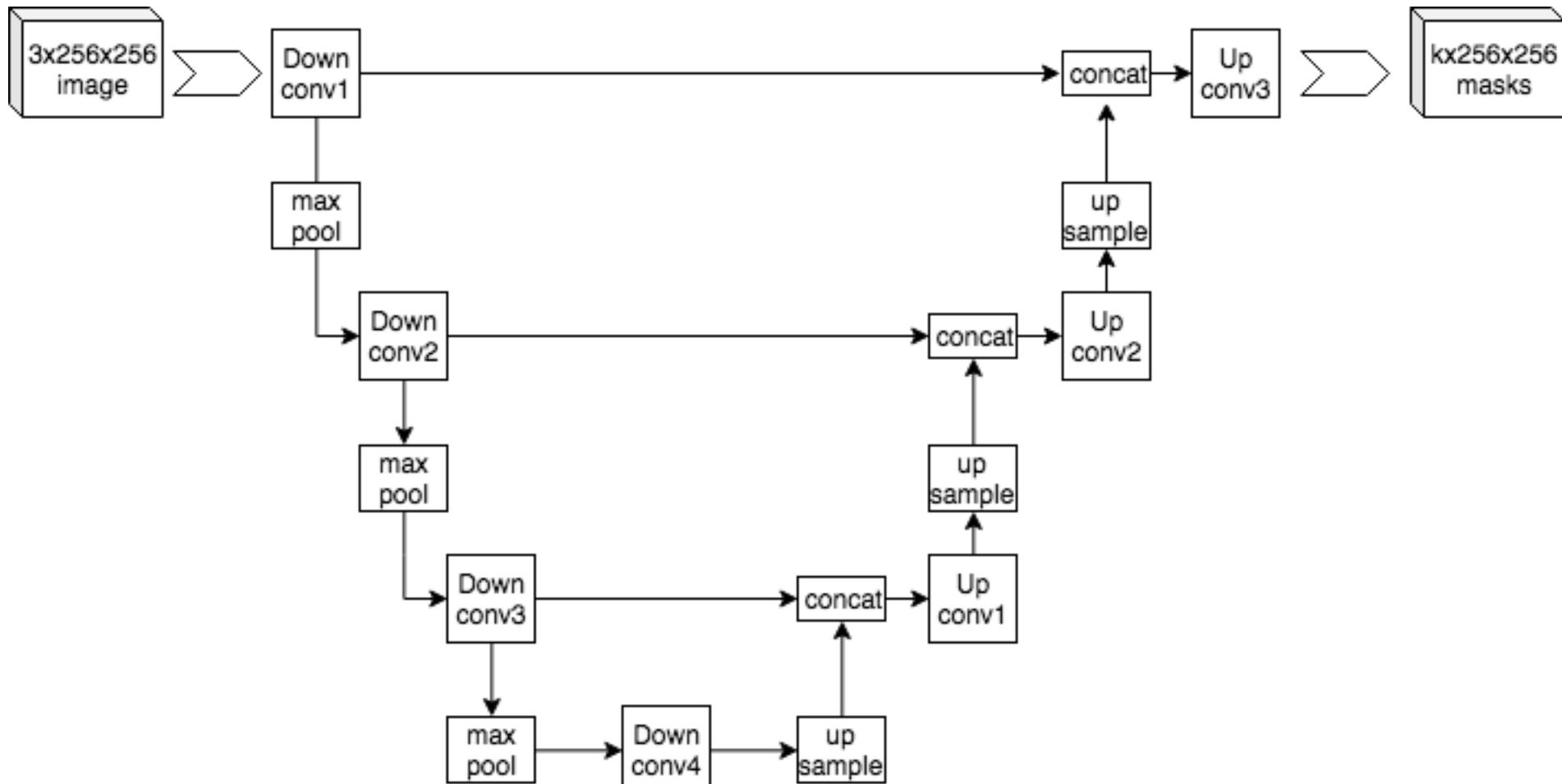
Autoencoder



Swapping Autoencoder



U-Net



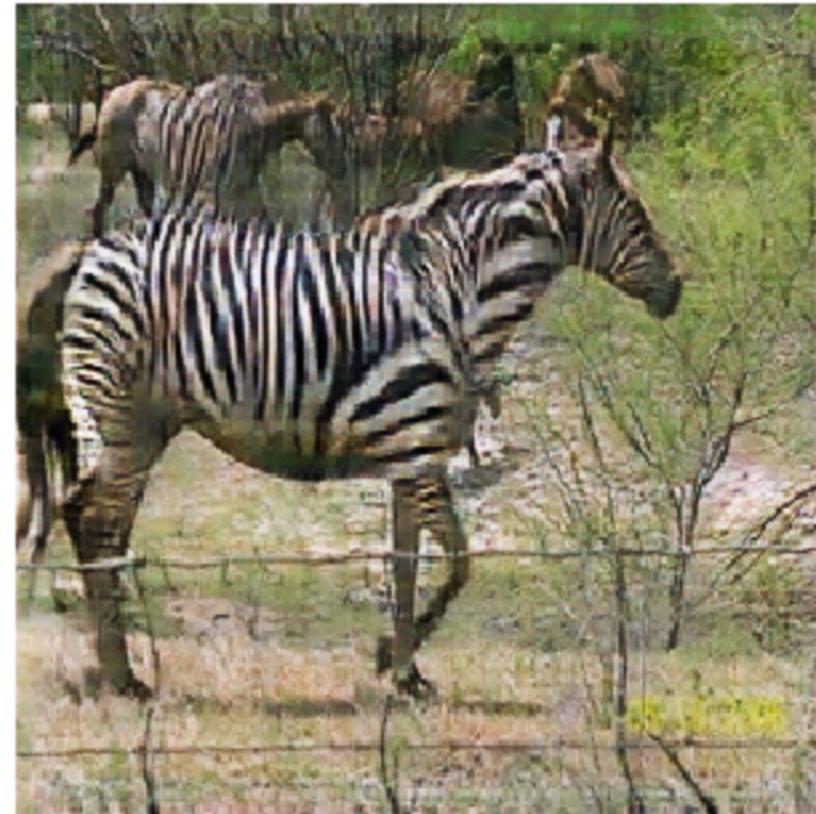
https://commons.wikimedia.org/wiki/File:Example_architecture_of_U-Net_for_producing_k_256-by-256_image_masks_for_a_256-by-256_RGB_image.png

CycleGAN (image <-> image)

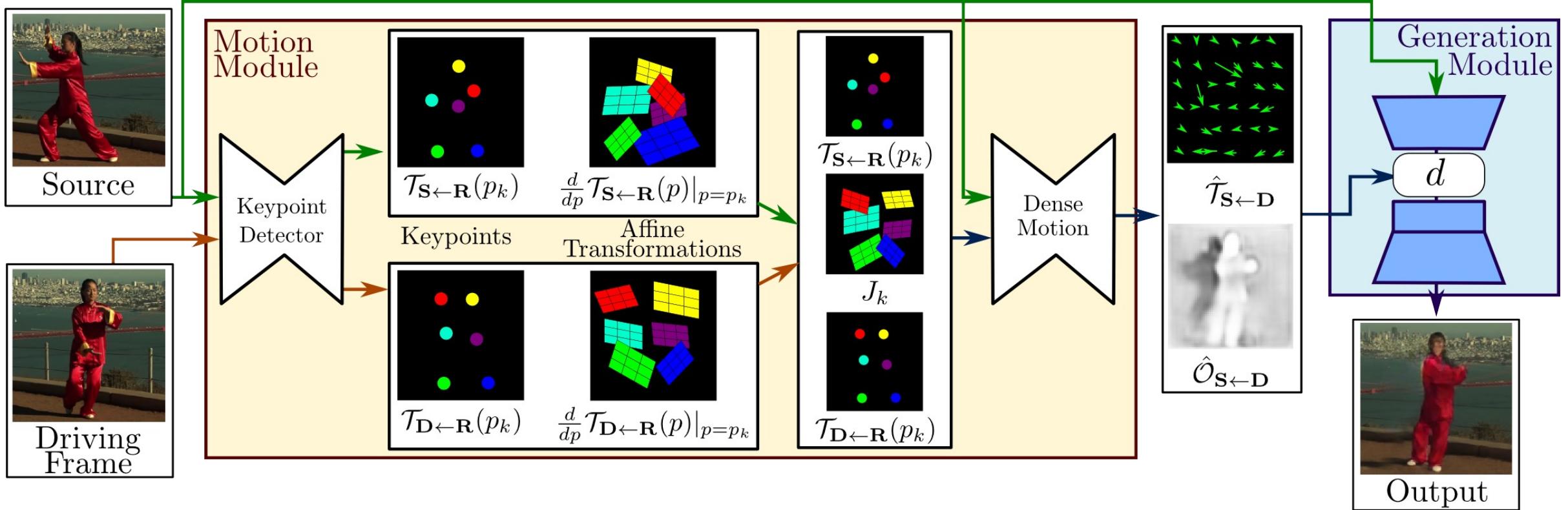
Input Image



Predicted Image

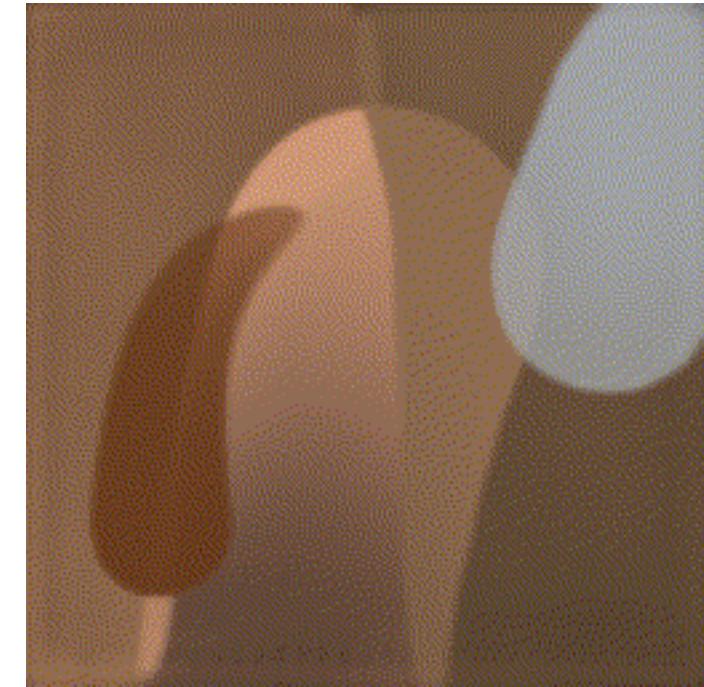


You don't need all this for a simple deepfake



Many other creative approaches

- Learning to draw an image stroke-by-stroke
 - PaintBot / <https://arxiv.org/abs/1904.02201>
 - Model-based RL: <https://arxiv.org/abs/1903.04411>
- Interactive image editing (GANPaint)
- Controls (e.g., [GANSpace](#))
- Use a sequence model (Transformers)
 - <https://compvis.github.io/taming-transformers/>
- Neural Rendering (NeRF)
 - <https://github.com/Kai-46/nerfplusplus>
 - <https://github.com/yenchenlin/awesome-NeRF>



Applications

- Video conferencing
 - Background segmentation (e.g., virtual backgrounds)
 - regenerate the face on the client side (e.g., [NVIDIA Maxine](#))
- Sim2real: train robots in simulations that perform well in real world
- Super-resolution, image/sound restoration
- Inpainting: remove distracting objects (now in Photoshop)
- Anomaly detection (e.g., tumor detection)
- Causal inference

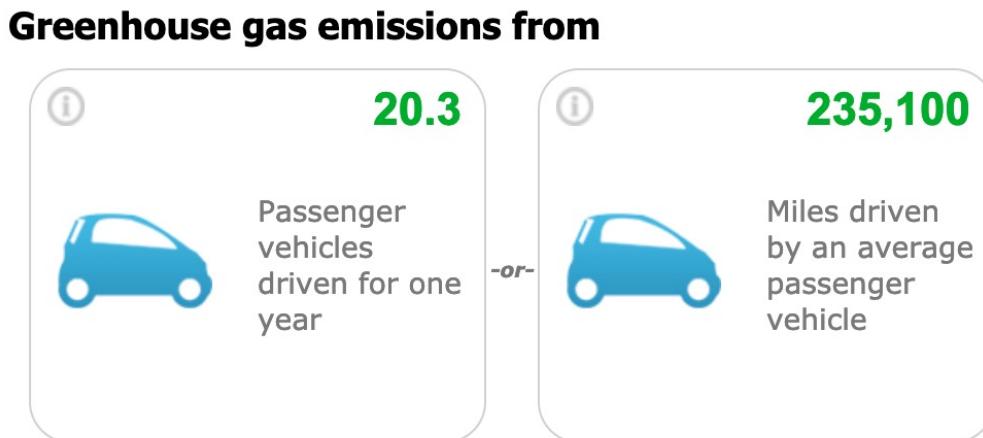
Practical tips

- Use pre-trained models
- Use battle-tested codeStart with already-running code and tweak it
- Consider other approaches besides GANs:
 - VAEs like <https://github.com/openai/dall-e> or Taming Transformers
 - <https://github.com/lucidrains/DALLE-pytorch>
 - CPPNs

Perspectives and Ethical Considerations

StyleGAN2's Energy Cost

The entire project, including all exploration, consumed 132 MWh of electricity, of which 0.68 MWh went into training the final FFHQ model. In total, we used about 51 single-GPU years of computation (Volta class GPU). A more detailed discussion is available in Appendix F.

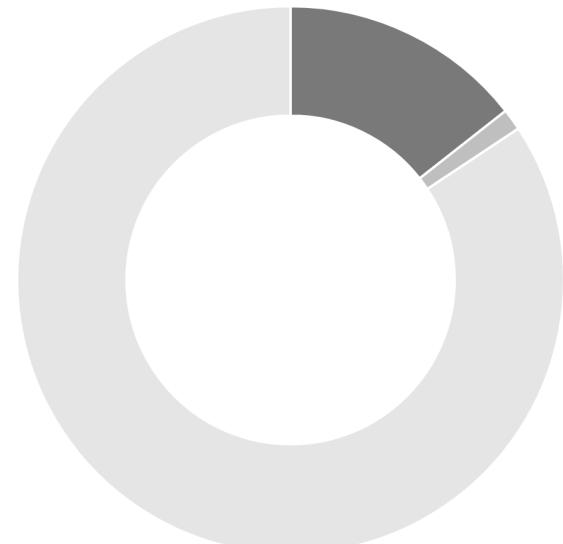


This Black Woman Does Not Exist

Below you see the first three “black” women generated in that session by StyleGAN, and one randomly selected “white” woman from the same session for comparison. The first image is less detailed, the following images is highly distorted; the final image (and notably a small variation of the other) is somewhat convincing, but unflattering.



- White Women
- Black Women
- Other



<https://www.intersectsy.com/intersectsy/2019/10/3/what-i-learned-about-bias-from-the-first-ai-generated-faces-on-wikimedia-commons>

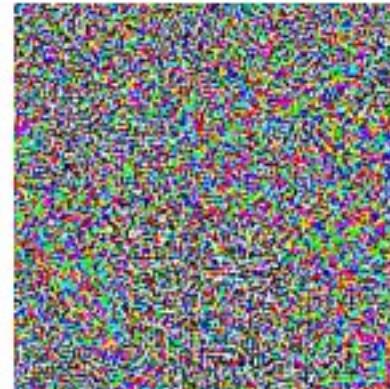
Adversarial Examples

- Generating an image to fool a classifier



\mathbf{x}
“panda”
57.7% confidence

+ .007 ×



$\text{sign}(\nabla_{\mathbf{x}} J(\boldsymbol{\theta}, \mathbf{x}, y))$
“nematode”
8.2% confidence

=



$\mathbf{x} +$
 $\epsilon \text{sign}(\nabla_{\mathbf{x}} J(\boldsymbol{\theta}, \mathbf{x}, y))$
“gibbon”
99.3 % confidence

Who owns the image? Who owns the model?

Other questions?