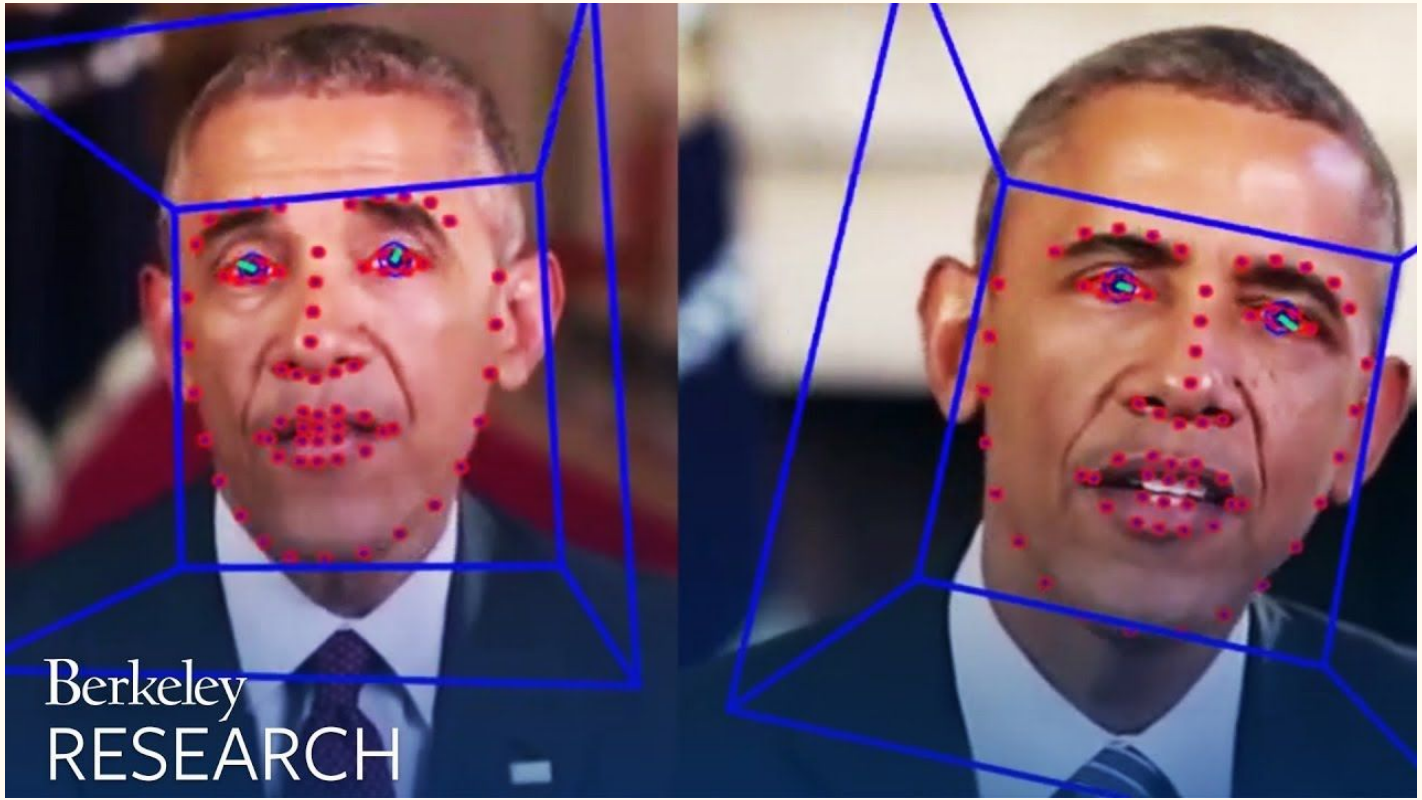
DeepRecognition

An exploration of attribute recognition in the quest to identify Deep Fakes.

Emily Brunelli, Andreana Chua, Adi Faintuch, Brooke Ryan

DeepFakes - What are they?

Real-World Applications of Attribute Recognition + DeepFakes



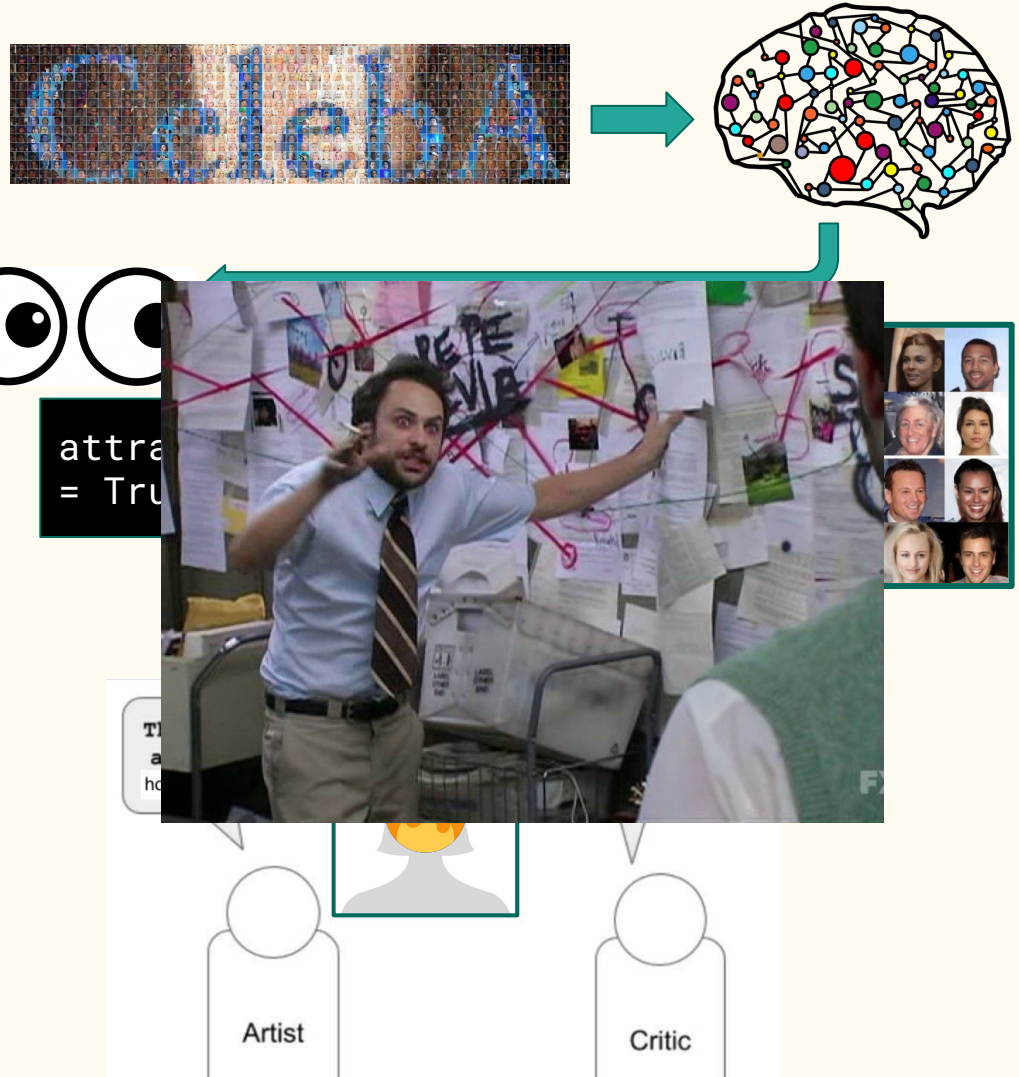
Real-World Applications of Attribute Recognition + DeepFakes

CelebA Dataset

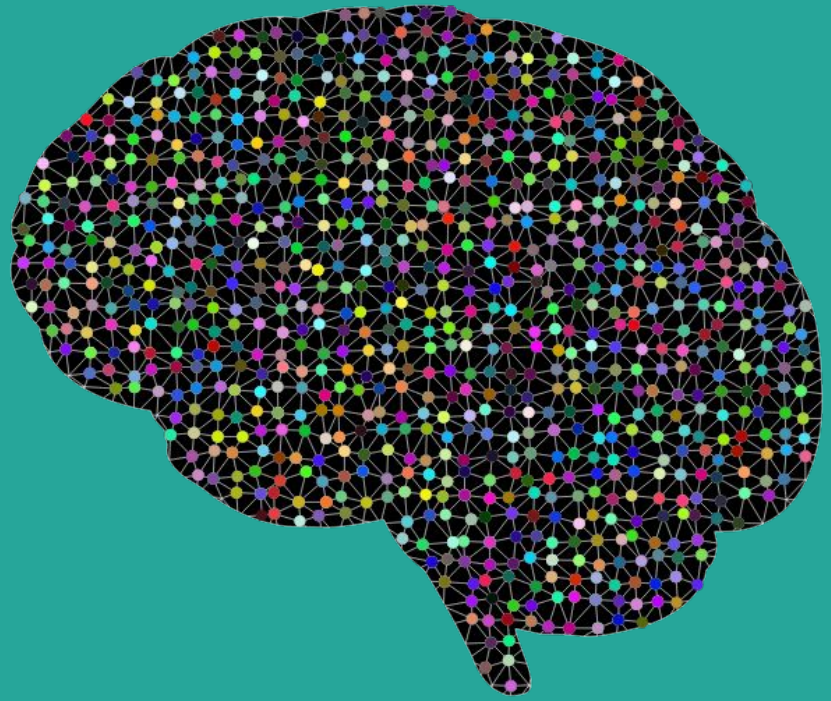
attractiveness?

Project Components

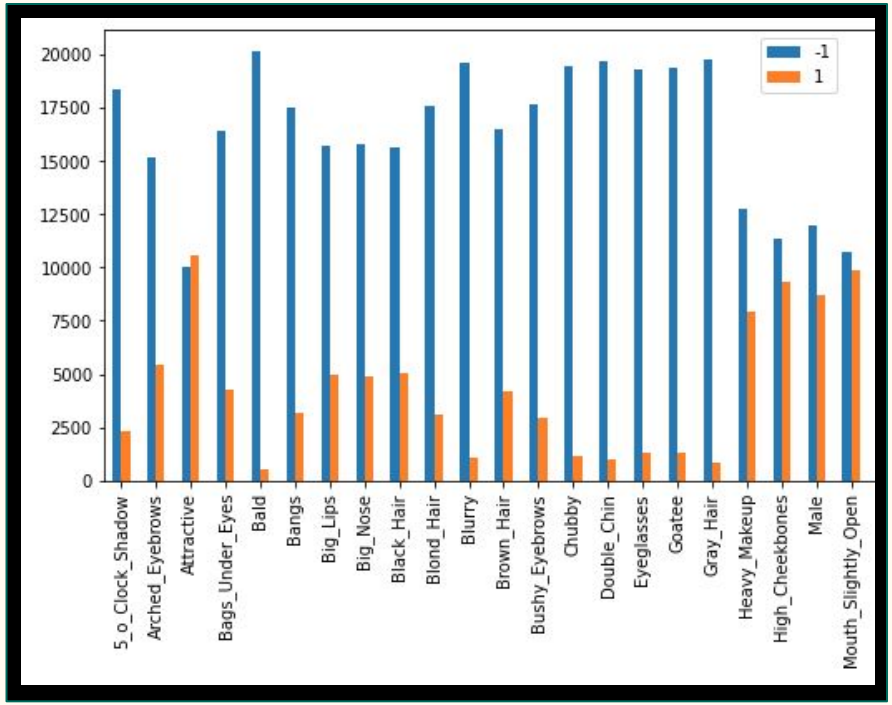
- CelebA Dataset
  - 40 attributes
  - Approx. 200,000 images
- Training of Convolutional NN to recognize specified attribute
  - Regular deep learning
  - Transfer learning
- Generative Adversarial Networks to create Deep Fake images / Deep Fake images of chosen attribute



Convolutional Neural Network/ Transfer Learning



Data Exploration and Preparation

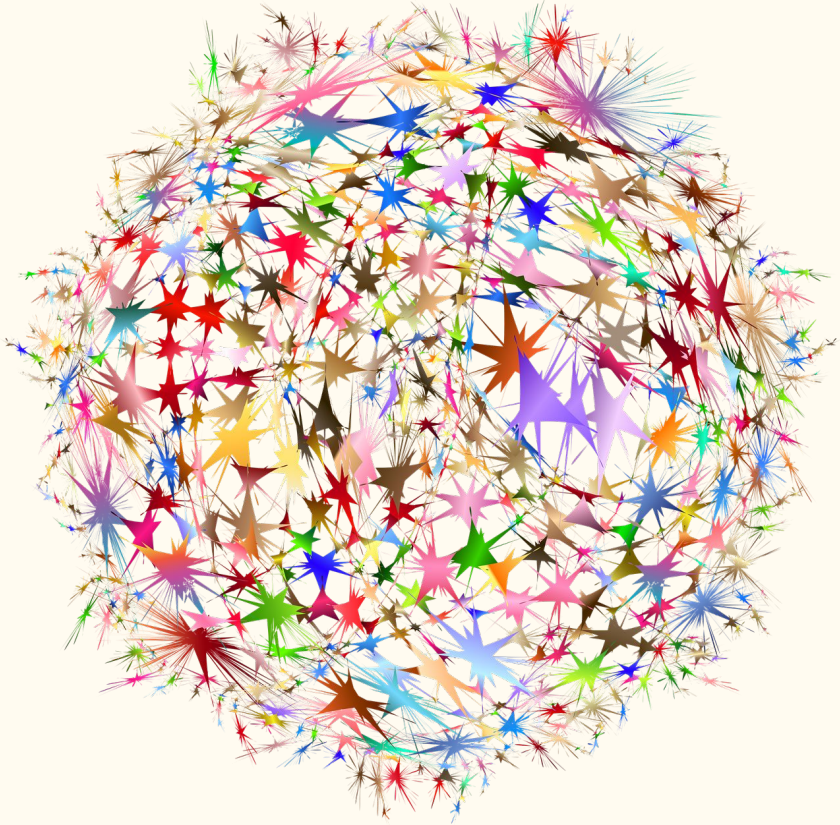


- Images
  - Split data into 60/20/20
  - Resized each image to 128x128x3
  - Data normalization
    - Min-max normalization
    - Divided features by 255

- Attributes
  - Total of 40 attributes
  - Binary attributes [-1, 1]
  - Chose “Attractive” to avoid imbalanced data

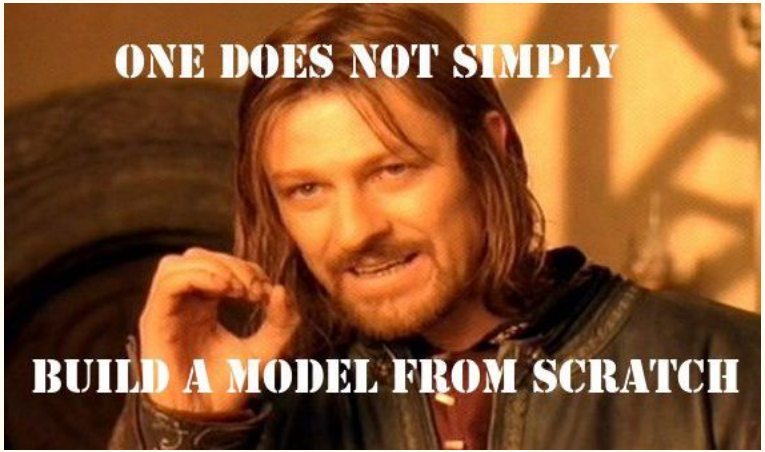
Building the CNN

- Most hyperparameters chosen from literature
  - Number of filters
  - Filter size
  - MaxPool size
  - Binary Cross-entropy loss
  - Adam optimizer
- Fine-tuned hyperparameter: Number of layers on validation set
  - Num layer: 1 → Accuracy: 0.7443
  - Num layer: 2 → Accuracy: 0.7580
  - **Num layer: 3 → Accuracy: 0.8460**
  - Num layer: 4 → Accuracy: 0.8142
- After training/testing
  - Accuracy: 0.8481



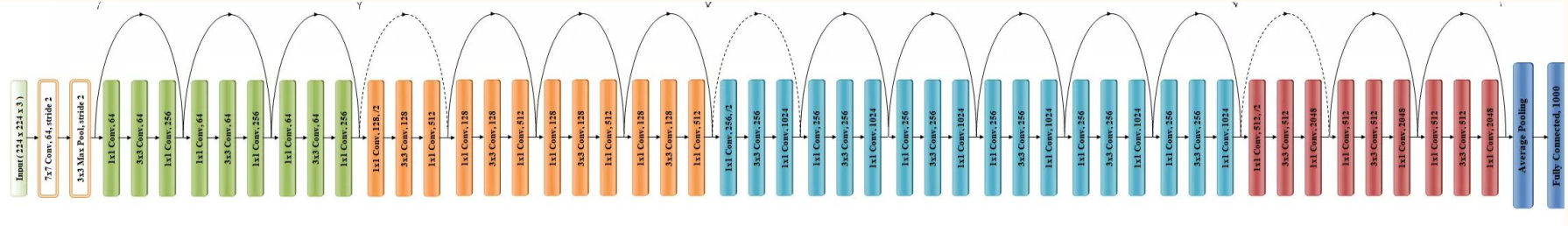
What is transfer learning?

- When a model for one task is used as a starting point of another task
- Pretrained model already trained on large datasets for a long period of time
  - Saves time on second task
- Especially useful for image-related tasks
  - Most models already learned common features of images
  - Beneficial if new dataset is significantly small
  - Just need a few more training iterations to fit new dataset



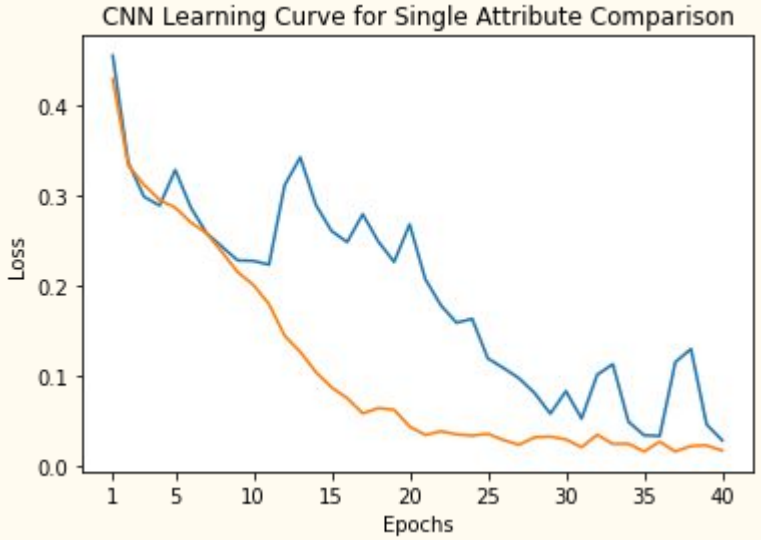
RESNET50

- 50 layers deep
- Built on stacking Residual Block
- Uses skip connections/shortcuts to jump over some layers
- Pretrained weights from training on >1,000,000 images from ImageNet, from 1000 image categories



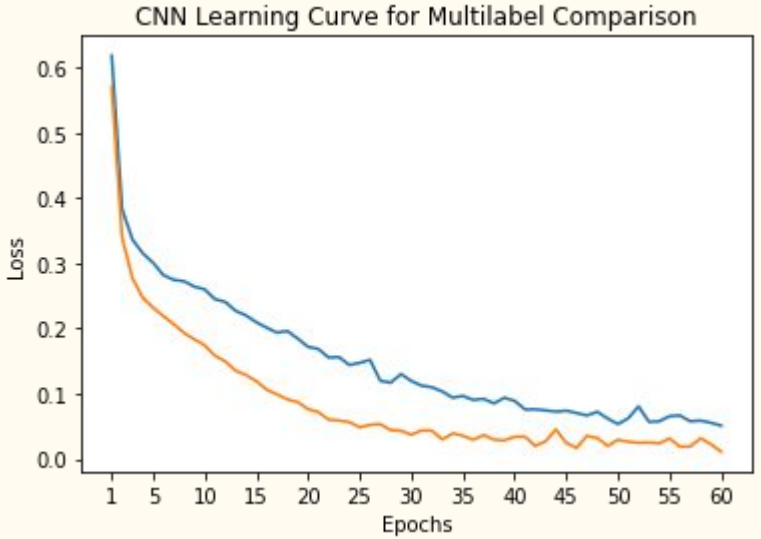
Our model

- Chopped off top layer to fit our images
- Chopped off bottom layer to add:
  - 3 FC Layers
  - 1 Output layer with 1 node
  - Used RELU for 3 layers, sigmoid for final
- Freezed the first 45 layers' weights
  - Trained all the other layers
  - Used dropout and batch normalization in last layers
- Accuracy: 0.8759 vs. 0.8481
- Better start and better asymptote

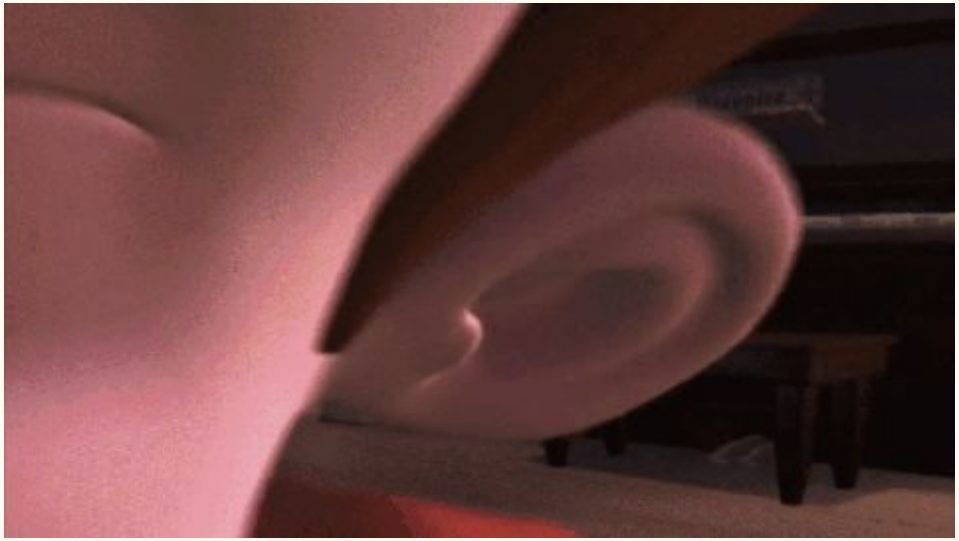


Multilabel accuracy

- Predicted heavy makeup, male, mouth slightly open, high cheekbones at once
- Used first 45 layers of Resnet50
- Identical except, softmax instead of sigmoid because multiclass classification
- Included batch normalization and dropout in-between layers
- Overall, better start and better asymptote for accuracy
- Accuracy of 0.634 on test vs. 0.591 for non-transfer learning



Project Component 1 Conclusions

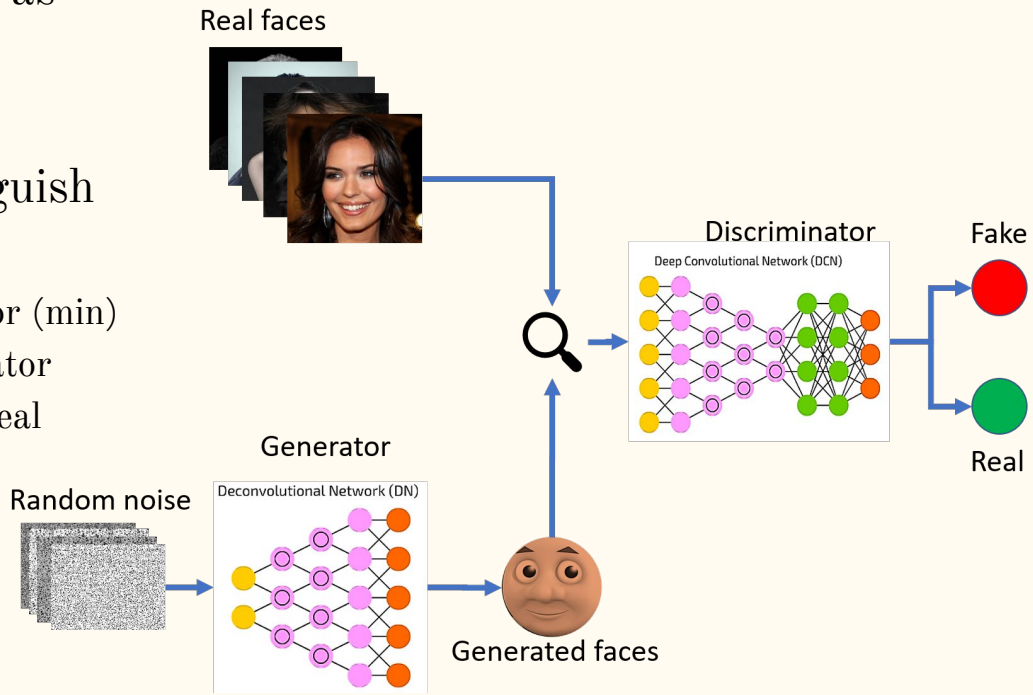


- Transfer learning accuracy: 0.8759
- Transfer learning significantly more accurate than regular deep learning
- Use of RESNET50 model successfully transfers to attribute facial recognition task

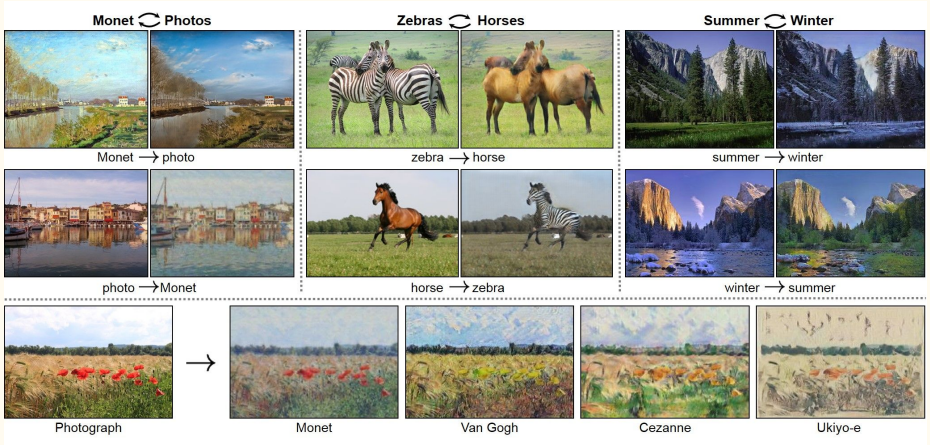
Generative Adversarial Networks

Using GAN to Generate Fake Faces

- Generator starts with random noise as the "fake image"
- Iteratively passes it into the discriminator which needs to distinguish between real and fake
  - min-max 'competition' between generator (min) and discriminator (max) until discriminator can't distinguish between the fake and real
    - optimal at 0.5 probability from discriminator



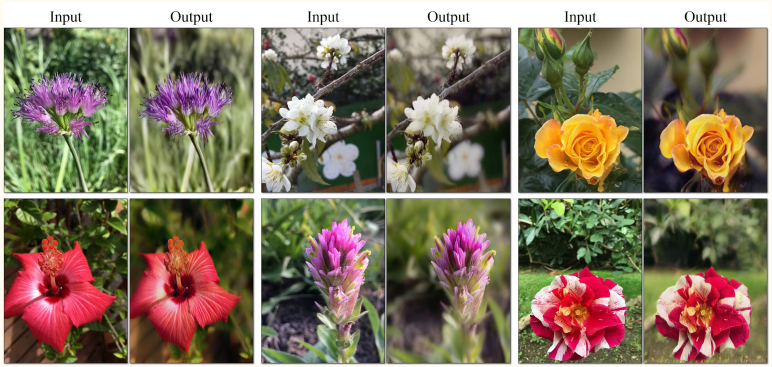
Same tech, different applications



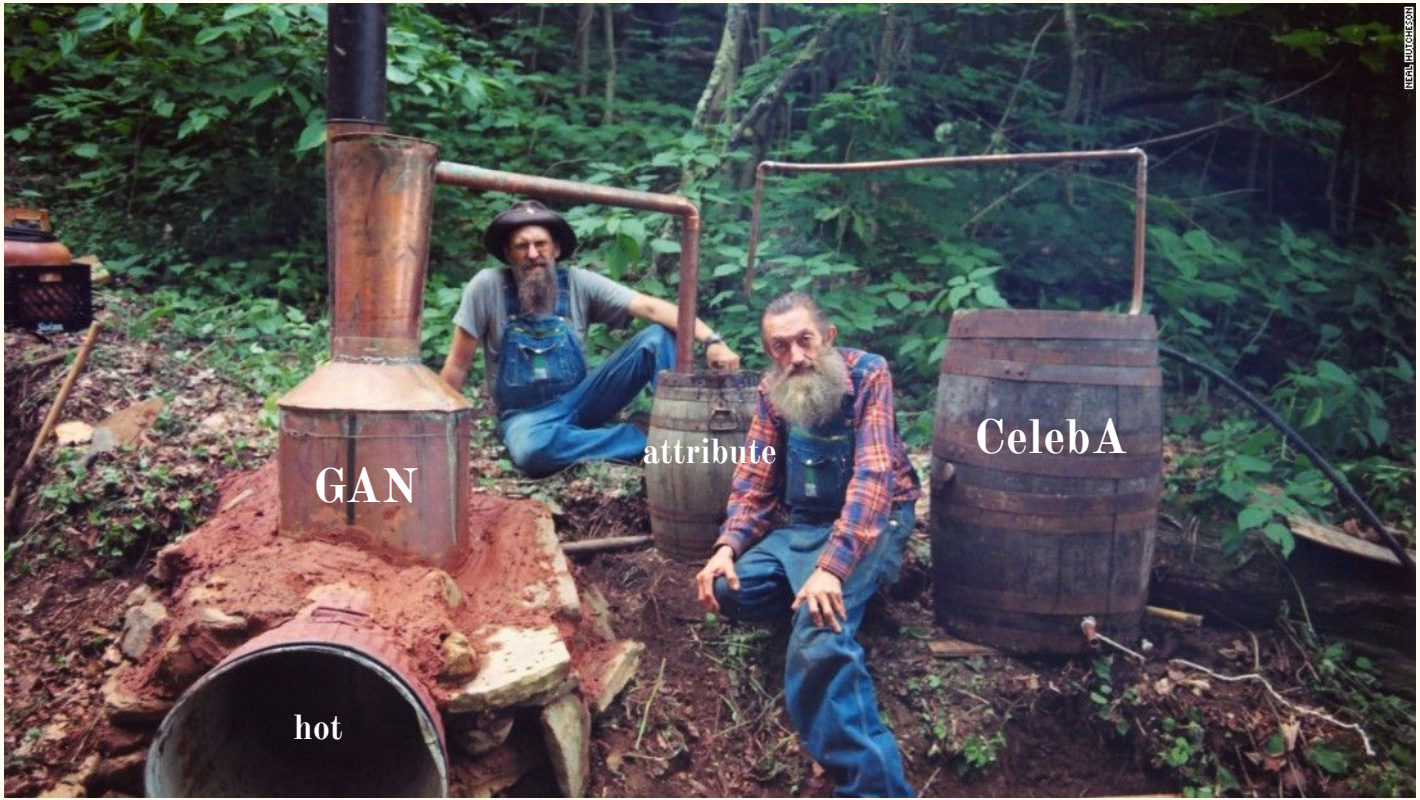
Examples:

- Summer → Winter image converter
- Monet image-style converter project
- Horse → Zebra
- Narrow depth-of-field, i.e. “portrait mode”
- Hot → not?

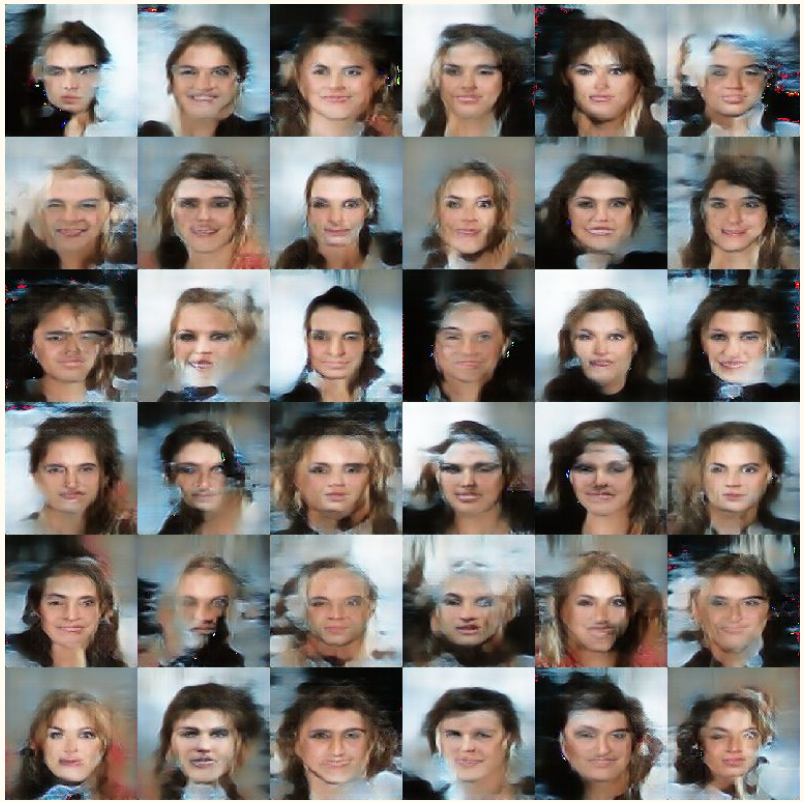
All use GANs!

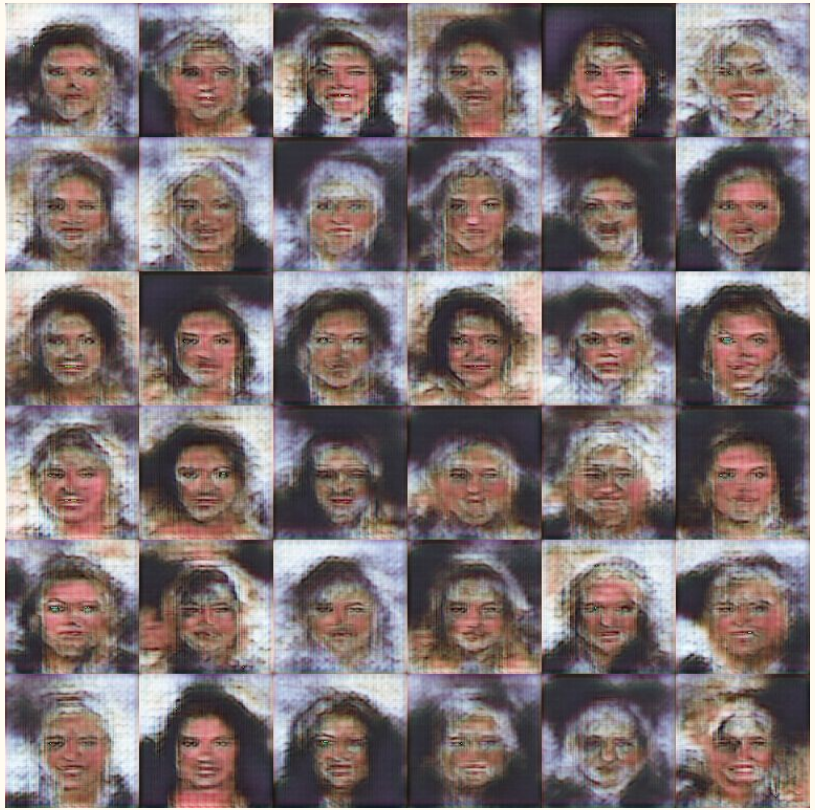


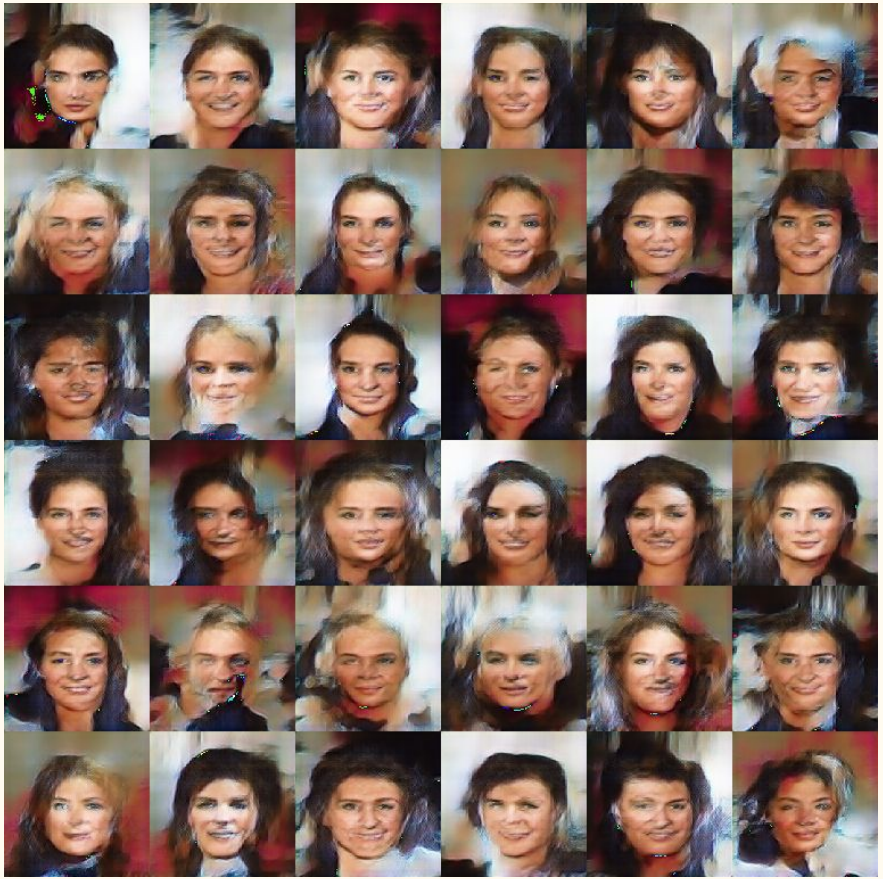
Home-grown Deep Fakes, anyone?



- Batch size: 16
- Step size: 50
    - Generates a face mosaic of 36 faces every 50 iterations



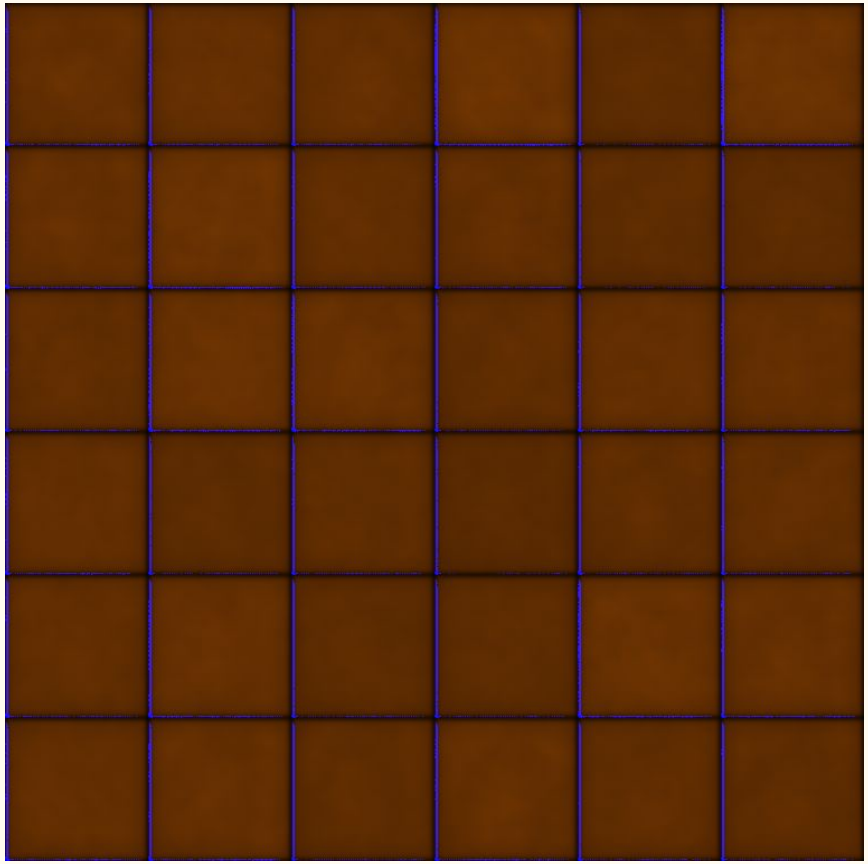


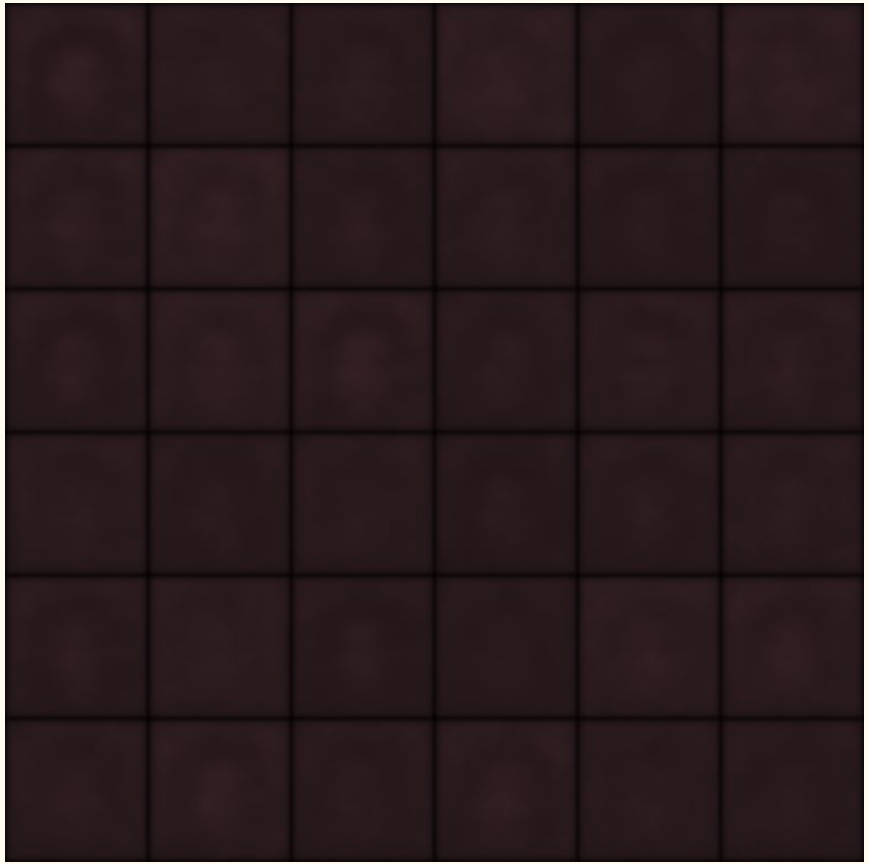


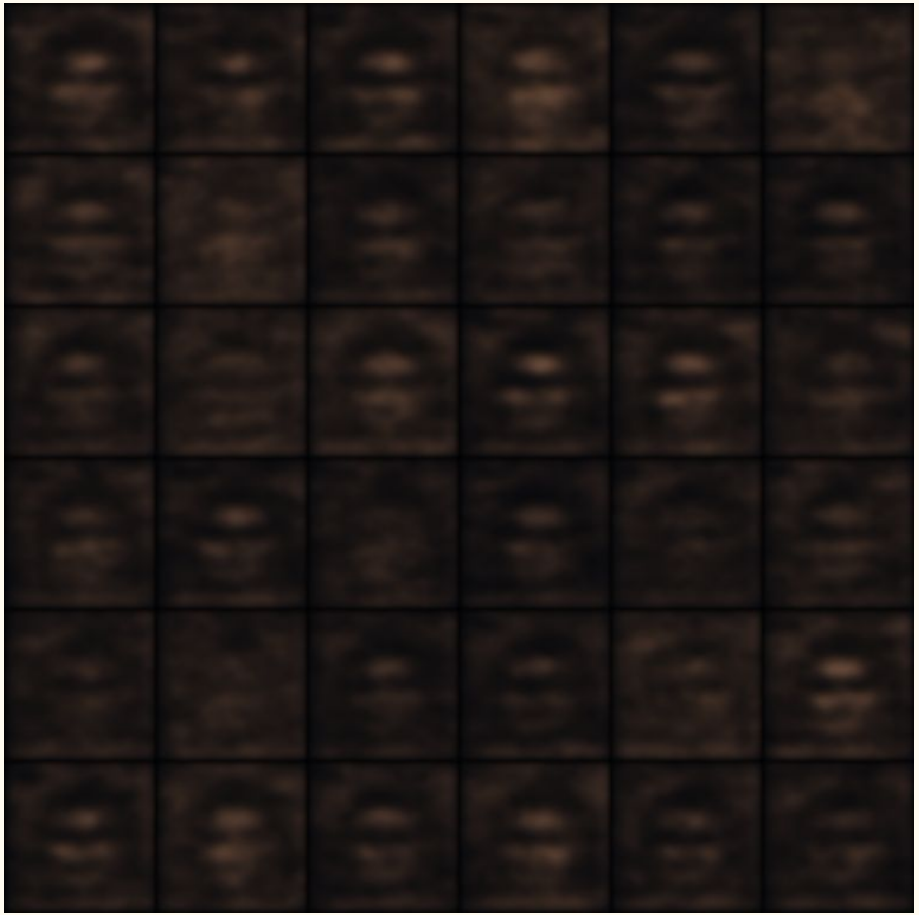
How do we generate attractive faces?

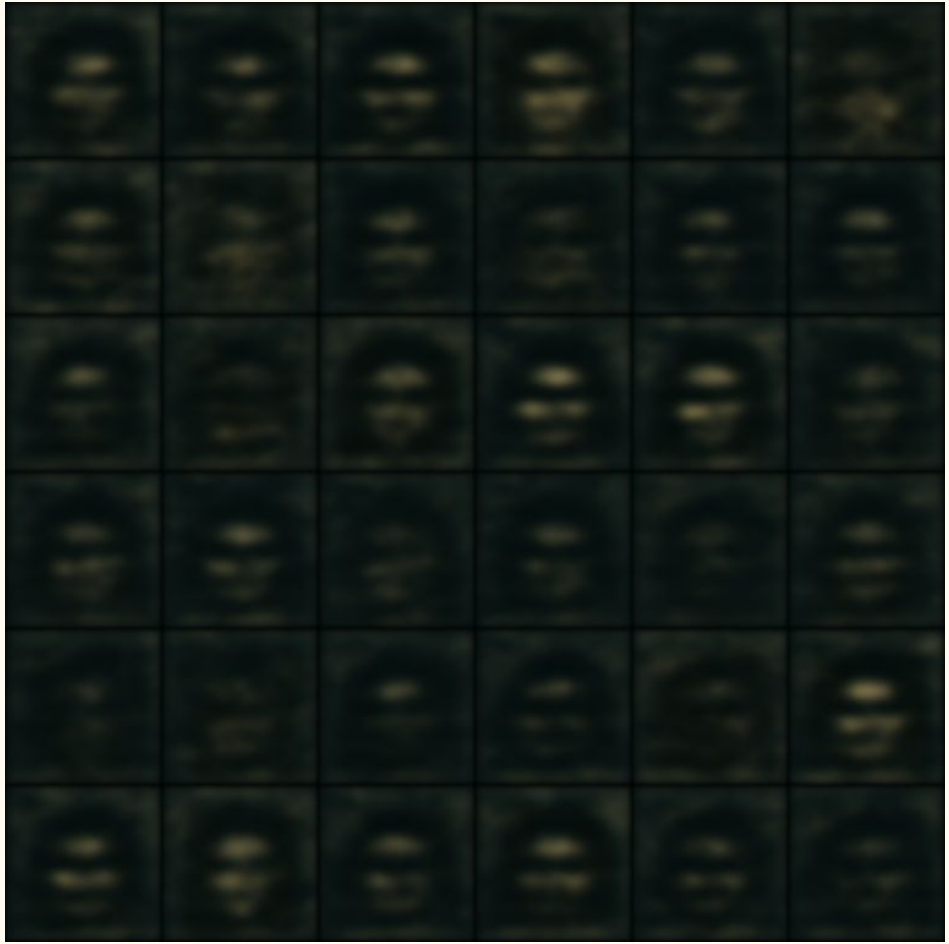
- Feed in face images to the GAN that have the attractive attribute

img_num	5_o_Clock_Shadow	Arched_Eyebrows	Attractive	Bags_Under_Eyes	Bald	Bangs
000001.jpg	-1	1	1	-1	-1	-1
000002.jpg	-1	-1	-1	1	-1	-1
000003.jpg	-1	-1	-1	-1	-1	-1
000004.jpg	-1	-1	1	-1	-1	-1
000005.jpg	-1	1	1	-1	-1	-1
000006.jpg	-1	1	1	-1	-1	-1
000007.jpg	1	-1	1	1	-1	-1
000008.jpg	1	1	-1	1	-1	-1

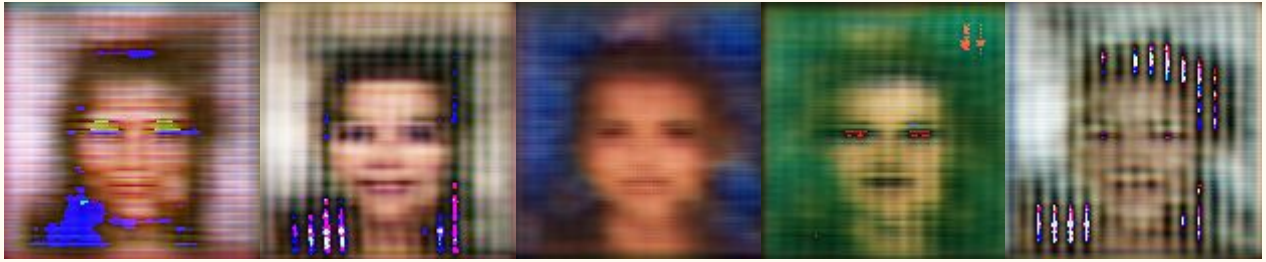






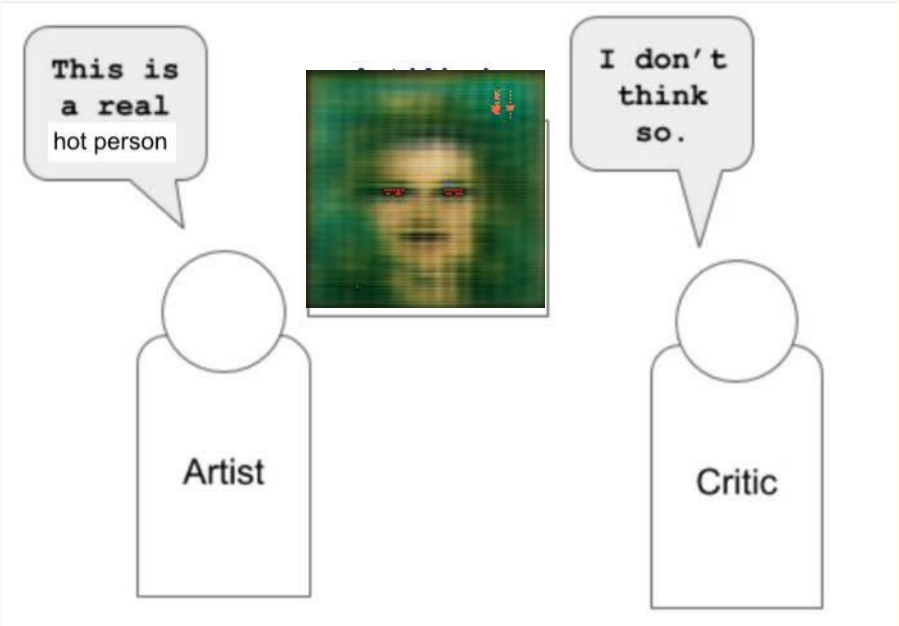


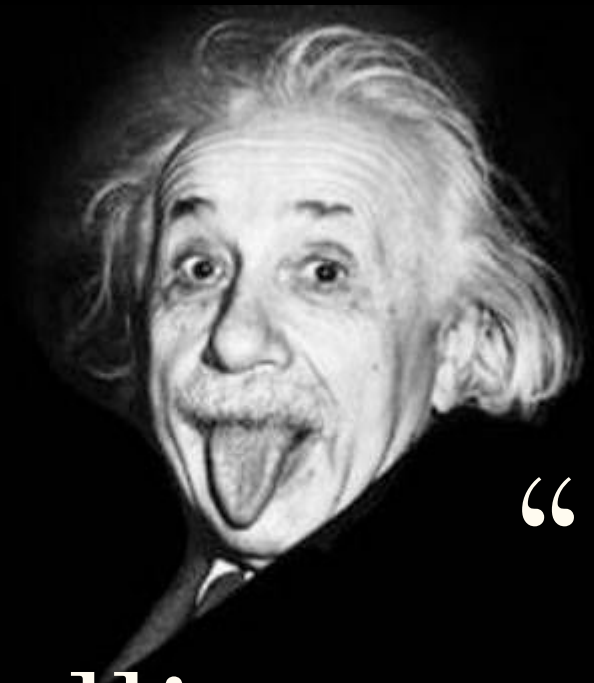
1000 iterations



Conclusions

- DeepFake creation and detection = computational “arms race” metaphorically represented by generator and discriminator concept
- Limitations of physical hardware
- What *is* attractiveness?





“Creativity is intelligence having fun”
- Albert Einstein

The end! Q&A