

The Rest of the Owl

Image Generation Using Conditional GANs
by Joshua Gottlieb

How to draw an owl

1.



2.



1. Draw some circles

2. Draw the rest of the fucking owl

Data Acquisition

Roughly ~30,000 images scraped from:

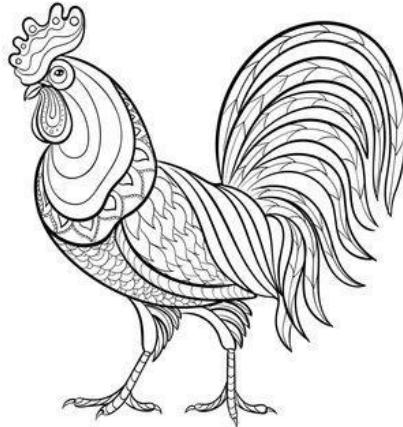
- Adobe Stock
- Fine Art America
- VectorStock

2381 images manually selected.

80/20 split into train and test.



Owl





Ground Truth



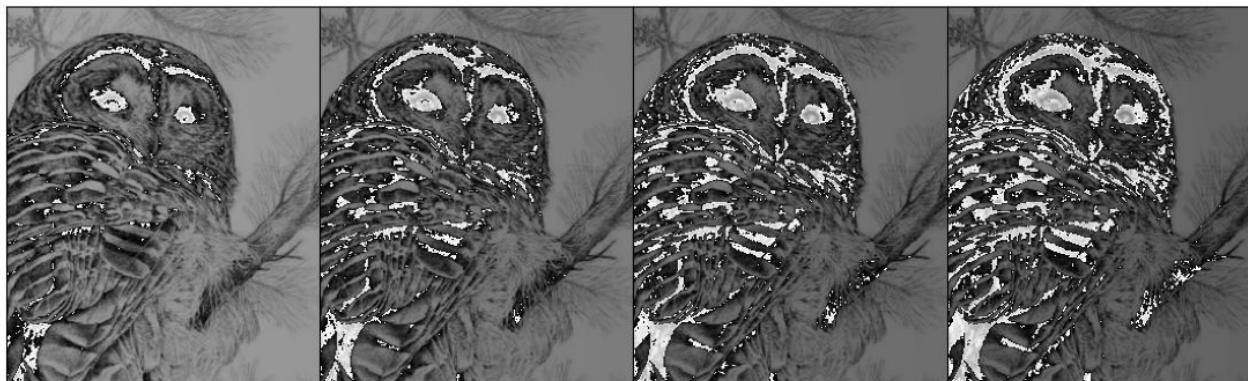
Weak Blur

- γ



Strong Blur

—

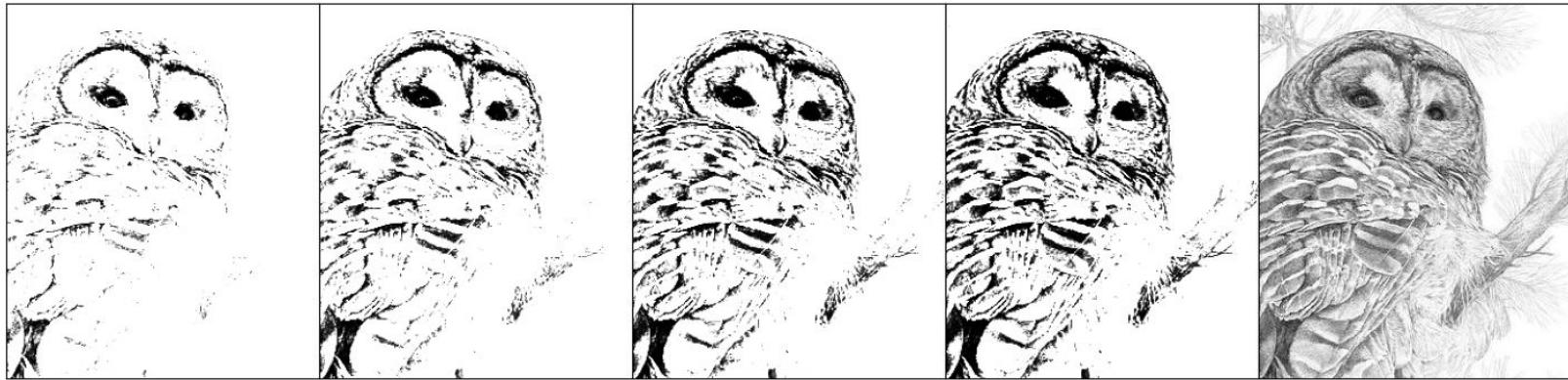


Gamma: 0.48

Gamma: 0.6

Gamma: 0.67

Gamma: 0.72



Threshold 0.01
Gamma: 0.48

Threshold 0.03
Gamma: 0.6

Threshold 0.05
Gamma: 0.67

Threshold 0.07
Gamma: 0.72

Ground-Truth



Threshold 0.01
Gamma: 0.32

Threshold 0.03
Gamma: 0.4

Threshold 0.05
Gamma: 0.47

Threshold 0.07
Gamma: 0.53

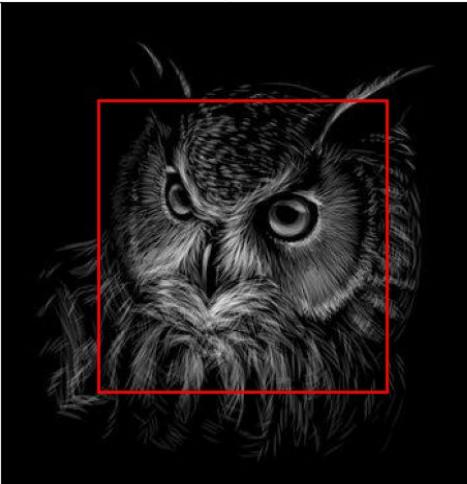
Ground-Truth

Inverted vs. Regular

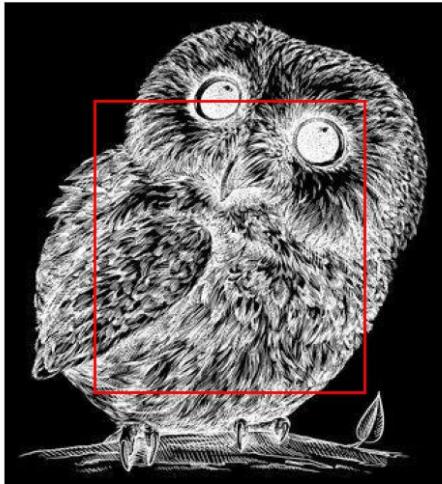
Inverted



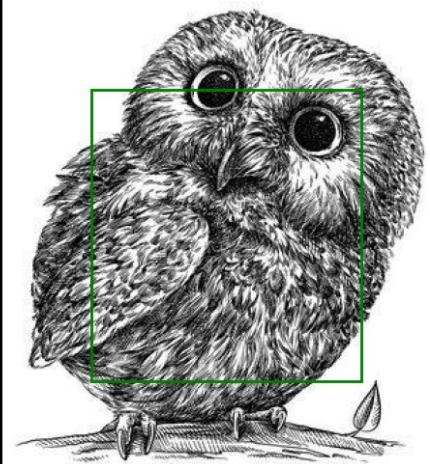
Regular



Inverted



Regular



BB%: 0.0000

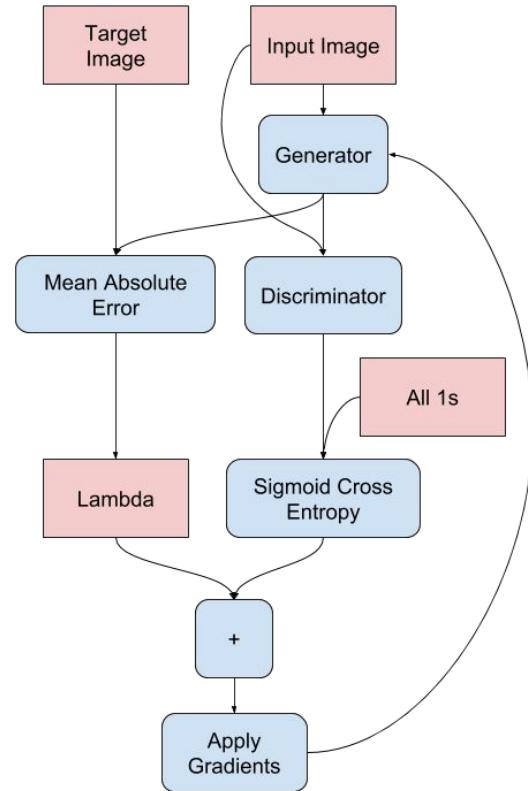
BB%: 0.9602

BB%: 0.6769

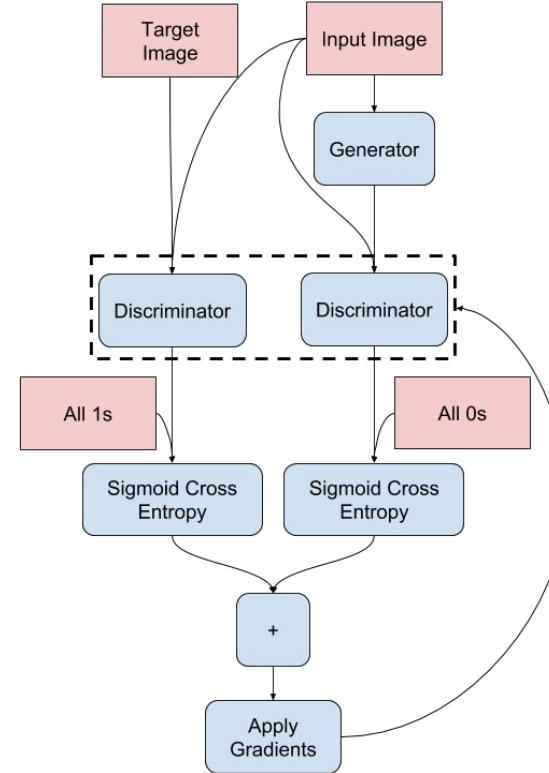
BB%: 0.0556

~55% of images regular, ~45% of images inverted

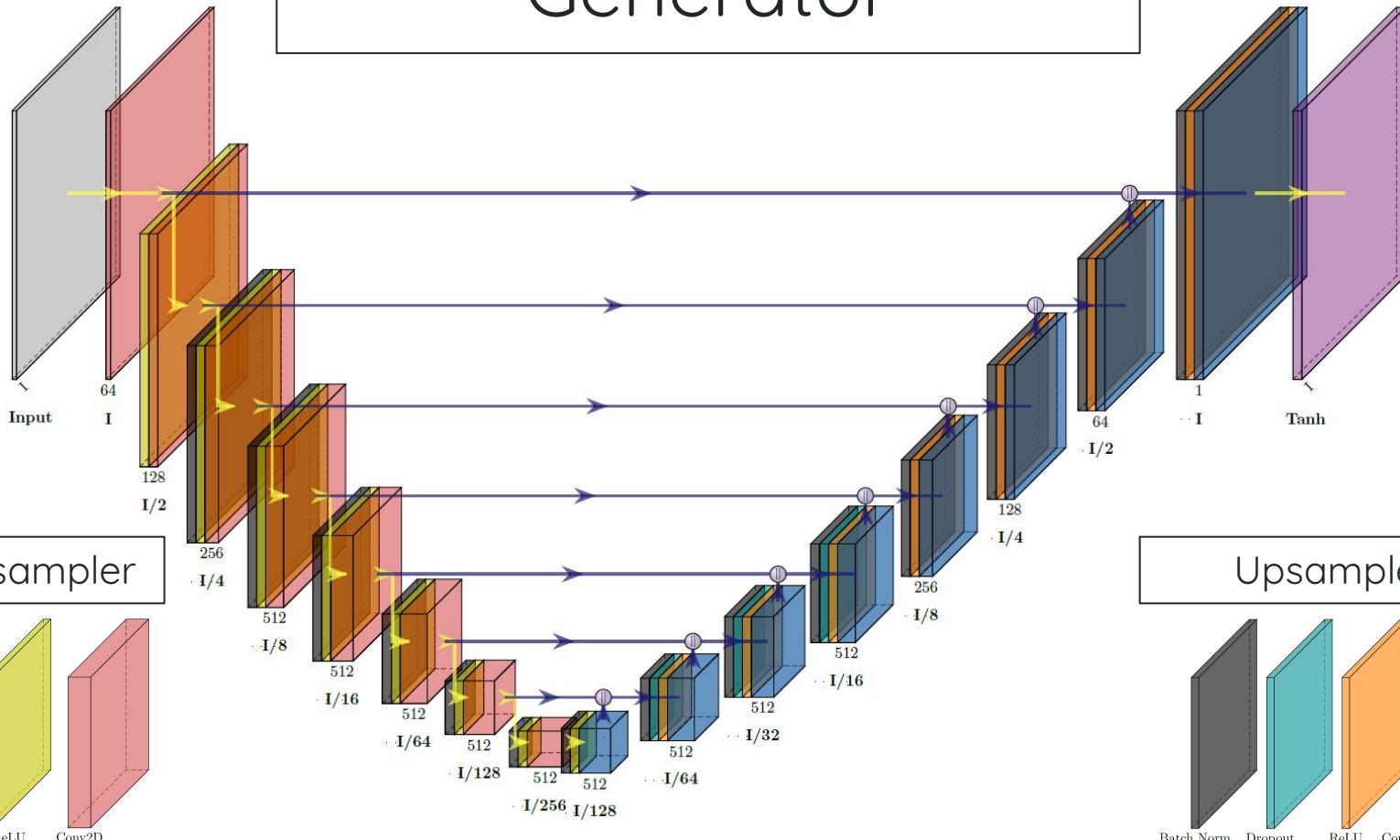
Generator Training Loop



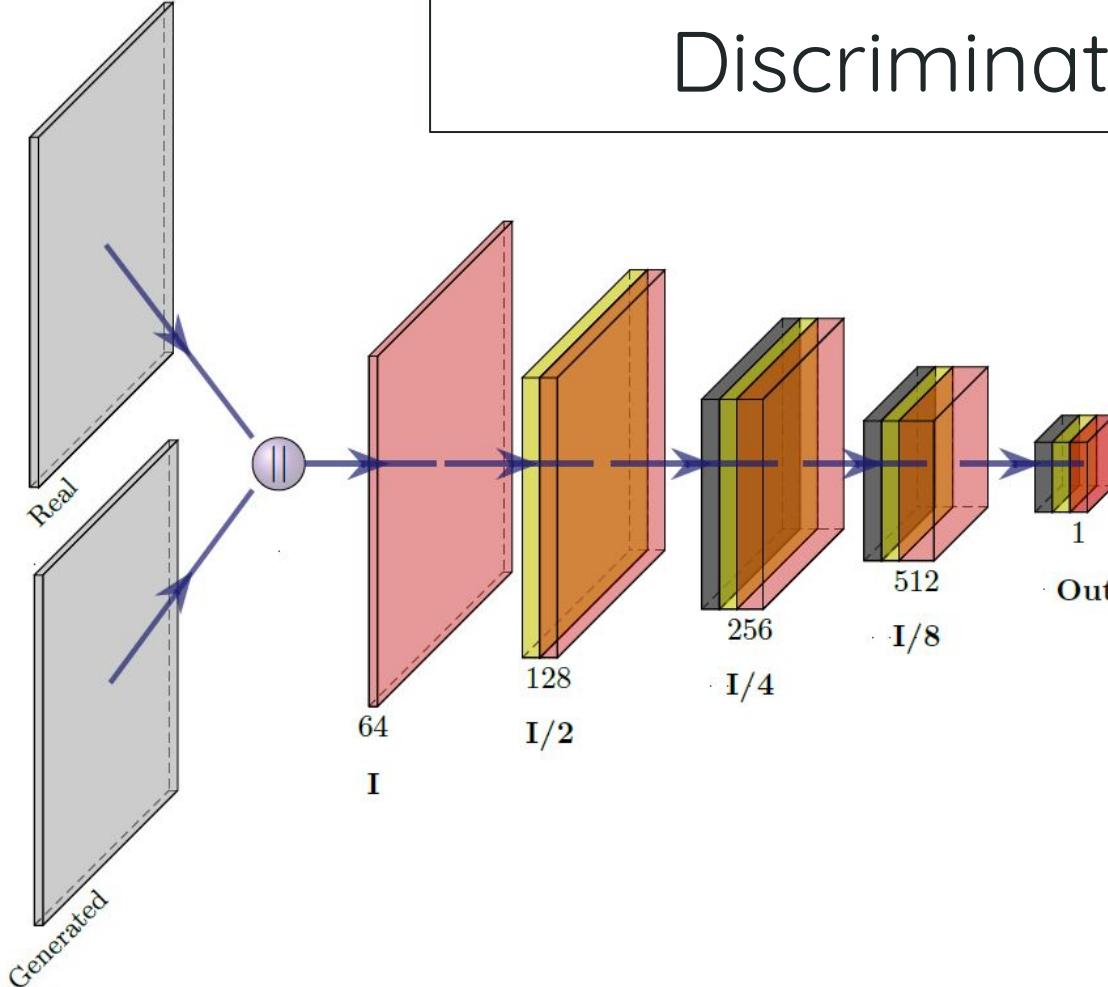
Discriminator Training Loop



Generator



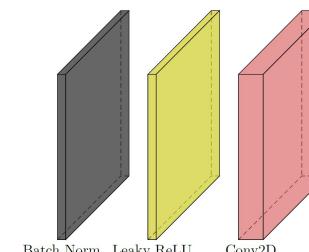
Discriminator



Output is an image of size (29, 29).

Binary Cross Entropy computes Sigmoid.

Downsampler



pix2pix Loss Functions

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,y} [\ln D(x, y)] + \mathbb{E}_{x,z} \left[\ln \left(1 - D(x, G(x, z)) \right) \right]$$

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z} [\|y - G(x, z)\|_1] \quad \mathcal{L}_{pix2pix} = \mathcal{L}_{cGAN} + 100 \cdot \mathcal{L}_{L1}$$

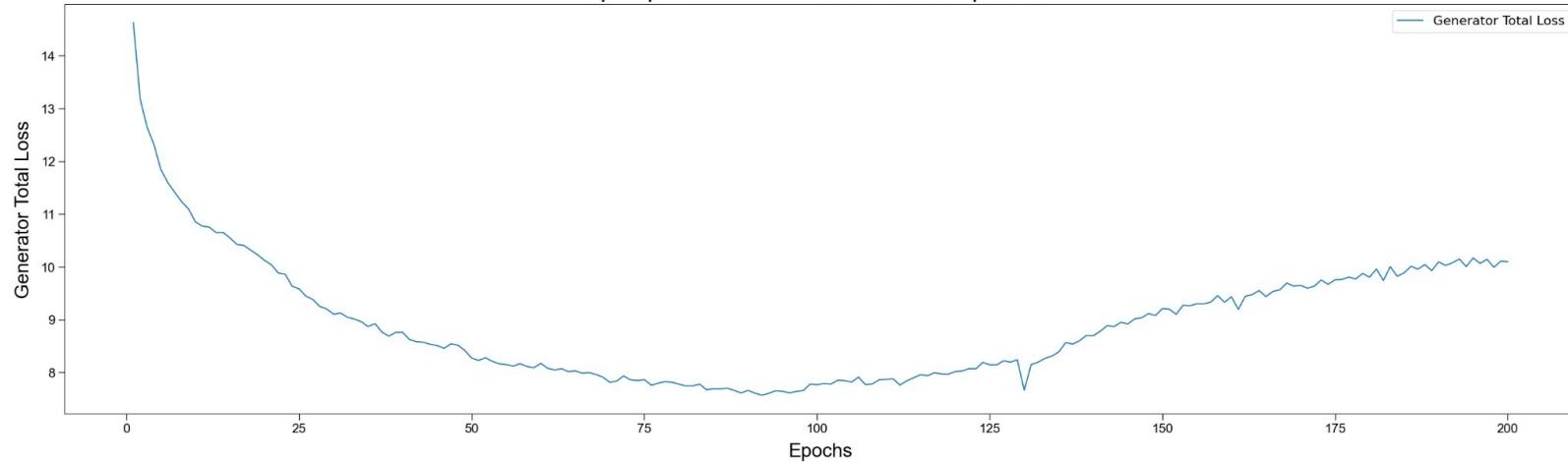
Autopainter Loss Functions

$$\mathcal{L}_{feature}(G, \phi) = \mathbb{E} [\|\phi_j(y) - \phi_j(G(x, z))\|_2] \quad \text{Using VGG16, Conv3_3}$$

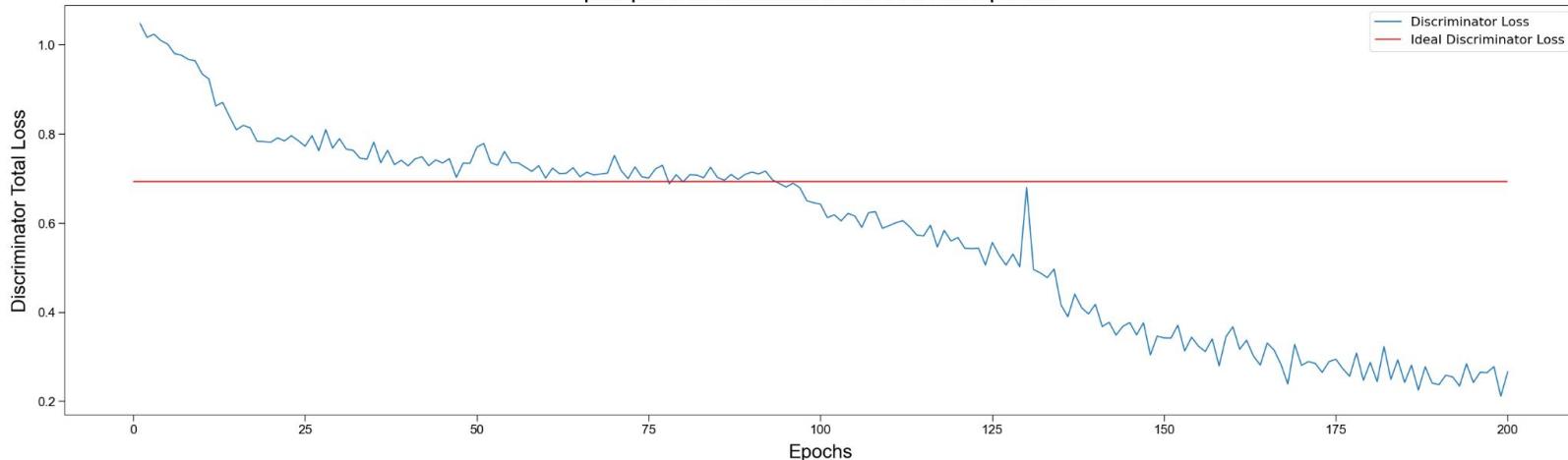
$$\mathcal{L}_{tv}(G) = \sqrt{\left(G(x, z)_{i+1,j} - G(x, z)_{i,j} \right)^2 + \left(G(x, z)_{i,j+1} - G(x, z)_{i,j} \right)^2}$$

$$\mathcal{L}_{autopainter} = \mathcal{L}_{cGAN} + 10 \cdot \mathcal{L}_{L1} + 10^{-5} \cdot \mathcal{L}_{tv} + 10^{-4} \cdot \mathcal{L}_{feature}$$

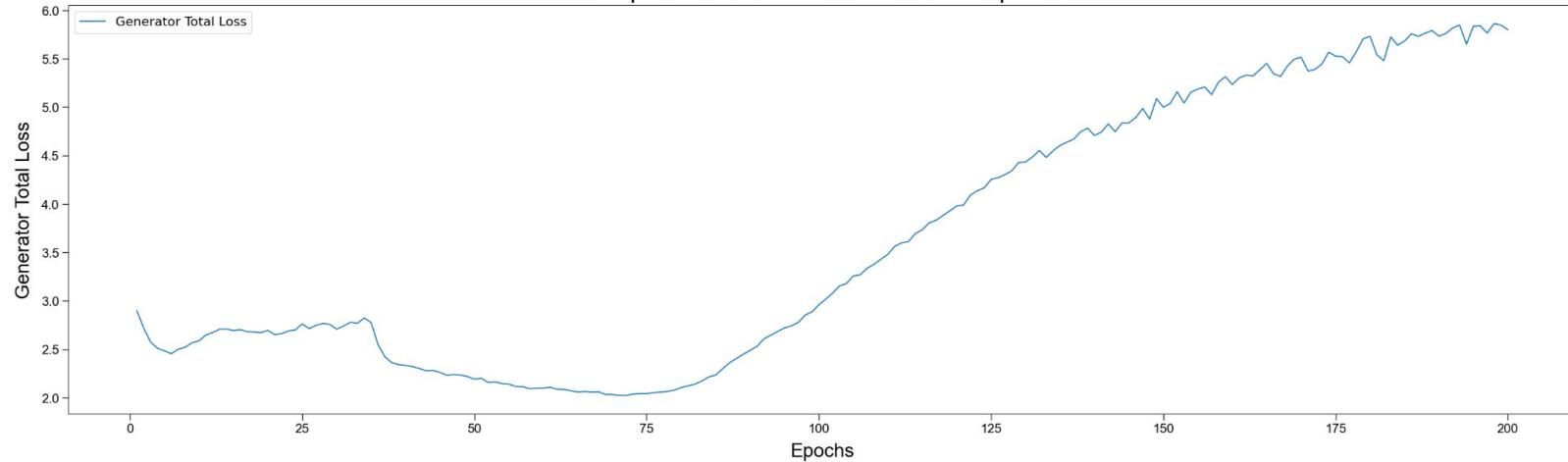
pix2pix Generator Loss Across Epochs



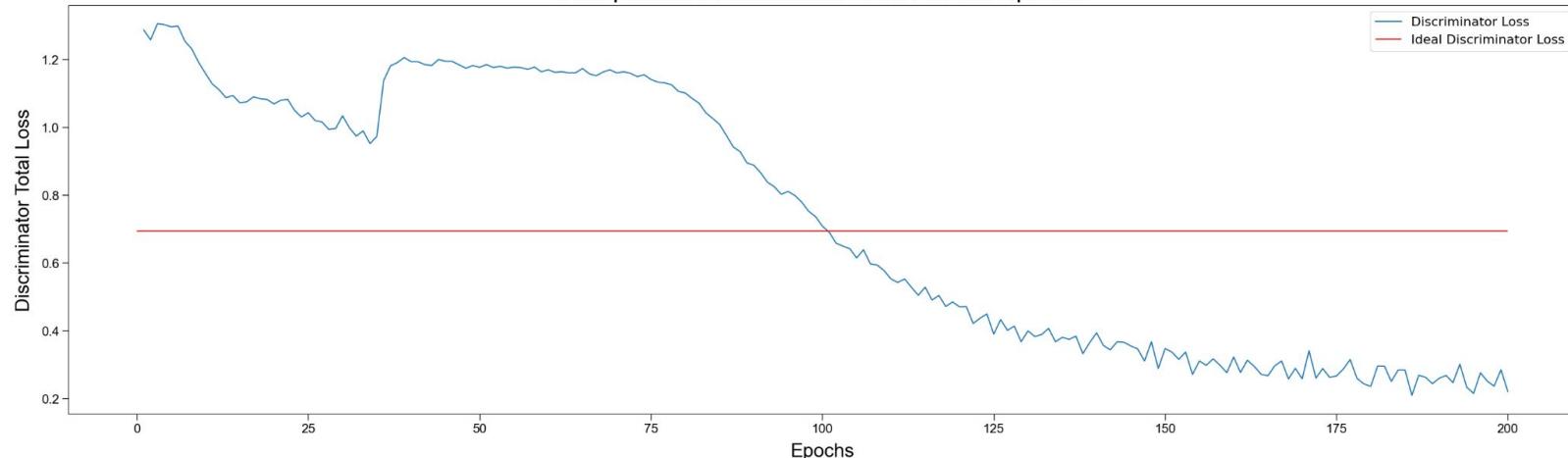
pix2pix Discriminator Loss Across Epochs



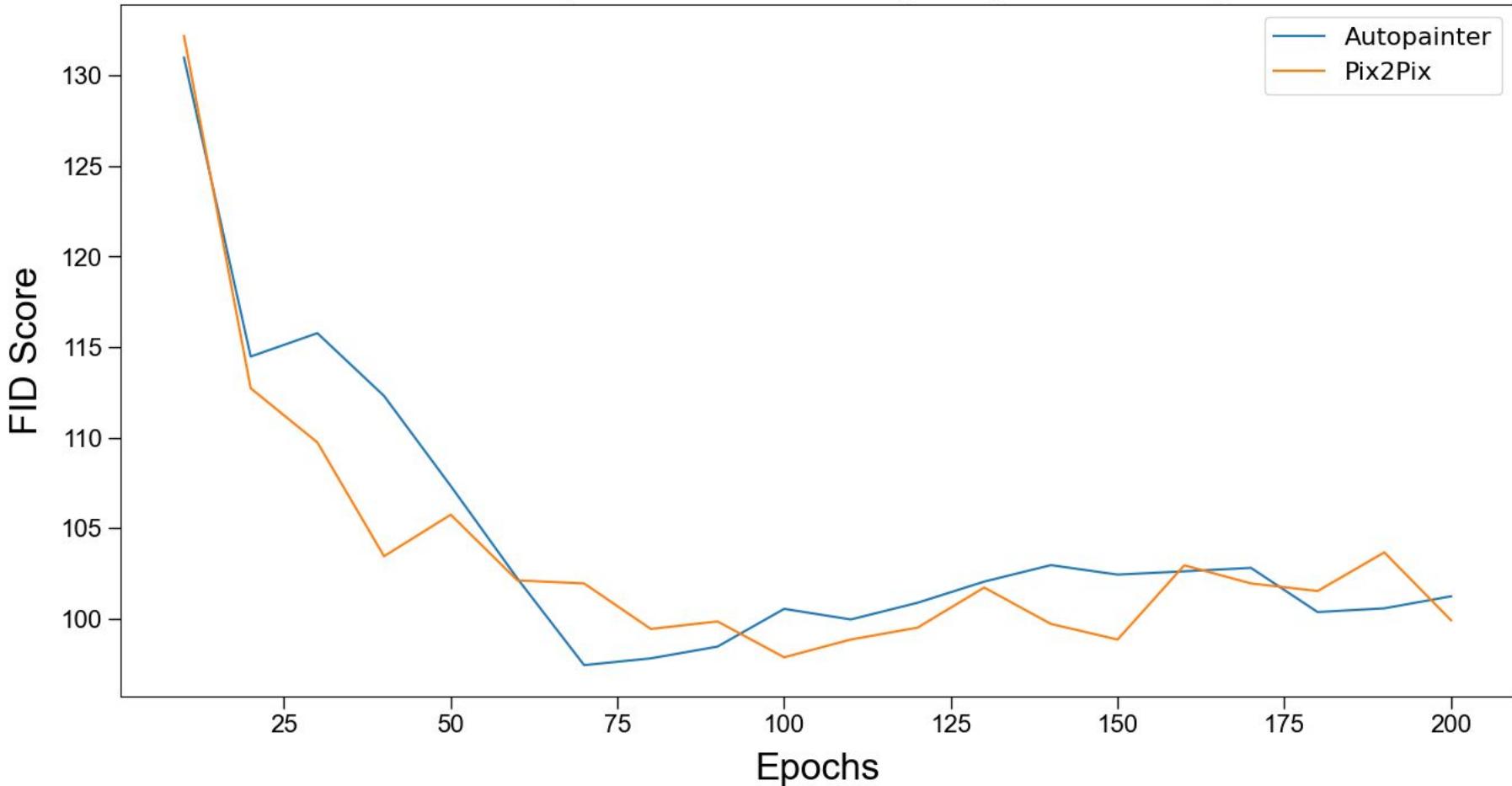
Autopainter Generator Loss Across Epochs



Autopainter Discriminator Loss Across Epochs

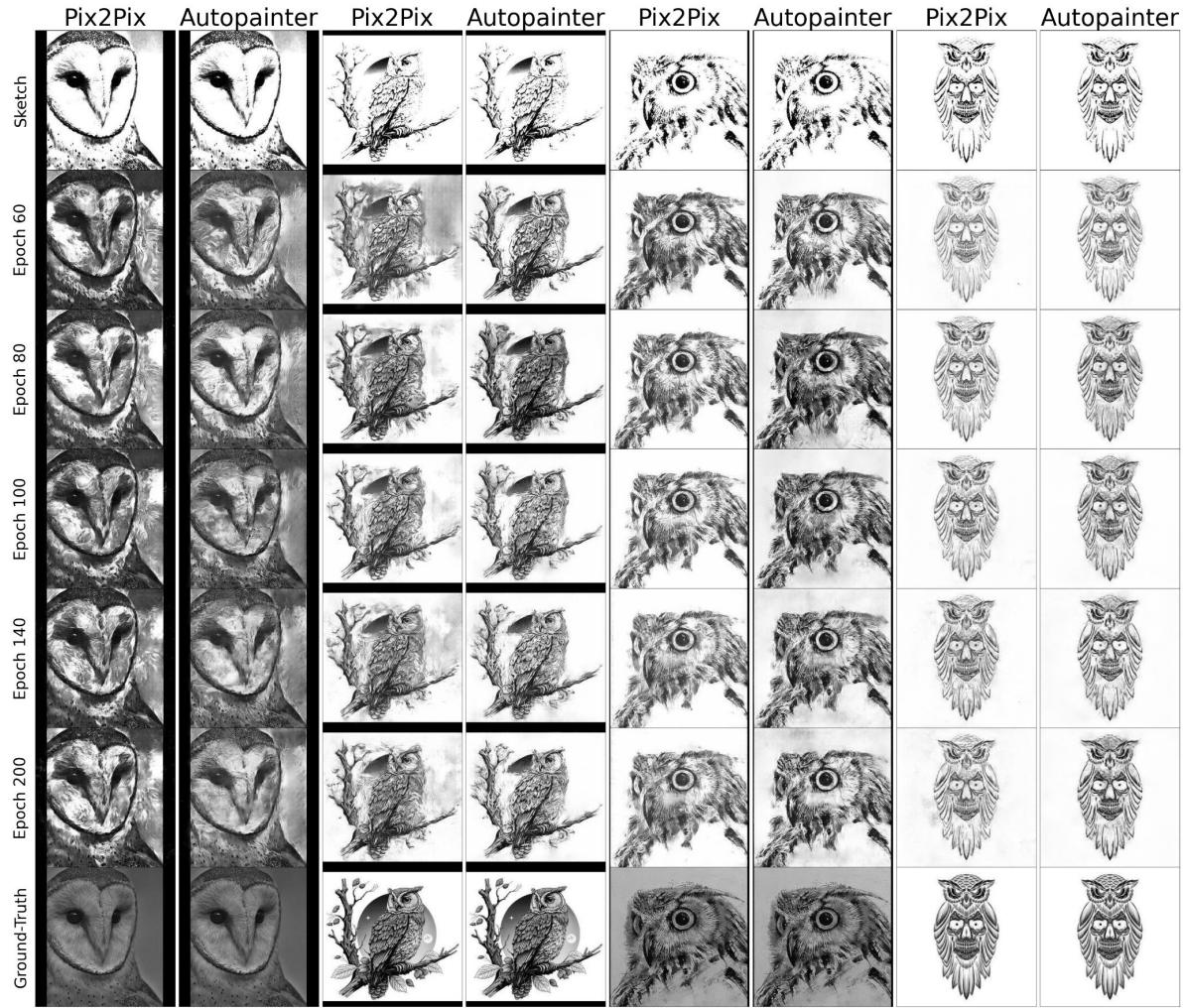


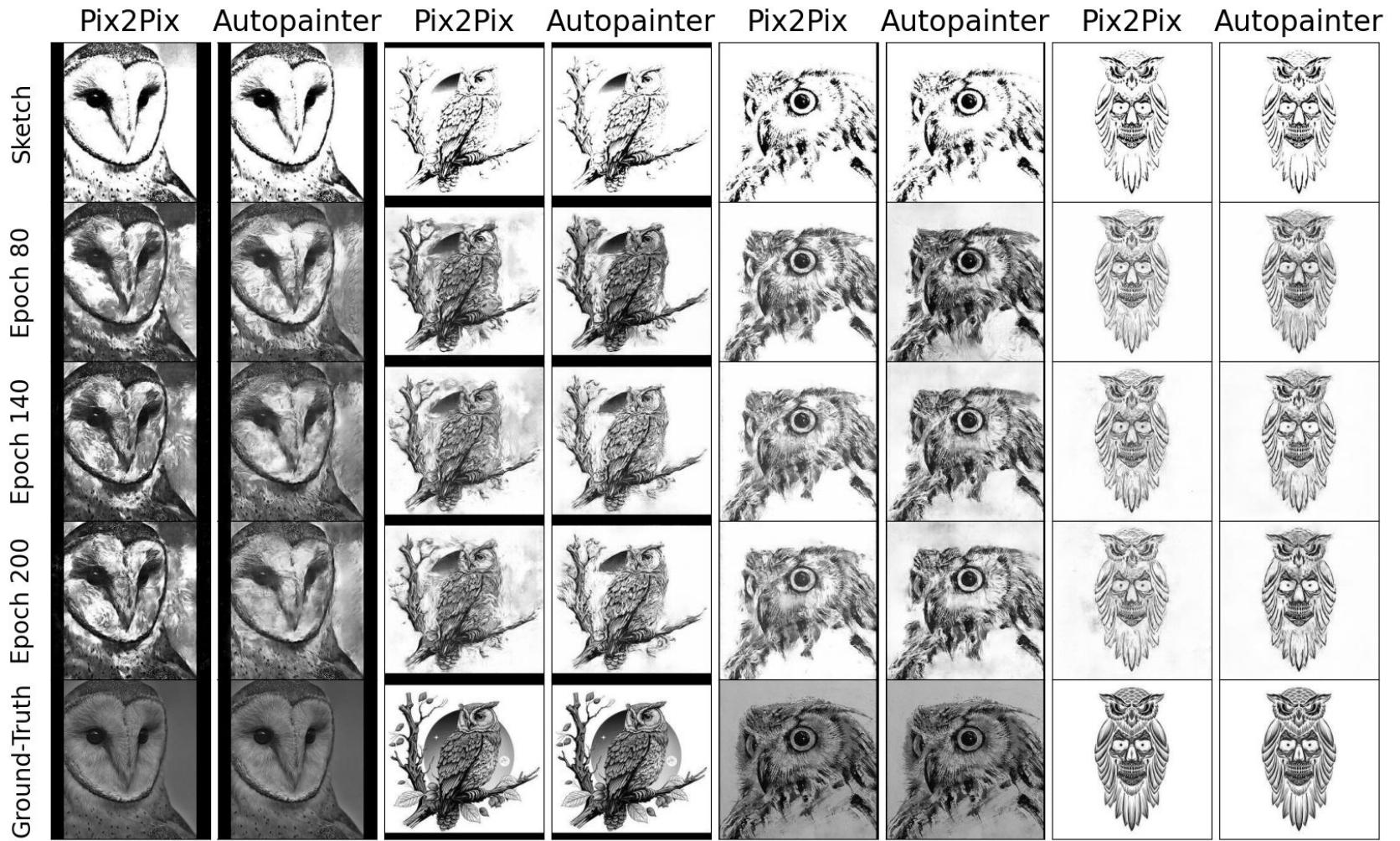
Frechet Inception Distance (FID) Across Epochs

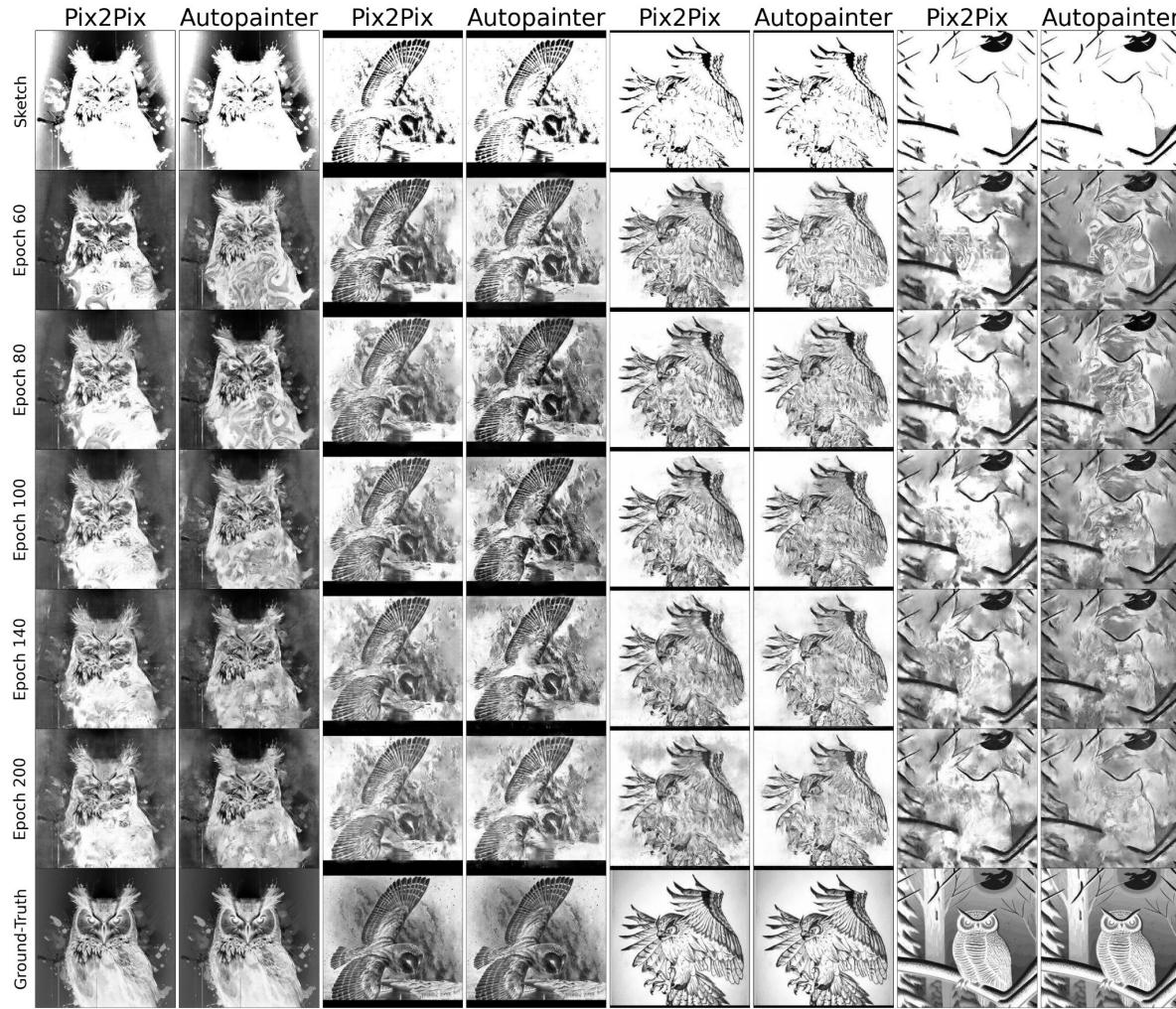


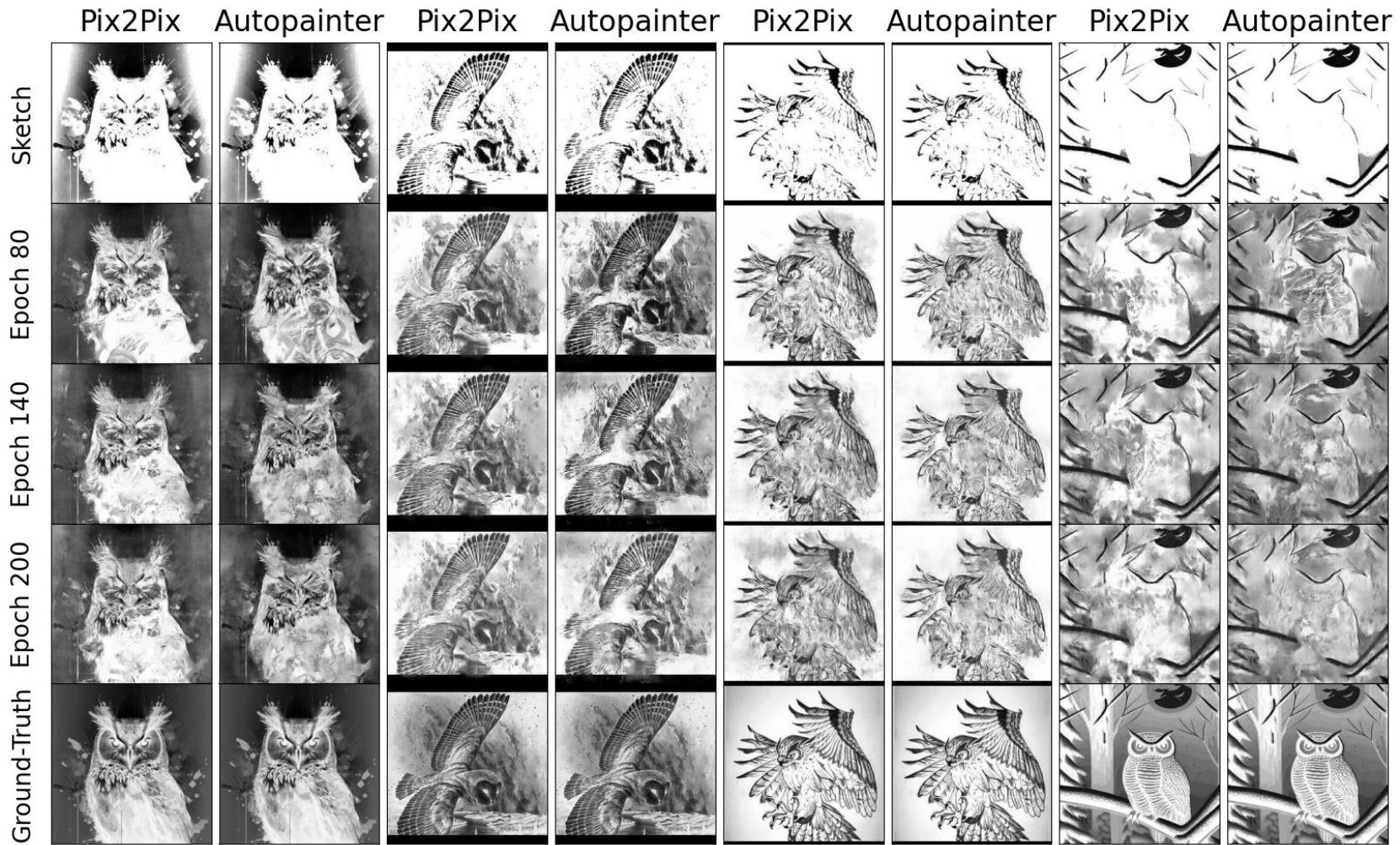
Models and Scores - FID Scores

| | 30 Epochs | 60 Epochs | 80 Epochs | 100 Epochs | 140 Epochs | 180 Epochs | 200 Epochs |
|-------------------------------------|-----------|-----------|---------------|---------------|------------|------------|------------|
| pix2pix (Best at 100) | 109.733 | 102.114 | 99.433 | 97.868 | 99.711 | 101.529 | 99.914 |
| Autopainter (Best at 70) | 115.766 | 102.215 | 97.809 | 100.543 | 102.958 | 100.366 | 101.236 |









Pix2Pix Autopainter Pix2Pix Autopainter Pix2Pix Autopainter



Things to Test in the Future

- Use more data, and further normalize data
- Test different batch sizes (used batch size of 1)
- Tune roughly a dozen hyperparameters

Thank you for listening!

Presenter: Joshua Gottlieb

Email: joshuadavidgottlieb@gmail.com

LinkedIn: linkedin.com/in/joshua-gottlieb

GitHub: github.com/JoshuaGottlieb