DENOISING DIFFUSION IN CONJUNCTION WITH VARIATIONAL AUTOENCODERS

Initial Model

- What is Denoising Diffusion?
- What is Guided Diffusion? (https://arxiv.org/abs/2212.07501)
- Guided Diffusion Github licensed under MIT license

Hyperparameters

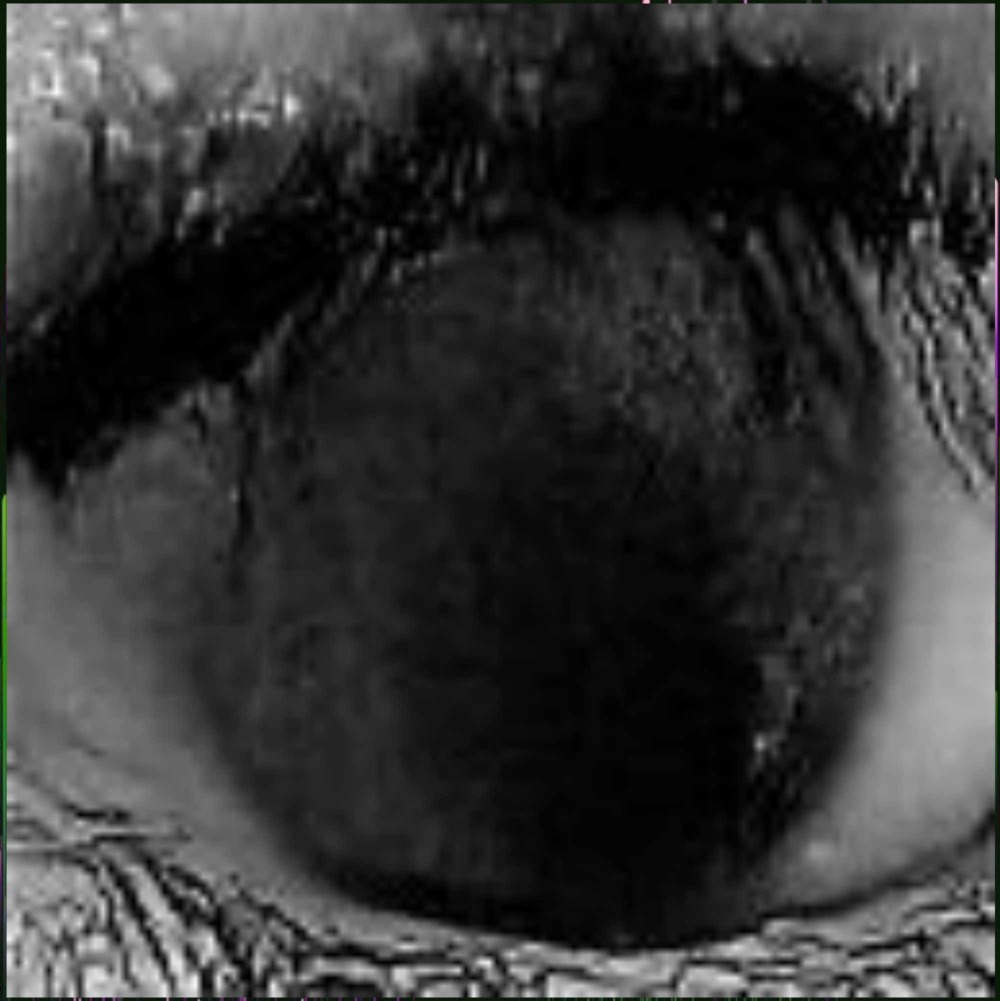
- Image Size=64x64
- Num Channels=128
- Skip Blocks=3
- Diffusion Steps=4000
- Noise Schedule=Linear
- Learning Rate=0.0001
- Batch Size=32
- Epochs=2000

RESULTS

Hyperparameters-2

- Image Size=128x128
- Num Channels=192
- Skip Blocks=4
- Diffusion Steps=4000
- Noise Schedule=Linear
- Learning Rate= 0.0001
- Batch Size=120
- Micro Batch=3
- Epochs=600

RESULTS



Problem with Results

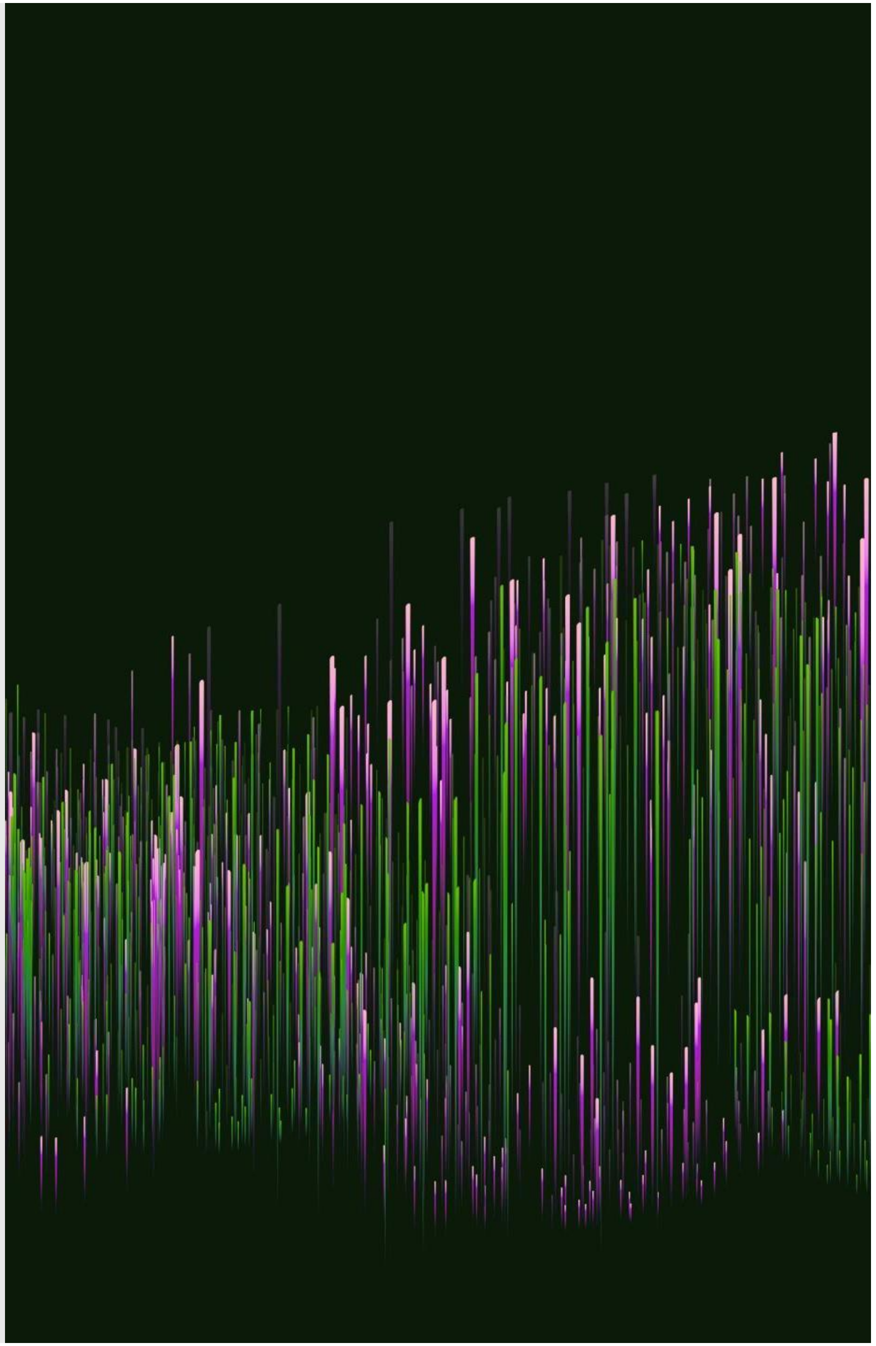
The results do not have the Pupil in the right shape and the black regions of the pupil flow into the Iris severely hampering the quality.

Solution: VAEs

- What are VAEs?
- What is Keras?
- Keras is classified under Apache License
- Apache License allows modification and redistribution

Combining the pipelines

- The vanilla guided diffusion model uses pure noise to generate images with maximum variance
- The vanilla model was designed to be ideal to produces images where shape isn't an absolute necessity like faces
- However, the Iris and Pupil need to be perfectly circular ideally
- VAEs can produces perfect shapes but lack the attention to detail, whereas Diffusion models are perfect to produce sharp textured results
- We pass the results from VAEs into the denoising pipeline to fix the shape instead of relying on pure noise and generate intricacies using denoising diffusion



Legalities

- We have used an open-source strong copyleft license, The GPL-v2 License (Formerly known as the GNU Public License)
- This license allows the modification and redistribution of code wherever needed without any responsibilities to the owners of this code, provided all the modifications made to this source code is made publicly available
- This is the same license as the Linux Kernel

Disadvantage to this approach

Though this approach greatly improves sampling stability and efficiency, the training becomes even heavier as now we have to train two models and the guided diffusion model needs to be perfect with identified the Iris texture as we no longer can rely on the randomness of noise to produce texture

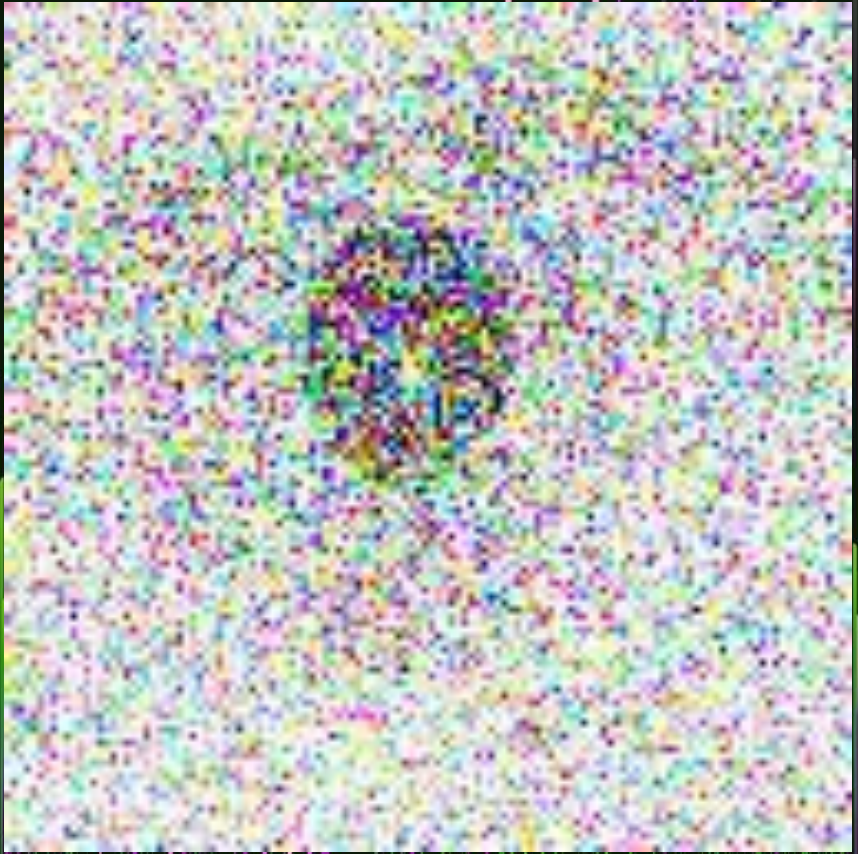
Hyperparameters-3

- Image Size=128x128
- Num Channels=192
- Skip Blocks=4
- Diffusion Steps=4000/5000
- Noise Schedule=Linear/Cosine
- Learning Rate=Gear System(0.0003, 0.0001)
- Batch Size=120
- Micro Batch=3
- Epochs=500

Reasons for Gear System

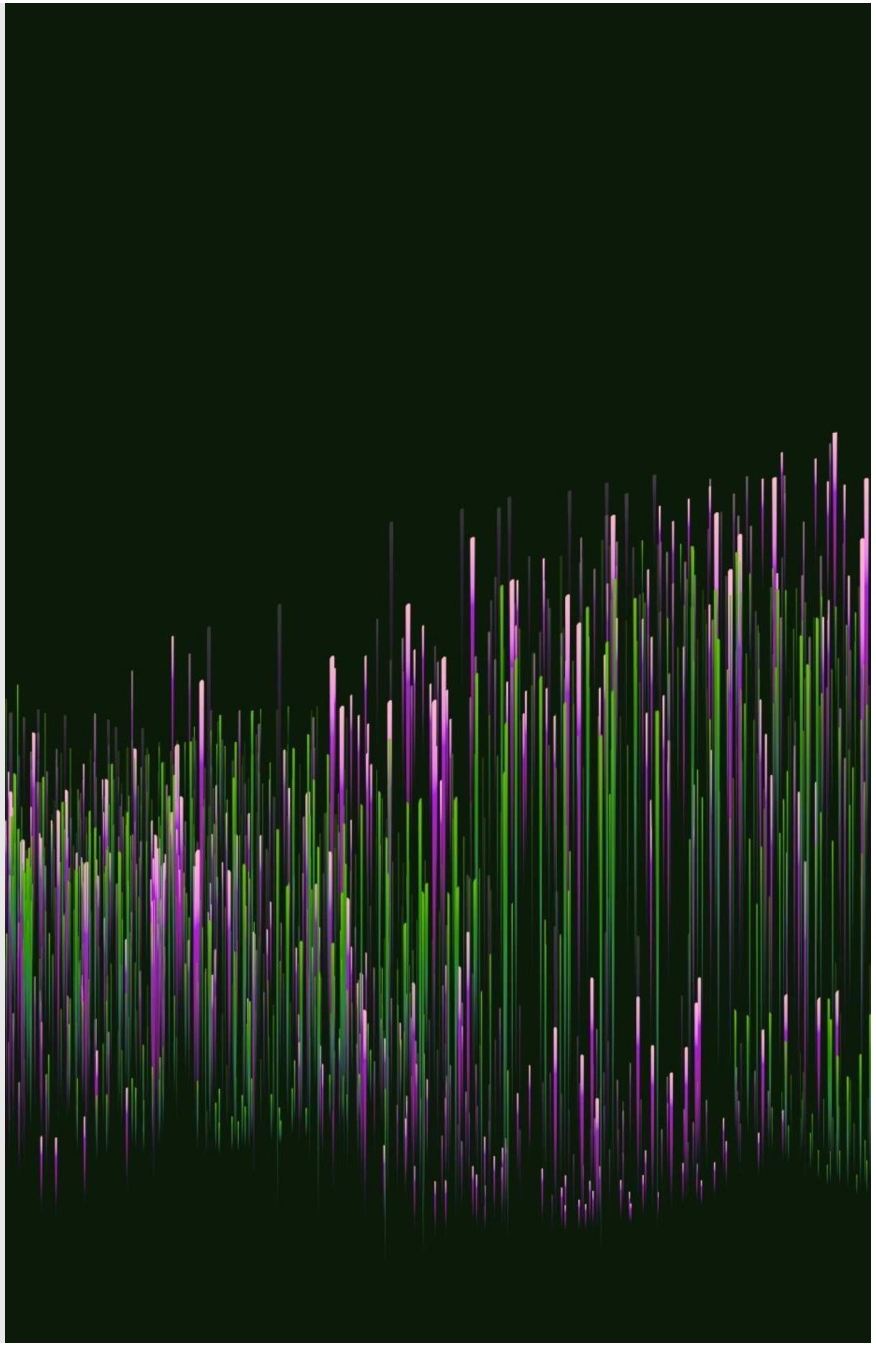
- Due to model complexity the training requires a lot more resources which Colab free isn't willing to spare
- Hypothesis: Higher learning rate in the beginning will help the model converge faster and Lower learning rate in the end will help the model with getting accurate weights

RESULTS



Reason for results

- The Gear system is to blame here
- The higher learning rate in conjunction with the ReLU activation layers is causing the backward propagation to overshoot and deaden neural nodes by assigning a 0 value to them.
- Solution avoid the gear system by training directly on the lower learning rate values
- This solution will need Colab Pro/Pro+
- What we still learn from the output image: The VAEs based noise still works as the shape of the pupil is still more or less intact, the model simply is unable to learn the texture in detail causing it to overshoot.



Path Forward

- We can create multiple burner gmail accounts which come with free Colab and 15GB storage each to train the model via pooling in their resources to train a stable 128x128 resolution model and hope that this hack will suffice for the eventual 320x320 image by downgrading the neural network
- Use the current combined pipeline with Colab Pro and the ideal hyperparameters