Jan 30th, 2025

Implementation of Manifold Learning algorithm

Main Directory Structure

```
└── scripts/
    ├── Wings
        └── Manifold Learning
```

At this stage, and following report from Jan 26th, 2025, the dataset has been curated and stored at a higher level in the folder structure. The script main will implement the manifold learning from the link:

scikit learn - Manifold Learning methods

A change in approach

My approximation to this implementation was influenced by a conversation with Claude Sonnet, where the use of an autoencoder was recommended, mainly as a form of obtaining a latent space representation.

It was decided to move forward with the implementation of a variational autoencoder that used the weights and biases of VGG 19. The first two blocks of convolution and resizing in VGG19 were frozen for training, and used as the encoder for the VAE. Two of such blocks were used.

The original images from the dataset were transformed as follows:

- images got squared to a resolution of [128,128,3]
- scaled to float32 precision from 0 to 1
- moving down the encoder and reaching a final tensor shape of [32,32,128].

The tensor was then flattened into a vector of size 128*32*32, to retrieve a latent representation of the mean and the variance. The latent mean and latent variance have a shape:

```
mean = [128*32*32, 2]
logvar = [128*32*32, 2]
```

The dimensionality in the latent space could be expanded to increase precision

Scripting

The files generated to implement the Variational Autoencoder are located in the following folder as for Jan 30th, 2025:

```
└── scripts/
    ├── Wings
        └── Autoencoder
              ├── constants.py
              ├── VGG19.py
              ├── VAE.py
              ├── VAE_train.py
              ├── VAE_validate.py
              ├── mosaic_VAE_visualization.py
```

constants.py

contains information on the device available to use (either cuda or GPU). This script also integrates the libraries needed to run other parts of the program, and will eventually host any terminal based interaction via argparse.

VGG19.py

Contains the VGG19 model, as found in the original pytorch hub. The weights and biases downloaded and located in the file **vgg19-dcbb9e9d.pth** rely on this structure.

VAE.py

The VAE builds upong VGG19, and describes the model architecture to be used by the main program.

VAE_train.py

A standardized worklflow that loads and handles the dataset and uses it to train the network VAE.

VAE_validate.py

A standardized workflow that loads and handles the dataset and uses it to validate the network VAE.

mosaic_VAE_visualization.py

A visualization tool, mostly usable when the dimensionality of the latent space is in 2D, where one of the rows in the latent space matrix is taken as the mean, and the other the variance. Loops are run to generate images within a specified range and number of steps to visualize the latent space.

Results

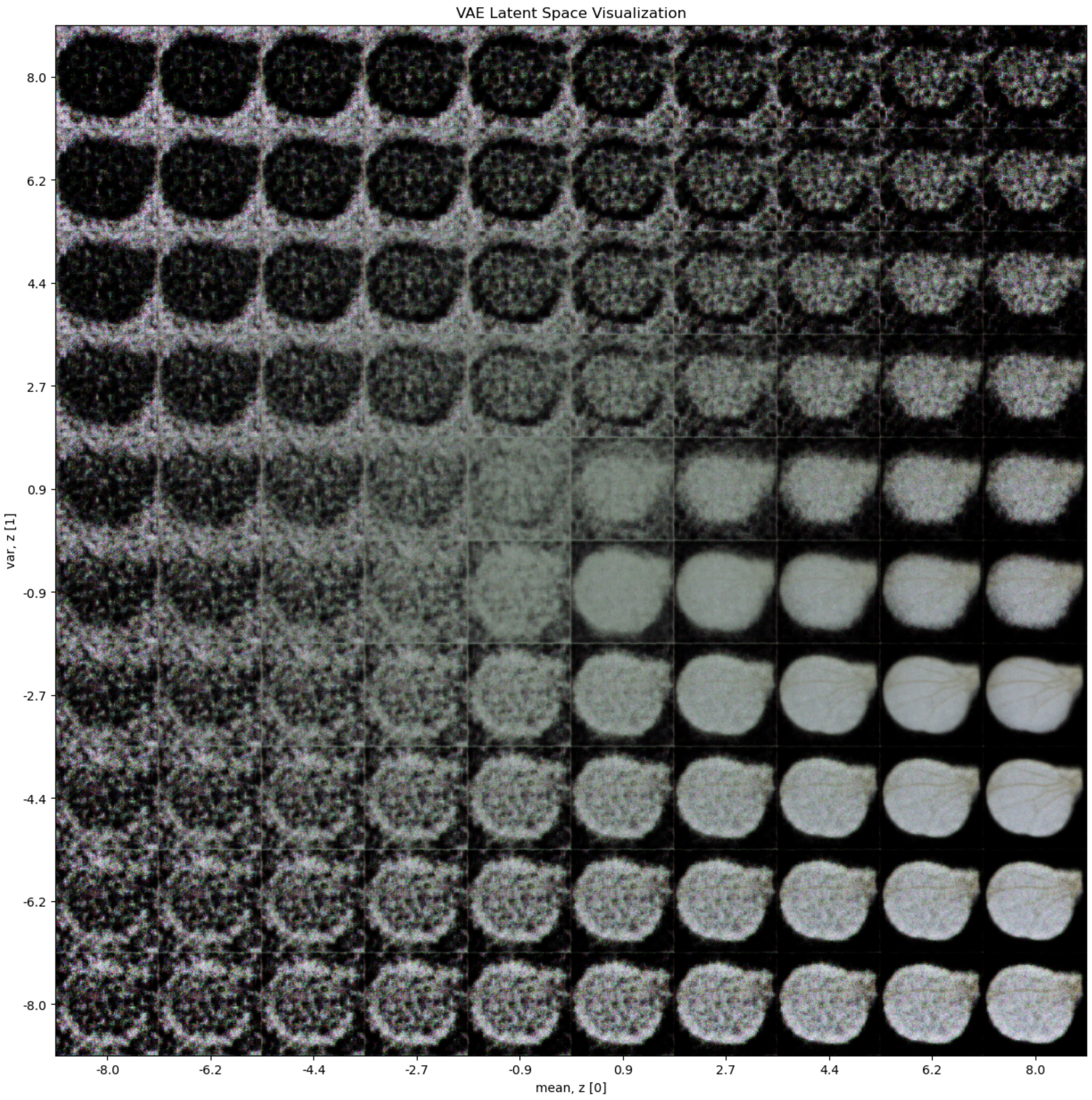
The following is a visualization on the performance of the model using the 133 images that composed the dataset. The model ran in around 5 minutes using an

NVIDIA GeForce GTX 1050 Ti with 4GB of dedicated GPU memory

.

```
    Epoch 1     Average Loss:  146451.028125
    Epoch 2     Average Loss:  50449.146875
    Epoch 3     Average Loss:  36639.30286458333
    Epoch 4     Average Loss:  36239.34947916667
    Epoch 5     Average Loss:  35266.13619791667
    Epoch 6     Average Loss:  34504.91223958333
    Epoch 7     Average Loss:  33151.82083333333
    Epoch 8     Average Loss:  31835.090104166666
    Epoch 9     Average Loss:  30120.800520833334
    Epoch 10    Average Loss:  28091.470052083332
    Epoch 11    Average Loss:  26061.31796875
    Epoch 12    Average Loss:  24489.798177083332
    Epoch 13    Average Loss:  23377.428385416668
    Epoch 14    Average Loss:  22620.6421875
    Epoch 15    Average Loss:  22034.674479166668
    Epoch 16    Average Loss:  21663.246484375
    Epoch 17    Average Loss:  21377.918489583335
    Epoch 18    Average Loss:  21179.863671875
    Epoch 19    Average Loss:  21040.101432291667
    Epoch 20    Average Loss:  20898.929817708333
    Epoch 21    Average Loss:  20818.372005208334
    Epoch 22    Average Loss:  20754.9921875
    Epoch 23    Average Loss:  20690.612760416665
    Epoch 24    Average Loss:  20652.725
    Epoch 25    Average Loss:  20606.744661458335
    Epoch 26    Average Loss:  20554.105989583335
    Epoch 27    Average Loss:  20512.516796875
    Epoch 28    Average Loss:  20484.321484375
    Epoch 29    Average Loss:  20456.496223958333
    Epoch 30    Average Loss:  20430.912890625
    Epoch 31    Average Loss:  20401.98515625
    Epoch 32    Average Loss:  20373.517317708334
    Epoch 33    Average Loss:  20344.584244791666
    Epoch 34    Average Loss:  20314.597265625
    Epoch 35    Average Loss:  20285.383333333335
    Epoch 36    Average Loss:  20248.404817708335
    Epoch 37    Average Loss:  20222.288541666665
    Epoch 38    Average Loss:  20186.488802083335
    Epoch 39    Average Loss:  20143.590885416666
    Epoch 40    Average Loss:  20107.611067708334
    Epoch 41    Average Loss:  20068.185677083333
    Epoch 42    Average Loss:  20034.291145833333
    Epoch 43    Average Loss:  20001.238932291668
    Epoch 44    Average Loss:  19969.5734375
    Epoch 45    Average Loss:  19940.616145833334
    Epoch 46    Average Loss:  19912.724479166667
    Epoch 47    Average Loss:  19892.953385416666
    Epoch 48    Average Loss:  19871.770833333332
    Epoch 49    Average Loss:  19848.977734375
    Epoch 50    Average Loss:  19827.171484375
```

The visualization tool is able to reconstruct an image from the dataset. With 50 epochs and batch size of 40 in a 133 dataset image:


VAE Latent Space Visualization
var, z [1]
mean, z [0]