Testing

This section will be used to document the usage of my colab notebook, so that someone coming along in the future and editing code could ensure that their code is still doing what it should. In training I will be using the dataset found under this notebook/datasets/passion flower. For the using the network section I will be using my dogs dataset work as I have all of these images made

|  |  |  |
| --- | --- | --- |
| Testing | | |
| Input | Result | Evidence |
| Preparing your dataset | | |
| Unzipping dataset zip from drive  (dataset.zip) | Dataset unzipped to local machine’s folder |  |
| Padding unzipped dataset (passion flower folder of jpgs, target resolution set to 1024) | Dataset padded to 1024x1024 |  |
| Converting your dataset to .tfrecords | | |
| Converting now padded images to tfrecords (file path of folder containing jpgs, ouput path of content/tfrecords/flower | Dataset converted to a set of .tfrecords files and saved to the output directory |  |
| Archiving tfrecords to google drive (file path of tfrecords) | Dataset saved to drive folder of the same name |  |
| Importing previously prepared datasets (file path of tfrecords within google drive), I will terminate and restart the session | Tfrecords copied from drive to colab local files |  |
| Training your network | | |
| Transfer learning run using ffhq dataset of the same resolution (1024) | A pkl network file in drive |  |
| Training the dataset further using the previously produced pkl | More pkl and log files |  |
| Using the trained network | | |
| Using generate.py on the afhqdog pretrained network (seed 100) | An image will be produced from the dataset based on the given seed: |  |
| Using generate.py on my own network pkl (seed 1000) | An image will be produced from the dataset based on the given seed: |  |
| Using generateEdited.py without supplying Json files | An Image will be produced from the given network, along with two json files |  |
| Using generateEdited.py, supplying jsons from the previous test | An identical image will be produced to the test above | On an image comparison site, zero differences  were detected: |
| Image Grid using my own network and custom rows + columns (shown in code screenshot) | A large amount of images in rows and columns based on the provided parameters |  |
| Projector.py using the afhqdog pretrained network, for the given target image | An mp4 file which shows the network adjusting an image to fit the target image | Gif is shown in my blog post under the stylegan2 header, will not put it here due to size constraints |