CS 6640 Project 4

Travis Allen, u1056595

December 5, 2021

1 Preliminaries

I chose to do project 4a: Image Mosaicing. The code for this can be found in the `proj_4.py` and `functions.py` files. `proj_4.py` contains the actual algorithm developed to make a mosaic of greyscale images whose correspondences are documented in a `.json` file. `functions.py` contains some useful functions that would have otherwise crowded the `proj_4.py` file. My solution to this project relies heavily on`numba`, so you must have that installed to run my code. I use it because it *dramatically* speeds up the run time.

2 Experiments

2.1 Given Dataset

3 Questions

3.1 How many control points does it take to get a 'good' transformation between images?

3.2 How does the algorithm behave at the theoretical minimum of the number of control points?

3.3 From your experiments, how does the accuracy of the control points affect the results?

4 Details

4.1 Contrast

4.2 Feathering

4.3 Image Size

Initially, for n number of images, I make a canvas that is $n + 1$ times the size of the largest image so that I have enough room to work with when placing images in the mosaic. However, this often results in a canvas which is much larger than it needs to be. To return the canvas to a more reasonable size for viewing once the mosaic is complete, I execute the following procedure. I search through the large canvas to find the first and last rows and columns which contain only zero elements. I do this by using the `numpy.sum()` function on each row and column and checkign to see if it is equal to 0. The canvas consists of all zeros before I place images on it, so this method works by assuming that a row of all zeros contains no image information. I perform it this way because I figure that `numpy`'s vectorization is faster than my own implementation of computing the sum or individually inspecting every element in the image.

- Place all of the images in a folder with a known path to the directory that contains `problem_4.py`
- Place all of the names of all of the images in a `.txt` file in the folder, with each name separated by a new line
- Write the names of the folder and the file in lines `27` and `28` of `problem_4.py`
- Write the maximum size of the images in lines `21` and `22` of `problem_4.py`