Assignment 4

I. Technical Overview

A. SIFT feature matching

The Scale-Invariant Feature Transform(SIFT) is an excellent feature extracting methodology that gives consideration to both performance and computation efficiency. To summarize, it generally extracts features based on the below 4 steps.

1) It uses the idea of Laplacian of Gaussian filter, and applies LoG of different σ onto the original image, which forms an octave, to smooth out clutters for scale invariance. The difference of adjacent images in the octave is saved into Difference of Gaussian(DoG). And the process is looped over by further sampling less pixels into a smaller octave and DoG, so finally we would get a downsampling pyramid, or a scale space.

2) To get the points of interest, every pixel is compared with its neighboring pixels amidst its adjacent scales, and would be selected if its local extrema.

3) Some optimizations are applied on the selected interest points, which include discarding low contrast points based on threshold, and inspecting the ratio between the highest and lowest eigenvalues of the 2×2 Hessian matrix. Larger ratio indicates an imbalanced edge detection on two directions and thus prone to be outliers.

4) Calculate the gradient magnitude and orientation for each pixel, and abstract every region into a 4×4 subregion, with each one being a 8-bin orientation histogram(parameters can be tuned). The output would be a feature descriptor that denotes the feature in an appended array.

5) Match the feature points between two images using nearest neighbors based on a ratio threshold for rejecting false matches. The point match could serve for further applications such as image stitching.

B. Homography Estimation

The homography algorithm computes the transformation matrix between two images on the points of interest. The Direct Linear Transformation(DLT) is a close-form like solution that uses SVD to approximate the best solution between the given correspondences. To achieve better transformation invariance, points are translated and normalized within $\sqrt{2}$ to the origin, and then DLT is applied for a number of iterations with randomized 4-correspondences. Inliers are computed for each round based on thresholding and the best solution is selected with most inliers.Further optimizations include using probabilistic methods to minimize cost functions based on the computed inliers, and updating correspondences for guided matching.

C. Multi-band Blending

The multi-band blending is an algorithm that smartly merges and blends multiple images all together with smooth transition and accuracy retention. It works by creating a weight function on the image intensities, which linearly decreases from the center to the edge. The image is splitted into various frequency wavelengths, and each range is weighted by the convoluted weight function(by a Gaussian filter). All ranges are summed and averaged to form the final blended output. So different bands will be applied a corresponding extent of blur on its weight map, resulting in a smooth transition in the image overlapping areas. And at the same time, the property of max-weight maps ensures that the image containing most responsible points dominate its weight, thus preserving high frequency details.

II. Experiment on the AutoStitch

36 pictures are taken for this experiment, with a large proportion of overlapping between any two adjacent images. They are displayed as below.

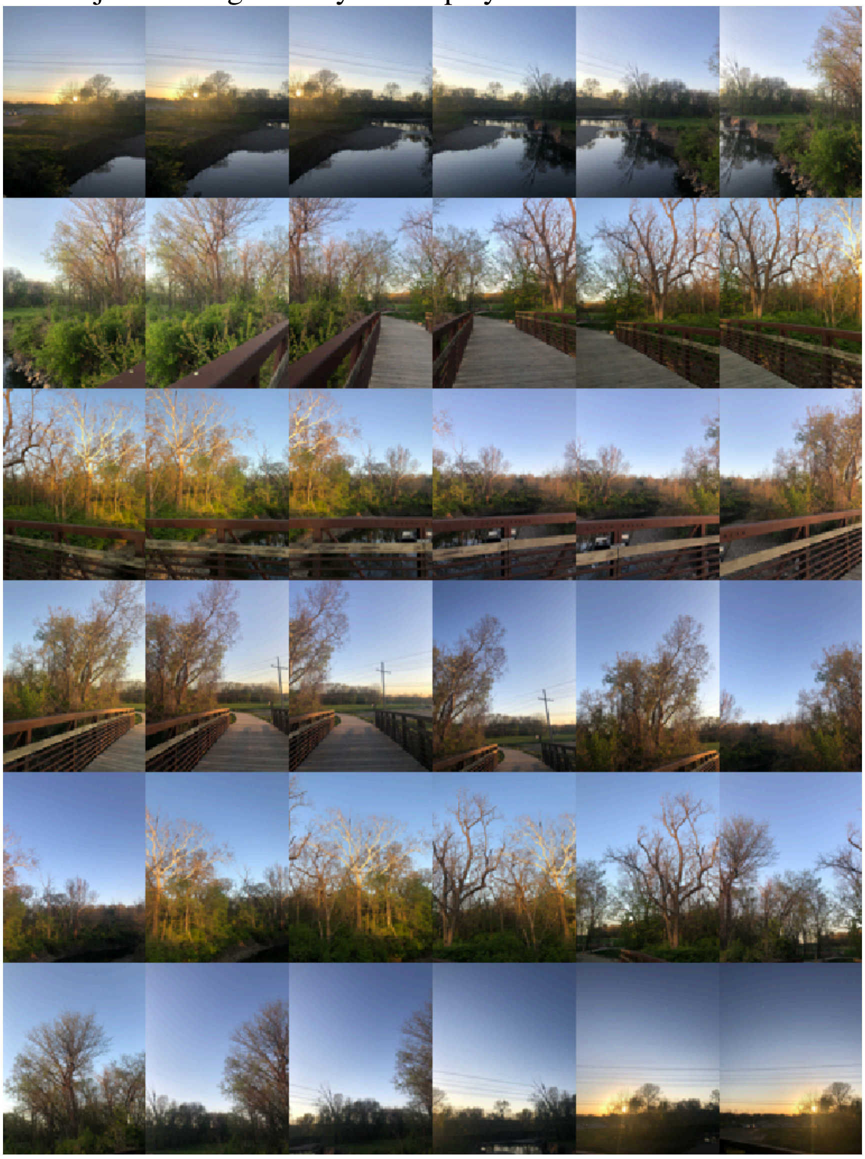

Fig 1: 36 images taken by rotating the camera around the optical center

And below is the stitching result.

Fig 2: the stitching result of the original images

The result shows a well-stitched panorama merging adjacent parts together with a natural color and illumination transition. The output achieves a quite good feature matching between adjacent images, except for the bridge railing break at the left side of the panorama. An interesting observation would be the stitch between the below 4 images, which were not taken in an adjacent time sequence, and thus some illumination variance exists.

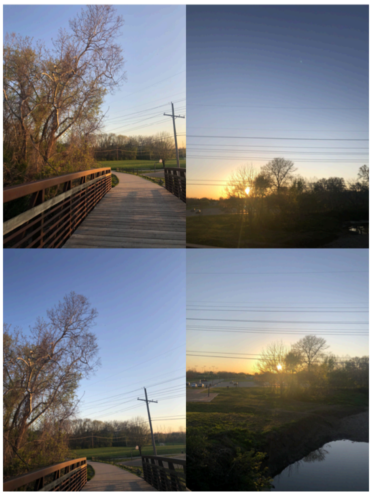

Fig 3: Four geographically adjacent images taken at different times

The images on the right contain less brightness and a little bit different hue as well. Observing from the output, the multi-band blending averages the convoluted weight at different frequency bands, gently smoothing the hue difference.

For comparison, some variance are added to the images randomly. First, 7 randomly chosen images are rotated 90 degrees, and the output stitch is as below.

Fig 3: Stitch output after 7 random images are rotated 90 degrees

The output is the same as the original one, indicating that the algorithm is rotation invariant. Also, a variance in illumination and hue is applied. Eight random images were applied different color filters that changes brightness, contrast, saturation, vignette and tint of the image, shown below.

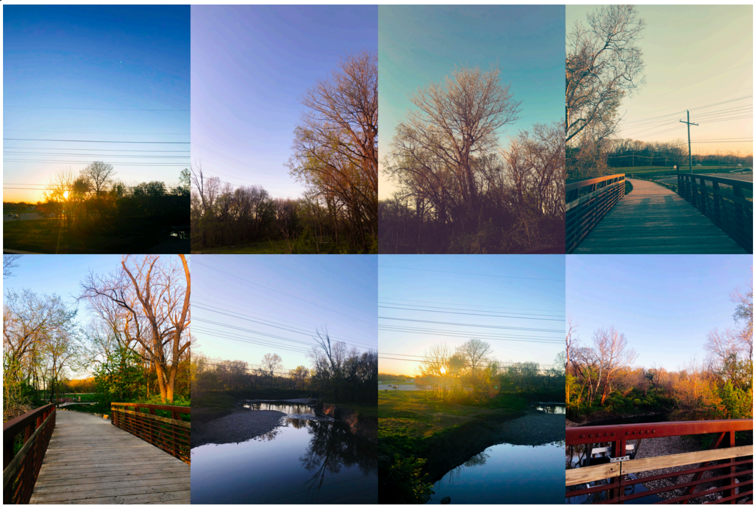

Fig 4: Eight random chosen images that were applied different filters

Fig 5: Stitch output after 8 random images are applied different color filters

Fig. 5 displays the stitch output. Generally the filtered images are quite naturally blended together. Also, looking from the output, the changes in color and illumination do not impact feature extraction and mapping, demonstrating a robust repeatability on feature extraction.

The OpenPano is also tried for comparison. The stitch output on original images is displayed below.

Fig 6: Stitch output by OpenPano

Comparing to AutoStitch, OpenPano seems to miss several images in my case, which I think is caused by failure to match corresponding features. Besides, it could be observed that there are apparent distortion on the edges due to its stitching algorithm. For filtered images, OpenPano failed to recognize one image for feature detection. It contains less robustness on illumination variance than the AutoStitch.

```
[optimize@incremental_bundle_adjuster.cc:162] BA: Error 10.192635 after 7 iterations
error: Found a tree of size 35!=36, image 31  are not connected well!
```