LAB 4 - Kyle Coelho

Camera Calibration

My phone camera was calibrated using the printed checkerboard pattern that was printed in class. A wide variety of angles were captured totalling 15 pictures. The images were then downsized to approximately 1 Megapixel (1024x1024). Following that, their grid corners were extracted using the same reference point. The images were then calibrated and the following results were obtained. With the first round of calibration, the following was obtained:

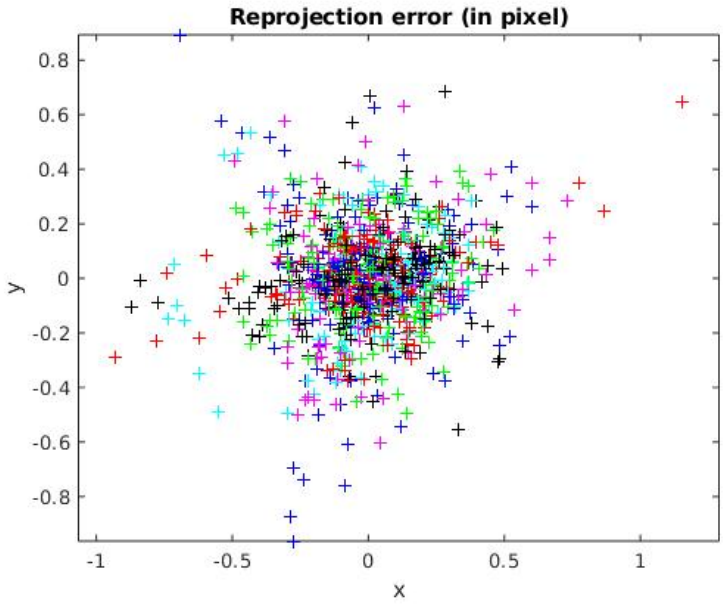

Fig. 1

While the mean pixel error for this was quite low, approximately 0.21, when undistorting my Forsyth images using these parameters, the results were not desirable as the 4 edges started getting curved and that did not seem desirable. In Fig 2, this can be seen as the sidewalk and Dana building start curving.

Fig. 2 Left shows original image and Right shows the imperfect edges

Hence, 3 more calibration images were taken with the checkerboard more in the frame of the camera. While my previous images all had the calibration image fill up most of the image, I tried to make sure that these new ones almost fit perfectly with the image frame. After doing this procedure again, the mean pixel error was higher, with a value of 0.51. The added images definitely introduced the error as the light blue points show below. However, it managed to undistort the images better and so was used moving forward.

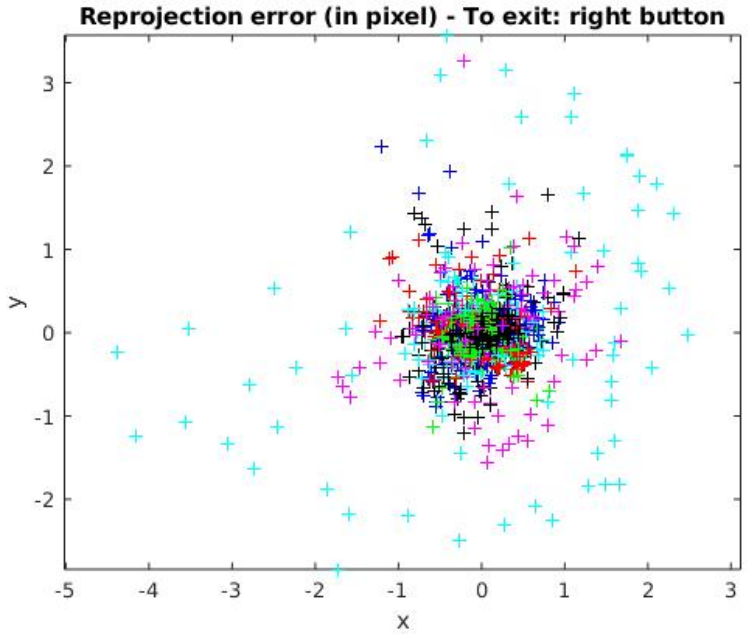

Fig. 3 Reprojection Error

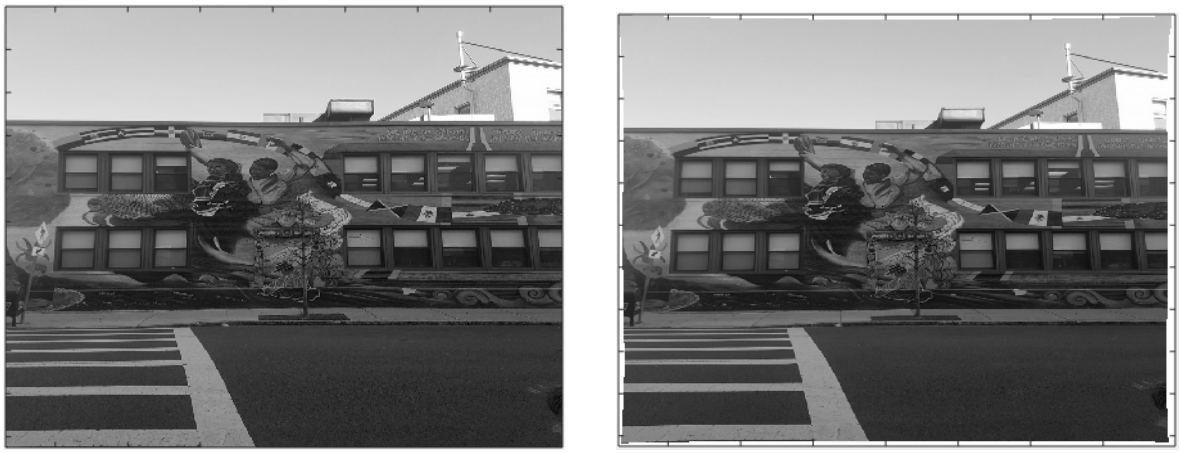

Fig. 4 New Undistorted Images

Photo Mosaicing

For the panorama, 8 pictures were taken of Forsyth like the ones shown above. The examples of undistorting images can also be seen in the Figures above and one that includes more of the side of the building is shown in Fig 5 below. This image also shows a clear example of why some of the future panorama fails because of someone bending at the bottom. However, I was not able to return to Forsyth again to retake the images.

Fig. 5 Undistorting

Following the undistortion of all images, I used the Harris edge detector to determine features well distributed across the image and then the extract features were added to the cornerPoint Matlab object. The factors I tried modifying for the Harris detector were the tile size, number of interest points and threshold. The parameters that I settled upon were

10,000 interest points, tile size of [2 2] and no specified threshold. 10,000 points were chosen to give it a higher chance of matching features when creating the panorama. It is definitely more than needed but I thought it would be better to have more features and then apply a threshold. A [2 2] tile size was better than [1 1] but it slowly did not matter as the tile size increased. The threshold was an important factor. When I took the images, my image included a lot of the ground and sky. As a result, the ground and sky were registering a lot of features unnecessarily and being used for matching. This was leading to poor panorama creation. The Figures below show this.

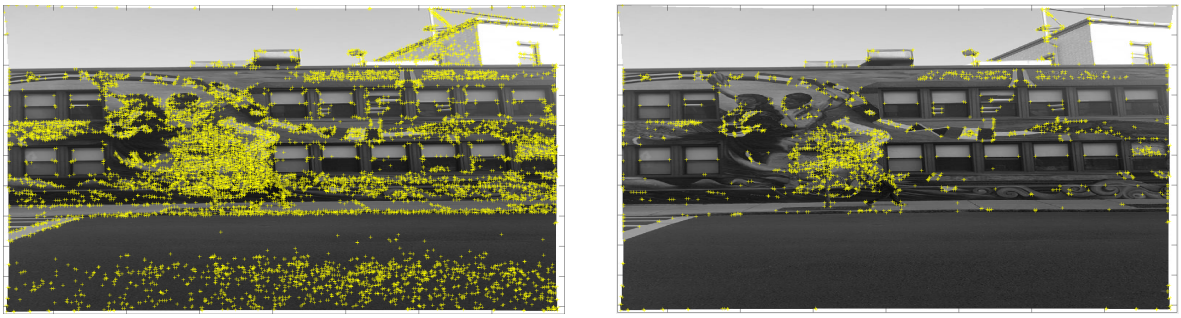

Fig. 6 Left with a lot of ground points and Right with the threshold applied

While the above is simply to show an example, I used trial and error to get a threshold that seems to capture the most amount of relevant features and minimum amount of road.

Fig 7 Panorama using unthresholded set of images

As shown above, the top right and bottom left of the panorama do not match up well at all. While trying to make the mural and road match up, the algorithm forsakes other clusters of less dense points. Then, when trying to adjust for that by applying the threshold and ignoring a lot of the road points, the panorama below is produced.

Fig. 8 Using the threshold set

It does not make too much of a difference overall. The top right pole now matches up well. But, the bottom left crossing and the sidewalk do not match up anymore. It is difficult to use a metric to compare the results, but from visual inspection they both do not particularly turn out well. The left side of the image definitely is getting much more compressed for some reason and is getting almost entirely overlayed by other images.

I also tried to use the uncalibrated images taken from my phone for the panorama since it might have been doing calibrations automatically. The results do not seem to be that much different. However, it does capture much more information that was somehow excluded entirely above, such as the end of the left sidewalk.

Fig. 9 Camera Panorama

Certainly, a solution to this would be to crop the initial images to exclude all these mistakes and be cropped only to Forsyth Building. Then, the same resizing, calibration and mosaicing would work much better.

Next, when trying to estimate the camera positions, the approach I used was finding the height and width centers from the transform limits already generated while running the script. Hence, the centers from the x,y coordinates were paired up and plotted on top of the images. It seemed to give a reasonable estimate. I can tell this purely from visual inspection because the camera seemed to be in the center of the image for the middle portion of the images, while it moved outward for the images on the side. It was unable to overlay it on the panorama so the figure shown below shows the camera position for a few of the calibrated images used in creating the mosaic.

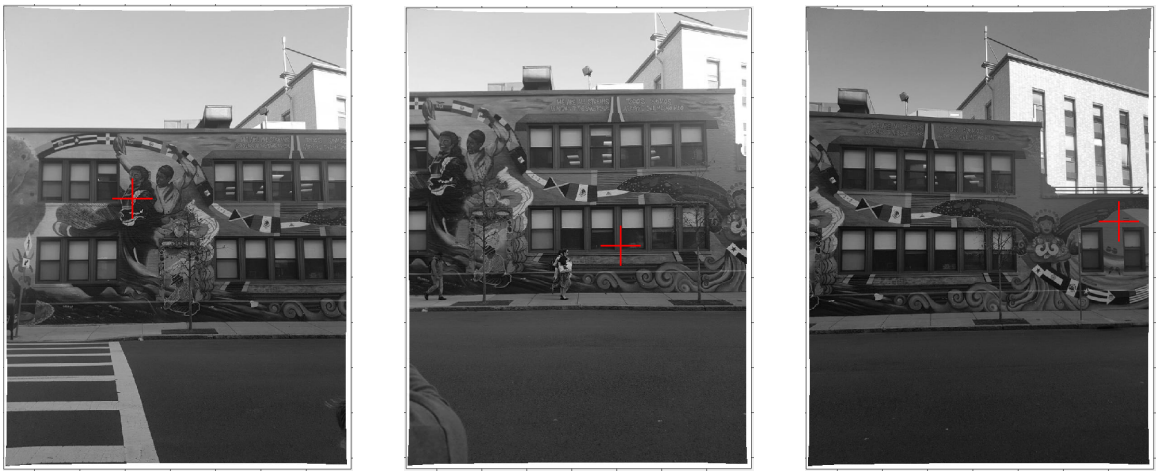

Fig. 10 Camera positions

They seem reasonable because the left image and middle image seem to be approximately where I would have centered my camera. The middle image also should be lower because as I moved to the right, my camera had to move slightly under a tree that was blocking the view. The image on the right seems like the camera was too far but that could be a result of flattening the image through the undistortion since I could have been angling my phone sideways to capture more of the building in the image.

Varying Overlap

50% Overlap

The same procedure was followed using a brick structure with 50% overlap. The same parameters that worked well for the Forsyth mosaic were used here. The overlap was 50% but I tried to ensure that some well-defined features were overlapping within the overlap. There is no difference in the custom calibrated and the raw resized photo mosaics as can be seen below. The result is quite good in that most of the pictures align with each other very well. However, there still seems to be something wrong with the way my phone is calibrated or with the set of pictures I took for calibration since the left side of the image, particularly bottom left, is problematic. I might have been tilting my phone too much initially but it is difficult to tell. Also, since the pole is only in a single image, the features do not get matched appropriately and causes it to focus on matching the arches on the door of the church, which show up more strongly with more features.

Fig. 11 50% overlap panoramas

What seems to give some insight is the camera position. The same procedure is used. While the camera position calculated is fine and seems to match up well for the last 3 images as

shown below. For the first 2 images, which are the leftmost images, they do not even appear on the image. This seems to signal that the camera is not in the frame when it should be somewhat in the center. It could indicate that the camera was tilted too much while taking the picture. Or it could be some intrinsic problem to my camera that I am unaware about. This would also explain why the leftmost images of my Forsyth dataset gets messy as the sidewalk, with very clear features and edges does not ever seem to get matched correctly.

Fig. 12 Camera Positions

I am unable to plot it properly by extending the lower image bounds but the camera position for the erroneous left images are all lower than what they are supposed to be. It seems like they are coming from closer to the ground. I am unsure what this means because this looks to be more of a setting issue on my phone that I am unaware of apart from just calibration.

15% Overlap

Again for this the settings were the same as above. However, this time the result came better than all the others even though it should be much harder with the limited set of features. The major difference could be that the painting is rich with features, much more even than the other sets. The lighting is also very consistent throughout the photo and there were no obstacles of any sort that were drawing features away from the main image. The background is white and for some reason did not contribute very much to the features. All this seemed to help in bringing together a good mosaic for the 15% overlap

Fig. 13 15% Overlap panorama

This time when analysing the camera positions, most of them again seem to be below the image frame. Only the image that the algorithm determined as one of the middle images seems to be perfectly in frame; however, all the others should be as well since the painting was at chest height.

Fig. 14 Camera Position

This makes it even harder to understand some of the errors in the previous pictures. While this was taken indoors, I do not believe that this could have had anything to do with it. Since this has come out so well, it seems like the imperfections in my other mosaics could be from instances of poor data collection on my part. Since I monitored the features extracted every time, I know that the features were all evenly distributed as shown in Fig 4. The phone setting, I realize was on 'Auto' on my Samsung S6 and could have added some adjustments based on the lighting or the way I could have been holding my phone. However, I do not know through the data what could have caused the imperfections in the other mosaics compared to this one.