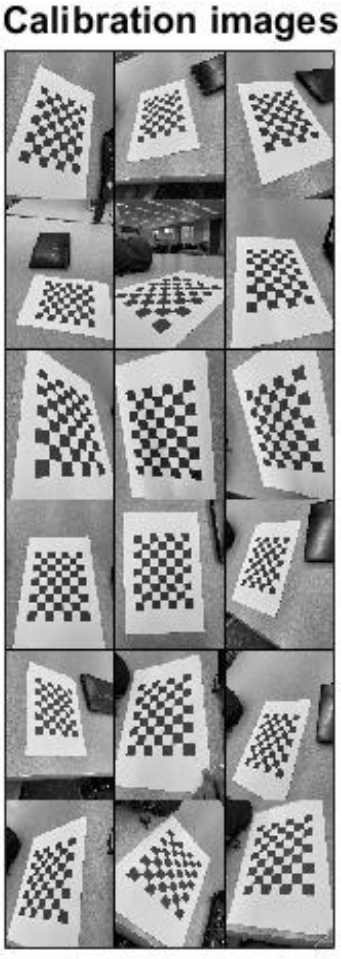
EECE 5554 Robotics Sensing and Navigation LAB-5 REPORT

Nitin Thakkar

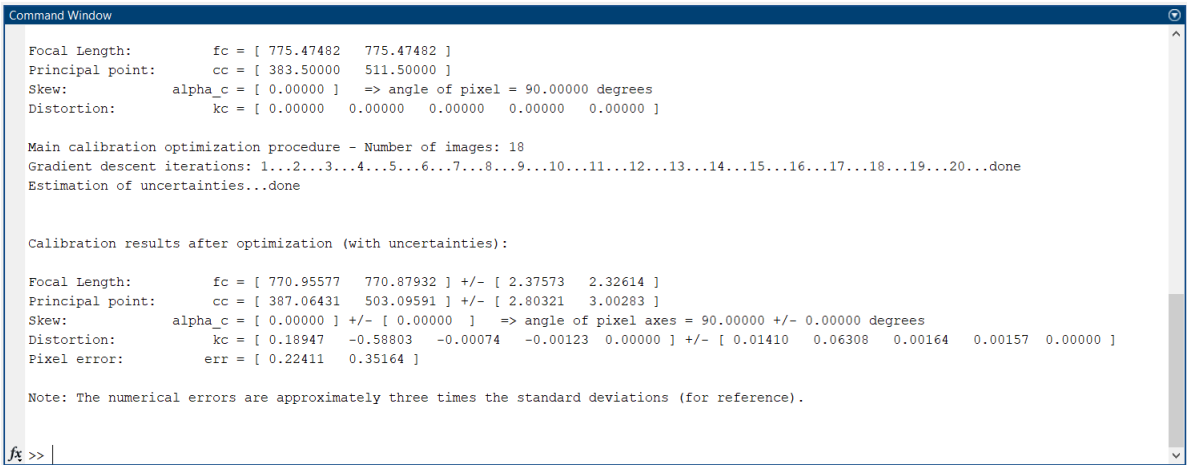
Camera Calibration:

1)Following images were used to performed camera calibrations:

Here to perform Camera Calibration it is important to have checker board orientation to different orientation since it help to estimate the intrinsic parameter of camera, where intrinsic parameters are focal length, principal point and lens distortion coefficients, hence by obtaining these parameter at different orientation, the calibration algorithm can estimate the intrinsic parameter more accurately which enable to remove distortion from images since it change the important information which can be useful for perception task for system.



2) By using Caltech Camera Calibration toolbox, Images were loaded on the memory

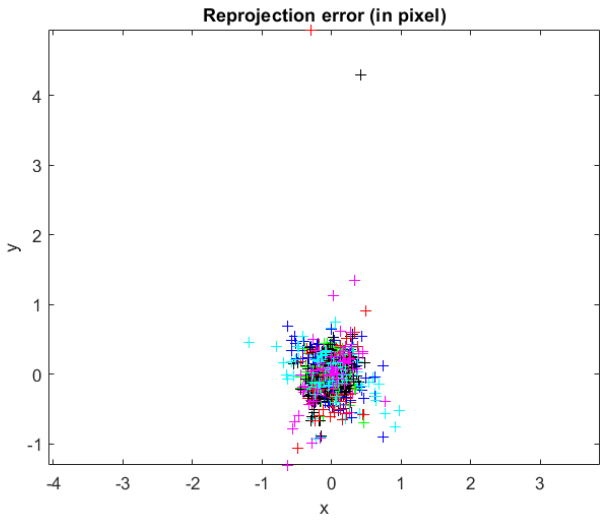


The figure above represents the parameter which are obtained using calibration tool where it can be witness that following 20 iterations were done for 18 images which shown earlier and able to obtain Focal Length, Principal Point, skew, Distortion and Pixel error we can used this calibration parameter to calculate the Reprojection Pixel error.

Reprojection Pixel error:

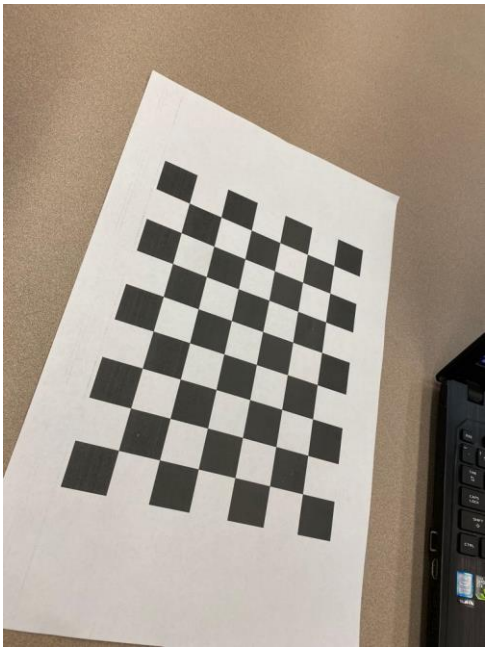
```
Number(s) of image(s) to show ([] = all images) =
Pixel error:      err = [0.22411   0.35164] (all active images)
```

For all Images toolbox estimated the error of 0.22411,0.35164 Pixel error and the graph below plotted is estimated Reprojection pixel of all images:

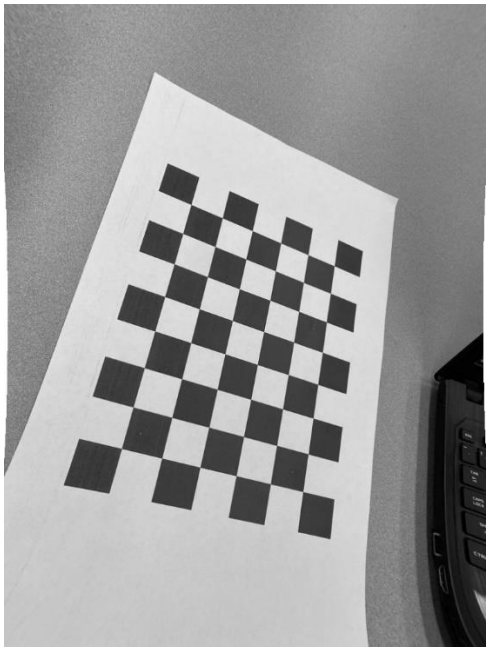


By plotting Reprojection Error it can be easily concluded that the performed calibration is successfully executed, and distortion of following images can be easily done using this calibration parameters.

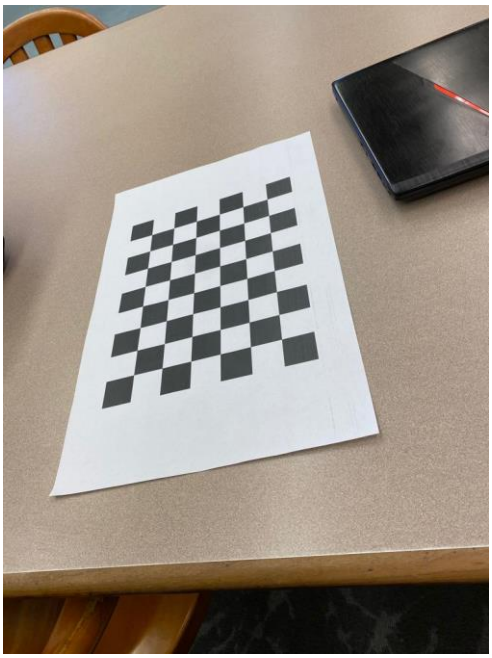
3) An image before and after distortion removal can be seen below: here the following calibration parameters are used to remove this distortion and the following image is observed for before and after this process.



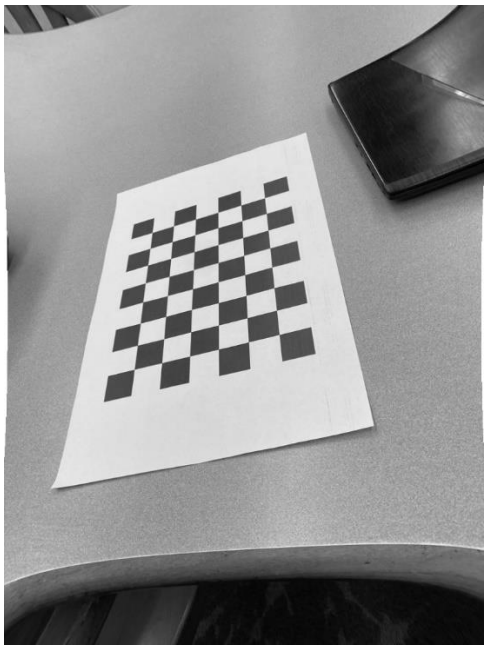
Before Distortion removal



After Distortion removal

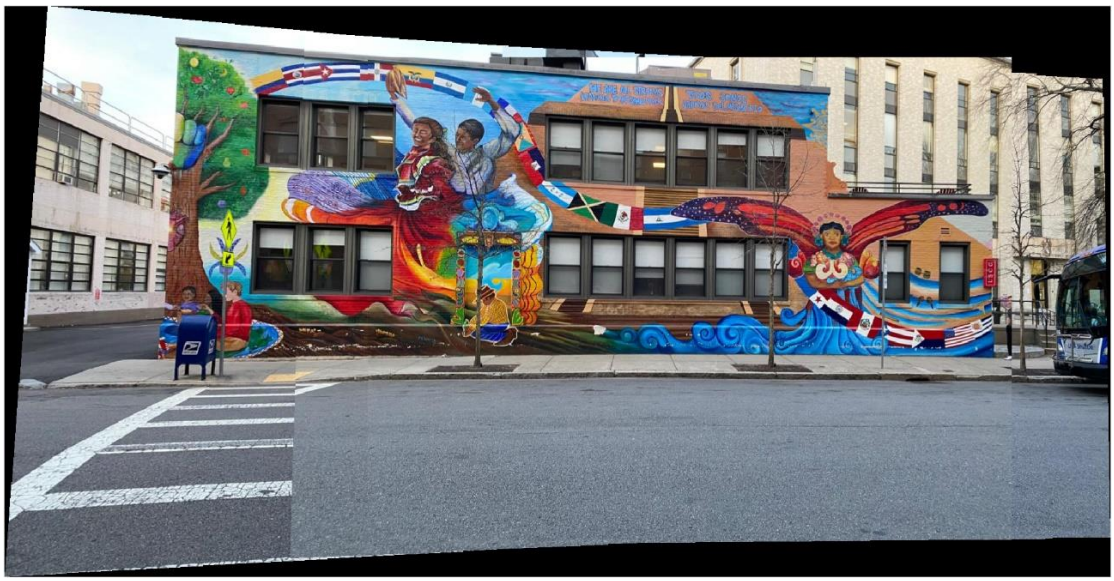


Before Distortion removal



After Distortion removal

4)Latino Student Center Mosaic: Following dataset was used for performing mosaic application:

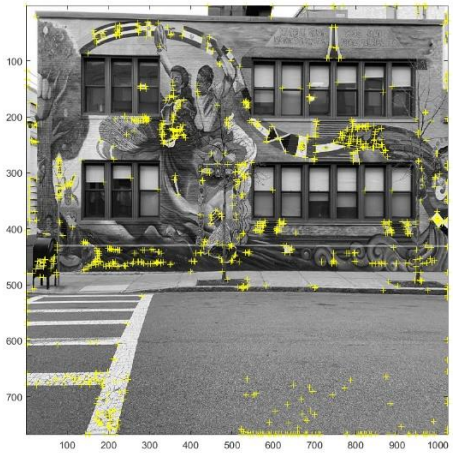


Result of Mosaic performed on Images.

This is the following result which was obtained in performing the mosaic whereby using Harris corner point detection algorithm was used to extract the corner point and then that points were used to stitch the image to generate this panorama and we can see that the panorama generated from this is good in terms of efficiency of stitching. Here the given 7 figures were used to perform this operation which is seen above.

5) Harris corner point across LSC image:

Original image

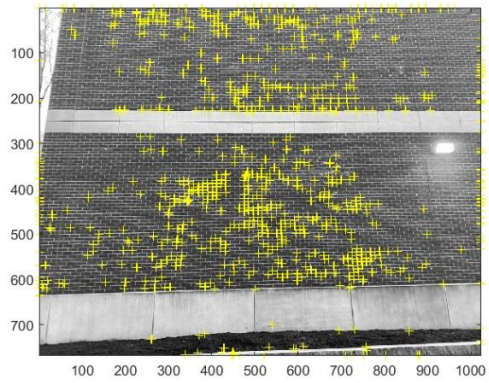


Harris Corner point

For this I took Photos from iPhone 11 pro max where at given available setting of 1080x1920 but that would be not sufficient for detecting the corner point since it can be noticeable that since higher pixel will generate high number of features which can be difficult to process hence as to make this process at ease, I downgrade the images at 768x1024 to work sufficiently for requirement for this operation.

Also, to have better panorama, grid was used in camera while collecting the photos to follow a straight line which can give great panorama also easier to stitch without error of missing corner points.

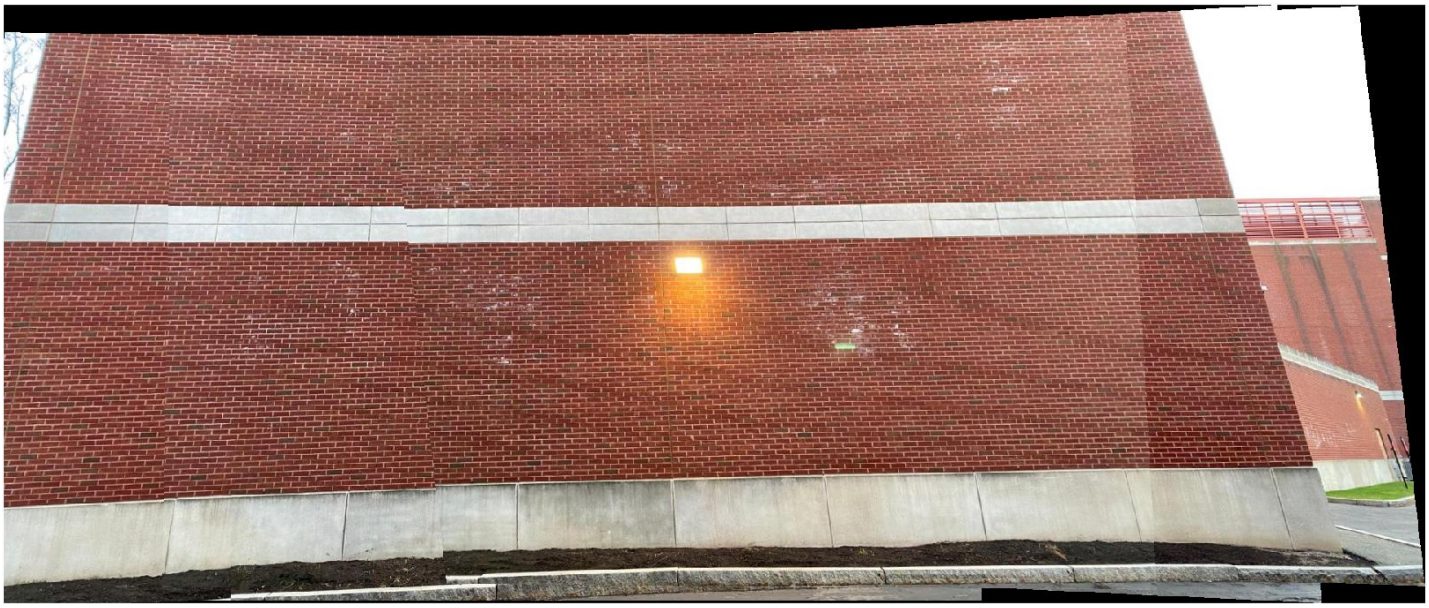
6)Cinder Block/brick wall Mosaic: for this operation points were taken 1000 tile of [2 2] to divide image into two sets to distribute point uniformly



Original image

Harris corner point

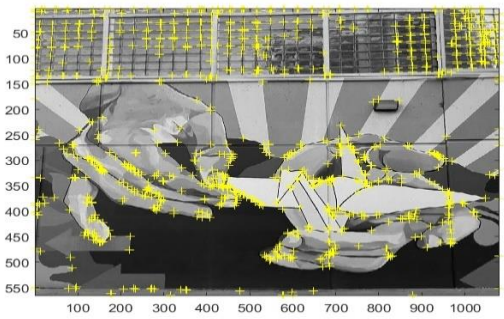
Panorama of Brick wall:



To take data I used grid as earlier and down sample the image at 768x1024 where I notice an issue where it took me few iteration to make good result since Harris corner point algorithm can detect walls which present the same feature point so hard to distinguish and we can see that wall have too many identical feature so proper mosaic was not able to generate and also while documenting this result I noticed it might be intended to take photos from close range since It would not create good mosaic compare to LSC data since there are identical feature and algorithm might not work as intended but we can see there are other feature such as concreate strips.

7)Mural -Third Mosaic with two different datasets of 15 % and 50 % overlapping feature of same mural:

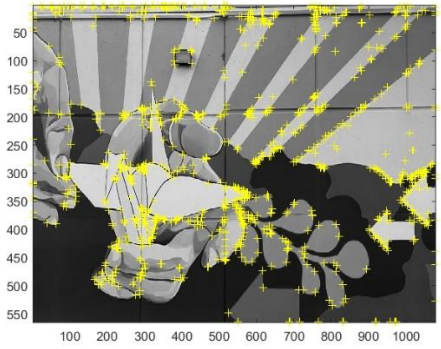
Initial images with Harris corners with 15% overlap:



Original image

Harris corner point of 15 % overlap

Initial images with Harris corners with 50% overlap:



Original Image

Harris corner point 50 % overlap

Final mosaic with 15% overlap:

Final mosaic with 50% overlap:

Discussion of performance with 15% and 50% overlap:

Comparing both results we can say both performed well to execute this operation where 50 percent overlap generated a better and cleaner mosaic compared to 15 % since the corner point where not easily identity in consecutive image since we include lesser feature to work with for that point hence it can be pre predicted that 50 % overlap will perform well and it surely did. Considering the 15 % overlap it does perform well where it able to stitch the image correctly but It was needed to change the feature point deployment for Harris corner algorithm where we taken 2000 feature point and tile division of [3 3] where before doing that it was not giving result which we can call mosaic hence after tweaking those parameter we were able to obtained really great result for 15 % overlap.

For this we do have to change some parameter, firstly for the rest of dataset we used feature point of 1000 but for 15 % overlap we have to change to 2000 point and tile of [3 3] and while for all dataset we used grid in camera to get straight dataset, here for till 50 percent overlay it was easily we were able to get result and data but for 15% overlap image it was needed to crop some images to remove ground and also straighten them up since due to human error it was tilt and then result to change in image resolution which was then fixed to 1080x566 to make uniformity along image pixel size.

For files Block.m is used since for the rest of the data only parameters and data location needed to change.