EECE 5554 Robotics Sensing and Navigation Lab-5 Report

Abhinav kumar (group 10)

Camera Calibration:

- Camera images used for calibration:

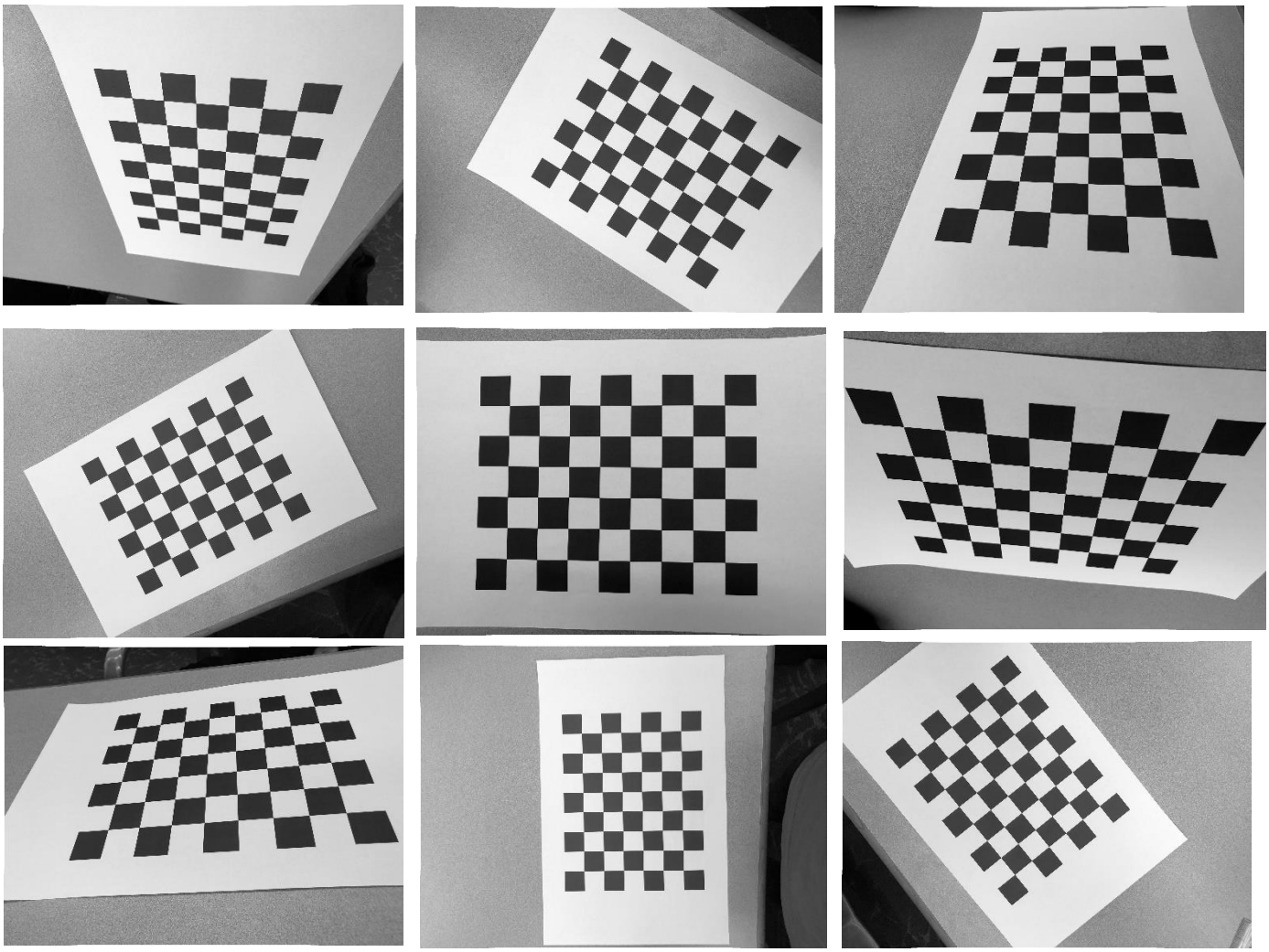
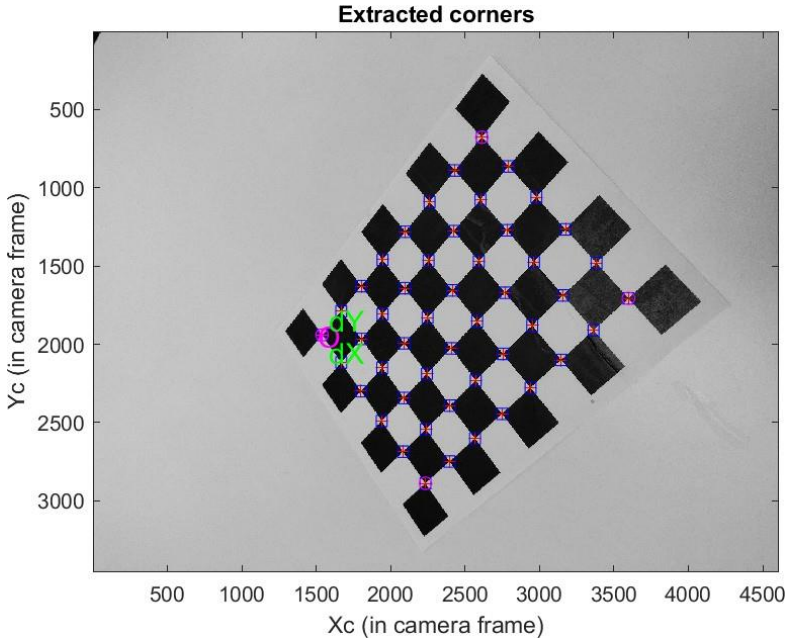


Fig-2 Extracted corners after of the checkerboard after manual marking of the outermost points of the board



Reprojection pixel error:

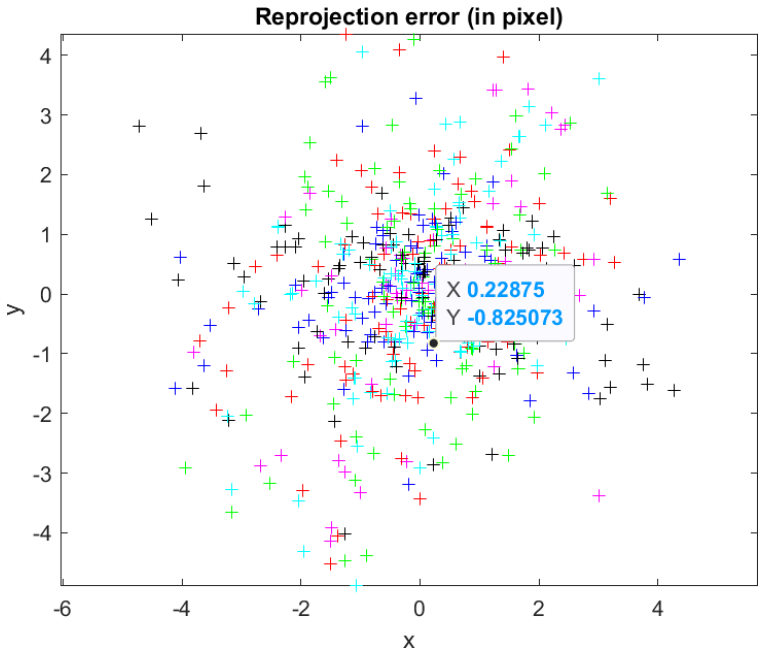


Fig-3 Reprojection error in pixels

Calibration parameters:

```
Initialization of the intrinsic parameters - Number of images: 11


Calibration parameters after initialization:

Focal Length:          fc = [ 3613.47720   3613.47720 ]
Principal point:       cc = [ 2303.50000   1727.50000 ]
Skew:             alpha_c = [ 0.00000 ]   => angle of pixel = 90.00000 degrees
Distortion:            kc = [ 0.00000   0.00000   0.00000   0.00000   0.00000 ]

Main calibration optimization procedure - Number of images: 11
Gradient descent iterations: 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...16...17...18...19...20...done
Estimation of uncertainties...done


Calibration results after optimization (with uncertainties):

Focal Length:          fc = [ 3607.93432   3607.51145 ] +/- [ 24.59673   24.12852 ]
Principal point:       cc = [ 2323.62613   1780.85479 ] +/- [ 27.29760   21.48619 ]
Skew:             alpha_c = [ 0.00000 ] +/- [ 0.00000  ]   => angle of pixel axes = 90.00000 +/- 0.00000 degrees
Distortion:            kc = [ 0.01691   -0.20701   -0.00173   0.00246  0.00000 ] +/- [ 0.02489   0.10538   0.00212   0.00245  0.00000 ]
Pixel error:          err = [ 2.29608   2.27071 ]
```

Fig-4 Calibration parameters after initialization

Pixel error, as seen in the figure above which is higher for a 4k resolution picture. So it is understood that corners were not detected correctly in some highly distorted images hence after initial calibration since the observed reprojection error seems to be high, using “Recomp” the image corners are recomputed on all images automatically. Once that is done recalibrating it has resulted in a reduced reprojection error and the results have been presented below

```
Re-extraction of the grid corners on the images (after first calibration)
Window size for corner finder (wintx and winty):
wintx ([] = 5) =
winty ([] = 5) =
Window size = 11x11
Number(s) of image(s) to process ([] = all images) =
Use the projection of 3D grid or manual click ([]=auto, other=manual):
Processing image 1...2...3...4...5...6...7...8...9...10...11...
done

Aspect ratio optimized (est_aspect_ratio = 1) -> both components of fc are estimated (DEFAULT).
Principal point optimized (center_optim=1) - (DEFAULT). To reject principal point, set center_optim=0
Skew not optimized (est_alpha=0) - (DEFAULT)
Distortion not fully estimated (defined by the variable est_dist):
     Sixth order distortion not estimated (est_dist(5)=0) - (DEFAULT) .

Main calibration optimization procedure - Number of images: 11
Gradient descent iterations: 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...16...17...18...done
Estimation of uncertainties...done


Calibration results after optimization (with uncertainties):

Focal Length:          fc = [ 3615.96083   3611.34151 ] +/- [ 16.10446   15.78036 ]
Principal point:       cc = [ 2306.87523   1783.92182 ] +/- [ 17.69455   13.99133 ]
Skew:             alpha_c = [ 0.00000 ] +/- [ 0.00000  ]   => angle of pixel axes = 90.00000 +/- 0.00000 degrees
Distortion:            kc = [ 0.02504   -0.25495   -0.00186   0.00151  0.00000 ] +/- [ 0.01597   0.06759   0.00137   0.00157  0.00000 ]
Pixel error:          err = [ 1.50995   1.47287 ]

Note: The numerical errors are approximately three times the standard deviations (for reference).
```

Fig-5 calibration results after optimization

The reduced reprojection error as observed from above figure is [1.50 1.47] which is reasonable for a 4k resolution pic.

An image before and after calibration is in report

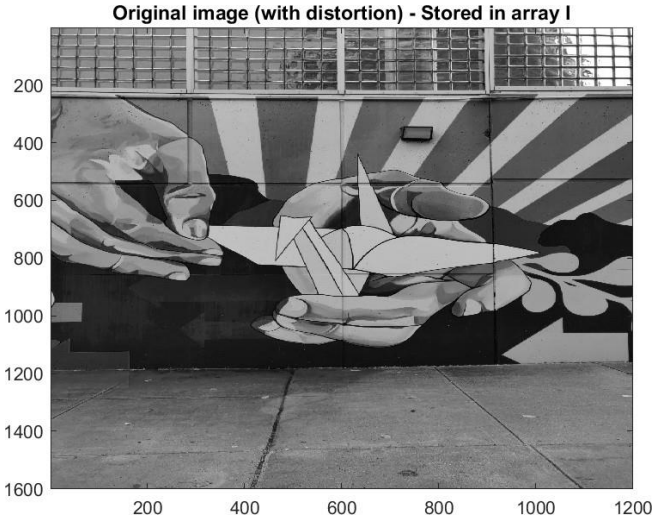


Fig-6 Original Image With distortion

Fig-7 Original Image Without distortion

2. Latinx Student Cultural Center (LSC) Mosaic

2.2.Discussion

- I have used seven images to create the panorama. But for brevity, I will show the first three images and will show all the images in last section of the report.
- I had to resize the image from 3042 × 4032 to 720 × 1024 because the resolution was too high and the Detector was not able to satisfyingly detect features that are in the image. The code remains almost unchanged from the example provided in this Lab. The lines of code that were changed to get the results in Figure 2 are given by:

  [y,x,m] = harris(grayImage, 5000, 'tile', [5 5]); points = [x, y];

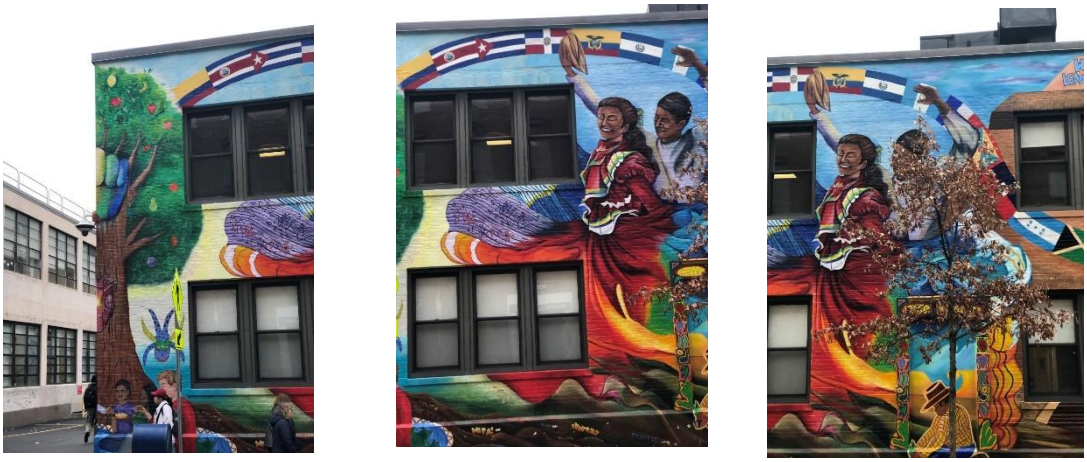
- Here,

  • Harris() Harris corner detector which has the ability to calculate the eigenvalues of the gradient matrix directly.

  • 5000 is the maximum number (N) of interest points to return. This value worked really well for this Image Data set.

  • tile is [5 5] break the image into regions [y x] to distribute feature points more uniformly. This value worked really well for this Image Data set.

  • points = [x, y]; converts the x and y of size M × 1, where M is the number of points, to a matrix of M × 2 i.e. appending the two values together.



(a) (b) (c)

(d) (e) (f)

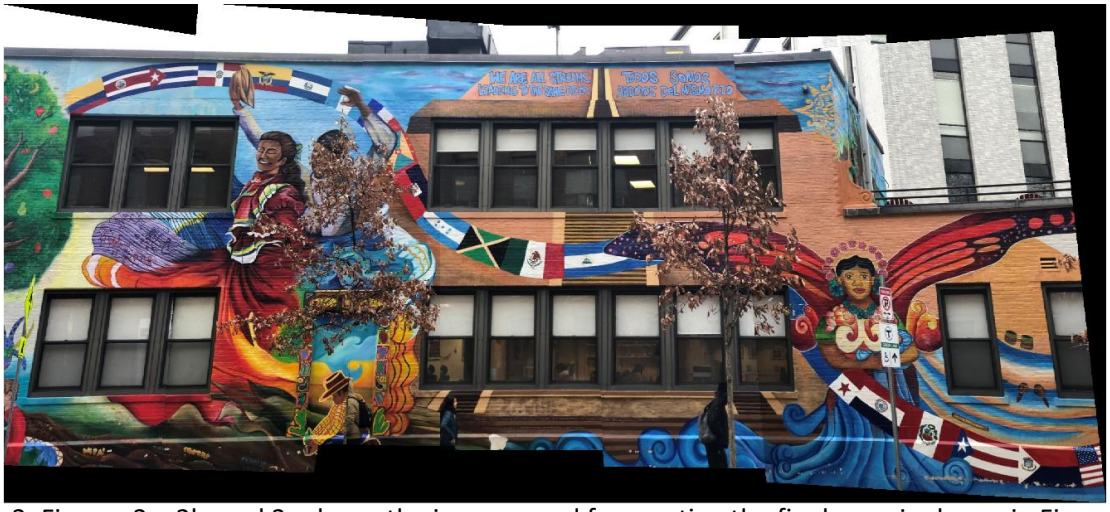
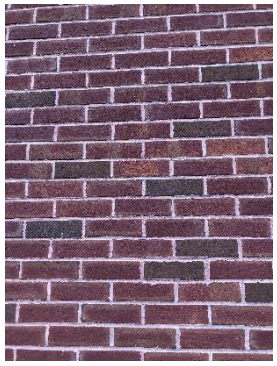


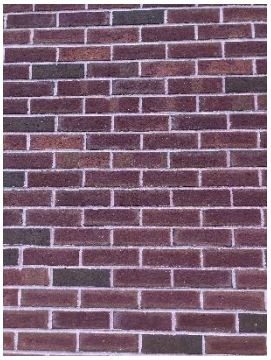
Figure 2: Figures 2a, 2b, and 2c shows the images used for creating the final mosaic shown in Figure 2g. Figures 2d, 2e, and 2f show the Harris Corner Detection Points superimposed over original images respectively.

3. Brick Wall Images

(a)



(b)

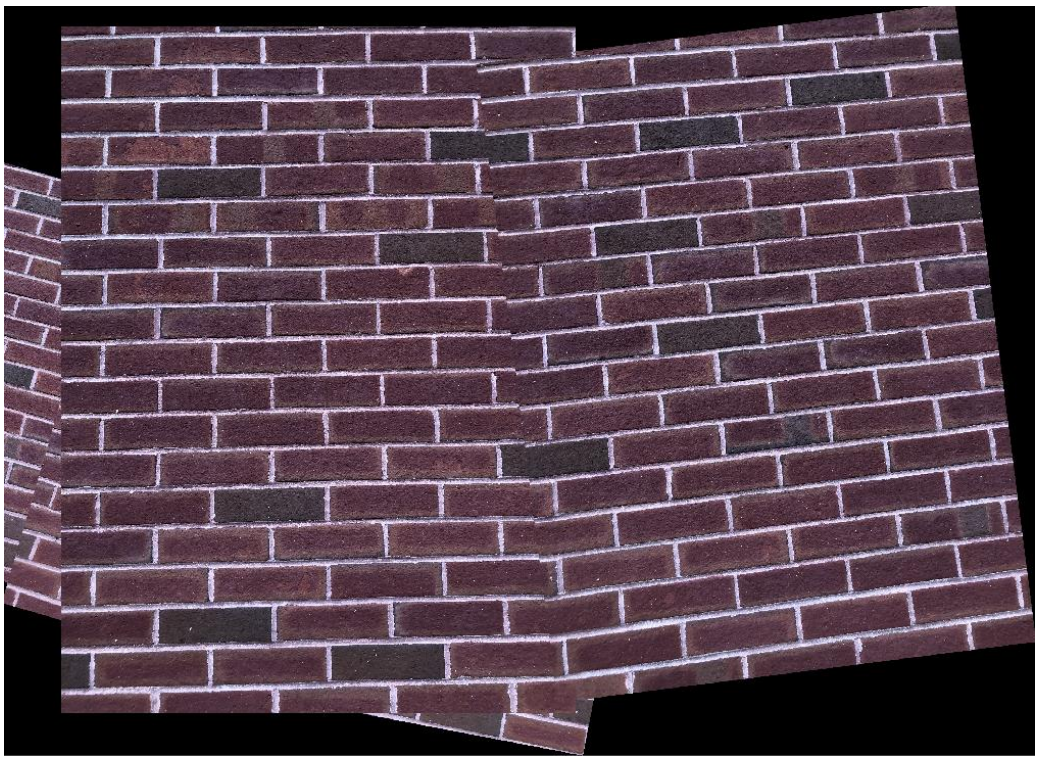


(c)

(d)

(e)

(f)



3.2. Discussion

I have used five images to create the panorama. But for now, I will show the first three images and will show all the images in the last section of the report. Compared to LSC, I had to make the following code changes to make the brick wall panorama:

tforms(numImages) = simtform2d;
tforms(n) = estgeotform2d(matchedPoints, matchedPointsPrev, ...
'similarity', 'Confidence', 99.9, 'MaxNumTrials', 2000);

Moreover, the performance is poor and the images are not stitched properly. This can be because of the repeating pattern of the brick wall making it hard for it to stitch the images. Finally, since it focuses on minor variations in the texture of the brick and not on the corners, which is the distinguishing feature, it is not able to stitch the images properly which can be seen in Figure 3g.

4. Third Mosiac 4.1. Images 4.1.1. 15% Overlap

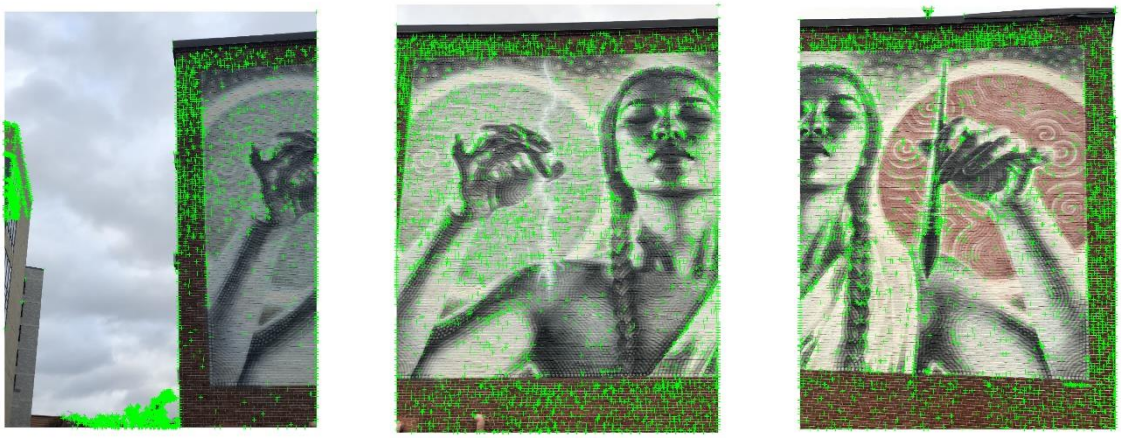
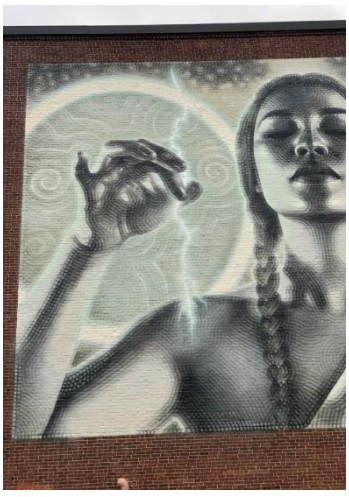
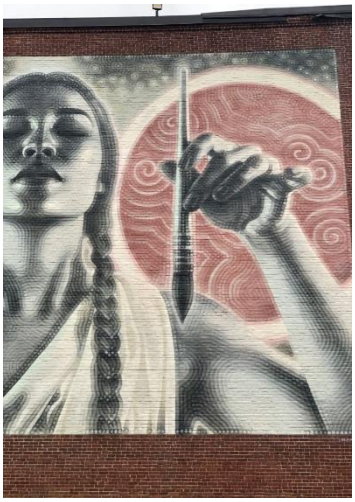


Figure 4: Figures 4a, 4b, and 4c shows the images used for creating the final mosaic shown in Figure 4g. Figures 4d, 4e, and 4f show the Harris Corner Detection Points superimposed over original images respectively.

4.1.2. 50% Overlap





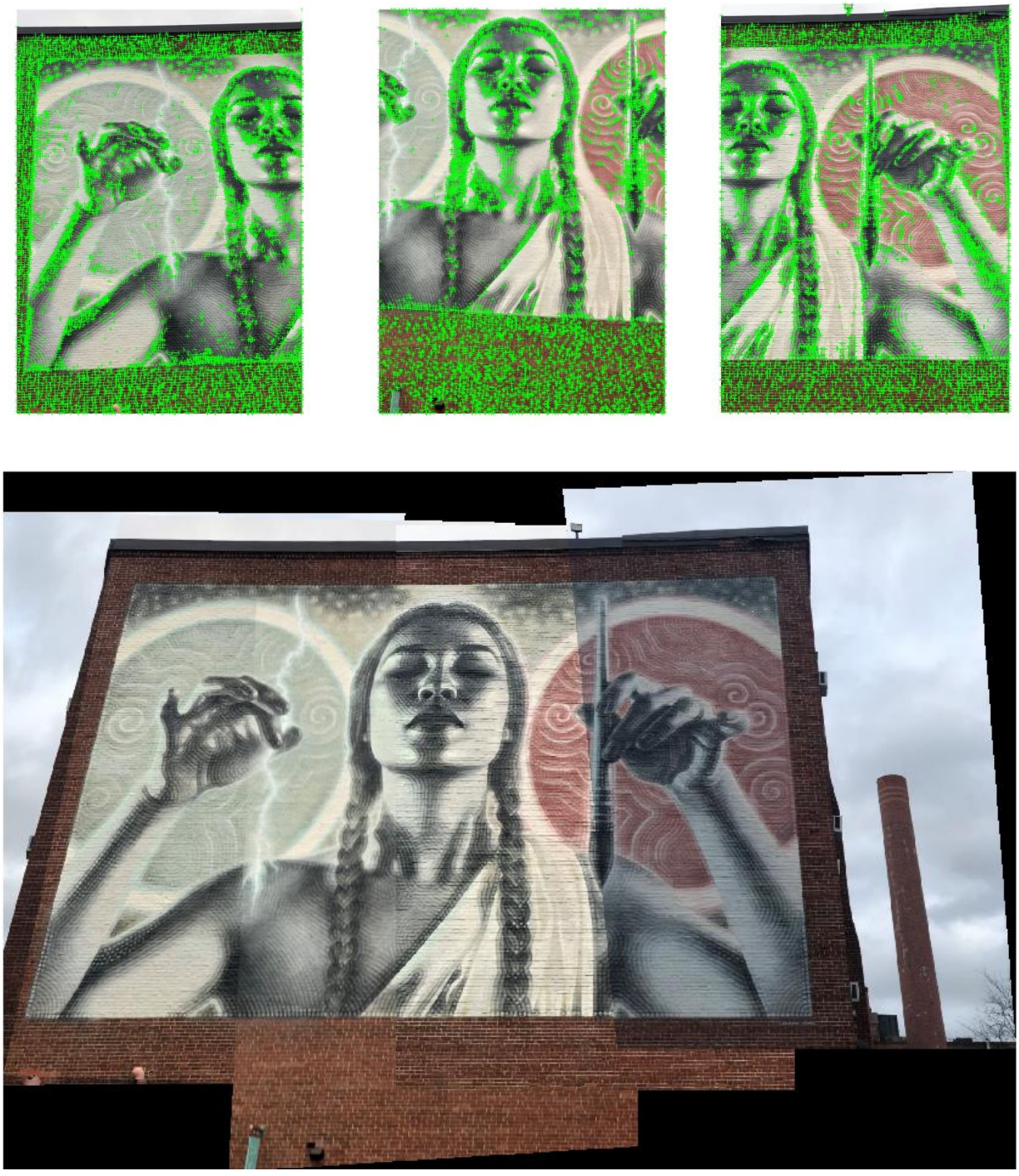


Figure 5: Figures 5a, 5b, and 5c shows the images used for creating the final mosaic shown in Figure 5g. Figures 5d, 5e, and 5f show the Harris Corner Detection Points superimposed over original images respectively.

4.2. Discussion

- For 15% Overlap, I have used four images to create the panorama. But for brevity, I will show the first three images and will show all the images in last section of report
- For 50% Overlap, I have used five images to create the panorama. But for brevity, I will show the first three images and will show all the images in last section of the report
- The only difference in between 15% and 50% coverage is the tiling and Maximum number of points to return in the Harris detector. This can be seen below:

  % 15% Overlap: [y,x,m] = harris(grayImage, 5000, 'tile', [3 3]);

  % 50% Overlap: [y,x,m] = harris(grayImage, 10000, 'tile', [2 2]);

Latinx Student Cultural Center (LSC) Images

Figure 8: Latinx Student Cultural Center (LSC) Montage.

harris corner detection

Figure 9: Latinx Student Cultural Center (LSC) with Harris Corner Detection Montage.

Brick Wall Images

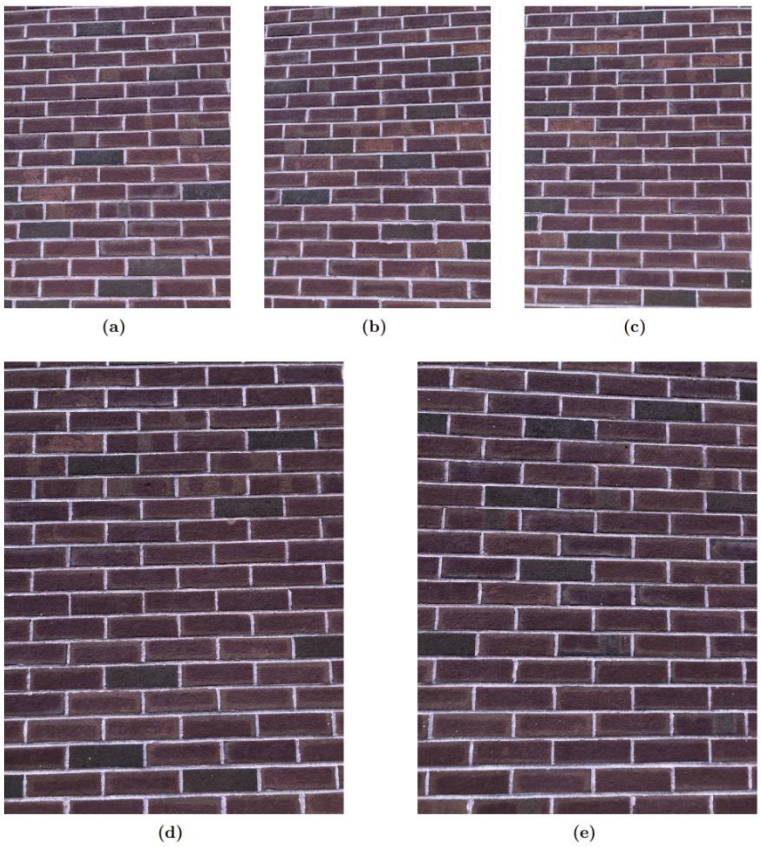


Figure 10: Brick Wall Montage.

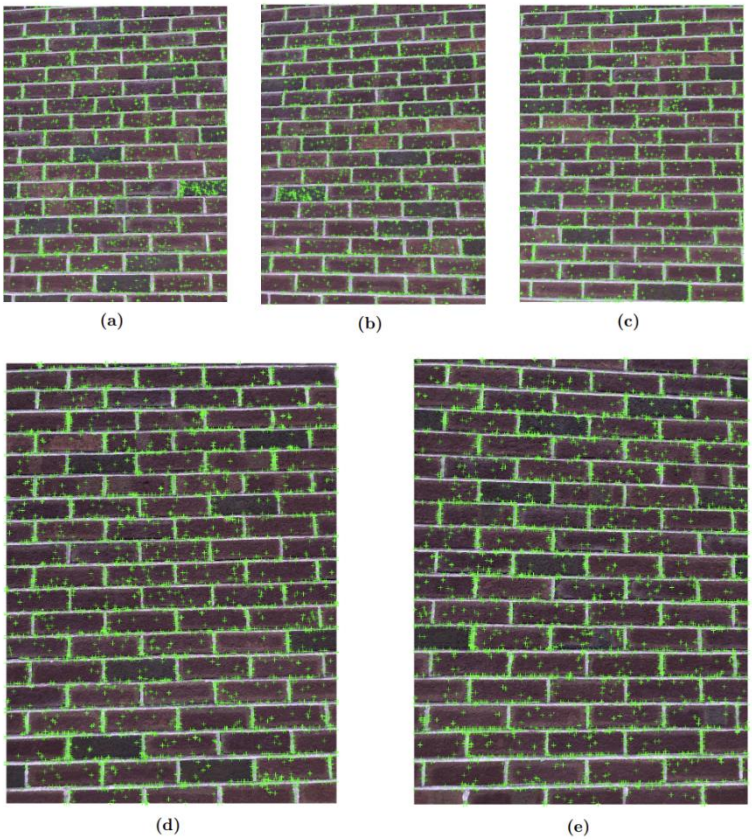


Figure 11: Brick Wall with Harris Corner Detection Montage.

Third Mosiac 15% Overlap

A4.1. 15% Overlap

(a) (b) (c) (d)

Figure 12: Third Mosaic with 15% Overlap

(a) (b) (c) (d)

Figure 13: Third Mosaic with 15% Overlap with Harris Corner Detection Montage.

Third Mosiac 50% Overlap

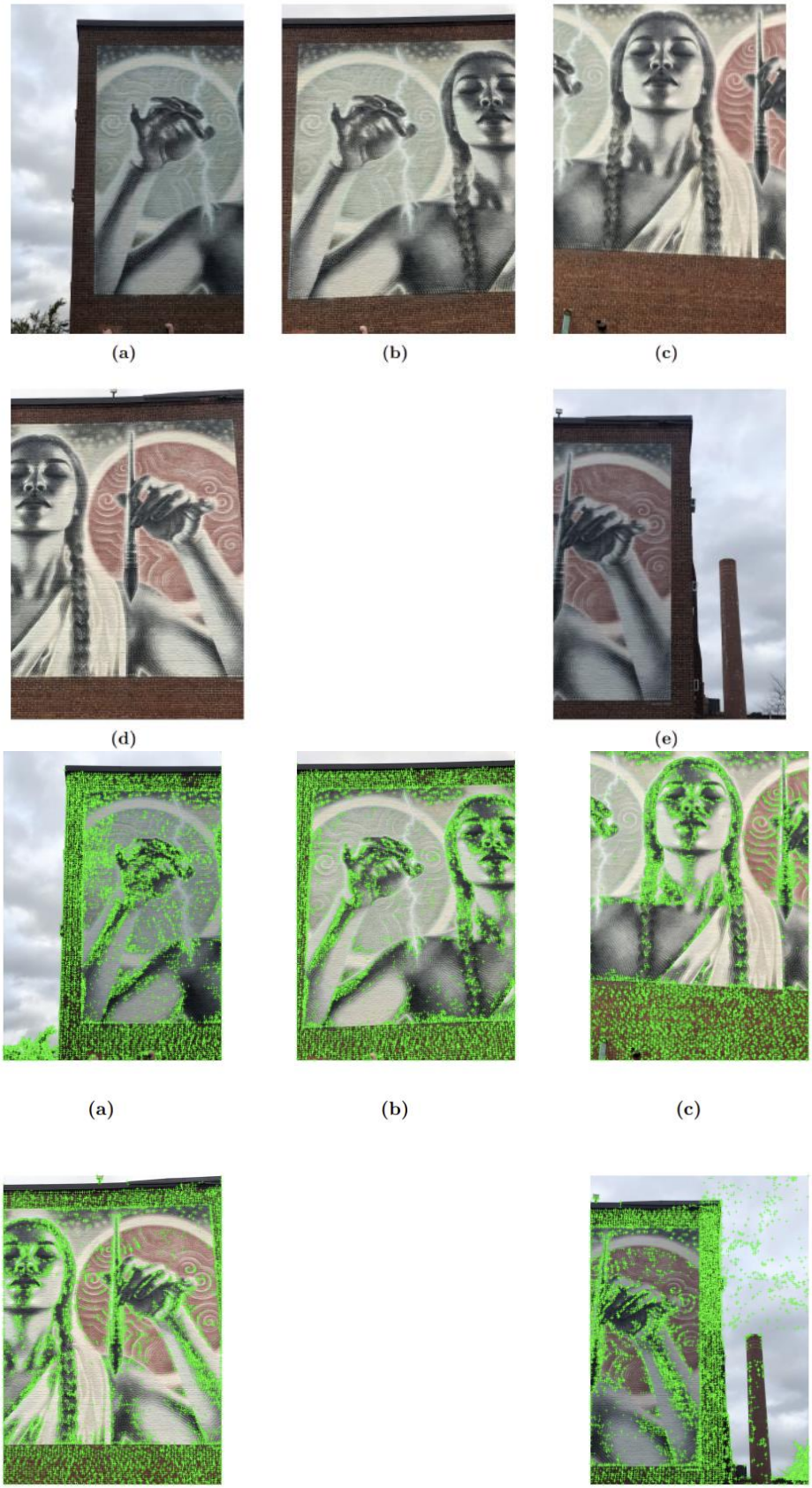


Figure 15: Third Mosaic with 50% Overlap with Harris Corner Detection Montage.