

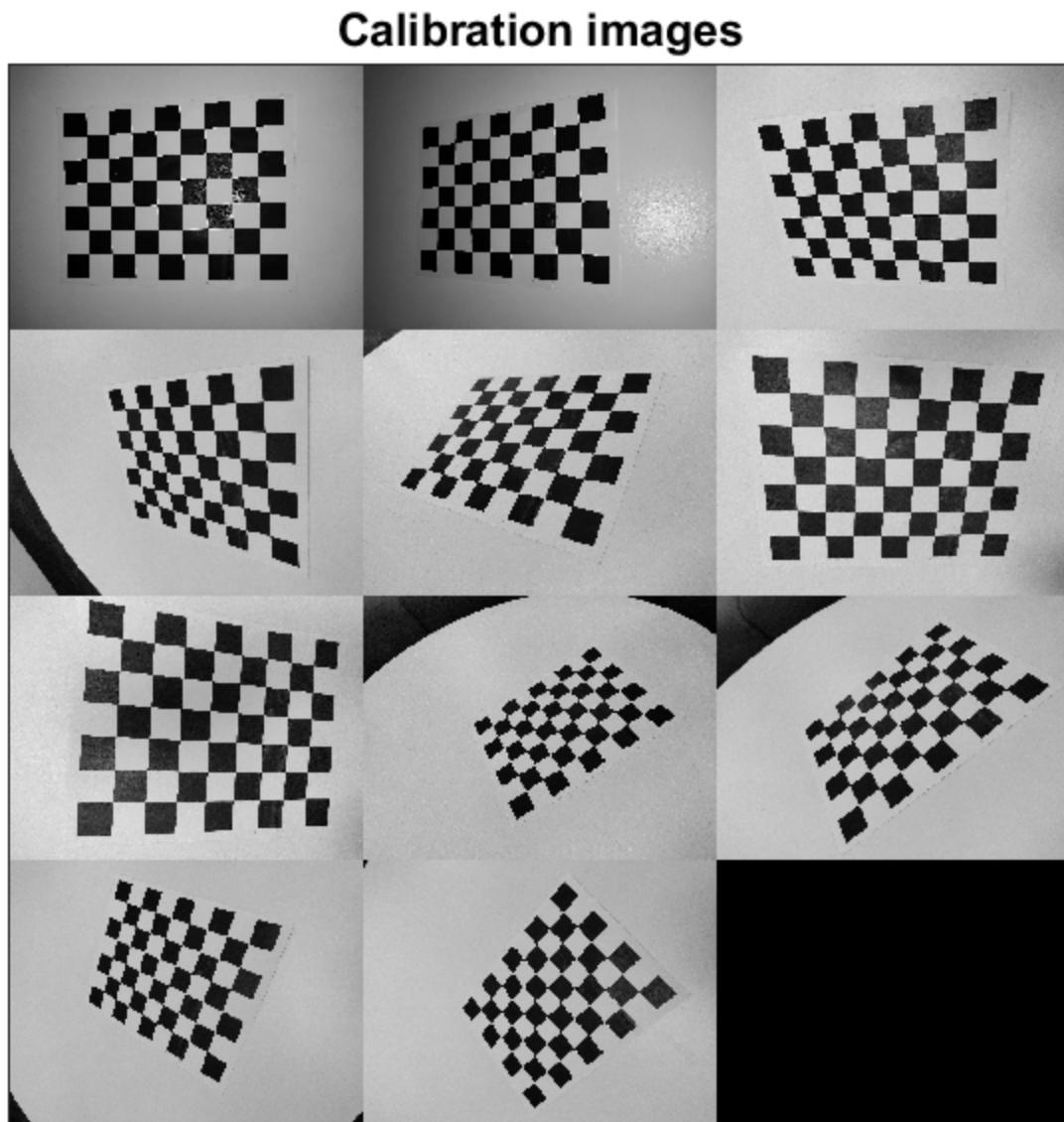
# EECE 5554 Robotics Sensing and Navigation

## Lab-5 Report

---

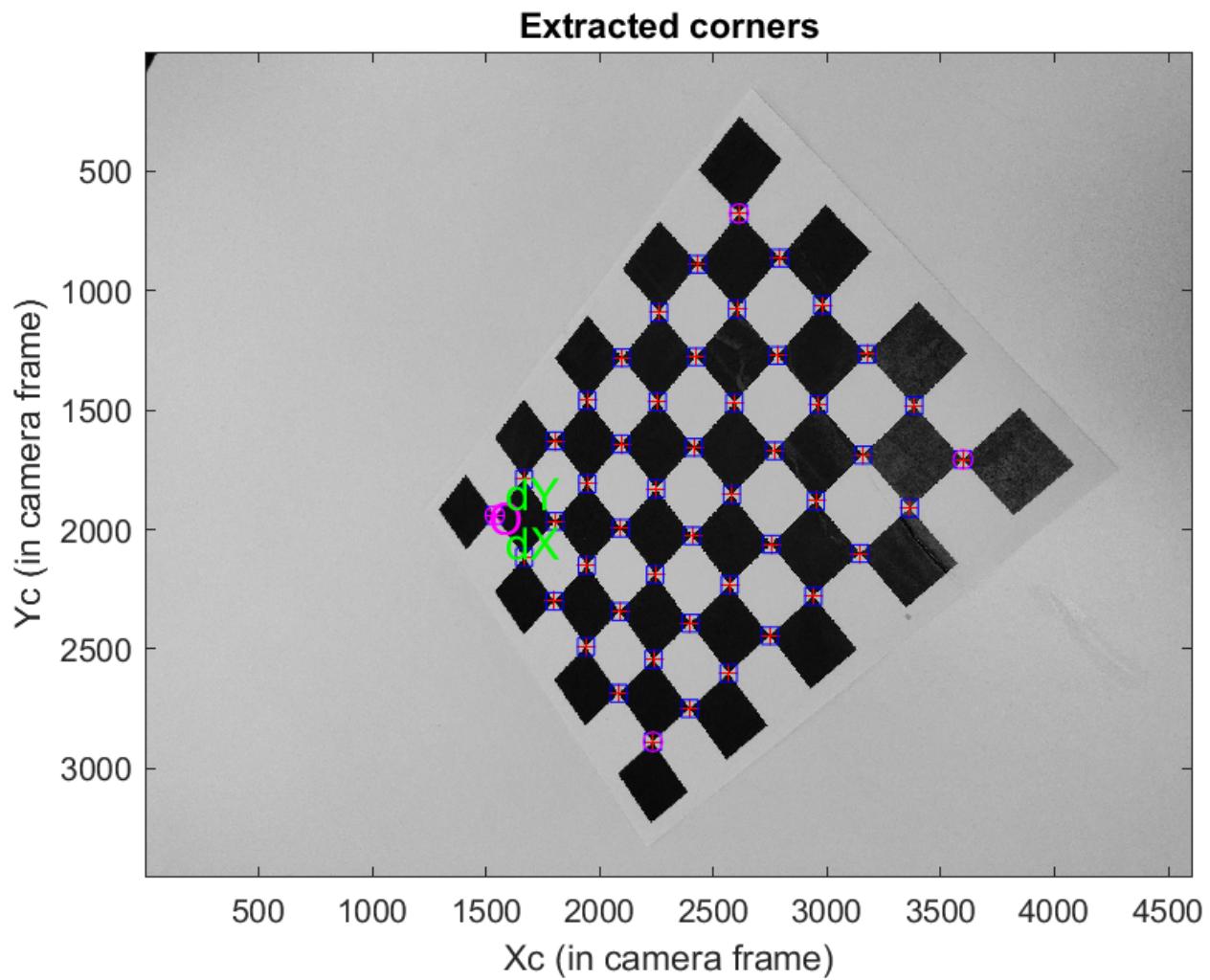
### Camera Calibration:

#### 1. Camera images used for calibration:



**Fig-1 Set of images used for calibration**

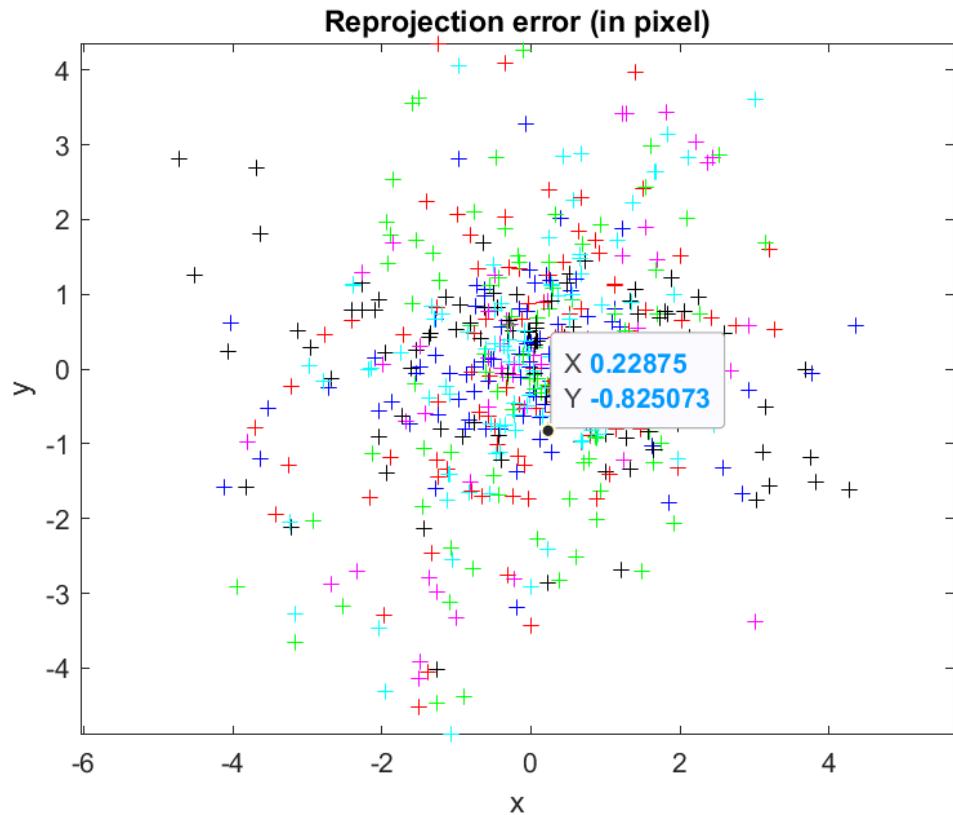
---



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

---

## 2. Reprojection pixel error:



**Fig-3 Reprojection error in pixels**

---

### 3. 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 initialisation

Pixel error, as seen in fig-4 is [2.39 2.27] 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 ]
```

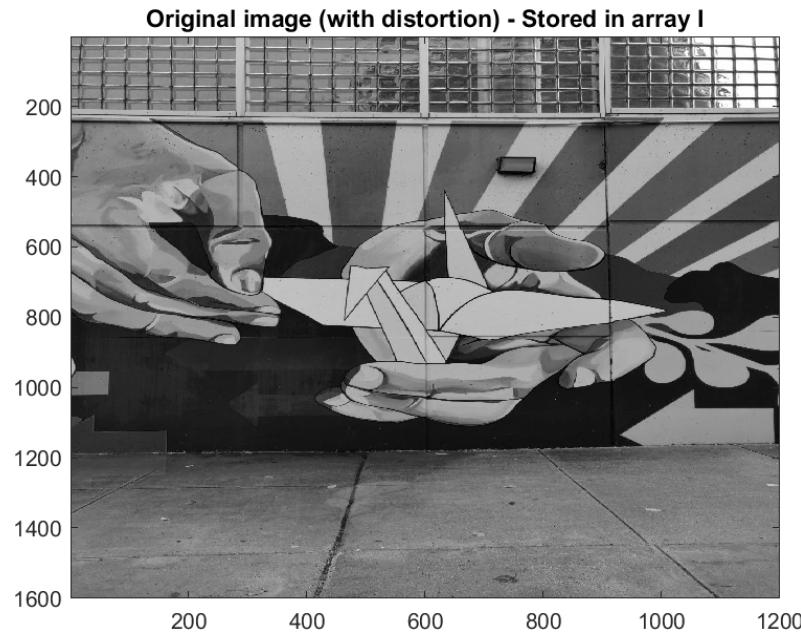
```
Note: The numerical errors are approximately three times the standard deviations (for reference).
```

### 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.

---

#### 4. An image before and after calibration is in report:



**Fig-6 Original Image  
With distortion**



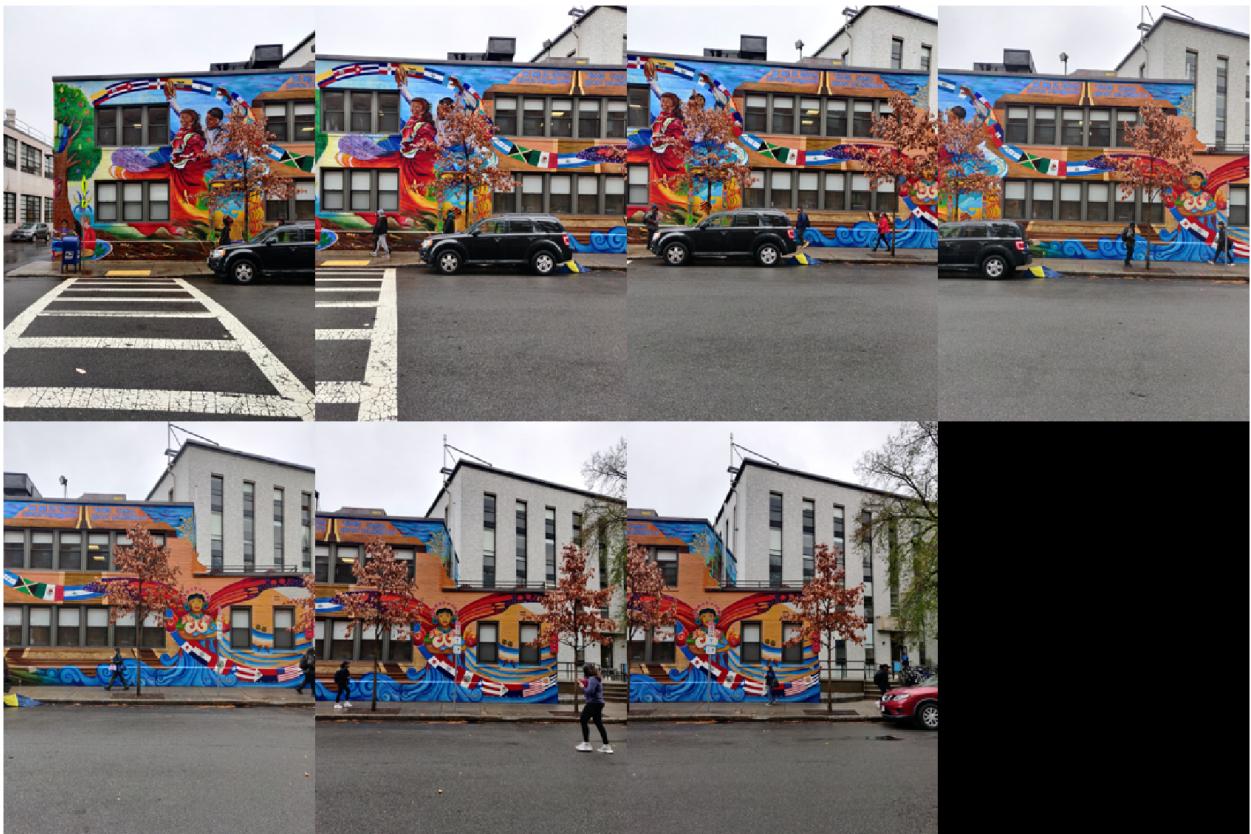
**Fig-7 Original Image  
Without distortion**

---

## LSC mosaic

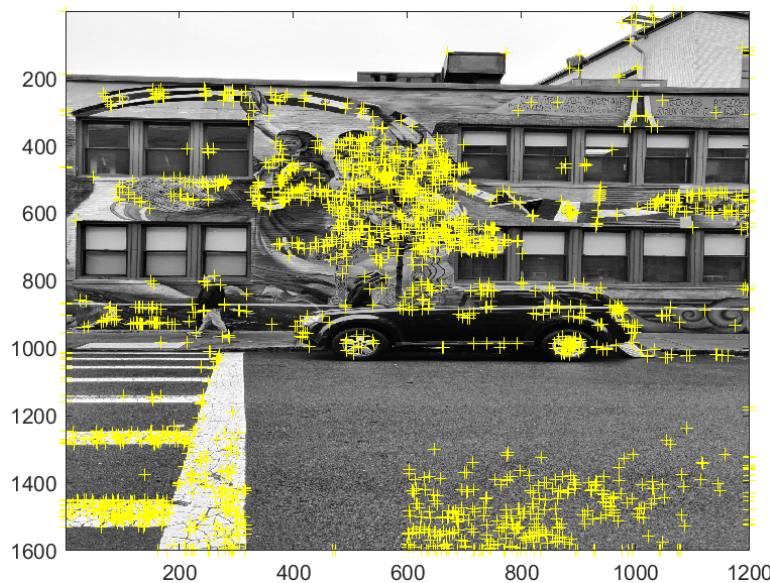
---

## Image set

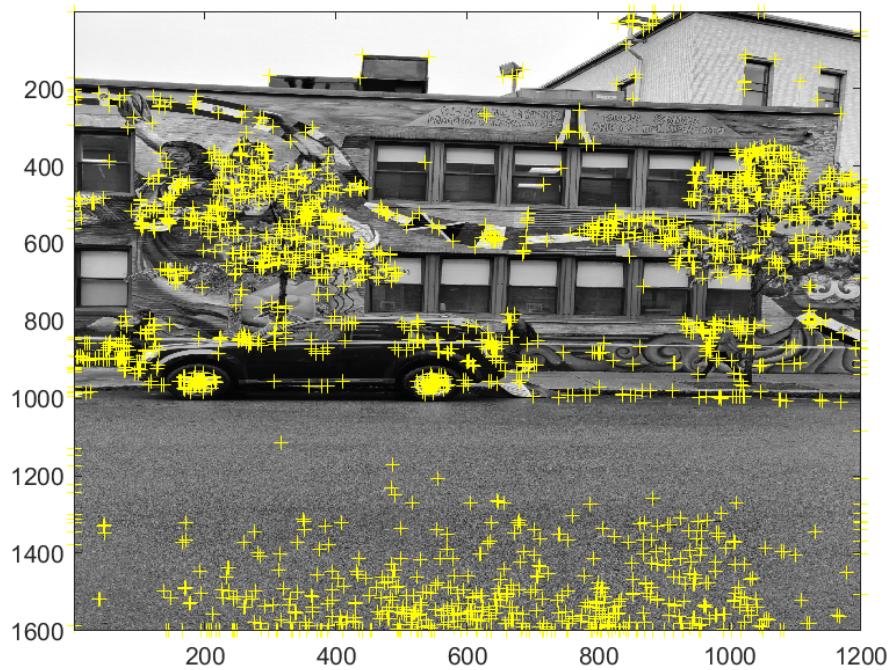


**Fig-8 LSC image set**

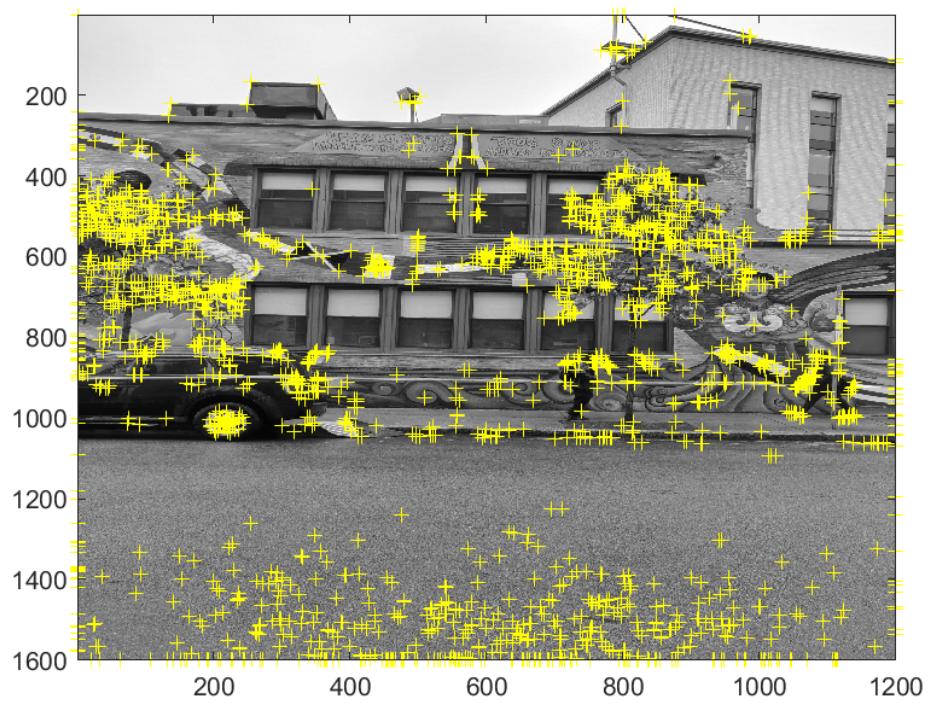
### Distribution of Harris corners across LSC image set:



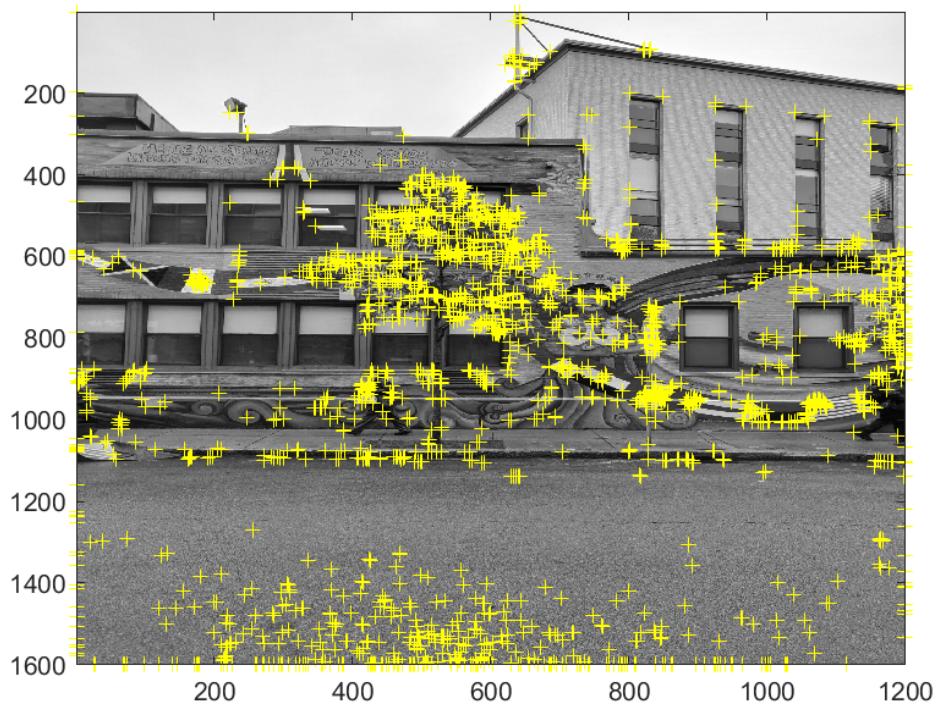
**Fig-9.1**



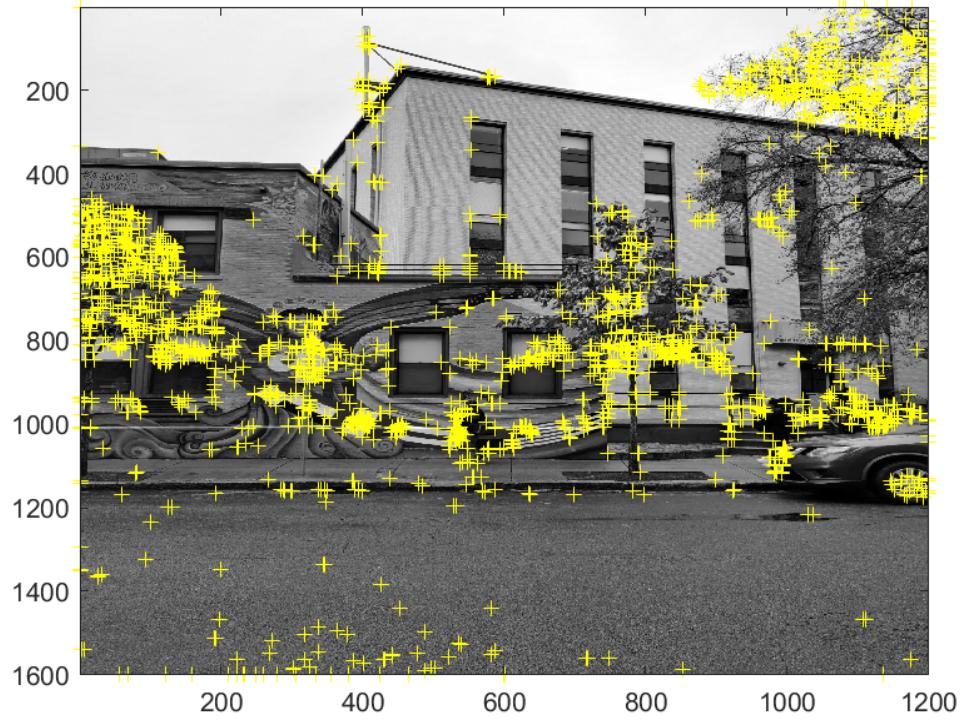
**Fig-9.2**



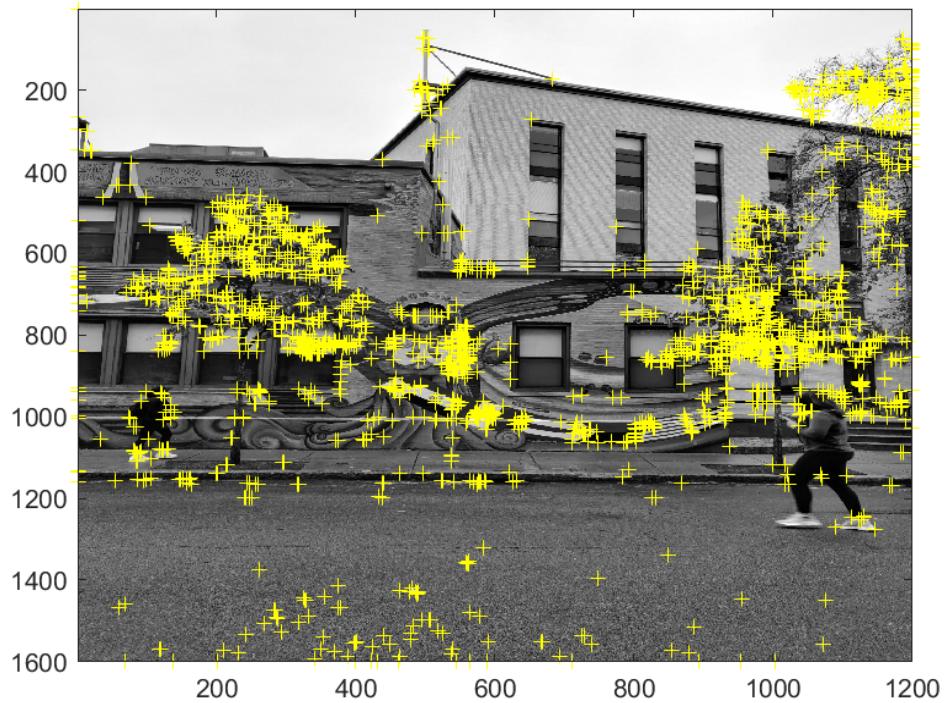
**Fig- 9.3**



**Fig-9.4**



**Fig-9.5**



**Fig-9.6**



**Fig-9.7**

---

## Final LSC mosaic

The image stitching was done by using projective transformation and Affine transformation. The results of the same are presented below

### Mosaic after using Projective transformation:



**Fig-10 Mosaic of LSC graffiti using Projective transformation**

---

## Mosaic after using Affine transformation



**Fig-11 Mosaic of LSC graffiti using Affine transformation**

Since the images were already calibrated by the mobile's camera and since further calibration of the image is causing errors for stitching the images we have directly used the camera images.

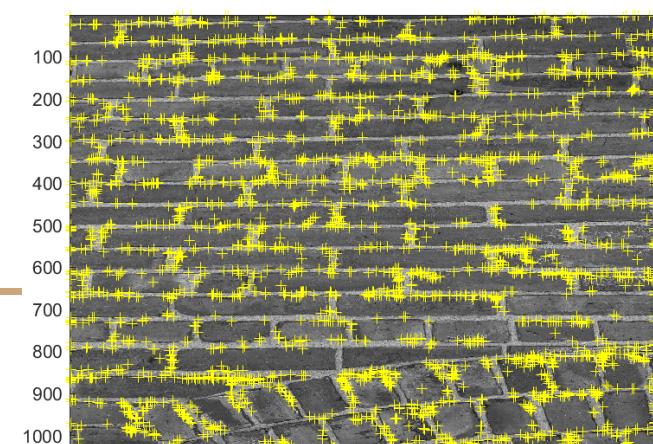
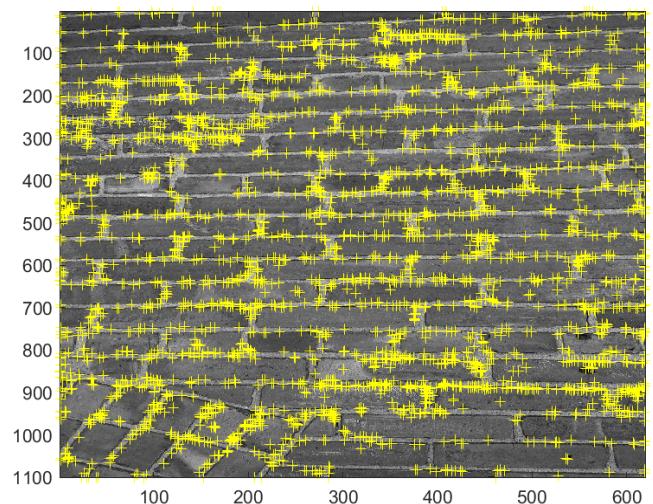
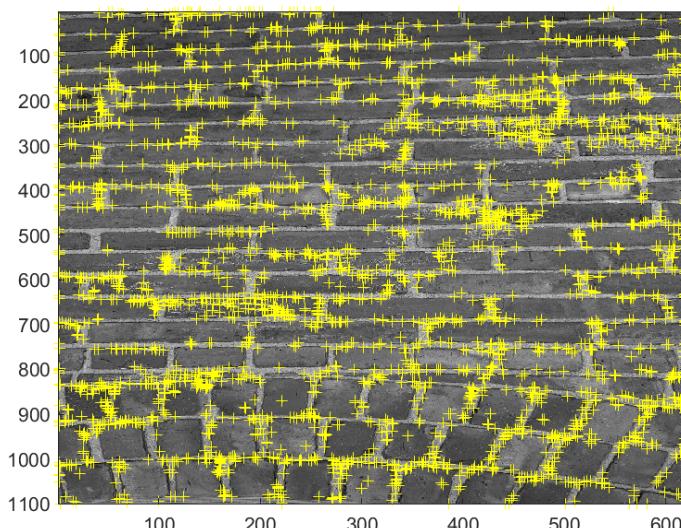
For the image mosaic using the harris features detector after reading the images, harris features were detected utilizing the harris.m file which was called in pano.m file and once the features were detected and plotted, using

```
tforms(n)=estgeotform2d(matchedPoints,matchedPointsPrev,"affine")
```

**"Projective"** ,**"Confidence"**,99,**'MaxNumTrials'**, 2000); where the confidence remained unchanged while using both the transformations. So, except obtaining the Harris corners and storing those corner points in points which were later used to compare matched points and the previous points in transformations to obtain the image mosaic no other changes were made to the parameters

## Cinder block/brick wall “mosaic”

### Initial images with Harris corners



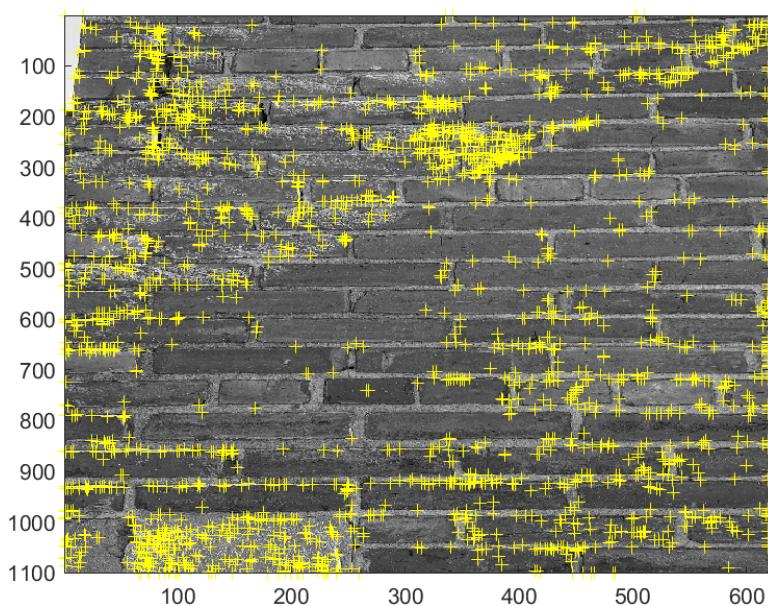
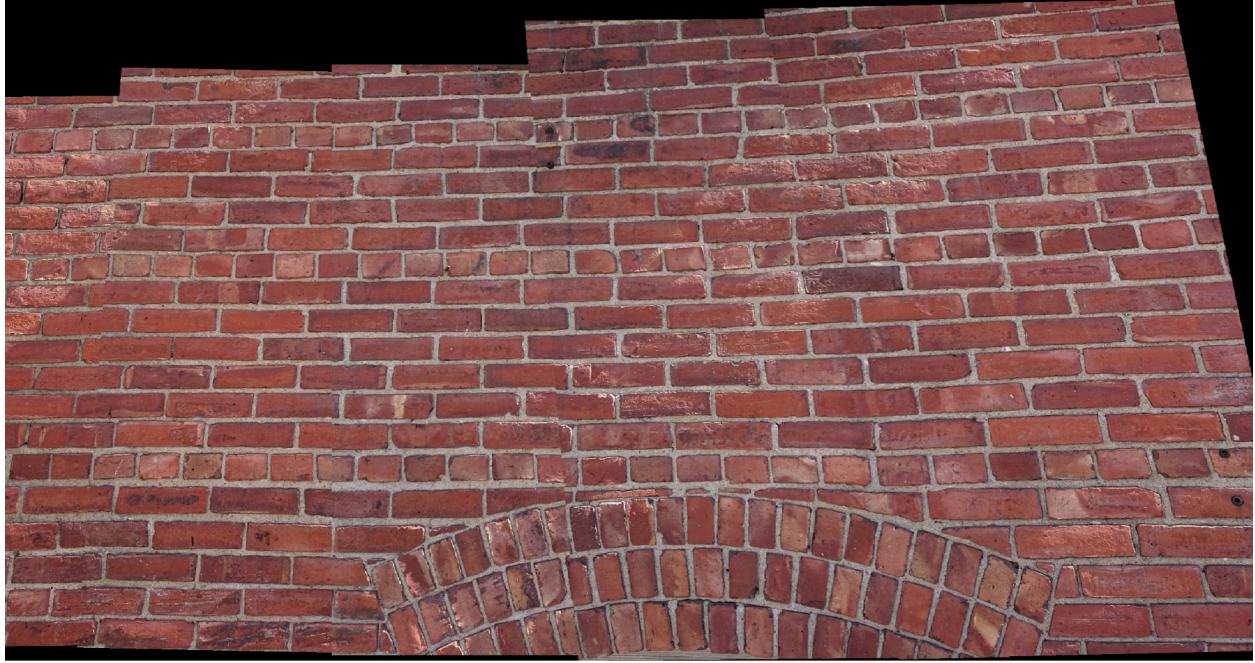


Fig-12(All cinder block images with harris corners)

---

## Final cinder block image



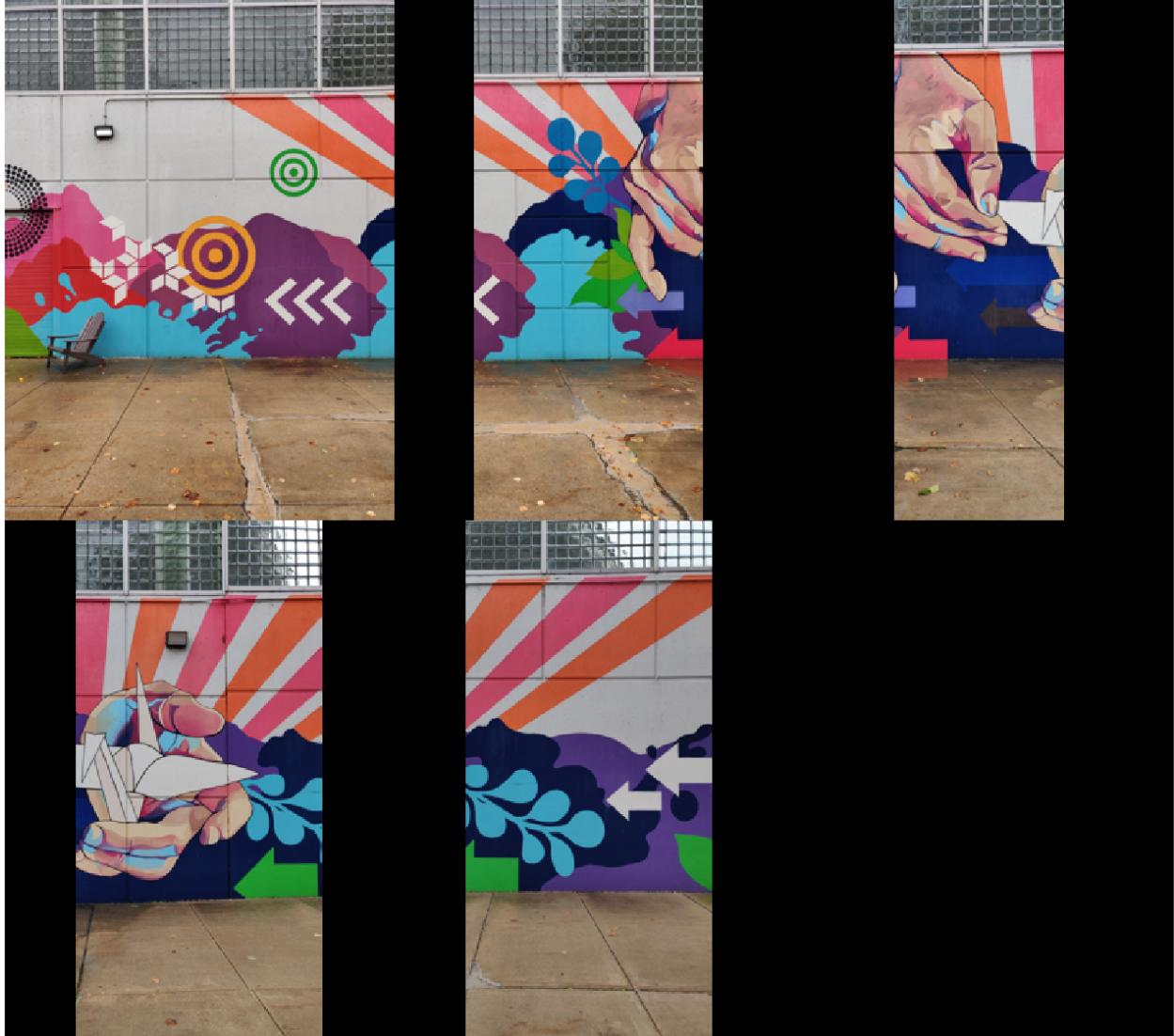
**Fig-13 Cinder block mosaic**

While making the mosaic of the Cinder blocks errors related to “morzcat” and “imwarp” and also a singularity error so the number of images was reduced to 5 and since the images were taken not from the cinder blocks as per the matlab’s prescription affine transformation was used. The tile size, confidence etc all were the same.

---

## Third “mosaic”

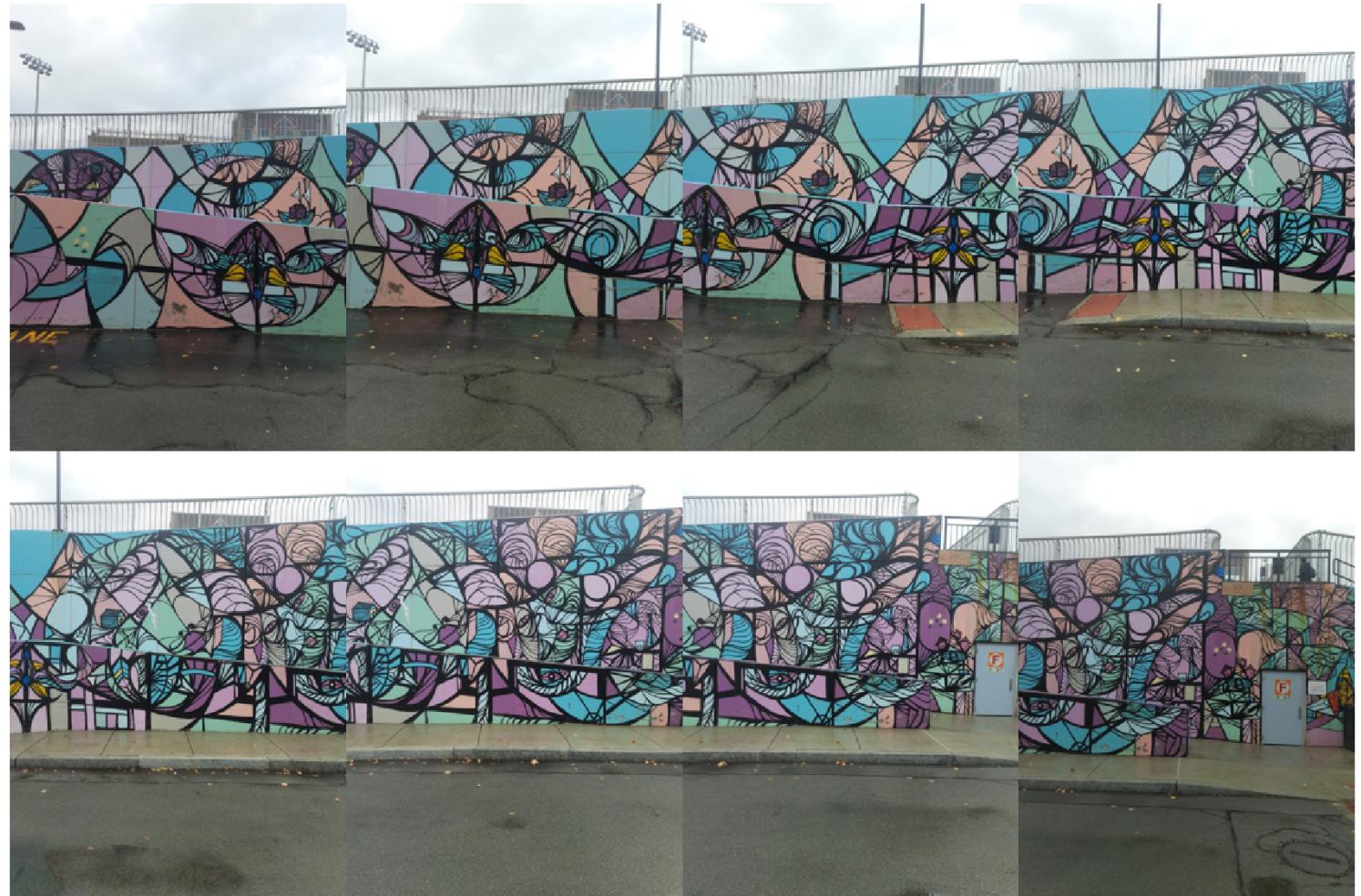
Initial images with Harris corners with 15% overlap



**Fig-14 15% overlap images**

---

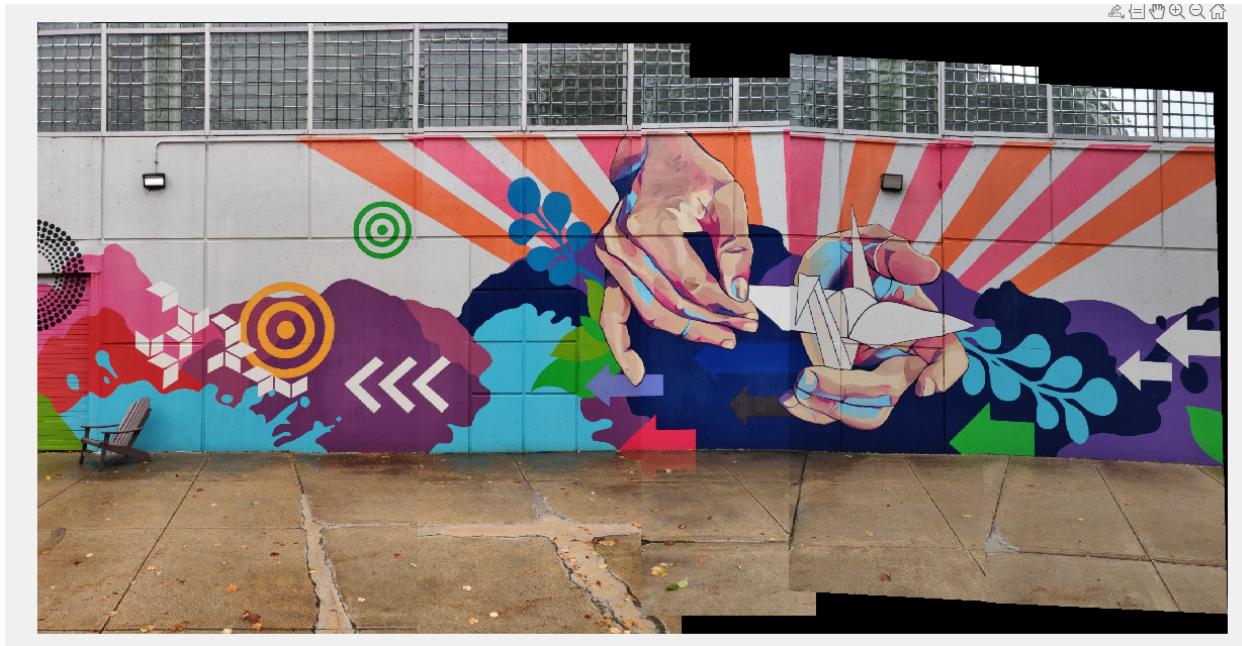
## Initial images with Harris corners with 50% overlap



**Fig-15 50% overlap images**

---

## Final mosaic with 15% overlap

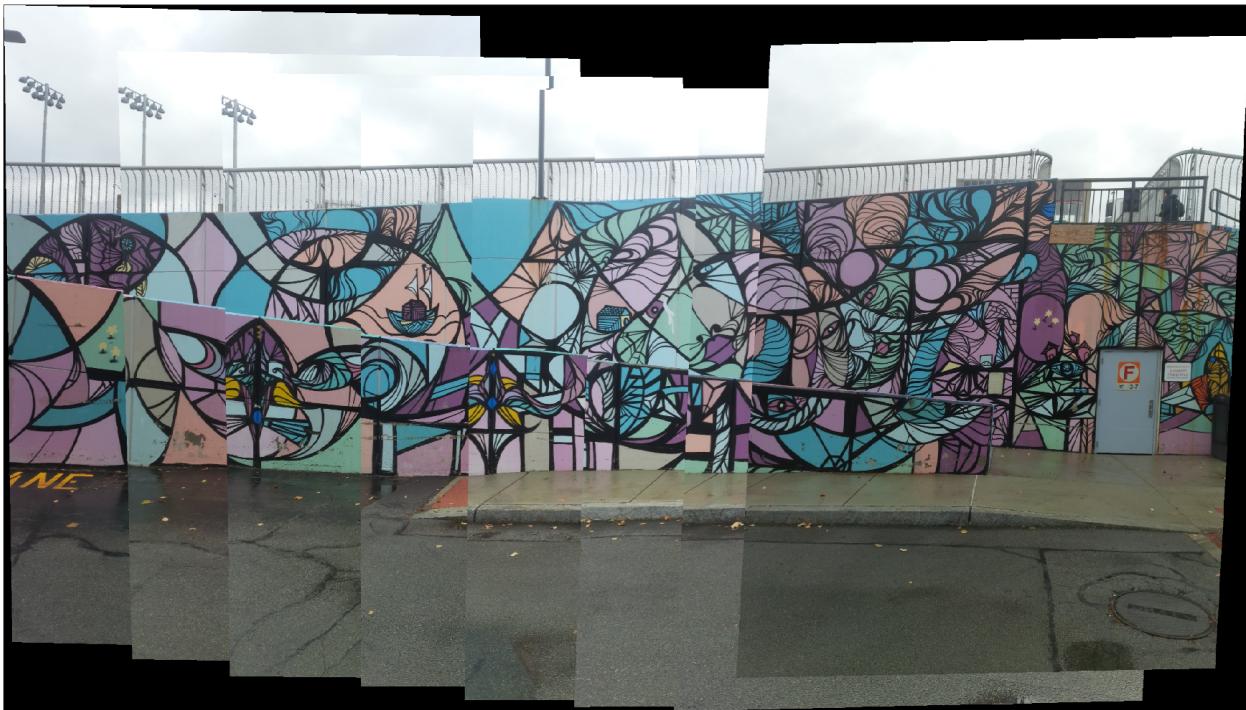


**Fig-16 15% overlap images Mosaic**

For the case of 15% overlap images, the tile size initially was set to [3 3], the maximum number of interest points was set to 1000 initially and while going through all the images the tile size and interest points were changed back to the [2 2] and 2000 respectively.

---

## Final mosaic with 50% overlap



**Fig-17 50% overlap images mosaic**

There were no changes made to the 50% overlap images code. The code has run without any errors and the certain dominant features which were highlighted with harris features were considered and compared to later make image stitching but certain features like light poles etc which weren't seen in the harris features points have occurred in multiple frames with improper outputs.