CSE 344: Computer Vision

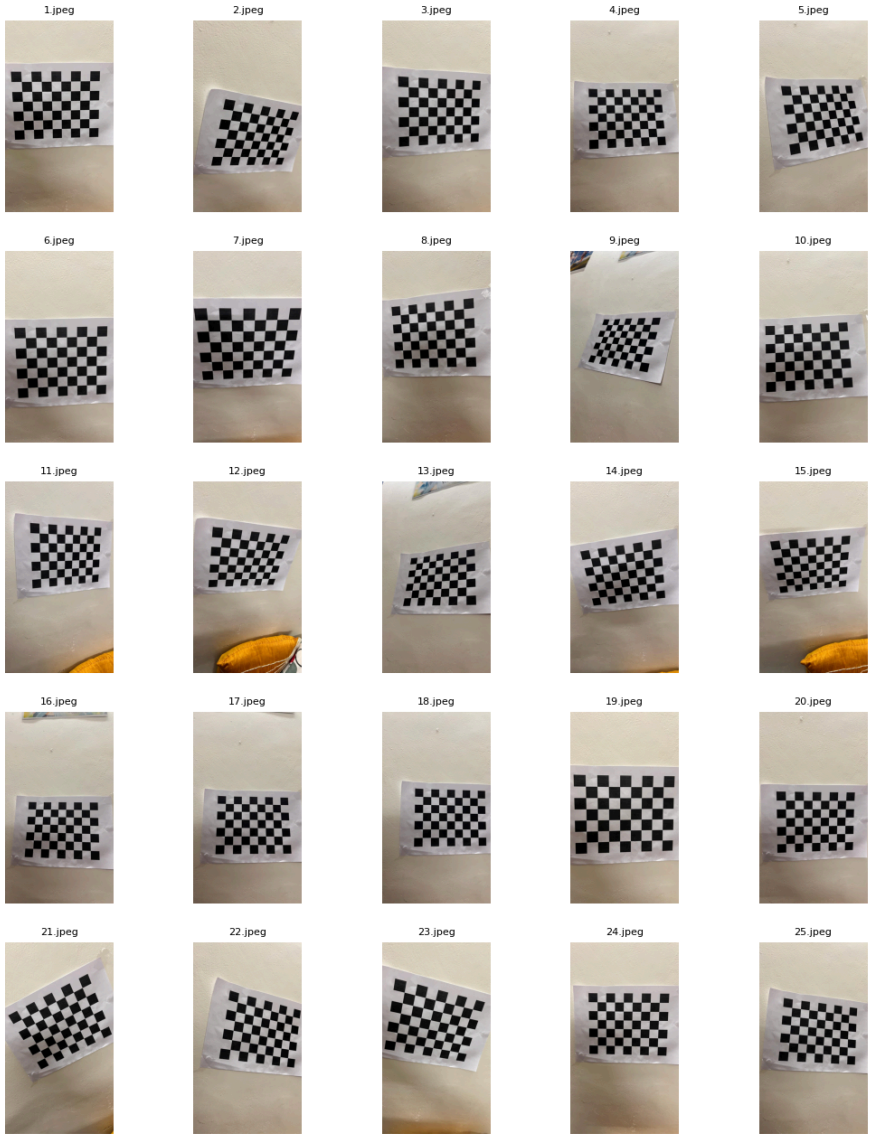
Assignment 2: Report

Aditya Aggarwal 2022028

---

1. Camera Calibration

Given Dataset:



Custom Dataset:

chessboard_dataset

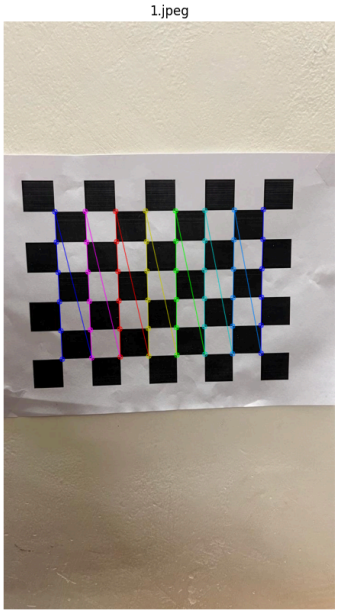
1.0 Corner Detection

Camera Calibration using OpenCV

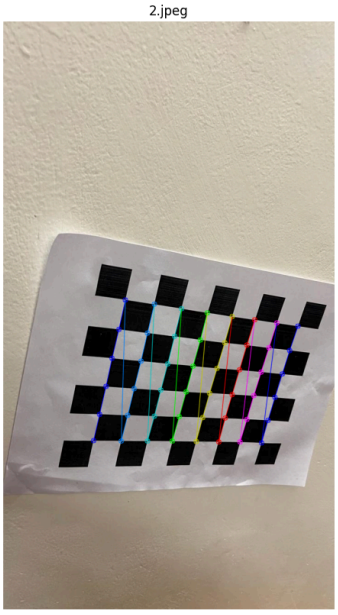
Detect and refine chessboard corners in a set of calibration images of (6,8) rows and columns using OpenCV functions. Convert each image to grayscale and locate the checkerboard corners based on a predefined pattern.

Given Dataset:

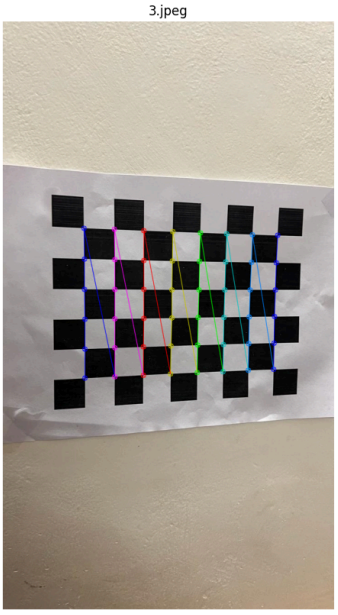
1.jpeg



2.jpeg

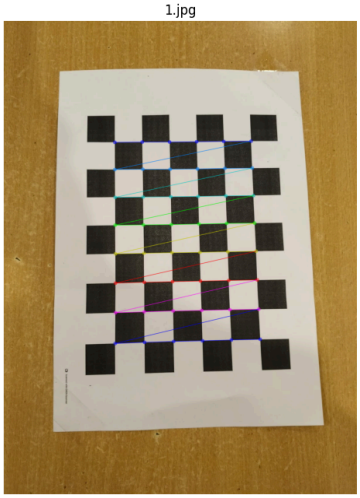


3.jpeg

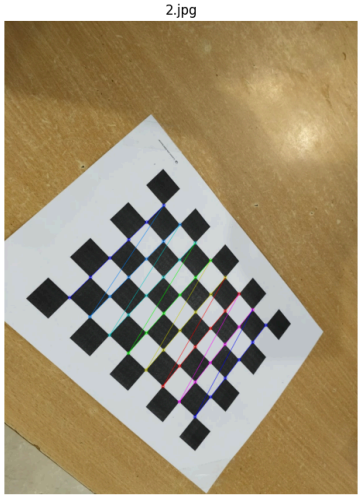


Custom Dataset:

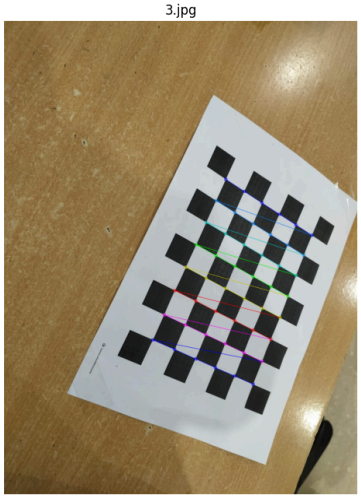
1.jpg



2.jpg



3.jpg



1.1 Intrinsic Parameters

Given Dataset:

```
{'focal_length': [956.6422400596589, 957.5599353826644],
 'skew': 0.0,
 'principal_point': [369.0232099213968, 651.4114714908994]}
```

Custom Dataset:

```
{'focal_length': [1218.088941376052, 1232.5553902758186],
 'skew': 0.0,
 'principal_point': [591.5126121445984, 803.9696071714228]}
```

1.2 Intrinsic Parameters (First 2 Images)

Given Dataset:

```
[{'image_id': '1.jpeg',
  'rotation_matrix': [[0.013871573976875209,
    -0.9970004266867422,
    -0.07614281727030023],
   [0.9885615437433356, 0.025111636003789333, -0.1487127431291831],
   [0.15017873906514623, -0.0732089811680742, 0.9859446188347172]],
  'translation_vector': [3.0411995933966915,
   -3.7929121167838735,
   15.171692040709857]},
 {'image_id': '2.jpeg',
  'rotation_matrix': [[0.2419429839445728,
    0.8905346282603145,
    -0.3852423502021286],
   [-0.9430765413866279, 0.1224538922043366, -0.30921138622347766],
   [-0.2281890216601845, 0.4381245486768255, 0.869469177280308]],
  'translation_vector': [-2.7069553381468427,
   4.0575120305983345,
   15.059177873803659]}]
```

Custom Dataset:

```
[{'image_id': '1.jpg',
  'rotation_matrix': [[0.9991404661804352,
    0.022309711615486727,
    -0.0349371665762495],
   [-0.0186174696148884, 0.9945427229507751, 0.10265555051138997],
   [0.03703672050656612, -0.10191687295679577, 0.994103230223515]],
  'translation_vector': [-2.303793648005554,
   -4.14981537176184,
   13.063347505560074]},
 {'image_id': '2.jpg',
  'rotation_matrix': [[0.658614824863372,
    0.6494431192715758,
    -0.3800659775631013],
   [-0.6983714392164676, 0.7156241143895954, 0.012627738943578645],
   [0.28018537677207506, 0.25711040767521126, 0.9248731766616314]],
  'translation_vector': [-3.8881303633537163,
   1.3852864773544284,
   13.076586145202226]}]
```

1.3 Radial Distortion Coefficients.

Given Dataset:

```
[0.19742550220623545, -0.7053337245225078, 0.04759644265609909]
```

Custom Dataset:

```
[0.16557763823987956, -0.9570989192419161, 1.9679548757448964]
```

Undistorted Images:

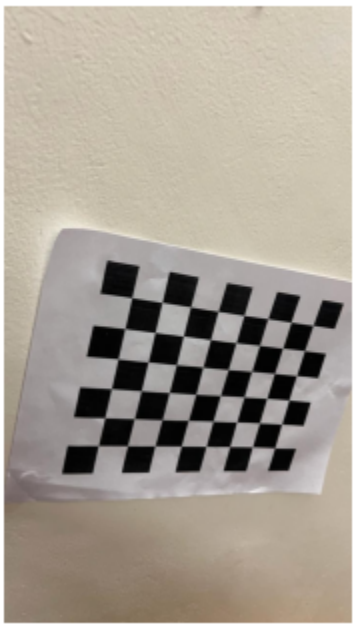
OpenCV Python Camera Calibration (Intrinsic, Extrinsic, Distortion)

Given Dataset:

Original: 1.jpeg

Undistorted: 1.jpeg

Original: 2.jpeg



Undistorted: 2.jpeg

Original: 3.jpeg

Undistorted: 3.jpeg

Original: 4.jpeg

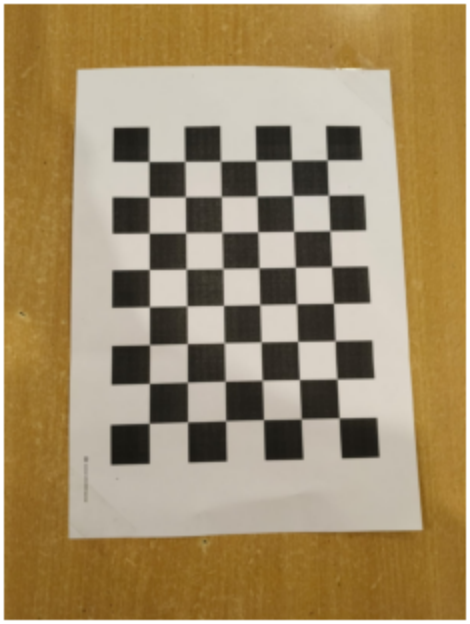
Undistorted: 4.jpeg

Original: 5.jpeg

Undistorted: 5.jpeg

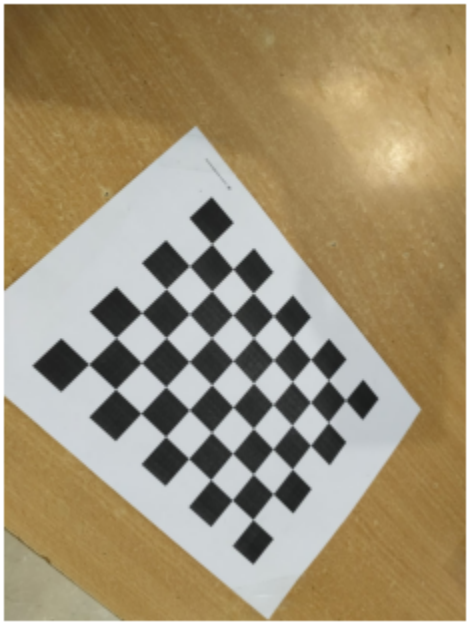
Custom Dataset:

Original: 1.jpg

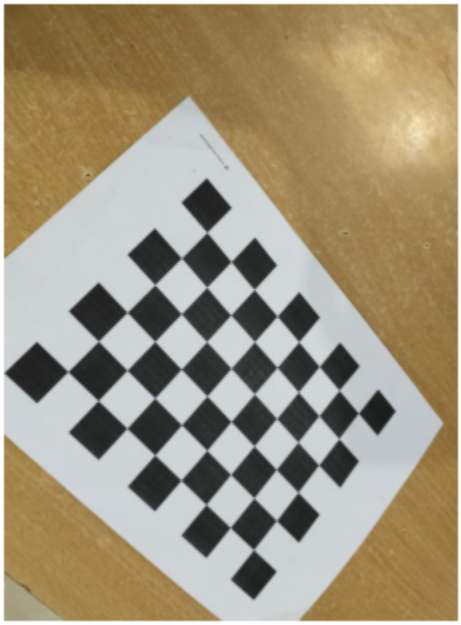


Undistorted: 1.jpg

Original: 2.jpg

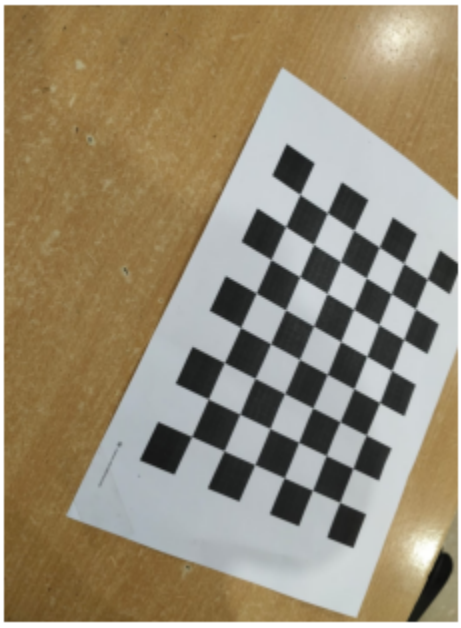


Undistorted: 2.jpg

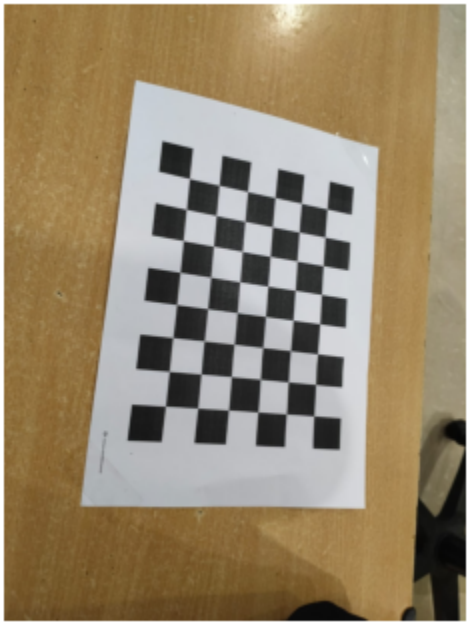


Original: 3.jpg

Undistorted: 3.jpg

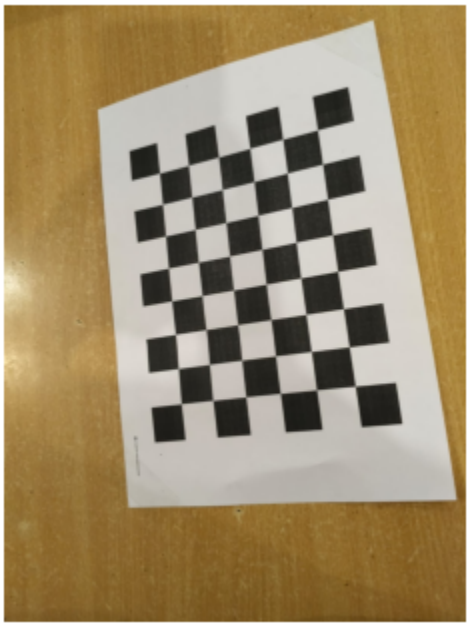


Original: 4.jpg



Undistorted: 4.jpg

Original: 5.jpg

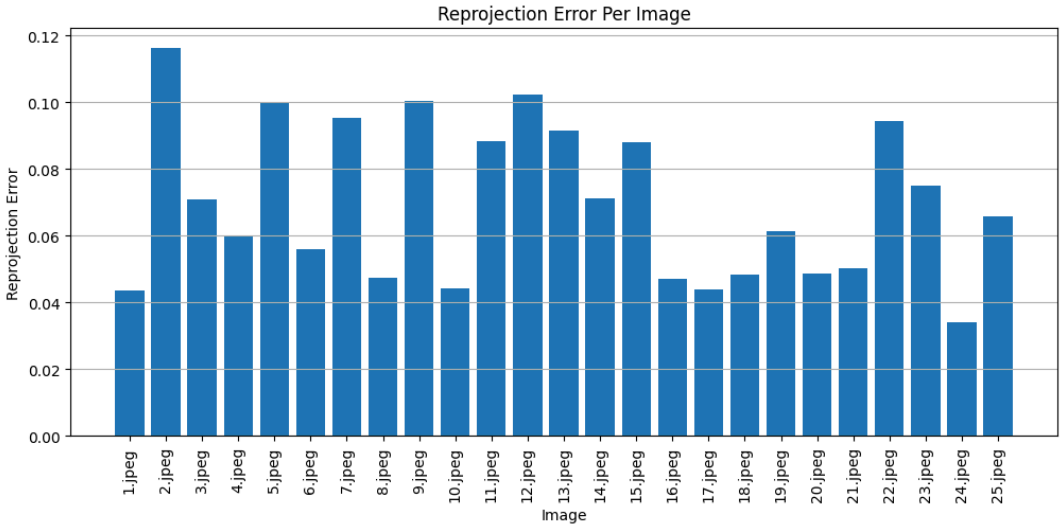


Undistorted: 5.jpg

1.4 Re-projection Error

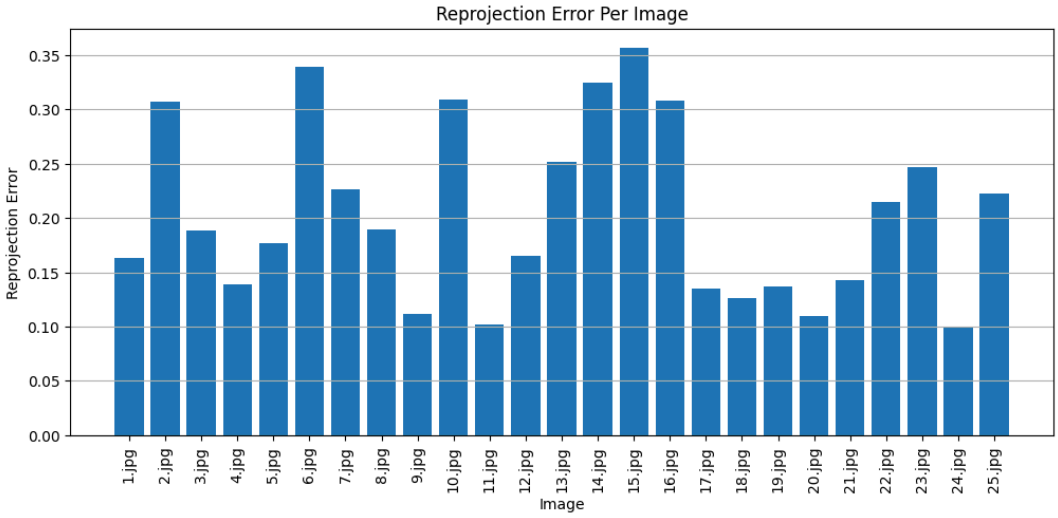
Learn Camera Calibration in OpenCV with Python Script

Given Dataset:




```
Mean Reprojection Error: 0.06971456911688327
Standard Deviation of Reprojection Error: 0.023215920420308844
```

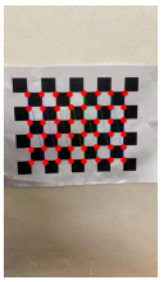
Custom Dataset:



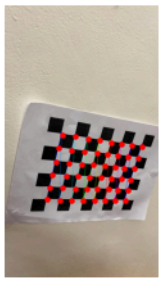

```
Mean Reprojection Error: 0.20372834975337686
Standard Deviation of Reprojection Error: 0.07989108265412283
```

Original Corners Detected (given dataset):

1.jpeg



2.jpeg

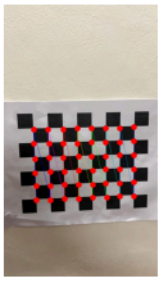


3.jpeg

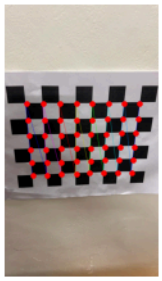
4.jpeg

5.jpeg

6.jpeg



7.jpeg



8.jpeg

9.jpeg

10.jpeg

11.jpeg

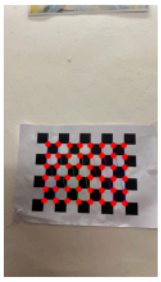
12.jpeg

13.jpeg

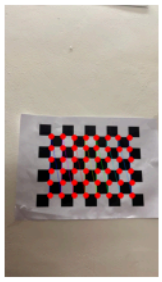
14.jpeg

15.jpeg

16.jpeg

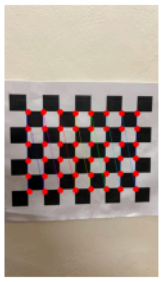


17.jpeg

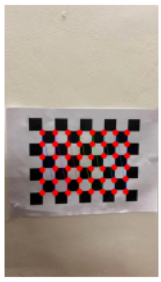


18.jpeg

19.jpeg



20.jpeg



21.jpeg

22.jpeg

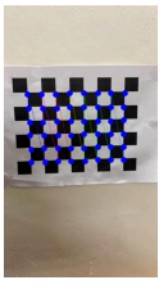
23.jpeg

24.jpeg

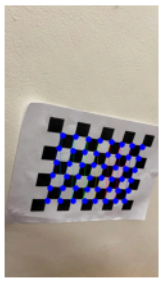
25.jpeg

Re-projected Corners Detected (given dataset):

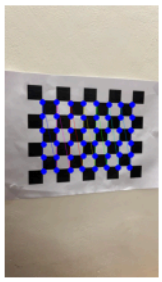
1.jpeg



2.jpeg



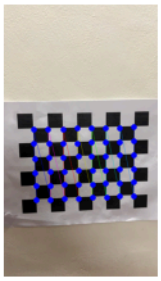
3.jpeg



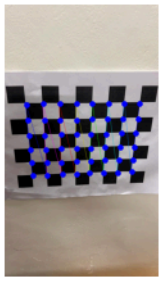
4.jpeg

5.jpeg

6.jpeg



7.jpeg



8.jpeg

9.jpeg

10.jpeg

11.jpeg

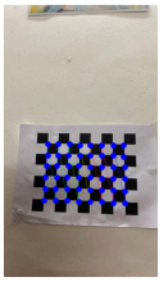
12.jpeg

13.jpeg

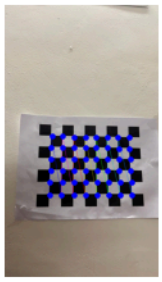
14.jpeg

15.jpeg

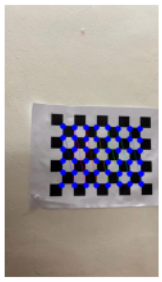
16.jpeg



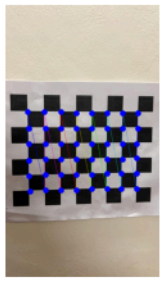
17.jpeg



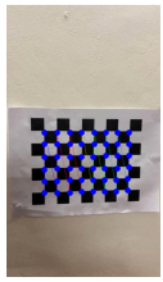
18.jpeg



19.jpeg

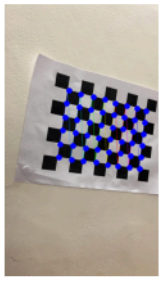


20.jpeg

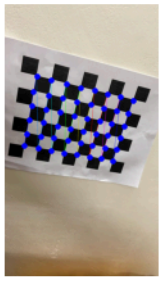


21.jpeg

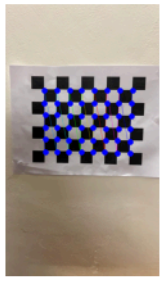
22.jpeg



23.jpeg



24.jpeg

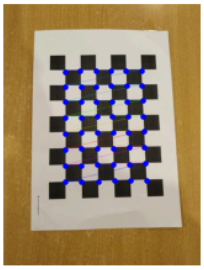


25.jpeg

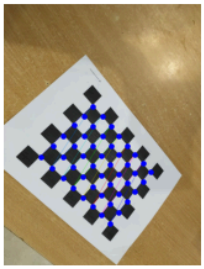
Original Corners Detected (custom dataset):

Re-projected Corners Detected (custom dataset):

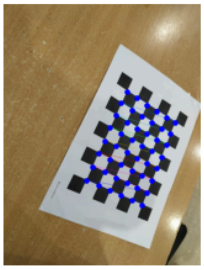
1.jpg



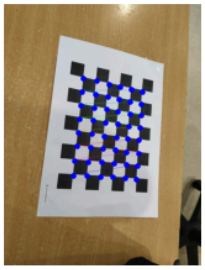
2.jpg



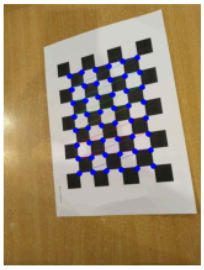
3.jpg



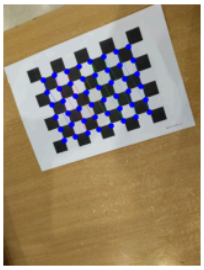
4.jpg



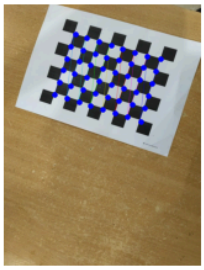
5.jpg



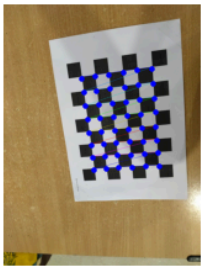
6.jpg



7.jpg

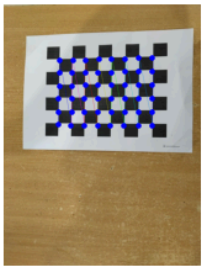


8.jpg

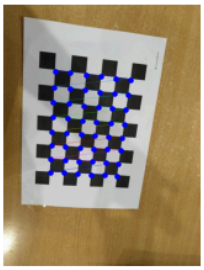


9.jpg

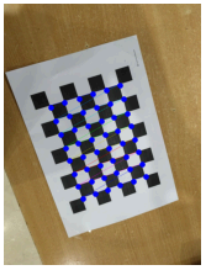
10.jpg



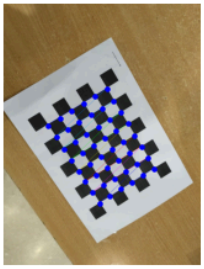
11.jpg



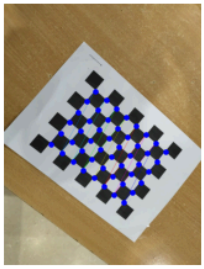
12.jpg



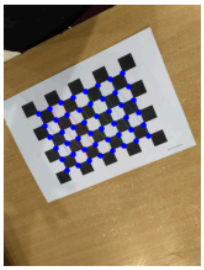
13.jpg



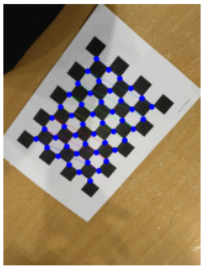
14.jpg



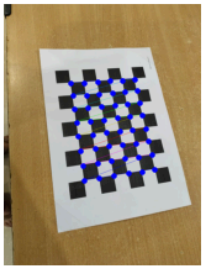
15.jpg



16.jpg

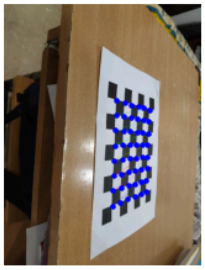


17.jpg

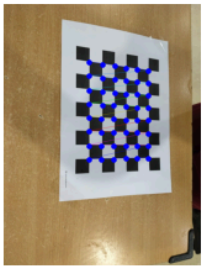


18.jpg

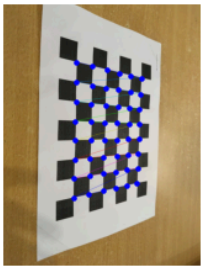
19.jpg



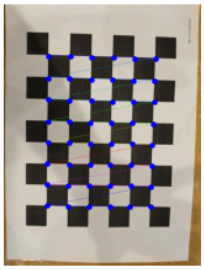
20.jpg



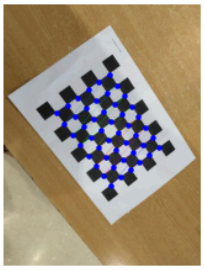
21.jpg



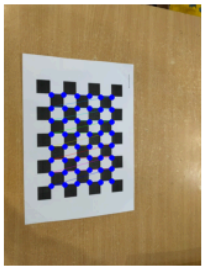
22.jpg



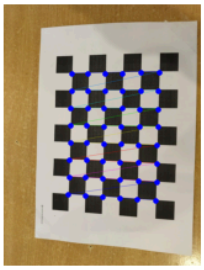
23.jpg



24.jpg



25.jpg



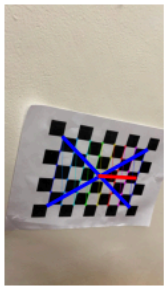
1.5 Plane Normals

Given Dataset:

```
Normals for Images:
1.jpeg: [-0.07614282 -0.14871274  0.98594462]
2.jpeg: [-0.38524235 -0.30921139  0.86946918]
3.jpeg: [-0.27137069 -0.09614749  0.95766049]
4.jpeg: [-0.16579513  0.16015865  0.97306792]
5.jpeg: [-0.39733691  0.19080926  0.8976164 ]
6.jpeg: [-0.08323149 -0.12754391  0.98833449]
7.jpeg: [-0.1031108  -0.40167836  0.90995751]
8.jpeg: [ 0.18878635 -0.06637134  0.97977271]
9.jpeg: [0.45457849 0.40009209 0.79579188]
10.jpeg: [-0.0228767   0.12387258  0.9920344 ]
11.jpeg: [-0.39823935 -0.18715614  0.89798552]
12.jpeg: [-0.3464131  -0.52809983  0.77531189]
13.jpeg: [0.35821827 0.39862471 0.84426182]
14.jpeg: [ 0.04524897 -0.32517786  0.94456968]
15.jpeg: [-0.12777695 -0.50727314  0.85225994]
16.jpeg: [0.0720445  0.21725405 0.97345276]
17.jpeg: [-0.095371    0.19722026  0.97570925]
18.jpeg: [-0.14605831  0.07386035  0.98651489]
19.jpeg: [-0.10768966 -0.08641984  0.9904214 ]
20.jpeg: [-0.13548788 -0.10599064  0.98509341]
21.jpeg: [-0.09760803 -0.27445631  0.95663285]
22.jpeg: [-0.36032229  0.01814003  0.93265148]
23.jpeg: [-0.16457517 -0.36281153  0.9172147 ]
24.jpeg: [ 0.03573087 -0.11054463  0.99322867]
25.jpeg: [-0.22319276  0.06828544  0.9723796 ]
```

1.jpeg

2.jpeg

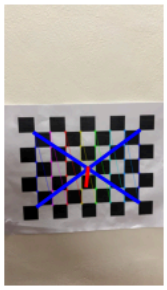


3.jpeg

4.jpeg

5.jpeg

6.jpeg



7.jpeg

8.jpeg

9.jpeg

10.jpeg

11.jpeg

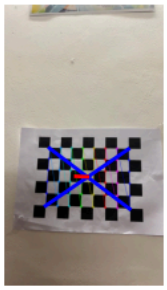
12.jpeg

13.jpeg

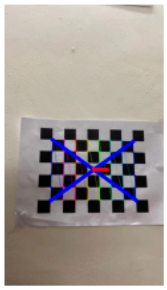
14.jpeg

15.jpeg

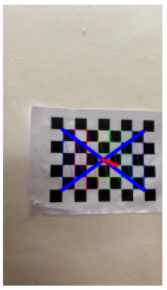
16.jpeg



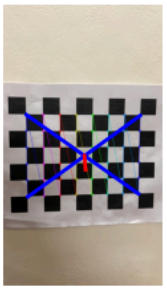
17.jpeg



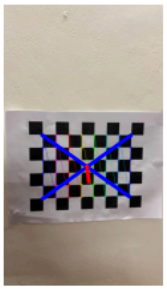
18.jpeg



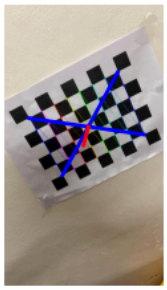
19.jpeg



20.jpeg

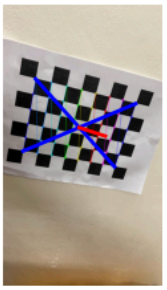


21.jpeg



22.jpeg

23.jpeg



24.jpeg

25.jpeg

Custom Dataset:

```
Normals for Images:
1.jpg: [-0.03493717  0.10265555  0.99410323]
2.jpg: [-0.38006598  0.01262774  0.92487318]
3.jpg: [-0.3689578  -0.01635891  0.92930217]
4.jpg: [-0.33909765  0.08416223  0.93697892]
5.jpg: [0.34281497 0.07719612 0.93622575]
6.jpg: [ 0.11523487 -0.12266638  0.9857352 ]
7.jpg: [0.06208322 0.06961607 0.99564013]
8.jpg: [ 0.16354056 -0.22216268  0.96119625]
9.jpg: [0.10505494 0.40885    0.90653469]
10.jpg: [0.09562612 0.07665679 0.99246128]
11.jpg: [-0.11246314 -0.25559996  0.96021909]
12.jpg: [-0.09936324 -0.1536228   0.98312104]
13.jpg: [-0.06733423 -0.12753634  0.98954564]
14.jpg: [-0.10807829 -0.05883326  0.99239999]
15.jpg: [ 0.04936391 -0.23613063  0.97046666]
16.jpg: [ 0.0139471  -0.1967244   0.98035962]
17.jpg: [-0.1761637   0.26303843  0.94856583]
18.jpg: [-0.18059222 -0.3431295   0.92176385]
19.jpg: [-0.77980811 -0.03503633  0.62503742]
20.jpg: [-0.34647562  0.03478541  0.93741379]
21.jpg: [-0.61709695 -0.05920324  0.78465682]
22.jpg: [-0.02746477 -0.05666531  0.9980154 ]
23.jpg: [-0.14654477 -0.09023985  0.98507939]
24.jpg: [-0.3140873  -0.03012249  0.94891612]
25.jpg: [0.15536374 0.00128565 0.9878565 ]
```


1.jpg
2.jpg
3.jpg
4.jpg
5.jpg
6.jpg
7.jpg
8.jpg
9.jpg
10.jpg
11.jpg
12.jpg
13.jpg
14.jpg
15.jpg
16.jpg
17.jpg
18.jpg
19.jpg
20.jpg
21.jpg
22.jpg
23.jpg
24.jpg
25.jpg

2. Panorama Generation

2.1 Keypoint Detection

Web Resource FLANN Feature Matching Example with OpenCV

Image 1:

```
Number of keypoints detected: 5890
Descriptor shape: (5890, 128)
```

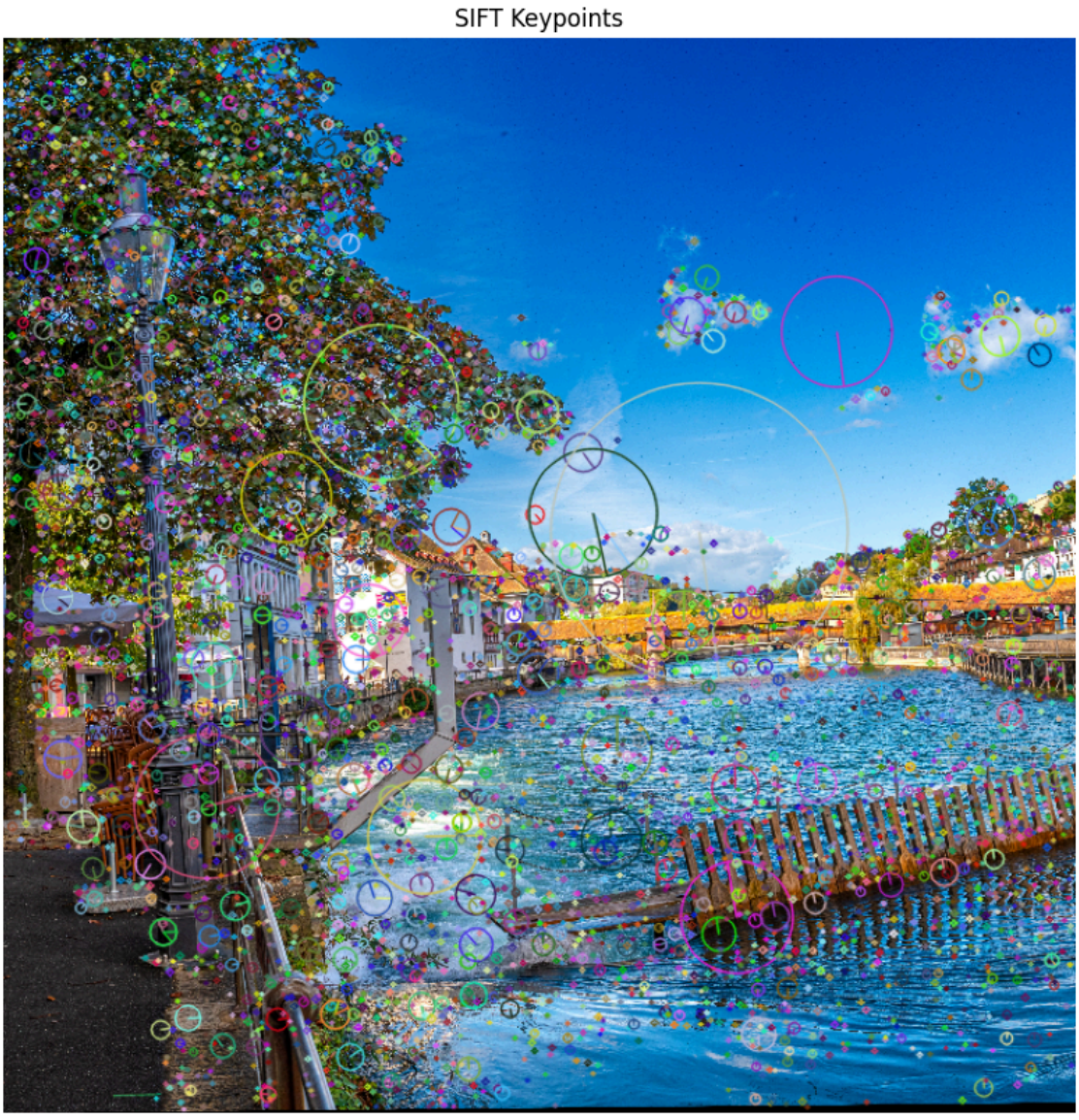


Image 2:

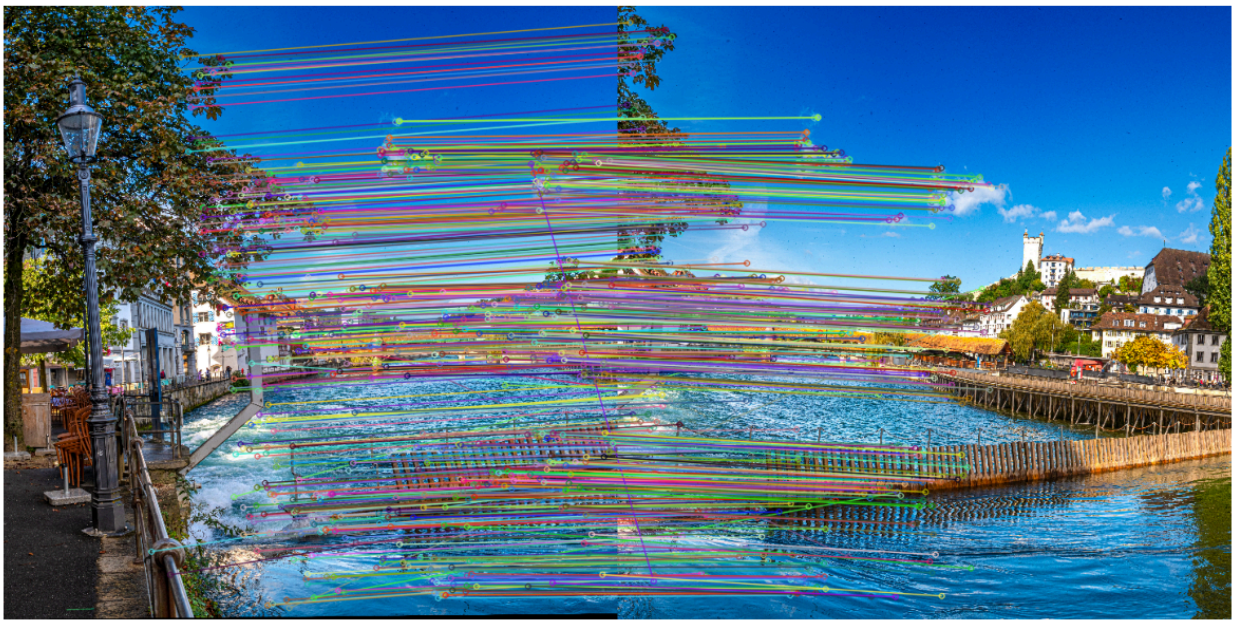
```
Number of keypoints detected: 4510
Descriptor shape: (4510, 128)
```

SIFT Keypoints

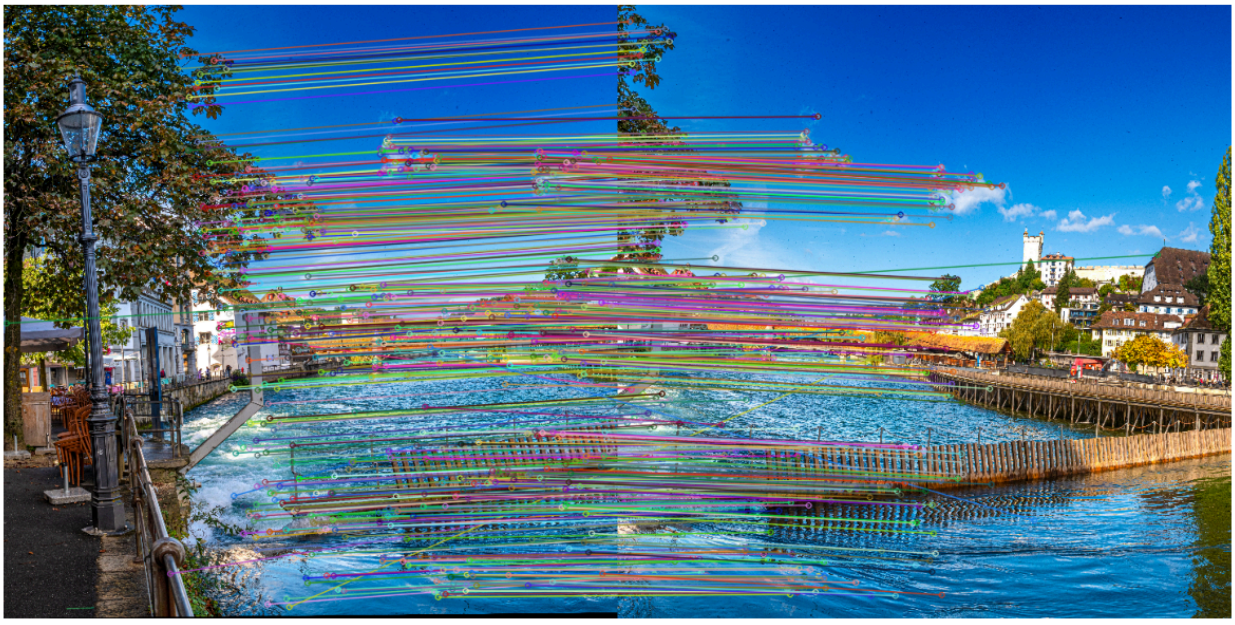
2.2 Feature Matching

[Web Resource]

BruteForce Matches



FLANN Matches



2.3 Homography

[Web Resource]

Homography Matrix

```
0.8988766002460821,-0.13159425784673895,-164.35939536724635
0.14310529360507335,0.9900953870417383,-69.31099221049534
-2.687354027631992e-07,-3.019992918103179e-07,1.0
```

Warped Image1 and Original Image2

2.4 Stitching

Image Stitching

Crop Function Based on Largest Contour Detection:

```
def crop(image):
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    _, thresh = cv2.threshold(gray, 1, 255, cv2.THRESH_BINARY)
    contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

    if not contours:
        return image

    largest_contour = max(contours, key=cv2.contourArea)
    x, y, w, h = cv2.boundingRect(largest_contour)
    return image[y:y+h, x:x+w]
```

Converts the input image to grayscale and apply binary thresholding to isolate the foreground. Detect external contours and identify the largest one based on area. Crop the image to the bounding rectangle of this contour to remove unnecessary background regions. This function is useful in cropping the generated panorama.

2.5 Panorama Generation

Panorma for Group-0

Panorma for Group-1

Panorma for Group-2