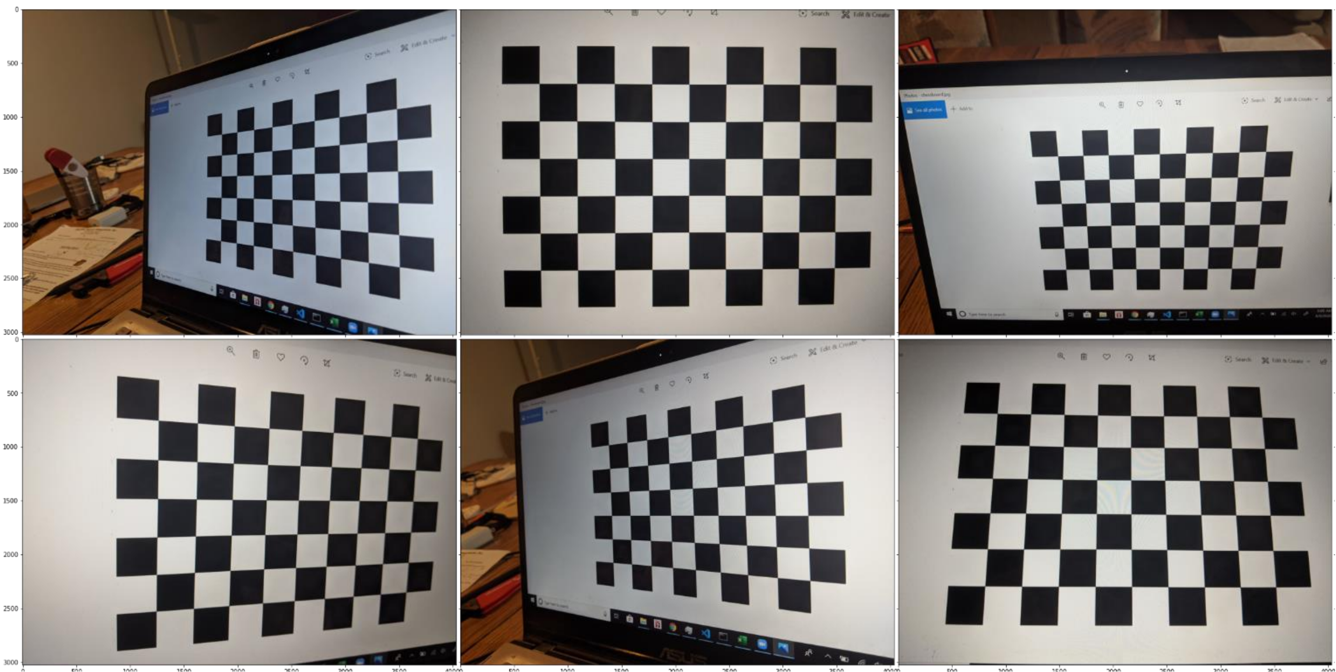
PROJECT 2: REPORT

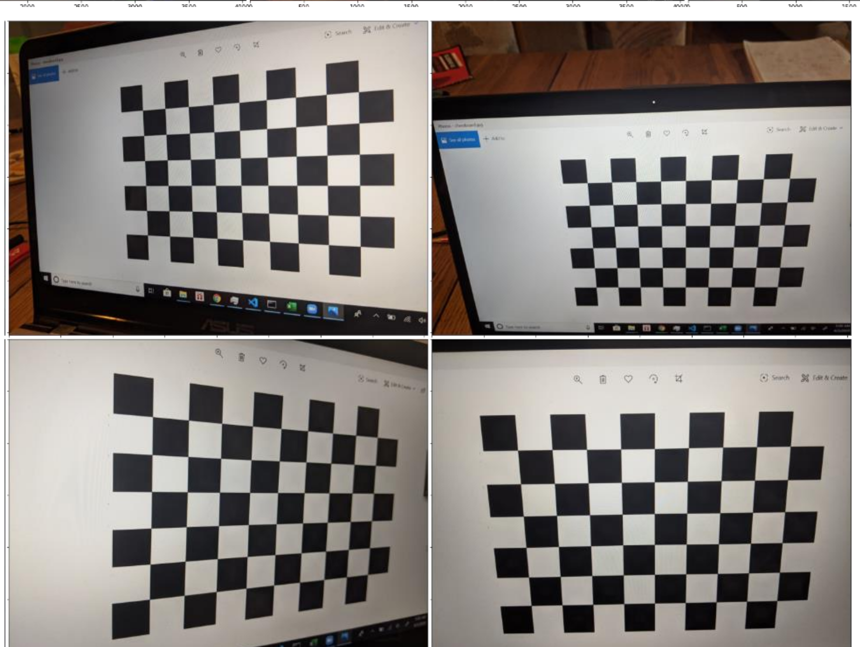
VHAL DHIMANT PUROHIT

Link to code:
https://colab.research.google.com/drive/1RS1k7eJ8E4bI8ozi9SGhKLuCS4RwVjBc#scrollTo=6QaBkt8NwflW&uniqifier=1

Part 1: Camera Calibration

We need to calculate the parameters of the camera before we can apply any transformations. We do that by first finding the corners in pixel coordinates, and since we already know the world coordinates, using the two of this we can calculate the parameters of the camera (radial distortion, intrinsic matrix K, rotation and translation vectors). These were the pictures used to calibrate the camera:





2) The intrinsic matrix is

```
[3240.60433542    0.              2028.73169396]
[   0.            3244.3704965    1504.56690725]
[   0.               0.              1.         ]
```

3) The distortion coefficients are

[[0.23966791, -1.35215859, 0.00088688, 0.00055514, 2.19938318]]

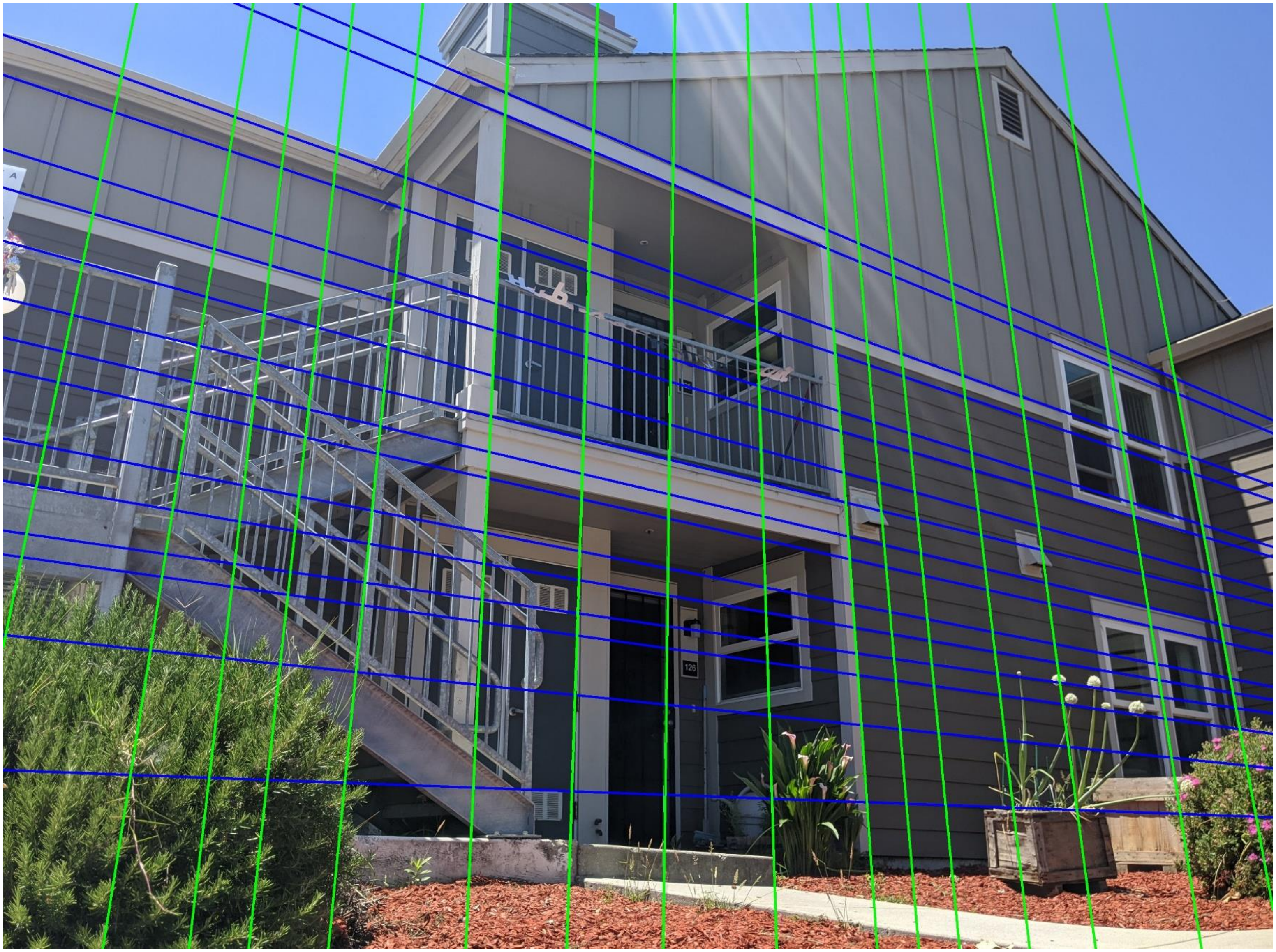
4) The reprojection mean square error is 0.049780902547380025

Part 2: Taking the picture

The original picture | The undistorted picture | Undistorted, cropped

Part 3: Find bundle of parallel lines

To find the bundle of parallel lines, I used OpenCV and Mouse Events. I opened my image, and dragged the mouse from one point to another, storing the x,y coordinates of these two points. Using these points, I calculated the line that passes through these points, and using the slope and intercept of this line, I found where this line intersected with the edges of the image and drew a line using the points of intersecting and cv2.line. (In case the two points had the same x coordinate, the slope is actually infinity hence for this edge case, I used the coordinates xi,0 and xi,max_y). Using this procedure, I drew 16 “vertical” and 16 “horizontal” lines. This is the picture that I obtained



The “horizontal” lines are drawn in blue and the “vertical” lines are drawn in green.

Part 3: Vanishing Directions

We use Median of Least Squares to calculate the normal to the vertical, horizontal and transversal planes. We calculate the lever vector for each line in a particular bundle (horizontal/vertical), ideally the cross product between any two levers in a bundle should give the same plane, but due to inaccuracies in the calculated lines and the distortions, there may be discrepancies, hence we use least median of squares, as mentioned in the problem statement. We get the following values:

1)VECTOR (m^c_V) is: [[0.01407251 0.93772932   -0.34708167]]
VECTOR (m^c_H) is: [[0.82236944   0.18296865   0.5387309]]

Vertical least median of residuals is: [0.00460766]
Horizontal least median of residuals is: [0.00502419]

Transversal plane (m^c_T) is: [[-0.56869301 0.29301281 0.76859076]]
(Calculated using the cross product between m^c_H and m^c_V and normalizing)

2)The angle between m^c_V and m^c_H is 90.21977284 degrees

3) The vanishing points of the three vectors are

Horizontal vanishing point: $[6975.49451514, 2606.44931988]^T$

Vertical vanishing point: $[1897.34064056, -7260.92427975]^T$

Transversal vanishing point: $[-369.04501167, 2741.43074071]^T$

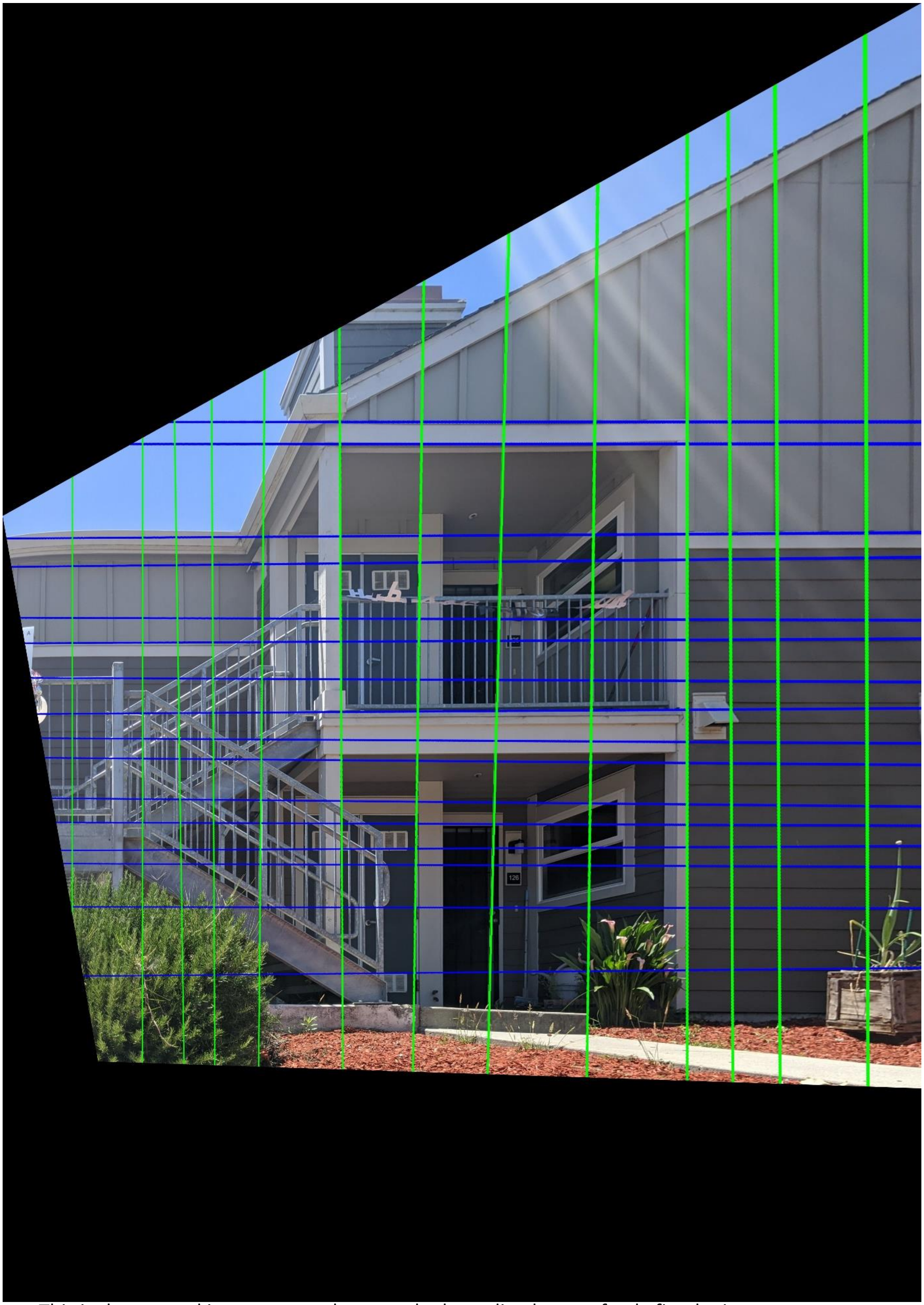
Part 4: Rectifying homography

1) The homography matrix H:

$$\begin{bmatrix} 0.47173083 & 0.37378814 & 1819.0132844 \\ -0.24960231 & 1.08657588 & -1091.00545231 \\ -0.00017749 & 0.00009134 & 1. \end{bmatrix}$$

Using this homography, plane normal vectors m^c_V and m^c_H are mapped to infinity. (Can be seen in the code)

2) The dimension of the new rectangular frame after warping is: (6333, 4482)



This is the warped image, as can be seen the bounding box perfectly fits the image