Computer Vision and Pattern Recognition
Project 2

Erifeoluwa Jamgbadi and Krunal Rathod

June 19th 2023

1 Description of Solution

We manually read the points off the image by selecting the points we wanted using mouseclick event. The red circles were put on the selected points and the coordinates of the selected points were put in a list.

1.1 Task 1

- The code initializes an empty list intersection_points to store the closest intersection point for each selected point.
- It iterates over the selected points and calculates the Euclidean distance between each selected point and the known world coordinates.
- The closest intersection point is determined by finding the world coordinate with the minimum distance to the selected point.
- The closest intersection points are stored in the intersection_points list.

1.2 Task 2

- The code initializes an empty matrix A to construct the linear system of equations.
- It iterates over the intersection points and constructs the matrix A using the Direct Linear Transform (DLT) algorithm.
- Singular Value Decomposition (SVD) is then used to decompose A and obtain the projection matrix P.

1.3 Task 3

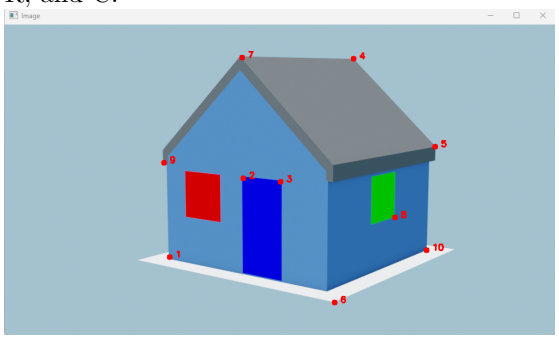
- The code applies the QR decomposition to extract the camera calibration matrix K and camera orientation R from the projection matrix P.
- The camera calibration matrix K is normalized by dividing by the value in the bottom-right corner.
- The translation vector T is computed by inverting K and multiplying it with the last column of P.
- The camera center C is then obtained by multiplying the transpose of R with T.
- Finally, the code prints the camera calibration matrix K, camera orientation R, and camera center C.

The code essentially allows to select points of interest in the image, finds the corresponding 3D points, and performs camera calibration and reconstruction to determine the camera parameters.

2 Results

2.1 House 1

The below image shows the points we clicked and our estimated numbers for K, R, and C.



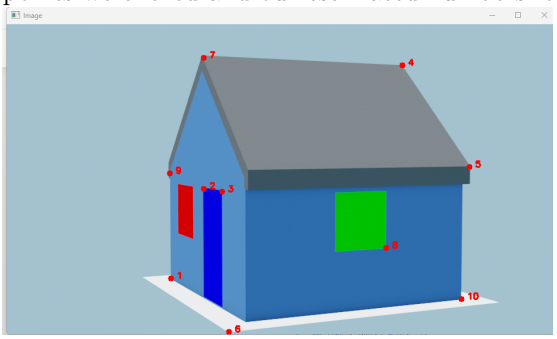

```
Camera Calibration Matrix K:
[[-5.10637061e-02 -9.98699624e-01 -7.03405941e-02]
 [ 1.00117345e+00 -5.09855571e-02 -2.90543337e-03]
 [-6.83008681e-04 -7.03972600e-02  1.00000000e+00]]

Camera Orientation R:
[[ 0.69511842 -0.71691698 -0.0013128 ]
 [ 0.          0.03732813 -0.03794965]
 [ 0.          0.         -0.0015809 ]]

Camera Center C:
[ 8.18642255e-05 -1.25018560e-04  4.29972450e-05]
```

2.2 House 2

We applied the same method to the house 2 image. The below image shows the points we clicked and our estimated numbers for K, R, and C.




```
Camera Calibration Matrix K:
[[-2.43208115e-01  9.69976546e-01  1.26604015e-02]
 [-9.70058672e-01 -2.43200451e-01 -2.16484049e-03]
 [ 9.79090089e-04 -1.28067827e-02  1.00000000e+00]]

Camera Orientation R:
[[-0.69386955  0.67660382  0.04241064]
 [ 0.          0.17020222 -0.1731553 ]
 [ 0.          0.         -0.00122002]]

Camera Center C:
[ 1.43650876e-04 -2.32256223e-04  8.56435908e-05]
```