

ADIPCV-2019 ASSIGNMENT-2
/* Calibration of Camera and homography */

Given an image you are asked to estimate the projection matrices of a camera (in the imaging system). For estimating the projection matrix develop an interactive interface, which can perform the following operations on an image.

- (a) Compute the pixel coordinate of a point in an image.
- (b) Compute the length in pixels between two image points.
- (c) Compute the vanishing point of two parallel lines (in real world) as identified by the user.
- (d) Prepare a data file with saved correspondences of image pixel coordinates and real world coordinates.

Using the above tool and also other associated functions as you need, form the projection matrices (they are different!) of the following two images with the information related to the world coordinates specified with it.

I. Name of the image: "Helipad.jpg"

Information of world coordinates: Consider the height of the vertical pillars is of 2 meters, separated by a distance of 4 meters, and width of the road including thickness of both the white borders is 1.5 meters.

II. Name of the image: "Palace.jpg"

Assume it is a case of affine projection, implying images of parallel lines are also parallel in the image.

Information of world coordinates: Use a plausible world coordinate system by assuming the walls parallel of XZ and YZ planes.

- (e) Find intrinsic and extrinsic parameters for both the camera matrices.
- (f) Consider the pentagonal artwork attached at the left wall (in the bottom part of the image) showing sketches of sword and shield and having a red border. Replace all the small windows shown at the topmost part of the walls by this artwork.

Marking Policy:

- (i) Interactive tool on various measurement: 20
- (ii) Solving the projection matrix for "Helipad.jpg": 15
- (iii) Solving the projection matrix of "Palace.jpg": 15
- (iv) Intrinsic and extrinsic parameters: 20
- (iv) Replacement of windows by pentagonal artwork in "Palace.jpg": 20.
- (v) Report: 10

Bonus: Quality of solution (10)

You may implement your programs in C++-OpenCV/MATLAB/ Python with necessary user's interfaces and visualization of your results and input.

Please provide a documentation for compiling and running the programs in a README file.

The whole project should be submitted in a single tar or zip file.