

ECE 472 Robotics and Vision

Prof. K. Dana

Homework 2: Homography Estimation, Image Formation Pipeline

1. Write a python notebook to “image” a simple object described by vertices (e.g., a simple house) in world coordinates from the view indicated by the extrinsic matrix below. . Use command *Line2D* to draw the 2D lines between the projected vertices. Do not use any 3d plotting commands. The projections should be handled by using the camera matrix (image formation pipeline) discussed in class.

The calibration parameters are given as follows:

$${}^cR_w = \begin{bmatrix} 0.707 & 0.707 & 0 \\ -0.707 & 0.707 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad {}^ct_w = \begin{bmatrix} -3 \\ -0.5 \\ 3 \end{bmatrix}$$

Therefore

$${}^cT_w = \begin{bmatrix} 0.707 & 0.707 & 0 & -3 \\ -0.707 & 0.707 & 0 & -0.5 \\ 0 & 0 & 1 & 3 \end{bmatrix}.$$

Also

$$K = \begin{bmatrix} -100 & 0 & 200 \\ -0 & -100 & 200 \\ 0 & 0 & 1 \end{bmatrix}$$

2. Where is the camera in the previous question (w.r.t world coordinates)?
3. Show the simple object from another camera view by changing the position and/or orientation of the camera. You may do this in one of the following two ways: (1) modify cT_w directly, or (2) choose a “target” point to look at in world coordinates and a new position of the camera in world coordinates and follow the method in class.
4. **[EXTRA CREDIT]** Image a calibration cube using the camera matrix in the first question. A calibration cube has multiple detectable corners on all 4 sides of the cubes (often a checkerboard but the black/white pattern is irrelevant here). Use *Line2D* to draw the image of 3 faces of the calibration cube. Set up and solve the system of equations for the DLT method of calibration. Show that you can recover the correct camera matrix M in this ideal no-noise scenario.

Coordinate Frame Transformations

5. Coordinate frame B is described as follows: Start with B coincident with a known frame A . Rotate B about \hat{z}_A , by -45° . Translate B by the vector $[1, 1, 0]$ (written with respect to A).
 - (a) Given ${}^AP = [1, 0, 2]$ what is BP ? That is, given a point that is $[1, 0, 2]$ when written with respect to the A frame, what is that point written with respect to the B frame?

6. Coordinate frame B is described as follows: Start with B coincident with a known frame A . Rotate B about \hat{y}_A , by 90° . Then Rotate B about \hat{z}_A by 90° Translate B by the vector $[1, 1, 0]$ (written with respect to A).
- What is \hat{y}_B written with respect to the A frame?
 - What is \hat{y}_A written with respect to the B frame?
 - Given ${}^A P = [1, 0, 10]$ what is ${}^B P$? That is, given a point that is $[1, 0, 10]$ when written with respect to the A frame, what is that point written with respect to the B frame?
7. Consider Figure 2.
- What is the the rotation matrix ${}^A R_B$?
 - What is the translation vector ${}^A t_B$?

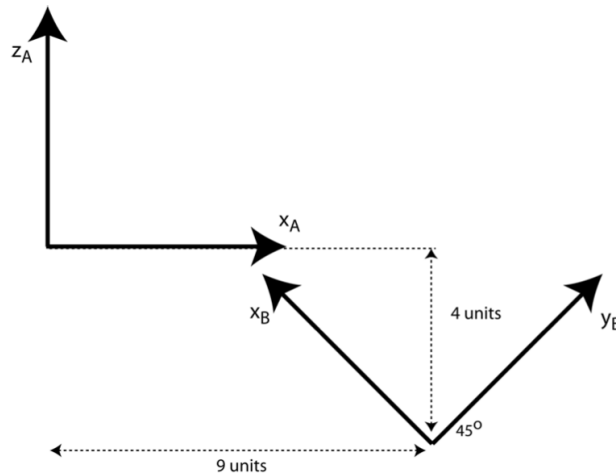


Figure 1: Coordinate frame A and B.

8. Write a python program to estimate a homography between a frontal view and side view image of a planar surface (e.g. side of a building, sign, photo). The program input should also be corresponding point pairs (Matlab's `ginput` is useful here). The program should estimate the homography using the DLT method. Report the homography parameters and warp the side view image to match a frontal view of the window. An example image is provided in *hpworld.png* (you may use a different image). You may use an external warping function only if it takes the homography as input (e.g. `skimage.transform.warp`).

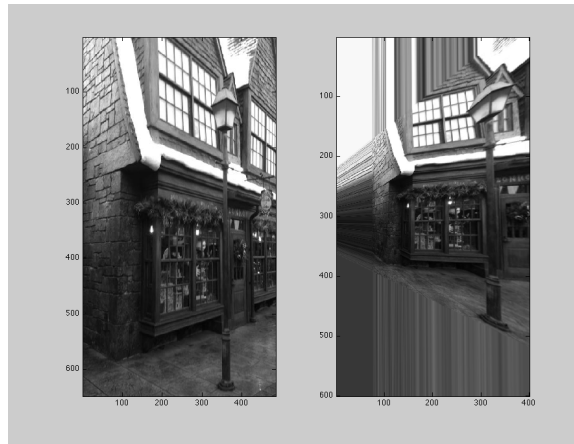


Figure 2: Remap the window region (left) to a frontal view (right) by estimating the appropriate homography and warping the image.