CS364/AM792 : Computer Vision 21 August 2017

Assignment 3: Camera Calibration and Epipolar Geometry

Please prepare a single, condensed and neatly edited document for submission. For each problem include a short description of what you did, results, and a brief discussion/interpretation. It is not necessary to include any code in your document (unless a snippet helps to explain your method). Hand in a printed copy **before 14:00** on the due date, and also e-mail me a zip-file containing your document and code.

You are free to use any programming language. I recommend Matlab or Python. If you are asked to implement a specific technique, the idea is that you do so from scratch; do not simply use a function from an image processing or computer vision library. Collaboration is restricted to the exchange of a few ideas. All code, results, and write-up that you submit must be your own work.

---

1. Consider the image `lego1.jpg`. The object in this image was built with Lego bricks, each being 32mm long, 16mm wide and 9.6mm high.

(a) Identify a number of corners on the object. Determine their 2D image coordinates in pixels as well as their 3D world coordinates in mm. Choose the axes of your world coordinate system to align with the edges of the object, and pick at least 28 correspondences (per the rule-of-thumb). Then use these correspondences to determine the camera matrix $\mathbf{P}$, as explained in section 5.2.1.

(b) Suppose $\mathbf{P} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \mathbf{p}_4 \end{bmatrix}$. From section 5.3.2 we know that $\mathbf{p}_4$ is a homogeneous representation of the image of the world origin. Furthermore $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$ are the images of the vanishing points of the world X-, Y- and Z-axes, respectively. Demonstrate these facts by de-homogenizing the four columns of your computed $\mathbf{P}$ and using the image coordinates thus obtained to draw* the projected world coordinate axes on top of the image.

*We discussed in class how you might have to flip some of your axes to have them point in the positive direction.

(c) The algorithm in section 5.2.2 of the notes provides a means of extracting the constituents of a given camera matrix. Prove the theoretical correctness of this algorithm. That is to say, if the algorithm returns $\mathbf{K}$, $\mathbf{R}$ and $\mathbf{c}$ from a given $\mathbf{P}$, prove that $\mathbf{K}$ is upper-triangular, $\mathbf{R}$ is orthogonal, and $\mathbf{K}\,\mathbf{R}\,[\,\mathbf{I} \mid -\mathbf{c}\,] = \mathbf{P}$.

(d) Decompose the camera matrix you obtained in part (a) into $\mathbf{K}$, $\mathbf{R}$ and $\mathbf{c}$. Show the three elements in your report (first scale the homogeneous matrix $\mathbf{K}$ such that its bottom-right entry is 1). There is a document on the website showing exactly how the algorithm in section 5.2.2 can be implemented.

---

2. (a) Repeat problems 1a, 1b and 1d for the image `lego2.jpg`. When selecting point correspondences it is advisable to use the same corners you identified in the first image, so that their world coordinates remain unchanged, and find new image coordinates for them.

(b) How far apart (in mm) were the centres of the two cameras when they captured these two images? What was the angle (in degrees) between their respective principle axes?

(c) Create a 3D plot of the points you selected in the world coordinate system. Then, in the same picture, draw the three coordinate axes of each of the two cameras. Indicate which camera axes are which (e.g. by labelling them with x_1, y_1, z_1, x_2, y_2 and z_2 respectively), and show a few different views of this 3D plot to clarify how each camera is positioned and orientated relative to the calibration object.

Note: for this 3D plot to be visually sensible, it is imperative to enforce equal unit lengths along the three coordinate axes. Achieve this with `axis('equal')` in Matlab, or `ax.axis('equal')` if `ax` is an Axes3D object from mplot3d in Python.

---

p.t.o.

3. Suppose the camera that captured `lego1.jpg` is "camera 1", and the camera that captured `lego2.jpg` is "camera 2". In keeping with the notation of the notes, let $\mathbf{P}$ be the camera matrix of camera 1 (as computed in problem 1a) and $\mathbf{P}'$ the camera matrix of camera 2 (as computed in problem 2a).

(a) Calculate the epipole $\mathbf{e}$ (which is the projection of the second camera centre onto the first image plane) as well as the epipole $\mathbf{e}'$ (the projection of the first camera centre onto the second image plane). Write down the de-homogenized image coordinates of both. Looking at the images, do these coordinates make sense?

(b) Calculate the fundamental matrix $\mathbf{F}$ from $\mathbf{P}$ and $\mathbf{P}'$, by means of equation (6.7) in the notes.

(c) Use $\mathbf{F}$ to determine and draw a number of corresponding epipolar lines across the two images. You should display these lines in a way that makes it clear which correspond with which (draw each corresponding pair in their own colour, for example).

Hand in: 7 September 2017