Exercise in Estimating View Geometry

Henrik Aanæs, haa@imm.dtu.dk

October 20, 2010

This is the third exercise in view geometry, related to Chapter 2 of my lecture notes. Firstly, there will be questions relation to point trinagulation, followed by questions on estimating the homography between two images.

1 Camera Model

1.1 Setting up the Camera

Here three cameras are given, where it is assumed that the images have been corrected for inner orientation. Thus, the inner orientation can be discarded from the camera model. The rotations of all three cameras is given by the identity matrix and their respective translations are given by

$$\mathbf{t_1} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad , \quad \mathbf{t_2} = \begin{bmatrix} -5 \\ 0 \\ 2 \end{bmatrix} \quad , \quad \mathbf{t_3} = \begin{bmatrix} 0.1 \\ 0 \\ 0.1 \end{bmatrix}$$

1. Implement the camera matrices for all three cameras.
2. Project the 3D point
$$Q_1 = \begin{bmatrix} 2 \\ 4 \\ 10 \\ 1 \end{bmatrix}$$
To all three cameras, and verify that it projects to
$$\mathbf{q}_{11} = \begin{bmatrix} 0.2 \\ 0.4 \\ 1 \end{bmatrix} \quad , \quad \mathbf{q}_{12} = \begin{bmatrix} -0.25 \\ \frac{1}{3} \\ 1 \end{bmatrix} \quad , \quad \mathbf{q}_{13} = \begin{bmatrix} 0.2079 \\ 0.3960 \\ 1 \end{bmatrix}$$
In the three cameras respectively.
3. Give a verbal/intuitive explanation of the 3 cameras location and viewing direction.

1.2 3D Solver

Implement a function `Q=Est3D(q1,P1,q2,P2)`, which given the projection of a 3D point in two images, `q1,q2`, and the respective camera matrices `P1,P2` estimates the position of the 3D point. Test that the projections into camera 1 and 2 corresponds to Q_1, thus validating your function.

1. Another 3D point, Q_2 projects to $\mathbf{q}_{21} = [-0.1667, 0.3333, 1]^T$ and $\mathbf{q}_{22} = [-0.5000, 0.2857, 1]^T$ in cameras one and two respectively. Use `Est3D` to estimate Q_2.
2. Calculate $\mathbf{Q}_{23}$ the projection of Q_2 into camera 3.

1.3 Accuracy

The last part of this exercise deals with measurement accuracy, since noise will be present in *all* real data. Here, the uncertainty is simulated by permuting the observations, but it should be noted that with real data the permutation is not know.

1. Permute the observation of Q_2 in cameras 2 and 3, $\mathbf{q}_{22}$ and $\mathbf{q}_{23}$ respectively, by adding 0.1 to all the coordinates. Denote these permutations by $\bar{\mathbf{q}}_{22}$ and $\bar{\mathbf{q}}_{23}$.
2. Re-estimate the 3D position of of Q_2 based on $\mathbf{q}_{21}$ and $\bar{\mathbf{q}}_{22}$. Naturally using cameras 1 and 2.
3. Re-estimate the 3D position of of Q_2 based on $\mathbf{q}_{21}$ and $\bar{\mathbf{q}}_{23}$.
4. Compare the estimates from question 2. and 3. with the 'ground truth' based on $\mathbf{q}_{21}$ and $\mathbf{q}_{22}$ (no $\bar{\ }$). Explain !

1.4 Real Images

From the two real images in the previous exercise, "Exercise in the Camera Model", in the file `TwoImageData.mat`, set up a script where you can annotate two points of the same physical entity, one in each of the images, and estimate the 3D coordinate of the annotated 3D entity. The latter should be achieved via `Q=Est3D(q1,P1,q2,P2)`. Try it out, and try to convince your self that the code is working properly, or fix the errors in the program.

2 3D Inference from a Plane

Lastly, consider the image of a badminton match in Figure 1 and the measurements of a badminton court in Figure 2. Find the homography that maps from Figure 1 to Figure 2. This homography is estimated by annotating image points in Figure 1 and associating them with the appropriate coordinates calculated from Figure 2.

Figure 1 is located in the file `petergade.png`. For annotating the images the function

```
ginput(1)
```

is recommended. Remember to normalize the coordinates as part of the homograph estimation, e.g. using the code:

```
Mean1=mean(X1')';
X1(1,:)=X1(1,:)-Mean1(1);
X1(2,:)=X1(2,:)-Mean1(2);
S1=mean(sqrt(diag(X1'*X1)))/sqrt(2);
X1(1:2,:)=X1(1:2,:)/S1;
T1=[eye(2)/S1,-Mean1(1:2)/S1;0 0 1];
```

Warp Figure 1, according to the estimated homography, here the functions should be of help (try looking the used function up in the MatLab help)

```
figure
%Note That H has to be transposed here if column vectors are used for poi
Tr=maketform('projective',H');
% XData and YData need to be specified to get the output coordinate syste
WarpIm=imtransform(Im,Tr,'YData',[0 q(2,3)],'XData',[0 q(1,6)]);
imagesc(WarpIm)
```

What is the position of the two players on the court?

Figure 1:

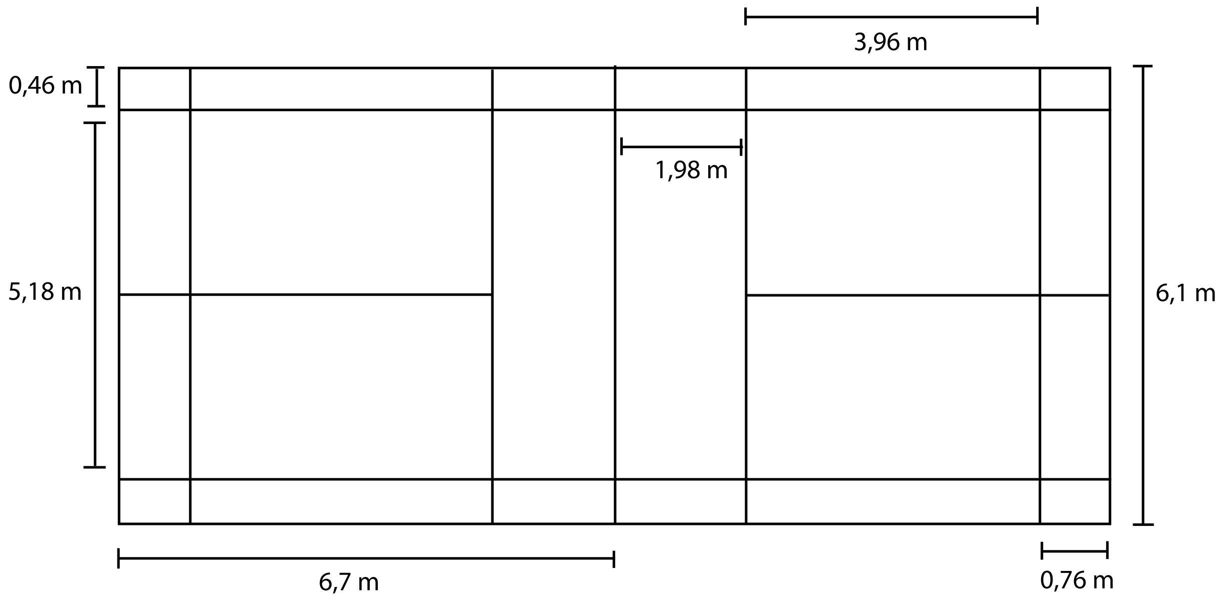

Figure 2: