Computer Vision Fundamentals - Assignment 2

Due: Tuesday 26 June, 2018 at 5:00 pm　　　　　　　　　　Grade Scale: 100 points

This assignment contains several problems in which you have to use MATLAB. Your deliverable should be in the form of a document in which you describe what you did in each problem and why. Include all your relevant code segments as well as your explanations and results figures. I highly encourage you to write the report as you are working on the solution, and not leave the report till the end, because that would just increase the amount of effort required.

Zip all the scripts for each problem, named by the problem number, and upload them along with your report.

There are some bonus problems in the assignment that can earn CS436 students up to 20% additional points. I would recommend that you attempt them in the end, only after you have written down the solution to each of the required problems. For CS5310 students, bonus problems are required.

Problem 1. [15 points] Consider the GIF image provided with this assignment, which shows a sequence of frames taken by a camera. A few frames from the sequence are shown below.

a. [5 points] Describe in words and through means of a diagram how this set of images were taken.
b. [10 points] Mathematically describe the relationship between the camera extrinsic and intrinsic parameters if such an image sequence is to be captured.
c. [Bonus: 5 points]. Using your phone camera, try to capture a similar sequence as accurately as possible. Describe the setup you used and what steps you took to ensure accuracy in your setup. Convert your sequence to a GIF and submit it with your assignment.

Problem 2. [35 points] Given

- two images of a plane
- taken by two cameras with
  - non-zero translation between them
  - and arbitrary orientations

a. [5 points] Algebraically prove that the images are related by a 2D homography (i.e. a 2D projective transformation).
b. [30 points] Use MATLAB to simulate
   i. [5 points] A set of at least a 100 random points on a randomly-oriented 3D plane. Show that your generated points are indeed on a plane.
   ii. [5 points] Set up two arbitrary camera matrices.
   iii. [5 points] Through a script, compute the image coordinates of the 3D points on the plane using these cameras.
   iv. Compute a homography between the image points. This you will do in two different ways.
      A. [5 points] Derive the Least Squared Error solution for computing homography between two sets of image points. Code your solution in a script and use it to compute homography.
      B. [5 points] Use the `homography2d` function from [Peter Kovesi's site]() for computing homography. Peter's function uses another technique, called Direct Linear Transform, for computing

homography, which gives more stable results for noisy data. DLT is described in Chapter 4 of HZ text, if you are interested, but we have not covered it in class.

v. [5 points] Now show that the two images are related by a homography i.e. the points from the first image coincide with the second image after the homography computed above. To show this, you will have to set up an error function between transformed points and original points. If they are indeed related by a homography, the error should be very low.

The whole script should be repeatable by using random points and cameras to validate the result for many point+camera configurations.

c. [Bonus 5 points] Take two images from your camera phone of a scenario that demonstrates the camera configuration in this problem. Mark at least ten correspondences between your images and use the correspondences to compute a homography between them (use Kovesi's function because your correspondences are noisy). To mark correspondences, you can use the `cpselect` tool in MATLAB. Show convincingly that your images are indeed related by a homography. .

Problem 3. [30 points] Given:

- two images of a 3D world (non-planar in general)
- taken by two cameras with
  - ZERO translation between them
  - and arbitrary (random) orientation

a. [5 points] Algebraically prove that the images are related by a 2D homography (i.e. a 2D projective transformation).

b. [25 points] Use MATLAB to simulate

i. [5 points] A set of at least 100 non-planar 3D points.

ii. [5 points] Set up two camera matrices with zero translation between them.

iii. [5 points] Take two images of the 3D points using these cameras. That is, using your camera matrices, compute the corresponding image points in each camera.

iv. [5 points] Compute a homography between the imaged points. You can use the `homography2d` function from [Peter Kovesi's site]() for computing the homography.

v. [5 points] Now show that the two images are related by a homography i.e. the points from the first image coincide with the second image after the homography computed above.

The whole script should be repeatable by using random points and cameras to validate the result for many point+camera configurations.

c. [Bonus 5 points] Take two images from your camera phone of a scenario that demonstrates the camera configuration in this problem. Mark at least ten correspondences between your images and use the correspondences to compute a homography between them. Show convincingly that your images are indeed related by a homography.

Problem 4. [20 points] The two examples above are scenarios where two images are related by homography. Discuss a scenario where two images are NOT related by homography.

a. [5 points] Provide a single counter-example and show it algebraically.

b. [15 points] Simulate a counter-example using MATLAB. Convincingly show, through your MATLAB simulation, that your images are indeed not related by a homography.

c. [Bonus 5 points] Take two real images from your camera phone of a scenario that is clearly not related by a homography. Mark at least 10 correspondences between those images and show that they are indeed not related by a homography.