Assignment 2: CS 763, Computer Vision

Due: 12th Feb before 11:55 pm

Remember the honor code while submitting this (and every other) assignment. All members of the group should work on and understand all parts of the assignment. We will adopt a zero-tolerance policy against any violation.

Submission instructions: You should ideally type out all the answers in Word (with the equation editor) or using Latex. In either case, prepare a pdf file. For assignment submission, follow the instructions for arrangement of folders and subfolders as given in `http://www.cse.iitb.ac.in/~ajitvr/CS763_Spring2017/HW2/HW2_Alignment.rar`. Create a single zip or rar file obeying the aforementioned structure and name it as follows: A2-IdNumberOfFirstStudent-IdNumberOfSecondStudent-IdNumberOfThirdStudent.zip. (If you are doing the assignment alone, the name of the zip file is A2-IdNumber.zip). Upload the file on moodle BEFORE 11:55 pm on 12th February. Late assignments will be assessed a penalty of 50% per day late. Note that only one student per group should upload their work on moodle. Please preserve a copy of all your work until the end of the semester. If you have difficulties, please do not hesitate to seek help from me.

1. Imagine you have two point-sets A and B, each containing a (possibly unequal) number of 3D points. You are told that the point-sets are related to each other by an unknown but small 3D rotation $\boldsymbol{R}$ and small 3D translation $\boldsymbol{t}$, besides a small amount of measurement noise, i.e. $\boldsymbol{a_i} = \boldsymbol{R}\boldsymbol{b_i} + \boldsymbol{t} + \boldsymbol{\eta_i}$ where $\boldsymbol{a_i}, \boldsymbol{b_i}$ are a pair of corresponding points. However, now suppose that the correspondence between the points is unknown to you. In other words the i^{th} point in A need not physically correspond to the i^{th} point in B. So consider the following iterative algorithm: (1) For each point in A, we assign its nearest neighbor in B (in terms of Euclidean distance) to be its corresponding point. (2) In the second step, we use a least squares method to determine the rotation and translation given all such pairs of corresponding points. (3) In the third step, we apply the estimated rotation and translation to every point in B. These three steps continue in an iterative manner until the estimated rotation in a given step is close to identity and the translation is close to 0.
   Your task is as follows: Implement the aforementioned algorithm using datasets provided in the homework folder. Your code should display an overlay of points from A and transformed points from B using the scatter3 command in MATLAB and include the plots from the first and last iteration in your report. Determine $\boldsymbol{R}$ and $\boldsymbol{t}$ and explain how you computed these quantities in your report. Your report should also mention which command in MATLAB you used for nearest neighbor computation. Use 'uigetdir' to allow the user to pick the folder containing the data. [20 points]

2. Enlist any four differences between the camera calibration algorithms EXPL_PARS_CAL and PROJ_MAT_CALIB from chapter 6 of the Trucco and Verri book. [10 points]

3. In the camera calibration algorithm we studied in class, it turns out that the estimate of the rotation matrix (let's call it $\hat{\boldsymbol{R}}$) is not orthonormal. The book by Trucco and Verri suggests the following procedure to 'correct' this issue by replacing $\hat{\boldsymbol{R}}$ by $\tilde{\boldsymbol{R}} = \boldsymbol{U}\boldsymbol{V}^T$ where $\hat{\boldsymbol{R}} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$ is the SVD of $\hat{\boldsymbol{R}}$. Prove that $\tilde{\boldsymbol{R}}$ as obtained by this procedure is given as $\tilde{\boldsymbol{R}} = \text{argmin}_{\boldsymbol{Q}} \|\boldsymbol{Q} - \hat{\boldsymbol{R}}\|_F^2$ subject to the constraint that $\boldsymbol{Q}\boldsymbol{Q}^T = \boldsymbol{I}$ (In other words, prove that $\tilde{\boldsymbol{R}}$ is the orthonormal matrix closest to $\hat{\boldsymbol{R}}$ in the Frobenius sense). Also this correction step brings out a limitation of this camera calibration algorithm. State that limitation. [7+3 = 10 points]

4. Consider two sets of corresponding points $\{\boldsymbol{p}_{1i} = (x_{1i}, y_{1i})\}_{i=1}^n$ and $\{\boldsymbol{p}_{2i} = (x_{2i}, y_{2i})\}_{i=1}^n$. Assume that each pair of corresponding points is related as follows: $\boldsymbol{p}_{2i} = \alpha \boldsymbol{R}\boldsymbol{p}_{1i} - \boldsymbol{t} + \boldsymbol{\eta}_i$ where $\boldsymbol{R}$ is an unknown rotation

matrix, $\boldsymbol{t}$ is an unknown translation vector, α is an unknown scalar factor and $\boldsymbol{\eta}_i$ is a vector (unknown) representing noise. Explain how you will extend the method we studied in class for estimation of $\boldsymbol{R}$ to estimate α and $\boldsymbol{t}$ as well. Derive all necessary equations (do not merely guess the answers even if they appear 'correct'). [20 points]

5. In this task, we will register two pairs of images with each other: (1) The famous barbara image (regarded as a fixed image) to be registered with its negative (regarded the moving image), and (2) a flash image (regarded as a fixed image) and a no-flash image (regarded as the moving image) of a scene. We will use the joint entropy criterion we studied in class as the objective function to be minimized for alignment. Download all required images from the homework folder. Convert all images to gray-scale (if they are in color). Note that the flash image and the no-flash image have different image intensities at many places, and the no-flash image is distinctly noisier. In the beginning you may want to either downsample or work with smaller portions of the flash and no-flash images.

   For each of the two cases, rotate the moving image counter-clockwise by 23.5 degrees, translate it by -3 pixels in the X direction, and add uniform random noise in the range [0,8] (on a 0-255 scale). Note that the rotation must be applied about the center of the image. Set negative-valued pixels to 0 and pixels with value more than 255 to 255. Now perform a brute-force search to find the angle θ and translation t_x to optimally align the modified moving image with the fixed image (in each case), so as to minimize the joint entropy. The range for θ should be between -60 and +60 in steps of 1 degree, and the range for t_x should be between -12 and +12 in steps of 1. Compute the joint entropy using a bin-size of 10 for both intensities. Plot the joint entropy as a function of θ and t_x using the surf and imshow commands of MATLAB. Comment on the difference (if any) between the quality of alignment for the first and second pair of images. Use 'uigetdir' to allow the user to input the images to be registered.

   Also, determine a scenario (for the first pair of images) where the images are obviously misaligned but the joint entropy is (falsely and undesirably) lower than the 'true' minimum. Again, display the joint entropy as mentioned before. Include all plots in your report. [20 points]

6. Refer to the paper 'Goal-directed video metrology' by Reid and Zisserman from `http://www.robots.ox.ac.uk/~vgg/publications/papers/reid96.pdf` which is an excellent and interesting application of visual metrology, and which is believed to have solved a long-standing controversy in a famous football match. The paper proposes an algorithm that takes as input two images I and I' (taken from two viewpoints) of a portion of the football field near a goal-post. Let P be the point of intersection of a line passing through an arbitrary point in 3D (say corresponding to the location of the football in mid-air) and striking the ground plane perpendicular to it. The paper aims to predict the location p and p' of the image of P inside I and I' respectively. Your task is to read and understand section 2 and figure 2 of the paper, and answer the following questions:

   (a) How does the homography computation help in obtaining p and p'?

   (b) Why is it important to compute the vertical vanishing point v and v' in the two images?

   (c) What is the significance of the lines L_s and L'_s in the task of obtaining p and p'? [10 points]

7. Consider a picture of a static (possibly non-planar) scene acquired by a camera fixed on a tripod. Now the camera is rotated but it remains fixed on the tripod without any translation, and another picture of the same scene is acquired. Let $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ be the pixel coordinates of the images of some physical point in the scene in the two pictures respectively. Note that $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ are in different coordinate systems. Derive a relation between $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ in terms of the matrix $\boldsymbol{R}$ which represents the rotational motion of the camera axes from the first position to the second, and the intrinsic parameter matrix $\boldsymbol{K_1}$ and $\boldsymbol{K_2}$ of the cameras in the two viewpoints. Note that the instrinsic parameters could change if you changed the focal length, or (hypothetically) the resolution. [10 points]