Assignment 1: CS 763, Computer Vision

Due: 2nd February 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. Put the pdf file and the code for the programming parts all in one zip file. The pdf should contain the names and ID numbers of all students in the group within the header. The pdf file should also contain instructions for running your code. Name the zip file as follows: A1-IdNumberOfFirstStudent-IdNumberOfSecondStudent-IdNumberOfThirdStudent.zip. (If you are doing the assignment alone, the name of the zip file is A1-IdNumber.zip). Upload the file on moodle BEFORE 11:55 pm on 2nd 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. Suppose you have acquired the image of a cricket pitch at the time instant that a ball thrown by the bowler landed on the ground (somewhere on the pitch) at some point say Q. Given this image, your task is to determine the perpendicular distance from Q to the line containing the wickets on the bowler's side. Make use of the standard dimensions of a cricket pitch as seen on `https://en.wikipedia.org/wiki/Cricket_pitch#/media/File:Cricket_pitch.svg`. Assume that the ball and all the sides of the pitch were clearly visible in the image. Now prove analytically that you do not need a calibrated camera for this calculation (ignoring errors due to discretization of the spatial coordinates). [3+3 = 6 points]

   Solution: Consider the two long parallel sides of the pitch. They appear to intersect at their vanishing point (say V) in the image plane. Consider a line passing through Q parallel to these two long sides. It will also have the same vanishing point V and hence line VQ intersects the two pitch baselines (one containing the wickets and one after the wickets on the bowler side) at 90 degrees at points B_1 and B_2. This produces four collinear points B_1, B_2, Q, V whose cross ratio is $\frac{B_1Q \times B_2V}{B_1V \times B_2Q}$ which is equal to the cross ratio in 3D space given as $\frac{d \times \infty}{\infty \times (d-122)} = \frac{d}{d-122}$ where d is the perpendicular distance (in 3D space) from Q onto the line containing the bowler side wickets.

   The above cross ratio inequality is valid if the points B_1, B_2, Q, V were expressed in camera coordinates as we have seen in class. But it is also true when their coordinates are expressed in the image coordinate system. Let (x_c, y_c) and (x_{im}, y_{im}) be the coordinates of a point in the camera and image coordinate system respectively. These coordinates are related by an affine transformation. We have seen in class that an affine transformation preserves the ratio of areas and also the ratios of lengths of collinear line segments. This proves that a calibrated camera is not required (except for discretization errors in marking out point coordinates - which can be assumed to be negligible if the camera resolution is fine enough).

   Marking scheme: 2 points for identifying that line VQ is parallel to the pitch sides in 3D space and 1 point for correct substitution of the cross ratios. 2 points for stating that the relationship between point coordinates in image and camera coordinate systems is an affine transformation and 1 point for stating that

affine transformation preserve ratios of lengths of collinear line segments (Deduct 0.5 points if the word collinear is missing).

2. Consider an image of a static 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 $\mathbf{p_1}$ and $\mathbf{p_2}$ be the pixel coordinates of the images of some physical point in the scene in the two images respectively. Note that $\mathbf{p_1}$ and $\mathbf{p_2}$ are in different coordinate systems. Derive a relation between $\mathbf{p_1}$ and $\mathbf{p_2}$ in terms of the matrix $\mathbf{R}$ which represents the rotational motion of the camera axes from the first position to the second, and the intrinsic parameter matrix $\mathbf{K}$ of the camera. Furthermore, if $\mathbf{K}$ is known, explain how will you determine $\mathbf{R}$. [6 points]

Solution: Consider $\mathbf{p_{1h}} = \mathbf{K}(\mathbf{R_1}|\mathbf{t_1})\mathbf{P}$ in homogeneous coordinates corresponding to the point $\mathbf{p_1}$ in pixel coordinates (see slide 60 of the camera geometry slides if you do not see the difference between $\mathbf{p_{1h}}$ and $\mathbf{p_1}$). Now we know $\mathbf{p_{2h}} = \mathbf{KR}(\mathbf{R_1}|\mathbf{t_1})\mathbf{P} = \mathbf{KRK}^{-1}\mathbf{p_{1h}} = \mathbf{Hp_{1h}}$ where $\mathbf{H}$ is a 3×3 homography matrix (not a planar homography matrix but a homography matrix of a different kind!). Given a single pair of corresponding points $\mathbf{p_1}$ and $\mathbf{p_2}$ it is not possible to determine $\mathbf{H}$, but it is possible to determine this matrix using $N \geq 8$ pairs of such points using our usual homography estimation algorithm. Now when $\mathbf{K}$ is known, we have $\mathbf{R} = \mathbf{K}^{-1}\mathbf{HK}$ if there is no noise. If there is noise, we would estimate $\mathbf{R}$ as $\text{argmin}_{\mathbf{R}*}\|\mathbf{H} - \mathbf{KR} * \mathbf{K}^{-1}\|_F^2$ subject to the constraint that $\mathbf{R} * \mathbf{R}*^T = \mathbf{I}$.

Marking scheme: 4 points for the relationship between $\mathbf{p_{1h}}$ and $\mathbf{p_{2h}}$. 1 point for the relationship $\mathbf{R} = \mathbf{K}^{-1}\mathbf{HK}$ and 1 point for stating that just one point pair is not sufficient for determining $\mathbf{H}$ (and hence $\mathbf{R}$).

3. In class, we have seen image formation on a flat screen (i.e. image plane) with a pinhole camera. Now suppose the screen was wrapped on the inner surface of a hemisphere and hence, the 3D points were projected onto a concave hemispherical surface. Derive a relationship between the coordinates of a 3D point $P = (X, Y, Z)$ and its image on such a screen (both in camera coordinate system). If you had to calibrate this sort of a system, what are the additional intrinsic parameters of the camera as compared to the case of an image plane ? [6 points]

Solution: In this answer, I am assuming the pinhole lies between the sphere and the object.
Let the equation of the sphere be $(x-a)^2+(y-b)^2+(z-c)^2 = r^2$ represented in vector form as $\|\mathbf{x}-\mathbf{d}\|^2 = r^2$ where $\mathbf{x} = (x, y, z), \mathbf{d} = (a, b, c)$. Let the image of a 3D point $\mathbf{P} = (X, Y, Z)$ on such a screen be denoted as $\mathbf{p}$. The coordinates of $\mathbf{p}$ will be given by the intersection of the line from $\mathbf{P}$ through the pinhole $\mathbf{O}$ (regarded as origin) onto the sphere. As $\mathbf{p}$ lies on line $\mathbf{O}P$, its coordinates are $\mathbf{p} = (tX, tY, tZ) = t\mathbf{m}$ where t is an unknown scalar and $\mathbf{m}$ represents the direction $\mathbf{OP}$. Then we have $\|t\mathbf{m} - \mathbf{d}\|^2 = r^2$, i.e.,

$$t^2\mathbf{m}^t\mathbf{m} - 2t(\mathbf{m}^t\mathbf{d}) + \mathbf{d}^t\mathbf{d} - r^2 = 0 \tag{1}$$

$$\therefore t = \frac{2(\mathbf{m}^t\mathbf{d}) \pm 2\sqrt{(\mathbf{m}^t\mathbf{d})^2 - \mathbf{m}^t\mathbf{m}(\mathbf{d}^t\mathbf{d} - r^2)}}{2(\mathbf{m}^t\mathbf{m})} \tag{2}$$

If the term in the square root is negative, it means that the line does not intersect the sphere and no image will be formed. If the term in the square root is 0, the solution is $t = \frac{\mathbf{m}^t\mathbf{d}}{\mathbf{m}^t\mathbf{m}}$ and corresponds to direction OP being tangential to the sphere. If the term in the square root is positive, we have two solutions corresponding to the two points of intersection. Out of these, we choose the one which is farther from the pinhole $\mathbf{O}$ (as we are considering the concave side of the hemisphere in this problem), i.e. the larger absolute value of t. The image coordinates as given $\mathbf{p} = (tX, tY, tZ) = t\mathbf{P}$.

To calibrate such a system, we would need to know r as well as $\mathbf{d}$. There is no need for a focal length parameter because given the center and radius of the sphere, the distance between any point on the sphere and $\mathbf{O}$ can be easily determined. Now, in image formation, we have to discretize the spatial coordinates. In this case, we have a sphere. We can discretize the coordinates in p in one of the following ways: (1) Project the sphere orthographically onto a screen parallel to the XOY plane. Discretize the x and y coordinates of the projection using sampling widths s_x and s_y yielding the equations $x = s_x(x_{discrete} - o_x)$ and

$y = s_y(y_{discrete} - o_y)$. OR (2) Convert the x, y, z coordinates in p to spherical coordinates ϕ and θ. Discretize ϕ and θ. Thus, in such a system, the intrinsic parameters would be: $r, (a, b, c), s_x, s_y, o_x, o_y$ OR $r, (a, b, c), s_\theta, s_\phi$.

Marking scheme: 2 points for identifying that the image is formed by the intersection of the sphere and the line **OP**. 2 points for the actual expression for the coordinates obtained by solving thw quadratic equation (deduct 1 point if there is no mention that the farther coordinate has to be picked as the screen is on the concave side of the hemisphere). 2 points for a reasonable subset of the intrinsic parameters.

4. In this exercise, we will prove the orthocenter theorem pertaining to the vanishing points Q, R, S of three mutually perpendicular directions OQ, OR, OS, where O is the pinhole (origin of camera coordinate system). Let the image plane be $Z = f$ without any loss of generality. Recall that two directions v_1 and v_2 are orthogonal if $v_1^T v_2 = 0$. One can conclude that OS is orthogonal to $OR - OQ$ (why?). Also the optical axis Oo (where o is the optical center) is orthogonal to $OR - OQ$ (why?). Hence the plane formed by triangle OSo is orthogonal to $OR - OQ$ and hence line oS is perpendicular to $OR - OQ = QR$ (why?). Likewise oR and oQ are perpendicular to QS and RS. Hence we have proved that the altitudes of the triangle QRS are concurrent at the point o. QED. Now, in this proof, I considered the three perpendicular lines to be passing through O. What do you think will happen if the three lines did not pass through O? [6 points]

   Solution: In this exercise, we will prove the orthocenter theorem pertaining to the vanishing points Q, R, S of three mutually perpendicular directions OQ, OR, OS, where O is the pinhole (origin of camera coordinate system). Let the image plane be $Z = f$. Recall that two directions v_1 and v_2 are orthogonal if $v_1^T v_2 = 0$. One can conclude that OS is orthogonal to $OR - OQ$ (**As $OS \cdot OR = OS \cdot OQ = 0$, so $OS \cdot (OR - OQ) = 0$**). Also the optical axis Oo (where o is the optical center) is orthogonal to $OR - OQ$ (**We assume the optical axis, i.e. Oo to coincide with the Z axis. But $OR - OQ$ is contained in the $Z = f$ plane, and is in fact the same as segment QR by the triangle law of vector addition. Thus Oo will be orthogonal to QR**). Hence the plane formed by triangle OSo is orthogonal to $OR - OQ$ and hence line oS is perpendicular to $OR - OQ = QR$ (**As OS and Oo are both perpendicular to QR, we must have oS perpendicular to QR by triangle law.**). Likewise oR and oQ are perpendicular to QS and RS. Hence we have proved that the altitudes of the triangle QRS are concurrent at the point o. QED. Now, in this proof, I considered the three perpendicular lines to be passing through O. How will you modify the proof if the three lines did not pass through O? **As parallel lines have the same vanishing point, there will no change to the proof.**

   Marking scheme: 1.5 points for each of the four questions to be answered in order to complete the proof.

5. Consider two sets of corresponding points $\{\mathbf{p}_{1i} = (x_{1i}, y_{1i})\}_{i=1}^n$ and $\{\mathbf{p}_{2i} = (x_{2i}, y_{2i})\}_{i=1}^n$. Assume that each pair of corresponding points is related as follows: $\mathbf{p}_{2i} = \alpha \mathbf{R} \mathbf{p}_{1i} + \mathbf{t} + \eta_i$ where $\mathbf{R}$ is an unknown rotation matrix, $\mathbf{t}$ is an unknown translation vector, α is an unknown scalar factor and η_i is a vector (unknown) representing noise. Explain how you will extend the method we studied in class for estimation of $\mathbf{R}$ to estimate α and $\mathbf{t}$ as well. Derive all necessary equations (do not merely guess the answers). [6 points]

   Solution: Let $\mathbf{P}_1$ be the $2 \times N$ matrix whose i-th column contains $\mathbf{p}_{1i}$. Likewise, define $\mathbf{P}_2$.
   We can find $\mathbf{t}$ by minimizing $E(\mathbf{t}) = \sum_{i=1}^N \|\mathbf{p}_{1i} - \alpha \mathbf{R} \mathbf{p}_{2i} - \mathbf{t}\|^2$, which gives $\sum_{i=1}^N (\mathbf{p}_{1i} - \alpha \mathbf{R} \mathbf{p}_{2i} - \mathbf{t}) = 0$, i.e. we get $\mathbf{t} = \frac{1}{N}(\sum_{i=1}^N \mathbf{p}_{1i} - \alpha \mathbf{R} \sum_{i=1}^N \mathbf{p}_{2i}) = \bar{\mathbf{p}}_1 - \alpha \mathbf{R} \bar{\mathbf{p}}_2$ where $\bar{\mathbf{p}}_1$ and $\bar{\mathbf{p}}_2$ are the average of all the points in sets $\mathbf{P}_1$ and $\mathbf{P}_2$ respectively. But we need α and $\mathbf{R}$ to find $\mathbf{t}$, so what do we do? Observe that $\sum_{i=1}^N \|\mathbf{p}_{1i} - \alpha \mathbf{R} \mathbf{p}_{2i} - \mathbf{t}\|^2 = \sum_{i=1}^N \|\mathbf{p}_{1i} - \alpha \mathbf{R} \mathbf{p}_{2i} - \bar{\mathbf{p}}_1 + \alpha \mathbf{R} \bar{\mathbf{p}}_2\|^2 = \sum_{i=1}^N \|(\mathbf{p}_{1i} - \bar{\mathbf{p}}_1) - \alpha \mathbf{R}(\mathbf{p}_{2i} - \bar{\mathbf{p}}_2)\|^2$. Now solve for $\mathbf{R}$ from the SVD of $\bar{\mathbf{P}}_2 \bar{\mathbf{P}}_1^T$ where $\bar{\mathbf{P}}_\mathbf{1}$ is the $2 \times N$ matrix whose i-th column contains $\mathbf{p}_{1i} - \bar{\mathbf{p}}_1$ (likewise $\bar{\mathbf{P}}_2$). In other words, $\mathbf{R} = \mathbf{V}\mathbf{U}^T$ where $\mathbf{U}\mathbf{S}\mathbf{V}^T = \bar{\mathbf{P}}_2 \bar{\mathbf{P}}_1^T$.

   What about α? Realize that it is only a scaling factor which will get absorbed in the $\mathbf{S}$ matrix. In other words if $\mathbf{X}$ has SVD given by $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, then theSVD of $\alpha \mathbf{X}$ is given by $\alpha \mathbf{X} = \mathbf{U} \alpha \mathbf{S} \mathbf{V}^T$. We can now solve for α by minimizing $\sum_{i=1}^N \|(\mathbf{p}_{1i} - \bar{\mathbf{p}}_1) - \alpha \mathbf{R}(\mathbf{p}_{2i} - \bar{\mathbf{p}}_2)\|^2$.

   Given α and $\mathbf{R}$, we can now find $\mathbf{t}$.

Marking scheme: 1 point for derivative w.r.t. $\mathbf{t}$ and the expression for $\mathbf{t}$ in terms of α and $\mathbf{R}$. 1 point for substituting this expression towards solving for $\mathbf{R}$. 3 points for the final expression for $\mathbf{R}$ out of which 2 points are for realizing that the α terms gets absorbed in the singular values. 1 point for the final answer for α (even a description of the approach without final expression deserves credit). Some students may substitute the parametric form for $\alpha\mathbf{R}$ in terms of α, $q_1 = \cos\theta$ and $q_2 = \sqrt{1-q_1^2} = \sin\theta$. This will produce quadratic equations. This is a sensible approach though not the most efficient one, but we will grant it full credit.

6. You are given two datasets in the folder `http://www.cse.iitb.ac.in/~ajitvr/CS763_Spring2016/HW1/Calib_data`. The file names are Features2D_dataset1.mat, Features3D_dataset1.mat, Features2D_dataset2.mat and
Features3D_dataset2.mat. Each dataset contains (1) the XYZ coordinates of N points marked out on a calibration object, and (2) the XY coordinates of their corresponding projections onto an image plane. Your job is to write a MATLAB program which will determine the 3×4 projection matrix $\mathbf{M}$ such that $\mathbf{P_1} = \mathbf{MP}$ where $\mathbf{P}$ is a $4 \times N$ matrix containing the 3D object points (in homogeneous coordinates) and $\mathbf{P_1}$ is a $3 \times N$ matrix containing the image points (in homogeneous coordinates). Use the SVD method and print out the matrix $\mathbf{M}$ on screen (include it in your pdf file as well). Write a piece of code to verify that your computed $\mathbf{M}$ is correct. For any one dataset, repeat the computation of the matrix $\mathbf{M}$ after adding zero mean i.i.d. Gaussian noise of standard deviation $\sigma = 0.05 \times max_c$ (where max_c is the maximum absolute value of the X,Y,Z coordinate across all points) to every coordinate of $\mathbf{P}$ and $\mathbf{P_1}$ (leave the homogeneous coordinates unchanged). Comment on your results. Include these comments in your pdf file that you will submit. **Tips:** A mat file can be loaded into MATLAB memory using the 'load' command. To add Gaussian noise, use the command 'randn'. [10 points]

Solution: In this section, we have to solve the equation $Am = 0$ where A is a matrix of size $N \times 12$ (N = number of points) and m is a vector of 12 elements containing elements of the projection matrix M. We solve using SVD and extract the singular vector (in V where $A = USV^T$) with the smallest singular value. Note that the question did not ask you to solve for individual parameters. You can perform a sanity check on your computation by projecting the original 3D points using the computed matrix and then computing a mean squared error between the given 2D points and the computed points. Check the code in the homework folder.

Marking scheme: 6 points for the code estimating $\mathbf{m}$, 2 points for verifying that it is correct by checking that $\|\mathbf{p} - \mathbf{MP}\|_F^2$ is small, and 2 points for the noise case.

7. In this exercise, you will estimate the homography between a pair of images using the method we studied in class. You should use the well-known SIFT algorithm to (1) detect salient feature points in both the images, and (2) determine pairs of matching points given the two point sets ('matching point pair' refers to points in the two images representing the same physical entity). The code for performing both these tasks is available at `http://www.cs.ubc.ca/~lowe/keypoints/`. We may study the internal details of how SIFT works in a separate set of lectures in class, but for this exercise, just assume this package is a magic blackbox. Now, given this set of matching pairs of points produced by the SIFT package, your job is to estimate the homography between the point sets. Write a routine of the form H = homography(im1,im2) where H is the homography matrix that will transform the first image. You will use data from the folder `http://www.cse.iitb.ac.in/~ajitvr/CS763_Spring2016/HW1/Homography/`. Do as follows:

   (a) Apply the homography transformation in the file 'Hmodel.mat' to the image 'goi1_downsampled.jpg' using reverse warping to generate a warped image. Now estimate the homography that transforms the first image into its warped version. Apply the estimated transformation to the first image (using reverse warping) and display all three images side by side in your report. Also print the model and estimated homography matrices (make sure you normalize both so that $H(3,3) = 1$ in both cases).

   (b) Determine the homography that transforms the image 'goi1_downsampled.jpg' to the second image 'goi2_downsampled.jpg'. Warp the first image (using reverse warping) and compare it to the second.

Display all three images side by side in your report. Also print the estimated homography matrix normalized so that $H(3,3) = 1$.

Note: You may not get perfect answers for the motion estimate due to errors in SIFT, but you should get a reasonable alignment. While warping, crop off the portions of the image that do not fit into the original size. You may use the nearest neighbor method for interpolation during warping. I encourage you to try out this experiment on images of planar surfaces from different viewpoints that you should take with a real camera. You will notice that the warp estimate will often be very wrong due to several incorrect matches (called as 'outliers'). In a subsequent assignment, we will implement a method that will be reasonably immune to these outliers. At that point, we will attempt to mosaic together two or more pictures as well. [10 points]

Marking scheme: For reference: my solution code is in homography_part7a.m and homography_part7b.m in the homework folder. Reverse warp code is in warp_image_homography. For SIFT, download the code from the website as mention and replace the file match.m by the one in the homework folder. 3 points for attempt at SVD based solution for homography, 3 points for reverse warp code and division by the homogeneous coordinate, 2 points for correct solution on part (a) - the warped image as per estimated homography should reasonably match the image warped by the 'model matrix' in Hmodel.mat, 2 points for correct solution in part (b) - the images should match reasonably well. Deduct 2 points if the report did not contain the image results (even if the code was correct).