Assignment 1: CS 663, Fall 2021

Due: 18th August before 11:55 pm

Remember the honor code while submitting this (and every other) assignment. You may discuss broad ideas with other students or ask me for any difficulties, but the code you implement and the answers you write must be your own. We will adopt a zero-tolerance policy against any violation.

Submission instructions: Follow the instructions for the submission format and the naming convention of your files from the submission guidelines file in the homework folder. Please see `assignment1.zip` in the homework folder. For all the questions, write your answers and scan them, or type them out in word/Latex. In eithe case, create a separate PDF file. The last two questions will also have code in addition to the PDF file. Once you have finished the solutions to all questions, prepare a single zip file and upload the file on moodle before 11:55 pm on 25th August. We will not penalize submission of the files till 10 am on 26th August. **No assignments will be accepted after this time.** Please preserve a copy of all your work until the end of the semester. **Your zip file should have the following naming convention:** RollNumber1_RollNumber2_RollNumber3.zip for three-member groups, RollNumber1_RollNumber2.zip for two-member groups and RollNumber1.zip for single-member groups.

1. Consider the following motion models: translation, rigid (translation + rotation), rigid and equal scaling in X,Y directions, rigid and unequal scaling in X,Y directions, affine, non-rigid. Consider each of the following applications separately. In each case, identify what is the optimal motion model and justify. Do not needlessly pick a more complex motion model if it is not needed. For example, in some cases, a rotation is enough, in which case please do not choose affine as it has additional degrees of freedom. [15 points]

   (a) Consider that you have scanned a document twice with the same scanner, when the document was potentially in slightly different positions. You now want to align these two images. Which motion model is needed here? Assume there is no stretching or bending of the paper.

   (b) In the earlier example, consider that the two images were respectively acquired from two different scanners with different resolutions. Assume that for both scanners, the X and Y resolutions were the same. Which motion model is needed here? Assume there is no stretching or bending of the paper.

   (c) Consider a document with words written on both sides with ink. When you scan such a document from one side, some portions from the other side are visible. This is called 'ink bleeding'. To remove bleeding artifacts, you need to acquire images of both sides of the document and first align them. Which motion model is needed here? Assume there is no stretching or bending of the paper.

   Solution:

   (a) The optimal motion model in the absence of stretching and bending is rigid (rotation + translation). 5 points for correct solution. Affine is not necessary. You will get only 2.5 points if you mention affine transformations.

   (b) The optimal motion model here is rigid motion followed by scaling in X and Y directions. 5 points for correct solution. You will get only 3 points if you mention affine as this is not the full affine model. If you forget to mention scaling, you will get only 2 points.

   (c) The optimal model here is rigid motion (rotation and translation) followed by reflection about X axis. 5 points for correct solution. You will get only 3 points if you mention affine as this is not the full affine model. If you forget to mention reflection, you will get only 2 points.

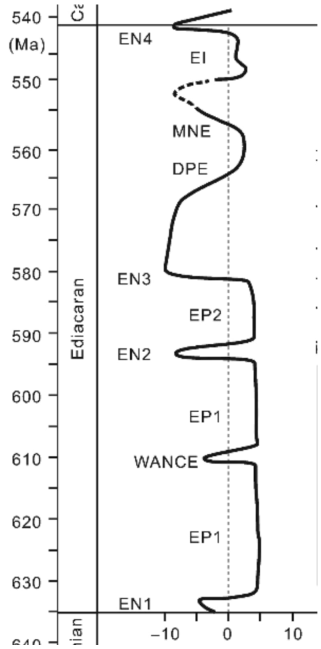

Figure 1: Graph used for question 2.

2. You are viewing the graph in Fig. 1 from a research paper. Unfortunately from the graph, the (x, y) values at only a few points can be observed. You need to obtain the (x, y) values at many other points. Hence you can do the following: you extract the image from the paper, and open it through MATLAB which provides a function called impixelinfo. This function gives you the (x, y) coordinates of any spatial location pointed by your mouse. However, the coordinate system of the graph and that of MATLAB will be different. Describe a procedure to convert the coordinates of any point from MATLAB's coordinate system to the coordinate system of the graph. This will help you obtain the (x, y) coordinates in the coordinate system of the graph. Support your answer with suitable equations. There is no need to write any code for this. [15 points]
Solution: Pick any 3-4 points from the graph with observable coordinates in the graph's coordinate system. Let us call these points $\{(g_{xi}, g_{yi})\}_{i=1}^4$. The easiest ones to pick are the two points where the dotted line intersects the two solid parallel horizontal lines, the point close to C1 (see figure) and the point of intersection of the lines EN1 and Ediacaran. However any 3+ points for which the coordinates in the graph are accurately observable, can be chosen. Open the figure in MATLAB and note down the corresponding coordinates in MATLAB's coordinate system using `impixelinfo`. Call these coordinates $\{(m_{xi}, m_{yi})\}_{i=1}^4$. Then we will have $\boldsymbol{G} = \boldsymbol{MA}$ where $\boldsymbol{G}$ is the 4×3 matrix whose first column contains x coordinates $\{g_{xi}\}$, second column contains y coordinates $\{g_{yi}\}$, and third column contains all ones. Likewise, $\boldsymbol{M}$ is the 4×3 matrix whose first column contains x coordinates $\{m_{xi}\}$, second column contains y coordinates $\{m_{yi}\}$, and third column contains all ones. Also, $\boldsymbol{A}$ is the 3×3 affine motion matrix which can be obtained as $\boldsymbol{A} = \boldsymbol{M}^T\boldsymbol{M}^{-1}\boldsymbol{M}^T\boldsymbol{G}$. Now, if you observe the coordinates of any point in MATLAB, you can get the coordinates of the same point in the graph coordinate system using the earlier relation.
marking scheme: The equations carry 7.5 points. The broad idea carries 7.5 points.

3. Consider two images I and J whose intensity values (in each location) are randomly drawn from the known probability mass functions (PMFs) $p_I(i)$ and $p_J(j)$ respectively. Derive an expression for the PMF of the image $I + J$. The expression resembles which operation that we are currently studying in class (and will discuss on 13th August during the interaction session)? [15+5=20 points]
Answer: Let the image $I+J$ be denoted as K. Then $p_K(k) = \int_{i=-\infty}^{+\infty} p_{IJ}(i, k-i)di = \int_{i=-\infty}^{+\infty} p_I(i)p_J(k-i)di$. The latter equality follows if I and J are independent, and leads to a convolution integral. If the integration were replaced by a summation for discrete-valued images, this is exactly the expression for convolution we have done in class.
Marking scheme: The first equality carries 10 points, 5 points for the second equality which uses independence. Identifying it as convolution carries another 5 points. Since the condition of independence wasn't

explicitly mentioned in the question, full points are to be given if the second equality is not mentioned or if the convolution integral is not mentioned.

4. Read in the images T1.jpg and T2.jpg from the homework folder using the MATLAB function `imread` and cast them as a double array. Let us call these images as J1 and J2. These are magnetic resonance images of a portion of the human brain, acquired with different settings of the MRI machine. They both represent the same anatomical structures and are perfectly aligned (i.e. any pixel at location (x, y) in both images represents the exact same physical entity). We are going to perform a simulation experiment for image alignment in a setting where the image intensities of physically corresponding pixels are different. To this end, do as follows:

   (a) Write a piece of MATLAB code to rotate the second image by $\theta = 28.5$ degrees anti-clockwise. You can use the `imrotate` function in MATLAB to implement the rotation using any interpolation method. Note that the rotation is performed implicitly about the centroid of the image. While doing so, assign a value of 0 to unoccupied pixels. Let us denote the rotated version of J2 as J3.

   (b) Our job will now be to align J3 with J1 keeping J1 fixed. To this end, we will do a brute-force search over θ ranging from -45 to +45 degrees in steps of 1 degree. For each θ, apply the rotation to J3 to create an intermediate image J4, and compute the following measures of dependence between J1 and J4:

   - the normalized cross-correlation (NCC)
   - the joint entropy (JE)
   - a measure of dependence called quadratic mutual information (QMI) defined as $\sum_{i_1} \sum_{i_2} (p_{I_1 I_2}(i_1, i_2) - p_{I_1}(i_1) p_{I_2}(i_2))^2$, where $p_{I_1 I_2}(i_1, i_2)$ represents the normalized joint histogram (*i.e.*, joint pmf) of I_1 and I_2 ('normalized' means that the entries sum up to one). Here, the random variables I_1, I_2 denote the pixel intensities from the two images respectively. For computing the joint histogram, use a bin-width of 10 in both I_1 and I_2. For computing the marginal histograms p_{I_1} and p_{I_2}, you need to integrate the joint histogram along one of the two directions respectively. You should write your own joint histogram routine in MATLAB - do not use any inbuilt functions for it.

   (c) Plot separate graphs of the values of NCC, JE, QMI versus θ and include them in the report PDF.

   (d) Determine the optimal rotation between J3 and J1 using each of these three measures. What do you observe from the plots with regard to estimating the rotation? Explain in the report PDF.

   (e) For the optimal rotation using JE, plot the joint histogram between J1 and J4 using the `imagesc` function in MATLAB along with `colorbar`. Include it in the report PDF.

   (f) We have studied NCC and JE in class. What is the intuition regarding QMI? Explain in the report PDF. (Hint: When would random variables I_1 and I_2 be considered statistically independent?) [2+10+2+3+3+5=25 points]

   Answer: See code in homework folder. The maximum of QMI and the minimum of JE are quite close to 29 degrees. However the NCC measure completely fails as the assumption of linear relationship between the intensities of the two images fails. The QMI measure determines the squared difference between the joint PMF and the product of the marginals. If this difference is 0 (its minimum value), then the two random variables I_1, I_2 are statistically independent. When two images are well-aligned, the intensities cannot be statistically independent. Hence, we seek to maximize the difference between p_{12} and $p_1 p_2$ for all intensities. This is in line with the empirical result where the QMI achieves a maximum value close to 29 degrees.
   Marking scheme: The marking scheme is already given. Deduct 1 point for every plot not included in the report (even if the code produces it).

5. Read in the images 'goi1.jpg' and 'goi2.jpg' from the homework folder using the MATLAB `imread` function and cast them as double. These are images of the Gateway of India acquired from two different viewpoints. As such, no motion model we have studied in class is really adequate for representing the motion between these images, but it turns out that an affine model is a reasonably good approximation, and you will see this. We will estimate the affine transformation between these two images in the following manner:

(a) Display both images using `imshow(im1)` and `imshow(im2)` in MATLAB. Use the `ginput` function of MATLAB to manually select (via an easy graphical user interface) and store $n = 12$ pairs of physically corresponding salient feature points from both the images. For this, you can do the following:
```
for i=1:12, figure(1); imshow(im1/255); [x1(i), y1(i)] = ginput(1);
figure(2); imshow(im2/255); [x2(i), y2(i)] = ginput(1);
```
Tips: Avoid selecting points which are visible in only one image. Try to select them as accurately as possible, but our procedure is robust to small sub-pixel errors. Make sure `x1(i),y1(i)` and `x2(i),y2(i)` are actually physically corresponding points. Salient feature points are typically points that represent corners of various structures.

(b) Write MATLAB code to determine the affine transformation which converts the first image ('goi1') into the second one ('goi2').

(c) Using nearest neighbor interpolation that you should implement yourself, warp the first image with the affine transformation matrix determined in the previous step, so that it is now better aligned with the second image. You are not allowed to use any implementation for this already available in MATLAB. Display all three images side by side in the report PDF.

(d) Repeat the previous step with bilinear interpolation that you should implement yourself. You are not allowed to use any implementation for this already available in MATLAB. Display all three images side by side in the report PDF.

(e) In the first step, suppose that the n points you chose in the first image happened to be collinear. Explain (in the report PDF) the effect on the estimation of the affine transformation matrix. [5+5+5+5+5=25 points]

Answer and marking scheme: Marking scheme is already provided. See code in homework folder. If the n points were collinear: Let the matrix of points from image 1 and image 2 be denoted as $\boldsymbol{P_1}$ and $\boldsymbol{P_2}$ respectively. Both have size $3 \times n$. We have the relationship $\boldsymbol{AP_1} = \boldsymbol{P_2}$, and the least squares solution for $\boldsymbol{A}$ is given as $\boldsymbol{A} = \boldsymbol{P_2}\boldsymbol{P_1}^t\text{inverse}(\boldsymbol{P_1}\boldsymbol{P_1}^t)$. However if the n points are collinear, then $\boldsymbol{P_1}\boldsymbol{P_1}^t$ will have rank equal to 1 and hence the solution for $\boldsymbol{A}$ will not be unique.