CIS581, Computer Vision
MIDTERM - FALL 2015

Name: ______________________________

- Closed lecture notes
- One cheat sheet allowed
- 1.5 hours
- Express numeric answers exactly unless otherwise stated

1. Convolution

(a) Edge Moving (20 points)

To match patterns in images, small visual change may lead to large dismatching. Therefore, we oftentimes perform a kernel convolution ahead of matching. In this question, we want to explore some intuition behind construction of such a kernal. Assume around-boundary reflection padding.

Consider that we want to match the following two edges that are slightly apart

$$I_1 = \begin{bmatrix} 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 1 & 2 & 0 \\ 0 & 1 & 2 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$I_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 2 & 1 & 0 \\ 0 & 2 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

The sum of squared distance $SSD(X,Y)$ is defined as $\sum_{i,j}(X(i,j) - Y(i,j))^2$. Therefore we know

$$SSD(I_1, I_2) = 8,$$

In addition, we could compute the mean of them,

$$I_m = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} & 2 \\ 0 & 0 & \frac{1}{2} & 2 & \frac{1}{2} \\ 0 & \frac{1}{2} & 2 & \frac{1}{2} & 0 \\ \frac{1}{2} & 2 & \frac{1}{2} & 0 & 0 \\ 2 & \frac{1}{2} & 0 & 0 & 0 \end{bmatrix}.$$

and find

$$SSD(I_1, I_m) = SSD(I_2, I_m) = 2,$$

which is much smaller than $SSD(I_1, I_2)$.

(1) (5 points) Find a 3×3 kernel K_1 such that $I_1 \bigotimes K_1 = I_m$, and a 3×3 kernel K_2 such that $I_2 \bigotimes K_2 = I_m$. ($\bigotimes$ denotes convoltuion.)

(2) (15 points) Construct a 3×3 kernel K by averaging K_1 and K_2. K transforms I_1 and I_2 to I_1' and I_2' respectively after convolution, i.e., $I_1 \bigotimes K = I_1'$ and $I_2 \bigotimes K = I_2'$. Please show I_1' and I_2', and also caculate the resultant $SSD(I_1', I_2')$. Compare $SSD(I_1', I_2')$ with $SSD(I_1, I_2)$, and briefly explain what you find and why.

(b) Image Enlargement (20 points)

In the image pyramid lecture, an EXPAND function was given to re-enlarge an image that had previously been down-sampled. This is similar to the "splatting" that can be used to fill holes when doing a forward image morph. Both operations can be performed by an image convolution with an appropriate kernel. The sample image I below was enlarged to double its size to get I' by adding empty rows and columns. In this question you will use convolution to splatter the values given to fill in the gaps. Rather than using the gaussian kernel from the pyramid lecture, we'll be using a simpler interpolation.

$$I = \begin{bmatrix} 128 & 64 & 32.5 \\ 32 & 64.5 & 128 \\ 64 & 32 & 128 \end{bmatrix} \quad I' = \begin{bmatrix} 128 & 0 & 64 & 0 & 32.5 \\ 0 & 0 & 0 & 0 & 0 \\ 32 & 0 & 64.5 & 0 & 128 \\ 0 & 0 & 0 & 0 & 0 \\ 64 & 0 & 32 & 0 & 128 \end{bmatrix}$$

(1) (10 points) Which kernel below is best for this interpolation? Why?

$$A) \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & 1 & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix} \quad B) \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix} \quad C) \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix} \quad D) \begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}$$

(2) (10 points) Convolve I' with your selected kernel assuming zero padding

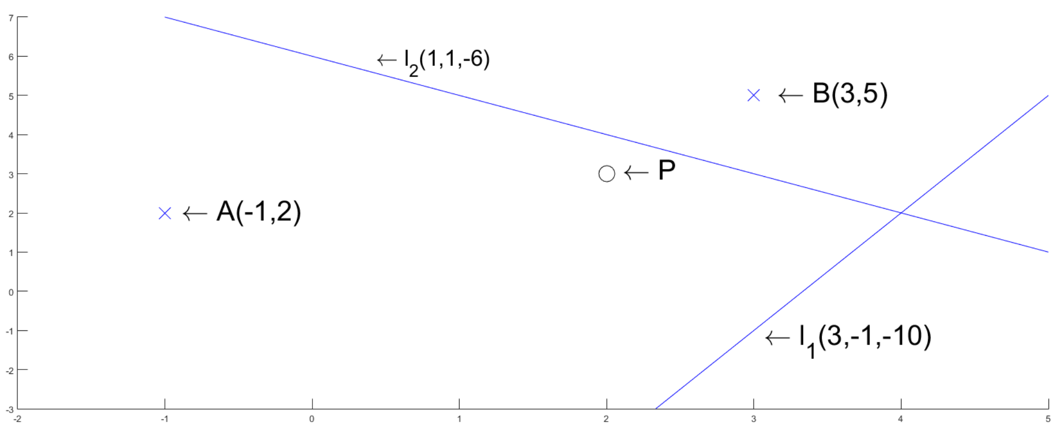
2. Warping

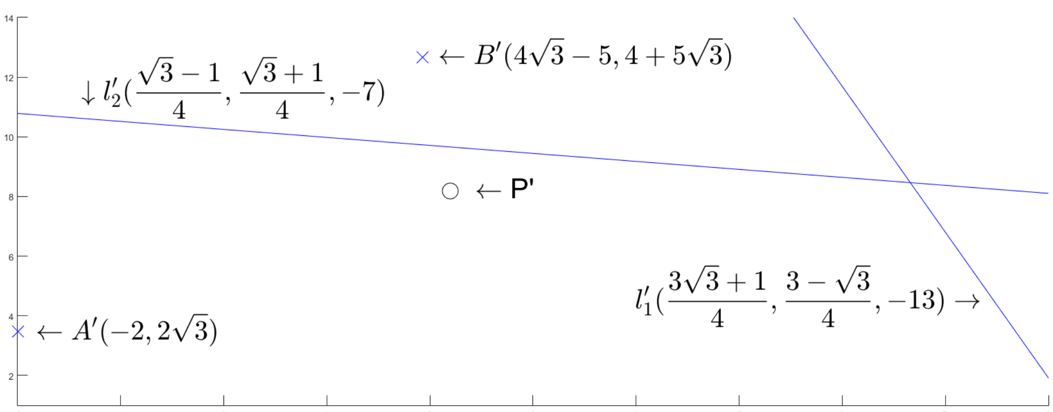
(a) Affine Transformation (20 points)

As discussed in class, both lines and points can be represented as triplets of numbers in homogeneous coordinates. Homogeneous coordinates for lines l_1, l_2, l_1', l_2' and points A, B, A', B' have been given below. This information is sufficient to determine a transformation between I and I'.

$$A = (-1, 2) \qquad B = (3, 5)$$
$$A' = (-2, 2\sqrt{3}) \qquad B' = (4\sqrt{3} - 5, 4 + 5\sqrt{3})$$

$$l_1 : 3x - y - 10 = 0 \qquad l_2 : x + y - 6 = 0$$
$$l_1' : \frac{3\sqrt{3}+1}{4}x + \frac{3-\sqrt{3}}{4}y - 13 = 0 \qquad l_2' = \frac{\sqrt{3}-1}{4}x + \frac{\sqrt{3}+1}{4}y - 7 = 0$$





(1) (10 points) Calculate the intersection of l_1, l_2 and also the intersection of l'_1 and l'_2 with homogeneous coordinates. *Note: If you are unsure as to how to do this, you may compute this in any way you want, but note that only partial credit will be awarded*

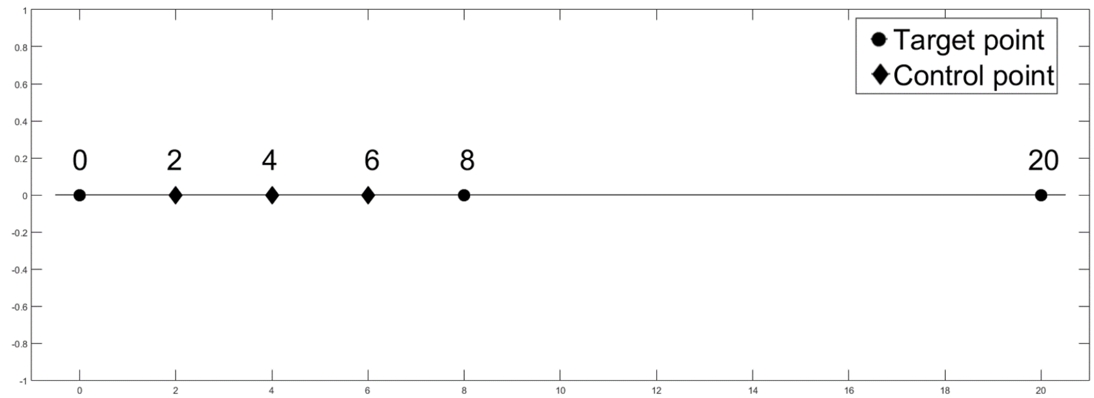
(2) (10 points) You are told that the transformation mapping I to I' is an Affine transformation involving a rotation, scale and translation (in that order).
There exists a point P in I such that

$$P = (2, 3)$$

Utilizing the concepts studied in class and applied in project 2, compute the position of the warp of P, P' in I'.

(b) Exploring Estimate TPS (Optional) (20 points)

A TPS model is built by solving a set of linear equations such that the model maps the control points to the target points. This model consists of 3 affine parameters and a weighted combination of functions centered on each control point. This problem attempts to illustrate what the estimate TPS function is doing on a simpler problem in one dimension.

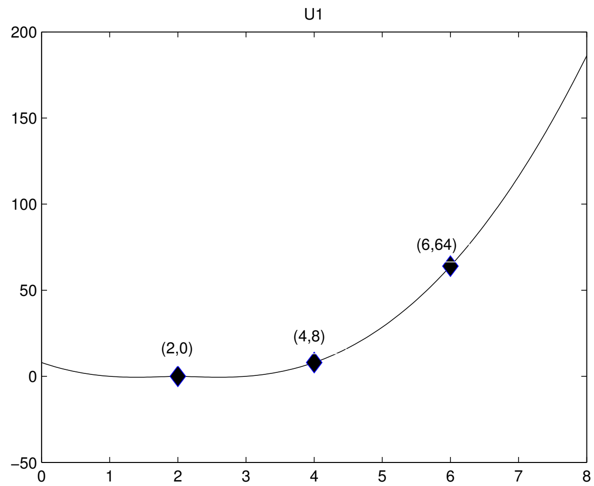


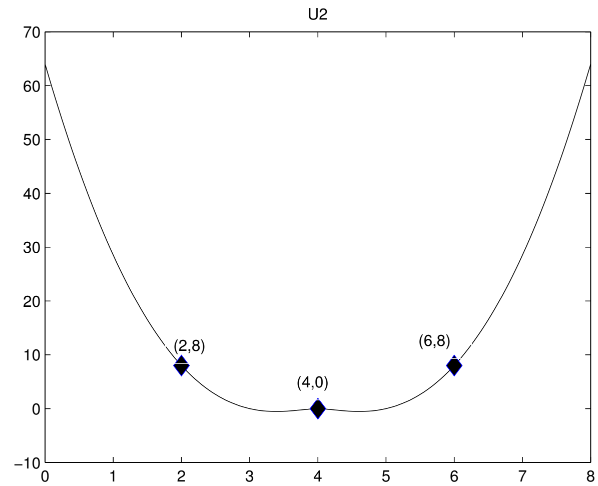
Consider the plot above. We want to find a function that maps the control points to the target points (2 to 0, 4 to 8, 6 to 20). Our model will be:

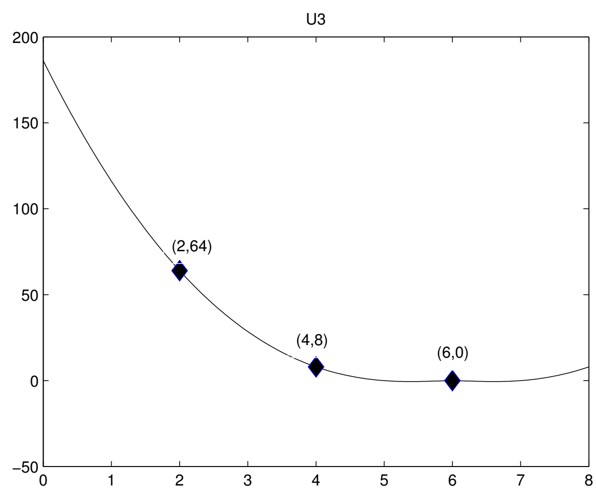
$$f(x) = \sum_{i=1}^{3} w_i U_i(x)$$

Where the U_i are basis functions given by the plots below and the w_i are the parameters in the model. (Note that the affine terms that would be present in a TPS model have been removed to reduce the number of variables)

Solve for w_1, w_2, and w_3.


U1
(2,0)
(4,8)
(6,64)


U2
(2,8)
(4,0)
(6,8)


U3
(2,64)
(4,8)
(6,0)

3. Image Carving and Image Mosaic

(a) Image Carving (10 points).

In image carving, to remove a vertical seam from the image, we compute the minimum energy path with dynamic programming. The energy map E is shown below,

E =

5.5	8.5	0.5	3	1
2	0.5	1	1.5	1.5
2.5	1.5	3.5	3	1

Recall that dynamic programming constructs matrix M recursively, i.e.,

$$M(i,j) = E(i,j) + \min(M(i-1,j-1), M(i-1,j), M(i-1,j+1)),$$

and $T(i,j) \in \{1,2,3\}$ records the path choice of "left", "center", and "right" from previous row.

Finish the cumulative minimum table M and the backtrack table T, and find the vertical seam to be removed.

M =

5.5	8.5	0.5	3	1

T =

0	0	0	0	0

(b) Image Stitching (10 points).

The task is to stitch images A, B and C together. After computing feature matching and RANSAC, we determine the homography transformation for warping points from image A to image B as T_A^B and from image C to image B as T_C^B.

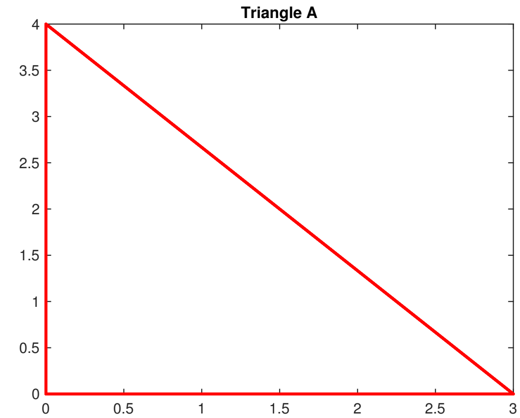
$$T_A^B = \begin{bmatrix} 1 & -1/5 & -300 \\ 1/3 & 1 & -150 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad T_C^B = \begin{bmatrix} 3/4 & 1/5 & 300 \\ -1/3 & 1 & 30 \\ 0 & 0 & 1 \end{bmatrix}.$$

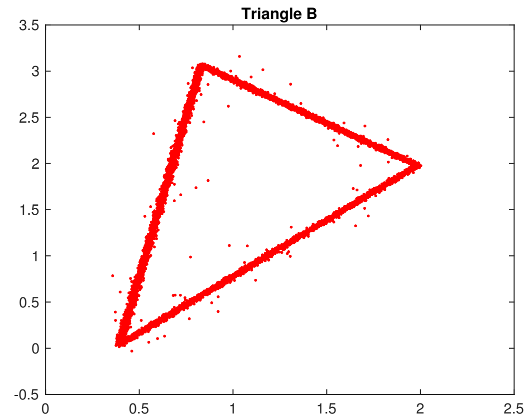
(1) The size of A, B, and C is 600 rows (height) and 500 columns (width), i.e., $(500, 600)$. The origin is $(0, 0)$.
Let A_{morph} (C_{morph}) be the image resulting from transformation T_A^B (T_C^B), which aligns with image B.

What are the spatial extent of the morphed images A_{morph} and C_{morph}? What is the size (height and width) of stitched image S?

(c) RANSAC (20 points).

A transformation from triangle A in image 1 to triangle B in image 2 can be modeled as an affine transformation. Because of noise in the data, we decide to use RANSAC for affine fitting.





(1) (10 points) Provide a description of how RANSAC works herefor this task. It could be a paragraph or pseudo code.
You may assume that you have points correspondences between images, a function to calculate the affine transformation, and a function to compute the distance of a point to the triangle.

(2) (10 points) Assume that there are 20% outliers, what is the minimum number of RANSAC interations needed to get, with probability 90%, at least one random sample that is free from outliers?

$\log_{10} 2$	0.30
$\log_{10} 3$	0.48
$\log_{10} 5$	0.70
$\log_{10} 7$	0.85
$\log_{10} 37$	1.57
$\log_{10} 61$	1.79
$\log_{10} 89$	1.95

Blank Page