Prelim
Computer Science 4670, Cornell University
Prof. Bala 17 March 2016

Solutions

1. Write your name and NetID. Eat a cookie/brownie. [0 pts]

2. Short answers. [21 pts] (parts a–d)

   (a) [6 pts] A student has collected a point cloud (collection of 3D points) obtained from scanning a wall in Gates Hall with a 3D scanner. To clean up the data, they decide to fit a 3D plane to the data. They have implemented a buggy version of the RANSAC algorithm. Help out the student by correcting the bugs in their pseudocode below. *Hint: there are at least 3 bugs.*

   Initialize the number of inliers M to 0.

   Repeat for T iterations

   - Sample 2 random points at random. Fit a plane Π to the sampled points.
   - Measure the distance of all points from the plane Π. All points with a distance $d > \epsilon$ are inliers. Let C be this count of inliers, and S be the set of inliers found. If $C > M$, $M = C$. $BestFit = \Pi$, $BestPoints = S$.
   - Improve the plane estimate by re-fitting it to all points in the dataset (i.e., find the plane with the lowest least squares error to all points in the dataset). This gives an improved plane Π'.

   Answer:

   *[2 pts] Sample **3** points*

   [2 pts] $d < \epsilon$

   [2 pts] Re-fit to inliers in BestPoints only.

   (b) [4 pts] Convolving a signal with a filter in the spatial domain corresponds to multiplication in the Fourier domain. Specify the Fourier transform that corresponds to the following filters in the spatial domain. Sketch them (roughly).

   i. box

   Answer:

   [2 pts] sinc

   ii. Gaussian

   Answer:

   [2 pts] Gaussian

(c) [5 pts] Precision and recall are often used to characterize the quality of computer vision algorithms. You are designing an object recognition algorithm for a robot that is sorting bottles and cans. Out of 500 items processed, 200 are bottles, and 300 are cans. Your algorithm reports that 200 are cans. Of these 150 are truly cans, and 50 are actually bottles.

i. What is your algorithm's precision?
Answer:

150/200

ii. What is your algorithm's recall?
Answer:

150/300

(d) [6 pts] A Harris corner detector identifies the 2 radii of an ellipse: λ_1 and λ_2.

i. Briefly explain the significance of these values.
Answer:
[2 pt] Show the error curve and its bowl shape. Recognize that the cross-section is an ellipse.

ii. Briefly explain how these values can be compared to determine if there is a corner. Consider all 4 cases.

A. Case 1:
Answer:

[1 pt] If the λ_1 and λ_2 are both small, no corner.

B. Case 2:
Answer:

[1 pt] For an edge the λ_1 is much greater than the λ_2, or vice versa.

C. Case 3:
Answer:

[1 pt] For an edge the λ_2 is much greater than the λ_1, or vice versa.

D. Case 4:
Answer:

[1 pt] If the λ_1 and λ_2 are approximately the same, corner.

3. Image Alignment [12 pts] (parts a–d)

You want to align two images I_1 and I_2 to create a spectacular panorama. You have already run your feature detector and successfully found three matching point pairs: $i \in \{1, 2, 3\} : p_i = (x_i, y_i)$ matches $p'_i = (x'_i, y'_i)$. Your task is to solve for the image transformation matrix T which aligns I_1 to I_2, such that the matching points are transformed on top of each other. $p'_i = Tp_i$

(a) [3 pts] We assume that T is a combination of scaling and translation, where s_x, s_y are the scaling on the x and y axes, and t_x and t_y are the translation on the x and y axes respectively. Write down the general form of this matrix.

Answer:

Solution:

$$i. \begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix} \quad or \quad \begin{bmatrix} s_x & 0 & s_x t_x \\ 0 & s_y & s_y t_y \\ 0 & 0 & 1 \end{bmatrix}$$

(1 pt) Partial credit for not writing the matrix, but wrote some sensible equation.

(b) [3 pts] Write down a linear equation which solves for the alignment transformation, where the unknown parameter column vector is:

$$h = \begin{bmatrix} s_x \\ s_y \\ t_x \\ t_y \end{bmatrix}$$

Answer:

Showing solution only for the first version.

$$A = \begin{bmatrix} x_1 & 0 & 1 & 0 \\ 0 & y_1 & 0 & 1 \\ x_2 & 0 & 1 & 0 \\ 0 & y_2 & 0 & 1 \\ x_3 & 0 & 1 & 0 \\ 0 & y_3 & 0 & 1 \end{bmatrix}, \; b = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \end{bmatrix} \quad and\ Ah = b.$$

(c) [3 pts] What expression would you minimize to get the least squares solution?

Answer:

$$\min_h ||Ah - b||_2$$

(d) [3 pts] How many point correspondences are required at a minimum to solve this equation?

Answer:

2

4. Filtering [20 pts] (parts a–h) Consider operations that take a grayscale image $f(m, n)$ as input, and output a grayscale image $g(m, n)$. Which of the following operations can be written as a single 2D convolutional filter $g = w * f$ for some w? The filter operates on pixels (i.e., it is discrete). The filter w must be constant (i.e., the filter w cannot depend on f or its size). If such a filter exists, write it down. Otherwise write "No such filter."

(a) [2 pts] Blur f with a 3x3 box filter.

Answer:

$w = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

(b) [2 pts] Increase the contrast by a factor $A > 1$: $g(m,n) = A \cdot f(m,n)$.

Answer:

$w = \begin{bmatrix} A \end{bmatrix}$

(c) [2 pts] Shift the brightness by a constant $C \neq 0$: $g(m,n) = C + f(m,n)$.

Answer:

No such filter.

(d) [2 pts] This function: $g(m,n) = 4f(m,n) - 2f(m,n-1) - f(m,n+1)$.

Answer:

$w = \begin{bmatrix} -1 & 4 & -2 \end{bmatrix}$

(e) [2 pts] Rotate f counter-clockwise by angle θ (in radians) about the origin.

Answer:

No such filter.

(f) [2 pts] Flip f about the first axis: $g(m,n) = f(-m,n)$.

Answer:

No such filter.

(g) [2 pts] The thresholding operator for some constant A:

$$g(m,n) = \begin{cases} 255 & \text{if } f(m,n) \geq A \\ 0 & \text{else} \end{cases}$$

Answer:

No such filter.

(h) [6 pts] Filtering f by a convolutional filter b, and then by a second filter a:

$$g = a * (b * f)$$

Both a and b are 3×1 convolutional filters:

$$a = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix} \qquad b = \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

Answer:

$w = \begin{bmatrix} 1 & 2 & 0 & -2 & -1 \end{bmatrix}$

5. Pyramids [14 pts] (parts a–c)

You are writing a unit test for code which computes image pyramids. You are going to work out the correct answer by hand for a small example to make sure you are computing it right. G_0 is the base level image and K is the kernel used to filter it. Note that we normally use a Gaussian for K, but for simpler math you will use a box filter for K. Use zero when sampling outside image boundaries.

$$G_0 = \begin{array}{|c|c|c|c|} \hline 1 & 1 & 0 & 1 \\ \hline 1 & 1 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 \\ \hline \end{array} \qquad K = \frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

(a) [6 pts] Compute the **Gaussian pyramid** of G_0 using K as the filter. Take the 1st and 3rd rows (columns) when downsampling.

Answer:

Rubric: *5a. +1 point per cell of G1, +2 points for G2 5b. +3 for the concept of DoG, +3 for the concept of pyramid shape 5c. +2 for the general case answer*

Answer:

$$G_1 = \frac{1}{9} \begin{array}{|c|c|} \hline 4 & 4 \\ \hline 2 & 4 \\ \hline \end{array} \qquad G_2 = \frac{1}{81} \begin{array}{|c|} \hline 14 \\ \hline \end{array}$$

(b) [6 pts]

Compute the **Laplacian pyramid** of G_0. Use a 2×2 box filter for upsampling. Use your answers from the previous section for G_i.

Answer:

$$L_0 = \frac{1}{9} \begin{array}{|c|c|c|c|} \hline 5 & 5 & -4 & 5 \\ \hline 5 & 5 & -4 & 5 \\ \hline -2 & -2 & -4 & 5 \\ \hline -2 & -2 & -4 & 5 \\ \hline \end{array} \qquad L_1 = \frac{1}{81} \begin{array}{|c|c|} \hline 22 & 22 \\ \hline 4 & 22 \\ \hline \end{array} \qquad L_2 = G_2$$

(c) [2 pts] You are estimating the amount of memory needed to store a pyramid. Express your estimate in terms of N. If it takes N megabytes to store G_0, how much memory is needed to store a complete **Gaussian pyramid**?

Answer:

$\frac{4}{3}N$ *megabytes.*

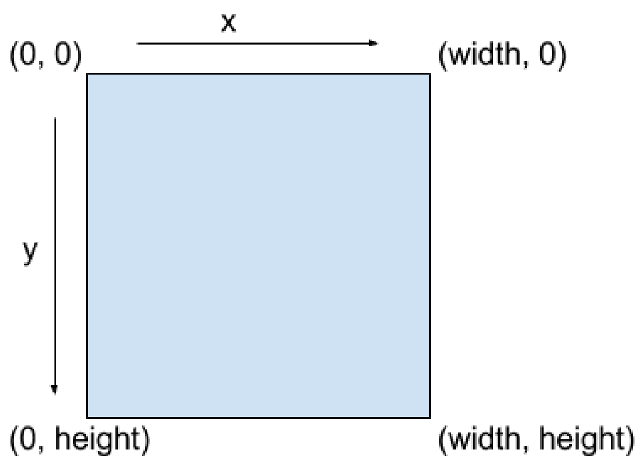
6. Transformations [33 pts] (parts a–f)

You are working on the next hot mobile photography app and you want to enable some simple image transformation operations. At your disposal is a well-tested routine that will transform and resample an image according to the following 3x3 transformation T. You would like to make the best use of this routine by specifying several common operations in terms of different T matrices.

$$T = \begin{bmatrix} \mathrm{A} & \mathrm{B} & \mathrm{C} \\ \mathrm{D} & \mathrm{E} & \mathrm{F} \\ \mathrm{G} & \mathrm{H} & 1 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} x_{\text{src}} w \\ y_{\text{src}} w \\ w \end{bmatrix} = T \begin{bmatrix} x_{\text{dst}} \\ y_{\text{dst}} \\ 1 \end{bmatrix} \tag{2}$$

The coordinates of your images are in the following format.



(a) [2 pts] What constraints do you need to apply to the elements of T such that T is always an affine transformation.

Answer:

G = H = 0

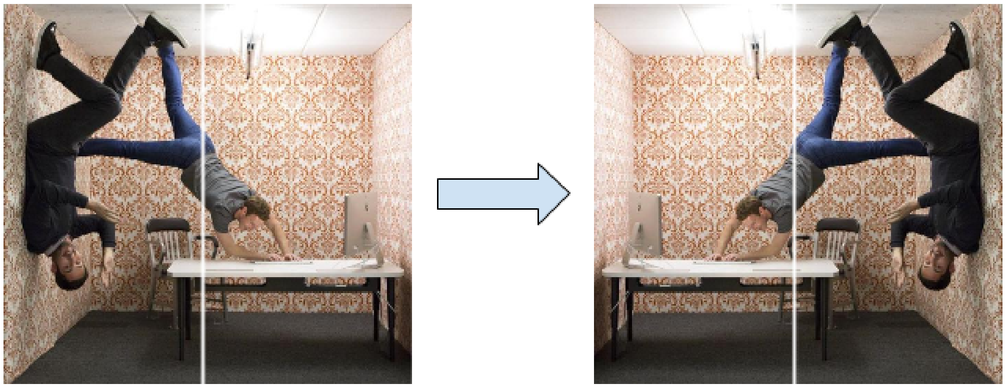
(b) [2 pts] Does T specify forward warping or inverse warping? Why might we choose this direction over the other?

Answer:

Inverse warping. Prevents holes.

Answer:

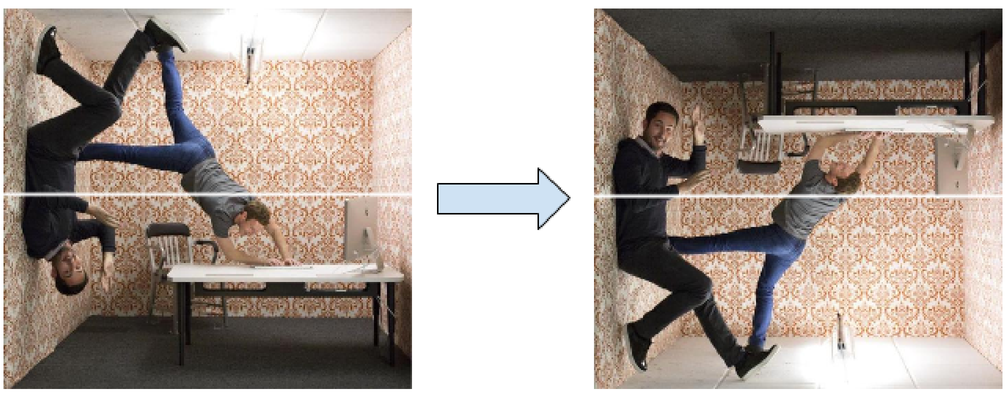
(c) [12 pts] Sometimes a user might like to reflect their image along either the vertical or horizontal axes or rotate about the center. **Remember that the upper left corner is (0, 0) in your coordinate system.** Assume the image is a square of 200×200 pixels. Write the transform T that achieves the following effects:



i. Reflect the image along the vertical axis.
   $T =$
   Answer:

$$T = \begin{bmatrix} \textit{-1} & \textit{0} & \textit{200} \\ \textit{0} & \textit{1} & \textit{0} \\ \textit{0} & \textit{0} & \textit{1} \end{bmatrix} \tag{3}$$

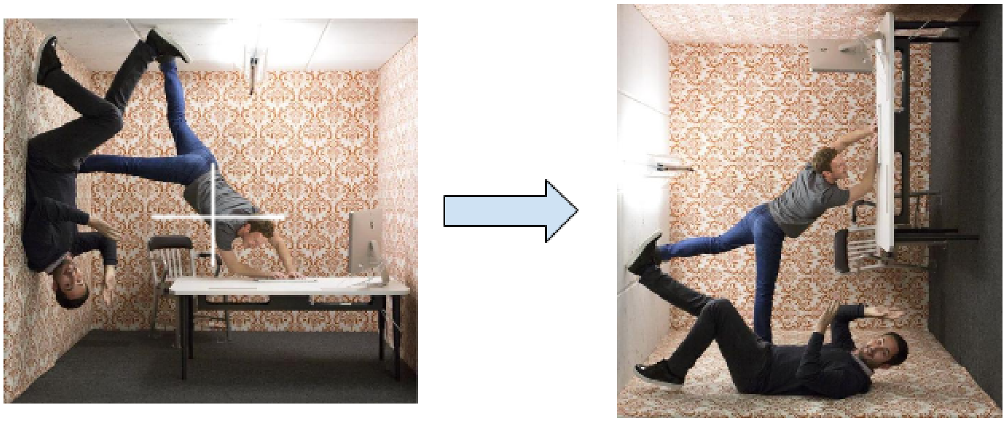
ii. Reflect the image along the horizontal axis.



   $T =$
   Answer:

$$T = \begin{bmatrix} \textit{1} & \textit{0} & \textit{0} \\ \textit{0} & \textit{-1} & \textit{200} \\ \textit{0} & \textit{0} & \textit{1} \end{bmatrix} \tag{4}$$

iii. Rotate the image 90 degrees. **Remember that the upper left corner is (0, 0) in your coordinate system.**



$T =$

Answer:

$$T = \begin{bmatrix} \mathit{0} & \mathit{-1} & \mathit{200} \\ \mathit{1} & \mathit{0} & \mathit{0} \\ \mathit{0} & \mathit{0} & \mathit{1} \end{bmatrix} \tag{5}$$

(d) [10 pts] 2D Transformations form a nested set of groups that preserve different properties. These transformations can be represented as described in Equations 1 and 2. Fill out the missing blanks in this table. Properties preserved are orientation (O), lengths (L), angles (A), parallelism (P), straight lines (S).

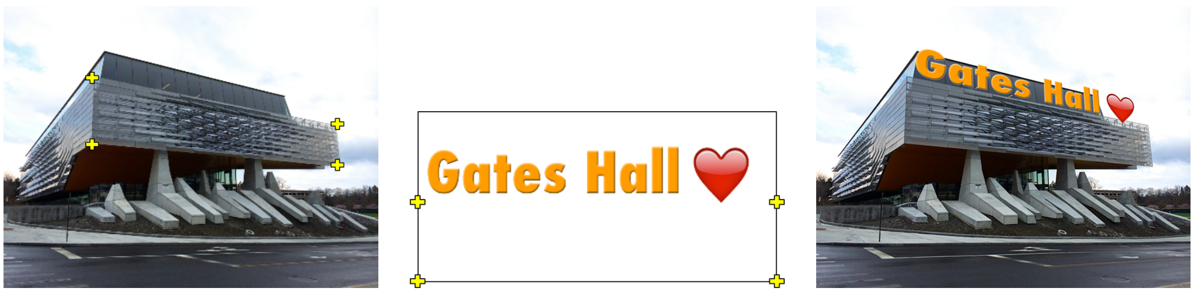
Name of transformation	Degree of freedom	Property preserved
		O + L + A + P + S
		A + P + S
		S
affine		
rigid (Euclidean)		

Answer:

1 point per missing entry.

Name of transformation	*Degree of freedom*	*Property preserved*
translation	*2*	*O + L + A + P + S*
similarity	*4*	*A + P + S*
projective	*8*	*S*
affine	*6*	*P+S*
rigid (Euclidean)	*3*	*L + A + P + S*

(e) [5 pts] Users might want to add 3D overlays to their photographs. For example, given a photo of Gates Hall, for fun they may want to add fun text that is flush with one side of the building.



Briefly describe the steps to take to solve this mapping problem.

i. How many point correspondences do you need between the Gates building image and the Gates banner image?

Answer:

[0.5 pt] 4

ii. Briefly describe the equations you would set up to solve for the mapping.

Answer:

(1 pt) Set up the 4 constraints, with 8 equations to solve for the 8 unknowns.
(1 pt) Formulate A such that $Ax = 0$*.*
(1.5 pt) Describe how you would solve for A; i.e., find the min eigenvalue, and use that to construct the eigenvector that corresponds to the solution.

(f) [2 pts] Given the required number of correspondences with different unique points, you may not be able to solve for the mapping. Explain briefly under what conditions this can happen.

Answer:

[1 pt] If the 4 points are collinear.