CSE 152 - Introduction to Computer Vision

Assignment 1

Due Tuesday, April 27

Question 1 (5 pts)

a) Write down the 3x4 perspective projection matrix in homogeneous coordinates for a pinhole camera. Assume that the camera is at the origin, pointing to the positive z direction, and that the image plane is the plane z=1.

b) Now move the camera to [4 0 0], and rotate the camera so that it points to the negative x direction. Compute and show the new projection matrix.

c) What is the image under this projection of the unit square (sides have length 1) that lies on the $y = -1$ plane and is centered at the origin. Write down the coordinates for each vertex of the square in the image plane and make a sketch of the result.

Question 2 (5 pts)

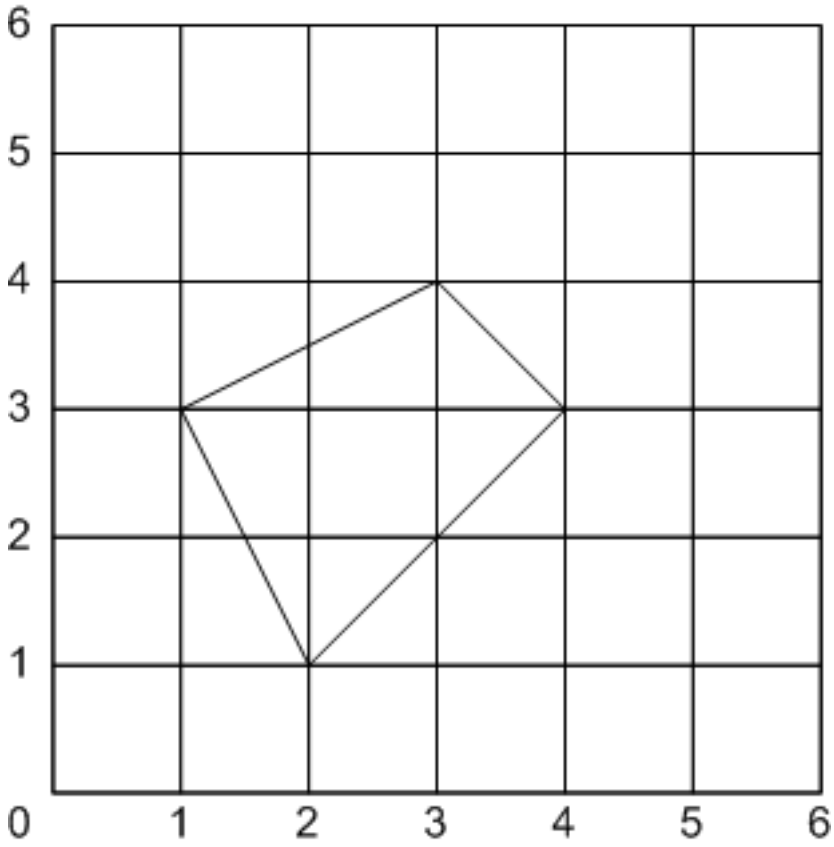
Figure shows the image of a square mapped under perspective projection. Calculate the vanishing point for each pair of parallel lines. Express your answer in homogeneous coordinates system.

Question 3 (5 pts)

Write a Matlab script to convert an image in RGB color space to HSV color space. Do NOT use the built-in `rgb2hsv` function in Matlab. Test your script with the 'color.bmp' image and display each color channel of the result image as an image in grayscale. i.e., display the H image, the S image, and the V image.

Question 4 (15 pts)

a) Implement the connected component algorithm described in class. You can use an 8-connectedness for your program. Do NOT use Matlab's '`bwlabel`' or other related functions (You may use them to compare your result, however). Your program should return a set of (binary) images with each one containing a single segmented object from the input image. Run your program on the 'binary1.bmp' and 'binary2.bmp' image.



b For each region in part A, compute the zeroth, first and second moments. From these, compute the centroid of each object. On each image from part A, draw the centroid as a circle. Then compute the Eigenvectors of the centralized second moment matrix, and draw the two Eigenvectors as rays emanating from the centroid. This should indicate the orientation of the region. m

c) Now design your own method to convert a color input image into a binary image. The input image has 'light' color for the background and 'dark' color for the foreground. A simple method is using a threshold. You may want to use a histogram to decide the threshold level. (Matlab: `imhist`). Test your program on '`diskcase.jpg`' and '`stopsign.jpg`' images.

What to hand in:

- Written part for Question 1 and 2.
- Three images for Question 3 (one for each color channel)

- Output images for Question 4.a and 4.b. Binary images and output segmentation for 4.c.
- A short description (no more than 100 words) for part 4.c.
- Print out of the code for question 3 and 4.
- Turn in during class on Tuesday, Apr 27.
- Submit source code using 'turnin'. For instruction, see the class web page.