

CS376: Computer Vision: Assignment 2

Due: March. 8th, 11:59 PM

Format for writeup: You may use any tool for preparing the assignment write up that you like, as long as it is organized and clear, and figures are embedded in an easy to find way alongside your descriptive text.

Submission: See the end of this document for submission instructions.

Assignment questions: Please see Piazza for questions and discussion from the class.

1 Short answer problems [20 pts]

1. Compare 1) Dilation + Erosion against 2) Erosion + Dilation.
2. List two examples of regular texture and two examples of near-regular texture.
3. What are the cases where optical flow is not well-defined? Please give two concrete examples.
4. What are the advantages of RANSAC when compared with Hough Transform?

2 Circle Detection (50 points)

Implement two circle detectors (one based on Hough Transformation and another based on RANSAC) that takes an input image and a fixed (known) radius, and returns the centers of any detected circles of about that size.

Include two functions with the following form:

$$[\text{centers}] = \text{detectCirclesHT}(\text{im}, \text{radius})$$
$$[\text{centers}] = \text{detectCirclesRANSAC}(\text{im}, \text{radius})$$

where ‘im’ is the input image, ‘radius’ specifies the size of circle we are looking for. Your detector should not exploit the gradient direction. The output centers is an $N \times 2$ matrix in which each row lists the x,y position of a detected circle’s center. Write whatever helper functions are useful.

Then experiment with the basic framework, and in your writeup analyze the following:

- (10 pts) Explain your implementation in concise steps (English, not code).
- (10 pts) Demonstrate the functions applied to the provided images ‘coins.jpg’ and ‘planets.jpg’ and one image of your choosing. Display the images with detected circle(s), labeling the figure with the radius. Note: you only need to select one reasonable radius and display all detected circles (i.e., those with highest votes) under that radius. You are not required to consider circles with a center off the image.

- (5 pts) For Hough Transform, explain how your implementation post-processes the accumulator array to determine automatically how many circles are present.
- (5 pts) For RANSAC, explain how your implement circle fitting.
- (5 pts) For one of the images, display and briefly comment on the Hough space accumulator array.
- (5 pts) For one of the images, demonstrate and explain the impact of the vote space quantization (bin size). In other words, alter the bin size and compare and contrast with a brief explanation why what happened makes sense.
- (10 pts) For one of the images, plot the progress of the RANSAC as the number of tries increase. The x axis of the plot should be the number of tries, and the y axis should be the number of inliers that the best model produces.

Useful Matlab functions: 'hold on'; 'plot', 'fspecial', 'conv2', 'im2double', 'sin', 'cos', 'axis equal'; 'edge', 'impxelinfo'; 'viscircles'

3 Image segmentation with k-means [30 pts]

For this problem you will write code to segment an image into regions using k-means clustering to group pixels.

- (15 pts) Given an $h \times w \times 3$ matrix 'Im', where h and w are the height and width of the image, apply k-means clustering to associate pixels with clusters. Return 'labelIm', an $h \times w$ matrix of integers indicating the cluster membership (e.g., from 1 to k) for each pixel. Please use the following form:

function [labelIm] = clusterPixels(Im, k)

- (10 pts) Detect cluster boundary pixels from 'labelIm'.

function [boundaryIm] = boundaryPixels(labelIm)

- (5 pts) Please test both functions on the provided images 'gumballs.jpg', 'snake.jpg', and 'twins.jpg' and one other image of your choosing, and then displays the results.

Submission instructions:

Create a single zip file so submit on Canvas that includes

- Your well-commented code, including the files and functions named as specified above.
- A pdf writeup of your results with embedded figures where relevant.

Please do not include any saved matrices or images etc. within your zip file.