



Shiraz University

Digital Image Processing(May 2023)

Homework #3

Due Date: 21/03/1402

Applying Huffman coding, Edge detection:

1. Edge detection:

Input images: star_fish.jpg, bell.png, clean_lung_image.jpg

1. Apply canny edge detector(use cv2.canny and edge function in MATLAB) to bell image and starfish image. Tweak the arguments of canny edge detector (threshold in MATLAB and minval and maxval in python) in order to remove undesired edges from the background and highlight edges of starfish and bell.

2. Apply Roberts, Prewitt, Sobel to *clean_lung_image.jpg* and compare all of them with a canny edge detector on the same image. Report the results and explain them.

3. Add Gaussian noise to the image and show the changes in the edge detection due to the existence of this noise.

a. Which one got better results after applying this noise? Why?

- b. What is the existing solution for solving this problem and apply this solution to the noisy image and explain why this solution will work?

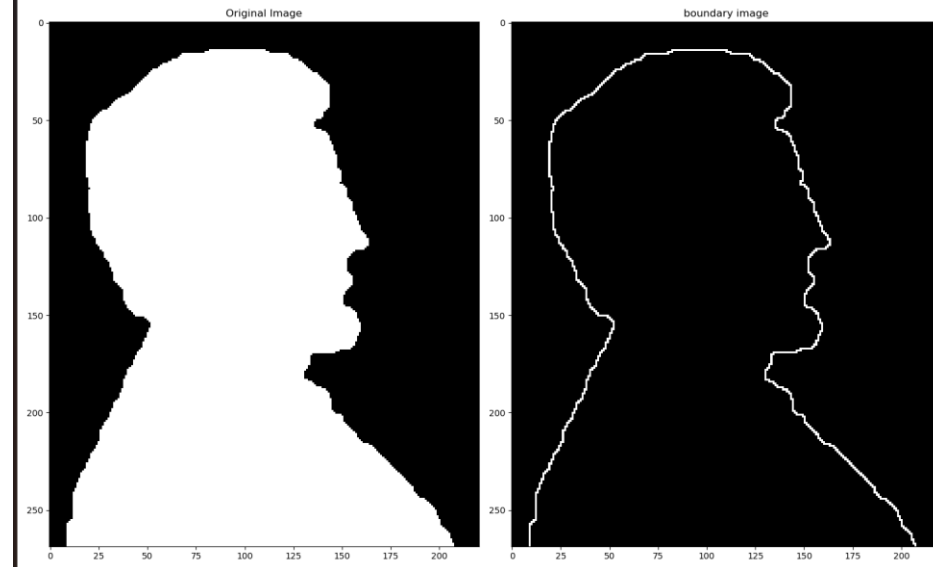
2. Morphological Operations:

Input images: objects.png, man.tif

1. Boundary Extraction

- a. Write a function to performing binary dilation and erosion with an arbitrary (i.e., specified) structuring element of size 3 x 3.
- b. Use your results from Part a to implement morphological boundary extraction

example:



2. Read in the binary image objects.png and write a script which uses the image as input and uses morphological and logical operations to answer the questions below.

- a. How many objects have one or more holes?
- b. How many square objects are in the image?
- c. Identify the square objects that have holes.
- d. Identify the circular objects that have no holes

3. Hough Transform:

1. Line detection (*bldg.png*, *greek.png*):
 - a. Perform line detection on *bldg.png* and *greek.png*. print the rho and theta value for the lines. Use any edge detector that you want and plot the original image, intermediary images and the final result next to each other.
2. Circle detection (*objects.png*)
 - a. Perform circle detection using hough transform on *objects.png*. Use any edge detector that you want and plot the original image, intermediary images and the final result next to each other.
3. (BONUS) Bounding box (*objects.png*)
 - a. Implement a procedure where given an image, detect all the circles in the image using hough transform and draw a bounding box around it.