

EEE-6512 Image Processing and Computer Vision

Fall 2017 Homework #4

October 6, 2017

Due: October 14, 2017, 11:59 PM

This assignment should be completed individually by the student. Late submissions will not be accepted. Proper citation should be provided for any references used.

Part I Textbook Questions [50 points]

Answer the following questions from the textbook:

4.2, 4.3, 4.4, 4.5, 4.6, 4.8, 4.12, 4.15, 4.17(a), and 4.17(b).

Part II MATLAB Programming [50 points]

Please read requirements carefully. Solutions that do not follow provided specifications will not receive credit.

- You are to write a program *convert_binary* that uses the grayscale image *tools.pgm* from HW2 and convert it into a binary image in which the shape of the three separate tools found in the image are as accurate as possible. **You are free to use any built-in functions within MATLAB to accomplish this task.** Keep in mind that preprocessing of the image may be necessary. At the end of your program, you should display the final binary image obtained
- You are to write a function, *color_objects*, that accepts a binary image as input and outputs an RGB image as output in which pixels that belong to the same object have the same color. The function should display the output image. For this function, you are not allowed to use built-in functions. Your algorithm should be based upon the **floodfill** algorithms described in the text. For simplicity, you can assume that your image will use at most ten different colors.

To receive full credit for this assignment, you should submit three files. 1.) A document containing answers to the textbook questions, the final binary image to be used as input to the *convert_binary* program, and the output image of the *color_objects* function (.DOC, .DOCX, or PDF file) 2.) An M-file containing commented MATLAB code for the program *convert_binary*. 3.) An M-file containing commented MATLAB code for the function *color_objects*. Students should ensure that their M-files execute without errors to avoid receiving point deductions.