`http://cs.ucf.edu/~bagci/`

[Programming Assignment] (2)

Robot Vision

Dr. Ulas Bagci ● (Spring) 2018 ● University of Central Florida (UCF)

Coding Standard and General Requirements

Code for all programming assignments should be **well documented**. A working program with no comments will receive **only partial credit**. Documentation entails writing a description of each function/method, class/structure, as well as comments throughout the code to explain the program flow. Programming language for the assignment is **Python**. You can use standard python built-in IDLE, or other IDLEs such as CANOPY, PyCharm Community Edition, PyScripter, CodeSculptor, Eric Python, Eclipse plus PyDev, etc.

Following libraries (or others if you can find in addition to those listed below) can be used throughout the course:

- PIL (The Python Imaging Library), Matplotlib, NumPy, SciPy, LibSVM, OpenCV, VLFeat, python-graph.

If you are asked to implement "Gaussian Filtering", you are not allowed to use a Gaussian function from a known library, you need to implement it from scratch.

Along with your well commented code, please submit a "brief" report as doc or pdf format.

Submit by **16th of February 2018**, 11.59pm.

Canny Edge Detection [10 pt]

Your tasks:

0 pt   Use four different images (picture1.png,...,picture4.png) to perform the following steps for getting Canny edges of the input image.

1 pts   Use 1-dimensional Gaussians to implement 2D Gaussian filtering yourself (do not use built-in functions).

2 pts   Obtain gradient images (x-dim, y-dim, gradient magnitude, and gradient orientation) by following the Canny algorithm that we have seen in the class. Show resulting gradient images on screen and in the report.

3 pts   Implement non-max suppression algorithm to reduce some of the falsely detected edges in the gradient images (from the previous step). Show the improved edge map on the screen and in the report.

2 pts   Implement hysteresis thresholding algorithm and use it to further enhance the edge map obtained from the previous step. Show the final Canny edge map on the screen and in the report.

1 pt   Show the effect of σ in edge detection by choosing three different σ values when smoothing. Note that you need to indicate which σ works best as a comment in your assignment.

1 pt Discuss about the different filtering approaches you took for four pictures. Since pictures are the same scene but different noise and smoothing patterns, you need to adjust your Canny edge filtering parameters to show similar results to Canny edges of the picture1.png.

Bonus: Image Enhancement and Sharpening [1 pt]

Your task in this question is to use picture2, 3, and 4.png for enhancing. Apply potential image sharpening methods such as unsharp masking for improving contrast. Feel free to use histogram equalization if necessary. You may need to try several parameters for a good result. Compare and discuss the findings.