Trinity College Dublin, The University of Dublin
School of Computer Science and Statistics
CVPR Lab, Dublin 2, Ireland

CS7GV1: Computer Vision

Subrahmanyam Murala

Assignment #01

1. Do the following.
   - **a.** Write your own code for computing convolution of the 2D (grayscale) image (template.png) and a 2D filter. Make the output matrix be the same size as the input image. Be careful to correctly deal with the border of the image. The easiest way to do this is to "zero-pad" the image prior to convolution. **(2 Points)**
   - **b.** Extend this code to handle RGB images and 3D filters (having the third dimension equal to 3). **(1 Point)**
   - **c.** Convolve the attached waldo.png with a (2D) Gaussian filter with σ = 1 and visualize the result (display the result of the convolution). You can use built-in functions for convolution. **(1 Point)**

2. Canny edge detector
   - **a.** Compute magnitude of gradients for the attached images waldo.png and template.png. **(1 Point)**
   - **b.** Write a function **MyCannyEdgeDetector(image, low_threshold, high_threshold):** It takes an image along with low and high threshold values as inputs and returns a binary edge-detected image**,** steps involved are as follows: **1.** Gaussian Smoothing, 2. Gradient Calculation, 3. Non-Maximum Suppression, and finally 4. Hysteresis Thresholding. Write small explanation for each step of Canny Edge Detection (in the jupyter notebook). *Note that the Python built-in `edge()` function is not utilized in this implementation.*
   - **c.** You need to calculate the similarity index (SSIM) between inbuilt command output and your custom **MyCannyEdgeDetector()** output. You can use inbuilt SSIM code for similarity index calculation. **(2 Point)**

***** Save the main file of assignment by "*entrynumber_assignment_1.py*" and call all required function in that file.**
***** There will be a Plagiarism check on your Python code.**

Date of Submission: October 11, 2024. Time: 23.59

Submissions after the deadline will not be considered.