

Question 1

Canny Edge Detection

1. Implement your canny edge detection. It should include four steps, Gaussian image blur, sobel filtering, non-maximum suppression, and edge linking. You are allowed to use OpenCV function for the first step (Gaussian image blur). But, you should implement the other three steps by your self instead of calling any OpenCV function. As usual, you can use OpenCV functions of image loading writing, grayscale image converter, thresholding, etc. Convert lenna.png to grayscale to test your implementation.
2. Try different low and high thresholds and put the result images in a report. Shortly write down what the differences are if you use smaller or higher thresholds, and narrow or wider range of low-high thresholds.
3. **Submission:** Submit your code and report (with the images and discussion).

Question 2

Image Pyramid

1. Implement the creation of image pyramid and image reconstruction, which allow creating at least 5 level pyramid. If the resolution of the original image is 512x512, you should down-sample to 32x32. When reconstructing the image to the original resolution, you should only use 32x32 image and error images of all levels only. Convert lenna.png to grayscale to test your implementation.

You are allowed to use the resize function in OpenCV to down-sample and up-sample images. As usual, you are also allowed to use OpenCV functions of image loading writing, grayscale image converter, thresholding, etc.
2. **Submission:** Submit your code and a report. For the pyramid creation, put the 32x32 image and all levels of error-images in your report. For image reconstruction, put all levels of reconstruction image and use OpenCV "resize" function to up-scale all of them to 512x512 (original resolution) in the report (this simulates the progressive image transmission in the old days).
3. **Hint:** tutorial of OpenCV resize function, <https://www.tutorialkart.com/opencv/python/opencv-python-resize-image/>