University of Washington
Department of Electrical Engineering
EE 341, Winter 2017

Report for Lab #3: Introduction to Image Processing

Report by:
Luke Jiang 1560831

1 Group Assignment III

Q1: Describe where you got your image and any modification you needed to make it comply with the size requirement.
A1: I got the image I worked with (cat.jpg) from google. The original image is colored, so I first used rgb2gray() function to convert the original image into grayscale (cat_g.jpg).

I scaled the original image by a factor of 3. Comparing the two scale-down methods introduced in the spec, I found the second one produces a result of better resolution, since the second method uses all the points in the original picture. For a map from a S*S square in the original matrix to a single point in the result matrix, the average of all the points in the square is a better representation of the color in that square than the center point of the matrix. Therefore, I chose the second method for scaling down my image.

 

(result for method 2) (result for method one)

Note: both enlarged to see the detail

2 GROUP ASSIGNMENT IV

Q2 Repeat scaling with your image, using the technique you expect to work best. Comment on the trade-offs of the two technique, and why you chose the one that you used with your image.

A2: The built-in interp2() function works better. As it can be seen along the edge of cat's ear, interpolation results in a smoother curve, where replication contains pixel boxes. The trade-off is that interp2() function gives a larger image than expected, due to the implementation of this function. A line that increases its value linearly would ne the least sensitive region to interpolation, since interpolation would produce a result same to the original image. For the same reason, a line that stays constant in value is least sensitive to replication.

(cat's ear from replication (left) and interpolation (right))

3 GROUP ASSIGNMENT V

Q3: Describe details of what you implemented for the two processed images (a mathematical description, not Matlab details which are uploaded separately) and a discussion of observable effects on your image. The report should include the 3-image concatenation that you generated.

A3: To get a blurred image, convolve the image M with h = (1/9) [1 1 1; 1 1 1; 1 1 1]. Since a square is 3-D counterpart of a 2-D pulse, it has soothing effect that buffers the edges, which makes the convolution result has a lower resolution. To get edge detected image, we need a horizontal edge detector and a vertical one, which have the formula h = [1 2 1; 0 0 0; -1 -2 -1] and h = [-1 0 1; -2 0 2; -1 0 1] respectively. An edge detector will highlight the parts where there is a huge value drop, and zeros out the part that changes less drastically. Thus, M * h1 * h2 will display the edge of the cat.