Aum Amriteshwaryai Namah

19AIE303 - Signal and Image Processing

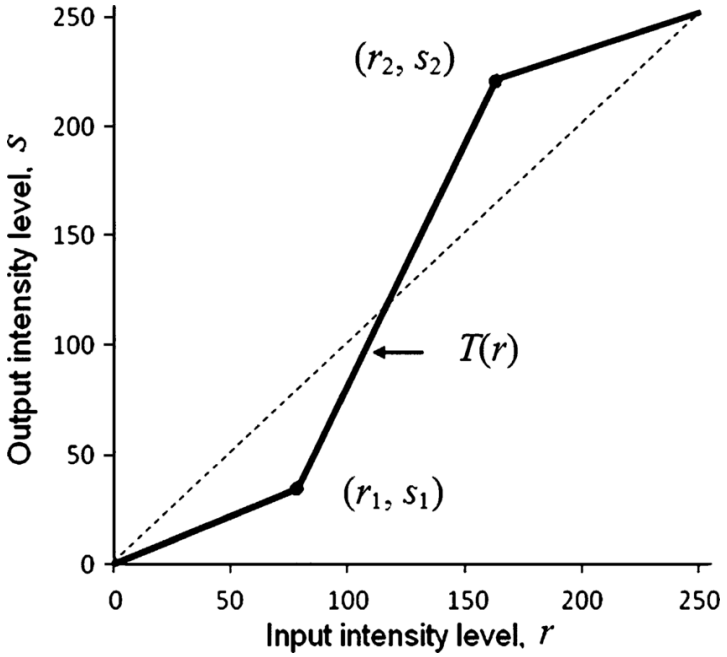
Assignment 4

Q1. Estimate the transformation function required for transforming the image ‘inputq1.jpeg’ to match the image ‘transformed.jpeg’.

a) Plot/Draw the transformation function
b) Apply the transformation function and store the resulting image as ‘outputq1.jpeg’

Q2. Consider the input image: ‘logndlinear.jpg’

a) The general form of the log transformation is s = c log(1 + r). Apply this transformation to the input image such that
   C = 255/(log (1 + m)), where m is the maximum pixel value in the image
   Store the result as ‘logq2.jpg’
b) Apply the following transformation function to the input image



Set the values of r1, s1, r2 and s2 as:
r1 = 70
s1 = 0
r2 = 140
s2 = 255

Q3. Suppose that a 3-bit image (L = 8) of size 64 x 64 pixels (MN = 4096) has the following intensity distribution. Compute the intensity probabilities in the equalised histogram.

r_k	n_k	$p_r(r_k) = n_k / MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

Q4. Consider the input image: ‘lowContrast.png’

a) Plot its histogram and save the plot
b) Perform histogram equalisation and save the equalised image
c) Plot the equalised histogram and save the plot

Q5. Blur the input image ‘building.png’ to three very distinct levels to result in images that look like images ‘blurred_1.jpg’, ‘blurred_2.jpg’ and ‘blurred_3.jpg’.

Q6. Estimate the shading pattern in the image ‘ChessBoardGrad.png’. Store the estimate of shading error as ‘shading.png’ and use this for shading correction. Store the corrected image as ‘corrected.png’

Q7. Consider the input image: 1200px-Monarch_In_May.jpg. Convert this image to grayscale and apply the following transformations:

a) Laplacian
b) Laplacian of Gaussian (gaussian filter of size 3x3)

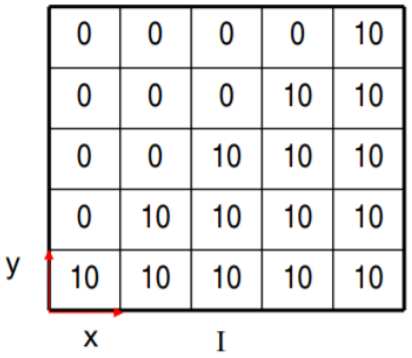
Save and compare the resulting images. Comment on the differences

Q8. Consider the input image: ‘ChessBoardGrad.png’.
Apply the following to the image:

a) Laplacian kernel
b) Sobel kernel in x direction
c) Sobel kernel in y direction
d) Canny edge detection

Save and compare the resulting images. Comment on the differences

Q9. Compute the convolution of image I with the Laplacian kernel. Use border values to extend the image



Apply both these kernels:

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

What to submit:

1. **A single python file** containing **code and comments** for all the questions. Demarcate both questions using comments
2. **Output images and snapshot for Q3 and 9 answers** pasted in a document (word or pdf)

Pls avoid submitting .zip, .tar.gz formats etc