Assignment 1

The purpose of this assignment is to give you some insight into image interpolation, point operations and histogram methods, as well as to give you some practical experience in processing images using Python with OpenCV.

Image Interpolation

1. Using the "cameraman" image as input, rescale the image to ¼ of its original size (1/2 in each dimension) and store the output image as "cameraman_rescaled".
2. Using the "cameraman_rescaled" image as input, rescale the image to 4 times of its size, i.e., back to the original size of "cameraman" using the
    a. Nearest neighbor interpolation
    b. Bilinear interpolation
    c. Bicubic interpolation

    Store the output images as "cameraman_nearest", "cameraman_bilinear" and "cameraman_bicubic", respectively.

Point Operations

Using the image "cameraman" as input:

1. Find the negative of the image and store the output as "cameraman_negative".
2. Apply power-law transformation on the image (experiment with different gamma values) and store the (best) output as "cameraman_power".
3. Apply contrast stretching on the image and store the output as "cameraman_contrast".

Histogram Processing

1. Apply histogram equalization on the "Einstein" image and store the output as "Einstein _equalized".
2. Apply histogram specification on "chest_x-ray1" image so that it matches the histogram of "chest_x-ray2". Store the output as "chest_x-ray3".

1. Deliverables:

Your deliverable should include the following material:
D1 – Complete source code. This should be complete, so that it could be easily rebuilt and executed on a mirror system.
D2 – Resulting output images. Make sure that you include all output images. Also, make sure to follow the given names for the output images. Non-compliance will result in marks deduction.
D3- Documentation, as indicated in "Report Content" below, in pdf format.

2. Report Content:

Write a short report that:

- Provides a brief theoretical overview of each process you implemented.
- Discusses the overall quality of your results. This includes a visual comparison of the quality of the three interpolation methods side-by-side, maybe by plotting difference images or just discussing the noticeable differences.
- The input parameters selection (e.g. the range of gamma values you tested for power-law transformation and why you selected one of those values as the optimal one for the final output).
- Implementation issues that you encountered, if any

3. Marking Scheme

Task	Marks
Downscaling the input image	1
Nearest-neighbor interpolation	2
Bilinear interpolation	1
Bicubic interpolation	1
Negative of the image	1
Power-law transformation	1
Contrast stretching	1
Histogram equalization	2
Histogram specification	3
Report	2
Total	**15**

4. Instructions

- Submit the material to MLS in a single zip file (described below).
- Inside your root directory, make a separate directory for each deliverable.
- Use Python version > 3.6 and OpenCV version > 3.4
- All code that you submit should be original. Do not mine the internet for solutions.

5. Bonus Task

- For bonus points: In addition to visually comparing the quality of each image interpolation output to the original image, quantify your results using the Mean Squared Error (MSE) metric. A description and formula for MSE can be found here.