Digital Image Processing EECS 740

Assignment 1

Lohith Nanuvala

2790468

Abstract:

This assignment is meant to get familiar with programming for image processing. Fundamentals like image filtering and enhancement algorithms were emphasized. Power law transformation, Histogram equalization, Gaussian noise and its removal, Laplacian enhancement technique are some of the concepts demonstrated. I have considered two images, one with underexposure and other with overexpose. The transformations discussed above have been applied on these images and results analysed.

**Power law transformation:**

* Power law transformation is one of the intensity transformation techniques.
* The transformation is given by the equation s=crᵞ where r is the intensity of the input image, ᵞ is the power of the transform, c is a constant, s is the intensity of the output image.
* When ᵞ value is larger, the lower intensities are stretched and vice versa.
* Power law transform is mainly used for contrast enhancement.

**Histogram:**

* Histogram of an image gives the distribution of # of times an intensity occurs.
* Histogram can be useful to understand the over and under exposure in the image, the brightness, contrast and dynamic range.
* Histogram only provides statistical information, it does not give any information about the spatial distribution of intensities, and thus we cannot use histogram to reconstruct an image.
* Matlab syntax: imhist(I), I is the input image.

**Histogram Equalization:**

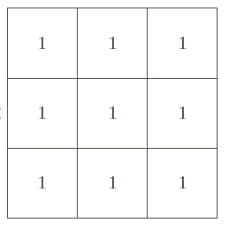
* Histogram equalization is the method of contrast adjustment using the image's histogram.
* We use a monotonic function to transform the input intensities into the output intensities which will be evenly distributed.
* Matlab syntax: histeq(I, N), Where I is the input image, N is the maximum intensity value.

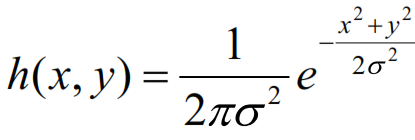
**Gaussian Noise:**

* Gaussian noise isstatistical noise having a probability density function (PDF) equal to that of the normal distribution.
* Principal sources of Gaussian noise in digital images arise during acquisition e.g. sensor noise caused by poor illumination and/or high temperature, and/or transmission e.g. electronic circuit noise.
* Gaussian noise can be reduced using a spatial filter, though when smoothing an image, an undesirable outcome may result in the blurring of fine-scaled image edges.

Filtering using average filter is obtained by convolving filter with the image.

Ex:

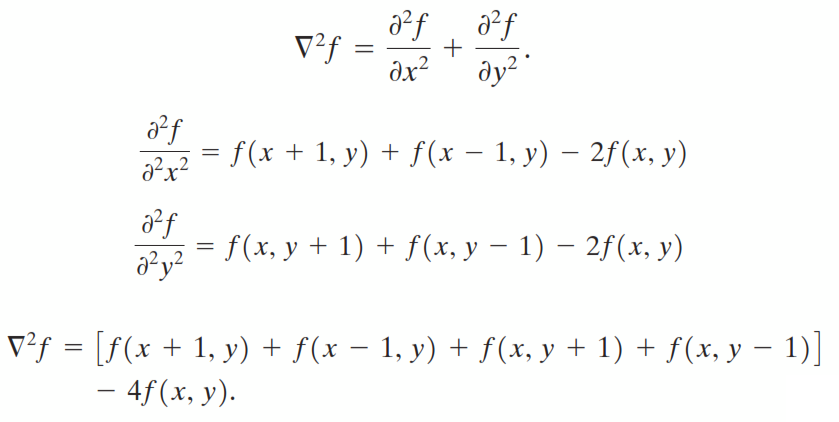
Is a 3x3 averaging filter.

A Gaussian filter is obtained by 

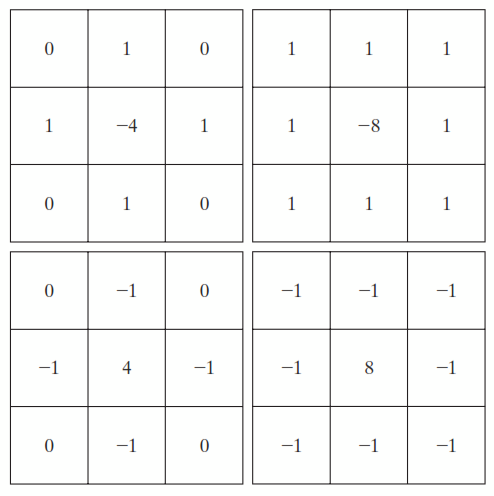
Laplacian Enhancement:

* Laplacian is a derivative operator, which highlights grey-level discontinuities in an image and deemphasizes regions with slowly varying grey levels.
* It is used for image sharpening.

Laplacian derivate operator is given by,



Some laplacian filter masks are,



**RESULTS:**

*Problem 1*: Perform a power-law transformation on the images with r=0.3 and r=3. Show the results and analyse the results.

Problem 2: Compute the histogram of the images, and then apply the histogram equalization on them. Show the histograms before and after equalization and the equalized images.

Problem 3: Add Gaussian noise to the images and use average filter and Gaussian filter to remove the noise. Show and analyse your results under two different noise levels and filter sizes.

Problem 4: Enhance the original images using the Laplacian enhancement technique.

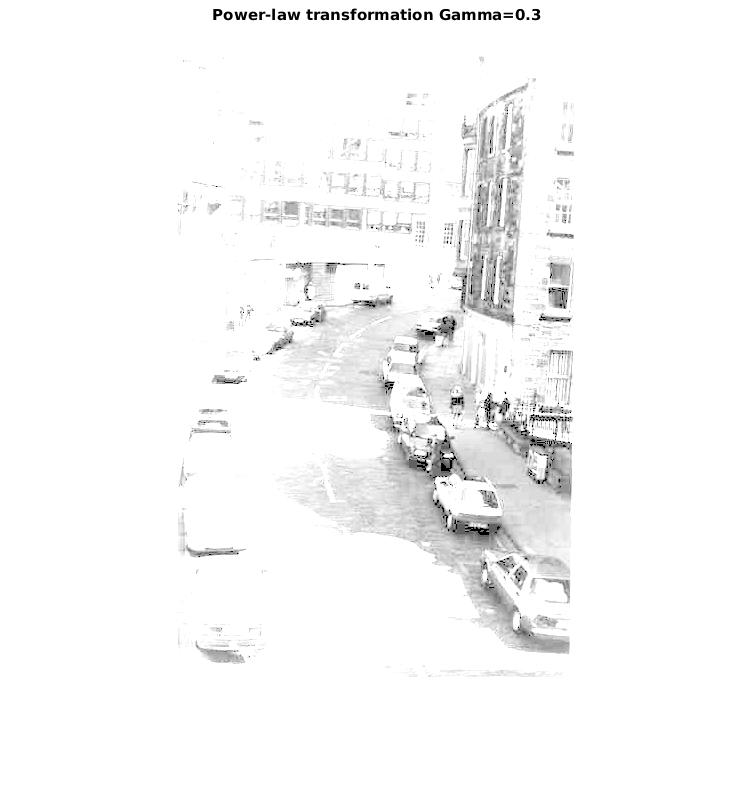
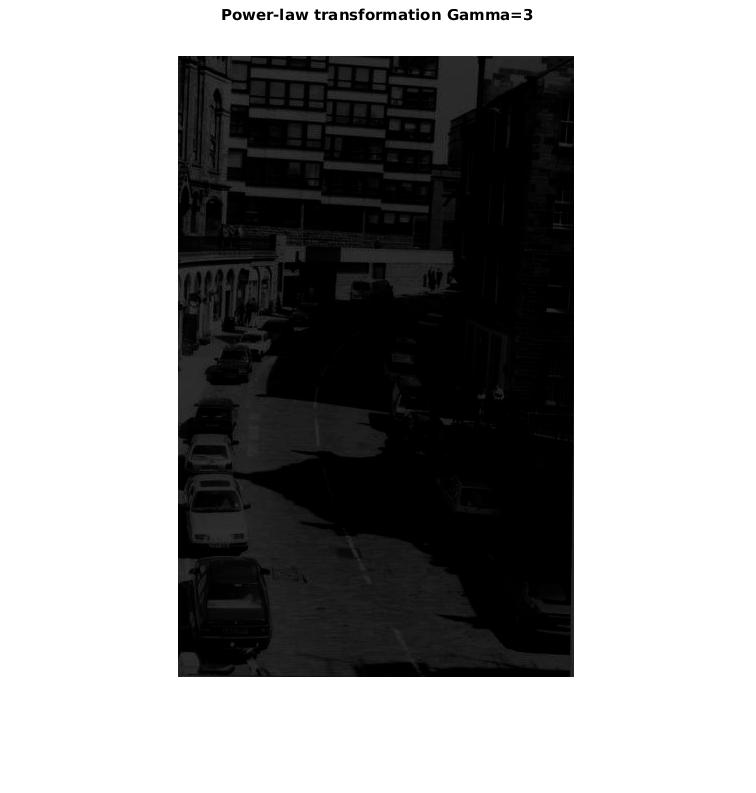
Results:

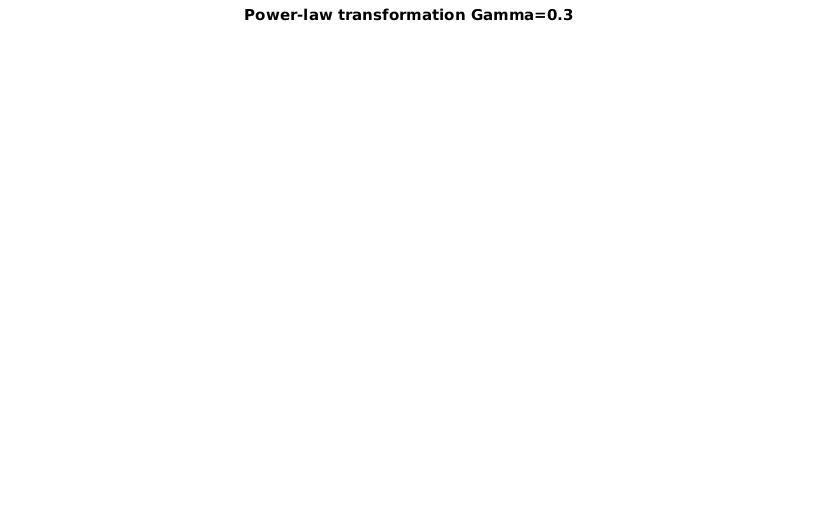
Over exposed image:

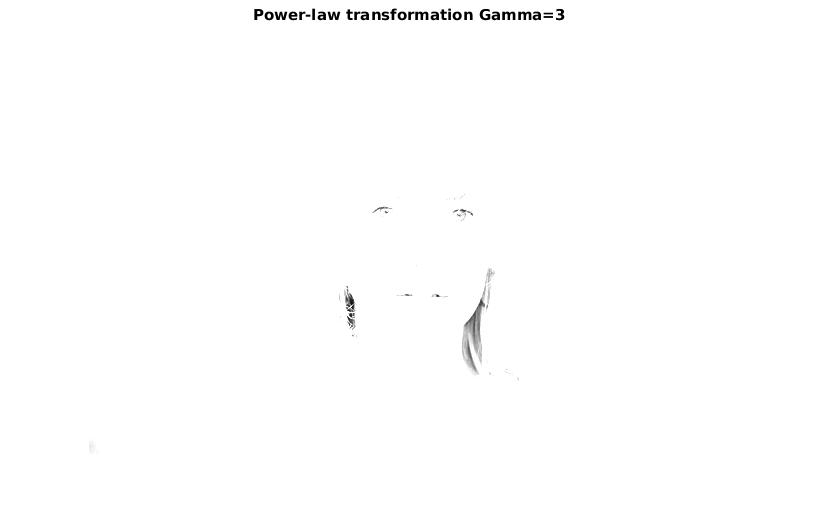


Under exposed image:

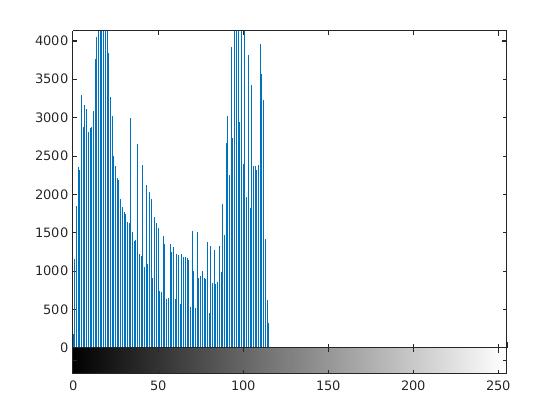




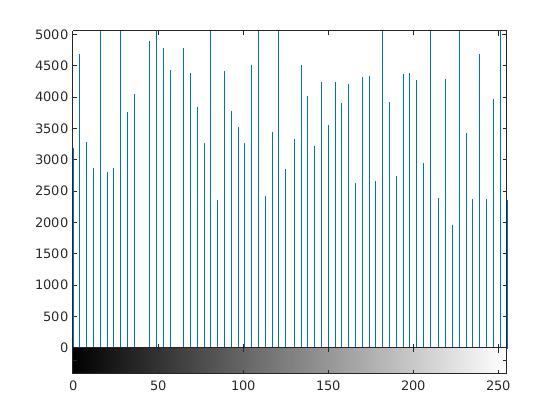




Under exposed image histogram:



Underexposed image with Histogram equalization:



**Appendix:**

Program for problem 1:

% this is for Power-Law Transform

%for gamma value less than 1 we get Bright image

%for gamma value greater than 1 we get Dark image

%RGB=imread('Uexposed.jpg');

RGB=imread('Oexposed.jpg');

% RGB to grey

I=rgb2gray(RGB);

% Changing to double

I=im2double(I);

% computing size m, n

[m n] = size(I);

% Computing s = c \* (r ^ gamma) where r and gamma are positive constants

c = 2;

%gamma values

g1=0.3;

g2=3.0;

for p = 1 : m

for q = 1 : n

I3(p, q) = c \* I(p, q).^g1;

I4(p, q) = c \* I(p, q).^g2;

end

end

figure, imshow(I3);title('Power-law transformation Gamma=0.3');

figure, imshow(I4);title('Power-law transformation Gamma=3');

Program for problem 2:

%reading the under exposed image

I=imread('car1.gif');

% displaying the histogram of the image

imhist(I);

% histogram equalization

J=histeq(I);

% histogram of output image

imhist(J);

figure, imshow(I); title(‘Original under exposed image’);

figure, imshow(J); title(‘Under exposed image after histogram equalization’);

% reading the over exposed image

A=imread('overexposure2.jpg');

% converting color image to grey scale

A1=rgb2gray(A);

% displaying the histogram of the image

imhist(A1);

% histogram equalization

B=histeq(A1);

% histogram of output image

imhist(B);

figure, imshow(A1); title('Original over exposed image');

figure, imshow(B); title('over exposed after histogram equalization');

%end

Program for problem 3:

%reading the over exposed image

I=imread('Oexposed.jpg');

%converting to gray scale image

I1=rgb2gray(I);

% adding Gaussian noise with default mean values

J1 = imnoise(I1,'gaussian');

figure, imshow(J1); title(‘Over exposed image with Gaussian Noise’);

% 3x3 averaging filter

h=ones(3,3)/9;

%figure, imshow(J1);

J2=imfilter(J1,h);

figure, imshow(J2); title(‘Over exposed image after filtering using 3x3 filter’);

% 5x5 averaging filter

h5=ones(5,5)/25;

J3=imfilter(J1,h5);

figure, imshow(J3); title(‘Over exposed image after filtering using 5x5 filter’);

%%%%%%%%%%%%%%%%%

%reading the under exposed image

I=imread('Uexposed.jpg');

%converting to gray scale image

I1=rgb2gray(I);

% adding Gaussian noise with default mean values

J1 = imnoise(I1,'gaussian');

figure, imshow(J1); title(‘Under exposed image with Gaussian Noise’);

% 3x3 averaging filter

h=ones(3,3)/9;

%figure, imshow(J1);

J2=imfilter(J1,h);

figure, imshow(J2); title(‘Under exposed image after filtering using 3x3 filter’);

% 5x5 averaging filter

h5=ones(5,5)/25;

J3=imfilter(J1,h5);

figure, imshow(J3); title(‘Under exposed image after filtering using 5x5 filter’);

%end