

Histogram Equalization & Lighting Correction

Objective:

To implement histogram equalization and lighting correction algorithms and test them on test images

Language used: Python

Running the code (mp03.py):

In a terminal, enter:

```
./mp03.py [image]
```

Example usage:

```
./mp03.py moon.bmp
```

Description of Histogram Equalization Algorithm:

1. Generate a histogram of the input image. This involves 2 steps:
 - a. Calculating the probability mass function (PMF) of all the pixels in the image (includes generating the 'bins' of histogram)
 - b. Calculating the cumulative distributive function (CDF) of the values determined by PMF
2. Carry out mapping of new values obtained using CDF into the original pixel values of the image. This is done using linear interpolation of CDF values.

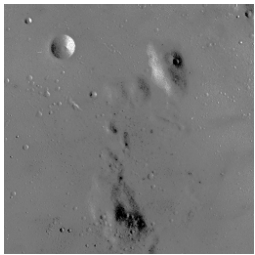

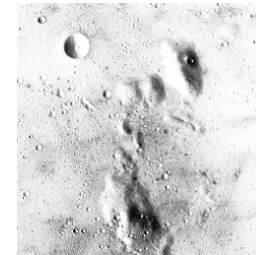
Description of Lighting Correction Algorithm (Log transformation):

1. The lighting correction was carried out using log transformation as follows:

$$Q(i, j) = c \log(|P(i, j)|)$$

Results Analysis:

The following table summarizes the results obtained using two test images:

Image No.	Input Image	Histogram Equalized Image	Lighting Corrected Image
1 (moon)			

The input image produced the desired output image for histogram equalization. However, application of log transformation produced a different output image. This was due to using a different transformation/algorithm than the one carried out on desired image.