Problem Statement

The goal of this assignment is to improve the provided washingdc.jpg image contrast using the histogram equalization.
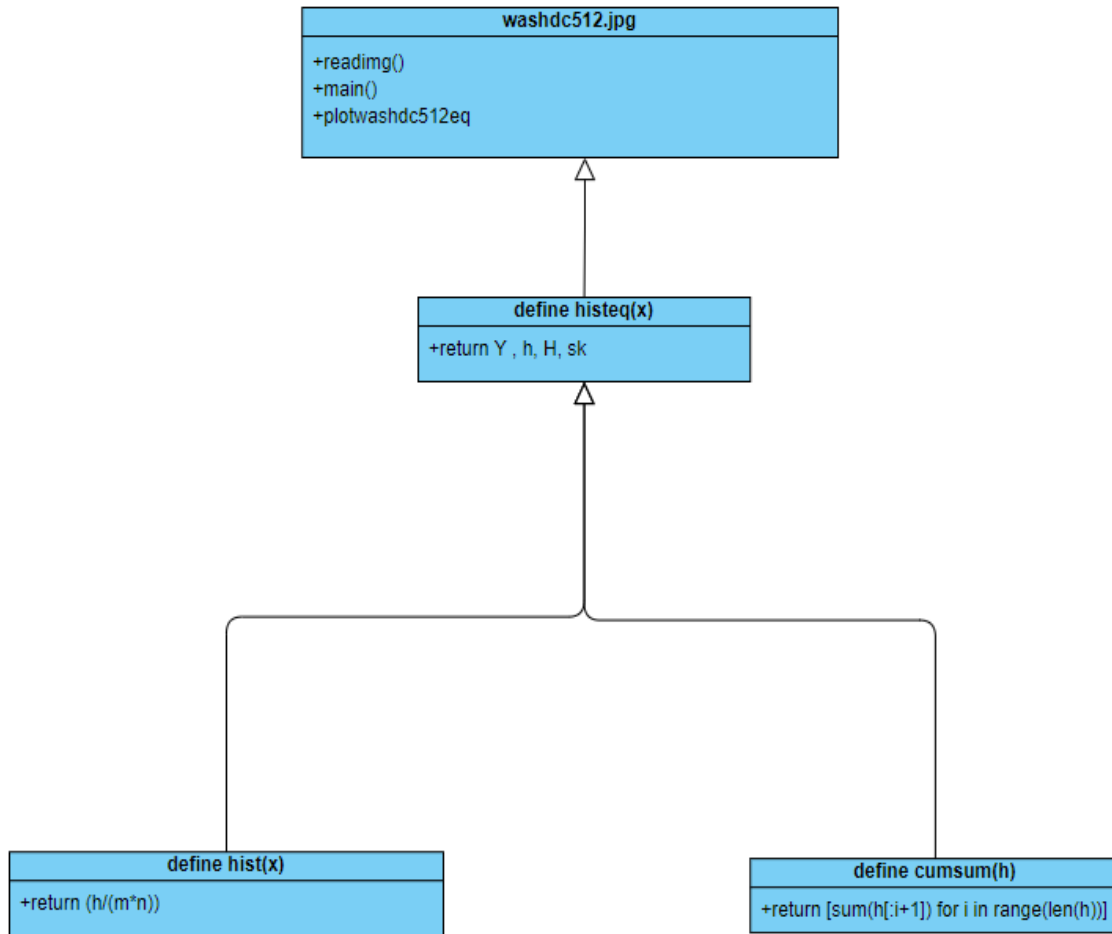
Design Approach

Figure 1: An image showing the design approach with washdc512.jpg as the root class

Code

Problem Statement:

The goal of this assignment is to improve the washingdc.jpg image contrast by histogram equalization

In [1]:
```python
import numpy as np
import matplotlib.pyplot as plt
```

Calculating the normalized histogram of an image

In [2]:
```python
def imhist(im):
    m,n = im.shape
    h = [0,0] * 255
    for i in range(m):
            for j in range(n):
                    h[im[i,j]]+=1
    return np.array(h)/(m*n)
```

Finds cumulative sum of a numpy array, list

In [3]:
```python
def cumsum(h):
    return[sum(h[:i+1]) for i in range(len(h))]
```

Calculate Histogram

In [4]:
```python
def histeq(im):
    h = imhist(im)
    cdf = np.array(cumsum(h))  #cumulative distribution function
    sk = np.uint8(255*cdf) #finding transfer function values
    s1,s2 = im.shape #equivalent m,n
    Y = np.zeros_like(im)
    # applying transferred values for each pixels
    for i in range(0,s1):
            for j in range(0,s2):
                    Y[i,j] = sk[im[i,j]]
    H = imhist(Y)

    return Y,h,H,sk
```

Main

In [5]:
```python
def main():
    img = np.uint8(plt.imread('washdc512.jpg'))
    newimg,h,newh,sk = histeq(img)

    # Show Old and New Image

    # Show Original Image
    plt.subplot(121)
    plt.imshow(img)
    plt.title('Original Image')
    plt.set_cmap('gray')

    #Show New Image
    plt.subplot(122)
    plt.imshow(newimg)
    plt.title('Histogram Equilization Image')
    plt.set_cmap('gray')
    plt.show()
    plt.imsave('washdc512histeqd.jpg', newimg, cmap = 'gray')

    #Plot histograms and transfer function


    fig = plt.figure()

     #Original Histogram
    fig.add_subplot(221)
    plt.plot(h)
    plt.title('Original Histogram')

    #Histogram of Equilized Image
    fig.add_subplot(222)
    plt.plot(newh)
    plt.title('New Histogram')


    plt.show()

    #Transfer Function
    plt.plot(sk)
    plt.title('Transfer Function')
    plt.show()
```
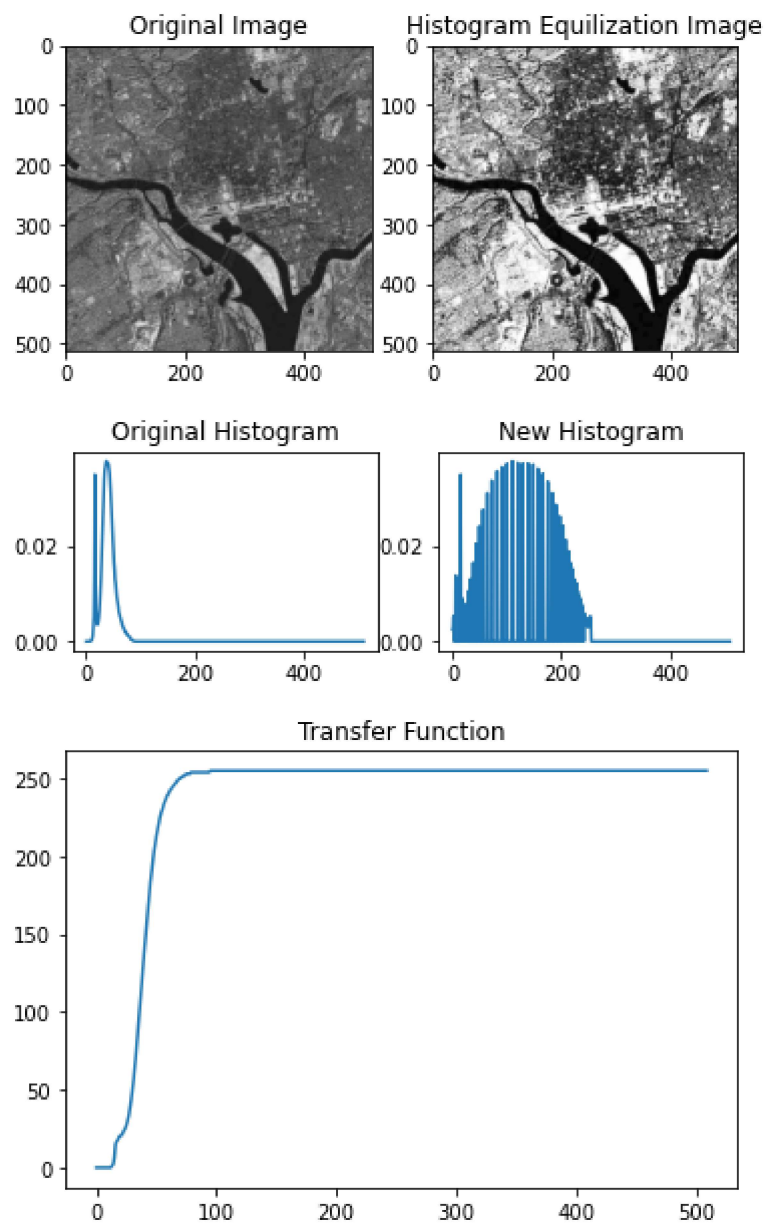
Results

In [6]:
```python
if __name__=='__main__':
    main()
```

Concluding Remarks

After performing the histogram equalization, we get the following results:

Figure 2: image on the left showing before performing histogram equalization process whereas the image on right shows after performing histogram equalization process.

After performing the histogram equalization process we can see that the image on the right has a higher contrast and is brighter and we are able to see things more clearly, where as the image on the left has intensity values and is therefore darker.