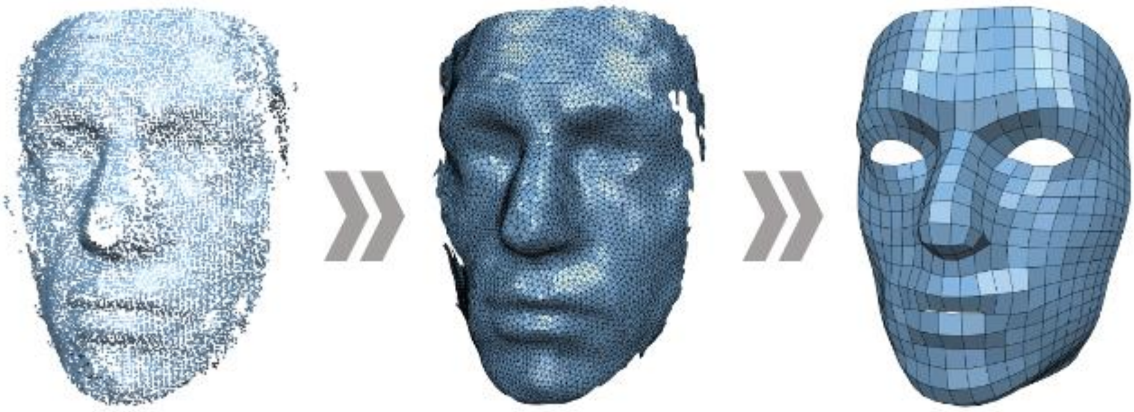
POKHARA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

APEX COLLEGE

MID-BANESHWOR, KATHMANDU



COMPUTER GRAPHICS AND IMAGE PROCESSING

REPORT: 3

SUBMITTED TO:

Abhinash Khanal

Bijaya Khadka

SUBMITTER: Piyush Manandhar

Signature

BCIS MAXTHON

INTRODUCTION

In this lab, we are introduced to various point operations on images and how these operations can change an unreadable image to an understandable one. Various point operations are used on various images based on their properties. Similarly, we will also learn use our own images and do the point operations on them to see the results for ourselves.

OBJECTIVES

- To study about Point Operations on Images.
- Be familiar with the principal techniques used for intensity transformations.
- Understand the physical meaning of image histograms and how they can be manipulated for image enhancement.
- Understand the mechanics of spatial filtering, and how spatial filters are formed.
- To study about intensity transformation and its types.

THEORY

A point operation takes a single input image into a single output image in such a way that each output pixel's gray level depends only upon the gray level of the corresponding input pixel.

1. Intensity Resolution:

Intensity of resolution means the number of pixels per square inch, which determines the clarity or sharpness of an image. It is a smallest discernible change in intensity level. It is an integer power of two (8 bits, 16 bits). Number of gray levels are the example for intensity resolution.

2. Spatial Resolution:

Spatial resolution is a term that refers to the number of pixels utilized in construction of a digital image. Images having higher spatial resolution are composed with a greater number of pixels than those of lower spatial resolution. For example: number of rows and columns of an image. (128 x 128, 256 x 256)

- **Spatial Domain Processes**:

  Spatial domain processes can be described using the equation: **g (x, y) = T [f (x, y)]** where **f (x, y)** is the input image, **T** is an operator on **f** defined over a neighborhood of the point **(x, y)** and **g (x, y)** is the output.

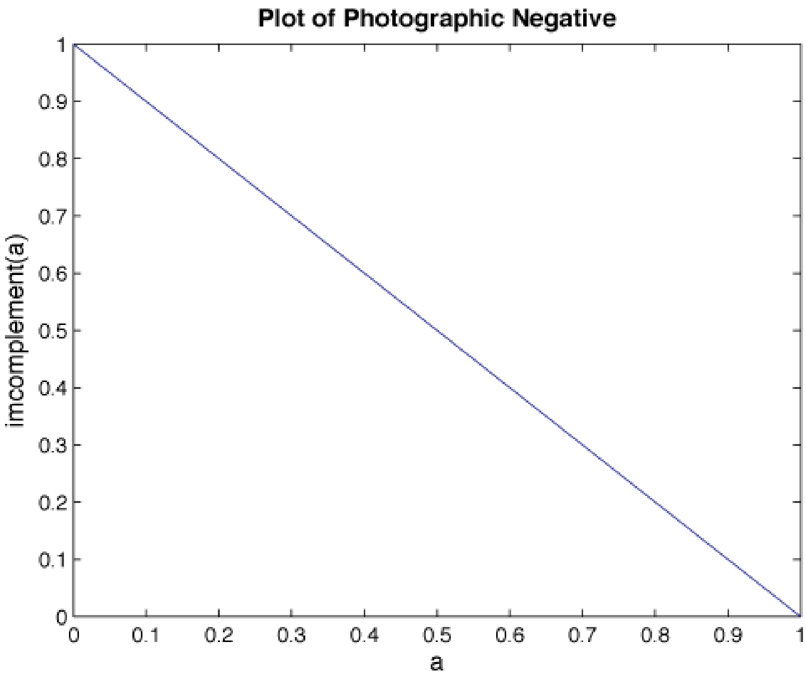
3. Intensity Transformation:

Intensity transformations are among the simplest of all image processing techniques. Some of the common types of intensity transformations are:

i. **Image Negative:**

The negative of an image with intensity levels in the range [0, L-1], where L=256 (generally) is obtained by using the negative transformation function.

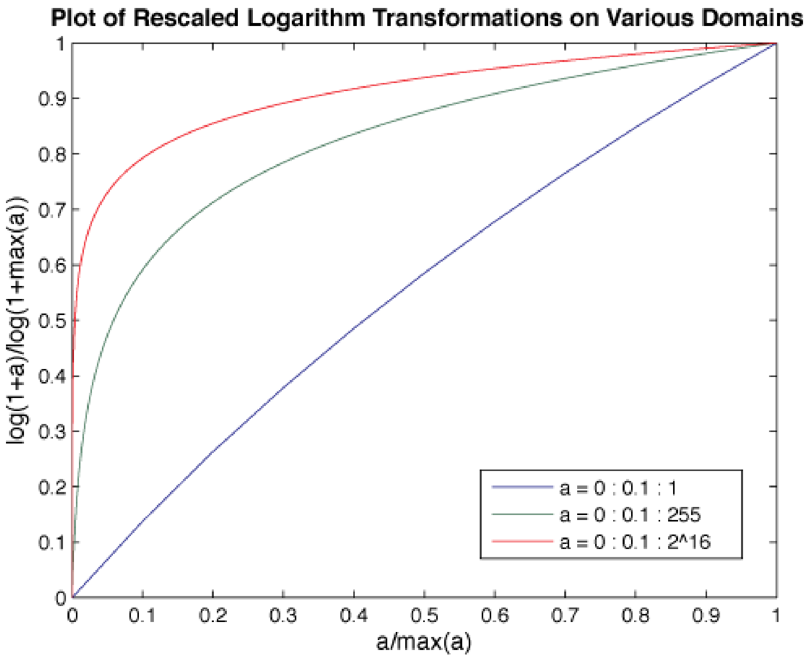
Then, the negative transformation can be described by the expression $s = L-1-r$ where r is the initial intensity level and s is the final intensity level of a pixel. This produces a photographic negative.



ii. **Log Transform**

The dynamic range of an image can be compressed by replacing each pixel value with its logarithm. This has the effect that low intensity pixel values are enhanced.

Mathematically, log transformations can be expressed as $s = c\log(1+r)$. Here, s is the output intensity, $r>=0$ is the input intensity of the pixel, and ***c*** is a scaling constant. ***c*** is given by ***255/ (log (1 + m))***, where m is the maximum pixel value in the image. It is done to ensure that the final pixel value does not exceed (L-1), or 255.



iii. Thresholding

Thresholding is a useful technique which can be used to separate out the regions of the regions of the image corresponding to the subjects in which we are interested, from those regions in the background. It provides an easy and convenient way to perform this segmentation on the basis of the different intensities or colors in the foreground and background.

Mathematically, thresholding can be achieved by:

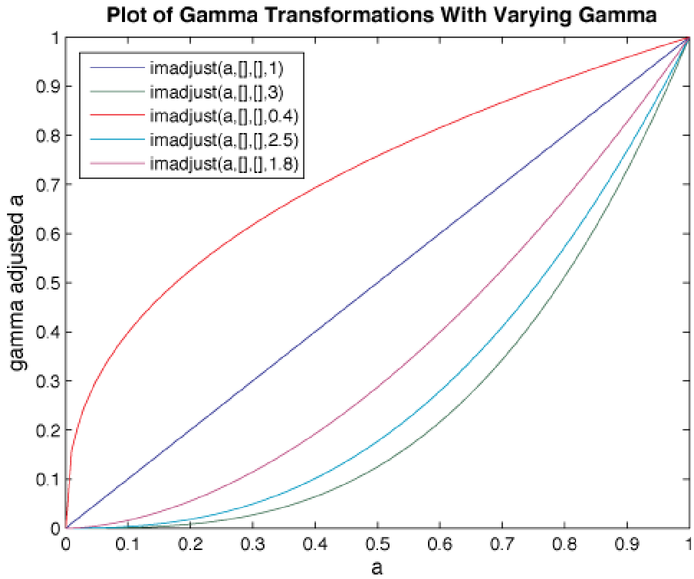
$g\ (x, y) = 1$ if $f\ (x, y) > T$ AND, $g\ (x, y) = 1$ if $f\ (x, y) <= T$

where, 1 is the object and 0 is the background.

iv. **Gamma Transform**

With Gamma Transformations, you can curve the grayscale components either to brighten the intensity (when gamma < 1) or darken the intensity (when gamma >1). Gamma correction is important for displaying images on a screen correctly, to prevent bleaching or darkening of images when viewed from different types of monitors with different display settings. This is done because our eyes perceive images in a gamma-shaped curve, whereas cameras capture images in a linear fashion.

Mathematically, it is expressed as $s = cr^{\gamma}$, where **r** is input pixel intensity, **c** is arbitrary constant and **s** is the output pixel intensity.



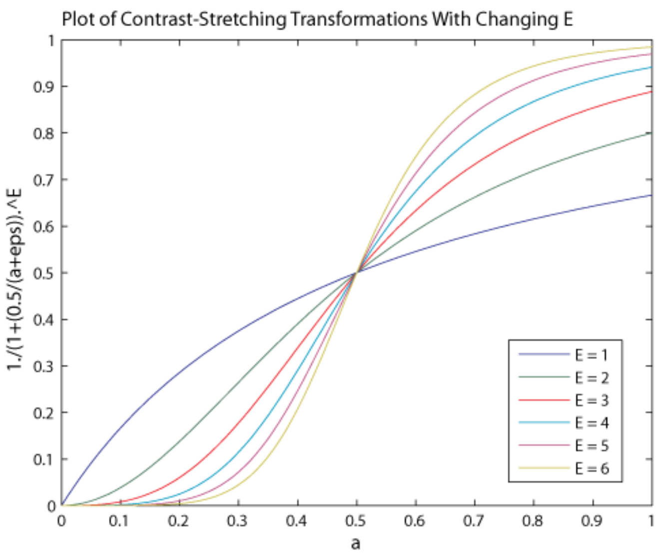
v. Contrast Stretching

Contrast stretching (often called normalization) is a simple image enhancement technique that attempts to improve the contrast in an image by `stretching' the range of intensity values it contains to span a desired range of values. Contrast-stretching transformations increase the contrast between the darks and the lights.

Mathematically, contrast stretching can be expressed by

$s = (255/(H-L)) * (r-L)$

where **H** is the highest intensity of input image, **L** is the lowest intensity of image, **r** and **s** are input and output pixel intensities respectively.

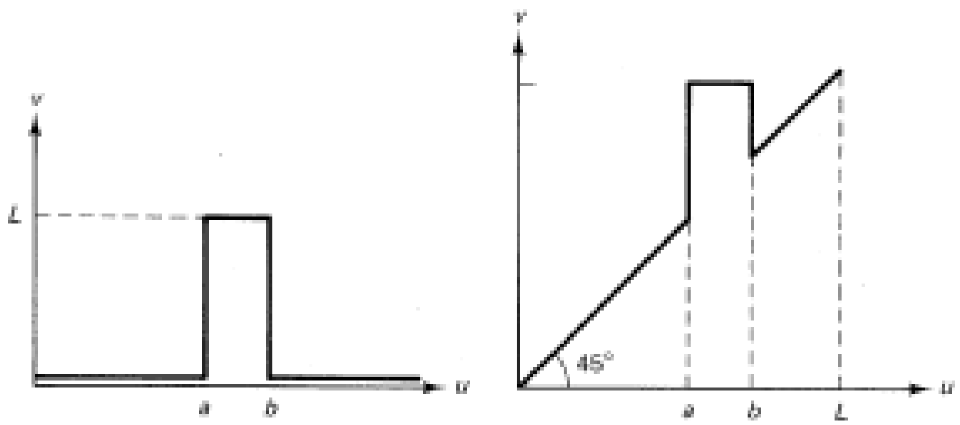


vi. **Intensity-level Slicing**

Intensity level slicing means highlighting a specific range of intensities in an image. In other words, we segment certain gray level regions from the rest of the image. There are two different themes this process can be applied.

We could highlight our desired range of intensities and set those pixels to white and everything else to black. This way we would produce a binary image.

Another option is we highlight our range of intensities and leave everything else unchanged.

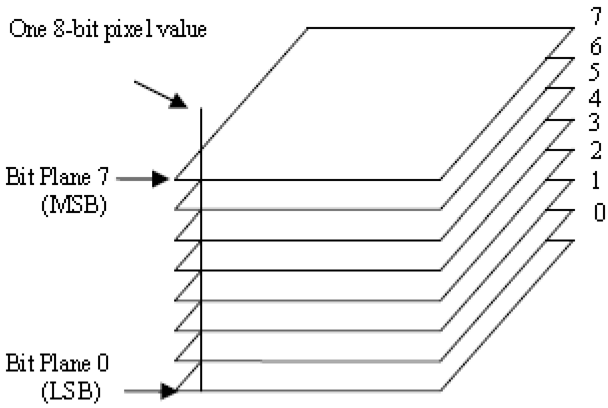


Graph above shows intensities; horizontal axis shows for input and vertical for output. “a” and “b” represent the lowest and the highest intensity levels to be highlighted. Or in other words, they define the position and width of the intensity level band to be highlighted.

Left graph shows the first method, where we produce a binary image. As for the right graph we can see that only a certain range of intensities is changed in the output, while other intensity levels stay unchanged.

vii. Bit-plane Slicing

Pixel values are integers composed of bits. For example, values in a 256-level gray-scale image are composed of 8 bits (one byte). Instead of highlighting intensity-level ranges, we could highlight the contribution made to total image appearance by specific bits. This method is called bit-plane slicing.



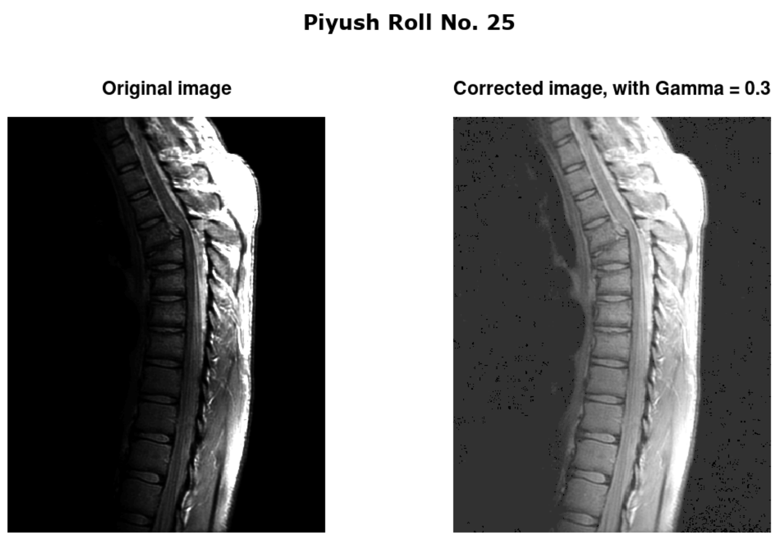
COMMANDS/ SYNTAX

1. **imcomplement (I):** Make negative image.
2. i**madjust (I, [], [], G):** Make gamma valued adjusted image.
3. **g = c * log (1 + double(f)):** Generate log adjusted image
4. **g = 1. / (1 + (m./(double(f) + eps)). ^ E):** Make contrast-stretched transformative image.

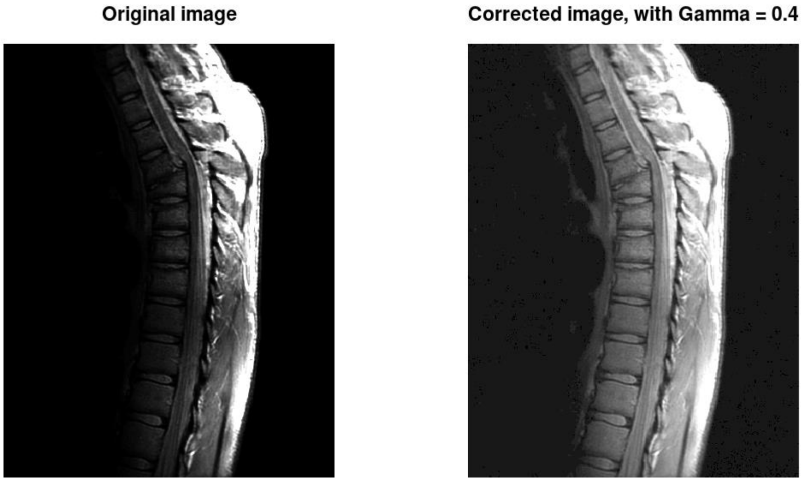
RUNNING THE LAB

1. Check the output with different gamma values: 0.4, 0.3, 0.5 in ‘spine.jpg’ and check output with different gamma values: 4, 5, 6 in ‘runway.jpg’ and discuss the output.

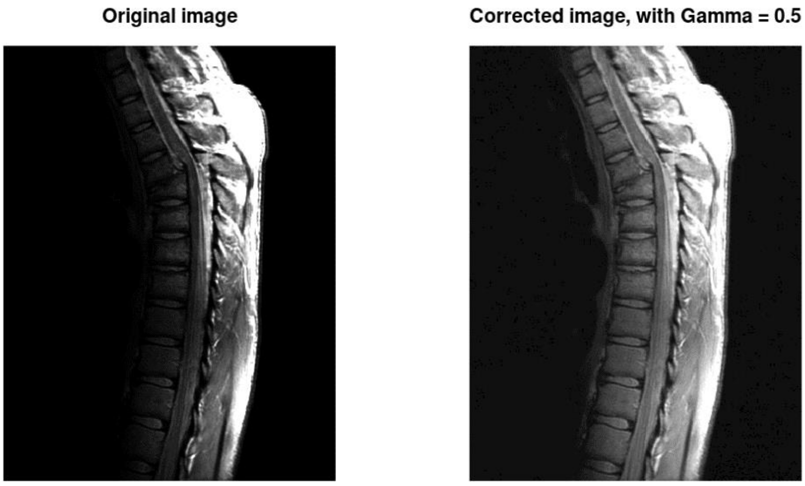
```
##using different gamma values
img = imread('spine.jpg');
img = double(img);
gammaValue = input('Enter the Gamma value, gamma = ');
correction = 255 * (img/255).^gammaValue;
subplot(121),imshow(img, []), title('Original image');
subplot(122),imshow(correction, []), title(['Corrected
image, with Gamma = ',num2str(gammaValue)]);
```



Piyush Roll No. 25



Piyush Roll No. 25

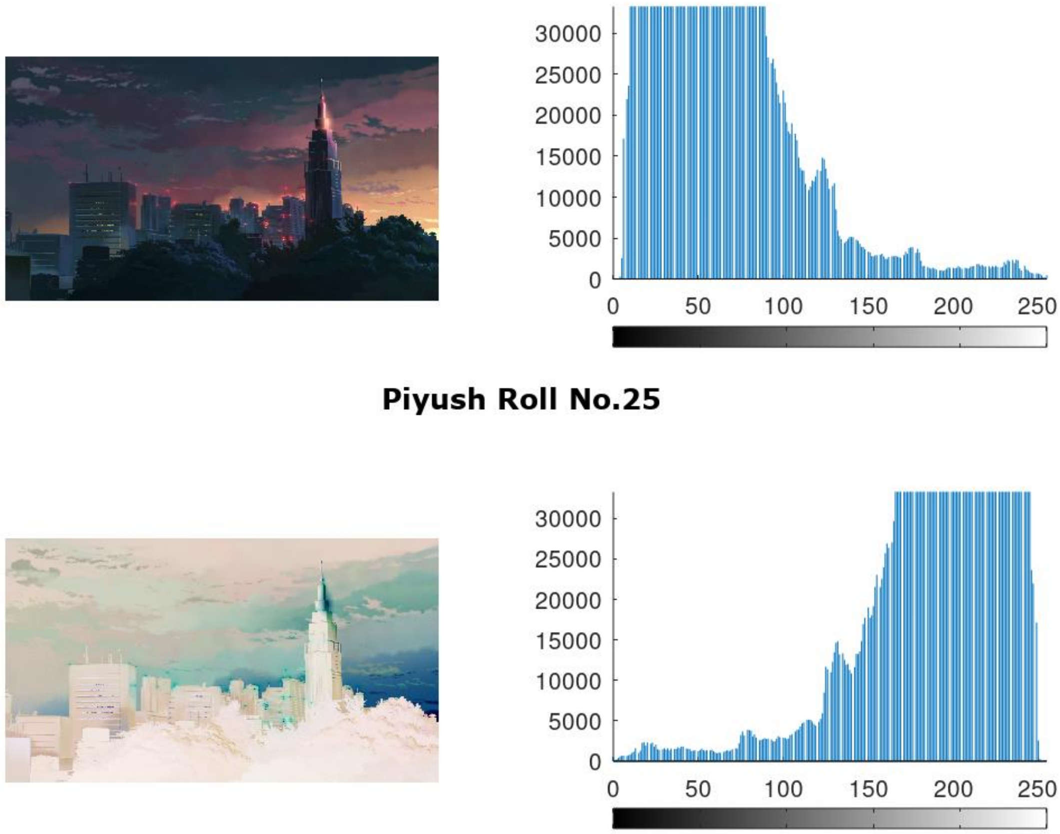


```
##constrast stretching
im = rgb2gray(imread('demo.png'));
[row col] = size(im);
H = max(im(:))
H = prctile(im(:),95)
L = min(im(:))
L = prctile(im(:),1)
m = 255/(H-L);
res = m * (im - L);
subplot(121),imshow(im,[])
subplot(122),imshow(res,[])
o_Max = max(res(:))
o_Min = min(res(:))
```

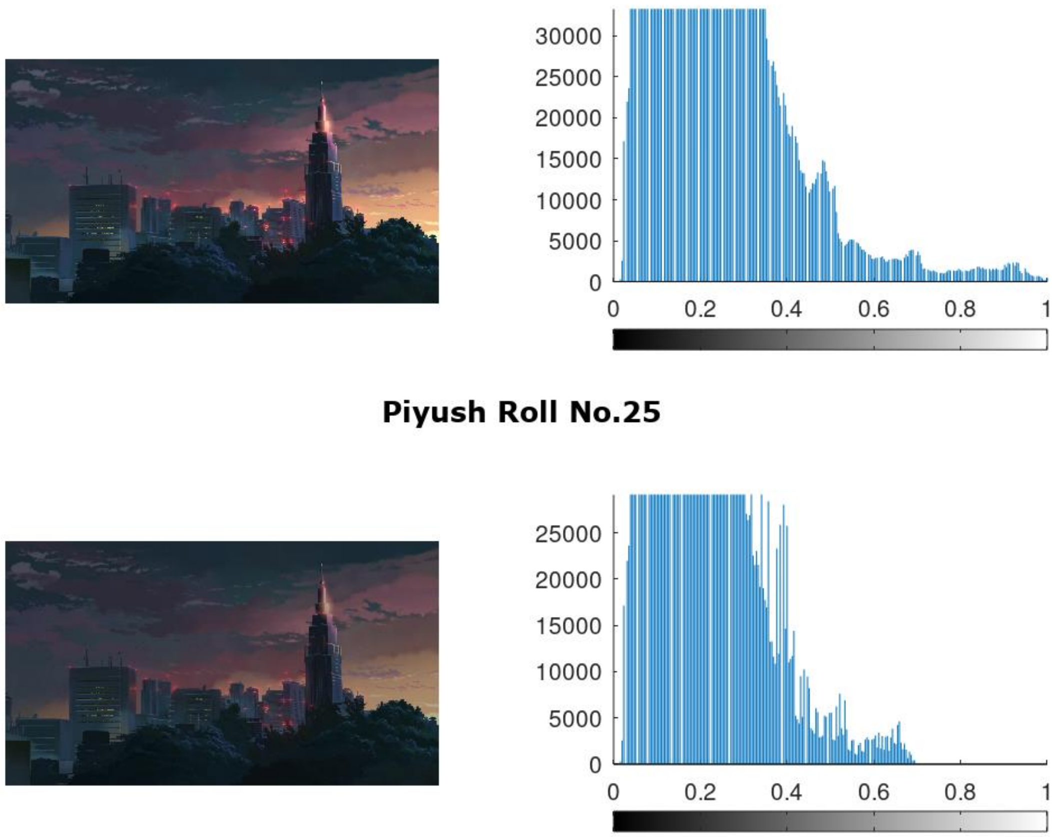
Piyush Roll No.25

2. Perform all mentioned intensity transformation using appropriate images for each operation.

```
##Negative Image
img = imread('demo.png');
result = 255 - img;
subplot(221),imshow(img)
subplot(222),imhist(img)
subplot(223),imshow(result)
subplot(224),imhist(result))
```

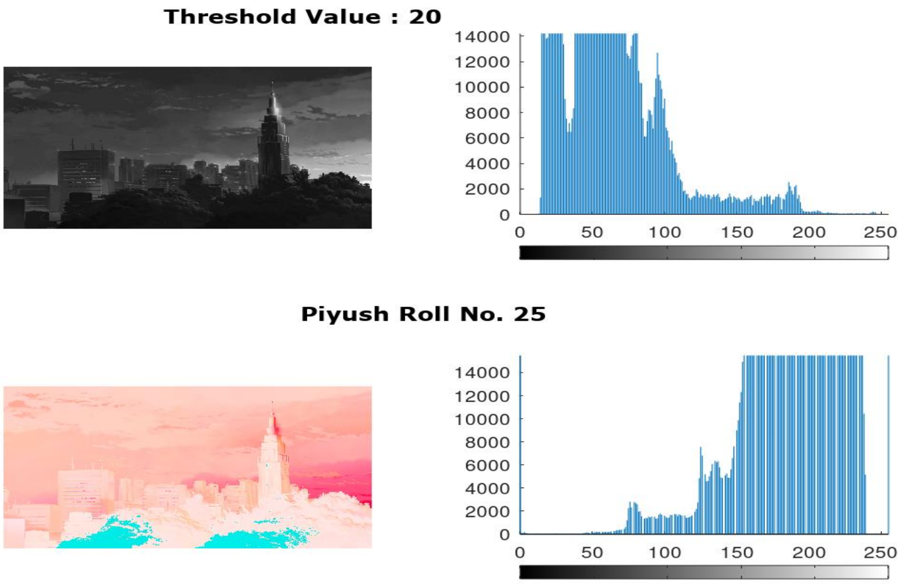


```
##Log Transform
I=imread('demo.png');
I2=im2double(I);
J=1*log(1+I2);
subplot(221),imshow(I2)
subplot(222),imhist(I2)
subplot(223),imshow(J)
subplot(224),imhist(J)
```



```
##Gamma Transform
img = imread('demo.png');
img1 = double(img);
gammaValue = input('Enter the Gamma value, gamma = ');
correction = 255 * (img1/255).^gammaValue;
subplot(121),imshow(img, []), title('Original image');
subplot(122),imshow(correction, []), title(['Corrected image, with Gamma = ',num2str(gammaValue)]);
```

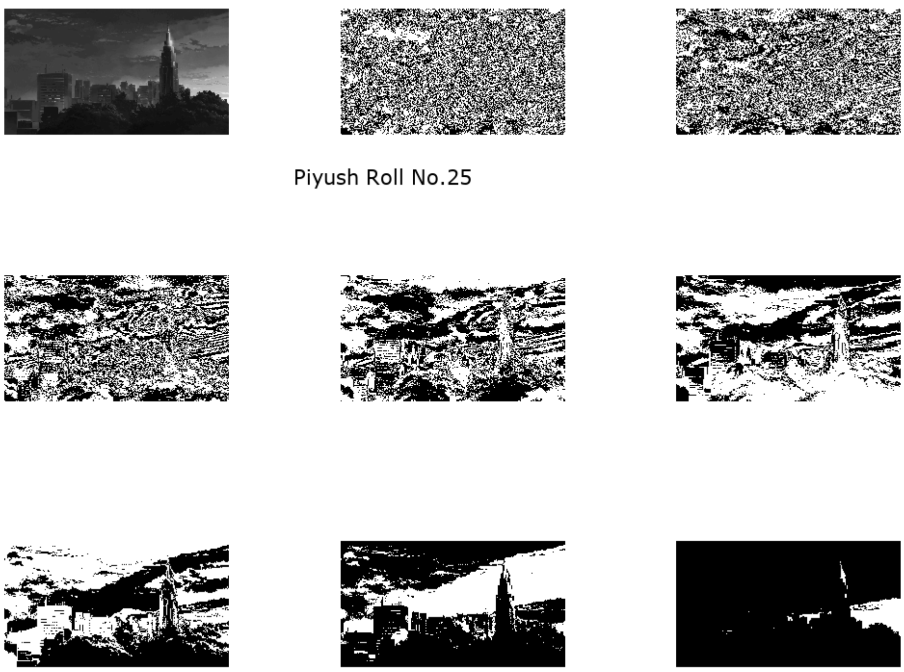
```
##Thresholding
img = rgb2gray(imread('demo.png'));
[row col] = size(img);
t = input('Enter the thresholding value ');
for x = 1:1:row
        for y = 1:1:col
                if img(x,y) >= t
                        result(x,y)= 255;
                else
                        result(x,y) = 0;
                end
        end
end
subplot(221),imshow(img)
subplot(222),imhist(img)
subplot(223),imshow(result,[])
subplot(224),imhist(result)
```



```
##Contrast Stretching
im = rgb2gray(imread('constr.jpg'));
[row col] = size(im);
%H = max(im(:))
H = prctile(im(:),95)
%L = min(im(:))
L = prctile(im(:),1)
m = 255/(H-L);
res = m * (im - L);
subplot(121),imshow(im,[])
subplot(122),imshow(res,[])
o_Max = max(res(:))
o_Min = min(res(:))
```

Piyush Roll No.25

```
##Bit-plane slicing
img = imread('demo.png');
[row col test] = size(img);
if test > 2
img = rgb2gray(img);
end
subplot(3,3,1),imshow(img);
img = double(img);
bl = cell(7,1);
for i= 0 : 7
bl{i+1} = bitget(img,i+1);
subplot(3,3,i+2),imshow(logical(bl{i+1}));
end
```



```
##Intensity-level Slicing
itemp = imread('demo.png');
image = itemp(:,:,1);
rmin = 100;
rmax = 180;
[r,c]= size(image);
s = zeros(r,c);
for i = 1:r
 for j = 1:c
 if (rmin < image(i,j)&& image(i,j) < rmax)
 s(i,j) = 0;
 else
  s(i,j) = image(i,j);
 end
 end
end
subplot(121),imshow(uint8(image))
subplot(122),imshow(uint8(s))
```

Piyush Roll No.25

DISCUSSION AND CONCLUSION

From this lab, we come to learn that using various operations to transform and enhance images. Different operations have different areas of application. Image negative transformation provides a photographic negative. Log transformation is used for image enhancement as it expands dark pixels of the image as compared to higher pixel values. Gamma transformation is used in MRIs. Thresholding provides an easy and convenient way to perform this segmentation on the basis of the different intensities or colors in the foreground and background.