



UFOP

Universidade Federal
de Ouro Preto

Lista 03

**Aluno em Graduação da Universidade
Federal de Ouro Preto do curso Ciência da**

Computação:

Halliday Gauss Costa dos Santos.

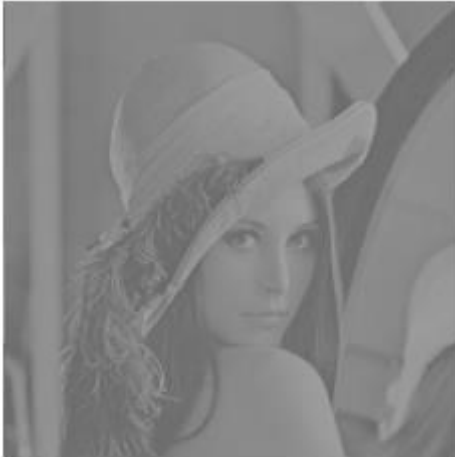
Matrícula: 18.1.4093.

Área: Processamento de Imagens.

Questão 1:

a) `imadjust()`

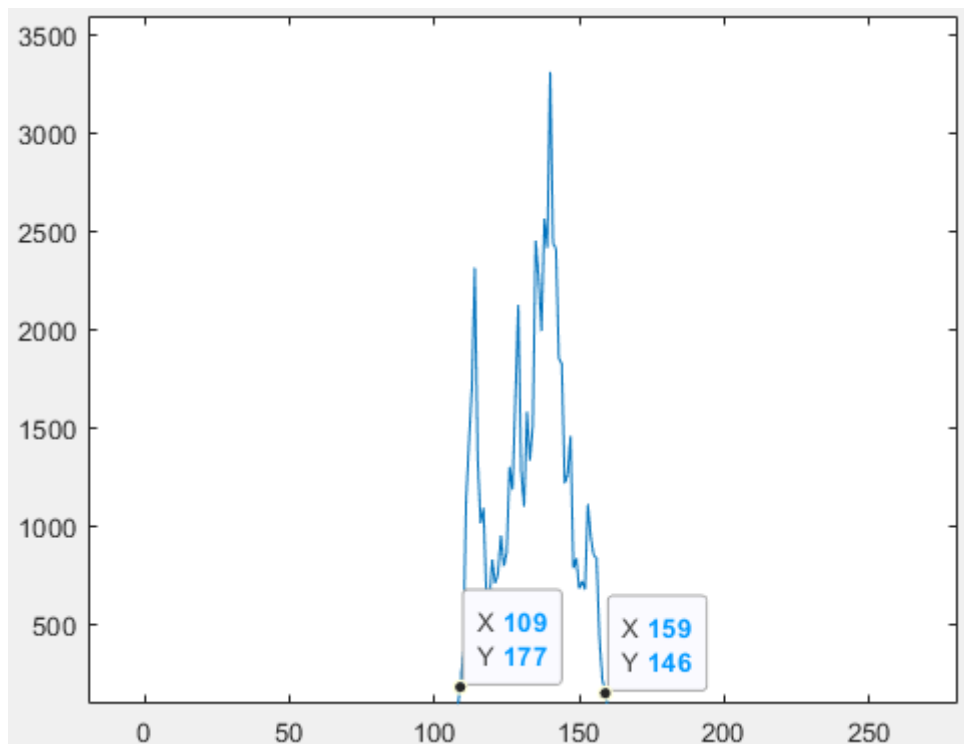
Dada a imagem:



E utilizando os seguintes comandos:

```
>> img = imread("C:\Users\halli\Desktop\7Periodo\PDI\Listas\Lista3\lennaCompress.png");  
>> h = imhist(img);  
>> figure; plot(h);
```

É possível gerar o histograma abaixo:



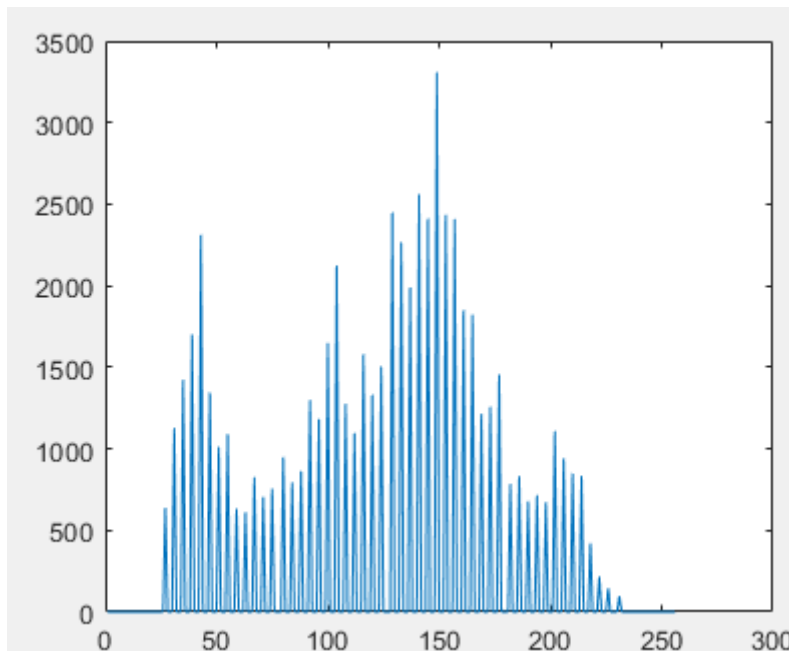
E através desse histograma é possível observar que a maioria das intensidades da imagem ficam entre 109 e 159. E o utilizando os seguintes comandos:

```
>> nimg = imadjust(img, [109/255 159/255], [0.1 0.9]);  
>> imshow(nimg);
```

É possível gerar uma nova imagem distribuindo as intensidades de maneira uniforme melhorando o contraste. E a nova imagem gerada será:



```
>> plot(imhist(nimg));
```



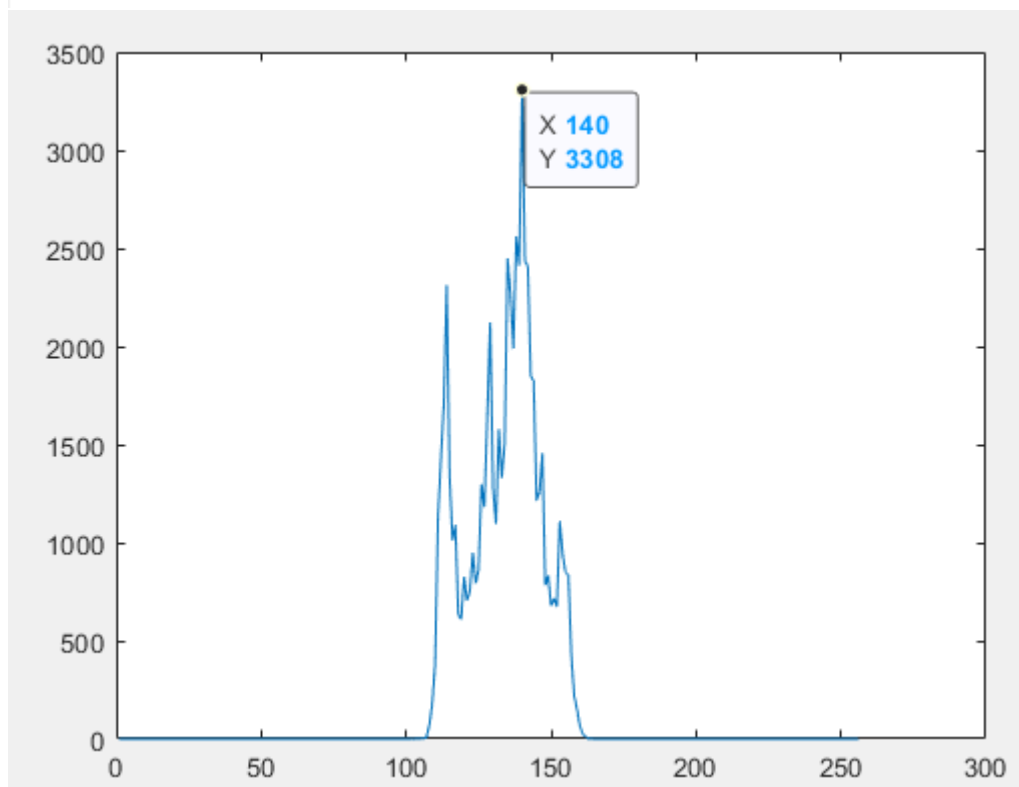
A imagem acima apresenta o novo histograma da imagem.

b) Função `contrast_stretching`

$$s = T(r) = \frac{1}{1 + (m/r)^E}$$

Através do comando abaixo é possível gerar o histograma e descobrir o valor de “m”, ou seja, a intensidade que mais se repete na imagem, e esse valor é 140.

```
>> figure; plot(imhist(img));
```



Através da função e dos comandos abaixo é retornado e mostrado na tela a imagem transformada(melhorada), junto do novo histograma da imagem:

```
function nimg = contrast_stretching(img, m, E)
    doubleimg = im2double(img);

    nimg = (1) ./ (1 + (m ./ doubleimg) .^ E);

    %nimg = uint8(timg);

    imshow(nimg);

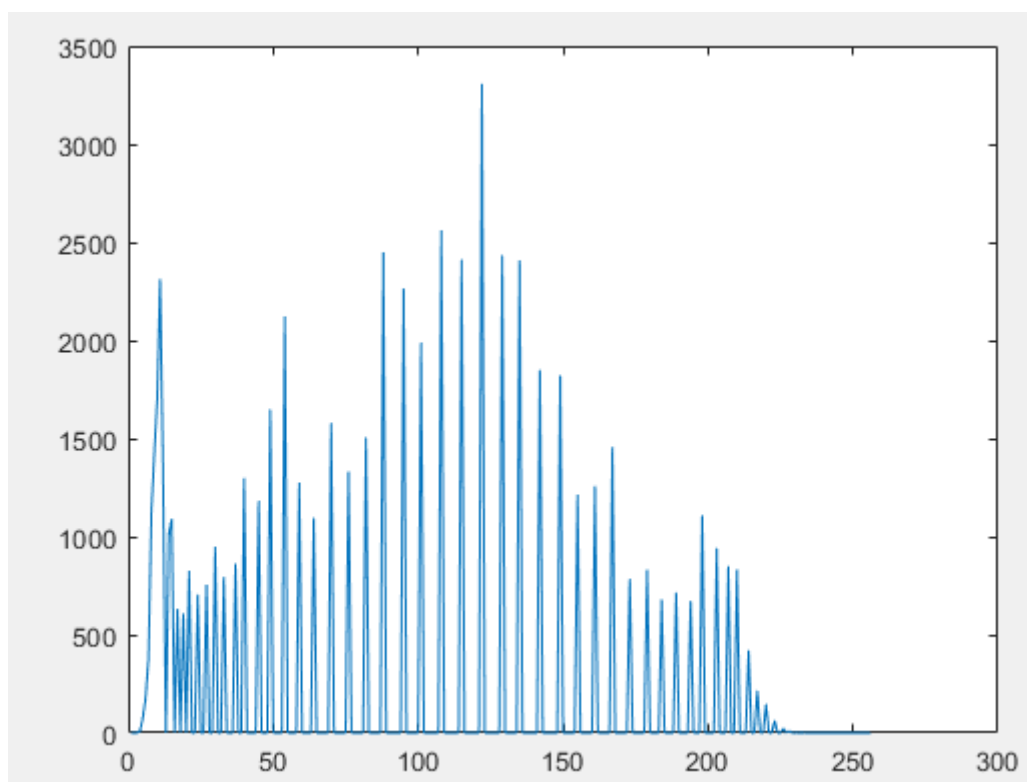
end
```

```
>> nimg = contrast_stretching(img, 140/255, 15);
>> figure; plot(imhist(nimg));
```

Imagem:



Novo histograma:



c) Função Power-law (ou função Gamma):

$$s = cr^\lambda$$

Através da função e dos comandos abaixo é retornado e mostrado na tela a imagem transformada(melhorada), junto do novo histograma da imagem:

```
function nimg = power_law(img, c, lambda)
    doubleimg = im2double(img);

    nimg = c .* doubleimg .^ lambda;

    %nimg = uint8(timg);

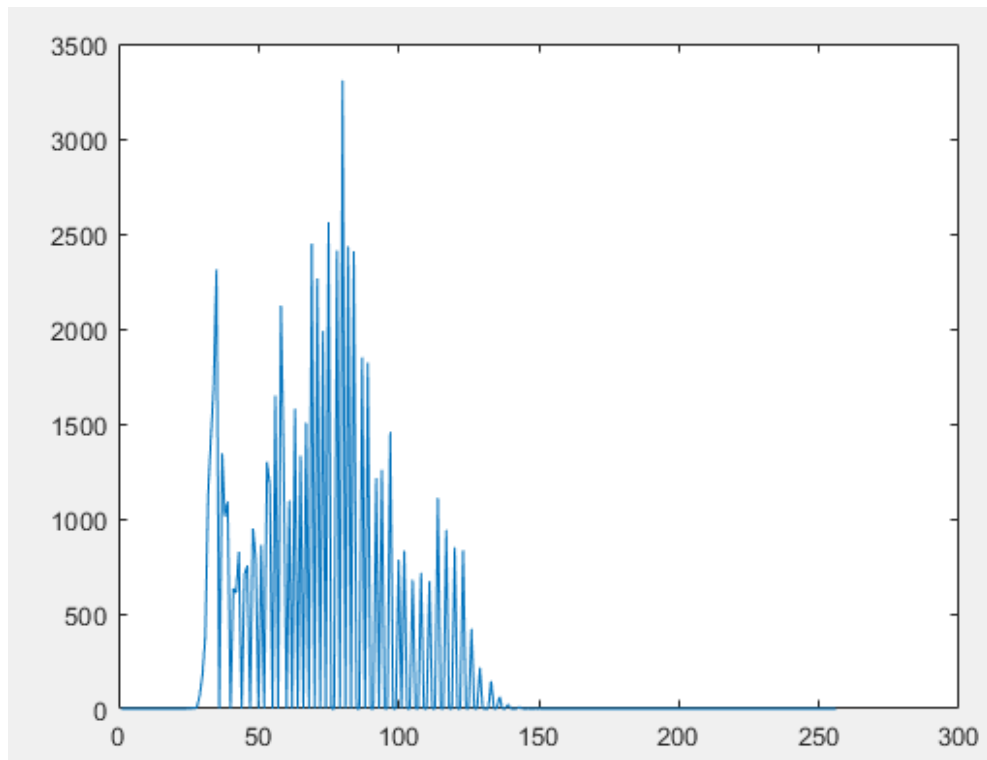
    imshow(nimg);

end
```

```
>> nimg = power_law(img, 3.5, 4);
>> figure;plot(imhist(nimg));
```

Imagem e Histograma gerados:





Questão 2)

Gerar bordas das imagens.

Através da função e do comando abaixo é retornado e mostrado na tela a imagem que representa as bordas da imagem original:

```
function nimg = bordas (img)
    [lin col] = size(img);

    nimg = uint8(zeros(lin, col));

    nimg(2: lin, 2:col)= img(1:(lin-1), 1:(col-1));

    nimg = abs(img - nimg);

end
```

```
>> imshow(bordas(img));
```

A imagem resultante foi:



Destacando mais ainda as bordas pelo comando abaixo a imagem ficará:

```
>> figure;imshow(nimg, [0 6]);
```

