

Manipulação de Imagens no Octave/Matlab

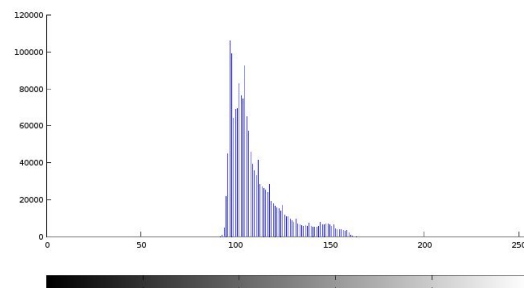
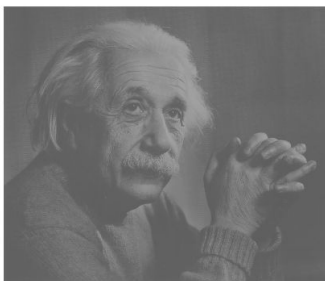
Exercício 1:

- Lendo as imagens e armazenando-as em uma variável para cada imagem lida:

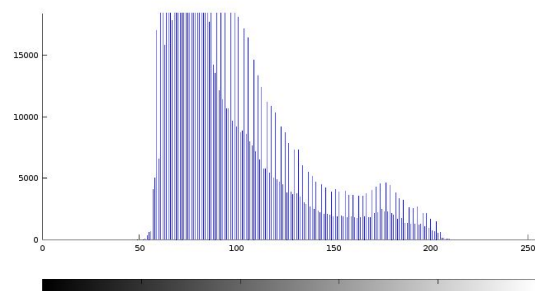
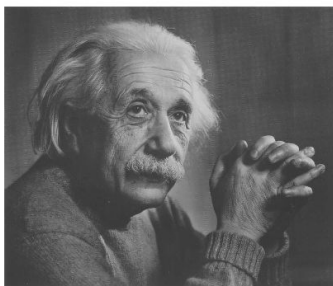
```
ein_low_con = imread("Einstein_low_contrast.png");  
ein_med_con = imread("Einstein_med_contrast.png");  
ein_high_con = imread("Einstein_high_contrast.png");
```

- Calculando o histograma de cada imagem e mostrando as imagens e seus respectivos histogramas:

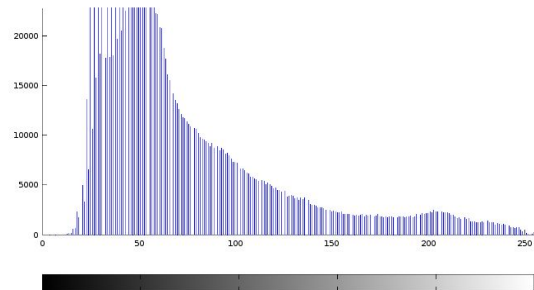
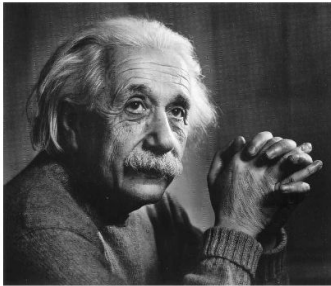
```
subplot(1,2,1), imshow(ein_low_con), subplot(1,2,2), imhist(ein_low_con);
```



```
subplot(1,2,1), imshow(ein_med_con), subplot(1,2,2), imhist(ein_med_con);
```

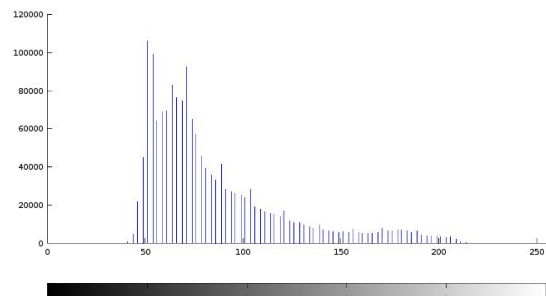
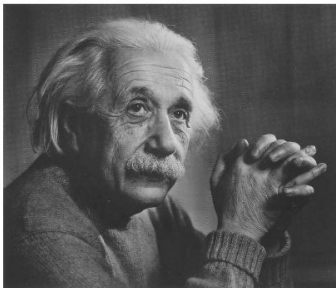


```
subplot(1,2,1), imshow(ein_high_con), subplot(1,2,2), imhist(ein_high_con);
```



- Utilizando *imadjust* para melhorar o contraste da imagem de baixo contraste:

```
ein_low_con_adj = imadjust(ein_low_con, [0.3; 0.7], [0; 1]);  
subplot(1,2,1), imshow(ein_low_con_adj), subplot(1,2,2), imhist(ein_low_con_adj);
```



Exercício 2:

- Lendo a imagem:

```
leme = imread("leme.bmp");
```

- Criando uma nova imagem ajustando o gamma da original para mostrar mais das partes escuras:

gama = 0.6:

```
leme_gamma = imadjust(leme,[0;1], [0;1], 0.6);  
leme_comb = [leme leme_gamma];  
imwrite(leme_comb, "leme_comb.png");
```



gamma = 0.7:

```
leme_gamma = imadjust(leme,[0;1], [0;1], 0.6);
leme_comb2 = [leme leme_gamma2];
imwrite(leme_comb2, "leme_comb2.png");
```



Exercício 3:

- Código:

```
pkg load image

lion = imread("lion.png");
woman = imread("woman.png");

f = [1/9 1/3 1/6 ; 1/4 1/2 1/8 ; 1/7 1/5 1/7];

lion_fil = imfilter(lion, f);
lion_comb = [lion lion_fil];
imwrite(lion_comb, "lion_comb_custom.png");

woman_fil = imfilter(woman, f);
```

```
woman_comb = [woman woman_fil];  
imwrite(woman_comb, "woman_comb_custom.png");
```

- Resultados:



Exercício 4:

- Código:

```
pkg load image  
  
lion = imread("lion.png");  
woman = imread("woman.png");  
  
avg = fspecial("average");  
gauss = fspecial("gaussian");  
lap = fspecial("laplacian");  
sobel = fspecial("sobel");
```



```

lion_avg = imfilter(lion, avg);
lion_gauss = imfilter(lion, gauss);
lion_lap = imfilter(lion, lap);
lion_sobel = imfilter(lion, sobel);

woman_avg = imfilter(woman, avg);
woman_gauss = imfilter(woman, gauss);
woman_lap = imfilter(woman, lap);
woman_sobel = imfilter(woman, sobel);

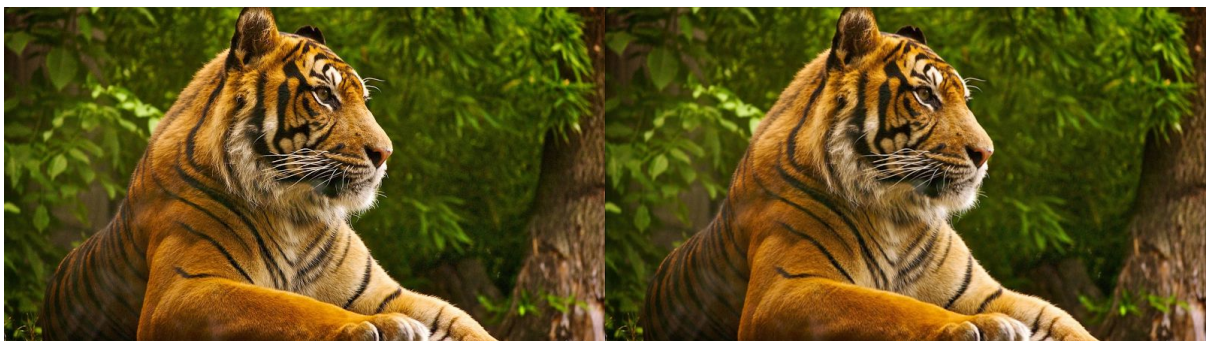
lion_comb_avg = [lion lion_avg];
imwrite(lion_comb_avg, "lion_comb_avg.png");
lion_comb_gauss = [lion lion_gauss];
imwrite(lion_comb_gauss, "lion_comb_gauss.png");
lion_comb_lap = [lion lion_lap];
imwrite(lion_comb_lap, "lion_comb_lap.png");
lion_comb_sobel = [lion lion_sobel];
imwrite(lion_comb_sobel, "lion_comb_sobel.png");

woman_comb_avg = [woman woman_avg];
imwrite(woman_comb_avg, "woman_comb_avg.png");
woman_comb_gauss = [woman woman_gauss];
imwrite(woman_comb_gauss, "woman_comb_gauss.png");
woman_comb_lap = [woman woman_lap];
imwrite(woman_comb_lap, "woman_comb_lap.png");
woman_comb_sobel = [woman woman_sobel];
imwrite(woman_comb_sobel, "woman_comb_sobel.png");

```

- Resultados:

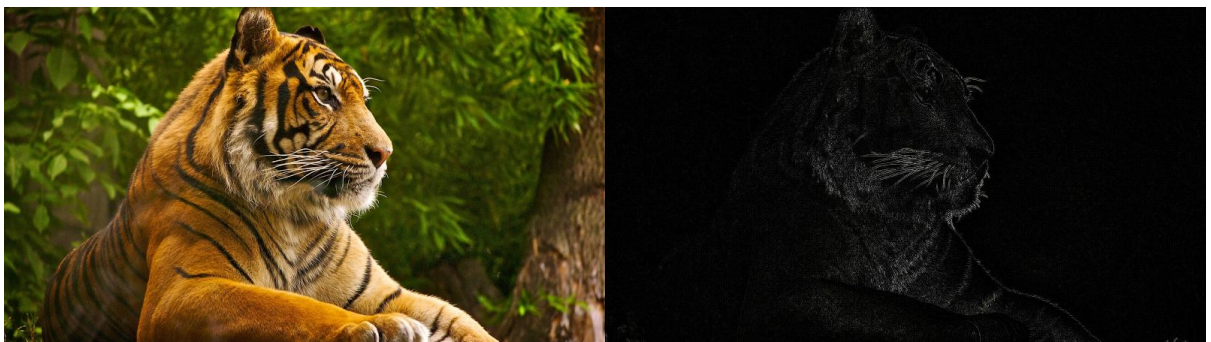
Média:



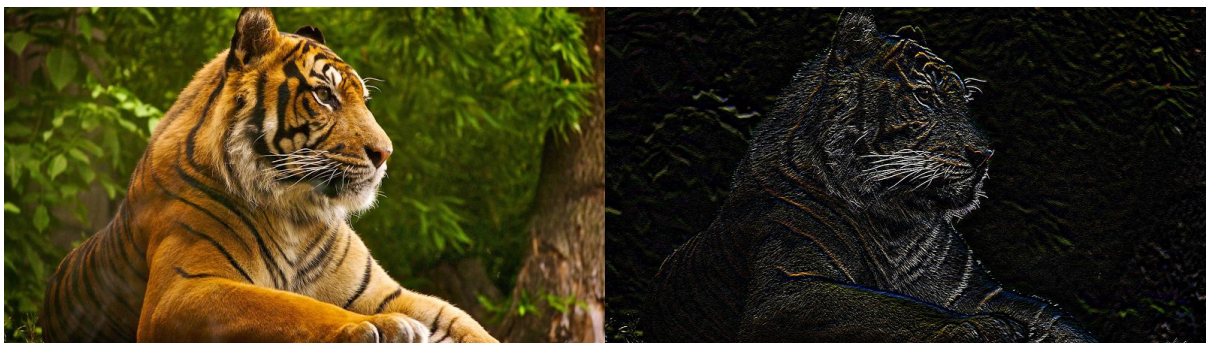
Gaussiana:



Laplaciana:



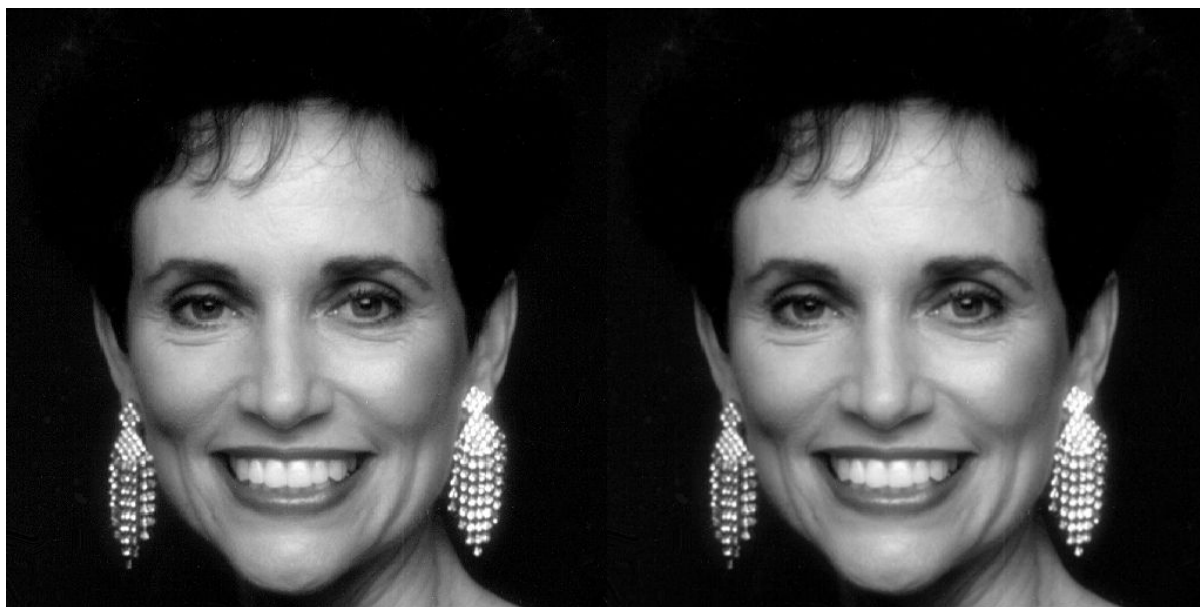
Sobel:



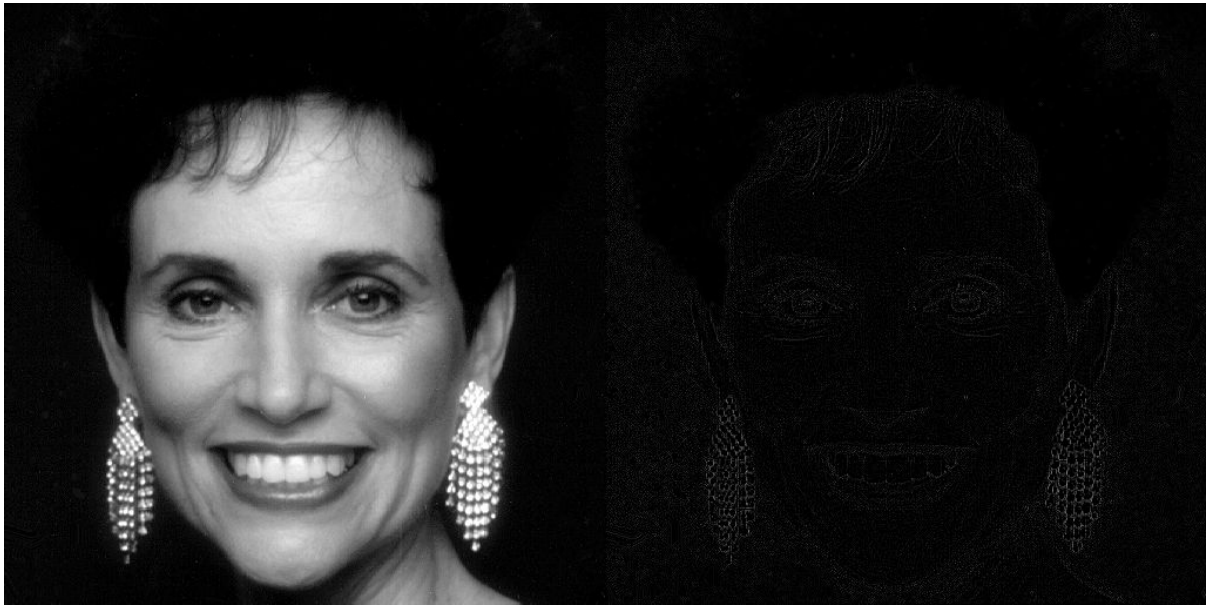
Média:



Gaussiana:



Laplaciana:



Sobel:



Exercício 5:

- Código:

```
pkg load image
```

```
coin = imread("Coins.png");  
imhist(coin);  
lim = 90;
```



```
mask = (coin > lim);
```

```
coin_lim = (coin.*mask);
```

```
coin_comb = [coin coin_lim];
```

```
imwrite(coin_comb, "coin_comb_lim.png");
```

- Resultado:

