

BCC36F – Processamento de Imagens

Prática 3

Prof^a. Dr^a. Aretha Barbosa Alencar
Universidade Tecnológica Federal do Paraná (UTFPR)
Campus Campo Mourão

Nessa atividade prática vocês irão implementar a técnica de **Equalização de Histogramas no Octave**.

1 Instalação do Pacote Image

Nessa atividade poderá se feito o uso da função `imhist()` do pacote `image` do Octave. Para carregar o pacote `image` no Octave você deverá usar a seguinte linha de comando:

```
pkg load image;
```

No entanto, antes de usar qualquer função do pacote `image` ou carregá-lo você antes precisa instalar ele no Octave.

1.1 Windows

1. Fazer o download do arquivo de instalação do pacote `image` na url: <http://octave.sourceforge.net/image/index.html>.
2. Mudar o diretório corrente do Octave para o local onde você salvou o arquivo de instalação do pacote `image`.
3. Usar o comando:

```
pkg install image-2.4.1.tar.gz
```

1.2 Linux

Instale o pacote `octave-image`.

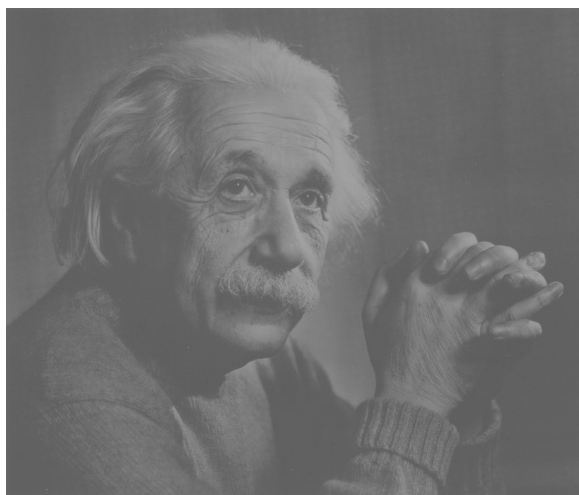


Figura 1: Imagem de entrada ‘imagem.jpg’.

2 Imagem de Entrada

A Figura 1 mostra a imagem de entrada ‘imagem.jpg’, que possui 256 níveis de intensidade de cinza e tem tamanho 800×679 pixels.

Como é possível observar na Figura 2, o histograma dessa imagem usa somente parte como níveis de intensidade mais altos (mais claros) do intervalo total de intensidades.

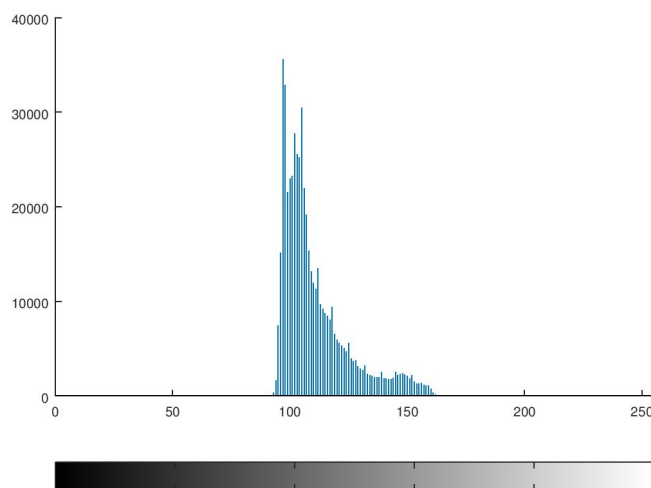


Figura 2: Histograma da imagem ‘imagem.jpg’.

3 Descrição Geral

Nessa atividade, você deve aplicar a técnica de **equalização do histograma**. Passos necessários:

1. Carregar a imagem de entrada ‘imagem.jpg’.

2. Desenvolver um algoritmo que implemente e aplique a técnica de equalização do histograma aprendida em aula.
3. Plote o gráfico de $T(r)$. Para isso você poderá usar a função `plot(x, y)` e a função `axis([x_lo x_hi y_lo y_hi])`.
4. Salvar a imagem em disco após a equalização do histograma.

Como dito anteriormente, é permitido o uso da função `imhist()`. Essa função pode ser utilizada para fazer o cálculo do histograma não normalizado, usando-se o fomato: `[counts, x] = imhist(...)`.

Já o uso da função `histeq()` do pacote `image` do Octave, não será aceito nessa atividade.

4 Resultado Esperado

A Figura 3 exibe a imagem de entrada após a aplicação da técnica de equalização de histograma para efeitos de comparação.

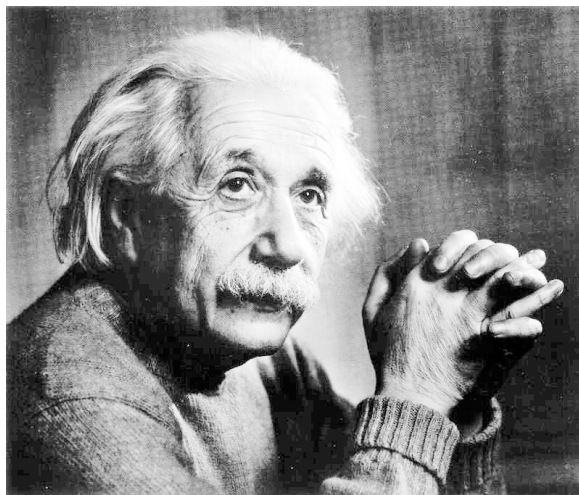


Figura 3: Imagem após a equalização do histograma.

Já a Figura 4 mostra o histograma da Figura 3, ou seja, após a aplicação da técnica de equalização de histograma. Perceba que agora existem pixels distribuídos em todo o intervalo de intensidades.

5 Documentação do Octave

A documentação das funções e sintaxe básicas do Octave encontra-se em https://www.gnu.org/software/octave/doc/interpreter/index.html#SEC_Contents.

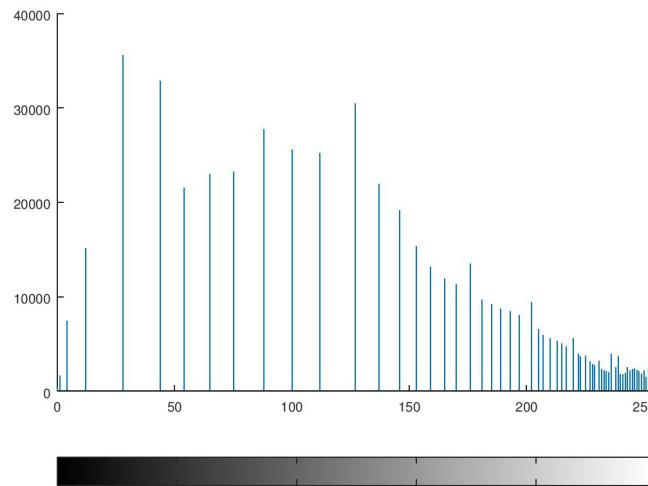


Figura 4: Histograma equalizado.

Já a **documentação das funções do pacote `image`** do Octave encontra-se em <http://octave.sourceforge.net/image/overview.html>.

Documentação de funções específicas no Octave úteis para essa atividade:

- Funções `imread(...)` e `imwrite(...)`: <https://www.gnu.org/software/octave/doc/interpreter/Loading-and-Saving-Images.html>
- Funções de utilidade para matrizes, como a função `zeros(...)`: <https://www.gnu.org/software/octave/doc/interpreter/Special-Utility.html>
- Função `imhist()`: <http://octave.sourceforge.net/image/function/imhist.html>
- Função `plot(x, y)`: https://www.gnu.org/software/octave/doc/interpreter/Two_002dDimensional-Plots.html
- Função: `axis([x_lo x_hi y_lo y_hi])`: <https://www.gnu.org/software/octave/doc/interpreter/Axis-Configuration.html>
- O comando `for` em Octave: <https://www.gnu.org/software/octave/doc/interpreter/The-for-Statement.html>