**Project number: Projeto #1**

**Course name:** FTL079 – PROCESSAMENTO DIGITAL DE IMAGENS

**Student's name:**

**#1: Diego Giovanni de Alcântara Vieira**

**Date due: 04 de outubro de 2020.**

**Date handed in: 02 de outubro de 2020.**

**Technical discussion and results**

(One to three pages - max).

O projeto consiste de dois scripts: o primeiro é solicitado a extração de valores de pixels de uma imagem conforme passado o valor das linhas e colunas desejadas, e o segundo é solicitado a visualização de valores de pixels conforme o clique do *mouse* dentro das regiões válidas da imagem. Há também a necessidade de criação de scripts de execução das funções solicitadas. Segue a discussão técnica detalhada do que foi feito em termos de código de MATLAB.

**Função main.m**

A função principal do projeto é responsável por executar as funções pedidas: pixVal4e.m e cursorValues4e.m. Além disso, na função principal implementou-se a limpeza do *workspace* (armazenamento de variáveis) e do console do MATLAB.

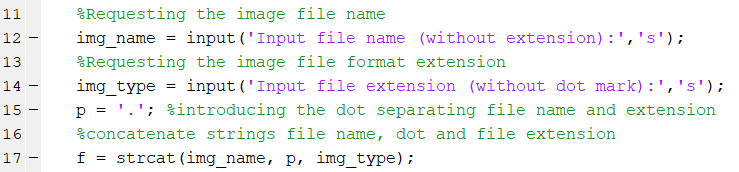
Outra função importante é a generalização da inserção da imagem a ser operada, o que foi feito da seguinte forma:

Figura 1 - Generalização de imagem

Conforme visto acima, por esse trecho de código é possível que o usuário insira qualquer imagem disponível no diretório raiz do MATLAB, bastando informar o nome do arquivo e sua extensão separadamente no console. Na linha 17, é feita a concatenação das respectivas *strings* que formarão o nome completo da imagem. Vale destacar que no MATLAB R2019b, versão utilizada, esse processo vale apenas para imagens formatadas. Extensões de arquivo do tipo raw, dat ou similares não poderão ser processadas por esse trecho de código.

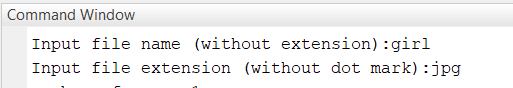


Figura 2 - Inserção da imagem via console

Na figura 2 mostra-se o processo de digitação da imagem escolhida para o processamento. Como dito anteriormente, o usuário informa separadamente o nome do arquivo e o formato da imagem. Tais informações serão concatenadas em uma única string, junto com o ponto separador, e então essa informação será passada como parâmetro para as funções.

O próximo passo é obter as informações da imagem: número de colunas, número de linhas, camada e densidade de pixels. Isso é feito no trecho de código mostrado a seguir:

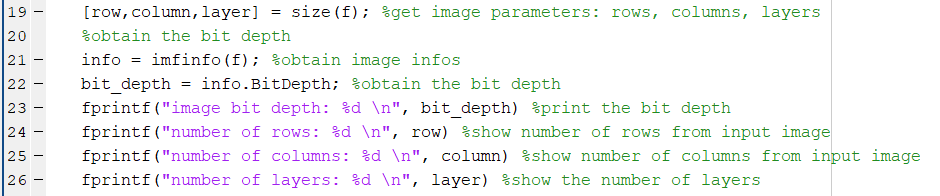


Figura 3 - Obtendo informações da imagem.

Para o arquivo ‘girl.jpg’ passado, as informações foram as seguintes, conforme o console do MATLAB:

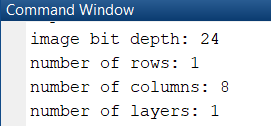


Figura 4 - Mostrando informações da imagem.

Em seguida, é feita a rotina de inserção de parâmetros para a função pixVal4e.m. Nesse caso, é solicitado do usuário a linha e a coluna da qual ele deseja obter o valor de pixel.

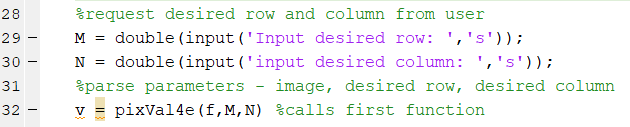


Figura 5 - Solicitando informações de linha e coluna.

Para o teste específico, foi solicitado os seguintes valores:



Figura 6 - Valores de teste da função pixVal4e.m

Esses valores são passados como parâmetros para a função pixVal4e conforme a linha 32 na figura 5. Os retornos apresentados são chamados dentro das funções, as quais começaremos a ver individualmente.

**Função pixVal4e.m**

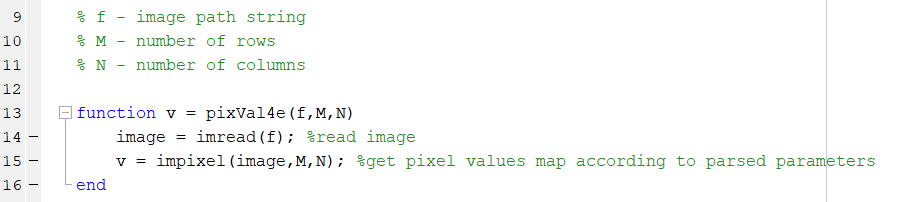
A função pixVal4e tem por objetivo retornar o valor de pixel tendo valores de linha e coluna como entrada através do console. Os valores passados foram mostrados na figura 6, e agora observaremos o que foi feito para obter o resultado.

Figura 7 - Escopo da função pixVal4e

O escopo da função é bem simples: ela recebe a string f referente ao caminho do diretório da imagem, obrigatoriamente o diretório raiz do MATLAB e os valores desejados de linhas (M) e colunas (N). A variável image recebe os dados da imagem através do comando imread, e a variável v, que será passada como retorno, recebe o valor do pixel de acordo com a imagem, valor da linha e valor da coluna. Na função main.m, os parâmetros são passados de acordo com a figura 8 e o retorno é sinalizado.

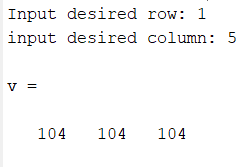


Figura - Parâmetros passados e retorno da função pixVal4e.m

**Função cursorValues4e.m**

A função cursorValues4e tem o mesmo princípio de funcionalidade da versão anterior, porém obtendo o valor de pixel através do click do mouse através de um handler de imagem próprio do MATLAB. O escopo da função é apresentado na figura a seguir:

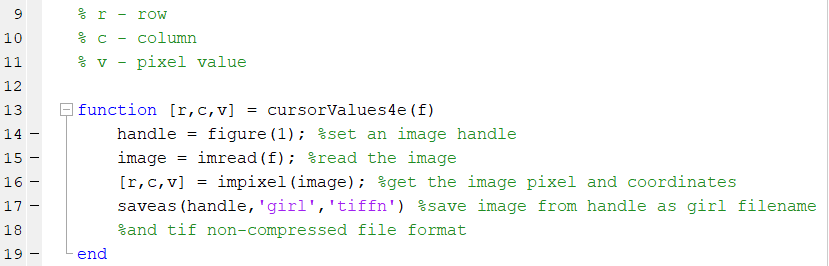


Figura - Escopo da função cursorValues4e

A função cursorValues4e.m recebe como parâmetro somente a string f contendo o caminho do diretório da imagem a ser lida. A seguir, ela gera um handler de figura do MATLAB configurado com o identificador “1”. Através da função impixel, recebendo o vetor [r,c,v], o handler de figura indica um cursor no botão do mouse para que o usuário clique no ponto da imagem que deseje obter a coluna, linha e valor do pixel referente. É possível fazer inúmeros cliques, obtendo diferentes quantidades de valores de pixel antes de interromper a execução com um toque no botão de espaço do teclado, um clique duplo ou um clique com o botão direito do mouse. Para o teste dessa função, foi feito dois cliques em regiões muito próximas da imagem em torno dos olhos da garota mostrada. Segue os resultados:

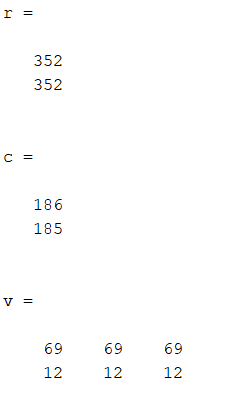


Figura - retornos da função cursorValues.m

Em seguida, conforme pedido salva-se a imagem em formato .tif sem compressão, passando-se uma variável contendo o handler da figura a ser salva, o nome do arquivo gerado e o parâmetro ‘tiffn’, correspondente ao formato solicitado, utilizando-se a função saveas (ver linha 17 da figura 9). Em seguida, na função principal, fecha-se o handler da imagem passando-se o comando close. 

Figura - Imagem utilizada durante a execução do trabalho.

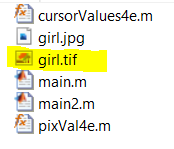


Figura - Diretório do MATLAB mostrando o arquivo tif criado.