

Muhammad Talha

P15-6087

Digital Image Processing

Code:

function varargout = assignment1(varargin)

% ASSIGNMENT1 MATLAB code for assignment1.fig

% ASSIGNMENT1, by itself, creates a new ASSIGNMENT1 or raises the existing

% singleton\*.

%

% H = ASSIGNMENT1 returns the handle to a new ASSIGNMENT1 or the handle to

% the existing singleton\*.

%

% ASSIGNMENT1('CALLBACK',hObject,eventData,handles,...) calls the local

% function named CALLBACK in ASSIGNMENT1.M with the given input arguments.

%

% ASSIGNMENT1('Property','Value',...) creates a new ASSIGNMENT1 or raises the

% existing singleton\*. Starting from the left, property value pairs are

% applied to the GUI before assignment1\_OpeningFcn gets called. An

% unrecognized property name or invalid value makes property application

% stop. All inputs are passed to assignment1\_OpeningFcn via varargin.

%

% \*See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one

% instance to run (singleton)".

%

% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help assignment1

% Last Modified by GUIDE v2.5 30-Sep-2017 02:29:05

% Begin initialization code - DO NOT EDIT

gui\_Singleton = 1;

gui\_State = struct('gui\_Name', mfilename, ...

'gui\_Singleton', gui\_Singleton, ...

'gui\_OpeningFcn', @assignment1\_OpeningFcn, ...

'gui\_OutputFcn', @assignment1\_OutputFcn, ...

'gui\_LayoutFcn', [] , ...

'gui\_Callback', []);

if nargin && ischar(varargin{1})

gui\_State.gui\_Callback = str2func(varargin{1});

end

if nargout

[varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});

else

gui\_mainfcn(gui\_State, varargin{:});

end

% End initialization code - DO NOT EDIT

% --- Executes just before assignment1 is made visible.

function assignment1\_OpeningFcn(hObject, eventdata, handles, varargin)

% This function has no output args, see OutputFcn.

% hObject handle to figure

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% varargin command line arguments to assignment1 (see VARARGIN)

% Choose default command line output for assignment1

handles.output = hObject;

% Update handles structure

guidata(hObject, handles);

% UIWAIT makes assignment1 wait for user response (see UIRESUME)

% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.

function varargout = assignment1\_OutputFcn(hObject, eventdata, handles)

% varargout cell array for returning output args (see VARARGOUT);

% hObject handle to figure

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure

varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.

function pushbutton1\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%image1 and image 2 is variables

global image1 image2

[path1,userCance]=imgetfile();

grayImage = imread(path1);

% Get the dimensions of the image.

% numberOfColorBands should be = 1.

[rows, columns, numberOfColorChannels] = size(grayImage);

if numberOfColorChannels > 1

% It's not really gray scale like we expected - it's color.

% Convert it to gray scale by taking only the green channel.

%grayImage = grayImage(:, :, 2); % Take green channel.

msgbox(sprintf('Error'),'Error','You enter incorrect image');

else

image1=imread(path1);

axes(handles.axes1);

image2=image1;

imshow(image1);

end

% end

% --- Executes on button press in pushbutton2.

function pushbutton2\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton2 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

imsave();

%if user cancel image then userCancel hans 1 ELSE 0

%filename is use for taking path of image that where the is image is saved

%it is optional

% --- Executes on button press in pushbutton3.

function pushbutton3\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton3 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%-----------------FLIP----------

global image1

flipImage=fliplr(image1);

imagesc(flipImage);

%flip left to right

% --- Executes on button press in pushbutton4.

function pushbutton4\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton4 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%------------------------NEGATIVE------------------------

global image1 image2

NEGATIVE=imcomplement(image1);

image2=NEGATIVE;

axes(handles.axes1);

imshow(NEGATIVE);

% --- Executes on button press in pushbutton5.

function pushbutton5\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton5 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%---------------CONTRAST------------------------

global image1 image2

%A=mean(image1);

%bb=imfinfo(image1);

%bb=bb.BitDepth

%disp(bb);

image2=image1;

[r, c,d] = size(image1);

myimage=int32(image2);

sum2=sum(image1(:));

sum1=0;

X=r\*c;

sum2=sum2/X;

for i=1:r

for j=1:c

pixel=myimage(i,j);

sum1=sum1+(sum2-pixel)^2;

end

end

var=sum1/X;

contrast=sqrt(double(var));

disp(contrast);

% --- Executes on button press in pushbutton6.

function pushbutton6\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton6 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%----------------------THRESHOLDING----------

global image1 image2

%A=mean(image1);

[r, c,d] = size(image1);

sum2=sum(image1(:));

X=r\*c;

sum2=sum2/X;

image\_thresholded = zeros(size(image1));

for i=1:r

for j=1:c

pixel=image1(i,j);

if pixel<=sum2 , pixel=0.000000000; end

if pixel>sum2 , pixel=255; end

image\_thresholded(i,j)=pixel;

end

end

im\_thresh=mat2gray(image\_thresholded);

image2=im\_thresh;

%figure%,imshow(im\_thresh);

axes(handles.axes1);

imshow(im\_thresh);

%title('thresholding');

% --- Executes on button press in pushbutton7.

function pushbutton7\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton7 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% ------------------LOG TRANSFORMATION------

global image1 image2

f= zeros(size(image1));

im=double(image1);

[r, c,d] = size(image1);

%constant value of c is 1

C=1;

for i = 1:r

for j = 1:c

R=im(i,j);

f(i,j)=C\*log10(1+R);

end

end

im\_log=mat2gray(f);

image2=im\_log;

axes(handles.axes1);

imshow(im\_log);

%imshow(f), figure, imshow(z);

% --- Executes on button press in pushbutton8.

function pushbutton8\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton8 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%-----------------CONTRAST STRETCHING-----------

global image1 image2

%i = image1(:,:,1);

rtemp = min(image1); % find the min. value of pixels in all the columns (row vector)

rmin = min(rtemp); % find the min. value of pixel in the image

rtemp = max(image1); % find the max. value of pixels in all the columns (row vector)

rmax = max(rtemp); % find the max. value of pixel in the image

[r, c,d] = size(image1);

f= zeros(size(image1));

m = 255/(rmax - rmin); % find the slope of line joining point (0,255) to (rmin,rmax)

for i=1:r

for j=1:c

pixel=image1(i,j);

value=(pixel-rmin)\*m;

f(i,j)=value;

end

end

f=mat2gray(f);

image2=f;

axes(handles.axes1);

imshow(f);

% --- Executes on button press in pushbutton11.

function pushbutton11\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton11 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%-------------------histogram-----------------

global image1 image2

imhist(image2);