LOSSLESS IMAGE COMPRESSION

Prac 10

Submitted to Prof TWINKLE BHAVSAR

~ Kanisha Shah (19BCE253)

CODE

function varargout = ImageCompression(varargin)

%GUI formation

gui\_Singleton = 1;

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

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

'gui\_OpeningFcn', @ImageCompression\_OpeningFcn, ...

'gui\_OutputFcn', @ImageCompression\_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

function ImageCompression\_OpeningFcn(hobj, event, handle, varargin)

handle.output = hobj;

guidata(hobj, handle);

guidata(hobj, handle);

set(handle.axes3,'visible','off')

set(handle.axes4,'visible','off')

axis off

axis off

function varargout = ImageCompression\_OutputFcn(hobj, event, handle)

varargout{1} = handle.output;

function pushbutton1\_Callback(hobj, event, handle)

global file\_name;

%the program is sufficient to work with all types of extension

file\_name=uigetfile({'\*.bmp;\*.jpg;\*.png;\*.tiff;';'\*.\*'},'Select an Image File');

fileinfo = dir(file\_name);

SIZE = fileinfo.bytes;

Size = SIZE/1024;

%displaying current size in the GUI

set(handle.text7,'string',Size);

imshow(file\_name,'Parent', handle.axes3)

function pushbutton2\_Callback(hobj, event, handle)

global file\_name;

if(~ischar(file\_name))

errordlg('Please select Images first');

else

I1 = imread(file\_name);

%masking the 1st frame

I = I1(:,:,1);

I = im2double(I);

%Discrete cosine transform matrix

T = dctmtx(8);

B = blkproc(I,[8 8],'P1\*x\*P2',T,T');

mask = [1 1 1 1 0 0 0 0

1 1 1 0 0 0 0 0

1 1 0 0 0 0 0 0

1 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0];

B2 = blkproc(B,[8 8],'P1.\*x',mask);

I2 = blkproc(B2,[8 8],'P1\*x\*P2',T',T);

%masking the 2nd frame

I = I1(:,:,2);

I = im2double(I);

T = dctmtx(8);

B = blkproc(I,[8 8],'P1\*x\*P2',T,T');

mask = [1 1 1 1 0 0 0 0

1 1 1 0 0 0 0 0

1 1 0 0 0 0 0 0

1 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0];

B2 = blkproc(B,[8 8],'P1.\*x',mask);

I3 = blkproc(B2,[8 8],'P1\*x\*P2',T',T);

%masking the 3rd frame

I = I1(:,:,3);

I = im2double(I);

T = dctmtx(8);

B = blkproc(I,[8 8],'P1\*x\*P2',T,T');

mask = [1 1 1 1 0 0 0 0

1 1 1 0 0 0 0 0

1 1 0 0 0 0 0 0

1 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0];

B2 = blkproc(B,[8 8],'P1.\*x',mask);

I4 = blkproc(B2,[8 8],'P1\*x\*P2',T',T);

%concatinating all 3 frames

L(:,:,:)=cat(3,I2, I3, I4);

%writing into the file

imwrite(L,'CompressedColourImage.jpg');

fileinfo = dir('CompressedColourImage.jpg');

SIZE = fileinfo.bytes;

Size = SIZE/1024;

%displaying in the gui

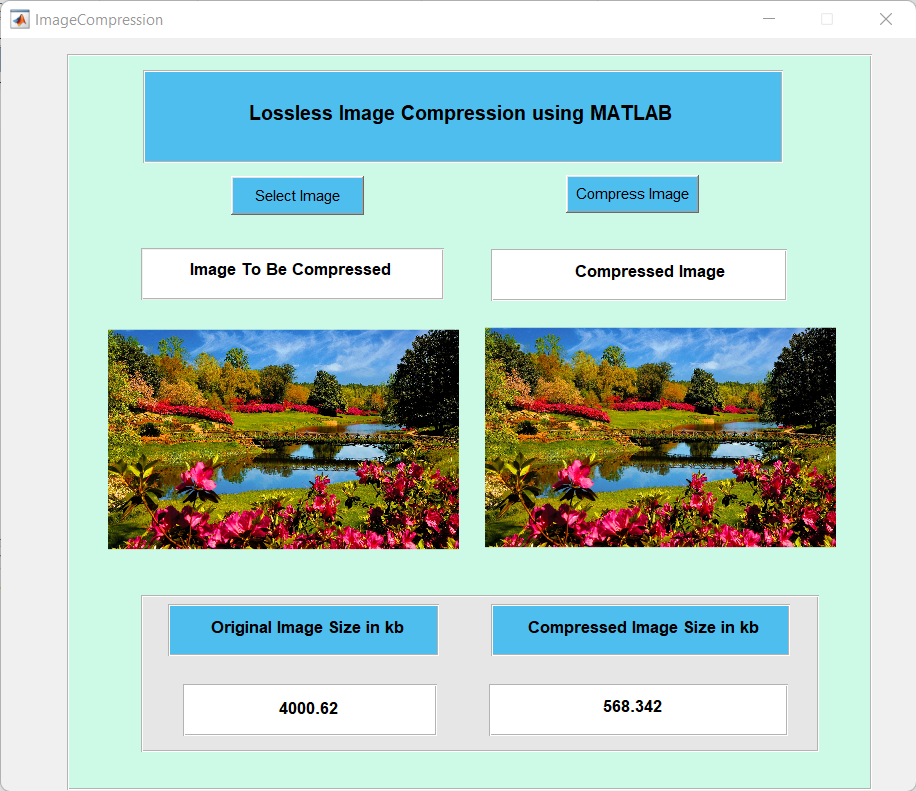
set(handle.text8,'string',Size);

imshow(L,'Parent', handle.axes4)

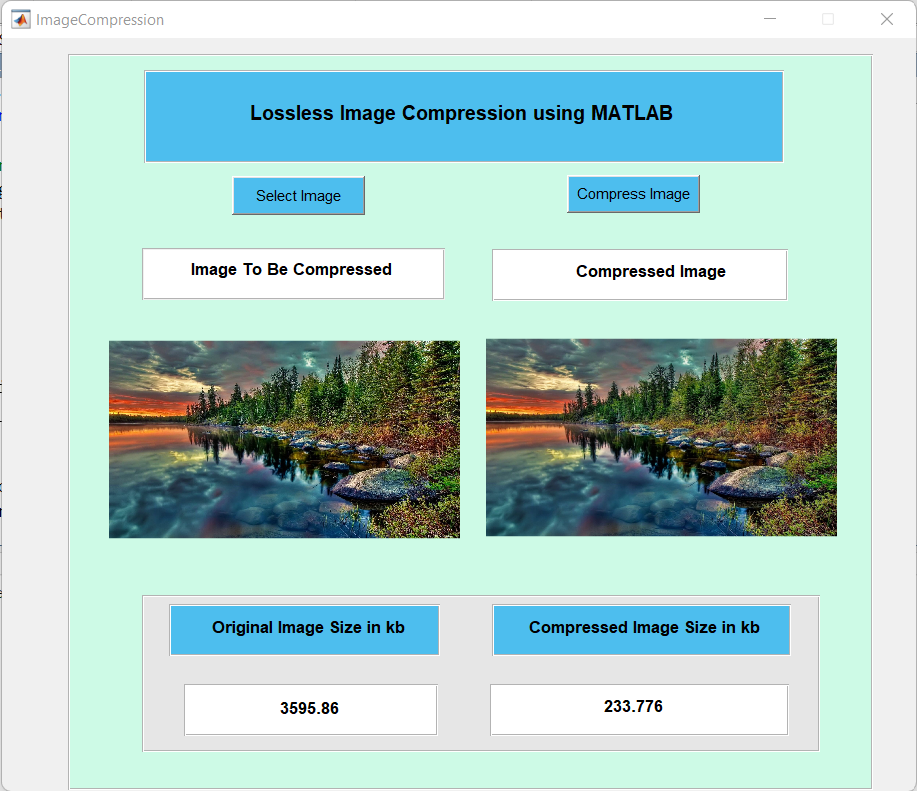
end

OUTUT

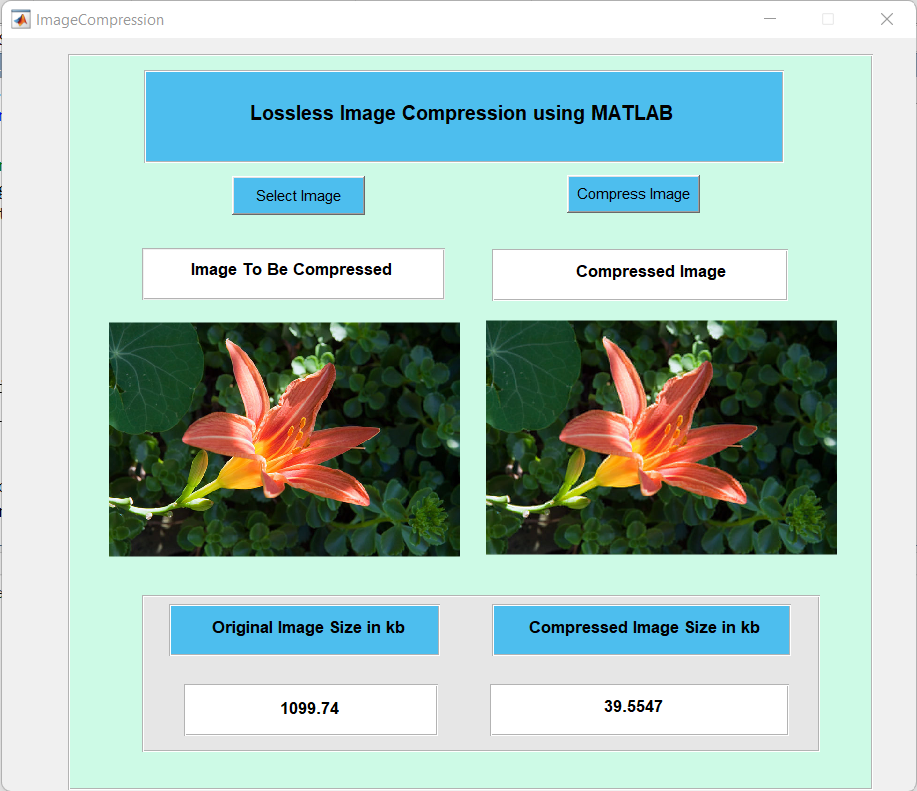
JPG



PNG



BMP



TIFF

Original:



Compressed:

