FUNDAMENTAL OF IMAGE PROCESSING

ASSIGNMENT 2

Image Enhancement in Spatial Domain

NAME:

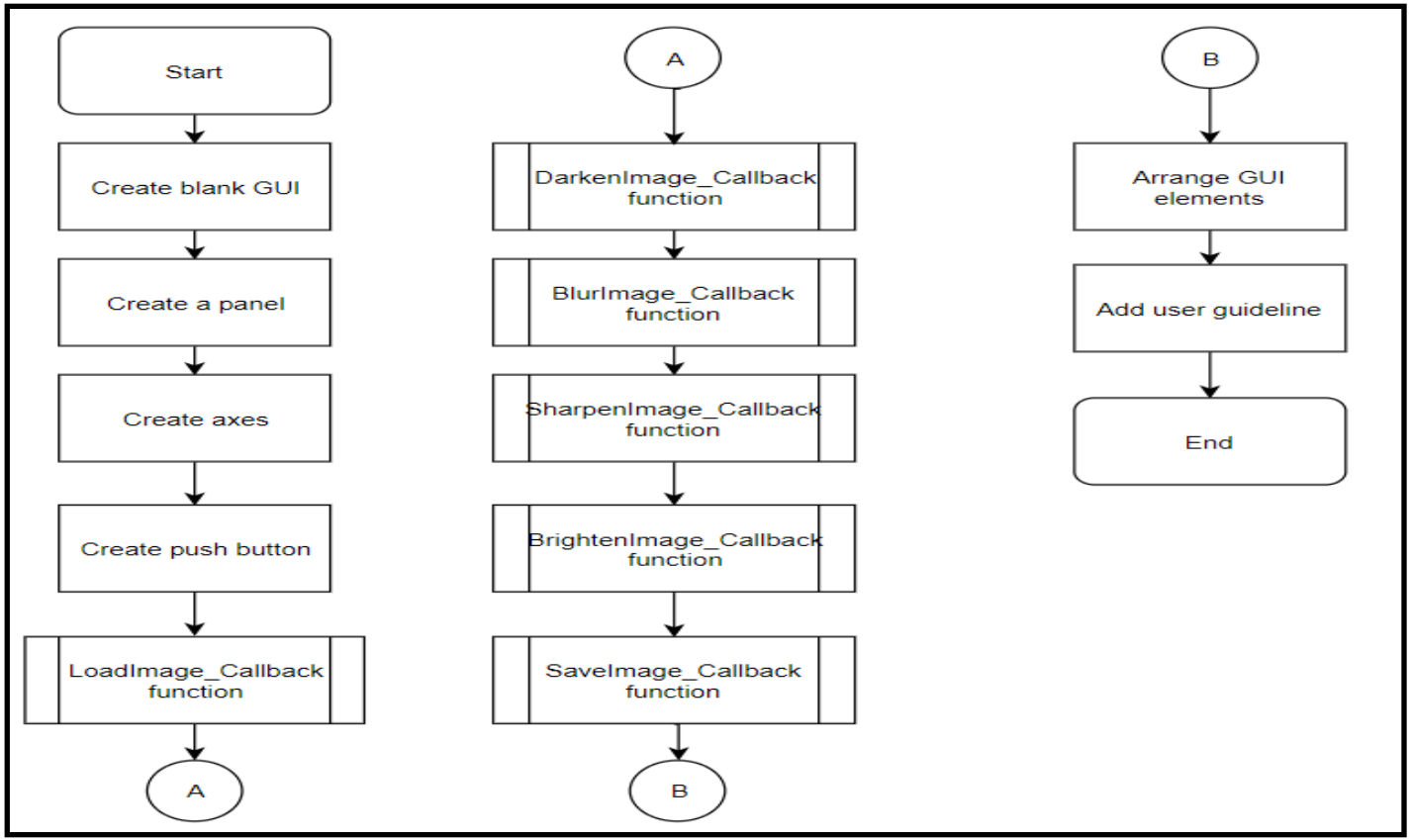
1) LIM BAO JING (A17CS0076)

2) LOW CHIA JING (A17CS0083)

3)TAN SEE JOU (A17CS0218)

LECTURER NAME: DR. MD. SAH BIN HJ. SALAM

Flow of process



Load Image

```
% --- Executes on button press in LoadImage.
function LoadImage_Callback(hObject, eventdata, handles)
% hObject    handle to LoadImage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
[file, path] = uigetfile({'*.jpg';'*.png';'*.bmp';'*.tif' });
fullFileName = fullfile(path, file);
A = imread(fullFileName);
axes(handles.axes1);
imshow(A);
axes(handles.axes2);
imshow(A);
setappdata(handles.axes1, 'img', A);
setappdata(handles.axes2, 'img2', A);
```

Darken Image

```
% --------------------DARKEN IMAGE--------------------------------
% --- Executes on button press in DarkenImage.
function DarkenImage_Callback(hObject, eventdata, handles)
% hObject    handle to DarkenImage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
%B = getimage(handles.axes2);
B = getappdata(handles.axes2, 'img2');
B_adj = imdivide(B, 3.0);       %to darken the image
                                %B_adj =immultiply(img,0.5);
bw = im2uint8(roipoly(B));
bw_cmp = bitcmp(bw);

roi = bitor(B_adj, bw_cmp);
not_roi = bitor(B, bw);
new_B = bitand(roi, not_roi);

axes(handles.axes2);
imshow(new_B);
setappdata(handles.axes2, 'img2', new_B);
```

Blur Image

```
% --------------------BLUR IMAGE-------------------------------
% --- Executes on button press in BlurImage.
function BlurImage_Callback(hObject, eventdata, handles)
% hObject    handle to BlurImage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
%B = getappdata(handles.DarkenImage, 'img2');
B = getappdata(handles.axes2, 'img2');
H =fspecial('disk', 20);
blurred = imfilter(B, H);       %to blur the image
                                %also can use blurred = imgaussfilt(B,20);
bw = im2uint8(roipoly(B));
bw_cmp = bitcmp(bw);

roi = bitor(blurred, bw_cmp);
not_roi = bitor(B, bw);
new_B = bitand(roi, not_roi);

axes(handles.axes2);
imshow(new_B);
setappdata(handles.axes2, 'img2', new_B);
```

Sharpen Image

```
% --------------------SHARPEN IMAGE--------------------------------
% --- Executes on button press in SharpenImage.
function SharpenImage_Callback(hObject, eventdata, handles)
% hObject    handle to SharpenImage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
%B = getappdata(handles.DarkenImage, 'img2');
B = getappdata(handles.axes2, 'img2');
sharpB = imsharpen(B);          %to sharp the image

bw = im2uint8(roipoly(B));
bw_cmp = bitcmp(bw);

roi = bitor(sharpB, bw_cmp);
not_roi = bitor(B, bw);
new_B = bitand(roi, not_roi);

axes(handles.axes2);
imshow(new_B);
setappdata(handles.axes2, 'img2', new_B);
```

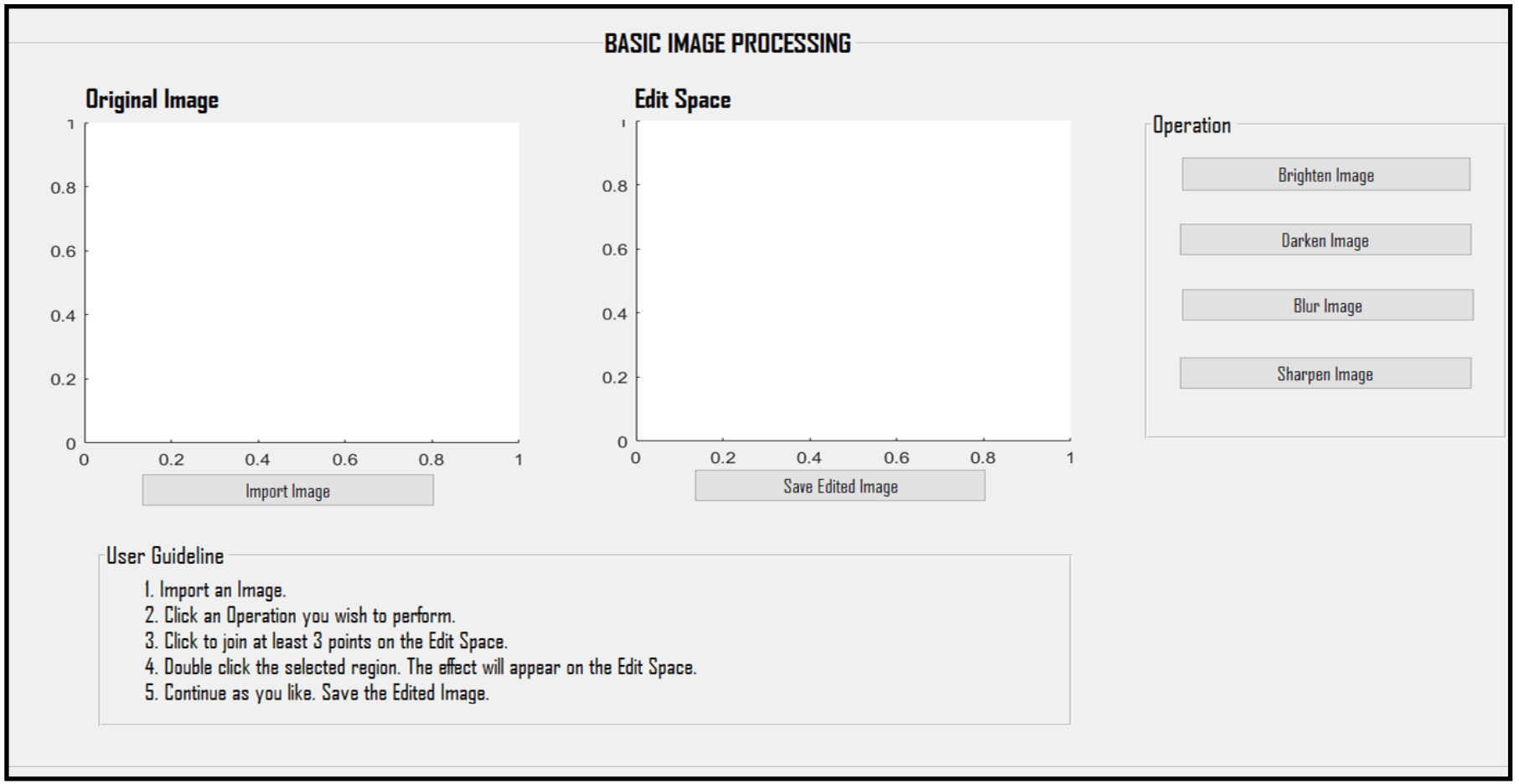
Brighten Image

```
% --------------------BRIGHTEN IMAGE-------------------------------
% --- Executes on button press in BrightenImage.
function BrightenImage_Callback(hObject, eventdata, handles)
% hObject    handle to BrightenImage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
%B = getappdata(handles.DarkenImage, 'img2');
B = getappdata(handles.axes2, 'img2');
brightB = immultiply(B,1.5);        %to brighten the image
                                    %also can use brightB = imdivide(B,0.5);
bw = im2uint8(roipoly(B));
bw_cmp = bitcmp(bw);

roi = bitor(brightB, bw_cmp);
not_roi = bitor(B, bw);
new_B = bitand(roi, not_roi);

axes(handles.axes2);
imshow(new_B);
setappdata(handles.axes2, 'img2', new_B);
```

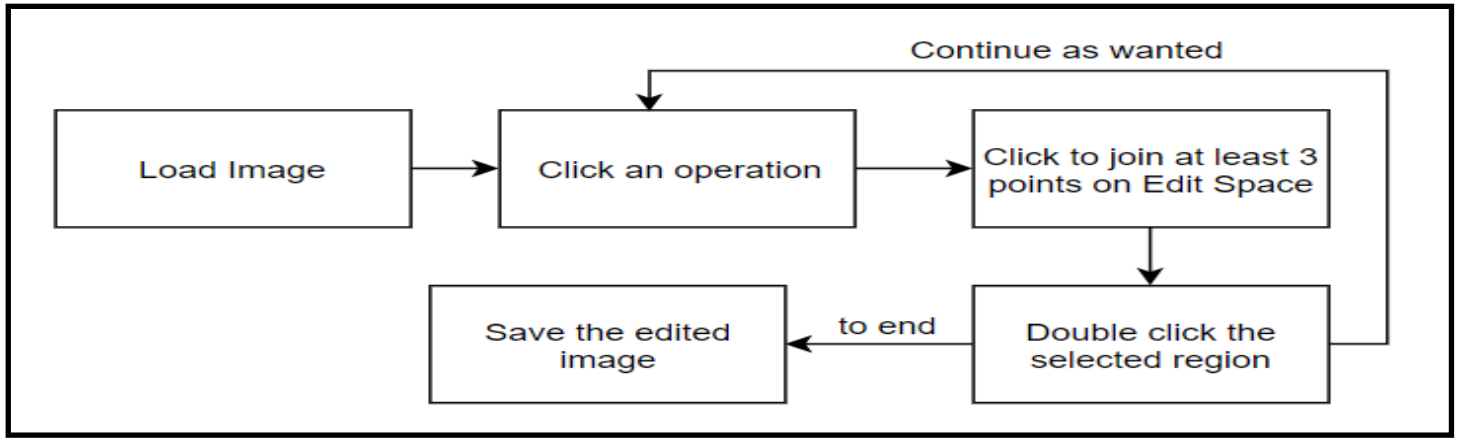
Save Image

```
% --------------------SAVE IMAGE--------------------------------
% --- Executes on button press in SaveImage.
function SaveImage_Callback(hObject, eventdata, handles)
% hObject    handle to SaveImage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
imsave(handles.axes2);
```

Final GUI


BASIC IMAGE PROCESSING
Original Image
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
1
Import Image
Edit Space
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
1
Save Edited Image
Operation
Brighten Image
Darken Image
Blur Image
Sharpen Image
User Guideline
1. Import an Image.
2. Click an Operation you wish to perform.
3. Click to join at least 3 points on the Edit Space.
4. Double click the selected region. The effect will appear on the Edit Space.
5. Continue as you like. Save the Edited Image.

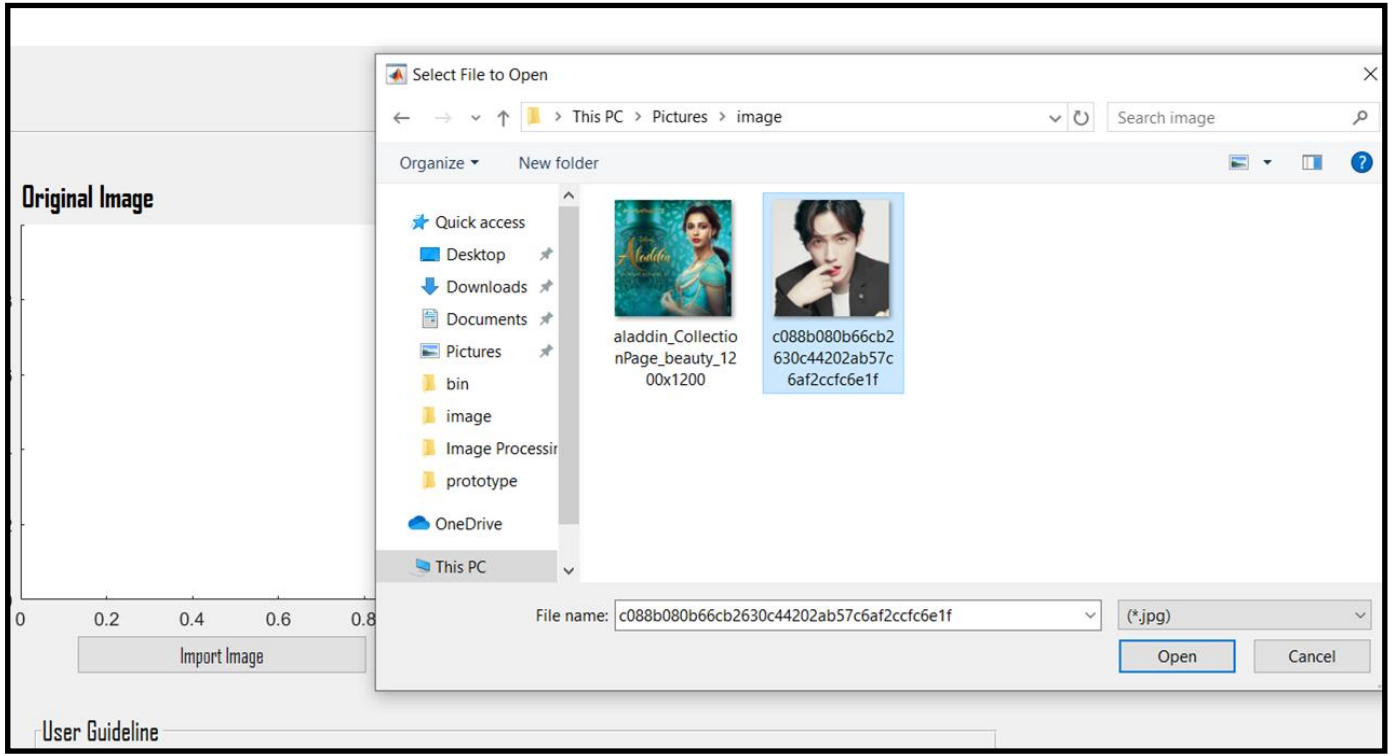
How the user uses the application



User Guideline

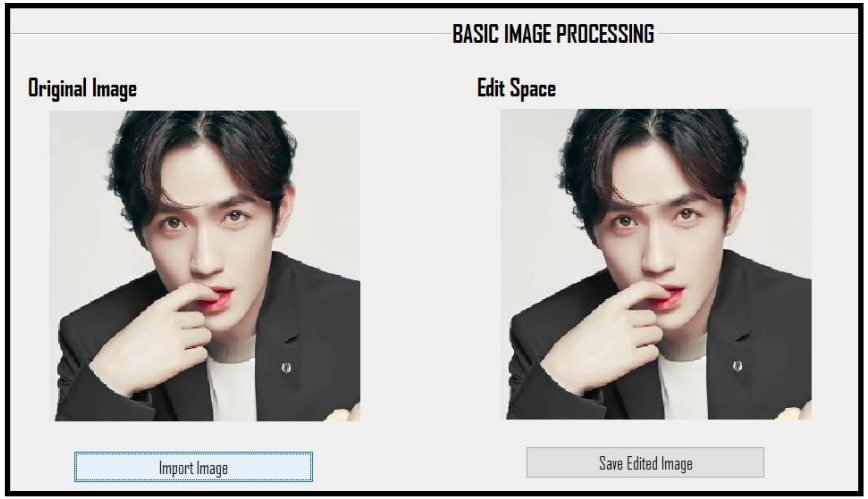
1. Import an Image.
2. Click an Operation you wish to perform.
3. Click to join at least 3 points on the Edit Space.
4. Double click the selected region. The effect will appear on the Edit Space.
5. Continue as you like. Save the Edited Image.

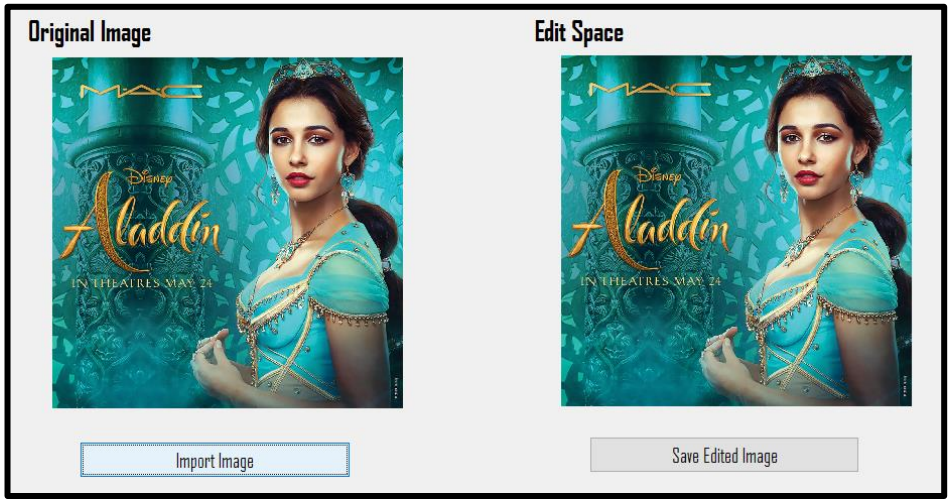
Load Image



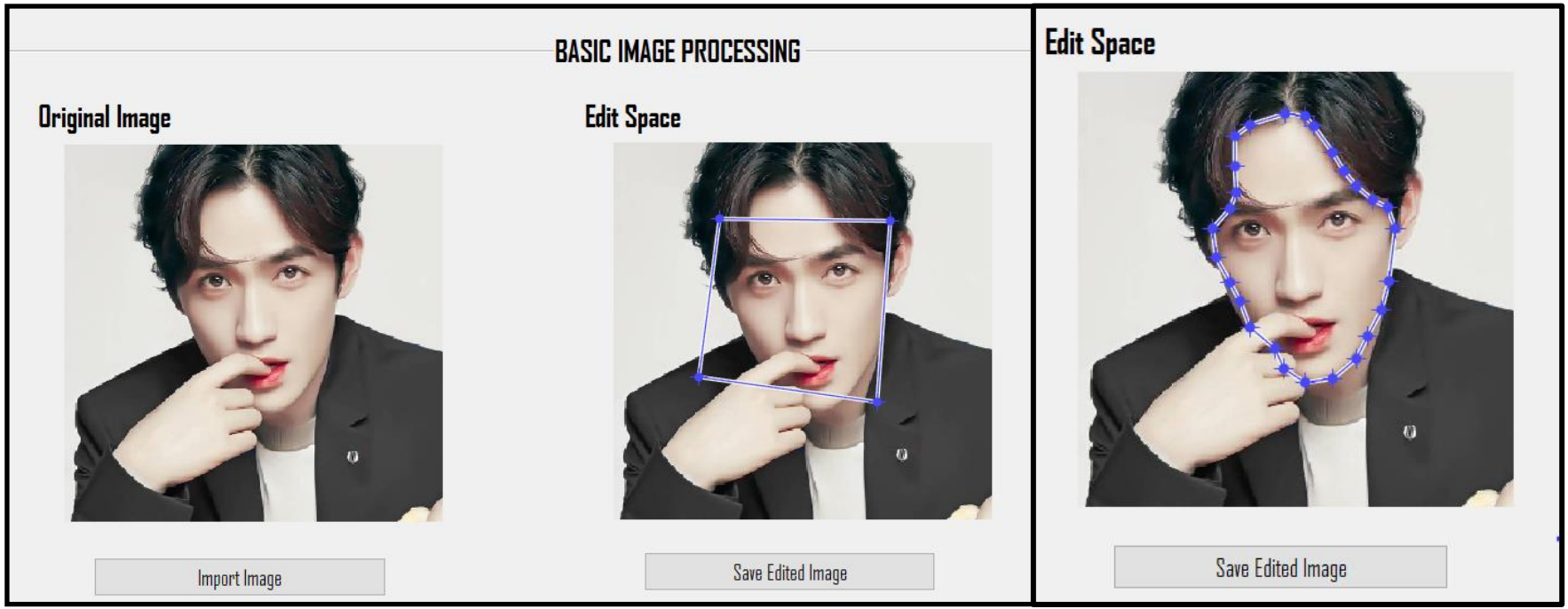
The application allows user to load any images from the computer.

Image loaded in both axes

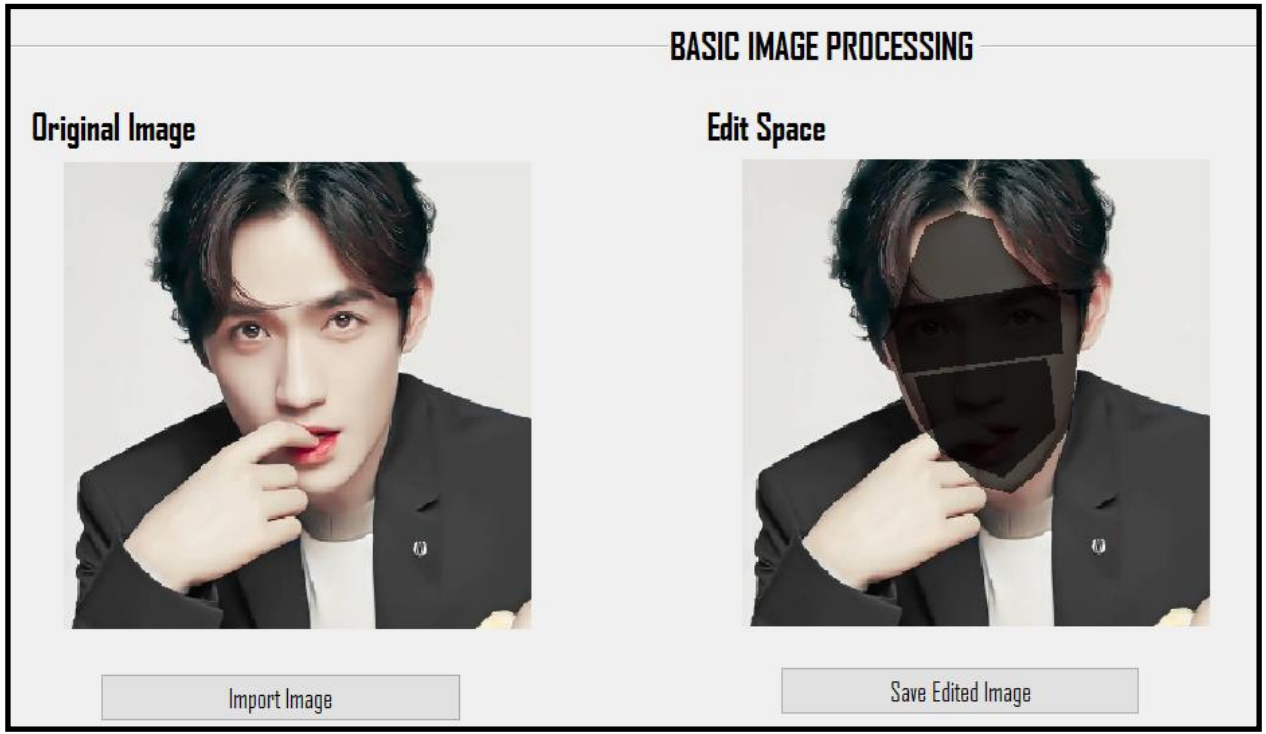




Choose part - in the Edit Space

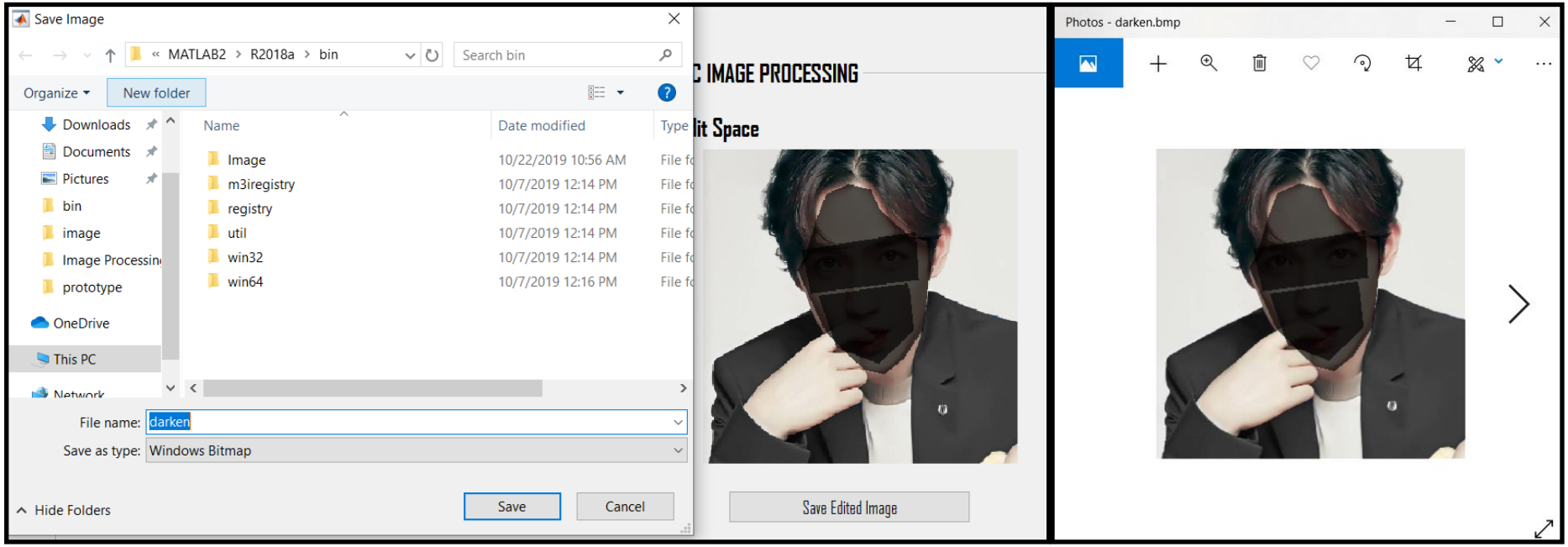


Darken Image

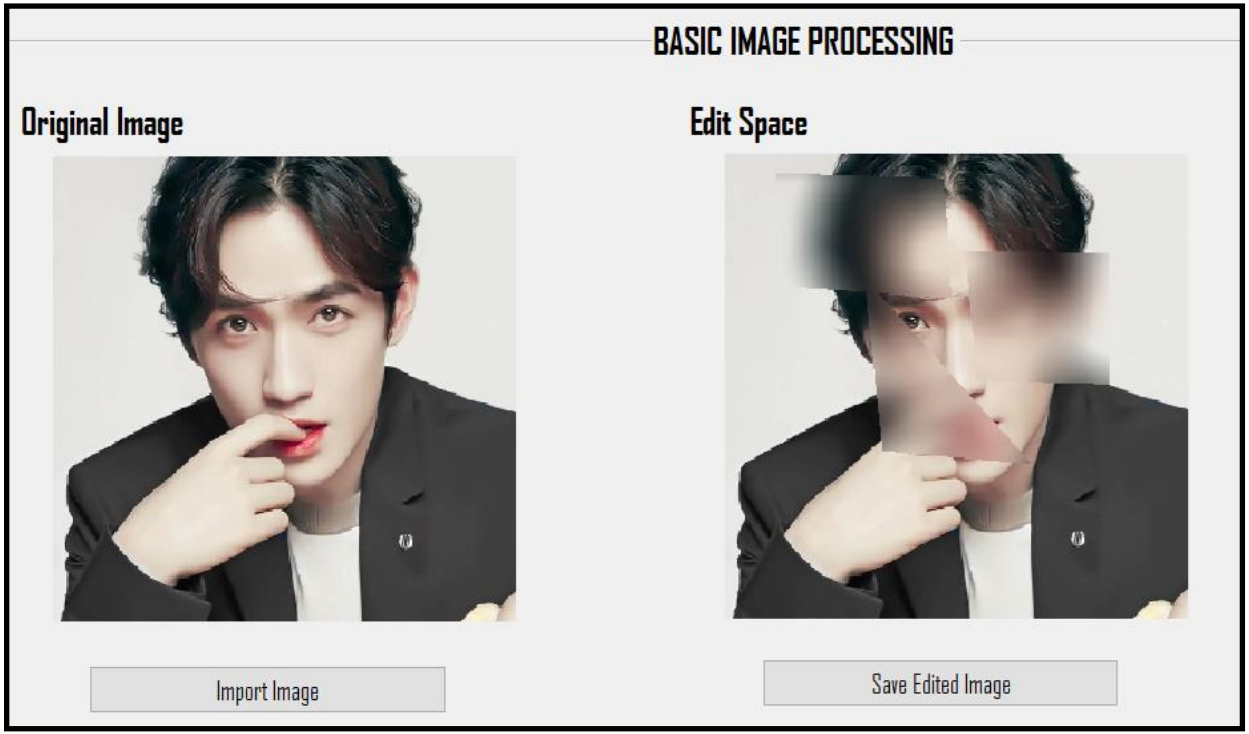


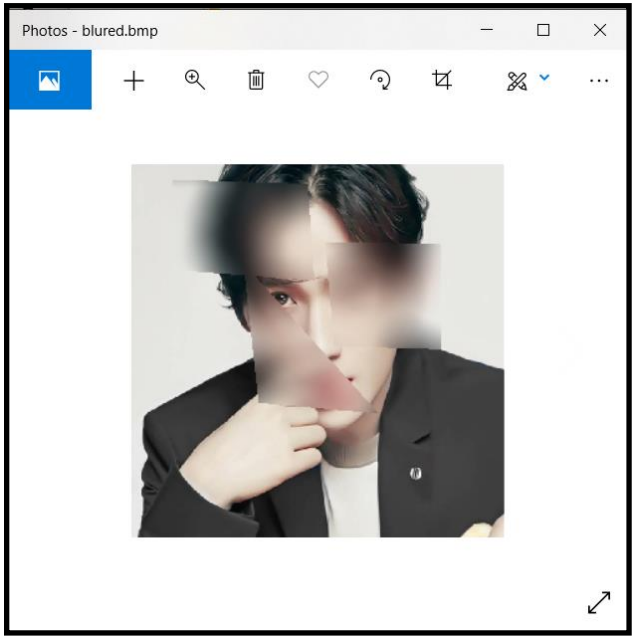
The application allows user to choose parts in the image multiple times.

Save Output Image

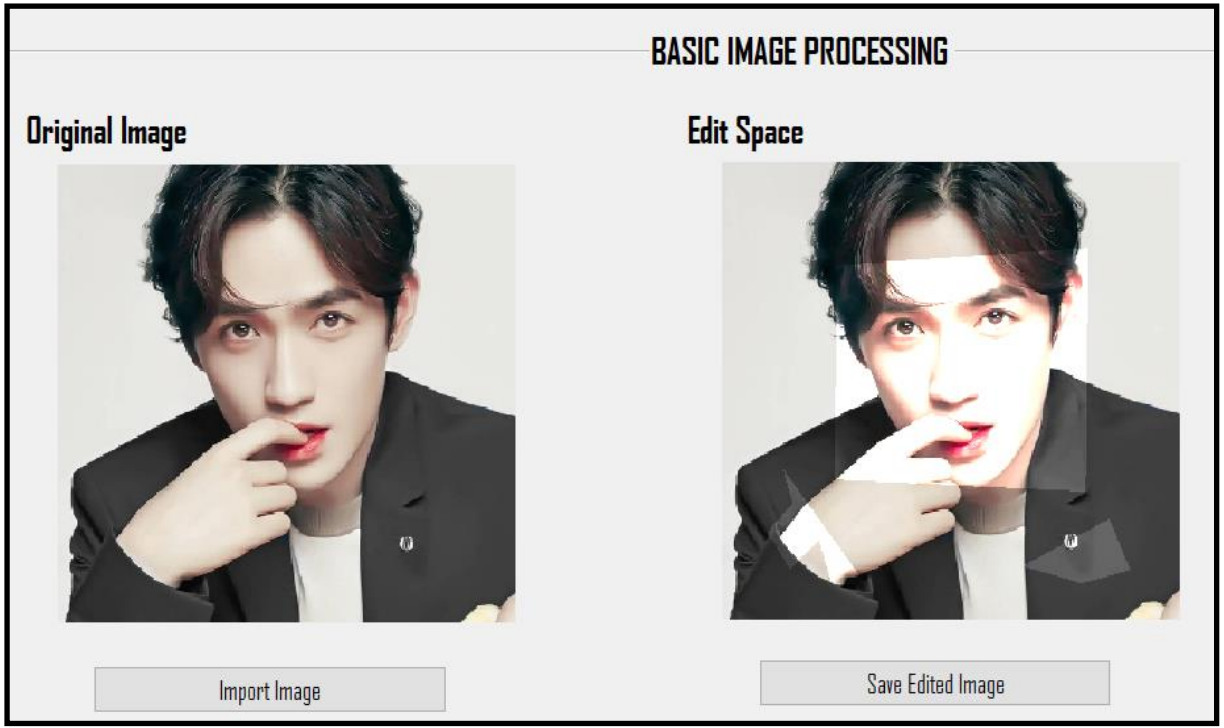


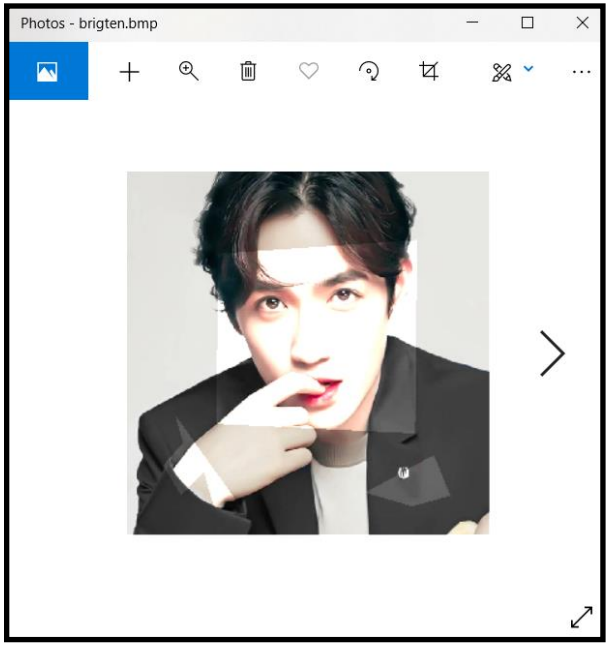
Blur Image


BASIC IMAGE PROCESSING
Original Image
Edit Space
Import Image
Save Edited Image

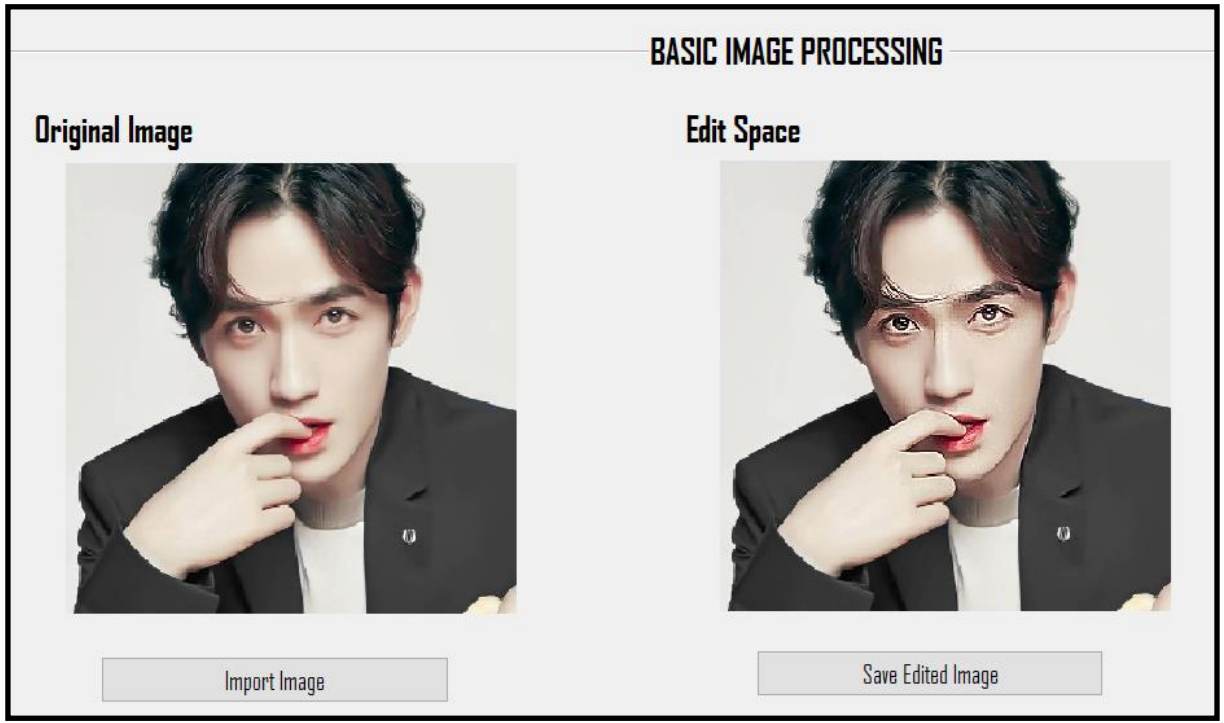

Photos - blured.bmp

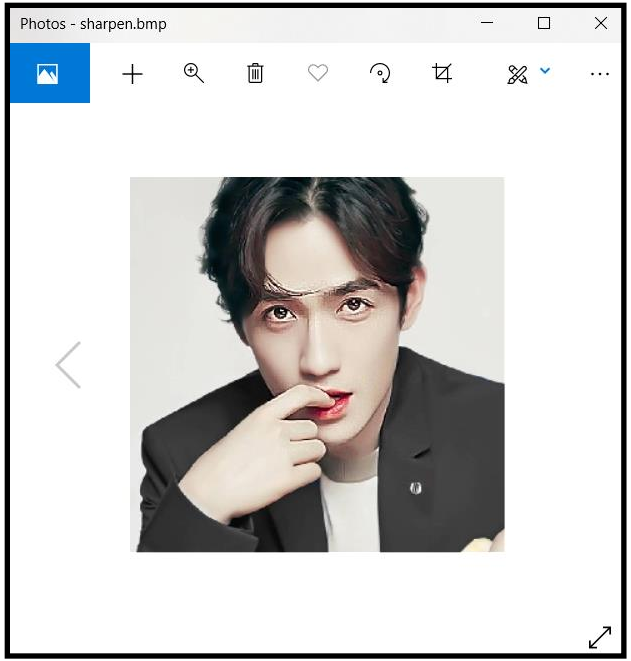
Brighten Image





Sharpen Image





Other sample output