Matlab code to determine the age of an individual by Justicetam

function varargout = Age\_Classification\_Voting\_System(varargin)

% AGE\_CLASSIFICATION\_VOTING\_SYSTEM MATLAB code for Age\_Classification\_Voting\_System.fig

% AGE\_CLASSIFICATION\_VOTING\_SYSTEM, by itself, creates a new AGE\_CLASSIFICATION\_VOTING\_SYSTEM or raises the existing

% singleton\*.

%

% H = AGE\_CLASSIFICATION\_VOTING\_SYSTEM returns the handle to a new AGE\_CLASSIFICATION\_VOTING\_SYSTEM or the handle to

% the existing singleton\*.

%

% AGE\_CLASSIFICATION\_VOTING\_SYSTEM('CALLBACK',hObject,eventData,handles,...) calls the local

% function named CALLBACK in AGE\_CLASSIFICATION\_VOTING\_SYSTEM.M with the given input arguments.

%

% AGE\_CLASSIFICATION\_VOTING\_SYSTEM('Property','Value',...) creates a new AGE\_CLASSIFICATION\_VOTING\_SYSTEM or raises the

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

% applied to the GUI before Age\_Classification\_Voting\_System\_OpeningFcn gets called. An

% unrecognized property name or invalid value makes property application

% stop. All inputs are passed to Age\_Classification\_Voting\_System\_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 Age\_Classification\_Voting\_System

% Last Modified by GUIDE v2.5 28-Sep-2021 12:32:26

% Begin initialization code - DO NOT EDIT

gui\_Singleton = 1;

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

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

'gui\_OpeningFcn', @Age\_Classification\_Voting\_System\_OpeningFcn, ...

'gui\_OutputFcn', @Age\_Classification\_Voting\_System\_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 Age\_Classification\_Voting\_System is made visible.

function Age\_Classification\_Voting\_System\_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 Age\_Classification\_Voting\_System (see VARARGIN)

% Choose default command line output for Age\_Classification\_Voting\_System

handles.output = hObject;

ah = axes('unit', 'normalized', 'position', [0 0 1 1]);

% Move the background axes to the bottom

uistack(ah, 'bottom');

% Load in a background image and display it using the correct colors

% The image used below, is in the Image Processing Toolbox. If you do not have %access to this toolbox, you can use Gender\_and\_Age\_Prediction\_System image file instead.

I=imread('example5.jpg');

hi = imagesc(I)

colormap gray

% Turn the handlevisibility off so that we don’t inadvertently plot into the axes again

% Also, make the axes invisible

set(ah,'handlevisibility','off','visible','off')

% Update handles structure

% Update handles structure

% Update handles structure

guidata(hObject, handles);

% UIWAIT makes Age\_Classification\_Voting\_System wait for user response (see UIRESUME)

% uiwait(handles.figure1);

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

function varargout = Age\_Classification\_Voting\_System\_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 load\_training\_data.

function load\_training\_data\_Callback(hObject, eventdata, handles)

% hObject handle to load\_training\_data (see GCBO)

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

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

msgbox('TRAINING STARTED ...PLEASE WAIT .....')

global DF

DF=[]

for i=1:10

str=int2str(i);

str=strcat(str,'.jpg')

a=imread(str);

[m n c]=size(a);

if c==3

b=rgb2gray(a);

else

b=a;

end

im =b;

faceDetector = vision.CascadeObjectDetector();

videoFrame = im;

bbox = step(faceDetector, videoFrame);

videoOut = insertObjectAnnotation(a,'rectangle',bbox,'Face');

cro=imcrop(a,bbox(1,:));

cro=rgb2gray(cro);

[AA H V D]=dwt2(cro,'db1');;

I=imresize(AA,[256 256]);

ii=1:8:256

[a1,b1]=ndgrid(ii,ii)

M=arrayfun(@(x,y) I(x:x+7,y:y+7),a1,b1,'un',0);

for i=1:32

for j=1:32

blk=M{i,j};

end

end

Input\_Im=I;

R=1;

if size(Input\_Im, 3) == 3

Input\_Im = rgb2gray(Input\_Im);

end

L = 2\*R + 1;

C = round(L/2);

Input\_Im = uint8(Input\_Im);

row\_max = size(Input\_Im,1)-L+1;

col\_max = size(Input\_Im,2)-L+1;

LBP\_Im = zeros(row\_max, col\_max);

for i = 1:row\_max

for j = 1:col\_max

A = Input\_Im(i:i+L-1, j:j+L-1);

A = A-A(C,C);

A(A>0) = 1;

LBP1(i,j) = A(C,L) + A(L,L)\*2 + A(L,C)\*4 + A(L,1)\*8 + A(C,1)\*16 + A(1,1)\*32 + A(1,C)\*64 + A(1,L)\*128;

end

end

s2 = regionprops(LBP1,'Eccentricity')

s2=s2.Eccentricity;

s3 = regionprops(LBP1,'Orientation')

s3=s3.Orientation;

s4 =regionprops(LBP1,'Perimeter')

s4=s4.Perimeter;

s5 =regionprops(LBP1,'Solidity')

s5=s5.Solidity;

med=median(median(LBP1));

[m n c]=size(LBP1);

for i=1:m

for j=1:n

if LBP1(i,j)>med ;

sel(i,j)=1;

else

sel(i,j)=LBP1(i,j);

end

end

end

sel(sel==0)=[1]

feat=[sel(1:20) s2 s3 s4 s5];

DF=[DF ; feat]

end

cd ..

msgbox('TRAINING FINISHED ...PLEASE RESTART APPLICATION TO START CLASSIFICATION .....')

% --- Executes on button press in browse\_input\_image.

function browse\_input\_image\_Callback(hObject, eventdata, handles)

% hObject handle to browse\_input\_image (see GCBO)

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

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

global videoOut

global DF

global cro

global AA

global H

global V

global D

global LBP1

global class

global out

insert = uigetfile('\*.jpg')

in=imread(insert);

img=in

faceDetector = vision.CascadeObjectDetector();

videoFrame = img;

bbox = step(faceDetector, videoFrame);

videoOut = insertObjectAnnotation(img,'rectangle',bbox,'Face');

cro=imcrop(img,bbox(1,:));

cro=rgb2gray(cro);

[AA H V D]=dwt2(cro,'db1');;

I=imresize(AA,[256 256]);

ii=1:8:256

[a1,b1]=ndgrid(ii,ii)

M=arrayfun(@(x,y) I(x:x+7,y:y+7),a1,b1,'un',0);

for i=1:32

for j=1:32

blk=M{i,j};

end

end

Input\_Im=I;

R=1;

if size(Input\_Im, 3) == 3

Input\_Im = rgb2gray(Input\_Im);

end

L = 2\*R + 1;

C = round(L/2);

Input\_Im = uint8(Input\_Im);

row\_max = size(Input\_Im,1)-L+1;

col\_max = size(Input\_Im,2)-L+1;

LBP\_Im = zeros(row\_max, col\_max);

for i = 1:row\_max

for j = 1:col\_max

A = Input\_Im(i:i+L-1, j:j+L-1);

A = A-A(C,C);

A(A>0) = 1;

LBP1(i,j) = A(C,L) + A(L,L)\*2 + A(L,C)\*4 + A(L,1)\*8 + A(C,1)\*16 + A(1,1)\*32 + A(1,C)\*64 + A(1,L)\*128;

end

end

% % shape based feature

s2 = regionprops(LBP1,'Eccentricity')

s2=s2.Eccentricity;

s3 = regionprops(LBP1,'Orientation')

s3=s3.Orientation;

s4 =regionprops(LBP1,'Perimeter')

s4=s4.Perimeter;

s5 =regionprops(LBP1,'Solidity')

s5=s5.Solidity;

med=median(median(LBP1));

[m n c]=size(LBP1);

for i=1:m

for j=1:n

if LBP1(i,j)>med ;

sel(i,j)=0;

else

sel(i,j)=LBP1(i,j);

end

end

end

sel(sel==0)=[1]

QF=[sel(1:20) s2 s3 s4 s5];

% % % classfier

train=DF;

xdata =train;

TrainingSet=double(xdata);

GroupTrain=[1;2;3;4;5;6;7;8;9;10]

TestSet=double(QF);

u=unique(GroupTrain);

numClasses=length(u);

result = zeros(length(TestSet(:,1)),1);

for k=1:numClasses

G1vAll=(GroupTrain==u(k));

models(k) = svmtrain(TrainingSet,G1vAll);

end

for j=1:size(TestSet,1)

for k=1:numClasses

if(svmclassify(models(k),TestSet(j,:)))

break;

end

end

result(j) = k;

end

out=result

axes(handles.axes1);

%

handles.InputImage=img;

imshow(img),hold on

title('Input Image')

% --- Executes on button press in detect\_face.

function detect\_face\_Callback(hObject, eventdata, handles)

% hObject handle to detect\_face (see GCBO)

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

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

global videoOut

axes(handles.axes2);

%

handles.InputImage=videoOut;

imshow(videoOut),hold on

title('Face detection')

% --- Executes on button press in detect\_age.

function detect\_age\_Callback(hObject, eventdata, handles)

% hObject handle to detect\_age (see GCBO)

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

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

global out

if out==1

set(handles.edit1,'String','0-17')

elseif out==2

set(handles.edit1,'String','0-17')

elseif out==3

set(handles.edit1,'String','0-17')

elseif out==4

set(handles.edit1,'String','0-17')

elseif out==5

set(handles.edit1,'String','18-45')

elseif out==6

set(handles.edit1,'String','18-45')

elseif out==7

set(handles.edit1,'String','18-45')

elseif out==8

set(handles.edit1,'String','18-45')

elseif out==9

set(handles.edit1,'String','46 and above')

elseif out==10

set(handles.edit1,'String','46 and above')

end

% --- Executes on button press in eligibility\_status.

function eligibility\_status\_Callback(hObject, eventdata, handles)

% hObject handle to eligibility\_status (see GCBO)

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

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

global out

if out==1

set(handles.edit2,'String','Not Eligible')

elseif out==2

set(handles.edit2,'String','Not Eligible')

elseif out==3

set(handles.edit2,'String','Not Eligible')

elseif out==4

set(handles.edit2,'String','Not Eligible')

elseif out==5

set(handles.edit2,'String','Eligible')

elseif out==6

set(handles.edit2,'String','Eligible')

elseif out==7

set(handles.edit2,'String','Eligible')

elseif out==8

set(handles.edit2,'String','Eligible')

elseif out==9

set(handles.edit2,'String','Eligible')

elseif out==10

set(handles.edit2,'String','Eligible')

end

% --- Executes on button press in new\_voter.

function new\_voter\_Callback(hObject, eventdata, handles)

% hObject handle to new\_voter (see GCBO)

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

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

Capture\_Voter\_Application;

% --- Executes on button press in clear\_screen.

function clear\_screen\_Callback(hObject, eventdata, handles)

% hObject handle to clear\_screen (see GCBO)

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

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

cla(handles.axes1,'reset')

cla(handles.axes1)

cla(handles.axes2,'reset')

cla(handles.axes2)

set([ handles.edit1, handles.edit2], 'String','');

function edit1\_Callback(hObject, eventdata, handles)

% hObject handle to edit1 (see GCBO)

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

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

% Hints: get(hObject,'String') returns contents of edit1 as text

% str2double(get(hObject,'String')) returns contents of edit1 as a double

% --- Executes during object creation, after setting all properties.

function edit1\_CreateFcn(hObject, eventdata, handles)

% hObject handle to edit1 (see GCBO)

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

% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.

% See ISPC and COMPUTER.

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit2\_Callback(hObject, eventdata, handles)

% hObject handle to edit2 (see GCBO)

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

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

% Hints: get(hObject,'String') returns contents of edit2 as text

% str2double(get(hObject,'String')) returns contents of edit2 as a double

% --- Executes during object creation, after setting all properties.

function edit2\_CreateFcn(hObject, eventdata, handles)

% hObject handle to edit2 (see GCBO)

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

% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.

% See ISPC and COMPUTER.

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end