```
function varargout = detec_alarme(varargin)

% Last Modified by GUIDE v2.5 17-Dec-2019 17:03:34
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @detec_alarme_OpeningFcn, ...
                   'gui_OutputFcn',  @detec_alarme_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 detec_alarme is made visible.
function detec_alarme_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = detec_alarme_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;
handles.logo1 = importdata ('logo2.jpg');
axes(handles.axes1);
imshow(handles.logo1);
text(10,20,'Detec Dif','FontSize',20,'Color',[0.5 0.2 0.3])
text(480,97,'Versão beta','FontSize',6,'Color','black')
handles.end_prov=cd; % grava a pasta do software
handles.con=0;
handles.bin=0;
handles.dims=0;
handles.prov=0;
guidata(hObject, handles);

function img_in_1_Callback(hObject, eventdata, handles)

[hObject,handles]=open_img(hObject,handles,1);
handles.img1=handles.img;
handles.mask=handles.img1; handles.mask(handles.mask~=0)==1;
guidata(hObject, handles);
set(handles.edit_img_in_1, 'String', handles.Path);
assignin('base','name',handles.info);

function edit_img_in_1_Callback(hObject, eventdata, handles)
user_endereco_img = get(hObject,'String'); % returns contents of
edit_img_in as text
```

```
handles.img1 = importdata (user_endereco_img); % Lê imagem
handles.mask=handles.img1; handles.mask(handles.mask~=0)==1;
guidata(hObject,handles)

function edit_img_in_1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function img_in_2_Callback(hObject, eventdata, handles)
[hObject,handles]=open_img(hObject,handles,1);
handles.img2=handles.img;
guidata(hObject, handles);
set(handles.edit_img_in_2, 'String', handles.Path);

function edit_img_in_2_Callback(hObject, eventdata, handles)
user_endereco_img = get(hObject,'String'); % returns contents of
edit_img_in as text
handles.img2 = importdata (user_endereco_img); % Lê imagem
guidata(hObject,handles)

function edit_img_in_2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Outputs from this function are returned to the command line.
function varargout = calibracao_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
handles.metricdata.logo1 = importdata ('logo2.jpg');
axes(handles.axes1);
imshow(handles.metricdata.logo1);
text(10,20,'OCP CalibRad','FontSize',20,'Color',[0.5 0.2 0.3])
text(480,97,'Versão beta','FontSize',6,'Color','black')
handles.end_prov=cd; % grava a pasta do software
guidata(hObject,handles)
% % img entrada
function img_in_Callback(hObject, eventdata, handles)

[hObject,handles]=open_img(hObject,handles,1);
guidata(hObject, handles);

% escreve em caixa de texto o endereço
    set(handles.edit_img_in, 'String', handles.Path);
% caixas de texto endereço img in

function edit_img_in_Callback(hObject, eventdata, handles)
user_endereco_img = get(hObject,'String'); % returns contents of
edit_img_in as text
handles.img = importdata (user_endereco_img); % Lê imagem
guidata(hObject,handles)

function edit_img_in_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% img saida

function img_out_Callback(hObject, eventdata, handles)

if isfield(handles,'Path')==1 % para voltar à pasta onde foi selecionada
a img de entrada

    cd(handles.PathName); % Pasta da imagem aberta
    [handles.FileName_salva,handles.PathName_salva] =
uiputfile('*.tif','Localize a pasta e indique o nome genérico do
arquivo');
    cd(handles.end_prov) % retorna a pasta do software
    [~,handles.FileName_salva,ext]=fileparts(handles.FileName_salva);

handles.Pathsalva=strcat(handles.PathName_salva,handles.FileName_salva,ex
t);
    set(handles.edit_img_out, 'String', handles.Pathsalva);
    guidata(hObject,handles)
else
    [handles.FileName_salva,handles.PathName_salva] =
uiputfile('*.tif','Localize a pasta e indique o nome genérico do
arquivo');
    [~,handles.FileName_salva,ext]=fileparts(handles.FileName_salva);

handles.Pathsalva=strcat(handles.PathName_salva,handles.FileName_salva,ex
t);
    set(handles.edit_img_out, 'String', handles.Pathsalva);
    guidata(hObject,handles)
end

function edit_img_out_Callback(hObject, eventdata, handles)
user_endereco_img = get(hObject,'String'); % returns contents of
edit_img_in as text
[handles.PathName_salva,handles.FileName_salva]=fileparts(user_endereco_i
mg); handles.PathName_salva=strcat(handles.PathName_salva,'\');
guidata(hObject,handles)

function edit_img_out_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in norm.
function norm_Callback(hObject, eventdata, handles)
handles.img11=double(handles.img1/255); % passa para double e converte
para uint8
handles.img22=double(handles.img2/255);
handles.img_dif_1_2=handles.img11-handles.img22;
idx=handles.img_dif_1_2;
idx(idx==0)=NaN;
```

```
figure, hist(idx(:),512);
prompt = {'Defina o valor do deslocamento:'};
dlgtitle = 'Input';
handles.dims = fitdist(idx(:),'Normal'); %Encontra a média pela
aproximação gaussiana

definput = {num2str(handles.dims.mu*-1)}; %Torna o valor negativo para
deslocar
answer = inputdlg(prompt,dlgtitle,1,definput);
handles.dims=str2num(cell2mat(answer));
handles.img_dif_1_2=handles.img_dif_1_2+handles.dims;
handles.img_dif_1_2(handles.img_dif_1_2==handles.dims)=0;
figure, hist(handles.img_dif_1_2(:),512);
myicon = imread('landOcean.jpg');
msgbox('Procedimento realizado com
sucesso!','Andamento','custom',myicon);
guidata(hObject,handles)

% --- Executes on button press in lim.
function lim_Callback(hObject, eventdata, handles)
handles.img11=double(handles.img1/255); % passa para double e converte
para uint8
handles.img22=double(handles.img2/255);
handles.img_dif_1_2=handles.img11-handles.img22;
handles.img_dif_1_2=handles.img_dif_1_2+handles.dims;
handles.img_dif_1_2(handles.img_dif_1_2==handles.dims)=0;

cor=jet; cor(31:33,:)=1;
cor(64,:)=0;
handles.img_dif_1_2(handles.mask==0)=70;
figure, imshow(handles.img_dif_1_2,[-50 50],'Colormap',cor);  colorbar;
handles.sl = uicontrol('style','slide','unit','pix','position',[20 10 260
30],'min',1,'max',10,'val',1,...
                'sliderstep',[1/20 1/20],'callback',{@sl_call,handles});
annotation('textbox', [0, 0.5, 0.1, 0.1], 'string', strcat('Lim =
',num2str(0)));
guidata(hObject,handles)

function [] = sl_call(varargin)
% Callback for the slider.
[hObject,handles] = varargin{[1,3]};  % calling handle and data
structure.
handles.lim=get(hObject,'value');
assignin('base','V',handles.lim);
cor=jet; cor(round(32-handles.lim/2):round(32+handles.lim/2),:)=1;
cor(64,:)=0;
handles.img_dif_1_2(handles.mask==0)=70;

imshow(handles.img_dif_1_2,[-50 50],'Colormap',cor);  colorbar;
assignin('base','name',handles.img_dif_1_2);
handles.sl = uicontrol('style','slide','unit','pix','position',[20 10 260
30],'min',0,'max',50,'val',handles.lim,...
                'sliderstep',[1/20 1/20],'callback',{@sl_call,handles});
annotation('textbox', [0, 0.5, 0.1, 0.1], 'string', strcat('Lim =
',num2str(round(handles.lim))));
```

```
% Filtragem

function v1_Callback(hObject, eventdata, handles)
handles.v1=str2double(get(hObject,'String'));
guidata(hObject,handles)
function v1_CreateFcn(hObject, eventdata, handles); if ispc &&
isequal(get(hObject,'BackgroundColor'),...
        get(0,'defaultUicontrolBackgroundColor'));
set(hObject,'BackgroundColor','white'); end

function filt_eo_Callback(hObject, eventdata, handles)
handles.lim = evalin('base', 'V');
handles.img_dif_1_2(handles.img_dif_1_2<handles.lim &
handles.img_dif_1_2>-handles.lim)=0;
se = strel('disk',handles.v1);
handles.img_dif_1_2 = imopen(handles.img_dif_1_2,se);
cor=jet; cor(round(33-handles.lim/2):round(30+handles.lim/2),:)=1;
cor(64,:)=0;
handles.img_dif_1_2(handles.mask==0)=70;
figure, imshow(handles.img_dif_1_2,[-50 50],'Colormap',cor); colorbar;
myicon = imread('landOcean.jpg');
msgbox('Procedimento realizado com
sucesso!','Andamento','custom',myicon);
guidata(hObject,handles)

% Salva mapas prov .tif
function salv_p_Callback(hObject, eventdata, handles)

handles.lim = evalin('base', 'V')
handles.img_dif_1_2(handles.img_dif_1_2<handles.lim &
handles.img_dif_1_2>-handles.lim)=0;
handles.prov=handles.prov+1;
guidata(hObject, handles);
if handles.con==1
nome=strcat(handles.PathName_salva,handles.FileName_salva,'_prov_',num2st
r(handles.prov),'_lim_',num2str(round(handles.lim)),'_con.tif');
salva_img(handles.R, handles.S,
handles.info,handles.infogeo,nome,abs(handles.img_dif_1_2));
end

if handles.bin==1
    handles.img_dif_1_2(handles.img_dif_1_2<handles.lim &
handles.img_dif_1_2>-handles.lim)=0;

handles.img_dif_1_2(handles.img_dif_1_2==0)=255;
handles.img_dif_1_2(handles.img_dif_1_2~=255)=0;
nome=strcat(handles.PathName_salva,handles.FileName_salva,'_prov_',num2st
r(handles.prov),'_lim_',num2str(round(handles.lim)),'_bin.tif');
salva_img(handles.R, handles.S,
handles.info,handles.infogeo,nome,handles.img_dif_1_2);
myicon = imread('landOcean.jpg');
msgbox('Salvo com sucesso!','Andamento','custom',myicon);
end
```

```
% --- Executes on button press in ok.
function ok_Callback(hObject, eventdata, handles)
handles.img_dif_1_2(handles.img_dif_1_2<handles.lim &
handles.img_dif_1_2>-handles.lim)=0;
% Salva mapas finais .tif
if handles.con==1

nome=strcat(handles.PathName_salva,handles.FileName_salva,'_con.tif');
salva_img(handles.R, handles.S,
handles.info,handles.infogeo,nome,abs(handles.img_dif_1_2));
end

if handles.bin==1
    handles.img_dif_1_2(handles.img_dif_1_2<handles.lim &
handles.img_dif_1_2>-handles.lim)=0;
handles.img_dif_1_2(handles.img_dif_1_2==0)=255;
handles.img_dif_1_2(handles.img_dif_1_2~=255)=0;
nome=strcat(handles.PathName_salva,handles.FileName_salva,'_bin.tif');
salva_img(handles.R, handles.S,
handles.info,handles.infogeo,nome,handles.img_dif_1_2);
myicon = imread('landOcean.jpg');
msgbox('Procedimento realizado com
sucesso!','Andamento','custom',myicon);
end

function cancela_Callback(hObject, eventdata, handles)
close;

function bin_Callback(hObject, eventdata, handles)
if get(hObject,'Value')==1 handles.bin=1;else handles.bin=0; end;
guidata(hObject,handles)

function con_Callback(hObject, eventdata, handles)
if get(hObject,'Value')==1 handles.con=1;else handles.con=0; end;
guidata(hObject,handles)
```