

LAMPIRAN

```
function varargout = otak(varargin)
% OTAK M-file for otak.fig
%   OTAK, by itself, creates a new OTAK or raises the existing
%   singleton*.
%
%   H = OTAK returns the handle to a new OTAK or the handle to
%   the existing singleton*.
%
%   OTAK('CALLBACK',hObject,eventData,handles,...) calls the local
%   function named CALLBACK in OTAK.M with the given input arguments.
%
%   OTAK('Property','Value',...) creates a new OTAK or raises the
%   existing singleton*. Starting from the left, property value pairs are
%   applied to the GUI before otak_OpeningFcn gets called. An
%   unrecognized property name or invalid value makes property application
%   stop. All inputs are passed to otak_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 otak

% Last Modified by GUIDE v2.5 28-Feb-2017 17:19:54

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @otak_OpeningFcn, ...
                  'gui_OutputFcn', @otak_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
% End initialization code - DO NOT EDIT

% --- Executes just before otak is made visible.
function otak_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 otak (see VARARGIN)

% Choose default command line output for otak
handles.output = hObject;
axes(handles.axes6);
imshow('logo.jpg')

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes otak wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% Create background axes and move them to the background
hback = axes('units','normalized','position',[0 0 1 1]);
uistack(hback,'bottom');
% Load background image and display it
[back map]=imread('logo5.jpg');
image(back)
colormap(map)

% Turn the handlevisibility off so that we don't inadvertently plot into
% the axes again. Also, make the axes invisible
set(hback,'handlevisibility','off','visible','off')
% --- Outputs from this function are returned to the command line.
function varargout = otak_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)
proyek=guidata(gcbo);
[namafile,direktori]=uigetfile({'*.bmp';'*.*'},'Buka Gambar');
i=imread(strcat(direktori,namafile));
set(proyek.figure1,'CurrentAxes',proyek.axes1);
set(imshow(i));
set(proyek.axes1,'Userdata',i);
set(proyek.figure1,'Userdata',i);

% --- 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)
proyek=guidata(gcbo);
g1=rgb2gray(i1); %(untuk mengubah foto warna menjadi grayscale)
t1=graythresh(g1); %(Untuk mempertajam citra biner)
a1=im2bw(g1,t1); %(untuk menghasilkan perhitungan grayscale dan citra biner)
l1 = +a1
m1 = [0 0 -1 0 0; 0 -1 -2 -1 0; -1 -2 16 -2 -1; 0 -1 -2 -1 0; 0 0 -1 0 0] % (rumus untuk
merapihkan tepi menghilangkan noise)

e1=conv2(l1,m1); %(untuk hasil perhitungan menggunakan log)
s1=imfill(e1,'holes'); %untuk memperjelas citra biner(memfilter)
n1=imclearborder(s1,18); %menghilangkan border yang menempel pada citra
seD = strel ('diamond',1);
d1 = imerode (n1, seD); %mengkomposisikan struktur element gambar
res1=imresize(d1,.1); %mengubah ukuran citra
pix2=bwarea(res1)/10^4 %menghitung jumlah piksel area
error toleransi=(pix2*0.05) %/10^4(mencari pixel 2 yang atas)
pix1=(pix2-error toleransi); %(menentukan kanker kulit sehat dan sakit)

i=get(proyek.axes1,'Userdata');
g=rgb2gray(i);
t=graythresh(g);
a=im2bw(g,t);
l = +a
m = [0 0 -1 0 0; 0 -1 -3 -1 0; -1 -2 16 -3 -1; 0 -1 -3 -1 0;0 0 -1 0 0 ]
e=conv2(l,m);

```

```

s=imfill(e,'holes');
n=imclearborder(s,1);
seD = strel ('diamond',1);
d = imerode (n,seD);
res=imresize(d,.1);
pix=bwarea(res)/10^4
hasil=pix;

set(proyek.figure1,'CurrentAxes',proyek.axes2);
set(imshow(a));
set(proyek.edit1,'string',hasil);
set(proyek.edit8,'string',pix);
set(proyek.edit7,'string',pix1);

set(proyek.axes1,'Userdata');
if pix<pix1,set(proyek.edit1,'String','Emergency !')
else set(proyek.edit1,'String','Ada Pendarahan'),end;

set(proyek.axes1,'Userdata');
if pix<pix1,set(proyek.edit2,'String','Perlu Segera Dilakukan Operasi !')
else set(proyek.edit2,'String','Perlu Dilakukan Pengobatan Rutin Dan Baiknya Di
Lakukan Operasi'),end;

% --- Executes during object creation, after setting all properties.
function axes1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to axes1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate axes1

% --- 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)

cla(handles.axes1);
cla(handles.axes2);
set(handles.edit1,'string','');
set(handles.edit2,'string','');
set(handles.edit7,'string','');
set(handles.edit8,'string','');

```

```
% --- Executes during object creation, after setting all properties.
function axes2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to axes2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate axes2
```

```
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
```

```
function edit6_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 edit6_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
```

```
% --- 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)
```

```
button = questdlg('Yakin Mau Menutup program ??', ...
    'Tutup','Iya','Tidak','Tidak');
```

```
switch button
    case 'Iya',
```

```

        diso.p('Close');
        %Save variables to matlab.mat
        %save
        exit
    case 'Tidak',
        quit cancel;
end

```

```

function edit7_Callback(hObject, eventdata, handles)
% hObject   handle to edit7 (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 edit7 as text
%        str2double(get(hObject,'String')) returns contents of edit7 as a double

```

```

% --- Executes during object creation, after setting all properties.
function edit7_CreateFcn(hObject, eventdata, handles)
% hObject   handle to edit7 (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 edit8_Callback(hObject, eventdata, handles)
% hObject   handle to edit8 (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 edit8 as text
%        str2double(get(hObject,'String')) returns contents of edit8 as a double

```

```

% --- Executes during object creation, after setting all properties.
function edit8_CreateFcn(hObject, eventdata, handles)
% hObject   handle to edit8 (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
```