DAFTAR LAMPIRAN

CODING PENGOLAHAN JST

```
clc; clear; close all; warning off all;

% membaca citra background
Back_asli = imread('C:\Users\BASIT\Dropbox\elangputih\Tesis
Basit\Program 3\background.JPG');
Back = rgb2gray(Back_asli);

% menetapkan jumlah data untuk pelatihan
jumlah_data = 25;

%%% DEPAN
a = zeros(jumlah_data,1);
t = zeros(jumlah_data,1);
max_a = 606;
max_t = 2466;

for n = 1:jumlah_data
    % membaca file citra
    Img = imread(['C:\Users\BASIT\Dropbox\elangputih\Tesis
Basit\Program 3\' num2str(n),' depan.JPG']);

    % melakukan rotasi citra dengan sudut rotasi sebesar 90
        Img = imrotate(Img,90);
        R = Img(:,:,1);
        G = Img(:,:,2);
        B = Img(:,:,3);
        R2 = Back_asli(:,:,1);
        G2 = Back_asli(:,:,2);
        B2 = Back_asli(:,:,3);
    % mengkonversi citra rgb menjadi citra grayscale
    Img_gray = rgb2gray(Img);
        % melakukan subtraksi dengan background dalam kelas double
    Subtraction = (double(Img_gray)-double(Back));
    Subtraction0 = Subtraction;
    % mengembalikan kelas citra ke dalam kelas uint8
    Min_S = min(Subtraction(:));
    Max_S = max(Subtraction(:));
    Subtraction = ((Subtraction-Min_S)/(Max_S-Min_S))*255;
    Subtraction = uint8(Subtraction);
    Subtraction1 = Subtraction;
    % melakukan thresholding dengan nilai threshold 0.4
    Subtraction = im2bw(Subtraction,.4);
    bw0 = Subtraction;
    % melakukan komplemen
    bw = ~Subtraction;
    bw1 = bw;
    % melakukan area opening untuk menghilangkan noise
    bw = bwareaopen(bw,1000);
    % mendeteksi bounding box
    [row,col] = find(bw==1);
    x = min(col);
    y = min(row);
%         max(col)
%         min(col)
%         max(row)
%         min(row)
    width = max(col)-min(col);
    height = max(row)-min(row);
    bbox = [x+0.125*width y 0.75*width height];
    % menampilkan bounding box pada citra rgb
    RGB = insertShape(Img, 'rectangle', bbox, 'LineWidth', 20);
    % menampilkan citra hasil deteksi
    % figure, imshow(RGB);
    % menghitung nilai a dan t
    a(n,1) = 0.75*width;
    t(n,1) = height;
end

%%% SAMPING
b = zeros(jumlah_data,1);
max_b = 610;

for n = 1:jumlah_data
    % membaca file citra
    Img = imread(['C:\Users\BASIT\Dropbox\elangputih\Tesis
Basit\Program 2\',num2str(n),' samping.JPG']);
    % melakukan rotasi citra dengan sudut rotasi sebesar 90
    Img = imrotate(Img,90);
    % mengkonversi citra rgb menjadi citra grayscale
    Img_gray = rgb2gray(Img);
    % melakukan subtraksi dengan background dalam kelas double
    Subtraction = (double(Img_gray)-double(Back));
    % mengembalikan kelas citra ke dalam kelas uint8
    Min_S = min(Subtraction(:));
    Max_S = max(Subtraction(:));
    Subtraction = ((Subtraction-Min_S)/(Max_S-Min_S))*255;
    Subtraction = uint8(Subtraction);
    % melakukan thresholding dengan nilai threshold 0.4
    Subtraction = im2bw(Subtraction,.4);
    % melakukan komplemen
    bw = ~Subtraction;
    % melakukan area opening untuk menghilangkan noise
    bw = bwareaopen(bw,1000);
    % mendeteksi bounding box
    [row,col] = find(bw==1);
    x = min(col);
    y = min(row);
%      max(col)
%      min(col)
    width = max(col)-min(col);
    height = max(row)-min(row);
    bbox = [x y width height];
    % menampilkan bounding box pada citra rgb
    RGB = insertShape(Img, 'rectangle', bbox, 'LineWidth', 20);
    % menampilkan citra hasil deteksi
    % figure, imshow(RGB);
    % menghitung nilai b
    b(n,1) = max(col)-min(col);
end

 % menggabungkan nilai a,b, dan t
 data = [a,b,t];
% % menetapkan target berat badan
target =
[52;51;60;54;40;48;50;45;50;48;58;50;65;85;64;50;49;50;53;80;60;..
.
     39;55;40;40];
 % menyiapkan input dan target untuk proses pelatihan
 input = data';
 target = target';
 % membangun jaringan syaraf tiruan
 net = newff(minmax(input),[50,1],{'logsig','purelin'},'trainlm');
 % load dan setting bobot awal
 load bobot_awal
 net.IW{1,1} = bobot_hidden;
 net.LW{2,1} = bobot_keluaran;
 net.b{1,1} = bias_hidden;
 net.b{2,1} = bias_keluaran;

 bobot_hidden = net.IW{1,1};
 bobot_keluaran = net.LW{2,1};
 bias_hidden = net.b{1,1};
 bias_keluaran = net.b{2,1};

 % setting parameter jaringan
 net.trainParam.epochs = 1000;
 net.trainParam.goal = 1e-6;
 net.trainParam.lr = 0.1;

 % menyimpan bobot awal
 save bobot_awal bobot_hidden bobot_keluaran bias_hidden
bias_keluaran
 % melakukan pelatihan jaringan
 net_keluaran = train(net,input,target);
 % menghitung nilai keluaran jaringan
 output = round(sim(net_keluaran,input));
 % menampilkan grafik regresi
 figure,
 plotregression(target,output);
 % menyimpan jaringan hasil pelatihan
 save net_keluaran net_keluaran
```

CODING GUIDE INTERFACE

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

% Last Modified by GUIDE v2.5 28-Mar-2019 14:54:14

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @main_program_OpeningFcn, ...
                   'gui_OutputFcn',  @main_program_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 main_program is made visible.
function main_program_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 main_program (see VARARGIN)

% Choose default command line output for main_program
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
movegui(hObject, 'center');

% membaca citra background
Back = imread('background.JPG');

% menyimpan variabel Back pada lokasi handles agar dapat dipanggil
oleh
% pushbutton yg lain
handles.Back = Back;
guidata(hObject, handles);

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


% --- Outputs from this function are returned to the command line.
function varargout = main_program_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 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)
% menampilkan menu browse file
[file,path] = uigetfile({'*.mat','MAT File (*.mat)';
    '*.*','All file type (*.*)'},'Browse File');

% jika ada file yg dipilih maka mengeksekusi perintah2 yg ada di
bawahnya
if ~isequal(file,0)
    % mereset button2
    axes(handles.axes1)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes2)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes3)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes4)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes5)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes6)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes7)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes8)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes9)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])
    axes(handles.axes10)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes11)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes12)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    set(handles.edit1,'String',[])
    set(handles.edit2,'String',[])
    set(handles.edit3,'String',[])
    set(handles.edit4,'String',[])

    set(handles.pushbutton1,'Enable','on')
    set(handles.pushbutton2,'Enable','off')
    set(handles.pushbutton3,'Enable','off')
    set(handles.pushbutton4,'Enable','off')
    set(handles.pushbutton6,'Enable','off')

    % menyimpan variabel path & file pada lokasi handles agar
dapat
    % dipanggil oleh pushbutton yg lain
    handles.path = path;
    handles.file = file;
    guidata(hObject,handles);
else
    return
end


% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% memanggil variabel2 yg ada pada lokasi handles
a = handles.a;
b = handles.b;
t = handles.t;
path = handles.path;
file = handles.file;

% menggabungkan nilai a,b, dan t
data_uji = [a;b;t];
```

```
% load jaringan
load(fullfile(path,file))

% Hasil testing
nilai_keluaran = round(sim(net_keluaran,data_uji));

% menampilkan kelas keluaran pada edit text
set(handles.edit4,'String',[num2str(nilai_keluaran),' kg'])

function edit4_Callback(hObject, eventdata, handles)
% hObject    handle to edit4 (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 edit4 as text
%        str2double(get(hObject,'String')) returns contents of
edit4 as a double


% --- Executes during object creation, after setting all
properties.
function edit4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit4 (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 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)

% mereset button2
axes(handles.axes9)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes10)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])
```

```
axes(handles.axes11)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes12)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

set(handles.edit3,'String',[])
set(handles.edit4,'String',[])
set(handles.pushbutton4,'Enable','on')
set(handles.pushbutton6,'Enable','off')

% menampilkan menu browse file
[nama_file, nama_path] = uigetfile('*.jpg');

% jika ada file yg dipilih maka akan mengeksekusi perintah di
bawahnya
if ~isequal(nama_file,0)
    % membaca file citra yg dipilih
    Img2 = imread(fullfile(nama_path,nama_file));
    % melakukan rotasi citra dengan sudut rotasi sebesar 90
    Img2 = imrotate(Img2,90);
    % menampilkan citra foreground pada axes
    axes(handles.axes7)
    imshow(Img2)
    title('Foreground')
    % menampilkan citra background pada axes
    axes(handles.axes8)
    Back = handles.Back;
    imshow(Back)
    title('Background')
    % menyimpan variabel Img2 pada lokasi handles agar dapat
dipanggil oleh
    % pushbutton yg lain
    handles.Img2 = Img2;
    guidata(hObject, handles)
else
    % jika tidak ada file yg dipilih maka akan kembali
    return
end

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% mereset button2
set(handles.edit4,'String',[])
set(handles.pushbutton6,'Enable','on')
```

```
% memanggil variabel Back & Img2 yg ada di lokasi handles
Back = handles.Back;
Img2 = handles.Img2;
% mengkonversi citra background menjadi grayscale
Back = rgb2gray(Back);
% mengkonversi citra foreground menjadi citra grayscale
Img_gray = rgb2gray(Img2);
% melakukan subtraksi dengan background dalam kelas double
Subtraction = (double(Img_gray)-double(Back));
% mengembalikan kelas citra ke dalam kelas uint8
Min_S = min(Subtraction(:));
Max_S = max(Subtraction(:));
Subtraction = ((Subtraction-Min_S)/(Max_S-Min_S))*255;
Subtraction = uint8(Subtraction);
% menampilkan citra hasil subtraksi pada axes
axes(handles.axes9)
imshow(Subtraction)
title('Hasil Subtraksi')
% melakukan thresholding dengan nilai threshold 0.4
Subtraction = im2bw(Subtraction,.4);
% menampilkan citra hasil thresholding pada axes
axes(handles.axes10)
imshow(Subtraction)
title('Hasil Thresholding')
% melakukan komplemen
bw = ~Subtraction;
% menampilkan citra hasil komplemen pada axes
axes(handles.axes11)
imshow(bw)
title('Hasil Komplemen')
% melakukan area opening untuk menghilangkan noise
bw = bwareaopen(bw,1000);
% mendeteksi bounding box
[row,col] = find(bw==1);
x = min(col);
y = min(row);
width = max(col)-min(col);
height = max(row)-min(row);
bbox = [x y width height];
% menampilkan bounding box pada citra rgb
RGB = insertShape(Img2, 'rectangle', bbox, 'LineWidth', 20);
% menampilkan citra hasil deteksi pada axes
axes(handles.axes12)
imshow(RGB)
title('Tampak Samping')
% menghitung nilai b
b = max(col)-min(col);
% menampilkan nilai b pada edit text
set(handles.edit3,'String',b)
% menyimpan variabel b pada lokasi handles agar dapat dipanggil
oleh
% pushbutton yg lain
handles.b = b;
guidata(hObject, handles)
```

```
function edit3_Callback(hObject, eventdata, handles)
% hObject    handle to edit3 (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 edit3 as text
%        str2double(get(hObject,'String')) returns contents of
edit3 as a double


% --- Executes during object creation, after setting all
properties.
function edit3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit3 (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 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)

% menampilkan menu browse file
[nama_file, nama_path] = uigetfile('*.jpg');

% jika ada file yg dipilih maka akan mengeksekusi perintah di
bawahnya
if ~isequal(nama_file,0)
    % mereset button2
    axes(handles.axes3)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes4)
    cla reset
    set(gca,'XTick',[])
    set(gca,'yTick',[])

    axes(handles.axes5)
    cla reset
```

```
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes6)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes7)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes8)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes9)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes10)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes11)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes12)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

set(handles.edit1,'String',[])
set(handles.edit2,'String',[])
set(handles.edit3,'String',[])
set(handles.edit4,'String',[])
set(handles.pushbutton2,'Enable','on')
set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton6,'Enable','off')

% membaca file citra yg dipilih
Img = imread(fullfile(nama_path,nama_file));
% melakukan rotasi citra dengan sudut rotasi sebesar 90
Img = imrotate(Img,90);
% menampilkan citra foreground pada axes
axes(handles.axes1)
imshow(Img)
title('Foreground')
```

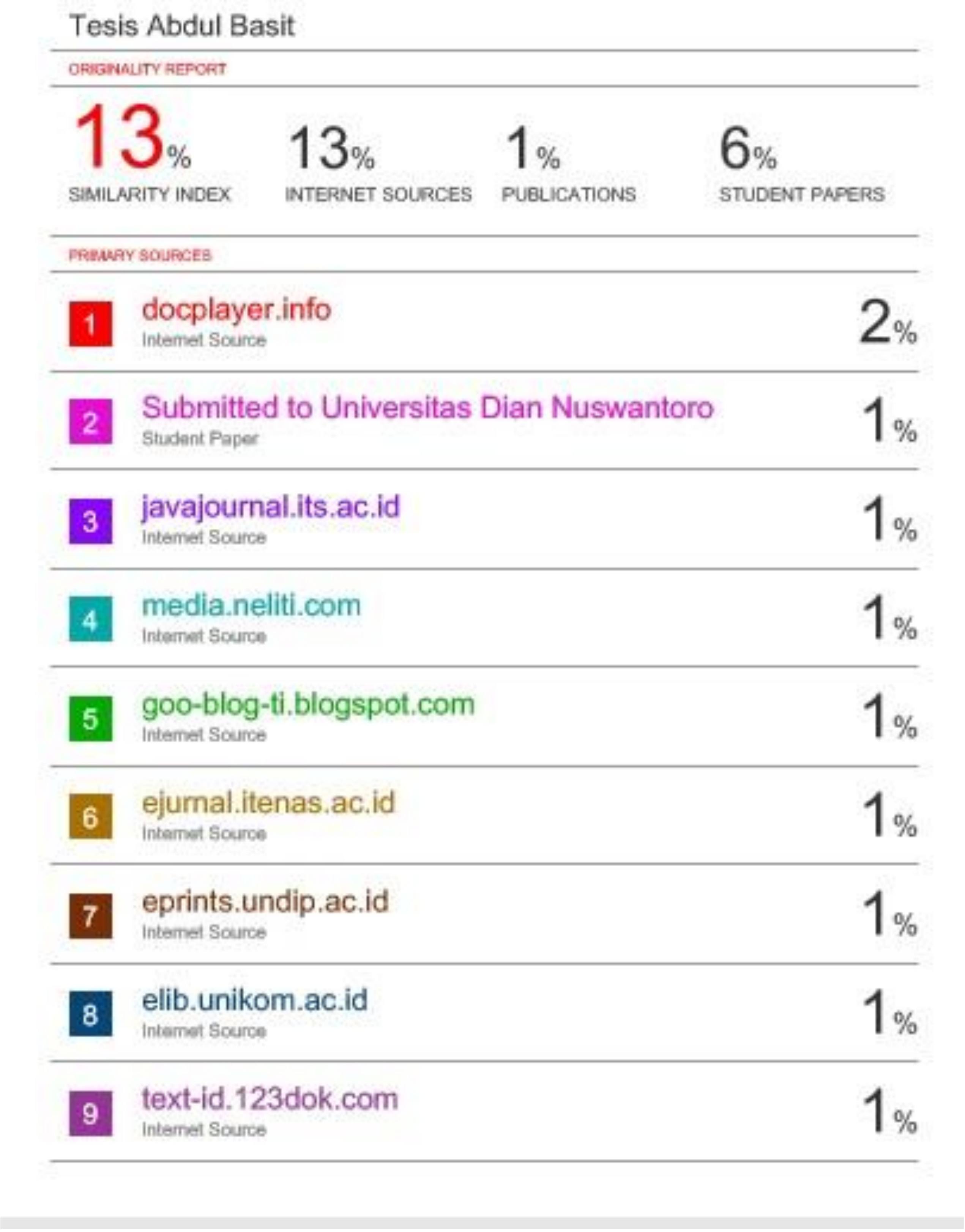
```
    % menampilkan citra background pada axes
    axes(handles.axes2)
    Back = handles.Back;
    imshow(Back)
    title('Background')

    % menyimpan variabel Img pada lokasi handles agar dapat
dipanggil oleh
    % pushbutton yg lain
    handles.Img = Img;
    guidata(hObject, handles)
else
    % jika tidak ada file yg dipilih maka akan kembali
    return
end


% --- 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)

% mereset button2
axes(handles.axes7)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes8)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes9)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes10)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes11)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes12)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])
set(handles.edit3,'String',[])
set(handles.edit4,'String',[])
set(handles.pushbutton3,'Enable','on')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton6,'Enable','off')

% memanggil variabel Back & Img yg ada di lokasi handles
Back = handles.Back;
Img = handles.Img;
% mengkonversi citra background menjadi grayscale
Back = rgb2gray(Back);
% mengkonversi citra foreground menjadi citra grayscale
Img_gray = rgb2gray(Img);
% melakukan subtraksi dengan background dalam kelas double
Subtraction = (double(Img_gray)-double(Back));
% mengembalikan kelas citra ke dalam kelas uint8
Min_S = min(Subtraction(:));
Max_S = max(Subtraction(:));
Subtraction = ((Subtraction-Min_S)/(Max_S-Min_S))*255;
Subtraction = uint8(Subtraction);
% menampilkan citra hasil subtraksi pada axes
axes(handles.axes3)
imshow(Subtraction)
title('Hasil Subtraksi')
% melakukan thresholding dengan nilai threshold 0.4
Subtraction = im2bw(Subtraction,.4);
% menampilkan citra hasil thresholding pada axes
axes(handles.axes4)
imshow(Subtraction)
title('Hasil Thresholding')
% melakukan komplemen
bw = ~Subtraction;
% menampilkan citra hasil komplemen pada axes
axes(handles.axes5)
imshow(bw)
title('Hasil Komplemen')
% melakukan area opening untuk menghilangkan noise
bw = bwareaopen(bw,1000);
% mendeteksi bounding box
[row,col] = find(bw==1);
x = min(col);
y = min(row);
width = max(col)-min(col);
height = max(row)-min(row);
bbox = [x+0.125*width y 0.75*width height];
% menampilkan bounding box pada citra rgb
RGB = insertShape(Img, 'rectangle', bbox, 'LineWidth', 20);
% menampilkan citra hasil deteksi pada axes
axes(handles.axes6)
imshow(RGB)
title('Hasil Deteksi')
% menghitung nilai a & t
a = max(col)-min(col);
t = max(row)-min(row);
% menampilkan nilai a & t pada edit text
```

```
set(handles.edit1,'String',a)
set(handles.edit2,'String',t)
% menyimpan variabel a & t pada lokasi handles agar dapat
dipanggil oleh
% pushbutton yg lain
handles.a = a;
handles.t = t;
guidata(hObject, handles)

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


% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton7 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% mereset button2
axes(handles.axes1)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes2)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes3)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes4)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes5)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes6)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes7)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes8)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes9)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes10)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes11)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

axes(handles.axes12)
cla reset
set(gca,'XTick',[])
set(gca,'yTick',[])

set(handles.edit1,'String',[])
set(handles.edit2,'String',[])
set(handles.edit3,'String',[])
set(handles.edit4,'String',[])

set(handles.pushbutton1,'Enable','off')
set(handles.pushbutton2,'Enable','off')
set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton6,'Enable','off')
```

Tesis Abdul Basit

by Abdul Basit

Submission date: 24-Jan-2020 10:43AM (UTC+0800)
Submission ID: 1245680042
File name: Tesis_Basit.docx (1.6M)
Word count: 11496
Character count: 64128



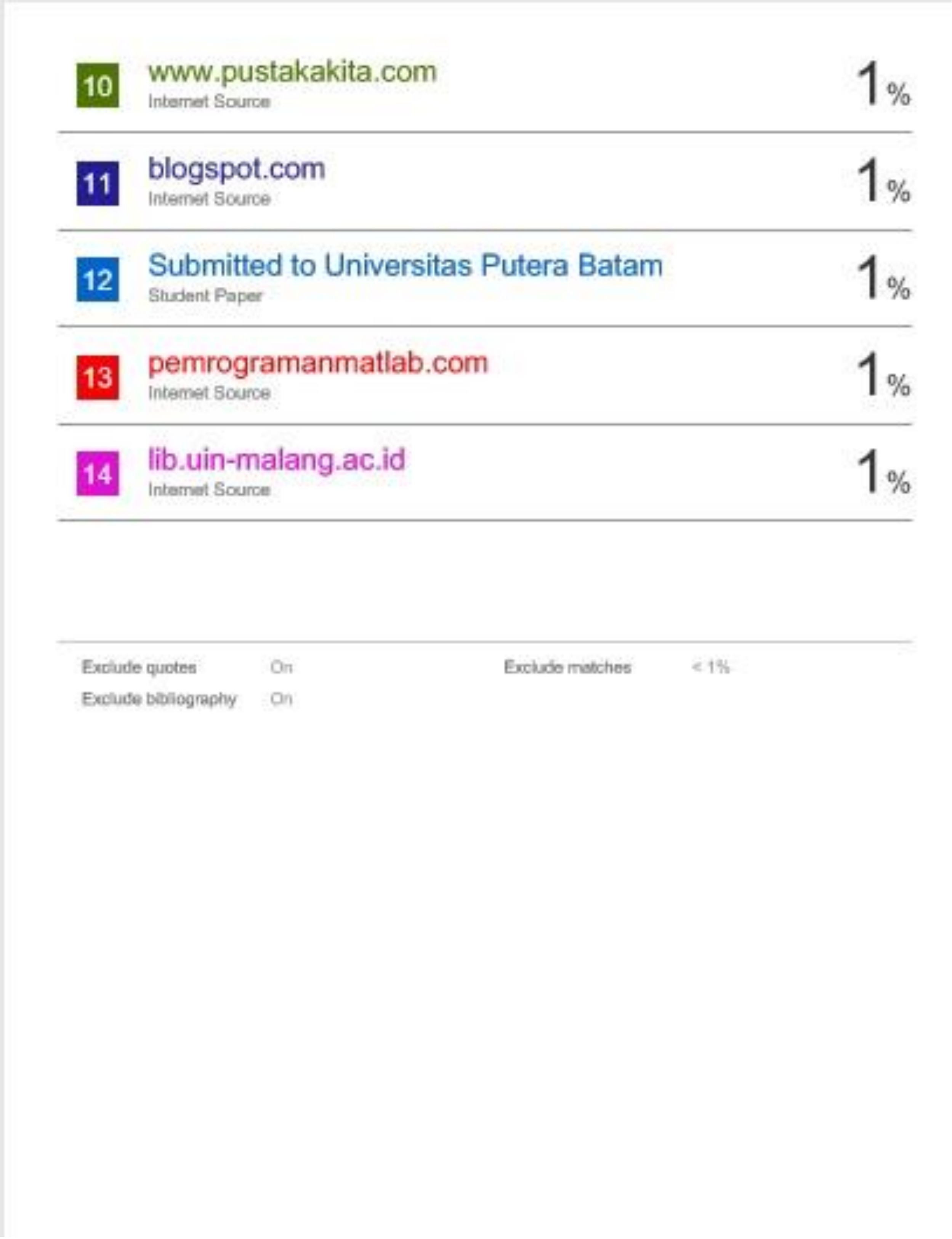
Tesis Abdul Basit

ORIGINALITY REPORT

13%	13%	1%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	docplayer.info Internet Source	2%
2	Submitted to Universitas Dian Nuswantoro Student Paper	1%
3	javajournal.its.ac.id Internet Source	1%
4	media.neliti.com Internet Source	1%
5	goo-blog-ti.blogspot.com Internet Source	1%
6	ejurnal.itenas.ac.id Internet Source	1%
7	eprints.undip.ac.id Internet Source	1%
8	elib.unikom.ac.id Internet Source	1%
9	text-id.123dok.com Internet Source	1%



10	www.pustakakita.com Internet Source	1%
11	blogspot.com Internet Source	1%
12	Submitted to Universitas Putera Batam Student Paper	1%
13	pemrogramanmatlab.com Internet Source	1%
14	lib.uin-malang.ac.id Internet Source	1%

Exclude quotes	On	Exclude matches	< 1%
Exclude bibliography	On		