

LAMPIRAN-1

Listing Program Matlab R2014a

```
function varargout = Watermarking4(varargin)
% WATERMARKING4 MATLAB code for Watermarking4.fig
%     WATERMARKING4, by itself, creates a new WATERMARKING4 or
raises the existing
%     singleton*.
%
%     H = WATERMARKING4 returns the handle to a new WATERMARKING4
or the handle to
%     the existing singleton*.
%
%     WATERMARKING4('CALLBACK',hObject,eventData,handles,...)
calls the local
%     function named CALLBACK in WATERMARKING4.M with the given
input arguments.
%
%     WATERMARKING4('Property','Value',...) creates a new
WATERMARKING4 or raises the
%     existing singleton*. Starting from the left, property
value pairs are
%     applied to the GUI before Watermarking4_OpeningFcn gets
called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to Watermarking4_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 Watermarking4

% Last Modified by GUIDE v2.5 14-Mar-2019 12:51:55

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Watermarking4_OpeningFcn,
                  ...
                  'gui_OutputFcn',  @Watermarking4_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 Watermarking4 is made visible.
function Watermarking4_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 Watermarking4 (see
VARARGIN)

% Choose default command line output for Watermarking4
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes Watermarking4 wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = Watermarking4_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)
[nama_file,nama_path] = uigetfile({'*.*'});

if ~isequal(nama_file,0)
    I = imread(fullfile(nama_path,nama_file));
    A=rgb2gray(I);
    %P1=im2double(A);
    %P2=imresize(P1,[512 512]);
    P2 = A;
    axes(handles.axes1);

```

```

        imshow(P2), title('Cover Image');
        handles.P2 = P2;
        csvwrite('gambar_I.csv',I)
        csvwrite('gambar_A.csv',A)
        guidata(hObject,handles);

else
    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)
[nama_file,nama_path] = uigetfile({'*.*'});

if ~isequal(nama_file,0)
    imw = imread(fullfile(nama_path,nama_file));
    imw2=rgb2gray(imw);
    %watermark1=im2double(imw2);
    %watermark=imresize(watermark1,[512 512]);
    watermark = imw2;
    axes(handles.axes2);
    imshow(watermark), title('Message Image');
    csvwrite('gambar_imw.csv',imw)
    csvwrite('gambar_imw2.csv',imw2)
    handles.watermark = watermark;
    guidata(hObject,handles);

else
    return
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)
P2 = handles.P2;
P1 = im2double(P2);
P3 = imresize(P1,[512 512]);
P4 = imresize(P1,[128 128]);
[m,n]=size(P3);
[F1,F2]= wfilters('haar', 'd');
[LL,LH,HL,HH] = dwt2(P3,'haar','d');
[LL1,LH1,HL1,HH1] = dwt2(LL,'haar','d');

csvwrite('gambar_P2.csv',P2)
csvwrite('gambar_P1.csv',P1)
csvwrite('gambar_P3.csv',P3)

```

```

csvwrite('gambar_LL.csv',LL)
csvwrite('gambar_LL1.csv',LL1)

%gambar 2
watermark = handles.watermark;

watermark1=im2double(watermark);
watermark=imresize(watermark1,[512 512]);
watermark_r_dwt=imresize(watermark1,[128 128]);
[L_L,L_H,H_L,H_H] = dwt2(watermark,'haar','d');
[L_L1,~,H_L1,H_H1] = dwt2(L_L,'haar','d');

Watermarkedimage=LL1+0.001*L_L1;
Watermarkedimage_level1=
idwt2(Watermarkedimage,LH1,HL1,HH1,'haar');
Watermarkedimage_final=idwt2(Watermarkedimage_level1,LH,HL,HH,'haar');
axes(handles.axes3) ;
imshow(Watermarkedimage_final,'DisplayRange',[]), title('Watermark
Image DWT');

csvwrite('gambar_watermark.csv',watermark)
csvwrite('gambar_watermark1.csv',watermark1)
csvwrite('gambar_watermark_r_dwt.csv',watermark_r_dwt)
csvwrite('gambar_L_L.csv',L_L)
csvwrite('gambar_L_L1.csv',L_L1)
csvwrite('gambar_Watermarkedimage.csv',Watermarkedimage)
csvwrite('gambar_Watermarkedimage_level1.csv',Watermarkedimage_level1)
csvwrite('gambar_Watermarkedimage_final.csv',Watermarkedimage_final)

%ekstrak
[F11,F22]= wfilters('haar', 'd');
[a b c d]=dwt2(Watermarkedimage_final,'haar','d');
[aa bb cc dd]=dwt2(a,'haar','d');

csvwrite('gambar_ektrak_a.csv',a)
csvwrite('gambar_ektrak_aa.csv',aa)

recovered_image=(aa-LL1)/0.001;
axes(handles.axes4) ;
imshow(recovered_image,'DisplayRange',[]), title('Extracted Host
Asli DWT');
csvwrite('gambar_recovered_image.csv',recovered_image)

%MSE
MSE = sum(sum((P4-recovered_image).^2))/(m*n);
%set(handles.edit9, 'String',MSE);
csvwrite('gamabr_P4.csv',P4)
csvwrite('gambar_recovered_image.csv',recovered_image)

%PSNR

```

```

PSNR = 10*log10(128*128/MSE);
set(handles.edit3, 'String',PSNR);

%NC
NC = corr2(P3,Watermarkedimage_final);
set(handles.edit4, 'String',NC);
guidata(hObject,handles);

% --- 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)

P2 = handles.P2;
P3 = im2double(P2);
P4 = imresize(P2,[128 128]);
P5 = im2double(P4);
[m,n]=size(P3);

[LL1,HL1,LH1,HH1] = dwt2(P2,'haar');
[LL2,HL2,LH2,HH2] = dwt2(LL1,'haar');
p = size(LL2);
[Uy,Sy,Vy] = svd(LL2);
q = size(Sy);
alpha = 0.75;

%applying SVD on watermarking
watermark = handles.watermark;
I1_w = imresize(watermark,p);
watermark_r_dwt_svd = imresize(watermark,[128 128]);
watermark_r_dwt_svd_d = im2double(watermark_r_dwt_svd);

[Uw,Sw,Vw] = svd(double(I1_w));

%embed watermark
Smark = Sy+alpha*Sw;

%Rebuild the sub-bands using SVD
LL2_1 = Uy*Smark*Vy';

%Apply inverse dwt to get watermarked image
LL1_1 = idwt2(LL2_1,HL2,LH2,HH2,'haar');
I_1 = idwt2(LL1_1,HL1,LH1,HH1,'haar');

axes(handles.axes3);
imshow(uint8(I_1)), title('Watermark Image DWT-SVD');
handles.I_1 = I_1;

[LL1_wmv,HL1_wmv,LH1_wmv,HH1_wmv]=dwt2(I_1,'haar');
[LL1_wmv,HL1_wmv,LH1_wmv,HH1_wmv]=dwt2(I_1,'haar');
[LL2_wmv,HL2_wmv,LH2_wmv,HH2_wmv]=dwt2(LL1_wmv,'haar');

```

```

[Uy_wmv,Sy_wmv,Vy_wmv]=svd(LL2_wmv);
Swrec=(Sy_wmv-Sy)/alpha;
WMy=Uw*Swrec*Vw';
axes(handles.axes4);
imshow(uint8(WMy)), title('Extracted Host Asli DWT-SVD');

%MSE
MSE = sum(sum((P2-watermark).^2))/(m*n);
%PSNR
PSNR = 10*log10(256*256/MSE);
set(handles.edit3, 'String',PSNR);

%NC
NC = corr2(P3,I_1);
set(handles.edit4, 'String',NC);
guidata(hObject,handles);

% --- 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)
af = 0.75;
P2 = handles.P2;
P3 = im2double(P2);
P4 = imresize(P2,[256 256]);
P5 = im2double(P4);
[m,n]=size(P3);

csvwrite('gambar2_P2.csv',P2)
csvwrite('gambar2_P3.csv',P3)

watermark = handles.watermark;
W = im2double(watermark);
[m1,n1]=size(W);
WW=zeros(m,n);
[U,S,V]=svd(P3);
for i=1:m1
    for j=1:n1
        WW(i,j)=W(i,j);
    end
end
S1=S+af*WW;
[U1,SS,V1]=svd(S1);
CWI=U*SS*V';
axes(handles.axes3);
imshow(CWI), title('Watermark Image SVD');

csvwrite('gambar2_watermark.csv',watermark)
csvwrite('gambar2_W.csv',W)
csvwrite('gambar2_WW.csv',WW)
csvwrite('gambar2_S.csv',S)
csvwrite('gambar2_U.csv',U)

```

```

csvwrite('gambar2_V.csv',V)
csvwrite('gambar2_S1.csv',S1)
csvwrite('gambar2_SS.csv',SS)
csvwrite('gambar2_U1.csv',U1)
csvwrite('gambar2_V1.csv',V1)
csvwrite('gambar2_CWI.csv',CWI)

NCWI=zeros(size(CWI));
AA=randn(size(CWI));
NCWII=CWI+AA*0.01;
[UU,S2,VV]=svd(NCWII);
SN=U1*S2*V1';
WN=(SN-S)/af;
WNN=zeros(m1,n1);
for i=1:m1
    for j=1:n1
        WNN(i,j)=WN(i,j);
    end
end
axes(handles.axes4);
imshow(WNN), title('Extracted Host Asli SVD');
csvwrite('gambar2_NCWI.csv',NCWI)
csvwrite('gambar2_NCWII.csv',NCWII)
csvwrite('gambar2_AA.csv',AA)
csvwrite('gambar2_WN.csv',WN)
csvwrite('gambar2_WNN.csv',WNN)

%MSE
MSE = sum(sum((P3-WNN).^2))/(m*n);
%PSNR
PSNR = 10*log10(256*256/MSE);
set(handles.edit3, 'String',PSNR);
%NC
NC = corr2(P3,CWI);
set(handles.edit4, 'String',NC);
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 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)
P2 = handles.P2;
P1=im2double(P2);
P3=imresize(P1,[512 512]);
P4_n = imresize(P1,[128 128]);
[m,n]=size(P3);
[F1,F2]= wfilters('haar', 'd');
[LL,LH,HL,HH] = dwt2(P3,'haar','d');
[LL1,LH1,HL1,HH1] = dwt2(LL,'haar','d');

```



```

%gambar 2
watermark = handles.watermark;

watermark1=im2double(watermark);
watermark=imresize(watermark1,[512 512]);
watermark_r_dwt_n=imresize(watermark1,[128 128]);
[L_L,L_H,H_L,H_H] = dwt2(watermark,'haar','d');
[L_LL,L_HL,H_LL,H_HL] = dwt2(L_L,'haar','d');
Watermarkedimage=LL1+0.0001*L_LL;
Watermarkedimage_level1=
idwt2(Watermarkedimage,LH1,HL1,HH1,'haar');
Watermarkedimage_final=idwt2(Watermarkedimage_level1,LH,HL,HH,'haar');

%noise
LEN = 20;
THETA = 10;
PSF = fspecial('motion', LEN, THETA);
Watermarkedimage_final_N = imfilter(Watermarkedimage_final, PSF,
'conv', 'circular');
axes(handles.axes5);
imshow(Watermarkedimage_final_N,'DisplayRange',[]),title('Watermark Image Blur DWT');

%ekstrak
%[F11,F22]= wfilters('haar', 'd');
[a b c, d]=dwt2(Watermarkedimage_final_N,'haar','d');
[aa bb cc dd]=dwt2(a,'haar','d');

recovered_image=(aa-LL1)/0.001;
axes(handles.axes6);
imshow(recovered_image,'DisplayRange',[]),title('Extracted Host Asli Blur DWT');

%MSE
MSE = sum(sum((P4_n-recovered_image).^2))/(m*n);

%PSNR
PSNR = 10*log10(256*256/MSE);
set(handles.edit5, 'String',PSNR);

%NC
NC = corr2(P4_n,recovered_image);
set(handles.edit6, 'String',NC);
guidata(hObject,handles);

% --- 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)

```

```

P2 = handles.P2;
P3 = im2double(P2);
P4 = imresize(P2,[128 128]);
P5_n = im2double(P4);
[m,n]=size(P3);

[LL1,HL1,LH1,HH1] = dwt2(P2,'haar');
[LL2,HL2,LH2,HH2] = dwt2(LL1,'haar');
p = size(LL2);
[Uy,Sy,Vy] = svd(LL2);
q = size(Sy);
alpha = 0.75;
%applying SVD on watermarking

watermark = handles.watermark;
I1_w = imresize(watermark,p);
watermark_r_dwt_svd_n = imresize(watermark,[128 128]);
[Uw,Sw,Vw] = svd(double(I1_w));

%embed watermark
Smrk = Sy+alpha*Sw;

%Rebuild the sub-bands using SVD
LL2_1 = Uy*Smrk*Vy';

%Apply inverse dwt to get watermarked image
LL1_1 = idwt2(LL2_1,HL2,LH2,HH2,'haar');
I_1 = idwt2(LL1_1,HL1,LH1,HH1,'haar');

%noise
LEN = 20;
THETA = 10;
PSF = fspecial('motion', LEN, THETA);
I_1_N = imfilter(I_1, PSF, 'conv', 'circular');

axes(handles.axes5);
imshow(uint8(I_1_N)), title('Watermark Image Blur DWT-SVD');
handles.I_1 = I_1;

[LL1_wmv,HL1_wmv,~,HH1_wmv]=dwt2(I_1_N,'haar');
[LL1_wmv,HL1_wmv,LH1_wmv,HH1_wmv]=dwt2(I_1_N,'haar');
[LL2_wmv,HL2_wmv,LH2_wmv,HH2_wmv]=dwt2(LL1_wmv,'haar');
[Uy_wmv,Sy_wmv,Vy_wmv]=svd(LL2_wmv);
Swrec=(Sy_wmv-Sy)/alpha;
WMy=Uw*Swrec*Vw';

axes(handles.axes6);
imshow(uint8(WMy)), title('Extracted Host Asli Blur DWT-SVD');
%imwrite(WMy,'image_asli_n.jpg');

%MSE
MSE = sum(sum((P5_n-WMy).^2))/(m*n);
%PSNR
PSNR = 10*log10(256*256/MSE);

```

```

set(handles.edit5, 'String',PSNR);

%NC
NC = corr2(P5_n,WMy);
set(handles.edit6, 'String',NC);
guidata(hObject,handles);

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton8 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
af = 0.75;
P2 = handles.P2;
P3 = im2double(P2);
P4 = imresize(P2,[256 256]);
P5_n = im2double(P4);
[m,n]=size(P3);

watermark = handles.watermark;
W = im2double(watermark);
[m1,n1]=size(watermark);
WW=zeros(m,n);
[U,S,V]=svd(P3);
for i=1:m1
    for j=1:n1
        WW(i,j)=W(i,j);
    end
end
S1=S+af*WW;
[U1,SS,V1]=svd(S1);
CWI=U*SS*V';

LEN = 20;
THETA = 10;
PSF = fspecial('motion', LEN, THETA);
CWI_N = imfilter(CWI, PSF, 'conv', 'circular');

%CWI_N = imnoise(CWI,'salt&pepper');
axes(handles.axes5);
imshow(CWI_N), title('Watermark Image SVD Blur');

NCWI=zeros(size(CWI_N));
AA=randn(size(CWI_N));
NCWI=CWI_N+AA*0.01;
[UU,S2,VV]=svd(NCWI);
SN=UU*S2*VV';
WN=(SN-S)/af;
WNN=zeros(m1,n1);
for i=1:m1
    for j=1:n1
        WNN(i,j)=WN(i,j);
    end
end
end

```

```

axes(handles.axes6);
WNN_N = WNN;
imshow(WNN_N, title('Extracted Host Asli SVD Blur '));

%MSE
MSE = sum(sum((P3-20).^2))/(m*n);
%PSNR
PSNR = 10*log10(256*256/MSE);
set(handles.edit5, 'String',PSNR);
%NC
NC = corr2(P3,WNN_N);
set(handles.edit6, 'String',NC);
handles.WNN_N = WNN_N;
guidata(hObject,handles);

% --- Executes on button press in pushbutton9.
function pushbutton9_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton9 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in pushbutton10.
function pushbutton10_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton10 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in pushbutton11.
function pushbutton11_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton11 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

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

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

function edit5_Callback(hObject, eventdata, handles)
% hObject    handle to edit5 (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 edit5 as text
%         str2double(get(hObject,'String')) returns contents of
edit5 as a double

```

```

% --- Executes during object creation, after setting all
properties.
function edit5_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit5 (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 edit6 (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 edit6 as text
%       str2double(get(hObject,'String')) returns contents of
edit6 as a double

% --- Executes during object creation, after setting all
properties.
function edit6_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit6 (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 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

```

```

% --- Executes on button press in pushbutton12.

```

```

function pushbutton12_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton12 (see GCBO)

```

```

% eventdata reserved - to be defined in a future version of
MATLAB
% handles structure with handles and user data (see GUIDATA)
watermark = handles.watermark;
WNN_N = handles.WNN_N;
a = watermark;
b = WNN_N;
m=size(a);
n=size(b);
a=im2double(a);
b=im2double(b);
a1=mean2(a);
b1=mean2(b);
c1=0;c2=0;
for i=1:m
    for j=1:n
        c1=c1+(a(i,j)-a1);
        c2=c2+(b(i,j)-b1);
        num=c1*c2;
        c3=(c1^2)*(c2^2);
        dem=sqrt(c3);
    end
end
nc=num/dem;
set(handles.edit6, 'String',nc);
guidata(hObject,handles);

function edit9_Callback(hObject, eventdata, handles)
% hObject handle to edit9 (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 edit9 as text
% str2double(get(hObject,'String')) returns contents of
edit9 as a double

% --- Executes during object creation, after setting all
properties.
function edit9_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit9 (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

```

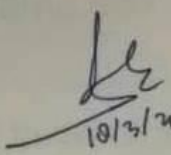

LAMPIRAN-2

Layout Program Matlab R2014a



Analisis Robustnes dan Imperceptibility pada Teknik Watermarking Citra Digital

by Rony D


10/2/2019
Ir. Suryan A., Ph.D.



Prof. Marwanto, ST., M.Eng.
210600018 / 06-2809-7200

mission date: 18-Mar-2019 04:26PM (UTC+0800)

mission ID: 1095230499

name: Rony-Analisis_Robustnes_dan_Imperceptibility.pdf (2.65M)

file count: 11538

character count: 56558

Analisis Robustnes dan Imperceptibility pada Teknik Watermarking Citra Digital

ORIGINALITY REPORT

23%

SIMILARITY INDEX

23%

INTERNET SOURCES

0%

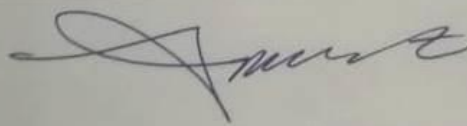
PUBLICATIONS

4%

STUDENT PAPERS

PRIMARY SOURCES

1	ee.ui.ac.id Internet Source	7%
2	ml.scribd.com Internet Source	6%
3	eprints.undip.ac.id Internet Source	4%
4	eprints.upnjatim.ac.id Internet Source	3%
5	majour.maranatha.edu Internet Source	2%
6	Submitted to Universitas Muhammadiyah Surakarta Student Paper	2%



18/3/19
Suryani A., Ph.D

Arief Marwanto, ST., M.Eng., Ph.D

Exclude quotes

290800018 / 06-2809-7501

Exclude matches

< 2%

Exclude bibliography

On