

Lampiran Koding

```
function varargout = kmeans1(varargin)
% KMEANS1 M-file for kmeans1.fig
%     KMEANS1, by itself, creates a new KMEANS1 or raises the
existing
%     singleton*.
%
%     H = KMEANS1 returns the handle to a new KMEANS1 or the
handle to
%     the existing singleton*.
%
%     KMEANS1('CALLBACK',hObject,eventData,handles,...) calls
the local
%     function named CALLBACK in KMEANS1.M with the given input
arguments.
%
%     KMEANS1('Property','Value',...) creates a new KMEANS1 or
raises the
%     existing singleton*. Starting from the left, property
value pairs are
%     applied to the GUI before kmeans1_OpeningFcn gets called. An
unrecognized
%     property name or invalid value makes
property application
%     stop. All inputs are passed to kmeans1_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 kmeans1

% Last Modified by GUIDE v2.5 16-Jan-2017 13:52:25

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @kmeans1_OpeningFcn, ...
                  'gui_OutputFcn',  @kmeans1_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 kmeans1 is made visible.
function kmeans1_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 kmeans1 (see VARARGIN)

% Choose default command line output for kmeans1
handles.output = hObject;
axes(handles.axes13);
imshow('logo.jpg')

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes kmeans1 wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command
line.
function varargout = kmeans1_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({'*.jpg'; '*.bmp'; '*.png'; '*.tif'}
, '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);
i=get(proyek.axes1, 'Userdata');

cform = makecform('srgb2lab');
lab = applycform(i, cform);

ab = double(lab(:, :, 2:3));
nrows = size(ab,1);
ncols = size(ab,2);
ab = reshape(ab,nrows*ncols,2);

nColors = 3;
[cluster_idx, cluster_center] =
kmeans(ab, nColors, 'distance', 'sqEuclidean', ...
        'Replicates', 3);

pixel_labels = reshape(cluster_idx, nrows, ncols);
RGB = label2rgb(pixel_labels);

segmented_images = cell(1,3);
rgb_label = repmat(pixel_labels, [1 1 3]);

for k = 1:nColors
    color = i;
    color(rgb_label ~= k) = 0;
    segmented_images{k} = color;

set(proyek.figure1, 'CurrentAxes', proyek.axes2);
set(imshow(lab));
set(proyek.figure1, 'CurrentAxes', proyek.axes3);
set(imshow(RGB, []));
set(proyek.figure1, 'CurrentAxes', proyek.axes4);

```

```
set(imshow(segmented_images{k}));
```

```
end
```

```
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
```

```
% --- 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)
```

```
proyek=guidata(gcbo);
```

```
i=get(proyek.axes1,'Userdata');
```

```
s= imresize(i,0.2,'bilinear');
```

```
J=rgb2gray(i);
```

```
t=graythresh(J);
```

```
a=im2bw(J,t);
```

```
l = +a
```

```
m = [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]
```

```
e=conv2(l,m);
```

```

B = imresize (J,2); % Citra binaryzation"
bw1=edge(e, 'log');

%c=bwlabel(B);bw1
%I1=imcrop(J);
[w h]=size(J);% coding lbp
for i=2:w-1
    for j=2:h-1
        val=J(i,j); scale=2.^[0 1 2;7 -inf 3;6 5 4];
        mat=[J(i-1,j-1) J(i-1,j) J(i-1,j+1);J(i,j-1) J(i,j)
J(i,j+1);J(i+1,j-1) J(i+1,j) J(i+1,j+1)];
        mat=mat>=val; fin=mat.*scale;
J(i,j)=uint8(sum(sum(fin)));
    end
end

%A = medfilt2(B,[5,5]); % Median Filtering"

%resize gambar
%gbr='plat.jpg';
%I = imread(gbr);
%J = imresize(i,0.6,'bilinear');
%G = rgb2gray(J)

%coba imclose
se= strel('disk',10);
closebw = imclose(bw1,se);

%menghilangkan noise (10 pixel)
pix=bwarea(B)/(10^4)*10000;
hasil=pix;

if pix==3.4141e+006;
    hasil='Kelayakan Pelumas';
end

set(projek.edit1,'string',hasil);
% --- Executes during object creation, after setting all
properties.
function axes13_CreateFcn(hObject, eventdata, handles)
% hObject    handle to axes13 (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 axes13

% --- 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)
cla(handles.axes3);
cla(handles.axes2);
cla(handles.axes1);
cla(handles.axes4);
set(handles.edit1, 'string', 'hasil...??');

% --- 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)
pertama;
```