

Pemrograman Matlab

Pengolahan Citra Digital, Pengolahan Video, Pengenalan Pola, dan Data Mining

Logika Fuzzy untuk Sistem Pengatur Kecepatan Mesin

AUG 24
Posted by adi pamungkas



(<https://pemrogramanmatlab.files.wordpress.com/2015/08/logika-fuzzy-menggunakan-matlab.gif>)

Dalam paper yang berjudul Fuzzy Sets* (<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwi7zry5t6PPAhUFI5QKHRgHIInformation%2520and%2520Control-1965.pdf&usg=AFOjCNEhAOfW8lbyz5CssyP4X5wfPAg14Q>), pada tahun 1965 Dr. Lotfi Aliasker Zadeh (https://en.wikipedia.org/wiki/Lotfi_A._Zadeh) (Ilmuwan Amerika Serikat berkebangsaan Iran dari University of California, Berkeley, California) memperkenalkan teori fuzzy yang mampu memetakan nilai masukan menuju nilai keluaran. Tidak seperti pada logika Boolean yang menyatakan suatu nilai dengan tegas (0 atau 1, hitam atau putih, ya atau tidak, benar atau salah, hidup atau mati), teori fuzzy menggunakan logika yang menyatakan bahwa suatu nilai dapat memiliki range atau derajat level (0 s.d 1, hitam s.d putih). Logika fuzzy dapat diartikan sebagai logika yang samar, kabur, tidak jelas, atau tidak tegas. Logika fuzzy umumnya diterapkan pada masalah-masalah yang mengandung unsur ketidakpastian (*uncertainty*), ketidaktepatan (*imprecise*), *noisy*, dan sebagainya. Logika fuzzy menjembatani bahasa mesin yang presisi dengan bahasa manusia yang menekankan pada makna atau arti (*significance*). Logika fuzzy dikembangkan berdasarkan bahasa manusia (bahasa alami).

Berikut ini merupakan contoh aplikasi pemrograman **logika fuzzy** untuk sistem pengatur **kecepatan mesin** menggunakan **sensor suhu** dan **sensor cahaya** sebagai masukan. Langkah-langkahnya adalah sebagai berikut:
1. Menyusun konsep sistem kontrol dengan **logika fuzzy**
Misalnya keadaan **sensor suhu (input 1)** dibagi menjadi lima kategori yaitu:

INPUT 1	
SUHU	KETERANGAN

Dingin	$0^{\circ}\text{C} - 15^{\circ}\text{C}$
Sejuk	$11^{\circ}\text{C} - 25^{\circ}\text{C}$
Normal	$21^{\circ}\text{C} - 30^{\circ}\text{C}$
Hangat	$28^{\circ}\text{C} - 40^{\circ}\text{C}$
Panas	$36^{\circ}\text{C} - 50^{\circ}\text{C}$

keadaan **sensor cahaya (input 2)** dibagi menjadi tiga kategori yaitu:

INPUT 2	
CAHAYA	KETERANGAN
Gelap	0 Cd– 35 Cd
Normal	31 Cd– 85 Cd
Terang	81 Cd- 100 Cd

keadaan **kecepatan mesin (output)** dibagi menjadi tiga kategori yaitu:

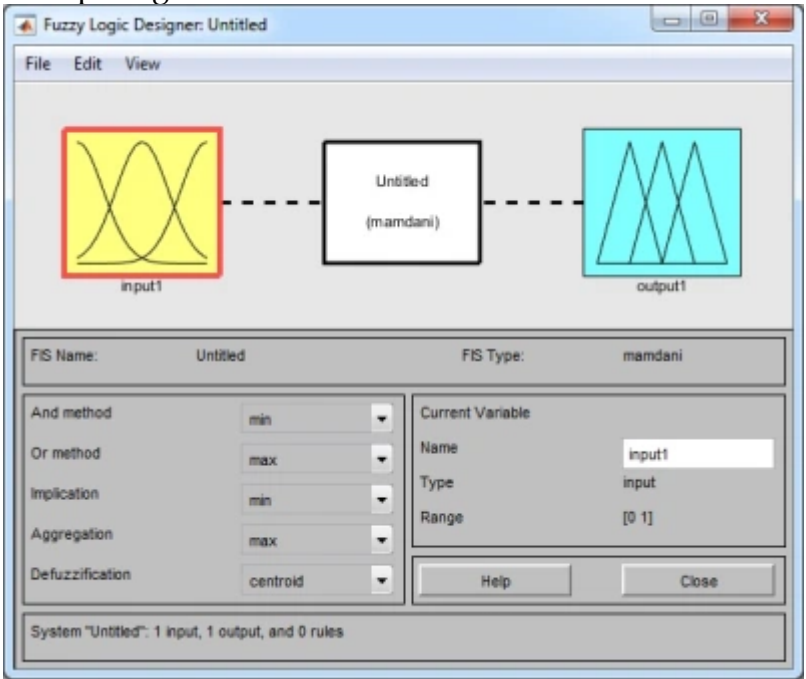
OUTPUT	
KEC. MESIN	KETERANGAN
Lambat	0 m/s– 15 m/s
Sedang	11 m/s – 21 m/s
Cepat	19 m/s – 45 m/s

aturan /rules adalah sebagai berikut:

No.	INPUT		OUTPUT
	SUHU	CAHAYA	KEC. MESIN
1	Dingin	Gelap	Lambat
2	Dingin	Normal	Lambat
3	Dingin	Terang	Lambat
4	Sejuk	Gelap	Lambat
5	Sejuk	Normal	Lambat
6	Sejuk	Terang	Sedang
7	Normal	Gelap	Sedang
8	Normal	Normal	Sedang

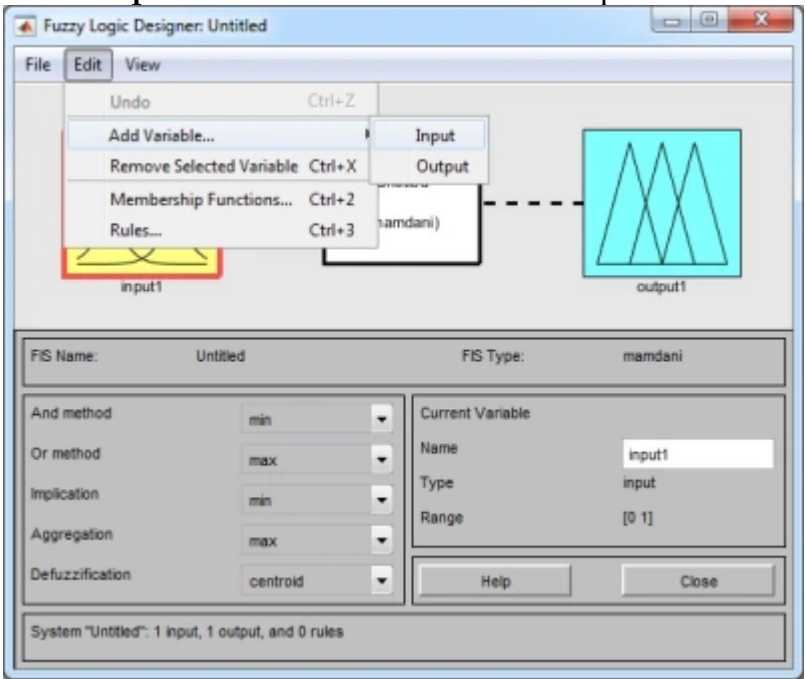
9	Normal	Terang	Sedang
10	Hangat	Gelap	Sedang
11	Hangat	Normal	Cepat
12	Hangat	Terang	Cepat
13	Panas	Gelap	Cepat
14	Panas	Normal	Cepat
15	Panas	Terang	Cepat

2. Setelah konsep sistem kontrol dibentuk, maka kita dapat membuat pemrogramannya. Ketiklah **“fuzzy”** pada **command window** untuk membuka jendela **Fuzzy Inference System (FIS) editor**, sehingga muncul tampilan seperti gambar di bawah ini:



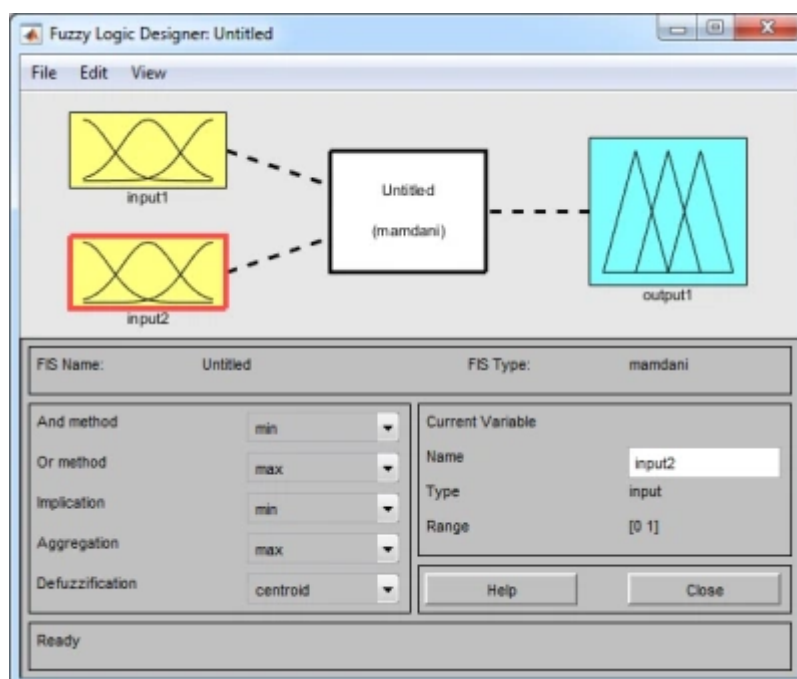
(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab.jpg>)

3. Pilih **edit >> add variable >> input** untuk menambah variable input



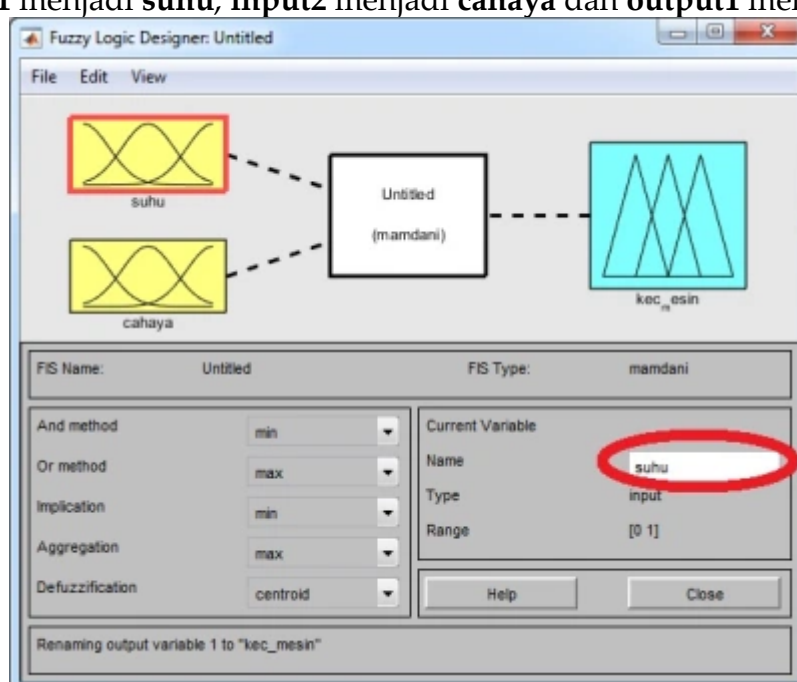
(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-menambah-variabel.jpg>)

sehingga akan muncul tampilan seperti pada gambar di bawah ini:



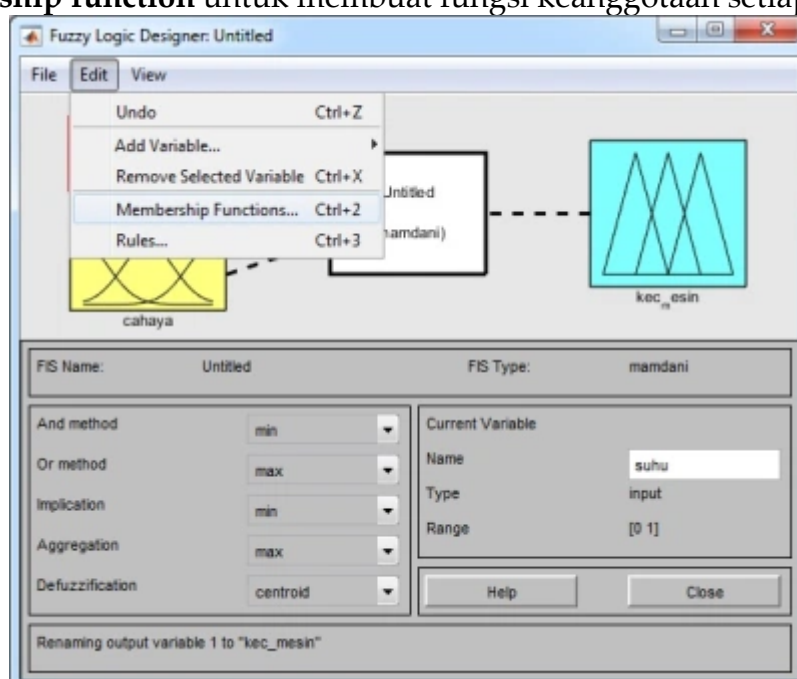
(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-penambahan-variabel.jpg>)

3. Ubahlah nama **input1** menjadi **suhu**, **input2** menjadi **cahaya** dan **output1** menjadi **kec_mesin**



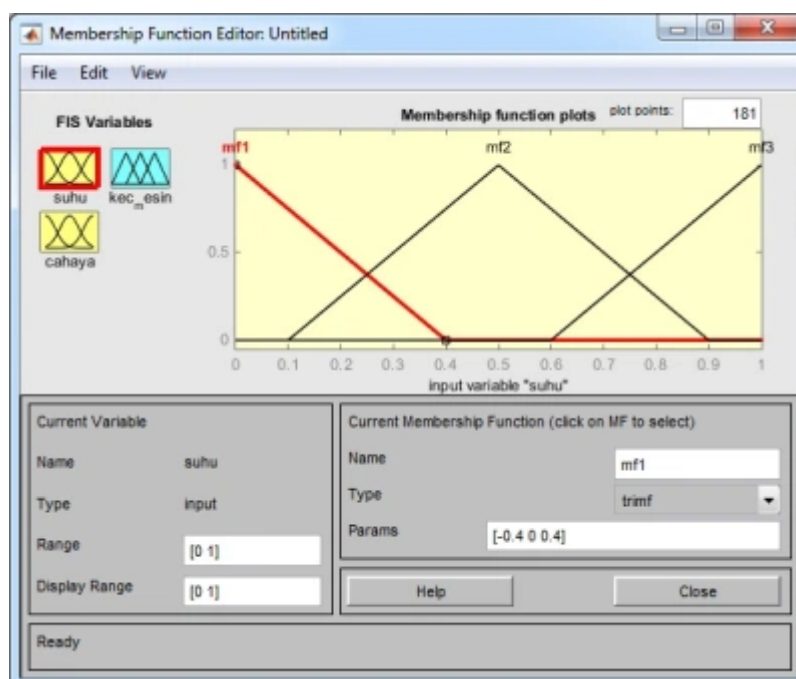
(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-edit-nama-variabel-input.jpg>)

4. Pilih **edit >> membership function** untuk membuat fungsi keanggotaan setiap variabel



(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function.jpg>)

sehingga akan muncul tampilan **Membership Function Editor** seperti pada gambar di bawah ini:

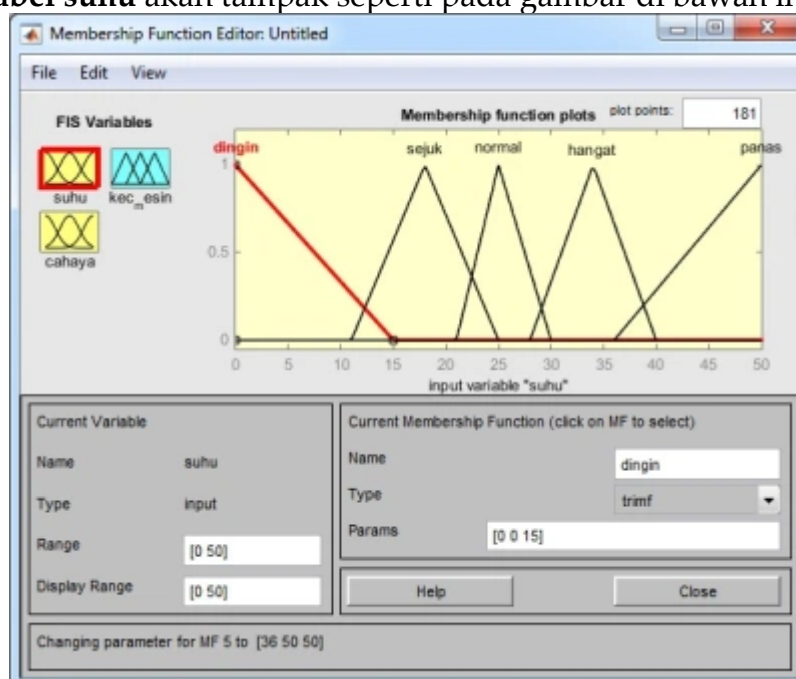


(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor.jpg>)

5. Pada **variabel suhu**, ubahlah **range** menjadi [0 50],
 nama **mf1** menjadi **dingin**, type **trimf**, Params [0 0 15]
 nama **mf2** menjadi **sejuk**, type **trimf**, Params [11 18 25]
 nama **mf3** menjadi **normal**, type **trimf**, Params [21 25 30]

pilih **edit >> add mfs** untuk menambah **membership function**
 nama **mf4** menjadi **hangat**, type **trimf**, Params [28 34 40]
 nama **mf5** menjadi **panas**, type **trimf**, Params [36 50 50]

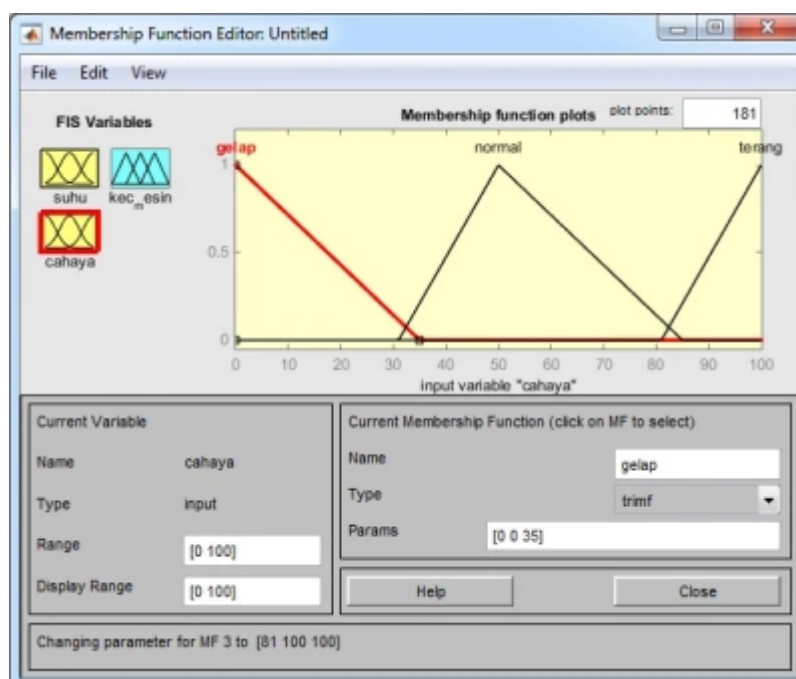
sehingga tampilan **variabel suhu** akan tampak seperti pada gambar di bawah ini:



(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-variabel-input.jpg>)

6. Pada **variabel cahaya**, ubahlah **range** menjadi [0 100],
 nama **mf1** menjadi **gelap**, type **trimf**, Params [0 0 35]
 nama **mf2** menjadi **normal**, type **trimf**, Params [31 50 85]
 nama **mf3** menjadi **terang**, type **trimf**, Params [81 100 100]

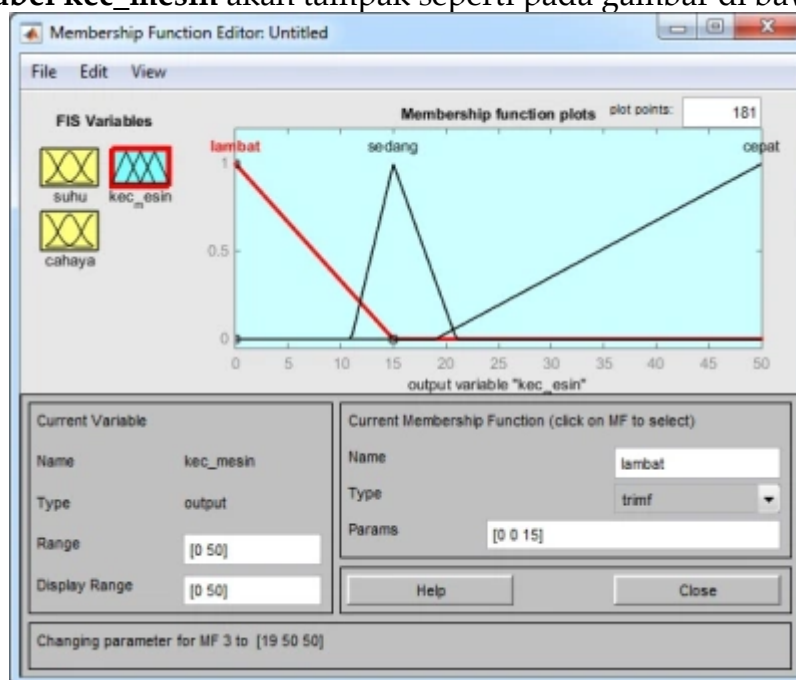
sehingga tampilan **variabel cahaya** akan tampak seperti pada gambar di bawah ini:



(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-variabel-input-2.jpg>)

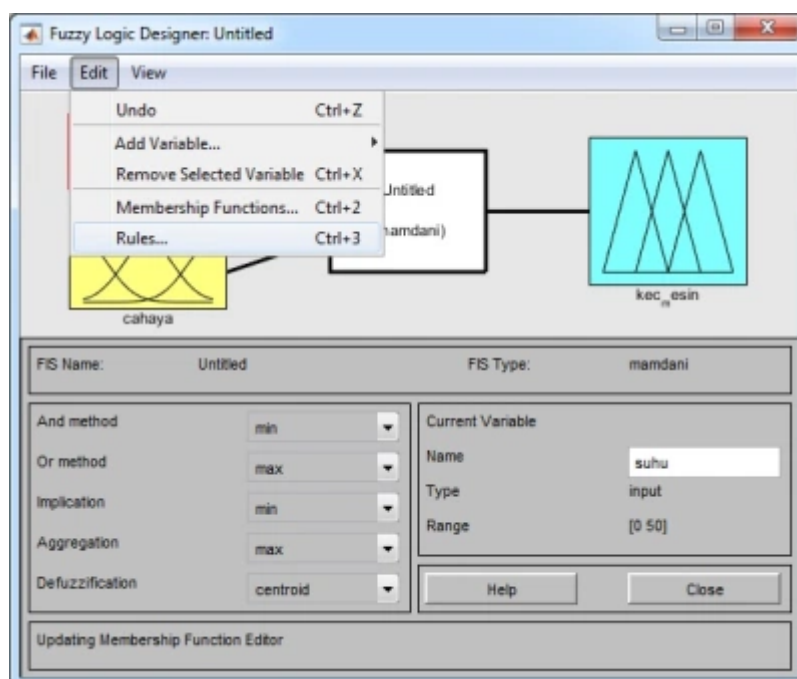
7. Pada **variabel kec_mesin**, ubahlah **range** menjadi [0 50],
 nama **mf1** menjadi **lambat**, type **trimf**, Params [0 0 15]
 nama **mf2** menjadi **sedang**, type **trimf**, Params [11 15 21]
 nama **mf3** menjadi **cepat**, type **trimf**, Params [19 50 50]

sehingga tampilan **variabel kec_mesin** akan tampak seperti pada gambar di bawah ini:



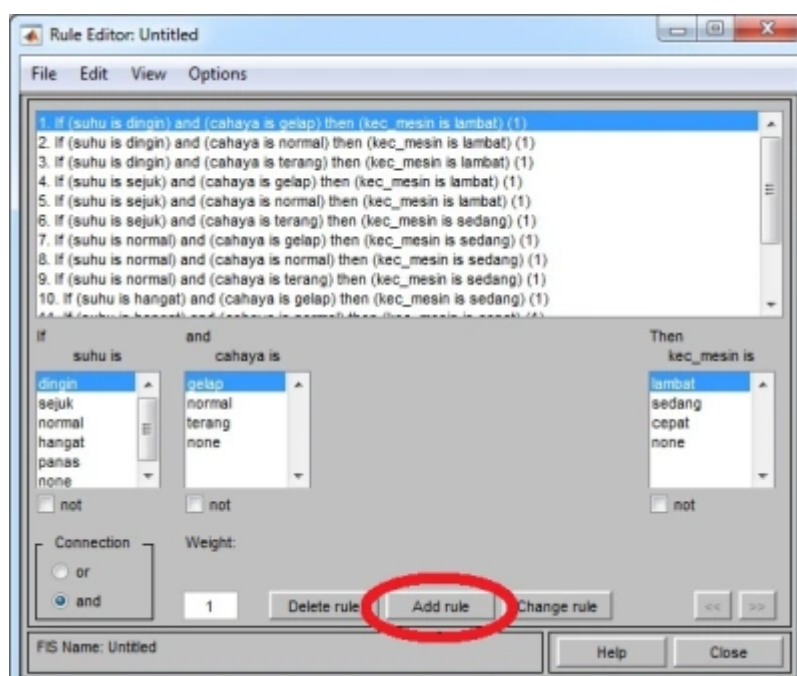
(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-variabel-output.jpg>)

8. Pilih **edit >> rules** untuk membuka jendela **rule editor**



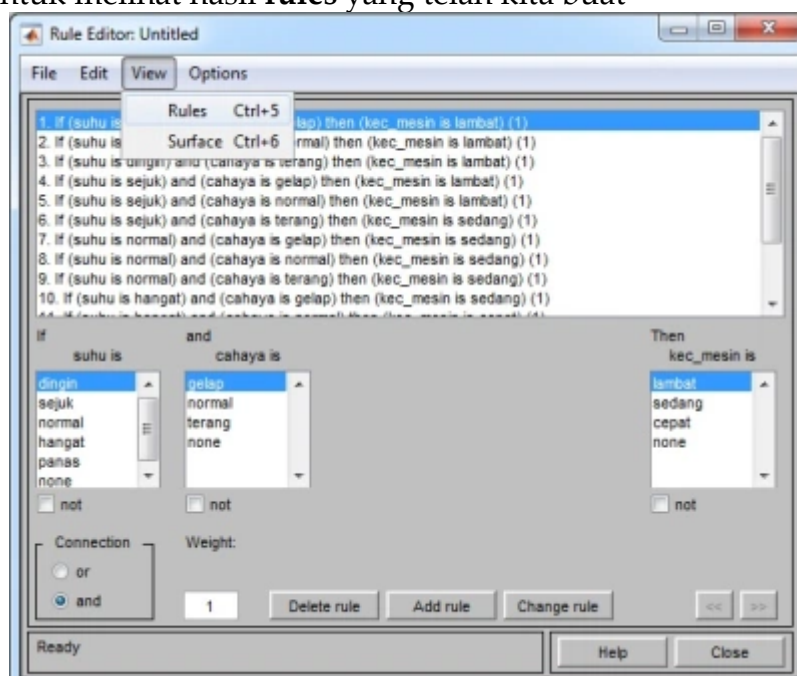
<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-rules.jpg>

buatlah aturan pada **rule editor** sesuai dengan konsep sistem kontrol yang sebelumnya telah dibuat. Misalnya if (suhu is dingin) and (cahaya is gelap) then (kec_mesin is lambat), dan seterusnya sampai dengan 15 rules.



<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-penambahan-aturan.jpg>

9. Pilih **view >> rules**, untuk melihat hasil **rules** yang telah kita buat



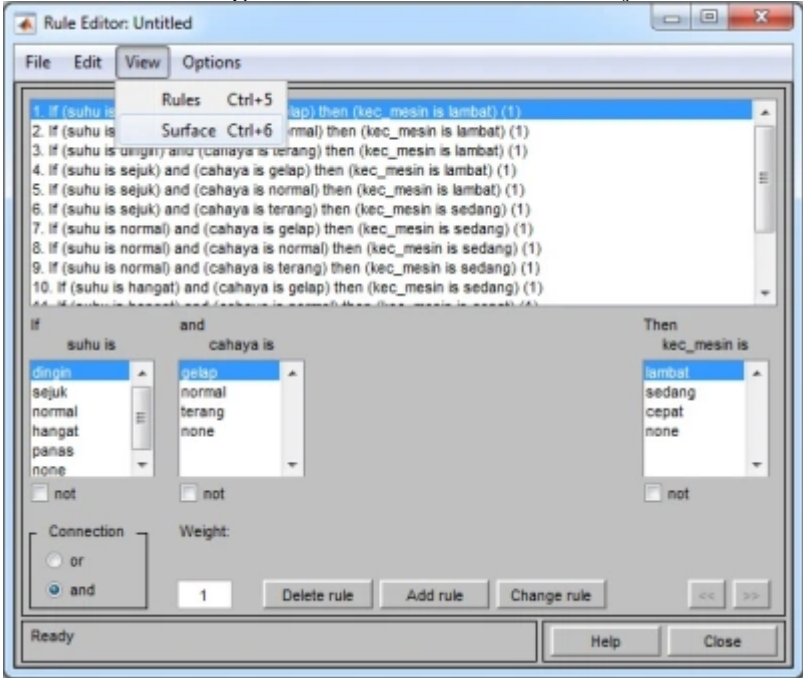
<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-visualisasi-aturan.jpg>

kita dapat menggeser-geser nilai **suhu (input1)** dan **cahaya (input2)** sehingga menghasilkan nilai keluaran pada **kec. mesin (output)**



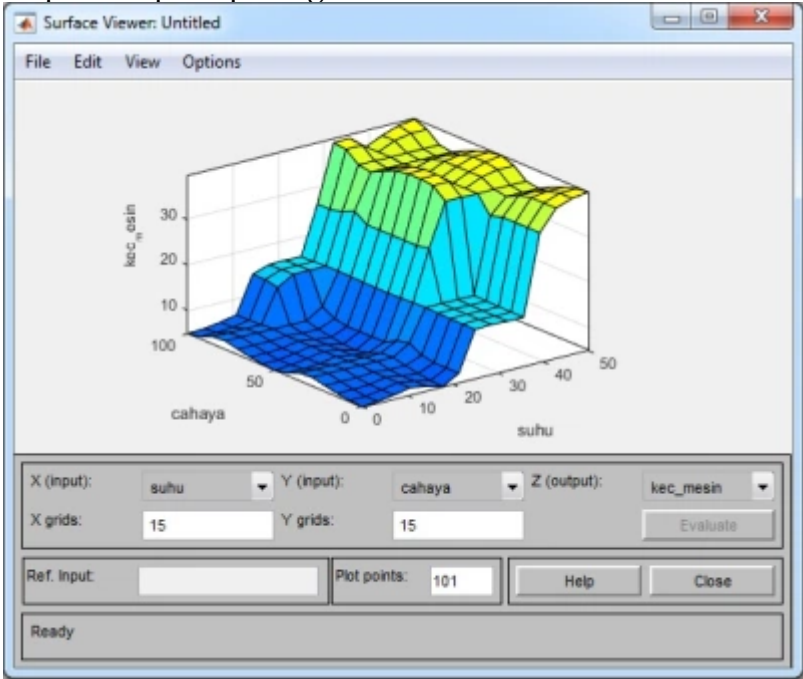
<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-tampilan-aturan-rules.jpg>

10. Pilih **view >> surface**, untuk melihat grafik 3D antara suhu, cahaya, dan kec. mesin



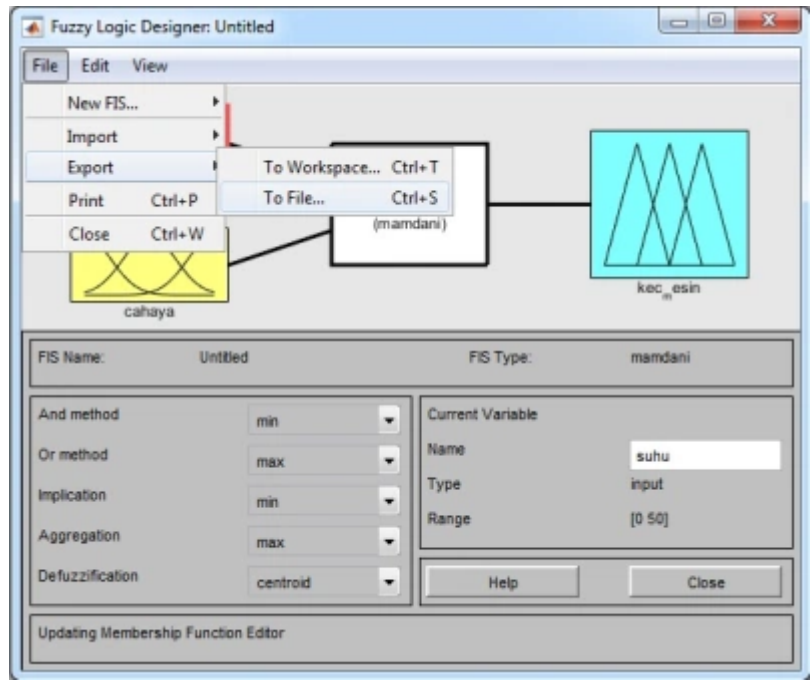
<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-tampilan-aturan-rules-2.jpg>

sehingga akan muncul tampilan seperti pada gambar di bawah ini



(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-tampilan-aturan-rules-3.jpg>)

11. Simpanlah **FIS** yang telah dibuat dengan cara mengklik **file >> export >> to file** misalnya simpan dengan nama **“mesin.fis”**



(<https://pemrogramanmatlab.files.wordpress.com/2015/08/tampilan-fuzzy-inference-system-editor-matlab-membership-function-editor-menyimpan-fis.jpg>)

12. Untuk mengecek hasil keluaran dari **FIS** yang telah dibuat, dapat kita lakukan dengan mengetik kode berikut pada **command window**:

```
fis = readfis('mesin');
output = evalfis([10 20],fis)
```

Hasilnya adalah:

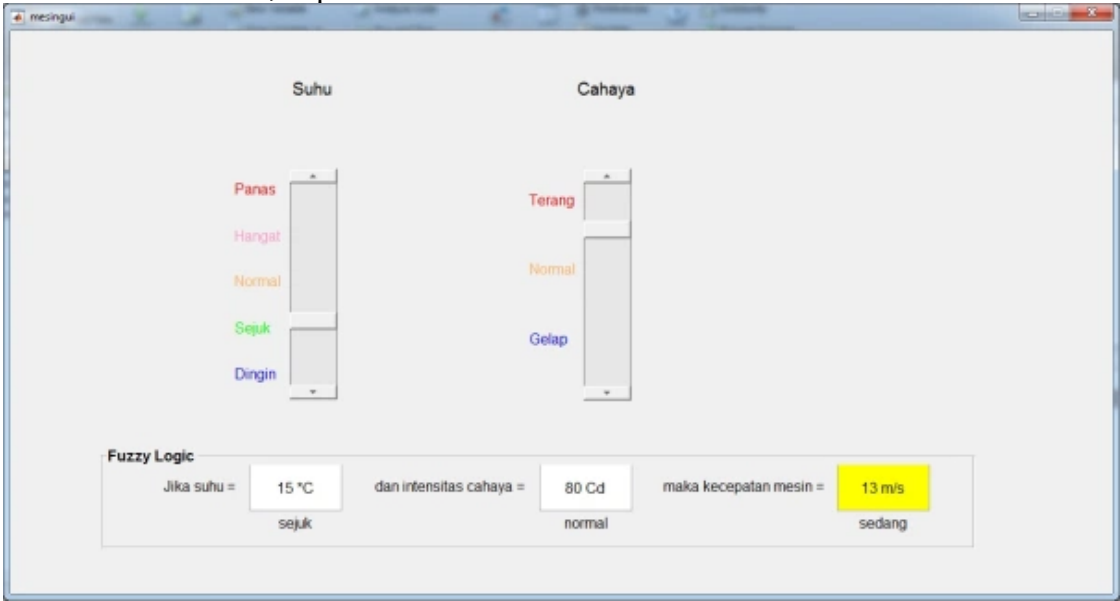
```
output =

6.2059
```

Nilai ini artinya: Jika **suhu = 10⁰C (dingin)** dan **cahaya = 20 Cd (gelap)**, maka **kec. mesin = 6.2059 m/s (lambat)**

Dapat disimpulkan bahwa hasil keluaran **FIS** sesuai dengan konsep sistem kontrol yang sebelumnya telah dibuat.

Kita dapat membuat tampilan program yang telah kita buat menjadi lebih interaktif menggunakan **Graphical User Interface (GUI)**, seperti contoh **GUI** di bawah ini :



(<https://pemrogramanmatlab.files.wordpress.com/2015/08/pemrograman-matlab-tampilan-gui-logika-fuzzy.jpg>)

Listing program nya adalah sebagai berikut:

```

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

% Last Modified by GUIDE v2.5 04-Oct-2013 22:13:32

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

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

% Update handles structure
guidata(hObject, handles);
movegui(hObject, 'center');
% UIWAIT makes mesingui wait for user response (see UIRESUME)
% uiwait(handles.figure1);

```

```

% --- Outputs from this function are returned to the command line.
function varargout = mesingui_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 slider movement.
function slider2_Callback(hObject, eventdata, handles)
% hObject      handle to slider2 (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,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of slider
slider_suhu = get(hObject,'Value');
set(handles.edit2, 'string', strcat(num2str(slider_suhu), ' *C'));

if slider_suhu <= 10
    suhu = 'dingin';
elseif slider_suhu > 10 && slider_suhu <= 20
    suhu = 'sejuk';
elseif slider_suhu > 20 && slider_suhu <= 27
    suhu = 'normal';
elseif slider_suhu > 27 && slider_suhu <= 35
    suhu = 'hangat';
else
    suhu = 'panas';
end

set(handles.text14, 'string', suhu);

slider_cahaya = get(handles.slider3,'Value');

input = [slider_suhu slider_cahaya];
fis = readfis('mesin');
out = evalfis(input,fis);

if out <= 10
    kec_mesin = 'lambat';
elseif out > 10 && out <= 18
    kec_mesin = 'sedang';
else
    kec_mesin = 'cepat';
end

set(handles.edit4,'string',strcat(num2str(out),' m/s'));
set(handles.text16, 'string', kec_mesin);

% --- Executes during object creation, after setting all properties.
function slider2_CreateFcn(hObject, eventdata, handles)
% hObject      handle to slider2 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.

```

```

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Executes on slider movement.
function slider3_Callback(hObject, eventdata, handles)
% hObject      handle to slider3 (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,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of slider
slider_cahaya = get(hObject,'Value');
set(handles.edit3, 'string', strcat(num2str(slider_cahaya),' Cd'));

if slider_cahaya <= 30
    cahaya = 'gelap';
elseif slider_cahaya > 30 && slider_cahaya <= 80
    cahaya = 'normal';
else
    cahaya = 'terang';
end

set(handles.text15, 'string', cahaya);

slider_suhu = get(handles.slider2,'Value');

input = [slider_suhu slider_cahaya];
fis = readfis('mesin');
out = evalfis(input,fis);

if out <= 10
    kec_mesin = 'lambat';
elseif out > 10 && out <= 18
    kec_mesin = 'sedang';
else
    kec_mesin = 'cepat';
end

set(handles.edit4,'string',strcat(num2str(out),' m/s'));
set(handles.text16, 'string', kec_mesin);

% --- Executes during object creation, after setting all properties.
function slider3_CreateFcn(hObject, eventdata, handles)
% hObject      handle to slider3 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
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,'defaultUicontrolBackgrou
    set(hObject,'BackgroundColor','white');
end

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,'defaultUicontrolBackgrou
    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,'defaultUicontrolBackgrou
    set(hObject,'BackgroundColor','white');
end

```

File source code lengkap logika fuzzy pada pemrograman di atas dapat diperoleh melalui halaman berikut ini: [Source Code \(https://pemrogramanmatlab.com/source-code-gui-matlab/\)](https://pemrogramanmatlab.com/source-code-gui-matlab/)

Save on August 24, 2015, in [Data mining](#), [Pengenalan Matlab](#), [Pengenalan Pola](#) and tagged [algoritma fuzzy untuk sistem kontrol](#), [aplikasi logika fuzzy menggunakan matlab](#), [cara membuat aplikasi gui matlab sederhana](#), [cara membuat fuzzy pada matlab](#), [cara membuat gui matlab logika fuzzy](#), [cara membuat logika fuzzy menggunakan matlab](#), [contoh fuzzy menggunakan matlab](#), [contoh logika fuzzy matlab](#), [contoh program aplikasi fuzzy matlab](#), [contoh program gui matlab](#), [contoh program gui matlab fuzzy](#), [contoh program logika fuzzy matlab](#), [contoh program matlab sederhana](#), [data mining logika fuzzy menggunakan matlab](#), [fuzzy logic](#), [fuzzy logic in matlab](#), [fuzzy matlab](#), [gui matlab untuk logika fuzzy](#), [logika fuzzy](#), [logika fuzzy menggunakan gui matlab](#), [logika fuzzy sederhana matlab](#), [logika fuzzy sederhana menggunakan gui matlab](#), [logika fuzzy untuk sistem kontrol](#), [tutorial fuzzy matlab](#), [tutorial logika fuzzy matlab](#). Bookmark the [permalink](#). 12 Comments.

- **Leave a comment**

- **Comments 12**

herry | October 13, 2016 at 6:04 pm
gan, kenapa gui nya eror ?

adi pamungkas | October 13, 2016 at 6:24 pm
Errornya gimana mas?
Karena sdh saya coba di komputer saya tdk error
Pakai matlab versi berapa?

David Center | October 22, 2016 at 11:30 pm
mas jika dipakai untuk sistem kendali suhu dan kelembapan bisa gak, dengan arduino n matlab

adi pamungkas | October 22, 2016 at 11:44 pm
matlab merupakan software yang bisa diintegrasikan dengan perangkat keras seperti arduino, raspberry, dll

Kukuh Priambodo | December 7, 2016 at 11:31 pm
sangat bermanfaat mas, terma kasih banyak

adi pamungkas | December 7, 2016 at 11:34 pm
Alhamdulillah
sama sama kukuh

al farisi | December 19, 2016 at 11:21 am
mas kok GUI nya gk bisa dijalankan
ini Pakai matlab versi berapa?

adi pamungkas | December 19, 2016 at 11:30 am
koding tsb bisa dijalankan menggunakan matlab r2015b

dadang | April 18, 2018 at 5:06 pm
permisi mas mau tanya kalau membuat rule base nya itu aturannya gimana punya saya kok eror

adi pamungkas | April 18, 2018 at 7:52 pm
Aturan rule base dapat dipelajari pada materi di atas

M. Nizal | April 20, 2018 at 9:11 am
Selamat pagi Mas skrg saya sedang mengerjakan tugas akhir untuk deteksi kebocoran pipa, disini saya menggunakan data dari perubahan sensor tekanan, untuk menentukan model svm yang dipakai dan cara membuat data latih dan data uji nya itu bagaimana ya mas, terima kasih mas sebelumnya

adi pamungkas | April 20, 2018 at 9:25 am

selamat pagi nizar

materi mengenai svm bisa dipelajari pada halaman berikut ini

<https://pemrogramanmatlab.com/data-mining-menggunakan-matlab/support-vector-machine-svm-menggunakan-matlab/>