





function varargout = mesh(varargin)

% MESH M-file for mesh.fig

% MESH, by itself, creates a new MESH or raises the existing

% singleton\*.

%

% H = MESH returns the handle to a new MESH or the handle to

% the existing singleton\*.

%

% MESH('CALLBACK',hObject,eventData,handles,...) calls the local

% function named CALLBACK in MESH.M with the given input arguments.

%

% MESH('Property','Value',...) creates a new MESH or raises the

% existing singleton\*. Starting from the left, property value pairs are

% applied to the GUI before mesh\_OpeningFcn gets called. An

% unrecognized property name or invalid value makes property application

% stop. All inputs are passed to mesh\_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 mesh

% Last Modified by GUIDE v2.5 02-Sep-2009 16:42:36

% Begin initialization code - DO NOT EDIT

gui\_Singleton = 1;

gui\_State = struct('gui\_Name', mfilename, ...

'gui\_Singleton', gui\_Singleton, ...

'gui\_OpeningFcn', @mesh\_OpeningFcn, ...

'gui\_OutputFcn', @mesh\_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 mesh is made visible.

function mesh\_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 mesh (see VARARGIN)

% Choose default command line output for mesh

handles.output = hObject;

set(hObject,'toolbar','figure');

% Update handles structure

guidata(hObject, handles);

% UIWAIT makes mesh wait for user response (see UIRESUME)

% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.

function varargout = mesh\_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;

function filename\_Callback(hObject, eventdata, handles)

% hObject handle to filename (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 filename as text

% str2double(get(hObject,'String')) returns contents of filename as a double

% --- Executes during object creation, after setting all properties.

function filename\_CreateFcn(hObject, eventdata, handles)

% hObject handle to filename (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 [x, y, z, varargout] = stlread(filename)

% This function reads an STL file in binary format into matrixes X, Y and

% Z, and C. C is optional and contains color rgb data in 5 bits.

%

% USAGE: [x, y, z, c] = stlread(filename);

%

% To plot use patch(x,y,z,c), or patch(x,y,z)

%

% Written by Doron Harlev modified and some comments added by Mahdi Yoozbashizadeh

if nargout>4

error('Too many output arguments')

end

use\_color=(nargout==4);

fid=fopen(filename, 'r'); %Open the file, assumes STL Binary format,fid = fopen(filename) opens the file filename for read access. (On PCs, fopen opens files for binary read access.) .

if fid == -1

error('File could not be opened, check name or path.')

end

ftitle=fread(fid,80,'uchar=>schar'); % Read file title = fread(fid) reads data in binary format from the file specified by fid into matrix ftitle.

%Open the file using fopen before calling fread. The fid argument is the integer file identifier obtained

%from the fopen operation. MATLAB reads the file from beginning to end, and then positions the file pointer

%at the end of the file (see feof for details).

%A = fread(fid, count) reads the number of elements specified by count. At the end of the fread, MATLAB sets the file pointer to the next byte to be read. A subsequent fread will begin at the location of the file pointer

num\_facet=fread(fid,1,'int32'); % Read number of Facets

fprintf('\nTitle: %s\n', char(ftitle'));

fprintf('Num Facets: %d\n', num\_facet);

% Preallocate memory to save running time

x=zeros(3,num\_facet); y=zeros(3,num\_facet); z=zeros(3,num\_facet);

if use\_color

c=uint8(zeros(3,num\_facet));

end

h = waitbar(0,'Please wait...');

for i=1:num\_facet,

norm=fread(fid,3,'float32'); % normal coordinates, ignored for now

ver1=fread(fid,3,'float32'); % vertex 1( for facet i)

ver2=fread(fid,3,'float32'); % vertex 2

ver3=fread(fid,3,'float32'); % vertex 3

col=fread(fid,1,'uint16'); % color bytes

if (bitget(col,16)==1 & use\_color)

r=bitshift(bitand(2^16-1, col),-10);

g=bitshift(bitand(2^11-1, col),-5);

b=bitand(2^6-1, col);

c(:,i)=[r; g; b];

end

x(:,i)=[ver1(1); ver2(1); ver3(1)]; % convert to matlab "patch" compatible format

y(:,i)=[ver1(2); ver2(2); ver3(2)];

z(:,i)=[ver1(3); ver2(3); ver3(3)];

if mod(i,floor(num\_facet/10))==0

waitbar(i/num\_facet,h);

end

end

if use\_color

varargout(1)={c};

end

fclose(fid);

close(h);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% --- Executes on button press in plot1.

function plot1\_Callback(hObject, eventdata, handles)

% hObject handle to plot1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

a = get(handles.filename,'String')

axes(handles.axes1)

[p,t,tnorm]=STL\_Import(a,1);

X=(p\*[1;0;0])';

Y=(p\*[0;1;0])';

Z=(p\*[0;0;1])';

set(handles.points\_output,'String',length(X));

[m,n] = size(t)

set(handles.triangles\_output,'String',m);

xextent=max(X)-min(X)

set(handles.x\_extent,'String',xextent)

yextent=max(Y)-min(Y)

set(handles.y\_extent,'String',yextent)

zextent=max(Z)-min(Z)

set(handles.z\_extent,'String',zextent)

[x, y, z] = stlread(a)

modelcolor=get(handles.color,'String')

plot3(x,y,z,modelcolor);

back=get(handles.backgroundcolor,'String')

whitebg(back)

%adds a title, x-axis description, and y-axis description

title('Wire Frame');

xlabel('X data');

ylabel('Y data');

zlabel('Z data');

%set(handles.triangles\_output,'String',length(X1));

guidata(hObject, handles); %updates the handles

% --- Executes on button press in plot2.

function plot2\_Callback(hObject, eventdata, handles)

% hObject handle to plot2 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

a = get(handles.filename,'String')

axes(handles.axes1)

axes(handles.axes1)

[p,t,tnorm]=STL\_Import(a,1);

X=(p\*[1;0;0])';

Y=(p\*[0;1;0])';

Z=(p\*[0;0;1])';

set(handles.points\_output,'String',length(X));

[m,n] = size(t)

set(handles.triangles\_output,'String',m);

xextent=max(X)-min(X)

set(handles.x\_extent,'String',xextent)

yextent=max(Y)-min(Y)

set(handles.y\_extent,'String',yextent)

zextent=max(Z)-min(Z)

set(handles.z\_extent,'String',zextent)

%set(handles.triangles\_output,'String',));

trisurf(t,X,Y,Z)

%adds a title, x-axis description, and y-axis description

title('Solid');

xlabel('X data');

ylabel('Y data');

zlabel('Z data');

guidata(hObject, handles); %updates the handles

function points\_output\_Callback(hObject, eventdata, handles)

% hObject handle to points\_output (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 points\_output as text

% str2double(get(hObject,'String')) returns contents of points\_output as a double

% --- Executes during object creation, after setting all properties.

function points\_output\_CreateFcn(hObject, eventdata, handles)

% hObject handle to points\_output (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 triangles\_output\_Callback(hObject, eventdata, handles)

% hObject handle to triangles\_output (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 triangles\_output as text

% str2double(get(hObject,'String')) returns contents of triangles\_output as a double

% --- Executes during object creation, after setting all properties.

function triangles\_output\_CreateFcn(hObject, eventdata, handles)

% hObject handle to triangles\_output (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 x\_extent\_Callback(hObject, eventdata, handles)

% hObject handle to x\_extent (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 x\_extent as text

% str2double(get(hObject,'String')) returns contents of x\_extent as a double

% --- Executes during object creation, after setting all properties.

function x\_extent\_CreateFcn(hObject, eventdata, handles)

% hObject handle to x\_extent (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 y\_extent\_Callback(hObject, eventdata, handles)

% hObject handle to y\_extent (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 y\_extent as text

% str2double(get(hObject,'String')) returns contents of y\_extent as a double

% --- Executes during object creation, after setting all properties.

function y\_extent\_CreateFcn(hObject, eventdata, handles)

% hObject handle to y\_extent (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 z\_extent\_Callback(hObject, eventdata, handles)

% hObject handle to z\_extent (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 z\_extent as text

% str2double(get(hObject,'String')) returns contents of z\_extent as a double

% --- Executes during object creation, after setting all properties.

function z\_extent\_CreateFcn(hObject, eventdata, handles)

% hObject handle to z\_extent (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 color\_Callback(hObject, eventdata, handles)

% hObject handle to color (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 color as text

% str2double(get(hObject,'String')) returns contents of color as a double

% --- Executes during object creation, after setting all properties.

function color\_CreateFcn(hObject, eventdata, handles)

% hObject handle to color (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 backgroundcolor\_Callback(hObject, eventdata, handles)

% hObject handle to backgroundcolor (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 backgroundcolor as text

% str2double(get(hObject,'String')) returns contents of backgroundcolor as a double

% --- Executes during object creation, after setting all properties.

function backgroundcolor\_CreateFcn(hObject, eventdata, handles)

% hObject handle to backgroundcolor (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 plot3.

function plot3\_Callback(hObject, eventdata, handles)

% hObject handle to plot3 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

a = get(handles.filename,'String')

axes(handles.axes1)

[p,t,tnorm]=STL\_Import(a,1);

X=(p\*[1;0;0]);

Y=(p\*[0;1;0]);

Z=(p\*[0;0;1]);

set(handles.points\_output,'String',length(X));

[m,n] = size(t)

set(handles.triangles\_output,'String',m);

xextent=max(X)-min(X)

set(handles.x\_extent,'String',xextent)

yextent=max(Y)-min(Y)

set(handles.y\_extent,'String',yextent)

zextent=max(Z)-min(Z)

set(handles.z\_extent,'String',zextent)

[x, y, z] = stlread(a)

m=get(handles.color,'String')

scatter3(X,Y,Z);

back=get(handles.backgroundcolor,'String')

whitebg(back)

%adds a title, x-axis description, and y-axis description

title('Dots');

xlabel('X data');

ylabel('Y data');

zlabel('Z data');

%set(handles.triangles\_output,'String',length(X1));

guidata(hObject, handles); %updates the handles

% --- Executes on button press in plot4.

function plot4\_Callback(hObject, eventdata, handles)

% hObject handle to plot4 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)