
```

function varargout = Question2(varargin)
% QUESTION2 MATLAB code for Question2.fig
%     QUESTION2, by itself, creates a new QUESTION2 or raises the
%     existing
%     singleton*.
%
%     H = QUESTION2 returns the handle to a new QUESTION2 or the
%     handle to
%     the existing singleton*.
%
%     QUESTION2('CALLBACK',hObject,eventData,handles,...) calls the
%     local
%     function named CALLBACK in QUESTION2.M with the given input
%     arguments.
%
%     QUESTION2('Property','Value',...) creates a new QUESTION2 or
%     raises the
%     existing singleton*. Starting from the left, property value
%     pairs are
%     applied to the GUI before Question2_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property
%     application
%     stop. All inputs are passed to Question2_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 Question2

% Last Modified by GUIDE v2.5 12-Nov-2019 17:50:33

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @Question2_OpeningFcn, ...
                  'gui_OutputFcn',  @Question2_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 Question2 is made visible.
function Question2_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 Question2 (see VARARGIN)

% Choose default command line output for Question2
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes Question2 wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = Question2_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 WienerFilter(hObject, handles)
    type = get(handles.listbox1, 'value');
    path = './Data\';
    switch type
        case 1
            path = [path 'lossfocus_'];
            ang = 0;
        case 2
            path = [path 'cameramotion_'];
            ang = 0;
        case 3
            path = [path 'objectmotion_'];
            ang = 90;
    end
    I1 = im2double(rgb2gray(imread([path '1.jpg'])));
    axes(handles.axes1);imshow(I1);
    I2 = im2double(rgb2gray(imread([path '2.jpg'])));
    axes(handles.axes2);imshow(I2);
    I3 = im2double(rgb2gray(imread([path '3.jpg'])));
    axes(handles.axes3);imshow(I3);
    len = floor(get(handles.slider1, 'value'));
    set(handles.text2, 'string', '###' + string(len));
    Var = get(handles.slider2, 'value');

```

```

set(handles.text3, 'string', '####' + string(Var));
PSF = fspecial('motion', len, ang);
estimated_nsr = Var / var(I1(:));
wnr1 = deconvwnr(I1, PSF, estimated_nsr);
imwrite(wnr1, [path '1_filter.jpg']);
axes(handles.axes4);imshow(wnr1);
estimated_nsr = Var / var(I2(:));
wnr2 = deconvwnr(I2, PSF, estimated_nsr);
imwrite(wnr2, [path '2_filter.jpg']);
axes(handles.axes5);imshow(wnr2);
estimated_nsr = Var / var(I3(:));
wnr3 = deconvwnr(I3, PSF, estimated_nsr);
imwrite(wnr3, [path '3_filter.jpg']);
axes(handles.axes6);imshow(wnr3);

% --- 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)

% --- 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)

% --- 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)

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject    handle to slider1 (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

% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider1 (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 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

% --- 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 selection change in listbox1.
function listbox1_Callback(hObject, eventdata, handles)
% hObject    handle to listbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
    WienerFilter(hObject, handles);
% Hints: contents = cellstr(get(hObject,'String')) returns listbox1
%         contents as cell array
%         contents{get(hObject,'Value')} returns selected item from
%         listbox1

% --- Executes during object creation, after setting all properties.
function listbox1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to listbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
%             called

% Hint: listbox controls usually have a white background on Windows.
%         See ISPC and COMPUTER.

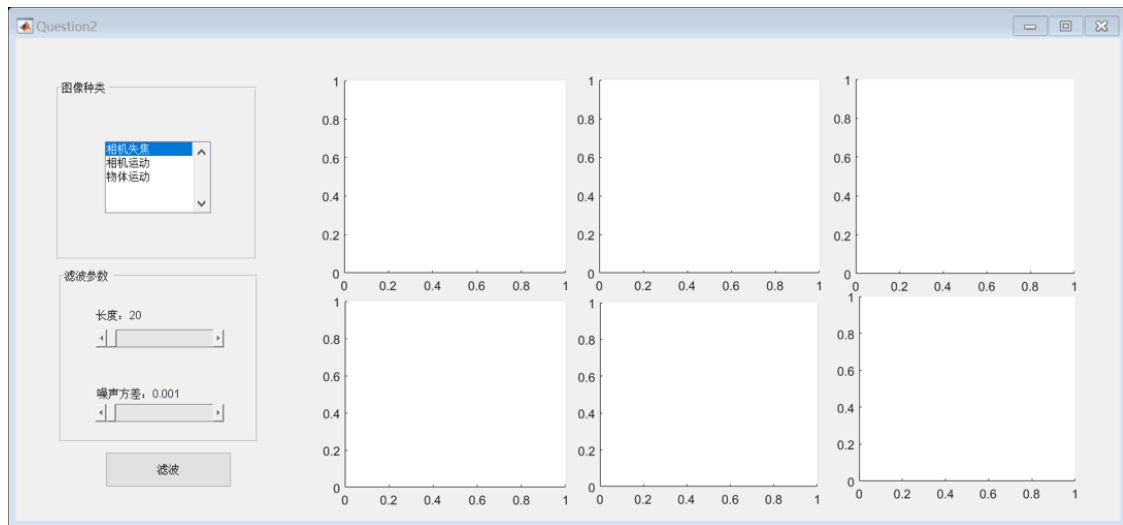
```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in 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)
    WienerFilter(hObject, handles);

```



Published with MATLAB® R2018b