USMAN INSTITUTE OF TECHNOLOGY

Affiliated with NED University of Engineering & Technology, Karachi

Department of Electrical Engineering

(Computer Systems Engineering)

CE430 – Digital Image Processing

Semester Assignment

Title: CEREMIC TILES QUALITY INSPECTION

Group Members:

MUHAMMAD ADEEN (20B-044-CE)
EMROZE MUGHAL (20B-046-CE) HINA
SAEED (20B-032-CE)

CODE :

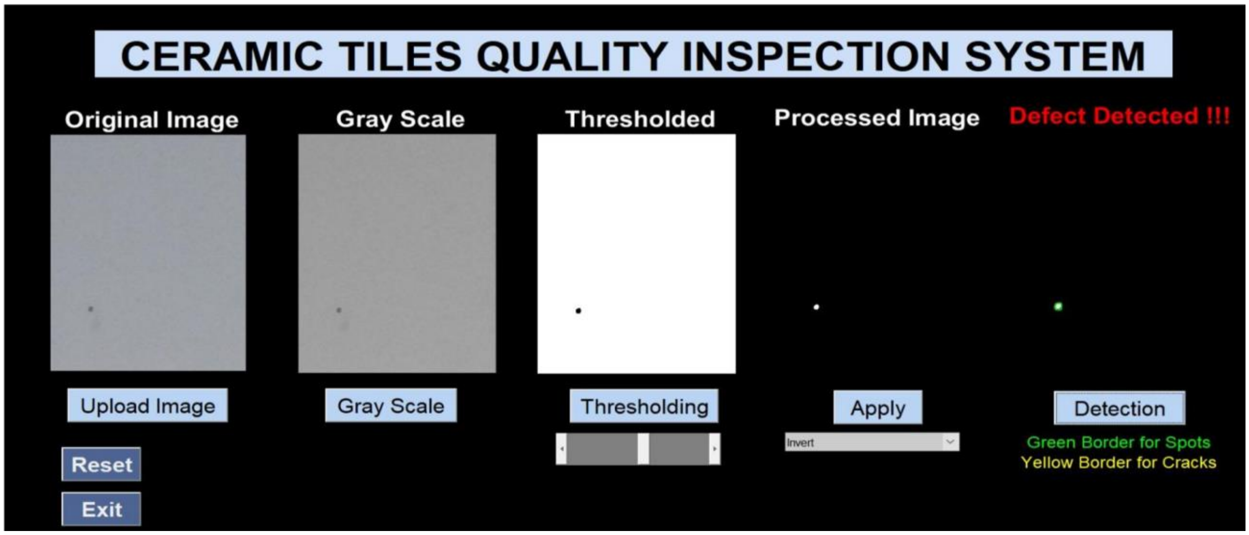
```
function varargout = Ceramic_Tiles_inpection_system(varargin)
% CERAMIC_TILES_INPECTION_SYSTEM MATLAB code for Ceramic_Tiles_inpection_system.fig
%      CERAMIC_TILES_INPECTION_SYSTEM, by itself, creates a new
CERAMIC_TILES_INPECTION_SYSTEM or raises the existing
%      singleton*.
%
%      H = CERAMIC_TILES_INPECTION_SYSTEM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

% Last Modified by GUIDE v2.5 22-Dec-2023 00:24:28

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Ceramic_Tiles_inpection_system_OpeningFcn, ...
                   'gui_OutputFcn',  @Ceramic_Tiles_inpection_system_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 Ceramic_Tiles_inpection_system is made visible.
function Ceramic_Tiles_inpection_system_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 Ceramic_Tiles_inpection_system (see VARARGIN)
% Choose default command line output for Ceramic_Tiles_inpection_system
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

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


% --- Outputs from this function are returned to the command line.
function varargout = Ceramic_Tiles_inpection_system_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 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 upload_image.
function upload_image_Callback(hObject, eventdata, handles)
% hObject    handle to upload_image (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global img1 img2 path
[path, nofile] = imgetfile();
if nofile
    msgbox (sprintf('Image not selected!!!'), 'Error','warning');
    return
end
img1= imread(path);
img1= im2double(img1);
img2= img1;
axes(handles.axes1);
imshow(img1);
```

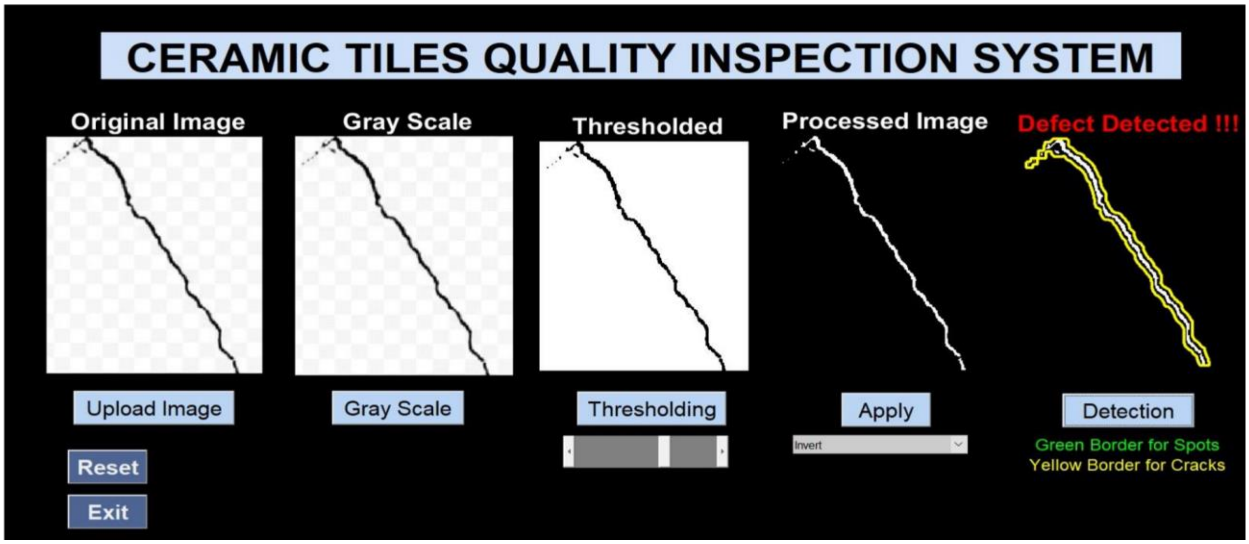
```
title('\fontsize{17}\color[rgb]{1,1,1} Original Image');

% --- Executes on button press in gray_scale.
function gray_scale_Callback(hObject, eventdata, handles)
% hObject    handle to gray_scale (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global img1
axes(handles.axes2);
img3=rgb2gray(img1);
imshow(img3)
title('\fontsize{17}\color[rgb]{1,1,1} Gray Scale');
axes (handles.axes2);


% --- Executes on button press in Thresholding.
function Thresholding_Callback(hObject, eventdata, handles)
% hObject    handle to Thresholding (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global img1
axes(handles.axes3);
k=rgb2gray(img1);
data = get(handles.slider2, 'Value');
bw = im2bw(k,data);
imshow(bw)
title('\fontsize{17}\color[rgb]{1,1,1} Thresholded');
axes(handles.axes3);



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


global img1
axes(handles.axes4);
if isempty(img1)
    disp('Please load an image first.');
    return;
end


grayImage = rgb2gray(img1);
bw = im2bw(grayImage,0.55);




k = get(handles.filters, 'Value');

switch k
    case 2
        processedImage = imcomplement(bw);
    case 3
        processedImage = edge(bw, 'sobel');
    case 4
        processedImage = edge(bw, 'prewitt');
    case 5
        processedImage = edge(bw, 'roberts');
    case 6
        processedImage = edge(bw, 'log');
    case 7
        processedImage = edge(bw, 'canny');
    otherwise
        msgbox (sprintf('Filter not selected!!!'), 'Error','warning');
        disp('No filter selected');
        return;
end

handles.processedImage = processedImage;

guidata(hObject, handles);

imshow(processedImage);
title(['\fontsize{17}\color[rgb]{1,1,1} Processed Image']);
axes(handles.axes4);

% --- Executes on button press in Detection.
function Detection_Callback(hObject, eventdata, handles)
% hObject    handle to Detection (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global img1
axes(handles.axes5);
k=rgb2gray(img1);
bw = im2bw(k,0.55);
invertedImage = imcomplement(bw);
label= bwlabel(invertedImage);
stats = regionprops(label,'Solidity', 'Area');
%density = [stats.Solidity];
area = [stats.Area];
%high_dense_area = density > 0;
%max_area =  (area(high_dense_area));
defect = find(area);
defectedarea= ismember(label, defect);
structure_element =strel('square',8);
defectedarea =  imdilate(defectedarea, structure_element);
Bound = bwboundaries(defectedarea, 'noholes');
imshow(invertedImage);
hold on

if ~isempty(Bound)
    crackThreshold = 100;
    spotThreshold = 20;
    numBoundaries = length(Bound);

    for i = 1:numBoundaries
        boundary = Bound{i};
        boundaryLength = size(boundary, 1);

        if boundaryLength > crackThreshold
            %Cracks
            plot(boundary(:, 2), boundary(:, 1), 'y', 'linewidth', 2);
            %set(handles.edit2, 'String', 'Cracks');
        elseif boundaryLength > spotThreshold
            %Spots
            plot(boundary(:, 2), boundary(:, 1), 'g', 'linewidth', 1);
            %set(handles.edit2, 'String', 'Spots');
        end
    end

    title('\fontsize{17}\color[rgb]{1,0,0} Defect Detected !!!');
```

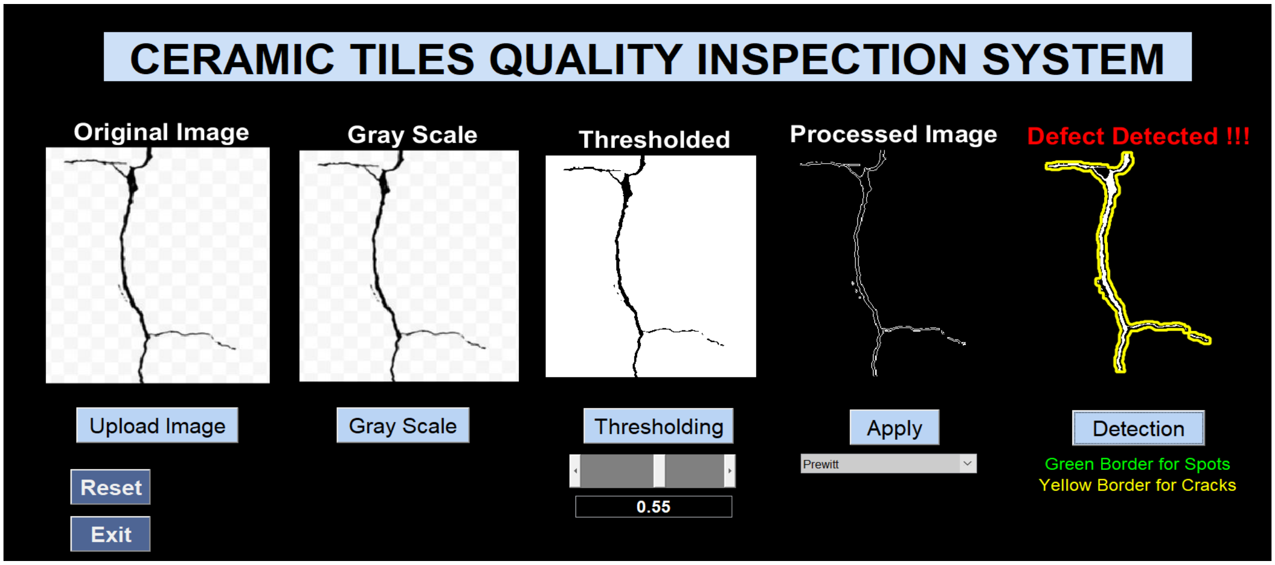
```matlab
else
    title('\fontsize{17}\color[rgb]{0,1,0} Accurate Tile!');
end
hold off;
axes (handles.axes5);


% --- Executes on mouse press over axes background.
function axes1_ButtonDownFcn(hObject, eventdata, handles)
% hObject    handle to axes1 (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 Exit.
function Exit_Callback(hObject, eventdata, handles)
% hObject    handle to Exit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close all;


% --- Executes on button press in Reset.
function Reset_Callback(hObject, eventdata, handles)
% hObject    handle to Reset (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
axes(handles.axes1);
hold off;
cla reset;
axes(handles.axes2);
hold off;
cla reset;
axes(handles.axes3);
hold off;
cla reset;
axes(handles.axes4);
hold off;
cla reset;
axes(handles.axes5);
hold off;
cla reset;



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 selection change in filters.
function filters_Callback(hObject, eventdata, handles)
% hObject    handle to filters (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns filters contents as cell
array
%        contents{get(hObject,'Value')} returns selected item from filters


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

% Hint: popupmenu 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 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
data = get(handles.slider2, 'Value');
data1=round(data,2)
data2=num2str(data1)
set(handles.edit3,'String',data2)

% --- 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



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
```

OUTPUT:







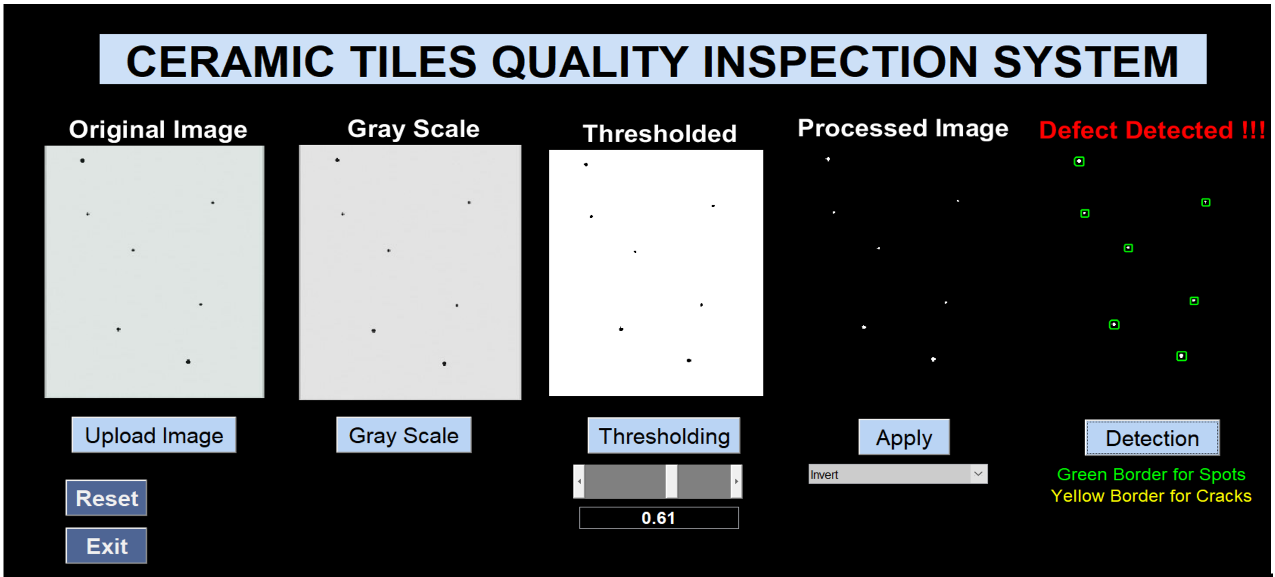
CERAMIC TILES QUALITY INSPECTION SYSTEM

Defect Detected !!!

0.55

Prewitt

Green Border for Spots
Yellow Border for Cracks



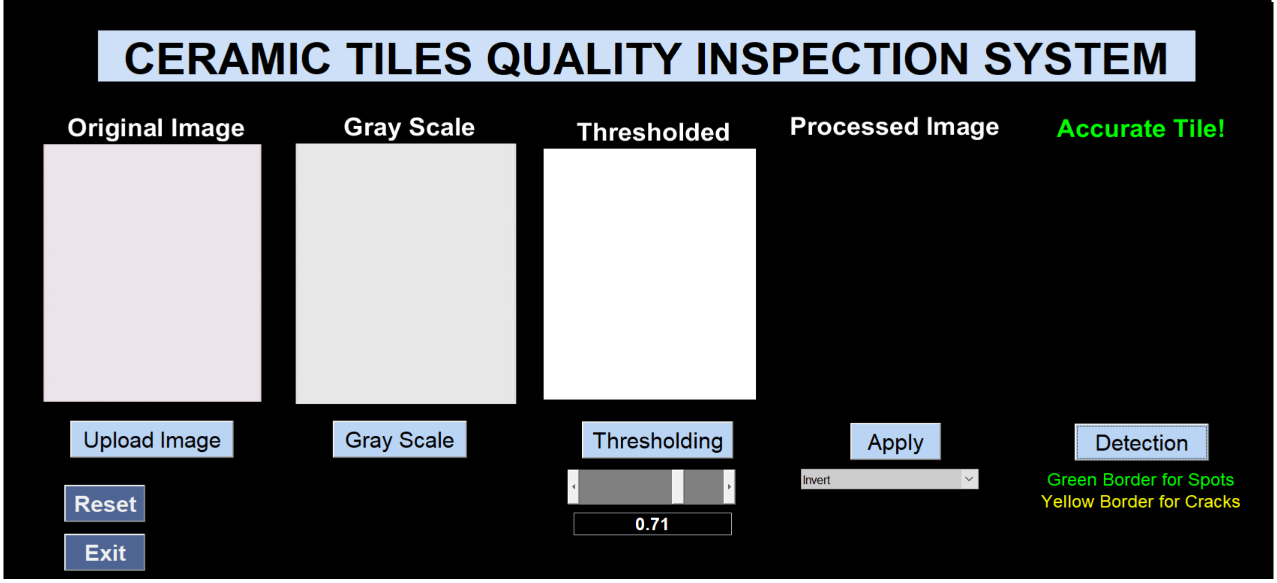
CERAMIC TILES QUALITY INSPECTION SYSTEM

Defect Detected !!!

0.61

Invert

Green Border for Spots
Yellow Border for Cracks



CERAMIC TILES QUALITY INSPECTION SYSTEM

Accurate Tile!

0.71

Invert

Green Border for Spots
Yellow Border for Cracks