

# CS446 Applied Digital Image Processing

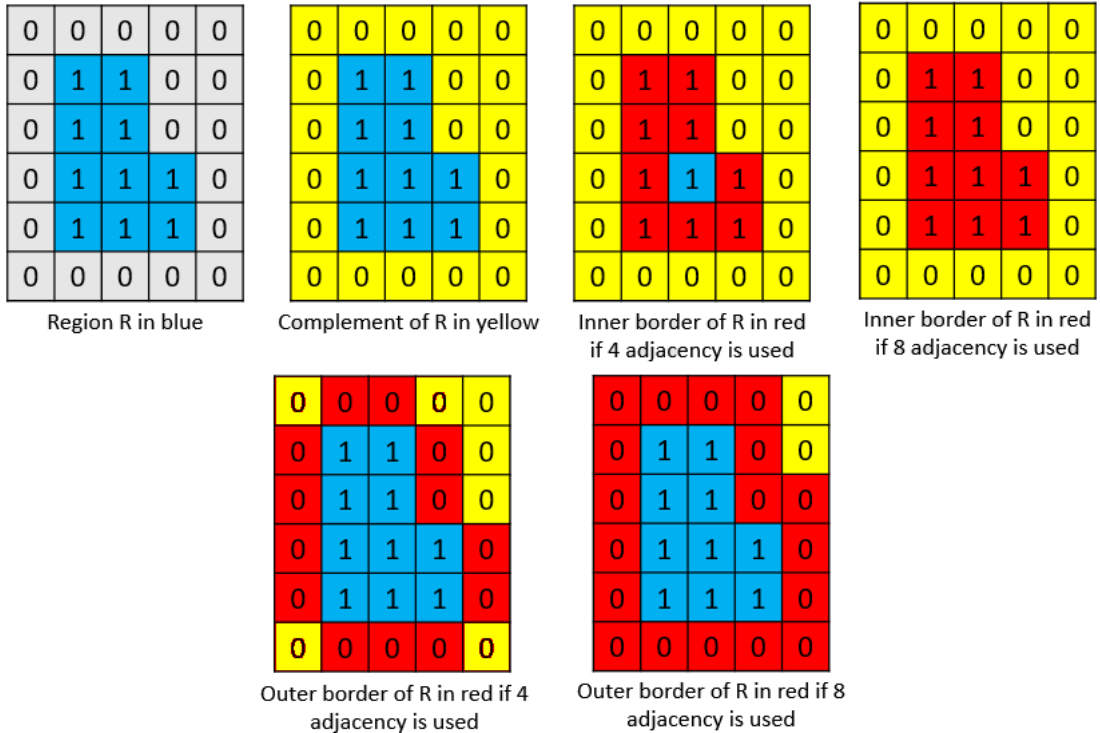
## Assignment 1:

CS Program  
Habib University

Fall 2023  
Due Date: 20 September 2023 @ 11:59PM

### 1 Introduction

We covered 4 and 8 connected neighbors and 4 and 8 adjacency to calculate boundaries. An example from our lecture notes is shown in Figure 1.



**Figure 1:** Boundary detection using 4 and 8 adjacency.

For this assignment, you are required to find and display the inner boundary in the given binary image **without using the `bwboundaries` function**. You should assume that the background pixels are 0 and foreground pixels are 1 and that there will always be a single connected region in the input image without holes. For generating a connected boundary contour, you will use the Radial Sweep Algorithm (Click on Radial Sweep Algorithm to access the webpage). Your script should not only generate the boundary points in order, it should also display them on the original figure as shown in the Sample Inputs and Outputs section.

For overlaying drawing on top of your image displayed using **`imshow`**, you can call **`hold on`**; and then make a call to your drawing functions. Once done, you should release the hold by calling the **`hold off`**; call. Supposing you have your boundary array *B* containing the list of 2D

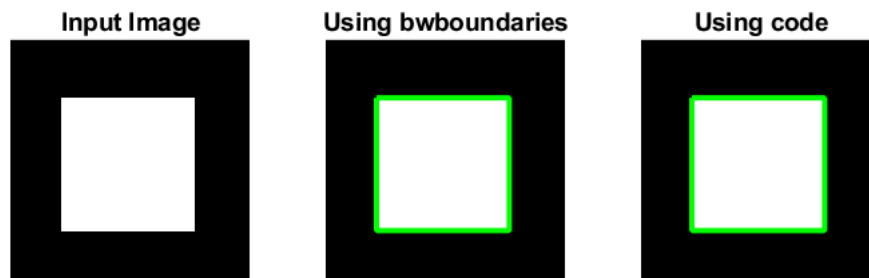
boundary points, you can use the following code snippet to display the boundary as an overlay on you image.

```
hold on
X=[];
Y=[];
for k = 1:length(B)
    x = B(k,1);
    y = B(k,2);

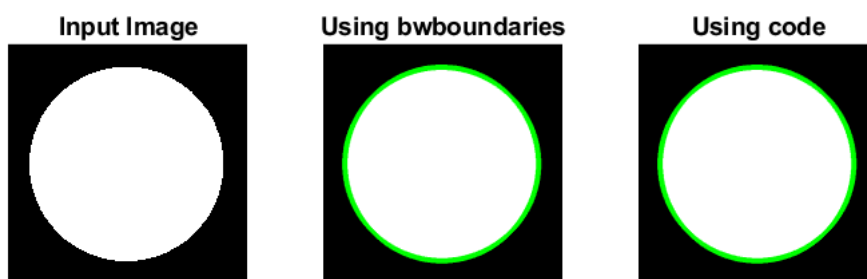
    X = [X, x];
    Y = [Y, y];
end
plot(X, Y, 'g', 'LineWidth', 2)
hold off
```

## 2 Sample Inputs and Outputs

Binary image will be input. Background pixels are 0 and foreground pixels are 1. Your code should show the input image, then the boundary obtained using `bwboundaries` and then the boundary obtained using your implementation of the radial sweep algorithm as shown in Figure 2 and Figure 3.



**Figure 2:** A filled square binary image.



**Figure 3:** A filled circle binary image.

## 3 Deliverables

Provide a Matlab script (.m) file or a Python script (.py) or a C/C++ source file (.c) containing your code. Rename the file to you Habib ID and upload to the Canvas submission module.

## 4 Marks Distribution

The assignment carries 100 marks. The details of breakage is as follows:

1. Radial Sweep Algorithm correctly implemented. (+30)
2. Viva questions were satisfactorily answered. (+30)
3. Code compiles and gives all correct outputs. (+20)
4. Code uses meaningful variable names, follows proper formatting and indentation, has sufficient code comments. (+20)

## 5 Using chatGPT or other AI software

You are not allowed to use any AI software to obtain the code for this assignment. Appropriate tool will be used to evaluate your submission for AI tool usage. If you are found using such a tool, you will be given a straight 0 and the Academic Conduct will be filed against you for academic dishonesty.

## 6 Plagiarism Policy

We have zero tolerance for plagiarism. Every submission will be screened using a plagiarism detection software. If there is any evidence of plagiarism, the case will be reported to the Office of Academic Conduct and all offenders will get a 0. This is applicable even for cases when the code is copied or a significant amount has been obtained from an online repository on open source platforms like bitbucket or github without proper attribution. In case you are taking any material from online sources, we expect that a proper credit/reference is given to the source.

## 7 Late Submission Policy

Late submission policy as per syllabus will be applied. That is a submission post deadline will have a 20% reduction in grade. An additional 10% reduction will be made per additional day requested beyond the deadline. This means that a late submission after a week will result in a score of 0.