Project 4 Write Up

Benjamin Berger
Bberger3@binghamton.edu

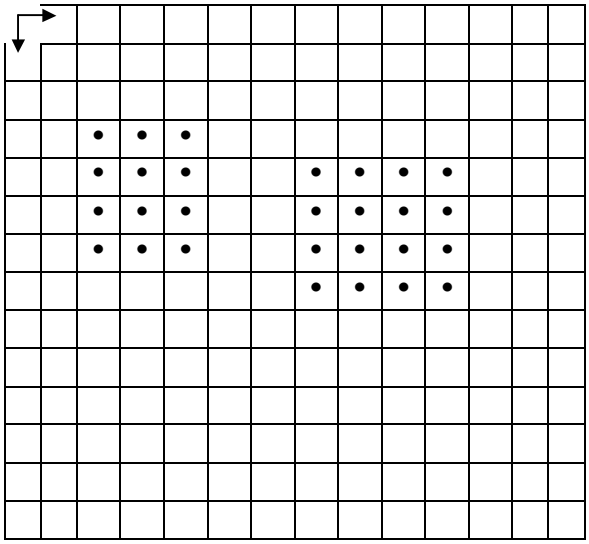
Methods explained below the programming section

Part A:

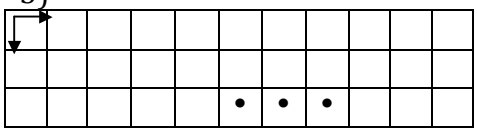
1)
$A \oplus B = \{Z \mid (B^)_z \cap A \neq \varnothing \}$
Becomes
$A \oplus B = \{Z \mid (B^)_z \cap A \subseteq A \}$

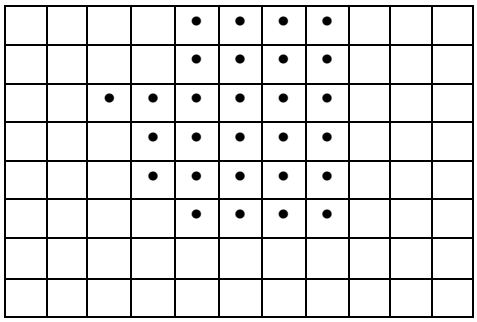
But we also know that
$A \oplus B = U_{b \in B} (A)_b$
And if
$A = \{a\}, B=\{b\}, A \oplus B = U_{b \in B} (A)_b$
Then
$A \oplus B = (A)_b U_{b \in B}$
Therefore
$B \oplus A == A \oplus B$

2)



3)





3) Assuming that ~ means reverse

4) a)

(~e) – (~f) + (~d) + [(a) + (~a)]

b)

[(~e) – (~f)] + [(~e) – (~f)]

Part B

./p4-1 TestImage.bmp Binary Image Threshold 200

./p4-1 First spacebar showing just erosion

Second spacebar showing just Dilation

Third spacebar showing Erosion then Dilation

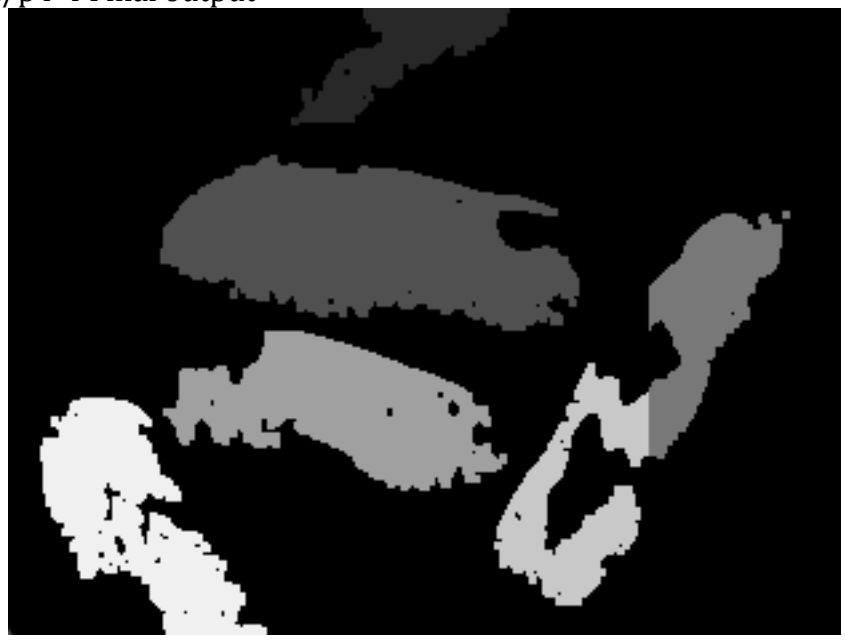
./p4-3 Applying the morphological operations to the original grey scale image. Not sure if this was correct in how to do it.

First spacebar showing just erosion

Second spacebar showing just Dilation.

Third spacebar showing Erosion then Dilation

./p4-4 Final output



Process:

The first thing I did was turn the image into a binary one. I tried the Otsu method but it wasn't working correctly. So then I tried various thresholds till I found on I liked.

After that I applied a 3x3 erosion filter to the image, followed by a 3x3 dilation one. This left me blocky images that I could work with.

Then I used recursive floodfill to label the different blocks. I then went through and eliminated any labels that had less than 1000 pixels. At this point I was about to count the remaining labels. The last picture is just me going through the labels and making them distinct.

Prints the console the number of pigs.