Stephen Ferro

MSAI 495

4/11/2024

Machine Problem 2

Introduction:

The goal of this assignment was to create functions to perform erosion, dilation, opening, and closing on binary images, as well as create a function for creating a boundary edge line.

Algorithms:

1. Erosion:
    a. Iterate through each pixel of the input image
    b. For each pixel, compare the surrounding pixels to the pixels of the structuring element/kernel, assuming the current pixel is located at the center of the kernel.
    c. If all white pixels in the kernel correspond to white pixels in the image, then the center pixel will be white in the output image.
    d. Repeat until all pixels are analyzed
2. Dilation
    a. Iterate through each pixel of the input image
    b. For each white pixel in the input image, set the values of the surrounding pixels in the output image according to the kernel. Assume the current pixel of the input image is located at the center of the kernel image.
    c. Repeat until all pictures are analyzed.
3. Opening
    a. Perform erosion on an input image using a specified kernel.
    b. Dilate the eroded image using the same kernel.
4. Closing
    a. Perform dilation on an input image using a specified kernel.
    b. Erode the dilated image using the same kernel.
5. Boundary

a. Perform erosion on the input image. For a 1 pixel wide boundary line, a 3x3 grid of pixels should be used as the kernel
b. Iterate through the pixels, comparing the input image and the eroded image. Points where the pixels in both images match should be set to black in the output. Pixels where the values from the two images differ should be set to white. This will generate a white boundary line in the output image.

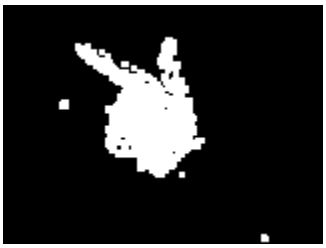
Results:

I used three kernels/structuring elements for this assignment, a 3x3, a 5x5, and a 7x7. See the end of this section for the exact kernel shape used.

Examples of each operation are shown below. I found I was able to get the cleanest image with the least noise by first closing the image with the large 7x7 kernel and then closing with the 5x5 kernel. That is the combination used in the examples below.

Erosion Using Kernel 1:


Dilation Using Kernel 1:

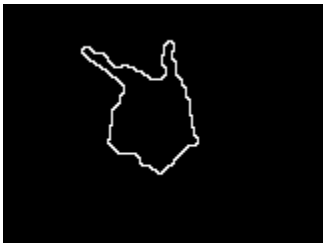



Opening Using Kernel 1:


Closing Using Kernel 1:


“Denoised” Image Using Closing with Kernel 3 Followed by Opening with Kernel 2:


Boundaries of "Denoised” Images:




kernel1 = ([[1,1,1],

[1,1,1],

[1,1,1]])

kernel2 = ([[0,1,1,1,0],

[1,1,1,1,1],

[1,1,1,1,1],

[1,1,1,1,1],

[0,1,1,1,0]])

kernel3 = ([[0,0,1,1,1,0,0],

[0,1,1,1,1,1,0],

[1,1,1,1,1,1,1],

[1,1,1,1,1,1,1],

[1,1,1,1,1,1,1],

[0,1,1,1,1,1,0],

[0,0,1,1,1,0,0]])