Tim Herrmann
EECS 332
MP2
Morphological Operators

Programming Language:

- Python 2.7.6 with the following dependencies
  - matplotlib
  - numpy
  - cv2

Function Inputs and Outputs:

Function	Inputs	Output	Example
dilate	Binary image Structure element (5x5) Output filename	An image dilated using the specified structure element	dilate(ImageIn,se,ImageOut)
erode	Binary image Structure element (5x5) Output filename	An image eroded using the specified structure element	erode(ImageIn,se,ImageOut)
opening	Binary image Structure element (5x5) Output filename	An image first eroded, then dilated by the specified structure element	opening(ImageIn,se,ImageOut)
closing	Binary image Structure element (5x5) Output filename	An image first dilated, then eroded by the specified structure element	closing(ImageIn,se,ImageOut)
boundary	Binary image Structure element (5x5) Output filename	An image representing the difference between the original image and the image eroded by the specified structure element	boundary(ImageIn,se,ImageOut)

Methods:

Dilate.py and Erode.py – These functons start by reading in the InputImage and the Structure Element. The inputs can be passed as strings representing the filenames or as images that have already been read in as numpy arrays. Next, it calls the helper function “readse” which takes the structure element and returns a list of all the “bright” pixels in the SE. Third, it creates a separate binary array that is a copy of the original image. This is done to save the original image state while operations are performed. Finally, the function will translate the SE onto each pixel in the copied image.

For dilate, it checks to see if any of the pixels included in the translated SE are “bright”. If it finds any “bright” pixels, it will change that pixel in the original image to white.

For erode, it checks to see if any of the pixels included in the translated SE are “dark”. If it finds any “dark” pixels, it sets the pixel in the original image to black

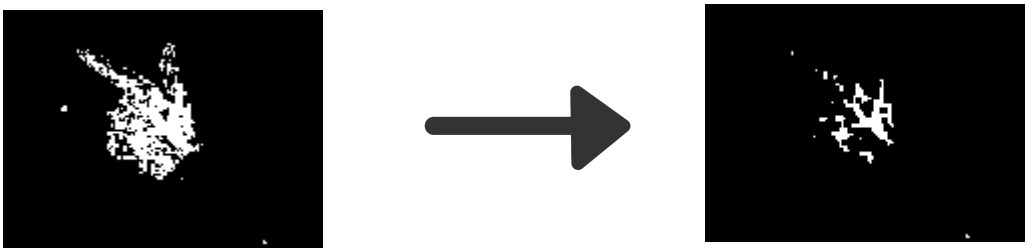
The updated images are written to the specified output filename and also returned by the function

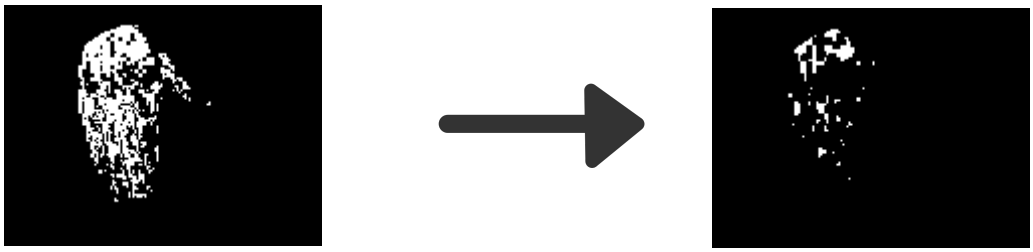
Opening.py and Closing.py – These functions simply make use of the dilate and erode functions above. Opening will first erode using the specified SE, then dilate. Closing will dilate then erode.

Boundary.py – This functions performs the same erode function as above, but instead of returning the eroded image, it returns an image that is the difference of the original image and the eroded image. Different SEs can be used to generate different boundary sizes.

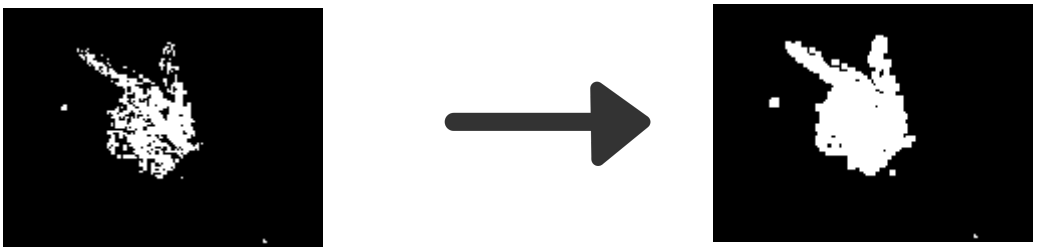
EXAMPLES

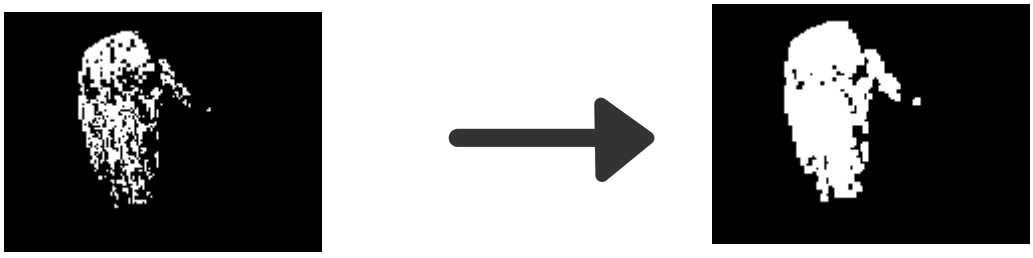
Erosion with a 3x3 SE



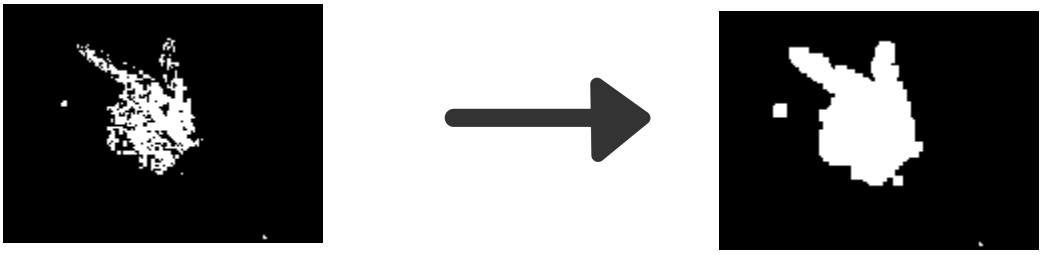


Dilation with a 3x3 SE

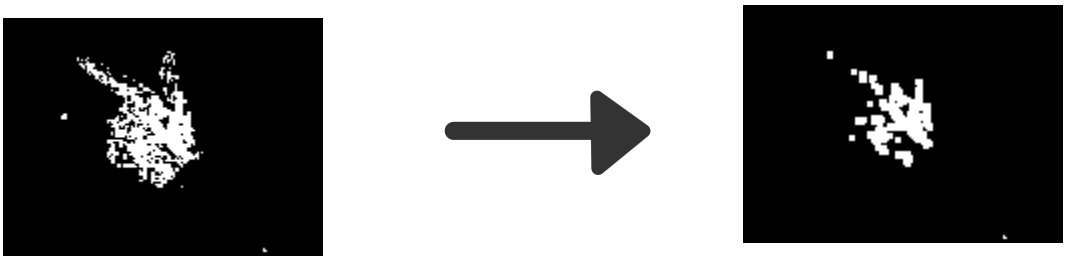


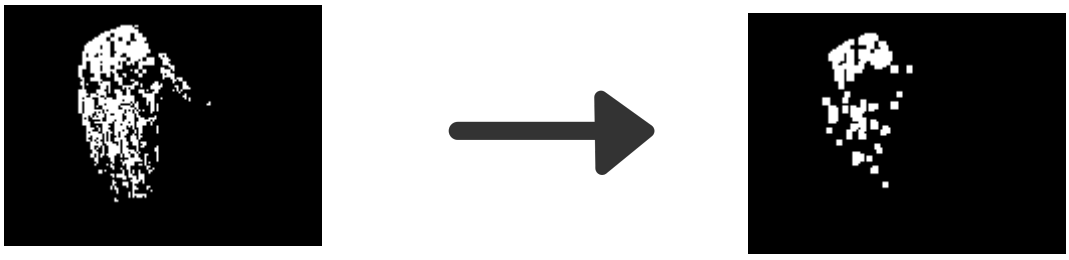


Dilation with a 5x5 SE

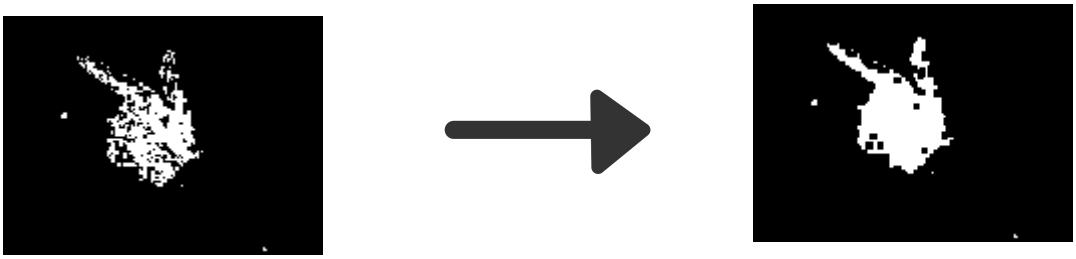


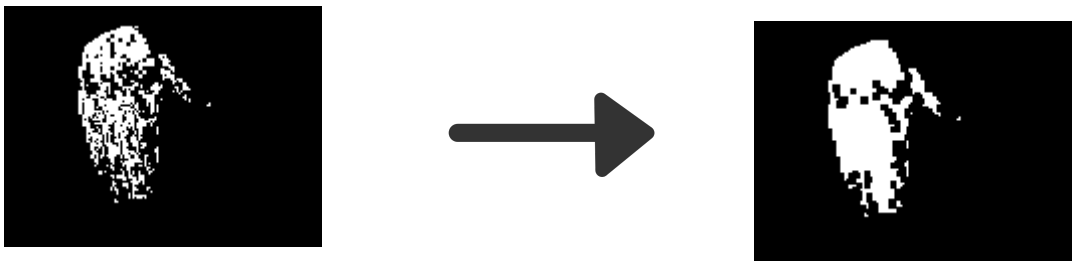
Opening with a 3x3 SE



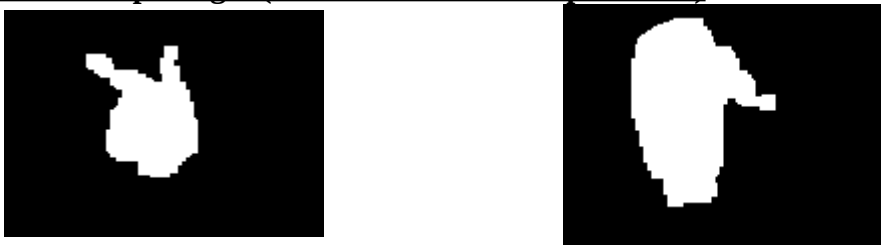


Closing with a 3x3 SE





Cleaned up images (combination of all 4 operations)



Boundaries (using cleaned up images)