Venkatakrishnan Ranganathan
MP2

Morphological operations

The given code performs morphological operations on binary images using the dilation, erosion, opening, , boundary detection and closing operations. These operations are used to enhance and manipulate binary images for various applications such as object detection, segmentation, and noise reduction.

The code takes in binary images in the form of a 2D numpy array and applies the specified operations using kernel sizes and padding values. The dilation operation is applied using a kernel of size 13, while the erosion operation uses a kernel of size 5. The opening operation is a combination of erosion followed by dilation, while the closing operation is the opposite, dilation followed by erosion.

The dilation operation expands the boundaries of the binary image by setting the value of a pixel to 1 if any of its neighboring pixels in the kernel are also 1. This operation helps in filling gaps between objects and increasing the size of objects.

The erosion operation, on the other hand, shrinks the boundaries of the binary image by setting the value of a pixel to 0 if any of its neighboring pixels in the kernel are 0. This operation helps in removing noise and small details from the image.

The opening operation is useful for removing small details and noise from the binary image while preserving the shape and size of larger objects. The closing operation is useful for filling small gaps between objects and smoothing the edges of objects.

The boundary following algorithm is for finding the boundaries of objects in a binary image. The algorithm follows the edges of the foreground pixels using an iterative approach that starts from an initial pixel and moves to the next neighboring pixel in a clockwise direction. The algorithm continues until it returns to the initial pixel, forming a closed loop of boundary pixels. The boundary pixels are collected into a list and returned as the output of the function.
The function takes a 2D NumPy array as input, representing a binary image with 1's for foreground pixels and 0's for background pixels.

The implementation creates a binary array to keep track of visited pixels and loops through each pixel in the binary image. If the pixel is a foreground pixel and has not been visited, the function starts a new boundary list and follows the boundary of the object using the boundary following technique. When the boundary is complete, the function appends the list of boundary pixels to a list of boundaries.

One improvement that can be made to the code is to add more flexibility to the kernel size and shape. Currently, the code uses fixed kernel sizes and shapes, which may not be suitable for all images and applications. Adding the ability to specify custom kernel sizes and shapes would make the code more versatile and adaptable.

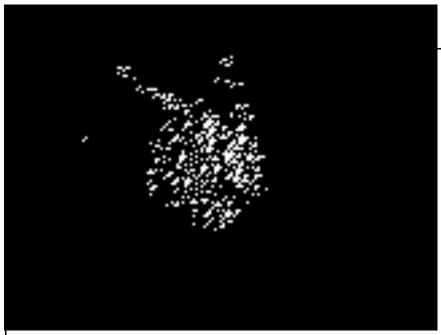
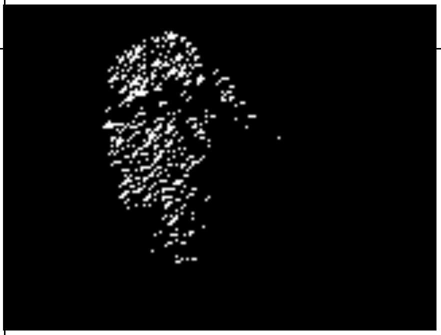
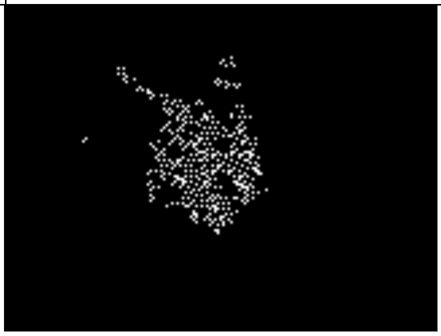
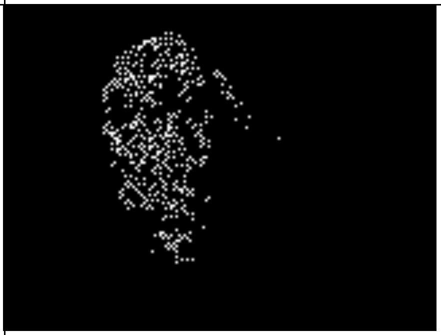
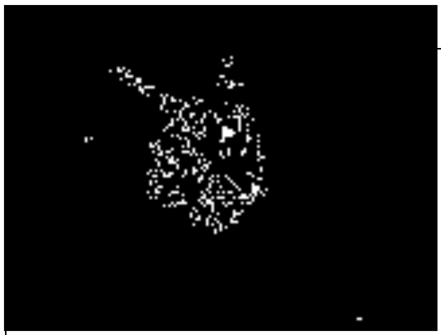
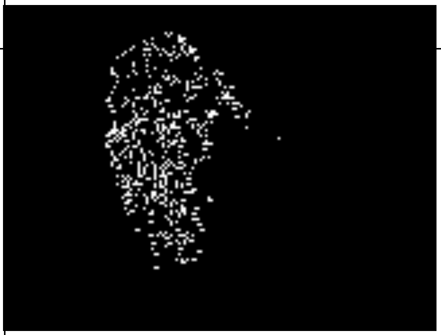
Venkatakrishnan Ranganathan
MP2

Another improvement could be to incorporate different structuring elements for the morphological operations. Currently, only rectangular structuring elements are used, but other shapes such as circular or elliptical structuring elements could also be useful for certain applications.

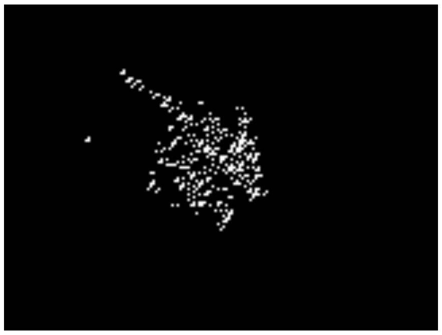
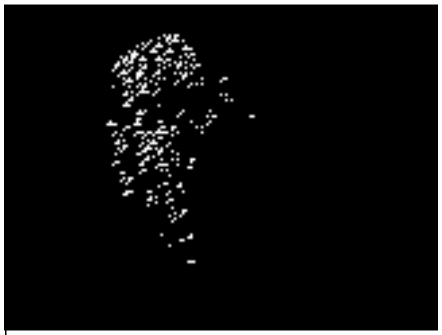
One potential improvement to the implementation is to add an optional argument to save the boundary images as PNG files. The modified implementation would create a new binary image with the same shape as the input image, set the boundary pixels to white, invert the colors of the boundary image, and save it as a PNG file. This would be useful for visualizing the boundaries of objects in an image and could be helpful for debugging or analysis.

Venkatakrishnan Ranganathan
MP2

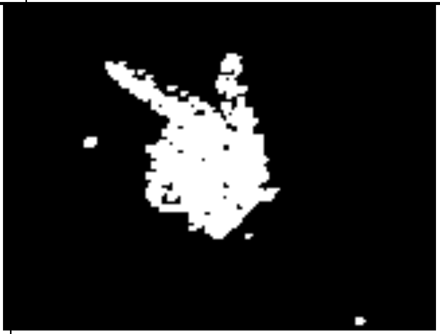
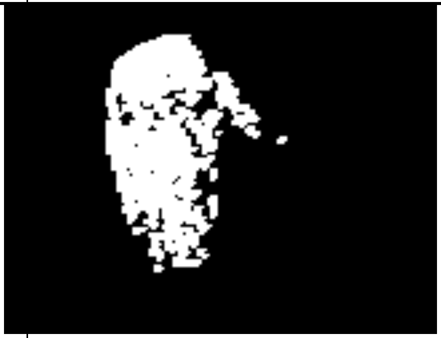
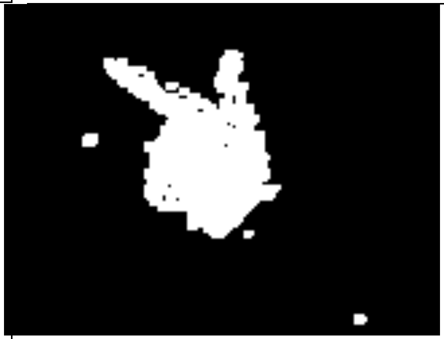
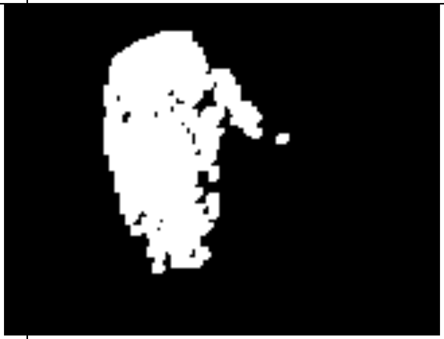
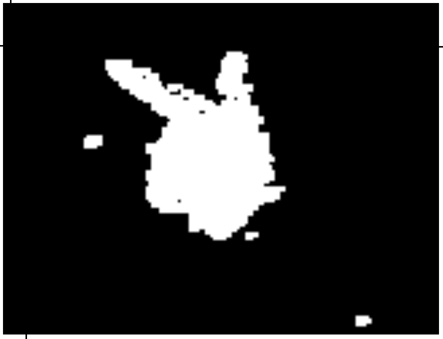
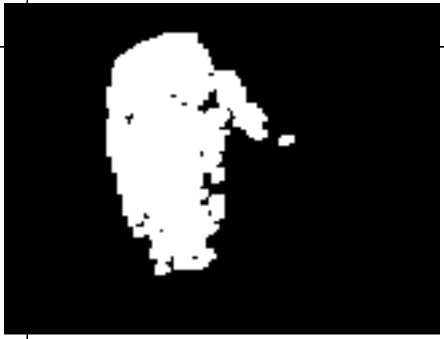
Erosion

Kernel	gun.bmp	palm.bmp
[1,1,0], [0,1,0], [0,0,0]		
[0,1,0], [1,1,0], [0,0,0]		
[1,1,1], [0,0,0], [0,0,0]		

Venkatakrishnan Ranganathan
MP2

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

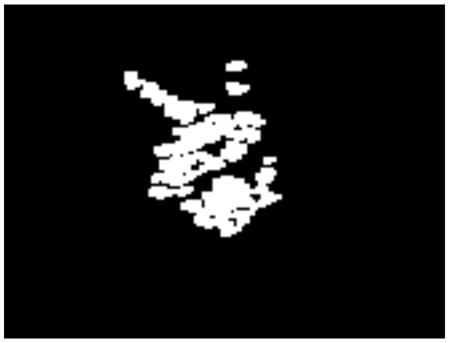
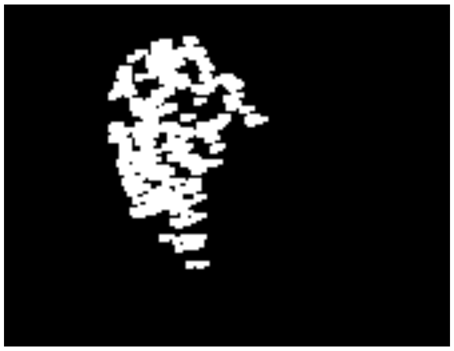
Dilation:

Kernel Size	gun.bmp	palm.bmp
3		
4		
5		

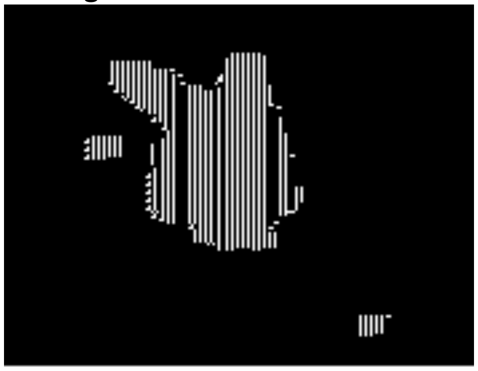
Venkatakrishnan Ranganathan
MP2

7

Opening:

Closing:



Boundary:
Result Unclear- I will try to resubmit if I get it right.