

Assignment-4

Patel Vrajesh (20110134)

Background: The process of partitioning a digital image into multiple image segments is known as image segmentation. The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Some common applications of image segmentation include medical imaging, video redaction software, and traffic light systems. Out of the several methods for image segmentation, this assignment will focus on using the Random Walker algorithm.

Problem Statement: Implement the Random Walker Algorithm for Image Segmentation.

My understanding of the algorithm:

Understanding of Algorithm:

Objective: To segment/split image into different classes based on color, background, foreground etc.

Algorithm:

- First we label some pixels in a given image. (say N_L)
- Now we will have some unlabelled pixels that we need to segment (say N_{UL})
- Initialize N_{UL} random walkers from the unlabelled pixels.
- Let the random walkers traverse the grid until they reach one of the labelled pixels.
- The starting pixel of the random walk gets the label of the labelled pixel it first reached.
- The more similar the pixel, the higher the probability. This is where the right choice of β (beta) matters.
- We compare cumulative probability of travelling in a direction to choose the direction where we traverse.

Number of Segments → Depends on the image.

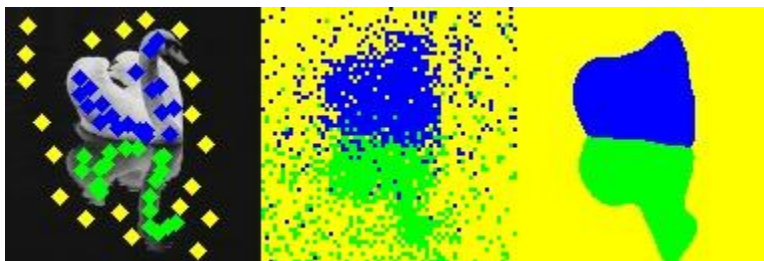
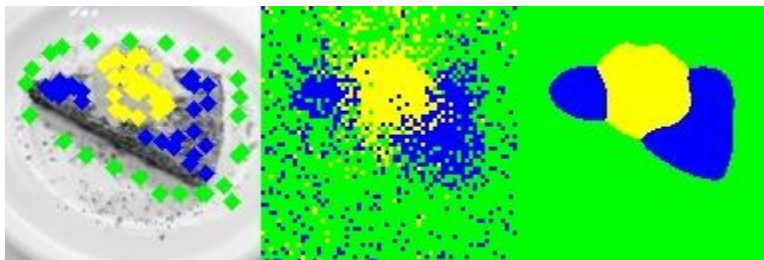
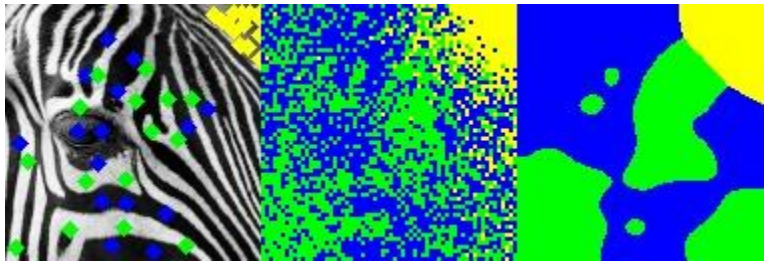
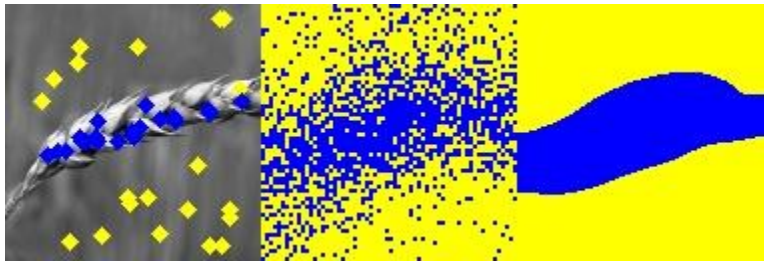
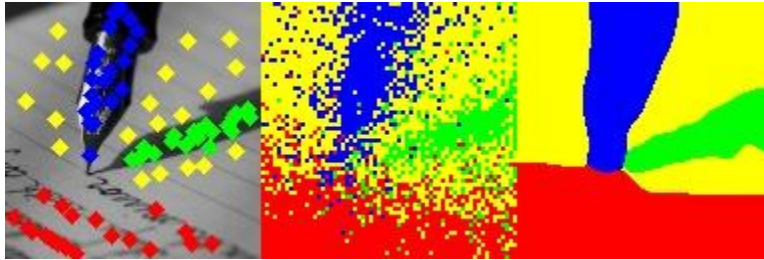
Labeled pixels for each segment → 25

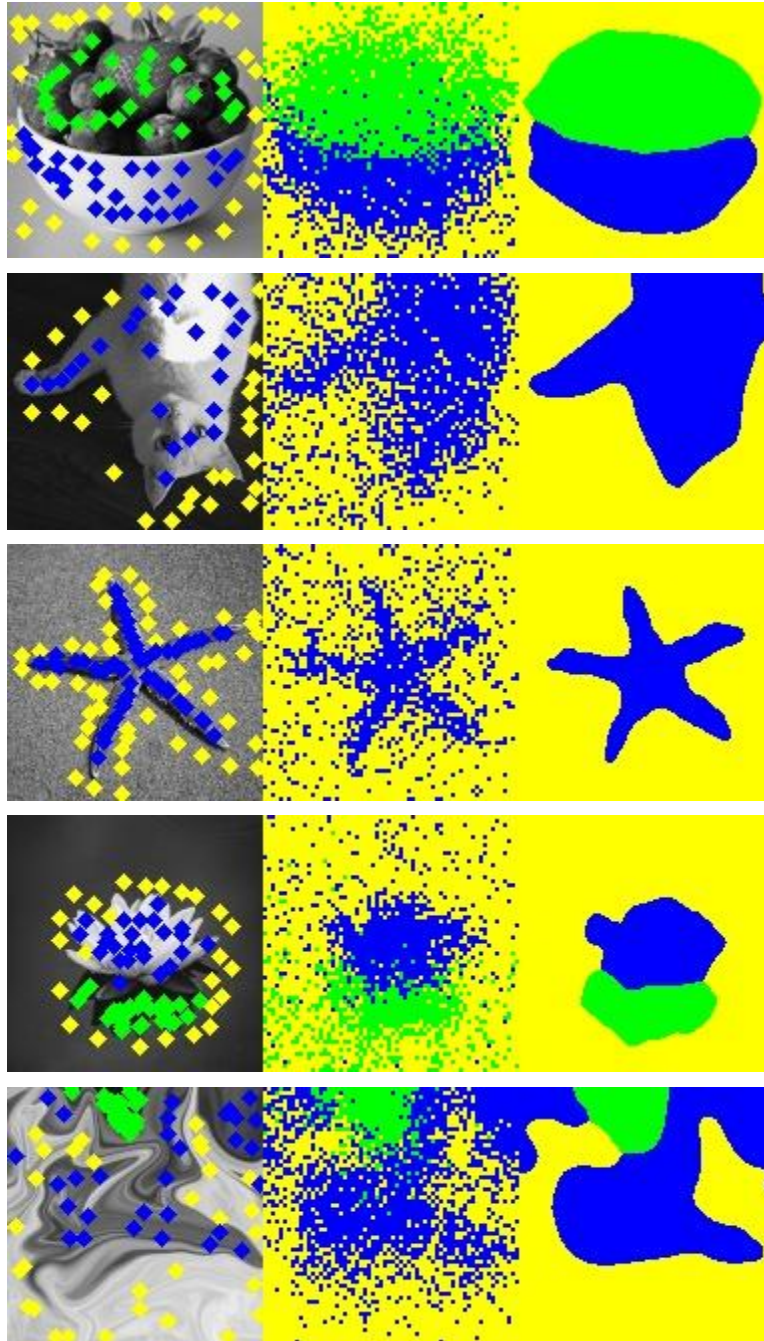
Beta → 10

Input Image

My_Output

Builtin_Output





```
File "c:\Users\vraje\OneDrive\Desktop\Assignment4\my_random_walker.py", line 148, in absolute_mean_error
    err = np.sum(abs(imageA-imageB))
ValueError: operands could not be broadcast together with shapes (128,128,3) (0,)
PS C:\Users\vraje\OneDrive\Desktop\Assignment4> & C:/Users/vraje/AppData/Local/Programs/Python/Python310/python.exe c:/Users/vraje/OneDrive/Desktop/Assignment4/my_random_walker.py
Enter the image number you want to segment: 1
Enter the number of segments? (Maximum 4): 2
Enter the number of pixels being marked?: 20
Enter the value of beta: 10
Mark pixels for segment 0
Mark pixels for segment 1
53.15093994140625
PS C:\Users\vraje\OneDrive\Desktop\Assignment4>
```

The above image shows the absolute mean error of the image1. The same can be found out for all the images in the dataset. The **comparison** between the images is already shown above.

Conclusion:

As it is evident from the above images, the output from the builtin_output and myoutput are almost the same. The accuracy of the output depends on the values of beta and the number of pixels we label for each of the images. The value of beta chosen is 10 and the number of pixels labeled for each of the segments is roughly 25 for all the images. The number of segments the image is divided into depends on the number of different classes present in the image.

References:

1. https://github.com/emmanuelle/random_walker
2. https://github.com/scikit-image/scikit-image/blob/main/skimage/segmentation/random_walker_segmentation.py
3. https://scikit-image.org/docs/stable/auto_examples/segmentation/plot_random_walker_segmentation.html
4. <https://github.com/gihanjayatilaka/EM509-image-segmentation-using-random-walks>
5. https://scikit-image.org/docs/stable/auto_examples/segmentation/plot_random_walker_segmentation.html