Part (a): Mean Shift Segmentor

The Mean Shift Segmentor implementation is carried out using the PyrMeanShiftFiltering function. The parameters required for working with this function is are:

Parameters	
src	The source 8-bit, 3-channel image.
dst	The destination image of the same format and the same size as the source.
sp	The spatial window radius.
sr	The color window radius.
maxLevel	Maximum level of the pyramid for the segmentation.
termcrit	Termination criteria: when to stop meanshift iterations.

Source:
http://docs.opencv.org/3.1.0/d4/d86/group__imgproc__filter.html#ga9fabdce9543bd602445f5db3827e4cc0

For our case the spatial window radius and the color window radius are the parameters to be experimented with. The maximum level of the pyramid for segmentation is always set to 1 as suggested by the question.

The term criteria for our case is again set up to 5 max iterations with the accuracy for difference in the epsilon set to 1 by default. This is essential for the movement of the Region of Interest with regards to the attraction basin.

METHOD: Mean Shift seeks modes or local maxima of density in local space. Clustering is done such that all the data points are the attraction basin of a mode. Attraction basin specifies the region to which all the trajectories lead to the mode.

We create a set of superpixels by first initialising the windows at each of the features and performing mean shift for each window until convergence. We then merge the pixels that end up in the same catchment or attraction basin.

EFFECT OF PARAMETERS: Since the mean shift is a vector which is the difference between the Gaussian valued region of interest or ROI and the centre of mass, the size of the Gaussian mask is an important criterion for the segmentation results. This size directly affects the convergence rate and the number of iterations needed.

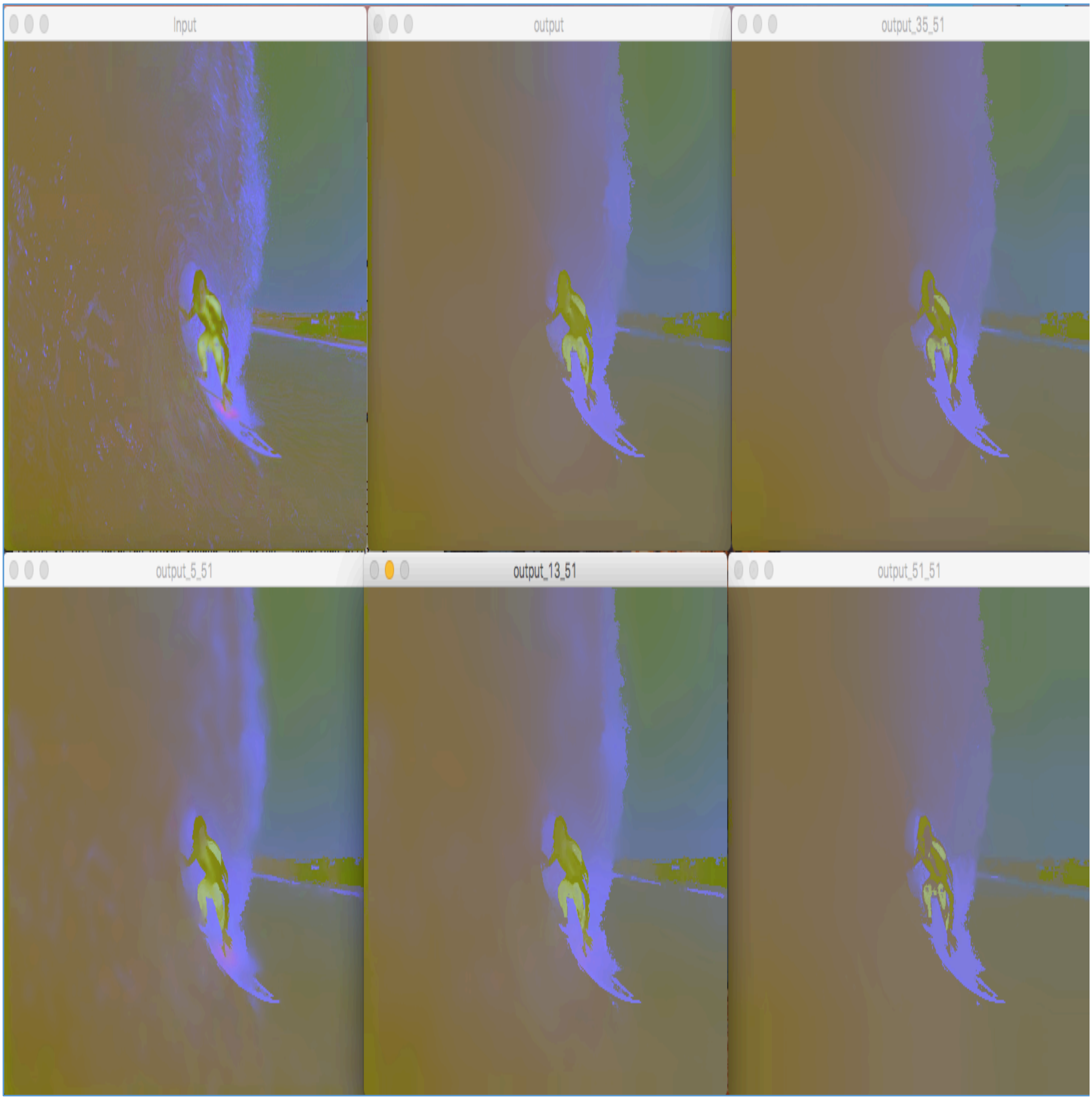
There is a strong dependence on the size of the window. Computational expense is large when larger iterations are needed based on the size of the window.

Lab color space is generally a standard color space since it is device independent and is the closest approximate of the human visual system. It also aids the storing of colors in limited precision values which in turn improve the reproduction of tones.

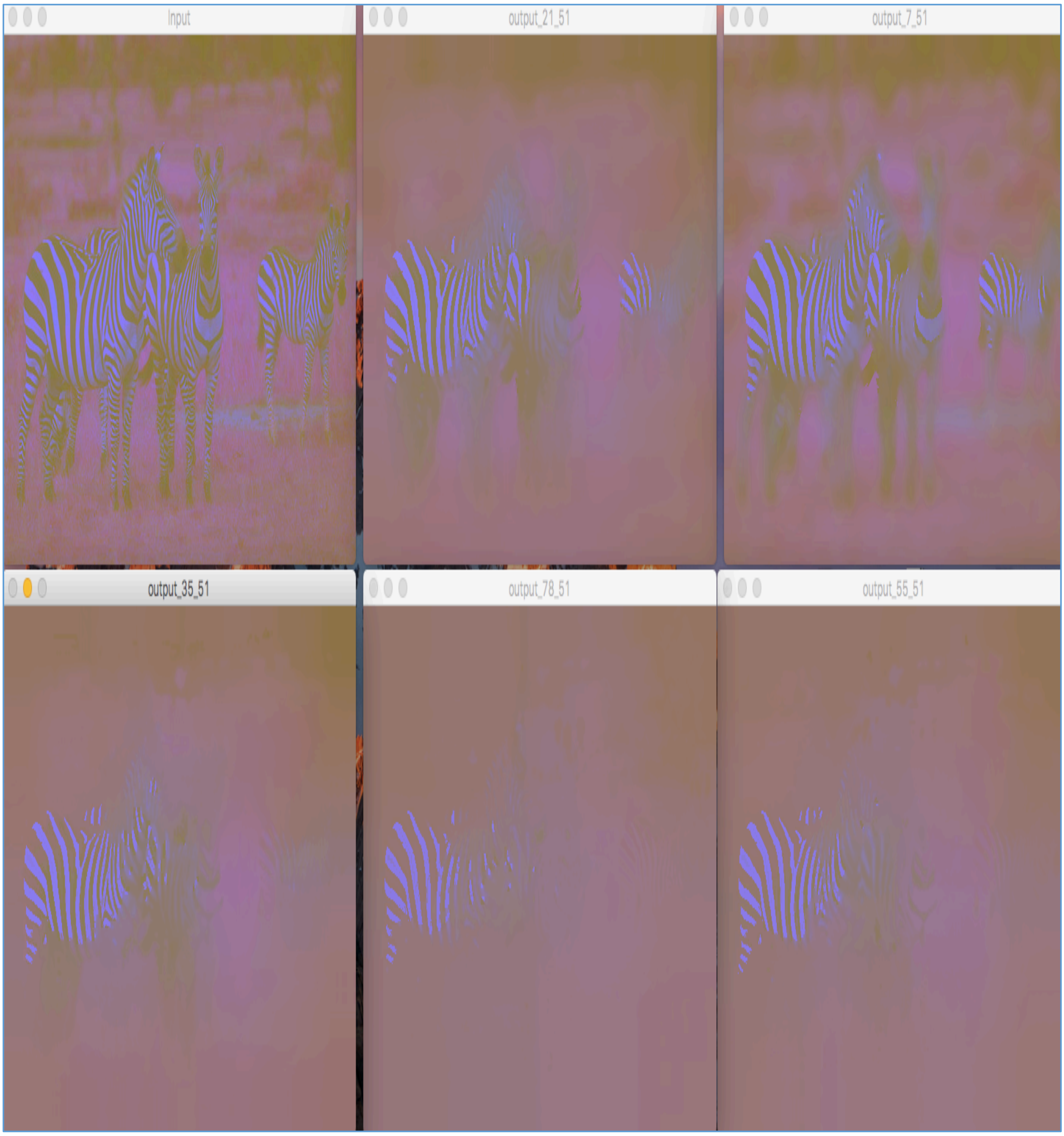
The spatial window radius affects the size of the ROI and hence largely changes the results of segmentation. Color window radius determines which differences in the value of the signal are need to be considered negligible. Since the modes of the feature or the pixel space are fixed, traversing through different radius highly affect the results of the segmentation.

OBSERVED RESULTS: The observations found are based on keeping the color window radius fixed and changing the spatial window radius in the first case and then by changing the color window radius keeping the spatial window radius fixed in the next case.

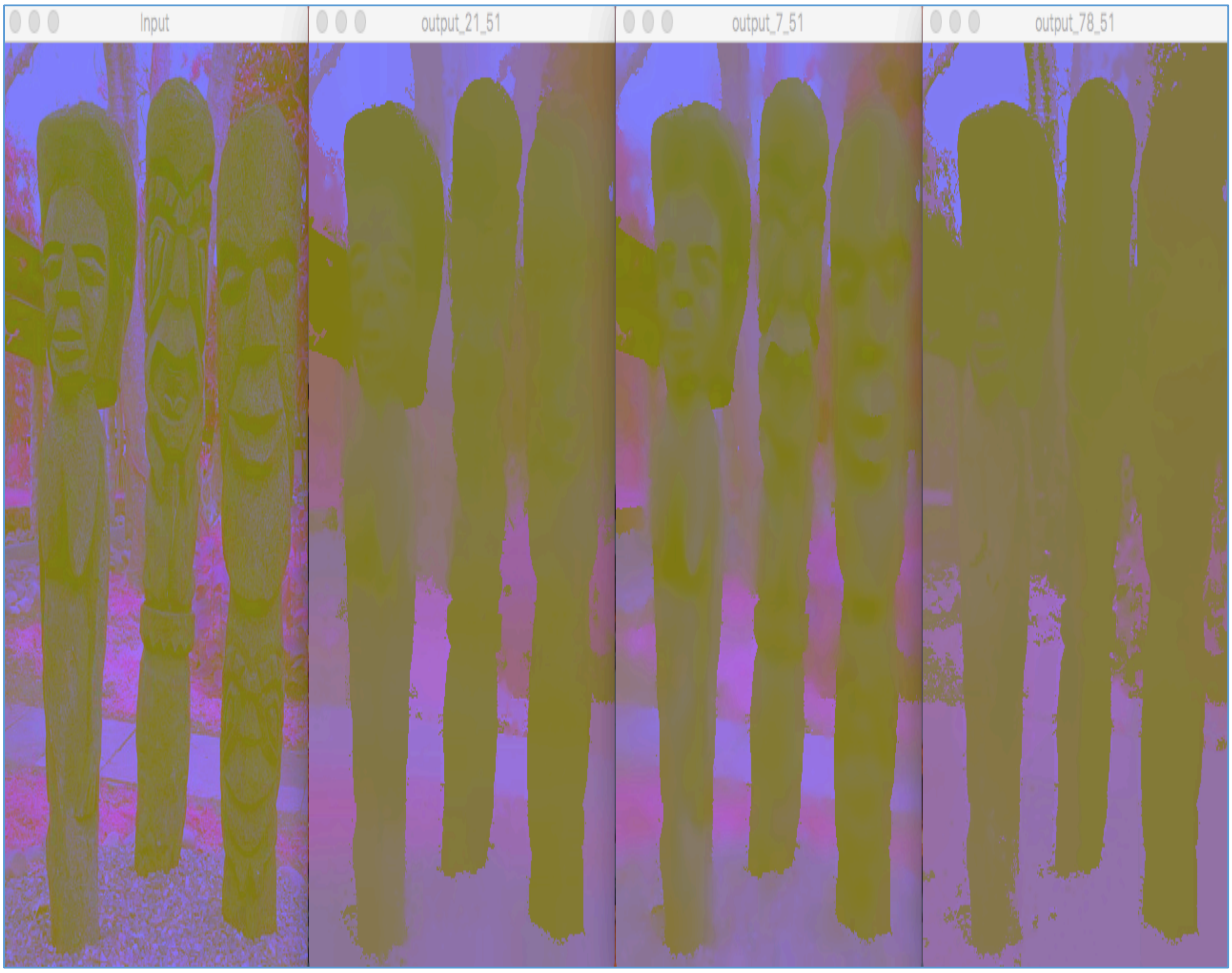
<u>CASE 1</u>: Spatial window radius different and Color window radius fixed at 51.



Results for the Surfer image with the top left specifying the input in the Lab space with color window radius fixed at 51.

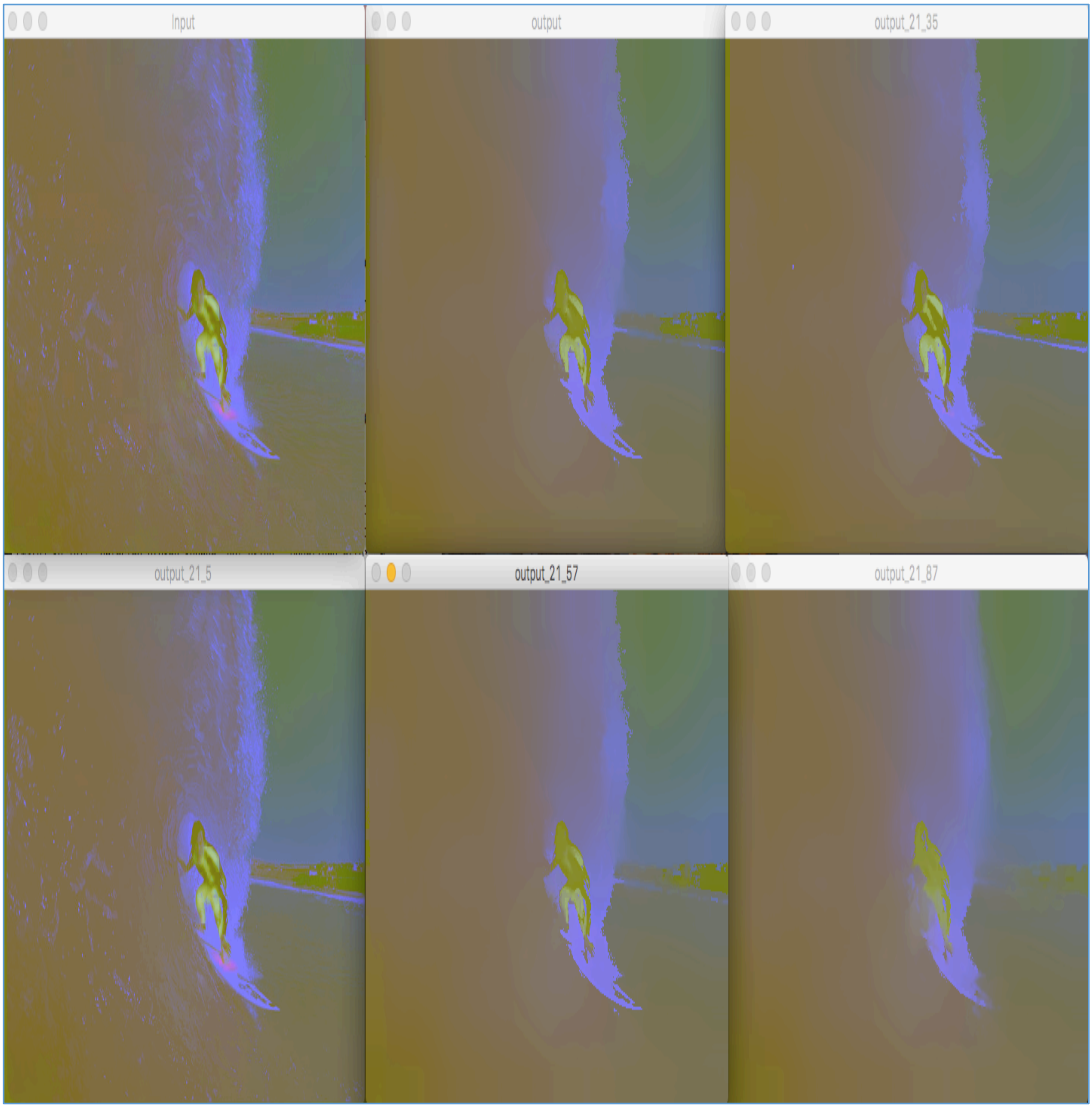


Results for the Zebra image with the top left specifying the input in the Lab space with color window radius fixed at 51.



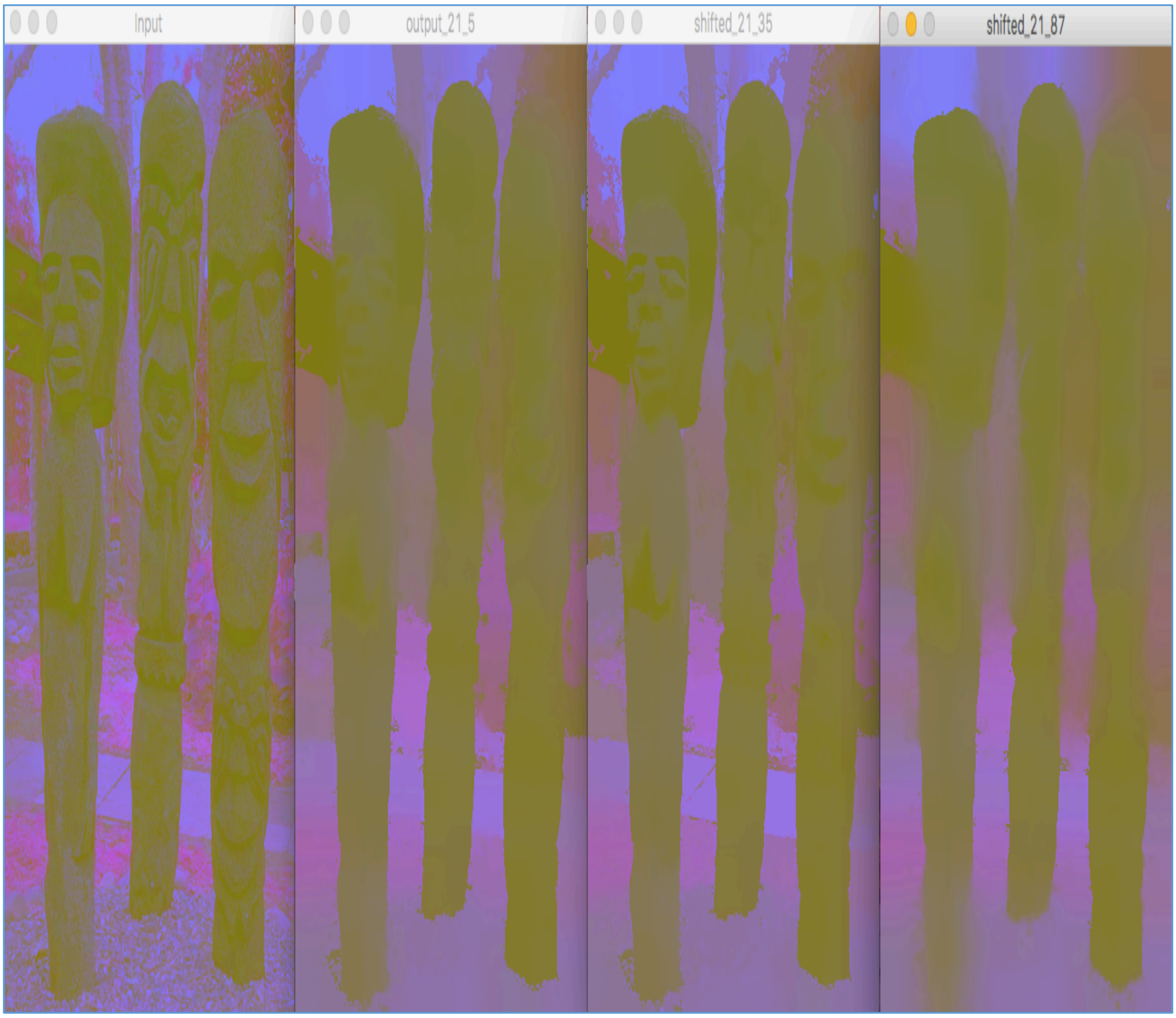
Results for the 3 statues image with the top left specifying the input in the Lab space with color window radius fixed at 51.

CASE 2: Spatial window radius fixed at 21 and Color window radius varying.



Results for the Surfer image with the top left specifying the input in the Lab space with spatial window radius fixed at 21.

Results for the Zebra image with the top left specifying the input in the Lab space with spatial window radius fixed at 21.



Results for the 3 statues image with the top left specifying the input in the Lab space with spatial window radius fixed at 21.

ANALYSIS OF THE OBSERVATIONS

It is evident from case 1 where the spatial window radius is variable and the color window radius is fixed at 51 that:

As the spatial window radius increases from 5 to 13 to 35 through 87, there is clearly evident lesser segmentation and lower number of clusters that have been formed.

In case of the Skate board image as the size of the spatial window increases the segmentation of the grass and the skate board is entirely lost. For the Zebra image as the size is increased the zebras in the background are merged with the background with only 1 visible among the 3 present . The 3 statues image also is indicative of the loss in details along with lower cluster formation.

If the spatial window size hence if kept too large leads to classification only based on the strongest mode in the image. This in turn leads to loss of the other attraction basins as the movement of the mean depends on the centre of mass inside the window which leads to missed attraction basins and lower number of modes.
If the spatial window radius is too small, there is insufficient separation between the clusters and bleed through effects are visible.
In all the 3 images with spatial window radii 3,5 and 7 we can observe that there have been larger segments created as the size is low and large number of iterations are needed to converge. If the size is too small and accurate results are expected, then larger number of iterations are needed which is computationally expensive as well.
From case 2 where the spatial window radius is fixed at 21 and the color window radius is flexible we can state that:
In case of all images, the difference between the Lab spaces is calculated and if the color window is too large, modes are omitted and the if the window is too small variations in the signal will be neglected and hence the larger segments will be created.
By looking at all the cases, we can state that the sizes of the spatial window radius as well as the color window radius affect the segmentation. Intermediate values for both the spatial window radius and the color window radius provide the best results.

Part (b): Watershed Segmentor

The watershed segmentation algorithm is carried out in OpenCV using the function watershed. It just takes markers and the source image as the parameters to carry out the watershed algorithm. According to Prof, I implemented the watershed algorithm using the second approach where I created a marker array of image size with all zero values. Then set marker pixels at some fixed spacing, say every 16 pixels apart in both rows and columns and made sure that these markers all had a different integer value, such as 1,2,3.... . Then simply applied the Watershed function with image and this marker. These results are like superpixel segmentation. Number of regions will depend on the spacing of the pixels; this is the only parameter can be changed.

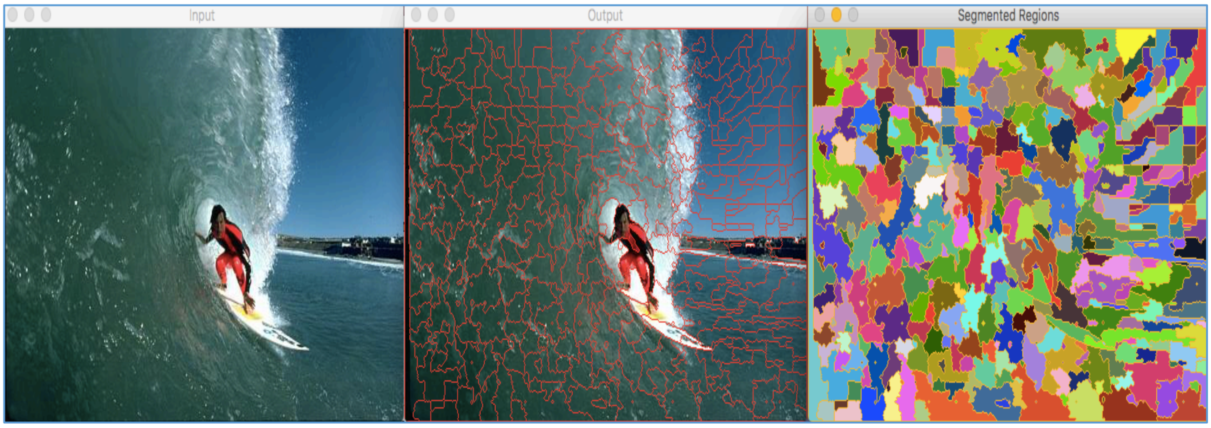
METHOD: The placement of the markers with regards to the spacing in the x and y direction are the parameters that can be changed and the total number of markers decide the segmentation. These marker pixels hence act like peaks and water is filled across the valleys. The boundary between these peaks is the segmentation that we are looking for.

The markers are labelled with positive integer indices and the background is set to black (0) initially. After the watershed algorithm is applied using the markers the pixels inside a segment are replaced with the corresponding index of the marker and the boundaries are set to -1.

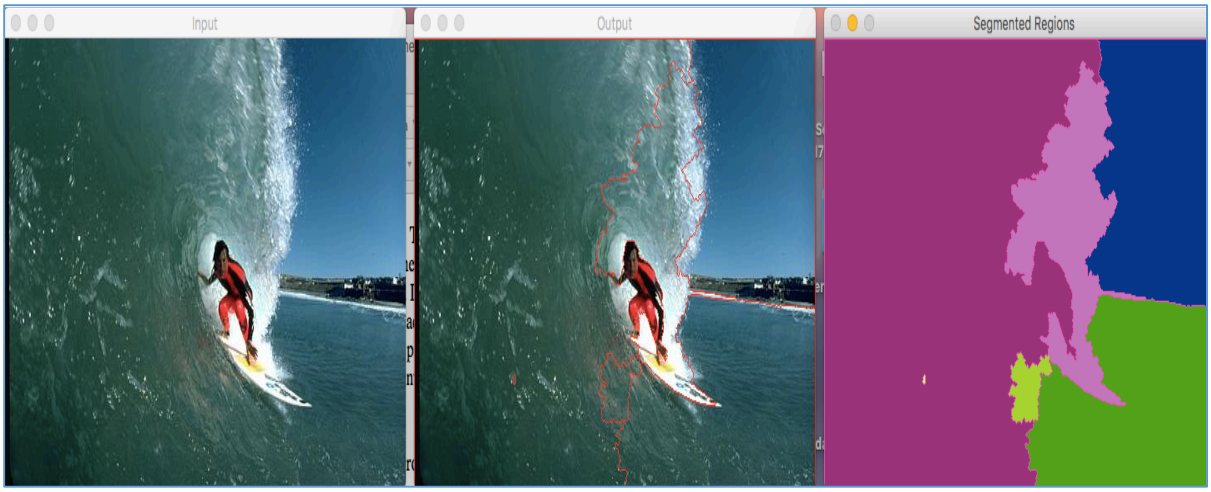
EFFECT OF PARAMETERS: The only parameters while carrying out such a process are the placement control gaps between the markers which can be dealt with. If larger number of markers are present due to relatively lower placement gaps, large amount of segmentation is caused. If larger placement gaps are produced, then there are lower number of segments and hence control completely depends on the markers. If markers are not utilised then over segmentation largely is seen and hence there is a need for the right selection of markers.

OBSERVED RESULTS: The process of setting up markers at uniform distances is carried out by keeping the placement gaps small (16) in both the axes followed by very large gaps (120) each which produce lower segments. Intermediate placement gaps of 50 and 80 in the x and y directions respectively are also shown for each of the images.

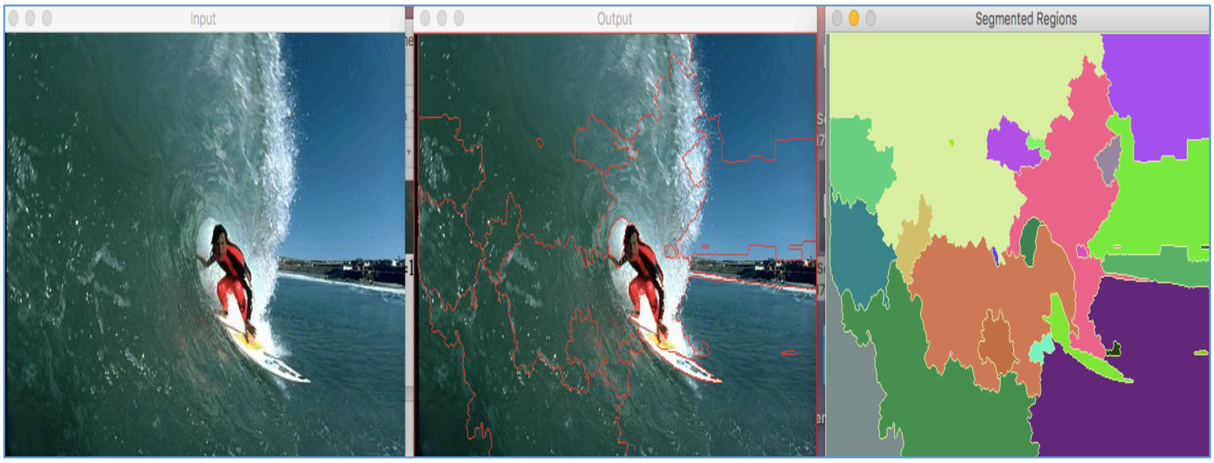
IMAGE 1: Skate board image



Placement gaps=16 in both directions

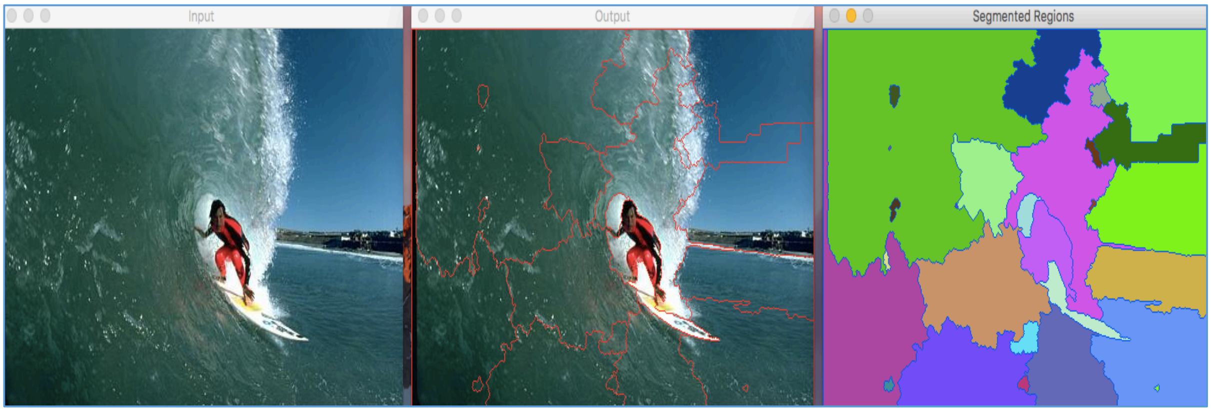


Placement gaps=120 in both directions



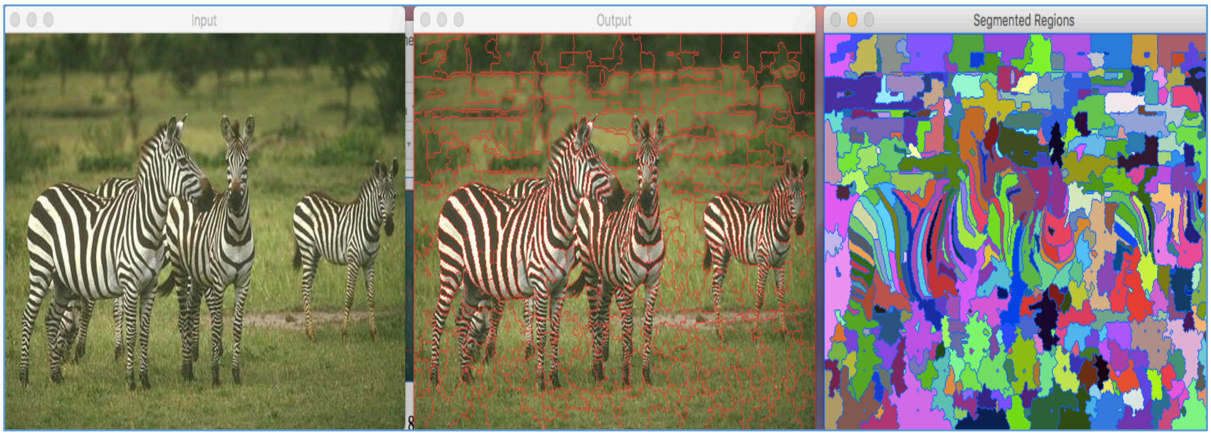
Placement gaps of 50 in x and 80 in y direction

.

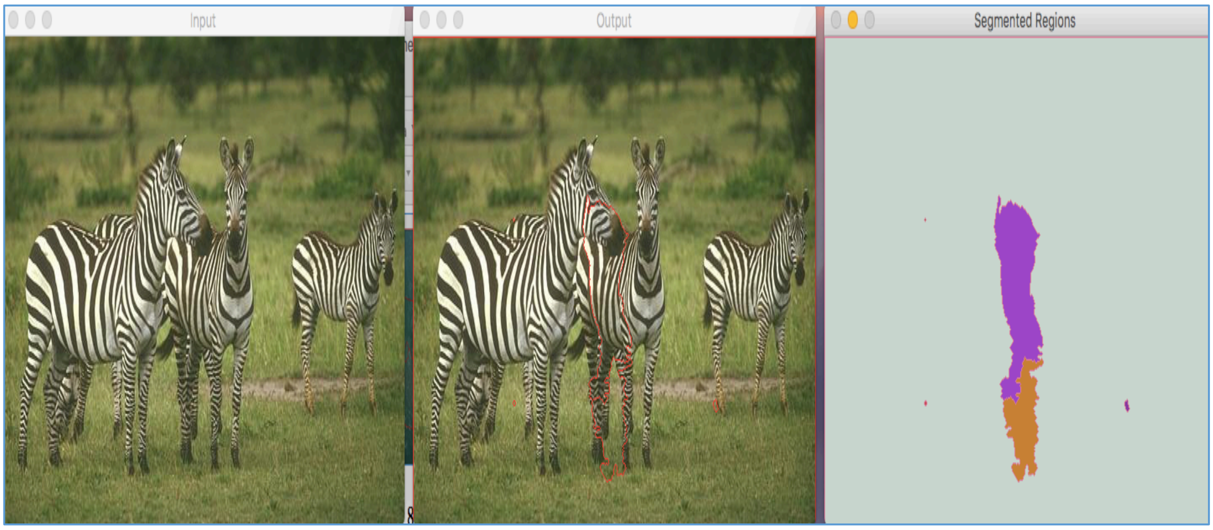


Placement gaps of 80 in x and 50 in y direction

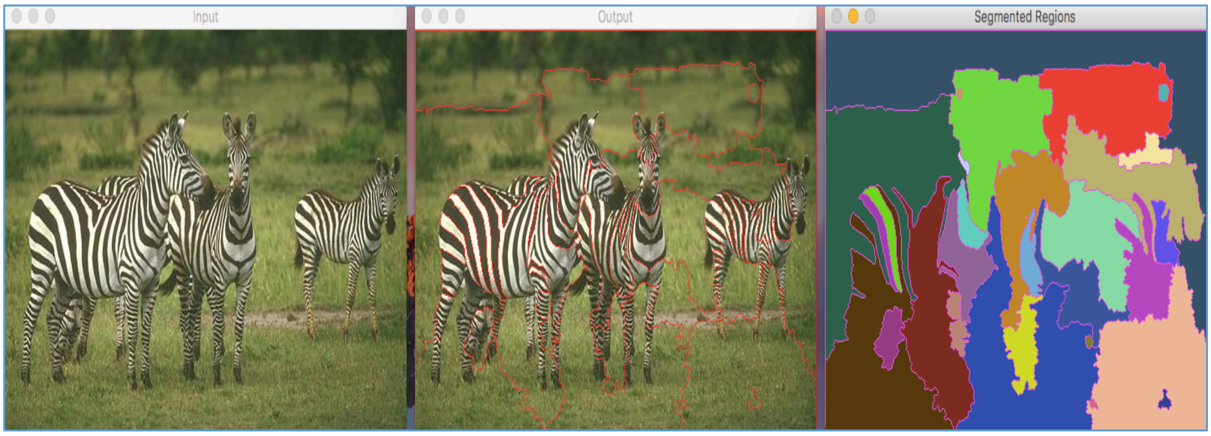
IMAGE 2: Zebra image



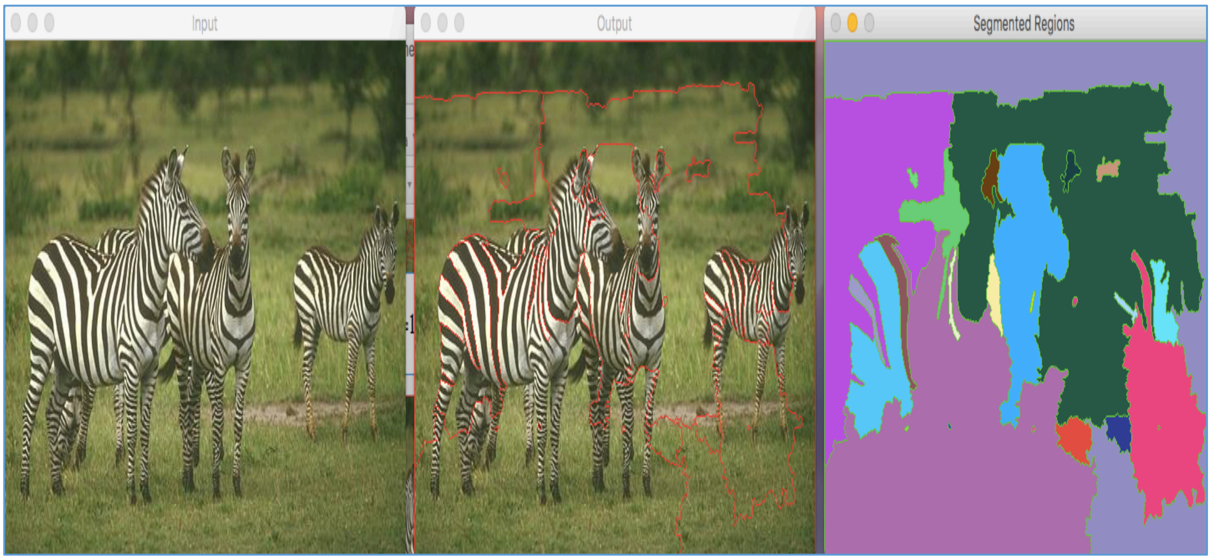
Placement gaps=16 in both directions



Placement gaps=120 in both directions



Placement gaps of 50 in x and 80 in y direction



Placement gaps of 80 in x and 50 in y direction

IMAGE 3: 3 statues image.

Placement gaps=120 in both directions

Placement gaps of 50 in x and 80 in y direction

Placement gaps=16 in both directions

Placement gaps of 80 in x and 50 in y direction

ANALYSIS OF THE OBSERVATIONS

From all the above images we can notice that as the as the placement gaps increase the segments decrease and vice versa. The total number of marker segments in case of 120 placement gap is 12 and in case of 16 each is 600 segments markers. It is 40 in the case when 50 placement gap is x direction and 80 in the y and is 42 in the vice versa case. It becomes clear that with the change even in the orientation of the placement of the markers the segmented results change.

In order to produce desired results using the markers technique, we can implement the selections of the number of markers on MOUSE click or by manual selection by keyboard. This will in turn produce better segmented results but needs human intervention of labelling whereas the former can be set automatically.

CONCLUSION

Both the methods of segmentation were explained in brief with the evidences of the results shown. Both the segmentation algorithms do not provide semantic labelling and hence cannot accurately segment the regions as expected from the ground truths. The effects of each of the parameters is also mentioned in the discussion above. Both the algorithms are completely different as one uses a probability density function to move the vector in the mean shift segmentor whereas the other depends on the intensity peaks and valleys to decide the catchment basin. Advantages of both can be utilised in specific targets with mean shift being used for tracking robustly in videos with initial frame marking to watershed being a pre-processing step that helps create super pixel like segments that reduce the total feature space and help deal with lower number of pixels or features for the next processing steps.