Software for the Data Analysis of Super Resolution Microscopy Images

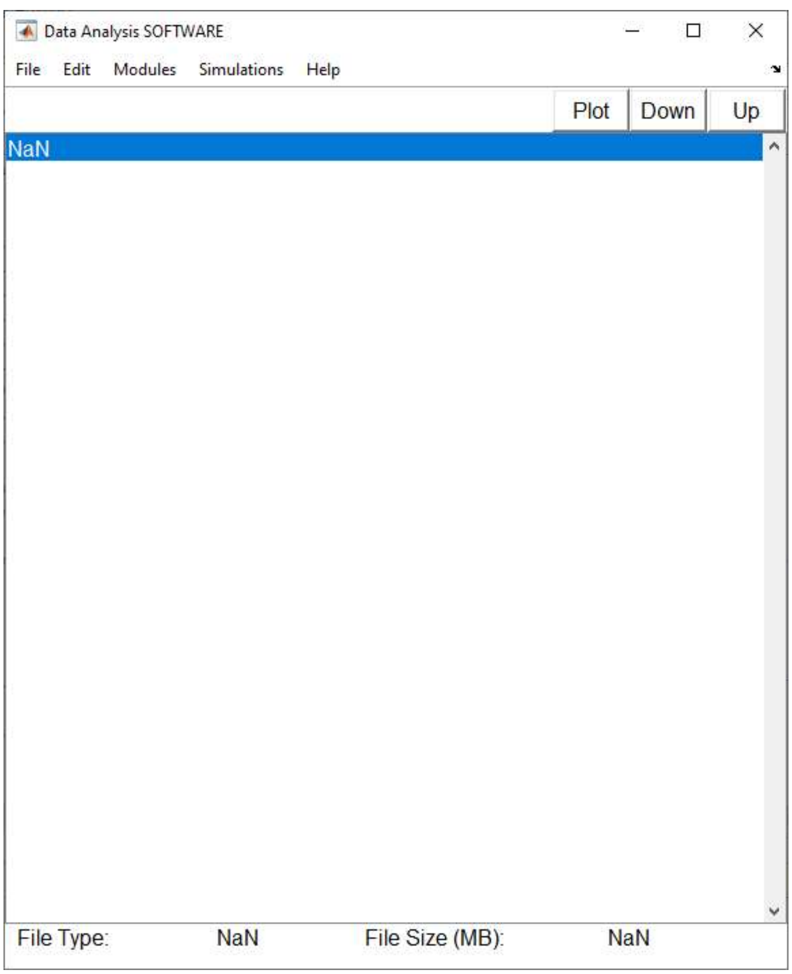
Arian Arab, Melike Lakadamyali
University of Pennsylvania
Department of Physiology

Software can handle different files types:

- image
- loc_list; point patterns, single-molecule locations, localizations.
- spt; single particle tracks

Batch Processing: software can load several files at the same time and perform the same analysis on all the datasets.

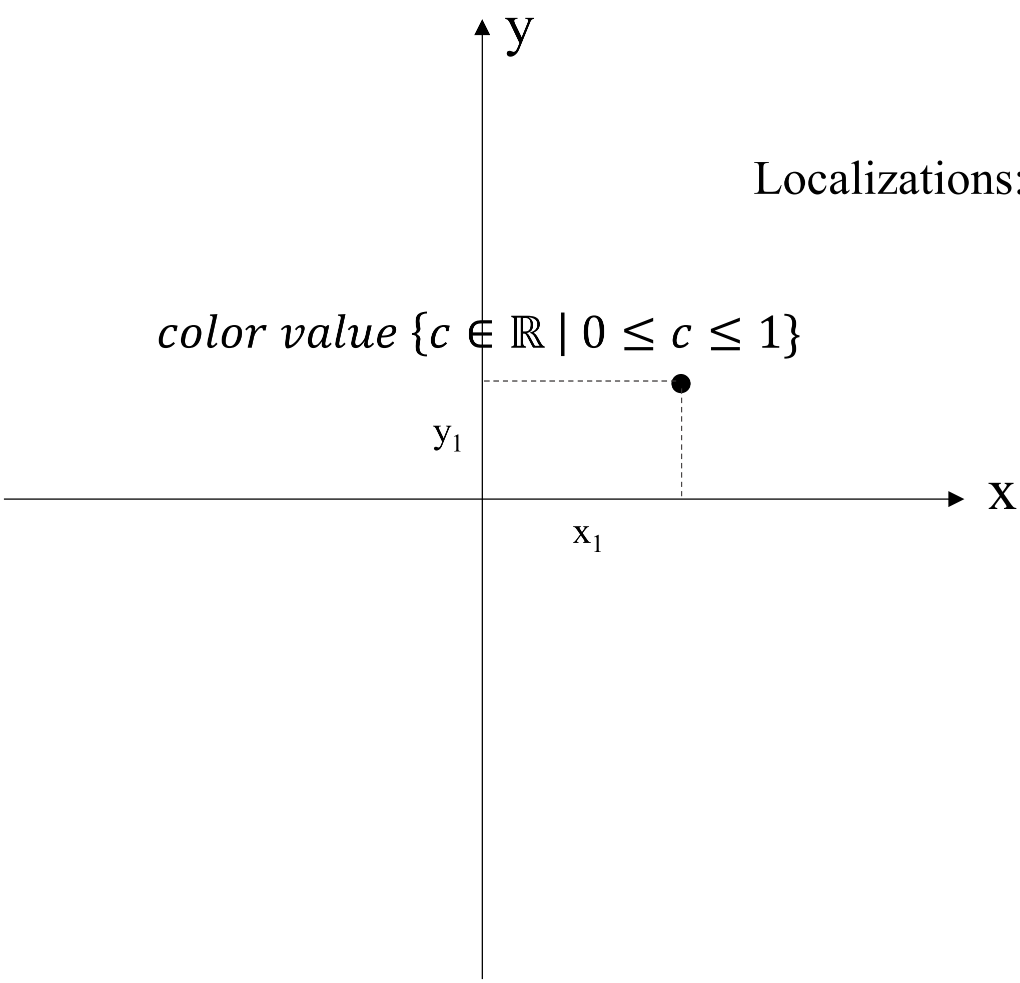
After each analysis, the work-session can be saved and load for future reference (loading and saving sessions).



STORM Microscopy localization data

STORM data contains a localization list which is a set of points in the two-dimensional (x-y) plane obtained from single-molecule localization of fluorophore molecules blinking during the image acquisition process.

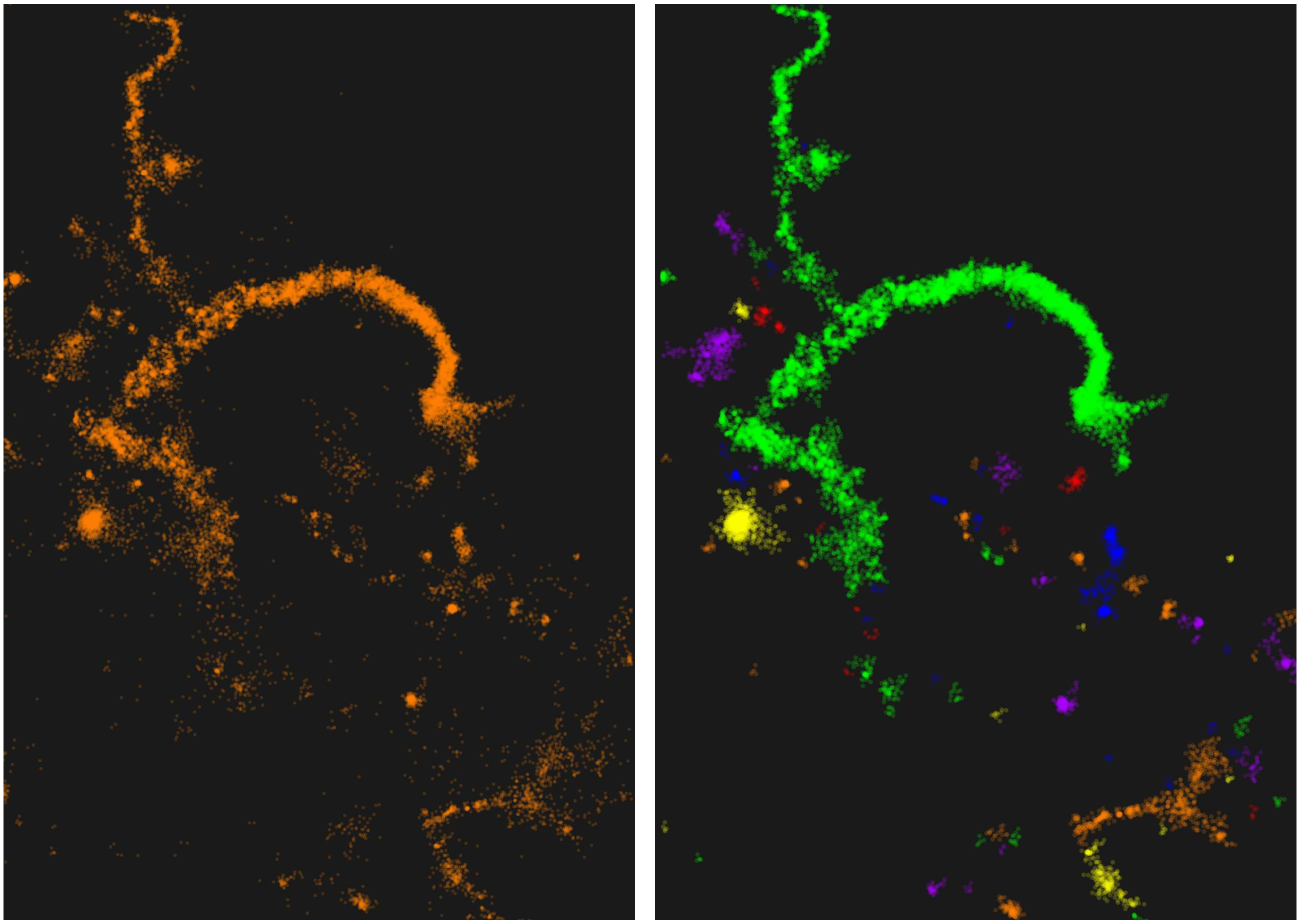
Here, for each localization we attribute a color value. This color-value will be helpful for image-segmentation. Raw input images will contain one color value for all the localizations. However, segmented-images will have different color values as required for different segments of the segmented-image. Here, each segment will be called a cluster (a cluster of points having the same color).



Localizations:

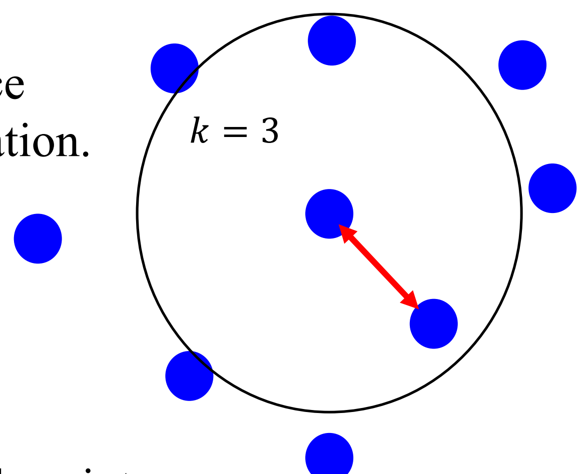
x 1	y 2
-80.9842	667.8633
-80.8757	670.5156
-80.6116	667.9143
-79.9542	671.9676
-79.7301	651.9576
-79.6991	671.0201
-79.6430	673.7319
-79.5677	673.7574
-79.5633	667.8085
-79.5345	668.7341
-79.5080	668.6187
-79.4754	672.4706
-79.3983	652.6793
-79.3401	672.6433
-79.3275	668.4178
-79.3238	671.3583
-79.3219	652.8129
-79.2808	672.1857
-79.2748	669.9576
-79.2637	668.9144
-79.2528	668.8826
-79.2417	671.1658
-79.2283	676.1459
-79.2217	668.6361
-79.1721	669.0104

Here, raw input data is shown with a single color (orange) while the segmented data is shown with different colors for each segment:



K-nearest neighbors in STORM

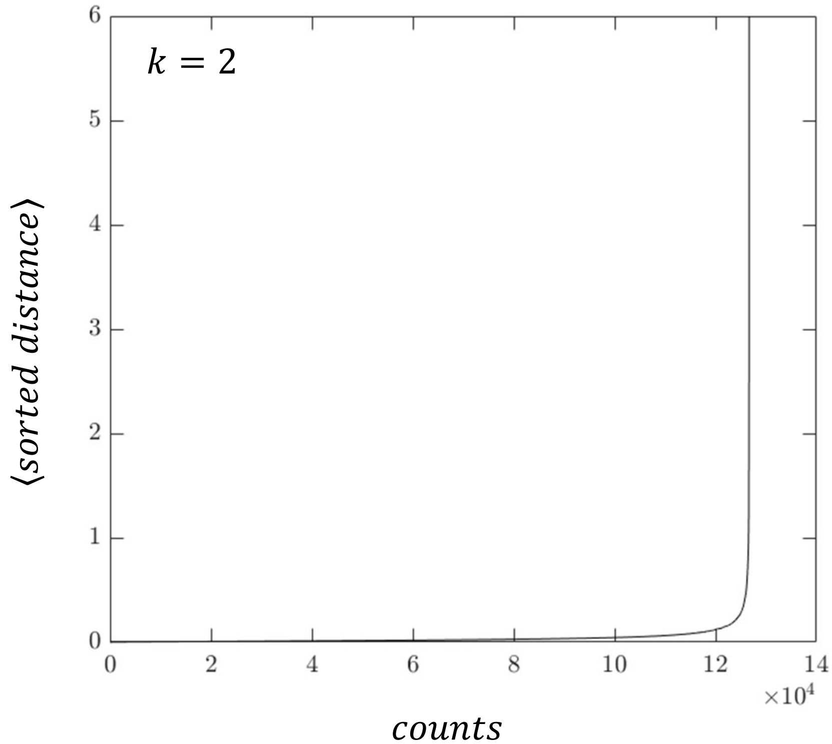
K-nearest neighbor is the average distance of the k-1 nearest points for each localization.



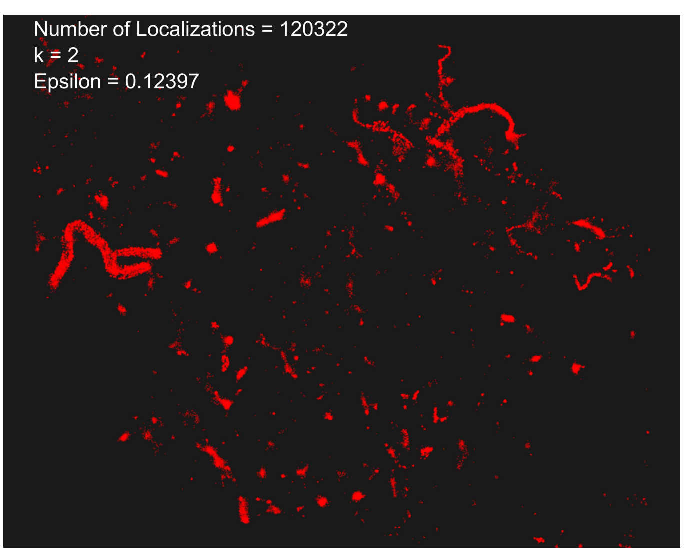
For example, in the figure shown here, for k=3, two closest points are shown.

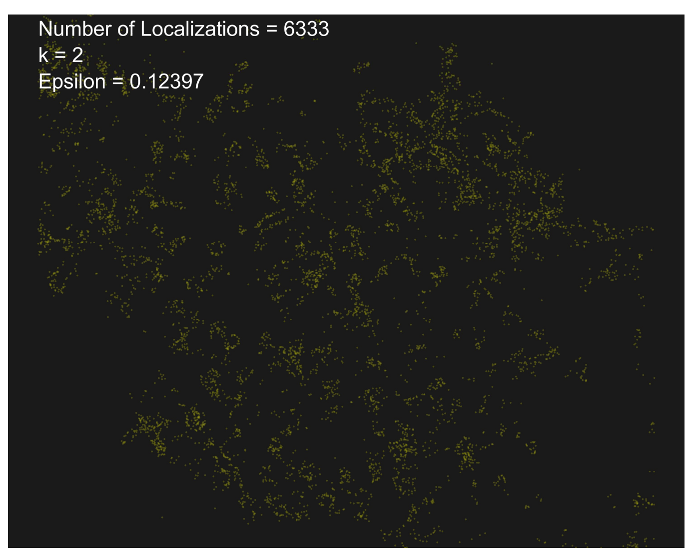
We can plot the average distances of the k-points for each of the localizations and obtain a sorted plot of the average distances.

k-nearest-neighbor
(including the search point)

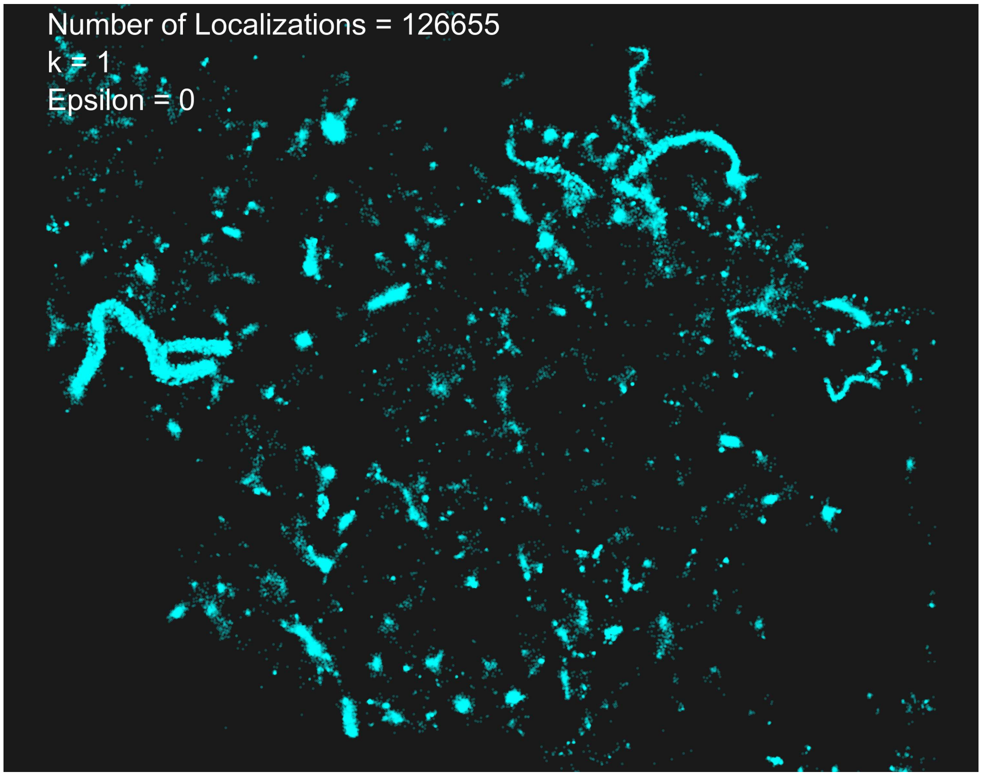


We can then find the 95th percentile value of the distance plot shown above and then remove the localizations which are having an average k-distance value corresponding to 95th percentile or higher. This way, we can remove the localizations which are located further away from each other as compared to other localizations.





We can also use this k-average distance information to create a density map of the STORM image:



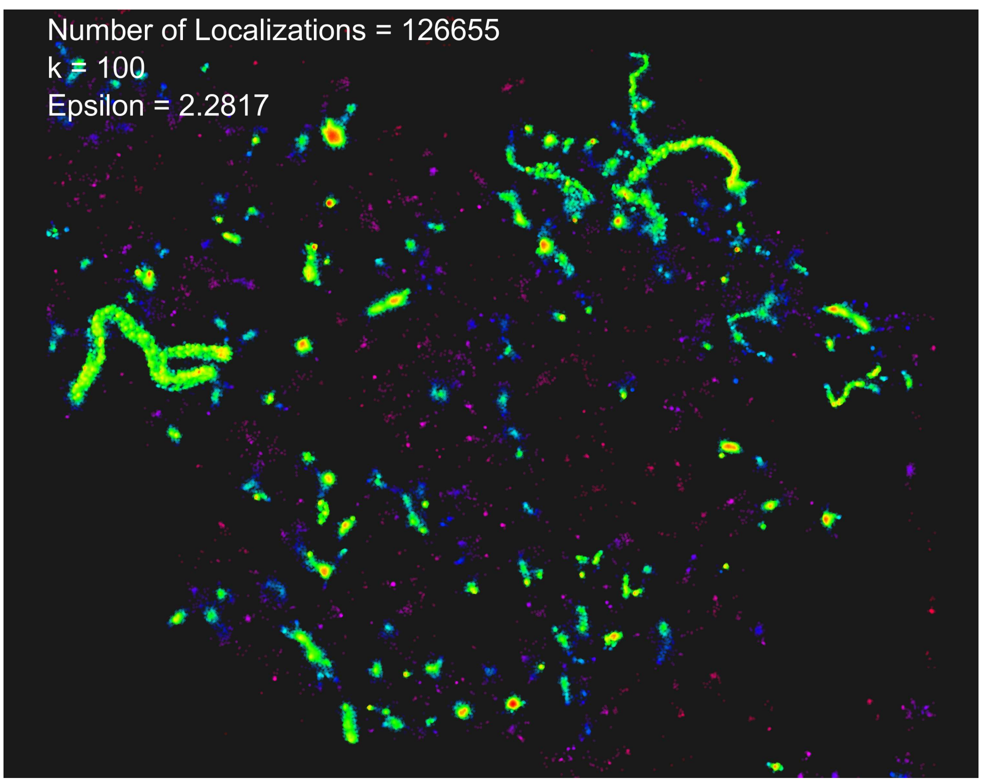
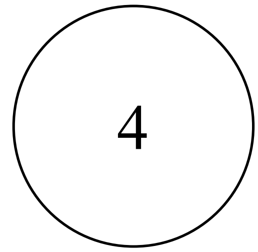


Image Segmentation using DBSCAN

DBSCAN:



Input: Minimum Number of Points
Search Radius (Epsilon)

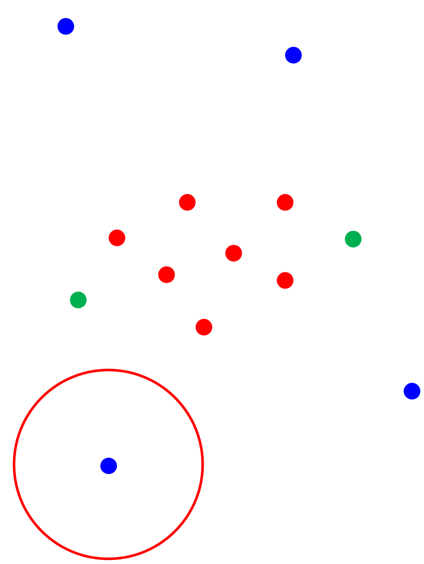
Output: Core Points
Boundary Points
Noise Points

For each of the localizations we do the following:

1) If the number of localizations falling in the circle with radius of epsilon centered around each of the localizations is equal or greater than the minimum number of points set by the user then that point is a core point.

2) If the situation above is not met but there is at least one core in the circle surrounding the corresponding point, then that localization is a boundary point.

3) If the two conditions above are not met, the that point is a noise point.

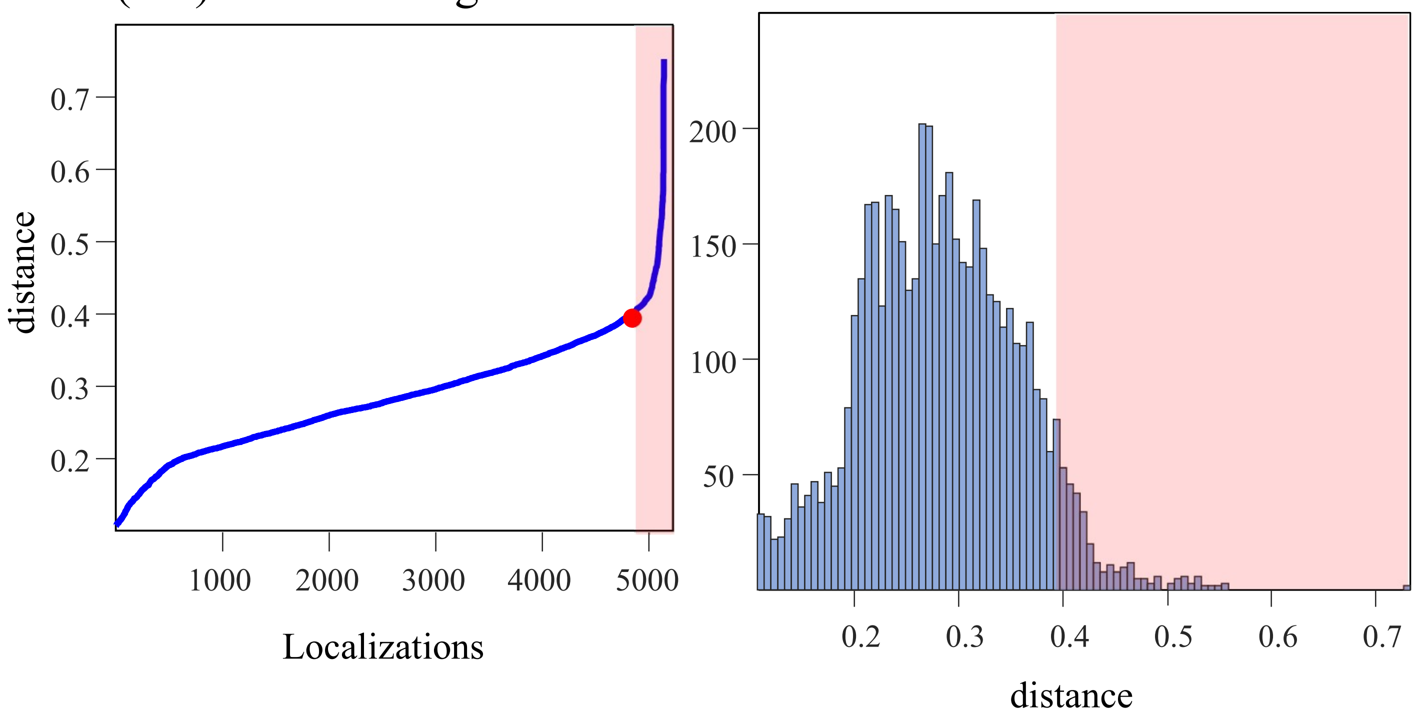


Elbow method to remove the epsilon value:

If the minimum number of points in DBSCAN is set to be k, we can calculate the average distances of the k-1 nearest points and then obtain the elbow point from the sorted average distance graph:

Calculate the (k-1)-nearest neighbor distances
(k = Minimum Number of Points)

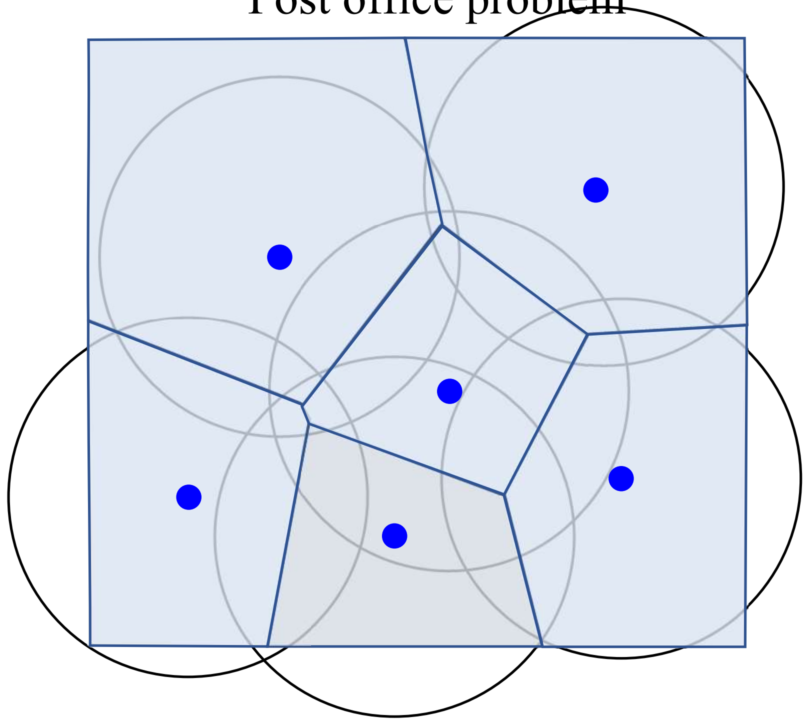
(k-1)th nearest neighbor distances



As a results, the epsilon value can be found from reading the value of the elbow point in the distance plot shown above (red circle, value of 0.4).

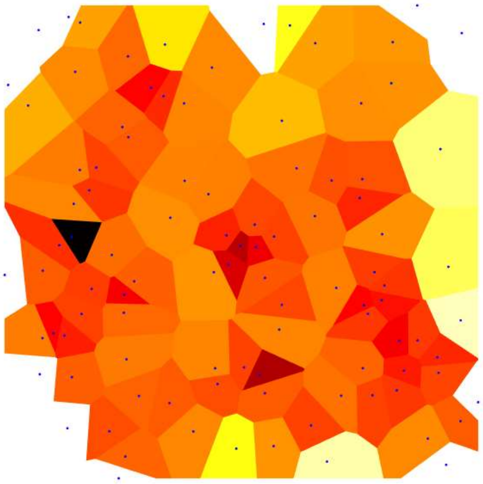
Image Segmentation using Voronoi Diagram

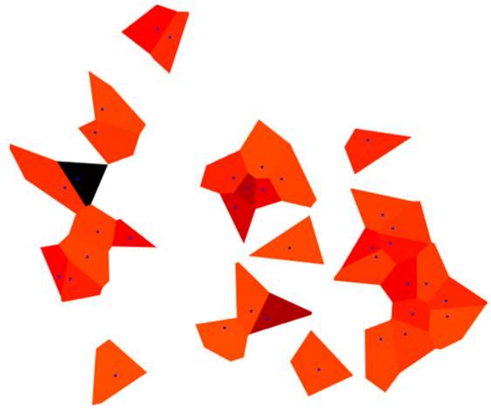
Post office problem

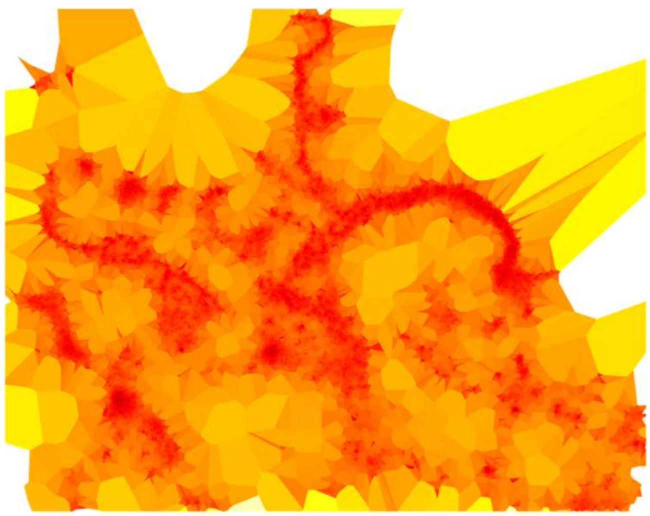


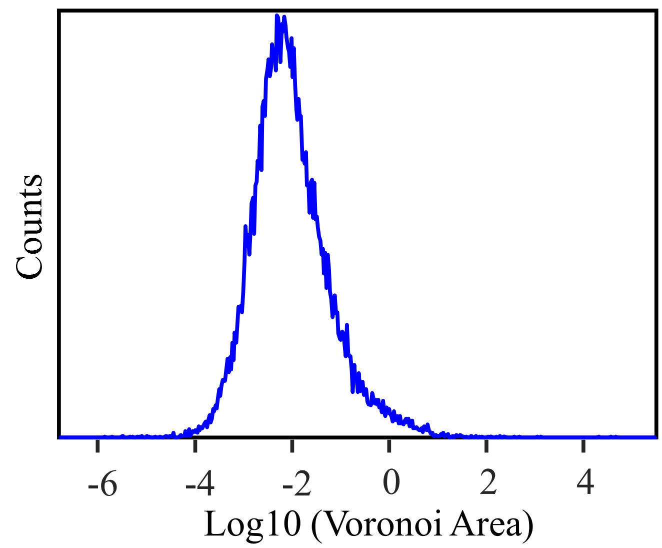
Voronoi diagrams can be found in nature all the time. For a Voronoi diagram, in each cell, no other point will be found closer to the point corresponding to that cell than to all the points in the diagram. There is a famous problem in business stated as post-office problem. The zip codes will be found for a cluster of post offices to deliver mail requiring that the postmen do not travel more than necessary.

For the localizations from STORM images, we can obtain these Voronoi cells and calculate each Voronoi cell area (polygon area). By putting a minimum threshold on the Voronoi cell areas, we can remove the localizations having an area larger than the threshold value and then only keep the ones below the Voronoi area threshold value. Furthermore, if two or more cells are connected, we will form a cluster of points. As a result, the image can be segmented into different and unique clusters.

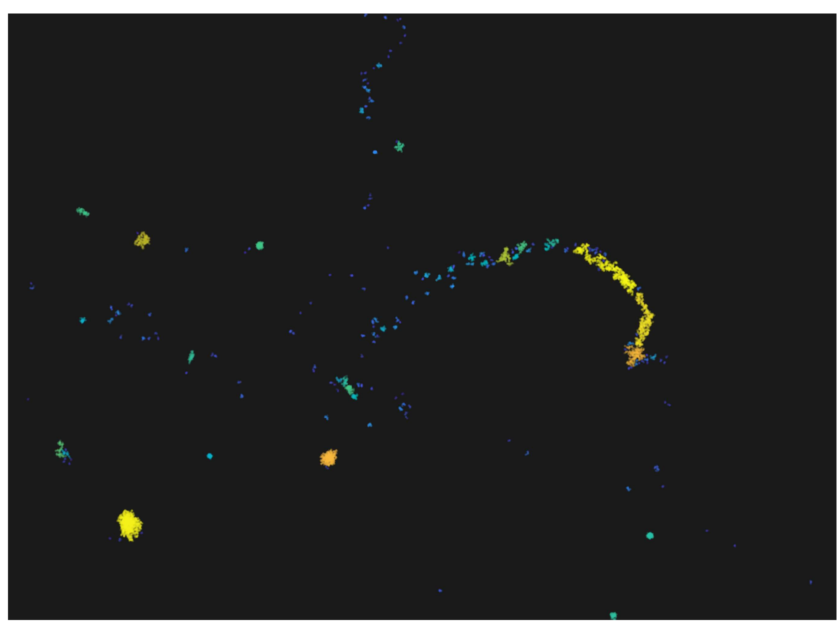




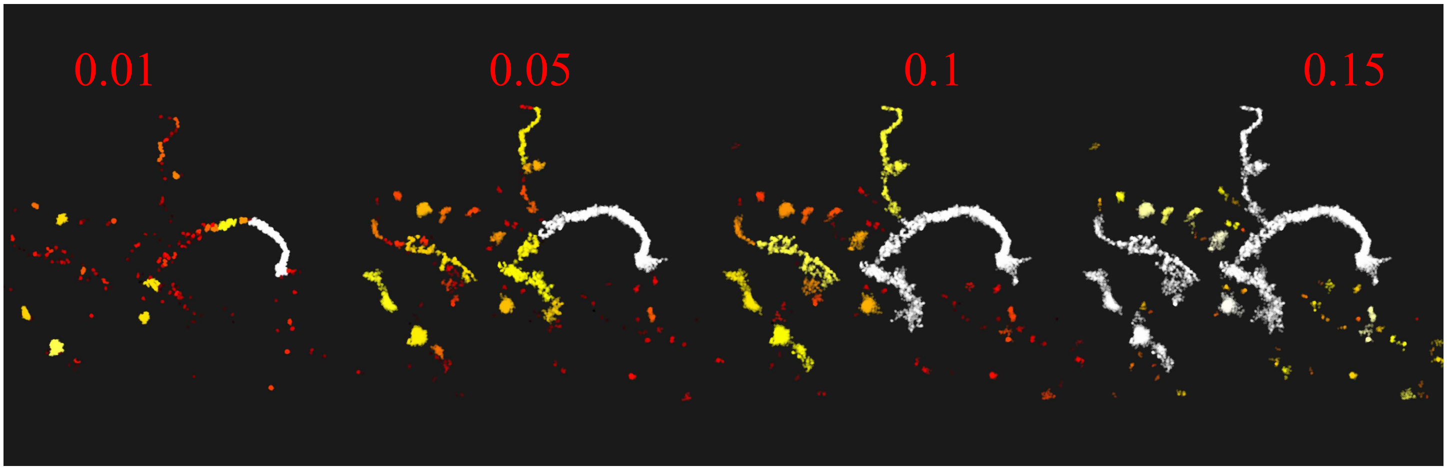




Here minimum threshold area is set to 40th percentile value of all the Voronoi cells areas:

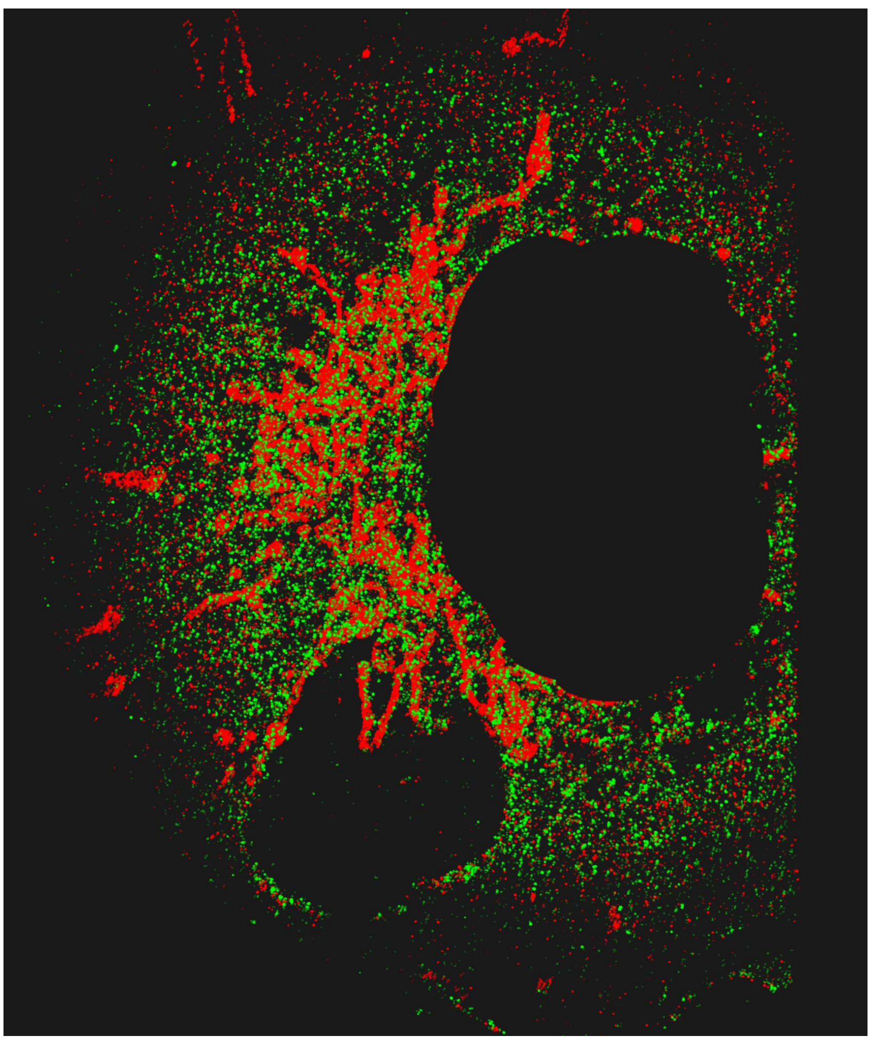


Here, results from using different threshold values are shown:



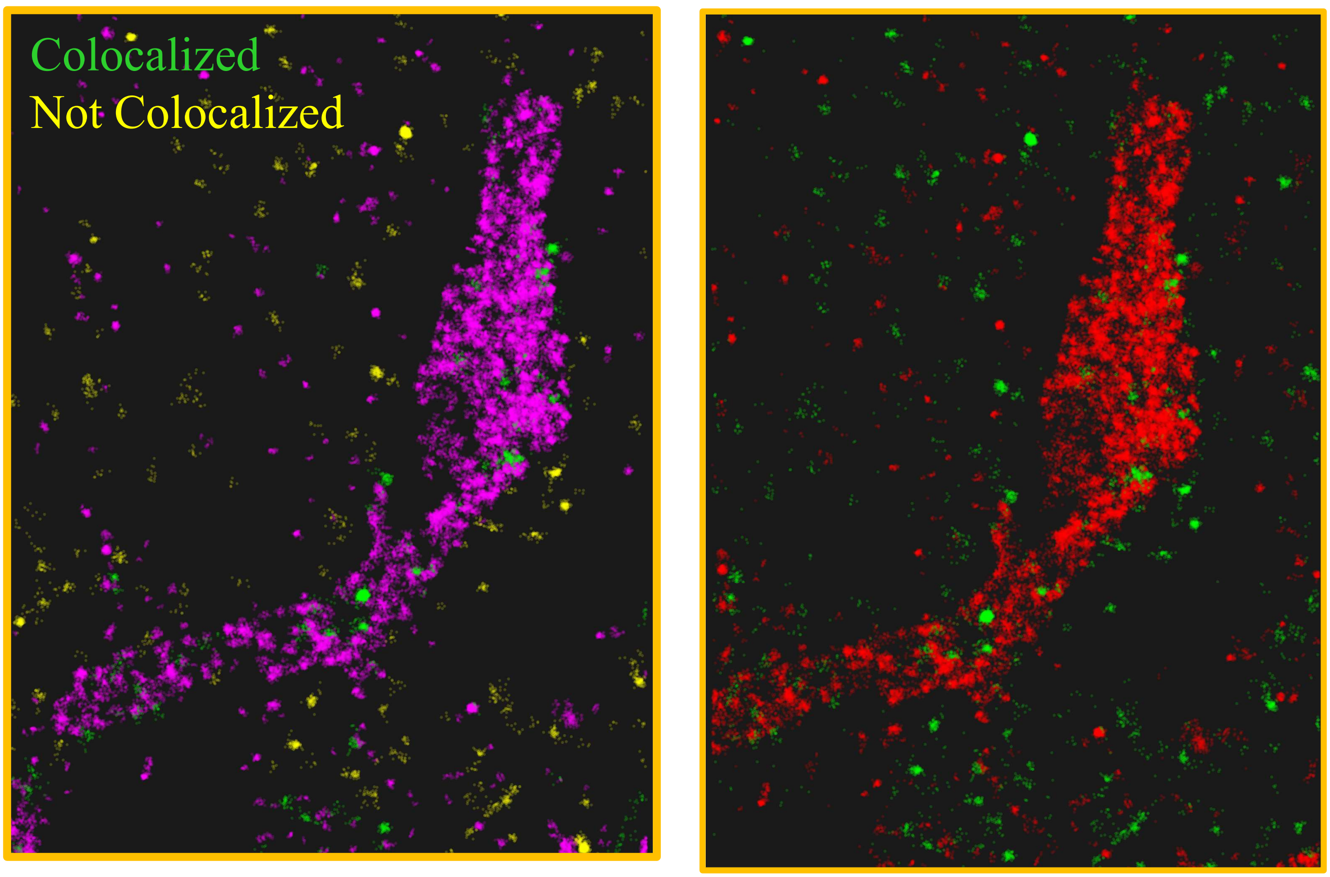
Colocalization Module

We can also perform colocalization analysis by having storm images from two different experiments (shown with green and red color):



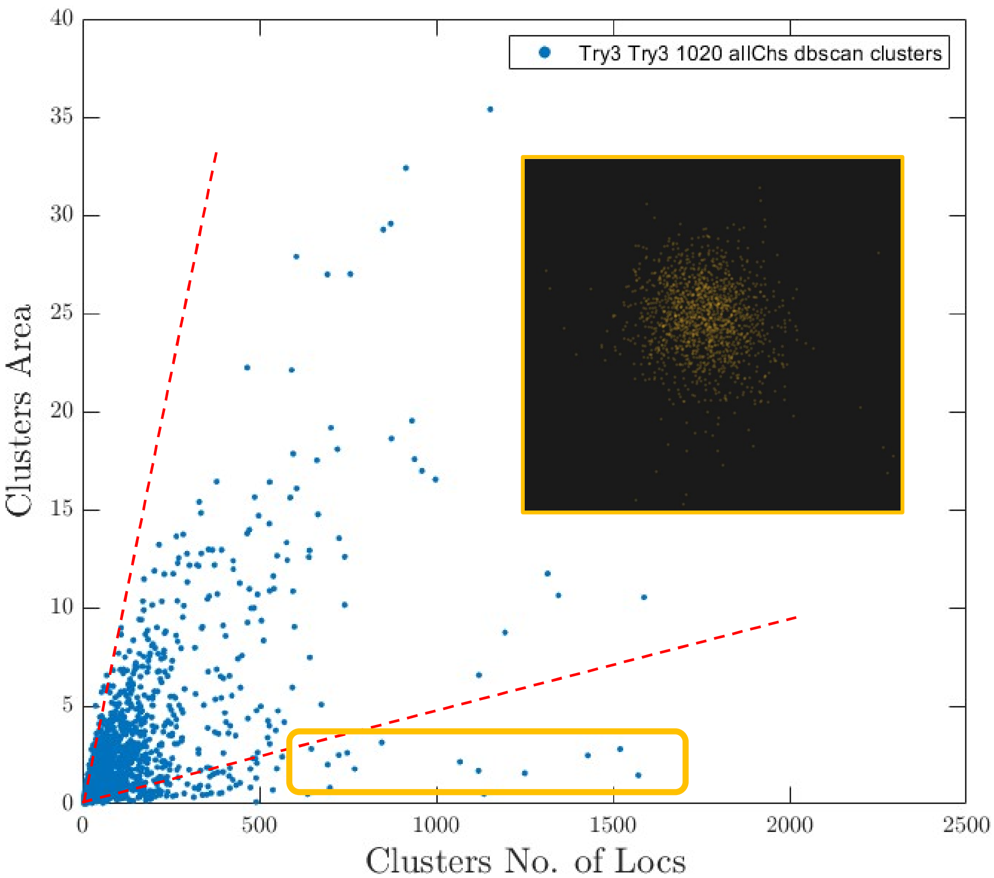
If we look at the figure below (right-hand side), some of the localizations shown with green-color overlap with the localizations shown with the red-color.

More specifically, we need to first segment the red-colored and green-colored STORM images separately and then find the overlap between clusters of localizations instead of individual localizations for better analysis and more accurate results:



Removing Outlier Clusters:

After image segmentation, each cluster will have a number of localizations and a specific value for its area. We can plot the scatter graph of the clusters number of localizations versus cluster's areas.



As can be seen from the figure above, some of these clusters are falling off the red dashed lines. These are the clusters having an unusual behavior. Either having a large number of localizations while having small area or having a low number of localizations and large area. In order to correct for these outliers, we can plot the histogram of the clusters number of localizations versus clusters areas. We can then fit this histogram graph to a Gaussian distribution and then use the three-sigma rule to keep the 99.7% of the data and remove the 0.3% of the data corresponding as outliers.