Kmeans Clustering Algo:

1. Get the centroid values from HBCE algo (CC[0], CC[1],….etc)
2. Define the number of clusters as number of centroids (list in my case)
3. Now for each pixels of image, check the pixel is nearest to which centroid
4. Put the pixel coordinates in that cluster
5. So, after the loop is exhausted, all the pixel coordinates will be in their nearest centroid(i.e.) clustered together
6. Now, calculate the new means for all the clusters, and update the CC values appropriately
7. Repeat the steps 3-6 until epoch get exhausted

Otsu Thresholding:

1. Find total mean from histogram of image
2. Assume each intensity as the threshold and For each intensity calculate between class variance(between class variance should be more and within class variance should be less for good segmentation)
3. So, finally we will be get the intensity value for which between class variance is max (Threshold)
4. With Threshold intensity, segment the image between foreground and background

Graph cut alpha beta swap:

1. V(a,b) = 0 iff a = b
2. V(a,b) = V(b,a) >=0
3. V(a,b)<=V(a,g)+V(g,b)

If 1,2,3 then V is metric

If 1 and 2 then V is semimetric