Graph Cut:

1. It is an algorithm/method employed for image segmentation.

2. It is based on the concept of normal graph data structure with nodes and edges.

3. The image pixels are represented as nodes. There are 2 extra nodes representing the background node and foreground node.

4. The aim is to segment the image pixels into background and foreground pixels.

5. It is based on the colour histogram-based inference of the properties of the background and foreground nodes.

6. It uses the probabilistic approach using the colour histogram, to assign each pixel to background and foreground.

7. Depending on the intensity of the pixel the probabilistic value of it being in the foreground or background is assigned to the pixels.

8. The probability assigns weight the pixels using the formula h_f(l_p)/max(h_f).

9. A pixel with more probability in foreground has darker edges with the foreground node and lighter edge with background node and vice versa. These edges define the edges of the graph.

10. It removes the artefact caused by saliency as there are also edges between the pixels based on its similarity with its neighbours.

11. The more the similarity with the 8 neighbours the darker the edges of the pixels with its 8 neighbours else the edges are lighter.

12. So basically we have three kinds of edges of each pixel. One with foreground, other with background and last with its neighbours.

13.These probabilities define the weights of the edges.

14. These weight are then used to get the min cut such that the edges used to divide the pixels into background or foreground label has minimum sum of weights probabilities.

15.It is majorly used in case of monochromatic images.

Grab Cut:

1. This method/algorithm is also used for image segmentation.

2. This is based on the graph cut technique only but a specialised version of Graph Cut.

3. Unlike the graph cut, it is majorly employed on colour images.

4. Instead of histogram inference, it makes use of k-means clustering-based inference for the background and foreground nodes.

5. Each cluster has its own mean or the centroid, the weight(importance based on the maximum number of labelled nodes) and covariance or the spread.

6. It is based on the assumption that it employs a Gaussian mixture model.

7. Unlike the graph cut, it does not use a probabilistic approach.

8. The building of the entire graph is same as the Graph Cut algorithm. The difference come with how the weights are assigned to the edges of the graph.

9. It assigns an edge weight based on the summation of 3 factors.

10.The first factor, the distance of the pixel from each of the cluster centroids of the foreground and background label.

11. The lesser the distance the more the chance of pixel to be in that cluster and hence more thicker edge to the cluster of that kind(foreground or background).

12. The second factor, the weight of each cluster. The more the weight, the more the chances to attract the pixels into that cluster and hence stronger or thicker the edges of the pixel with that kind of cluster.

13. The third factor, the more the covariance of the cluster, the more the chance of pixel to attract that cluster and hence stronger or thicker the edges of the pixel with that kind of cluster.

14. It employs the same method of similarity pixel as the graph cut and uses min-cut to separate to foreground and background same as the Graph Cut algorithm.