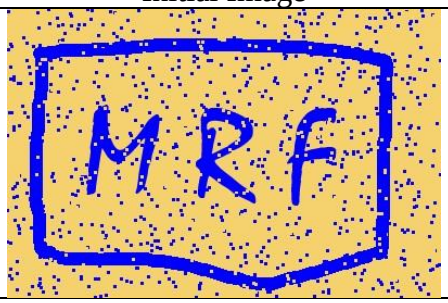
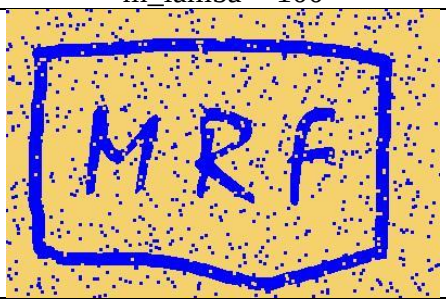
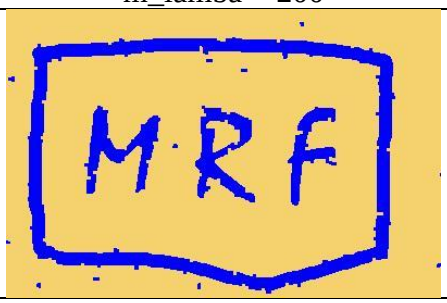
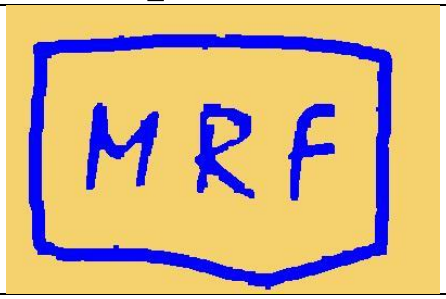
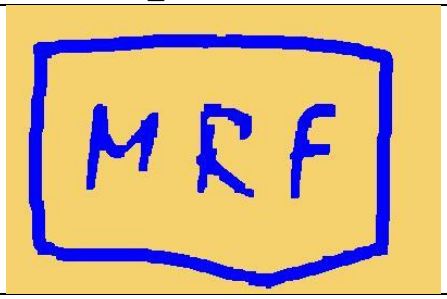
EE4212 – Part 2 Assignment 2 [Markov Random Field and Graphcuts]

Tay Kai Yang, A0136114U

Part 1. Noise cleaning

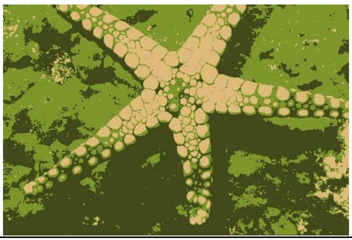
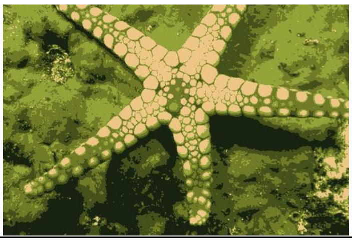
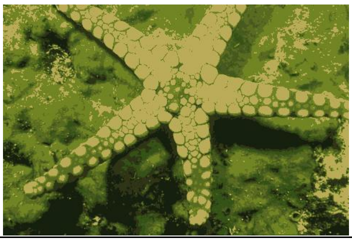
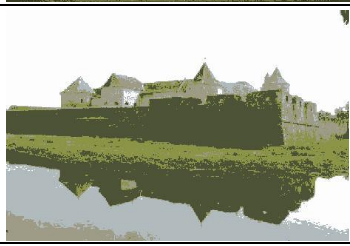
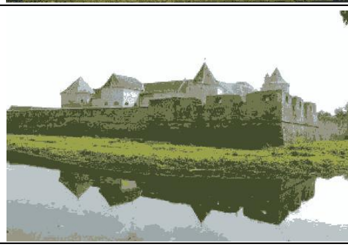
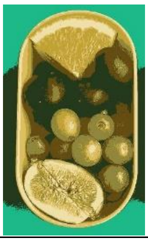
Cleaned image:

Initial Image	
	
Cleaned Images	
m_lamba = 100	m_lamba = 200
	
m_lamba = 400	m_lamba = 600
	

It is observed that at low values of m_lambda, there is almost no change to the original image. However, as we increase m_lambda, there is less noise in the image as seen in m_lambda = 200 where many of the dots have disappeared. As we continue to increase the value of m_lambda, the denoising effects improve, where there are almost no more noise artefacts in the image. However, too large a value of m_lambda will have detrimental effects on the image. Such as when m_lambda = 600, we start to lose bits of the original image such as a part of the letter R being cut off.

(Code for part1 has to be put in the same folder as the Bk_matlab files. Will encounter an error while running if not within the same folder.)

Part 2. Color segmentation

Image	K=3	K=5	K=7
			
			
			
			
			
			

Discussion

Drawbacks:

1. The K-means algorithm takes a long time to run. Using 1000 iterations on image 216053, K=3 takes about 13 seconds while K=10 takes 34 seconds. As the number of clusters increase, the total computing time for the algorithm increases.
2. Depending on the initial clusters of the K-means algorithm, we may end up with different cluster centers after clustering. This would result in inconsistent outputs.
3. The graph cut algorithm has a large memory consumption. To set the neighbour costs, we would require a (NumPixels x NumPixels) sized array. This is method will be unfeasible for large image sizes.
4. There are still instances of cutting isolated pixels.

Improvements:

1. To improve the reliability of the K-means clustering, we can run the K-means algorithm multiple times with different random initial clusters. After which, we pick the clusters with minimal loss.
2. We can group pixels together into larger units (e.g. 3x3 squares) and take their average pixel intensity to be the unit representation. We can then do the graph cut algorithm and assign a label to the units, and all pixels within the same unit get the same label. This would help reduce the overall memory consumption as the input size used for the graph cut is decreased, and the grouping could help reduce isolated pixels.