SYDE 372 - Lab 3

10 April 2018

Professor: Dr. Alex Wong

Ali Akram 20526098

Presish Bhattachan 20553154

Chenlei Shen 20457272

Timothy Tsang 20556306

1. Introduction

This lab focuses on using built classifiers to classify images, specifically texture images using MICD classifier and K-Means clustering methods. Here, classification with labels and without labels are used. Additionally, texture analysis is performed on several images using predetermined extracted features.

2. Feature Analysis

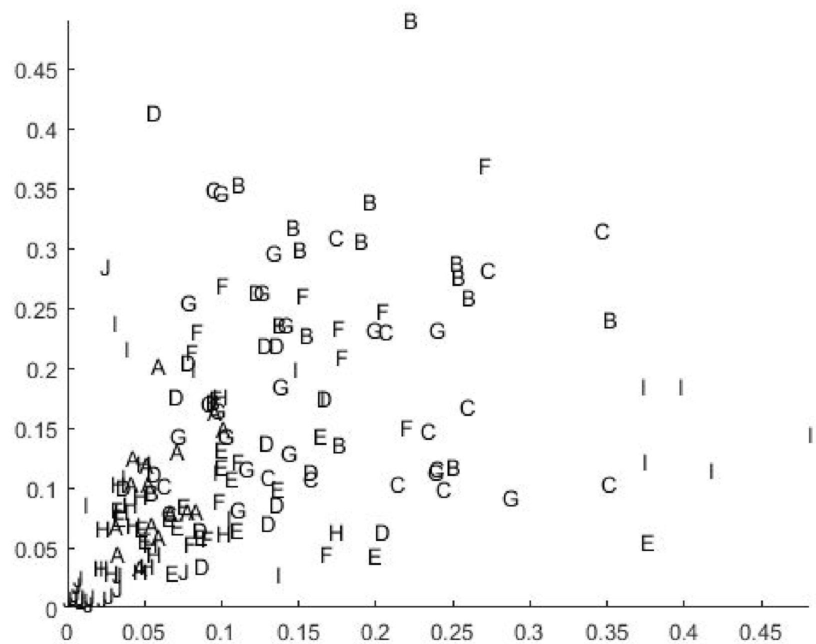

Figure 2-1: Alphabet Plot of Feature Points

In terms of image that are quite similar, paper and corkscrew (f and g) being mixed up, due to the similarity of the textures that each of them show. The pattern of the structures are similar to each other. Cloth and stone may also be mistaken for one another, due to the fact that part of the cloth zoomed in looks similar to what the rock looks like.

In terms of images that are unique among the others, the face seems to be the most distinct from all of the other objects. There are no consistent patterns within the face structure, so that means that none of the other images are similar to the face structure, or have patterns similar to it. The raiffa is also another object that doesn’t have components or features that are similar to other images. While other images do have repeating units, the contrast between the light repeating units and the dark repeating units makes it unique, where raffia is the only image with a light set of repeating units.

3. Labelled Classification

The results of the MICD classification are available in Table 3.1, Table 3.2 and Table 3.3. Each row adds up to 16, the total number of features points in each feature set. By qualitatively observing the 3 tables, it is clear that as n increases, the misclassification rate decreases. Table 3.3 contains the confusion matrix for n = 32. In comparison to Table 3.1 and Table 3.2 that contains the confusion matrix for n = 2 and n = 8, Table 3.3 have a much lower misclassification rate since most of the values are 0 except for the diagonal values, which are the correct classifications. Where in n = 2, there was only 1 image that classified better than 75%, n =4 had 4 images that classified better than 75%, and in n = 32, more than 80% of the images classified better than 75%. This is expected since as the n, the number of pixels in each feature block increases, each block have a better representation of features of the image leading to better overall performance. With a higher block size, distinguishing features can be more easily seen, leading to a better overall classification.

Table 3.1 Test 2 Data (n = 2)

	Classified									
Truth	1	2	3	4	5	6	7	8	9	10
1	3	0	0	1	4	1	0	5	2	0
2	0	6	1	2	0	3	3	0	1	0
3	0	1	1	2	2	2	7	0	1	0
4	2	2	3	2	1	4	0	2	0	0
5	2	2	0	1	4	1	1	1	4	0
6	1	1	1	2	1	5	3	1	1	0
7	0	3	1	2	1	3	4	0	2	0
8	2	0	0	3	5	1	0	5	0	0
9	0	1	3	2	2	3	4	0	1	0
10	1	0	1	0	0	0	0	2	0	12

Table 3.2 Test 8 Data (n = 8)

	Classified									
Truth	1	2	3	4	5	6	7	8	9	10
1	15	0	0	1	0	0	0	0	0	0
2	0	13	3	0	0	0	0	0	0	0
3	0	2	4	0	1	5	0	0	4	0
4	1	0	1	9	0	1	3	0	1	0
5	2	0	0	0	8	0	0	3	3	0
6	0	0	4	5	0	3	4	0	0	0
7	0	2	4	0	0	3	7	0	0	0
8	3	0	0	0	0	0	0	13	0	0
9	0	0	1	1	7	2	0	0	5	0
10	2	0	0	0	0	0	0	2	0	12

Table 3.3 Test 32 Data (n = 32)

	Classified									
Truth	1	2	3	4	5	6	7	8	9	10
1	16	0	0	0	0	0	0	0	0	0
2	0	16	0	0	0	0	0	0	0	0
3	0	0	8	0	0	8	0	0	0	0
4	0	0	0	16	0	0	0	0	0	0
5	0	0	0	0	16	0	0	0	0	0
6	0	0	1	0	0	14	1	0	0	0
7	0	0	0	0	0	3	13	0	0	0
8	1	0	0	0	0	0	0	14	0	1
9	0	0	0	0	1	0	0	0	15	0
10	1	0	0	0	0	0	0	4	0	11

4. Image Classification and Segmentation

Next, an image with multiple images and different textures was classified and segmented. The image, which has data from f8 data set, was classified using the MICD classification. There were two features that were looked into and each pixel in the multf8 was classified, with the following results:

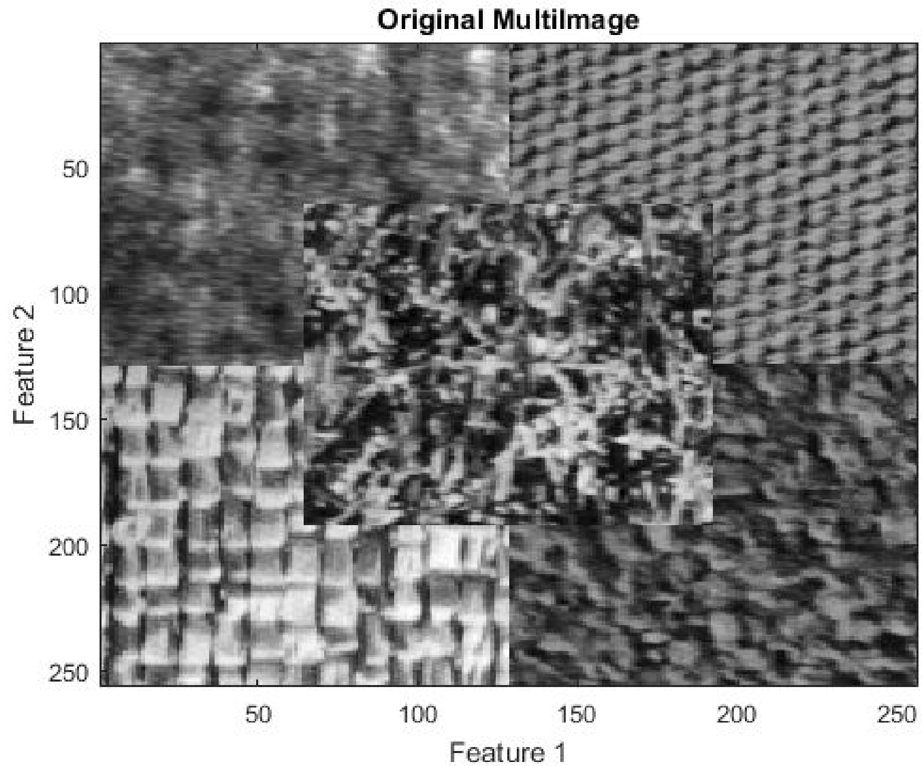

Figure 4-1 Original Multi Image

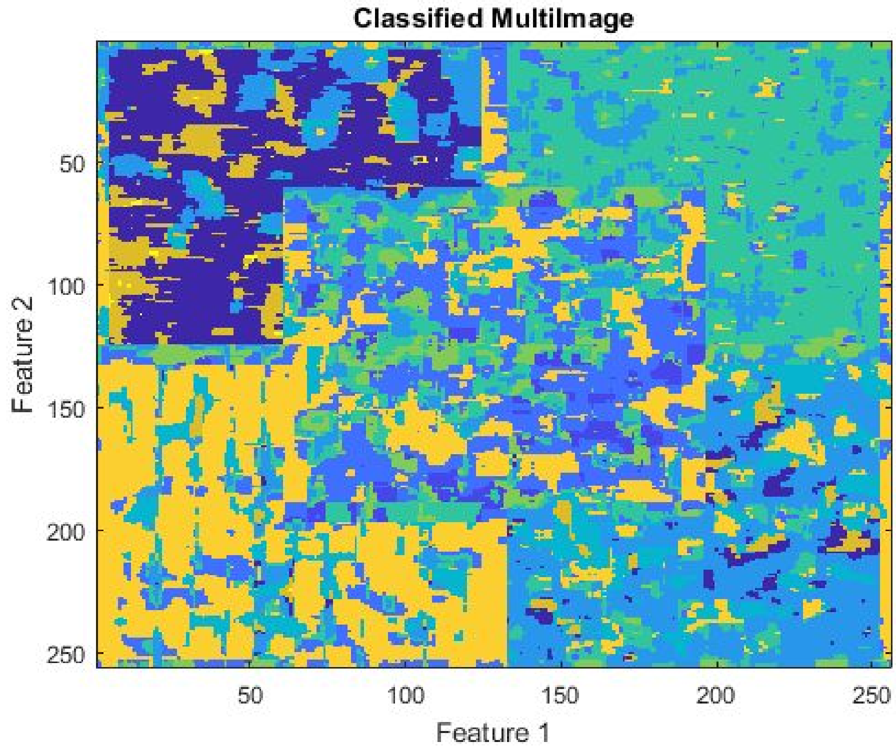

Figure 4-2 Classified Multi Image

Here, we can see that the images were quite similar, seeing that each quadrant has basically one color that dominates compared to the others. In the center, you can see more of a mixture of different colors, which represent the various classes, since the middle image had texture that was the most convoluted and hardest to distinguish, compared to others. Using the MICD classifier, it seemed to be a good fit to match the various textures with the corresponding class. Of course, more work has to be done to improve on the classification and segmentation process, since there is still a lot of mixture of colors that are not belonging.

5. Unlabelled Clustering

5.1 K-means Clustering with K=10

The result of the initial 3 runs of the K-means algorithm are shown in Figure 5-1, Figure 5-2 and Figure 5-3. Feature points from f32 are plotted in the backgrounds represented by a letter of the feature set class (texture) using the `aplot` function. The green circle points represents the initial random prototypes and the red circle points represents the converged prototypes. Note that some of the red circle points, which are the converged points overlaps and covers the green circle points, the initial random points. Few more K-means algorithm were ran but the results are not shown here.

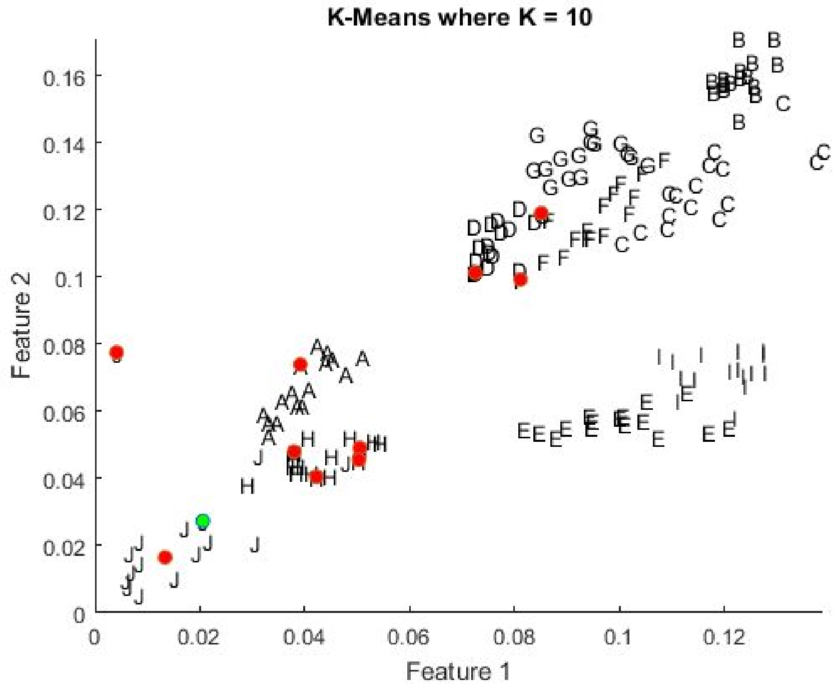

Figure 5-1 K-means First Run

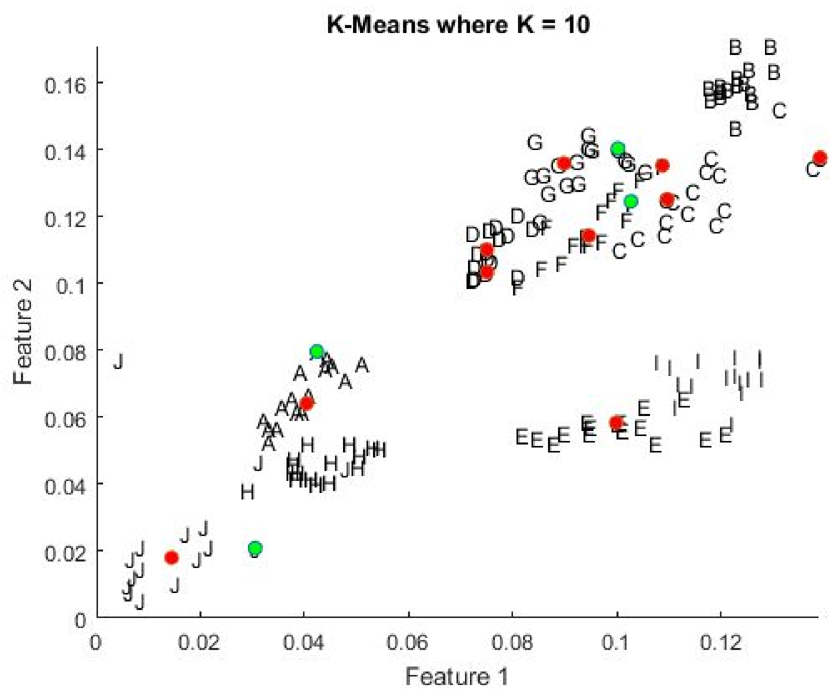

Figure 5-2 K-means Second Run

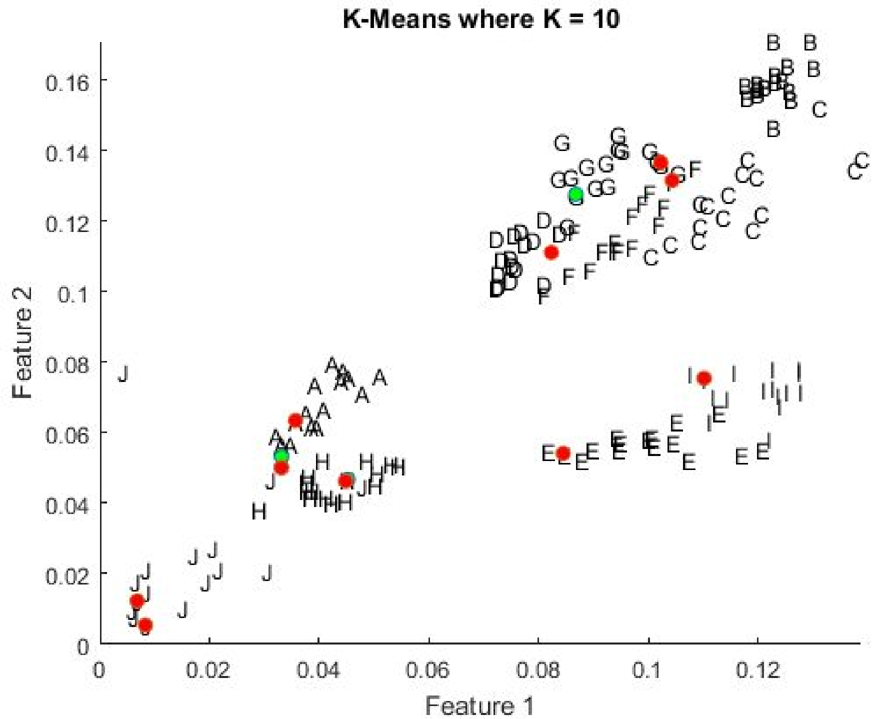

Figure 5-3 K-means Third Run

By observing the plots shown above and the plots generated by more runs, it is clear that the converged prototype varies in each trial. Since all other variables stay the same except for the initial prototypes that varies in every run, it is safe to confer that the variation of the converged prototypes in every run is highly dependent on where the initial prototypes are.

5.2 Fuzzy K-means Clustering with b = 2

The Fuzzy K-means plots are generated using MATLAB's fcm function. The red circle points show the converged points. The steps of the algorithm is found on the MATLAB website [1].
In the first step of the fcm function, random cluster membership values are initialized. Then, the cluster center is calculated with the following formula.

$$c_j = \frac{\sum_{i=1}^{D} \mu_{ij}^m x_i}{\sum_{i=1}^{D} \mu_{ij}^m}.$$

Next, the cluster membership values are updated with the following formula.

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{N} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{\frac{2}{m-1}}}.$$

Finally, the process above is iterated to improve the objective function J_m.

Similar to k-means algorithm, it was found that the converged points varies in every run from looking at the 3 trial runs displayed below and more trial runs that are not shown here.

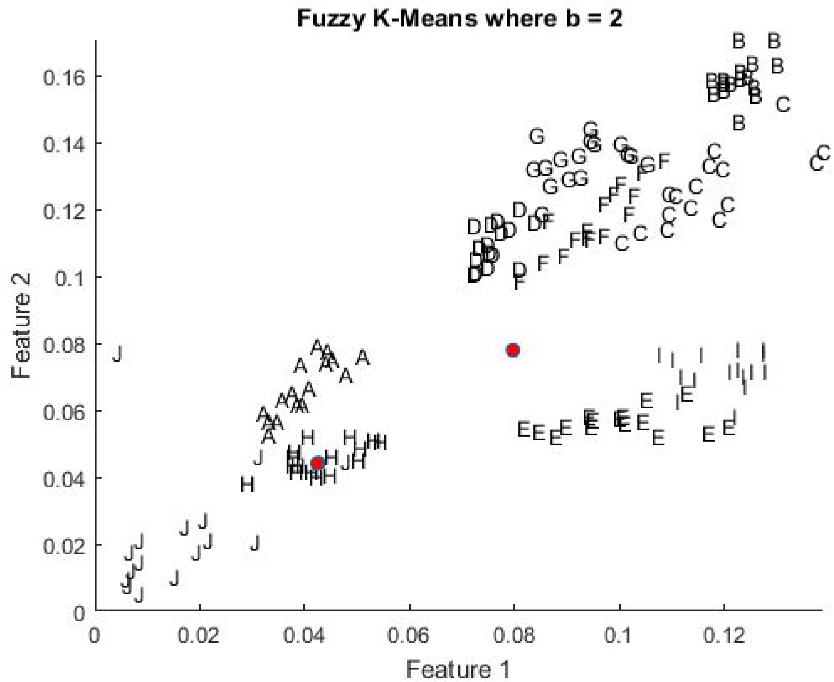

Figure 5-1 K-means First Run

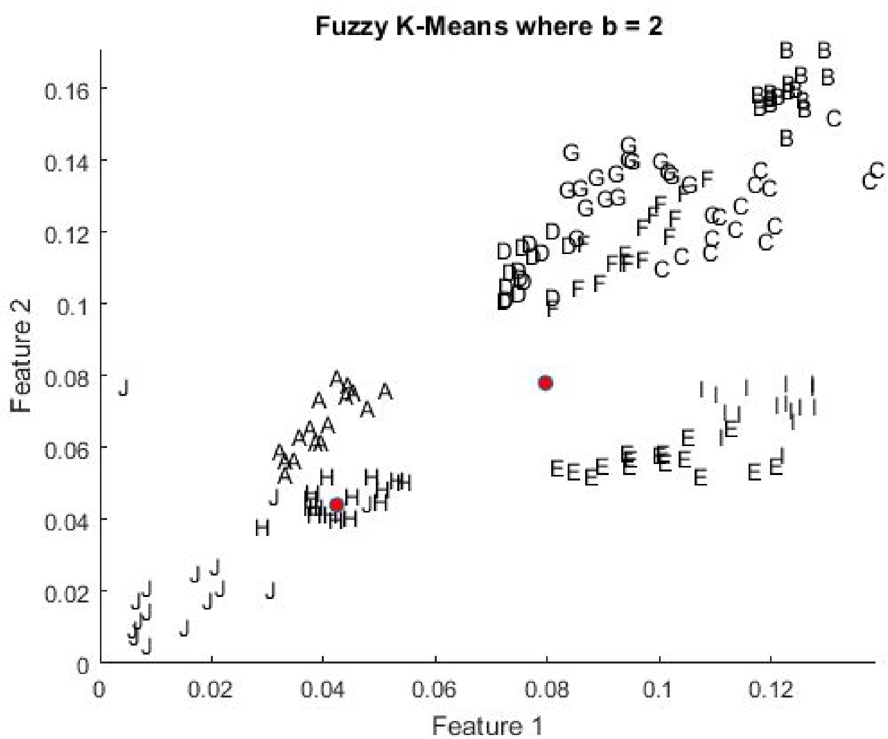

Figure 5-2 K-means Second Run

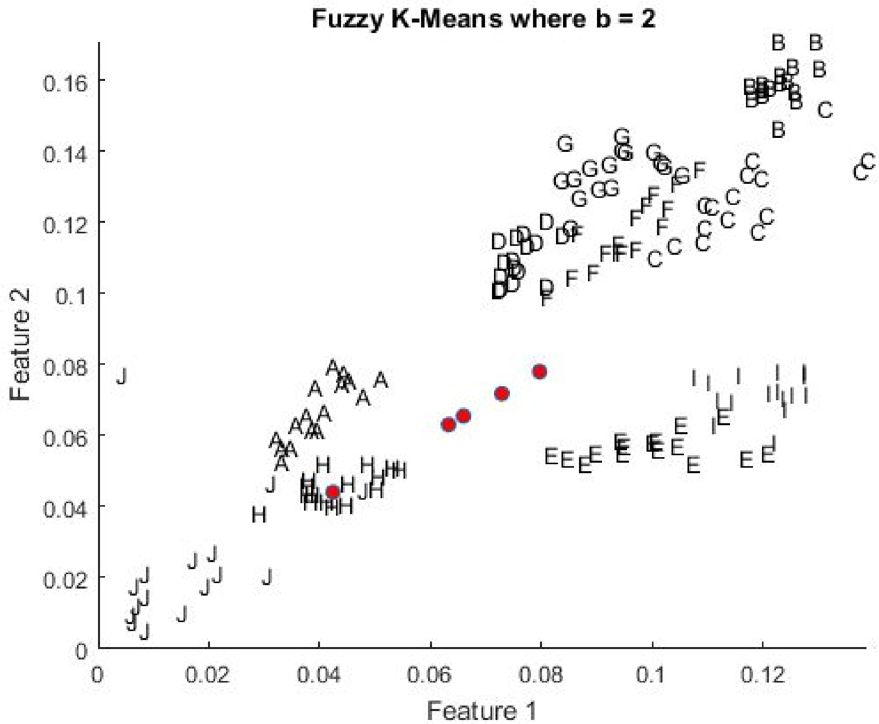

Figure 5-3 K-means Third Run

5.3 K-means Conclusion

In comparison to the MICD classifier for n = 32, the unlabelled clustering have a much poorer performance. In the unlabeled clustering plots for both k-means and fuzzy k-means, most of the time the converged prototypes are not centered in the feature set cluster and sometimes different converged prototypes even overlaps the same feature set. In addition, there are the variability of the resulting prototypes due to random initial prototype.

By taking all of the points in multf8 and plotting it in Figure 5-4, our prediction is that the unlabelled clustering algorithm would perform poorly. As seen in Figure 5-4, it’s difficult enough for human eyes to discern the feature points. The unlabelled clustering technique that relies on distance of the feature points to prototype to converge would have a even more difficult time generating a good prototype.

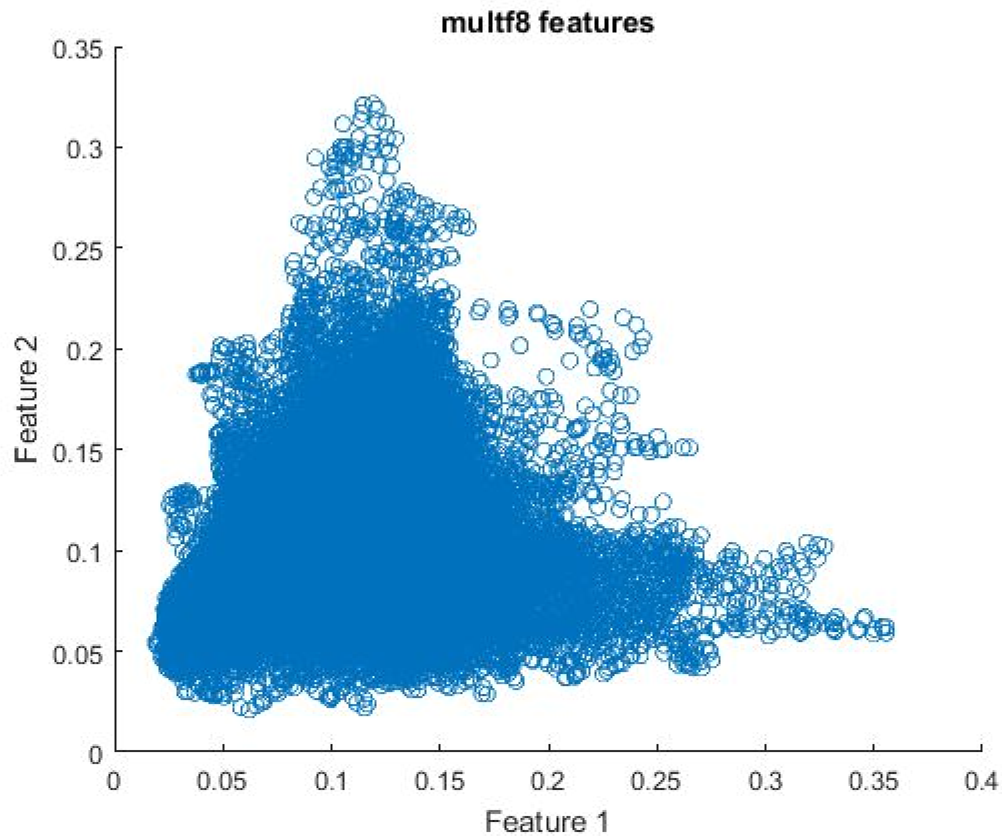

Figure 5-4 multf8 Features Plot

6. Summary and Conclusion

After completing texture analysis and image classification on the test images and data, it could be concluded that as the number of pixels for each feature block increases, the blocks would be better representations of the features in the images. Additionally, higher block size would allow features to be distinguished better, which leads to an overall better classification.

Unlabelled clustering is also something that performs poorly against identified clustering. For the test images, it could be noted that the prototypes found were not centered in the feature set cluster and many times the converged prototypes would overlap different feature sets. Overall for this lab, the performance of labelled clustering was more superior in terms of correct classification than unlabelled clustering.

Reference

[1] https://www.mathworks.com/help/fuzzy/fcm.html