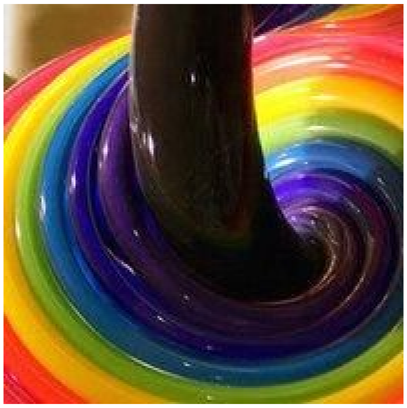
Larry Gates, Aaron Mercier

CM1004

CSSE463

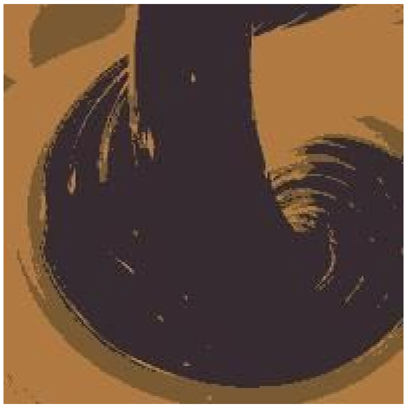
Lab 6: K-Means Cluster

Original Image:



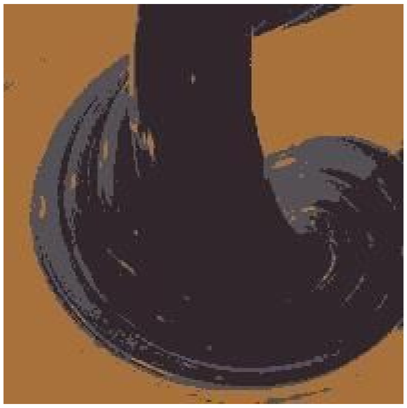
We used this image because of the multitude of colors available in the image. Since we could choose a larger number of clusters, this would make it easier to see how the mean was successful. From our examination of the image, we counted about 10 different colors.

K=3 Clustering (Iterations = 10):



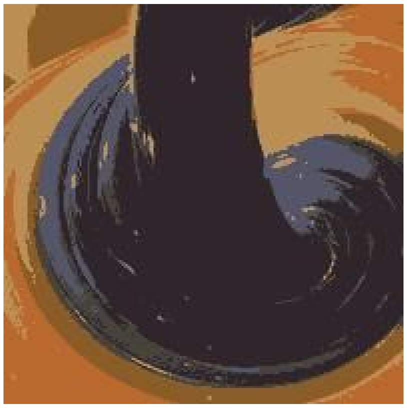
The resulting image when using K=3 clusters shows that the algorithm appears to be working as expected. The results show 3 distinct colors representing the 3 clusters used by the algorithm. The brighter colors such as the red and yellow appear in one cluster, the green and darker orange colors as another, and the dark colors such as the black, purple, light blue, and dark blue appear as the third cluster. We knew 3 clusters were not enough to display an image similar to the original, but the algorithm did a reasonable job at separating the colors.

K=3 Clustering (Iterations = 20):



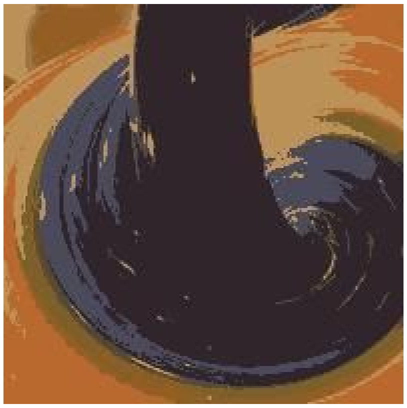
Because the algorithm runs for a set number of iterations passed into the function as a parameter instead of checking for convergence, we tested with k=3 with 20 iterations instead of 10 to see if we would achieve different results. Because the results are different from the 10 iterations, this could be a result of the means not converging in 10 iterations or a result of the means being initialized randomly at the start of the program (although running twice produced similar results). The red, yellow, and green were all clustered in one, the bright blue/purple as another, and the dark purple and black as the last cluster.

K=6 Clustering (Iterations = 10):



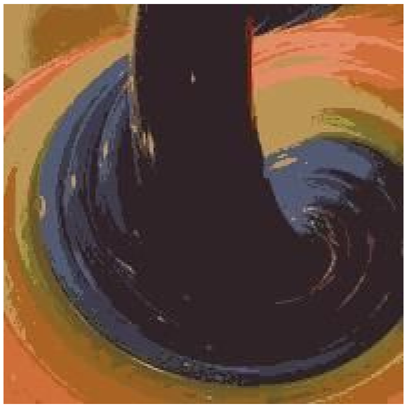
We estimated that there are about 10 distinct colors in the image so 6 clusters would be in the middle between the low K-value (3) and the estimated value (10). The results were surprisingly good. We can see that the red and yellow were combined into one color, the green was separated properly, the blue/bright purple were combined into one color, and the black and dark purple still appear as one color as well.

K=10 Clustering (Iterations = 10):



Because the image contains about 8 main colors (red, orange, yellow, green, light blue, dark blue, purple, and black), using 10 clusters should get close to separating all of the colors. The iteration count is consistent through our tests, besides one case, should appropriately divide the clusters in a short amount of time. The 10 iterations also seem like an appropriate value for convergence, since it is obvious there are at least 8 colors, and various created by shadows. Not all the main colors were detected in the image. One problem is how similar the light blue, dark blue, and purple are to each other, along with the red and the orange being hard to distinguish in the original image. The shadowing takes part of the blame, making colors darker. The background in the bright light is also very close to the yellow in the ring.

K-40 Clustering (Iterations = 10):



As an extreme case, we wanted to run with 40 clusters to see if we could get an image that is closer to the original image. The clustering result allowed for more distinct colors to be visible over using only 10 clusters. The algorithm returned pixels in 39 clusters of the 40 clusters. A possible explanation for losing a single cluster could be a mean being extremely close to another, and the other mean taking all of the pixels. Using more clusters showed significant differences in color, showing the segments of colors. Since it is obvious there are at least 8 colors (red, orange, yellow, green, light blue, dark blue, purple, and black), k=40 provide enough to meet those standards along with the shading of the colors.