CSCI 631 Foundation of Computer Vision
HW 03

First name: Ashwath Sreedhar
Last name: Halemane

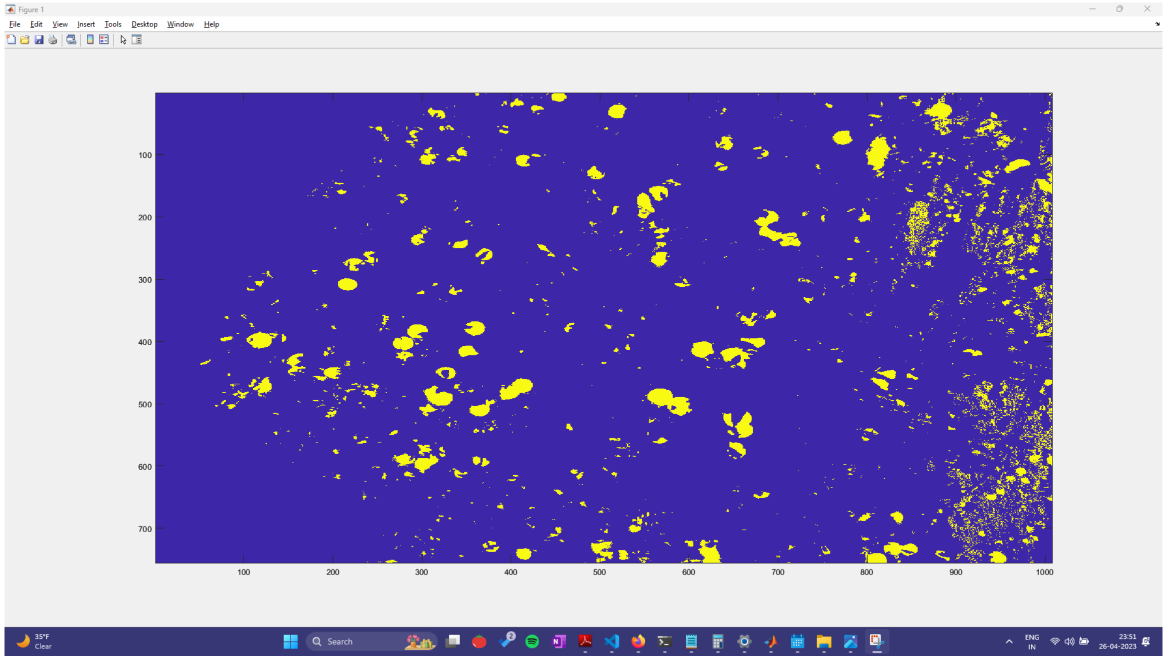
1. **Conclusion:**

   I learned how to use Mahalanobis distance to segment images for this computer vision homework. A driver function that executes the three subroutines HWNN_Email1_Cluster_Colors, HWNN_Email1_FIND_RASPBERRIES, and HWNN_Email1_FIND_ORAGES was the intended outcome.

   The first program, HWNN_Email1_Cluster_Colors, reads in an image and uses k-means to separate it into several colors. Since Matlab's kmeans() algorithm no longer supports the Mahalanobis distance, my strategy was to use the Squared Euclidean distance. In a color space that was almost perceptually homogeneous, this method performed well. However, sub-sampling the image is a feasible alternative to speed things up because k-means can take a while to execute on a huge image. K-means can also be pre-seeded with starting seeds.

   The second subroutine, HWNN_Email1_FIND_RASPBERRIES, reads in an image and uses a built-in classifier to determine whether certain pixels are raspberries or not. I created the classifier by gathering raspberry-shaped pixels from the supplied photos and then building a cluster center in a particular color space. I made the decision to employ the Mahalanobis Distance Classifier, which determines if a pixel is a raspberry if it is sufficiently close to a red center.
   The third subroutine, HWNN_Email1_FIND_ORAGES, reads in an image and uses k-means to separate it into various colors. The method is the same as the second subroutine. I could have also used kmeans to find the centroids of oranges and used that clusters centers as orange point and used that to find mahal of pixel data to the cluster center as well.

CSCI 631 Foundation of Computer Vision
HW 03



Overall, this homework helped me have a practical grasp of how to divide photos into different parts using k-means and Mahalanobis distance. It also gave me the chance to practice using classifiers that were already established.