Caitlin Kimbrell
Nikhil Raina

HW 07

The program takes an image as input for the first parameter, then a 1 or a 0 for the second parameter. The image passed will have its pixels that are oranges identified. The 1 or 0 passed will change the interactive flag. If the interactive flag is true with a 1, that means we will prompt the user to interact with the image, selecting pixels in the image that make up oranges at least 5 times, and then selecting pixels that do not make up oranges for the same number of times. To submit, the user has to hit enter. After that, all the data is saved in an external file, for when the interactive parameter is 0 and we want to just use stored data.

Once all that is done, we convert the image into an HSV image, and then separate the image into different variables, isolating hue, and values. Then using the user input or stored data, we identify the hues and values of the pixels selected as objects of interest, in this case oranges, and get the indices. We do the same to the background, or the objects that are not of interest, which is in this case everything except the oranges. Then we plot the respective hue and value points for foreground and background. Foreground value points, the oranges, are indicated by red squares. Background value points, everything else, are indicated by blue circles.

Then we convert the image to LAB format and separate the A and B channels into different variables. Then we pull the specific x and y coordinates from the foreground, but from the A and B channels. We do the same to the background. We plot it the same way we plotted in the previous paragraph.

Once those are done, we begin to compute the mean and covariance for the foreground data points from ab space. We do the same to the background data points. We then calculate the mahalanobis distance of the image a and b channels with reference to the foreground a and b channels We also calculate the mahalanobis distance of the image a and b channels with reference to the background a and b channels. We then classify pixels as foreground objects if the mahalanobis distance is less than the distance from background mahalanobis distance. The resulting binary matrix will highlight oranges in white and the background will be black. We display this new binary image. This is our classification using mahalanobis distance.

We then show a plot of the foreground distances to give the user an idea of what the data looks like. Using these distances and outlier removal, we can throw out data that might be displayed as oranges under the mahalanobis classifier, but may not be oranges at all. We throw out all data outside of one standard deviation and then readjust the mean accordingly. Using distance we get the threshold and use that to determine whether a pixel makes up an orange or not. We binarize the image according to this threshold then display.

Some problems that we ran into was how exactly to include the interactive flag without having unreachable code. We did this by making it a parameter passed into the function. Another problem we ran into was that the example code had the naming convention wrong when retrieving the results of the method histc(). They were flipped, so we flipped them back to where they should be to avoid confusion.

Conclusion

We learned a lot in this homework assignment. We had previously never used user interaction in my matlab code, so asking and then taking in input from the user was brand new

to us. We also used many unfamiliar functions such as mahal(Y, X) which returns the squared mahalanobis distance of Y in reference to X. I also thought this was a fun method name because mahal in my native language, Tagalog, means love, and I had a laugh every time I typed it out or looked it up. We also learned how to use distances to create a classifier, and how to make those classifiers even better by incorporating outlier removal. We also learned that it's hard to tell when oranges are oranges even for humans. I had an orange that I thought for sure was a weirdly small grapefruit until I cut into it. It makes me wonder if this classifier would’ve known it was an orange before I had (probably).