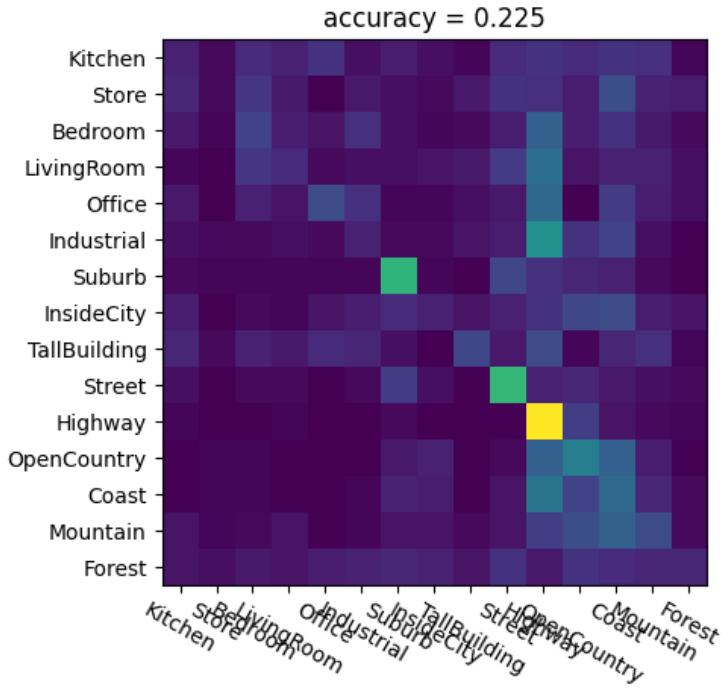
The main purpose of this assignment is to classify images of different scenes with three different methods and explain their performance by visualizing their confusion matrix and accuracy.

**1. Tiny Image representation + KNN classifier**

In this method we make features of our training and testing images and will call it feature using get_tiny_image. This features can be achieved by resizing the images to 16x16 pixels and will be subtracted by mean and normalized at the end.

We will make a model and fit it with training features and labels, then finally can predict the labels of our test set using NearestClassifier.

I could get 22.5% accuracy by using k=10 neighbors for my KNearestClassifier.

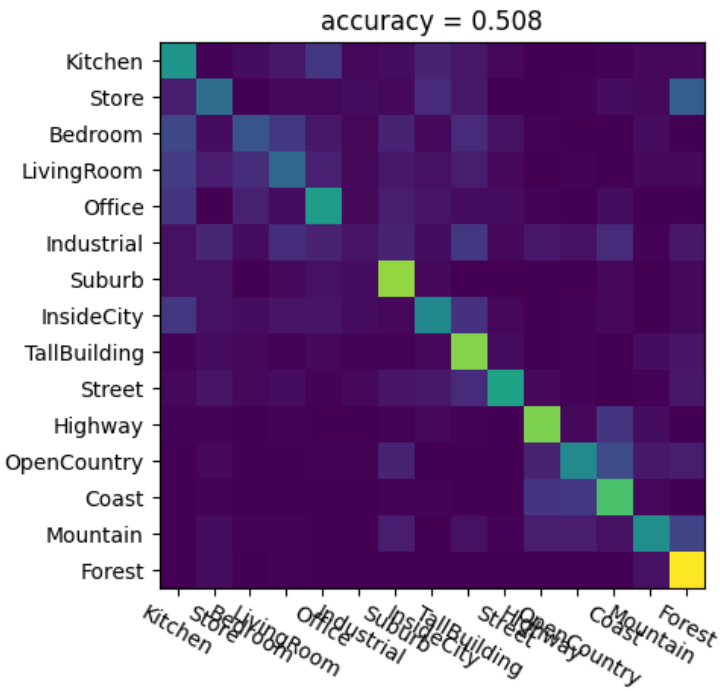



**2. Bag Of Word representation + KNN classifier**

In this method, we will practice Bag of Word representation. We need to follow Algorithm 2 of the HW for this method.

In this representation, we have to follow few steps to extract features and build representation:

1. Extract sift descriptors of all training and testing images and then make a pool of sift features. I've used stride=16 meaning I will move every 16 pixels and that would be my key point x,y coordinates, then will compute the sift descriptors of those key points with given x,y and size of 16 for key point diameter.
2. Build a vocabulary of visual words for our training set using Mean clustering and find the center of each clusters. I've used default values for Kmeans parameters.
3. Compute Bag of Words for training set by combining sift features and cluster centroids and then apply KNearestClassifier. I've used 50 for dict size in the program.
4. Compute Bag of Words for test set by combining test sift features and cluster centroids of training set and then apply KNearestClassifier.
5. Finally using KNearestClassifier again to make predictions on test set. k=15 gave me accuracy of 50.8%




**3. Bag Of Word representation + SVM classifier**

Following the exact steps from method 2 except using SVM classifier to make the predictions. This method is expected to have the highest accuracy of all these three methods which is greater than 60%. I've used C=4 and dictionary size=70 for my LinearSVC classifier and the rest of the parameters are having default values and finally got 63.2% accuracy.