

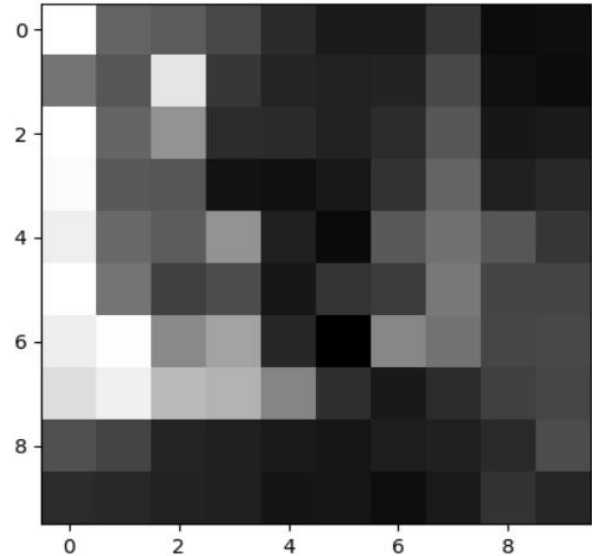
Assignment 3 – Scene Recognition

Part 1: Tiny image KNN classification

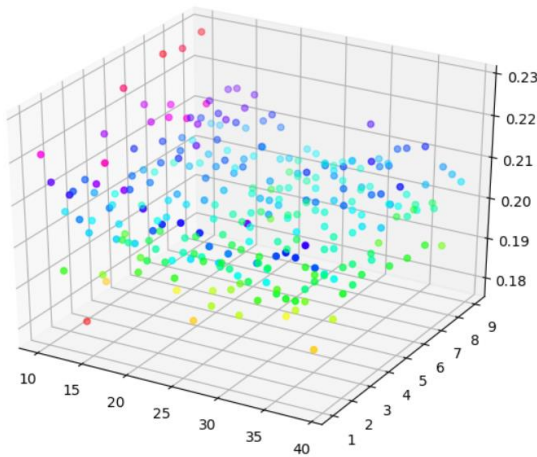
In this part of the assignment, we have used K nearest neighbors classifier for scene recognition. First, we reshaped the image using OpenCV's resize method and then applied k nearest neighbors algorithm to classify test labels. Image size of 10, k of 9 and metric of Euclidean gave the best accuracy of 22.7%



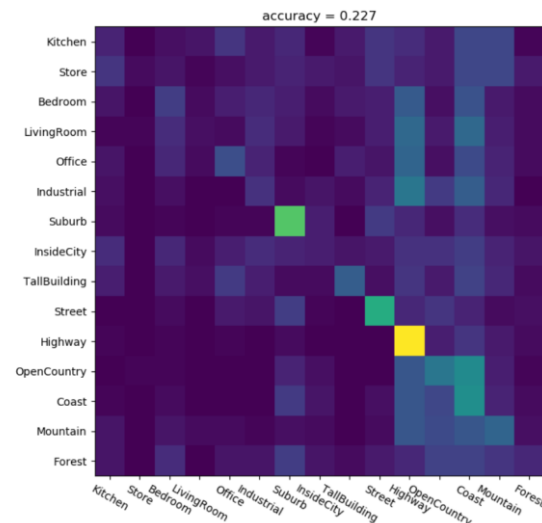
Original image



Resized image



Accuracy plot (x: imageSize, y: k, z: accuracy)



Confusion matrix for knn

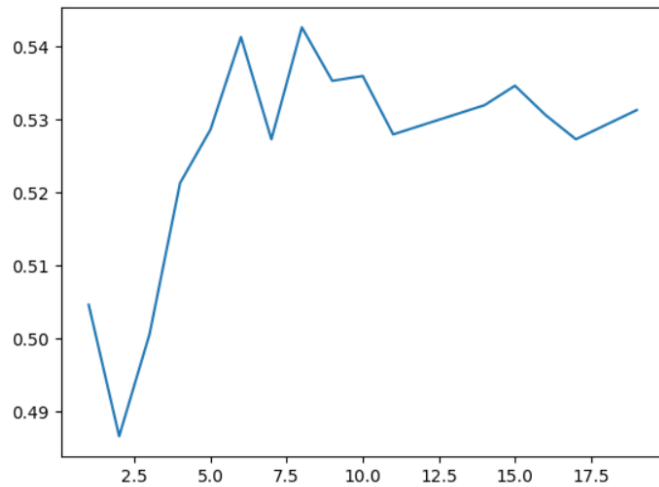
Part 2: BoW + KNN

In this part, we first computed bag of words for the given train images by following steps:

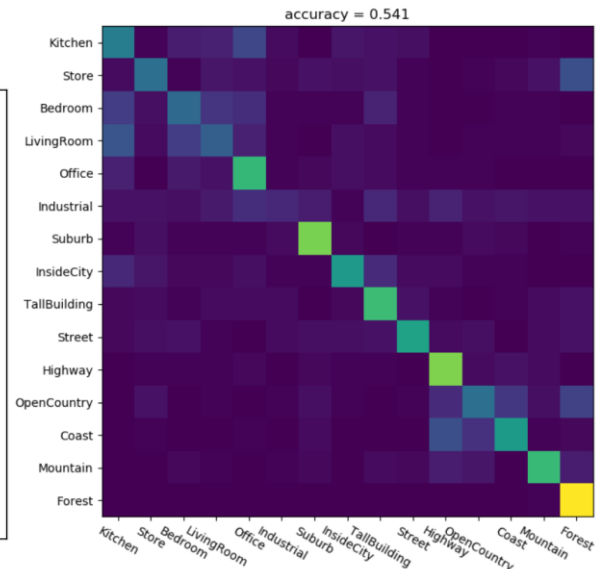
1. Compute d-sift of train images with **size of 20 and stride of 10**
2. Prepare a dense feature set of train images and find **200 cluster centers** using k-means algorithm
3. Prepare vectors(representations) for train and test images using nearest neighbors algorithm – these feature are also know as bag of words

These bag of words features are being used for Part 2 and Part 3, since stride of 10 and vocabulary size of 200 gave me good results for both, I am not computing the vocabulary and bag of words features again in part 3 and

simply reading it from saved numpy files. K nearest neighbors is applied on these features to get the predictions for test images. **K = 8** gave the accuracy of **54.1%**.



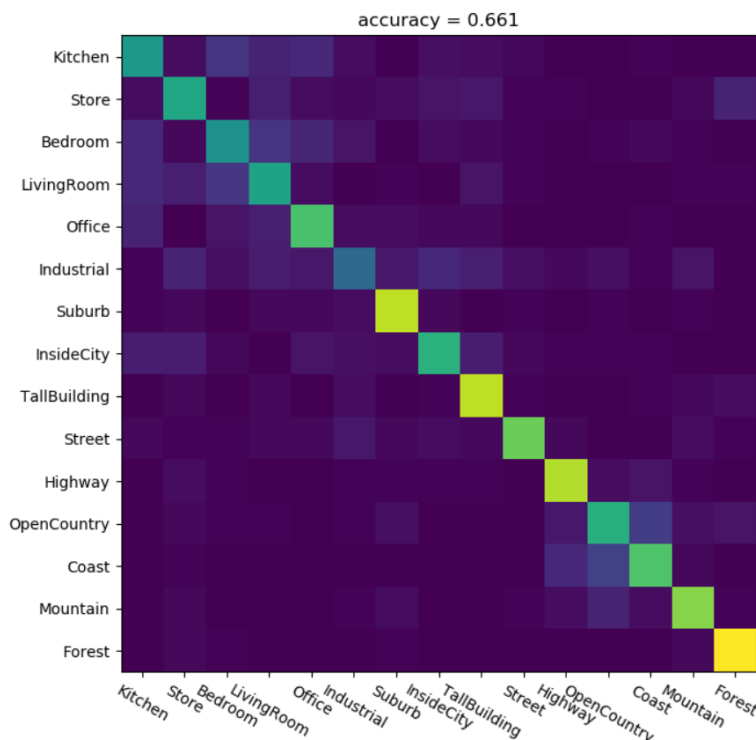
Accuracy graph (x: k, y: accuracy)



Confusion matrix Bow+KNN

Part 3: BoW + SVM

In this I have applied LinearSVC to our feature set. 15 models were built for each of the classes and decision_function method of model is used to get the confidence of each model's prediction. These models were used as one vs rest classifiers and prediction with maximum confidence is considered to be the predicted class. Since parameters of bag of words feature generation didn't change from part 2, in part 3 I am just reading the values from numpy files and the code of generating these is commented. LinearSVC with **C value of 1** gave **66.1%** accuracy.



Confusion matrix BoW+LinearSVC

Model	Parameters	Accuracy
Tiny image + KNN	Image size: 10 K: 9 Metric: euclidean	22.7%
BoW + KNN	Size: 20 Stride: 10 Dictionary size: 200 K: 8	54.1%
BoW + LinearSVC	Size: 20 Stride: 10 Dictionary size: 200 C: 1	66.1%

Parameters and accuracies summary