Project 1 - Practical Machine Learning
Comparison between two supervised methods
Alexandru Rusu - group 507

Dataset

For this project I used COIL-100 dataset (dataset link) which consists of 7200 unique images of 100 different objects. Each object is photographed in 72 different angles: a complete 360 degrees rotation with pictures taken every 5 degrees.

Fig. 1 Images for one of COIL-100 objects

Every image is stored as a **.png* file of size 128x128 pixels with the following name convention: obj[number]_[degrees].png, by that the dataset is already labeled through file name convention.

An important aspect about this dataset is the fact that for every image, the background has been subtracted and replaced with a black one.

Solution proposed

Looking at the dataset, it is obvious that this is an image classification problem. By having the easy to interpret labeling system of the dataset, supervised algorithms can be applied on it.

For this image classification task I used Gaussian Distribution Naive Bayes classifier, K-nearest-neighbors classifier and One-vs-rest Support Vector Machine algorithm.

For these algorithms to be able to classify the images, I had to generate a feature set for every image. I took two approaches to do that. One of them was to simply generate a raw-pixels feature vector. For every image, I did a resize from 128x128 to 32x32 and flatten the image, in the end resulting a nparray of shape (3072,) -- 32x32x3

(RGB channel). The other one was to generate a normalized 3D color histogram from HSV color space using the number of ‘bins’ (8,8,8) per channel.

After iterating through all the images, three lists are generated containing labels, raw-pixels vectors and histogram vectors. Using train_test_split() from sklearn, all data required by classifiers is ready. For training I used 75% of the data, the rest of 25% being reserved for testing.

Gaussian naive bayes and knn algorithm are straight forward and run very fast, considering the fact that the entire dataset has a feature set size of 21.6 MB for raw-pixels and 14.4 MB for HSV histogram. The accuracy obtained for these two methods can be seen in the following figure.

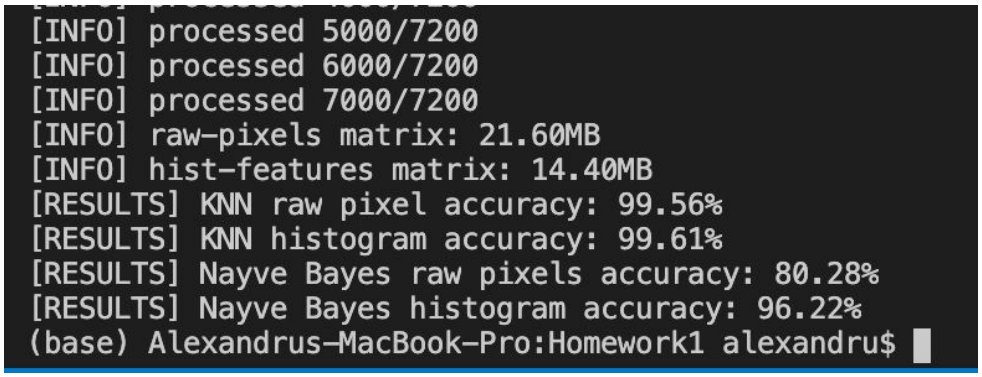

Fig. 2 KNN & Naive Bayes accuracies | feature set space occupied

SVM classifier is a slow way to classify this size of feature set, because internally SVM algorithm needs to generate a matrix containing the entire train-set. This means that for raw-pixels feature set, SVM needs to allocate a matrix of size 16.2MBs x 16.2 MBs, that equals to approx. 263 MBs. A good practice was to give a smaller train-set for SVM approach. By using 25% of the dataset as train-set for SVM One-vs-All classifier, the accuracy is lower than other methods presented, and training time was extremely high compared with the other two. Even if HSV histogram increased the first two methods’ accuracy, for SVM it was the opposite.

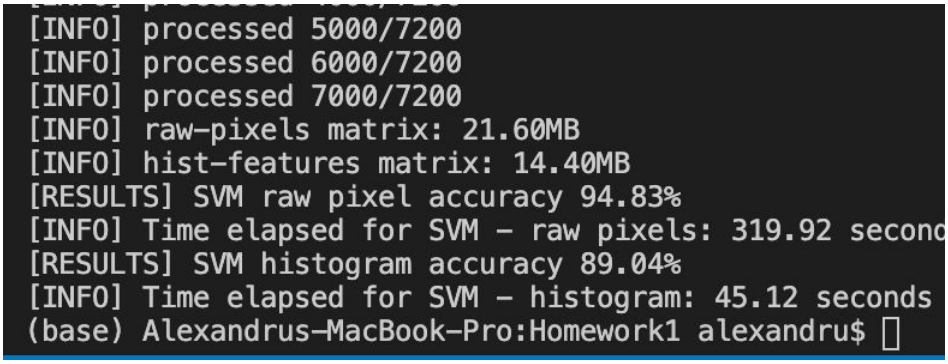

Fig. 3 SVM OvA accuracy and time consumed

Conclusion

Even if the train-set was smaller for SVM, this algorithm is popular for the ability to learn using a small amount of data. The moment when the training was done using HSV histogram and the accuracy decreased, it was caused due to similarity in colors for some objects (red car, red soda can, red cup, etc). KNN algorithm was capable with only one neighbor as classification input parameter, because the dataset contains 100 classes and as in the SVM histogram failure, the color similarity between objects could lead the knn to group together red objects. The Gaussian distribution applied worst on raw pixels train set, due to the sparsity of the pixels and incompatibility to obtain a normal distribution from them. When the histogram feature set got involved, there was a clearer way of how pictures are represented.