Multi-class image classifier algorithm

Team name : Data_Strikers

Team members : Disha Sarawgi, Aman Aniket

Software Used : MATLAB, VL_FEAT (A library of MATLAB)

Feature Extraction Process : dense SIFT (Scale Invariant Feature Transform)

Similarity/Distance Measures : L2 norm (MATLAB function alldist2)

Classifier : VL_FEAT function vl_svmtrain gave the Weight Matrix and Offset constant for the SVM of each class. We calculated the score for each class by the formulae : W’ *BagOf WordsFrequencyHistogramVector + B

Feature Extraction step

We used the dense SIFT feature extraction technique to extract the feature vectors from each image by the function ***vl_dsift*** available in VL_FEAT library. This produces a regular 2D matrix of feature vectors which are used for clustering in the next step. We chose dense SIFT over the standard SIFT as it better time complexity, the frame overlap distance can be altered and produces a regular matrix for a given set of parameters in contrast to regular SIFT.

Training Algorithm

Input Format

Input consisted of a concatenated matrix with each column containing the feature vector of a training image.

Tunable Parameters

1) The number of clusters or visual words in the vocabulary
2) Lambda value in ***vl_svmtrain***

Output Format

A Support Vector machine for each of the classes consisting of a Weight vector and offset value each used for calculating the score for each class to the testing image.

We used a bag of words model for training of SVM. This is a weakly supervised procedure where we first applied a clustering algorithm (K-means clustering, Elken’s algorithm) to obtain the centroids of the clusters or the visual words. We then obtained a frequency histogram for each training image by assigning each feature for an image to a cluster. We then normalized the histogram to make it scale invariant. This gave us the feature descriptors for each training image which was used to train the algorithm by the function vl_svmtrain.

Validation and Parameter Tuning

We validated the algorithm by the validation data provided. Initially the number of clusters taken for the k-means clustering was taken to be *squareRoot(n/2)* as a rule of thumb. Later on, the value was fine-tuned based on CH criterion which takes inter-cluster distance, intra-cluster packing among other parameters to determine the optimal value of the number of clusters. We also fine-tuned the bin size and frame overlap distance in the feature extraction process.

Prediction Algorithm

We predicted the label of the validation/test image by the help of the weight vector and offset value calculated by the function ***vl_svmtrain***. The score for a particular image for a label is **S = W’*BagOfWordsHistFrequencyVector + Offset.**

Interpretation of results on validation data

The bag of words is by far the most adopted approach among the parametric approaches and the non-parametric approaches. Although, we weren’t able to obtain the desired accuracy and there is definitely a lot of scope for fine tuning of parameters and improvement in the algorithm, specially the classification technique.