Object Categorization Using Sparse Nearest Neighbor Distances For Improved Accuracy and Scalability

Sancho McCann and David G. Lowe

Laboratory for Computational Intelligence
University of British Columbia
Vancouver, Canada

Overview

Boiman et al. [1] showed that the common practice of vector-quantizing visual descriptors in the bag-of-words model results in a loss of discriminativity.

Using a feature-based nearest neighbor classification algorithm to compute image-to-class distances achieved state-of-the-art performance.

The method from [1] scales linearly with the number of classes. Our improvements increase classification accuracy and give a significant speed-up when scaling to large numbers of classes.

Naive Bayes Nearest Neighbors

The original algorithm (from [1]):

1. Compute descriptors $d_1, \dots, d_n$ of the query image Q.
2. $\forall d_i, \forall C$, compute the NN of d_i in C: $\mathrm{NN}_C(d_i)$.
3. $\hat{C} = \mathrm{argmin}_C \sum_{i=1}^{n} \|d_i - \mathrm{NN}_C(d_i)\|^2$.

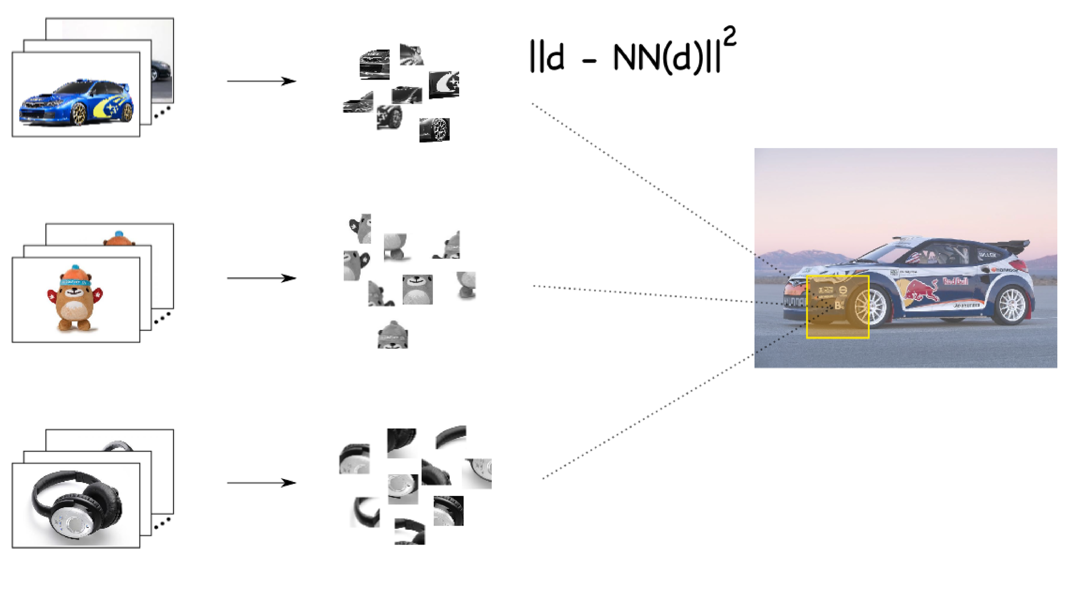

Figure 1: Naive Bayes Nearest Neighbors accumulates the squared distances from each query feature to each of the classes.

Complexity:

N_D is the number of descriptors per image, N_C is the number of classes, N_T is the number of training images per class, and **c** is the number of checks done in the approximate nearest neighbor structure.

$$O(cN_DN_C\log(N_TN_D))$$

Sparse Update Nearest Neighbors

A large time savings can come from changing the search strategy for nearest neighbors. Instead of searching for a query descriptor's nearest neighbor in each of the classes, we search for the nearest neighbors in a merged dataset comprising all training features from all classes.

Searching this large index for a few nearest neighbors is much faster than searching each class's index for a nearest neighbor.

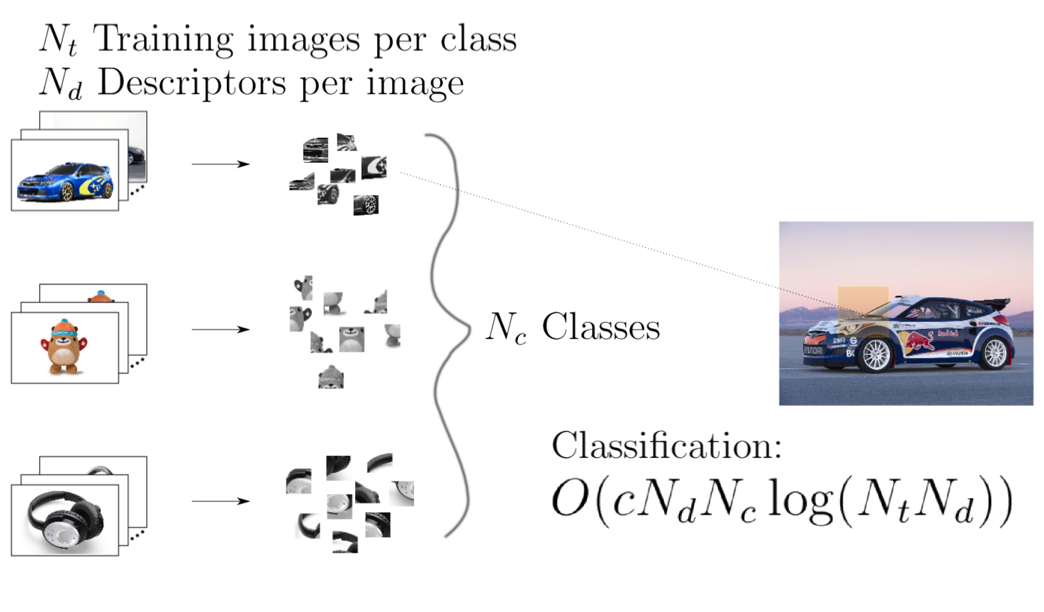

Figure 2: The complexity of the original algorithm is linear in the number of classes.

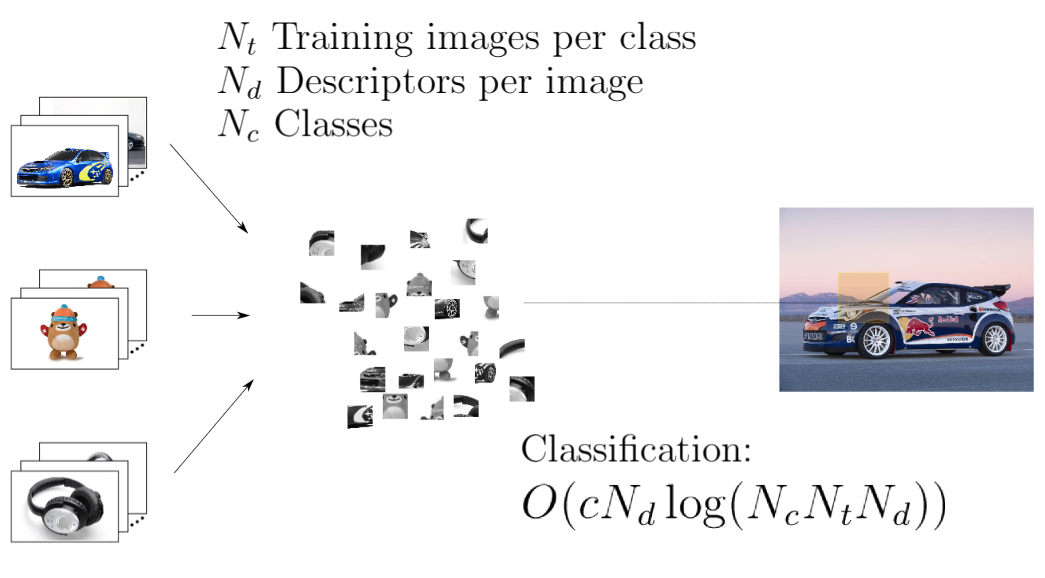

Figure 3: Complexity of our improvement is log in the number of classes.

Additionally, for each query feature, we only find the **k nearest neighbors** in this merged dataset, updating the associated classes' distances, and lower-bound the distances to the non-retrieved classes to be the distance to the k+1 nearest neighbor.

Scalability

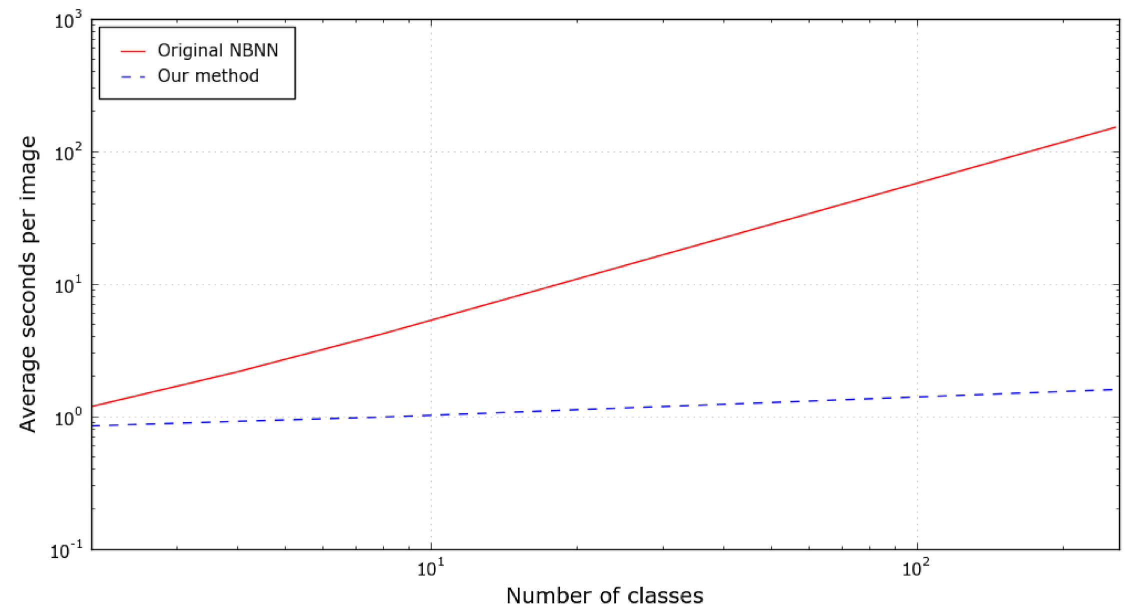

Figure 4: Empirical timing results when increasing the number of classes from 2 up to 256. On Caltech 256, our method was 100x faster.

Accuracy vs Speed

Accuracy-vs-Speed Tradeoff:

We use FLANN, an approximate nearest neighbor search structures based on automatically tuned randomized KD-trees [2]. One parameter is **c**, the number of nodes to be checked in the KD-trees. This parameter affects the accuracy and computation required.

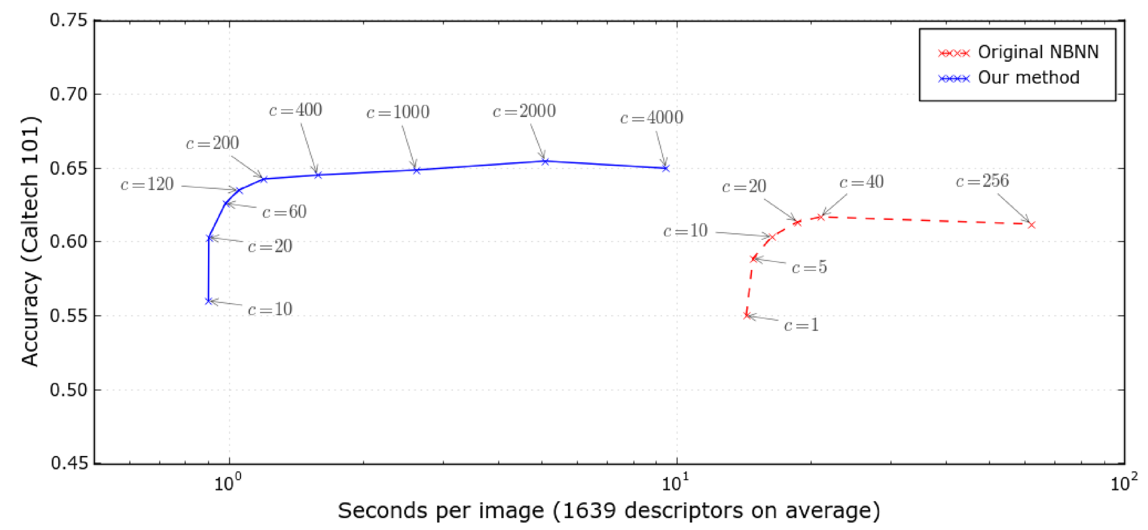

Figure 5: Even doing only a single node check in each of the 101 separate indices is more expensive than one search with thousands of node checks in our merged index. These results are from the Caltech 101 dataset.

Improved Classification

Tuning k:

How many nearest neighbors in the merged index do we need to retrieve? We only need a small number of nearest neighbors, retrieving neighbors from only the most likely classes for each query descriptor.

This sparsity not only saves on computation time, but improves performance of the classifier.

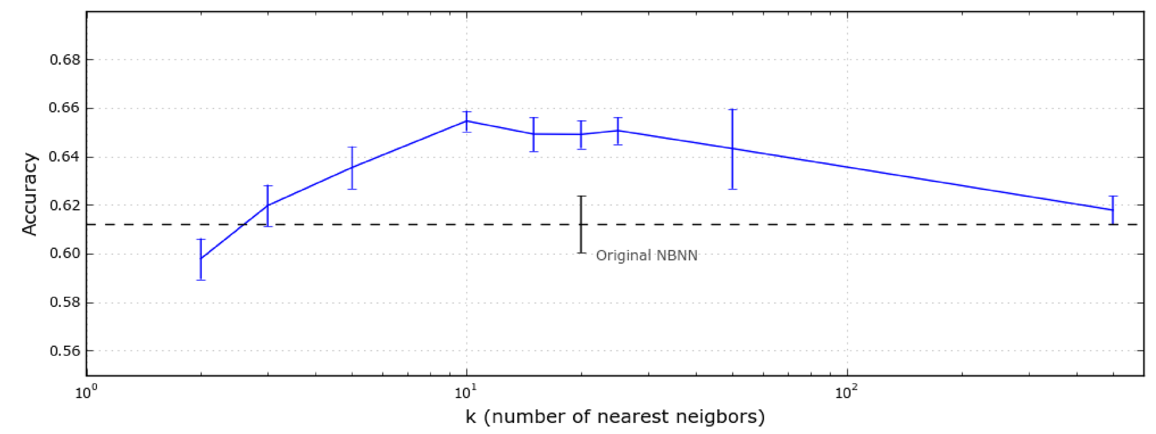

Figure 6: Searching for only the 10-15 nearest neighbors for each query descriptor gives optimal performance on Caltech 101. If many more neighbors are retrieved, enough to find an example from each class, the benefit of the sparsity disappears, and performance reverts to that of the original.

Caltech 101 Results (15 training images per class):

Method	Performance
Spatial Pyramid Match (Nearest Neighbor)	42.1± 0.81%
Spatial Pyramid Match (SPM)	56.4%
Griffin's Implementation of SPM	59%
NBNN (Our implementation)	61.1 ± 1.32%
NBNN (Our improvement)	65.6 ± 0.42%

References

[1] Oren Boiman, Eli. Shechtman, and Michal Irani. “In Defense of Nearest-Neighbor Based Image Classification”. In proceedings of the IEEE Conference on Computer Vision and Patern Recognition (CVPR), 2008.

[2] Marius Muja and David G. Lowe. “Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration”. In the International Conference on Computer Vision Theory and Application (VISAPP), 2009.