Supervised Incremental Hashing

Bahadir Ozdemir[1]
ozdemir@cs.umd.edu

Mahyar Najibi[1]
najibi@cs.umd.edu

Larry S. Davis[2]
lsd@umiacs.umd.edu

[1] Department of Computer Science
University of Maryland
College Park, MD 20742

[2] Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742

This paper presents an incremental strategy for learning hash functions with kernels for large-scale image search. Despite the fact that new images are added to online photo databases every day, no supervised hashing method learns hash functions incrementally for newly added images. We identified three main objectives for our supervised hashing method – as being incremental and parallelizable, avoiding overfitting by better generalization, and balancing $+1/-1$ in learned binary codes.

To address this problem, we introduce *Supervised Incremental Hashing* (SIH); a method based on binary and multi-class SVMs. SIH treats binary codes as intermediate variables between the feature space and the semantic space. In the first stage of classification, binary codes are considered as class labels by a set of binary SVMs; each corresponds to one bit. In the second stage, binary codes become the input space of a multi-class SVM. We formulate our hashing objectives in a joint optimization task that provides better generalization with regularizations and maximizes the entropy by balancing binary codes.

We describe an algorithm that solves the optimization problem efficiently by an incremental strategy. In this approach, the NP-hard problem of finding optimal binary codes is solved via cyclic coordinate descent and the SVMs are trained in a parallel fashion. Considering a dataset with class information, SIH can be adapted to modifications like adding new classes, deleting existing classes, and adding images to existing classes efficiently. Furthermore, we present an upper bound for the convergence of our method when these changes happen.

Figure 1 shows a simulation of our incremental SVM approach on a sample dataset with 6 classes represented by colors in part (a). Assignments of $+1$ and -1 are indicated by filled and empty shapes, respectively. Part (b) shows the changes in the hyperplanes during training.
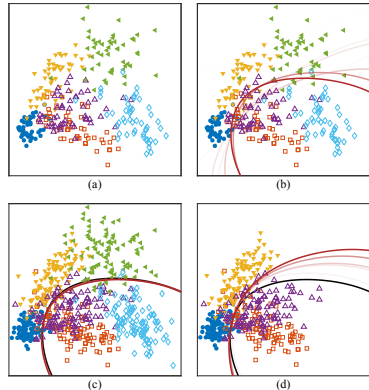

Figure 1: A simulation of SIH.

Increasing transparency of the lines indicates earlier iterations of the execution. In part (c), the number of data points is increased to 600 by adding new points from the same distributions in (b). Red lines represent the hyperplanes when our method is initialized with the solution at (b) shown by a black line. Finally, in part (d) two classes are deleted from the dataset.

We evaluate our method on three large-scale datasets: CIFAR10, MNIST, and NUS-WIDE. Our method outperforms the state-of-the-art hashing methods in retrieval performance while it has competitive execution time. The significance of our incremental strategy is observed when it is applied on dynamic datasets where new images are added and existing images are delete. Our incremental hashing strategy reaches the same retrieval performance as the from-scratch hashing strategy while requiring shorter training time.

Experiments validate that the incremental hashing strategy for dynamic datasets is capable of updating hash functions efficiently. Besides, the proposed approach provides higher quality codes with well-balanced bits and better generalization.