Journal of Machine Learning Research 6 (2005) 589–613 Submitted 11/04; Revised 3/05; Published 4/05

Active Learning to Recognize Multiple Types of Plankton

Tong Luo TLUO2@CSEE.USF.EDU
Kurt Kramer KKRAMER@CSEE.USF.EDU
Dmitry B. Goldgof GOLDGOF@CSEE.USF.EDU
Lawrence O. Hall HALL@CSEE.USF.EDU
Department of Computer Science and Engineering
University of South Florida
Tampa, FL 33620, USA

Scott Samson SAMSON@MARINE.USF.EDU
Andrew Remsen AREMSEN@MARINE.USF.EDU
Thomas Hopkins THOPKINS@MARINE.USF.EDU
College of Marine Science
University of South Florida
St. Petersburg, FL 33701, USA

Editor: David Cohn

Abstract

This paper presents an active learning method which reduces the labeling effort of domain experts in multi-class classification problems. Active learning is applied in conjunction with support vector machines to recognize underwater zooplankton from higher-resolution, new generation SIPPER II images. Most previous work on active learning with support vector machines only deals with two class problems. In this paper, we propose an active learning approach "breaking ties" for multi-class support vector machines using the one-vs-one approach with a probability approximation. Experimental results indicate that our approach often requires significantly less labeled images to reach a given accuracy than the approach of labeling the least certain test example and random sampling. It can also be applied in batch mode resulting in an accuracy comparable to labeling one image at a time and retraining.

Keywords: active learning, support vector machine, plankton recognition, probabilistic output, multi-class support vector machine

1. Introduction

Recently, an advanced shadow image particle profiling evaluation recorder (SIPPER II) was developed to produce 3-bit grayscale images at 25 μm resolution. SIPPER II uses high-speed digital line-scan cameras to continuously sample plankton and suspended particles in the ocean. The high sampling rate of SIPPER II requires the development of an automated plankton recognition system. For example, in a previous study using approximately 150,000 SIPPER images from a two hour sampling deployment it took over one month to manually classify the images (Remsen et al., 2004). Also, this automated system is expected to continuously evolve from an existing model to a more accurate model created by training after adding some new labeled images into the training set. Since it is impossible to manually label all images during the time they are acquired on the ship, active learning to label the *most important images* seems attractive.

For plankton recognition, Luo et al. (2004b) developed an automated system to recognize 1-bit binary SIPPER (SIPPER I) (Samson et al., 2001) images at 50 μm resolution. Due to the instability of contour features, Luo et al. (2004b) designed several image features which did not depend heavily on contours, and applied a support vector machine (SVM) (Vapnik, 2000) to classify the feature vectors. The wrapper approach was used to do feature selection effectively reducing the feature vector from 29 to 15 features. Also, a new way of computing probabilistic output in a multi-class support vector machine was developed. Tang et al. (forthcoming) proposed several new features for the SIPPER I images and applied multilevel dominant eigenvector methods to select the best subset of features. A Gaussian classifier was employed to recognize the image features and validate the feature selection methods on selected identifiable plankton.

Recently, active learning with SVMs has been developed and applied in a variety of applications (Tong and Koller, 2000; Schohn and Cohn, 2000; Campbell et al., 2000; Sassano, 2002; Warmuth et al., 2003; Brinker, 2003; Wang et al., 2003; Onoda et al., 2003; Baram et al., 2004; Luo et al., 2004a; Nguyen and Smeulders, 2004; Park, 2004; Mitra et al., 2004a,b). The most representative and relevant work is reviewed in the following.

A similar active learning method for support vector machines (SVMs) in two class problems was independently developed by several researchers Tong and Koller (2000), Schohn and Cohn (2000), and Campbell et al. (2000). These approaches, which we term "simple", caused the new examples closest to the decision boundary to be labeled. Tong and Koller (2000) used version spaces to analyze the hypotheses space of SVMs. It was shown that "simple" approximately found the examples which most dramatically reduced the version space. Compared to random sampling, "simple" reduced the required number of labeled images in experiments on text classification. Mitra et al. (2004a) argued that the greedy search method employed in "simple" is not robust and a confidence factor was proposed to measure the closeness of the current SVM to the optimal SVM. A random sampling factor was introduced when the confidence factor was low. Their proposed method performed better than "simple" in a set of experiments.

Roy and McCallum (2001) used a probability model to label examples which could maximize the posterior entropy on the unlabeled data set. We call this method "conf" in this paper. The "conf" method amounts to improving the current classifier's classification confidence on the unlabeled data set. Although it initially was applied with naive bayes classifiers, it could be easily extended to any classifier with probability outputs. For example, the probability outputs of SVMs can be roughly approximated by a sigmoid function (Platt, 2000).

Baram et al. (2004) observed that there was no single winner from different active learning strategies on several data sets. They proposed dynamically selecting from four learning algorithms: "simple", "conf", random sampling and sampling examples furthest from the current labeled data set. The automatic selection was done by solving a multi-armed bandit problem through online learning.

Similar selection methods to label several examples at a time for two-class problems were developed by Brinker (2003) and Park (2004) and named "combined" by Brinker (2003). Based on the "simple" method, they chose to label examples which are close to the decision boundary and have the largest angles to previously selected candidates. A parameter λ was introduced to control the trade-off between the two criteria. Although Brinker (2003) did not provide a method to set the optimal value of λ, "combined" performed better than "simple" in batch mode on several data sets (when labeling several images at a time).

There are two elements of our work which differentiate it from previous approaches. The images sampled from first generation SIPPER (SIPPER I) did not have clear contours. The low image quality resulted in many unidentifiable particles, which made it important to create robust image features and handle unidentifiable particles (Luo et al., 2004b). Higher resolution SIPPER (SIPPER II) images provide relatively better quality images with clear contours. Also, 3-bit graylevel images have more texture information than binary images. As a result, handling many unidentifiable particles is no longer an issue. The higher resolution required new contour and texture features to improve recognition. Moreover, little previous work in active learning has been done with multiple class SVMs, which is required in plankton recognition. SVMs solve multiple class problems by building several two-class SVMs and a new example usually has different distances to the decision boundaries in those two-class SVMs. It is hard to use the "simple" approach because we do not know which distance to choose. In a very recent paper Mitra et al. (2004b) applied "simple" to each binary SVM in a multi-class SVM. For a multi-class problem with N binary SVMs, N examples were labeled at a time. However, this method is far from elegant. They did not suggest how to choose which example was best for all binary SVMs. It is not unusual that an "informative" example for one binary SVM is useless for other binary SVMs. The "combined" method suffers from the same problem. It is not clear which distance to minimize and which angle to maximize. The "conf" approach seems to be a natural solution for multi-class problems as long as there is a probability estimation for the output from a multi-class SVM. However, applying the "conf" approach involves estimating the decision boundary after adding each unlabeled example into the training data in each round. Suppose m is the number of unlabeled examples and c is the number of classes, "conf" needs to train a SVM cm times to decide which example to label next. Although there are several heuristics to speedup such a procedure, it remains quite computationally expensive.

A new image feature set was developed Luo et al. (2004a) which added some contour features and texture features into a previous feature set (Luo et al., 2004b). A least certainty active learning approach was proposed and evaluated for multiple class SVMs. In this paper we expand the work reported by Luo et al. (2004a) and propose a new active learning strategy for one-versus-one multi-class SVMs. After developing a probability model for multiple class SVMs, we label the example which has the smallest difference in probability between its most likely class and second most likely class. We compare our approach with other methods like random sampling and least certainty for the plankton recognition problem. To obtain the same classification accuracy, we show our approach required many fewer labeled examples than random sampling. It also outperformed the least certainty approach in terms of needed examples to reach a given accuracy level. Our proposed method can run in batch mode, labeling up to 20 images at a time, with an accuracy comparable to labeling one image at a time and retraining. In a simulation where plankton images come as a stream, active learning resulted in higher classification accuracy than random sampling.

This paper is organized as follows. Section 2 introduces our active learning approach for support vector machines and our approach to assigning a classification probability for a multi-class support vector machine. Experimental results are presented in Section 3. Finally we summarize our work and propose some ideas for future work in Section 4.

2. Active Learning Approach with Multi-Class Support Vector Machines

A soft margin support vector machine was used in this work. A probability model has been added to the support vector machine to help evaluate multi-class decision problems. The probability model was used in the development of an active learning model.

2.1 Support Vector Machines

Support vector machines (SVMs) (Vapnik, 2000) have received increasing attention recently and have been shown to have very good accuracy for pattern recognition, text classification, etc. (Cristianini and Shawe-Taylor, 2000).

SVMs first map the data into a higher dimension feature space with $\phi(x)$, then use a hyperplane in that feature space to separate the data into two classes. In the feature mapping stage, the kernel $k(x,y) = \langle \phi(x) \cdot \phi(y) \rangle$ is used to avoid explicit inner product calculation in the high-dimensional feature space. C-SVM (Vapnik, 2000), a typical example of soft margin SVMs, is described in the following.

Given m examples: $x_1, x_2, ..., x_m$ with class label $y_i \in$ {-1,1}.

C-SVM:

$$\text{minimize } (\frac{1}{2}\langle w, w \rangle + C \sum_{i=1}^{m} \xi_i) \tag{1}$$

$$\text{subject to: } y_i(\langle w, \phi(x_i) \rangle + b) \geq 1 - \xi_i, \tag{2}$$

where w is normal to the class separating hyperplane, C is a scalar value that controls the trade off between the empirical risk and the margin ($\frac{2}{|w|}$) , ξ_i is the slack variable to handle non-separable examples, b is a scalar value, and $C, \xi_i > 0$.

With Lagrange multipliers, the constraint optimization problem in Eq. (1) and (2) can be solved. The decision function is

$$f(x) = \sum_i \alpha_i y_i k(x_i, x) + b,$$

where α_i is a Lagrange multiplier. Both α_i and b are scalar values.

The Karush-Kuhn-Tucker condition of the optimal solution to Eq. (1) and (2) is

$$\alpha_i(y_i(\langle w, \phi(x_i) \rangle + b) - 1 + \xi_i) = 0.$$

α_i is nonzero only when

$$y_i(\langle w, \phi(x_i) \rangle + b) = 1 - \xi_i. \tag{3}$$

In this case the x_i contributes to the decision function and is called a support vector (SV).

We applied the one-vs-one approach to extend SVMs to multiple class problems. All possible groups of 2 classes were used in building binary SVMs. In the N class case, we will build $\frac{N(N-1)}{2}$ binary SVMs. We chose the one-vs-one method because it showed superior accuracy in several experiments (see Hsu and Lin, 2002) over other multi-class methods–one-vs-all (Vapnik, 2000) and the decision directed acyclic graph (Platt et al., 2000).

2.2 Assigning Probability Values in Support Vector Machines

A probability associated with a classifier is often very useful and it provides an indication of how much to believe the classification result. The classification probability can be used to develop an active learning strategy for a multi-class SVM.

In Platt (2000) the sigmoid function was introduced as a probability model to fit $P(y=1|f)$ directly, where f is the decision function of the binary SVM. The parametric model is shown in Eq. (4).

$$P(y=1|f) = \frac{1}{1+exp(Af+B)}, \tag{4}$$

where A and B are scalar values, which are fit by maximum likelihood estimation.

A method to estimate classification probability for a series of pairwise classifiers was proposed by Hastie and Tibshirani (1998). Given the estimated probability for each binary classifier(P_{pq}), the probability of being class p in a binary classifier (class p vs. class q), they minimized the average Kullback-Leibler distance between P_{pq} and $\frac{P(p)}{P(p)+P(q)}$, where $P(p)$ and $P(q)$ were the probabilities of a given example belonging to classes p and q, respectively. An iterated algorithm was given to search for $P(p)$. Following this line of the work, Wu et al. (2004) developed two new criteria for the goodness of the estimated probabilities and applied their method to multi-class SVMs. Their approach has three steps to get the probability estimation. First, a grid-search is used to determine the best SVM parameters (C, g) based on a k-fold cross validation accuracy, where C is the regularization constant in Eq. (1) and g is the kernel parameter in the kernel function k. Second, with the optimal (C, g) found in the first step, A and B were fit individually for each binary SVM. Third, a constrained quadratic programming method was used to optimize the criteria they proposed.

However, this approach is time consuming. The second step involves estimating $N(N-1)$ parameters for SVMs using a one-vs-one approach. The third step requires quadratic programming to solve N variables for each example. On a data set with m examples, this step needs to run m times. Another issue was that the SVM parameters (C, g) were estimated based on accuracy and thus might not be good for probability estimation in the following two steps.

In real-time plankton recognition, the probability computation needs to be fast since retraining the probability model will be frequently needed as more plankton images are acquired on a cruise.

We (Luo et al., 2004b) developed a practical approximation method to compute the probability value, while avoiding expensive parameter fitting. By normalizing the real valued output $f(x)$ from each binary SVM, the probability model assumes the same A for all binary SVMs. Also, our approach can optimize SVM parameters (C, g) together with the probability parameter A simultaneously using a log-likelihood criterion.

1. We assume $P(y=1|f=0) = P(y=-1|f=0) = 0.5$. This means that a point lying on the decision boundary will have a 0.5 probability of belonging to each class. This allows the elimination of B.

2. Since each binary SVM has a different margin, a crucial criterion in assigning the probability, it is not fair to assign a probability without considering the margin. Therefore, the decision function $f(x)$ is normalized by its margin in each binary SVM. The probability model of SVMs is shown in (5) and (6). P_{pq} represents the probability output for the binary SVM on class p vs. class q, class p is +1 and class q is -1. We added the negative sign before A to ensure A is positive:

$$P_{pq}(y=1|f)=\frac{1}{1+exp(\frac{-Af}{\|w\|})}, \tag{5}$$

$$P_{pq}(y=-1|f)=1-P_{pq}(y=1|f)=P_{qp}(y=1|f). \tag{6}$$

3. Assuming $P_{pq}, q=1,2,...$ are independent, the final probability for class p is computed as follows:

$$P(p)=\prod_{q}^{q\neq p} P_{pq}(y=1|f). \tag{7}$$

   Normalize $P(p)$ to make $\sum_p P(p)=1$.

4. Output $\hat{y}=arg\ max_p P(p)$ as the prediction.

(A, C, g) are determined through numeric search based on the cost function L from (8), where t_i is the true class label of x_i:

$$L=-\sum_i logP(t_i). \tag{8}$$

Although it is arguable whether P_{pq} and P_{pk} are really independent since P_{pq} and P_{pk} are both estimated using data from class p, the one-vs-one approach does not suffer much from any dependence. Consider differentiating examples from three classes (p, q and k). If a classifier is built for classes p and q with another built for p and k, there is clearly a relationship but one class is different. So, following this type of argument the classifiers will have a weak dependence. Knowing there is only a weak dependence between P_{pq} and P_{pk}, Eq. (7) provides a reasonable approximation.

Grid-search can be used to find the optimal (C, g, A) on the initial small labeled data set. It has the potential to be run in parallel to significantly reduce the computation time. If we want to update the probability model after adding more labeled images, we can fix C and g, and only search for A. As a result, it is very fast to update the probability model. Moreover, we directly optimize (C, g, A) together by minimizing the negative log-likelihood function in Eq. (8). Normalizing f by its margin and assuming the same A for each binary SVM trades off some flexibility to gain a regularization effect and speedup since it restricts the otherwise big ($N(N+1)$) parameter space. Experiments for this probability model were done on SIPPER images by Luo et al. (2004b).

2.3 Active Learning for Multi-Class SVMs

The least certainty active learning approach for SVMs (Luo et al., 2004a), which makes use of the estimated probability described in Section 2.2, provides good performance in multi-class SVM classification. The idea for it can be traced back to Lewis and Gale (1994), who used "uncertainty sampling" to label the examples with the least classification certainty. We call the least certainty approach for SVMs by Luo et al. (2004a) "LC". In this paper, we propose another active learning approach–"breaking ties" (BT). The idea of "BT" is to improve the confidence of the multi-class classification. Recall in a multi-class SVM with probability outputs, we assign the class label of x to $argmax_p P(p)$. Suppose $P(a)$ is the largest and $P(b)$ is the second largest probability for example x,

where a, b are class labels. "BT" tries to improve the $P(a) - P(b)$. Intuitively, improving the value of $P(a) - P(b)$ amounts to breaking the tie between $P(a)$ and $P(b)$, thus improving the classification confidence. The difference between "LC" and "BT" is that "LC" tries to improve the value of $P(a)$ rather than $P(a) - P(b)$.

The two algorithms work as follows:

1. Start with an initial training set and an unclassified set of images.
2. A multi-class support vector machine is built using the current training set.
3. Compute the probabilistic outputs of the classification results for each image on the unclassified set. Suppose the class with highest probability is a and the class with second highest probability is b. Record the value of $P(a)$ and $P(b)$ for each unclassified image.
4. If LC: Remove the image(s) from the unclassified set that have the smallest classification confidence, obtain the correct label(s) from human experts and add the labeled image(s) to the current training set.
5. If BT: Remove the image(s) from the unclassified set that have the smallest difference in probabilities between them ($P(a) - P(b)$) for the two highest probability classes, obtain the correct label from human experts and add the labeled image(s) to the current training set.
6. Go to 2.

3. Experiments

The experimental data set consisted of 8440 SIPPER II images selected from the five most abundant types of plankton: 1688 images from each type of plankton. There were 1000 images (200 each type of plankton) randomly selected as the validation set used in the active learning experiments. The remaining 7440 image were used as the training set and to simulate the unlabeled image pool. Figures 1(a) to 1(e) are typical examples of the images produced by SIPPER II for the five most abundant plankton classes.

Given this new higher resolution data, 49 image features were developed (Luo et al., 2004b; Luo, forthcoming) consisting of: moment invariants, weighted moment invariants, granulometric features, Fourier descriptor, texture features and several domain specific features.

The Libsvm (Chang and Lin, 2001) support vector machine software was modified to produce probabilistic outputs. Rifkin and Klautau (2004) argued the one-vs-all approach was essentially as good as other voting algorithms, however, without postprocessing binary SVMs, we observed the one-vs-one approach provided better accuracy and required less training time than the one-vs-all approach in our previous experiments (Luo et al., 2004b). Also, when updating models with several more labeled examples in N class problems, the one-vs-one approach only requires the update of N binary SVMs built with a portion of the data, while the one-vs-all approach requires the update of N binary SVMs built with all the labeled data. Therefore, the one-vs-one approach was used in our experiments. In all experiments the Gaussian radial basis function (RBF) was used as the kernel: $k(x,y) = exp(-g\|x-y\|^2)$ where g is a scalar value.

The optimal feature subset was determined beforehand by our wrapper based feature selection method (Luo et al., 2004b) after the best (g, C) parameters were found by 5-fold cross validation.

(a) Calanoid copepod

(b) Larvacean

(c) Marine snow

(d) Oithona copepod

(e) Trichodesmium

Figure 1: Five most abundant types of plankton from SIPPER II

We started off with all 49 image features and systematically eliminated features using best first search and later beam search. Five-fold cross validation on 80% of the training data was used to select the best feature subset for each feature set size. Then the best feature subsets were tested on the remaining 20% of the training set. This feature selection method is described in detail by Luo et al. (2004b). As a result, 17 out of 49 features were selected. In all the active learning experiments, we used the best 17 feature subset instead of the 49 feature set.

In the kind of problem embodied by plankton recognition, there is only a small amount of initial training data available. Therefore, the best parameter set for the probability model would be estimated from a small data set. The parameters (g, C, A) were optimized by performing a grid-search across a randomly selected 1000 images consisting of 200 images per class. We believe the parameters were obtained from a relatively small set of data and were reasonably stable. A five-fold cross validation was used to evaluate each combination of parameters based on the loss function L from (8). The parameters (g, C, A) were varied with a certain interval in the grid space. Since the parameters are independent, the grid-search ran very fast in a parallel implementation. The values of $g = 0.04096$, $C = 16$, and $A = 100$ were found to produce the best results.

We began with N randomly selected images per class as the initial training set. A series of retrainings were done for the two active learning methods and with random sampling. Each experiment was performed 30 times and the average statistics were recorded. Instead of exhausting all of the unlabeled data set, we only labeled 750 more images in each experiment because exhausting all unlabeled data was not a fair criterion for comparing between different sample selection algorithms. For example, active learning labeled the most "informative" new examples, which were available in the beginning of the experiment. As more "informative" examples were labeled, only "garbage" examples were left unlabeled in the late stages of the experiment. The term "garbage" examples means the examples correctly classified by the current classifier and far from the decision boundary. Therefore, "garbage" examples have no contribution to improving the current classifier. In contrast to active learning, random sampling labeled average "informative" examples throughout the whole experiment. It surely would catch up with active learning in the later stages when active learning only had "garbage" examples to label. Moreover, when the plankton recognition system is employed on a cruise, the unlabeled images come like a stream. The nature of such an application prevents one from exhausting all the unlabeled images because the time required to label them is prohibitive. Therefore, it made more sense to compare different algorithms in the early stage of the experiment when the unlabeled data set is not exhausted. To get an idea of the upper limit on the classification accuracy, we built a SVM using all 7400 training images. Its prediction accuracy was 88.3% on the 1000 held-out data set.

Several variations of the procedure described above were performed. We varied both the number of initial labeled images per class (IIPC) and the number of images selected for labeling at each retraining step (IPR).

3.1 Experiments with IPR=1, IIPC Varied

Figures 2–5 show the experimental results of active learning methods using different IIPC values. A paired-t test was used to test if there exists a statistically significantly difference. We used standard error for the error bars in the figures because the denominator of t test is in the form of standard error.

As shown in Figure 2, with only 10 images per class in the initial training sets we started off with rather poor accuracy (64.6%). At p=0.05, "BT" is statistically significantly more accurate than "LC" and both active learning methods are statistically significantly more accurate than random sampling. At 81% accuracy, random selection required approximately 1.7 times the number of newly labeled images compared to "BT".

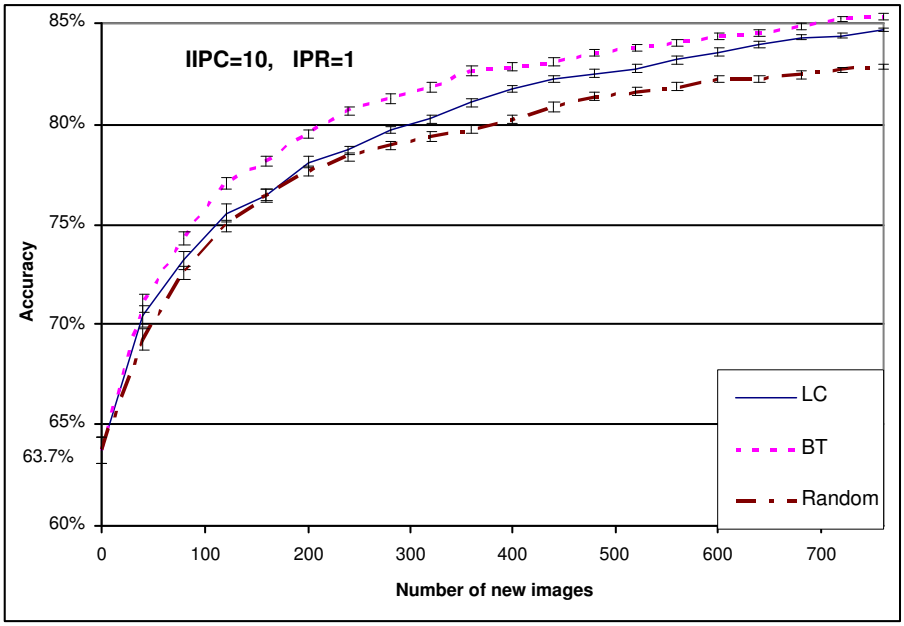
Active learning is designed to label the most "informative" new images, thus improving a newly trained classifier. In SVMs, the decision boundary is represented by support vectors (SVs). In general, an effective active learning approach finds more SVs than random sampling. Figure 2 also shows the average number of SVs versus the number of images added into the initial training set from the 30 runs. Active learning resulted in many more SVs than random sampling. Also, the slope of both active learning curves is about 0.9, which means that 90% of the labeled images turn out to be SVs. Our active learning approach efficiently captured support vectors. We note that a high slope of the support vector curve is not a sufficient condition for effective active learning because there are many SVs to be added into the current model and different SVs lead to different improvements. Ideally, a very effective active learning method discovers the SVs which provide the most improvement to the current model. In contrast, an active learning method, which always finds the SVs misclassified by the current classifier and far from its decision boundary, may result in poor performance because such SVs are very likely to be noise. Therefore, we cannot compare active learning methods based only on slight differences in the support vector curves.

With 50 IIPC in the initial training set as shown in Figure 3, the initial accuracy was 77%. Compared to 10 IIPC, the accuracy for both active learning approaches improved faster than random sampling. At the 81% accuracy level, random sampling required about 2.5 times and 1.7 times the number of images compared with using "BT" and "LC", respectively. The slopes of support vector curves for active learning are higher than those of random sampling. Also, "BT" again outperformed "LC", however, it is not as obvious as with IIPC=10.

In Figures 4 and 5, the initial accuracy was greater than 80% when using 100 and 200 initial images from each class, and active learning was very effective. Random sampling required more than 3 times the number of images to reach the same level of accuracy as both active learning approaches. The two active learning methods effectively capture many more SVs than random sampling. Also, our newly proposed active learning approach, "BT", requires less images to reach a given accuracy than "LC" after adding 450 labeled images. Before adding 450 labeled images, however, "BT" performs similarly to "LC".

It seems reasonable that the accuracy of the initial classifier affects the performance of active learning and random sampling. Active learning greedily chooses the most "informative" examples based on the previous model. So an un-informed model tends to be provide less important examples for labeling. Hence, their addition may not help to improve the classifier accuracy much. While random sampling provides the classifier with average "informative" examples whatever the initial classifier accuracy. Therefore, if the initial classifier helps active learning to choose examples more informative than average (random sampling), active learning will result in a more accurate classifier with fewer labeled examples. The better the initial classifier, the more labeling effort is saved.

When comparing the two active learning methods, "BT" outperformed "LC" under all four starting conditions. However, the difference in accuracy between them was insignificant as the initial classifier became more accurate. The justification is that an accurate initial classifier allows for less error reduction using active learning. "BT" improved the accuracy by more than 20% when IIPC=10 while it boosted the accuracy by less than 4% when IIPC=200. Therefore, as the amount



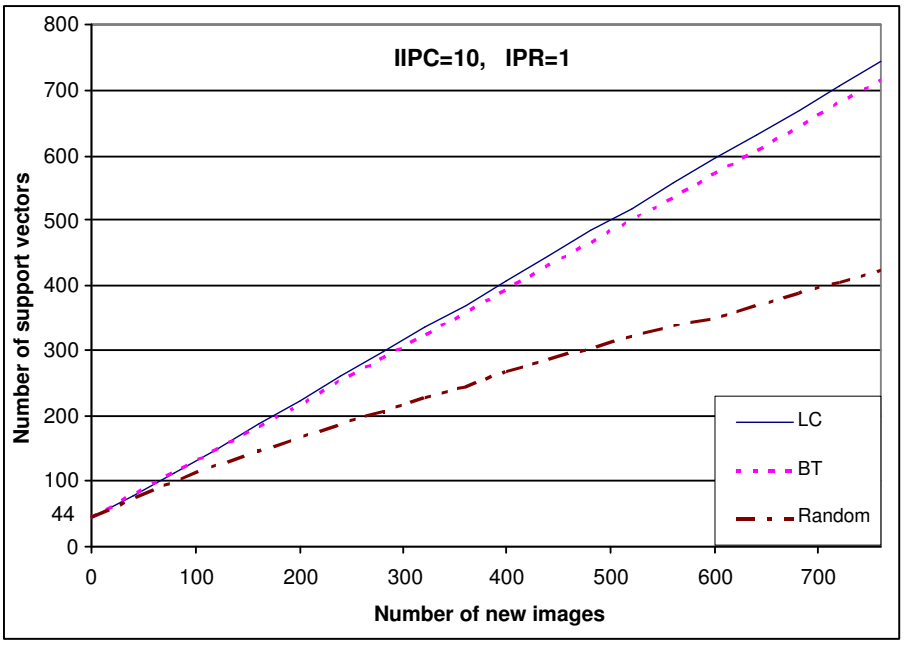


Figure 2: Comparison of active learning and random sampling in terms of accuracy and number of support vectors: initial training images per class are 10, one new labeled image added at a time. The error bars represent the standard errors.

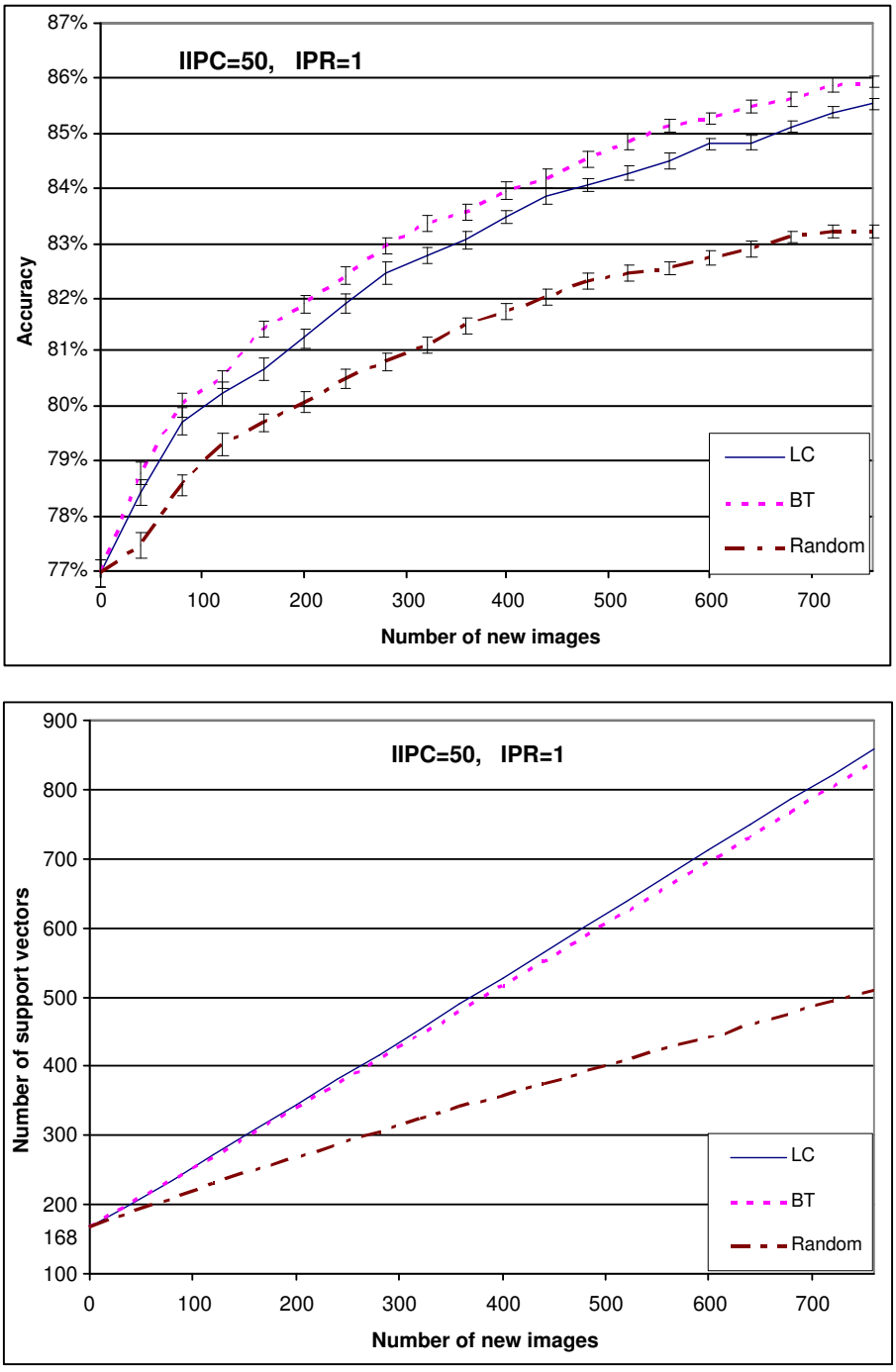


Figure 3: Comparison of active learning and random sampling in terms of accuracy and number of support vectors: initial training images per class are 50, one new labeled image added at a time. The error bars represent the standard error.

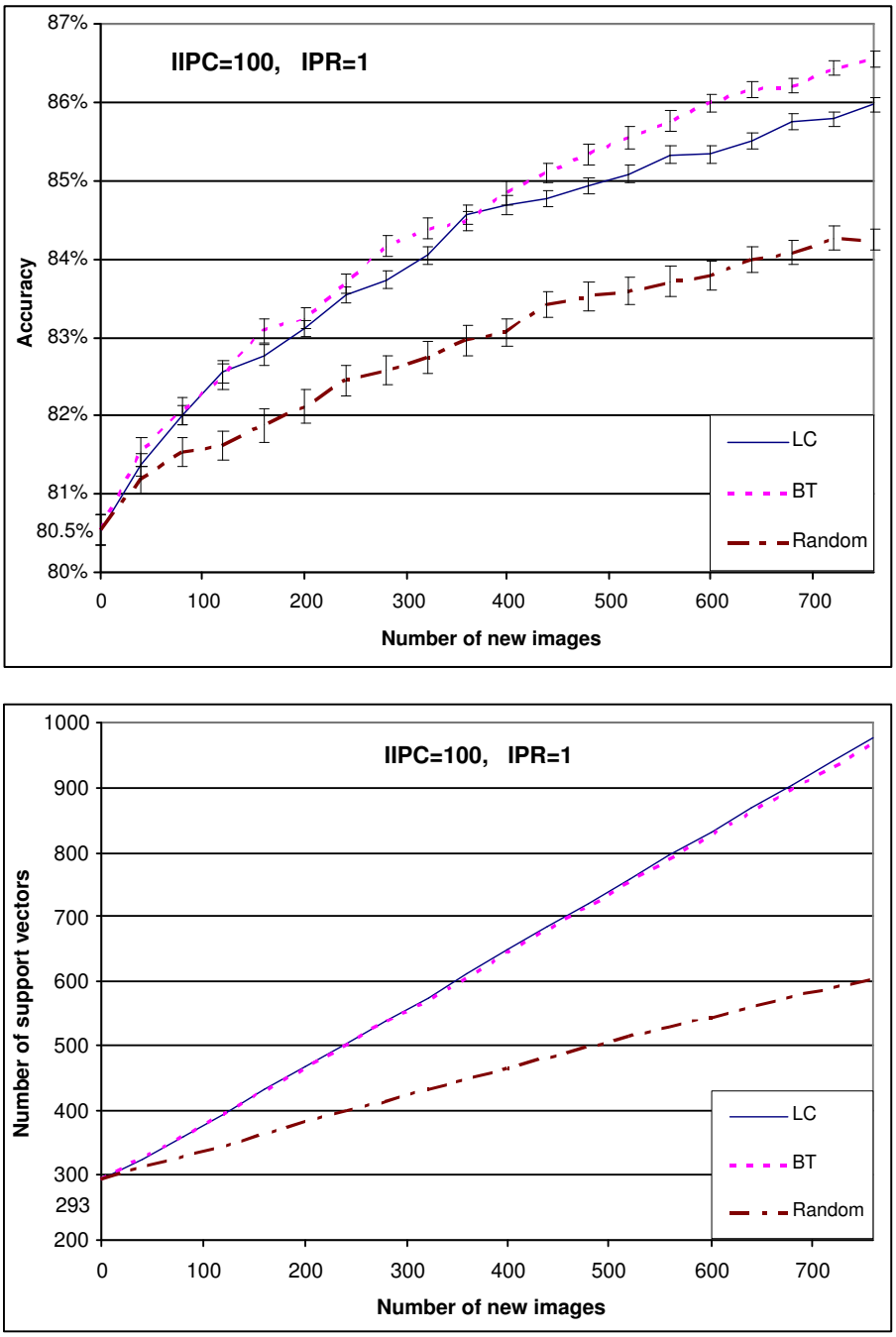


Figure 4: Comparison of active learning and random sampling in terms of accuracy and number of support vectors: initial training images per class are 100, one new labeled image added at a time. The error bars represent the standard error.

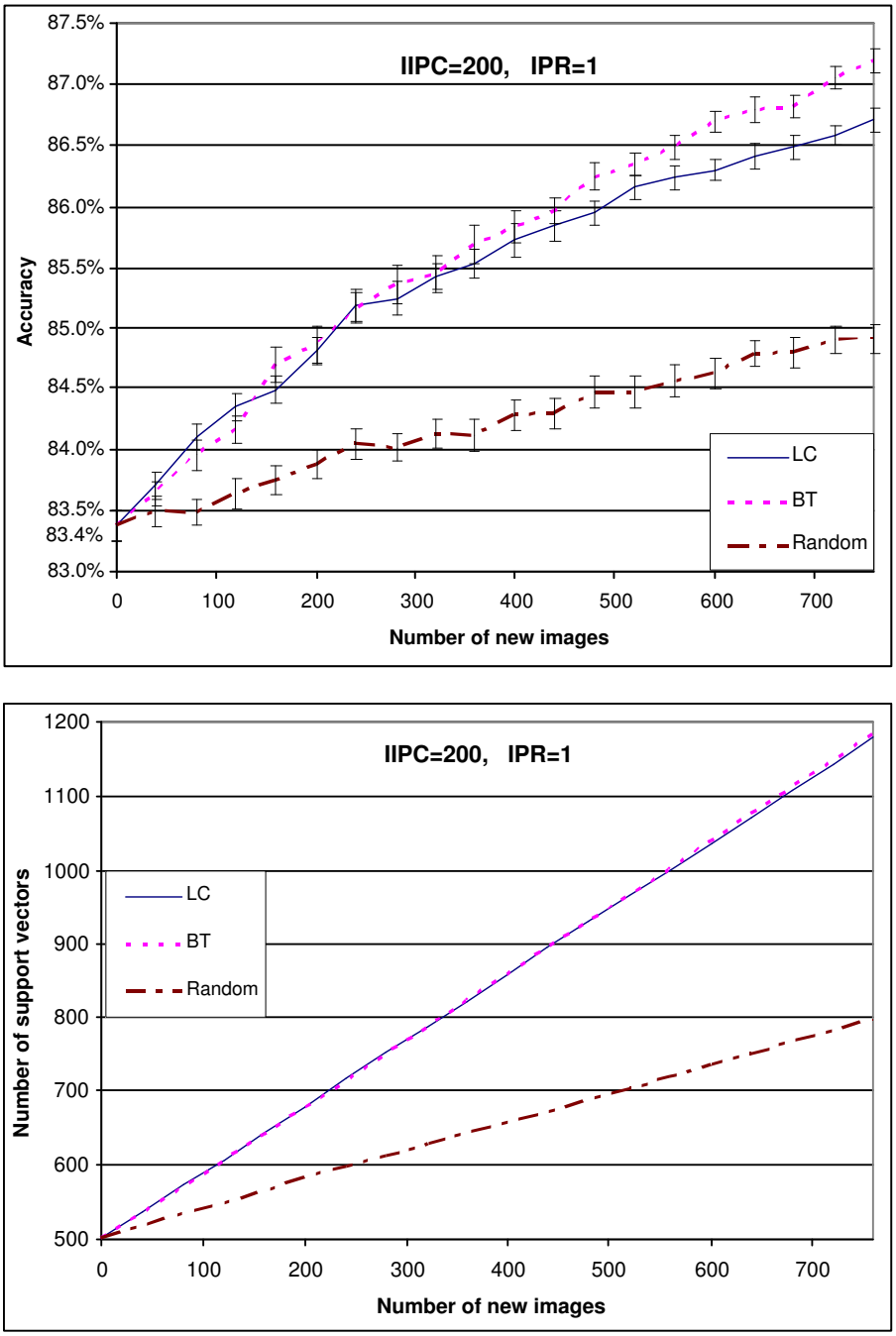


Figure 5: Comparison of active learning and random sampling in terms of accuracy and number of support vectors: initial training images per class are 200, one new labeled image added at a time. The error bars represent the standard error.

of available accuracy improvement was small, the difference in accuracy between the two active learning methods became insignificant.

3.2 Varying the IPR

One might expect that, in actual practice, more than one image would typically be labeled and added to the training set for retraining. It is convenient for an expert to label several images instead of one at a time. Also, given the total number of newly labeled images is U, it is approximately k times faster if we label k images at a time because it only requires a new model be learned $\frac{U}{k}$ times. Although an incremental SVM training algorithm was proposed by Cauwenberghs and Poggio (2000) to reduce the retraining time, model updating by labeling one image at a time was still quite time consuming, especially when many images are to be labeled. Therefore, we expected active learning to be effective even when adding several labeled images at a time.

The active learning method "BT" was good for adding only one "informative" example at a time, however there was no guarantee that adding several examples at a time would still favor "BT". The reason is that adding one "informative" example will update the model, which in turn changes the criterion for the next "informative" example. Therefore, the most "informative" example set is different from simply grouping several most "informative" examples together. However, such an optimal example set is very hard to compute. Therefore, we expect grouping several most "informative" examples together is a reasonable approximation of the optimal example set, or at least is superior to randomly selecting several examples.

Figures 6 to 9 show the experimental results using "BT" by varying IPR for each IIPC. In all the experiments, the IPR was varied from 1 to 50. We only show the error bars for random sampling because adding error bars to "BT" will make the graph too busy. We again used a paired-t test to compare "BT" with random sampling. Somewhat surprisingly, classification accuracy with large IPRs is almost as good as with small IPRs although a very large IPR (IPR=50) resulted in slightly less accurate classifiers than a small IPR in many cases. In all situations, even a large IPR (up to 50) enabled "BT" to result in a statistically significantly more accurate classifier than random sampling at p=0.05. These results indicate that our active learning approach "BT" can run in batch mode, labeling tens of examples at a time, to achieve speedup with at most a little compromise in accuracy.

3.3 Other Experiments

We experimented with "BT" in a streaming data simulation. In this experiment, unlabeled data was treated as a stream with only a block of unlabeled data available at a given time. The algorithm selectively labeled data within this block. Then all unlabeled data in the block was discarded and a new data block was pulled from the stream. In our experiment, we started off with 10 labeled images randomly selected from each class (IIPC=10). The size of data block was 100. From each data block, 10 images were selected to label at a time (IPR=10). Figure 10 shows a comparison of "BT" with random sampling in the stream setup. We also show "BT" without streaming for comparison. All the curves were averaged over 30 runs in which the order of the data blocks was randomized.

"BT" in a streaming simulation performed very well. At p=0.05, it was more accurate than random sampling and was as accurate as "BT" in a non-streaming setup. This experiment indicates that "BT" works well in "data streaming" situations.

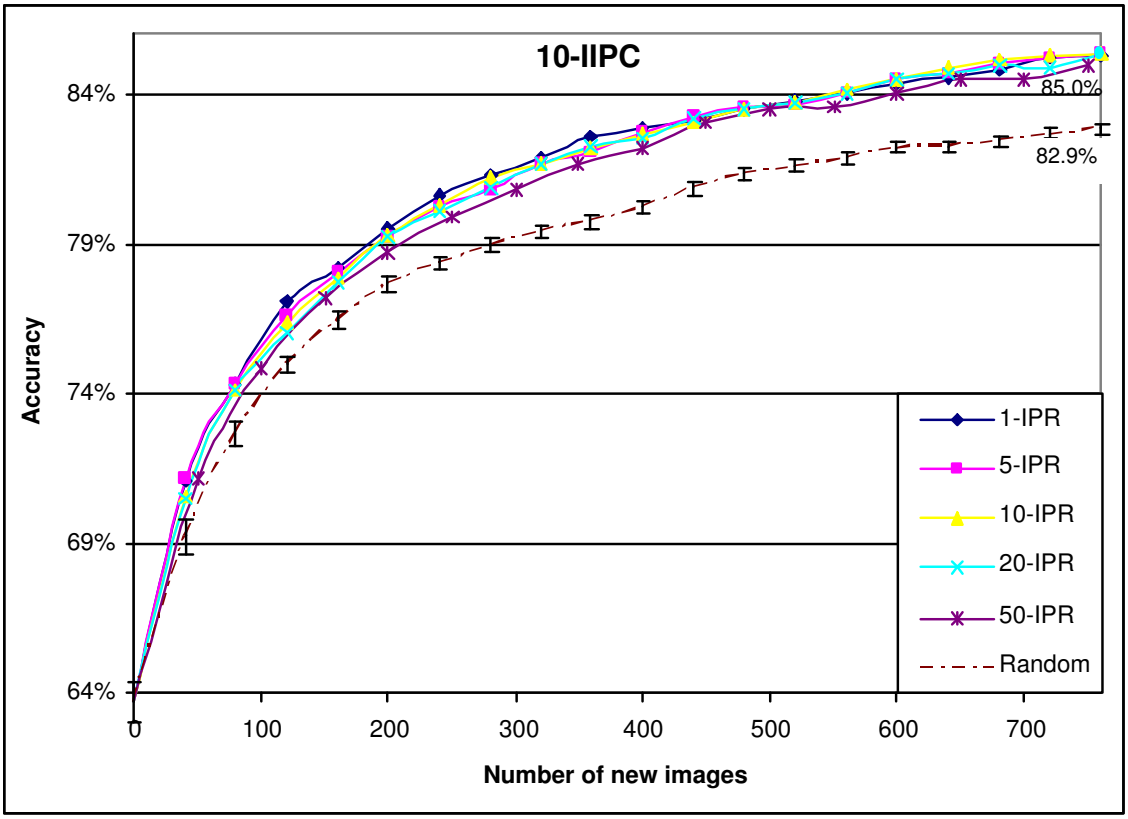


Figure 6: Comparison of active learning and random sampling in terms of accuracy with different IPR: initial training images per class are 10. Standard error bars are on the random sampling curve.

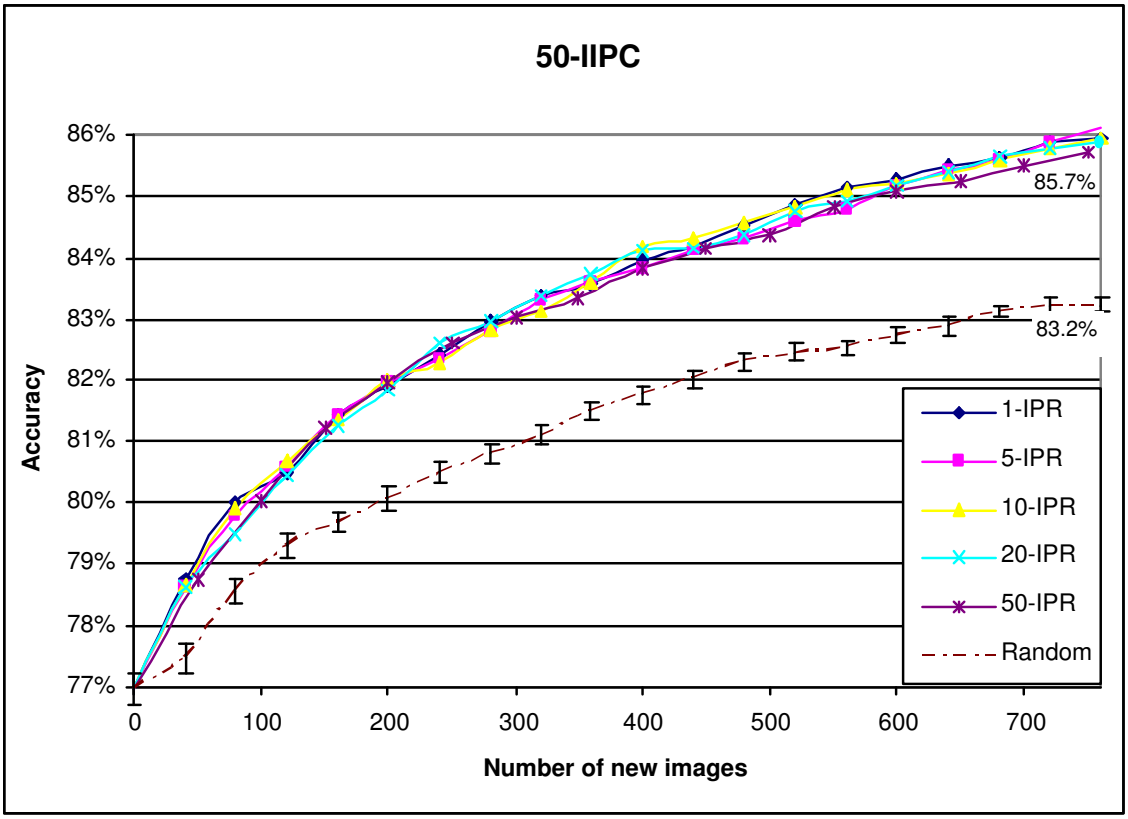


Figure 7: Comparison of active learning and random sampling in terms of accuracy with different IPR: initial training images per class are 50. Standard error bars are on the random sampling curve.

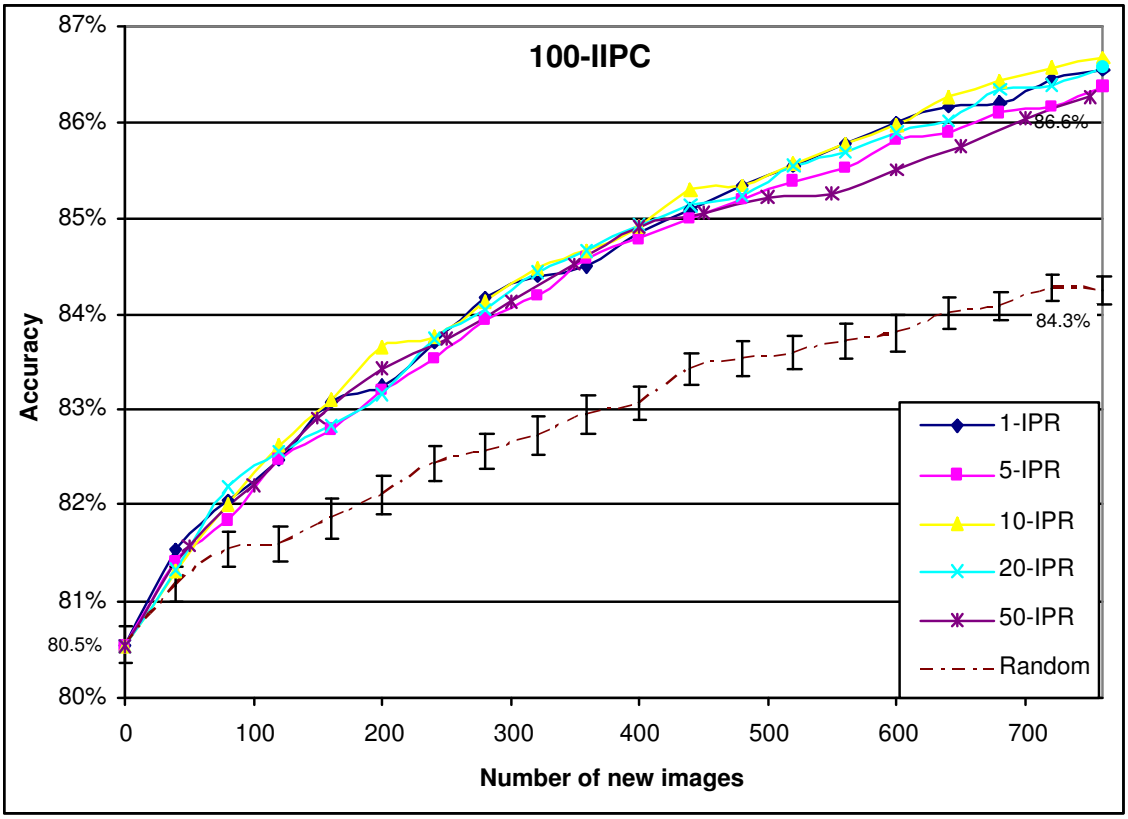


Figure 8: Comparison of active learning and random sampling in terms of accuracy with different IPR: initial training images per class are 100. Standard error bars are on the random sampling curve.

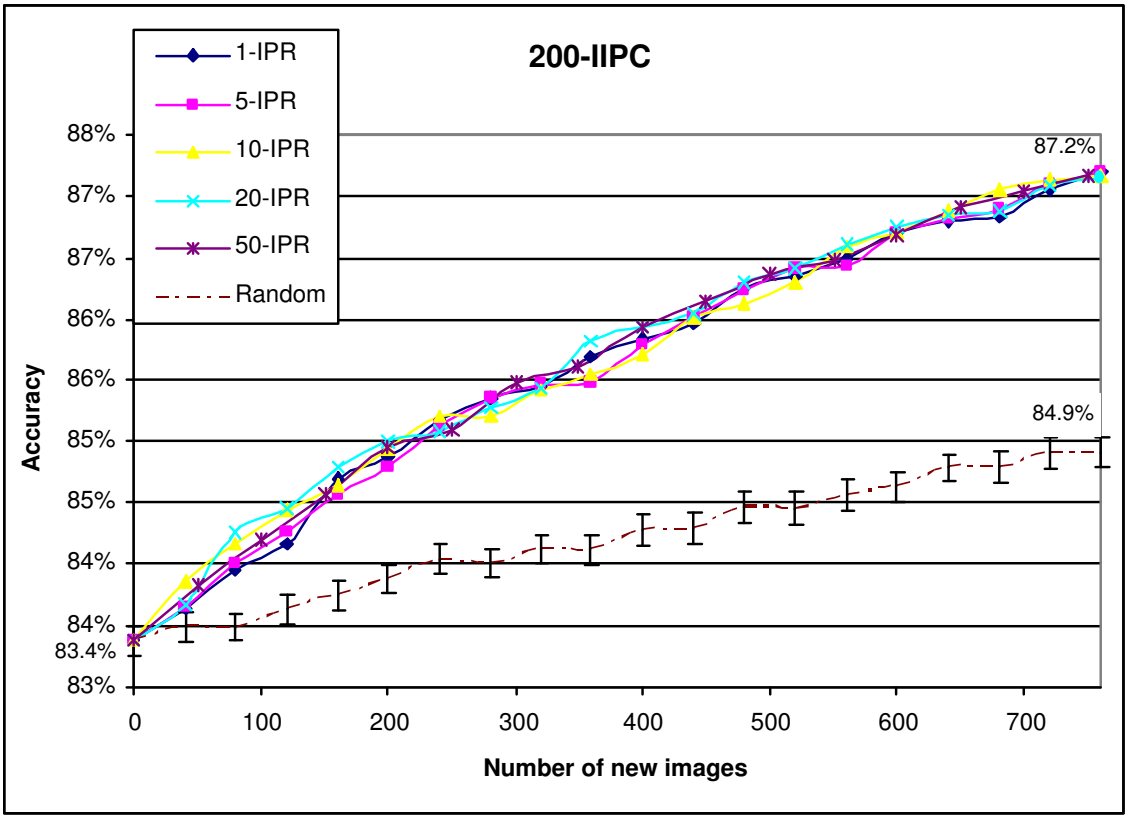


Figure 9: Comparison of active learning and random sampling in terms of accuracy with different IPR: initial training images per class are 200. Standard error bars are on the random sampling curve.

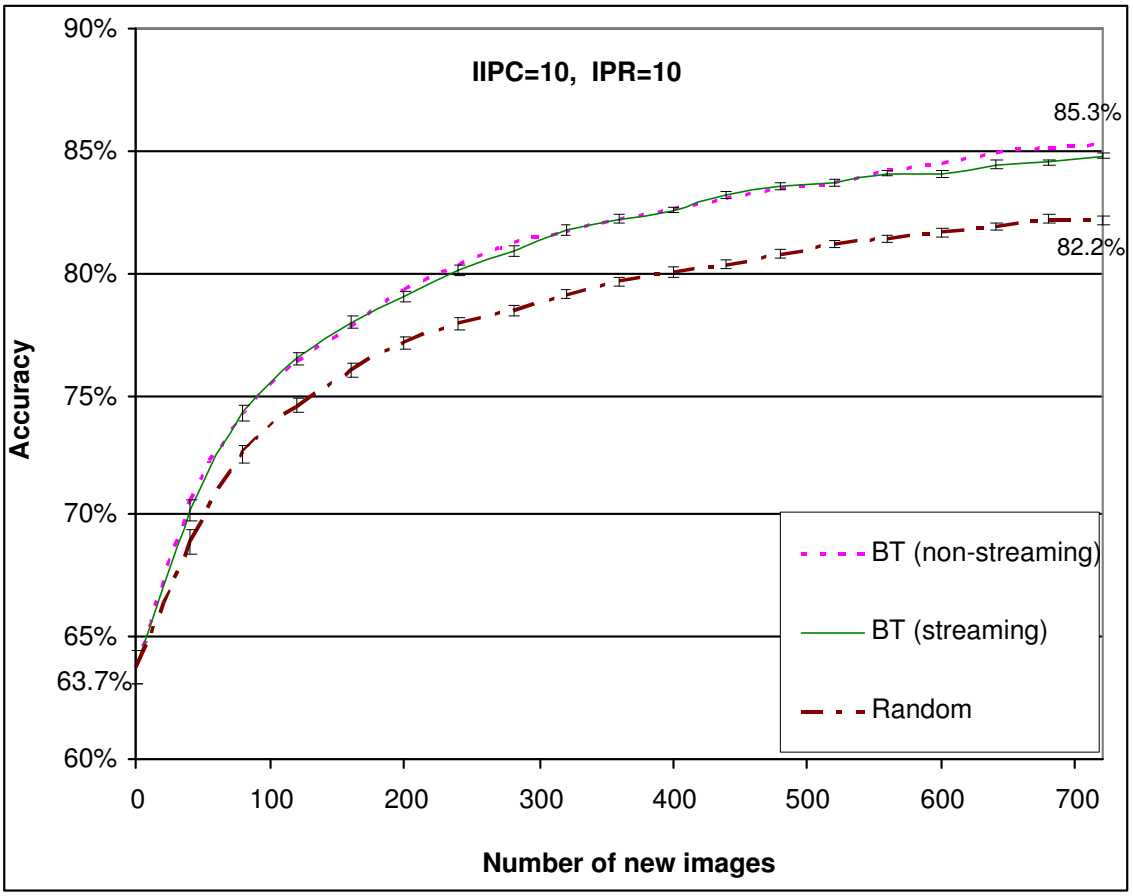


Figure 10: Comparison of active learning and random sampling in a data streaming simulation: initial training images per class are 10, 10 newly labeled images added at a time. The error bars represent the standard error.

In the previous experiments, all the classes have equal priors. We investigated how "BT" performed when the priors for each type of plankton were different. It should be noted that under the unequal prior condition, our probabilistic interpretation is different from traditional probability model in two aspects. First, the sigmoid function directly estimates $P(y|f)$ rather than computing the probability density function $P(f|y)$. Therefore, we can not simply apply Bayes rule to include the priors. Second, our probability model is built on top of a trained SVM. When the class distribution becomes skewed, the decision function of a SVM will vary accordingly. As a result, our probability model (built on the SVM) implicitly incorporates the unequal prior information. In our experiment, we selected three types of plankton with unequal priors. The ratio among the three types of plankton was 1:2:4. Both the unlabeled pool and the held-out data set had 200 larvacean, 400 oithona and 800 copepod images. We started off with a total of 30 initial labeled images randomly taken from the above distribution and labeled one image at a time per retraining (IPR=1). The initial 30 labeled image set consisted of 4 larvacean, 9 oithona, and 17 copepod images. The initial labeled images used unequal priors because they would typically be randomly sampled from the unlabeled pool and therefore would likely have the same class distribution.

Figure 11 shows the experimental results for the unequal prior experiment. When the distribution of different plankton was skewed, "BT" still outperformed random sampling. "BT" was statistically significantly more accurate than random sampling at p=0.05.

4. Discussion and Conclusions

This paper presents an active learning approach to reduce domain experts' labeling efforts in recognizing plankton from higher-resolution, new generation SIPPER II images. It can be applied to any data set where the examples will be labeled over time and one wants to use the learned model as early as possible. The "breaking ties" active learning method was proposed and applied to a multi-class SVM using the one-vs-one approach on newly developed, image features extracted from gray-scale SIPPER images. The experimental results showed that our proposed active learning approach successfully reduced the number of labeled images required to reach a given accuracy level when compared to random sampling. It also outperforms the least certainty approach previously proposed by us. The new approach was also effective in batch mode, allowing for labeling up to 20 images at a time with classification accuracy which was similar to that achieved when labeling one image at a time and retraining. This results in a significant speedup in the training phase. In the following, we address and discuss several active learning in SVM issues which deserve further exploration.

One critique of active learning is the overhead related to searching for the next candidate to label. Random sampling just selects an example to label at random, but active learning needs to evaluate every unlabeled example. This overhead becomes significant when the unlabeled data set is very large. A simple solution would be random subset evaluation. Each time one searches for the next candidate example to label, instead of evaluating the entire unlabeled data set, can only evaluate a randomly drawn subset. We indicate without proof here that for IPR=1, we needed to sample 59 examples, which provided 95% probability confidence that the best candidate from the 59 example subset is superior to 95% data from the total unlabeled set (see Schölkopf and Smola, 2002, chap. 6.5). Also, the experiment with a "data streaming" simulation indicated "BT" worked well when the evaluation and active learning was performed on a small subset of the data.

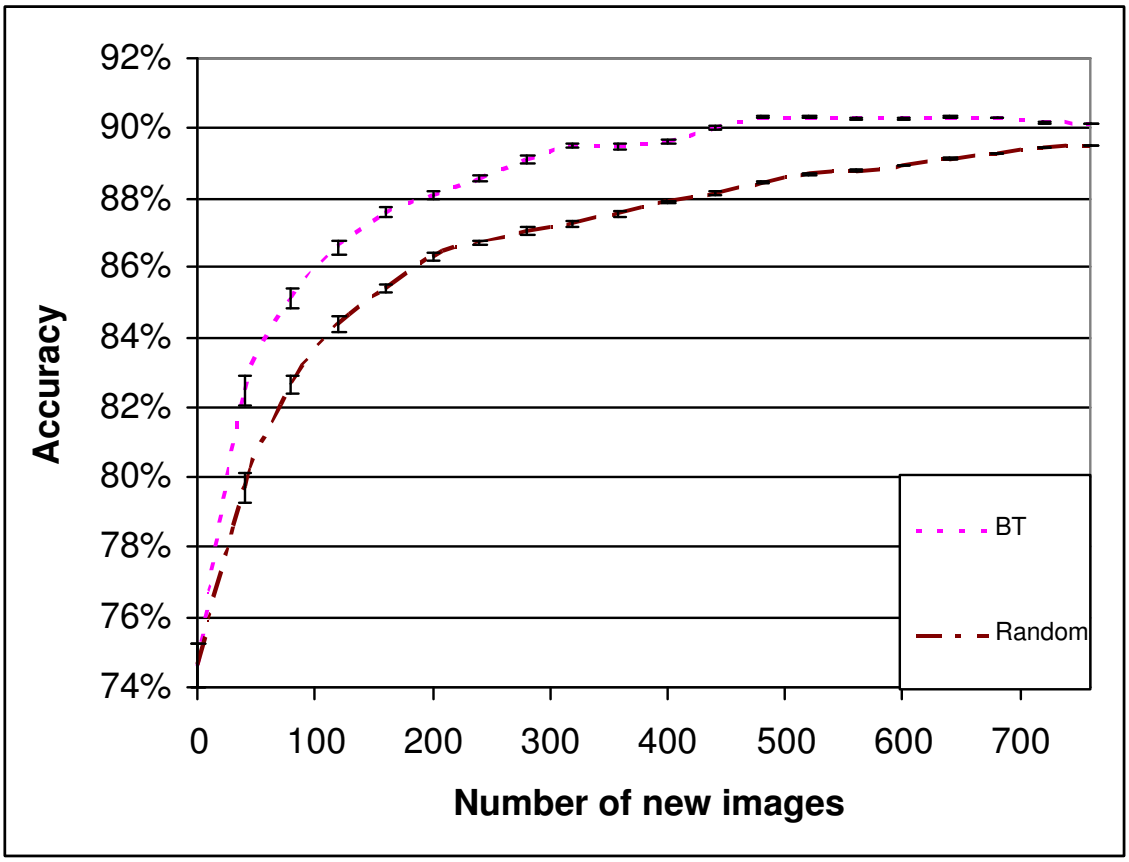


Figure 11: Unequal prior experiment. The ratio among the three classes are 1:2:4. The error bars represent the standard errors.

Another important issue is the change of optimal kernel parameters. We can find the optimal kernel parameters from the initial labeled data set. As more labeled data are added, however, such kernel parameters may no longer be optimal. Unless we can afford a held-out, labeled data set, it is difficult to tune the kernel parameters online. The key reason is we do not have a good method to evaluate different kernel parameters as active learning proceeds. The standard methods like cross-validation and leave-one-out tend to fail because active learning chooses biased data samples. Such failures were observed and discussed by Baram et al. (2004). An important future direction is to find a good online performance evaluation method for active learning. Otherwise, one could take it as one of the biggest bottlenecks for using a SVM as the classifier in active learning because a SVM depends heavily on good kernel parameters. An effort towards solving this problem was proposed by Baram et al. (2004) who used the classification entropy maximization (CEM) criterion to evaluate the performance of different active learners. Their work shows CEM can help select the best active learner on several data sets.

An important thing omitted in most active learning+SVMs literature is to try active learning in batch mode. Unless labeling an example is extremely expensive, it is always convenient and practical to use active learning in batch mode, namely labeling several examples at a time and then retraining. As indicated in this paper, the best candidate set to label might be found in a different way from a single best candidate point. The "combined" approach only works for two-class problems. A criterion for the best set of data to label in multi-class SVMs needs to be addressed in future active learning work. At the very least, existing active learning methods need to be shown to work well in batch mode. Fortunately, our proposed active learning method did work well in batch mode without requiring a new criterion for selecting a set of data to label.

In this paper, we do not deal with a significant amount of label noise, which means one assigns incorrect class labels to the examples. In general, active learning tries to minimize the redundancy of labeled examples to reach a given accuracy. Therefore, noisy labels will hurt its performance. In our case, we only selectively label a few images and we expect small labeling noise due to the relatively small labeling effort. Please see Kearns (1998) for more detail about handling noisy labels in statistical queries.

Acknowledgments

This research was partially supported by the United States Navy, Office of Naval Research, under grant number N00014-02-1-0266 and the NSF under grant EIA-0130768.

References

Y. Baram, R. Yaniv, and K. Luz. Online choice of active learning algorithms. *Journal of Machine Learning Research*, 5:255–291, 2004.

K. Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of the Twentieth International Conference on Machine Learning*, pages 59–66, 2003.

C. Campbell, N. Cristianini, and A. Smola. Query learning with large margin classifiers. In *Proceedings of 17th International Conference on Machine Learning*, pages 111–118, 2000.

G. Cauwenberghs and T. Poggio. Incremental and decremental support vector machine learning. In *Advances in Neural Information Processing Systems*, volume 13, pages 409–415, 2000.

C. Chang and C. Lin. LIBSVM: a library for support vector machines (version 2.3). 2001. URL `http://www.csie.ntu.edu.tw/ cjlin/papers/libsvm.pdf`.

N. Cristianini and J. Shawe-Taylor. *Introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000.

T. Hastie and R. Tibshirani. Classification by pairwise coupling. In *Advances in Neural Information Processing Systems*, volume 10, pages 507–513, 1998.

C. W. Hsu and C. J. Lin. A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425, 2002.

M. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6): 983–1006, 1998.

D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 3–12, 1994.

T. Luo. *Scaling up support vector machines with applications to plankton recognition*. PhD thesis, University of South Florida, forthcoming.

T. Luo, K. Kramer, D. Goldgof, L.O. Hall, S. Samson, A. Remsen, and T. Hopkins. Active learning to recognize multiple types of plankton. In *17th conference of the International Association for Pattern Recognition*, volume 3, pages 478–481, 2004a.

T. Luo, K. Kramer, D. Goldgof, L.O. Hall, S. Samson, A. Remson, and T. Hopkins. Recognizing plankton images from the shadow image particle profiling evaluation recorder. *IEEE Transactions on System, Man, and Cybernetics–Part B: Cybernetics*, 34(4):1753–1762, August 2004b.

P. Mitra, C. A. Murthy, and S. K. Pal. A probabilistic active support vector learning algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):413–418, 2004a.

P. Mitra, B. U. Shankar, and S. K. Pal. Segmentation of multispectral remote sensing images using active support vector machines. *Pattern Recognition Letters*, 25(9):1067–1074, 2004b.

H. T. Nguyen and A. Smeulders. Active learning using pre-clustering. In *Twenty-first International Conference on Machine learning*, 2004.

T. Onoda, H. Murata, and S. Yamada. Relevance feedback with active learning for document retrieval. In *Proceedings of the International Joint Conference on Neural Networks 2003*, volume 3, pages 1757–1762, 2003.

J. M. Park. Convergence and application of online active sampling using orthogonal pillar vectors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1197–1207, 2004.

J. Platt. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In *Advances in Large Margin Classiers*, pages 61–74, 2000.

J. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin DAGs for multiclass classification. In *Advances in Neural Information Processing Systems 12*, pages 547–553, 2000.

A. Remsen, T. L. Hopkins, and S. Samson. What you see is not what you catch: a comparison of concurrently collected net, optical plankton counter, and shadowed image particle profiling evaluation recorder data from the northeast gulf of mexico. *Deep Sea Research Part I: Oceanographic Research Papers*, 51:129–151, 2004.

R. Rifkin and A. Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101–141, 2004.

N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of 18th International Conference on Machine Learning*, pages 441–448, 2001.

S. Samson, T. Hopkins, A. Remsen, L. Langebrake, T. Sutton, and J. Patten. A system for high resolution zooplankton imaging. *IEEE journal of ocean engineering*, pages 671–676, 2001.

M. Sassano. An empirical study of active learning with support vector machines for japanese word segmentation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 505–512, 2002.

G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *Proceedings of 17th International Conference on Machine Learning*, pages 839–846, 2000.

B. Schölkopf and A. J. Smola. *Learning with kernels*. The MIT Press, 2002.

X. Tang, F. Lin, S. Samson, and A. Remsen. Feature extraction for binary plankton image classification. *accepted by IEEE Journal of oceanic engineering*, forthcoming.

S. Tong and D. Koller. Support vector machine active learning with applications to text classification. In *Proceedings of 17th International Conference on Machine Learning*, pages 999–1006, 2000.

V. N. Vapnik. *The nature of statistical learning theory*. Springer, 2000.

L. Wang, K. L. Chan, and Z. h. Zhang. Bootstrapping SVM active learning by incorporating unlabelled images for image retrieval. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 629–634, 2003.

M. K. Warmuth, G. Rätsch, M. Mathieson, J. Liao, and C. Lemmen. Support vector machines for active learning in the drug discovery process. *Journal of Chemical Information Sciences*, 43(2): 667–673, 2003.

T. F. Wu, C. J. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975–1005, 2004.