# Coaching the Exploration and Exploitation in Active Learning for Interactive Video Retrieval

Xiao-Yong Wei, *Member, IEEE*, and Zhen-Qun Yang

***Abstract*— Conventional active learning approaches for interactive video/image retrieval usually assume the query distribution is unknown, as it is difficult to estimate with only a limited number of labeled instances available. Thus, it is easy to put the system in a dilemma whether to explore the feature space in uncertain areas for a better understanding of the query distribution or to harvest in certain areas for more relevant instances. In this paper, we propose a novel approach called coached active learning that makes the query distribution predictable through training and, therefore, avoids the risk of searching on a completely unknown space. The estimated distribution, which provides a more global view of the feature space, can be used to schedule not only the timing but also the step sizes of the exploration and the exploitation in a principled way. The results of the experiments on a large-scale data set from TRECVID 2005–2009 validate the efficiency and effectiveness of our approach, which demonstrates an encouraging performance when facing domain-shift, outperforms eight conventional active learning methods, and shows superiority to six state-of-the-art interactive video retrieval systems.**

***Index Terms*— Coached active learning, interactive video retrieval, query-distribution modeling.**

## I. Introduction

MOST of the video retrieval applications are implemented with a single-round searching scheme, where a user types a query (usually a short text phrase) and then waits aside in the hope that the system is able to figure out what she/he wants and returns the results on demand. However, the system can seldom do so because it is often difficult for the user to describe her/his specific need with a short phrase. As a complementary technique to this single-round scheme, interactive retrieval models the search as an iterative process and allows the users to give *relevance feedback* so as to help the systems "understand" their query intentions. In each iteration, once the system returns a set of resulting items, a user can label some of them as "relevant" or "irrelevant". These labeled instances are then used for further estimating the user's query intention, on the basis of which the system starts a new round of searching. As users cannot keep constant patience with the iterative labeling, how to learn users' query intentions *effectively* is critical in designing an interactive retrieval system, a problem called *query learning* [1].

Active learning [2] is a commonly adopted scheme for this purpose, which speeds up the process by reducing the number of instances needed to be labeled. In active learning, the search is considered a process to train a classifier which distinguishes relevant instances from irrelevant ones. The training is generally conducted on the labeled instances (with relevant instances as positive examples and irrelevant ones as negative) and repeats every time when freshly labeled instances are obtained through the users' labeling. To reduce the efforts of labeling, existing approaches usually choose the instances closest to the decision boundary to query the users, because those are the instances the machine is most uncertain of, and the labeling of them will potentially cause a significant change of the classifer in the next round so as to ensure a faster convergence. This greedy strategy is termed as *uncertainty sampling* [3]–[10].

Even probably being the most popularly employed, however, uncertainty sampling has been found not of a "kindred soul" with interactive search. The spirit of uncertainty sampling is to reduce the efforts of labeling by avoiding querying users with instances that the machine is most confident with, e.g., those farthest from the decision boundary on the positive side. Nevertheless, in interactive search, those are exactly what the users are looking for and delay of labeling them may frustrate the users at early stages, resulting in abortion of the learning. In other words, uncertainty sampling aims to achieve a high quality classifier at the final stage and thus is not eager to "harvest" relevant instances returned by the precursor classifiers. In interactive search (especially at early stages), we concentrate more on the harvest and put less attention on the quality of the final classifier, a conflict also observed by A. G. Hauptmann *et al.* [11]. The problem can be related to an open question known as trade-off between *exploitation* and *exploration* [12], [13], where a learner always hesitates whether to *exploit* knowledge at hand for obtaining immediate rewards or to *explore* the unknown area for improving future gains. This is a dilemma encountered by any learning process conducted on an unknown distribution, but has not been addressed sufficiently by prior studies of active learning.

In terms of uncertainty sampling, the dilemma refers to the fact that in each run, a leaner has to choose between the *exploitation* of harvesting the instances that the machine is confident with so as to fulfill the retrieval and the *exploration* of querying the user with instances that the machine is

Manuscript received December 26, 2011; revised July 7, 2012; accepted September 3, 2012. Date of publication October 5, 2012; date of current version January 24, 2013. This work was supported in part by the National Natural Science Foundation of China under Grant 61001148 and Grant 61272256 and the Open Projects Program of National Laboratory of Pattern Recognition of China. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mark Liao.

The authors are with the College of Computer Science, Sichuan University, Chengdu 610065, China (e-mail: cswei@scu.edu.cn; jessicayang@scu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2012.2222902

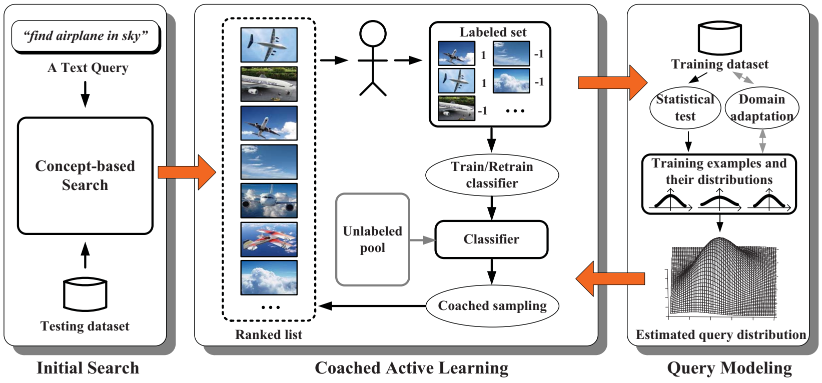


Fig. 1. Interactive retrieval framework consisting of three modules: initial search, query modeling, and coached active learning.

uncertain about so as to have a better understanding of the target space. The reason here is that the exploration of uncertainty sampling relies too much on the posterior knowledge indicated by the decision boundary, which provides only the local information about a small portion of the space, and, therefore, the exploitation and exploration cannot be balanced because of the lack of the global perspective. To address this problem, we have conducted a pilot study [14] in attempting to utilize prior knowledge (learnt from training dataset) to guide the active learning globally, so that the exploitation and exploration can be well scheduled and balanced. However, due to the expensiveness of the human experiments, the nature of the method has not been fully revealed in [14]. Furthermore, we notice that the prior knowledge learnt from training will sometimes suffer cross-domain issue when the target domain is different from that of the training.

In this paper, we extend our prior work [14] and propose a more sophisticated approach for active learning, which balances the exploitation and exploration, and addresses the cross-domain issue within an unified framework. Though a much more comprehensive study on a larger dataset, we will demonstrate that the extension is not only able to offer further performance improvements, but also reveals the principal factors (and their interdependence) related to interactive retrieval on the basis of which we can improve our future design. In the sense that the learning is jointly guided by the prior and posterior knowledge, we name it as coached active learning ($CAL$) to distinguish it from conventional approaches without the coaching process. Compared with existing methods in active learning, $CAL$ offers the following advantages:

1) *Predictable Query Distribution*: Conventional approaches of active learning usually assume the query distribution is unknown, because it is difficult to estimate with only a limited number of labeled instances. We argue that the query distribution is predictable in the scenario of interactive video search when enough training examples are available. To bypass the difficulty of estimating it on labeled instances directly, we use a statistical test to find training examples which are from the same distribution(s) as the labeled instances, and utilize the distribution(s) of those examples (much more abundant than the labeled instances) to predict that of the query. We will show that by analyzing the semantics carried by the training examples (found by the statistical test), we can even obtain the users' query intention *semantically*. With the estimated query distribution, we avoid the risk of blindly searching on a completely unknown distribution;
2) *Exploitation Versus Exploration*: The estimated query distribution, together with the distribution inferred from the decision boundary of current classifier, will be used to evaluate the relevancy and uncertainty of the instances, on the basis of which we can balance the exploitation and exploration in a principled way by controlling the proportion of two types of instances to query next, namely those with high relevancy (for boosting the exploitation) and those with high uncertainty (for improving exploration). The proportion is determined on the harvest history, so that we encourage exploitation to keep the users' patience if the number of freshly labeled relevant instances maintains a growth momentum statistically, and, otherwise, we encourage exploration to push the decision boundary towards uncertain area so as to increase the chance of locating more relevant instances in a long run;
3) *Domain Adaptivity*: It often happens that the domain of the testing dataset are different from that of the training dataset, making the prior knowledge learnt from training less applicable to the target domain. In $CAL$, we narrow the domain gap by gradually mixing the predictor distributions (learnt from the training examples) with the distribution of the labeled relevant instances (representing knowledge from target domain) in each round. This will keep the prior knowledge update-to-date and thus result in a more precise model of the query distribution.

With $CAL$, our interactive retrieval framework is shown in Fig. 1. It consists of three modules: initial search, query distribution modeling, and coached active learning. We employ a concept-based search method [15] to perform the initial search, which returns an initial ranked list for labeling. In query modeling, every time when the labeled set is updated by freshly labeled instances, we find training examples which are statistically from the same distribution(s) as the labeled relevant instances through a statistical test, and use their distributions as predictors to estimate a query distribution. During the coached active learning, the estimated query distribution together with the distribution of the current classifier outputs will be used to coach the sampling process, which selects unlabeled instances to organize a new ranked list to query the user in the next round. At the end of each round, we can integrate the distribution of the labeled instances into those of the predictor distributions to address the cross-domain issue.

## II. Related Work

### A. Active Learning

Active learning has been intensively studied over the last two decades, and has been applied to solve a diverse range of problems. In this section, we mainly review active learning within the scenario of video/image retrieval. For more comprehensive surveys, the reader is referred to B. Settles *et al.* [16] for active learning on general-purpose and T. S. Huang *et al.* [17] on multimedia retrieval.

The term *active learning* in literature is popularly referred to as pool-based active learning [18], where a classifier is trained iteratively on a small set of labeled instances and a large pool of unlabeled instances. In every iteration of learning, the major task is to selectively sample instances from the unlabeled pool to query users, so as to retrain the classifier with the updated labeled set in the next round. The sampling strategy, which is designed to minimize the number of instances needed to be labeled, is thus the core of active learning and a fundamental mark that distinguishes one existing method from others. Two types of strategies are popularly used: uncertainty sampling [3]–[10], and query-by-committee [18]–[20].

Uncertainty sampling is probably the most commonly employed scheme, where the idea is to select the instances that the learner is most uncertain of their labels. For a binary classification problem using a probabilistic model, it can be done by simply querying the instances whose posterior probability of being positive is nearest 0.5 [3], or more generally (and possibly most popularly), by using *entropy* to measure the uncertainty of an instance based on its posteriors probability predicted by the current classifier(s) [4], [5]. In video/image retrieval with active learning, the SVM classifier is widely adopted [6]–[10], [12], with which uncertainty sampling can be straightforwardly implemented to select the instances closest to the separating hyperplane where the classifier is usually confused and works awkwardly.

Different from uncertainty sampling in which inference is normally obtained through a single hypothesis (boundary), query-by-committee (QBC) organizes a committee with several classifiers for making a decision with the "wisdom of the crowds" [18]–[20]. Instances whose predicted labels are most inconsistent among committee classifiers are usually candidates to query next. Although being claimed to reduce the classification error and variance more efficiently than a single expert (classifier), QBC introduces computational burden in training more classifiers. Therefore, it is seldom used in large-scale video/image indexing and retrieval.

Additional to uncertainty sampling and QBC, there are other sampling strategies proposed with different premises, for example, of maximizing the expected model change (e.g., [21]), and of reducing the expected generalization error (e.g., [22]). The reader is referred to [16] for more details.

### B. Exploitation–Exploration Dilemma

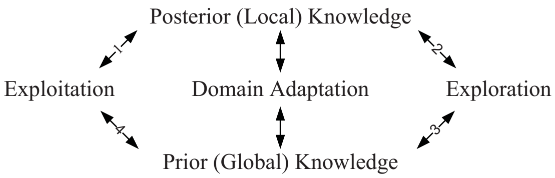
In the scenario of multimedia retrieval, uncertainty sampling with a SVM classifier is still the most prevalent scheme [3]–[10]. However, as we have introduced, the current way of using active learning for reducing labeling efforts may delay the harvest of relevant instances, which is inconsistent with the ultimate goal of multimedia retrieval [11], [23]. The algorithms proposed by Mitra *et al.* [12] and Thomas Osugi *et al.* [13] are two examples of the few works we found to address this open question of exploitation-exploration dilemma.

The algorithm in [12] adaptively estimates a confidence factor based on the population proportion of relevant instances to irrelevant ones along the current decision boundary, so as to encourage exploitation when the two groups of instances are well balanced (an indication that the current decision boundary is approaching the optimal position) and to encourage exploration otherwise. However, the searching along the boundary in this method is expensive on a larger-scale dataset. In [13], the authors propose an algorithm which randomly decides whether to exploit or explore according to a probability of exploring, which is dynamically determined and is proportional to the change of decision boundary in two consecutive rounds. Nevertheless, defining an appropriate function to measure the change is not a trivial task. Furthermore, in either [12] or [13], without knowing the prior query distribution, the step size of exploitation or exploration is still difficult to decide, which may either causes an unstable learning (if the step size is too larger) or slows down convergence (if too small).

Considering the way of incorporating prior data distribution into active learning, our work is also related to [7] and its successor [24]. Although these works have not been proposed for solving the exploitation-exploration dilemma, the use of prior knowledge for adjusting the learning has made them exhibit "coached-like" nature, which can address the dilemma to some extent. In [7], prior to the learning, the authors divide the target dataset by clustering and learn the prior distributions of the clusters in advance. The knowledge about those prior distributions is then used to weight the uncertainty of an unlabeled instance with reference to which cluster it belongs to. This method is further combined with uncertainty sampling in [24] for balancing between representativeness (indicated by the instances' prior distributions on clusters) and uncertainty. However, the parameters for clustering (e.g., number of clusters) are difficult to be determined adaptively in these two methods. More importantly, it is well known that instances in the same cluster of a feature space may carry diverse semantic information, because instances similar in low-level feature space are not necessarily related semantically.

### C. Three Principal Factors Related to Interactive Retrieval

To summarize the discussions above, there are indeed three principal factors related to interactive video retrieval, namely, *exploitation vs. exploration*, *prior vs. posterior knowledge* and *domain adaptation*. Understanding the impacts of these factors can help not only to explain the advantages/disadvantages of existing methods, but also to improve our future designs of the sampling strategies. As shown below,



the three factors are indeed interdependent, in the way that, 1) as the final goal of retrieval, exploitation is usually conducted based on our posterior understanding about a local area of the space; 2) therefore, to avoid to be stuck locally, we need to explore; 3) however, to be well-scheduled, the exploration relies on the guidance of the prior (global) knowledge; 4) the knowledge is dependent on the exploitation to keep up-to-date through domain adaptation. In this case, we can see that the intuition by always sampling the most relevant instances to query next is easy to be stuck locally because only relationship 1) is considered, and that most active learning approaches focus only on the relationship 2) and thus have an inconsistent goal with video retrieval. The works by Mitra *et al.* [12] and Thomas Osugi *et al.* [13] try to balance the exploitation and exploration may have considered both relationships 1) and 2) but ignored 3) and therefore are lack of global perspective, while the approaches [7] and [24] utilizing prior knowledge have considered 3) but put less emphasis on 1) and 2) because they are not designed for solving the exploitation-exploration dilemma. Our prior work [14] has considered 1), 2) and 3) but still ignored 4), making the "collaboration" between the prior and posterior knowledge not in a "mutual" manner (i.e., only the prior knowledge can influence the posterior knowledge, but not vice versa). In this paper, we consider all these relationships within an unified and more principled framework. We will demonstrate that, by further considering relationship 4), not only can the prior knowledge be kept up-to-date for more precise query modeling, but also are the three factors concerning interactive retrieval connected into a "virtuous circle", which brings further performance gains.

## III. Modeling Query Distribution

At each round, we can obtain a set $\mathcal{L}^+$ consisting of instances which are labeled as "relevant". Since $\mathcal{L}^+$ is just an incomplete sampling of the query distribution with limited number of instances, it is difficult to infer the query distribution from $\mathcal{L}^+$ directly. We propose an indirect way to infer the query distribution: first we organize training examples into nonexclusive groups according the semantics they carry; then we employ statistical test to find groups which are from the same distribution as $\mathcal{L}^+$; finally, the distributions of those semantic groups (which include much more abundant instances than $\mathcal{L}^+$) are used to approximate that of the query. At the end of each round, we use domain adaption to update the distribution of each semantic group, aiming to address the domain shift between the target and the training domains.

### A. Organizing Semantic Groups

In organizing training examples into semantic groups, we propose to maximize the semantic generalizablity of the resulting groups so that they can semantically cover as many queries as possible. We use *concept combination* [25] to achieve this goal. The idea is first organizing examples into elementary groups of concepts, and then combining those groups to create new groups covering more complicated semantics. Given one of the abundantly available concept definition and annotation collections[1] (e.g., Large-Scale Concept Ontology for Multimedia (LSCOM) [27]), we denote $G_i$ the set of instances labeled with concept $i$. Intersection or union operators are then used to combine those sets into new groups (i.e., $G_{i\cap j}$ and $G_{i\cup j}$). The combination can be done recursively among existing groups ($G_i$'s, $G_{i\cap j}$'s and $G_{i\cup j}$'s) to achieve more complicated ones, as long as the number of training examples in every combined group is larger than a fixed threshold (say 30), so that the group is statistically meaningful.

### B. Mapping Between $\mathcal{L}^+$ and Semantic Groups

With $\mathcal{L}^+$, the set including instances labeled as "relevant", we need to find groups that are from the same distribution as those in $\mathcal{L}^+$. This is a two-sample test problem which has a solid foundation in statistics. In our case, we employ the widely-used two-sample Hotelling T-Square test [28], where the hypothesis is that two samples are from the same multivariate distribution, and the significance level is 0.01. However, conducting Hotelling test which uses $\mathcal{L}^+$ to compare with candidate semantic groups ($G$'s) one by one is time-consuming, because the number of $G$'s can be up to a scale of hundreds of thousands. We employ two techniques to address this problem. First, we represent each $G$ with three parameters ($|G|$, $\boldsymbol{\mu}^G$, $\Sigma^G$), namely the number of examples, the mean feature vector, and the covariance matrix of $G$ respectively, which we can learn offline. As $\mathcal{L}^+$ can also be represented in a similar way, the Hotelling test is thus performed between parameters instead of the direct comparison between instances in $\mathcal{L}^+$ and examples in $G$, effectively avoiding the intensive computation. Second, we reduce the number of candidate groups to compare by pruning groups not semantically related to the query. More specifically, at query time, we use query-to-concept mapping [15], [29], [30] to select a set of concepts (denoted as $\mathcal{C}$) which are semantically related to the query. In Hotelling test, we only test $\mathcal{L}^+$ with groups that concern any concept in $\mathcal{C}$, while the sematic groups not including any concept in $\mathcal{C}$ will be pruned. For example, if *car* is one of the concepts in $\mathcal{C}$, we only investigate groups $G_{car}$, $G_{car\cap road}$, $G_{car\cup bus}$ and so on. In our experiment, this reduces the number of candidate groups to a scale of thousands, significantly accelerating the comparison.

### C. Estimating Query Distribution

With $\mathcal{L}^+$ and semantic groups selected in the last section, we use a gaussian mixture model (GMM) to estimate the

[1] These annotation collections are previously developed for training concept classifiers to fulfill the task of high-level concept indexing in TRECVID [26].

distribution of the current query $\mathcal{Q}$. Instances in $\mathcal{L}^+$ are used as training examples in GMM. Denoting the set of selected groups as $\mathcal{P}$ (predictor set), the probability density function (*pdf*) of the $\mathcal{Q}$ in GMM is:

$$P(\boldsymbol{x}|\mathcal{Q}) = \sum_{G\in\mathcal{P}} \omega_G P(\boldsymbol{x}|G) \tag{1}$$

where $\boldsymbol{x}$ is the feature vector of any instance, $P(\boldsymbol{x}|G)$ is the *pdf* of semantic group $G$, and $\omega_G$ is the weight for $P(\mathbf{x}|G)$ with the summation of all $\omega_G$ equal to one. Assuming each $G$ is generated by a standard multivariate normal distribution, $P(\boldsymbol{x}|G)$ can be easily estimated using the three parameters ($|G|$, $\boldsymbol{\mu}^G$, $\Sigma^G$). To estimate the weights $\omega_G$, we employ the most popular and well-established algorithm expectation-maximization (EM). With the instances in $\mathcal{L}^+$ as training examples, EM seeks an optimal set of weights $\omega_G^*$ (i.e., $\{\omega_1^*, \omega_2^*, \ldots, \omega_{|\mathcal{P}|}^*\}$) which maximizes the GMM likelihood

$$\{\omega_1^*, \omega_2^*, \ldots, \omega_{|\mathcal{P}|}^*\} = \underset{\{\omega_1,\omega_2,\ldots,\omega_{|\mathcal{P}|}\}}{\text{argmax}} \prod_{i=1}^{|\mathcal{L}^+|} P(\boldsymbol{x}_i|\mathcal{Q}). \tag{2}$$

To accelerate the convergence of EM, we assign each predictor distribution an initial weight

$$\omega_i^0 = \frac{sim(\mathcal{L}^+, G_i)}{\sum_{G\in\mathcal{P}} sim(\mathcal{L}^+, G)} \tag{3}$$

where $sim(.)$ is a function to compute similarity of two distributions. We implement this function as

$$sim(\mathcal{L}^+, G) = \exp\Big\{-\frac{1}{2}\Big(\sqrt{(\boldsymbol{\mu}^+ - \boldsymbol{\mu}^G)'(\Sigma^+)^{-1}(\boldsymbol{\mu}^+ - \boldsymbol{\mu}^G)} + \sqrt{(\boldsymbol{\mu}^G - \boldsymbol{\mu}^+)'(\Sigma^G)^{-1}(\boldsymbol{\mu}^G - \boldsymbol{\mu}^+)}\Big)\Big\}. \tag{4}$$

The similarity is indeed determined by the Mahalanobis distances between the mean vectors of the two distributions, with the intuition that the closer $G$'s distribution to $\mathcal{L}^+$'s, the larger the contribution of $P(\boldsymbol{x}|G)$ to $P(\boldsymbol{x}|\mathcal{Q})$. The EM will start from the initial $\omega_i^0$ and finally converges to $\omega_G^*$. Note that there might be other metrics available for measuring the distribution distance in Eq. (4) (e.g., Jensen-Shannon divergence). In our experiment, however, $\omega_G^*$ basically converges to the same optimal point even different metrics are used. This is an indication that searching for GMM weights in Eq. (2) is a convex optimization problem, and thus using different metrics will not affect the performance of $CAL$. We employ Eq. (4) because it is simple and can work seamless with our parameter-based representation of semantic groups.

### D. Addressing the Cross-Domain Problem in $CAL$

It could happen to all learning problems that, during testing, the target domain is different from the development domain where the models are trained. In $CAL$, we address this issue by involving a domain adaptation process to update the candidate sematic distributions (i.e., *pdf* for all semantic groups) at the end of each round. The idea is to merge the newly labeled relevant instances $\mathcal{L}^+$ of current round into each candidate semantic group $G$ and relearn the three parameters ($|G|$, $\boldsymbol{\mu}^G$, $\Sigma^G$) (cf. Section III-B) of $G$ using the mixed examples/instances. To be computationally effective, we will show this can also be implemented at a "parameter-level" without recalling the training examples of each group. To distinguish the data in different rounds, we add a subscript $i$ (indicating the $i$-th round) for these parameters. It is easy to prove that the parameters of group $G$ at the $(i+1)$-th round can be calculated as

$$|G_{i+1}| = |G_i| + |\mathcal{L}_i^+| \tag{5}$$

$$\boldsymbol{\mu}_{i+1}^G = \frac{\boldsymbol{\mu}_i^G|G_i| + \boldsymbol{\mu}_i^+|\mathcal{L}_i^+|}{|G_i| + |\mathcal{L}_i^+|} \tag{6}$$

$$\Sigma_{i+1}^G = \frac{\Phi_i + \Phi_i^+ - \boldsymbol{\mu}_{i+1}(\boldsymbol{\mu}_{i+1}^G)'|G_{i+1}|}{|G_i| + |\mathcal{L}_i^+| - 1} \tag{7}$$

where $\Phi_i$ and $\Phi_i^+$ are the inner product matrices of feature vector matrices of $G$ and $\mathcal{L}^+$ respectively. The proof can be found at our demo page[2]. To be practical, we have indeed maintained $\Phi_i$ at each round instead of $\Sigma_i^G$, since it can be obtained easily through Eq. (7). At the end of each round, we update these parameters for semantics groups that have been selected for modeling the query distribution in the current round. Note that, even this domain adaptation is not proposed to solve the general problem of domain-shift, it helps to keep the distributions of sematic groups up-to-date, on the basis of which we can obtain a more accurate estimation of the query distribution. More importantly, as we discussed in Section II-B, the domain adaptation connects the balance between the exploitation and exploration, and the collaboration between the prior and posterior knowledge into a "virtuous circle", which brings further performance improvements of $CAL$ (as we will validate experimentally in Section VI).

## IV. Coached Active Learning

In this section, we jointly utilize the prior knowledge (indicated by the estimated query distribution) and posterior knowledge (indicated by current decision boundary) to balance exploitation and exploration. This can be formulated as a function $\rho(\boldsymbol{x})$ that gives the priority of selecting an instance to query next. That is

$$\rho(x) = \lambda\, Harvest(\boldsymbol{x}) + (1-\lambda)\, Explore(\boldsymbol{x}) \tag{8}$$

where $Harvest(\boldsymbol{x})$ is a function to compute how likely the exploitation will be boosted if selecting $\boldsymbol{x}$ to query next, $Explore(\boldsymbol{x})$ computes how likely the exploration will be improved if selecting $\boldsymbol{x}$ to query next, and $\lambda$ is a balancing factor to control the exploitation-exploration trade-off. In our proposed method, the use of $\lambda$ is mainly for determining the timing of exploitation and exploration, and the step size is controlled in $Harvest(\boldsymbol{x})$ and $Explore(\boldsymbol{x})$ respectively. We will introduce the implementations of $Harvest(\boldsymbol{x})$ and $Explore(\boldsymbol{x})$ first, and then turn to $\lambda$.

[2] http://www.cs.cityu.edu.hk/~xiaoyong/CAL/

### A. Exploitative Priority

To boost the exploitation, one will intuitively select instances with high posterior probabilities $P(\mathcal{Q}|\boldsymbol{x})$ (i.e., the most relevant instances estimated by current classifier) to query next. However, as the current classifier is trained with $\mathcal{L}^+$ (a biased sampling of relevant instances to $\mathcal{Q}$), instances with high $P(\mathcal{Q}|\boldsymbol{x})$ are not necessarily lying in the densest area of query distribution. To encourage the exploitation to shift towards the dense area where the instances have high prior probabilities[3] $P(\boldsymbol{x}|\mathcal{Q})$, we implement the $Harvest(\boldsymbol{x})$ as

$$Harvest(\boldsymbol{x}) = P(\boldsymbol{x}|\mathcal{Q})P(\mathcal{Q}|\boldsymbol{x}). \tag{9}$$

The function indeed adjusts the posterior probability of $\boldsymbol{x}$ by further considering its prior probability. As in Fig. 2(b), the function will put higher priority to harvest the instances between the centers of prior and posterior distributions, instead of those around the center of the posterior distribution. During the iterative harvesting, the two centers will merge step by step. The step size is indirectly controlled with respect to the distance between the two centers.

### B. Explorative Priority

By considering both prior and posterior probabilities, we implement $Explore(\boldsymbol{x})$ as

$$\begin{aligned} Explore(\boldsymbol{x}) = &\Big\{-P(\boldsymbol{x}|\mathcal{Q})\log(P(\boldsymbol{x}|\mathcal{Q}))\Big\} \\ &\times\Big\{-P(\mathcal{Q}|\boldsymbol{x})\log(P(\mathcal{Q}|\boldsymbol{x}))\Big\} \end{aligned} \tag{10}$$

which is intuitively a summarization of two uncertainties inferred from the entropies of prior and posterior distributions respectively. As in Fig. 2(c), $Explore(\boldsymbol{x})$ puts high priority on exploring the instances residing in the intersection of the uncertain areas of prior and posterior distributions. There are three peaks for $Explore(\boldsymbol{x})$. The leftmost one is the highest peak which corresponds to the area both the prior and posterior distributions are most uncertain of, and the area where the left side of the optimal boundary is probably located. The peak in the middle corresponds to the area where prior and posterior distributions have a conflicting understanding. Referring to Fig. 2(a), the area corresponds to the region where the current decision boundary passes through the central area of the prior distribution, and it is thus also worth exploring to clarify the misunderstanding (if the centers of prior and posterior distributions are aligned exactly, the middle peak will disappear.) The rightmost peak corresponds to the area far from the current decision boundary but the prior distribution "thinks" that the right side of the optimal boundary should be located there. Exploring this area helps to clarify the right side of the optimal boundary. If the query distribution is well estimated, exploring the areas under the three peaks will have the effect of pulling the current boundary towards the optimal one. The step size of the exploration is indirectly controlled by the distance between the current decision boundary and the "imagined" optimal one by prior distribution.

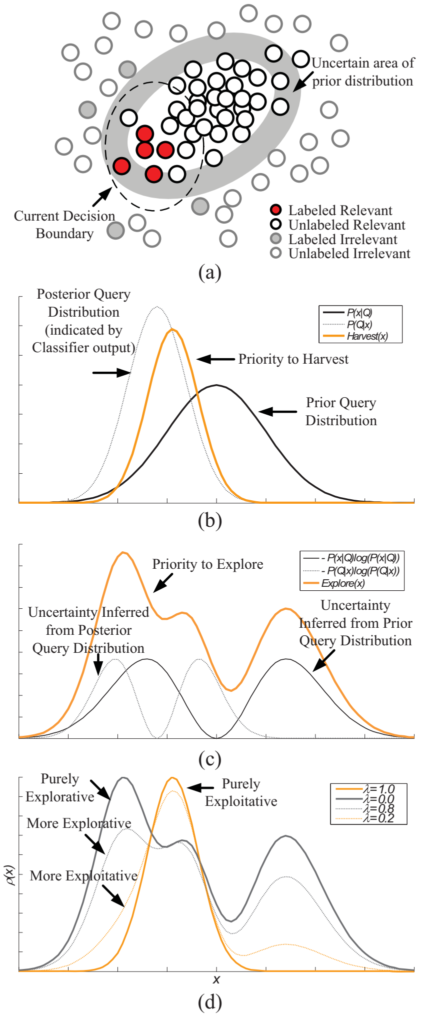


Fig. 2. Illustration of exploitative priority and explorative priority. (a) Example of a query distribution in target space. (b) Prior and posterior distributions and the distribution of the corresponding exploitative priority $Harvest(\boldsymbol{x})$. (c) Distributions of the uncertainties determined on the prior and posterior distributions and the distribution of the corresponding explorative priority $Explore(\boldsymbol{x})$. (d) Effect of using.

### C. Balancing Exploitation and Exploration

To balance the exploitation-exploration trade-off *adaptively*, every time when $\mathcal{L}^+$ is updated by freshly labeled instances, we update $\lambda$ based on the harvest history, i.e., number of relevant instances obtained in each round. We encourage exploitation when there are potentially more relevant instances to label, and encourage the exploration otherwise. To distinguish the data in different rounds, we add a subscript $i$ (indicating the $i$-th round) for both $\mathcal{L}^+$ and $\lambda$. The $\lambda_{i+1}$, balancing factor at the $(i+1)$-th round, can be updated as

$$\lambda_{i+1} = \lambda_i + \frac{|\mathcal{L}^+_{i+1}| - \bar{R}}{\bar{R}} \tag{11}$$

where $|\mathcal{L}^+_{i+1}|$ is the number of relevant instances in $\mathcal{L}^+_{i+1}$, and $\bar{R}$, which we present later, is the expected number of relevant instances to be labeled in the $(i+1)$-th round. In Eq. (11), $\lambda$ increases when the previous round harvested more relevant instances than expected, so that the learner will put more emphasis on exploitation, otherwise, $\lambda$ decreases resulting in more emphasis on exploration. When the updating in Eq. (11)

[3] Here we have slightly abused the term "prior probability" for $P(\boldsymbol{x}|\mathcal{Q})$ to emphasize the probability is learnt from the prior knowledge.

causes $\lambda$ out of the interval [0,1], we cut its value to the corresponding boundary. The expected number of relevant instances $\bar{R}$ in the $(i+1)$-th round, is computed according to the harvest history as

$$\bar{R} = \sum_{k=1}^{i} \frac{1}{2^{(i-k+1)}} |\mathcal{L}_k^+| \tag{12}$$

which is a weighted summarization of the number of instances harvested so far. Becasue the weight $\frac{1}{2^{(i-k+1)}}$ decreases dramatically from the current round to those of the previous rounds, the result of Eq. (12) is mainly influenced by the harvest of several most recent iterations[4]. The intuition is that we expect to harvest in the new round a little bit more than the number of relevant instances we have harvested recently. The effect of balancing with $\lambda$ is illustrated in Fig. 2(d), where the distribution of the priority function $\rho(x)$ appears more similar to that of exploitative priority $Harvest(\boldsymbol{x})$ if $\lambda$ is larger, and more similar to that of the explorative priority $Explore(\boldsymbol{x})$ otherwise.

### D. On the Use of Irrelevant Instances

So far, we have only discussed the use of labeled relevant instances (i.e., $\mathcal{L}^+$) in $CAL$. We should notice the use of labeled irrelevant instances (denoted as the set $\mathcal{L}^-$ hereafter) is also of great importance, as argued by Gosselin *et al.* [31]. The major concern is that the sizes of $\mathcal{L}^+$ and $\mathcal{L}^-$ should be balanced when selected for training. Since in $CAL$ the size of $\mathcal{L}^-$ is often larger than that of $\mathcal{L}^+$, we simply address this issue by randomly sampling irrelevant instances from $\mathcal{L}^-$ with the constrain that the number of negative samples is not greater than 3 times of $|\mathcal{L}^+|$, avoiding the number of negative samples in training is overwhelming to that of positive ones. Even appearing heuristic, the constrain has been reported with promising results in TRECVID evaluations in recent years. Since how to make a good use of negative samples for training is still an open question for machine learning, we leave its practice in $CAL$ for future investigations, because it is beyond the topic of this paper.

## V. Experiment-I: Study of the Essential Characteristics of $CAL$

Due to the large scale of the experiments, we divide the study of $CAL$ into three parts. In this section, we investigate $CAL$ in a real interactive retrieval environment for obtaining the essential characteristics of $CAL$ and the statistics about the users' behavior, on the basis of which, in Section VI, we pay particular attention to the coaching process of $CAL$, and finally conduct a comprehensive comparison between $CAL$ and the state-of-the-art in Section VII.

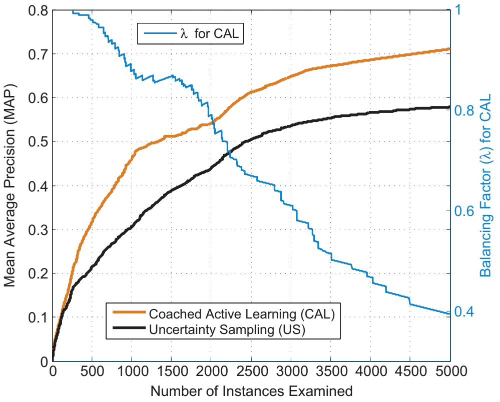


Fig. 3. Performance comparison of CAL and US at increasing depths of results, using the TRECVID 2005 test set. The MAP at each point is the average of MAPs of 36 users. The balancing factor $\lambda$ for CAL is also the average over 24 queries and across all users.

### A. Learning Candidate Semantic Distributions

With the method introduced in Section III-A, we first learn the candidate semantic distributions using LSCOM [27], the largest concept definition and annotation collection publicly available. LSCOM includes 2000+ concepts, of which 449 have been labeled on the development dataset of TRECVID 2005. To be practical, when learning semantic groups using the *concept combination*, we limit our interests to semantic groups being combined with no more than 2 concepts. After filtering out those with less than 30 training examples, we obtain 79,514 non-exclusive semantic groups, with the three parameters ($|G|$, $\boldsymbol{\mu}^G$, $\Sigma^G$) learnt for each group $G$. To further fulfill the normality assumption in Section III-C, we perform Mardia's test [28] (at significance level 0.05) on each distribution to check its similarity to the multivariate normal distribution and filter out those being rejected in the test. This finally results in 23,046 candidate distributions which will be used at query time to estimate the query distribution. The results can be downloaded from our demo webpage[2].

### B. Experimental Setting

We conduct experiments in this section using the TRECVID 2005 (TV05) test set, which is composed of 45,765 shots of news videos from multi-lingual sources including English, Chinese and Arabic. Twenty four search topics, together with their ground-truth provided by TRECVID, are used in the experiments. We only consider the text queries, imagining that most users perform search with a short sentence. We represent each shot with a concept vector which is composed of the detection scores on the shot by a set of semantic concept detectors. In our case, we use VIREO-374 [32] which includes detectors for 374 LSCOM semantic concepts. The dimensionality of our concept vector is thus 374. To train the classifier for each round, we use the well-known LIBSVM [33] with radial basis function (RBF) as kernel, gamma and cost empirically set to 0.386 and 8 respectively, and "−b" parameter enabled to output probability estimations (used as posterior probabilities in the experiments).

[4] The estimation of the expected harvest iteratively is a *Nonstationary Problem*, to which *Exponential, Recency-Weighted Average* is a commonly adopted solution. Eq. (12) is in fact a special case of this solution. A mathematical justification can be found on our demonstration webpage[2].

To compare the performance of our method to conventional approaches of active learning, we conduct an experiment of interactive search involving 36 searchers (volunteers aging from 19 to 26, 8 females and 28 males, 29 undergraduates and 7 graduates) on 3 PCs with exactly the same configuration (Intel(R) Core(TM) i3 CPU M3502.27GHz, 2GB Memory). Note that, to save the human power, we reuse data from 12 searchers which have been obtained in our previous work [14], where no domain adaptation (cf. Section III-D) has been employed in $CAL$. Therefore, for the rest of 24 searchers, in this section, we temporarily remove the domain adaptation in $CAL$, making the querying strategies consistent over all searchers. The impact of with and without integrating domain adaptation in $CAL$ will be studied specifically in the next section. Due to the expensiveness of the human experiment again, we only compare our coached active learning ($CAL$) with the most popularly employed uncertainty sampling ($US$), which chooses the instances closest to the SVM separating hyperplane to query next. In this case, each searcher has to conduct a total of 48 searches (24 TV05 queries for each method), resulting in 1,728 searches in total. To avoid the situation that the users might be negatively involved in searching new queries when they were frustrated by some "difficult" queries previously, the users are recommended to separate the task into several times within twenty weeks with each time at most 6 queries being taken. Queries are assigned to a user in stochastic order and experimented at each time with a randomly selected active learning method (i.e., the order of using $CAL$ or $US$ is also stochastic). We restrict the interval between the searches with $CAL$ and $US$ on a same query to at least four weeks, for reducing the effect that a user's experience on searching the query with one method may facilitate the searching with another.

The two sampling strategies are experimented within the same user interface as shown at our demo webpage[2], and the interactive search is conducted with the framework we present in Fig. 1 (to search with $US$, we replace the active learning module with uncertainty sampling). For initial search, we employ the work in [15], a concept-based search method which has been reported with encouraging results. All searches are experimented using the TRECVID model of interactive search, where a search has to be finished within 15 minutes, but users can give up early if they feel that there is no hope to find new relevant shots. The final result (i.e., the retrieved list) is evaluated with average precision (AP), following TRECVID standard. To aggregate the performance over multiple queries, we use the mean of their APs (known as MAP).

### C. Comparison With Uncertainty Sampling

Fig. 3 shows performance comparison of $CAL$ and $US$, where the MAP at each number of examined instances is the average MAP of 36 users. As a reference, we also plot the change of balancing factor $\lambda$ in $CAL$. It is easy to see that $CAL$ outperforms $US$ across the X-axis.

There are basically four stages for $CAL$, divided by the points 200, 1050, and 1800, and contributing 22.26%, 44.98%, 7.14%, and 25.62% of the total MAP growth, respectively. Overall, the $\lambda$ decreases gradually from left to right, indicating the fact that, while more and more relevant instances are harvested, $CAL$ can adaptively shift its focus from exploitation to exploration. This becomes more clear when looking at the second stage (from 200 to 1050), in which, even the relevant instances returned by the initial search are usually harvested out at the first stage, the estimated query distributions can successfully lead the classifiers to the dense areas, and the abundant relevant instances there results in fruitful harvest and stimulates the MAP growth. Accordingly, the value of $\lambda$ drops a little to encourage searching of the dense areas and remains as high as 0.90 to harvest in these areas.

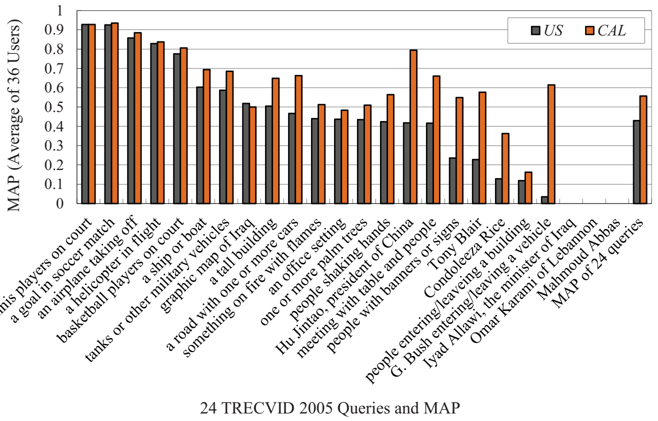


Fig. 4. Performance comparison of CAL and US over 24 TRECVID 2005 queries. The MAP for each query is the average of APs of 36 users. The texts of queries is manually modified to fit the figure.

Compared with that of $CAL$, MAP of $US$ increases more gently. The major disadvantage of $US$ is that, by always encouraging exploration and ignoring harvesting, $US$ may easily frustrate the users and cause an early give-up of search. To validate if there is true, we investigate $CAL$ and $US$ from three perspectives, namely the query-dependent, classifier-dependent, and user-dependent performance comparison.

*1) Query-Dependent Comparison:* The query-dependent comparison of $CAL$ and $US$ is illustrated by Fig. 4, where the final MAPs are obtained by evaluating the top-1000 shots in ranked lists that have been submitted after the 15-minute search. One can easily find that the performance superiority of $CAL$ over $US$ are more apparent among those "difficult" queries (in which both approaches exhibit moderate MAPs) than among the rest. For example, $CAL$ obtains performance gains over $US$ on queries "G. Bush entering or leaving a vehicle" and "Condoleeza Rice" by 1661% and 183.17% respectively. However, those of queries "tanks or other military vehicles" and "a ship or boat" are just by 16.85% and 15.09% respectively, even the MAPs on these two queries are comparably high. The former two queries include named entities. It is well known that this type of queries is too specific to enable an initial search without text-based search modules to collect enough instances. This is exactly what happened in our concept-based search, which thus returns insufficient relevant shots to satisfy the users at the beginning. In the pure exploration-oriented $US$, it becomes even worse and finally causes the user to give up early. On the contrary, $CAL$, with

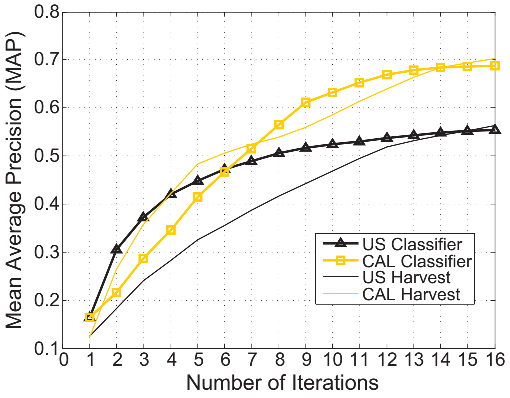


Fig. 5. Quality comparison of classifiers trained in $CAL$ and $US$. The harvest rate (measured by MAP) at each iteration is plotted for reference.

TABLE I

STATISTICS ABOUT USERS' PATIENCE ON LABELING INCLUDING: THE NUMBER OF SHOTS EXAMINED PER QUERY (#EXAMINED SHOTS), THE NUMBER OF IRRELEVANT SHOTS CONTINUOUSLY ENCOUNTERED BEFORE SUBMITTING CURRENT LABELS IN EACH ROUND (#IRRELEVANT SHOTS), AND THE AVERAGE RATE AND TIME OF USERS' EARLY GIVE-UP (IN MIN) BEFORE THE 15-min LIMIT

| Method | #Examined Shots | | | #Irrelevant Shots | | | Early Give-Up | |
|---|---|---|---|---|---|---|---|---|
| | Max | Avg. | Min | Max | Avg. | Min | Rate | Time |
| CAL | 5381 | 2402 | 213 | 213 | 62 | 14 | 12.51% | 13.23 |
| US | 4705 | 2013 | 132 | 132 | 31 | 13 | 18.63% | 10.42 |

the help of prior knowledge on query distribution, may have better chance to locate the dense areas and consequentially encourage the users on labeling in the succeeding iterations.

2) *Classifier-Dependent Comparison:* As introduced, in $US$, the learning focuses more on the quality of the classifier but ignores the harvest meanwhile. This is confirmed in Fig. 5, where we measure the quality of a classifier at each round by applying it on all the instances (no matter labeled or not) for classification, then sorting the instances in descending order according to their relevancy, and finally evaluate the top-1000 instances in the sorted list with AP. Each point in Fig. 5 is thus the average AP across 24 queries. For reference, the harvest rate at each iteration has also been evaluated by using AP on all the labeled relevant instances so far. It is easy to see that the harvest rate of $US$ at early iterations climbs slower than that of $CAL$, even the quality of its classifier is improving faster. By contrast, the classifier of $CAL$ increases moderately at early iterations, when it focus on harvest and its decision boundary is stuck locally. However, the situation starts to change after the 6-th iteration, where $CAL$ reaches its limit on harvesting in the dense area and moves its emphasis to exploration. The iterations from 5 to 10 of $CAL$ is roughly corresponding to its stage-3 in Fig. 3.

3) *User-Dependent Comparison:* It is worth mentioning that, compared with those in our previous study [14] where only 12 searchers are involved, the AP and MAP performances of $US$ in this section (with 36 searchers) generally decrease. This is due to the fact that the proportion of undergraduate students has increased from 66.7% to 80.5%. Since most of them are less experienced and patient than the graduate students, it is easier for them give up early in purely explorative strategies like $US$. The increase of undergraduate students, therefore, finally degrades the performance of $US$. By contrast, for $CAL$, we have not observed significant drop on its performance, which may further confirm $CAL$'s better ability to maintain users' patience.

### D. Study of User Patience on Labeling

To further study the effects of different sampling strategies on user patience, in Table I, we present some statistics about the users' patience recorded during the experiment. The statistics show that users are evidently more patient with $CAL$ than with $US$, indicated by their willingness to examined more shots overall and patience of waiting when irrelevant shots are continuously appeared. The maximum numbers of irrelevant shots continuously encountered before each submission are collected from the query "Mahmoud Abbas", where initial search fails to find any irrelevant shots. In this extreme case, the users are still able to examine 213 shots with $CAL$ before giving up, 81 more than with $US$. Moreover, the give-up rate of $CAL$ is 6.12% lower than that of $US$, and the users are able to sticking on the labeling for 2.81 minutes longer in $CAL$ than in $US$ before they give up early.

It is worth mentioning that, even $CAL$ has introduced two additional processes for estimating query distribution and adjusting step size of exploration, the online time cost per round for $CAL$ estimation is only 0.547 seconds on average. According to our questionnaire, no searcher has noticed the difference between the online time costs of $CAL$ and $US$ (0.113 seconds per round). In addition, despite which sampling strategy is used, we observe that the labels provided by users include errors by 2.04% on average for an user to label an irrelevant instance as relevant and by 5.38% vice versa.

## VI. EXPERIMENT-II: INVESTIGATION OF THE THREE FACTORS RELATED TO THE PERFORMANCE OF CAL

In this section, we study $CAL$'s performances by varying its configurations on three factors related to the coaching process, namely *exploitation vs. exploration*, *prior vs. posterior knowledge*, and *domain adaptation*. The aim is to investigate the questions like: Is the balance between exploration and exploitation important, or which one of them is more important than another? What the influence of the prior and posterior knowledge to the performance? Will a joint consideration of both the knowledge better than utilizing them separately? Is the domain adaptation able to boost the performance, and in what conditions it will or will not work?

### A. Experimental Setting

In the experiments, we further include shots from TRECVID (TV) 2006-2009 test datasets, which consist of 79,484 shots from TV06, 18,142 shots from TV07, 67,452 shots from TV08, and 93,902 shots from TV09. This finally results in a set of 304,745 shots from various domains (news and documentary), various periods (black-and-white and colored), various

TABLE II

PERFORMANCE COMPARISON OF CAL VARIANTS. THE BEST RESULT FOR EACH YEAR IS IN BOLD

| TRECVID- | Adaptive λ | Fixed λ | | | Prior Versus Post. | | Adaptive λ | Fixed λ | | | Prior Versus Post. | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $CAL$ | $CAL_0$ | $CAL_{.5}$ | $CAL_1$ | $PriCAL$ | $PosCAL$ | $CAL^A$ | $CAL_0^A$ | $CAL_{.5}^A$ | $CAL_1^A$ | $PriCAL^A$ | $PosCAL^A$ |
| 2005 | **0.563** | 0.459 | 0.482 | 0.32 | 0.324 | 0.461 | 0.558 | 0.465 | 0.497 | 0.334 | 0.331 | 0.469 |
| 2006 | 0.422 | 0.316 | 0.397 | 0.283 | 0.275 | 0.318 | **0.432** | 0.318 | 0.403 | 0.296 | 0.294 | 0.327 |
| 2007 | 0.493 | 0.391 | 0.448 | 0.293 | 0.283 | 0.402 | **0.508** | 0.405 | 0.473 | 0.312 | 0.308 | 0.449 |
| 2008 | 0.331 | 0.208 | 0.266 | 0.169 | 0.164 | 0.213 | **0.352** | 0.215 | 0.291 | 0.170 | 0.182 | 0.225 |
| 2009 | 0.380 | 0.248 | 0.309 | 0.148 | 0.136 | 0.252 | **0.397** | 0.257 | 0.358 | 0.172 | 0.151 | 0.274 |

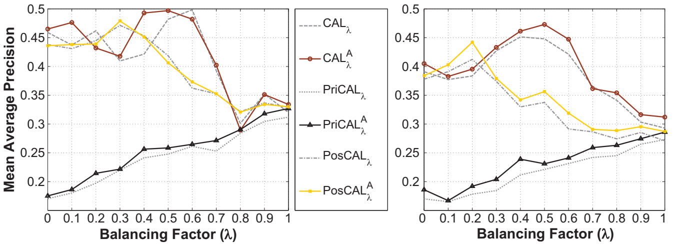
cultures (English, Chinese, Arabic and Dutch), and various photographic styles. There are 120 queries in our experiments (24 per year). Following the standard of TRECVID, we use average precision (AP) on TV05 and inferred average precision (InfAP) [34] on TV06–09 to evaluate each list.

Due to the expensiveness of human experiments, however, it is impractical to conduct the study in such a large scale using human searchers as in Experiment-I. To bypass the difficulty, we use machine searchers instead. A machine searcher is an agent program who "knows" the groundtruth and behaves in the interactive process according to the patience parameters we have learned in Table I. Random errors are added according to two error probabilities (that we learnt in Section V-D) by 2.04% for a machine searcher to label an irrelevant instance as relevant and by 5.38% vice versa. A 0.08 second delay are empirically added for the machine searchers to "browse" each shot, imitating the behavior of real searchers.

### B. Insights Into the Coaching Process

To fully study the nature of $CAL$, we investigate several variants of $CAL$ obtained by fixing the balancing factor $\lambda$ (cf. Section IV-C), by considering the prior or posterior knowledge separately or jointly, or by adding the domain adaptation (cf. Section III-D). In the following text, we represent all variants in an unified form of $CAL_\lambda^A$, where the superscript $A$ appears if the distribution adaptation has been embedded in the coaching (no superscript, otherwise), and the subscript $\lambda$ is a value ranging from 0 (purely explorative) to 1 (purely exploitative) if the balancing factor is fixed (no subscript if the $\lambda$ is adaptively determined). In addition, we add prefixes to these variations with $Pri$ to indicate only prior knowledge is considered (i.e, the $P(\mathcal{Q}|\boldsymbol{x})$ will be removed from both Eq. (9) and Eq. (10), and with $Pos$ to indicate only posterior knowledge is considered (i.e, the $P(\boldsymbol{x}|\mathcal{Q})$ will be removed from both Eq. (9) and Eq. (10)). No prefixes will be added if the two knowledge are considered jointly.

The performances of these variations on TV05-09 are shown in Table II. It is clear that $CAL^A$ and $CAL$ can always outperform the six variants using fixed balance factor $\lambda$ (i.e., $CAL_\lambda$'s and $CAL_\lambda^A$'s) with the improvements of $MAP$ ranging from 6.27% to 156.76%, and also outperform the four variants considering prior and posterior knowledge separately (i.e., $PriCAL$, $PosCAL$, $PriCAL^A$ and $PosCAL^A$) with the improvements of $MAP$ ranging from 13.14% to 179.41%. The superiorities have confirmed the importance of the adaptive exploitation-exploration balancing strategy and that of the joint consideration of prior and posterior knowledge. Comparing the two groups of variants with and without domain adaptation, the variants with domain adaptation have demonstrated superior performances over those of their correspondences without the adaptation, which confirms the effectiveness of the domain adaptation. To further investigate the influence (and the interdependence) of the three factors (i.e., *exploitation vs. exploration*, *prior vs. posterior knowledge*, and *domain adaptation*), Fig. 6 shows a detailed performance comparison of the variants with a more diverse range of configurations of these factors.



(a) (b)

Fig. 6. Detailed performance comparison of CAL variants under different configurations of $\lambda$'s on datasets of (a) TV05 and (b) TV07. The advantage of employing domain adaptation becomes clearer in (b) when the problem of domain shift is much more serious on TV07 than on TV05.

*1) Exploitation Versus Exploration:* It is easy to observe from Fig. 6 that, by varying the balance factor $\lambda$, basically all variants are approaching their maxima when the exploitation-exploration are well balanced at a certain point of $\lambda$. In addition, except $PriCAL_\lambda$ and $PriCAL_\lambda^A$, it is a consistent phenomenon that these variants perform better when more emphasis are put on the exploration (i.e., when $\lambda$ is low) than on the exploitation. This has confirmed our analysis of exploitation-exploration interdependence in Section II-C that it will be intuitively better to explore the unknown areas for finding new relevant instances than to purely harvest and be stuck locally. However, by considering the two exception runs $PriCAL_\lambda$ and $PriCAL_\lambda^A$, we will see that the conclusion is dependent on the use of the prior and posterior knowledge.

*2) Prior Versus Posterior Knowledge:* In Fig. 6, $PosCAL_\lambda$ and $PosCAL_\lambda^A$ have demonstrated apparently superior performances over those of $PriCAL_\lambda$ and $PriCAL_\lambda^A$. The reason is that, by only considering the prior knowledge, $PriCAL_\lambda$ and $PriCAL_\lambda^A$ can only harvest from either the dense or uncertain area of the global distribution which seldom changes, and thus will be stuck when the relevant instances from these two areas are all exploited. By contrast, by only considering the posterior knowledge, the active learner's understanding about the feature space, even being local, is expanding because the posterior knowledge will be updated after each iteration, which thus brings better ability for $PosCAL_{\lambda}$ and $PosCAL_{\lambda}^{A}$ to locate novel instances. However, without the global guidance of the prior knowledge, the exploration is still in a blind manner. Therefore, by jointly considering prior and posterior knowledge, $CAL_{\lambda}$ and $CAL_{\lambda}^{A}$ obtain superior performances over those variants that consider the two knowledge separately. This has confirmed our analysis of the prior and posterior knowledge interdependence in Section II-C. Furthermore, by combining the analysis in Section VI-B.1, we can see that the interdependence of prior and posterior knowledge has caused the interdependence of the exploitation and exploration, but the "collaborations" between the prior and posterior knowledge are also conducted through the exploitation and exploration.

*3) Addressing Cross-Domain Issue:* Due to the fact that our training dataset (TV05 development set) for learning the semantic groups is from news domain and the testing datasets are from both news (TV05-TV06 testing sets) and documentary (TV07-TV09 testing sets) domains, we are able to study the performance of $CAL$ when facing domain shift. As shown in Table II, the superiorities of $CAL^{A}$ to $CAL$ has increased when moving from news to documentary domain, indicated by the observation that $CAL^{A}$ outperforms the $CAL$ by 0.74% (±2.3%) on TV05-TV06, but the superiority rises up to 4.62% (±1.66%) on TV07-TV09. Therefore, by further considering domain adaptation, $CAL^{A}$ may have better capacity of handling the cross-domain issue. This has also been validated in Fig. 6 by the superiorities of $CAL_{\lambda}^{A}$'s to $CAL_{\lambda}$'s, $PriCAL_{\lambda}^{A}$'s to $PriCAL_{\lambda}$'s and $PosCAL_{\lambda}^{A}$'s to $PosCAL_{\lambda}$'s. It is interesting to see that $CAL^{A}$'s does not always outperform $CAL$'s on TV05. There are one such exception in Table II and several others in Fig. 6(a) where $CAL^{A}$'s performances are just comparable to those of $CAL$'s. However, all those exceptions disappear on TV07-TV09 (see the consistent superiority of $CAL^{A}$'s over $CAL$'s in Table II and Fig. 6(b)), indicating that domain adaptation gives its full play at where the domain-shift becomes serious on TV07-TV09. The advantage of using domain adaptation has also confirmed our analysis in Section II-C on the interdependence of the three factors, in the way that, in the variants with domain adaptation, posterior knowledge has obtained the chance to affect the prior knowledge and enables the "collaborations" between the two knowledge to be "interactive". However, in the variants without domain adaptation, the posterior dose not have any impact on the prior knowledge and thus the prior knowledge may not be up-to-date.

After the investigation of this section, we can see that the three factors contribute to the interactive retrieval through different paths, in the way that exploitation and exploration manipulate the active learning directly, the prior and posterior knowledge have to put their impacts on retrieval indirectly by affecting the exploitation and exploration, and domain adaptation has the longest path that updates prior and posterior knowledge first with the hope that its impacts will be transferred to exploitation and exploration and later to retrieval. However, our experimental results show that the three factors do help each other so as to form the "virtuous circle" as we discussed in Section II-C.

## VII. Experiment-III: Comparisons With the State-of-the-Art

In this section, we conduct a more comprehensive study of $CAL$'s performance from two aspects: the effectiveness comparing with eight querying strategies which are popularly employed in literature or closely related to the proposed method, and the comparison to five best interactive search systems reported in TRECVID. The experiments have been conducted on TV05-09 with machine searchers.

### *A. Effectiveness of Querying Strategies*

With the help of the machine searchers, we are able to compare $CAL$ and $CAL^{A}$ with a wide range of querying strategies as follows:

1) Nearest Neighbors ($NN$): samples instances to query next according to their proximities to the labeled relevant instance set $\mathcal{L}^{+}$, the simplest and most intuitive strategy which is purely exploitative;
2) Most Relevant ($MR$): samples instances according to their posterior probabilities by current classifier, an exploitative strategy utilizing only local information (current decision boundary);
3) Uncertainty Sampling ($US$) [3]–[10]: samples instances closest to the current decision boundary, a purely explorative strategy utilizing only local information;
4) Query by Committee ($QBC$) [18]–[20]: samples instances whose predicted labels are most inconsistent among committee classifiers (3 SVMs here), an explorative strategy but utilizing more comprehensive information than $US$ because of the employment of multiple classifiers;
5) Statistical Queries SVM ($StatQ$) [12]: samples instances according to a confidence factor which indicates the quality of current classification boundary, so as to encourage the exploitation when the boundary is approaching optimal and encourage exploration otherwise;
6) Hybrid Learning ($Hybrid$) [13]: a hybrid method which selectively chooses either an exploitative or an explorative strategy to sample instances according to a "probability of exploring" which is determined by the change of decision boundary of two consecutive rounds;
7) Pre-Clustering ($PreCls$) [7]: sample instances by considering both uncertainty and the prior data distribution learnt from clustering, so as to give higher probabilities to instances which are both uncertain to current classifier and representative in their clusters;
8) Dual Strategy ($DUAL$) [24]: sample instances using $PreCls$ before the derivative of the expected error is approaching zero and then combining $PreCls$ with conventional uncertainty-based strategy after that;

Table III shows the performances of these strategies on TV05-09 respectively, with the results of significance test (at level 0.05) on each year attached. It is clear that $CAL^{A}$ and $CAL$

TABLE III

PERFORMANCE COMPARISON OF DIFFERENT QUERYING STRATEGIES AND THE RESULTS OF SIGNIFICANCE TEST (AT LEVEL 0.05, AND X $\gg$ Y INDICATES THAT X IS SIGNIFICANTLY BETTER THAN Y). THE BEST RESULTS ARE IN BOLD

| **TRECVID-** | **Coached** | | **Coached-Like** | | | | **Exploitative** | | **Explorative** | | **Results of Significance Test** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $CAL$ | $CAL^A$ | $StatQ$ | $Hybrid$ | $PreCls$ | $DUAL$ | $NN$ | $MR$ | $US$ | $QBC$ | |
| 2005 | **0.563** | 0.558 | 0.423 | 0.363 | 0.446 | 0.463 | 0.208 | 0.317 | 0.431 | 0.452 | $CAL, CAL^A \gg DUAL, QBC, PreCls \gg US, StatQ \gg Hybrid \gg MR \gg NN$ |
| 2006 | 0.422 | **0.432** | 0.284 | 0.309 | 0.297 | 0.368 | 0.193 | 0.224 | 0.294 | 0.287 | $CAL^A \gg CAL \gg DUAL \gg Hybrid, PreCls, US \gg QBC, StatQ \gg MR \gg NN$ |
| 2007 | 0.493 | **0.508** | 0.388 | 0.397 | 0.403 | 0.442 | 0.237 | 0.286 | 0.374 | 0.382 | $CAL^A \gg CAL \gg DUAL \gg PreCls, Hybrid \gg StatQ, QBC \gg US \gg MR \gg NN$ |
| 2008 | 0.331 | **0.352** | 0.169 | 0.203 | 0.198 | 0.249 | 0.113 | 0.158 | 0.186 | 0.179 | $CAL^A \gg CAL \gg DUAL \gg Hybrid, PreCls \gg US, QBC \gg StatQ, MR \gg NN$ |
| 2009 | 0.380 | **0.397** | 0.213 | 0.249 | 0.242 | 0.277 | 0.129 | 0.136 | 0.227 | 0.235 | $CAL^A \gg CAL \gg DUAL \gg Hybrid, PreCls \gg QBC, US, StatQ \gg MR \gg NN$ |

can always outperform other eight conventional strategies with the improvements of $MAP$ ranging from 11.54% to 211.5%.

*1) Comparison With Purely Exploitative and Explorative Strategies:* The advantage of $CAL$ over the conventional exploitative strategies can be seen more clearly when we set $CAL$ to be purely explorative (i.e,. $\lambda = 1$). By referring to Table III, even $CAL_1$ and $CAL_1^A$ can outperform $NN$ and $MR$ by 41.86% (±15.07%) and 12.68% (±11.21%), respectively. The reason is that, as introduced in Section IV-A, by further utilizing the global information indicated by the estimated query distribution, the coaching process can iteratively guide the leaner to the center of the estimated distribution, putting the harvest in an orderly manner (i.e., from the places with densely located relevant instances to those with less). By contrast, the harvest processes of $NN$ and $MR$ are more "short sighted" in the way that they blindly search the neighboring areas around the relevant instances found in the previous rounds with the assumption that relevant instances are continuously distributed. However, this assumption is rarely true, because relevant instances are often mixed with irrelevant ones. It thus takes $NN$ and $MR$ a great effort filtering out the irrelevant instances before finding the true query distribution.

In cases that $CAL$ are purely explorative (i.e., $\lambda = 0$), by referring to Table III, $CAL_0$ and $CAL_0^A$ also outperform the conventional explorative strategies $US$ and $QBC$ by 9.28% (±3.32%) and 8.49% (±6.10%), respectively. This is again attribute to the coaching of the estimated query distribution, which selectively sample instances at places where the prior and posterior distributions both unsure of or have conflicting understandings, making the exploration well-targeted (cf. Section IV-B). However, $US$ and $QBC$, which sample instances only referring to local information, are lack of global perspective and easy to be stuck at local optimums.

*2) Comparison With Coached-Like Strategies:* Compared to the four coached-like strategies, $CAL^A$ and $CAL$ outperform $StatQ$, $Hybrid$, $PreCls$ and $DUAL$ by 59.26% (±30.33%), 48.58% (±15.88%), 45.33% (±27.43%) and 24.98% (±16.25%), respectively. By investigating the confidence factor of $StatQ$ [12], we find a dramatic fluctuation during retrieval which makes $StatQ$ extremely unstable on TRECVID dataset. The reason can date back to the basic assumption of $StatQ$ that the decision boundary is approaching optimal when the positive and negatives examples along it are well balanced. It is generally true when the distributions of relevant and irrelevant instances are far apart from each other. However, on TRECVID dataset, the majorities of the two distributions are often mixed, which will frequently change $StatQ$'s confidence about whether the boundary is approaching optimal. By contrast, we do not have this issue in $CAL$, because the prior distribution for coaching is learnt by excluding the influence of irrelevant instances, which will thus lead the learner to the most dense areas of the relevant instances, no matter how the relevant and irrelevant are mixed.

To measure the change of decision boundary, $Hybrid$ [13] concatenates the real-valued hypothesis (i.e., the probabilities to be relevant in case of SVM) of all samples in the dataset into a vector at each round and calculates the inner product of the two vectors in two consecutive rounds. It also meets problem on TRECVID dataset, because the population of the irrelevant instances is extremely overwhelming compared to the relevant ones. This results in most of the entities of the hypothesis vector seldom change.

Benefitting from the idea of integrating prior knowledge into active learning, $PreCls$ [7] has demonstrated better performance than all the conventional approaches using the exploitative, explorative and coached-like strategies. However, there are still several issues when $PreCls$ is applied to interactive video retrieval. First, the clusters in $PreCls$ are obtained by performing clustering on the whole dataset using the the similarity between feature vectors as the metric. This makes the instances in the same cluster are those visually similar to each other, but not necessarily sharing the same semantic meaning, on the basis of which it is thus inherently difficult for $PreCls$ to model the query distribution accurately. Second, the number of clusters in $PreCls$ is a highly sensitive parameter. We have done a large-scale of experiments to search the optimal number of clusters for $PreCls$ and finally we obtain a diverse results across TV05-09 (i.e., 507, 323, 435, 713, and 649 on TV05-09, respectively). The results reported in Table III are using these optimal numbers. However, the parameter tuning is impossible to conduct in real applications because the groudtruth is usually unavailable. Furthermore, the authors of [7] have suggested that performing a re-clustering after each round to update the prior knowledge can further improve the results. However, this is impractical in our case due to the large-scale of TRECVID dataset. It is easy to see that in CAL we can address the first issue by using semantic groups to model the query distribution, and address the last issue by updating the prior knowledge with domain adaptation.

By combining $PreCls$ with uncertainty sampling, $DUAL$ [24] has obtained the best performance among the eight conventional strategies. However, compared to $PreCls$, the improvement (16.02% (±8.58%)) is not as significant as reported in [24]. The reason is that, in our experiments, $DUAL$ spends a much longer period on $PreCls$ than on the combined strategy, making its performance mainly relying on that of $PreCls$. This is simply due to the large-scale of TRECVID dataset, on which, within the 15-minute limitation, it is not easy for $PreCls$ to reach to switching point where the derivative of the expected error is approaching zero. Therefore, the advantage of $DUAL$ has been limited on the large-scale interactive video retrieval.

### B. Comparisons With State-of-the-art

In this section, we compare the performance of $CAL$ with six state-of-the-art interactive search systems including: (1) MediaMill'05 [35], the best interactive search system reported in TRECVID 2005; (2) Extreme Video Retrieve (EVR) [11], the one reported with comparable performance to MediaMill'05. For EVR, we can only compare to the MAP of the method called manual paging with variable pagesize ($MPVP$) in [11], because even there are other methods proposed in the paper which can beat $MPVP$, but no exact MAPs are reported; (3) CMU'06 [36], (4) IBM'07 [37], (5) MediaMill'08 [38] and (6) MediaMill'09 [39] the best interactive search systems reported in TRECVID 2006–2009, respectively. Table IV shows the results. It is easy to see that $CAL$ ($CAL^A$) outperforms the best interactive systems by 38.67% (37.44%), 39.27% (42.57%), 36.19% (40.33%), 70.62% (81.44%), and 54.47% (61.38%) on TRECVID 2005–2009, respectively, and outperforms $MPVP$ by 35.99% (37.44%). A significance test also shows that both $CAL$ and $CAL^A$ outperform the five best interactive search systems[5] of TRECVID 2005–2009 at level 0.05. It again confirms that both $CAL$ and $CAL^A$ have state-of-the-art performance.

TABLE IV

PERFORMANCE COMPARISON ON TV 2005–2009 BETWEEN CAL AND THE BEST INTERACTIVE SEARCH SYSTEMS ($Best$). THE RESULTS ARE EVALUATED WITH THE AVERAGE OF APs OVER TOPICS ON 2005, AND OF InfAPs ON THE REST OF YEARS. THE BEST RESULTS ARE IN BOLD

| | 2005 | 2006 | 2007 | 2008 | 2009 |
|---|---|---|---|---|---|
| $MPVP$ | 0.414 | N/A | N/A | N/A | N/A |
| $Best$ | 0.406 | 0.303 | 0.362 | 0.194 | 0.246 |
| $CAL$ | **0.563** | 0.422 | 0.493 | 0.331 | 0.38 |
| $CAL^A$ | 0.558 | **0.432** | **0.508** | **0.352** | **0.397** |

## VIII. CONCLUSION

We have presented $CAL$ for addressing the exploration-exploitation dilemma in interactive video retrieval. By using pre-learnt semantic groups, $CAL$ makes the query distribution predictable, and thus avoids the risk of searching on a completely unknown feature space. With the coaching of the predicted query distribution and the posterior distribution indicated by current decision boundary, $CAL$ can balance between the exploration and exploitation in a principled way so as to keep the user's patience on searching as well as find their query intention effectively. The experiments on a large-scale dataset show that the proposed approach works satisfactorily when facing domain-shift and outperforms eight popular querying strategies, as well as six state-of-the-art systems. Through the study of $CAL$, we have identified the three principal factors related to interactive retrieval and their interdependence, which provides not only a higher perspective to review the existing works along the same line, but also a reference for improving our future design of sampling strategy.

## REFERENCES

[1] C. Campbell, N. Cristianini, and A. Smola, "Query learning with large margin classifiers," in *Proc. Int. Conf. Mach. Learn.*, 2000, pp. 111–118.

[2] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *J. Artif. Intell. Res.*, vol. 4, no. 1, pp. 129–145, 1996.

[3] D. Lewis and W. Gale, "A sequential algorithm for training text classifiers," in *Proc. ACM SIGIR Res. Develop. Inf. Retr.*, 1994, pp. 3–12.

[4] M. Tang, X. Luo, and S. Roukos, "Active learning for statistical natural language parsing," in *Proc. Annu. Meeting Assoc. Comput. Linguist.*, 2002, pp. 120–127.

[5] C. Zhang and T. Chen, "An active learning framework for content-based information retrieval," *IEEE Trans. Multimedia*, vol. 4, no. 2, pp. 260–268, Jun. 2002.

[6] M. R. Naphade and J. R. Smith, "Active learning for simultaneous annotation of multiple binary semantic concepts," in *Proc. IEEE Int. Conf. Multimedia Expo*, Mar. 2004, pp. 77–80.

[7] H. T. Nguyen and A. Smeulders, "Active learning using pre-clustering," in *Proc. Int. Conf. Mach. Learn.*, 2004, pp. 79–86.

[8] R. Yan, J. Yang, and A. Hauptmann, "Automatically labeling video data using multi-class active learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 516–524.

[9] L. Wang, K. L. Chan, and Z. Zhang, "Bootstrapping SVM active learning by incorporating unlabelled images for image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Mar. 2003, pp. 629–634.

[10] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proc. ACM Int. Conf. Multimedia*, 2001, pp. 107–118.

[11] A. G. Hauptmann, W.-H. Lin, R. Yan, J. Yang, and M.-Y. Chen, "Extreme video retrieval: Joint maximization of human and computer performance," in *Proc. ACM Int. Conf. Multimedia*, 2006, pp. 385–394.

[12] P. Mitra, C. A. Murthy, and S. K. Pal, "A probabilistic active support vector learning algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 413–418, Mar. 2004.

[13] T. Osugi, D. Kun, and S. Scott, "Balancing exploration and exploitation: A new algorithm for active machine learning," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2005, pp. 330–337.

[14] X.-Y. Wei and Z.-Q. Yang, "Coached active learning for interactive video search," in *Proc. ACM Int. Conf. Multimedia*, 2011, pp. 443–452.

[15] X.-Y. Wei, C.-W. Ngo, and Y.-G. Jiang, "Selection of concept detectors for video search by ontology-enriched semantic spaces," *IEEE Trans. Multimedia*, vol. 10, no. 6, pp. 1085–1096, Oct. 2008.

[5] $MPVP$ has not been included in this significance test, because the detailed APs for the 24 queries of TRECVID 2005 are not reported in [11].

[16] B. Settles, "Active learning literature survey," Dept. Comput. Sci., Univ. Wisconsin–Madison, Madison, Tech. Rep. 1648, 2009.

[17] T. Huang, C. Dagli, S. Rajaram, E. Y. Chang, M. I. Mandel, G. E. Poliner, and D. P. W. Ellis, "Active learning for interactive multimedia retrieval," *Proc. IEEE*, vol. 96, no. 4, pp. 648–667, Apr. 2008.

[18] A. McCallum and K. Nigam, "Employing EM and pool-based active learning for text classification," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 350–358.

[19] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," in *Proc. Int. Conf. Mach. Learn.*, 2000, pp. 999–1006.

[20] V. S. Iyengar, C. Apte, and T. Zhang, "Active learning using adaptive resampling," in *Proc. ACM SIGKDD Knowl. Discovery Data Mining*, 2000, pp. 92–98.

[21] B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 20. Cambridge, MA: MIT Press, 2008, pp. 1289–1296.

[22] X. Zhu, J. Lafferty, and Z. Ghahramani, "Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions," in *Proc. Workshop Continuum Labeled Unlabeled Data Mach. Learn. Data Mining*, 2003, pp. 58–65.

[23] M.-Y. Chen, M. Christel, A. G. Hauptmann, and H. Wactlar, "Putting active learning into multimedia applications: Dynamic definition and refinement of concept classifiers," in *Proc. ACM Int. Conf. Multimedia*, 2005, pp. 902–911.

[24] P. Donmez, J. G. Carbonell, and P. N. Bennett, "Dual strategy active learning," in *Proc. 18th Eur. Conf. Mach. Learn.*, 2007, pp. 116–127.

[25] X.-Y. Wei, Y.-G. Jiang, and C.-W. Ngo, "Concept-driven multi-modality fusion for video search," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 1, pp. 62–73, Jan. 2011.

[26] A. F. Smeaton, P. Over, and W. Kraaij, "Abstract evaluation campaigns and TRECVid," in *Proc. ACM SIGMM Int. Workshop Multimedia Inf. Retr.*, 2006, pp. 321–330.

[27] M. R. Naphade, J. R. Smith, J. Tešić, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis, "Large-scale concept ontology for multimedia," *IEEE Multimedia*, vol. 13, no. 3, pp. 86–91, Jul.–Sep. 2006.

[28] K. V. Mardia, J. T. Kent, and J. M. Bibby, *Multivariate Analysis*. New York: Academic, 1979.

[29] C. G. M. Snoek, B. Huurnink, L. Hollink, M. de Rijke, G. Schreiber, and M. Worring, "Adding semantics to detectors for video retrieval," *IEEE Trans. Multimedia*, vol. 9, no. 5, pp. 975–986, Aug. 2007.

[30] S.-Y. Neo, J. Zhao, M.-Y. Kan, and T.-S. Chua, "Video retrieval using high level features: Exploiting query matching and confidence-based weighting," in *Proc. Int. Conf. Image Video Retr.*, 2006, pp. 143–152.

[31] P.-H. Gosselin and M. Cord, "Active learning methods for interactive image retrieval," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1200–1211, Jul. 2008.

[32] Y.-G. Jiang, C.-W. Ngo, and J. Yang, "Toward optimal bag-of-features for object categorization and semantic video retrieval," in *Proc. ACM Int. Conf. Image Video Retr.*, 2007, pp. 494–501.

[33] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.

[34] J. A. Aslam and E. Yilmaz, "Inferring document relevance via average precision," in *Proc. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2006, pp. 601–602.

[35] C. G. M. Snoek, J. C. van Gemert, J. M. Geusebroek, B. Huurnink, D. C. Koelma, G. P. Nguyen, O. de Rooij, F. J. Seinstra, A. W. M. Smeulders, C. J. Veenman, and M. Worring, "The MediaMill TRECVID 2005 semantic video search engine," in *Proc. NIST TRECVID Workshop*, 2005, pp. 1–16.

[36] A. G. Hauptmann, M.-Y. Chen, M. Christel, D. Das, W.-H. Lin, R. Yan, J. Yang, G. Backfried, and X. Wu, "Multi-lingual broadcast news retrieval," in *Proc. NIST TRECVID Workshop*, 2006, pp. 1–12.

[37] M. Campbell, A. Haubold, M. Liu, A. Natsev, J. R. Smith, J. Tešić, L. Xie, R. Yan, J. Yang, "IBM research TRECVID-2007 video retrieval system," in *Proc. NIST TRECVID Workshop*, 2007, pp. 1–15.

[38] C. G. M. Snoek, O. D. Rooij, B. Huurnink, J. C. van Gemert, J. R. R. Uijlings, J. He, X. Li, I. Everts, V. Nedović, M. van Liempt, R. van Balen, F. Yan, M. A. Tahir, K. Mkolajczyk, J. Kittler, M. de Rijke, J. M. Geusebroek, T. Gevers, M. Worring, A. W. M. Smeulders, and D. C. Koelma, "The MediaMill TRECVID 2008 semantic video search engine," in *Proc. NIST TRECVID Workshop*, 2008, pp. 1–14.

[39] C. G. M. Snoek, O. D. Rooij, B. Huurnink, J. R. R. Uijlings, M. van Liempt, M. Bugalho, I. Trancoso, F. Yan, M. A. Tahir, K. Mikolajczyk, J. Kittler, M. de Rijke, J. M. Geusebroek, T. Gevers, M. Worring, D. C. Koelma, and A. W. M. Smeulders, "The MediaMill TRECVID 2009 semantic video search engine," in *Proc. NIST TRECVID Workshop*, 2009, pp. 1–14.

**Xiao-Yong Wei** (M'10) received the Ph.D. degree in computer science from the City University of Hong Kong, Kowloon Tong, Hong Kong, in 2009.

He is an Associate Professor with the College of Computer Science, Sichuan University, Chengdu, China. He is one of the founding members of the VIREO multimedia retrieval group, City University of Hong Kong. He was a Senior Research Associate with the Department of Computer Science and Department of Chinese, Linguistics and Translation, City University of Hong Kong, in 2009 and 2010, respectively. He was a Manager of Software Department, Para Telecom Ltd., Shanghai, China, from 2000 to 2003. His current research interests include multimedia retrieval, data mining, and machine learning.

**Zhen-Qun Yang** is currently pursuing the Ph.D. degree with the College of Computer Science, Sichuan University, Shanghai, China.

She was a Research Assistant with the Department of Computer Science, City University of Kong Kong, Kowloon, Hong Kong, from 2007 to 2008, and a Senior Research Assistant with the Department of Chinese, Linguistics and Translation, City University of Hong Kong, in 2009. Her current research interests include multimedia retrieval, image processing and pattern recognition, and neural networks.