An User Preference Information Based Kernel for SVM Active Learning in Content-based Image Retrieval *

Hua Xie and Antonio Ortega
Integrated Media Systems Center and Signal and Image Processing Institute
Department of Electrical Engineering-Systems
University of Southern California,
Los Angeles, CA 90089
Email: huaxie@sipi.usc.edu, ortega@sipi.usc.edu

ABSTRACT

Relevance feedback is a critical component for content-based retrieval systems. Effective learning algorithms are needed to accurately and quickly capture the user's query concept, under the daunting challenges of high dimensional data and small number of training samples. It has been shown that support vector machines (SVMs) can be used to conduct effective relevance feedback in content-based image retrieval. Most recent work along these lines has focused on how to customize SVM classification for the particular problem of interest. However, not much attention has been to paid to the design of novel kernel functions specifically tailored for relevance feedback problems and traditional kernels have been directly used in these applications. In this paper, we propose an approach to derive an information divergence based kernel given the user's preference. Our proposed kernel function naturally takes into account the statistics of the data that is available during relevance feedback for the purpose of discriminating between relevant and non-relevant images. Experiments show that the new kernel achieves significantly higher (about 17%) retrieval accuracy than the standard radial basis function (RBF) kernel, and can thus become a valid alternative to traditional kernels for SVM-based active learning in relevance feedback applications.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance Feedback*

General Terms

Algorithms, Human Factors

*This research has been funded in part by NASA under grant AIST-0122-0005 and by Integrated Media Systems Center, a National Science Foundation Engineering Research Center.

Keywords

Support vector machines, relevance feedback, kernel, probabilistic

1. INTRODUCTION

In the past few years, we have seen fast proliferation of multimedia information over the Internet. Content-based Information Retrieval (CBIR) systems are proposed for automatically indexing and accessing large amounts of information. In such systems, multiple features (color, texture, shape, etc.) are extracted from the query signals. Retrieval is performed using a similarity matching, where given an input feature pattern the goal is to search for similar patterns in the database. However, there is a major difficulty associated with CBIR schemes: the semantic gap between low-level features and high-level human concepts. Thus substantial efforts have been devoted to designing techniques that introduce the user into the loop, so that the system can learn the user's particular query preferences.

Relevance feedback provides a way for the user to interactively tune the system to her own interest by asking whether certain proposed images are relevant or not. The system then learns from these labeled examples to tune the parameters and returns a new set of similar images, iteratively repeating this process until the user is satisfied with the result. The construction of such a query updating scheme can be regarded as a machine learning task.

A majority of proposed approaches for relevance feedback in CBIR systems have been developed based on various forms of feature re-weighting [12][11], where the weights associated with each feature for a typical K-Nearest-Neighbor classifier are adjusted based on user feedback. The intuition is to emphasize (i.e., give them a more significant weight in the distance computation) those features that are best at discriminating between positive samples and negative ones.

A more systematic formulation of the relevance feedback problem can be achieved by setting up an optimization problem [8], where the goal is to find the optimal linear transformation to map the feature space into a *new* space, that has the property of clustering together positive examples, making it easier to separate them from negative ones.

More recently, several researchers have proposed the use of support vector machines as an active learning method for the relevance feedback problem in content-based retrieval [3] [7] [5] [6]. In [7], SVMs were first incorporated as an automatic tool to evaluate the preference weights of the relative images, which was then utilized to compute the query refinement [12]. A one-class SVM scheme was developed in [5] that tries to fit a tight hyper-sphere in the non-linearly transformed feature space (through a kernel) to include most positive samples. This scheme only employs the positive samples while totally neglecting the information provided by the negative samples. As an extension, a biased SVM was proposed in [6] to incorporate negative information by employing a pair of hyper-spheres, the inner one includes most of the positive instances while the outer one pushes out most of the negative samples. The unlabeled samples will then be classified as relevant if falling inside the inner sphere and non-relevant if falling outside the outer sphere. We can see that a key assumption made in both schemes is that the positive samples will actually be clustered together in the transformed space. Clearly, there is no guarantee that this will always hold true. Whether clustering does occur (in which case these SVM techniques are likely to be very successful) depends on the distribution of positive and negative samples and on the choice of kernel function.

Since the kernel function is a key factor to determine the discrimination ability of a SVM in this paper we propose a kernel function based on the information divergence between the probabilities of positive and negative samples inferred from the user's preferences. To the best of our knowledge this approach has not been used for relevance feedback in content-based image retrieval systems. Our work is inspired by [10] where a Kullback-Leibler (KL) divergence was used to derive the kernel function for SVM classification in speaker identification and image classification. Note that in [10] domain knowledge is available to model the data distributions that are used in computing the KL divergence. Statistical models such as Gaussian Mixture Models (GMM) or Hidden Markov Models (HMM) can very well model the data and the Expectation Maximization(EM) algorithm can be employed to learn and estimate the parameters. A more theoretical analysis of the use of Kullback-Leibler divergence to derive similarities between image classes, where each image class is modeled as Gaussian Mixtures, can be found in [15]. Although the idea of applying the Kullback-Leibler divergence to SVM learning is not new, in this paper we propose an extension of the framework in [10] for cases where the data distribution model is not known *a priori* and has to be inferred from user feedback.

In relevance feedback applications, there are no generic models for data distributions since the query concept is unknown and time-varying. We propose to employ an empirical method to capture the probabilistic information of the user's preference from the positive and negative samples and derive a new kernel called User Preference Information Divergence (UPID). Our scheme makes no prior assumptions on the data distribution, which is exactly what we are trying to learn. We performed the experiments based on a variety of image categories (from natural scenes such as Sunsets, coasts, to human civilizations such as Mayan & Aztec, Land of the Pyramids), the results show that the new kernel achieves significantly higher (about 17%) retrieval accuracy than RBF kernel, and even better than other kernel choices. Near 100% top-50 retrieval accuracy is achieved using the proposed kernel function after 6 relevance feedback iterations.

The paper is organized as follows. In section 2 we briefly review the concept of active learning for relevance feedback and support vector machines. Then we present our algorithm in section 3. Experimental results are shown in section 4, section 5 concludes our work.

2. SUPPORT VECTOR MACHINES FOR RELEVANCE FEEDBACK

Suppose that we are given L observations, with each observation consisting of a pair: a feature vector $\mathbf{x}_i \in R^n$, $i = 1, ..., L$, and the associated semantic class label y_i, which can be either +1 (relevant) or -1 (irrelevant), based on the user feedback. $\mathbf{x}$ can be modeled as a random variable drawn from a distribution with probabilities $\{P(\mathbf{x}|y = +1), P(\mathbf{x}|y = -1)\}$. The goal of relevance feedback is to learn the mapping $g : \mathbf{x}_i \mapsto y_i$ based on the labeled training set.

In the ideal case where we are able to estimate $\{P(\mathbf{x}|y = +1), P(\mathbf{x}|y = -1)\}$, the optimal mapping simply resolves to a maximum likelihood classifier (1):

$$g(\mathbf{x}) = \arg\max_i P(\mathbf{x}|y = i) \tag{1}$$

However, in relevance feedback applications, we are confronted with the difficulty of small sample problem [16], i.e., the number of available training samples is quite small relative to the dimensionality of the data. Thus it will be unrealistic to use traditional density estimation techniques for this purpose. Support vectors machines are adequate tools to address these challenges as they do not suffer from the Hughes phenomenon (or curse of dimensionality).[1]

We here give a brief introduction to the basic concepts of SVMs [13] [2]. Let $\{\mathbf{x}_i, y_i\}, i = 1, \cdots, L, y_i \in \{-1, +1\}, \mathbf{x}_i \in R^n$ be the labeled training set. SVMs are hyper-planes that separate the training data by a maximal margin, with all vectors labeled +1 lying on one side and all vectors labeled -1 lying on the other side (see Fig. 1):

$$\begin{aligned} \mathbf{w} \cdot \mathbf{x}_i + b &\geq +1 \quad for\ y_i = +1 \\ \mathbf{w} \cdot \mathbf{x}_i + b &\leq -1 \quad for\ y_i = -1 \end{aligned} \tag{2}$$

where $\mathbf{w}$ is normal to the hyperplane H. The training vectors that lie on hyper-planes $H_0 : \mathbf{w} \cdot \mathbf{x}_i + b = 1$ and $H_1 : \mathbf{w} \cdot \mathbf{x}_i + b = -1$, are called *support vectors*. It can be shown that the margin between the two hyperplanes H_0 and H_1 is simply $\frac{2}{\|\mathbf{w}\|}$, thus searching for the optimal separating hyperplane becomes a typical constrained optimization problem [2]: minimizing $\| \mathbf{w} \|^2$ subject to the constraints given by (2). By introducing Lagrange multipliers, this then leads to maximizing the Lagrangian objective function (3) with respect to positive Lagrange multipliers $\alpha_i, i = 1, \cdots, L$, subject to constrains $\sum_i \alpha_i y_i = 0$.

$$\max(\sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j) \tag{3}$$

If the training samples are not linearly separable in the original space χ, suppose that we first map the data to some other Euclidean space $\mathcal{H}$ (possibly infinite dimensional) using a mapping $\Phi : \chi \mapsto \mathcal{H}$. Since the training algorithm

[1] For a limited number of training samples, the classification accuracy decreases as the dimensionality increases.

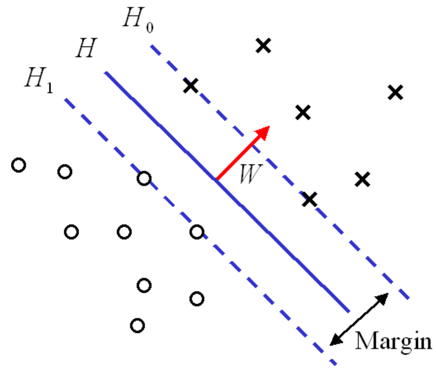

Figure 1: The optimal hyper-plane is the one that separates the positive samples from the negative ones with maximum margin.

only depends on the inner products between sample vectors, we can define a kernel function K such that $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$. Then we would only need to replace the inner product $\mathbf{x}_i \cdot \mathbf{x}_j$ by $K(\mathbf{x}_i, \mathbf{x}_j)$ everywhere in the training algorithm (3) and would never need to explicitly compute the mapping Φ. The resulting classifier takes the form of $g(\mathbf{x}) : \sum_{i=1}^{N_s} \alpha_i y_i K(\mathbf{x_i}, \mathbf{x}) + b$. $\{\alpha_i, i = 1, \cdots, N_s\}$ and b are the parameters that can be learned using quadratic programming [2]. N_s is the number of support vectors.

Most of the flexibility and classification power of support vector machine resides in the kernel function, since these make it possible to discriminate within challenging data sets, e.g., those where linear discrimination may be suboptimal. Typical kernel functions include: linear, polynomial and radial basis function (RBF):

$$\begin{aligned} \textit{Linear}: K(\mathbf{x}, \mathbf{z}) &= \mathbf{x} \cdot \mathbf{z} && (4) \\ \textit{Polynomial}: K(\mathbf{x}, \mathbf{z}) &= (A\mathbf{x} \cdot \mathbf{z} + B)^p && (5) \\ \textit{Radial Basis}: K(\mathbf{x}, \mathbf{z}) &= e^{-\gamma \|\mathbf{x}-\mathbf{z}\|^2} && (6) \end{aligned}$$

where $\mathbf{z}$ is another vector of the same dimension as $\mathbf{x}$ and $(\cdot)$ denotes the inner product of two vectors. A, B, p and γ are constants which are set a priori. It is important to note that these kernels are generic and do not explicitly take into account the statistics of user-provided feedback information available in content-based retrieval systems. Thus, if using an SVM in such a system, one would have to select a kernel *a priori* and then the performance of the system will depend significantly on the nature of the feedback provided by the user. In what follows we show how user feedback can be exploited in order to create a modified kernel function. Our experimental results demonstrate that these modified kernel functions consistently outperform others previously proposed in the literature.

3. KERNEL BASED ON USER PREFERENCE INFORMATION DIVERGENCE

Relevance feedback in content-based retrieval systems is based on high-level user preferences, thus making the learning of low-level similarity metrics a very difficult task. The perceptual interpretations of an image depends upon the user, the context of usage and the application. There are no generic models that are applicable to all scenarios. Our proposed method aims at learning the user's preference empirically, through probabilistic information that is contained in the user's feedback, and then using this information to derive a kernel that is customized for the specific user and task.

We assume that each image is represented by one feature vector $\mathbf{x} \in R^n$. Then the relevance feedback problem can be regarded as a machine learning task. The goal is to infer the user's preference on the unlabeled images (either relevant or non-relevant to the user's interest) based on the information learned from the user-labeled data.

For a given feature vector $\mathbf{x} = (x_1, x_2, \cdots, x_n)^t$, we define the marginal probability of each label for each component of the feature vector x_l as $\{P(y = +1|x_l), P(y = -1|x_l)\}$. These marginal distributions for each component x_l can be empirically estimated from the training data (i.e., from the feedback data in our case).Clearly, this estimation process is challenging because i) x_l can in general take values in either a large discrete set or a continuous range, and ii) limited amounts of data are available for training.

Given these challenges, parametric models for $\{P(y = +1|x_l), P(y = -1|x_l)\}$ could be considered but this in turn would imply that some prior assumptions need to be made about the distributions. Thus, in order to have as much flexibility as possible, we choose a non-parametric probability estimation approach. In what follows training data will refer to data obtained by accumulating successive iterations of user feedback.

For each feature vector component x_l we define a quantizer $\mathcal{A}_l$ that consists of B_l reconstruction levels r_{lk} with $B_l - 1$ decision boundaries denoted as $\{b_1, \cdots, b_{B_l-1}\}$. We estimate the probabilities $\{P(y = +1|x_l), P(y = -1|x_l)\}$ by counting the number of samples that fall in each bin:

$$P(y = \pm 1|x_l = r_{lk}) = \frac{\sum_{i=1}^{L} 1(y_i = \pm 1) 1(|x_{il} - r_{lk}| \leq \Delta_{lk})}{\sum_{i=1}^{L} 1(|x_{il} - r_{lk}| \leq \Delta_{lk})} \tag{7}$$

where the indicator function $1(\cdot)$ takes value one when its argument is true and zero otherwise. L is the number of labeled training data. x_{il} is the l-th component of training vector $\mathbf{x}_i$. $2\Delta_{lk}$ is the size of the quantization interval along dimension l centered at reconstruction value r_{lk}. For those quantization bins where there is no training data, we simply set the probability to zero since they make no contribution to differentiating classes. Obviously the design of quantizers $\mathcal{A}_l$s plays an important role in probability estimation. In this paper we use a simple uniform quantization scheme where all quantization bins in a given feature dimension have the same size $2\Delta_{lk}$, which is computed from the dynamic range of the data $[\max(x_l), \min(x_l)]$ (note that this range is changing from iteration to iteration) and the number of quantization levels applied B_l:

$$\Delta_{lk} = \Delta_l = \frac{\max(x_l) - \min(x_l)}{2 \times B_l} \tag{8}$$

More sophisticated techniques, such as K-nearest-neighbor, least-squares estimation etc., can also be used. We plan to explore this in our future work.

With the probability model we just described we can view a feature vector $\mathbf{x} = (x_1, x_2, \cdots, x_n)^t$ as a sample drawn from a random source, which has relevance statistics given by $P^+(\mathbf{x}) = (p_1^+, \cdots, p_n^+)$ and $P^-(\mathbf{x}) = (p_1^-, \cdots, p_n^-)$. $p_l^\pm =$

$P(y = \pm 1|x_l)$ are estimated by quantizing the component x_l using $\mathcal{A}_l$ based on the training data obtained from relevance feedback.

Assume that we wish to estimate the distance between $\mathbf{x}$ and $\mathbf{z}$, another feature vector with probability vectors $Q^+ = (q_1^+, \cdots, q_n^+)$ and $Q^- = (q_1^-, \cdots, q_n^-)$. We propose to use a distance based on the Kullback-Leibler divergence of their probability vectors P and Q:

$$D(\mathbf{x}||\mathbf{z}) = \sum_{l=1}^{n} p_l^+ \log(\frac{p_l^+}{q_l^+}) + \sum_{l=1}^{n} p_l^- \log(\frac{p_l^-}{q_l^-}) \qquad (9)$$

We assume $0 \times \log(0) = 0$ by continuity arguments. Since the KL divergence is not symmetric we define based on (9) a symmetric distance measure $D_s(\mathbf{x}, \mathbf{z})$ (the same technique was used in [10] to obtain a symmetric distance starting from the KL divergence):

$$D_s(\mathbf{x}, \mathbf{z}) = D(\mathbf{x}||\mathbf{z}) + D(\mathbf{z}||\mathbf{x}). \qquad (10)$$

We then define our proposed user preference information divergence (UPID) kernel function in the generalized form of RBF kernels with the original Euclidean distance $d()$ replaced by the proposed distance of (10):

$$K(\mathbf{x}, \mathbf{z}) = e^{-\rho D_s(\mathbf{x},\mathbf{z})} \qquad (11)$$

The distance (11) is a positive definite metric [10], thus the proposed UPID kernel satisfies Mercer's condition [2]. As the model parameters α_i, b and N_s are learned from the training set, we evaluate the likelihood that an unknown object $\mathbf{x}$ is relevant to the query by computing its score $f(\mathbf{x})$:

$$f(\mathbf{x}) = \sum_{i=1}^{N_s} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b \qquad (12)$$

Where $\mathbf{x}_i$ is the ith support vector and there are a total of N_s support vectors which is decided from the learning process. The larger the score is, the more likely it is that the unknown object belongs to the relevant class and thus shall be returned and displayed to the user.

4. EXPERIMENTS

As an experimental evaluation of the proposed scheme, we compare the performance of five types of learning methods: the query refinement and re-weighting (QRR) algorithm [12], SVM using polynomial kernel (Polynomial), SVM using Radial Basis function kernel(RBF), SVM using proposed probabilistic kernel (UPID), and SVM using linear kernel (Linear). 1500 real world images are chosen from the COREL Image CDs [1]. The image set includes 15 different categories[2], with 100 images for each category. Our experimental set up is very similar to that of [14], the only difference being that we replace the categories *auto racing* and *Roses*, with *Exotic cars* and *flowers*, respectively, since we do not have access to the former. We use 80% of each category (1200 images in total) as the database, and 20% (300 images in total) as the query images. The splitting of the data (80% and 20%) is to be consistent with the tradition that is used in machine learning.

We employ the feature extraction algorithm in [4]. Three different features are extracted to represent the images: color, texture and shape. The color features are computed as the histograms in CIELab color space. The texture feature is formed by applying Sobel operator to the image and histograming the magnitude of the local image gradient. The shape feature is characterized by histograming the angle of the edge. Dimensionality of the feature vector we used in our experiments is 72 (the number of bins used for each histogram is 8).

For the experiments, we assume that the query feedback is based on the actual image categories. The quality of the retrieval result is measured by two quantities: *precision* and *recall*. Precision is the percentage of relevant objects in the retrieved set to the query image, it measures the purity of the retrieval. Recall is a measurement of completeness of the retrieval, computed as the percentage of retrieved relevant objects in the total relevant set in the database.

When the system is presented with a query image, it will first search for the K nearest neighbors based on the Euclidean distance between the query image and each of the images in the database. Then the returned images which belong to the same category as the query image will be labeled as positive, and all the others in the returned set labeled as negative. The system learns the new model parameters and returns a new round of images and repeats this process. The labeled images accumulate from iteration to iteration as the system gets more feedback from the user. For Support Vector Machine based methods, the parameters α_is and b are learned using (3) based on the labeled images. The next round of retrieval will be carried out using the classifier (12) with the new set of parameters. The images with highest score are most likely to be the target images for the user. For Query Refinement and Re-weighting method, we implemented the algorithm proposed in [12]. Then the new query vector and new weights are used to perform a K-Nearest-Neighbor classification. The precision and recall are averaged over all the test images.

The support vector machine learning algorithms are implemented based on the SVM^{light} library [9]. Fig.2 shows the precision-recall curves comparing proposed method(with parameter ρ set to 1), Query Refinement and Re-weighting (QRR), SVM with RBF kernel(with parameter γ set to 1), SVM with polynomial kernel(degree p = 4, $A = B = 1$), and linear kernel. In proposed scheme, we fix the number of quantization bins for all dimensions to be the same.

Top-k retrieval precision as a function of the number of returned images are plotted in Fig.3. We can clearly see that proposed method achieves significantly higher search accuracy than the other methods.

Table 1 shows the top-k accuracy(mean and variance) after 6 relevance feedback iterations. We can see that SVM based active learning methods perform significantly better than the query refinement/re-weighting method. SVM with proposed empirical probabilistic kernel function is the best performer among all SVM based methods. It achieves almost 100% top-50 accuracy, while RBF kernels get around 92%. In [3], the performance of RBF and polynomial were reported for a 4-category dataset. The top-50 accuracy after 3 iterations are 92.7% for polynomial degree 4, and 96.8% for RBF. Considering that learning is more accurate with a

[2]Sunset, Coasts, Flowers (volume II), Exotic cars, Mayan & Aztec, Fireworks,Ski scenes, Owls, Religious Stained glass, Arabian horses, Glaciers & Mountains, English country gardens, Divers & diving, Land of the pyramids, and oil paintings

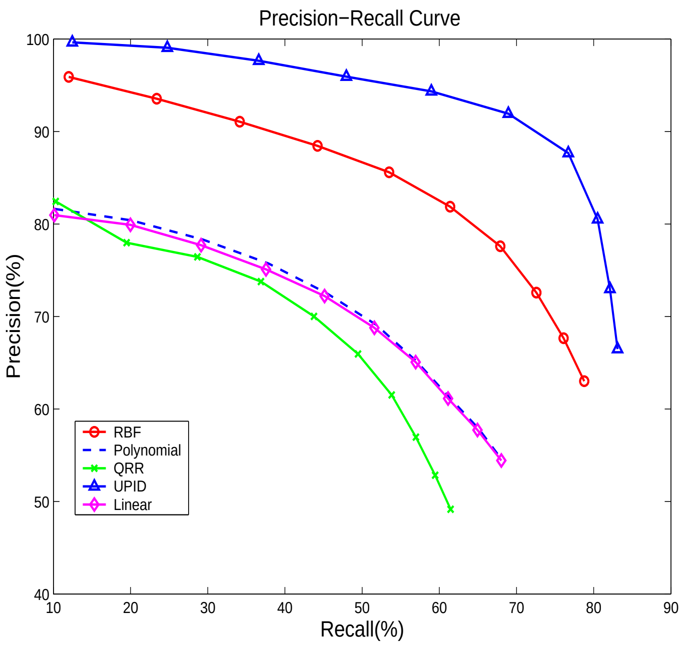

Figure 2: Precision-Recall curves after 3 relevance feedback iterations, comparing four methods: SVM with RBF kernel(Circles), SVM with Polynomial degree 2(Dashed lines), Query Refinement and Re-weighting(Cross), SVM with Proposed UPID kernel (Triangles), and SVM with Linear Kernel(Diamonds).

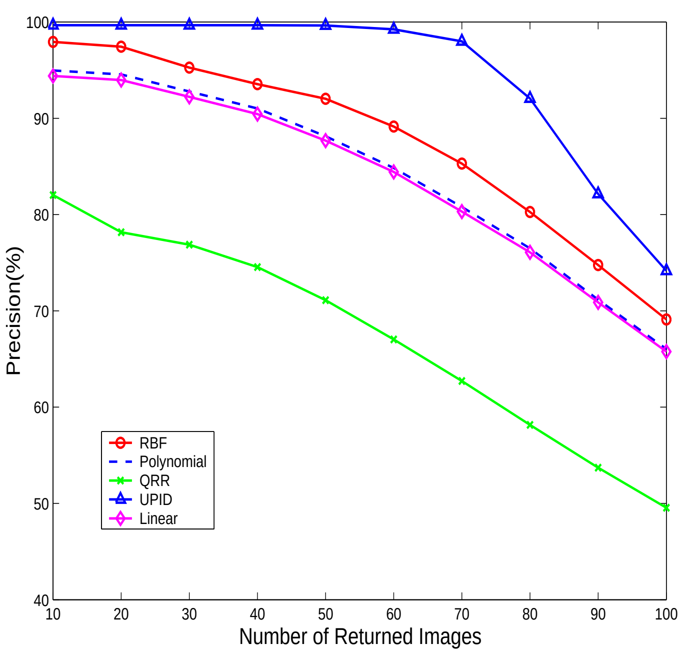

Figure 3: Top-k accuracy as a function of the number of returned images after 6 relevance iterations. We can see that compared to other methods, proposed method has a more *c*ompact display of the relevant images (Precision is relatively flat in the beginning and gets a sharper tail off.)

smaller dataset, our results are consistent with theirs.

Algorithm	Top-20	Top-50	Top-80
RBF	97.43±0.44	92.03±0.77	80.27±0.79
Polynomial	94.60±0.64	88.69±0.85	77.12±0.78
UPID	**99.67±0.33**	**99.64±0.33**	**92.05±0.73**
Linear	93.97±0.82	87.69±1.07	76.08±0.83
QRR	78.17±7.4	71.11±7.56	58.15±4.96

Table 1: Top-k accuracy(mean and variance) after 6 relevance feedback iterations comparing various methods. Bold numbers indicate the best performer. The parameters chosen are: $\gamma = 1$ for the RBF kernel, $p = 4, A = B = 1$ for the polynomial kernel, and $\rho = 1$ for proposed UPID kernel. We implemented the query refinement and re-weighting based on the algorithm in [12].

We also show in Fig.4 the improvement of the retrieval accuracy as a function of the number of interaction rounds. It basically gives us an idea that to what extent and how fast (how many interactions are needed in order to achieve a certain accuracy) the system is able to capture the query concept through the information provided at each interaction round.

We see that proposed kernel outperforms the other three most popular kernels. About 17% higher accuracy than RBF kernel is achieved using proposed kernel after the first iteration. It is encouraging since usually very small number of positive samples are available at the beginning of the interaction.

We then investigated the reliability of proposed empirical estimation scheme (7) by varying the number of quantization bins B_l. We tested on $B_l = 10, 15, 20$ and figure 5 shows the precision-recall curves of proposed scheme after 3 and 6 relevance feedback iterations. We can see that varying the number of quantization bins does not have a significant effect on the learning performance. Another fact worth noticing is that the number of labeled positive samples is relatively small in the beginning of the learning process, and still the accuracy improvement is remarkable after only one relevance feedback using proposed method (17% higher than RBF kernel, see Fig. 4). We plan to investigate adapting the quantization during the process of learning for each iteration as newly labeled sample occurrences are incorporated.

5. CONCLUSIONS AND FUTURE WORK

In this paper we proposed a new method of employing the data statistics for active learning using SVM in content-based image retrieval. The derivation of the new kernel is empirical and requires no domain knowledge, it is thus a practical approach for relevance feedback learning tasks where the query concept is not known and can be time varying. Our experiments have shown promising performance using proposed scheme compared with other kernels. Our future work includes designing adaptive method for estimating the marginal distributions(current version uses a simple uniform quantization to estimate the probabilities), taking into account the data imbalance problem (the number of negative samples is much larger than the number of positive samples), and speeding up of the learning. We are also investigating how to incorporate the ranking information (i.e.,

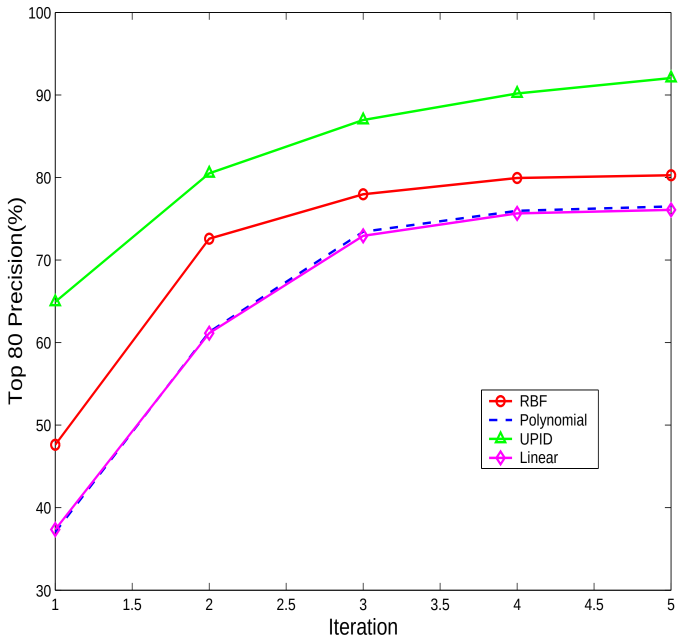

Figure 4: Comparison of learning accuracy of three different kernels (evaluated as the top-80 retrieval precision) as a function of the number of relevance feedback iterations. The accuracy without relevance feedback is 40.78%, it is obtained by a K-Nearest-Neighbor classifier with the weights equal for all feature components.

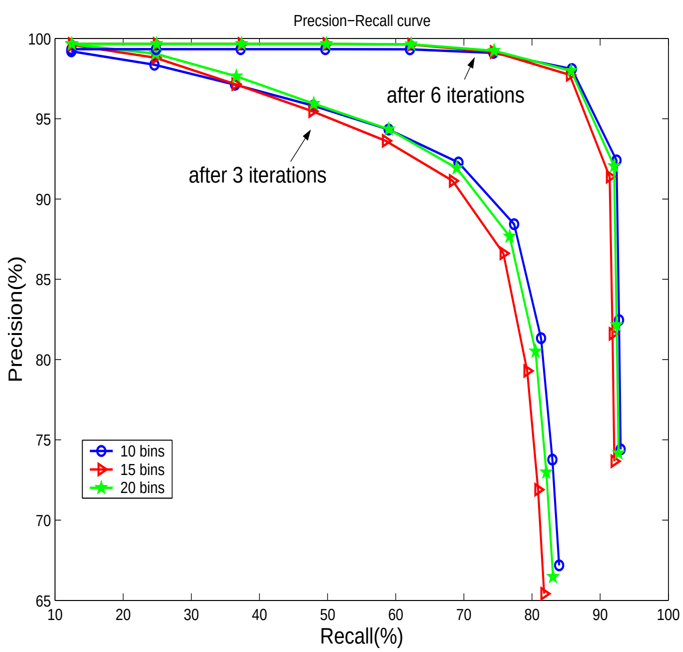

Figure 5: Precision-Recall curves of proposed scheme after 3 and 6 relevance feedback iterations using different number of quantization bins for probability estimation. We can see that varying the number of quantization bins doesn't has much effects on the learning performance, thus the proposed empirical estimation scheme is very reliable.

cases when the user has different degrees of preference for the relevant images) into our framework.

6. REFERENCES

[1] Corel stock photo library. Corel Corp. Ontario, Canada.

[2] C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge discovery*, pages 121–167, 1998.

[3] E. Chang and S. Tong. Support vector machine active learning for image retrieval. *ACM International Conference on Multimedia*, pages 107–118, October 2001.

[4] J.-Y. Chen, C. A. Bouman, and J. Dalton. Similarity pyramids for browsing and organization of large image databases. *Proc. of SPIE Conf. on Human Vision and Electronic Imaging III*, 3299, Jan. 1998.

[5] Y. Chen, X. Zhou, and T. S. Huang. One-class support vector machine for learning in image retrieval. *IEEE International Conference on Image Processing*, 2001.

[6] C. Hoi, C. Chan, K. Huang, M. R. Lyu, and I. King. Biased support vector machine for relevance feedback in image retrieval. *International joint conference on Neural Networks*, 2004.

[7] P. Hong, Q. Tian, and T. S. Huang. Incorporate support vector machines to content-based image retrieval with relevance feedback. *IEEE International Conference on Image Processing*, 2000.

[8] T. S. Huang and X. S. Zhou. Image retrieval with relevance feedback: From heuristic weight adjustment to optimal learning methods. *IEEE International Conference on Image Processing*, 2001.

[9] T. Joachims. *Advances in Kernel Methods - Support Vector Learning*, chapter 11: Making Large-scale support vector machine Learning Practical. MIT Press, 1998.

[10] P. J. Moreno, P. Ho, and N. Vasconceles. A kullback-leibler divergence based kernel for svm classification in multimedia applications. *Neural Information Processing Systems*, 2003.

[11] J. Peng, B. Bhanu, and S. Qing. Probabilistic feature relevance learning for content-based image retrieval. *Computer Vision and Image understanding*, 75:150–164, 1999.

[12] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: A power tool in interactive content-based image retrieval. *IEEE trans on Circuits and Systems for Video Technology*, 8:644–655, 1998.

[13] V. Vapnik. *The nature of statistical learning theory.* Springer-Verlag, New York, 1995.

[14] N. Vasconcelos. Exploiting group structure to improve retrieval accuracy and speed in image databases. In *International conference on Image Processing*, 2002.

[15] N. Vasconcelos. On the efficient evaluation of probabilistic similarity functions for image retrieval. *IEEE Trans. on Information Theory*, pages 1482–1496, July 2004.

[16] X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval: A comprehensive review. *ACM Multimedia Systems Journal*, 2002.