Active Concept Learning in Image Databases

Anlei Dong and Bir Bhanu, *Fellow, IEEE*

***Abstract*—Concept learning in content-based image retrieval systems is a challenging task. This paper presents an active concept learning approach based on the mixture model to deal with the two basic aspects of a database system: the changing (image insertion or removal) nature of a database and user queries. To achieve concept learning, we a) propose a new user directed semi-supervised expectation-maximization algorithm for mixture parameter estimation, and b) develop a novel model selection method based on Bayesian analysis that evaluates the consistency of hypothesized models with the available information. The analysis of exploitation versus exploration in the search space helps to find the optimal model efficiently. Our concept knowledge *transduction* approach is able to deal with the cases of image insertion and query images being outside the database. The system handles the situation where users may mislabel images during relevance feedback. Experimental results on Corel database show the efficacy of our active concept learning approach and the improvement in retrieval performance by concept transduction.**

***Index Terms*—Concept transduction, mixture model, model selection, relevance feedback, semi-supervised expectation-maximization (SS-EM) algorithm.**

I. Introduction

In RECENT years, content-based image retrieval (CBIR) [1] has received widespread research interest in the fields of computer vision and pattern recognition and their applications. Based on the visual features (such as color, texture, and shape) extracted from images, CBIR systems attempt to cater to the needs of users who want to retrieve images belonging to their desired concepts in mind.

The *relevance feedback* mechanism [2] makes it possible for CBIR systems to learn human concepts since users provide some positive and negative image labeling information, which helps systems to dynamically adapt and update the relevance of images to be retrieved. The main techniques in relevance feedback include *query shifting* [3], *relevance estimation* [4], [5], and *Bayesian inference* [6]. The technique of support vector machine is also proposed for image retrieval [7]. These methods learn concepts in a single relevance feedback session only. They do not learn concepts in the feature space systematically. Furthermore, once the user is done with a query and starts a new query, the meta knowledge gained by the systems with previous queries is lost. Meta knowledge is the experience of each query image with various users. This experience consists of the classification of each image into various classes (clusters), relevances (weights) of features and the number of times this image is selected as a query and marked as positive or negative.

Since real image databases experience retrievals from many users, it is possible to exploit previous retrieval experiences (meta knowledge) to learn and refine visual concepts. Some CBIR systems exploiting meta knowledge for concept learning and retrieval improvement have appeared recently. Yin *et al.* [8] combine traditional relevance feedback methods with the technique of *virtual feature* which is derived from long-term retrieval experience. The image dissimilarity measure can be adapted dynamically depending on the estimate of the relevance probability derived from the virtual feature. Fournier and Cord [9] combine users' annotations in parallel with the content-based similarity which is called the *compound query* technique. Bhanu and Dong [10] integrate feature relevance learning with fuzzy clustering which partitions the image database for efficient indexing with the help of meta knowledge. In all these works, with retrieval experiences in conjunction with relevance feedback, the concept learning is improved, which helps to capture users' desired concepts more precisely and, thus, future retrieval performance is improved. However, this process necessitates a model for the learning process; otherwise, it is only empirical.

A pioneering work in the related field of information retrieval with the strict model is presented by Yu *et al.* [11], who devised a statistical model for relevance feedback. They gave a theoretical proof that the query shifting formula can improve retrieval performance and justified how the values of parameters in this formula affect this improvement. In image retrieval research, few statistical models have been developed for strict theoretical analysis of concept learning. Vasconcelos [12], [13] analyzed the probabilistic image retrieval model based on mixture densities for the quality of the solution and computational complexity. However, the exploitation of meta knowledge is not considered for the model. Barnard and Forsyth [14] organized images (with associated words) by a hierarchical model for browsing and searching. In their work, some images are used to train the clustering directly; however, this training stage is unreliable since the training data set may not represent the image distribution of the entire database, especially when some images are added or removed during the database lifetime. Table I compares our concept learning approach presented in this paper with these two works. Our system distinguishes from them as it gradually improves concept learning based on multiple users' relevance feedback in the long term, instead of estimating the model only at one time.

In this paper, we model the database image distribution in feature space as a mixture of Gaussian densities [12], and our task is to estimate this model (called *model fitting*) to achieve concept learning. The task of concept learning is to explore the characteristics of the features that can represent a concept as

Manuscript received August 10, 2003; revised May 17, 2004. This work was supported in part under Grant F49620-02-1-0315 and Grant IIS-0-114-036. The contents of the information do not necessarily reflect the position or the policy of the U.S. Government.This paper was recommended by Guest Editor J. Peng.

The authors are with the Center for Research in Intelligent Systems, University of California, Riverside, CA 92521 USA (e-mail: adong@cris.ucr.edu; e-mail: bhanu@cris.ucr.edu).

Digital Object Identifier 10.1109/TSMCB.2005.846653

TABLE I
COMPARISON OF MIXTURE-MODEL-BASED RESEARCH FOR IMAGE DATABASES. THE SYMBOL $\oplus$ STRESSES THE ADVANTAGES OF OUR SYSTEM

Characteristics	**Vasconcelos [12][13]**	**Barnard and Forsyth [14]**	**Dong and Bhanu** (this paper)
• Database	images	images with words	images
• Dynamic	no	no	$\oplus$ yes
• Relevance Feedback	yes	no	yes
• Users' Labelling Noise during Relevance Feedback	no	N/A	$\oplus$ process and collect knowledge to deal with
• Model Selection	no	no	$\oplus$ automatic
• Mixture Parameter Estimation	standard EM	standard EM with training data initially	$\oplus$ user directed SS-EM algorithm

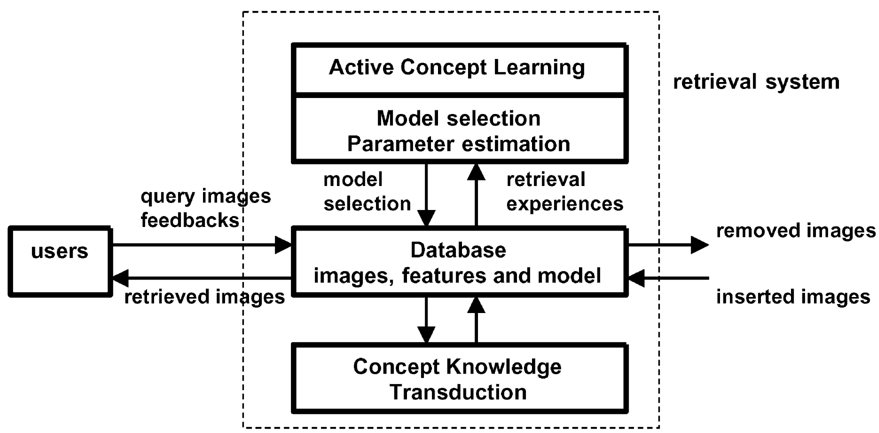


Fig. 1. System diagram with active concept learning and concept knowledge transduction for dynamic image databases.

perceived by various users. Specifically, for our Gaussian mixture model assumption, concept learning estimates the parameters (class prior probability, mean and covariance matrix) for the Gaussian corresponding to a concept. Fig. 1 illustrates our system framework for concept learning and transduction.

One of the major approaches to estimate (finite) mixture model (including Gaussian mixture model) is to use *expectation-maximization* (EM) algorithm to estimate mixture model parameters ([15]). However, this estimation is based on the availability of the number of components in the mixture model. Usually, the number of components is unknown, and we have to estimate this number (called *model selection*). Thus, the fitting task for mixture model generally consists of two steps: 1) model selection; and 2) parameter estimation. Model selection is the prerequisite for parameter estimation, and it is a more challenging and unsolved task. The integration of our system with relevance feedback mechanism makes it possible to carry out model selection in a semi-supervised manner, which is still an unexplored research topic to date.

The key contributions of this paper are: a) a new active learning approach for mixture model fitting is proposed. It includes a model selection method and a user directed semi-supervised EM (SS-EM) algorithm; b) the retrieval experiences derived from previous users' feedback are used to achieve concept learning, which may help to improve future retrieval performance; and c) An efficient concept knowledge transduction approach is presented to deal with the cases of image insertion and query images being outside the database. The approaches of [8], [9] are incapable of dealing with this situation.

This paper is organized as follows. Section II introduces the motivation of our work, including mixture model for image databases, related work on semi-supervised learning, and the key task: active concept learning. Section III presents the technical approach including the active concept learning achieved by our semi-supervised EM (SS-EM) algorithm (Section III-A), active model selection based on Bayesian analysis (Section III-D), the method to improve retrieval performance (Section III-B) and the concept knowledge transduction method (Section III-C). In Section IV, the experimental results on Corel database show the efficacy of our approach. Section V provides conclusions of the paper.

For the convenience of the readers, Table II provides the notations of the variables which will appear in this paper frequently.

II. MOTIVATION

A. Mixture Model for Image Databases

In this paper, we mainly consider those concepts which can be represented by some low-level visual features. For such a concept, its images may possess some similar visual features. These images form a cluster (in the sense of some visual feature similarity measurement) in the feature space of image databases. It is possible that a single concept may contain images which form multiple clusters in the feature space. Such concepts are at higher level, and we explore them in our concept learning approach.

In the feature space of an image database, the distribution of image feature vectors is assumed to be a c-component mixture

TABLE II
NOTATIONS OF KEY VARIABLES USED IN THE PAPER

VARIABLES	NOTATIONS
$\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$	the set of N image features in the image database
$\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_c\}$	the c components in mixture model for the feature space of the image database
$\{\pi_i, \mu_i, \Sigma_i\}, i = 1, 2, \ldots, c$	the proportion(π_i), mean (μ_i), and covariance (Σ_i) of Component i
$z_{ji}, j = 1, 2, \ldots, N, i = 1, 2, \ldots, c$	component-indicator: Image j is belonging to Component i (=1) or not (=0)
$\tau_{ji}, j = 1, 2, \ldots, N, i = 1, 2, \ldots, c$	the estimation of z_{ji}
$\mathcal{E} = \{\mathcal{X}^+, \mathcal{X}^-\}$	a retrieval experience
	($\mathcal{X}^+$ are positive-labelled images, $\mathcal{X}^-$ are negative-labelled images)
t $(= 0, 1, 2, \ldots)$	the system running time
λ_1 and λ_2	the possion parameters for users' queries (λ_1) and database changes (λ_2)
$r = \frac{\lambda_1}{\lambda_2}$	the relative occurrence rate of users' queries and database changes
$\mathcal{M}_c$	the assumed mixture model with c components
JC	Jaccard coefficient (to evaluate clustering results)

model $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_c\}$, whose probability density function (pdf) is

$$f(x) = \sum_{i=1}^{c} \pi_i f_i(x) \tag{1}$$

where x is d-dimensional feature data, $f_i(x)$ are component densities and π_i $(i = 1, 2, \ldots, c)$ are component proportions $(0 \leq \pi_i \leq 1$ and $\sum_{i=1}^{c} \pi_i = 1)$. The component densities are specified by parameter vectors $\theta = \{\theta_1, \theta_2, \ldots, \theta_c\}$. In the case of Gaussian mixture models, since the ith component pdf is specified by its mean μ_i and covariance $\Sigma_i, \theta_i = (\mu_i, \Sigma_i)$, i.e.,

$$f_i(x) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \times \exp\left[-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)\right] \tag{2}$$

for $i = 1, 2, \ldots, c$. d is the feature dimensionality. Let the vector Ψ contain all the unknown parameters in the mixture model, i.e.,

$$\Psi = (\pi_1, \ldots, \pi_{g-1}, \xi^T)^T \tag{3}$$

where ξ contains all the parameters in $\theta_1, \theta_2, \ldots, \theta_c$, and one of c component proportion π_g is arbitrarily omitted for the reason that the summation of π_i is one. So (1) can be rewritten as

$$f(x; \Psi) = \sum_{i=1}^{c} \pi_i f_i(x; \theta_i). \tag{4}$$

A complete review on mixture models can be found in [15]. Following the above definitions, for our image database system with N images, the visual feature vector is d-dimensional, and there are c concepts each of which is corresponding to one component. We also assume that each component is characterized by the Gaussian distribution. This assumption is consistent with the observation that in feature space, the images belonging to one concept form a cluster whose center attracts most of these images in its neighborhood while fewer images can be found with distance being further away from the cluster center. The different covariances of the c Gaussian components reflect the fact that different concepts relate to multiple visual features to different extents and some of the features may not be independent.

The task of concept learning is accomplished by fitting the mixture model, i.e., estimating the number of components c and the mixture model parameters Ψ. From the mathematical point of view, the mixture-based analysis defines a strict mathematical model, provides well-established statistical techniques and a test of the validity of cluster structure. Thus, the mixture modeling for image databases provides a powerful analysis tool for concept learning.

EM algorithm: We first assume that the number of components c is known by the system in advance. Given a set of of N independent and identical distribution (i.i.d.) samples $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$ from the model (4), the *maximum likelihood* (ML) estimate of the unknown parameter vectors θ_i can be obtained by the *Expectation-Maximization* (EM) approach. Set the associated binary component-indicator vectors for $\mathcal{X}$ as $\mathcal{Z} = \{z_1, z_2, \ldots, z_N\}$, where $z_j = (z_{j1}, \ldots, z_{jc})$ with

$$z_{ji} = \begin{cases} 1, & \text{if } x_j \text{ is from } i\text{th component} \\ 0, & \text{otherwise} \end{cases}$$

for $j = 1, 2, \ldots, N; i = 1, 2, \ldots, c$. The complete data log-likelihood function is given by

$$\log L(\mathcal{X}, \mathcal{Z}; \Psi) = \sum_{j=1}^{N} \sum_{i=1}^{c} z_{ji} \log[\pi_i f_i(x_j; \theta_i)]. \tag{5}$$

The EM algorithm produces a sequence of estimates $\{\hat{\Psi}(k), k = 0, 1, 2, \ldots\}$ by proceeding iteratively in two steps (E-step and M-step) until some termination criterion is met.

- **E-step**: Define the conditional expectation of $\mathcal{Z}$, whose elements are defined as $\tau_{ji} = E_{\hat{\Psi}(k)}(z_{ji} \mid \mathcal{X}) = \text{prob}_{\hat{\Psi}(k)}\{z_{ji} = 1 \mid \mathcal{X}\}$. By Bayesian theorem, it can be derived as

$$\tau_{ji} = \frac{\pi_i^{(k)} f_i\left(x_j; \hat{\theta}_i^{(k)}\right)}{\sum_{h=1}^{c} \pi_h^{(k)} f_h\left(x_j; \hat{\theta}_h^{(k)}\right)}. \tag{6}$$

- **M-step**: Update the estimate of Ψ by

$$\hat{\Psi}^{(k+1)} = \arg\max \Phi(\Psi; \hat{\Psi}^{(k)}).$$

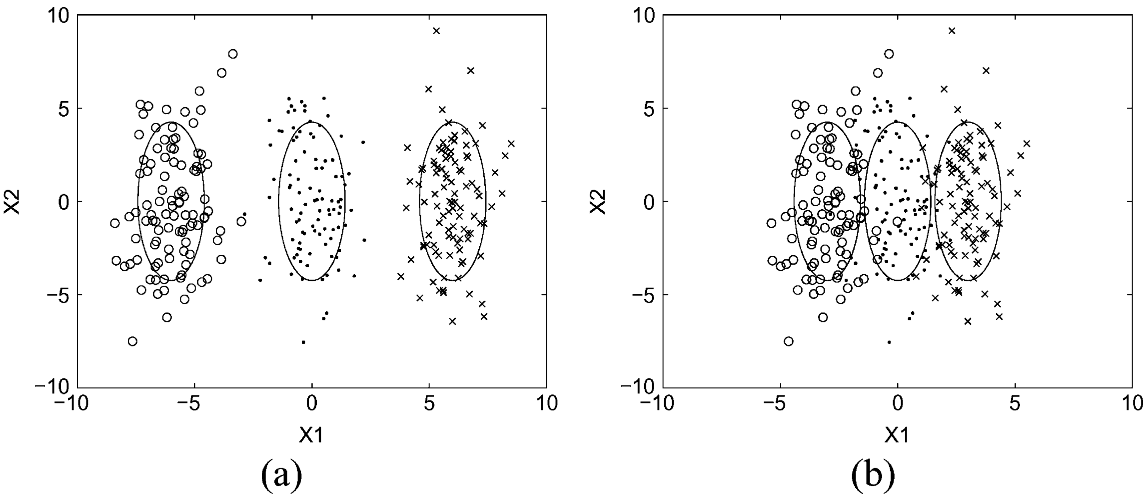


Fig. 2. Data for different 3-component mixture models. (a) Well-separated components. (b) Components with overlaps.

The updated expression for component proportions is

$$\pi_i^{(k+1)} = \frac{\sum_{j=1}^{N} \tau_{ji}^{(k)}}{N} \quad (i = 1, \ldots, c). \tag{7}$$

For Gaussian mixture models, the expressions for means and covariances are

$$\hat{\mu}_i^{(k+1)} = \left(\sum_{j=1}^{N} \tau_{ji}^{(k)}\right)^{-1} \sum_{j=1}^{N} x_j \tau_{ji}^{(k)} \tag{8}$$

$$\hat{\Sigma}_i^{(k+1)} = \left(\sum_{j=1}^{N} \tau_{ji}^{(k)}\right)^{-1} \sum_{j=1}^{N} \left(x_j - \hat{\mu}_i^{(k)}\right)\left(x_j - \hat{\mu}_i^{(k)}\right)^T \tau_{ji}^{(k)}. \tag{9}$$

If there exists one-to-one correspondence between the mixture components and the clustering groups, EM algorithm can be viewed as an effective model-based clustering approach. To partition the sample set $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$ into c groups, we need to find the associated binary component-indicator vectors $\mathcal{Z} = \{z_1, z_2, \ldots, z_N\}$ for these samples. In this sense, $\tau_{ji}(j = 1, 2, \ldots, N, i = 1, 2, \ldots, c)$, the conditional expectation of z_{ji}, represents the estimated posterior probability that the sample x_j belongs to ith component, and it is analogous to the membership element in the partition matrix as the result of fuzzy c-means (FCM) clustering [16].

We can derive hard (versus fuzzy) clustering in a way that is simply estimate of binary component-indicator vector z_j by $\hat{z}_j$ whose elements are defined as

$$\hat{z}_{ji} = \begin{cases} 1, & \text{if } i = \arg\max_{\{h=1,2,\ldots,c\}} \tau_{jh} \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

for $j = 1, 2, \ldots, N$ and $i = 1, 2, \ldots, c$.

Model selection: The above EM algorithm assumes that the number of components c is already given, and it is only *mixture parameter estimation*. Since c is usually unknown for the image retrieval system, *model selection* is necessary. In the traditional unsupervised cases, assuming that c is in the range of $\{c_{\min}, \ldots, c_{\max}\}$, it can be generally selected according to some criterion function by

$$\hat{c} = \arg\max_c \{\mathcal{F}(\hat{\Psi}_c, c), c \in \{c_{\min}, \ldots, c_{\max}\}\} \tag{11}$$

where $\hat{\Psi}_c$ is the mixture parameter estimation when the model is assumed to contain c components, and the criterion function $\mathcal{F}(\hat{\Psi}_c, c)$ usually consists of two terms as

$$\mathcal{F}(\hat{\Psi}_c, c) = -\log L(\mathcal{X}, \mathcal{Z}; \hat{\Psi}_c) + \mathcal{P}(c). \tag{12}$$

The first term is the log-likelihood of the data for the model, and the second term is to penalize higher values of c. Many researchers attempted to select model by using various criteria, such as *minimum description length* (MDL) criterion [17], *minimum message length criterion* (MML) [18], *Bayesian inference criterion* (BIC) [19], and Akaike's *information criterion* (AIC) [20].

B. Semi-Supervised Learning

The traditional clustering methods such as FCM are often frustrated by the fact the lower values of objective function do not necessarily lead to better data partitions. This actually reflects the gap between numeric-oriented feature data and concepts understood by humans. For a clustering task with regard to a mixture model, if different sets of feature data belonging to different clusters (concepts) are well separated, it is very likely that some clustering algorithm merely based on numeric feature data may yield a good partition, e.g., Fig. 2(a). However, if feature cannot represent human concept well enough (and this is common for pattern recognition tasks in the real world), it is difficult and sometimes impossible to rely on a pure feature-based clustering (unsupervised clustering) algorithm to obtain satisfactory clustering result. A typical example is shown in Fig. 2(b) where the three Gaussian components are so close that there exist overlaps between different components.. Similarly, the model selection methods solely based on numeric criteria are not reliable.

The semi-supervised fuzzy c-means (SS-FCM) clustering method attempts to overcome this limitation when the labels of some of the data are already known. Pedrycz *et al.* [21] modify the FCM objective function as the summation of two terms, which are related to the numeric features and labeled data respectively. However, only a heuristic scaling factor value is used to make balance between these two terms. In [22], the data labeling information is heuristically embedded into the *alternating optimization* (AO) process for the FCM algorithm. For example, in the partition matrix, only the elements related to unlabeled data are initialized and updated while the rest of the elements are determined by the labeling information and never updated during the AO process.

Recently, some papers on semi-supervised learning based on mixture models have been published. Wu and Huang [23] integrate *multiple discriminant analysis* (MDA) with EM framework so that the labeled data are enlarged by deriving "similar" samples from the unlabeled data. In this way, weak classifiers are boosted by exploring discriminant features in a self-supervised fashion. The nonlinear D-EM method based on kernel multiple discriminant analysis (KMDA) which is also proposed in [23] outperforms other supervised and semi-supervised learning algorithms for different visual learning tasks. Another approach dealing with labeled and unlabeled data from Gaussian mixture models [24], [25] is to modify the mixture log-likelihood function as the combination of two terms: the one for unlabeled data and the other for labeled ones.

The above mentioned semi-supervised learning approaches assume that the labeled data belong to some specified classes (clusters). In reality, another kind of labeling information that "some data do NOT belong to some classes (clusters)" is also available in many applications, and it may also help the learning. Also the above mentioned approaches assume that the labeling information is correct. Another lethal weakness of these approaches is that they are only for parameter estimation and do not deal with model selection. Thus, their practical applications are limited.

III. Technical Approach

In this section, we first present system events that are encountered with a relevance feedback mechanism (Section III-A), and then define active model fitting problem (Section III-B). This is followed by our semi-supervised EM (SS-EM) algorithm (Section III-C) for mixture parameter estimation with the assumption that the number of components is known in advance. With the concept learning knowledge, retrieval performance can be improved (Section III-D). Furthermore, a concept knowledge transduction approach is proposed to deal with the cases of image insertion and query images being outside the database (Section III-E). We then present the active model selection approach (Section III-F) by using Bayesian analysis based on the previous subsections. Finally, we analyze the computational complexity of our algorithm (Section III-G).

A. System Events

An image retrieval system with relevance feedback mechanism may encounter two kinds of events at any time during the long-term running: users' queries and database changes (i.e., image insertion or removal). We model the occurrences of these two events as *Poisson* random processes, whose distributions are $P[\mathcal{N}(t)=k]=((\lambda_i t)^k)/(k!)e^{-\lambda_i t}$ ($k=0,1,\ldots$, and t is the system running time) with $i=1$ and 2 respectively, and $\mathcal{N}(t)$ represents the number of the event occurrences within t. Note that λ_1 is the parameter for retrieval event, and λ_2 is the parameter for image insertion event. The ratio of the two distribution parameters $r=\lambda_1/\lambda_2$ specifies the relative occurrence rate of these two events.

Since different users make a variety of queries and perceive visual content differently, they may provide different sets of positive and negative labeling information, each of which is defined as a *retrieval experience* $\mathcal{E}=\{\mathcal{X}^+,\mathcal{X}^-\}$, where $\mathcal{X}^+=\{x_1^+,x_2^+,\ldots,x_{N^+}^+\}$ are labeled as belonging to (positive for) a certain but unknown concept while another portion of samples $\mathcal{X}^-=\{x_1^-,x_2^-,\ldots,x_{N^-}^-\}$ are labeled as NOT belonging to (negative for) that unknown concept. Note that $x_i^+(i=1,2,\ldots,N^+)$ and $x_j^-(j=1,2,\ldots,N^-)$ are visual image feature vectors.

B. Active Model Fitting

To accomplish the task of concept learning for the image database, model fitting needs to be performed over the data whose population is continually changing due to the dynamic nature of the database. Since retrieval experiences contain positive and negative labeling information, they can contribute to model fitting process; thus, our model fitting is done in a semi-supervised manner. Moreover, the fitting is updated actively with the acquisition of new retrieval experiences as images are added to or removed from the database.

Assume that the true number of components in the mixture model is known to be in the range of $\{c_{\min},\ldots,c_{\max}\}$. This range can be derived by a) implementing an unsupervised mixture model fitting on the data $\mathcal{X}$ (e.g., [18]); b) the *a priori* knowledge on this range; and c) the combination of a) and b). At time t of the system running, let the system has obtained $\iota(t)$ retrieval experiences and the current image data is $\mathcal{X}(t)$, our task is to find an optimal mixture model

$$\hat{\mathcal{M}}=\arg\max_{\mathcal{M}\in\Omega}\text{prob}\{\mathcal{M}\mid\mathcal{X}(t),\mathcal{E}_1,\ldots,\mathcal{E}_{\iota(t)}\}\tag{13}$$

where Ω is the search space containing all possible models in the range of $\{c_{\min},\ldots,c_{\max}\}$. The model $\mathcal{M}$ is specified by the number of components c and the parameter estimation Ψ_c (defined in Section II-A). Note that when $\iota(t)=0$, the problem becomes the traditional unsupervised mixture model fitting task in [18]–[20]. For convenience, we omit "(t)" in the notations for $\mathcal{X}(t)$ and $\iota(t)$ in the following text.

C. SS-EM Algorithm

1) Short-Term SS-EM Algorithm: We first consider learning with only a single retrieval experience $\mathcal{E}$ and extend it for multiple retrieval experiences in Section III-C2. We first assume that the component with regard to $\mathcal{E}$ is already known, and let it be the hth component with $h\in\{1,2,\ldots,c\}$. From the positive and negative labeling information, we already know some binary component-indicator vector values such that

$$z_{ji}=\begin{cases}1, & \text{if } i=h\\ 0, & \text{otherwise}\end{cases}$$

for $i = 1, 2, \ldots, c$ and $j \in \mathcal{J}^+$, and $z_{jh} = 0$ for $j \in \mathcal{J}^-$, where $\mathcal{J}^+ = \{j_1^+, j_2^+, \ldots, j_{N^+}^+\}$ and $\mathcal{J}^- = \{j_1^-, j_2^-, \ldots, j_{N^-}^-\}$ are the indices for the image in $\mathcal{X}^+$ and $\mathcal{X}^-$ respectively. Thus, we modify the data log-likelihood function (5) as

$$\begin{aligned}\log L(\mathcal{X}, \mathcal{Z}; \Psi) = &\sum_{j \in \mathcal{J}^u} \sum_{i=1}^{c} z_{ji} \log\{\pi_i f_i(x_j; \theta_i)\} \\ &+ \sum_{j \in \mathcal{J}^+} \log\{\pi_h f_h(x_j; \theta_h)\} \\ &+ \sum_{j \in \mathcal{J}^-} \sum_{i=1, i \neq h}^{c} z_{ji} \log\{\pi_i f_i(x_j; \theta_i)\}.\end{aligned}$$

In the above expression of the log-likelihood function, the first term is with regard to those unlabeled data $\mathcal{X}^u$, and it is in the same form as that in (5). The second term is for the positive labeled data $\mathcal{X}^+$ whose component-indicator vectors are already known so that there is no need to estimate them. For the negative labeled data $\mathcal{X}^-$, their component-indicator vectors are not totally available and only one of the elements in each vector is pre-determined to be zero. Thus, the indicator vectors for $\mathcal{X}^-$ have to be estimated, as demonstrated by the third term.

Based on the modified log-likelihood function, we can implement EM algorithm to estimate parameters in a similar manner as introduced in Section II-A. In E-step, for those pre-determined binary component-indicator vector elements, there is no need to estimate them, i.e., their estimation values are their "real" values, i.e.,

$$\tau_{ji} = \hat{z}_{ji} = \begin{cases} 1, & \text{if } i = h \\ 0, & \text{otherwise} \end{cases}$$

for $i = 1, 2, \ldots, c$ and $j \in \mathcal{J}^+$, and $\tau_{jh} = \hat{z}_{jh} = 0$ for $j \in \mathcal{J}^-$. For other unknown component-indicator vector elements, we have to estimate them, and their estimation expression is given by (6). In M-step, the component proportion estimation is the same as that by (7). For Gaussian mixture components, the estimations for means and covariances are given by (8) and (9), respectively. The result can be derived by using Lagrangian multipliers method to optimize the modified likelihood function, and we do not include it in this paper due to space limitation.

From the above analysis, in the case where the cluster index h is already known, the EM algorithm for this semi-supervised learning task is the same as the procedure introduced in Section II-A except that some component-indicator vector elements are pre-determined instead of being estimated.

If the cluster index h is unknown, we can first implement unsupervised EM algorithm on the data, and obtain the clustering result represented by the component-indicator estimations. Based on this initial clustering result, h can be derived from the positive and negative labeling information using a probabilistic method such that

$$h = \arg \max_{i=1,2,\ldots,c} \mathrm{Prob}(i) \tag{14}$$

where $\mathrm{Prob}(i)$ is equal to

$$\begin{aligned}&\mathrm{prob}(x_1^+ \in C_i, \ldots, x_{N^+}^+ \in C_i, x_1^- \notin C_i, \ldots, x_{N^-}^- \notin C_i) \\ &= \left\{\prod_{j=1}^{N^+} \mathrm{prob}(x_i^+ \in C_i)\right\} \left\{\prod_{j=1}^{N^-} \mathrm{prob}(x_j^- \notin C_i)\right\} \\ &= \left\{\prod_{j \in \mathcal{J}^+} \tau_{ji}\right\} \left\{\prod_{j \in \mathcal{J}^-} (1 - \tau_{ji})\right\}\end{aligned}$$

for $i = 1, 2, \ldots, c$. Note that the identification of h is dependent on the initial clustering result. One may argue that, if the initial clustering is not good enough, h may be misidentified so that the SS-EM algorithm is misled and its clustering result becomes worse. In the following text, we discuss this problem and give an efficient approach to overcome it.

2) Long-Term SS-EM Algorithm: The SS-EM method presented in Section III–C1 can be extended to the optimization problem with multiple retrieval experiences.

If some images are randomly selected from a single mixture component, it is possible that their covariance matrix is close to the covariance of the original component [12]. Unfortunately, the labeled images from a single retrieval are not sufficient; these images form a very small agglomeration in feature space compared to the size of a component. This requires that we refine the mixture model fitting continuously until enough experiences are accumulated.

In reality, users making queries on an image database do not always have enough patience to correctly label all the images presented to them by the system. More importantly, different users may ascribe the same image to different concepts. For an image with multiple opinions on its cluster ascription, it should belong to the cluster according to the opinion supported by the majority of the users.

In order to capture and accumulate previous users' retrieval experiences in the long-term history, we designate a *positive matrix* $P_{N \times c}$ and a *negative matrix* $Q_{N \times c}$ to represent this kind of knowledge. At the very beginning, when no retrieval has ever been executed on the system, P are Q are initialized to be zero matrices. After a retrieval experience, the elements $\{p_{j_1^+,h}, \ldots, p_{j_{N^+}^+,h}\}$ in P and the elements $\{q_{j_1^-,h}, \ldots, q_{j_{N^-}^-,h}\}$ in Q are increased by 1. Thus, the values of p_{jh} and q_{jh} represent to what extent people agree and disagree to ascribe an image j into the cluster h, respectively.

With the accumulated knowledge contained in P and Q, the component-indicator vector elements τ_{ji} derived in (6) can be modified as

$$\tilde{\tau}_{ji} = \begin{cases} \tau_{ji} + \frac{p_{ji} - q_{ji}}{\sum_{\iota=1}^{c}(p_{j\iota} + q_{j\iota})} & \text{if } p_{ji} > q_{ji} \\ 0 & \text{if } p_{ji} < q_{ji} \\ \tau_{ji} & \text{if } p_{ji} = q_{ji} \end{cases} \tag{15}$$

for $j = 1, 2, \ldots, N$ and $i = 1, 2, \ldots, c$. Then we normalize these modified component-indicator vectors so that $\sum_{i=1}^{c} \tilde{\tau}_{ji} = 1$. This means that, based on numeric feature data, the component-indicator estimation is modified with labeling knowledge derived from users' retrieval experience. This modification step is inserted between E-step and M-step so that the concept learning result is closer to human understanding. The

Algorithm 1 User directed semi-supervised EM (SS-EM) algorithm for long-term concept learning.

Given the data $\mathcal{X}$, the number of clusters c, the number of images N.
Initialize positive matrix $P_{N\times c}$ and negative matrix $Q_{N\times c}$ to be zero matrices.
Implement standard EM algorithm on $\mathcal{X}$.
repeat
1. Derive a retrieval experience, update P and Q.
2. E-step: Estimate $\mathcal{Z}$ by (6).
3. Use P and Q to modify estimation of $\mathcal{Z}$ by τ_{ji} (15).
4. M-step: Compute component proportions, means and covariances by (7), (8) and (9).
5. Go to 2 until termination criterion is met.

until the database finishes its lifetime.

user directed SS-EM algorithm is presented in Algorithm 1, shown at the top of the page. EM-algorithm for model estimation is computationally intense. To avoid clustering lagging behind retrieval experience derivation in the system, we implement user directed SS-EM algorithm after every $s(s \geq 1)$ retrieval experiences, where s is the update step.

This long-term SS-EM algorithm is the extension of short-term SS-EM information from a single retrieval experience while the former has to deal with multiple retrieval experiences. Due to the knowledge accumulation mechanism of matrices P and Q, the learning improvement of the system is guaranteed even though it is possible that the concepts being sought are occasionally misidentified by (14). For the same reason, in the case that users may mislabel images during relevance feedback, the system can still learn although the learning rate will be slower.

D. Improving Retrieval Performance

The knowledge of mixture model estimation derived from concept learning of the image database can help to improve retrieval performance. We use the component-indicator estimation $\tau_{ji}(j = 1, 2, \ldots, N; i = 1, 2, \ldots, c)$ to modify the image dissimilarity measurement for the initial search after a query is presented to the system, whose retrieval performance is the most important compared with the subsequent iterations. For the initial K nearest neighbor (K-NN) search, the Euclidean distance in the feature space from one database image $x_j(j = 1, 2, \ldots, N)$ to the query x_q is defined as $D(x_q, x_j)$, which we modify as

$$D'(x_q, x_j) = e^{-\frac{\beta n}{N}} D(x_q, x_j) - \sum_{i=1}^{c} \tau_{ji}\tau_{qi} \qquad (16)$$

where N is the database size and n is the number of retrieval experiences. The second term on the right side is with regard to concept learning knowledge, which is derived from

$$\begin{aligned} &\text{prob}\{\text{Query } q \text{ and Image } j \text{ belong to same class}\} \\ &= \sum_{i=1}^{c} \text{prob}\{q \in C_i \text{ and } j \in C_i\} \\ &= \sum_{i=1}^{c} \text{prob}\{q \in C_i\}\text{prob}\{j \in C_i\}. \end{aligned}$$

As the concept learning is improved with the retrieval experiences increased, the second term in (16) should be given more credit. The parameter β is to make balance between these two terms. Note the speed of learning improvement depends on the database size N.

E. Concept Knowledge Transduction

When a new image is inserted, the database size N is increased by 1, and the positive matrix $P_{N\times c}$ and the negative matrix $Q_{N\times c}$ are both modified with one additional row, whose elements are all zero. The component-indicator estimation of this new image can be computed by (6) with $j = N + 1$ and component proportions, means and covariances are already known. In this way, the database absorbs the new image with concept knowledge transduction. When some images are removed from the database, the corresponding rows in the matrices $P_{N\times c}$ and $Q_{N\times c}$ are deleted. The relationship between the rates of new user query processing and database changes (addition and deletion of images) influence the speed of concept learning. When database changes occur more frequently compared to the event of new user queries, i.e., when the value of relative occurrence rate r is lower, the concept learning becomes slower.

If the query image does not belong to the database, the system extracts its visual features, computes $\tau_{qi}(i = 1, 2, \ldots, c)$ by (6), and implements K-NN search using the distance metric given in (16). Compared with the traditional K-NN search that is solely based on visual feature Euclidean distance measurement, this approach yields better retrieval performance since concept knowledge is adopted.

It is possible that some newly inserted images do not belong to any one of the present components (classes). These images are outliers for the existing components (or clusters), and outliers may mislead the clustering and lower the system performance. There has been extensive research on robust clustering [26], which can detect outliers and be insensitive to their misleading effects. The component of such outlier detection can be integrated into the EM algorithm, so that our system may deal with the existence of some images which are not belonging to any component. In this paper, since our focus is to present the semi-supervised concept learning based on mixture model assumption, we assume that any database image belongs to one of the present components. However, in our future work, we will integrate robust clustering techniques into our system to develop a unified framework.

F. Model Selection

As presented in Section III-A, if we initially select the model with the true number of components, the SS-EM algorithm with increased retrieval experiences will lead to the model estimation which is very close to the ground-truth model. On the other hand, if the initial assumption on the number of components is not correct, the fitting over time will never yield a good model estimation (at least it will be worse than the case based on the true number of components after enough time).

The previous retrieval experiences obtained by the system can help to select the optimal model. An obvious way to achieve this is to keep all the candidate model fittings with $\{\Psi_{c_{\min}}, \ldots, \Psi_{c_{\max}}\}$. When a new retrieval experience is obtained, the system updates all of these fittings by SS-EM algorithm. The optimal model is the one which is most consistent with the feature data and the positive and negative information contained in the previous retrieval experiences.

However, when the range of $\{c_{\min}, \ldots, c_{\max}\}$ is large, and the computational load of EM algorithm is heavy, the updating for all the candidate models will be very slow; thus, the model fitting may lag far behind the available retrieval experience, i.e., the system cannot digest the retrieval experiences on time.

To overcome this computational load problem, we propose an adaptive model selection approach. Assume that the system allows the computation of EM algorithms for $\mathcal{K}(1 \leq \mathcal{K} < c_{\max} - c_{\min} + 1)$ models at the same time ($\mathcal{K}$ is determined by the capability of the available computational resources), we select $\mathcal{K}$ out of $c_{\max} - c_{\min} + 1$ models based on their consistencies with the data and the previous retrieval experiences. The consistency of a model $\mathcal{M}_c$ with ι retrieval experiences can be measured by its probability given the data and these retrieval experiences

$$\begin{aligned}
&\mathrm{prob}(\mathcal{M}_c \mid \mathcal{E}_1, \ldots, \mathcal{E}_\iota; \mathcal{X}) \\
&\quad= \frac{\mathrm{prob}(\mathcal{E}_1, \ldots, \mathcal{E}_\iota \mid \mathcal{M}_c; \mathcal{X})\mathrm{prob}(\mathcal{M}_c \mid \mathcal{X})}{\mathrm{prob}(\mathcal{E}_1, \ldots, \mathcal{E}_\iota \mid \mathcal{X})} \\
&\quad\propto \mathrm{prob}(\mathcal{E}_1, \ldots, \mathcal{E}_\iota \mid \mathcal{M}_c; \mathcal{X})\mathrm{prob}(\mathcal{M}_c \mid \mathcal{X}) \qquad (17)\\
&\quad= \left\{\prod_{i=1}^{\iota} \mathrm{prob}(\mathcal{E}_i \mid \mathcal{M}_c; \mathcal{X})\right\} \mathrm{prob}(\mathcal{M}_c \mid \mathcal{X}). \qquad (18)
\end{aligned}$$

The condition for independency required for the deduction from (17) to (18) will be discussed later in this section. For a single retrieval experience $\mathcal{E}$, we have

$$\begin{aligned}
&\mathrm{prob}(\mathcal{E} \mid \mathcal{M}_c; \mathcal{X}) \\
&\quad= \sum_{i=1}^{c} \left\{\prod_{j=1}^{N^+} \mathrm{prob}(x_j^+ \in C_i) \prod_{j=1}^{N^-} \mathrm{prob}(x_j^- \notin C_i)\right\} \\
&\quad= \sum_{i=1}^{c} \left\{\prod_{j\in\mathcal{J}^+} \tau_{ji} \prod_{j\in\mathcal{J}^-} (1 - \tau_{ji})\right\} \qquad (19)
\end{aligned}$$

where $\mathcal{J}^+$ and $\mathcal{J}^-$ are the indices for the image in $\mathcal{X}^+$ and $\mathcal{X}^-$ respectively, and τ_{ji} is conditional expectation that the image x_j belongs to C_i as defined in Section II-A.

The term $\mathrm{prob}(\mathcal{M}_c \mid \mathcal{X})$ appeared in (18) can be derived by the likelihood function in the unsupervised learning proposed in [27],

$$\mathcal{L}(\Psi_c, \mathcal{X}) = \log p(\mathcal{X} \mid \Psi_c) + \log \sum_{i=1}^{c} \sum_{j=1}^{N} \tau_{ji} \log \tau_{ji} \qquad (20)$$

where the second term is the estimated entropy used to penalize the model for its complexity (high value of c). Since there are $c_{\max} - c_{\min} + 1$ candidate models, we approximate the models' probabilities as

$$\mathrm{prob}(\mathcal{M}_c \mid \mathcal{X}) \simeq \frac{\exp\{\mathcal{L}(\Psi_c, \mathcal{X})/\mathcal{L}_{\max}\}}{\sum_{k=c_{\min}}^{c_{\max}} \exp\{\mathcal{L}(\Psi_k, \mathcal{X})/\mathcal{L}_{\max}\}} \qquad (21)$$

where $\mathcal{L}_{\max} = \max_{c\in\{c_{\min},\ldots,c_{\max}\}} |\mathcal{L}(\Psi_c, \mathcal{X})|$, which is used to normalize the likelihood functions.

We define $\mathcal{M}_c$'s consistency $\mathrm{cons}(\mathcal{M}_c; \mathcal{E}_1, \ldots, \mathcal{E}_\iota, \mathcal{X})$ with the data $\mathcal{X}$ and ι retrieval experiences as the log-based value of (18)

$$\begin{aligned}
&\mathrm{cons}(\mathcal{M}_c; \mathcal{E}_1, \ldots, \mathcal{E}_\iota, \mathcal{X}) \\
&\quad= \sum_{i=1}^{\iota} \log \mathrm{prob}(\mathcal{E}_i \mid \mathcal{M}_c; \mathcal{X}) + \log \mathrm{prob}(\mathcal{M}_c \mid \mathcal{X}) \qquad (22)
\end{aligned}$$

which appropriately represents the probability $\mathrm{prob}(\mathcal{M}_c \mid \mathcal{E}_1, \ldots, \mathcal{E}_\iota; \mathcal{X})$. By (19), (20), and (21), the value of (22) can be computed. Note that when $\iota = 0$ (no retrieval experience), the consistency only depends on the second term of (22), which is derived from the model likelihood function for unsupervised learning. This is the case at the very beginning of the system running. With retrieval experiences increased, the unsupervised criterion represented by the second term of (22) exerts less influence while the accumulated retrieval experiences plays a more important role on the consistency measurement. Thus, the current optimal model is

$$\hat{c}_{\mathrm{opt}} = \arg \max_{c\in\{c_{\min},\ldots,c_{\max}\}} \mathrm{cons}(\mathcal{M}_c; \mathcal{E}_1, \ldots, \mathcal{E}_\iota, \mathcal{X}). \qquad (23)$$

The selection of $\mathcal{K}$ out of $c_{\max} - c_{\min} + 1$ models for SS-EM updating is based on the models' current consistencies. However, we do not directly choose the $\mathcal{K}$ models which have highest consistency values; instead, we give an opportunity of being selected to each of the $c_{\max} - c_{\min} + 1$ models, whose probability is

$$\mathrm{prob}(\mathcal{M}_c) = \frac{\exp\{\eta t \cdot \mathrm{cons}(\mathcal{M}_c; \mathcal{E}_1, \ldots, \mathcal{E}_\iota, \mathcal{X})\}}{\sum_{i=c_{\min}}^{c_{\max}} \exp\{\eta t \cdot \mathrm{cons}(\mathcal{M}_i; \mathcal{E}_1, \ldots, \mathcal{E}_\iota, \mathcal{X})\}} \qquad (24)$$

where η is a parameter to be discussed later. This equation implies that models with higher consistency values have better chances of being selected. Thus, the search direction for optimal model tends to be toward the models which have good consistencies with the feature data and the accumulated retrieval experiences, i.e., the optimal model search should *exploit* the current model estimation. On the other hand, due to the possibility that the model estimation based on the true number of components may not be good (especially at the early stage of the database), we allow the models with lower consistency values, i.e., we want to *explore* the whole search space. The relationship between *exploitation* and *exploration* changes with time t: at the early stage, *exploration* is more important so that all of the candidate models have good chances of being selected for updating with retrieval experiences. With more retrieval experiences improving these model estimations, *exploitation* becomes the main concern since the model with good consistency is very likely to

Algorithm 2 Active model selection algorithm.

- $t = 0$. Given the data $\mathcal{X}(t)$, the range for the number of components $\{c_{min}, \dots, c_{max}\}$.
- Implement standard EM algorithm on $\mathcal{X}(t)$ for $c_{min}, \dots, c_{max}$, and get parameter estimations $\Psi_{c_{min}}, \dots, \Psi_{c_{max}}$.

repeat
  $t = t + 1$.
  if new user executes retrieval and get retrieval experience $\mathcal{E}_t$ **then**
    1. $\mathcal{X}(t) \leftarrow \mathcal{X}(t-1)$.
    2. Count $\mathcal{E}_t$ for consistency measurement if the independence condition is satisfied; otherwise, go to "repeat."
    3. Compute consistencies for $\mathcal{M}_{c_{min}}, \dots, \mathcal{M}_{c_{max}}$ by (22), and normalize by dividing them with $\mathcal{S} + 1$.
    4. Select $\mathcal{M}_{\hat{c}_{opt}}$ as current optimal model by (23).
    5. Randomly select $\mathcal{K}$ models for SS-EM updating by the probabilities given in (24) and get $\mathcal{K}$ updated parameter estimations. Prevent component-indicator elements to be 0 by (25).
  else if Image I^+ is added **then**
    $\mathcal{X}(t) \leftarrow \mathcal{X}(t-1) \bigcup I^+$.
  else if Image I^- removed **then**
    $\mathcal{X}(t) \leftarrow \mathcal{X}(t-1) - I^-$.
  end if
until the database finishes its lifetime

be the optimal model. The term ηt in (24) assigns the probabilities for models being selected for SS-EM updating in the way that reflects this *exploitation* and *exploration* relationship.

The independency condition for the deduction from (17) to (18) may not be satisfied if all the obtained retrieval experiences are used to measure model consistency. For example, if two retrieval experiences with the same image sets (they are not independent) are considered in the computation of (18) to measure model consistency, the positive and negative information contained in these two experiences may be overly used; thus, the derived consistency may not correctly reflect whether or not the model is good. To avoid this problem, we select retrieval experiences such that each pair has no overlap, i.e., no common images. However, this condition is too restrictive for the system to achieve enough retrieval experiences for consistency measurement. Thus, we allow each pair of selected retrieval experiences to share a small percentage of common images. We set an independency threshold ϱ such that the maximal allowed number of common images between two retrieval experiences is ϱl (l is the number of images that are presented to the user at each relevance feedback iteration). Thus, enough retrieval experiences can be accumulated and the condition for independency is not invalidated. Another advantage of this experience selection method is that it alleviates the load of computation for consistency; otherwise, the retrieval experiences used for consistency computation may tend toward infinity with time t increased.

For each retrieval experience $\mathcal{E}$, to compute $\mathrm{prob}(\mathcal{E} \mid \mathcal{M}_c; \mathcal{X})$, there are $(\bar{l} - 1)$ multiplications by (19) with $\bar{l}$ being the average size of $\mathcal{E}$. Thus, with ι retrieval experiences, the computation complexity of the consistency for each model is $\mathcal{O}(\iota\bar{l})$ by (22). For all of the candidate models, the consistency computation complexity is $\mathcal{O}((c_{\max} - c_{\min} + 1)\iota\bar{l})$. Since the retrieval experiences are selected to compute the model consistency value to guarantee the independency condition, ι cannot go to infinity; instead, its upper-bound is $(N/\varrho\bar{l})$, where N is the database size and $\varrho\bar{l}$ is the average number of common images allowed for different retrieval experiences for consistency computation.

For consistency computation, there is an overflow problem that cannot be ignored. For (19) whose computation involves the multiplications of many probabilities, if there exist some probabilities whose values are zero or very small, $\mathrm{prob}(\mathcal{E} \mid \mathcal{M}_c; \mathcal{X})$ is zero. This will cause overflow problem for computing log-based value in (22). To avoid this overflow problem, we adopt a small value threshold ε to prevent the probabilities from being too small in the way that

$$\tau_{ji} = \frac{\tau_{ji} + \varepsilon}{(1 + c\varepsilon)} \tag{25}$$

for $j = 1, 2, \dots, N; i = 1, 2, \dots, c$. We use $\varepsilon = 0.01$ in this paper.

From (22), we observe that consistency value becomes smaller with ι increased (more retrieval experiences). Although this does not influence the ranks of the model consistencies at time t, it influences the probabilities for the models being selected for updating as shown in (24). By setting the summation of the image numbers in all the ι retrieval experiences as $\mathcal{S}$, the model consistency in (22) can be regarded as the summation of $\mathcal{S} + 1$ log-based probabilities. Thus, we normalize the consistency by dividing the expression in (22) with $\mathcal{S} + 1$. Algorithm 2, shown at the top of the page, summarizes our algorithm for the active model selection.

G. Computational Complexity

To estimate the parameters of each candidate model, the EM algorithm is executed. The major computational load of EM is for the computation of the inverse matrices of component covariances, whose time-complexity is $\mathcal{O}(kcd^3)$ where k is the number of iterations for the EM algorithm, c is the number of components, and d is the feature dimensionality. Thus, the computational complexity for all the candidate models is $\mathcal{O}((c_{\max} - c_{\min} + 1) * (c_{\max} + c_{\min})kcd^3)$. Obviously, the high feature dimensionality d will make the EM computation very slow. Fortunately, this EM computation is performed off-line, and the batch mode can make the concept learning avoid too much lag behind the arrival of retrieval experiences. The EM computational load may be alleviated by using feature reduction techniques such as the discriminant-EM (D-EM) [23]. Furthermore, the EM computation can be accelerated by the techniques such as the partial E-step method [28]. We will integrate these techniques into our future work when we extend our system to deal with very large databases.

Fig. 3. Sample images of the 12 classes in the database obtained from Corel stock photo library.

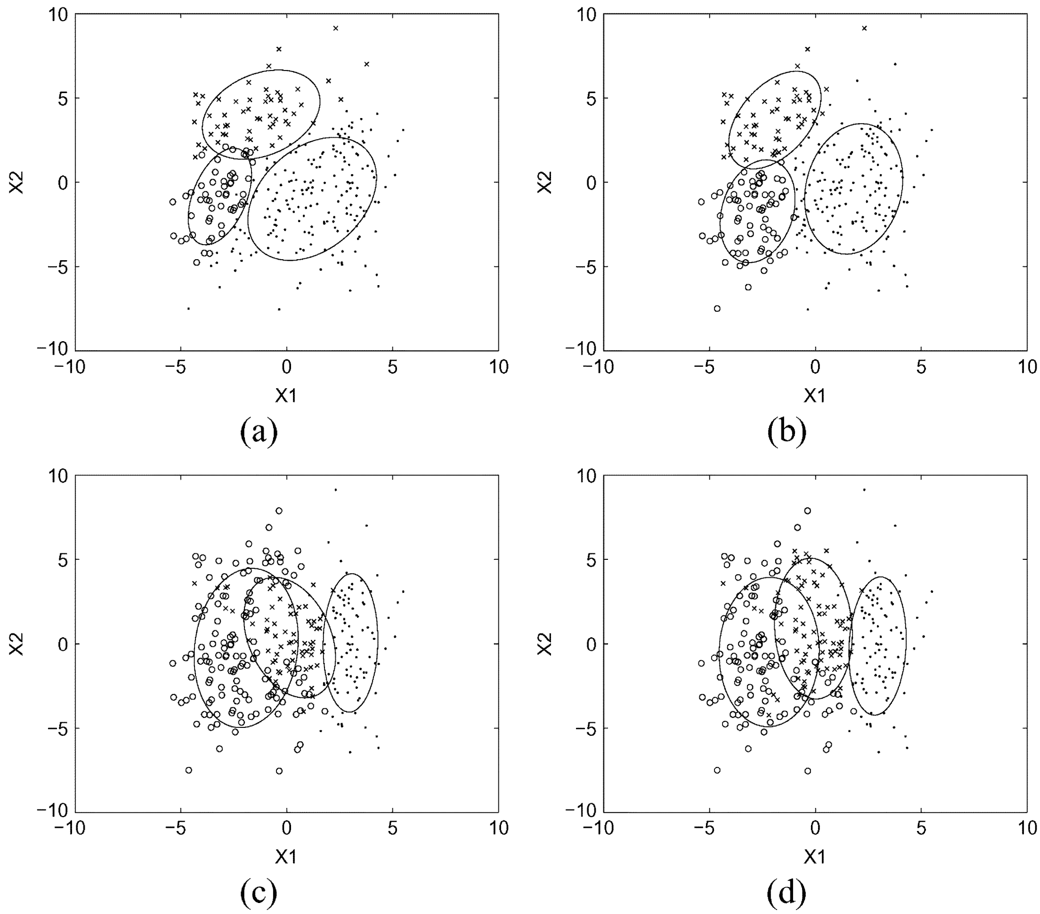


Fig. 4. Clustering result of SS-EM algorithm: (a) 0 users, $\mathrm{JC} = 60.5\%$. (b) 5 users, $\mathrm{JC} = 69.8\%$. (c) 15 users, $\mathrm{JC} = 81.5\%$. (d) 35 users, $\mathrm{JC} = 90.1\%$.

IV. Experiments

To validate the proposed active concept learning approach, we design three experiments all of which are based on *Corel* images.[1] First, we show the efficacy of our semi-supervised EM algorithm (Section III-C), which is the foundation of the active concept learning approach. Second, we implement concept knowledge transduction method on a dynamic image database, and explore the influence of the dynamic nature of the database on concept learning. Third, we implement the active concept learning algorithm with the number of components being unknown and evaluate its performance.

We construct an image database with 1200 images, which are selected from Corel stock photo library and divided into 12 classes. These classes are corresponding to the CDs (series number) in the library: *Mayan & Aztec Ruins* (33), *Horses* (113), *Owls* (75), *Sunrises & Sunsets* (1), *North American Wildflowers* (127), *Ski Scenes* (61, 62), *Coasts* (5), *Auto Racing* (21), *Firework Photography* (73), *Divers & Diving* (156), *Land of the Pyramids* (161), and *Lions* (105). We remove some images from these CDs since they do not have good visual features to represent the corresponding concepts, and we add some images from other CDs to some of the 12 classes. Fig. 3 shows sample images for all of the 12 concepts. We use texture and color features to represent images. The texture features are derived from 16 Gabor filters [5]. We extract means and standard deviations from the three channels in HSV color space. Thus, each image is represented by 22 features.

The concept learning result is quantified by clustering validation. To validate a clustering result $\mathcal{R} = \{\mathcal{R}_1, \ldots, \mathcal{R}_{\hat{c}}\}$ from an algorithm, we compare $\mathcal{R}$ with the ground-truth mixture model $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_c\}$ by using *Jaccard coefficient* [29]. A pair of

[1][Online] Available: http://www.corel.com.

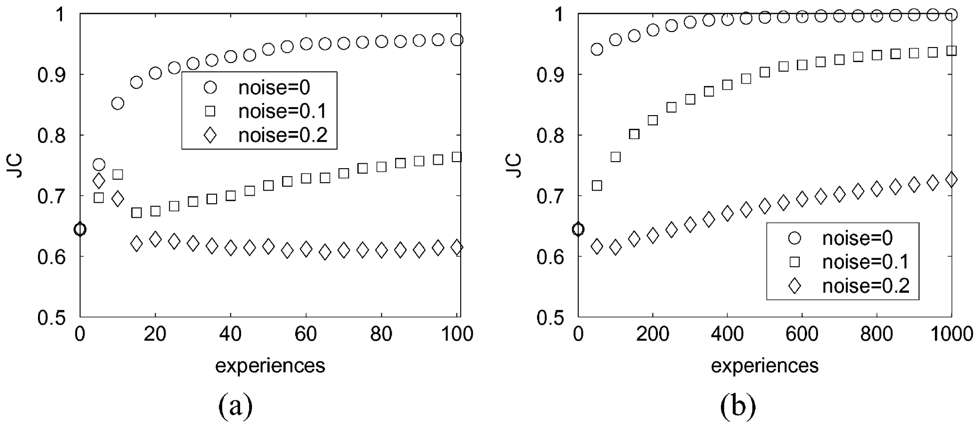


Fig. 5. Synthetic data: concept learning is improved with increased retrieval experiences. (a) Initial stage. (b) Long term.

vectors $\{x_i, x_j\}$ are referred to as a) BS if both vectors belong to the same component in $\mathcal{C}$ and to the same cluster in $\mathcal{R}$; b) SD if both vectors belong to the same component in $\mathcal{C}$ and to different clusters in $\mathcal{R}$; and c) DS if both vectors belong to different components in $\mathcal{C}$ and to the same cluster in $\mathcal{R}$. Let ξ_1, ξ_2, and ξ_3 be the number of BS, SD, and DS pairs of vectors of $\mathcal{X}$, respectively. Jaccard coefficient is defined as $\mathrm{JC} = (\xi_1)/(\xi_1 + \xi_2 + \xi_3)$ and is used to evaluate clustering result. One of the advantages of adopting Jaccard coefficient is that it can evaluate a clustering result of an algorithm whose cluster number is not necessarily the true number of clusters.

In the experiments, we simulate the process of a retrieval system for which queries are selected randomly from the images in the database. We use the *probabilistic feature relevance learning* (PFRL) approach [5] for relevance learning. Let the number of images the user is presented at each relevance feedback iteration l be 20.

A. Experiments for SS-EM Algorithm

1) Synthetic Data: To help the reader to understand the theory of SS-EM, we present an experiment on synthetic data. Fig. 2(b) shows three Gaussian components whose data are difficult to partition correctly with an unsupervised learning method. Each component generates 100 patterns. The component centers are $[-3\ 0]^T, [0\ 0]^T$, and $[3\ 0]^T$ respectively, and their covariance matrices are all $[1\ 0;\ 0\ 9]^T$. We implement our SS-EM algorithm on this synthetic data with $c = 3, N = 300$ and $s = 5$. Simulating the system with increased experiences, we randomly select a data as the query for each retrieval, and identify the concept (component) that is sought by the current user. An example of this process is shown in Fig. 4(a)–(d), in which the clustering result is improved with the number of increased experiences.

Fig. 5 shows that, with different mislabeling noise ratio ν (the probability that the user mislabels an image), the average Jaccard coefficient is increased with increased retrieval experiences in the long term. The average is computed over all the retrieval sessions with all of the images in the database being simulated as query images. When there is no noise, the clustering result immediately converges to the ground-truth component distribution after the initial few ($\sim$200) experiences. The learning convergence is slower with noise ratio ν increased. If noise exists, JC value of the initial few experiences may become lower since the noise misleads the learning; then JC value increases monotonically since our knowledge accumulation mechanism can overcome the noise.

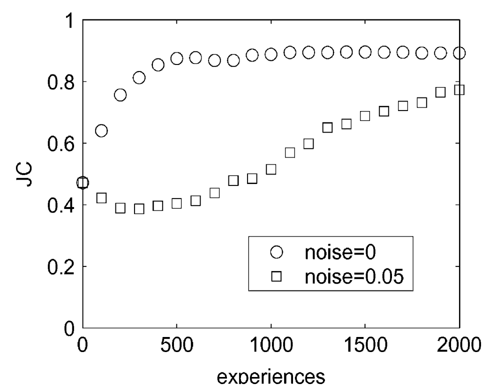
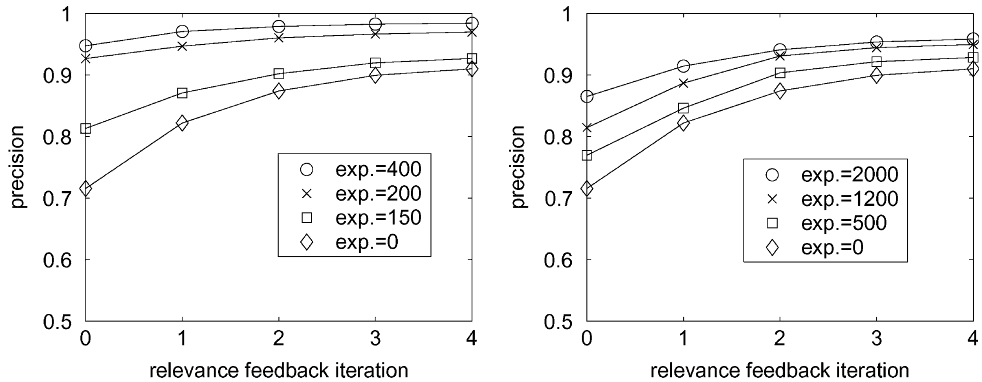


Fig. 6. Real data: concept learning is improved with increased retrieval experiences with and without noise.

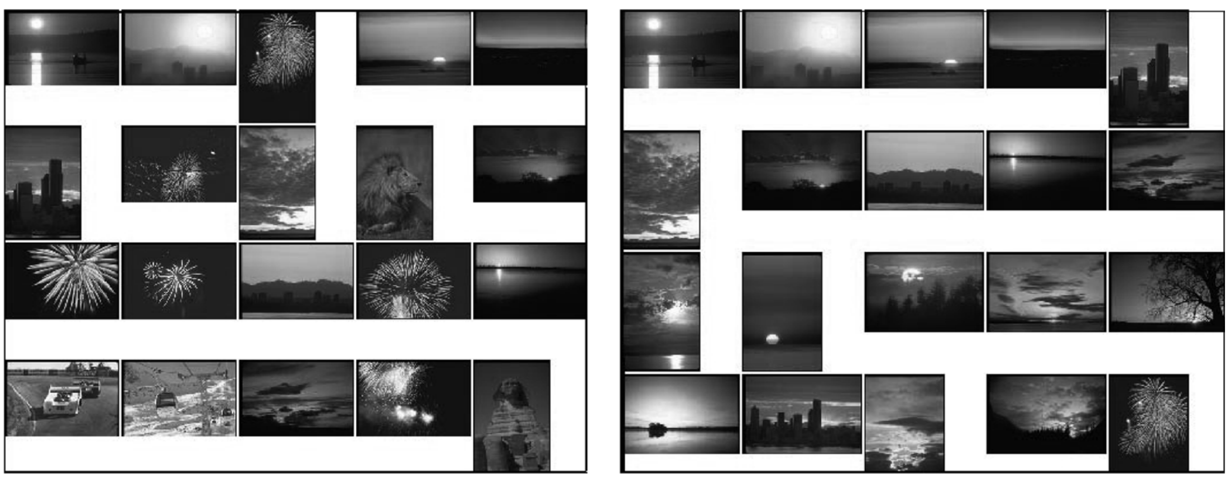
2) Real Data: We implement our concept learning approach on this database with $c = 12, N = 1200, s = 50$, and $\beta = 100$. Initially, the Jaccard coefficient by standard EM algorithm is 47.1%. Fig. 6 shows that the average Jaccard coefficient is increased with increased retrieval experiences in the long term. Compared with the synthetic data, the concept learning improvement is slower due to the fact that there are more components for real data and components need more data samples for higher dimensional features. When the mislabeling noise ratio ν is 0.05, the learning also converges in the long term although the improvement speed is slower. Note the probability that the user mislabels any image at a single relevance feedback iteration is $\mathrm{prob(error)} = 1 - (1-\nu)^l$ with l being the number of images presented to the user at each iteration. Thus, when $\nu = 0.05$ and $l = 20$, prob(error) is 0.64, which is quite high.

For unsupervised learning, it is usually impossible to achieve satisfactory estimation for mixture model based on such limited number of samples in the high-dimensional feature space. This is also validated by the experimental result in Fig. 6: when $t = 0$ (without retrieval experience), the mixture model estimation is not good, i.e., the value of JC is low. However, with more labeling information obtained from retrieval experiences, the estimation becomes better, even without using any feature reduction



(a) (b)

Fig. 7. Real data: retrieval performance with various amounts of retrieval experiences. (a) No noise. (b) Noise ratio $\nu = 0.05$.



(a) (b)

Fig. 8. Retrieval results for the same query (the first image) with different retrieval experiences. The user is looking for *sunset* images. (a) No retrieval experience: precision $= 10/20$. (b) After 200 retrieval experiences: precision $= 19/20$.

technique. This has also been demonstrated in Fig. 6 where as t increases, the mixture model estimation is improved. This reflects the advantage of our semi-supervised concept learning approach. Even with a limited number of data samples in high-dimensional feature space, the exploitation of labeling information may still achieve good mixture model estimation which is impossible for the unsupervised learning.

Fig. 7 presents the retrieval performances with different amounts of retrieval experiences with and without labeling noise. Again, the precision is for the top 20 images. The *retrieval precision* is defined as the percentage of positive retrievals out of the total retrievals. We select an image in this database as the query, implement our retrieval strategy, and repeat this experiment by changing query until each of the 1200 images has been selected as a query. Then we calculate the average precision at each iteration. With increased retrieval experiences, the average precision is improved, especially at initial K-NN search iteration. This is important for retrieval performance in practical applications: although the retrieval experiences exploited by the system are obtained from previous users' relevance feedbacks, the system may present good retrieval results to future users directly even without executing relevance feedbacks.

Fig. 8 shows two different retrieval results with the same query image with different retrieval experiences where there is no labeling noise. In Fig. 8(a), there is no retrieval experience, and K-NN search yields only 10 out of 20 *sunset* images (row 1: image 1, 2, 4, 5; row 2: 1, 3, 5; row 3: 3, 5; row 4: image 3). In Fig. 8(b), after 200 retrieval experiences, 19 *sunset* images are presented (except the last one) by our approach.

B. Experiment for Concept Knowledge Transduction

We randomly select 800 out of the 1200 images as the initial database images, i.e., $N = 800$, and insert the other 400 images while the system is running. Our concept learning approach on the database is implemented with $c = 12$, $s = 50$ and $\beta = 100$. We set the system running time as $t = 0, 1, 2, \ldots$; at each t, one of the two events happens: user presents a new query or inserts a new image. This is a random process derived from the two events' poisson random processes with their relative occurrence rate r as defined in Section III-C. Queries from the database and images to be inserted are randomly selected. We try different values of r to study the improvement of concept learning and retrieval performance. Fig. 9(a) shows the concept learning improvement. Initially, the Jaccard coefficient by standard EM algorithm is 45.6%. If there are only image insertions in the

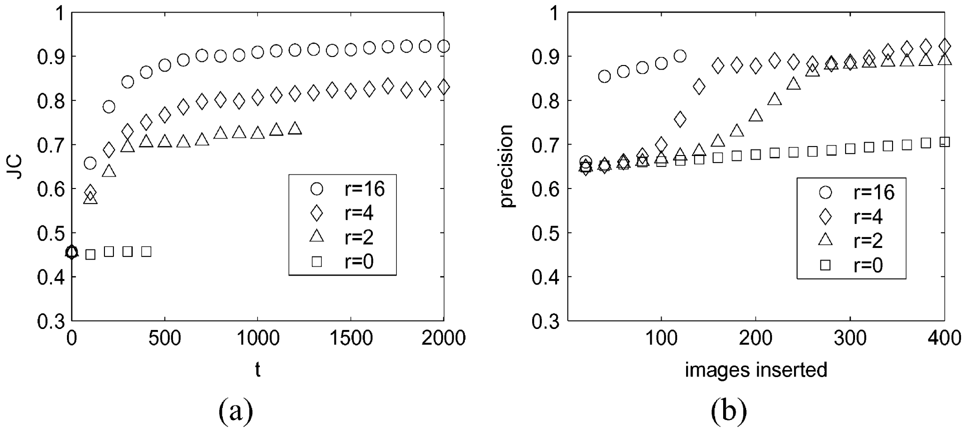


Fig. 9. Performances with different values of relative event rates r. (a) Concept learning. (b) Retrieval precision.

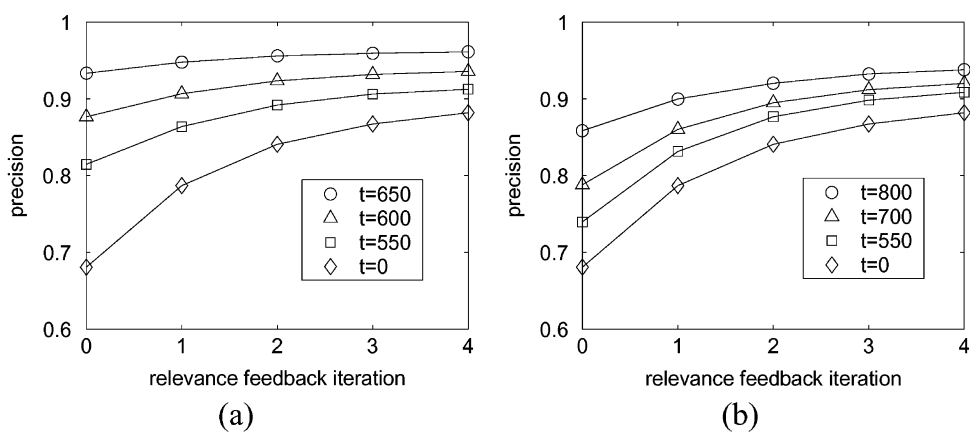


Fig. 10. Retrieval performance with various amounts of time. (a) $r = 16$. (b) $r = 2$.

random process, i.e., $r = 0$, the concept learning cannot be improved. When the users' queries happen more frequently, i.e., the value of r is higher, the concept learning will be faster. Note that when $r = 0$ and $r = 2$, after all the 400 images are inserted into the system, t is around 400 and $(400 + 2 * 400) = 1200$ respectively, and it cannot reach $t = 2000$.

After an image is inserted, we use the rest of the images outside the database as queries and implement concept transduction method to compute the retrieval precision (at relevance feedback iteration 0) by (16). As shown in Fig. 9(b), the precision increases with more images being inserted to the system due to the reason that concept learning is improved with more retrieval experiences. Note that another factor that improves the precision is that with more images being inserted, there are more relevant images within each class; thus, the probability increases to find more relevant images for a given query image. This also explains that when $r = 0$, the precision is slightly higher with more images being inserted although concept learning is not improved. Since we only observe the process with t being from 0 to 2000, when $r = 16$, only around $2000/(1+16) = 117$ images are inserted. This is the reason that the curve for $r = 16$ cannot reach 400 for the value on the x axis.

Fig. 10 presents the retrieval performance improvement with increased running time for $r = 16$ and $r = 2$. We select an image in this database as the query, implement our retrieval strategy, and repeat this experiment by changing query until each of the database images has been selected as a query. Then we calculate the average precision at each iteration. With increased retrieval experiences, the average precision is improved, especially at initial K-NN search iteration. This is important for retrieval performance in practical applications since users usually do not have enough patience to repeat relevance feedback iterations to search the images.

Fig. 11 shows two different retrieval results with the same query image (outside the database) after different running time. In Fig. 11(a), there is no retrieval experience, and K-NN search only yields 11 out of 20 *sunset* images (row 1: all the 5 images; row 2: 1, 2, 4; row 3: 5; row 4: 1 and 5). In Fig. 11(b), when $t = 300$, 19 *sunset* images are presented (except the 3rd image on the last row) by our concept transduction approach.

C. Experiments for Active Concept Learning

To validate the model selection approach proposed in Section III-F, we use the dynamic database similar to that in Section IV-B. Let the relative event occurrence rate $r = 16$. Our active concept learning approach with model selection on the database is implemented with $c_{\min} = 10, c_{\max} = 13$, the number of models selected for SS-EM updating $\mathcal{K} = 1$, the independency threshold $\varrho = 0.25$, the update step is $s = 10$ and $\beta = 100$ [see (16)].

At the initial stage of the system running, since the models' consistencies with the limited retrieval experiences may not have convincing statistical significance, we give all the models

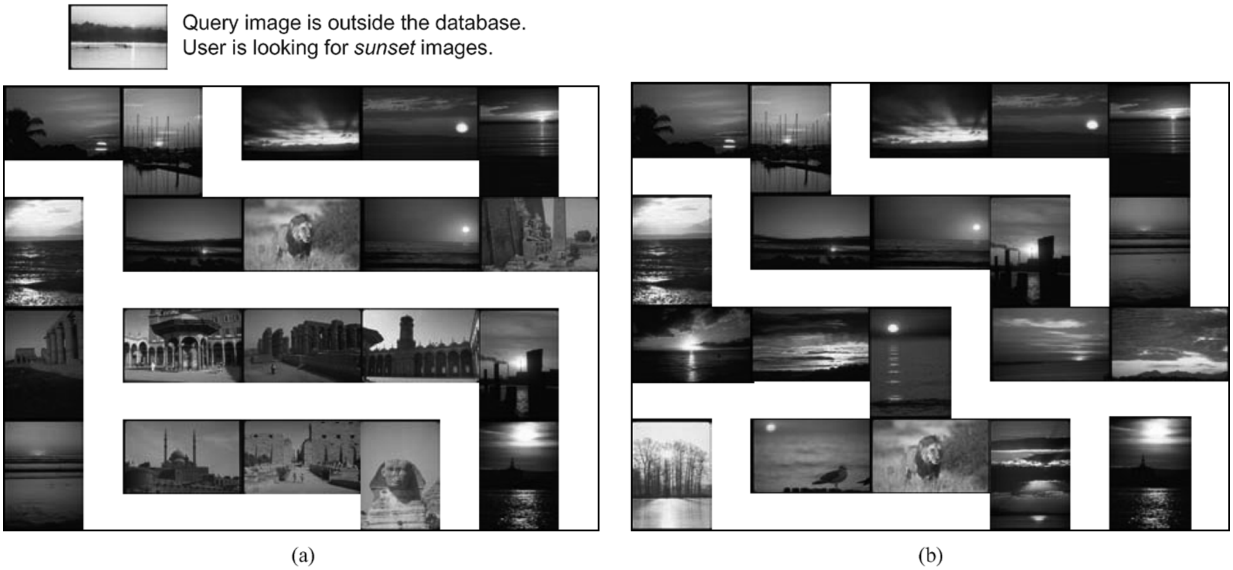


Fig. 11. Retrieval precision is improved as the number of retrieval experiences increases. (a) No retrieval experience ($t = 0$): precision $= 11/20$. (b) $t = 300$: precision $= 19/20$.

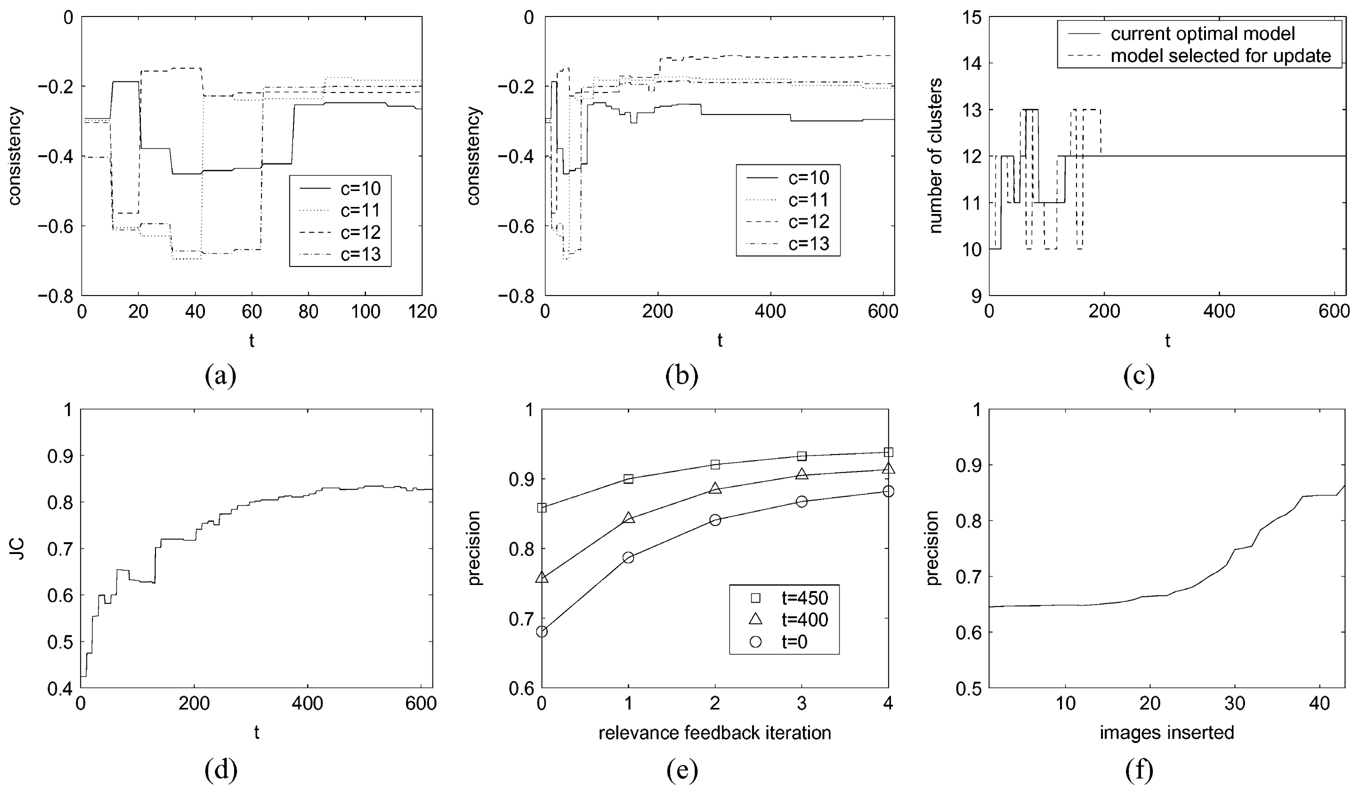


Fig. 12. Active concept learning process. (a) Consistency at initial stage. (b) Consistency in long term. (c) Model selection. (d) Concept learning evaluation. (e) Retrieval precision. (f) Outside queries.

equal probabilities to be selected for SS-EM updating, instead of assigning the probabilities by (24). This conservative strategy refrains the possibility that the model with the true number of components is discarded for future updating because of its low consistency initially. We set this initial stage to be $t \in [0, 200]$. After the initial stage, the probabilities are assigned by (24), where we set $\eta = 0.5$.

Fig. 12 shows the active model selection process. From Fig. 12(a), we observe the oscillations of the consistency values for all the models from $t = 0$ to $t = 80$ due to the reason that the retrieval experiences are limited initially. Then all the consistency curves become relatively smooth, and the consistency of $\mathcal{M}_{12}$ gradually increases with a slow speed [Fig. 12(b)]. This means that the model with the true number of components fits the obtained retrieval experiences better. Another observation is that $\mathcal{M}_{12}$ has the highest consistency after $t = 140$; thus, $\mathcal{M}_{12}$ is always regarded as the current optimal model by the system [Fig. 12(c)]. After $t = 200, \mathcal{M}_{12}$ has the dominant probability to be selected for SS-EM updating.

Fig. 12(d) shows that the model fitting is improved during the process of the active concept learning as more retrieval experiences are obtained. Note that in spite of the overall increasing trend for the JC curve, the JC value occasionally decreases after SS-EM updating, which is caused by the reason that the concept (component) sought by the user may not be correctly identified by (14) as we have discussed in Section III-B. Fig. 12(e) presents the improvement in retrieval performance with increased t. At time t, we select an image in this database as the query, implement our retrieval strategy, and repeat this experiment by changing the query until each of the database images has been selected as a query. Then we compute the average precision at each iteration. With increased retrieval experiences, the average precision is improved, especially at initial K-NN search iteration. This is important for retrieval performance in practical applications since users usually do not have enough patience to repeat relevance feedback iterations to search the images.

To demonstrate the efficacy of our concept knowledge transduction method, at the moment t when a new image is inserted during active concept learning process, we use the rest of the images outside the database (i.e., the 1200 images-$\mathcal{X}(t)$) as queries and simulate the relevance feedback iterations by using the knowledge transduction method. In Fig. 12(f), the average retrieval precision at iteration 0 [K nearest search results using (16)] increases with more images being inserted to the system due to the reason that concept learning is improved since more retrieval experiences are obtained in the process. Note that another factor that improves the precision is that with more images being inserted, there are more relevant images within each class; thus, the probability that relevant images are selected increases.

D. Experiments for Deeper Exploration of the Database

We present two more experiments to demonstrate the important characteristics of our concept learning method: 1) the ability to deal with the case that a semantic concept may contain multiple Gaussian components and 2) the effectiveness of exploiting negative labeling information. We set the database to be static as in Section IV-A, and these characteristics will not be lost for dynamic databases.

- **The exploitation of negative labeling information:** As we have mentioned before, retrieval experiences provide both positive and negative labeling information, and this is different from the traditional training and testing scenario, which actually provides positive labeling information only. If the concept learning ignores the negative labeling information contained in the retrieval experiences, the learning should not be as good as the case of exploiting both positive and negative labeling information. This is validated by Fig. 13, which compares the concept learning performance for these two cases for the 1200 image database. Thus, we conclude that negative labeling information also contributes to the learning.

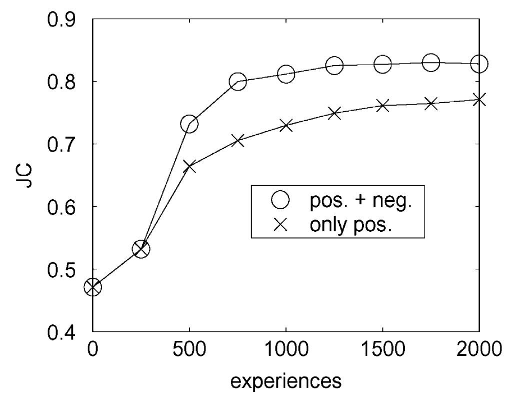


Fig. 13. Performance comparison for concept learning using both positive and negative labeling information versus only using positive labeling.

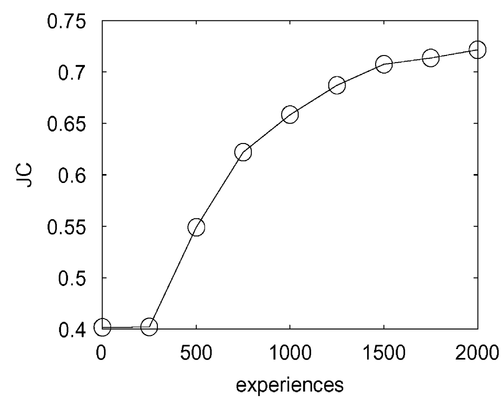


Fig. 14. Concept learning improvement for concepts containing multiple mixture components.

- **Multiple components for a concept:** A semantic concept may contain multiple Gaussian components, which is not reflected in the previous experiments in which the concepts and components are only one-to-one mapping. We add two new image CDs into the original 1200 images, so that some concepts contain multiple components. The two new CDs (series number) are *Hawks Falcons* (70) and *Tulips* (258). The class of *Hawks Falcons* and the class of owls form a concept of *Bird*, and the class of *Tulips* and the class of *North American Wildflowers* form a concept of *Flowers*. Thus, there are 14 Gaussian components, and 12 high-level concepts. For the two newly formed concepts, each of them contains two components; each of the rest of ten concepts contain a single component.

  During the simulation of users' relevance feedbacks, the label of an image is determined by the ground-truth concepts to which an image belongs. To evaluate the concept learning, the Jaccard coefficient is computed based on the ground-truth components, since our goal is to estimate the Gaussian components, instead of the higher level concepts. Fig. 14 presents the concept learning improvement over time for the two concepts that contain multiple mixture components (so there are totally four components contained in these two concepts). It is clear that our concept learning is still effective for the concepts containing multiple components.

V. Conclusion

This paper proposed a unified framework of an active concept learning approach for dynamic image databases. We model the database image distribution in feature space as a mixture of Gaussian densities, and the concept learning is achieved by estimating the mixture model parameters *via* semi-supervised learning. To reduce the gap between low-level visual features and high-level human concepts, the retrieval experiences obtained from multiple users's retrieval sessions are exploited to help the concept learning.

The key contributions in this paper are: a) we present a novel semi-supervised EM algorithm for mixture model parameter estimation. By inserting a modification step between E-step and M-step based on the labeling information obtained from multiple users, we achieve reliable concept learning which is close to the ground-truth image distribution; b) we propose a novel semi-supervised model selection algorithm, which can efficiently learn the number of components in the mixture model. By exploiting multiple users' labeling information, we use Bayesian inference approach to estimate the posterior probabilities of the candidate models, and achieve better model selection with more retrieval experiences. To save the computational load, the analysis of exploitation versus exploration in the search space helps to find the optimal model efficiently; c) to use the concept learning knowledge to improve the retrieval performance of dynamic databases, we present a concept knowledge transduction approach that can efficiently deal with the cases of image insertion and query images being outside the database, while many previous approaches (e.g., [8], [9]) are incapable of dealing with this situation; and d) A variety of experimental results on Corel database show the efficacy of our active concept learning approach and the improvement in retrieval performance by concept transduction.

Our concept learning algorithm has a good potential to be used for large databases. For example, the entire feature space can be partitioned into subregions [30], and our concept learning approach can be implemented for each of the subregions. The concept learning will be improved for each of the subregions; thus, the overall retrieval performance over the entire database is improved.

The ability to overcome users' labeling noise during relevance feedback is necessary for a real database system, since different users may have different opinions on the same image, or some users may even label the images in a random manner rather than using image contents or concepts in a systematic manner. Although the knowledge accumulation mechanism of our system may deal with this phenomenon to some extent, it is possible to find a more efficient way.

Another important issue to be solved is feature selection/reduction which is necessary to save computational load in the mixture model estimation. Traditional methods for feature reduction such as principal component analysis (PCA) and correspondence analysis (CA) are not suitable for our system since the feature reduction has to be adaptive to the increased retrieval experiences, instead of only reducing the feature dimensionality at the initial stage of building the image database system.

References

[1] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, Dec. 2000.

[2] X. S. Zhou and T. S. Huang, "Relevance feedback for image retrieval: A comprehensive review," *Multimedia Syst.*, vol. 8, no. 6, pp. 536–544, 2003.

[3] J. J. Rocchio and G. Salton, "Information search optimization and iterative retrieval techniques," in *Proc. American Federation of Information Processing Societies*, vol. 27, 1965, pp. 293–305.

[4] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A power tool for interactive content-based image retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 644–655, Sep. 1998.

[5] J. Peng, B. Bhanu, and S. Qing, "Probabilistic feature relevance learning for content-based image retrieval," *Comput. Vis. Image Understand.*, vol. 75, no. 1–2, pp. 150–164, Jul.–Aug. 1999.

[6] I. J. Cox, M. L. Matt, T. P. Minka, T. V. Papathoma, and P. N. Yianilos, "The Bayesian image retrieval system, *PicHunter*: Theory, implementation, and psychophysical experiments," *IEEE Trans. Image Process.*, vol. 9, no. 1, pp. 20–37, Jan. 2000.

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

[8] P. Yin, B. Bhanu, K. Chang, and A. Dong, "Improving retrieval performance by long-term relevance information," in *Proc. Int.Conf. Pattern Recognition*, vol. III, Aug. 2002, pp. 533–536.

[9] J. Fournier and M. Cord, "Long-term similarity learning in content-based image retrieval," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, Sep. 2002, pp. 441–444.

[10] B. Bhanu and A. Dong, "Concept learning with fuzzy clustering and relevance feedback," *Eng. Applicat. Artif. Intell.*, vol. 15, pp. 123–138, Apr. 2002.

[11] C. T. Yu, W. S. Luk, and T. Y. Cheung, "A statistical model for relevance feedback in information retrieval," *J. ACM*, vol. 23, no. 2, pp. 273–286, 1976.

[12] N. Vasconcelos, "Bayesian Models for Visual Information Retrieval," Ph.D. dissertation, MIT, Cambridge, 2000.

[13] ——, "On the complexity of probabilistic image retrieval," in *Proc. IEEE Int. Conf. Computer Vision*, vol. 2, Jul. 2001, pp. 400–407.

[14] K. Barnard and D. Forsyth, "Learning the semantics of words and pictures," in *Proc. IEEE Int. Conf. Computer Vision*, vol. 2, 2001, pp. 408–415.

[15] G. McLachlan and D. Peel, *Finite Mixture Models*. New York: Wiley, 2000.

[16] J. C. Bezdek, J. Keller, R. Krisnapuram, and N. R. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Norwell, MA: Kluwer, 1999.

[17] J. Rissanen, *Stochastic Complexity in Stastistical Inquiry*, Singapore: World Scientific, 1989.

[18] M. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 381–396, Mar. 2002.

[19] G. Schwarz, "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, pp. 461–464, 1978.

[20] M. Whindham and A. Cutler, "Information ratios for validating mixture analysis," *J. Roy. Statist. Soc.*, vol. 87, pp. 1188–1192, 1992.

[21] W. Pedrycz and J. Waletzky, "Fuzzy clustering with partial supervision," *IEEE Trans. Syst., Man, Cybern., B: Cybern.*, vol. 27, no. 5, pp. 787–795, Oct. 1997.

[22] A. M. Bensaid, L. O. Hall, J. C. Bezdek, and L. P. Clarke, "Partially supervised clustering for image segmentation," *Pattern Recognit.*, vol. 29, no. 5, pp. 859–871, May 1996.

[23] Y. Wu and T. S. Huang, "Toward self-exploring discriminating features for visual learning," *Eng. Applicat. Artif. Intell.*, vol. 15, pp. 139–150, Apr. 2002.

[24] L. K. Hansen, J. Larsen, and A. Szymkowiak, "Probabilistic hierarchical clustering with labeled and unlabeled data," *Int. J. Knowledge-Based Intell. Eng. Syst.*, vol. 6, no. 1, pp. 56–63, Jan. 2002.

[25] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Mach. Learn.*, vol. 39, pp. 103–134, 1999.

[26] R. N. Dave and R. Krishnapuram, "Robust clustering methods: A unified view," *IEEE Trans. Fuzzy Syst.*, vol. 5, no. 2, pp. 270–293, May 1997.
[27] R. J. Hathaway, "Another interpretation of the EM algorithm for mixture distributions," *Statist. Probability Lett.*, vol. 4, pp. 53–56, 1986.
[28] B. Thiesson, C. Meek, and D. Heckerman, "Accelerating EM for large databases," *Mach. Learn.*, vol. 45, pp. 279–299, 2001.
[29] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. New York: Academic, 1999.
[30] C. Bohm, S. Berchtold, and D. A. Keim, "Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases," *ACM Comput.Surv.*, vol. 33, no. 3, pp. 322–373, Sep. 2001.

Anlei Dong received the B.S. and M.S. degrees in automation from the University of Science and Technology of China, Hefei, and the Ph.D. degree in electrical engineering from the University of California at Riverside (UCR) in 2004.

He currently serves as a member of the technical staff at FastVDO, LLC. His research interests are machine learning and computer vision including object detection, image retrieval, representation, understanding, indexing, and recognition.

Bir Bhanu (S'72–M'82–SM'87–F'95) received the S.M. and E.E. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, the Ph.D. degree in electrical engineering from the Image Processing Institute, University of Southern California, Los Angeles, and the M.B.A. degree from the University of California, Irvine.

Dr. Bhanu has been the founding Professor of Electrical Engineering and served its first Chair at the University of California at Riverside (UCR). He has been the Cooperative Professor of Computer Science and Engineering and Director of Visualization and Intelligent Systems Laboratory (VISLab) since 1991. Currently, he also serves as the founding Director of an interdisciplinary Center for Research in Intelligent Systems (CRIS) at UCR. Previously, he was a Senior Honeywell Fellow at Honeywell Inc., Minneapolis, MN. He has been on the faculty of the Department of Computer Science at the University of Utah, Salt Lake City, and has worked at Ford Aerospace and Communications Corporation, CA, INRIA-France, and IBM San Jose Research Laboratory, CA. He has been the principal investigator of various programs for DARPA, NASA, NSF, AFOSR, ARO, and other agencies and industries in the areas of learning and vision, image understanding, pattern recognition, target recognition, biometrics, navigation, image databases, and machine vision applications. He is the coauthor of *Evolutionary Synthesis of Pattern Recognition Systems* (New York: Springer-Verlag, 2005), *Computational Algorithms for Fingerprint Recognition* (Norwell, MA: Kluwer, 2004), *Genetic Learning for Adaptive Image Segmentation* (Norwell, MA: Kluwer, 1994), and *Qualitative Motion Understanding* (Norwell, MA; Kluwer, 1992), and the co-editor of *Computer Vision Beyond the Visible Spectrum*, (New York: Springer-Verlag, 2004). He holds 11 U.S. and international patents and over 230 reviewed technical publications in the areas of his interest.

Dr. Bhanu has received two outstanding paper awards from the Pattern Recognition Society and has received industrial and university awards for research excellence, outstanding contributions, and team efforts. He has been on the editorial board of various journals and has edited special issues of several IEEE transactions and other journals. He has been General Chair for the IEEE Conference on Computer Vision and Pattern Recognition, IEEE Workshops on Applications of Computer Vision, IEEE Workshops on Learning in Computer Vision and Pattern Recognition; Chair for the DARPA Image Understanding Workshop, and Program Chair for the IEEE Workshops on Computer Vision Beyond the Visible Spectrum. He is a Fellow of the American Association for the Advancement of Science (AAAS), International Association of Pattern Recognition (IAPR), and the International Society for Optical Engineering (SPIE).