Reinforcement Learning for Combining Relevance Feedback Techniques

Peng-Yeng Yin[1], Bir Bhanu[2], Kuang-Cheng Chang[1], and Anlei Dong[2]
[1]*Department of Information Management, National Chi-Nan University, Nantou 545, Taiwan*
[2]*Center for Research in Intelligent Systems, University of California, Riverside, California 92521, USA*
{pyyin, kcchang}@ncnu.edu.tw, {bhanu, adong}@vislab.ee.ucr.edu

Abstract

Relevance feedback (RF) is an interactive process which refines the retrievals by utilizing user's feedback history. Most researchers strive to develop new RF techniques and ignore the advantages of existing ones. In this paper, we propose an image relevance reinforcement learning (IRRL) model for integrating existing RF techniques. Various integration schemes are presented and a long-term shared memory is used to exploit the retrieval experience from multiple users. Also, a concept digesting method is proposed to reduce the complexity of storage demand. The experimental results manifest that the integration of multiple RF approaches gives better retrieval performance than using one RF technique alone, and that the sharing of relevance knowledge between multiple query sessions also provides significant contributions for improvement. Further, the storage demand is significantly reduced by the concept digesting technique. This shows the scalability of the proposed model against a growing-size database.

1. Introduction

Since the users, in general, do not know the make-up of the image database and the techniques used for indexing, the query formulation process should be treated as a series of tentative trials until the target images are found. Relevance feedback (RF) is an automatic process which fulfills the query formulation. Let a user initialize a query session by submitting an image $Q=(q_1, q_2, \ldots, q_t)$ where t is the number of selected features and q_i is the calculated value of the ith feature. The retrieval system compares the query image with each database image $D=(d_1, d_2, \ldots, d_t)$ and returns the top k similar database images. If the user is not satisfied, he/she can activate an RF process by identifying retrievals as relevant or nonrelevant. The system will adapt its internal parameters to involve more desirable images in the next retrievals. The process is repeated until the user is satisfied or the results cannot be further improved.

Most researchers strive to develop a new RF technique and ignore the possible synergism among existing ones. In this paper, we develop a new model named image relevance reinforcement learning (IRRL) that can integrate multiple RF techniques and make a full use of their advantages.

2. Related Works and Our Contributions

2.1 Existing relevance feedback techniques

The query vector modification (QVM) approach [1] repeatedly reformulates the query vector as the mean difference vector between relevant images and nonrelevant ones, in an attempt to redirect the query vector toward a more desired area. The feature relevance estimation (FRE) approach [2] assumes, for a given query, some specific features may be more important than others when computing the similarities between images and the query. The most natural way of estimating the individual feature relevance is to verify the retrieval ability using each feature alone. Finally the feature relevance is used as a weight incorporated into the dissimilarity metric. The Bayesian inference-based (BI) approach [3] estimates the posterior probability that a database image is relevant to the query given the prior feedback history. The probability distribution over all database images is updated after each feedback iteration, the system is therefore able to improve future retrieval performance.

These methods suffer their respective shortcomings. *First*, the QVM puts equal emphasis on every feature dimension, however, relevant images are not consistently relevant to the query on every feature dimension. *Second*, in FRE, the query vector is not reformulated and the query cannot be moved toward a more desired region. *Third*, both QVM and FRE assume that the distributions of relevant images in the feature space form an intrinsic cluster while no matter how sophisticated features are selected the relevant images usually do not form a single cluster. *Fourth*, the BI approach is theoretically the most flexible one since it does not rely on the nearest neighbor criterion, however, it is computationally intensive.

Moreover, all the three kinds of RF approaches deal with a single query based on the relevance knowledge

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set
0-7695-1950-4/03 $17.00 © 2003 IEEE
IEEE COMPUTER SOCIETY

learned from the corresponding query session. The knowledge is erased after the user has terminated the feedback iterations, and it will not be used for the next query processing. Hence, they maintain a form of short-term memory that captures the user's intention for only this specific query.

2.2 Contribution of this paper

The original contribution of this paper includes the following aspects. (1) Two integration schemes, named combination and hybridization, are presented to attain the maximum synergism between different RF techniques. (2) Our system is the first one that can automatically choose the optimal RF approach for a given query at a particular feedback iteration. (3) A shared long-term memory is used to accumulate the relevance knowledge exploited from multiple users' experiences. The long-term relevance knowledge significantly improves the retrieval performance.

3. The Proposed Model

The system diagram for the proposed image relevance reinforcement learning (IRRL) model is illustrated in Fig. 1. When a user starts a new query session, the prior relevance information about the query formulation, feature weights, and prior probabilities of relevant and nonrelevant images is retrieved. When entering the session, the reinforcement learning will navigate the model to select the optimal RF technique for the query at every feedback iteration. When the user terminates the session, the latest relevance information is captured in the knowledge base for updating the entry.

3.1 Integration of multiple RF approaches

Let the retrieval system be provided by the three RF techniques, namely the QVM, FRE, and BI, and also let the system improve the retrievals by executing several RF iterations. We define an RF *strategy* as a sequence of selected RF techniques to be applied at various feedback iterations. Since the retrieval improvement is mainly done at the first two iterations [4], we focus our discussion on this case. For those strategies that apply distinct RF techniques at different iterations, there are two means for integrating them. The first type of integration, called *combination*, simply applies one RF technique at the first iteration and applies the other RF technique at the second. However, the second type of integration, called *hybridization*, applies one RF technique at the first iteration and when the other RF technique is performed at the second iteration, the first RF technique is applied again simultaneously to strengthen the synergetic effect.

- **Integration between QVM and FRE:** Without loss of generality, we assume QVM is applied at the first iteration and FRE at the second. Fig. 2 gives an illustration. Let the original query be the *i*th database image and be denoted by $X_i^{(0)}$. The system returns the closest images to $X_i^{(0)}$ with feature weights $w_1 = w_2$ (see Fig. 2(a)), and requests for relevance feedback. The user identifies relevant and nonrelevant images from the retrievals. By performing QVM at the first RF iteration, the new query vector $X_i^{(1)}$ is derived by $X_i^{(1)} = \alpha X_i^{(0)} + \beta \sum_{Y_j \in R} Y_j / |R| - \gamma \sum_{Y_j \in N} Y_j / |N|$, where R and N denote the sets of relevant and nonrelevant images, and α, β and γ are the parameters controlling the relative importance. Since FRE is not applied, the feature weights remain unchanged. The new query vector is moved to a location closer to the mass centroid of relevant images (see Fig. 2(b)). At the second RF iteration where FRE is applied, there are two integration schemes. For the combination scheme (see Fig. 2(c)), the query vector is not changed ($X_i^{(2)} = X_i^{(1)}$) because the QVM is not applied, only the feature weights are updated ($w_1 > w_2$) according to FRE so as to stretch the boundary of the query's neighborhood. On the other hand, for the hybridization scheme (see Fig. 2(d)), in addition to updating the feature weights ($w_1 > w_2$) based on FRE, the query vector $X_i^{(2)}$ is reformulated by $X_i^{(2)} = \alpha X_i^{(1)} + \beta \sum_{Y_j \in R} Y_j / |R| - \gamma \sum_{Y_j \in N} Y_j / |N|$. As such, the FRE is hybridized with the QVM. It is observed that both types of integration schemes preserve the advantages of each approach, and improve the retrieval performance than using the same approach at all feedback iterations.
- **Integration between QVM and BI:** Here, for simplicity, we refer to BI as a simple Gaussian classifier instead of the sophisticated one introduced in [3]. Assume QVM is applied at the first iteration and BI at the second. The situations before the second RF iteration are as those in the previous case. At the second RF iteration where BI is applied, for the combination scheme the conditional probabilities of $p(Y|R)$ and $p(Y|N)$ are estimated using the observed samples in R and N identified at the second RF iteration. If we assume the relevant images form a Gaussian

IEEE COMPUTER SOCIETY

density, then $p(Y|R) \equiv N(\mu^R, \sigma^R)$ with $\mu^R = \mu(\{Y| \forall Y \in R\})$ and $\sigma^R = \sigma(\{Y| \forall Y \in R\})$ where $\mu(\cdot)$ and $\sigma(\cdot)$ denote the mean and standard deviation. We also use a Gaussian density to model nonrelevant images and let $p(Y|N) \equiv N(\mu^N, \sigma^N)$ with $\mu^N = \mu(\{Y| \forall Y \in N\})$ and $\sigma^N = \sigma(\{Y| \forall Y \in N\})$. The most relevant images are then determined using the Bayesian classifier. For the hybridization scheme, the query vector $X_i^{(2)}$ is reformulated, according to QVM. Since $X_i^{(2)}$ is an estimate for the mass centroid of all possible relevant images, we let $\mu^R = X_i^{(2)}$. Similarly, the mean vector of nonrelevant images is determined by $\mu^N = \rho \sum_{Y_j \in N} Y_j / |N| - \eta \sum_{Y_j \in R} Y_j / |R|$, where ρ and η are relative weights. The standard deviation vectors are derived as in the combination scheme. As such, the BI is hybridized with the QVM by replacing the estimates for the mean vectors with those obtained by QVM.

- **Integration between FRE and BI:** Assume FRE is applied at the first iteration and BI at the second. The system first retrieves the closest images to $X_i^{(0)}$ with $w_1 = w_2$, and requests for relevance feedback. By performing FRE, the weights are updated as $w_1 > w_2$ and the query vector remains unchanged ($X_i^{(1)} = X_i^{(0)}$). At the second RF iteration where BI is applied. For the combination scheme, the conditional probabilities of $p(Y|R)$ and $p(Y|N)$ are simply estimated based on R and N. However, for the hybridization scheme, the weights are further updated due to FRE. Since a larger weight is resulted by a denser distribution on the feature component, the standard deviation of the Gaussian density is inversely proportional to the weight. Let $\sigma_j^R = (1-w_j) \sum_{k=1}^{d} \sigma_k^R \Big/ \sum_{k=1}^{d} (1-w_k)$, where σ_j^R denotes the standard deviation of the Gaussian density of relevant images on the jth feature component. Also, the standard deviation vector of nonrelevant images is derived as $\sigma_j^N = (1-w_j) \sum_{k=1}^{d} \sigma_k^N \Big/ \sum_{k=1}^{d} (1-w_k)$. The mean vectors are computed as in the combination scheme. Hence, the BI is hybridized with FRE by replacing the estimates for the standard deviation vectors.

3.2 Image relevance reinforcement learning

To learn the optimal strategy and the best within-session integration scheme, we propose an image relevance reinforcement learning (IRRL) model. A user initializes a query session by submitting a query image to an agent which is a CBIR system with multiple RF mechanisms, for example, the RF mechanism FRE/hybridization instructs the agent to apply FRE at the current feedback iteration and hybridize FRE with the RF technique that is applied at the preceding iteration, if it exists. The agent applies an action selection rule to perform an RF mechanism. The nearest t images to the query are computed by the selected RF mechanism, these images are then returned to the environment (the end user) for requesting a relevance feedback. The user identifies relevant and nonrelevant images from the retrieved result, and a precision rate about the retrievals can be computed. The state of the environment is, therefore, changed to another state. The precision rate is also provided to the agent as a reward that reveals the desirability about the state transition. The agent observes the new state and repeats the cycle again. This process produces a sequence of states s_i, actions a_i, and rewards r_i. The agent's goal is to learn an optimal strategy for selecting an action in a given state that maximizes the expected sum of total rewards.

Let the image database contain a collection of n images, and let the agent is allowed to perform an RF process on a specific query for at most m iterations. Assume the agent is provided a set of u possible RF mechanisms to choose from. Some notations of the IRRL model are defined as follows.

- A set of states, $S = \{s_{i,j,k}\}_{1 \le i \le n, 0 \le j \le m, 0 \le k \le u}$. A state is characterized by three elements, namely the query image i, feedback iteration j, and the last RF mechanism k performed to this query.
- A set of actions, $A = \{a_h\}_{1 \le h \le u}$. Performing an action corresponds to executing an existing RF mechanism to the query image.
- Positive real-valued rewards, $r \in [0, 1]$. The reward can be described by the precision rate regarding the user's desirability about the current retrievals and it is given by $r = \text{Positive\_R etrievals} / \text{Total\_Retr ievals}$.
- A state transition function, $\delta : S \times A \to S$. By the above definitions, we have $\delta(s_{i,j,k}, a_h) = s_{i,j+1,h}$.
- A reward function, $\tau : S \times A \to R^+$. In particular, $\tau(s_{i,j,k}, a_h)$ will return the precision rate that is calculated on the current retrievals obtained when the agent performs action a_h in state $s_{i,j,k}$.

The IRRL model learns the optimal strategy,

IEEE COMPUTER SOCIETY

$\pi^*: S \to A$, that maximizes the cumulative rewards received over time,

$$\pi^* = \arg\max_{\pi} \{r_0 + \gamma r_1 + \gamma^2 r_2 + \cdots\} = \arg\max_{\pi} \sum_{v=0}^{\infty} \gamma^v r_v ,$$

where r_v is the reward received v steps into the future using strategy π to select actions, and $\gamma \in [0,1]$ is the discounting factor that determines the relative value of immediate and delayed rewards. We apply the *Q*-learning algorithm [5] to learn the optimal RF strategy. Let $Q(s_{i,j,k}, a_h)$ be the maximum cumulative reward which can be received by performing action a_h in state $s_{i,j,k}$ and then proceeding optimally using π^*. The *Q*-learning algorithm iteratively approximates the Q function by the following recursive definition,

$$Q(s_{i,j,k}, a_h) = \tau(s_{i,j,k}, a_h) + \gamma \max_{a_l} Q(\delta(s_{i,j,k}, a_h), a_l)$$
$$= r + \gamma \max_{a_l} Q(s_{i,j+1,h}, a_l) ,$$

The precise *Q*-learning algorithm for the IRRL model is presented in Fig. 3 and is explained as follows. First, the algorithm initializes a table of estimate of the Q function. When a user starts a new query session by submitting a query, say, image *i*, if the image was never used as a query before, the algorithm computes the *t* nearest images according to the Euclidean distance; otherwise, the algorithm retrieves all relevance knowledge (involving query formulation, feature weights, and prior probabilities of relevant and nonrelevant images) about this query, and then computes the *t* nearest images according to the last performed action. Next, if the user is not satisfied with the retrieved results, he/she can perform an RF process to improve the retrievals. At each RF iteration, the user marks the retrievals as relevant or nonrelevant. The algorithm performs an action that is chosen according to a probabilistic action selection rule as

$$p(a_h | s_{i,j,k}) = \hat{Q}(s_{i,j,k}, a_h) \Big/ \sum_{l=1}^{u} \hat{Q}(s_{i,j,k}, a_l) .$$

As such, the probability with which an action is chosen is linearly proportional to the corresponding $\hat{Q}$ estimate. Then, the algorithm computes *t* nearest images according to the performed action, observes an immediate reward and a new state, then updates the corresponding $\hat{Q}$ table entry. The algorithm iteratively approximates the optimal strategy and guides the agent to maximize the expected sum of total rewards (retrieval precisions obtained at all feedback iterations).

3.3 Convergence analysis and storage reduction

Since the Q function estimate approximates the maximal expected sum of precision rates, a near-optimal selection for an RF strategy is learned if the estimate value is significantly larger than others. Let the agent in state $s_{i,j,k}$ have *u* choices of actions, each of which is assigned a selection probability, $p(a_h | s_{i,j,k})$, *h* = 1, 2, …, *u*. We compute the information entropy regarding to these probabilities as $E(s_{i,j,k}) = -\sum_{h=1}^{u} p(a_h | s_{i,j,k}) \log_2 p(a_h | s_{i,j,k})$. The smaller the value of $E(s_{i,j,k})$, the more deterministic the action selection in state $s_{i,j,k}$. Thus, the CBIR agent has learned a dominant strategy $\Phi(X_i)$ for query X_i if the entropy values in the initial state and those states sensed during all subsequent feedback iterations using this strategy are all less than a small threshold *e*.

To save the storage demand, we present a concept digesting method as follows. Assuming that two images determine the same dominant strategy, the relative entry values of the two $\hat{Q}$ tables must be very similar. Accordingly, we let the IRRL agent merge, by averaging, the $\hat{Q}$ tables of those images that determine the same dominant strategy. The $\hat{Q}$ entry update of these images is then operated on the same table. Nevertheless, for those images that have not determined their dominant strategy yet, each of them should still be prepared a separate $\hat{Q}$ table for learning the optimal strategy. So there are two types of concepts: a determined concept consisting of those images that determine the same dominant strategy and a nondetermined concept containing an image that has not determined a dominant strategy. Since the storage demand is proportional to the number of concepts, the storage demand is reduced as the IRRL agent digests determined concepts by merging many nondetermined ones.

4. Experimental Results

We have implemented the comparative approaches and made experiments with the UCR database [6] which contains 10,038 images covering a variety of real-world scenes (see Fig. 4). The images are manually labeled into 56 classes. The number of images in each class varies from 20 to 695.

IEEE COMPUTER SOCIETY

- **Experiment 1: Integration of reinforcement learning with relevance feedback:** We experiment with 200,000 random queries using each RF technique and compute the average precision rates obtained at three different stages, namely the one before any relevance feedback (PR0), the one after the first feedback iteration (PR1), and the one after the second feedback iteration (PR2). It is seen from the first three rows of Table 1 that the three RF approaches have comparable performances. The fourth row gives the average retrieval precisions of the three methods and will be used for assessing the proposed model. The traditional short-term leaning scheme always starts a new query session with a null hypothesis about the query formulations, feature weights, or probability distributions. In contrast, the long-term learning scheme presented here keeps a global memory for each database image for storing the latest relevance information. We apply separately the short-term and long-term learning schemes in the proposed model and the results are shown at the bottom of Table 1. The improvement ratio is defined as the increment on the precision rate divided by the average precision of existing methods. It is seen that the proposed model using either learning schemes obtains substantially higher retrieval precisions than the average performances. Fig. 5(a) shows the first retrieval result of a particular query (in a decreasing order of similarity and the first retrieved image is also the query image itself) obtained by the short-term IRRL. With the human labeling, 3 images are identified as relevant (a precision rate of 30%), and the others as nonrelevant. Fig. 5(b) shows the retrievals using the short-term IRRL after the first feedback iteration, a retrieval precision of 50% is achieved. On the other hand, we also submit the same query image to the long-term IRRL agent, Fig. 5(c) shows the retrievals after the first feedback iteration, a higher retrieval precision of 80% is observed.
- **Experiment 2: Demonstration of concept digesting method:** Fig. 6 shows the number of concepts (e is set to 1.0) with the number of processed queries. When the database is just created, the number of concepts is equivalent to the number of images (10038) since every image corresponds to a nondetermined concept. As the IRRL agent experiences more query sessions and digests determined concepts by merging many nondetermined concepts, the number of total concepts decreases. Finally, the number of concepts is reduced to 1480 which is about only 15% of the original number, the storage demand is also reduced to 15% of its original complexity. Thus the IRRL agent is suited to work with a dynamic database and is able to perform relevance learning for newly added images.

4. Conclusions

Most researchers strive to develop a new relevance feedback approach and ignore the possible synergetic contribution of existing ones. In this paper, we have proposed an image relevance reinforcement learning model that learns the optimal strategy for selecting the right relevance feedback technique, at the right iteration, for the right query image. A long-term learning scheme has been presented to derive the prior relevance knowledge, such that, the relevance learning can start from the preceding state instead of a null hypothesis. The average precision rates obtained using the proposed model are significantly higher than those obtained using the traditional methods. Experimental results manifest that the proposed concept digesting method can reduce the storage demand significantly.

Acknowledgements: This work is supported in part by the National Science Council of Taiwan under grant NSC-91-2213-E-260-027 and US AFOSR grant F49620-02-1-0315.

5. References

[1] J. J. Rocchio, Jr., Relevance feedback in information retrieval, in *The Smart System*, Prentice Hall, NJ (1971) 313-323.

[2] J. Peng, B. Bhanu, and S. Qing, Probabilistic feature relevance learning for content-based image retrieval, Computer Vision and Image Understanding 75 No. 1-2 (1999) 150-164.

[3] I. Cox, M. Miller, T. Minka, T. Papathomas and P. Yianilos, The Bayesian image retrieval system, PicHunter: theory, implementation, and psychophysical experiments, IEEE Trans. On Image Processing 9 (2000) 20-37.

[4] Y. Rui, T. S. Huang, M. Ortega and S. Mehrotra, Relevance feedback: a power tool for interactive content-based image retrieval, IEEE Trans. On Circuit System for Video Technology 8 (1998) 644-655.

[5] C. Watkins and P. Dayan, *Q*-learning, Machine Learning 8 (1992) 279-292.

[6] Vision and Intelligent Systems Lab (VISLab), University of California, Riverside. http://www.vislab.ucr.edu.

IEEE COMPUTER SOCIETY

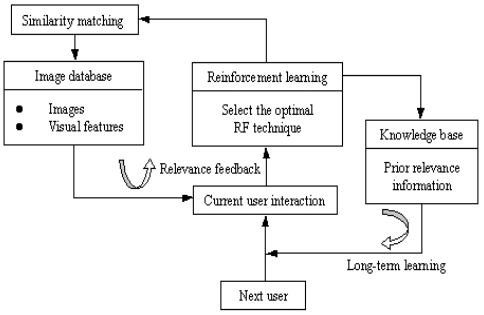


Fig. 1 The system diagram for the image relevance reinforcement learning (IRRL) model.

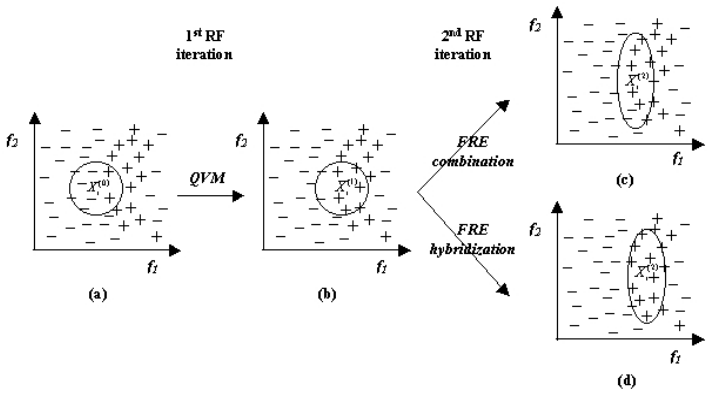


Fig. 2 Illustration of combination and hybridization schemes of QVM and FRE.

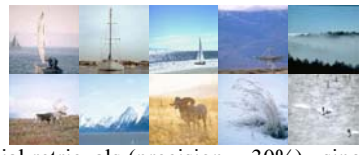
1. Initialize $\hat{Q}(s_{i,j,k}, a_h) = 1.0$ for each i, j, k, and h.
2. While a user starts a query session Do
   2.1. Identify query index i and set iteration index $j = 0$.
   2.2. If there is no prior relevance information for current query, set the last performed action $k = 0$, goto Step 2.4.
   2.3. Identify the last performed action k for current query and retrieve the corresponding relevance information.
   2.4. Set the current state to $s_{i,j,k}$.
   2.5. If $k = 0$ compute t nearest images according to the Euclidean distance; otherwise, compute t nearest images according to the last performed action a_k.
   2.6. While user is not satisfied with retrieved result Do
      (a) User marks the t images as relevant or nonrelevant
      (b) Perform an action a_h, $1 \le h \le u$, chosen with the selection probability
      $$p(a_h | s_{i,j,k}) = \hat{Q}(s_{i,j,k}, a_h) \Big/ \sum_{l=1}^{u} \hat{Q}(s_{i,j,k}, a_l)$$
      (c) Compute t nearest images according to the performed action a_h.
      (d) Observe an immediate reward, $r = \tau(s_{i,j,k}, a_h)$
      (e) Observe a new state, $s_{i,j+1,h} = \delta(s_{i,j,k}, a_h)$.
      (f) Update the table entry by
      $$\hat{Q}(s_{i,j,k}, a_h) = r + \gamma \max_{a_l} \hat{Q}(s_{i,j+1,h}, a_l)$$
      (g) $j = j + 1$, $k = h$.
   end
end

Fig. 3 Q-learning algorithm for the IRRL model.

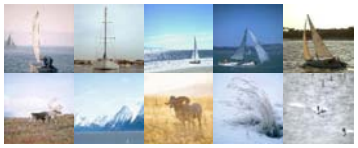
Table 1. Comparative performances.

Approaches	PR0	PR1	PR2
QVM	25.41%	40.40%	41.95%
FRE	25.41%	40.40%	42.68%
BI	25.41%	41.25%	41.67%
Average	25.41%	40.68%	42.10%
short-term IRRL	25.41%	51.34%	53.73%
Improvement ratio	0%	26.17%	27.62%
long-term IRRL	47.84%	58.03%	58.83%
Improvement ratio	88.27%	42.61%	39.74%

Fig. 4 Sample images from UCR database.



(a) the initial retrievals (precision = 30%) using short-term IRRL, images 4-10 are identified as nonrelevant



(b) the second retrievals (precision = 50%) after the first feedback iteration using short-term IRRL, images 6-10 are identified as nonrelevant

(c) the second retrievals (precision = 80%) after the first feedback iteration using long-term IRRL, images 8 and 10 are identified as nonrelevant.

Fig. 5 Retrieval examples

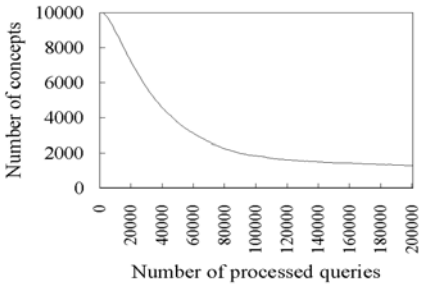


Fig. 6 Number of concepts vs. the number of processed queries.

IEEE COMPUTER SOCIETY