Extreme Video Retrieval: Joint Maximization of Human and Computer Performance

Alexander G. Hauptmann, Wei-Hao Lin, Rong Yan, Jun Yang and Ming-Yu Chen

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, USA 15213
1-412-268-{7003, 7799, 1448, 7458}

{alex, whlin, yanrong, juny, mychen}@cs.cmu.edu

ABSTRACT

We present an efficient system for video search that maximizes the use of human bandwidth, while at the same time exploiting the machine's ability to learn in real-time from user selected relevant video clips. The system exploits the human capability for rapidly scanning imagery augmenting it with an active learning loop, which attempts to always present the most relevant material based on the current information. Two versions of the human interface were evaluated, one with variable page sizes and manual paging, the other with a fixed page size and automatic paging. Both require absolute attention and focus of the user for optimal performance. In either case, as humans search and find relevant results, the system can invisibly re-rank its previous best guesses using a number of knowledge sources, such as image similarity, text similarity, and temporal proximity. Experimental evidence shows a significant improvement using the combined extremes of human and machine power over either approach alone.

Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems – *video.*

General Terms

Experimentation, Algorithms, Human Factors.

Keywords

Video retrieval, relevance feedback, active learning, human performance optimization.

1. INTRODUCTION

Video search, as well as image search, is an extremely complex task, requiring text retrieval, as well as image and video "understanding" to be done very well to succeed [2]. Despite a flurry of recent research in video retrieval, objectively demonstrable success in searching video has been limited, especially when contrasted with text retrieval. Most of the mean average precision in standard evaluations can be attributed to text transcripts associated with the video, with small additional benefit derived from video analysis.

When comparing results of fully *automated* video retrieval to *interactive* video retrieval [9], one finds a big gap in performance. The fully automated search (no user in the loop) succeeds with good recall for many topics, but low precision because relevant shots tend to be distributed throughout the top 5000 slots in the ordered shot list, causing the standard metric of mean average precision for automated search to lag well behind almost any interactive system. One explanation is that a search of the text portion of the query finds the relevant stories, but finding the individual relevant clips is very difficult. Interactive system performance [11] appears strongly correlated with the system's ability to allow the user to efficiently survey many candidate video clips (or keyframes) to find the relevant ones. The best interactive systems allow the user to type in a text query, look at the results, drill deeper if appropriate, , choose relevant shots for color, texture and/or shape similarity match and iterate in this by reformulating or modifying the query [9,10,11,22].

From this insight, we developed a system that relies on superior human visual perception to compensate for low precision in automatic search of the visual contents of video [16]. The human user can filter the best automatically generated results and produce a better set that retains the relevant shots, while using the systems ability to learn from the user's selection, resulting in much greater efficiency of human attention and increased search precision. We named this approach extreme video retrieval (XVR), as it combines the best machine performance with maximal use of human perception skills. Our system explores two types of approaches to human filtering: rapid serial visual presentation and manually controlled browsing with resizing of pages, and combines the computer's ability to rapidly learn from the user with maximizing the user efficiency.

Users have long been offered a more active role in information retrieval through relevance feedback techniques, where after interactively marking the correct items (and sometimes also the incorrect items as negative) returned by a query, a follow-up query can be made more precise. Limitations with relevance feedback techniques, however, include the user's unwillingness to invest time to explicitly label data and concern for introducing extra cognitive load to the user's primary tasks.

The extreme video retrieval system described here simplifies the retrieval task by reducing the user's task to identifying relevant

MM'06, October 22, 2006, Santa Barbara, CA.

keyframes. After an initial ordering of (all) potentially relevant candidate result clips, the system's task is limited to quickly reranking the initially suggested video clips based on the user relevance feedback implicit in the user's selection of relevant examples. This greatly reduces the need for labeled data by taking advantage of active learning. The success of XVR relies heavily on the initial fully automatic retrieval result to contain as many relevant shots early in the automatically generated result list. To study the machine extremes of our automatic retrieval system we take an baseline automatic result [21, 10] produced by our system and plot MAP over 24 TRECVID 2005 search topics. Figure 1 shows the mean average precision (MAP) that could be achieved if a user were to look at different numbers of keyframes (representing the shot), plotted along the x axis. As the user looks at more shots provided by the automatic search (increasing the depth of the search in terms of the number of shots examined) and makes the (*optimal*) decision for each shot whether it is relevant to the query, mean average precision increases, as shown in Figure 1. As fewer shots are relevant, MAP increases somewhat more slowly. In contrast, calculating just the MAP at each shot as ordered by the *automatic* search result yields little improvement after 1000 shots.

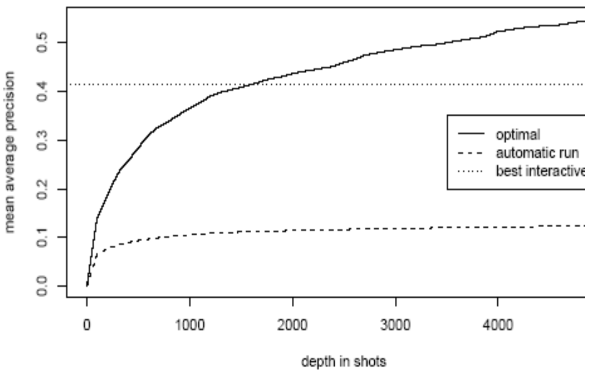


Figure 1. Comparison of automatic retrieval against the best interactive retrieval system. A plot of the result of the automatic retrieval with a perfect (human) re-ranking function that looks at all the shots from the automatic result list up to a certain depth is shown labeled as 'optimal'. At a depth of 1800 shots per topic, a perfect re-ranking function on the automatically obtained data would surpass the results of best interactive user obtained after spending 15 minutes per topic. Only the top 1000 results of the interactive system were judged.

The automatic result demonstrates respectable performance, relative to other automatic video retrieval systems, achieving MAP of around 0.1 at the depth of 1000 shots commonly chosen in TRECVID evaluations. However, this compares very unfavorably to the best interactively system results [22] on the same data, which were able to achieve a MAP of over 0.4. After depth of 1000 shots, MAP of the automatic system reaches a plateau, mainly due to the severe penalty for ranking relevant shots low in the calculation of average precisions.

The well known mean average precision (MAP) is the mean of the average precision for each search topic, with average precision defined as:

$$AP = \frac{1}{R}\sum_{i=1}^{R}\left(\frac{i}{r_i}\right)$$

where R is the total number of relevant shots in the ranked list; and r_i is the rank of the relevant section i.

Assuming an optimal ranking function, the optimal ranking curve becomes equivalent to recall at depth k. Clearly our automatic retrieval system has decent recall at various depths of search, shown as the curve labeled "optimal" in Figure 1. If we equate a human with an optimal ranking function, the results show that anyone who had browsed through the top 2000 shots (2.56% of whole TRECVID 2005 test set) for each query topic could have achieved a result equivalent to the official best search performance in TRECVID 2005, and even better performance if she had looked at more than 2000 shots (pre-ranked by the automatic video search system) in the 15 minutes that were standardized by NIST in interactive search for each evaluation query topic [16] to make different interactive system results comparable.

The rest of this paper is organized as follows: the automatic video retrieval system (baseline) is described in Section 2. Section 3 presents the human interface with section 4 discussing active learning in this context and section 5 contains the discussion and conclusion.

2. AUTOMATIC VIDEO RETRIEVAL

Our video retrieval experiments are built on a relevance-based probabilistic retrieval model, which aims to combine diverse knowledge sources from different retrieval components and semantic concept outputs. This model translates the retrieval task into a supervised learning problem with the parameters learned discriminatively (off-line) from labeled training data gathered from earlier queries. Rather than treating retrieval as a classification problem, we use an algorithm called "ranking logistic regression" by accounting for the order information in training data, so that the optimization is closely associated with the retrieval performance criteria.

Our automatic system employs a basic relevance-based probabilistic retrieval model as a principled framework to combine diverse knowledge sources in multimedia retrieval. In the world of the text-based retrieval, relevance-based probabilistic models rank documents by sorting the conditional probability that each document would be judged relevant to a given query, i.e., P(y=1|D, Q). The underlying principle using probabilistic models for information retrieval is called Principal Ranking Principle that suggests sorting the documents D by the log-odds of the relevance given their presence. There exist a number of models to estimate the log-odds O(y|D,Q) in the literature and most of them have a root in the Binary Independence Model (BIM). Inspired by BIM, we can model the video retrieval problem as the probability that a shot S_i is relevant based on all the available rankings by the various knowledge sources, in the following form:

$$P(Y=1\mid D,Q)=\sigma\left(\sum_{i=0}^{N}\lambda_i P\left(S_i\mid D,Q\right)\right)=\left(1+\exp\left(-\sum_{i=0}^{N}\lambda_i P\left(S_i\mid D,Q\right)\right)\right)^{-1}$$

where $\sigma(x)=\left(1+e^{-x}\right)^{-1}$ is the logistic function and λ_i is the combination parameters for the outputs from different knowledge sources *P(S$_i$|D,Q)*.

With all the outputs of video descriptors available, the next step in the probabilistic retrieval model is to estimate the corresponding combination parameters λ_i. After taking multiple factors into account in the context of the multimedia retrieval, we adopted discriminative models to estimate the parameters, which can directly model the classification boundary and require fewer model assumptions.

However, several problems arise when we cast the retrieval task as a binary classification problem, such as logistic regression. For instance, in most retrieval scenarios the amount of positive data is vastly smaller than the negative data. More importantly, the optimization criterion of classification has little relationship to the retrieval performance measure, namely, average precision. This can lead to some unexpected effects on the learned weights. Therefore, we utilized a new approach called "ranking logistic regression" which takes the ranking information into account. Rather than directly trying to classify the positive and negative examples, it attempts to maximize the gaps between each pair of positive and negative examples. Note that this is different from the margin maximization in a max-margin classifier which only considers the examples near the classification boundary. Formally, the model can be written as,

$$\max_{\lambda}\sum_{q\in Q}\sum_{d_1\in D^+}\sum_{d_2\in D^-}\log\sigma\left(\sum_{i=0}^{N}\lambda_i\left[P\left(S_i\mid d_1,q\right)-P\left(S_i\mid d_2,q\right)\right]\right)$$

where D_+ and D_- are the collections of positive/negative documents. It can be proven that the minimization of the 'disorder' in the examples provides a lower bound of the average precision measure. However, optimizing the above loss function in a brute force manner is computationally expensive. For instance, the association between 100 positive and 900 negative examples results in an explosive 90,000 training pairs. Fortunately, we have come up with an approximation of the above loss function in the form of

$$\max_{\lambda}\sum_{q\in Q}\sum_{d\in D}w_d\log\sigma\left(\sum_{i=0}^{N}\lambda_i\left(P\left(S_i\mid d,q\right)-a_i\right)\right)$$

where w_d are the additional weights, each as ratio between the number of positive/negative data, and a_i is a shift factor. It can be proved that this approximation is tight and the optimization complexity is the same as the logistic regression. Our retrieval model was built upon this approximated version of "ranking logistic regression".

2.1 Query Analysis

In the previous paragraphs, we considered a relevance-based probabilistic retrieval model for knowledge combination in multimedia retrieval, but a lot of previous work showed that simply adopting a query-independent knowledge combination strategy is not flexible enough to handle the variations in users' information needs. It is extremely desirable to develop more advanced methods to incorporate *query information* into the probabilistic retrieval model, as different queries require emphasis of different knowledge sources. To achieve this, we make the following assumptions on the query space:

1) The entire query space can be described by a finite number of mixtures, where the queries from each mixture have similar characteristics and share the same combination function
2) Query descriptions can be used to indicate which mixture the query belongs to.

The simplest approach defines the query types using the human knowledge. Formally, the retrieval model can be represented as:

$$P(y_+\mid D,Q)=\sum_{k=1}^{K}P(z_k\mid Q)\cdot\sigma\left(\sum_{i=0}^{N}\lambda_{ki}P\left(S_i\mid D,Q\right)\right)$$

where z_k are the variables indicating the defined query types. There is one and only one z_k is set to 1 while the others are set to 0. We assigned each query to one of five different types:

Named person: queries for finding a named person, possibly with certain actions

Named object: queries for a specific object with a unique name or an object with consistent visual appearance.

General object: queries for a general category of objects instead of a specific one among them

Sports: queries related to sport events

Scene: queries depicting a scene with multiple types of objects in certain spatial relationships

The query type classification method can be found in [21]. After each query is associated with a single query type, the parameters can be estimated in the way described in last section except the training data are restricted in the specific query type.

All combination weights were obtained using the TRECVID 2004 queries and official results as training data. Text retrieval combined the English speech recognizer transcripts with the English translations of the Arabic and Chinese speech transcripts.

The combination weights were trained for five different types of retrieval components, i.e., text retrieval, color/texture/edge-based retrieval [5] and Person-X retrieval [20]. In addition, combination weights were estimated for 14 frequently-used semantic concepts [14] in the query analysis stage, specifically: Face [17], Anchor, Commercial, Studio, Graphics, Weather, Sports, Outdoor, Person, Crowd, Road, Car, Building and Motion [10].

To obtain the initial retrieval result of each query, we stripped the common head (e.g., "Find shots of") off the original query text and extracted the noun phrases to form the query keywords. Note that for "text" we used the combination of automatic speech recognition, translation and Video OCR. This baseline automatic retrieval system achieved a MAP of 0.117 for the 2005 query topics when evaluating the top 1000 shots. The recall on this set of 1000 shots was 0.394 (equivalent to the MAP of an optimal re-ranking function on this set).

3. MAXIMIZING HUMAN INFORMATION PROCESSING THROUGHPUT

In this section we discuss the human component of the video retrieval system, examining two ways to maximize the efficiency of a human in finding video clips. We assume the person has some query text and perhaps some sample images/video that were submitted to the system. Our analysis of human efficiency starts at that point, with an attempt to allow the user to review candidate results quickly. In the first case, we will discuss Rapid Serial Visual Presentation, where images are rapidly flashed at the user. In the second case, Manual Paging with Variable Pagesize, we afford the users more control within almost the same displays, allowing them to decide when to view the next page of images, as well as controlling the number of image on a page. In both cases, the goal is to allow the user to examine as many keyframes as possible in a given time.

3.1 Rapid Serial Visual Presentation

Rapid Serial Visual Presentation (RSVP) is a technique for rapidly presenting a series of images, and has been widely used in visualization and psychophysics experiments [4, 18]. The core idea behind RSVP is to eliminate all eye movements, as they take time away from the task of looking at an image. Thus all images are presented on the same location on the screen. Research published in the literature has shown that people are able to detect the presence of a specific letter in a collection of simple line images presented at frame rates up to 10 frames per second, or one new frame every 100 millisecond. This was the fastest presentation rate or our system.

3.1.1 Keyhole Presentation

The basic version of RSVP, known as the keyhole mode, presents a sequence of images in the same position of the screen, where the following image replace the previous one every n milliseconds, n is thus the interval between two images. Users can vary the presentation speed (adding or subtracting 100ms from n) with two keys A (accelerate) and S (slow) on the left hand. When a relevant image is shown on the screen, users press the '7' key with the right hand to mark the current image as relevant. All unmarked images are assumed to be irrelevant. Because of the delay in human reaction time, the system also marks the image before a marked image as relevant. Early experiments demonstrated that frequently a user decided to mark an image as relevant but by the time the key was physically pressed, the system had already flashed onto the next keyframe, resulting in many false alarms and misses. Since two images are marked for each relevant key press, a second phase, called the correction phase is needed to carefully page through all marked images and validate the judgments. The duration of this correction phase was dependent on the number of relevant images and was included in the total time allowed. During the correction phase, each relevant image was re-examined more carefully, and marked as relevant or irrelevant. The act of marking the image with a key press during the correction phase automatically brought up the next image to be verified. Note that during the correction phase false alarms could be corrected, but missed images remained unrecoverable in this session. In addition, during the correction phase, if the user was unsure about the relevance of a shot, it can be marked as "maybe"; where all "maybe" shots will be sorted after those ranked as "relevant".

In general, users preferred to start out slower (400 - 500 milliseconds between images) until their eyes and brains adapted to the task, usually within a minute or so. At that point, they increased the speed in increments of 100 milliseconds. When fatigue set in during the 15 minute period allocated for one search topic, users could either pause the system completely (e.g. to rub their eyes) or slow back down to a more leisurely pace until they had recovered. Since all our users were researchers and motivated to perform well, they reported at times not wanting to even blink more than necessary in order to have more time looking at the images.

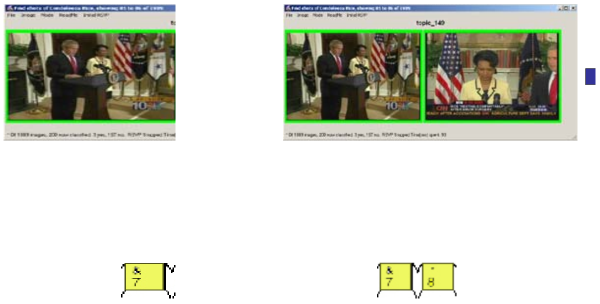
3.1.2 Stereo RSVP

A variation of the keyhole method called the stereo RSVP display was also evaluated. Here the system displayed two images on a page at one time. The motivation was to present one image to each eye, exploiting the natural parallelism of human binocular vision. In this mode, the user used the left ('7') key to mark the left image as relevant or the right ('8') key to mark the right image as relevant. Based on pilot experiments with the system, for each key press, the previous two images were also marked as relevant. A preliminary analysis had shown that users did not consistently press the correct side key corresponding to a relevant image, and human response time delays again resulted in keys presses occurring after an image had already been replaced.

Again, a correction phase was necessary to eliminate the false alarms caused by the extra marked images.

3.2 Manual Paging with Variable Pagesize (MPVP)

Manual Paging with Variable Pagesize (MPVP) is a different strategy for interactive search, which gives the user more control of the display. The right hand used the keys as in RSVP. First of all, whenever users had examined the images on a page, they used the left hand to push the "f" key, which swapped in the next set of images for a new page. The "d" key could be used to page back. Since the time a user spends browsing each page depends on the page size, the visual complexity of the answer, and the number of correct shots to label on the page, this time can vary dramatically with different pages. The user may occasionally need to turn back to previous pages to correct erroneous labels. Thus MPVP gains one advantage by letting users turn pages using a forward and backward keyboard key. However, a conservative user might



(a) **(b)**

Figure 2. Display options used for the RSVP display. (a) shows the keyhole format, with a single key (right hand) to mark an image as relevant. (b) shows the stereo RSVP display, with two keys to mark either the left or right image as relevant. The green bounding boxes indicate the shots labeled relevant, and the keyboard section below a page shows the keys for labeling the respective shots

perform sub-optimally by taking too much time per page to double and triple check every selected answer.

Secondly, unlike RSVP, where the same number of shots per page was used throughout the search, MPVP allows the user to change the page size from one image per page up to a 4x4 image grid layout. This was found to be effective when the density of relevant shots decreased as the user reached deeper into the ordered list of shots hypothesized as relevant by the system. Initially, early in the rankings, when the number of relevant shots was high, users preferred a 2x1 display, marking it with the corresponding buttons on the keyboard. Thus, at the beginning when relevant shots are frequent, a small page size was used since multiple relevant shots were likely on one page, which demanded more attention (per image) and key presses to label them. As the density of relevant shots decreased, users preferred a larger display, since they could visually eliminate all image on a page very quickly. Thus, later in the search when the system was selecting very low probability results and when relevant shots became infrequent, larger page sizes (2x2, and eventually 3x3) were more efficient since it was unlikely that multiple relevant shots would appear even on a large page. MPVP thus reduces the overhead of page turning and the number of necessary key presses for relevant images on a page. Individual users made the decision when to change page size depending on their impression of the density of results and their comfort level.

MPVP also allows up to 16 keys (in a 4x4 layout on the keyboard) for labeling 16 shots simultaneously, with one key corresponding to each presented image. Moreover, another key was available to label all the shots on the current page and automatically turn the page. The keys were laid out such the paging backward and forward and selecting images could be done with minimal finger movement on the keyboard.

Although a page can include any layout of images (e.g., 3x3, 2x5, 4x4, etc), we ended up using only 1x2, 2x2, and 3x3 for two reasons. First, with practice, one hand can conveniently label any shot(s) in layouts up to 3x3 shots, but not more than 9 shots per page. Second, visually inspecting more than 9 shots per page is less time-efficient.

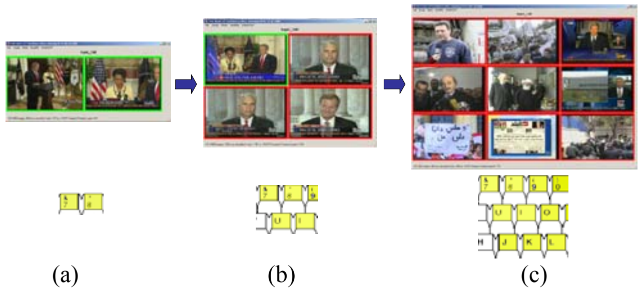


Figure 3. Manual paging with different page size layouts: (a) 1x2 at the beginning, (b) 2x2 in a later stage, and (c) 3x3 for the rest of the shots. The 4x4 display was never selected by the users. The green bounding boxes indicate the shots labeled relevant, and the keyboard section below a page shows the keys for labeling the respective shots.

As the user must label as many shots as possible in a fixed time, errors are inevitable due to time pressures. While missed relevant shots cannot be found during the verification phase, usually one or two minutes were used to correct false alarm errors. As in RSVP, during the correction phase, if the user was unsure about the relevance of a shot, it could be marked as "maybe"; where all "maybe" shots will be sorted after those ranked as "relevant".

3.3 Results of Human Extreme Retrieval

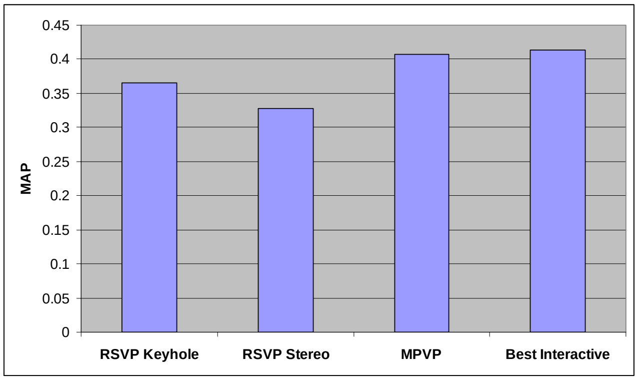


Figure 4. Comparison of MAP results of RSVP Keyhole, RSVP Stereo, MPVP and the best TRECVID 2005 interactive system on the top 1000 marked shots.

The results were evaluated on 24 topics on the 2005 TRECVID data set. Figure 4 shows that the MPVP method (3 users) averaged to a MAP score of .406, while the RSVP keyhole method (1 user) achieved a respectable MAP of .366. The RSVP stereo display method (2 users) was less effective with an MAP of .326. These numbers compared well to the best interactive system [22] which had achieved a MAP of .414. The difference between RSVP Keyhole and the best interactive system was not statistically significant. Looking at the MAP at different depths in Figure 5, we see that all system essentially reach their peak within a few hundred shots, with little improvement afterwards, which is what you would expect for human judgments. No depth information was available for the best interactive system result [22], only its official final MAP score, computed at a depth of 1000 shots. On all queries, all subjects easily reached 1500 shots for examination, with around 2000 shots being typical for most queries. For certain queries, some subjects were able to mark 5000 shots in the 15 minutes allowed in the MPVP condition.

4. ACTIVE LEARNING FROM

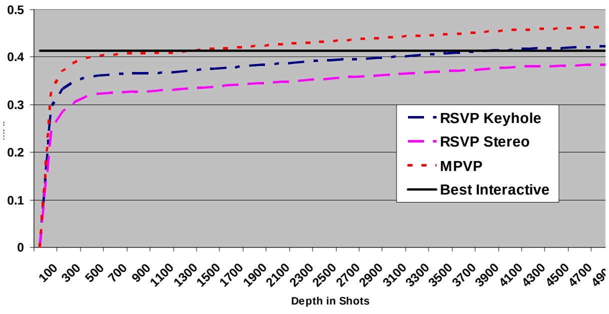


Figure 5. Actual MAP at different depths for the RSVP keyhole, stereo and MPVP conditions. MAP of the best interactive TRECVID submission is shown for reference.

RELEVANCE FEEDBACK IN RETRIEVA

4.1 WHAT IS ACTIVE LEARNING

As outlined in [1, 6, 8, 19, 20], relevance feedback can be used as a query refinement scheme to derive or learn a user's query concept. To solicit feedback, a refinement scheme generally displays a few video shot instances and the user then labels each shot as "relevant" or "not relevant." Based on the responses, another set of shots from the database is presented to the user for labeling. It is hoped that after a few such querying rounds, the refinement scheme has returned a sufficient number of instances from the database that seem to fit the complete needs of the user [7]. The construction of such a query refinement scheme can be regarded as a machine learning task. In particular, it can be seen as a case of pool-based active learning [12]. In pool-based active learning the query refinement scheme, i.e., the *learner,* has access to a pool of unlabeled data and can request the user's label for a certain number of instances in the pool. In the video retrieval domain with shots as the unit of information retrieval, the unlabeled pool would be the entire database of video. An instance would be a video shot, and the two possible labels for each shot would be "relevant" or "not relevant". The goal for the active learner system is to learn the user's query concept.

Continuing the summary of [1], the main issue with active learning is finding a method for choosing informative shots within the pool to ask the user to label. The request for the labels of a set of shots can be termed a pool query. Most machine learning algorithms are passive in the sense that they are generally applied using a randomly selected training set. The key idea with active learning is that it should choose its next pool query based upon the past answers to previous pool queries. In general, and for the video retrieval task in particular, such a learner must meet two critical design goals. First, the learner must learn target concepts accurately. Second, the learner must grasp a concept quickly, with only a small number of labeled instances, since most users are too impatient or preoccupied with more critical tasks to provide a great deal of feedback.

Active learning has demonstrated its effectiveness in reducing the cost of labeling data [15, 13, 3]. Given an unlabeled pool *U*, an active learner *l* has three components (*f*, *q*, *x*). The first component is a classifier, *f*(x)→ (-1,1), trained on the current labeled data *x*. The second component *q*(x) is the querying function that, given a labeled set *x*, decides which instance in *U* to query next. The active learner can return a classifier *f* after each iteration or after some fixed number iterations. Figure 1 illustrates the framework of active learning. Given labeled data *x* (upper left pile), the classifier *f* trains a model based on *x*. The querying function *q* selects the informative data from unlabeled pool (the rectangle). Users annotate the selected data and feed them into the labeled data set.

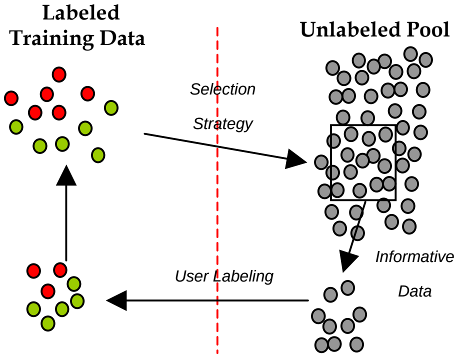


Figure 6. Illustration of basic active learning.

In our extreme retrieval model, each time the user looks at an image and marks it as relevant or leaves it unmarked, we have accumulated one additional piece of labeled training data. While our initial search results were ranked in order of relevance estimated by the system, the shots were not actually labeled by a human. Therefore they should be considered the unlabeled pool for the purposes of active learning.

In 'standard' active learning, we want to choose the most informative data to annotate. Following the procedure of [3], systems generally train a margin-based classifier (e.g. SVM) on the existing labeled data and choose as the next examples those which come closest to the margin or hyperplane. However, in our case, we want to return the best-ranked data to the user, because the user is not interested in building a better classifier, but wants to see results NOW. Thus it is important to improve precision at the top ranked and unseen results. In addition, since the size of our relevant result set tends to be minuscule compared to the full size of the search space, we want to emphasize finding positive examples instead of overloading the user and the active learning algorithm with irrelevant, negative results.

4.2 REWEIGHTING THE EVIDENCE

At the beginning, we had trained the initial combination weighting parameters for the combination of text, color, texture, motion, and core semantic concept retrieval sources with respect to each of the query classes. After mapping the query into these classes, we knew exactly which weights to apply. However, there is some query information that could not be captured this query-type representation. For example, the query "finding the maps of Baghdad" has strong hints to suggest incorporating the output from the semantic concept "maps". More examples are shown in the following table. Given the limited number of query types, we cannot easily take this information into account. However, once we have some relevance feedback in the form of labeled training data, we can further refine the combination weights, i.e., when we find there is direct match between query descriptions and the semantic concepts, the corresponding concepts will be associated with a positive weight. In our current implementation, the concept weights are set to be equal to the weight of text retrieval.

Table 1 Examples of TREC'05 queries and corresponding semantic concepts

TRECVID'05 Queries	Semantic Features
Find the *maps* of Baghdad	maps
Find one/more *cars* on the *road*	cars, roads
Find a *meeting* with a large table	meeting
Find one/more *ships* and *boats*	ship_and_boat

4.2.1 Relevance Feedback

As one of the useful techniques to improve the retrieval performance, the relevance feedback algorithm proceeds by requesting users to annotate a small number of selected video documents from the initial retrieval results and then feeding them back to update the retrieval models. Formally, we denote the relevance information as $y_1,...,y_F$ associated with the feedback documents $D_1,...,D_F$. It can be viewed as a learning component in a retrieval system, where the system learns from a small amount of relevant examples to adjust the ranking function accordingly. In this proposal we mainly consider using the additional annotated data to adjust the combination parameters λ in the probabilistic retrieval models.

Given the relevance judgments, we propose the following model-based relevance feedback approach by computing the maximum a posteriori estimation for the updated combination parameters,

$$\lambda^* = \arg\max_{\lambda} P(\overline{\lambda} \mid y, D, Q, \lambda)$$

$$= \arg\max_{\lambda} P(\overline{\lambda} \mid \lambda) \prod_t P(y_t \mid D_t, Q, \overline{\lambda})$$

$$= \arg\max_{\lambda} \left[\log P(\overline{\lambda} \mid \lambda) + \sum_t \log P(y_t \mid D_t, Q, \overline{\lambda}) \right]$$

where λ are the initial parameters for combination and λ* are the updated parameters after relevance feedback. The prior probability can be defined in many ways and we particularly define it as a Gaussian distribution with mean λ and a pre-defined variance. This formulation can also be interpreted from the maximum likelihood estimation point of view, which is actually making the compromise between two factors: one is minimizing the distance between the updated model parameters and the initial model parameters, and the other is maximizing the likelihood for the feedback data.

4.2.2 Results of Re-Weighting from Context Analysis

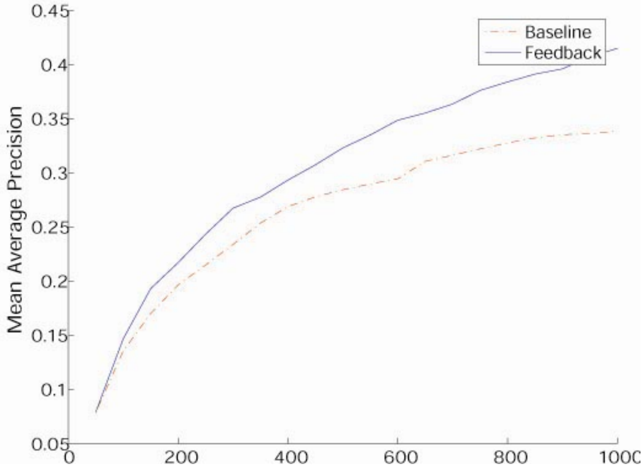


Figure 7. MAP results of relevance feedback context analysis compared to optimal re-ranking of the initial retrieval set at increasing depths of results, using the TRECVID 2005 test set.

Figure 7 shows the results after using active learning re-ranking described in section 4.2.1 with user relevance feedback updated every 50 documents. In this case we see that there is a significant increase in the number of shots examined before the performance equals or exceeds the performance of the best interactive system [22]. After merely providing feedback on 1000 shots, the MAP has equaled that of the best system. This indicates a dramatic improvement over Figure 1, where about 1800 shots needed to be examined to achieve comparable performance to the best interactive search result.

4.3 SELECTING TEMPORAL NEIGHORS

We have also developed an alternative selection strategy that is computationally less complex and shows great promise for extreme retrieval. The crucial insights come from an analysis of temporal sequences in video concepts. After noticing that the semantic concept in a keyframe of a shot is the single best predictor for the concept in the next shot, we tested whether this would also hold true for query results. In other words, if we find a relevant shot, we predict that the same 'query concept' is likely to be relevant in the adjacent shot. This gives a new framework for re-ranking. When a shot in the ranked list of queries is marked as relevant by the user, we simply insert the neighbors of this shot at the top of the shots to be presented to the user on the next display page. Clearly, the main parameter for this re-ranking based on temporal proximity is the number of adjacent neighbors to include, where we experimented with windows of 1, 2, 3, 5, and 10 shots on both sides of a shots marked as relevant by the user.

4.3.1 Results of Temporal Neighbor Reranking

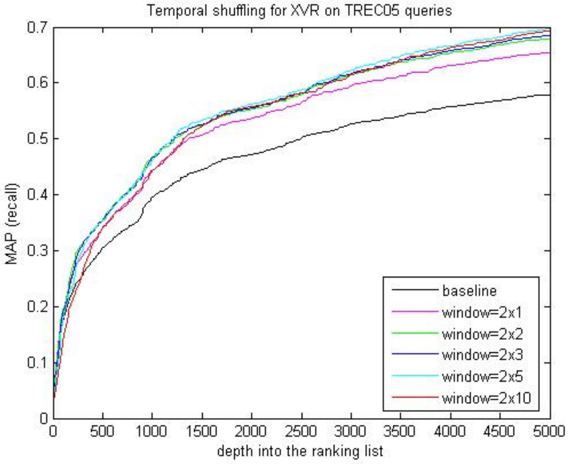


Figure 8. MAP results of temporal reranking at different window sizes in the TRECVID 2005 test set. If a relevant shot is marked at a given depth, its neighbors before and afterwards within the specified window are promoted to the top of the ranked list and subsequently examined by the user.

Figure 8 shows the results of temporal reranking with different window sizes. All windows show an improvement over the baseline, with window sizes of 2, 3 and 5 shots (in both directions from the relevant shot) performing equivalently well at all depths. A window size of 10 shots seems to result in lower MAP at the very early phases of the search. Performance equals the best interactive system baseline [22] after examination of about 1000 shots.

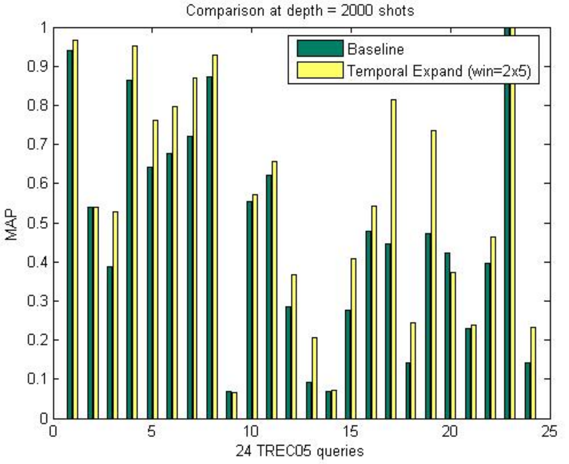


Figure 9. Per topic results comparing the baseline to temporal reranking with a window of 5 shots at a depth of 2000 shots.

Figure 9 shows the temporal re-ranking version of active learning can consistently improve over the optimal re-ranking of the baseline automatic system, which only uses a human to identify relevant results, but does not incrementally learn from partial feedback. In this case, we selected a window of 5 shots before and after a relevant shot, and evaluated performance after 2000 shots had been examined.

5. DISCUSSION AND CONCLUSIONS

A number of researchers have contributed work toward determining what imagery should be labeled next in the iterative step of active learning [3, 19, 20] for better model performance.

In this paper, we have shown that human efforts, with a most simplistic interface, can rival the best interactive user interface systems in terms of finding relevant results. Our MPVP method seems superior to the RSVP methods, because the additional user control prevents errors and allows for a more human-suitable process. However we feel that this will be about the limit of human performance and do not expect additional breakthroughs on the human interface side.

Our active learning for re-ranking with new combination weightings derived from the local query relevance feedback context shows that we improve the human performance substantially further.

Finally, the simple method of using temporal proximity to relevant results to rerank the relevant shot candidates has been shown to be nearly as effective as the more computationally expensive relevance feedback.

Video retrieval remains a difficult problem. Despite much research, the goal of automatically finding the relevant results from a multimodal query has been elusive, and state-of-the-art systems can barely achieve one fourth of the accuracy of an interactive searcher. In this paper we investigate the potential for dividing the labor between humans and computers, splitting the labor so that we can utilize the exceptional capabilities humans have to quickly identify a visually relevant image without getting diverted to other tasks, such as telling the system what to do next, or deciding whether to change to a different query given the absence of good results. The system takes over the rephrasing of the queries, looking at the relevance feedback results in the local context of the current query and presenting the user with a more suitable list of potentially relevant results.

Our human experiments show that even with a very simple interface and a basic automatic retrieval result, it is possible to rival the best interactive results by sophisticated interfaces. But we can far exceed that performance, when we utilize the strength of active learning, leveraging the results of the user interactions and providing an ever improving set of relevant result candidates. This approach maximizes what humans do best and leverages the speed of the computer at presenting and recalculating results.

Future work will include analyzing the difficulties specific types of queries for humans, and finding what types of queries might prevent them from quickly looking through larger sets of results. A hybrid system, that combines the control of MPVP with the blistering speed of RSVP through extra pause, forward, backward buttons (preventing misses) as well as allowing access to the classic interactive search paradigm where the user is allowed to completely rephrase or modify the query should also be investigated.

We plan to also conduct an examination of the types of errors humans make when looking at a list of results but failing to see the relevant shots (misses) as well as the situations where humans are erroneously marking shots as relevant (false alarms). In the long run, we intend to incorporate these insights into the system to further improve the active learning by taking the human weaknesses into account as well as the strengths.

6. ACKNOWLEDGMENTS

This material is based on work supported by the Advanced Research and Development Activity (ARDA) under contract H98230-04-C-0406 and by the National Science Foundation under contract IIS-0535056.

7. REFERENCES

[1] Chang, E.Y., Tong, S., and Goh, K.-S. Support Vector Machine Concept-Dependent Active Learning for Image Retrieval. *IEEE Transactions on Multimedia* (anticipated 2005), http://mmdb2.ece.ucsb.edu/~echang/mm000540.pdf.

[2] Chang, S.-F., (moderator), Multimedia Access and Retrieval: The State of the Art and Future Directions. In *Proc. ACM Multimedia '99* (Orlando FL, Nov. 1999), ACM Press, 443-445.

[3] Chen, M-Y., and Hauptmann, A., Searching for a Specific Person in Broadcast News Video, International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04), Montreal, Canada, May 17-21, 2004

[4] Derthick, M., Interfaces for Palmtop Image Search. Proc. JCDL (Portland, OR, July 2002), 340-341.

[5] Forsyth, D., and Ponce, J. *Computer Vision: A Modern Approach*. Prentice Hall, Englewood Cliffs, NJ, 2002.

[6] Freund, Y., and Schapire, R.E. A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting. *Journal of Computer and System Sciences, 55,* 1, 1997, 119-139.

[7] Gosselin, P.H., and Cord, M. RETIN AL: An active learning strategy for image category retrieval. In *Proc. IEEE Conf. Image Processing* (Singapore, October 2004), 2219-2222.

[8] Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.

[9] Hauptmann, A.G., and Christel, M.G. Successful Approaches in the TREC Video Retrieval Evaluations. *Proc. ACM Multimedia '04*, ACM Press (2004), 668-675.

[10] Hauptmann, A. G., Christel, M., Conescu, R., Gao, J., Jin Q., Lin, W.-H., Pan, J.-Y., Stevens, S. M., Yan, R., Yang, J., and Zhang, Y. CMU Informedia's TRECVID 2005 Skirmishes, in TRECVid 2005 - Text REtrieval Conference TRECVID Workshop, Gaithersburg, MD, 14-15 Nov. 2005.

[11] Lee, H. and Smeaton, A.F. Designing the User Interface for the Físchlár Digital Video Library, *J. Digital Info.* 2(4), http://jodi.ecs.soton.ac.uk/Articles/v02/i04/Lee/, May 2002.

[12] McCallum, A., and Nigam, K. Employing EM in pool-based active learning for text classification. In *Proc. Int'l Conf. on Machine Learning*. Morgan Kaufmann, 1998, 350-358.

[13] Naphade, M., and Smith, J.R. Active Learning for Simultaneous Annotation of Multiple Binary Concepts. In *Proc. IEEE Intl. Conf. on Multimedia and Expo (ICME)* (Taipei, Taiwan, June, 2004), 77-80.

[14] Naphade, M.R., and Smith, J.R. On the Detection of Semantic Concepts at TRECVID. *Proc. ACM Multimedia '04*, ACM Press (2004), 660-667.

[15] Nguyen, H.T., and Smeulders, A. Active Learning Using Pre-clustering. In *Proc. Int'l Conf. on Machine Learning* (Banff, Canada, July 2004). ACM Press, 2004.

[16] Over P, Kraaij W and Smeaton A.F. TRECVID 2005 - An Introduction. TRECVid 2005 - Text REtrieval Conference TRECVID Workshop, Gaithersburg, MD, 14-15 Nov. 2005.

[17] Schneiderman, H., and Kanade, T. Probabilistic Modeling of Local Appearance and Spatial Relationships of Object Recognition. In *Conf. Computer Vision and Pattern Recognition (CVPR '98)* (Santa Barbara, CA, June, 1998). IEEE Computer Society, 1998, 45-51.

[18] Spence, R., Rapid, Serial and Visual: A presentation technique with potential. Information Visualization, 1(1):13–19, 2002.

[19] Tong, S., and Chang, E. Support Vector Machine Active Learning for Image Retrieval. In *Proc. ACM Multimedia 2001* (Ottawa, Canada, October, 2001). ACM Press, 2001, 107-118.

[20] Wang, L., Chan, K.L., and Zhang, Z. Bootstrapping SVM Active Learning by Incorporating Unlabelled Images for Image Retrieval. In *Conf. Computer Vision and Pattern Recognition (CVPR '03)* (Madison, WI, June, 1998). IEEE Computer Society, 2003, 629-634.

[21] Yan, R., Yang, J., and Hauptmann, A., Learning Query-Class Dependent Weights in Automatic Video Retrieval, Proceedings of ACM Multimedia 2004, New York, NY, pp. 548-555, October 10-16, 2004

[22] Snoek, C. G. M., van Gemert, J. C.,Geusebroek, J. M., Huurnink, B., Koelma, D. C., Nguyen, G. P., De Rooij, O., Seinstra F. J., Smeulders, A. W. M., Veenman, C. J., Worring, M., The MediaMill TRECVID 2005 Semantic Video Search Engine. In Proceedings of the 3rd TRECVID Workshop , Gaithersburg, USA, November 2005

.