Video Search in Concept Subspace: A Text-Like Paradigm

Xirong Li, Dong Wang, Jianmin Li and Bo Zhang
State Key Lab. of Intelligent Tech. and System
Department of Computer Science and Technology
Tsinghua University, Beijing, 100084, China
{lxr,wdong01}@mails.tsinghua.edu.cn,{lijianmin,dcszb}@mail.tsinghua.edu.cn

ABSTRACT

Though both quantity and quality of semantic concept detection in video are continuously improving, it still remains unclear how to exploit these detected concepts as semantic indices in video search, given a specific query. In this paper, we tackle this problem and propose a video search framework which operates like searching text documents. Noteworthy for its adoption of the well-founded text search principles, this framework first selects a few related concepts for a given query, by employing a *tf-idf* like scheme, called *c-tf-idf*, to measure the informativeness of the concepts to this query. These selected concepts form a concept subspace. Then search can be conducted in this concept subspace, either by a *Vector Model* or a *Language Model*. Further, two algorithms, i.e., *Linear Summation* and *Random Walk through Concept-Link*, are explored to combine the concept search results and other baseline search results in a reranking scheme. This framework is both effective and efficient. Using a lexicon of 311 concepts from the LSCOM concept ontology, experiments conducted on the TRECVID 2006 search data set show that: when used solely, search within the concept subspace achieves the state-of-the-art concept search result; when used to rerank the baseline results, it can improve over the top 20 automatic search runs in TRECVID 2006 on average by approx. 20%, on the most significant one by approx. 50%, all within 180 milliseconds on a normal PC.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.2.4 [**Database Management**]: Systems—*multimedia databases, query processing*

General Terms

Algorithms, Design, Experimentation

CIVR'07, July 9–11, 2007, Amsterdam, The Netherlands.

Keywords

Video Search, Query Concept Mapping, Concept Subspace

1. INTRODUCTION

Multimedia Information Retrieval (MIR) has received broad attentions in recent years, thanks to the rapidly growing quantities of multimedia data, such as broadcast video from thousands of TV stations worldwide, and millions of multimedia resources residing in the Web, and the subsequent huge need for accessing and managing these information effectively and efficiently. However, unlike text content which can be retrieved by computer directly, access to video is limited to noisy text associated with the video content, such as automatically recognized and possibly machine translated speech, close captions, and video OCR text. The *semantic gap* between visual features and real video content prevents meaningful interpretation of the video corpora.

Organized to promote research activities in MIR, recent benchmark campaigns like TRECVID [1] have helped and witnessed great advances in this area as a few hundreds of semantic concepts, also termed as High-Level Features in the literature, can be automatically detected while scaling to 1000+ concepts are expected in the near future. For example, 101 concepts are defined in [24] and 834 in [13][1]; 491 concepts are detected in [22], 374 in [8] and 311 in this work. These predefined concepts includes various roles of people, objects, scenes and events (please refer to [13] for more details on the concept definition). Treated as semantic video index, these concepts can serve as a basis for new video search paradigm. Intuitively, if queries can be successfully mapped to existing semantic concepts, search performance will improve significantly since the concept detection accuracy is generally much higher than the search baseline, as already shown in past TRECVID benchmark. For example, a query as "scenes with snow" will surely benefit from concept "'Snow' or even 'Sky" since a snowy scene is often with sky present. Besides, the concepts may serve as the basic "visual terms" for describing the video content, which draws the resemblance of video to text and thus enables new approaches for both semantic search and interface design in video retrieval.

Nonetheless, it remains unclear how to exploit these successfully detected concepts in video search. Two research problems arise here:

1. How to map the query to the concepts automatically,

[1]The 834 concepts were further filtered based on observability to produce a fully annotated subset of 449 concepts.

reliably and in a scalable way (abbreviated as QUCOM (query-concept-map) hereafter),

2. How to effectively search in the concept space and combine with other modalities to improve the retrieval performance, when the concepts related to a query are available.

Ideally, the solution to QUCOM should be on a per query basis, determining both the number of related concepts and their respective weights while accounting for the varying performance of the underlying concepts. Furthermore, it should be done on-the-fly due to the realtime search need. While adding more concepts is only a matter of labeling and training on a certain data set, it is not so easy to find a solution which meets all the above criteria and subsequently to solve the second problem accordingly.

There are at least two difficulties to solve QUCOM. Firstly, it is unrealistic to assume that users can either remember more than 50 concepts in the restricted vocabulary before search or recall them exactly during the search process. Secondly, it is difficult to design an innovative user interface which can display all information about the related concepts to a query, e.g. their meaning, usefulness and detection accuracy. To worsen the matter, different people may have different understanding for the same concept name (such as the concept "map"). Recently, methods have been designed to utilize the detected concepts in search, such as text match between the query keywords and the concept description or a predefined concept ontology, regardless of the varying performance of the underlying concepts [6, 23]. Neo et al. [17] take the concept detection performance into account, yet still use a text match approach. Moreover, these methods ignore the visual aspect of the concepts, which might be also of the same importance for solving QUCOM. For example, it is not straightforward to relate "Mosques" to the query "Helicopters in flight" or relate "Furniture" to the query "People reading a newspaper". However, they are really relevant and the connection can be mined through visual cues, cf. Section 5.3 for details. Though concept suggestion through user feedback works reasonably well in an interactive search mode, it is infeasible in the automatic scenario. We focus on the latter since user attention is a scared resource during search.

Fortunately, example images sometimes are also provided by the user. If treated properly, these images can establish a direct link between the query intention and the semantic concepts and thus offer a feasible solution for QUCOM. However, few previous work is conducted in this direction. Predicting the concepts on the example images, the resulting scores can be concatenated into a vector in the concept space which is of the same dimension as the vectors of the keyframes in the video corpus. Taking the concept space as a whole, pseudo negative examples are drawn and a bag of support vector machine (SVM) and k-NN classifiers are combined to produce a ranked list [6, 15]. Another interesting approach uses the concepts by a pointwise mutual information weight scheme (PMIWS) [29] which weights all concepts in a information theory motivated way. However, it is clear that the query is certainly not related to all concepts. Thus these two approaches may suffer from the irrelevant feature dimensions in the concept space, as they overlook the QUCOM problem. When the detected concepts increase in number, this problem calls for a solution urgently.

Viewing shot (basic video retrieval unit) as visual document, and concept as visual term, the parallelism between video and text document is created naturally and a large amount of well-established approaches are thus ready to use. Inspired by this observation, we propose a *tf-idf* like scheme to solve QUCOM. Once a few related concepts are selected for a query, they form a concept subspace in the whole concept space. Then search can be conducted in this subspace, e.g., either by a *Vector Model* or a *Language Model*, both of which are borrowed from the text area. We further show that direct search through concept subspace achieves the state-of-the-art result within this modality.

Clearly, a successful video search approach should flexibly leverage available multimodal cues for better performance. Though search through concept subspace has showed promising results, it still cannot beat the text baseline, specifically for news video where ASR text is a strong clue of visual content. On the other hand, the text (TXT) and low-level visual features (LVF) modalities, when combined (TXT+LVF) together, produce a reasonable baseline. If appropriately taken into account, the concept subspace can be a good complement to TXT+LVF. Bearing these in mind, we combine the search result of concept subspace with results from other modalities under a reranking framework, as it is quite exciting to see how much we can improve by adding information from this concept subspace. Other reason for adopting the reranking framework is its efficiency and simplicity. Seen from the reranking point of view, reranking through concept subspace provides a different perspective from traditional reranking methods which utilizes either the text [9, 27] or visual pseudo-positives [12, 15] by pseudo-relevance feedback (PRF) techniques.

To sum up, in this paper we propose a video search framework which operates in the concept subspace and adopts the well-founded text search principles. Given a query, it first solves QUCOM and constructs a concept subspace, and then searches or reranks the TXT+LVF search results within this concept subspace. In constructing the concept subspace, we employ *c-tf-idf*, a *tf-idf* like scheme, to measure the informativeness of the concepts to the query. In the search process, we rank shots either via a *Vector Model* or a *Language Model*. In the reranking stage, we investigate two algorithms to leverage between multiple query example images and the baseline search results, i.e., *Linear Summation* and *Random Walk through Concept-Link*. One advantage of this framework is its effectiveness and efficiency. Using a lexicon of 311 concepts from the LSCOM concept ontology, our experiments conducted on the TRECVID 2006 search data set show that: when used solely, the concept subspace analysis method achieves an Mean Average Precision (MAP, cf. Section 5.1 for definition) of 0.046, which is the state-of-the-art concept search result, to the best of our knowledge; when used to rerank the baseline results, it can improve over the top 20 automatic search results in TRECVID 2006 on average by approx. 20%, on the most significant one by approx. 50%, all within 180 milliseconds on a normal PC.

The rest of the paper is organized as follows. In Section 2, we present a brief introduction on generating the concept indices. The proposed video search framework is described in detail and experimental results are then reported in Section 3 and Section 4, respectively. The conclusion is drawn in Section 5.

2. GENERATING CONCEPT INDICES

Given the LSCOM concept annotation on a training set, we follow the state-of-the-art concept detection system [7] to build the concept indices. We filter out concepts with less than 20 positive examples in the training set and get a number of 311 concepts left. Our concept detection model consists of three parts as feature extraction, modeling and fusion.

Feature Extraction. Five kinds of features are extracted for each keyframe, namely as follows:

- Global Color Auto-Correlogram (GAC) 64 dimensional (dim), global color auto-correlogram extracted in the HSV color space,
- Global Keypoint Histogram (GKH), 500 dim, global SURF [5] keypoint histogram extracted on the gray image,
- Edge Histogram Grid (EHG), 320 dim, localized edge histogram extracted from a 5-region layout consisting of four corner regions and a center region overlapping with the other four, where 64 dim edge histogram is extracted for each region,
- Color Moments Grid (CMG), 108 dim, localized color extracted from a 4×3 grid and represented by the first 3 moments for each grid region in HSV color space.
- Keypoint Histogram Grid (KHG), 300 dim, localized SURF [5] keypoint histogram from a 3×2 grid and quantized into 50 bins for each region.

SVM Modeling. Proved by past concept detection experiments [3], SVM classifier (cf. [25] for details) is appreciated for its generalization ability. We follow this respectable tradition. We choose the generalized Gaussian kernel $k(x_i, x_j) = \exp\left(\frac{-D(x_i,x_j)}{\sigma^2}\right)$ where $D(x_i, x_j)$ is the specific distance measure since this allow us to incorporate different distance measures in the same framework. Usually the Euclidean distance is adopted. This is the original Gaussian kernel. For the Keypoint histogram features, the χ^2 distance is adopted instead. Given the feature and kernel, the parameters are determined through cross-validation. After training the model, a sigmoid regression is applied to convert the classifier output into a posterior probability estimate which always lies in [0,1].

A RankBoost [11] based sequential re-sampling procedure is introduced to fully utilize the limited positive examples for each concept. Then accordingly, five SVM classifiers are built sequentially based on these examples. This RankBoost based re-sampling procedure has two nice properties: generate balanced positive/negative examples from the imbalanced data set; generate a linear combination weight w_i for the produced classifier c_i. The automatically generated weights $\{w_i\}$ can be used for fusion afterwards.

Fusion. We adopt the simple weighted average fusion algorithm. Given a pool of N produced classifiers associated with their weights $\{c_i, w_i\}$ where each c_i defines a normalized output function g_i for each example x_j as $g_i(x_j)$, we take the fusion function as $f(x_j) = \sum_{i=1}^{N} g_i(x_j)$. Though simple, this function performs well for concept detection on our internal validation data set. One possible reason for this success is the principled resampling and weight assigning procedure.

3. CONCEPT SUBSPACE ANALYSIS

3.1 Motivation

Given the concept detection results for both the search corpus and the query example images, a concept space is constructed as a linear space where shots and query images are all points in this space. It is clear that not all concept dimensions are related to the query points, and irrelevant ones should be abandoned before subsequent employments. As shown in later experiments (Section 5.4), taking more concepts into account will augment the risk of bringing in more irrelevant and even noisy ones, and may degenerate the retrieval performance. Therefore, an effective measurement is required to evaluate the relevance of concepts to the query. This process can be in general termed as query-to-concept map (QUCOM), as discussed in Section 1.

3.2 Concept Selection via c-tf-idf Metric

From an information-theoretic point of view, the relevance of a term to a query can be interpreted as the information the term bears when the query is observed [2]. Motivated by this observation, we resort to the *tf-idf*, the best known term-informativeness assessment in Information Retrieval (IR) area [4]. By viewing concepts as virtual terms (the occurrence frequency of a concept in a shot is a real value in $[0, 1]$), we can extend to the *concept tf-idf* (*c-tf-idf*). The *c-tf-idf* of concept c in a shot d is defined as follows,

$$c{-}tf{-}idf(c, d) = freq(c, d) \log(\tfrac{N}{freq(c)}), c \in C \qquad (1)$$

where $freq(c, d) \approx P(c|d)$ is the occurrence frequency of c in d, $freq(c) = \sum_d freq(c, d)$ the occurrence frequency of c in the corpus, N the size of the corpus, and C the concept set. $P(c|d)$ is the probability of finding c in d (or generating c by d), estimated by the concept detectors (see Section 2).

The intuition is that more frequent concepts are more likely to be relevant; while concepts with larger inverse document frequency might be more distinctive. Specifically, the *tf* measures the concept popularity; while the *idf* measures the concept specificity. The *c-tf-idf* is thus a good combination of the two properties. The essence of this *tf-idf* based concept selection method is to pick out concepts which maximally reduce the uncertainty of the corpus's relevance to the query [2].

For the query q, concepts are ranked in terms of *c-tf-idf*(c, q) as defined by Equation 1. If multiple query images exist, we assume that they have consistent information need, and therefore $freq(c, q) = freq(c, Q) = \frac{1}{|Q|} \sum_{q' \in Q} P(c|q')$, where Q is the query image set. Then the top k concepts are selected to form a concept subspace which will be further exploited in the subsequent search and fusion stages.

3.3 Search in Concept Subspace

From the perspective of semantic video indexing, a shot can be decomposed into several distinctive concepts which are relevant to the shot, in light of their visual and/or semantic coherence. Concepts have the potential to bridge the *semantic gap* to some extent, since they tend to capture both the visual similarity and the semantic correlation. By viewing concepts as visual terms and shots as visual documents, the parallelism between video and text documents is naturally created and a large amount of well-established approaches in the text area are thus ready to apply to the multimedia area. With this premise, we borrow from the

text search paradigm two well-founded retrieval models, i.e., *Vector Model* and *Language Model*, to perform search within the selected concept subspace.

3.3.1 Vector Model

Known as one of the most classical models in the information retrieval field, *Vector Model* [4] considers a document d and a user query q as t-dimensional vectors $\vec{d}$ and $\vec{q}$, respectively, where each dimension is a weight associated with a distinctive term and t is the size of the term lexicon. The relevance of d with regard to q is measured by the correlation between $\vec{d}$ and $\vec{q}$, which can be quantified, for instance, by the *cosine* of the angle between these two vectors.

In this context, d and q are both points in the concept space, and the relevance metric is defined as

$$sim(d,q) := \vec{d} \bullet \vec{q} = \sum_{c \in C_s} w(c,d) w(c,q) \quad (2)$$

where C_s is the selected concept subset, $w(c,d)$ the *c-tf-idf*(c,d), and $w(c,q)$ the *c-tf-idf*(c,q).

3.3.2 Language Model

In *Language Model* [20], each document is viewed as a language sample, and a query as a generation process. The retrieved documents are ranked according to the probability of generating the query from the corresponding language models of these documents. Specifically, by treating the query as a sequence of terms and each term as an independent event, the probability of producing the query can be formalized as,

$$P(q|d) = P(t_1, \ldots, t_m|d) = \prod_{i=1}^{m} P(t_i|d) \quad (3)$$

where $t_1, \ldots, t_m$ is the sequence of terms in q, and $P(t_i|d)$ the probability of generating t_i from the model of d.

We adopt *Language Model* by rewriting Equation 3 as,

$$sim(d,q) := \log P(q|d) = \sum_{c \in C_s} freq(c,q) \log P(c|d) \quad (4)$$

Generally in *Language Model*, smoothing techniques are utilized to improve the estimation accuracy. Therefore, we further smooth $P(c|d)$ by the *Jelienk-Mercer* method [28], that is,

$$P_\lambda(c|d) = (1-\lambda) P(c|d) + \lambda P(c) \quad (5)$$

where $P(c) = \frac{1}{N} \sum_d P(c|d)$ is the relative frequency of c in the corpus, and $\lambda = 0.1$ throughout this study.

4. FUSION VIA RERANKING

Experiments on TRECVID benchmark show that concept-based search is still not sufficient, as it cannot outperform the text baseline [18]. However, if properly leveraged, the concept modality can be a good complement to other modalities (e.g., text and low-level visual feature). In this part, we study the multi-modal fusion problem under a reranking framework, that is, given a search result list obtained from certain modalities (e.g., text), we target at improving the search quality by reranking the list within the concept subspace. Two algorithms are investigated, respectively, i.e., *Linear summation* and *random walk through concept-link*.

4.1 Linear Summation

Given a search result list, we divide the reranking process into two steps: 1) using a retrieval model to rank shots within the list, for instance, *Vector Model* and *Language Model* discussed in Section 3.3; 2) linearly combining the initial list with the reranked list, as

$$sim_{new}(d,q) = \beta \cdot sim_{initial}(d,q) + (1-\beta) sim_{rerank}(d,q) \quad (6)$$

where β is a weighting factor, indicating the framework's bias on the two ranked list. Ideally, the one with higher precision should be more favored. We use an unbiased weighting scheme for the sake of simplicity, i.e., setting $\beta = 0.5$. In the future, we will study the potential of estimating the search result performance, and develop an adaptive weighting strategy.

With regard to $sim(d,q)$, we use a common rank-based normalization method [10],

$$sim(d_i,q) \approx \frac{N+1-i}{N}, i = 1, ..., N$$

where d_i is the i^{th} shot in the ranked list, and N the list's length[2].

Though the ranked-based normalization will lost the original information to some extent, it has two advantages: Firstly, it is robust as the scores assigned to each shot are smoothed. Secondly and more importantly, it can be used when ranking orders rather than scores are available.

4.2 Random Walk through Concept-Link

We further employ a *Random Walk* model, called *Random Walk through Concept-Link* (RWCL), to leverage the concept modality.

A *Random Walk* (RW) on a given graph $G = \{V, E\}$, where V is the vertex set of size N and E the edge set, describes how a random walker jumps among vertices following the edges with certain probabilities. This can be characterized by a discrete time Markov chain which allows us to compute the probability x_p of being located in each vertex p at time t. Suppose that the transition probability matrix is P and the probability distribution over all the vertices is $x(t) = [x_1(t), \ldots, x_N(t)]^T$, a unique stationary distribution x^* is readily derived since P is a stochastic matrix having its maximum eigenvalue equal to one and this guarantees the convergence (see e.g. [21], chapter 4).

Random Walk models such as PageRank [19] has showed great success in IR area. In the PageRank model, Web pages are connected by forward or backward hyperlinks to form a giant graph. A walker surfs on the graph by following the links, and may restart with a probability α.

Analogous to page links in the Web, in the concept space, points of shots and queries can be viewed as connected by certain concept-links, as shown in Figure 1. Based on this, we attempt to rerank the search results via *Random Walk through Concept-Link* (RWCL).

We follow the PageRank algorithm, which can be formalized in a compact matrix form, as shown in Equation 7.

$$x(t+1) = (1-\alpha) W x(t) + \alpha y \quad (7)$$

where x(t) is the ranking scores of all pages at time t, y the initial ranking scores, $\alpha \in [0,1)$ the restart probability as mentioned before, and W the page-link matrix with the

[2]In this work, we rerank top 1000 results for each query.

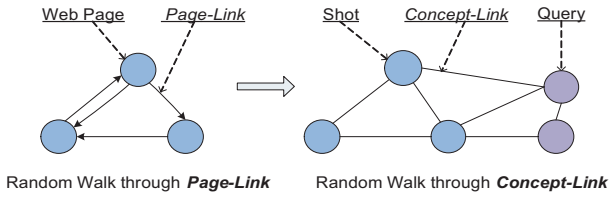

Figure 1: Random Walk through Concept-Link

sum of each column normalized to 1. Let x^* denote the limit of the sequence $\{x(t)\}$. Though the direct solution exists ($x^* = \alpha(I - (1-\alpha)W)^{-1}y$), iterative update is used in real applications.

In our case, y is the initial relevant scores of shots and query points, and W the concept-link matrix. The search results are reranked according to x^*(largest ranked first).

Given the initial search result list $D = \{d_1, \ldots, d_N\}$ and the query points $Q = \{q_1, \ldots, q_M\}$, let y be a $(N+M) \times 1$ vector, where $y(i) = 0.5 - i/N$ for $d_i \in D, i = 1, \ldots, N$, and $y(N+j) = 1$ for $q_j \in Q, j = 1, \ldots, M$. The intuition is that bottom ranked shots should be penalized, while query images are relevant since they are provided by users.

The concept-link matrix $W_{(N+M)\times(N+M)}$ is defined by

$$W(i,j) = \sum_{c \in C_s} P(c|d_i)P(c|d_j), d_i, d_j \in D \cup Q$$

where C_s is the selected concept subset. We argue that the linear kernel is a sensible metric of the concept-links, compared with other ones, such as the common Gaussian kernel. The main reason is that $\|P(c|d_i) - P(c|d_j)\|$ cannot tell whether the concept-link between d_i and d_j is strong or weak, as both cases might have a very small $\|P(c|d_i) - P(c|d_j)\|$. Note that the sum of each column in W is normalized to 1 to ensure convergence.

There are two parameters in the RWCL model: α and the iteration number. The parameter α is to control the score propagation of a point through its concept-links. We simply set $\alpha = 0.01$ so that the relevance scores can fully spread within the concept subspace. With regard to the iteration number, we find that the RWCL converges very fast and a 5-step iteration is generally sufficient.

Compared with the *Linear Summation* scheme, there might be two advantages of the RWCL model: firstly, it can naturally take into consideration the initial ranking orders via y; secondly and more importantly, it may better capture the intrinsic structure of the concept subspace by propagating scores through concept-links.

5. EXPERIMENTS

A serial of experiments are conducted on the TRECVID 2006 (TV06) search data set to give a comprehensive evaluation of the proposed paradigm.

The experiments are split into four parts: First of all, we employ concept detectors (which are trained on the TRECVID 2005 (TV05) data set) to build the 311-concept indices. Secondly, the query-concept-map via the *c-tf-idf* criterion are reported in Section 5.3. We then leverage *Vector Model* and *Language Model* to search within the selected concept subspace, and compare with the state-of-the-art methods which use the same modality. And finally, the reranking approaches are evaluated on a set of baseline search results which consist of the top 20 submitted runs in TV06 automatic search track.

5.1 Data Set and Evaluation Metric

TRECVID is organized by NIST and provides an open, metrics-based evaluation via a common large data set for video retrieval and indexing techniques. The TV06 data set consists of 150-hour multilingual news video captured from MSNBC/NBC/CNN (English), LBC/ALH (Arabic) and CCTV/PHOENIX/NTDTV (Chinese), with 79,484 shots and an official set of 144k image keyframes. The concepts from the LSCOM [13] multimedia concept ontology are annotated on an 80 hours training set on a keyframe basis. The video data are segmented into shots and each shot is represented by a few keyframes. Please refer to [1] for more details about the data set.

We use all 24 multimedia search queries defined in TV06 for the experiments. They express the information need of users for video search concerning people, things, events, locations, etc. and combinations of these needs. The performance is evaluated by Average Precision (AP) on shot level. AP is adopted as the ranking goodness measurement since we are detecting the concepts for retrieval. Given a ranked list L, AP is defined as $\frac{1}{R}\sum_{j=1}^{S}\frac{R_j}{j}I_j$ where R is the number of true relevant instances in a set of size S; R_j the number of relevant instances in the top j instances; $I_j = 1$ if the j^{th} instance is relevant and 0 otherwise. It can be seen as an approximation to the area under the Precision-Recall curve. The relevant shots are judged by NIST using a pooling method. To compare results across queries, Mean Average Precision (MAP) is defined as the mean AP scores involved for all queries.

5.2 Concept Detection Results

We divided the whole training set into two folds, i.e., video 141 to 240 for training and video 241 to 277 for validation. The validated MAP for all 311 concepts is 0.24. This is comparable to the MAP of 0.26 mentioned in [8][3] for 374 concepts which are also derived from the LSCOM ontology.

5.3 Query Concept Mapping Results

The QUCOM is implemented by ranking the 311 concepts in terms of their *c-tf-idf* (see Section 3.2) values. The results for the 24 queries are listed in Table 1 (only top 3 concepts are given due to page limitation). It can be seen that most concepts judged relevant to the queries do make sense, which shows the effectiveness of the *c-tf-idf* measurement. For example, the three concepts *Soccer*, *Sports*, and *Lawn* are brought out for the query "**0195. soccer goalposts**", and *snow* for the query "**0196. scenes with snow**". It is true that such kinds of concepts might also be triggered by text-match methods, thanks to their strong semantic connections to the query keywords. However, besides this kind of concepts, we find certain concepts which are not explicitly related to the queries, such as *Us_Flags* for query **0179** and *Street_Battle* for query **0182**. Across the baseline search results, the AP of these two queries are increased on average by 42% and 21%, respectively, in the following reranking stage (see Section 5.5 for details). One more example is the query **0187** of finding helicopters in flight. The concept

[3]The results (0.26, 0.39) reported in [8] for 39 and 374 concepts are misplaced and should be exchanged.

Table 2: The comparison of concept search methods

Method	Concept Lexicon Size	Search MAP
text match [8]	39	0.019
WordNet similarity [16]	39	0.018
lexicon mapping [16]	39	0.029
concept model vector [16]	39	0.034
PMIWS [29]	39	0.032
Language Model	39	**0.038**
Vector Model	39	**0.040**
text match [8]	374	0.024
text match [22]	491	0.044
ontology [22]	491	0.011
PMIWS [29]	311	0.025
Language Model	311	**0.046**
Vector Model	311	**0.045**

Mosques is predicted as a relevant one. It is not surprising if we notice that many shots are about the Iraq war, and there exists the coincidence of helicopters and mosques. All of these concepts are not easy to detect by text-match methods. This result further shows the strength of the *c-tf-idf* measurement, that is, the capability of taking into consideration the implicit visual aspects of the concepts.

Let us revisit the QUCOM problem discussed in Section 1. A good solution to QUCOM should automatically measure the relevance of a specific concept to the query, and further determine the number of concepts for subsequent employments. The *c-tf-idf* has showed a great potential to solve the first problem. Still, it has difficulty to answer the latter one. QUCOM is certainly a nontrivial and tough problem. In the following experiments, the size of the selected concepts for each query is fixed to 3 due to the lack of better alternatives for determining the size.

5.4 Concept Search Results

In this section, we evaluate the effectiveness of the two retrieval models, i.e., *Vector Model* and *Language Model*, and compare them with other methods which utilize the same modality following the same automatic search protocol. These methods are either variants of text match (TM) [8, 22, 16] or based on the whole concept space [16, 29], which are the state-of-the-art, to the best of our knowledge. The results are organized as two parts in Table 2 according to the concept lexicon size. The first part is on a limited lexicon size of 39 concepts (LSCOM-Lite [14]) and the second one is on a much larger size.

As shown in Table 2, *Vector Model* and *Language Model* perform comparably and scale gracefully across the two lexicon sizes. Moreover, they achieve the best results on both lexicons, outperforming TM [8, 16], PMIWS, and TM [22]. It is worth noting that these performances are reached with only approx. 60% concept lexicon size (311/491), compared with [22].

The Impact of Concept Lexicon Size. We find that the increase of concept lexicon size (from 39 to 311) gives rise to significant improvements for both *Vector Model* and *Language Model*, with 13% and 21% gains in MAP, respectively. In contrast, for the PMIWS method, this change causes an obvious drop on MAP (from 0.032 to 0.025 in Table 2). One possible reason is, as PMIWS seeks to combine all concepts by auto weighting, it may handle well when the amount of

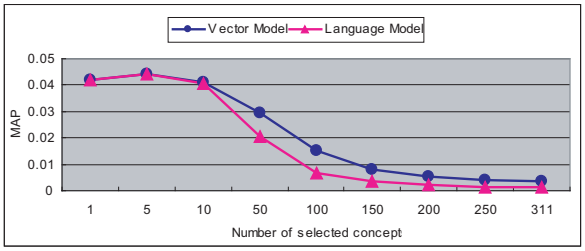

Figure 2: The impact of the selected concept number on the MAP performance of the 24 queries.

Table 3: Improving baseline search results via reranking in the concept subspace. (LS+VM: Linear Summation using Vector Model to rerank; LS+LM: Linear Summation using Language Model to rerank; RWCL: Random Walk through ConceptLink; AMAP: the average MAP across the top 20 runs)

	Baseline	LS+VM	LS+LM	RWCL
AMAP	0.057	0.068	0.067	0.068

available concepts is small. However, as the lexicon scales up, taking many concepts into account will bring in more irrelevant and even noisy ones. The experimental results show the importance of concept selection, specifically for a large-scale concept lexicon.

The Impact of the Number of Selected Concepts. Further, we investigate the following problem: whether the number of concepts selected for retrieval counts. As shown in Figure 2, the performances of both models (i.e., *Vector Model* and *Language Model*) degenerate as the selected concepts increase. This observation again demonstrates the importance of concept selection and the significance of QUCOM when we intend to exploit the concept modality.

5.5 Reranking Results

To evaluate the concept-subspace based reranking framework in a general scenario, we collect the top 20 submitted runs of automatic search track in TV06, which are used as our baselines. All runs contain a 1000-shot search result list for each of the 24 queries. The performances of these runs have been evaluated by NIST [1], with MAP ranging from 0.087 to 0.041, and an average MAP 0.057.

The performances of the reranked results are given in Table 3. Here we adopt the average MAP (AMAP) across the 20 baseline search results to evaluate the overall improvements. It can be seen that "LS+VM" (*Linear Summation* with *Vector Model* to rerank) and RWCL reach the best performance, both enhancing AMAP of the baselines from 0.057 to 0.068 (a 19.3% gain). And "LS+LM" (*Linear Summation* with *Language Model* to rerank) performs comparably, with a 17.5% improvement.

We further check the improvements on individual runs, as shown in Figure 3. Note that the result of "LS+LM" is not present since it is similar to "LS+VM" and both of them are under the same reranking framework, i.e., *Linear Summation*. We find that most of the original search results are improved by the reranking methods. And the most signif-

Table 1: Query-Concept Mapping Results (Top 3 concepts per query are listed)

Queries	Related Concepts	Queries	Related Concepts
0173. emergency vehicles in motion	Car, Ground_Vehicles, Vehicle	**0185**. people reading a newspaper	Furniture, Scene_Text, Hospital
0174. tall buildings and the top story visible	Cityscape, Sky, Urban_Scenes	**0186**. a natural scene	Waterscape_Waterfront, Lakes, Landscape
0175. people leaving or entering a vehicle	Vehicle, Ground_Vehicles, Airport	**0187**. helicopters in flight	Mosques, Airplane_Flying, Helicopters
0176. soldiers, police, or guards escorting a prisoner	Emergency_Room, Protesters, Sunny	**0188**. something burning with flame visible	Smoke, Explosion_Fire, Exploding_Ordinance
0177. daytime demonstration or protest with building visible	People_Marching, Crowd, Protesters	**0189**. people dressed in suits, seated, and with newspaper	Meeting, Conference_Room Flags
0178. US Vice President Dick Cheney	Head_of_State, George_Bush, First_Lay	**0190**. at least one person and at least 10 books	Flags, Us_Flags, Politics
0179. Saddam Hussein with another persons face visible	Us_Flags, Pedestrian_Zone, Parking_Lot	**0191**. at least one adult person and at least one child	First_Lady, Armed_Person, Kitchen
0180. people in uniform and in formation	Crowd,Protesters, People_Marching	**0192**. a greeting by at least one kiss on the cheek	Old_People, Protesters, Demonstration_Or_Protest
0181. US President George W. Bush, Jr. walking	Agent, Head_of_State, First_Lady	**0193**. smokestacks, chimneys, or cooling towers with smoke	Tower, Smoke_Stack, Mosques
0182. soldiers or police with weapons and military vehicles	Armed_Person, Rifles Street_Battle	**0194**.Condoleeza Rice	head_And_Shoulder, Head_of_State, Suits
0183. water with boats or ships	Lakes,Waterways, Waterscape_Waterfront	**0195**. soccer goalposts	Soccer, Sports, Lawn
0184. people seated at a computer with display visible	Furniture, Cables, Computer_Or_Television_Screens	**0196**. scenes with snow	Sky, Snow, Ship

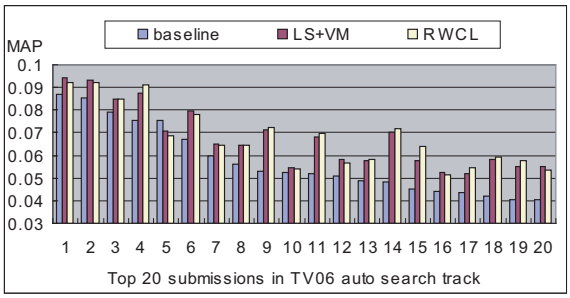

Figure 3: The effectiveness of reranking within concept subspace: a per-run analysis

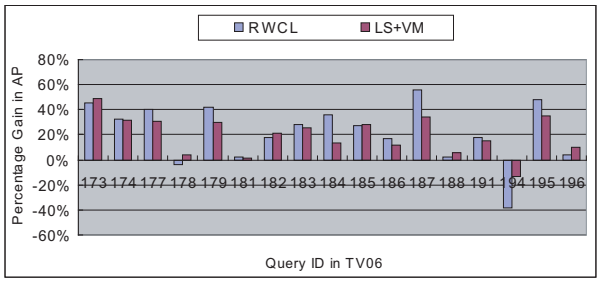

Figure 4: Percentage gain in AP per query in TV06 (queries with AP less than 0.01 are removed)

icant improvement comes from the reranking result of the run 14 (one of the best text baselines in TV06 [22]), leading to a 50% gain in MAP, from 0.048 to 0.072.

Moreover, we question that the improvements might be dominated by one or two well performed queries, in which case the effectiveness of the algorithms is still problematic. Therefore, a further experiment is conducted to examine the percentage gain in AP for individual queries (here, the AP of each query is the average value across the 20 runs). Notice that a change in a very low AP, say, from 0.001 to 0.002, does not make sense in real applications. So those queries with AP less than 0.01 are removed before evaluation. The results are reported in Figure 4. It can be seen that almost all queries benefit from the reranking algorithms. We also examine those very few queries on which the algorithms do not perform well, and find that the main loss comes from the query "**0194**. Find shots of Condoleeza Rice". By re-examining the QUCOM results in Table 1, we find that the selected concepts for the query, e.g., *head_And_Shoulder*, *Head_of_State*, and *Suits*, might be general to some extent, even though they do have certain connections to the query. This specific result points out a further direction for research.

Time Efficiency. Both the search and reranking processes are very efficient, as shown in Table 4. The experiments are all conducted on a standard PC with 3.4 GHz Intel Pentium-4 CPU and 1 GB memory. On average, it takes about 0.17 second to search through the corpus (79,484 shots in total). It takes another 15 milliseconds for the *Linear Summation* model to rerank a 1000-shot search result list, and 0.5 second for the *Random Walk* model. Note that the performances are obtained via an unoptimzied prototype system.

Besides, predicting each concept's presence in each query image is also on the order of milliseconds and it is a highly parallel process which can be executed distributively almost in no time. This advantage makes our method feasible in a practical search engine where real-time execution is a must.

6. CONCLUSIONS

The main contribution of this work is that we proposed a text-like paradigm for leveraging the concept modality in

Table 4: Average Search Time per Query

	VM	LM	LS	RWCL
time (ms)	165	177	15	500

video search, i.e., to find a few related concepts with regard to a given query to generate a concept subspace, to search in the subspace, and to rerank within the subspace the search results obtained via multiple modalities. We show that when treated properly, video search can be conducted in the concept subspace, just as what we had done with text documents. Furthermore, this search result can be easily integrated into the results from other modalities.

In constructing the concept subspace, we employ the *c-tf-idf* metric, a *tf-idf* like scheme, to estimate the relevance of concepts to the query. Then in the concept search process, two retrieval models, i.e., *Vector Model* and *Language Model* are utilized, respectively. And finally, in the reranking stage, we investigate two algorithms to leverage between multiple query example images and the baseline search results, i.e., *Linear Summation* and *Random Walk through Concept-Link*. Comprehensive experiments conducted on the TRECVID 2006 search data set verify both effectiveness and efficiency of the proposed paradigm: when used solely, the concept-subspace based retrieval models reach the state-of-the-art concept search results, to the best of our knowledge; when used to rerank the baseline results, it can improve over the top 20 automatic search runs in TRECVID 2006 by approx. 20%, on the most significant one by approx. 50%, all within 180 milliseconds on a normal PC.

Currently we are exploring new directions to solve the QUCOM problem which can combine the cues provided by the text and visual parts of the query, and/or determine the number of the related concept. Possible further work includes integrating this work into the query classification framework, learning the weights for each concept in a principled way and intelligently suggesting the users to select potential relevant concepts.

7. ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China under the grant No. 60621062 and 60605003, and the National Key Foundation R&D Projects under the grant No. 2003CB317007 and 2004CB318108.

8. REFERENCES

[1] Trecvid home page. `http://www-nlpir.nist.gov/projects/trecvid`.

[2] A. Aizawa. An information-theoretic perspective of tf-idf measures. *Information Processing and Management*, 39:45–65, January 2003.

[3] A. Amir, J. Argillandery, M. Campbell, A. Haubold, G. Iyengar, S. Ebadollahi, F. Kang, M. R. Naphade, A. P. Natsev, J. R. Smith, J. Tešić, and T. Volkmer. IBM research trecvid-2005 video retrieval system. In *Proc. of TRECVID*, 2005.

[4] R. A. Baeza-Yates and B. A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press/Addison-Wesley, 1999.

[5] H. Bay, T. Tuytelaars, and L. Gool. Surf: Speeded up robust features. In *Proc. of ECCV*, 2006.

[6] M. Campbell and et al. IBM research trecvid-2006 video retrieval system. In *Proc. Of TRECVID*, 2006.

[7] J. Cao, Y. Lan, J. Li, and et al. Tsinghua university at trecvid 2006. In *Proc. of TRECVID*, 2006.

[8] S.-F. Chang, W. Hsu, W. Jiang, L. Kennedy, D. Xu, A. Yanagawa, and E. Zavesky. Evaluating the impact of 374 visualbased lscom concept detectors on automatic search. In *Proc. Of TRECVID*, 2006.

[9] T.-S. Chua, S.-Y. Neo, K.-Y. Li, G. Wang, R. Shi, M. Zhao, and H. Xu. Trecvid 2004 search and feature extraction task by nus pris. In *Proc. of TRECVID*, 2004.

[10] K. M. Donald and A. F. Smeaton. A comparison of score, rank and probability-based fusion methods for video shot retrieval. In *Proc. of CIVR*, 2005.

[11] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.

[12] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking via information bottleneck principle. In *Proc. of ACM Multimedia 2006*.

[13] M. Naphade, J. R. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-scale concept ontology for multimedia. *IEEE Multimedia Magazine*, 2006.

[14] M. R. Naphade, L. Kennedy, J. R. Kender, S.-F. Chang, J. R. Smith, P. Over, and A. H. A. A light scale concept ontology for multimedia understanding for trecvid 2005. In *Proc. of TRECVID*, 2005.

[15] A. P. Natsev, M. R. Naphade, and J. Tesic. Learning the semantics of multimedia queries and concepts from a small number of examples. In *Proc. of ACM Multimedia*, 2005.

[16] P. Natsev. IBM marvel for trecvid06 automatic search. In *Proc. of TRECVID*. 2006.

[17] S.-Y. Neo, J. Zhao, M.-Y. Kan, and T.-S. Chua. Video retrieval using high level features: Exploiting query matching and confidence-based weighting. In *Proc. of CIVR*, 2006.

[18] P. Over, T. Ianeva, W. Kraaij, and A. F. Smeaton. Trecvid 2005 - an overview. In *Proc. Of TRECVID*, 2005.

[19] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[20] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proc. of ACM SIGIR*, 1998.

[21] E. Seneta. *Non-Negative Matrices and Markov Chains*. Springer-Verlag, 1981.

[22] C. G. Snoek and et al. The MediaMill trecvid 2006 semantic video search engine. In *Proc. Of TRECVID*, 2006.

[23] C. G. Snoek, M. Worring, D. C. Koelma, and A. W. Smeulders. A learned lexicon-driven paradigm for interactive video retrieval. *IEEE Trans. Multimeida*, February 2007.

[24] C. G. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, , and A. W. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *Proc. of ACM Multimedia*, 2006.

[25] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

[26] D. Wang, J. Li, and B. Zhang. Relay boost fusion for learning rare concepts in multimedia. In *Proc. of CIVR*, 2006.

[27] R. Yan, A. Hauptmann, and R. Jin. Multimedia search with pseudo-relevance feedback. In *Proc. of CIVR*, 2003.

[28] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proc. of ACM SIGIR*, 2001.

[29] W. Zheng, J. Li, Z. Si, F. Lin, and B. Zhang. Using high-level semantic features in video retrieval. In *Proc. of CIVR*, 2006.