Facet-based Browsing in Video Retrieval : A Simulation-based Evaluation

Frank Hopfgartner, Thierry Urruty, Robert Villa, and Joemon M. Jose

Department of Computing Science
University of Glasgow
Glasgow, United Kingdom
{hopfgarf, thierry, villar, jj}@dcs.gla.ac.uk

Abstract. In this paper we introduce a novel interactive video retrieval approach which uses sub-needs of an information need for querying and organising the search process. The underlying assumption of this approach is that the search effectiveness will be enhanced when employed for interactive video retrieval. We explore the performance bounds of a faceted system by using the simulated user evaluation methodology on TRECVID data sets and also on the logs of a prior user experiment with the system. We discuss the simulated evaluation strategies employed in our evaluation and the effect on the use of both textual and visual features. The facets are simulated by the use of clustering the video shots using textual and visual features. The experimental results of our study demonstrate that the faceted browser can potentially improve the search effectiveness.

Key words: aspect based browsing, video retrieval, user simulation, log file analysis

1 Introduction

With the rapid increase of online video services, such as YouTube, the need for novel methods of searching video databases has become more pressing. Much recent work, such as that represented by the TRECVID [6] research effort, aims to tackle the more difficult problems of content based video retrieval. However, overall performance of video retrieval systems to date are unsatisfactory. A number of interactive retrieval systems are proposed to address the many limitations of the state-of-the-art systems (e.g. [1, 3]).

Most of these systems follow a "one result list only" approach, that is, the user query is focused on only one particular issue. In this approach, a user is not able to follow similar ideas he or she might think of during a retrieval session. Users can have a multi-faceted interest, which might evolve over time. Instead of being interested in only one topic at one time, users can search for various independent topics such as politics or sports, followed by entertainment or business.

In this paper, we study the concept of facet-based retrieval. We base our study on an improved version of an interactive video retrieval system which has

been introduced in [7] and propose a novel simulation methodology to evaluate the effectiveness of faceted browsing in which we simulate users creating new facets in an interface. We then discuss different strategies used in our simulation. Furthermore, we support our results by exploiting logfiles of a user study.

The rest of the paper is organised as follows. In Section 2 we introduce the research area of interactive video retrieval. Furthermore, we argue for the use of a facet-based graphical interface and present our system. Then, we propose a simulated user evaluation methodology. In Section 3, we first propose to iteratively cluster retrieval results based on their visual features. The results of the iterative clustering approach indicate that faceted browsing can be used to improve retrieval effectiveness. Subsequently, we analyse user logs from a previous user evaluation study [7] to verify our results in Section 4. In Section 5, we discuss the results of our experiment.

2 Interactive Video Retrieval

In this section, we discuss prior approaches in interactive video retrieval. Then, in Section 2.1, we introduce the concept of facet-based browsing and present our facet-based graphical interface in Section 2.2. Finally, in Section 2.3, we argue for the use of simulation to evaluate a video retrieval engine.

2.1 Background

Current interactive retrieval systems mainly addresses issues of query specification and result browsing. They all follow the same retrieval methodology: even though results will be presented in various different manners, the interfaces present the user only *one* single result list.

However, a user can have a multi-faceted interest, which might evolve over time. For example, a user interested in different aspects of football games wants to collect video shots showing (a) offside situations and (b) penalty kicks. Both situations are related, as they are both common situations in a football game; even though, within the game, there is no obvious relation between a player being offside and a penalty kick. While the state-of-the-art video retrieval interfaces provide only *one* search facility at one time, a user interested in the mentioned football aspects would have to perform his search tasks sequentially session by session, i.e. in searching for (a) and then starting a new session for (b). This means that if a user first wants to focus on finding video clips showing offside situations (a) and, during this search session, finds clips showing free kick situations (b), he has to decide whether he wants to continue searching for the offside situations (a) or to change his focus and continue searching for free kicks (b). Either way, at least one search trail will get lost, as the user has to ignore it. A facet-based retrieval interface, however, provides the user facilities to start additional retrieval sessions (facets) simultaneously and to reorganise materials between these parallel searches.

In order to address many of the problems identified here, we have developed a faceted retrieval system.

2.2 A Facet-Based Video Retrieval System

In this section, we introduce the implementation of a facet-based video retrieval system (see Figure 1). Further details can be found in [7]. As in most retrieval systems, it is divided into a frontend and a backend system.

It is split into one or more vertical panels, each panel representing a step or facet. When the system is initially started up, a single empty panel is displayed on the left of the screen, ready for the user. New panels can be created using the "Add new item" button on the top left of the screen, and will appear at the end of the storyboard, at the far right.

The interface makes extensive use of drag and drop. Shots on the search result list can be dragged and dropped onto the relevant shots area, which will add the shot to the facet's list of relevant shots. There is no restriction on what facet a result can be dragged onto, therefore it is possible to drag a result from one facet directly onto the relevant list of a different facet. Relevant shots can also be dragged and dropped between the different facets list of relevant shots, allowing the reorganisation of material across the different facets. Relevant shots can be removed from the relevance lists using a delete button given on the bottom left of each shot's keyframe.

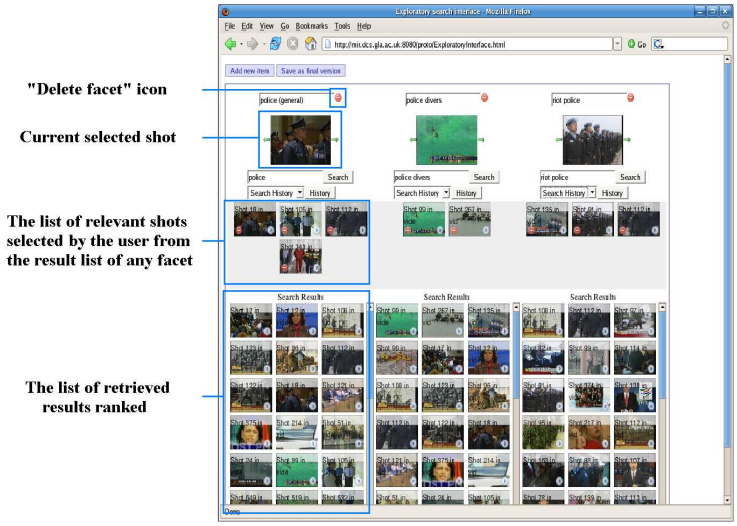

Fig. 1. Screenshot of the facet browsing interface

2.3 Evaluation Methodology

Most interactive video retrieval systems are evaluated in laboratory based user experiments. This methodology, based on the Cranfield evaluation methodology,

is inadequate to evaluate interactive systems [4]. The user-centred evaluation schemes are very helpful in getting valuable data on the behaviour of interactive search systems. However, they are expensive in terms of time and repeatability of such experiments is questionable. It is almost impossible to test all the variables involved in an interaction and hence compromises are needed on many aspects of testing. Furthermore, such methodology is inadequate in benchmarking various underlying adaptive retrieval algorithms. An alternative way of evaluating such systems is the use of simulations.

Finin [2] introduced one of the first user simulation modelling approaches. The "General User Modelling System" (GUMS) allowed software developers to test their systems by feeding them with simple stereotype user behaviour. Hopfgartner and Jose [4] employed a simulated evaluation methodology which simulated users interacting with state-of-the-art video retrieval systems. They argue that a simulation can be seen as a pre-implementation method which will give further opportunity to develop appropriate systems and subsequent user-centred evaluations. However, this approach to evaluate is not mature enough and we need to develop techniques to simulate user behaviour appropriate for the system under consideration.

Our objective is to study the bounds of the proposed faceted browser. We therefore employ the simulated evaluation methodology which assumes a user is acting on the system. If such a user is available, he or she will do a set of actions that, in their opinion, will increase the chance of retrieving more relevant documents. One way of doing this is to select relevant videos. By using a test collection like TRECVID, we will be able to use relevant documents available for our simulation.

In this paper, we adapt the simulation approach in simulating users interacting with a facet-based video retrieval interface. We discuss different strategies used in our evaluation. Starting from a text query, we first propose to iteratively cluster retrieval results based on their visual features in Section 3. The results of the iterative clustering approach indicate that faceted browsing can be used to improve retrieval effectiveness. In Section 4, we subsequently analyse user logs from a previous user study to verify our results.

3 Iterative Clustering

In this section, we present our method to simulate users creating new facets. The idea is to make use of clustering to create groups of similar objects. The clusters are assumed to be the facets of a user's search need and are hence used in the simulation. First, we explain the mechanisms of our algorithm using an iterative clustering technique, then we detail our experimental setup and the various simulations we made before finally discussing the experiment results.

3.1 Iterative Clustering Methodology

The main goal of our facet-based interface is to help the user to create a complex query with separated and structured views of different queries. Our iterative

clustering approach mainly aims to simulate the user in his or her search task. Clusters of our algorithm are assumed to be the facets a real user may create in a search process. A user's first query has a high probability of being general, with the retrieved set of results containing different semantic topics. Our iterative clustering algorithm starts at this step. First, we cluster the retrieved results using textual and visual features. We assume that the top k clusters form the k facets of a user's need and use them to create more specific queries. These queries will then be used to automatically propose new sets of results in new facets. Finally, the iterative clustering process is used to find new facets and refine the queries and consequently the retrieved results.

As we have a small set of retrieved results, we choose to use agglomerative hierarchical clustering and the single link method [5]. Let C, D be two clusters, So_C, So_D the respective set of objects of clusters C and D, the single linkage equation between C and D is given by the following formulas:

- for visual features of images representing video shots we use:

$$D_{visual\ SL}(C, D) = \text{Min}\{d(i, j), \forall i \in So_C \text{ and } \forall j \in So_D\}$$

  where $d(i, j)$ is the Euclidean distance;
- for text queries, we use:

$$D_{text\ SL}(C, D) = \text{Max}\{d(i, j), \forall i \in So_C \text{ and } \forall j \in So_D\}$$

  where $d(i, j)$ is the number of common annotation keywords between two documents.

The output of a hierarchical clustering algorithm is a dendogram. The number of clusters wanted is a parameter of our algorithm, which is used to create the k clusters. We then create a new query for each cluster. For visual features, we choose the medoid of the cluster to create the new visual query. The new text query is based on the most common keywords annotating the cluster. A new search is launched to retrieve k new sets of results corresponding to the k new queries.

We apply clustering on the initial results of the query above. The resulting clusters are used for identifying new facets and subsequently new queries are generated, as explained above. The process is repeated iteratively to identify new facets and hence new queries as well. This iteration can be done in two ways. The first method is completely automatic: results from the first clustering call are directly clustered again to add more precision to the queries. This requires a *number of iterative calls* parameter, denoted N_{ic}. The number of facets N_f that are proposed to the user at the end of the iterative phase is equal to $N_f = k^{N_{ic}}$, so both parameters k and N_{ic} should be low. A "facet waiting queue" may be required if these parameters are too high. The second method requires interactions with the user. At the end of the first clustering phase, new results are displayed in the facet-based interface. Then, for each facet, we simulate the user's actions, e.g., he may choose to delete it, to keep it, or to launch a new clustering

call. Such actions are simulated based on the number of relevant documents in each cluster. For example, clusters with more relevant documents are used as a facet. This "user-simulated interactive" method has some advantages: first it is better adapted to the free space of the interface as the user may delete non relevant facets before each new call; and finally, it does not require the N_{ic} parameter.

In the following sections, we present the experiment setup and our various experiments which lead to the main conclusion that faceted browsing can improve the effectiveness of the retrieval.

3.2 Experiment setup

Our different experiments are based on the TRECVID 2006 dataset. Each of the 24 topics provided in the data collection contains a query of several keywords and a judgement list of 60 to 775 relevant documents. We compute iteratively the precision values of the clusters and automatically select the k best sets of results for the next iterative call. These are the sets of results that have the highest precision, as our goal is to simulate the actions of a user creating new facets. For our experiments, we set $k = 3$, because our list containing 100 results is too small to perform a clustering for higher k values.

3.3 Simulation with single features

In our experiments, we simulate users creating new facets in the faceted browser. A visual query is based on visual features of one or several images. For this set of experiments, we separately used five different low-level features: dominant colour, texture, colour layout, contour shape and edge histogram. For text features, we test two different methodology: (1) query expansion with one, two and three keywords and (2) new text query using two to five new keywords. We record the evolving precision values for various steps of our iterative clustering approach based on visual feature query only.

Table 1 presents the results of our iterative clustering algorithm. For each topic and each feature, we compare the precision of our results with respect to the initial text query, and split the topics in three different categories:

- the precision value of the best results decreases more than 2%, denoted "−";
- the precision value of the best results is almost stable, denoted "=";
- the precision value of the best results increases more than 2%, denoted "+";

Table 2 presents the results for our text experiments. A "positive" effect means that our iterative clustering methodolgy within three facets improve the overall precision of the retrieved results after the initial text query.

As an example, the iterative clustering results based on the texture features increase the precision of results for six topics (out of 24). However, for half of the topics the precision decreases. The conclusion we can draw from both tables is that visual and text features are not reliable for every query. However, for some of the topics, they are useful and improve the precision of the retrieved results. This corroborates with the findings presented at the TRECVID workshop [6].

Visual features	−	=	+
dominant colour	14	10	0
colour layout	14	6	4
texture	12	6	6
edge histogram	11	5	8
contour shape	17	3	4
Average	13.6	6	4.4

Table 1. Results using only visual features queries compare to initial text query

Text queries	no effect	positive effect	number of new relevant documents
add 1	17	7	40
add 2	15	9	63
add 3	16	8	51
new 2	14	10	40
new 3	14	10	51
new 4	13	11	57
new 5	15	9	49
new 6	14	10	45

Table 2. Results using text query expansion and k new keywords as text queries

3.4 Combined Simulation with all Features

In this section, we consider the best facets obtained by individual features. The idea here is not to combine all features in one query but to present every feature in different facets, so the user can choose the relevant features and have a faceted browser showing a lot more relevant documents than the initial retrieved results.

DC	×	×	-	-	×	×	-	-	-	×	-	-	-	×	-	-
CL	×	-	×	-	-	-	-	-	-	-	-	-	-	-	-	-
T	×	×	-	-	-	×	-	×	-	-	-	-	×	-	-	-
EH	×	-	-	×	-	-	-	-	-	-	-	-	-	-	-	-
CS	×	×	-	-	-	-	×	×	×	×	-	-	-	-	×	-
TxA	×	-	-	-	-	-	×	-	-	-	×	-	-	-	-	-
TxN	×	×	-	-	×	-	-	-	×	-	-	×	-	-	-	-
ARD	19.3	43.1	47.8	47.9	48.1	48.1	48.9	49.0	49.1	50.0	50.4	50.5	50.5	51.4	52.5	53.8

Table 3. Best combinations of features

Figure 2 shows the evolution of the number of relevant documents displayed in the faceted browser with respect to the number of facets/features used. We observe that the more facets/features we combine, the more relevant documents are retrieved. So combining facets improves the recall value of relevant results but decreases the precision. A higher recall will help the user to select the relevant results for next query faster. The right part of the figure focuses on the most relevant combinations of three facets which are texture, edge histogram and one of the text feature, query expansion or new 4 keywords, and the less relevant combinations of five facets which contains both dominant colour, contour shape and both text features. These results show that the texture and edge histogram seems to be the best visual features to combine and also that using only one of the two text query models is enough.

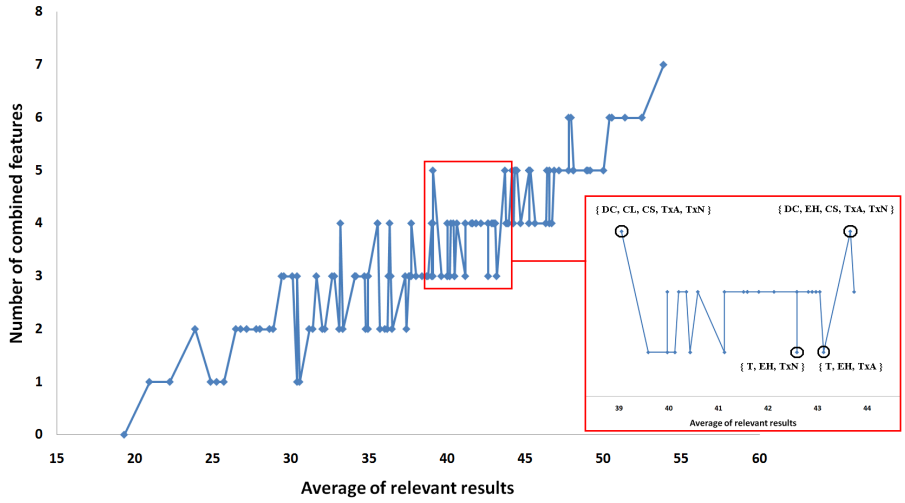

Fig. 2. Average number of relevant documents displayed in the interface with respect to the number of combined features used

Finally, we present in Table 3 the best combination of features to obtain the best relevance for the faceted browser. A "×" means that we do not use the feature in the combination and a "-" means that the feature is part of the combination. Each column represents a feature. We denote "DC", "CL", "T", "EH", "CS", "TxA" and "TxN" for dominant colour, colour layout, texture, edge histogram, contour shape, text query expansion adding 2 keywords and text query with 4 new keywords, respectively. The last column shows the average number of relevant documents per topic denoted "ARD". The first row shows the baseline run with no combination of facets, the second row presents the best combination of three feature, the next rows show the top combination of features. Thus, the last row shows the results of all features combination.

Observing these results, we conclude that colour layout and texture are the best visual features as they are almost always used in the top combination of features. It can also be observed that the feature contour shape is almost useless as we improve the average number of new relevant documents per topic by only one in the combination (see the difference between the two last rows of the table). Another interesting conclusion can be drawn when comparing the initial text query that has only an average of 19.3 relevant documents per topic with the best three combination of features ($ARD = 43.1$) or with all combined features ($ARD = 53.8$). We can double the effectiveness of the faceted browser using one of the best combination of up to three facets/features and almost triple its effectiveness with a combination of all features.

3.5 Discussion

In this section, we have presented various experiments which aim at showing the potential benefits of the faceted browser. Our results highlight the fact that new facets provided by iterative calls of the clustering algorithm might increase the precision of the retrieved results and have a higher probability of displaying new relevant documents in new facets of the interface.

We have evaluated all possible combinations of the best simulated facets representing one feature each which shows the real potential of the faceted browser. The number of relevant documents displayed doubles for a combination of three facets and almost triples with all facets.

Our fundamental premise in our simulated study is that users do actions that maximise the retrieval of relevant documents. For example, in a interactive user scenario, we assume that the users choose better relevant clusters or keywords to add to a new facet. He or she may also easily delete a facet that does not correspond to their search task, which we presume will result in much better results with real user interactions than with our simulated clustering methodology.

4 Exploiting User Experiments

In order to verify the above results, we conducted another set of simulated experiments based on logged data of a user experiment on the system described in Section 2.2. The user study was aimed at sudying the user perception, satisfaction and performance using the faceted browser. A brief overview is provided in Section 4.1. Exploiting the logfiles of this user study, we introduce and evaluate a new retrieval model which updates search queries by incorporating the content of other facets. The approach will be introduced in Section 4.2.

4.1 User Experiment

In the user experiment [7], two tasks were defined, aiming to reflect two separate broad user needs. Task A is the more open of the two tasks, and asks the user to discover material reflecting international politics at the end of 2005 (the period of time covered by the TRECVID 2006 data). Task B asked for a summary of the trial of Saddam Hussein to be constructed, including the different events which took place and the different people involved (such as the judge). This later task, which is still multi-faceted, was less open ended than the former task. Fifteen subjects took part in the study. Six users performed search Task A and nine participants performed search Task B for 30 minutes and filled in a questionnaire.

4.2 Methodology

Identifying Usage Patterns After performing the initial user study, we analysed the resulting logfiles and extracted user behaviour information. The following data was captured in the logs: *Creating* and *Deleting* a new facet, triggering

a new*Search* in a facet, *Moving from facet* a shot from the relevance list of facet F_1 to a different facet F_2, *Dragging from player* a shot directly onto a relevant results list of a facet, and finally, *Dragging from results* a shot onto a relevance list.

The log entries provide us with information about the users' interaction behaviour such as: when a user created a new facet, which search query he/she triggered or which results he/she judged to be relevant for this particular facet. We exploited these information in our simulation process. In the following section, we use these patterns to study how facet based browsing can influence the retrieval performance in repeating users' interaction steps and updating the retrieval results.

Relevance Judgements Since Tasks A and B are not from TRECVID, ground truth data for our simulation was based on pooling all sets R_i of shots d moved to the relevance list by user i. Let $\boldsymbol{d}_K =$ be a vector representing shot K, defined as

$$\boldsymbol{d}_K = \{d_{K1}...d_{KN}\}, \text{where } N \text{ is the number of users } and\ d_{Ki} = \begin{cases} 1, & \boldsymbol{d}_K \in R_i \\ 0, & \text{otherwise} \end{cases}$$

Using:

$$F_1(\boldsymbol{d}_K) = \begin{cases} 1, & (\sum_{i=1}^{N} \boldsymbol{d}_{Ki}) = 1 \\ 0, & \text{otherwise} \end{cases} \quad F_2(\boldsymbol{d}_K) = \begin{cases} 1, & (\sum_{i=1}^{N} \boldsymbol{d}_{Ki}) \geq 2 \\ 0, & \text{otherwise} \end{cases}$$

we created two relevance judgement lists:

$$L_1 = \{d_K : F_1(\boldsymbol{d}_K) = 1\}\ and\ L_2 = \{d_K : F_2(\boldsymbol{d}_K) = 1\} \tag{1}$$

(Assuming that a keyframe is relevant within the given topic when is was selected by any user for L_1 and by at least two users for L_2.)

Simulation Strategies The retrieval model of our user study was simple: the users enter textual search queries in each facet and the backend system returns a list of shots which are represented by a keyframe in the result list of the facet. Users interacted with the result list by selecting relevant shots, playing a shot, creating facets, etc. However, user feedback such as selecting a shot as relevant for this facet or the content and status of other facets are not used in retrieving or suggesting new facets. Hence, we use the user study as a baseline run B and try to improve its retrieval performance by introducing a new retrieval model which incorporates the content of other facets.

Our simulation procedure uses the following steps. First of all, we analysed the user queries in the log files and confirmed that users took advantage of the facets and used them to search for variations of the same concept. For instance in Task A (international politics), participants used the facets to search for

different politicians, i.e. "George Bush" in facet F_1 and "Tony Blair" in facet F_2. We concluded that facets were used to focus more on specific sub concepts of each topic. Following the identified pattern, we performed a simulation run S.

In this run, we took advantage of the explicit relevance feedback given by each user in marking shots as relevant for a facet. We used these shots as a query expansion source and determined query candidate terms for each iteration in each facet by expanding queries from the relevant rated keyframes at step x. If a term appears in more than one facet within this step x, we removed it from the facet which contained more candidate terms and used these candidate terms as a new search query. In other words, we reduce the number of query terms in a facet, when the query term is used in another facet with less query terms at the same time. This results in a more focused retrieval for the facets, as double entries will be avoided.

4.3 Results

For evaluating the performance of our baseline system and the simulation runs, we firstly divided the users's search sessions into separate steps, being the beginning of a new iteration in any facet. For each step, we then combined the result lists of each facet in its current iteration.

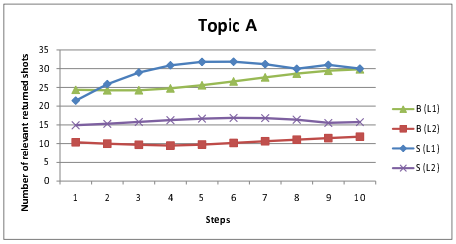

Fig. 3. Number of relevant returned results over all steps in Topic A

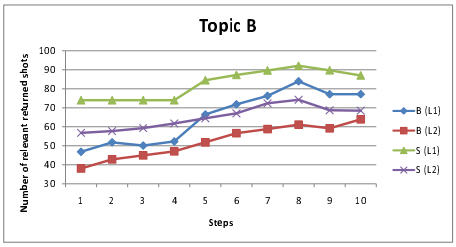

Fig. 4. Number of relevant returned results over all steps in Topic B

In a next step, we evaluated our runs using the two created relevance judgement lists L_1 and L_2 as introduced in Section 4.2. Figures 3 and 4 show the mean number of relevant retrieved results over all steps in Topic A and B, respectively. As expected, using the relevance judgements list L_1 returns a higher retrieval performance in all cases than using L_2. This matches with common sense, a larger list of relevant documents used for evaluation results in a higher number of relevant retrieved documents. The decreasing number of retrieved shots in some cases is the direct consequence of users closing facets in later steps of their retrieval session. The results within these facets hence get lost, resulting in a decrease of retrieved results.

It can be seen for both search tasks,that the simulation run S outperformed the baseline run B, which indicates that considering the content of other facets to re-define a user's search query can improve the retrieval performance. Hence, a retrieval model which takes the content of other facets into account can outpferm a classical "one-resultlist only" model.

5 Conclusion

In this paper, we have introduced a facet-based approach to interactive video retrieval. We employed clustering techniques to identify potential facets and used in our simulation. The results of our study demonstrate the potential benefits of a faceted search and browsing system. In addition to the results of our simulated evaluation on the TRECVID collection, we have explored the logs of a real user-centred evaluation and the results corroborate that of the simulation methodology.

The experiments were conducted on a large data set given by the TRECVID and hence support the validity of our experiments. However, it is well known that the TRECVID search topics are so diverse and the issue of performance variation from topic to topic. This explains some of the performance problems we encountered in some of the topics. In addition to this, simulated methodologies are one end of spectrum of a series of evaluations needed before multimedia systems are deployed. It allows us to benchmark various retrieval approaches and also search strategies like the faceted browsing. This results need to be verified by the use of a real user-centred evaluation which we are exploring now.

6 Acknowledgments

This research was supported by the European Commission under the contracts FP6-027026-K-SPACE and FP6-027122-SALERO.

References

1. M. Christel and R. Concescu. Addressing the challenge of visual information access from digital image and video libraries. In *JCDL (Denver, CO)*, pages 69–78, 2005.
2. T. W. Finin. GUMS: A General User Modeling Shell. *User Models in Dialog Systems*, pages 411–430, 1989.
3. D. Heesch, P. Howarth, J. Magalhães, A. May, M. Pickering, A. Yavlinski, and S. Rüger. Video retrieval using search and browsing. In *TREC2004*, 2004.
4. F. Hopfgartner and J. Jose. Evaluating the Implicit Feedback Models for Adaptive Video Retrieval. In *ACM MIR '07*, pages 323–332, 09 2007.
5. A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, 1988.
6. A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR '06*, pages 321–330, New York, NY, USA, 2006. ACM Press.
7. R. Villa, N. Gildea, and J. M. Jose. A Faceted Search Interface for Multimedia Retrieval. In *SIGIR'08*, pages 775–776, New York, NY, USA, 2008. ACM Press.