TV Scout: Guiding Users from Printed TV Program Guides to Personalized TV Recommendation

Patrick Baudisch[1]

Inf. Sciences and Technologies Lab.
Xerox Palo Alto Research Center
Palo Alto, CA 94304, U.S.A.
+1 (650) 812 4656
patrick.baudisch@acm.org

Lars Brueckner[1]

IT Transfer Office (ITO)
Darmstadt University of Technology
64283 Darmstadt, Germany
+49 (6151) 16-6217
brueckner@ito.tu-darmstadt.de

In this paper, we present *TV Scout*, a recommendation system providing users with personalized TV schedules. The TV Scout architecture addresses the "cold-start" problem of information filtering systems, i.e. that filtering systems have to gather information about the user's interests before they can compute personalized recommendations. Traditionally, gathering this information involves upfront user effort, resulting in a substantial entry barrier. TV Scout is designed to avoid this problem by presenting itself to new users not as a filtering system, but as a retrieval system where all user effort leads to an immediate result. While users are dealing with this retrieval functionality, the system continuously and unobtrusively gathers information about the user's interests from implicit feedback and gradually evolves into a filtering system. An analysis of log file data gathered with over 10,000 registered online users shows that over 85% of all first-time users logged in again, suggesting that the described architecture is successful in lowering the entry barrier.

Introduction

Information filtering systems [7] suffer from a bootstrapping problem. Before they can give personalized recommendations to a user, they have to find out what the user's interests are. Only then can filtering systems build user profiles and compute personalize recommendations. The problems resulting from this undesirable order of required user effort and delayed benefit is a well-known phenomenon in collaborative filtering, the so-called *cold start* problem [17]. Users are reluctant to invest effort, especially if they don't know whether the offered service will be worth the effort. This approach bears the risk that users will avoid the gamble and stick with a system offering more immediate benefit, such as a retrieval-oriented system. Users making this decision, however, will never come to discover the long-term benefits the filtering system would have offered. For additional studies on incentive structures and the results of the lack of incentives see [13].

In this paper, we describe an architecture designed to address this incentive problem and we will demonstrate this architecture at the example of our TV program recommendation system TV Scout. We will begin by briefly introducing the field of

[1] The work presented in this paper was carried out during the authors' affiliation at GMD-IPSI

TV recommendation. We will then discuss TV Scout and its user interface and discuss the underlying filtering architecture. Finally, we will report results of an analysis of TV Scout online usage data, discuss our findings, and present conclusions and future work.

Recommending TV Programs

In 1992, Belkin and Croft wrote "In particular, applications such as the recreational use of television programming pose special problems and opportunities for research in filtering" [7, p.37]. Several current trends make TV an interesting application area for information filtering. TV viewers are facing an information overload situation [10]. A number of technical improvements, such as cable, satellite, and digital TV technology have resulted in an increasing number of available TV channels. Today, hundreds of channels broadcast thousands of programs every day. Since the amount of content that is of interest for a given viewer has not increased proportionally, *planning* ones TV consumption has become a challenge. The amount of TV programs will soon exceed the limits of what can reasonably be printed and channel surfing is no longer fast enough to allow getting an overview of all channels [11]. Attempting to meet the changing requirements, web-based TV program guides (e.g. TV Guide, http://www.tvguide.com), set-top boxes with electronic program guides (EPGs, [20]), and digital VCRs (e.g. *Tivo* http://www.tivo.com) have emerged in the past few years.

There have been several research projects around TV recommendation in the past [11, 9], but most of them focused on set-top boxes and on the technical possibilities for monitoring user behavior rather then on web-based systems and usability. Current research in personalized TV evolves still around personalized EPGs [1], but also around new concepts, such as multi-agent recommender systems [14]. A more thorough overview of current research in the field of personalized TV recommendation can be found in [18].

TV Scout

TV Scout [3, 4] is a web-based TV recommendation system. Its goal is to support users in planning their personal TV consumption.

In order to understand the design requirements for such a system, we began our research with an informal survey among students [3]. The survey indicated that expectations about the functionality of an ideal TV recommendation system were dominated by experiences with printed TV program guides. While our goal was to eventually provide users with a personalized TV program at a single mouse click, our survey indicated that only a minority of the users we had interviewed would be willing to invest the required effort. We concluded that in order to attract users, a successful TV recommendation system would first have to emulate the expected print-like functionality, as well as the straightforward usage of printed guides: pick up the TV guide, find today's listing, pick a program, and watch TV. The challenge was to provide a seamless transition from this scenario to the filtering functionality we had in mind. To prevent the filtering functionality from conflicting with the user expectations and system learnability, we decided to create a system that would progressively disclose its filtering features to users.

Implementation

The TV Scout project was conducted in cooperation with the TV program guide publisher TV TODAY. While this resulted in TV Scout getting implemented as a web-based system, we see no architectural problems in porting the resulting architecture to a set-top box. To allow maintaining personal user profile data, first-time users have to create an account, which they access using a self-selected login name and password. The web-based TV Scout front end is implemented in HTML, Java, and JavaScript.

Figure 1: How TV Scout presents itself to first-time users (screenshots partially translated from German)

Retrieving Program Descriptions

To users entering TV Scout for the first time, the system presents itself as a retrieval system. Its functionality at this stage restricts itself to the functionality of a printed TV program guide, with a graphical user interface. Users specify a query (or simply hit a button for the default "what's on now"), sort through the resulting list and select programs to watch. Users can also print the list of selected programs for later use.

Figure 1 shows how users accomplish that using the TV Scout user interface. The interface consists of the menu frame on the left and the content frame on the right. The menu frame provides users with access to all retrieval and filtering functions and is permanently visible. The content frame is used to display various types of TV listings and all profile editing tools.

The system is used as follows. Users execute a query by picking a query from the query menu. Figure 2 shows several close-ups of this menu. In its current version, TV Scout offers four query groups: *text search*, *genres*, *user tips*, and *TV TODAY tips*, plus a *favorites* group that we will explain later. Text search allows users to search for keywords using optional Boolean syntax. The other three submenus are executed by picking the corresponding menu entry. To provide more precise queries, theses query groups contain hierarchies of submenus that can be browsed in a file system explorer-like fashion. *Genres* contains a historically grown genre classification of TV programs, such as *sports*, *comedy*, and *series* [14]. *User tips* contains recommendations volunteered by users who serve as self-proclaimed editors, so-called *opinion leaders* [4]. Finally, *TV TODAY tips*, are recommendations provided by the editors of TV Scout's printed counter part.

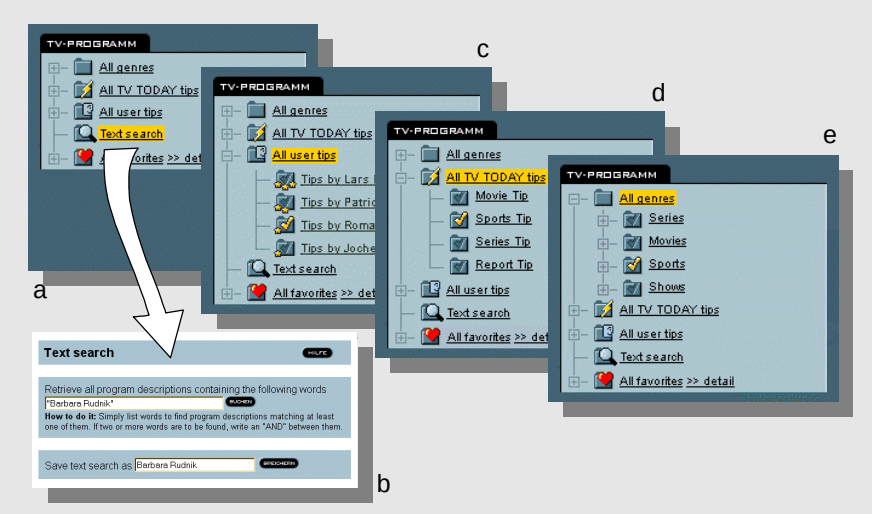

Figure 2: The query menu offers four groups of queries

By default, all queries are restricted to the programs starting within the current hour, but TV Scout provides customized controls that allow specifying arbitrary time

and date intervals using mouse drag interactions (Figure 4a). Channels can be selected from two predefined sets or can be selected in detail using a paintable interface (Figure 4b) [6].

When a query is executed, the resulting set of TV program descriptions (Figure 1 bottom left) is displayed in the content area. Descriptions consist of the program title, a rating describing how well the program matches the query, an extract of the program description, and links to a more detailed description. Users can choose between the display styles *ranked list* and *table*.

Two toggle switches per program description allow users to retain programs they plan to watch in the so-called *retention tool* (Figure 1 bottom left, circled). The retention tool *laundry list* can be used to print a list of programs; *video labels* are designed to retain and print programs to be videotaped. The retention menu allows users to display the content of their retention tools for reviewing or printing. The printed list can be used to remind users of the programs that they plan to watch.

Filtering Functionality: Creating "Bookmarks"

Using the functionality described so far, the effort for repeated usage is the same each time the service is used. The next step therefore is for the system to reduce the effort required of the user when querying, since the primary purpose of IF systems is to be "time-saving devices" [2].

When a user enters a query that is broader than necessary, the user is forced to sort through an unnecessarily long listing when trying to find desired programs. When the system detects that the user has used such a sub-optimal query repeatedly while another query with better precision exists, it makes a suggestion. Figure 3 shows an example. Let's assume that the user has repeatedly used the query "movies" to exclusively find and retain comedies and horror movies. By computing the overlap between the retained programs and all available queries [3], the system detects that the retained programs can also be covered by the more specific queries "horror movies" and "comedies". A dialog box opens and suggests using these queries instead. The user can execute the suggested queries like any other query, i.e. by clicking their names.

The more important function of the dialog box, with respect to our filtering concept, is that it also suggests retaining these queries as *bookmarks*. Users can do this by clicking the toggle switch that accompanies each query (a folder symbol with a check mark, see Figure 3a). Retained queries pop up in the user's *favorites* (Figure 3b). The *favorites* folder is collocated with the other query groups and can be executed the same way. Retained queries are listed in a flat hierarchy, thereby providing the users with convenient access to queries that would otherwise be hidden in multiple different submenus. This functionality corresponds to the bookmark folder in a web browser. Unlike web bookmarks these bookmarks are stored on the TV Scout server, allowing TV Scout to use them as input for additional computation.

Retention check boxes accompany all queries in the system (see Figure 3b), so users can bookmark queries anytime, independent of suggestions. The primary purpose of query suggestions is to inform users about the bookmaking concept and to encourage its usage.

Note the special importance of the retention tools. Although the declared purpose of the retention tools is to allow users to memorize programs and print schedules,

their primary purpose from the system's point of view is to serve as an information source about the user's interests. The content of the retention tools is considered an implicit positive rating for the retained programs, making the retention tools serve as a source of implicit retention feedback [16]. Although implicit feedback is commonly agreed to be a less reliable source of rating information than explicit feedback, it has the benefit of being unobtrusive, which we considered essential for this type of filtering system. See [3, 4] for how TV Scout uses the same implicit input for various types of filtering functionality based on collaborative filtering.

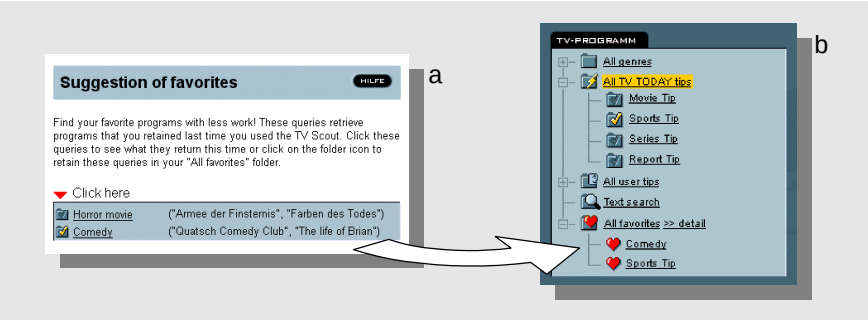

Figure 3: By clicking a checkmark-shaped button, queries can be retained in *All favorites*.

Filtering Functionality: One-click Personalized TV Schedules

To provide a container for bookmarked queries is not the only purpose of the *favorites* folder. The real value of this folder lies in the fact that users can execute it as a whole by clicking the top menu entry labeled *all favorites*. This executes all retained queries at once. The result listings of the individual queries, however, are not appended to each other—they are merged into a single relevance-ordered result list. This is the most powerful function of the TV Scout system—it fulfills the initial goal of generating personalized TV schedule with a single mouse click.

How are the individual query results merged in order to obtain a useful result? When the query profile *all favorites* is executed, a script running inside the TV Scout server executes all contained queries. This is done by delegating each query to the corresponding subsystem; text search, for example, is executed by FreeWAIS, while genre queries are executed by a relational database. As a result, the subsystems deliver sets of pairs (program, rating). The task of the query profile script is to merge all these results into a single ranked list. This requires transforming the individual ratings such that they include the user's perceived importance of the interest represented by the query. In order to express this perceived importance, the query profile stores a linear function (i.e. a factor and an offset) for each retained query. The resulting ratings are computed by transforming the ratings returned by the subsystem using this function. If a TV program is returned by multiple queries its ratings are summed up. Finally, programs are sorted by their result rating and returned to the user.

The critical factor is the parameters of the linear transformation. The system acquires these parameters through initialization, learning, and manual updating. When

queries are bookmarked, their functions are initialized using Zipf's law [19, p. 60]. This means that more specific queries are given positive offsets, propagating the results of these queries towards the top ranks of the resulting listings, thus preventing them from being buried inside the large result sets of less specific queries.

After initialization, the parameters of the rating transformations can be improved by two means. First, TV Scout continuously optimizes the query profile based on the same implicit retention feedback that was already used for suggesting queries. See [3] for a description of the algorithm. Second, interested users are allowed to manually inspect and update their profile. Clicking the ">>details" link in the *all favorites* menu invokes a profile editor. The simplest version of this editor provides users with a single pull-down menu per query (Figure 4c), allowing users to assign a symbolic rating to each query, such as "*Action movies* are [very important] to me" [3, 5].

Through the use of relevance feedback the query profile improves continuously, so that the quality of the rankings obtained by clicking *all favorites* increases over time.

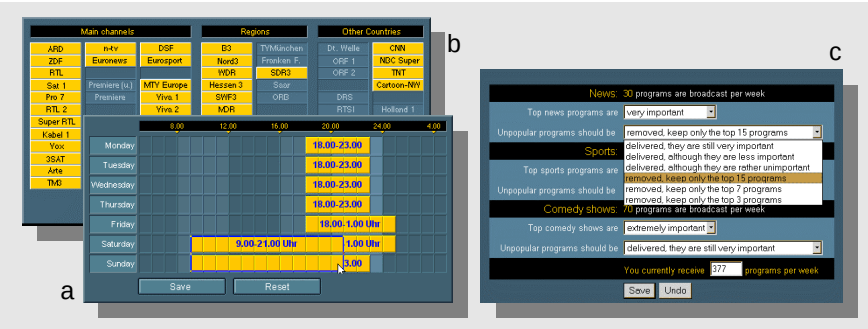

Figure 4: The TV Scout profile editing tools (a) viewing time profile editor, (b) channel profile editor, and (c) query profile editor.

Summary

Figure 5 summarizes how the usage of TV Scout by a given user can evolve over time. Each transition to a more personalized phase can be suggested by the system (T1-T3) or initiated by the user (U1-U3). However, users are not forced through these phases and may equally well settle with the functionality of one of the earlier phases.

1. **Query phase (S1):** Users can pick predefined queries (T1) or can formulate queries, such as text searches, manually (U1).
2. **Bookmark/reuse phase (S2):** If the system detects reoccurring or sub-optimal queries it proposes better-suited queries and suggests retaining them as *favorites* (U2). Independent of suggestion, users can bookmark queries anytime (T2).
   Profile creation (T*): The user's query profile is created automatically when the first query is bookmarked.
3. **Profile phase (S3):** Initially, the query profile provides users with a convenient way of executing all their bookmarks with a single click. Continuous supply of relevance feedback (T3) or manual profile manipulation (U3) improves the profile.

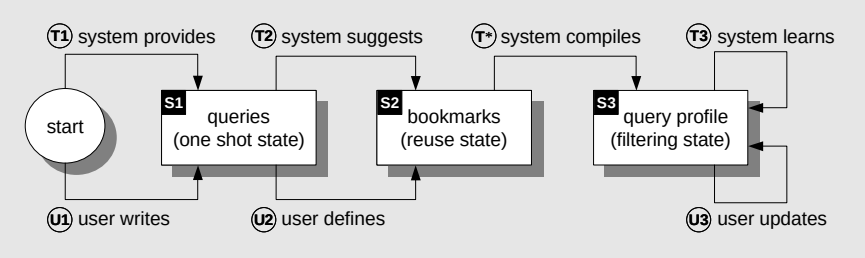

Figure 5: Evolving usage of a proposed filtering architecture

TV Scout Usage Data

The purpose of the TV Scout design is to reduce the entry barrier for new users by using a progressive disclosure of the filtering functionality. How can we verify the success of our interaction design? A controlled experimental comparison with a competing system would be problematic because of the vast amount of interface variables that would be difficult to control. In addition, modeling a realistic web-usage scenario in a lab setting is challenging. Alternatively, a naturalistic study of web use would provide more realistic data, but we would be unable to measure factors such as subjective satisfaction. Ultimately, we decided to conduct an informal analysis of log file data from actual web usage.

When we conducted our data analysis April 20, 2000, TV Scout had been publicly available at for 18 months. The entire 18 months of log file data are included in this study. All usage data was extracted from the web server log files and the system's database. With respect to the filtering functionality, this data was slightly biased, in that the suggestion feature became available later. Because of this, we expected that the usage of bookmarking would be underrepresented.

The main purpose of the analysis was to verify whether our filtering system design fulfilled the primary goal, namely to provide a low entry barrier. If our design was appropriate, then TV Scout would meet the expectations of first-time users and would not overwhelm them. Repeated usage would indicate that users had taken the entry hurdle; one-shot users would suggest the opposite.

We were also interested in learning more about the users' demand for the offered filtering functionality. How many users would adopt bookmarking functionality; how many would make use of their personal query profiles? Based on our informal survey, we expected the majority to be satisfied with the initial retrieval functionality, but we had no clear expectations about the percentages. Finally, we were interested in seeing how useful users would find the query profile. Once they had created one, would they continue to use it or would they abandon it rapidly?

Results

At the day we examined the log data, TV Scout had 10,676 registered users. In total, users had executed 48,956 queries. 53% of all queries (25,736 queries) were specific queries different from the default query.

Repeated log-ins: We found that 9,190 of the 10,676 registered users had logged in repeatedly, i.e. twice or more. This corresponds to a percentage or 86% repeated users. The most active user with 580 logins had logged in almost daily.

Bookmarks: 1770 users had bookmarked one or more queries. Together, these users had bookmarked 4383 queries, mostly genres. The most frequently executed queries were the genres movies (736 times) and information (364 times), and TV TODAY Movie tips (369 times). Over 300 text searches were bookmarked.

Query profiles: Out of the 1770 users who had bookmarked at least one query, 270 users (about 15%) executed their query profile at least once to obtain personalized listings. These users executed their query profiles a total of 5851 times, which corresponds to an average of 21 times per user. These users manually fine-tune their profiles a total of 1213 times, with an average of 4.5 times per user. These results indicate that query profiles were highly appreciated by those who used them.

Conclusions

We interpret the measured percentage of repeated users as a confirmation of our design. 86% of all first time users logged in repeatedly; we consider this to be a very high percentage for a web-based system. This indicates that presenting first-time users with a retrieval setting is a successful approach to keeping the entry barrier for first-time users low.

Only 17% of users made use of the bookmark feature; out of these, only 15% made used of the query profile. These numbers seem low even taking into account that the suggestion feature was not available most of the logged time. Does this result indicate that the filtering functionality is inappropriate or difficult to learn? Why did the majority of the users not reach the "goal" of the system?

This is not how we interpret these results. In an earlier TV usage survey we conducted [3] we found TV users to plan their TV consumption for very different timeframes. Most of these users only planned a TV schedule for the following day or they did not plan at all. Many users only used a guide to determine what was currently on TV. Only 12% of the users planned a TV schedule for the entire week. Considering that the filtering functionality of TV Scout addresses the relatively small subgroup of users who plan their TV consumption, the observed results seem appropriate. The majority of users who only used the retrieval functionality may have found the retrieval functionality of TV Scout to be the appropriate support for *their* information seeking strategy. An online survey as well as an experimental study should help to verify this interpretation.

Acknowledgements
We would like to thank Dieter Böcker, Joe Konstan, Marcus Frühwein, Michael Brückner, Gerrit Voss, Andreas Brügelmann, Claudia Perlich, Tom Stölting, and Diane Kelly.

References

1. L. Ardissono, F. Portis, P. Torasso. F. Bellifemine, A. Chiarotto and A. Difino Architecture of a system for the generation of personalized Electronic Program Guides. In *Proceedings of the UM 2001 workshop "Personalization in Future TV"*, July 13 to July 17, 2001, Sonthofen, Germany.

2. P.E. Baclace. Information intake filtering. In *Proceedings of Bellcore Workshop on High-Performance Information Filtering*, Morristown, NJ, November 1991.
3. P. Baudisch. Dynamic Information Filtering. Ph.D. Thesis. GMD Research Series 2001, No. 16. GMD Forschungszentrum Informationstechnik GmbH, Sankt Augustin. ISSN 1435-2699, ISBN 3-88457-399-3. Also at http://www.patrickbaudisch.com/publications
4. P. Baudisch. Recommending TV Programs on the Web: how far can we get at zero user effort? In Recommender Systems, Papers from the 1998 Workshop, Technical Report WS-98-08, pages 16-18, Madison, WI. Menlo Park, CA: AAAI Press, 1998.
5. P. Baudisch. The Profile Editor: designing a direct manipulative tool for assembling profiles. In *Proceedings of Fifth DELOS Workshop on Filtering and Collaborative Filtering*, pages 11-17, Budapest, November 1997. ERCIM Report ERCIM-98-W001.
6. P. Baudisch. Using a painting metaphor to rate large numbers of objects. In Ergonomics and User Interfaces, Proceeding of the HCI '99 Conference, pages 266-270, Munich, Germany, August 1999. Mahwah: NJ: Erlbaum, 1999.
7. N.J. Belkin and W.B. Croft. Information filtering and information retrieval: two sides of the same coin? CACM, 35(12):29-37, Dec. 1992.
8. A. Borchers, J. Herlocker, J. Konstan, and J. Riedl. Ganging up on information overload. Computer, 31(4):106-108, April 1998.
9. D. Das and H. ter Horst. Recommender systems for TV. In Recommender Systems, Papers from the 1998 Workshop, Technical Report WS-98-08, pages 35-36, Madison, WI. Menlo Park, CA: AAAI Press, 1998.
10. P. Denning. Electronic junk. CACM, 23(3):163-165, March 1982.
11. M. Ehrmantraut, T. Härder, H. Wittig, and R. Steinmetz. The Personal Electronic Program Guide—towards the pre-selection of individual TV Programs. In Proc. of CIKM'96, pages 243-250, Rockville, MD, 1996.
12. L. Gravano and H. García-Molina. Merging ranks from heterogeneous Internet sources. In Proceedings of the 23rd VLDB Conference, pages 196-205, Athens, Greece, 1997.
13. J. Grudin. Social evaluation of the user interface: who does the work and who gets the BENEFIT? In *Proc. of INTERACT'87*: pages 805-811, 1987.
14. M. Kuhn. The New European Digital Video Broadcast (DVB) Standard. ftp://ftp.informatik.uni-erlangen.de/local/cip/mskuhn/tv-crypt/dvb.txt.
15. K. Kurapati, S. Gutta, D. Schaffer, J. Martino and J. Zimmerman. A multi-agent TV recommender. In *Proceedings of the UM 2001 workshop "Personalization in Future TV"*, July 13 to July 17, 2001.
16. D.M. Nichols. Implicit ratings and filtering. In *Proceedings of Fifth DELOS Workshop on Filtering and Collaborative Filtering*, pages 31-36, Budapest, November 1997. ERCIM Report ERCIM-98-W001. Le Chesnay Cedex, France, European Research Consortium for Informatics and Mathematics, 1998.
17. P. Resnick and H. Varian (Eds.). Special issue on Recommender Systems. Communications of the ACM, 40(3):56-89, March 1997.
18. Proceedings of the UM 2001 workshop "Personalization in Future TV", July 13 to July 17, 2001, Sonthofen, Germany. Online at http://www.di.unito.it/~liliana/UM01/TV.html
19. G. Salton and M.J. McGill. *Introduction to Modern Information Retrieval*. New York: McGraw-Hill, 1983.
20. H. Wittig and C. Griwodz. Intelligent media agents in interactive television systems. In *Proc. of the International Conference on Multimedia Computing and Systems '95*, pages 182-189, Boston, May 1995. Los Alamitos, CA: IEEE Computer Science Press, 1995.