ELSEVIER

Displays 18 (1998) 215–220

DISPLAYS

Let's talk about it: dialogues with multimedia databases Database support for human activity

Arjen P. de Vries, Gerrit C. van der Veer, Henk M. Blanken

Centre for Telematics and Information Technology, University of Twente, P.O. Box 217, 7500 Enschede AE, The Netherlands

Abstract

We describe two scenarios of user tasks in which access to multimedia data plays a significant role. Because current multimedia databases cannot support these tasks, we introduce three new requirements on multimedia databases: multimedia objects should be active objects, querying is an interaction process, and query processing uses multiple representations. We discuss three techniques to handle multimedia objects as active objects. Also, we introduce a promising database architecture to meet the new user requirements. Agents within the database handle objects' representations, and a search engine on top of a conventional database handles relevance feedback and multiple representations. © 1998 Elsevier Science B.V.

Keywords: Multimedia databases; Multimedia modeling; Human–computer interaction; Content-based retrieval; Relevance feedback

1. Introduction

People deal with multimedia data every day. Every time we read a book, watch television or listen to some music, we work with multimedia data. Moreover, we organize and structure this information for ourselves such that we can easily retrieve this information when needed. We create photo albums of our holidays, we possess racks of compact discs and tapes with the music we like, we store past editions of magazines in boxes, and use a video recorder to record television programmes about topics of interest. For people with professions like fashion designer or journalist, the amount of information collected is even higher, and the retrieval task is more difficult.

Since the introduction of multimedia in personal computers, we can easily digitize part of our information. People now create their own homepages on the world-wide web as a means to manage the information they collect. A major advantage over shoeboxes stored in the attic, is that we can easily share our data collection with others. However, one look at the web makes clear that a computer with a web server is not the best tool to share our shoebox data. It is not easy to find what you want and the information *that* you find is often incorrect or has been moved to another location.

Database technology provides means to store and retrieve high volumes of data. However, until recently, we could not use databases for anything more advanced than names and numbers. Nowadays, we read a lot about multimedia databases. Unfortunately, anything that simply *stores* multimedia data is called a multimedia database. The capabilities of such databases suffice for typical applications of real estate and travel businesses, as these systems only deal with the presentation of otherwise statically used information. A *real* multimedia database should provide much more functionality than just storage and presentation. In this paper, we introduce three new requirements on database management systems, to make them useful tools for handling multimedia data.

2. Example scenarios of user tasks

To illustrate the real-life application of multimedia database systems, we outline two scenarios of user tasks, to demonstrate the functionality that the end user should get from a multimedia database system.

In the first scenario, imagine a journalist writing an article about the effects of alcohol on driving. Before he can start to do the actual work of writing the article, he has to collect newspaper articles about recent accidents, scientific reports giving statistics and explanations, photographs, television commercials broadcast for the government, and interviews with policemen and medical experts.

The second case focuses on a fashion designer developing a concept for a dress to be worn by the receptionists of some big retail office. To succeed in this creative design task, he first collects many different multimedia objects. The designer needs descriptions and pictures of the retailer's

0141-9382/98/$19.00 © 1998 Elsevier Science B.V. All rights reserved
PII S0141-9382(98)00023-7

products, video fragments of buyers at the premises, photographs revealing details of the entrance and reception area, advertisements in magazines, commercials on television, video and audio fragments of "vision development breakfasts", and many other pieces of information associated with the retailer. The designer also browses through previous designs, studies preferred dresses from colleagues, and views some videos of recent developments in fashion design.

It is easily understood that the people in both scenarios deal with large amounts of multimedia information to accomplish their tasks. Fashion designers working alone may not need advanced information technology. Piles on their desks and shoeboxes with old designs may provide easier ways to handle the data. However, design tasks are typically performed by a team of designers. Even if these people work at the same time in the same room, they would still need a tool to find what they need in the "organized mess" of the other team members.

3. Searching new media objects

Both user scenarios demonstrate that the key functionality a multimedia database should offer is access to multimedia information. With respect to access, Bertino et al. classified multimedia objects into two classes: *active* objects and *passive* objects [1]. Active objects really participate in the retrieval process. Users can specify conditions on active objects in the query, referring to the content or referring to the existence. Passive objects just exist in the database. It is not possible to condition on the content of passive objects. Most information systems that claim to be multimedia databases view images, audio and video objects as passive objects. Users can look at picture number 1500, or play the audio stream related to object 120 in WAV format. But they cannot search for "pictures like these", or "interviews about alcohol and driving".

If a database handles multimedia objects as passive objects, it is not more than a (huge) collection of multimedia data. A tool that mainly stores data is not a database, but a file system. It clearly does not meet the requirements of the fashion designer or the journalist from the previous section. Both scenarios demonstrate a need to condition on the content of the objects.

Therefore, in a true multimedia database system, all objects should be active objects. We want to use multimedia databases with photo and music collections like we use conventional databases to manage phone numbers. We do not just store names and phone numbers, and then check all records sequentially each time we want to call John. Instead, we simply ask the database system for John's number. We use a database system as a tool to recollect unknown properties of stored entities using some known properties.

Unfortunately, the properties of digitized multimedia objects are not as easily checked as the properties of numbers or strings. Applying an exact match on two digitized objects will only retrieve another object if it is bit-for-bit exactly the same. The question arises as to why you would search for a digitized object that you already used to formulate the query. It could be useful to find other properties of a multimedia object, similar to searching for the phone number using a person's name. Imagine the police officer who needs the name of the criminal he recognized from a photograph. However, in most practical situations, we do not have the exact picture that resides in the database. Hence, we need other means to handle the multimedia data as active objects.

3.1. Manually added descriptions

The straightforward approach to using multimedia objects is to manually add a textual description of the object. We know how to search using textual descriptions. An extra advantage of this option is that the search engine would be independent of the media type of the objects in the database. We could apply the fairly well understood text retrieval technology to search also for images and audiovisual data.

Obviously, manual indexing is rather expensive if we deal with a large amount of data. This approach is also problematic in three more fundamental ways. The common cause underlying these problems is the limited capability of capturing the full semantics of multimedia data in textual descriptions.

First, it is not likely that people describe objects with keywords in a standardized manner. Different people select different words to describe the same concepts. For example, one person may describe a picture of "an evening in the mountains" as "dark", while another person describes the same picture as "sombre". Both try to express approximately the same concept, but if the first searches for the picture in a database collected by the other, he will not find the picture even though it is in the database.

We may partly overcome ambiguity in natural language by using thesauri. However, the second problem cannot be solved with thesauri. Different people describe different aspects of the picture. The same picture described as "dark" may be associated with "evening" and "Mount Snowdon" by an enthusiastic hiker. Also, even a single person uses different descriptions depending on the specific situation when asked. In psychology, this is known as the *encoding specificity principle* [2]. For example, a hiker describes the picture with "dark" in his office during the week, but he writes down "evening" in his living room at the weekend.

Finally, substantial evidence exists that some semantic properties of multimedia objects cannot unambiguously be expressed verbally. In his book, Iaccino reviews psychological research in differences between the two hemispheres of the brain with respect to perception [3]. While the left hemisphere is verbal and analytic, the right hemisphere is nonverbal and holistic. Each hemisphere is specialized for a different kind of thinking or *cognitive style*.

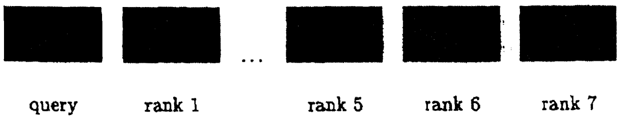

Fig. 1. Image retrieval based on colour features.

According to Barrow [4], the composer Carl Orff never admitted a boy to the Vienna Boys' Choir if he already knew how to read and write. Apparently, he believed that analytical skills block the creative processes needed to develop musical skills. Similarly, the famous composer Mozart asked his wife to read letters aloud during composing. He was convinced that the analytical part of his mind would be distracted by processing the speech and not disturb the creative part making music. At present, Bettie Edwards has developed a new method for teaching creative drawing, based on these insights into differences between the right and the left part of the brain [5].

Although a verbal–nonverbal dichotomy associated to the two sides of the brain is still considered speculative, the vast amount of research with split-brain patients and people with cerebral lesions reported in [3] shows convincingly that different areas in the brain are responsible for different perceptual processing. The pieces of the brain that handle language are not always involved in this processing. Some of the perceptual information is not mediated verbally, and therefore is hard to express in words. This observation implies that the usage of textual descriptions alone to search the database is too restricted. The user uses other valuation processes than the process modelled in the system.

3.2. Approximate retrieval

Another approach to the problem of multimedia search uses automatically derived properties called features [6]. The key to the retrieval process is *similarity* between objects. We search objects that are similar to the query instead of objects that are equal to the query. Therefore, we use the term approximate retrieval as opposed to exact retrieval. Because these features are calculated from the content of the objects, the approach is also known as content-based retrieval.

The features typically describe easy-to-calculate *syntactic* properties of the stored objects. We use the term syntactic to emphasize that these features are very low-level properties that mean little or nothing to the user in their bare form. But the user does not have to know what features the system uses for retrieval. Instead of explicitly dealing with these syntactic features, the user tells the system what kind of objects to search for by giving an example of a good object. We call this query paradigm "query by example".

The syntactic properties used in approximate retrieval hopefully capture some of the *semantic* characteristics of the multimedia object. The semantic properties are at the level of the user's perception. Unfortunately, we cannot automatically detect these properties of objects. We have to work with the syntactic properties that we can calculate.

The QBIC (*Query By Image Content*) system [7] first introduced this approach to accessing multimedia data in the domain of images. Features used for image retrieval include measures expressing the colour distribution of the image. Other features express the texture and the composition of the image. An image query is translated into a point query in some multi-dimensional feature space. The similarity between a query and a database object is estimated by using a distance function.

The most common example query to illustrate the approximate retrieval approach uses the picture of a sunset. With this query object, retrieval using colour features works very well. However, it is not trivial to find suitable features for the general situation and it is not always easy to judge why the system found that particular object similar. For example, Fig. 1 demonstrates the retrieved objects if one searches for pictures of red cars. We also retrieve images of buildings or waterfalls, which are semantically completely different. Syntactically though, the picture of a car can be very similar to the picture of a building, if we search in color space alone.

The approximate retrieval approach is not unique for image retrieval. In the Musclefish system, retrieval based on features is applied to the content-based retrieval of generic audio objects [8]. Measures based on pitch, energy and more advanced audio properties span the feature space. Since the early 1960s, a similar approach was applied to querying full-text retrieval systems in the field of information retrieval [9,10]. By using special-purpose speech recognizers, these text retrieval techniques may easily be extended to speech documents [11].

If the features have a clear perceptual interpretation, we may choose to let the user move directly through the feature space. The term navigational querying refers to that situation. Navigational querying has been demonstrated for musicians working with a database of musical instruments [12]. Essentially, it is just another way to use the approximate retrieval approach. In the QBIC system, users could directly manipulate the underlying color query. However, it is very hard to find features with a clear semantic interpretation for general multimedia objects. Therefore, the features are usually not exposed to the user.

3.3. Social information filtering

Shardanand and Maes introduced a slightly different method to help a user find multimedia objects [13]. The underlying idea of this *social information filtering* process is that several people have similar interests. We can collect the judgements of many people about objects in the database, for instance movies or compact discs, and use a nearest-neighbour algorithm to find judgement vectors that are similar. Next, the items that appear in a similar vector,

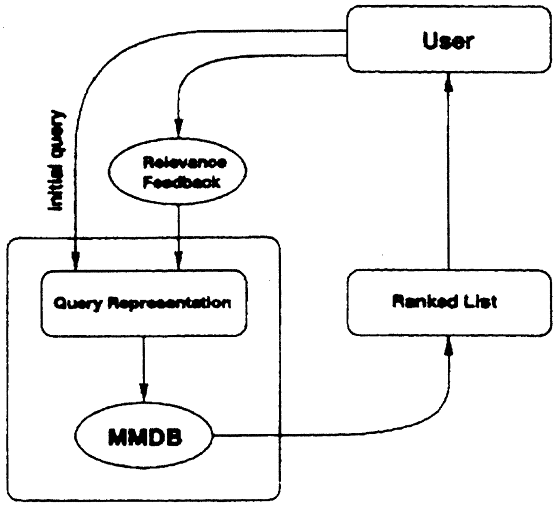

Fig. 2. The relevance feedback process.

but have not been judged by the user yet, can be advised to the user.

This technique has been commercialized in the firefly system [14]. Firefly can be used for advice on buying music, or to select a movie you would probably appreciate. Upon login, the system asks you to judge a selection of compact discs by several artists. This *profile* of your taste is then used to find people that like the same discs. If most ''similar'' people also judged another record highly, the system recommends it to you. The more people use firefly, the better the recommendations get.

Similarity between user judgements has three major benefits over similarity between the objects. First, it overcomes the problems with identifying suitable features for objects like music and art. Second, it has an inherent ability for serendipitous finds. You find objects that you like, but did not explicitly search for. Finally, the approach implicitly deals with qualitative aspects like style, which would be hardly possible with automatically derived features.

Technically, it should not be hard to integrate social information filtering with a multimedia database system. To perform approximate retrieval, we already process point queries in multi-dimensional spaces. The difference between both processes comes down to the difference between the space we map objects in, and the distance measure among these objects. However, this technique only works if the domain for which we collect judgements is rather narrow.

4. New requirements for multimedia databases

In this section, we show that accessing multimedia data puts new requirements on the database design. In the previous section, we discussed several approaches to handling multimedia objects as active objects. However, these techniques alone are not sufficient to provide multimedia retrieval. We first show that a multimedia database must support iterative search. Next, we discuss the need for a framework to combine the results from different search strategies. Finally, we conclude with the introduction of a promising new architecture suited for multimedia retrieval.

4.1. Interaction with a multimedia database

Interaction with a multimedia database faces us with a major problem that did not exist in the conventional database environment: we do not know how to formulate our multimedia query.

As we made clear in Section 3.1, a multimedia query cannot always be expressed verbally. Nonverbal aspects of multimedia, like emotional and aesthetic values, are hard to capture in words. These values may be more easily recognized and compared than described or expressed. The query by example paradigm certainly is a major improvement for some retrieval tasks. However, we cannot always come up with an example expressing our information need.

Although users cannot exactly express their information need with a query, they can judge retrieved objects for relevance. Thus, the solution for the problem of query formulation is to support an iterative search process, see Fig. 2. After an initial query has been processed, the user is asked to judge the retrieved objects. The relevance judgements are then used to adjust the query to better reflect the user's information need.

Querying multimedia needs a discourse and refinement phase for interaction between the user and the database. Relevance feedback has been used in text retrieval systems [10], but not in databases storing arbitrary objects. If we want to design multimedia databases, we need to change the database internals such that it can process relevance feedback.

4.2. Query processing using multiple representations

The techniques of Section 3 handle atomic new media objects, like *an* image or *a* sound fragment. However, the user is often interested in retrieving composite objects like newspaper articles or video documentaries. A video fragment can be represented by its subtitles, by the output from a speech recognizer, or by a sequence of keyframes [15]. It consists of several atomic objects. Also, for each atomic object, we can produce many different representations. For example, the keyframes can be represented with colour histograms, shape or texture features.

The usage of multiple representations of multimedia objects is crucial for a multimedia database system. Manually added descriptions are not sufficient for multimedia retrieval. Switching to approximate retrieval techniques overcomes some of the problems with textual descriptions, but introduces new problems because most features have only a syntactic value. As we saw in Fig. 1, retrieval with colour histograms retrieves also waterfalls and buildings instead of cars.

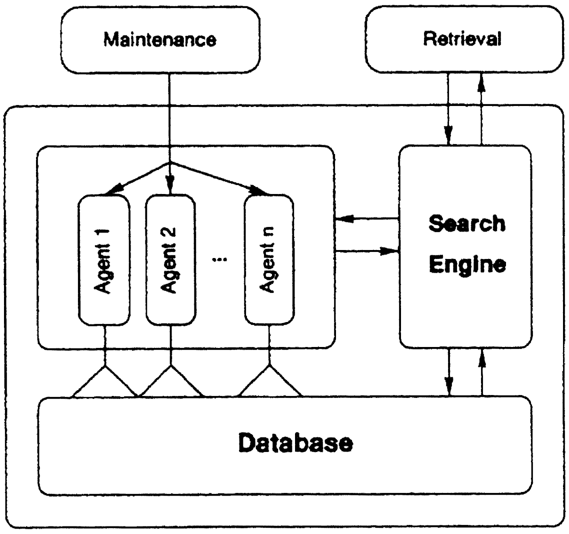

Fig. 3. Multimedia database architecture.

Rather than choosing one approach over the other, we should use the information from as many (imperfect) ways to describe objects as possible. Although a single representation is not sufficient, the combination of several representations may be. Recent experiments in image database research, as reported in [16], support the hypothesis that the combination of several feature representations improves the results of retrieval.

We cannot expect the user to search each representation separately and combine the results by hand. The user views information as a "gestalt", and each single representation is only a part of it [17]. Clearly, handling multiple representations in query processing is a task of the database management system.

4.3. A new database architecture for multimedia retrieval

In the previous sections, we discussed several issues with respect to the access to multimedia data in a database. Summarizing, we list three new requirements for multimedia databases:

- all objects are active objects;
- querying is an interaction process; and
- query processing uses multiple representations.

Although it is fairly easy to state these requirements, meeting them in an actual system is a tough problem. As an approach to design a system that can meet these requirements, we introduce the architecture of Fig. 3. We divide the design in a set of agents, a search engine and a conventional database to store the objects.

The set of agents takes care of the activeness of the objects. Each agent handles one representation of the multimedia objects. For example, one agent produces colour features of images. Another agent selects words from the title of text documents. Each agent knows how to find representations that are similar to the representation of a query object. It implements the similarity measure suitable for its domain. It also creates the necessary access structures in the database to speed up retrieval.

Agents may process manually added descriptions or features for approximate retrieval. How social information filtering can be used in our architecture is an open question. Theoretically, each agent can store a memory of user judgements for all instances. However, in practice this could be a rather costly solution.

The database system is used in two ways of operation: maintenance and retrieval. In maintenance mode, we can add or delete multimedia objects, or extent the database with new agents. After a change of the data collection, the agents update their internal representations.

The search engine bridges the database to the user in retrieval mode. It keeps track of the different agents that may participate in the retrieval process. The subtasks of the retrieval process are delegated to these agents. For each subtask, it asks an agent to provide objects that are similar to an example object.

The search engine contains knowledge about combining evidence from different representations. It also processes the relevance feedback from the user, and uses this feedback in the further iterations to refine the initial query. This part of the system takes care of the second and third requirement.

We need a unifying framework to describe the amount of evidence found for an object by each agent. In text retrieval systems, probability theory has been shown a good candidate for such a framework [10]. Probabilistic retrieval systems estimate the probability of usefulness to the user for each object. These probabilities can be combined according to probability theory, to realize combination of multiple representations, and processing of relevance feedback. INQUERY is a good text retrieval system demonstrating this functionality [18]. The MARS system, as described in [19], is the first image retrieval system to apply the probabilistic retrieval model in image databases. Our further research focuses on extending these results for the search engine in our multimedia database architecture.

5. Conclusions and further work

A fashion designer and a journalist work with high volumes of multimedia data, and they need a flexible storage and retrieval system to cope with their information collections and *especially* with those of their colleagues. In a closed environment, straightforward solutions with manually added descriptions may suffice. But, as soon as we work with a data collection for many users, about many topics, we need more powerful tools. In fact, everybody who collects and uses multimedia data is a candidate user of multimedia databases.

In this paper, we identified the properties that a true multimedia database system should have before we can effectively use computers to replace our bookshelves and shoeboxes. We gave three important requirements on multimedia databases. First, all objects should be handled as active objects. Next, retrieval in a multimedia database is necessarily an interactive process because the user cannot formulate his multimedia query. Finally, since no available technique to handle the objects as active objects is sufficient to provide access to the multimedia data on it's own, we have to combine the retrieval results for different representations.

These requirements can only be met if we extend the design of a conventional database system. We introduced an architecture that can provide access to multimedia data. Further research is necessary to investigate how the probabilistic text retrieval model can be applied to the retrieval of multimedia objects. Our current research investigates how to enhance this framework for multimedia retrieval. Although many aspects of multimedia databases have been studied, we still have a long road to take before multimedia database technology can realize its promises for human activity.

References

[1] E. Bertino, B. Catania, E. Ferrari, Query processing, in: Multimedia Databases in Perspective, Springer Verlag, 1997, pp. 181–217.

[2] G.A. Miller, P.N. Johnson-Laird. Language and Perception, Cambridge University Press, Cambridge, 1976.

[3] J.F. Iaccino. Left Brain–Right Brain Differences, Lawrence Erlbaum Associates, 1993.

[4] J.D. Barrow, The Artful Universe, Little, Brown and Company, 1995.

[5] B. Edwards, Drawing on the Right Side of the Brain, Harper Collins Publishers, London, 1993.

[6] C. Faloutsos, Searching Multimedia Databases by Content, Kluwer Academic Publishers, Boston/Dordrecht/London, 1996.

[7] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, The QBIC project: querying images by content using color, texture and shape, Technical Report RJ 9203, IBM Research Division, 1993.

[8] E. Wold, Th. Blum, D. Keisler, J. Wheaton, Content-based classification, search, and retrieval of audio, IEEE Multimedia, 3(3) (1996).

[9] G. Salton, Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Addison Wesley Publishing, 1989.

[10] C.J. van Rijsbergen. Information Retrieval, 2nd edn, Butterworths, London, 1979. Out of print, available online from http://www.dcs.glasgow.ac.uk/Keith/Preface.html.

[11] A.P. de Vries, Television information filtering through speech recognition, in: Interactive Distributed Multimedia Systems and Services, Springer, Berlin, 1996, pp. 59–69.

[12] B. Eaglestone, R. Vertegaal. Intuitive human interfaces for an audiodatabase, in: Proceedings of the Second International Workshop on Interfaces to Database Systems (IDS94), Lancaster University, 1994.

[13] U. Shardanand, P. Maes, Social information filtering: Algorithms for automating "word of mouth", in: ChI'95 Proceedings, Denver, CO, 1995.

[14] Firefly Networks Inc., Cambridge, MA 02142, USA. URL http://www.firefly.com/, 1997.

[15] H. Wactlar, T. Kanade, M. Smith, S. Stevens. Intelligent access to digital video: the information project, IEEE Computer, 29(5) (1996).

[16] T.P. Minka, R.W. Picard, Interactive learning using a "society of models", Technical Report TR-349, MIT Media Laboratory Perceptual Computing Section, Cambridge, MA, 1997. Submitted to special issue of Pattern Recognition on Image Databases: Classification and Retrieval.

[17] A. Gupta, R. Jain, Visual information retrieval Communications of the ACM 40 (5) (1997) 70–79.

[18] J.P. Callan, W.B. Croft, S.M. Harding, The INQUERY retrieval system, in: Proceedings of the 3rd International Conference on Database and Expert Systems Applications, 1992, pp. 78–83.

[19] M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, Th.S. Huang, Supporting similarity queries in MARS, in: Proceedings of ACM Multimedia 1997, Seattle, WA, November 1997.