The Relationship between IR and Multimedia Databases

Arjen P. de Vries
Centre for Telematics and Information Technology, University of Twente
Enschede, the Netherlands

Henk M. Blanken
Centre for Telematics and Information Technology, University of Twente
Enschede, the Netherlands

Abstract

Modern extensible database systems support multimedia data through ADTs. However, because of the problems with multimedia query formulation, this support is not sufficient. Multimedia querying requires an iterative search process involving many different representations of the objects in the database. The support that is needed is very similar to the processes in information retrieval. Based on this observation, we develop the miЯRor architecture for multimedia query processing. We design a layered framework based on information retrieval techniques, to provide a usable query interface to the multimedia database. First, we introduce a concept layer to enable reasoning over low-level concepts in the database. Second, we add an evidential reasoning layer as an intermediate between the user and the concept layer. Third, we add the functionality to process the users' relevance feedback. We then adapt the inference network model from text retrieval to an evidential reasoning model for multimedia query processing. We conclude with an outline for implementation of miЯRor on top of the Monet extensible database system.

1 Introduction

In the miЯRor project, we study multimedia query processing, and in particular its implications on database design. We assume a modern extensible database system, like Illustra [38], Starburst [20], or Monet [3]. By extending the database, advanced search techniques for multimedia objects can be incorporated in the database architecture.

Before we can query digitized multimedia data, we have to represent the data such that the database query processor can check if an object in the database matches a query. Obviously, matching the digitized data directly can only retrieve bit-for-bit identical objects, which is of little use in practice. Two different approaches exist to tackle the problem of access to multimedia objects.

The rather obvious approach is to represent multimedia data with manually added, textual descriptions. We query the database with exact match between query words and the words occurring in the textual description. For example, we can use subtitles as a representation for the video data in television archives to find a documentary about 'car' and 'Renault' [9]. A big problem is that textual descriptions cannot capture the full semantics of multimedia data [10]. It is not even sufficient when we have textual descriptions for all multimedia data in the database. Also, when subtitles are not readily available, a manual annotation process is very expensive for archiving large amounts of data.

The other approach to multimedia search is approximate retrieval. Querying is based on automatically derived properties called features [12], [49]. In image retrieval, common features are the color distribution, directionality, and circularity. Each feature space describes different (low-level) aspects of the images. Full-text information retrieval techniques can also be classified as approximate retrieval, in which the extracted features are based on words occurring in the text. This class of techniques is referred to as approximate retrieval, because we compare the objects using a measure of *similarity* instead of equality. To support approximate search in some feature space $\mathcal{F}$, we need one or more associated distance functions $\mathcal{D} : \mathcal{F} \times \mathcal{F} \rightarrow \mathbb{R}$ to calculate the distance between two points in $\mathcal{F}$. A feature extraction function and the distance function together specify an approximate retrieval method, see also [12], [1].

The query is usually not formulated on the features themselves, but derived from an example of the kind of object searched for. The query processor identifies objects that are similar to the query object with respect to the distance function, and typically ranks them in decreasing similarity for evaluation by the user. Approximate retrieval has been applied to retrieval of objects of various data types: images [13], speech documents [36], text documents [35], and fragments of audio [47].

From a user perspective, the main unsolved problem is how to use these different representations and techniques to fulfill an information need. A detailed analysis of querying multimedia data leads to new requirements on multimedia databases [10]: A multimedia query processor must support an iterative query process. Also, it should combine and select evidence from different sources. The query processor should decide what representations to use for answering the query, in interaction with the user.

In the following sections, we first describe how multimedia retrieval techniques are integrated in a modern extensible database system. We identify the problems with query formulation in such a database, to clarify the functionality that is lacking from current multimedia databases. We then present a new approach to query processing, drawing heavily on research in information retrieval, image databases, and machine learning, and we propose a database architecture to support this approach. Finally, we compare our work with other research to multimedia query processing and identify the challenges for database research that have to be solved before we can realize a multimedia database with useful query facilities.

2 Extensible databases and multimedia retrieval

In this section, we develop the major primitives for managing multimedia data like video fragments. After defining ‘multimedia object’ more precisely with respect to miЯRor, we explain the support of approximate retrieval in an extensible database system. Finally, we identify the problems for multimedia query processing that have not been solved in the extensible database approach.

2.1 What is a multimedia object?

A video fragment can be represented in several ways. One representation of the fragment is the set of subtitles. We can also represent the video by the output from a speech recognizer, or by a sequence of keyframes [46]. These keyframes can then be used for retrieval through feature representations based on color, texture or shape. The database system must provide the functionality to manage the video fragments and their representations.

A multimedia object like a video fragment can be described as a *structure* over a set of *atomic component objects*. In this paper, we ignore the structure of multimedia objects. The video fragment is simply modelled as a collection of atomic objects: a text object for the subtitles, an audio object for the audio track, and several image objects for the identified keyframes. The authors realize that multimedia search must handle composite objects as a whole rather than by its constituting components. However, we have to understand multimedia query processing on the level of the atomic component objects more deeply, before we can try to solve the extra problems introduced by composite objects.

The miЯRor data model for multimedia objects consists of three different entities:

- (atomic) component objects;
- digitized representation objects;
- feature space representations.

The (atomic) component objects enable us to abstract from specific details like the data format. We can store traditional attributes at this level. A multimedia object is associated with its *digitized representation object*. Because our main concern is improving query processing itself, we restrict ourselves to objects with a *single* digitized representation object. Note that we thus ignore the problem that a different digitized representation may result in different values in feature space for the same object. Each feature space representation of a component object represents an aspect of its content, such that it can be used in approximate search techniques. These features are automatically extracted from the digitized representation object.

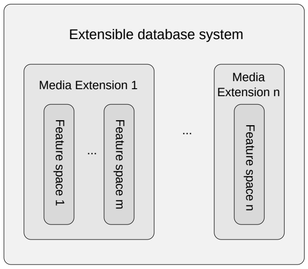

Figure 1: Multimedia objects in an extensible database system

From now on, we term both composite and atomic objects ‘multimedia object’. Admittingly, the term multimedia object is somewhat confusing, as atomic objects are really *single* media objects. However, conforming to the vocabulary common in database literature, we employ ‘multimedia object’ to distinguish these objects from more traditional alphanumerical objects.

2.2 Database support for multimedia retrieval

Modern database systems are extensible with *abstract data types* (ADTs). An ADT adds new base types and operations to the database system. In this section, we illustrate how to use extensibility to support the multimedia object model presented in the previous section. We adopt the Illustra terminology as used in [38]. In particular, we refer to an extension of base types and operations as a *datablade*.

We depict the general idea underlying multimedia support through ADTs in figure 1. We extend the database with *media extensions* $\mathcal{M}_1 \ldots \mathcal{M}_n$. A media extension supports a single medium, eg. image or text. It handles digitized representation objects, as well as the distance measures and feature extraction functions for the feature spaces we want to support. For instance, the image media extension might consist of an image datablade providing the base data types to add JPEG images to the type system, and maybe one or more basic retrieval methods like color histogram retrieval. An additional datablade can provide a search space specialized for face retrieval.

The feasibility of supporting approximate retrieval with extensible databases has been proven in the Chabot image database. A similar approach using Monet, a research prototype of a more advanced database system to achieve even better performance, is discussed in [26]. Chabot is implemented on top of the extensible Postgres database [27]. Methods defined in the image datablade are used to express conditions for approximate search. The distance function is defined as the user function `MeetsCriteria`. The following query defines a color histogram search over a table with images:

```
SELECT * FROM PhotoTable q
WHERE MeetsCriteria(‘SomeOrange’,q.histogram);
```

2.3 ADTs are not sufficient!

What most research in multimedia databases tends to overlook, see eg. [7], is that a database with these type extensions is *not* sufficiently usable for multimedia retrieval.

The following example illustrates the shortcomings of the ADT approach. Imagine a multimedia database of television programs. The database system has special datablades for searching the subtitles, and for searching several feature representations of the keyframes. We assume the user has to give a lecture on Dutch history, and searches a video fragment that satisfies the following information need $\mathcal{I}$:

($\mathcal{I}$:) People on a ship sailing the seas.

A multimedia database should support the retrieval of video fragments matching this information need $\mathcal{I}$. In the (unlikely) case that some movie star had the lines ‘People, here we stand, on this ship sailing the stormy seas’, the text-based representation derived from the subtitles is sufficient for finding a good video fragment. The *precision* of this strategy could be high: the fragments annotated with ‘people’, ‘ship’, and ‘sea’, are likely to satisfy the information need. However, *recall* is definitely going to be very low: in most of the relevant fragments, the subtitles are not likely to contain any of these words.

Approximate retrieval on the image features of the key frames may be more effective. However, a major problem with the approximate retrieval techniques is query formulation. It is hard to come up with a good query object. Querying the features directly is problematic as well. First, the range of colors that captures the image of a ship on sea retrieves also many pictures of other scenes. Second, most features lack a clear perceptual interpretation. Most users cannot imagine at all what kind of images are retrieved by a query like ‘circularity $\simeq$ 0.8’.

An additional problem is that a single representation is often not sufficient to decide whether an object is interesting or not. Color alone retrieves not just sea, but also rivers and waterfalls, and it is not uncommon that pictures of cars and buildings have ‘similar’ color histograms. The combination of several representations can capture more aspects of the content of an object. The combination of several feature representations might retrieve mostly pictures showing sea.

Current database systems do not provide functionality to capture the user’s information need in multiple representations. Conversely, the user views information as a ‘gestalt’, and each single representation is only a part of it [19]. We cannot expect users to search each representation separately - under the incorrect assumption that they know how to formulate a query to represent their information need - and combine the results by hand. Combination of representations is clearly a task for the database system.

3 An IR approach to multimedia retrieval

The problem of query formulation is best handled through an interactive search process. Although users cannot express their information need as conditions at the level of features, they *can* tell us which of the retrieved objects are relevant for their internal information need. Therefore, the user’s relevance judgements about the retrieved objects must be used to adjust the query to better reflect the user’s information need. This approach has been proven effective for both text retrieval and image retrieval (eg. [21], [50], and [37]).

In the miЯRor approach to multimedia retrieval, we access the video database to satisfy $\mathcal{I}$ in the following manner. We start with a simple keyword query to find some scenes annotated with ‘sea’ or ‘ship’. Alternatively, we may formulate a color histogram query directly to find mostly blue images. Next, we select those fragments from the retrieved objects that really contain sea, and those that really contain ships. Now, imagine that the system figures out that all relevant objects have a low score on circularity. It reformulates the initial query to retrieve more objects with low circularity.

In short, we provide a simple query at the start, that does not express our information need very precisely. The system helps us to formulate a better query in several iterations. The most promising feature representations are selected for further investigation using our relevance judgements from the previous iterations.

The type of query processing we just described is very similar to the approach taken in information retrieval (IR). In his influential paper [44], Van Rijsbergen gives the following definition of information retrieval:

> *The user expresses his information need in the form of a request for information. Information retrieval is concerned with retrieving those documents that are likely to be relevant to his information need as expressed by his request. It is likely that such a retrieval process will be iterated, since a request is only an imperfect expression of an information need, and the documents retrieved at one point may help in improving the request used in the next iteration.*

We like to emphasize the analogy with the miЯRor approach to database access sketched before. The task of an information retrieval system is to retrieve documents that are likely to be relevant to the user. If in this definition we replace ‘document’ with ‘multimedia object’, this is exactly what we expect from a multimedia database system. It is even harder to formulating a query for multimedia objects than for text documents. Because the query (or request) is not a perfect expression for the information need, we use an iterative process.

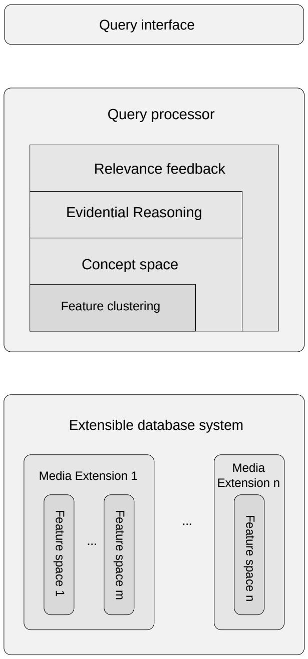

Figure 2: Multimedia database architecture

4 The miЯRor architecture

We conclude from section 3 that multimedia query processing is very similar to the processes in the definition of information retrieval. The only way to develop multimedia databases with sufficient query functionality, is to recognize the relationship between IR and multimedia databases, and to the design the multimedia database accordingly.

To solve the problems with multimedia querying, we introduce a high-level query processor in the database architecture, as shown in figure 2. Wong and Yao start their paper on probabilistic information retrieval with the following characterization [48]:

> *The three fundamental issues in information retrieval are the choice of an appropriate scheme to represent the documents, the query formulation and the construction of a suitable ranking function which determines the extent to which a document is relevant to a query.*

These three fundamental issues in IR are reflected in the three layers in our design. At the bottom, we find the concept layer, to manage the basic concepts to represent the content of the media objects. The user's information need $\mathcal{I}$ is expressed in a request consisting of these concepts, example objects, and relevance judgements about previously retrieved objects. The middle layer implements the ranking function. Query formulation is reflected in the module for relevance feedback.

In the following subsections, we discuss the motivation, the design considerations, and an outline for implementation for each of these layers of the multimedia query processor. Before proceeding, we like to emphasize that our main research goal in this project is not to participate in the quest after the best retrieval model. Our focus is on the development of a usable query interface for a multimedia database system. As a first attempt to such an interface, we combine results from information retrieval with the approximate retrieval techniques common in multimedia systems. In short, our contribution is a generalization of the ideas from text retrieval to address the problem of query formulation in multimedia databases.

4.1 Concept layer

We have to obtain a representation of each document and query, that is suitable for a computer to use [43]. The concept layer covers the scheme to represent the multimedia objects in the retrieval process.

In information retrieval literature, the units in the document representative are usually referred to as the *indexing terms*. Most systems use the words occuring in a document as indexing terms. Sometimes, the words are first stemmed to their roots, and words that occur on a stoplist are removed. Researchers have studied many ways to reduce the effects of ambiguity in natural language on the document representatives. People experimented with term clustering, thesauri and augmenting the indexing vocabulary with phrases. Unfortunately, the straightforward representation of documents by their words often performs as well in retrieval experiments, evaluated on recall and precision, and sometimes even better.

The usage of words occurring in a document as basic units of content probably performs so (amazingly) well, because words naturally refer to *concepts* in the real world, and not to specific instances. The main problem of using the feature descriptions from approximate retrieval techniques as basic units of content, is that these feature descriptions are quite different from words in natural language. The feature description of an object, a point in multidimensional space, is usually unique. It only identifies that particular *instance*, and not a concept.

Hence, before we can use feature descriptions as indexing terms, we group several instances together. We *cluster* points in feature space into some low-level concept space $\mathcal{C}^*$. Essentially, the cluster algorithm can be described as a mapping $\Gamma_C^i : \mathcal{F}^i \rightarrow \mathcal{C}^*$. We then use these concepts as the indexing terms, similar to the words in text retrieval.

In [14], the author shortly mentions a *supervised* clustering strategy. He suggests to train the system with example objects for so-called ‘content features’, examples of which are ‘type of light’ (artificial or natural), and ‘source’ (photograph versus painting). Apart from the costs associated with manual training, we think it is going to be problematic to decide which ‘content features’ to define, especially in an environment with many different users and varying information needs. This results in the same problems we identified for textual descriptions [10].

Instead of the top-down definition of concepts mentioned before, we can apply *unsupervised* techniques to select clusters in feature space. The advantage of this bottom-up approach is that we let the system identify an appropriate set of concepts. These concepts are not exposed to the user for query formulation. The system uses the concepts only internally, and detects them through the dialogue with the user, from the example objects and from relevance judgements. This unsupervised approach to conceptualizing a feature space has been succesfully applied in the FourEyes learning agent for the Photobook image database [25].

4.2 Evidential reasoning

Recalling the definition of information retrieval, the core of a retrieval system is the software that estimates which objects are likely to be relevant to the user’s information need as expressed by his request. Wong and Yao called this the ranking function.

Estimating the relevance of a candidate object is based on the evidence found in the object representatives. The estimation process draws conclusions from available evidence. We infer a relationship between objects and queries. Thus, the matching process in *any* retrieval system is a theory of *evidential reasoning*, formulated either explicitely or implicitely.

Because we are not certain about both the representation of the user’s information need, and the representation of the information found in the objects, (multimedia) retrieval is best described as *plausible inference under uncertainty*. In traditional information retrieval, this evidence is the presence or absence of terms in the text of a document. Analogously, in our model, the evidence consists of the presence or absence of the concepts defined in the concept layer.

The classical method for evidential reasoning is based on probability theory. Other approaches include Dempster-Shafer theory, fuzzy logic, and fuzzy set theory [29]. Like Turtle did for text retrieval [41], we adopt the paradigm of Bayesian belief networks for evidential reasoning in miЯRor. The details of the application of Bayesian inference networks for retrieval in multimedia databases are taken up in section 5.

The layered structure of our design opens the possibility to experiment with other inference procedures. For example, the aforementioned Foureyes learning agent reasons using higher-level groupings over the concept space [25]. These groupings are generated from relevance feedback using an algorithm based on the greedy AQ algorithm from machine learning. Although quite different from traditional approaches in information retrieval, this approach can be implemented as an alternative.

4.3 Relevance feedback

As discussed before, we believe that the key to successful multimedia query processing is to support a *dialogue* between the user and the database system [10]. The main goal in this dialogue is to infer the low-level concepts that best describe the user's information need. Another purpose of the analysis of relevance feedback is to learn dependencies between feature spaces, and between concepts.

The user should be encouraged to start a new dialogue for each information need. This way, we can be more certain that the sets of queries and relevance judgements that we collect are related to a specific information need. The query interface needs primitives to start and end a dialogue, similar to the way we start and end transactions in traditional database applications.

We distinguish two approaches to process the user's relevance feedback. In *query-space modification*, we use these relevance judgments to change the relative importance of terms in the query. Also, we can extend the queries with new terms that occur often in the positive examples, but rarely in the negative examples [32]. In *document-space modification*, we collect a reasonably sized set of queries for which a document was found relevant or irrelevant [15]. We use this set to adapt the document representatives, such that they better reflect the true semantics of the document. Indexing terms may be added to or removed from the document representative.

Both types of relevance feedback are important in miЯRor. During interaction, we can only perform query-space modification, because changing the object representations cannot be done real-time and the user is not willing to wait a long time between each step in the dialogue. The examples of good and bad objects are analyzed to identify the representations that are most likely to be informative. These representations are then used to produce a new ranking of multimedia objects.

An analogy of document-space modification is necessary to update the clustering of feature space in concepts. It is unlikely that the concepts found by an unsupervised clustering algorithm are very similar to human perception of concepts. While the interactive adaption of the weighting scheme is the only way to process relevance feedback in an interactive mode, a set of queries and relevance judgements can be used offline to improve the mapping of feature space to concepts. Hopefully, the longer the system operates, the better the concepts will reflect human perception.

5 Bayesian networks and multimedia retrieval

In this section, we further discuss our preference for Bayesian networks in the evidential reasoning layer. We review the benefits of the network retrieval model, and explain how inference is used during retrieval. We conclude the section with the proposed extensions for multimedia retrieval.

5.1 Why Bayesian networks?

A Bayesian network is an efficient approach to probabilistic reasoning [30]. Although Bayesian inference in arbitrary networks is known to be NP-hard [8], probabilistic reasoning is tractable when we apply some restrictions on the network topology.

Turtle and Croft modelled information retrieval using a restricted class of Bayesian inference networks [40]. They showed how to express the boolean, probabilistic, and vector space retrieval models as inference in Bayesian networks [42]. INQUERY, the text retrieval system based on this inference network model, has proven that the theory works well in practice at several TREC and Tipster conferences.

In his thesis [41], Turtle argues:

> *Given the availability of a number of representation techniques that capture some of the meaning of a document or information need, our basic premise is that decisions about which documents match an information need should make use of as many of the representations as practical.*

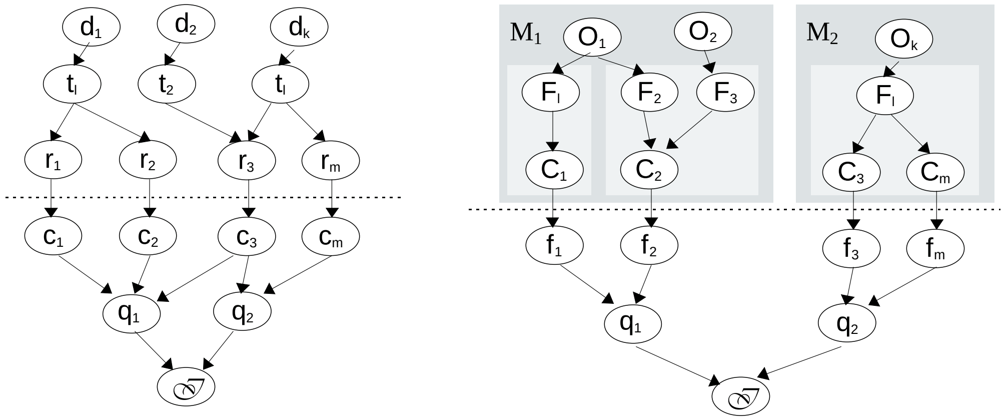

Figure 3: The Turtle and Croft model next to our adapted network model

The main motivation to introduce the formalism of Bayesian networks in information retrieval, was the lack of a consistent approach to handling multiple schemes to represent the documents. But, research had consistently shown that different representation schemes often retrieve different relevant documents. Also, documents retrieved by several methods are usually highly relevant. Bayesian networks provide the theoretic basis to use multiple representations for information retrieval. Fung and Del Favero supplied to this argument that Bayesian networks provide an intuitive representation of dependencies between representations [17]. They also explain how to model dependencies between concepts, while maintaining a tractable inference procedure.

A retrieval model based on inference networks seems an appropriate foundation for evidential reasoning in the miЯRor architecture. Evidential reasoning with multiple representations is exactly what we need for multimedia query processing. The possibility to model dependencies easily without the need to derive a closed form expression is attractive because several feature spaces may capture overlapping perceptual concepts. In an extendible database system, we cannot know in advance what feature spaces are available at runtime. With techniques for learning Bayesian networks, eg. [22] and [5], dependencies between feature spaces can be detected automatically.

5.2 Retrieval with Bayesian networks

We first review the network structure introduced for text retrieval, and then explain some minor changes that are useful in a multimedia retrieval model. The model introduced by Turtle and Croft is shown on the left part of figure 3. The network consists of two component networks: the document network and the query network. All nodes represent binary variables.

The document network is built once for a given collection of documents, and its structure does not change during query processing. It consists of a document node d_i for each document, text nodes t_j representing the document text, and representation concepts r_k for terms occuring in the document. The text nodes are not really necessary, but allow sharing pieces of text among documents. The concept representation nodes can be divided into several subsets, each corresponding to a single representation scheme.

A query network is constructed dynamically for each information need. It is modified in an interactive setting by changing existing queries, or adding new queries. The query network consists of the information need $\mathcal{I}$ at the bottom, expressed in a number of queries or requests q_i, and the query concepts c_j occuring in these requests.

The document network and the query network are joined together during query processing through links between representation concepts and query concepts. To rank the documents with respect to an information need, we use probabilistic inference in the network to calculate the probability $\mathcal{P}(\ \mathcal{I} \mid d_i = \mathrm{true}\ )$ for each document d_i.[1] Usually, the

[1] We really calculate the probability $\mathcal{P}(\ \mathcal{I} \mid d_i = \mathrm{true} \wedge d_j = \mathrm{false},\ \forall j \neq i\ )$. For the sake of brevity, we do not explicitely mention the values of the d_j that are false.

query concepts are mapped to the representation concepts with a single link. However, more complex connections are useful to represent thesaurus relationships, or phrase dependencies. Also, the processing of relevance feedback may add extra links between the query concepts and representation concepts.

It is easy to read the independence assumptions from the network. First, it is assumed that the presence or absence of one representation concept in a document does not influence the belief in the presence or absence of other concepts in this document. This assumption, better known as the Binary Independence assumption, is obviously violated for terms that are identical or closely related. But, as the assumption of independence is necessary to keep inference tractable, it is widely used in information retrieval. Another independence assumption in the network that is usually not true in practice, is that relevance of one document does not influence the relevance of another document. Recall and precision experiments showed that reducing the effects of these assumptions by adding links between thesaurus terms, or explicitely representing nearest neighbour clusters of documents, can indeed improve retrieval results.

5.3 Adapting the model for multimedia retrieval

On the right hand side of figure 3, we show the proposed network structure for multimedia information retrieval. We do not make many changes to the original network model. Instead of document nodes, we define multimedia object nodes for the objects stored. We will refer to the top component of the network as the object network. Concept nodes C_j represent the concepts from $\mathcal{C}^*$ that have been identified in the concept layer.

In section 2.1, we defined multimedia objects as flat objects. This decision is reflected in the network model presented so far, as the object network is not really one directed acyclic graph, but a collection of individually disconnected subgraphs. Each of these subgraphs handles the retrieval process in one media extension $\mathcal{M}_i$.

The gray boxes in the figure correspond to media extensions, and the light gray boxes to different feature spaces in these extensions. We could have put all concepts direct under the object nodes, similar to the way multiple document representations are used in the original network model. Because we want to make all our independency assumptions explicit, we choose to introduce an additonal feature space node F_k above the concepts identified in feature space $\mathcal{F}^k$. The introduction of these feature space nodes has three benefits. First, we can change the network structure to better reflect dependencies between the representations using relevance judgments. Also, this design makes it easier to restrict the representations that we use for answering the query. Finally, when we extend our theoretical formalism to include utility theory, we can also model decisions based on the costs associated to accessing specific representations.

The conditional probability $\mathcal{P}(C_j \mid F_i)$ is the probability that concept C_j is represented in that feature representation of the multimedia object. For text retrieval in INQUERY, these probabilities are approximated by traditional $tf \cdot idf$ measures. In a general feature space, this probability should be estimated using the relative position of that point in the cluster and the distribution of feature points in that cluster. We investigate the application of the cluster-based probability model from [31] as an estimate. An unsupervised classification algorithm may also provide the probability of class membership, eg. AutoClass [6].

We can take a first step towards a retrieval model for compound objects by adding a graph on top of the network described so far, that models the structure of the composite object. We can add a video node v_l per fragment, and connect it to the multimedia objects associated to v_l. We should measure experimentally if the representations among different media extensions $\mathcal{M}_i$ are really independent. Additional links between not so independent representations, like the output of a speech recognizer and the text in the closed captions, may improve retrieval results in this model for compound objects. However, this even harder problem of retrieving compound objects should only be addressed after we established experimental evaluation of the network shown in figure 3.

6 Comparison to other work

We are not aware of research efforts in the database community to address the problems identified in section 2.3. Several database groups study deductive databases to model complex queries involving temporal and structural relationships, see eg. [39]. This line of research is subsidiary to our work. Although the expressiveness of query languages in such database systems makes it easier to specify constraints involving structure of the data, it does not reduce the query formulation problems involved with the specification of constraints on content.

The close relationship with information retrieval research has already been addressed in detail. Another field that relates to our project is content-based image retrieval research. Recently, the MARS project introduced techniques from information retrieval in content-based image retrieval [28]. They also use relevance feedback techniques to improve the initial retrieval results [34], [33]. We already mentioned the relationship of our work with the image retrieval research in the MIT Media Lab. Their results with advanced machine learning techniques to using different feature representations for retrieval demonstrate the feasibility of our design [25]. While these groups' research concentrates on improving the performance of ranking and retrieval algorithms, we are mainly interested in the usage and integration of these techniques in extensible database systems.

Integration of IR and databases has been a research topic for several years, but from a different viewpoint. The main interest in databases from IR researchers has been the more traditional database support: concurrent addition, update, and retrieval of documents, as well as scalability of the system, see eg. [24], [18], and [11]. We start from an extensible database system for multimedia retrieval, handling many different representations of objects that are hard to manipulate for the user. The required database functionality is very similar to information retrieval, so we adopt techniques from IR to assist the users with query formulation. We will implement these techniques in a database system similar to the implementation of text retrieval in a database by Vasanthakumar et al. [45].

The desire to model uncertainty in database systems inspired the recent development of probabilistic relational algebras, eg. [16] and [23]. This development will certainly lead to a better framework to implement the evidential reasoning layer. However, these algebras require changes in the core of extended relational database systems. Furthermore, query optimization and physical database design are still research issues for these models.

In miЯRor, we do not really need a generic framework for probabilistic reasoning under uncertainty over the data in the database. The evidential reasoning layer is not part of the interface to the user, but a backbone to support query formulation. Hence, we develop a reasoning framework satisfying our specific requirements. When efficient probabilistic and extensible relational databases become available, we can use probabilistic algebra to implement miЯRor's inference procedures.

7 Further work

The next step in the miЯRor project is the implementation of the retrieval model in an extensible database system. We decided to use the Monet database system [3]. Monet is a database kernel developed to experiment with implementation techniques for novel application domains on parallel and distributed processing platforms. The design of Monet is suited for traditionally hard and query-oriented database applications: its power has been demonstrated for geographic information systems [4], for image retrieval [26], and also for data mining. Ongoing research develops an object algebra [2]. Such an algebra is especially useful when we integrate the structure of multimedia objects in the retrieval process.

In our initial prototype implementation, we plan to use AutoClass as an external tool to cluster the feature space in concepts. AutoClass is a Bayesian unsupervised clustering tool [6]. Development of an efficient clustering algorithm that can incrementally maintain the clusters under updates will be addressed in the future. Similar to [45], we are extending the database with special operators for Bayesian inference. In Monet, these operators are integrated deeper in the database kernel, thanks to its extensible data model. The relevance feedback layer generates the query sequences to be executed by the database.

We need experimental evaluation before we can decide on some aspects of the model. One of these research questions is whether we may restrict ourselves to clusters within each feature space, or if we should consider concepts spanning several feature spaces as well. Evaluation can also point out if learning dependencies between media and dependencies between feature spaces improves the query performance.

A major problem for experimental evaluation is the lack of standard test sets as available for text retrieval. Although we have some ideas for experiment design, evaluation of multimedia retrieval performance is still an open problem.

8 Conclusions

Multimedia query processing involves approximate search techniques. Modern extensible database systems can support these techniques through ADTs. However, multimedia querying requires a higher level of query processing than is provided by these extensible database systems. Formulating a query for an information need is not an easy task, and has to be supported by the system in an interactive mode.

The kind of query facilities we expect from a multimedia database system are very similar to the processes in information retrieval. This observation urges the integration of IR techniques and databases. We have designed a three-layered architecture for the query processor in a multimedia database system, and explained in detail what the tasks of each layer are.

We have introduced an adapted form of the inference network model from text retrieval as the evidential reasoning layer in miЯRor. We compared our approach with other research in multimedia databases, to demonstrate that we address new problems that have not been properly recognized before. Finally, we outlined our proposal to implement the architecture, and identified the design decisions that can only be made after experimental evaluation.

We will now develop a prototype implementation on the Monet database system. We also have to design an experimental evaluation strategy to test the performance of multimedia query processing in miЯRor.

References

[1] J.R. Bach. The Virage image search engine: An open framework for image management. In *SPIE Vol. 2670 Storage and Retrieval for Still Image and Video Databases IV*, pages 76–87, 1996.

[2] P. Boncz, A.N. Wilschut, and M.L. Kersten. Flattening an object algebra to provide performance. In *International Conference on Data Engineering*, 1998. To appear.

[3] P.A. Boncz and M.L. Kersten. Monet: An impressionist sketch of an advanced database system. In *BIWIT'95: Basque international workshop on information technology*, July 1995.

[4] P.A. Boncz, C.W. Quak, and M.L. Kersten. Monet and its geographic extensions. In *Proceedings of the 1996 EDBT conference*, 1996.

[5] W. Buntine. A guide to the literature on learning probabilistic networks from data. *IEEE Transactions on knowledge and data engineering*, 8(2):195–210, April 1996.

[6] P. Cheeseman and J. Stutz. Bayesian classification (AutoClass): Theory and results. In *Advances in Knowledge Discovery and Data Mining*. AAAI Press, 1995.

[7] M. Colton. Illustra: The multi-media DBMS. Technical report, Illustra Information Technologies, Inc., 1994.

[8] G.F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42:393–405, 1990.

[9] A.P. de Vries. Radio and television information filtering through speech recognition. In *Interactive Distributed Multimedia Systems and Services*, pages 59–69, Berlin, Germany, March 1996. Springer Verlag. Also available as CTIT Technical Report 95-20.

[10] A.P. de Vries, G.C. van der Veer, and H.M. Blanken. Let's talk about it: Dialogues with multimedia databases. Technical Report CTIT 97-13, Centre for Telematics and Information Technology, 1997. Full paper presentation at Multimedia Minded 1997; accepted for journal publication by Displays.

[11] S. DeFazio, A. Daoud, L.A. Smith, and J. Srinivasan. Integrating IR and RDBMS using cooperative indexing. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'95)*, pages 84–92, 1995.

[12] C. Faloutsos. *Searching multimedia databases by content.* Kluwer Academic Publishers, Boston/Dordrecht/London, 1996.

[13] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3:231–262, 1994.

[14] R. Ferber. Accessing documents by knowledge discovery methods and intelligent retrieval. In *Delos Working Group Reports, ERCIM-97-W001*, pages 17–22, 1996. Available online from http://www.darmstadt.gmd.de/˜ferber/delos/.

[15] N. Fuhr and C. Buckley. A probabilistic learning approach for document indexing. *ACM Transactions on Office Information Systems*, 9(3):223–248, July 1991.

[16] N. Fuhr and Th. Rölleke. A probabilistic relational algebra for the integration of information retrieval and database systems. *ACM Transactions on Information Systems*, 15(1):32–66, January 1997.

[17] R.M. Fung and B.A. Del Favero. Applying Bayesian networks to information retrieval. *Communications of the ACM*, 38(3):43–48, March 1995.

[18] J. Gu, U. Thiel, and J. Zhao. Efficient retrieval of complex objects: Query processing in a hybrid DB and IR system. In *Proceedings of the 1st German National Conference on Information Retrieval*, 1993.

[19] A. Gupta and R. Jain. Visual information retrieval. *Communications of the ACM*, 40(5):70–79, May 1997.

[20] L.M. Haas, W. Chang, G.M. Lohman, J. McPherson, P.F. Wilms, G. Lapis, B. Lindsay, H. Pirahesh, M. Carey, and E. Shekita. Starburst mid-flight: As the dust clears. *IEEE Trans. on Knowledge and Data Engineering*, 2(1):143–160, March 1990.

[21] D. Haines and W.B. Croft. Relevance feedback and inference networks. In *Proceedings of the sixteenth annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'93)*, pages 2–11, 1993.

[22] D. Heckerman. A tutorial on learning with Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, Advanced technology division, March 1995. Revised edition November 1996.

[23] L.V.S. Lakshmanan, N. Leone, R. Ross, and V.S. Subrahmanian. ProbView: A flexible probabilistic database system. *ACM Transactions on Database Systems*, 22(3):419–469, September 1997.

[24] I.A. Macleod. Text retrieval and the relational model. *Journal of the American society for information science*, 42(3):155–165, 1991.

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

[26] N.J. Nes, C. van den Berg, and M.L. Kersten. The Acoi algebra: A query algebra for image retrieval systems. Submitted to EDBT'98.

[27] V.E. Ogle and M. Stonebraker. Chabot: retrieval from a relational database of images. *IEEE Computer*, 28(9):40–48, September 1995.

[28] M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, and Th.S. Huang. Supporting similarity queries in MARS. In *Proceedings of ACM Multimedia 1997*, Seattle, Washington, November 1997.

[29] S. Parsons. Current approaches to handling imperfect information in data and knowledge bases. *IEEE Transactions on knowledge and data engineering*, 8(3):353–372, June 1996.

[30] J. Pearl. *Probabilistic reasoning in intelligent systems: Networks of Plausible Inference.* Morgan Kaufmann, California, 1988.

[31] K. Popat and R.W. Picard. Cluster-based probability model and its application to image and texture processing. *IEEE Transactions on Image Processing*, 6(2):268–284, February 1997.

[32] S.E. Robertson. On term selection for query expansion. *Journal of documentation*, 46(4):359–364, 1990.

[33] Y. Rui, Th.S. Huang, and S. Mehrotra. Relevance feedback techniques in interactive content-based image retrieval. In *Proceedings of IS&T and SPIE Storage and Retrieval of Image and Video Databases VI*, San Jose, CA, January 1998.

[34] Y. Rui, Th.S. Huang, S. Mehrotra, and M. Ortega. Automatic matching tool selection via relevance feedback in MARS. In *Proc. of The 2nd Int. Conf. on Visual Information Systems*, San Diego, California, December 1997.

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

[36] P. Schäuble. *Multimedia information retrieval. Content-based information retrieval from large text and audio databases*. Kluwer Academic Publishers, 1997.

[37] S. Sclaroff, L. Taycher, and M. La Cascia. Imagerover: A content-based image browser for the world wide web. In *Proceedings IEEE Workshop on Content-based Access of Image and Video Libraries*, San Juan, Puerto Rico, June 1997.

[38] M. Stonebraker and Dorothy Moore. *Object-relational DBMSs: The next great wave*. Morgan Kaufmann Publishers, Inc., 1996.

[39] V.S. Subrahmanian and S. Jajodia. *Multimedia database systems. Issues and research directions.* Springer Verlag, 1996.

[40] H. Turtle and W.B. Croft. Evaluation of an inference network-based retrieval model. *ACM Transactions of information systems*, 9(3), 1991.

[41] H.R. Turtle. *Inference networks for document retrieval.* PhD thesis, Univeristy of Massachusetts, 1991.

[42] H.R. Turtle and W.B. Croft. A comparison of text retrieval models. *The computer journal*, 35(3):279–290, 1992.

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

[44] C.J. van Rijsbergen. A non-classical logic for information retrieval. *The computer journal*, 29(6):481–485, 1986.

[45] S.R. Vasanthakumar, J.P. Callan, and W.B. Croft. Integrating INQUERY with an RDBMS to support text retrieval. *Bulletin of the technical committee on data engineering*, 19(1):24–34, March 1996.

[46] H. Wactlar, T. Kanade, M. Smith, and S. Stevens. Intelligent access to digital video: The Informedia project. *IEEE Computer*, 29(5), May 1996.

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

[48] S.K.M. Wong and Y.Y. Yao. On modeling information retrieval with probabilistic inference. *ACM Transactions on Information Systems*, 13(1):38–68, January 1995.

[49] J.K. Wu, A. Desai Narasimhalu, B.M. Mehtre, C.P. Lam, and Y.J. Gao. CORE: a content-based retrieval engine for multimedia information systems. *Multimedia Systems*, 3:25–41, 1995.

[50] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th International Conference on Research and Development in Information Retrieval (SIGIR ’96)*, pages 4–11, Zürich, Switzerland, 1996.