4

information retrieval

Searching for information on the web is cumbersome. Given our experiences today, we may not even want to think about searching for multimedia information on the (multimedia) web. Nevertheless, in this chapter we will briefly sketch one of the possible scenarios indicating the need for multimedia search. In fact, once we have the ability to search for multimedia information, many scenarios could be thought of. As a start, we will look at two media types, images and documents. We will study search for images, because it teaches us important lessons about content analysis of media objects and what we may consider as *being similar*. Perhaps surprisingly, we will study text documents because, due to our familiarity with this media type, text documents allow us to determine what we may understand by effective search.

4.1 scenarios

Multimedia is not only for entertainment. Many human activities, for example medical diagnosis or scientific research, make use of multimedia information. To get an idea about what is involved in multimedia information retrieval look at the following scenario, adapted from [MMDBMS],

Amsterdam Drugport

> *Amsterdam is an international centre of traffic and trade. It is renowned for its culture and liberal attitude, and attracts tourists from various ages, including young tourists that are attracted by the availability of soft drugs. Soft drugs may be obtained at so-called coffeeshops, and the possession of limited amounts of soft drugs is being tolerated by the authories.*
>
> *The European Community, however, has expressed their concern that Amsterdam is the centre of an international criminal drug operation. Combining national and international police units, a team is formed to start an exhaustive investigation, under the code name* Amsterdam Drugport.

Now, without bothering ourselves with all the logistics of such an operation, we may establish what sorts of information will be gathered during the investigation, and what support for (multimedia) storage and (multimedia) information retrieval must be available.

Information can come from a variety of sources. Some types of information may be gathered continuously, for example by video cameras monitoring parking lots, or banks. Some information is already available, for example photographs in a (legacy database) police archive. Also of relevance may be information about financial transactions, as stored in the database of a bank, or geographic information, to get insight in possible dug traffic routes.

From a perspective of information storage our informatio (data) include the following media types: images, from photos; video, from surveillance; audio, from interviews and phone tracks; documents,from forensic research and reports; handwriting, from notes and sketches; and structured data, from for example bank transactions.

We have to find a way to store all these data by developing a suitable multimedia information system architecture, as discussed in chapter 6. More importantly, however, we must provide access to the data (or the information space, if you will) so that the actual police investigation is effectively supported. So, what kind of queries can we expect? For example, to find out more about a murder which seems to be related to the drugs operation.

retrieval

- *image query* – all images with this person
- *audio query* – identity of speaker
- *text query* – all transactions with BANK Inc.
- *video query* – all segments with victim
- *complex queries* – convicted murderers with BANK transactions
- *heterogeneous queries* – photograph + murderer + transaction
- *complex heterogeneous queries* – *in contact with* + murderer + transaction

Apparently, we might have simple queries on each of the media types, for example to detect the identity of a voice on a telephone wiretap. But we may also have more complex queries, establishing for example the likelihood that a murderer known by the police is involved, or even *heterogeneous* queries (as they are called in [MMDBMS]), that establish a relation between information coming from multiple information sources. An example of the latter could be, *did the person on this photo have any transactions with that bank in the last three months*, or nore complex, *give me all the persons that have been in contact with the victim (as recorded on audio phonetaps, photographs, and video surveillance tapes) that have had transactions with that particular bank.*

I believe you'll have the picture by now. So what we are about to do is to investigate how querying on this variety of media types, that is images, text, audio and video, might be realized.

research directions– *information retrieval models*

Information retrieval research has quite a long history, with a focus on indexing text and developing efficient search algorithms. Nowadays, partly due to the wide-spread use of the web, research in information retrieval includes modeling, classification and clustering, system architectures, user interfaces, information visualisation, filtering, descriptive languages, etcetera. See [IR].

Information retrieval, according to [IR], deals with the representation, storage, organisation of, and access to information items. To see what is involved, imagine that we have a (user) query like:

> *find me the pages containg information on ...*

Then the goal of the information retrieval system is to retrieve information that is useful or relevant to the user, in other words: *information that satisfies the user's information need.*

Given an information repository, which may consist of web pages but also multimedia objects, the information retrieval system must extract syntactic and semantic information from these (information) items and use this to match the user's information need.

Effective information retrieval is determined by, on the one hand, the *user task* and, on the other hand, the *logical view* of the documents or media objects that constitute the information repository. As user tasks, we may distinguish between *retrieval* (by query) and *browsing* (by navigation). To obtain the relevant information in retrieval we generally apply *filtering*, which may also be regarded as a ranking based on the attributes considered most relevant.

The logical view of text documents generally amounts to a set of index terms characterizing theb document. To find relevant index terms, we may apply operations to the document, such as the elimination of stop words or text stemming. As you may easily see, full text provides the most complete logical view, whereas a small set of categories provides the most concise logical view. Generally, the user task will determine whether semantic richness or efficiency of search will be considered as more important when deciding on the obvious tradeoffs involved.

information retrieval models In [IR], a great variety of information retrieval models is described. For your understanding, an information retrieval model makes explicit how index terms are represented and how the index terms characterizing an information item are matched with a query.

When we limit ourselves to the classic models for search and filtering, we may distinguish between:

information retrieval models

- boolean or set-theoretic models
- vector or algebraic models
- probabilistic models

Boolean models typically allow for *yes/no* answers only. The have a set-theoretic basis, and include models based on fuzzy logic, which allow for somewhat more refined answers.

Vector models use algebraic operations on vectors of attribute terms to determine possible matches. The attributes that make up a vector must in principle be orthogonal. Attributes may be given a weight, or even be ignored. Much research has been done on how to find an optimal selection of attributes for a given information repository.

Probabilistic models include general inference networks, and belief networks based on Bayesan logic.

Although it is somewhat premature to compare these models with respect to their effectiveness in actual information etrieval tasks, there is, according to [IR], a general consensus that vector models will outperform the probabilistic models on general collections of text documents. How they will perform for arbitrary collections of multimedia objects might be an altogether different question!

Nevertheless, in the sections to follow we will focus primarily on generalized vector representations of multimedia objects. So, let's conclude with listing the advantages of vector models.

vector models

- attribute term weighting scheme improves performance
- partial matching strategy allows retrieval of approximate material
- metric distance allows for sorting according to degree of similarity

Reading the following sections, you will come to understand how to adopt an attribute weighting scheme, how to apply partial matching and how to define a suitable distance metric.

So, let me finish with posing a research issue: *How can you improve a particular information retrieval model or matching scheme by using a suitable method of knowledge representation and reasoning?* To give you a point of departure, look at the logic-based multimedia information retrieval system proposed in [Dolores].

4.2 images

An image may tell you more than 1000 words. Well, whether images are indeed a more powerful medium of expression is an issue I'd rather leave aside. The problem how to get information out of an image, or more generally how to query image databases is, in the context of our *Amsterdam Drugport* operation more relevant. There are two issues here

- obtaining descriptive information
- establishing similarity

These issues are quite distinct, although descriptive information may be used to establish similarity.

descriptive information

When we want to find, for example, all images that contain a person with say sunglasses, we need to have of the images in our database that includes this information one way or another. One way would be to annotate all images with (meta) information and describe the objects in the picture to some degree of detail. More challenging would be to extract image content by image analysis, and produce the description (semi) automatically.

According to [MMDBMS], content-based description of images involves the identification of objects, as well as an indication of where these objects are located in the image, by using a *shape descriptor* and possibly *property descriptors* indicating the pictorial properties of a particular region of the object or image.

Shape and property descriptors may take a form as indicated below.

shape

- bounding box – (XLB,XUB,YLB,YUB)

property

- property – name=value

As an example of applying these descriptors.

example

> shape descriptor: XLB=10; XUB=60; YLB=3; YUB=50
> property descriptor: pixel(14,7): R=5; G=1; B=3

Now, instead of taking raw pixels as the unit of analysis, we may subdivide an image in a grid of cells and establish properties of cells, by some suitable algorithm.

definitions

- image grid: $(m * n)$ cells of equal size
- cell property: (Name, Value, Method)

As an example, we can define a property that indicates whether a particular cell if black or white.

example

> property: (bwcolor,{b,w},bwalgo)

The actual algorithm used to establish such a property might be a matter of choice. So, in the example it is given as an explicit parameter.

From here to automatic content description is, admittedly, still a long way. We will indicate some research directions at the end of this section.

similarity-based retrieval

We need not necessarily know what an image (or segment of it) depicts to establish whether there are other images that contain that same thing, or something similar to it. We may, following [MMDBMS], formulate the problem of similarity-based retrieval as follows:

> *How do we determine whether the content of a segment (of a segmented image) is similar to another image (or set of images)?*

Think of, for example, the problem of finding all photos that match a particular face.

According to [MMDBMS], there are two solutions:

- *metric approach* – distance between two image objects
- *transformation approach* – relative to specification

As we will see later, the transformation approach in some way subsumes the metric approach, since we can formulate a distance measure for the transformation approach as well.

metric approach What does it mean when we say, the distance between two images is less than the distance between this image and that one. What we want to express is that the first two images (or faces) are more alike, or maybe even identical.

Abstractly, something is a distance measure if it satisfies certain criteria.

metric approach

distance $d : X \rightarrow [0, 1]$ is distance measure if:

d(x,y) = d(y,x)
d(x,y) ⩽ d(x,z) + d(z,y)
d(x,x) = 0

For your intuition, it is enough when you limit yourself to what you are familiar with, that is measuring distance in ordinary (Euclidian) space.

Now, in measuring the distance between two images, or segments of images, we may go back to the level of pixels, and establish a distance metric on pixel properties, by comparing all properties pixel-wise and establishing a distance.

pixel properties

- objects with pixel properties $p_1, \ldots, p_n$
- pixels: $(x, y, v1, \ldots, v_n)$
- object contains w x h (n+2)-tuples

Leaving the details for your further research, it is not hard to see that even if the absolute value of a distance has no meaning, relative distances do. So, when an image contains a face with dark sunglasses, it will be closer to (an image of) a face with dark sunglasses than a face without sunglasses, other things being equal. It is also not hard to see that a pixel-wise approach is, computationally, quite complex. An object is considered as

complexity

a set of points in k-dimensional space for k = n + 2

In other words, to establish similarity between two images (that is, calculate the distance) requires n+2 times the number of pixels comparisons.

feature extraction Obviously,we can do better than that by restricting ourselves to a pre-defined set of properties or features.

feature extraction

- maps object into s-dimensional space

For example, one of the features could indicate whether or not it was a face with dark sunglasses. So, instead of calculating the distance by establishing color differences of between regions of the images where sunglasses may be found, we may limit ourselves to considering a binary value, yes or no, to see whether the face has sunglasses.

Once we have determined a suitable set of features that allow us to establish similarity between images, we no longer need to store the images themselves, and can build an index based on feature vectors only, that is the combined value on the selected properties.

Feature vectors and extensive comparison are not exclusive, and may be combined to get more precise results. Whatever way we choose, when we present an image we may search in our image database and present all those objects that fall within a suitable *similarity range*, that is the images (or segments of images) that are close enough according to the distance metric we have chosen.

transformation approach Instead of measuring the distance between two images (objects) directly, we can take one image and start modifying that until it exactly equals the target image. In other words, as phrased in [MMDBMS], the principle underlying the transformation approach is:

transformation approach

> *Given two objects* o1 *and* o2*, the level of* dissimilarity *is proportional to the (minimum) cost of transforming object* o1 *into object* o2 *or vice versa*

Now, this principle might be applied to any representation of an object or image, including feature vectors. Yet, on the level of images, we may think of the following operations:

$to_1, \ldots, to_r$ – translation, rotation, scaling

Moreover, we can attach a cost to each of these operations and calculate the cost of a transformation sequence TSby summing the costs of the individual operations. Based on the cost function we can define a distance metric, which we call for obvious reasons the *edit distance*, to establish similarity between objects.

cost

- $cost(TS) = \Sigma_{i=1}^{r} cost(to_i)$

distance

- d(o,o') = min { cost(TS) | TS in TSeq(o,o') }

An obvious advantage of the *edit distance* over the pixel-wise distance metric is thatwe may have a rich choice of transformation operators that we can attach (user-defined) cost to at will.

For example, we could define low costs for normalization operations, such as scaling and rotation, and attach more weight tooperations that modify color values or add shapes. For face recognition, for example, we could attribute low cost to adding sunglasses but high cost to changing the sex.

To support the *transformation approach* at the image level, our image database needs to include suitable operations. See [MMDBMS].

operations

```
rotate(image-id,dir,angle)
segment(image-id, predicate)
edit(image-id, edit-op)
```

We might even think of storing images, not as a collection of pixels, but as a sequence of operations on any one of a given set of base images. This is not such a strange idea as it may seem. For example, to store information about faces we may take a base collection of prototype faces and define an individual face by selecting a suitable prototype and a limited number of operations or additional properties.

research directions – *multimedia repositories*

What would be the proper format to store multimedia information? In other words, what is the shape multimedia repositories should take? Some of the issues involved are discussed in chapter **??**, which deals with information system architectures. With respect to image repositories, we may rephrase the question into *what support must an image repository provide, minimally, to allow for efficient access and search?*. In [MMDBMS], we find the following answer:

image repository

- *storage* – unsegmented images
- *description* – limited set of features
- *index* – feature-based index
- *retrieval* – distance between feature vectors

And, indeed, this seems to be what most image databases provide. Note that the actual encoding is not of importance. The same type of information can be encoded using either XML, relational tables or object databases. What is of importance is the functionality that is offered to the user, in terms of storage and retrieval as well as presentation facilities.

What is the relation between presentation facilities and the functionality of multimedia repositories? Consider the following mission statement, which is taken from my research and projects page.

mission

> *Our goal is to study aspects of the deployment and architecture of virtual environments as an interface to (intelligent) multimedia information systems ...*

Obviously, the underlying multimedia repository must provide adequate retrieval facilities and must also be able to deliver the desired objects in a format suitable for the representation and possibly incorporation in such an environment. Actually, at this stage, I have only some vague ideas about how to make this vision come through. Look, however, at chapter **??** and appendix **??** for some initial ideas.

4.3 documents

Even in the presence of audiovisual media, text will remain an important vehicel for human communication. In this section, we will look at the issues that arise in querying a text or document database. First we will characterize more precisely what we mean by effective search, and then we will study techniques to realize effective search for document databases.

Basically, answering a query to a document database comes down to string matching. However, some problems may occur such as synonymy and polysemy.

problems

- synonymy – topic T does not occur literally in document D
- polysemy – some words may have many meanings

As an example, *church* and *house of prayer* have more or less the same meaning. So documents about churches and cathedrals should be returned when you ask for information about 'houses of prayer'. As an exampleof polysemy, think of the word *drum*, which has quite a different meaning when taken from a musical perspective than from a transport logistics perspective.

precision and recall

Suppose that, when you pose a query, everything that is in the database is returned. You would probably not be satisfied, although every relevant document will be included, that is for sure. On the other hand, when nothing is returned, at least you cannot complain about non-relevant documents that are returned, or can you?

In [MMDBMS], the notions of *precision* and *recall* are proposed to measure the effectiveness of search over a document database. In general, precision and recall can be defined as follows.

effective search

- precision – how many answers are correct
- recall – how many of the right documents are returned

For your intuition, just imagine that you have a database of documents. With full knowledge of the database you can delineate a set of documentsthatare of relevance to a particular query. Also, you can delineate a set that will be returned by some given search algorithm. Then, *precision* is the intersection of the two sets in relation to whatthe search algorithm returns, and *recall* that same intersection in relation to what is relevant. In pseudo-formulas, we can express this as follows:

precision and recall

precision = (returned and relevant) / returned
recall = (returned and relevant) / relevant

Now, as indicated in the beginning, it is not to difficult to get either perfect recall (by returning all documents) or perfect precision (by returning almost nothing). But these must be considered anomalies (that is, sick cases), and so the problem is to find an algorithm that performs optimally with respect to both precision and recall.

For the total database we can extend these measures by taking the averages of precision and recall for all topics that the database may be queried about.

Can these measures only be applied to document databases? Of course not, these are general measures that can be applied to search over any media type!

frequency tables

A *frequency table* is an example of a way to improve search. Frequency tables, as discussed in [MMDBMS], are useful for documents only. Let's look at an example first.

example

term/document	d0	d1	d2
snacks	1	0	0
drinks	1	0	3
rock-roll	0	1	1

Basically, what a frequency table does is, as the name implies, give a frequency count for particular words or phrases for a number of documents. In effect, a complete document database may be summarized in a frequency table. In other words, the frequency table may be considered as an index to facilitate the search for similar documents.

To find a similar document, we can simply make a word frequency count for the query, and compare that with the colums in the table. As with images, we can apply a simpledistance metric to find the nearest (matching) documents. (In effect, we may take the square root for the sum of the squared differences between the entries in the frequence count as our distance measure.)

The complexity of this algorithm may be characterized as follows:

complextity

compare term frequencies per document – O(M*N)

where M is the number of terms and N is the number of documents. Since both M and N can become very large we need to make an effort to reduce the size of the frequency table.

reduction

- stop list – irrelevant words
- word stems – reduce different words to relevant part

We can, for example, introduce a *stop list* to prevent irrelevant words to enter the table, and we may restrict ourselves to including *word stems* only, to bring back multiple entries to one canonical form. With some additional effort we could even deal with synonymy and polysemy by introducing, respectively equivalence classes, and alternatives (although we then need a suitable way for ambiguation). By the way, did you notice that frequency tables may be regarded as feature vectors for documents?

research directions– *user-oriented measures*

Even though the reductions proposed may result in limiting the size of the frequency tables, we may still be faced with frequency tables of considerable size. One way to reduce the size further, as discussed in [MMDBMS], is to apply *latent sematic indexing* which comes down to clustering the document database, and limiting ourselves to the most relevant words only, where relevance is determined by the ratio of occurrence over the total number of words. In effect, the less the word occurs, the more discriminating it might be. Alternatively,the choice of what words are considered relevant may be determined by taking into account the area of application or the interest of a particular group of users.

user-oriented measures Observe that, when evaluating a particular information retrieval system, the notions of precision and recall as introduced before are rather system-oriented measures, based on the assumption of a user-independent notion of relevance. However, as stated in [IR], different users might have a different interpretation on which document is relevant. In [IR], some user-oriented measures are briefly discussed, that to some extent cope with this problem.

user-oriented measures

- *coverage ratio* – fraction of known documents
- *novelty ratio* – fraction of new (relevant) documents
- *relative recall* – fraction of expected documents
- *recall effort* – fraction of examined documents

Consider a reference collection, an example information request and a retrieval strategy to be evaluated. Then the *coverage ratio* may be defined as the fraction of the documents known to be relevant, or more precisely the number of (known) relevant documents retrieved divided by the total number of documents known to be relevant by the user.

The *novelty ratio* may then be defined as the fraction of the documents retrieved which were not known to be relevant by the user, or more precisely the number of relevent documents that were not known by the user divided by the total number of relevant documents retrieved.

The *relative recall* is obtained by dividing the number of relevant documents found by the number of relevant documents the user expected to be found.

Finally, *recall effort*may be characterized as the ratio of the number of relevant documents expected and the total number of documents that has to be examined to retrieve these documents.

Notice that these measures all have a clearly ’subjective’ element, in that, although they may be generalized to a particular group of users, they will very likely not generalize to all groups of users. In effect, this may lead to different retrieval strategies for different categories of users, taking into account levelof expertise and familiarity with the information repository.

questions

information retrieval

1. (*) What is meant by the *complementarity of authoring and retrieval*? Sketch a possible scenario of (multimedia) information retrieval and indicate how this may be implemented. Discuss the issues that arise in accessing multimedia information and how content annotation may be deployed.

concepts

2. How would you approach *content-based description* of *images*?
3. What is the difference between a *metric* approach and the *transformational* approach to establishing similarity between images?
4. What problems may occur when searching in text or document databases?

technology

5. Give a definition of: *shape descriptor* and *property descriptor*. Give an example of each.
6. How would you define *edit distance*?
7. Characterize the notions *precision* and *recall*.
8. Give an example (with explanation) of a *frequency table*.