COMP30311: Advanced Databases

Introduction to Multimedia Databases

Goran Nenadic
University of Manchester
g.nenadic@manchester.ac.uk

Multimedia data types

- Traditional data types
  - numbers, characters, strings, dates
  - structured data with 'clear' meaning
- Multimedia data
  - text, graphics (drawings, illustrations), images, animations, audio, video, composed (mixed) multimedia
  - un- and semi-structured data, no 'clear' or correct meaning assigned

Example: insurance company

- Data types in an insurance company
  - accident reports (text)
  - images of accidents (image)
  - reconstruction of accident (video, animation)
  - audio recording of the parties involved (audio)
  - medical reports (text)
  - supporting medical materials (images)
- Query example
  - retrieve all accidents with broken arms that have happened on zebra crossing and involve caravan cars

What is presented here?

ID	Name	Salary
1	John	23,000
2	Mary	23,200
3	Ann	43,000
4	Pete	36,000
5	Mark	33,000
6	Emma	40,100
7	Ian	20,100

What is presented here?

Traditional databases

- In traditional DBs, data is described via attributes, with "known" meaning and possible values (e.g. range)
  - meta-data describe the attributes, not the data

    product(ID, type, colour, weight)
  - retrieval is simple and accurate
  - no "subjective interpretation"

continued

Traditional databases

- Is a traditional DB with stored multimedia objects a multimedia DB?
  - e.g. a DB containing holiday photos and videos

ID	date	place	object (photo/video)
1	2.4.1998.	Paris	12345.jpg
2	2.4.1998.	Paris	12346.jpg
3	7.5.2004.	Milan	12.mpeg
4	8.5.2004.	Paris	34.jpg

Multimedia DBs

- Multimedia databases (MDBs) are DBs that manipulate multimedia data
- MDBs provide
  - efficient models for storage and querying various types of multimedia information, in addition to basic functionalities such as insert new data, retrieve and change existing data, delete data
- Multimedia data will dominate many (most) aspects of business, science and leisure IS
  - handling such data is thus important!

Multimedia: main challenge

- How to assign "meaning" to multimedia so that we can retrieve data based on its content?
  - **content-based retrieval**
- How to represent **semantics** of multimedia?
  - e.g. *what does this image represent/mean*?
  - *retrieve all photos with President Obama smiling*?
- Two main problems
  - information is mainly implicit
  - interpretation can be subjective: an object means different things to different people

Multimedia DB – plan

- Nature of multimedia data
  - Describing multimedia: meta-data
  - challenges
- Multimedia data management
  - retrieval models
  - storing multimedia data
  - querying data

10

Meta-data: data about data

- In traditional DBMS, metadata describes "simple" features assigned to relations/tables
  - schema definition, meaning of attributes, attribute ranges, integrity, primary/foreign keys, ...
- In multimedia DBMS, meta-data describes semantics of relations and individual objects
  - meta-data captures semantics and makes it as explicit as possible
  - how to generate meta-data?

continued

Meta-data: data about data

- Use of metadata is essential for querying multimedia databases
- Generate meta-data when storing an object
  - this is one of the main issues in multimedia DBs
- Requirements for generating meta-data
  - apply a model known/used by expected users
  - support "near" matches
    - cater for interpretation and subjectivity (both of those who provide and those who use the meta-data)

continued

Meta-data: data about data

- Query multimedia databases by querying meta-data
  - meta-data is explicit
- Model and content of meta-data depends on the application area
- *Subjectivity*: different interpretations of the content both
  - when providing meta-data and
  - when querying

continued

Meta-data: data about data

- Types of meta-data
  - textual descriptions: short summaries of a given multimedia object
  - annotations: descriptors (i.e. categories) assigned from a set of values
    - open-ended keywords
    - controlled vocabularies (pre-defined, no structure)
    - taxonomies and ontologies (pre-defined, structured vocabularies)

Textual descriptions

- Query a multimedia object by querying its textual representation
- Still difficult to process and "understand"
  - *variability*: numerous ways to express features, entities and relationships among them
  - *ambiguity*: different meanings attached to the same expressions
- Text-based querying and retrieval is not easy
  - (see later, text processing)

Lists of keywords

- Making the content explicit by providing
  - an index to a repository of objects
  - a query model based on keywords
- Open-ended keyword lists
  - e.g. social networks
- Controlled vocabularies
  - pre-defined **set** of descriptors (i.e. keywords) that represent a common terminology (i.e. categories) over a domain
  - no links between descriptors/keywords
  - problem is in "coverage"

Lists of keywords - ontologies

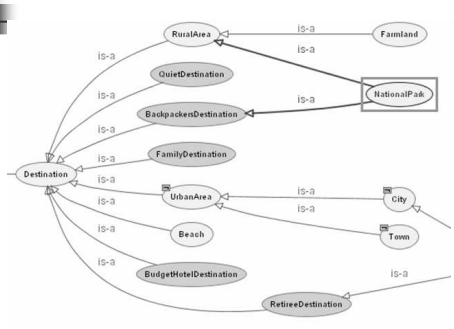
- An ontology is an explicit description of concepts in a domain, that is designed for sharing and reuse across multiple applications
- Used to "organise" and represent knowledge
  - includes links and different relations between concepts
  - [taxonomies are seen as less broad as they rely mainly on parent-child (is_a) hierarchies]

continued

Ontologies

- Define the basic concepts and various relations as a community reference
  - for search, integration and exchange of data
- Used for
  - *modelling*: a well founded model of the reality
  - *communication*: knowing what we are talking about
  - *annotation:* describing objects
  - *reasoning*: drawing conclusion from the knowledge
    - e.g. *City* is an *UrbanArea* and is a *Destination*

Ontology example (1)



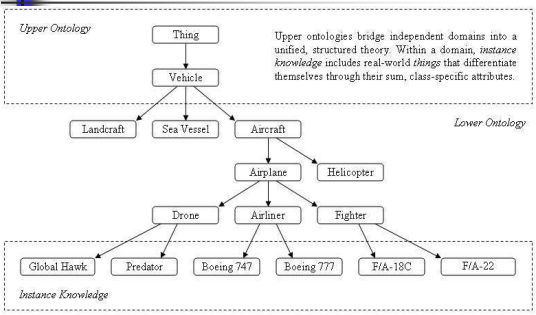
Using ontologies for annotation

- Assign a set of ontological nodes (from chosen ontologies) to multimedia objects or their attributes
  - nodes are used as potential *descriptors*
  - can be from several ontologies explaining different data aspects (represented by different attributes)
- Use logical representation (e.g. hierarchy) to infer additional relations
  - e.g. retrieving 'RuralArea' objects would also include all entities *annotated* as 'NationalPark'

Example annotation

- Farmland (is_a RuralArea)
- Quiet destination
- Retiree destination

Ontology example (2)

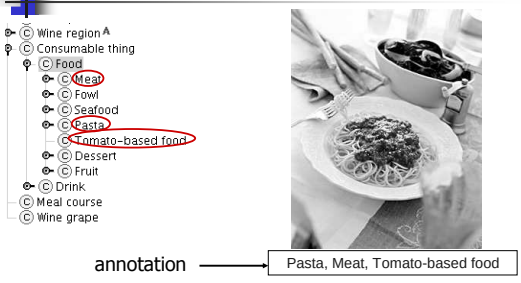


Ontology example (3)

Organization **Business** Corporation Partnership Non-Profit Government_Body Public Private **Department** Marketing Accounting Human_Resources **Employee** Manager President Staff Intern	**Industry** Agriculture Farm Ranch Finance Bank Brokerage Food Food_Distribution Food_Wholesale Restaurant Supermarket Food_Production Beverages Insurance Accident Casualty Health

example

Image annotation



annotation → Pasta, Meat, Tomato-based food

exercise

Annotate this image

using (a) text description, (b) your own keywords, (c) controlled keywords; (d) ontology

Consumables
Pork
Spinach
Wine
Carrots
Beverage
Food
Beef
Meat
Banana
Vegetables
Red Wine
Apple
White Wine
Fruit

compare the results

Problems with ontologies

- Need to be widely accepted to be useful
  - not easy to agree on an ontology for a given problem
  - updating, versions etc.
- Annotation (i.e. assigning ontological terms to an object) is difficult
  - one must know the ontology/keywords to be able to annotate and/or query
  - subjective (differences in selecting concepts for annotation)

Assigning meta-data

- Manually
  - need to know descriptors or write text description
  - subjective and can be very costly (e.g. 7 min needed for an image)
- Automatic indexing
  - extract/assign descriptors automatically
  - e.g. pattern-based: identify simple patterns in an image, and "index" it with this controlled set of patterns
  - inferring features: e.g. when/where/who has taken a photo

Summary so far

- Multimedia databases are important
  - text, graphics, images, audio, video
  - numerous application areas
- Main challenge: representing semantics
  - generate and maintain meta-data that describe multimedia objects
- Meta-data
  - text descriptions
  - annotations using ontologies, controlled or open-ended keywords

28

Multimedia data management

Multimedia data management

- Modelling and designing MDBs
  - database vs. information retrieval
- Storage
  - representation, compression, archiving, buffering
- Querying and retrieval
  - content-based indexing and retrieval

Modelling: DB vs. IR

- How to support search, querying and retrieval
- Database model
  - rigid structure
  - precise results: Boolean "match" (over meta-data)
  - no ranking of results (all are good i.e. correct)
- Information retrieval (IR) model
  - use meta-data to obtain a set of ranked "answers"
  - rank reflects how well the result matches the query
  - retrieved objects are based on **similarity**

Storage models

- How to represent and store multimedia objects and their meta-data
- Main problems
  - storing meta-data with multimedia objects
  - huge size (compared to traditional data types)
- Many DBs provide data types that can be used to store multimedia objects
  - BLOB, CLOB, DATALINK, etc.
  - XML documents

continued

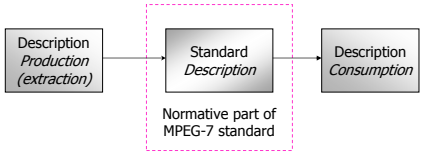
Storage models - examples

- BLOB: binary large object
  - a collection of binary data stored as a single entity
  - for images, videos, and sound, programs (2-4GB)
- CLOB: character large object
  - for storing text data; e.g. TEXT (mySQL)
- Storing objects externally (as files, URL, etc.)
  - data types: DATALINK, BFILE
  - typically read-only access
  - standard data formats for multi-media

example

MPEG-7

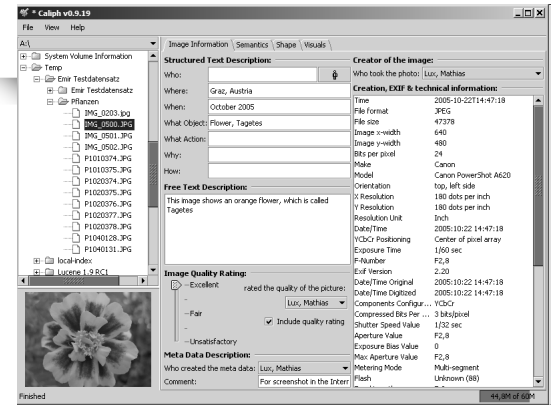
- **MPEG-7**: a data standard that allows integration of metadata and facilitates content-based retrieval of audio, video and image data
  - does not offer any solutions to feature extraction, use of descriptions, similarity between contents,
  - just for storage



example

MPEG-7

- Standardises
  - a core set of **Descriptors** (Ds) that can be used to describe the various features of multimedia content
    - e.g. Visual Descriptors (Color Descriptors, Texture Descriptors, Shape Descriptors, Motion Descriptors for Video)
  - pre-defined structures of Descriptors and their relationships, called **Description Schemes** (DSs)
  - a language to define Description Schemes and Descriptors, called the **Description Definition Language** (DDL);



http://www.semanticmetadata.net/features/

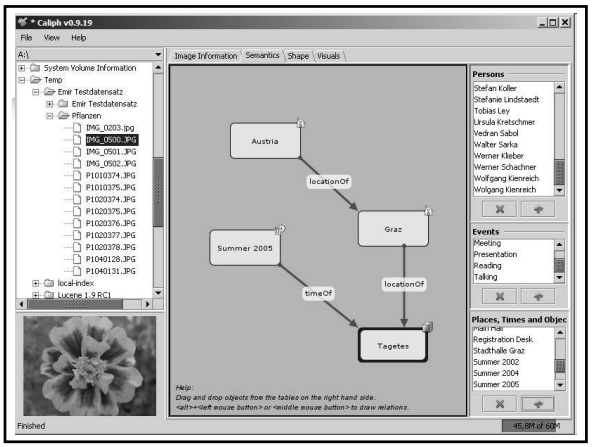
An example tool: Caliph

- Caliph is an MPEG-7 based Java prototype for digital photo and image annotation supporting graph like annotation for semantic metadata and content based image retrieval using MPEG-7 descriptors.
- Extraction of following image retrieval descriptors
  - ColorLayout (measures color distribution in an image)
  - ScalableColor (basically a color histogram)
  - EdgeHistogram (measures *edginess*)
  - DominantColor (measure dominant colors)
- Annotation of images with
  - Free text metadata
  - Quality rating
  - Structured text metadata
  - Meta^2 data (metadata about metadata)

http://www.semanticmetadata.net/features/

An example tool: Caliph

- Semantic Annotations
  - Based on a editable catalogue for reusable semantic objects (like specific persons, places, …)
  - Fully visual & graphical user interface
  - Automatic placement and layout of visual descriptions
  - Export of semantic descriptions to Structured Text Descriptions
- Annotation of shapes
- Load and save annotations as MPEG-7 XML file



continued

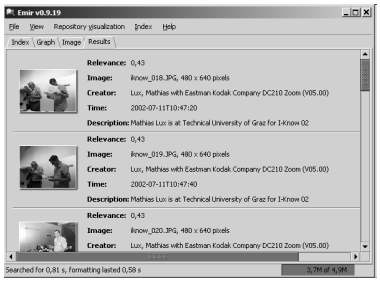
MPEG-7

- Potential benefits
  - retrieval engines can process values of defined standard parameters
    - and known non-standard ones (see *Emir*)
  - improve interoperability – improve searching for image/video/audio data
- Problems
  - define a set of preferred/useful retrieval features

http://www.semanticmetadata.net/features/

An example tool: Emir

- Metadata-based image retrieval
- supports retrieval from photo repositories created with Caliph



XML and multimedia

- XML can be used to describe multimedia objects
- Element and attribute values can be used for
  - "free" textual descriptions
  - annotations (by keywords, controlled vocabularies, ontologies)
  - other important features (author, date, etc.)
- Multimedia XML databases
  - use an XSD to define a schema for multimedia DB
  - query multimedia XML representations (XQuery, XPath)

http://www.semanticmetadata.net/features/

Caliph and Emir

- In Caliph, the MPEG-7 description consists of
  - metadata description, creation information, media information, textual annotation, semantics, visual descriptors.
  - all annotations saved as MPEG-7 XML file
- Emir uses metadata for image retrieval tool
  - with XPath queries
  - with keyword queries
  - with quality constraints
  - with example images for content based image retrieval

A Simplified MPEG-7 Example

```
<VideoSegment id = "VS1" >
        <MediaTime>
                <MediaTimePoint> T0:0:0   </MediaTimePoint>
                <MediaDuration>   PT10M  </MediaDuration>
        </MediaTime>

        <StructuredAnnotation>
                This is an example of video segment
        </StructuredAnnotation>

        <GofGopColorHistogram HistogramTypeInfo = "Average">
                ....
        </GofGopColorHistogram>

        <SegmentDecomposition Gap = "true" Overlap = "true" DecompositionType = "temporal">
                <VideoSegment id = "VS2" > .... </VideoSegment>
                <VideoSegment id = "VS3" > .... </VideoSegment>
                <VideoSegment id = "VS4" > .... </VideoSegment>
        </SegmentDecomposition>
</VideoSegment>
```

example

Multimedia object representation

```
<Description xsi:type="ContentEntityType">
 <MultimediaContent xsi:type="VideoType">
  <Video>
   <TemporalDecomposition>
    <VideoSegment>
     <TextAnnotation type="scene" relevance="1" confidence="1">
      <FreeTextAnnotation>Indoors</FreeTextAnnotation>
      <FreeTextAnnotation>Laboratory</FreeTextAnnotation>
      <FreeTextAnnotation>Person_Speaking</FreeTextAnnotation>
     </TextAnnotation>
     <MediaTime>
      <MediaTimePoint>T00:00:00:0F25</MediaTimePoint>
      <MediaIncrDuration mediaTimeUnit="PT1N25F">916</MediaIncrDuration>
     </MediaTime>
    </VideoSegment>
   </TemporalDecomposition>
  </Video>
 </MultimediaContent>
</Description>
```

e.g. open-ended keywords

example


```
<PubmedArticle>
    <MedlineCitation Owner="NLM" Status="MEDLINE">
        <PMID>15542023</PMID>
        <DateCreated>
            <Year>2004</Year>
            <Month>11</Month>
            <Day>15</Day>
        </DateCreated>
        <DateCompleted>
            <Year>2005</Year>
            <Month>04</Month>
            <Day>18</Day>
        </DateCompleted>
        <Article PubModel="Print">
            <Journal>
                <ISSN IssnType="Print">1532-0464</ISSN>
                <JournalIssue CitedMedium="Print">
                    <Volume>37</Volume>
                    <Issue>6</Issue>
                    <PubDate>
                        <Year>2004</Year>
                        <Month>Dec</Month>
                    </PubDate>
                </JournalIssue>
                <Title>Journal of biomedical informatics. </Title>
            </Journal>
            <ArticleTitle>Term identification in the biomedical literature.</ArticleTitle>
            <Pagination>
                <MedlinePgn>512-26</MedlinePgn>
            </Pagination>
            <Abstract>
                <AbstractText>Sophisticated information technologies are needed for effective data acquisition
and integration from a growing body of the biomedical literature. Successful term identification is key
to getting access to the stored literature information, as it is the terms (and their relationships)
that convey knowledge across scientific articles. Due to the complexities of a dynamically changing biomedical
terminology, term identification has been recognized as the current bottleneck in text mining, and--as
a consequence--has become an important research topic both in natural language processing and biomedical
communities. This article overviews state-of-the-art approaches in term identification. The process of
identifying terms is analysed through three steps: term recognition, term classification, and term mapping.
For each step, main approaches and general trends, along with the major problems, are discussed. By assessing
previous work in context of the overall term identification process, the review also tries to delineate
needs for future work in the field.</AbstractText>
```


Object features: date, title, etc.

textual description

example


```
        <Author ValidYN="Y">
            <LastName>Hemadic</LastName>
            <ForeName>Goran</ForeName>
            <Initials>G</Initials>
        </Author>
    </AuthorList>
    <Language>eng</Language>
    <PublicationTypeList>
        <PublicationType>Journal Article</PublicationType>
    </PublicationTypeList>
</Article>
<MedlineJournalInfo>
    <Country>United States</Country>
    <MedlineTA>J Biomed Inform</MedlineTA>
    <NlmUniqueID>100970413</NlmUniqueID>
</MedlineJournalInfo>
<CitationSubset>IM</CitationSubset>
<MeshHeadingList>
    <MeshHeading>
        <DescriptorName MajorTopicYN="N">Abbreviations</DescriptorName>
    </MeshHeading>
    <MeshHeading>
        <DescriptorName MajorTopicYN="N">Abstracting and Indexing</DescriptorName>
        <QualifierName MajorTopicYN="Y">methods</QualifierName>
    </MeshHeading>
    <MeshHeading>
        <DescriptorName MajorTopicYN="N">Algorithms</DescriptorName>
    </MeshHeading>
    <MeshHeading>
        <DescriptorName MajorTopicYN="N">Animals</DescriptorName>
    </MeshHeading>
    <MeshHeading>
        <DescriptorName MajorTopicYN="N">Artificial Intelligence</DescriptorName>
    </MeshHeading>
    <MeshHeading>
        <DescriptorName MajorTopicYN="N">Computational Biology</DescriptorName>
        <QualifierName MajorTopicYN="Y">methods</QualifierName>
```


More object features: authors, language, etc.

ontological annotation

example

Example

```
<?xml version= "1.0"?>
<?xml-stylesheet href="img.xsl" type ="text/xsl"?>
<image>
  <Professor>
    <name> K.Cook</name>
    <degree>Ph.D.</degree>
    <granting-institution>Univ. of Michigan</granting-institution>
  </Professor>
  <Student>
    <name>Jane</name>
    <major>computer science</major>
    <enrollment>2002</enrollment>
  </Student>
</image>
```

Not only in XML

- Multimedia can be stored using other database models, e.g.
  - relational or
  - object-relational
    - different media types represented by classes, attributes used for annotation
- Most relational and object-relational databases have support for multimedia data types
  - BLOB, CLOB, etc.

Querying and retrieval

- "Trivial" for traditional attributes and types
  - e.g. author, date, location
- Content-based querying can be based on
  - pre-assigned meta-data
    - querying by descriptors (e.g. using SQL, XPath, XQuery)
    - text-based querying
  - similarity calculation (similarity-based querying)
    - querying by objects

Using meta-data for querying

- Restriction/filtering by using descriptors
  - query XML representation (XQuery, XPath)

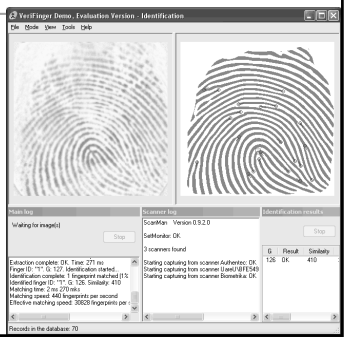
    FOR $x IN /books/book[@keyword ="cycling"]/title
    RETURN $x
  - query relational representation (SQL)
- Querying textual descriptions
  - built-in operators (LIKE, MATCH()… AGAINST)
  - use text mining techniques (IR, IE, QA)

Similarity-based querying

- Retrieval based on similarity calculation
  - "*find images that are similar to a given one*" (e.g. in a finger prints database)
  - define **distance function** among objects
  - typically on segment-to-segment bases
    - use e.g. colour layout (measures colour distribution in an image), edge histograms, dominant colour etc
  - if distance is low – there is possibility of match
  - can be used for images, audio, text

Example

biometric authentication (fingerprints, eye retinas and irises)

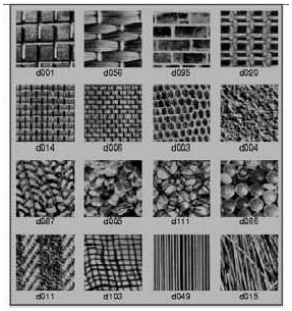


Example – texture matching

- Texture characterises small-scale regularity
  - colour describes pixels, texture describes regions
- Described by several types of features
  - smoothness (changes within texture), periodicity (how often pattern changes), directionality, etc.
- Retrieval aims to match region size with image characteristics
  - select images with more of a given texture
- Perform weighted vector space matching
  - usually in combination with a colour histogram

example

Example – texture matching



Transformation-based querying

- Transformation of query objects
  - transform a query object in order to improve chance of finding a match
  - e.g. rotation, translation, scaling
  - may be computationally time consuming

Example

- Automatic number plate recognition
  - uses optical character recognition (OCR) on images to read licence plates on vehicles
  - used by police, for electronic toll collection, monitoring traffic activity
  - uses a series of image manipulation techniques to detect, normalise and enhance the image of the number plate, and then OCR to extract the alpha-numerics of the plate

examples

Content-based image retrieval

- Commercial systems
  - Virage
  - QBIC (IBM)
- Academic
  - Blobworld
  - VisualSeek
  - Chabot
  - Viper (http://viper.unige.ch; http://viper.unige.ch/demo/php/demo.php)
  - Caliph and Emir

Summary

- Storage models:
  - built-in data types + meta-data
- Use XML for storing/describing multimedia
  - semi/un-structured data
- Querying
  - meta-data based (e.g. XPath, XQuery), text-based
  - similarity-based + transformation
- Be prepared to
  - use XML/XSD to model and/or represent multimedia
  - query XML descriptions using XPath/XQuery

Further reading

- Chapters 24 (24.3.2) and 29 (29.2, 29.3) in [Elmasri & Navathe]
- CJ Date: An Introduction to Database Systems (8th ed.), Chapter 27 [contains high level of details]
- L. Dunckley, Multimedia Databases: An Object Relational Approach, Addison-Wesley, 2003
- http://www.w3.org/
- http://www.w3schools.com/
- RIAO conferences on Large-Scale Semantic Access to Content (Text, Image, Video and Sound)
  - http://riao.free.fr/

60