Introduction: Multimedia Databases
Outline of Today’s Lecture

- What is multimedia data?
- What are multimedia data applications?
- How should multimedia applications access different media types?
- What kinds of operations do users wish to perform on multimedia data?
- What infrastructure is needed to support these functionalities:
  - **Content**: Extracting media content. Indexing it. Querying it.
  - **Physical Storage**: Storing media data on primary, secondary, and tertiary storage devices. Building media servers.
  - **Creating Presentations**: How should a media presentation be created (perhaps in response to a query)? How should it be delivered to users at remote sites?
- Overview of the course
Today’s Media Types

- Text/Document
- Image
- Video
- Audio
- “Classical” Data (e.g. relations, flat files, object bases, etc.).

Video and audio differ from the other media types listed above because of their temporal nature. In particular:

- Video/Audio retrievals must appear to be continuous, hiccup-free presentations.
- Video/Audio support operations like fast-forward, rewind, and pause, that were not supported by classical data types.
What is an MM_DBMS

A multimedia database management system (MM_DBMS) is a framework that manages different types of data potentially represented in a wide diversity of formats on a wide array of media sources. An MM_DBMS must:

- Have the ability to uniformly query data (media data, textual data) represented in different formats.

- Have the ability to simultaneously query different media sources and conduct classical database operations across them.

- Have the ability to retrieve media objects from a local storage device in a smooth jitter free (i.e. continuous) manner.

- Have the ability to take the answer generated by a query (the notion of “answer to a query” may be mathematical structure of some sort) and develop a presentation of that answer in terms of audio-visual media.

- Have the ability to deliver this presentation in a way that satisfies various Quality of Service requirements.
Sample Multimedia Scenario

Consider a police investigation of a large-scale drug operation. This investigation may generate the following types of data:

- **Video** data captured by surveillance cameras that record the activities taking place at various locations.
- **Audio** data captured by legally authorized telephone wire-taps.
- **Image** data consisting of still photographs taken by investigators.
- **Document** data seized by the police when raiding one or more places.
- **Structured relational data** containing background information, bank records, etc., of the suspects involved.
- **Geographic information systems** data containing geographic data relevant to the drug investigation being conducted.
Data sources used in Sample Multimedia Scenario

- Surveillance Video
- Audio (phone) data
- Geographic Info.
- Police Application
- Still Image Data
- Relational Data
- Document Data
Example Image Queries for Multimedia Scenario

Query 1:

- Police officer John Macho-Dude has a photograph in front of him.
- He wants to find the identity of the person in the picture.
- Query: “Retrieve all images from the image library in which the person appearing in the (currently displayed) photograph appears.”

Query 2:

- Police officer John Macho-Dude wants to examine pictures of Denis Dopeman.
- Query: “Retrieve all images from the image library in which Denis Dopeman appears.”
Issues Raised by Example Image Queries

- Two basic kinds of queries:
  - Image-based queries
  - Keyword-based queries
- In the first, police officer John Macho-Dude, gives a image as input (query image). He wants as output, a ranked list of images that are “similar” to the query image.
- To support this, we need to know what “similarity” means.
- We need to know what “ranking” means.
- We need to be able to efficiently support these operations.
- In the second, police officer John Macho-Dude, gives a keyword as input (name of suspect – Denis Dopeman). He wants as output, those photographs that are known to contain an image-object whose name attribute is Denis Dopeman.
- To support this, we need to know how to associate different attributes with images (or parts of images).
- We need to know how to effectively index and retrieve images based on such attributed.
Example Audio Query in Sample Multimedia Scenario

Query 1:

- Police officer John Macho-Dude is listening to an audio surveillance tape.
- Example: Tape contains a conversation between individual A (person under surveillance) and individual B (somebody meeting person A).
- Query: “Find the identity of individual B, given that individual A is Denis Dopeman.”

Query 2:

- Police officer John Macho-Dude wants to review all audio logs that Denis Dopeman participated in during some specified time period.
- Query: “Find all audio tapes in which Denis Dopeman was a participant.”
Example Text/Video Query

Text Query:

- Police officer John Macho-Dude is browsing an archive of text documents – these include old newspaper archives, police department files on old, unsolved murder cases, witness statements, etc.

- Query: “Find all documents (from the corpora of text documents) that deal with the Cali drug cartel’s financial transactions with ABC Corp.”

Video Query:

- Police officer John Macho-Dude is examining a surveillance video of a particular person being fatally assaulted by an assailant. However, the assailant’s face is occluded and image processing algorithms return very poor matches. John Macho-Dude thinks the assault was by someone known to the victim.

- Query: “Find all video segments in which the victim of the assault appears.”

- By examining the answer to the above query, John Macho-Dude hopes to find other people who have previously interacted with the victim.
Simple Heterogeneous Query

- All queries discussed thus far involve one media type.
- For example, each query accesses only image or audio or video data, but does not access a mix of these media types.
- Complex queries will “mix and match” data from these different media sources.
- Such “mixing and matching” is difficult, even for purely textual data sources.
- TEXTUAL EXAMPLE: Find all individuals
  - who have been convicted of attempted murder in North America and
  - who have recently had electronic fund transfers made into their bank accounts from ABC Corp.
- Answering this query is problematic because:
  - Determining all people convicted of different crimes may require accessing a wide variety of databases belonging to different police jurisdictions and courts
  - ABC Corp. may have accounts in hundreds of banks worldwide each of which uses different formats and different database systems
Heterogeneous Multimedia Query

- Find all individuals who have been photographed with Jose Orojuelo and who have been convicted of attempted murder in North America and who have recently had electronic fund transfers made into their bank accounts from ABC Corp.

- This query requires that:
  - We find all people satisfying the conditions in the “Simple” Heterogeneous Query and
  - We access a mugshot-database, `mugshotdb`, containing the names and pictures of various individuals
  - We access a surveillance photograph database of still images
  - We access a surveillance video database to see if a meeting between the suspect and Jose Orojuelo was recorded on video
  - We access image processing algorithms to determine who occurs in which video/still photograph.
Let us suppose (for now) that we already have a pre-existing body of multimedia data that we wish to access.

- Need a single language within which multimedia data of different types can be accessed.

- In classical database theory, there are binary operations that combine different relations in different ways (e.g. join, union, difference, intersection, Cartesian Product). In the same way, this language must be able to specify combination operations across different media types (rather than just across different relations).

- This language must be able to access:
  - “metadata” describing the content of different media sources and
  - “raw” data supported by the different media sources.

- This language must be able to merge, manipulate and “join” together, results from different media sources.

- Once such languages are devised, we need techniques to:
  - optimize a single query (through the notion of a query plan
  - develop servers that can optimize processing of a set of queries.
Multimedia Research Issues: Content

- What is content of a media source? Under what conditions can content be described textually and under what conditions must it be described directly through the original media type?

- How should we extract the content of:
  - an image
  - a video-clip
  - an audio-clip
  - a free/structured text document.

- How should we index the results of this extracted content?

- What is retrieval by similarity?

- What algorithms can be used to efficiently retrieve media data on the basis of similarity?

- If one has to design a multimedia database using legacy media sources as well as to-be-organized media data, what is the best way of creating such a database?

- How should queries be relaxed so that not only the originally stated query, but also “similar” queries get processed?

- What are efficient algorithms for processing such queries?
Multimedia Research Issues: Storage

- How do the following (standard) storage devices work?
  - disk systems
  - CD-ROM systems
  - tape systems and tape libraries

- How is data laid out on such devices?

- How do we design disk/CD-ROM/tape servers so as to optimally satisfy different clients concurrently, when these clients execute the following operations
  - playback
  - rewind
  - fast forward
  - pause