Modeling Temporal Aspects of Visual and Textual Objects in Multimedia Databases

Carlo Combi

Laboratory of Artificial Intelligence
Department of Mathematics and Computer Science
University of Udine

combi@dimi.uniud.it
Talk Overview

- Introduction;

- Motivation;

- The Multimedia Temporal Data Model
  - basic concepts;
  - composing visual data;
  - the temporal dimension of visual data;
  - integrating visual and textual data;
  - temporal aspects of observations.

- Final Outlines.
Introduction

The integrated management of video, audio, and textual data is a need for several application domains:

- geographical information systems;
- medical information systems;
- video/movie archive systems.
Introduction

Modeling multimedia information at conceptual and logical levels:

- *Composition of visual data;*

- *Temporal dimension of visual data;*

- *Multimedia data as integration of visual and textual data;*

- *Temporal aspects of textual observations related to visual data.*
A Motivating Application Domain

- Cardiac angiography is a technique adopted to study the situation of coronary vessels (coronary angiography) and of heart functionalities (left ventriculography).

- The result of a cardiac angiography consists of an X-ray movie.

- Diagnoses based on the content of the movie consist of identifying stenoses (i.e., reductions of vessel lumen) and problems in the movement (contraction/relaxation) of the heart.
A Motivating Application Domain

The stored movies can be used in many different ways:

- the physicians could be interested in composing a video, where different movies collected on the same patient can be viewed in sequence, to control the patient’s state evolution;

- it could be useful to compose other videos, on the basis of movies from several patients, showing, for didactic/research reasons, different approaches/results in a given patients population.
The Temporal Object-Oriented Data Model

GCH-OODM (Granular Clinical History - Object- Oriented Data Model) is an object-oriented data model extended to consider and manage the valid time of information.

Basic concepts

- **class** (type) and **object**: a database schema consists of a set of classes; objects are created as instances of a class.
  - state (attributes)
  - interface (methods)

- **object identity**; **abstract data types**; **single inheritance**; **polymorphism**; management of complex objects; **persistence**
GCH-OODM classes

• the usual types (char, string, int, real);

• the usual type constructors (array, list, set, ...);

• the class hierarchy composed by classes el_time, instant, duration, and interval

  – el_time (elementary time) allows us to model a chronon;

  – instant allows us to represent a time point, identified by the granule, i.e. a set of contiguous chronons, containing it;

  – duration allows us to model a generic duration, specified at arbitrary granularity;

  – interval models a generic interval, i.e. a set of contiguous time points. Notation: FROM <instant> TO <instant> FOR <duration>.
The Temporal Object-Oriented Data Model

- GCH-OODM relies on a three-valued logic modeled by the class `bool3`, allowing the management of the uncertainty coming from comparisons between temporal dimensions expressed with different granularities.

- Classes modeling objects having a temporal dimension inherit from the class `Temporal_Object`: the method `valid_interval()` defined for this class, returns an object of the class `interval`, thus allowing one to consider the valid time for that object.
Composing Visual Data

In our multimedia data model, we define three abstraction layers for video data:

- the *physical layer*, where we model the data sequence (stream) coming from an acquisition device;

- the *logical layer*, where we are able to identify meaningful frame sequences into the raw stream;

- the *compositional layer*, where we can associate frame sequences from different streams, to compose videos.
Composing Visual Data

- the class VideoStream allows one to store video data, while the class Image allows the storage of static images;

- VideoClip and VideoFrame allow the user to identify suitable subparts into a video stream and to refer to it;

- the class Video allows one to create different videos by composing image sequences from different video streams.
Composing Visual Data

The Booch Class Diagram for Visual Classes
Composing Visual Data

class VideoClip: public Temporal_Object {
  VideoStream relatedVideoStream();
  FrameRange start_end_Frames();
  void play();
};
class Video: public Temporal_Object {
    list < VideoClip >  videocomposition();
    long totalFrames();
    ...
    ...
    void play();
};

Composing Visual Data
The Temporal Dimension of Visual Data

- **intrinsic time**: the time we can use to identify some frames inside the frame sequence, on the basis of their distance from the first frame of the sequence;

- **extrinsic time**: the usual valid time, possibly given at different granularities.
The Temporal Dimension of Visual Data

$v = [c_1, \ldots, c_n]$ is an object of the class Video;
$c_1, \ldots, c_n$ are objects of the class VideoClip;
$I = \{i_1, \ldots, i_n\}$ is the set of valid times of objects $c_1, \ldots, c_n$.

The valid time of $v$ is

$$vt_v \equiv (vt_v.\text{start}(), vt_v.\text{end}(), vt_v.\text{dur}()), \text{ where:}$$

\[
\begin{align*}
vt_v.\text{start}().\text{inf}() &= \min(i_j.\text{start}().\text{inf}()), \ i_j \in I \\
vt_v.\text{start}().\text{sup}() &= \min(i_j.\text{start}().\text{sup}()), \ i_j \in I \\
vt_v.\text{end}().\text{inf}() &= \max(i_j.\text{end}().\text{inf}()), \ i_j \in I \\
vt_v.\text{end}().\text{sup}() &= \max(i_j.\text{end}().\text{sup}()), \ i_j \in I \\
vt_v.\text{dur}().\text{inf}() &= \max(\max(i_j.\text{dur}().\text{inf}()), \\
& \quad (vt_v.\text{end}().\text{inf}() - vt_v.\text{start}().\text{sup}()), \ i_j \in I \\
vt_v.\text{dur}().\text{sup}() &= \max(\max(i_j.\text{dur}().\text{sup}()), \\
& \quad (vt_v.\text{end}().\text{sup}() - vt_v.\text{start}().\text{inf}()), \ i_j \in I
\end{align*}
\]
The Temporal Dimension of Visual Data

Example

frame rate = 30 fps

c1.relatedVideoStream() returns the videostream vs1

c1.start_end_Frames() returns [21, 147]

vs1.valid_interval() returns FROM 98Jul18/13 FOR 20 ss

c3.relatedVideoStream() returns the videostream vs3

c3.start_end_Frames() returns [1, 41]

vs3.valid_interval() returns FOR 15 ss TO 98Jul18/13:15
The Temporal Dimension of Visual Data

Example

```
c1.valid_interval() returns
   FROM <98Jul18/13:00:0.66, 98Jul18/14:00:0.65>
   FOR 4.23 ss

   c3.valid_interval() returns
   FOR 1.36 ss
   T0 <98Jul18/13:14:46.37, 98Jul18/13:15:46.36>

   v2.valid_interval() returns
   FROM <98Jul18/13:00:0.63, 98Jul18/13:15:44.99>
   T0 <98Jul18/13:14:46.37, 98Jul18/14:00:4.88>
   FOR <4.23 ss, 1 hh 0 min 4.22 ss>
```
Integrating Visual and Textual Data: Observations

An observation is any kind of textual information related to a visual object.

class Observation : public Temporal_Object {
  string description();
  set < Object > relatedObjects();
  Role role( Object );
  array < Object, Role > obsObjRoles();
  array<Video, set<FrameRange>> framesVideo();
  set < Image > relatedImages();
  set < interval > nonConvexInterval();
};

class Video : public Temporal_Object {
  list < VideoClip > videocomposition();
  long totalFrames();
  ..... 
  ..... 
  ..... 
  void play();
  Tree obsTree();
};
Integrating Visual and Textual Data: Observations

Example

\[\text{video v1} \quad \text{video v2}\]

\[
\begin{array}{c|cccc}
\text{videoclip c1} & 1 & 11 & 100 & 127 & 128 \\
\text{videoclip c2} & 154 & 182 & 232 \\
\end{array}
\]

\[\text{stenosis-presence} \]

\[
\begin{array}{c|cccc}
\text{videoclip c1} & 1 & 11 & 100 & 127 & 128 \\
\text{videoclip c3} & 169 \\
\end{array}
\]

\[\text{stenosis-presence framesVideo()} \text{ returns} \]

\[(v1, \{[11, 128], [154, 182]\}) \text{ and} \]

\[(v2, \{[11, 127]\})\]
Temporal Aspects of Observations

*Dynamic vs. static properties of a video subsequence.*

[Diagram showing video clips and time frames]
Temporal Aspects of Observations

Types of observation in a multimedia database

- **concatenable observations**;
  if a concatenable observation is valid on the consecutive frame intervals \([i, k]\) and \([k + 1, j]\), it is valid on the frame interval \([i, j]\).

  *Example*: “perfusion of the contrast agent through the coronary vessels”.

- **point-upward observations**;
  if a point-upward observation is valid on the consecutive frame intervals \([i, i]\) and \([i + 1, i + 1]\), \([i + 1, i + 1]\), \ldots, \([i + n, i + n]\)
  (i.e., on \(n\) consecutive frames), it is valid on the frame interval \([i, i + n]\).

  *Example*: 


Temporal Aspects of Observations

Types of observation in a multimedia database

- **weakly-upward-hereditary observations**;
  given a set of \( n \) (possibly intersecting) frame intervals \([i_s, i_e]\) over which a weakly-upward-hereditary observation holds, the observation holds also on the intervals obtained as union of the \( n \) frame intervals \([i_s, i_e]\).
  
  Example: “perfusion of the contrast agent through the coronary vessels” is weakly-upward too.

- **downward-hereditary observations**;
  a downward hereditary observation holding on a frame interval \([i_s, i_e]\) holds on any frame interval \([j_s, j_e]\), where \(i_s \leq j_s \land j_e \leq i_e\).
  
  Example: “the contrast agent highlights less than half of the left coronary tree”.
Temporal Aspects of Observations

Types of observation in a multimedia database

- **liquid observations;**
  those observations which are both downward and point-upward hereditary, are termed as liquid.
  
  Example: “presence of a stenosis”.

- **solid observations;**
  a solid observation holding on a frame interval \([i_s, i_e]\)
  cannot hold on any frame interval \([j_s, j_e]\), for which \((i_s \leq j_s \land j_s \leq i_e) \lor (i_s \leq j_e \land j_e \leq i_e)\).
  
  Example: “exactly a cardiac cycle, from the systole (empting phase) to the diastole (filling phase)”.

23
Temporal Aspects of Observations

Types of observation in a multimedia database

- **gestalt observations**;
  a gestalt observation holding on a frame interval \([i_s, i_e]\)
  cannot hold on any frame interval \([j_s, j_e]\), for which \((i_s \leq j_s \land j_e \leq i_e) \lor (j_s \leq i_s \land i_e \leq j_e)\).
  
  Example: “two cardiac cycles”.

- **disjointed observations**;
  if a disjointed observation is associated to a frame interval \([i_s, i_e]\), it cannot be associated to any interval \([j_s, j_e]\)
  such that: \(j_s \leq i_s \leq i_e \leq j_e \lor i_s \leq j_s \leq i_e \leq j_e\).

  Example:
Temporal Aspects of Observations

class Observation: public Temporal_Object {
    string description();
    ........;
    ........;
    bool3 is_in(Video, FrameRange);
    bool3 is_valid(Video, FrameRange);
};

![Diagram showing temporal aspects of observations with video clips and frame ranges]

stenosis-presence.is_in(v1, [124,128]) returns True
stenosis-presence.is_valid(v2, [124,128]) returns True

two-cardiac-cycles.is_in(v1, [100,110]) returns True
two-cardiac-cycles.is_valid(v1, [100,110]) returns False
two-cardiac-cycles.is_valid(v1, [100,154]) returns True
Temporal Aspects of Observations

\( o \) is an object of the class `Observation`;
\( I = \{i_1, i_2, \ldots, i_n\} \) the set of time intervals related to \( o \) by the associations between \( o \) and several frame intervals of different videos.

The **valid time** of \( o \) is
\[
vt_o \equiv (vt_o.\text{start}(), vt_o.\text{end}(), vt_o.\text{dur}()), \text{ where:}
\]
\[
\begin{align*}
vt_o.\text{start}().\text{inf}() &= \min(i_j.\text{start}().\text{inf}()), \ i_j \in I \\
vt_o.\text{start}().\text{sup}() &= \min(i_j.\text{start}().\text{sup}()), \ i_j \in I \\
vt_o.\text{end}().\text{inf}() &= \max(i_j.\text{end}().\text{inf}()), \ i_j \in I \\
vt_o.\text{end}().\text{sup}() &= \max(i_j.\text{end}().\text{sup}()), \ i_j \in I \\
vt_o.\text{dur}().\text{inf}() &= \max(\max(i_j.\text{dur}().\text{inf})), \\
(\text{vt}_o.\text{end}().\text{inf}() - vt_o.\text{start}().\text{sup}()), \ i_j \in I \\
vt_o.\text{dur}().\text{sup}() &= \max(\max(i_j.\text{dur}().\text{sup})), \\
(\text{vt}_o.\text{end}().\text{sup}() - vt_o.\text{start}().\text{inf}()), \ i_j \in I
\end{align*}
\]
Temporal Aspects of Observations

Example

Intervals related to \textit{stenosis-presence}:

FROM \textit{<98Jul18/13:00:0.99, 98Jul18/14:00:0.98>}
FOR 3.9ss (from clip c1 in videos v1 and v2);
FOR 0.03 ss (from clip c2 in video v1);
FOR 0.96 ss (from clip c2 in video v1).

\textit{stenosis-presence.valid_interval()} returns
FROM \textit{<98Jul18/13:00:0.99, 98Jul18/14:00:0.98>}
TO \textit{<98Aug10/15:30:3.45, 98Aug10/15:31:3.44>
Temporal Aspects of Observations

Given the set \( I = \{i_1, i_2, \ldots, i_n\} \) of intervals related to the frame interval \([j, k]\) of a video, given the set \( \{vt_1, vt_2, \ldots, vt_m\} \) of valid times of \( m \) temporal objects involved in the considered observation, we can associate the observation to the frame interval \([j, k]\) of the video, only if

\[
\forall vt_z \ (z = 1, \ldots, m) \forall i_w \ (w = 1, \ldots, n) \\ i_w.DURING(vt_z) \text{ returns True}
\]

Example

\( stenosis\text{-}presence\).relatedObjects() \text{ returns} \{sten1\}

\( stenosis\text{-}presence\).valid_interval()

    \text{ during (sten1.valid_interval())}

must return True.
Final Outlines

- *Composition of temporal visual data.*
  
  - three-layer approach to compose videos;
  
  - valid time of visual objects at different granularities and/or with indeterminacy.

- *Integration of temporal visual and textual data.*
  
  - taxonomy for observations based on their temporal features;
  
  - valid time of observations and constraints with valid times of other involved database objects.
REFERENCES


