Collaborative Multimedia Authoring: Scenarios and Consistency Maintenance

Bo Xiao
Integrated Publication and Information Systems
Fraunhofer Institute, Germany
xiaobo@ipsi.fhg.de
Motivation

- Research in collaborative editing
  - Text, spreadsheet, graphics, XML, bitmap, etc.
  - Distributed discreet media presentation
  - What about editing time-based imedia collaboratively?

- Research in multimedia
  - Distributed multimedia
  - Interactive multimedia

- Collaborative multimedia
  - Interesting potential applications in areas such as media production, education, entertainment, business, and so on
Scenario for Collaborative Multimedia

- A team makes a movie cooperatively in a distributed setting.
- Steps including pre-production, production, post-production.
- The team might work back and forth on production and post-production tasks.
- Tasks: to outline, script, capture, import, assemble, edit, composite and output
Collaborative Multimedia Authoring

- CMA (Collaborative Multimedia Authoring)
  - In a broad sense: the process for the preparation and consumption of digital media by a team for a defined goal
  - In a narrow sense: editing time-based media in the temporal and the spatial domains

- Multimedia authoring mode (temporal)
  - Time-based mode
  - Relationship-based mode
  - Mixed mode
CMA System Model

- Three-layered model
  - Collaborative Authoring UI Layer
    - User view of a CMA environment
    - The track metaphor is used here.
  - Collaborative Media Interaction Layer
    - Logical model of time-dependent media
  - Collaborative Media Data Layer
    - Physical data: format, protocol, etc
Model for CMA

- Data model
  - A multi-track, temporal structured model
  - Tracks, media entities
  - Segments (time intervals), time points
  - Media entities do not overlap along the time axis.

- Operation model
  - Primitive operations: insert and delete
  - Composite operations: move, cut, copy, paste, etc
The “insert” operation

- Semantics: inserting some media entities into a track at a time point
- Effect:
  - New media entities are added to the set of media entities.
  - If new entities overlap existing entities, some of the existing entities will be “pushed” right.
  - In no circumstances will any entity be “pushed” left.
The “delete” operation

- Semantics: deleting media entities from a track.
- Effect:
  - The media entities are removed from the set of media entities.
  - After the deletion, there will be “empty segments” where the deleted entities once were.

Compatibility

- The definition of “insert” and “delete” here are basically compatible with the counterparts in practical video editing tools, such as Adobe Premiere.
Inconsistency Problem

- In CMA, there can be inconsistency problems. As an example,
  - $O_1 = \text{insert}(E_1, T, P_1)$: insert a media entity $E_1$ to the track $T$ at the timepoint of $P_1$
  - $O_2 = \text{delete}(E_2, T)$: delete a media entity $E_2$ from the track $T$
  - There can be inconsistent results after concurrent performing of the operations
Inconsistency Problem (Cont.)

- Illustration of the example

initial state

E_1 to be inserted

state after O_2 + O_1

state after O_2 + O_1
Approach

- Applying operational transformation
  - The generic operational transformation algorithms will be used.
  - Transformation functions for specific domain semantics are needed.

- Solving inconsistency problems in CMA
  - Identifying temporal relationships which is necessary
  - Designing transformation functions for the primitive operations
  - Applying the operational transformation method
Part of The Solution

- **Temporal relationships**
  - (a) before, (b) before with intersection, (c) containing,
  - (d) contained by, (e) after with intersection, (f) after

- **Transformation function for the “insert” operation**

\[
O_1' = IT(O_1, O_2) \quad (O_1: \text{insert}, O_2: \text{delete})
\]

<table>
<thead>
<tr>
<th>Relationship between $E_1$ and $E_2$</th>
<th>Result of $O_1'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a), (f)</td>
<td>$O_1$</td>
</tr>
<tr>
<td>(d), (e)</td>
<td>$\text{insert}(E_1, T, P(E_2) + L(E_2))$</td>
</tr>
<tr>
<td>(b), (c)</td>
<td>$O_1$, if $E_2$ is the leftmost one under (b) and (c) $O_1 + \text{insert}($EMPTY SEG$)$, otherwise</td>
</tr>
</tbody>
</table>
Comparison

- With graphic editing
  - Similarity: a few basic graphic editing functions
  - Difference: constrained in the tracks, and no overlapping

- With text editing
  - Similarity: “insert” and “delete” as primitive operations
  - Difference: “empty segments” are different from spaces
Future Work

- Concurrency control in CMA systems
- Low-level media issues
- Spatial aspect of CMA
- Undo/redo in CMA
Thank you very much!