Ontologically and Graphically Assisted Media Authoring with Multiple Media Types

Insook Choi

Entertainment Technology and Emerging Media, New York City College of Technology
City University of New York
300 Jay Street, V411, Brooklyn, NY 11201
ichoi@CityTech.CUNY.edu

Abstract

A prototype system for authoring interactive cross-media presentations is supported by an ontological inference engine and a scalable GUI. This prototype provides a context to explore the development of a scalable reference structure for fine calibration of multimedia information extraction (MMIE) and ontological query. Ontologies are structured as graphs of concept nodes with edges for subclass-superclass relationships and edges for asserted non-taxonomic relationships. Authoring is implemented as path-planning in ontological space; path members are concept nodes that generate queries and return media resources coupled to real-time displays. Real-time feedback enables users to fine tune their resource exploration and to annotate the paths. Ontological organization enables users to author and explore media resources by concept-based navigation rather than by resource type; exploration and path-authoring is facilitated by a graphical interface representing concept graph space. We assert this approach entails two merits: 1. query induced fluid cross-media display replaces the representation of individual media types arranged in time-linear track-like formats derived from conventions of non-digital media; 2. concept formation takes a significant role in authoring processes and media resources explorations.

Introduction: Authoring and multiple media types

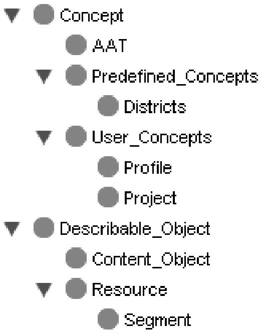
Digital media technology has been progressing steadily to support interactive and procedurally-driven processes. The final configuration of moment-by-moment content can be assembled with respect to user's actions and often at the point of delivery, on the user's device. Dominant practice remains in broadcast-related formats – a completed program is streamed or downloaded to a user's pod. However emerging trends and concomitant ISO standards, for example MPEG-4 support the delivery of programs that are not completely pre-packaged, enabling user preferences and interests to impact the final content (Koenen 2002). The tasks of authoring in this context will involve a process of pre-configuring combinations of media resources and procedural conditions for achieving locally-deliverable and customizable program contents.

The prototype system under development presents a dual capacity for *authoring of interactive media* and for *interactive authoring*. To disambiguate, the former refers to a production of media configurations capable of responding to a user's interactive input signals for processing media resources; the latter refers to an authoring process supported by real-time feedback with media display during the authoring process. It is our objective to support both capacities in close proximity with proper system architecture and performance. Currently-supported media resources include two-dimensional graphic documents such as photographs, diagrams and architectural plans, pre-recorded and procedurally-generated sounds, 3D graphical models, and camera movements in a 3D graphical scene.

As multiple media types and resources have to be flexibly configurable in real time and in dynamic environments we anticipate there will be greater needs for efficient multimedia information extraction and feature recognition techniques, as well as advanced classification criteria. The classification optimally accounts for both semantic and qualitative aspects of a media resource, and the ability to identify semantic and qualitative relationships across media resources of unlike types. Metadata stored with individual media resources is often inadequate to meet these criteria, as the metadata describes attributes related to a media production device. For example "lens aperture 4.2" or "microphone type cardiod": when presenting a recorded sound with a still or moving image, it may be desirable to combine an intimate recorded ambience with a close-up field of view. Image metadata will provide aperture information and audio metadata will provide microphone information; but metadata does not provide a structure to define relationships on data of unlike types, comparing aperture settings with microphone patterns. Such evaluations are computationally supportable using ontological data.

Proposed Relationship: Parallel structure of Ontology and MMIE

Symmetries may be identified between MMIE and ontologically-defined search; we propose cross-development of analysis techniques and MMIE search terms for integration into authoring systems. We describe computational implementation of semantic structure in the form of modular and combinatorial ontological queries that can represent complex scenes and complex relationships across sets of media resources. The parallel organization and fine-tuning of ontologies and information extraction templates will reduce search space for feature recognition. Many MMIE approaches leverage significant analysis of media content according to program type and develop working assumptions about extraction context; ontologies that closely couple search and display functions may provide an alternative means to formally define and quantify these sets of assumptions supporting extraction context, by working directly with the authoring process. Two benefits are envisioned, the automated identification and integration of media resources into structured ontological data sets, and the close fitting of MMIE performance with respect to interactive media production needs and goals.



- Concept
  - AAT
  - Predefined_Concepts
    - Districts
  - User_Concepts
    - Profile
    - Project
- Describable_Object
  - Content_Object
  - Resource
    - Segment

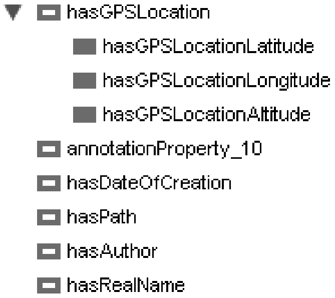
Figure 1: Root node OWL classes. Concept and Describable_Object are parallel root nodes

Design of relationships between ontological data and metadata

Concepts describe media resources of multiple types, and *Properties* denote media resource metadata. Figure 1 summarizes the main components in the ontology, a dual-root node structure of Concepts and Resources, with Describable_Objects serving as the root node for Resources (discussed below).

Properties and Metadata

A property identifies data or relationships specific to an individual resource, such as metadata related to a specific media type. For example the property *hasPhotoSpecific* includes lens focal length metadata and the property *has3DModelSpecific* includes polygon count metadata. Properties may also indentify metadata common to multiple types of media resources, as in Figure 2.



- hasGPSLocation
  - hasGPSLocationLatitude
  - hasGPSLocationLongitude
  - hasGPSLocationAltitude
- annotationProperty_10
- hasDateOfCreation
- hasPath
- hasAuthor
- hasRealName

Figure 2: Examples of properties that indicate metadata shared by multiple types of media resources

Properties denote quantitative or semantic metadata that can be applied to interactive display processing. Whereas ontologies describe resources in unordered sets, resources may ordered by analyzing metadata values assigned to their properties.

Concepts

The core concept tree used in our prototype is the Getty Art and Architectural Thesaurus [Getty 2005]. In addition to the AAT, *Predefined_concepts* are created by the system designers to enhance the AAT with respect to specific projects. This enables the design of complex logical relationships that are not easily represented in a graphical interface designed for novice users. *User-_Concepts* are concepts created by a user while navigating concepts and exploring media resources, enabling a user to focus on and customize queries, as well as to generate results that meet her needs. User-defined concepts are created on-the-fly by selecting previously available concepts and applying operations such as unions, intersections, and filters on metadata properties and stored values. Examples of metadata property values that can be used as criteria for filters in user-defined concepts include dates, GPS locations, polygon counts, and focal length settings.

Resources

Media resources are digital objects stored in a data set using the Concepts and Properties ontology. Resource are described in two ways, by what they semantically represent and by their digitized material structure – based upon the type of resource they are. For example a photograph can be described by what it depicts and by the structural properties related to how the photograph was made (focal length, aperture, resolution in pixel ratio, etc.). Additional properties may be created to denote contextual metadata useful to MMIE, such as date, time, or conditions of capture of individual resources. Concepts may also be created to represent these contexts on an ontological (non-resource specific) level. These designs should be developed in coordination with designs of MMIE applications.

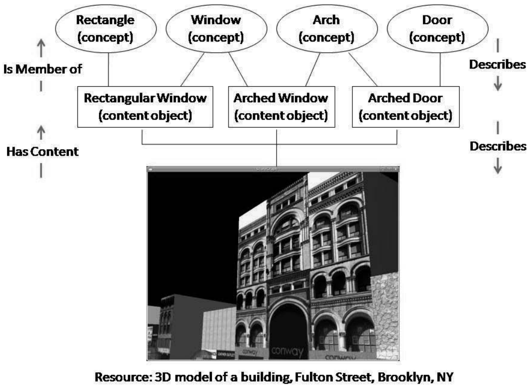

Rectangle (concept)
Window (concept)
Arch (concept)
Door (concept)
Is Member of
Describes
Rectangular Window (content object)
Arched Window (content object)
Arched Door (content object)
Has Content
Describes
Resource: 3D model of a building, Fulton Street, Brooklyn, NY

Figure 3: Content_Object supports modular semantic representations of media resource contents.

Concept membership is asserted for a resource to describe a general property of that resource. To distinguish concepts in general from identities of specific describable objects we created *Content Object* (Figure 3), a level of indirection that enables parallel description of multiple identities in a single resource. A resource may depict more than one content object, and multiple resources may depict the same content object. Figure 3 provides a tutorial example: a 3D model of a Brooklyn street with a historical building; arched windows and rectangular windows are represented as separate content objects that combine membership in the concepts “arch” or “rectangle” with “window.” This structure may be thought of as “Resource—depicts—Content Object” and “Content Object—is member of—Concept.”

The content objects in the above example could also be expressed as intersections of concepts, given that windows are relatively general objects. In practice we use content objects to provide efficient and precise expression of unique types of objects including real-world identities that may be depicted in multiple resources: the Brooklyn Bridge, Walt Whitman, Cadman Plaza are examples. We envision the possibility of coupling content objects to MMIE templates for specific objects or object classes. Coupling may help reduce search space for recognition of object types or known real-world objects that appear in multiple resources under variations of lighting, camera angle, ambient noise, occlusion and other irregularities.

Structured Indirection

The Content_Object structure provides an important level of indirection between Concepts and Resources. This indirection supports the dual-root node structure (Concept; Describable_Object) seen in Figure 1. The simplest way to state this indirection is that resources are not children of concepts, and concepts are not children of resources, yet resources and concepts are connected through well-defined classes of relationships. These relationships are inferred as well as asserted, enabling queries applied to concepts to return resources, and resources to function as pivots between concepts.

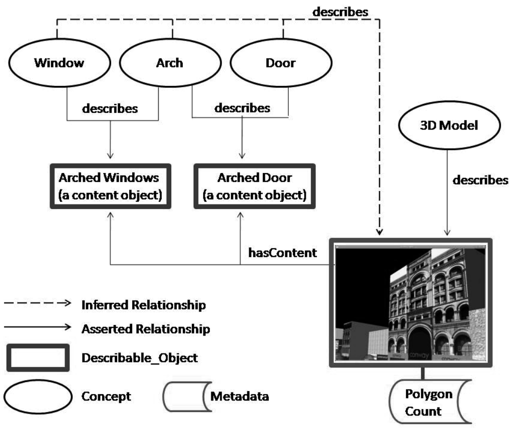


Figure 4: Asserted and inferred relationships of concepts, content objects, resources and metadata.

By avoiding direct dependencies between concepts and resources, changes can be applied to the concept graph without requiring modifications to resources already stored in the system, and resources may be added without modifications to the concept graph. Figure 4 shows content objects in a pivotal role as descendants of both concepts and resources. The initial representation of a resource or concept is created by assertion, whereas inference determines most relationships returned by a query. Inference is reveals valid relationships in the system that may not be known in advance by the user or designer.

An interface for navigating relationships of multiple media types

Ontological structure is defined as a set of logical expressions, unions and intersections of sets, which we interpret as a graph structure, then apply the structure to generate a display for a graphical user interface (GUI). Figure 5 shows the GUI as a collection of dynamic nodes; the display represents a limited region of a much larger ontology. Several levels of interaction are defined. Mousing-over a node displays its concept name. Double-clicking on a node selects that node to generate a resource query and modifies the display to reveal all nodes that are nearest neighbors to the queried node.

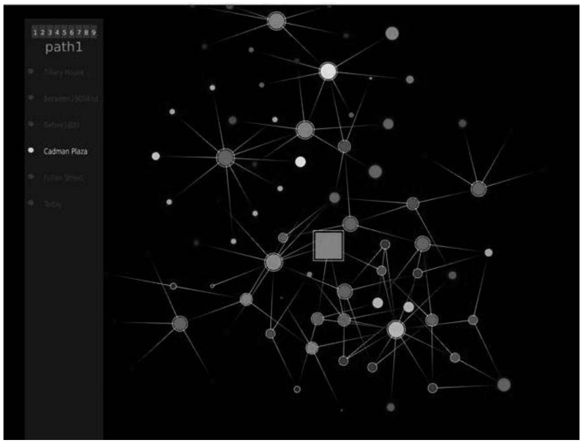


Figure 5: Graphical User Interface representation of concepts as interactive nodes. Color indicates subclass-superclass or non-taxonomic relationship. Size indicates number of links to a node.

Nodes remain hidden until a nearest neighbor is selected. Nodes are displayed with animated ball-and-spring dynamics, aiding visual identification of relationships. The square node is anchored and can only be moved by direct mouse dragging; it represents a concept that is part of an annotated *query path*, a starting point for concept exploration designated by authoring (discussed below). The "current location of the user" is defined as the most recently selected node generating the most recent query. Various criteria are applied to hide nodes that become remote from the current query location.

Path-planning and Interactive Authoring

The idea of making paths through a digital document space can be traced to multiple sources, including the Memex technology proposed by Vannevar Bush in the late 1940's (Bush 1945). These proposals focus on "trails of documents" using text processing for cross referencing and indexing to achieve more efficient storage and retrieval. Media asset management systems are beginning to adopt these approaches; however computational path-planning has not been adopted for interactive media production.

Our prototype introduces path-planning techniques for interactive media production. Path-planning is computationally robust for combining interactive media and pre-structured media. A path-planning model can accommodate user exploration and improvisation while maintaining linear structure as priorities. Our prototype differs from "trails of documents" proposals by implementing *paths of queries;* paths through concept space generating queries as a method of applying order to a diverse collection of resources. Queries generate sets of related resources; queries organized in paths may be designed to generate linear structures by making a series of selections from media resources that can function in multiple semantic contexts.

In Figure 5 the vertical array on the left of the GUI is a sequence of concept nodes that have been arranged as a path. The path can be traversed in top-to-bottom order or in any other order. Each path member displays a square anchor node and a neighborhood that can be expanded for exploration. Selecting a path member generates a query that returns a set of media resources and visualizes the concept neighborhood; exploration of neighboring nodes produces further queries returning related resources.

An Interactive Media Use Case Example

A system is configured to respond to queries by scheduling the display of 2D images, sounds and virtual camera movements in a 3D environment. The sounds and images have been entered as resources in the ontological data set; the camera movements are determined by positional data of 3D models that also have been entered as resources. When a query returns one or more 2D images, sounds or 3D objects, separate media display engines receive the addresses of these resources and schedule their display using a sound synthesis system and two image projections, one showing arrays of 2D images and the other showing a 3D scene. In an example query path the first node is the concept "FultonStreet2000toPresent"; it returns photographs of storefronts, sounds of bus traffic, pedestrians and street vendors captured 2006-08 on Fulton Street, and a 3D camera movement slowly "flying" (tracking) along virtual Fulton street. Selecting a second path node while these resources are displayed, "BoroHall2000toPresent" introduces new photos and sounds, with smooth visual and audio cross-fades effecting the transition. The 3D camera movement interpolates from Fulton Street to a new position hovering above the model of Borough Hall. Some images and sounds are returned by both queries; these persist in the display across the transition. "FultonStreet1880to1920," the third path node sends the 3D camera to resume a flyover of Fulton Street; the 3D scene now includes a model of the Brooklyn elevated train from that period, and contemporary buildings are absent. Photographs of hip-hop shops and cell phone vendors are replaced by historical drawings, lithographs, and photos including images of the elevated train that were used as references for modeling its 3D graphics counterpart. Sounds captured on Fulton Street are replaced by sounds from a SFX library: horses, carriages, a steam engine, and pedestrians on a boardwalk.

Transitions in each media display are computed to dynamically arrange image arrays and sound mixes and smoothly interpolate virtual camera movements in the 3D scene. When a user generates a query at the GUI, transitions are effected immediately to provide feedback to the user. Scheduling constraints impose minimum duration between queries. We introduce *Display Grammar* as a term for the configuration of rules for display signal processing and scheduling of multiple resources.

Concluding Remarks

Automated feature and resource classification will be highly beneficial to ontological systems for both their creation and use. MMIE is context dependent, determined on models such as user intent, media capture and production. These use contexts are strongly semantic, and ontology serves well to represent relationships extending beyond metadata across multiple-resource structures. Ontology also supports robust classification of semantic relationships underlying capture and use contexts. Query paths can represent dynamic context as membership of semantic sets changing over time; paths provide formalization of temporal context that could support segmentation for time-based MMIE. Additionally, ontologies are robust across multiple resource types, and could support data models for cross-modal MMIE.

We hypothesize content objects may serve as suitable units for parallel organization with MMIE templates. Content objects function as “semantic anchors” to substantiate concepts with real-world identities. Content objects may be shared across resources of different types—for example a Describable Object may have both visual and auditory features; and content objects are able to denote multimodal markers to support cross-modal MMIE such as discussed in (Xiong et. al.) . Properties associated to resources can indentify unique context data regarding capture or intended use, and could be indexed to signal processing templates for use in MME.

Potential Semantic Calibration with MMIE

A possible approach to “tuning” semantic structure and MMIE might be as follows: (1) using a query path generate a stream of ontologically-inferred media resources, (2) apply MMIE to identify features in the resulting streams, (3) compare the identified features with query concepts that generated the media stream, and apply feedback through iteration to align MMIE-identified features with semantic units—concepts and content objects—that identify the media resources that depict those features. As queries are well-structured and computationally robust it should be possible to arrive at parallel unit representations of semantic and feature data. Resulting MMIE templates would be useful for reliably updating and extending a set of media resources relevant to a defined semantics, and could also be used to maintain ontologies in sync with relevant cases of real-world media resources.

Acknowledgements

Hardware systems and software development support for this project were provided by the School of Technology and Design, New York City College of Technology, CUNY. Robin Bargar contributed software designs and supervised code development. Thanks to Arthur Peters, research programmer for API and software design and development, and to Brian Macmillan and Marius Schebella, graduate student assistants, for their implementations.

References

Bush, Vannevar, 1945. As We May Think. *Atlantic Monthly* July 1945.

Getty Trust, J. Paul. 2005. *Art and Architecture Thesaurus (AAT).*

Koenen, Rob, ed. 2002. Overview of the MPEG-4 Standard. *International Organisation for Standardisation (ISO) JTC1/SC29/WG11 Coding of Moving Pictures and Audio*. March 2002.

Xiong, Z, Regunathan, R., Divakaran, A., Rui, Y. and Huang, T. 2006. *A Unified Framework for Video Summarization, Browsing and Retrieval*. Elsevier Academic Press, Burlington, MA.