Setareh Rafatirad
University of California Irvine
srafatir@uci.edu

Amarnath Gupta
University of California San Diego
gupta@sdsc.edu

Ramesh Jain
University of California Irvine
jain@ics.uci.edu

Composing Events

Abstract: Most real-world multimedia information are captured to record and commemorate events in our lives. It is therefore imperative that we should treat events to provide a primary form of contextual semantics for multimedia objects. In this paper, we briefly present a formal model of events and explore how they relate to media content. Then we propose an extensive mechanism to compose smaller events into larger ones, especially along the superevent-subevent relationships, and investigate how we can use composition operations on events can lead to automated construction of composite multimedia information. We conclude by presenting a real-world use case where these event-based operations can help one to organize their multimedia information and develop "multimedia summaries" of composite events they have experience.

Keywords: composite events, event data model, event composition operators, spatiotemporal composition

1. Introduction

Capturing multimedia information is a part of our daily lives. We take pictures and shoot videos whenever we want to preserve a memorable piece of the everyday happenings around us. Consider a typical scenario in Prof. Joe's life. Prof. Joe goes to a conference in India, a country he has never been to. He spends the first few days attending the conference. He really likes the hotel, and takes pictures of its architecture. He is very intrigued by the typical city life of Delhi, and he takes pictures of mosques, temples, bazaars, and the multicultural crowd as he walks about in the city. At the conference venue, he takes pictures with some of his old colleagues he has not met in a while, pictures of his students presenting their talk, pictures of the slides of some presentations he found interesting. During the same time, his students take pictures with him in front of their demo, pictures of him getting an award for his excellent research over the last decade, pictures of him giving an award speech, and so forth. After the conference, he goes to couple of sightseeing trips to nearby historical places and places of natural beauty. For this segment, he is joined by his wife, and they take pictures of the train rides, the monuments, artifacts at the museum, their enjoyable boat trip, and so on. When he returns, he finds that he has accumulated 300 pictures that were taken by him, his wife, his students, and other people of the conference. Pleased with his image collection, he wants to organize the images in a digital repository in a meaningful way so that he can use these organized collection to:

1. Send a photo album of a subset of these images to his daughter, and another to a professional friend.
2. Produce a multimedia diary of "interesting things I learnt at the conference" for his students.
3. Create a multimedia report on "My 2009 Delhi trip" to a travel website he contributes to.

Clearly, with today's technology, he could create a new directory in his file system and store the collection in a hierarchy of subdirectories based upon location and date, or by people he has met, or by semantic categories like "at the conference" "during a train ride from Delhi to Agra". But these categories would not bring out the full semantic context of the picture. For example, the same picture can be assigned two different contexts like "professional collaboration meeting" and "a scene at a street corner café" and creating a single hierarchical organization over all pictures will only partially capture the semantics of the images. He could also assign multiple "semantic tags" to each image and create tag clusters. However, semantic tags are strings, and grouping the images according to their tags is a non-trivial task. This leads us to investigate the following questions. *What is a concrete principle by which one can create a flexible organization of images to capture their contextual semantics and enable diverse user needs (like the three needs of Prof. Joe)? Is there a formal way to represent the image context and content, such that* ***one can transform the user needs to computational tasks over the collection of images****?*

It is important to recognize that this problem, although related, is distinct from well-researched multimedia problems like content based retrieval [23,35], context-based object recognition, automated semantic annotation of media [9], or identification of events in video objects. The problem of content based retrieval for instances, uses a combination of low-level features and high-level classification in determining the "content" of images. This task can be adequately aided by the use of what Mylonas [25] calls the "visual context", i.e., knowledge such as "a beach is very likely to be adjacent to a region of water". The problem of knowledge-based object recognition creates a model of objects based on its features and partonomic hierarchy and attempts to find regions of an image that closely match these object models. The problem of automated semantic annotation might is essentially a classification process that may be aided by a machine learning process, and may allow the use of context whenever disambiguation is needed. For instance, the recognition of a ball is further refined to that of a "basketball" if it is detected in the context of a basket. The problem of event detection in videos, relies on the spatiotemporal co-occurrence of a number of features, computed from different media streams (audio, image sequence, …), that collectively provide the evidence that a known recognizable events is taking place in a video segment. All of these methods essentially deal with analysis of a single image or a single video segment with the help of features computable from that image (or segment) and additional semantic knowledge that may be learnt or encoded in the system. In contrast, the problem that we refer to is about formalizing the relationships among the images, so that their collective semantics can be modeled through these relationships. This goal does not preclude us from using any content analysis, feature extraction and feature comparison techniques for individual images. However, our concern here is not the extraction of an individual image's semantics, but a common semantic theme that tie together all images of a coherent collection like that of Prof. Joe's Delhi trip.

The basic insight offered in this paper is that we can use a formal ***event-based model*** as the underlying unifying structure to express the semantic context and content of images; algorithmic manipulation of this event-based model can lead to organization of the media objects for any purpose that the user intends. We specifically show that for any user-level task events can be *algebraically selected and combined* to compose newer events and this can naturally lead to reorganization, aggregation, assimilation and synthesis of images necessary for these tasks. The intuition behind this argument comes from some basic observations about images and how they are captured by people.

a) Except for special cases like professional photographers taking pictures of still life, most ordinary pictures taken by people are to capture an event.
b) Typically, one picture depicts a single small (atomic) event [21] or the state of an object of interest during an event. Thus, a picture of Prof. Joe and his longtime friend Dr. Li represents that they met at the Delhi conference.
c) When people want to capture a larger event, they take multiple pictures – these pictures are usually captured in a temporal sequence, potentially for multiple perspectives (a long shot of Prof. Joe from the audience when he gets up to the podium, and a close-up of his side face as he gives the acceptance speech).
d) Often pictures of events are correlated in multiple, semantic ways. Thus, a picture Prof. Joe took of the standing train before it left can be correlated with that of a village his wife shot from the moving train because they are both part of a larger travel event undertaken by the couple.
e) The multimedia depiction of many events amounts to the aggregation or assimilation, or composition (detailed later in the paper) of images that are associated with "smaller" events that are parts of the larger event of interest. For example, the multimedia depiction of the event "trip to Agra" may be composed of pictures of the Taj Mahal, and the pictures of the journeys to and from Agra.

The primary contribution of this paper is to present comprehensive set of algebraic operators that serve to perform various kinds of event combinations and a set of event-to-media mapping operations that enable the construction of a media organization corresponding to the composed events. The rest of the paper is organized as follows. In Section 2, we briefly present the basic event data model and show how it relates to multimedia objects. In Section 3, we will explain the composition operators along multiple properties of events. Section 4 contains some use cases of the introduced operators, and finally we conclude with addressing some open research areas in connection to our presented work.

2. Modeling Events

2.1 The Basic Event Data Model

In our data model, an event corresponds to the ontological concept of an *occurrent* [4,12], something that takes place at a certain time point or interval. While it is not necessary for an event to have specific location of occurrence, typically events occur at one or more specifiable locations [1]. The *spatiotemporal extent* of an event is a subregion of the $\Re^2$ (or $\Re^3$) $\times$ $\Re$ where space is defined in two (or three) dimensions and time is defined as a continuous entity. An event is *atomic* if no other recognizable (sub)-event takes place within the spatiotemporal extent of this event [3]. Within this conceptual framework, our event data model is structured like a graph. For every application we represent the events and objects and their properties as a graph, which we call an *event graph*. The vertices of the graph represent typed entities like objects (e.g., a hotel) or events; the edges represent binary relationships between an event pair, and object pair, or across events and objects, or between entities and their literal properties. In the rest of this section, we will touch upon the primary features of our event data model – the details of the model is out of the scope of this paper and can be found in [26].

Node Types: The set $\mathbf{T}$ of node types is a vocabulary of terms that represent different type names. The minimal vocabulary is the set {object, event, media, literal} where a literal stands for scalar data types, a time-interval data type, and a family of spatial data types such as (latitude, longitude) pairs designating points, an open sequence of points designating paths, a closed sequence of points to designate spatial regions without holes. However, a typical node type vocabulary will have more application dependent type names. For example, for Prof. Joe's application, "person", "meeting" or "trip" might be added as new node types. Every node *n* of the event graph $\mathbf{G}$ is considered to be an *instance* of some node type $t \in \mathbf{T}$. Usually, adding a new node type term *t* is associated with a set of structural constraints, i.e., constraints that partially specify the neighborhood of a node that is an instance of *t*. We illustrate this later in the section.

Node Type Hierarchy: Terms in $\mathbf{T}$ can be organized in a partial order relationship $\leq$ that designates a *subtype relationship*. In the case of Prof. Joe's application, the partial order will possibly represent relationships like "person" <

"Object", "meeting" < "event", "boat trip" < "trip", "hotel" < "building". The data model allows a node type to have multiple parent types. Note that the node type hierarchy is a general structure that defines the type hierarchy over both objects and events.

Nodes: A node has a unique identifier, a type, and a label. The label of node is a term that usually provides a short human readable descriptor that states what the node stands for. The node (001837, speech, "Tom's talk") may represent an event node where the type of the event is "speech" and the label of the event is "Tom's talk". Similarly, (203382, point-location, (28.555878,77.190371)) is a literal node of the spatial type "point-location", whose coordinates are as specified above.

Edge Types: Like a node type vocabulary, the data model also allows a vocabulary **L** of edge types that designate different types of relationships.

Table 1. The fundamental, built-in edge types in our data model.

Edge Label	NodeType → NodeType	Description
subevent-of	Event → Event	Event B is a subevent-of event A if the spatiotemporal extent of B is contained in that of A and B is an integral component of A
has-some-location	Event → Space	Event A has-some-location S implies that A occurs *somewhere* within the spatial extent given by S
has-all-location	Event → Space	Event A has-all-location S implies that A occurs *everywhere* within the spatial extent given by S
occurs-some-time	Event → Time	Event A occurs-some-time T implies that A occurs *sometime during* the temporal extent given by interval T
occurs-all-time	Event → Time	Event A occurs-all-time T implies that A occurs *during the entire temporal extent* given by interval T
participates-in	Object → Event	Object O participates in event A
part-of	Object → Object	Object O2 is a part of object O1
segment-of	Media → Media	Media item M2 is a piece of the media item M2. For example, M2 can be a portion of an image or a video.
witnesses	Media → Event	Media item M1 "shows" the occurrence of event E
depicted-in	Object → Media	Object O is depicted in appears in media M if it is identifiable in M
has-value	Variable → Value	A literal attribute L has a value V

The fundamental, built-in relationships are given in Table 1. However, the application can add any other application dependent relationship types that may be needed.

Edge Type Hierarchy: The edge labels **L** described so far can also be organized in a hierarchy $<_p$ in the same vein as property hierarchies allowed in the semantic web language RDF [27]. There are six nodes at the first level of the property hierarchy that group these relationships into six categories, which are called *facets of an event* by Westermann and Jain[2]:

- Spatial: Spatial edges such as *has-some-location* in Table 1 can be of two types, absolute and relative. The edge type *has-exact-location* $<_p$ *has-some-location* because it is more specific.The edge type *adjacent-to* is a spatial (topological) relationship where the location of one object is specified relative to another. For the purpose of this paper, we do not consider relationships that need additional parameters. Thus, we do not admit directly spatial relationships like S1 *is-within-3-blocks-of* S2. However, in most cases we can model such information indirectly. For example, one can state that event (E *has-some-location* S1), (S2 *has-value* circle(*pos*, 1.5, km)) which states that E occurred within a 1.5 km radius of location *pos*, which will be actually expressed as absolute coordinates.
- Temporal: Temporal edges, like spatial edges, can also be absolute or relative, and shares a similar mode of expression. For example, while we cannot express the statement "events E1, E2 happened within 2 days of each other", knowing that E2 occurred at time T2, we can state (E *occurs-some-time* T1), (T1 *has-value* time-distance(*T2*, 2, day)), where the function time-distance specifies the length of a temporal window around a time constant. Unlike space, however, we use a special value for time called *NOW* to refer to temporal extents that are valid at the current time.
- Structural: A structural edge is an event-to-event edge like *subevent-of* or *next-event* that places a constraint between two events without referring to their spatiotemporal extents. The *subevent-of* relationship is transitive, i.e., if E1 is a *subevent-of* E2, which is a *subevent-of* E3, one can infer that E1 is a *subevent-of* E3. Consequently, the inverse relationship, *superevent-of* is also transitive. Further, a subevent must occur within the spatiotemporal extent of a superevent. However, the containment one event's spatiotemporal extent within that of another does not necessarily mean that they are structurally related. So structurally, a question-answer session is a subevent of any conference presentation, the ringing of a mobile phone during a presentation

does not imply that the mobile phone ringing is a subevent of the presentation. An event with no subevents is called an *atomic event*. Similarly, the *next-event* relationship makes the second event a *logical next occurrence* after the first event instead of just being a spatiotemporal neighbor that follows the first event. For example, the presentation of the third paper in a conference session is the next event after the discussion-section on the second paper of the conference. So, the event that some more people walked into the conference hall after the second presentation, although temporally juxtaposed to the discussion-section, is not the next event in the application setting of the conference schedule.

- <u>Experiential</u>: An experiential edge is an edge between media and another node type. In Table 1, we have shown three experiential edge types: the media-to-event edge type "witnesses", the media-to-object edge type "depicts", and the media-to-media edge type "segment-of". Among them, the "witnesses" edge is used as follows. In our conference example, the event node (001837, speech, "Tom's talk") and the media node (449802, image, "Tom at the podium") can be connected by the edge (449802, witnesses, 001837).
- <u>Causal</u>: Not shown in Table 1 is a class of edges which relate two events by a causality or evidentiary connection. If event E2 causes event E1, we write (E1 *is-caused-by* E2). The *is-caused-by* edge type is not directly transitive. If event E1 *is-caused-by* event E2, and event E2 *is-caused-by* event E3, one can infer E1 *is-indirectly-caused-by* E3. The *is-indirectly-caused-by* relationship has a ***path-length-limited transitivity***. That means, after a chain of *k* consecutive occurrences of the *is-indirectly-caused-by* edge, the relationship would cease to hold.

A second edge type of the same family is *evidence-of*. If we want to state event E2 is an evidence that event E1 has occurred (but is not directly observed) we create the edge (E2 *evidence-of* E1). Often a media object will be a witness of an observable event, and then serve as an indirect evidence of an unobserved but inferred event. For example, an image of door may witness the event that the door was forced open, which in turn may be evidence that a burglary had occurred.

In most real cases, it is often impractical to assign all causality or evidentiary relationships with absolute certainty; so both these relationships will be probabilistic. However, a detailed treatment of probabilistic relationships is beyond the scope of this paper.

- Informational: An informational edge type is usually associated with all other information about an event. For example, the participants of an event, or a general description of the event, the narrator or author of the event are examples of informational edges that relate events to metadata related to the event. The edge type hierarchy is most predominantly observed over informational edge types. For example, if the "conference" is a subclass of "event" and "person" is an subclass of "object" in an application, one can create edge types like *organizer-of* and *session-chair-in* as subproperties of *participates-in*.

The above discussion also illustrates the interplay between the node type and edge type hierarchies in the context of an application. In most applications, one tends to use only application specific node types, and for each node type one would create a set of edge types that capture the relationships formed by that node type with others. This will be developed more fully in the next section, where we view multimedia content through the glasses of our event model.

2.2 Events – A Unified Context for Multimedia Content

Analyzing the content and semantics of multimedia data has been the area of active research for over two decades [16, 22, 24, 28, 31, 32], and a large number of techniques continue to be developed till date. Among these techniques, a subset of techniques focuses on the extraction of events such as the scoring of a goal in a soccer video [13] by analyzing multiple media streams in synchrony to detect the event. In this paper, we do not focus on the content/event extraction or semantics classification techniques. Instead, we look at the "output side" of these algorithms and consider a media description collectively produced by a suite of algorithms acting together on a media object. Consider the accompanying the image shown in Figure 1.

Figure 1. A dinner during a conference

There can be a face recognition algorithm that specifies that there are four faces (and hence four persons) in the image, an EXIF record-based classification system can classify it to be an indoor picture, the camera records the date, time and the GPS of the location where this image was taken, other model based image understanding software can recognize plates, bowls and bottles, and possibly a classification algorithm based on machine learning can identify that the lower part of the image is most likely to be dining table, and a higher level image understanding software can potentially put all the information together and interpret the image as an "indoor dining event" with four participants. So a complete description as obtained by all the algorithms collectively can possibly be described as in Figure 2(a). Note that this description, albeit fairly complete, is confined to the information extracted from the image. Now consider how this information can be contextualized by placing it in terms of events. Figure 2(b) describes the event setting in the form of an event graph (we use triples in the style of RDF to represent edges of the graph) with pointers to information obtained from the image analysis. This enhances the image-based description with a context which can then relate to other information. For example, if we have an additional audio clip that is a speech by person-1 about the collaboration, this clip can be ***semantically aligned*** (synchronized) with the image, based on the fact that they co-reference the same event. In a later section, we will show that media items that are semantically aligned can be assimilated together based on their association with events.

3. Composing Events

Composition is the process of constructing a higher level entity based on the properties of a set of lower-level entities. For events, one would like to (a) *infer the properties of a higher-level event* by operating on the properties of events "under" it, and (b) *construct a new higher level event* by operating on a set of events whose properties are known. There are three built-in relationships *subtype-of*, *subevent-of* and (*caused-by* ∪ *indirectly-caused-by*) that constitute hierarchies over which one can define higher-level events. Of these, we do not consider the *subtype-of* relationship because it is the same as the *is-a* relationship in standard object-oriented models, and does not contribute to anything specific with respect to events. We consider the other two hierarchies, one at a time, to be the ***event composition axis***, and explore how the other, to-be-composed property classes (i.e., edge types) can be inferred around the axis. In this paper, most of our effort will focus on the *subevent-of* as the event composition axis as well as our initial work on event composition [29]. However this paper is the extension to our

ImageID: 23875 Date: 11/03/2006 Time: 12:34 PM Latitude: 30.2 Longitude: 120.01 LocationName: Hangzhou ImageClass: indoor ImageEventClass: dining ObjectsRecognized: Humans: 4 Person[1]: "ABC" FaceCoordinates: <polygon> … Objects: 10 DiningTable: ObjectCoordinates: <polygon> Bottle: ObjectCoordinates: <polygon> …. Relationships: Person[1] *adjacent-to* DiningTable …	(event982 isa dining-event) (event982 has-exact-location-name "Oriental Hotel"_ (event983 isa collaboration-meeting) (event983 subevent-of collaboration21) (event983 co-located-with event982) (event 984 isa toasting-event) (event984 subevent-of event982) (event985 isa decision-event) (event985 has-description "We agreed on the scope of the collaboration") (event984 is-celebration-of event985) (image23875 witnesses event984) (person1 same-as Person:ID:20) (person2 same-as Person:ID:251) …

Figure 2. (a) Description based on image content analysis (b) An event-based contextualization of the image. The arrows show the correspondence between the two descriptions.

previous work since we will discuss all facets of events here, rather than only spatial and temporal.

To achieve this, we introduce O_{space}= {S_{space}, R_{space},, L_{space}, H_{space}} representing Spot, Region, Line, and Hybrid spatial operators; O_{time}= {M_{time}, I_{time}, TH_{time}} representing moment/time-point[7], time-Interval, and Time-Hybrid temporal operators, O_i representing informational operators ,and O_{mm} representing experiential operators**.** Figure 3 shows two events e1 and e2 are composed via spatial, temporal, informational, and experiential operators to form composite event E.

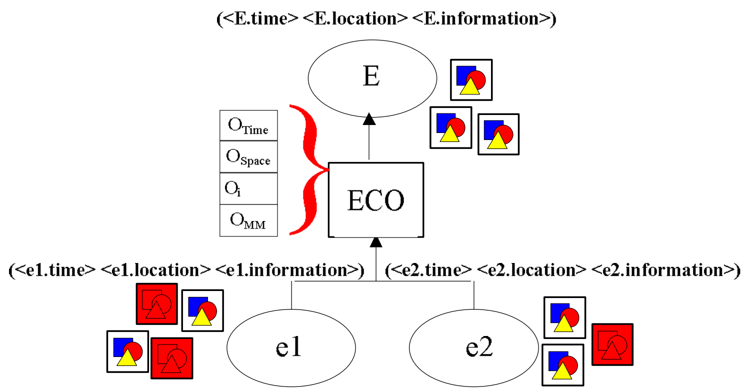

Figure 3. Event Composition

We define operators independent of application domains. However in various domains different operators are used manually or automatically. The types of attributes are identified and proper operators are utilized. For instance, to compose two atomic events, both with spatial attribute *spot* into a composite event with spatial attribute *line*; spatial operator L_{spot} (L indicates the output is a line) which is a member of L_{space} family (introduced later) is used.

3.1 An Informal Look at Event Composition

Consider the oversimplified view of a conference as shown in Figure 4.

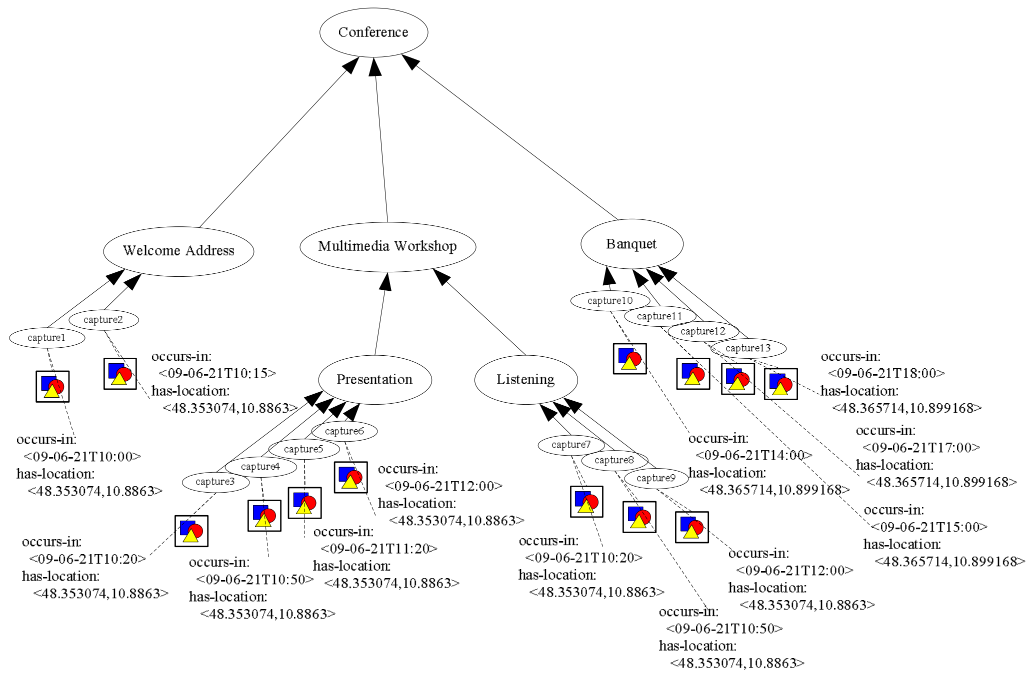

Figure 4. A simplified view of a conference

In this Figure, the term capture refers to an image that has been taken at the conference; all capture events are atomic events, while "Welcome Address", Presentation, Listening, "Multimedia Workshop" etc. are composite events. The temporal extent of the event "Presentation" can be constructed (composed) from the capture events under it, and is computed as [09-06-21T10:20, 09-06-21T12:00]. In the same vein, its experiential property is the set of images associated with the events capture3 – capture6 (we call them capture3.P_E etc., to designate the picture associated with the capture event). Notice that in the first case, the temporal attributes were constructed by point-to-interval conversion, followed by interpolated-interval-stitching, while in the second case, the operation was a (recursive) set union. The goal of this paper is formalize these operations for arbitrary property classes.

3.2 Event Composition for Subevents

We will structure this section based on the property classes (edge types) discussed in Section 2. Our goal is to take compose the other 5 property classes around subevents.

3.2.1 Composition of Informational Properties

The attribute domains of informational properties are usually infinite but enumerable. This makes composition of these properties a simple case of recursive union, much like standard property aggregation in object-based systems. Thus, the list of participants in a higher level event is the union of all participants in all its subevents. Despite the simplicity of this case, it is often very useful in the context of creating multimedia presentations. For example, to prepare a multimedia response for a query like “who participated in the banquet?” one can not only use images that were explicitly associated with the banquet or any of its subevents, but also photographs of people, who were known to be present. Another composition operation regarding informational properties of events is merging the social networks comprised of participants of subevents in a composite event to create a precise social network for that composite event. For this purpose we introduce a binary operator that takes two input graphs with nodes representing participants of the relevant subevents, and the edges representing the social relationship between them. The operator task is to merge the input graphs possibly into one (with no duplications in nodes). In addition to that, the operator takes an input set of rules to help in the merging process. A subset of these rules specify the reasoning criteria (for instance rule “a person can only have one and only one father” is a reasoning criterion), while another subset defines the characteristics of social relationships (i.e. being transitive, etc). According to such rule, to merge a graph that represents “Joe father-of Lisa” with another graph “Jon friend-of Susan”, it is invalid to create edge “father-of” from “Jon” to “Lisa” since she already has a father. Also, if “ancestor-of” is defined to be transitive edge, from “Joe ancestor-of Helen” and “Helen ancestor-of Naomi”, it can draw the conclusion that “Joe ancestor-of Naomi”, and create a direct edge with label “ancestor-of” from “Joe” to “Naomi”. Table 2 lists the informational operators $O_i=\{U_i,X_i\}$ for union and merge operations respectively.

Table 2: O_i

Signature	Output	Comments
$U_i(P_1[], P_2[])$	$(P_1[] \cup P_2[]) - (P_1[] \cap P_2[])$	$P_1[]$, $P_2[]$ = participant-list
$X_i(N_1, N_2, RULE)$	N_3	N_1,N_2 = social-network-graph, RULE = rule set

3.2.2 Composition of Causal Properties

We posit that the causality *hierarchy is orthogonal to the subevent hierarchy*, i.e., the relationship (A *is-caused-by* B) has no logical implication toward the relationship (B *subevent-of* C) even when (A *subevent-of* C). Therefore, no causality property can propagate along the subevent axis of composition.

3.2.3 Composition of Spatial Properties

We first elaborate on spatial properties. Let us consider spot, region, and line as *space elements*. Spot is a single point in the space represented, for instance, by a <lat,long> specification, e.g., Jon's-house *has-spatial-representation* spot-house, spot-house *has-latitude* "33.645115", and spot-house *has-longitude* "-117.831877". Thus spot-house has a *has-lat-long* edge to the value <33.645115,-117.831877> (where *has-latitude*, *has-longitude*, and *has-lat-long* are subproperties of *has-exact-location*). We treat a spot as a 0-D space indicated in Figure 5-(a). Line is a 1-D space represented by a collection of spots(i.e., lat-long pairs) which comprise connected collection of straight line-segments [34] indicated in Figure 5-(b). Line may be correlated with time since it sometimes represents a directed path starting from a source (src) and ending to a destination (dst); in that case the line has a direction: Figure 5-(c). Thus here the order of src, dst, and other spots located between src and dst on the line trajectory matters i.e. it implies a temporal order (the temporal location of src is always before the temporal location of dst). Region on the other hand is a 2-D space and expressed as a collection of <lat,long> pairs indicated in Figure 5-(d), e.g., his-farm *has-spatial-representation* region-farm, region-farm *contains* (spot-1 *and* spot-2), spot-1 *has-lat-long* <33.647639,-117.831774>, spot-2 *has-lat-long* <33.66355,-117.70679>. In fact we consider region to be a convex hull that is the boundary of the minimal convex set containing a non-empty finite set of spots X in a real vector space V. The mathematical constraint on such convex hull which makes it distinguished from a straight line on a 2-D surface, is being a simple closed polygonal chain containing non-collinear spots $s \in X$. This is only for input set of *spots*. For other types of input space elements, other formulations may have been used in Table 4. In this paper we only consider 2-D space for line and region. Extensions to consider altitude as the third dimension appear to be straightforward and are not considered in this paper for brevity. We define Hybrid as a set containing union of space elements; an example is a collection of a spot, a region, and a line. These concepts are used in Formulation of spatial operators. Having the assumption that the Hybrid set in Figure 6 is obtained as a result of a spatial operator, the collection of "spot-1 *is-entrancePoint-of* region-1", and "line-1 *is-disjoint-with* region-1" is an example of applying higher level composition techniques on the Hybrid set to stitch spot-1, region-1, and line-1 together and ultimately obtain the structure in Figure 6. Some of these relationships like *entrancePoint-of* express semantics of domain knowledge. RCC-8 relationships [6] (e.g. *disjoint-with*) on a hybrid set can as well create a SpatialGraph similar to Figure 6 showing a stitched holistic view of the elements based on inter-space-element relationships, though they don't need domain knowledge. In order to understand the utility of Hybrid type, we use an example. Let's imagine that a robbery has taken place in a jewelry store. There are multiple spatial locations associated with composite event *robbery*: a spot at which the thief has been shot, and the region in which the evidence of robbery is present. Police needs to realize the semantic relationship between these two spatial elements for its investigation. Without the existence of the hybrid set which consists of that spot and region, no tool can express semantics of the domain knowledge to state the spot is actually the *entrancePoint-of* that region. Thus not only RCC-8 relationships can be expressed on a hybrid element, but also semantic relations derived from domain knowledge can be applied on the sub-elements of a hybrid element.

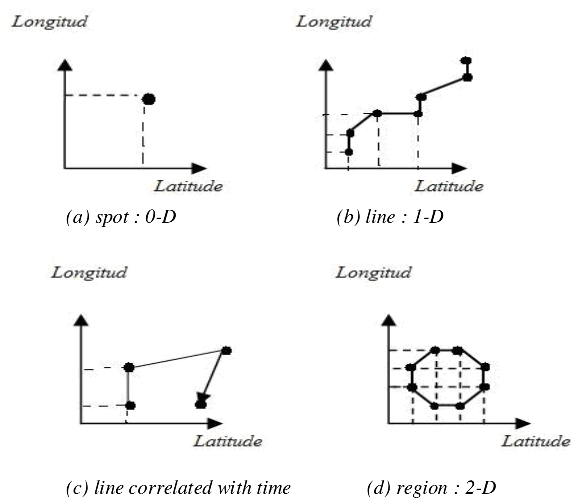

Figure 5: Spatial Elements

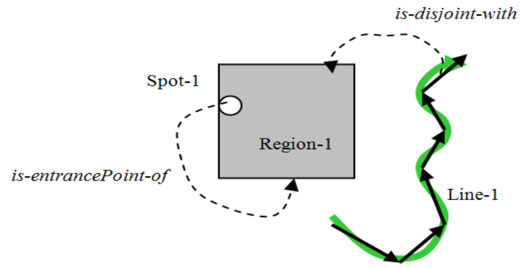

Figure 6: SpatialGraph

We categorize the main spatial operators into four groups S_{space}, R_{space}, L_{space}, and H_{space} such that the associated derivations of each group are supposed to return a spot, region, line, and hybrid representation of their subordinate elements respectively. The spatial operators using the introduced notions (i.e., spot, line, region, and hybrid) are listed below:

$S_{space}= S_S$;

$R_{space}= \{R_S , R_L , R_R , R_{S-R} , R_{L-R} , R_{S-H} , R_{L-H} , R_{R-H} , R_H\}$;

$L_{space}= \{L_S , L_{S-L} , L_L , L_{S-H} , L_{L-H} , L_H\}$;

$H_{space}= \{H_S ,H_{S-R} ,H_{S-L} ,H_{S-H} ,H_R ,H_{R-L} ,H_{R-H} ,H_L ,H_{L-H}, H_H\}$;

We have used "S" as "spot", "R" as "region", "L" as "line", and "H" as "hybrid" for the sake of brevity in the notations. The term *space* used in these notations refers to the type of input parameters on which these operators function on. For instance S_S indicates the input parameters of this operator are of type *spot* and the operator is expected to return a spot, on the other hand R_{S-R} indicates that the type of input parameters are of type spot and region, and the operator is expected to return a region (i.e. output). Tables 3-6 show spatial operators return heterogeneous spatial properties as a function of their input parameters. In each formulation, the spatial relationships between the input parameters yields a different sub-formula (e.g. in case of operator R_R in Table 4, if region parameters are equal, sub-formula-(b) is used, while if they are overlapping, sub-formula-(d) is used) . The asterisk symbol in all the tables of this paper implies applying multiple operators to the input parameter(s).

The result obtained by an operator is a function of its properties i.e., being *commutative*, *reflexive*, and *associative*. All the spatial operator types except L_{space} are *reflexive* in a subset of their family. The remaining subset in each type that functions on heterogeneous space-elements cannot have the reflexive property. For instance S_{S-R} is not reflexive since it can only be valid to function on a spot and region, not on a spot and itself. All the spatial operator types except L_{space} that is correlated with time are *commutative*. In contrast with the addressed properties, a*ssociative* property cannot be determined in the level of operators as it is purely based on the application specification, thus we don't talk about that here. Please consider the following abbreviations used in tables 3-6 for RCC-8 spatial relations: EQ = EQuals, EC = Externally Connected, IC=Internally Connected, DIS = DISjoint, CON = CONtains, OV = partially OVerlaps. Note that *is-<relationship-name>* opposites the implication of the relationship i.e., *r1 is-CON r2* means *r1 is-contained-in r2* OR *r2 contains r1*. We describe the families of S_{space}, R_{space}, L_{space}, and H_{space} in the following tables.

Table 3. S_{space} Family

Signature	Output	Comments
$S_S(S_1,S_2)$	a) $S_S(S_1$ EQ $S_2) = S_1$ OR S_2; b) $S^c_S(S_1$ EQ $S_2) = S_1$; c) $S^f_S(S_1$ EQ $S_2) = S_2$; d) $S_S(S_1$ DIS $S_2)$= undef.;	a) No comments. b) Coarser presicion heterogeneity c) Finer presicion heterogeneity d) undef ≡ not valid.

Table 4. R_{space} Family

Signature	Output	Comments
$R_S(S_1,S_2)$	a)$R_S(S_1$ DIS/EC $S_2)$= R ; b)$R_S(S_1$ EQ $S_2)$= undef;	a) R=convex hull containing S_1 and S_2. b) invalid output.
$R_L(L_1,L_2)$	a)$R_L(L_1 \underset{\geq 2}{\cap} L_2) = R$; b)$R_L(L_1 \underset{=0,1}{\cap} L_2)$= undef;	a) region surrounded by L_1 and L_2 b) invalid output
R_{S-H} (S,H)	a)$R_{S-H}(S \in H)=R_S*R_{S-R}*R_R*R_{L-R}*R_L$ (H); b)$R_{S-H}(S \notin H)$ = undef.	a) Applying all the specified operators to H.
$R_R(R_1,R_2)$	a)$R_R(R_1$ IC/is-CON $R_2)$= R_2; b)$R_R(R_1$ EQ $R_2)$=R_1 OR R_2;	a) No comments b) No comments

	c)$R_R(R_1$ EC $R_2)=(R_1 \cup R_2)$; d)$R_R(R_1 OV R_2)=(R_1 \cup R_2)-(R_1 \cap R_2)$; e)$R_R(R_1$ DIS $R_2)$= undef ;	c) creates a new region d) creates a new region e) invalid output
R_{S-R} (S,R)	a)R_{S-R}(S DIS/EC R)= undef ; b)R_{S-R}(S IC/is-CON R) =R.	No comments
R_{L-R}(L,R)	a)R_{L-R}(L is-CON/IC/EC R)=R; b)R_{L-R}(L OV R)= undef.	a line does not have width, only length.
R_{L-H}(L,H)	a)$R_{L-H}(L \in H)=R_S* R_{S-R}* R_R* R_{L-R}* R_L(H)$; b)$R_{L-H}(L \notin H)$ = undef.	a) Applying all the specified operators to H.
R_{R-H}(R,H)	a)$R_{R-H}(R \in H)=R_S*R_{S-R}*R_R*R_{L-R}*R_L(H)$; b)$R_{R-H}(R \notin H)$ = undef.	a) Applying all the specified operators to H.
$R_H(H_1,H_2)$	$R_H(H_1,H_2)= R_S*R_{S-R}* R_R* R_{L-R}* R_L((H_1 \cup H_2)- (H_1 \cap H_2))$	Applying all the specified operators to .non-duplicated union of input sets.

Table 5. L_{space} Family

Signature	Output
$L_S(S_1,S_2)$	a)$L_S(S_1$ DIS/EC $S_2)$=L[src: S_1, dst: S_2]; b)$L_S(S_1$ EQ $S_2)$= undef;

L_{S-L}(S,L)	a)L_{S-L}(S is-CON/IC L)= L;
	b)L_{S-L}(S EC L)=L'[src:S, trajectory: {L.src U all $S_i \in$ (L.trajectory)},dst:L.dst];
	c)L_{S-L}(S DIS L)= undef;

$L_L(L_1,L_2)$	a)$L_L(L_1$ DIS $L_2)$= undef;
	b)$L_L(L_1$ EQ $L_2)= L_1$ OR L_2;
	c) L_L (L_1 is-CON/IC L_2)=L_2;
	d)$L_L(L_1$ EC $L_2)$=L[src:L_1.src, trajectory: {all $S_i \in$ (L_1.trajectory)U L_1.dst U L_2.src U all $S'_i \in$ (L_2.trajectory)}, dst: L_2.dst];
	e)$L_{line}(L_1$ OV $L_2)$=L[src:L_1.src,trajectory:{ all $S_i \in$ (L_1.trajectory) U L_1.dst U L_2.srcU all $S'_i \in$ (L_2.trajectory)-(($S_i \in$ (L_1.trajectory) U L_1.dst))$\cap$(L_2.srcU $S'_i \in$ (L_2.trajectory)))}, dst:L_2.dst].

L_{S-H}(S,H)	a)$L_{S-H}(S \in H)= L_S* L_{S-L}* L_L(H)$; b)$L_{S-H}(S \notin H))= L_S* L_{S-L} * L_L(S,H)$.
L_{L-H}(L,H)	a)$L_{L-H} (L \in H)= L_S* L_{S-L} * L_L(H)$; b)$L_{L-H}(L \notin H))= L_S* L_{S-L} * L_L(L,H)$.

$L_H(H_1,H_2)$	$L_H(H_1,H_2)=L_S*L_{S-L}*L_L((H_1 U H_2)-(H_1 \cap H_2))$.

Table 6. H_{space} Family

Signature	Output
$H_S(S_1,S_2)$	a) $H_S(S_1$ DIS/EC $S_2)= (S_1 U S_2)$; b) $H_S(S_1$ EQ $S_2)= S_1$ OR S_2.
$H_{S-R}(S,R)$	$H_{S-R}(S,R)=(S U R)$;
$H_{S-L}(S,L)$	$H_{S-L}(S,L)=(S U L)$;
$H_{S-H}(S,H)$	a) $H_{S-H}(S \in H) = H$; b) else $H_{S-H}(S,H) =(S U H)$.
$H_R(R_1,R_2)$	a) $H_R(R_1$ DIS/EC $R_2)=(R_1 U R_2)$; b) $H_R(R_1$ EQ $R_2)= R_1$ OR R_2.
$H_{R-L}(R,L)$	$H_{R-L}(R,L)=(R U L)$;
$H_{R-H}(R,H)$	a) $H_{R-H}(R \in H) = H$; b) else $H_{R-H}(R,H) = (R U H)$.
$H_L(L_1,L_2)$	a) $H_L(L_1$ EQ $L_2)=L_1$ OR L_2 ; b) $H_L(L_1,L_2)=(L_1 U L_2)$.
$H_{L-H}(L,H)$	a) $H_{L-H}(L \in H) = H$; b) else $H_{L-H}(L,H) = (L U H)$.
$H_H(H_1,H_2)$	$H_H(H_1, H_2)= (H_1 U H_2) – (H_1 \cap H_2)$.

Example1:S_S

There is an interesting observation in parameters of operators S_S-(a,b,c) in Table 3. In all of these three operators the parameters are equal but it does not mean they have the same level of precision. Given two spots S_1 and S_2 with <lat,long> values <33.64,-117.83> and <33.64763,-117.83177> respectively, one can realize that these spots are at different level of precision. Operators S_S-(b,c) are created for this type of situation. It is up to the application whether it prefers to have the finer or coarser precision. If it prefers finer precision, S_S-(c) is used; otherwise S_S-(b) is used (coarser precision). S_S-(a) on the other hand is used when the precision level of parameters is the same.

Example2: H_S

Figure 7 shows composing 2 events on their spatial aspect spot-1 and spot-2 using operator H_S described in Table 6. H_S takes the spatial attributes of both events as its input parameters.

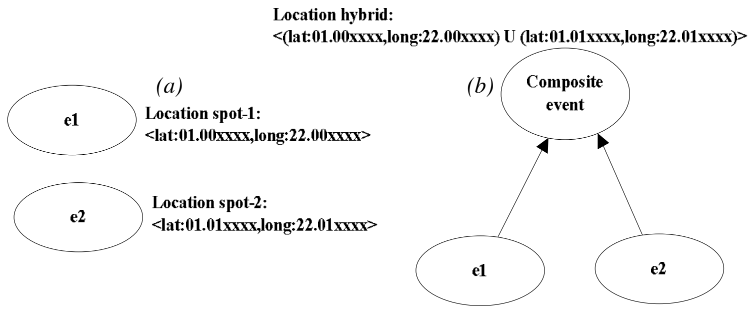

Figure 7: Function of operator S_S (a) before composition; (b) after composition

Having descriptions indicated in the Figure 7-(a), and "spot-1 *is-disjoint-from* spot-2", concludes H_S(spot-1 DIS spot-2) = <spot-1 U spot-2> which is produced in Figure 7-(b); The output can be represented afterwards with small circles on a geographical map (based on the corresponding lat-long coordinates) to show their concentration on a political geographical region of the map. In a trip scenario this can be used to show the most visited region identified by the concentration factor. Another use case of this operator is for the *robbery* example we emphasized earlier.

Example3: R_L

Assume atomic events e1 and e2 have spatial attributes line1 and line2 in Figure 8. Line1 and line2 have *2* intersected spots in common.

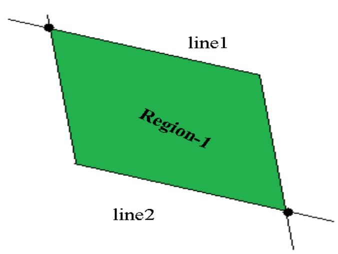

Figure 8. Function of Operator R_L

If the application needs to have the spatial attribute of the composite event in form of a region, operator R_L is used for the computation. Operator R_L-(a) in Table 4 with input parameters line1 and line2 returns the colored region

"Region-1" surrounded by line1 and line2. Please note that for each line, all the spots on the trajectory of the line are taken into account to surround the colored region in Figure 8.

3.2.4 Composition of Temporal Properties

We consider time as a linearly ordered property. Temporal operators have classification $O_{time} = \{M_{\mathrm{time}}, I_{\mathrm{time}}, TH_{\mathrm{time}}\}$. *M* implies *moment* or an instant time-point on a timeline, while *I* as interval is the continuous representation of set of *moments* ,or the linearly ordered set of *moments* with two outstanding moments: M_{start} and M_{end} representing the starting and ending moments of the interval respectively such that $M_{\mathrm{start}} \neq M_{\mathrm{end}}$. We define THybrid as a set containing union of time elements, e.g. a collection of a moment, and an interval. Similar to the Hybrid type, THybrid type is also beneficial in terms of allowing other tools express the semantics of domain knowledge on the sub-elements of a thybrid set. For instance in the composite event *robbery* stated in the previous subsection, according to the domain knowledge some tool can express that "the moment at which the thief is shot" *is-the-important-moment-of* "the interval associated with composite event *robbery*". Without Thybrid type, representing such domain knowledge semantics would be impossible.

We represent a time interval by one of the notations [m1-m2], [m1-m2) , (m1-m2], or (m1-m2) -- $m_1 = M_{\mathrm{start}}$, $m_2 = M_{\mathrm{end}}$ -- referencing the mathematical notions "bounded", "left-bounded", "right-bounded", and "unbounded" respectively to specify the underlying temporal boundaries. We categorize the main temporal operators into three groups M_{time} (all its derivations return moment representation of their subordinate elements), I_{time} (all its derivations return interval representation of their subordinate elements), and TH_{time} (all its derivations return temporal hybrid representation of their subordinate elements. Applying Allen's relationships [5] on a thybrid set obtained from a temporal operator can create a TemporalGraph similar to Figure 9. The TemporalGraph presents a stitched holistic view of the elements. The temporal operators using the introduced notions (i.e., moment, interval, and thybrid) are listed below:

$M_{\mathrm{time}} = M_{\mathrm{M}}$;

$I_{\mathrm{time}} = \{ I_{\mathrm{M}} , I_{\mathrm{I}} , I_{\mathrm{M\text{-}I}}, I_{\mathrm{M\text{-}TH}}, I_{\mathrm{I\text{-}TH}} , I_{\mathrm{TH}}\}$;

$TH_{\mathrm{time}} = \{ TH_{\mathrm{M}} , TH_{\mathrm{I}} , TH_{\mathrm{M\text{-}I}}, TH_{\mathrm{M\text{-}TH}} , TH_{\mathrm{I\text{-}TH}} , TH_{\mathrm{TH}} \}$.

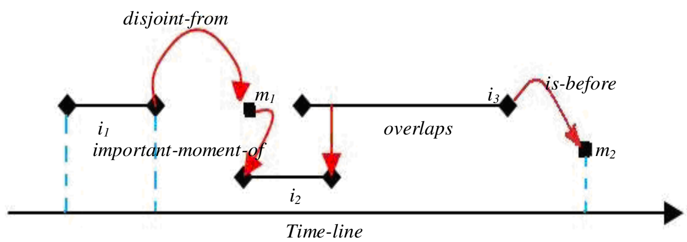

Figure 9. TemporalGraph

We have used "M" as "moment", "I" as "interval, and "TH" as "thybrid" in the temporal operators' notations for brevity. The term *time* used in temporal notations refers to the types of parameters on which temporal operators function on. For instance M_M indicates that the input parameters of this operator are of type *moment* and the operator is expected to return a moment, on the other hand $I_{M,I}$ indicates that the type of input parameters are of types moment and interval, and the operator is expected to return an interval. However sometimes according to inter-time-line-element relationships, temporal operators may be converted into each other. Temporal operators are formulated in tables 7-9. Abbreviations used in notations (in next tables) are based on Allen's Time Calculus: BF=before, DIS=disjoint, AF=after, EQ=equals, MT=meets, ST=starts, OV=overlaps, DUR=during, FIN=finishes, CON=contains. Temporal operators return heterogeneous temporal properties as a function of their input parameters. In each formulation, the temporal relationships between the input parameters yields a different sub-formula (e.g. in case of operator M_M in Table 7, if input parameters are equal and no precision heterogeneity is involved, sub-formula-(a) is used, while if precision heterogeneity is considered, sub-formula-(b) or –(c) is used) for finer or coarser precision heterogeneity respectively).

For I_M, we can not convert two equal moments to an interval since both moments are temporally projected to the same temporal location. However if the moments are not equal --I_M_(b) in Table 8)-- the operator is an interpolated operator. In this case, the interpolation is among discrete set of time-points (moments). Subsections b-1) – b-4) show 4 possible types of the output interval: bounded, right-bounded, left-bounded, and unbounded. In signature $I_M^{\boldsymbol{i}:[-,-]}(M_1,M_2)$, symbol $\boldsymbol{i}$ indicates that the operator embraces interpolation operation and converts the input time-points to a bounded interval.

We describe family of M_{time}, I_{time}, and TH_{time} in the following.

Table 7. M_{time} Family

Signature	Output
$M_M(M_1,M_2)$	a)$M_M(M_1$ EQ $M_2)=M_1$ OR M_2; b) $M^f_M(M_1$ EQ $M_2)= M_1$; c) $M^c_M(M_1$ EQ $M_2)= M_2$;

Table 8. I_{time} Family

Signature	Output
$I_M(M_1,M_2)$	a) $I_M(M_1$ EQ $M_2)=$ undef; b) if $(M_1<M_2)$ AND (M_1 MT/DIS M_2): b-1) $I_M^{i:[-,-]}(M_1 , M_2)=[M_1, M_2]$; b-2) $I_M^{i:(-,-]}(M_1 , M_2)=(M_1, M_2]$; b-3) $I_M^{i:[-,-)}(M_1 , M_2)=[M_1, M_2)$; b-4) $I_M^{i:(-,-)}(M_1 , M_2)=(M_1, M_2)$;
$I_I (I_1,I_2)$	a) $I_I (I_1$ EQ $I_2) =I_1$ OR I_2; b) if (I_1 OV/MT I_2): b-1) $I_I^{[-,-]}(I_1 , I_2)=[I_1.M_{start},I_2.M_{end}]$; b-2) $I_I^{(-,-]}(I_1 , I_2)=(I_1.M_{start},I_2.M_{end}]$; b-3) $I_I^{[-,-)}(I_1 , I_2)=[I_1.M_{start},I_2.M_{end})$; b-4) $I_I^{(-,-)}(I_1 , I_2)=(I_1.M_{start},I_2.M_{end})$; c) $I_I (I_1$ DUR/ST/FIN $I_2) = I_2$; d) $I_I (I_1$ DIS $I_2)$ =undef

$I_{M-I}(M,I)$	a) I_{M-I}(M DUR/ST/FIN I)=I;
	b) if (M MT I):
	b-1) $I_{M-I}^{[-,-]}(M,I) =I'[M,I.M_{end}]$;
	b-2) $I_{M-I}^{(-,-]}(M,I) =I'(M,I.M_{end}]$;
	b-3) $I_{M-I}^{[-,-)}(M,I) =I'[M,I.M_{end})$;
	b-4) $I_{M-I}^{(-,-)}(M,I) =I'(M,I.M_{end})$;
	c) I_{M-I}(M DIS I) = undef;

$I_{M-TH}(M,TH)$	a) $I_{M-TH}(M \in TH) = I_M* I_{M-I}*I_I (TH)$; b) $I_{M-TH}(M\notin TH)=I_M* I_{M-I}* I_I(M \cup TH)$.
I_{I-TH} (I,TH)	a) $I_{I-TH} (I \in TH) = I_M* I_{M-I}* I_I (TH)$; b) $I_{I-TH} (I\notin TH) = I_M* I_{M-I}* I_I(I \cup TH)$.
$I_{TH}(TH_1,TH_2)$	$I_{TH}(TH_1,TH_2) =I_M* I_{M-I}* I_I ((TH_1 \cup TH_2)$ $-(TH_1 \cap TH_2))$.

Table 9: TH_{time} Family

Signature	Output
$TH_M(M_1,M_2)$	a)$TH_M(M_1$ DIS/EC $M_2)=(M_1 \cup M_2)$; b)$TH_M(M_1$ EQ $M_2)= M_1$ OR M_2.
$TH_I(I_1,I_2)$	a)$TH_I(I_1$ DIS/EC $I_2)= (I_1 \cup I_2)$; b)$TH_I(I_1$ EQ $I_2)= I_1$ OR I_2.
$TH_{M-I}(M,I)$	$(M \cup I)$;
$TH_{M-TH}(M,TH)$	a)$TH_{M-TH}(M \in TH)=TH$; b)$TH_{M-TH}(M \notin TH)=(M \cup TH)$.
$TH_{I-TH}(I,TH)$	a)$TH_{I-TH}(I \in TH)=TH$; b)$TH_{I-TH}(I \notin TH)=(I \cup TH)$.
$TH_{TH}(TH_1,TH_2)$	$(TH_1 \cup TH_2)-(TH_1 \cap TH_2)$.

Example 4: TH_M

For a query like "create a discrete temporal presentation that shows events of *My 2009 Delhi trip* starting to happen in a sequential order", Prof. Joe chooses operator TH_M for union of the starting moments corresponding to subevents of *My 2009 Delhi trip*. The assumption is (conference , visiting Delhi , site-seeing trips) *subevent-of* My 2009 Delhi trip. Given Table 10, the following operator uses the required values:

TH_M(conference. M_{start}, visiting Delhi. M_{start}, site-seeing trips. M_{start})= <09-10-21T> U <09-10-22T> U <09-11-01T>.

Table 10. Input values

Starting moment	Value
conference. M_{start}	<09-10-21T>
visiting Delhi. M_{start}	<09-10-22T>
site-seeing trip. M_{start}	<09-11-01T>

Figure 10 shows the temporal presentation of the output on a timeline. Such composed result if provided to higher level application dependent composition techniques, can lead to interesting information. For instance if the application queries for "at which season of the year events of *My 2009 Delhi trip* started to happen?", a change-of-resolution technique returns <season: Winter> given the result obtained from TH_M. We don't discuss such techniques in this paper. The purpose is only to show how the result obtained by our proposed operators can provide required input data to other higher level operators in different applications.

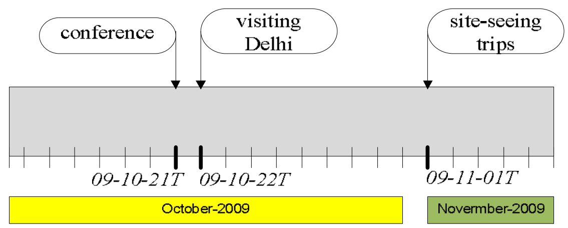

Figure 10. A temporal presentation for "My 2009 Delhi Trip"

Example 5: I_M

An apt example for this family of temporal operators is composing two disjoint moments into a single time interval using I_M operator described in Table 8. This is actually the interpolation operator that creates a new temporal element within the range of the discrete set of moments. To augment the temporal property of composite event "Welcome Address", one can use $I_M^{[-,-]}$ (capture1.M , capture2.M) that yields the closed temporal range [09-06-21T10:00, 09-06-21T10:15]. The result is the closed/bounded interval during which the "Welcome Address" took place.

3.2.5 Composition of Experiential Properties

Composition of experiential properties is more challenging compared to the cases we have seen so far. Therefore, we first provide an intuitive overview of the problem. Consider the query, "present a multimedia view of what happened at the conference". One response is to create a *union* of the pictures of all subevents of the conference and present a gallery of pictures ordered by time. A second response would be to present the *subevent hierarchy* as a data structure and for each subevent, present the pictures related to it. A third response would be to do the same as the second, but *eliminate pictures that are nearly similar* in content. A fourth response could be similar to the second, except that *some pictures are merged* into panoramic views because the system knows sufficient information about the relative coordinates of some pictures so that they can be stitched together meaningfully. Yet another answer will arrange the pictures by *composing them into a video*. We can come up with many such possible forms of response, each of which is a distinct form of media composition. Suppose we alter the query slightly to "present a multimedia view of what happened at the conference – use between 20-30 pictures". To accommodate the additional constraint on the minimum and/or maximum number of pictures, we would need to add *a filtering step* and/or *a ranking criterion* to select the images that should be used to represent a subevent within the cardinality budget. This leads to us to select the following operator classes that will be algebraically combined to achieve the composition of experiential properties as appropriate for different applications.

$O_{mm} = \{ F_M, E_M, U_M, St_M, O_M, C_M, D_M \}$

These operators are combination of operators that manipulate the media objects themselves (e.g., extraction), and operators that organize media objects into data structures that can be associated with higher level events (e.g., union).

1. F_M (Filtering): This operator, akin to a selection operation in database systems, uses an input list of media and a set of predicates P to be applied conjunctively. The filtering conditions can be based on metadata like EXIF parameters [14], or any content-based criteria on media features e.g., only outdoor images containing faces. One interesting application of filtering can be found in [19], where one can provide a set of M images and a number N ($N \ll M$) and the system uses a function F to "summarize" (i.e., filter) the correct set of N images that based characterize the event. In our setting, the set of M images will be provided by finding the witnesses of all subevents under the higher-level event we want to summarize.
2. E_M (Extraction): This operator accepts an input list of media and extracts a segment (SEG) from the media given a function that specifies what to extract. The function can be as simple as extracting a rectangular portion of an image, or complex like extraction of the minimal bounding rectangle containing an object type (e.g., people), extracting a keyframe from a video, and so forth.
3. U_M (Union): This is a binary operator that performs a set union over images, thereby eliminating duplicates. The mechanism of duplicate detection is algorithm dependent. If no explicit parameters are provided, the duplicates are based on image file names. If a content similarity algorithm and a threshold

are provided as parameters, the algorithm is used to detect content similarity and then duplicates are eliminated based on the threshold.

4. St_M (Stitching): The stitching operator accepts a list of media of any one kind (all images, all audio) and a set of *geometric join conditions* [17,18, 33] that specify how they should be composed together to create a new, geometrically larger media object. In a simple case example, two audio objects can be juxtaposed to construct a new audio. In our current formulation, the "stitching" is quite simple. However, we envision that an extended form of this operator can "stitch" multiple tracks of homogeneous or heterogeneous media into a composite MIDI like form.
5. O_M (Organizer): This operator accepts a set of media objects and organizes them into a data structure according to an allocation algorithm, which decides which media object will be placed in which part of the data structure. One example, discussed before, is to place all images on a directed acyclic graph that represents the subevent structure of a given higher-level event. A second example would be to create a spatial arrangement of images on a map so that the location of the images is mapped to the closed coordinates of the map. In this case, the map shows a spatial view of the higher-level event. In a third example, we might want to pick a higher-level event H and get a sense of the participation profile of this event. – to achieve this, we can associate the H with a social network graph algebraically constructed by augmenting nodes (which represent people) with media associated with those subevents of H where they participate.

The next two operators are different from the previous operators because they help in the construction of higher level events from a set of atomic or composite images.

7. C_M (Cluster): This operator accepts an input set of media objects (together with their metadata) and a suitable clustering algorithm and outputs a potentially hierarchical grouping over this set of objects. The clustering algorithm would determine which metadata or image features are used and whether the groups are allowed to overlap. This groups thus obtained can then be used to define a subevent hierarchy over the atomic events that the individual media objects represent.

8. D_M (DescriptiveAggregator): This operator invokes any arbitrary algorithm to compose the content descriptors (e.g. MPEG7) of media objects, potentially created from the output of the cluster operation, into one content descriptor (Dscr). This is useful when a new composite media object has been generated with absence of content descriptor. We assume that each media object has a field for its content descriptor.

The signature specification regarding the above experiential operators is given in Table 11.

Table 11. O_{mm} Family

Signature	Output	Comments
$F_M(M_1[m], \langle P \rangle)$	$M_2[n]$, s.t. n<<m;	<P> = set of predicates; M= media-obj
$E_M(M_1[m], \langle F(X) \rangle)$	SEG[m];	F(X) = function, X = extraction criteria
$U_M(IMG_1, IMG_2)$	$(IMG_1 \cup IMG_2) - (IMG_1 \cap IMG_2)$	Based on IMG file name
$U_M^{<X_1,\ldots,X_n>}(IMG_1, IMG_2)$	$(IMG_1 \cup IMG_2) - (IMG_1 \cap IMG_2)$	based on $<X_1,\ldots,X_n>$, (explicit parameters)
$St_M(M_1[m], \langle \cup_g \rangle)$	M', s.t. $M \gg M_1[i]$, i=1,..,m	M' = large new media-obj $\langle \cup_g \rangle$= set of geo-join conditions
$O_M(M_1[m], Ds, \langle a\lg o^{allocation} \rangle)$	Ds allocated with $M_1[i]$,i=1,…m	Ds = data structure
$C_M^{metadata}(M_1[m], \langle a\lg o^{clustering} \rangle)$	H	H = hierarchical grouping
$D_M^{Dscr}(M_1[m], \langle a\lg o^{DscrCompose} \rangle)$	$M_1[m]$ with updated Dscr field	Dscr = media-obj content-descriptor

We emphasize that these operators are really operator classes because many of them accept different functions (e.g., a clustering algorithm) as part of their input.

3.3 Event Composition along Causality Hierarchy

Composing events and their properties along the causality hierarchy is a difficult problem. Unlike the subevent hierarchy where the containment of a subevent within a superevent necessarily implies a full aggregation of properties, the causality hierarchy created through both *is-caused-by* and *evidence-of* relationships are more "selective". For example, just because event A is *evidence-of* event B, it is not true that all properties of event A participate in the evidentiary role for event B. For example, while the time of occurrence for event A can be a critical factor with respect to the time of occurrence of event B, the exact location of event A may have little to do with its evidentiary value with respect to event B. Similarly, only a fraction of the pictures taken at event A may be relevant as evidences for event B. This requires us to develop a set of secondary relationships that state which properties of event A serve as evidence for which aspects of event B. This gets compounded when we consider probabilistic relationships. We consider a detailed treatment of the subject as beyond the scope of this paper, and mention it here for the sake of completeness.

4. Bringing It All Together

In the typical scenario of Prof. Joe's life raised at the beginning of this paper, we aimed three targets, each of which requires a particular set of composition operators to augment attributes of events. We discuss them case by case. Each case addresses particular aspects of events.

4.1.1 Case 1: Experiential ,Structural, and Informational

In this case, Prof. Joe intends to send a photo album of his trip to two different people with completely different tastes: his daughter, and a professional friend of his.

Let's say he knows that his daughter is interested only in outdoor activities. Thus a suitable photo album for his daughter is the one containing images corresponding to subevents of his trip, which are considered to be outdoor activities:

"(visiting-Delhi , site-seeing) *subtype-of* outdoor-activities", and

"(visiting-Delhi , site-seeing) *subevent-of* Prof.Joe's-trip".

Now that the required subevents are identified, the next step is to use U_M union operator to perform a set union over images of "visiting-Delhi" and "site-seeing", thereby eliminating duplicates, using either image file name,or a content similarity algorithm.

For his professional friend, he knows the taste is to see what people attended the conference in the trip. Rephrasing the previous statement is send photo album of subset of images containing:

"people *participates-in* conference", and

"conference *subevent-of* Prof.Joe's-trip".

Thus the next step is to use operator U_i on input participants of conference subevents (i.e. welcome-address, Multimedia-workshop, and Banquet) to obtain non-duplicated union of their participants. Of course if the informational property regarding any of the conference subevents is empty, the operator is used in recursive union fashion on lower subevents to augment the underlying property. The last step is using operator F_M to filter the union of images associated to "conference" along with a set of predicates that specify the filtering criteria based on the list of participants obtained from operator U_i in the previous step.

4.1.2 Case 2: Experiential and Causal

The aim of Prof. Joe in this case is producing a multimedia diary of interesting things he learnt at the conference for his students. Each interesting thing is an unobserved but inferred event *(evidence-of)*$^{-1}$ an observed event during his trip. Also each underlying observed event *(witnesses)*$^{-1}$ some media object. Hence the media object witnesses the unobserved event (interesting thing) indirectly. The multimedia diary is the union of interesting events Prof. Joe has experienced, indirectly witnessed by media objects associated to observed events that actally happened during the conference.

4.1.3 Case 3: Experiential, Spatial and Temporal

Finally Prof. Joe wants to post his a multimedia report on a travel website to describe his trip. Operator O_M (organizer) can place one image from each cluster obtained from operator C_M related to the trip on a map, organizing and mapping their location to the closed coordinates of map, and creating a multimedia report. He also uses spatial operator L_S to sketch a line on GoogleMap to visualize his trip. The output line passes through all the lat-long spots representing the places he has visited in a relevant temporal order. Also he creates a timeline similar to Figure 10 to report the temporal order in which each of subevents in his trip took place, and the duration of each using operators like TH_M and I_M.

5. Conclusion

Multimedia data is captured to represent experiential component of an event. Depending on the nature of the data, it may itself represent an event (like taking a photo) or may contain other subevents of the event (like in audio and video). Independent of its nature, multimedia data has very intimate relationship to events and can be analyzed and represented effectively and efficiently using events. Given the hierarchical nature of events and the fact that most data related to events is usually captured at atomic level and is associated data representing time, location, information and experiential attributes, one must compute similar attributes of higher level events by aggregating, assimilating, or compositing events. In this paper, we presented a brief sketch of our event data model and the role of multimedia information in this model. Based on this model, we formalized the concept of event composition and developed a set of basic operations that consolidate and propagate properties of lower level events to higher level events. We showed how heterogeneous types of data like time, location, information, and experiential data can be transformed and combined across other events at the same level to result in an appropriate type of attribute for the higher level. Composition operations on event properties open a wide range of research towards construction of event web [8]. While we focused our attention primarily to the problem of event composition along the subevent hierarchy, our general methodology is usable for other hierarchies that can be defined over events. In an application, a designer must decide about the selection of type for different attributes and then use appropriate operators to propagate them.

Considering how different sensors are being used to create microevents and systems are being built to understand how composite events must be created to understand real world situations, this paper represents the first step in a this exciting emerging area intimately tied to automated construction of composite multimedia objects.

Although we have taken the first step, many challenges remain. The problem of composing events across the causality hierarchy is an open problem. In case the events are related by a more semantic model such as an event ontology, how can we use the ontology to select the appropriate operators? One also needs to explore how application-based criteria for event and media composition can be used on top of or in conjunction with the composition operators we have considered. In applications where events are detected based on single or multiple sensor data streams, the processes of atomic event detection and higher-level event detection are performed in a bottom-up manner using signal processing techniques – it is unclear how these techniques might interact with the operator classes we have defined. We have not investigated the issue of runtime performance of these operators – we need to develop a set of experiments to evaluate whether these operations perform efficiently enough for use in applications that need "on-demand composition". These issues will be part of our future work. However, we believe that this work identifies a novel and important research area that should be addressed for developing sophisticated complex event based multimedia applications.

6. References

[1] A.Quinton,"Objects and Events", Mind, vol. 88, no. 350, 1979, pp. 197–214.

[2] U.Westermann, R.Jain, "Toward a Common Event Model for Multimedia Applications", *IEEE MultiMedia, Jan 2007.*

[3] U.Westermann,R.Jain, "Events in Multimedia Electronic Chronicles (E-Chronicles)", *International Journal of Semantic Web and Information Systems (IJSWIS), vol.2 No. 2 pp. 1-27, July 2006.*

[4] A.Scherp,T.Franz,C.Saathoff,S.Staab," F—A Model of Events based on the Foundational Ontology DOLCE+DnS Ultralite", *K-CAP, 2009.*

[5] J.F. Allen,"Maintaining knowledge about temporal intervals", *Comm. ACM 26 ,1983.*

[6] S.Li, H.Wang,"RCC8 binary constraint network can be consistently extended", *Artificial Intelligence,2006.*

[7] P.Terenziani, R.T.Snodgrass, "Reconciling Point-Based and Interval-Based Semantics in Temporal Relational Databases: A Treatment of the Telic/Atelic Distinction", *IEEE Transactions on Knowledge and Data Engineering, May 2004.*

[8] R.Jain, "Toward Event Web", *IEEE Computer Society, Sep 2007.*

[9] A.R.J.Francois,R.Nevatia,J.Hobbs,R.C.Bolles,"VERL: An Ontology Framework for Representing and Annotating Video Events",*Journal IEEE MultiMedia,2005.*

[10] J. Hunter, "Reconciling MPEG-7 and MPEG-21 Semantics through a Common Event-Aware Metadata Model," *Computing Research Repository (CoRR), Oct. 2002.*

[11] D.Pack, R. Singh, S.Brennan, R.Jain, "An event model and its implementation for multimedia information representation and retrieval", *Proc. 2004 IEEE Int'l Conf.Multimedia and Expo(ICME),IEEE press,2004,pp. 1611-1614.*

[12] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, "WonderWeb deliverable D18 ontology library", *IST WonderWeb project, 2003.*

[13] J.R. Wang and N. Parameswaran, "Survey of sports video analysis: research issues and applications", Conference in Research and Practice in Information Technology, Australia,*2003.*

[14] Technical Standardization Committee on AV & IT Storage Systems and Equipment : http://www.exif.org/specifications.html, "Exchangeable image file format for digital still cameras: Exif Version 2.2", *Apr 2002.*

[15] J.M. Martinez, R. Koenen, F. Pereira,"MPEG-7 The Generic Multimedia Content Description Standard, Part 1", IEEE , *2002.*

[16] D. Brezeale, and D. J. Cook. “Automatic Video Classification: A Survey of the Literature”, IEEE Trans. on Systems, Man, and Cybernetics, vol. 38, no. 3, *May 2008*,

[17] Photosynth: http://photosynth.net/.

[18] Autostitch: http://people.cs.ubc.ca/~mbrown/autostitch/autostitch.html.

[19] P. Sinha., H. Pirsiavash, and R. Jain,”Personal photo album summarization” In *Proceedings of the Seventeen ACM international Conference on Multimedia*. MM '09. ACM, New York, NY, *Beijing, China, October 19 - 24, 2009*.

[20] R. Arndt, R. Troncy, S. Staab, L. Hardman, “COMM: A Core Ontology for Multimedia Annotation”, International Semantic Web Conference (ISWC 2007), pp 30-43, *Nov 2007*.

[21] A. Scherp, S. Agaram, R. Jain , “Event Centric Media Management”, In: Gevers, T., Jain, R.C., Santini, S. (eds.) Multimedia Content Access: Algorithms and Systems II. Proceedings of the SPIE Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, vol. 6820, pp. 68200C-68200C-15, *Jan 2008*

[22] Yang, Z., & Kuo, C. “Survey on image content analysis, indexing, and retrieval techniques and status report of MPEG-7”. *Tamkang Journal of Science and Engineering,1999*

[23] R. Datta,J. Li, and J.Z. Wang “Content-based image retrieval: approaches and trends of the new age”, *Proceedings of the 7th ACM SIGMM international Workshop on Multimedia information Retrieval* .ACM, New York, NY, 253-262,Singapore, *Nov 10 - 11, 2005*.

[24] U. Srinivasan , S. Pfeiffer , S. Nepal , M. Lee , L. Gu and S. Barrass “A survey of MPEG-1 audio, video and semantic analysis techniques,” *Multimedia Tools Appl.*, vol. 27, pp. 105, *2005*.

[25] E. Spyrou, G. Tolias, Ph. Mylonas, “A relation-based contextual approach for effcient multimedia analysis”, Semantic Media Adaptation and Personalization, SMAP '08. Third International Workshop, 2008.

[26] ,“E*: A Faceted, Graph-based Event Model”, TR#1: University of California Irvine, Computer Science Department

[27] W3C Recommendation 10: http://www.w3.org/TR/rdf-schema/, “RDF Vocabulary Description Language 1.0: RDF Schema”, *Feb 2004*.

[28] S. Tsekeridou and I. Pitas. Audio-visual content analysis for content-based video indexing. In IEEE Int. Conf. On Multimedia Computing and Systems (ICMCS'99), volume I, pages 667--672, Florence, Italy, 7--11 *June 1999*.

[29] “Suppressed for blind review”.

[30] A. Satish, R. Jain, A. Gupta, “Tolkien: An Event Based Storytelling System”, PVLDB 2(2): 1630-1633 *Aug 2009*.

[31] Pfeiffer, S., Fischer, S., and Effelsberg, W. “Automatic audio content analysis”. In *Proceedings of the Fourth ACM international Conference on Multimedia* (Boston, Massachusetts, United States, November 18 - 22, 1996). MULTIMEDIA '96. ACM, New York, NY, 21-30. DOI= http://doi.acm.org/10.1145/244130.244139, *1996*

[32] T. Zhang and C.-C. J. Kuo. Audio content analysis for online audiovisual data segmentation and classification. IEEE Transactions on Speech and Audio Processing, 3(4), *2001*.

[33] J. Xing, Z. Miao, “An Improved Algorithm on Image Stitching based on SIFT features”, IEEE, *2007*.

[34] Noma T, Kunii TL, Kin N, Enomoto H, Aso E, Yamamoto, “Constructive picture description with euclidean geometry”, Visual Comput 5:40–52, *1989*.

[35] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-Based Image Retrieval at the End of the Early Years”, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349-1380, *Dec. 2000*.