SPaRC©

Similar Picture and Relationship Constructor

SPaRC Your Memory!

Scott Bishop, Chunyong Fei, Jensen Galan, and Stacie Riddle

Multimedia Studies
Department Of Computer Science

San Francisco State University
San Francisco, California
December 12, 2005

Introduction

(Scott Bishop)

The purpose of this work is to create a system to organize images using location, date, event annotation, and semantic noun relationships via WordNet [16][17]. An automated date time event clustering algorithm is used to segment the images into date time hierarchies; these segments are further grouped and bound by a location. An annotation can then be applied to the event timespan which introduces a semantic hierarchy used to automatically locate related events in the digital image collection. As suggested by [21] users have three general requirements when approaching a search for images. The first is identifying images at the event level; the second is the identification of individuals in the images; and the third is finding images at different events that share some common relationship. We approach the first and third requirements by incorporating event detection based on location and date time points of reference. The third requirement is addresed by finding related events based on weighted keywords given by event annotations and WordNet searches.

Proposed Problem Formulation

(Scott Bishop)

SPaRC uses existing technology in order to redefine the methods associated with organizing digital images. Other works that use location data [17][23][27][32], date time [2][23] and other attributes such as time of day [17] to organize information exclude an important facet of cognition, the semantic meaning. SPaRC directly addresses this requirement by incorporating a lexical memory component [16] to help facilitate the automatic clustering of images into events. SPaRC uses this additional information to find related events that span both spatial and temporal differences. These related events can give rise to special correlations such as cause and effect given that location and date time hierarchies are established before the semantic relationships are formed.

This work is important because it attempts to build upon current work in event and contextual digital image clustering using semantic, location, and temporal data to provide an automatic event and related events image browsing application.

High-Level Description

Event Clustering (Jensen Galan)

The goal of clustering in our system is to automatically segment the image collection into temporal clusters, which will further be grouped using the location data, representing a meaningful occurrence in somebody's life. By grouping and presenting the images in this manner, the users of our system will be able to quickly locate an image based on the memory of where and when the photo was taken. These event clusters are presented as a hierarchical tree, ordered by time and location, and will be further used for annotations and semantic correlation.

Semantic Correlations between Events (Jensen Galan)

Since it is very expensive for users to annotate every image in their collection, we make use of our event clustering to prompt the user to simply annotate the event. This method allows the annotation to be propagated to every image in the event, and also forces the user to think in a more abstract manner when annotating events than they would for a single image. These annotations are then stop-listed and fed into WordNet, a lexical reference system, generating a keyword hierarchy. Whenever a user selects an event in the GUI (represented by its timespan), our system will calculate and present the top 5 related events based on the keywords attributed to the events (the weighting algorithm is discussed below). The goal of our system is to allow the user to explore semantically correlated events, evoking connections between memories as they relive their life's meaningful events captured in images.

System Architecture (Chunyong Fei)

The goal of system architecture design is to facilitate the communications within different components of the system and simplify the individual work of each member in the team. Figure 1 shows the main class diagram of the Digital Album.

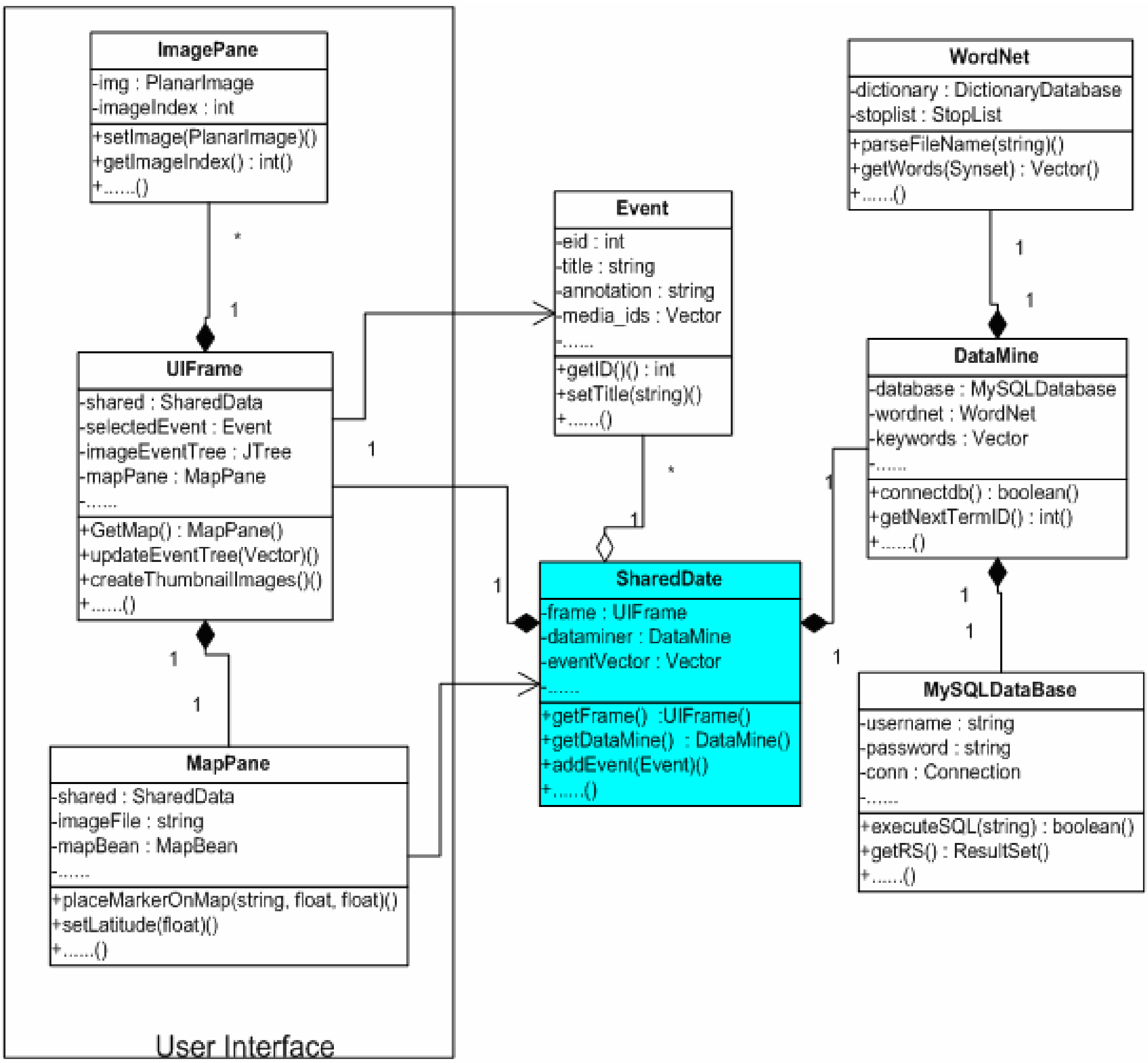


Figure 1: System Architecture Class Diagram

The SharedData class is essential for achieving the aforementioned characteristics of architecture. SharedDate consists of all the instances of major components of the system, i.e. UIFrame, DataMine, as well as a list (vector) of Events and a string array of image file name. By using the intermediate object – SharedData , every single component of the system gains the accessibility to other components through read and write interface functions defined in the SharedData. For example, the main UI component (UIFrame) is able to display all the imaged imported by user or stored in the database simply through reading the string array list in the SharedData. To get all the semantically related events of a selected event in the UI, all the UI component needs to do is to call the function getEventCorrelations (int _eid) in SharedData. Also, MapPane component gets the access to the DataMine and the associated database simply through GetDataMine() function in the SharedData.

As mentioned above, a well-defined architecture helps each team member fully focus on his/her individual work. In addition, the introduction of an intermediate class can improve the efficiency of communication and debugging.

Prior Work in the Area

Event Clustering (Jensen Galan)
In the extensive studies conducted by Rodden and his colleagues [20][21], when people were asked what features they desired in software for organizing digital photographs, the ability to arrange photographs according to events was the highest rated feature. In a subsequent study of digital photographers' practices [20], Rodden reported that "the most important use of digital photographs is to record holidays or other significant events." The importance of events is also evident when users search through their archived collections. Individuals will usually group photos into folders (directories) according to the events corresponding to their memories [2]. Even in searching for an individual photograph, users often first recall the event associated with the specific photo. While many researchers treat the duration of an event anywhere from an hour to a couple of weeks, we have chosen to segment temporal events much more finely. Since we are further clustering our events with respect to location, the event clusters our algorithm yields will still be presented in a meaningful and useful manner. For example, a couple on a vacation to Hawaii will visit a museum for a couple of hours, and then a few hours later attend a luau. Our algorithm will find the gap between these events, and cluster them separately with respect to time. However, using the location data, the events will appear in order under the Hawaii heading on the hierarchy. We feel these temporal events should be grouped separately on the lowest level of the hierarchy, but joined together on a higher level.

Semantic Correlations between Events (Jensen Galan)
While there have been a variety of projects dealing with semantic relationships of annotations for image retrieval [1][28][31], none have adequately dealt with semantic correlations between events to explore event clustering. These previous systems use WordNet to make connections between semantically related image annotations using the first synset and the related four levels of hypernym hierarchies. Our system makes use of all the available synsets for a noun along with their related hypernym, hyponym, meronym, holonym hierarchies. By incorporating "Primary and Secondary Annotations" where keywords will have different weights associated with them, a large pool of keywords allows our system to explore semantic correlations not apparent on first glance. Our goal is not only to correlate trivial semantics between events, but also to explore cause-effect type relationships.

Description of Work

User Interface (Chunyong Fei)
Since events are the fundamental organizer and data entrance of the Digital Album, users need to check into the images from different points of view and based on an event hierarchy. Thus, how to display the event-based media data intuitively and efficiently becomes very important. The design of user interface has the key features as follows:

1) The UI supports exploration of the temporal and spatial characteristics of events information.
2) The UI supports direct interactions between the users and the events. First, the UI should have an intuitive events editing function to support users to annotate and name a specific event. Second, the UI should support the function of refining the segmentation obtained by the automatic events clustering algorithm.
3) The UI enables users to look through image thumbnails associated with a selected event dynamically.
4) The UI has a zoom-in functionality that allows users to check full sized image.
5) Different visualization techniques are used. In Figure 2, for example, when users select an event, a group of markers indicating all the images associated with that event will be displayed in the map panel, and the map panel also supports zoom-in and zoom-out functionality that allows checking the map information at different resolutions.

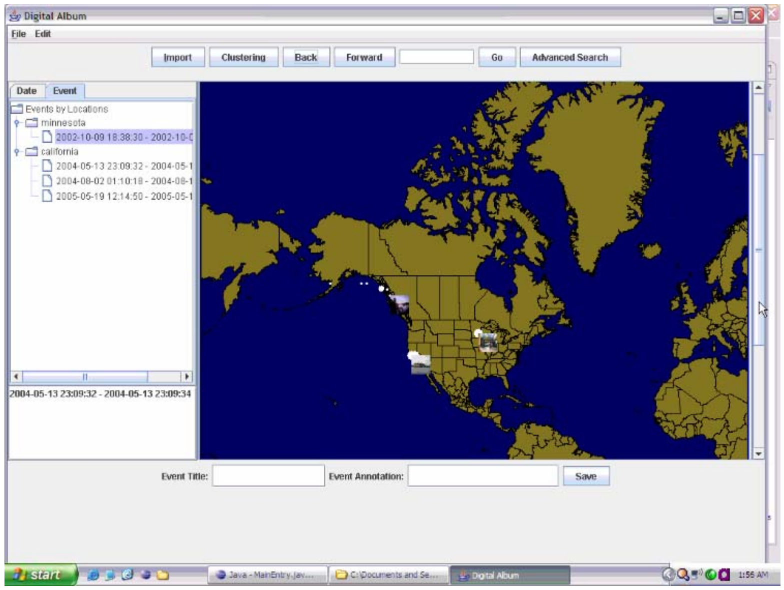


Figure 2: map view of the images

Image Manipulation with JAI (Chunyong Fei)

Java Advanced Imaging (JAI) supports a simple image processing model that can be readily used in the Digital Album application without a tremendous mechanical programming overhead. In JAI, images and image processing operations are defined as objects. JAI encapsulates image data formats and remote method invocations within a re-usable image data object, allowing creating image object from existing image file, to re-creating image at different size and resolution. Figure 3 shows a typical image manipulation in the Digital Album project.

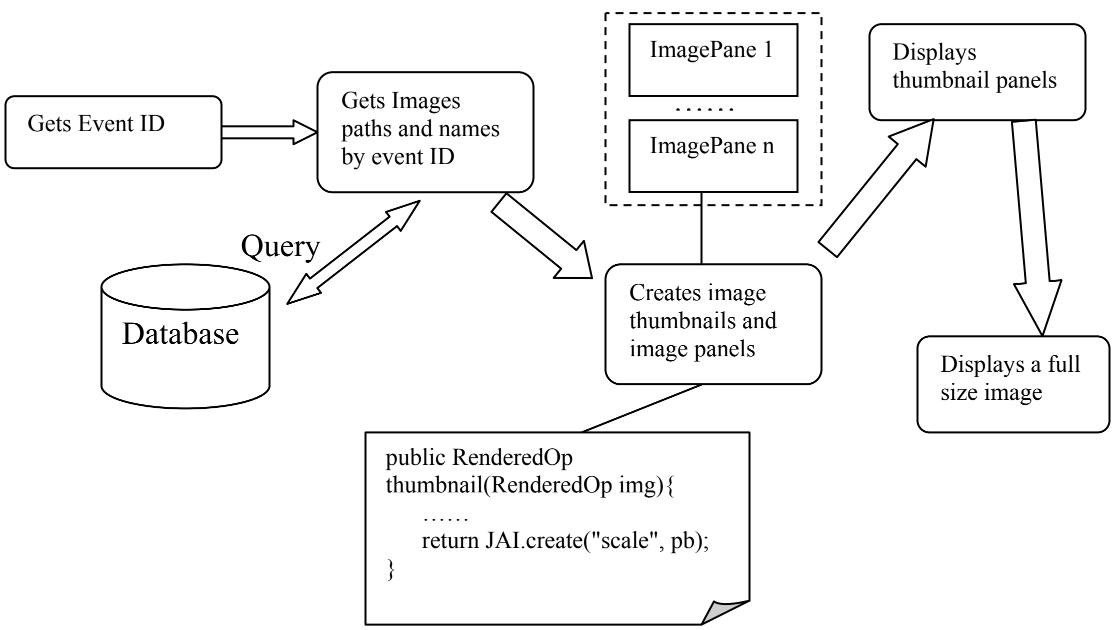


Figure 3: Displays images based on the selected event

When an event is selected (in the hierarchy tree), the program gets the ID of that event, and retrieves related image names and file paths. Thumbnails of the images are created by using JAI functions, and every thumbnail object is capsulated in a image panel object, all the image panels are displayed in a view area. After a specific thumbnail is clicked, the program evokes another JAI function to display the full size image in the same view area.

Event Exploring (Chunyong Fei)
Normally, people associate event with one specific location where they had taken photos. Thus, after the all the events are segmented by the clustering algorithm, we would like users to be able to quickly explore the locations of the events then the events that happened in that specific location. The events tree implementation of the Digital Album project uses a 3-level hierarchy. The first level of the hierarchy as shown in Figure 4 is the location level which is represented by the location name (i.e. city, county, state, etc.).

The second level is the event level. Because temporal information of a specific event, for example, the starting time and ending time, are also intuitively associated by people with event, the event level is represented by "starting time – ending time" of the event. The third level is the photo level, when user clicks an event node on the tree, all the thumbnail images which taken in that event will be displayed in the thumbnail view area. In addition, after a specific event is clicked by user, all the semantically correlated events (also represented by "starting time – ending time") will be displayed in a lower small panel.

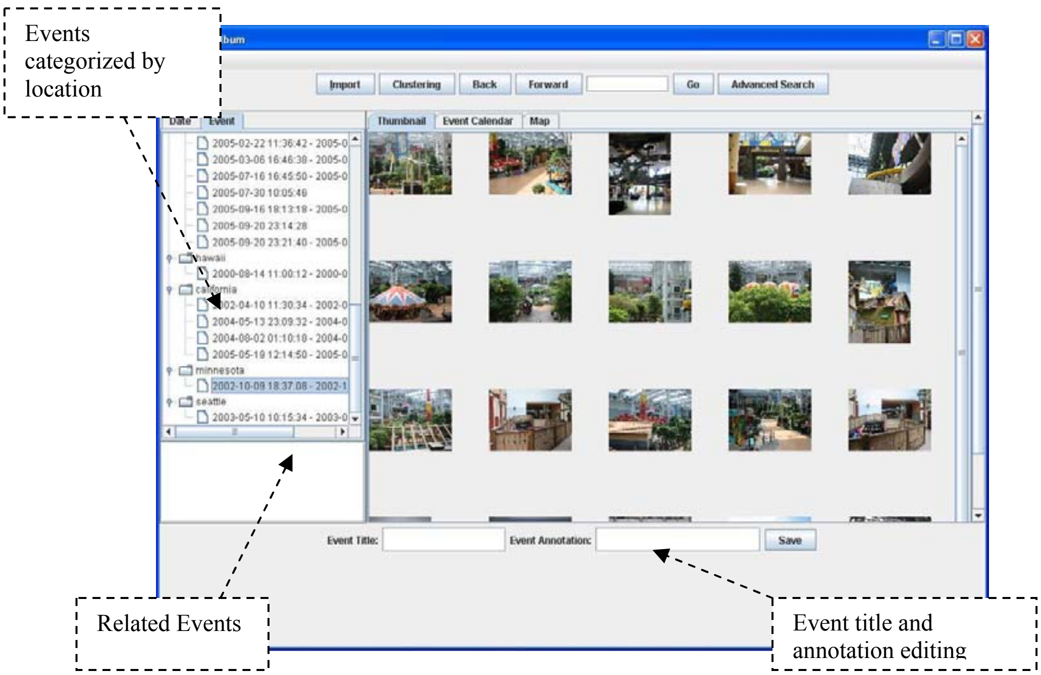


Figure 4: Event tree hierarchy

Event Clustering (Jensen Galan)
The automatic clustering of images into events is not a trivial problem, and many multimedia groups have recently focused their research into this problem [2][17][18]. Complex algorithms have been applied to the event-clustering problem, with some being more successful than others. How successful an event-clustering algorithm is depends heavily on the data set being used for testing, especially if a threshold is used. However, for a personal image collection, there are some distinct temporal qualities observed that can be applied to our development of an event clustering algorithm. The most important observation is the bursty nature of personal photos: an individual will snap a series of photos when an exciting, meaningful event is occurring in their lives, followed by a temporal gap until the next "life-event" comes along. Using this information, we have decided to apply a sliding window algorithm to the temporal data available in our image files.

In order to automatically cluster our image collection into events, the date and time information is first extracted from the image metadata. The images are then sorted chronologically using the Quicksort algorithm, since a personal image collection can easily reach thousands (even tens of thousands) of images. Next, the temporal difference (Δti) between adjacent images is calculated, along with the average temporal difference (Δtavg) for adjacent images within the sliding window. For our test data, experiments showed that a window size of 5 worked nicely (this could easily be changed for other sets of data). If the temporal difference (Δti) for two adjacent images is greater than the average temporal difference (Δtavg) for adjacent images within the sliding window, our algorithm marks a boundary for a new event. Otherwise, the current image being analyzed is appended to the existing event I.D.

While the sliding window algorithm we implemented may not be the most exotic or glamorous clustering algorithm, the results that our tests returned were surprisingly accurate. The algorithm implemented over-segmented some events (ie: 1 event was clustered into 2); however, by further combining the clusters generated by temporal information with spatial information, we were able organize and present this information to the user in a useful way. The hierarchical tree implemented (discussed in the GUI section), allows users to navigate through their image collection by recalling not only when, but also where the event occurred. Designing an effective clustering algorithm was not the focus of our research, which could easily be extended to a term project on its own. Our goal was to implement a decent clustering algorithm so that we could focus on exploring semantic correlations between these events.

Semantic Correlations between Events (Jensen Galan)

Once our image collection has been clustered using temporal information and further grouped using spatial information, the user of our system is prompted to annotate the event selected. It is well-known that users will not annotate every image in a collection, nor can we depend on filenames to provide annotations for our system (most images end up with filenames such as DSC00842.jpg which reveal absolutely nothing about the content). Therefore, we make use of the clustered events so that a single annotation can be applied to all images occurring in the event. To support developing semantically meaningful event annotations, a local implementation of WordNet (JWord) was used. This saves the application from having to connect to the Internet in order to make use of the extensive WordNet database, significantly reducing overhead. The annotation is then stop-listed to remove words deemed unimportant and fed into WordNet, a lexical reference system that links semantic relationships. Our system uses all synonym sets of WordNet's first two levels of its hierarchy, and makes use of all the various relationships WordNet yields (Hypernym/Hyponym is-a relationship and Meronym/Holonym part-of relationship). When inserting our annotation data into the database, we devised a weighting scheme which classifies Primary Annotations as anything the user has entered, and Secondary Annotations as any keywords generated by WordNet. Primary Annotations receive a weight of 10, while Secondary Annotations receive a weight of 5.

The weighting scheme is necessary to rank the matched terms when querying the database for semantically related events. When a user selects an event from our hierarchy presented in the interface, our weighting algorithm assigns a rank for every matched term present in the database as follows:

(cureid term weight=10) matches (related eid term weight=10) -> add weight 10
(cureid term weight=10) matches (related eid term weight=5) -> add weight 5
(cureid term weight=5) matches (related eid term weight=10) -> add weight 5
(cureid term weight=5) matches (related eid term weight=5) -> add weight 1

For each event I.D. (eid), the cross-referenced weights are summed, giving the event a ranking relative to its semantic correlation with regard to the queried event. The five events with the highest rankings are then presented to the user as correlated events, and thumbnails for all of the related events are displayed on the image pane of the GUI. Grouping events in this manner allows the presentation of semantic correlations, either trivial or cause-effect, between events that emerge only after exploring the linguistic relationships of the annotated events across space and time.

Image Metadata (Scott Bishop)

Two attributes of an image were utilized for presenting images in an organized hierarchical structure. These were location and date time. These attributes were assumed to be in the Exif header [12] and a third attribute, the location name, was garnered by reverse geo-coding. Reverse geo-coding is the process of looking up a named location from the latitude and longitude. The latitude and longitude specify a position on the earth's surface. Latitude is defined by the horizontal lines on a globe and longitude is given by the vertical lines. Latitude's natural start position is the earth's equator with latitude of 0 degrees. Longitude has an arbitrary starting position and is located at the Prime Meridian in the Atlantic Ocean. Thus by using a cross reference table of known location names a meaningful name can be found for any given pair of coordinates. The only limitation in reverse geo-coding is the size of the reference table. For reverse geo-coding SPaRC assumes that a picture was taken in the United States.

The purpose of extracting metadata associated with an image is to produce attributes that can be used to establish a point of reference for past events. Events in the context of SPaRC are defined as being at a unique location during a given period of time. Two physical points of reference have been established using image header metadata. The spatial, or location reference, and the temporal, or date time reference. The motivation behind gathering this information is to provide SPaRC with enough points of reference in order for meaningful relationships to be made between images that may be spatially and temporally isolated.

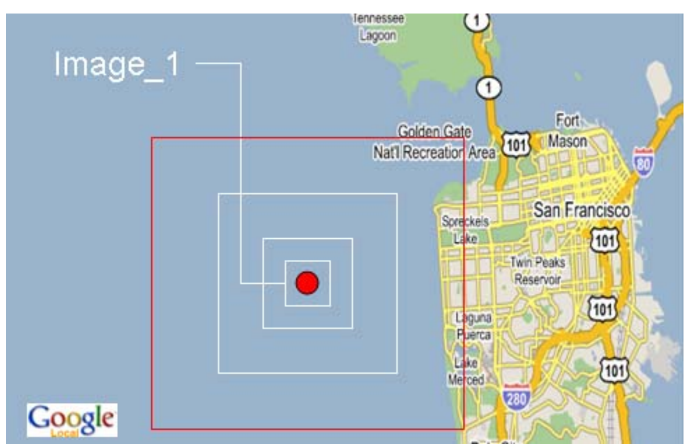
In order to perform such a correlation beyond space and time annotations were used to label events. SPaRC's sliding window algorithm (SWA) segments pictures into date time groups given a predefined window size. Each date time group can be thought of as an event spanning the range from the earliest image to the most recent image in the group. The date time groups established by the SWA are then organized hierarchically by their respective location. Once these two operations are performed an annotation can be added to further define a date time event.

The annotation is the third point of reference for SPaRC. Annotations introduce the semantic reference for finding related events among possibly unrelated spatially and temporally grouped images. The semantic meaning in SPaRC is a direct result of the WordNet Lexical Database. SPaRK uses WordNet nouns to associate annotations from one event to another. Noun associations are well defined by WordNet and are associated with lexical inheritance. An example of inheritance is easily described as an 'is a' relationship. So the word chair 'is a' piece of furniture or a calico 'is a' type of cat. Thus by annotating a date time event SPaRC can associate other date time events based on lexical inheritance rules [16].

Bounding-Box Reverse Geocoding (Scott Bishop)

Reverse geo-coding is the process of locating a named location from a latitude and longitude coordinate pair. SPaRC is concerned with providing a meaningful name and thus chooses city names to represent an image's location name. The process is straight forward given a large enough database to represent all the meaningful places located on earth. Since this is unreasonable for the scope of this application reverse geo-coding was performed only on locations in the United States. The Bounding-Box procedure to produce a meaningful location name from a latitude and longitude is best represented by Example 1. In this example what location name should be used for Image_1? San Francisco, Lake Merced, or the Pacific Ocean. This question is directly related to the data that is available in the cross reference table and what location name is important in the respective application.

In order to find a location the latitude and longitude are incremented a fixed amount North, West, East, and South until a location name is returned. This name is then used as the location name for the image and provides the named location used in the event hierarchy creation.



Example 1. Simple bounding box location name look-up.

Naaman et al have provided a facility to name a location based on the highest number of hits from a search engine query [17]. Such a feature may be useful but in developing SPaRC we were not as concerned with precisely defining location and date time events given that the semantic point of reference would be more useful in organizing related events that may be spatially and temporally unrelated.

Map Panel (Scott Bishop)

The spatial component of images is most easily viewed graphically. Images that have location data can then be positioned on a map or grouped according to location and a marker can be used in its respective position. SPaRC's assumption that location data is available in every image is thus exploited to provide another feature in the interface for image organization. SPaRC uses an open source mapping component called OpenMap for the map interface. OpenMap is a 2-Dimensional map that uses the WGS 1984 [29] coordinate system and provides a rich user library capable of displaying several objects including text, geometric primitives, and raster images [19].

A useful feature of SPaRC displayed in Figure 5 is the use of a log scaled circular marker to represent images by frequency at many locations. This is done in order to avoid map clutter that can occur if images are all positioned at one location or if there are a large number of images. The marker's diameter is determined by the number of images at a given location and is scaled logarithmically [27]. The map gives a complete view of the images over the spatial domain and provides several features to enhance a user's ability to navigate and locate images.

These features include zoom-in, zoom-out, panning, and named location retrieval output to the console by clicking the map. Image thumbnails are also placed on the map when a user clicks an image in the preview pane of the application. Political boundaries are another feature added to the map that are beneficial to a user in identifying cities, states, and countries of the world while navigating on the map panel.

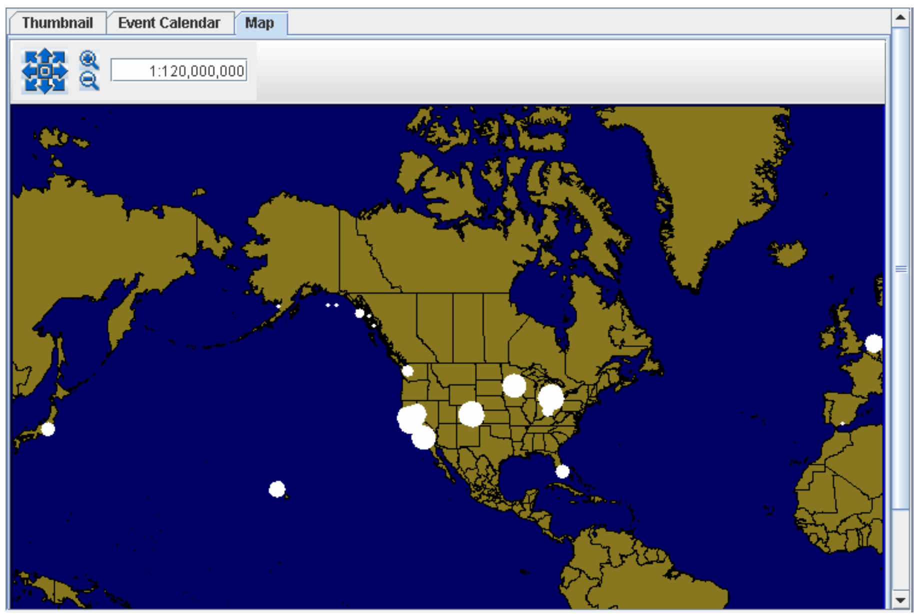


Figure 5. SPaRC's map panel showing image markers.

Hierarchical Grouping by (Scott Bishop)

SPaRC's focus was to combine the hierarchical nature of spatial and temporal data with the semantic relationships of WordNet. Given these two points of reference it was determined that the physical properties of space and time were sufficient to group events in order to further make use of the third semantic point of reference.

As such we tested two simple date time segmentation algorithms; nearest neighbor search [13] and a Sliding Window Algorithm (SWA). Both algorithms successfully segmented the images into date time groups but the SWA was more efficient at over segmentation that was necessary for the first level of image organization. There exist several clustering algorithms that can perform searches in high dimension space using a distance as the metric of similarity as can be found in [2][11][13][30][31]. This approach albeit useful in certain situations is not required given the physicality of the first two points of reference and the semantic mental lexicon of the third reference that is an abstraction of human language [16].

Database Design (Stacie Riddle)

The design for this Digital Album database was created with object type flexibility, annotation flexibility, and organization flexibility in mind. For flexibility with object types, the database's central object, the Media Object, was designed generically such that the application could be expanded to hold objects of types other than images such as sound files and html documents. Figure 6 below attempts to depict the main database objects and their relationships to each other.

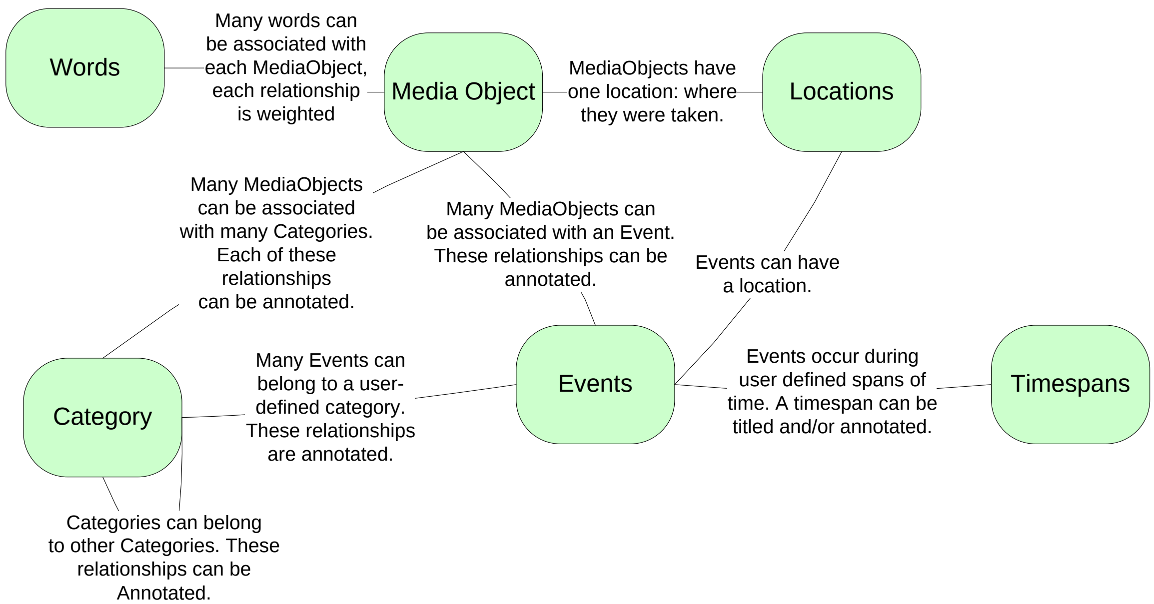


Figure 6: Object Relationship Diagram

In addition to the flexibility for the actual types of Media Object to be held in the database, the structure for organizing those objects was designed to be flexible and useful for a variety of users. To accomplish this, three types of organizational containers were identified: Events, Categories, and Timespans. Events are the primary containers. Events should have both time and location data associated with them. Ideally, all Media Objects in the database will have one "parent" event. This "parent" event should be the most atomic event possible, containing no other sub-events. An example of a complicated Event structure would be a European Vacation. In this database, this event would ideally contain other events such as "Trip to Berlin" and "Visit in Venice".

Categories are defined in the context of this database to represent semantic, possibly abstract, correlations between Media Objects and Events. Unlike Events, Categories are time and location independent. Like Events, Categories are allowed to have a hierarchical structure defined by the user. For example, there could be a Category of "Vacations", with sub-Categories such as "Camping" and "Cruises". Eventually these Categories, when envisioned as a tree, will ultimately have specific Events at the leaves. It is also possible, in this implementation, for individual Media Objects to be associated with Categories other than those in their immediate, primary, hierarchical structure. An example of where this would be appropriate is shown in Figure 2 with the "Forest Bride" Media Object. This photo was supposedly taken of a wedding occurring in Muir Woods while the user was backpacking. The user later decides he/she would like to categorize all images that depict weddings in a Category named "Weddings", which, as can be seen in the diagram, also contains wedding Events the user may have attended.

Timespans are the third type of container structure in the database design. These are intended to allow the user to label periods of time with semantic meaning, such as the college they were enrolled at, the city or country they were living in, or the years they spent in the military, just to name a few possibilities. In the database, associations with Timespans should never be explicit. They are designed to be inferred from the time data assigned to Event objects. Events occur within Timespans. It is hoped that this structure could aid in the derivation of cause and effect relationships between Events. For example, Timespans labeled with education and employment could be related to the pictures the user took of a new house or car. The possibilities are too numerous to list and definitely deserve further investigation.

To accomplish flexibility with the annotations, columns were created in the database tables to hold user annotations of individual Media Objects, the Events those Media Objects belong to, the Categories those Events (and Media Objects or other Categories) belong to, as well as the Timespans the user defined. The design does not stop there. The relationships between these objects can also be annotated by the user.

For an example of relationship annotation, consider an instance where the user has an image of a friend Bill at a birthday party. The user may have already annotated the birthday party event, and possibly even the image of Bill itself. Now, consider the user has created a category to hold all images that are somehow related to Bill. For example, there could be other pictures detailing Bill's favorite car, his wife and his kids. When the user adds the aforementioned image of Bill at the birthday party to this new category of images, the interface would ideally prompt the user to explain why this image belongs to the category they are adding it to. Hopefully the user would respond with something meaningful, such as "Because this is an image of Bill". This is an example of the annotation of the relationship between that image and that category, rather than an annotation of the image itself or the category it is being added to. From this, the hope is that as the user continues to customize the album, more and more meaningful, multi-dimensional, content will be added regarding the images themselves.

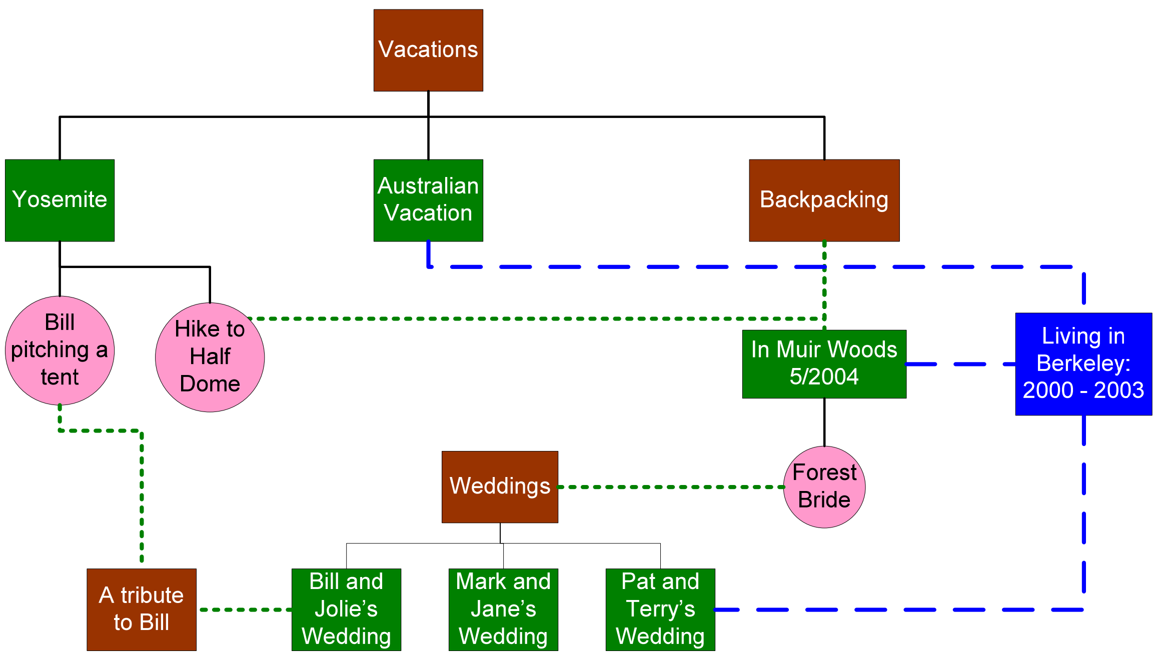


Figure 7: Example Instance of Relationships between Media Objects, Events, Categories, and Timespans

Database Implementation (Stacie Riddle)

The database for the Digital Album utilizes a relational design which was implemented using MySQL. The Category, Event, and Timespan containers were found to have more similarities than differences, where the main differences are simply the presence or absence of time and/or location data. Other differences are largely based in their semantic meanings. As such, these different types of organizational containers were implemented as the same physical table in the database.

As mentioned before, the annotations are captured into their respective fields in the database tables and remain there unaltered by the system. Separate tables were setup for the processing of the annotations: the

WordList and WordFrequency tables. The WordList table is intended to capture all words that are related to the album into the album's database; essentially creating a subset of WordNet driven by user annotations. The WordFrequency table is then used to pair words with all annotatable objects, with a frequency field meant for specifying the weight of that association. Refer to Figure 3 for the actual database schema.

The integration of the Java application interface with the MySQL database implementation is largely achieved via the DataMine class file ("DataMine.java").

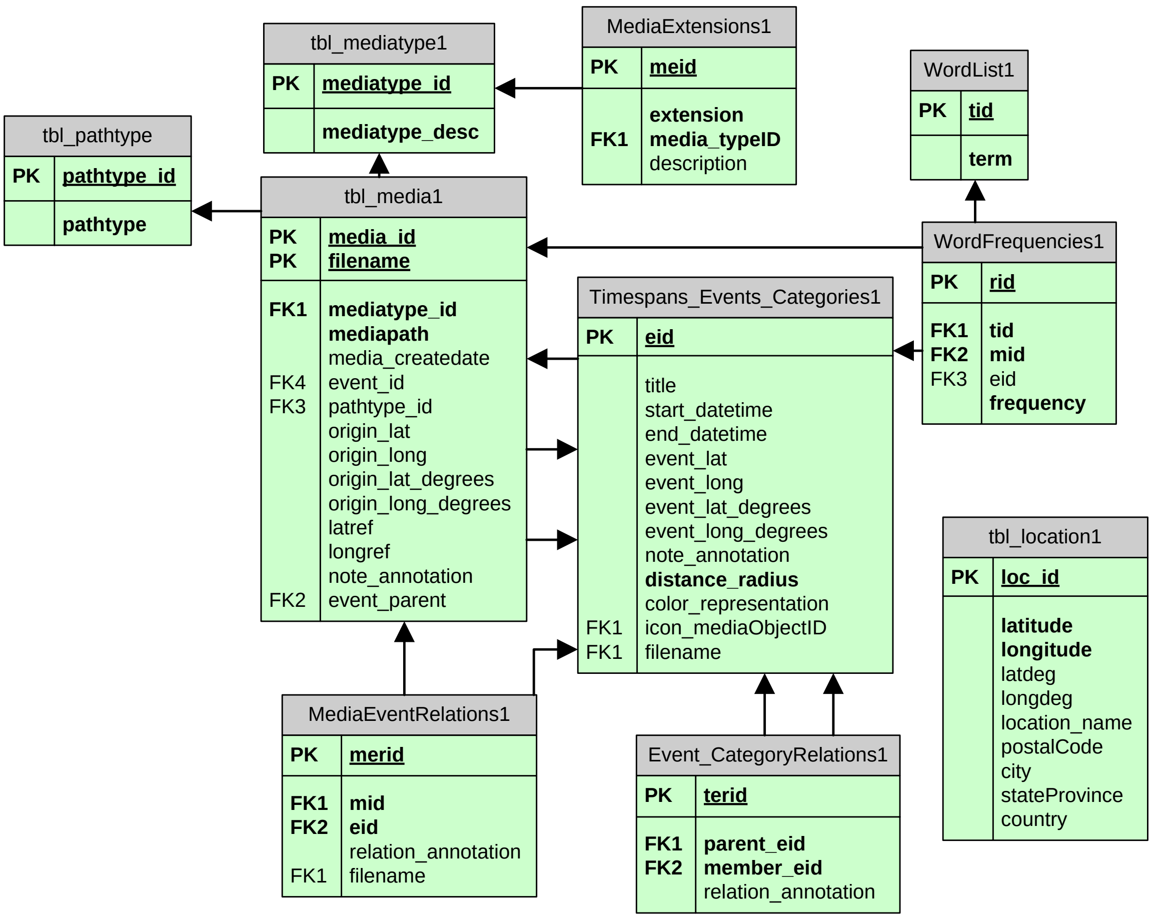


Figure 8: Database Schema Diagram

<u>Individual Contribution</u> (Stacie Riddle)

Stacie began by researching various possible database technologies for use in implementing this project and communicating those to the group. The relational SQL implementation was decided upon because of overall existing familiarity with the technology. She then set about researching the feasibility of incorporating sound into the digital album, which was later refined to speech analysis research. Please refer to the Midterm Project Report for details on the results of this research as it relates to this project. With regards to sound, the group decided to focus its energy on refining the album's ability to derive semantic relationships amongst images held within the database, rather than dilute the project with the addition of functionality for another object type.

With this decision, Stacie focused her energy on expanding and refining the database design in the hopes of facilitating these goals. Stacie also spent a considerable amount of time and effort to gather test data for the album's deeper semantic functions.

In the scope of the database, Stacie designed the database, provided updated diagrams for each phase the design went through, as well as SQL scripts to be used to update the database and load it with initial global data (such as the file extensions for various types of Media). Stacie also diagramed the abstract object relationships being implemented by the database, a simplification of which is provided in this paper. In addition to providing scripts and diagrams with each database revision, Stacie wrote and updated some of the Java code for providing functional access to the database. In particular, she wrote the following functions located in the DataMine class (please see code for details):

- public void insertKeyword(int event_id, String kw, int type)
- public int getNextTermID()
- public ResultSet getRelatedEventTerms(int event_id)
- public ResultSet getEventTerms(int event_id)
- public Event getEventFromDB(int event_id)
- public int addEventToDB(Event anEvent)
- public int getNextEventId()
- public int getNextMediaId()
- public Vector getMediaIDsForEvent(int eid)
- public void addMediaIDsToEvent(Vector media_ids, int event_id)
- public boolean eventIdExists(int event_id)
- public boolean mediaIdExists(int media_id)
- public void remMediaIDsFromEvent(Vector media_ids, int event_id)

In the scope of the test data, Stacie collected and organized hundreds of images, looking up realistic latitude and longitude data for each Event collection. She then wrote an application which generated a SQL script (dbUpload.sql) for uploading the annotated images into the main Media Object table. This application was then modified to also generate a batch file of exif commands to be used for tagging all test images with the approximated latitude and longitude data. She created and circulates a website (see Appendix B) for eliciting annotations for the test album data from individuals outside of the group. These annotations were collected into a database Stacie setup and then extracted and converted into SQL scripts by an application she wrote.

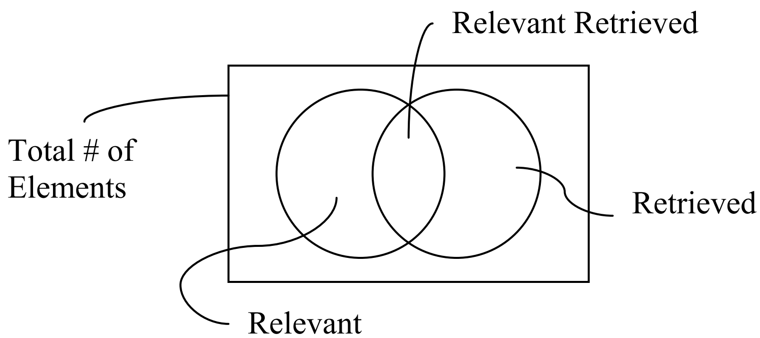
For the final paper, Stacie wrote the Database Design and Implementation sections, as well as the sections detailing the data used to test the album.

Experimental Evaluations

Test Data (Stacie Riddle)
The images were collected from a pool of "real" personal collections and as such, retained their original file names. The events included trips to two different cities in Germany on the same vacation, other trips to European countries on widely varying dates, many events in and around San Francisco (assumed to be the home city of this instance of photo album user), as well as trips to other states within the United States. See Appendix A for specific data collected for these events. The latitude and longitude of each event was looked up using The Getty Thesaurus of Geographic Names Online, located at http://www.getty.edu/research/conducting_research/vocabularies/tgn/.
The images were then annotated with exif headers utilizing a free utility provided by Hugsan Pty. Ltd. at http://www.hugsan.com/EXIFutils/html/downloads.html. This was done such that the album's import and auto-clustering functionality could be tested.
Next, a website was created for gathering event annotations from people not directly related to the project. Co-workers, family members and friends of the team members were asked to contribute these annotations, which were collected into a database and then used to generate SQL scripts which could be used to load the database with each "album instance" for testing. Annotation providers were prompted with generic event titles and a sample of up to 6 images per event and told to enter their annotations in the field below the event. Subjects were also told to feel free to skip annotating an event if they would not normally annotate such an event in their own Digital Album. As such, some test annotations do contain null values, whereas others contain none. See Appendix B for specifics regarding this website.

Event Clustering (Jensen Galen)
In order to measure the effectiveness of the sliding window clustering algorithm, the standard measures of precision and recall were used. Precision is defined as the fraction of retrieved documents that are relevant, while recall the fraction of relevant documents that are retrieved, given by the following formulas:

Precision = Relevant Retrieved / Total Retrieved
Recall = Relevant Retrieved / Total Relevant



Fundamental to the development of our temporal clustering algorithm is the understanding that we desire the over-segmentation of the events based on time, since we are grouping this bottom-level together with spatial information to present the events in a meaningful way. With this assumption in mind, we ran the sliding window algorithm on a real-life image collection consisting of about 400 images. A subjective analysis of the collection resulted in a relevant event measurement of 37 events. Our sliding window clustering algorithm yielded 39 total events, 36 of which were relevant. Thus, the results of the standard precision and recall measurements are:

Precision = 36 / 39 = 0.923
Recall = 36 / 37 = 0.973

The discrepancy was a result of the algorithm over-segmenting the data by three events (ie: 3 relevant events were clustered into 6). Since we our system further groups these temporal clusters using spatial information, this was acceptable for our purposes. The error was caused by the algorithm missing one relevant event (ie: 1 event should have been clustered into 2).

Semantic Correlations between Events
Quantitatively measuring a subjective notion such as the semantic meaning of events is not possible; we, therefore, turn our experimental evaluations to a case-study to gauge the effectiveness of semantic correlations between events.

Case 1 – A Trivial Correlation:
In the first scenario, a sports fan who has traveled across the United States visiting stadiums where his/her favorite teams play is taken into consideration. The events have been clustered into 4 separate events, all in different timespans and locations (Oakland, California; Indianapolis, Indiana; Miami, Florida; Detroit, Michigan). Once the events had been clustered, the user annotated these events with the obvious annotation corresponding to the images – the sports being played in each event (ie: baseball, football, basketball, hockey). Once the events had been annotated, the user selected the "football" event again, causing our system to display the thumbnails of all related events (figure 9).

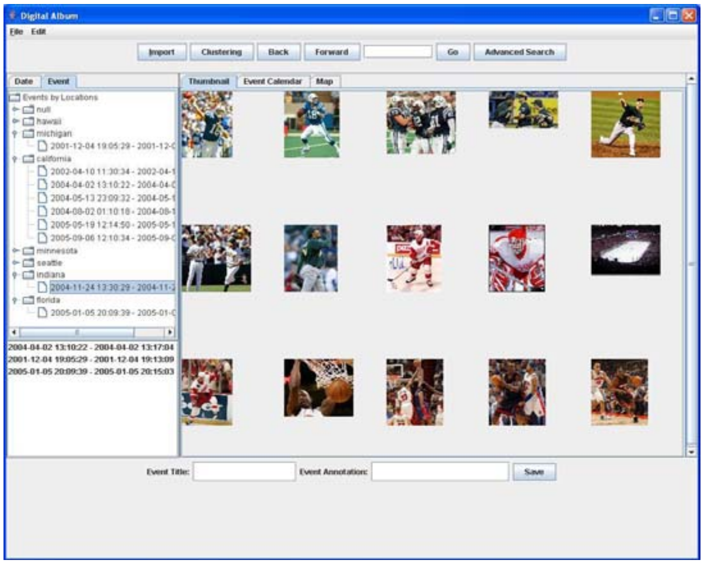


Figure 9

In the figure on the left, we can observe our algorithm to rank semantic correlations in action. The user has selected the Indiana event in the upper pane (the hierarchy) causing the related events to be calculated and presented in the bottom pane, ranked in order of their relevance. Because the user-entered event annotations have been fed into WordNet, our correlation algorithm returns the "baseball" event first, since it is also a "field sport" played with a "ball". The "hockey" event is listed next, since this sport and football are "contact sports". Finally, the "basketball" event is listed, since both sports are played with a "ball." This scenario illustrates the usefulness in populating our database with all relationships available from WordNet (Hypernym/Hyponym is-a relationship and Meronym/Holonym part-of relationship). While this semantic correlation can be considered trivial, it is a good example of the kind of relationships our system can connect across time and space using the linguistic relationships of WordNet.

Case 2- A Cause-Effect Correlation:
In this scenario, we explore a cause-effect type correlation that our system presents. Again we have 4 events separated across time and space: a wedding in Hawaii, a couple's first house in California, a family reunion in Seattle, and a child's birthday party in California, captured in chronological order respectively. After our test user annotated the temporally clustered events, the family reunion event was selected causing our system to calculate and display the related events (figure 10).

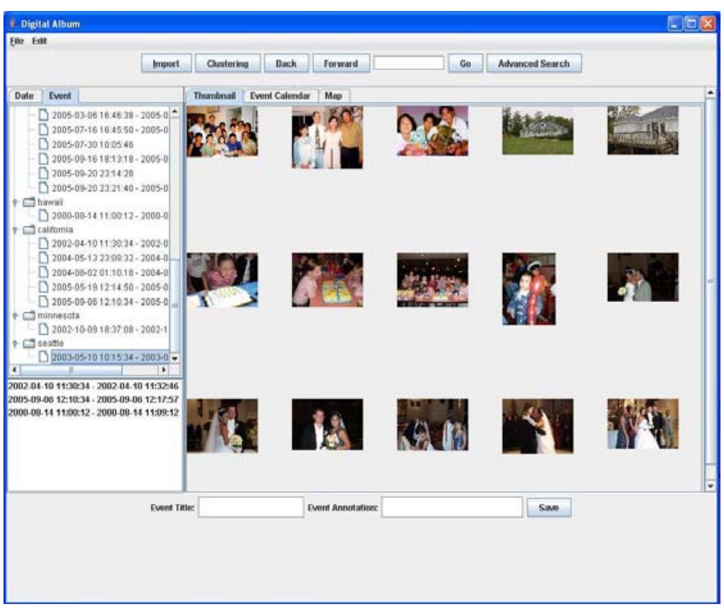


Figure 10

The figure illustrates how our system has connected these semantically correlated events in a meaningful way. All thumbnails for the related events have been displayed, and analyzing the timespans displayed for the events, a deeper cause-effect relationship begins to emerge. We can observe that the "wedding" event happened chronologically first, which led to the couple buying their first "house" almost 2 years later. The next year, they attended a "family" reunion in Seattle. Finally, if these previous events had not happened, they would not have had a "child" together and celebrated her birthday in 2005. By clustering our events temporally, grouping them spatially, and correlating them semantically, our system allows users to explore the deeper cause-effect correlations usually hidden in a scattered image collection. Not only do these relationships allow the system to present the images in a meaningful way, but also force the user to take an introspective look at the memories captured in their images.

This type of cause-effect relationship was also observed in the correlating of events represented as images of an office party at an employee's "job" and his commencement ceremony marking his graduation from "university." Temporally, the events were almost 5 years apart and in different parts of California. However, the linguistic relationships yielded by WordNet allowed these events to be connected and presented for the user to subjectively draw semantic meaning from. The images of the office party juxtaposed with images of graduating from university illustrate how our system can be used to find events causing others from a semantic perspective.

Case 3 – A False Positive/False Negative Correlation:
The final scenario considered was one in which our system not only did not correlate 2 events which were connected as a cause-effect relationship, but also incorrectly correlated 2 events which had no apparent relation. This author imported images of his first "computer," hoping that our system would find a connection between this event and images of a Graduate Showcase where a piece of music software was presented. Granted that this connection could have been found had both events been annotated with "computer;" however, this was not the case. One event was annotated with "computer" and the other with "college." While the obvious connection between "college," "university," and "job" was correctly identified (see previous case), the event of buying my first computer causing me to attend Graduate school to study Computer Science was completely missed. Not only was this correlation missed, the "computer" event was found to be related to images taken on a "party bus." The word "bus" was generated as a keyword for both events, though the meaning of this term is different is each context. Currently, WordNet does not contain capabilities to get past this polysemy limitation.

Conclusion

High Level (Scott Bishop)
Three points of reference are used by SPaRC to automatically organize images; location, date time, and event annotations. The location and date time are physical properties and are grouped in semi-rigid hierarchies. Date time is the base level grouping that provides the initial event identification. Location is used to further organize the images into meaningful geographic locations for display in the user interface. The final and most interesting component in SPaRC are the event annotations. The annotations allow for related events to be found based on WordNets noun hierarchies that "reflects a different way of categorizing experience" [16]. This categorization allows for related events to be found across space and time giving true meaning to SPaRC's name: Similar Picture and Relationship Constructor.

SPaRC's usefulness depends on how well it organizes information for the user. In this work we have presented a physical and semantic approach to automatically solve the problem of image organization and retreival. The physical location, date, and semantic components enhanced by WordNet have been introduced into the system in order to better understand the underlying data. The purpose of this understanding is to present the data to the user in a meaningful way. All of these approaches while not unique in their own right are a powerful combination which produces an application that successfully finds relationships between images across spatial, temporal and lexical domains

Thus a new frame of reference for organizing and retrieving images has been created by combining physical properties with psycholinguistic semantic memory features using WordNet [15]. SPaRC takes advantage of this information to increase a user's access to these media providing a meaningful mechanism for viewing large sets of images.

Event Clustering (Jensen Galan)
While our experimental evaluations show that our simple sliding window algorithm achieved usable results for our purposes, improvements to this algorithm are possible to fine-tune for more accurate clustering. One possibility to tone down the over-segmentation (which was desirable for this project) is to make an additional pass over the ordered image timestamps with a larger window size in order to merge together multiple events that should have been a single event. In our experiments, larger window sizes (i.e.: winSize = 10) tended to yield a lower number of events. Location data could also be incorporated into this second pass to identify candidates in the same general vicinity that should be merged together. Once again, our

goal for this project was just to design a usable algorithm that would allow us to focus on the more interesting work of finding semantic correlations between these event clusters.

Semantic Correlations between Events (Jensen Galan)

Overall, our group was able to develop a system to explore the semantic correlations between annotated events. By first clustering an image collection based on temporal information, and then calculating related events based on the WordNet hierarchy, cause-effect relationships began to emerge when the related events were displayed in conjunction with their timespans. By assigning each related event a weighted ranking score dependent on the Primary or Secondary annotations matched, we were able to present all related events in this order to the user in the image pane. These semantic correlations will hopefully spark the user's memory to make their own subjective connections. Further work should include the manner in which these correlations are presented to the user. In its current state, our system only displays the related event thumbnails according to their ranking scores, with the selected event being displayed first. The option should also be available to display these related events in chronological order, as this will allow cause-effect relationships to move to the forefront. In addition, currently our implementation of the WordNet hierarchy only makes use of "nouns." In order to be able to fully explore cause-effect relationships, and to add functionality that supports real-world annotations, "verb" synsets from the WordNet hierarchy must be integrated into our system.

Some form of user-feedback should also be incorporated into our system. Ideas for this functionality include transparently capturing the user's click on any of the calculated related events in the bottom pane of the hierarchy. Since this pane lists possible related events to the event already selected, any event that the user selects obviously has some connection to the original event (no matter how subjective). This information would also allow our system to capture correlations that are just not possible with temporal/spatial information and linguistic relationships. This would bring us one step closer to creating a system for managing people's memories.

References

[1] Aslandogan, Y. Alp, Thier, Chuck, Yu, Clement T., Zou, Jon, Rishe, Naphtali. "Using Semantic Contents and WordNet in Image Retrieval," Department of EECS, University of Illinois at Chicago. 1997. ACM-0-89791-836.

[2] Cooper, Matthew, Foote, Jonathan, Girgensohn, Andreas, Wilcox, Lynn. "Temporal Event Clustering for Digital Photo Collections." FX Palo Alto Laboratory. ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 1, No. 3, August 2005, pages 269-288.

[11] Hare, Jonathon S., Lewis, Paul H. "On Image Retrieval Using Salient Regions with Vector-Spaces and Latent Semantics, Lecture Notes in Computer Science", Volume 3568, Aug 2005, Pages 540 - 549

[12] Japan Electronics and Information Technology Industries Association. "Exchangeable image file format for digital still cameras: Exif Version 2.2", JEITA CP-3451, 4/2002. downloaded 10/2/05 from http://www.exif.org/.

[13] Larose, T. Daniel. "Discovering Knowledge in Data", Wiley Interscience. 2005.

[15] Miller, A. George. "Nouns in WordNet: A Lexical Inheritance System". Revised August 1993.

[16] Miller, A. George, Beckwith, Richard, Fellbaum, Christiane, Gross, Derek, and Miller, Katherine. "Introduction to WordNet: An On-line Lexical Database", revised August 1993.

[17] Naaman, Mor, Song, Yee Jiun, Paepcke, Andreas , Garcia-Molina, Hector. "Automatic. Organization for Digital Photographs with Geographic Coordinates", JCDL'04, June 7–11, 2004. Downloaded 10/11/05 from ACM Digital Library.

[19] OpenMap. Downloaded 9/24/05 from http://openmap.bbn.com/.

[18] Parsons, Lance, Haque, Ehtesham, Liu, Huan. "Subspace Clustering for High Dimensional Data : A Review." Sigkdd Explorations, Volume 6, Issue 1, pages 90-105.

[20] Rodden. K. "Evaluating similarity-based visualizations as interfaces for image browsing". Ph.D. thesis, University of Cambridge, 2002.

[21] Rodden. K. 2002 and Wood. K. 2003 "How do people manage their digital photographs?" In proceeding of the ACM Conf. on Human factors in computing systems (CHI). ACM, New York, 409-416.

[22] Samet, Hanan. "The Design and Analysis of Spatial Data Structures", Addison-Wesley Publishing Co. 1990.

[23] Singh, R., Knickmeyer, R. L.,Gupta, P., and Jain, R. "Designing Experiential Environments for Management of Personal Multimedia", Proceedings of the 12th annual ACM international conference on Multimedia, P:496 – 499, 2004. Downloaded 9/24/05 from ACM Digital Library.

[27] Toyama, Kentaro, Logan,Ron, and Roseway, Asta."Geographic Location Tags on Digital Images", Microsoft Research. ACM 2003. Downloaded 10/4/05 from ACM Digital Library.

[28] L. Wenyin, S. Dumais, Y. Sun, H. Zhang, M. Czerwinski, B. Field, "Semi-Automatic Image Annotation," Proceedings of the eighth ACM International Conference on Multimedia, October 2000.

[29] World Geographic System 1984. www.wgs1984.com

[30] Zaiane, O.R., Lee ,Chi-Hoon, "Clustering spatial data in the presence of obstacles: a density-based approach", Database Engineering and Applications Symposium, 2002. Proceedings. International 17-19 July 2002 Page(s):214 – 223. Downloaded on 11/5/05 from ieeexplore.

[31] B. Zambrano, R. Singh, B. Bhattarai, "Using Linguistic Models for Image Retrieval," San Francisco State University.

[32] Zhou, Changching, Ludford, Pamela, Frankowski, Dan, Terveen, Loren. "An Experiment in Discovering Personlly Meaningful Places from Location Data", CHI 2005 | Late Breaking Results: Short Papers. April 2-7, 2005, Portland, OR.