6. A Suburban Ontology

Knowledge representation is slave to the interface and servant to the task. No knowledge can be used if it can not be entered and displayed. Furthermore, knowledge is not necessary unless it can be used to accomplish a task.

After the video source material was viewed and edited, a draft ontology was developed. This ontology was refined as it was used to log video. Its design is strongly influenced by the specific video source material being used. This chapter will document the current state of the representation, issues that arise, and suggestions for future development.

The base ontology is divided into ten categories: basic segment data (including the segment name, in and out points), people, people in audio only, locations, event, actions, significant objects, date, camera properties, and subjective properties. This base representation can be extended by the user. For example, the system provides a location category "vacation-spot." The user may add "The-Grand-Canyon" as an instance of vacation spot. Spatial representation will be discussed further in section 6.3.

6.1 People

6.1.1 Extending the Representation: New Buttons and Object Editors

The system begins with an empty list of people. The user creates people by clicking on the "new" button. This creates a "new person" dialog box in which the user may enter the person's first name, last name, and birth date. Double-clicking on the icon for a person recreates the same dialog box, enabling the user to edit the values (see Figure 6.1).

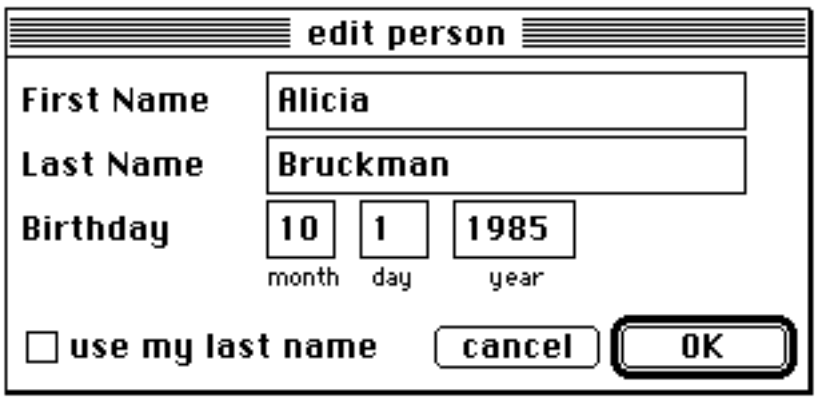

Figure 6.1: The "edit person" dialog box.

Note that a person's first name and last name are values on slots and not the name of the unit itself. This makes it easier to change a person's name. It also facilitates managing when to use a person's full name or just their first name. If the box on the person-editor "use my last name" is checked, then the person will always be referred to by his/her full name. This mimics how duplicate first names are often handled in social situations. A family member named Bernie is referred to as simply "Bernie," and a family friend named Bernie is referred to as "Bernie Zucker."[1]

As was discussed in section 5.4.2, maintaining a list of known people helps to assure consistency of logging; it is less likely that the same person will be referred to by two different nicknames. It also allows the system to track information about people, such as their birthday and family relations.

6.1.2 Inferencing Capability: Family Relationships

Family relationships are a useful data abstraction for home-video stories. One might want to tell a story about a group of siblings, a nuclear family, an extended family, or the like. In some cultures, cousin, uncle, or aunt relationships have great importance, and those relations might be used to help construct stories.

A benefit of using a semantic network is the ability to make inferences. For example, if Mari is Alicia's parent, then the inverse relationship holds: Alicia is Mari's child. Furthermore, if it was already known that Danielle is Mari's child, then it follows that Danielle and Alicia are either half or full sisters (see Figure 2.1).

The computer's ability to make these inferences saves the user work, and also makes the system seem "smarter." Researchers working on the Cyc Project are trying to determine whether, given enough information and inferencing capability, something greater might emerge from a system, perhaps something resembling "common sense" [LG 90]. When a semantic network is implemented on a small scale, however, what is gained is efficiency -- the relationships inferred by the system could be entered by a human. This extra efficiency may be essential to usability of a system. However, it is useful to remember that there is nothing magic about inferencing capability; it merely saves the user work.

---

[1] It would be possible for the system to check lists of people for duplicate first names in first-name-only mode, and automatically add last names to avoid ambiguity. This feature is not implemented yet.

6.1.3 The Need for a Graphical Interface

Family relationships are currently logged in lisp code, because a satisfactory interface has not yet been developed. To state that Mari is Alicia's mother, one can type:

```
(assertion (human-name-to-unit "Alicia")
           'mother
           (human-name-to-unit "Mari"))
```

What would interface code for this property look like? One simple solution would be to add fields for a few key family relations to the person-editor object. Thus, there could be "mother," "father," and "children" fields with either pull-down menus or list-browsers to select from lists of people to fill those slots. There are two problems with this solution. First, it limits the number of relationships which can be asserted. Thus, to say that Enid is Alicia's great-grandmother, it is necessary to create the intermediate family members and assert the information indirectly.

More importantly, there is no natural way in this scheme to inform the user of the inferences being made. If the system is now told that Mari is Danielle's mother, it will infer that Danielle and Alicia are sisters. However, the user will not be notified that this inference has been made. There is no way for the user to visualize the underlying family tree and watch it evolve. The user will be unaware of gaps or errors in the representation.

A graphical representation would help to solve these problems. People can be represented as points or icons on a plane. To assert a relationship, the user selects the relationship and drags an arrow between the two people. Inferred relationships are then drawn in a different color or line thickness.

Graphical methods are a common way to display and manipulate information in a semantic network. One problem commonly arises: if all relationships are marked, the display of even a simple semantic network can become so crammed with information as to be illegible [Travers 89].

This problem can be eased by giving the system a representation of which relationships are relevant to display and hiding those that are not currently needed. For example, the default might be to display parent-child relationships but not grandparent-grandchild relationships. The user then can change which relationships are displayed in order to accomplish a specific task.

6.1.4 But What Can All of This Information Be Used For?

In the abstract, it seems useful to be able to track family relationships in a home video system. However, for the quantity of video information the Electronic Scrapbook currently uses, this information is useful but not essential. Nevertheless, I believe that the benefits increase as the amount of available data increases. If many people are present in the video database, then family relationships may be a powerful abstraction. This hypothesis needs to be tested with a larger volume of data.

Consider this example. Knowing family relationships, it is possible to make a story model for "child-with-siblings" which focuses attention on the child selected and includes a roughly equal number of shots of the child with each of his/her siblings. This is equivalent to a more general model, person-with-others; the only difference is that the system is calculating the list of others. The representation of siblings has two benefits: it suggests a possible way to tell the story to the user, and it saves typing time if there are a large number of children in the family.

The system currently has a representation for a person's immediate family and extended family. Your immediate family is defined as your parents, siblings, spouse, and children. Your extended family is defined as the kleene-star of immediate family; in other words, it recursively includes the immediate family of your immediate family.

In a video with a large number of people present, this information could be used to help focus attention on the important people. In fact, the system has a representation of who the important people are at different types of events. For example, the important people at a wedding are defined as the members of the wedding party plus the immediate families of the bride and groom. The important people at a birthday party are the immediate family of the guest of honor.

6.1.5 Focus Person

It would be useful in the future to add the ability to mark the focus person for a video segment. Consider, for example, a shot of a child learning to walk. The child's parents and older siblings are present in the scene, and everyone is walking. Merely checking who is performing the action "walking" will not reveal the key information: this is a shot of the youngest child learning to walk. The solution is to allow the user to mark some of the people in a shot as more important.

For the example of a child learning to walk, the Electronic Scrapbook currently finds the important person by checking ages of the participants. If there are a one-year-old, five-year-old, and two forty-year old people in a shot

of the event "learning to walk,"[2] it is likely that the one-year-old is the focus person. This assumption could, of course, be wrong: for example, one of the adults could be recovering from a stroke. However, even without morbid counter examples, the focus person is an important and useful concept and it is preferable to represent this information explicitly. The inferred value should be made a default which can be changed by the user.

6.2 People in Audio Only

6.2.1 The Need for Separate Representations of Video and Audio

It is often the case that someone speaks but does not appear in the video. For example, the person holding the camera may narrate or engage in conversation. If Robert is narrating, he is present but one would not want the shot to turn up in a search for video of Robert. He can be logged as being present in audio only.

This phenomenon is only the most obvious manifestation of a more general problem: the audio and video can have different content. Consider, for example, this video segment: Mari and Alicia are in the pool. Alicia is splashing in the water. Robert narrates "Ali can crawl backwards, but not forwards yet. She climbed one step today in the new house." What is happening in the picture is different from what is being discussed. Ideally, the content of the audio needs to be represented separately from the video.

6.3 Locations

6.3.1 Functional rather than Geographic Information

Locations for home video center around the home, local recreational facilities, and vacation spots. In home video, the functional properties of a place tend to be more important than the geographic ones. In other words, it is more useful to know that the Grand Canyon is a tourist attraction than to know that it is in Northwestern Arizona.[3] The basic locations hierarchy now is:

```
COMMERCIAL SPACE
  HOTEL
  OFFICE
  RESTAURANT
```

[2] See section 6.4 for further discussion of events.

[3] If specific geographical information was needed, then a map-based graphical interface would be useful. A tool for anthropological or geological research, a news archive, and a large stock footage archive are examples of applications in which this information might be useful.

```
    STORE
  INDOORS
  INSTITUTIONAL SPACE
    GOVERNMENT-OFFICE
    HOSPITAL
    PLACE-OF-WORSHIP
    SCHOOL
  OUTDOORS
  RECREATIONAL-SPACE
    BEACH
    LAKE
    MOUNTAINS
    PARK
    POOL
    RIVER
    STADIUM
    TOURIST-ATTRACTION
    VACATION-SPOT
  RESIDENTIAL-SPACE
    FRIEND-OR-RELATIVES-HOME
    HOME
    VACATION-HOME
  ROOMS[4]
    BALCONY
    BATHROOM
    BEDROOM
    KITCHEN
    LIVING-ROOM
    PORCH
    YARD
  STREET
```

By using the "new" button, a user can create subclasses of locations to represent specific places. If Italy is defined as a vacation spot, Rome as a part of Italy, and the Spanish Steps as a part of Rome, then a search for the location "Italy" will include video shot on the Spanish Steps.[5] However, all user-created classes are treated by the system in the same way as the parent class. In other words, "the-grand-canyon" and "the-spanish-steps" are both simply specific instances of tourist-attraction.

6.3.2 Linking Space and Time

It would be interesting in the future to link the representations of space and time. All video of Italy must be from August 1987, the month that my family took a trip there. Video of San Rafael can be assumed to be from before 1986, the year that they moved to Mill Valley. In a personal context, space and time are linked.

---

[4]It would be preferable if rooms appeared hierarchically below other locations in the way that scenes appear below events. (See section 5.4.3.) This change will be made in the future.

[5]The new button for locations has not been implemented yet; new locations must be hand coded. Only the new button for people has been completed.

6.3.3 Interface Coding for a Multi-level Representation

The representation of locations does not yet include slots for what is a part of what. For example, to say that the video of Danielle eating her first solid food was shot at home in the kitchen, one would now mark "home, kitchen, indoors." A preferable representation would be to have the kitchen as a part of home. Furthermore, indoors versus outdoors should be a slot on locations with a default value. Kitchens are assumed to be indoors and balconies are assumed to be outdoors, unless otherwise indicated.

The knowledge representation required to implement these features is easy; the interface coding is a challenge. How is the user to be informed that the system is assuming that a balcony is outdoors? There is, in fact, an indoor balcony in my family's home videos. One solution would be to automatically annotate selected locations with assumed properties. Assumptions could be displayed in a different font or color to attract the user's attention. This approach will work only if there is a reasonably small number of inferences being made.

6.3.4 Multiple Inheritance

Although indented lists can be used to represent single inheritance, it is difficult to use this form to represent multiple inheritance. For example, if I shot video at my friend's cabin in Vermont, I might want to create a location friends-vacation-home which inherits from both friend-or-relatives-home and vacation-home. A graphical representation could display this information more naturally than a list. In the current system, multiple inheritance is accomplished by simply marking all appropriate locations in a list.

6.4 Events

In use of The Electronic Scrapbook to date, the events category has proved to be a powerful and useful abstraction. Events are similar to Roger Schank's work on scripts [SA 77]. Like scripts, events are made up of scenes. Double-clicking on an event adds a list of its likely scenes below that event, as was discussed in section 5.4.3. The scene list for one event type, birthday-parties, is included below.

"Trip-to-a-pumpkin-patch" is an example of an event that would be added by the user. Since the user-added category "trip-to-a-pumpkin-patch" inherits from "family-outings," it has the same scenes as family-outings.

The events hierarchy is:

```
DAILY-ROUTINES
  COOKING-SCENE
  TAKING-A-BATH
FAMILY-OUTINGS
  DAY-AT-THE-BEACH
  PICNIC
  TRIP-TO-A-PUMPKIN-PATCH
  TRIP-TO-THE-ZOO
FILMING-SCENERY
HOLIDAY
  CHRISTMAS
  EASTER
  FATHERS-DAY
  HALLOWEEN
  JULY-FOURTH
  MOTHERS-DAY
PARTIES
  BIRTHDAY-PARTIES
    GUESTS-ARRIVING
    DANCING-SCENE
    EATING-SCENE
    GIVING-PRESENTS
    OPENING-PRESENTS
    PLAYING-GAMES
    THE-CAKE-SCENE
      PUTTING-CANDLES-IN-THE-CAKE
      LIGHTING-THE-CANDLES
      CAKE-ARRIVING
      SINGING-HAPPY-BIRTHDAY
      BLOWING-OUT-THE-CANDLES
      CUTTING-THE-CAKE
      EATING-THE-CAKE
    WATCHING-PERFORMER
    GUESTS-LEAVING
    THE-MESS-AFTERWARDS
PERFORMANCE
PLAYING-WITH-PETS
WATCHING-CHILDREN
  BIRTH
  BOTTLE-FEEDING-BABY
  CHILD-PLAYING
    CHILD-PLAYING-WITH-FRIEND
    CHILD-PLAYING-WITH-RELATIVE
    SIBLINGS-PLAYING
  FIRST-SOLID-FOOD
  HOLDING-BABY
    BURPING-BABY
  LEARNING-TO-CRAWL
    CRAWLING-BACKWARDS
  LEARNING-TO-SIT-UP
  LEARNING-TO-STAND-UP
  LEARNING-TO-SWIM
```

```
  LEARNING-TO-WALK
    FIRST-STEPS
  NEW-FOOD
  NURSING-BABY
WEDDINGS
```

6.4.1 The Power of the "Event" Abstraction

The Electronic Scrapbook is primarily event driven. Useful information about home video is structured around events. For example, a birthday party is likely to have an "opening-presents" scene. There are probably going to be too many shots in the opening-presents scene, so the system includes a heuristic for trimming them down to a reasonable number. In contrast, the "blowing-out-the-candles" scene is important and is unlikely to need to be shortened; in a birthday party story, the system simply includes all video of blowing-out-the-candles. Story models are discussed further in chapter 8.

In practical use of the system, I have found that events are often the most accurate way to search for a remembered piece of video. Searching for "picnic" is easier than trying to remember the date and place and more accurate than searching for "food" and "outdoors."

An event is information about the function of a particular chunk of video in the lives of the participants. Knowledge about typical home video is structured around event types. This information can be used to help a home-video artist by anticipating his/her needs. The system has a vocabulary of likely events and a library of templates for stories about those events.

6.5 Actions

Actions are subtly different from events. A shot of the event "learning-to-crawl" is a specific event; the fact that the child is crawling is the subject of the video. The child may, however, perform the action "crawling" in many shots.

Actions currently used by the system are:

```
BATHING
CLAPPING
COOKING
CRAWLING
CRYING
DANCING
EATING
FIGHTING
FROWNING
KISSING
```

```
LAUGHING
LIFTING-CHILD
PLAYING
POSING
POUTING
READING
RUNNING
SHOPPING
SINGING
SITTING
SLEEPING
SMILING
STANDING-UP
SWIMMING
WALKING
WORKING
YELLING
```

6.5.1 Emotional Actions

To date, actions have been most useful for emotional actions such as "smiling," "laughing," "crying," and the like. It is easier for the user to make the objective judgement that subject is smiling than the subjective judgement that the shot is "happy." Emotional actions are useful for many of the functions I originally anticipated would be performed by the subjective-properties category.[6]

6.5.2 Complex Information

The representation of actions is currently simplistic. A shot merely contains an action or it does not. The subject, direct object, and indirect object of the action are not represented. This part of the representation has not been more fully developed because to date it has been surprisingly unimportant. Actions are similar to events, but events are more useful for structuring home video stories.

What are the potential benefits of a more robust representation of actions? Certainly marking the subject of actions is important and will be implemented in the future. It is useful to know that Danielle is the one who is performing the action of running. Is it useful to know that Danielle is running from the north side of the pumpkin patch to the south side? Is it useful to know that she is running away from the camera and to the right within the frame? Is it useful to know that she is running towards the parking lot and away from the other family members? How will all of this information be used? How will it be entered by the editor? How long will it take to enter? Are the benefits worth the time invested? These are unanswered questions.

---

[6]See section 6.9 for further discussion of subjective properties.

6.6 Significant Objects

The current significant-objects hierarchy is:

```
ANIMALS[7]
  WILD-ANIMALS
  ZOO-ANIMALS
CLOTHING
  HAT
  JEWELRY
FOOD
  BABY-FOOD
  CAKE
  JUNK-FOOD
  MESSY-FOOD
  MILK
  SPECIAL-FOOD
HOLIDAY-OBJECTS
  CHRISTMAS-TREE
  EASTER-EGG
  GIFTS
  MENORAH
  PARTY-HATS
  PUMPKIN
HOUSEHOLD-OBJECTS
  BROOM
  FORK
  KNIFE
  SINK
  SPOON
  TOILETRIES
    HAIRBRUSH
    SOAP
    TOOTHBRUSH
PLANTS
  FLOWERS
  TREES
TOYS
  BALL
  BLOCKS
  BOARD-GAME
  BUBBLE-STUFF
  DOLL
  FAVORITE-BLANKET
  FAVORITE-TOY
```

[7]The unit "animals" is actually called "animalia" and has a print name "animals." There was already a unit "animals" in the Arlotje base ontology. That representation of animals is structured differently than the representation used by the Electronic Scrapbook. The Electronic Scrapbook unit was renamed "animalia" to avoid the conflict.

The purpose of having a base ontology in a knowledge representation system is to save the user work by providing a set of basic objects. In practice, however, someone else's representation is as often an obstacle as a benefit. This is evidence for the fact that it is difficult to develop a general-purpose knowledge representation.

```
    FRISBEE
    RINGS
    SPORTS-EQUIPMENT
      BASEBALL-GLOVE
    STUFFED-ANIMAL
  VEHICLE
    AIRPLANE
    AUTOMOBILE
    BICYCLE
    BOAT
    BUS
    MOTORCYCLE
    TRICYCLE
  WATER
```

6.6.1 How Much Representation is Necessary?

As was discussed in the introduction, it would be difficult to represent every object in the frame, and a comprehensive list is unnecessary, unless you are the continuity supervisor for a feature film [Lasky 90]. In the Electronic Scrapbook, the objects category is called "Significant Objects" to attempt to convey to the user that a subjective judgement of what is important should be made.

The question then remains, should the system be able to make inferences about what objects *might* be in a scene? It could, for example, assume that there is probably water in a shot taken at the beach or in the bathroom, that there is probably a sink in the kitchen, and that there might be a turkey in a shot taken in the dining room on Thanksgiving. The ability to make these kind of inferences is something that a semantic network is useful for.

The unanswered question is in what circumstances this information is desirable. Would one ever want to do a search for "water," "sink," or "turkey"? What is the trade off between the benefits of these inferences and the cost of errors? There is not, after all, always a turkey on Thanksgiving. And while there will almost always be a sink in a kitchen, it may not be visible in the frame.

6.7 Segment Date

6.7.1 Qualitative Reasoning

Chronology is a useful organizing principle for home video subjects. Once chronology is violated, then one needs a tremendous amount of data to maintain continuity. In a fictional interactive disc made at the Media Lab in 1987 which tells the story of two people eating a spaghetti dinner, the system constantly tracks how full each person's wine glass is [Schroeder 87]. A segment of the glass being refilled is used as a transition if one wants

suddenly to have more wine in the glass. With documentary subject matter, one can usually ignore these sorts of issues if natural chronology is never violated.[8]

The segment date is recorded in the form month/day/year. All date-related functions tolerate ambiguity. It is possible to omit any part of the date; a segment could be marked as "June 6th, 1987," "June 1987," "1987," or simply "June," if that is all that is known about it.

The possibility of supporting qualitative reasoning about dates was considered during the design of the system. For example, it would be convenient to give segments seasonal dates such as "Spring 1987," or relative dates like "before Christmas 1990," or "before [event x]."

However, these features become unnecessary, because in the future, it will be possible for much of the work of logging video to be done automatically. Date and time are the easiest feature to log automatically. Many currently available camcorders have clocks and can superimpose the date and time on the video. Instead of displaying it on the screen, this information could be recorded in an unused portion of the video tape such as the vertical blanking interval. There is no technological limitation to implementing this feature; it will become available when manufacturers perceive a demand for it [DSP 91].

6.8 Camera Properties

The current camera-properties hierarchy is:

```
CAMERA-SPECIAL-EFFECTS
  CAMERA-GENERATED-TITLES
  FADE
    FADE-IN
    FADE-OUT
CAMERA-MOVEMENT
  PAN
    CAMERA-FOLLOWS-OBJECT
    PAN-LEFT
    PAN-RIGHT
  TILT
    TILT-DOWN
    TILT-UP
  WALKING-WITH-THE-CAMERA
FOCAL-LENGTH
  CLOSE-UP
  EXTREME-CLOSE-UP
  MACRO-SHOT
  MEDIUM-SHOT
```

[8]Parallel editing is an interesting case in which natural chronology is not used. See section 8.5 for further discussion of parallel editing.

```
  WIDE-SHOT
  ZOOM
    ZOOM-IN
    ZOOM-OUT
OBJECT-MOVEMENT
  OBJECT-AWAY-FROM-CAMERA
  OBJECT-TO-CAMERA
PROBLEM-WITH-AUDIO
  AUDIO-VERY-LOUD
  AUDIO-VERY-SOFT
  BACKGROUND-NOISE
  GARBLED-AUDIO
  SENTENCE-CUT-OFF
  WORD-CUT-OFF
PROBLEM-WITH-VIDEO
  BACK-LIT
  JIGGLE
  MAJOR-PROBLEM-WITH-VIDEO
  MINOR-PROBLEM-WITH-VIDEO
  OUT-OF-FOCUS
  TOO-BRIGHT
  TOO-DARK
SEQUENCE-OF-SHOTS
TRIPOD-SHOT
```

6.8.1 Video Quality

In experience with the system to date, camera properties have been most often used for indicating shot quality. If one is trying to tell a story about a child at a certain age, then there is usually too much material and one can eliminate all shots with any "problem-with-video" or "problem-with-audio." However, if one is looking for the shot of a child's first steps, it doesn't really matter if the video is too dark or out of focus; the moment is important, and the quality constraints should be relaxed or eliminated.

6.8.2 Stylistic Constraints

In the original system design, I anticipated that camera information could be used to fulfill some common stylistic constraints such as avoiding jarring jump cuts and violations of the 180° degree rule.[9] However, since there is

---

[9]The book *The Filmmaker's Handbook* [PA 84] provides good descriptions of these conventions: "A disconcerting mismatch between shots is called a *jump cut.* To cut from a shot of a person sitting to a shot of the same person standing in the same spot creates a disconcerting jump in time. A slight change in image size from one shot to the next (for example, two medium shots of the same person cut next to eachother) makes the cut appear to jump and feels unmotivated. A larger change in image size or angle could make the same cut in action seem natural." (p. 280)

For the 180° rule it states that "If a subject is moving or looking right in one shot, it is best not to let screen direction change when cutting to the next shot. For example, when watching football on television, the blue team is seen moving from screen left to screen right. If the camera were now to shift to the opposite side of the field, the blue team would appear to be moving in the opposite direction (that is, their screen direction has changed from right to left.)" (p. 158)

usually only one camera and there are not multiple takes of the action, in practice these stylistic problems do not often occur.

In my family's home videos, jump cuts are a serious problem in only one instance: my father interviewed my grandparents about their lives before they came to America. These interviews consist of long, static, tripod shots. Editing them creates jump cuts. However, even if the system could detect these jump cuts, there is nothing that can be done to eliminate them. The standard way to prevent a jump cut is to put something else in between -- a "cutaway" shot[10] [PA 84, p. 159]. However, there were no cutaways taken; my father's interest was in documenting his parents lives, and he did not anticipate editing the interviews. As a typical novice videographer, he was not aware of the need for cutaway shots.

In the current implementation, the system does not try to make sequences in the traditional sense. The computer may select a shot of a child falling, but it can not put together a wide shot of the child walking, followed by a close up of a tree root and the child's foot getting caught, followed by a medium shot of the child crying on the ground. The raw material to make such a sequence is rarely if ever available in typical home video. However, if the system were ever to attempt to construct a sequence depicting an event, then precise information about camera properties would be needed.

6.9 Subjective Properties

The subjective properties hierarchy is:

```
AESTHETICS
  BEAUTIFUL-SHOT
  PRETTY-SHOT
  UGLY-SHOT
ANGRY
CUTE
  VERY-CUTE
FUNNY
  VERY-FUNNY
HAPPY
INTERESTING
  VERY-INTERESTING
SAD
SENSITIVE-SUBJECT
```

[10]Some filmmakers support the use of jump cuts in this situation. Special effects such as dissolves can also be used to create transitions.

6.9.1 Lack of Perspective on the Subject Matter

By the time I logged its subjective properties, I had seen my video source material too many times to be able to decide what was cute and what was funny. Every segment already had to be either cute, funny, or somehow interesting to be included on the half-hour laser disc. It might have been easier to judge subjective properties had I marked them the first time I viewed the video.

Lack of perspective on the subject matter will probably be a common problem for home video artists. My sisters are always cute and everything my father says is funny. A normally sane colleague of mine believes that every moment of the four hours of his wedding videos is interesting. How then can one encourage users to throw things out?

When the Electronic Scrapbook is moved to a digital-video platform, users will have a limited amount of disk space to store video. Furthermore, digitizing each clip will take time. In the near future, limitations on time and storage space will force people to make choices and pare the total material down to a reasonable volume. In fact, the digitizing process will serve the same function as the need to press a laser disc served in this version.[11]

6.9.2 Sensitive Subject Matter

One subjective property which proved to be unexpectedly useful is "sensitive-subject-matter." For example, the laser disc includes video of my half sisters being born and one shot of my stepmother not fully clothed nursing the newborn Alicia. By marking these as sensitive-subject-matter, the user can decide when to exclude them from video shown.

---

[11]Digitizing speeds and compression ratios are rising, and the price of storage is falling. If these trends continue, in the future the user may be able to store a very large quantity of video. A key problem then arises: how can one encourage a user to throw out bad video? Experienced documentary filmmakers learn the skill of knowing when to turn the camera off; having too much raw material is a burden [Leacock 88].