M A R C D A V I S
www.marcdavis.me

P U B L I C A T I O N S
info@marcdavis.me

Media Streams: An Iconic Visual Language for Video Annotation

Bibliographic Reference:

Marc Davis. "Media Streams: An Iconic Visual Language for Video Annotation." *Telektronikk 4.93* (1993): 59-71.

Media Streams:
An Iconic Visual Language for Video Annotation

Marc Davis

Interval Research Corporation
1801-C Page Mill Road
Palo Alto, CA 94304
davis@interval.com

Abstract

In order to enable the search and retrieval of video from large archives, we need a representation of video content. Although some aspects of video can be automatically parsed, a detailed representation requires that video be annotated. We discuss the design criteria for a video annotation language with special attention to the issue of creating a global, reusable video archive. We outline in detail the iconic visual language we have developed and a stream-based representation of video data.

Our prototype system, Media Streams, enables users to create multi-layered, iconic annotations of video content. Within Media Streams, the organization and categories of the Director's Workshop allow users to browse and compound over 2500 iconic primitives by means of a cascading hierarchical structure which supports compounding icons across branches of the hierarchy. Icon Palettes enable users to group related sets of iconic descriptors, use these descriptors to annotate video content, and reuse descriptive effort. Media Time Lines enable users to visualize and browse the structure of video content and its annotations. The problems of creating a representation of action for video are given special attention, as well as describing transitions in video.

1 Introduction: The Need for Video Annotation

The central problem in the creation of robust and extensible systems for manipulating video information lies in representing and visualizing video content. Currently, content providers possess large archives of film and video for which they lack sufficient tools for search and retrieval. For the types of applications that will be developed in the near future (interactive television, personalized news, video on demand, etc.) these archives will remain a largely untapped resource, unless we are able to access their contents. Without a way of accessing video information in terms of its content, a thousand hours of video is less useful than one. With one hour of video, its content can be stored in human memory, but as we move up in orders of magnitude, we need to find ways of

creating machine-readable and human-usable representations of video content. It is not simply a matter of cataloging reels or tapes, but of representing and manipulating the content of video at multiple levels of granularity and with greater descriptive richness. This paper attempts to address that challenge.

Given the current state of the art in machine vision and image processing, we cannot now, and probably will not be able to for a long time, have machines "watch" and understand the content of digital video archives for us. Unlike text, for which we have developed sophisticated parsing technologies, and which is accessible to processing in various structured forms (ASCII, RTF, PostScript), video is still largely opaque. We are currently able to automatically analyze scene breaks, pauses in the audio, and camera pans and zooms [41, 21, 31, 33, 34, 38, 39], yet this information alone does not enable the creation of a sufficiently detailed representation of video content to support content-based retrieval and repurposing.

In the near term, it is computer-supported human annotation that will enable video to become a rich, structured data type. At this juncture, the key challenge is to develop solutions for people who already devote time and money to annotating video, because they will help create the necessary infrastructure (both economically and in terms of the content itself) to support the ubiquitous use and reuse of video information. Today, simple queries often take tens of hours and cost thousands of dollars. If recorded reusable video is going to become a ubiquitous medium of daily communication, we will need to develop technologies which will change the current economics of annotation and retrieval.

1.1. Video Annotation Today

In developing a structured representation of video content for use in annotation and retrieval of video from large archives, it is important to understand the current state of video annotation and to create specifications for how future annotation systems should be able to perform. Consequently, we can posit a hierarchy of the efficacy of annotations:

> *At least*, Pat should be able to use Pat's annotations.
>
> *Slightly better*, Chris should be able to use Pat's annotations.
>
> *Even better*, Chris's computer should be able to use Pat's annotations.
>
> *At best*, Chris's computer and Chris should be able to use Pat's and Pat's computer's annotations.

Today, annotations used by video editors will typically only satisfy the first desideratum (Pat should be able to use Pat's annotations) and only for a limited length of time. Annotations used by video archivists aspire to meet the second desideratum (Chris should be able to use Pat's annotations), yet these annotations often fail to do so

if the context of annotation is too distant (in either time or space) from the context of use. Current computer-supported video annotation and retrieval systems use keyword representations of video and ostensibly meet the third desideratum (Chris's computer should be able to use Pat's annotations), but practically do not because of the inability of keyword representations to maintain a consistent and scalable representation of the salient features of video content.

In the main, video has been archived and retrieved as if it were a non-temporal data type which could be adequately represented by keywords. A good example of this approach can be seen in Apple Computer's *Visual Almanac* which describes and accesses the contents of its video and image archive by use of "keywords" and "image keys" [4]. This technique is successful in retrieving matches in a fairly underspecified search but lacks the level of granularity and descriptive richness necessary for computer-assisted and automatic video retrieval and repurposing. The keyword approach is inadequate for representing video content for the following reasons:

- Keywords do not describe the complex *temporal* structure of video and audio information.
- Keywords are not a *semantic* representation. They do not support inheritance, similarity, or inference between descriptors. Looking for shots of "dogs" will not retrieve shots indexed as "German shepherds" and vice versa.
- Keywords do not describe *relations* between descriptions. A search using the keywords "man," "dog," and "bite" may retrieve "dog bites man" videos as well as "man bites dog" videos—the relations between the descriptions highly determine salience and are not represented by keyword descriptions alone.
- Keywords do not *scale*. As the number of keywords grows, the possibility of matching the query to the annotation diminishes. As the size of the keyword vocabulary increases, the precision and recall of searches decrease.

Current paradigms of video representation are drawn from practices which arose primarily out of "single-use" video applications. In single-use applications, video is shot, annotated, and edited for a given movie, television program, or video. Annotations are created for one single use of the video data. There do exist certain cases today, like network news archives, film archives, and stock footage houses, in which video is used multiple times, but the level of granularity of the annotation and the semantics of the annotations do not support a wide reusability of video content. The challenge is to create representations which support "multi-use" applications of video. These are applications in which video may be dynamically resegmented, retrieved, and resequenced on the fly by a wide range of users *other than those who originally created the data*.

Today, in organizations and companies around the world whose business it is to annotate, archive, and retrieve video information, by and large, the structure of the data is mostly represented in the memories of the human beings whose job it is to handle it. Even in situations in which keyword-based computer annotation systems are "used," short-term memory and long-term memory are the real repositories of information about the content of video data. "Joe and Jane in the basement" are the real indexing and retrieval mechanisms in almost all video archives. Human memory is very good at retrieving video due to its associative and analogical capabilities; it has memory structures which any computerized retrieval system would want to emulate. Nevertheless, there are significant problems in sharing the contents of one human memory with others and of transferring the contents of one human memory to another. There are also severe limitations in terms of storage capacity and speed for human memory that aren't acceptable if we are going to scale up to a global media archive in which video is accessed and manipulated by millions of people everyday.

We need to create a language for the representation of video content which enables us to combine automatic, semi-automatic, and human annotation so as to be able to make use of today's annotation effort long into the future.

1.2. Video Annotation Tomorrow

In the near future, we can imagine a world in which video annotation, search, and retrieval are conducted not just by professionals for professionals, but by anyone interested in repurposing footage. In a world where digital media are produced anywhere by anyone and are accessible to anyone anywhere, video will need to accrete layers of content annotations as it moves around the globe throughout its life cycle of use and reuse. In the future, annotation, both automatic and semi-automatic, will need to be fully integrated into the production, archiving, retrieval, and reuse of video and audio data. In production, cameras will encode and interpret detailed information about where, when, and how they are recording and attach that information to the digital data stream: global satellite locators will indicate altitude, longitude and latitude, time will be stamped into the bit stream, other types of sensing data—temperature, humidity, wind—as well as how the camera moves (pans, zooms, etc.) and how far away the camera is from its subjects (range data for example) will all provide useful layers of annotation of the stream of video and audio data which the camera produces. Still there will exist many other annotations of a more semantic nature which these cameras won't be able to automatically encode, and for which we will want to have formats so that humans working with machines will be able to easily annotate video content. In a sense, the challenge is to develop a language of description which humans can read and write and which computers can read and write which will enable the integrated description and creation of video data. Such a language would satisfy the fourth desideratum of video annotation (Chris's computer and Chris should be able to use Pat's and Pat's computer's annotations).

By having a structured representation of video content—meaningful bits about the bits—future annotation and retrieval technology will enable users to mix video streams according to their contents and to manipulate video at various levels of granularity. With this kind of representation, annotation, and retrieval technology we will create tools which enable users to operate on higher level content structures of video data instead of being stuck with just bits, pixels, frames, or clips.

2 Design Criteria for Video Annotation Languages

A language for video annotation needs to support the visualization and browsing of the structure of video content as well as search and retrieval of video content. There has been some excellent work in visualizing and browsing video data [37, 40, 31, 33, 21] with which our work has affinity. The limitations of these systems rest in the question of their scalability and, a related problem, their lack of a developed video annotation language. For as visualization and browsing interfaces must accommodate larger and larger video databases, they need to be able to work with video according to its content as well as its structure, and hence, annotation and retrieval become necessary components of the system.

A video annotation language needs to create representations that are durable and sharable. The knowledge encoded in the annotation language needs to extend in time longer than one person's memory or even a collective memory, and needs to extend in space across continents and cultures. Today, and increasingly, content providers have global reach. German news teams may shoot footage in Brazil for South Korean television which is then accessed by American documentary filmmakers, perhaps ten years later. We need a global media archiving system that can be added to and accessed by people who do not share a common language, and the knowledge of whose contents is not only housed in the memories of a few people working in the basements of news reporting and film production facilities. Visual languages may enable the design of an annotation language with which we can create a truly global media resource. Unlike other visual languages that are used internationally (e.g., for traffic signage, operating instructions on machines, etc. [18]) a visual language for video annotation can take advantage of the affordances of the computer medium. We can develop visual languages for video that utilize color, animation, variable resolution, and sound in order to create durable and sharable representations of video content.

3 Representing Video

3.1. Streams vs. Clips

In designing a visual language for video content we must think about the structure of what is being represented. A video camera produces a temporal stream of

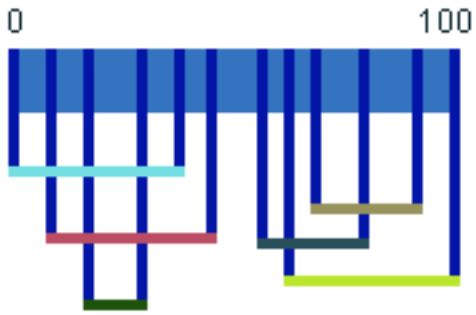
image and sound data represented as a sequence of frames played back at a certain rate—normally 30 frames per second. Traditionally, this stream of frames is segmented into units called clips. Current tools for annotating video content used in film production, television production, and multimedia, add descriptors (often keywords) to clips. There is a significant problem with this approach. By taking an incoming video stream, segmenting it into various clips, and then annotating the content of those clips, we create a *fixed segmentation* of the content of the video stream. Imagine a camera recording a sequence of 100 frames.

Stream of 100 Frames of Video

Traditionally, one or more parts of the stream of frames would be segmented into clips which would then be annotated by attaching descriptors. The clip is a fixed segmentation of the video stream that separates the video from its context of origin and encodes a particular chunking of the original data.

A "clip" from Frame 47 to Frame 68 with Descriptors

In our representation, the stream of frames is left intact and is annotated by multi-layered annotations with precise time indexes (beginning and ending points in the video stream). Annotations could be made within any of the various categories for media annotation discussed below (e.g., characters, spatial location, camera motion, dialogue, etc.) or contain any data the user may wish. The result is that this representation makes annotation pay off—the richer the annotation, the more numerous the possible segmentations of the video stream. Clips change from being fixed segmentations of the video stream, to being the results of retrieval queries based on annotations of the video stream. In short, in addressing the challenges of representing video for large archives what we need are representations which make clips, not representations of clips.



The Stream of 100 Frames of Video with 6 Annotations Resulting in *66* Possible Segmentations of the Stream (i.e., "clips")

3.2. Categories for Media Annotation

A central question in our research is the development of a minimal representation of video content. This has resulted in the development of a set of categories for, and a way of thinking about, describing video content. Let us build up these categories from examining the qualities of video as a medium. One of the principal things that makes video unique is that it is a temporal medium. Any language for annotating the content of video must have a way of talking about temporal events—the actions of humans and objects in space over time. Therefore, we also need a way of talking about the characters and objects involved in actions as well as their setting, that is, the spatial location, temporal location, and weather/lighting conditions. The objects and characters involved in actions in particular settings also have significant positions in space relative to one another (beneath, above, inside, outside, etc.).

These categories—*actions, characters, objects, locations, times, and weather*—would be nearly sufficient for talking about actions in the world, but video is a *recording* of actions in the world by a camera, and any representation of video content must address further specific properties. First, we need ways of talking about *cinematographic properties*, the movement and framing of the camera recording events in the world. We also need to describe the properties of the *recording medium* itself (film or video, color or black & white, graininess, etc.) Furthermore, in video, viewers see events depicted on screens, and therefore, in addition to relative positions in space, screen objects have a position in the two-dimensional grid of the frame and in the various layered vertical planes of the screen depth. Finally, video recordings of events can be manipulated as objects and rearranged. We create transitions in video in ways not possible in the real world. Therefore, *cinematic transitions* must also be represented in an annotation language for video content.

These categories need not be *sufficient* for media annotation (the range of potential things one can say is unbounded), but we believe they are *necessary* categories for

media annotation in order to support retrieval and reuse of particular segments of video data from an annotated stream.

These minimal annotation categories attempt to represent information about media content that can function as a substrate:

- on top of which other annotations may be layered
- out of which new annotations may be inferred
- within which the differences between consensual and idiosyncratic annotations may be articulated

3.3. Video Syntax and Semantics

In attempting to create a representation of video content, an understanding of the semantics and syntax of video information is a primary concern. Video has a radically different semantic and syntactic structure than text, and attempts to represent video and index it in ways similar to text will suffer serious problems.

First of all, it is important to realize that video images have very little intrinsic semantics. Syntax is highly determinative of their semantics, as evidenced by the Kuleshov Effect [30]. The Kuleshov Effect is named after Lev Kuleshov, a Soviet cinematographer whose work at the beginning of the century deeply influenced the Soviet montage school and all later Soviet cinema [19, 20]. Kuleshov was himself an engineer who, after only having worked on one film, ended up heading the Soviet film school after the October Revolution. Kuleshov was fascinated by the ability of cinema to create artificial spaces and objects through montage (editing) by virtue of the associations people create when viewing sequences of shots, which if the shots were taken out of sequence would not be created. In the classic Kuleshov example, Kuleshov showed the following sequence to an audience:

the passive face of an actor — a bowl of soup — go to black
the same face of the actor — a coffin — go to black
the same face of the actor — a field of flowers — go to black

Upon interviewing audience members and asking them what they saw, they said, "Oh, he was hungry, then he was sad, then he was happy." The same exact image of the actor's face was used in each of the three short sequences. What the Kuleshov Effect tells us then is that the semantics of video information is highly determined by what comes before and what comes after any given shot. It is the Kuleshov Effect which makes the construction of cinematic sequences possible at all and which enables us to reuse existing footage to make new sequences.

The syntax of video sequences determines the semantics of video data to such a degree that any attempts to create context-free semantic annotations for video must be carefully scrutinized so as to determine which components are context-dependent and

which preserve their basic semantics through recombination and repurposing. Any indexing or representational scheme for the content of video information needs to be able to facilitate our understanding of how the semantics of video changes when it is resequenced into new syntactic structures. Therefore, the challenge is twofold: to develop a representation of those salient features of video which, when combined syntactically, create new meanings; and to represent those features which do not radically change when recontextualized.

4 Media Streams: An Overview

Over the past two years, members of the MIT Media Laboratory's Learning and Common Sense Section (Marc Davis with the assistance of Brian Williams and Golan Levin under the direction of Prof. Kenneth Haase) have been building a prototype for the annotation and retrieval of video information. This system is called *Media Streams*.[1] Media Streams has developed into a working system that soon will be used by other researchers at the Media Lab and in various projects in which content annotated temporal media are required. Media Streams is written in Macintosh Common Lisp [2] and FRAMER [25, 24], a persistent framework for media annotation and description that supports cross-platform knowledge representation and database functionality. Media Streams has its own Lisp interface to Apple's QuickTime digital video system software [3]. Media Streams is being developed on an Apple Macintosh Quadra 950 with three high resolution color displays.

The system has three main interface components: the Director's Workshop (See Figure 1); Icon Palettes (See Figure 2); and Media Time Lines (See Figure 3). The process of annotating video in Media Streams using these components involves a few simple steps:

1) In the Director's Workshop, the user creates iconic descriptors by cascading down hierarchies of icons in order to select or compound iconic primitives.

2) As the user creates iconic descriptors, they accumulate on one or more Icon Palettes. This process effectively groups related iconic descriptors. The user builds up Icon Palettes for various types of default scenes in which iconic descriptors are likely to co-occur, for example, an Icon Palette for "treaty signings" would contain icons for certain dignitaries, a treaty, journalists, the action of writing, a stateroom, etc.

3) By dragging iconic descriptors from Icon Palettes and dropping them onto a Media Time Line, the user annotates the temporal media

[1] A paper on an early version of this system was presented at the AAAI-91 Workshop on Intelligent Multimedia Interfaces [14] and a shorter version of this current paper was presented at the 1993 IEEE Symposium on Visual Languages in Bergen, Norway [15].

represented in the Media Time Line. Once dropped onto a Media Time Line, an iconic description extends from its insertion point in the video stream to either a scene break or the end of the video stream. In addition to dropping individual icons onto the Media Time Line, the user can construct compound icon sentences by dropping certain "glommable" icons onto the Media Time Line, which, when completed, are then added to the relevant Icon Palette and may themselves be used as primitives. For example, the user initially builds up the compound icon sentence for "Arnold, an adult male, wears a jacket" by successively dropping the icons 

,  , and  onto the Media Time Line. The user then has the compound icon 

on an Icon Palette to use in later annotation. By annotating various aspects of the video stream (time, space, characters, character actions, camera motions, etc.), the user constructs a multi-layered, temporally indexed representation of video content.

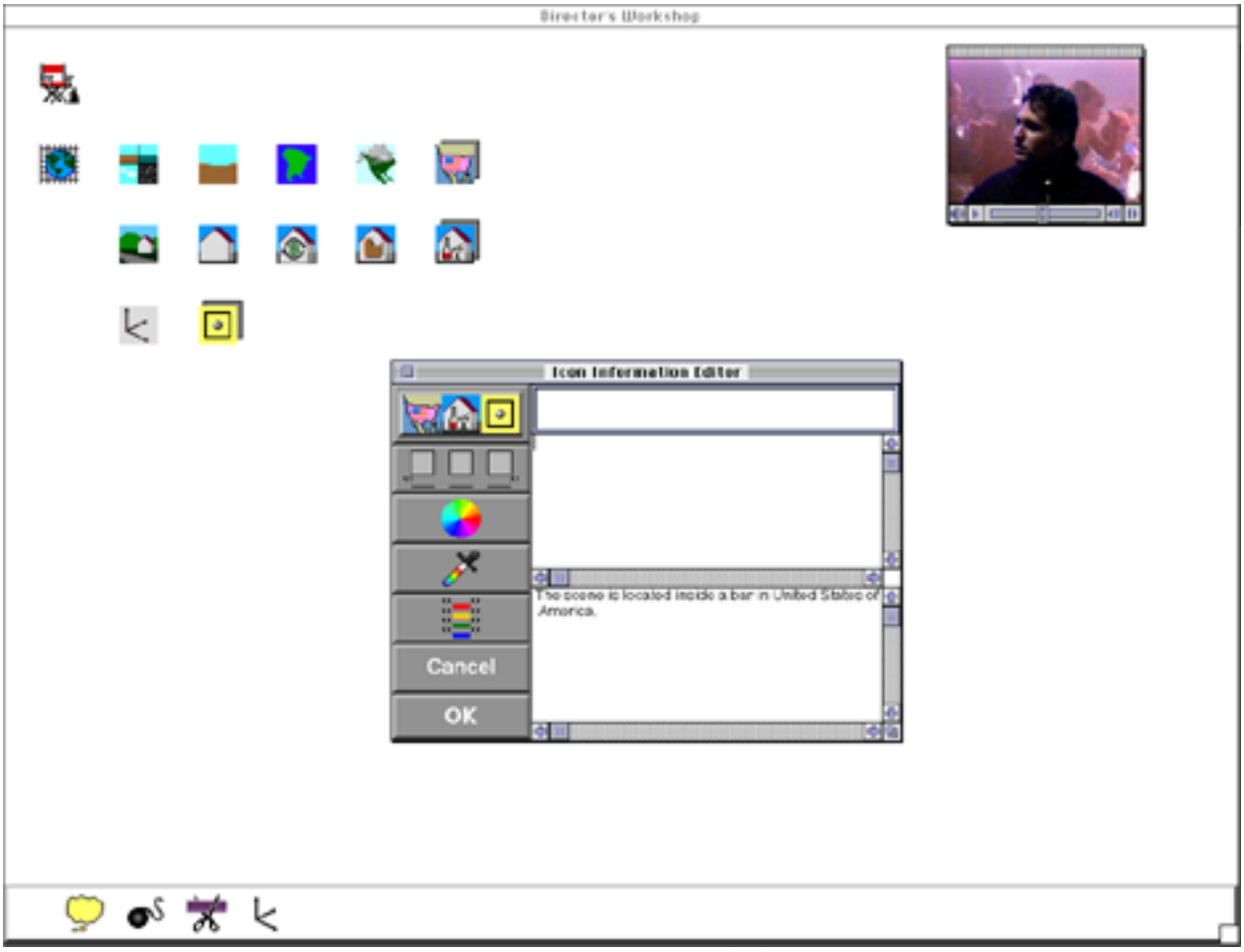


Figure 1: Director's Workshop

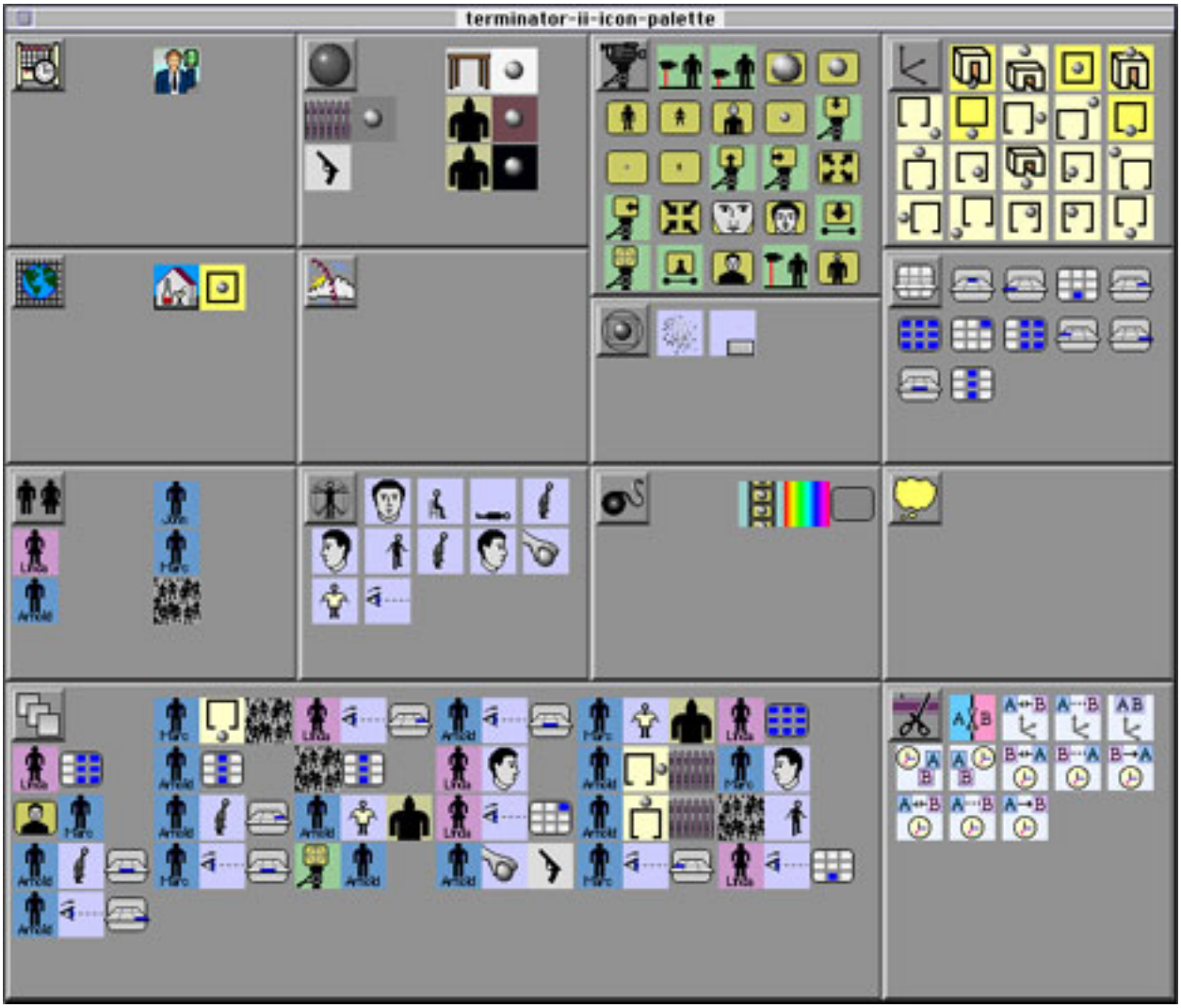


Figure 2: Icon Palette

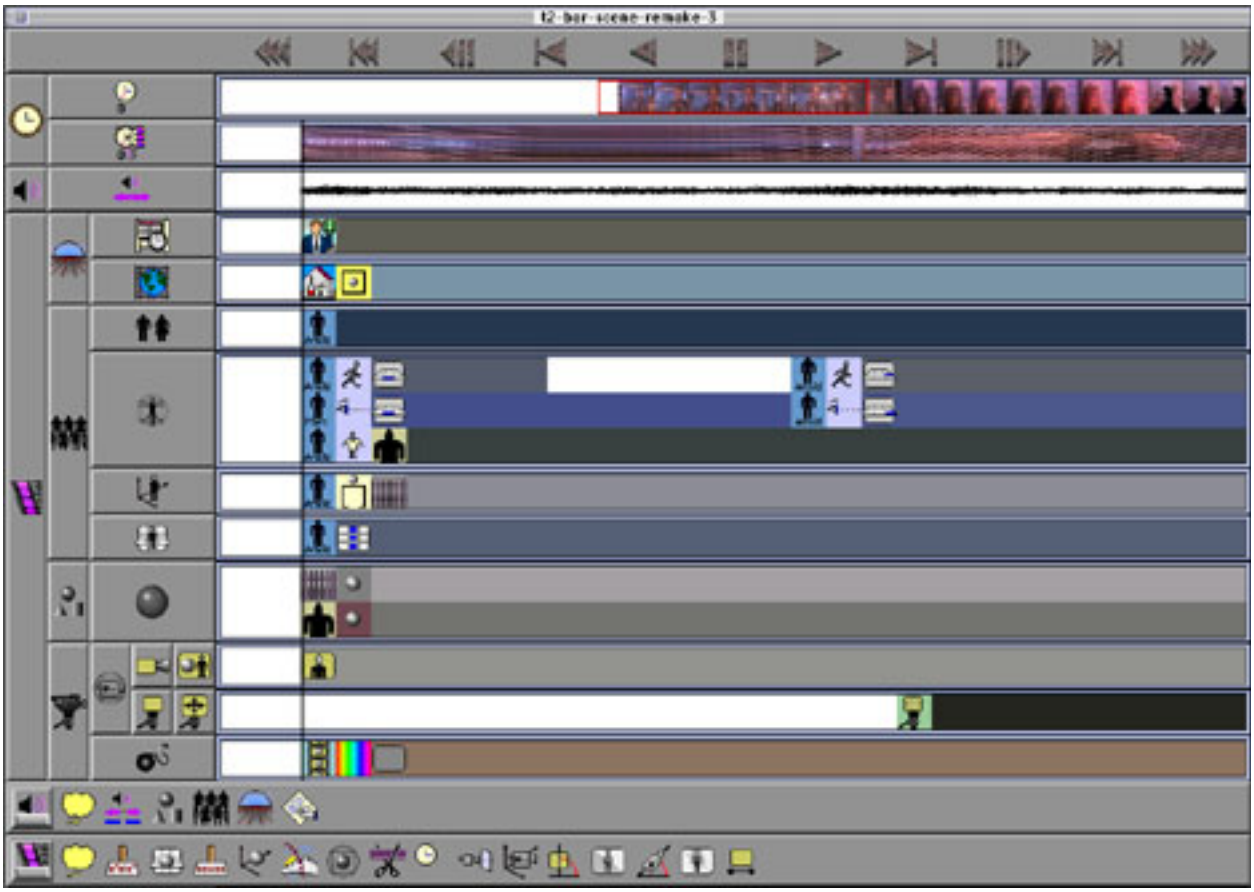


Figure 3: Media Time Line

Media Streams is a large system that attempts to address many questions in video representation. In this paper we will focus on Media Streams' language for video annotation. It is an iconic visual language that allows composition of iconic primitives in order to form complex expressions. It has a syntax for the composition of iconic sentences and means for extending the visual language.

5 Why Icons?

There have been serious efforts to create iconic languages to facilitate global communication [7] and provide international standard symbols for specific domains [18]. We developed Media Streams' iconic visual language in response to trying to meet the needs of annotating video content in large archives. It seeks to enable:

- quick recognition and browsing of content annotations
- visualization of the dense, multi-layered temporal structure of video content

- an accurate and readable time-indexed representation of simultaneous, sequential, overlapping and contained actions (natural languages are not very good at this task)
- articulation of the boundaries between consensual and idiosyncratic annotations (icons can have attached textual annotations and can thus function as the explicit consensual tokens of various idiosyncratic textual descriptions)
- global, international use of annotations
- visual similarities between instances or subclasses of a class (visual resonances in the iconic language)

Media Streams' iconic language encompasses icons which denote both things and actions and thus embodies a distinction analogous to Chang's [12] distinction between object icons and process icons. The difference here is that the objects and processes denoted by the Media Streams' icons are not computational ones, but aspects of the video content which they annotate.

The iconic language gains expressive power and range from the compounding of primitives and has set grammars of combination for various categories of icons. In Korfhage's sense Media Streams is an iconic language as opposed to being merely an iconography [28]. Similar to other syntaxes for iconic sentences [13, 35] icon sentences for actions have the form of subject-action, subject-action-object, or subject-action-direction, while those for relative positions have the form subject-relative position-object. Icon sentences for screen positions are of the form subject-screen position, while cinematographic properties are of the form camera-movement-object (analogous to subject-action-object), as in "the camera-is tracking-Steve" or "the camera-zooms in on-Sally."

6 Director's Workshop

The Director's Workshop is the interface for the selection and compounding of the iconic primitives in Media Streams (See Figure 1). To date we have over 2500 iconic primitives. What enables the user to navigate and make use of such a large number of primitives is the way the Director's Workshop organizes these icons into cascading hierarchies. We refer to the iconic primitives in the Director's Workshop as "cascading icons." The Director's Workshop has two significant forms of organization for managing navigational and descriptive complexity:

- *Cascading Hierarchy with Increasing Specificity of Primitives on Subordinate Levels*

Cascading icons are organized in hierarchies from levels of generality to increasing levels of specificity. Similar to cascading menus on the Macintosh, when a user cascades down an icon hierarchy by clicking on a cascading icon, its subordinate icons are displayed to the right of the cascading icon. These subordinate icons are arranged *horizontally* and represent an increased level of specificity. Some of the icon hierarchies cascade to as many as 7 or 8 levels deep, yet, similarly to the semantic hierarchies of the CYC Project [29], the design of the categories themselves and their first three or four levels is the hardest and most important representational task.

- *Compounding of Hierarchically Organized Primitives Across Multiple Axes of Description*
  In many icon hierarchies on the Director's Workshop, there exists an additional form of organization. When subordinate icons are arranged *vertically*, they represent independent axes of description whose icon hierarchies can be cascaded through separately and whose respective subordinate icons can be compounded together across these axes to form compound iconic descriptors. This form of organization enables a relatively small set of primitives to be compounded into a very large and rich set of descriptors.

To illustrate these forms of organization in our iconic language we can look at how the compound icon for "the scene is located inside a bar in United States of America,"  , was created. Figure 1 shows the cascading icon hierarchy for "space" extended out to the icons for "United States of America," "bar," and "inside" which the user compounded to create the icon for "the scene is located inside a bar in United States of America" which appears in the Icon Information Editor. The user clicked on the space icon, which cascaded to show its subordinate icons "geographical space," "functional space," and "topological space" *vertically* arranged. Each of these cascading icons has further *horizontally* arranged subordinate icons each of which may go several levels deep. For example, the icons in the path from "geographical space" to "United States of America" each represents a distinct level of progressive specification (geographical space->land->continent->North America->United States of America). As indicated by the gray square behind the "United States of America" icon, it too has further levels of specificity below it which can be displayed by clicking on the icon. In the Director's Workshop, at all but the terminal levels in the hierarchy, there exist many icons which themselves have further levels of specification. At any level in the hierarchy, icons can be compounded across the vertical organization to create compound icons. In addition to clicking, cascading icons can be accessed by voice (using the Voice Navigator II™), by typing in text for their names, or by dropping an existing icon onto the Director's Workshop which opens the icon hierarchies up to the terminals of the

components of the dropped icon. In all these ways, a vast, but structured space of icons can be easily navigated by the user.

It is also important to note that the icon hierarchy of the Director's Workshop is structured not as a tree, but as a graph. The same iconic primitives can often be reached by multiple paths. The system encodes the paths users take to get to these primitives; this enriches the representation of the compounds which are constructed out of these primitives. This is especially useful in the organization of object icons, in which, for example, the icon for "blow-dryer" may be reached under "hand-held device," "heat-producing device," or "personal device." These paths are also very important in retrieval, because they can guide generalization and specialization of search criteria by functioning as a semantic net of hierarchically organized classes, subclasses, and instances.

6.1. A Language for Action

The central problem of a descriptive language for temporal media is the representation of dynamic events. For video in particular, the challenge is to come up with techniques for representing and visualizing the complex structure of the actions of characters, objects, and cameras. There exists significant work in the formalization of temporal events in order to support inferencing about their interrelationships [1] and to facilitate the compression and retrieval of image sequences by indexing temporal and spatial changes [5, 16, 17]. Our work creates a representation of cinematic action which these and other techniques could be usefully applied to. For even if we had robust machine vision, temporal and spatial logics would still require a *representation* of the video content, because such a representation would determine the units these formalizations would operate on for indexing, compression, retrieval, and inferencing.

A representation of cinematic action for video retrieval and repurposing needs to focus on the granularity, reusability, and semantics of its units. In representing the action of bodies in space, the representation needs to support the hierarchical decomposition of its units both spatially and temporally. Spatial decomposition is supported by a representation that hierarchically orders the bodies and their parts which participate in an action. For example, in a complex action like driving an automobile, the arms, head, eyes, and legs all function independently. Temporal decomposition is enabled by a hierarchical organization of units, such that longer sequences of action can be broken down into their temporal subabstractions all the way down to their atomic units. In [29], Lenat points out the need for more than a purely temporal representation of events that would include semantically relevant atomic units organized into various temporal patterns (repeated cycles, scripts, etc.) For example, the atomic unit of “walking” would be “taking a step” which repeats cyclically. An atomic unit of “opening a jar” would be “turning the lid” (which itself could theoretically be broken down into smaller units—but much of the challenge of representing action is knowing what levels of granularity are useful).

Our approach tries to address these issues in multiple ways with special attention paid to the problems of representing human action as it appears in video. It is important to note in this regard—and this holds true for all aspects of representing the content of video—that unlike the project of traditional knowledge representation which seeks to represent the world, our project is *to represent a representation of the world.* This distinction has significant consequences for the representation of human action in video. In video, actions and their units do not have a fixed semantics, because their meaning can shift as the video is recut and inserted into new sequences [30, 27]. For example, a shot of two people shaking hands, if positioned at the beginning of a sequence depicting a business meeting, could represent "greeting," if positioned at the end, the same shot could represent "agreeing." Video brings to our attention the effects of context and order on the meaning of represented action. In addition, the prospect of annotating video for a global media archive brings forward an issue which traditional knowledge representation has largely ignored: cultural variance. The shot of two people shaking hands may signify greeting or agreeing in some cultures, but in others it does not. How are we to annotate shots of people bowing, shaking hands, waving hello and good-bye? The list goes on. In order to address the representational challenges of action in video we do not explicitly annotate actions according to their particular semantics in a given video stream (a shot of two people shaking hands is not annotated as "greeting" or alternately as "agreeing"), but rather according to the motion of objects and people in space. We annotate using physically-based description in order to support the reuse of annotated video in different contexts—be they cinematic or cultural ones. We create analogy mappings between these physically-based annotations in their concrete contexts in order to represent their contextual synonymy or lack thereof.

6.2. Character Actions and Object Actions

We subdivide character actions *horizontally* into full body actions, head actions, arm actions, and leg actions (See Figure 4). Under each of these categories of human action (and their own subdivisions) action is represented in two ways:

- conventionalized physical motions
- abstract physical motions

We built into our ontology many commonly occurring, complex patterns of human motion which seem to have cross-cultural importance (e.g., walking, sitting, eating, talking, etc.). We also provide a hierarchical decomposition of the possible motions of the human body according to articulations and rotations of joints. Since Media Streams enables multi-layered annotation, any pattern of human motion can be described with precision by layering temporally indexed descriptions of the motion of various human body parts.

 Object actions are subdivided *horizontally* into actions involving a single object, two objects, or groups of objects (See Figure 5). Each of these is divided according to object *motions* and object *state changes*. For example, the action of a ball rolling is an object motion; the action of a ball burning is an object state change.

We represent actions for characters and objects separately in the Director's Workshop because of the unique actions afforded by the human form. Our icons for action are *animated* which takes advantage of the affordances of iconography in the computer medium as opposed to those of traditional graphic arts.

6.3. Characters and Objects

 Characters are subdivided*vertically* into characters (female, male, unknown gender, non-human, and crowd), occupations (personal care, commercial, institutional, religious, sports) and number (one, two, three...many) (See Figure 6). Characters do not have "essential" identities in cinema. Characters are what they seem to be. For our purposes, someone dressed like a doctor *is* a doctor. Marcus Welby is an MD.

 Objects are subdivided *vertically* into various types of objects and number of objects.

6.4. Relative Positions

 Relative positions are used to describe the spatial relationship between characters and objects and are subdivided *horizontally* into inside, on the threshold of, outside, on top of, underneath, above, and below.

6.5. Mise-En-Scene: Time, Space, and Weather

 Time is subdivided *vertically* into historical period (from the age of the dinosaurs through the twentieth century on into the future), time of year (spring, summer, fall, and winter), and time of day or night (morning, afternoon, sunset, midnight, etc.) (See Figure 7).

 Space is subdivided *vertically* into geographical space (land, sea, air, and outer space), functional space (buildings, public outdoor spaces, wilderness, and vehicles), and topological space (inside, outside, above, behind, underneath, etc.) (See Figure 8).

 Weather is subdivided *vertically* into moisture (clear, partly sunny, partly cloudy, overcast, rainy, and snowy) and wind (no wind, slight wind, moderate wind, and heavy wind) (See Figure 9). Temperature is not something that can be directly seen. A video of a cold clear day may look exactly like a video of a hot clear day. It is the presence of snow or ice that indirectly indicates the temperature.

We use these icons to represent two very different types of space, time, and weather on a Media Time Line: the actual time, space, and weather of the recorded video and the visually inferable time, space, and weather of the video. The difference can be made clear in the following example. Imagine a shot of a dark alley in Paris that looks like a generic dark alley of any industrialized city in Europe (it has no distinguishing signs in the video image which would identify it as a *Parisian* dark alley). The actual recorded time, space, and weather for this shot differ from its visually inferable time, space, and weather. This distinction is vital to any representation for reusable archives of video data, because it captures both the scope within which a piece of video can be reused and the representality of a piece of video, i.e., some shots are more representative of their actual recorded time, space, and weather than others.

6.6. Cinematography

 Through discussion with people who have everyday experience with Hollywood production and by researching camera description languages in film theory [10], we have developed a camera language which is both comprehensive and precise. In order to represent the cinematographic aspects of video we conceptualize the motion of the recording device which produced the images which the annotator sees. In cinema, the recording device typically has three interconnected parts which move independently to produce complex camera motions. The lens of the camera moves (to create different framings, zooms, etc.), what the camera is on—either a tripod or someone's hand—moves (to create pans, to track a moving figure), and what what the camera is on—a "truck" or "dolly" in cinematic terms, or someone's legs, or even a vehicle as in the case of shots taken from a moving car—may move as well (to create truck in, truck out, etc.). Each part of the recording device may also have important states as in the focus, camera angle, camera height, etc. In Media Streams, camera motions are subdivided *horizontally* into "lens" actions (framing, focus, exposure), "tripod" actions (angle,

canting, motion), and "truck" actions (height and motion) (See Figure 10). By layering these iconic descriptors on the Media Time Line, the user can describe simple to very complex camera motions.

6.7. Recording Medium

In addition to representing the motions and states of the recording device we also can represent the "look" of the recording medium. Icons for recording medium are subdivided *vertically* into stock (70 mm film, 8mm video, etc.), color quality (color, black & white, sepia, etc.), and graininess (fine, medium, coarse, etc.) (See Figure 11).

6.8. Screen Positions

Screen positions are subdivided *horizontally* into two-dimensional screen position and screen depth.

6.9. Thoughts

Archivists, for example, would tell us that producers would come to them with queries for footage, saying: "Get me something with a lot of action in it!" Or, regarding the framing of a shot: "I want a well composed shot of three Japanese kids sitting on some steps in Tokyo." These subjective assessments about the qualities of video are addressed in our representation by thoughts' icons which are subdivided *vertically* into thoughts about the screen (framing, activity, color) and evaluation (from three thumbs up to three thumbs down).

6.10. Transitions

The icon categories described above enable the user to produce representations of the content of video at the shot level. Transitions between shots are both the tools editors use to construct scenes and sequences out of a series of shots, and the gaps in a

video stream of recorded space-time which are bridged by the viewer's inferential activity [9, 10]. For example, if a viewer sees the two shot sequence:

Shot 1: person enters elevator, elevator doors close

Shot 2: elevator doors open, person exits elevator

The viewer infers that a certain amount of time has passed and that a certain type of spatial translation has occurred. Noel Burch has developed a systematic categorization of spatio-temporal transitions between shots in cinema [11]. He divides temporal transitions into continuous, forward ellipses in time of a determinate length, forward ellipses of an indeterminate length, and the corresponding transitions in which there is a temporal reversal. Spatial transitions are divided into continuous, transitions in which spatial proximity is determinate, and transitions in which spatial proximity is indeterminate. Burch's categorization scheme was used by Gilles Bloch in his groundbreaking work in the automatic construction of cinematic narratives [8]. We adopt and extend Burch's categorization of shot transitions by adding "temporal overlaps" as a type of temporal transition and the category of "visual transitions" for describing transition effects which unlike traditional cuts, can themselves have a duration (icons for transition effects which have durations are animated icons). In the Director's Workshop, we *horizontally* subdivide transitions between shots according to temporal transitions, spatial transitions, and visual transitions (cuts, wipes, fades, etc.) (See Figure 12).

When a transition icon is dropped on the Media Time Line, Media Streams creates a compound icon in which the first icon is an icon-sized (32 x 32 pixels, 24 bits deep) QuickTime Movie containing the first shot, the second icon is the transition icon, and the third icon is an icon-sized QuickTime Movie containing the shot after the

transition. So returning to our example of the two-shot elevator sequence, the compound icons would be as follows:

Temporal Transition
(forward temporal ellipsis of a determinate length)

Spatial Transition
(spatial translation of a determinate proximity)

Visual Transition
(simple cut with no duration)

We intend to use transition icons to improve Media Streams' knowledge about the world and to facilitate new forms of analogical retrieval. A search using the icons above would enable the user to find a "matching" shot in the following way. The user could begin with a shot of a person getting into an automobile and use one or more of the transition icons as analogical search guides in order to retrieve a shot of the person exiting the automobile in a nearby location. The query would have expressed the idea of "find me a Shot B which has a similar relation to Shot A as Shot D has to Shot C."

Figure 7: Time

Figure 8: Space

Figure 9: Weather

Figure 6: Characters

Figure 4: Character Actions

Figure 5: Object Actions

Figure 11: Recording Medium

Figure 10: Cinematography

Figure 12: Transitions

6.11. Extensibility of the Icon Language

Currently, we have two ways of extending the iconic visual language of Media Streams beyond the composition of iconic primitives. Icons and the components of compound icons can be titled in the Icon Title Editor of the Icon Information Editor (See Figure 13). This enables the user to attain a level of specificity of representation while still making use of the generality and abstraction of icons. For example, if the user were to annotate video of an automobile with the descriptor "XJ7," this description may be very opaque. If, however, the user titles a car icon "XJ7," in addition to the computer learning that XJ7 is a type of car, a human reading this annotation can see simply and quickly the similarity between an XJ7 and other types of automobiles. A form of system maintenance would be to periodically find titles for which there are many occurrences and create an icon for them.

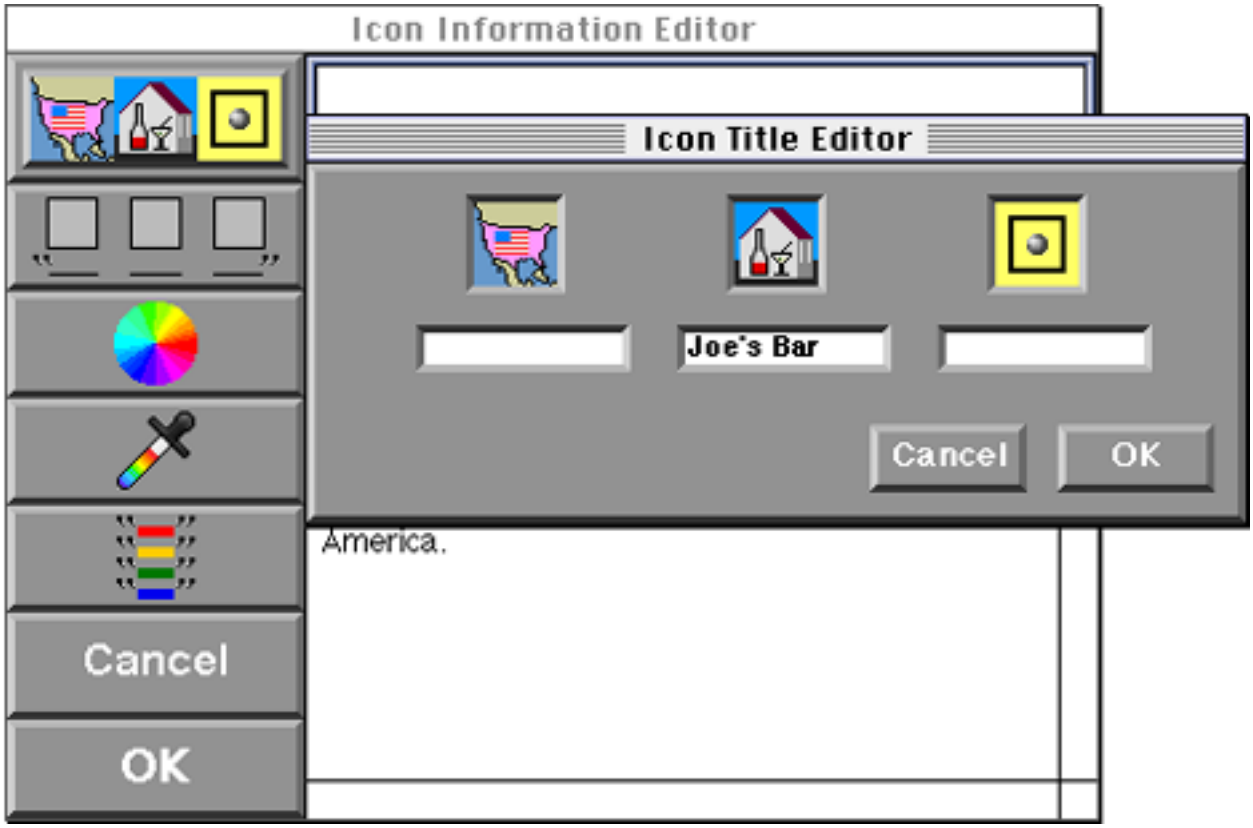


Figure 13: Icon Title Editor

Users can also create new icons for character and object actions by means of the Animated Icon Editor (See Figure 14). This editor allows users to define new icons as subsets or mixtures of existing animated icons. This is very useful in conjunction with our complete body model, because a very wide range of possible human motions can be described as subsets or mixtures of existing icons.

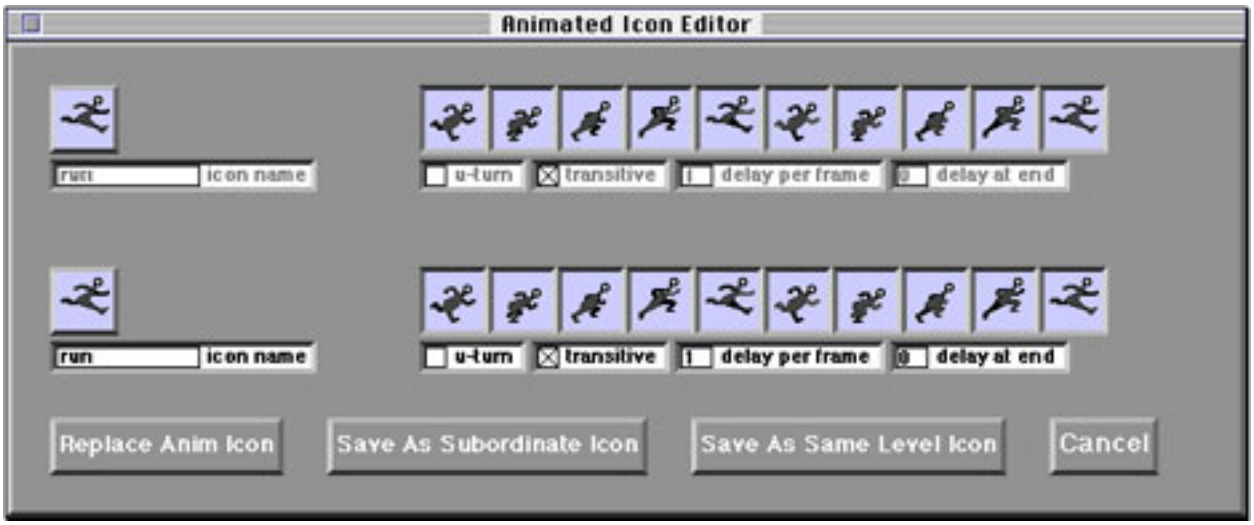


Figure 14: Animated Icon Editor

Applying the results of work on automatic icon incorporation would be a fruitful path of exploration [22]. Already in our icon language, there are many iconic descriptors which we designed using the principle of incorporation (by which individual iconic elements are combined to form new icons). Creating tools to allow users to automatically extend the language in this way is a logical extension of our work in this area.

7 Media Time Lines

The Media Time Line is the core browser and viewer of Media Streams (See Figure 3). It enables users to visualize video at multiple timescales simultaneously, to read and write multi-layered iconic annotations, and provides one consistent interface for annotation, browsing, query, and editing of video and audio data.

Since video is a temporal medium, the first challenge for representing and annotating its content is to visualize its content and structure. In the Media Time Line we represent video at multiple timescales simultaneously by trading off temporal and spatial resolution in order to visualize both the content and the dynamics of the video data. We create a sequence of thumbnails of the video stream by subsampling the video stream one frame every second. For longer movies, we sample a frame every minute as well. The spatial resolution of each thumbnail enables the user to visually inspect its contents. However, the temporal resolution is not as informative in that the sequence is being subsampled at one frame per second.

In order to overcome the lack of temporal resolution, we extend a technique pioneered by Ron MacNeil of the Visible Language Workshop of the MIT Media Laboratory [31] and used in the work of Mills and Cohen at Apple Computer's Advanced Technology Group [33]. We create a videogram. A videogram is made by grabbing a center strip from every video frame and concatenating them together.

Underneath the subsampled thumbnail frames of the video is the videogram in which the concatenated strip provides fine temporal resolution of the dynamics of the content while sacrificing spatial resolution. Because camera operators often strive to leave significant information within the center of the frame, a salient trace of spatial resolution is preserved.

In a videogram, a still image has an unusual salience: if a camera pans across a scene and then a center strip is taken from each video frame, a still will be recreated which is coherently deformed by the pace and direction of the camera motion and/or the pace and direction of any moving objects within the frame. Our contribution is that by simultaneously presenting two different, but coordinated views of video data—the thumbnails, with good spatial resolution and poor temporal resolution, and the videogram, with poor spatial resolution but good temporal resolution—the system enables the viewer to use both representations simultaneously in order to visualize the structure of the video information (See Figure 15). This idea of playing *spatial* and *temporal* resolutions off one another is also utilized in Laura Teodosio's work on "salient stills" [36] and holds promise as a general guideline for creating new visualizations of video data. An example of this spatial/temporal tradeoff can be seen in the figure below in which the movement of Arnold through the frame is visible in the right hand side of the videogram and the fact that that swath of extended face corresponds to the central figure can be seen from the thumbnail above.

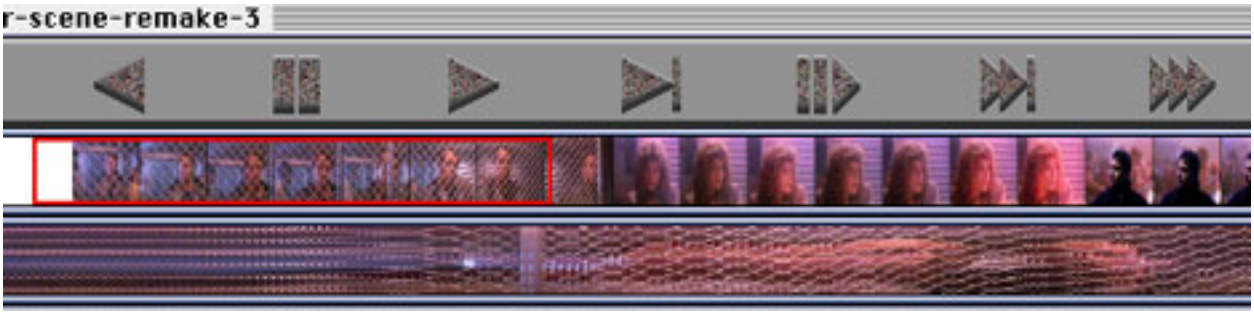


Figure 15: Media Time Line Detail—
Video Thumbnails and Videogram

With little practice, users can learn to read this representation to quickly scan the dynamics of video content from this spatial representation. Scene breaks are clearly visible as are camera pans, zooms, tracking, and the difference between handheld and tripod recorded video footage. The deformation of the still image in the videogram provides a signature of camera and/or object motion as in the example above.

Audio data in the media timeline is represented by a waveform depicting amplitude as well as pause bars depicting significant breaks in the audio. Currently our algorithm uses a set threshold which works fairly well for many videos but a more robust algorithm is needed. Significant work has been done by Barry Arons on pause detection and audio and speech parsing in general [6]; we hope to incorporate these results into our system. Arons' work uses dynamic threshholding and windowing techniques to facilitate better detection of pauses in speech and the separation of speech from background noise in unstructured audio recordings. Similarly, work by Michael

Hawley in developing specialized audio parsers for musical events in the audio track could be applied to automatically parsing the structure and enriching the representation of audio data [26].

In annotating the presence or absence of audio events within the data stream, our representation makes use of the fact that in thinking about audio, one thinks about the source that produced the audio. Icons for different objects and characters are compounded with the icon for the action of producing the heard sound in order to annotate audio events. This concept correlates to Christian Metz's notion of "aural objects" [32].

Annotation of video content in a Media Time Line is a simple process of dropping down iconic descriptors from the Icon Space onto the Media Time Line. Frame regions are then created which may extend to the end of the current scene or to the end of the entire movie. The select bar specifies the current position in a movie and displays the icons that are valid at that point in time. Icons are "good-til-canceled" when they are dropped onto the Media Time Line. The user can specify the end points of frame regions by dragging off an icon and can adjust the starting and ending points of frame regions by means of dragging the cursor. A description is built up by dropping down icons for the various categories of video annotation. The granularity and specificity of the annotation are user determined.

8 Conclusions and Future Work

Media Streams is about to be subjected to some rigorous real-world tests. In addition to several internal projects at the MIT Media Laboratory which will be building other systems on top of Media Streams, external projects involving large archives of news footage will be exploring using Media Streams for video annotation and retrieval. Clearly these internal and external projects will teach us much about the claim made in this paper: that an iconic visual language for video annotation and retrieval can support the creation of a stream-based, reusable, global archive of digital video. We believe that this goal articulates an important challenge and opportunity for visual languages in the 1990's [23].

Acknowledgments

The research discussed above was conducted at the MIT Media Laboratory and Interval Research Corporation. The support of the Laboratory and its sponsors is gratefully acknowledged. I want to thank Brian Williams and Golan Levin for their continually creative and Herculean efforts and my advisor, Prof. Kenneth Haase, for his insight, inspiration, and support. Thanks also to Warren Sack, Axil Comras, and Wendy Buffett for editorial and moral support.

Marc Davis is currently completing his Ph.D. from the MIT Media Laboratory while interning at Interval Research Corporation in Palo Alto, California.

References

1. Allen, J.F., *Maintaining Knowledge about Temporal Intervals,* in *Readings In Knowledge Representation,* R.J. Brachman and H.J. Levesque, Editor. Morgan Kaufmann Publishers, Inc.: San Mateo, California. p. 510-521. 1985.

2. Apple Computer, *Macintosh Common Lisp Reference*. Cupertino, California: Apple Computer. 1993.

3. Apple Computer, *QuickTime Developer's Guide*. Cupertino, California: Apple Computer. 1993.

4. Apple Multimedia Lab, *The Visual Almanac*. San Francisco: Apple Computer. 1989.

5. Arndt, T. and S.-K. Chang. "Image Sequence Compression by Iconic Indexing." In: *Proceedings of 1989 IEEE Workshop on Visual Languages*. Rome, Italy: IEEE Computer Society Press. p. 177-182. 1989.

6. Arons, B. "SpeechSkimmer: Interactively Skimming Recorded Speech." Forthcoming in: *Proceedings of UIST'93 ACM Symposium on User Interface Software Technology*. Atlanta: ACM Press. 1993.

7. Bliss, C.K., *Semantography-Blissymbolics*. 3rd ed. Sydney, N.S.W., Australia: Semantography-Blissymbolics Publications. 1978.

8. Bloch, G.R., *From Concepts to Film Sequences*. Unpublished Paper. Yale University Department of Computer Science: 1987.

9. Bordwell, D., *Narration in the Fiction Film*. Madison: University of Wisconsin Press. 1985.

10. Bordwell, D. and K. Thompson, *Film Art: An Introduction*. 3rd ed. New York: McGraw-Hill Publishing Company. 1990.

11. Burch, N., *Theory of Film Practice*. Princeton: Princeton University Press. 1969.

12. Chang, S.-K., *Visual Languages and Iconic Languages,* in *Visual Languages,* S.-K. Chang, T. Ichikawa, andP.A. Ligomenides, Editor. Plenum Press: New York. p. 1-7. 1986.

13. Chang, S.K., *et al.* "A Methodology for Iconic Language Design with Application to Augmentative Commmunication." In: *Proceedings of 1992 IEEE Workshop on Visual Languages*. Seattle, Washington: IEEE Computer Society Press. p. 110-116. 1992.

14. Davis, M. "Director's Workshop: Semantic Video Logging with Intelligent Icons." In: *Proceedings of AAAI-91 Workshop on Intelligent Multimedia Interfaces*. Anaheim, California: AAAI Press. p. 122-132. 1991.

15. Davis, M. "Media Streams: An Iconic Visual Language for Video Annotation." In: *Proceedings of 1993 IEEE Symposium on Visual Languages*. Bergen, Norway: IEEE Computer Society Press. p. 196-202. 1993.

16. Del Bimbo, A., E. Vicario, and D. Zingoni. "A Spatio-Temporal Logic for Image Sequence Coding and Retrieval." In: *Proceedings of 1992 IEEE Workshop on Visual Languages*. Seattle, Washington: IEEE Computer Society Press. p. 228-230. 1992.

17. Del Bimbo, A., E. Vicario, and D. Zingoni. "Sequence Retrieval by Contents through Spatio Temporal Indexing." In: *Proceedings of 1993 IEEE Symposium on Visual Languages*. Bergen, Norway: IEEE Computer Society Press. p. 88-92. 1993.

18. Dreyfuss, H., *Symbol Sourcebook: An Authoritative Guide to International Graphic Symbols*. New York: McGraw-Hill. 1972.

19. Eisenstein, S.M., *The Film Sense*. San Diego: Harcourt Brace Jovanovich, Publishers. 1947.

20. Eisenstein, S.M., *Film Form: Essays in Film Theory*. San Diego: Harcourt Brace Jovanovich, Publishers. 1949.

21. Elliott, E.L., *WATCH • GRAB • ARRANGE • SEE: Thinking with Motion Images via Streams and Collages*. M.S.V.S. Thesis. Massachusetts Institute of Technology Media Laboratory: 1993.

22. Fuji, H. and R.R. Korfhage. "Features and a Model for Icon Morphological Transformation." In: *Proceedings of 1991 IEEE Workshop on Visual Languages*. Kobe, Japan: IEEE Computer Society Press. p. 240-245. 1991.

23. Glinert, E., M.M. Blattner, and C.J. Frerking. "Visual Tool and Languages: Directions for the 90's." In: *Proceedings of 1991 IEEE Workshop on Visual Languages*. Kobe, Japan: IEEE Computer Society Press. p. 89-95. 1991.

24. Haase, K., *FRAMER: A Persistent Portable Representation Library*. Internal Document. MIT Media Laboratory: Cambridge, Massachusetts. 1993.

25. Haase, K. and W. Sack, *FRAMER Manual*. Internal Document. MIT Media Laboratory: Cambridge, Massachusetts. 1993.

26. Hawley, M., *Structure out of Sound*. Ph.D. Thesis. Massachusetts Insitute of Technology: 1993.

27. Isenhour, J.P., "The Effects of Context and Order in Film Editing." *AV Communications Review*, 23(1): p. 69-80. 1975.

28. Korfhage, R.R. and M.A. Korfhage, *Criteria for Iconic Languages,* in *Visual Languages,* S.-K. Chang, T. Ichikawa, and P.A. Ligomenides, Editor. Plenum Press: New York. p. 207-231. 1986.

29. Lenat, D.B. and R.V. Guha, *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Reading, Massachusetts: Addison-Wesley Publishing Company, Inc. 1990.

30. Levaco, R., ed. *Kuleshov on Film: Writings by Lev Kuleshov*. University of California Press: Berkeley. 1974.

31. MacNeil, R. "Generating Multimedia Presentations Automatically Using TYRO: the Constraint, Case-Based Designer's Apprentice." In: *Proceedings of 1991 IEEE Workshop on Visual Languages*. Kobe, Japan: IEEE Computer Society Press. p. 74-79. 1991.

32. Metz, C., "Aural Objects." *Yale French Studies*, 60: p. 24-32. 1980.

33. Mills, M., J. Cohen, and Y.Y. Wong. "A Magnifier Tool for Video Data." In: *Proceedings of CHI'92*. Monterey, California: ACM Press. p. 93-98. 1992.

34. Otsuji, K., Y. Tonomura, and Y. Ohba, "Video Browsing Using Brightness Data." *SPIE Visual Communications and Image Processing '91: Image Processing*, SPIE 1606: p. 980-989. 1991.

35. Tanimoto, S.L. and M.S. Runyan, *PLAY: An Iconic Programming System for Children,* in *Visual Languages,* S.K. Chang, T. Ichikawa, and P.A. Ligomenides, Editors. Plenum Press: New York. p. 191-205. 1986.

36. Teodosio, L., *Salient Stills*. M.S.V.S. Thesis. Massachusetts Institute of Technology Media Laboratory: 1992.

37. Tonomura, Y. and S. Abe. "Content Oriented Visual Interface Using Video Icons for Visual Database Systems." In: *Proceedings of 1989 IEEE Workshop on Visual Languages*. Rome, Italy: IEEE Computer Society Press. p. 68-73. 1989.

38. Tonomura, Y., *et al.* "VideoMAP and VideoSpaceIcon: Tools for Anatomizing Content." In: *Proceedings of INTERCHI'93 Conference on Human Factors in Computing Systems*. Amsterdam, The Netherlands: ACM Press. p. 131-136. 1993.

39. Ueda, H., *et al.* "Automatic Structure Visualization for Video Editing." In: *Proceedings of INTERCHI'93 Conference on Human Factors in Computing Systems*. Amsterdam, The Netherlands: ACM Press. p. 137-141. 1993.

40. Ueda, H., T. Miyatake, and S. Yoshizawa. "IMPACT: An Interactive Natural-Motion-Picture Dedicated Multimedia Authoring System." In: *Proceedings of CHI '91*. New Orleans, Louisiana: ACM Press. p. 343-350. 1991.

41. Zhang, H., A. Kankanhalli, and S.W. Smoliar, "Automatic Partitioning of Full-Motion Video." *Multimedia Systems*, 1: p. 10-28. 1993.