Hypermedia Data Modeling, Coding, and Semiotics

RUBEN GONZALEZ, MEMBER, IEEE

This paper reviews the key issues in hypermedia systems as an overture to the proposal of a new semiotic paradigm for hypermedia data and coding models. The hypertext concept permits users to interact with and manage data as high-level conceptual objects rather than as symbol streams. Current hypermedia systems can best be defined as an amalgamation of hypertext and multimedia. While the hypertext data model enables this goal, that is not true for the data models of other media forms. A new semiotic paradigm that addresses these deficiencies and supports object-oriented interaction with compressed multimedia streams is proposed. This paper initially presents an overview of the hypertext data model, contrasting it with existing multimedia data and coding models. The framework for the new paradigm is then presented in a brief review of cognitive, psychological, and semiotic principles. This analysis culminates in the proposal of semiotically based data models and representations predisposed to the hypermedia paradigm.

Keywords*—*Audio coding, data models, hypermedia, hypertext systems, image coding, information retrieval, multimedia information systems, psychology, semiotics, signal representations, source coding.

I. INTRODUCTION

More than 50 years after its inception, hypermedia is finally on the verge of becoming a reality. This can be observed in the current popularity of multimedia-enhanced hypertext systems such as the World Wide Web. These enhanced hypertext systems do not correspond to true hypermedia systems since the data models used for the multimedia data do not have the required characteristics. The main problem is the reliance on stream-based, unstructured representations. In this paper, a distinction is drawn between pure hypertext, multimedia-enhanced hypertext, and true hypermedia. This paper investigates the issues central to the development of true hypermedia and attempts to answer the question of how stream-based media can be converted to structured representations.

To answer this question, we must first understand what makes hypertext distinct from more conventional information systems. We also need to understand the shortcomings of existing multimedia technologies to meet these requirements. Then, to move beyond the existing technologies, we shall step back and review salient cognitive and semiotic issues that are fundamental to hypermedia. Semiotics is the study of the role of signs in communication and understanding. From this investigation, a new semiotic paradigm will be proposed as the basis for the next generation of true hypermedia systems. Last, this paper proposes specific data models for multimedia data concurring with the new paradigm and presents rudimentary coding schemes based on these models.

Manuscript received July 25, 1996; revised April 10, 1997.

The author was with the Institute of Telecommunications Research Center, University of Wollongong, Australia. He is now with Griffith University, Gold Coast Campus, School of Information Technology, Australia.

Publisher Item Identifier S 0018-9219(97)05308-5.

A. Data Models, Coding, and Representations

The critical component in, and identifying feature of, an information source or system is its data model. The data model determines the capabilities of the system by defining the nature of its elemental components and defining or delimiting any relationships and interactions both among and with these components. For any given data model, various distinct representations may be feasible. The representation scheme determines the accessibility to the elemental components, compaction, interactive manipulability, and the decoding complexity of encoded data. In this context, the role of data models in information systems and coding models in coding schemes are essentially identical.

The difference between a data or coding model and its representation is that the model specifies *what* elements are in terms of which the data is to be encoded and their organization. The representation determines *how* these elements are encoded. For example, given a data model consisting of a collection of smooth curves, each curve may be represented as a list of polynomial coefficients, a chain code, or a string of coordinates. Alternatively, in the specific case of the coefficients, these may be stored as scaled integers or normalized rational numbers, or even written out textually.

Given that we know the general characteristics of the data model required for hypertext [1], we would like to develop multimedia representations based on this model. We denominate these representations hypermedia, denoting the distinction between hypertext and normal text. The

Table 1

Approaches	Information Agents
Statistical	Statistically significant symbol or measure
Syntactical	Relationships among structural elements
Semantic	Abstract human dependent meaning

problem remains of determining modality-specific mappings from the domain of each medium into this general data model. Three distinct approaches to this problem are possible: statistical, syntactical, or semantic (refer to Table 1). Syntax concerns only the relationships among symbols and the ways in which they can be manipulated, while semantics concerns the relationships among symbols and their human-dependent meanings. The traditional engineering approach to audio and video data processing has been statistical through signal processing techniques. Since statistical methods alone are unable to generate the required mappings, these have often been supplemented with semantic processes. Little attention has been given to use of syntactical methods for this purpose.

Assuming that we could generate such mappings, we then also need to specify 1) adequate representations that allow direct and independent access to each component object in the representation and 2) encoding techniques to generate these representations automatically from the raw data. This would result in encoded data that is structured and interactively manipulable.

Existing data models for multimedia information management have evolved from traditional database, semantic modeling approaches [2], [3] for which automatic processing may be impossible. In these, the data model is foreign to the data itself. These models only treat multimedia data as separate renditions of given semantic entities [4], completely separating the layout and logical structures from the conceptual structure [5]. In this paper, we consider data models where the representation itself encapsulates both the logical and conceptual structures, eliminating the need for multiple structures. An early attempt at using structured data representations was the Multos multimedia system [6]. This system was based on using object recognition to build separate conceptual structures of images. While the aims of the Multos system in attempting to handle both images and text consistently were excellent, its use of semantic methods limited it to the recognition of synthetic vector graphic images.

B. Proposed Approach

Semantic methods have played a dominant role in multimedia information systems in the form of either direct human intervention or constrained automatic object recognition. Semantic methods require knowledge of what an entity is before any action can be taken toward or with it. More than just a matching process, recognition involves the unambiguous interpretation of data to identify and associate objects with appropriate attributes in a given knowledge base. A constant need for knowledge about the definition of new objects and their properties is required to contend with unfamiliar environments. This limits the use of unsupervised semantic methods as a general tool.

Information exists and can be defined at various levels. In its most basic and raw form, a given data stream (such as from radio astronomy) can be analyzed statistically to determine the existence of any significant components. Assuming the absence of noise in the process, these components are symbols that may occur according to predefined relationships among themselves. The syntax exhibited by these elements defines or infers a grammar that creates a context for each symbol even in the absence of prior knowledge. The appreciation of the symbols within their grammatical contexts gives rise to meaning or semantic information. For example, in its simplest form, speech can be described in terms of temporal variations in a spectral energy distribution. The statistically significant components that largely comprise formats may be identified. These combine to form phonemes, which in turn combine more or less syntactically to form semantically significant words, phrases, and sentences.

This paper advocates the proposition that syntax, not semantics, is the key to converting stream-based media into hypermedia automatically. Unlike semantics, automatic syntactic analysis does not require any external or prior knowledge. The versatility of syntax is that while it can exist on its own, independent of any human interpretation or intervention, the argument can also be made that semantic understanding can arise from syntactical analysis [94]. Using a syntactical approach, we can potentially generate systems with semantic meaning automatically, although the meaning itself is unknown to the syntactical process. One specific question this paper will attempt to answer is: What is the nature of the syntactic elements for formulating appropriate hypermedia data models?

C. Paper Outline

To establish an appropriate context for the semiotic paradigm, this paper surveys a number of areas. Section II commences with an introductory review of hypermedia, its underlying data model, and the existing deficiencies in its realization. Section III discusses multimedia information systems, their access methods, and their implied data models. Section IV reviews the data models underlying current multimedia coding schemes. Section V leads up to the new paradigm by reviewing salient issues in cognition, psychology, and semiotics. It explores the nature of structured data representations in the early perceptual processes and discusses the role of Gestalt phenomena in their generation. Section VI summarizes the requirements for and presents the new semiotic paradigm. Multimedia data models and preliminary representations based on a semiotic articulation are then proposed and discussed.

II. HYPERMEDIA SYSTEMS

This section introduces hypertext and hypermedia systems. Section II-A outlines their historical development. Section II-B discusses the underlying cognitive issues and

objectives. Section II-C describes the general data model. Sections II-D and II-E explore multimedia extensions and their deficiencies, respectively. Section II-F discusses what is outstanding from Vannevar Bush's original vision.

A. Historical Context

The concept of hypertext and hypermedia is not a recent development. In 1945, Bush proposed a machine for storing, browsing, and annotating information on an extensive on-line graphical system supporting both text and pictures [7]. The purpose of this system was to manage the ever growing amount of information and scientific literature that was becoming unmanageable even then. He called this machine the "memex."

One essential feature of this system was its ability to link together items within and between multimedia documents in a manner Bush called "trail building." This linking process is the central mechanism for supporting associative indexing as a supplement to conventional indexing schemes. This reflects the associative recall and random access of the human mind. Bush realized that many technological breakthroughs were required to make the "memex" a reality.

Almost 20 years later, in 1963, inspired by Bush's ideas, Engelbart [8] also anticipated a system for augmenting the capabilities of the human mind. This system was to support high-resolution three-dimensional (3-D) graphics display and the visual manipulation of concepts and ideas as symbols. Since machines capable of delivering and manipulating multimedia information were not available at the time, the concepts espoused by Bush and Engelbart were first applied to text. The term hypertext was coined in 1965 by Ted Nelson [9] to describe these text-only systems.

B. Hypertext Concepts and Aim

The hypertext concept is based on a cognitive model of the communication process. This model defines a procession of distinct stages in cognition that transform a linear message into a nonlinear network of ideas in the mind. Simplistically, in the case of reading comprehension, it starts by recognizing the constituent symbols (signs) in a text string. The relationships between these signs are evaluated, isolating the concepts presented. These concepts are structured hierarchically and absorbed into long-term memory as a network of ideas. The two predominant characteristics of this process are the grouping of symbols into conceptual units and the formation of relationships between them.

Modeling and representing a text according to semantic or conceptual units rather than lexical units in this way allows it to be manipulated and accessed as a collection of related ideas and not just as a string of letters. This abstraction is a powerful tool for information management because it allows interaction with a given body of text at a much higher level. The interaction can revolve around what message is being conveyed and not how it is being conveyed. It allows the manipulation of the information structure without needing to deal with the information itself.

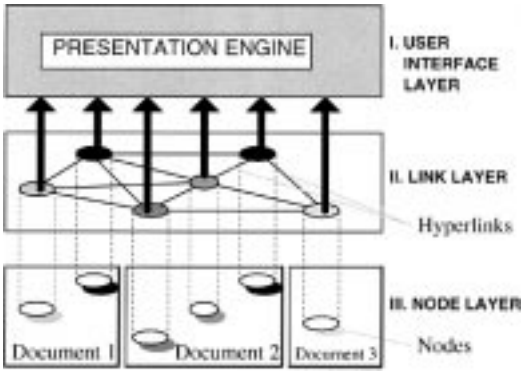


Fig. 1. Hypermedia system architecture.

Accordingly, hypertext assists its reader in the process of transforming knowledge from the primitive form of a symbol stream into the network- or graph-style structure used in the latter stages of cognition.

C. Hypertext Data Model

Hypertext abstracts textual data into a set of nodes and links representing, respectively, conceptual units and the relationships between them. To create a hypertext, a given body of text first must be manually partitioned or chunked into a set of nodes, also known as frames or cards. These are self-contained units of information, each encompassing a specific concept. The relationships may then be formed by connecting the nodes with hyperlinks. The origin of a hyperlink is some anchor point, typically a key word or expression within a given reference node. The destination of a hyperlink is generally another node but may also be another anchor point within a node. Various attributes, such as its type, may be attached to a link specifying the nature of the relationship it defines, its directionality, or any activation conditions. The total set of these links forms what is known as a web or hypergraph. By activating individual links, users can navigate through the information network or cognitive space defined by the hypergraph. In addition to link traversal, nodes also may be retrieved individually through structured browsing or query-based searching, depending on the system.

This simple node- and link-based data model is central to hypertext. This model is devoid of any information on how the data is to be presented or rendered, such as font selection and text layout. These details must be considered and encapsulated separately, although each node is generally presented within a separate view window. This creates the basic three-layer structure of hypertext system architectures depicted in Fig. 1. The presentation or user-interface layer controls the presentation of the data and supplies an interface to perform the navigation. The link layer contains and manages the relationships between nodes, and the node layer contains various appropriately structured documents. A hypermedia engine typically manages all three layers simultaneously.

The Dexter reference model [10] refers to these three layers as the run-time, within-component, and storage lay-

ers, respectively. Accordingly, the creation of hypertext systems involves three distinct phases that are not always decoupled in practice. The initial node-authoring phase is the process of segmenting or chunking the raw data into a structured collection of nodes. Next is the design of the data presentation or rendition. Last, the link-authoring phase involves defining the relationships between the nodes by defining anchor points and creating links.

D. Multimedia Extensions

Early efforts to incorporate multimedia information into existing hypertext systems were initially restricted to treating pictures as single destination nodes not containing any anchor points. Similar support for audio and video data was later added, permitting only sequential access and limiting interaction to playing, stopping, or pausing. This limitation was due to the unstructured data models used for the continuous media. Being only bit streams, these models do not provide any referenceable components within the streams that may serve as anchor points or link destinations. This situation is contrary to the primary goal of hypertext, which is to provide nonsequential access.

A higher level of interactivity in the form of clickable pictures was eventually fulfilled through the use of image maps. These are manually generated overlays specifying hot regions serving as anchor points. Through the mediation of an image map, the semblance of structured representation can be projected onto the unstructured data. The information contained by the overlays is separate from the picture itself, with the demarcation of nodes in overlay-based systems falling within the presentation layer rather than the node layer. This violates the Dexter model, which specifies a structured representation where the node demarcation is inherent to the data itself.

A similar approach has been taken to a certain extent with video and audio data. An example of this level of integration between multimedia and hypertext is the Amsterdam model [11], which has stream-based support for unstructured multimedia data. Another is the Hy-Time (ISO/IEC 10 744) standard, which is largely based on presentation-level integration through mapping the multimedia data at run time into a 3-D spatio-temporal presentation space. Since annotating continuous media in this manner is extremely tedious, large-scale deployment of these approaches is less prevalent. Generally, only time-based indexing is used for continuous media, limiting access to the frame level in video.

The evolution of this approach for integrating support for continuous media in the World Wide Web system is suitably represented by the Vosaic system [12]. Two specific goals of Vosaic were to address the lack of efficient 1) flexible access in the form of browsing, hierarchical access, and searching and 2) reuse of continuous media. The support for flexible access is provided by manually generated index files containing semantic information about the media stream. This textual annotation contains attributes specific to media and encoding schemes as well as frame-number-based structural information and indexing keywords. Intraframe hyperlinks are supported by specifying the location of hot regions at start and end frames to be linearly interpolated over the interval.

E. Multimedia Deficiencies

The CCITT/ISO standard techniques for encoding multimedia data used in most hypermedia systems include the Joint Photographic Experts Group (JPEG) algorithm for still images and the related Motion Pictures Experts Group (MPEG)-1 algorithm for video. The MPEG requirement specification stated that it should provide functionality similar to that "normally associated with VCR's." This objective is far from the kind of functionality that hypermedia demands. While limited interaction with partially decoded MPEG and JPEG data streams can occur, this is confined to the frequency domain. With current coding techniques, unrestricted data access is possible only after fully decoding the compressed stream. This is because compression has been the only objective in the development of multimedia data representations, without consideration of information-management issues. Even after decoding enables access, further processing is required to actually extract salient information. While flexible access can be manually supported through separate index files, this should be intrinsic to the encoded data and fully automatic.

It would be unfortunate to think of hypermedia as an amalgamation of old technologies. Rather, its interdisciplinary nature places new and challenging demands on existing technologies, provoking the development of new technologies where the old are incapable of meeting them. Existing multimedia data representations are clearly inadequate in this sense. The recognition of these deficiencies is evidenced in the fundamental goal of the upcoming MPEG-4 standard [13], which is "[t]o efficiently code interactive 2D and 3D environments consisting of real-time audio, video, and synthetic objects" supporting interaction for "individual objects rather than at the level of the composited video frame."

Of the various recent developments in the area of multimedia-enhanced hypertext, only virtual reality mark-up language (VRML), which is a graphical counterpart to hypertext mark-up language (HTML), makes any real progress in the support for navigable nontextual media. It provides a highly interactive structured data representation based on an object-oriented data model. VRML exhibits qualities essential for true hypermedia, such as individually referenceable components and intrinsic support for flexible access and intramedia navigation. It is, however, essentially limited to synthetic 3-D graphics.

F. Trail Blazing

In his seminal paper "As We May Think" [7], Bush called for the creation of a new profession of what he called trail blazers. He defined this vocation as "the task of establishing useful trails through the enormous mass of the common record." Rather than binding users of his machine to an onerous and mundane task in order to incorporate new material into his hypermedia system, new material

simply was supposed to "drop into place." Trail blazers were then to form link trails relatively effortlessly in the data space. According to Bush, "the users of it are free to use their brains for something more than repetitive detailed transformations in accordance with established rules."

It is noteworthy to observe here that these statements do not actually reflect the current process of hypermedia authoring. While systems capable of automatically generating hypertext with some success have emerged, this does not extend to multimedia data. Rather than spending their time creating links and trails, hypermedia authors typically spend a large amount of time in laborious manipulation of the underlying data, either restructuring it into nodes to permit the creation of links between them or generating intermediary overlay or metainformation files. A large component of this effort in demarcating node boundaries is highly repetitive and could be partially automated. Information management of the node database in large systems is another part of this problem.

The discrepancy between Bush's vision about hypermedia authoring and the current situation is largely due to the inadequacy of compression technologies in the context of hypermedia information management. Bush makes a critical observation in his paper regarding the required developments in information systems that has been overlooked by many. While he agrees that "[c]ompression is important, however when it comes to costs," he further states that, "[m]ere compression, of course, is not enough; one needs not only to store a record but also to be able to consult with it, and this aspect of the matter comes later." This statement cuts right to the essence of the problem. The emphasis here is on compact manipulable data representations supporting direct access and information management rather than just on compression alone or information management alone.

Clearly, Bush envisioned that apart from being compact, data should also be manageable, permitting random and content-based (associative) interaction. This includes intrinsic support for arbitrary intramedia navigation rather than just intermedia navigation. This requires the existence of referenceable components in the data representation and the ability to label any nodes individually in any medium as a link source or destination. Both textual and nontextual data should be handled homogeneously, providing the ability to cut and paste objects between multimedia documents, as with text-based systems. Accordingly, it should be possible to restructure the data arbitrarily by adding, moving, or deleting nodes. This currently cannot be achieved with the data models used for multimedia data and their respective unstructured data representations. To achieve these goals in a multimedia data representation, each node should be indexible, randomly accessible, and individually decodable. The node and link structure must also be independent of presentation/application issues.

III. Multimedia Information Systems

This section surveys existing data models for multimedia information systems. Section III-A presents multimedia information management principles. Sections III-B, C, and D discuss the data models as well as structuring mechanisms for image, video, and audio databases, respectively. Section III-E critically evaluates the relevance of these methods for hypermedia data modeling.

A. Information Management

Hypermedia systems are specialized multimedia information management systems (MIMS) and hence share many fundamental problems. One is that of defining appropriate access mechanisms into data streams [14]. It is often better to access large data streams as a set of individual components rather than as a whole, requiring the segmentation of the data. The management of these components can then be facilitated by appropriately labeling and indexing them. The principle difference between hypermedia and conventional MIMS is that with MIMS, instead of modeling the internal information in the multimedia data, the data is typically used as a rendition of some entity of an externally imposed schema. Conversely, the data model should be intrinsic to the hypermedia data itself since this must consist of uniquely referenceable nodes to serve as link anchor points.

Other information-management problems are encountered in large hypermedia systems, such as resource discovery and content-based retrieval of multimedia data. Browsing or query-handling support is often provided to overcome these problems. Query-based access requires searching through indexes containing keys or labels consisting of some salient semantic, syntactic, or statistical attribute of each node. Alternatively, browsing requires the classification of the data within some given hierarchical organization. This organization may naturally exist within the data itself as a manifestation of some structural property of the medium or it may arise in a set of discrete, syntactically unrelated elements through categorizing the specific attributes of each element. Conventional MIMS generally utilize separate index files or metafiles to support this type of functionality, often relying on implicit and/or incidental data models and thereby avoiding the issues of structured representations. The following sections briefly review the data models used for each modality.

B. Image Databases

Traditional pictorial data management is based on manually annotating images as indivisible objects. These semantic textual annotations are highly dependent on both the annotator's choice of vocabulary and immediate context. This limits the scope of retrieval and impedes reuse in a different context. These annotations may also arise in a variety of indirect forms such as a preexisting file name, a caption [15], or the anchor text in HTML as used by the Harvest resource discovery system [16].

Automatic object recognition has also been used for labeling by extracting semantic descriptions from the images. This attempts to classify the interpretation of geometric structure from the image data into predefined semantic groups. Recognition typically is restricted to simple, pre-

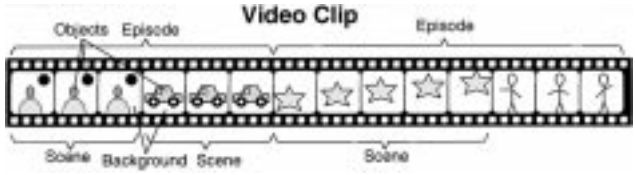


Fig. 2. Structure of a video.

defined polyhedra in highly constrained environments. The recognition process is not only equivocal and computationally intensive but the task of precisely specifying many individual objects is ponderous.

More recently, simpler statistical labeling techniques permitting inexact matching have been used. This precludes semantic-based queries but first- or second-order statistical labels can be generated quickly and automatically for unconstrained images without the need for prior knowledge. Labels are formed by extracting a number of attributes from images using a variety of statistical feature analysis routines. Typical attributes include the average global color, local variance or texture [17], or algebraic moments of the image. Searching can be performed by evaluating statistical similarity.

Hybrid approaches [18] combine semantic features from graphical annotations with statistical features such as color and texture. The manual graphical annotations are used to define the outlines of semantically consistent image regions. Labels may then be generated for individual objects in an image. The closed contour shape descriptions are often used for additional labeling information.

C. Video Computing

The predominant feature in video is its temporal structure. While individual frames provide the simplest and most common access mechanism, the importance of higher level mechanisms can be appreciated when one considers that a two-hour video typically consists of over half a million individual frames. Cognitively, people perceive episodes, scenes, and moving objects, as depicted in Fig. 2, but not individual frames. A scene in a video is a sequence of frames that are considered to be semantically consistent. Scene changes therefore demarcate changes in semantic context. Segmenting a video into its constituent scenes permits it to be accessed in terms of meaningful units.

As with still images, the initial indexing attempts were based on semantic methods [19]. Since manual annotation is clearly unsuitable for volume work, attention focused on template-based scene-recognition techniques. Subject to stringent spatio-temporal structural constraints, these methods can automatically perform both segmentation and labeling (Fig. 3). They are typically restricted to news broadcasts, which exhibit a high amount of regularity. In these cases, the demarcation between each different news item can be detected by the alternation between the regular spatial structure of the news room and the news footage [20], [21]. Character recognition is used to generate annotation text from the subtitles for each news

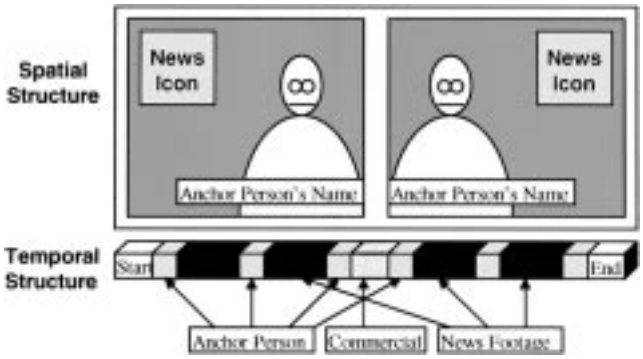


Fig. 3. Structure of a news broadcast.

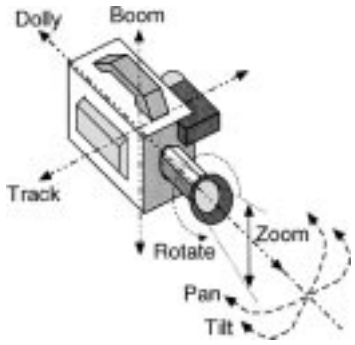


Fig. 4. Camera work.

item. Template matching, however, is too constrained for general use.

In spite of their semantic origins, scene boundaries may be detected automatically using statistical methods. The common techniques are based on frame difference analysis, pair-wise pixel comparison, or temporal variation of color composition [22]. Alternatively, the similarity of low-frequency images can be compared [23]. Nonlinear access to video is often supported in the form of temporally compressed browsing [24], where each individual scene is either represented as a micon (moving icon) or salient video still [25]. Scene labels may be generated from the attributes of a representative image of each scene or from the temporal properties of the scene. Scene aggregation and clustering may also be used to extract attributes regarding the relationships between scenes.

Within any scene, changes in global motion may be used to perform further segmentation, while the motion itself can be used as a generic labeling attribute. Shot classification is used for this dual purpose and involves determining the global motion induced by camera work and may include panning, tilting, zooming, tracking, booming, or dollying (Fig. 4). This can be performed by analyzing the structure of the flow field defined by motion vectors in motion-compensated video [26], optical flow analysis, or feature correspondence methods [27], among others [28].

Local motion can also be exploited for further segmentation and labeling. Since it is difficult to perform deformable object tracking under translation, rotation, and scaling, as well as occlusion, lighting, and background changes, many attempts have been limited to simple translational motion

with limited rotation [29]. These schemes often rely on an operator's tracing around the outline of each object to be tracked in the initial image, followed by a simple search to find matching areas in the succeeding frames. Simple regions generated automatically using edge, texture, histogram-splitting, or motion-based image-segmentation techniques [30] may also be tracked.

D. Audio Computing

Little support exists for nonspeech audio data since most of the work in this area has focused on recorded speech. While automatic speech recognition [31] would be an ideal solution for transforming linear speech into "hyperspeech," it unfortunately only works across a fairly narrow range of conditions. Hence, manual annotation in the form of synchronized transcripts is typically used to support nonsequential access. While it is useful for component labeling, speaker-dependent word spotting is also too constrained for general unsupervised use [32].

Other, less constrained statistical techniques also exist in speech processing such as detecting pauses, changes of speaker [33], gender identification, and possibly voicing and prosodic features [34]. These typically rely on evaluating energy measures, zero crossing rates, autocorrelation, and/or linear predictive coder (LPC) coefficients [35]. Similarly, simpler generic statistical methods may be used for arbitrary audio data. These are based on extracting features such as signal power, centroid (brightness), pitch, bandwidth, and harmonicity from the short-time Fourier spectrum (STFT) of the audio signal [36].

In the specific case of music, it is also possible to exploit the inherent organization contained in the music itself. Aigrain *et al.* [37] propose a representation of music based on a hierarchy of objects that are automatically delimited. This representation is composed of *strokes*, *patterns*, which are collections of up to 100 strokes, and *sections*, which are delimited by silence and/or scansions. The strokes are roughly equivalent to chords or notes. The continuity of the fundamentals can be used to delimit harmonic groupings. Individual notes may be detected as the local minima in the smoothed amplitude signal.

E. Intrinsic Information Management

Current MIMS try to organize data according to semantic or statistical criteria, often implicitly forming incidental data models. All support for information management is totally external to the data itself and based on antecedently generated indexes. The entities in these indexes form the basis of the data models so that the data itself as a rendition of an entity is only peripheral to the data model. Since the indexes and data are normally separate, support for information management is not intrinsic to the actual data but instead specific to the system application layer (or DBMS). This creates a problem with portability and reusability of the data, requiring the generation of new indexes whenever the data is reused in a new environment or system. This can only be overcome by appending or interleaving the index into the data stream and embedding the application as well [38]. In any case, the actual multimedia data representation remains unchanged and, therefore, unstructured.

This indexing information, however, constitutes a potentially large amount of storage overhead. Consider the amount of data required to index just one hour of video. A current method is based on storing binary image masks for each object in the video [18]. Assuming that for each frame only a single binary mask was used, the storage overhead would be 5% of the total video. This is quite a significant amount, approaching 3.5 Gb for 1 h of video. (i.e., $640 \times 480 \times 1$ b/frame $\times$ 25 f/s $\times$ 3600 s/h). Compressing the indexing data is only a partial solution since this would be offset by the increased query processing required to access the compressed index data.

Ideally, the source media representation itself should intrinsically support information management based on its data model without requiring a separate index. Such hypermedia data-representation schemes would encode the data in terms of a structure where the salient characteristics of its elements are explicit and directly accessible. This obliterates the overhead of storing persistent indexes separately and removes the need to decode and process the data before it can be manipulated. Such representations should not necessarily offer less compression than current coding technologies.

Except in highly constrained environments [39], syntactical methods have been largely overlooked as the basis of information management. The use of suitably abstracted syntactical over statistical information is cognitively more appropriate for similarity matching [40]. It is also more flexible and less constrained than semantic methods. A large component of the primary information required for this type of analysis in the visual domain is available through low-level vision techniques [41], [42].

IV. Multimedia Coding Models and Representations

This section reviews existing multimedia coding models and representation schemes. Section IV-A introduces the principles of multimedia coding and representations. Section IV-B discusses computer graphics models and representations. Sections IV-C, D, and E, respectively, review common audio-, image-, and video-coding schemes from the perspective of their assumed data models.

A. Coding Principles

Traditionally, the sole objective in audio, image, and video coding has been to compress the data. Multimedia coding methodologies have accordingly approached the problem by regularizing or conditioning the data to make it well behaved in light of the selected coding technique. Most of the techniques used to achieve high compression do so at the expense of information-management interests by obfuscating the salient perceptual and structural information in the underlying data.

Table 2

	Shading	Shape	Representation	Colour Format	Interactive	Structure
Video	Preserved	Ignored	Complex - Raster	Separate Component	No	Obfuscated
Graphics	Approximate	Preserved	Metafile-Primitive	Composite	Yes	Explicit

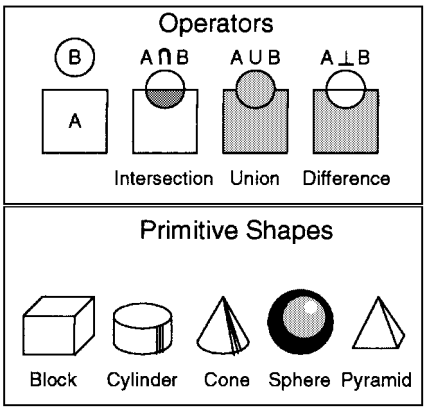


Fig. 5. Constructive solid geometry.

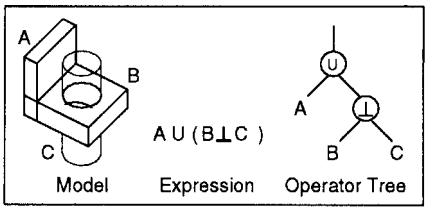


Fig. 6. CSG surface generation.

Table 3

Data Model	Intraframe Coding Schemes
Pixels	PCM / Statistical / Predictive
Vectors	Simple Run length Encoding, WBS
Polygons	2D RLE / Quadtrees / Polygonisation
Smooth shaded regions	Contour-Texture / V.Q. / Transform
3D Objects	Model Based Coding

Typical examples are the standard JPEG and MPEG coding techniques, which uniformly segment images into small blocks that are transformed into the frequency domain. The coefficients in each block are then reordered according to a "zig-zag" pattern and run-length encoded using variable-length codes. The mapping from the initial spatial domain into this final representation is extremely complex. This encoding process occludes most of the spatial information that is present in the image, which is then unavailable for interactive manipulation or information management in any form.

In image coding, the intensity or shading information is of the utmost importance and accurately encoded. Little attention is given to the spatial information, and most techniques typically segment images into small uniform blocks without consideration of the underlying image data. Also, elements in encoded images typically are neither randomly accessible nor individually decodable. Conversely, the encoding of computer graphics has pursued the objective of supporting interactive manipulation. Hence, graphic images are stored as a list of explicit and readily accessible unrendered graphic primitives in metafiles [43]. The shapes of primitives are accurately encoded while the color is only approximated. Each primitive is randomly accessible and individually decodable. Table 2 contrasts these approaches.

B. Computer Graphics

The basic data model in 3-D graphics normally consists of a small set of parametized graphic primitives. Techniques such as constructive solid geometry (CSG) permit the creation of compound objects by merging primitives through the use of set theoretic operators (Fig. 5). This uses a tree-structure representation (Fig. 6) where the nodes contain the operations and the leaves contain the geometric primitives [44]. There is a strong correlation between this representation and the hypertext data model. In CSG, each primitive corresponds to a node while each operator corresponds to a typed hyperlink. The operator tree encapsulates syntactical information regarding the composition of an object. The overall semantic interpretation of the object is a function of the semantics of each component primitive in conjunction with the syntactic information of the operator tree.

Using this basic model, complex objects can be generated through deformations of simple primitives or by using more complex primitives, such as superquadrics [45]. Alternatively, as in the case of quadtrees [46] and octrees [47], a single primitive may be used to tessellate a complex data space hierarchically. Conversely, instead using solid primitives, boundary models rely on two-dimensional (2-D) primitives to model a 3-D object. These representations define a 3-D wire frame or polygon mesh as a list of its composite flat-shaded 2-D polygons or smooth-shaded surface patches in 3-D space.

C. Image-Coding Models

The classical approach to image coding has been to model the statistical distribution of the interpixel luminance variations across the 2-D image plane. It is possible to classify the existing coding schemes according to the data models suggested by the spatial relationships of the interpixel variations. This approach presents five common data models that are consistent with the evolution of image coding first identified in [126]. These models consist of elements that are given in Table 3.

The earliest image-coding techniques, like pulse-code modulation (PCM), attempt to encode the data as an ordered set of statistically independent pixels. Schemes falling into this category include predictive [48] and statistical or entropy-based encoding, which normally encode each element as an independent symbol. This model cannot convey any significant information about the data since the

Table 4

Dimensionality	Temporal Domain Primitives	Interframe Coding Model
0 D	Still Images	None
1 D	Stationary Change	Conditional Replenishment
2 D	Planar Motion	Motion Compensation
2½ D	Layered Motion	Object-Background Schemes
3 D	Full 3D Motion	3D Model Based Coding

granularity of the elements is too small and the relationships between elements are fixed to the raster scan order.

Vector-based schemes such as run-length encoding (RLE) and white-block skipping exploit correlation between adjacent pixels in one dimension. These model the data as a sequence of fixed color, horizontal, variable-length vectors [49]. One advantage of RLE is that some information is directly accessible in the compressed representation from the distribution of run lengths. Since the granularity is not as fine, the elements may convey some limited information. The orientation and order of the vectors is fixed, however, limiting the possible structural information conveyed.

Polygon-based schemes model images as a set of regions where the pixel values are stationary in two dimensions. Examples of this model include tree-based schemes, 2-D RLE, and polygonization techniques. The 2-D RLE schemes [50] typically produce huge numbers of minute irregular regions for continuous tone images. Polygonization schemes [51], [52], which attempt to fit large, simple flat-shaded polygons to the image data, must introduce substantial loss to form the polygons. Alternatively, tree-structured representations [53] like quad and binary trees [54] hierarchically decompose images into many flat-shaded rectangular regions with both size and location constraints. While the granularity of this model is better, the stringent regularizing constraints needed obscure any inherent important information in the image.

Most schemes model the data as an array of nonoverlapping surface patches, regions where the pixel values vary smoothly in two dimensions. This model permits a significantly higher level of information to be encapsulated by each element, such as perspective or depth information from the surface shading. Examples of this model include contour-texture coding, vector quantization, fractals, and transform-based coding. These are distinguished by the representation used to describe the surface-intensity variations. In the simplest case, vector quantization schemes often directly specify the pixel values for each surface [55]. Transform coding [56] represents surfaces as weighted sums of transform coefficients. Contour-texture coding typically represents the surface intensity as a low-degree 2-D polynomial approximation [57]. Fractal coding schemes [58] represent images as a set of surfaces, each defined as a 3-D contractive affine transformation of a given attractor. Fractal schemes are related to grammatical image models that interpret a regular language as an image [59].

Most of the surface-based schemes uniformly segment each image into small blocks to best exploit local stationarity. Even the methods that explicitly attempt to preserve the shapes of natural image regions, such as contour-texture coding, heavily regularize the data. This is due to the simplistic descriptions typically used for the region shapes [60], using either small rectangles [61] or very low order polynomial approximations. The severe constrains on the shape and locality of each surface and the complex representations typical of implementations of this data model limit their usefulness.

Ultimately, it is possible to model the image data as a 3-D environment. Model or analysis-synthesis-based coding [62], [63] relies on updating a predefined 3-D geometric model of an image. These schemes assume a lot of knowledge *a priori* regarding the scene, relying heavily on object recognition [64], [65]. The types of scenes they can cope with are accordingly restricted. One result of this dependence on recognition is that much semantic and structural knowledge is encapsulated by the representations. Identifying an object as a face or the relationship between the eyebrows and the eyes conveys much meaning. These techniques cannot handle arbitrary environments containing unknown or deformable objects, although some work is addressing this problem [66], [67].

It should be noted that in the cases of JPEG and MPEG, the final data representation is a compound generated by successively applying various encoding methods. A different data model is used for each stage. First, an image is modeled as an array of smoothly varying blocks. After transformation and quantization, the data within each block is modeled as a sequence of vectors and accordingly run-length encoded. Last, the resulting data is modeled and encoded as a set of statistically independent variables.

D. Video Interframe Coding Models

Interframe coding schemes are distinguished by the manner in which they attempt to model data changes between consecutive frames. Typically, interframe differences are all assumed to have been generated by some form of motion. This motion can be described according to its dimensionality [126], as in Table 4. At the lowest level, no motion in any dimension yields still images.

The stationary-change model includes simple predictive [68] and conditional replenishment techniques [69], [70]. It assumes the absence of any image flow so that any changes are due only to an "in-place" change of pixel values. The motion is purely orthogonal to the image plane, which is the color domain. This model can only indicate that change has occurred and its location. It is not very robust, failing in the presence of global image motion or even just a large amount of object motion.

Planar motion models assume that motion is purely translatory and confined to a single plane. These seg-

Table 5

Data Models	Coding Scheme
Amplitude Variations	PCM, Temporal Predictive
Spectral lines / Frequency Bands	ATC, DCT, spectral VQ
Frequency tracks	Sinusoidal Transform
Harmonic groups	DHC, Vocoders

ments frame spatially into two parts, the unchanged background and the displaced regions, providing motion vectors for each displaced region. Effective structural information can be conveyed by the relationships between the motion vectors. Uniform patterns may be indicative of certain forms of global motion, while nonconforming vectors indicate the presence of independent object motion. Motion-compensation schemes [71] fall into this category and operate either at a pixel level, such as the differential methods and pel-recursive schemes [72], or at a block level. Second-order geometric or affine transformations [73] may be used additionally to model rotation, skew, and zooming as well as to compensate for global motion [74].

Layered (2 1/2-D) schemes model planar motion assumed to occur in multiple coexistent parallel planes and consist of a background image and an ordered set of planar objects undergoing motion. These schemes are used to implement background-preserving prediction algorithms, which eliminate the need to retransmit background segments when they are revealed after having been occluded. Implementations of this model may operate at either the pixel [75] or block level with simple translational motion, although perspective and affine transformations may also be supported with small image regions [76]. This coding model directly provides information regarding motion velocity, depth order, and motion continuity in the event of object collisions.

Three-dimensional motion models are expressed in analysis-synthesis-based coding techniques [77], which rely on object recognition and mainly perform tracking tasks. They accordingly require a predefined geometric scene structure. This *a priori* knowledge about the scene permits high-level interpretation of the scene motion. A wide variety of techniques have been used to perform this type of analysis [78]. Alternatively, modeling and parametizing the unconstrained 3-D motion that is occurring in an unknown scene [79] is a difficult task.

E. Audio Coding Models

Traditionally, audio coding has been based on either time- or frequency-domain representations. Many of the coding techniques can be applied to either domain. Audio signals can be defined by their frequency, intensity, and time, and most coding schemes can be classified according to the degrees of freedom that the individual elements of their data models have in this 3-D framework (Table 5). Time-domain representations can be considered to be based on collapsing the frequency dimension into a single channel. A few 4-D representations also exist, which use the periodicity of the signal in terms of its frequency decomposition as the other dimension, such as correlograms and wefts [80].

The simplest schemes model the audio signal as a sequence of unit-length amplitude samples. The frequency composition of the signal is disregarded. Coding schemes in this category include PCM, differential PCM, and temporal-domain vector quantization, which represents the data as discrete segments of waveform samples. This model cannot provide much significant information about the audio data.

Next are schemes that model the signal as a set of spectral lines or frequency bands, which are permitted to vary in amplitude. Again, each element is of unit length but this time localized in frequency. This permits each model element to convey information regarding its pitch and intensity. Typical examples include subband coding [81], which encodes the signal as a relatively small number of independent frequency bands, and adaptive transform coding (ATC) [82], which generates spectrogram-like representations with homogeneously treated high-resolution frequency bands.

A signal may also be modeled as a set of sinusoidal frequency tracks. These may vary across both amplitude and frequency in time. Each track additionally conveys information about the time evolution of pitch contours and the presence of frequency modulation. The temporal and frequency relationships between these tracks also provide cues for stream segregation [83]. An example of this model is sinusoidal transform coding [84], which encodes a signal as a polynomial description of the amplitude and phase evolution of the frequency tracks to be reconstructed in each frame of an STFT.

A harmonic group is defined as a set of simultaneous frequency tracks having similar time evolution but being displaced in frequency. Direct harmonic coding (DHC) [85] is an example of this class attempting to identify harmonics in a signal based on its STFT and pitch estimation. A special case of this model is vocoding, which models speech as a set of formants together with other voicing parameters. A formant is defined as a set of adjacent frequency tracks forming specific frequency distributions that remain approximately constant over time. The most common form of vocoder is the LPC, which extracts the formants directly from the predictor coefficients that represent an optimal estimate to a spectrum for a given number of poles.

F. Coding Models for Hypermedia

The sole objective in coding and representation schemes for multimedia data has been compression for bandwidth reduction. This has been pursued without consideration of information management issues, resulting in unstructured stream-based data representations. Accordingly, the compressed data can only be accessed sequentially, and interactive manipulation is impossible. While existing coding schemes make use of a variety of data models, their convoluted representations and regularizations obfuscate the structural information of the underlying data, making them unsuitable for hypermedia. Hence, nonlinear access can only be supported after extensive processing to generate

separate index files. While these may permit some random access, the data are still not interactively manipulable.

Only the model-based (recognition) coding schemes seem to provide suitable support for hypermedia. The requirement of precisely knowing what an object is prior to being able to access or interact with it, however, is an unnatural imposition. It is the process of interaction (if only in the form of exposure) with an unknown object that leads to its classification (at a late stage in cognition) within a semantic network in the mind based on the nature and outcome of the interaction.

Accordingly, semantic methods are inappropriate as generic techniques for generating structured data representations. Instead of attempting to recognize specific objects or first understand the data semantically, the problem should be approached through abstraction. In this case, a subsumption-style architecture [86] is more appropriate. The architecture consists of simple layers, each building on and utilizing the functionality of the preceding layer to perform increasingly more complex tasks. This alleviates the lower levels from being overburdened with knowledge that is irrelevant to their function. Instead of a system that can recognize and identify a limited number of specific objects, a system is required that can identify the presence of objects and their characteristics without necessarily recognizing what they are. Object recognition can be delegated to some later stage of processing if it is so desired. Rather than semantic information, syntactical or structural information should be exploited as the basis for these coding models, as in cognition.

V. Cognition, Semiotics, and Perceptual Psychology

This section surveys pertinent cognitive, psychological, and semiotic issues for multimedia data models. Section V-A reviews general cognitive principles and data models. Section V-B examines the mental representations in the early perceptual processes. Section V-C surveys semiotics and its relation to hypermedia systems. Section V-D reviews semiotic articulation in multimedia data streams. Section V-E discusses Gestalt theory in relation to the creation of structured data representations.

A. Cognitive Data Models

Since hypermedia is meant to imitate the cognitive process, it would not be inappropriate to base any hypermedia data models on the mental representations that underlie cognition. However, cognition is a complex process composed of different tasks proceeding in various stages concurrently, for each of which a new representation is used. While our understanding of cognition is still very primitive, some basic principles may be exploited in the formulation of suitable data models for hypermedia.

The study of eye movements during reading [87] reveals much about the early cognitive processes. In essence, word recognition relies on a feature-analytic approach operating at three levels with feature-, letter-, and word-specific

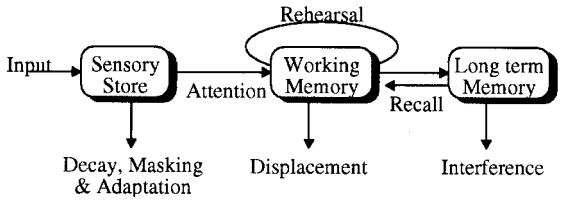


Fig. 7. Memory model.

detectors. From a formal language theory perspective, the lexical word-formation process is followed by sentence formation through syntactical analysis. The semantics are then evaluated and the meaning is integrated with past experience in the mind through pragmatic processes. The contextual theory of meaning specifies that the meaning of a symbol is a syntactic function of its relation to other symbols. Thus, the reading process utilizes at least five distinct representations composed of features, letters, words, sentences, and semantic structures. This description is somewhat simplistic, for in reality, there are various feed-forward and feedback systems that mediate in the processes and influence them based on contextual factors and expectations. A similar process occurs when speaking with a new representation formed in a different region of memory as it proceeds from semantic through syntactic, morphologic, and phonological systems.

According to the memory-spatial metaphor often used to help explain this principle, memories are treated as objects stored in specific locations in the mind. The common multistore memory model (Fig. 7) specifies three main types of memory, each with very different data representations. These are 1) a predominantly feature-analytic, modality-specific, brief sensory store, 2) the working memory [88], which seems to contain about seven pointers [89] to previously stored memories, much like address registers in a computer [90], and 3) the long-term store with unlimited capacity. According to a long tradition arguing that all knowledge is in the form of associations [91], the long-term memory stores knowledge in the form of either an associative, semantic, or declarative network.

Semantic networks consist of nodes, each representing a single concept, connected by links of various types and activation strengths. The constitution of the nodes may be explained in part by the attribute theory of concepts, which states that semantics are captured by conjunctive lists of attributes. These attributes may be one of two types: defining or characteristic. This theory also specifies that the concepts themselves are hierarchically organized, probably through link-based inferences. Coding theory attempts to describe the analogical or propositional representations of concepts as syntactically based primitive codes in the mind composed of imagens or logogens [92]. Kosslyn [93] proposed a computational model of imagery stating that in long-term memory, analogical information is stored about the spatial representation of images and is linked to propositional information about the parts of visual objects and how these are related to each other.

Essentially, cognition revolves around the formation and manipulation of a hierarchical network of mental representations. At the bottom are the simple features detected by the early perceptual processes, which are somehow transformed into meaningful conceptual units at the top. How the semantic understanding actually takes place probably can best be understood in the context of the fundamental principle of understanding, which states that to understand something is either to understand it in terms of something else (a recognition task) or to get used to it [94].

In the first case, understanding is externally relative since it concerns correspondence between two domains: a previously understood semantic domain and the new one providing only syntactic information. While the syntactic domain is understood in terms of the semantic domain, at some previous time the semantic domain must also have been understood in this way in terms of another, so that understanding is recursive in this manner. This is known as the correspondence continuum [95], which affirms that an element may be either syntactic or semantic depending on the point of view. This dual role of cognitive objects may be partly appreciated through the overlap between syntax and semantics, since both are concerned with the relations that exist among symbols.

In the second case, understanding can only be internally relative and therefore can only concern syntax. In absence of external relations, semantic understanding is reduced to syntactic understanding. Without any correspondences with which to define the meaning of any given symbols, they must be understood in terms of themselves. Therefore, the syntactic domain becomes its own semantic domain. This base case, the last semantic domain in a correspondence continuum, can only be understood syntactically. The cognitive process of transforming sensory data to perceptual features and finally into a semantic network representation is reduced to a purely syntactical process in this instance. Given that these transformations rely on syntactical processes, two questions remain: What are the syntactical units at each level and what is the nature of these syntactical transformations?

B. Perceptual Data Models

Some insight into the modality-specific representations and transformations found in the sensory store is provided by psychophysical evidence. While containing many both inhibitory and excitory feedback and feed-forward paths, the neurological organization tends to be predominantly hierarchical. This structure consists of increasingly more complex receptive fields in succeeding levels, forming specifically tuned pathways. The receptive fields at each level are composed of simple configurations of its subordinate elements and detect increasingly more abstract features.

In the case of vision, we know that while the spatial layout is preserved, the representation generated by the retina is heavily distorted due to the physical limitations of the eye and the properties of the retina [96], [97]. A number of processes also specifically enhance the visibility

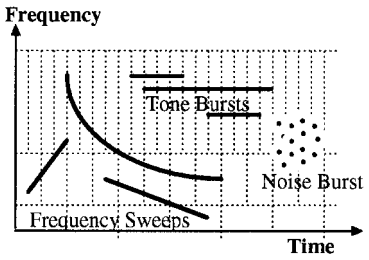


Fig. 8. Mental audio representation.

of perceptually important features such as luminance edges [98], [99]. Color is encoded according to an opponent color model [100] providing lower spatio-temporal resolution to the chromatic detail. In fact, the perception of color is often extrapolated from luminance edges via a filling-in mechanism [101].

It has been shown that there are two main pathways operating in parallel in the visual cortex. It is known that structure and motion are processed separately from color, form, and texture [102]. The color pathway mainly performs recognition tasks, while the other is dedicated to structure and motion analysis. This distinction is interesting from a cognitive viewpoint because it implies that structural understanding is to a certain extent separate from recognition. This suggests that semantic understanding is intrinsically related to, yet separate from, syntactic understanding in the mind.

The first data representation in the visual cortex [103] is defined by the incipient neurones, which have center-surround, circularly symmetric receptive fields. These feed into "simple" cells, which respond to specifically oriented line segments. Next, temporally modulated, specifically oriented line segments are detected by "complex" cells. Corners and ends of line segments are next detected by orientation-specific "hypercomplex" or end-stopped cells [104]. Each higher level is less dependent on spatial localization, and cells that respond to hand images and faces have even been found.

The initial data representation in audition is a tonotopically organized frequency decomposition of the acoustic signal performed by the basilar membrane in the cochlea [105]. Signal masking arising in the cochlea has the effect of accentuating dominant frequencies. The frequency separation is logarithmic due to the placement of the innervating nerve fibers. Below about 4–5 kHz, they also encode timing information of the stimulus waveform [106]. Beyond the cochlea, temporal and intensity information are separately processed in two parallel pathways [107]. At these higher levels in the cortex, neurones detect three main types of features: tone bursts, noise bursts, and frequency- or intensity-modulated components [108] (Fig. 8). Some neurones detect specific frequency or intensity modulation rates while others respond to the direction or speed of frequency sweeps. Others detect repetition rates or the onsets or offsets of stimuli, or are stimuli-duration selective.

The characteristic of isolating dominant frequencies together with directional sweep and modulation detectors in-

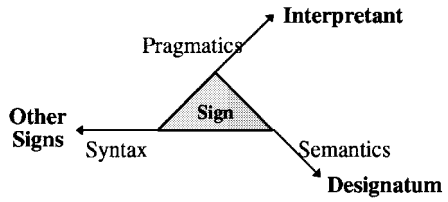


Fig. 9. Dimensions of semiosis.

dicates that some type of frequency and amplitude tracking is performed by the auditory system. One could hypothesize the existence of a mental auditory representation composed of tracks in frequency-time-intensity space. In reality, there are many interconnected representations in the cortex but we have very little information about what these are. It is known that the higher level representations are based on the lower level features but little is known about how the latter are combined into higher level representations. There is, however, clear evidence that the grouping of these primitive features underlies the phenomenon of stream segregation [109].

There are a number of factors that influence the gradual segregation of auditory stimuli into acoustic objects. Each factor competes for supremacy in determining groupings, forming various segregation propositions that are evaluated in parallel, of which the most probable is chosen. Some of these factors include the synchrony and harmonicity of the frequency partials, correlations in frequency or intensity modulation, suggestive signal transitions, the presentation rate of the stimuli, and the intensity of the partials, with more intense higher frequency partials tending to segregation.

C. Signs and Semiosis

Semiosis is the process of making and using signs to effect communication and understanding. Semiotics [110] is the study of communication and understanding. It is concerned with the relationships of meaning of the signs. Apart from the classical verbal and lexical communication processes to which it is applied, semiotics is pertinent to a wider range of interactive information processing. Morris [111] describes semiotics in the context of three basic phases of interaction. First is the perceptual stage, which is based on seeking signs or objects. Second is the manipulatory stage, which is gaining control of the signs. Third is the consumatory stage, which lets the signs perform their function. There are also three corresponding types of inquiry that can be performed and three different relationships that can be held with the signs. In the perceptual stage, signs are primarily designative in that they signify what to expect from them. In the manipulatory stage, signs are prescriptive because they signify appropriate courses of action. In the consumatory stage, signs are appraisive because they reveal how well they respond to the desired manipulation.

Fig. 9 shows the three dimensions of semiosis: signs, designatum (what a sign stands for), and the interpretant (or user). These three dimensions have correlates in semiotics, which are syntax, semantics, and pragmatics. The syntactical dimension of semiotics defines the formal relationships between individual signs and how these may be combined to form compound signs. Semantics defines the meaning of the signs themselves through the relationship between each sign and its designatum. Pragmatics is the integration of the meaning with the interpretant's past experience. It defines the relationships between signs and their interpreters and is based on the origin, uses, and effects of the signs. The domain of semiotics also embraces the classical engineering realm of information theory. In semiotic terms, Shannon's information theory deals with efficient sign vehicle transmission where a sign vehicle is a sign independent of its significance.

Semiotics traditionally has been applied to the external representational systems used for explicit communication. These representations correspond to lower levels in the cognitive-communicative process that are predominated by syntactical considerations. A distinguishing feature of human communication is the fundamental principle of double articulation, which specifies a two-level structure for communication [112]. Classical semiotics accordingly has focused on the analysis of signs (monemes) and their composition in areas such as text and speech. The signs (which are defined as the smallest units of meaning) are constituted by subsigns, which are meaningless but distinctive units whose only function is to distinguish the monemes. Typically, semiotics has involved the study of words (which are primarily syntactic units) as monemes and their composition.

It is also possible to extend the semiotic model to higher level knowledge representations. Metasigns are formed by grouping signs in the same manner as subsigns are grouped to form signs. These metasigns may be considered to be true semantic units, given that semantics arise within appropriate groupings of syntactic units. A group of these metasigns can be considered to define the graph of a semantic network, with each metasign corresponding to a node instance and the designatum being the conceptual unit represented by each sign. The syntactical domain specifies the links defining the relations to other nodes. In this manner, a hierarchical semiotic structure may be defined where the higher level signs may be recursively decomposed into their component subsigns.

Hypertext specifically attempts to model the data as a network of semantic or conceptual units (Fig. 10). Hypertext nodes are more appropriately called hypersigns, conveying potent semantics and typically consisting of a number of metasigns. At this level, the (hyper)signs become more amorphous and the focus is on the relationships between the concepts they designate. In hypertext, the sign vehicles are anchor keywords or phrases that are directly linked to their designatum, the nodes. This reduces the role of the interpretant since both the sign vehicles and designatum are concurrently present in the media. The syntax or relationship between the signs in hypertext is defined by other sign vehicles embedded within each node. The semiotic analysis of each hypertext node in isolation

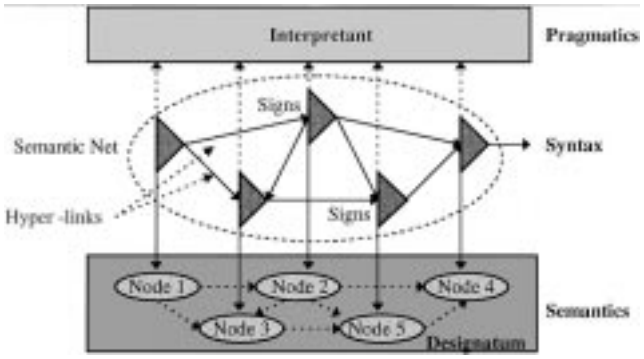


Fig. 10. Semiotic dimension of hypertext.

may additionally be performed in terms of its component signs and subsigns.

The semiotics of hypertext systems can be readily analyzed since the articulation of text and speech is quite evident, but this is not the case for other modalities. Determining the articulation in multimedia data is not straightforward since there may be little correspondence between the physical manifestation of the data and mental representation of the sign vehicles. For example, phonetic representations are quite different from the time-domain speech signals. Generating suitable data models for hypermedia requires the identification of the semiotic articulation in multimedia data. This requires identifying the subsigns in each media and how these may be combined to form signs.

D. Semiotic Articulation

While classical semiotics is based on the theory of double articulation, each different communication system has its own domain-specific set of articulatory units. A good example of a well-developed theory of double articulation is linguistics. Phonemes, which are meaningless sounds, are the subsigns that can be combined to form monemes (or morphemes) equating roughly to syllables.

In the textual domain, the subsigns are known as graphemes and correspond to alphabetical letters in English. This is an example of cenemic writing, where the graphemes represent phonetic elements such as phonemes or syllables (Table 6). Conversely, in pleremic writing systems, the graphemes refer to semantic units such as pictographs. Accordingly, the semiotics of writing is viewed as either an autonomous or heteronomous system. Depending on which view is accepted, graphemes are either signs or subsigns. In the autonomous view, the monemes equate to words, while in the heteronomous view, graphemes are already signs so that when grouped they become metasigns.

Articulation is also evident in music, although it is more abstract than other forms of communication. This is because there is no clear separation of form or expression from the content in music since the expression is the content. Instead of information, it mainly communicates emotion since music is the logical expression of feelings. Music also has a powerful referential potential, which assigns meaning through association to past experiences. Music has

Table 6

Textual Element	Letters	Words	Sentences	Paragraphs
Type of Unit	Lexical	Syntactic	Semantic	Conceptual
Designation	Subsign	Sign	Metasign	Hypersign

a highly evolved structure with definite rules much like normal grammars, resembling the hierarchical organization of text. While music can be physically expressed in terms of frequency, time, and intensity, in musicological terms, music has three dimensions: melody, harmony, and rhythm. Melody is the progression of tones produced by adding them horizontally, while adding tones vertically generates chords and adding chords sequentially produces harmony. Rhythm is produced by periodic repetition. The smallest subsign of music is therefore a tone or toneme. While a single tone has no embodied meaning, a short series of tones can readily convey an emotional experience [113]. If the series is ascending, it expresses outgoing emotion; if descending, it expresses incoming emotion. If it is in a major key, joy is conveyed; alternatively, sorrow [114]. The logical theory of semiotics in music [115] postulates that at least three notes are required to form monemes.

Articulation also exists in pictures. Various proposals for the articulatory units include the concept of *chromemes* (color elements) and *formemes* (shape elements), among others [116], [117]. However, it is difficult to foresee any lexical constructs to combine *chromemes* to produce meaningful units. Another approach that has been proposed is Marr's model [118], which postulates the existence of three different representation systems starting with an initial 2-D primal sketch and progressing to a viewer-centric 2 1/2-D sketch and finally to an object-centric 3-D representation for semantic recognition. Apart from the vague notion of *texturemes*, which are difficult to isolate and are not distinctive, Marr's model does not really provide primitive elements that could be considered suitable articulatory units. A better approach is based upon Gestalt psychology, although a suitable definition of subsigns or primitive elements in pictures is currently lacking.

Video or film communication is a composite medium of a sequence of images undergoing motion. Since the articulation within each frame is the same as for still images, the primary feature of video and film is its temporal domain. Accordingly, the first level of articulation is the shot (or scene) and is known as the *videme* [119], [120]. Some uncertainty has been expressed regarding the existence of a second level of articulation. One proposal is that it is composed of spatial-graphical objects called *cinemes* (or *iconemes*). These already represent meaningful elements, however, and are therefore unsuitable. An alternative would be to exploit the predominantly temporal nature of video and the observation that interframe changes are largely motion induced. Accordingly, it would not be inappropriate to propose that the second-level units be composed of a set of motion primitives. Eco [121] considers motion primitives the dynamic units of a third level of articulation called *cinemorphs*.

Table 7

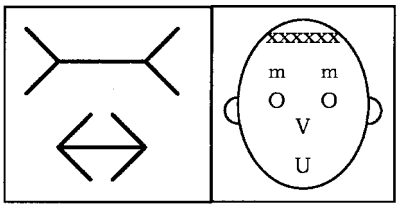
Level	Text	Speech	Music	Images	Video
Sign	Words	Monemes	Motifs	Objects	Videmes
Subsign	Graphemes	Phonemes	Tonemes	Graphic Primitives	Motion Primitives

The double articulation in classical semiotics implies the existence of only a single level of cognitive units, each capable of equivalent semantic value, and a single level of precognitive detection units. In reality, there is a continuum of cognitive units having increasingly higher semantic significance. For example, phonemes in speech can be described in terms of formants, voicing, and manner of articulation. Suprasegmental phonemes in speech (secondary phonemes or *prosodemes*) include the pitch and melody of speech, which in some languages, like Chinese, are essential in determining meaning. Graphemes in text can also be defined in terms of simpler primitives consisting of oriented straight lines, intersections, and closed or open curves in certain configurations. Fortunately, from psychophysics, we know the general nature of these most primitive elements for multimedia data. We also have a fair idea as to the nature of the signs and metasigns (Table 7). The task remaining to formulate perceptually concurring data models is to determine the specific nature of these primitive elements and how they combine together at each level to form signs and metasigns.

Understanding how these elements can be grouped to form single conceptual units is a significant difficulty since this typically requires semantic knowledge. At higher levels, the rules defining how elements may be combined are also increasingly more complex. The task of assigning a semantic to a given metasign is a recognition process heavily dependent on pragmatics. More than just a clustering problem, the question of how the meaning of each individual sign is modified by the grouping and determination of the overall meaning of the metasign is perplexing. However, divorced from semantic and pragmatic issues, the individual low-level signs, being syntactic units, are relatively easy to identify in a given communication medium. Gestalt psychology attempts to offer some insights into the question of how syntactic units are grouped together to form semantically significant units.

E. Gestalt Psychology

Gestalt theory is perhaps one of the best established yet poorly defined theories of perception [122]. This theory postulates that perception is based on sets of stimuli where the whole has a meaning or significance that is not predictable from its elements. These semantic groups are known as Gestalten. For example, a square is semantically more significant than a group of lines and a tune is more than the sum of its notes. This leads to some interesting questions, namely, what is the nature of the groupings that have increased significance over other arbitrary groupings? This question has two implicit components. First, what bearing does the relationship between the elements have on the significance of the whole? Second, are there any specific characteristics required on the part of an element for it to contribute to the forming of a more significant whole?



(a) (b)

Fig. 11. (a) Contextual effects. (b) Contextual influence.

When a whole is greater than the sum of its parts, it creates a recursive relationship in that the meaning of the whole must then influence the meaning of each part. Context clearly influences perception. In fact, we seem to respond more to relationships among stimuli than to the specific characteristics of the individual stimuli. This explains why it is possible to replace original parts of a stimulus with other parts and still manage to retain the quality of the whole. Examples of this are commonly found in musical transposition and in the phenomenon of brightness constancy. The importance of contextual influences in perception is readily exemplified in Fig. 11(a), where horizontal lines of equal length appear to be disparate. In this case, the diagonal lines impart depth cues that affect the interpretation of the line length.

The role of contextual influences in perception extends beyond simultaneous context to historical influences such as familiarity and expectations. There is the tendency to classify stimulatory events according to past experience. In this case, categorical event perception takes precedence over sensory perception and may override it. In Fig. 11(b), the letters O, V, U, m, and x are interpreted as facial features and not as letters. A powerful example of contextual effects is the filling-in mechanism, which automatically interpolates stimuli to preserve the perception of continuity even when the stimulus itself is discontinuous. This phenomena can be found in audio perception, where a gap due to signal dropout in a tone or in a frequency sweep can be masked by presenting narrow-band noise in synchrony with the onset and offset of the gap. In the visual domain, the filling-in mechanism is more powerful and can completely eliminate certain image contours and create the perception of surfaces that do not exist. In the Kanizsa diagrams [123], white polygons are clearly perceived through visual interpolation even though they do not exist (Fig. 12).

Apart from complex contextual influences that affect the formation of perceptual groupings, another difficulty in defining grouping rules for perceptual organization is that the mind is constantly searching for alternate organizations. Also, groupings may be difficult to define in complex

Table 8

Semiotics	Hypertext	Cognition	Graphics (CSG)
Semantics / Designatum	Node Layer	Defining Attributes	Geometric Primitives
Pragmatics / Interpretant	Presentation	Characteristic Attributes	Shape Parameters
Syntax / Relationships	Link Layer	Associations	Set Theoretic Operators

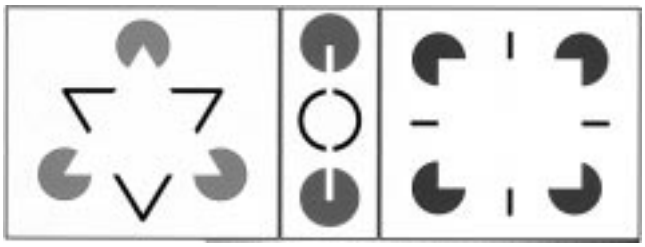


Fig. 12. Visual interpolation.

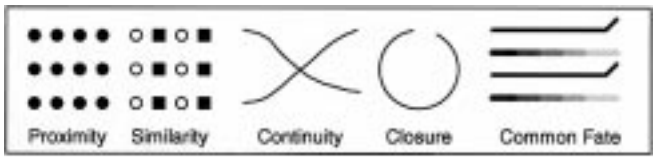


Fig. 13. Grouping rules.

patterns, and a single component may only be assigned to a single group. In spite of this, various principles have been proposed [124] that are based on the proximity, similarity, continuity, common fate, and closure of the stimuli (Fig. 13). The common-fate principle is based on correlations in the form of synchronization or frequency or intensity modulation. Closure implies the continued perception of obscured stimuli via an interpolative or filling-in mechanism.

While general grouping principles have been suggested, Gestalt theory has difficulty in specifying the definition of the primitive elements themselves. This is because contextual factors and relationships interfere with the interpretation and definition of the elements. However, from Section V-B, we know that the elements must be hierarchically defined. Also, since the relationships among perceptual stimuli are more important than the absolute values of the stimuli, it appears appropriate to propose a hierarchy of primitive elements based on clearly defined relationships between the elements.

VI. Hypermedia Data Models: A New Semiotic Paradigm

This section presents the new semiotic paradigm for hypermedia data modeling. Section VI-A discusses the characteristics desired for hypermedia models and introduces the new paradigm. Section VI-B presents new syntactic data models for multimedia based on a semiotic articulation. Section VI-C discusses the information-management support provided by the data models. Section VI-D presents rudimentary compressed representations for audio and video data based on the models.

A. Semiotic Paradigm

As a communication system operating at an advanced cognitive level, the semiotics of hypermedia is complex. Semiotics is established on the fact that all communication is based on the generation and perception of signs. All but the most primitive communication is highly structured, and semiotics attempts to determine the nature of these structures and their constituent elements. For hypermedia systems to be effective, the data representations should be based on data models, which permit direct access to the semiotic structures in the data. The problem is how to identify and isolate these structures. Initially, we may begin by abstracting the three dimensions of semiotics (semantics, pragmatics, and syntax) into the type definition, expression or form, and relations. The basic type definition of an object epitomizes the core semantic value and is its principal attribute. The form of rendition or expression of an object provides additional interpretational cues suggesting specific semantic detail similar to prosodics in speech. This encapsulates an object's characteristic attributes and may be conveyed through parameterization. The relations define the possible interactions between the objects or signs. We can use this abstracted framework to compare the general equivalence of diverse information-processing systems, as demonstrated in Table 8.

Additionally, we can identify a number of general principles from cognition and Gestalt psychology (Fig. 14) that govern semiotic structures in multimedia data. Essentially, they are semihierarchical, multilayer network structures. Each element or node in this structure is separately described in terms of both its defining and characteristic attributes, and the relationships between elements are explicitly defined. Furthermore, to permit intramedia nodes to be fully linkable as a source or destination and to support both information management and interactive manipulation of the data, each element must also be indexible, individually decodable, and randomly accessible.

Another consideration is that the basis of this organization should not be semantic but syntactical to permit environment-independent automatic processing. The representation should not obfuscate any inherent structural and perceptually important information in the data. Also, in conformity with the concept of subsumption architectures for information processing, the representation should be midlevel and generic rather than distinct for each specific application. This would permit the representation, given suitable supplementation, to be used for a variety of different applications. It should make all information explicitly available to higher level processes such as content-based retrieval, structured browsing, editing, recognition, and understanding without attempting to interpret the information in any way. Searching in such a compact data space

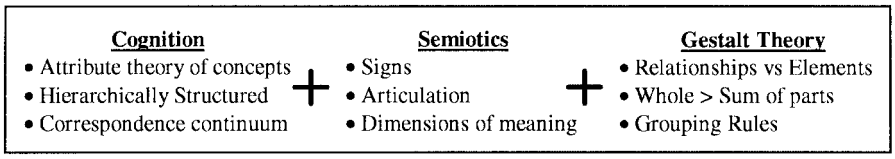


Fig. 14. Principal properties.

Table 9

Hypermedia Properties	Existing Representations	Required Representations
Structured Representation	No - bit steam	Multilayer Network Structure
Indexible Components	At best Frame based	Per Object/Node Indexing
Random Access	No	Content Based
Decoding Granularity	Often entire stream	Individually Decodable Nodes
Internode Relationships	None or serial only, concealed	Unconstrained and Explicit
Link Support / Navigation	Only as a destination	Per node source and destination
Content Based Access / Retrieval	Manually Annotated Indexes	Intrinsic Support
Structured Browsing	No	Hierarchical, intrinsic
Restructureable	No	Add, Move, and Delete Nodes
Manipulable Representation	Virtually full decoding required	Fully Compressed Editing
Compressed Data	Yes	Same amount as existing
Interaction	Extremely limited, frame based	Extensive node / content based
Navigational Dependencies	Presentation level integration	Environment independent
Authoring Support	Not reusable, system dependent	Reusable, Portable
Information abstraction / preservation	No / Obfuscated	Yes / Explicit

would only involve supplying similarity-matching algorithms without needing to decompress or further process the data, or to create separate index files. Object recognition and understanding could be performed simply by directly interpreting the structural information that is made explicitly available in the same compressed data. Table 9 contrasts these properties with those provided by existing coding schemes.

A general semiotic framework based on syntactic principles can be formulated to create data models and representation schemes to meet these requirements. Syntactical models can explicitly reveal the structure of the information, permitting efficient, generic, and interactive access to any encapsulated semantic information without permitting the semantic information to encumber the interaction. While not necessarily revealing what the encapsulated semantics are, they do not preclude the inferral of semantic interpretations. The constituent elements in syntactical structures are easy to identify and extract through statistical techniques. They are also capable of conveying powerful semantic information given the right association through the relationships between them. Specific statistical information about each syntactic entity can be obtained by individually accessing each syntactic element. The proposed data model for each medium is accordingly constituted of three specific components: primitive syntactical units, the characteristic attributes or parameterizations for each primitive, and the set of relationships between the units. Corresponding representation schemes would preserve these three information components, making them explicit and directly accessible.

B. Semiotic Data Models

While various ad hoc attempts have been made to define the primitive syntactical units in multimedia data, no systematic approaches have been suggested. It is proposed that these elements should be defined in terms of a multidimensional decomposition of the data space itself, creating a hierarchy of elements with decreasing degrees of freedom as the dimensional constraints increase. This permits the generation of higher level elements from simple linear groupings of those below them that mimic the organization of the early perceptual processes. The dimensionality of an element becomes its defining attribute while any additional parameterizations are the characteristic attributes.

Table 10

Domain	Images - Spatial	Video -Temporal	Audio -Acoustic
MetaSign	Picture	Episode	Episode
Signs	Objects	Scene	Phrase / Motif
SubSigns 1	Surfaces	Shot / Global Motion	Harmonic Group
SubSigns 2	Lines	Object / Local Motion	Pitch Contour / Track
SubSigns 3	Pixels	Stationary Change	Tone Burst

The elements at the higher levels of this hierarchy contain greater semantic power than those at the lower levels due to the Gestalt principle that the whole is greater than the sum of its parts. For example, at the lowest level of an image, a row of picture elements (pixels) forms a line, and an appropriately structured group of four lines forms a square. The square is much more than just four lines; it has an extra quality of "squareness." Each line is more than just a cluster of pixels; it exhibits "lineness." Judicious placement of lines produces simple vector and cartoon-like images that nonetheless can carry very powerful semantics, such as computer-aided design drawings. Rather than focusing on the capacity for semantic expression through Gestalt phenomenon or determining possible grammars at this stage, however, we are interested only in defining the primitive elements. Table 10 identifies these elements for each domain according to their dominant modes.

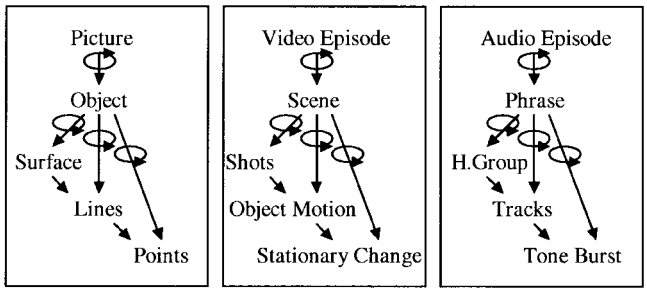


Fig. 15. Semiotic data model.

Each of these primitive elements or subsigns in isolation, a simple polygon, or a motion primitive or tone burst have little (if any) semantic qualities. To constitute signs, it is necessary to group these primitives together appropriately to create semantically significant elements. For example, phonemes are a select grouping of frequency tracks and a face is an appropriate grouping of surface patches. Omitted from this analysis is any definition of the grammars required to generate these signs. Such definitions are beyond the scope of this analysis since there are a plethora of potential grammars, one for each different semantic entity. It should be possible to infer groupings for a particular data stream, however, by identifying commonly occurring configurations, which, while not necessarily constituting semantic units, may be used to enhance coding gains.

Each sign can be decomposed into one or more of the subsigns in its domain (Fig. 15), and each subsign also recursively can be defined in terms of the simpler ones. Rather than representing data by a single type of element, as many coding schemes do, these models encourage the simultaneous use of all the elements for a given domain. This enables the concept of layering where complex data can be defined by the superposition of simpler elements. It is intended that higher level elements define the basic data characteristics and lower ones are used to supply any additional fine detail not adequately conveyed by the more complex elements. While it is possible exclusively to use the lowest level subsigns to represent a given media, it would be making poor use of the model. The main purpose of the lower level elements is to provide a fallback mode to compensate for when the higher level elements fail to model the underlying data accurately. Video coding is an example of this where planar motion compensation alone is insufficient to compensate for interframe changes. In this case, the residue can be appropriately modeled as the result of additional stationary changes. Layering also facilitates progressive refinement of data and the ready discarding of fine detail if needed.

Each primitive element in this generic data model also has a set of parameters associated with it that is specific to the element's domain. These are identified in Table 11. The abstraction provided by this model permits all of the important cognitive information to be explicitly available for later high-level processing. For humans, it is difficult to perceive something without simultaneously interpreting its meaning. This recognition of meaning is highly dependent

Table 11

Defining	Characteristic Attributes		
Pixel	Colour / Intensity	Location	
Line	Path (length, orientation)	Intensity contour	
Surface	Shape (area, orientation)	Shading	Texture
Stationary change	Region Shape / Area	Region Content	Time / Duration
Object Motion	Affine Transformation	Depth Order	Time / Duration
Shot	Camera Motion	Lighting Changes	Time / Duration
Pure Tone	Frequency	Amplitude	Duration
Frequency Track	Pitch Contour (melody)	Intensity Contour	Modulation
Harmonic Group	Frequency spacing	Energy distribution	Periodicity

on the interpretation of the relationships between syntactic elements. Alternatively, in this model no attempt is made to interpret this information in any way, thereby not binding any semantics to the data. This preserves the generality of the information encapsulated by the model by not constraining its ultimate designation. This also permits generalized scene descriptions conveying the core semantics to be generated from the data by simply considering the defining and relational attributes alone. A formal description of the proposed data models for image, audio, and video data is presented in the following definitions.

Definition 1—Image Points: An image is a function of the set of all ordered pairs (x, y) of real numbers $f: \Re^2 \rightarrow \Re$ or $(x, y) \xrightarrow{f} z$. The pair (x, y) is referred to as a point. The set of all points, $C \subset \Re^2$, given by $\{(x, y) | 0 \leq x \leq w, 0 \leq y \leq h\}$, defines the image plane.

Definition 2—Relational Vectors: Given an ordered set of n points $E = \{(e_1, e_2, \cdots, e_n) | e \in C\}$, one can define an ordered set of n vectors $\boldsymbol{V} = \{(v_1, v_2, \cdots, v_n) | v \in \Re^2\}$ specifying spatial relationships between these points. A graphical pattern is then defined as the biproduct $G = E \oplus \boldsymbol{V} = \{(e_1 + v_1, e_2 + v_2, \cdots, e_n + v_n)\}$. If these are time variant $\boldsymbol{V}(t)$, they become relational motion vectors.

Definition 3—Adjacency: The neighborhood of a point $a = (x, y)$ in the image plane can be defined as the set $N(a) = N(x, y) = \{(u, v); |x-u| + |y-v| = 1, |x-u| = |y-v| = 1, (x, y) \in C, (u, v) \in C\}$. The adjacency of two points a and b is denoted by $a * b \Leftrightarrow a \in N(b), b \in N(a)$. Adjacency can also be expressed by the relationship defined by the vectors $\boldsymbol{V} = \{[(u, v), (u+i, v+j)] | u, v \in \Re, i, j \in \{-1, 0, +1\}\}$.

Definition 4—Paths: Joining a string of adjacent points forms a "path." A path Q of length n is defined as the set of n adjacent points $Q = \{(p_1, p_2, p_3, \cdots, p_n) | p_1 * p_2 * p_3 * \cdots * p_n\}$. Each point can be defined parametrically as $p(t) \rightarrow [x(t), y(t)]$ subject to the constraint that $p(t+1) = [x(t) + i, y(t) + j]$ where $i, j \in \{-1, 0, +1\}$. The path definition then becomes $Q = \{p(t) | p(t) \in C, p(t) * p(t+1), 0 \leq t \leq n\}$. A closed path is subject to the additional constraint that $p_n * p_0$.

Property 1: Paths $f(t)$ and $g(s)$ are said to be connected if at least two points (one point from each) are adjacent. This is denoted by the commutative relation $f(t) \blacklozenge g(s) \Leftrightarrow \exists t \in \Re | f(t) \in N[\{g(s) | 0 \leq s \leq m\}]$.

Property 2: Two paths $f(t)$ and $g(s)$ can be said to be adjacent if they are connected and every point in path f is adjacent to a point in path g over the common interval $s \cap t$. We then say $f(t) * g(s) \Leftrightarrow f(t) \blacklozenge g(s), \forall h \in s \cap t, f(h) \in N[\{g(s); 0 \leq s \leq m\}], g(h) \in N[\{f(t); 0 \leq t \leq n\}]$.

Definition 5—Domains: Joining adjacent paths that lie side by side forms a domain $D = \{(q_1, q_2, q_3, \cdots, q_n)|q_1 * q_2 * q_3 * \cdots * q_n\}$. This equally can be expressed as $D = \{q(s)|q(s) \in Q^*, q(s) * q(s+1), 0 \leq s \leq n\}$ where Q^* is the infinite set of all possible paths. Since $p(t) \rightarrow [x(t), y(t)] \in q(s)$, then any point on D can be specified by $p(s, t) \rightarrow [x(s, t), y(s, t)]$. Alternatively, a domain may also be defined as the set of all points lying within a closed path or closed set of connected paths, i.e., $D = \{q(s)|q(s) \in Q^*, q(s)\blacklozenge q(s+1), q(0)\blacklozenge q(n), 0 \leq s \leq n\}$.

1) Image Model: An image can alternatively be defined as being composed of a set of ordered three-tuples $(x, y, i) \supset C$ where i is the color or intensity of the image at location x, y. An image then becomes $I = \{(x, y, i)|0 \leq x \leq w, 0 \leq y \leq h, 0 \leq i \leq 1\}$. The most primitive elements in an image are the pixels or picture elements $P = (x, y, i)$, which have two distinct attributes: their color/intensity i and spatial location.

Adding a number of pixels together in any given direction forms a line that has the additional property of length. Its orientation can be disregarded as being a function of an arbitrary frame of reference. The formal definition of a line L is given by the set of pixels lying on a path Q where $L = \{[p(t), i(t)]|p(t) \in Q, 0 \leq t \leq 1\}$. Both the path and the intensity of the line along the path are defined as simple parametric functions. Two special classes exist: flat-shaded lines where $i(t)$ is constant over Q and the case where $p(t)$ defines a straight line.

Adjacent lines form a surface. A surface S is defined by a set of adjacent lines that contain all of the pixels in a domain D. Formally, $S = \{[p(s, t), i(s, t)]|p(s, t) \in D, 0 \leq s \leq m, 0 \leq t \leq n, s \perp t\}$. The function $i(s, t)$ defining the intensity contour is assumed to produce a smooth surface. A polygon is a special case where $i(s, t)$ remains constant over the surface area. Due to this smoothness constraint, to model a natural image region properly, one may require the addition of a zero mean, stationary noise component. In this case, the surface model becomes $Sn = S + \{[\eta(x, y)]|(x, y) \in D\}$ where $\eta(x, y)$ is the noise function at point (x, y). If the nonzero components in the noise function are separated out, the model becomes a superposition of a set of pixels over the original surface.

An object O is defined as a collection of connected parametrically defined surfaces, lines, and pixels that have been translated to the origin. The spatial relationships between them are given by the set of vectors $\boldsymbol{V} \subset \Re^2$. Thus $O = \boldsymbol{V} \oplus \{(e_1, e_2, e_3, \cdots, e_n)|e \in S^* \cup L^* \cup P^*, e_1\blacklozenge e_2\blacklozenge e_3\blacklozenge \cdots \blacklozenge e_n\}$ where S^*, L^*, and P^* are the infinite sets of all possible surfaces, lines, and points, respectively. An image is finally defined as a finite set of spatially related objects, formally $I = O^+ \oplus \boldsymbol{V}$.

While the typical image-coding models are based on an array of nonoverlapping uniform rectangles, the elements in this model are spatially unconstrained in their shape and localization. Every image can be considered to be composed of these primitive elements. The discontinuities in the image form the boundaries between each element. The nature of the discontinuities defines the type of primitive inferred. For example, macrodiscontinuities such as edges bound the region defining a smooth-shaded surface patch. A microdiscontinuity will bound the region defining a flat-shaded polygon or vector. In this sense, any image can be considered to be a product of primitive instancing.

A simplistic example can best serve to illustrate the application of this model. Fig. 16 is an image composed of four identifiable foreground objects (sun, plant, ant, and cloud) and the background. Each of these occurs in a certain spatial relationship to the other. Placing a grid of 9×7 elements over the image, we can additionally specify the spatial relationships between each of these objects in reference to an origin, say the bottom left-hand corner. The respective relations then become approximately $\{(0, 6), (5, 4), (2, 1), (7, 5)\}$, which could alternatively be defined relative to each other. The full definition of the diagram becomes the set of constituent elements, the relationships between them, and the parameterizations for each element, which have been omitted for simplicity. The symbols O^+, L^+, S^+, and P^+, respectively, denote the sets of all objects, lines, surfaces, and pixels in the image (see Fig. 16).

2) Audio Model: Audio signals can be represented as a time-frequency distribution defined as an ordered set of three-tuples (f, y, i) where i is the amplitude of the signal at frequency f and time y. A time-frequency representation of audio then becomes $A = \{(f, y, i)|0 \leq f \leq h, 0 \leq y \leq l, 0 \leq i \leq 1\}$.

The most primitive audio element is a pure tone burst or spectral line localized in both time and frequency. In its simplest form it is defined by a point (f, y, i) that has constant intensity and unit-length duration. A noise burst is a set of statistically uncorrelated points. The general form of a tone burst has arbitrary duration and is time variant in intensity. This is parametrically defined as $F = \{[f, y(t), i(t)]|0 \leq t \leq 1\}$ being localized in both time and frequency and having an intensity contour.

The set of unit tone bursts lying on a path Q, which is permitted to vary parametrically in frequency, forms a frequency track. These have the additional property of pitch contour and are defined as $T = \{[f(t), y(t), i(t)]|0 \leq t \leq 1\}$. A set of m synchronous tracks, which follow the same path but are displaced in frequency, form a harmonic group that is defined as $H = \{t(n)|t(0) = T, t(n+1) = (f, 0, 0) + t(n), 0 \leq f \leq h, 0 \leq n < m)\}$. These (Fig. 17) have the additional properties of harmony or timbre.

A phrase or moneme M is defined as a temporally connected group of frequency tracks, tone bursts, and harmonic groups. Translating each of these to the origin, the spectral and temporal relationships between them are given by the set of vectors $\boldsymbol{V} \subset \Re^2$. Thus $M = \boldsymbol{V} \oplus \{(e_1, e_2, e_3, \cdots, e_n)|e \in T^* \cup F^* \cup H^*\}$ where T^*, F^*, and H^* are the infinite sets of all possible tones, tracks, and harmonic groups, respectively. Both rhythm and meter are higher level concepts beyond the scope of this analysis.

3) Video Model: At the most primitive level, a video episode can be considered to be composed of an

Image = {**sun, plant, ant, cloud** | **cloud** $\in O^+$} $\oplus$ {(0, 6), (5, 4), (2, 1), (7, 5)}

sun = {corona, 5 rays | corona $\in S^+$, rays $\subset L^+$} $\oplus$ {V | rays radiate out from corona}

plant = {centre, stem, 8 petals | stem $\in S^+$} $\oplus$ {V | stem is beneath centre, radiating petals}

centre = {oval, spots | oval $\in S^+$, spots $\subset P^+$} $\oplus$ {V | spots lie within oval}

petals = {surface, texture | texture $\subset P^+$} $\oplus$ {V | texture radiates from centre}

ant = {head, thorax, abdomen | thorax $\in S^+$} $\oplus$ {V | thorax ♦ head ♦ abdomen}

head = {oval, antennae | oval $\in S^+$, antennae $\subset L^+$} $\oplus$ {V | antennae above head}

thorax = {oval, legs | oval $\in S^+$, legs $\subset L^+$} $\oplus$ {V | legs below body}

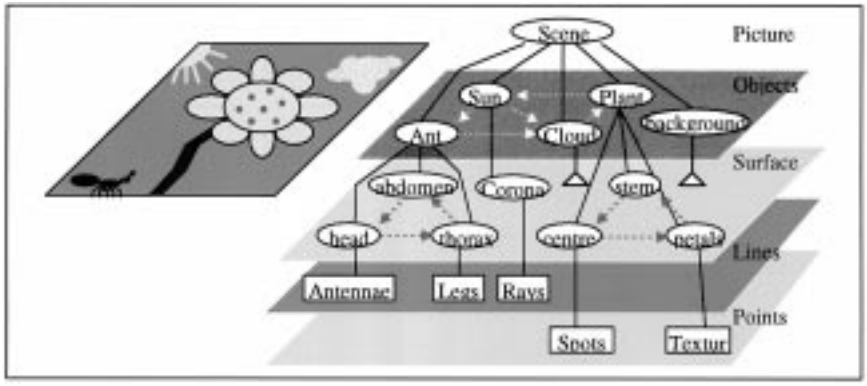


Fig. 16. Image articulation.

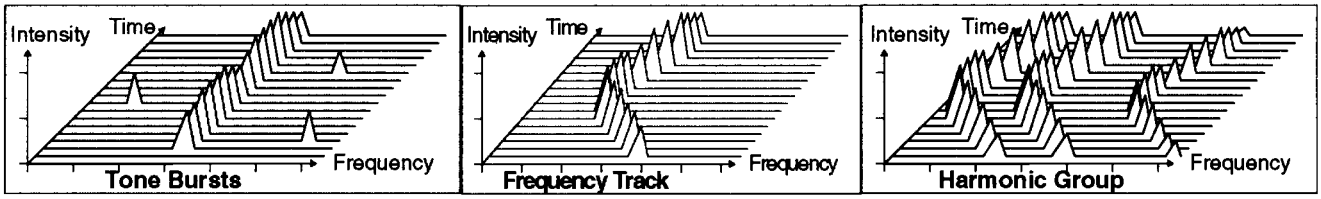


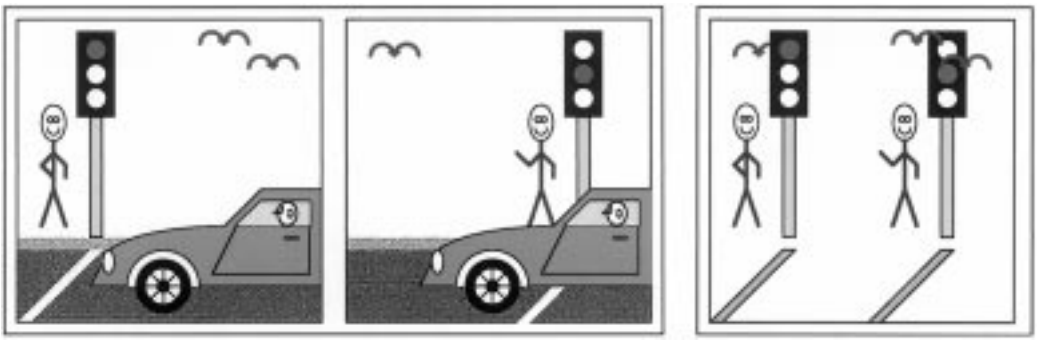
Fig. 17. Audio structural primitives.

ordered set of scenes. A scene is a set of images $\{(I_0, I_1, I_2, \cdots, I_n)|I_n \approx I_{n+1}\}$ delimited by a scene change or sharp discontinuity $I_n \ll\gg I_{n+1}$. A scene may be interpreted as a single time-variant image, denoted $I(t)$, but it may be equivalently expressed as an ordered set of tuples $[x, y, i(t)]$ where i is the color value of the scene at a location or point (x, y) for the image at time t. Formally, a scene is the set $\{[x, y, i(t)]|[x, y, i(t)] \approx [x, y, i(t+1)], 0 \le x \le w, 0 \le y \le h, 0 \le i \le 1, 0 \le t \le n\}$. Three basic types of time-variant phenomena may occur over the duration of a scene: stationary change (SC), planar motion (PM), and global motion. Fig. 18(a) depicts a two-image scene exemplifying all three types of changes and Fig. 18(b) shows all of the nine regions in the scene that change over the two images.

Stationary change is the simplest primitive produced by only an "in-place" color change. This is defined as the set of time-variant color pixels such that $\mathrm{SC} = \{[\mathrm{x}, \mathrm{y}, \mathrm{i}(\mathrm{t})]| \exists \mathrm{t} \in \Re|\mathrm{i}(\mathrm{t}) \neq \mathrm{i}(\mathrm{t}+1)\}$. While the SC primitive assumes that changed pixels are spatially fixed, an alternative interpretation considers the color value of each pixel to be fixed and its spatial location to be time variant. This gives rise to planar motion where the direction and distance of the motion is parametrically defined in time as the set of points such that $\mathrm{PM} = \{[\mathrm{x}(\mathrm{t}), \mathrm{y}(\mathrm{t}), \mathrm{i}]| \exists \mathrm{t} \in \Re|\mathrm{i}(\mathrm{t}) \neq \mathrm{i}(\mathrm{t}+1)\}$.

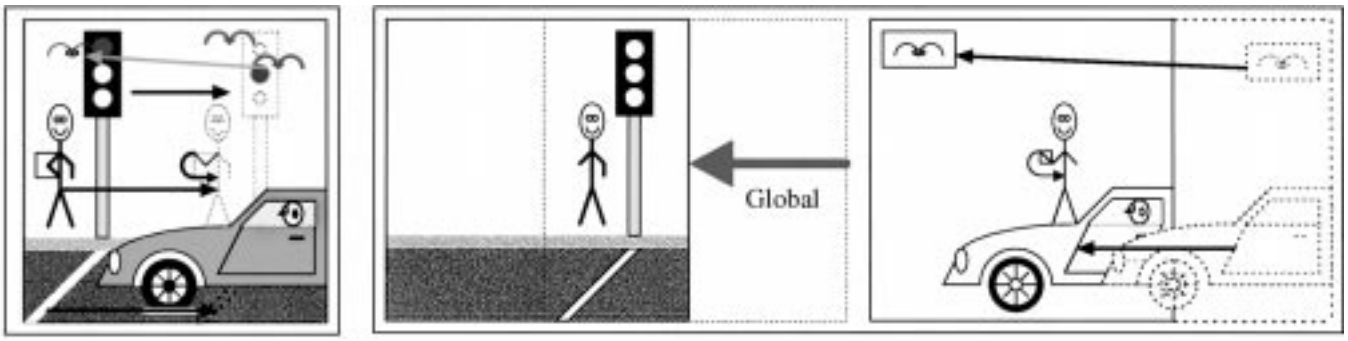
Stationary change or planar motion normally occur simultaneously to groups of adjacent pixels in given domains of the scene having the properties of the region shape and the nature of the change. In these cases, instead of modeling motion at the pixel level, it is more convenient to do so at the region or object level. Hence, an image undergoing planar motion can best be described as a set of objects whose spatial relationships are time variant. From the image model, we have $I(t) = O^+ \oplus \boldsymbol{V}(t)$ where O^+ is the set of all objects in the scene and the relationship vectors are temporally variant, becoming relational motion vectors $\boldsymbol{V}(t) = \{[v_0(t), v_1(t), v_2(t), \cdots, v_n(t)]|0 \le t \le \mathrm{frames}\}$. Rather than just describing the two-dimensional translational motion, this planar-motion definition may be extended to account for layered motion (LM) by defining the vectors as three-dimensional entities $\boldsymbol{v}(t) = [x(t), y(t), z(t)]$ where $z(t)$ specifies the depth information.

Since planar motion assumes the content of each region to remain constant, it is unable to account for all of the



(a) (b)

Fig. 18. (a) Motion in a scene. (b) Stationary change.



(a) (b)

Fig. 19. (a) Planar motion vectors. (b) Global motion compensation in scene.

changes in Fig. 18. In this case, it is necessary to fall back onto stationary change to complete the scene. Fig. 19(a) shows the five motion vectors for the scene, which includes one bird and the street marking, person, and traffic light. It is unable to account for the changed state of the traffic light or the missing bird, which must be compensated for using the SC primitive.

Since planar or simple layered motion can only exist in very constrained environments, this definition is too restrictive to model the motion of real objects since these are subject to more complex 3-D transformations. In this case, it is convenient to consider a scene to be a time-constrained view of a 3-D space. Returning to our object image model $I(t) = O^{+} \oplus \boldsymbol{V}(t)$, the vectors become six-dimensional time-variant entities, each defining an affine transform. This permits the objects to move in simulated 3-D space by rotating, translating, and scaling. Planar and layered motion are just special cases of object motion, which is defined by motion vectors of the form

$$\boldsymbol{v}(t) = \begin{bmatrix} a(t) & b(t) & x(t) \\ c(t) & d(t) & y(t) \\ p(t) & q(t) & 1 \end{bmatrix}$$

where the translation is defined by the parameters x, y, the scaling by a, b, and the rotation by the combination of a, b, c, d. Perspective transformations are accounted for by parameters p, q. Rather than just considering the motion of independent regions, uniformity in the time-variant set of motion-relationship vectors may be suggestive of the existence of global motion. This global motion also takes the form of an affine or perspective transform that applies to the entire image. Taking the global motion vector component $g(t)$ into consideration, the set of relational motion vectors in a scene becomes $\boldsymbol{V}(t) = g(t) \bullet W(t)$ where $W(t)$ defines the globally compensated local motion vectors. Since the use of multiple motion models is required to describe the motion in a scene adequately, then the addition of the global-motion model further enhances the flexibility. In this case, compensating for the translational global motion reduces to the number of motion vectors required to describe the motion. Fig. 19(b) shows the result of applying a global motion transformation to the previous example. Three local motion vectors and two cases of stationary change must be updated.

A set of frames in a scene where the global motion is uniform defines a shot. A scene can be considered to be composed of an ordered subset of shots. A shot is formally defined as the set of adjacent images such that $I(t) = O^{+} \oplus [g(t) \bullet W(t)] | g(t) = g(t+1),\ 0 \leq t \leq$ frames.

C. Content-Based Access and Management

In addition to forming the basis for compact representations, these data models permit the access and management techniques of the general cognitive memory model to be imitated in various ways. The memory model consists of a modality-specific sensory store that receives and analogically encodes input data from the perceptual processes. This data is then passed to the working memory, where

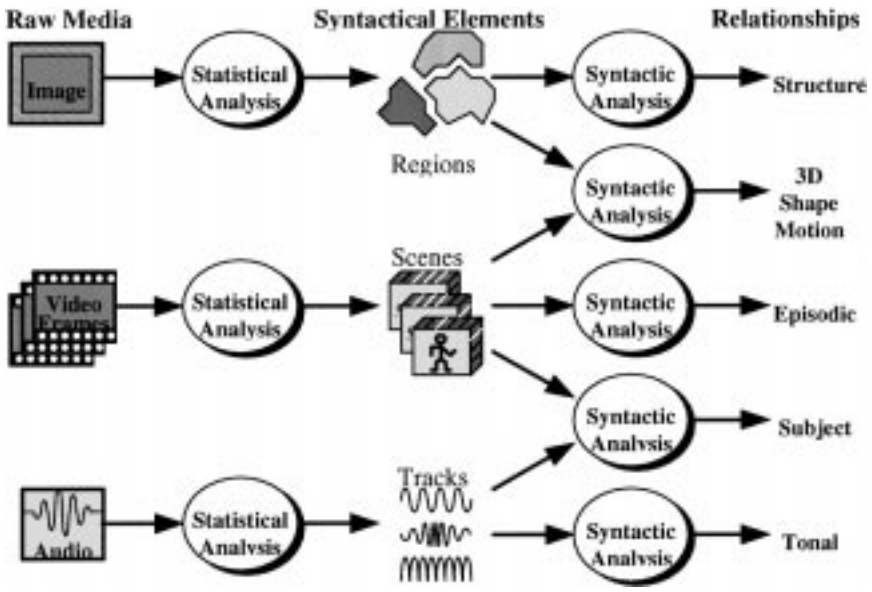


Fig. 20. Syntactic analysis.

control is consciously performed, and finally placed in long-term storage. Rather than containing representations of the actual sensory data, the working memory operates with labels or pointers to conceptual objects or groups of objects, permitting the manipulation of highly complex entities or groups of these. The counterparts to these components in a hypermedia system are the input processing, hypermedia engine, and hyperbase.

The input-processing section of such a system should accept raw multimedia data and encode it according to the semiotic data model. Following this, the hypermedia engine can effortlessly mediate in all of the user-directed control functions such as navigation, authoring, or integrating new data elements into the existing hyperbase. Content-based retrieval functions can be performed by directly querying the node attributes. Additionally, in contrast to existing compressed time-based browsing of audio and video data streams, browsing can be content-based since the support for this is intrinsically provided by the data models. To be particularly effective, however, this access should be based on semiotic signs generated from the subsigns through the application of Gestalt principles.

In the case of content-based retrieval, there is no need for additional processing to extract labels or create indexes since the data models explicitly represent the data in terms of their salient attributes. These models provide a wide range of information for generating component labels, which may be interpreted to be of a statistical, syntactical, or semantic nature. Primarily, statistical information is directly encapsulated by them since the primitive elements are statistically defined. More abstract information is provided by the syntactic analysis of the relationships between the elements and their defining attributes. Depending on the analysis domain, this high-level syntactical analysis may provide either structural, episodic, tonal, 3-D motion and shape, or subject information. This information may be used in conjunction with Gestalt principles or audio-stream segregation to generate semantic groupings from the syntactical units. For example, adjacent image primitives that have synchronized motion have a high probability of forming a semantically consistent and meaningful object. Fig. 20 demonstrates the basic concept behind this analysis.

The lowest level elements in the data models readily produce statistical information about their subject. For example, the average color or texture can be directly calculated from the pixels or vectors in an image. In video, the amount of stationary change is a good indication of the occurrence of scene changes. Likewise, in the audio domain, the energy distribution exhibited by tone bursts can be used to perform source classification. Music predominantly consists of long harmonic tracks with rare periods of silence. Noise is composed of many short discordant tone bursts. Silence is defined as the absence of any frequency tracks. Speech is a combination of relatively short noise and tone bursts interspersed with frequency sweeps and many pauses.

Alternatively, syntactical information is predominantly conveyed by the primitive elements at the higher levels. In images, the relationships between the region segmentation suggested by the surface elements may be used for structural analysis. Shot classification in video also conveys syntactic information by defining the relationship between frames and image elements in a scene over time. In audio, the relationship between temporally adjacent frequency tracks can be directly used for retrieval by defining pitch contours or melodies. Additionally, speaker changes and gender may be determined from the fundamental frequency or pitch of the lowest track. Speaker emphasis may be detected from the change in relative amplitude of the tracks.

The highest level elements may be used to generate semantic information directly. The surface shading of objects in a scene can be used to perform shape estimation and identify region shapes used for object recognition. The generation of this semantic information may often

Table 12

Attributes	Still Images	Video	Audio
Statistical	Colour and Texture Geometric moments	Scene Change Detection Scene lengths, activity.	Spectral Energy Distribution Dynamics, Silence detection
Syntactical	Image and segmentation structure	Shot Classification Scene transition graphs Object motion tracking	Pitch contours, Timbre, Source Classification, Chord Structure,
Semantic	Shape Estimation Object Recognition	Motion Estimation Gesture Analysis	Speech and melody recognition, word spotting,

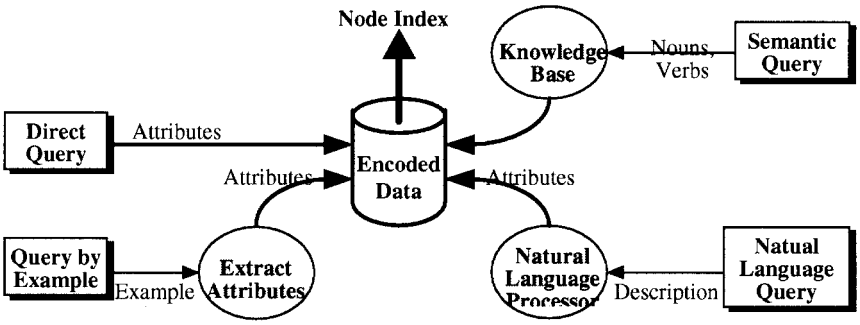


Fig. 21. Query processing for content-based retrieval.

be contingent on the specification of appropriate grouping rules or the nature of the relationships between primitive elements. Certain groupings may have specific semantic interpretations such as in face recognition. In audio, certain harmonic groupings may be indicative of particular musical instruments. Additionally, even with these higher level primitives, the statistical and syntactical attributes are still readily available since each element is linearly decomposable into its constituent lower level elements. Table 12 summarizes information that may be immediately extracted or directly generated from the model.

To provide effective content-based retrieval, a variety of query methods must be supported, which must be mapped into the attribute set irrespective of how they are posed. Given an attribute set, queries may be posed by directly specifying attribute values. For example, to locate a red image, one could enter $\langle 255, 0, 0 \rangle$ into the color field of a form-based query. Alternatively, a natural language interface would permit the expression of these values within language like constructs by saying "find images with color 255, 0, 0." A more advanced query mechanism would rely on the mediation of an expert system where the knowledge base would maintain descriptive lists of real-world objects in terms of their attributes. Posing a semantic query by specifying the name of an object (e.g., find a red image) would result in a set of attributes' being submitted to the retrieval engine. All of these queries are fundamentally similar in that they are expressed alphanumerically. Alternatively, a query may be posed by example. Synthetic visual queries could be formulated by using a drawing tool or by compositing an image from a feature database. Similar methods can be used for audio queries, such as humming a tune. Fig. 21 shows some options for posing queries and their mapping mechanisms.

D. Example Representations

Thus far, the data models and their management aspects have been discussed but not the issue of their derivative representations and encoding. This will determine the accessibility of the information provided by the data models. This section describes rudimentary audio and video representation schemes, which seek to make this information explicitly available in compressed form. These coding schemes are two specific instances of the data models and are by no means definitive. They permit information management data and interactive manipulation in its compressed form to a certain extent. More work is required to develop representations that fully implement the data models.

The communication systems of advanced animals are composed of three classes of audio signals: noise bursts, tone bursts, and frequency sweeps. Frequency resolution is better at low frequencies while temporal resolution is better at high frequencies, leading to a scalogram-like time-frequency distribution (TFD). Instead of an STFT, a multiresolution discrete cosine transform (MDCT) is used to generate the frequency decomposition since the DCT domain has the advantage that it does not require separate phase information and has higher data compaction. The MDCT is preferred due to lower blocking artifacts. Constant-length, variable-resolution analysis windows are used to generate the analysis bands separated by octaves [Fig. 22(a)]. The higher frequency bands exhibit low frequency but high temporal resolution while the low frequency bands have high frequency and low temporal resolution.

The overall encoding algorithm is depicted in Fig. 22(b). After the scalogram is generated, it is processed by applying masking and quiet thresholds to remove perceptually redundant data. Peak picking and tracking is then performed to

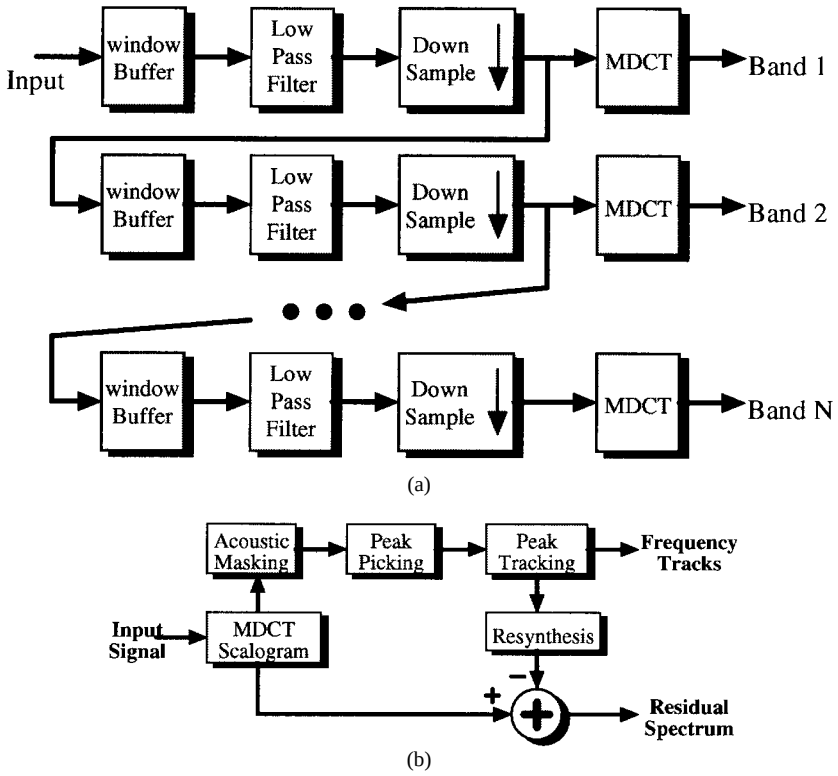


Fig. 22. (a) MDCT scalogram generation. (b) Coding algorithm.

extract the frequency tracks. Peaks are defined as the set of all points in the TFD such that $|X(f_{i-1})| < |X(f_i)| > |X(f_{i+1})|$ where $X(f)$ is the magnitude at frequency f at each instance i in time. Tracking is performed by joining the peaks in time that are closest in frequency within limits. Each track is then classified according to its type (noise, tone burst, or sweep) and represented as differential chain codes. Following simple stream segregation, the tracks are encoded in groups. The residual spectrum is differentially encoded as a set of unit-length tone bursts. The representation structure is defined as a set of groups, sweeps, and tones. To simplify access, the header information contains a table of contents in chronological order of the elements containing only the type of element, its starting time, and a pointer to the location of any additional parameters in the stream. This structure is as follows:

$$\begin{aligned}
\text{AUDIO} &= \{\text{(Header Information, GROUP}^*\text{, TRACK}^*\text{,} \\
&\qquad \text{TONE}^*\text{, NOISE}^*\} \\
\text{GROUP} &= \{\text{SWEEP, nPartials, offsets[nPartials],} \\
&\qquad \text{energy[nPartials]}\} \\
\text{SWEEP} &= \{\text{TONE, frequency[length]}\} \\
\text{TONE} &= \{\text{length, frequency, intensity[length]}\} \\
\text{NOISE} &= \{\text{frequency, intensity}\}.
\end{aligned}$$

Using this representation, it is immediately possible to perform source classification by looking at the length, track type, and frequency localization. Following from this, it is possible to perform further class-specific analysis on the tracks. Since the tracks are defined parametrically, this analysis amounts simply to comparing the values of the parameters for each track. Voiced/unvoiced speech detection can be performed by evaluating the ratio of noise to tracks over a short time period. Change of speaker and gender may be determined from the frequency of the lowest track. Speaker emphasis may be detected from the change in relative amplitude of the tracks. The possibility of the use of this feature set for speech recognition needs to be investigated. The distribution of partials in harmonic groups can be analyzed to evaluate timbre. Other attributes may be directly determined from the tracks, such as the modulation, tempo, frequency, duration, dynamics, periodicity, pitch contour, and harmonics of the audio data, that may be used for content-based retrieval.

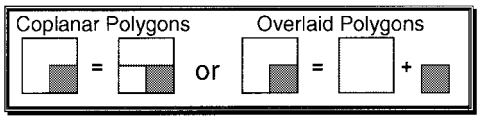


Fig. 23. Primitive layering.

In the video representation [125], each frame is viewed as an image composed of primitive elements in various layers at different levels of detail. This permits complex-shaped elements to be defined in terms of an overlay of simpler elements. The representational efficiency of using simple overlaid primitives (Fig. 23) instead of coplanar primitives is apparent in that only two overlaid instead of three coplanar rectangles are required to represent the same complex pattern. Since the sensitivity of the human visual system decreases under temporal variations in the stimulus, an image can be updated progressively in terms of these layers and the finer detail layers can be updated at a slower rate.

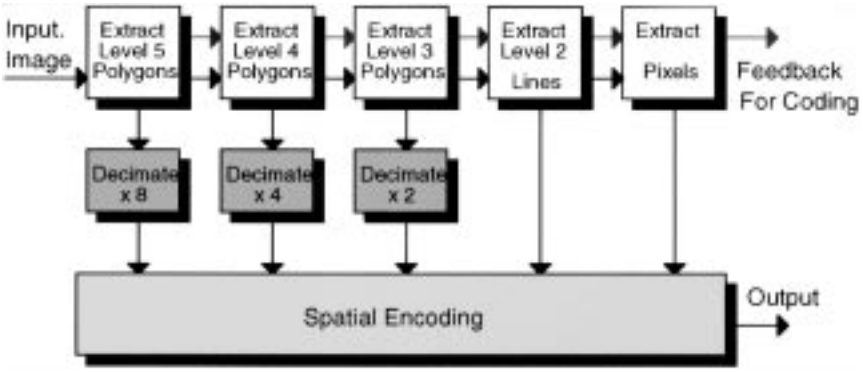


Fig. 24. Spatial layering.

The intraframe or image encoding algorithm extracts the primitive elements from each level in succeeding stages, as shown in Fig. 24. The encoder initially attempts approximately to fit the surfaces that are the largest primitives to the underlying image. The encoding process then proceeds by attempting to fit the next lower level, smaller primitives to the residual data. The adequacy of the fit is based on estimating the cost of encoding the same data using a coplanar arrangement of lower order primitives alone versus a layered representation. If no coding gain is achieved through encoding the region as the high-order primitive plus the required lower order primitives, then it is discarded and the region is encoded with the lower order primitives alone. This provides a multiresolution representation since each layer consists of primitives of a different spatial/temporal/spectral resolution. Less obvious is that it also provides a multilevel information system since each layer has different semantic value. To simplify development of the representation, however, it has been restricted to utilizing constrained lower level primitive elements.

The intraframe data model consists of pixels, lines, and surfaces. The lines may be of any orientation but are straight. Since the surfaces are also constrained to being flat-shaded rectangles with predefined sizes for simplicity, additional flexibility is provided by extracting each sized surface as a separate layer. No attempt at this stage has been made to isolate objects as special groupings of these primitives, and further work is required in this area. The defining attribute of each primitive is its type or shape, while its characteristic attributes include color, size, and orientation. The relationships between the elements are encoded as their relative position along a path that visits all the elements per layer. This path is defined by a pseudo-random adaptive raster scanning technique based on predicting both the scan direction and changes in direction [126].

The representation encodes each type of primitive in separate layers and grouped by color, since the video representation is color mapped. The representation for a layer becomes a layer-type specifier followed by the number of different color groups and each color group. Each color group is defined as the color for the group, the number of elements in the group, and the string of elements. Each element is defined by its spatial relationship with its preceding element and any additional shape information. The resulting information is encoded using variable-length codes. The simplified structure of the representation is as follows:

$$\begin{aligned}
\text{VIDEO} &= \{(\text{Header Information, Color Map,} \\
&\qquad \text{IMAGES[frames]})\} \\
\text{IMAGES} &= \{\text{SURFGROUP, LINEGROUP,} \\
&\qquad \text{PELGROUP}\}\text{; Layers of PGROUP} \\
\text{PGROUP} &= \{(\text{nColors, CGROUP[nColors]})\} \\
\text{CGROUP} &= \{(\text{Color, quantity, PRIMITIVES} \\
&\qquad \text{[quantity]})\} \\
\text{PRIMITIVES} &= \{\text{localization, shape parameterization}\}
\end{aligned}$$

Fig. 25 shows the progressive or layer-based update of both full images and a conditional replenishment image. In the interframe coding, only stationary changes have been modeled. Instead of spreading the update data over a larger time window, as is typical to reduce burstiness, we perform spatially localized temporal subsampling by temporally modulating the replenishment threshold. Varying the amplitude and period of the modulation allows high-contrast areas to be updated at a higher rate than low-contrast areas. This is in accordance with perceptual psychology since high-contrast areas are more quickly detected than low-contrast areas due to the characteristics of the probability summation detection process. In combination with the layered representation, this forms a new approach to exploiting the reduced sensitivity of the visual system based on spatio-temporal layering. This allows one to restrict spatial subsampling to specific regions of the image where motion is occurring as well as controlling the replenishment rate of individual regions based on the perceived contrast change. If a primitive at any layer has been encoded, none of the lower layers will be encoded for the area covered by that primitive during that frame. The finer resolution data will have to wait for the next frame to be updated. Additionally, deliberate dropping of the lower level data that has higher resolution permits graceful degradation to occur.

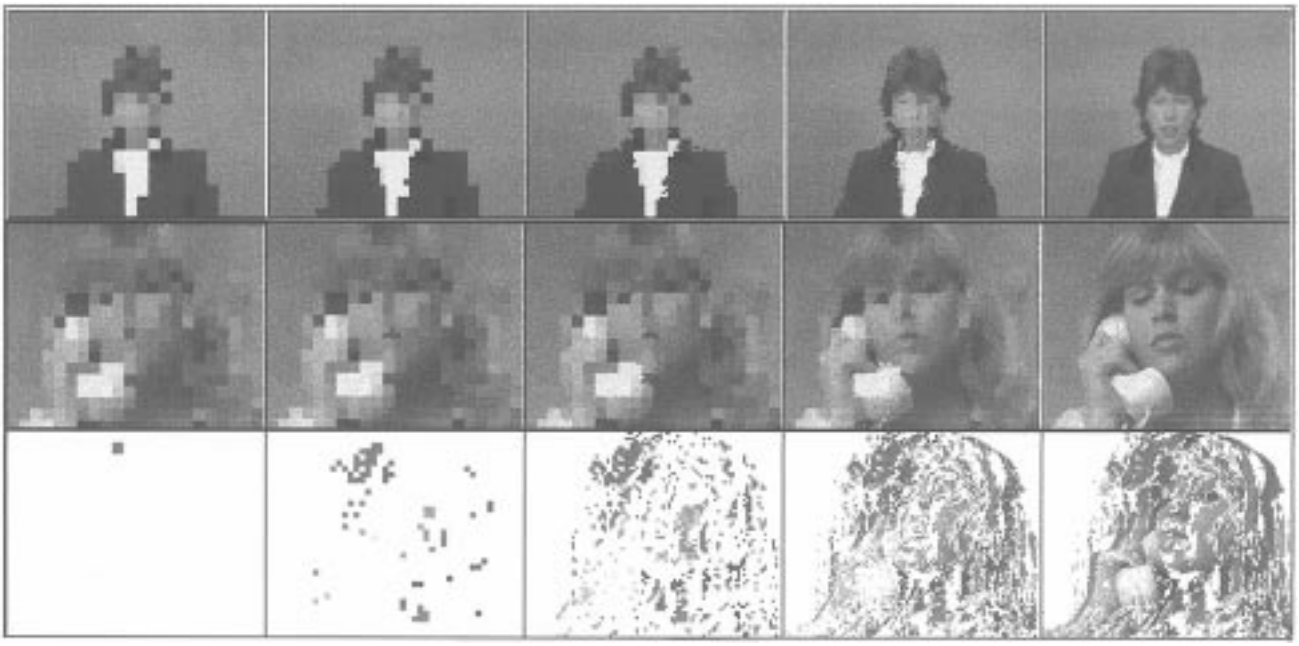


Fig. 25. Layered images.

This algorithm is ideally suited to scene-change replenishment, bringing the higher contrast changes into view first and then gradually bringing into view the changes that are of a lower contrast nature. Since the human visual system is particularly sensitive to edges, and especially moving edges, this scheme is also aptly suited to handling image motion. This is because a moving edge generally constitutes high-contrast changes, which will be updated faster than areas where the change is due to a moving image gradient. These will constitute low-contrast changes and need not be updated as quickly since their detection will be slower.

While maintaining relatively high image quality, this manipulable representation scheme can achieve compressed data rates on par with traditional schemes. The video encoding gives compression ratios around 20:1 to 80:1 depending on the amount of motion in the video. The average data rates for quarter common intermediate format (176×144 pixels) teleconferencing-type color video are about 2 kb per frame. Because of the decoding simplicity, faster than real time decoding is possible even on low-power PC's. For example, on an IBM PC 486/33 MHz with a standard VGA display and ISA bus, decoding can achieve rates of up to 70 frames per second on average.

The versatility of the representation for content-based retrieval is evident since scene changes can be detected simply by a combination of thresholding the number of primitives per frame and the change in color composition given by the color fields. At most, this involves only adding together a few of the data fields in the representation. The average image color similarly can be calculated from the color and number of elements in each color group. The nature of the texture is given by the shapes and sizes of elements and their color contrast in any given region. Additionally, distinctly colored objects can be directly isolated and manipulated by simply identifying which color group(s) they belong to. Higher level information would be available given the full implementation of the data model, which requires further work.

VII. Conclusions

This paper has presented a new semiotic paradigm for hypermedia data modeling. Commencing with the hypermedia vision from its inception and progressing to the existing multimedia extended hypertext systems, a brief review has been presented of the data-model-related issues along with the existing shortcomings and requirements for true hypermedia systems. The retarded state of multimedia technology in this situation is prevalent in its deficiency to provide random and associative (content-based) access, interactive manipulation, and a structured representation. Since the only objective of multimedia encoding has been compression, the resulting bit-stream-based data model is antagonistic to these requirements. The necessity for manipulable representations based on suitable data models is mandated to achieve the ultimate goals of hypermedia. Semiotics is presented as an avenue by which to achieve these goals.

The goals of hypermedia demand the consideration of cognitive and semiotic issues as the basis for any proposed data models for hypermedia. This will impinge on the nature of the information conveyed or encapsulated by the data model itself. Of the three domains that may form the basis of this information—semantic, syntactic, and statistical—only the syntactic domain is capable of providing the framework required to generate suitable data

models. Modeling the data as semantic units requires human intervention since semantic analysis is subject to a constant need for knowledge, being unable to cope in unconstrained environments. Statistical data models, which are currently used for obtaining compression, are unstructured and cannot convey any meaning about the data. Alternatively, modeling the data as syntactic units can be performed automatically, and there is significant evidence regarding the role of syntactical analysis in cognition. Since semantics can arise within a grouping of syntactical units, being able explicitly to access and interactively manipulate the syntactic units in a given media allows one to generate new semantics by restructuring them.

A review of the existing coding models used for vector graphics, image, video, and audio representations reveals their unsuitability for hypermedia since they virtually encrypt the underlying information. Coding schemes are required that can provide both compression and support for retrieval. Additionally, a review of existing multimedia information-management technologies reveals that existing management support is external to the data itself, relying on separate indexes, and is highly dependent on semantic methods. The same level of random and content-based access provided by multimedia databases should be supported by hypermedia systems. Rather than supporting this functionality through separate indexes, as is currently done, this should be intrinsically supported within the encoded hypermedia data through the mechanism of the data-coding model.

The cognitive principles governing semiosis used to formulate suitable hypermedia data models were presented in a brief review of cognition, semiotics, and perceptual psychology. Some of these principles include the dependence of semantic understanding on syntactic processes and the suggestion that structural (syntactic) understanding is processed distinctly from, yet simultaneously with, recognition or semantic understanding. The grouping of suitable syntactic elements to form perceptually significant units was discussed. Gestalt theory cannot define the nature of these elements but does suggest some general grouping rules. Alternatively, semiotics has traditionally defined a double articulation as being composed of the smallest semantically meaningful units of data (signs) and their constituent elements (subsigns). To extend and apply these principles to hypermedia data models, it is required recursively to decompose these signs into subsigns. This decomposition permits the definition of primitive elements that are known to combine syntactically into signs but it does not provide a grammar to define how they may be combined. The process of encoding a given data set in terms of these syntactic elements essentially becomes the task of inferring a grammar or alternatively defining the relationships required between elements to reconstruct the original data set.

Based on this framework, a new semiotic paradigm has been proposed for hypermedia data models and representations. Cognitively based semiotic articulations for multimedia data have been identified from which semiotic data models have been proposed for image, video, and audio data, permitting structured data representations to be developed. Each model element is separately treated in terms of its defining, characteristic, and relational attributes. The data models support content-based access to the data by providing direct access to statistical and syntactic information, and may be used to infer semantic information as well. The suitability of these data models is demonstrated through rudimentary encoding schemes, which provide compact representations while preserving direct access to the underlying information for content-based retrieval purposes. Further work involves a complete implementation of the data models for the various modalities and the extension of the models to include semantically more significant articulated and deformable objects.

In conclusion, a new semiotic paradigm has been proposed for hypermedia data modeling and the basis for hypermedia representations. The need for the new paradigm has been established and its relationship with existing technologies in hypermedia has been presented. Data models based on semiotic articulation for multimedia data have also been proposed, and their utility as the basis for hypermedia representations has been demonstrated and explored.

ACKNOWLEDGMENT

The author wishes to express his gratitude to Dr. A. Qureshi for his considerable assistance in the preparation of this manuscript, to A. Wardhani and K. Melih for their contributions, and to the reviewers for their valuable comments.

REFERENCES

[1] J. Conklin, "HyperText: An introduction and survey," *IEEE Comput. Mag.*, pp. 17–41, Sept. 1987.
[2] M. H. O'Docherty and C. N. Daskalakis, "Multimedia information systems—The management and semantic retrieval of all electronic data types," *Comput. J.*, vol. 34, no. 3, pp. 225–238, 1991.
[3] J. L. Schnase, J. J. Leggett, D. L. Hicks, and R. L. Szabo, "Semantic data modeling of hypermedia associations," *ACM Trans. Inform. Syst.*, vol. 11, no. 1, pp. 27–50, 1993.
[4] G. H. Scholss and M. J. Wynblatt, "Providing definition and temporal structure for multimedia data," *Multimedia Syst.*, vol. 3, pp. 264–277, 1995.
[5] J. Gu and E. J. Neuhold, "A data model for multimedia information retrieval," in *Proc. 1st Int. Conf. Multimedia Modeling*, Singapore, Nov. 9–12, 1993, pp. 113–127.
[6] C. Meghini, F. Rabitti, and C. Thanos, "Conceptual modeling of multimedia documents," *IEEE Comput. Mag.*, pp. 23–29, Oct. 1991.
[7] V. Bush, "As we may think," *Atlantic Monthly*, pp. 101–108, July 1945.
[8] D. C. Engelbart, "A conceptual framework for the augmentation of man's intellect," in *Vistas Inform. Handling*, vol. 1. Washington, D.C.: Sparton Books, 1963.
[9] T. Nelson, "Getting it out of our system," in *Information Retrieval: A Critical Review*, G. Schechter, Ed. Washington, D.C.: Thompson, 1967.
[10] F. Halasz and M. Schwartz, "The Dexter hypertext reference model," *Commun. ACM*, vol. 37, no. 2, pp. 30–39, Feb. 1994.
[11] L. Hardman, D. C. A. Bulterman, and G. van Rossum, "The Amsterdam hypermedia model: Adding time and context to the Dexter model," *Commun. ACM*, vol. 37, no. 2, pp. 50–62, Feb. 1994.

[12] Z. Chen, S. Tan, R. Campbell, and Y. Li, "Real time video and audio in the World Wide Web," in *Proc. 4th Int. World Wide Web Conf.*, Boston, MA, 1995.
[13] "MPEG-4 synthetic/natural hybrid coding call for proposals," ISO/IEC JTC1/SC29/WG11 N1195, Firenze, Italy, Mar. 1996.
[14] F. G. Halasz, "Reflections on notecards: Seven issues for the next generation of hypermedia systems," *Commun. ACM*, vol. 31, no. 7, pp. 836–852, July 1988.
[15] M. Dunlop, "Multimedia information retrieval," Ph.D. dissertation, Glasgow University, Rep. 1991/R21.
[16] D. R. Hardy and M. F. Schwartz, "Customized information extraction as a basis for resource discovery," Dept. Computer Science, University of Colorado, Boulder, Tech. Rep. CU-CS-707-94, Mar. 1994; revised Feb. 1995. See also *ACM Trans. Comput. Syst.*, to be published.
[17] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, "Efficient and effective querying by image content," *J. Intell. Inform. Syst.*, no. 3, pp. 231–262, 1994.
[18] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steel, and P. Yanker, "Query by image and video content: The QBIC system," *IEEE Comput. Mag.*, pp. 23–31, Sept. 1995.
[19] R. Weiss, A. Duda, and D. K. Gifford, "Composition and search with a video algebra," *IEEE Multimedia Mag.*, pp. 12–25, Spring 1995.
[20] D. Swanberg, C.-F. Shu, and R. Jain, "Knowledge guided parsing in video databases," in *Proc. IS&T/SPIE Symp. Electronic Imaging: Science and Technology*, San Jose, CA, Feb. 1993.
[21] H. Zhang, S. Y. Tan, S. W. Smoliar, and Y. Gong, "Automatic parsing and indexing of news video," *Multimedia Syst.*, vol. 2, pp. 256–266, 1995.
[22] S. W. Smoliar and H. Zhang, "Content-based video indexing and retrieval," *IEEE Multimedia Mag.*, pp. 62–72, Summer 1994.
[23] Y. Nakajima, "A video browsing using fast scene cut detection for an efficient networked video database access," *IEICE Trans. Inform. and Syst.*, vol. E77-D, no. 12, Dec. 1994.
[24] H. J. Zhang, S. W. Smoliar, and J. H. Wu, "Content-based video browsing tools," in *Proc. SPIE Multimedia Computing and Networks*, San Jose, CA, Feb. 6–8, 1995, vol. 2417, pp. 389–398.
[25] L. Teodosio and W. Bender, "Salient video stills," in *Proc. ACM Multimedia'93*, Anaheim, CA, 1993, pp. 39–46.
[26] N. Dimitrova and F. Golshani, "Motion recovery for video content classification," *ACM Trans. Inform. Syst.*, vol. 13, no. 4, pp. 408–439, Oct. 1995.
[27] A. Zakhor and F. Lari, "Edge-based 3-D camera motion estimation with application to video coding," *IEEE Trans. Image Processing*, vol. 2, pp. 481–498, Oct. 1993.
[28] Y. Tonomura, A. Akutsu, Y. Taniguchi, and G. Suzuki, "Structured video computing," *IEEE Multimedia Mag.*, pp. 34–43, Fall 1994.
[29] M. R. Dobie and P. H. Lewis, "Object tracking in multimedia systems," in *Proc. 4th Int. Conf. Image Processing Applications*, The Netherlands, Apr. 1992, pp. 41–44.
[30] Y. Gong and M. Sakauchi, "An object-oriented method for color video image classification using the color and motion features of video images," in *Proc. ICARCV '92, 2nd Int. Conf. Animation, Robotics Computer Vision*, Sept. 1992, paper CV-10.6.
[31] J. Picone, "Continuous speech recognition using hidden Markov models," *IEEE Acoust., Speech, Signal Processing Mag.*, pp. 26–41, July 1990.
[32] L. D. Wilcox, I. Smith, and M. A. Bush, "Wordspotting for voice editing and audio indexing," in *Proc. CHI*, Monterey, CA, Mar. 1992.
[33] L. D. Wilcox, F. R. Chen, D. G. Kimber, and V. Balasubramanian, "Segmentation of speech using speaker identification," in *Proc. ICASSP'94*, Adelaide, Australia, Apr. 1994.
[34] F. R. Chen and M. M. Withgott, "The use of emphasis to automatically summarize a spoken discourse," in *Proc. ICASSP'92*, San Francisco, CA, Mar. 1992.
[35] B. S. Atal and L. R. Rabiner, "A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 201–212, June 1976.
[36] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based classification, search and retrieval of audio," *IEEE Multimedia Mag.*, pp. 27–36, Fall 1996.
[37] P. Aigrain, P. J. V. Longueville, and P. Lepain, "Representation-based user interfaces for the audiovisual library of year 2000," in *Proc. SPIE Multimedia Computing Networks*, vol. 2417, San Jose, CA, Feb. 6–8, 1995, pp. 35–45.
[38] P. S. Kumar and G. P. Babu, "Intelligent multimedia data: Data + indices + inference," *Multimedia Syst. J.*, to be published.
[39] J. K. Wu, Y. H. Ang, P. C. Lam, S. K. Moorthy, and A. D. Narasimhalu, "Facial image retrieval, identification, and inference system," in *Proc. ACM Multimedia 93*, Singapore, pp. 47–53.
[40] A. Tversky, "Features of similarity," *Psychological Rev.*, vol. 84, no. 4, pp. 327–352, July 1977.
[41] O. D. Faugeras, Ed., *Fundamentals in Computer Vision.* Cambridge: Cambridge Univ. Press, 1983.
[42] J. Sanz, Ed., *Advances in Image Processing and Machine Vision.* Berlin: Springer Verlag, 1993/1994.
[43] J. D. Foley, A. Van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics Principles and Practice*, 2nd ed. Reading, MA: Addison-Wesley, 1990.
[44] R. A. Earnshaw, R. D. Parslow, and J. R. Woodwark, Eds., *Geometric Modeling and Computer Graphics.* Brookfield, VT: Gower, 1987.
[45] A. P. Pentland, "Perceptual organization and the representation of natural form," in *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, M. A. Fischler and O. Firschein, Eds. Los Altos, CA: Morgan Kaufmann, 1987, pp. 680–699.
[46] I. P. Stewart, "Quadtrees: Storage and scan conversion," *Comput. J.*, vol. 29, no. 1, pp. 60–75, 1986.
[47] A. Poggi and G. Adoni, "An octree object-oriented geometric modeller," in *Proc. SPIE, Vol. 1293: Appl. Artificial Intelligence VIII*, 1990, pp. 152–159.
[48] A. N. Netravali and B. Prasada, "Adaptive quantization of picture signals using spatial masking," *Proc. IEEE*, vol. 65, pp. 536–548, Apr. 1977.
[49] H. G. Musmann, "Comparison of redundancy reducing codes for facsimile transmission of documents," *IEEE Trans. Commun.*, vol. COM-25, pp. 1425–1433, Nov. 1977.
[50] T. S. Huang, "Run-length coding and its extensions," in *Picture Bandwidth Compression*, T. S. Huang and O. J. Tretiakpp, Eds. New York: Gordon and Breach, 1972, pp. 231–263.
[51] T. Hata, S. Tomita, M. Nakada, and R. Ohnishi, "A graphic command coding scheme for multi-color images," in *IEEE Global Telecommun. Conf.*, Dec. 1986, pp. 1143–1149.
[52] S. Shlien, "Raster to polygon conversion of images," in *Computers & Graphics*, vol. 7, nos. 3/4. New York: Pergamon, 1983, pp. 327–332.
[53] Y. Cohen, M. S. Landy, and M. Pavel, "Hierarchical coding of binary images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 7, pp. 284–298, May 1985.
[54] H. Samet, *Applications of Spatial Data Structures.* Reading, MA: Addison-Wesley, 1989.
[55] N. M. Nasrabadi and R. A. King, "Image coding using vector quantization: A review," *IEEE Trans. Commun.*, vol. 36, pp. 957–971, Aug. 1988.
[56] M. Rabbani and P. W. Jones, *Digital Image Compression Techniques.* Bellingham, WA: SPIE, 1991.
[57] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second-generation image-coding techniques," *Proc. IEEE*, vol. 73, pp. 549–573, Apr. 1985.
[58] A. E. Jacquin, "Fractal image coding: A review," *Proc. IEEE*, vol. 81, pp. 1451–1465, Oct. 1993.
[59] Y. Liu and H. Ma, "ω-Orbit finite automata for data compression," in *Proc. Data Compression Conf. '91*, Snowbird, UT, pp. 166–175.
[60] M. Eden and M. Kocher, "On the performance of a contour coding algorithm in the context of image coding—Part I: Contour segment coding," *Signal Process.*, vol. 8, no. 4, pp. 381–386, July 1985.
[61] B. B. Chaudhuri and M. K. Kundu, "Digital line segment coding: A new efficient contour coding scheme," *Proc. Inst. Elect. Eng.—E, Comput. Digital Techniques*, vol. 131, no. 4, pp. 143–147, July 1984.
[62] T. Akimoto, Y. Suenaga, and R. S. Wallace, "Automatic creation of 3D facial models," *IEEE Comput. Graph. Appl. Mag.*, vol. 13, pp. 16–22, Sept. 1993.
[63] K. Aizawa and T. S. Huang, "Model-based image coding: Advanced video coding techniques for very low bit-rate ap-

plications," *Proc. IEEE*, vol. 83, pp. 259–271, Feb. 1995.
[64] D. E. Pearson, "Developments in model-based video coding," *Proc. IEEE*, vol. 83, pp. 892–906, June 1995.
[65] R. Forchheimer and T. Kronander, "Image coding—From waveforms to animation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 2008–2023, Dec. 1989.
[66] H. Morikawa and H. Harashima, "Structure and motion of deformable objects from image sequences," in *Proc. ICASSP'91*, Toronto, Ontario, Canada, May 14–17, 1991, pp. 2433–2436.
[67] F. Kappei and C. E. Liedtke, "Modeling of a natural 3-D scene consisting of moving objects from a sequence of monocular TV images," in *Real Time Image Processing: Concepts and Technologies*, vol. 860, *Proc. SPIE*, 1987, pp. 126–132.
[68] A. K. Jain, "Image data compression: A review," *Proc. IEEE*, vol. 69, pp. 349–389, Mar. 1981.
[69] B. G. Haskell, "Frame replenishment coding of television," in *Image Transmission Techniques.* New York: Academic, 1979, pp. 189–217.
[70] D. N. Hein and N. Ahmed, "Video compression using conditional replenishment and motion prediction," *IEEE Trans. Electromag. Compat.*, vol. EMC-26, pp. 134–142, Aug. 1984.
[71] T. Ishiguro and K. Iinuma, "Television bandwidth compression transmission by motion-compensated interframe coding," *IEEE Commun. Mag.*, vol. 20, pp. 24–30, Nov. 1982.
[72] R. J. Moorhead II, S. A. Rajala, and L. W. Cook, "Image sequence compression using a pel-recursive motion-compensated technique," *IEEE J. Select. Areas Commun.*, vol. 5, pp. 1100–1114, Aug. 1987.
[73] Y. Nakaya and H. Harashima, "An iterative motion estimation method using triangular patches for motion compensation," in *Proc. SPIE Visual Communications Image Processing'91*, vol. 1605, pp. 546–557.
[74] Y. T. Tse and R. L. Baker, "Global zoom/pan estimation and compensation for video compression," in *Proc. ICASSP'91*, Toronto, Ontario, Canada, May 14–17, 1991, pp. 2725–2728.
[75] S. C. Brofferio, "An object-background image model for predictive video coding," *IEEE Trans. Commun.*, vol. 37, pp. 1391–1394, Dec. 1989.
[76] H. G. Musmann, M. Hotter, and J. Ostermann, "Object-oriented analysis-synthesis coding of moving images," *Signal Processing: Image Communication*, vol. 1, no. 2, pp. 117–138, Oct. 1989.
[77] K. Aizawa, H. Harashima, and T. Saito, "Model-based analysis synthesis image coding (MBASIC) system for a person's face," in *Signal Processing: Image Communication*,, vol. 1, no. 2, pp. 139–152, Oct. 1989.
[78] J. K. Vega-Riveros and K. Jabbour, "Review of motion analysis techniques," *Proc. Inst. Elect. Eng.*, Pt. I, vol. 136, no. 6, pp. 397–404, Dec. 1989.
[79] A. Mitiche and J. K. Aggarwal, "A computational analysis of time-varying images," in *Handbook of Pattern Recognition and Image Processing*, N. G. Einspruch, Ed. New York: Academic, ch. 13, pp. 311–332.
[80] D. Ellis and D. Rosenthal, "Mid-level representations for computational auditory scene analysis," presented at the *Int. Joint Conf. Artificial Intell.—Workshop Computational Auditory Scene Anal.*, Montreal, Quebec, Canada, Aug. 1995.
[81] R. Cox *et al.*, "New directions in subband coding," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 391–409, Feb. 1988.
[82] J. D. Johnston, "Transform coding of audio signals using perceptual noise criteria," *IEEE J. Select. Areas Commun.*, vol. 6, Feb. 1988.
[83] D. P. W. Ellis and B. L. Vercoe, "A perceptual representation of sound for auditory signal separation," in *Proc. 23rd Meeting Acoustical Society America*, Salt Lake City, Utah, May 1992.
[84] T. F. Quatieri and R. J. McAulay, "Speech transformations based on a sinusoidal representation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1449–1463, Dec. 1986.
[85] D. L. Thomson, "Parametric models of the magnitude/phase spectrum for harmonic speech coding," in *Proc. IEEE ICASSP*, 1988.
[86] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Trans. Robot. Automat.*, vol. RA-2, Mar. 1986.
[87] R. G. Crowder and R. K. Wagner, *The Psychology of Reading—An Introduction*, 2nd ed. London: Oxford Univ. Press, 1992.
[88] A. Baddeley, "Working memory," in *The Cognitive Neurosciences*, M. S. Gazzaniga, Ed. Cambridge, MA: MIT, 1995, ch. 47, pp. 755–764.
[89] G. A. Miller, "The magic number seven plus or minus two: Some limits on our capacity for information processing," *Psychological Rev.*, vol. 63, no. 2, pp. 81–96, 1956.
[90] D. E. Broadbent, *Decision and Stress.* New York: Academic, 1971.
[91] M. W. Eysenck and M. T. Keane, *Cognitive Psychology—A Students Handbook.* Hove, U.K.: Lawrence Erlbaum, 1990.
[92] H. G. Geissler, Ed., *Modern Issues in Perception.* New York: North-Holland, 1983.
[93] S. M. Kosslyn, *Image and Mind.* Cambridge, MA: Harvard Univ. Press, 1980.
[94] W. J. Rapaport, "Understanding understanding: Syntactic semantics and computational cognition," in *Philosophical Perspectives*, vol. 9, *AI, Connectionism and Philosophical Psychology*, J. E. Tomberlin, Ed. Atascadero, CA: Ridgeview, 1995, pp. 49–88.
[95] B. C. Smith, "The correspondence continuum," Center for the Study of Language and Information, Stanford, CA, Rep. CSLI-87-71, 1987.
[96] S. Stenström, *Optics and the Eye.* London: Butterworth, 1964.
[97] L. Levi, *Applied Optics.* New York: Wiley, 1980.
[98] L. A. Olzak and J. P. Thomas, "Seeing spatial patterns," in *Handbook of Perception and Human Performance*, vol. 1, K. R. Boff, L. Kaufman, and J. P. Thomas, Eds. New York: Wiley, 1986, ch. 7.
[99] A. B. Watson, "Temporal sensitivity," in *Handbook of Perception and Performance, Handbook of Perception and Human Performance*, vol. 1, K. R. Boff, L. Kaufman, and J. P. Thomas, Eds. New York: Wiley, 1986, ch. 6.
[100] G. Wyszecki and W. S. Stiles, *Color Science.* New York: Wiley, 1967.
[101] J. Larimer and T. Piantanida, "The impact of boundaries on color: Stabilized image studies," in *SPIE*, vol. 901, *Image Processing, Analysis, Measurement, and Quality*, Jan. 1988, pp. 241–247.
[102] V. Bruce and P. R. Green, *Visual Perception: Physiology, Psychology & Ecology*, 2nd ed. Hove, U.K.: Lawrence Erlbaum, 1990.
[103] D. H. Hubel and T. N. Wiesel, "Brain mechanisms of vision," *Sci. Amer.*, vol. 241, no. 3, pp. 130–145, Sept. 1979.
[104] R. Von Der Heydt, "Form analysis in visual cortex," in *The Cognitive Neurosciences.* Cambridge, MA: MIT, 1995, ch. 23.
[105] B. Scharf and S. Buus, "Audition I," in *Handbook of Perception and Human Performance*, vol. 1, K. R. Boff, L. Kaufman, and J. P. Thomas, Eds. New York: Wiley, 1986, ch. 14.
[106] B. C. J. Moore, *An Introduction to the Psychology of Hearing.* New York: Academic, 1989.
[107] N. Suga, "Processing of auditory information carried by species-specific complex sounds," in *The Cognitive Neurosciences.* Cambridge, MA: MIT, 1995, ch. 18.
[108] M. Konishi, "Neural mechanisms of auditory image formation," in *The Cognitive Neurosciences*, M. S. Gazzaniga, Ed. Cambridge, MA: MIT, 1995, ch. 16.
[109] S. Handel, *Listening.* Cambridge, MA: MIT, 1989.
[110] C. Morris, *Foundations of the Theory of Signs.* Chicago, IL: Univ. Chicago Press, 1938.
[111] ——, *Signification and Significance.* Cambridge, MA: MIT, 1964.
[112] W. Nöth, *Handbook of Semiotics.* Bloomington: Indiana Univ. Press, 1990.
[113] H. E. Fiske, *Music and Mind.* Lewiston, NY: Edwin Mellen, 1990.
[114] D. Cooke, *The Language of Music.* London: Oxford Univ. Press, 1959.
[115] R. Monelle, *Linguistics and Semiotics in Music.* New York: Harwood, 1992.
[116] M. Bense, *Zeichen und Design: Semiotische Ästhetic.* Baden-Baden: Agis, 1971.
[117] C. L. Carter, "Syntax in language and painting," *Structurist*, vol. 12, pp. 45–50, 1972.
[118] D. Marr, *Vision.* New York: Freeman, 1982.
[119] S. Worth, "The development of a semiotic of film," *Semiotica*, vol. 1, pp. 282–321.
[120] G. Davenport, T. A. Smith, and N. Pincever, "Cinematic primitives for multimedia," *IEEE Comput. Graph. Appl.*, pp. 67–74, July 1991.
[121] U. Eco, *Einführung in die Semiotik.* München: Fink, 1968.

[122] I. E. Gordon, *Theories of Visual Perception.* New York: Wiley, 1989.
[123] G. Kanizsa, "Subjective contours," *Sci. Amer.*, vol. 234, no. 4, Apr. 1976.
[124] K. Koffka, *Principles of Gestalt Psychology.* New York: Harcourt Brace Jovanovich, 1935.
[125] R. Gonzalez, "Software decodeable video for multimedia based on a computer graphics model," Ph.D. dissertation, University of Technology, Sydney, Australia, 1994.
[126] R. Gonzalez and A. Ginige, "A video coding scheme for interactive multimedia based on a computer graphics model," in *Proc. 1st Int. Conf. Multimedia Modeling*, Singapore, Nov. 9–12, 1993, pp. 177–191.

Ruben Gonzalez (Member, IEEE) received the B.E.(Hons.) and Ph.D. degrees in electrical engineering from the University of Technology, Sydney.

He was involved in software engineering for commercial image processing systems and also was with the multimedia group of OTC's research laboratories and a number of commercial software and hardware engineering positions. His research interests include video and audio coding, hypermedia systems, computer graphics, content-based retrieval, signal processing, and mobile multimedia terminals.