Journal of Electronic Imaging 5(2), 176–181 (April 1996).

Design and development issues for multimedia information systems

Young Francis Day
Edward J. Delp
Purdue University
School of Electrical and Computer Engineering
Distributed Multimedia Systems Laboratory
West Lafayette, Indiana 47907

Mistsutoshi Iino
Fuji Electric Company
1, Fuji-Machi, Hino-Shi
Tokyo 191, Japan

Azam Khan
Arif Ghafoor
Purdue University
School of Electrical and Computer Engineering
Distributed Multimedia Systems Laboratory
West Lafayette, Indiana 47907

Abstract. *The organization and management of multimedia information has been the subject of extensive research and development since the mid-1980s. However, a number of issues need to be addressed prior to developing a general purpose multimedia information system. We discuss an abstract architecture of such a system and highlight the issues of data modeling and multimedia information authoring, and also provide a systematic argument in favor of using an object-oriented approach over the relational approach as the basic framework for developing a multimedia information system.*

1 Introduction

Multimedia is emerging as a new information technology that allows users to store, retrieve, share, and manipulate complex types of information. A variety of fields including business, manufacturing, education, computer-aided design (CAD) and computer-aided engineering (CAE), medicine, weather, entertainment, etc., are expected to benefit from this technology. These applications require the use of multimedia data consisting of images, pictures, audio segments, full motion video, and text. Depending upon the application, such data has differing quality requirements. For example, in medical information systems, electronic images (x-rays, MRIs, etc.) and sonograms may require high-resolution storage and display systems. Due to the diverse nature of the multimedia data, systems designed to store, transport, display, and, in general, manage such data must have considerably more functionalities and capabilities than conventional information management systems. In general, multimedia technology requires substantial changes in the approaches used in designing conventional databases and storage systems.

In multimedia information systems, monomedia may represent individual data entities that serve as components of some multimedia object such as electronic documents, medical records containing electronic images and sonograms, etc. These monomedia objects can be grouped together for efficient management and access. It is essential that the user should be able to identify and address different objects and be able to compose them in time and space. The composition should be be based on a model that is visually presentable to the user. It is, therefore, desirable that a general framework for spatio-temporal should be available for composing and storing multimedia documents.

Recently, some attempts have been made to develop conceptual models for multimedia information. However, most of these models are primarily aimed at synchronization aspects of the multimedia data while others are more concerned with the browsing aspects of information. A few of them, which are based on graphs, have the additional advantage of pictorially illustrating synchronization and spatial semantics, and are suitable for visual orchestration of multimedia presentations. Irrespective of the media type, it is imperative that a model should render itself efficiently to an ultimate specification of the database schema and it must be integrated with the underlying data models of various monomedia. It must also render itself effectively for developing higher level user interfaces.

Paper MMS-06 received Dec. 23, 1995; accepted for publication Apr. 5, 1996.
1017-9909/96/$6.00

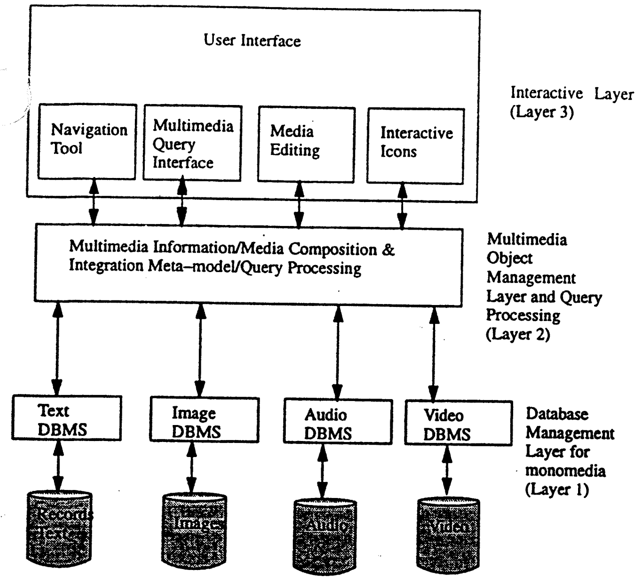

Fig. 1 Abstraction for a multimedia database management system.

Based on this discussion we develop a conceptual architecture for a general purpose multimedia information system as shown in Fig. 1. The architecture consists of three layers: a monomedia database management layer, an object management layer, and a user interface layer.

The monomedia database management layer can provide the functionalities essential for managing individual media including formatted data (text and numeric) and unformatted data (audio, video, images). One of the key aspects of each database at this level is to maintain efficient indexing mechanism(s) and to allow users to develop semantic-based modeling and grouping of complex information associated with each media. The primary objective is to process content-based queries and facilitate retrieval of appropriate pieces of monomedia data, such as a video clips, parts of an image, audio segments, etc. A major consideration at the time of retrieval is the quality of information that can be sustained by the system, both at the database site and the user site. Therefore, it is important that some quality of presentation (QoP) parameters, such as speed, resolution, delay bounds, etc., should be specified by the user and maintained by the system at this layer.

The middle layer can provide the functionality of integration of monomedia for composing multimedia documents as well as integrating/cross-linking information stored across monomedia databases. Integration of media can span multiple dimensions including space, time, and logical abstractions (such as hypermedia or object-oriented). Therefore, the primary function of this layer is to maintain some metaschema for media integration along with some unconventional information, e.g., the QoP parameters discussed above. The objective is to allow efficient searching and retrieval of multimedia information with the bounds of desired quality. Since there is a growing need for management of multimedia documents and libraries, the need for efficient integration models is becoming one of the key research issues in developing a general-purpose multimedia database management system (DBMS).

The interactive layer consists of various user interface facilities that can support graphics and other multimedia interface functionalities. Various database querying and browsing capabilities are provided in this layer.

In the next section we discuss the QoP issues pertaining to all the layers of the architecture of Fig. 1. The detailed discussion of all the layers is the theme of the rest of the paper.

2 Quality Requirements for Multimedia Information

The QoP parameters constitute a set of the user-specified tolerable degradations of multimedia presentation, which can occur due to resource limitations. These parameters are used to quantify the presentation process from the user's point of view and can be used to specify the resources required to present multimedia information with a certain QoP. In the following sections we identify some of the user-perceivable presentation parameters that influence the play-out quality of multimedia information.

2.1 *Reliability*

Isochronous objects, such as video and audio, can tolerate some information loss without affecting the play-out quality. These tolerance levels depend on the application and the media type used in that application, and hence can be chosen as a QoP parameter. For a user, *reliability* can describe the bound for acceptable delivery for a multimedia object type due to resource constraints. An option to overcome this limitation is to pre-fetch enough data. However, due to the bounds on buffer availability this may not be possible for all the multimedia objects for every user. Another factor that can affect reliability is early arrival of data at a destination, which may result in buffer overflow. Depending on the required reliability, at the time of authoring the document, a user can quantify the acceptable data loss for each object due to buffer overflow by specifying *maximum allowable percentage data loss due to buffer overflow* for each object O_i of the document. The resolution and presentation rate of a multimedia object can be used to determine which and how much data to drop, in case of limited resources. For applications requiring a lower resolution and/or presentation rate, we can afford to drop a fraction of the object at the source. For example, if the user can tolerate a presentation rate of 20 frames per second for a video object (instead of 30 frames per second), one in every three frames can be dropped without degrading the required play-out quality beyond the acceptable level. This data dropping can be specified by the user in the form of a ratio called *presentation rate ratio* and is represented by θ_i for an object O_i. Formally, the ratio θ_i is defined for isochronous objects (video and audio) as the ratio of the required rate of presentation to the nominal one. In case of anisochronous objects such as text and images, the ratio θ_i is defined on resolution instead of rate of presentation.

2.2 *Skew*

A strict presentation requires that multimedia objects be played out with zero slippage. This can be achieved through interstream synchronization, which necessitates all objects to be delivered prior to their deadlines. Concurrent objects may be delivered independently, and hence may experience different delays, causing some of them to miss their temporal deadlines. Objects that missed their deadlines start lagging while data on other streams are continuously consumed during presentation. We refer to such a synchronization failure as *deadline miss*. The acceptable delay for an object O_i with respect to its deadline π_i is denoted by α_i. The value of α_i depends on the object type and the application. The maximum relative skew between two concurrent objects O_i and O_j is $\max[\alpha_i,\alpha_j]$. We can express the acceptable level of presentation degradation due to skew in terms of the maximum tolerable percentage of deadline misses by a factor greater than the delay α_i. We can formally characterize the maximum tolerable deadline miss for an object probabilistically as the maximum tolerable probability with which each presentation unit of object O_i can miss its deadline by more than a delay α_i.

3 Development Issues for Multimedia Information Systems

The issue of computation, composition, and QoP is prevalent in multimedia data. Development of a general-purpose multimedia information system (such as the one shown in Fig. 1) is a challenging problem. It is nontrivial to use a unified family of schema for managing the bottom two layers of Fig. 1, as the domain for a given multimedia application can be quite complex and ill-structured. Moreover, the metaschema may also incorporate some unconventional information, e.g., the quality of service (QoP) parameters for presentation, as discussed in the previous section. Therefore, it is essential to scrutinize the possible implementation approaches, which can be classified into two categories:

- using the conventional data model such as relational, which has been used for the last fifteen years
- semantic-based object-oriented solution, which will prevail throughout this decade and onward.

The conventional data modeling techniques lack the ability of managing semantics of multimedia objects in a heterogeneous multimedia database environment. Semantic-based integration of different types of data may be required for a large number of multimedia applications. The relational model has a drawback of losing semantics, which can cause erroneous interpretation of multimedia data. Limitation of this model becomes obvious when we deal with the issue of semantic modeling of unstructured monomedia data (layer 1 of Fig. 1), such as video, audio, or images. This is discussed in detail in the following section.

3.1 *Data Modeling and Management of Monomedia: Layer 1*

The key characteristic of monomedia data (especially the video data) is its spatial/temporal dimensions, which makes these data very different from the conventional textual/alphanumeric data. A user can generate queries containing both temporal and spatial semantics while accessing video data. Moreover, considerable semantic heterogeneity may exist among users of such data due to differences in their preconceived interpretations or intended use of the information in the database. For example, for video/image data, Fig. 2(b) describes abstraction hierarchies in space and time that need to be generated, in order to develop content-based indexing mechanism for a video/image database management system. Such hierarchical abstractions may result in considerable semantic heterogeneity, which has been a difficult problem for conventional databases, and even today this problem is not clearly understood. Consequently, providing a comprehensive interpretation of such data is a complex problem.

However, the semantic-based object-oriented models can provide a richer set of abstractions that are particularly useful for the users to extract/define views of information at various levels of these hierarchies. These models can help users in developing their own custom-tailored access tools without misinterpreting the underlying metalevel concepts. Through such an approach, the real meaning of attributes can be captured using common concepts and thus provides a highly interactive environment.

From the abstractions in Fig. 2(b) we can extract the necessary design specifications for the development of an automated video information management system, as elaborated in Fig. 2(a). The figure highlights the complex integration of various diverse technologies including computer vision and image processing (CVIP),[1,2] AI, and database management systems. The generation of content-based indices requires a broad spectrum of computations that are intermingled with data and entities at various levels.

Most of the existing multimedia database systems (especially video and image) either employ limited image processing techniques for semiautomated indexing of data or use manual approaches to generate keywords or annotated textual descriptors. On the other hand, in semiautomated image processing based systems, special emphasis is given to content-based analysis of the data. Due to the shear volume of the data and desired capabilities of multimedia database systems such as automatic indexing and abstraction of video data, efficient computer vision/image processing algorithms need to be employed both at the preprocessing stage and during on-line query processing. Furthermore, powerful data retrieval techniques are needed to support on-line query processing. An automated video database system requires an effective and robust recognition of objects present in the video database. Due to the diverse nature of the video data, we can use various techniques currently available according to the requirements of different contents that may occur in a given video. For each input video clip, using a database of known objects, first we need to identify the corresponding objects, their sizes and locations, their relative positions and movements, and then encode this information using the proposed model. This encoded information can be used to answer on-line queries. Also, in case the video/image data are needed in a multimedia document, the encoded information can be integrated directly by the composition schema maintained by the middle layer of Fig. 1. Integration and management of heterogeneous computation with data entities at these levels would be clumsy if we attempt to use a relational model. As mentioned earlier, the data/computation encapsulation feature of object-oriented technology would be more suitable for such a purpose.

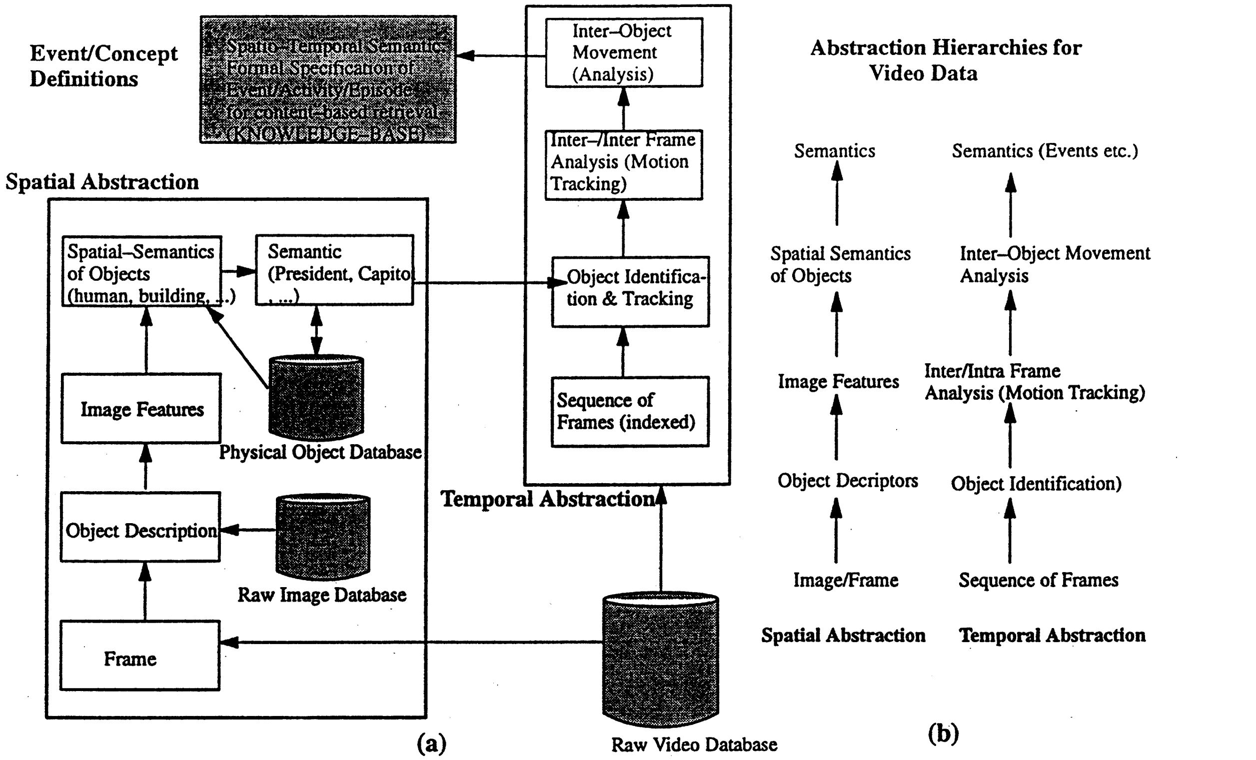

Fig. 2 An architecture for an automated video/image database management system for on-line query processing.

The object-oriented technology, on the other hand, can provide a powerful paradigm to meet the requirements of multimedia composition and developing semantic models. Its data and computational encapsulation features offer elegant semantic modeling capabilities at various levels of information granularity in multimedia database systems.

3.2 *Integrated Composition Model for Multimedia Information: Layer 2*

Presentation of preorchestrated and composed multimedia information requires synchronous playout of time-dependent multimedia data according to some prespecified temporal relations and QoP requirements, as discussed in Section 2. Such functionality is provided by the middle layer of Fig. 1. At the time of creation of multimedia information, a user needs a model to specify temporal constraints among various data types and the QoP requirements that must be observed at the time of playback for each object. Examples of models that specify the temporal relationships among the multimedia objects are object composition petri-net (OCPN)[3,4] and hypermedia/time-based document structuring language (HyTime). In this paper we concentrate on the OCPN model, since with a little modification it can also include QoP parameters for each object type along with their temporal relationships. Also, the major advantage of using the OCPN model is its graphical appeal to the user. The model can provide a spatio-temporal composition process visually on the screen.

A place in the modified OCPN represents the play-out process of multimedia object O_i, which may be textual data, image, or a video/audio segment. The play-out deadline π_i of each object O_i is the time instance at which the playout of that object is scheduled to start. Quality attributes associated with an object can include its type (continuous/discrete media), size s_i, duration of its presentation τ_i, allowable skew α_i, probability of deadline misses, probability of buffer overflow, presentation rate ratio θ_i, etc., as depicted in Fig. 3. These attributes are given

Parameter	Symbol
duration	τ
size	s
deadline	π
allowable skew	α
display area	A
rate	r
reliablity	ω
contents	

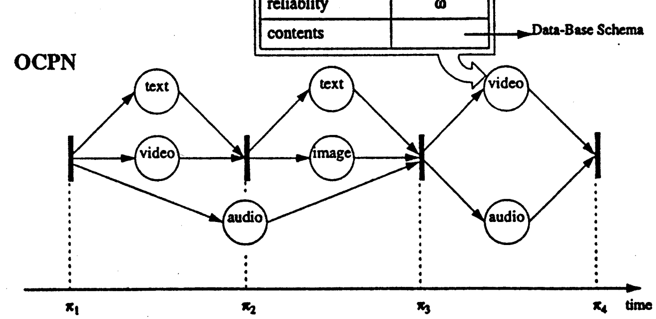

Fig. 3 An example OCPN with QoP parameters.

by the user at the time of creation of the multimedia document and incorporated in the OCPN model. A transition in the OCPN represents a synchronization point as it marks the play-out start time or deadline of new concurrent objects. At the time of presentation, the OCPN structure is evaluated and objects associated with places are retrieved and communicated to the end user. Due to multiplicity of concurrent objects (e.g., video and audio) not only intrastream but interstream synchronization is necessary.

Anisochronous data (e.g., text and images) need to be available at the play-out device at the destination before the play-out deadline specified by the OCPN. On the other hand, isochronous objects, such as video and audio, can be transmitted at the same rate as the play-out rate.

From the development point of view, the object-oriented technology can provide a powerful paradigm to implement multimedia composition models and allow semantic modeling of multimedia documents.[5,6]

3.3 *Query and User Interface: Layer 3*

The query language of a database system is strongly related to the underlying data model. For multimedia applications, a highly interactive environment is needed that allows content-based search, browsing,[11] or partial match retrieval. For content-based retrieval, a query in its simplest form can be based on keywords/descriptors or some formal spatio-temporal expression. The other extreme is to allow the user to sketch images or to draw examples of spatio-temporal events. Such query-by-visual-example (QVE)[7–10] capability places stringent requirements on the design of the user interface. For this purpose a graphical interface or visual language is more useful and friendly than the conventional textual query language, such as structured query language (SQL), since the former can allow representation of objects by images or icons and it is easy to use.

As we expect, object-oriented schemes may be used as a toolkit for developing such interfaces. The integration of these components, data definition languages, and database programming languages pose major challenges in designing the query language of the system.

Some progress has been made in query languages recently. Most of the proposals are centered around extending existing languages, such as SQL, toward multimedia applications. However, incorporating content-based search and using these languages in a programming environment is a major issue to be addressed by the database community.

In our view, using some graphical query interface is the most appropriate and user-friendly approach. For this purpose, the OCPN model, shown in Fig. 2, is a natural choice, since to the end user it can provide all the important composition and QoP information in a comprehensive manner.

Figure 4 shows a graphical user interface for interactively creating an OCPN-based structure of an arbitrary multimedia document. The interface provides a set of icons for building an OCPN with all the desired QoP parameters, as depicted in Fig. 3.

4 Discussion on Data Models

Integration of monomedia is a key issue for developing a general-purpose multimedia information system. It is still being debated in the database community whether or not integration problems of even textual databases can be solved through schema integration alone. A family of solutions using the notion of external and internal schema has been proposed. This approach has its own strengths and weaknesses. Although some relational database systems have started supporting access to multimedia objects in the form of pointing to binary large objects (BLOBs), there is no provision for interactively accessing various portion of objects, as a BLOB is treated as a single entity in its entirety. A somewhat related problem is that the relational model does not provide an elegant mechanism for maintaining a complex logical structure of multimedia information/documents. As argued above, the object-oriented paradigm can be a powerful candidate for developing most of the functionalities of Fig. 1.

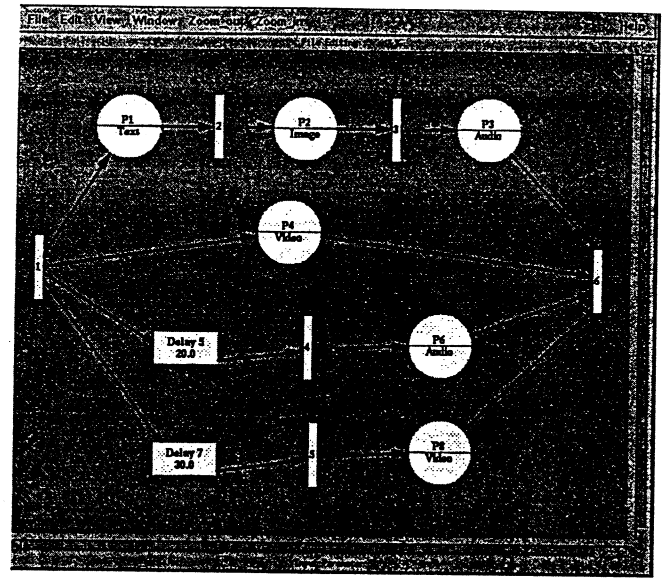

Fig. 4 A Graphical view of document querying/authoring process.

Abandoning the relational model altogether for the implementation of a multimedia information system can cause a serious dilemma, as extensive management of a large number of indices, the metaschema, and physical databases (e.g., shown in Fig. 1), as well as interfacing with the operating system for real-time retrieval, can be technologically challenging. Since most of the management data itself is textual in nature, it can be efficiently managed using the traditional relational database technology. However, if we insist on using the relational technology as the bases of the whole system, the capabilities of the relational model need to be expanded substantially, in order to overcome the problem of mismatch between the relational and object-oriented technologies.[6]

In summary, most of the existing multimedia database approaches lack the ability to provide a general-purpose automatic mechanism that can render itself to semantic-based description of data. In order to address the issues related to a user-independent view and semantic heterogeneity, a *semantically unbiased abstraction* approach would be desirable, which can allow representation of spatio-temporal information associated with objects (persons, buildings, vehicles, etc.) at some *basic level*. However, such a technique would require effective and robust CVIP algorithms. Such a basic but unified framework can provide a good flexibility to the users to express and for the system to process semantically heterogeneous queries on the unbiased encoded data. This modeling approach can provide an efficient indexing mechanism for on-line query processing without performing computations on the raw video data since such computation can be quite extensive.

5 Conclusion

One of the main requirements of multimedia information systems is that they will need data models more powerful and more versatile than the relational model, without compromising the advantages of the former. The relational data model exhibits limitations in terms of complex object management, indexing, and content-based retrieval of video/image data, and the facility for handling the spatiotemporal dimensions of objects. To address these issues, we have emphasized two key requirements for multimedia databases: the process of spatiotemporal modeling and the computational needs for automatic indexing of spatiotemporal data. We have highlighted various other challenges that need to be tackled before multimedia information management systems become a reality.

Acknowledgment

The research presented in this paper has been funded by the Fuji Electric Company Ltd. of Japan.

References

1. A. Nagasaka and Y. Tanaka, "Automatic video indexing and full video search for object appearances," *in 2nd Working Conference on Visual Database Systems*, Budapest, Hungry, IFIP WG 2.6, pp. 119–133 (October 1991).
2. *IEEE Computer*, special issue on "Content-based image retrieval systems," **28** (9) (Sep. 1995).
3. T. D. C. Little and A. Ghafoor, "Synchronization and storage models for multimedia objects," *IEEE J. Selected Areas Communications* **8**(3), 413–427 (Apr. 1990).
4. T. D. C. Little and A. Ghafoor, "Interval-based conceptual models for time-dependent multimedia data," *IEEE Trans. Knowledge and Data Engineering* **5**(4), 551–563 (Aug. 1993).
5. E. Oomoto and K. Tanaka, "OVID: design and implementation of a video-object database ssystem," *IEEE Trans. Knowledge and Data Engineering* **5**(4), 629–643 (Aug. 1993).
6. Y. Masunaga, "Design issues of OMEGA: an object-oriented multimedia database management system," *J. Information Processing* **14**(1), 60–74 (1991).
7. K. Hirata and T. Kato, "Query by visual example," *Advances in Database Technology EDBT '92*, pp. 56–71, Springer-Verlag (1992).
8. S.-K. Chang, C.-M. Lee, and C. R. Dow, "Two-dimensional string matching algorithm for conceptual pictorial queries," *Proc. SPIE* **1662**, 47–58 (1992).
9. K. Hirata and T. Kato, "Query by visual example," *Advances in Database Technology EDBT '92*, pp. 56–71 (1992).
10. A. Del Bimbo, M. Campanai, and P. Nesi, "3-D visual query language for image databases," *J. Visual Languages and Computing*, **3**, 257–271 (1992).
11. P. D. Stotts and R. Furuta, "Petri-net-based hypertext: document structure with browsing semantics," *ACM Trans. Office Automation Systems*, **7**(1), 3–29 (Jan. 1989).

Young Francis Day received the B.S. degree in computer science and information engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1989, and M.S. degree in computer engineering from the Pennsylvania State University, University Park, Pennsylvania, in 1991. Currently he is a doctoral student and a research assistant in the School of Electrical and Computer Engineering at Purdue University, West Lafayette, Indiana. His area of research is multimedia database management systems, with focus on multimedia document modeling, video data modeling, and multimedia query language using object-oriented paradigm.

Edward J. Delp: Biography and photograph appear with the special section guest editorial in this issue.

Mistsutoshi Iino received his MS from Waseda University in 1983. Currently, he is an associate manager of the multimedia application development group at Fuji Electric Co., Ltd., in Japan. His research interests include multimedia databases, multimedia data transfer protocol, and multimedia applications for industrial computing.

Azam Khan is working as a consultant with the World Bank in Washington, D.C. He obtained both his MS in Civil Engineering and an MBA from Purdue University. In the past, he has worked on various projects in the areas of process reengineering, project management, and information systems.

Arif Ghafoor received his BS degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 1976, and his MS, MPhil, and PhD from Columbia University in 1977, 1980, and 1985, respectively. He joined the faculty of the Purdue University School of Electrical Engineering in 1991, where he is currently an associate professor and directs a research lab in distributed multimedia systems. Prior to joining Purdue University, he was on the faculty of Syracuse University since 1984. His research interests include design and analysis of distributed multimedia systems and broadband networking technology. His research in these areas has been funded by many industrial and government agencies. He has served on the program committees of various conferences. Currently, he is serving on the editorial boards of *Multimedia Systems Journal*, the *Journal of Multimedia Tools and Applications* and the *Journal of Parallel and Distributed Databases*. He served as a guest/co-guest editor of special issues of various journals including the *Journal on Parallel and Distributed Computing, Multimedia Systems Journal*, the *IEEE Journal on Selected Areas in Communication* and the *Journal of Multimedia Tools and Applications*. These issues are on various aspects of multimedia systems and technology. He is consultant to many industrial and government organizations.