Proceedings, IEEE Visual Signal Processing and Communications Workshop, New Brunswick, NJ, September 1994.

Development of Advanced Image/Video Servers in A Video on Demand Testbed

Shih-Fu Chang, Dimitris Anastassiou, Alexandros Eleftheriadis, Jianhao Meng, Seungyup Paek, Sassan Pejhan, and John R. Smith

Department of Electrical Engineering & Center for Telecommunications Research
Columbia University, New York, NY 10027

Abstract

This paper describes our current effort in building a video server with advanced capabilities of storage/indexing/ retrieval. The target application at this stage is Columbia University's Video on Demand (VOD) testbed. Our research focus covers innovative content-based visual query, multi-resolution (MR) video coding, efficient manipulation of compressed video, and optimal real-time video storage architectures. An important aim of our research is to enhance interoperability. Heterogeneous communication connections (ATM and Internet) have been installed. We discuss research progress and design strategies in various technical areas of this project.

1. Introduction

At the Image and Advanced Television laboratory of Columbia University, we have started a project to develop and implement an image/video server (in short, *VS*) with advanced features of image representation, storage, query, and retrieval. The main objective is to use this prototype as a platform for the state-of-the-art multimedia research and application development. Among the potential applications are the VOD service, digital libraries, and interactive multimedia systems.

Designing a full-function VS for general multimedia applications requires extensive interdisciplinary knowledge and skills. Several research groups have reported progress in various aspects. Multi-resolution representations for image databases were studied in [13]. Innovative methods for indexing/searching images by image contents were proposed in [2, 14, 3]. Dedicated storage architectures for real-time multi-access VS have been studied in [7,8,9,15, 10]. System-level studies are presented in [12]. Systematic approaches to the design of VS are being undertaken in [16, 17, 18, 1] as well. In addition, many field trials of VOD services using proprietary high-performance VS technologies have made news headlines recently. Lastly, a major international forum called DAVIC has been established to come up with timely recommendations for critical protocols and interfaces for achieving *interoperability* between various audio-visual applications [19].

With the VOD system as the current driving application, we are focusing on the following R & D areas —

- Multi-resolution image/video coding and efficient compressed bitstream manipulation algorithms.
- Innovative methods for image/video feature extraction and indexing, allowing advanced mechanisms of image search and retrieval, such as *content-based visual query.*
- Optimization techniques for single-disk and multi-disk storage architecture design, allowing multi-user real-time access.
- Interactive navigation tools allowing effective search and browsing of visual data.
- Quality of Service (QoS) negotiation and guarantee in the heterogeneous network environment.
- Synchronization between different types of media (video, audio, captions, etc.)

All these issues are being addressed in our VOD testbed. We describe our VOD system architecture in the next section. We present research progress and design strategies in various technical areas in Section 3 to Section 5.

2. Columbia's VOD Testbed

The underlying computing and communication infrastructure of Columbia's VOD testbed at this point includes a graphic supercomputer as the server and the campus-wide heterogeneous network connections (ATM and Internet) among several departments and schools in the university, as well as outside institutions.

Figure 1 shows the testbed system architecture. The supercomputer used as the server is equipped with high-end computing power and 3D graphic capability, which are needed in many interactive multimedia applications and real-time video manipulations. Dedicated disk array secondary storage is connected to the server, while local storage systems are available in end user workstations. Many user-site workstations are also equipped with high-end graphic hardware which can be used to enhance the video playback performance. Extension to settop-based and PC-based user terminals are included in the near-future plan.

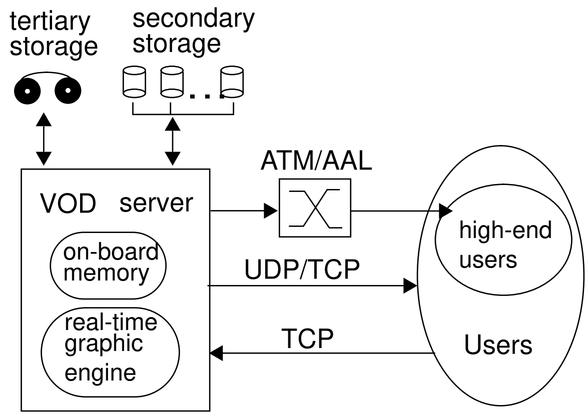

FIGURE 1. A VOD testbed architecture.

A suite of client-server communication protocols and interfaces have been developed. Downstream channels towards the clients use TCP/IP protocol stacks for transmitting delay-insensitive data and UDP/IP for isochronous video streams over Internet. ATM/AAL protocols are used for delivering real-time video over the ATM network. The reverse channel uses TCP/IP for delivering control data back to the server such as browse, retrieval, and query commands. In some cases, we also use the RPC function of UNIX to transfer short control messages for efficiency.

We have developed flexible interface facilities for digitization and encoding of visual data from various sources (e.g., live camera, VCR, and LD player). Columbia's standard-conforming full-function MPEG-1/MPEG-2 software decoder is utilized. We have also implemented software MPEG decoder and playback routines with VCR interactive control functions. The decoder and playback routines are included as resident programs on the client side.

Audio-video synchronization is accomplished by two different approaches. The default mode is to playback audio in the real time while displaying video with the best effort, i.e., skipping video frames (e.g., B frames of MPEG) whenever video playback falls behind. The other approach tries to play back every video frame as fast as possible, while making the best effort in keeping the audio track synchronized with video. Typically, this requires intelligent algorithms to adjust the sampling rate of the audio signal without losing its pitch information [20].

We have completed the first prototype of our VOD testbed. User evaluations are being undertaken within and outside the laboratory. Various optimization techniques for performance enhancement and new research approaches will be incorporated in the future generations. We also intend to use this testbed to participate in the upcoming DAVIC interoperability events.

3. Video Compression & Manipulation

3.1 Multi-Resolution Video Coding

Multi-resolution video coding is adopted in order to accommodate heterogeneous QoS requirements, e.g., different channel bandwidth or different display resolutions. Different multi-resolution coding methods, such as MPEG-2 high-profile scalable coding options, subband coding, and spatio-temporal pyramid, are being examined and analyzed in the context of VOD.

Currently, we use the high-profile spatial scalability feature of Columbia's MPEG-2 software to provide two different display sizes for each video sequence stored in the server [23]. Users equipped with high-speed connections (such as ATM) and high-end processing/display resources can request both the base stream and the enhancement stream to reconstruct the full-size video, while users with low-end resources retrieve the base stream only and view the same video with a smaller size. However, our experiments indicate that the final video quality is typically limited by the decoder speed at the client. Software implementations of the MPEG-2 decoder on today's moderate-range workstations (like SUN Sparc2 or SGI Indigo2) still cannot decode and playback a 1 Mbps MPEG-2 compressed video stream in the real time. We are currently exploring ways to alleviate this bottleneck but we hope that the situation will be improved very soon by the rapidly advancing hardware technology.

MPEG-2 scalability can provide multiple resolutions up to 3 levels, which may be sufficient for some applications. However, in a heterogeneous environment supporting different types of network links, user terminals, and user preference, more resolution levels are needed. Also, very often users may want to view the same visual data at different resolutions (e.g., incremental query and hierarchical retrieval), or view different image regions with different resolutions (e.g., zoom in). A greater multiplicity of resolutions is desirable in order to make these interactive manipulations feasible and efficient.

One promising technique to create multiplicity of resolutions (or bit rates) from pre-encoded video bitstreams is to insert an intermediate process to manipulate the encoded bitstream in such a way that output of this intermediate process will conform to the desired requirements. Specifically, the Dynamic Rate Shaping (DRS) technique that we are developing at Columbia University will be used to adjust the bit rate of the retrieved video [11]. This process can be implemented either in the server or in some third-party locations which conceivably can retrieve compressed video from various distributed servers and add certain service values such as rate conversion, special effects, etc. One crucial objective of the DRS process is to obtain as high video quality as possible at the minimum computational cost.

3.2 Compressed Video Manipulation

Rate conversion is one of many functions that may need to be applied on the retrieved compressed streams before transmission. Among others are format conversion (*video transcoding*), image enhancement, zooming, geometrical transformation, and image warping. We are exploring potential innovative algorithms to achieve these functions directly in the compressed domain in order to minimize the incurred computational cost.

The benefits for manipulating video in the compressed domain are twofold. First, the data rate in the compressed domain is much less than that in the uncompressed domain and thus the required computational cost is lower. Second, for the cases in which both input and output are compressed, decoding and re-encoding of the video stream can be avoided. Thus, the overall computational cost can be further reduced.

Most existing image/video compression standards include the transform coding, such as the Discrete Cosine Transform (DCT). We have successfully derived a set of algorithms for manipulating transform-encoded images directly in the DCT domain [21]. Functions like geometrical transformation, linear filtering, and image compositing can be efficiently implemented in the transform domain. In other words, the transform coefficients of the retrieved video can be directly processed without being decoded first. We have also shown these techniques can be generalized to all kinds of orthogonal transform coding techniques including Discrete Fourier Transform and Discrete Sine Transform. The actual computational speedup by using the transform-domain approach depends on the compression rate of the input video. Some manipulation functions (such as block-based operations) benefit more from the transform-domain approach than others due to their compatibility with the block structure of typical transform algorithms.

For many video compression standards (such as MPEG and H.261) based on Motion Compensation (MC) as well as the transform coding algorithm, we have proposed one transform-domain conversion technique to convert the motion compensated video back to the transform domain, in which the derived transform-domain manipulation algorithms can be applied. Again, the computational complexity of the transform-domain approach depends on the compression rate and the motion vector distribution of the input video.

4. Content-Based Visual Query

Another important research focus of our VS project is to explore new ways of image/video indexing and query by visual contents. Most existing approaches to image indexing and retrieval use the textual keyword, which naturally lends itself to the usage of conventional textual-based query. Search and retrieval are performed on the keyword records and the associated images are retrieved after matches are found. Some image databases provide enhancement by supporting query by pictorial examples of pre-determined visual objects, such as mechanic design diagrams, electronic schema, and office designs [22]. In some cases, semantic-level descriptions (such as objects in the picture, relationships among objects, and actions associated with objects) are provided by users and used as index of the visual data.

All the above approaches rely on some forms of manual inputs from users. It will become difficult to use this manual approach to index a huge amount of visual data in the video server. Also, it is difficult to obtain consistent and complete subjective descriptions of visual data. To overcome this problem, we are investigating innovative approaches for automatic indexing and searching of visual data by generic signal features such as object shape, texture, color, motion, video scene, etc. We consider this content-based approach complimentary with existing user-assisted keyword-based approach and semantic-level approach. Only by providing a rich multiplicity of interfaces towards indexing and retrieving visual data can users efficiently search through thousands to millions of pictures in the server.

Figure 2 shows the role of the content-based indexing/query engine in the VS. Users interact with the server through the Query/Presentation interface, which produces the textual or visual keys to the Query Engine to search for the intended objects. The visual features and textual indexes are stored in the Meta-data Index File which is kept separate from the actual compressed image/video data. Logical links and pointers are created to bind related objects. The query results will be forwarded to the Retrieval Scheduler which in turn schedules the actual retrieval of the returned data, through the management of the Storage Controller.

4.1 Texture-Based Visual Query

One crucial task for content-based visual query is for computers to evaluate the content similarity between different images and videos. As a first step to compare the image content discrimination capabilities, we have extracted *texture* feature sets from several image representation schemes [3,4]. In the context of image/video server, data compression is also one of the main objectives in designing image representation schemes. Compression schemes considered so far include DCT transform coding, uniform subband coding, and wavelet subband coding. Using the full Brodatz collection of 112 texture classes, performance of texture content classification is analyzed and compared [5]. The effects of various parameters, such as the training class size and the size of feature set, are also examined.

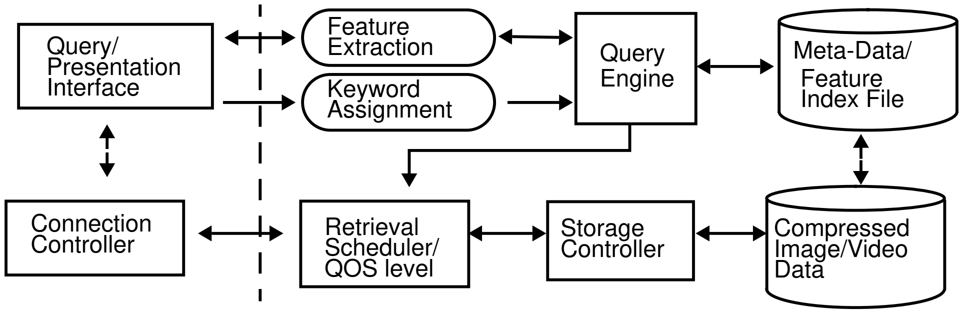

FIGURE 2. system organization of the VS in our VOD testbed

We are also investigating strategies for applying our optimized texture classification techniques to search for arbitrarily-shaped image segments in each frame of the video sequences. Based on a modified quad-tree segmentation technique, texture feature sets of each image block are extracted and adaptive conditions for merging neighboring blocks are determined by a texture discriminant function. Image regions with similar texture features are grouped together and indexed by the corresponding texture feature vector. So far, our preliminary experiment results show this texture-based video object search scheme promising. Given an arbitrary-shaped image, relevant texture features and information about the minimal image size with consistent texture features are extracted. This information is input to the query engine for finding all images or video sequences with similar texture patterns.

One of our immediate goals for feature-based query is that users can select any arbitrarily-shaped image segments (or 3D video objects) by editing displayed image examples or using interactive picture synthesis tools. The selected object of interest can be used as the search key and relevant features can be analyzed and extracted from this search key. In return, the visual query system should be able to retrieve all images which contains at least one area with features similar to those of the key image segment. This scenario is illustrated in Figure 3.

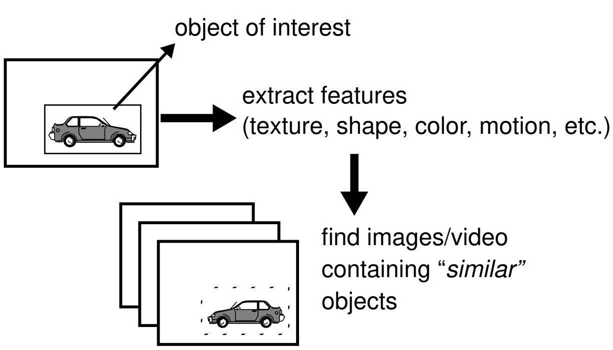

FIGURE 3. Feature-based object search in visual query systems.

4.2 Scene-Based Video Analysis and Indexing

In the case of video indexing, we take a divide-and-conquer approach which segments the entire video sequence into individual scene cuts. We assume each video scene contains consistent contents during the entire scene period. Therefore, we can index each individual scene with its representative features, such as object shape, texture, motion, and relations among objects. In order to explore the maximum synergy among compression and feature extraction, we are pursuing image analysis directly in the compressed domain. One example is to perform video scene change detection directly in the compressed domain, with minimal decoding of the compressed bitstreams (i.e. MPEG-2 encoded bitstreams) [6]. Our approach is to use the distribution of motion vectors and DCT coefficients of the motion compensated residual errors in MPEG-2 compressed bitstreams as cues in detecting dissimilarity between image frames. Compared to the traditional approach, the detection is performed with only a partial decoding of the compressed bit stream. A full decode of the compressed bitstream is not necessary and therefore computation time can be saved. We have tested our algorithms on synthetic sequences and real sequences. We have been able to catch most scene changes successfully, except the *dissolve* technique (i.e. fade in/fade out) which is often used in classical movies. We are trying to solve this problem by examining the statistics of a longer series of image frames, instead of individual frames only. Another possible technique is to take advantage of the special property of typical dissolve cuts, e.g., significant reduction of the illumination level during a dissolve cut.

4.3 Interactive User Interface

In a video server containing huge image/video collections, an effective interactive user interface must provide the capability for users to browse through retrieved images and video sequences at different scales (spatially or temporally). This multi-scale capability can be achieved by using hierarchical image/video coding algorithms mentioned earlier. Through the interactive tools, users can also manipulate the retrieved images to meet their specific needs. Envisioned image manipulations include geometrical transformation (zooming, rotation, etc.), image quality enhancement (for rare preserved document images or medical images), halftoning (for converting continuous-tone images to halftone displays or printers), color space conversion (to accommodate displays with different color depth), and multiple image objects compositing, among others

For video, a video *scene browser* is useful summarizing the contents of the retrieved video sequence. For example, users can quickly browse through the representative image frames of different scenes contained in each video sequence.

We will also provide interactive tools for users to select arbitrary image segments from retrieved displayed images, specify and extract interesting features (e.g. texture, color, shape), composite features from multiple image segments (e.g. color of object A combined with texture of object B), to reformulate a new image query.

Users should also be allowed to combine visual features with text keywords, which can be provided by user's input or summary from explanatory text documents associated with retrieved images. For example, a user might be interested in all text and image materials related to a specific topic. Therefore, keywords describing this topic should spawn searches through the text and image sections of the archive. Through a common interface to both text and image type searches, the user should be allowed the freedom to choose the data-type domains to be searched. The retrieval mechanism must then integrate data from different domains, into a single set of query results.

Figure 4 shows a suite of graphic user interface in our VOD testbed. Through the *QoS setup panel*, users can retrieve the same video at different spatial resolutions based on our MPEG-2 spatial scalability implementation. Through the *interactive video playback interface*, users can execute VCR control functions during a video playback session. Traditional *bibliographic search interface* is also provided, in conjunction with the *texture-based image query interface*.

5. Storage Architecture Optimization

In a large-scale video server serving a large number of users, how to actually store video signals in the disk storage and how to design memory/storage hierarchy are critical issues that will significantly affect the overall system performance, in terms of storage utilization efficiency, maximal number of users supported, and required buffer size, etc. All these issues stem from the real-time retrieval requirement of video data. Arranging the video signal components in a periodical fashion in the disk storage has been proposed to reduce the data access delay (by minimizing the disk head seeking time) at the cost of user interactivity [7]. In the case of disk array, interleaving signal components (e.g., different image frames or different frequency bands) across multiple disks can provide simultaneous access to more users [8,9]. In general, this is a challenging optimization problem for mapping video data from a multi-dimensional space (different video sequences, different image frames, and different signal components) to multi-dimensional storage space (e.g., different blocks and tracks on different disks). Figure 5 illustrates this mapping problem.

Currently, we are studying the optimal data placement and retrieval scheduling scheme in our video server prototype, which will provide multi-resolution

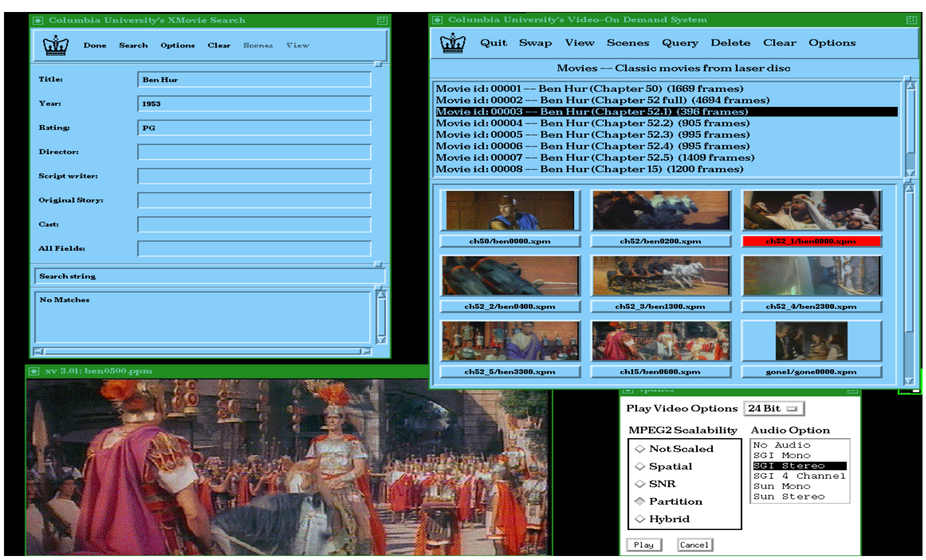

FIGURE 4. Current user interface of Columbia University's VOD testbed. This snapshot shows the video scene browser, the MPEG-2 playback interface, the QoS setup panel, and the bibliographic search database.

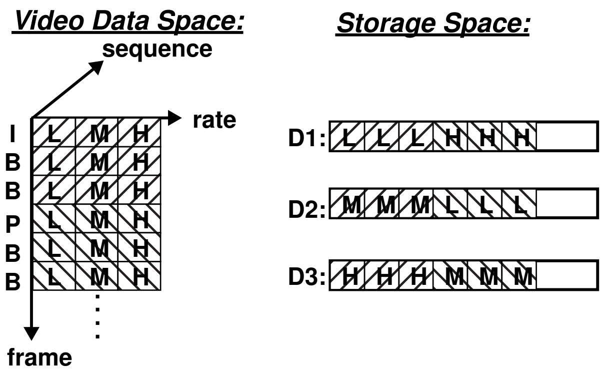

FIGURE 5. Mapping multiple components of each image frame of each video sequence to multiple disks. L,M,H represents different rate components, I,B,P are different frame types of MPEG.

retrieval capability to multiple users in a networked multi-server environment. We have shown some preliminary promising work for optimization of video data access in a single disk environment [10]. More complication will be introduced when user interactivity such as the VCR control function is required. Various optimization techniques are being examined, such as optimal interleaving, user access modelling, data prediction, load balancing, and cache management.

6. Conclusions

With the campus-wide VOD service as the initial target application, we are building an image/video server with advanced capabilities of storage/retrieval/query. Our research focus at this point covers innovative content-based image query interface, optimized real-time video data placement/retrieval scheduling, and heterogeneous QoS provision through multi-resolution coding and scalability options in the MPEG-2 standard. Heterogeneous communication connections (ATM and Internet) with other institutions within and outside the campus have been installed to explore potential applications of such advanced video servers in important interdisciplinary areas, such as digital libraries. The first generation of our VOD prototype incorporating many research results have been up and running.

7. References

1. S. -F. Chang, A. Eleftheriadis, and D. Anastassiou, “Some Interoperability Issues in Columbia’s Video on Demand Testbed”, Contribution to International Digital Audio/Visual Council, June 1994
2. R. Barber, W. Equitz, M. Flickner, W. Niblack, D. Petrovic, P. Yanker, “Efficient Query by Image Content for Very Large Image Database”, COMPCON ‘93, San Francisco, CA., 1993, pp. 17-19.
3. J.R. Smith and S.-F. Chang, “Quad-Tree Segmentation for Texture-Based Image Query,” to appear on ACM 2nd Multimedia Conference, San Francisco, Oct. 1994.
4. J.R. Smith and S.-F. Chang, “Transform Features for Texture Discrimination and Classification in Large Image Databases,” to appear on IEEE Inter. Conf. on Image Processing, Nov. 1994, Austin, TX.
5. P. Brodatz, Textures: a Photographic Album for Artists and Designers, Dover, New York, 1965.
6. Y. Juan and S.-F. Chang, “Scene Change Detection in a MPEG Compressed Video Sequence,” submitted to IEEE Intern. Conference on Data Engineering, 1994.
7. H. M. Vin, P. V. Rangan, “Designing a Multi-User HDTV Storage Server”, IEEE Journal on Selected Areas in Communications vol. 11, No. 1, January 1993
8. K. Keeton and R. Katz, “The Evaluation of Video Layout Strategies on a High-Bandwidth File Server,” Intern. Workshop on Network and Operating System Support for Digital Audio and Video, Lancaster, England, UK, Nov. 1993.
9. E. Chang and A. Zakhor, “Scalable Video Data Placement on Parallel Disk Arrays,” SPIE Symposium on Imaging Technology, San Jose, 1994.
10. P. Bocheck, H. Meadows, and S.-F. Chang, “A Disk Partitioning Technique for Reducing Multimedia Access Delay,” to appear on Intern. Conference on Distributed Multimedia Systems and Applications, Honolulu, Aug., 1994.
11. A. Eleftheriadis, and D. Anastassiou, “Optimal Data Partitioning of MPEG-2 Coded Video, “to appear on IEEE 1st Intern. Conf. on Image Processing, 1994.
12. *Special Issue on Video on Demand*, IEEE Communications Magazine, May 1994, Vol. 32, No. 5.
13. J. White and A. Klinger, “Image Coding in Visual Databases”, in Visual Database Systems, II, E. Knuth and L.M. Wegner (Editors), Elsevier Science Publishers B.V., 1992.
14. A. Nagasaka and Y. Tanaka, “Automatic Video Indexing and Full-Video Search for Object Appearances” In E. Knuth and L. M. Wegner, editors, *Video Database Systems, II*, Elsevier Science Publishers B.V., North-Holland, 1992, pp. 113 - 127.
15. Ming-Syan Chen, Dilip D. Kandlur, and Philip S. Yu, “Support for Fully Interactive Playout in A Disk-Array-Based Video Server”, to appear on ACM 2nd Multimedia Conference, San Francisco, Oct. 1994.
16. L. A. Rowe and R. Larson, “A Video-on-Demand System,” private communications.
17. S.W. Smoliar and H. Zhang, “Content-Based Video Indexing and Retrieval,” IEEE Multimedia Magazine, Vol.1, No.2, Summer 1994.
18. T.D.C. Little, G. Ahanger, R.J. Folz, J.F. Gibbon, W.W. Reeve, D.H. Schelleng, and D. Venkatesh, “A Digital On-Demand Video Service Supporting Content-Based Queries,” ACM 1st Multimedia Conference, Anaheim, CA, Aug. 1993.
19. The Digital Audio-Visual Council (DAVIC) Opening Forum, San Jose, CA, June 1-3, 1994.
20. S.E. Youngberg, “Rate/pitch modification of speech using the constant Q transform,” IEEE, ICASSP ‘79, Washington, DC, April 1979.
21. S.-F. Chang and D.G. Messerschmitt, “Manipulation and Compositing of MC-DCT Compressed Video,” to appear on IEEE Journal of Selected Areas in Communications, Special Issue on Intelligent Signal Processing, 1994.
22. P. Stanchev, A. Smeulders, and F. Groen, “An Approach to Image Indexing of Documents,” in *Visual Database Systems II*, Elsevier Science Publishers, 1992.
23. Tihao Chiang and Dimitris Anastassiou, “Hierarchical Coding of Digital Television,” IEEE Communications Magazine, May 1994