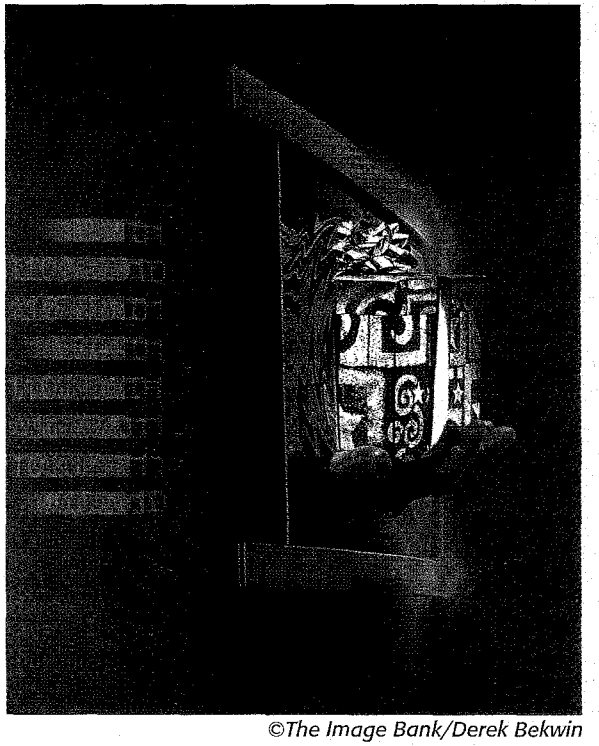
MPEG Digital Video-Coding Standards

Delivering Picture-Perfect Compression for Storage, Transmission, and Multimedia Applications

The efficient digital representation of image and video signals has been the subject of considerable research over the past 20 years. Digital video-coding technology has developed into a mature field and products have been developed that are targeted for a wide range of emerging applications, such as video on demand, digital TV/HDTV broadcasting, and multimedia image/video database services. With the increased commercial interest in video communications, the need for international image- and video-compression standards arose.

To meet this need, the Moving Picture Experts Group (MPEG) was formed to develop coding standards. MPEG-1 and MPEG-2 video-coding standards have attracted much attention world-wide recently, with an increasing number of very large scale integration (VLSI) and software implementations of these standards becoming commercially available. MPEG-4, the most recent MPEG standard that is still under development, is targeted for future content-based multimedia applications.

In this article we provide an overview of the MPEG video-coding algorithms and standards and their role in video communications. We review the basic concepts and techniques that are relevant in the context of the MPEG video-compression standards and outline MPEG-1 and MPEG-2 video-coding algorithms. The specific properties of the standards related to their applications are presented, and the basic elements of the forthcoming MPEG-4 standard are also described. We also discuss the performance of the standards and their success in the market place.



©The Image Bank/Derek Bekwin

Development of MPEG Standards

Modern image- and video-compression techniques offer the possibility to store or transmit the vast amount of data necessary to represent digital images and video in an efficient and robust way. New audio-visual applications in the fields of communication, multimedia, and broadcasting became possible based on digital video-coding technology. As manifold as applications for image coding and different approaches and algorithms are today, so were the first hardware implementations and even systems in the commercial field, such as private teleconferencing systems [1, 2]. However, with the ad-

1053-5888/97/$10.00©1997IEEE

vances in VLSI technology it became possible to open more application fields to a larger number of users and, therefore, the necessity for video-coding standards arose.

Commercially, international standardization of video communication systems and protocols aims to serve two important purposes: *interoperability* and *economy of scale*. Interworking between video communication equipment from different vendors is a desirable feature for users and equipment manufactures alike. It increases the attractiveness for buying and using video communication equipment because it enables large-scale international video data exchange via storage media or via communication networks. An increased demand can lead to "economy of scale"—the mass production of VLSI systems and devices—which in turn makes video equipment more affordable for a wide field of applications and users.

The ultimate goal of video source coding is the bit-rate reduction for storage and transmission.

From the beginning of the 1980s on, a number of international video and audio standardization activities started within the International Telephone Consultive Committee (CCITT), followed by the International Radio Consultive Committee (CCIR), and the International Organization of Standardization/International Electrotechnical Commission (ISO/IEC) [4]. MPEG was established in 1988 in the framework of the Joint ISO/IEC Technical Committee (JTC 1) on Information Technology with the mandate to develop standards for coded representation of moving pictures, associated audio, and their combination when used for storage and retrieval on digital storage media with a bit rate at up to about 1.5 Mbit/s. The standard was nicknamed MPEG-1 and was issued in 1992. The scope of the group was later extended to provide appropriate MPEG-2 video and associated audio-compression algorithms for a wide range of audio-visual applications at substantially higher bit rates not successfully covered or envisaged by the MPEG-1 standard. Specifically, MPEG-2 was given the charter to provide video quality not lower than NTSC/PAL and up to CCIR 601 quality with bit rates targeted between 2 and 10 Mbit/s. Emerging applications, such as digital cable TV distribution, networked database services via asynchronous transfer mode (ATM), digital video tape recorder (VTR) applications, and satellite and terrestrial digital broadcasting distribution, were seen to benefit from the increased quality expected to result from the emerging MPEG-2 standard. The MPEG-2 standard was released in 1994.

The MPEG-1 and MPEG-2 video-compression techniques developed and standardized by the MPEG group have developed into important and successful video-coding standards world-wide, with an increasing number of MPEG-1 and MPEG-2 VLSI chip-sets and products becoming available on the market. One key factor for the success is the generic structure of the MPEG standards, supporting a wide range of applications and applications-specific parameters. To support the wide range of applications profiles, a diversity of input parameters including flexible picture size and frame rate can be specified by the user. Another important factor is the fact that the MPEG group only standardized the decoder structures and the bit-stream formats. This allows a large degree of freedom for manufactures to optimize the coding efficiency (or, in other words, the video quality at a given bit rate) by developing innovative encoder algorithms even after the standards were finalized.

Anticipating the rapid convergence of telecommunications, computer, and TV/film industries, the MPEG group officially initiated a new MPEG-4 standardization phase in 1994—with the mandate to standardize algorithms and tools for coding and flexible representation of audio-visual data to meet the challenges of future multimedia applications and applications environments. In particular, MPEG-4 addresses the need for universal accessibility and robustness in error-prone environments, high interactive functionality, coding of natural and synthetic data, as well as high compression efficiency. Bit rates targeted for the MPEG-4 video standard are between 5-64 kbits/s for mobile or public switched telephone network (PSTN) video applications and up to 4 Mbits/s for TV/film applications. The release of the MPEG-4 International Standard is targeted for November 1998.

Fundamentals of MPEG Video-Compression Algorithms

Generally speaking, video sequences contain a significant amount of *statistical* and *subjective* redundancy within and between frames. The ultimate goal of video source coding is the bit-rate reduction for storage and transmission by exploring both statistical and subjective redundancies and to encode a "minimum set" of information using entropy coding techniques. This usually results in a compression of the coded video data compared to the original source data. The performance of video-compression techniques depends on the amount of redundancy contained in the image data as well as on the actual compression techniques used for coding. With practical coding schemes a trade-off between coding performance (high compression with sufficient quality) and implementation complexity is targeted. For the development of the MPEG compression algorithms the consideration of the capabilities of "state of the art" (VLSI) technology foreseen for the lifecycle of the standards was most important.

Dependent on the applications requirements we may envisage "lossless" and "lossy" coding of the video data. The aim of "lossless" coding is to reduce image or video

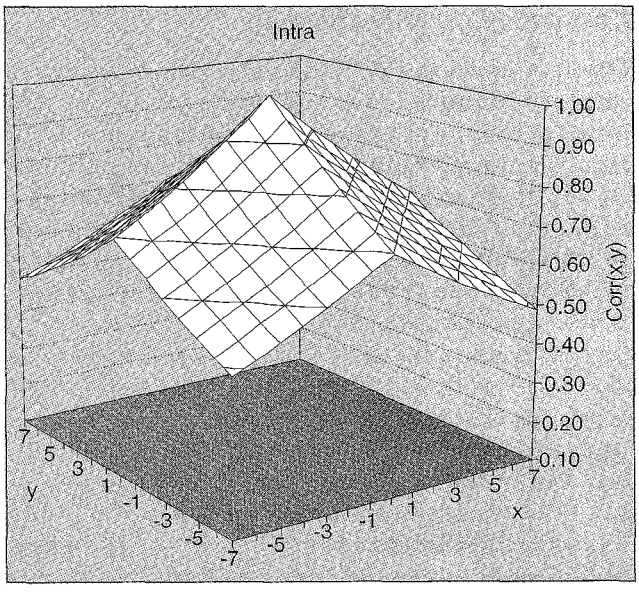
data for storage and transmission while retaining the quality of the original images—the *de*coded image quality is required to be identical to the image quality prior to *en*coding. In contrast, the aim of "lossy" coding techniques—and this is relevant to the applications envisioned by MPEG-1, MPEG-2, and MPEG-4 video standards—is to meet a given target bit rate for storage and transmission. Important applications comprise transmission of video over communications channels with constrained or low bandwidth and the efficient storage of video. In these applications high video compression is achieved by degrading the video quality—the *de*coded image "objective" quality is reduced compared to the quality of the original images prior to encoding (i.e., taking the mean-squared-error between both the original and reconstructed images as an objective image quality criteria). The smaller the target bit rate of the channel the higher the necessary compression of the video data and usually the more coding artifacts become visible. The ultimate aim of lossy coding techniques is to optimize image quality for a given target bit rate subject to "objective" or "subjective" optimization criteria. It should be noted that the degree of image degradation (both the objective degradation as well as the amount of visible artifacts) depends on the complexity of the image or video scene as much as on the sophistication of the compression technique—for simple textures in images and low video activity a good image reconstruction with no visible artifacts may be achieved even with simple compression techniques.

The MPEG Video-Coder Source Model

The MPEG digital video-coding techniques are statistical in nature. Video sequences usually contain statistical redundancies in both temporal and spatial directions. The basic statistical property upon which MPEG compression techniques rely is interpel correlation, including the assumption of simple, correlated translatory motion between consecutive frames. Thus, it is assumed that the magnitude of a particular image pel can be predicted from nearby pels within the same frame (using intraframe coding techniques) or from pels of a nearby frame (using interframe techniques). Intuitively, it is clear that in some circumstances, i.e., during scene changes of a video sequence, the temporal correlation between pels in nearby frames is small or even vanishes—the video scene then assembles a collection of uncorrelated still images. In this case, intraframe coding techniques are appropriate to explore spatial correlation to achieve efficient data compression.

The MPEG compression algorithms employ discrete cosine transform (DCT) coding techniques on image blocks of 8×8 pels to efficiently explore spatial correlations between nearby pels within the same image. However, if the correlation between pels in nearby frames is high, i.e., in cases where two consecutive frames have similar or identical content, it is desirable to use interframe differential pulse code modulation (DPCM) coding techniques employing temporal prediction (motion-compensated prediction between frames). In MPEG video-coding schemes an adaptive combination of both temporal motion-compensated prediction followed by transform coding of the remaining spatial information is used to achieve high data compression (hybrid DPCM/DCT coding of video).

Figure 1 depicts an example of intraframe pel-to-pel correlation properties of images, here modeled using a rather simple, but nevertheless valuable, statistical model. The simple model assumption already inherits basic correlation properties of many "typical" images upon which the MPEG algorithms rely, namely the high correlation between adjacent pixels and the monotonical decay of correlation with increased distance between pels. We will use this model assumption later to demonstrate some of the properties of transform-domain coding.



▲ *1. Spatial interelement correlation of "typical" images as calculated using an AR(1) Gauss Markov image model with high pel-pel correlation. Variables x and y describe the distance between pels in horizontal and vertical image dimensions, respectively.*

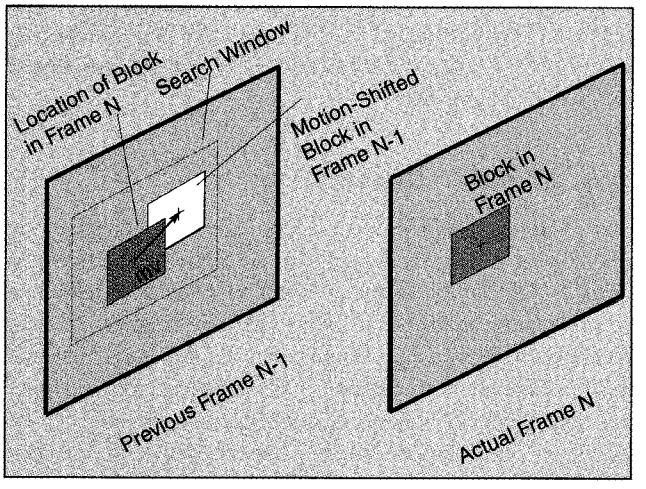
Subsampling and Interpolation

Almost all video-coding techniques described in the context of this article make extensive use of subsampling and quantization prior to encoding. The basic concept of subsampling is to reduce the dimension of the input video (horizontal dimension and/or vertical dimension) and thus the number of pels to be coded prior to the encoding process. It is worth noting that for some applications video is also subsampled in the temporal direction to reduce frame rate prior to coding. At the receiver the decoded images are interpolated for display. This technique may be considered as one of the most elementary compression techniques, which also makes use of specific physiological characteristics of the human eye and thus removes subjective redundancy contained in the video data—i.e., the human eye is more sensitive to changes in

brightness than to chromaticity changes. Therefore, the MPEG coding schemes first divide the images into YUV components (one luminance and two chrominance components). Next the chrominance components are subsampled relative to the luminance component with a Y:U:V ratio specific to particular applications (i.e., with the MPEG-2 standard a ratio of 4:1:1 or 4:2:2 is used).

Motion-Compensated Prediction

Motion-compensated prediction is a powerful tool to reduce temporal redundancies between frames and is used extensively in MPEG-1 and MPEG-2 video-coding standards as a prediction technique for temporal DPCM coding. The concept of motion compensation is based on the estimation of motion between video frames, i.e., if all elements in a video scene are approximately spatially displaced, the motion between frames can be described by a



▲ *2. Block matching approach for motion compensation: One motion vector (mv) is estimated for each block in the actual frame, N, to be coded. The motion vector points to a reference block of the same size in a previously coded frame, N – 1. The motion-compensated prediction error is calculated by subtracting each pel in a block with its motion shifted counterpart in the reference block of the previous frame.*

limited number of motion parameters (i.e., by motion vectors for translatory motion of pels). In this simple example the best prediction of an actual pel is given by a motion-compensated prediction pel from a previously coded frame. Usually both prediction error and motion vectors are transmitted to the receiver. However, encoding one motion information with each coded image pel is generally neither desirable nor necessary. Since the spatial correlation between motion vectors is often high, it is sometimes assumed that one motion vector is representative for the motion of a "block" of adjacent pels. To this aim, images are usually separated into disjoint blocks of pels (i.e., 16 × 16 pels in MPEG-1 and MPEG-2 standards) and only one motion vector is estimated, coded, and transmitted for each of these blocks (Fig. 2).

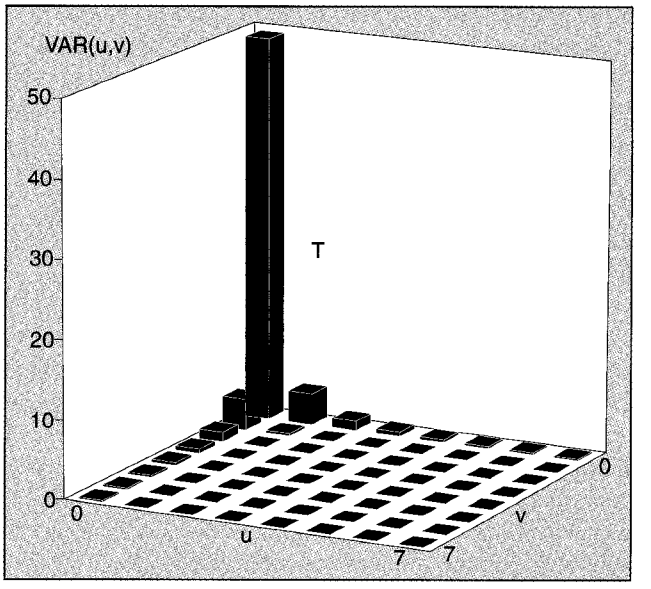
In the MPEG compression algorithms the motion-compensated prediction techniques are used for reducing temporal redundancies between frames and only the prediction error images—the difference between original images and motion-compensated prediction images—are encoded. In general, the correlation between pels in the motion-compensated interframe error images to be coded is reduced compared to the correlation properties of intraframes in Fig. 1 due to the prediction based on the previous coded frame.

Transform-Domain Coding

Transform coding has been studied extensively during the last two decades and has become a very popular compression method for still-image coding and video coding. The purpose of transform coding is to de-correlate the intra- or interframe error image content and to encode transform coefficients rather than the original pels of the images. To this aim the input images are split into disjoint blocks of pels, $\boldsymbol{b}$ (i.e., of size $N \times N$ pels). The transformation can be represented as a matrix operation using an $N \times N$ transform matrix, $\boldsymbol{A}$, to obtain the $N \times N$ transform coefficients, $\boldsymbol{c}$, based on a linear, separable, and unitary *forward* transformation

$$\boldsymbol{c} = \boldsymbol{A}\,\boldsymbol{b}\,\boldsymbol{A}^T.$$

Here, $\boldsymbol{A}^T$ denotes the transpose of the transformation matrix, $\boldsymbol{A}$. Note that the transformation is reversible since the original $N \times N$ block of pels, $\boldsymbol{b}$, can be reconstructed using a linear and separable *inverse* transformation (for a



▲ *3. The figure depicts the variance distribution of DCT coefficients "typically" calculated as average over a large number of image blocks. The variance of the DCT coefficients was calculated based on the statistical model used in Fig. 1. u and v describe the horizontal and vertical image transform domain variables within the 8x8 block. Most of the total variance is concentrated around the DC DCT-coefficient (u=0, v=0).*

unitary transform the inverse matrix A^{-1} is identical with the transposed matrix A^T, that is $A^{-1} = A^T$:

$$b = A^T c A.$$

Upon many possible alternatives, the DCT applied to smaller image blocks of usually 8 × 8 pels has become the most successful transform for still image and video coding [5]. In fact, DCT based implementations are used in most image- and video-coding standards due to their high decorrelation performance and the availability of fast DCT algorithms suitable for real-time implementations. VLSI implementations that operate at rates suitable for a broad range of video applications are commercially available today.

MPEG digital video-coding techniques are statistical in nature.

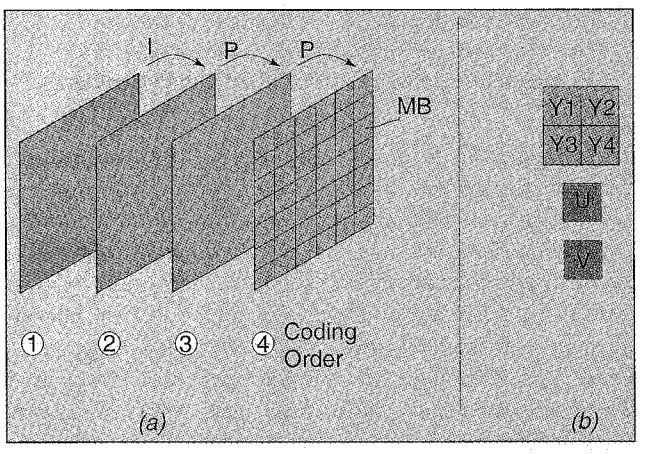
A major objective of transform coding is to make as many transform coefficients as possible small enough so that they are insignificant (in terms of statistical and subjective measures) and need not be coded for transmission. At the same time it is desirable to minimize statistical dependencies between coefficients with the aim to reduce the amount of bits needed to encode the remaining coefficients. Figure 3 depicts the variance (energy) of an 8 × 8 block of intraframe DCT coefficients based on the simple statistical model assumption already discussed in Fig. 1. Here, the variance for each coefficient represents the variability of the coefficient as averaged over a large number of frames. Coefficients with small variances are less significant for the reconstruction of the image blocks than coefficients with large variances. As may be depicted from Fig. 3, on average only a small number of DCT coefficients need to be transmitted to the receiver to obtain a valuable approximate reconstruction of the image blocks. Moreover, the most significant DCT coefficients are concentrated around the upper left corner (low DCT coefficients) and the significance of the coefficients decays with increased distance. This implies that higher DCT coefficients are less important for reconstruction than lower coefficients. Also, by employing motion-compensated prediction the transformation using the DCT usually results in a compact representation of the temporal DPCM signal in the DCT domain—which essentially inherits the similar statistical coherency as the signal in the DCT domain for the intraframe signals in Fig. 3 (although with reduced energy)—which is the reason why MPEG algorithms successfully employ DCT coding also for interframe compression [4].

The DCT is closely related to discrete Fourier transform (DFT) and it is of some importance to realize that the DCT coefficients can be given a frequency interpretation close to the DFT. Thus, low DCT coefficients relate to low spatial frequencies within image blocks and high DCT coefficients to higher frequencies. This property is used in MPEG coding schemes to remove subjective redundancies contained in the image data based on human visual systems criteria. Since the human viewer is more sensitive to reconstruction errors related to low spatial frequencies than to high frequencies, a frequency-adaptive weighting (quantization) of the coefficients according to the human visual perception (perceptual quantization) is often employed to improve the visual quality of the decoded images for a given bit rate.

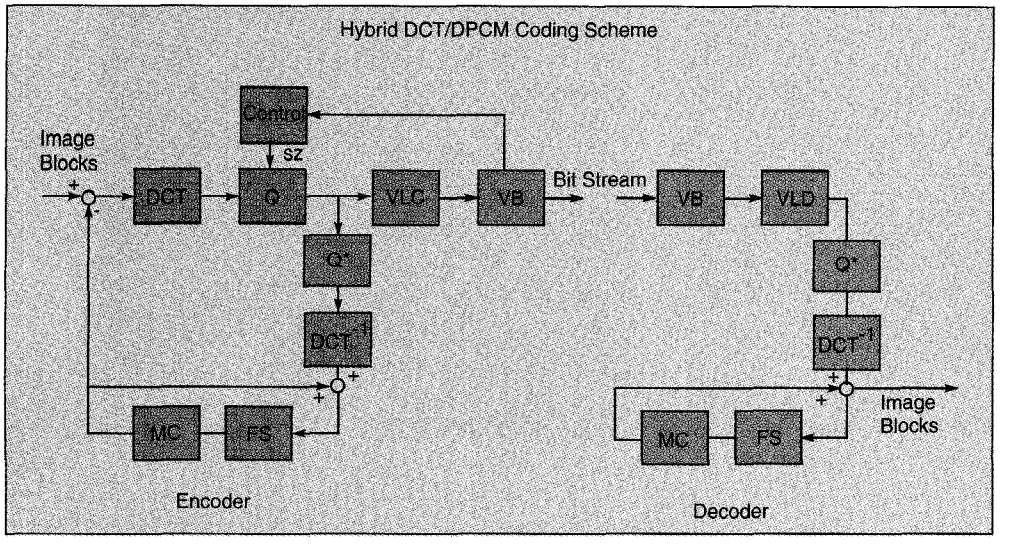
The combination of the two techniques described above—temporal motion-compensated prediction and transform-domain coding—can be seen as the key elements of the MPEG coding standards. A third characteristic element of the MPEG algorithms is that these two techniques are processed on small image blocks (of typically 16 × 16 pels for motion compensation and 8 × 8 pels for DCT coding). For this reason the MPEG coding algorithms are usually referred to as hybrid block-based DPCM/DCT algorithms.

The MPEG-1 Standard

The video-compression technique developed by MPEG-1 covers many applications from interactive systems on CD-ROM to the delivery of video over telecommunications networks. The MPEG-1 video-coding standard is thought to be generic. To support the wide range of applications profiles, a diversity of input parameters including flexible picture size and frame rate can be specified by the user. MPEG has recommended a constraint parameter set: every MPEG-1 compatible decoder must be able to support at least video-source parameters up to TV size: including a minimum number of 720 pix-



▲ *4. (a) Illustration of I-pictures (I) and P-pictures (P) in a video sequence. P-pictures are coded using motion-compensated prediction based on the nearest previous frame. Each frame is divided into disjoint "macroblocks." (b) With each macroblock, information related to four luminance blocks (Y1, Y2, Y3, Y4) and two chrominance blocks (U, V) is coded. Each block contains 8x8 pels.*



▲ *5. Block diagram of a basic hybrid DCT/DPCM encoder and decoder structure.*

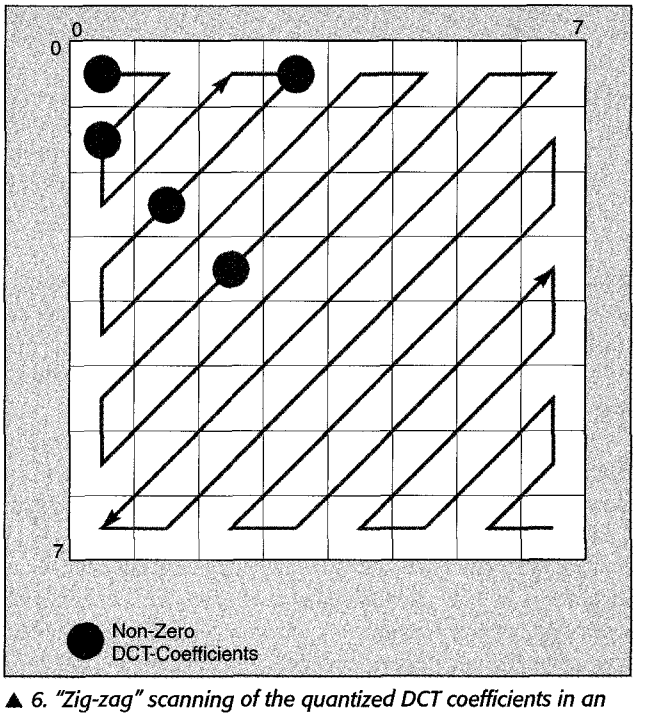
The block diagram of the basic hybrid DPCM/DCT MPEG-1 encoder and decoder structure is depicted in Fig. 5. The first frame in a video sequence (I-picture) is encoded in INTRA mode without reference to any past or future frames. At the encoder the DCT is applied to each 8 × 8 luminance and chrominance block and, after output of the DCT, each of the 64 DCT coefficients is uniformly quantized (Q). The quantizer stepsize (sz) used to quantize the DCT-coefficients within a macroblock is transmitted to the receiver. After quantization, the lowest DCT coefficient (DC coefficient) is treated differently from the remaining coefficients (AC coefficients). The DC coefficient corresponds to the average intensity of the component block and is encoded using a differential DC prediction method (Because there is usually strong correlation between the DC values of adjacent 8 × 8 blocks, the quantized DC coefficient is encoded as

els per line, a minimum number of 576 lines per picture, a minimum frame rate of 30 frames per second, and a minimum bit rate of 1.86 Mbits/s. The standard video input consists of a noninterlaced video picture format. It should be noted that by no means is the application of MPEG-1 limited to this constrained parameter set.

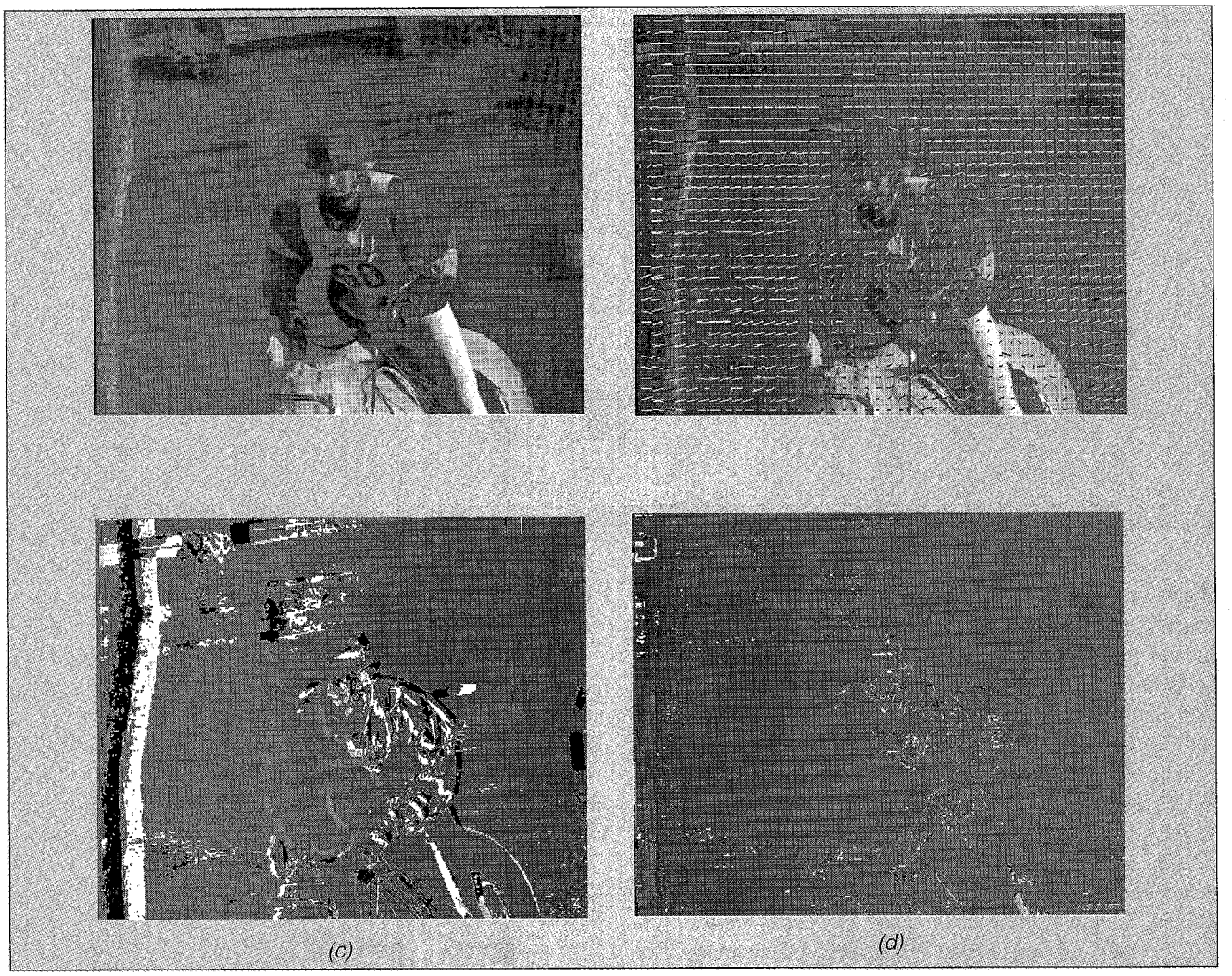
The MPEG-1 video algorithm has been developed with respect to the JPEG and H.261 activities. It was sought to retain a large degree of commonalty with the CCITT H.261 standard so that implementations supporting both standards were plausible. However, MPEG-1 was primarily targeted for multimedia CD-ROM applications, requiring additional functionality supported by both encoder and decoder. Important features provided by MPEG-1 include frame based *random access* of video, *fast forward/fast reverse (FF/FR)* searches through compressed bit streams, *reverse playback* of video and *editability* of the compressed bit stream.

The Basic MPEG-1 Interframe Coding Scheme

The basic MPEG-1 (as well as the MPEG-2) video-compression technique is based on a macroblock structure, motion compensation, and the conditional replenishment of macroblocks. As outlined in Figure 4(a), the MPEG-1 coding algorithm encodes the first frame in a video sequence in intraframe coding mode (I-picture). Each subsequent frame is coded using interframe prediction (P-pictures)—only data from the nearest previously coded I- or P-frame is used for prediction. The MPEG-1 algorithm processes the frames of a block-based video sequence. Each color input frame in a video sequence is partitioned into nonoverlapping "macroblocks" as depicted in Figure 4(b). Each macroblock contains blocks of data from both luminance and co-sited chrominance bands—four luminance blocks (Y_1, Y_2, Y_3, Y_4) and two chrominance blocks (U, V), each with a size of 8 × 8 pels. Thus, the sampling ratio between Y:U:V luminance and chrominance pels is 4:1:1.



▲ *6. "Zig-zag" scanning of the quantized DCT coefficients in an 8x8 block. Only the nonzero quantized DCT coefficients are encoded. The possible locations of nonzero DCT coefficients are indicated in the figure. The zig-zag scan attempts to trace the DCT coefficients according to their significance. With reference to Fig. 3, the lowest DCT-coefficient (0,0) contains most of the energy within the blocks and the energy is concentrated around the lower DCT coefficients.*



▲ *7. (a) Frame at time instance N to be coded. (b) Frame at instance N – 1 used for prediction of the content in frame N (note that the motion vectors depicted in the image are not part of the reconstructed image stored at the encoder and decoder). (c) Prediction error image obtained without using motion compensation—all motion vectors are assumed to be zero. (d) Prediction error image to be coded if motion-compensated prediction is employed.*

the difference between the DC value of the previous block and the actual DC value.). The nonzero quantizer values of the remaining DCT coefficients and their locations are then "zig-zag" scanned and run-length entropy coded using variable length code (VLC) tables.

The concept of "zig-zag" scanning of the coefficients is outlined in Fig. 6. The scanning of the quantized DCT-domain 2-dimensional signal followed by variable-length code-word assignment for the coefficients serves as a mapping of the 2-dimensional image signal into a 1-dimensional bit stream. The nonzero AC coefficient quantizer values (length, •) are detected along the scan line as well as the distance (run) between two consecutive nonzero coefficients. Each consecutive (run, length) pair is encoded by transmitting only one VLC code word. The purpose of "zig-zag" scanning is to trace the low-frequency DCT coefficients (containing the most energy) before tracing the high-frequency coefficients. (The location of each nonzero coefficient along the zig-zag scan is encoded relative to the location of the previous coded coefficient. The zig-zag scan philosophy attempts to trace the nonzero coefficients according their likelihood of appearance to achieve an efficient entropy coding. With reference to Fig. 5 the DCT coefficients most likely to appear are concentrated around the DC coefficient with decreasing importance. For many images the coefficients are traced efficiently using the zig-zag scan.)

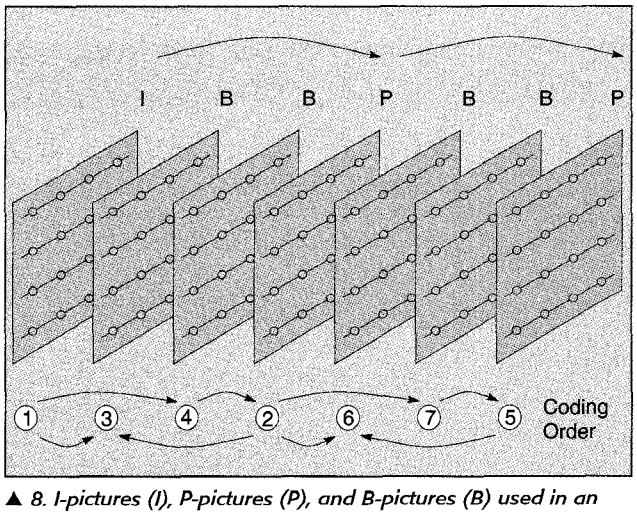
The decoder performs the reverse operations, first extracting and decoding (VLD) the variable-length coded words from the bit stream to obtain locations and quantizer values of the nonzero DCT coefficients for each block. With the reconstruction (Q^*) of all nonzero DCT coefficients belonging to one block and subsequent inverse DCT (DCT^{-1}), the quantized block pixel values are obtained. By processing the entire bit stream all image blocks are decoded and reconstructed.

For coding P-pictures, the previously I- or P-picture frame $N-1$ is stored in a frame store (FS) in both the encoder and decoder. Motion compensation is performed on a macroblock basis—only one motion vector is esti-

mated between frame N and frame $N-1$ for a particular macroblock to be encoded. These motion vectors are coded and transmitted to the receiver. The motion-compensated prediction error is calculated by subtracting each pel in a macroblock with its motion-shifted counterpart in the previous frame. An 8×8 DCT is then applied to each of the 8×8 blocks contained in the macroblock followed by quantization (Q) of the DCT coefficients with subsequent run-length coding and entropy coding (VLC). A video buffer (VB) is needed to ensure that a constant target bit-rate output is produced by the encoder. The quantization stepsize (sz) can be adjusted for each macroblock in a frame to achieve a given target bit rate and to avoid buffer overflow and underflow.

The decoder uses the reverse process to reproduce a macroblock of frame N at the receiver. After decoding the variable-length words (VLD) contained in the VB the pixel values of the prediction error are reconstructed (Q^{-1}-, and DCT^{-1}-operations). The motion-compensated pixels from the previous frame, $N-1$, contained in the FS are added to the prediction error to recover the particular macroblock of frame N.

The advantage of coding video using the motion-compensated prediction from the previously reconstructed frame, $N-1$, in an MPEG coder is illustrated in Figs. 7(a)-7(d) for a typical test sequence. Figure 7(a) depicts a frame at time instance N to be coded and Figure 7(b) shows the reconstructed frame at instance $N-1$, which is stored in the FS at both encoder and decoder. The block motion vectors (mv, see also Fig. 2) depicted in Figure 7(b) were estimated by the encoder motion-estimation procedure and provide a prediction of the translatory motion displacement of each macroblock in frame N with reference to frame $N-1$. Figure 7(c) depicts the pure frame-difference signal (frame N – frame $N-1$), which is obtained if no motion-compensated prediction is used in the coding process—thus, all motion vectors are assumed to be zero. Figure 7d depicts the motion compensated frame difference signal when the motion vectors in Figure 7b are used for prediction. It is apparent that the residual signal to be coded is greatly reduced using motion compensation if compared to pure frame-difference coding in Figure 7(c).



▲ *8. I-pictures (I), P-pictures (P), and B-pictures (B) used in an MPEG-1 video sequence. B-pictures can be coded using motion-compensated prediction based on the two nearest already-coded frames (either I-picture or P-picture). The arrangement of the picture coding types within the video sequence is flexible to suit the needs of diverse applications. The direction for prediction is indicated in the figure.*

Conditional Replenishment

An essential feature supported by the MPEG-1 coding algorithm is the possibility to update macroblock information at the decoder only if needed—if the content of the macroblock has changed in comparison to the content of the same macroblock in the previous frame (conditional macroblock replenishment). The key for efficient coding of video sequences at lower bit rates is the selection of appropriate prediction modes to achieve conditional replenishment. The MPEG standard distinguishes mainly between three different macroblock coding types (MB types):

▲ *Skipped MB*: prediction from previous frame with zero motion vector. No information about the macroblock is coded nor transmitted to the receiver.

▲ *Inter MB*: motion-compensated prediction from the previous frame is used. The MB type, the MB address and, if required, the motion vector, the DCT coefficients, and quantization stepsize are transmitted.

▲ *Intra MB*: no prediction is used from the previous frame (intraframe prediction only). Only the MB type, the MB address, and the DCT coefficients and quantization stepsize are transmitted to the receiver.

Rate Control

An important feature supported by the MEPG-1 encoding algorithms is the possibility to tailor the bit rate (and thus the quality of the reconstructed video) to specific applications requirements by adjusting the quantizer stepsize (sz) in Fig. 5 for quantizing the DCT coefficients. Coarse quantization of the DCT coefficients enables the storage or transmission of video with high compression ratios, but, depending on the level of quantization, may result in significant coding artifacts. The MPEG-1 standard allows the encoder to select different quantizer values for each coded macroblock—this enables a high degree of flexibility to allocate bits in images where needed to improve image quality. Furthermore it allows the generation of both constant and variable bit rates for storage or real-time transmission of the compressed video.

The rate-control algorithm used to compress video is not part of the MPEG-1 standard and it is thus left to the implementers to develop efficient strategies. It is worth emphasizing that the efficiency of the rate-control algorithms selected by manufacturers to compress video at a given bit rate heavily impact on the visible quality of the video reconstructed at the decoder.

The MPEG-1 video-coding standard is thought to be generic.

Specific Storage Media Functionalities

For accessing video from storage media, the MPEG-1 video-compression algorithm was designed to support important functionalities such as random access and FF and FR playback functionalities. To incorporate the requirements for storage media and to further explore the significant advantages of motion compensation and motion interpolation, the concept of B-pictures (bi-directional predicted/bi-directional interpolated pictures) was introduced by MPEG-1. This concept is depicted in Fig. 8 for a group of consecutive pictures in a video sequence. Three types of pictures are considered: intrapictures (I-pictures) are coded without reference to other pictures contained in the video sequence, as already introduced in Fig. 4. I-pictures allow access points for random access and FF/FR functionality in the bit stream but achieve only low compression. Interframe predicted pictures (P-pictures) are coded with reference to the nearest previously coded I-picture or P-picture, usually incorporating motion compensation to increase coding efficiency. Since P-pictures are usually used as reference for prediction for future or past frames, they provide no suitable access points for random access functionality or editability. Bi-directional predicted/interpolated pictures (B-pictures) require both past and future frames as references. To achieve high compression, motion compensation can be employed based on the nearest past and future P-pictures or I-pictures. B-pictures themselves are never used as references.

The user can arrange the picture types in a video sequence with a high degree of flexibility to suit diverse applications requirements. As a general rule, a video sequence coded using I-pictures only (I I I I I I ...) allows the highest degree of random access, FF/FR, and editability, but achieves only low compression. A sequence coded with a regular I-picture update and no B-pictures (i.e., I P P P P P P I P P P P ...) achieves moderate compression and a certain degree of random access and FF/FR functionality. Incorporation of all three pictures types, as depicted in Fig. 8, for example, (I B B P B B P B B I B B P ...), may achieve high compression and reasonable random access and FF/FR functionality but also increases the coding delay significantly. This delay may not be tolerable for applications such as videotelephony or videoconferencing.

Coding of Interlaced Video Sources

The standard video input format for MPEG-1 is noninterlaced. However, coding of interlaced color television with both 525 and 625 lines at 29.97 and 25 frames per second, respectively, is an important application for the MPEG-1 standard. A suggestion for coding Rec.601 digital color television signals has been made by MPEG-1 based on the conversion of the interlaced source to a progressive intermediate format. In essence, only one horizontally subsampled field of each interlaced video input frame is encoded, i.e., the subsampled top field. At the receiver the even field is predicted from the decoded and horizontally interpolated odd field for display. The necessary preprocessing steps required prior to encoding and the postprocessing required after decoding are described in detail in the Informative Annex of the MPEG-1 International Standard document [6].

The MPEG-2 Standard

A key factor for the world-wide success of MPEG-1 is the generic structure of the standard, which supports a broad range of applications and applications-specific parameters. However, MPEG continued its standardization efforts in 1991 with a second phase (MPEG-2) to provide a video-coding solution for applications not originally covered or envisaged by the MPEG-1 standard. Specifically, MPEG-2 was given the charter to provide video quality not lower than NTSC/PAL and up to CCIR 601 quality. Emerging applications, such as digital cable TV distribution, networked database services via ATM, digital VTR applications, and satellite and terrestrial digital broadcasting distribution, were seen to benefit from the increased quality expected to result from the new MPEG-2 standardization phase. Work was carried out in collaboration with the ITU-T SG 15 Experts Group for ATM video coding and in 1994 the MPEG-2 Draft International Standard (which is identical to the ITU-T H.262 recommendation) was released [2]. The specification of the standard is intended to be generic—hence, the standard

Table 1. Upper bound of parameters at each level of a profile.

Level	Parameters
HIGH	1920 samples/line 1152 lines/frame 60 frames/s 80 Mbit/s
HIGH 1440	1440 samples/line 1152 lines/frame 60 frames/s 60 Mbit/s
MAIN	720 samples/line 576 lines/frame 30 frames/s 15 Mbit/s
LOW	352 samples/line 288 lines/frame 30 frames/s 4 Mbit/s

Table 2. Algorithms and functionalities supported with each profile.	
Profile	Algorithms
HIGH	Supports all functionality provided by the Spatial Scalable Profile plus the provision to support: • 3 layers with the SNR and Spatial scalable coding modes • 4:2:2 YUV-representation for improved quality requirements
SPATIAL Scalable	Supports all functionality provided by the SNR Scalable Profile plus an algorithm for: • spatial scalable coding (2 layers allowed): • 4:0:0 YUV-representation
SNR Scalable	Supports all functionality provided by the MAIN Profile plus an algorithm for: • SNR scalable coding (2 layers allowed) • 4:2:0 YUV-representation
MAIN	Nonscalable coding algorithm supporting functionality for: • coding interlaced video • random access • B-picture prediction modes • 4:2:0 YUV-representation
SIMPLE	Includes all functionality provided by the MAIN Profile but: • does not support B-picture prediction modes • 4:2:0 YUV-representation

aims to facilitate the bit-stream interchange among different applications and transmission and storage media.

Basically, MPEG-2 can be seen as a superset of the MPEG-1 coding standard and was designed to be backward compatible to MPEG-1—every MPEG-2-compatible decoder can decode a valid MPEG-1 bit stream. Many video-coding algorithms were integrated into a single syntax to meet the diverse applications requirements. New coding features were added by MPEG-2 to achieve sufficient functionality and quality, thus prediction modes were developed to support efficient coding of *interlaced video*. In addition, *scalable video* coding extensions were introduced to provide additional functionality, such as embedded coding of digital TV and HDTV, and graceful quality degradation in the presence of transmission errors.

However, implementation of the full syntax may not be practical for most applications. MPEG-2 has introduced the concept of "Profiles" and "Levels" to stipulate conformance between equipment not supporting the full implementation. Profiles and Levels provide means for defining subsets of the syntax and, thus, the decoder capabilities required to decode a particular bit stream. This concept is illustrated in Tables 1 and 2.

As a general rule, each Profile defines a new set of algorithms added as a superset to the algorithms in the Profile below. A Level specifies the range of the parameters that are supported by the implementation (i.e., image size, frame rate, and bit rates). The MPEG-2 core algorithm at MAIN Profile features nonscalable coding of both progressive and interlaced video sources. It is expected that most MPEG-2 implementations will at least conform to the MAIN Profile at MAIN Level, which supports nonscalable coding of digital video with approximately digital TV parameters—a maximum sample density of 720 samples per line and 576 lines per frame, a maximum frame rate of 30 frames per second, and a maximum bit rate of 15 Mbit/s.

MPEG-2 Nonscalable Coding Modes

The MPEG-2 algorithm defined in the MAIN Profile is a straightforward extension of the MPEG-1 coding scheme to accommodate coding of interlaced video, while retaining the full range of functionality provided by MPEG-1. Identical to the MPEG-1 standard, the MPEG-2 coding algorithm is based on the general hybrid DCT/DPCM coding scheme as outlined in Fig. 5, incorporating a macroblock structure, motion compensation, and coding modes for conditional replenishment of macroblocks. The concept of I-pictures, P-pictures, and B-pictures as introduced in Fig. 8 is fully retained in MPEG-2 to achieve efficient motion prediction and to assist random-access functionality. Note that the algorithm defined with the MPEG-2 SIMPLE Profile is basically identical with the one in the MAIN Profile, except that no B-picture prediction modes are allowed at the encoder. Thus, the additional implementation complexity and the additional frame stores necessary for the decoding of B-pictures are

Every MPEG-2-compatible decoder can decode a valid MPEG-1 bit stream.

not required for MPEG-2 decoders only conforming to the SIMPLE Profile.

Field and Frame Pictures

MPEG-2 has introduced the concept of *frame pictures* and *field pictures* along with particular *frame prediction* and *field prediction* modes to accommodate coding of progressive and interlaced video. For interlaced sequences it is assumed that the coder input consists of a series of odd (top) and even (bottom) fields that are separated in time by a field period. Two fields of a frame may be coded separately (field pictures, see Fig. 9). In this case each field is separated into adjacent nonoverlapping macroblocks and the DCT is applied on a field basis. Alternatively, two fields may be coded together as a frame (frame pictures) similar to conventional coding of progressive video sequences. Here, consecutive lines of top and bottom fields are simply merged to form a frame. Notice, that both frame pictures and field pictures can be used in a single video sequence.

Field and Frame Prediction

New motion-compensated field-prediction modes were introduced by MPEG-2 to efficiently encode field pictures and frame pictures. A simplified example of this new concept is illustrated in Figure 9 for an interlaced video sequence, here assumed to contain only three field pictures and no B-pictures. In field prediction, predictions are made independently for each field by using data from one or more previously decoded fields, i.e., for a top field a prediction may be obtained from either a previously decoded top field (using motion-compensated prediction) or from the previously decoded bottom field belonging to the same picture. Generally, the interfield prediction from the decoded field in the same picture is preferred if no motion occurs between fields. An indication to which reference field is used for prediction is transmitted with the bit stream. Within a field picture all predictions are field predictions.

Frame prediction forms a prediction for a frame picture based on one or more previously decoded frames. In a frame picture either field or frame predictions may be used and the particular prediction mode preferred can be selected on a macroblock-by-macroblock basis. It must be understood, however, that the fields and frames from which predictions are made may have themselves been decoded as either field or frame pictures.

MPEG-2 has introduced new motion-compensation modes to efficiently explore temporal redundancies between fields, namely the "Dual Prime" prediction and the motion compensation based on 16 × 8 blocks. A discussion of these methods is beyond the scope of this article.

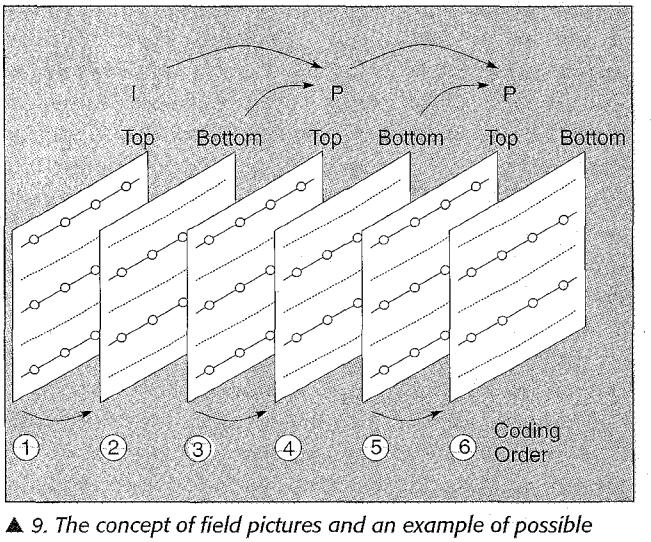
Chrominance Formats

MPEG-2 has specified additional Y:U:V luminance and chrominance subsampling ratio formats to assist and foster applications with the highest video quality requirements. Next to the 4:2:0 format already supported by MPEG-1, the specification of MPEG-2 is extended to 4:2:2 formats suitable for studio video-coding applications.

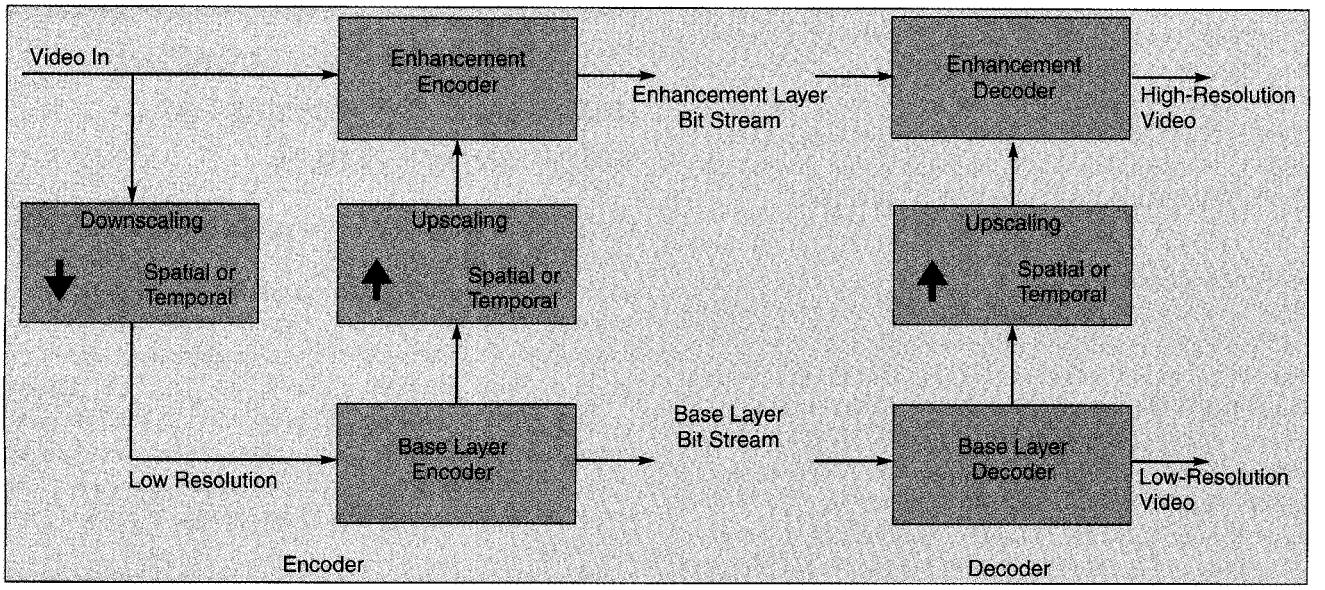
MPEG-2 Scalable Coding Extensions

The scalability tools standardized by MPEG-2 support applications beyond those addressed by the basic MAIN Profile coding algorithm. The intention of scalable coding is to provide interoperability between different services and to flexibly support receivers with different display capabilities. Receivers either not capable or willing to reconstruct the full resolution video can decode subsets of the layered bit stream to display video at lower spatial or temporal resolution or with lower quality. Another important purpose of scalable coding is to provide a layered video bit stream that is amenable for prioritized transmission. The main challenge here is to reliably deliver video signals in the presence of channel errors, such as cell loss in ATM-based transmission networks or cochannel interference in terrestrial digital broadcasting.

Flexibly supporting multiple resolutions is of particular interest for interworking between HDTV and standard definition television (SDTV), in which case it is important for the HDTV receiver to be compatible with



▲ *9. The concept of field pictures and an example of possible field prediction. The top fields and the bottom fields are coded separately. However, each bottom field is coded using motion-compensated interfield prediction based on the previously coded top field. The top fields are coded using motion-compensated interfield prediction based on either the previously coded top field or based on the previously coded bottom field. This concept can be extended to incorporate B-pictures.*



▲ *10. Scalable coding of video.*

the SDTV product. Compatibility can be achieved by means of scalable coding of the HDTV source, and the wasteful transmission of two independent bit streams to the HDTV and SDTV receivers can be avoided. Other important applications for scalable coding include video database browsing and multiresolution playback of video in multimedia environments.

Figure 10 depicts the general philosophy of a multiscale video-coding scheme. Here two layers are provided, each layer supporting video at a different scale, i.e., a multiresolution representation can be achieved by downscaling the input video signal into a lower resolution video (downsampling spatially or temporally). The downscaled version is encoded into a base-layer bit stream with reduced bit rate. The upscaled reconstructed base-layer video (upsampled spatially or temporally) is used as a prediction for the coding of the original input video signal. The prediction error is encoded into an enhancement-layer bit stream. If a receiver is either not capable or willing to display the full quality video, a downscaled video signal can be reconstructed by only decoding the base-layer bit stream. It is important to notice, however, that the display of the video at highest resolution with reduced quality is also possible by only decoding the lower bit rate base layer. Thus, scalable coding can be used to encode video with a suitable bit rate allocated to each layer in order to meet specific bandwidth requirements of transmission channels or storage media. Browsing through video data bases and transmission of video over heterogeneous networks are applications expected to benefit from this functionality.

During the MPEG-2 standardization phase it was found impossible to develop one generic scalable coding scheme capable to suit all of the diverse applications requirements envisaged. While some applications are constricted to low implementation complexity, others call for very high coding efficiency. As a consequence, MPEG-2 has standardized three scalable coding schemes: signal-to-noise ratio (SNR) scalability (quality), spatial scalability, and temporal scalability—each of which are targeted to assist applications with particular requirements. The scalability tools provide algorithmic extensions to the nonscalable scheme defined in the MAIN profile. It is possible to combine different scalability tools into a hybrid coding scheme; i.e., interoperability between services with different spatial resolutions *and* frame rates can be supported by means of combining the spatial scalability and the temporal scalability tool into a hybrid-layered coding scheme. Interoperability between HDTV and SDTV services can be provided *along* with a certain resilience to channel errors by combining the spatial scalability extensions with the SNR scalability tool [7]. The MPEG-2 syntax supports up to three different scalable layers.

SNR Scalability

This tool has been primarily developed to provide graceful degradation (quality scalability) of the video quality in prioritized transmission media. If the base layer can be protected from transmission errors, a version of the video with gracefully reduced quality can be obtained by decoding the base layer signal only. The algorithm used to achieve graceful degradation is based on a frequency (DCT domain) scalability technique. Both layers in Fig. 10 encode the video signal at the same spatial resolution. At the base layer the DCT coefficients are coarsely quantized to achieve moderate image quality at reduced bit rate. The enhancement layer encodes the difference between the nonquantized DCT coefficients and the quantized coefficients from the base layer with finer quantization stepsize. The method is implemented as a simple and straightforward extension to the MAIN Profile MPEG-2 coder and achieves excellent coding efficiency.

It is also possible to use this method to obtain video with lower spatial resolution at the receiver. If the de-

coder selects the lowest $N \times N$ DCT coefficients from the base-layer bit stream, nonstandard inverse DCTs of size $N \times N$ can be used to reconstruct the video at reduced spatial resolution [8, 9]. However, depending on the encoder and decoder implementations the lowest-layer downscaled video may be subject to drift [10].

Spatial Scalability

Spatial scalability has been developed to support displays with different spatial resolutions at the receiver—lower spatial resolution video can be reconstructed from the base layer. This functionality is useful for many applications including embedded coding for HDTV/TV systems, allowing a migration from a digital TV service to higher spatial resolution HDTV services [6, 11, 12]. The algorithm is based on a classical pyramidal approach for progressive image coding [13, 14]. Spatial scalability can flexibly support a wide range of spatial resolutions but adds considerable implementation complexity to the MAIN Profile coding scheme.

Temporal Scalability

The temporal scalability tool was developed with an aim similar to spatial scalability—stereoscopic video can be supported with a layered bit stream suitable for receivers with stereoscopic display capabilities. Layering is achieved by providing a prediction of one of the images of the stereoscopic video (i.e., left view) in the enhancement layer based on coded images from the opposite view transmitted in the base layer.

Data Partitioning

Data partitioning is intended to assist with error concealment in the presence of transmission or channel errors in ATM, terrestrial broadcast, or magnetic recording environments. Because the tool can be entirely used as a post-processing and preprocessing tool to any single-layer coding scheme, it has not been formally standardized with MPEG-2 but is referenced in the Informative Annex of the MPEG-2 DIS document [11]. Similar to the SNR scalability tool, the algorithm is based on the separation of DCT coefficients and is implemented with very low complexity compared to the other scalable coding schemes. To provide error protection, the coded DCT coefficients in the bit stream are simply separated and transmitted in two layers with different error likelihood.

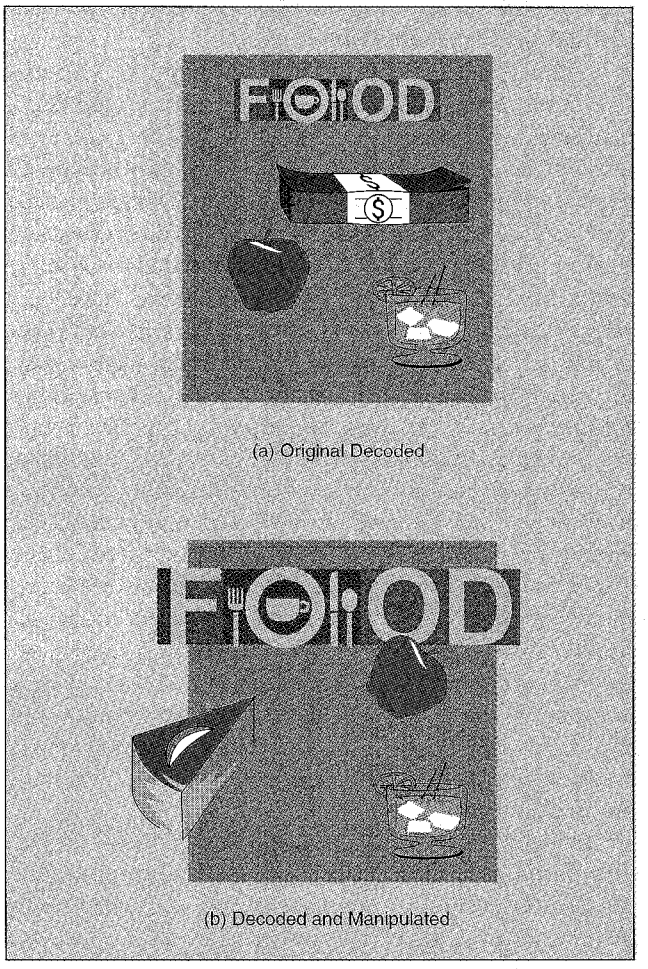
The Emergence of MPEG-4 for Multimedia

Anticipating the rapid convergence of telecommunications, computer, and TV/film industries, the MPEG group officially initiated a new MPEG-4 standardization phase in 1994—with the mandate to standardize algorithms and tools for coding and flexible representation of audio-visual data to meet the challenges of future multimedia applications and applications environments [15-16]. In particular, MPEG-4 addresses the need for:

The MPEG-4 video standard introduces the concept of video object planes.

▲ *Universal accessibility and robustness in error prone environments*: Multimedia audio-visual data need to be transmitted and accessed in heterogeneous network environments, possibly under severe error conditions (e.g., mobile channels). Although the MPEG-4 standards will be network (physical-layer) independent in nature, the algorithms and tools for coding audio-visual data need to be designed with awareness of network peculiarities.

▲ *High interactive functionality*: Future multimedia applications will call for extended interactive functionalities to assist the user's needs. In particular, the flexible, highly interactive access to and manipulation of audio-visual data will be of prime importance. It is envisioned that—in addition to conventional playback of audio and video sequences—the user will need to access "content" of



▲ *11. An example of flexible content-based access and manipulation of objects in MPEG-4 image sequences.*

audio-visual data to present and manipulate/store the data in a highly flexible way.

▲ *Coding of natural and synthetic data*: Next-generation graphics processors will enable multimedia terminals to present both pixel-based audio and video data together with synthetic audio/speech and video in a highly flexible way. MPEG-4 will assist the efficient and flexible coding and representation of both natural (pixel-based) as well as synthetic data.

▲ *Compression efficiency*: For the storage and transmission of audio-visual data a high coding efficiency, meaning a good quality of the reconstructed data, compression efficiency is required. Improved coding efficiency, in particular at very low bit rates below 64 kbits/s, continues to be an important functionality to be supported by the MPEG-4 video standard.

Bit rates targeted for the MPEG-4 video standard are between 5-64 kbits/s for mobile or PSTN video applications [17] and up to 4 Mbits/s for TV/film applications. Seven new (with respect to existing or emerging standards) key video-coding functionalities have been defined that support the MPEG-4 focus and provide the main requirements for the work in the MPEG video group [16]. The requirements cover the main topics related to "Content-Based Interactivity," "Compression" and "Universal Access." The release of the MPEG-4 International Standard is targeted for November 1998.

Content-Based Interactivity

In addition to standard MPEG-1- or MPEG-2-like provisions for efficient coding of conventional image or audio sequences, MPEG-4 will enable an efficient coded representation of the audio and video data that can be "content based," with the aim to use and present the data in a highly flexible way [15-16]. In particular, it is envisioned to allow the access and manipulation of audio-visual objects in the compressed domain at the coded data level—to assist future multimedia data-base access applications such as the flexible presentation of image or audio content in the World Wide Web, computer games, and related applications.

Video

The basic concept of the envisioned MPEG-4 "content-based" functionality for image/video applications is illustrated in Fig. 11 for a simple example of an image scene containing a number of video objects, here the background, several items, and a text overlay. The attempt is to encode the sequence in a way that will allow the separate decoding and reconstruction of the objects for the user—to assist the presentation and manipulation of the original scene in a flexible way.

The MPEG-4 video-coding standard will provide an "object layered" bit stream to assist this functionality. Each object is coded into a separate object bit-stream layer. The shape and transparency of the object—as well as the spatial coordinates and additional parameters describing scales and location, such as object zoom, rotation, translation, or related—are included in the bit stream. The user can either reconstruct the original sequence in its entirety, by decoding all "object layers" and by displaying the objects at original sizes and scales, and at the original location as indicated in Figure 11(a). Alternatively, it is possible to manipulate the image sequence by simple operations. For example, in Figure 11(b) some objects were not decoded and used for reconstruction, while others were decoded and displayed using subsequent scaling, rotation, or translation. The scaling, rotation, and translation parameters employed for manipulation of the image sequence can be altered in the bit stream by means of simple bit-stream editing operations—without the need for further transcoding. In addition, new objects can be included that did not belong to the original scene—or original objects may be neglected. Since the bit stream of the sequence is organized in an "object layered" form, the inclusion or deletion of additional objects in the image sequence is performed on the bit-stream level by adding/deleting the appropriate object bit-streams—again, without the need for further transcoding. It is targeted to provide to the user the different video objects also with various scales of quality, size, or frame rates to assist the flexible presentation of the data.

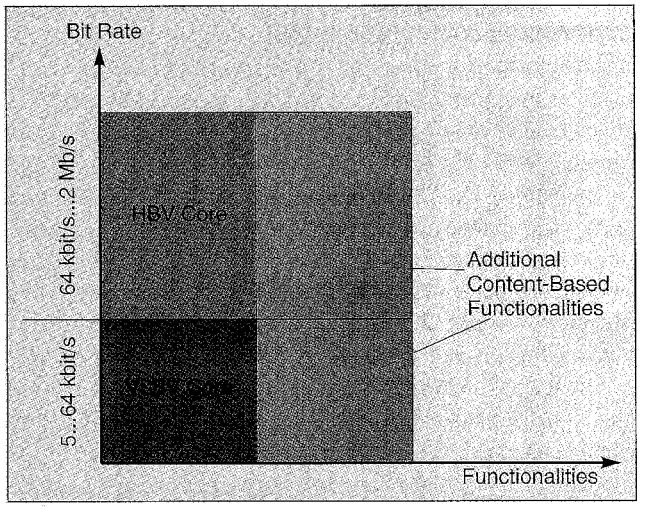
The standardization process has become significantly more efficient and faster.

Audio and Speech

Similar to the content-based functionalities outlined for the video applications above, MPEG-4 is targeted to provide object-layered audio bit streams to assist the access to and manipulation of content in audio and speech sequences. An example of an application is the content-based audio coding of a violin concerto played by an orchestra where, on demand, it is possible to extract and enhance the sound of single instruments. Alternatively, the concerto may be replayed with or without the solo violin.

Synthetic Natural Hybrid Coding (SNHC)

The standard is envisioned to provide the above capabilities for both synthetic (S) and natural (N) audio-visual objects as well as for hybrid (H) coding (C) and representation of natural and synthetic objects. As an example, it is targeted to allow the coding and generation of text overlays based on graphics primitives. This would greatly reduce the bits needed to store and transmit text and allow a high degree of flexibility for representing or altering text—e.g., it will be possible to select various types and fonts for display in Fig. 11. Other functionalities envi-



▲ *12. Structure of the MPEG-4 video-coding standard.*

sioned are the efficient coding of computer-animated, texture mapped wire-grid faces and human bodies.

Systems

The MPEG-4 architecture will allow the separate coding of audio or video objects, natural or synthetic, and the appropriate multiplexing of the separate object elementary streams into a single bit stream. Similar to the MPEG-1 and MPEG-2 standards, a MPEG-4 "Systems" standard will be developed to assist multiplexing of elementary streams, synchronization, and packetization. Additionally, as described above, the MPEG-4 systems multiplex will provide the basic representation/manipulation parameters (translation, rotation, or zoom of an object in relation to reference coordinates and scales) in the header of the bit-stream layer of each object.

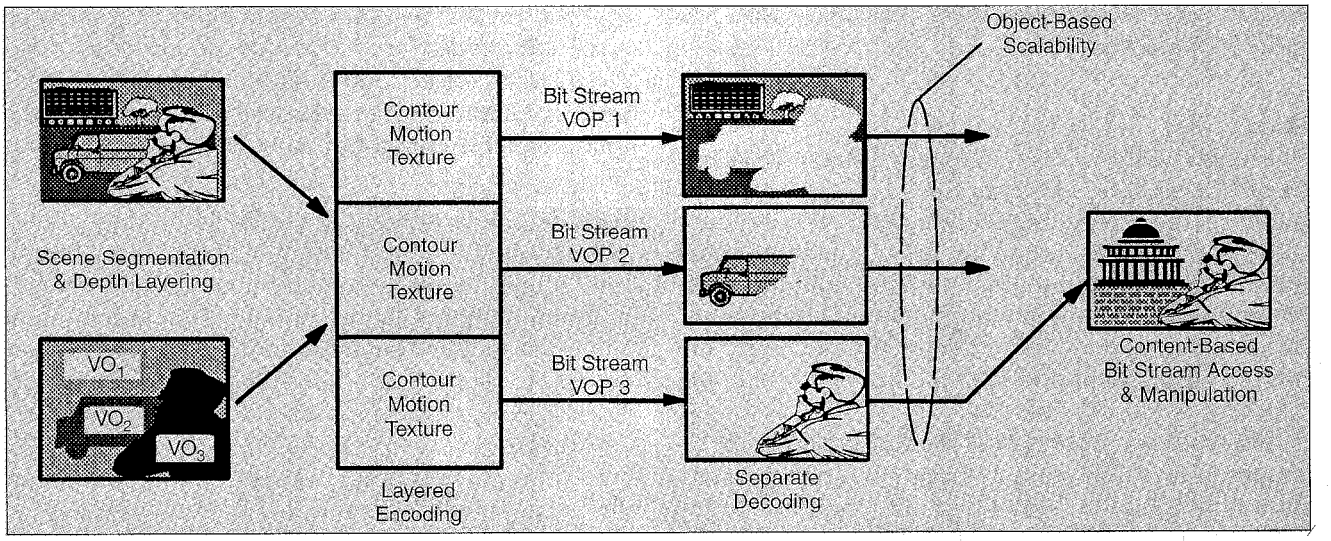
The MPEG-4 Video-Coding Standard

The MPEG-4 video-coding algorithms will eventually support all functionalities already provided by MPEG-1 and MPEG-2, including the provision to efficiently compress standard rectangular-sized image sequences at varying levels of input formats, frame rates, and bit rates. The content-based functionalities also will be assisted.

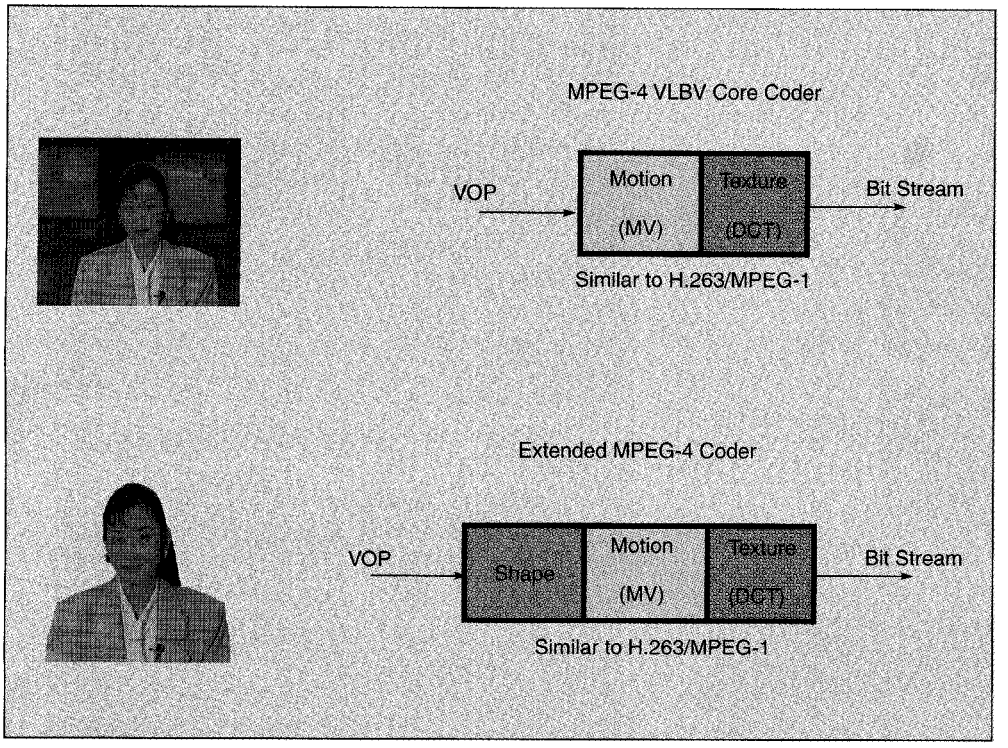
A basic classification of the bit rates and functionalities currently provided by the MPEG-4 video-coding model under development is depicted in Fig. 12, with the attempt to clusters bit-rate levels vs. sets of functionalities and basic applications profiles. At the bottom end a "VLBV core" (VLBV: very low bit rate video) provides algorithms and tools for applications operating at bit rates between 5-64 kbits/s, supporting image sequences with lower spatial resolutions (from a few pixels per line and row up to common intermediate format (CIF) resolution (352 × 288 pels)) and lower frame rates (ranging from 0 Hz for still images to 15 Hz). The basic applications-specific functionalities supported by the VLBV core include a) conventional VLBV coding of rectangular-size image sequences with high coding efficiency and high error robustness/resilience, low latency and low complexity for real-time multimedia communications applications, and b) provisions for "random-access" and FF/FR operations for multimedia data-base storage and access applications.

The same basic functionalities outlined above are also supported by a higher bit rate video core (HBV, maximum bit rates probably around 4 Mb/s) with a higher range of spatial and temporal input parameters up to R.601 resolutions—employing identical algorithms and tools as the VLBV core. The reader is referred to references [16] and [17] for a more detailed description of the techniques under development for the VLBV and HBV cores.

At the heart of the additional "content-based" functionalities is the support for the separate encoding and decoding of content (i.e., physical objects in a scene) as already discussed with Fig. 11. Within the context of MPEG-4 this functionality—the ability to identify and selectively decode and reconstruct video content of interest—is referred to as "Content-Based Scalability." This



▲ *13. The "object-layered" coding approach taken by the MPEG-4 video-coding standard.*



▲ *14. MPEG-4 VLBV core and the extended core coder.*

MPEG-4 feature provides the most elementary mechanism for interactivity and manipulation with/of content of images or video in the compressed domain without the need for further segmentation or transcoding at the receiver. The extended MPEG-4 algorithms and tools for content-based functionalities can be seen as a superset of the VLBV and HBV cores—thus, the tools provided by the VLBV and HBV cores are complemented with additional elements [17].

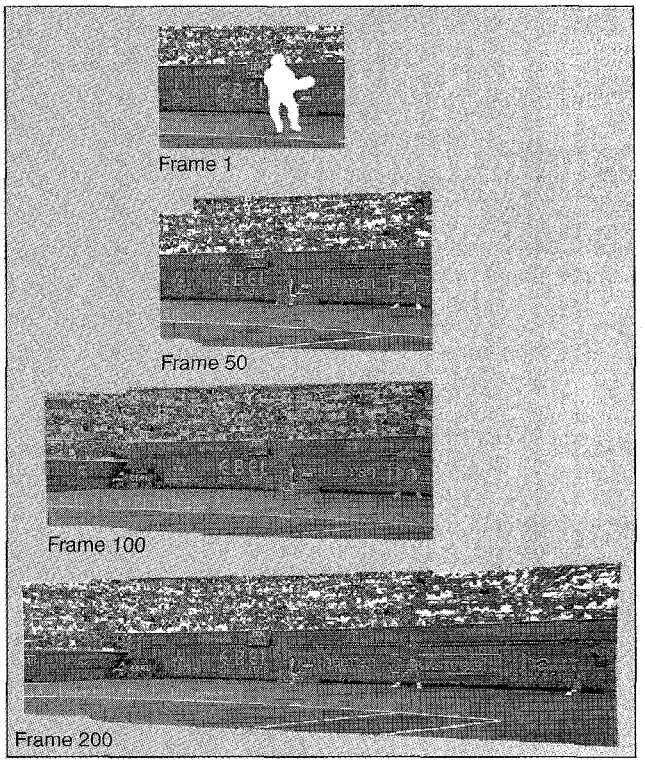
To enable the envisioned content-based interactive functionalities, the MPEG-4 video standard introduces the concept of video object planes (VOPs). This concept is illustrated in Fig. 13. It is assumed that each frame of an input video sequence is segmented into a number of arbitrarily shaped image regions (video object planes)—each of the regions may possibly cover particular image or video content of interest, i.e., describing physical objects or content within scenes. In contrast to the video source format used for the MPEG-1 and MPEG-2 standards, the video input to be coded by the MPEG-4 Verification Model is thus no longer considered a rectangular region. The input to be coded can be a VOP image region of arbitrary shape, and the shape and location of the region can vary from frame to frame. Successive VOPs belonging to the same physical object in a scene are referred to as video objects (VOs)—a sequence of VOPs of possibly arbitrary shape and position. The shape, motion, and texture information of the VOPs belonging to the same VO is encoded and transmitted or coded into a separate VOL (video object layer). In addition, relevant information needed to identify each of the VOLs—and how the various VOLs are composed at the receiver to reconstruct the entire original sequence—is also included in the bit stream. This allows the separate decoding of each VOP and the required flexible manipulation of the video sequence as indicated in Figs. 11 and 13. Notice that the video source input assumed for the VOL structure either already exists in terms of separate entities (i.e., generated with chroma-key technology) or is generated by means of on-line or off-line segmentation algorithms.

Notice that MPEG-4 images as well as image sequences are generally considered to be arbitrarily shaped—in contrast to the standard MPEG-1 and MPEG-2 definitions, which encode rectangular-size image sequences. The MPEG-4 content-based approach can be seen as a logical extension of the conventional MPEG-4 VLBV and HBV core coding approach toward image input sequences of arbitrary shape. In particular, if the original input image sequences are not decomposed into several VOLs of arbitrary shape, the coding structure simply degenerates into a single-layer representation that supports coding of conventional image sequences of rectangular shape.

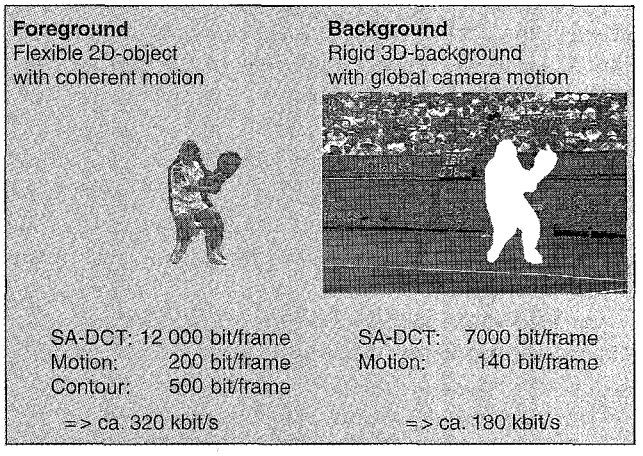
As illustrated in Fig. 14, the MPEG-4 video standard will support the coding of rectangular-size image sequences, which is similar to conventional MPEG-1/2 coding approaches and involves motion prediction/compensation followed by DCT-based texture coding. For the content-based functionalities where the image-sequence input is of arbitrary shape and location, this approach is extended by also coding shape and transparency information.

The MPEG-4 Sprite Coding Technology

A number of tools are being investigated within the MPEG-4 video development that attempt to provide higher quality compared to MPEG-1 and MPEG-2 as well as additional content-based functionalities for sequences with restricted content. An interesting example is the MPEG-4 "Sprite" prediction [18, 20, 21]. The "Sprite" coding allows the efficient transmission of background scenes where the changes within the background content is mainly caused by camera motion. Thus, a static sprite is a possibly large still image (i.e., static and flat background panorama) that is transmitted to the receiver



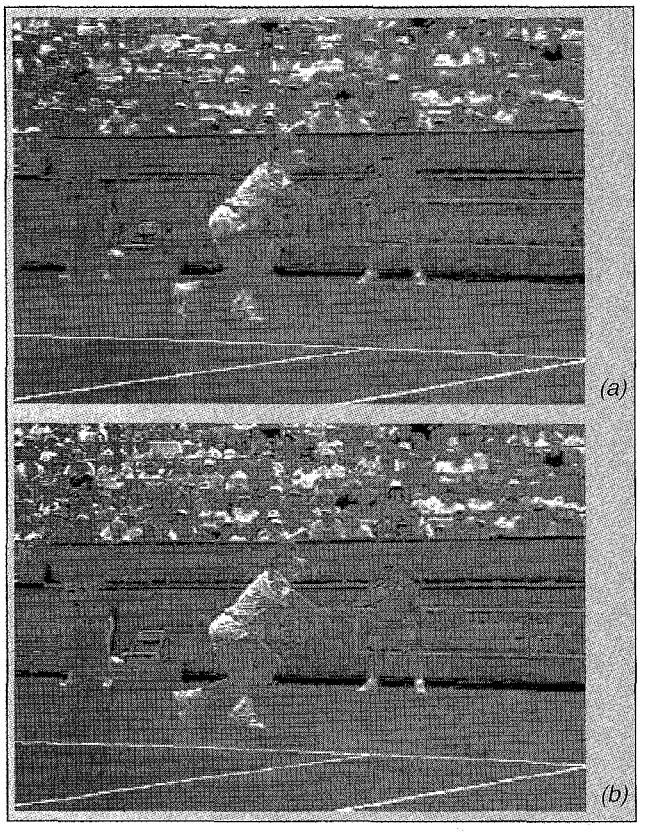
▲ *15. Sprite background generation.*



▲ *16. Foreground tennis player and separate sprite coded background.*

first and then stored in a frame store at both encoder and decoder. The camera parameters are transmitted to the receiver for each frame so that the appropriate part of the scene can be mapped (or warped—including zoom, rotation, and translation within the Sprite image) at the receiver for display.

Consider the case that for a given video sequence the content in a scene can be separated into foreground object(s) and a (static) background Sprite. This may be done off-line by analysis of the content of a scene prior to coding. Figure 15 illustrates the Sprite (background) generation for a video sequence that contains a tennis match with high camera motion and texture. One tennis player is moving in front of a background scene. Starting from frame 1, through successive image analysis and with the



▲ *17. Sequence coded using MPEG-1 (a) and MPEG-4 with sprite technology (b) at approximately 1 Mb/s.*

help of the camera motion, the final Sprite background image is derived in frame 200. Notice that the Sprite generation is not standardized, since it can be seen as a post-processing tool.

Using the MPEG-4 Sprite coding technology the foreground content can be coded and transmitted separately from the receiver. If the background is static, only one frame needs to be transmitted at the beginning of a scene (i.e., frame 200 in Fig. 15) plus the camera parameters. The receiver composes the separately transmitted foreground and background to reconstruct the original scene. Figure 16 illustrates this concept using the example above. The foreground object tennis player is coded separately from the background as an object of arbitrary shape. The background (Figure 16, right) is reconstructed from the Sprite background image in Fig. 15 stored at the decoder. Only eight motion parameters were transmitted to the receiver to indicate which part of the Sprite is being used under what kind of perspective transformation. Only a few bits are being spent for the background information.

The coding gain using the MPEG-4 Sprite technology over existing compression technology appears to be substantial in the example given above (depicted in Fig. 17). Notice, however, that the technique described cannot be seen as a tool that is easily applied to generic scene content. The gain described above can only be achieved if substantial parts of a scene contain regions where motion is described by simple motion models—and if these re-

gions can be extracted from the remaining parts of the scene by means of image analysis and postprocessing. This certainly is an assumption that can be considered feasible to improve video quality for multimedia database applications but most certainly not for broadcast applications where on-line processing and coding is a necessity.

Discussion

International standardization in image coding has made a remarkable evolution from a committee-driven process dominated by telecoms and broadcasters to a market-driven process incorporating industries, telecoms, network operators, satellite operators, broadcasters, and research institutes. With this evolution the actual work of the standardization bodies also has changed considerably and has evolved from discussion circles of national delegations into international collaborative R&D activities. The standardization process has become significantly more efficient and faster—the reason is that standardization has to follow the accelerated speed of technology development because, otherwise, standards are in danger of becoming obsolete before they are agreed upon by the standardization bodies.

It has to be understood that video-coding standards have to rely on compromises between what is theoretically possible and what is technologically feasible. Standards can only be successful in the market-place if the cost-performance ratio is well balanced. This is specifically true in the field of image and video coding where a large variety of innovative coding algorithms exist, but may be too complex for implementation with state-of-the-art VLSI technology.

In this respect, the MPEG-1 standard provides efficient compression for a large variety of multimedia terminals with the additional flexibility provided for random access of video from storage media and for supporting a diversity of image source formats. A number of MPEG-1 encoder and decoder chip sets from different vendors are currently available on the market. Encoder and decoder PC boards have been developed using MPEG-1 chip sets. A number of commercial products use the MPEG-1 coding algorithm for interactive CD applications, such as the CD-I product.

The MPEG-2 standard is becoming more and more successful because there is a strong commitment from industries, cable and satellite operators, and broadcasters to use this standard. Digital TV broadcasting, pay TV, pay-per-view, video-on-demand, interactive TV, and many other future video services are the applications envisaged. Many MPEG-2 MAIN Profile at MAIN Level decoder prototype chips are already developed. The world-wide acceptance of MPEG-2 in consumer electronics will lead to large production scales making MPEG-2 decoder equipment cheap and therefore also attractive for other related areas, such as video communications and storage and multimedia applications in general.

The scope and potential of the forthcoming MPEG-4 standard was discussed in the context of future audio-visual multimedia communications environments. The MPEG-4 standard will provide tools and algorithms for coding both natural and synthetic video, audio, and speech data—and provisions to represent the data at the user terminal in a highly flexible way.

Acknowledgment

The MPEG video standards are developed in a collaborative effort involving some 200 video experts from companies and research institutes from North America, Europe, Asia, and Australia who meet four times a year in various places around the world. The outstanding input and effort of these experts form the basis of the technology discussed in the context of this article. The MPEG-1 video standard and the major part of the MPEG-2 video-coding standards have been developed under the chairmanship of Dr. Didier LeGall from C-Cube Microsystems, USA, who has been instrumental in the success of the standards.

The author would like to thank colleagues Peter Kauff and Jan DeLameillieure from HHI, Germany, for providing the experimental results presented in the article.

Dr. Thomas Sikora is a Research Manager in the Image Processing Department of the Heinrich-Hertz Institute (HHI) for Communication Technology in Berlin, Germany. He has served as the Video Chairman of the Moving Pictures Expert Group since 1995.

References

1. W. Chen and D. Hein, "Motion Compensated DXC System," in *Proceedings of 1986 Picture Coding Symposium*, Vol. 2-4, pp. 76-77, Tokyo, April 1986.
2. B.R. Halhed, "Videoconferencing Codecs: Navigating the MAZE," *Business Communication Review*, Vol. 21, No. 1, pp. 35-40, 1991
3. C.-F.Chen and K.K.Pang, "The Optimal Transform of Motion-Compensated Frame Difference Images in a Hybrid Coder," *IEEE Trans. Circuits and Systems - II: Analog and Digital Signal Processing*, pp. 289-296 September 1963.
4. R.Schäfer and T. Sikora, "Digital Video Coding Standards and Their Role in Video Communications," *Proceedings of the IEEE* Vol. 83, pp. 907-923, 1995.
5. N. Ahmed, T. Natrajan and K.R. Rao, "Discrete Cosine Transform," *IEEE Trans. on Computers*, Vol. C-23, No. 1, pp. 90-93, December 1984.
6. ISO/IEC 11172-2, "Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s - Video," Geneva, 1993.
7. J. De Lameillieure and R. Schäfer, "MPEG-2 Image Coding for Digital TV," *Fernseh und Kino Technik*, 48. Jahrgang, pp. 99-107, March 1994 (in German).
8. C. Gonzales and E.Viscito, "Flexibly scalable digital video coding," *Signal Processing: Image Communication*, Vol. 5, No. 1-2, February 1993.
9. T.Sikora, T.K. Tan and K.N. Ngan, "A performance comparison of frequency domain pyramid scalable coding schemes," *Proc. Picture Coding Symposium*, Lausanne, pp. 16.1 - 16.2, March 1993.

10. A.W. Johnson, T. Sikora, T.K. Tan, and K.N. Ngan, "Filters for Drift Reduction in Frequency Scalable Video Coding Schemes," *Electronic Letters*, Vol. 30, No.6, pp. 471-472, 1994

11. ISO/IEC JTC1/SC29/WG11 N0702 Rev, "Information Technology - Generic Coding of Moving Pictures and Associated Audio, Recommendation H.262," Draft International Standard, Paris, 25 March 1994.

12. J. De Lameilieure and G. Schamel, "Hierarchical Coding of TV/HDTV within the German HDTVT Project," Proc. Int. Workshop on HDTV'93, pp. 8A.1.1 - 8A.1.8, Ottawa, Canada, October 1993.

13. A. Puri and A. Wong, "Spatial Domain Resolution Scalable Video Coding," *Proc. SPIE Visual Communications and Image Processing*, Boston, MA, November 1993.

14. P.J. Burt and E. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE Trans. COM*, Vol. COM-31, pp. 532-540, 1983

15. L.Chiariglione, "MPEG and Multimedia Communications," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 7, No. 1, pp. 5-18, Feb. 1997.

16. T. Sikora, "The MPEG-4 Video Standard Verification Model," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 7, No. 1, pp. 19-31, Feb.1997.

17. T. Sikora, "MPEG-4 Very Low Bit Rate Video," *Proc. IEEE ISCAS Conference*, Hong Kong, June 1997.

18. T. Sikora and L. Chiariglione, "MPEG-4 Video and its Potential for Future Multimedia Services" *Proc. IEEE ISCAS Conference*, Hong Kong, June 1997.

19. T. Sikora, "MPEG-4 and Beyond - When Can I Watch Soccer on ISDN," *Proc. of the Montreux International Television Symposium - Future Technology Forum*, Montreux, June 1997.

20. M.-C. Lee, W. Chen, C.B. Lin, C. Gu, T. Markoc, S.I. Zabinsky and R. Szeliski, "A Layered Video Object Coding System Using Sprite and Affine Motion Model," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 7, No. 1, pp. 130-145, Feb. 1997.

21. P. Kauff, B. Makai, S. Rauthenberg, U. Gölz, J.L.P. DeLameillieure and T. Sikora, "Functional Coding of Video Using a Shape-Adaptive DCT Algorithm and Object-Based Motion Prediction Toolbox," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 7, No. 1, pp. 181-196, Feb.1997.