Trends and Perspectives in Image and Video Coding

THOMAS SIKORA, SENIOR MEMBER, IEEE

Invited Paper

The objective of the paper is to provide an overview on recent trends and future perspectives in image and video coding. Here, I review the rapid development in the field during the past 40 years and outline current state-of-the art strategies for coding images and videos. These and other coding algorithms are discussed in the context of international JPEG, JPEG 2000, MPEG-1/2/4, and H.261/3/4 standards. Novel techniques targeted at achieving higher compression gains, error robustness, and network/device adaptability are described and discussed.

Keywords*—Discrete cosine transform (DCT), distributed source coding, embedded coding, error concealment, image coding, International Telecommunication Union-Telecommunications (ITU-T), Joint Photographic Experts Group (JPEG), motion compensation, JPEG 2000, Motion Picture Experts Group (MPEG), standardization, video coding, wavelets.*

I. INTRODUCTION

Modern image and video compression techniques today 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 [1]. Digital image and video coding research started in the 1950s and 1960s with spatial DPCM coding of images. In the 1970s, transform coding techniques were investigated. In 1974, Ahmed *et al.* [9] introduced the famous block-based discrete cosine transform (DCT) strategy. Motion compensated prediction error coding also started in the 1970s and matured into practical technology around 1985 with the advent of the basic hybrid block-based motion compensation/DCT systems (MC/DCT). MC/DCT coding strategies are implemented in all of today's MPEG and ITU video coding algorithms [2]–[8]. MC/DCT technology provided a significant compression gain versus pure INTRA frame DCT coding (i.e., JPEG) for video compression. However, much complexity is added to the encoder who has to perform motion estimation and compensation (MC), as depicted in Fig. 1. Recent extensions of the basic MC/DCT approach (i.e., those standardized with H.263, MPEG-1/2/4, H.264) have further improved compression efficiency at the expense of more complex decoders and even more complex encoders.

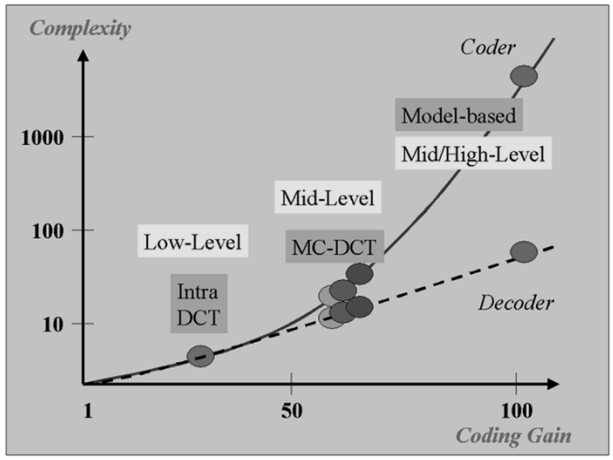

Fig. 1. Complexity of various video coding strategies versus coding gain.

Discrete wavelet transforms (DWT) were applied to image and video coding starting in the 1980s and now provide the core technology for the MPEG-4 texture coding standard and JPEG 2000 still image coding standard.

Intense research is being carried out with the goal to make image and video compression algorithms more efficient, more flexible, and more robust against bit and packet errors. Much research is focused toward the recognition of meaningful, possibly semantic, information in images for advanced motion prediction. The video coding research is moving from blind low-level computer vision approaches (DCT) via semiblind midlevel (MC/DCT) to high-level computer vision strategies. It is expected that future segmentation and model-based solutions will require much

Manuscript received January 18, 2004; revised July 21, 2004.

The author is with the Technical University of Berlin, Berlin D-10587, Germany (e-mail: www.nue.tu-berlin.de).

Digital Object Identifier 10.1109/JPROC.2004.839601

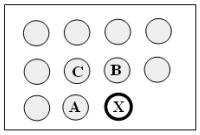

Fig. 2. Predictive coding of image pixels. X: pixel to be predicted and coded. A, B, C: already coded and stored or transmitted pixels. JPEG lossless coding standard specifies the following possible predictors: $\mathrm{X} = \mathrm{A}, \mathrm{X} = \mathrm{B}, \mathrm{X} = \mathrm{C}, \mathrm{X} = \mathrm{A} + \mathrm{B} - \mathrm{C}, \mathrm{X} = \mathrm{A} + (\mathrm{C} - \mathrm{B})/2, \mathrm{X} = (\mathrm{A} + \mathrm{C})/2, \mathrm{X} = \mathrm{C} + (\mathrm{A} - \mathrm{B})/2$.

increased computational processing, both at the encoder and decoder—hopefully with mature technology and significantly increased compression gains as forecasted in Fig. 1.

The purpose of this paper is to outline the basic coding strategies that enable today's successful image and video applications—as well as to highlight selected research frontiers in the field that pave the development toward higher compression gains or extended functionalities.

II. Basic Strategies for Coding Images and Video

Dependent on the applications requirements, we may envisage "lossless" and "lossy" coding of the video data [10], [1]. 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 is required to be bit-identical to the image prior to *en*coding. The lossless coding mode of JPEG is an example of a lossless coder strategy. In contrast, the aim of "lossy" coding techniques—and this is relevant to the applications envisioned by lossy compression standards such as lossy JPEG, MPEG, and ITU—is to meet a given target bit rate for storage and transmission. Decoders reconstruct images and video with reduced quality. Quality requirements are balanced with bit-rate requirements. Important applications comprise image and video storage and transmission both for low-quality Internet applications as well as for high-quality applications such as digital TV.

A. Predictive Coding of Images

The purpose of predictive coding strategies is to decorrelate adjacent pel information and to encode a prediction error image rather than the original pels of the images. Images usually contain significant correlation in spatial directions [11]. Video sequences in both spatial and temporal directions.

Predictive coding is very efficient for removing correlation between pels prior to coding [10]. To this end, a pel value to be coded is predicted from already coded and transmitted/stored adjacent pel values—and only the small prediction error value is coded. Note that predictive coding is suitable for both lossless and lossy coding.

Fig. 2 depicts a predictive coding strategy for images, which is in fact identical to the one employed by the lossless coding mode of the JPEG standard [5]. A pel to be coded is predicted based on a weighted combination of values of the most adjacent neighboring pels. A JPEG lossless coder has to decide on one of the predictor's outlines with Fig. 2. The predictor is stored/transmitted with the bit stream. Thus, the decoder can perform the same prediction from the already decoded pels and reconstruct the new pel value based on the predicted value and decoded prediction error.

Fig. 3. Predictive coding of test image "Lenna." Top left: original image. Top right: 2-D prediction. Bottom left: 1-D prediction, $\mathrm{B} = \mathrm{C} = 0$. Bottom right: 1-D prediction, $\mathrm{A} = \mathrm{C} = 0$.

Fig. 3 illustrates the prediction error images achievable using one-dimensional (1-D) as well as two-dimensional (2-D) predictors. It is apparent that the prediction error image that needs to be coded contains much less amplitude variance compared to the original—this results in significantly reduced bit rate [10].

B. Transform Domain Coding of Images and Video

Transform coding is a strategy that has been studied extensively during the past two decades and has become a very popular compression method for lossy still image and video coding. The purpose of transform coding is to quantize and encode decorrelated transform coefficients rather than the original pels of the images. As such, the goal is identical to predictive coding strategies described above. The most popular and well-established transform techniques are the celebrated DCT [9] used in the lossy JPEG, MPEG, and ITU coder standard and the DWT [4], [7], [12] standardized in MPEG-4 and JPEG 2000. The DCT is applied in these standards strictly as a block-based approach usually on blocks of size 8×8 pels. The DWT in contrast is usually implemented in JPEG 2000 and MPEG-4 as a frame-based approach applied to entire images; block partitions are also possible [4], [7].

For video compression, transform coding strategies are usually combined with motion compensated prediction into a hybrid MC/transform approach to achieve very efficient coding of video. We will describe these techniques in Sections II-C and II-D in more detail.

1) Discrete Cosine Transform (DCT) for Coding Images: For the block-based DCT transform approach, the input images are split into disjoint blocks of $N \times N$ pels $\underline{I}$ (e.g., of size 8×8 pels) as indicated in Fig. 4. In general, a linear, separable, and unitary *forward* 2-D-transformation strategy can be represented as a matrix operation on each

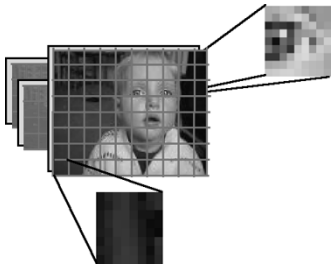

Fig. 4. Decomposition of images into adjacent, nonoverlapping blocks of $N \times N$ pels for transform coding (i.e., with the JPEG lossy coding standard). Color information in images is usually separated into RGB or YUV color images; UV components often subsampled and coded separately.

block matrix $\underline{I}$ using a $N \times N$ transform matrix $\underline{A}$—to obtain the $N \times N$ transform coefficients $\underline{C}$

$$\underline{C} = \underline{A}\underline{I}\underline{A}^T. \tag{1}$$

Here, $\underline{A}^T$ denotes the transpose of the 1-D-transformation matrix $\underline{A}$ [11]. The transformation is reversible as long as the transform matrix is invertible and the transform coefficients are not quantized, $\underline{C}^* = \underline{C}$. For a unitrary transform the inverse matrix $\underline{A}^{-1}$ is identical with the transposed matrix $\underline{A}^T$, that is $\underline{A}^{-1} = \underline{A}^T$. The unitrary transformation is reversible, since the original $N \times N$ block of pels $\underline{I}$ can be reconstructed using a linear and separable *inverse* transformation. The reconstruction can then be described as

$$\underline{I} = \underline{A}^T \cdot \underline{C}^* \cdot \underline{A} = \sum_{k=1}^{N} \sum_{l=1}^{N} \underline{B}(k,l) \cdot C^*(k,l). \tag{2}$$

$\underline{B}(k,l)$ is a basis-image with index (k,l) of size $N \times N$. In a practical coding scheme, the coefficients will be quantized ($\underline{C}^* \neq \underline{C}$) and the original image block $\underline{I}$ is approximated as a weighted superposition of basis-images $\underline{B}(k,l)$, weighted with the quantized transform coefficients $C^*(k,l)$. The coding strategy thus results in a lossy reconstruction—the coarser the quantization the less bits required to store/transmit the image, the worse the reconstruction quality at the decoder.

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 so far. Fig. 5 depicts the basis-images $\underline{B}(k,l)$ of the 8×8 DCT that are used for reconstruction of the images according to (2). In most standards coding schemes, i.e., lossy JPEG, each 8×8 block of coefficients is quantized and coded into variable or fixed length binary code words [11].

Today, block-based DCT transform strategies 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.

The DCT is a so-called compact transform, because most signal energy is compacted into the lower frequency coefficients. Most higher coefficients are small or zero after quantization, and small or zero-valued coefficients tend to

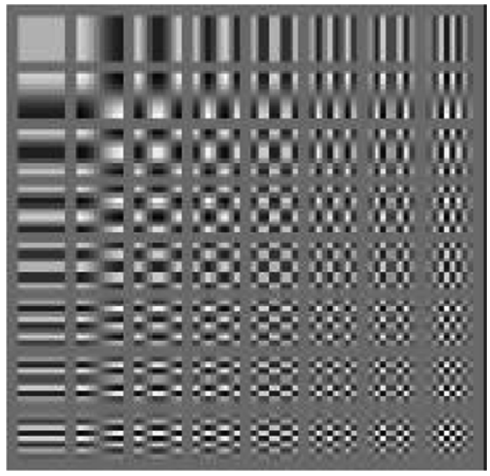

Fig. 5. The 64 basis images $\underline{B}(k,l)$ of the 2-D 8×8 DCT. A block-based DCT decoder, such as JPEG, reconstructs the stored or transmitted image blocks as a weighted superposition of these basis images, each weighted with the associated decoded DCT coefficient $C^*(k,l)$.

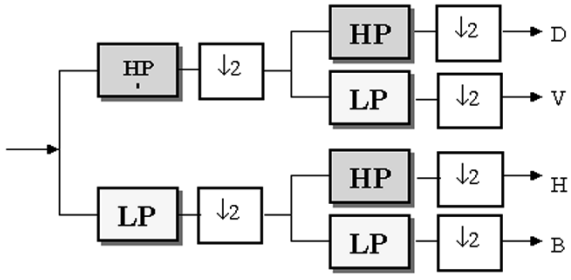

Fig. 6. One stage of 1-D DWT decomposition, composed of low-pass (LP) and high-pass (HP) filters with subsequent subsampling.

be clustered together [10], [13]. Thus, on average only a small number of quantized DCT coefficients need to be transmitted to the receiver to obtain a good approximated reconstruction of the image blocks based on the basis images in Fig. 5 and (2). At very low bit rates, only few coefficients will be coded and the well-known DCT block artifacts will be visible (i.e., in low-quality JPEG or MPEG images and video).

2) Discrete Wavelet Transform (DWT): The DWT provides the key technology for the JPEG 2000 and the MPEG-4 still image coding standard and is applied to the entire image rather than to separate blocks of pels [7].

A DWT may be implemented as a filter bank as illustrated in Fig. 6 and a suitable choice of filters may enable perfect reconstruction in the reverse process. The example filter bank decomposes the original image into horizontal (H), vertical (V), diagonal (D), and baseband (B) subband images, each being one-fourth the size of the original image. Multiple stages of decomposition can be cascaded together to recursively decompose the baseband. The subbands in this case are usually arranged in a pyramidal form as illustrated in Fig. 7 (two stages). Similar to the linear DCT approach, most signal energy is compacted by the DWT into the lower-frequency subbands; most coefficients in higher subbands are

Fig. 7. Two-scale 2-D DWT of an image in a pyramidal arrangement of the subbands.

small or zero after quantization and small or zero-valued coefficients tend to be clustered together. Also, clusters of small or zero-valued coefficients tend to be located in the same relative spatial position in the subbands.

There is good evidence that DWT coding provides improved coding gains compared to DCT strategies. Most importantly, the DWT enables in combination with embedded quantizers—alongside with excellent compression efficiency—so-called fine-granularity embedded coding functionalities fully integrated into the coder [7]. DWT embedded coding enables to reconstruct images progressively in fine-granular stages using partial bit-stream information, a capability in high demand in modern applications, yet absent from prior nonembedded standards. Embedded coding is discussed in more detail in Section V-A. DWT embedded image coders are essentially built upon three major innovative components: the DWT, successive approximation quantization, and significance-map encoding (i.e., using zero-trees [14]). It is also important to note that in JPEG 2000 the DWT is employed both in the lossless and lossy coding mode.

C. Predictive Coding of Video

For video sources, it is assumed that pels in consecutive video images (frames of the sequence) are also correlated. Moving objects and part of the background in a scene then appear in a number of consecutive video frames—even though possibly displaced in horizontal and vertical direction and somehow distorted when motion of objects or camera motion/projection is not purely translatory. Thus, the magnitude of a particular image pel can be predicted from nearby pels within the same frame. Even more efficiently from pels of a previously coded frame—so-called motion compensated (MC) prediction [11]. This requires that one or more coded frames are stored at encoder and decoder.

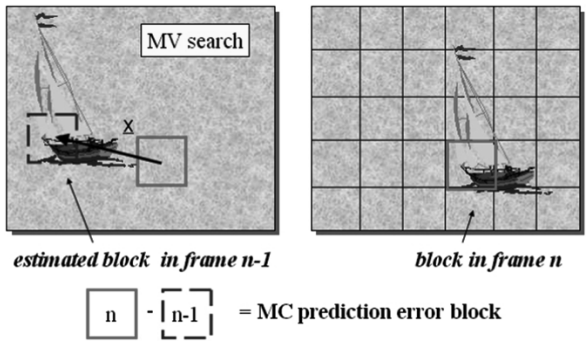

Fig. 8. MPEG/ITU-type block matching approach for motion prediction and compensation: A motion vector $\underline{x}$ points to a reference block of same size in a previously coded frame $\mathrm{n}-1$. The motion compensated (MC) prediction error is calculated by subtracting each pel in a block with its motion shifted counterpart in the previous frame.

The strategy for predicting motion of objects in video sequences is vital for achieving high compression gains. Research on this subject has been the focus of intense research during the past 25 years and continues to be one of the prime areas where most advances in compression efficiency may be seen in the future. The most established and implemented strategy is the so-called block-based motion compensation technique employed in all international MPEG or ITU-T video compression standards described below [2]. Other strategies employ Global Motion compensation [4], SPRITE motion compensation [15], [16], segmentation-based [17], or object-based motion compensation [18].

The "block-based" motion compensation strategy is based on motion vector estimation outlined in Fig. 8 [2]. The images are separated into disjoint blocks of pels as shown in Fig. 4. The motion of a block of pels between frames is estimated and described by only one motion vector $\underline{x}$ (MV). This assumes that all pels have the same displacement. The motion vectors are quantized to pel-or subpel accuracy prior to coding. The higher the precision of the MVs, the better the prediction. The bit-rate overhead for sending the MV information to the receiver needs to be well balanced with the gain achieved by the motion compensated prediction. MPEG and ITU-T video coding standards employ 1-pel to $1/4$-pel MV accuracy for MC blocks of sizes between 4×4 and 16×16 pels. It is worth noting that motion estimation can be a very difficult and time-consuming task, since it is necessary to find one-to-one correspondence between pels in consecutive frames. Fig. 9 depicts the impressive efficiency of motion compensated prediction for a TV-size video sequence.

The purpose of block-based motion compensated prediction is to decorrelate the image signals prior to coding. Fig. 10 compares the autocorrelation function of MC prediction error and original image in a video sequence in Fig. 9. Even though the temporal prediction is very efficient, it is obvious that there is still remaining spatial correlation in the error image that can be eliminated using subsequent spatial transform or spatial prediction strategies. The DCT performs again close to optimum for the MCFD [13]. This

Fig. 9. Efficiency of MPEG/ITU-type video coding using block-based motion prediction. Top left: frame to be coded. Top right: estimated motion vectors $\underline{x}$ for each 16×16 block. Bottom right: motion compensated prediction error image. Bottom left: prediction error using simple frame difference (all motion vectors are assumed zero).

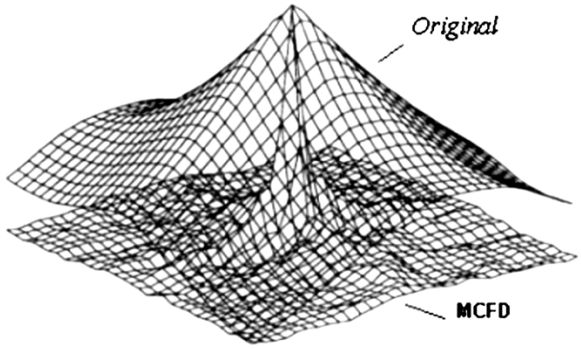

Fig. 10. 2-D spatial pel correlation function in blocks of the original image and prediction error image (MCFD, motion compensated frame difference) as reported in [13].

is the reason why most of today's standard video coding algorithms apply the DCT to MC prediction error images with good success—the so-called hybrid MC/DCT approach. The ITU-T H.264 standard employs an integer transform similar to the DCT.

D. Hybrid MC/DCT Coding for Video Sequences

The combination of temporal block-based motion compensated prediction and block-based DCT coding provides the key elements of the MPEG and ITU-T video coding standards [1], [2], [6]. For this reason, the MPEG and ITU-T coding algorithms are usually referred to as hybrid block-based MC/DCT algorithms. DWT coding so far has not shown significant compression gains versus DCT for video coding. The basic building blocks of such a hybrid video coder are depicted in Fig. 11—in a broad sense a JPEG coder plus block-based motion compensated prediction.

In basic MPEG and ITU-T video coding schemes, the first frame in a video sequence (I-picture) is encoded in INTRA mode without reference to any past or future frames [1], [2], [11]. At the encoder the DCT is applied to each $N \times N$ block (8×8 pels in MPEG standards) and, after output of the DCT, each of the $N \times N$ DCT coefficients are uniformly quantized (Q) and coded with Huffman code words of variable lengths (VLC). The decoder performs the reverse operations.

For motion predicted coding (P-pictures), the previously coded I- or P-picture frame $N-1$ is stored in a frame store (FS) in both encoder and decoder. Motion compensation (MC) is performed on a Macroblock basis (typically 16×16 pels in MPEG)—only one motion vector is estimated 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. A $N \times N$ DCT is then applied to each of the $N \times N$ blocks contained in the Macroblock followed by quantization (Q) of the DCT coefficients with entropy coding (VLC). The quantization step size (sz) can be adjusted for each Macroblock in a frame to achieve a given target bit rate.

E. Content-Based Video Coding

The coding strategies outlined above are designed to provide the best possible quality of the reconstructed images at a given bit rate. At the heart of "content-based" functionalities (i.e., ISO MPEG-4 standard) is the support for the separate encoding and decoding of content (i.e., physical objects in a scene) [1], [4]. This extended functionality provides the most elementary mechanism for flexible presentation of single video objects in video scenes without the need for further segmentation or transcoding at the receiver. Fig. 12 illustrates a virtual environment teleconference application example that makes extensive use of this functionality. Participants of the conference call at diverse locations are coded as arbitrarily shaped objects in separate streams. The receiver can place the objects flexibly into a virtual environment, i.e., to provide the impression that all participants are gathering around a table as in a normal conference situation.

The content-based approach can be implemented as an algorithmic extension of the conventional video coding approach toward image input sequences of arbitrary shape. In MPEG-4, this may be achieved by means of advanced shape coding algorithms and a mandatory low-complexity shape-adaptive DCT approach [19]–[22].

III. Coding Standards

A. JPEG and JPEG 2000

The JPEG still image coding standard (released in 1990) is the most widely employed compression algorithm for still color images today. JPEG finds applications in many diverse storage and transmission application domains, such as the Internet, digital professional and consumer photography and video. The standard handles very small image sizes as well as huge size images.

The lossless coding strategy employed by JPEG involves predictive coding as outlined in Fig. 2 using one of the standardized predictors per image as described with the figure.

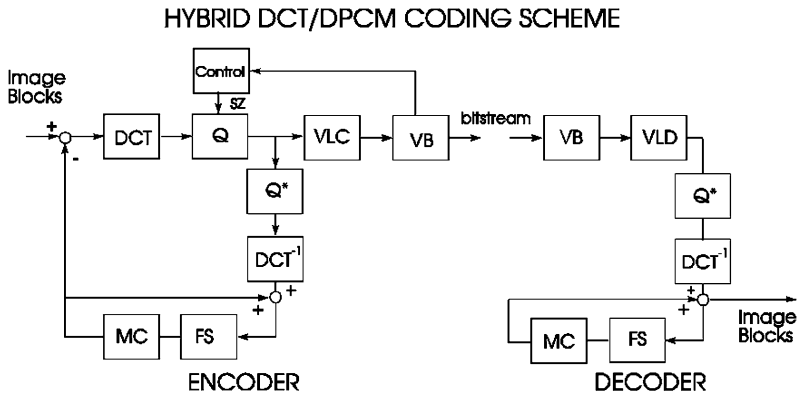

Fig. 11. Block diagram of a basic hybrid MC/DCT encoder and decoder structure (i.e., as used in MPEG/ITU-type video coders).

Fig. 12. Immersive virtual environment video conference setup (FhG HHI, Berlin, Germany). Persons from different remote locations are arranged around a virtual table. Each participant can be compressed as arbitrarily shaped video object using the MPEG-4 object-based coding modes.

The well-known lossy compression algorithm is based on DCT transform coding of image blocks of size 8×8, followed by quantization of the DCT coefficients and subsequent assignment of variable length binary code words to code position and amplitudes of DCT coefficients.

The JPEG 2000 standard was approved in 2002 and commercial deployment is still in its early stage. Compared to JPEG, it is probably the very efficient progressive coding functionality that sparks most interest in industry—a functionality due to the DWT that is not efficiently provided in JPEG. A compression performance comparison between JPEG and JPEG 2000 coding has been reported in [23] and indicates also superior compression gains by the JPEG 2000 standard in terms of peak signal-to-noise ratio (PSNR).

B. Video Coding Standards

Standardization work in the video coding domain started around 1990 with the development of the ITU-T H.261 standard—targeted at transmission of digital video-telephone signals over ISDN channels at data rates of $n \times 64$ kb/s $(n = 1, 2, \ldots, 30)$. Since then, a number of international video coding standards were released by ITU-T and ISO-MPEG standards bodies targeted at diverse application domains. Table 1 summarizes the basic features of the standards available today. The H.263 standard was released by ITU-T in 1995 for transmission of video-telephone signals at low bit rates and H.264 (as a joint development with ISO-MPEG as MPEG-4 AVC) in 2002. H.264 essentially covers all application domains of H.263 while providing superior compression efficiency and additional functionalities over a broad range of bit rates and applications.

MPEG-1 was released by ISO-MPEG in 1993 and is—together with MPEG-2—today the most widely introduced video compression standard. Even though MPEG-1 was primarily developed for storage of compressed video on CD, MPEG-1 compressed video is widely used for distributing video over the Internet. MPEG-2 was developed for compression of digital TV signals at around 4–6 Mb/s and has been instrumental in enabling and introducing commercial digital TV around the world. MPEG-4 was developed for coding video at very low bit rates and for providing additional object-based functionalities. MPEG-4 has found widespread application in Internet streaming, wireless video, and digital consumer video cameras as well as in mobile phones and mobile palm computers.

All the above standards build on the block-based motion compensated DCT approach outlined in Fig. 11. However, during the past ten years, various details of the basic approach were refined and resulted in more complex but also more efficient compression standards. It appears that significant compression gains have been achieved based on advanced motion vector accuracy and more sophisticated motion models—ITU-H.264 is presently the most advanced standard in terms of compression efficiency. In the H.264 (MPEG-4 AVC) standard, more precise motion compensation prediction using variable size MC blocks is employed along with context based arithmetic coding [24]. Another novelty is the introduction of long-term frame memories in H.264 that allows the storage of multiple frames of the past for temporal prediction. Needless to say, most of the advanced techniques are implemented with much increased complexity at encoder and decoder—a tendency which was already discussed with Fig. 1. However, the past ten years have also witnessed much improved processor speed and

Table 1
Basic Features of International Video Coding Standards

	MPEG-1 (1993)	MPEG-2 (1995)	MPEG-4 (2000)	H.261 (1993)	H.263 (1995)	H.264/ MPEG-4 AVC (2002)
Transform	8x8 DCT	8x8 DCT	8x8 DCT	8x8 DCT	8x8 DCT	4x4
MC Block Size	16 x 16	16 x 16 8x16	8 x 8, 16 x 16	16 x 16	8 x 8, 16 x 16	16x16, 16x8, 8x8, 8x4, 4x4
MC Accuracy	_ -pel	_ -pel	_ -pel	1-pel	_ -pel	1/8 -pel
Additional Motion Prediction Modes	- B-Frames	- B-Frames - Interlace	- B-Frames - Interlace - GMC (Global MC) - SPRITE Coding	-	- B-Frames	- B-Frames - Long term frame memory - in-loop deblocking filter - CAVLC/CABAC

improvements in VLSI design to allow real time implementation of the algorithms. Future coding strategies will meet even less complexity constraints.

IV. Compression Efficiency—From Pixels to Objects

The coding strategies outlined in Section II form the basis for the efficiency and functionalities of today's state-of-the-art coding techniques. Intense research is being carried out worldwide with the goal to further increase coding efficiency using mid and high level computer vision and image processing algorithms. Progress in the domain of motion prediction will be vital to achieve higher compression gains for coding of video sequences. In essence, understanding content (or even semantics) in image sequences will most likely provide the key for significant progress in the field. In the following, we will describe some of the strategies that extend the basic block-based motion prediction model toward segment-based and model-based approaches. Advances in texture coding will also be necessary for both advanced image and video coding.

A. Segmentation-Based Coding of Images and Video

The goal of segmentation-based coding of images is to achieve high compression gains. These techniques can be seen as a midlevel computer vision image processing approach that divide the image into regions of coherent textures [17], [25], [26]. Note that this is an extension of the block-based approach in that regions are now arbitrarily shaped. The shape of the regions need to be coded using shape coding algorithms. Popular shape coding algorithms include Quadtree decomposition [27] as well as shape coding strategies similar to the ones standardized with MPEG-4 [4], [19]. The potential advantage over block-based approaches is that significant edges may be better preserved and dedicated texture models may be employed (i.e., using the above shape-adaptive DCT).

The idea is extended for video coding toward segmentation-based motion prediction and compensation [17], [26]. Larger collections of segments that belong to the same object—or at least move coherently—can be described by one segment. One set of motion parameters can then be used to predict the motion of the segment between frames. The advantage over block-based MC approaches is that motion prediction can be more accurate. More sophisticated motion models with more than two parameters (such as affine or perspective and parabolic motion models with typically 6–12 motion parameters) can significantly improve the estimation accuracy. Again, the shape of the regions and extended motion parameters need to be transmitted, which in turn requires significant bit overhead. Excellent results for video coding have been reported using segmentation-based motion prediction [17].

B. SPRITE Coding of Video

The segmentation-based approach above can be extended toward the "SPRITE" coding approach—sometimes also referred to as "Panorama" coding [15], [16], [28]. SPRITE coding algorithms are also part of the MPEG-4 object-based coding strategy [1], [4]. The basic SPRITE approach assumes that images in video sequences can be decomposed into static background and moving foreground objects. Sprite coding reconstructs and transmits the background separately from the foreground using very sophisticated motion analysis and prediction strategies.

The background in video sequences is essentially assumed to be a more or less flat static region (like a flat wall with large still Panorama picture)—content in the Panorama picture is generally assumed to be constant. As the camera pans, zooms, and rotates over the scene, the SPRITE coder learns the appearance of the large background panorama picture by estimating the camera parameters between successive frames of the video sequence and mapping each frame background content into the Panorama picture using the estimated motion parameters. As such, the method can be seen as a midlevel to high-level computer vision strategy that attempts to identify coherent background in video [15].

Camera motion is usually estimated from the video sequence to be coded using computationally expensive iterative parameter estimation algorithms. Fig. 13 depicts one frame of the video test sequence "Stefan" and the large SPRITE background reconstructed from 300 frames (10 s)

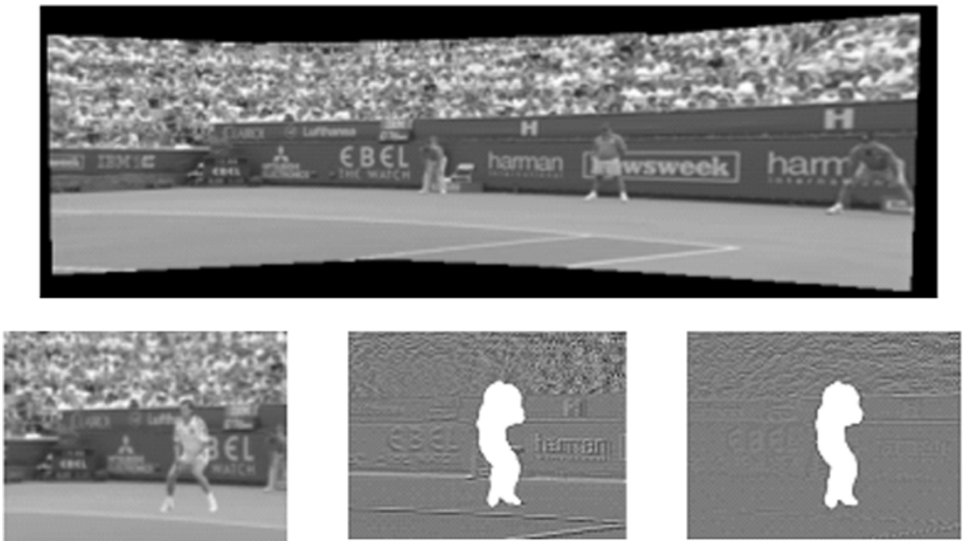

Fig. 13. SRITE coding of video. Top: the large background SPRITE image of the scene is reconstructed as the camera pans and zooms over the scene. Prediction of background between frames is performed using the SPRITE and the estimated camera parameters. Bottom left: a typical frame to be predicted. Bottom middle: MC prediction error using six-parameter affine motion model. Bottom right: error with 12-parameter parabolic model.

Fig. 14. Comparison of MPEG-4 SPRITE coded TV video (left) versus standard MPEG-4 block-based MC/DCT coding at 1 Mb/s (right).

of the video sequence using a parabolic motion model. Note that the segmentation/estimation algorithm assigned all objects that were either very small (audience) or that did not move significantly during the 10-s analysis stage to the background.

Using the MPEG-4 Sprite coding technology, the foreground content can be coded and transmitted separately from the receiver. The large static panorama picture is transmitted to the receiver first and then stored in a background frame store at both encoder and decoder side. The camera parameters are transmitted separately to the receiver for each frame, so that the appropriate part of the background scene can be reconstructed (using the motion parameters applied to the SPRITE image) at the receiver for display. The receiver composes the separately transmitted foreground and background to reconstruct the original scene.

The coding gain using the MPEG-4 SPRITE technology over existing block-based MPEG-4 compression technology appears to be substantial. Fig. 14 compares MPEG-4 SPRITE coding versus MPEG-4 at 1 Mb/s on the TV size test sequence. The overall subjective video quality is much improved. The coding gain is due to the fact that the background is transmitted only once and the background motion is described by only eight parameters/frame, instead of one MV/block.

SPRITE coding 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 coherent—and if these regions can be extracted from the remaining parts of the scene by means of image analysis and postprocessing.

C. Object-Based Coding of Video

Object-based coding strategies attempt to identify semantic objects in images, such as people in scenes, and represent their properties using sophisticated 2-D/3-D object

Fig. 15. 3-D model and SPRITE image of a person used for model-based coding (Source: P. Eisert, FhG HHI, Berlin).

models to improve coding efficiency. Object-based coding has been extensively investigated during the past 15 years in the context of video-telephone applications with one person in front of a static background—i.e., typical head and shoulder scenes [18], [29]–[31].

This approach can be seen as an extension of the SPRITE coding strategy toward a 3-D model of human head and shoulders. Fig. 15 depicts a suitable 3-D wire-grid model and the SPRITE that is mapped onto the 3-D surface to represent the texture details of the person [30]. Both encoder and decoder will have to use the same 3-D model and SPRITE for prediction—thus, the 3-D model and SPRITE will have to be sent to the decoder with the first frame. There is a very limited degree of freedom in how a person can move the body, head, mouth, and eyes. Few parameters need to be transmitted with such an approach to represent motion of the person.

On the other hand, such a rough model will not predict the pixel motion of all parts of the body with the accuracy required in common motion compensated prediction error coding. As a consequence, model-based coders usually only transmit prediction error signals for very few areas in the image, i.e., for eyes and mouth regions. Such an approach is well suited for very low bit-rate video coding applications, where it is not important that all parts of an image completely represent the original. Model-based prediction error coding approaches have also been investigated for higher bit-rate applications [18]. Transmission of 2–6 motion parameters/frame is sufficient to arrive at an excellent prediction of the face region [32].

V. Meeting Network and End-User Constraints

Compressed video is often transmitted over heterogeneous networks. Different networks have different characteristics. Some networks, i.e., packet networks such as the Internet, have variable bit rate and cannot guarantee quality of service for real-time video applications. Often significant bit-error rates cause bit errors or packet losses during transmission. All the above factors impact the end-to-end performance and thus the design of a video transmission system.

A. Network and Device Adaptability

In many applications, network and end-device resources are not known to the image or video coder. Examples include streaming/broadcasting of video over heterogeneous networks (i.e., the Internet). Both network and user device

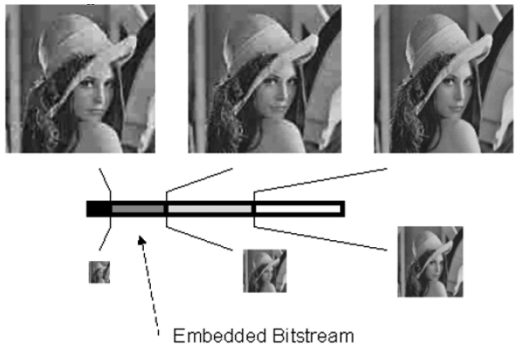

Fig. 16. Embedded hierarchical coding of images and video. Partial decoding of the bit stream allows reconstruction of lower quality or lower size images.

capabilities may vary drastically, i.e., in terms of network bandwidth, device display size, frame rate, power, and computing allocation. As a consequence, networks may not be able to transmit—or end-user devices may not be able to decode—the full bit stream. This will result in either no transmission/decoding at all or in the transmission or decoding of partial information only. Thus, severe packet loss and image impairment will be inevitable [33]. Even if the full bit stream may be decoded, small size terminals may not be able to display the information without subsequent downsizing spatial or temporal resolutions of images or video.

1) Embedded Coding of Images and Video: Many embedded coder strategies have been developed in the past to address this problem. The prime goal is to generate at the sender a suitable binary description of the image or video—to help the network or decoder to extract partial information that fits the allocated network bandwidth or end-user device constraints. Embedded coders generate a bit stream or a sequel of bit streams that each contain low-quality representations of the original image or video data, either in a hierarchical or nonhierarchical manner.

An excellent and very efficient example of a hierarchical, embedded image coder is the JPEG 2000 standard described in Section IV, which builds on the DWT decomposition. The bit stream is embedded in that it contains a large amount of smaller bit streams in a hierarchical way, each with a fraction of the total bit rate. By extracting only parts of the bit stream (few of the most important subband images) the decoder can reconstruct lower sizes or lower quality images, depending on the capabilities or priorities. In a typical Internet application, the end-user device may terminate a download session after the reconstructed image quality has reached a sufficient level. Fig. 16 illustrates this hierarchical approach toward image coding. In a hierarchical approach, it is important that lower quality bit-stream layers are decoded one after the other in a given progressive order, starting with the base layer.

An example of a nonhierarchical approach to embedded coding is multiple description coding [34]–[36]. The goal is to generate a number of bit-stream layers, each containing equally valid information about the source. Each layer bit stream can be used at the decoder to reconstruct a lower

quality/resolution image or video. The more layers the decoder receives, the better the quality of the reconstructed video. In contrast to the hierarchical approach, bit streams are not required to be decoded in a particular order.

In heterogeneous networks, smart network nodes may extract and transmit a subset of an embedded bit stream with reduced bit rate to fit the channel capacity. In this case, the end user is only provided with a subset of the original bit stream and can reconstruct a lower quality image. Note that for a hierarchical approach, this scenario requires that the meaning and priority of the various bit-stream layers is known to the network and the network has some processing capability. This is not the case for the nonhierarchical approach.

Hierarchical approaches for video coding have been researched extensively over the past 15 years and are urgently required for many applications. Most of the strategies reported so far are usually not efficient enough for practical applications, in that they compromise decoder image quality too severely, or do not provide a sufficient number of embedded bit-stream layers [37]–[39]. Examples of such hierarchical approaches include data partitioning and fine granularity scalability (FGS) standardized with MPEG-2 and MPEG-4 [4], [40]. These strategies transmit DCT coefficients into separate bit streams with high granularity. The methods are low complex in nature. However, at the decoder the reconstruction from partial bit streams may result in significant degradation of quality over time (drift problem) [4]. Most recently, promising results for fine granularity embedded coding using 3-D wavelet decomposition and wavelet lifting have been reported [41].

2) Transcoding of Images and Video: The embedded coding strategies above provide network, device, and user adaptability by generating suitable bit streams directly. An alternative approach is to preprocess decoded video (or video in the compressed domain) to arrive at lower spatial or temporal resolution video. In contrast to embedded strategies, it is also possible to transcode to a standardized format in addition to meeting network or user constraints, i.e., transcoding 4-Mb/s MPEG-2 video into 1-Mb/s MPEG-4 format. Compressed domain transcoders process the original bit stream and generate a suitable transcoded bit stream, i.e., by requantizing data. The advantages of transcoding in the compressed domain are the low computational complexity. For video transcoding, requantization usually results in drift problems [42]. Transcoders can be deployed at the sender (to provide suitable bit streams upon request), in the network or at the receiver site.

B. Combating Transmission Errors

Many transmission channels cause severe challenges for streaming or broadcasting video due to bit errors or packet loss. Compressed video, which uses predictive coding algorithms, is sensitive to network impairments [33]. A single bit error can cause substantial degradation if there is no action taken at either coder or decoder, and can cause severe error propagation [10]. The loss of entire compressed video

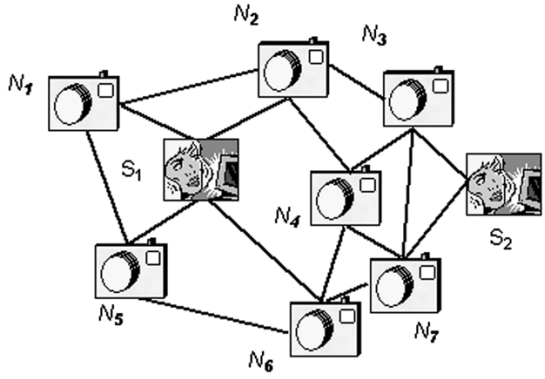

Fig. 17. Image sensor networks provide new challenges for efficient low complex, low power coding, and transmission. Each sensor N_1–N_7 also acts as a network node to transmit the information toward the two sinks S_1 and S_2.

data packets is even more serious. Depending on the length of a packet, packet loss may cause very long bursts of bit errors and thus severe degradation of the reconstructed video. Real-time video streaming applications are also sensitive to transport delay and delay variation. A packet of compressed video arriving too late to the decoder will be useless if the delay is too large and regarded as a lost packet.

1) Error Concealment: Even if encoders are not optimized for transmission errors, synchronization headers in the bit stream allow the detection of errors at the decoder and it is possible to employ powerful concealment strategies. These strategies include the replacement of lost blocks with pixel content within the same or the previous frame [33].

2) Error Robust Source Coding: The past 15 years have witnessed very intense research toward error robust encoder and decoder strategies [33]. These strategies reduce impaired regions or impairments in decoded images and some techniques have already been standardized with existing standards, such as JPEG, JPEG 2000, MPEG-1/2/4, and H.263/4.

3) Joint Source-Channel Coding: An important field that has developed over the past years is the attempt to perform source and channel coding for images or video jointly—in order to optimize the overall rate versus decoder distortion. Usually these strategies analyze the different parts of the bit stream in order to protect them according to their visual impact when lost in the network. This often leads to hierarchical representations of the image or video data with unequal error protection. Channel codes and channel decoder strategies are especially optimized for the video data [43].

C. Wireless Sensor Networks

Emerging or future applications and networks impose new challenges and constraints for the coding and transmission of images and video [44]. An example of such a network is a wireless video surveillance system comprising a dense field of video sensors—each video sensor also acts as a network node. Fig. 17 illustrates the concept whereby the information flow travels toward common sinks where the decoders reconstruct the information. Such wireless systems are usually not expensive to install, but low power consumption and

thus low computational demand at the sensors is of prime concern. Since every sensor also acts as a network node with limited throughput, video sensor data needs to be compressed efficiently—yet with very low encoder complexity. There is usually limited constraint at the decoder.

Slepian and Wolf showed that when adjacent sensors measure correlated data, these data can be coded with a total rate not exceeding the joint entropy, even without nodes explicitly communicating with each other [45]. This surprising theorem sparked much research interest recently, also toward very low complex coding strategies for standard video sequences—a true frontier in image and video compression [46].

VI. Summary and Discussion

In this paper, an overview was given about the state-of-the-art and recent trends in image and video coding algorithms. Research on digital image and video compression algorithms started in the 1960s. This field has since then matured rapidly—and has drastically shaped multimedia communications during the past 15 years and continues to do so. Examples include images and video on the internet, on digital photo and video cameras, on CD-ROM and DVD and the emergence of digital broadcast television.

Within this scenario, the development of international image and video coding standards, such as JPEG 2000, H.263/4, and MPEG-1/2/4, was vital for the acceptance of the technology in the market place. Most recent standards, such as JPEG, JPEG 2000, MPEG-4, and H.264, implement state-of-the-art compression technology. An analysis reveals that all these video compression standards are designed around the successful block-based hybrid MC/DCT algorithm. Wavelet transform based compression of still images has found its way into JPEG 2000 and the still image coding part of the MPEG-4 standard.

Research in the field continues to be of prime concern for many companies and research institutes around the world. Efforts focus primarily on improved compression efficiency, error robustness, and extended functionalities, such as scalable coding of images and video. In this paper, selected video coding research frontiers were discussed, primarily targeted for improved compression efficiency. Enhanced motion prediction is seen by many researchers in the field as the prime element to improve coding efficiency. To this end novel compression strategies employ sophisticated mid- or high-level computer vision techniques to extract and model content in video sequences. Examples include region-based and model-based video coding algorithms.

It remains to be seen how much these and other strategies can be developed into mature algorithms to compress video more efficiently than today's standards coders. Future video compression algorithms may employ multiple of the above mentioned motion prediction strategies in a prediction toolbox—and switch to various model prediction techniques whenever adequate. However, such scenarios will significantly increase implementation complexity both at the encoder and decoder. This disadvantage may be balanced with much improved processor capabilities in the future.

While improved compression efficiency continues to be important for many applications, new functionalities and requirements will be imposed by user devices and network constraints. That is, emerging wireless image and video sensor networks requirements may drastically change existing coding paradigms.

References

[1] T. Sikora, "MPEG digital video coding standards," *IEEE Signal Process. Mag.*, vol. 14, no. 5, pp. 82–100, Sep. 1997.

[2] D. Legall, "The MPEG video compression algorithm," *Image Commun.*, vol. 4, pp. 129–140, Apr. 1992.

[3] B. G. Haskell, A. Puri, and A. N. Netravali, *Digital Video: An Introduction to MPEG-2*. London, U.K.: Chapman & Hall, 1997, Digital Multimedia Standards Series.

[4] F. Pereira and T. Ebrahimi, *The MPEG-4 Book*. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[5] G. K. Wallace, "JPEG still picture compression standard," in *Commun. ACM*, vol. 34, Apr. 1991, pp. 30–44.

[6] I. Richardson, *Video Codec Design: Developing Image and Video Compression Systems*. New York: Wiley, 2002.

[7] D. Lee, "JPEG 2000: Retrospective and New Developments," *Proc. IEEE*, vol. 93, no. 1, pp. 32–41, Jan. 2005.

[8] *Special Issue on the H.264/AVC Video Coding Standard, IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, Jul. 2003.

[9] N. Ahmed, T. Natrajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1, pp. 90–93, Dec. 1984.

[10] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[11] A. Netravali and B. Haskel, *Digital Pictures: Representations, Compression, and Standards (Applications of Communications Theory)*. New York: Plenum, 1995.

[12] M. Vetterli and J. Kovacevič, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[13] C.-F. Chen and K. K. Pang, "The optimal transform of motion-compensated frame difference images in a hybrid coder," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, no. 6, pp. 393–397, Jun. 1993.

[14] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3445–3462, Dec. 1993.

[15] 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 Syst. Video Technol.*, vol. 7, no. 1, pp. 181–196, Feb. 1997.

[16] A. Smolic, T. Sikora, and J.-R. Ohm, "Long-term global motion estimation and its application for sprite coding, content description, and segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1227–1242, Dec. 1999.

[17] M. Karczewicz, J. Niewglowski, and P. Haavisto, "Video coding using motion compensation with polynomial motion vector fields," *Signal Process. Image Commun.*, vol. 10, no. 1–3, pp. 63–91, 1997.

[18] A. Smolic, B. Makai, and T. Sikora, "Real-time estimation of long-term 3-D motion parameters for SNHC face animation and model-based coding applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 255–263, Mar. 1999.

[19] J. Ostermann, E. S. Jang, J. S. Shin, and T. Chen, "Coding the arbitrarily shaped video objects in MPEG-4," in *IEEE Int. Conf. Image Processing*, Santa Barbara, CA, 1997, pp. 496–499.

[20] T. Sikora, "Low complexity shape-adaptive DCT for coding of arbitrarily shaped image segments," *Signal Process. Image Commun.*, vol. 7, no. 4–6, pp. 381–395, Nov. 1995.

[21] T. Sikora, S. Bauer, and B. Makai, "Efficiency of shape-adaptive 2-D transforms for coding of arbitrarily shaped image segments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 3, pp. 254–258, June 1995.

[22] P. Kauff and K. Schuur, "A shape-adaptive DCT with block based DC separation and delta-DC correction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 3, pp. 237–242, June 1998.

[23] D. Santa-Cruz, R. Grosbois, and T. Ebrahimi, "JPEG 2000 performance evaluation and assessment," *Signal Process. Image Commun.*, vol. 17, no. 1, pp. 113–130, 2002.
[24] *Special Issue on the H.264/AVC Video Coding Standard, IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 557–728, July 2003.
[25] L. Torres and M. Kunt, *Video Coding: The Second Generation Approach*. Englewood Cliffs, NJ: Kluwer, 1996.
[26] P. Salembier, L. Torres, F. Meyer, and C. Gu, "Region-based video coding using mathematical morphology," *Proc. IEEE*, vol. 83, no. 6, pp. 843–857, Jun. 1995.
[27] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," in *IEEE Trans. Image Process.*, vol. 3, May 1994, pp. 327–331.
[28] 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 Syst. Video Technol.*, vol. 7, no. 1, pp. 130–145, Feb. 1997.
[29] H. Musmann, M. Hötter, and J. Ostermann, "Object oriented analysis-synthesis coding of moving images," *Image Commun.*, vol. 1, no. 2, pp. 117–132, Oct. 1989.
[30] P. Eisert, T. Wiegand, and B. Girod, "Model-aided coding: A new approach to incorporate facial animation into motion-compensated video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 3, pp. 344–358, Apr. 2000.
[31] H. Li and R. Forchheimer, "Two-view facial movement estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no. 3, pp. 276–287, Jun. 1994.
[32] A. Smolic and T. Sikora, "Coding of image sequences using a layered 2D/3D model-based coding approach," in *Proc. PCS'97, Picture Coding Symp.*, Berlin, Germany, Sep. 10–12, 1997.
[33] A. Reibman and M. T. Sun, Eds., *Wireless Video*. New York: Marcel Dekker, 2000.
[34] V. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Process. Mag.*, vol. 18, pp. 74–93, Sep. 2001.
[35] Y. Wang, A. R. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proc. IEEE*, vol. 93, no. 1, pp. 57–70, Jan. 1995.
[36] S. Ekmekci and T. Sikora, "Unbalanced quantized multiple description video transmission using path diversity," in *Proc. IS&T/SPIE's Electronic Imaging 2003*, Santa Clara, CA, Jan. 2003.
[37] C. Gonzales and E. Viscito, "Flexibly scalable digital video coding," *Signal Process. Image Commun.*, vol. 5, no. 1–2, pp. 5–20, Feb. 1993.
[38] T. Sikora, T. K. Tan, and K. N. Ngan, "A performance comparison of frequency domain pyramid scalable coding schemes," in *Proc. Picture Coding Symp.*, Lausanne, Switzerland, Mar. 1993, pp. 16.1–16.2.
[39] A. Puri and A. Wong, "Spatial domain resolution scalable video coding," in *Proc. SPIE Visual Communications and Image Processing*, Boston, MA, Nov. 1993.
[40] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 301–317, Mar. 2001.
[41] J. Ohm, "Advances in scalable video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 42–56, Jan. 1995.
[42] P. Yin, A. Vetro, B. Liu, and H. Sun, "Drift compensation for reduced spatial resolution transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 1009–1020, Nov. 2002.
[43] M. W. Garrett and M. Vetterli, "Joint source/channel coding of statistically multiplexed real time services on packet networks," in *IEEE/ACM Trans. Netw.*, vol. 1, Feb. 1993, pp. 71–80.
[44] H. Karl, A. Willig, and A. Wolisz, Eds., *Proceedings of the 1st European Workshop on Wireless Sensor Networks*. Berlin, Germany: Springer-Verlag, 2004.
[45] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. 19, no. 4, pp. 471–480, Jul. 1973.
[46] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 1995.

Thomas Sikora (Senior Member, IEEE) received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from Bremen University, Bremen, Germany, in 1985 and 1989, respectively.

In 1990, he joined Siemens Ltd. and Monash University, Melbourne, Australia, as a Project Leader responsible for video compression research activities in the Australian Universal Broadband Video Codec consortium. Between 1994 and 2001, he was the Director of the Interactive Media Department, Heinrich Hertz Institute (HHI) Berlin GmbH, Germany. He has been involved in international ITU and ISO standardization activities as well as in several European research activities for a number of years. As the Chairman of the ISO-MPEG (Moving Picture Experts Group) video group, he was responsible for the development and standardization of the MPEG-4 and MPEG-7 video algorithms. He is appointed as Research Chair for the VISNET and 3DTV European Networks of Excellence. He is an Appointed Member of the Advisory and Supervisory board of a number of German companies and international research organizations. He frequently works as an industry consultant on issues related to interactive digital audio and video. He is currently Professor and Director of the Communication Systems Department, Technical University Berlin, Germany. He has published more than 150 papers related to audio and video processing. He is an Advisory Editor for the EURASIP *Signal Processing: Image Communication* journal and an Associate Editor of the EURASIP *Signal Processing* journal.

Dr. Sikora is a Member of the German Society for Information Technology (ITG) and a recipient of the 1996 ITG Award. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. From 1998 to 2002, he was an Associate Editor of the IEEE SIGNAL PROCESSING MAGAZINE.