Chapter 1

Introduction

1.1 A Brief Introduction to Compression Theory

The ultimate aim of data compression is the removal of redundancy from the source signal. This, therefore, reduces the number of binary bits required to represent the information contained within the source. Achieving the best possible compression ratio requires not only an understanding of the nature of the source signal in its binary representation, but also how we as humans interpret the information that the data represents.

We live in a world of rapidly improving computing and communications capabilities, and owing to an unprecedented increase in computer awareness, the demand for computer systems and their applications has also drastically increased. As the transmission or storage of every single bit incurs a cost, the advancement of cost-efficient source-signal compression techniques is of high significance. When considering the transmission of a source signal that may contain a substantial amount of redundancy, achieving a high compression ratio is of paramount importance.

In a simple system, the same number of bits might be used for representing the symbols "*e*" and "*q*". Statistically speaking, however, it can be shown that the character "*e*" appears in English text more frequently than the character "*q*". Hence, on representing the more-frequent symbols with fewer bits than the less-frequent symbols we stand to reduce the total number of bits necessary for encoding the entire information transmitted or stored.

Indeed a number of source-signal encoding standards have been formulated based on the removal of predictability or redundancy from the source. The most widely used principle dates back to the 1940s and is referred to as Shannon–Fano coding [2, 3], while the well-known Huffman encoding scheme was contrived in 1952 [4]. These approaches, however, have been further enhanced many times since then and have been invoked in various applications. Further research will undoubtedly endeavor to continue improving upon those techniques, asymptotically approaching the information theoretic limits.

Digital video compression techniques [5–9] have played an important role in the world of wireless telecommunication and multimedia systems, where bandwidth is a valuable commodity. Hence, the employment of video compression techniques is of prime importance

Video Compression and Communications Second Edition
L. Hanzo, P. J. Cherriman and J. Streit © 2007 John Wiley & Sons, Ltd

Table 1.1: Image Formats, Their Dimensions, and Typical Applications

Resolution	Dimensions	Pixel/s at 30 frames/s	Applications
Sub-QCIF	128 × 96	0.37 M	Handheld mobile video and videoconferencing via public phone networks
QCIF	176 × 144	0.76 M	
CIF	352 × 288	3.04 M	Videotape recorder quality
CCIR 601	720 × 480	10.40 M	TV
4CIF	704 × 576	12.17 M	
HDTV 1440	1440 × 960	47.00 M	Consumer HDTV
16CIF	1408 × 1152	48.66 M	
HDTV	1920 × 1080	62.70 M	Studio HDTV

in order to reduce the amount of information that has to be transmitted to adequately represent a picture sequence without impairing its subjective quality, as judged by human viewers. Modern compression techniques involve complex algorithms which have to be standardized in order to obtain global compatibility and interworking.

1.2 Introduction to Video Formats

Many of the results in this book are based on experiments using various resolution representations of the “Miss America” sequence, as well as the “Football” and “Susie” sequences. The so-called “Mall” sequence is used at High Definition Television (HDTV) resolution. Their spatial resolutions are listed in Table 1.1 along with a range of other video formats.

Each sequence has been chosen to test the codecs’ performance in particular scenarios. The “Miss America” sequence is of low motion and provides an estimate of the maximum achievable compression ratio of a codec. The “Football” sequence contains pictures of high motion activity and high contrast. All sequences were recorded using interlacing equipment. *Interlacing* is a technique that is often used in image processing in order to reduce the required bandwidth of video signals, such as, for example, in conventional analog television signals, while maintaining a high frame-refresh rate, hence avoiding flickering and video jerkiness. This is achieved by scanning the video scene at half the required viewing-rate — which potentially halves the required video bandwidth and the associated bitrate — and then displaying the video sequence at twice the input scanning rate, such that in even-indexed video frames only the even-indexed lines are updated before they are presented to the viewer. In contrast, in odd-indexed video frames only the odd-indexed lines are updated before they are displayed, relying on the human eye and brain to reconstruct the video scene from these halved scanning rate even and odd video fields. Therefore, every other line of the interlaced frames remains un-updated.

For example, for frame 1 of the interlaced “Football” sequence in Figure 1.1 we observe that a considerable amount of motion took place between the two recoding instants of each

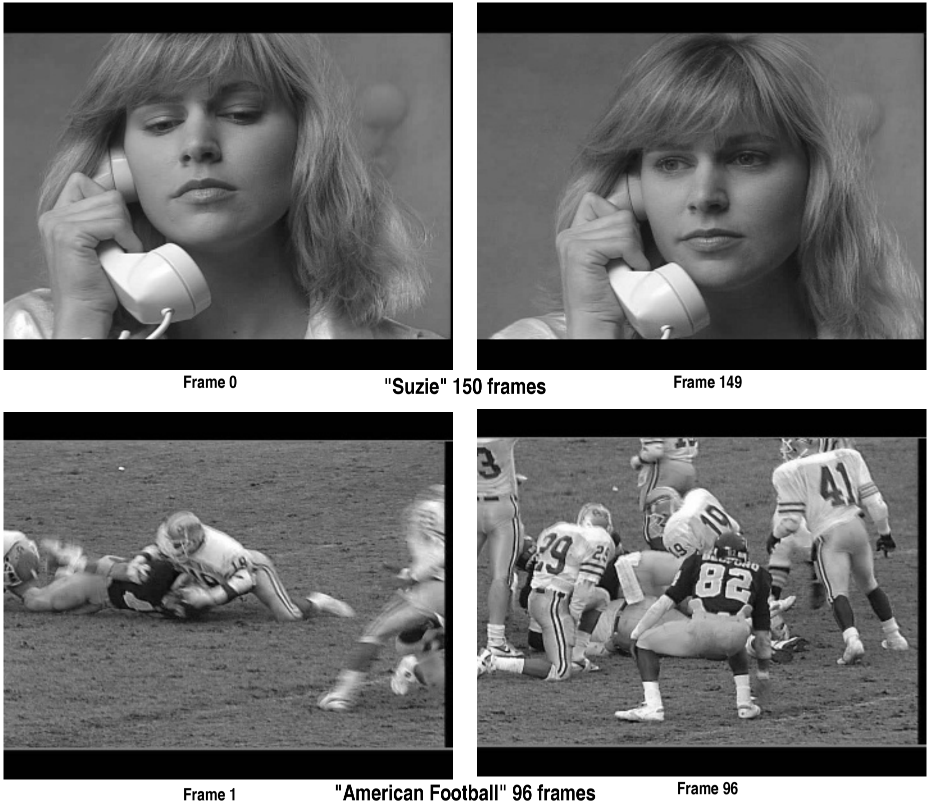

Figure 1.1: 4CIF video sequences.

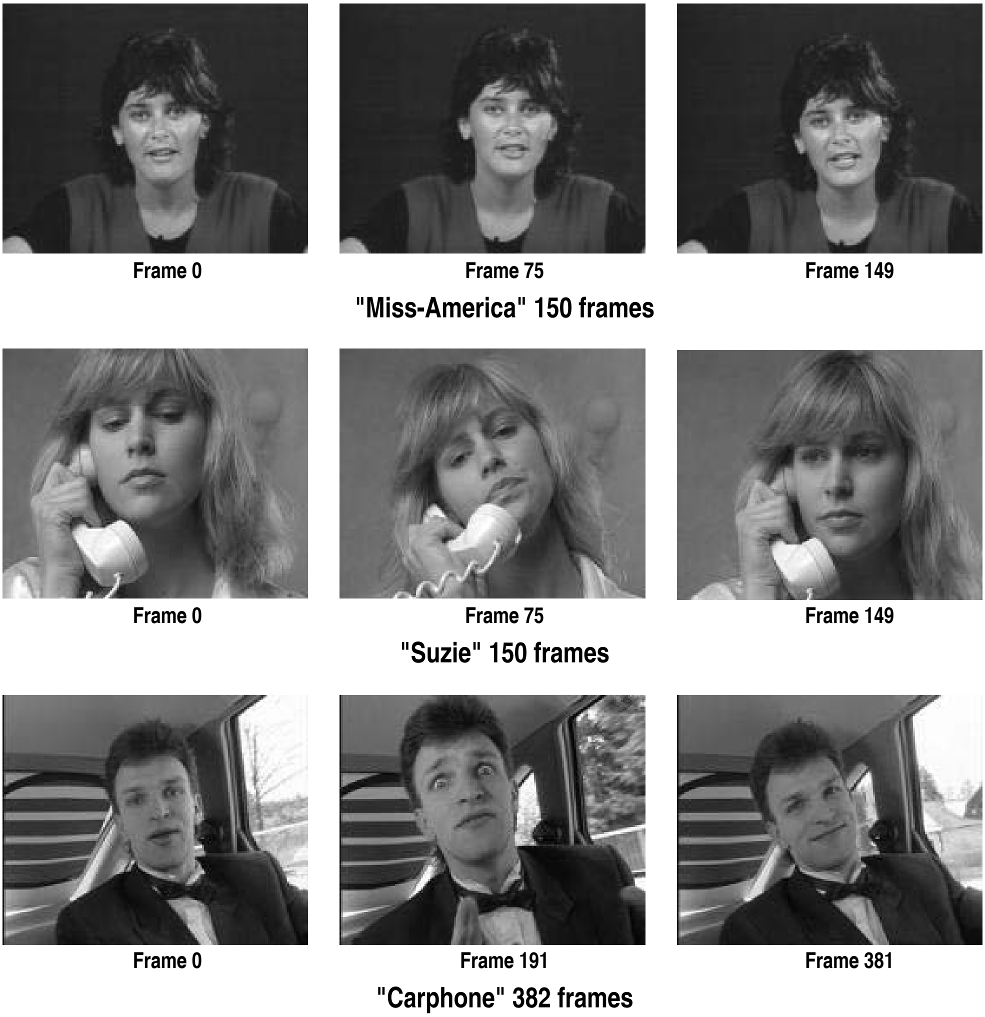

Figure 1.2: QCIF video sequences.

frame, which correspond to the even and odd video fields. Furthermore, the “Susie” sequence was used in our experiments in order to verify the color reconstruction performance of the proposed codecs, while the “Mall” sequence was employed in order to simulate HDTV sequences with camera panning. As an example, a range of frames for each QCIF video sequence used is shown in Figure 1.2. QCIF resolution images are composed of 176×144 pixels and are suitable for wireless handheld videotelephony. The 4CIF resolution images are suitable for digital television, which are 16 times larger than QCIF images. A range of frames for the 4CIF video sequences is shown in Figure 1.1. Finally, in Figure 1.3 we show a range of frames from the 1280×640-pixel “Mall” sequence. However, because the 16CIF resolution is constituted by 1408×1152 pixels, a black border was added to the sequences before they were coded.

We processed all sequences in the YUV color space [10] where the incoming picture information consists of the luminance (Y) plus two color difference signals referred to as chrominance U (Cr_u) and chrominance V (Cr_v). The conversion of the standard Red–Blue–Green (RGB) representation to the YUV format is defined in Equation 1.1:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.146 & -0.288 & 0.434 \\ 0.617 & -0.517 & -0.100 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}. \tag{1.1}$$

It is common practice to reduce the resolution of the two color difference signals by a factor of two in each spatial direction, which inflicts virtually no perceptual impairment and reduces the associated source data rate by 50%. More explicitly, this implies that instead of having to store and process the luminance signal and the two color difference signals at the same resolution, which would potentially increase the associated bitrate for color sequences by a factor of three, the total amount of color data to be processed is only 50% more than that of the associated gray-scale images. This implies that there is only one Cr_u and one Cr_v pixel for every four luminance pixels allocated.

The coding of images larger than the QCIF size multiplies the demand in terms of computational complexity, bitrate, and required buffer size. This might cause problems, considering that a color HDTV frame requires a storage of 6 MB per frame. At a frame rate of 30 frames/s, the uncompressed data rate exceeds 1.4 Gbit/s. Hence, for real-time applications the extremely high bandwidth requirement is associated with an excessive computational complexity. Constrained by this complexity limitation, we now examine two inherently low-complexity techniques and evaluate their performance.

1.3 Evolution of Video Compression Standards

Digital video signals may be compressed by numerous different proprietary or standardized algorithms. The most important families of compression algorithms are published by recognized standardization bodies, such as the International Organization for Standardization (ISO), the International Telecommunication Union (ITU), or the Motion Picture Expert Group (MPEG). In contrast, proprietary compression algorithms developed and owned by a smaller interest group are of lesser significance owing to their lack of global compatibility and interworking capability. The evolution of video compression standards over the past half-a-century is shown in Figure 1.4.

As seen in the figure, the history of video compression commences in the 1950s. An analog videophone system had been designed, constructed, and trialled in the 1960s, but it required a high bandwidth and it was deemed that using the postcard-size black-and-white pictures produced did not substantially augment the impression of telepresence in comparison to conventional voice communication. In the 1970s, it was realized that visual identification of the communicating parties may be expected to substantially improve the value of multi-party discussions and hence the introduction of videoconference services was considered. The users' interest increased in parallel to improvements in picture quality.

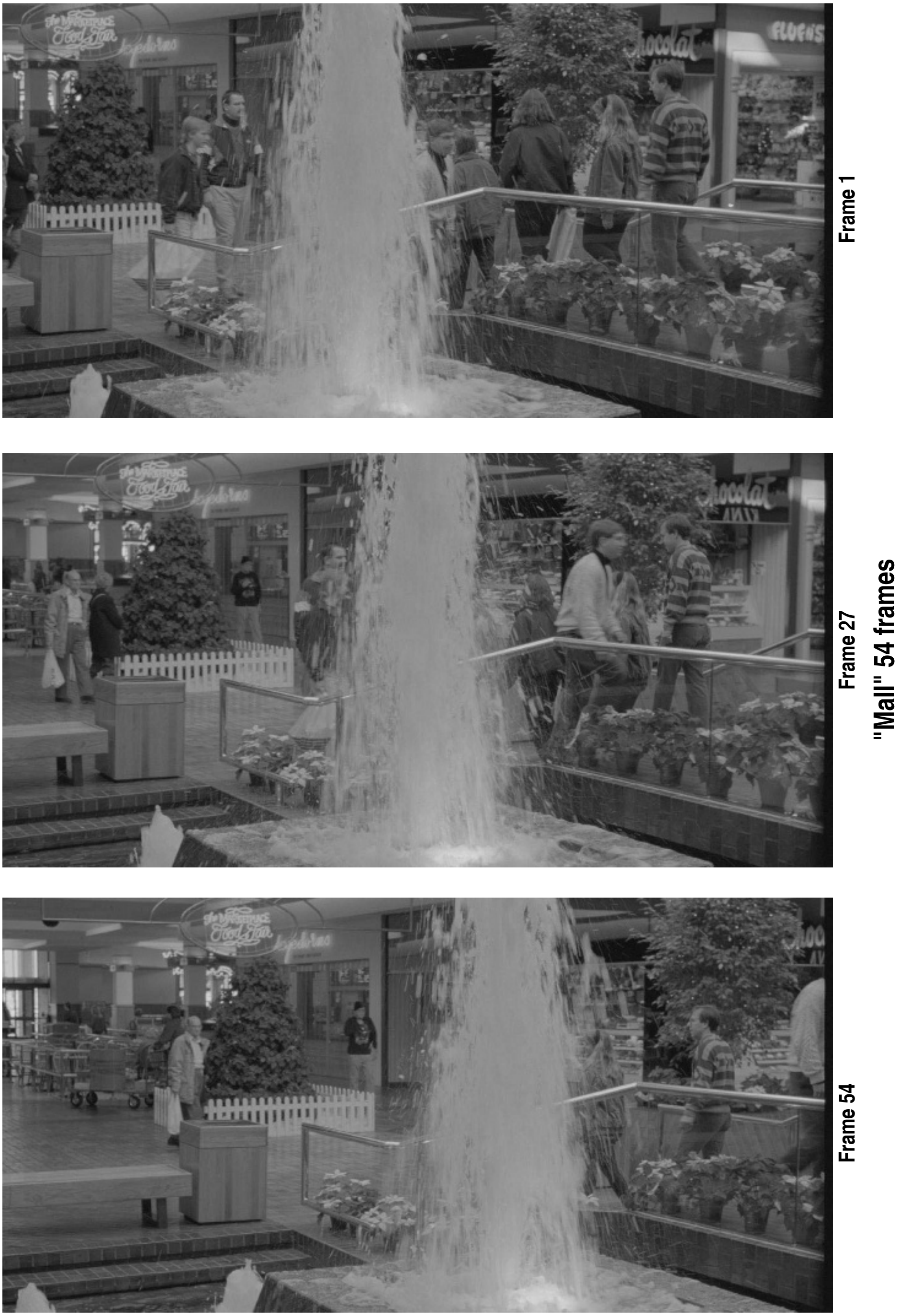

Figure 1.3: 16CIF “Mall” video sequence.

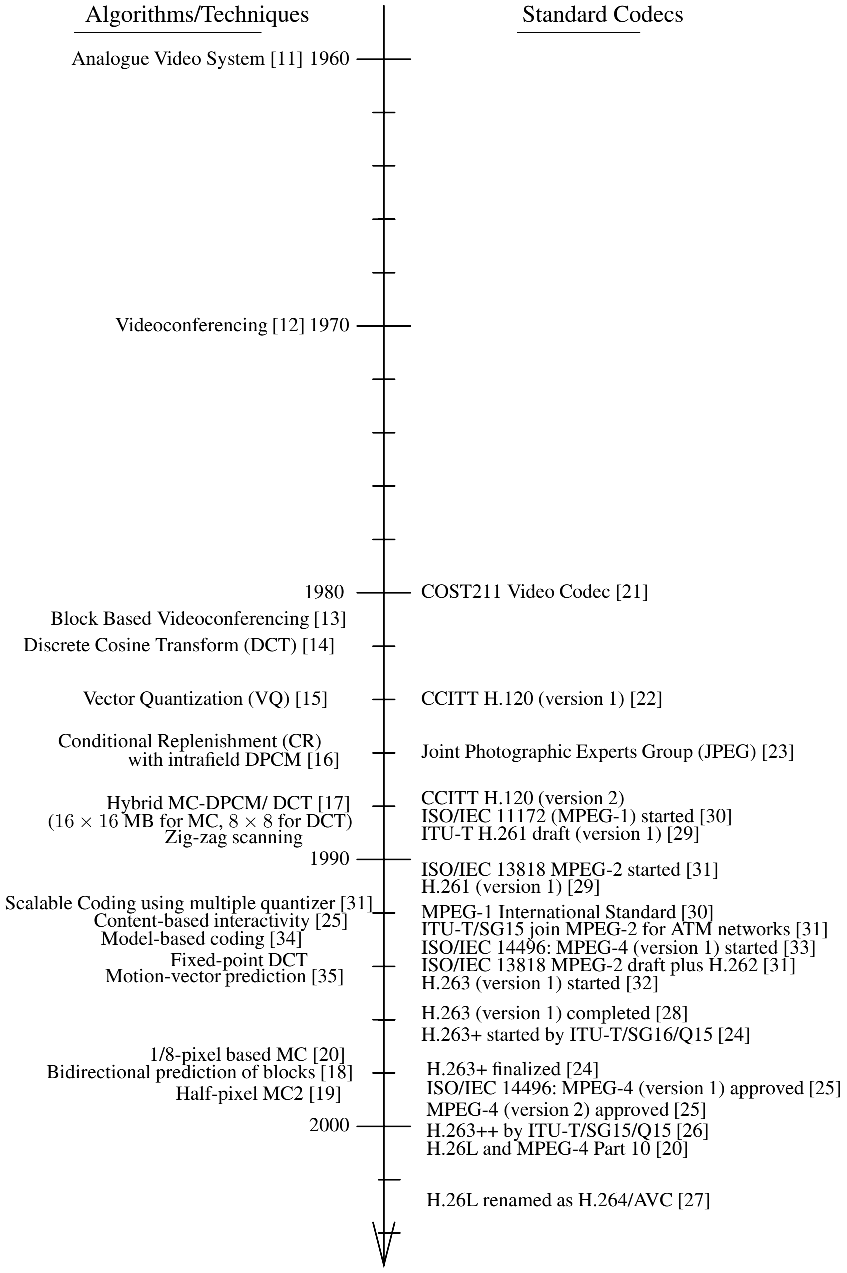
Algorithms/Techniques
Standard Codecs
Analogue Video System [11] 1960
Videoconferencing [12] 1970
1980
COST211 Video Codec [21]
Block Based Videoconferencing [13]
Discrete Cosine Transform (DCT) [14]
Vector Quantization (VQ) [15]
CCITT H.120 (version 1) [22]
Conditional Replenishment (CR) with intrafield DPCM [16]
Joint Photographic Experts Group (JPEG) [23]
Hybrid MC-DPCM/ DCT [17]
(16 × 16 MB for MC, 8 × 8 for DCT)
Zig-zag scanning
CCITT H.120 (version 2)
ISO/IEC 11172 (MPEG-1) started [30]
ITU-T H.261 draft (version 1) [29]
1990
ISO/IEC 13818 MPEG-2 started [31]
H.261 (version 1) [29]
Scalable Coding using multiple quantizer [31]
Content-based interactivity [25]
Model-based coding [34]
Fixed-point DCT
Motion-vector prediction [35]
MPEG-1 International Standard [30]
ITU-T/SG15 join MPEG-2 for ATM networks [31]
ISO/IEC 14496: MPEG-4 (version 1) started [33]
ISO/IEC 13818 MPEG-2 draft plus H.262 [31]
H.263 (version 1) started [32]
H.263 (version 1) completed [28]
H.263+ started by ITU-T/SG16/Q15 [24]
1/8-pixel based MC [20]
Bidirectional prediction of blocks [18]
Half-pixel MC2 [19]
H.263+ finalized [24]
ISO/IEC 14496: MPEG-4 (version 1) approved [25]
MPEG-4 (version 2) approved [25]
2000
H.263++ by ITU-T/SG15/Q15 [26]
H.26L and MPEG-4 Part 10 [20]
H.26L renamed as H.264/AVC [27]

Figure 1.4: A brief history of video compression.

Video coding standardization activities started in the early 1980s. These activities were initiated by the International Telegraph and Telephone Consultative Committee (CCITT) [34], which is currently known as the International Telecommunications Union — Telecommunication Standardisation Sector (ITU-T) [22]. These standardization bodies were later followed by the formation of the Consultative Committee for International Radio (CCIR); currently ITU-R) [36], the ISO, and the International Electrotechnical Commission (IEC). These bodies coordinated the formation of various standards, some of which are listed in Table 1.2 and are discussed further in the following sections.

1.3.1 The International Telecommunications Union's H.120 Standard

Using state-of-the-art technology in the 1980s, a video en**co**der/**dec**oder (codec) was designed by the Pan-European Cooperation in Science and Technology (COST) project 211, which was based on Differential Pulse Code Modulation (DPCM) [59, 60] and was ratified by the CCITT as the H.120 standard [61]. This codec's target bitrate was 2 Mbit/s for the sake of compatibility with the European Pulse Code Modulated (PCM) bitrate hierarchy in Europe and 1.544 Mbit/s for North America [61], which was suitable for convenient mapping to their respective first levels of digital transmission hierarchy. Although the H.120 standard had a good spatial resolution, because DPCM operates on a pixel-by-pixel basis, it had a poor temporal resolution. It was soon realized that in order to improve the image quality without exceeding the above-mentioned 2 Mbit/s target bitrate, less than one bit should be used for encoding each pixel. This was only possible if a group of pixels, for example a "block" of 8 × 8 pixels, were encoded together such that the number of bits per pixel used may become a non-integer. This led to the design of so-called block-based codecs. More explicitly, at 2 Mbit/s and at a frame rate of 30 frames/s the maximum number of bits per frame was approximately 66.67 kbits. Using black and white pictures at 176 × 144-pixel resolution, the maximum number of bits per pixel was 2 bits.

1.3.2 Joint Photographic Experts Group

During the late 1980s, 15 different block-based videoconferencing proposals were submitted to the ITU-T standard body (formerly the CCITT), and 14 of these were based on using the Discrete Cosine Transform (DCT) [14] for still-image compression, while the other used Vector Quantization (VQ) [15]. The subjective quality of video sequences presented to the panel of judges showed hardly any perceivable difference between the two types of coding techniques. In parallel to the ITU-T's investigations conducted during the period of 1984–1988 [23], the Joint Photographic Experts Group (JPEG) was also coordinating the compression of static images. Again, they opted for the DCT as the favored compression technique, mainly due to their interest in progressive image transmission. JPEG's decision undoubtedly influenced the ITU-T in favoring the employment of DCT over VQ. By this time there was worldwide activity in implementing the DCT in chips and on Digital Signal Processors (DSPs).

Table 1.2: Evolution of Video Communications

Date	Standard
1956	AT&T designs and construct the first Picturephone test system [37]
1964	AT&T introduces Picturephone at the World's Fair, New York [37]
1970	AT&T offers Picturephone for $160 per month [37]
1971	Ericsson demonstrates the first Trans-Atlantic videophone (LME) call
1973 Dec.	ARPAnet packet voice experiments
1976 March	Network Voice Protocol (NVP), by Danny Cohen, USC/ISI [38]
1981 July	Packet Video Protocol (PVP), by Randy Cole, USC/ISI [39]
1982	CCITT (predecessor of the ITU-T) standard H.120 (2 Mbit/s) video coding, by European COST 211 project [22]
1982	Compression Labs begin selling $250,000 video conference (VC) system, $1,000 per hour lines
1986	PictureTel's $80,000 VC system, $100 per hour lines
1987	Mitsubishi sells $1,500 still-picture phone
1989	Mitsubishi drops still-picture phone
1990	TWBnet packet audio/video experiments, portable video players (pvp) (video) from Information Science Institute (ISI)/Bolt, Beranek and Newman, Inc. (BBN) [40]
1990	CCITT standard H.261 ($p \times 64$) video coding [29]
1990 Dec.	CCITT standard H.320 for ISDN conferencing [41]
1991	PictureTel unveils $20,000 black-and-white VC system, $30 per hour lines
1991	IBM and PictureTel demonstrate videophone on PC
1991 Feb.	DARTnet voice experiments, Voice Terminal (VT) program from USC/ISI [42]
1991 June	DARTnet research's packet video test between ISI and BBN. [42]
1991 Aug.	University of California, Berkeley (UCB)/Lawrence Berkeley National Laboratories (LBNL)'s audio tool vat releases for DARTnet use [42]
1991 Sept.	First audio/video conference (H.261 hardware codec) at DARTnet [42]
1991 Dec	dvc (receive-only) program, by Paul Milazzo from BBN, Internet Engineering Task Force (IETF) meeting, Santa Fe [43]
1992	AT&T's $1,500 videophone for home market [37]
1992 March	World's first Multicaster BackBONE (MBone) audio cast (vat), 23rd IETF, San Diego
1992 July	MBone audio/video casts (vat/dvc), 24th IETF, Boston
1992 July	Institute National de Recherche en Informatique et Automatique (INRIA) Videoconferencing System (ivs), by Thierry Turletti from INRIA [44]
1992 Sept.	CU-SeeMe v0.19 for Macintosh (without audio), by Tim Dorcey from Cornell University [45]
1992 Nov.	Network Video (nv) v1.0, by Ron Frederick from Xerox's Palo Alto Research Center (Xerox PARC), 25th IETF, Washington DC
1992 Dec.	Real-time Transport Protocol (RTP) v1, by Henning Schulzrinne [46]
1993 April	CU-SeeMe v0.40 for Macintosh (with multipoint conferencing) [47]
1993 May	Network Video (NV) v3.2 (with color video)
1993 Oct.	VIC Initial Alpha, by Steven McCanne and Van Jacobson from UCB/LBNL
1993 Nov.	VocalChat v1.0, an audio conferencing software for Novell IPX networks

Table 1.2: Continued

Date	Standard
1994 Feb.	CU-SeeMe v0.70b1 for Macintosh (with audio), audio code by Charley Kline's Maven [47]
1994 April	CU-SeeMe v0.33b1 for Windows (without audio), by Steve Edgar from Cornell [47]
1995 Feb.	VocalTec Internet Phone v1.0 for Windows (without video) [48]
1995 Aug.	CU-SeeMe v0.66b1 for Windows (with audio) [47]
1996 Jan.	RTP v2, by IETF avt-wg
1996 March	ITU-T standard H.263 ($p \times 8$) video coding for low bitrate communication [28]
1996 March	VocalTec Telephony Gateway [49]
1996 May	ITU-T standard H.324 for Plain Old Telephone System (POTS) conferencing [50]
1996 July	ITU-T standard T.120 for data conferencing [51]
1996 Aug.	Microsoft NetMeeting v1.0 (without video)
1996 Oct.	ITU-T standard H.323 v1, by ITU-T SG 16 [52]
1996 Nov.	VocalTec Surf&Call, the first Web to phone plugin
1996 Dec.	Microsoft NetMeeting v2.0b2 (with video)
1996 Dec.	VocalTec Internet Phone v4.0 for Windows (with video) [48]
1997 July	Virtual Room Videoconferencing System (VRVS), Caltech-CERN project [53]
1997 Sept.	Resource ReSerVation Protocol (RSVP) v1 [54]
1998 Jan.	ITU-T standard H.323 v2 [55]
1998 Jan.	ITU-T standard H.263 v2 (H.263+) video coding [24]
1998 April	CU-SeeMe v1.0 for Windows and Macintosh (using color video), from Cornell University, USA [47]
1998 May	Cornell's CU-SeeMe development team has completed their work [47]
1998 Oct.	ISO/IEC standard MPEG-4 v1, by ISO/IEC JTC1/SC29/WG11 (MPEG) [25]
1999 Feb.	Session Initiation Protocol (SIP) makes proposed standard, by IETF music-work group [56]
1999 April	Microsoft NetMeeting v3.0b
1999 Aug.	ITU-T H.26L Test Model Long-term (TML) project, by ITU-T SG16/Q.6 (VCEG) [20]
1999 Sept.	ITU-T standard H.323 v3 [57]
1999 Oct.	Network Address Translation (NAT) compatible version of iVisit, v2.3b5 for Windows and Macintosh
1999 Oct.	Media Gateway Control Protocol (MGCP) v1, IETF
1999 Dec.	Microsoft NetMeeting v3.01 service pack 1 (4.4.3388)
1999 Dec.	ISO/IEC standard MPEG-4 v2
2000 May	Columbia SIP user agent sipc v1.30
2000 Oct.	Samsung releases the first MPEG-4 streaming 3G (CDMA2000-1x) video cell phone
2000 Nov.	ITU-T standard H.323 v4 [58]
2000 Nov.	MEGACO/H.248 Protocol v1, by IETF megaco-wg and ITU-T SG 16
2000 Dec.	Microsoft NetMeeting v3.01 service pack 2 (4.4.3396))
2000 Dec.	ISO/IEC Motion JPEG 2000 (JPEG 2000, Part 3) project, by ISO/IEC JTC1/SC29/WG1 (JPEG) [23]

Table 1.2: Continued

Date	Standard
2001 June	Windows XP Messenger supports the SIP
2001 Sept.	World's first Trans-Atlantic gallbladder surgery using a videophone (by surgeon Lindbergh)
2001 Oct.	NTT DoCoMo sells $570 3G (WCDMA) mobile videophone
2001 Oct.	TV reporters use $7,950 portable satellite videophone to broadcast live from Afghanistan
2001 Oct.	Microsoft NetMeeting v3.01 (4.4.3400) on XP
2001 Dec.	Joint Video Team (JVT) video coding (H.26L and MPEG-4 Part 10) project, by ITU-T SG16/Q.6 (VCEG) and ISO/IEC JTC1/SC29/WG 11 (MPEG) [20]
2002 June	World's first 3G video cell phone roaming
2002 Dec.	JVT completes the technical work leading to ITU-T H.264 [27]
2003	Wireless videotelephony commercialized

1.3.3 The ITU H.261 Standard

During the late 1980s it became clear that the recommended ITU-T videoconferencing codec would use a combination of motion-compensated inter-frame coding and the DCT. The codec exhibited a substantially improved video quality in comparison with the DPCM-based H.120 standard. In fact, the image quality was found to be sufficiently high for videoconferencing applications at 384 kbits/s and good quality was attained using 352 × 288-pixel Common Intermediate Format (CIF) or 176 × 144-pixel Quarter CIF (QCIF) images at bitrates of around 1 Mbit/s. The H.261 codec [29] was capable of using 31 different quantizers and various other adjustable coding options, hence its bitrate spanned a wide range. Naturally, the bitrate depended on the motion activity and the video format, hence it was not perfectly controllable. Nonetheless, the H.261 scheme was termed as a $p \times 64$ bits/s codec, $p = 1, \ldots, 30$ to comply with the bitrates provided by the ITU's PCM hierarchy. The standard was ratified in late 1989.

1.3.4 The Motion Pictures Expert Group

In the early 1990s, the Motion Picture Experts Group (MPEG) was created as Sub-Committee 2 of ISO (ISO/SC2). The MPEG started investigating the conception of coding techniques specifically designed for the storage of video, in media such as CD-ROMs. The aim was to develop a video codec capable of compressing highly motion-active video scenes such as those seen in movies for storage on hard disks, while maintaining a performance comparable to that of Video Home System (VHS) video-recorder quality. In fact, the basic MPEG-1 standard [30], which was reminiscent of the H.261 ITU codec [29], was capable of accomplishing this task at a bitrate of 1.5 Mbit/s. When transmitting broadcast-type distributive, rather than interactive of video, the encoding and decoding delays do not constitute a major constraint, one can trade delay for compression efficiency. Hence, in contrast to the H.261 interactive codec, which had a single-frame video delay, the MPEG-

1 codec introduced the bidirectionally predicted frames in its motion-compensation scheme.

At the time of writing, MPEG decoders/players are becoming commonplace for the storage of multimedia information on computers. MPEG-1 decoder plug-in hardware boards (e.g. MPEG magic cards) have been around for a while, and software-based MPEG-1 decoders are available with the release of operating systems or multimedia extensions for Personal Computer (PC) and Macintosh platforms.

MPEG-1 was originally optimized for typical applications using non-interlaced video sequences scanned at 25 frames/s in European format and at 29.9 frames/s in North American format. The bitrate of 1.2 to 1.5 Mbits/s typically results in an image quality comparable to home Video Cassette Recorders (VCRs) [30] using CIF images, which can be further improved at higher bitrates. Early versions of the MPEG-1 codec used for encoding interlaced video, such as those employed in broadcast applications, were referred to as MPEG-1+.

1.3.5 The MPEG-2 Standard

A new generation of MPEG coding schemes referred to as MPEG-2 [8, 31] was also adopted by broadcasters who were initially reluctant to use any compression of video sequences. The MPEG-2 scheme encodes CIF-resolution codes for interlaced video at bitrates of 4–9 Mbits/s, and is now well on its way to making a significant impact in a range of applications, such as digital terrestrial broadcasting, digital satellite TV [5], digital cable TV, digital versatile disc (DVD) and many others. Television broadcasters started using MPEG-2 encoded digital video sequences during the late 1990s [31].

A slightly improved version of MPEG-2, termed as MPEG-3, was to be used for the encoding of HDTV, but since MPEG-2 itself was capable of achieving this, the MPEG-3 standards were folded into MPEG-2. It is foreseen that by the year 2014, the existing transmission of NTSC format TV programmes will cease in North America and instead HDTV employing MPEG-2 compression will be used in terrestrial broadcasting.

1.3.6 The ITU H.263 Standard

The H.263 video codec was designed by the ITU-T standardization body for low-bitrate encoding of video sequences in videoconferencing [28]. It was first designed to be utilized in H.323-based systems [55], but it has also been adopted for Internet-based videoconferencing.

The encoding algorithms of the H.263 codec are similar to those used by its predecessor, namely the H.261 codec, although both its coding efficiency and error resilience have been improved at the cost of a higher implementational complexity [5]. Some of the main differences between the H.261 and H.263 coding algorithms are listed below. In the H.263 codec, half-pixel resolution is used for motion compensation, whereas H.261 used full-pixel precision in conjunction with a smoothing filter invoked for removing the high-frequency spatial changes in the video frame, which improved the achievable motion-compensation efficiency. Some parts of the hierarchical structure of the data stream are now optional in the H.263 scheme, hence the codec may be configured for attaining a lower data rate or better error resilience. There are four negotiable options included in the standard for the sake of potentially improving the attainable performance provided that both the encoder and decoder are capable of activating them [5]. These allow the employment of unrestricted motion vectors, syntax-based arithmetic coding, advanced prediction modes as well as both forward- and backward-frame prediction. The latter two options are similar to the MPEG codec's Predicted (P) and Bidirectional (B) modes.

1.3.7 The ITU H.263+/H.263++ Standards

The H.263+ scheme constitutes version 2 of the H.263 standard [24]. This version was developed by the ITU-T/SG16/Q15 Advanced Video Experts Group, which previously operated under ITU-T/SG15. The technical work was completed in 1997 and was approved in 1998. The H.263+ standard incorporated 12 new optional features in the H.263 codec. These new features support the employment of customized picture sizes and clock frequencies, improve the compression efficiency, and allow for quality, bitrate, and complexity scalability. Furthermore, it has the ability to enhance the attainable error resilience, when communicating over wireless and packet-based networks, while supporting backwards compatibility with the H.263 codec. The H.263++ scheme is version 3 of the H.263 standard, which was developed by ITU-T/SG16/Q15 [26]. Its technical content was completed and approved in late 2000.

1.3.8 The MPEG-4 Standard

The MPEG-4 standard is constituted by a family of audio and video coding standards that are capable of covering an extremely wide bitrate range, spanning from 4800 bit/s to approximately 4 Mbit/s [25]. The primary applications of the MPEG-4 standard are found in Internet-based multimedia streaming and CD distribution, conversational videophone as well as broadcast television.

The MPEG-4 standard family absorbs many of the MPEG-1 and MPEG-2 features, adding new features such as Virtual Reality Markup Language (VRML) support for 3D rendering, object-oriented composite file handling including audio, video, and VRML objects, the support of digital rights management and various other interactive applications.

Most of the optional features included in the MPEG-4 codec may be expected to be exploited in innovative future applications yet to be developed. At the time of writing, there are very few complete implementations of the MPEG-4 standard. Anticipating this, the developers of the standard added the concept of "Profiles" allowing various capabilities to be grouped together.

As mentioned above, the MPEG-4 codec family consists of several standards, which are termed "Layers" and are listed below [25].

- Layer 1: Describes the synchronization and multiplexing of video and audio.
- Layer 2: Compression algorithms for video signals.
- Layer 3: Compression algorithms for perceptual coding of audio signals.
- Layer 4: Describes the procedures derived for compliance testing.
- Layer 5: Describes systems for software simulation of the MPEG-4 framework.
- Layer 6: Describes the Delivery Multimedia Integration Framework (DMIF).

1.3.9 The H.26L/H.264 Standard

Following the finalization of the original H.263 standard designed for videotelephony, which was completed in 1995, the ITU-T Video Coding Experts Group (VCEG) commenced work on two further developments, specifically, a “short-term” effort including adding extra features to the H.263 codec resulting in Version 2 of the standard and a “long-term” effort aiming at developing a new standard specifically designed for low-bitrate visual communications. The long-term effort led to the draft “H.26L” standard, offering a significantly better video compression efficiency than the previous ITU-T standards. In 2001, the ISO MPEG recognized the potential benefits of H.26L and the Joint Video Team (JVT) was formed, including experts from both the MPEG and the VCEG. The JVT’s main task was to develop the draft H.26L model1 into a full international standard. In fact, the outcome of these efforts turned out to be two identical standards, namely the ISO MPEG-4 Part 10 scheme of MPEG-4 and the ITU-T H.264 codec. The official terminology for the new standard is Advanced Video Coding (AVC), although, it is widely known by its old working title of H.26L and by its ITU document number H.264 [62].

In common with earlier standards, such as the MPEG-1, MPEG-2, and MPEG-4 schemes, the H.264 draft standard does not explicitly define an unambiguous coding standard. Rather, it defines the syntax of an encoded video bitstream and the decoding algorithm for this bitstream. The basic functional elements, such as motion prediction, transformation of the motion-compensated error residual, and the quantization of the resultant DCT coefficients as well as their entropy encoding are not unlike those of the previous standards, such as MPEG-1, MPEG-2, MPEG-4, H.261, H.263, etc. The important advances found in the H.264 codec occur in the specific implementation of each functional element. The H.264 codec is described in more detail in Section 12.2.

This book reports on advances attained during the most recent years of the half-a-century history of video communications, focussing on the design aspects of wireless videotelephony, dedicating particular attention to the contradictory design aspects portrayed in Figure 1.5.

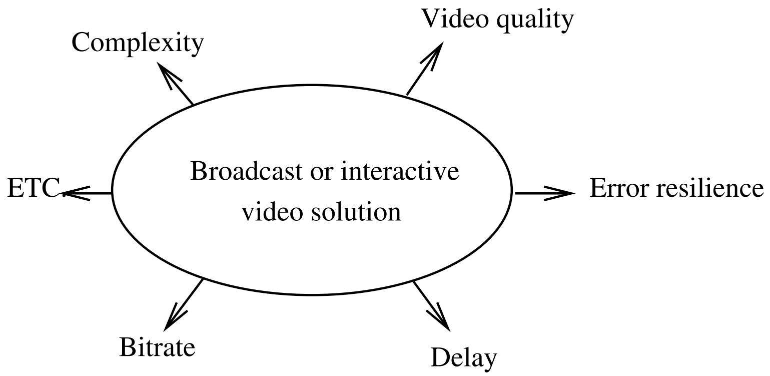

Figure 1.5: Contradictory system design requirements of various video communications systems.

1.4 Video Communications

Video communication over rate-limited and error-prone channels, such as packet networks and wireless links, requires both a high error resilience and high compression. In the past, considerable efforts have been invested in the design and development of the most efficient video compression schemes and standards. For the sake of achieving high compression, most modern codecs employ motion-compensated prediction between video frames, in order to reduce the temporal redundancy, followed by a spatial transform invoked for reducing the spatial redundancy. The resultant parameters are entropy-coded, in order to produce the compressed bitstream. These algorithms provide high compression, however, the compressed signal becomes highly vulnerable to error-induced data losses, which is particularly detrimental when communicating over best-effort networks. In particular, video transmission is different from audio transmission because the dependency across successive video frames is much stronger owing to the employment of inter-frame motion-compensated coding. In this book, a network-adaptive source-coding scheme is proposed for dynamically managing the dependency across packets, so that an attractive trade-off between compression efficiency and error resilience may be achieved. Open standards such as the ITU H.263 [63], H.264 [27] and the ISO/IEC MEPG-4 video codecs [25] were invoked in our proposed schemes.

To address the challenges involved in the design of wireless video transmissions and video streaming, in recent years the research efforts of the community have been directed particularly towards communications efficiency, error resilience, low latency, and scalability [5, 64, 65]. In video communications, postprocessing is also applied at the receiver side for the sake of error concealment and loss recovery. The achievable subjective quality was also improved by an adaptive deblocking filter in the context of the H.264/MPEG-4 video codec [66]. A range of techniques used to recover the damaged video frame areas based on the characteristics of image and video signals have been reviewed in [67]. More specifically, spatial-domain interpolation was used in [68] to recover an impaired macroblock; transform-domain schemes were used to recover the damage inflicted by partially received DCT coefficients, as presented in [69–72]. Temporal-domain schemes interpolate the missing information by exploiting the inherent temporal correlation in adjacent frames. Application examples include, for example, interpolated motion compensation [73, 74] and state recovery [75]. More specifically, the conventional video compression standards employ an architecture, which we refer to as a single-state architecture, because, for example, they have a prediction loop assisted with a single state constituted by the previous decoded frame, which may lead to severe degradation of all subsequent frames, until the corresponding state is reinitialized in the case of loss or corruption. In the state recovery system proposed by Apostolopoulos [75], the problem of having an incorrect state or that of encountering error propagation at the decoder is mitigated by encoding the video into multiple independently decodable streams, each having its own prediction process and state, such that if one stream is lost, the other streams can still be used for producing usable video sequence. Other schemes, such as the temporal smoothness method [76], the coding mode recovery scheme of [72, 76], and the Displaced Frame Difference (DFD) as well as the Motion Vector (MV) recovery management of [77–80] have also resulted in substantial performance. These schemes can also be combined with layered coding, as suggested in [81, 82].

The development of flexible, near-instantaneously adaptive schemes capable of maintaining a perceptually attractive video quality regardless of the channel quality encountered is one of the contributions of this book. Recently, significant research interests have also been devoted to Burst-by-Burst Adaptive Quadrature Amplitude Modulation (BbB-AQAM) transceivers [83, 84], where the transceiver reconfigures itself on a near-instantaneous basis, depending on the prevalent perceived wireless channel quality. Modulation schemes of different robustness and different data throughput have also been investigated [85, 86]. The BbB-AQAM principles have also been applied to Joint Detection Code Division Multiple Access (JD-CDMA) [83, 87] and OFDM [88]. A range of other adaptive video transmission schemes have been proposed for the sake of reducing the transmission delay and the effective of packet loss by Girod and co-workers [89, 90].

Video communication typically requires higher data transmission rates than other sources, such as, audio or text. A variety of video communications schemes have been proposed for increasing the robustness and efficiency of communication [91–95]. Many of the recent proposals employ Rate Distortion (R-D) optimization techniques for improving the achievable compression efficiency [96–98], as well as for increasing the error resilience, when communicating over lossy networks [99, 100]. The goal of these optimization algorithms is to jointly minimize the total video distortion imposed by both compression and channel effects, subject to a given total bitrate constraint. A specific example of recent work in this area is related to intra/inter-mode switching [101, 102], where intra-frame coded macroblocks are transmitted according to the prevalent network conditions for mitigating the effects of error propagation across consecutive video frames. More specifically, an algorithm has been proposed in [102–104] for optimal intra/inter-mode switching, which relies on estimating the overall distortion imposed by quantization, error propagation, and error concealment.

A sophisticated channel coding module invoked in a robust video communication system may incorporate both Forward Error Correction (FEC) and Automatic Re-transmission on Request (ARQ), provided that the ARQ-delay does not affect “lip-synchronization”. Missing or corrupted packets may be recovered at the receiver, as long as a sufficiently high fraction of packets is received without errors [5, 105, 106]. In particular, Reed–Solomon (RS) codes are suitable for this application as a benefit of their convenient features [107, 108]. FEC is also widely used for providing Unequal Error Protection (UEP), where the more vulnerable bits are protected by stronger FEC codes. Recent work has addressed the problem of how much redundancy should be added and distributed across different by prioritized data partitions [108–112]. In addition to FEC codes, data randomization and interleaving have also been employed for providing enhanced protection [75, 113, 114]. ARQ techniques incorporate channel feedback and employ the retransmission of erroneous data [5, 115–118]. More explicitly, ARQ systems use packet acknowledgments and time-outs for controlling which particular packets should be retransmitted. Unlike FEC schemes, ARQ intrinsically adapts to the varying channel conditions and hence in many applications tends to be more efficient. However, in the context of real-time communication and low-latency streaming the latency introduced by ARQ is a major concern. Layered or scalable coding, combined with transmission prioritization, is another effective approach devised for providing error resilience [73, 109, 119–121]. In a layered scheme, the source signal is encoded such that it generates more than one different significance group or layer, with the base layer containing the most essential information required for media reconstruction at an acceptable quality, while the enhancement layer(s) contains information that may be invoked for reconstruction at an enhanced quality. At high packet loss rates, the more-important, more strongly protected layers can still be recovered, while the less-important layers might not. Commonly used layered techniques may be categorized into temporal scalability [122], spatial scalability [123, 124], Signal-to-Noise Ratio (SNR) scalability [25], data partitioning [27], or any combinations of these. Layered scalable coding has been widely employed for video streaming over best-effort networks, including the Internet and wireless networks [121, 125–128]. Different layers can be transmitted under the control of a built-in prioritization mechanism without network support, such as the UEP scheme mentioned above, or using network architectures capable of providing various different Quality of Service (QoS) [129–132]. A scheme designed for optimal intra/inter-mode selection has recently been proposed for scalable coding, in order to limit the inter-frame error propagation inflicted by packet losses [133]. Another scheme devised for adaptive bitrate allocation in the context of scalable coding was presented in [134]. Layered scalable coding has become part of various established video coding standards, such as the members of the MPEG [25,30,31] and H.263+ codec family [24].

Dogan *et al.* [135] reported promising results on adopting the MPEG-4 codec for wireless applications by exploiting the rate control features of video transcoders and combined them with error-resilient General Packet Radio Service (GPRS) type mobile access networks.

The employment of bidirectionally predicted pictures during the encoding process is capable of substantially improving the compression efficiency, because they are encoded using both past and future pictures as references. The efficiency of both forward and backward prediction was studied as early as 1985 by Musmann *et al.* [136]. Recent developments have been applied to the H.264/MPEG codecs, amongst others by Flierl and Girod [137] and Shanableh and Ghanbari [138]. In order to achieve even higher compression in video coding, Al-Mualla, Canagarajah and Bull [139] proposed a fast Block Matching Motion Estimation (BMME) algorithm referred to as the Simplex Minimization Search (SMS), which was incorporated into various video coding standards such as the H.261, H.263, MPEG-1, and MPEG-2 for the sake of both single or multiple reference aided motion estimation.

A plethora of video coding techniques designed for the H.264 standard have been proposed also in the excellent special issues edited by Luthra, Sullivan and Wiegand [140]. At the time of writing there are numerous ongoing research initiatives with the objective of improving the attainable video transmission in wireless environments. Wenger [141] discussed the transmission of H.264 encoded video over IP networks while Stockhammer *et al.* [142] studied the transmission of H.264 bitstreams over wireless environments. Specifically, the design of the H.264 codec specifies a video coding layer and a network adaptation layer, which facilitate the transmission of the bitstream in a network-friendly fashion. As for the wireless networking area, a recent publication of Arumugan *et al.* [143] investigated the coexistence of 802.11g WLANs and high data rate Bluetooth-enabled consumer electronic devices in both indoor home and office environments.

Further important contributions in the area of joint source and channel coding entail the development of a scheme that offers the end-to-end joint optimization of source coding and channel coding/modulation over wireless links, for example those by Thobaben and Kliewer [144] or Murad and Fuja [145].

1.5 Organization of the Monograph

- **Part I** of the book is dedicated to video compression basics. These introductory topics are revised in the context of a host of fixed but arbitrarily programmable rate video codecs based on fractal coding, on the DCT, on VQ codecs, and quad-tree-based codecs. These video codecs and their associated Quadrature Amplitude Modulated (QAM) video systems are described in Chapters 2–5.

- **Part II** of the book is focussed on high-resolution video coding, encompassing Chapters 6 and 7.

- **Part III** of the book entails Chapters 8–14, which characterize the H.261 and H.263 video codecs, constituting important representatives of the family of hybrid DCT codecs. Hence, the associated findings of these chapters can be readily applied in the context of other hybrid DCT codecs, such as the MPEG family, including the MPEG-1, MPEG-2 and MPEG-4 codecs. Chapters 8–14 also portray the interactions of these hybrid DCT video codecs with HSDPA-style near-instantaneously reconfigurable multimode QAM transceivers.

  Chapter 11 provides an overview of the MPEG-4 video codec. This chapter will assist the reader in following our further elaborations in the forthcoming chapters. Chapter 12 has been divided into two parts, the first half provides an overview of the H.264 video codec, while the second part is constituted by a comparative study of the MPEG-4 and H.264 video codecs.

  Video compression often invokes Variable Length Coding (VLC). All of the standard video codecs have adopted these techniques, because they are capable of dramatically reducing the bitrate. However, VLC techniques render the bitstream vulnerable to transmission errors. In Chapter 13 we show the effects of transmission errors on the MPEG-4 codec, quantifying the sensitivity of the various bits.

  The book concludes in Chapter 14 by offering a range of system design studies related to wideband burst-by-burst (BbB) adaptive TDMA/TDD, OFDM, and CDMA interactive as well as distributive mobile video systems and their performance characterization over highly dispersive transmission media. More specifically, both H.263/H.264 as well as MPEG-4 compression-based interactive videophone schemes are proposed and investigated, using BbB adaptive High Speed Downlink Packet Access (HSDPA) style iterative detection-aided turbo transceivers. Amongst a range of other system design examples, Coded Modulation-Aided JD-CDMA video transceivers are investigated, which are capable of near-instantaneously dropping as well as increasing their source coding rate and video quality under transceiver control as a function of the near-instantaneous channel quality.

  Several MPEG-4 compression based videophone schemes are also studied. First of all, we investigate an Iterative Parallel Interference Cancellation (PIC) aided CDMA MPEG-4 videophone scheme. Following this study, we investigated a novel turbo-detected unequal error protection MPEG-4 videophone scheme using a serially concatenated convolutional outer code, trellis-coded modulation-based inner code, and space–time coding. We also proposed a simple packetization scheme, where we partitioned the MPEG-4 bitstream into two bit-sensitivity classes and assigned an Unequal Protection Scheme (UEP).