CHAPTER

An Introduction to Video Coding

1

David R. Bull
Bristol Vision Institute, University of Bristol, Bristol BS8 1UB, UK

Nomenclature

1-D	one dimensional
2-D	two dimensional
3-D	three dimensional
AC	alternating current. Used to denote all transform coefficients except the zero frequency coefficient
ADSL	asymmetric digital subscriber line
ASP	advanced simple profile (of MPEG-4)
AVC	advanced video codec (H.264)
B	bi-coded picture
bpp	bits per pixel
bps	bits per second
CCIR	international radio consultative committee (now ITU)
CIF	common intermediate format
codec	encoder and decoder
CT	computerized tomography
CTU	coding tree unit
CU	coding unit
DC	direct current. Refers to zero frequency transform coefficient.
DCT	discrete cosine transform
DFD	displaced frame difference
DFT	discrete Fourier transform
DPCM	differential pulse code modulation
DVB	digital video broadcasting
EBU	European Broadcasting Union

Academic Press Library in Signal Processing. http://dx.doi.org/10.1016/B978-0-12-420149-1.00001-6

FD	frame difference
fps	frames per second
GOB	group of blocks
GOP	group of pictures
HDTV	high definition television
HEVC	high efficiency video codec (H.265)
HVS	human visual system
I	intra coded picture
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
IP	internet protocol
ISDN	integrated services digital network
ISO	International Standards Organization
ITU	International Telecommunications Union. -R Radio; -T Telecommunications
JPEG	Joint Photographic Experts Group
kbps	kilobits per second
LTE	long term evolution (4G mobile radio technology)
MB	macroblock
mbps	mega bits per second
MC	motion compensation
MCP	motion compensated prediction
ME	motion estimation
MEC	motion estimation and compensation
MPEG	Motion Picture Experts Group
MRI	magnetic resonance imaging
MV	motion vector
P	predicted picture
PSNR	peak signal to noise ratio
QAM	quadrature amplitude modulation
QCIF	quarter CIF resolution
QPSK	quadrature phase shift keying
RGB	red, green, and blue color primaries
SG	study group (of ITU)
SMPTE	Society of Motion Picture and Television Engineers
TV	television

UHDTV	ultra high definition television
UMTS	universal mobile telecommunications system
VDSL	very high bit rate digital subscriber line
VLC	variable length coding
VLD	variable length decoding
YC_bC_r	color coordinate system comprising luminance, Y, and two chrominance channels, C_b and C_r

5.01.1 Introduction

Visual information is the primary consumer of communications bandwidth across all broadcast, internet, and mobile networks. Users are demanding increased video quality, increased quantities of video content, more extensive access, and better reliability. This is creating a major tension between the available capacity per user in the network and the bit rates required to transmit video content at the desired quality. Network operators, content creators, and service providers therefore are all seeking better ways to transmit the highest quality video at the lowest bit rate, something that can only be achieved through video compression.

This chapter provides an introduction to some of the most common image and video compression methods in use today and sets the scene for the rest of the contributions in later chapters. It first explains, in the context of a range of video applications, why compression is needed and what compression ratios are required. It then examines the basic video compression architecture, using the ubiquitous hybrid, block-based motion compensated codec. Finally it briefly examines why standards are so important in supporting interoperability.

This chapter, necessarily only provides an overview of video coding algorithms, and the reader if referred to Ref. [1] for a more comprehensive description of the methods used in today's compression systems.

5.01.2 Applications areas for video coding

By 2020 it is predicted that the number of network-connected devices will reach 1000 times the world's population; there will be 7 trillion connected devices for 7 billion people [2]. Cisco predict [3] that this will result in 1.3 zettabytes of global internet traffic in 2016, with over 80% of this being video traffic. This explosion in video technology and the associated demand for video content are driven by:

- Increased numbers of users with increased expectations of quality and mobility.
- Increased amounts of user generated content available through social networking and download sites.
- The emergence of new ways of working using distributed applications and environments such as the cloud.
- Emerging immersive and interactive entertainment formats for film, television, and streaming.

5.01.2.1 Markets for video technology

A huge and increasing number of applications rely on video technology. These include:

5.01.2.1.1 Consumer video

Entertainment, personal communications, and social interaction provide the primary applications in consumer video, and these will dominate the video landscape of the future. There has, for example, been a massive increase in the consumption and sharing of content on mobile devices and this is likely to be the major driver over the coming years. The key drivers in this sector are:

- Broadcast television, digital cinema and the demand for more immersive content (3-D, multiview, higher resolution, frame rate, and dynamic range).
- Internet streaming, peer to peer distribution, and personal mobile communication systems.
- Social networking, user-generated content, and content-based search and retrieval.
- In-home wireless content distribution systems and gaming.

5.01.2.1.2 Surveillance

We have become increasingly aware of our safety and security, and video monitoring is playing an increasingly important role in this respect. It is estimated that the market for networked cameras (non-consumer) [4] will be $4.5 billion in 2017. Aligned with this, there will be an even larger growth in video analytics. The key drivers in this sector are:

- Surveillance of public spaces and high profile events.
- National security.
- Battlefield situational awareness, threat detection, classification, and tracking.
- Emergency services, including police, ambulance, and fire.

5.01.2.1.3 Business and automation

Visual communications are playing an increasingly important role in business. For example, the demand for higher quality video conferencing and the sharing of visual content have increased. Similarly in the field of automation, vision-based systems are playing a key role in transportation systems and are now underpinning many manufacturing processes, often demanding the storage or distribution of compressed video content. The drivers in this case can be summarized as:

- Video conferencing, tele-working, and other interactive services.
- Publicity, advertising, news, and journalism.
- Design, modeling, simulation.
- Transport systems, including vehicle guidance, assistance, and protection.
- Automated manufacturing and robotic systems.

5.01.2.1.4 Healthcare

Monitoring the health of the population is becoming increasingly dependent on imaging methods to aid diagnoses. Methods such as CT and MRI produce enormous amounts of data for each scan and these need to be stored as efficiently as possible while retaining the highest quality. Video is also becoming

increasingly important as a point-of-care technology for monitoring patients in their own homes. The primary healthcare drivers for compression are:

- Point-of-care monitoring.
- Emergency services and remote diagnoses.
- Tele-surgery.
- Medical imaging.

It is clear that all of the above application areas require considerable trade-offs to be made between cost, complexity, robustness, and performance. These issues are addressed further in the following section.

5.01.3 Requirements of a compression system

5.01.3.1 Requirements

The primary requirement of a video compression system is to produce the highest quality at the lowest bit rate. Other desirable features include:

- **Robustness to loss:** We want to maintain high quality when signals are transmitted over error-prone channels by ensuring that the bitstream is error-resilient.
- **Reconfigurability and flexibility:** To support delivery over time-varying channels or heterogeneous networks.
- **Low complexity:** Particularly for low power portable implementations.
- **Low delay:** To support interactivity.
- **Authentication and rights management:** To support conditional access, content ownership verification, or to detect tampering.
- **Standardization:** To support interoperability.

5.01.3.2 Trade-offs

In practice, it is usual that a compromise must be made in terms of trade-offs between these features because of cost or complexity constraints and because of limited bandwidth or lossy channels. Areas of possible compromise include:

Lossy vs lossless compression: We must exploit any redundancy in the image or video signal in such a way that it delivers the desired compression with the minimum perceived distortion. This usually means that the original signal cannot be perfectly reconstructed.

Rate vs quality: In order to compromise between bit rate and quality, we must trade off parameters such as frame rate, spatial resolution (luma and chroma), dynamic range, prediction mode, and latency. A codec will include a rate-distortion optimization mechanism that will make coding decisions (for example relating to prediction mode, block size, etc.) based on a rate-distortion objective function [1,5,6].

Complexity vs cost: In general, as additional features are incorporated, the video encoder will become more complex. However, more complex architectures invariably are more expensive and may introduce more delay.

Delay vs performance: Low latency is important in interactive applications. However, increased performance can often be obtained if greater latency can be tolerated.

Redundancy vs error resilience: Conventionally in data transmission applications, channel and source coding have been treated independently, with source compression used to remove picture redundancy and error detection and correction mechanisms added to protect the bitstream against errors. However, in the case of video coding, alternative mechanisms exist for making the compressed bitstream more resilient to errors, or dynamic channel conditions, or for concealing errors at the decoder. Some of these are discussed in Chapters 8 and 9.

5.01.3.3 How much do we need to compress?

Typical video compression ratio requirements are currently between 100:1 and 200:1. However this could increase to many hundreds or even thousands to one as new more demanding formats emerge.

5.01.3.3.1 Bit rate requirements

Pictures are normally acquired as an array of color samples, usually based on combinations of the red, green and blue primaries. They are then usually converted to some other more convenient color space, such as Y, C_b, C_r that encodes luminance separately to two color difference signals [1]. Table 1.1 shows typical sampling parameters for a range of common video formats. Without any compression, it can be seen, even for the lower resolution formats, that the bit rate requirements are high—much higher than what is normally provided by today's communication channels. Note that the chrominance signals are encoded at a reduced resolution as indicated by the 4:2:2 and 4:2:0 labels. Also note that two formats are included for the HDTV case (the same could be done for the other formats); for broadcast

Table 1.1 Typical Parameters for Common Digital Video Formats and their (Uncompressed) Bit Rate Requirements

Format	Spatial sampling (V × H)	Temporal sampling (fps)	Raw bit rate (30 fps, 8/10 bits)
UHDTV (4:2:0) (ITU-R 2020)	Lum: 7680 × 4320 Chrom: 3840 × 2160	24, 25, 30, 50, 60,120	14,930 Mbps[a]
HDTV (4:2:0) (ITU-R 709)	Lum: 1920 × 1080 Chrom: 960 × 540	24, 25, 30, 50, 60	933.1 Mbps[a]
HDTV (4:2:2) (ITU-R 709)	Lum: 1920 × 1080 Chrom: 960 × 1080	24, 25, 30, 50, 60	1244.2 Mbps[a]
SDTV (ITU-R 601)	Lum: 720 × 576 Chrom: 360 × 288	25, 30	149.3 Mbps
CIF	Lum: 352 × 288 Chrom: 176 × 144	10–30	36.5 Mbps
QCIF	Lum: 176 × 144 88 × 72	5–30	9.1 Mbps

[a] *Encoding at 10 bits.*

UHDTV = Ultra High Definition Television; HDTV = High Definition Television; SDTV = Standard Definition Television; CIF = Common Intermediate Format; QCIF = Quarter CIF.

quality systems, the 4:2:2 format is actually more representative of the original bit rate as this is what is produced by most high quality cameras. The 4:2:0 format, on the other hand, is that normally employed for transmission after compression.

Finally, it is worth highlighting that the situation is actually worse than that shown in Table 1.1 especially for the new Ultra High Definition (UHDTV) standard [7] where higher frame rates and longer wordlengths will normally be used. For example at 120 frames per second (fps) with a 10 bit wordlength for each sample, the raw bit rate increases to 60 Gbps for a single video stream! This will increase even further if 3-D or multiview formats are employed.

5.01.3.3.2 *Bandwidth availability*

Let us now examine the bandwidths available in typical communication channels as summarized in Table 1.2. This table shows the theoretical maximum bit rates under optimum operating conditions and it should be noted that these are rarely, if ever, achieved in practice. The bit rates available to an individual user at the application layer will normally be significantly lower than the figures quoted in Table 1.2. The effective throughput is influenced by a large range of internal and external factors including: overheads due to link layer and application layer protocols; network contention, congestion, and numbers of users; asymmetry between download and upload rates; and of course the prevailing channel conditions. In particular, as channel conditions deteriorate, modulation and coding schemes will need to be increasingly robust. This will create lower spectral efficiency with increased coding overhead needed in order to maintain a given quality. The number of retransmissions will also inevitably increase as the channel worsens. As an example, DVB-T2 will reduce from 50 Mbps (256QAM @ 5/6 code-rate) to around 7.5 Mbps when channel conditions dictate a change in modulation and coding mode down to 1/2 rate QPSK. Similarly for 802.11n, realistic bandwidths per user can easily reduce well below 10 Mbps. 3G download speeds never offer 384 kbps—more frequently they will be less than 100 kbps.

Consider the example of a digital HDTV transmission at 30 fps using DVB-T2, where the average bit rate allowed in the multiplex (per channel) is 15 Mbps. The raw bit rate, assuming a 4:2:2 original at 10 bits, is approximately 1.244 Gbps, while the actual bandwidth available dictates a bit rate of 15 Mbps. This represents a compression ratio of approximately 83:1. Download sites such as YouTube typically support up to 6 Mbps for HD 1080p format, but more often video downloads will use 360p or 480p (640 × 480 pixels) formats at 30 fps, with a bit rate between 0.5 and 1 Mbps encoded using the H.264/AVC [8] standard. In this case the raw bit rate, assuming color subsampling in 4:2:0 format, will be 110.6 Mbps. As we can see, this is between 100 and 200 times the bit rate supported for transmission.

Table 1.2 Theoretical Bandwidth Characteristics for Common Communication Systems

Communication system	Maximum bandwidth
3G mobile (UMTS)	384 kbps
4G mobile (4 × 4 LTE)	326 Mbps
Broadband (ADSL2)	24 Mbps
Broadband (VSDL2)	100 Mbps
WiFi (IEEE 802.11n)	600 Mbps
Terrestrial TV (DVB-T2 (8 MHz))	50 Mbps

5.01.4 The basics of compression

A simplified block diagram of a video compression system is shown in Figure 1.1. This shows an input being encoded, transmitted, and decoded.

5.01.4.1 Still image encoding

If we ignore the blocks labeled as motion compensation, the diagram in Figure 1.1 describes a still image encoding system, such as that used in JPEG [9]. The intra-frame encoder performs coding of the picture without reference to any other frames. This is normally achieved by exploiting spatial redundancy through transform-based decorrelation followed by variable length symbol encoding (VLC). The image is then conditioned for transmission using some means of error-resilient coding that makes the encoded bitstream more robust to channel errors. At the decoder, the inverse operations are performed and the original image is reconstructed at the output.

5.01.4.2 Video encoding

A video signal can be considered as a sequence of still images, acquired typically at a rate of 24, 25, 30, 50, or 60 fps. Although it is possible to encode a video sequence as a series of still images using intra-frame methods as described above, we can achieve significantly higher coding efficiency if we also exploit the temporal redundancy that exists in most natural video sequences. This is achieved using inter-frame motion prediction as represented by the motion compensation block in Figure 1.1. This block predicts the structure of the incoming video frame based on the contents of previously encoded

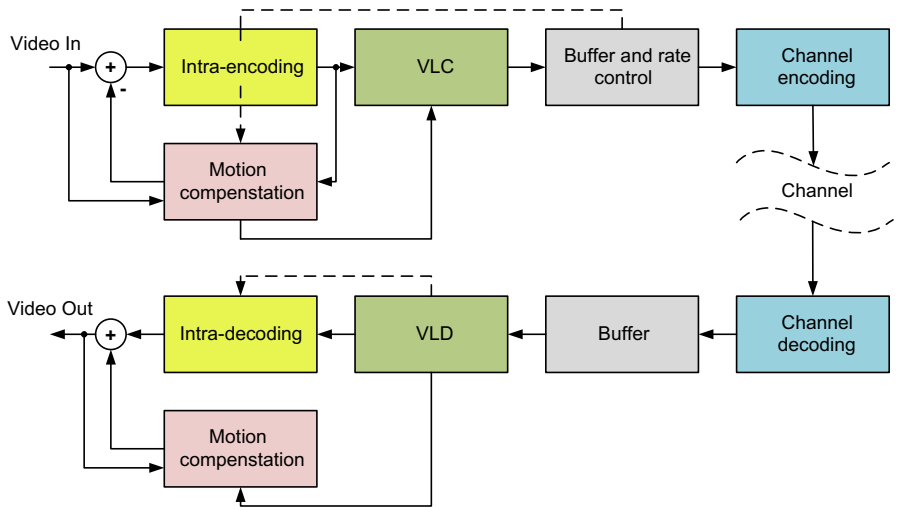

FIGURE 1.1

Simplified high level video compression architecture.

frames. The encoding continues as for the intra-frame case, except this time the intra-frame encoder block processes the low energy residual signal remaining after prediction, rather than the original frame. After variable length encoding, the encoded signal will be buffered prior to transmission. The buffer serves to smooth out content-dependent variations in the output bit rate and the buffer is managed by a rate controller algorithm which adjusts coding parameters in order to match the video output to the instantaneous capacity of the channel.

Because of the reliance on both spatial and temporal prediction, compressed video bitstreams are more prone to channel errors than still images, suffering from temporal as well as spatial error propagation. Methods of mitigating this, making the bitstream more robust and correcting or concealing the resulting artifacts are described in later chapters.

5.01.4.3 Coding units and macroblocks

Video compression algorithms rarely process information at the scale of a picture or a pixel. Instead the coding unit is normally a square block of pixels. In standards up to and including H.264, this took the form of a 16×16 block, comprising luma and chroma information, called a macroblock.

5.01.4.3.1 Macroblocks

A typical macroblock structure is illustrated in Figure 1.2. The macroblock shown corresponds to what is known as a 4:2:0 format [1] and comprises a 16×16 array of luma samples and two subsampled 8×8 arrays of chroma (color difference) samples. This macroblock structure, when coded, must include all of the information needed to reconstruct the spatial detail. For example, this might include transform coefficients, motion vectors, quantizer information, and other information relating to further block partitioning for prediction purposes. A 16×16 block size is normally the base size used for motion estimation; within this, the decorrelating transforms are normally applied at either 8×8 or 4×4 levels.

5.01.4.3.2 Coding tree units

The recent HEVC coding standard [10,11] has extended the size of a macroblock up to 64×64 samples to support higher spatial resolutions, with transform sizes up to 32×32. It also provides much more flexibility in terms of block partitioning to support its various prediction modes. Further details on the HEVC standard are provided in Chapter 3.

5.01.4.4 Video quality assessment

The most obvious way of assessing video quality is to ask a human viewer. Subjective testing methodologies have therefore become an important component in the design and optimization of new compression systems. However, such tests are costly and time consuming and cannot be used for real-time rate-distortion optimization. Hence objective metrics are frequently used instead as these can provide an instantaneous estimate of video quality. These are discussed alongside subjective evaluation methods in Chapter 7. In particular, metrics that can more accurately predict visual quality, aligned with the HVS are highly significant in the context of future coding strategies such as those discussed in Chapters 5 and 6.

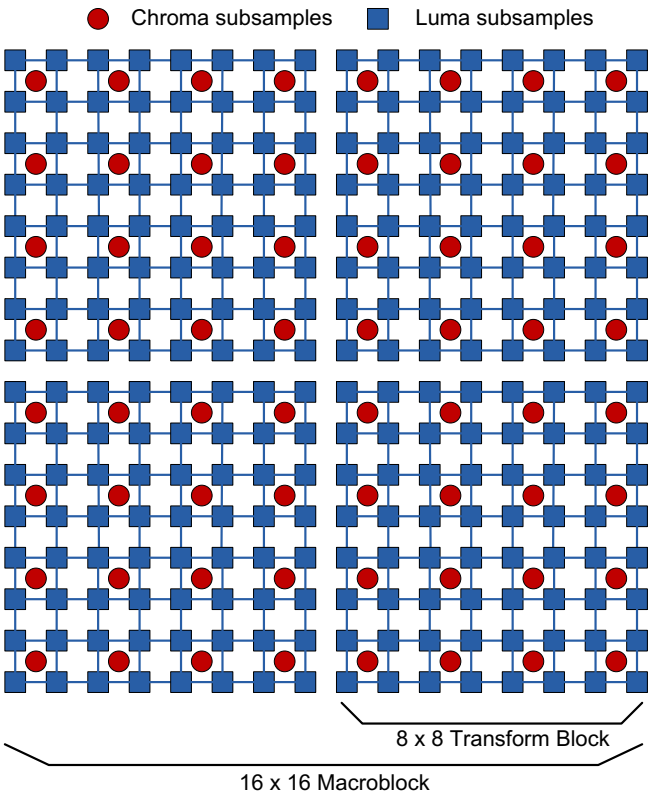

FIGURE 1.2

Typical macroblock structure.

5.01.5 Decorrelating transforms

Transformation presents a convenient basis for compression and this comes about through three mechanisms:

1. It provides data decorrelation and creates a frequency-related distribution of energy allowing low energy coefficients to be discarded.
2. Retained coefficients can be quantized, using a scalar quantizer, according to their perceptual importance.
3. The sparse matrix of all remaining quantized coefficients exhibits symbol redundancy which can be exploited using variable length coding.

For the purposes of transform coding, an input image is normally segmented into small $N \times N$ blocks where the value of N is chosen to provide a compromise between complexity and decorrelation

performance. Transformation maps the raw input data into a representation more amenable to compression. Decorrelating transforms, when applied to correlated data, such as natural images, produce energy compaction in the transform domain coefficients and these can be quantized to reduce the dynamic range of the transformed output, according to a fidelity and/or bit rate criterion. For correlated spatial data, the resulting block of coefficients after quantization will be sparse. Quantization is not a reversible process, hence once quantized, the original signal cannot be perfectly reconstructed and some degree of signal distortion is introduced. This is thus the basis of lossy compression.

5.01.5.1 The discrete cosine transform (DCT)

The discrete cosine transform was first introduced by Ahmed et al. in 1974 [12] and is the most widely used unitary transform for image and video coding applications. Like the discrete Fourier transform, the DCT provides information about a signal in the frequency domain. However, unlike the DFT, the DCT of a real-valued signal is itself real valued and importantly it also does not introduce artifacts due to periodic extension of the input data.

With the DFT, a finite length data sequence is naturally extended by periodic extension. Discontinuities in the time (or spatial) domain therefore produce ringing or spectral leakage in the frequency domain. This can be avoided if the data sequence is symmetrically (rather than periodically) extended prior to application of the DFT. This produces an even sequence which has the added benefit of yielding real-valued coefficients. The DCT is not as useful as the DFT for frequency domain signal analysis due to its deficiencies when representing pure sinusoidal waveforms. However, in its primary role of signal compression, it performs exceptionally well.

As we will see, the DCT has good energy compaction properties and its performance approaches that of the optimum transform for correlated image data. The 1-D DCT, in its most popular form, is given by:

$$c(k) = \sqrt{\frac{2}{N}}\varepsilon_k \sum_{m=0}^{N-1} x[m] \cos\left(\frac{\pi k}{N}\left(m + \frac{1}{2}\right)\right). \tag{1.1}$$

Here N is the transform dimension, $c(k)$ are the transform coefficients, and $x[m]$ are the input data. Similarly the 2-D DCT is given by:

$$c(k,l) = 2\frac{\varepsilon_k \epsilon_l}{\sqrt{NM}} \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n] \cos\left(\frac{\pi k}{N}\left(m + \frac{1}{2}\right)\right) \cos\left(\frac{\pi l}{N}\left(n + \frac{1}{2}\right)\right), \tag{1.2}$$

$$\varepsilon_k = \begin{cases} \frac{1}{\sqrt{2}} & k = 0, \\ 1 & \text{otherwise.} \end{cases}$$

The 2-D DCT basis functions are shown for the case of the 8×8 DCT in Figure 1.3. Further details on the derivation and characteristics of the DCT can be found in [1].

5.01.5.2 Coefficient quantization

Quantization is an important step in lossy compression as it provides the basis for creating a sparse matrix of quantized coefficients that can be efficiently entropy coded for transmission. It is however an irreversible operation and must be carefully managed—one of the challenges is to perform quantization

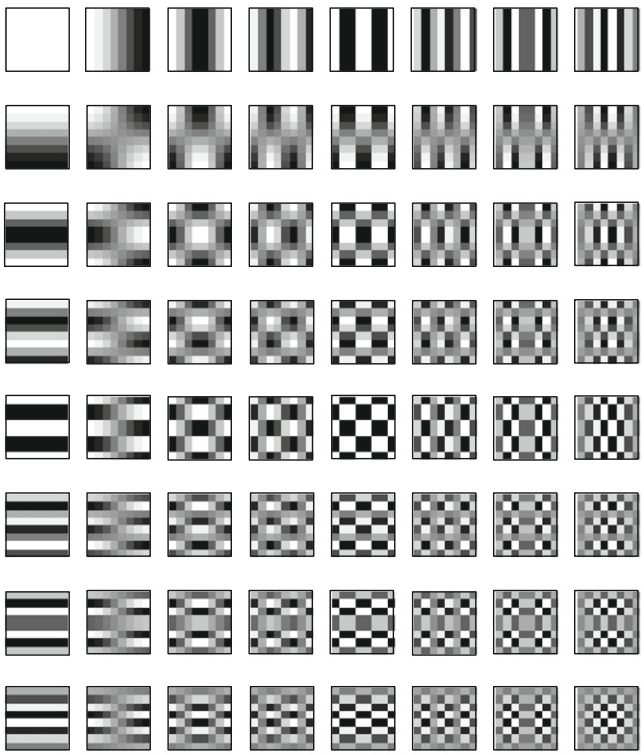

FIGURE 1.3

2-D DCT basis functions for $N = 8$.

in such a way as to minimize its psychovisual impact. The quantizer comprises a set of decision levels and a set of reconstruction levels.

Intra-frame transform coefficients are normally quantized using a uniform quantizer, with the coefficients pre-weighted to reflect the frequency dependent sensitivity of the human visual system. A general expression which captures this is given in Eq. (1.3), where Q is the quantizer step-size, k is a constant, and $\mathbf{W}$ is a coefficient-dependent weighting matrix obtained from psychovisual experiments.

$$c_Q(i,j) = \left\lfloor \frac{kc(i,j)}{QW_{i,j}} \right\rceil. \tag{1.3}$$

After transmission or storage, we must rescale the quantized transform coefficients prior to inverse transformation, thus:

$$\tilde{c}(i,j) = \frac{c_Q(i,j)QW_{i,j}}{k}. \tag{1.4}$$

An example of the effects of coefficient quantization on reconstruction quality for a range of block types is shown in Figure 1.4. It can be observed that more textured blocks require larger numbers of

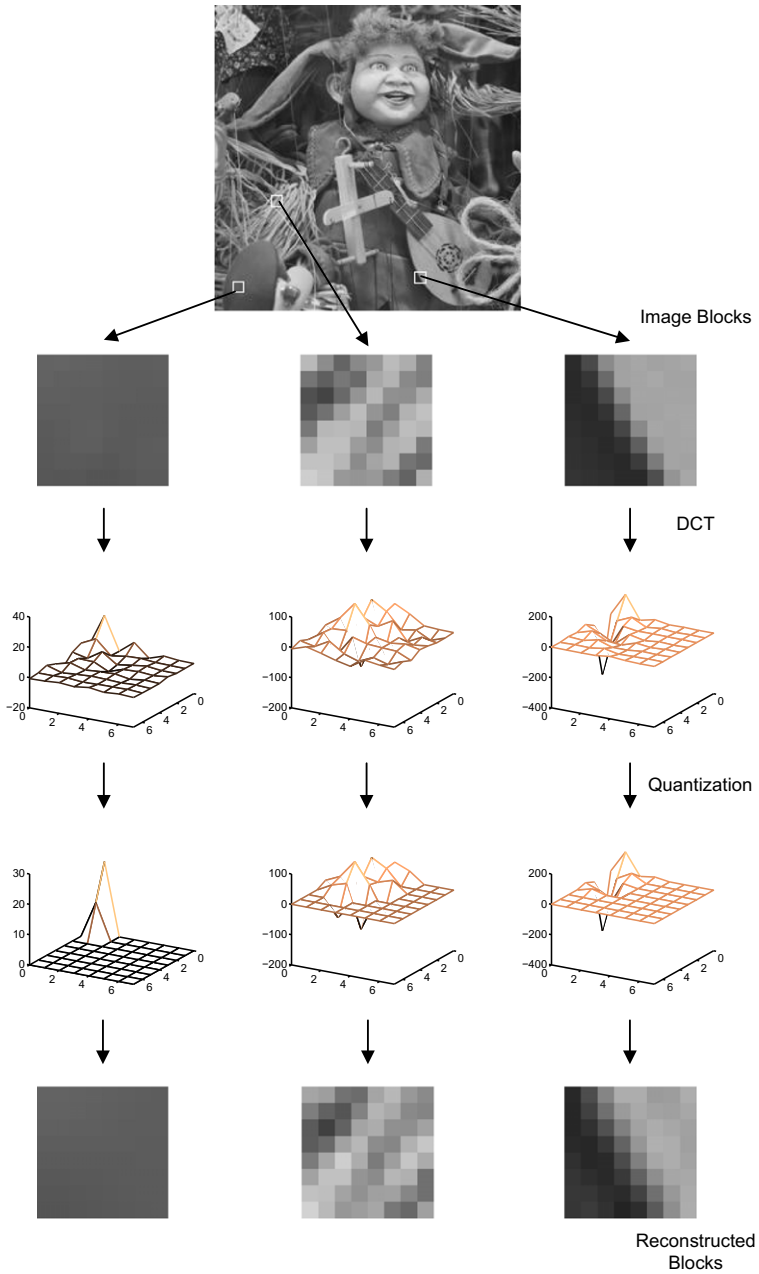

FIGURE 1.4

Effects of coefficient quantization on various types of data block.

coefficients in order to create a good approximation to the original content. The best reconstruction can be achieved with fewer coefficients for the case of untextured blocks, as shown for the left-hand block in the figure.

5.01.6 Symbol encoding

The sparsity of the quantized coefficient matrix can be exploited (typically by run-length coding) to produce a compact sequence of symbols. The symbol encoder assigns a codeword (a binary string) to each symbol. The code is designed to reduce coding redundancy and it normally uses variable length codewords. This operation is reversible.

5.01.6.1 Dealing with sparse matrices

After applying a forward transform and quantization, the resulting matrix contains a relatively small proportion of non-zero entries with most of its energy compacted toward the lower frequencies (i.e. the top left corner of the matrix). In such cases, *run-length coding* (RLC) can be used to efficiently represent long strings of identical values by grouping them into a single symbol which codes the value and the number of repetitions. This is a simple and effective method of reducing redundancies in a sequence.

In order to perform run-length encoding, we need to convert the 2-D coefficient matrix into a 1-D vector and furthermore we want to do this in such a way that maximizes the runs of zeros. Consider for example the 6×6 block of data and its transform coefficients in Figure 1.5. If we scan the matrix using a zig-zag pattern, as shown in the figure, then this is more energy-efficient than scanning by rows or columns.

5.01.6.2 Entropy encoding

Several methods exist that can exploit data statistics during symbol encoding. The most relevant of these in the context of image and video encoding are:

- **Huffman coding** [13]**:** This is a method for coding individual symbols at a rate close to the first-order entropy, often used in conjunction with other techniques in a lossy codec.
- **Arithmetic coding** [14–16]**:** This is a more sophisticated method which is capable of achieving fractional bit rates for symbols, thereby providing greater compression efficiency for more common symbols.

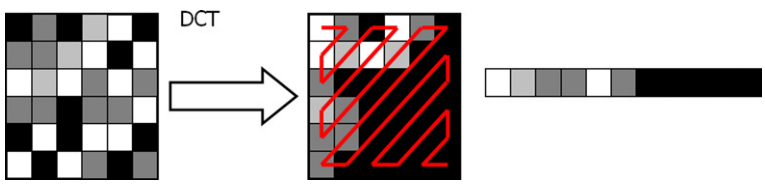

FIGURE 1.5

Zig-zag scanning prior to variable length coding.

Frequently, the above methods are used in combination. For example, DC DCT coefficients are often encoded using a combination of predictive coding (DPCM) and either Huffman or arithmetic coding. Furthermore, motion vectors are similarly encoded using a form of predictive coding to condition the data prior to entropy coding.

5.01.7 Motion estimation

For still natural images, significant spatial redundancies exist and we have seen that these can be exploited via decorrelating transforms, The simplest approach to encoding a sequence of moving images is thus to apply an *intra-frame* (still image) coding method to each frame. This can have some benefits, especially in terms of the error resilience properties of the codec. However it generally results in limited compression performance.

For real-time video transmission over low bandwidth channels, there is often insufficient capacity to code each frame in a video sequence independently (25–30 fps is required to avoid flicker). The solution is thus to exploit the temporal correlation that exists between temporally adjacent frames in a video sequence. This *inter-frame* redundancy can be reduced through motion prediction, resulting in further improvements in coding efficiency.

In motion compensated prediction, a motion model (usually block-based translation only) is assumed and motion estimation (ME) is used to estimate the motion that occurs between the reference frame and the current frame. Once the motion is estimated, a process known as motion compensation (MC) is invoked to use the motion information from ME to modify the contents of the reference frame, according to the motion model, in order to produce a prediction of the current frame. The prediction is called a *motion-compensated prediction* (MCP) or a *displaced frame* (DF). The prediction error is known as the *displaced frame difference* (DFD) signal. Figure 1.6 shows how the pdf of pixel values is modified for FD and DFD frames, compared to an original frame from the *Football* sequence.

A thorough description of motion estimation methods and their performance is provided in Chapter 2 and in [1].

5.01.8 The block-based motion-compensated video coding architecture

5.01.8.1 Picture types and prediction modes

5.01.8.1.1 Prediction modes

Four main classes of prediction are used in video compression:

- **Intra-prediction:** Blocks in the picture are predicted spatially from data adjacent to the current block being coded.
- **Forward prediction:** The reference picture occurs temporally before the current picture.
- **Backward prediction:** The reference picture occurs temporally after the current picture.
- **Bidirectional prediction:** Two (or more) reference pictures (forward and backward) are employed and the candidate predictions are combined in some way to form the final prediction.

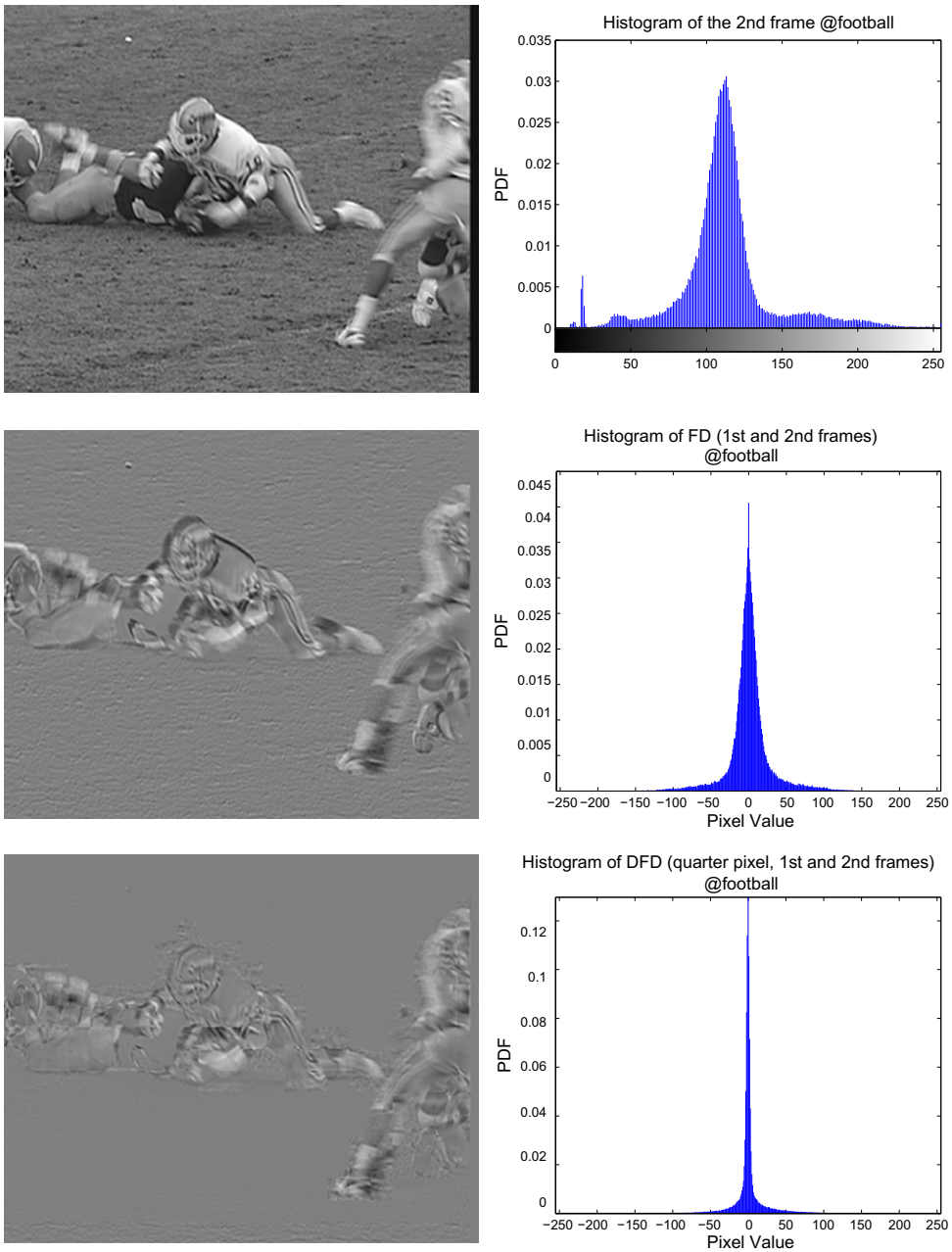

FIGURE 1.6

Probability distributions for an original frame and FD and DFD frames from the *Football* sequence.

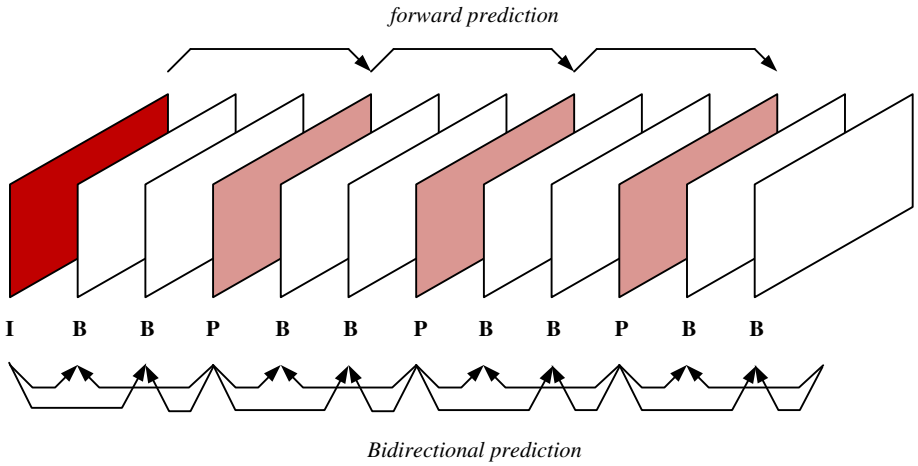

FIGURE 1.7

Typical group of pictures structure.

5.01.8.1.2 *Picture types*

Three major types of picture (or frame) are employed in most video codecs:

- **I-pictures:** These are intra-coded (coded without reference to any other pictures).
- **P-pictures:** These are inter-coded with forward (or backward) prediction from another I- or P-picture.
- **B-pictures:** These are inter-coded with prediction from more than one I- and/or P-picture.

Coded pictures are arranged in a sequence known as a *Group of Pictures* (GOP). A typical GOP structure comprising 12 frames is shown in Figure 1.7. A GOP will contain one I-picture and zero or more P- and B- pictures. The 12 frame GOP in Figure 1.7 is sometimes referred to as an IBBPBBPBBPBB structure and it is clear, for reasons of causality, that the encoding order is different to that shown in the figure since the P-pictures must be encoded prior to the preceding B-pictures.

5.01.8.2 Operation of the video encoder

5.01.8.2.1 *Intra-mode encoding*

A generic structure of a video encoder is given in Figure 1.8. We first describe the operation of the encoder in intra-mode:

1. The inter/intra switch is placed in the intra position.
2. A forward decorrelating transform is performed on the input frame which is then quantized according to the prevailing rate-distortion criteria: $\mathbf{C}_k = Q(\mathrm{DCT}(\mathbf{E}_k))$.
3. The transformed frame is then entropy coded and transmitted to the channel.

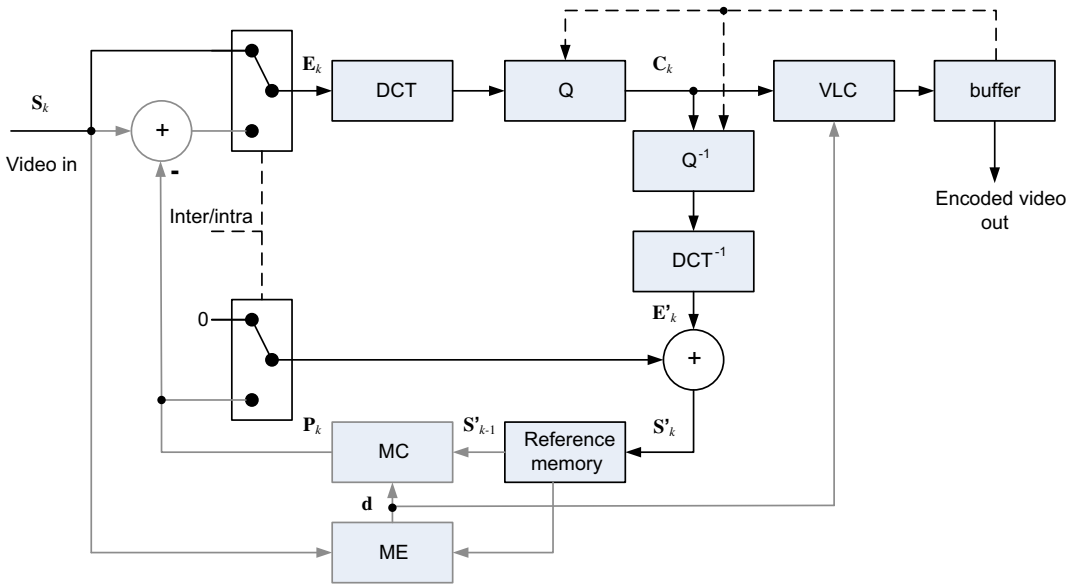

FIGURE 1.8

The block-based motion-compensated video encoder.

4. $\mathbf{C}_k$ is then inverse quantized and inverse DCT'd to produce the same decoded frame pixel values as at the decoder: $\mathbf{E}'_k = \mathrm{DCT}^{-1}(Q^{-1}(\mathbf{C}_k))$.
5. The reference memory is finally updated with the reconstructed frame: $\mathbf{S}'_k = \mathbf{E}'_k$.

5.01.8.2.2 Inter-mode encoding

After the first intra-frame is encoded, the following frames in the GOP will be encoded in inter-mode and this is described below:

1. The inter/intra switch is placed in the inter position.
2. Firstly the motion vector for the current frame is estimated: $\mathbf{d} = \mathrm{ME}(\mathbf{S}_k, \mathbf{S}_{k-1})$.
3. Next the motion compensated prediction frame, $\mathbf{P}_k$, is formed: $\mathbf{P}_k = \mathbf{S}'_{k-1}[\mathbf{p} - \mathbf{d}]$.
4. This is subtracted from the current frame to produce the displaced frame difference (DFD) signal: $\mathbf{E}_k = \mathbf{S}_k - \mathbf{P}_k$.
5. A forward decorrelating transform is then performed on the DFD and the result is quantized according to the prevailing rate-distortion criteria: $\mathbf{C}_k = Q(\mathrm{DCT}(\mathbf{E}_k))$.
6. The transformed DFD, motion vectors and control parameters are then entropy coded and transmitted to the channel.
7. $\mathbf{C}_k$ is then inverse quantized and inverse DCT'd to produce the same decoded frame pixel values as at the decoder: $\mathbf{E}'_k = \mathrm{DCT}^{-1}(Q^{-1}(\mathbf{C}_k))$.
8. Finally the reference memory is updated with the reconstructed frame: $\mathbf{S}'_k = \mathbf{E}'_k + \mathbf{P}_k$.

5.01.8.3 Operation of the video decoder

The structure of the video decoder is illustrated in Figure 1.9 and described below. By comparing the encoder and decoder architectures, it can be seen that the encoder contains a complete replica of the decoder in its prediction feedback loop. This ensures that (in the absence of channel errors) there is no drift between the encoder and decoder operations. Its operation is as follows, firstly in intra-mode:

1. The inter/intra switch is placed in the intra position.
2. Entropy decoding is then performed on the transmitted control parameters and quantized DFD coefficients.
3. The data $\mathbf{C}_k$ is inverse quantized and inverse DCT'd to produce the decoded frame pixel values: $\mathbf{E}'_k = \mathrm{DCT}^{-1}(Q^{-1}(\mathbf{C}_k))$.
4. Finally the reference memory is updated with the reconstructed frame which is also output to a file or display: $\mathbf{S}'_k = \mathbf{E}'_k$.

Similarly in inter-mode:

1. The inter/intra switch is placed in the inter position.
2. Entropy decoding is firstly performed on the control parameters, quantized DFD coefficients, and motion vector.
3. $\mathbf{C}_k$ is then inverse quantized and inverse DCT'd to produce the decoded DFD pixel values: $\mathbf{E}'_k = \mathrm{DCT}^{-1}(Q^{-1}(\mathbf{C}_k))$.

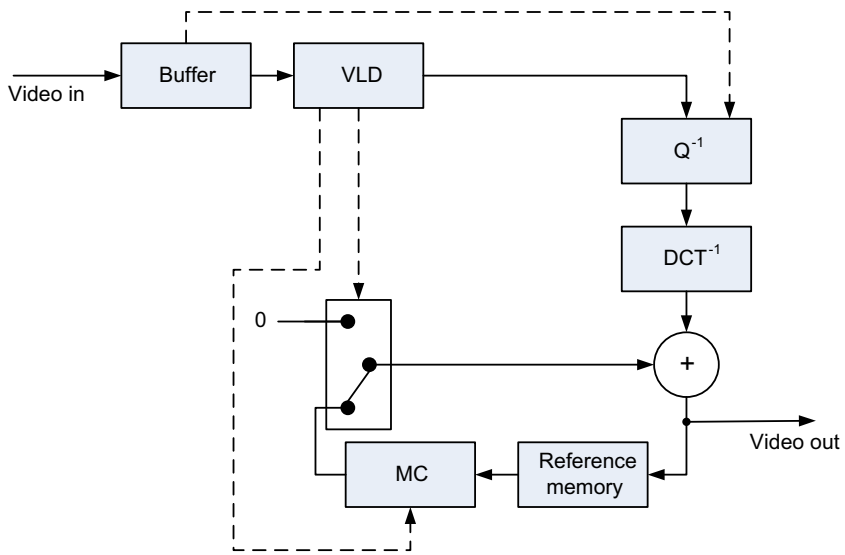

FIGURE 1.9

The block-based motion-compensated video decoder.

4. Next the motion compensated prediction frame, $\mathbf{P}_k$, is formed: $\mathbf{P}_k = \mathbf{S}'_{k-1}[\mathbf{p} - \mathbf{d}]$.
5. Finally the reference memory is updated with the reconstructed frame and this is also output to a file or display: $\mathbf{S}'_k = \mathbf{E}'_k + \mathbf{P}_k$.

5.01.9 Standardization of video coding systems

Standardization of image and video formats and compression methods has been instrumental in the success and universal adoption of video technology. An overview of coding standards is provided below and a more detailed description of the primary features of the most recent standard (HEVC) is provided in Chapter 3.

Standards are essential for interoperability, enabling material from different sources to be processed, and transmitted over a wide range of networks or stored on a wide range of devices. This interoperability opens up an enormous market for video equipment, which can exploit the advantages of volume manufacturing, while also providing the widest possible range of services for users. Video coding standards define the bitstream format and decoding process, not (for the most part) the encoding process. This is illustrated in Figure 1.10. A standard-compliant encoder is thus one that produces a compliant bitstream and a standard-compliant decoder is one that can decode a standard-compliant bitstream. The real challenge lies in the bitstream generation, i.e. the encoding, and this is where manufacturers can differentiate their products in terms of coding efficiency, complexity, or other attributes. Finally it is important to note that the fact that an encoder is standard-compliant, provides no guarantee of absolute video quality.

5.01.9.1 A brief history of video encoding standards

A chronology of video coding standards is represented in Figure 1.11. This shows how the International Standards Organization (ISO) and the International Telecommunications Union (ITU-T) have worked both independently and in collaboration on various standards. In recent years, most ventures have benefited from close collaborative working.

Study Group SG.XV of the CCITT (now ITU-T) produced the first international video coding standard, H.120, in 1984. H.120 addressed videoconferencing applications at 2.048 Mbps and 1.544 Mbps

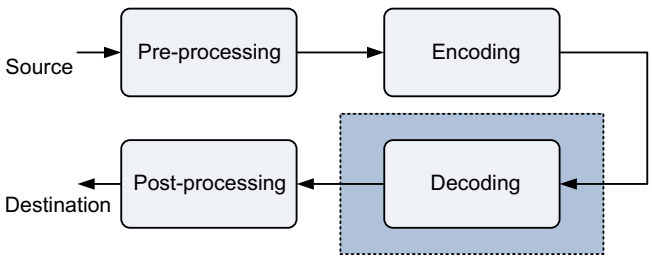

FIGURE 1.10

The scope of standardization.

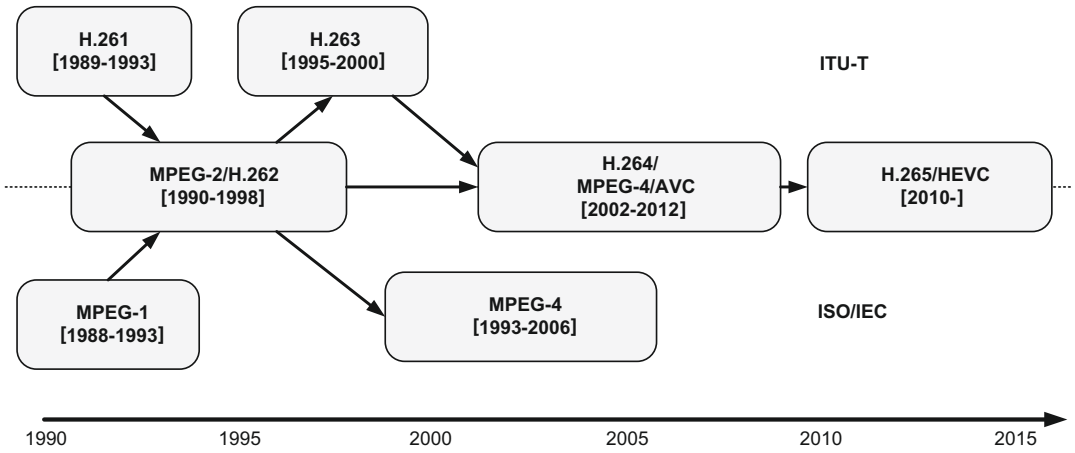

FIGURE 1.11

A chronology of video coding standards from 1990 to the present date.

for 625/50 and 525/60 TV systems respectively. This standard was never a commercial success. H.261 [17] followed this in 1989 with a codec based on a $p \times 64$ kbps ($p = 1 \cdots 30$) targetted at ISDN conferencing applications. This was the first block-based hybrid compression algorithm using a combination of transformation (the Discrete Cosine Transform (DCT)), temporal Differential Pulse Code Modulation (DPCM), and motion compensation. This architecture has stood the test of time as all major video coding standards since have been based on it.

In 1988 the Moving Picture Experts Group (MPEG) was founded, delivering a video coding algorithm targeted at digital storage media at 1.5 Mbs/s in 1992. This was followed in 1994 by MPEG-2 [18], specifically targetted at the emerging digital video broadcasting market. MPEG-2 was instrumental, through its inclusion in all set-top boxes for more than a decade, in truly underpinning the digital broadcasting revolution. A little later in the 1990s ITU-T produced the H.263 standard [19]. This addressed the emerging mobile telephony, internet, and conferencing markets at the time. Although mobile applications were slower than expected to take off, H.263 had a significant impact in conferencing, surveillance, and applications based on the then-new Internet Protocol.

MPEG-4 [20] was a hugely ambitious project that sought to introduce new approaches based on object-based as well as, or instead of, waveform-based methods. It was found to be too complex and only its Advanced Simple Profile (ASP) was used in practice. This formed the basis for the emerging digital camera technology of the time.

Around the same time ITU-T started its work on H.264 and this delivered its standard, in partnership with ISO/IEC, in 2004 [8]. In the same way that MPEG-2 transformed the digital broadcasting landscape, so has H.264/AVC transformed the mobile communications and internet video domains. H.264/AVC is by far the most ubiquitous video coding standard to date. Most recently in 2013, the joint activities of ISO and ITU-T delivered the HEVC standard [10,21], offering bit rate reductions of up to 50% compared with H.264/AVC.

Table 1.3 Comparison of Video Coding Standards for Entertainment Applications. Average Bit-Rate Savings are Shown for Equal Objective Quality Values Measured Using PSNR

Video standard	Relative bit rate savings (%)		
	H.264/AVC	MPEG-4	MPEG-2
HEVC MP	35.4	63.7	70.8
H.264/AVC HP	X	44.5	55.4
MPEG-4 ASP	X	X	19.7
H.263 HLP	X	X	16.2

Further extensions to video compression standards have been developed that enable the efficient processing of new formats, in particular more immersive formats such as stereoscopic 3-D. The compression challenges and architectures associated with stereoscopic and multiview coding are described in detail in Chapter 4.

5.01.9.2 The performance of current standards

An excellent comparison of coding standards is provided by Ohm et al. in [21], where a comprehensive comparison between HEVC and previous standards is reported. We reproduce their results here in Table 1.3, noting that even greater improvements were reported for interactive applications. As a rule of thumb video coding standards have, since the introduction of H.120 in 1984, delivered a halving of bit rate for the equivalent video quality every 10 years. This is evidenced by the results in Table 1.3.

5.01.10 Conclusions

This chapter has introduced the requirements for, and applications of video compression. It has examined the architecture of a compression system in the context of modern communication systems and has shown that its design is often a compromise between coding rate, picture quality, and implementation complexity. The basic operation of a block-based motion compensated video encoding system has been described, explaining its major components, and its operating characteristics. Finally the justification for universal compression standards has been presented, and the performance of recent standards compared in terms of their rate-distortion characteristics.

The remainder of this section will cover many of these topics in more detail and place them in the context of current and future standards, formats, and transmission requirements. Chapter 2 expands on the important area of motion estimation, demonstrating efficient means of achieving temporal decorrelation in the coding process. Chapter 3 provides the reader with an overview of the new HEVC standard and Chapter 4 considers the extensions to compression mechanisms that are required to deal with multiview and stereoscopic content. Chapters 5 and 6, provide an insight into future coding possibilities based on an increased awareness of perceptual properties an limitations. Chapter 7 covers the important

area of how we can assess video quality, both for comparing the performance of different codecs and for optimizing coding decisions within the compression process. Finally, Chapters 8 and 9 address the topics of content delivery, network adaptation, and error concealment.

Additional resources

1. http://www.poynton.com/Poynton-video-eng.html. Lots of information on color conversion, formats, and video preprocessing.
2. http://mpeg.chiariglione.org/. This is the home page of the Moving Picture Experts Group (MPEG), a working group of ISO/IEC with the mission to develop standards for coded representation of digital audio and video and related data. Lots of information on standards with tutorial content.

Glossary of terms

1080p30	a means of representing the sampling structure of the video format. In this case representing an HD format of 1080 lines with progressive temporal sampling at 30 fps
Discrete cosine transform	a popular decorrelating transform used in image and video coding. Achieves close to optimum performance with a fixed set of basis functions
Entropy coding	the process used to efficiently encode symbols formed from scans of quantized transform coefficients (or motion vectors). A reversible process, usually performed using Huffman or arithmetic coding
Error resilience	the process of making an encoded bitstream more robust to the effects of loss during transmission
Lossless coding	a reversible coding process where the original signal can be exactly reconstructed after compression
Lossy coding	an irreversible coding process where the original signal cannot be exactly reconstructed after compression. Distortions are introduced during coding
Macroblock	the basic coding unit used in many standards. Typically comprising (for 4:2:0) a luma block of 16 by 16 samples and two chroma blocks each of 8 by 8 samples
Motion estimation	the process of temporally predicting the current picture based on the content of other pictures in the sequence
Quantization	the process of approximating the output from the decorrelating transform. A primary mechanism for achieving rate-distortion trade-offs during coding
Streaming	compressed video delivery from a server. Media is constantly received by and presented to an end-user while being delivered
Zettabyte	10^{23} bytes
X:Y:Z	a means of describing the color subsampling method used in a format or by a codec. e.g. 4:2:0 means that the chroma (color difference signals are subsampled by a factor of two horizontally and vertically prior to coding or transmission

References

[1] D. Bull, Communicating Pictures, Academic Press, 2014.

[2] <http://www.wireless-world-research.org/fileadmin/sites/default/files/publications/-Outlook/Outlook4.pdf>.

[3] Cisco Visual Networking Index: Forecast and Methodology, 2011–2016 (update 2012–2017). <http://www.cisco.com/en/US/netsol/ns827/networking_solutions_sub_solution.html>.

[4] Network Camera and Video Analytics Market – Global Forecast, Trend & Analysis – Segmentation by Technology, Function, Resolution, Product & Service Type, System Architecture, Verticals, Application and Geography (2012–2017) Report by marketsandmarkets.com, Report Code: SE 1238, 2012.

[5] A. Ortega, K. Ramchandran, Rate-distortion methods for image and video compression, IEEE Signal Process. Mag. 15 (6) (1998) 23–50.

[6] G. Sullivan, T. Wiegand, Rate-distortion optimization for video compression, IEEE Signal Process. Mag. 15 (6) (1998) 74–90.

[7] Recommendation ITU-R BT.2020 (08/2012), Parameter Values for Ultra-High Definition Television Systems for Production and International Programme Exchange, ITU-R, 2012.

[8] ITU-T and ISO/IEC JTC 1, Advanced Video Coding for Generic Audiovisual Services, ITU-T Rec. H.264 and ISO/IEC 14496-10 (AVC), Version 1, 2003; Version 2, 2004; Versions 3, 4, 2005; Versions 5, 6, 2006; Versions 7, 8, 2007; Versions 9, 10, 11, 2009; Versions 12, 13, 2010; Versions 14, 15, 2011; Version 16, 2012.

[9] ISO/IEC International Standard 10918-1, Information Technology – Digital and Coding of Continuous-Tone Still Images – Requirements and Guidelines, 1992.

[10] G. Sullivan, J.-R. Ohm, W. Han, T. Wiegand, Overview of the high efficiency video coding (HEVC) standard, IEEE Trans. Circ. Syst. Video Technol. 22 (12) (2012) 1648–1667.

[11] Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 ISO/IEC 23008-2 and ITU-T Recommendation H.265, High Efficiency Video Coding (HEVC), January 2013.

[12] N. Ahmed, T. Natarajn, K. Rao, Discrete cosine transform, IEEE Trans. Comput. 23 (1974) 90–93.

[13] D.A. Huffman, A method for the construction of minimum-redundancy codes, Proc. IRE 40 (9) (1952) 1098–1101.

[14] N. Abramson, Information Theory and Coding, McGraw-Hill, 1963.

[15] J. Rissanen, Generalized Kraft inequality and arithmetic coding, IBM J. Res. Dev. 20 (1976) 198–203.

[16] A. Said, Introduction to arithmetic coding – theory and practice, in: K. Sayood (Ed.), Lossless Compression Handbook, Academic Press, 2003.

[17] Int. Telecommun. Union-Telecommun. (ITU-T), Recommendation H.261, Video Codec for Audiovisual Services at $p \times 64$ kbit/s, Version 1, 1990; Version 2, 1993.

[18] ITU-T and ISO/IEC JTC 1, Generic Coding of Moving Pictures and Associated Audio Information – Part 2: Video, ITU-T Rec. H.262 and ISO/IEC 13818-2 (MPEG-2 Video), Version 1, 1994.

[19] ITU-T, Video Coding for Low Bitrate Communication, ITU-T Rec. H.263, Version 1, 1995; Version 2, 1998; Version 3, 2000.

[20] ISO/IEC JTC 1, Coding of Audio-Visual Objects – Part 2: Visual, ISO/IEC 14496-2 (MPEG-4 Visual), Version 1, 1999; Version 2, 2000; Version 3, 2004.

[21] J.-R. Ohm, G. Sullivan, H. Schwartz, T. Tan, T. Wiegand, Comparison of the coding efficiency of video coding standards-including high efficiency video coding (HEVC), IEEE Trans. Circ. Syst. Video Technol. 22 (12) (2012) 1669–1684.