CHAPTER 3

Multimedia Processing in Communications

Chapter Overview

Multimedia has at its very core the field of signal-processing technology. With the exploding growth of the Internet, the field of multimedia processing in communications is becoming more and more exciting. Although multimedia leverages numerous disciplines, signal processing is the most relevant. Some of the basic concepts, such as spectral analysis, sampling theory and partial differential equations, have become the fundamental building blocks for numerous applications and, subsequently, have been applied in such diverse areas as transform coding, display technology and neural networks. The diverse signal-processing algorithms, concepts and applications are interconnected and, in numerous instances, appear in various reincarnated forms.

This chapter is organized as follows. First, we present and analyze digital media and signal processing elements. To address the challenges of multimedia signal processing while providing higher interactivity levels with the media and increased capabilities to access a wide range of applications, multimedia signal-processing methods must allow efficient access to processing and retrieval of multimedia content. Then, we review audio and video coding. During the last decade new digital audio and video applications have emerged for network, wireless, and multimedia computing sys-

tems and face such constraints as reduced channel bandwidth, limited storage capacity and low cost. New applications have created a demand for high-quality digital audio and video delivery. In response to this need, considerable research has been devoted to the development of algorithms for perceptually transparent coding of high-fidelity multimedia.

Next, we describe a general framework for image copyright protection through digital watermarking. In particular, we present the main features of an efficient watermarking scheme and discuss robustness issues. The watermarking technique that has been proposed is to hide secret information in the signal so as to discourage unauthorized copying or to attest the origin of the media. Data embedding and watermarking algorithms embed text, binary streams, audio, image or video in a host audio, image or video signal. The embedded data is perceptually inaudible or invisible to maintain the quality of the source data.

We also review the key attributes of neural processing essential to intelligent multimedia processing. The objective is to show why NNs are a core technology for efficient representation for audio-visual information. Also, we will demonstrate how the adaptive NN technology presents a unified solution to a broad spectrum of multimedia applications (image visualization, tracking of moving objects, subject-based retrieval, face-based indexing and browsing and so forth).

Finally, this chapter concludes with a discussion of recent large-scale integration programmable processors designed for multimedia processing, such as real-time compression and decompression of audio and video as well as the next generation of computer graphics. Because the target of these processors is to handle audio and video in real time, the promising capability must be increased compared to that of conventional microprocessors, which were designed to handle mainly texts, figures, tables and photographs. To clarify the advantages of a high-speed multimedia processing capability, we define these chips as multimedia processors. Recent general-purpose microprocessors for workstations and personal computers use special built-in hardware for multimedia processing.

3.1 Introduction

Multimedia signal processing is more than simply putting together text, audio, images and video. It is the integration and interaction among these different media that creates new systems and new research challenges and opportunities. Although multimedia leverages numerous disciplines, signal processing is the most relevant. Some of the basic concepts, such as spectral analysis, sampling theory and partial differential equation theory, have become the fundamental building blocks for numerous applications and, subsequently, have been reinvented in such diverse areas as transform coding, display technology and NNs. The diverse signal-processing algorithms, concepts and applications are interconnected.

The term "multimedia" represents many different concepts. It includes basic elementary components, such as different audio types. These basic components may originate from many diverse sources (individuals or synthetic). For audio, the synthetics may be traditional musical presentation. One may also argue that multimedia is based on the extended visual experience, which includes representation of the real world, as well as its model, through a synthetic representation.

The "multimedia" technologies have dramatically changed and will keep changing. However, it is erroneous to favor advances simply because the final product is based on better technology.

Multimedia consists of `{multimedia data}+{set of instructions}`. Multimedia data is informally considered as the collection of the three multimedia data, that is, multisource, multitype and multiformat data [3.1]. The interactions among the multimedia components consist of complex relationships without which multimedia could be a simple set of visual, audio and often data [3.2].

We define multimedia signal processing as the representation, interpretation, encoding and decoding of multimedia data using signal-processing tools. The goal of multimedia signal processing is effective and efficient access, manipulation, exchange and storage of multimedia content for various multimedia applications [3.3].

The Technical Committee (TC) on MMSP is the youngest TC in the IEEE Signal Processing (SP) society. It took them a long time to raise some questions like the following:

- What is multimedia signal processing all about?
- What impact has signal processing brought to multimedia technologies?
- Where are the multimedia technologies now?

Multimedia signal-processing technologies will play major roles in the multimedia-network age. Researchers today working in this area have the privilege of selecting the future direction of MMSP technologies, so what they are doing will deeply influence our future society.

3.2 Digital Media

Digital media take advantage of advances in computer-processing techniques and inherit their strength from digital signals. The following distinguishing features make them superior to the analog media:

- *Robustness*—The quality of digital media will not degrade as copies are made. They are most stable and more immune to the noises and errors that occur during processing and transmission. Analog signals suffer from signal-path attenuation and generation loss (as copies are made) and are influenced by the characteristics of the medium itself.
- *Seamless integration*—This involves the integration of different media through digital storage and processing and transmission technologies, regardless of the particular media properties. Therefore, digital media eliminate device dependency in an integrated environment and allow easy data composition of nonlinear editing.
- *Reusability and interchangeability*—With the development of standards for the common exchange formats, digital media have greater potential to be reused and shared by multiple users.
- *Ease of distributed potential*—Thousands of copies may be distributed electronically by a simple command.

Digital image Digital images are captured directly by a digital camera or indirectly by scanning a photograph with a scanner. They are displayed on the screen or printed.

Digital images are composed of a collection of pixels that are arranged as a 2D matrix. This 2D or spatial representation is called the image resolution. Each pixel consists of three components: red (R), green (G) and blue (B). On a screen, each component of a pixel corresponds to a phosphor. A phosphor glows when excited by an electron gun. Various combinations of different RGB intensities produce different colors. The number of bits to represent a pixel is called the color depth, which decides the actual number of colors available to represent a pixel. Color depth is in turn determined by the size of the video buffer in the display circuitry.

The resolution and color depth determine the presentation quality and the size of image storage. The more pixels and the more colors there are means the better the quality and the larger the volume. To reduce the storage requirement, three different approaches can be used:

- *Index color*—This approach reduces the storage size by using a limited number of bits with a color lookup table (or color palette) to represent a pixel. Dithering can be applied to create additional colors by blending colors from the palette. This is a technique taking advantage of the fact that the human brain perceives the media color when two different colors are adjacent to one another. With palette optimization and color dithering, the range of the overall color available is still considerable, and the storage is reduced.
- *Color subsampling*—Humans perceive color as brightness, hue and saturation rather than as RGB components. Human vision is more sensitive to variation in the luminance (or brightness) than in the chrominance (or color difference). To take advantage of such differences in the human eye, light can be separated into the luminance and chrominance components instead of the RGB components. The color subsampling approach shrinks the file size by down-sampling the chrominance components, that is, using less

bits to represent the chrominance components while having the luminance component unchanged.

- *Spatial reduction*—This approach, known as data compression, reduces the size by throwing away the spatial redundancy within the images.

Digital video Video is composed of a series of still-image frames and produces the illusion of movement by quickly displaying frames one after another. The Human Visual System (HVS) accepts anything more than 20 Frames Per Second (fps) as smooth motion. Television and video are usually distinguished. Television is often associated with the concept of broadcast or cable delivery of programs, whereas video allows more user interactivity, such as recording, editing and viewing at a user-selected time.

The biggest challenges posed by digital video are the massive volume of data involved and the need to meet the real-time constraints on retrieval, delivery and display. The solution entails the compromise in the presentation quality and video compression. As for the compromise in the presentation quality, instead of video with full frame, full fidelity and full motion, one may reduce the image size, use less bits to represent colors, or reduce the frame rate. To reduce the massive volume of digital video data, compression techniques with high compression ratios are required. In addition to throwing away the spatial and color similarities of individual images, the temporal redundancies between adjacent video frames are eliminated.

Digital audio Sound waves generate air pressure oscillations that stimulate the human auditory system. The human ear is an example of a transducer. It transforms sound waves to signals recognizable by brain neurons. As with other audio transducers, two important considerations are frequency response and dynamic range. Frequency response refers to the range of frequencies that a medium can reproduce accurately. The frequency range of human hearing is between 20 Hz and 20 KHz. Dynamic range describes the spectrum of the softest to the loudest sound-amplitude levels that a medium can reproduce. Human hearing can accommodate a dynamic range greater than a factor of millions. Sound amplitudes are perceived in logarithmic ratio rather than linearly. Humans perceive sounds across the entire range of 120 dB, the upper limit of which will be painful to humans. Sound waves are characterized in terms of frequency (Hz), amplitude (dB) and phase (degree), whereas frequencies and amplitudes are perceived as pitch and loudness, respectively. Pure tone is a sine wave. Sound waves are additive. In general, sounds are represented by a sum of sine waves. Phase refers to the relative delay between two waveforms. Distortion can result from phase shifts.

Digital audio systems are designed to make use of the range of human hearing. The frequency response of a digital audio system is determined by the sampling rate, which in turn is determined by the Nyquist theorem.

Example 3.1 The sampling rate of Compact Disk (CD) quality audio is 44.1 KHz. Thus, it can accommodate the highest frequency of human hearing, namely, 20 KHz. Telephone quality sound adopts an 8 KHz sampling rate. This can accommodate the most sensitive frequency of human hearing, up to 4 KHz.

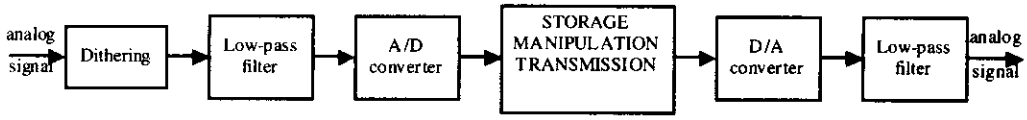

Figure 3.1 Block diagram for digital audio signal processing.

Digital audio aliasing is introduced when one attempts to record frequencies that exceed half the sampling rate. A solution is to use a low-pass filter to eliminate frequencies higher than the Nyquist rate. The quantization interval, or the difference in value between two adjacent quantization levels, is a function of the number of bits per sample and determines the dynamic range. One bit yields 6 dB of dynamic range. For example, 16 bits audio contributes 96 dB of the dynamic range found in CD-grade audio, which is nearly the dynamic range of human hearing. The quantized samples can be encoded in various formats, such as Pulse Code Modulation (PCM), to be stored or transmitted. Quantization noise occurs when the bit number is too small. Dithering, which adds white noise to the input analog signals, may be used to reduce quantization noise. In addition, a low-pass filter can be employed prior to the digital-to-analog (D/A) stage to smooth the stairstep effect resulting from the combination of a low sampling rate and quantization. Figure 3.1 summarizes the basic steps for processing digital audio signals [3.4].

The quality of digital audio is characterized by the sampling rate, the quantization interval and the number of channels. The higher the sampling rate, the more bits per sample and the more channels means the higher the quality of the digital audio and the higher the storage and bandwidth requirements.

Example 3.2 A 44.1 KHz sampling rate, 16-bit quantization and stereo audio reception produce CD-quality audio, but require a bandwidth of 44,100x16x2=1.4 Mb/s. Telephone-quality audio, with a sampling rate of 8 KHz, 8-bit quantization and mono audio reception, needs only a data throughput of 8,000x8x1=64 Kb/s. Digital audio compression or a compromise in quality can be applied to reduce the file size.

Integrated media systems will only achieve their potential if they are truly integrated in three key ways: integration of content, integration with human users and integration with other media systems. First, such systems must successfully combine digital video and audio, text, animation and graphics and knowledge about such information units and their inter-relationships in real time. Second, they must integrate with the individual user by cooperatively interactive multidimensional dynamic interfaces. Third, integrated media systems must connect with other such systems and content-addressable multimedia databases, both logically (information sharing) and physically (information networking, compression and delivery).

3.3 Signal-Processing Elements

Many classical signal-processing procedures have become deeply embedded in the multidimensional fields. A key driver is optimization for representation of multimedia components, as well as the associated storage and delivery requirements. The optimization procedures range from very simple to sophisticated. Some of the principal techniques are the following:

- Nonlinear analog (video and audio) mapping
- Quantization of the analog signal
- Statistical characterization
- Motion representation and models
- 3D representations
- Color processing

A nonlinear analog (video and audio) mapping procedure may be purely analog. Its intention may be the desire to enhance the delivery process. It could also be introduced to mask the limitations of various components of the overall multimedia chain. Typical constraints are introduced by bandwidth limitations and constrained dynamic range in the display terminal.

Quantization of the analog signal is fundamental to any digital representation that has originated in the analog world. The quantization process is an inherently lossy procedure and fundamentally noninvertible. This classical signal-processing element still remains the basic constraint in limiting performance, although not very exciting compared with other multimedia issues [3.5]. Quantization techniques comprise a whole field by themselves. The major relevant issues include uniform and nonuniform techniques and adaptive and nonadaptive procedures [3.6].

Statistical concepts and applications are directly and indirectly strongly embedded in processing components associated with multimedia. This relevant field is part of classical signal processing, and we can only highlight the major categories. A spectral analysis is fundamental to the entire range of image models for filtering and algorithm design. The procedures are critical to both visual and audio data components [3.7, 3.8]. Statistical redundancy is the basic concept upon which the entire field of data compression is based. Mathematical extension of the concept leads to optimum transform for decorrelation. This in turn leads to the entire field of modern transform-coding technology [3.9]. Model-based representations, primarily for compression, are determined from assumed or derived statistical models. The classes of transform-coding algorithms are based on this technology [3.10]. The utility of Fourier transform and its discrete extensions such as Discrete Cosine Transform (DCT), wavelets and others are based on the principle that these transforms asymptotically approach the optimum transform, assuming a reasonable statistical behavior [3.11]. Visual and audio models are fundamental to the relevant multimedia representations, primarily compression procedures. These models are based on fundamental statistical representations of the elementary components, including their evaluation by the human observer [3.12, 3.13].

The models are:

- Implementation of motion detection and associated compensation in subsequent image frames can significantly reduce the required bandwidth. Successful prediction of image segment locations in future frames reduces the required information update to the required motion vectors. Thus, under this condition, the associated update information is dramatically reduced.

- Combining the presence of motion in video segments with the limitations for human visual systems provides additional bandwidth-reduction potentials. Because the human vision deteriorates when observing moving areas, image blur associated with these regions becomes significantly less noticeable. Consequently, additional image compression can be introduced in segments that contain motion, with minimal noticeable effect.

Human vision is basically 3D. Efficient representation of a 3D signal is a major challenge of multimedia. The most common 3D techniques are based on 2D display techniques. The 3D scene is projected onto two dimensions in the rendering phase of the multimedia chain. The proper hierarchy of object elements and behavior maintains the 3D illusion. The relevant processes include shadowing consideration and preserving the proper hidden body behavior. The required processing resources are still significant. A substantial industry produces various processing components, such as chip sets and graphics boards, to develop solutions for many diverse applications including desktop computing. The associated technology is very effective in high-end applications. Virtual reality models using large screens are impressive even though the presentation remains 2D. In 3D representations, the stereo projection is the best known. The same 3D scene is recorded from two slightly different perspectives, essentially replicating our eyes. The two separate recordings are subsequently presented to the eyes separately. Unlike the early stereo film-based recordings, modern techniques are heavily dependent on digital processing, which corrects for camera-projection inaccuracies, resulting in significantly enhanced stereo display.

Projection techniques comprise an effective group to recreate multidimensionality from individual projections through the original object. Although this technology has been used very effectively in medical applications, its utility to multimedia applications is not likely to be useful in the near future. The primary limitations are complexity and lack of easy real-time implementation [3.14].

For efficient representation of color processing, modeling and communication applications, color plays a very important role. The correlation properties among color planes are used in image and video compression algorithms.

3.4 Challenges of Multimedia Information Processing

Novel communications and networking technologies are critical for a multimedia database system to support interactive dynamic interfaces. A truly integrated media system must connect with individual users and content-addressable multimedia databases. This will be a logical connection through computer networks and data transfer.

To advance the technologies of indexing and retrieval of visual information in large archives, multimedia content-based indexing would complement the text-based search. Multimedia systems must successfully combine digital video and audio, text animation, graphics and knowledge about such information units and their interrelationships in real time.

The operations of filtering, sampling, spectrum analysis and signal representation are basic to all of signal processing. Understanding these operations in the multidimensional (mD) case has been a major activity since 1975 [3.15, 3.16, 3.17]. More key results since that time have been directed at the specific applications of image and video processing, medical imaging, and array processing. Unfortunately, there remains considerable cross-fertilization among the application areas.

Algorithms for processing mD signals can be grouped into four categories:

- Separable algorithms that use 1D operators to process the rows and columns of a multidimensional array
- Nonseparable algorithms that borrow their derivation from their 1D counterparts
- mD algorithms that are significantly different from their 1D counterparts
- mD algorithms that have no 1D counterparts.

Separable algorithms operate on the rows and columns of an mD signal sequentially. They have been widely used for image processing because they invariably require less computation than nonseparable algorithms. Examples of separable procedures include mD Discrete Fourier Transforms (DFTs), DCTs and Fast Fourier Transform (FFT)-based spectral estimation using the periodogram. In addition, separable Finite Impulse Response (FIR) filters can be used in separable filter banks, wavelet representations for mD signals and decimators and interpolators for changing the sampling rate.

The second category contains algorithms that are uniquely mD in that they cannot be decomposed into a repetition of 1D procedures. These can usually be derived by repeating the corresponding 1D derivation in an mD setting. Upsampling and downsampling are some examples. As in the 1D case, bandlimited multidimensional signals can be sampled on periodic lattices with no loss of information. Most 1D FIR filtering and FFT-based spectrum analysis algorithms also generalize straightforwardly to any mD lattice [3.18]. Convolutions can be implemented efficiently using the mD DFT either on whole arrays or on subarrays. The window method for FIR filter design can be easily extended, and the FFT algorithm can be decomposed into a vector-radix form, which is slightly more efficient than the separable row/column approach for evaluating multidimensional DFTs [3.19, 3.20]. Nonseparable decimators and interpolators have also been derived that may eventually be used in subband image and video coders [3.21]. Another major area of research has been spectral estimation. Most of the modern spectral estimators, such as the maximum entropy method, require a new formulation based on constrained optimization. This is because their 1D counterparts depend on factorization properties of polynomials [3.22]. An interesting case is the maximum likelihood method, where the 2D version was developed first and then adopted to the 1D situation [3.23].

There are also mD algorithms that have no 1D counterparts, especially algorithms that perform inversion and computer imaging. One of these is the operation of recovering an mD distribution from a finite set of its projections, equivalently inverting a discretized Radon transform. This is the mathematical basis of computed tomography and positron emission tomography.

Another imaging method, developed first for geophysical applications, is Fourier integration. Finally, signal recovery methods unlike the 1D case are possible. The mD signals with finite support can be recovered from the amplitudes of their Fourier transforms or from threshold crossings [3.24].

3.4.1 Pre and Postprocessing

In multimedia applications, the equipment used for capturing data, such as the camera, should be cheap, making it affordable for a large number of users. The quality of such equipment drops when compared to their more expensive and professional counterparts. It is mandatory to use a preprocessing step prior to coding in order to enhance the quality of the final pictures and to remove the noise that will affect the performance of compression algorithms. Solutions have been proposed in the field of image processing to enhance the quality of images for various applications [3.25, 3.26]. A more appropriate approach would be to take into account the characteristics of the coding scheme when designing such operators. In addition, pre- and postprocessing operators are extensively used in order to render the input or output images in a more appropriate format for the purpose of coding or display.

Mobile communications is an important class of applications in multimedia. Terminals in such applications are usually subject to different motions, such as tilting and jitter, translating into a global motion in the scene due to the motion of the camera. This component of the motion can be extracted by appropriate methods detecting the global motion in the scene and can be seen as a preprocessing stage. Results reported in the literature show an important improvement of the coding performance when a global motion estimation is used [3.27].

It is normal to expect a certain degree of distortion of the decoded images for very low-bit-rate applications. However, an appropriate coding scheme introduces the distortions in areas that are less annoying to the users. An additional stage could be added to reduce the distortion further due to compression as a postprocessing operator. Solutions were proposed in order to reduce the blocking artifacts appearing at high compression ratios [3.28, 3.29, 3.30, 3.31, 3.32, 3.33]. The same types of approaches have been used in order to improve the quality of decoded signals in other coding schemes, reducing different kinds of artifacts, such as ringing, blurring and mosquito noise [3.34, 3.35].

Recently, advances in postprocessing mechanisms have been studied to improve lip synchronization of head-and-shoulder video coding at a very low bit rate by using the knowledge of decoded audio in order to correct the positions of the lips of the speaker [3.36]. Figure 3.2 shows an example of the block diagram of such a postprocessing operation.

3.4.2 Speech, Audio and Acoustic Processing for Multimedia

The primary advances in speech and audio signal processing that contributed to multimedia applications are in the areas of speech and audio signal compression, speech synthesis, acoustic processing, echo control and network echo cancellation.

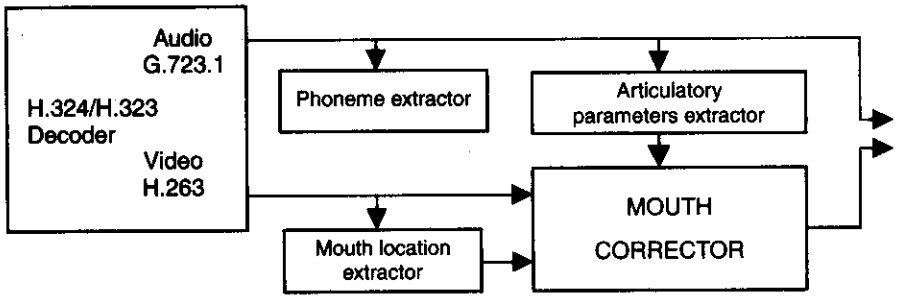

Figure 3.2 Block diagram for audio-assisted head and shoulder video [3.36]. ©1998 IEEE.

Speech and audio signal compression Signal compression techniques aim at efficient digital representation and reconstruction of speech and audio signals for storage and playback as well as transmission in telephony and networking.

Signal-analysis techniques such as Linear Predictive Coding (LPC) [3.37], and all-pole autoregressive modeling [3.38] and Fourier analysis [3.39], played a central role in signal representation. For compression, VQ [3.40, 3.41] marks a major advance. These techniques are built upon rigorous mathematical frameworks that have become part of the important bases of digital signal processing. Incorporation of knowledge and models of psychophysics in hearing have been proven as beneficial for speech and audio processing. Techniques such as noise shaping [3.42] and explicit use of auditory masking in the perceptual audio coder [3.43] have been found very useful. Today, excellent speech quality can be obtained at less than 8 Kb/s, which forms the basis for cellular as well as Internet telephony. The fundamental structure of the Code- Excited Linear Prediction (CELP) coder is ubiquitous in supporting speech coding at 4 to 16 Kb/s, encompassing such standards as G.728 [3.44], G.729 [3.45], G.723.1, IS-54 [3.46], IS-136 [3.47], GSM [3.48] and FS-1016 [3.49]. CD or near-CD-quality stereo audio can be achieved at 64 to 128 Kb/s, less than one twelfth of the original CD rate, and is ready for such applications as Internet audio (streaming and multicasting) and digital radio (digital audio broadcast). Advances in audio-coding standards are supported in MPEG activities.

Speech synthesis The area of speech synthesis includes generation of speech from unlimited text, voice conversion and modification of speech attributes such as time scaling and articulatory mimic [3.50]. Text-to-speech conversion takes text as input and generates human-like speech as output [3.51]. Key problems in this area include conversion of text into a sequence of speech inputs (in terms of phonemes, dyades or syllables), generation of the associated prosodic structure and intonation and methods to concatenate and reconstruct the sound waveform. Voice conversion refers to the technique of changing one person's voice to another, from person A to person B or from male to female and vice versa. It is useful to be able to change the time scale of a signal (to speed up or slow down the speech signal which changes the pitch) or to change the mode of the speech (making it sound happy or sad) [3.52]. Many of these signal-processing techniques have appeared in animation and computer graphics applications.

Acoustic processing and echo control Sound pickup and playback is an important area of multimedia processing. In sound recording, interference, such as ambient noise and reverberation, degrade the quality. The idea of acoustic signal processing and echo control is to allow straightforward high-quality sound pickup and playback in applications, such as a duplex device like a speakerphone, a sound source-tracking apparatus like microphone arrays, teleconferencing systems with stereo input and output, hands-free cellular phones and home theatre with 3D sound.

Signal processing for acoustic echo control includes modeling of reverberation, design of dereverberation algorithms, echo suppression, double-talk detection and adaptive acoustic echo cancellation, which is still a challenging problem in stereo full-duplex communication environments [3.53].

Example 3.3 For typical environments, the system modeling time for reverberation is of the order of 100 ms. This at a sampling rate of 16 KHz translates into a echo-canceling filter of 1600 taps, requiring seconds to converge.

For sound pickup, acoustic processing aims at the design of transducers or transducer arrays to achieve a durable directionality (beam steering and width control) as well as noise resistance. Understanding of near and far-field acoustics is important in achieving the required response in specific applications [3.54]. Various 1D and 2D microphone arrays have been shown in teleconferencing and auditorium applications with good results [3.55].

Network echo cancellation In telephony, both near-end and far-end echo exists due to the hybrid coil that is necessary for two-wire and four-wire conversions. Network echo can be so severe that it hampers telephone conversation. Network echo cancellers were invented to correct the problem in the late 1960s, based on the Least Mean Squares (LMS) adaptive echo cancelation algorithm [3.56]. The network echo delay is of the order of 16 ms, typically requiring a filter with 128 taps at a sampling rate of 8 KHz.

3.4.3 Video Signal Processing

Digital video has many advantages over conventional analog video, including bandwidth compression, robustness against channel noise interactivity and ease of manipulation. Digital-video signals come in many formats. Broadband TV signals are digitized with ITU-R 601 format, which has 30/25 fps, 720 pixels by 488 lines per frame, 2:1 interlaced, 4:3 aspect ratio, and 4:2:2 chroma sample. With the advent of high-definition digital-video, standardization efforts between the TV and PC industries have resulted in the approval of 18 different digital video formats in the United States. Exchange of video signals between TV and PCs requires effective format conversion. Some commonly used interframe/field filters for format conversion, for example, ITU-R 601 to the Source Input Format (SIF) and vice versa and 3:2 pull-down to display 24 Hz motion pictures in 60 Hz format, have been reviewed [3.57]. As for video filters, they can be classified as interframe/field (spatial), motion-adaptive and motion-compensated filters [3.58]. Spatial filters are easiest to implement. However, they do not make use of the high temporal correlation in the video signals. Motion-compensated filters require highly accurate motion estimation between successive views. Other more sophisticated format conversion methods include

motion-adaptive field-rate doubling and deinterlacing [3.59] as well as motion compensated frame rate conversion [3.58].

Video signals suffer from several degradations and artifacts. Some of these degradations may be acceptable under certain viewing conditions. However, they become objectionable for freezeframe or printing from video applications. Some filters are adaptive to scene content in that they aim to preserve spatial and temporal edges while removing the noise. Examples of edge-preserving filters include median, weighted median, adaptive linear mean square error and adaptive weighted-averaging filtering [3.58]. Deblocking filters can be classified as those that do require a model of the degradation process (inverse, constrained, least square, and Wiener filtering) and those that do not (contrast adjustment by histogram specification and unsharp masking). Deblocking filters smooth intensity variations across amounts of temporal redundancy. Namely, successive frames generally have large overlaps with each other. Assuming that frames are shifted by subpixel amounts with respect to each other, it is possible to exploit this redundancy to obtain a high-resolution reference image (mosaic) of the regions covered in multiple views [3.60]. High-resolution reconstruction methods employ least-squares estimation, back projection, or projection-autoconvex sets methods based on a simple instantaneous camera model or a more sophisticated camera model including motion blur [3.61].

One of the challenges in digital video processing is to decompose a video sequence into its elementary parts (shots and objects). A video sequence is a collection of shots, a shot is a group of frames and each frame is composed of synthetic or natural visual objects. Thus, temporal segmentation generally refers to finding shot boundaries, spatial segmentation corresponds to extraction of visual objects in each frame and object tracking means establishing correspondences between the boundaries of objects in successive frames.

Temporal segmentation methods edit effects as cuts, dissolves, fades and wipes. Thresholding and clustering using histogram-based similarity methods have been found effective for detection of cuts [3.62] . Detection of special effects with high accuracy requires customized methods in most cases and is a current research topic. Segmentation of objects by means of chroma keying is relatively easy and is commonly employed. However, automatic methods based on color, texture and motion similarity often fail to capture semantically meaningful objects [3.63]. Semiautomatic methods, which aim to help a human operator perform interactive segmentation by tracking boundaries of a manual initial segmentation, are usually required for object-based video editing applications. Object-tracking algorithms, which can be classified as boundary region or model-based tracking methods, can be based on 2D or 3D object representations. Effective motion analysis is an essential part of digital video processing and remains an active research topic.

Storage and archiving of digital video in shared disks and servers in large volumes, browsing of such databases in real time and retrieval across switched and packet networks pose many new challenges, one of which is efficient and effective description of content. The simplest method to index content is by assigning manually or semiautomatically the content to programs, shots and visual objects [3.64]. It is of interest to browse and search for content using compressed data because almost all video data will likely be stored in compressed format [3.65].

Video-indexing systems may employ a frame-based, scene-based or object-based video representation. The basic components of a video-indexing system are temporal segmentation, analysis of indexing features and visual summarization. The temporal-segmentation step extracts shots, scenes and/or video objects. The analysis step computes content-based indexing features for the extracted shots, scenes, or objects. Content-based features may be generic or domain dependent. Commonly used generic indexing features include color histograms, type of camera-motion direction and magnitude of dominant object motion entry and exit instances of objects of interest and shape features for objects [3.66, 3.67]. Domain-dependent feature extraction requires a priori knowledge about the video source, such as new programs, particular sitcoms, sportscasts and particular movies. Content-based browsing can be facilitated by a visual summary of the contents of a program, much like a visual table of contents. Among the proposed visual summarization methods are story boards, visual posters and mosaic-based summaries.

3.4.4 Content-Based Image Retrieval

To address their challenges, multimedia signal-processing methods must allow efficient access to processing and retrieval of content in general, and visual content in particular. This is required across a large range of applications, in medicine, entertainment, consumer industry, broadcasting, journalism, art and e-commerce. Therefore, methods originating from numerous research areas, that is, signal processing, pattern recognition, computer vision, database organization, human-computer interaction and psychology, must contribute to achieving the image-retrieval goal. An example of image retrieval is

```
Given:      A query
Retrieve:   All images that have similar content to that of the query.
```

Image-retrieval methods face several challenges when addressing this goal [3.68]. These challenges, which are summarized in Table 3.1, cannot be addressed by text-based image retrieval systems, which have had an unsatisfactory performance so far. In these systems, the query keywords are matched with keywords that have been associated to each image. Because of difficult automatic selection of the relevant keywords, time consuming and subjective manual annotation is required. Moreover, the vocabulary is limited and must be expanded as new applications emerge.

To improve performance and address these problems, content-based image retrieval methods have been proposed. These methods have generally focused on using low-level features such as color, texture and shape layout, for image retrieval, mainly because such features can be extracted automatically or semiautomatically.

Texture-Based Methods

Statistical and syntactic texture description methods have been proposed. Methods based on spatial frequencies, co-occurrence matrixes and multiresolution methods have been frequently employed for texture description because of their efficiency [3.69]. Methods based on spatial frequencies evaluate the coefficients of the autocorrelation function of the texture. Co-occurrence matrixes identify repeated occurrences of gray level pixel configurations within the tex-

Table 3.1 Image retrieval challenges [3.68].

Challenges	Remarks
Query types	Color based/shape based/color and shape based
Query forms	Quantitative, for example, find all images with 30% amount of red
	Query by example, for example, image region/image/sketch/other examples
Various content	For example, natural scenes/head-and-shoulder images/MRIs
Matching types	Object to object/image to image/object to image
Precision levels	Application specific
	Exact versus similarity-based match
Presentation of results	Application specific

ture. Multiresolution methods describe the texture characteristics at coarse-to-fine resolutions. A major problem that is associated with most texture description methods is their sensitivity to scale, that is, the texture characteristics may disappear at low resolutions or may contain a significant amount of noise at high resolutions [3.70, 3.71, 3.72].

Shape-Based Methods

Describing quantitatively the shape of an object is a difficult task. Several contour-based and region-based shape description methods have been proposed. Chain codes, geometric border representations, Fourier transforms of the boundaries, polygonal representations and deformable (active) models are some of the boundary-based shape methods that have been employed for shape description. Simple scalar region descriptors, moments, region decompositions and region neighborhood graphs are region-based methods that have been proposed for the same task [3.73, 3.74]. Contour-based and region-based methods are developed in either the spatial or transform domains, yielding different properties of the resulting shape descriptors. The main problems that are associated with shape description methods are high sensitivity to scale, difficult shape description of objects and high subjectivity of the retrieved shape results.

Color-Based Methods

Color description methods are generally color histogram based, dominant color based and color moment based [3.75, 3.76]. Description methods that employ color histograms use a quantitative representation of the distribution of color intensities. Description methods that employ dominant colors use a small number of color ranges to construct an approximate representation of color distribution. Description methods that use color moments employ statistical measures of the image characteristics in terms of color.

The performance of these methods typically depends on the color space, quantization, and distance measures employed for evaluation of the retrieved results. The main problem that is associated with histogram-based and dominant-color-based methods is their inability to allow the localization of an object with the image. A solution to address this problem is to apply color segmentation, which allows both image-to-image matching and object localization. The main problem of color-moment-based methods is their complexity, which makes their application to browsing or other image-retrieval functionalities difficult.

Examples of content-based image and video-retrieval systems are included in Table 3.2. Some or all of the limitations of these systems are the following [3.68]:

- Few query types are supported
- Limited set of low-level features
- Difficult access to visual objects
- Results partially match user's expectations
- Limited interactivity with the user
- Limited system interoperability
- Scalability problems

Table 3.2 Examples of content-based image and video-retrieval systems [3.68].

Features	System	Image/Video	Provider
Color and text	WebSeek	I, V	Columbia University
	Picasso	I	University of Florence
	Chabot	I	University of California, Berkeley
	*	I	University of Toronto
Color, texture and shape	QBIC	I	IBM
	PhotoBook	I	MIT
	BlobWorld	I	University of California, Berkeley
	VIR	I, V	Virage
Color, shape and scale	Nefertiti	I	National Research Council of Canada

* No name has been adopted for the corresponding system.

Table 3.2 Examples of content-based image and video-retrieval systems [3.68]. (Continued)

Features	System	Image/Video	Provider
Color, texture, shape and spatial location	NeTra	I	University of California, Santa Barbara
	Digital storyboard	I	Kodak
Color, texture and motion	WebClip	V	Columbia University
	Jacob	I, V	University of Palermo
	*	V	IMAX
N/A	*	V	NASA

* No name has been adopted for the corresponding system.

3.5 Perceptual Coding of Digital Audio Signals

Audio coding is used to obtain compact digital representations of high-fidelity (wideband) audio signals for the purpose of efficient transmission or storage. The central objective in audio coding is to represent the signal with a minimum number of bits while achieving transparent signal reproduction, for example, while generating output audio that cannot be distinguished from the original input, even by a sensitive listener [3.77].

The introduction of the CD brought to the fore all of the advantages of digital audio representation, including high fidelity, dynamic range and robustness. However, these advantages came at the expense of high data rates. Conventional CD and Digital Audio Tape (DAT) systems are typically sampled at 44.1 or 48 KHz, using PCM with a 16-bit sample resolution [3.78]. This results in uncompressed data rates of 705.6/768 Kb/s for monaural channel, or 1.41/1.54 Mb/s for a stereo pair at 44.1/48 KHz, respectively. Although high, these data rates were accommodated successfully in first-generation digital-audio applications, such as CD and DAT [3.79]. Unfortunately, second-generation multimedia applications and wireless systems in particular are often subject to bandwidth or cost constraints that are incompatible with high data rates. Because of the success enjoyed by the first generation, end users have come to expect CD quality audio reproduction from any digital system [3.80]. Therefore, new network and wireless multimedia digital audio systems must reduce data rates without compromising reproduction quality. These and other considerations have motivated considerable research in the area of compression schemes that can satisfy simultaneously the conflicting demands of high compression ratios and transparent reproduction quality for high-fidelity audio signals [3.81].

3.5.1 General Perceptual Audio-Coding Architecture

Although the enormous capacity of new storage media, such as digital versatile disk (DVD), can accommodate lossless audio coding, the research interests are lossy compression schemes

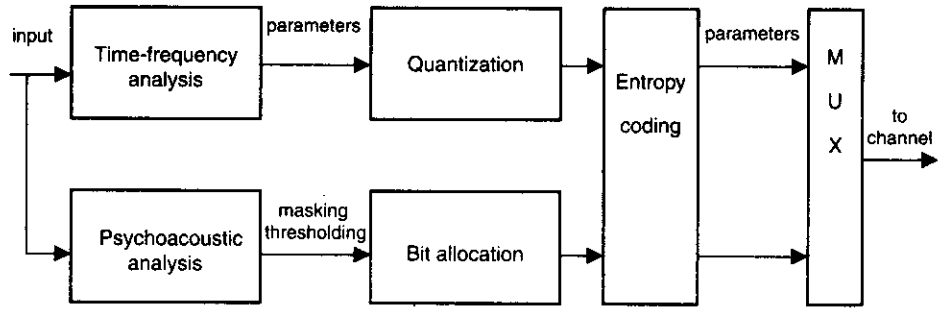

Figure 3.3 Perceptual audio-coder architecture [3.82]. ©1997 IEEE.

[3.82], which seek to exploit the psychoacoustic principles. Lossy schemes offer the advantage of lower bit rates, for example, less than 1 bit per sample relative to lossless schemes (for example, 1 bit per sample).

The lossy compression systems achieve coding gain by exploiting both perceptual irrelevancies and statistical redundancies. All of these algorithms are based on the architecture shown in Figure 3.3.

The coders typically segment input signals into quasistationary frames ranging from 2 to 50 ms in duration. A time-frequency analysis section then decomposes each analysis frame. The time-frequency analysis approximates the temporal and spectral analysis properties of the human auditory system. It transforms input audio into a set of parameters, which can be quantized and encoded according to a perceptual distortion metric. Depending on the overall system objectives, the time-frequency analysis section may contain the following:

- Unitary transform
- Time-invariant bank of uniform bandpass filters
- Time-varying, critically sampled bank of nonuniform bandpass filters
- Hybrid transform/filterbank signal analyzer
- Harmonic/sinusoidal analyzer
- Source-system analysis (LPC/multipulse excitation).

The choice of time-frequency analysis methodology always involves a fundamental tradeoff between time and frequency resolution requirements. Perceptual distortion control is achieved by a psychoacoustic signal analysis section that estimates signal masking power based on psychoacoustic principles. The psychoacoustic model delivers masking thresholds that quantify the maximum amount of distortion that can be injected at each point of the time-frequency plane during quantization and encoding of the time-frequency parameters, without introducing audible artifacts in the reconstructed signal. Therefore, the psychoacoustic model allows the quantization and encoding section to exploit perceptual irrelevancies in the time-frequency parameter set. The quantization and encoding section can also exploit statistical redundancies through classical techniques, such as Differential Pulse Code Modulation (DPCM) or Adaptive

DPCM (ADPCM). Quantization might be uniform or pdf optimized (Lloyd-Max quantizer). It might be performed on either scalar or vector quantities. After a quantized compact parametric has been formed, remaining redundancies are removed through Run-Length (RL) and entropy-coding techniques, for example Huffman, arithmetic, and Lempel-Ziv-Welch (LZW). Because the psychoacoustic distortion control model is signal adaptive, most algorithms are inherently variable rate. Fixed-channel rate requirements are usually satisfied through buffer feedback schemes, which often introduce encoding delays. The study of Perceptual Entropy (PE) suggests that transparent coding is possible in the neighborhood of 2 bits per sample for most high- fidelity audio sources [3.83]. Regardless of design details, all perceptual audio coders seek to achieve transparent quality at low rates with tractable complexity and manageable delay.

3.5.2 Review of Psychoacoustic Fundamentals

Audio-coding algorithms must rely upon generalized receiver models to optimize coding efficiency. In the case of audio, the receiver is ultimately the human ear, and sound perception is affected by its masking properties. The field of psychoacoustics has made significant progress toward characterizing human auditory perception and the time-frequency analysis capabilities of the inner ear [3.84, 3.85, 3.86]. Most current audio coders achieve compression by exploiting the fact that irrelevant signal information is not detectable by even a well-trained or sensitive listener. Irrelevant information is identified during signal analysis by incorporating into the coder several psychoacoustics principles, including:

- Absolute threshold of hearing
- Critical band frequency analysis
- Simultaneous masking and the spread of masking
- Temporal masking

Combining these psychoacoustic notions with basic properties of signal quantization has also led to the development of PE, a quantitative estimate of the fundamental limit of transparent audio-signal compression [3.87].

Absolute Threshold of Hearing

The absolute threshold of hearing is characterized by the amount of energy needed in a pure tone such that it can be detected by a listener in a noiseless environment. The frequency dependence of this threshold was quantified when test results for a range of listeners were reported [3.88]. The quiet threshold is well approximated by the nonlinear function

$$T_q(f) = 3.64(f/1000)^{-0.8} - 6.5e^{-0.6(f/1000-3.3)^2} + 10^{-3}(f/1000)^4 \quad \text{[dB SPL]} \tag{3.1}$$

where SPL is the sound pressure level.

This is representative of a young listener with acute hearing. When applied to signal compression, $T_q(f)$ can be interpreted as a maximum allowable energy level for coding distortions introduced in the frequency domain. The absolute threshold of hearing is shown in Figure 3.4. Algorithm designers have no a priori knowledge regarding actual playback levels. Therefore, the

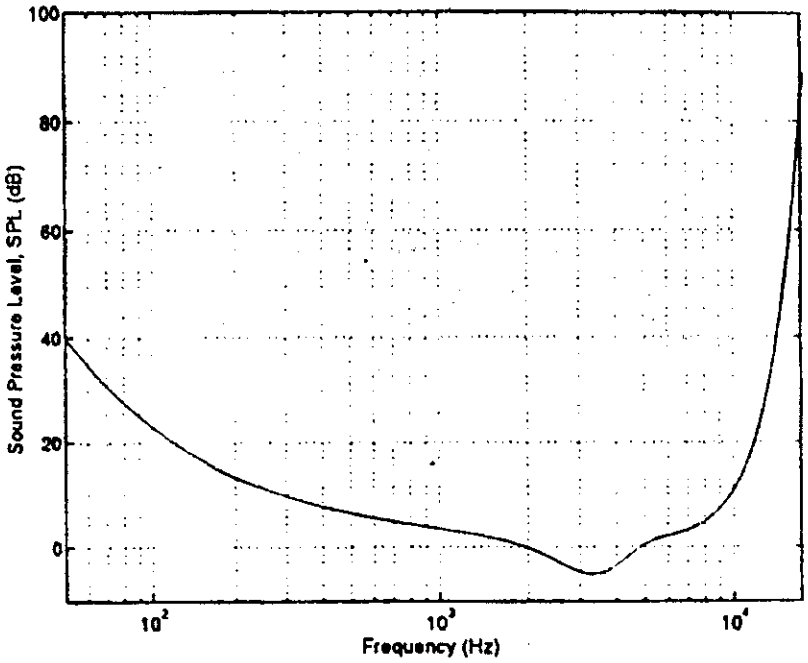

Figure 3.4 Absolute threshold of hearing [3.85].

Sound Pressure Level (SPL) curve is often referenced to the coding systems by equating the lowest point on the curve (that is, 4 KHz) to the energy in +/-1 bit of signal amplitude. Such a practice is common in algorithms that use the absolute threshold of hearing.

Critical Band Frequency Analysis

Using the absolute threshold of hearing to shape the coding-distortion spectrum represents the first step toward perceptual coding. Consider how the ear actually does spectral analysis. It is evident that a frequency-to-place transformation takes place in the inner ear along the basilar membrane. Distinct regions in the cochlea, each with a set of neural receptors, are tuned to different frequency bands. In the experimental sense, critical bandwidth can be loosely defined as the bandwidth at which subjective responses change abruptly. For example, the perceived loudness of a narrowband noise source at a constant sound-pressure level remains constant even as the bandwidth is increased up to the critical bandwidth. The loudness then begins to increase.

Example 3.4 Critical band-measurement methods are presented in Figure 3.5. In a different experiment (Figure 3.5a), the detection threshold for a narrow-band noise source between two masking tones remains constant as long as the frequency separation between the tones remains within a critical bandwidth. Beyond the bandwidth, the threshold rapidly decreases (Figure 3.5c). A similar notched-noise experiment can be constructed with measure and mask roles reversed (Figure 3.5b, and d).

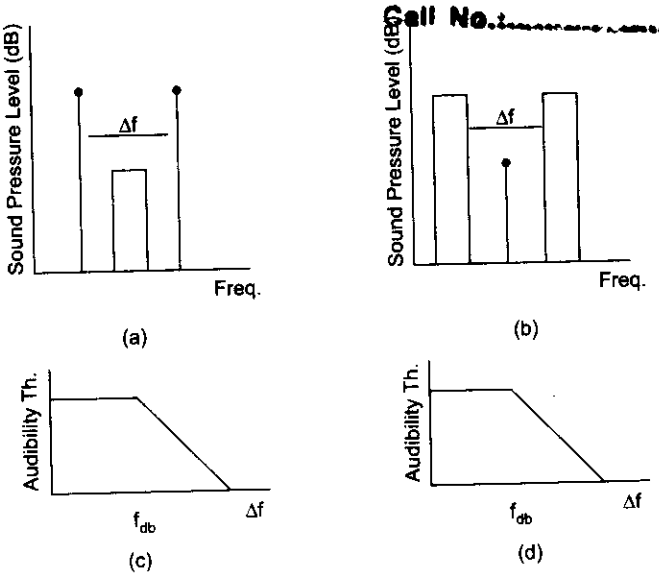

Figure 3.5 Control band-measurement methods [3.82].

Simultaneous Masking and the Spread of Masking

Masking refers to a process where one sound is rendered inaudible because of the presence of another sound. Simultaneous masking refers to a frequency-domain phenomenon that has been observed within critical bands (in-band). For the purposes of shaping coding distortions, it is convenient to distinguish between two types of simultaneous masking: tone-masking noise and noise-masking tone [3.85]. In the first case, a tone occurring at the center of a critical band masks noise of any subcritical bandwidth or shape, provided the noise spectrum is below a predictable threshold directly related to the strength of the masking tone. The second masking type follows the same pattern with the roles of masker and mask reversed. A simplified explanation of the mechanism for both masking phenomena is as follows. The presence of a strong noise or tone masker creates an excitation of sufficient strength on the basilar membrane at the critical band location to block transmission of a weaker signal effectively. Interband masking has also been observed. It means that a masker centered with one critical band has some predictable effect on detection thresholds in other critical bands. This effect, also known as the spread of masking, is often modeled in coding applications by an approximately triangular spreading function that has slopes of +25 and -10 dB per bark. A convenient analytical expression is given by

$$SF_{dB}(x) = 15.81 + 7.5(x + 0.474) - 17.5\sqrt{1 + (x + 0.474)^2} \quad [dB] \tag{3.2}$$

where x has units of barks, and basilar spreading function $SF_{dB}(x)$ is expressed in dB [3.82]. After critical band analysis is done and the spread of masking has been accounted for, masking thresholds in psychoacoustic coders are often established by the decibel [dB] relations

$$TH_N = E_T - 14.5 - B \tag{3.3}$$

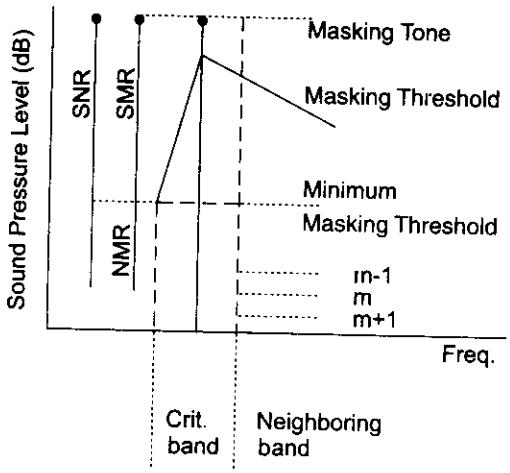

Figure 3.6 Schematic representation of simultaneous masking [3.82]. ©1997 IEEE.

$$TH_T = E_N - K \tag{3.4}$$

where TH_N and TH_T . are the noise and tone-masking thresholds, respectively, due to tone-masking noise and noise-masking tone. E_N and E_T are the critical band noise and tone- masker energy levels, while B is the critical band number [3.89]. Depending upon the algorithms, the parameter K has typically been set between 3 and 5 dB. Masking thresholds are commonly referred to in the literature as functions of Just Noticeable Distortion (JND). One psychoacoustic coding scenario might involve first classifying masking signals as either noise or tone, next computing appropriate thresholds and then using this information to shape the noise spectrum beneath JND. The absolute threshold (T_{abs}) of hearing is also considered when shaping the noise spectra. Also, MAX(JND, T_{abs}) is most often used as the permissible distortion threshold.

Schematic representation of simultaneous masking is shown in Figure 3.6. Consider the case of a single masking tone occurring at the center of a critical band. All levels are given in terms of dB SPL. A hypothetical masking tone occurs at some masking level. This generates an excitation along the basilar membrane, which is modeled by a spreading function and a corresponding masking threshold. For the band under consideration, the minimum masking threshold denotes the spreading function in-band minimum. Under the assumption, the masker is quantized using an m-bit uniform scalar quantizer, and noise might be introduced at the level m. Signal-to-Noise Ratio (SNR) and Noise-to-Mask Ratio (NMR) denote the log distances from the minimum masking threshold to the masker and noise levels, respectively.

Temporal Masking

In the context of audio-signal analysis, abrupt signal transients create pre-and postmasking regions in a time during which a listener will not perceive signals beneath the elevated audibility

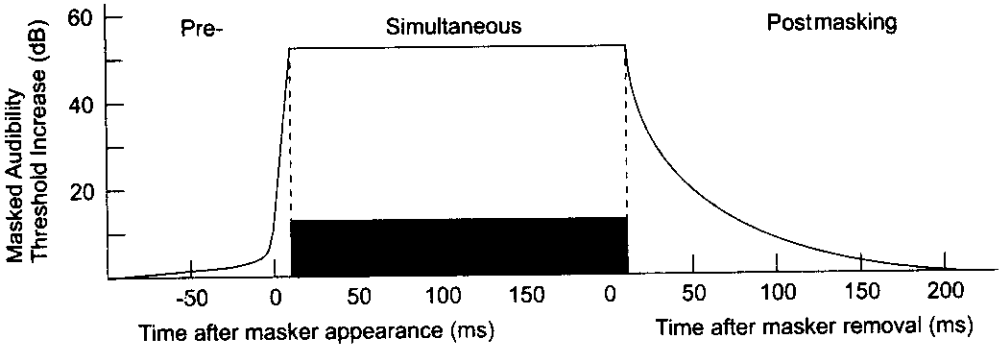

Figure 3.7 Schematic representation of temporal masking properties of the human ear [3.82]. ©1997 IEEE.

thresholds produced by a masker. Schematic representation of temporal masking properties of the human ear is shown in Figure 3.7. Absolute audibility thresholds for masked sounds are artificially increased prior to, during and following the occurrence of a masking signal. Premasking tends to last only about 5 ms. Postmasking will extend anywhere from 50 to 300 ms, depending upon the strength and duration of the masker [3.86]. Temporal masking has been used in several audio-coding algorithms [3.82]. In particular, premasking has been exploited in conjunction with adaptive block-size transform coding to compensate for pre-echo distortions.

PE

This is a measure of perceptually relevant information contained in any audio record. Expressed in bits per sample, PE represents a theoretical limit on the compressibility of a particular signal. PE measurements are reported in Johnston [3.87], who suggests that a wide variety of CD quality audio material can be transparently compressed at approximately 2.1 bits per sample.

The PE estimation process is accomplished as follows. The signal is first windowed and transformed to the frequency domain. A masking threshold is then obtained using perceptual rules. Finally, a determination is made on the number of bits required to quantize the spectrum without injecting perceptible noise. The PE measurements are obtained by constructing a PE histogram over many frames and then choosing a worst-case value as the actual measurement.

Masking thresholds are obtained by performing critical band analysis, masking a determination of the noise or toneline nature of the signal, applying thresholding rules for the signal quality and then accounting for the absolute hearing threshold. First, real and imaginary components are converted to power spectral components.

$$P(w) = [\mathrm{Re}(\omega)]^2 + [\mathrm{Im}(\omega)]^2 \tag{3.5}$$

Then, a discrete bark spectrum is formed by summing the energy in each critical band

$$B_i = \sum_{\omega=bl}^{bh} P(\omega) \tag{3.6}$$

where the summation limits are the critical band boundaries (*bl*-bandlow, *bh*-bandhigh). The range of the index, i, is sampling rate dependent, and in particular for $i \in \{1, 25\}$ CD-quality signals. A basilar spreading function (3.2) is then convolved with the discrete bark spectrum

$$C_i = B_i * SF_i \tag{3.7}$$

to account for interband masking. An estimation of the toneline or noiselike quality for C_i is then obtained using the spectral flatness measure (SFM)

$$SFM = \frac{M_g}{M_a} \tag{3.8}$$

where M_g and M_a correspond to geometric and arithmetic means of the power-spectral-density components for each band. The SFM has the property that it is bounded by 0 and 1. Values close to 1 will occur if the spectrum is flat in a particular band, indicating a decorrelated (noisy) band. Values close to zero will occur if the spectrum in a particular band is nearly sinusoidal. A coefficient of tonality, α, is next derived from the SFM on a dB scale, that is

$$\alpha = \min\left(\frac{SFM_{dB}}{-60}, 1\right) \tag{3.9}$$

This is used to weigh the thresholding rules (3.3) and (3.4), with $K = 5.5$ as follows for each band to form an offset.

$$O_i = \alpha(14.5 + i) + (1 - \alpha)5.5 \quad [dB] \tag{3.10}$$

A set of JND estimates in the frequency power domain are then formed by subtracting the offsets from the bark spectral components.

$$T_i = 10^{\log_{10} C_i - O_i/10} \tag{3.11}$$

These estimates are scaled by a correction factor to simulate deconvolution of the spreading function. Then each T_i is checked against the absolute threshold of hearing and replaced by $\max(T_i, T_{ABS}(i))$. As previously noted, the absolute threshold is referenced to the energy in a 4 KHz sinusoid of +/-1 bit amplitude. By applying uniform quantization principles to the signal and associated set of JND estimates, it is possible to estimate a lower bound on the number of bits required to achieve transparent coding. The perceptual entropy in bits per sample is represented by [3.82]

$$PE = \sum_{i=1}^{25} \sum_{\omega=bl_i}^{bh_i} \log_2\left\{2\left|\text{int}\left(\frac{\text{Re}(\omega)}{\sqrt{6T_i / k_i}}\right)\right| + 1\right\} + \log_2\left\{2\left|\text{int}\left(\frac{\text{Im}(\omega)}{\sqrt{6T_i / k_i}}\right)\right| + 1\right\} \tag{3.12}$$

where i is the index of critical band, bl_i and bh_i are the upper and lower bounds of band i, k_i is the number of transform components in band i, T_i is the masking threshold in band i (3.11), while *int* denotes rounding to the nearest integer. If 0 occurs in the log, we assign 0 for the result.

3.6 Transform Audio Coders

Transform coding for high-fidelity audio makes use of unitary transforms for the time/frequency analysis. These algorithms typically achieve high-resolution spectral estimates at the expense of

adequate temporal resolution. Many transform-coding algorithms for wideband and high-fidelity audio have been proposed in the last 15 years. The individual algorithms have been proposed by several groups [3.77, 3.78, 3.90, 3.91, 3.92, 3.93]. Much of this work was motivated by standardization activities, and the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) eventually clustered these proposals into a single candidate algorithm, Adaptive Spectral Entropy Coding (ASPEC) of high-quality music signals, which competed successfully for inclusion in the ISO/IEC MPEG audio-coding standards [3.94].

The algorithms that were eventually clustered into the ASPEC proposal submitted to ISO/IEC for MPEG audio came from researchers in both the United States and Europe. Novel transform-coding algorithms are not associated with ASPEC. Here, we will report some proposals for transform audio coding, like the following:

- Optimum coding in the frequency domain
- Perceptual transform coder
- Hybrid coder
- Transform coding that exploits DFT interblock redundancy
- ASPEC
- Differential perceptual audio coding
- DFT noise substitution
- DCT with vector quantization
- Modified Discrete Cosine Transform (MDCT)
- MDCT with vector quantization

3.6.1 Optimum Coding in the Frequency Domain

The 132 Kbps algorithm known as OCFD was proposed by Brandenburg [3.77]. It is in some respects similar to the well-known Adaptive Transform Coding (ATC) for speech. Block scheme of the OCFD is shown in Figure 3.8. The input signal is first buffered in 512 sample blocks and transformed to the frequency domain using the DCT. Next, transform coefficients are quantized and entropy coded. A single quantizer is used for all transform coefficients. Adaptive quantization and entropy coding work together in an iterative procedure to achieve a fixed bit rate. The initial quantizer step size is derived from the SFM given by formula (3.8). In the inner loop, the quantizer step size is iteratively increased, and a new entropy-coded bit stream is formed at each update until the desired bit rate is achieved. Increasing the step size at each update produces fewer levels, which in turn reduces the bit rate. Using a second iterative procedure, psychoacoustic masking is introduced after the inner loop is done. First, critical bound analysis is applied. Then, a masking function is applied which combines a flat 6 dB masking threshold with an interband masking threshold leading to an estimate of JND for each critical band. If after inner loop quantization and entropy coding the measured distortion exceeds JND in at least one critical band, quantization step sizes are adjusted in the out-of-tolerance critical bands only. The outer loop repeats until JND criteria are satisfied or a maximum loop cannot be reached. Entropy-coded transform coefficients are then

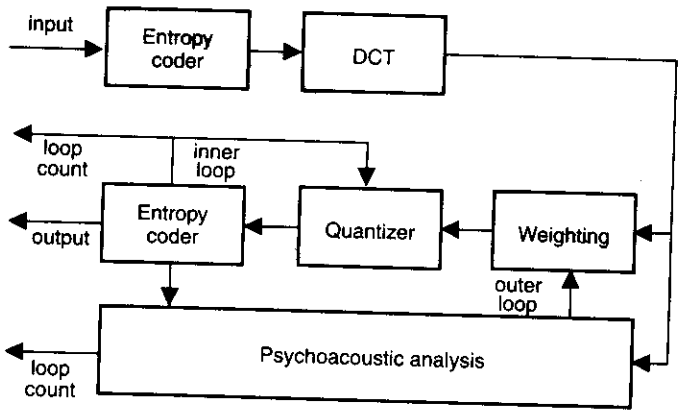

Figure 3.8 Optimum coding in the frequency domain [3.77]. ©1987 IEEE.

transmitted to the receiver along with side information, which includes the log-encoded SFM, the number of quantizer updates during the inner loop and the number of step-size reductions that occurred for each critical band in the outer loop. This side information is sufficient to decode the transform coefficients and to perform reconstruction at the receiver.

3.6.2 Perceptual Transform Coder

The idea behind the perceptual transform coder is to estimate the amount of quantization noise that can be inaudibly injected into each transform domain subband using PE estimates. The coder is memoryless and works as follows [3.78]. The signal is first windowed into overlapping (1/16) segments and transformed using a 2,048-point FFT. Next, the PE procedure is used to estimate JND thresholds for each critical band. Then, an iterative quantization loop adapts a set of 128 subband quantizers to satisfy the JND thresholds until the fixed rate is achieved. Finally, quantization and bit packing are performed. Quantized transform coefficients are transmitted to the receiver along with appropriate side information. Quantization of subbands consists of 8 sample blocks of complex-valued transform coefficients. The quantizer adaptation loop fast initializes the $j \in [1,128]$ subband quantizers (1,024 unique FFT coefficients /8 coefficients per subband) with k_j levels and step sizes of T_j as follows:

$$k_j = 2 * \text{int}\left(\frac{P_j}{T_i} + 1\right) \tag{3.13}$$

where T_i are the quantized critical band JND thresholds, P_j is the quantized magnitude of the largest real or imaginary part of transform coefficients in the j-th subband, while int(.) is the nearest integer rounding function. The block scheme of the perceptual transform coder is shown in Figure 3.9. The adaptation process involves repeated application of two steps. First, bit packing is attempted using the current quantizer set. Although many bit-packing techniques are possible, one simple scenario involves sorting quantizers in k_j order and then filling 64-bit words

with encoded transform coefficients according to the sorted results. After bit packing, T_i are adjusted by a carefully controlled scale factor, and the adaptation cycle repeats. Quantizer adaptation halts as soon as the packed data length satisfies the desired bit rate. Both P_j and the modified T_i are quantized on a dB scale using 8-bit nonuniform quantizers with a 170-dB dynamic range. These parameters are transmitted as side information and are used at the receiver to recover quantization levels (and thus implicit bit allocations) for each subband. They are in turn used to decode quantized transform coefficients.

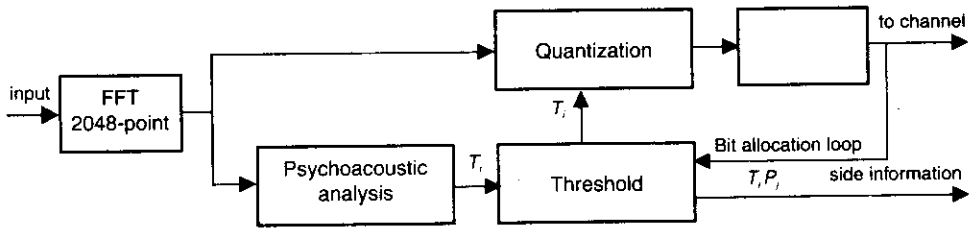

Figure 3.9 Perceptual transform coder.

3.6.3 Hybrid Coder

A hybrid coder is both a subband and transform coder. The idea behind the hybrid coder is to improve time and frequency resolution to OCFD and perceptual transform coder by constructing a filter bank that more closely resembles the human ear. This is accomplished at the encoder by first splitting the input signal into four octave-width subbands using a Quadrature Mirror Filter (QMF) filterbank. The decimated output sequence from each subband is then followed by one or more transforms to achieve the desired time-frequency resolution. Filterbank structure is shown in Figure 3.10 [3.95].

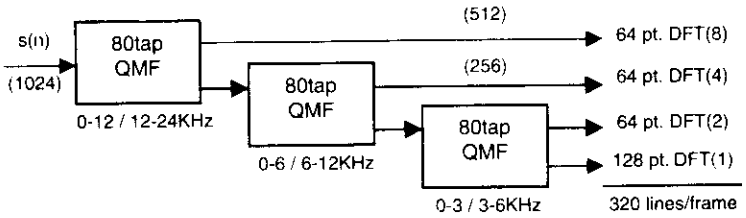

Figure 3.10 Filterbank structure [3.95].

Both DFT and MDCT transforms were investigated. Given the tiling of the time-frequency plane shown in Figure 3.11, frequency resolution at low frequency (24 KHz) is well matched to the ear, while the time resolution at high frequencies (2.7 ms) is sufficient for pre-echo control [3.95]. The quantization and coding schemes of the hybrid coder combine elements from both perceptual transform coder and OCFD. Masking thresholds are estimated using the perceptual

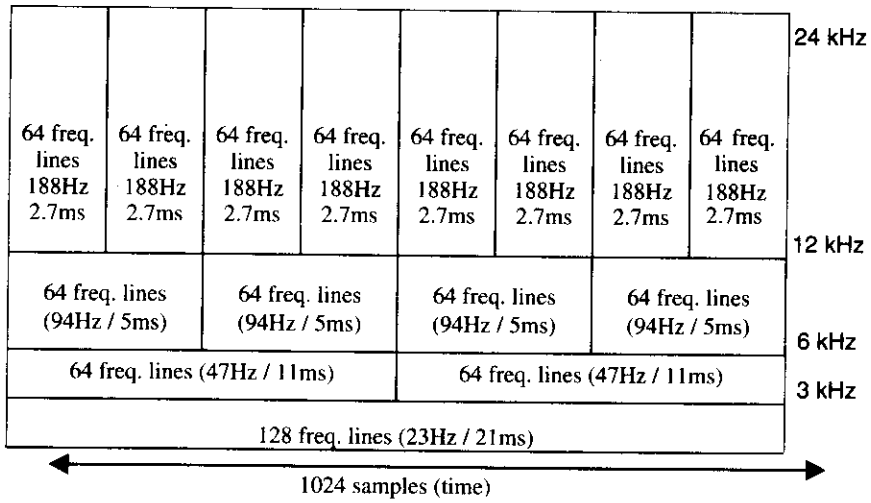

Figure 3.11 Time-frequency tiling [3.95].

transform coder approach for eight time slices in each frequency subband. The hybrid coder employs a quantization and coding scheme borrowed from OCFD.

As far as quality, the hybrid coder without any explicit pre-echo control mechanisms was reported to achieve quality better than or equal to OCFD at 64 Kb/s. The only disadvantage noted by the authors was increased complexity.

3.6.4 Transform Coding Using DFT Interblock Redundancy

A DFT-based audio-coding system that introduced a scheme to DFT interblock redundancy was proposed by Mahieux et al. [3.92]. Nearly transparent quality for 15 KHz audio at 96 Kbps, except for some highly harmonic signals, was reported. The encoder applies first-order backward-adaptive predictors (across time) to DFT magnitudes and differential pulse components. Then it quantizes separately the prediction residuals. Magnitudes and differential phase residuals are quantized using an adaptive nonuniform pdf-optimized quantizer designed for a Laplacian distribution and an adaptive uniform quantizer, respectively. The backward-adaptive quantizers are reinitialized during transients. Bits are allocated during step-size adaptation to shape quantization noise such that a psychoacoustic noise threshold is satisfied for each block. The use of linear prediction is justified because it exploits magnitude and different phase-time redundancy, which tends to be large during periods when the audio signal is quasistationary, especially for signal harmonics. For example, quasistationarity might occur during a sustained note.

3.6.5 ASPEC

ASPEC was claimed to produce better quality than any of the individual coders at 64 Kbs. The structure of ASPEC combines elements from all of its predecessors. Like OCFD, and transform

coding using DFT interblock redundancy, ASPEC uses the MDCT for time-frequency mapping. The masking model is similar to that used in the perceptual transform coder and the hybrid coder, including the sophisticated tonality estimation scheme at lower bit rates. The quantization and coding procedures use the pair of nested loops, which is the block differential coding scheme developed in transform coding using DFT interblock redundancy. Moreover, long runs of masked coefficients are run length and Huffman encoded. Quantized scale factors and transform coefficients are Huffman coded also. Pre-echoes are controlled using a dynamic window-switching mechanism. Pre-echoes occur when a signal with a sharp attack begins near the end of a transform block immediately following a region of low energy. ASPEC offers several modes for different quality levels, ranging from 64 to 192 Kbps per channel. A real-time ASPEC implementation for coding one channel at 64 Kbps was realized on a pair of 33 MHz Motorola DSP56001 devices.

3.6.6 Differential Perceptual Audio Coder

DPAC makes use of a scheme for exploiting long-term correlations [3.96]. DPAC works as follows. Input audio is transformed using modified DCT (see Section 3.6.9). A two-state classifier then labels each new frame of transform coefficients as either a reference frame or a simple frame. Reference frames contain significant audible differences from the previous frame. The classifier labels nonreference frames as simple. Reference frames are scalar quantized and encoded using psychoacoustic bit-allocation strategies similar to perceptual audio coder. However, simple frames are subjected to coefficient substitution. The magnitude differences of coefficients with respect to the previous reference frame below an experimentally optimized threshold, those coefficients are replaced at the decoder by the corresponding reference frame coefficients. The encoder then replaces subthreshold coefficients with zeros, thus saving transmission bits. Unlike the interframe predictive coding schemes, the DPAC coefficient substitution system is advantageous in that the simple frame bit allocation will always be less than or equal to the bit allocation that would be required if the frame was coded as a reference frame. Superthreshold simple frame coefficients are coded in the same way as reference frame coefficients. DPAC performance can be evaluated for frame classifiers using different selection criteria.

Under the Euclidean criterion, test frames satisfying the inequality

$$\left[\frac{\mathbf{s}_d^T \mathbf{s}_d}{\mathbf{s}_r^T \mathbf{s}_r}\right]^{1/2} \leq \lambda \tag{3.14}$$

are classified as simple, where the vectors s_r and s_i, respectively, contain reference and test frame time-domain samples, while the difference vector, s_d is defined as

$$\mathbf{s}_d = \mathbf{s}_r - \mathbf{s}_t \tag{3.15}$$

Under the PE criterion (3.12), a test frame is labeled as simple, if it satisfies the inequality

$$\frac{PE_s}{PE_R} \leq \lambda \tag{3.16}$$

where PE_s corresponds to the PE of the simple (coefficient-substituted) version of the test frame, and PE_R corresponds to the PE of the unmodified test frame. λ is the decision threshold.

Finally, under the SFM criterion (3.8), a test frame is labeled as simple if it satisfies the inequality

$$abs\left(10\log_{10}\frac{SFM_T}{SFM_R}\right) \leq \lambda \tag{3.17}$$

where SFM_T corresponds to the test frame SFM, and SFM_R corresponds to the SFM of the previous reference frame. The decision threshold λ was experimentally optimized for all three criteria. Best performance was obtained while encoding source material using the PE criterion. As far as overall performance is concerned, NMR measurements were compared between DPAC and perceptual transform coding algorithm at 64, 88 and 128 Kbps. Despite an average drop of 30 to 35% in PE measured at the DPAC coefficient-substitution stage output relative to the coefficient substitution input, comparative NMR studies indicated that DPAC outperforms PTC only below 88 Kbps and then only for certain types of source material such as pop or jazz music. The desirable PE reduction led to an undesirable drop in reconstruction quality.

3.6.7 DFT Noise Substitution

In this method, noise-like spectral regions are identified as follows [3.97]. First, LMS adaptive Linear Predictors (LP) are applied to the output channels of a multiband QMF analysis filterbank that has as input, the original audio $s(n)$. A predicted signal, $\hat{s}(n)$, is obtained by passing the LP output sequences through the QMF synthesis filterbank. Prediction is done in subbands rather than across the entire spectrum to prevent classification errors that could result if high-energy noise subbands are allowed to dominate predictor adaptation, resulting in misinterpretation of low-energy tonal subbands as noisy. Next, the DFT is used to obtain magnitude $\left(S(k),\hat{S}(k)\right)$ and phase components $\left(\Theta(k),\hat{\Theta}(k)\right)$ of the input, $s(n)$, and prediction, $\hat{s}(n)$, respectively. Then, tonality, $T(k)$, is estimated as a function of the magnitude and phase predictability, for example,

$$T(k) = \alpha\left|\frac{S(k)-\hat{S}(k)}{S(k)}\right| + \beta\left|\frac{\Theta(k)-\hat{\Theta}(k)}{\Theta(k)}\right| \tag{3.18}$$

where α and β are experimentally determined constants. Noise substitution is applied to contagious blocks of transform coefficient bins for which $T(k)$ is very small. The 15% average bit-rate savings realized using this method in conjunction with transform coding is offset to a large extent by a significant complexity increase due to the additions of the adaptive linear predictors and a multiband analysis-synthesis QMF filterbank.

3.6.8 DCT With Vector Quantization

After computing the DCT on 512 audio sample blocks, the algorithm uses Multistage Tree-structured VQ (MSTVQ) scheme for quantization of normalized vectors, with each vector containing 4 DCT components. Bit allocation and vector normalization are derived at both the encoder and decoder from a sampled power-spectral envelope that consists of 29 groups of transform coefficients. A simplified masking module assumes that each sample of the power-spectral envelope represents a single masker [3.79]. Masking is assumed to be additive, as in the ASPEC algorithm. Thresholds are computed as a fixed offset from a masking level. To achieve high quality, a strong correlation between SFM and the amount of offset was observed. Two-segment scalar quantizers that are piecewise linear on a dB scale are used to encode the power-spectral envelope. Quadratic interpolation is used to restore full resolution to the subsampled envelope.

In another approach to quantization of transform coefficients, Constrained-Storage Vector Quantization (CS-VQ) techniques are combined with the MSTVQ from the original coder, allowing the new coder to handle peak NMR requirements without impractical codebook storage requirements [3.98]. In fact, CS-MSTVQ enabled quantization of 1,274 coefficient vectors using only 4 unique quantizers. Power-spectral envelope quantization is enhanced by extending its resolution to 127 samples. The samples are then encoded using a two-stage process. The first stage applies Nonlinear Interpolative Vector Quantization (NLIVQ), a dimensionality reduction process that represents the 127-element power spectral envelope vector using only a 12-dimensional feature power-spectral envelope. Unstructured VQ is applied to the feature power envelope. Then a full-resolution quantized envelope is obtained from the unstructured VQ index in the corresponding interpolation codebook. In the second stage, segments of the envelope residual are encoded using a set of 8-, 9- and 10-element quantizers.

3.6.9 MDCT

The MDCT offers the advantage of overlapping time windows while managing to preserve critical sampling. The analysis window must be carefully designed such that the time domain aliasing introduced by 50% overlap and 2:1 decimation will cancel in the inverse transformation [3.99]. The MDCT analysis expression is

$$X(k) = \sum_{n=0}^{2N-1} h(n)\, x(n) \cos\left[\frac{\pi}{2N}(2k+1)(2n+1+N)\right] \tag{3.19}$$

where $k = 0, 1, \ldots, 2N-1$. The analysis window must satisfy

$$h^2(N-1-n) + h^2(n) = 2, \quad 0 \le n < N \tag{3.20}$$

$$h^2(N+n) + h^2(2N-1-n) = 2, \quad 0 \le n < N \tag{3.21}$$

An example analysis window that produces the desired time-domain aliasing cancellation is given by

$$h(n) = \pm\sqrt{2}\, \sin\left[\left(n+\frac{1}{2}\right)\frac{\pi}{2N}\right] \tag{3.22}$$

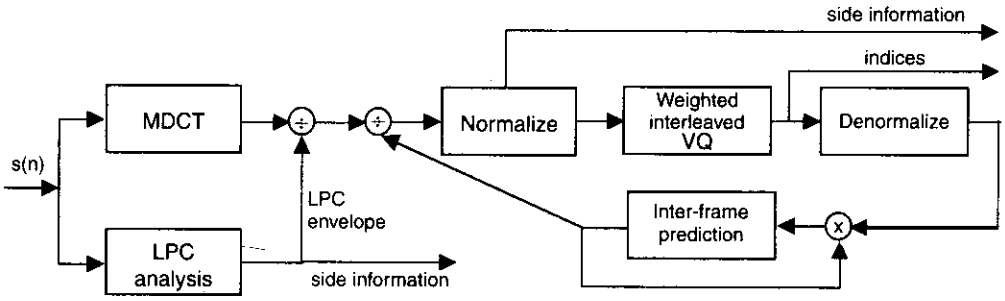

Figure 3.12 TWIN-VQ encoder [3.100]. ©1995 IEEE.

The development of FFT-based fast algorithms for the MDCT has made it viable for real-time applications [3.100, 3.101, 3.102].

3.6.10 MDCT with VQ

Transform Domain-Weighted Interleave Vector Quantization (TWIN-VQ), an MDCT-based coder that involves transform coefficients VQ, was developed by Iwakani et al. [3.100]. This algorithm exploits LPC analysis, spectral interframe redundancy, and interleaved VQ. A TWIN-VQ encoder is presented in Figure 3.12.

At the encoder, each frame of MDCT coefficients is first divided by the corresponding elements of the LPC spectral envelope, resulting in a spectrally flattened quotient (residual) sequence. This procedure flattens the MDCT envelope, but does not affect the fine structure. Therefore, the next step divides the first step residual by a predicted fine-structure envelope. This predicted fine-structure envelope is computed as a weighted sum of three previously quantized fine structure envelopes, that is, using backward prediction. Interleave VQ is applied to the normalized second step residual. The interleave VQ vectors are structured in the following way. Each N-sample normalized second step residual vector is split into K subvectors, each containing N/K coefficients. Second step residuals from N-sample vector are interleaved in the K subvectors such that subvector i contains elements $i+nK$, where $n = 0,1,\ldots,(N/K)-1$. Perceptual weighting is also incorporated by weighting each subvector by a nonlinearly transformed version of its corresponding LPC envelope component prior to the codebook search. VQ indexes are transmitted to the receiver. Side information consists of VQ normalization coefficients and the LPC-encoded envelope. Enhancements to the weighted interleaving scheme and LPC envelope representation are reported by Moriya et al. [3.101]. This has enabled real-time implementation of stereo decoders on Pentium and power PC platforms.

3.7 Audio Subband Coders

Like the transform coders, subband coders also exploit signal redundancy and psychoacoustic irrelevance in the frequency domain [3.102]. Instead of unitary transforms, these coders rely upon frequency-domain representations of the signal obtained from banks of bandpass filters.

The audible frequency spectrum (20Hz to 20KHz) is divided into frequency subbands using a bank of bandpass filters. The output of each filter is then sampled and encoded. At the receiver, the signals are demultiplexed, decoded, demodulated and then summed to reconstruct the signal [3.103, 3.104, 3.105]. Audio subband coders realize coding gains by efficiently quantizing and encoding the decimated output sequences from perfect reconstruction filterbanks. Efficient quantization methods usually rely upon psychoacoustically controlled dynamic bit allocation rules, which allocate bits to subbands in such a way that the reconstructed output signal is free of audible quantization noise or other artifacts [3.106]. In a generic subband audio coder, the output signal is first split into several uniform or nonuniform subbands using some critically sampled, perfect reconstruction filterbanks. Nonideal reconstruction properties in the presence of quantization noise are compensated for by using subband filters that have very good attenuation. This requires high-order filters. Then, decimated output sequences from the filterbank are normalized and quantized for short, 2 to 10 ms blocks. Psychoacoustic signal analysis is used to allocate an appropriate number of bits for the quantization of each subband. The usual approach is to allocate a just-sufficient number of bits to mask quantization noise in each block while simultaneously satisfying some bit-rate constraint. Because masking thresholds and hence, bit-allocation requirements, are time varying, buffering is often introduced to match the coder output to a fixed rate. The encoder sends to the decoder quantized subband output samples, normalization scale factors for each block of samples and bit-allocation side information. Bit allocation may be transmitted as explicit side information, or it may be implicitly represented by some parameters, such as the scale-factor magnitudes. The decoder uses side information and scale-factors in conjunction with an inverse filterbank to reconstruct a coded version of the original input. Numerous subband coding algorithms for high-fidelity audio have appeared in the literature [3.106, 3.107, 3.108, 3.109, 3.110].

3.7.1 Wavelet Decompositions

In-depth technical information regarding wavelets is available in many references [3.111]. It is useful to summarize some basic wavelet characteristics. Wavelets are a family of basis functions in the space of square integrable signals. A finite energy signal can be represented as a weighted sum of the translates and dilates of a single wavelet. Continuous-time wavelet-signal analysis can be extended to discrete time and square summable sequences. Under certain assumptions, the Discrete Wavelet Transform (DWT) acts as an orthonormal linear transform $T: R^N \rightarrow R^N$. For a finite support wavelet of length K, the associated transformation matrix, Q, is fully determined by a set of coefficients $\{c_k\}$ for $0 \leq k \leq K-1$. This transformation matrix has an associated filter bank interpolation. One application of the transform matrix, Q, to an Nx1 signal vector, x, generates an Nx1 vector of wavelet-domain transform coefficients, y. The Nx1 vector y can be separated into two $(N/2)$x1 vectors of approximation and detail coefficients, y_{lp} and y_{hp}, respectively. The spectral content of the signal x captured in y_{lp} and y_{hp} corresponds to the fre-

quency subbands realized in 2:1 decimated-output sequences from a QMF filterbank that obeys the power complementary condition, that is,

$$\left|H_{lp}(\Theta)\right|^2 + \left|H_{lp}(\Theta+\pi)\right|^2 = 1 \tag{3.23}$$

where $H_{lp}(\Theta)$ is the frequency response of the lowpass (LP) filter.

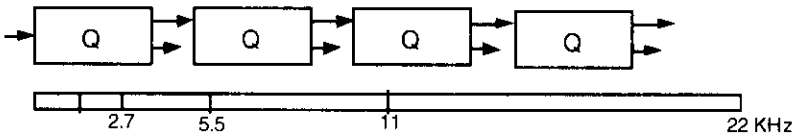

Figure 3.13 Wavelet decomposition [3.82]. ©1997 IEEE.

Successive applications of the DWT can be interpreted as passing input data through a cascade of banks of perfect reconstruction LP and highpass (HP) filters followed by 2:1 decimation. In effect, the forward/inverse transform matrixes of a particular wavelet are associated with a corresponding QMF analysis/synthesis filterbank. The usual wavelet decomposition implements an octave-band filterbank structure shown in Figure 3.13. Frequency subbands associated with the coefficients from each stage are schematically represented for an audio signal sampled at 44.1 KHz.

Wavelet packet representations decompose both the detail and approximation coefficients at each stage of the tree as shown in Figure 3.14. Frequency subbands associated with the coefficients from each stage are schematically represented for an audio signal sampled at 44.1 KHz. A filterbank interpolation of wavelet transforms is attractive in the context of audio coding algorithms for at least two reasons. First, wavelet or wavelet packet decompositions can be tree structured as necessary (unbalanced trees are possible) to decompose input audio into a set of frequency subbands tailored to some application. For example, it is possible to approximate the critical band auditory filterbank using a wavelet packet approach. Second, many K-coefficient finite support wavelets are associated with a single magnitude frequency-response QMF pair. Therefore, a specific subband decomposition can be realized while retaining the freedom to choose a wavelet basis, which is in some sense optimal.

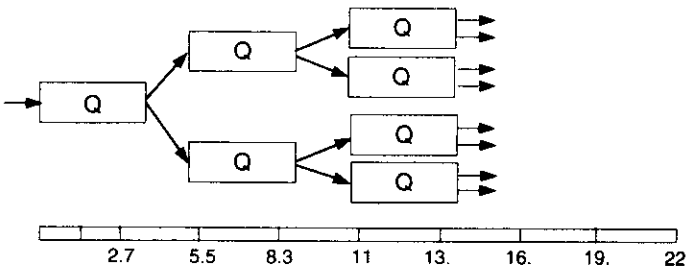

Figure 3.14 Wavelet packet decomposition [3.82]. ©1997 IEEE.

3.7.2 DWT-based Subband Coders

The basic idea behind DWT-based subband coders is to quantize and encode efficiently the coefficient sequences associated with each stage of the wavelet decomposition tree. Irrelevancy is exploited by transforming frequency-domain masking thresholds to the wavelet domain and shaping wavelet-domain quantization noise such that it does not exceed the masking threshold. Wavelet-based subband algorithms also exploit statistical signal redundancies through differential, run-length, and entropy coding schemes. The next few subsections concentrate on DWT-based subband coders developed in [3.112, 3.113, 3.114], including the hybrid sinusoidal/wavelet transform algorithm [3.115]. Other studies of DWT-based audio coding schemes concerned with low-complexity, low-delay, combined wavelet/multipulse LPC coding, and combined scalar/vector quantization of transform coefficients were reported, respectively in [3.116, 3.117, 3.118, 3.119, 3.120].

3.8 Speech Coder Attributes

Speech-coding attributes can be divided into four categories: bit rate, delay, complexity and quality [3.121]. It is possible to relax requirements for the less important attributes so that the more important requirements can be met.

Bit rate is the attribute that most often comes to mind first when discussing speech coders. The range of bit rates that has been standardized is from 2.4 Kb/s for secure telephony to 64 Kb/s for network applications. The coders standardized by the ITU are of primary interest [3.122]. The 64 Kb/s G.711 PCM coder is used in digital telephony networks and switches throughout the world. The 32 Kb/s G.726 Adaptive Differential (AD) PCM coder is used for circuit-manipulation equipment to increase effectively the capacity of undersea cables and satellite links. The 64/56/48 Kb/s G.722 and 16 Kb/s G.728 coders are used in video teleconferencing across ISDN or frame relay connections. The G.723.1 and G.729 coders have been standardized for low-bit-rate multimedia applications across telephony modems.

The codec delay can have a large impact on individual coder suitability for a particular application. Speech coders for real-time conversations cannot have too much delay, or they will quickly become unsuitable for network applications. For multimedia storage applications with only one-way transmission of speech, the coder can have virtually unlimited delay and still be suitable for the application. Psychologists who have studied conversational dynamics know that, if the one-way delay of a conversation is greater than 300 ms, the conversation will become more like a half-duplex, or a push-to-talk experience, rather than an ordinary conversation. In contrast, a speech or audio file of 300 ms or more before starting will be virtually imperceptible to the user. Thus, a conversation is an application that is the most sensitive to coder delay, while one involving speech storage is the least-delay-sensitive application. The components of the total system delay include the frame size, the look-ahead, other algorithmic delay, multiplexing delay, processing delay for computation and transmission delay. The algorithm used for the speech coder will determine the frame size and the look-ahead. Its complexity will have an

impact on the processing delay. The network connection determines the multiplexing and transmission delays.

Recent multimedia speech coders have been implemented on the host Central Processing unit (CPU) of personal computers and workstations. The measures of complexity for a Digital Signal Processor (DSP) and a CPU are somewhat different due to the nature of these two systems. At the heart of complexity is the row number of computational instructions required to implement the speech coder. DSP chips from different vendors have different architectures and consequently different efficiencies in implementing the same coder. The measure used to indicate the computational complexity is the number of instructions per second required for implementation. This is usually expressed in Millions of Instructions per Second (MIPS). The numbers given in Table 3.3 are DSP MIPS for the ITU-T standards-based coders.

Table 3.3 ITU-T speech-coding standards [3.121]. ©1998 IEEE.

Standard	Bit Rate	Frame Size/Look-Ahead	Complexity
G.711 PCM	64 Kb/s	0 / 0	0 MIPS
G.726, G.721, G.723, G.727, ADPCM	16, 24, 32, 40 Kb/s	0.125 ms / 0	2 MIPS
G.722 Wideband coder	48, 56, 64 Kb/s	0.125 / 1.5 ms	5 MIPS
G.728 LD-CELP	16 Kb/s	0.625 ms / 0	30 MIPS

The ideal speech coder has a low bit rate, high perceived quality, low signal delay and low complexity. No ideal coder as yet exists with all these attributes. Real coders make tradeoffs among these attributes, for example, trading off higher quality for increased bit rate, increased delay or increased complexity. Figure 3.15 shows a plot of speech quality as measured subjectively in terms of Mean Opinion Scores (MOSs) for a range of telephone bandwidth coders spanning bit rates from 64 Kb/s down to 2.4 Kb/s. Curves of quality based on measurements made in 1980 and 1990 are shown. The MOS subjective test of speech quality uses a five-point rating scale with the following attributes:

5—Excellent quality, no noticeable impairments

4—Good quality, only very slight impairments

3—Fair quality, noticeable but acceptable impairments

2—Poor quality, strong impairments

1—Bad quality, highly degraded speech

As can be seen, the telephone bandwidth coders maintain a uniformly high MOS for bit rates ranging from 64 Kb/s down to about 8 Kb/s, but fall steadily for bit rates below 8 Kb/s.

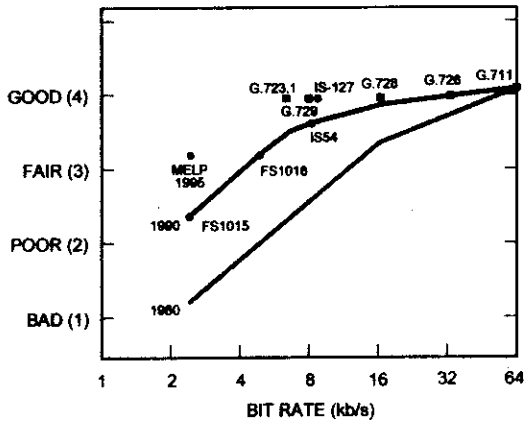

Figure 3.15 Subjective quality of various speech coders versus bit rate [3.121]. ©1998 IEEE.

3.9 CD Audio Coding for Multimedia Applications

A key aspect of multimedia systems is the ability to provide CD quality audio across telecommunications networks. Because uncompressed CD audio requires 1.4 Mb/s for transmission [3.121], high-quality coding of CD audio is essential for its practical use in any multimedia application. The state of the art in CD audio coding has improved dramatically in the course of the last decade. This is due to the relentless exploitation of unknown and well-understood properties of the human auditory system. Almost all modern CD audio coders use a quantitative model of the human auditory system to drive the signal quantization so that the resulting distortion is imperceptible. Those features of the audio signal that are determined not to be audible are discarded. The amount of quantization noise that is inaudible can also be calculated. This combination of discarding inaudible features and quantizing the remaining features so that the quantization noise will be inaudible is well known as perceptual coding. It has made its greatest impact to date in the field of audio coding and has been extended to speech, image and video coding. Remember that perceptual coding is a lossy coding method, that is, the output signal is not the same as the input signal. The imperceptible information removed by the perceptual coder is called the irrelevancy. In practice, most perceptual coders attempt to remove both irrelevancy and redundancy in order to make a coder that provides the lowest bit rate possible for a given quality. Most perceptual coders have lower SNR than source coders, but have better subjective quality for an equivalent bit rate.

3.10 Image Coding

Image coding involves compressing and coding a wide range of still images, including bilevel (fax) images, photographs and document images containing text, handwriting, graphics, and so forth. There are a number of important multimedia applications. These include the following:

- Slideshow graphics for applications such as ordering from catalogs, shopping from home, viewing real estate, and so forth. For this class of applications, it is essential that the images be presented at a variety of resolutions, including low resolution for fast browsing and searching and high resolution for detailed inspection. This type of application demands a 3D model that allows the user to view the image from different frames of reference.
- Creation, display, editing, access and browsing of banking images (electronic checks) and forms (insurance forms, medical forms, and so forth). For this application, it is essential that the user be able to interact with the system that creates and displays the images and that multimedia attachments be easily accessible by the user.
- Medical applications where sophisticated high-resolution images need to be stored, indexed, accessed and transmitted from site to site on demand. Examples of such images include medical X-rays, NMR, Electroencephalogram (EEG) and Electrocardiogram (EKG) scans, and so forth.

In order to compress and code image and video signals, it is essential to take advantage of any observable redundancy in the signal. The obvious forms of signal redundancy in most image and video signals include spatial redundancy and temporal redundancy [3.123].

Spatial redundancy takes a variety of different forms in an image, including correlations in the background image, correlations across an image and spatial correlations that occur in the spectral domain. A variety of techniques have been devised to compensate for spatial redundancy in image and video sequences [3.121]. Temporal redundancy takes two different forms in video sequences. The most obvious form is redundancy from repeated objects in consecutive frames of a video sequence. Such objects can move horizontally, vertically or any combination of directions, and they can disappear from the image as they move out of view [3.124].

The second basic principle of image coding is to take advantage of the HVS, which is used to view the coded image and video sequences in the same way that we take advantage of the human hearing system for listening to speech and audio signals. By understanding the perceptual masking properties of the HVS and their sensitivity to various types of distortion as a function of image intensity, texture and motion, we can develop a profile of the signal levels that provide JND in the image and video signals. By creating this JND profile for each image to be coded, it is possible to create image quantization schemes that hide the quantization noise under the JND profile and thereby make the distortion become perceptually invisible.

To illustrate the potential coding gains that can be achieved using a perceptual coding distortion measure, consider the set of curves of PE (distortion) of a black-and-white still image versus the image-coding rate measured in bits/sample or equivalently bits/pixel as shown in Figure 3.16. The upper curve (uniform quantization) shows that, in theory, it would take more than 8 bits/pixel to achieve low (essentially zero) distortion. By using proper compounding methods, this zero-distortion point can be reached with just 8 bits/pixel in practice as shown by the second curve. Taking into account noiseless coding methods (Huffman coding or arithmetic coding), the

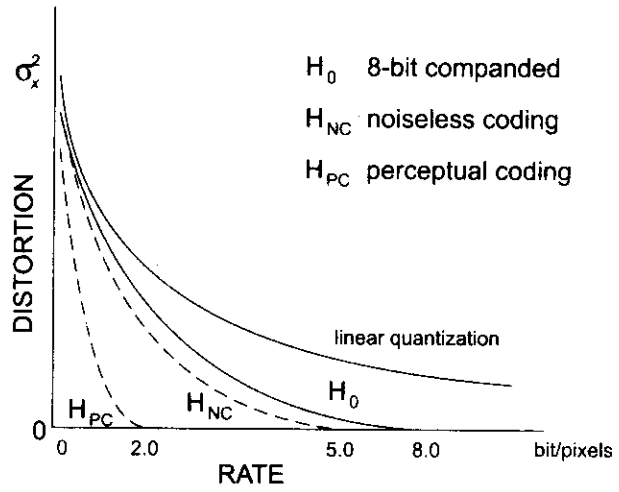

Figure 3.16 PE versus the coding rate [3.89]. ©1993 IEEE.

Noiseless Coding (NC) threshold can be reduced by a factor of nearly 2 to 1, down to 5 bits/pixel, as shown by the third curve. Last, by exploiting the perceptual model, the perceptually lossless coding threshold can be reduced by another factor of nearly three, down to around 2 bits/pixel or below depending on the image as shown by the dotted curve.

A block diagram of a generic image coding algorithm based on perceptual-coding principles is shown in Figure 3.17. The first step is to perform a short-term spectral analysis of the input image in order to determine a profile of image intensity and image texture. This short-term profile is then fed into the JND estimation box, which converts the measured image characteristics into a JND profile. It is then used as the input to an adaptive coder, which provides either a constant bit-rate (variable quality) output or a constant quality (variable bit rate) output signal.

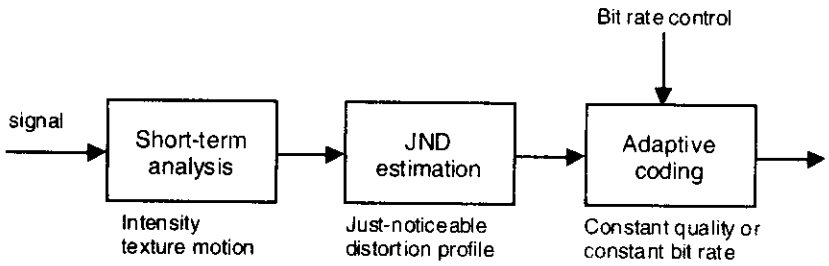

Figure 3.17 Image-coding algorithm based on perceptual coding [3.89]. ©1993 IEEE.

3.11 Video Coding

Video signals differ from image signals. The most important difference is that video signals have an associated frame rate of anywhere from 15 to 60 frames/s, which provides the illusion of motion in the displayed signal. A second difference is the frame size. Video signals may be as small as Quarter Common Intermediate Format (QCIF), 176x144 pixels, and as large as HDTV (1920x1080 pixels), whereas still images are sized primarily to fit PC color monitors (640x480 pixels or 124/768 pixels). A third difference between images and video is the ability to exploit temporal masking as well as spectral masking in designing compression methods for video. One can also take advantage of the fact that objects in video sequences tend to move in predictable patterns and can therefore be motion compensated from frame to frame if we can reliably detect both the object and its motion trajectory over time.

Some major initiatives in video coding have led to a range of video standards. These include the following:

- Video coding for video teleconferencing, which has led to ITU standards H.261 and for ISDN video conferencing [3.125], H.263 for POTS video conferencing [3.126, 3.127], and H.262 for ATM/broadband video conferencing, and digital TV.
- Video coding for storing movies on CD Read-Only Memory (ROM), on the order of 1.2 Mb/s allocated to video coding and 256 Kb/s allocated to audio coding, which led to the initial ISO MPEG-1 standard [3.128].
- Video coding for storing broadcast video on digital video disk (DVD), on the order of 2-15 Mb/s allocated to video and audio coding, which led to the ISO MPEG-2 standard [3.129, 3.130, 3.131, 3.132].
- Video coding for low bit-rate video telephony over POTS networks, with as little as 10 Kb/s allocated to video and as little as 5.3 Kb/s allocated to voice coding, which led to H.324 standard [3.57].
- MPEG-4 audio-video coding (both synthetic and natural). Also object/content-based coding with user interactivity. Designed for Internet and mobile communications. A Very Low Bit Rate (VLBR) core coder is to be compatible with H.263.
- Video coding for advanced HDTV, with 15 to 4000 Mb/s allocated to the video coding.
- MPEG-7 Multimedia content description interface. To facilitate truly integrated multimedia search engines based on descriptors, description schemes and description definition language.

3.11.1 TC and Subband Coding (SBC)

TC and SBC refer to compression systems where the signal decomposition is implemented using an analysis filterbank. At the receiver, the signal is reassembled by a synthesis filterbank. By TC, we usually mean that the linear transform is block based. When transform coding is interpreted as an SBC technique, the impulse responses of the analysis and synthesis filters are at most as long as the subsampling factor employed in the subbands. Thus, the image can be subdivided into

blocks that are processed in an independent manner. On the other hand, general SBC allows the impulse responses to overlap and thus includes transform coding as a special case.

One of the most important tasks of the transform is to pack the energy of the signal into as few transform coefficients as possible. The DCT yields nearly optimal energy concentration [3.133]. Almost all image transform coders today employ the block-wise DCT, usually with a block size of 8x8 pixels. The transform is followed by quantization and entropy coding. Typically, the transform coefficients are run-length encoded. That is, successive zeros along a zigzag path are grouped with the first nonzero amplitude into a joint symbol, which is then Huffman coded. The Lapped Orthogonal Transform (LOT) could be substituted for the DCT to avoid some of the typical blocking artifacts that become visible with coarse quantization.

The full potential of SBC is unleashed when nonuniform band splitting is used to build multiresolution representations of an image. Besides excellent compression, multiresolution coders provide the successive approximation feature. As higher-frequency components are added, higher-resolution, better quality images are obtained. Moreover, multiresolution techniques fit naturally into joint source-channel coding schemes. Subband coders with octave band decomposition are often referred to as DWT coders, or wavelet coders [3.134].

At present, many state-of-the-art multiresolution image coders draw on the ideas introduced by Shapiro in his Embedded Zero-Tree Wavelet (EZW) algorithm [3.135]. The algorithm employs a data structure called zero-tree, where one assumes that, if a coefficient at a low frequency is zero, it is highly likely that all the coefficients at the same spatial location at all higher frequencies will also be zero. Thus, when encountering a zero-tree root, one can discard the whole tree of coefficients in higher frequency bands. Moreover, the algorithm uses successive approximation quantization, which allows termination of encoding or decoding at any point. These ideas have produced a new class of algorithms aimed at exploiting both frequency and spatial phenomena [3.136]. Research has shown that wavelet coders can produce superior results, but transform coders employing a block-wise DCT are still dominant. After years of use, DCT coders are very well understood and many improvements have been made, for example, in the area of fast algorithms or by imposing perceptual criteria.

3.11.2 Predictive Coding

Except when used with SBC or TC, predictive coders do not decompose the image into independent components. Instead, both the coder and decoder calculate a prediction value for the current signal sample. Then, the prediction error, rather than the signal sample itself, is transmitted. This principle can be used for both lossy and lossless image coding. The predictors calculate linear combinations of previous image samples because general nonlinear predictors, addressed by combinations of, say, 8-bit pixels, would often require enormous lookup tables for the same performance. For lossy predictive coding, DPCM has been used since the early days of image coding. Intraframe DPCM exploits dependencies within a frame of a video sequence. Typically, pixels are encoded in line-scan order, and previous samples in the current line and samples from the previous line are combined for prediction. Today, this simple scheme has been displaced by

vastly superior transform/SBC schemes. In fact, lossy predictive intraframe coding is alive and well in the form of predictive closed-loop pyramid coders that feed back the quantization error before encoding the next higher-resolution layer. It has been shown that closed-loop pyramid coders even outperform the equivalent open-loop over complete pyramid representations when combined with scalar quantizers [3.137].

For interframe coding where statistical dependencies between successive frames of a video sequence are exploited, DPCM is the dominating scheme at present and for the foreseeable future. For example, other than spatio-temporal SBC, interframe DPCM avoids undesirable delay due to buffering of one or several frames. It is straightforward to incorporate motion adaptation and motion compensation into a temporal prediction loop and to combine motion-compensated prediction with other schemes for encoding the prediction error.

3.11.3 Motion-Compensated Video Coding

Motion compensation is a key element in most video coders. All modern video compression coders, such as those standardized in the ITU-T Rec. H.261 [3.125] and H.263 [3.126] or in the ISO MPEG standards [3.138], are motion-compensated hybrid coders. Motion-compensated hybrid coders estimate the displacement from frame to frame and transmit the motion vector field as side information in addition to the motion-compensated prediction error image. The prediction error image is encoded with an intraframe source encoder that exploits statistical dependencies between adjacent samples. The intraframe encoder is an 8x8 DCT coder in all current video coding standards, but other schemes, such as SBCs or VQs, can be used as well.

Figure 3.18 shows a block diagram of a motion-compensated image coder. The key idea is to combine TC in the form of the DCT of 8x8 pixel blocks with predictive coding in the form of differential PCM in order to reduce storage and computation of the compressed image and at the same time to give a high degree of compression and adaptability. Because motion compensation is difficult to perform in the transform domain, the first step in the interframe coder is to create a motion-compensated prediction error using macroblocks of 16x16 pixels. This computation requires only a single frame store in the receiver. The resulting error signal is transformed using DCT, followed by an adaptive quantizer, entropy encoded using a Variable Length Coder (VLC) and buffered for transmission across a fixed-rate channel.

The way that a motion estimator works is illustrated in Figure 3.19. A 16x16 pixel macroblock in the current frame is compared with a set of macroblocks in the previous frame to determine the one that best predicts the current macroblock. The set of macroblocks includes those within a limited region of the current macroblock. When the best matching macroblock is found, a motion vector is determined that specifies the reference macroblock and the direction of the motion of that macroblock.

Motion compensation works well for low spatial frequency components in the video signal. For high spatial frequency components, even a small inaccuracy of the motion compensation will render the prediction ineffective. Hence, it is important to use spatially a lowpass filter for the prediction signal by a loop filter. This loop filter is explicitly needed for integer-pixel

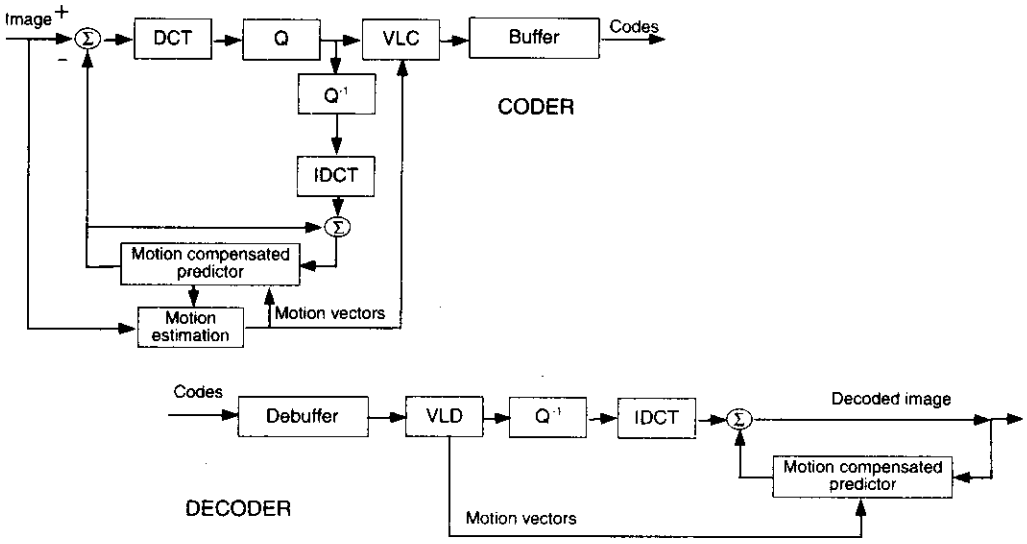

Figure 3.18 Motion-compensated coder and decoder for interframe coding [3.121]. ©1998 IEEE.

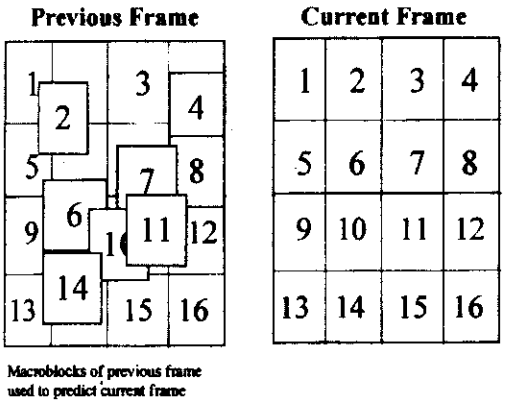

Figure 3.19 Illustration of motion-compensated coding of interframe macroblocks [3.121]. ©1998 IEEE.

accurate motion compensation. For subpixel accurate motion compensation, it can be incorporated into the interpolation kernel required to calculate signal samples between the original sampling positions. The loop filter also improves prediction by acting as a noise-reduction filter. Prediction can be further improved by combining multiple independently motion-compensated signals. Examples are the bidirectionally predicted B-frames in MPEG [3.138] or overlapped block motion compensation [3.139] that has also been incorporated in the ITU-T Recommendation H.263 [3.126, 3.127].

Motion-compensated hybrid coding can theoretically outperform an optimum interframe coder by at most 0.8 bit/pixel in moving areas of an image if motion compensation is performed

with only integer-pixel accuracy [3.140]. For half-pixel accuracy, this gain can be up to 1.3 bits/pixel. In addition, in nonmoving areas, or other parts of the image that can be predicted perfectly, no prediction error signal has to be transmitted, and these areas can simply be repeated from a frame store, a technique often referred to as conditional replenishment.

At low bit rates, motion compensation is severely constrained by the limited bit rate available to transmit the motion vector field as side information. Therefore, rate constrained estimation and a rate-efficient representation of the motion vector field are very important [3.141]. For simplicity, most practical video-coding schemes still employ blockwise constant motion compensation. More advanced schemes interpolate between motion vectors, employ arbitrarily shaped regions or use triangular meshes for representing a smooth motion vector field.

3.12 Watermarking

Data transmitted through a network may be protected from unauthorized receivers by applying techniques based on cryptography [3.142]. Only people who possess the appropriate private key can decrypt the received data using a public algorithm implemented either in hardware or in software. Fast implementation of encryption-decryption algorithms is highly desirable. Data-content manipulation can be performed for various legal or illegal purposes (compression, noise removal or malicious data modification). The modified product is not authentic with respect to the original one.

The technology of multimedia services grows rapidly, and distributed access to such services through computer networks is a matter of urgency. However, network access does not protect the copyright of digital products that can be reproduced and used illegally. An efficient way to solve this problem is to use watermarks [3.143, 3.144]. A watermark is a secret code described by a digital signal carrying information about the copyright property of the product. The watermark is embedded in the digital data such that it is perceptually not visible. The copyright holder is the only person who can show the existence of his own watermark and to prove the origin of the product.

Reproduction of digital products is easy and inexpensive. In a network environment, like the Web, retransmission of copies all throughout the world is easy. The problem of protecting the intellectual property of digital products has been treated in the last few years with the introduction of the notion of watermarks.

The following requirements should be satisfied by a watermarking algorithm:

- Alterations introduced in the image should be perceptually invisible.
- A watermark must be undetectable and not removable by an attacker.
- A sufficient number of watermarks in the same image, detectable by their own key, can be produced.
- The detection of the watermark should not require information from the original image.
- A watermark should be robust, as much as possible, against attacks and image processing, which preserves desired quality for the image.

Watermarks slightly modify the digital data to embed nonperceptible encoded copyright information. Digital data embedding has many applications. Foremost is passive and active copyright protection. Digital watermarking has been proposed as a means to identify the owner or distributor of digital data. Data embedding also provides a mechanism for embedding important control, descriptive or reference information in a given signal. A most interesting application of data embedding is providing different access levels to the data [3.145]. Most data-embedding algorithms can extract the hidden data from the host signal with no reference to the original signal [3.146].

The first problem that all data-embedding and watermarking schemes need to address is that of inserting data in the digital signal without deteriorating its perceptual quality. We must be able to retrieve the data from the edited host signal. Because the data insertion and data recovery procedures are intimately related, the insertion scheme must take into account the requirement of the data-embedding applications. Data insertion is possible because the digital medium is ultimately consumed by a human. The human hearing and visual systems are imperfect detectors. Audio and visual signals must have a minimum intensity or contrast level before they can be detected by a human. These minimum levels depend on the spatial, temporal and frequency characteristics of the human auditory and visual systems. Most signal-coding techniques exploit the characteristics of the human auditory and visual systems directly or indirectly. Likewise, all data-embedding techniques exploit the characteristics of the human auditory and visual systems implicitly or explicitly. A diagram of a data-embedding algorithm is shown in Figure 3.20. The information is embedded into the signal using the embedding algorithm and a key. The dashed lines indicate that the algorithm may directly exploit perceptual analysis to embed information. In fact, embedding data would not be possible without the limitations of the human visual and auditory systems.

Data embedding and watermarking algorithms embed text, binary streams, audio, image or video in a host audio, image or video signal. The embedded data are perceptually inaudible or invisible to maintain the quality of the source data. The embedded data can add features to the host multimedia signal, for example, multilingual soundtracks in a movie, or they can provide copyright protection.

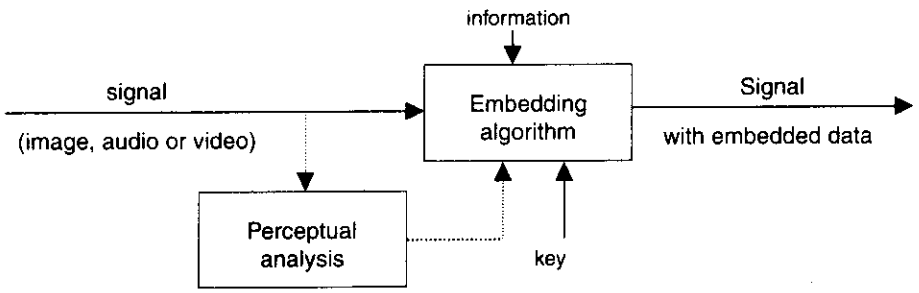

Figure 3.20 Block diagram of a data-embedding algorithm [3.146]. ©1998 IEEE.

3.12.1 Watermarking Techniques

Different watermarking techniques have been proposed by various authors in the last few years. The proposed algorithms can be classified into two main classes on the basis of the use of the original image during the detection phase: the algorithms that do not require the original image (blind scheme) [3.147, 3.148, 3.149] and the algorithms where the original image is the input in the detection algorithms along with the watermarked image (nonblind scheme) [3.150, 3.151, 3.152, 3.153]. Detectors of the second type have the advantage of detecting the watermarks in images that have been extensively modified in various ways.

Watermarking embedding can be done either in the spatial domain or in an appropriate transform domain, like a DCT domain [3.148, 3.152, 3.153], a wavelet transform domain [3.150, 3.151] or a Fourier transform domain [3.154]. In certain algorithms, the imposed changes take into account the local image characteristics and the properties of the human visual system (perceptual masking) in order to obtain watermarks that are guaranteed to be invisible [3.148, 3.150, 3.155].

The DCT-based watermarking method has been developed for image watermarking that could survive several kinds of image processings and lossy compression [3.156]. In order to extend the watermarking techniques into video sequences, the concept of temporal prediction exploited in MPEG is considered. For intraframe, the same techniques of image watermarking are applied, but for non-intraframe, the residual mask, which is used in image watermarking to obtain the spatially neighboring relationship, is extended into the temporal domain according to the type of predictive coding. In considering the JPEG-like coding technique, a DCT-based watermarking method is developed to provide an invisible watermark and also to survive the lossy compression.

The human eyes are more sensitive to noise in a lower frequency range than its higher frequency counterpart, but the energy of most natural images is concentrated in the lower frequency range. The quantization applied in lossy compression reflects the human visual system, which is less sensitive to quantization noise at higher frequencies. Therefore, to embed the watermark invisibly and to survive the lossy data compression, a reasonable trade-off is to embed the watermark into the middle-frequency range of the image. To prevent an expert from extracting the hidden information directly from the transform domain, the watermarks are embedded by modifying the relationship of the neighboring blocks of midfrequency coefficients of the original image instead of embedding by an additive operation.

Example 3.5 The original image is divided into 8x8 blocks of pixels, and the 2D DCT is applied independently to each block. Then, the coefficients of the midfrequency range from the DCT coefficients are selected. A 2D subblock mask is used in order to compute the residual pattern from the chosen midfrequency coefficients.

Let the digital watermark be a binary image. A fast 2D pseudorandom number-traversing method is used to permute the watermark so as to disperse its spatial relationship. In addition to the pixel-based permutation, a block-based permutation according to the variances of both the image and watermark is also used. Although the watermark is embedded into the mid-frequency

coefficients, for those blocks with little variances, the modification of DCT coefficients introduces quite visible artifacts. In this image-dependent permutation, both variances of the image blocks and watermark blocks are sorted and mapped according to importance of the invisibility. After the residual pattern is obtained for each marked pixel of the permuted watermark, the DCT coefficients are modified according to the residual mask, so that the corresponding polarity of residual value is reversed. Finally, inverse DCT of the associated results is applied to obtain the watermarked image.

Example 3.6 The extraction of a watermark requires the original image, watermarked image and also the digital watermark. At first, both the original image and the watermarked images are DCT transformed. Then, we make use of the chosen midfrequency coefficients and the residual mask to obtain the residual values. Perform the EXCLUSIVE-OR operation on these two residual patterns to obtain a permuted binary signal. Reverse both the block and the pixel-based permutations to get the extracted watermark.

A video sequence is divided into a series of Group of Pictures (GOP). Each GOP contains an interframe (I-frame), forward-predicted frame (P-frame) and bidirectional predicted/interpolated frame (B-frame). P-frame is encoded relative to intraframe or another P-frame. B-frame is derived from two other frames, one before and one after. These non-intraframes are derived from other reference frames by motion-compensation that uses the estimated motion vectors to construct the images. In order to insert the watermark into such kind of motion-compensated images, the residual patterns of neighboring blocks are extended into the temporal domain and other parts of the image. Watermarking techniques can be applied directly into non-intraframes.

For a forward-predicted P-frame, the residual mask is designed between the P-frame and its reference I- or P-frame, that is, the watermarks are embedded by modifying the temporal relationship between the current P-frame and its reference frame. For a bidirectionally predicted or interpolated B-frame, the residual mask is designed between the current B-frame and its past and future reference frames. The polarity of the residual pattern is reversed to embed the watermark.

3.12.2 Main Features of Watermarking

Watermarks are digital signals that are superimposed on a digital image causing alternations to the original data. A particular watermark belongs exclusively to one owner who is the only person that can proceed to a trustworthily detection of the personal watermark and, thus, prove the ownership of the watermark from the digital data. Watermarks should possess the following features [3.157]:

- ***Perceptual invisibility***—The modification caused by the watermark embedding should not degrade the perceived image quality. However, even hardly visible differences may become apparent when the original image is directly compared to the watermarked one.
- ***Trustworthily detection***—Watermarks should constitute a sufficient and trustworthily part of ownership of a particular product. Detection of a false alarm should be

extremely rare. Watermark signals should be characterized by great complexity. This is necessary in order to be able to produce an extensive set of sufficiently well distinguishable watermarks. An enormous set of watermarks prevents the recovery of a particular watermark by trial-and-error procedure.

- ***Associated key***—Watermarks should be associated with an identification number called watermark key. The key is used to cast, detect and remove a watermark. Subsequently, the key should be private and should exclusively characterize the legal owner. Any digital signal, extracted from a digital image, is assumed to be a valid watermark if and only if it is associated to a key using a well established algorithm.
- ***Automated detection/search***—Watermarks should combine easily with a search procedure that scans any publicly accessible domain in a network environment for illegal deposition of an owner's product.
- ***Statistical invisibility***—Watermarks should not be recovered using statistical methods. For example, the possession of a great number of digital products, watermarked with the same key, should not disclose the watermark by applying statistical methods. Therefore, watermarks should be image dependent.
- ***Multiple watermarkings***—We should be able to embed a sufficient number of different watermarks in the same images. This feature seems necessary because we cannot prevent someone from watermarking an already watermarked image. It is also convenient when the copyright property is transferred from one owner to another.
- ***Robustness***—A watermark that is of some practical use should be robust to image modifications up to a certain degree. The most common image manipulations are compression, filtering, color quantization/color-brightness modifications, geometric distortions and format change. A digital image can undergo a great deal of different modifications that may deliberately affect the embedded watermark. Obviously, a watermark that is to be used as a means of copyright protection should be detectable up to the point that the host image quality remains within acceptable limits.

Example 3.7 A visible watermark must be obvious to any person with normal or corrected vision (including the color blind), be flexible enough that it can be made as obstructive or unobstructive as desired, have bold features that form a recognizable image, allow the features of the unmarked image to appear in the marked image, and be very difficult, if not impossible, to remove.

To fulfill these criteria, we begin with the construction of a mask corresponding to the watermark. The mask determines which pixels in an image will remain unchanged and which will have their intensity altered. The mask is then resized, and the masking purpose and the location at which the watermark will be placed are chosen. Last, the intensity in the pixels specified by the mask is altered. We use a mathematical model of the intensity in an image

$$\hat{Y}_{m,n} = Y_{m,n} + C \times \Delta L \tag{3.24}$$

where $Y_{m,n}$ and $\hat{Y}_{m,n}$ represent the intensity of the (m,n)th pixel in the original and marked images, respectively, the constant C is a function that reflects various properties of the specific image and watermark mask, and L is the intensity, that is, the amount of the light received by the eye, regardless of color [3.158]. The appearance of the watermark is controlled by varying the intensity L by ΔL. If the same value of ΔL were used to alter all the pixels that fall under the mask, the watermark could be easily removed by a hostile party. To render robustness to the mask, randomness is introduced by using $2R_{m,n}\,\Delta L$ in place of ΔL, where $R_{m,n} \in [0,1]$ is a discrete random variable that satisfies

$$\lim_{M\to\infty}\lim_{N\to\infty}\frac{2}{M\,N}\sum_{m=1}^{M}\sum_{n=1}^{N}R_{m,n}\,\Delta L = \Delta L$$

(if ΔL is truly randomly distributed).

A watermark needs to have bold features because the introduction of the random variable $R_{m,n}$, depending on its values, can make fine details of the mark less discernible.

3.12.3 Application Domains

Because a generic watermarking algorithm cannot fit for a variety of applications, there is need for standardization of watermarking technology on an application-by-application basis. Table 3.4 [3.157] shows six different application domains and their requirements in terms of robustness and resilience:

Table 3.4 Categorization of applications by different robustness and resilience and the need of detection in every decoder [3.157].

Application Domains	Unintentional Attacks		Intentional Attacks			Every Decoder	High Capacity	Application Examples
	AT1	AT2	AT3	AT4	AT5			
A1.	Yes	Yes	Maybe	No	No	Yes	Yes	Value-added metadata
A2.	Yes	Yes	Yes	Yes	Yes	Yes	No	Copy protection
A3.	Yes	Yes	Yes	Yes	Yes	No	No	Ownership/fingerprint
A4.	Yes	No	No	No	Some	Yes	No	Authentication
A5.	Yes	Yes	No	No	Yes	Yes	Yes	Broadcasting
A6.	Yes	Yes	Maybe	Maybe	Yes	No	Yes	Secret communication

- **A1. Carrying value-added metadata**—This kind of application uses an embedded watermark to carry some additional information, for example, hyperlinks, annotation and content-based indexing information, which is to survive common content-preserving transformations. The watermarks should be detectable when the content is used not only in MPEG-4, but also in MPEG-1 or MPEG-2 if possible. In general, these applications often require higher capacity watermarks compared to other applications. On the other hand, in many cases, such watermarks are not subject to purely malicious attacks.
- **A2. Copy protection and conditional access**—In this kind of application, an embedded Intellectual Property Rights (IPR) system uses an embedded watermark to control Intellectual Property Management and Protection (IPMP)-related issues, for example, view options and copy options, in compliant decoders (for example, in the second generation of DVD). Every compliant decoder must be able to trigger protection or royalty collection mechanisms at the time when the contents are decoded. In general, such systems are predicated on the fact that unauthorized people should not be able to forge, remove or invalidate the watermarks by any means.
- **A3. Ownership assertion, recipient tracing**—In this kind of application, content owners use embedded watermarks to establish ownership or to determine the origin of unauthorized duplication. Normally, consumers do not need to know the original ownership when playing back the content. This kind of information is thus required only when prosecuting for IPR infringement. Again, such systems' effectiveness depends on the inability of unauthorized users to counterfeit, remove or invalidate watermarks.
- **A4. Authentication and verification**—Applications can use fragile or semi-fragile (for example, robust authentication) watermarks for verification of authenticity of source and content, verification of integrity of content and so forth. If contents are altered, watermarks should disappear or change to indicate the extent and possibly location of changes, depending on the precise scheme. In general, the viewer would like to know the verification results when playing back the contents. Although concerns of people removing or invalidating watermarks are less rampant in such applications, preventing forgery and counterfeiting becomes a particularly important issue to address.
- **A5. Broadcast monitoring**—We embed watermarks into contents and monitor where and when the contents are played. For example, advertisers may watermark their commercial. Then, upon verifying transmission of the commercial, the advertisers will pay a certain amount of money to whomever the advertisement is distributed. Every broadcast monitoring decoder that receives the commercial should report the usage. In general, detecting watermarks from heavily degraded content is less of an issue in such applications. However, depending on the precise application, watermark removal, invalidation and/or forgery can be significant concerns. In the broadcast-monitoring case, for example, counterfeiting should be intractable for the system to be effective.
- **A6. Secret communication or steganography**—This application, that is, data hiding in its most general sense, uses an embedded watermark to carry some secret information.

Often, these applications may need higher capacity watermarks than many other applications. Maintaining secrecy of the message so that unauthorized users cannot extract the watermark may often be an overriding concern in these kinds of applications. Removal and counterfeiting may or may not be an issue depending on the application.

The watermarking application environment is similar to that of compression. The intended information recipients (users of decompression or watermark extraction) need to know which method is used by the information provider (compression or watermark insertion). For those application domains that require every decoder to detect the watermarks, there is a need for standardization (A1, A2, A4 and A5). For those application domains that do not require every decoder to detect the watermarks, there is no rush for standardization (A3 and A6).

The watermark robustness testing conditions are defined as follows:

- AT1 Basic attacks—Lossy compression, frame dropping and temporal rescaling, that is, frame-rate changes
- AT2 Simple attacks—Blurring, median filtering, noise addition, gamma correction, and sharpening
- AT3 Normal attacks—Translation, cropping and scaling
- AT4 Enhanced attacks—Aspect ratio change and random geometric perturbations, for example, Stirmark, and local permutation of pixels
- AT5 Advanced attacks—Delete/insert watermarks, single-document watermark estimation attacks and multiple-document statistical attacks

3.13 Organization, Storage and Retrieval Issues

Once multimedia material is compressed and coded, it needs to be sent across a network to one or more end-users. To ensure that communication does not break down in transmission, we need to ensure the method and speed of delivery of the material using either a streaming implementation or a full download. We have to provide mechanisms for different resolutions of the receiving system and a guaranteed QoS so that the received multimedia signal has essentially the quality that is expected and is being paid for [3.121].

3.13.1 Streaming Issues for Speech and Audio

Streaming refers to the transmission of multimedia signals for real-time delivery without waiting for the entire file to arrive at the user terminal. Streaming can be either narrowcast (from the server to just one client) or broadcast (one transmission stream to multiple clients). The key point is that the real-time information is flowing solely from one source in both cases. There are four main elements to a streaming system:

- The compressed (coded) information content, for example, audio, video, speech, multimedia, data, and so forth

- The content, which is stored on a server
- The server, which is connected to the Internet and/or possibly other networks (POTS, ISDN, ATM or frame relay)
- The clients

Each of these elements can cause impairments. There are well-established methods for minimizing the degradations to the signal that result from these impairments. To set the scenario, assume that a user has requested a real-time audio or video stream from the Internet. Further, we assume that the client is not directly connected to the stream on the Internet, but instead accesses it through an Internet Service Provider (ISP). The access could be through a modem on a POTS line, ISDN, a corporate LAN running on IP or could even include ATM or frame relay in the access link. Although the backbone of the Internet is a high-speed data network, there are a number of potential bottlenecks within the path from the server that streams the data to the final client that receives the real-time stream. Heavy traffic causes congestion and results in variable delays and possibly dropping packets. Thus, the two manifestations of network congestion on the packet stream that represents the real-time signal are highly variable delays and lost packets. The degree to which these two problems occur determines the QoS that can be provided. The solutions to these problems must focus on these two issues: delayed and lost packets [3.121].

One potential way of addressing these problems is through the compression scheme. Using compression to represent the real-time signal provides an excellent mechanism for dealing with lost packets. Because the signal has been analyzed and reduced to its component parts as part of the compression scheme, this often makes handling lost and delayed packets practical because parts of the lost signal are often highly predictable. Although the coding algorithms attempt to remove all redundancy in the signal, some redundancy still remains, even for the most advanced coders. We can also use concepts of statistical predictability to extrapolate some of the component parts for missing packets. Hence, for both speech and audio, the best extrapolation policy for handling lost and delayed packets seems to be to assume stationarity of the missing signal. The degree to which this assumption is valid makes the strategy more or less viable in practice. If the signal is in a transitional state, holding its statistical properties constant will cause an obvious and highly perceptual degradation. However, shutting of the signal (playing silence in place of the signal) would lead to an even more obvious degradation most of the time. There is also the problem that lost packets cause the decoder state to lose synchronization with the encoder state. Generally, forward adaptive coders can resynchronize the encoder and decoder faster than backward adaptive coders. Hence, forward-adaptive coders are preferred for a highly congested data network with streaming speech signals.

If the speech or audio was encoded or uncompressed, an analysis of the immediate past material could be used to determine the amount of prediction that could be applied during periods of lost or delayed packets. To the extent that this occurs, the QoS may become unacceptable.

A technique that can be used with speech and also to some extent with audio is to change the buffer size to match the network delay characteristics. This method can also be used to account for any long term differences in the clock rates of the server and the client.

The client's software needs must be adapted to the network so that it can respond to missing packets. Obviously, the client must take some action to deal with the variation in the time of receipt of packets due to the variable delay of packets. Most commonly, the delay is accounted for by creating a buffer at the client to smooth over the variations in delay. For information retrieval, this is generally acceptable. It means that the start of playback will be delayed in proportion to the size of the buffer, but, for audio distribution, the extra delay will probably not be noticed.

The server's function in a streaming transaction is to transmit the information (the coded speech or audio) at an average rate designed to maintain real-time decoding of the compressed material. If the server is working too slowly (that is, heavily overloaded), the real-time signal received by the client will have gaps caused by the decoder running out of bit stream to decode. If the server is transmitting it quickly (that is, under loaded conditions), the bit stream will build up in the decoder buffers, eventually overfilling them and causing a loss of signal because of the buffer overflow. The server must serve multiple clients and must respond to changes in the network due to variable traffic and congestion. Thus, if the server is requested to reduce the amount of traffic it is generating on the network, it will be expected to do so. If it has too many clients, it can also cause interruptions in the regular transmission of packets. In general, because it is serving many clients, the server will not transmit packets at regular intervals. Namely, there will necessarily be some jitter in its transmission instants. Additionally, it will be more efficient to transmit packets of larger size.

The transmitted streaming information is now sent across the Internet and into the user's network. The principal problem is congestion at a few nodes or across a few links. The congestion results in both variable delays and lost packets. If the rate of packet loss is too high, real-time streaming is just not viable. The extrapolation techniques may be adequate to cover up losses of a few percent, but will not suffice when the loss rate is within the focus of percent range. Similarly, if the delay variation is too great, the longest delayed packets will be treated as lost packets.

3.13.2 Streaming Issues for Video

Video has a much higher data rate than speech or audio, and in a great deal of the time video requires only one-way streaming (movies, shows, documentaries, video clips, and so forth) and can therefore tolerate long delays in the streaming network. A successful streaming application requires a well-designed system that takes into account each of these elements [3.159]. Streaming in data networks is implemented as part of the application-layer protocols of the transmission, that is, it uses User Datagram Protocol (UDP) and TCP at the transport layer. Because of the known shortcomings of TCP, most streaming implementations are based on the inherently unreliable UDP. Thus, whenever there is network congestion, packets are dropped. Also, because delays can be large (on the order of seconds) and often unpredictable on the Internet, some packets may arrive after their nominal presentation time, effectively turning them into lost

packets. The extent of the losses is a function of the network congestion, which is highly correlated with the time of day and the distance (in terms of the number of the routers) between the client and the multimedia source [3.160]. The practical techniques that have evolved for improving the performance of streaming-based real time signal delivery can be classified into four broad areas:

- Client-side buffer management, which determines how much data needs to be buffered both prior to the start of the streaming playback and during the playback. It also determines a strategy for changing the buffer size as a function of the network congestion and delay and the load on the media server.
- Error-resilient transmission techniques, which increase client-side resilience to packet losses through intelligent transport techniques, such as using higher priority for transmitting more important parts (headers, and so forth) of a stream and/or establishing appropriate retransmission mechanisms where possible.
- Error-resilient coding techniques, which use source and perhaps combined source and channel-coding techniques that have built-in resilience to packet losses.
- Media control mechanisms, which use efficient implementations of Video Cassette Recorder (VCR)-type controls when serving multiple clients.

None of these techniques is sufficient to guarantee high-quality streaming, but, in combination, they serve to reduce the problems to manageable levels for most practical systems.

3.14 Signal Processing for Networked Multimedia

Real-time transmission of multimedia data across packet networks poses several interesting problems for signal-processing research. Although the range of these problems covers a large variety of topics, two groups attract the most attention. The first group concerns adapting the signal-compression techniques to address the special requirements imposed by the packet networks, including accommodating for packet losses, delays and jitter; providing capability for multipoint and coping with the heterogeneous nature of today's networks. The second group of problems is related to protecting the IPR associated with the transmitted multimedia data. The increasing availability of high bandwidth networking makes it extremely easy to duplicate and disseminate digital information illegally. Unless mechanisms can be established to protect the rights of the content providers, commercial use of networked multimedia will remain extremely limited.

Adapting signal compression techniques to networked applications may require some changes in the fundamental approach to this problem. The goal of classical signal compression is to achieve the highest possible compression ratio. The compression and transmission aspects have generally been treated as separate issues. The first problem with this approach is that the resulting compression algorithms usually do not address the needs of networked transmission.

Example 3.8 In a networked multimedia multicast, several receivers may be connected to the network with bandwidths that may range from very low, for example, a 28.8 Kbs modem, to very high, for example, a 150 Mbps optical link. Using a compression rate that satisfies the high bandwidth, receivers will cut off the low bandwidth ones completely.

On the other hand, using the lowest bandwidth will not be acceptable for the high bandwidth receivers. The alternate approach, simulcasting several data rates, requires about a two-fold increase in the main network bandwidth. A compression algorithm designed with this application requirement in mind must provide for easy extraction of data streams having different rates from a single compressed stream. Such compression algorithms are also useful for adapting to the changing network conditions caused by congestion [3.161, 3.162]. The second difficulty in separately designing the compression and transmission components is caused by the fact that a successful compression algorithm removes all the redundancy. Hence, the compressed data must be delivered error free. As an example, if a single bit of a compressed picture is lost, the entire picture may become undecodable. Effective error-concealment techniques must be present in the receiver to minimize the visual impact of any errors. A straightforward approach to provide robustness is to insert pointers into the compressed data to make the partially received data usable [3.163]. This approach works when the compressed data have natural boundaries (blocks in an image) and its error resilience increases as the size of the independently decodable data segments gets smaller. Because the added pointers (restart markers in JPEG images and slice start codes in MPEG video) introduce overhead, using smaller segments reduces the compression efficiency. A compression method designed to generate an output with independently decodable segments may perform better than this.

Another consideration in designing compression techniques for network use is to identify the impact of losing different portions of a compressed stream. For example, losing a header that contains the quantization information may render a large segment of data useless, but the same size loss in the data portion may only destroy a short segment that could be concealed at the receiver. Considering the relative ease of providing error-free transmission for shorter data segments, it is preferable to have the important parts of a compressed stream concentrated into a short and identifiable segment.

3.15 NNs for Multimedia Processing

Future multimedia technologies will need to handle information with an increasing level of intelligence, that is, automatic recognition and interpretation of multimodal signals. The main attribute of neural processing is its adaptive learning capability, which enables machines to be taught to interpret possible variations of some object or pattern, for example, scale, orientation and perspective [3.163, 3.164]. Moreover, we are able to approximate accurately unknown systems based on sparse sets of noisy data. Some neural models have also effectively incorporated statistical signal processing like expectation-maximization, Gaussian mixture and optimization techniques. In addition, spatial/temporal neural structures and hierarchical models are promising for multirate, multiresolution multimedia processing. NNs have thus received increasing attention in many multimedia applications, such as the following:

- Human perception—Facial expression and emotional categorization, human color perception and multimedia data visualization [3.165, 3.166, 3.167, 3.168]
- Computer-human communication—Face recognition, lip-reading analysis and human-human and computer-human communications
- Multimodal representation and information retrieval—Hyperlinking of multimedia objects, queries and search of multimedia information; 3D object representation and motion tracking, image sequence generation and animation [3.169, 3.170, 3.171, 3.172, 3.173, 3.174, 3.175, 3.176].

A major impact may be achieved by integrating adaptive neural processing into the state-of-the-art multimedia technologies. A complete multimedia system consists of many information processing stages, for which neural processing offers an efficient and unified core technologies.

Neural processing and Intelligent Multimedia Processing (IMP) share the following characteristics:

- A universal data-processing engine for multimodal signals
- Multimodality: multiple sensor/data sources
- Unsupervised clustering and/or supervised learning by example mechanisms

Future IMP applications include speech recognition/understanding, character recognition, texture classification, image/video segmentation, face-object detection/recognition, tracking 3D objects and analysis of facial expressions and gestures.

3.15.1 NNs for Optimal Visualization

For some image-processing applications (medical), a display that maximizes diagnostic information would be very desirable. NNs have been successfully applied for optimal visualization so that information can be more noticeably displayed. Note that raw data may contain more bits than what can be displayed in an ordinary computer monitor.

Example 3.9 A magnetic resonance image contains 12-bit data, but most monitors only have an 8-bit display. To map 12-bit data to an 8-bit display, the appearance of the image depends upon a proper selection of window width/center, which is a typical representation of image dynamic range in the medical field. An NN-based system is used to estimate window width/center parameters for optimal display.

To reduce the input dimension of NN, a feature vector of an input image is first extracted through PCA transformations. Then a competitive layer (unsupervised) NN is applied to label the feature vector into several possible classes with their confidence measure. For each class, both nonlinear and linear adaptive estimators are used to best calibrate window width/center. A nonlinear estimator is vulnerable to drastic and very unreasonable failures, but is very efficient in reaching local optimum. To alleviate such concern, a safety net is provided through a linear estimator.

A final data fusion scheme outputs the optimal window/center parameters by combining the results from all possible classes with appropriate weighting of the confidence measures [3.68].

3.15.2 Neural Techniques for Motion Estimation

Neural techniques for motion estimation have been under investigation. A motion estimation algorithm based on the Expectation Maximization (EM) technique was proposed by Fan, Nomazi and Penafiel [3.177]. First, the motion field is represented by a model characterized by a series of motion coefficients. Smoothness of motion is imposed on the assumption. Then the EM-based iterative algorithm is adopted to estimate the image motion coefficients from noisy measurements.

A feature true motion technique (TMT) for object-based motion tracking was proposed by Chen, Lin and King [3.178]. Based on a neighborhood relaxation neural model, it can effectively find true motion vectors of the prominent features of an object. By prominent feature, we mean the following:

- Any region of an object contains a good number of blocks that have motion vectors that exhibit certain consistency.
- Only true motion vectors for a few blocks per region are needed.

Therefore, at the outset, it would disqualify some reference blocks that are doomed unreliable to track. The method adopts a multicandidate prescreening to provide some robustness in selecting motion candidates. Furthermore, assuming that the true motion field is piecewise continuous, the method calculates the motion of a feature block after consulting all of its neighboring blocks' motions. This precaution allows a singular and erroneous motion vector to be corrected by its surrounding motion vectors, yielding an effect very much like median filtering. The tracker has also found useful application in motion-based video segmentation [3.177].

Example 3.10 One Foreman example is shown in Figure 3.21. Two frames of Foreman sequence are represented. Motion vectors found by the original full-search block-matching algorithm are shown also. Finally, we have motion vectors obtained by the neural method through neighborhood relaxation.

3.15.3 NN Application to Face Detection and Recognition

NNs have been recognized as an established and mature tool for many pattern-classification problems. Particularly, they have been successfully applied to face-recognition applications. By combining face information with other biometric features such as speech, feature fusion should not only enhance accuracy, but also provide some fault tolerance, that is, it could tolerate temporary failure of one of the bimodal channels. For many visual monitoring and surveillance applications, it is important to determine human eye positions from an image or an image sequence containing a human face. After the human eye positions are determined, all of other important facial features, such as positions of the nose and mouth, can easily be determined. The basic

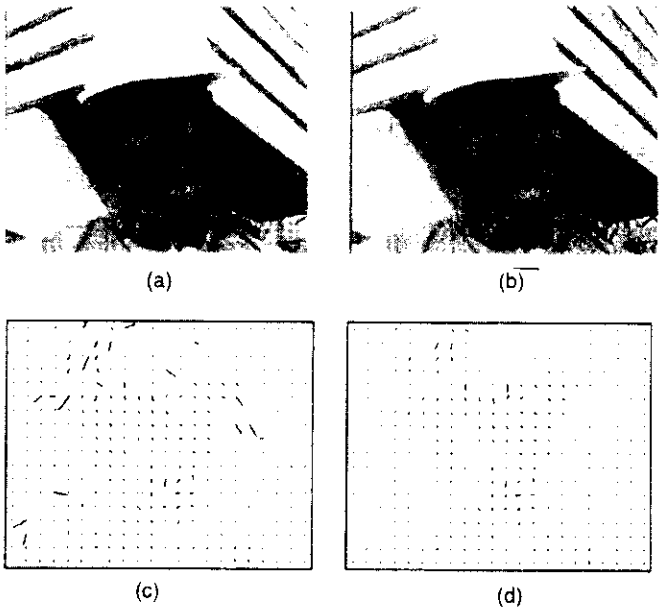

Figure 3.21 Two frames of the Foreman sequence (a, b). Motion vectors found by the original full-search block-matching algorithm (c). Motion vectors obtained by the neural method through neighborhood relaxation (d) [3.179]. ©1998 IEEE.

facial geometry information, such as the distance between eyes, nose and mouth size, can further be extracted. This geometry information can then be used for a variety of tasks, such as to recognize a face from a given face database.

There are many successful NN examples for face detection and recognition. Eigen-face subspace is used to determine the classes of the face patterns in [3.42]. Eigen-face and Fisher-face recognition algorithms were studied and compared in [3.68]. Cox et al. proposed mixture-distance VQ network for face recognition and reached a 95% success rate on a large database (685 persons) [3.180]. In Lin, Chan and King [3.181], NNs have been successfully applied to find such patterns with specific applications to detecting human faces and locating eyes in the faces.

3.15.4 Personal Authentication by Fusing Image and Speech

The fusion network has been applied to person recognition. The nonlinear fusion network proposed by Huang et al. [3.182] is based on the McCulloch-Pits NN where information from image and speech channels is combined as in Figure 3.22. The combined use of two channels yields a performance that is much improved over using either channel alone [3.182]. Noisy face images, each containing a 64x64, 8-bit grayscale image, serve as one source of information for classification. The images were decomposed into 13 channels by 4-level biorthogonal kernels. In

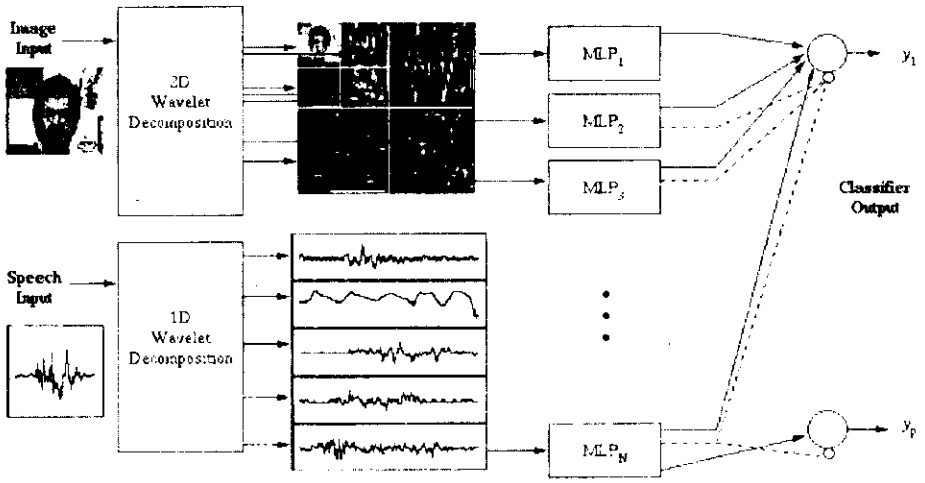

Figure 3.22 The nonlinear fusion network [3.182]. ©1997 IEEE.

speech channel, we have a noisy segment of speech, consisting of the spoken name of the same person. Speech segments digitized at 8 KHz were decomposed into eight channels using a length-eight wavelet kernel.

3.15.5 Subject-Based Retrieval for Image and Video Databases

A NN-based tagging algorithm was proposed for subject-based retrieval for image and video data bases [3.183]. Object classification for tagging is performed offline using Decision-Based NN (DBNN). A hierarchical multiresolution approach is used, which helps out the search space for looking for a feature in an image. The classification is performed in two phases. In the first phase, color is used, and in the second, texture features are applied to refine the classification, both using DBNN.

Compared to most of the other existing content-based retrieval systems, which only support similarity-based retrieval, the subject-based indexing system supports subject-based retrieval, allowing the system to retrieve by visual object. The difference between subject- and similarity-based retrieval lies in the necessity for visual object recognition. NNs provide a natural effective technology for intelligent information processing.

The tagging procedure includes four steps. In the first step, each image is cut into 25 equal-size blocks. Each block may contain single or multiple objects. In the second step, color information is employed for an initial classification, where each block is classified into one of the following families in the color space: black, gray, white, red, yellow, green, cyan, blue and magenta. In the next step, texture features are applied to refine the classification using DBNN if the result of color classification is a nonsingleton set of subject categories. Each block may be further classified into one of the following categories: sky, foliage, fleshtone, blacktop, white-object, ground, light, wood, unknown, and unsure. Last, an image tag generated from the lookup

table using the object-recognition results is saved in the tag database. The experimental results on the Web-based information show that this model is very efficient for a large film or television, program-oriented digital video database.

3.15.6 Face-Based Video Indexing and Browsing

A video indexing and browsing scheme based on human faces is proposed by two studies [3.181, 3.184]. The scheme is implemented by applying face detection and recognition techniques [3.185]. In many video applications, browsing through a large amount of video material to find the relevant clips is an important task. The video database indexed by human faces provides users the facility to acquire video clips about the person of interest, efficiently. A probabilistic DBNN face-based video browsing system is shown in Figure 3.23. A scene-change algorithm divides the video sequences into several shots. A face detector examines all the representative frames to see if they contain human faces. If they do, the face detector passes the frame to a face recognizer to find out whose face it is. The scheme contains three steps. The first step of our face-based video browser is to segment the video sequence by applying a scene-change-detection algorithm. Scene-change detection gives an introduction of when a new shot starts and ends. Each segment created by scene change detection can be considered as a story unit of this sequence. After video sequence segmentation, a probabilistic DBNN face detector is invoked to find the segments (shots) that might possibly contain human faces. From every video shot, we take its representative frame (Rframe) and feed it into face detector. Those representative frames from which the detector gives high face detection confidence scores are annotated and serve as the indexes for browsing. This scheme can be very helpful to algorithms for constructing hierarchies of video shots for video-browsing purposes.

3.16 Multimedia Processors

In the area of multimedia processors, the pipelining used in Reduced Instruction Set Computer (RISC) chips has been a key advance. These chips are used as host CPUs for PCs and workstations with operating systems. On the other hand, DSPs have achieved their high performance by incorporating hardware function units, such as multiply accumulators, Arithmetic and Logic Units (ALUs) and counters, which are controlled by parallel operations with moderate clock frequencies. These processors are used for speech compression for mobile phones, voice-band modems, and facsimile machines, as well as for the acceleration of sound and still-picture image processing on PCs.

Multimedia processing is the driving force in the evolution of both microprocessors and DSPs. The introduction of digital audio and video was the starting point of multimedia because it enabled audio and video as well as text, figures and tables to be used in a digital form in a computer and to be handled in the same manner. However, digital audio and video require a tremendous amount of information bandwidth unless compression techniques are used. Also, the amount of audio and video data for a given application is highly dependent on the required quality and can vary across a wide range. For example, HDTV (1920x1080 pixels with 60 fields/s) is

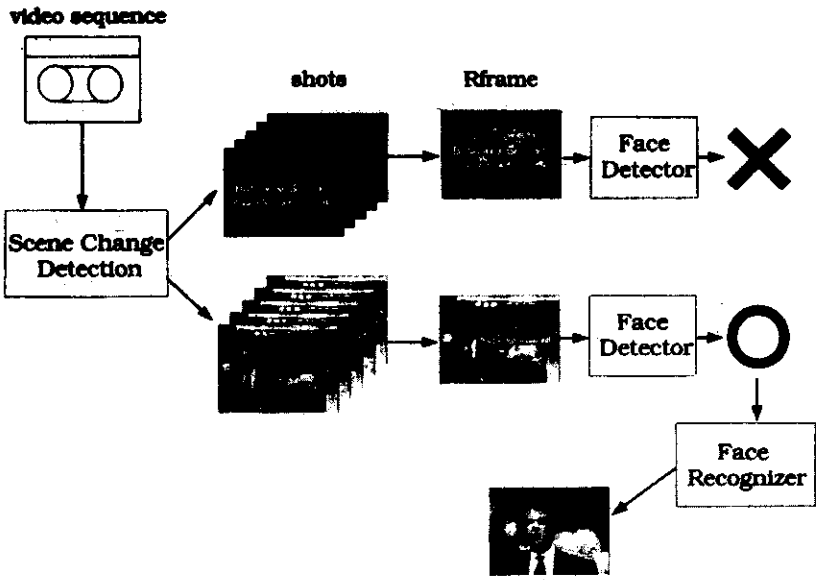

Figure 3.23 Probabilistic DBNN [3.181]. ©1996 IEEE.

expected to be compressed into around 20 Mb/s, but an H.263 video-phone terminal using sub-quarter-common intermediate format (128x96 pixels) with 7.5 frames/s is expected to be compressed into 1 to 20 Kb/s. Compression techniques call for a large amount of processing, but this also depends on the desired quality and information throughput. The required processing rate for compression ranges from 1,000 Mega Operations Per Second (MOPS) to more than 1 tera operations per second.

The wide variety of demands for processing multimedia has led to the software implementations of such compression techniques on microprocessors and DSPs to create an affordable multimedia environment. Microprocessors and programmable DSP chips offer powerful processing capabilities that enable real-time video and audio compression and decompression. Because a wide variety of applications is available to users, careful selection of these chips is essential to ensure flexibility in the independent application areas. This is because the chips' basic architectures differ significantly, and their respective advantages are highly related to their architectures.

3.16.1 Image-Processing Hardware and Software

We are now seeing PC systems that have the capability to acquire and process digital images as part of their display systems through the use of media coprocessors [3.186] and/or that are built into the instruction set of the microprocessor [3.187]. Another driving force in hardware and software is the fact that digital images and video are now part of many commercial and consumer applications.

In the early 1980s, Texas Instruments introduced the TMS320 digital signal processor for applications in speech and audio [3.188]. Others including AT&T/Lucent, Motorola, and Analog Devices have also developed DSP architectures. These DSPs have the capability of implementing real-time digital video processing. Other developments include Chromatic's Mpact [3.186] media processor, which is capable of accelerating several multimedia functions simultaneously [3.189].

The new microprocessors that are being developed have the capability of real-time image-processing operations. As clock speeds exceed 300 MHz, many real-time image-processing operations can now be performed on these processors without hardware acceleration. The most visible use is Intel's Multimedia Extension (MMX) processors [3.187, 3.190]. In MMX, 57 special instructions are added to the Intel processors that allow speedup in the execution of DSP operations.

The development of image-processing software has trailed behind the development of faster processors. Quite recently, few tools have been widely available for image-processing software designed for specific imaging applications. Software tools that have appeared in recent years include Mathematica [3.191], MATLAB's Image Processing Toolbox [3.192], Lab-View [3.193], and NIH Image [3.194].

The image and video-processing field will continue to benefit from the trends in computer technology. DSPs and microprocessors will be able to do real-time image processing. New software tools will be developed that will allow one to use the new microprocessors/PCs that will be arriving soon on everyone's desk.

3.16.2 Multimedia Processors' Classification

With the goal of improving multimedia-processing capabilities a wide variety of processors have been derived from microprocessors and DSPs. The multimedia processors can be classified in terms of their structure into the five categories as shown in Figure 3.24. These categories are

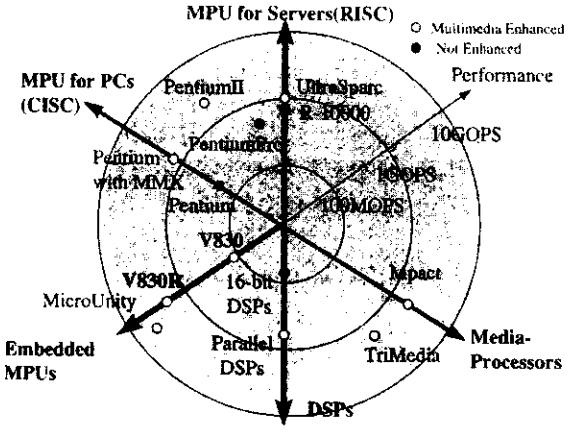

Figure 3.24 Multimedia Processor unit (MPU) classification [3.195]. ©1998 IEEE.

RISC microprocessors for workstations and servers, Complex Instruction Set Computer (CISC) microprocessors for PCs, embedded microprocessors, low power consumption DSPs and DSPs for PC acceleration, which are also called media processors [3.195]. This figure reflects certain trends in multimedia processor features and development. First, microprocessors for servers and PCs have been enhanced to handle multimedia processing while maintaining compatibility with their previous generations. To allow compatibility with the huge amount of existing software, there have been many restrictions on their enhancement. In other words, the heavy burden of hardware increase resulted. This is why these processors use very high clock frequencies and consume a lot of power.

New architectures with aggressive performance enhancements are being introduced for embedded microprocessors and DSPs, even though some of them have the functionality of a standalone CPU with high-level language support.

Multimedia processors target MPEG-2 decoding software. Thus, the performance and functionality for MPEG-2 video decoding are key issues in the design of the multimedia processor architectures. Decoding and playback of the compressed bitstream start with variable-length decoding, followed by inverse quantization to DCT coefficients from the compressed bitstream. Then an inverse DCT operation produces the prediction error signal. This signal retrieves the decoded frame with the addition of a motion-prediction signal. This signal is calculated by pixels' interpolation using one or two previously decoded frames (Figure 3.18). The decoded frame is transformed into a display format and transferred to the video Random Access Memory (RAM) and video output buffer. This transformation to display formats includes a Luminance Bandwidth Chrominance (YUV) to red-green-blue transform as well as dithering. This decompression process is carried out by a square image block called the macroblock (16x16 pixels with color components) or the block (8x8 pixels). Table 3.5 shows the major parameters for MPEG-2 MP@ML [3.196].

MPEG decoding for multimedia processors requires that the following five important functions be included in the system architectures:

- Bit-manipulation function, which parses and selects bit strings in serial bit streams. Variable-length encoding and decoding belong to this category.
- Arithmetic operations, which consist of multiplication, add/subtract, and other specific arithmetic operations, such as the sum of the absolute difference for motion estimation. Different word lengths are also desirable to improve hardware efficiency in handling many different media, such as video and audio data. Parallel processing units are also important for efficient Inverse DCT (IDCT) processing, which requires a lot of multiplications due to the nature of 2D IDCT algorithms.
- Memory access to a large memory space, which provides a video frame buffer that usually cannot reside in a processor on chip memory. The frequent access to the frame buffer for motion compensation requires a high-bandwidth memory interface.
- Stream data Input/Output (I/O) for media streams such as video and audio as well as compressed bitstreams. The I/O functionality is also needed for compressed bit streams

Table 3.5 MPEG-2 MP@ML parameters [3.196].

Parameter	MPEG-2 MP@ML
Horizontal size (pixels)	720
Vertical size (lines)	480
Frames/s	30
Display bits/s	15.55 M
Compressed bits/s	4-15 M
Number of macroblocks/s	40,500
Number of blocks/s	243,000

for storage media, such as hard disks; bit streams for storage media, such as hard disks and compact disks and for the communication networks.

- Real-time task switching, which supports hard real-time deadlines. This requires sample-by-sample and frame-by-frame time constraints. One example is switching between different types of simultaneous media processing to synchronize video and audio decoding.

In what follows, we discuss how the different processors are enhanced in terms of five key functions.

3.16.3 General Purpose Microprocessors

At present, high-end general-purpose microprocessors can issue two to four instructions per cycle by using superscalar control, which enables more than one floating-point instruction or several multimedia instructions to be issued at one time [3.197]. This control mechanism has two types of issuing mechanisms. One is the in-order-issue control, which issues instructions in the order that they are stored in the program memory. The other is the out-of-order issue control, where the issue order depends on the data priority rather than the storage order. This is effective for microprocessors that operate above 200 MHz and have long pipeline latency instructions where out-of-order control can maximize the high-speed pipelined ALU performance.

An example of an out-of-order superscalar microprocessor is shown in Figure 3.25. Implementation of out-of-order control requires several additional hardware functional units, such as a reorder buffer that controls the instruction issue and completion and a reservation station that reorders the actual instruction issues for the execution units and renamed register files, as well as

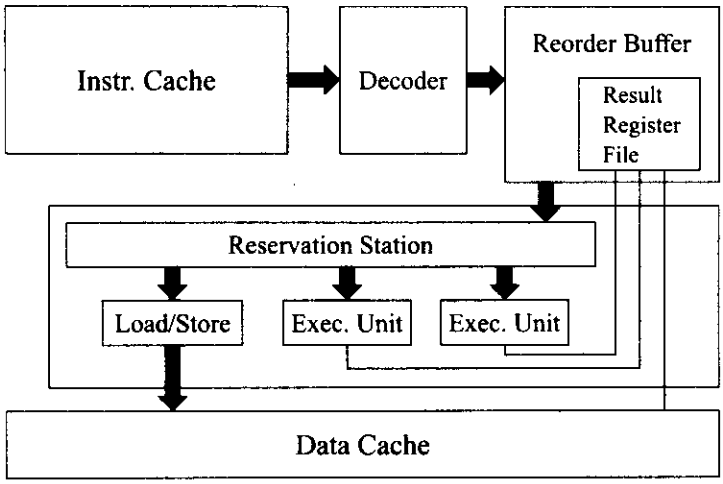

Figure 3.25 Superscalar microprocessor [3.195]. ©1998 IEEE.

control circuits for these units. These components take up a large part of the silicon area and contribute to the power dissipation.

In out-of-order control, the number of registers can actually be increased by register renaming. This improves the processing performance by reducing the number of load and store operations of intermediate calculation results to the memory as well as by reducing the processor stall cycles due to data dependencies. On the other hand, image processing has inherent parallelism in pixels and macroblocks. Although a large amount of hardware must be implemented for superscalar control, the issue of two to four parallel instructions does not fully take advantage of the parallelism in image processing. General-purpose microprocessor architecture related to media processing is illustrated in Figure 3.26. The arithmetic operations of microprocessors have a word-length problem. Microprocessor word lengths have been increased to 32 or 64 bits. The word lengths needed for multimedia processing are 8, 16 or 24 bits, which are much shorter than the word lengths of current state-of-the art microprocessors. This provides extravagant margins if we handle multimedia data with arithmetic instructions on microprocessors.

The memory access to a large memory space in microprocessors is a problem. Namely, in image processing, there is frequent access to large video frames that cannot reside in the first- or second-level cache. Therefore, we cannot expect the high-cache bit rate usually assumed in general-purpose applications. Moreover, a difference in the data locality affects the memory access performance. The cache mechanism is designed to use the 1D locality of consecutive addresses, but image processing has the 2D locality of access. In media processing for audio and video, processing of each audio sample or video frame should be completed within a predetermined sample or frame-interval time. This requires the predictability of the execution time, which is not easy to achieve in microprocessors. Data-dependent operations and memory access using cache mechanisms make it difficult to predict the processing cycles in the micro-

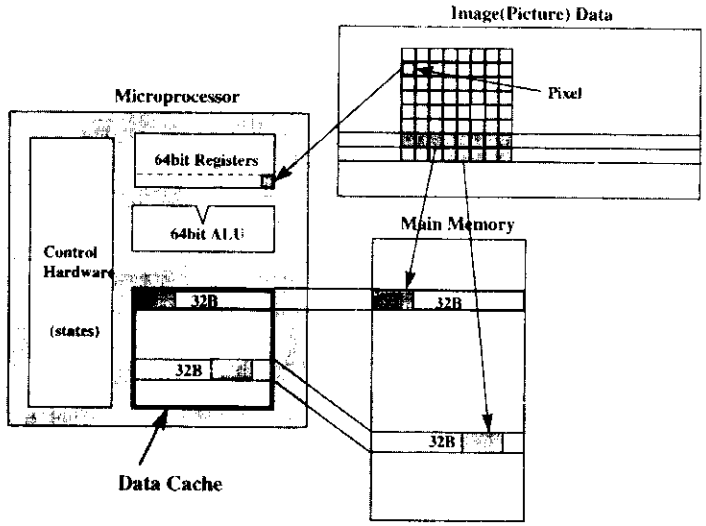

Figure 3.26 General-purpose microprocessor architecture [3.195]. ©1998 IEEE.

processors. Task switching with less overhead is also required. However, internal state introduced by out-of-order superscalar processors makes it difficult to achieve fast task switching.

The multimedia processing capability of recent microprocessors has been improved by multimedia extensions. These microprocessors enhance the arithmetic performance by dividing the long-word arithmetic unit (for example, 64 bits) to execute two to eight operations in parallel by using Single Instruction Multiple Data (SIMD)-type multimedia instructions as shown in Figure 3.27.

These instructions are implemented in either an integer data path or a floating point data path (coprocessor) in multiprocessors [3.197, 3.198, 3.199, 3.200]. The cache-miss penalties by off-chips main memory access are going to be a serious problem in microprocessor-based multimedia processing because the memory access latencies of microprocessors tend to be longer due to their higher clock frequencies. For most multimedia applications, the address for the memory access can be calculated beforehand.

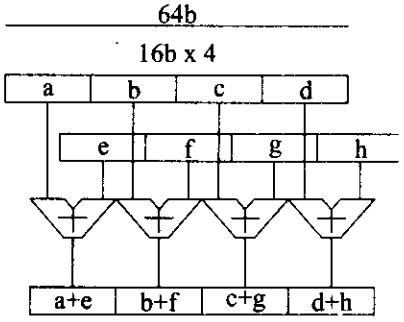

Figure 3.27 SIMD [3.195]. ©1998 IEEE.

3.16.4 Microprocessors for Embedded Applications

Target applications of this class of microprocessors include applications that originally used DSPs. In addition to these applications, multimedia applications, such as Internet terminals, set-top boxes, car navigations and personal digital assistants will be targets for these microprocessors.

A class of embedded RISC processors are inexpensive and consume little power. They do not employ complicated control mechanisms such as out-of-order controls. The arithmetic performance of embedded microprocessors can be enhanced by using a hardware multiply accumulator such as that shown in Figure 3.28. The requirements for real-time processing and the large

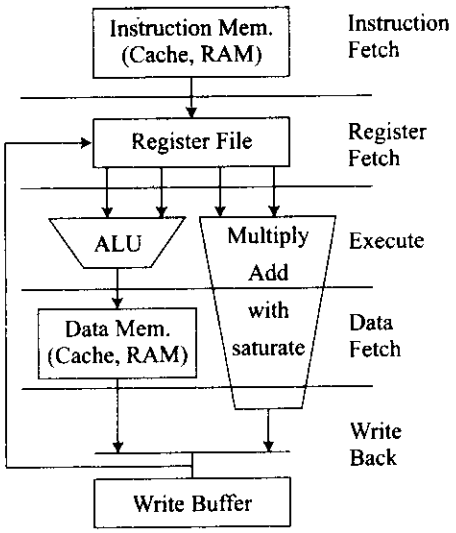

Figure 3.28 Data path for the arithmetic performance of embedded microprocessors [3.195]. ©1998 IEEE.

memory space are met by having both caches and buffers. Because microprocessors are normally used in low-cost systems that do not have a second-level cache, the cache-miss penalty is likely to be heavy. To present this, an internal RAM that is guaranteed not to cause a cache miss has been developed.

Another class of embedded microprocessors includes more sophisticated chips, such as the MicroUnit Mediaprocessor [3.201]. The processor has special memory interfaces as well as a stream I/O input/output interface with accompanying chips as shown in Figure 3.29. High-bandwidth memory access is realized by using special memory interfaces such as Synchronous Dynamic RAM (SDRAM) and ram bus dynamic random access memory [3.202]. The processor also has the capability for general-purpose microprocessors, such as virtual memory and memory management for standalone use. The arithmetic performance of the processor has been enhanced by using SIMD-type multimedia instructions with long word size. A large register file

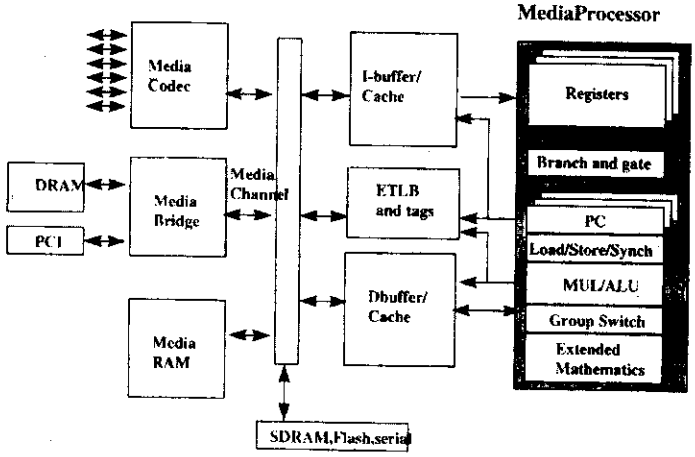

Figure 3.29 MicroUnit Media processor [3.195]. ©1998 IEEE.

(128x32 bits) also helps to improve the arithmetic performance of the processor. A memory-mapped I/O avoids coherence and latency problems in a Division Multiple Access (DMA)-based I/O system. Analog interfaces for audio and video are implemented on a separate chip [3.203]. With the advanced features described here and a higher clock frequency (1 GHz), this processor should be able to handle broadband media. This will result in higher power consumption.

3.17 Concluding Remarks

MMSP technologies play major roles in the multimedia network age. At this point, it is important to consider what the major issues are for MMSP technologies. Engineers have been concentrating their research on computer models that have functions in communicating with human beings.

The human communication mechanism can be explained by using a two-layer model. The first layer, which is the surface, enables communications based on language usage. This means that the logical information involved in human communications comes from this layer. However, this logical information is only a part of the whole information that constitutes speech. Other information, like information on emotions or senses, is also included.

The interaction layer is the second layer and is a deeper layer. It controls interactive behaviors in human communications. This layer also controls very basic behaviors like turning our face toward the direction of a sound or closing our eyes upon suddenly sensing strong light. These functions are considered to be important keys to humanlike behavior in daily life.

Algorithms for processing m-dimensional signals can be grouped into four categories:

- The separable algorithms that use 1D operators to process the rows and columns of a multidimensional array
- The nonseparable algorithms that borrow their derivation from their 1D counterpart

- The MD algorithms that are significantly different from their 1D counterparts
- The MD algorithms that have no 1D counterparts

Separable algorithms operate sequentially on the rows and columns of an MD signal. They have been widely used for image processing since the 1960s, because they require less computation than nonseparable algorithms. Examples include MD DFT, DCT, and FFT-based spatial estimation. In addition, separable FIR filters can be used in separable filter banks, wavelet representations for MD signals and decimations and interpolators for changing the sampling rate.

Algorithms that are MD and cannot be decomposed into a repetition of 1D procedures are uniquely MD in that they cannot be decomposed into a repetition of 1D procedures. They are a straightforward generalization of 1D techniques. They can usually be derived by repeating the corresponding 1D derivation in an MD setting. Sampling and downsampling are one example. As in the 1D case, band-limited MD signals can be sampled on periodic lattices with no loss of information. Most 1D filtering and FFT-based spectrum analysis algorithms generalize straightforwardly to any MD lattice. The window method for FIR filter design can be easily extended, and FFT algorithms can be decomposed into a vector-radix form, which is slightly more efficient than the separable row/column approach for evaluating MD DFTs.

The MD algorithms that have no 1D counterpart are algorithms that perform inversion and computer imaging. One of these is the operation of recovering an MD distribution from a finite set of its projections, equivalently inverting a discretized Radon transform. This is the mathematical basis of computer tomography. Another imaging method, developed first for geophysical applications, is Fourier migration. This is an efficient algorithm for image formation. Finally, signal recovery methods unlike the 1D case are possible. The MD signals with finite support can be recovered from the amplitudes of their Fourier transforms or from threshold crossing.

Adapting signal compression to networked applications may require some changes in the fundamental approach to this problem. The compression and transmission aspects have generally been treated as separate issues. The first problem with this approach is that the resulting compression algorithms usually do not address the needs of networked transmission. A successful compression algorithm removes all the redundancy, and, hence, the compressed data must be delivered error free. Another consideration in designing compression techniques for network use is to identify the impact of losing different portions of a compressed stream. It is preferable to have the important parts of the compressed stream concentrate into a short and identifiable segment.

Signal-processing techniques can be valuable for hiding a watermark (or identifying information) in the media. Watermarks can play a number of roles. First, a watermark can mark or identify the original owner of the content, such as the image creator. Second, it can identify the recipient of an authorized single-user copy. Third, a watermark can be used to identify when an image has been appreciably modified. An appropriate solution for the watermarking problem requires understanding of both the signal coding and networking or security issues.

Multimedia processors that realize multimedia processing through the use of software include those for bit manipulation, arithmetic operations, memory access, stream data I/O and

real-time switching. The programmable processors for multimedia processing are classified into media-enhanced microprocessors (CISC or RISC), embedded microprocessors, DSPs and media processors.

Many critical research topics remain yet to be solved. From the commercial system perspective, there are many promising application-driven research problems. These include analysis of multimodal scene-change detection, facial expressions and gestures, fusion of gesture/emotion and speech/audio signals; automatic captioning for the hearing impaired or second-language television audiences; multimedia telephone and interactive multimedia services for audio, speech, image and video contents.

From a long-term research perspective, there is a need to establish a fundamental and coherent theoretical ground for intelligent multimedia technologies. A powerful preprocessing technique capable of yielding salient object-based video representation would provide a healthy footing for online, object-oriented visual indexing. This suggests that a synergistic balance and interaction between representation and indexing must be carefully investigated. Another fundamental research subject needing our immediate attention is modeling and evaluation of perceptual quality in multimodal human communication. For a content-based visual query, incorporating user feedback in the interactive search process will be also a challenging but rewarding topic.