

Comparing Media Codecs for Video Content

Jeremiah Golston (Distinguished Member Technical Staff, Texas Instruments)

j-golston@ti.com

#250 at Embedded Systems Conference – San Francisco 2004

Introduction

Digital video is being adopted in an increasing range of applications including video telephony, security/surveillance, DVD, digital television, Internet video streaming, digital video camcorders, cellular media, and personal video recorders. Video compression is an essential enabler for these applications and an increasing number of video codec (compression/decompression) industry standards and proprietary algorithms are available to make it practical to store and transmit video in digital form. Compression standards are evolving to make use of advances in algorithms and take advantage of continued increases in available processing horsepower in low-cost integrated circuits such as digital media processors. Differences exist in the compression standards and within implementation of standards based on optimizations for the primary requirements of the target application. This paper provides an overview of the different compression standards and highlights where they are best suited. It also provides an overview of compression rules of thumb for different standards and the corresponding performance requirements for real-time implementations.

The Video Compression Challenge

A major challenge for digital video is that raw or uncompressed video requires lots of data to be stored or transmitted. For example, standard definition NTSC video is typically digitized at 720x480 using 4:2:2 YCrCb at 30 frames/second. This requires a data rate of over 165 Mbits/sec. To store one 90-minute video requires over 110 GBytes or approximately 140x the storage capability of a CDROM. Even lower resolution video such as CIF (352x288 4:2:0 at 30 frames/second) which is often used in video streaming applications requires over 36.5 Mbits/s - much more than can be sustained on even broadband networks such as ADSL. So, it is clear that compression is needed to store or transmit digital video.

The goal for image and video compression is to represent (or encode) a digital image or sequence of images in the case of video using as few bits as possible while maintaining its visual appearance. The techniques that have emerged are based on mathematical techniques but require making subtle tradeoffs that approach being an art form.

Compression Tradeoffs

There are many factors to consider in selecting the compression engine to use in a digital video system. The first thing to consider is the image quality requirements for the application and the format of both the source content and target display. Parameters include the desired resolution, color depth, the number of frames per second, and whether the content and/or display are progressive or interlaced.

Compression often involves tradeoffs between the image quality requirements and other needs of the application. For example, what is the maximum bit rate in terms of bits per second? How much storage capacity is available and what is the recording duration? For two-way video communication, what is the latency tolerance or allowable end-to-end system delay? The various compression standards handle these tradeoffs including the image resolution and target bit rate differently depending on the primary target application.

Another tradeoff is the cost of real-time implementation of the encoding and decoding. Typically newer algorithms achieving higher compression require increased processing which can impact the cost for encoding and decoding devices, system power dissipation, and total memory in the system.

Standards Bodies

There have been two primary standards organizations driving the definition of image and video compression standards. The International Telecommunications Union (ITU) is focused on telecommunication applications and has created the H.26x standards for video telephony. The Internal Standards Organization (ISO) is more focused on consumer applications and has defined the JPEG standards for still image compression and MPEG standards for compressing moving pictures.

The two groups often make slightly different tradeoffs based on their primary target applications. On occasions the two groups have worked together such as recent work by the JVT (or Joint Video Team) on a common standard referred to as both H.264 and MPEG-4 AVC. While almost all video standards were targeted for a few specific applications, they are often used to advantage in other kinds of applications when they are well suited

Standards have been critical for the widespread adoption of compression technology. The ITU and ISO have been instrumental in creating compression standards the marketplace can use to achieve interoperability. These groups also continue to evolve compression techniques and define new standards that deliver higher compression and enable new market opportunities.

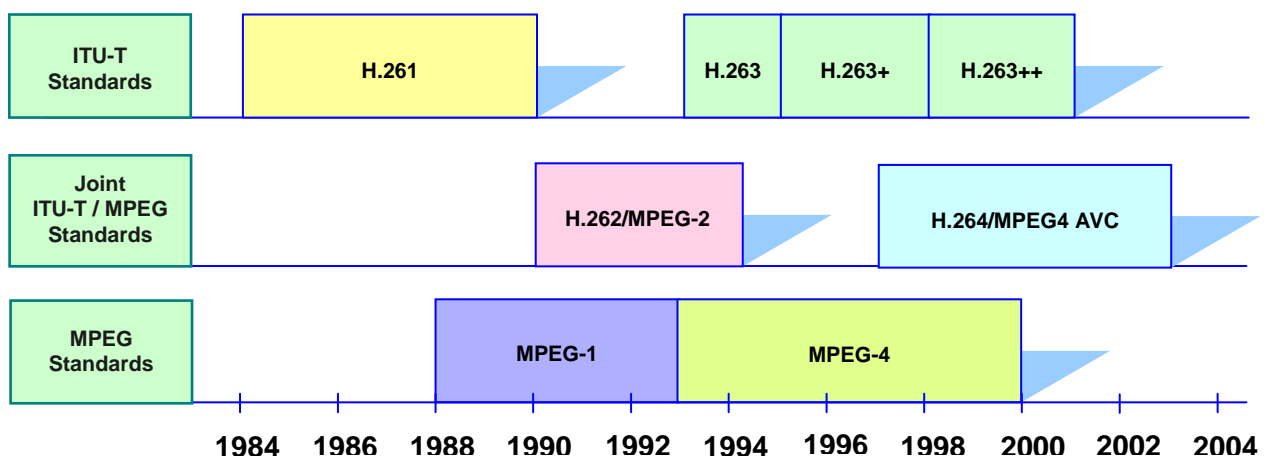


Figure 1. Progression of the ITU-T Recommendations and MPEG standards.

In addition to industry standards from the ITU and ISO, several popular proprietary solutions have emerged particularly for Internet streaming media applications. These include Real Networks Real Video (RV10)

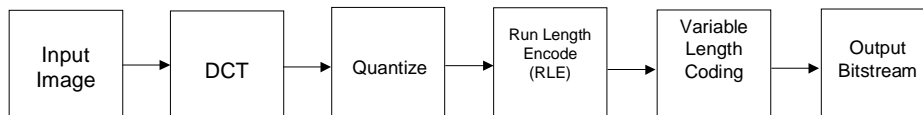
Microsoft Windows Media Video 9 Series, ON2 VP6, and Nancy among others. Because of the installed base of content in these formats, they can become de facto standards.

The number of standards and de facto standards is rapidly increasing creating an increasing need for flexible solutions for encoding and decoding. We'll step through some of the industry standard formats in a little more detail in the next few sections focusing on key features and target applications.

JPEG

JPEG developed by the ISO was the first widespread image compression standard [1]. It was designed to allow compression of digital images and is now widely used for Internet web pages and digital still cameras. The compression quality at a given rate is somewhat dependent on the image content such as the amount of detail or high frequency content in the image. However, using JPEG compression factors of 10:1 can typically be achieved without introducing serious effects from the compression. As you go above this 10 to 1 ratio, you can start to easily notice compression artifacts such as blockiness, contouring, and blurring of the image.

Although JPEG is not optimized for video, it has often been used for coding video with what is sometimes referred to as "Motion JPEG". Motion JPEG is not defined in the standard but typically consists of independently coding individual frames in a digital video sequence using JPEG. A common application for Motion JPEG is network video surveillance. Since each image is coded independently, it is easy to search through the content and also has benefits for interoperability with PC browsers.



Key Components

- DCT (translate spatial data to frequency domain)
- Quantization (scale the bit allocation for different frequencies generating many zero valued coefficients)
- Run-length code (RLE) non-zero coefficients
- Variable Length Coding (VLC)
 - Entropy code (e.g., Huffman) run-length codes

Figure 2: JPEG (Intra-frame) Compression Block Diagram

The main functions in the JPEG standard shown in Figure 2 formed the core for all of the major compression algorithms that followed. Key functions include the following:

Block-based Processing: Dividing each frame into blocks of pixels so that processing of the image or video frame can be conducted at the block level.

Intra-frame Coding: Exploiting the spatial redundancies that exist within the image or video frame by coding the original blocks through transform, quantization, and entropy coding. The frame is coded based on spatial redundancy only. There is no dependence on surrounding frames.

8x8 DCT: Each 8x8 block of pixel values is mapped to the frequency domain producing 64 frequency components

Perceptual Quantization: Scale the bit allocation for different frequencies typically generating many zero valued coefficients.

Run-length Coding: Represent the quantized frequency coefficients as a non-zero coefficient level followed by runs of zero coefficients and a final end of block code after the last non-zero value.

Variable Length (Huffman) Coding: Huffman coding converts the run-level pairs into variable length codes (VLCs) with the bit-length optimized for the typical probability distribution.

JPEG has extensions for lossless and progressive coding. Unlike most of the video compression standards, JPEG supports a variety of color spaces including RGB and YCrCb.

JPEG2000

JPEG2000 is a new still image coding standard from the ISO that was adopted in December 2000 [2]. It was targeted at many of the same applications as JPEG including high-quality digital still cameras, hard copy devices and Internet picture applications. The primary goals were to provide improved compression along with more seamless quality and resolution scalability.

JPEG2000 achieves key improvements in scalability of resolution and bitrate through use of several key functions that are not used by the JPEG, MPEG, and H.26x standards.

Discrete Wavelet Transform: The wavelet transform is used replacing the DCT to achieve higher compression and improve support for scalable transmission. Wavelets are new basis functions, unlike the usual cosines (DCT) and sines (FFT). They are called wavelets because they look like small waves. They have an excellent ability to represent both stationary as well as transient phenomena with few coefficients. Wavelets represent signals as a linear summation of shifted and translated versions of a basic wave. JPEG2000 is coded in frequency sub-bands using the wavelet transform to allow resolution scalability. The same bitstream can be decoded at different resolutions. Also, a thumbnail can be sent providing excellent quality at lower resolution and the resolution can be gradually increased as more sub-bands are received. This structure also helps improve error resilience for wireless and Internet applications.

Bit Plane Coding: The quantized sub-bands from the wavelet transform are divided into code blocks. Code blocks are entropy coded along bit planes using a combination of a bit plane coder and binary arithmetic coding. In JPEG2000, embedded block coding with optimized truncation (EBCOT) is used to implement bit plane coding. The algorithm uses symmetries and redundancies within and across the bit planes. The bit plane coding structure can be used to offer bitrate scalability since increasing detail can be added as more bit planes are decoded. Also, different quality bitstreams can be decoded at the same resolution, depending on the client's bandwidth without having to re-encode separately for each client.

Binary Arithmetic Coding: The bit plane coding outputs are entropy coded using binary arithmetic coding to generate the bitstream. Binary coding allows more flexibility than Huffman coding because symbols don't have to be represented by an integer number of bits. The JPEG2000 arithmetic coder uses predetermined probability values and the adaptation state machine is also supplied by the standard.

JPEG2000 can sustain much better quality than JPEG at high compression ratios because the wavelet transform degrades more gracefully. As the compression rate decreases, the gap narrows between JPEG and JPEG2000. For excellent quality, the wavelet transform yields about ~30% more compression (e.g., 13:1) than JPEG (10:1). The wavelet transform is more computationally intensive than the DCT but it is the really the bit-plane coding and binary arithmetic encoding functions that add most of the complexity to JPEG2000.

JPEG2000 includes both lossy and lossless compression modes. Motion JPEG2000 support is also being defined in the standard. One of the potential uses of motion JPEG2000 is for applications such as digital cinema requiring some compression but with the primary focus on highest video quality. JPEG2000 can support pixel depths greater than 8-bit such as 10 or 12-bits/pixel and is flexible in terms of the color space.

The widespread benefit of wavelets for low-bit rate video compression is still not clear. Motion estimation typically works best on small blocks. However, dividing images into small blocks degrades the wavelet performance. This makes it difficult to apply motion compensation using wavelets in the spatial domain. Meanwhile, wavelets are not shift invariant so shifted versions of the image result in a totally new representation. This creates problems for recognizing shifted versions of objects in images in transform domain. Since the transform is no longer block based, research is ongoing to find efficient ways to exploit motion in the hierarchical or sub-band domain. There have been some proprietary video codecs that use wavelets for I pictures, and DCT for P and B pictures. However resulting bit-rates are not comparable with today's advanced media codecs such as H.264.

H.261

H.261 defined by the ITU was the first major video compression standard [3]. It was targeted for video conferencing applications and was originally referred to as Px64 since it was designed for use with ISDN networks that supported multiples of 64 kbps. As seen in Figure 3, H.261 and the video compression standards that followed use similar core functions as JPEG such as block-based DCT but add features to explore the temporal redundancy or commonality from one image to the next in typical video content. For example, background areas often stay the same from one frame to the next and do not need to be retransmitted each frame. Video compression algorithms typically encode the differences between neighboring frames instead actual pixel values. Key features added in H.261 over JPEG include:

Inter-frame Coding: Coding uses both spatial redundancy and temporal redundancy to achieve higher compression.

P Frames: Frames are coded using data the previous decoded frame to predict the content in the new frame and exploiting any remaining spatial redundancies within the video frame by coding the residual blocks, i.e., the difference between the original blocks and the corresponding predicted blocks, using DCT, quantization, and entropy coding.

Motion Estimation: Used in the encoder to account for motion between the reference frame and the frame being coded to allow best possible prediction. This is usually the most performance intensive function in

video compression and is part of why video encoders typically require much more processing than the corresponding video decoder.

Motion Compensation: Process in the decoder of bringing in the predicted data from the reference frame accounting for the motion identified by the motion estimation in the encoder.

Fixed Quantization: Unlike JPEG and MPEG standards, H.261 and H.263 use fixed linear quantization across all the AC coefficients.

Loop Filtering: 2D separable 121 filter used to smooth out quantization effects in the reference frame. Must be applied in bit exact fashion by both the encoder and the decoder.

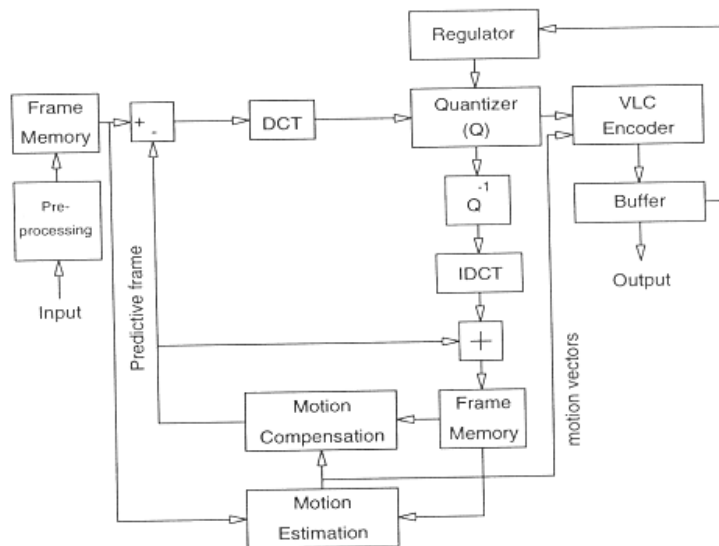


Figure 3: Inter-frame Video Compression Block Diagram

Due to its focus on two-way video, H.261 includes only techniques that do not introduce major added delay in the compression and decompression process. In general, the encoder is designed to avoid extra complexity since most applications require simultaneous real-time encoding and decoding.

H.263

H.263 was developed after H.261 with a focus on enabling better quality at even lower bitrates [4]. One of the major original targets was video over ordinary telephone modems that ran at 28.8 Kbps at the time. The target resolution was from SQCIF (128x96) to CIF. The basic algorithms are similar to H.261 but with the following new features:

Motion Estimation: Support for half-pel motion vectors and larger search range. Several annexes with optional features including 4 motion vectors, overlapped motion compensation, and unrestricted motion vectors.

3D VLC: Huffman coding which combines an end of block (EOB) indicator together with each Run Level pair. This feature is specifically targeted at low-bit rate where many times there are only one or two coded coefficients.

Adequate quality over ordinary phone lines proved to be very difficult and videophones over standard modems are still a challenge today. Because H.263 generally offered improved efficiency over H.261, it became used as the preferred algorithm for video conferencing with H.261 support still required for compatibility with older systems. H.263 grew over time as H.263+ and H.263++ added optional annexes supporting compression improvements and features for robustness over packet networks. For this reason, it has found some use in networked security applications as an alternative to motion JPEG. H.263 and its annexes formed the core for many of the coding tools in MPEG-4.

MPEG-1

MPEG-1 was the first video compression algorithm developed by the ISO [5]. The driving application was storage and retrieval of moving pictures and audio on digital media such as video CDs using SIF resolution (352x240) at 30 fps. The targeted output bitrate was 1.15 Mbps, which produces effectively 25:1 compression. MPEG-1 is similar to H.261 but encoders typically require more performance to support the heavier motion found in movie content versus typical video telephony.

Major new tools in MPEG-1 included:

B Frames: Individual macroblocks can be coded using forward, backward or bi-directional prediction as illustrated in Figure 4. An example of the benefit is the ability to match a background area that was occluded in the previous frame using forward prediction. Bi-directional prediction can allow for decreased noise by averaging both forward and backward prediction. Leveraging this feature in encoders requires additional processing since motion estimation has to be performed for both forward and backward prediction which can effectively double the motion estimation computational requirements. B frame tools require a more complex data flow since frames are decoded out of order with respect to how they are captured and need to be displayed. This feature results in increased latency and thus is not suitable for some applications. B frames are not used for prediction so tradeoffs can be made for some applications. For example, they can be skipped in low frame rate apps without impacting the decoding of future I and P frames.

Adaptive Perceptual Quantization: A quantization scale factor is applied specific to each frequency bin to optimize for human visual perception.

Broadcasters and content providers were generally not happy with the quality that could be achieved at the target bitrates for MPEG-1 and so an effort was started on a new standard that would support higher resolution video using higher bitrates.

- Intra (I) Frame Coding
 - Frame is coded based on spatial redundancy only
 - No dependence on surrounding frames
- P (Predicted) Frame Coding
 - Frame is coded using prediction from prior encoded I or P frame(s)
- B (Bi-directionally) Predicted Frame
 - Frame is coded with bi-directional (forward and backward) prediction
 - B frames are never used for prediction

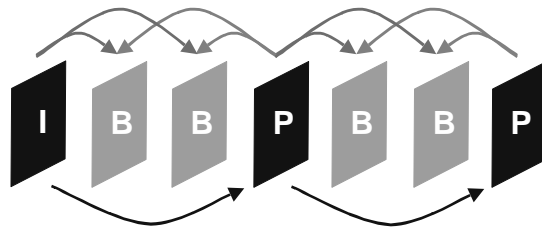


Figure 4: Frame Coding Types

MPEG-2

MPEG-2 was developed targeting digital television and soon became the most successful video compression algorithm so far [6]. It supports standard television resolutions including interlaced 720x480 at 60 fields per second for NTSC used in the US & Japan and interlaced 720x576 at 50 fields per second for PAL used in Europe.

MPEG-2 built on MPEG-1 with extensions to support interlaced video and also much wider motion compensation ranges. Encoders taking full advantage of the wider search range and the higher resolution require significantly more processing than H.261 & MPEG-1.

Interlaced Coding Tools: Includes ability to optimize the motion estimation supporting both field and frame based predictions and support for both field and frame based DCT/IDCT.

Wide Search Range: Due to the target for higher resolution video, MPEG-2 supports vastly wider search ranges than MPEG-1. This greatly increases the performance requirement for motion estimation versus the earlier standards.

MPEG-2 performs well at compression ratios around 30:1. The quality achieved with MPEG-2 at 4-8 Mbps was found to be acceptable for consumer video applications and it soon became deployed in applications including digital satellite, digital cable, DVDs, and now high-definition TV.

The processing requirements for MPEG-2 decoding were initially very high for general-purpose processors and even DSPs. Optimized fixed function MPEG-2 decoders were developed and have become inexpensive

over time due to the high volumes. Availability of cost-effective silicon solutions is a key ingredient for the success and deployment of video codec standards.

MPEG-4

MPEG-4 was initiated by the ISO as a follow-on to the success of MPEG-2 [7]. Some of the early objectives were increased error robustness supporting wireless networks, better support for low bitrate applications, and a variety of new tools to support merging graphic objects with video. Most of the graphics features have not gained significant traction yet in products and implementations have focused primarily on the improved low bitrate compression and error resiliency.

MPEG-4 simple profile (SP) starts from H.263 baseline and adds new tools for improved compression:

Unrestricted Motion Vectors: Supports prediction for objects when they partially move outside of the boundaries of the frame.

Variable Block Size Motion Compensation: Allows motion compensation at either 16x16 or 8x8 block granularity.

Intra DCT DC/AC Prediction: Allows the DC/AC coefficients to be predicted from neighboring blocks either to the left or above the current block.

Error resiliency features added to support recovery for packet loss include:

Slice Resynchronization: Establishes slices within images that allow quicker resynchronization after an error has occurred. The standard removes data dependencies between slices to allow error-free decoding at the start of the slice regardless of the error that occurred in the previous slice.

Data Partitioning: A mode that allows partitioning the data within a video packet into a motion part and DCT data part by separating these with a unique motion boundary marker. This allows more stringent checks on the validity of motion vector data. If an error occurs you can have better visibility at what point the error occurred to avoid discarding all the motion data when an error is found.

Reversible VLC: VLC code tables designed to allow decoding them backwards as well as forwards. When an error is encountered, it is possible to sync at the next slice or start code and work back to the point where the error occurred.

The MPEG-4 advanced simple profile (ASP) starts from the simple profile and adds B frames and interlaced tools similar to MPEG-2. It also adds quarter-pixel motion estimation and an option for global motion compensation. MPEG-4 advanced simple profile requires significantly more processing performance than the simple profile and has higher complexity and coding efficiency than MPEG-2.

MPEG-4 was used initially primarily in Internet streaming and became adopted for example by Apple's QuickTime player. MPEG-4 simple profile is now finding widespread applications in emerging cellular media phones. MPEG-4 ASP forms the foundation for a proprietary implementation called DivX that has become popular.

H.264/MPEG-4 AVC

A major breakthrough is now happening with the introduction of a new standard jointly promoted by the ITU and ISO [8,9]. H.264/MPEG-4 AVC delivers a significant break-through in compression efficiency generally achieving around 2x reduction in bit rate versus MPEG-2 and MPEG-4 simple profile. In formal tests conducted by the JVT, H.264 delivered a coding efficiency improvement of 1.5x or greater in 78% of the 85 testcases with 77% of those showing improvements 2x or greater and as high as 4x for some cases [10].

This new standard has been referred to by many different names as it evolved. The ITU began work on H.26L (for long term) in 1997 using major new coding tools. The results were impressive and the ISO decided to work with the ITU to adopt a common standard under a Joint Video Team. For this reason, you sometimes hear people refer to the standard as JVT even though this is not the formal name. The ITU approved the new H.264 standard in May 2003. The ISO approved the standard in October of 2003 as MPEG-4 Part 10, Advanced Video Coding or AVC.

The 2x improvement offered by H.264 creates new market opportunity such as the following possibilities:

- VHS-quality video at about 600 Kbps. This can enable video delivery on demand over ADSL lines.
- An HD movie can fit on one ordinary DVD instead of requiring new laser optics.

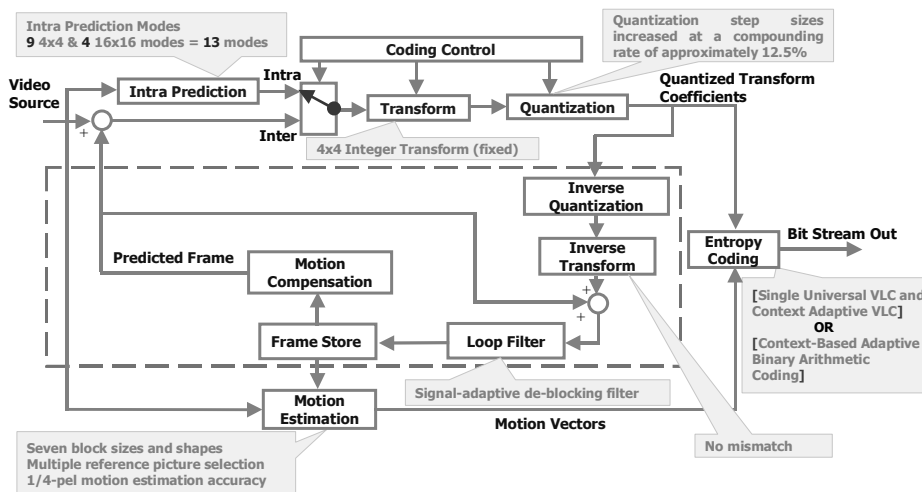


Figure 5: H.264 Block Diagram and Key Features

While H.264 uses the same general coding techniques as previous standards, it has many new features that distinguish it from previous standards and combine to enable improved coding efficiency. The main differences are summarized in the encoder block diagram in Figure 5 and described briefly below:

Intra Prediction and Coding: When using intra coding, intra prediction attempts to predict the current block from the neighboring pixels in adjacent blocks in a defined set of directions. The difference between the block and the best resulting prediction is then coded rather than actual block. This results in a significant improvement in intra coding efficiency.

Inter Prediction and Coding: Inter-frame coding in H.264 leverages most of the key features in earlier standards and adds both flexibility and functionality including various block sizes for motion compensation, quarter-pel motion compensation, multiple-reference frames, and adaptive loop deblocking.

Block sizes: Motion compensation can be performed using a number of different block sizes. Individual motion vectors can be transmitted for blocks as small as 4x4, so up to 32 motion vectors may be transmitted for a single macroblock in the case bi-directional prediction. Block sizes of 16x8, 8x16, 8x8, 8x4, and 4x8 are also supported. The option for smaller motion compensation improves the ability to handle fine motion detail and results in better subjective quality including the absence of large blocking artifacts.

Quarter-Pel Motion Estimation: Motion compensation is improved by allowing half-pel and quarter-pel motion vector resolution.

Multiple Reference Picture Selection: Up to five different reference frames can be used for inter-picture coding resulting in better subjective video quality and more efficient coding. Providing multiple reference frames can also help make the H.264 bitstream more error resilient. Note that this feature leads to increased memory requirement for both the encoder and the decoder since multiple reference frames must be maintained in memory.

Adaptive Loop Deblocking Filter: H.264 uses an adaptive deblocking filter that operates on the horizontal and vertical block edges within the prediction loop to remove artifacts caused by block prediction errors. The filtering is generally based on 4x4 block boundaries, in which two pixels on either side of the boundary may be updated using a 3-tap filter. The rules for applying the loop deblocking filter are intricate and quite complex.

Integer Transform: H.264 employs a purely integer 4x4 spatial transform which is an approximation of the DCT instead of a floating-point 8x8 DCT. Previous standards had to define rounding-error tolerances for fixed point implementations of the inverse transform. Drift caused by mismatches in the IDCT precision between the encoder and decoder were a source of quality loss. The small 4x4 shape helps reduce blocking and ringing artifacts.

Quantization and Transform Coefficient Scanning: Transform coefficients are quantized using scalar quantization with no widened dead-zone. Thirty-two different quantization step sizes can be chosen on a macroblock basis similar to prior standards but the step sizes are increased at a compounding rate of approximately 12.5%, rather than by a constant increment. The fidelity of chrominance components is improved by using finer quantization step sizes compared to luminance coefficients, particularly when the luminance coefficients are coarsely quantized.

Entropy Coding: The baseline profile uses a Universal VLC (UVLC)/Context Adaptive VLC (CAVLC) combination and the main profile also supports a new Context-Adaptive Binary Arithmetic Coder (CABAC).

UVLC/CAVLC: Unlike previous standards that offered a number of static VLC tables depending on the type of data under consideration, H.264 uses a Context-Adaptive VLC for the transform coefficients and a single Universal VLC approach for all the other symbols. The CAVLC is superior to previous VLC implementations but without the full cost of CABAC.

Context-Based Adaptive Binary Arithmetic Coding (CABAC): Arithmetic coding uses a probability model to encode and decode the syntax elements such as transform coefficients and motion vectors. To increase the coding efficiency of arithmetic coding, the underlying probability model is adapted to the changing statistics within a video frame, through a process called context modeling. Context modeling provides estimates of conditional probabilities of the coding symbols. Utilizing suitable context models, the

given inter-symbol redundancy can be exploited by switching between different probability models, according to already coded symbols in the neighborhood of the current symbol. Each syntax element maintains a different model (for example, motion vectors and transform coefficients have different models). CABAC can provide up to about 10% bitrate improvement over UVLC/CAVLC.

H.264 supports three profiles: baseline, main, and extended. Currently the baseline profile and main profile are generating the most interest. The baseline profile requires less computation and system memory and is optimized for low latency. It does not include B frames due to its inherent latency and CABAC due to the computational complexity. The baseline profile is a good match for video telephony applications as well as other applications that require cost-effective real-time encoding [11].

The main profile provides the highest compression but requires significantly more processing than the baseline profile making it difficult for low-cost real-time encoding and also low-latency applications. Broadcast and content storage applications are primarily interested in the main profile to leverage the highest possible video quality at the lowest bitrate [12].

Windows Media Video 9 Series

Windows Media is a leading format for music and video subscription services and streaming video on the Internet. In 2002, Microsoft introduced the Windows Media Video 9 Series codec providing a major improvement in video compression efficiency. Like H.264 it includes many advanced coding tools although the details are different for most of the blocks. Some key features supporting improved compression include:

Multiple VLC Tables: WMV9 main profile contains multiple sets of VLC tables that are optimized for different types of content. Tables can be switched at a frame level to adjust to the characteristics of the input video.

DCT/IDCT Transform Switch: WMV9 supports multiple DCT block sizes including 8x8, 8x4, 4x8, and 4x4.

Quarter-Pel Motion Compensation: ¼ cubic interpolation in addition to support for ½ pixel bilinear interpolation.

Adaptive In-loop Deblocking: Similar function as H.264 but with different details on the filters and adaptive decisions.

WMV9 achieves significant performance improvements over MPEG-2 and MPEG-4 simple profile and has fared well in some perceptual quality rating comparisons with H.264 [13]. Microsoft has submitted the WMV9 algorithm including the simple, main, and new advanced profile to SMPTE to be considered as a formal international standard referred to as VC9 to address concerns of a proprietary format.

WMV9 has lower complexity than main profile H.264 while delivering similar compression efficiency. WMV9 is used heavily in the PC environment and could also become important in networked consumer appliances. WMV9 is gaining momentum with Hollywood and independent film industry with various movie titles starting to be released encoded in WMV9 for high-definition playback on PC DVDs. WMV9 has been under consideration as a compression option for supporting consumer HD DVDs using standard red laser.

DV-25

Many current digital video camcorders use a format referred to as DV to record video on a narrow tape only 1/4" wide. The DV-25 format records 4:1:1 video (unique versus MPEG and H.26x standards which use 4:2:0) achieving a 5-to-1 compression ratio [14]. This results in a data rate of 25 megabits per second. DV uses intra-frame compression based on DCT similar to JPEG.

Compression Ratios Rules of Thumb

Now that we've reviewed some of the major image and video compression standards, let's look at some general rules of thumb that can be used to help characterize the compression factors supported by the different compression standards.

Typical compression ratios to maintain excellent quality are:

- 10:1 for general images using JPEG
- 30:1 for general video using H.263 and MPEG-2
- 60:1 for general video using H.264 and WMV9

For still images, good image quality can be achieved at about 10 to 1 compression ratio. With JPEG2000, this number approaches 13 or 14 to 1. H.263, MPEG-4 simple profile and MPEG-2 provide good image quality at about 30 to 1. And at the expense of 4-10 times the computational capacity of MPEG-2, H.264 provides the much-needed 60 to 1 compression ratio to enable streaming media into the home and HDTV broadcasting. Microsoft's WMV9 provides an alternative with similar efficiency and slightly less complexity for the decoder.

Of course these numbers are very image dependent. As the level of detail in the image content and the amount of motion in the video increases such as with panning of the camera, these numbers can drop drastically. The actual amount will depend on the degree of these changes and how much the viewer is willing to tolerate resulting artifacts. In security applications where the camera may be staring at a still background for several minutes or hours, the compression ratio can far exceed these numbers since there is little to no motion. In these cases the compression ratio can approach 1000 to 1 depending on the content in the scene. In the end the compression factor will depend on the application and the expected performance based on the content being encoded.

Frames Rates for Various Networks

Now that we've looked at what can be achieved with today's state of the art compression technology, let's revisit what types of frames rates can be achieved over the available bandwidth for some of today's networks. Table 1 includes three columns representing our rules of thumb for image compression at 10:1, video compression at 30:1, and advanced video compression at 60:1. The table shows the frames/s at SIF resolution that can be sustained for specific network bandwidths.

<i>Frame Rates</i>		<i>Image Compression</i>		<i>Video Compression</i>		<i>Adv. Video Compression</i>	
		<i>Compressed 10:1</i>		<i>Compressed 30:1</i>		<i>Compressed 60:1</i>	
<i>Network</i>	<i>Kbps</i>	<i>Frames/Second</i>		<i>Frames/Second</i>		<i>Frames/Second</i>	
GSM Digital Cellular	14	1	7	1	2	1	1
56K Modem (PSTN)	56	1	2	2	1	4	1
DSL or Cable Up-Link	128	1	1	4	1	8	1
Cellphone TV	300	3	1	9	1	18	1
DSL Down-Link	768	8	1	23	1	45	1
Wireless LAN (802.11)	11,000	109	1	326	1	652	1

(e.g., JPEG) (e.g., MPEG-4) (e.g., H.264)

Maximum theoretical frame rates for transmitting generic VHS-quality digital video data (352x240 frames) using various networks and compression techniques

Table 1: Frame Rates for Network Bitrate and Compression Ratio Combinations

Lets look at a few of the data points. First of all, using 56 Kbps modem technology over standard phone lines, even advanced video compression at 60:1 only provides 4 frames per second for SIF video. This provides a good indication of the challenge to provide high quality video over phone lines with regular modems.

An emerging application in Asia is TV broadcast to cell phones. At a sample bit rate of 300 Kbps, 30:1 compression allows less than 10 frames/s. 60:1 compression enables support for better than 15 frames per second.

ADSL downlinks can commonly support 768 Kbps in many regions. At this rate, SIF video can be supported at over 30 frames per second. Extrapolating this a bit further, 60:1 compression enables ½ D1 resolution at 1 Mbps and full D1 at 2 Mbps. Previously full D1 required over 4 Mbps. For higher bandwidth networks such as Wireless LAN, SIF resolution can be transmitted comfortably. The emerging challenge will be leveraging advanced video codecs to deliver High-Definition Video.

Real-time Implementation Considerations

Now that we've looked at the features and compression efficiency for the various codecs, lets look at some specific performance requirements for implementing real-time encoding and decoding on a high performance DSP. Table 2 shows the percentage of a 600 MHz Texas Instruments DM642 Digital Media Processor required to sustain D1 resolution for various standards [16,17].

Note that the encoding percentages shown are based on typical test data for an existing implementation or detailed performance estimate. Encoder loading can vary dramatically depending on the target application. Compression standards specify the required syntax and available tools but many algorithm decisions are left up to the implementation. Key variables include the bitrate control approach, single-pass versus multi-pass encoding, ratio of I/B/P frames, motion search range, motion search algorithm, and whether all the available individual tools and modes are used. This flexibility allows different tradeoffs between computational loading and incremental quality improvements.

Percentage of DM64x™ DSP Cycles Required at 600 MHz

Standalone Video Codecs	DM642 - 600 MHz Encode	DM642 - 600 MHz Decode
JPEG	22% (D1)	22% (D1)
MPEG-4 Simple Profile	50% (D1)	12% (D1)
MPEG-2 Main Profile at Main Level	85% (D1)	25% (D1)
Windows Media Video 9 Main Profile	90% (D1)	40% (D1)
H.264 Main Profile	Multi-chip	83% (D1 up to 4 Mbps)
H.264 Baseline (for videophone)	70% (VGA)	30% (VGA)

- ◆ For 4:2:0 video, 30 frames/sec, D1 (720x480); VGA (640x480)
- ◆ Based on typical test data for an existing implementation or detailed performance estimate
- ◆ Encoder implementations can vary significantly depending on feature set

Table 2: DM642 Performance Requirements for Various Video Codecs

As shown in Table 2, Motion JPEG can easily be sustained at 30 frames/s at D1 resolution for both encoding and decoding. Note that the processing requirements for JPEG encode and decode are roughly the same since motion estimation isn't required and reference frames do not need to be maintained in the encoder.

MPEG-4 simple profile requires the least processing of the shown video compression algorithms. Encoding can be done using as little as half of a DM642. More performance can be used to enable slightly higher quality with more extensive motion estimation and higher quality bit rate control. Decoding can be achieved with as low as 12% of the processing. This means simultaneous encoding and decoding is possible with headroom for audio and other control functions.

WMV9 main profile requires significantly more performance for encoding than MPEG-4 simple profile but can still be done in real-time on the DM642. This makes it an attractive candidate for applications like security cameras, real-time video streaming, and personal video recorders. The decoder requires less than half of the processor leaving ample headroom for other functions or the option for 1280 x 720 (720P) @ 30 fps high definition resolution.

H.264 main profile is the most compute intensive of the current standards. It is very attractive though due to the high compression efficiency. Since many of the coding tools are new, fixed function video decoder devices cannot support it without developing new optimized silicon. The DM642 media processor supports real-time main profile decoding at D1 resolution and is finding early market acceptance for products such as IP set-top boxes.

H.264 baseline profile is targeted for video telephony and requires significantly lower performance than the main profile. This is partly a function of the tools and secondly the more relaxed requirements for motion estimation and overall quality versus broadcast content. The DM642 can support simultaneous encoding and decoding of H.264 baseline at up to VGA resolution.

Table 2 illustrates an advantage of programmable digital media processors such as the DM642, which can support all of the existing industry standard and proprietary video formats using the same programmable digital media processor.

Market Trends and Applications

Compression technology is enabling a growing number of digital video products in the market. Figure 6 shows a sampling of the end equipments that can be built using digital video compression ranging from battery operated portable devices to high-performance infrastructure equipment. The optimal processor solution depends on the specific segment. TI has a wide variety of DSPs that support multiple standards and fit key design and system constraints including the low power consuming TMS320C5000 DSPs and OMAP processors to the high performing C6000 DSPs and DM64x digital media processors.

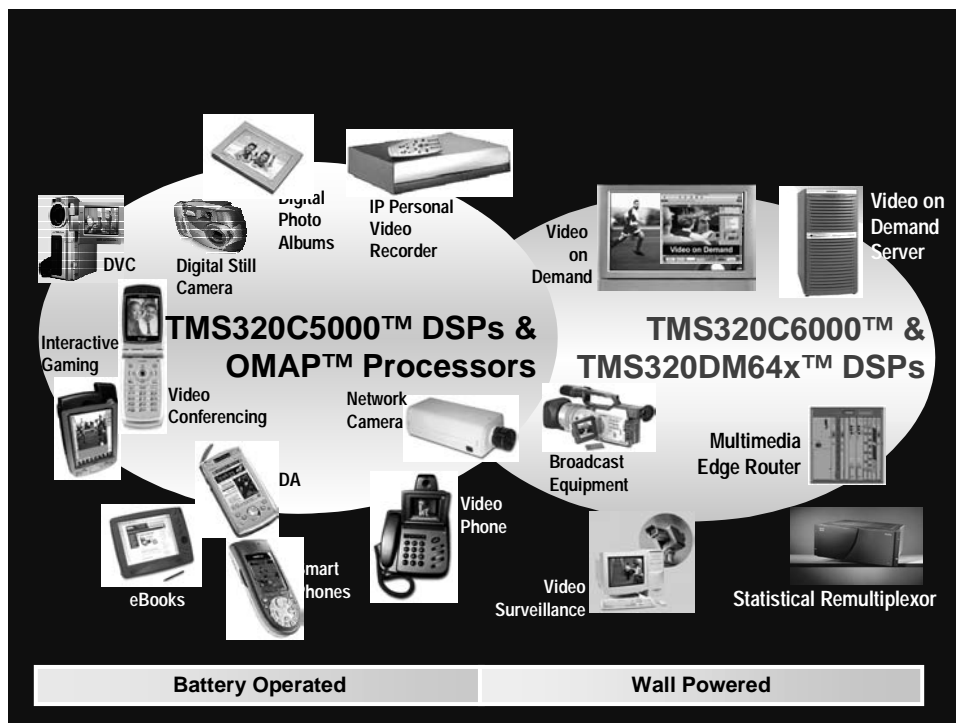


Figure 6: Digital Video End Equipment Categories

Below is a snapshot of some of the existing codecs and potential roadmap for some of the different end systems:

Security/Surveillance: Motion JPEG and H.263 moving to MPEG-4 simple profile. Security digital video recorders often support encoding multiple channels simultaneously and performing intelligent transmission based on image analysis features. MPEG-4 offers a good balance between compression efficiency and processor loading. There is some interest in JPEG2000 for scalability and H.264 baseline profile and WMV9 main profile for higher compression.

Videophone/Videoconferencing: Predominately H.263 with H.261 required for compatibility with legacy systems. H.264 baseline profile provides higher compression while maintaining low delay. Various video conferencing room system vendors have already introduced H.264 baseline profile support using software updates to their previous products based on media processors. IP videophones will soon follow.

Internet Streaming: Windows Media, Real Video, and MPEG-4 (Quicktime and DivX) are common formats. Frequent updates are possible to the algorithms since the primary end devices today are programmable PCs. Programmable decoding will be an important feature as streaming media support gets integrated into consumer appliances. Home gateways and servers may also need to support transcoding from the plethora of formats on the Internet to a subset of formats supported by low-cost clients such as current set-top boxes, which only support MPEG-2.

DVD: Current DVDs use MPEG-2 MP@ML. MPEG-2 MP@HL cannot provide high quality HD content at the data rates supported by current red laser technology. There is ongoing debate in the DVD Forum between staying with red laser and selecting an advanced media codec such as H.264 or WMV9 to support HD versus adopting new blue laser technology. Content is starting to be introduced for high-definition DVD playback on PC using WMV9. China is defining a new optical disk format called EVD and is considering creating their own standard called AVS or using advanced codecs such as On2 VP6 for supporting high definition.

Digital Terrestrial TV: MPEG-2 with support for both standard and high definition. Others standards are being considered in regions other than North America that have not already implemented HD terrestrial broadcast.

Satellite: MPEG-2. Growing interest in H.264 main profile for increased channel capacity.

DSL-based Video On Demand: Interest in WMV9, H.264 and proprietary codecs such as On2 VP6 to achieve better than VHS quality video at less than 1 Mbps.

Digital Still Cameras: Motion JPEG and MPEG-4 simple profile. Possible migration to H.264 baseline profile or WMV9 to allow storing longer video clips and supporting higher resolutions.

Digital Video Camcorders: DV with trend to MPEG-2 and MPEG-4.

Cellular Media: MPEG-4 simple profile with interest in Real Video and H.264.

This brief summary highlights the growing variety of compression formats being used and the growth in the target algorithms being adopted.

Conclusions

A growing number of video compression standards offer increasing compression efficiency and a wider variety of tools that can be tailored for specific end applications. Today's state of the art video compression such as H.264 and WMV9 deliver good quality at around 60:1 compression which represents about a 2x

improvement over the previous generation of video codecs. There are other key factors in choosing a video codec including latency, processor performance, and memory requirements

There is increasing demand for programmable platforms with the market dynamics in today's digital video environment. Digital media processors such as the DM642 offer the performance headroom and architecture flexibility to quickly bring to market implementations of new standards including H.264 and WMV9. Algorithms can be implemented during the standard definition phase and updates made in software to keep up with minor and major adjustments to the standard.

The proliferation of multiple standards and proprietary algorithms make it difficult to select one standard especially since hardware decisions are often made far in advance of the product deployment. Also, the growing trend toward networked connectivity means that many products will increasingly have to support more than one standard. As the number of standards that must be supported increases, there is a growing trend toward flexible media processors in digital video systems.

References

- [1] J. L. Mitchell and W. B. Pennebaker, JPEG Still Image Data Compression Standard. New York: Van Nostrand, 1993.
- [2]. ISO/IEC 15444-1:2000, "Information technology – JPEG 2000 image coding system – Part 1: Core coding system", July 31, 2003.
- [3] ITU-T Recommendation H.261: 1993. Video codec for audiovisual services at px64 Kbit/s.
- [4] ITU-T Recommendation H.263: 1998. Video coding for low bit rate communication.
- [5] ISO/IEC 11172-2:1993. Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbit/s -- Part 2: Video.
- [6] ISO/IEC 13818-2:1995. Generic coding of moving pictures and associated audio information: Video.
- [7] ISO/IEC 14496-2:2001. Information technology – Generic coding of audio-visual objects – Part 2: Visual.
- [8] ISO/IEC 14496-10:2003. Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding.
- [9] Emerging H.264 Standard: Overview and TMS320DM642-Based Solutions for Real-time Video Applications. UB Video Inc. www.ubvideo.com.
- [10] Report on the Formal Verification Tests on AVC (ISO/IEC 14496-10 | ITU-T Rec. H.264), ISO/IEC JTC1/SC29/WG11, MPEG-2003/N6231, December 2003, Waikoloa.
- [11] H.264 Based Video Conferencing Solution Overview and TMS320DM642 Digital Media Platform Implementation. UB Video Inc. www.ubvideo.com.
- [12] UBLive-264MP: An H.264-Based Solution on the DM642 for Video Broadcast Applications. UB Video Inc. www.ubvideo.com.
- [13] J. Bennett, A. Bock, "In-Depth Review of Advanced Coding Technologies for Low Bit Rate Broadcast Applications," International Broadcasting Convention 2003 Conference Publication, pp. 464-472.
- [14] Draft Spec SMPTE Standard for Television: VC-9 Compressed Video Bitstream Format and Decoding Process. SMPTE Technology Committee C24 on Video Compression. 2003-09-07.
- [15] IEC-61834
- [16] DM642 Technical Overview. Texas Instruments (SPRU615) 2002. www.ti.com.
- [17] TMS320C6000 CPU and Instruction Set Reference Guide (SPRU 189F), Texas Instruments, 2002. www.ti.com