ANALOG DEVICES

AN-336A
APPLICATION NOTE

ONE TECHNOLOGY WAY • P.O. BOX 9106 • NORWOOD, MASSACHUSETTS 02062-9106 • 617/329-4700

Image Compression: Spelling Out the Options

by Chris Cavigioli

Been confused about the approaches being developed for this vital technology? This could help.

Image data compression can be used to reduce channel bandwidth requirements when transmitting still or moving images over a band-limited channel, and enables images to be stored in a smaller memory space. These are among the most important requirements for furthering imaging in the immediate future. Compression is a prime subject of discussion in electronic publishing, videoteleconferencing, medical image sharing, real-time missle guidance by visual control and space exploration—or in looking at the future of color copiers, still video cameras, ISDN networks (voice, video, text documents and images), video channels on commercial broadcast satellites, image databases, digital office FAX and PC FAX boards.

There exist many approaches to reducing image data; the end application dictates which technique is best suited for compressing data. For example, some techniques are *lossy* techniques while others are *lossless*. Medical images, for example, often require completely lossless compression because the exact value of each pixel could be significantly important in a mathematical computation or in a legal dispute. Browsing an ID card database, or guiding a missile in the battlefield, on the other hand, could be carried out without having to know the exact value of each pixel. The human visual system is adaptive enough to be "fooled," seeing a lossy-compressed image and confusing it—accepting it—for the original. In fact, most pictures can be processed to remove 97% of the original data, and the human eye would have a very hard time at noticing any degradation.

Chris Cavigioli is DSP Applications Engineer, focusing particularly on communictaions applications, at Analog Devices, Inc., Norwood MA, a major vendor of DSP ICs and other imaging components.

The compression techniques

Compression techniques include: transform-based coding, sub-band coding, entropy coding, run-length coding, vector quantization, predictive coding, block-truncation coding, and coding by modeling a scene. The latter has recently become quite exciting, as ways of applying completely new approaches involving fractals are being discovered. Since there are so many different compression techniques available, and since certain compression applications actually employ several of those techniques in combination, I will try to navigate through several techniques, discussing each along the way.

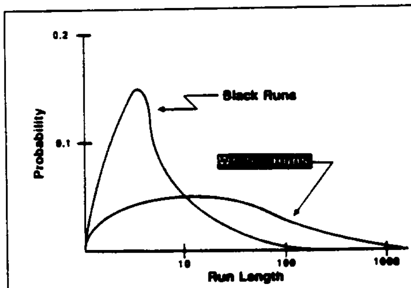

Image-Pixel Run-length	Corresponding Modified Huffman Codes: white runs	black runs
0	00110101	0000110111
1	000111	010
2	0111	11
3	1000	10
4	1011	011
5	1100	0011
6	1110	0010
7	1111	00011
8	10011	000101
9	10100	000100
10	00111	0000100
11	01000	0000101
12	001000	0000111
13	000011	00000100

Fig. 1. Run length coding.

Transform coding applies a mathematical transform to the pixel data to put the resulting data values into another domain. A transform is chosen which consolidates pixel information into a more compact form. Once in this form, other techniques are employed to compress the information even further. For example, in a lossy system, thresholding and quantization can be used to select the more significant values from the less significant ones, and then transmit only the most important ones. For lossless systems, thresholding and quantization wouldn't be allowed. In either case, entropy coding (such as Huffman coding) and run-length encoding can follow to reduce the bit rate even further. The most popular transform today would be the discrete cosine transform (DCT). In the past, before high performance, programmable DSPs were available, the Hadamard transform enjoyed some usage. The DCT is relatively compute-intensive, requiring fast multiplies and sophisticated data addressing. The Hadamard transform was initially popular since it can be computed with simple ±1 operators, thereby eliminating the need for hardware multiplication.

Run-Length coding capitalizes on the high pixel-to-pixel correlation in the image, and then recodes the data to eliminate the redundancies. To achieve a high degree of compression, the run-length data is recoded into digital codes of varying length, with shorter codes assigned to the more likely original code values (see Figure 1). The common facsimile machine uses the CCITT Group 3 standard to first determine the length of a continuous run of white (or black) image pixels, and then assigns (via a modified Huffman code table) a corresponding code between 2 and 13 bits long.

Sub-band coding passes an image through a bank of special filters called quadrature mirror filters (QMFs). A QMF filter has the special property of reducing the input sampling rate by two and splitting the input bandwidth into two output channels, each having half of the original bandwidth. For example, say

Reprinted from Advanced Imaging — October 1990

we have a signal which is sampled at 100 KHz: the input bandwidth spans 0 Hz to 50 KHz. By passing this signal through a quadrature mirror filter, two output channels result—one has all the 0 Hz to 25 KHz information while the other has the 25 KHz to 50 KHz information. Both output channels are each sampled at 50 KHz (half of the original sampling rate). After several stages of QMF filtering, an image is separated into several smaller images, each image holding data within a known spatial frequency band. The lower frequency imformation is the most important information, while the high frequencies hold the crispy details of a picture. By selectively transmitting only some of the lower frequency portions, data compression is achieved.

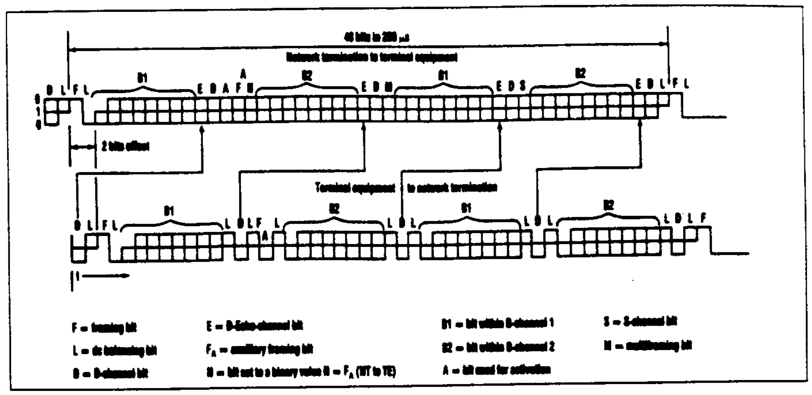

Fig. 2. Basic Access ISDN.

Progressive coding and JPEG

Sub-band coding is ideally suited for a technique called *progressive coding*. Progressive coding is especially useful for archival database browsing. When browsing through a database, only the lowest frequency image portion are first retrieved: the viewer can get a good idea of the image. The longer the viewer observes the image before going on to the next, the more detail is added to the picture. This is done by progressively adding the next higher frequency portions until all the original data is recreated.

Progressive coding is also possible with transform-based techniques such as the discrete cosine transform. For example, the much-discussed JPEG still-image compression standard developed by the International Standards Organization (ISO) specifies a sequential DCT technique as its absolute minimum (baseline): this means that the algorithm performs its compression tasks on the whole input image in a sequential fashion. First an image is subdivided into smaller blocks; then the blocks are transformed using the DCT; then the DCT coefficients are thresholded and quantized; and finally they are entropy coded. After all the tasks are done, the data can then be stored or transmitted. On the receiving side, the compressed data is received and sequentially decompressed. When that is all done, the decompressed image data is finally displayed on the monitor.

JPEG permits the offering of more advanced features above the baseline requirement—one of these features is a progressively-coded DCT implementation. The user sees a rough image immediately, which is then progressively improved in quality. Progressive DCT coding in JPEG can be achieved in one of two ways: *spectral selection*, where selection of DCT coefficients to transmit is based on their associated frequency; and *successive approximation*, a method whereby an averaged approximation of the DCT coefficients is sent first, followed by successively better correction values sent later to improve the image.

While progressive coding usually means that a larger bandwidth requirement is sacrificed for a longer transmit duration on a narrow bandwidth, *flexible bandwidth coding* can be used to describe a scheme in which the channel bandwidth itself can be altered in order to accomodate various picture quality requirements in real-time.

Phone transmission

A good example of this would be videotelephony on the telephone network. The CCITT recommendation H.261 for moving video compression is used in picture phone and videoteleconferencing equipment and is sometimes referred to as *p x 64*. The name p x 64 stems from the fact that the channel bit rate can be an integer (p) multiple of 64 Kbps. A standard digital voice channel on today's telephone network is a 64 Kbps channel. A p x 64 video codec can selectively utilize several 64 Kbps channels to achieve the desired picture quality. H.261 was originally drafted for video telecommunications over digital telephone lines having multiple channels multiplexed in time. It should also be mentioned that one of the 64 Kbps channels is reserved for the audio portion of the telephone call.

The most common *digital carriers* for H.261 would be the standard T1 carrier (North America) or the CEPT carrier (Europe): both of these carriers are refered to as the DS1 (digital service 1) layer. The phone line which comes to your house carries one voice channel at 64 Kbps—this is the DS0 layer. At the central office, several DS0 channels are time-multiplexed (TDMA) onto the DS1 layer. In North America, 24 channels are combined into a frame on the DS1 layer; this protocol yields a total bit rate of 24 x 64 Kbps or 1.536 Mbps and is called T1. In Europe, 32 DS0 channels are framed onto Europe's DS1 layer; this protocol is called CEPT and yields 32 x 64 Kbps or 2.048 Mbps.

Four *enabling factors* have emerged over the last few years to make videotelephony a reality: plummetting cost of DS1 services and associated tariffs, standardization of the H.261 recommendation by the CCITT (June 1990), DSP chips enabling affordable implementation of videotelephony products, and live demonstrations of working products by at least three firms in the U.S.A.

To make videotelephony even more attractive, the telephone company now also sells new digital services to the subscriber including: fractional T1, switched 56, and ISDN. *Fractional T1* implies that instead of purchasing all 24 channels of T1 service, it is now possible to purchase, for example, only 12 of the channels at a reduced cost. *Switched 56* is the common name for dial-up lines which carry a single channel of digital data at 56 Kbps. In this case, you would have to purchase one line for video, albeit at a reduced picture quality, and one line for the audio portion. The price of switched 56 is the factor that is making it immensely popular.

ISDN is a bit slower in being widely accepted in the U.S.A., but its time is definitely on the way. *Basic Access ISDN* consists of two independent B channels together with a D channel (see Figure 2). The B channels are 64 Kbps and the D channel is only 16 Kbps for basic access; *Primary Access ISDN* is at the DS1 level. In this case, there are 23 B channels (North America) in conjunction with a 64 Kbps D channel. In Europe, there are 30 B channels with a 64 Kbps D channel.

Images into the stream

With that said, let's look at how a picture is taken from a camera and compressed into a low data-rate bit stream. To get a better idea of the image compression task, consider the magnitude of the data reduction required. In the H.261 recommendation, the input picture frame size can be in one of two spatial resolutions: there is the *CIF* (Common Intermediate Format) which has a resolution of 360x288 pixels, and there is the *QCIF* (Quarter Common Intermediate Format)

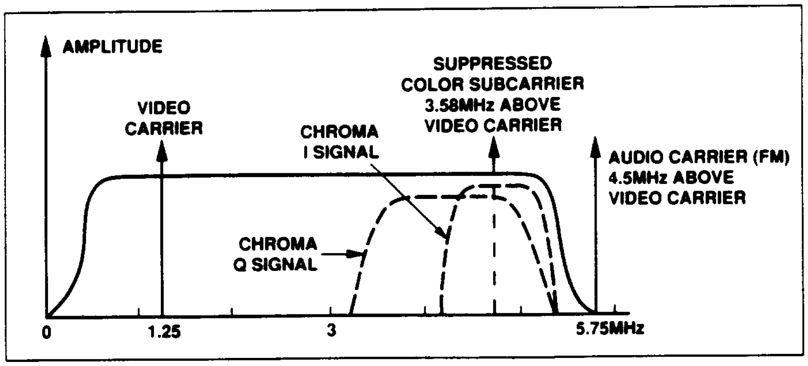

Fig. 3. Broadcast television signal.

which has a resolution of 180x144 pixels. Two formats enable basic units to support QCIF for a lower price, while higher-priced units support both formats. For a moving picture, the frame is updated many times every second: full motion is regarded as 30 frames per second. Since this represents a very large amount of data, the input data is often temporally subsampled—subsampling at a rate of 2, 3, or 4 yields a frame rate of 15, 10 or 7.5 frames per second respectively.

The number of bits per pixel depends upon whether each frame is color or not. The data format closely resembles that of standard television. In other words, instead of one frame each of red, green and blue as found in your computer CRT, television signals are based on luminance (the gray scale or black-and-white portion) information and chrominance (the color) information. (This was developed historically so that black-and-white television and color television could coexist.) Either receiver demodulates the luminance information, and color TVs have additional circuitry to demodulate the chrominance information. Chrominance comes in two parts which are quadrature modulated; television signals are sometimes refered to as being in YIQ format. Y stands for luminance, and I and Q are the two chrominance components (see Figure 3). Since the human eye is more responsive to luminance degradation versus chrominance degradation, the chrominance is often spatially subsampled. The CIF and QCIF formats subsample chrominance at a 2:1 rate.

Video signals are sampled at 8 bits resolution. Consequently, a full-motion, color, CIF bitstream runs at 37.3 Mbps! All this data must somehow be shoehorned into p x 64 Kbps—this formidable task can be achieved with the CCITT H.261 compression algorithm. To help reduce the data to be compressed, provisions for motion compensation have been included in the algorithm.

Motion compensation is a technique whereby a frame is compared to the previous frame (in time) and the movement of objects in the scene is estimated. Stationary objects are not required to be recoded and retransmitted since the receiver already has these objects in its frame memory from previous frames. Using this technique, it becomes possible to code a frame, then for several next frames, simply code and transmit the difference information to the receiver instead of completely new frames. Motion compensation aids tremendously in cases such videoteleconferencing where a subject's head and lip movements are practically the only moving objects in the frame sequence.

Motion compensation enables large data reduction performance at a very slight expense of increased control signal requirements. There are times when motion compensation will work and times when it cannot. Signalling bits are specified in H.261, informing the receiver whether it should be in *interframe* mode versus *intraframe* mode. Interframe mode indicates that data being sent is frame difference data which should be used to update the receiver current frame store. Intraframe mode means that the receiver should throw away its stored frame and start over from scratch with a completely new picture. Intraframe mode is applicable anytime that the interframe differences are too great and a fresh frame would be better, for instance, when the very first picture is being sent, a scene cut has happened, or too much time has elapsed since the last complete frame was sent.

We will look at working examples of practical methods of image compression in a future issue of this magazine— including a close look at discrete cosine transform-based compression employing DSP hardware.