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

JPEG Compression

by Chris Cavigioli

Software-based? Or which hardware type? Lossy or lossless? More mysteries clarified

In part one of this image compression overview, ("Image Compression: Spelling Out the Options," *Advanced Imaging*, Oct. '90) we addressed the very general issue of compressing image data, looking at options for image transmission over limited bandwidth channels.

In this second part, we focus on the compression of continuous tone images using the much-discussed, sometimes understood ISO/JPEG compression algorithms. (JPEG, incidentally, stands for Joint Photographic Experts Group: this is a committee which is proposing a standard for compressing high-quality, still images. The standardization activity (e.g., JPEG) throughout the U.S. is coordinated by ANSI (American National Standards Institute), and ANSI, in turn, casts the U.S. vote at the International Standards Organization (ISO) meetings.)

Our focus is on two questions: Which JPEG variant should be used to compress a given image, and should compression be done with PC software, a JPEG VLSI chip set, or a programmable DSP?

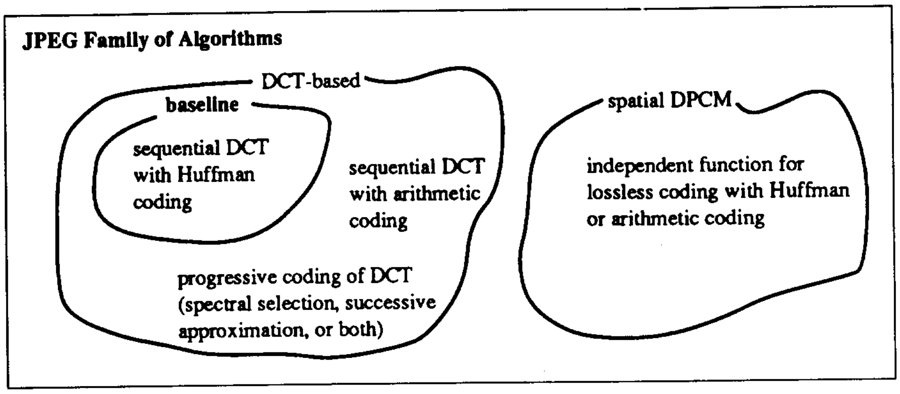


Fig. 1. The Family of JPEG algorithms.

Which JPEG?

JPEG is not a single, fixed algorithm, but rather a family of several techniques, each member best suited for certain applications. All JPEG coders must support the baseline technique, with the exception of lossless JPEG coders, which are not required to support baseline. Some JPEG coders will support extended systems which are above and beyond the fundamental baseline technique. Fig. 1 shows the relationship among all current JPEG algorithms.

JPEG compression algorithms have been tested on images of natural scenes. These images averaged 16 bits/pixel and were stored in the CCIR-601 4:2:2 (Y:Cb:Cr) format. Table 1 shows experimental compression ratios achieved and the resulting image quality.

Lossy, high compression techniques are DCT-based (Discrete Cosine Transform), while the lossless but low compression techniques are DPCM-based (Differential Pulse Code Modulation). Within the DCT-based realm, there is the baseline mode which all DCT-based JPEG coders must support as well as the extended systems. The full name for the baseline system is: *sequential DCT system with Huffman coding.* Let's see how this compares with other JPEG algorithms.

Table 1: JPEG Compression Ratios

Bits/Pixel	Image Quality	Compression Ratio
0.1	Recognizable image	160 : 1
0.25	Useful image	64 : 1
0.75	Excellent quality	22 : 1
1.5	Indistinguishable from original	11 : 1
8	Lossless JPEG method	2 : 1

Progressive vs. sequential coding

Sequential and progressive coding refer to the order in which the compressed information is sent from encoder to decoder and subsequently displayed. Sequential systems begin upper left-hand corner of the image, then proceed to completely encode the image from left to right across the rows, and top to bottom down the columns. When you observe the decoder reconstructing the image, you see the final image appearing in the same order, analogous to covering a picture by a piece of paper and slowly revealing the picture by sliding the paper away.

With the same decoder in progressive mode, a coarse approximation of the image instead fills the screen right away, giving you a good idea of what the final image looks like. The image progressively improves in quality until it is indistinguishable from the sequentially-generated image. The progressive technique would be better suited for applications such as image database browsing, while the sequential technique is practically mandatory in other applications, such as hardcopy printing.

Progressive coding choices

Two progressive coding techniques, spectral selection and successive approximation, are defined by JPEG. The DCT transforms spatial pixel data into frequency data. By picking only certain frequency bands of interest (by choosing only some of the DCT coefficients), it is possible to approximate an image by first using only low frequency components, then add detail by sending some missing high frequency components. When all the coefficients have been sent, the final picture results. This is the spectral selection technique.

Reprinted from Advanced Imaging – March 1991

The successive approximation technique sends the approximate value of the DCT coefficients by truncating their LSBs (least significant bits) in the first stage. The missing low order magnitude bits are sent in the later stages, one bit plane at a time. Hybrid algorithms employing a combination of both techniques are allowed.

All of the images in the examples *(at right)* are JPEG compressed at 15:1. There are three images in each compression mode, giving 10%, 80%, and 100% of complete image transmission. One set shows baseline mode, the second set, progressive mode using spectal selection only, and the final set, progressive mode using both spectral selection and successive approximation.

Entropy coding issues

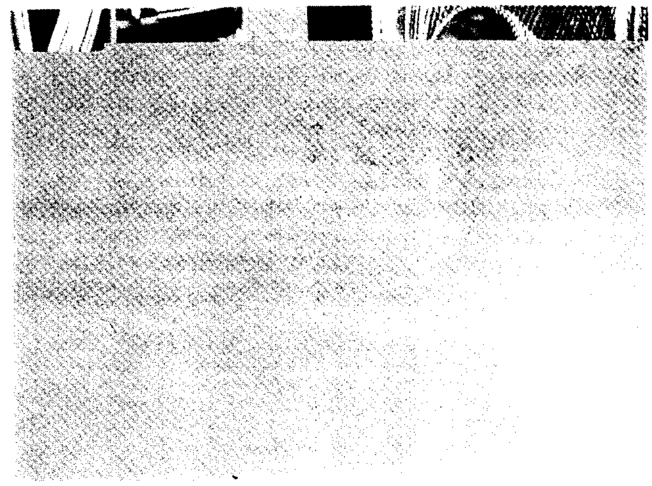
Some of the JPEG extended systems replace the Huffman coding technique of the baseline system with adaptive binary arithmetic coding. Either method is a lossless way of entropy coding the quantized DCT coefficients. Huffman coding replaces the quantized coefficients with variable length codewords (VLCs). The coefficient is the look-up index into a table, and the VLC extracted from that location is the data sent. The codeword length is shortest for statistically more frequent coefficient values and longer for less frequent coefficients, so the total amount of data transmission is reduced. (Morse code is an example of Huffman-like coding for the characters in the English language.) Obviously, different images possess different statistical distributions of DCT coefficients.

For this reason, in addition to the baseline default tables, JPEG allows custom Huffman tables to be generated from optional signaling information. Custom tables can be used for certain classes of images or even specifically optimized for each individual image. Adaptive binary arithmetic coding provides adaptive entropy coding in a single pass, eliminating the need for custom tables used in the Huffman technique, by dynamically adapting its set of probability estimates to each image or even to regions within an image.

Independent function for lossless

A separate, independent JPEG function (not associated with DCT-based techniques) is specified by JPEG for sequential, lossless coding. This algorithm is a spatial algorithm in the pixel domain as opposed to the transform domain. It is based on the DPCM coding model originally developed for the DC coefficients of the DCT in the DCT-based systems. However, the model is extended by incorporating two-dimensional prediction.

Lossless compression cannot achieve high compression ratios like the lossy ones can—but lossless compression may be an end-application requirement. For example, medical images usually are not allowed to be lossy compressed. Other images where the pixel data is qualitatively important may be scientific data, remote sensed or computer-generated visualization images.



JPEG alternatives: 10%, 80% and 100% transmissions (left to right) using baseline (top), spectral selection only (middle), and both spectral selection and successive approximation (bottom). Note missing stripes on her clothes when transmission is only 80% complete.

Which way?

After deciding which JPEG algorithm is needed, the final issue is implementation—by what system should the JPEG compression algorithm be carried out?

CPUs and RISCs: Compressing images with software programs executing on standard computational platforms such as a PC, for example, has advantages. Software is available today, and such software can be modified, ported, or embedded into other products. Execution speed is the major drawback of this approach. Today's 80286 processors or even RISC chips cannot do the calculations fast enough for fast compression, let alone realtime compression. Using general purpose CPU software for JPEG compression only makes sense when compression requests are infrequent.

Building blocks and chipsets: ICs are available which perform computationally intensive tasks very quickly. These ICs—chip sets—can be connected together to implement JPEG compression on a circuit board in hardware. Examples of these building blocks would be 8x8 DCT chips, motion vector estimation chips, and now even some quantization and entropy coding accelerators. Both SGS-Thompson/INMOS and LSI Logic offer these types of chips. The advantage to using these chips is primarily their speed. Typically, you would select one vendor and use that chipset; mixing vendors may be difficult if the timing and other signals between chips are incompatible.

The disadvantages of chip sets are high cost of the completed implementation and relatively little flexibility. Flexibility is extremely important as users demand

dynamic solutions and as systems migrate towards multimedia capabilities. For example, today's chipsets are fine for sequential DCT systems with Huffman coding (if the final standard is identical to the current version), but what happens if the end-application requires arithmetic coding, or lossless compression, or maybe even simply progressive coding?

JPEG VLSI chips: Single-chip VLSI JPEG coders, such as the CL550A from C-Cube, combine the separate building block ICs of chip sets into a single IC, reducing cost, power, and complexity. The drawback for the end-user is that this is the most rigid and inflexible solution. All the inflexibility of chip sets exist with single-chip solutions, but more severely, since the "chip set" has been already arranged for you in VLSI and there is no access to the internal architecture for modifications, no reprogrammability in a turnkey, baseline JPEG chip.

Case for programmable DSPs

For many applications, the best balance between cost, flexibility, and processing throughput is achieved by reprogrammable digital signal processors. DSPs are optimized for signal processing computations, much more so than even the fastest RISC chips. Because DSPs are general purpose, cost is low due to the large volume of DSP chips manufactured. Development time is sharply reduced because of mature and powerful development aids. DSPs are available from Analog Devices, Texas Instruments, Motorola, and AT&T, for example.

Flexibility is their outstanding feature: the same DSP can perform JPEG compression at one instant, digital audio compression at another, even handle incoming telephone calls upon interrupt control, and interface to UARTs, MIDI controllers, as examples. However, DSPs are probably not the best choice in rigid, single-application, embedded systems where JPEG is the sole task to be performed at real-time, because those applications do not require the DSP flexibility and probably could not afford the slower DSP throughput.

DSPs today are available on bus plug-in boards. Future personal computers with multimedia options will use powerful DSPs on the motherboard, alongside the CPU, to perform various signal processing tasks as they are developed. Ironically, much of the progress in multimedia applications will be made by developing the algorithms right on the personal computer—by bringing this capability to the general engineering public.

By choosing a DSP in a related family, based on a powerful and flexible internal open architecture, code compatibility is assured. This allows an upgrade path and portable software without risking the investment already made in code development. An example of such a family is the ADSP-2100 family of DSPs from Analog Devices, Inc. with a variety of memory, I/O and packaging options, but with a unified, common register set and basic architecture.

For example, due to their algebraic syntax instruction set, assembly language programming on these DSPs is much easier than typical "assemblers." Gone are the cryptic mnemonics, operators, and operands associated with early (and many present) DSPs. Features of these DSPs which are needed for DCT computations include the nestable zero-overhead looping, the independent barrel shifter, and the data address generators (DAGs) with bit-reverse capability. For zigzag scanning of the DCT coefficients, the DAGs are indispensable, because of the ability to mix-and-match index registers with modify registers, allowing single-cycle access in a zigzag fashion. With each access, the index pointer is updated by one of four available modify registers, depending on which direction the zigzag scan proceeds. The independent barrel shifter (capable of N-bit left/right shifts in a single cycle) is especially powerful in variable length entropy coding (VLCs, Huffman, arithmetic).

With compression algorithms and associated ICs proving the enabling technology at affordable prices, a major revolution in imaging storage and transmission is imminent. Until now, the availability of moving color pictures with sound has been limited to non-computer electronics such as television. Providing this capability to the user at the computer interface will significantly reduce the barrier between computers and the general public.

Our appreciation to Jørgen Vaaben at autoGraph International A/S, Lyngby, Denmark, for the images.