The Basics of Video Compression

Marko Slyz

February 18, 2003

(Sourcecoders talk)

Outline

1. Non-technical Survey of Video Compressors
2. Basic Description of MPEG 1
3. Discussion of Other Compressors
4. Numerical Comparisons

Some Applications of Video Compressors

MPEG-1 Video-CD

MPEG-2 HDTV, DVD (usually), some camcorders, current digital cable, TiVo/ReplayTV

MPEG-4 (a.k.a. divx) internet video, future digital cable, at least one camcorder, cell phones.

H.261, H.263 video conferencing (e.g. Polycom)

H.26L (a.k.a. MPEG-4 part 10, JVT H.264?) (Not yet done?) Intended for both low-delay and high-delay applications.

Another Classification of Video Compressors

	formally standardized	not formally standardized
requires royalties	MPEG-1 (?) MPEG-2 MPEG-4	Corona (MS) RealVideo Sorenson
doesn't require royalties	H.261 H.263 H.26L (base)	vp3 (on2) Tarkin (ogg) WaveVideo (?)

ISO/IEC Standard

ITU-T Standard

Basic Idea of MPEG 1 or 2 (and . . .)

An uncompressed video file consists of a sequence of frames.

MPEG (often) uses DPCM between frames, and uses a block-based DCT within a frame.

MPEG has lots of forward-adaptive parts, lots of static parts, and no (?), except for the DPCM just mentioned, backward-adaptive parts.

MPEG 1 or 2 Beginnings

To code a particular frame, start by

1. transforming its colors into Y,Cb,Cr space,
2. then partitioning it into macroblocks, each of which consists of a 16x16 block of pixels from Y, and 8x8 blocks of the subsampled versions of Cb and Cr (4:2:0).

Intra Macroblock Compression

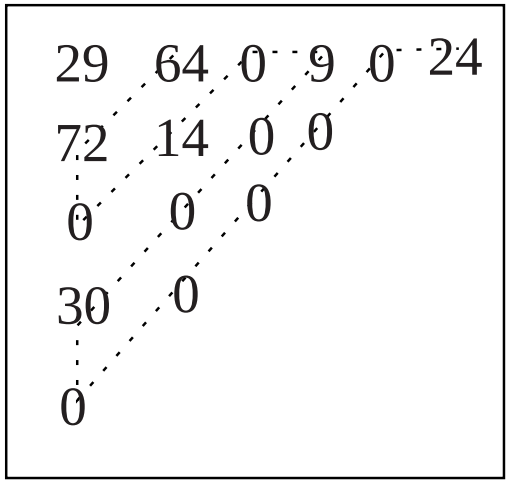
One way of compressing a macroblock is by

1. DCT transforming it,
2. then scalar quantizing it (with larger step-sizes for higher frequencies) (and deadzones in each quanitzer)

```
8? 16 19 22 26 27 29 34
16 16 22 24 27 29 34 37
19 22 26 27 29 34 34 38
22 22 26 27 29 34 37 40
22 26 27 29 32 35 40 48
26 27 29 32 35 40 48 58
26 27 29 34 38 46 56 69
27 29 35 38 46 56 69 83
```

3. Huffman coding the difference between the current and last DC coefficient.
4. scanning the AC coefficients in zig-zag order, and coding the run-lengths of zeros, and the non-zero terminating values.

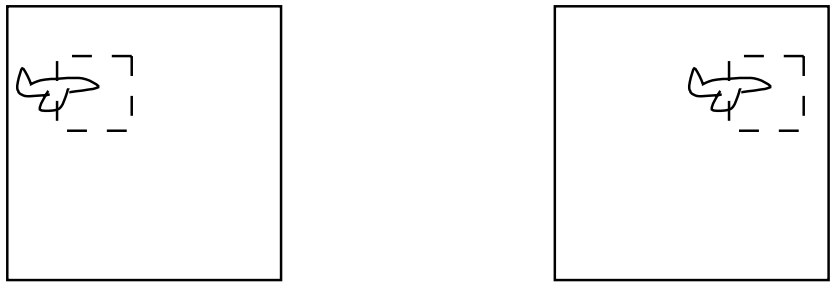
A More-Detailed Look at Coding AC Coefficients



Matrix of DCT
Coefficients

Before Huffman coding, this would be converted into the symbols
$\{(0,64),(0,72),(1,14),(1,9),(2,30),(5,24),\ldots\}$.

Backward Predictive Macroblock Coding



previous frame (”anchor”)

current frame

1. Find a macroblock in the previous frame that’s as similar as possible to the current macroblock.
2. Code the offset to that block (the “motion vector”).
3. DCT the difference between the two macroblocks.
4. Quantize (USQ with step size 16) the DCT coefficients.

Step 1 can be done to 1/2 pixel resolution.

I and P Frames

Each frame in the sequence could be an

I frame, which consists only of intra coded macroblocks, or a

P frame, which consists of either intra or backward predicted macroblocks. *Skipped* macroblocks are another possibility.

Examples of valid frame assignments for a 10 picture sequence:

```
I  I  I  I  I  I  I  I  I  I

I  P  P  P  P  I  I  P  I  P
```

B Frames

A forward predicted macroblock uses a *future* frame as the source of its matching macroblock.

A bidirectionally predicted macroblock averages a macroblock from a past frame with a macroblock from a future frame.

A **B frame** can have skipped or intra, backward, forward or bidirectionally predicted macroblocks.

Example of an video sequence with B frames:

```
I B I P B B B P I B
```

Only the nearest P or I frame can be used for prediction.

Disadvantages of Using B Frames

Slows down motion compensation.

Also, requires more buffering because B frames must be decoded after the frames they depend on.

So

$I_1\ B_2\ I_3\ P_4\ B_5\ B_6\ B_7\ P_8\ I_9\ B_{10}$

might be transmitted as

$I_1\ I_3\ B_2\ P_4\ P_8\ B_5\ B_6\ B_7\ I_9\ B_{10}$

Advantages of Using B frames

1. Predictions are better, because they can account for newly appeared objects.
2. The compressor can tolerate more distortion in B frames, since they aren’t used for predictions of other frames.
3. B frames can speed up access, since the compressor can skip B frames when seeking to a particular position. [Thanks Kevin.]

Rate Control

To keep a buffer from underflowing MPEG can use bit stuffing.

To keep a buffer from over or underflowing MPEG can scale all the quantizer step sizes for certain macroblocks, or it can increase the number of skipped macroblocks, or it can low-pass filtering the video before doing anything. [Gibson]

MPEG 2

1. Can handle
   (a) interlaced video
   (b) other color subsamplings (4:2:2 and 4:4:4)
   (c) More frequent (every picture) specification of quantization matrices.
2. Can produce scalable video (in several ways), i.e. it can transmit a low-rate base video layer, and then an enhancement layer.
3. Has a more flexible Systems layer (which combines video and audio).

H.26L

1. Can use (simple) predictive coding for intra blocks.
2. Can use more than one anchor frame for motion compensation.
3. Can use smaller blocks (down to 4x4) for motion compensation.
4. 1/4-pixel accuracy motion estimation.
5. Uses a deblocking filter.
6. Integer DCT.
7. Context-based entropy coding (??)
8. Can be low-delay if that's important (like for video-conferencing).
9. Needs about three times the number of gates as MPEG-2 for the same speed. (http://www.eetimes.com/story/OEG20020920S0049)

MPEG-4

Can segment each image into a bunch of objects, and encode each one separately.

Can remember the background, as the camera pans.

Can handle synthetic data (like text) specially.

Can work at very low rates.

Numerical Comparisons

Output is at 1024 kbps on a 30 fps 352x288 color images (called “Mobile”).

	SNR on Y	SNR on Cb	SNR on Cr
H.26L	34.9	36.7	37.2
MPEG-4(ASP)	31.4	35.4	35.8
H.263	29.7	34.5	34.9
MPEG-2	27.7	32.3	32.5

from

http://216.239.39.100/search?q=cache:g4rDRJCqbSAC:standards.pictel.com
/ftp/video-site/0109_San/VCEG-N18.doc+h.26l+mpeg-2+comparison&hl=en&ie=UTF-8