Performance of a Software MPEG Decoder

ACM Multimedia 93

Ketan Patel, Brian C. Smith, and Lawrence A. Rowe
Computer Science Division - EECS
University of California at Berkeley
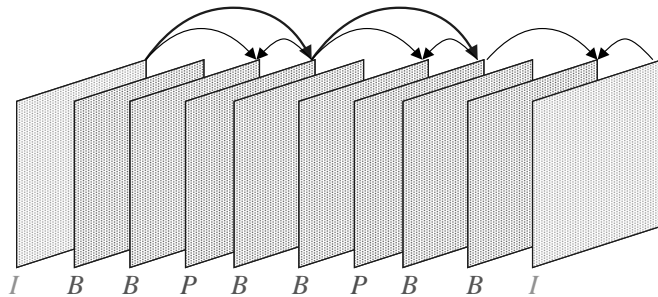**(larry@cs.berkeley.edu)**

Outline

MPEG video compression

Software decoder

Performance

Future plans

MPEG Video Stream

I   B   B   P   B   B   P   B   B   I
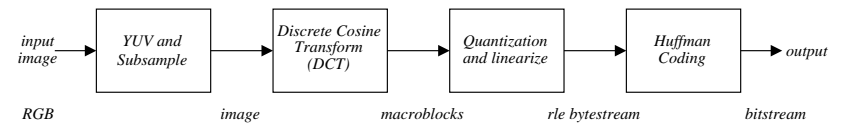
Different types of frames

I - intracoded frame
P - forward predicated frame
B - bi-directional/interpolated frame
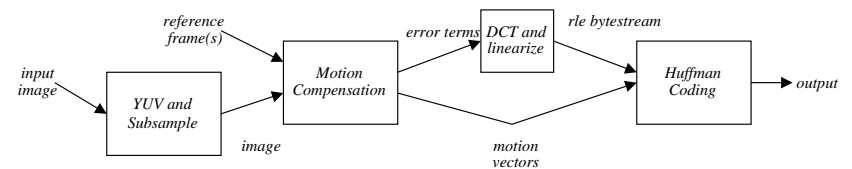
Frames delivered in decode order

Frame Compression

I frames:

input image → YUV and Subsample → Discrete Cosine Transform (DCT) → Quantization and linearize → Huffman Coding → output

RGB   image   macroblocks   rle bytestream   bitstream

P/B frames:

input image → YUV and Subsample → Motion Compensation → error terms → DCT and linearize → rle bytestream → Huffman Coding → output

image   motion vectors

reference frame(s)

Decoder

Algorithm - parse bitstream and undo compression

Written in C (15K lines of code)

Ported everywhere

  Unix/X Windows

  PC/Windows 3.x

  Macintosh

Code freely distributed

  FTP from *toe.cs.Berkeley.edu: pub/multimedia*

Relative Performance

Original code spent 60-80% time dithering

Using ordered dither into a fixed color map ...

Operation	%Time
Parsing	17%
Inverse DCT	14%
Reconstruction	32%
Dithering	24%
Misc arithmetic	10%
Other	3%

IDCT is not the bottleneck

Reconstruction and dithering are bottleneck, problem is memory bandwidth

---

Real Time?

Canyon: 144x112, 49:1 compression (1:1:4, 0.49 bpp, 0.24Mbs)
Flower: 320x240, 50:1 compression (1:4:10, 0.49 bpp, 1.00Mbs)
Berkeley MPEG decoder v2.0 running on Unix

Machine	Canyon	Flower	Clock	Cache (I/D)
DEC AXP 3k/500	43.1 fps	8.9 fps	150 mHz	?
HP 9k/750	74.7 fps	15.4 fps	66 mHz	256/256
Intel 486DX2	13.4 fps	3.3 fps	66 mHz	8/0
SGI Indigo	54.6 fps	11.7 fps		
Sun Sparc 10/30	38.1 fps	8.2 fps	36 mHz	16/20
Intel 486DX2	22.0 fps	5.5 fps	running v1.2 on Windows	
DEC AXP 3k/500	68.3 fps	16.3 fps	running DEC decoder	

Small sized images can play real time

Medium sized images are within a factor of 2

CIF format (352x288) in real time on Phillips 50 mHz PRPA VLIW processor

---

The Global Village

Internet distribution has been very successful

  First release in November 1991

  Many contributions by others: bug fixes, feature extensions, and performance improvements

  Over 1500 copies distributed per month (7/93)

  Over 8000 mpeg movies distributed per month (7/93)

Special acknowledgements...

  Toshihiko Kawai of Sony

  Tom Lane of the Independent JPEG Group

  Reid Judd of Sun Microsystems

  Todd Brunhoff of NVR

  Earl Killian of IDT, Inc.

  Chad Fogg of U. of Washington

  Paulo Villegas of Telefonica I+D

  Arian Koster of PTT

Future Plans

Integrate MPEG video stream into CMPlayer

Full-function VCR commands with frame drops
caused by decoder CPU and network limitations

Distribute portable, parallel MPEG video encoder

Experiment with other compression algorithms

MPEG-2, wavelets, 3D subband coding, ...

**Complete Berkeley Video-on-Demand server, meta
database, and archive server**

Conclusions

MPEG-1 decoding is not that difficult

... within a factor of 2 for CIF images

... low cost chips/boards will be released real soon

Implementation experience

Biggest problem is memory bandwidth, not CPU

IDCT is only 15% of time

Dithering and reconstruction over 50% of time

Playing movies on your screen is great fun, try it!