An Algorithm For Constant-Quality Compressed Video

UW CSE Industrial Affiliates Meeting
February 26, 2004

Michael F. Ringenburg
Richard E. Ladner
Eve A. Riskin

The Quick Introduction to Video Compression

- 4 Steps:
  - Motion compensation.
    - Outputs motion vectors, which predict the frame from the previous frame.
    - Actual - Predicted = Residual
  - Transform the residual.
    - Concentrates the energy (information).
    - Lossy encoding of the transformed residual.
  - Decoding.
    - Generates an approximation of the original frame.

Motion Compensation

Residual Examples (Boosted)

Transform Example (DCT)
The Quick Introduction to Video Compression

- 4 Steps:
  - Motion compensation.
  - Outputs motion vectors, which predict the frame from the previous frame.
  - Actual - Predicted = Residual
  - Transform the residual.
  - Concentrates the energy (information).
  - Lossy encoding of the transformed residual.
  - Quantization vs. Embedded Coding
  - Decoding.
  - Generates an approximation of the original frame.

Embedded Bit Plane Coding

- The n-th bit plane consists of the n-th bit of each coefficient.
- Each bit plane is compressed with an embedded, lossless code.
- Send the bit planes in order of significance.
  - The most important information is sent first.
- Can terminate the stream between or within bit planes.

The GTV Coder

- An embedded bit plane video coder developed at the University of Washington.
  - Written by Jeff West, Gidon Shavit and Michael Ringenburg, with Richard Ladner and Eve Riskin.
  - Motion Compensation on 16 x 16 blocks
  - 8 x 8 Discrete Cosine Transform
  - Lossy coding based on the GT-DCT image coder developed by Ed Hong with Richard Ladner and Eve Riskin
  - All results in this talk use the GTV coder

Interframe Bit Allocation

- Given: An embedded coder, a maximum bit budget $B$ and a video with $F$ frames.
- How many bits $b_i$ should we allocate to each frame $f_i$ in order to achieve constant quality video?
- Formally, our constraint is:
  $$\sum_{i=1}^{F} b_i \leq B$$

Why Is This Important?

- Some frames require more bits (are harder) to encode, because …
  - Scene changes prevent good motion compensation.
  - As do lighting changes.
  - And, to a lesser extent, dramatic actions.
- For example, set your TiVo to its lowest quality/highest compression setting. Watch the scene change blockiness at playback.
Why Is This Difficult?
- Motion Compensation!!!
  - The quality of frame \( n \) is a function of the allocation to frame \( n \), and the quality of frame \( (n-1) \).
  - Quality of frame \( (n-1) \) is a function of the allocation to frame \( (n-1) \) and the quality of frame \( (n-2) \). Et cetera.
  - Thus, the quality of frame \( n \) is a function of \( n \) variables.
- The quality function (a.k.a. rate-distortion curve) is different for every frame.
- The rate-distortion curves are not monotonic.
  - Approximately monotonic, though.

MultiStage Algorithm
- A global allocation algorithm (allocates across the entire video).
- Alternates between two stages which provide feedback to each other.
  - The Constant Quality Stage
  - The Target Allocation Stage
- After every stage, check if we have constant quality and the target allocation.

Example, Step 1
Constant Quality
![Graph showing Bytes and PSNR for different frames with T=2000]

Example, Step 2
Target Allocation
![Graph showing Bytes and PSNR for different frames with T=1000]

Example, Step 3
Constant Quality
![Graph showing Bytes and PSNR for different frames with T=800]

Example, Step 4
Target Allocation
![Graph showing Bytes and PSNR for different frames with T=1000]
Conclusions

- Developed two new global interframe bit allocation algorithms for embedded video coders
  - MultiStage algorithm
  - Ratio algorithm (not presented here)
- MultiStage has the lowest variance, and the highest minimum PSNR. Nearly constant quality.
- Full paper at: