CSE 490 GZ
Introduction to Data Compression
Winter 2002

Video Compression

Human Perception of Video

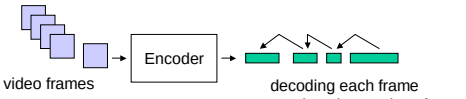
- 30 frames per second seems to allow the visual system to integrate the discrete frames into continuous perception.
- If distorted, nearby frames in the same scene should have only small details wrong.
  - A difference in average intensity is noticeable
- Compression choice when reducing bit rate
  - skipped frames cause stop action
  - lower fidelity frames may be better

Applications of Digital Video

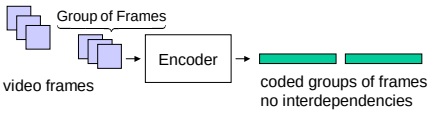
- Teleconference or video phone
  - Very low delay (1/10 second is a standard)
- Live Broadcast Video
  - Modest delay is tolerable (seconds is normal)
  - Error tolerance is needed.
- Video-in-a-can (DVD, Video-on-Demand)
  - Random access to compressed data is desired
  - Encoding can take a lot of time
- Decoding must always be at at least the frame rate.

Video Encoding

Frame-by-Frame coding



Group-of-Frames coding



Coding Techniques

- Frame-by-frame coding with prediction
  - Very low bit rates
  - low delay
  - Not error resilient
- Group-of-frames coding
  - Higher bit rates – within a group prediction is used
  - Error resilient
  - Random Access
  - Higher delay

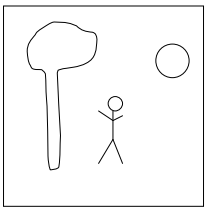
Digital Video Data

- CCIR 601 (4,2,2 scheme)
  - 13.5 MHz sample rate for luminance channel
  - 6.75 MHz sample rate for each of two chrominance channels
  - 8 bits per sample is a bit rate of 27 x 8 = 216 Mb per second
  - MPEG-SIF – ½ sample rate for luminance and ¼ for chrominance – 81 Mb per second
- CIF (Common Interchange Format)
  - 288 x 352 pixels per frame for luminance channel
  - 144 x 176 pixels per frame for each of two chrominance
  - 8 bits per pixel and 30 frames per second gives 48.7 Mb per second
  - QCIF (Quarter - CIF) is ¼ the data or 12.2 Mb per second.

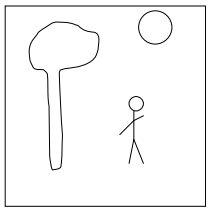
High Compression Ratios Possible

- Nearby frames are highly correlated. Use the previous frame to predict the current one.
- Need to take advantage of the fact that usually objects move very little in 1/30 th of a second.
  - Video coders use motion compensation as part of prediction

Motion Compensation

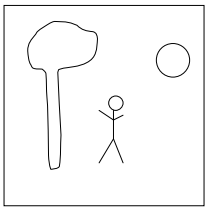


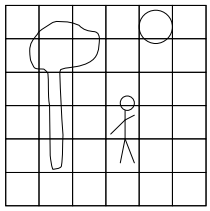
Previous Frame



Frame

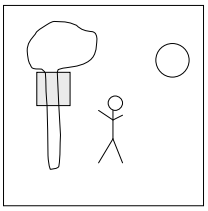
Block Based Motion Compensation

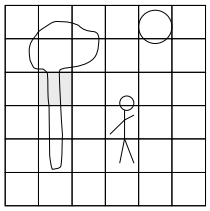




motion compensation blocks

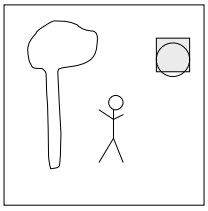
Motion Vectors

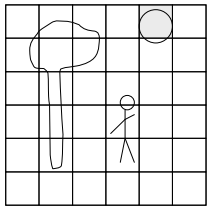




motion vector = (0,0)

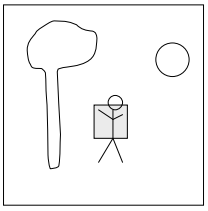
Motion Vectors

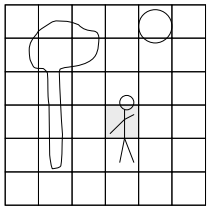




motion vector = (20,5)
20 down and 5 to right

Motion Vectors





motion vector = (0,-6)
0 down and 6 to left

Motion Compensation

- For each motion compensation block
  - Find the block in the previous frame that gives the least distortion.
  - If the distortion is too high then code the block independently. (intra block)
  - Otherwise code the difference (inter block)

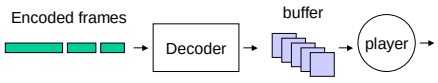
Issues

- Distortion measured in squared error or absolute error
  - Absolute error is quicker to calculate
- Distortion measured from raw or compressed previous frame
  - Compressed is better because that is what the decoder has
- Block size
  - Too small then too many motion vectors
  - Too large then there may be no good match
- Searching range to find best block
  - Too large a search range is time consuming
  - Too small then may be better matches
  - Prediction can help.

Rate Control

- After decoding, uncompressed stream is sent to a player.
  - We don't want to store the uncompressed stream
- Rate Control Problem
  - Frames are compressed at different rates
  - Solutions
    - Buffer the uncompressed frames for constant frame rate output.
    - Encode frames using a rate control algorithm to minimize the problem

Rate Control



- The buffer is emptied at a constant rate 30 frames per second.
- The buffer is filled at a variable rate. But its average rate is 30 frames per second.

Rate Control Situations

- Encoded data can be requested quickly (DVD or other storage medium)
  - Frames can always be decoded in time less than the 30 frames per second.
  - Decoder requests encoded frames to keep the buffer from overflowing.
- Encoded data received without request (Streaming media on the internet)
  - Encoder should anticipate what the decoder and buffer will be doing and encode using a rate control algorithm.

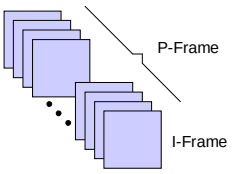
Simple Rate Control Algorithm

- If the buffer would overflow then compress the frames less.
- If the buffer would underflow then compress frames more.
  - Extreme case – skip frames to catch up. The decoder on not receiving a frame will send of copy of the current frame to the player. This is what causes the stop action or gaps in video.
  - If frame arrive too late to be played the decoder will throw it away.

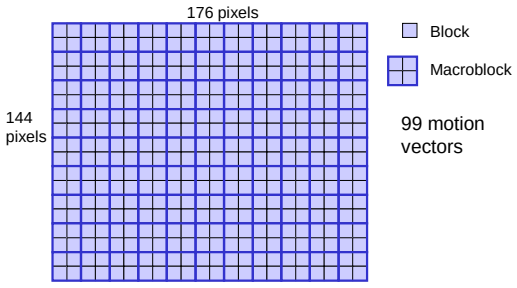
H.261

- Application – low bit rate streaming video
- Frame-by-frame encoder
- DCT based with 8x8 coding block
  - Uses JPEG style coding
- Motion compensation based on 16x16 macroblocks.

H.261

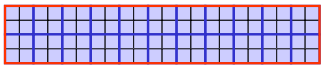


H.261(QCIF)

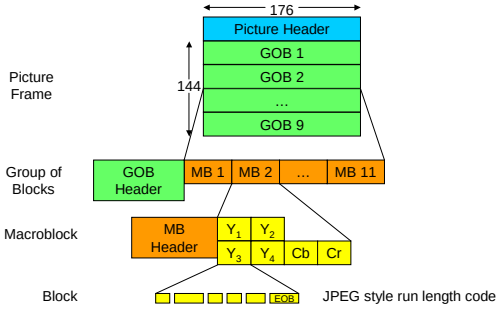


H.261

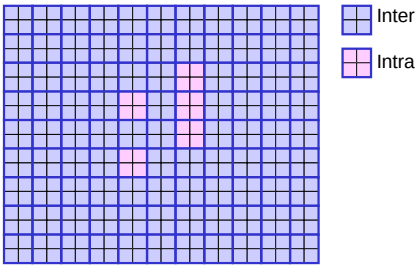
- Within a group of blocks (GOB) prediction is used with motion vectors for coding.



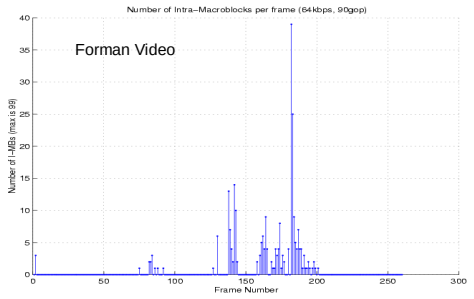
H.261 Organization



P-Frame



Intra-Macroblock Distribution

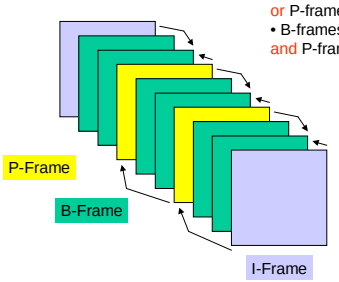


MPEG-1

- Application – Video coding for random access
- Group-of-frames encoder
- DCT based with 8x8 coding block
  - Uses JPEG style coding
- Motion compensation based on 16x16 macroblocks.
- Forward and Backward Prediction within a group of frames

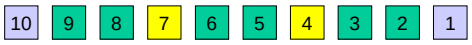
MPEG-1

- P-frames predicted by I-frames or P-frames
- B-frames predicted by I-frames and P-frames



Orders

Display Order



Coding/Decoding Order

Added delay is one frame time

MPEG-1 Notes

- Random access unit = Group-of-Frame
  - Called GOP for group-of-pictures
- Error resilient
  - B-frames can be damaged without propagation
- Added delay
  - Coding order different than display order
- Encoding time consuming
  - Suitable for non-interactive applications

Beyond MPEG-1

- MPEG-2
  - Application independent standard
- MPEG-4
  - Multimedia applications
  - Model based coding
- H.263
  - More error resilience
- 3-D Wavelet Coding
  - Third dimension is time
  - 3-D SPIHT has been implemented and claims have been made
  - Delay is large because GOP is large

Questions

- Can bit-plane coding replace JPEG style coding to improve video compression?
- Motion compensation is very time consuming and is not done for some real-time high bandwidth applications. Is there a good alternative to block based motion compensation?