INT 307
Multimedia Security System

Video Representation and Compression

Sichen.Liu@xjtlu.edu.cn

XJTLU

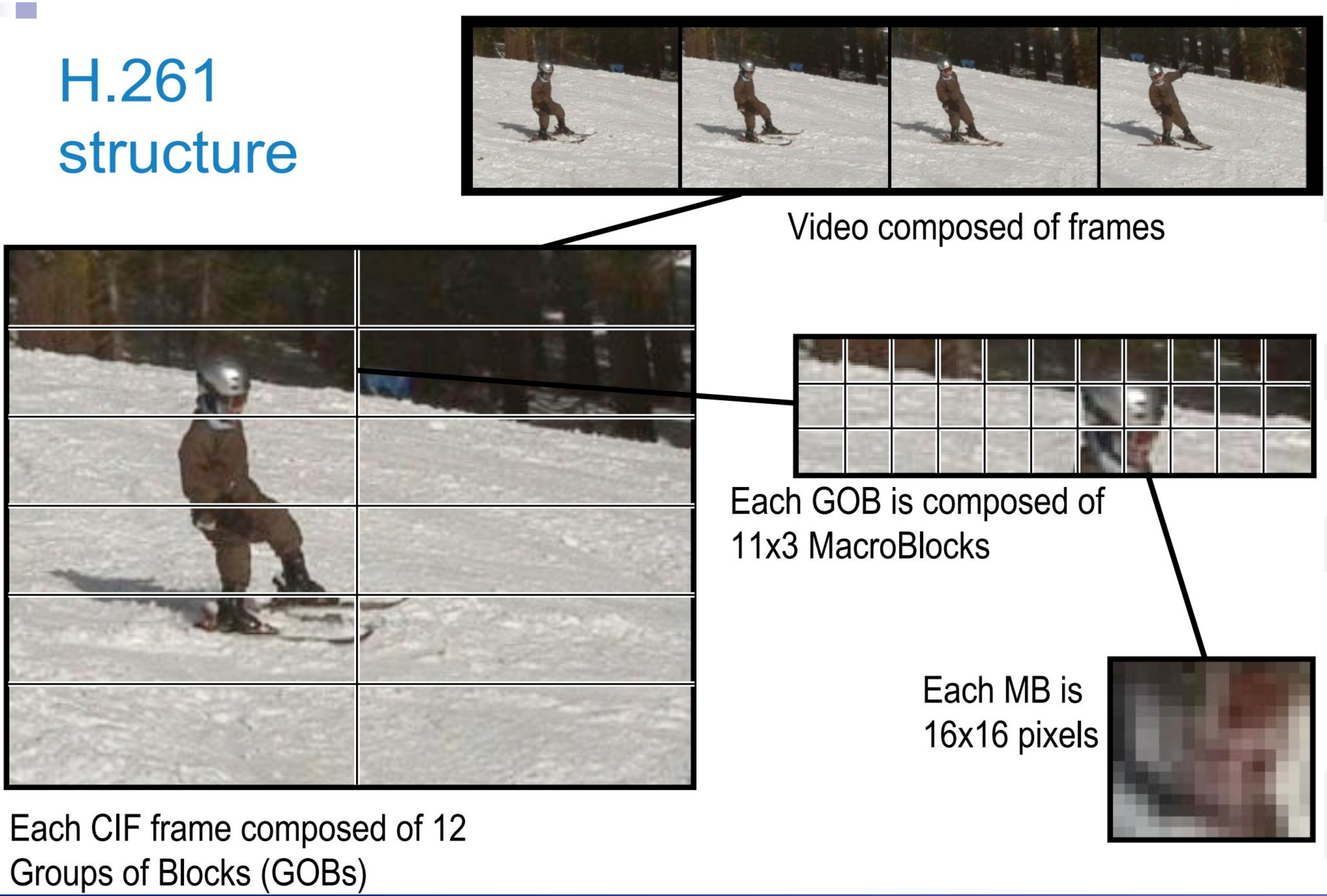
Xi'an Jiaotong-Liverpool University
西交利物浦大学

H.261 Video

- H.261 Compression was designed for videotelephony and video conferencing applications.
  - Developed by CCITT (now ITU-T) in 1988-1990
  - Intended for use over ISDN telephone lines, as part of the H.320 protocol suite.
  - Data rate was specified as multiples of 64Kb/s
- Goals for ISDN videotelephony
  - Low end-to-end delay.
  - Constant bit rate.

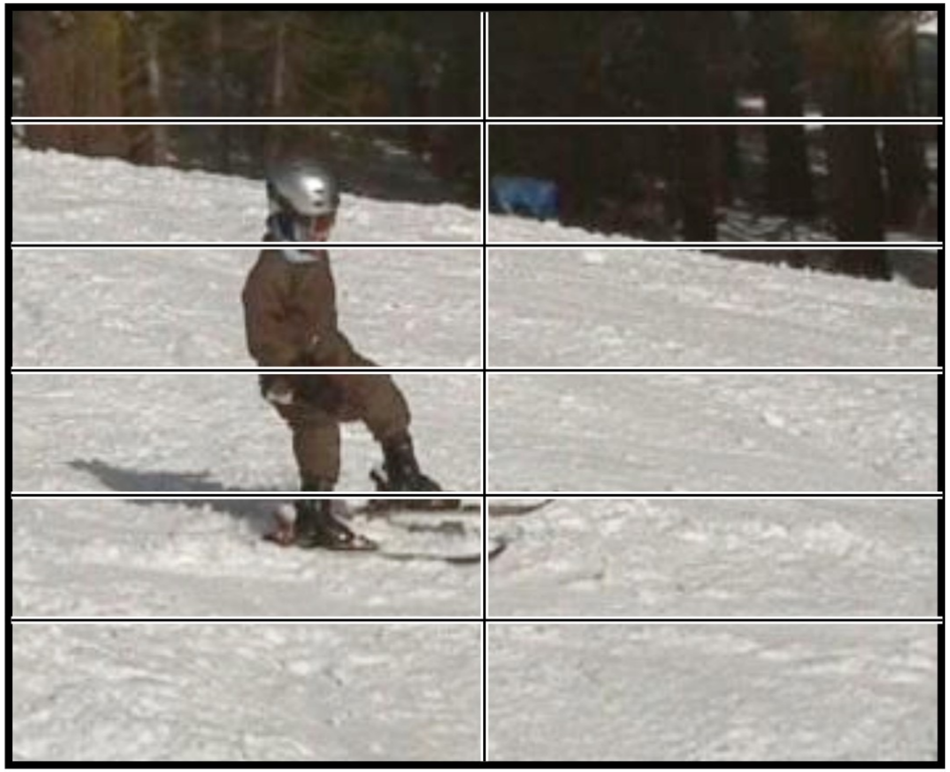
H.261 Structure

H.261 structure



CIF and QCIF Frame Formats

- CIF (Common Image Format): 352x288
- QCIF (Quarter CIF): 176x144



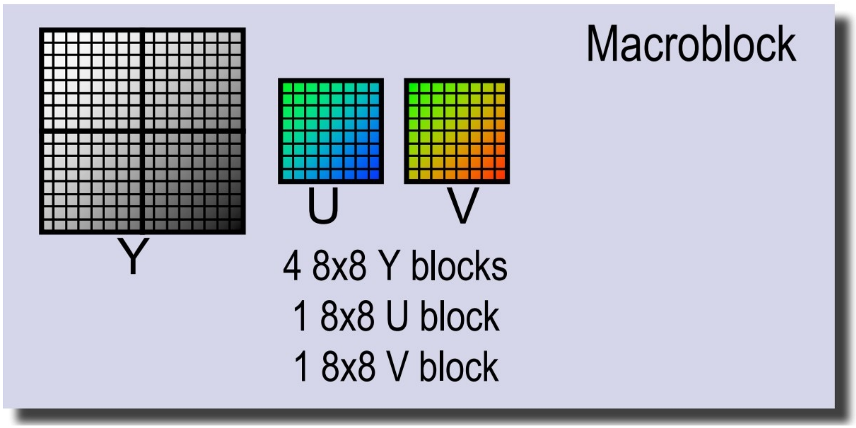
Each CIF frame (352x288 pixels) is composed of 12 Groups of Blocks (GOBs)

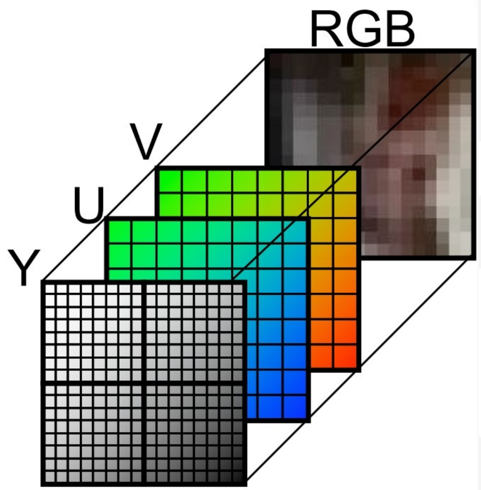
Each QCIF frame (176x144 pixels) is composed of 3 Groups of Blocks (GOBs)

GOB and MacroBlock format is identical in both frame formats.

Macroblocks

- Macroblock is basic unit for compression.
- Each macroblock is 16x16 pixels.
  - Represent as YUV 4:2:0 data.
  - 16x16 Luminance (Y) and subsampled 8x8 U, 8x8 V
- Represent this as 6 Blocks of 8x8 pixels





Macroblock coding

- Three ways to code a Macroblock
  - Don't.
    - If it hasn't changed since last frame, don't send it.
  - Intra-frame compression
    - Do DCT, Quantize, Zig-zag, Run-length encoding, and Huffman coding. Just like JPEG.
  - Inter-frame compression
    - Calculate difference from previous version of same block.
    - Can use motion estimation to indicate block being difference can from a slightly different place in previous frame.
    - Same DCT/quant/Huffman coding as Intra.

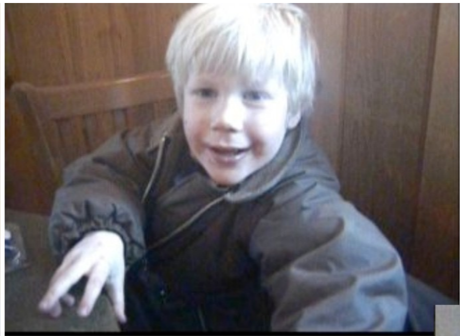
H.261 intra-frame compression

- Intra-coding of blocks is very similar to JPEG
  - DCT.
  - Quantize.
    - Unlike JPEG, H.261 uses the same quantizer value for all coefficients.
    - Feedback loop changes quantizer to achieve target bitrate.
  - Order coefficients in zig-zag order.
  - Run-length encode.
  - Huffman code what remains.

H.261 inter-frame compression

- Basic compression process is the same as intra-frame compression, but the data is the differences from the immediately preceding frame rather than the raw samples themselves
- Often the amount of information in the difference between two frames is a lot less than in the second frame itself

Frame 1



Frame 2

Difference:
Frame 2 - 1

Motion

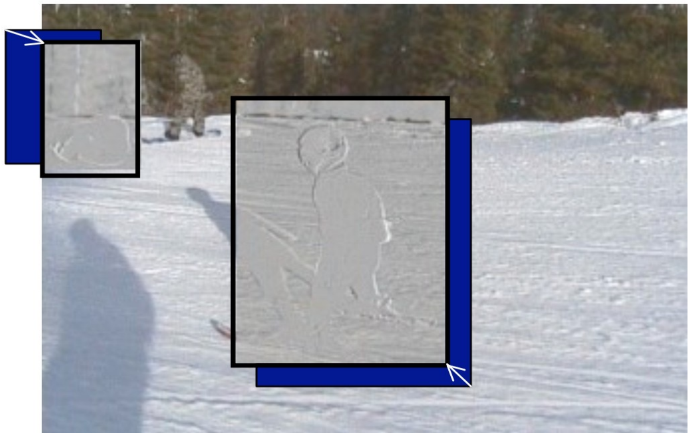
- Motion in the scene will increase the differences.
- If you can figure out the motion (where each block came from in the previous frame):
  - Encode the motion as a motion vector (two small integers indicating motion in x and y directions)
  - Encode the differences from the moved block using DCT + quantization + RLE + Huffman encoding

Frame 1

Frame 2

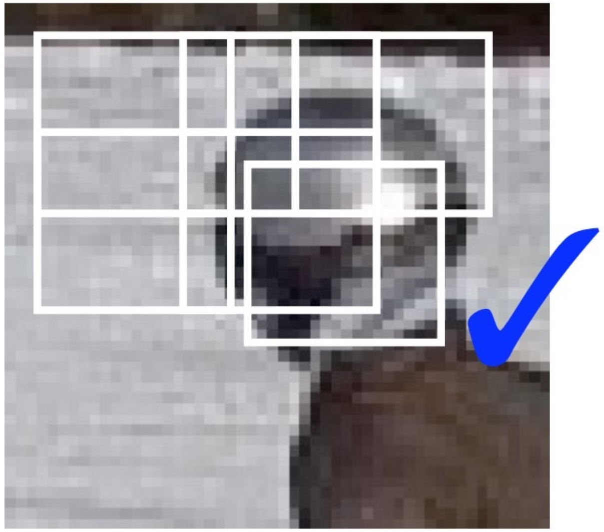
Coding from moved part of previous image can reduce the differences

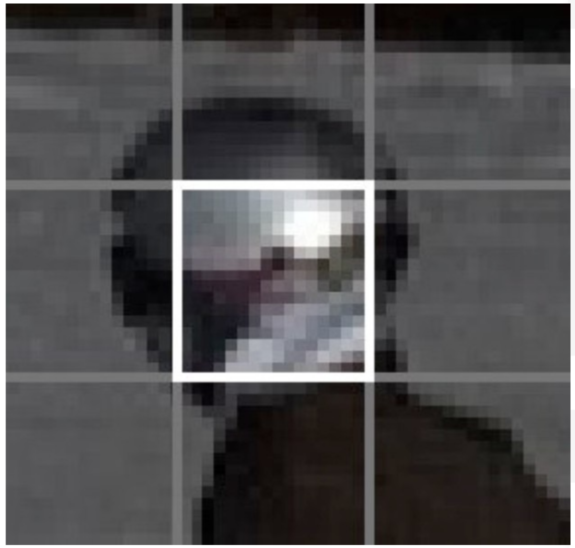
Frame 2 - 1
(lots of motion)



Motion Vector Search

- **Where did this Macroblock come from in the previous frame?**

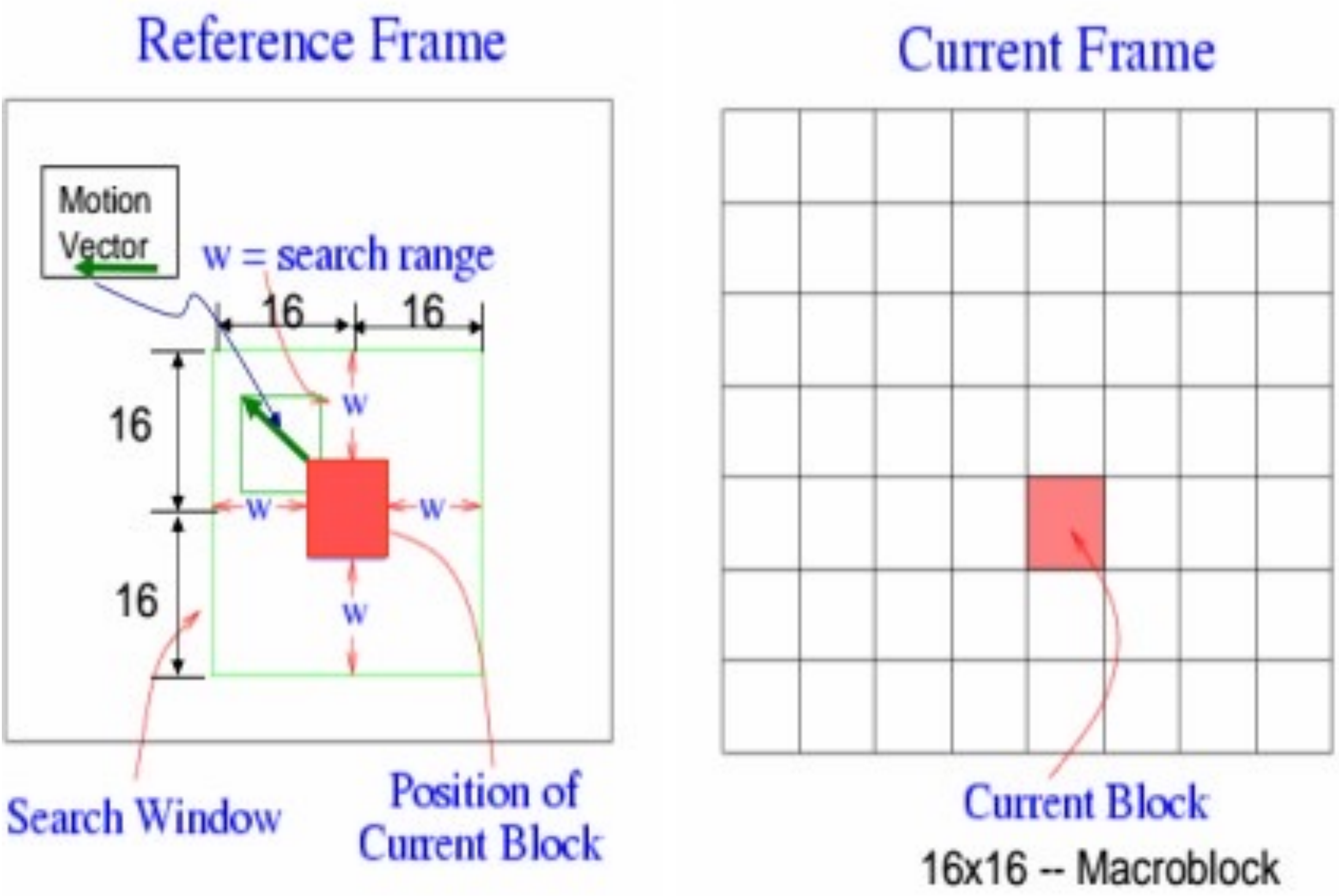




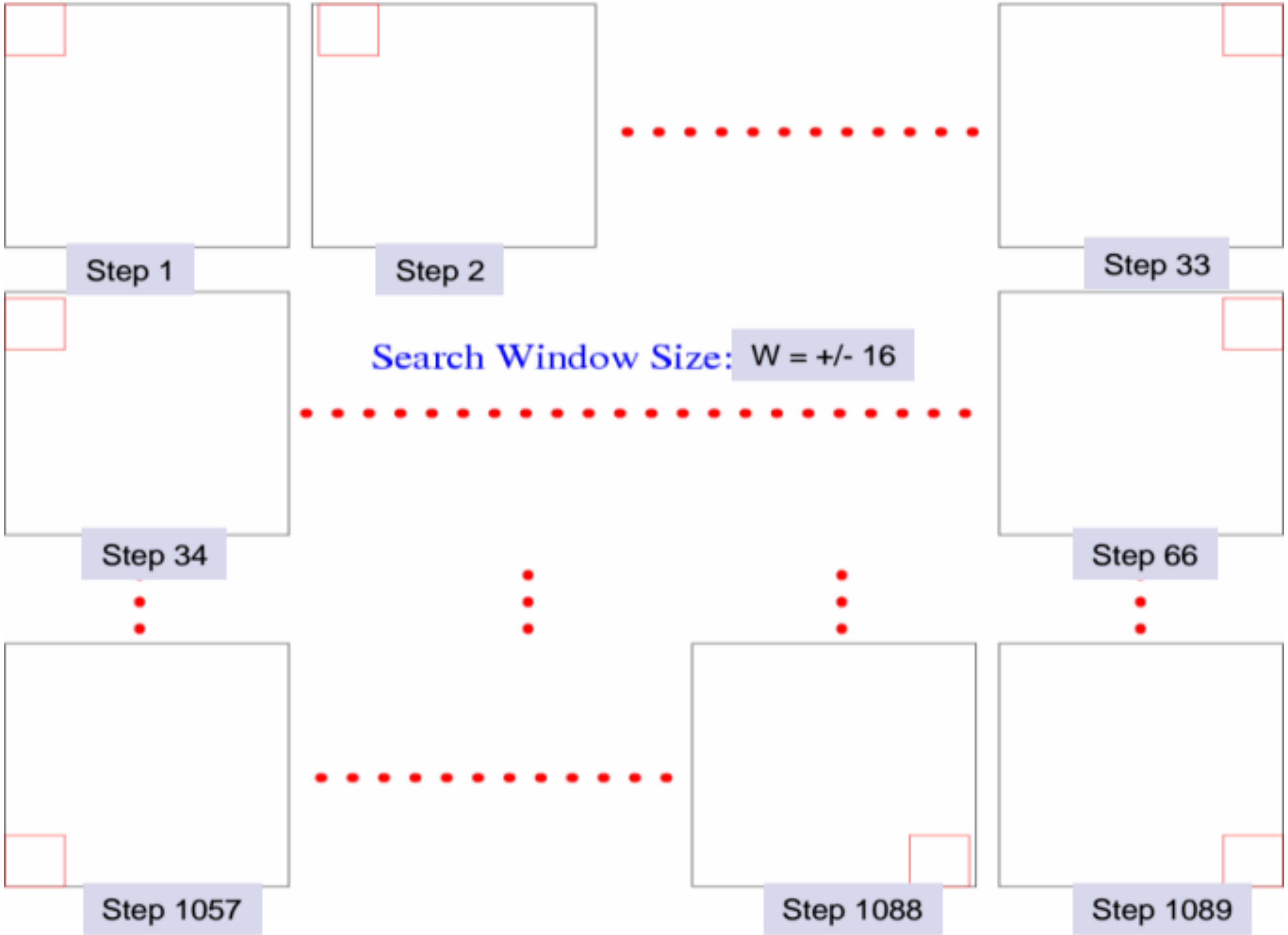
Block Based Motion Estimation

- Each picture is divided into block based subimage for motion estimation rather than using each pixel as a unit.
- 16x16 Macroblock is defined for Motion Estimation.
- To reduce computational and storage requirements, a limited search area is defined to the position around the current block
- The range of possible displacement in the X and Y directions is $[-16, +16]$.
  - 5 bits to specify the Horizontal or Vertical displacement
- The total motion vector is 10 bits/pixel.

Block Based Motion Estimation



Full Search Algorithm



Full Search Algorithm

- For each Macroblock (MB)
  - Total search steps = 33x33 = 1089
  - Total pixel operations = 1089x256=278784
  - For CCIR Rec.601 video (704 pixels x 576 pixels)
  - Pixel operation/frame = 441.6 million/frame
  - Pixel operation/second=11.0 billion/second (25 frame/second)
- Conclusion
  - Motion Estimation is Highly Computationally Intensive !!!!!!!

Fast Motion Estimation Techniques

- Full search motion estimation is computationally complex.
- Several **sub-optimum** fast search techniques have been developed.
- Many work on the assumption that block matching will improve monotonically as the search moves closer to the optimum point.
- Since they do not examine all of the candidate blocks, the choice of matching block might not be as good as that chosen by a full search. However, the quality-cost trade-off is usually worthwhile

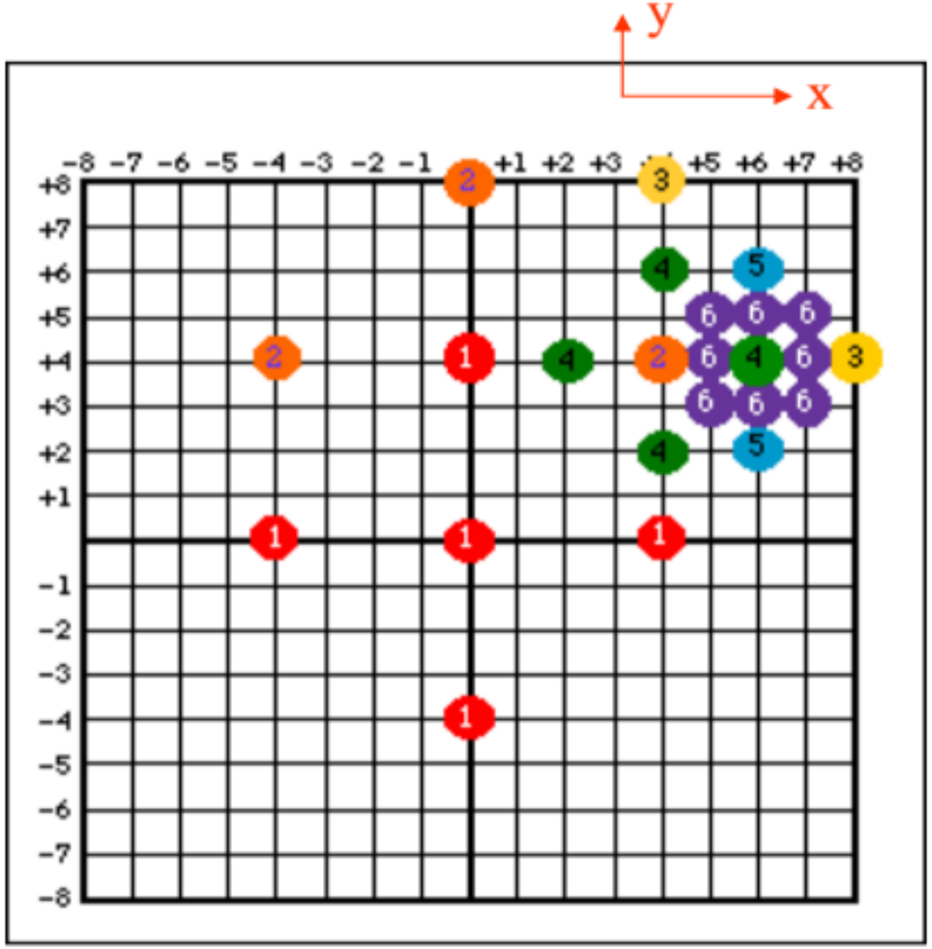
Two-dimensional logarithmic search (TDL)

- 1. The block at the center of the search area and four blocks at distance s from the center on the X and Y axes are searched for a best match.
- 2. If the position of best match is the center, have the step size (s/2). Else, if the best match is in one of the four outer positions, then it becomes the new centre point ([cx,cy]) for the next stage.
- 3. If the step size s is 1, then all nine blocks around the centre are examined, and the best match chosen for the target block. Otherwise, blocks at positions ([cx,cy], [cx+s,cy], [cx-s, cy], [cx,cy+s], and [cx,cy-s]) are searched, and the algorithm goes to stage 2.

- The processing of a search step relies on the previous search steps; therefore, the search steps cannot be performed in parallel

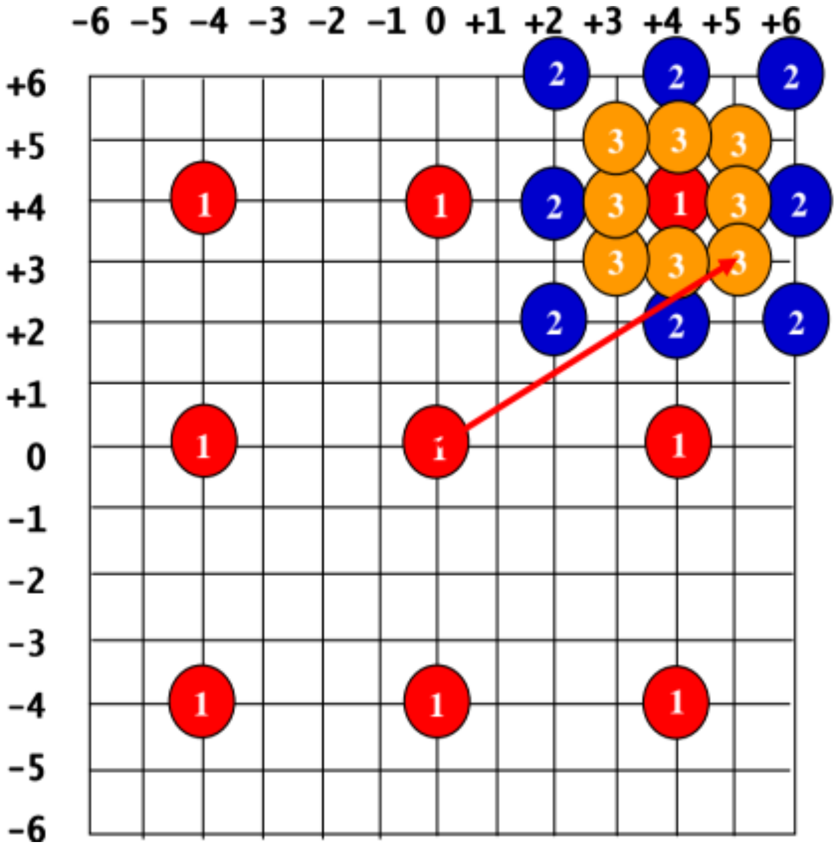
Two-dimensional logarithmic search (TDL)

- Note: The points [0,+4], [+4,+4], [+6,+4] are the minima at each stage, and finally [+7,+4] is chosen as the matching block. For step size s and search window size w, the step update algorithm is given by $s_0 = 2^{\lfloor \log_2 w \rfloor - 1}$



Three Step Search (TSS)

- Very similar to TDL search, and developed around the same time.
- The three step search tests eight points around the centre instead of four, with the position of minimum distortion becoming the new centre.
- After each stage the step size is reduced.



MPEG Family

- MPEG-1
  - Similar to H.263 CIF in quality
- MPEG: Motion Pictures Expert Group
- MPEG-2/MPEG-4/H.264
- Works well for HDTV too.
- Optimized for bit rates around 1-1.5Mb/s.
- Optimized for video resolutions
  - 352x240 pixels at 30 fps (NTSC)
  - 352x288 pixels at 25 fps (PAL/SECAM)

MPEG-1 Frame Types

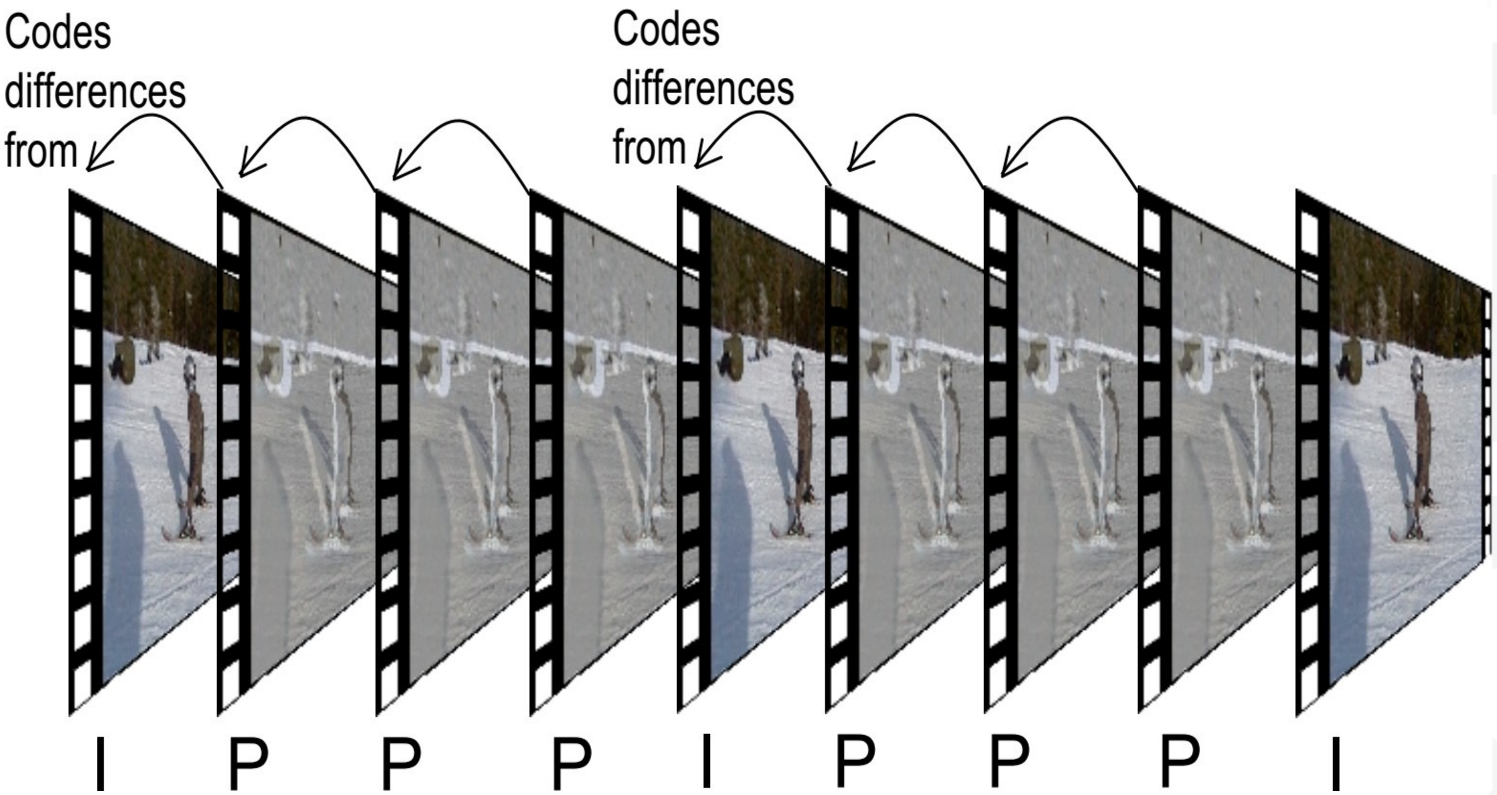
- Unlike H.261, each frame must be of one type.
- Three types in MPEG
  - I-frames (like H.261 intra-coded frames)
  - P-frames ("predictive", like H.261 inter-coded frames)
  - B-frames ("bidirectional predictive")

MPEG I-frames

- Similar to JPEG, except
  - Luminance and chrominance share quantization tables.
  - Quantization is adaptive (table can change) for each macroblock.
- An I frame and the successive frames to the next I frame (n frames) is known as a Group of Pictures.

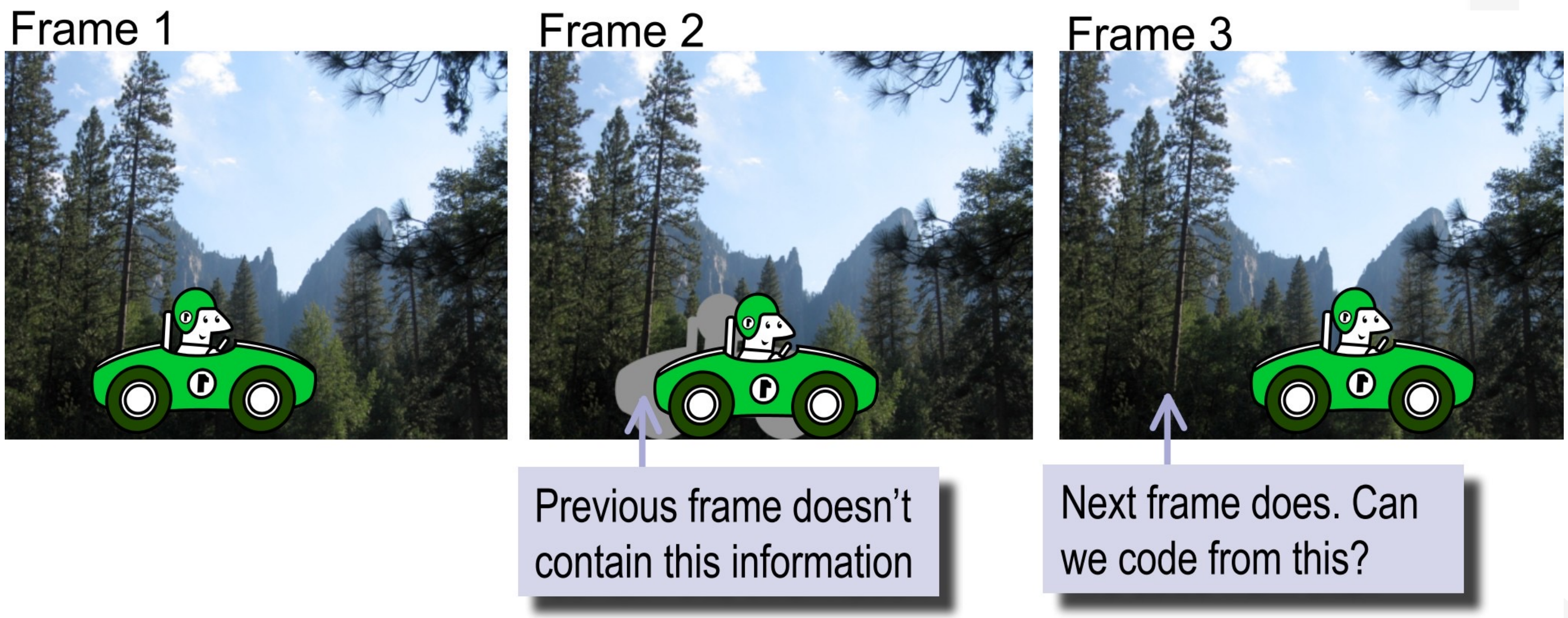
MPEG P-Frames

- Similar to an entire frame of H.261 inter-coded blocks.



Object occlusion

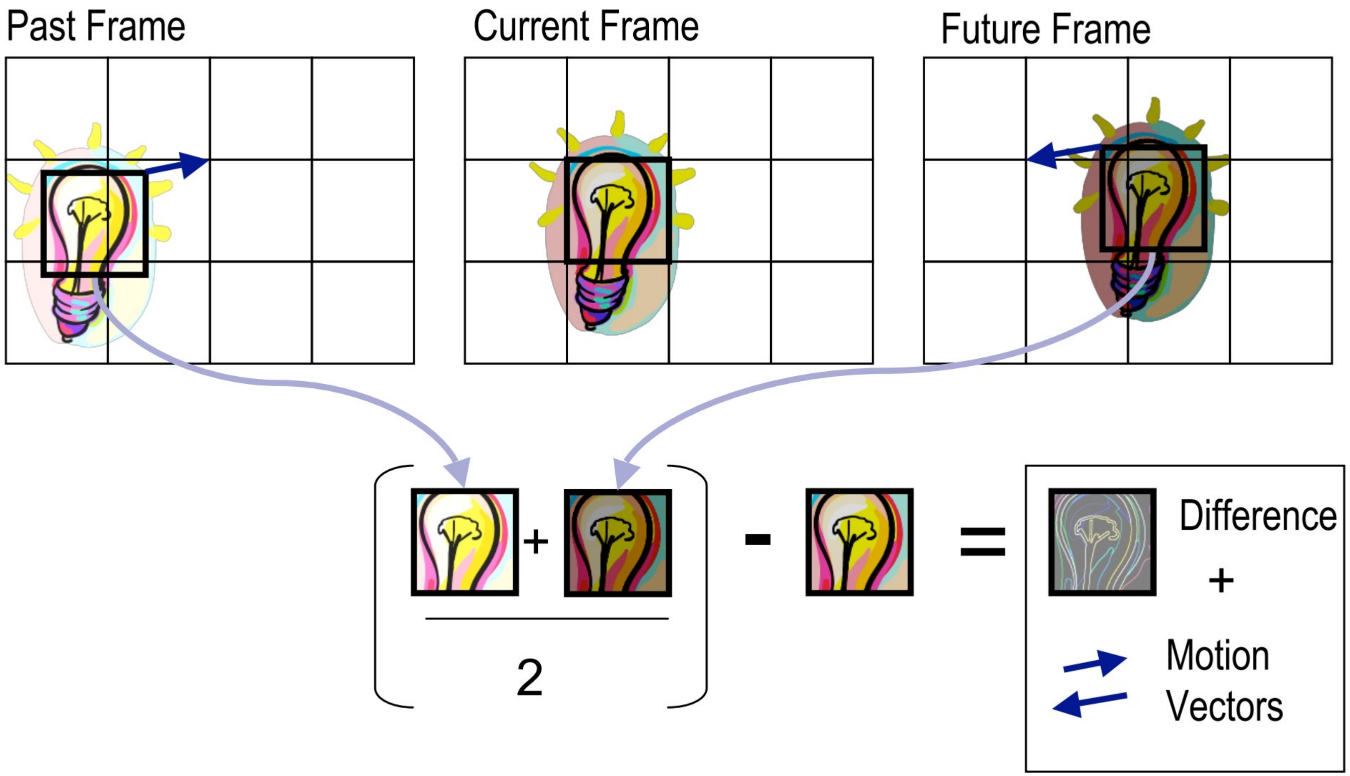
- Often an object moves in front of a background.
- P frames code the object fine, but can't effectively code the revealed background.



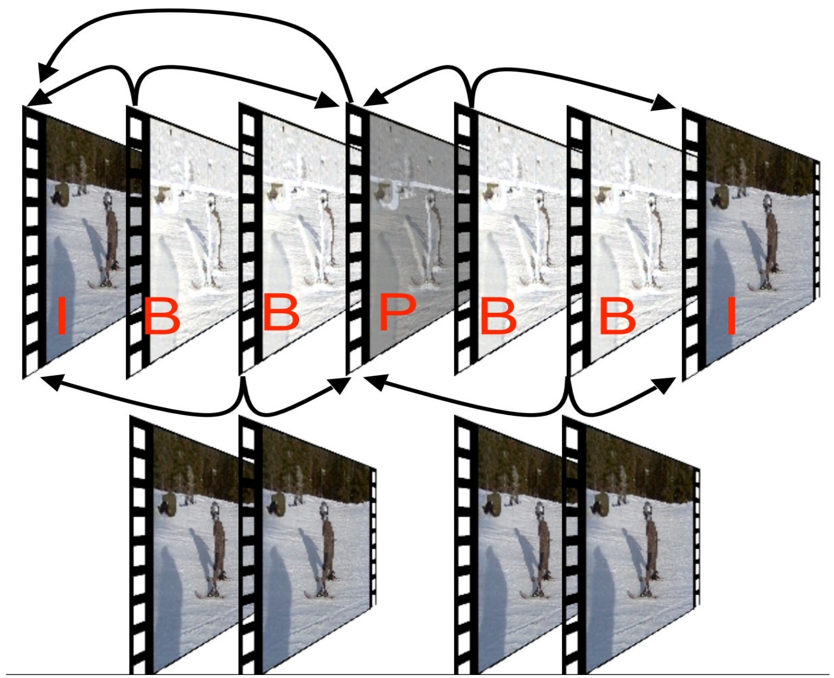
B-frames

- Bidirectional Predictive Frames.
- Each macroblock contains two sets of motion vectors.
  - Coded from one previous frame, one future frame, or a combination of both.
    - Do motion vector search separately in past reference frame and future reference frame.
    - Compare:
      - Difference from past frame.
      - Difference from future frame.
      - Difference from average of past and future frame.
    - Encode the version with the least difference.

B-frames: Macroblock averaging



Encoding Order



1. Encode I–frame 1
2. Store frame 2
3. Store frame 3
4. Encode P frame 4
5. Encode B frame 2
6. Encode B frame 3
7. Store frame 5
8. Store frame 6
9. Encode B frame 7
10. Encode B frame 5
11. Encode B frame 6

Transmission Order

- Frames are encoded out of order
- Need to be decoded in the order they're encoded.
    - Common to send out of order.

Eg: $I_1B_2B_3P_4B_5B_6I_7B_8B_9P_{10}B_{11}B_{12}I_{14}$

sent in the order

$I_1P_4B_2B_3I_7B_5B_6P_{10}B_8B_9I_{14}B_{11}B_{12}$

- Allows decoder to decode as data arrives, although it still has to hold decoded frames until it has decoded prior B frames before playing them out.

B-frame disadvantages

- Computational complexity.
  - More motion search, need to decide whether or not to average.
- Increase in memory bandwidth.
  - Extra picture buffer needed.
  - Need to store frames and encode or playback out of order.
- Delay
  - Adds several frames delay at encoder, need later frame.
  - Adds several frames delay at decoder holding decoded I/P frame, while decoding and playing prior B-frames that depend on it

B-frame advantages

- B-frames increase compression.
- Typically use twice as many B frames as I+P frames.

Type	Size	Compression
I	18KB	7:1
P	6KB	20:1
B	2.5KB	50:1
Average	4.8KB	27:1

Typical MPEG-1 values.

Really depends on video content.

THANK YOU

VISIT US

WWW.XJTLU.EDU.CN

FOLLOW US

@XJTLU

Xi'an Jiaotong-Liverpool University
西交利物浦大学

XJTLU | SCHOOL OF FILM AND TV ARTS