Video Coding

Trac D. Tran
ECE Department
The Johns Hopkins University
Baltimore MD 21218

Outline

- Video coding algorithms and standards: an overview
  - Main observations. Several simple video codecs
  - Motion-compensated DCT coding (MC-DCT)
  - Motion-compensated wavelet coding (MC-DWT)
  - 3D wavelet and subband coding
- Block motion estimation (BME) and motion compensation
  - Principle. Error measures. Optimality. Search strategies.
  - Examples of fast sub-optimal algorithms
  - Fast optimal algorithms: spiral search, successive elimination, projection matching...
  - Combination searches: hierarchical, spatial correlation, rate-constrained...
- Advanced topics in motion estimation / motion compensation
  - Global motion estimation
  - Advanced motion models: sub-pixel motion, affine motion, meshes, variable-block-size MEMC...

Main Observations

- Video signal is a sequence of still images or frames
- Correlation in video sequence
  - Temporal correlation: similar background with a few moving objects in the foreground
  - Spatial correlation: similar pixels seem to group together – just like spatial correlation in images
  - There is usually much more temporal correlation then spatial
- Motion model in video sequence
  - Natural motion
    - moving people & moving objects
    - translational, rotational, scaling
  - Camera motion
    - camera panning, camera zooming, fading

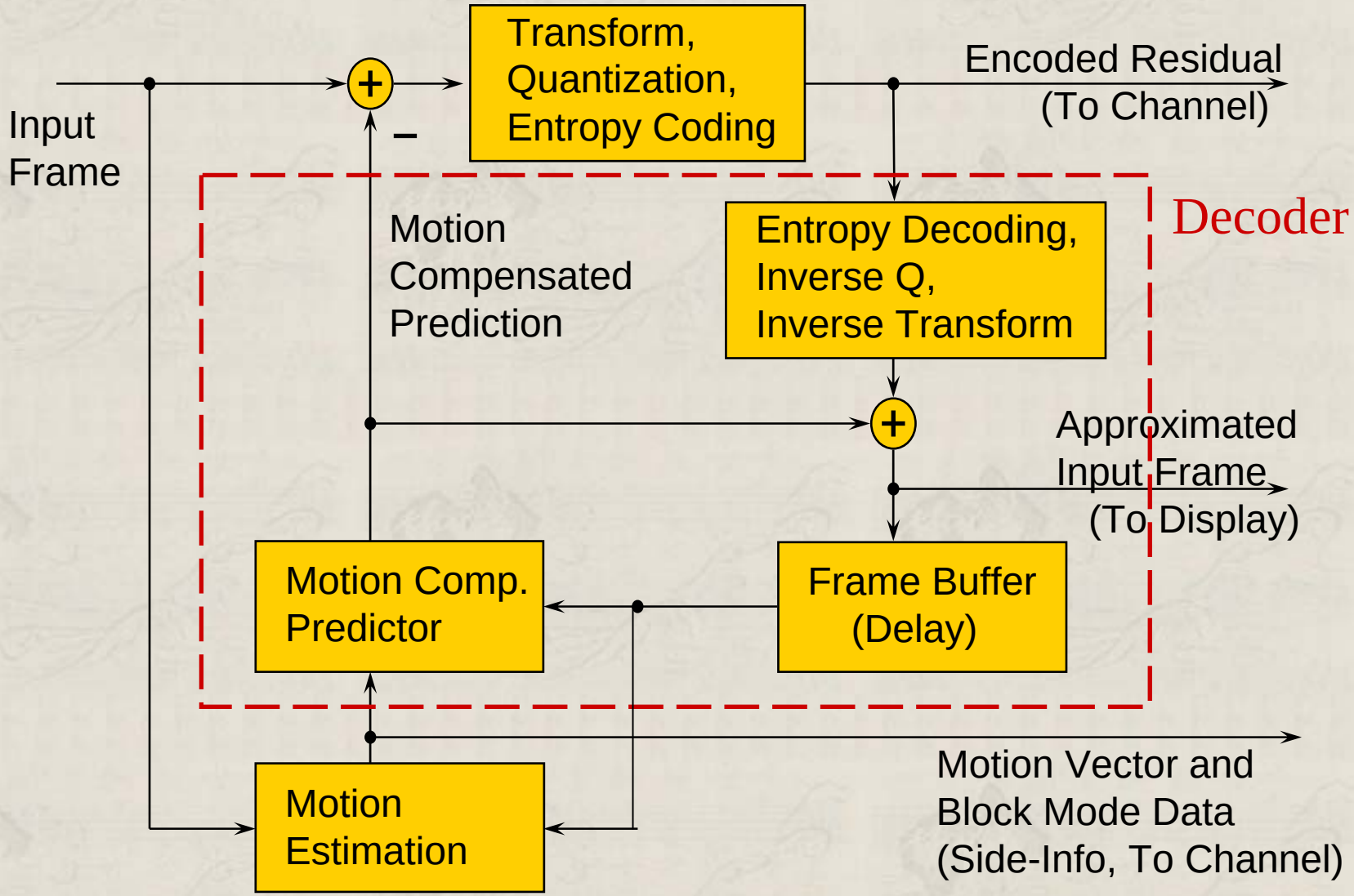
Simple Video Coders

- JPEG/JPEG2000 encodes every frame independently
  - Quite popular: Motion-JPEG, Motion-JPEG2000
  - Does not take into account any temporal correlation
  - Advantage: very simple and fast, no latency (frame delay), easy frame access for video editing
  - Disadvantage: low coding performance
- JPEG/JPEG2000 encodes frame difference
  - Requires very stationary background to be effective
  - Advantage: improves compression, low latency
  - Disadvantage: still low coding performance, open-loop design, quantization error accumulation from coder/decoder mismatch
- JPEG/JPEG2000 encodes frame difference – closed-loop
  - Advantage: no error accumulation, low latency, simple
  - Disadvantage: too simple motion model, still low compression

Motion-Compensated Framework

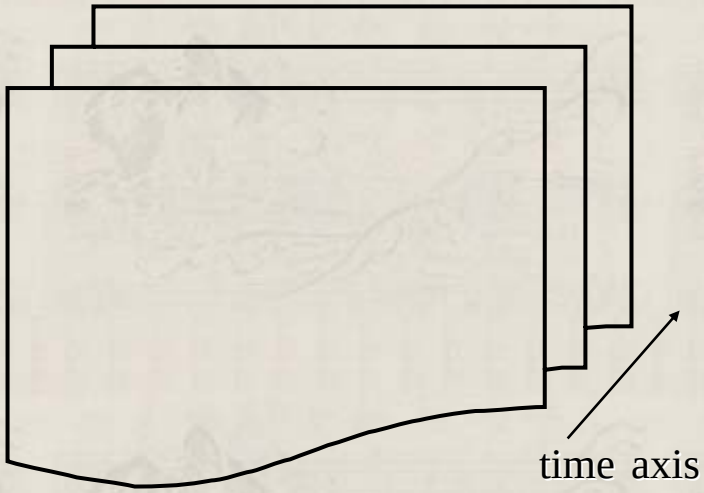
- Transform-based coding on motion-compensated prediction error (residue)
- Popular transformations: block DCT or wavelet
- Closed-loop DPCM to prevent error propagation (drifting)
- Usually employed block-translation motion model
- All international video coding standards are based on this coding framework
  - Video teleconferencing: H.261, H.263, H.263++, H.26L/H.264
  - Video archive & play-back: MPEG-1, MPEG-2 (in DVDs) MPEG-4

Typical MC Codec



Motion Estimation

- Goal: extract correlation between adjacent video frames to improve compression efficiency
- General problem statement:



Given the current frame $C(x, y)$
and the reference frame $R(x,y)$,
find functions $f(x, y)$ and $g(x, y)$ to
minimize

$$E = d\{C(x,y), R(f(x,y), g(x,y))\}$$

- Practical motion model: small objects or regions moving in translational fashion $f(x, y) = x - dx;\ g(x, y) = y - dy$
- Block-based motion estimation (BME) and compensation (BMC)

Intra- and Inter-Coding

- Inter-coding:
  - blocks of predicted motion error labeled P-block is encoded
- Intra-coding
  - any block that motion estimation fails to find a good match is labeled I-block and encoded as is (without any motion compensation)
- Conditional replenishment
  - when prediction error is low, no coding or decoding is necessary
  - we simply record the motion vector and replenish the block from the reference frame.

Motion Models

- Translation

$$\boxed{\begin{cases} f(x,y) = x - d_x \\ g(x,y) = y - d_y \end{cases}}$$

- Affine

$$\begin{cases} f(x,y) = a_{00}x + a_{01}y + d_x \\ g(x,y) = a_{10}x + a_{11}y + d_y \end{cases}$$

- Bilinear

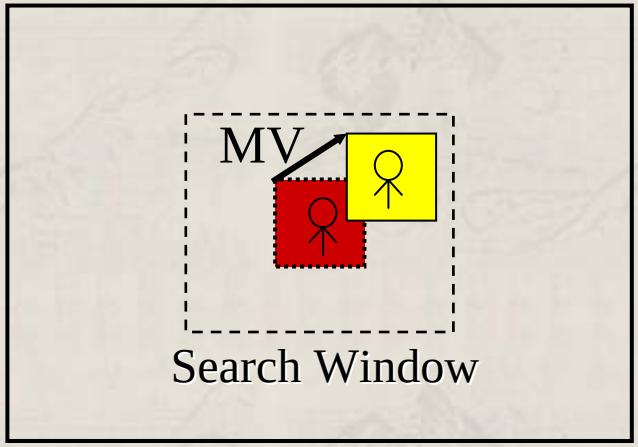
$$\begin{cases} f(x,y) = a_{00}x + a_{01}y + a_{02}xy + d_x \\ g(x,y) = a_{10}x + a_{11}y + a_{12}xy + d_y \end{cases}$$

- Perspective

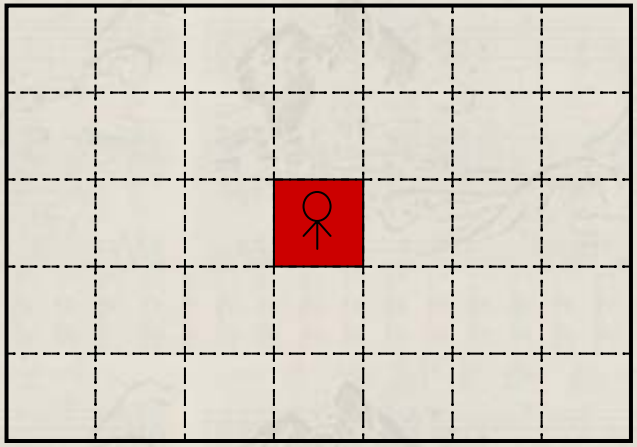
$$\begin{cases} f(x,y) = \dfrac{a_{00}x + a_{01}y + a_{02}}{a_{20}x + a_{21}y + a_{22}} \\ g(x,y) = \dfrac{a_{10}x + a_{11}y + a_{12}}{a_{20}x + a_{21}y + a_{22}} \end{cases}$$

Principle of BME

- Partition current frame into small non-overlapped blocks called macro-blocks (MB)
- For each block, within a search window, find the motion vector (displacement) that minimizes a pre-defined mismatch error
- For each block, motion vector and prediction error (residue) are encoded



Reference Frame



Current Frame

BME: Error Measure

- Sum of absolute differences

search range

$$SAD(dx,dy) = \sum_{i=0}^{Rx}\sum_{j=0}^{Ry}\left|C(x+i, y+j) - R(x+i+dx, y+j+dy)\right|$$

current block reference block

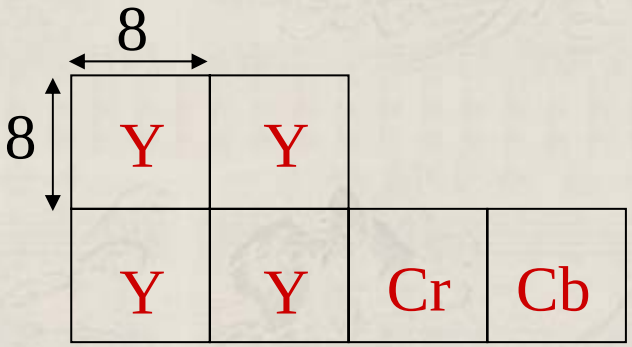
- Sum of squared errors (mean-squared error)

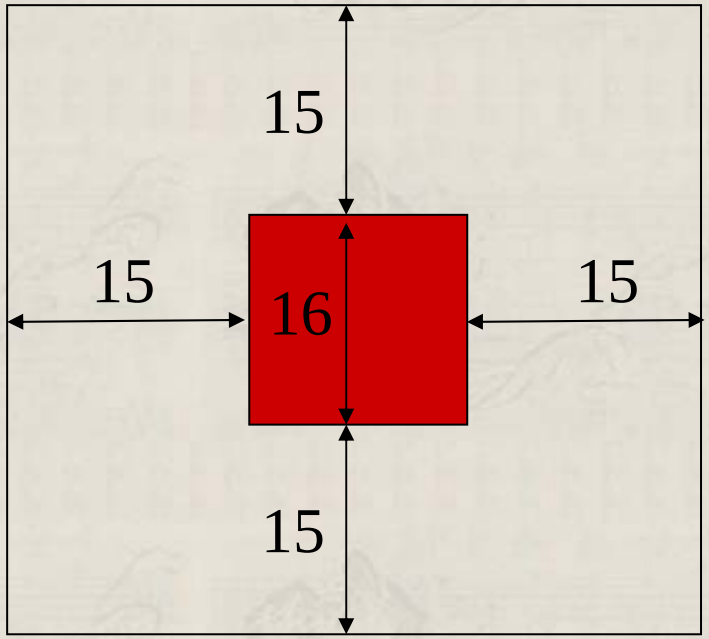
$$SSE(dx,dy) = \sum_{i=0}^{Rx}\sum_{j=0}^{Ry}\left|C(x+i, y+j) - R(x+i+dx, y+j+dy)\right|^2$$

- Discussions:
  - Other norms, correlation measure have been tested
  - Approximately same coding performance
  - SAD is less complex for some hardware architectures

Common Setting

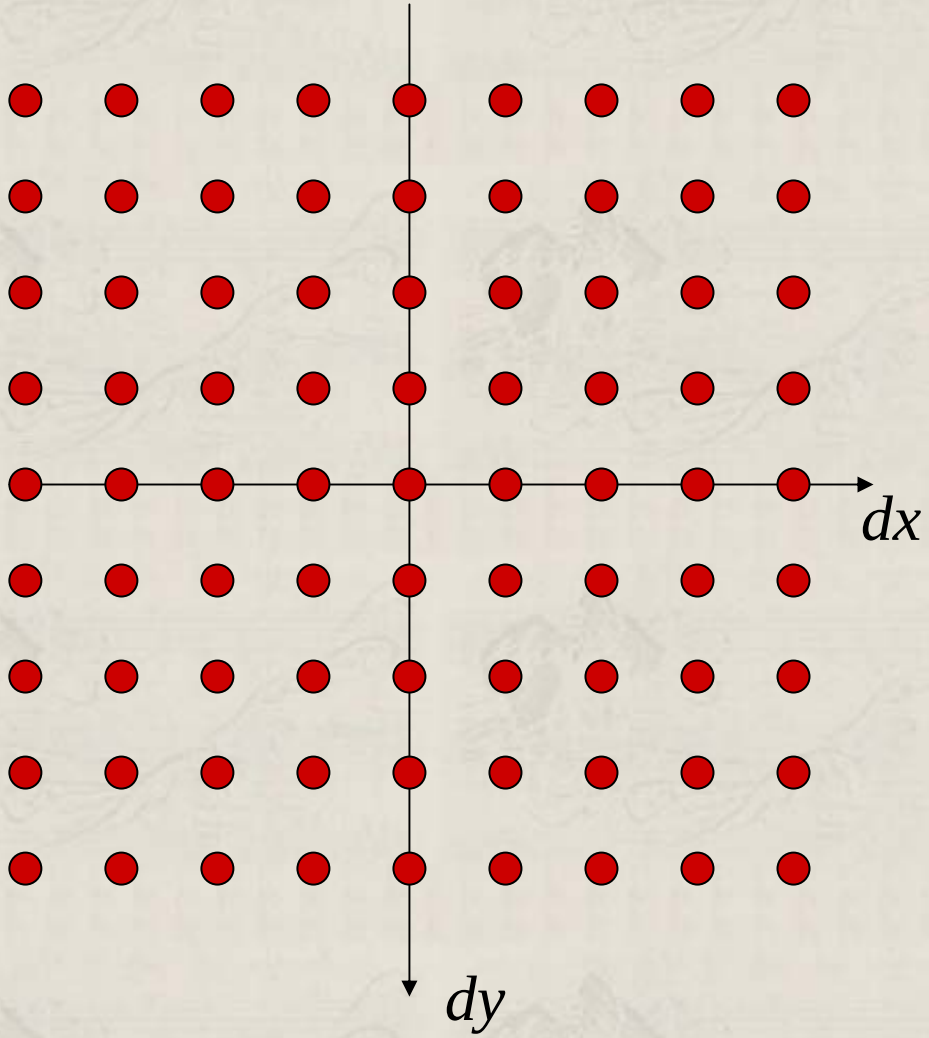
- Macro-block
  - Luminance: 16x16, four 8x8 blocks
  - Chrominance: two 8x8 blocks
  - Motion estimation only performed for luminance
- Motion vector range
  - [-15, 15]





Search Area

Search Strategies I



Exhaustive Search

- All possible MV candidates within the search range are investigated
- Very computationally expensive
- Optimal, highly regular, parallel computable

Search Strategies II

Divide and Conquer

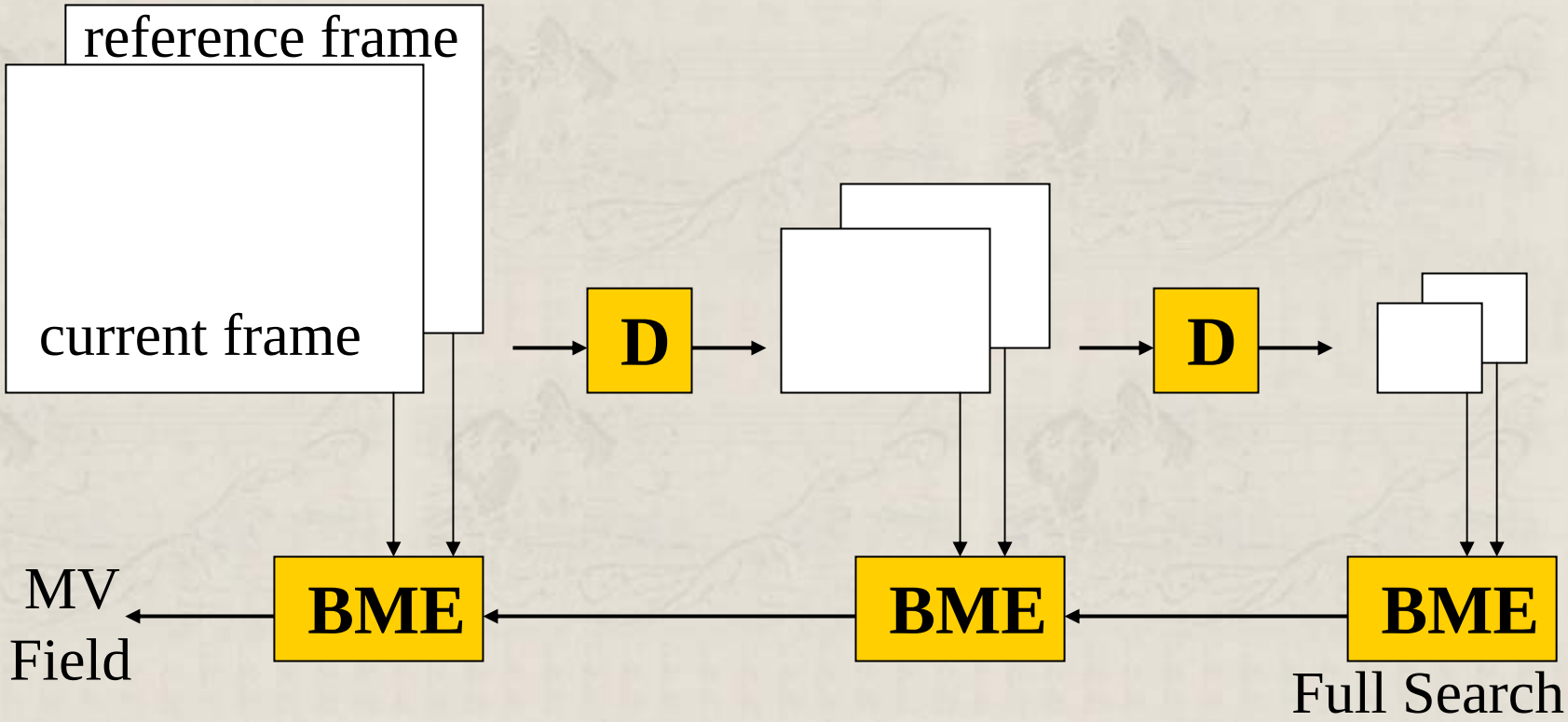
- Sampling the search space
- Divide search space into regions
- Pick a center point for each region
- Perform more elaborate search on the region with best center

Gradient Search

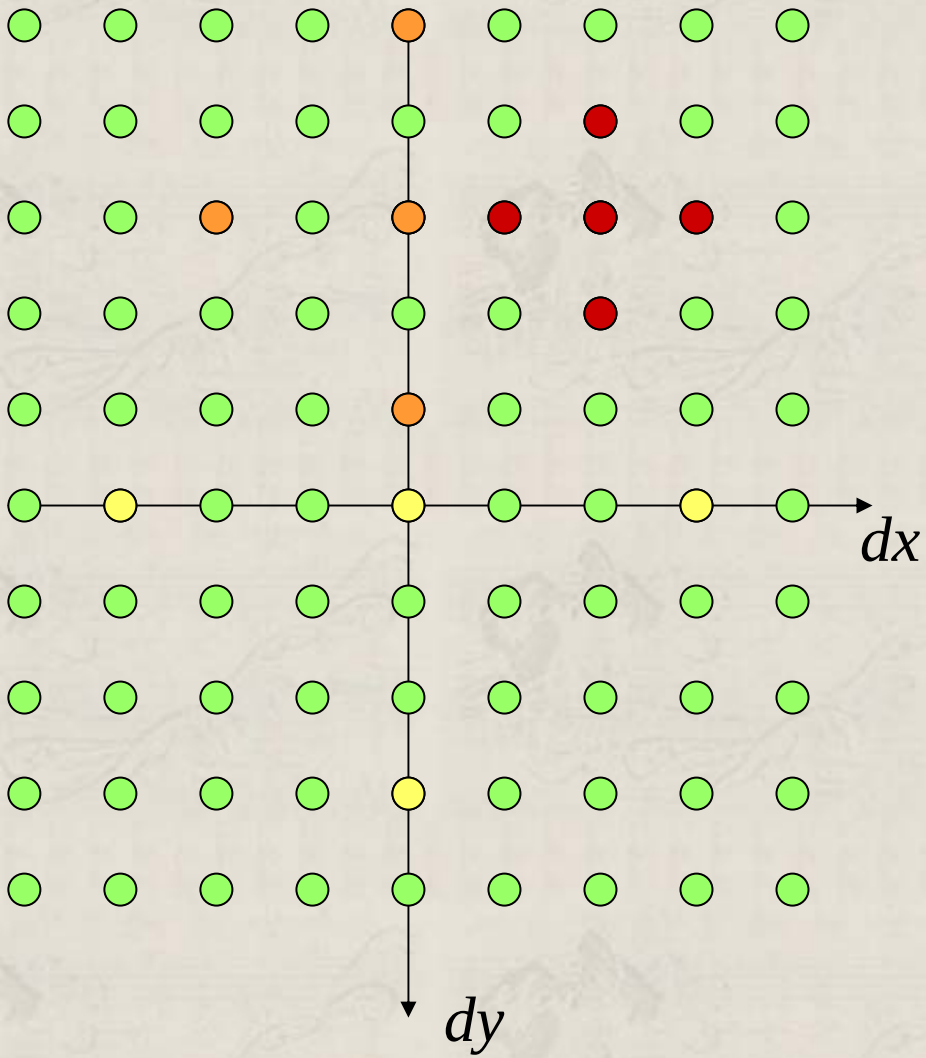
- Sampling the search space
- At every stage, investigate a few candidates
- Move in the direction of the best match
- Can adaptively reduce the displacement size for each stage

Search Strategies III

Multi-resolution or Hierarchical Search



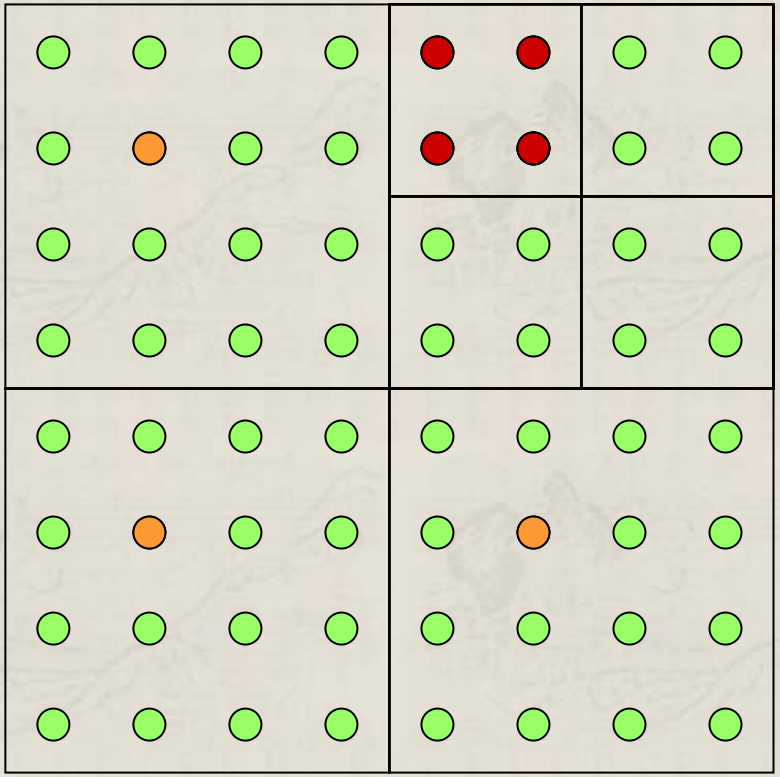
Fast BME Algorithm: Example I



2D Log Search

- Use + pattern at each stage (5 candidates)
- Move center to best match
- Reduce step size at each stage
- Fast, reasonable quality
- Similar algorithms: three-step-search, four-step-search, cross search...

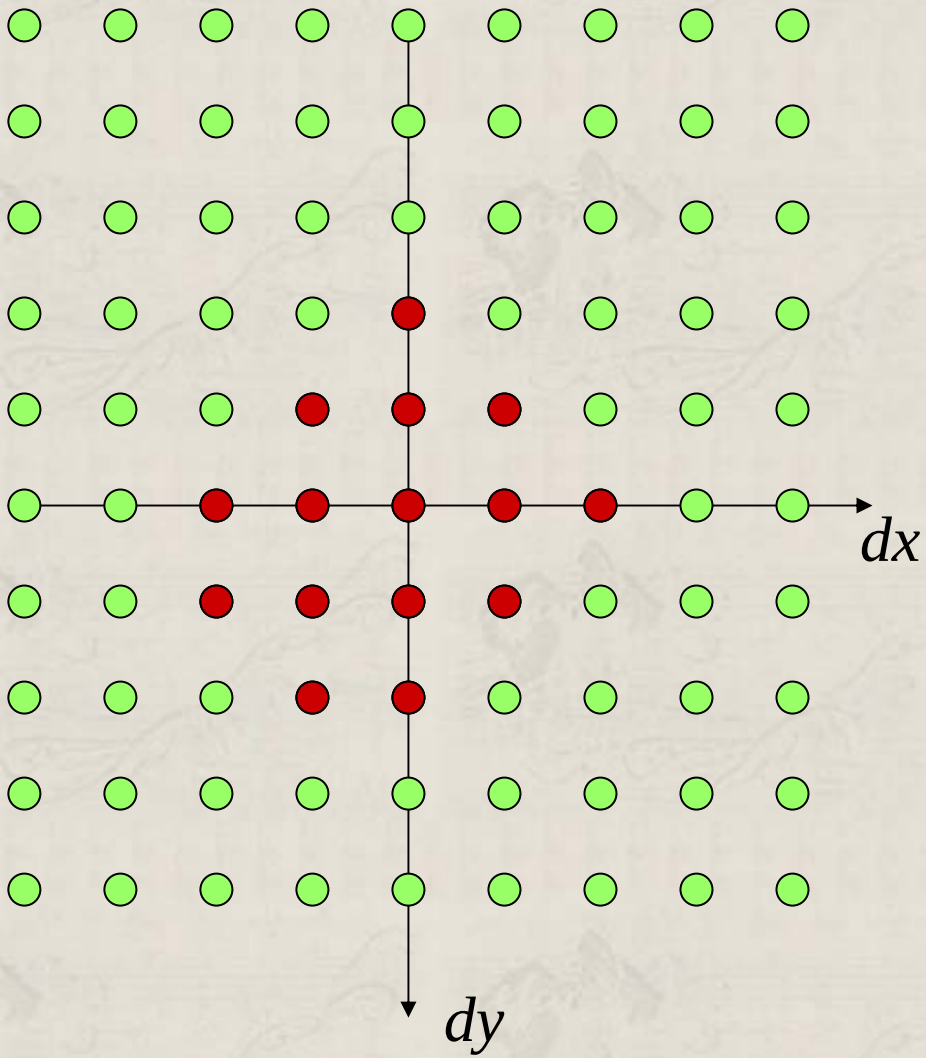
Fast BME Algorithm: Example II



Binary Search

- Divide-and-conquer
- Move search to region with best match
- Finer search at later stages

Optimal BME: Spiral Search



Spiral Search

- Early termination
- Start with a predicted search center, default=(0,0)
- Spiral search around center in diamond or square pattern
- Keep track of best match so far; update whenever a better candidate is found
- MPEG-4, H.263+

Optimal BME: Partial Matching

- Triangle inequality

partial sums

$$MAD(dx,dy) = \sum_{block} |C - R| \geq \left| \sum_{block} (C - R) \right| = \left| \sum_{block} C - \sum_{block} R \right|$$

- Strategy
  - Compute partial sums for macro-blocks in both current and reference frame
  - Eliminate candidates based on partial sums
  - Nice partial sums: row projections, column projections!
  - Avoid full 2D matching, translate the problem into 1D matching
  - Can be combined with spiral search and early termination

Spatial Correlation Based BME

- Exploit intra-frame spatial correlation to narrow down search space for BME
- Blocks covering the same object should move together
- MVs from causal neighboring blocks can be used to predict MV for the current block

2	**3**	**4**
1	**C**	

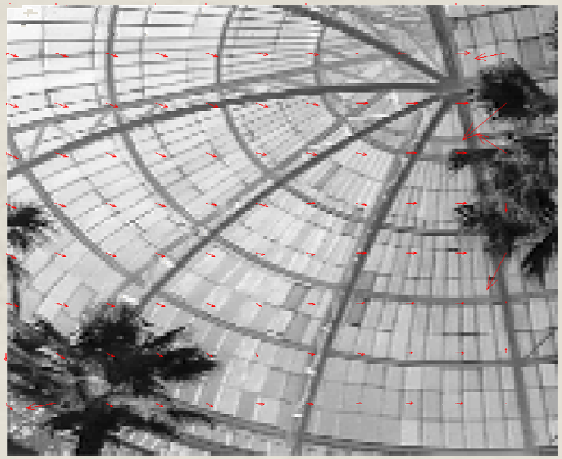
- Reduce bit-rate for MV encoding: regularize the MV field
- Reduce BME complexity

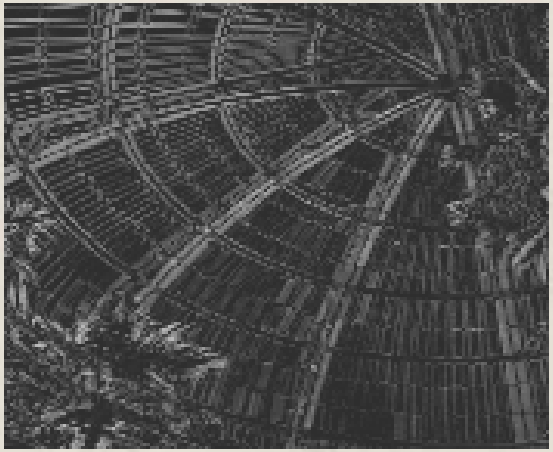
BME/BMC: Example I

Previous Frame

Current Frame

Motion Vector Field





Frame Difference

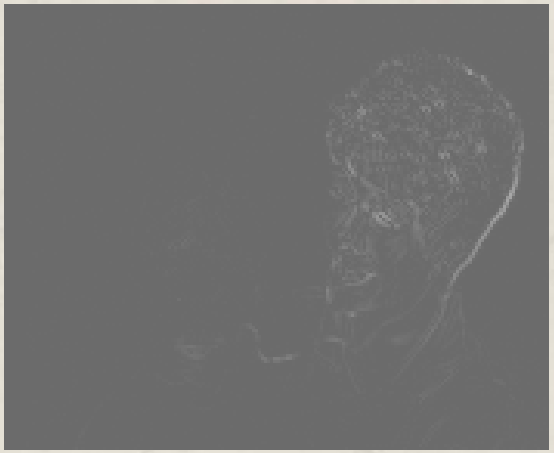
Motion-Compensated Difference

BME/BMC: Example II

Previous Frame

Current Frame

Motion Vector Field

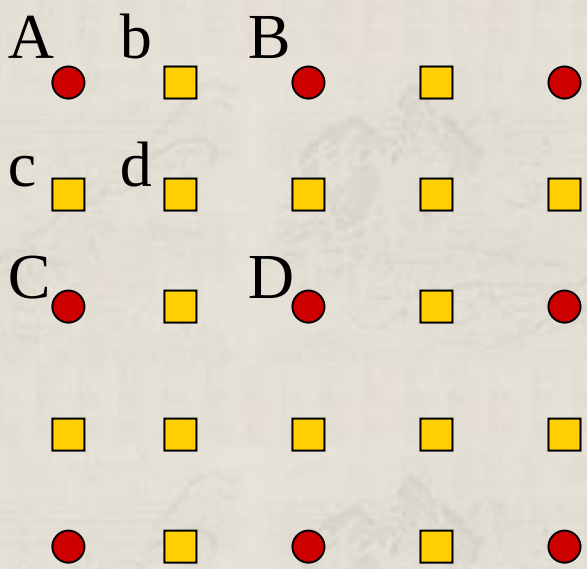


Frame Difference

Motion-Compensated Difference

Sub-Pixel Motion Estimation

- Sub-pixel motion vector resolution
- Use linear/bilinear interpolation to fill in sub-pixels
- Trade-offs: motion accuracy versus MV bit-rate and complexity increase
- H.26L uses down to 1/4-MEMC, maybe even 1/8

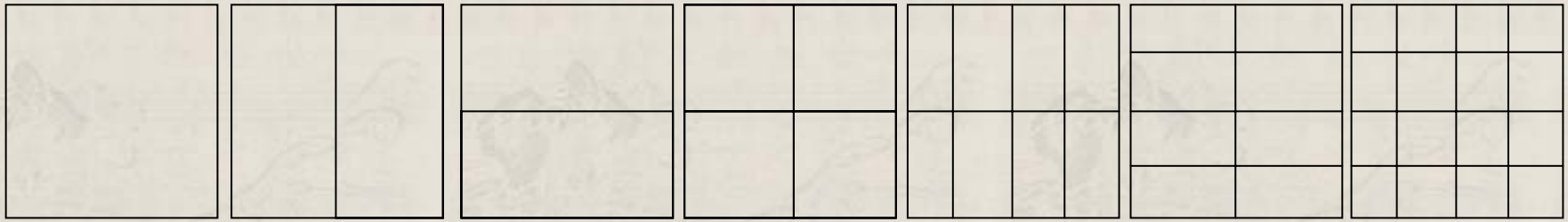


integer pixel

half pixel

b = round[(A+B)/2]
c = round[(A+C)/2]
d = round[(A+B+C+D)/4]

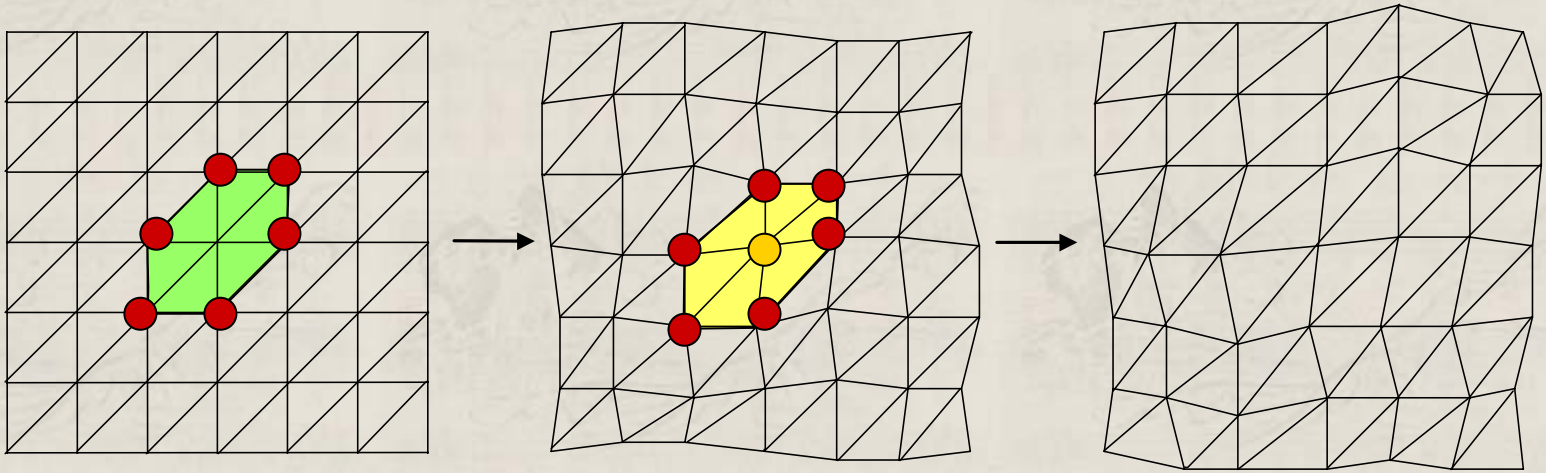
Variable-Block-Size BME



Mode 1 Mode 2 Mode 3 Mode 4 Mode 5 Mode 6 Mode 7

- Motion model for H.264
- Generalization of the traditional translation BME framework
- Sub-divide macro-block: 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4

Mesh-Based Motion Estimation

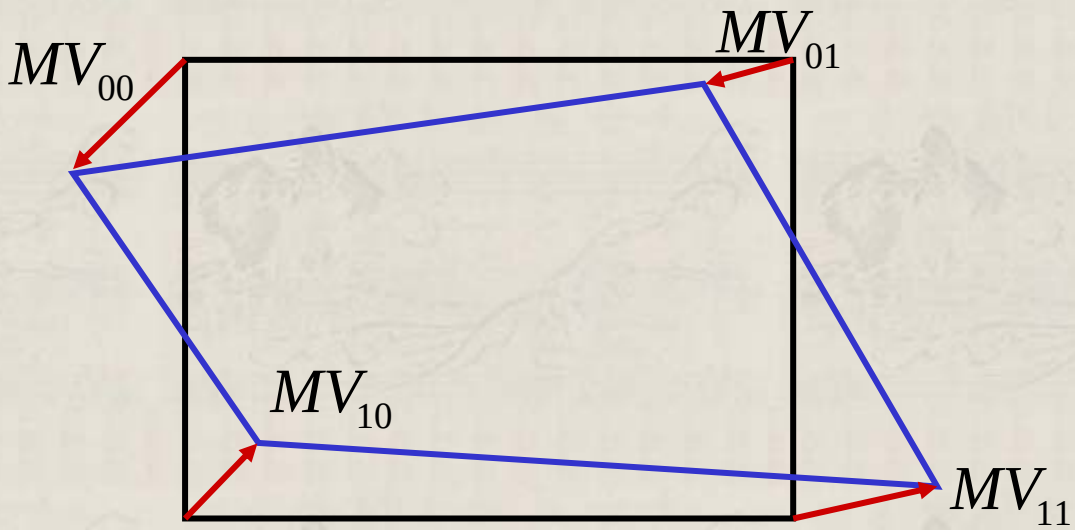


- MPEG-4 object motion
- Affine warping motion model
- Deformable polygon meshes

$$\begin{cases} f(x,y) = a_{00}x + a_{01}y + d_x \\ g(x,y) = a_{10}x + a_{11}y + d_y \end{cases}$$

- Similar MAD, SSE error measures
- Trade-offs: more accurate ME vs. tremendous complexity
- Bilinear and perspective motion models are rarely used in video coding

Global Motion Estimation



- Rarely used in practice: BME/BMC mostly suffices
- Reference frame resampling: an option in H.263+/H.263++
- Global affine motion model: special-effect warping
- 3D subband & wavelet coding: align frames before temporal filtering [Taubman]

Conclusion

- Temporal correlation dominates spatial correlation in video sequences
- Motion estimation and motion compensation improve coding performance the most in video coding
- State-of-the-art video coders, such as the current H.26L verification model, still employ BME-block DCT coding framework
- Despite its simplicity, BME is still the bottleneck of real-time video communications

Video Compression Standards

Trac D. Tran
ECE Department
The Johns Hopkins University
Baltimore MD 21218

Outline

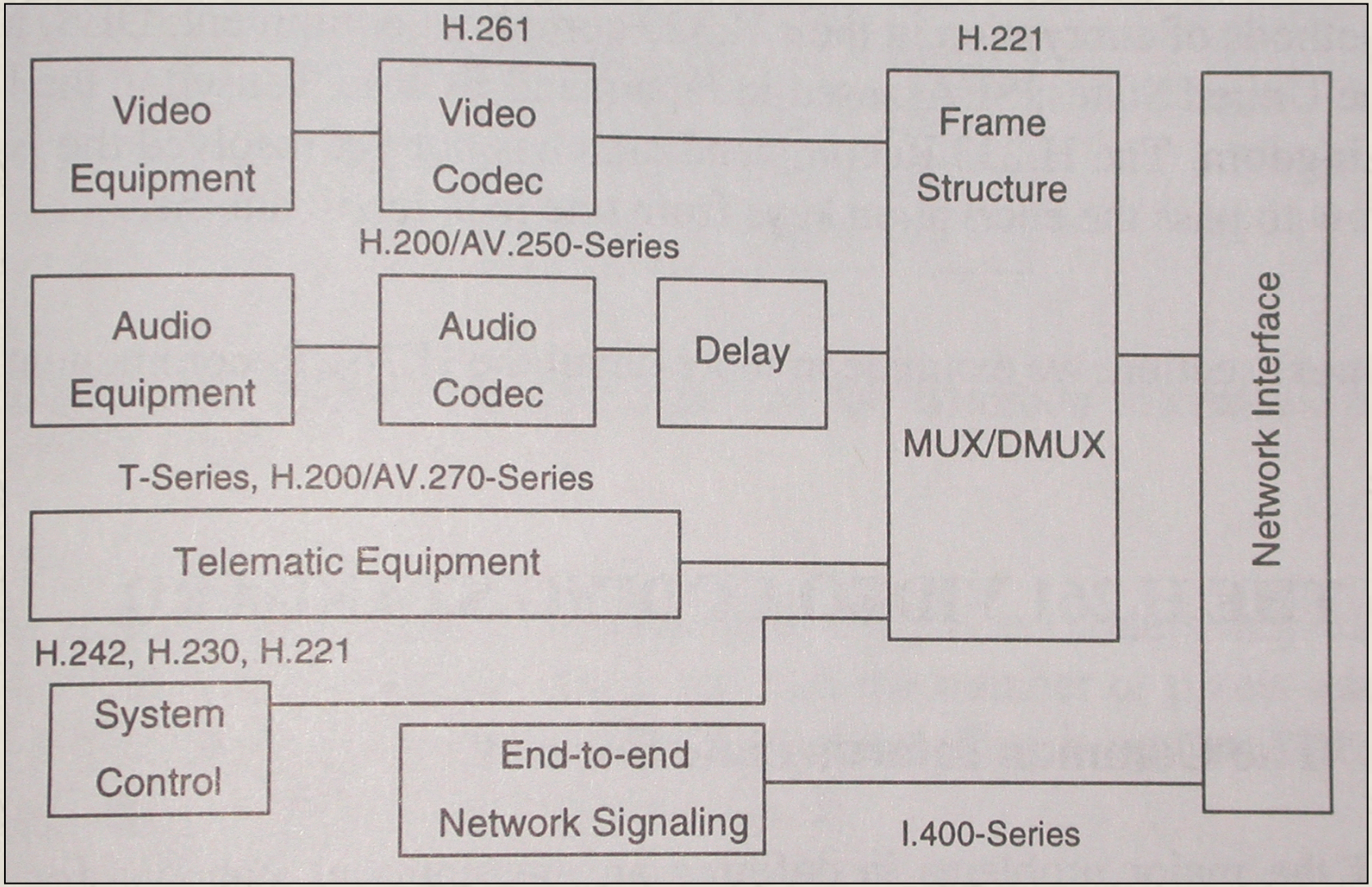
- Hybrid motion-compensated DCT coding framework
- H.261
  - First international video teleconferencing coding standard
  - History. Design criteria and goals. Signal format: CIF
  - Features
    - Coding layers: block – macro-block – slice – frame
    - Inter and Intra coding mode. Inter-Intra switching
    - Motion estimation and compensation
    - Quantization: uniform step-size with dead zone
- H.263, H.263+
- MPEG family
  - Coding and communications of moving pictures and associated audio for digital storage and archival
  - MPEG-1, MPEG-2, MPEG-4. Main features

History and Goals

- ITU-T (International Telecommunications Union), Standardization Sector, Study Group 15
- Standardization activities: Dec. 1984 – Dec. 1990
- Goal: real-time videophone and video teleconferencing at rather low bit rates and low delay
- Bit rates: 64 kbps – 1.92 Mbps
- H-Series system
  - H.261: video codec for *p x 64 kbps, p=1,2,…,30*
  - G.722, G.726, G.728: audio codec for *16 – 64 kbps*
  - H.242, H.230, H.221: system control, frame structure, mux, handshaking protocols for compliant equipments / components

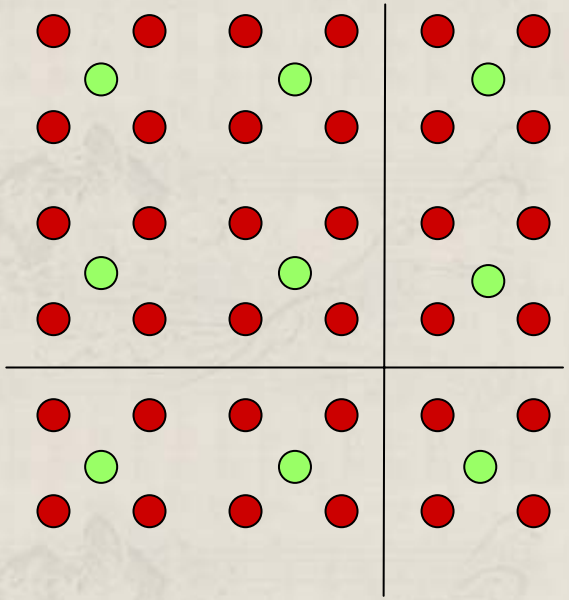
  } H.320

Generic H.320 System



Signal Format

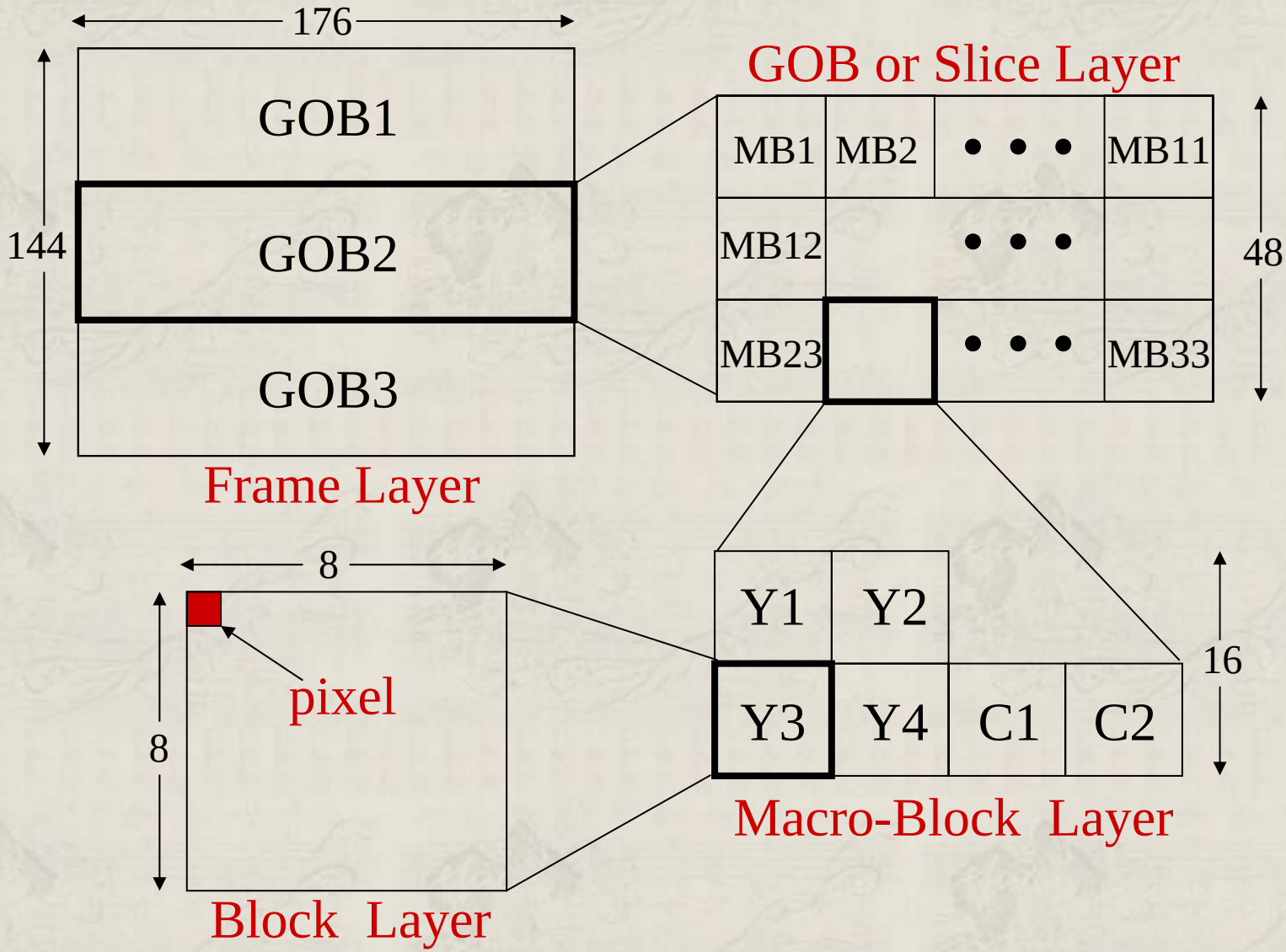
- Only 2 formats are allowed in H.261
  - CIF & QCIF
- CIF
  - Common Intermediate Format
  - 4:2:0 Y Cr Cb format, progressive (non-interlaced)
  - *352 x 288* luminance frame, *176 x 144* chrominance frame
  - This is a compromise between
    - NTSC: Japan/North America, 525 lines, interlaced, 30 fps
    - PAL/SECAM: Others, 625 lines, interlaced, 25 fps
- QCIF
  - Quarter-CIF
  - 4:2:0 Y Cr Cb format, progressive (non-interlaced)
  - *176 x 144* luminance frame, *88 x 72* chrominance frame



block boundary

- luminance sample
- chrominance sample

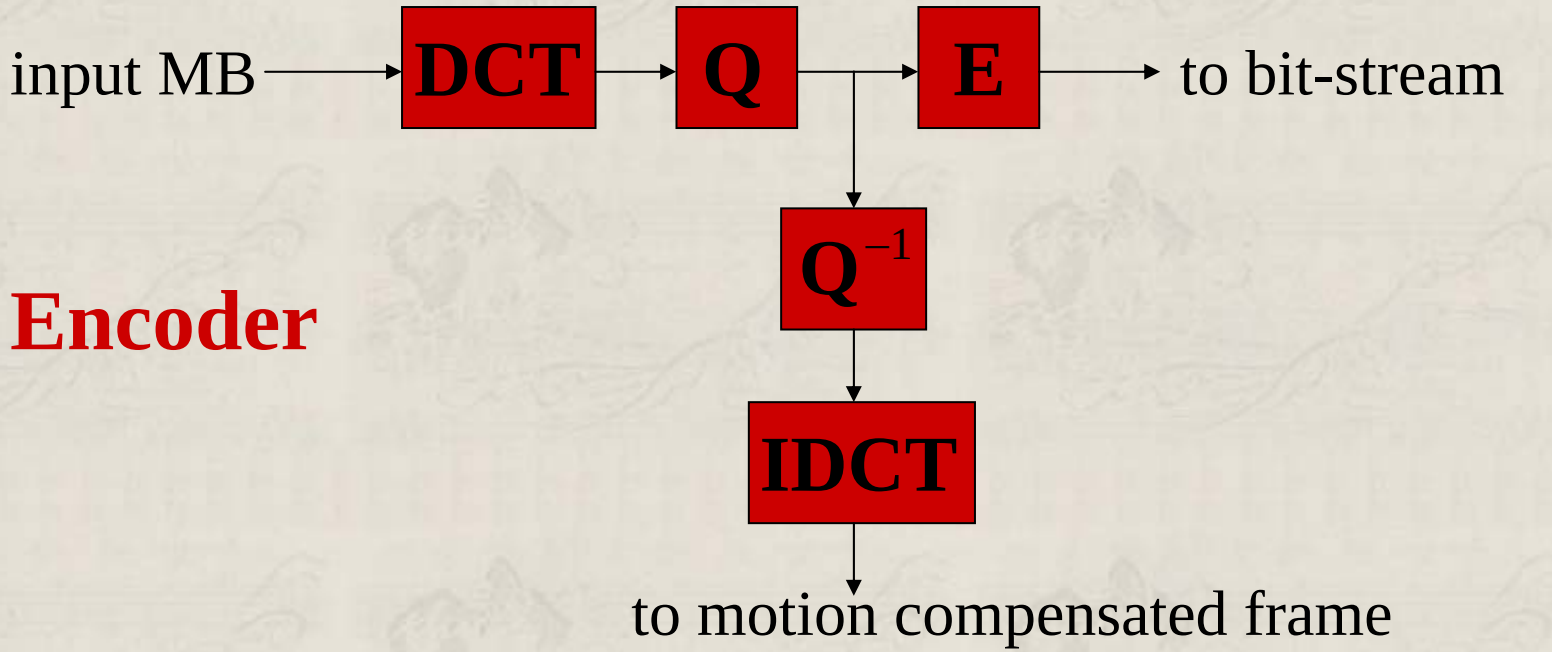
Hierarchical Block Structure



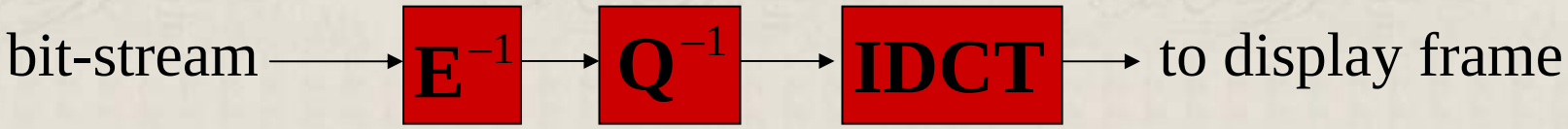
Inter and Intra Coding

- Intra
  - MB is encoded as is without motion compensation
  - DCT followed by Q, zig-zag, run-length, Huffman
- Inter
  - Block-matching motion estimation
  - Predictive motion residue from best-match block is DCT encoded (similarly to intra-mode)
  - Motion vector is differentially encoded

Intra-Coding Mode

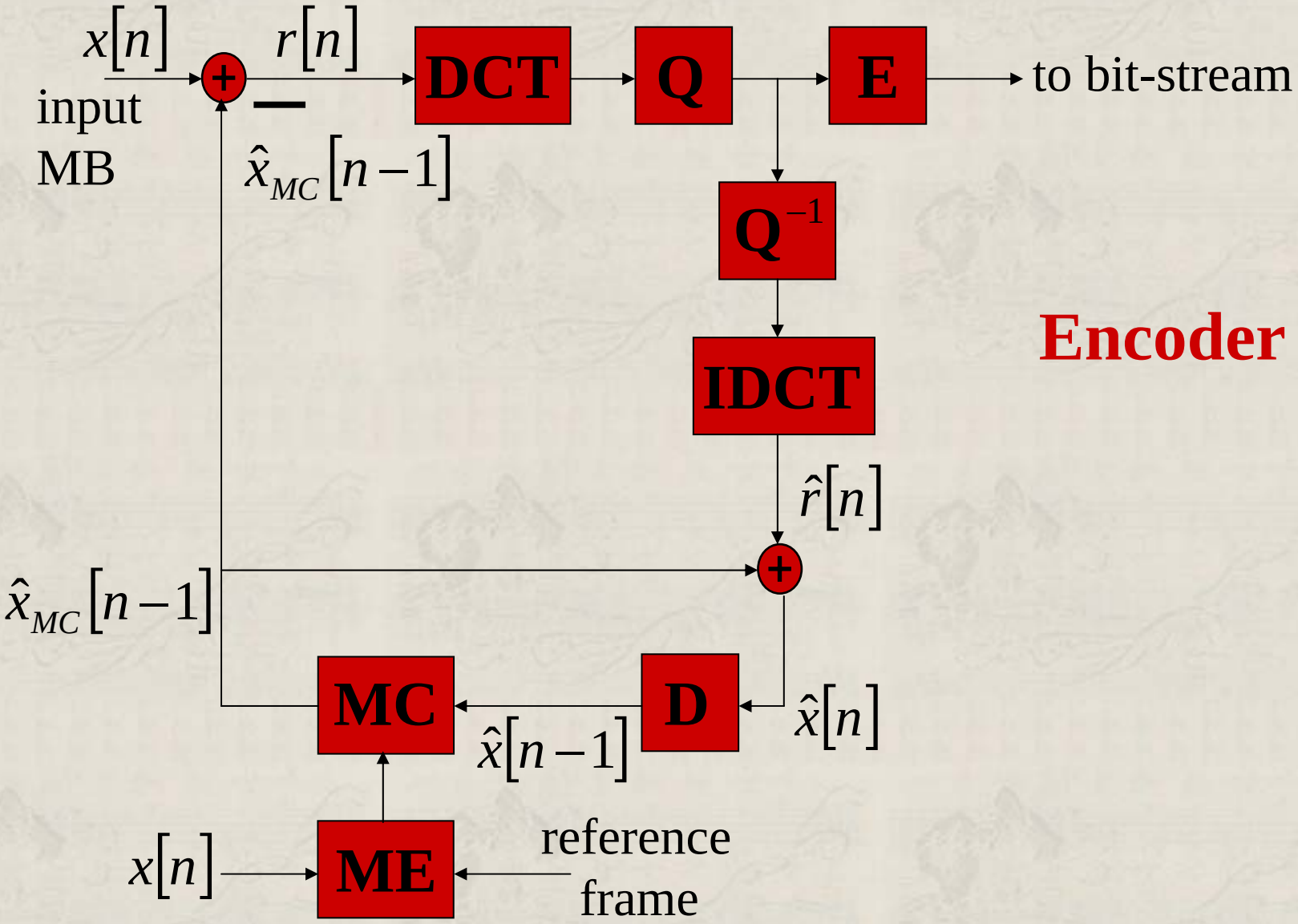


Encoder



Decoder

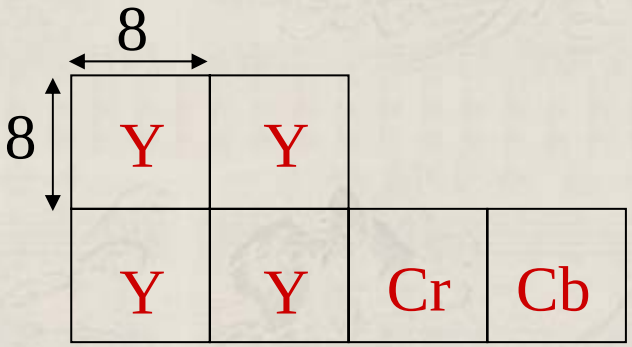
Inter-Coding Mode

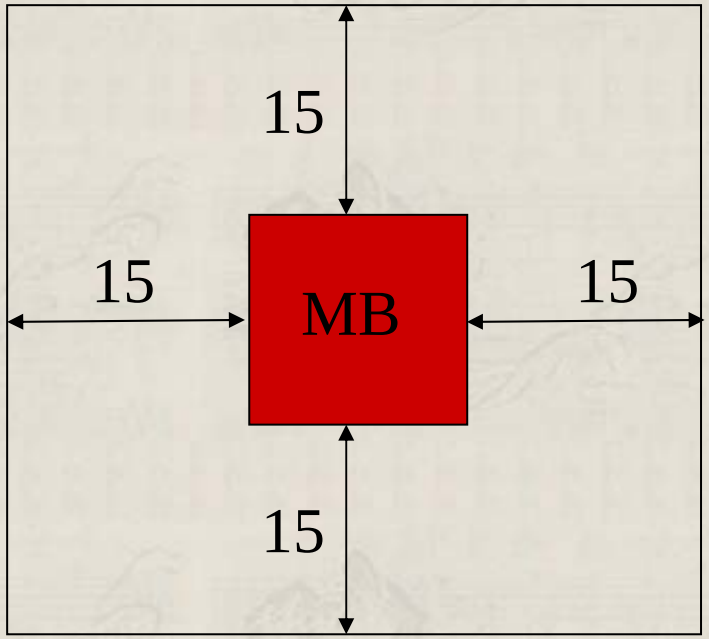


Encoder

Motion Estimation in H.261

- Macro-block
  - Luminance: 16x16, four 8x8 blocks
  - Chrominance: two 8x8 blocks
  - Motion estimation only performed for luminance component
- Motion vector range
  - [-15, 15]

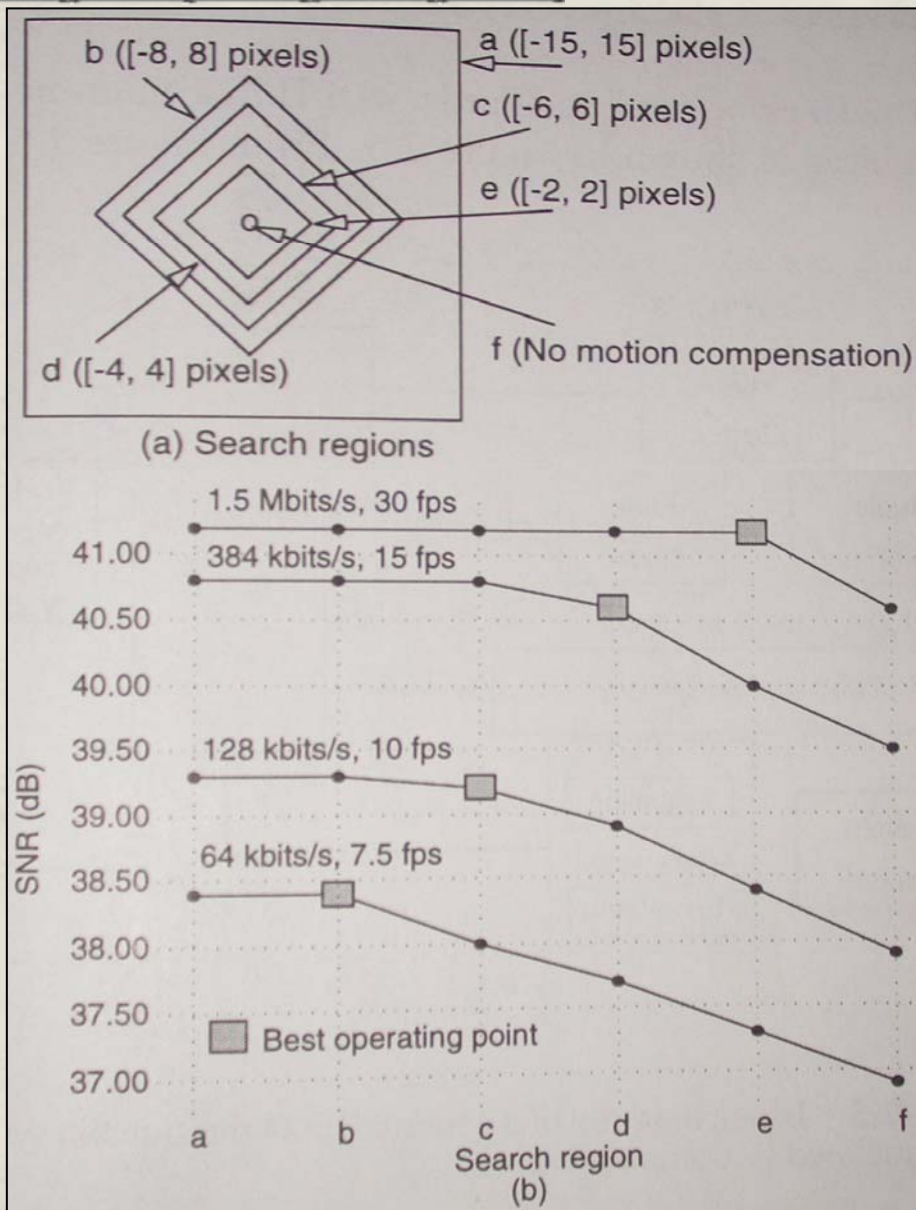




Search Area in Reference Frame

Motion Estimation & Video Quality

- Video teleconferencing
  - Head & shoulders
  - Small diamond-shape search region yields good performance
  - In high bit-rate situations, no benefit from large search window
  - ME complexity can be significantly reduced
  - [Bhaskaran et al, 1997]



Coding of Motion Vectors

- MV has range [-15, 15]
- Integer pixel ME search only
- Motion vectors are differentially & separably encoded

$$MVD_x = MV_x[n] - MV_x[n-1]$$

$$MVD_y = MV_y[n] - MV_y[n-1]$$

- 11-bit VLC for MVD
- Example

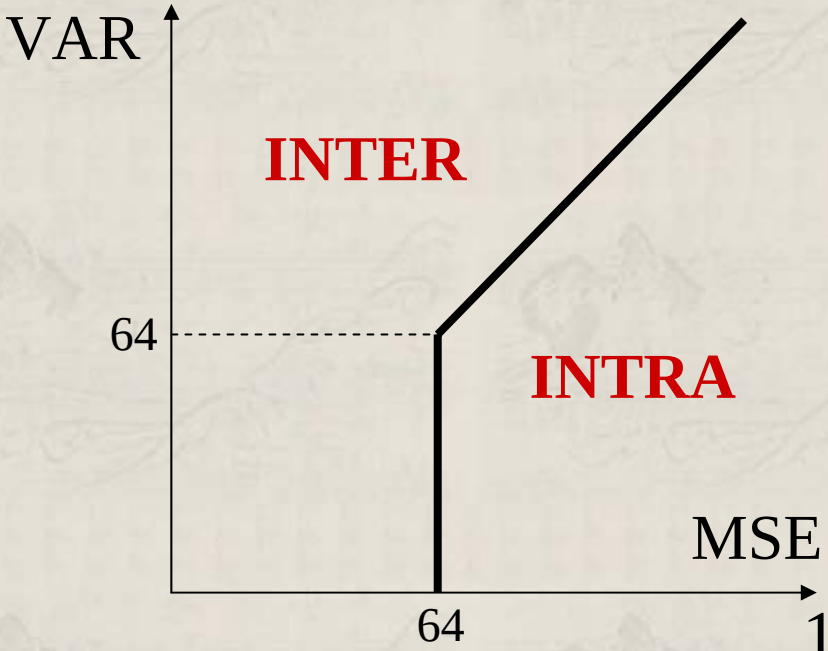
MV = 2 2 3 5 3 1 -1…
MVD = 0 1 2 -2 -2 -2…

Binary: 1 010 0010 0011 0011 0011…

MVD	VLC
…	…
-2 & 30	0011
-1	011
0	1
1	010
2 & -30	0010
3 & -29	0001 0

Inter/Intra Switching

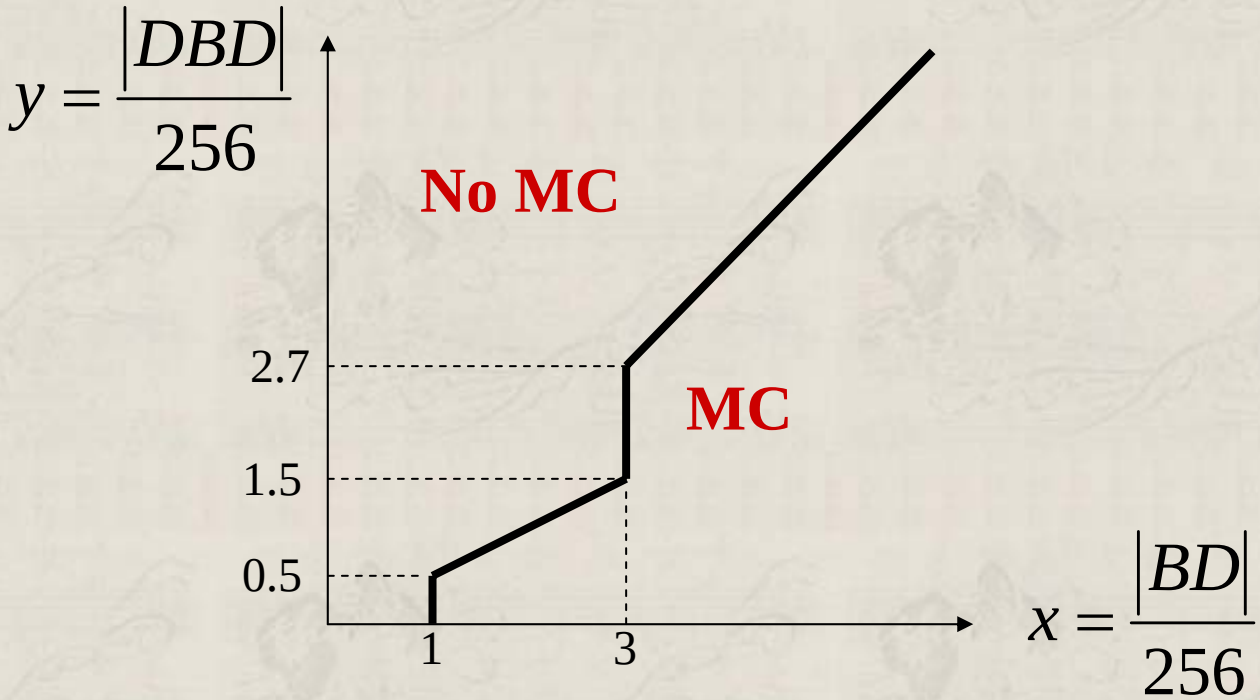
- Based on energy of prediction error
  - High energy: scene change, occlusions, uncovered areas…
    ➔ use intra mode
  - Low energy: stationary background, translational motion …
    ➔ use inter mode



$$VAR = \frac{1}{256}\sum_{MB}\left(c[x,y]-\bar{c}\right)^2$$

$$MSE = \frac{1}{256}\sum_{MB}\left(c[x,y]-r[x+dx,y+dy]\right)^2$$

MC or No MC?



$$BD = \frac{1}{256} \sum_{MB} |c[x, y] - r[x, y]|$$

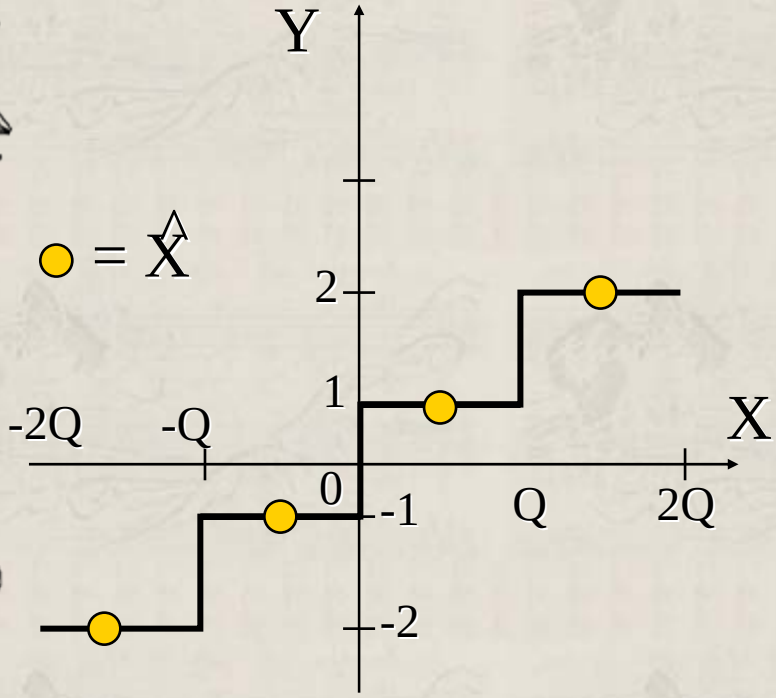
$$DBD = \frac{1}{256} \sum_{MB} |c[x, y] - r[x + dx, y + dy]|$$

Loop Filter

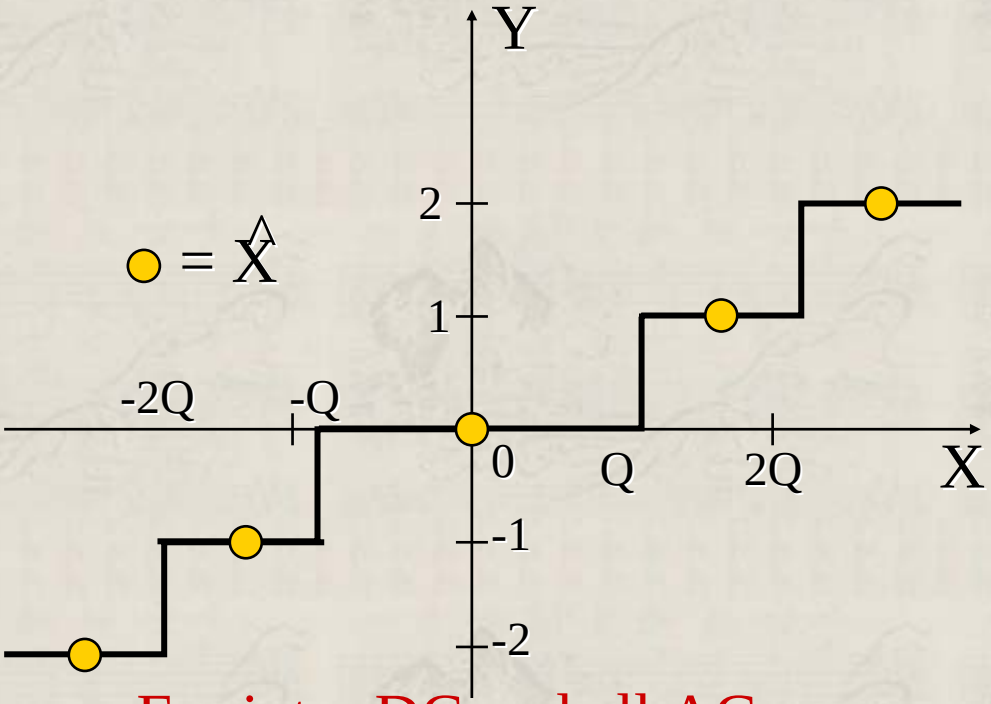
- Optional
- Can be turned on or off for each block, usually go together with MC
- Advantage
  - Decreases prediction error by smoothing the prediction frame
  - Reduces high-frequency artifacts like mosquito effects
- Disadvantage
  - Increases complexity & overhead
- 3-tap FIR separable low-pass filter
  - At block boundary: h[n]=[0 1 0] (no filtering)
  - Inside: h[n] = [1 2 1]/4

Quantization

- Uniform mid-rise quantizer for intra DC coefficients
- Uniform mid-tread quantizer with double dead zone for inter DC and all AC coefficients

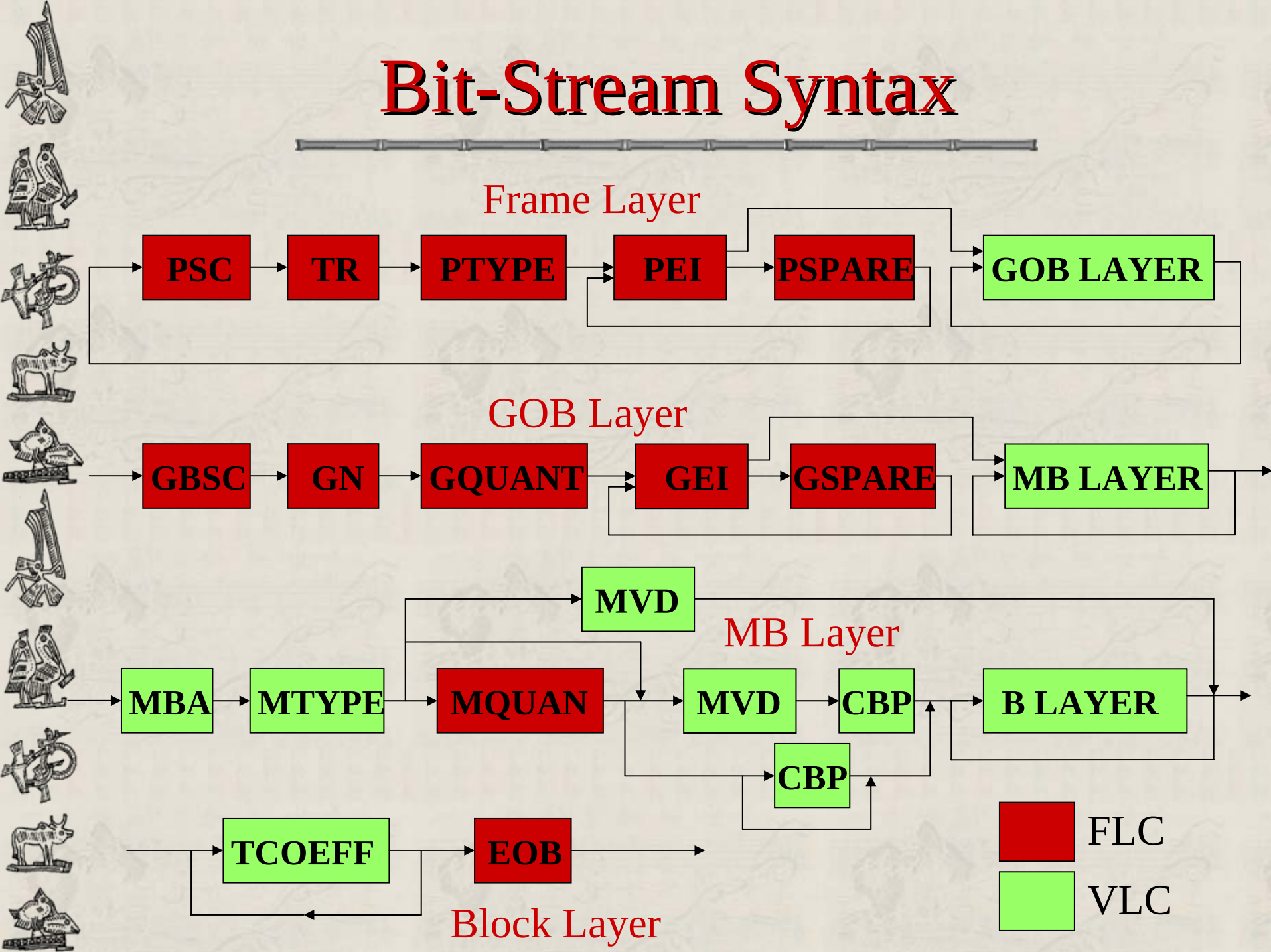


For intra DC



For inter DC and all AC

Bit-Stream Syntax



Bit-Stream Syntax

- Examples of FLC
  - PSC: Picture Start Code, 20 bits
  - PTYPE: Picture Type, 6 bits
  - GBSC: GOB Start Code, 16 bits
  - GN: Group Number, 4 bits, indexing 12 GOBs
  - GQUANT: Group Quantization information, 5 bits
  - MQUANT: MB Quantization information, 5 bits
  - EOB: End-of-Block
- Examples of VLC
  - MBA: MB Address, indexing MBs within a GOP, 11 bits max
  - MTYPE: MB Type information
  - MVD: Motion Vector Data, 11 bits max, 32 VLCs
  - CBP: Coded Block Pattern, 9 bits max, 63 VLCs
  - TCOEFF: Transform Coefficients

H.261 vs MPEG-1

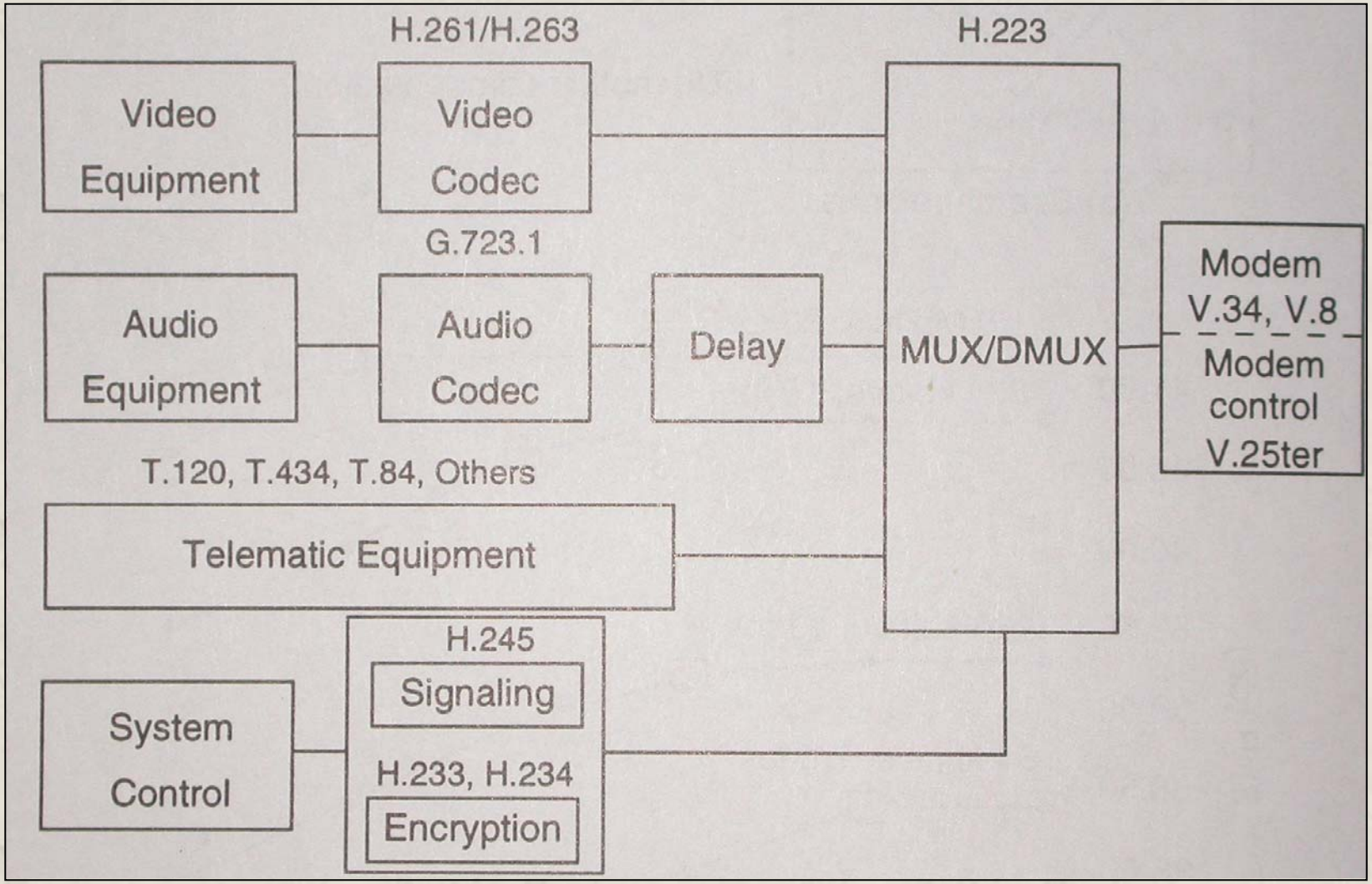
MPEG-1	H.261
Can have higher resolution	CIF & QCIF
Variable aspect ratio	4:3 aspect ratio
GOP: Group of Picture	No GOPs
I, P, B MB	Only I, P MB
Typical 1 Mbps	64 kbps – 2 Mbps
No skip restriction	Can only skip 1, 2, 3 frames
Half-pixel MEMC	Integer MEMC
Typical MV range [-15, 15]	Typical MV range [-7, 7]
End-to-end delay is not critical	Delay is critical

B: Bi-directional motion estimation & compensation

H.263

- Standardization effort started Nov 1993
- Aim
  - low bit-rate video communications, less than 64 kbps
  - target PSTN and mobile network: 10-32 kbps
- Near-term
  - H.263 and H.263+: established late 1997
- Long-term
  - H.26L, H.264: still under investigation
- Main properties
  - H.261 with many MPEG features optimized for low bit rates
  - Performance: 3-4 dB improvements over H.261 at less than 64 kbps; 30% bit rate saving over MPEG-1

H.263 & H.324



block diagram of a generic H.324 multimedia system

H.261 vs H.263

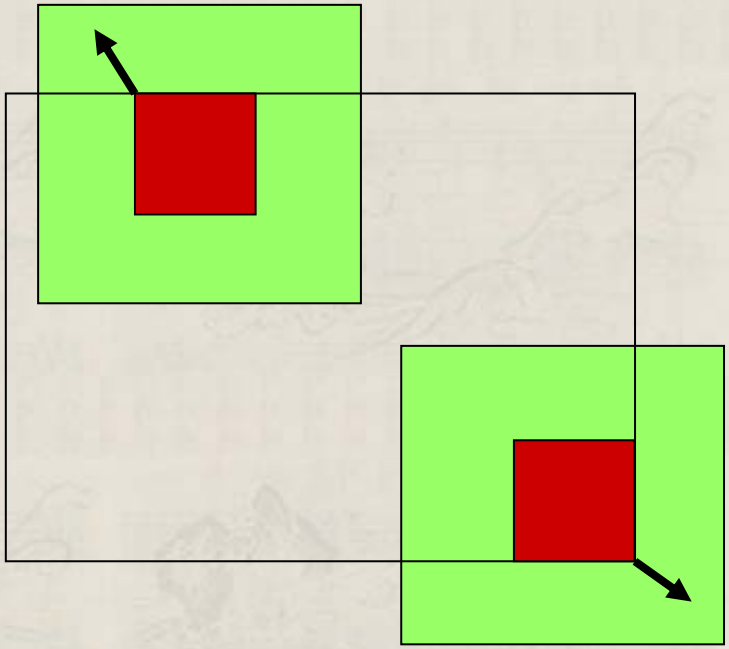
Features	H.261	H.263
Bit Rate	Reasonable, 64 kbps – 2 Mbps	Very low, 64 kbps or less
Format	CIF & QCIF	Sub-QCIF, QCIF, CIF, 4CIF, 16CIF
GOB Structure	GOB=33 MBs	GOB=1 row of MBs
Error Correction	BCH single-error correction code	No specification. Optional
Motion	Integer-pixel	Half-pixel
Loop Filter	Yes	No
MV Coding	Differential coding for MVs	Predictive coding for MVs

H.263+ Advanced Features

- Unrestricted motion vectors mode
- Can have four motion vectors per MB (each for an 8x8 quadrant)
- Overlapped block motion compensation (OBMC)
- Syntax-based adaptive arithmetic coding mode
- PB-frames mode

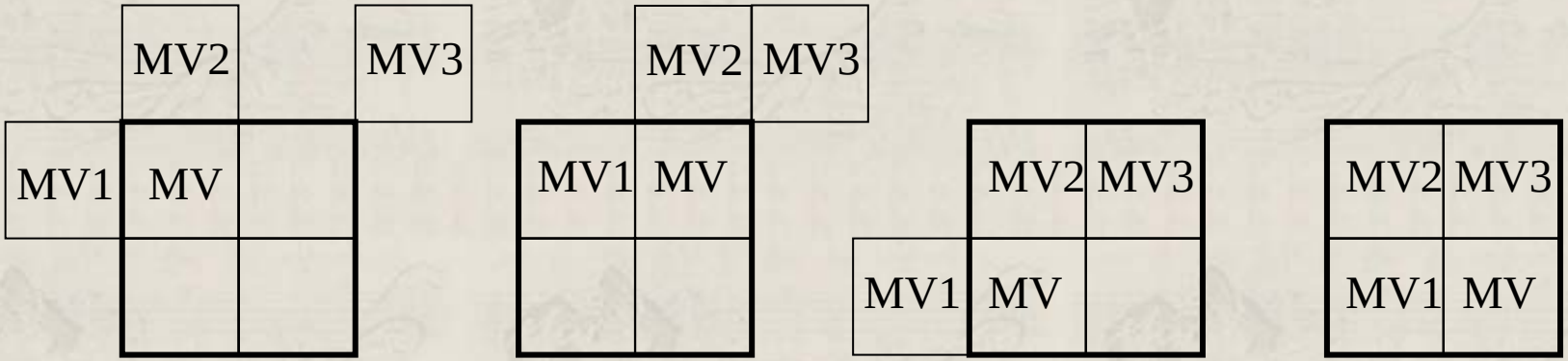
Unrestricted Motion Vectors

- Improve motion accuracy
  - Motion vector range is extended from [-16, 15.5] to [-31.5, 31.5]
  - Motion vector can point *outside* of the frame
  - Closest edge pixel is used (edge pixel is repeated)
- UMV dramatically improves motion estimation when moving objects are entering/exiting the frame or moving around the frame border

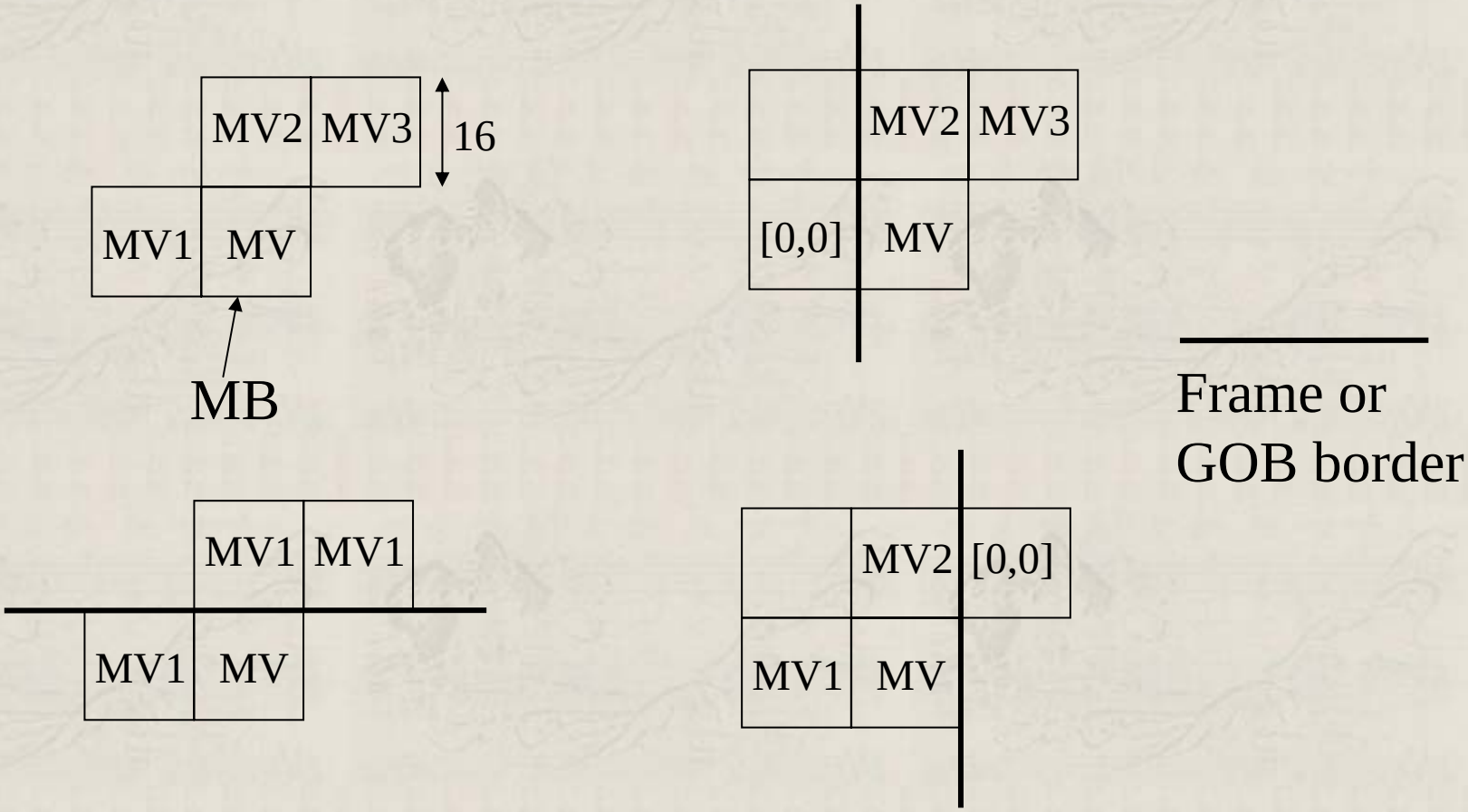


Advanced Prediction Mode

- This option significantly improves image quality
  - Four motion vectors for a macro-block
  - Overlapped block motion compensation (OBMC)
- Four MVs per MB
  - Can be enabled on a block-by-block basis
  - Each MV covers an 8x8 quadrant
  - MV for Cr or Cb is computed by averaging the 4 luminance MVs & scaling by 2 (rounding to nearest half-pixel)



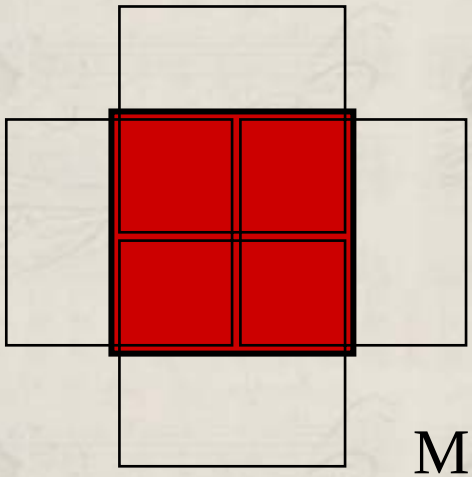
MV Predictive Coding



- MVD = MV – median(MV1, MV2, MV3)

OBMC

$$H_0 = \begin{bmatrix} 4 & 5 & 5 & 5 & 5 & 5 & 5 & 4 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 6 & 6 & 6 & 6 & 5 & 5 \\ 5 & 5 & 6 & 6 & 6 & 6 & 5 & 5 \\ 5 & 5 & 6 & 6 & 6 & 6 & 5 & 5 \\ 5 & 5 & 6 & 6 & 6 & 6 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 4 & 5 & 5 & 5 & 5 & 5 & 5 & 4 \end{bmatrix} \quad H_1 = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 2 & 2 & 2 & 2 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 2 & 2 & 2 & 2 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{bmatrix} \quad H_2 = \begin{bmatrix} 2 & 1 & 1 & 1 & 1 & 1 & 1 & 2 \\ 2 & 1 & 1 & 1 & 1 & 1 & 1 & 2 \\ 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 \\ 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 \\ 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 \\ 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 \\ 2 & 1 & 1 & 1 & 1 & 1 & 1 & 2 \\ 2 & 1 & 1 & 1 & 1 & 1 & 1 & 2 \end{bmatrix}$$



adjacent block

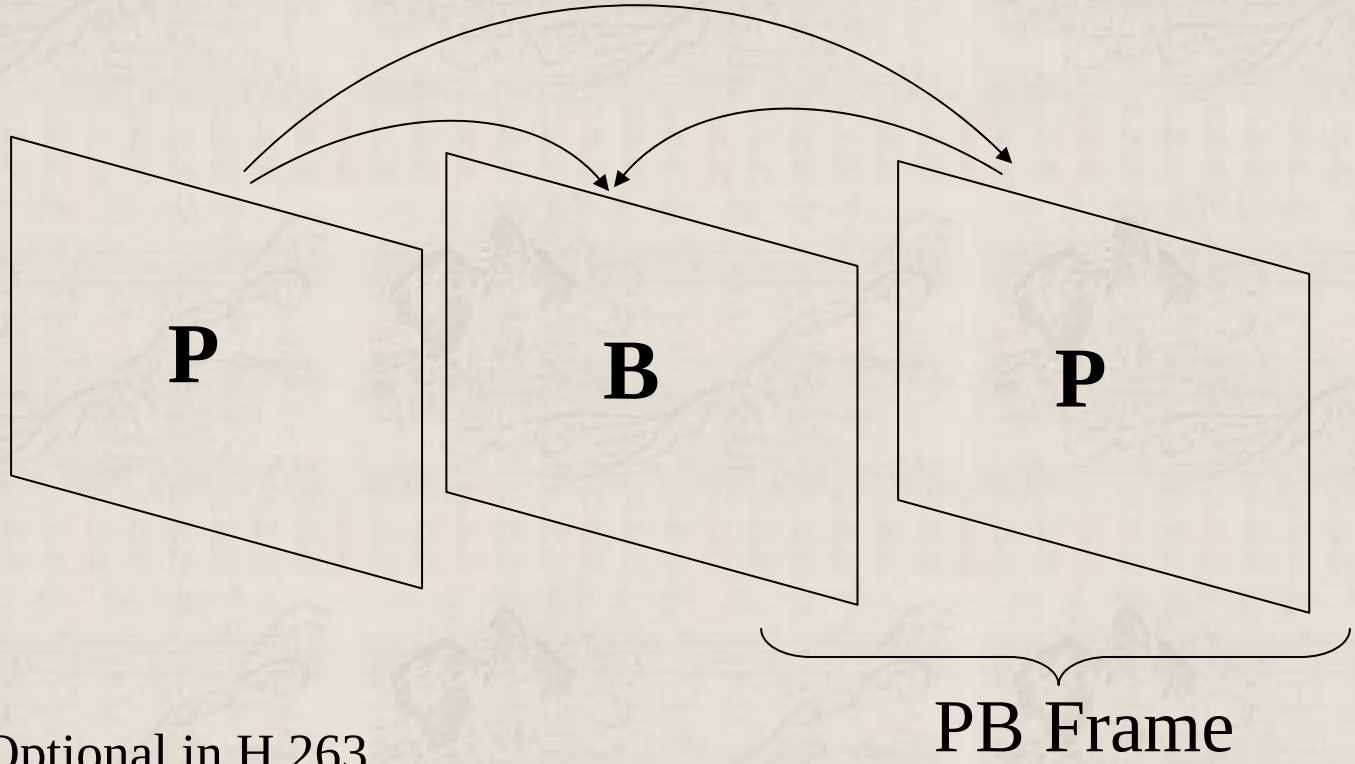
$$p = \frac{1}{8}\left(x_0 H_0 + x_1 H_1 + x_2 H_2 + 4\right)$$

MC prediction current block

Syntax-Based Arithmetic Coding

- Adaptive arithmetic coding based on appropriate syntax is used to replace all VLC operations
- Improves coding efficiency significantly
  - For inter-MB, AC yields 3-4% bit rate reduction
  - For intra-MB, AC yields ~10% bit rate reduction

PB-Frames Mode



- Optional in H.263
- Two frames are encoded as one unit: P and B
- P is predicted from a previously decoded P frame
- B is predicted from the P frame in the unit and another previously decoded P frame
- 12 blocks from a MB: 6 from P and 6 from B

MPEG

- Coding and communications of moving pictures and associated audio for digital storage and archival
- MPEG: Moving Picture Expert Group
- MPEG family
  - MPEG-1, Nov 1992
  - MPEG-2, Nov 1994
  - MPEG-4, Oct 1998
  - MPEG-7, ongoing work
- Main features of the MPEG video family
  - Bi-directional MEMC
  - I-frame, P-frame, B-frame
  - Structure: Group of Pictures (GOP), picture, slice, macro-block
  - Coding decisions

MPEG Goals and Applications

- MPEG-1
  - Optimized for applications that support a continuous transfer bit rate of about 1.5 Mbps (example, CD-ROM)
  - Target 1.2 Mbps for video and 250-300 kbps for audio, around analog VHS quality
  - Does not support interlaced sources
  - Main target source: SIF YCrCb 4:2:0 360 x 240 x 30 fps
  - VCD
- MPEG-2
  - The most commercially successful international coding standard
  - Wide range of bit rates: 4 – 80 Mbps; optimized for 4 Mbps
  - Target high-resolution, high-quality video broadcast & playback
  - DVD, Digital TV: DirecTV, HDTV…

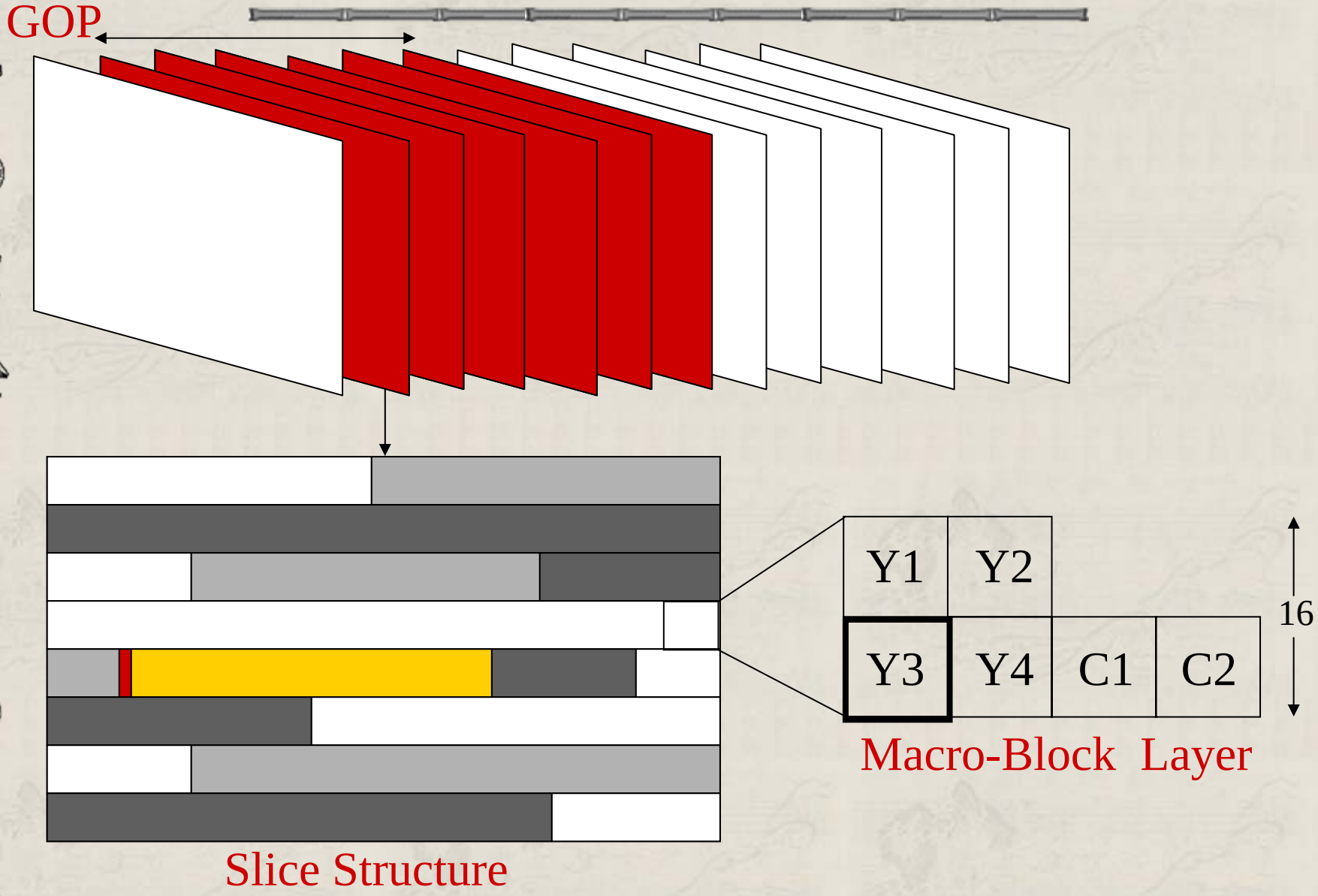
Requirements

- Coding of generic video at around 1.5 Mbps at reasonable quality (VHS)
- Random access capability, frequent access point
- Fast forward and fast rewind capability
- Audio-video synchronization during play and access
- Simple decoder
- Flexibility of data format
- Certain degree of robustness to communication errors
- Real-time encoder possibility

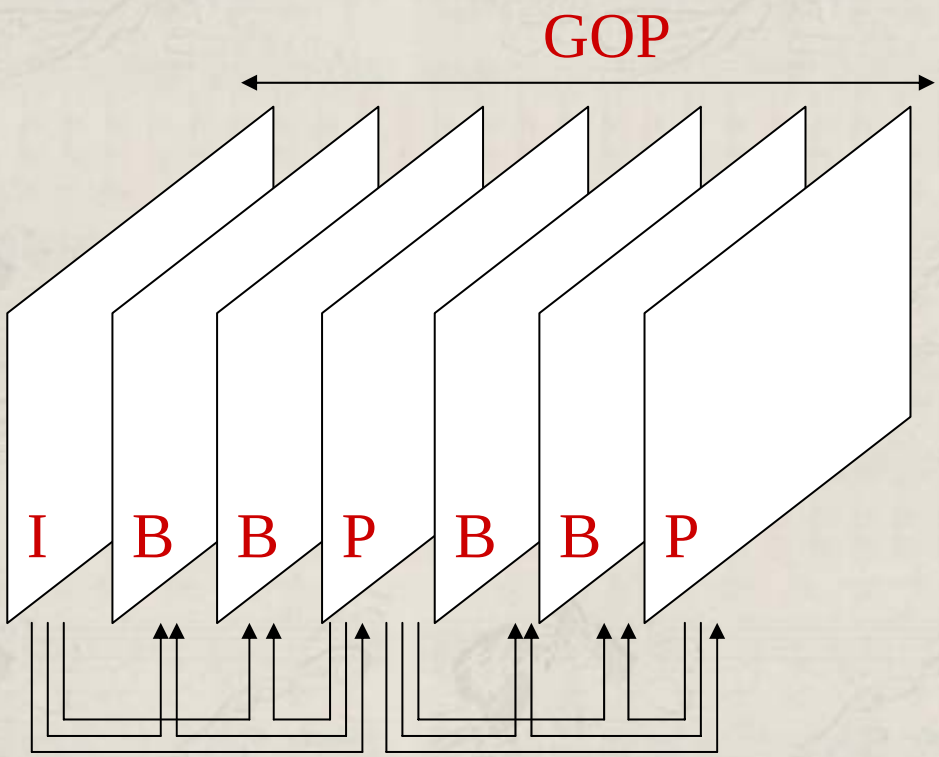
From H.261 to MPEG-1

- There are a few new features in MPEG-1 comparing to the pioneering H.261 codec
  - Flexible data sizes and frame rates
  - More flexible slice structure to replace the fixed GOB structure
  - Data structure: introducing Group of Picture (GOP) allowing frequent access points
  - Bi-directional motion compensation, B-frames
  - Half-pixel motion compensation
  - More finely tuned VLCs for different purposes
  - Quantization table (like JPEG) replaces single Q step size

Hierarchical Data Structure



GOP

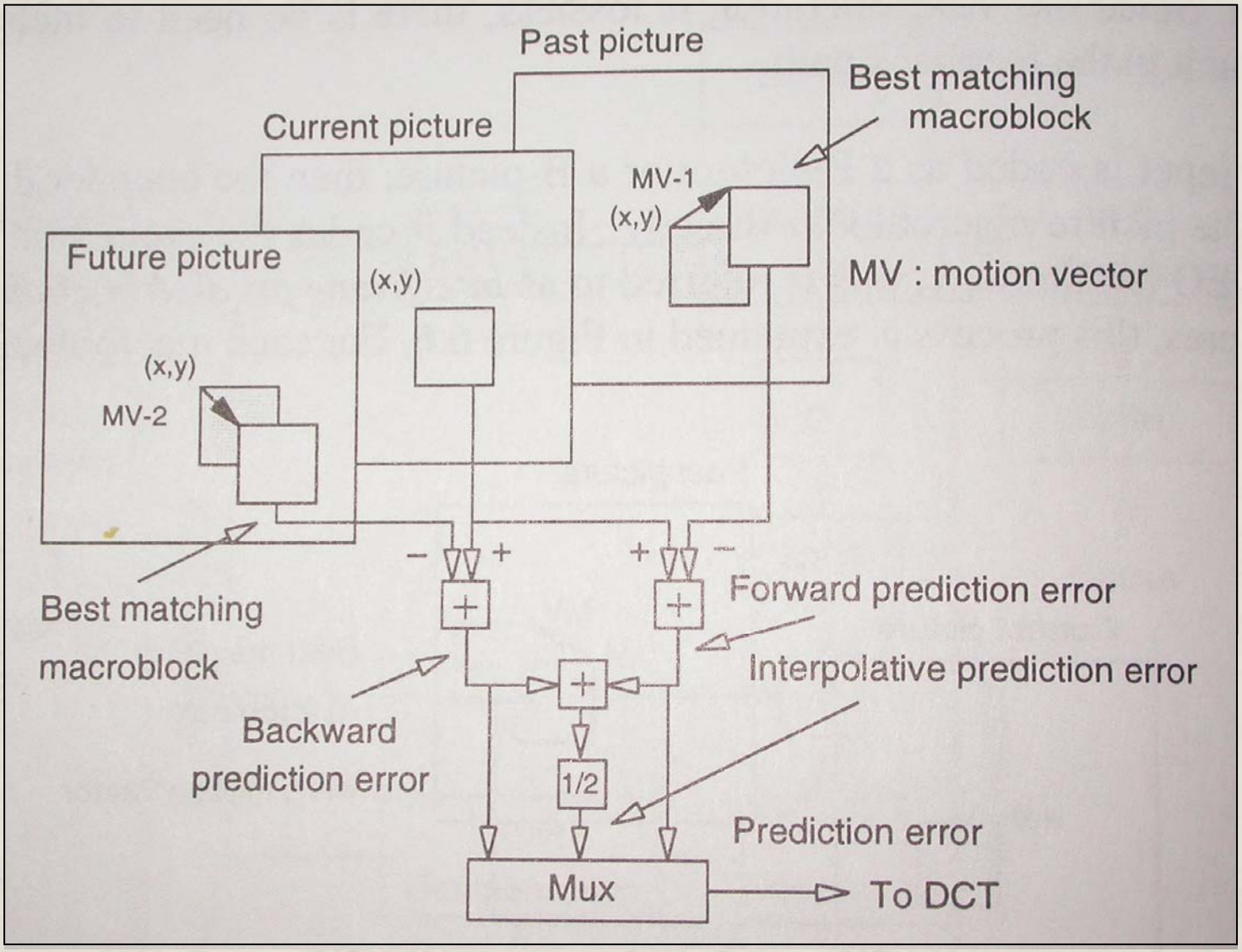


- N = number of frames (pictures) in a GOP
- M = number of B-frames between I- or P-frame + 1
- There is one I-frame for each GOP
- I-frame: intra coded only
- P-frame: forward prediction and MC
- B-frame: both forward and backward prediction and MC

Encode/Decode Sequence: I P B B P B B

Display Sequence: I B B P B B P

Bidirectional MEMC

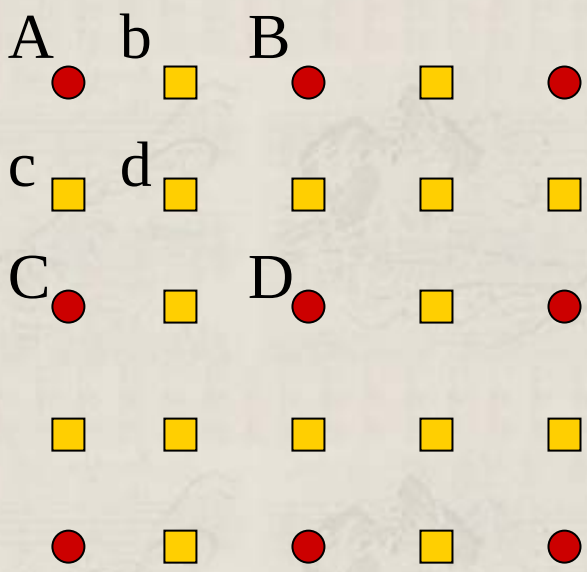


Bidirectional MC Properties

- Advantage
  - Higher coding efficiency, frame rate can be increased significantly with few bits
  - More accurate motion estimation & compensation
  - No error propagation
- Disadvantage
  - More memory buffer for frame storage (minimum of 3)
  - More end-to-end delay

Half-Pixel Motion Estimation

- Use linear/bilinear interpolation to fill in sub-pixels
- Trade-offs: motion accuracy versus MV bit-rate and complexity increase
- H.26L uses down to 1/4-MEMC, maybe even 1/8



integer pixel

half pixel

b = round[(A+B)/2]
c = round[(A+C)/2]
d = round[(A+B+C+D)/4]

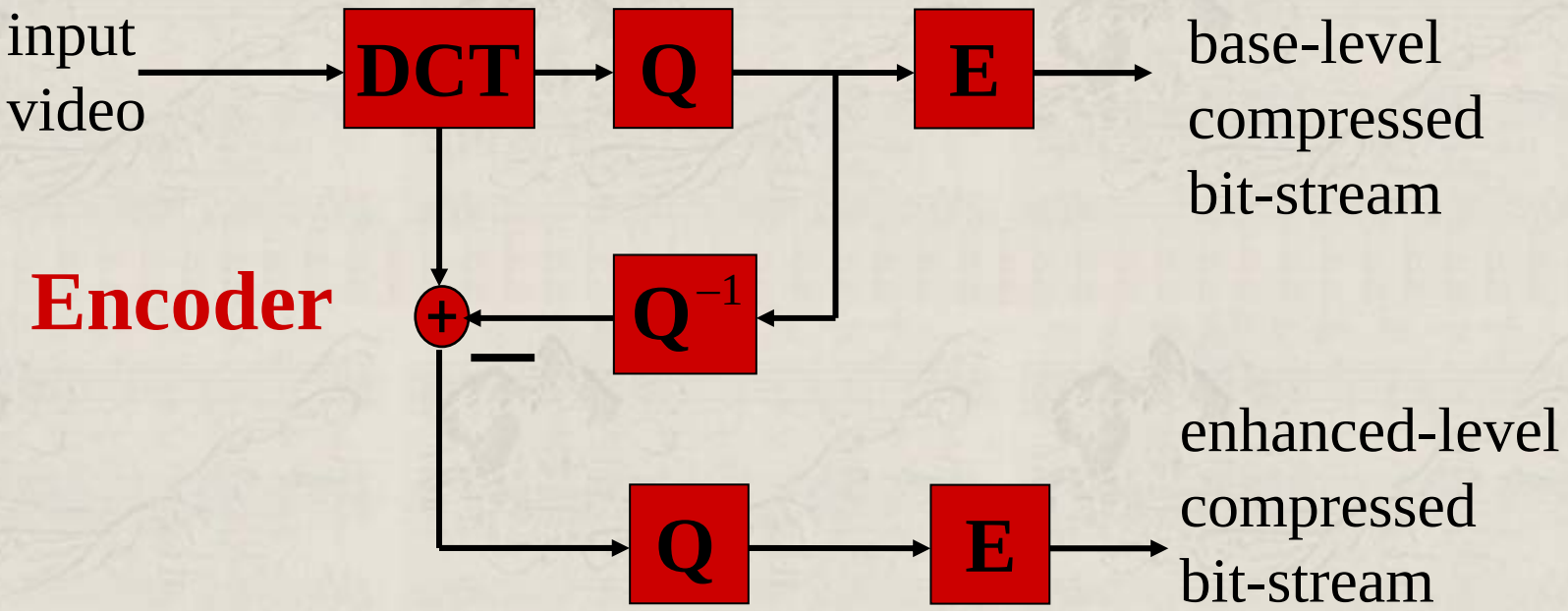
Computational Complexity

Decoding Function	Load %
Bit-stream header parsing	0.44
Huffman decoding and inverse Q	19.00
8 x 8 IDCT	22.10
Motion compensation	38.64
Color transformation and display	19.82

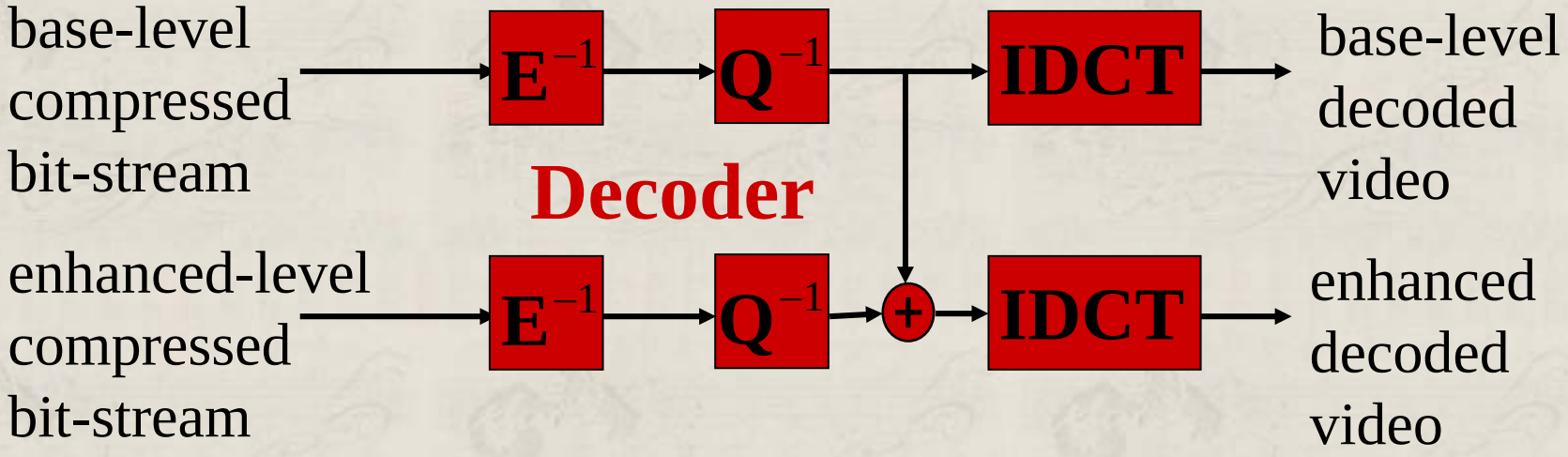
Video: Coarse- & Fine-Granularity

- Bit-plane coding schemes such as EZW & SPIHT are classified as fine-granularity scalability coding
  - Many layers can be added to improve quality. Each layer comes from a bit plane
  - Exact bit rate control
- Coarse-granularity scalability
  - Several bit planes can be combined together to yield a layer
  - For example, the top half of the bit planes can form the base layer whereas the remaining form the enhancement layer
  - Less flexibility but improved coding efficiency

Encoder: SNR Layer Scalability

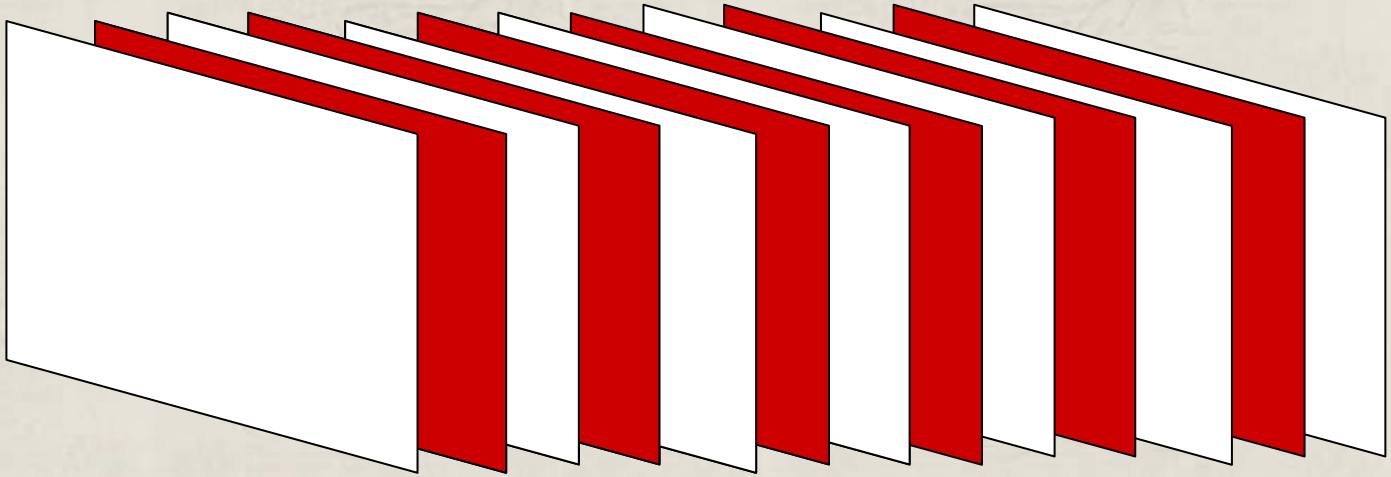


Decoder: SNR Layer Scalability

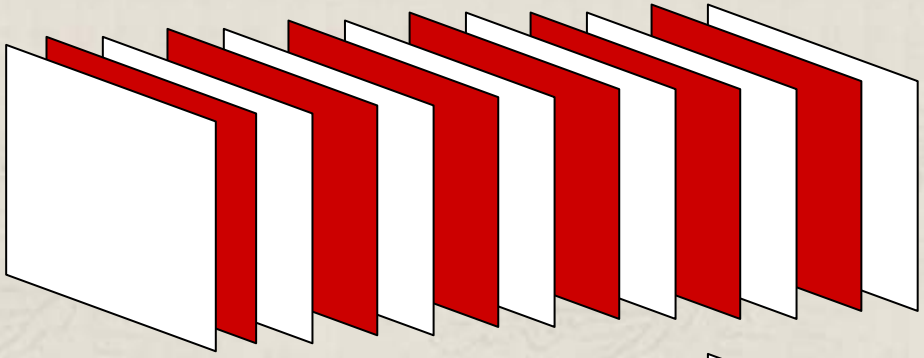


Spatial & Temporal Scaling

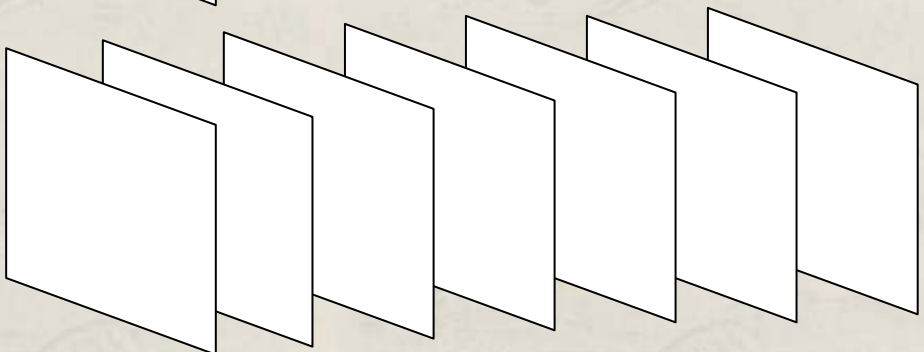
Original Video



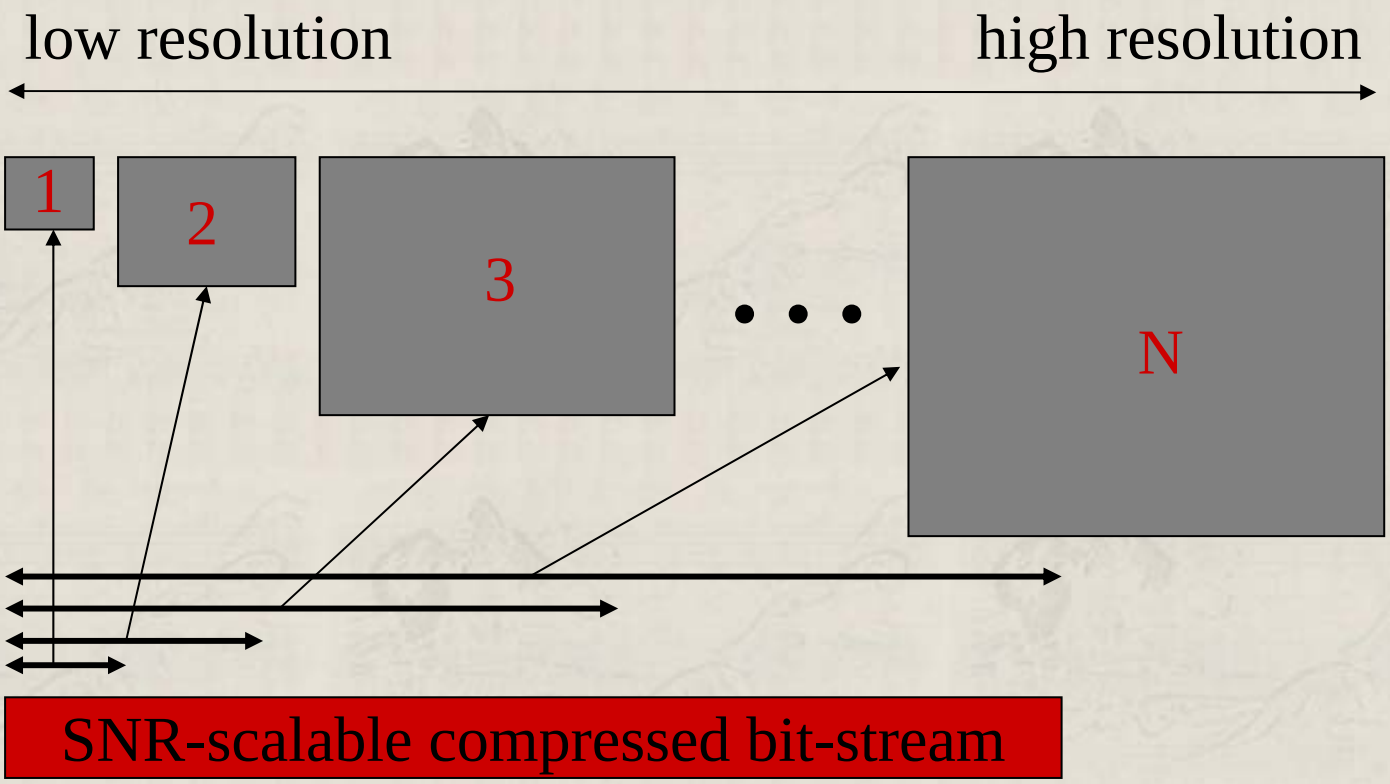
Spatial Scaling:
Half Resolution



Spatial & Temporal Scaling:
Half resolution
& Half frame rate

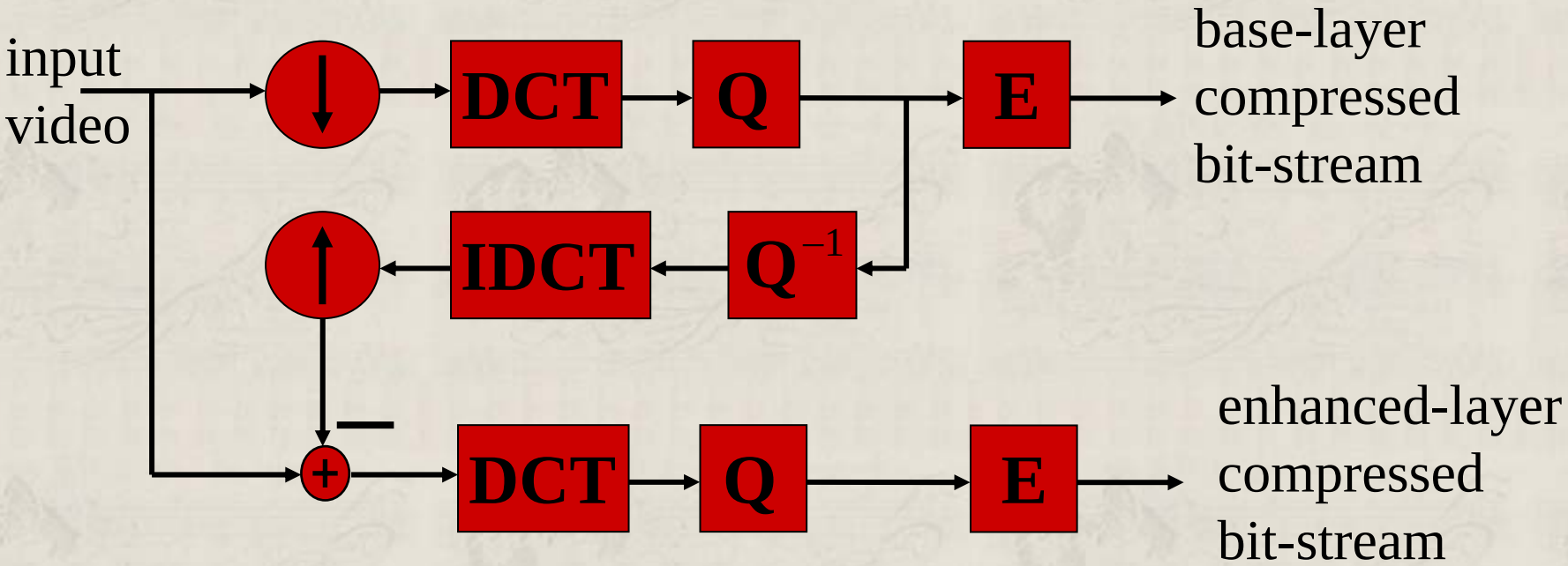


Spatial Scalability



- N layers of spatial scalability

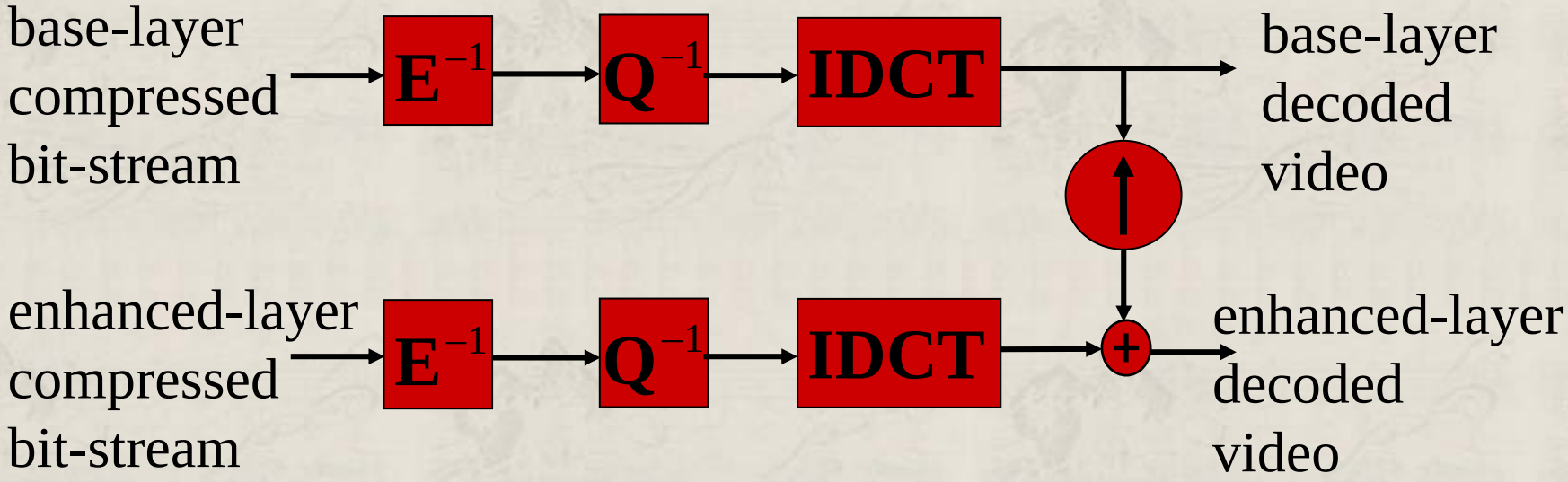
Encoder: Spatial/Temporal Scalability



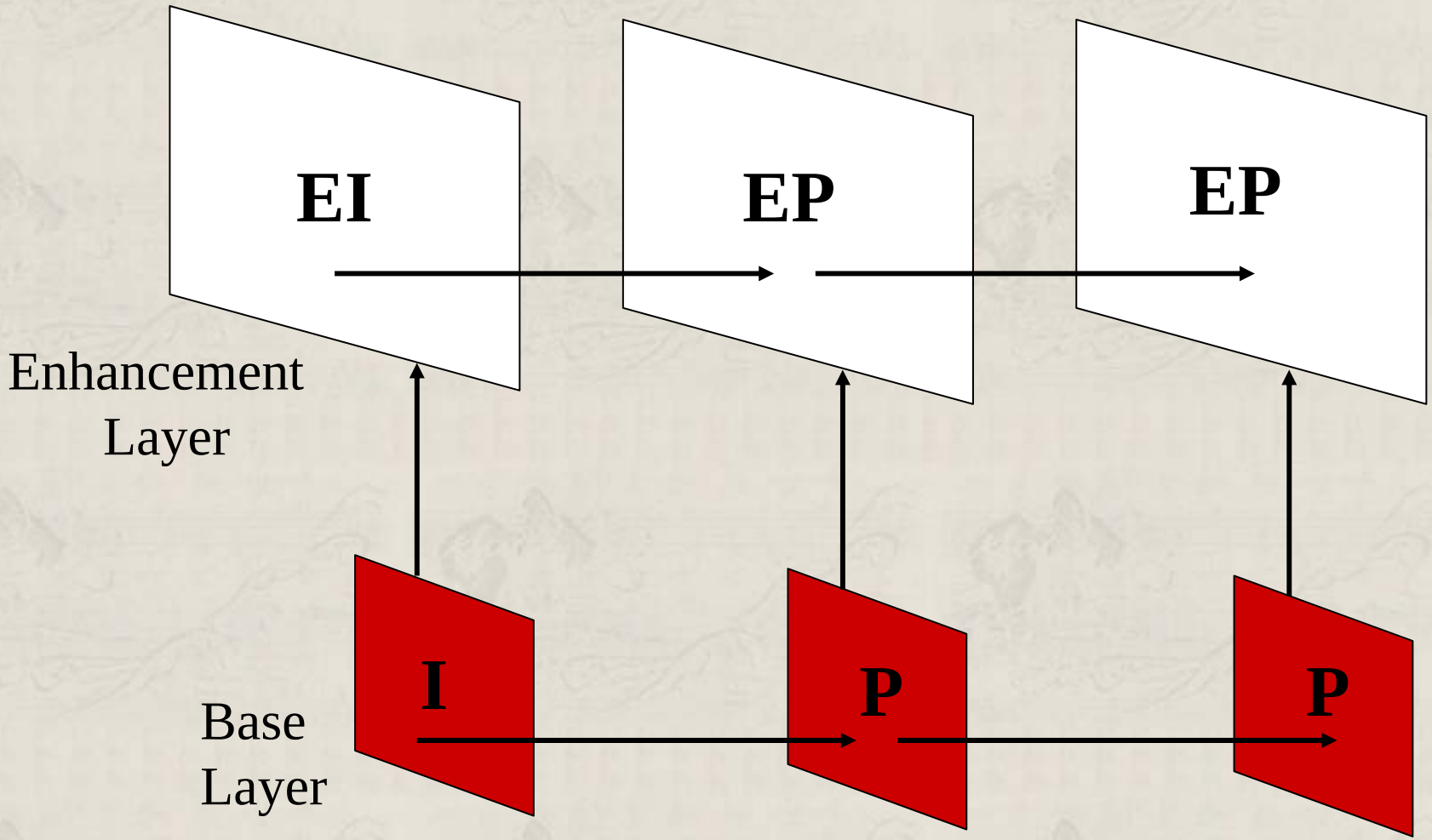
Spatial/temporal decimator

Spatial/temporal interpolator

Decoder: Spatial/Temporal Scalability

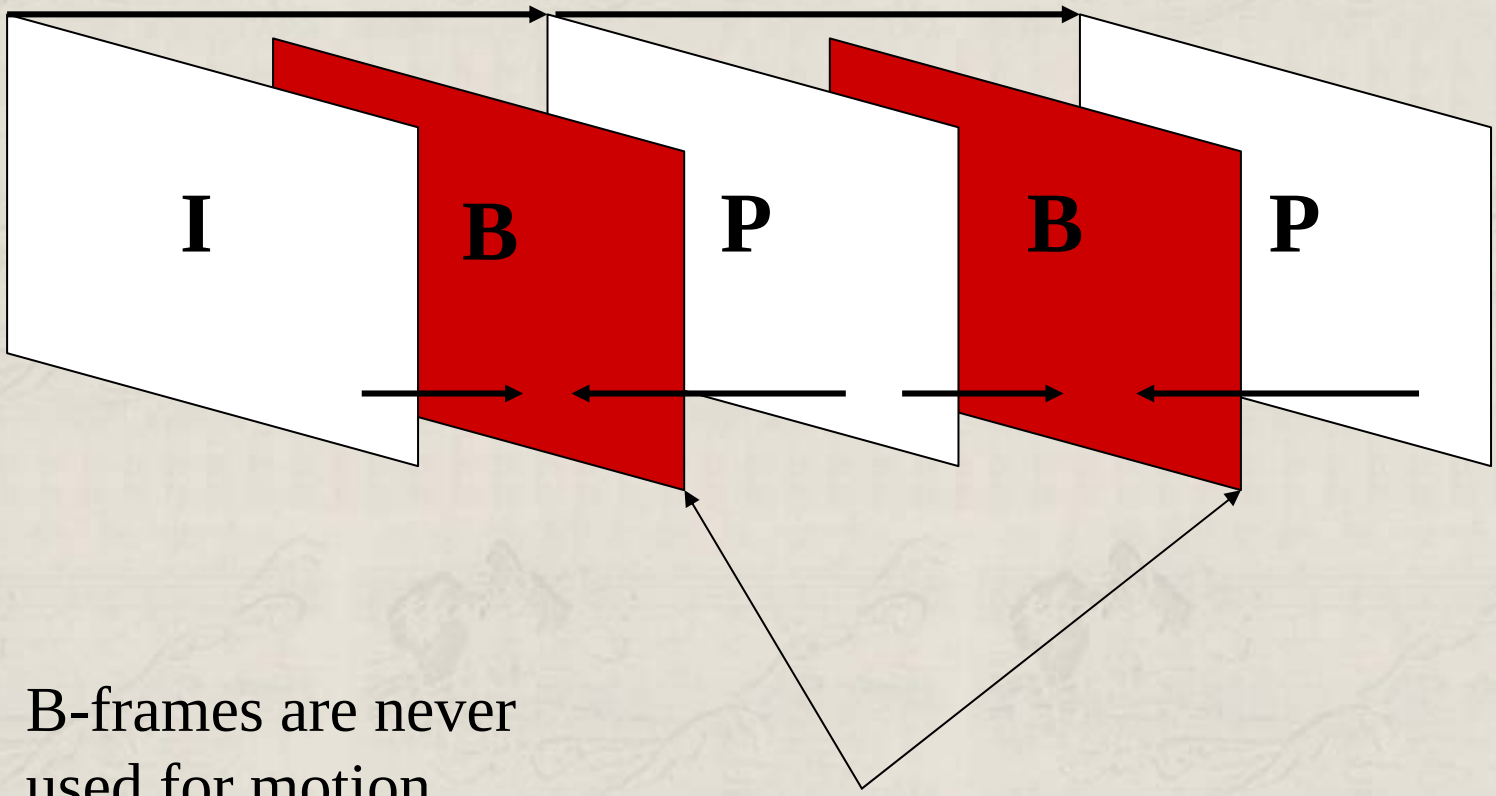


MEMC & Spatial Scalability



- Careful with encoder/decoder mismatch which causes drifting

MEMC & Temporal Scalability



- B-frames are never used for motion estimation and compensation

Enhancement Layer

Reference

- K. R. Rao and J. J. Hwang, *Techniques and Standards for Image Video and Audio Coding*, Prentice-Hall, 1996
- V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards: Algorithms & Architecture*, Kluwer Academic, 1997
- Y. Wang, J. Ostermann, and Y. Q. Zhang, *Video Processing and Communications*, Prentice Hall, 2002
- J. L. Mitchell et al, *MPEG Video: Compression Standard*, Chapman & Hall, 1997