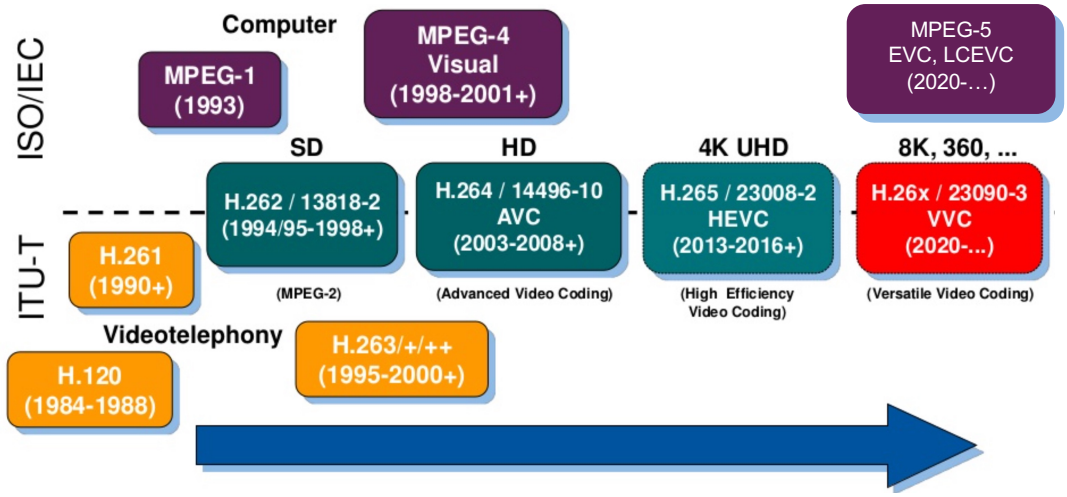
Video Compression Standards I

Prof. Touradj Ebrahimi

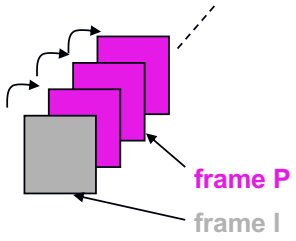
- Three international standards setting organizations are mainly responsible for creation of widely deployed video compression approaches
  - International Telecommunication Union (ITU-T)
  - International Organization for Standards (ISO)
  - International Electrotechnical Commission (IEC)

Video compression standards



Predictive coding with motion compensation

- INTRA (I) frame coding



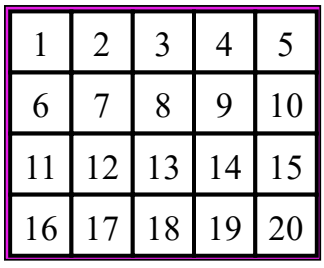
- INTER (P) frame coding
  - Prediction from a previously coded frame (I, P)

Predictive coding with motion compensation

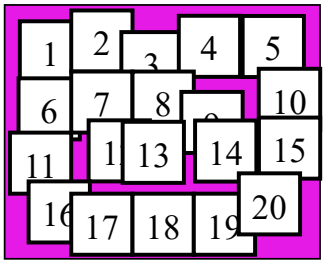
- Backward
  - Predict where the pixels were in a previous frame
- Forward
  - Predict where the pixels will be in a next frame

Forward

- Produces holes or overlapping regions



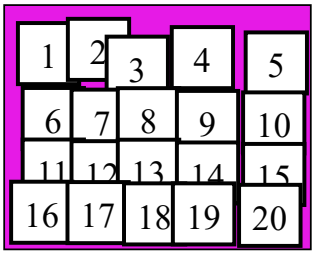
Current frame



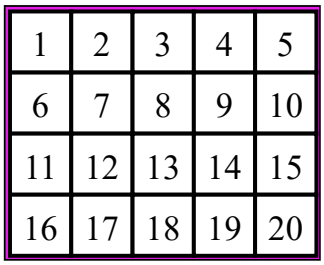
Future predicted frame

Backward

- No holes or overlapping regions
- Popular approach to predictive coding
- Produces a reasonable coding delay



Previous frame

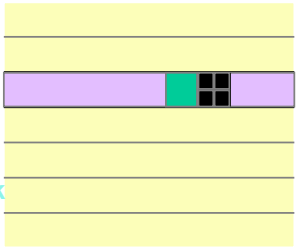


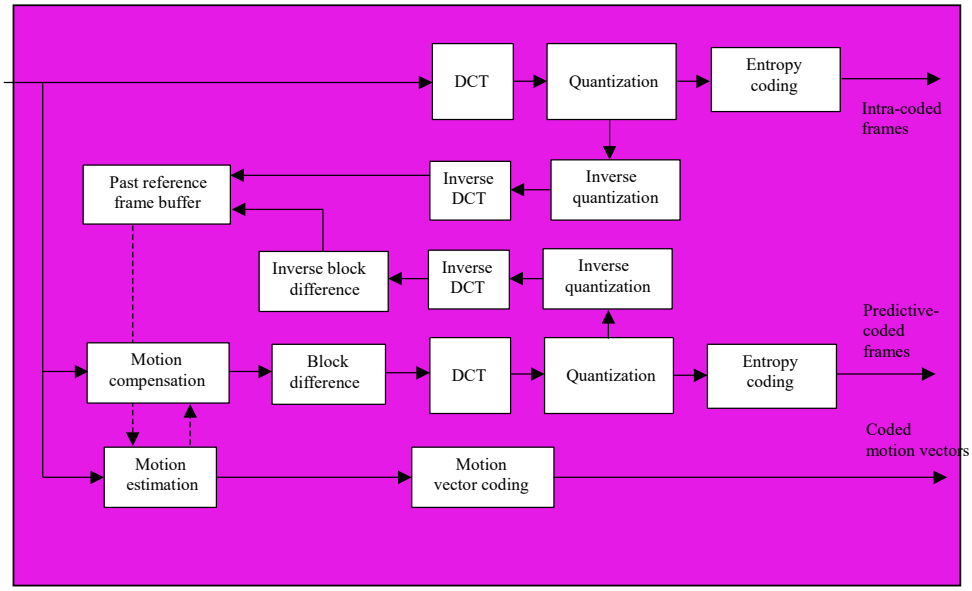
Current predicted frame

Bitstream syntax

A typical compressed video bitstream is composed of:

- Several frames of different types:
  - ...IPBBPBBPBBP…I...
  - Every frame is composed of:
    - Group of Macroblocks (synchronization)
      - *Macroblocks (motion+texture)*
        - *Blocks (DCT)*



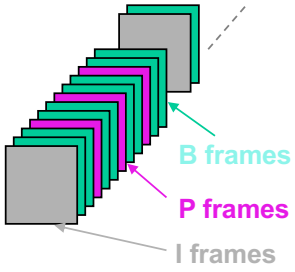


Recommendation ITU-T H.261 (p X 64)

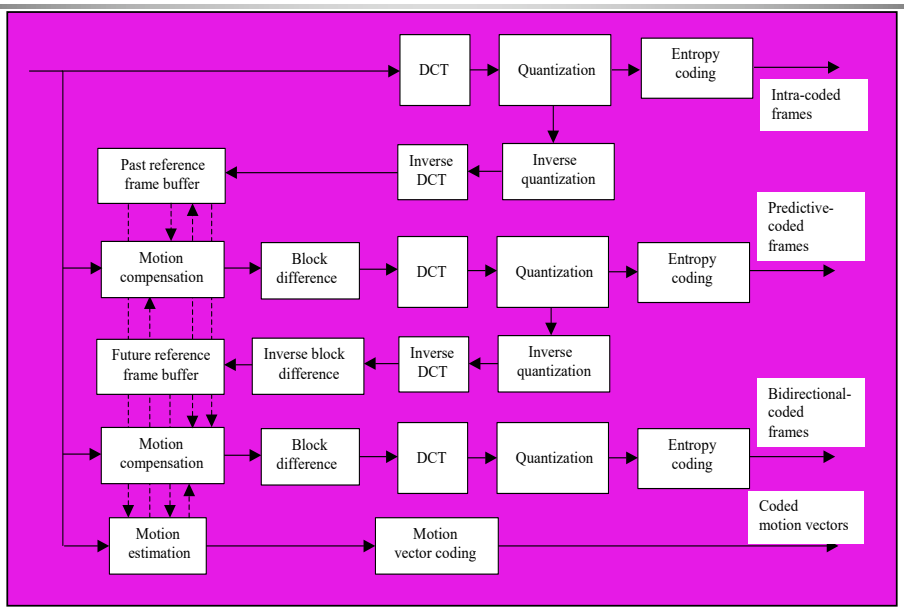
- Only compressed bitstream syntax and decoding process are standardized
- Compressed video is made of I and P frames
- I frames are compressed based on DCT similar to JPEG (not compatible)
- P frames are predicted from previously coded frames (I or P)
- Prediction is based on a motion compensation approach applied to blocks of 16 X 16 pixels

Predictive and interpolative coding with motion compensation

- INTRA coded frames:
  - Random access
  - Robustness to errors
- INTER (P) coded frames:
  - Prediction from a previously coded frame (I, P)
- BI-INTER (B) coded frames:
  - Prediction from previous and/or future coded frames (I, P)



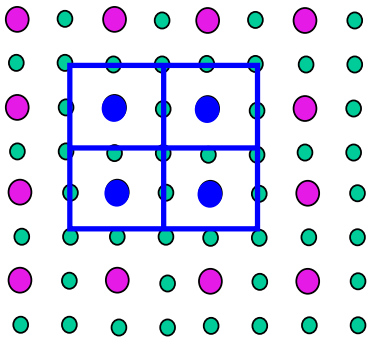
MPEG-1



MPEG-1 video coding at 1.5 Mbps

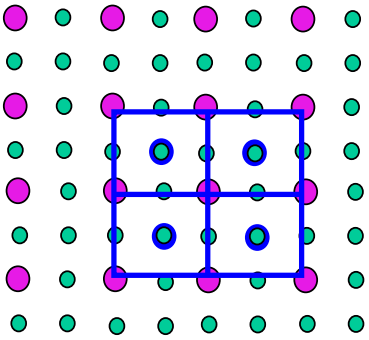
- Only the compressed video bitstream syntax and decoding process are standardized
- Compressed bitstream is composed of I, P, and B frames
- Compressed bitstream is divided into a group of frames starting from an I frame
- P frames are predicted from previously coded frames (I or P)
- B frames are predicted from previous and future frames (I or P)
- Prediction is based on motion compensation on blocks of size 16 X 16 pixels with ½ pixel resolution

- Motion estimation at ½ pixel resolution
  - Motion estimation from an interpolated reference frame

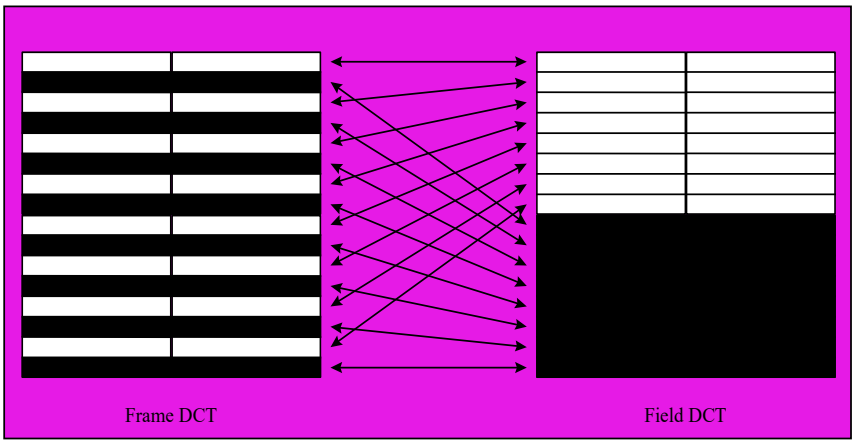


Fine motion estimation

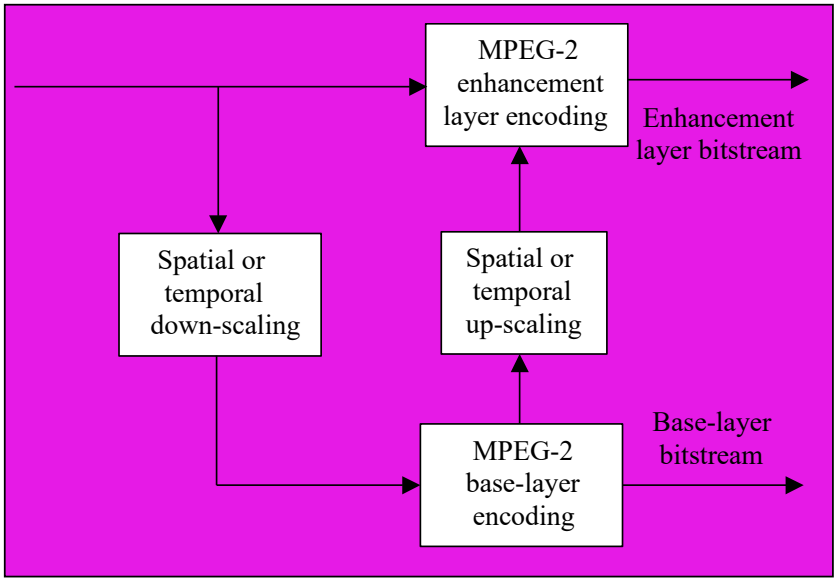
- Motion estimation at ½ pixel resolution
  - Motion estimation from an interpolated reference frame



Interlaced video compression



Scalable video coding with MPEG-2



MPEG-2 – Highlights

MPEG-2 Generic Video Compression

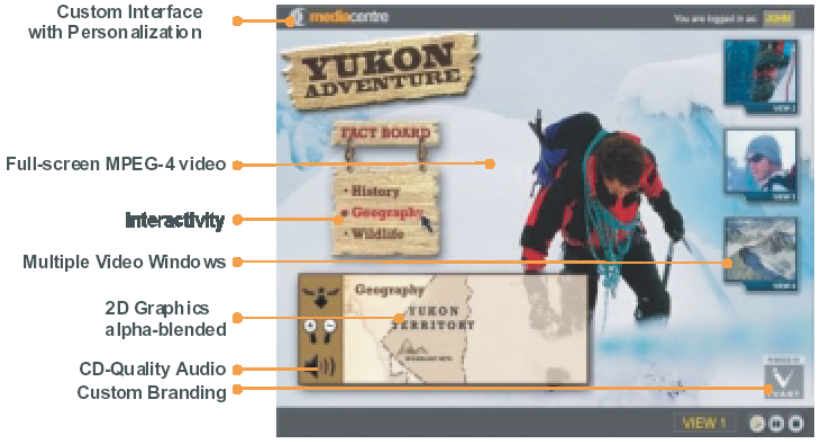
- Only the bitstream syntax and decoding process are standardized
- Several modes are used to efficiently compress interlaced video (*field based modes*)
- Generic coding at various frame sizes and bitrates
- Temporal, spatial and quality (PSNR) scalability

Recommendation ITU-T H.263

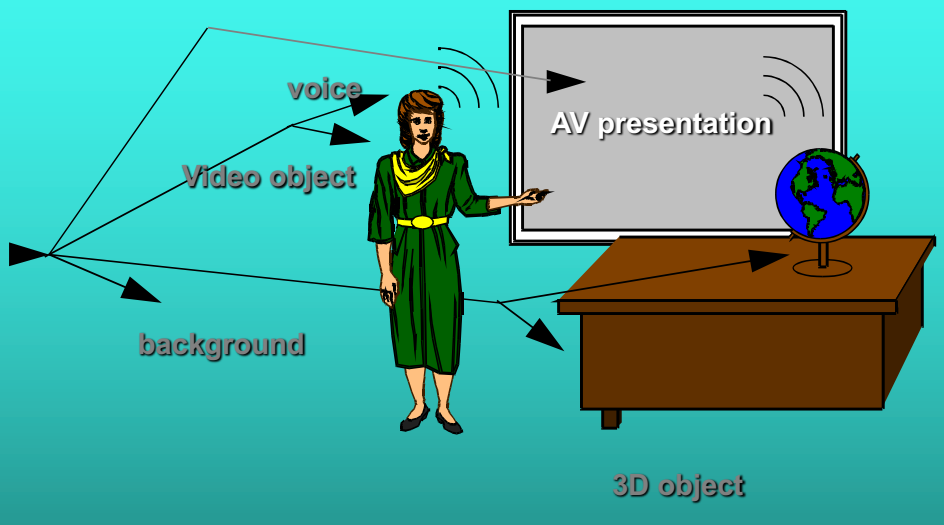
- Compressed video is composed of frames of types I, P, and B
- Advanced motion compensation (1 mv per blocks 16x16 or 4 times 1 mv per block 8x8)
- Motion estimation and compensation at ½ pixel
- Extended motion estimation (outside of the region of definition of reference frames)
- Arithmetic coding can be used instead of Huffman (option)

- MPEG-4 proposes tools for **conventional** coding of audiovisual content with excellent **robustness** against errors
- MPEG-4 proposes an architecture where several **audiovisual objects** can be coded and decoded independently
- Audiovisual objects can be **composed** in a flexible way to create different scenes
- MPEG-4 handles both **natural** and **synthetic** objects efficiently

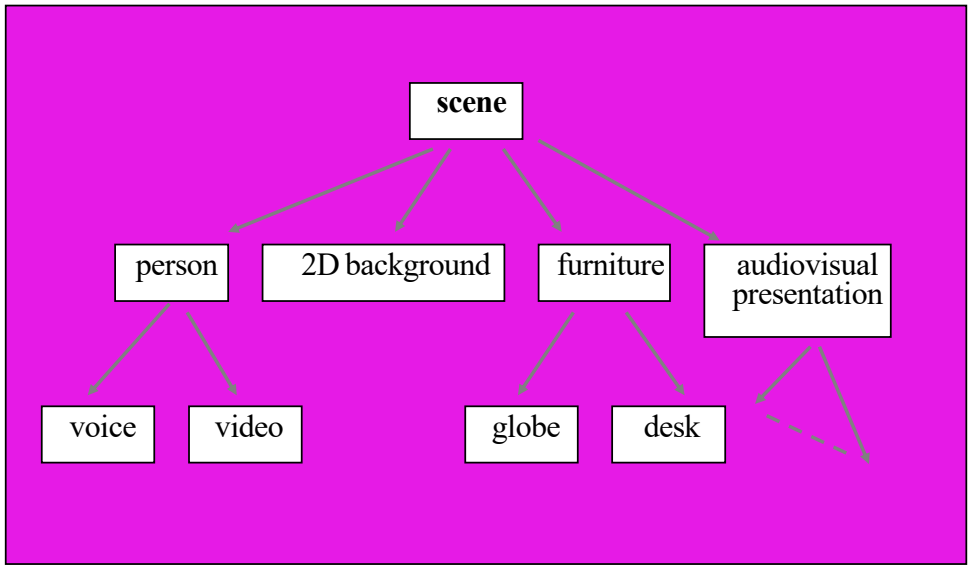
MPEG-4 Example



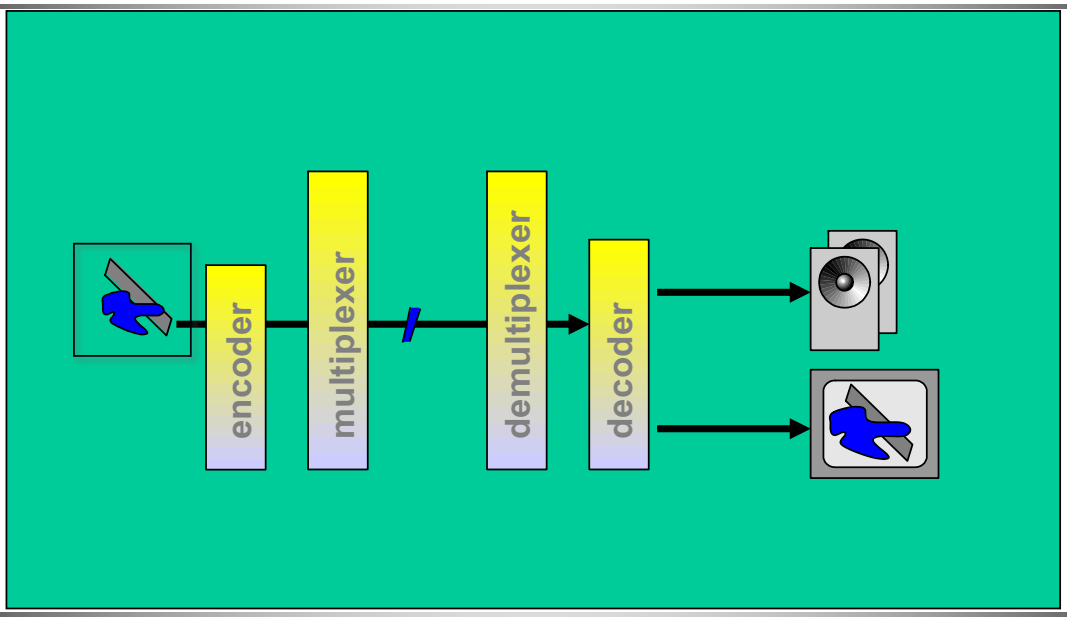
Example of a typical MPEG-4 scene



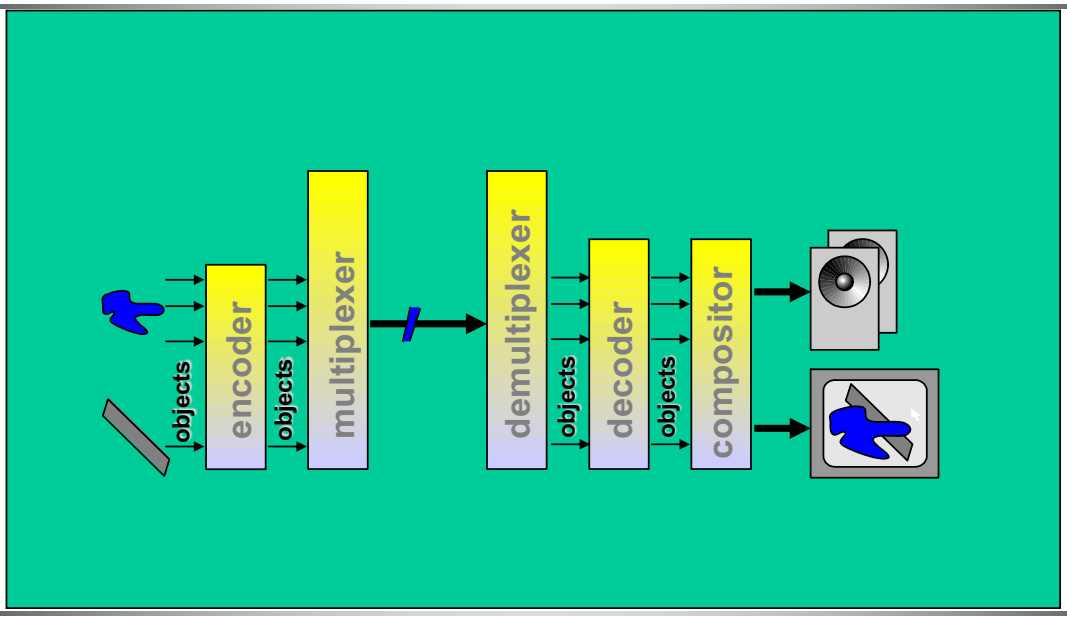
Conceptual representation of an MPEG-4 scene



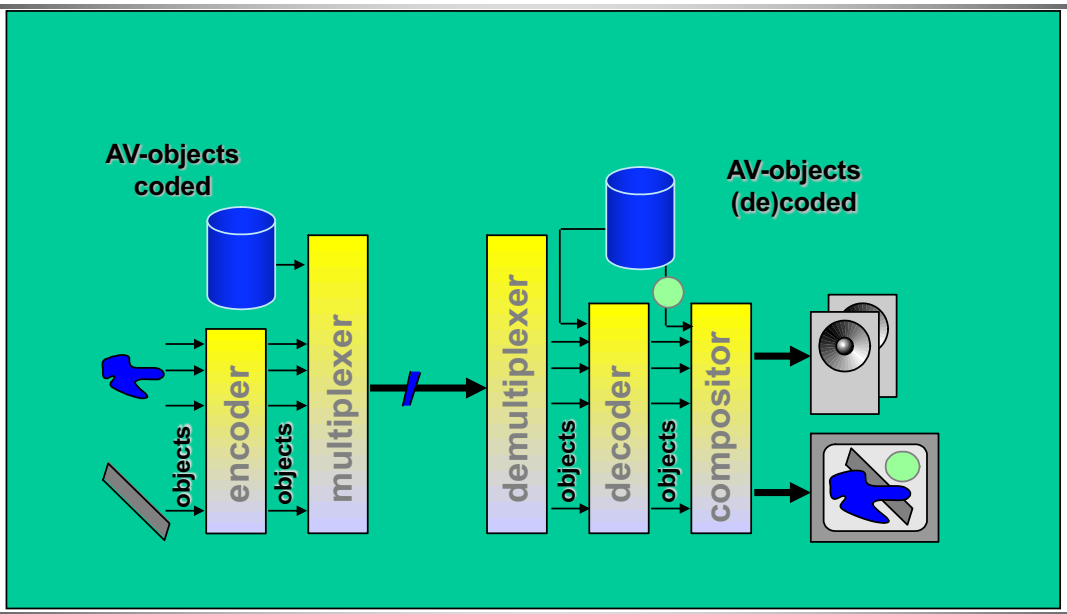
MPEG-4 acting as a conventional coding system



MPEG-4 emulating a conventional coding system



MPEG-4 deploying its object-based functionalities

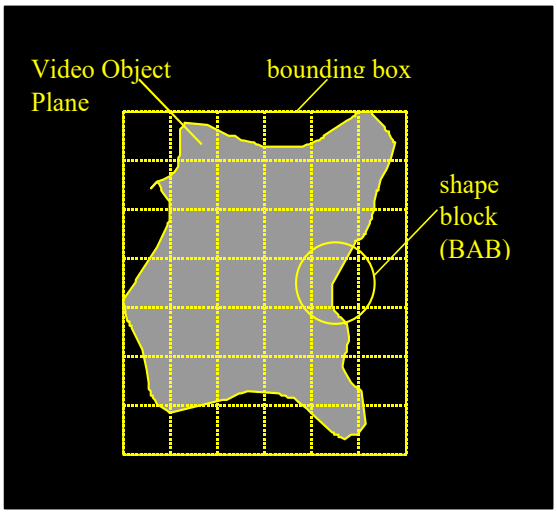


List of major tools for natural video coding in MPEG-4

Binary shape	
Padding	
Motion compensation	Overlapped motion compensation Advanced motion compensation
Quantization	Method 1 Method 2
AC/DC prediction	Non-linear
Scanning	Type 1 Type 2
I, P, B modes	
Temporal scalability	Slice synchronization
Spatial Scalability	Extended header code
Error resilience	Data partitioning Reversible VLC
Static sprites	Basic Low delay
Interlaced coding	
12-bit video	
Static texture	Scalable

MPEG-4 shape coding (1/3)

- Arbitrary shaped video object
  - texture (3 color components YUV)
  - Transparency plane

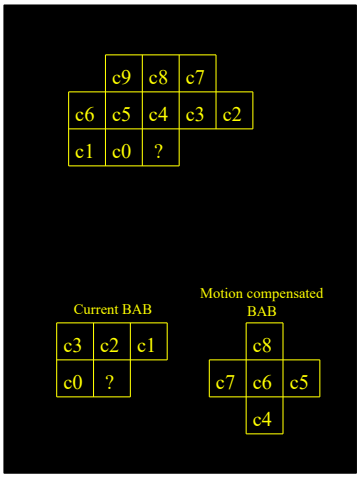


MPEG-4 shape coding (2/3)

- Coding modes
  - Opaque
  - Transparent
  - No-update
  - Intra Context based Arithmetic Encoding
  - Inter Context based Arithmetic Encoding
- Lossless
- Lossy
  - Motion compensation without update
  - Subsampling by a factor of 2 or 4

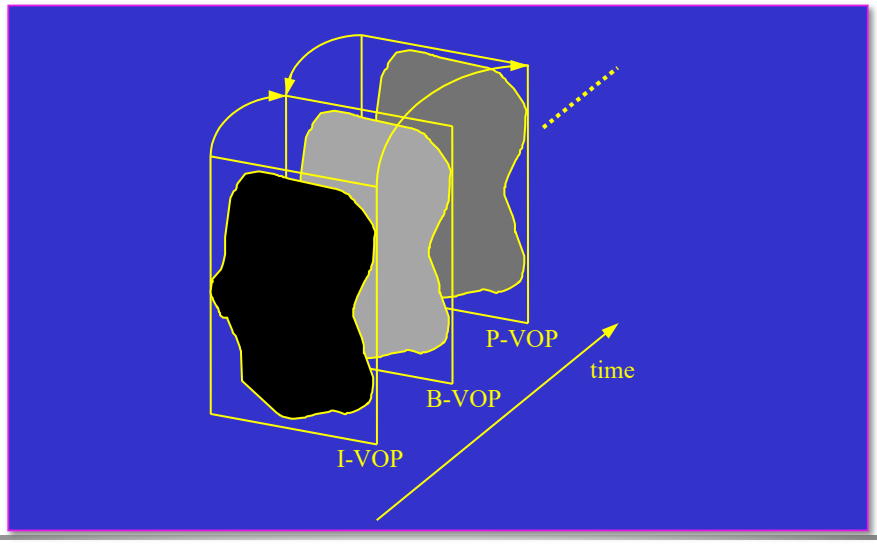
MPEG-4 shape coding (3/3)

- Context based Arithmetic Encoding
  - Intra
  - Inter

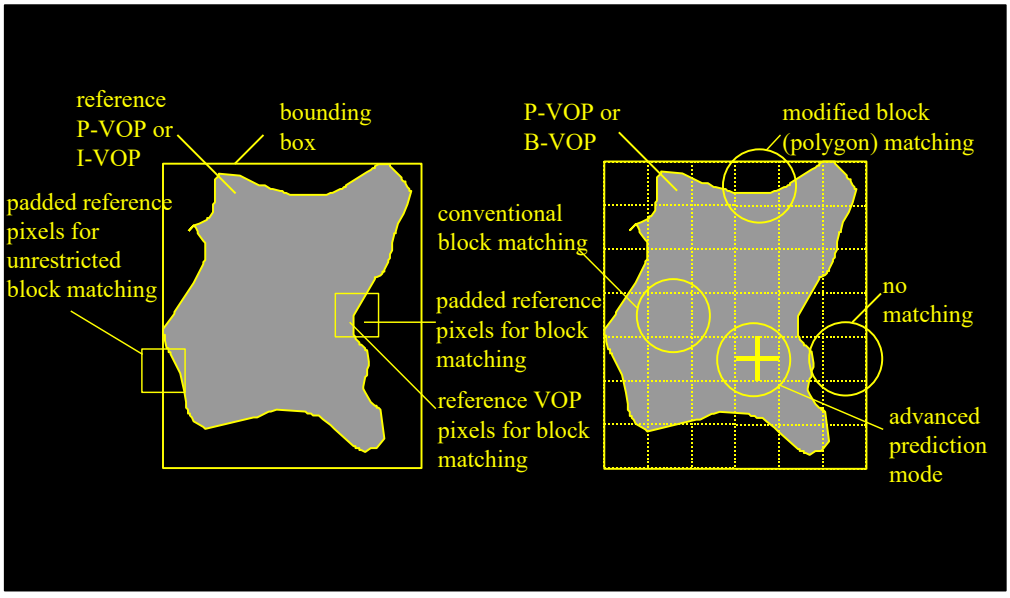


Motion compensation (1/3)

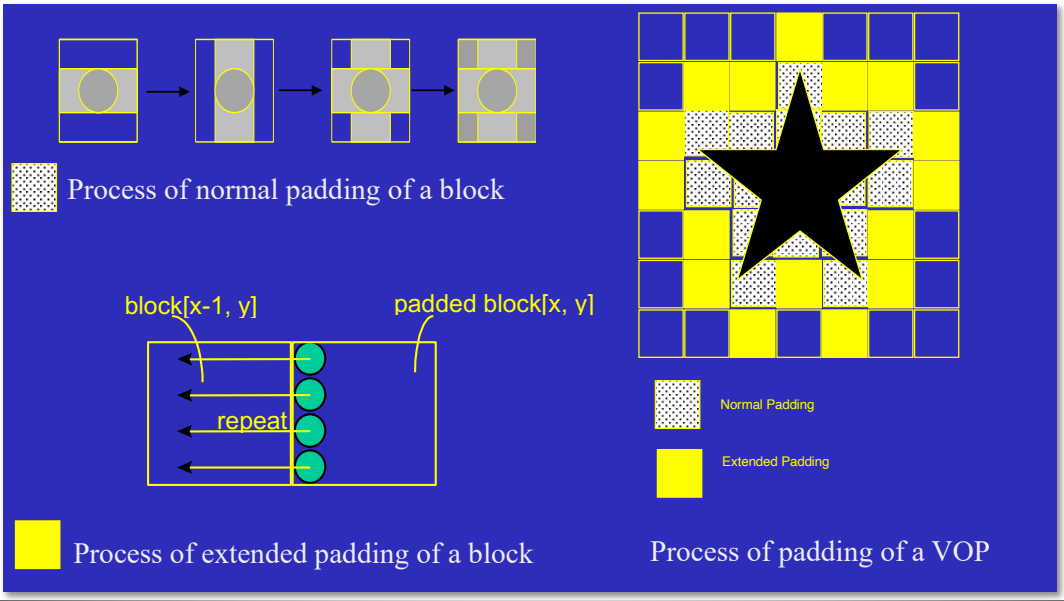
An object can be coded in one of the following modes: I, B, P



Motion compensation (2/3)



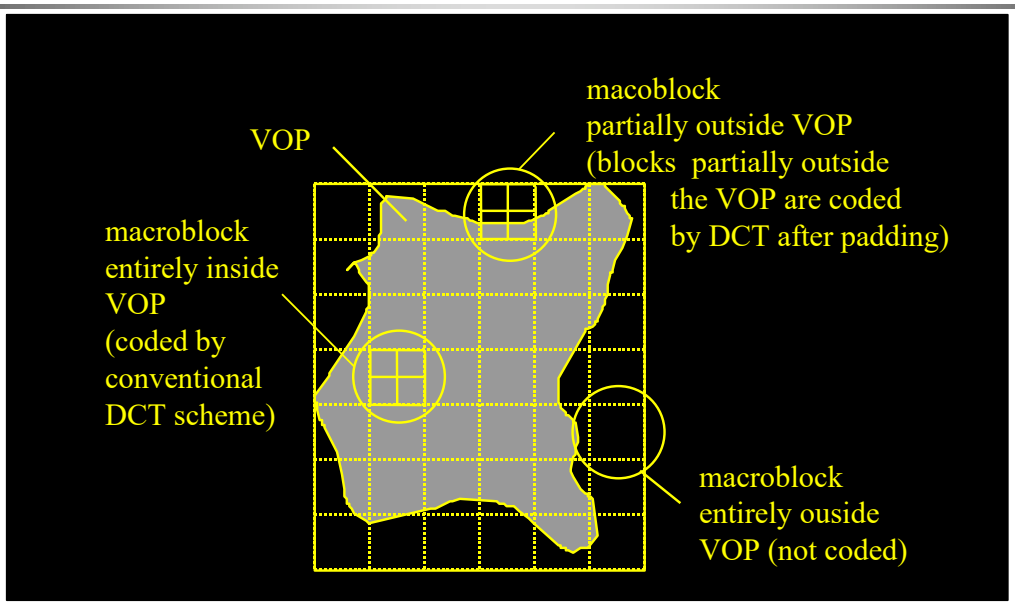
Motion compensation - padding (3/3)



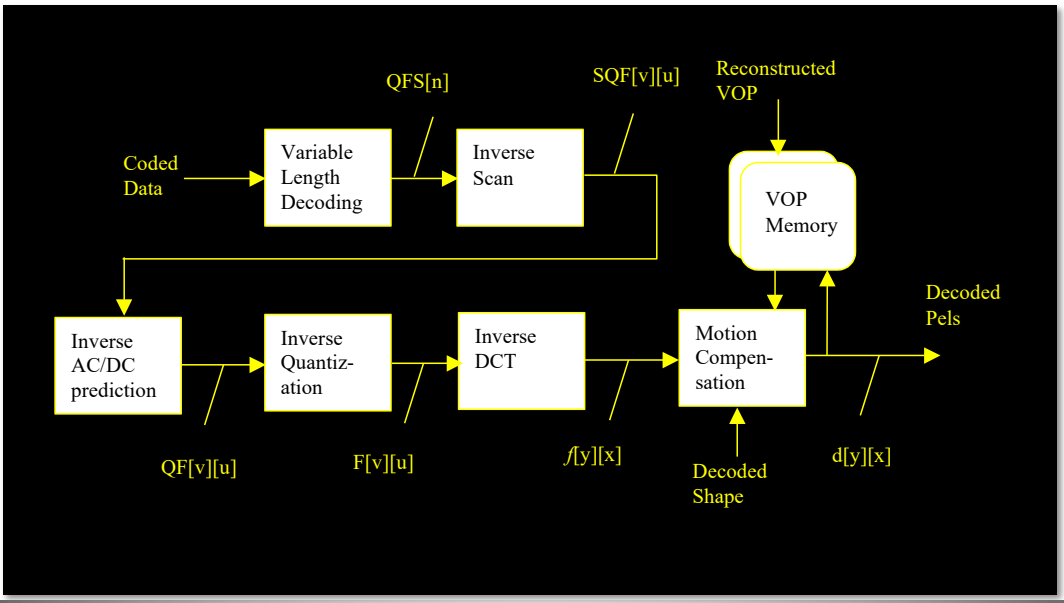
Padding summary

- Object can change and some matched parts in the original reference can be transparent
- Transparent pixels are padded and rthe esulting VOP is used as reference
- Macroblocks in current object are matched against the reference

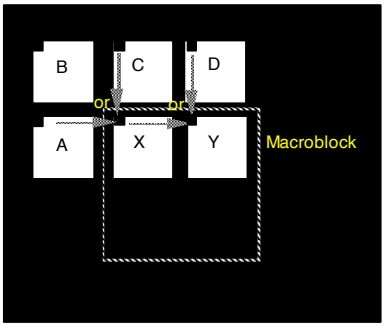
Texture coding (1/2)



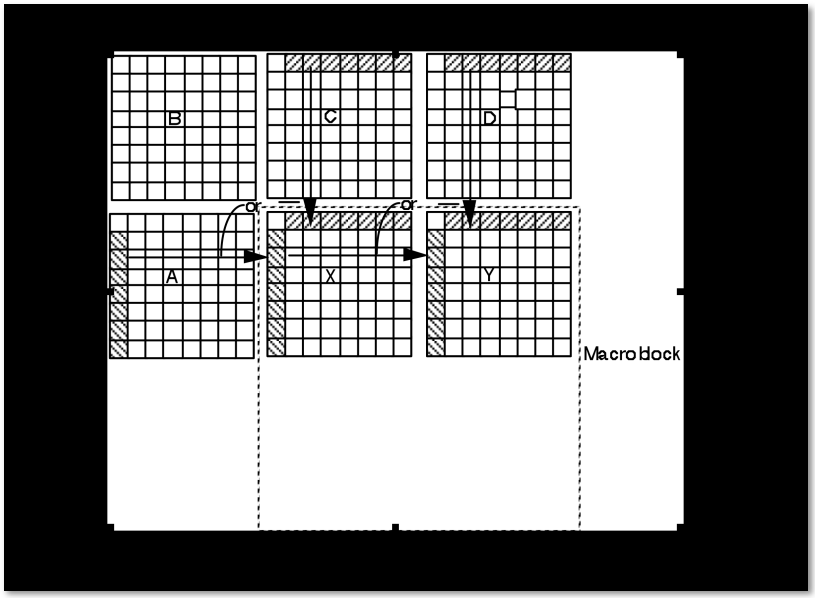
Texture coding (2/2)



Adaptive DC prediction



Adaptive AC prediction



Scanning of coefficients

0	1	2	3	10	11	12	13
4	5	8	9	17	16	15	14
6	7	19	18	26	27	28	29
20	21	24	25	30	31	32	33
22	23	34	35	42	43	44	45
36	37	40	41	46	47	48	49
38	39	50	51	56	57	58	59
52	53	54	55	60	61	62	63

Alternate-Horizontal scan

0	4	6	20	22	36	38	52
1	5	7	21	23	37	39	53
2	8	19	24	34	40	50	54
3	9	18	25	35	41	51	55
10	17	26	30	42	46	56	60
11	16	27	31	43	47	57	61
12	15	28	32	44	48	58	62
13	14	29	33	45	49	59	63

Alternate-Vertical scan

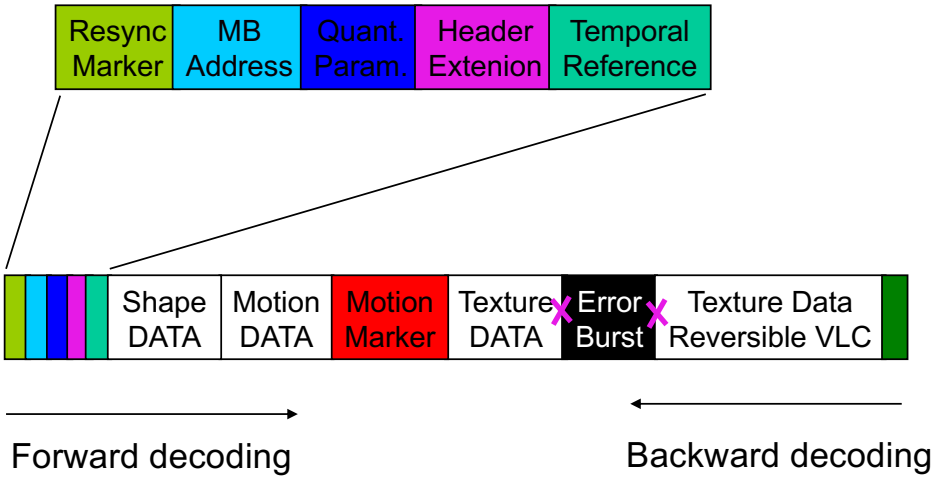
0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

zig-zag scan

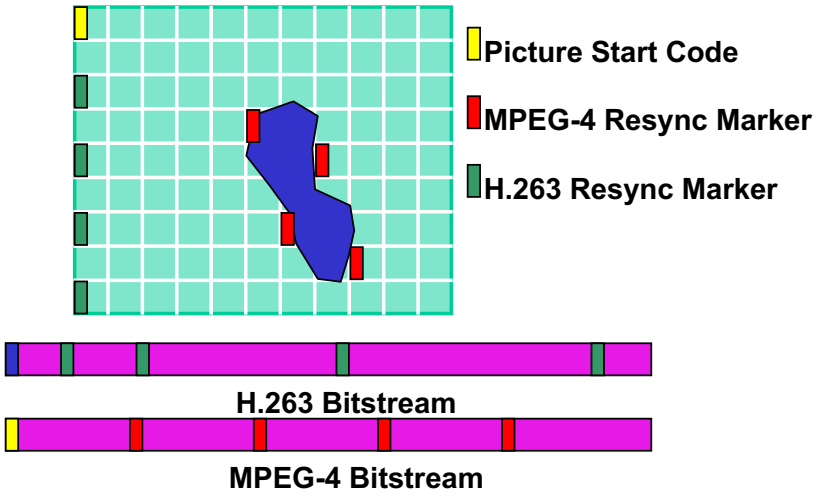
Error resilience tools (1/3)

- Resynchronization markers
- Extended header code
- Data partitioning
- Reversible VLCs

Error resilience tools (2/3)

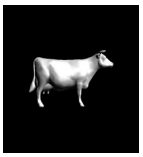


Error resilience tools (3/3)



Synthetic objects in MPEG-4

- Face coding and its animation
- Body coding and its animation
- Photo-realistic 3D model coding
- Animated 3D model coding



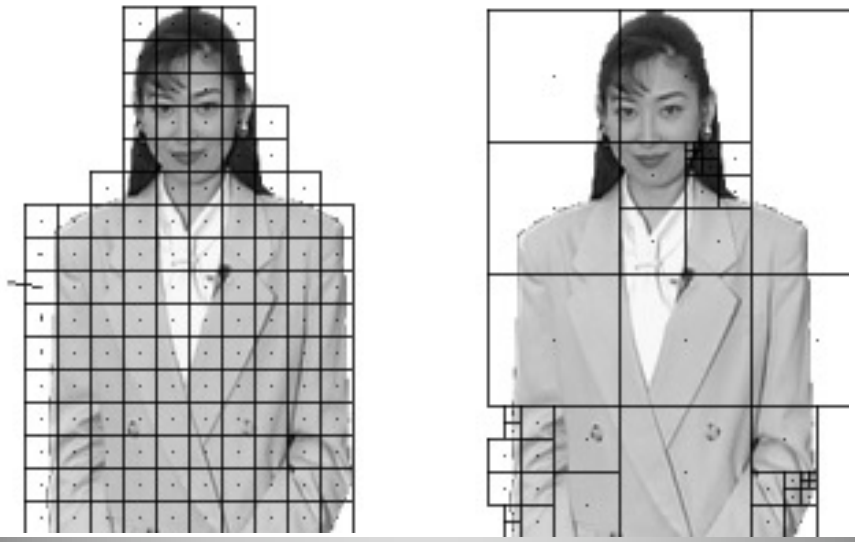
Advanced Video Coding H.264

- Also known as MPEG-4 Part 10 pr AVC
- Became international standard in 2003
- Similar general structure as in H.263
  - But differences in each module
- Better compression efficiency when compared to MPEG-2 and MPEG-4 part 2
  - Mainly due to a better prediction
- But more complex
  - Remains reasonable especially as decoder

H.264 – Highlights

- Compressed video is decomposed into I, P and B frames but also SI et SP (*stream switching*)
- Spatial prediction (in I frames)
  - By 4x4 (luma, 9 modes), 16x16 (luma, 4 modes) or 8x8 (chroma, 4 modes) blocks
- Temporal prediction (in B or P frames)
  - Motion estimation at ¼ pixel accuracy
  - Variable motion block sizes: 16x16, 16x8, 8x16, 8x8, 8x4, 4x8 et 4x4
  - Multiple reference frames (I, B or P)
  - Weighted prediction

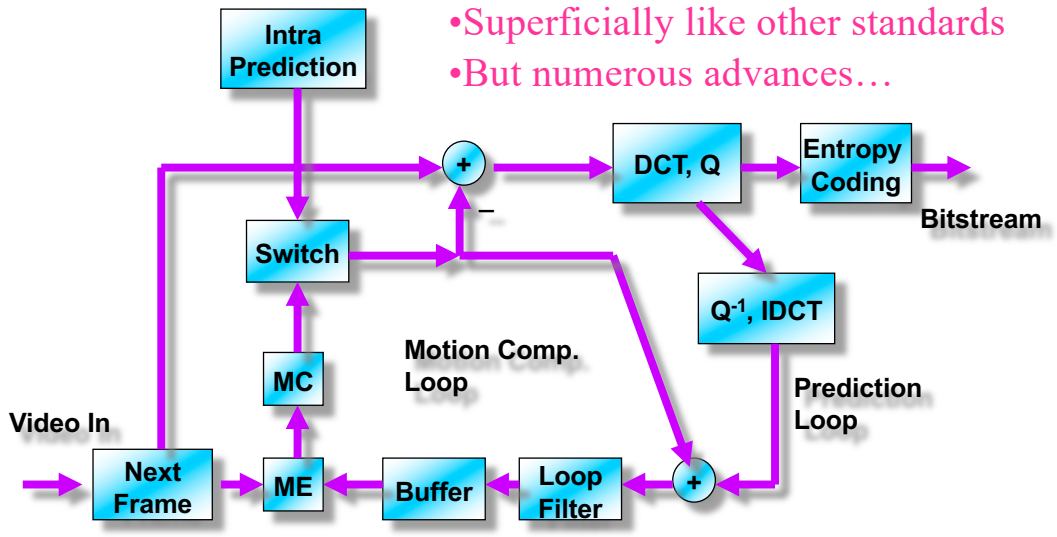
Fixed size vs. variable size block matching



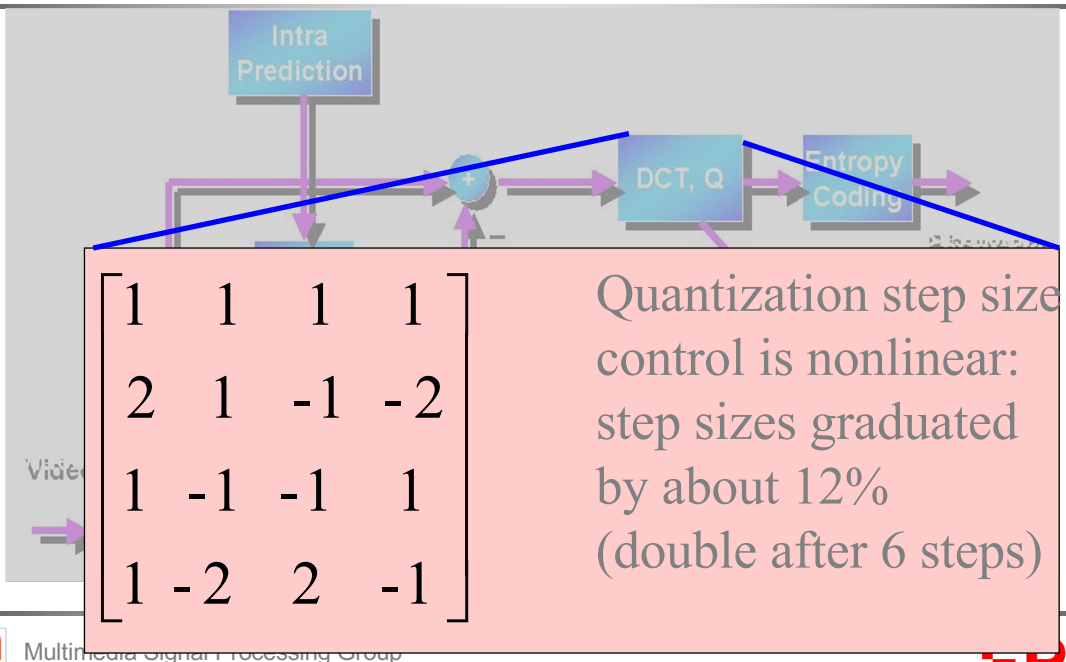
H.264 – Highlights

- Integer transform in 4x4 blocks
  - Integer approximation of DCT
- Context-adaptive binary arithmetic coding (CABAC)
  - More complex but 5~10 % better compression efficiency
- Context adaptive entropy coding (CAVLC)
  - Faster and simpler but less efficient than CABAC

Block Diagram: H.264 Encoder



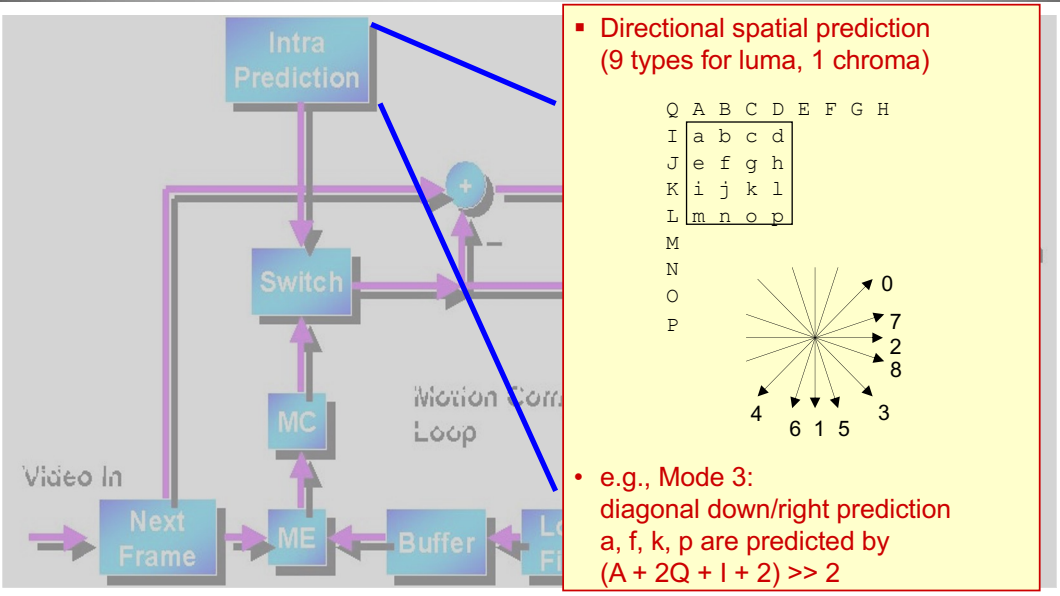
A 16-Bit 4x4 DCT



$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$$

Quantization step size control is nonlinear: step sizes graduated by about 12% (double after 6 steps)

Intra Pred., 9 Modes

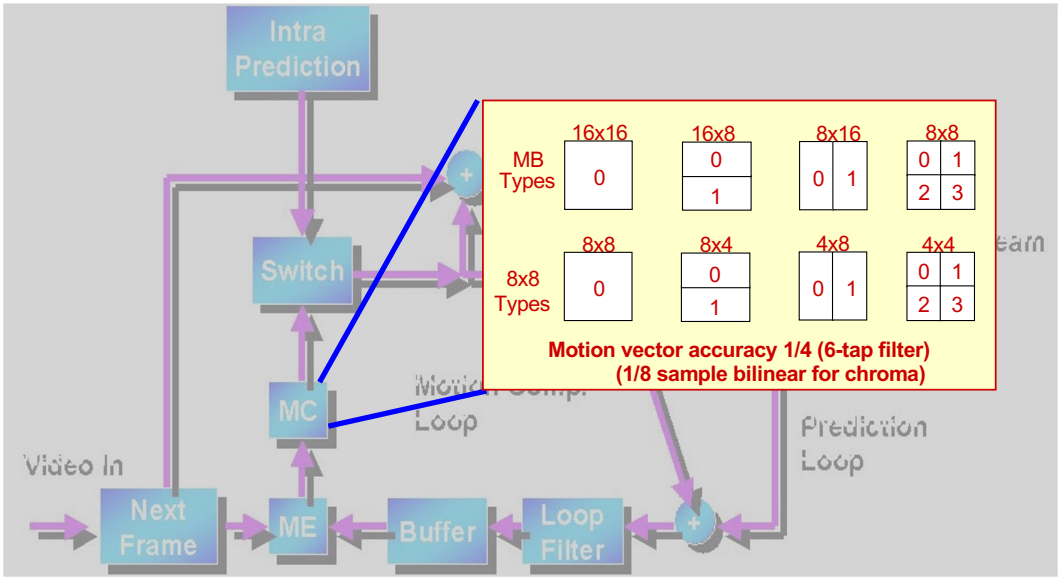


- Directional spatial prediction (9 types for luma, 1 chroma)

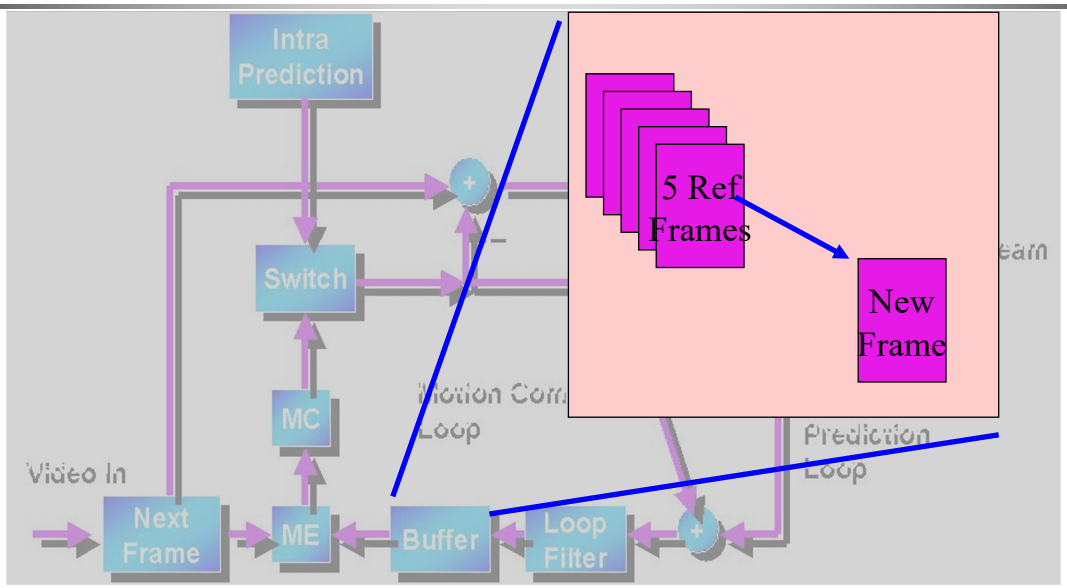
```
  Q A B C D E F G H
I   a b c d
J   e f g h
K   i j k l
L   m n o p
M
N
O
P
```


- e.g., Mode 3: diagonal down/right prediction a, f, k, p are predicted by (A + 2Q + I + 2) >> 2

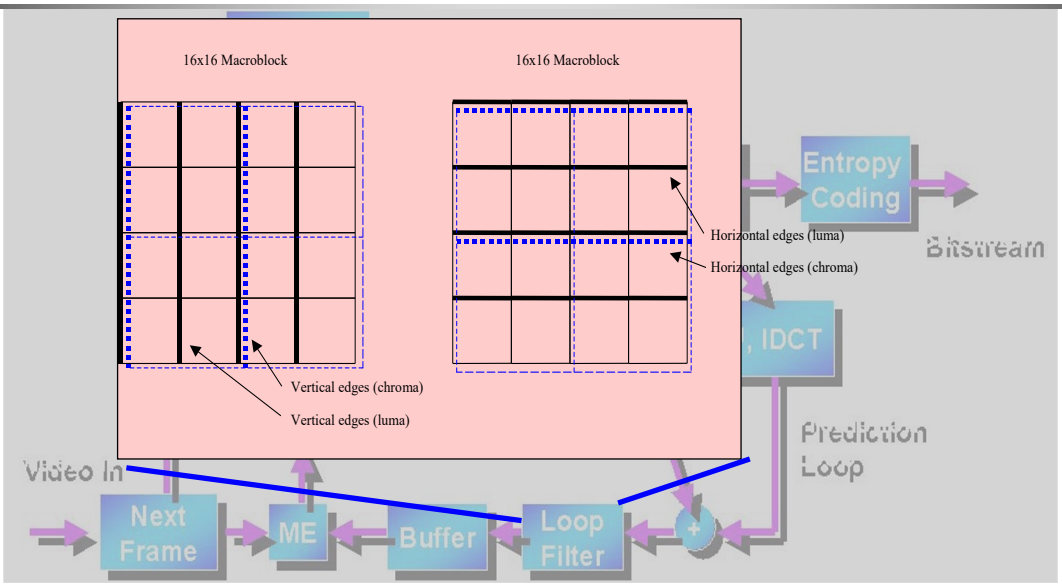
Flexible Block MC
- 7 Segmentation Shapes



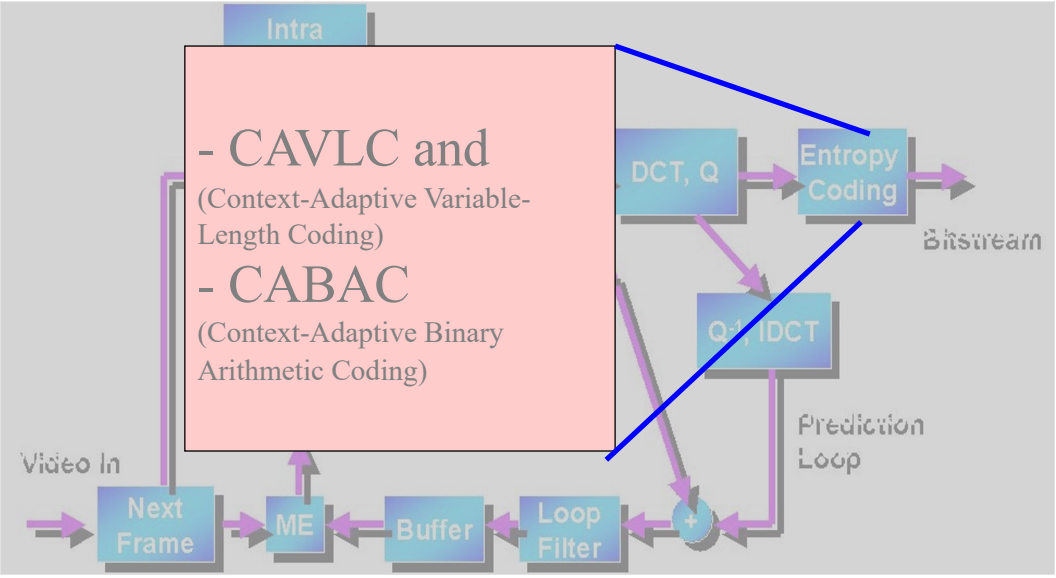
5 Ref. Frames



In-Loop Deblocking



Two Entropy Codes (VLC and CABAC Arith. Code)



Weighted Prediction

- Inter-picture prediction with scaling and offset
- Encoder-specified or timing-derived weights
- Multiple weights per picture supported
- Tremendous advantage for fade-in, fade-out, and cross-fades

Test – Mobile & Calendar (CIF)