Video Coding

# Video Compression: Principles, Practice, and Standards

John G. Apostolopoulos
Streaming Media Systems Group
HP Labs, Palo Alto, CA
japos@hpl.hp.com

Video Coding

# Goals for Today’s Talk

Goals:

- Provide brief overview of image & video coding and key standards
- Assume no prior knowledge of compression
- Explain the “alphabets soup” of coding standards, acronyms, etc

Video Coding

# Outline

- Why do we need compression?
- Image compression
  - Color processing, DCT, etc
- Video compression
  - Exploit temporal dimension of video signal • Prediction
  - Generic (MPEG-type) video coder architecture
- Current video compression standards
  - What are they?
  - What do the standards specify?
- Brief overview of H.264 / MPEG-4 AVC video coding standard

Video Coding

# Motivation for Compression: Example of HDTV Video Signal

- Problem:
  - Raw video contains an immense amount of data
  - Communication and storage capabilities are limited & expensive
- Example HDTV video signal:
  - 720x1280 pixels/frame, progressive scanning at 60 frames/s:

$$\left(\frac{720\times1280\, pixels}{frame}\right)\left(\frac{60\; frames}{\sec}\right)\left(\frac{3 colors}{pixel}\right)\left(\frac{8 bits}{color}\right)=1.3Gb/s$$

  - 20 Mb/s HDTV channel bandwidth

$\rightarrow$ Requires compression by a factor of 70 (equivalent to 0.35 bits/pixel)

# Example Video Applications and Required Compression Ratios



| Video Format | Pixels/frame | Fps | bits/pixel | Raw bit rate | Channel bit rate | Required compression ratio |
|---|---|---|---|---|---|---|
| HDTV | 720 x 1280 | 60 | 24 | 1.3 Gb/s | 20 Mb/s (HDTV) | ~ 70 |
| SDTV | 480 x 720 | 30 | 24 | 250 Mb/s | 5 Mb/s (DVD) | ~ 50 |
| CIF | 288 x 352 | 30 | 24 | 73 Mb/s | 384 kb/s (3G) | ~ 190 |
| QCIF | 144 x 176 | 15 | 24 | 9 Mb/s | 64 kb/s (3G) | ~ 140 |

Wireless video today

Wireless video soon

Lots of compression required!!!

# Achieving Compression

- Reduce redundant or repeated information
  - Temporal: Adjacent frames highly correlated
  - Spatial: Nearby pixels are often correlated
  - Color space: RGB components are correlated
  - → Relatively straightforward to exploit
- Reduce irrelevant information
  - Perceptually unimportant information, i.e. what the human visual system (HVS) can not see
  - → Difficult to model and exploit

Video Coding

# Spatial and Temporal Redundancy

- Why can video be compressed?
  - Video contains much spatial and temporal redundancy.

- Spatial redundancy: Neighboring pixels are similar
- Temporal redundancy: Adjacent frames are similar

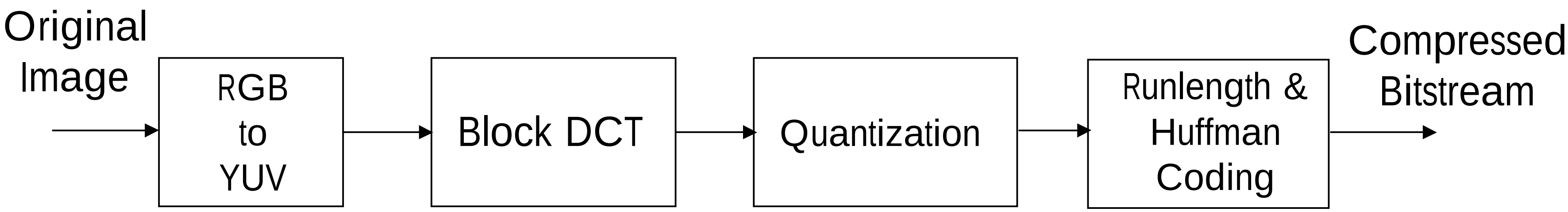
Compression is achieved by exploiting the spatial and temporal redundancy inherent to video

# Outline

- Why do we need compression?
- Image compression
  - Color processing, DCT, etc
- Video compression
  - Exploit temporal dimension of video signal • Prediction
  - Generic (MPEG-type) video coder architecture
- Current video compression standards
  - What are they?
  - What do the standards specify?
- Brief overview of H.264 / MPEG-4 AVC video coding standard

Video Coding

# Image Compression: Overview




- Coding an image (single frame):
  - RGB to YUV color-space conversion
  - Partition image into 8x8-pixel blocks
  - 2-D DCT of each block
  - Quantize each DCT coefficient
  - Runlength and Huffman code the nonzero quantized DCT coefficients

→ Basis for the JPEG Image Compression Standard

→ JPEG-2000 uses wavelet transform and arithmetic coding

# Color Space Processing

- Important properties
  - RGB components are highly correlated
  - HVS perception differs for luminance than for chrominance
- Goal: Convert RGB to a different color space where these properties can be exploited
- 3x3 Linear color-space transformation:
  - R G B $\rightarrow$ Y U V
  - Y: Luminance
  - U & V: Chrominance

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} .299 & .587 & .114 \\ -.147 & -.289 & .436 \\ .615 & -.515 & -.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

# Color Space Processing (cont.)

Advantages of color space conversion:

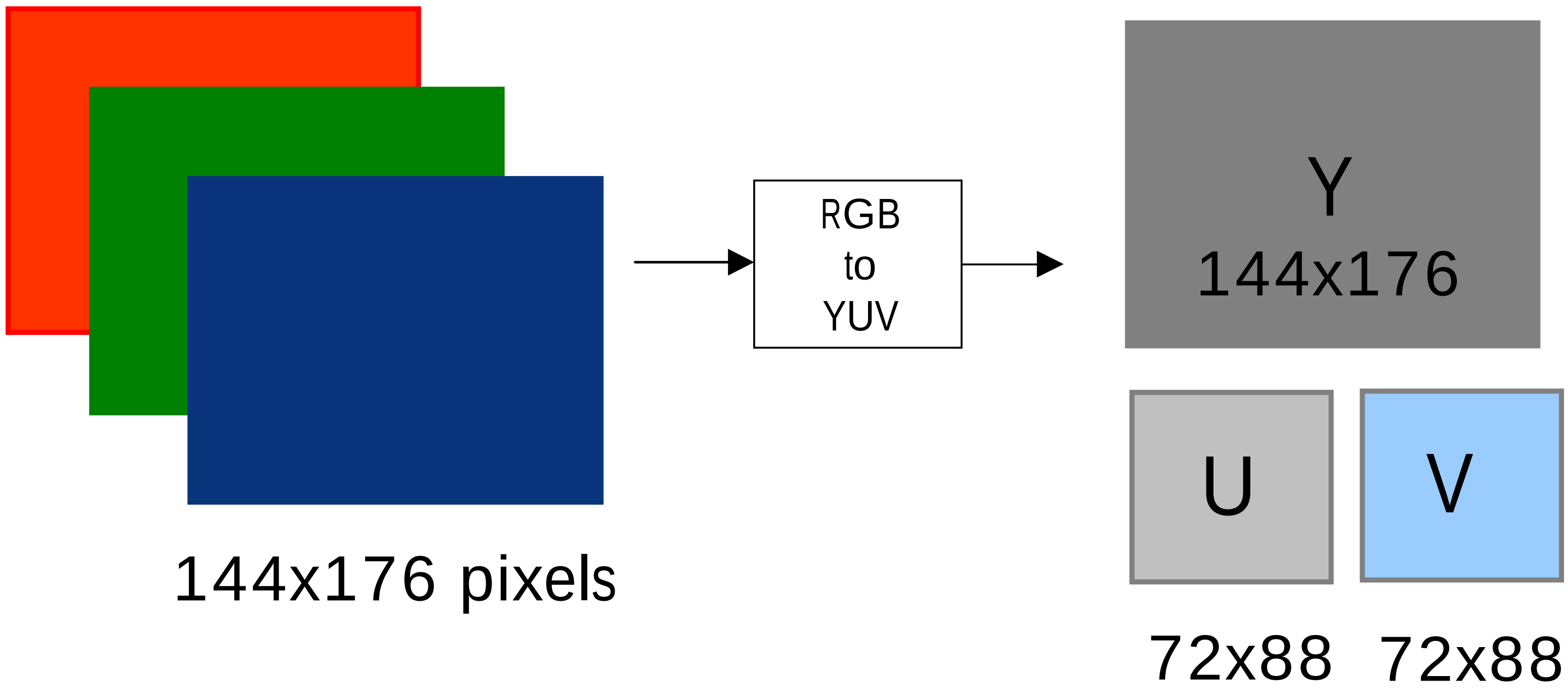
- HVS has lower spatial frequency response to U and V than to Y
  - $\rightarrow$ Reduce sampling density for U and V
- HVS has lower sensitivity to U and V than to Y
  - $\rightarrow$ Quantize U and V more coarsely

- Reasonable assumption: An RGB image requires 3x bit rate of B&W image (single-color image)
- Key result: RGB image only requires ~1.25x bit rate

# Color Space Processing (cont.)

RGB color components are:

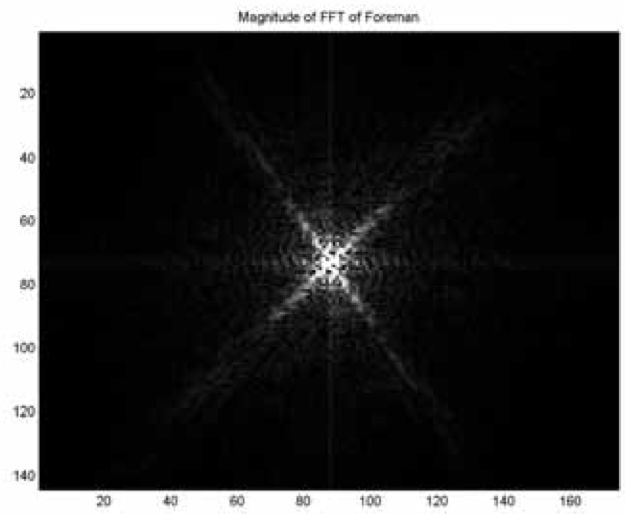
1) Converted to YUV
2) U and V components are subsampled by 2x2



Video Coding

# Transform Image Coding


First frame of Foreman Sequence




Magnitude of FFT of Foreman

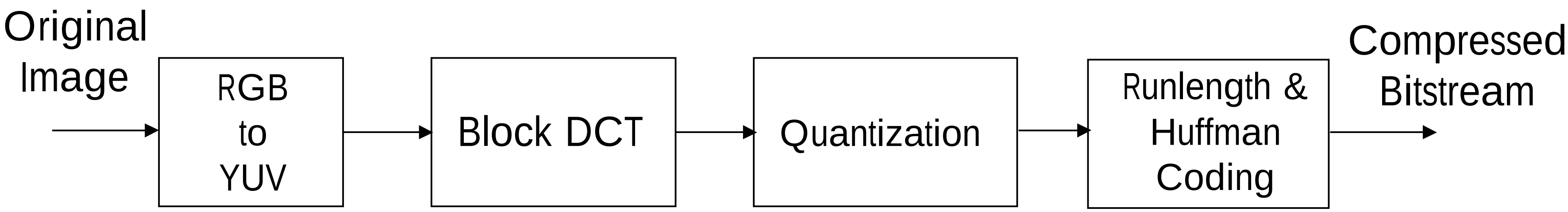
- Goal: Transform image so that most of the information (energy) is concentrated into only a small fraction of the coefficients
  - Coding only these small fraction of the coefficients and discarding the rest can often lead to excellent reconstructed quality
  - $\rightarrow$ The more energy compaction the better!
- Enables perceptual processing (exploiting HVS response to different frequency components)

# Spatial Processing: Block DCT

- Block Discrete Cosine Transform (DCT)
  - Split image into 8x8 pixel blocks
  - Each block independently transformed and processed
    - Compute 8x8 2-D DCT of each block
    - Quantize and encode each block
- Advantages:
  - Enables simple, spatially-adaptive processing
  - Reduces computation and memory requirements
  - Suitable for parallel processing
- Basic building block for most current image and video compression standards including:
  - JPEG, MPEG-1/2/4, H.261/3/4

Video Coding

# Image Compression: Summary




- Coding an image (single frame):
  - RGB to YUV color-space conversion
  - Partition image into 8x8-pixel blocks
  - 2-D DCT of each block
  - Quantize each DCT coefficient
  - Runlength and Huffman code the nonzero quantized DCT coefficients

→ Basis for the JPEG Image Compression Standard

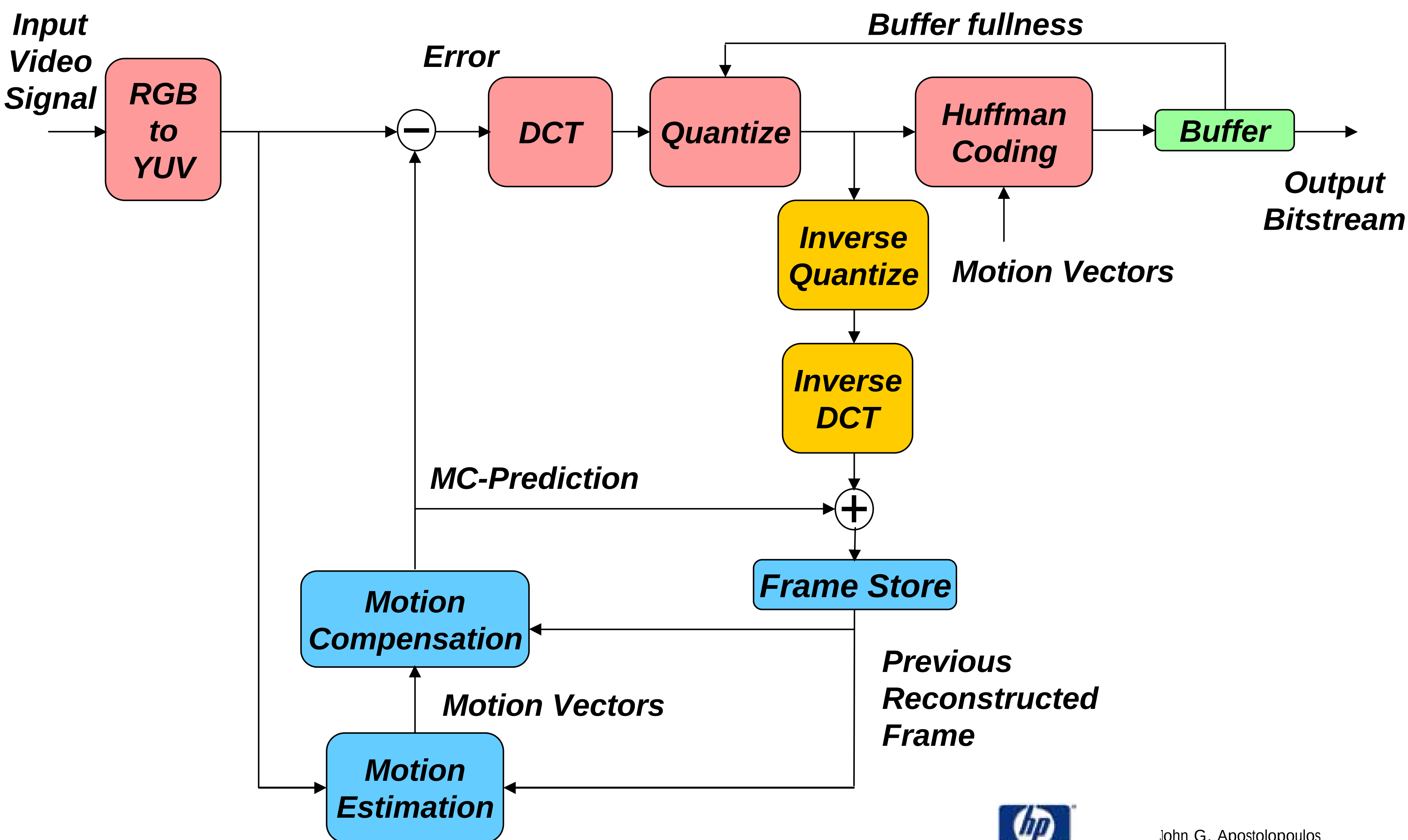
→ JPEG-2000 uses wavelet transform and arithmetic coding

Video Coding

# Outline

- Why do we need compression?
- Image compression
  - Color processing, DCT, etc
- Video compression
  - Exploit temporal dimension of video signal • Prediction
  - Generic (MPEG-type) video coder architecture
- Current video compression standards
  - What are they?
  - What do the standards specify?
- Brief overview of H.264 / MPEG-4 AVC video coding standard

# Example Video Encoder



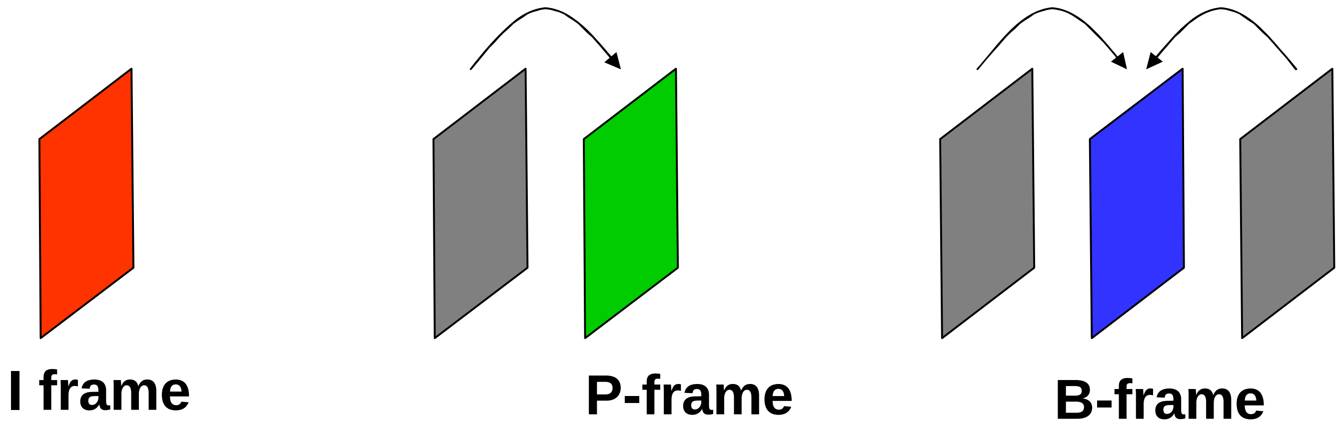
# Video Compression

- Video: Sequence of frames (images) that are related
  - Related along the temporal dimension
- Main addition over image compression
  - Temporal redundancy
  - Usually high frame rate: Significant temporal redundancy

  $\rightarrow$ Video coder must exploit the temporal redundancy

# Video Compression

- Goal: Exploit the temporal redundancy
- Predict current frame based on previously coded frames
- Three types of coded frames:
  - I-frame: Intra-coded frame, coded independently of all other frames
  - P-frame: Predictively coded frame, coded based on previously coded frame
  - B-frame: Bi-directionally predicted frame, coded based on both previous and future coded frames

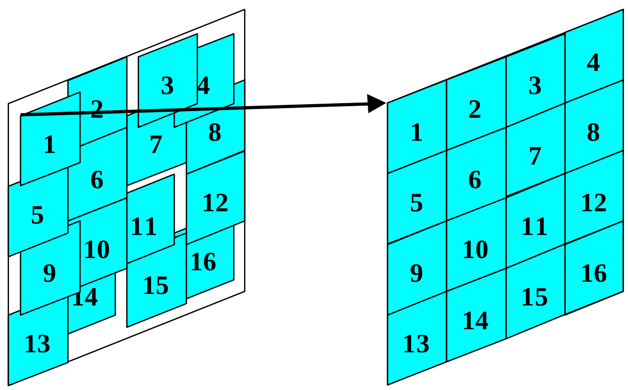


# Prediction for Two Consecutive Frames: Block-Matching Motion Estimation & Motion-Compensated Prediction

Previously Coded Frame
(Reference Frame)

Current Frame
(To be Predicted)

Block-matching overview:
1) Split current frame into 16x16-pixel blocks
2) Find best match for each block from prior frame



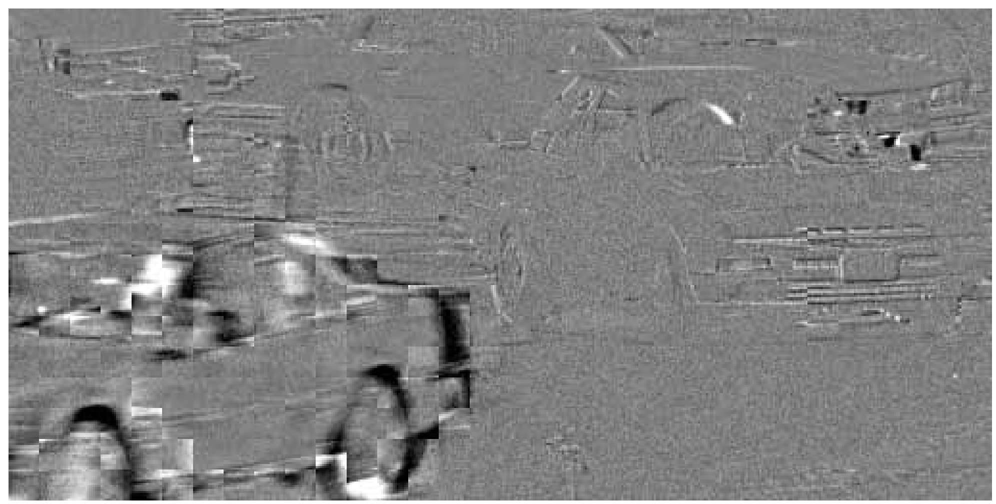

***Reference Frame*** ***Predicted Frame***

Video Coding

# Example: MC-Prediction for Two Consecutive Frames (cont.)

Prediction of Current Frame

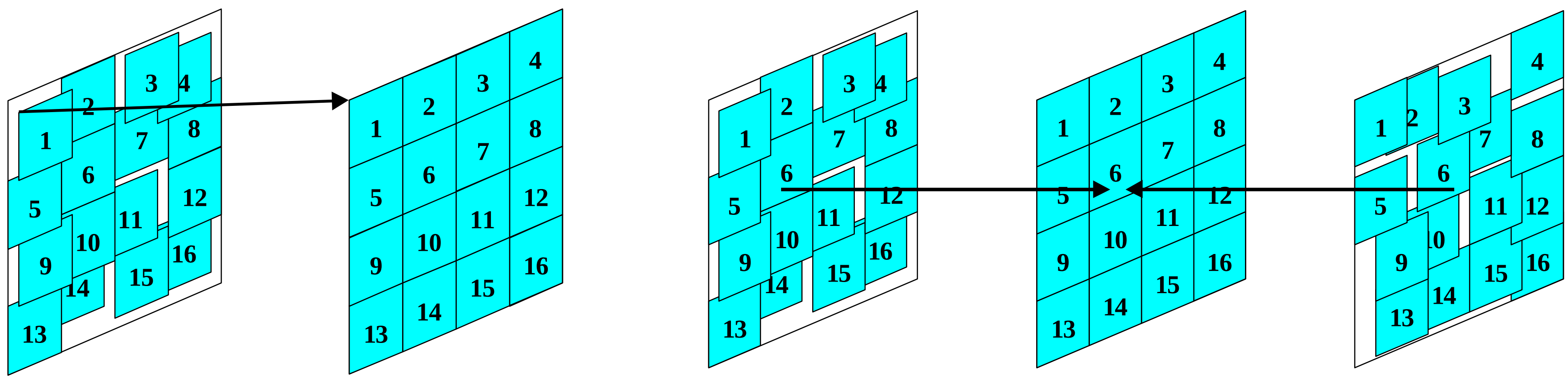
Prediction Error (Residual)



Video Coding

# MC-Prediction and Bi-Directional MC-Prediction (P- and B-frames)

- Motion compensated prediction: Predict the current frame based on reference frame(s) while compensating for the motion
- Examples of block-based motion-compensated prediction (P-frame) and bi-directional prediction (B-frame):

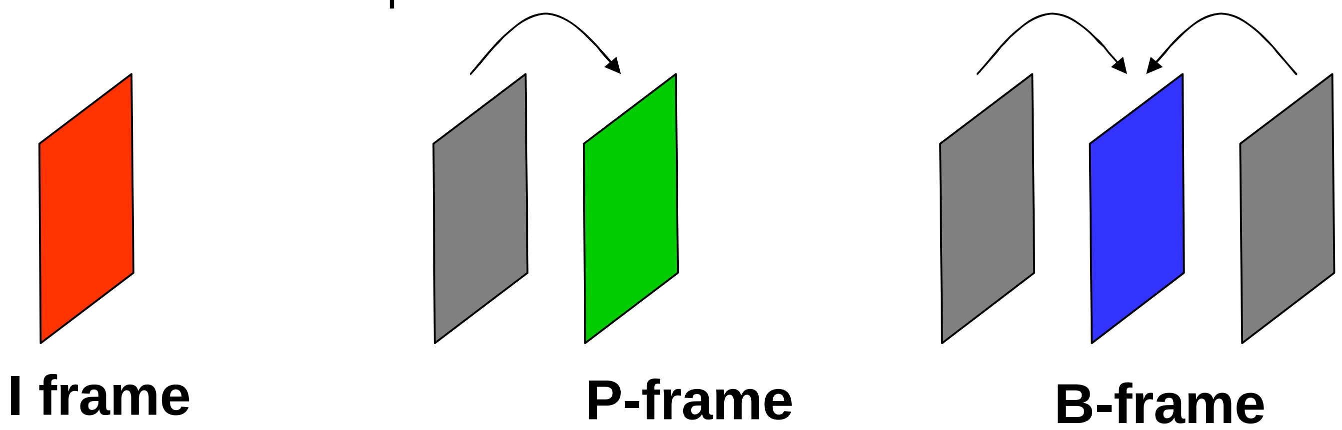


**Previous Frame** **P-Frame** **Previous Frame** **B-Frame** **Future Frame**

Video Coding

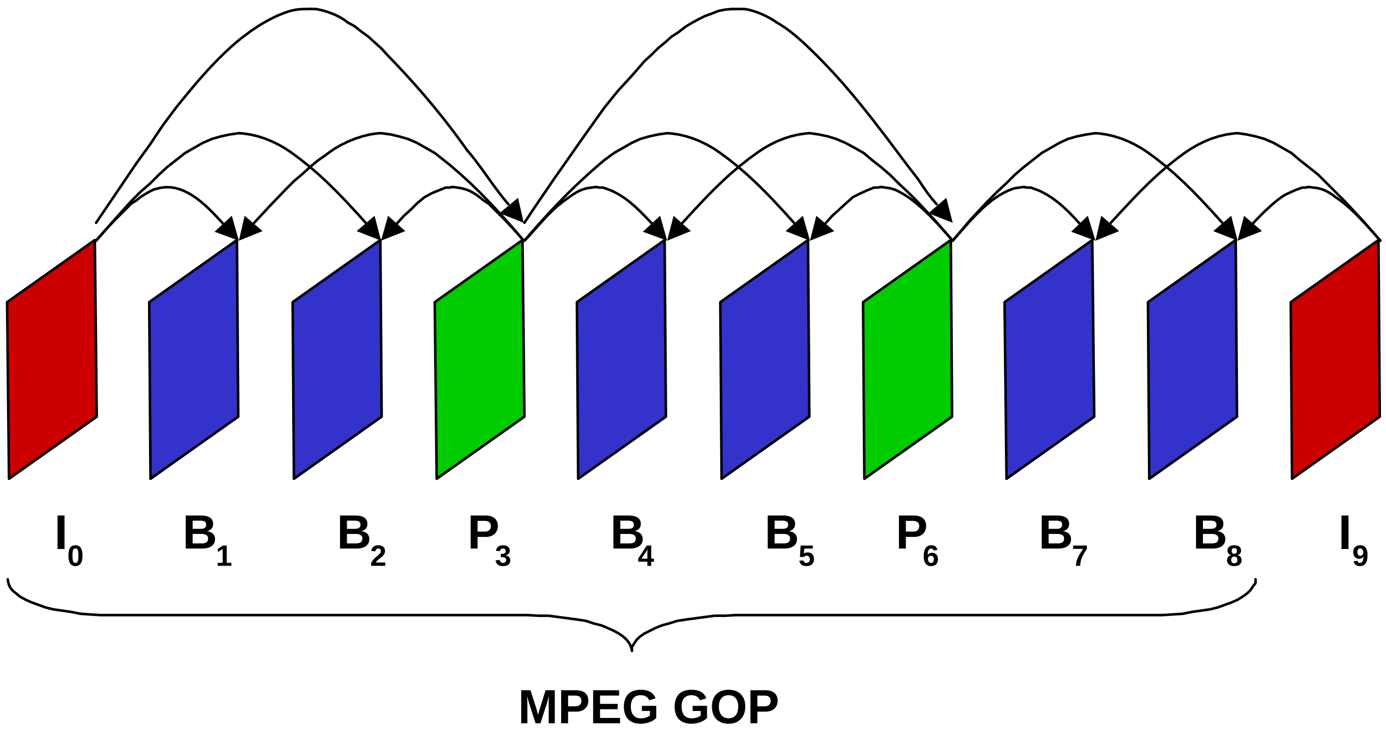
# Video Compression

- Main addition over image compression:
  - Exploit the temporal redundancy
- Predict current frame based on previously coded frames
- Three types of coded frames:
  - I-frame: Intra-coded frame, coded independently of all other frames
  - P-frame: Predictively coded frame, coded based on previously coded frame
  - B-frame: Bi-directionally predicted frame, coded based on both previous and future coded frames



# Example Use of I-,P-,B-frames: MPEG Group of Pictures (GOP)

- Arrows show prediction dependencies between frames



# Summary of Temporal Processing

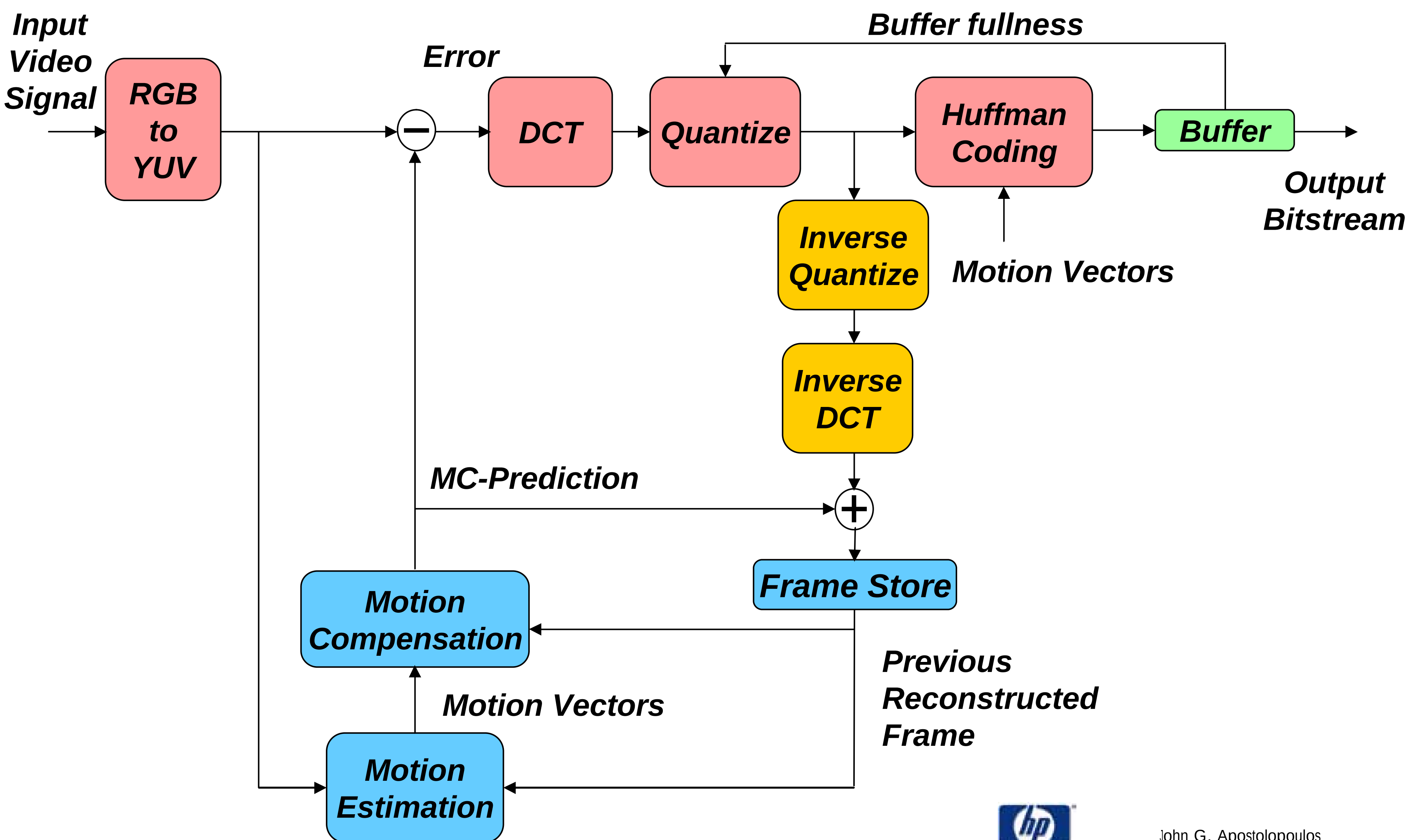
- Use MC-prediction (P & B frames) to reduce temporal redundancy
- MC-prediction yields:
  1) Motion vectors
  2) MC-prediction error → Code with conventional image coder
- Sometimes MC-prediction may perform badly
  - Examples: Complex motion, new imagery (occlusions)
  - Approach:
    1. Identify frame or individual blocks where prediction fails
    2. Code without prediction

Video Coding

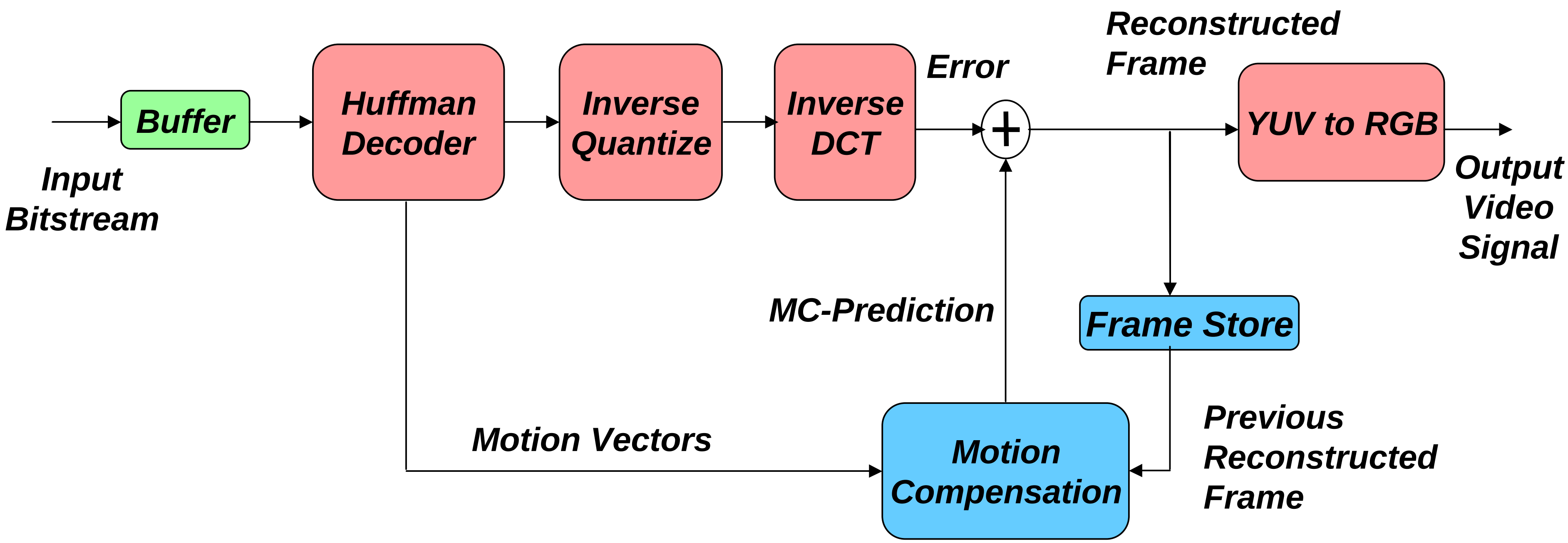
# Basic Video Compression Architecture

- Exploiting the redundancies:
  - Temporal: MC-prediction (P and B frames)
  - Spatial: Block DCT
  - Color: Color space conversion
- Scalar quantization of DCT coefficients
- Zigzag scanning, runlength and Huffman coding of the nonzero quantized DCT coefficients

# Example Video Encoder



# Example Video Decoder



Video Coding

# Outline

- Why do we need compression?
- Image compression
  - Color processing, DCT, etc
- Video compression
  - Exploit temporal dimension of video signal • Prediction
  - Generic (MPEG-type) video coder architecture
- → Current video compression standards
  - What are they?
  - What do the standards specify?
- Brief overview of H.264 / MPEG-4 AVC video coding standard

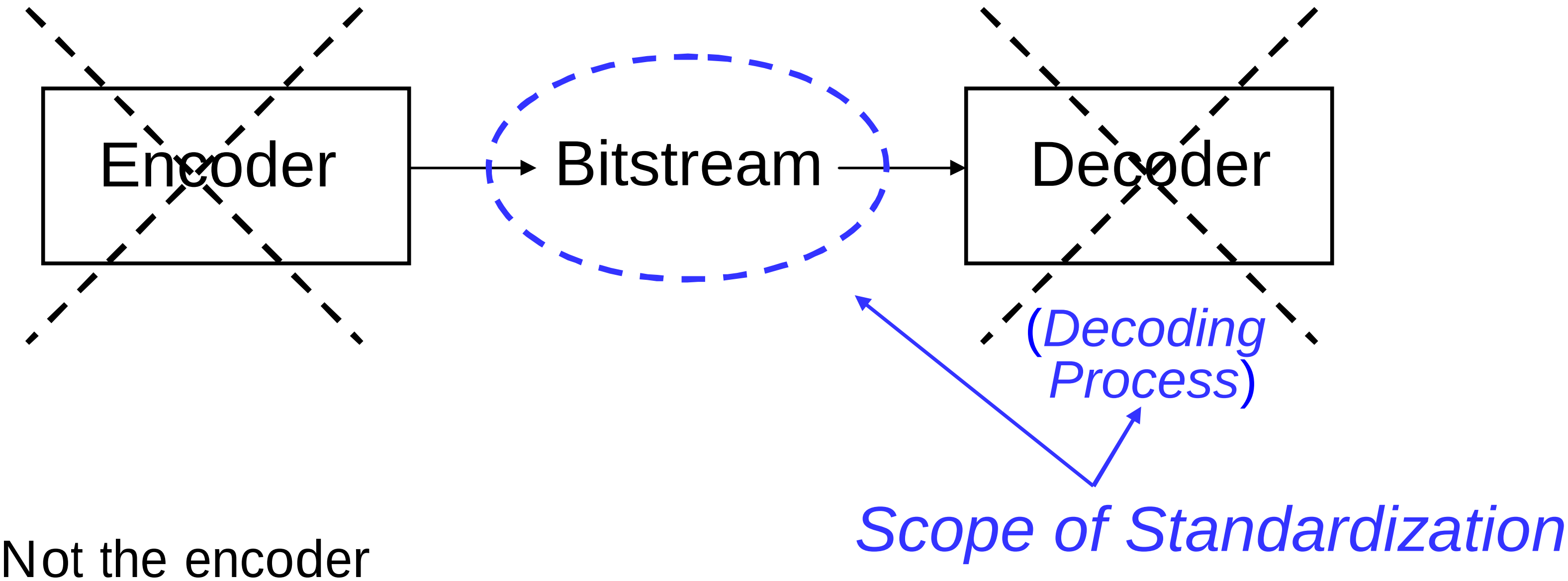
Video Coding

# Motivation for Standards

- Goal of standards:
  - Ensuring interoperability: Enabling communication between devices made by different manufacturers
  - Promoting a technology or industry
  - Reducing costs

# What do the Standards Specify?

# What do the Standards Specify?




- Not the encoder
- Not the decoder
- Just the bitstream syntax and the decoding process (e.g. use IDCT, but not how to implement the IDCT)

  → Enables improved encoding & decoding strategies to be employed in a standard-compatible manner

Video Coding

# Current Image and Video Compression Standards

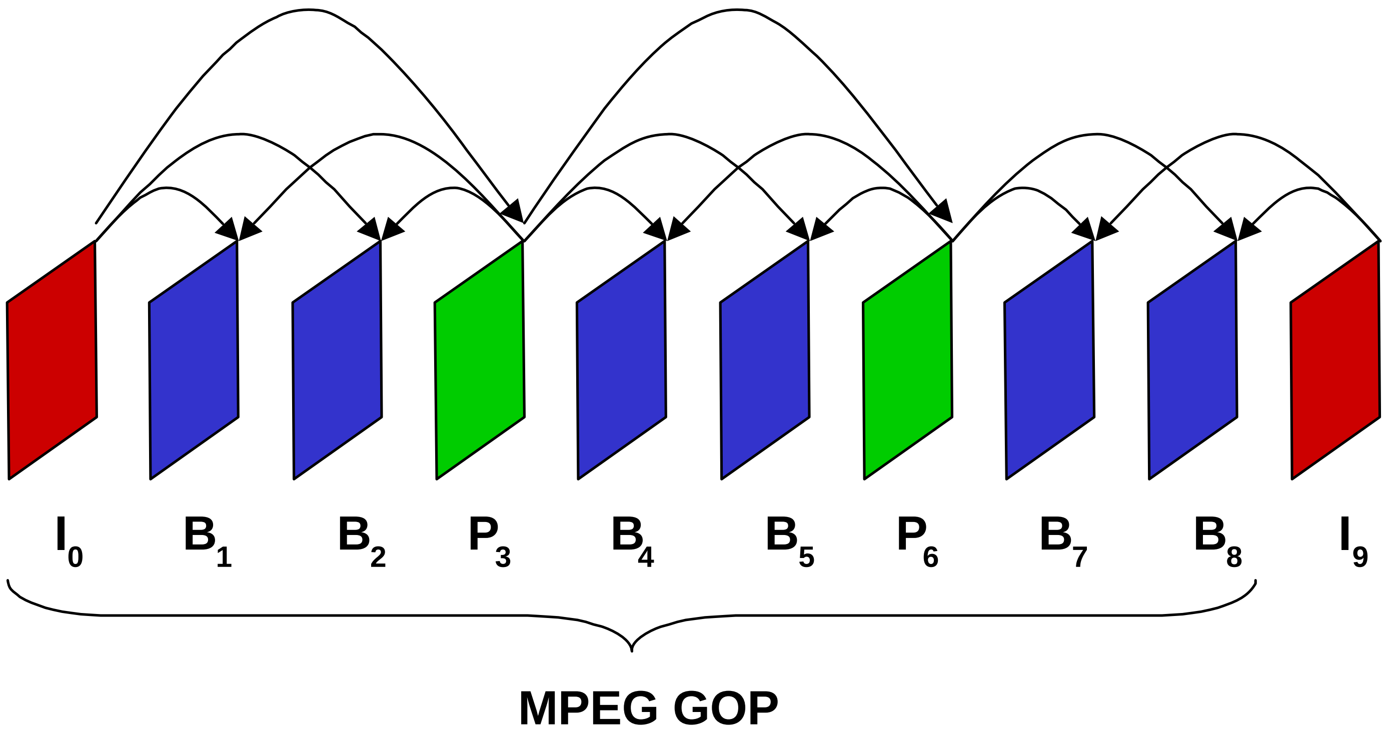
| Standard | Application | Bit Rate |
|---|---|---|
| JPEG | Continuous-tone still-image compression | Variable |
| H.261 | Video telephony and teleconferencing over ISDN | p x 64 kb/s |
| MPEG-1 | Video on digital storage media (CD-ROM) | 1.5 Mb/s |
| MPEG-2 | Digital Television | 2-20 Mb/s |
| H.263 | Video telephony over PSTN | 33.6-? kb/s |
| MPEG-4 | Object-based coding, synthetic content, interactivity | Variable |
| JPEG-2000 | Improved still image compression | Variable |
| H.264 / MPEG-4 AVC | Improved video compression | 10's kb/s to Mb/s |

ISO: JPEG & MPEG family of standards

ITU : H.26x family of standards

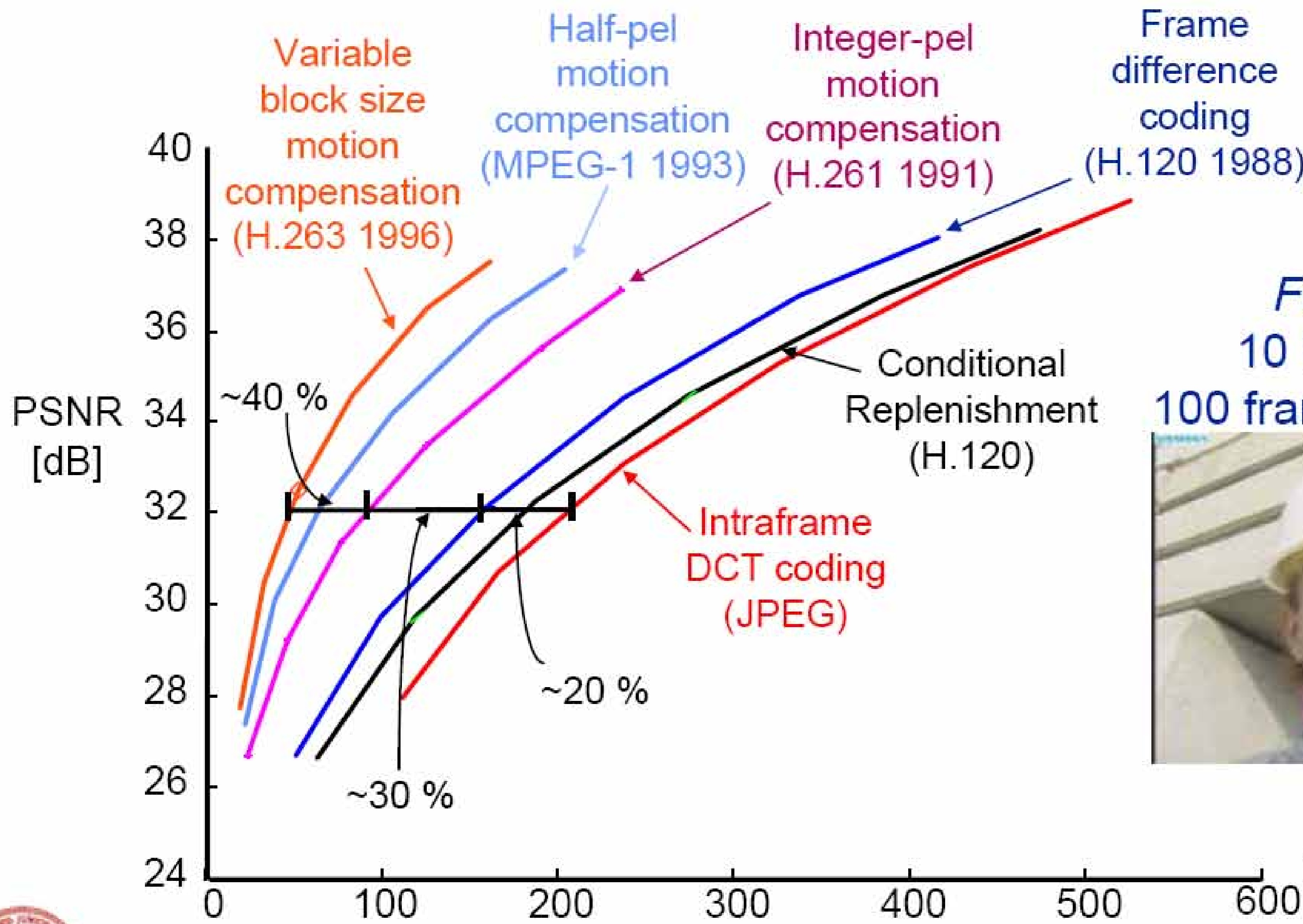
# MPEG Group of Pictures (GOP) Structure

- Composed of I, P, and B frames
- Arrows show prediction dependencies




- Periodic I-frames enable random access into the coded bitstream
- Parameters: (1) Spacing between I frames, (2) number of B frames between I and P frames

# Video compression progress




*Foreman*
10 Hz, QCIF
100 frames encoded

Video Coding

# Outline

- Why do we need compression?
- Image compression
  - Color processing, DCT, etc
- Video compression
  - Exploit temporal dimension of video signal • Prediction
  - Generic (MPEG-type) video coder architecture
- Current video compression standards
  - What are they?
  - What do the standards specify?
- Brief overview of H.264 / MPEG-4 AVC video coding standard

# H.264 / MPEG-4 Part-10 Advanced Video Coding (AVC)

Jointly designed by ISO and ITU standards bodies

- Known by different names: H.264, MPEG-4 Part-10 Advanced Video Coding (AVC), Joint Video Team (JVT), etc.

Goals:

- Significant improvement in compression
- Network friendliness from ground up:
  - Packet-based delivery
  - Carry over RTP/UDP/IP, MPEG-2 TS, H.324M, MPEG-4 file format, etc.
- Error & loss resilience
- Exact match decoding (e.g. integer transforms)

Additional recent goals:

- Professional applications, e.g. > 8 bits per sample, 4:4:4 color sampling, …

# H.264 / MPEG-4 AVC: Enhancements over Prior Standards

Improved temporal prediction:

- Adaptive block-size MC-prediction (16x16 to 4x4 pixels)
- Multiple reference frames stored for possible use for prediction
- ¼-pixel motion vector accuracy
- Deblocking filter within prediction loop

Improved spatial prediction:

- Directional spatial prediction of pixels
- Integer DCT-like transform (4x4 pixel blocks)

Improved entropy coding (probability prediction):

- Context-based adaptive binary arithmetic coder (CABAC)

Enhanced features:

- Improved error-resilience techniques
- SP & SI switching pictures

Video Coding

# H.264 / AVC Profiles & Levels

- Profile: Subset of tools applicable to specific application(s)
- H.264 version 1 defined 3 profiles:
  – Baseline (e.g., videoconferencing & wireless)
  – Main (e.g., broadcast, entertainment)
  – Extended (e.g., streaming) [Superset of Baseline]

| Coding Tools | Baseline | Main | Extended |
|---|---|---|---|
| I & P slices | X | X | X |
| B slices | | X | X |
| Interlaced | | X | X |
| CABAC | | X | |
| Error Resilience | X | | X |
| More Error Resilience | | | X |
| SP & SI slices | | | X |

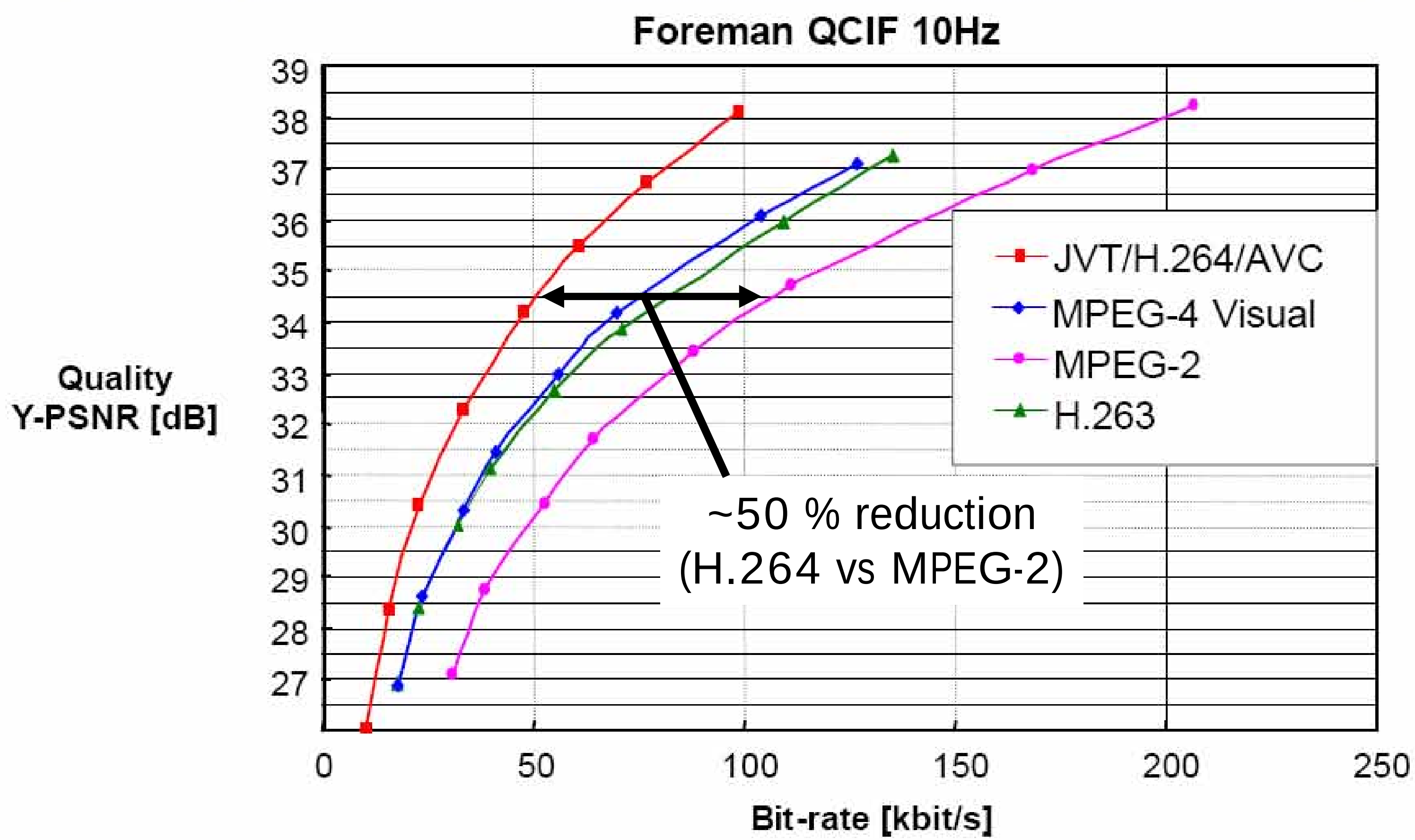
- Levels specify constraints on maximum computation and memory
  – Pixels per second, coded bit rate, # reference frames

# H.264 / MPEG-4 AVC Amendment 1: Fidelity Range Extensions (FRExt)

- Designed for professional applications which require higher quality in terms of bits/pixel, bit rates, spatial resolutions, etc.
- New "High" profiles:
  - High: Added 8x8 transform, improved quantization
  - High 10: Above plus support for 10 bits/pixel
  - High 4:2:2 : Above plus 4:2:2 format
  - High 4:4:4 : Above plus 4:4:4 format & 12 bits/pixel
- High profile likely for broadcast & entertainment quality apps
- High 4:2:2 for studio environments

## Example Comparison: Results depend strongly on specific sequence & coding tools employed!

# Comparison to MPEG-2, H.263, MPEG-4p2




H.264/AVC July '05 Gary Sullivan 5

Video Coding

# Summary

- Overview of basic principles & practice of image and video coding, and key standards
- Video coding is critical enabler for many applications:
  - Digital TV
  - DVD
  - Video streaming
  - Wireless video, e.g. over 3G cellular networks
- Future of video coding:
  - Continued improvements in compression
  - New or improved functionalities
    e.g. Scalable Video Coding (SVC) standard underway
  - Cross-layer design
    e.g. Coding and packetization or transmission

# Additional Information

- "Video Streaming: Concepts, Algorithms, and Systems", HPL Tech Report HPL-2002-260

  www.hpl.hp.com/techreports/2002/HPL-2002-260.html
- "Video Compression and Video Streaming", lectures at MIT 6.344, Spring 2004

  www.mit.edu/~6.344/Spring2004/Spring2004.html
- "Enterprise Streaming: Different Challenges from Internet Streaming", HPL Tech Report HPL-2005-98

  www.hpl.hp.com/techreports/2005/HPL-2005-98.html