VIDEO COMPRESSION

TOPICS TO BE COVERED

Introduction
- **Video Characteristics and Types**
- **Applications**

Why is Video Compression different from Image Compression?

Modalities of Video Coding – Intraframe and Interframe

Motion Vectors and Motion Compensation Techniques

Standards and Implementation
- **H.261 (ITU)**
- **H.263 (ITU)**
- **MPEG-1 (ISO)**
- **MPEG-2 (ISO)**
- **H.264 / MPEG-4 AVC**
- **HEVC**

INTRODUCTION

Digital Video in comparison to Analog Video

Advantages of Digital Video

- **Higher levels of quality**
- **Easily manipulated**
- **Easily stored and copied or duplicated**
- **Easily transmitted over networks**
- **Easy integration with other digital media**

Digital Video is characterized by

- **Frame rate (creates illusion of motion)**
- **Frame dimension (width and height)**
- **Pixel Depth (bits per pixel)**

APPLICATIONS OF DIGITAL VIDEO

Application	*Frame Rate*	*Dimensions*	*Pixel Depth*
Multimedia	**15**	**320x240**	**16**
Entertainment (TV)	**25**	**640x480**	**16**
Industry Applications	**5**	**640x480**	**8-12**
Video Telephony	**10**	**320x240**	**8-12**
HDTV	**25**	**1920x1080**	**24**

WHY IS VIDEO COMPRESSION DIFFERENT FROM IMAGE COMPRESSION?

Video consists of a stream of images – but is also characterized by -

- **Scanning format (progressive, interlaced)**
- **Frame size (typically standardized)**
- **Frame rate (15-30 Hz)**

Like images Spatial Redundancy exists in each frame

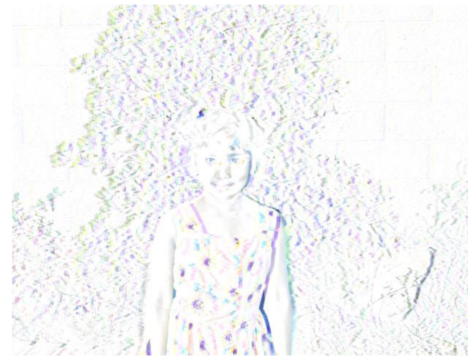
Temporal Redundancy in between frames and can be can be exploited for compression reasons!

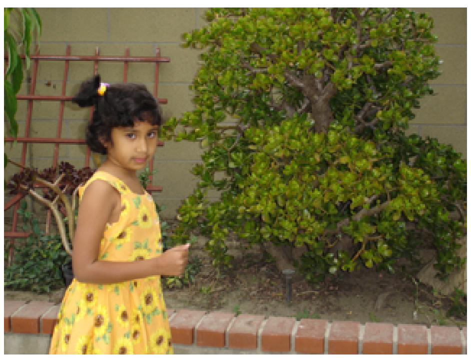
- **Areas of the image in the sequence may remain constant, or**
- **Areas tend to move in a *predictable fashion*, and therefore can be "*predicted*" from frame to frame**

EXPLOITING TEMPORAL REDUNDANCY


Frame *n*


Frame *n+1*


Frame *(n+1)* – Frame *n*


Frame *n*


Frame *n+1*


Frame *(n+1)* – Frame *n*

Successive frames in a video. (low motion on top, high motion at the bottom). Also shown is the frame difference of the Y channel. The difference large in high motion compared to low motion.

EXPLOITING TEMPORAL REDUNDANCY – MOTION COMPENSATION

This prediction is called *Motion Compensation* and is of two types –

- **Local Motion Compensation (or just motion compensation)**
- **Global Motion Compensation**

Each video frame may be encoded differently depending on whether to use spatial or temporal redundancies

MODALITIES OF VIDEO COMPRESSION

Depending on which mode (spatial or temporal) you want to exploit for compression reasons, you have two modes of compression for video

Intraframe

- **Each frame is encoded as an individual entity (like an "image")**
- **Uses Image Compression techniques (eg DCT)**

Interframe

- **Predictive Encoding but for the temporal domain**
- **Instead of encoding the current frame directly, we encode the *difference* between the current frame and a *prediction* based on previous frames**
- **Term Used – *Motion Compensation***

INTRAFRAME CODING

These frames are compressed using a combination of a Lossy Scheme such as Transform Coding or Subsampling/Quantization and Lossless Entropy Coding such Huffman or Arithmetic.

For example, the MPEG/ITU Standard compresses these frames as discussed in the previous lecture –

- **Get 8x8 blocks (for each component)**
- **DCT on each block**
- **Quantization of all coefficients**
- **AC zigzag ordering**
- **DC $\Rightarrow$ DPCM $\Rightarrow$ (size) (value)**
- **AC $\Rightarrow$ Runlength Encoded $\Rightarrow$ (runlength, size) (value)**
- **Both AC and DC Coefficients get Huffman encoded to form a bit stream**

CHANGES FROM FRAME TO FRAME

What can happen to a pixel (or pixel region) from one frame to another?

- **Nothing! – Like an unchanging background – so do we need to encode information here?**
- **Changes (slight) due to quantization and noise**
- **Changes due to the motion of the object**
- **Changes due to motion of the camera**
- **Changes due to environment and lighting**

If nothing has changed – no need to encode

If changes in pixel color or pixel value due to motion of object or camera, maybe we can predict how the pixels have moved, thereby needing to encode only the change vector

MOTION VECTOR

Lets assume that if a pixel has moved from frame to frame, the color of the pixel has not changed

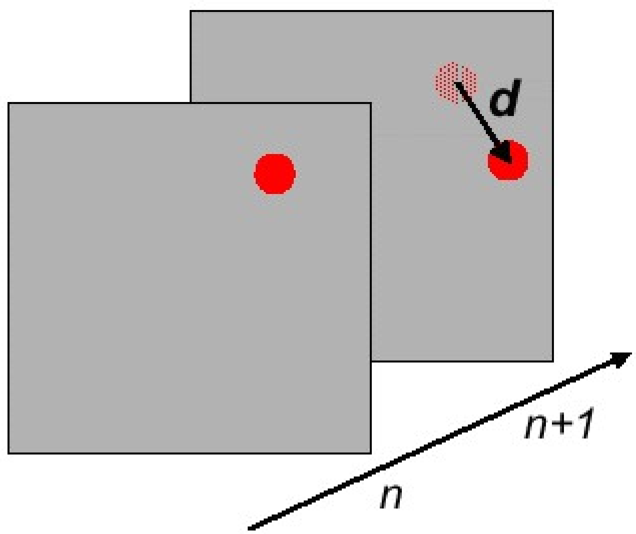
In other words: A point *(x,y)* in frame *n+1* with color $c_{n+1}(x,y)$ corresponds to some point *(x',y')* in frame *n*.

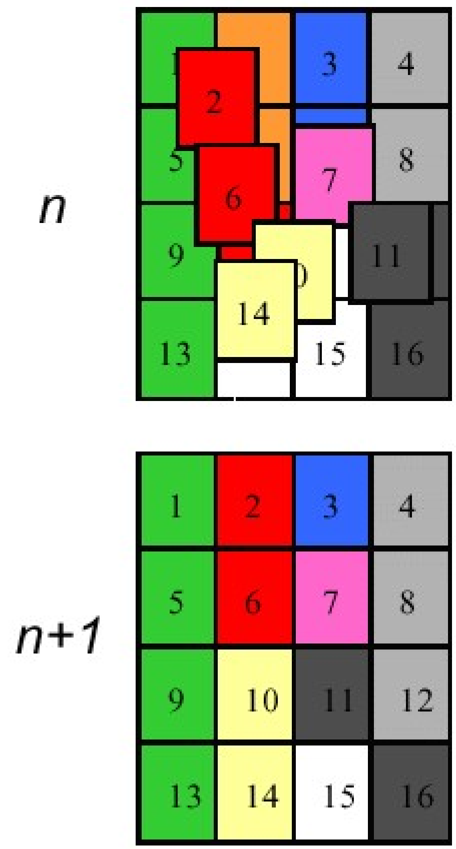
$$c_{n+1}(x, y) = c_n(x', y')$$

If we call this *displacement* or *motion vector* $d = (d_x, d_y)$ then we have

$$d = (d_x, d_y) = (x, y) - (x', y')$$

MOTION VETOR EXAMPLE





MOTION COMPENSATION

If our *motion assumption* is (approximately) valid, then $c_{n+1}(x,y)$ can be predicted from $c_n(x-d_x, y-d_y)$

Like with *differential encoding*, we encode and transmit the residual error

$$e(x,y) = c_{n+1}(x,y) - c_n(x-d_x, y-d_y)$$

We must also encode and transmit the motion vectors (d_x, d_y)

If we use *lossy encoding*, we should use a *closed-loop* scheme!

MOTION COMPENSATION - MACROBLOCKS

Do we need to *compute* and *transmit* one motion vector *d* **per pixel?**

- **Computationally intensive!**
- **Lots of data to send**

Instead: transmit only 1 motion vector per groups of pixels called *macroblock* (e.g., 16x16 pixels)

Advantage & Disadvantages:

Fewer motion vectors to be transmitted
Faster computationally
Less precise motion prediction if motion is not constant within the macroblock (e.g., if macroblock covers the edge of a moving object)

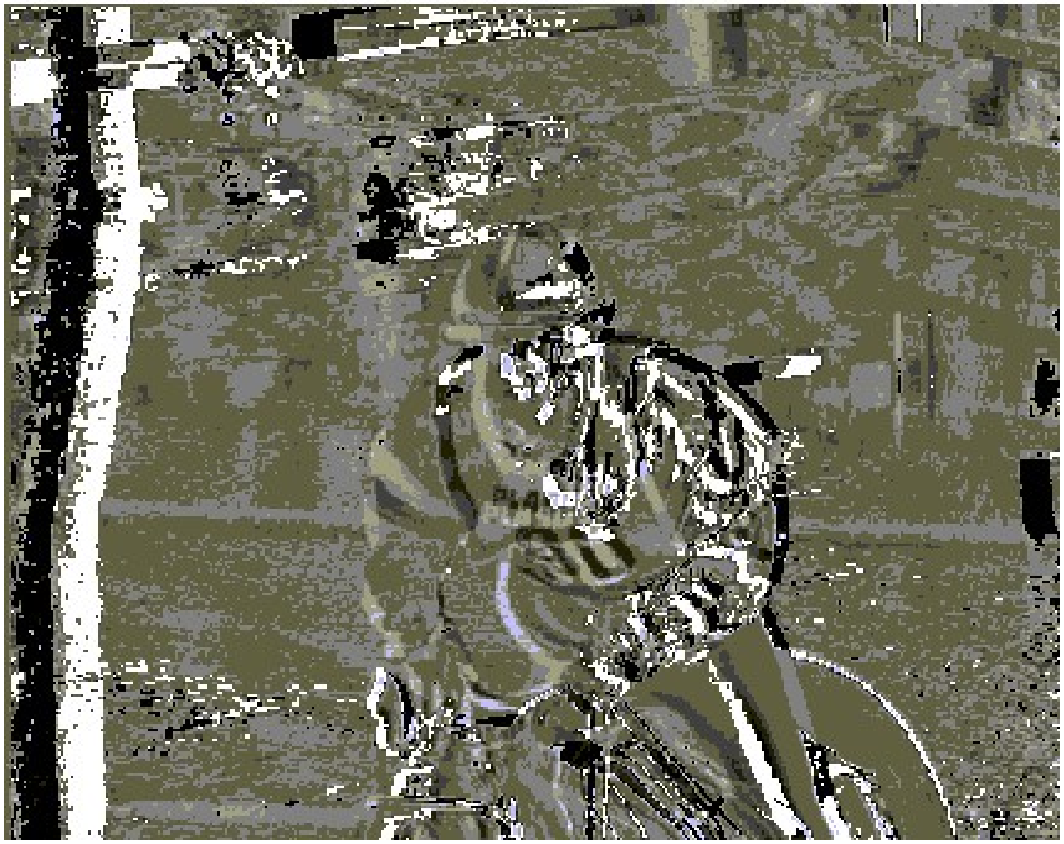
MACROBLOCK MOTION VECTORS – EXAMPLE

Frame at time *n*

MACROBLOCK MOTION VECTORS – EXAMPLE

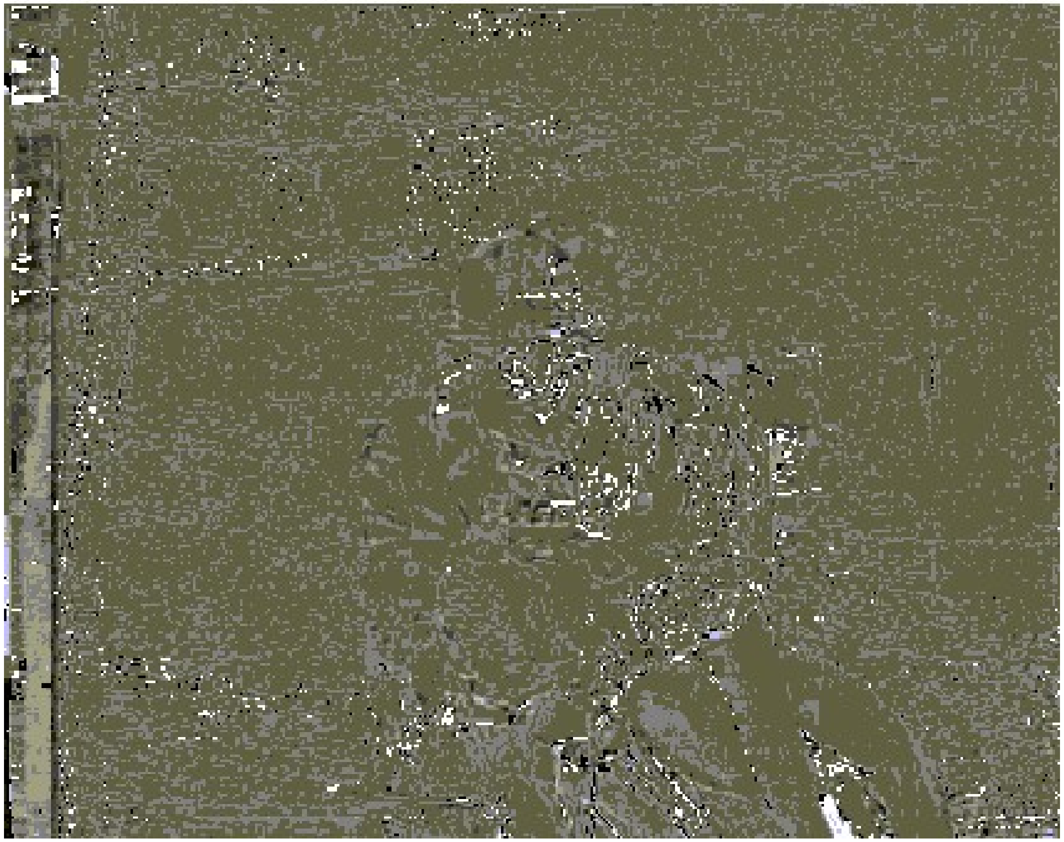
Frame at time n-1 showing motion vectors computed using frame at time n

MACROBLOCK MOTION VECTORS – EXAMPLE



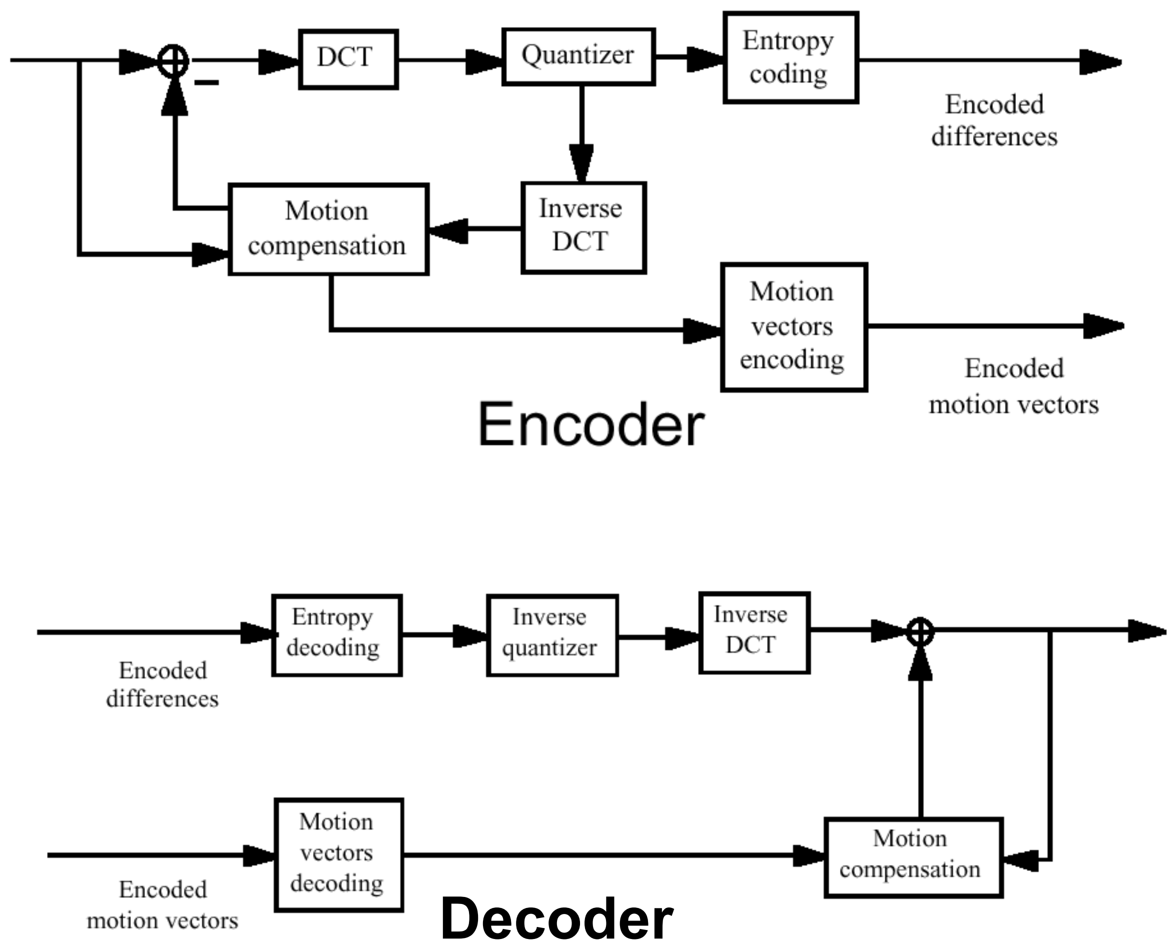
Difference between frames *n* and *n-1*, without motion compensation

MACROBLOCK MOTION VECTORS – EXAMPLE



Difference between frames *n* and *n-1*, *with motion compensation*

CLOSED LOOP MOTION COMPENSATION



A NOTE ON FRAME SEGMENTATION

This also refers to deciding a *size* and *shape* of these macroblocks that are used in computing motion

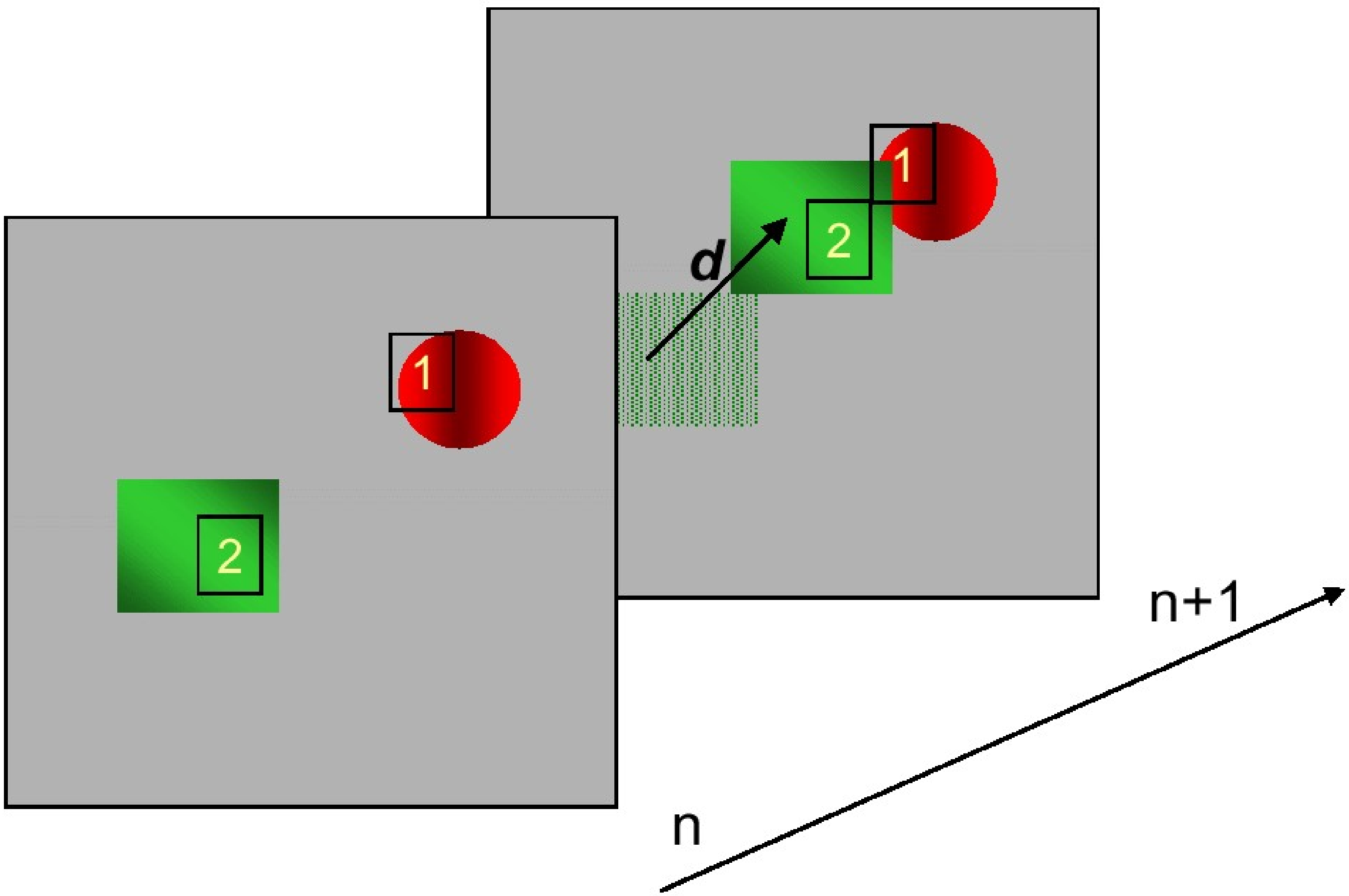
Normally a frame is divided in non-overlapping blocks of a certain size *B* (16x16) and shape (square). Block size and shape affects the performance of compression

Issues

- **Large *B* $\Rightarrow$ fewer blocks to search $\Rightarrow$ fewer motion vectors to encode**
- **For large *B*, the movement of objects do not coincide with boundaries of *B* $\Rightarrow$ larger errors or residuals that need to be encoded**

Thus block size represents a trade off between minimizing the number of motion vectors and maximizing the quality of the matching blocks

MACROBLOCKS AT MOTION BOUNDARIES



SEARCHING AND BLOCK MATCHING

If the difference between the target block and the candidate block at the same position in the past frame is below some threshold $\Rightarrow$ no motion, else search

Block matching is the most time consuming – for accurate results you need to do an exhaustive search which is - given a block *B* in the current frame search for a match block *A* in the entire previous frame. This may be further optimized: Instead of entire frame,

- **Limit search to a region**
- **Logarithmic Search**
- **Hierarchical**

In practice, this search can be limited to a range around the "target block"

MATCHING CRITERIA

Various Criteria are used to decided whether the current block matches a target block –

- **Mean Absolute Difference (MAD)**
- **Mean Square Difference (MSD)**
- **Pel Difference Classification (PDC)**
- **Integral Projection**

(Formulae of each to be mentioned in the lecture …)

MOTION COMPENSATION – ALGORITHM

Given a sequence of frames (each have macroblocks) –

Encode first frame as IntraFrame

For each corresponding macroblock of next frame and current frame, find the difference.

- **If difference less than threshold $\Rightarrow$ no motion, find residual error**
- **If difference above threshold $\Rightarrow$ may be motion, look in search range to find a matching block using matching criteria discussed above. Note motion vector and residual error**

If difference (or total residual error) is *too large* for a majority of macroblocks, and/or after regular intervals encode the current frame as an Intraframe and proceed to previous step

MOTION COMPENSATION – ENCODING

There are two things to encode here:

- **Motion Vector for every macroblock**
- **Difference or residual for every macroblock**

Motion vectors are typically encoded losslessly (similar to JPEG lossless mode)

The *residuals e(x, y)* are encoded lossy + lossless (DCT + Entropy) producing *variable bit rate (VBR).*

If smooth motion or no motion:

- **Motion prediction is good (residuals are small)**
- **Entropy coded with few bits**

If complex motion or change of scene:

- **Motion prediction is bad (residual are large)**
- **Entropy coded with many bits**

MAIN PROPOSED DIGITAL VIDEO STANDARDS

ITU standards

- **H.261 (videoconferences over ISDN)**
- **H.263 (videoconferencing and video telephony over POTS)**
- **H.264**

ISO standards

- **MPEG-1 (movies on CD-ROM)**
- **MPEG-2 (digital television, movies on DVD)**
- **MPEG-4 (more versatile distribution)**
- **MPEG4 AVC (Advanced Video Coding)**
- **HEVC**

ITU H.261

The ITU H.261 standard was initially designed for ISDN and was intended to support video conferencing applications, which have relatively small amounts of motion (mainly head and shoulder movements).

It and supports the following features

- **Produces bit-rates of *k* x 64 Kb/s**
- **Only non-interlaced video**
- **Only CIF and QCIF formats**
- **Can encode in *intraframe* and *interframe* mode**

***Intraframe*: DCT on 8x8 blocks (like JPEG)**

***Interframe*: computes motion vectors on 16x16 macroblocks from a *reference frame,* which may be a frame encoded in intraframe or interframe mode.**

ITU H.263

Supports a wider range of picture formats, including 4CIF (704x576) and 16CIF (1408x1152).

H.263 is part of the H.324 standard for communication over POTS with a modem with a maximum available rate of 33.6 Kb/s and a normal available bit-rate of 26-28 Kb/s

The other components of H.324 are: G.723 speech codec standard, framing and control protocols (H.223 and H.245) and data-sharing protocols

Based on the same DCT and motion compensation technique used in H.261. Incremental improvements are

- Half-pixel motion compensation
- Advanced motion prediction mode, including
- Overlapped block motion estimation
- Bi-directional motion estimation

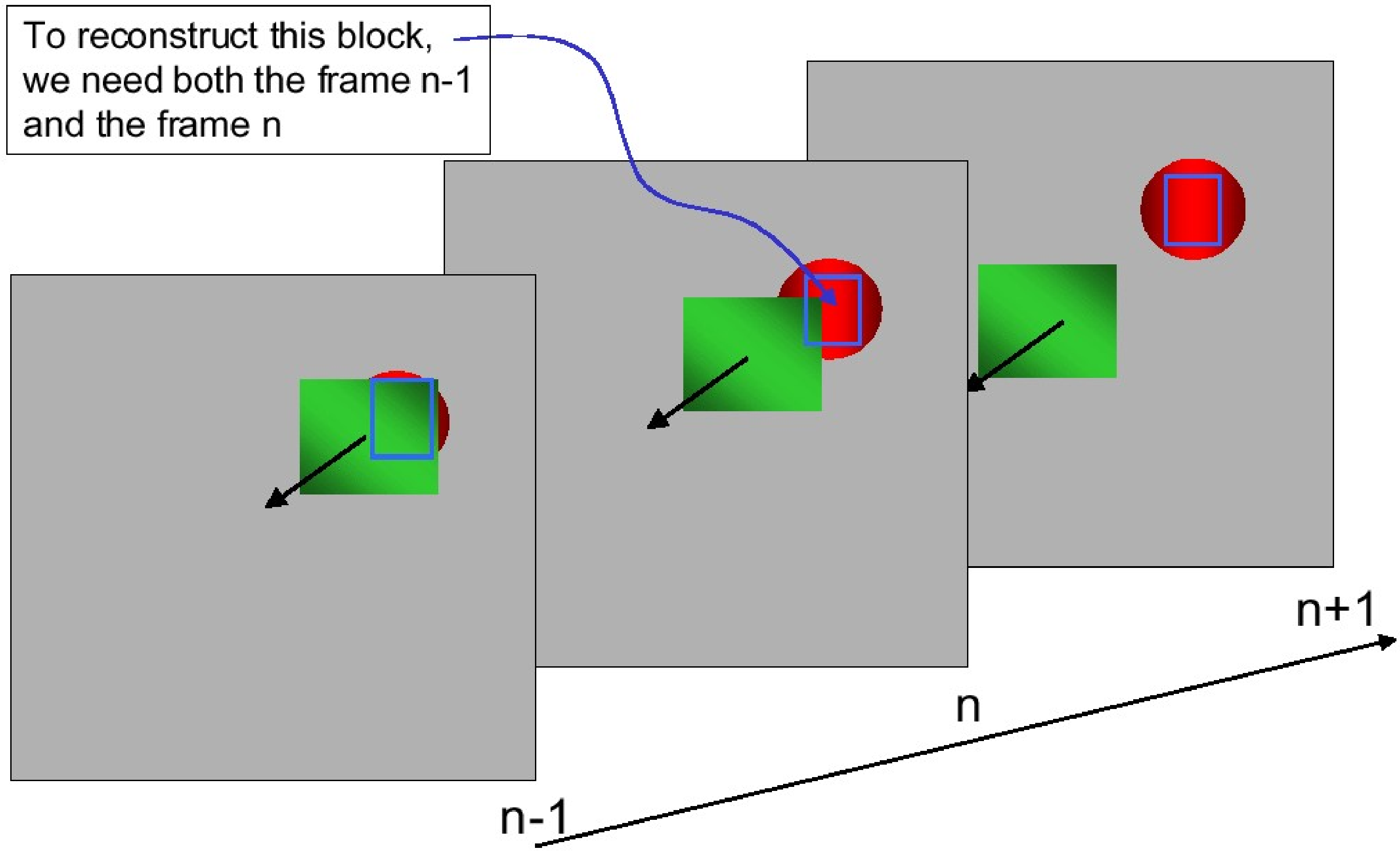
BIDIRECTIONAL MOTION COMPENSATION

A block of a bidirectionally coded frame is predicted from both a previous frame and a later frame

Bidirectional Motion Compensation implies:

- **A *delay* (to decode the current frame, we need to have received and decoded a later frame)**
- ***Different order* between the sequence of acquired and displayed frames and the sequence of encoded frames**

BIDIRECTIONAL MOTION EXAMPLE



MPEG-1 VIDEO

MPEG-1: true multimedia standard with specifications for coding and transmission of audio, video and data streams in a series of synchronized, mixed packets

Driving focus: storage of multimedia content on CD-ROMs (1.4 Mb/s, 600 MB)

Picture format: SIF

- **non-interlaced: 352x288, 25 f/s**
- **interlaced: 354x240, 30 non-interlaced f/s**

Quality: VHS VCR-like video and audio

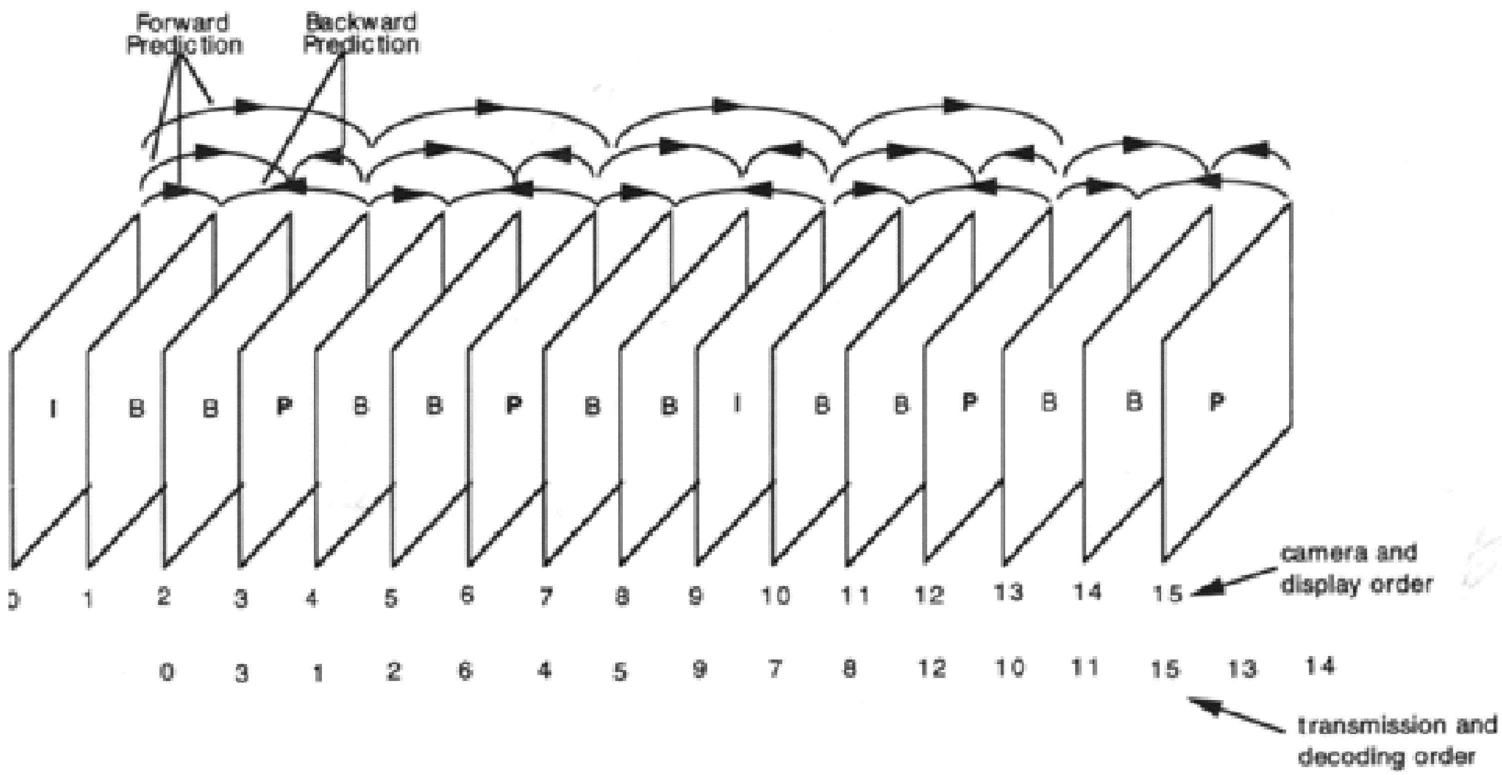
MPEG-1 VIDEO (2)

Coding mechanism similar to H.26x

Three types of frames:

- ***I-frames*** **(coded in intraframe mode)**
- ***P-frames*** **(coded with motion compensation using as reference a previous I or P frame)**
- ***B-frames*** **(coded with bidirectional motion compensation based on a previous and later I or P frame)**

ACQUISITION/TRANSMISSION ORDER



MPEG-2

MPEG-2 was designed to provide the capability for compressing, coding, and transmitting high-quality, multichannel, multimedia signals over broadband networks

MPEG-2 standard specifies the requirements for video coding, audio coding, systems coding for combining coded audio and video with user defined private data streams, as well as conformance testing

MPEG-2 video was originally designed for high-quality encoding of interlaced video from standard TV (4-9 Mb/s). Over time, the MPEG-2 video standard was expanded to include high-resolution video (such as HDTV), as well as *hierarchical* (or *scalable* or *layered*) video coding

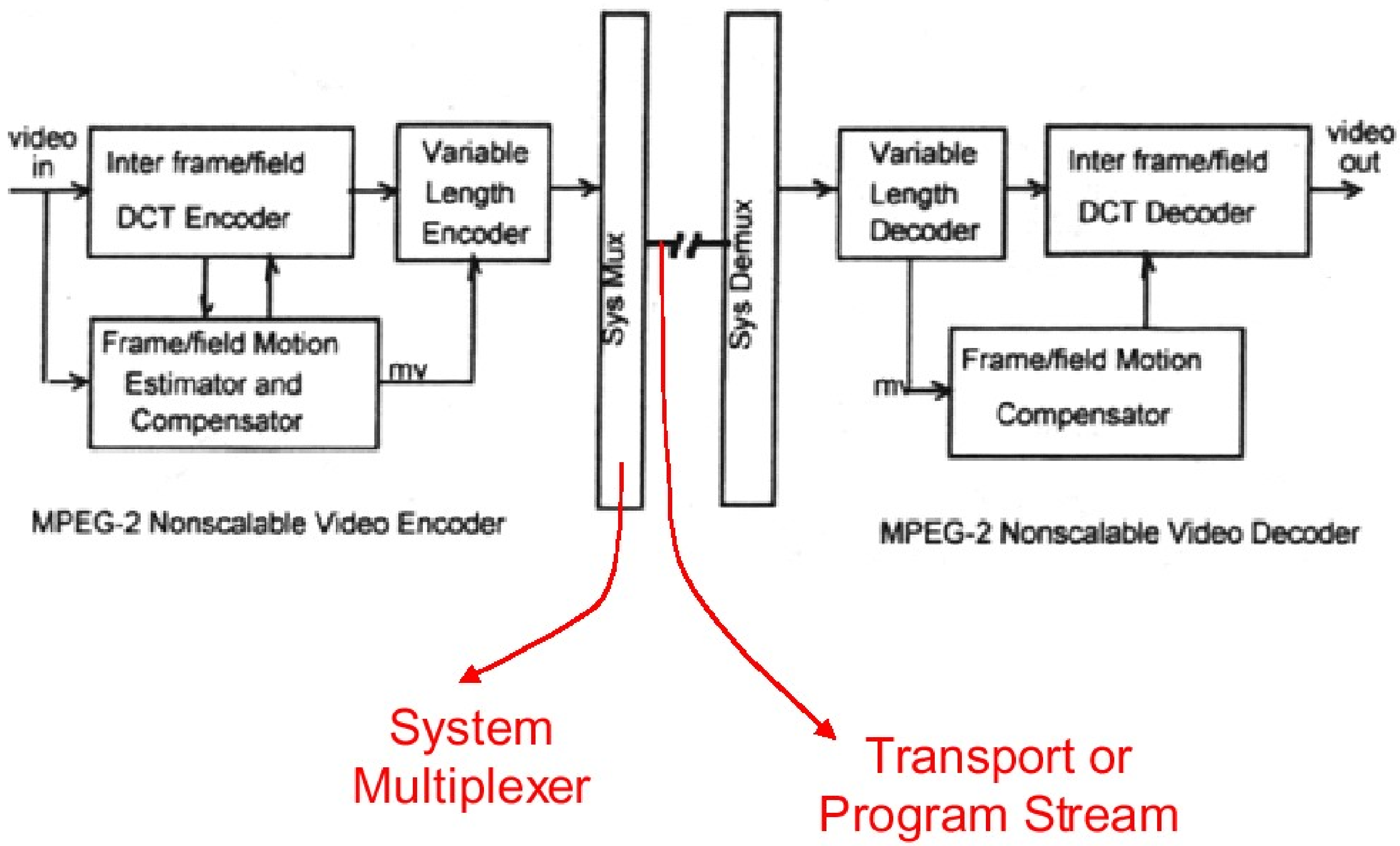
MPEG-2 SYSTEMS LEVEL

Because MPEG-2 was designed as a *transmission standard*, it supports a variety of packet formats and provides error-correction capability for noisy channels

Two kinds of streams:

- ***Program stream*: uses long and variable-length *packets*. Well suited for software-based processing and error-free environments**
- ***Transport stream*: uses fixed-length packets (188 bytes). Well-suited for delivering compressed video and audio over error-prone channels such as CATV networks and satellite transponders. Allows one to include multiple programs in a single stream**

NON-SCALABLE MPEG-2 VIDEO ENCODER



SCALABLE MPEG-2 VIDEO

The coded representation (bit-stream) is generated in such a way that decoders of various complexities are able to decode video of different resolution/quality from the same bit stream

> **If the bitstream is truly scalable, decoders of different complexities can coexist: *inexpensive decoders* would be expected to decode only small portions of the same bitstream producing basic quality pictures, while *more sophisticated and expensive decoders* will produce higher quality pictures**

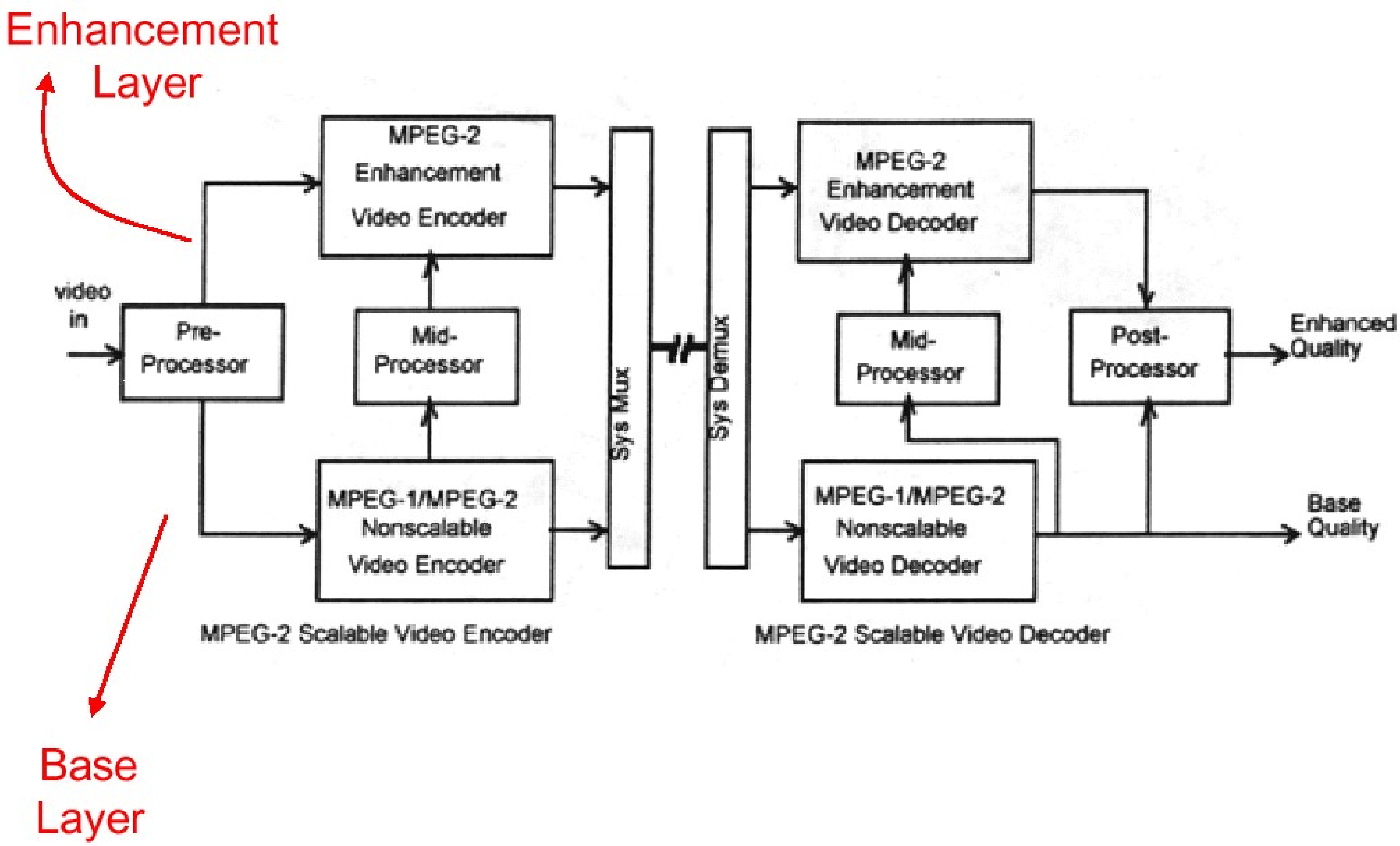
Finds applications in networks with multi-quality video services and windowed video on computer workstations

SCALABILITY LAYERS IN MPEG-2 VIDEO

Input video goes through a pre-processor producing a *base layer* signal and an *enhancement layer*

- ***Base layer*** **is decoded independently by a standard MPEG-2 non-scalable video encoder**
- ***Enhancement layer*** **is encoded *with respect to the base layer* by a MPEG-2 *enhancement encoder***

SCALABLE MPEG-2 VIDEO ENCODER



H.264 OR MPEG-4 AVC

Latest Video Coding standard issued by ITU-VCEG and ISO-MPEG

Better video quality than earlier codec standards at same or less bitrates.

Designed for technical solutions addressing

- Broadcast over cable, satellite, Cable Modem. DSL
- High quality interactive storage on optical/magnetic devices – DVD
- Video on Demand over DSL, Cable Modem, wireless
- MMS over Ethernet, LAN, Wireless

Industry excitement over applications and deployments using H.264 / AVC

FEATURES OF H.264 / AVC

Features that make this codec better

- Directional spatial prediction for Intra Coding
- Small block size transform
- Variable block size motion compensation
- QPEL – quarter pixel accurate prediction
- Multiple reference picture motion compensation for P frames
- Bi prediction for B frames (not necessarily one future and one past)


MPEG4-2,
at a bit rate of 1.2Mbps
Resolution (544 x 368)
MPEG4-10,
at a bit rate of 1.2Mbps
Resolution (544 x 368)

H.264

HEVC

- **Macroblocks Vs CTUs (Coding Tree Units)**
- **Broadcast TV industry, MSFT, AAPL**
- **Google and VP9**
- **Alliance for OpenMedia and AV1 format**