The H.264/AVC video co-decoding standard

Outline

- ✓ Introduction to data compression problem and video coding systems
- ✓ Brief history of video coding standards
- ✓ Technical overview of H.264/AVC
- ✓ H.264: feature highlights
- ✓ H.264: robustness and error resilience tools
- ✓ Conclusion

INTRODUCTION TO VIDEO COMPRESSION

The need for compression - Image -

- Digital colour image: 352x288 pixel
- RGB representation: 24 bpp (8bits for red,green,blue)
- Total amount of bytes: > 300K
- JPEG: common image compression standard, < 20K, similar quality

iginal Image
ze: 300k

Compressed Image
Size: 20k
PSNR: > 30dB

The need for compression - Video -

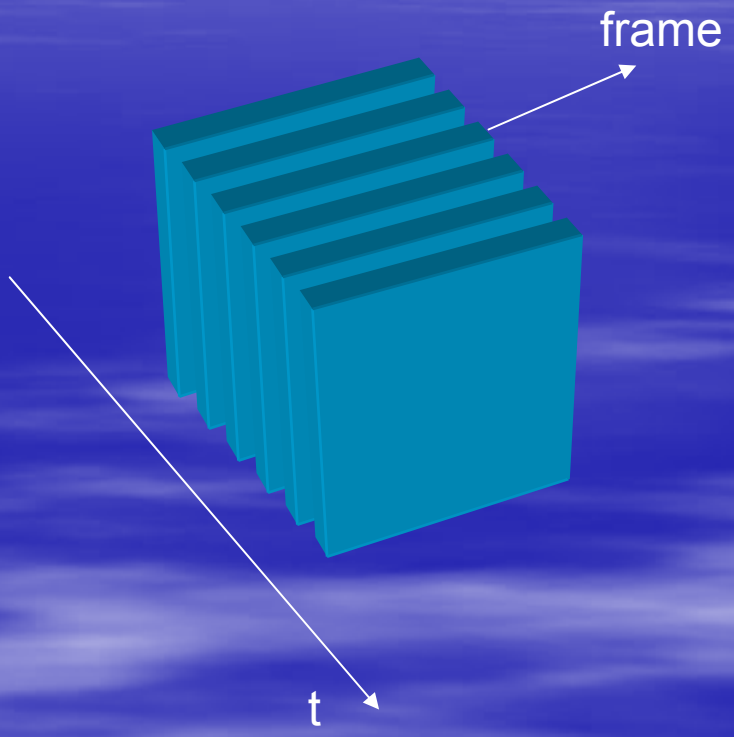
- Video signal:sequence of frames (images) related among temporal dimension
- TV video quality: 704x576 pixels per frame, 12 bpp, 25 frames per second

  > 121 Mbps
- Too much data for video transmission or storage
- Increasing importance of multimedia communication:

NEED FOR COMPRESSION &

GOOD PERCEPTUAL QUALITY

NEED FOR VIDEO CODING STANDARD



Entropy coding - I

- Map symbols to bit: assign short code to more frequently occurring symbols
- Lossless compression
- *Huffman*: optimal for discrete sources with known statistics.One code for each symbol.
- *Exp-Golomb*: belong to the class of Huffman codes (variable length codes with a regular construction)
- *Run-Lenght*: reduces the length of a repeating character sequence. A long sequence of the same character is replaced by two symbols (character and counter)

Entropy coding - II

- *Arithmetic*: assign to each sybol an interval in the rage (0,1) with amplitude proportional to his cumulative probability. A unique code for the whole sequence of symbols is generated. Achieve better performance (< 1 bit/symbol). More sensible to bit error
- *Context based encoders*: a *context model* is a probability model assigned to each symbol to code. May be chosen from a selection of available models depending on the statistics of the recently coded symbols

Video color space - I

- Human visual system perceives scene content in term of brightness and color information
- We are more sensitive to the detail of brightness than colour
- H264/AVC (as previous standards) use $Y\,C_b\,C_r$ color space
- Y called luminance (*luma*) represents brightness (black and white signal)
- $C_b\,C_r$ called chrominance (*chroma*) components represent color difference signals
- Luminance-chrominance formation from RGB components:

$$Y=0.299R+0.587G+0.11B$$

$$C_b = B - Y$$

$$C_r = R - Y$$

Video color space - II

- Luminance-chrominance formation from RGB components:

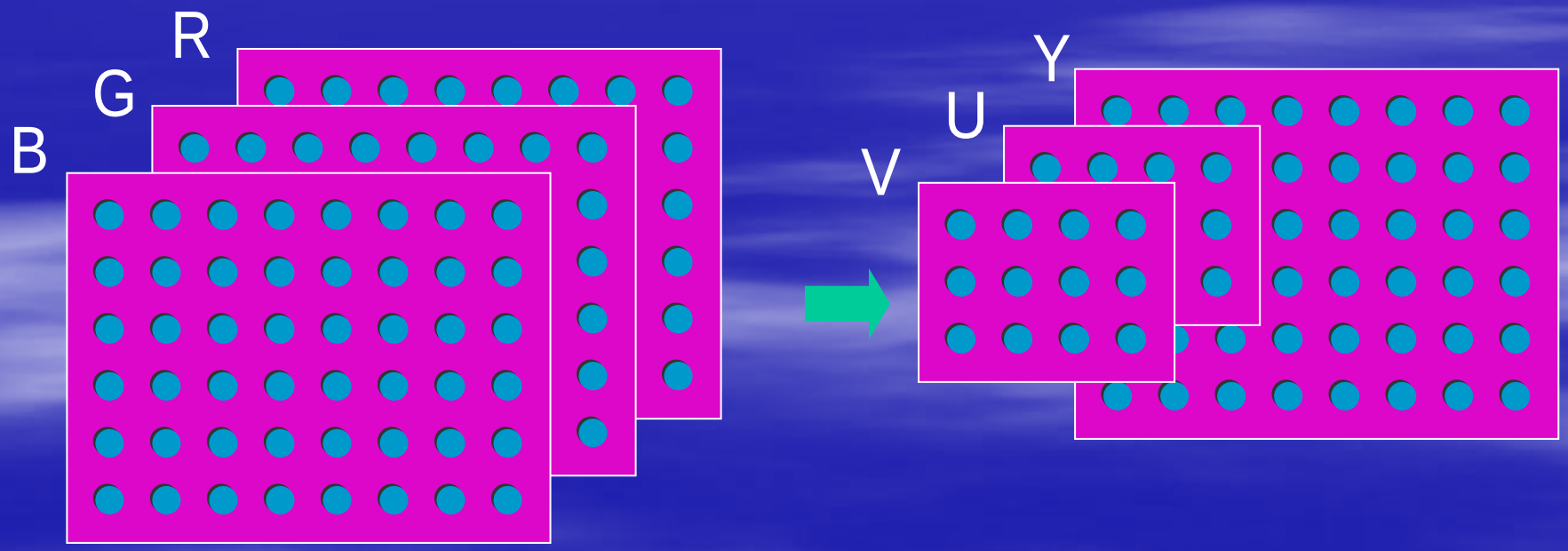
$$Y = 0.299R + 0.587G + 0.11B$$

$$C_b = B - Y$$

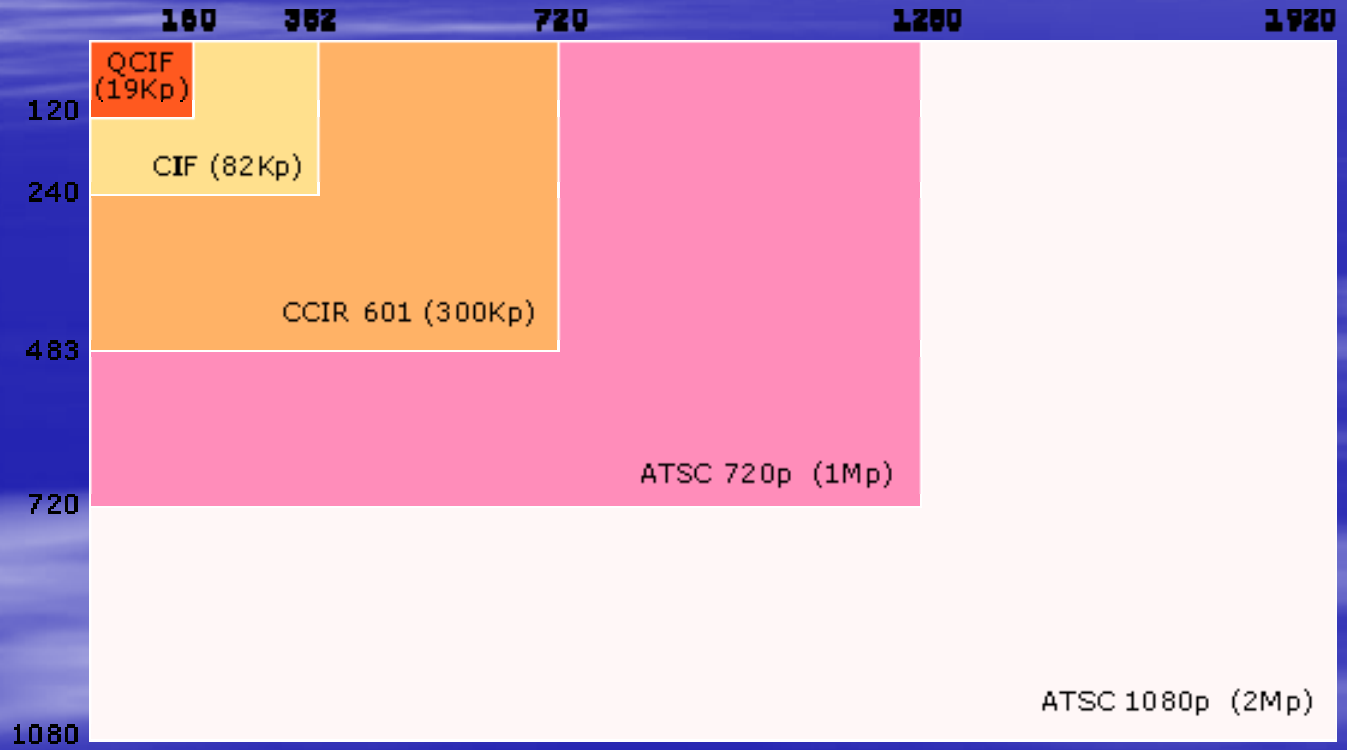
$$C_r = R - Y$$

Video color space - II

- Because human visual system is more sensitive to *luma* than *chroma,* in H264/AVC C_b, C_r components are subsampled and have ¼ of the number of samples than the Y component ➡ 4:2:0 sampling



Video formats



Qcif: Mobile video communication

Cif: Videoconference

CCIR: Standard Definition TV

ATSC: High Definition TV

Basic of video coding - I

- Reduce *redundancy* and *irrelevancy*
- Sources of redundancy:
  - temporal: adjacent frames highly correlated
  - spatial: nearby pixels often correlated (as in still images)
- Irrelevancy:
  - Perceptually unimportan information

Basic of video coding - II

- Spatial redundancy reduction (compression)
  - transform coding, spatial predictive coding
- Temporal redundancy reduction (compression)
  - motion compensation/estimation (MC/ME)
  - temporal predictive coding
- Entropy coding
  - from symbols to bits
- Bitstream syntax
  - specific vocabular

Basic of video coding - III

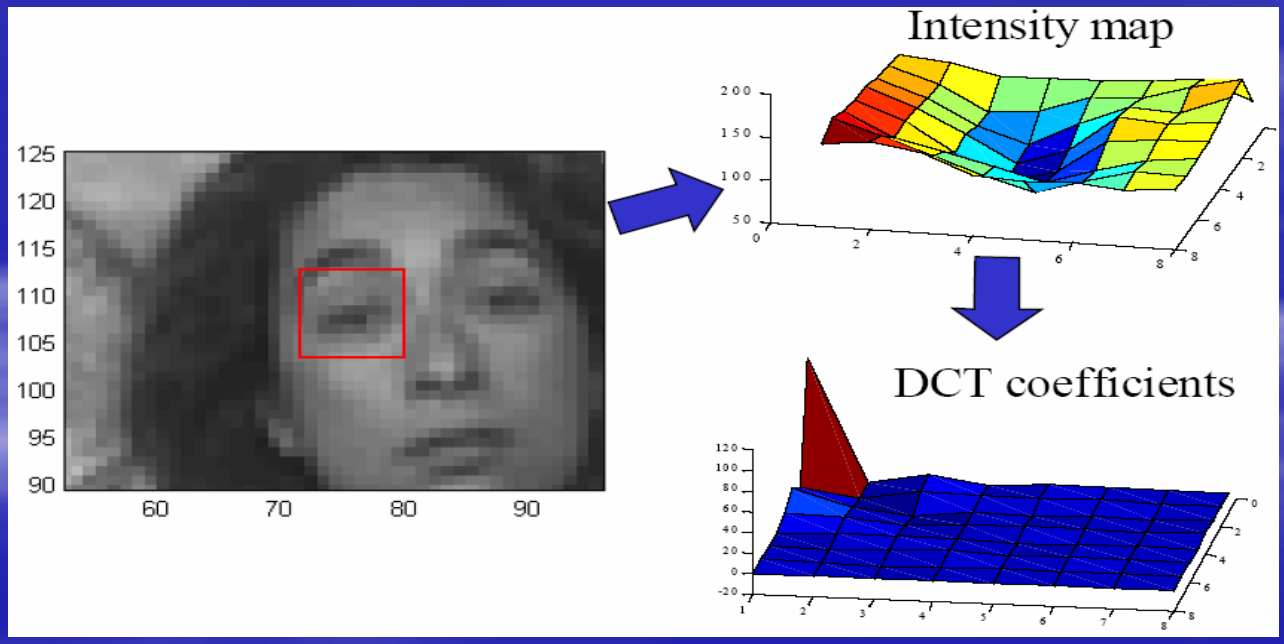
- To encode a frame each operation is performed at *macroblock* (*MB*) level (*n*x*n* block of pixel)
- Intra coded frame (I): every MB of the frame is coded using spatial redundancy
- Inter coded frame (P): most of the MBs of the frame are coded exploiting temporal redundancy (in the past)
- Bi-predictive frame (B): most of the MBs of the frame are coded exploiting temporal redundancy in the past and in the future.
- Group of Picture (GOP): sequence of pictures between two I-frames.

Encoding intra frames - I

- Spatial redundancy to be exploited
- *Intra* prediction: use Discrete Cosine Transform (DCT) to remove spatial pixel correlation
- DCT is applied to a $n \times n$ block: a block of coefficients with the same size is generated
- Intra frame are coded like a JPEG image
- First frame of the sequence is always coded intra

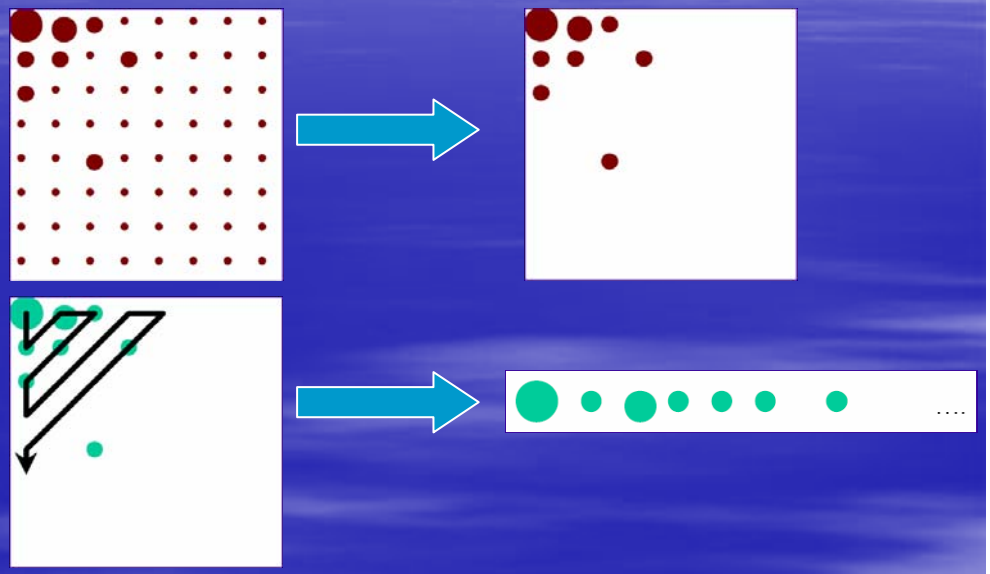
Encoding intra frames
DCT

- Transform a block of *n*x*n* pixels into a block of *n*x*n* spatial frequency coefficients
- Energy tends to be concentrated into a few significant coefficients
- Other coefficients are close to zero and not significative



Encoding intra frames - II

- Weighted scalar quantization: loss of precision, few non-zero coefficients are left
- Zig-zag scan: non-zero coefficients tend to be grouped together
- Run-Level encoding: encode each coefficient value as a (run,level) pair
  - run: number of zeros preceding values
  - lenght: non-zero value



Example:
Original data 14,3,4,0,0,-3,0,0,0,0,0,14,…
(Run,level) (0,14)(0,3)(0,4)(2,-3)(5,14)...

Encoding intra frames – III
Entropy Coding

- Entropy coding of R-L symbols: map symbols to bit assigning assign short code to more frequently occurring symbols

Encoding inter frames - I

- Main idea: predict current frame using previously coded one (*reference frame*)
- For each MB motion information is extracted from current and reference frame ➡ Motion Estimation (ME)
- Temporal predicted frame is obtained from reference frame using motion information estimated ➡ Motion Compensation (MC)
- The residual (original-MC) and motion information are coded using transform coding like for intra frame
- The prediction can be improved using for prediction even frames in the future (bi-directional prediction)

Encoding inter frames – II
Motion Estimation

- Find the MB in the reference frame which is the most similar to the current MB
- Error metric: MSE o SAD (*Sum of Absloute Differences*)

Encoding inter frames – III
Motion Estimation

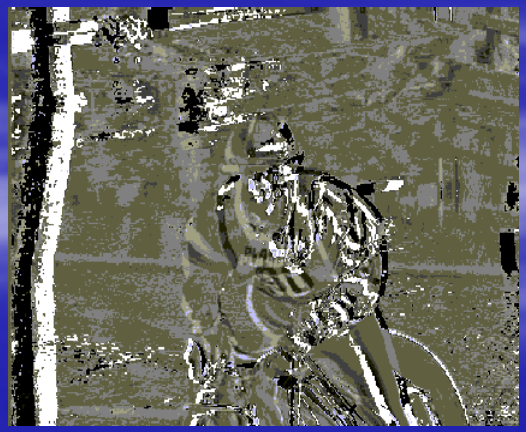
- Assuming a simplified translational model, motion information is described with 2 parameters for each block (*motion vector*)
- *Motion vector:* relative horizontal and vertical offsets (mv_1, mv_2) of a given macroblock from one frame to another
- *Motion vector field:* collection of motion vectors for all the macroblocks in a frame

Encoding inter frames – IV
Motion Compensation

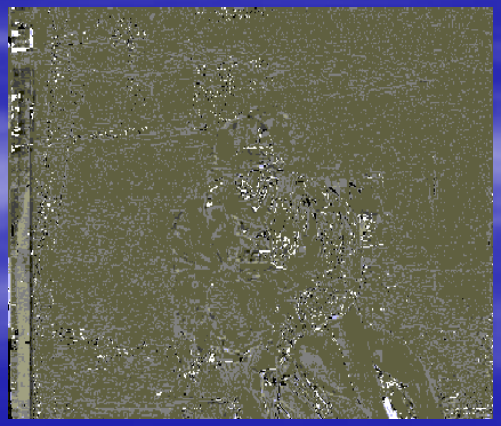
- Adding estimated motion information to the reference frame a ***motion compensated*** predicted frame is obtained
- The *Displaced Frame Differences* (*DFD = [original – MC]*) is calculated and encoded like for intra frames

Original Frame(N)

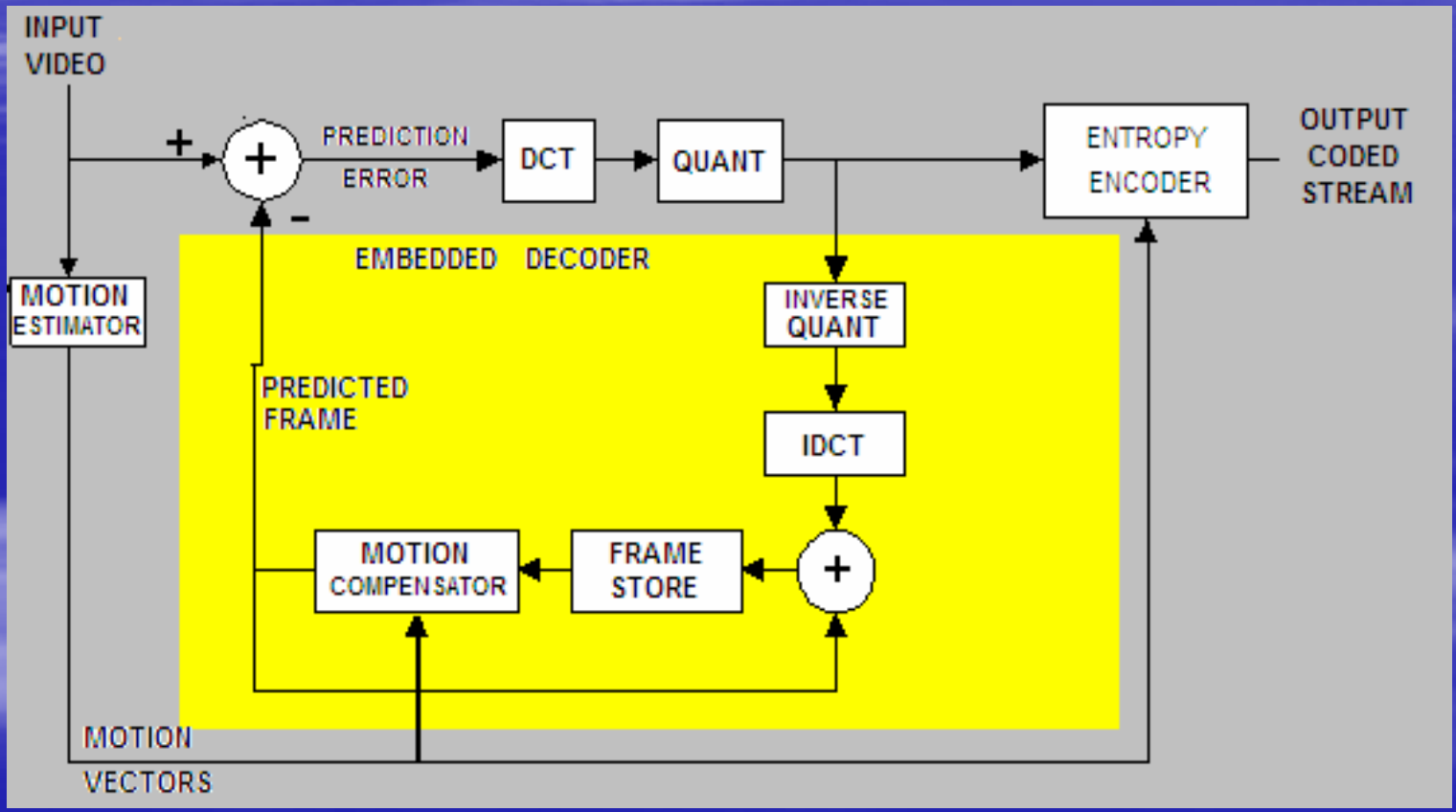
Frame(N)-Frame(N-1)



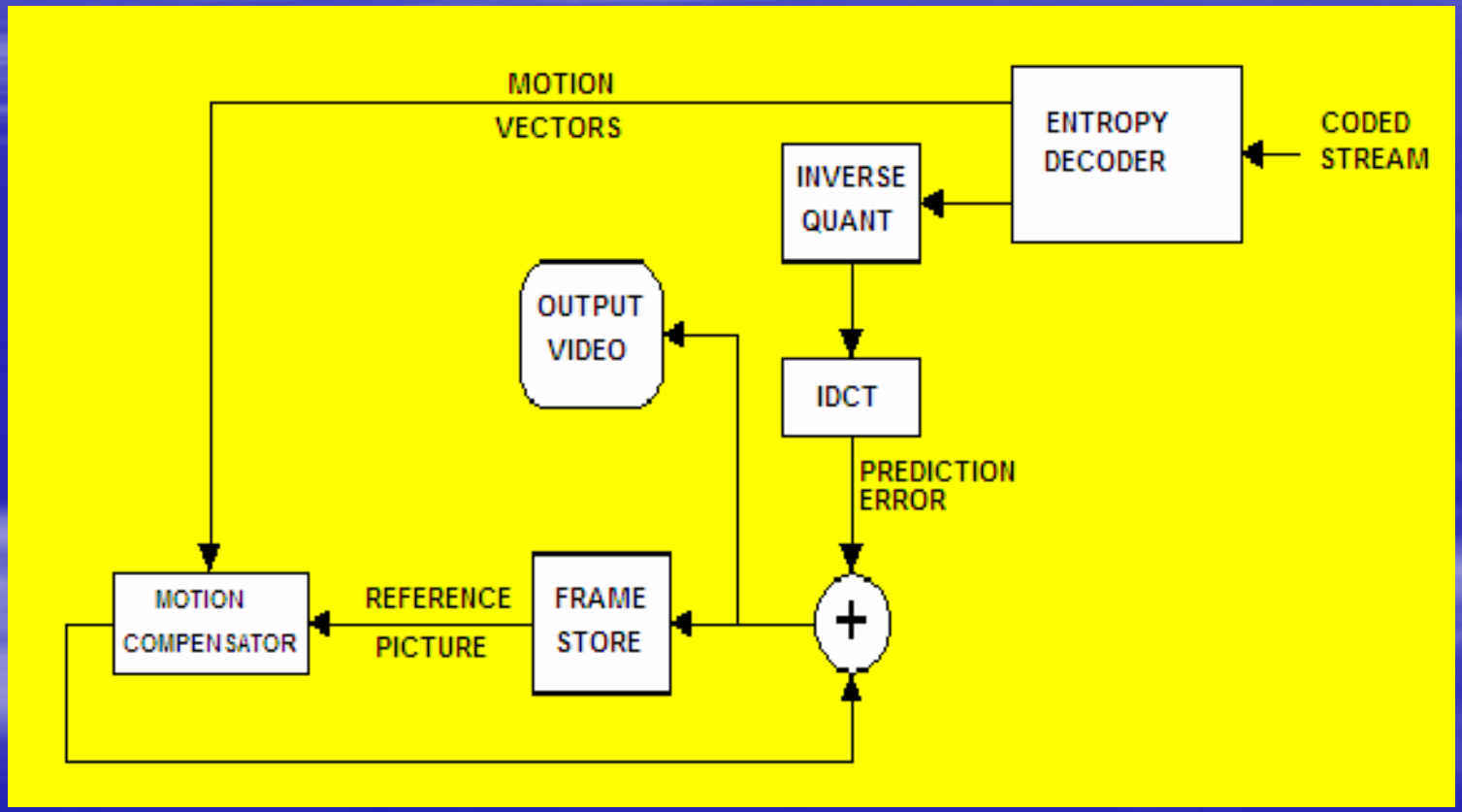
DFD



General video encoding scheme



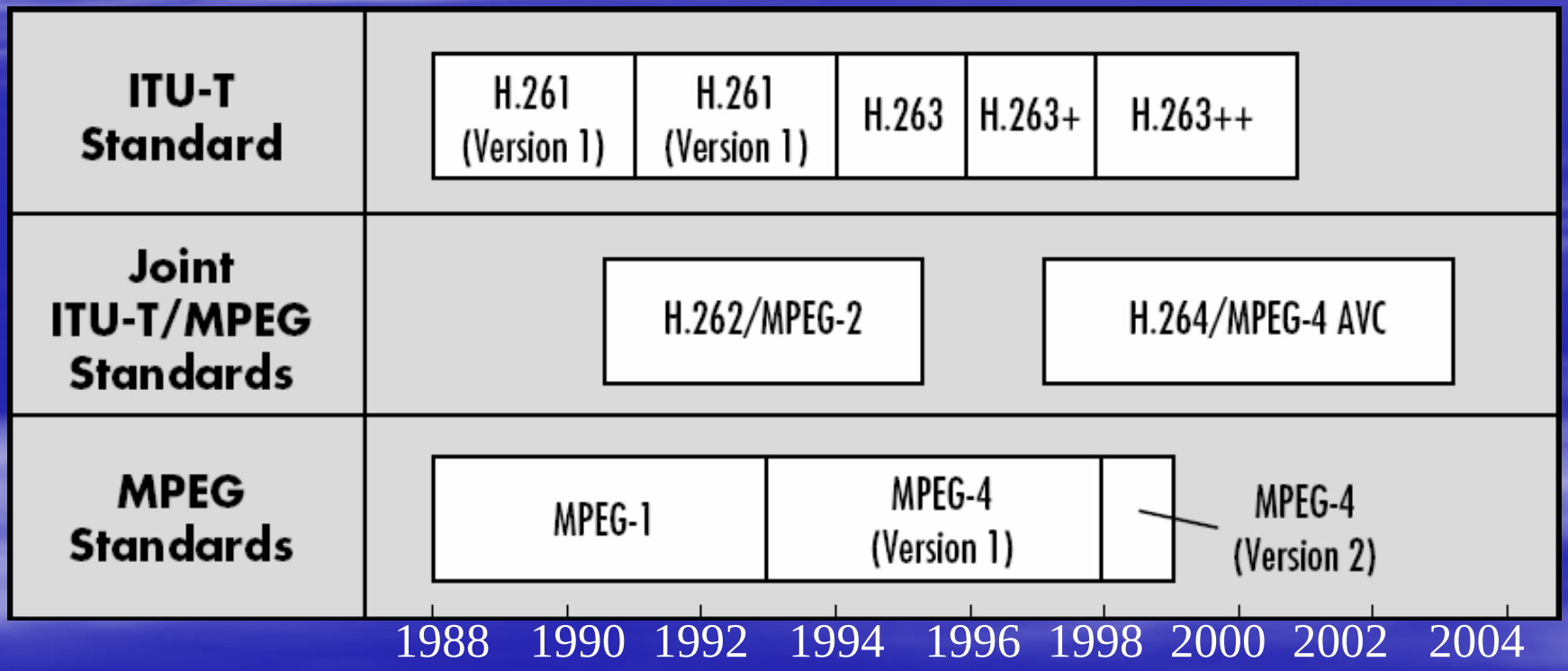
General video decoding scheme



Video compression standards– I

- Based on the same building blocks (ME/MC, DCT, ecc.)
- Additional tools added for different applications (improved compression, error resilience, scalability…)
- The standards specify the *bit-stream syntax* and the *decoding process* (e.g.: use IDCT but NOT how to implement IDCT)
- **ITU-T**: International Telecom. Union (Telecom. standardization). *"..development of standards that benefit telecommunication users worldwide… "*
- **MPEG**: Moving Picture Experts Group. Working group of ISO/IEC in charge of the development of standards for coded representation of digital audio and video.
- The evolution of the video coding standard is the natural answer to the growing demand for video-based services requiring increasingly higher coding efficiency

Video compression standards– II



MPEG-1

- Medium quality and medium bit rate video and audio compression
- Video on digital storage media
- Target bit-rate = 1.5 Mbit/s (CD-ROM storage, ~70 min of video on a CD support)
- CIF and CCIR 601 video format
- Is the engine of all the following standards MPEG-1,2,4

MPEG – 2

- Specification for broadcast TV
- Target bit rate: larger than 1.5 Mb/s, up to 35 Mb/s
- Extension of MPEG-1: higher quality, higher rate
- Supports interlaced TV systems
- Includes high quality audio
- Allows variable bit rate
- Usual figures: 4-8 Mbit/s
- Applications:
  - digital TV / HDTV
  - Terrestrial/Satelite broadcasting
  - Video editing and storage

Others Standards

- H.261 (ITU-T): ISDN video-phone and video conferencing at low bit rate and medium quality (64 kb/s up to 2Mb/s, p$\times$64 kb/s $1<p<30$). Hybrid DCT/DPCM coding algorithm with motion compensation.
- H.263 (ITU-T): very low bit-rate video coding (8 Kbit/s up to 1.5 Mbit/s). Video telephony over PSTN. Extension of H.261. After finalising the standard (1995), ITU-T started to work on 2 further projects:

  - *short term* effort: add extra features
  - *long term* effort: new standard H.26L with better video compression efficiency

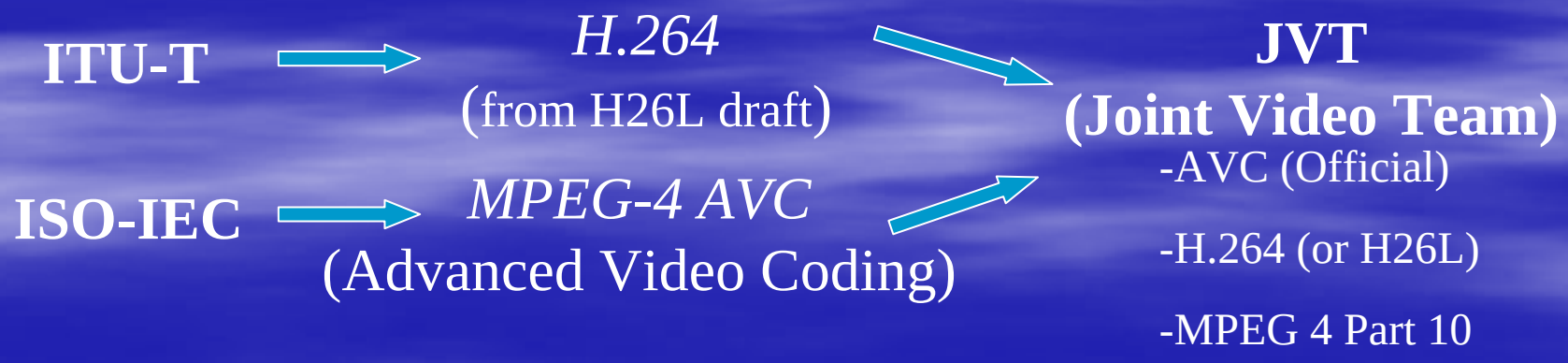
MPEG 4

- Second generation video coding technique: object based video coding
- High coding efficiency, very low bit rate (core coder)
- Robustness in error-prone environments (core coder)
- Content based interactivity (enables manipulation and editing, high interaction with scene content...extended core coder)
- Compatible with H.263, pretty close to MPEG-2

H.264/MPEG4 AVC

H.264 introduction - I

- Significant improvement over all previous video standard (2x compression, substantial perceptual quality)
- Jointly developed by ITU-T (H.264) and ISO/IEC (MPEG4). Approved in 2003



H264 introduction - II

- It will address the full range of video applications:
  - low bit-rate wireless applications
  - Standard definition/High definition broadcast television
  - Video streaming over internet
  - High definition DVD content
  - Digital cinema application
- Represents the single largest improvement in coding efficiency and quality since the introduction of MPEG-2
- H.264 is espected to displace MPEG-2/4 in many existing application
- Much higher complexity than previous standards (5/6 times more complex than MPEG-2)

H.264 introduction - III

Comparison of video coders (QCIF, 30 fps, 100 kbit/s)

Original

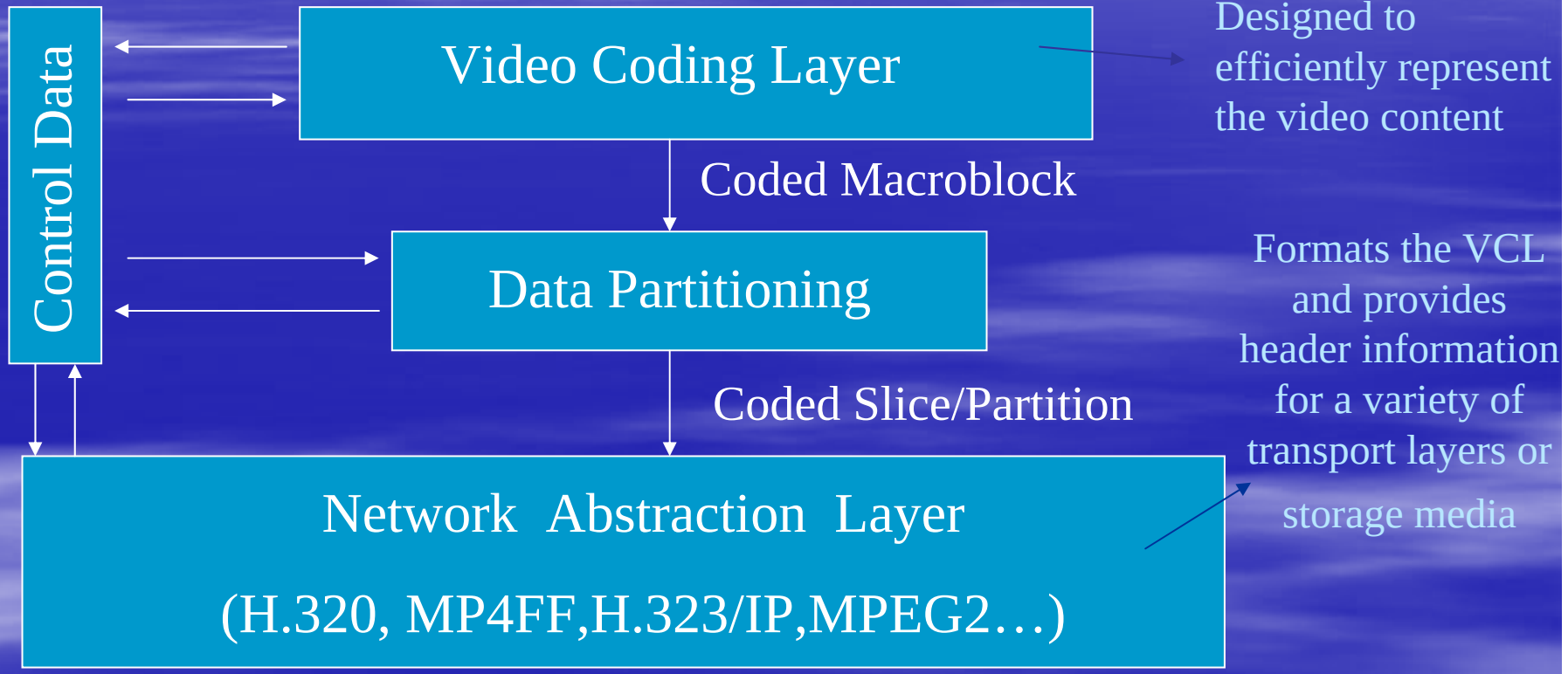
H.263 baseline (33 dB)

H.263+ (33.5 dB)

MPEG-4 core (33.5 dB)

H.264 (42 dB)

H.264 introduction - IV



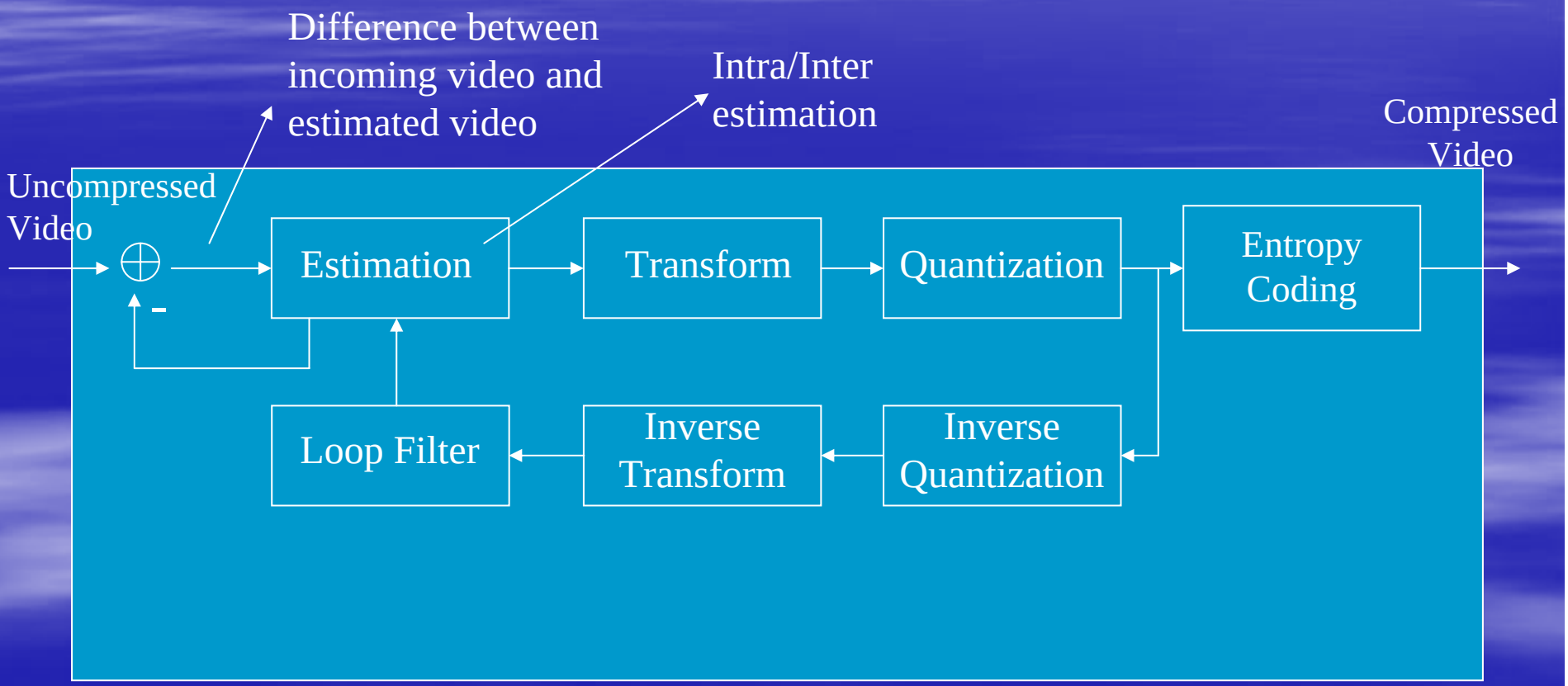
Network Abstraction Layer

- Map H.264/AVC VCL data to a broad variety of transport layers (RTP/IP for real time wire-line/wireless internet services, file formats ecc.)
- *NAL units*: packets that contains an integer number of bytes
- NAL units are classified as VCL and non-VCL:

  - VCL NALs contain data representing the values of the samples in the video pictures

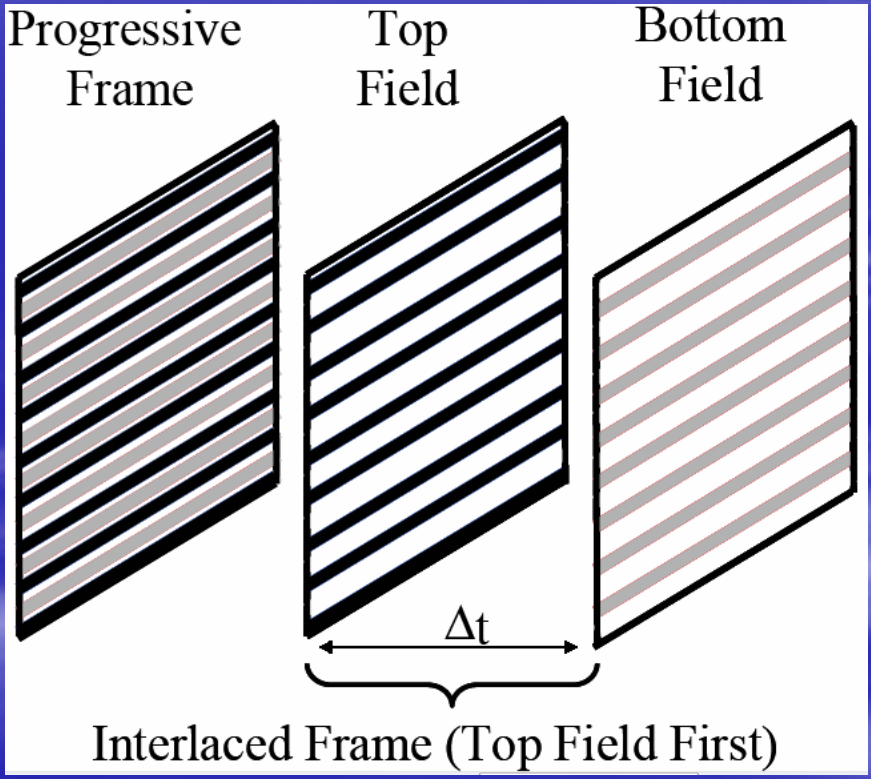
  - Non VCL NALs contain additional information (header or enhancement information)
- *Acces unit*: set of NAL units corrisponding to one decoded picture
- *IDR*: instantaneous decoding refresh access unit; contains an intra picture. Won't be a picture in the stream that will require a reference to any previous picture in the NAL unit stream.

H.264 technical overview - I



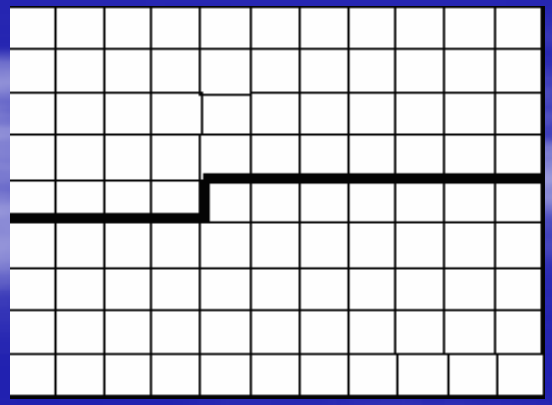
H.264 technical overview - II

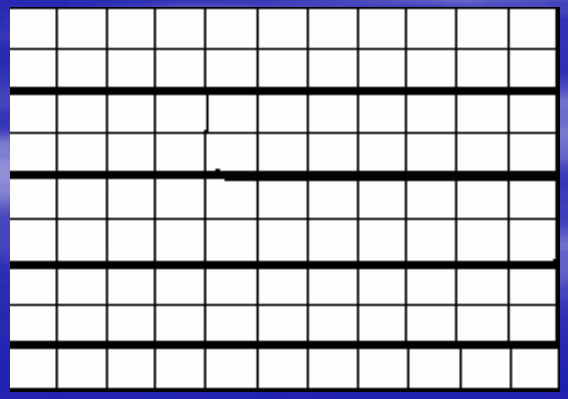
- A coded picture can represent an entire *frame* or a single *field*
- A *field* contains half of the lines of a coded frame
  - *top field*=>even numbered rows
  - *bottom field*=>odd numbered rows)
- *Interlaced frames*: the 2 fields of the frame are captured at diffente time instants
- *Progressive frame*: 2 fields captured at the same time



Macroblocks and slices

- A picture to code is partitioned into fixed-size area of 16x16 pixels (macroblocks, MB)
- MB is the basic coding element
- MBs are grouped into slice
- A slice is defined as self-contained: can be decoded without using data from other slices (i.e. no intra prediction between slice boundaries)
- Slice size is very flexible in H.264





Qcif frame:

-176x144

-11x9 MB

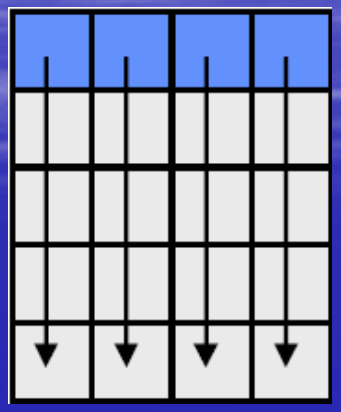
Intra-frame prediction - I

- Used for IDR/intra frame or when motion estimation cannot be exploited
- Attempts to predict the current block considering adjacent blocks in a defined set of different directions
- To improve prediction in parts of the picture with many details, MB can be broken down into smaller block
- **The difference between the predicted and the original block is then coded: approach useful for situations where spatial redundancy exists (flat background)**

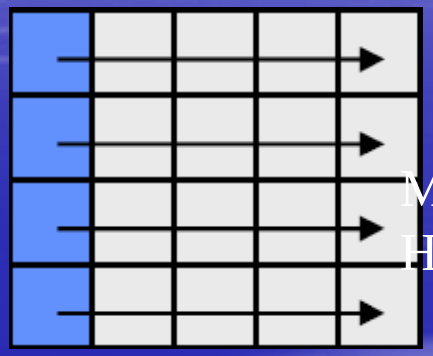
Intra-frame prediction - II

- Intra coding types:
  - Intra_4x4 => detailed areas
  - Intra_16x16 => smooth areas
  - IPCM => sampled are directly send (no prediction/no transform). Useful for anomalous picture content
- Intra coding modes (many!!): how samples of the current block are predicted
  - 9 modes for Intra_4x4 types
  - 4 modes for Intra_16x16 types

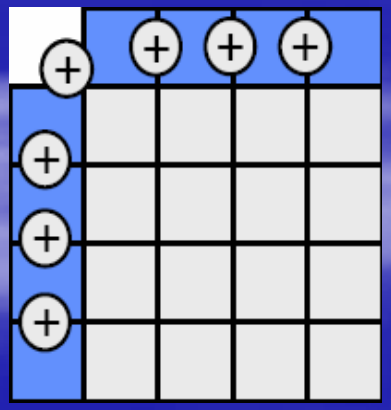
Intra frame prediction
Prediction modes (Intra_4x4)



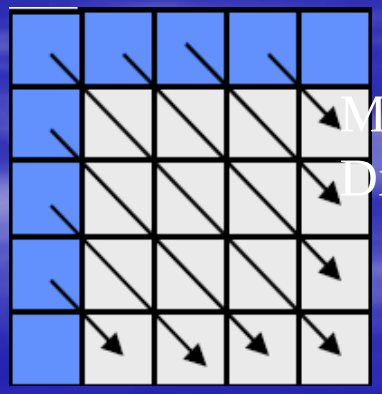
Mode 0
Vertical



Mode 1
Horizontal



Mode 2
DC



Mode 4
Diagonal Down/Right

Inter frame prediction - I

- Block size and shape variable and flexible to improve coding efficiency (16x16,16x8,..,8x8,4x8,...,minimun size 4x4)
- *mv* components are differentially coded using median or directional prediction from neighbouring blocks (like for intra coding)
- A maximum of 16 *motion vectors* can be transmitted for each MB (*greater precision* in motion vectors and strong motion isolation)
- A inter MB can also be coded in a *skipped.* No information (prediction error/mv) is transmitted.The reconstructed sampled are just copied from the reference frame in the same position (as if it were *mv=(0,0)*). Useful for coding large areas with constant motion

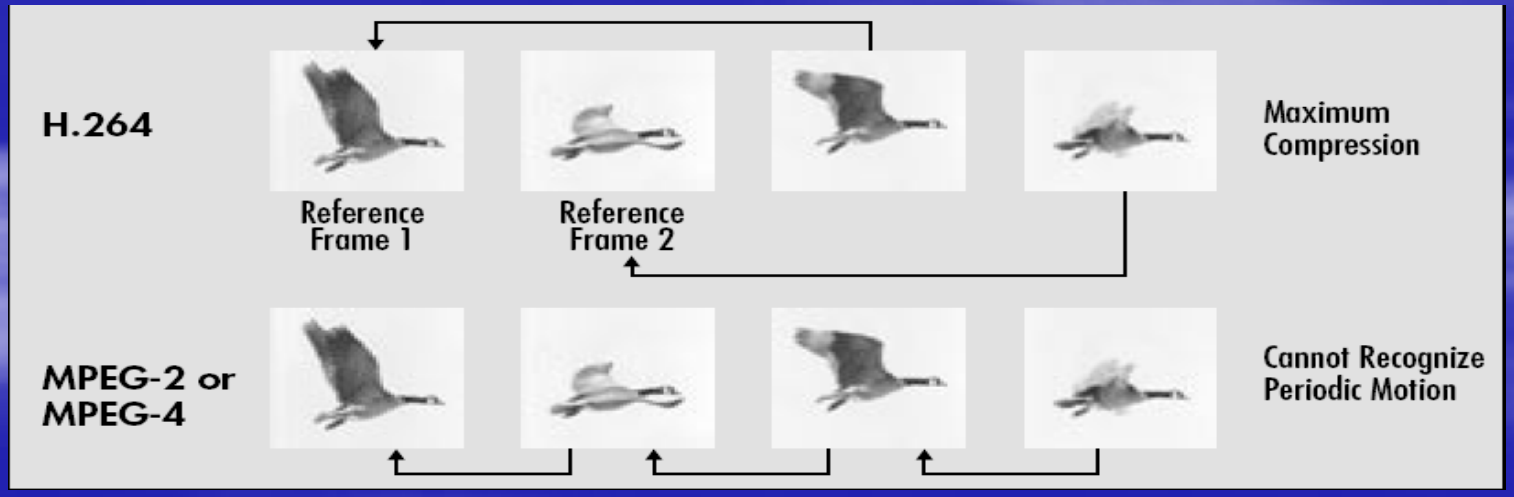
Inter frame prediction – II
Half/Quarter pixel ME

- To estimate sub-pixel motion (in case the *mv* points a non integer sample position) half/quarter sample motion estimation is enabled
- Half sample position are obtained by interpolation using a 6-tap FIR filter horizontally and vertically
- Quarter sample position are generated by averaging samples at *integer* and *half* position
- **Using *full/half/one-quarter* represents a great improvement compared to earlier standards:**

  More accurate motion rappresentation

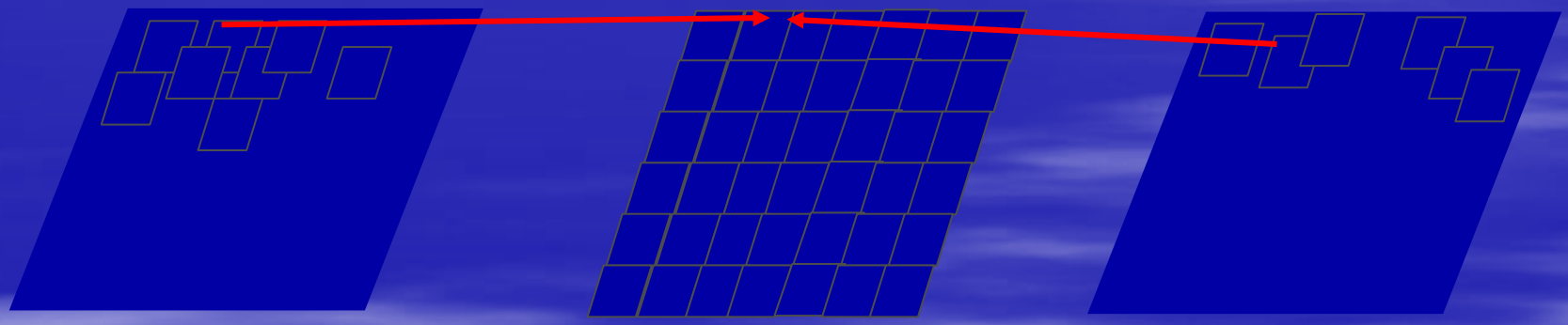
Inter frame prediction – III
Multiple Reference Frames

- The encoder can select for motion compensation among a larger number of pictures previosly decoded and stored
- Useful when dealing with motion that is periodic or in presence of camera switching between 2 scenes



B slices – I

- As in previous standards it is possible to temporally predict MBs from both previous and future coded frames



Previous frame

Current frame (B- frame)

Future frame

B slices - II

- In H264/AVC the concept of B slices is generalized
- B slices uses two lists of reference frames (past/future)
- For B slices prediction can be done in the past (*list0*), in the future (*list1*) or from both (bi-predictive: the prediction is an average of MC prediction signals)
- B slices utilize a similar MB partitioning as P slices (16x16, 16x8, 8x16 …)
- **Can be used as reference for prediction of other pictures**

Adaptive frame/field coding

- When dealing with interlaced frames, the encoder can decide to:

  - combine the 2 fields and code them as a single coded frame (frame mode)

  - code separately the 2 fields (field mode)

- If a frame consists of mixed moving and static regions is tipically more efficient to code non moving regions in *frame* mode and moving regions in *field* mode
- The frame/field encoding decision can also be made each pair of MBs (Macroblocks adaptive frame field, MBAFF)

H.264 transform coding

- As in previous standards transform coding of the prediction residual is used
- **In H.264/AVC transformation is applied to 4x4 block (instead of 8x8): blocking and ringing artifacts are reduced**
- **An integer DCT-like transform is used: exact-match inverse transform (exact equality of decoded video from all decoders)**
- Smaller transform => less computation

Quantization

- Reduces precision of integer coefficients and tends to eliminate high frequency coefficients
- A quantization parameter (*qp*) is used to determine perceptual quality (52 values)

qp 10

4.2 Mbps

qp 36

160 Kbps

Entropy coding - I

- Before entropy coding can take place, quantized coefficients are scanned in zig-zag fashion and serialized
- Differents methods of entropy coding are supported:
  - VLC (Variable Length Coding), Huffman-like
  - CAVLC (Context Adaptive VLC): for transmitting transform coefficients, an adaptive scheme is employed: VLC tables for each syntax element are switched depending on already transmitted syntax element

Entropy coding – II
VLC

- Uses conversion tables:
  - MB_Type => Code_number
  - Coeff_luma => Code_number
  - .....
- A single codeword table is used (exp-Golomb code)
- Used for all syntax element exept the quantized transform coefficients

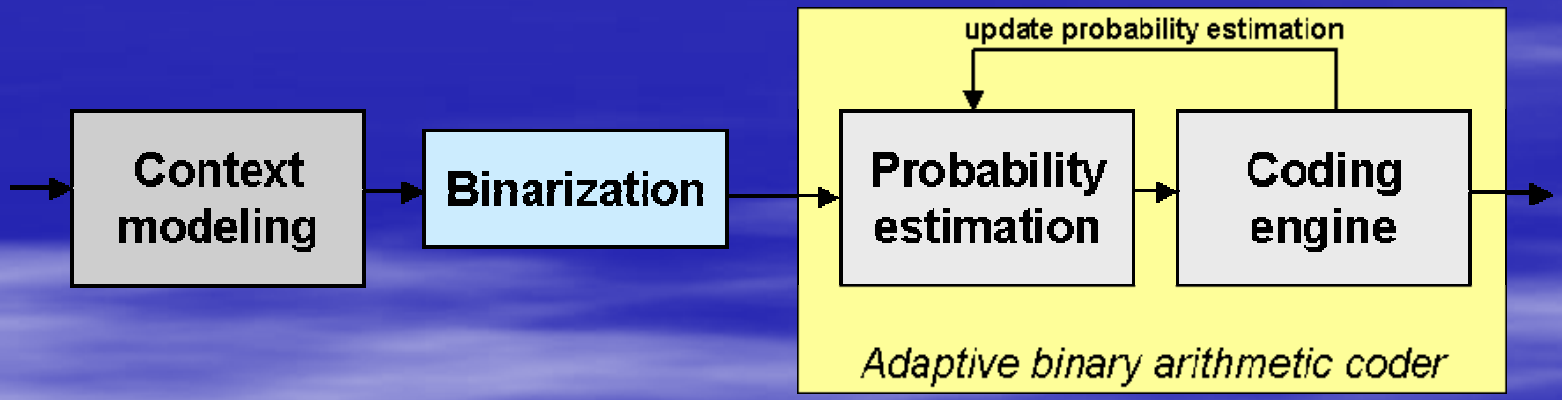
Code number	Codewords
0	1
1	0 1 0
2	0 1 1
3	0 0 1 0 0
..	. . . . .

Entropy coding - II

- Instead of CAVLC entropy coding can be further improved by using Context-Adaptive Binary Arithmetic Coding (CABAC)
- The usage of an arithmetic coding allows the assignment of a non integer number o bits to each symbol
- Context modeling: the statistics of already coded syntax element are used to estimate conditional probabilities: these are used for switching several probability models
- Compared to CAVLC, CABAC provides a reduction in bit-rate between 5-15%

Entropy coding - II

- Instead of CAVLC entropy coding can be further improved by using Context-Adaptive Binary Arithmetic Coding (CABAC)



Deblocking filter

- Block artifact is one of the main disavantageof block-based coding
- In H.264 an adaptive deblocking filter tries to smooth block edges
- The strength of the filter is controlled by the value of several syntax element (*qp*)

No deblock

Deblock

Profiles

- *Profile:* set of coding tools or algorithms that can be used in generating a conforming bit stream
- All decoders conforming to a specific profile must support all features in that profile
- *Baseline*, *Main*, *Extended* profiles

Error resilience Tools in H.264/AVC

Introduction

- Data communication across a channel may generate errors:
  - radio transmission (AWGN noise, fading, interference)
  - network transmission (congestion)
- *Error resilience* means:
  - adding redundancy to the bitstream
  - facilitate a better error concealment (i.e. recover lost parts of the picture)
- Apart from better coding efficiency, the standard has given strong emphasis to robustness to data errors/losses
- H.264/AVC employs various error resilience schemes

Parameter set and data partitioning

- *Parameter set*: key or consecutive coded video pictures is separated from coded representation of samples. Can be treated better protected (repeated, transmitted “out of band”)
- *Data partitioning(DP)*: enables unequal error protection according to syntax element importance. A normal slice can be partitioned into 3 parts (A/B/C) each one encapsulated into a separate NAL packet

  - DP-A contains header information more important than the remaining slice data (MB type, qp, mv ecc.)
  - DP-B contains intra coded block pattern (CBP) and transform coefficients of I-blocks
  - DP-C contains inter CBP and coefficients of P.blocks

Redundant Pictures

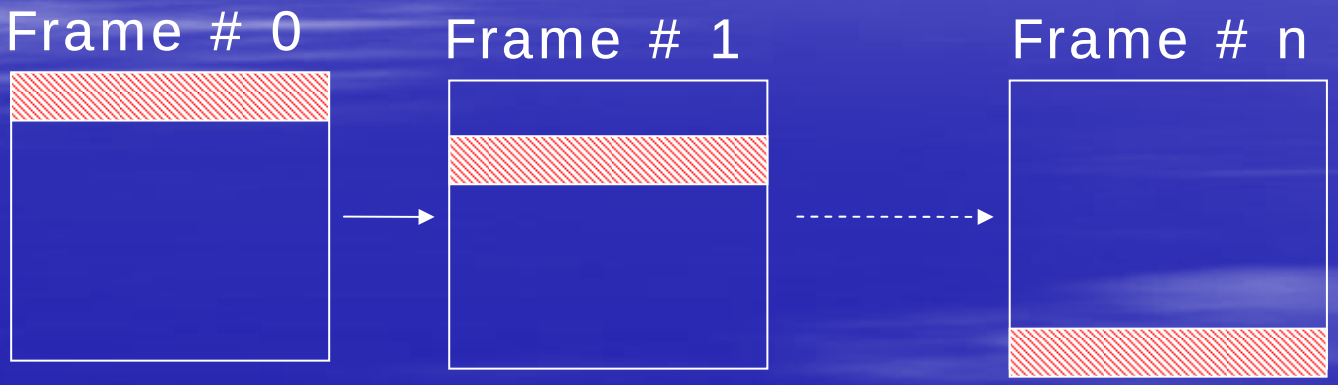
- The encoder has the ability to send redundant representations of regions of pictures
- This (tipically degradated) representation can be used to represent regions of pictures for which the primary representation has been lost during transmission
- The easiest approach is to retransmit the whole picture using a coarser quantizer

Intra refresh policies - I

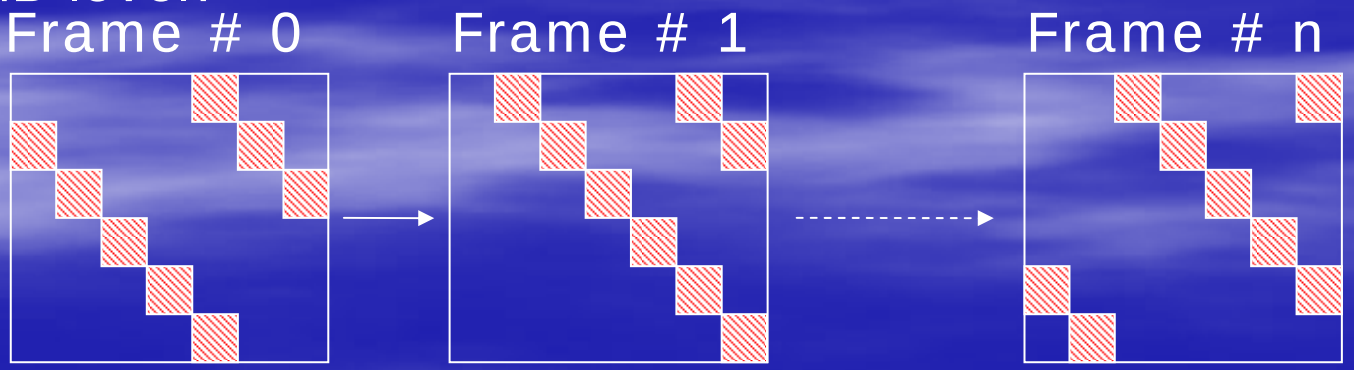
- To stop error propagation and drift due to predictive coding, an intra frame is inserted after a series of P/B-coded frames. This defines the GOP.
- As alternative, is possible to force the intra coding of a certain numbers of MB for each frame
- A complete intra refresh is perfomed after *n* frames, being 1/*n* the ratio of the total number of MBs for each frame forced to be intra coded
- Permits to avoid peaks of rate due to the intra code of a whole frame
- It's possible to tune the number of intra MB in function of channel conditions

Intra refresh policies - II

-Slice level:



-MB level:



MB forced to be intra coded

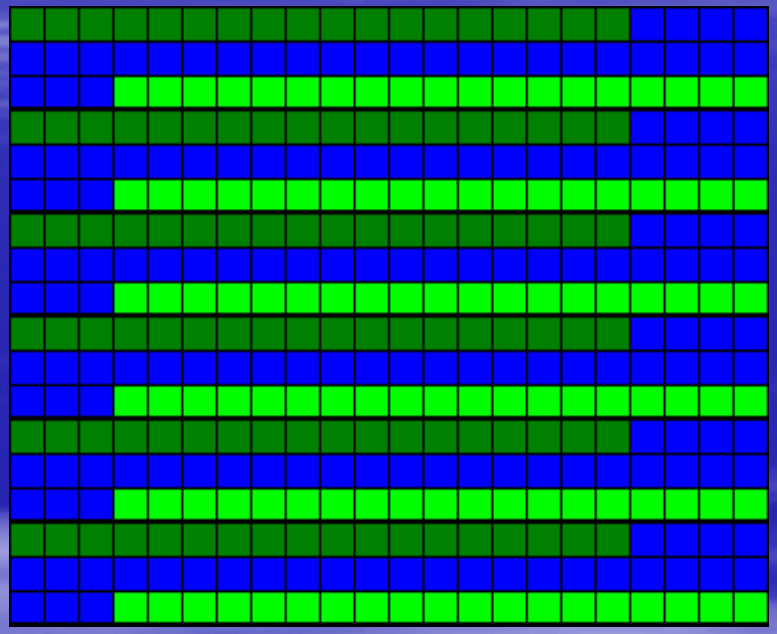
Applicazione

Flexible macroblock ordering FMO

- Inside a slice, MBs are usually processed in raster scan order
- Spatial and temporal prediction are confined within each slice
- *Slice group*: a set of MBs defined by a MB to slice group map. The processing order is defined by the map
- Each slice group can be partitioned into one or more slices
- Several mapping functions are defined by the standard (8 FMO types)
- Using FMO a picture can be split into many MB scanning patterns
  - Error robustness ☺
  - Coding efficiency ☹

Flexible macroblock ordering

Interleaved slices map group



=> slice group #0

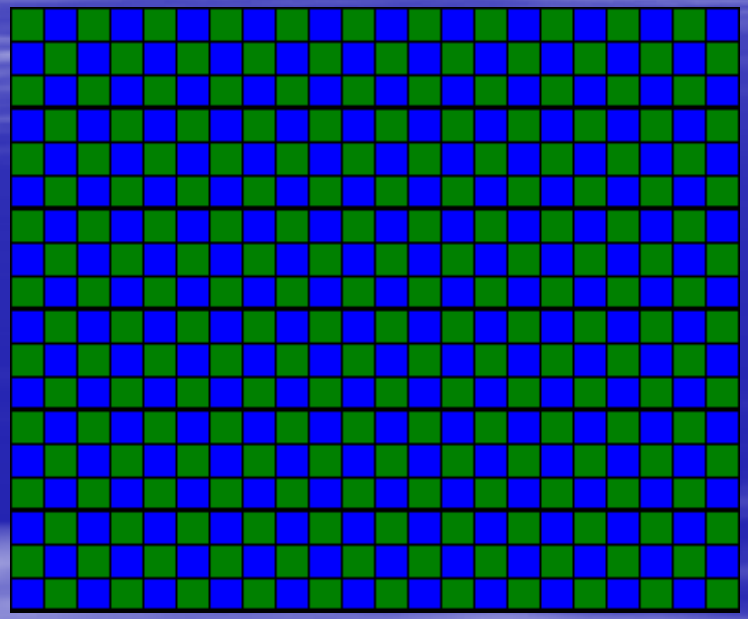
=> slice group #1

=> slice group #2

- Example for a CIF frame (22x18 MBs)
- # of slice groups -> 3
- Slice group map type -> 0
- Run lenght slice group 0 -> 18
- Run lenght slice group 1 -> 29
- Run lenght slice group 2-> 19

Flexible macroblock ordering

Scattered slices map group



=> slice group #0

=> slice group #1

- Example for a CIF frame (22x18 MBs)
- # of slice groups -> 2
- Slice group map type -> 1

Applicazione

Conclusion

Conclusion

- H.264/AVC offers significant bit rate and quality advantages over all previous standards (50% bit rate saving for equivalent perceptual quality)
- In addiction to excellent coding efficiency, the network adaptation provides large flexibility for its use in a broad variety of network types and application domains
- **Important differences:**

  - Enhanced motion prediction capability
  - Use of a small block-size, exact-match transform
  - Adaptive in-loop deblocking filter
  - Enhanced entropy coding methods