Video Compression

Dr. Kemal Ugur

Research Leader, Nokia Research Center

kemal.ugur@nokia.com

kemal.ugur@avcrtech.com

NOKIA
Connecting People

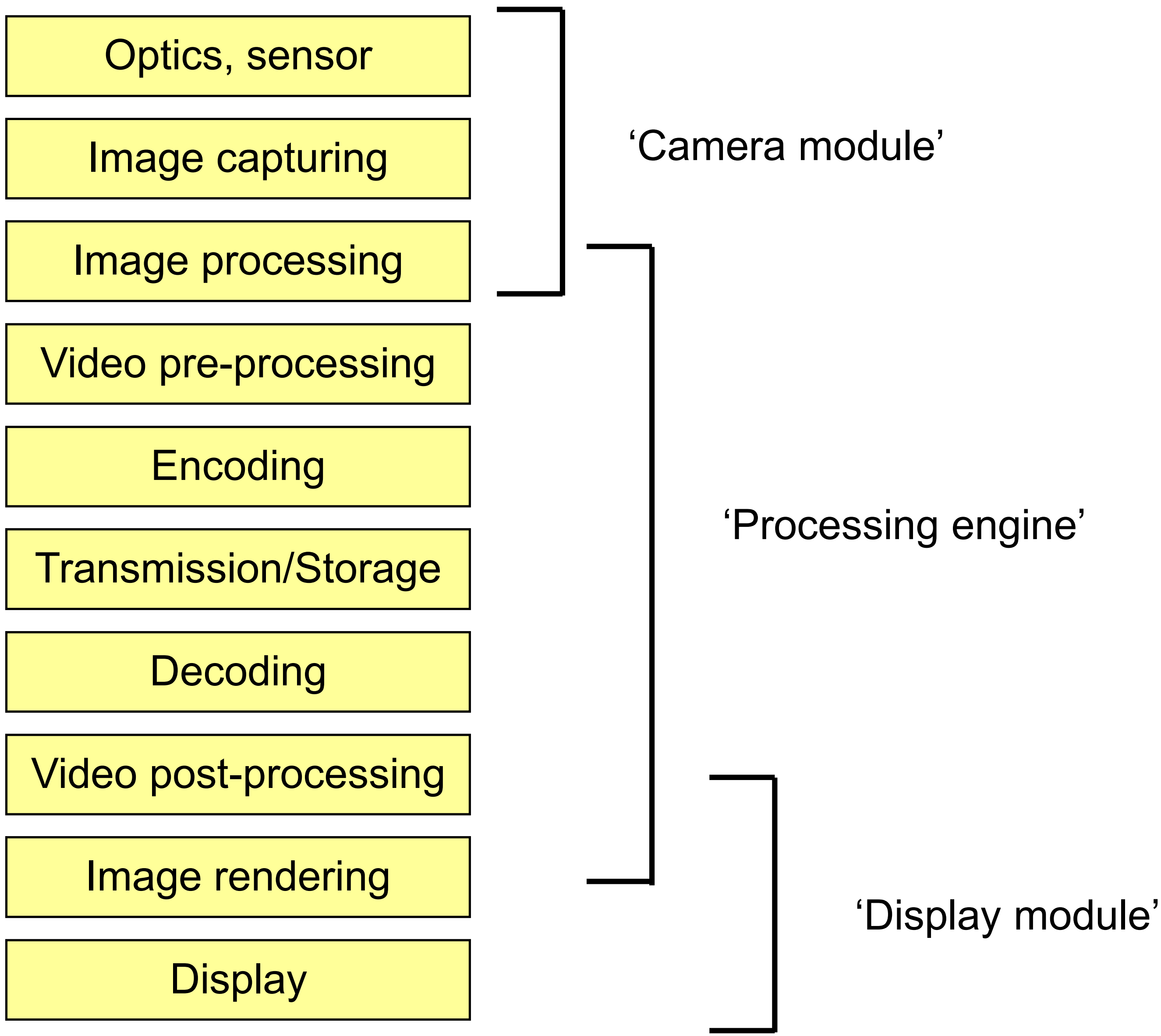
About the presenter

Participated to almost all JVT/MPEG/VCEG meetings since 2006.

Led the Nokia team in JCT-VC and have been active in H.265/HEVC development since its beginning.

- Our proposal was one of the winning proposals where the initial design of H.265/HEVC standard is based upon
- Authored / co-authored more than 100 contributions
  - Many of those are included in the final standard

Typical Image/Video Processing Chain



Remarks on Processing Steps

- Optics, sensor
- Image capturing
- Image processing
- Video pre-processing
- Encoding
- Transmission/Storage
- Decoding
- Video post-processing
- Image rendering
- Display

- Standardized parts
  - Communication protocols
  - Decoding of error-free bitstreams
- Huge differences in quality can be achieved in the other processing steps
- Some algorithms depend on HW
  - Pre-processing includes "concealment" of camera HW artefacts
  - Post-processing includes "concealment" of display imperfections (e.g. contrast enhancement)
- Some algorithms depend on target application
  - E.g. different encoding algorithm for error-free transfer of files (such as MMS) and error-prone transmission (such as video telephony)

Video Applications

- VCD, DVD, Blu-ray
- Digital TV Broadcasting
- Video conferencing and telephony

- Internet video
  - Video on demand
  - P2P TV, Downloading
  - Video Sharing, Personal Broadcasting
- Mobile video
  - Camcorder, MMS, streaming, video telephony, mobile TV

Imaging Basics

Digital Image Representation

- Displayed images are composed of units called picture elements, pixels or pels.

Height

Pixel

Red = 246
Green = 148
Blue = 134

24 bit image.
(8 bits / component)

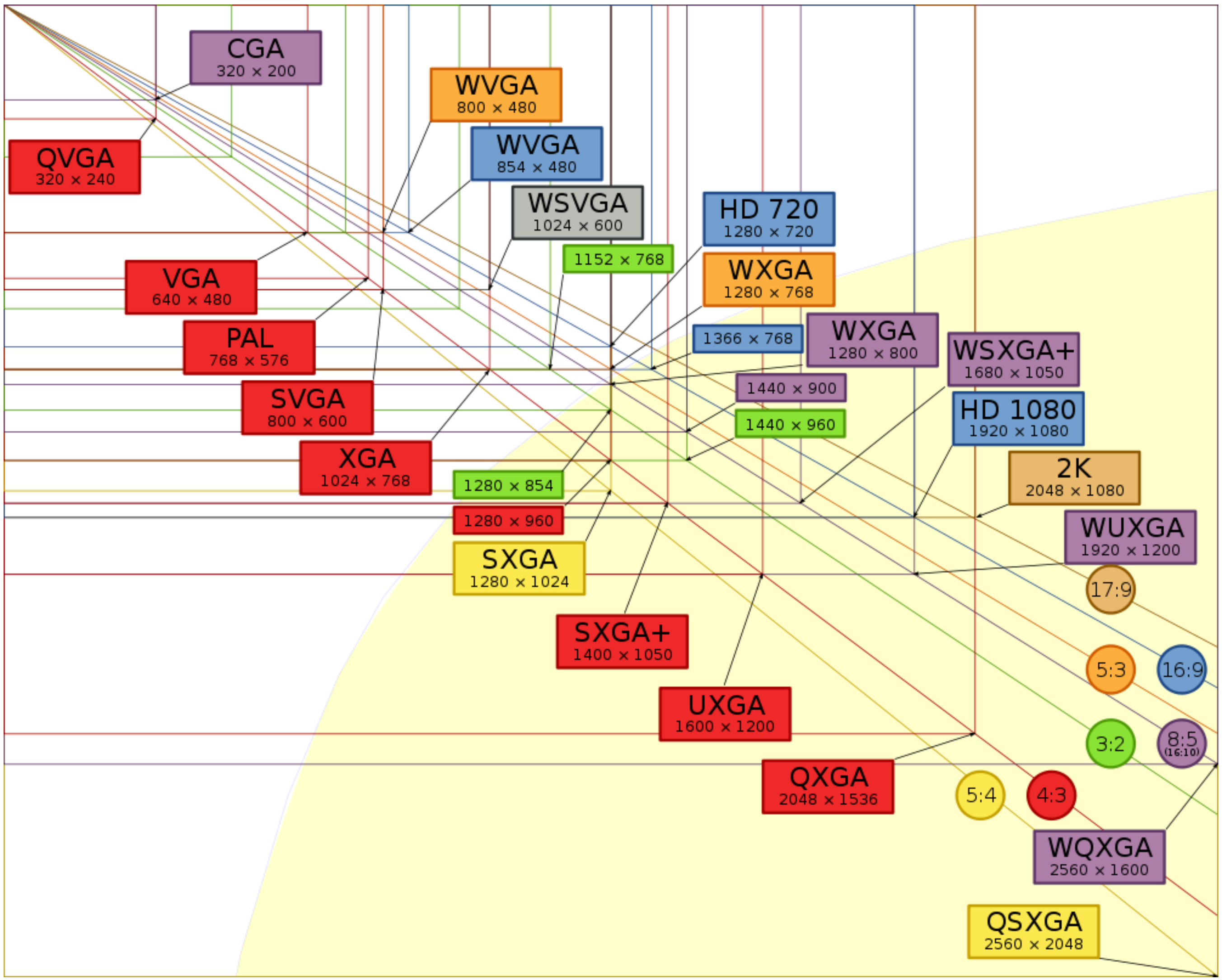
Width

Basic Definitions

- Pixel = typically three components: red (R), green (G), blue (B)
- Digital picture = array of pixels
- Digital video = sequence of digital pictures

- Picture = frame or field
- Frame = all pixels in picture area
- Field = alternate rows of a frame
- Progressive vs interlaced video systems
  - Progressive – handles frames
  - Interlaced – handles fields, fields refreshed separately

Picture Formats



Source: Wikipedia

NOKIA
Connecting People

Image Resolution

- Resolution: number of pixels (or samples)

  - Indicates theoretical amount of information in an image. (Maximum resolving power, highest spatial frequency)

  - When used in context of an camera does not necessary tell very much on image 'sharpness'. Transfer function of optics and camera together define the true resolving power.

Size vs. Resolution

- Same size but not the same information content.

Image/Video Compression

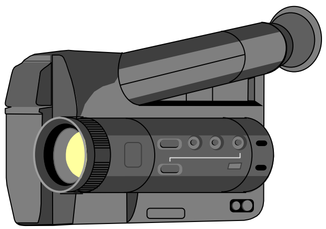
Connecting People

Why is visual compression important ?

- "An image is worth 1000 words" …

… but it also requires a much larger storage space.

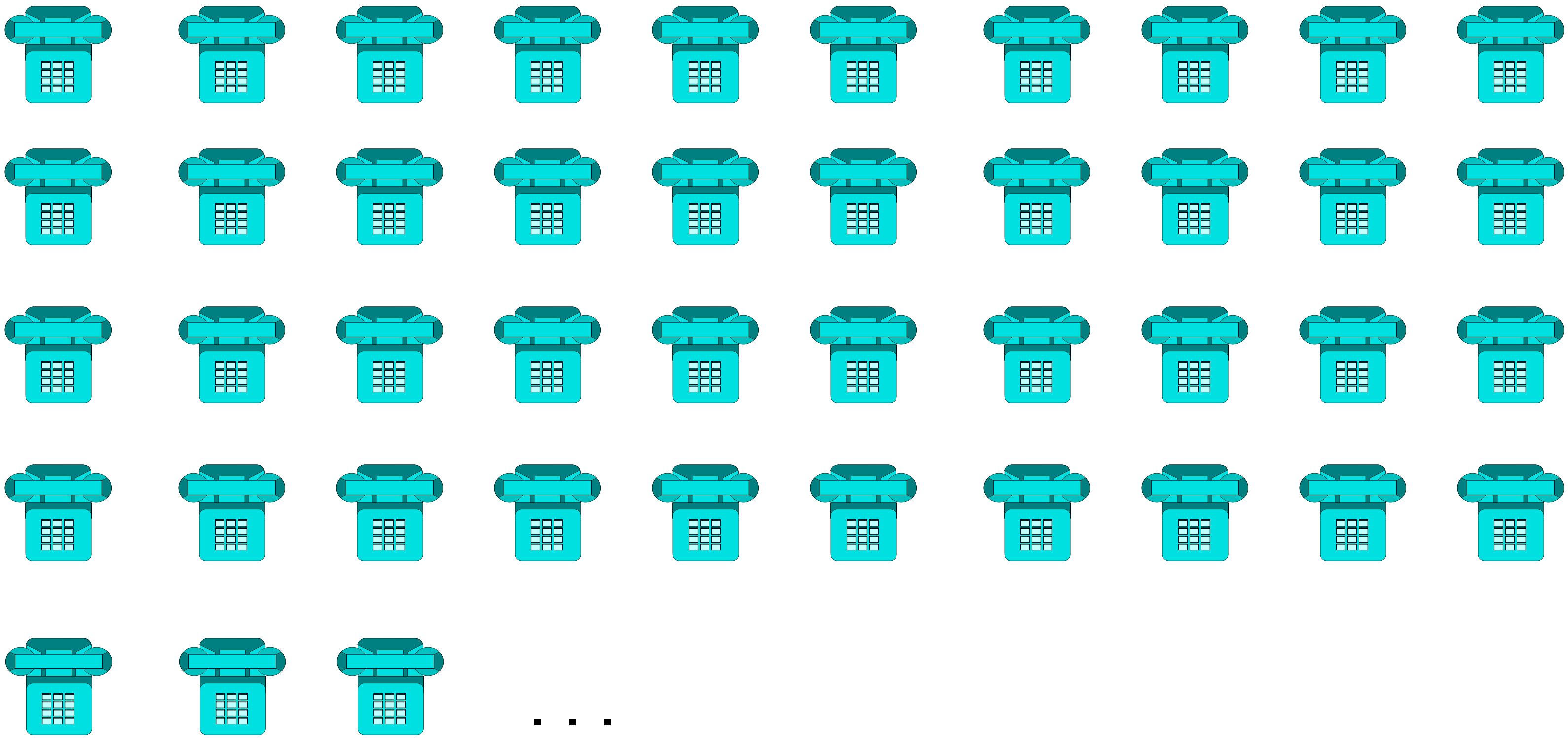
Example



Two hours of video correspond to roughly …

Why is visual compression important ?

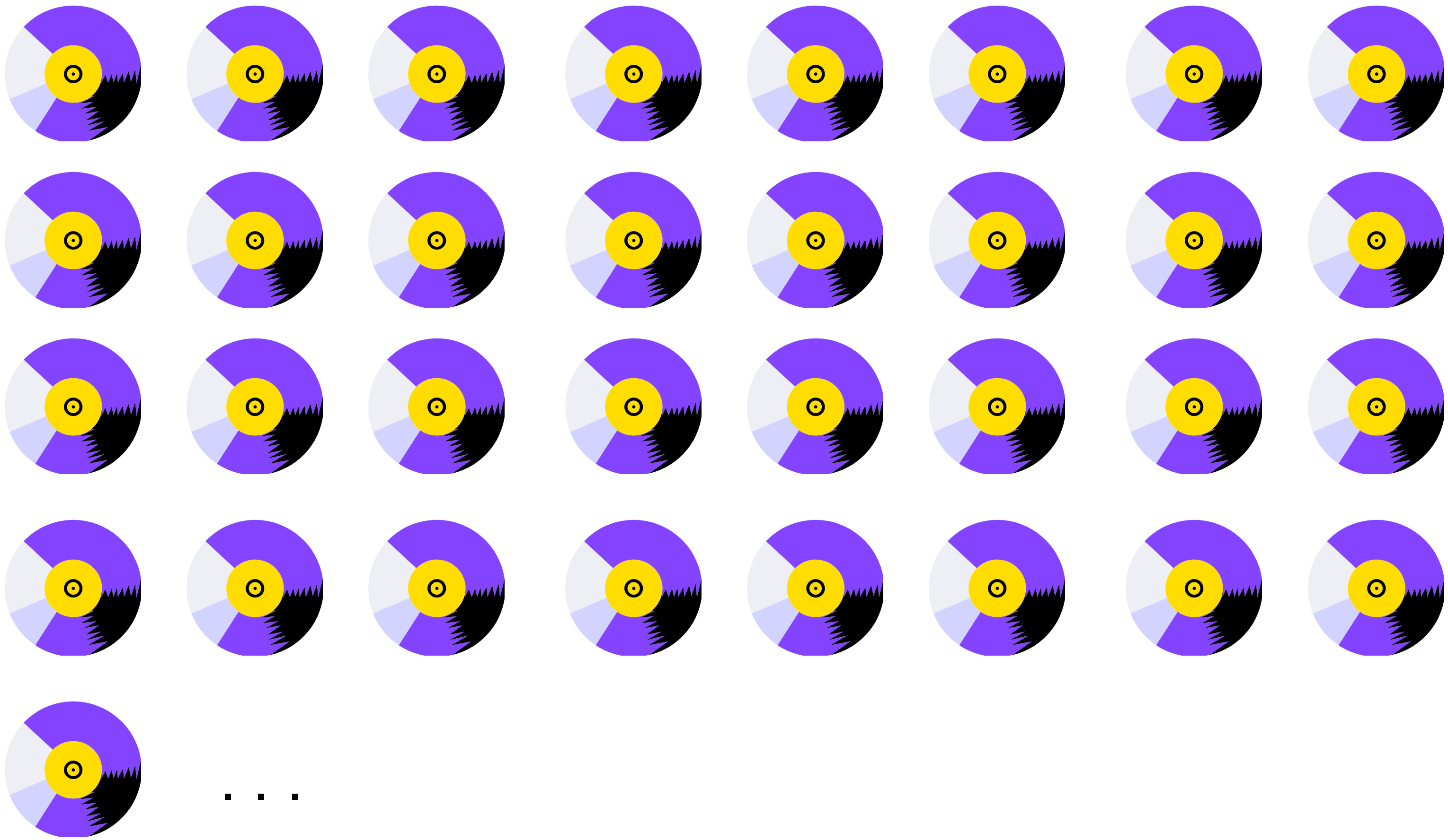
- … 8000 hours of telephone conversation ...



. . .

Why is visual compression important ?

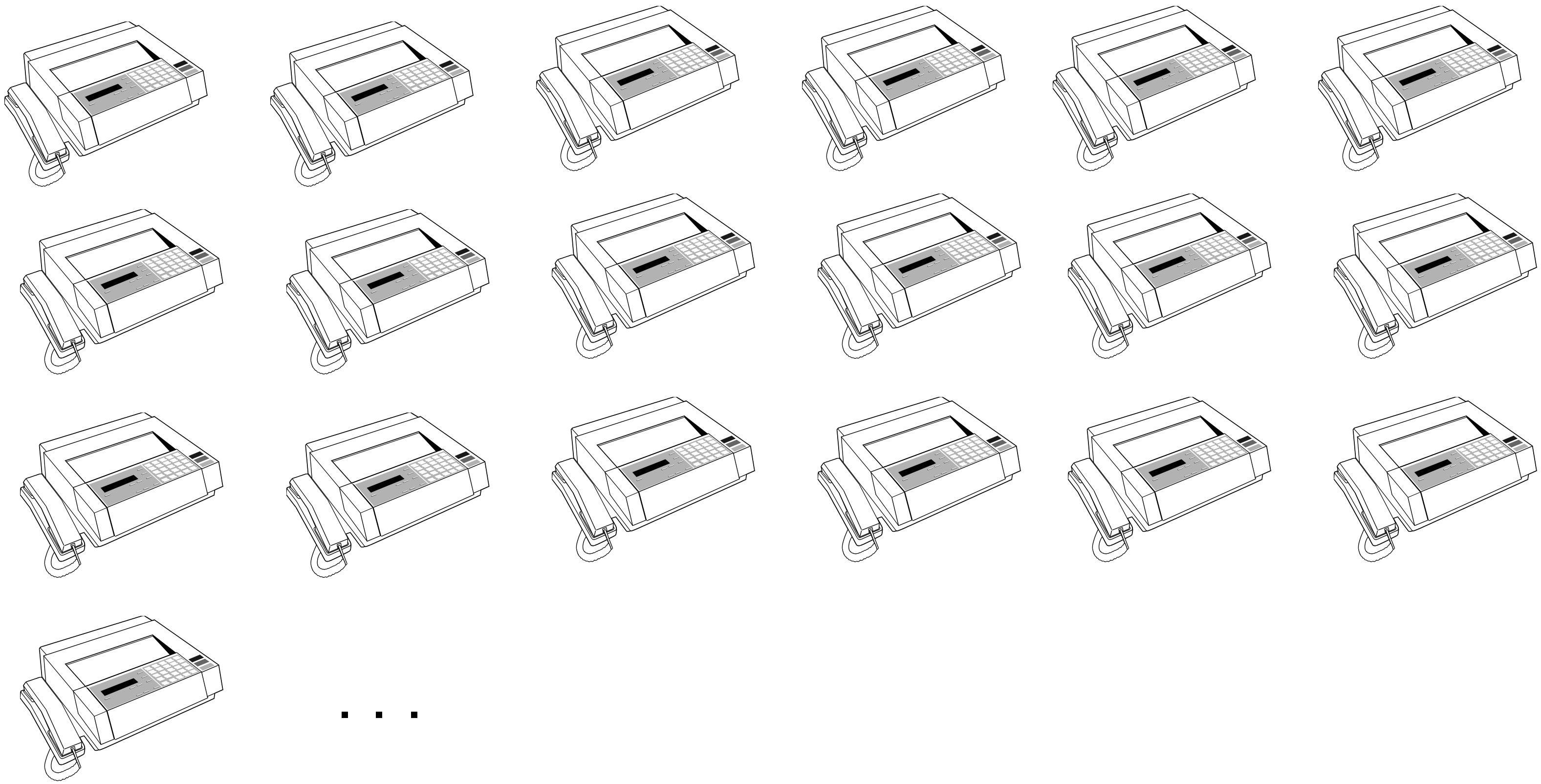
- … 150 audio compact disks ...



. . .

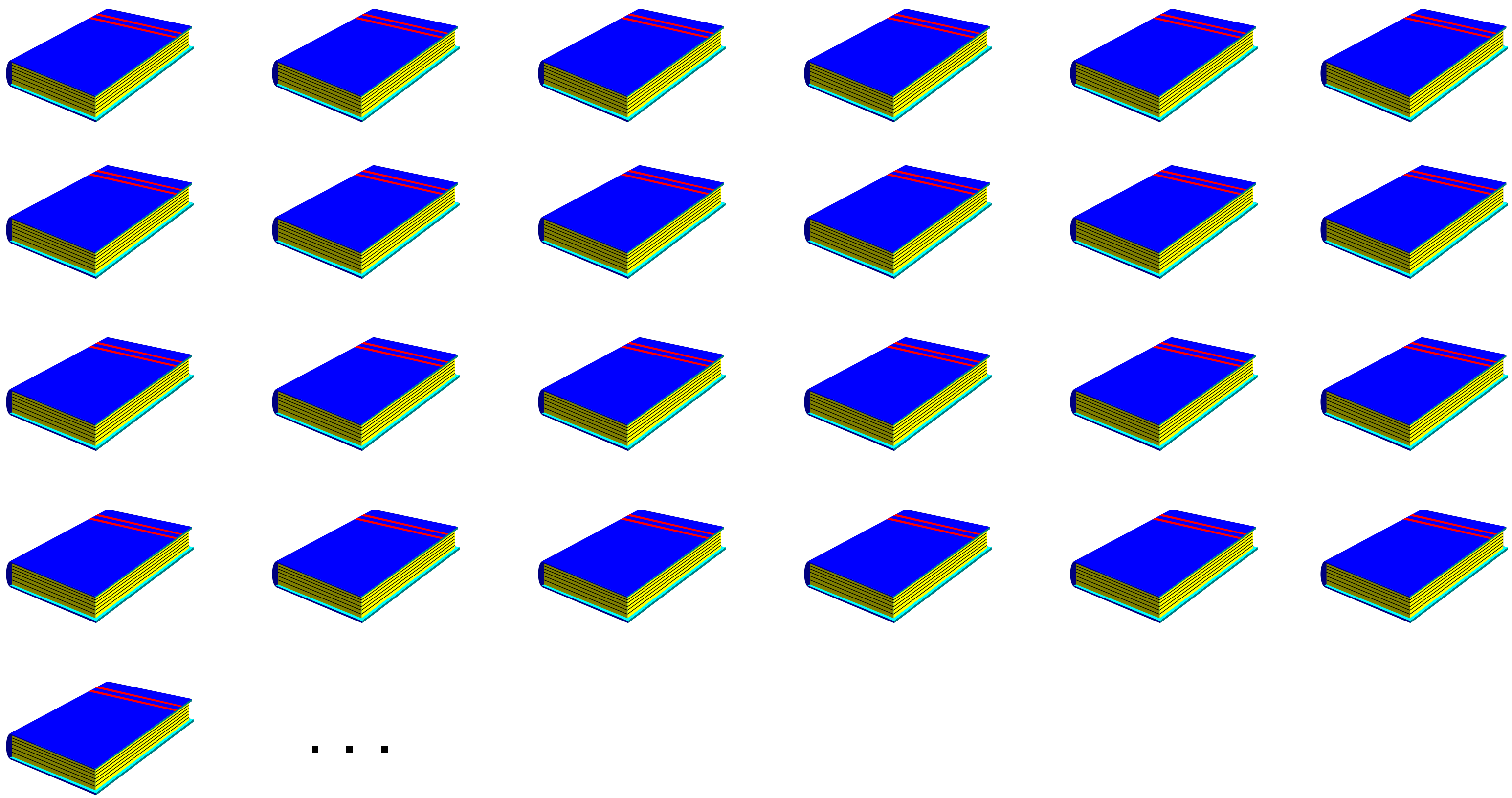
Why is visual compression important ?

- … 250’000 pages of fax ...



Why is visual compression important ?

- … 100’000 books.



Video Compression

Typical Uncompressed Video : 24 bits per pixel

Compressed Video could be: 0.02 – 0.3 bits per pixel

Compression by factor of **1000!!!**

Introduction of Video Compression

- A video coder can achieve 1000x compression. But how?
- In a video signal, there exists redundancies and a video coder exploits these to achieve compression
    1. Spatial Redundancy: A pixel in the frame is correlated to the neighboring pixels.
    2. Temporal Redundancy: A frame in the video is slightly different from the neighboring frames.
    3. Statistical Redundancy: Not all pixels in an image (or in transformed image), occur with the same probability.

- In addition, video coders exploit human-visual system to achieve increased compression ratios:
    - E.g. : Color subsampling

- The amount of compression achievable depends very significantly on the content of the video material
    - Easily a factor of 10 difference between “easy” and “difficult” sequences

NOKIA
Connecting People

Introduction of Video Compression

- Spatial Redundancy

Examples of correlated areas.

Instead of coding each pixel independently, a video coder tries to use information from neighboring pixels

Introduction of Video Compression

- Temporal Redundancy
  - Each video frame differs from neighboring ones slightly
  - Higher compression rations could be achieved by using information from neighboring frames

History of Video Compression

- The technical developments in this area happen within standardization bodies these days.
  - Motion Pictures Experts Group (MPEG) of ISO-IEC
  - Video Coding Experts Group (VCEG) of ITU-T
- Latest video coding standard is H.265/HEVC, finalized in 2013, by Joint Collaborative Video Team (JCT-VC) of MPEG and VCEG
  - Delivers the same subjective quality over H.264/AVC at half the bitrate.
- Earlier standards include MPEG-2 (used in DVD), MPEG-4 (used in DivX), H.263, H.263+ (used in video conferencing)
- But the history of video compression dates way back…

NOKIA
Connecting People

[Second Edition.]

PATENT SPECIFICATION

Convention Date (United States): April 25, 1929.

Application Date (in United Kingdom): April 25, 1930. No. 12,805 / 30.

Complete Specification Accepted: Jan. 22, 1931.

341,811

COMPLETE SPECIFICATION.

Improvements relating to Electric Picture Transmission Systems.

We, THE BRITISH THOMSON-HOUSTON COMPANY LIMITED, a British Company, having its registered office at Crown House, Aldwych, London, W.C. 2, (Assignees of RAY DAVIS KELL, of 111, Sanders Avenue, Scotia, County of Schenectady, State of New York, United States of America, a citizen of the United States of America), do hereby declare the nature of this invention and in what manner the same is to be performed, to be particularly described and ascertained in and by the following

fineness of detail is limited only by the speed of the action to be transmitted.

The invention will be better understood from the following description when considered in connection with the accompanying drawings in which Fig. 1 illustrates a picture transmitting apparatus wherein the invention has been embodied; and Figs. 2 to 5 illustrate various details of an apparatus which may be utilised to receive the difference between the successive images of a picture or moving object.

It has been customary in the past to transmit successive complete images of the transmitted picture.”

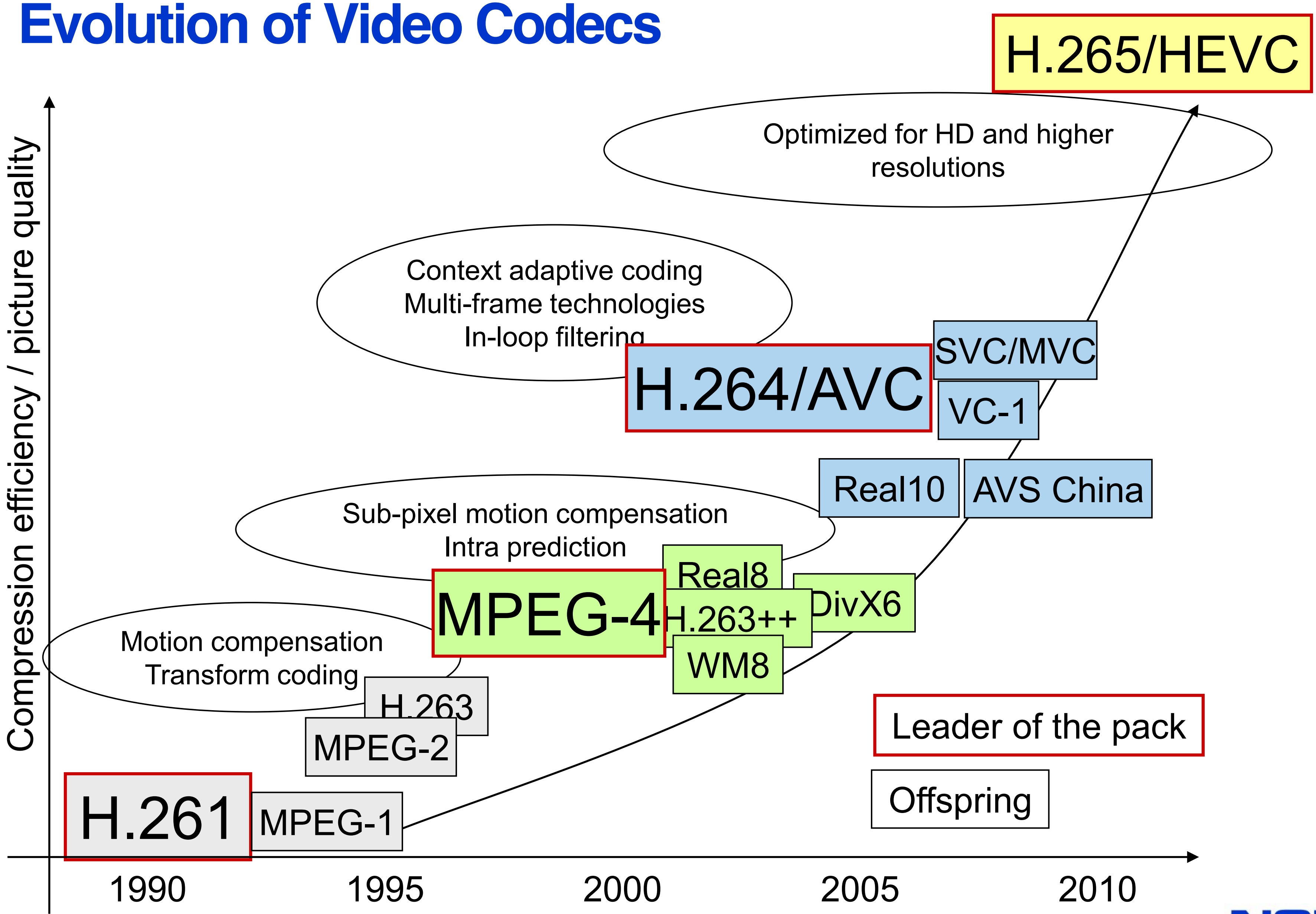
[...]

“In accordance with this invention, this difficulty is avoided by transmitting only the difference between successive images of the object.

Source : Bernd Girod, ICIP 2003 keynote:

http://www.stanford.edu/~bgirod/Talks.html

Evolution of Video Codecs



Applications of H.264/AVC

- Mobile Multimedia
  - Mandatory or recommended video coding standard in all 3GPP and DVB-H services
  - iPod Video
- High Definition Video
  - Used in Blu-Ray, High-def DVB Broadcasing, Blu-ray 3D
- Internet Video
  - YouTube uses H.264/AVC, HTML5 (some browsers support), Hulu, and many others

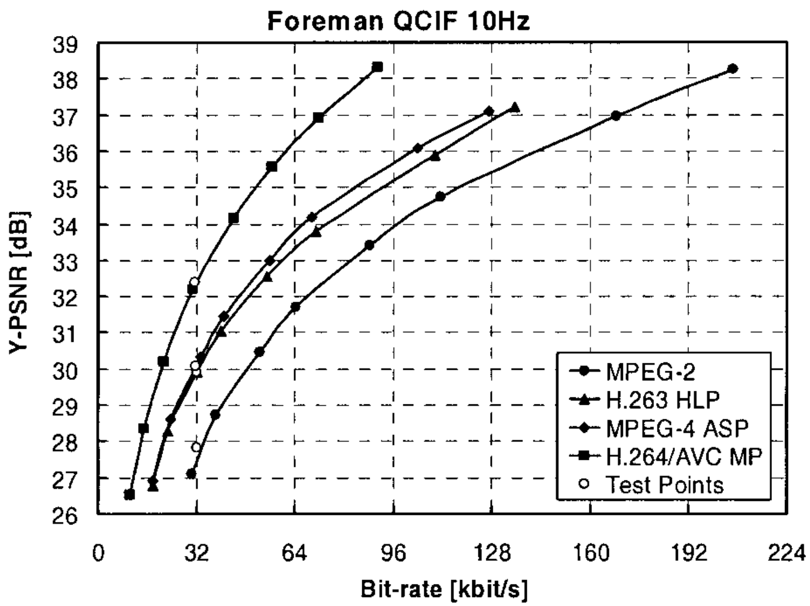
- JVT was awarded with two prime time Emmy awards!!

Emmy Award

JVT Chairs: Prof. Jens R. Ohm, Dr. Gary Sullivan, Prof. Thomas Wiegand, Dr. Ajay Luthra

Rate-Distortion Optimization

- Video coding is all about Rate Distortion Optimization
  - Achieve the highest quality video at lowest bitrate
- Distortion can be measured many ways, PSNR being the most common
  - PSNR is not perfect



Video examples (half the bitrate)

H.265/HEVC
2.9 Mbps / Full HD

H.264
6.0 Mbps / Full HD

Video examples (half the bitrate)

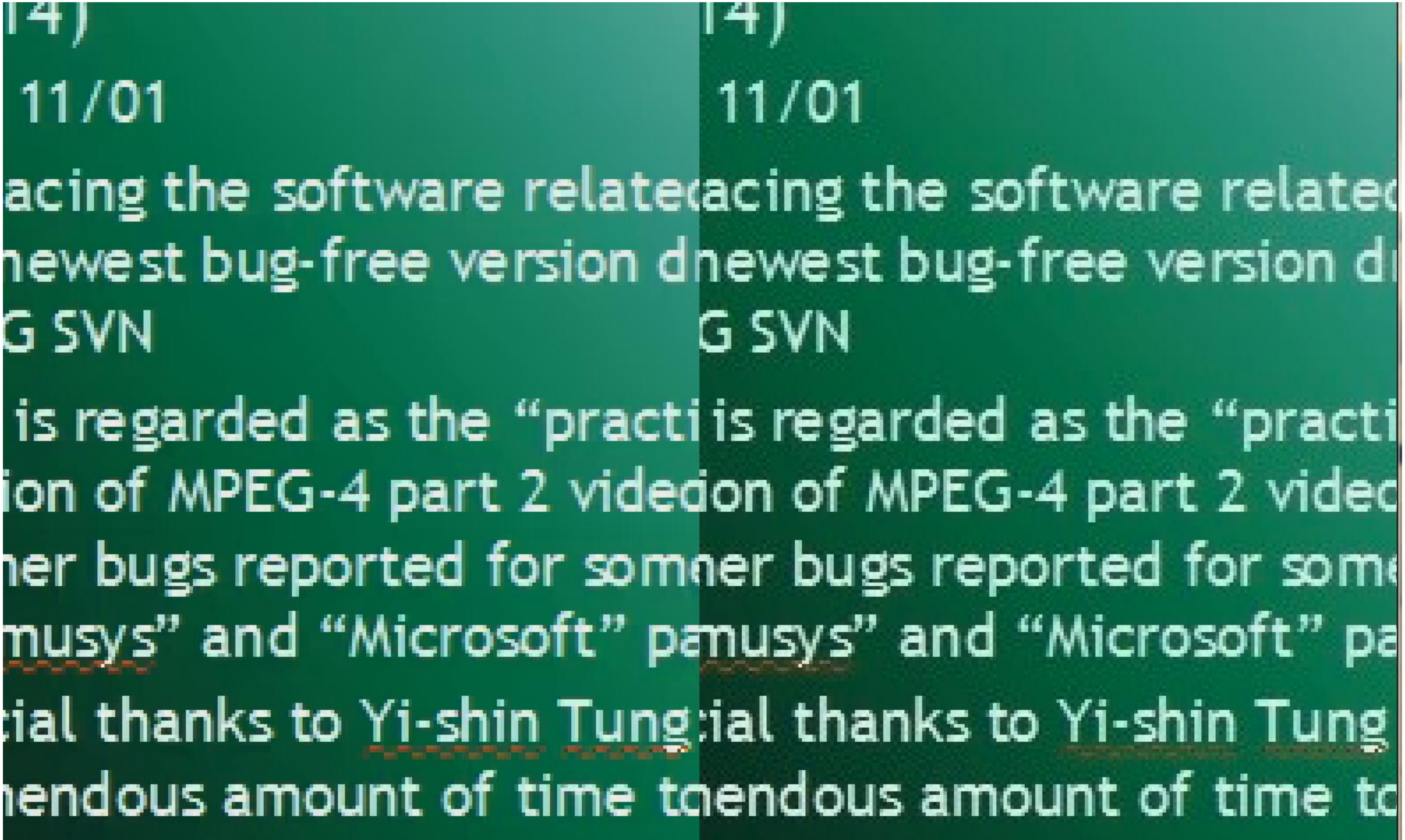
HEVC
0.95 Mbps / WVGA

H.264
2.0 Mbps / WVGA

Still image examples (equal file size)

HEVC	JPEG-XR	JPEG
131 kbits (0.14 bpp)	133 kbits (0.14 bpp)	137 kbits (0.15 bpp)

Still image examples (half the file size)



HEVC
1.13 bpp

JPEG
2.26 bpp **2X bitrate**

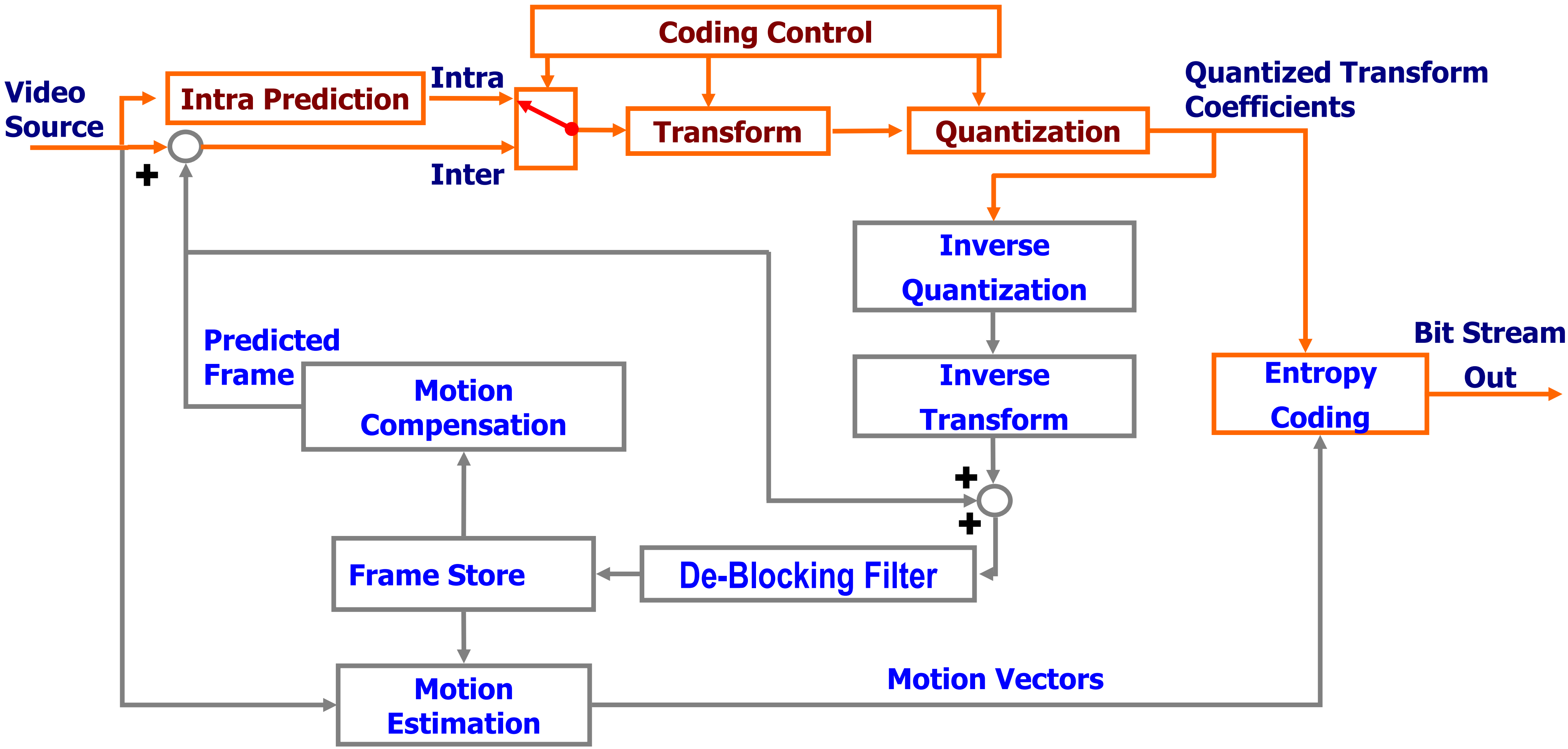
Basics of Video Compression

- The technical tools within the video coder could be roughly separated into two parts:

1. **Prediction**
   - Obtain the best possible prediction for the current signal so that the amounf of information that needs to be coded is minimized

2. **Coding the Residual**
   - Code the residual using as few bits as possible

H.264/AVC Building Blocks



NOKIA
Connecting People

Image and Video compression: Definitions

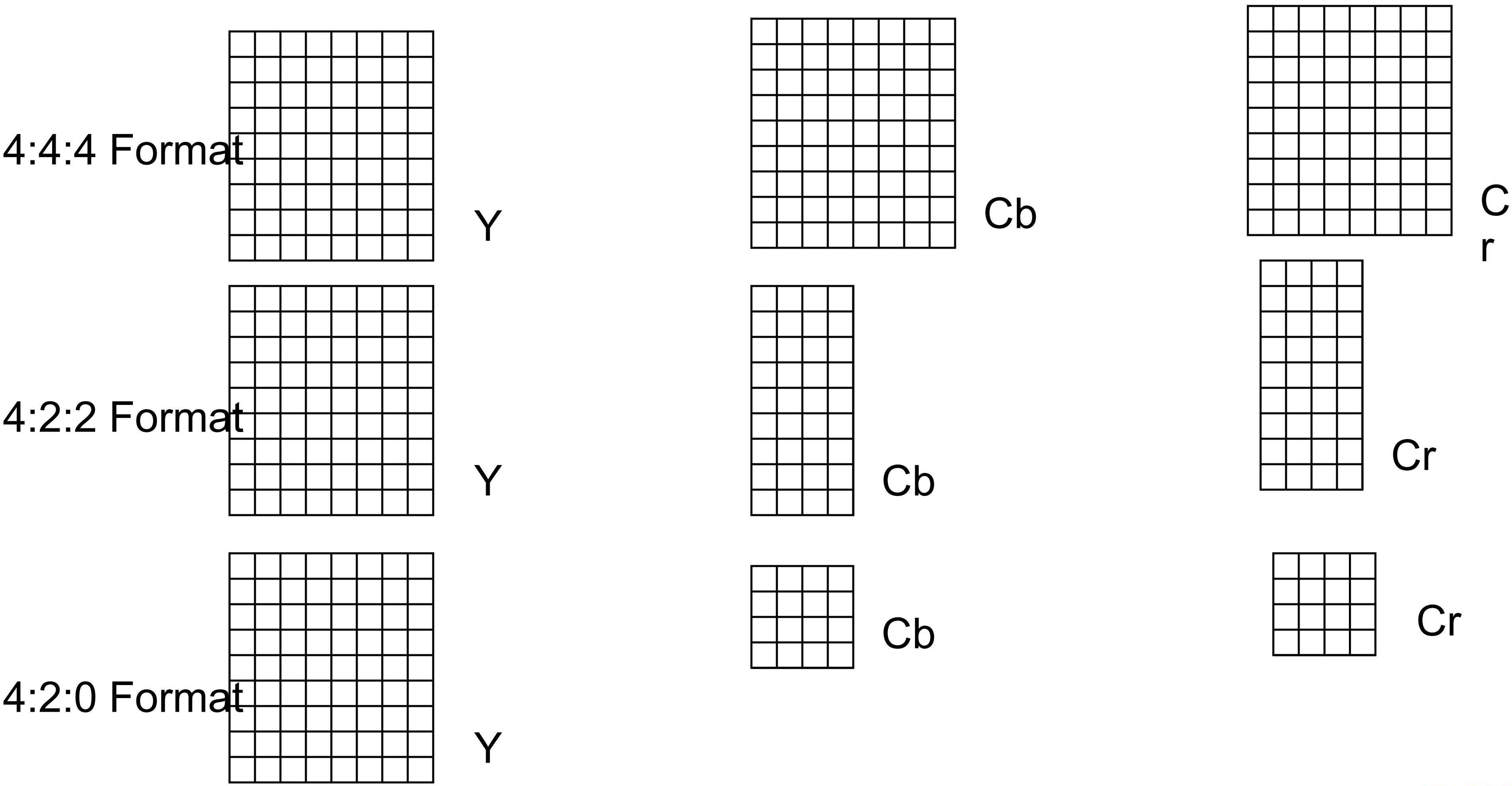
- Compression (also sometimes referred as coding) aims to reduce amount of the data needed to represent an object.
- This can be done in two ways:
  - Without changing the quality of the data -> Lossles compression
  - Removing part of the information -> Lossy compression.
  - Lossless compression not relevant for motion video unless transmission data rates and storage space easily allow it
- The amount of compression achievable depends very significantly on the content of the video material
  - Easily a factor of 10 difference between “easy” and “difficult” sequence

Before Compression

- Get rid of the part of the data:
  - Reduce resolution if possible (downscaling)
  - Colors are correlated, remove correlation by converting RGB values to diffrerent colour space like Y Cb Cr.
  - Downscale color part (Cb and Cr components) human visual system is not sensitive to this !

Downscaling color

- Subsampling:

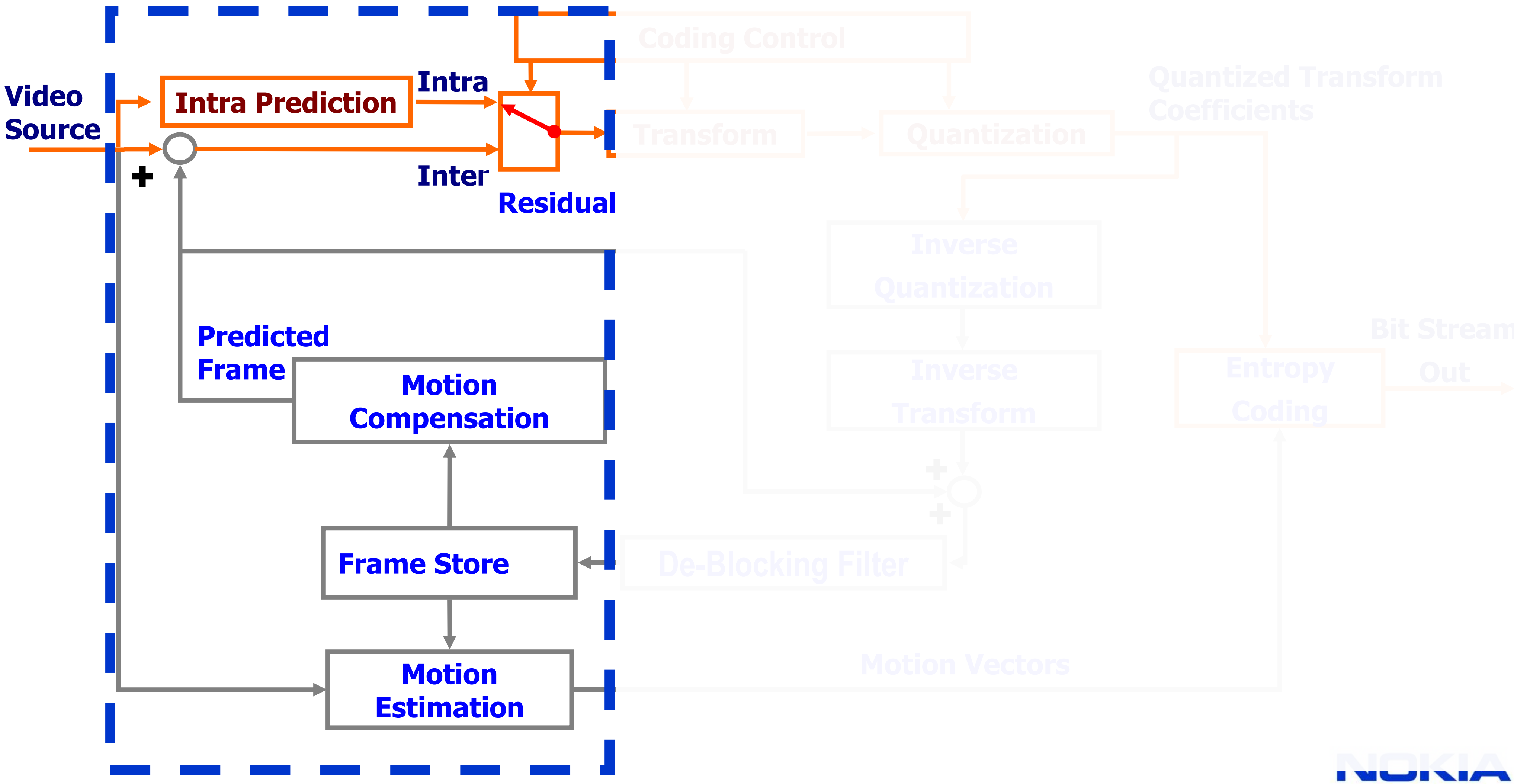


Video Compression Basics

Part I : Prediction

NOKIA
Connecting People

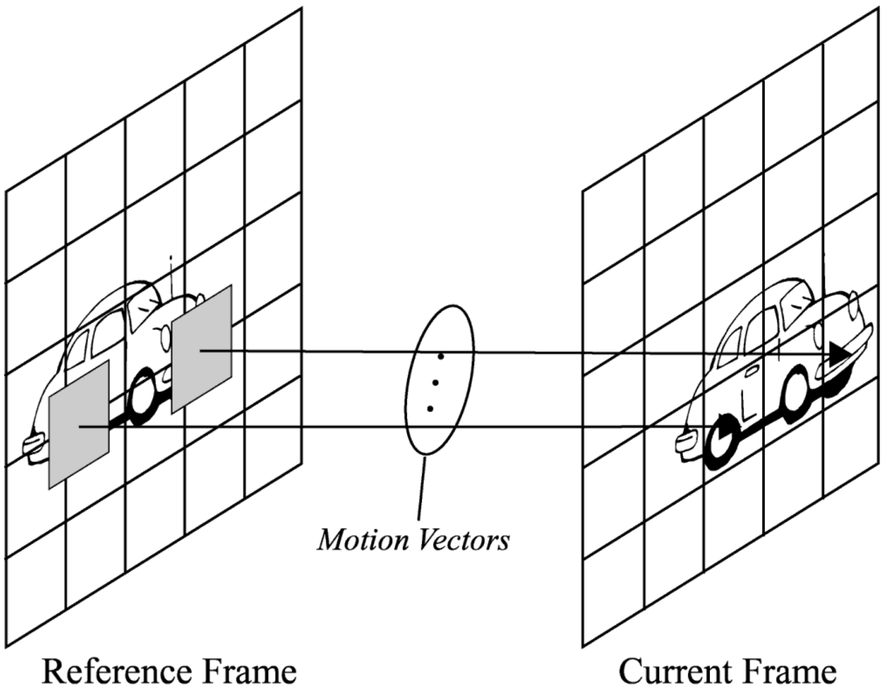
H.264/AVC Building Blocks



Why Predict?

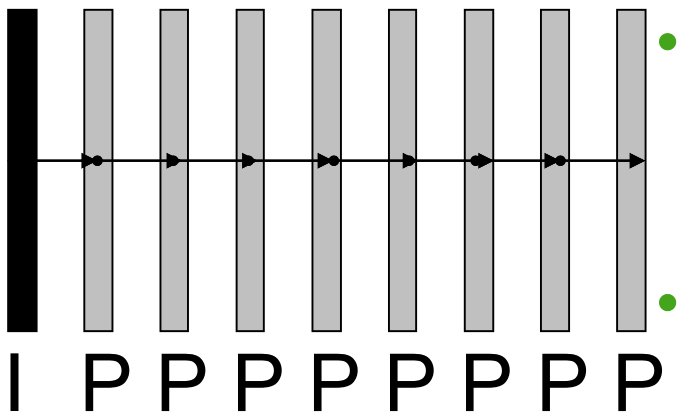
- Prediction is used to remove redundancy in video sequence.
  - Intra prediction : To remove spatial redundancy
  - Inter prediction : To remove temporal redundancy
- The residual signal has less entropy than the original signal
  - Can be compressed more efficiently

Predictive Coding

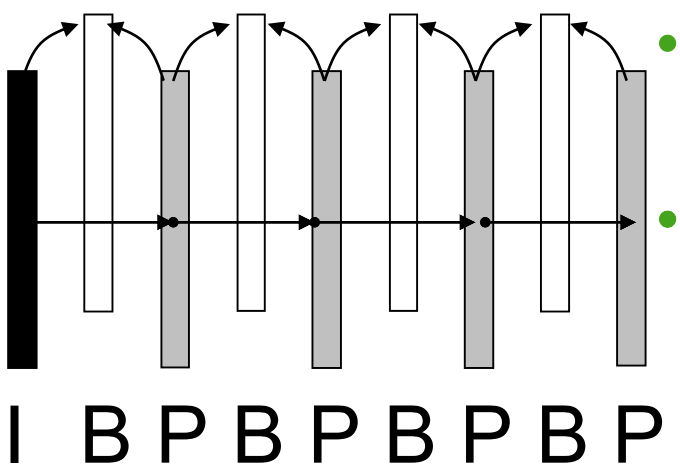


- Current frame is divided into macroblocks (16x16 in H.264/AVC)
- For each macroblock a prediction signal is found
- Encoder transmits the following:
  - Difference between original and prediction signal (residual)
  - Information to generate the prediction at the decoder side (e.g. Motion Vectors)

Motion Compensated Prediction



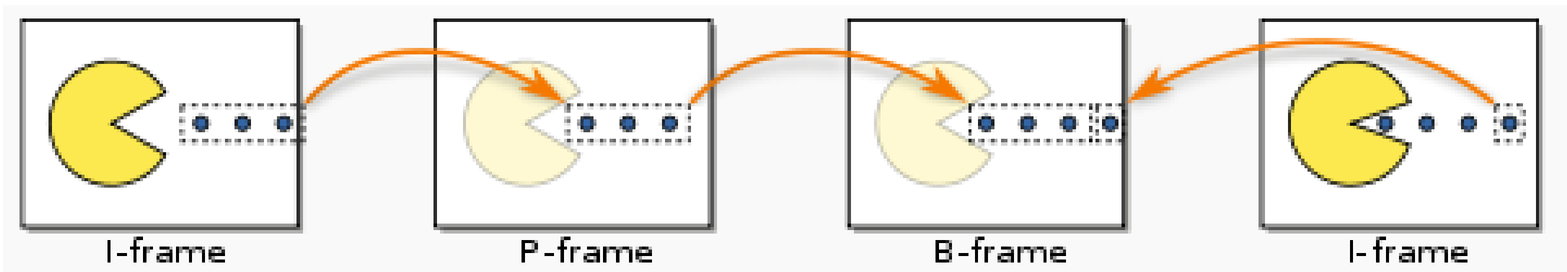
- IPP coding structure
  - I : Intra
  - P : Predictive
- Suitable for low delay



- IBP coding structure
  - B: Bidirectional
- Higher coding efficiency
  - Higher delay

NOKIA
Connecting People

Motion Compensated Prediction

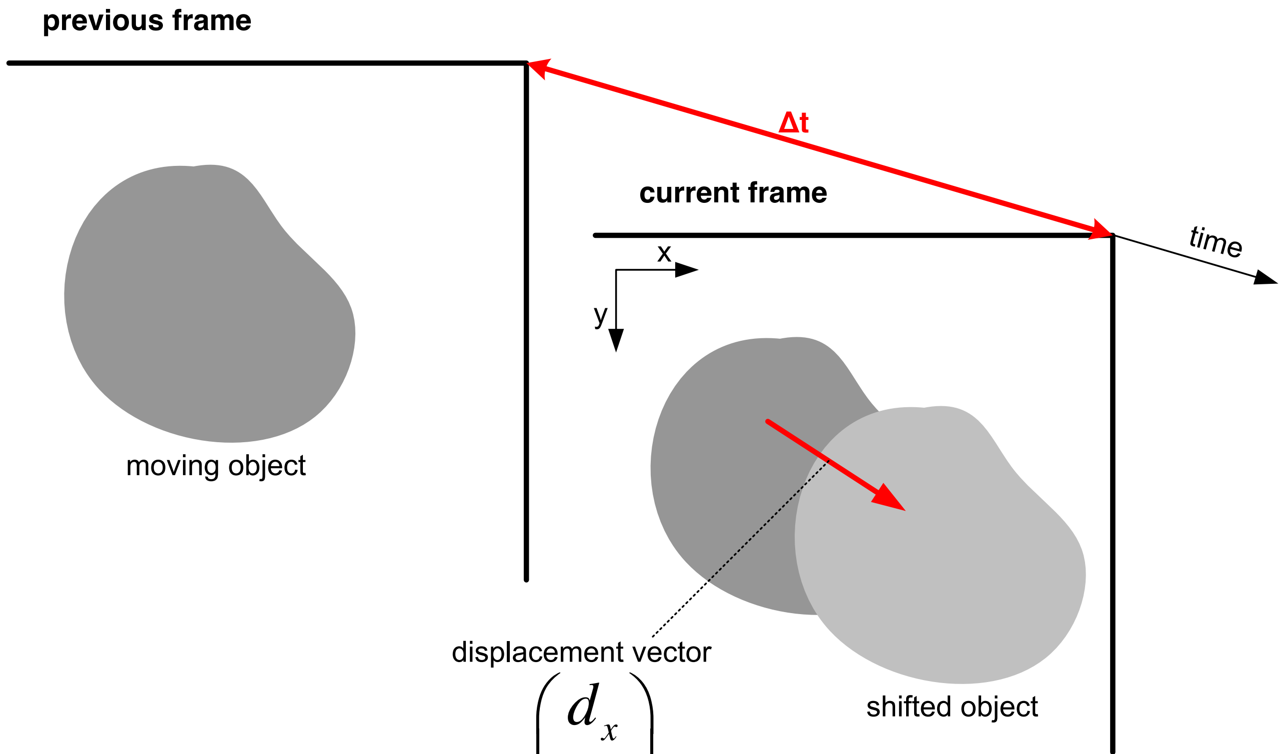


Source: wikipedia

- Increased coding efficiency for bi-directional prediction

NOKIA
Connecting People

Motion Compensated Prediction

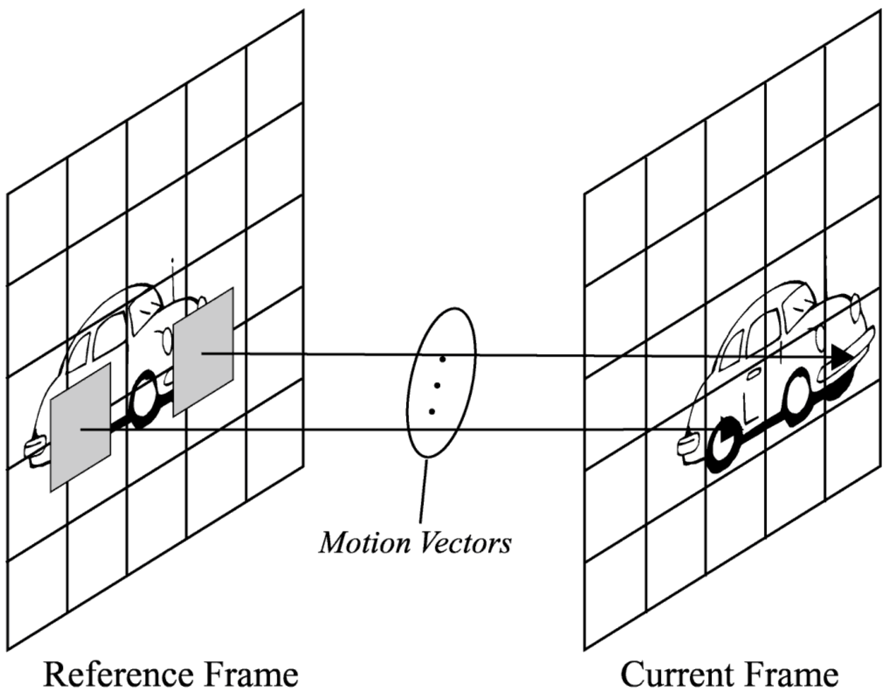


$$\begin{pmatrix} d_x \\ d_y \end{pmatrix}$$

Prediction for a luminance point (x,y,t) within the moving object:

$$\boxed{S(x,y,t) = (S - d_x, y - d_y, t - \Delta t) + \text{Residual}}$$

Basic Motion Model



- Translational Motion Vectors
  - Motion vector has x and y displacement in integer pixels
- One motion vector per macroblock
- One reference frame

Some ideas to improve the prediction:

NOKIA
Connecting People

Flexible Reference Frames in H.264/AVC

- H.264/AVC allows flexibility in multiple reference frame concept
  - Coding and display order is completely decoupled
  - Any picture can be marked as reference, independent of coding types
  - More than picture could be used in motion compensation

Flexible Reference Frames in H.264/AVC

- Generalized B pictures

More than one prediction signal.

These prediction signals could be arbitrarily weighted

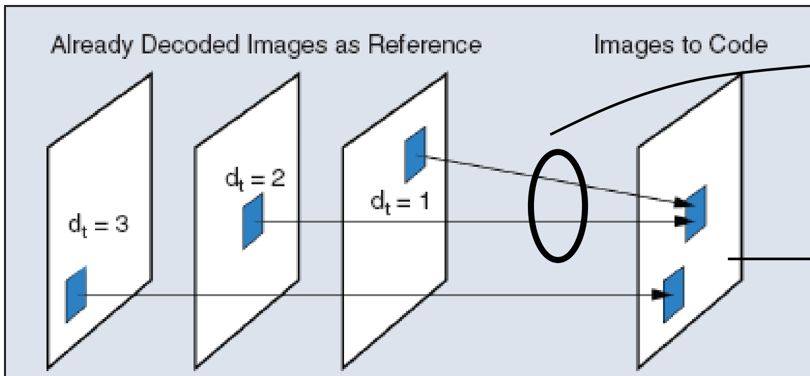


Figure 9. Motion-compensated prediction with multiple reference images. In addition to the motion vector, also an image reference parameter d_t is transmitted.

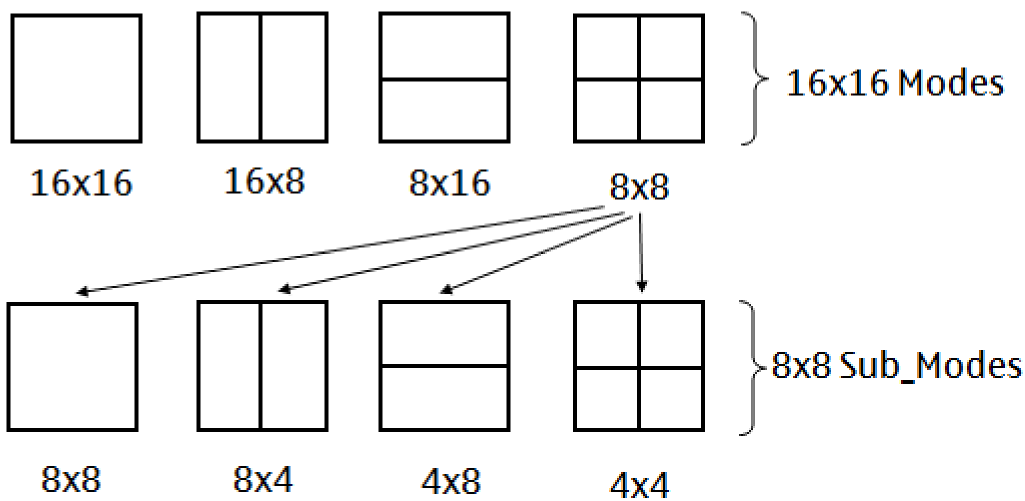
Three reference frames are used

(could be more)

- Can jointly exploit scene-cuts, aliasing, uncovered background, fades

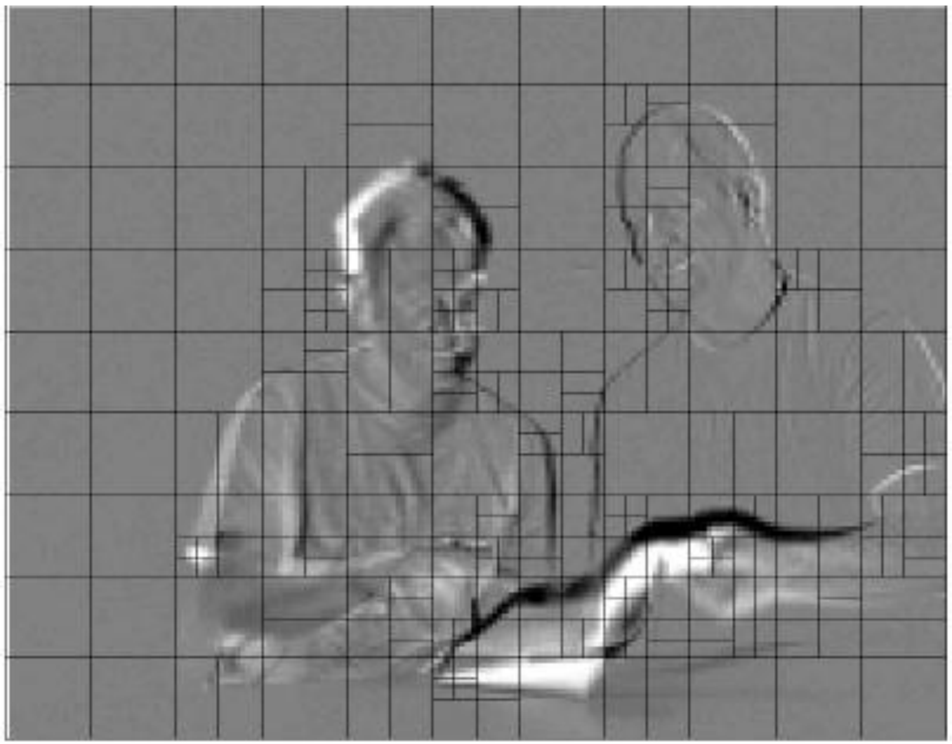
Variable Block-Size Motion Compensation

- Some areas in an image might under-go complex motion, whereas other areas are simpler
  - Use smaller block sizes for areas with complex motion



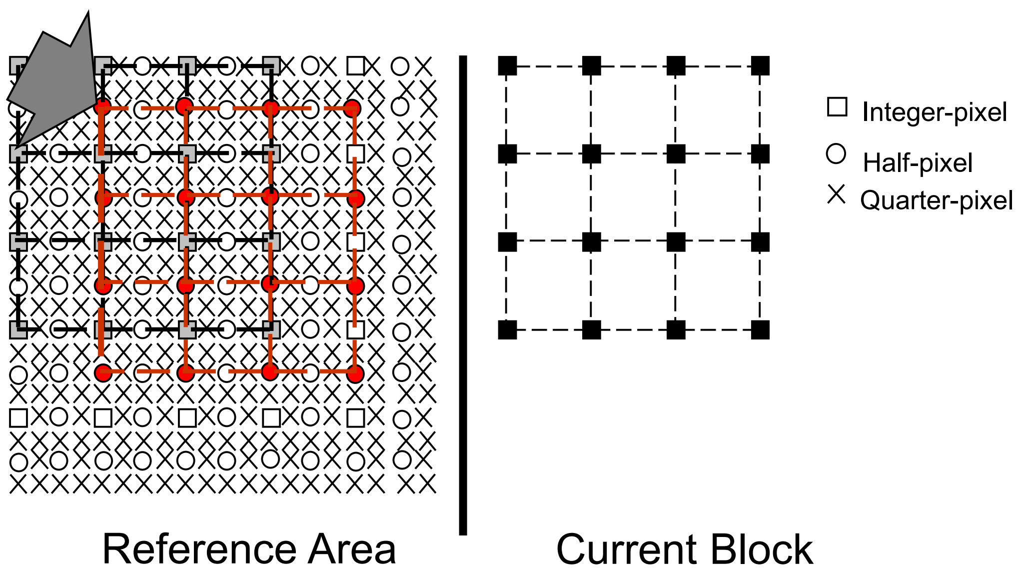
Variable Block-Size Motion Compensation

- An example



Sub-Pixel Accuracy Motion Compensation

- The natural motion does not fit to integer pixel motion vectors
- Increasing the accuracy of motion vectors improve the prediction quality significantly
- H.264/AVC supports motion vectors with ¼ pixel accuracy



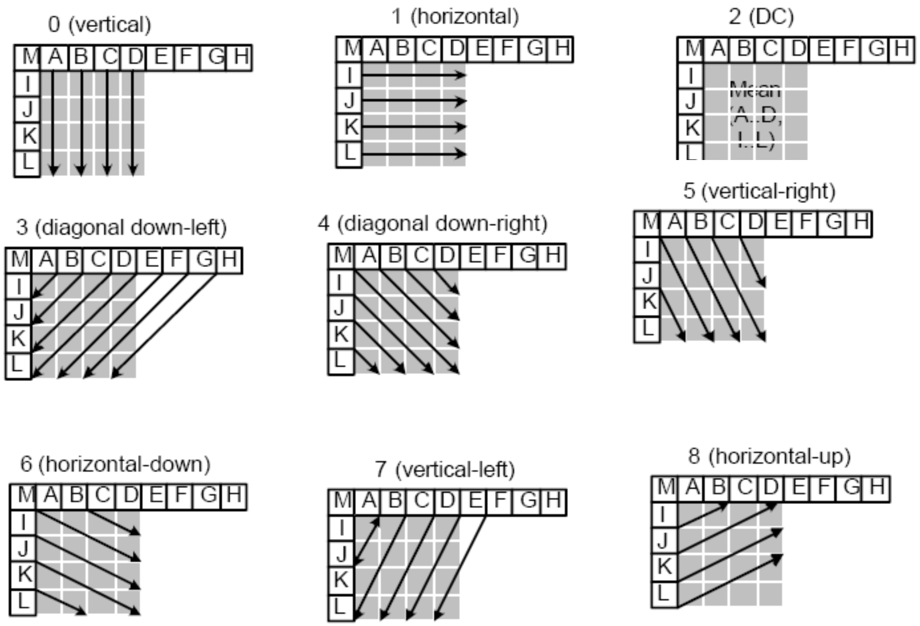
NOKIA
Connecting People

Block Matching

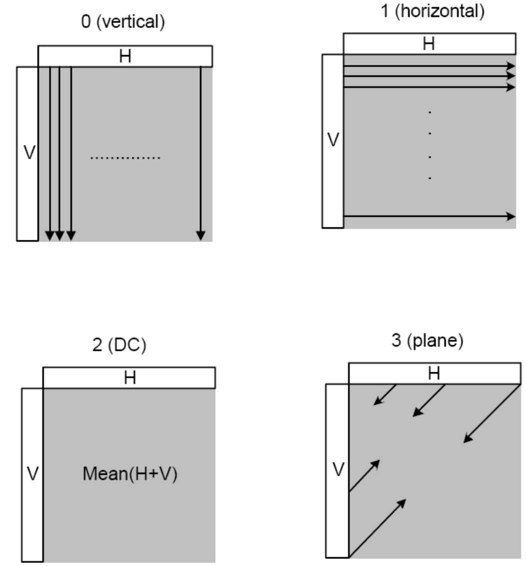
Camera movement

- optimal motion vector?
    - investigate all positions within a search window
    - keep the one with minimum cost
    - motion vector = corresponding translation

INTRA Prediction



INTRA_4x4 Prediction Modes



Figures from www.vcodex.com

INTRA_16x16 Prediction Modes

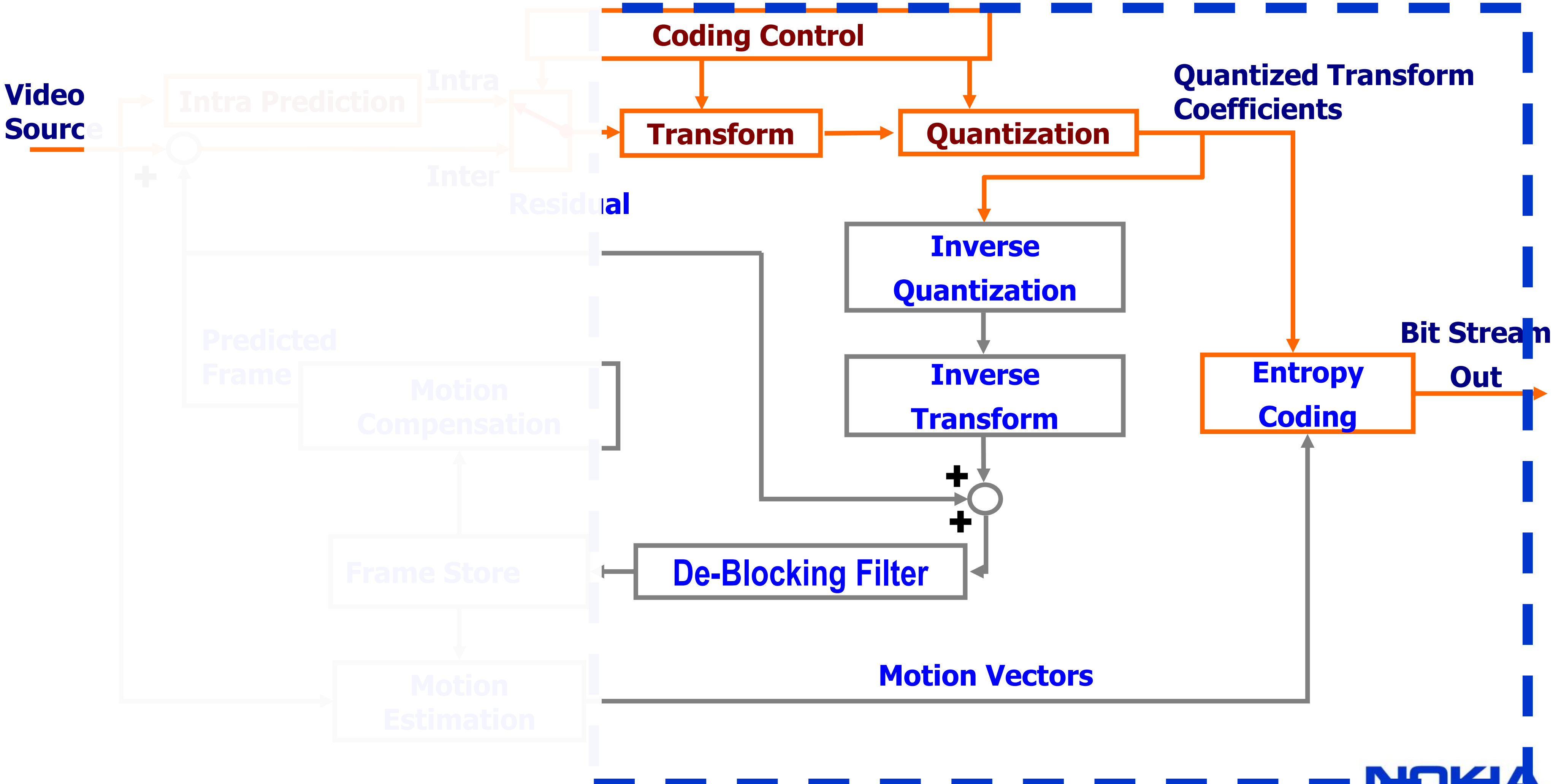
- A macroblock could be INTRA predicted either at a 4x4 (INTRA_4x4) or 16x16 (INTRA_16x16) block level
  - 9 prediction directions for INTRA_4x4
  - 4 prediction directions for INTRA_16x16

NOKIA
Connecting People

Video Compression Basics

Part II : Coding Residual

H.264/AVC Building Blocks



Connecting People

Transform Coding

- Transform coding is used to efficiently code the residual signal. It consists of three steps:
  - Linear transformation of signal so that it is more easily compressible.

    Decorrelates the residual signal
  - Quantization of coefficients

    Reduces the accuracy of coefficients (lossy)
  - Entropy coding of the quantized coefficients

    Encoding the quantized transform coefficients (using smallest number of bits possible)

- Note: Transform coding deserves a separate lecture itself. Here only the basics will be overviewed

NOKIA
Connecting People

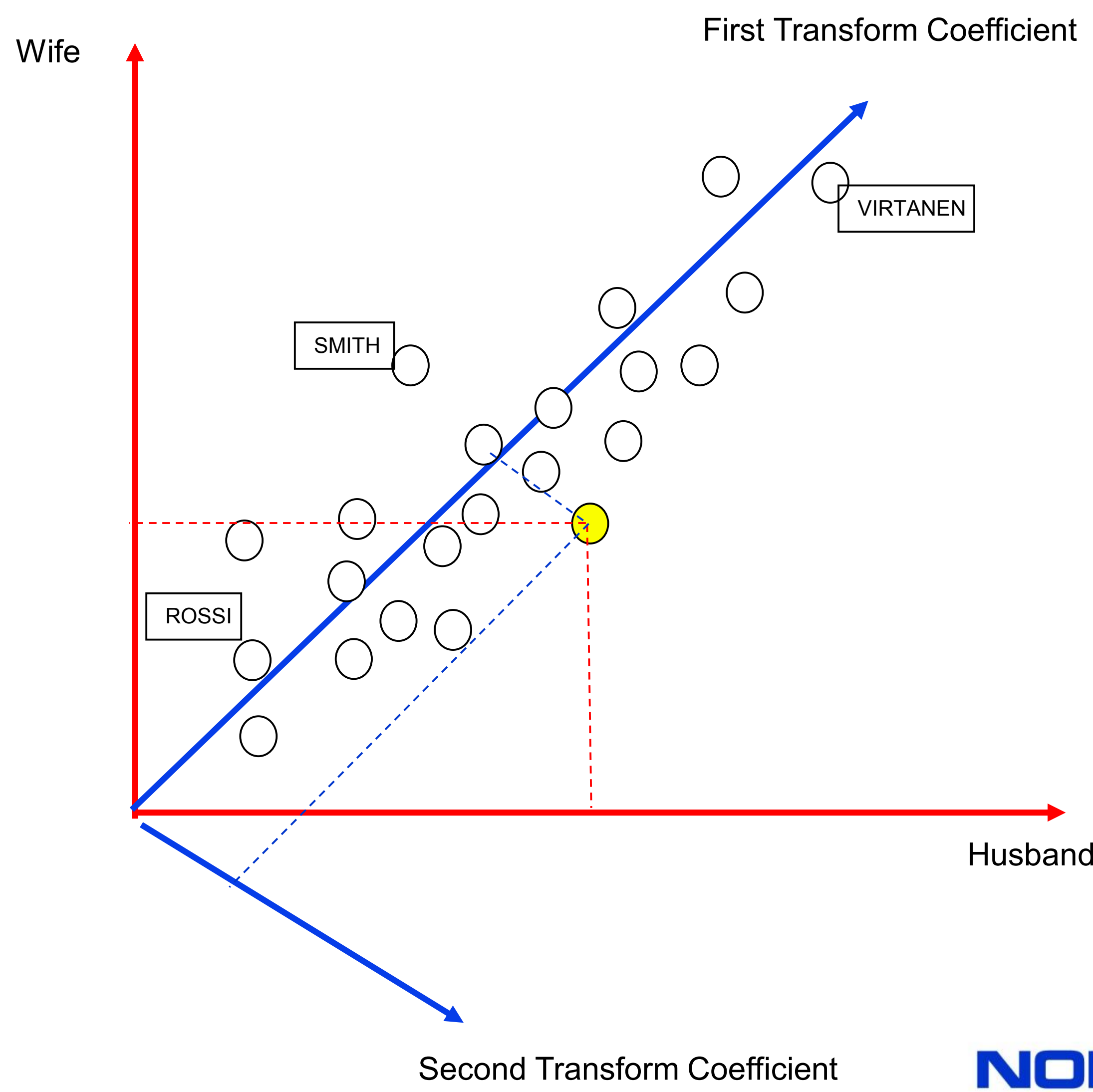
Transform Coding

- Residual signal of prior to transform is still correlated

- Goal of transform:
  - Pack the energy of the signal to as few coefficients as possible
  - Energy Compaction property

Transform Coding example

- Example:
  - Population is divided in pairs
  - Coordinate axes are rotated to alter the distribution of height values
  - Most of the information is in the first transform coefficient
  - The Second transform coefficient can be coded very efficiently



NOKIA
Connecting People

Spatial redundancy reduction

- The most common way of reducing spatial redundancy in an image is to operate a transformation from the grey level domain in a frequency domain.
- The goal is to reorganize the informative content of the picture, so that the information is concentrated in a limited number of coefficients.
- Several transformations have been studied, with different properties
- The most commonly used transform used in image and video processing is the Discrete Cosine Transform.

Discrete Cosine Transform

- The signal s(n) is expressed as a linear combination of cosinusoids (base functions) with different frequency.
- The coefficients of the linear combination (= "*how much*" of each base functions we are using) are called DCT coefficients S(k).

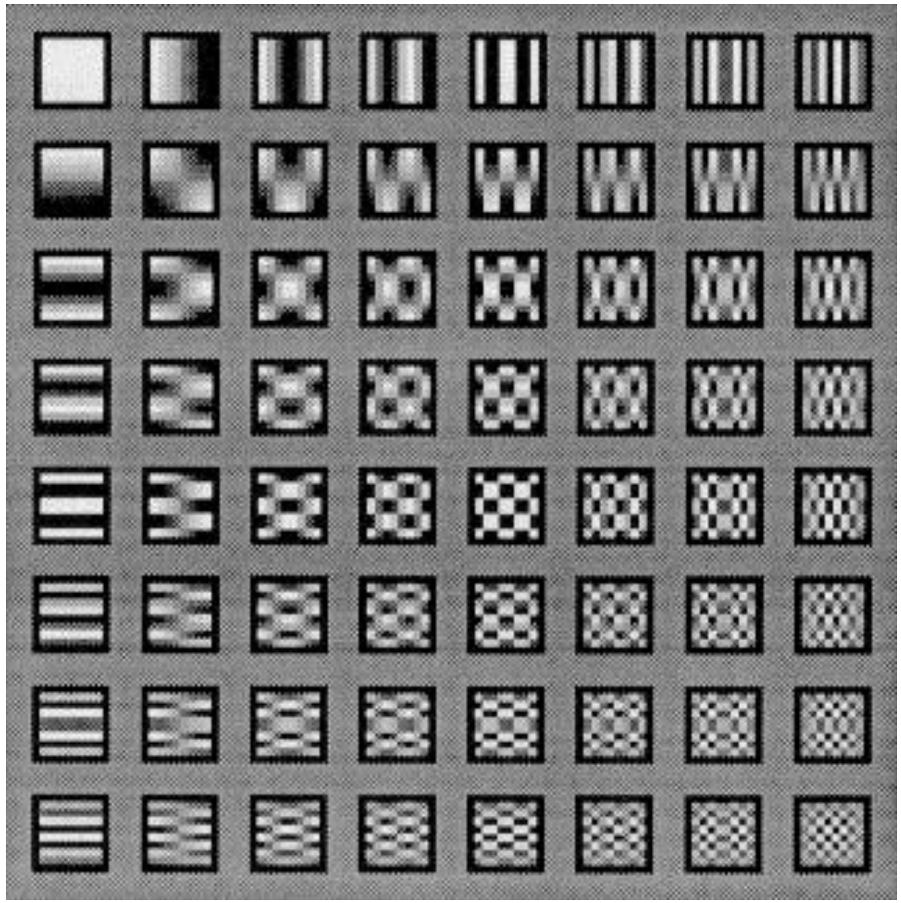
$$\underbrace{s(n)}_{\text{Signal}} = \sum_{k=0}^{N-1} c(k)\underbrace{S(k)}_{\text{DCT coefficients}} \cdot \underbrace{\cos\left(\frac{2n+1}{2N}\cdot k\pi\right)}_{\text{Base functions}}$$

$$0 \le n \le N-1$$

$$c(k) = \sqrt{\frac{1}{N}}, k = 0$$

$$c(k) = \sqrt{\frac{2}{N}}, 1 \le k \le N-1$$

2D Discrete Cosine Transform base functions

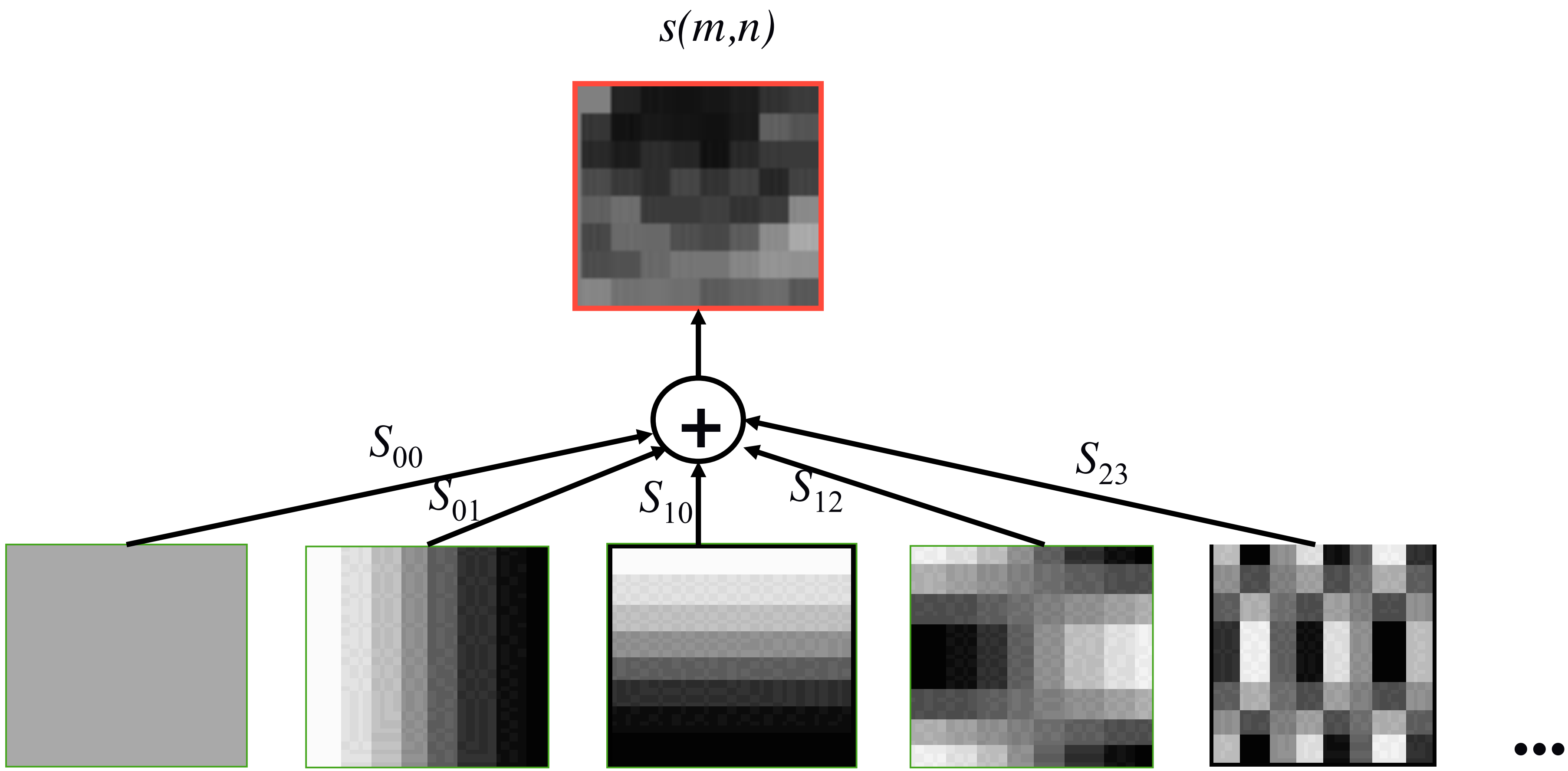


Express a 8x8 block of pixels

Input image

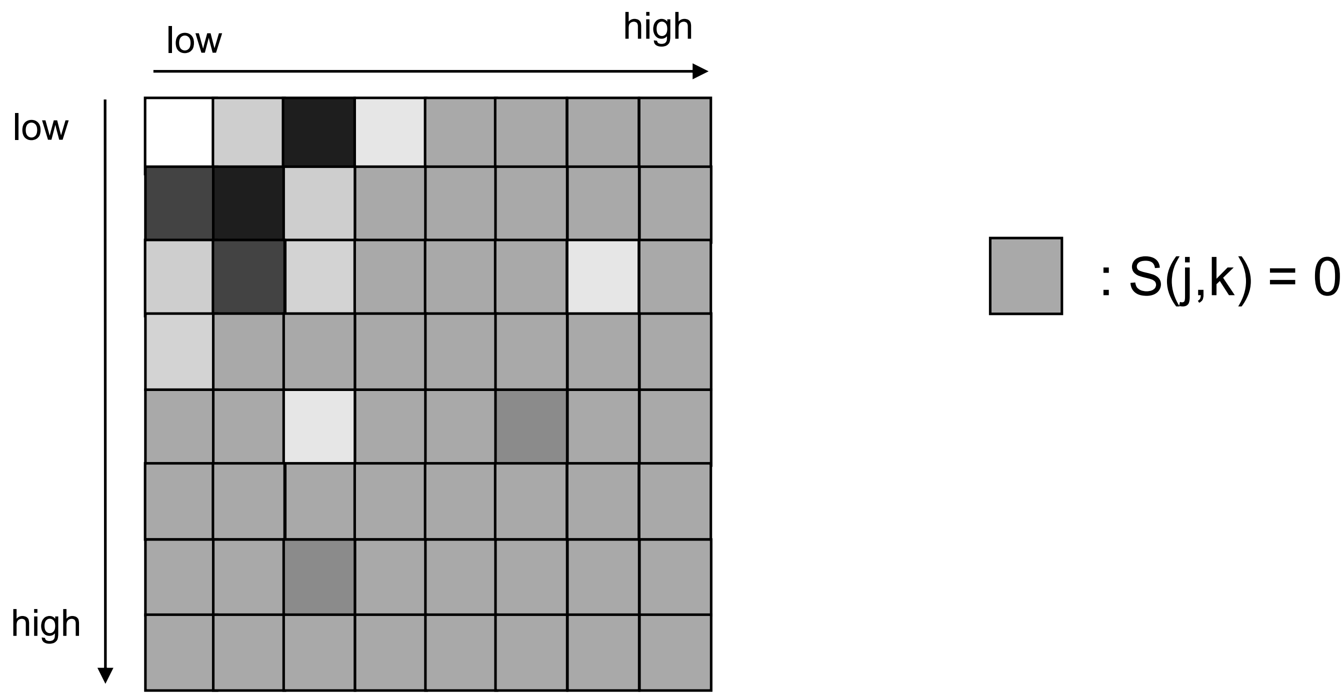
8x8 block

..as a linear combination of 64 2D base functions



What does S(j,k) look like?

- Most of the “energy” of the picture is concentrated in a few coefficients, close to the low-low frequencies



Spatial Redundancy Reduction

- Transform from the sample value domain to coefficients to pack the energy
- Conventional coding standards use Discrete Cosine Transform (DCT)
- Image quality degradation: quantization of transform coefficients
- Used to code the difference between a source picture and a motion-compensation prediction picture too

Original block

76	80	77	82	82	82	77	60
72	72	84	76	57	57	62	48
73	62	54	55	63	61	58	71
52	46	54	58	56	61	59	58
57	52	64	71	58	53	58	60
78	70	59	59	67	64	64	65
48	53	59	64	71	71	60	73
53	56	52	64	62	54	55	59

Block after DCT

504	2	-8	0	7	3	0	-1
28	20	-1	2	-10	9	0	1
25	6	-11	1	-6	-3	-5	-1
30	6	-5	3	-5	1	-2	0
-7	-6	-12	-1	0	6	-4	-2
5	-23	-1	11	0	-10	3	-1
9	3	16	15	6	-8	-5	0
3	-9	-1	-3	6	-4	1	-1

Spatial Redundancy Reduction
- DCT and Quantization

Original block

76	80	77	82	82	82	77	60
72	72	84	76	57	57	62	48
73	62	54	55	63	61	58	71
52	46	54	58	56	61	59	58
57	52	64	71	58	53	58	60
78	70	59	59	67	64	64	65
48	53	59	64	71	71	60	73
53	56	52	64	62	54	55	59

Block after DCT

504	2	-8	0	7	3	0	-1
28	20	-1	2	-10	9	0	1
25	6	-11	1	-6	-3	-5	-1
30	6	-5	3	-5	1	-2	0
-7	-6	-12	-1	0	6	-4	-2
5	-23	-1	11	0	-10	3	-1
9	3	16	15	6	-8	-5	0
3	-9	-1	-3	6	-4	1	-1

Block after quantization

50							
2	**2**			**-1**			
2		**-1**					
3							
		-1					
	-2		**1**		**-1**		
		1	**1**				

Sparse matrix: lots of zeros!

Transform Quantization

- H.264/AVC Transform
  - 4x4 and 8x8 Integer Transform (close approximation of DCT)
    - Integer transform does not suffer from mismatches due to floating point calculations
  - 8x8 is more efficient for high resolution sequences
  - Using smaller transform reduces ringing artefacts
  - Integer transform means, no error accumulation due to floating point arithmetic

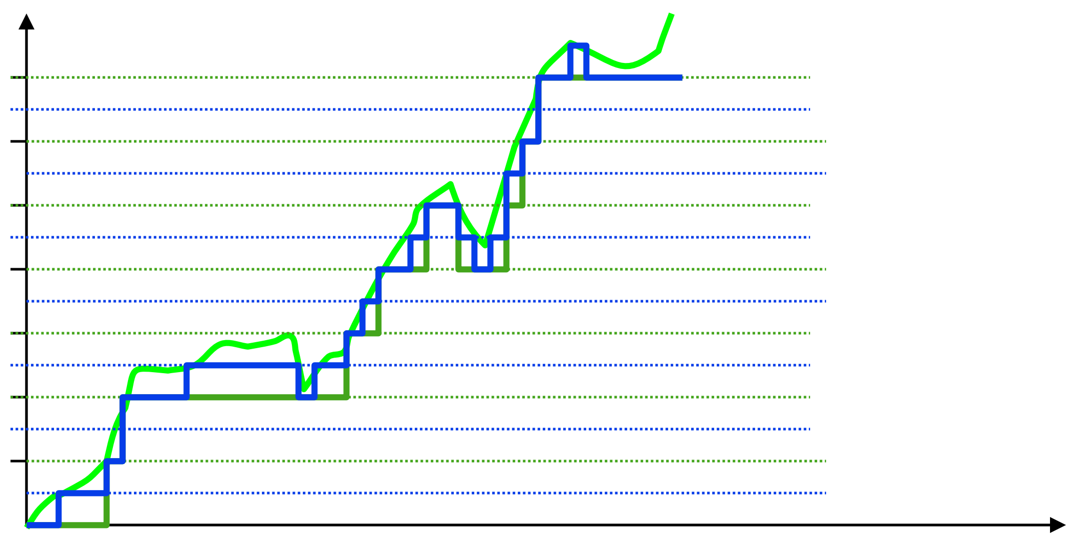
Quantization

Transform coefficients are quantized before coding them in bitstream

- Main reason for information loss in video compression

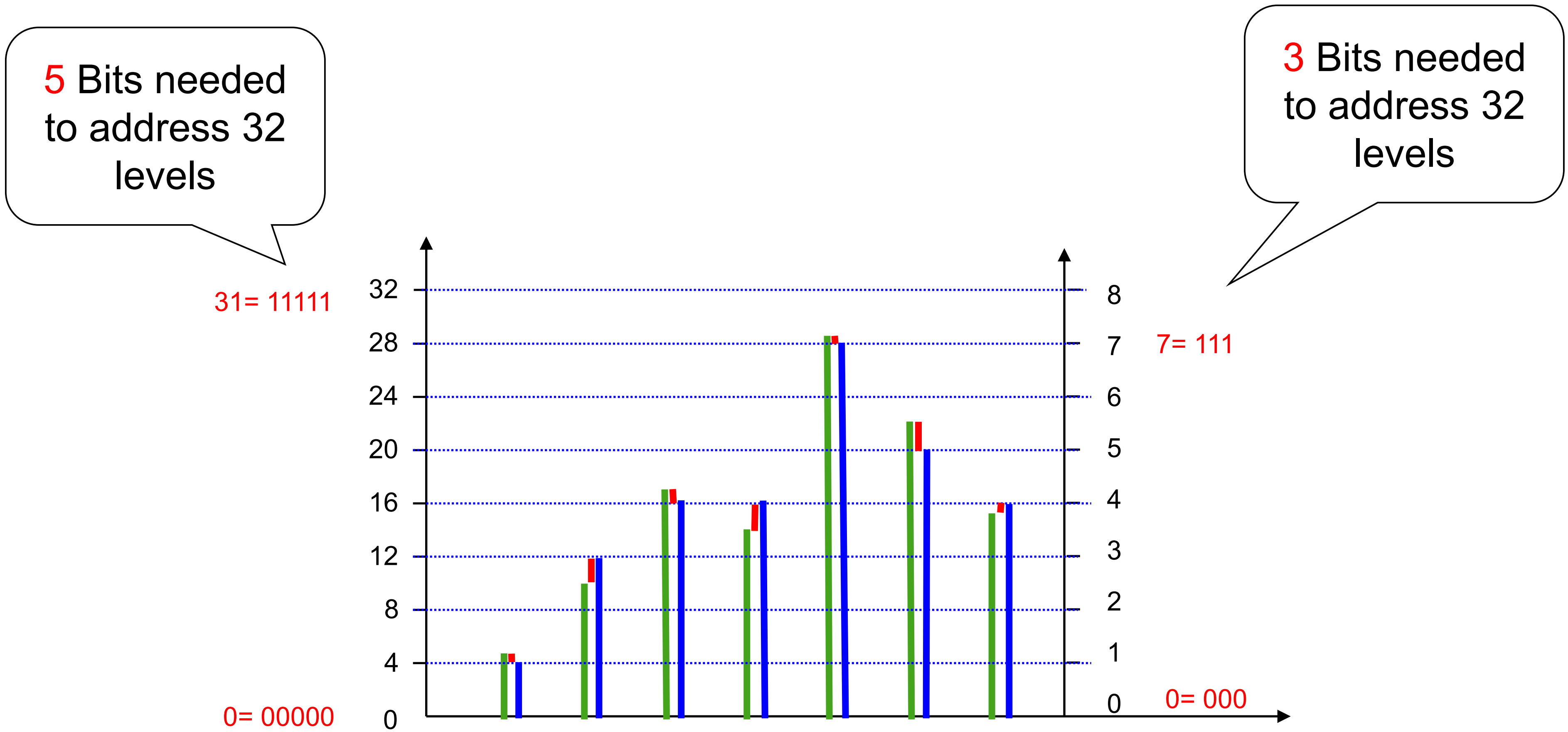
$$\text{Quantized value} = \text{Round}\left(\frac{\text{Coefficient}}{\text{Quantum Value}}\right)$$

Coefficient = Quantized value x Quantum value



Connecting People

Quantization



NOKIA Connecting People

Transform / Quantization

$$\begin{bmatrix} 100 & 104 & 103 & 102 \\ 99 & 103 & 101 & 100 \\ 100 & 102 & 102 & 101 \\ 98 & 99 & 97 & 102 \end{bmatrix} \xrightarrow{\text{Transform}} \begin{bmatrix} 403.25 & -1.94 & -2.25 & -2.72 \\ 3.98 & 0.82 & -3.21 & 0.43 \\ -0.75 & -1.17 & 1.75 & -0.87 \\ 2.41 & -0.07 & -0.56 & 1.18 \end{bmatrix}$$

Quantize

$$\begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Inverse Quantize

$$\begin{bmatrix} 100 & 100 & 100 & 100 \\ 100 & 100 & 100 & 100 \\ 100 & 100 & 100 & 100 \\ 100 & 100 & 100 & 100 \end{bmatrix} \xleftarrow{\text{Inverse Transform}} \begin{bmatrix} 400 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Entropy Coding

- Losslessly code all the information and put it in the bitstream:
    - Transform coefficients
    - Motion Vectors
    - Mode information
    - Etc.

- In other words, convert the data to 1's and 0's

Entropy Coding

- Example macroblock bitstream structure:

```
mb_header() {
  mb_mode;
  if(mb_mode == INTER)
      mvd_x;
      mvd_y;
  }
  cbp;
  for (each coded block)
      transform_coeffs();
}
```

Motion vector difference.
Motion vectors are also predictive coded

Coded block pattern
Indicates if the transform blocks have all-zero coefficients

Entropy Coding

- H.264/AVC supports two types of entropy coding:
  - Context Adaptive Variable Length Coding (CAVLC)
  - Context Adaptive Binary Arithmetic Coding (CABAC)

- CABAC uses binary arithmetic codes, whereas CAVLC uses VLC tables
  - CABAC is more efficient, but also more complex

- Biggest improvement comes from context adaptivity:
  - Earlier, specifically tailored but fixed variable length codes (VLCs) were used.
  - Problem?: Assumes statistics are stationary, which is not the case

Context Adaptivity

- Suppose we want to code the motion partition of macroblock.

Partition	Probability (p)	Code
16x16	0.25	00
16x8	0.25	01
8x16	0.25	10
8x8	0.25	11

Partition	Probability (p)	Code
16x16	0.50	0
16x8	0.25	10
8x16	0.125	110
8x8	0.125	111

- VLC codes depend on the statistics
  - If statistics change, loss of compression efficiency
- Solution: Adapt the entropy coding to context

Video Quality

Video quality

- Defined by
  - Compression method
  - Compression parameters (resolution, framerate, bitrate)

- Affected by
  - Encoder implementation (processing power limitations)
  - Decoder implementation (error resilience)

  - Pre- and post processing methods
    - Noise reduction
    - Video stabilisation
    - Image enhancement.
    - Etc.

  - Camera performance / Imaging conditions
  - Display properties
  - Observer properties

NOKIA
Connecting People

Some Sources of Degradation in Video Quality

Optics, sensor	Imperfect optics, dead pixels in sensor, …
Image capturing	Wrongly tuned exposure, white balance, …
Image processing	Loss of detail and aliasing in downscaling, ...
Video pre-processing	
Encoding	Coding error/artefacts
Transmission/Storage	Transmission errors and losses, transmission delay jitter
Decoding	
Video post-processing	
Image rendering	Tearing, jitter in display times
Display	Insufficient dynamic range, …

NOKIA
Connecting People

Video quality

- Video quality will be a key competition point for mobile multimedia.
- Nokia should ensure that not only the best codecs are chosen, but also that our implementations are the best.

  - How to measure the quality of the video signal reliably?

  - Use objective metrics:
    - Calculate a score using a formula and the compressed video signal and maybe the original video signal.

  - Use subjective metrics:
    - Show the compressed video signal (and maybe the original signal) to different viewers, and ask their opinion about their quality.

  - Subjective evaluation is the most important but time-consuming and difficult.

Objective Codec Testing – PSNR

- Mean Square Error:

$$MSE = \frac{1}{XYT}\sum_{t=1}^{T}\sum_{y=1}^{Y}\sum_{x=1}^{X}\left(i(x,y,t)-r(x,y,t)\right)^2, \; where$$

$$i(x,y,t) = original\ signal,\ r(x,y,t) = reconstructed\ signal$$

  - Does not generally match with subjective quality
    - Not sensitive to local impairments
  - But, provides consistent results if compared signals are affected by the same type of impairment
- Peak Signal-to-Noise Ratio

$$PSNR = 10\log_{10}\frac{255^2}{MSE}$$

  - Logarithmic scale (dB) correlates better with subjective quality
  - The higher the value of PSNR, the better the quality
  - Rule of thumb: 1 dB = clearly noticeable difference

NOKIA
Connecting People

Quality Factors of Coded Video

- Basic factors affecting coded video quality
  1. Coding scheme
  2. Bitrate
  3. Picture rate
  4. Picture size
  5. Sharpness – blockiness/blurriness of single pictures
  6. Error resilience strength
- One or more factors typically fixed by application
- What is the subjectively best operation point for the non-fixed factors? Examples:
  - What are the bitrate and picture size corresponding to VHS quality?
  - Given a certain bitrate and picture size, what is an optimal tradeoff between picture rate and still picture quality?
- Answers to many of these questions are unknown
- Optimal operation point depends on the content
- Problems for video encoders – we have no answers:
  - How to create algorithms from subjective results?
  - How to analyse source stream to select parameter values for bit rate control algorithm?

Why Does Encoder Implementation Affect Quality?

- Video coding standards specify the bitstream syntax and semantics and decoder operation
- Encoders have to produce ”legal” bitstreams
  - No other constraints for encoder implementations
- Bitstream conformance can be checked by Hypothetical Reference Decoder (HRD) / Video Buffering Verifier (VBV) of the coding standard
- Profiles and levels
  - Specify restrictions on bitstreams
  - May be used to indicate decoder capability
  - A profile-compliant encoder does not have to use all coding tools of a profile
- Some factors affecting the encoded picture quality
  - Used coding tools
  - Motion estimation
  - Bit rate control
  - Macroblock mode selection (intra / inter)

Video Quality versus Bitrate

- The quality that can be reached with a certain bitrate depends on content
  - Content with little motion and few details can be compressed more efficiently than content with high motion and many details
- Video CD (VCD) standard
  - MPEG-1 video coding at constant bitrate 1.152 Mbps
  - Picture size 352x288 (for PAL content) or 352x240 (for NTSC content)
  - Said to correspond to “VHS quality”
- DVD standard
  - MPEG-2 video coding at variable bitrate up to 9.8 Mbps
  - Typical bitrate from 3.5 to 5 Mbps
  - Picture size up to 720x576 (PAL) and 720x480 (NTSC)
  - Corresponds roughly to “TV quality”

Summary of Coding Artefacts

- Blurring
  (loss of detail)
- Blockiness
  (visibility of DCT and motion compensation structures)
- Ringing
  (repetitive lines parallel to the strong luma edge)
- Contouring
  (effect of course quantization in smooth areas)
- Shadow image
  (moving object leaves a “shadow” to uncovered background)
- Mosquito noise
  (temporal flicker of spatially colocated pixels)
- Jerky motion
  (effect of low frame rate and/or buffer overflow)
  - Particularly annoying when large or frequent changes in frame rate

- Post-processing of decoded pictures can be used to coding artefacts
  - E.g. deblocking filter

Coding Artefacts

Blurring
(loss of details)

Uncompressed QCIF
picture
scaled up

H.263 Baseline
43 kbps

Coding Artefacts

Blockiness
(visibility of DCT and motion compensation structures)

Uncompressed QCIF
picture
scaled up

H.263 Baseline
43 kbps

Coding Artefacts

Ringing
(repetitive lines parallel to the strong luma edge)

Contouring
(effect of course quantization in smooth areas)

Uncompressed QCIF picture
scaled up

H.263 Baseline
43 kbps

Connecting People

Coding Artefacts

Shadow image
(moving object leaves a “shadow” to uncovered background)

time = t_0

uncompressed QCIF

time = $t_0+\Delta t$

H.263 Baseline 43 kbps

NOKIA
Connecting People

Blocking

- Consider a smooth 1D surface of 8 pixels [1 2 3 4 5 6 7 8]

Samples	DCT
[1 2 3 4]	[5 -2.23 0 -0.16]
[5 6 7 8]	[13 -2.23 0 -0.16]

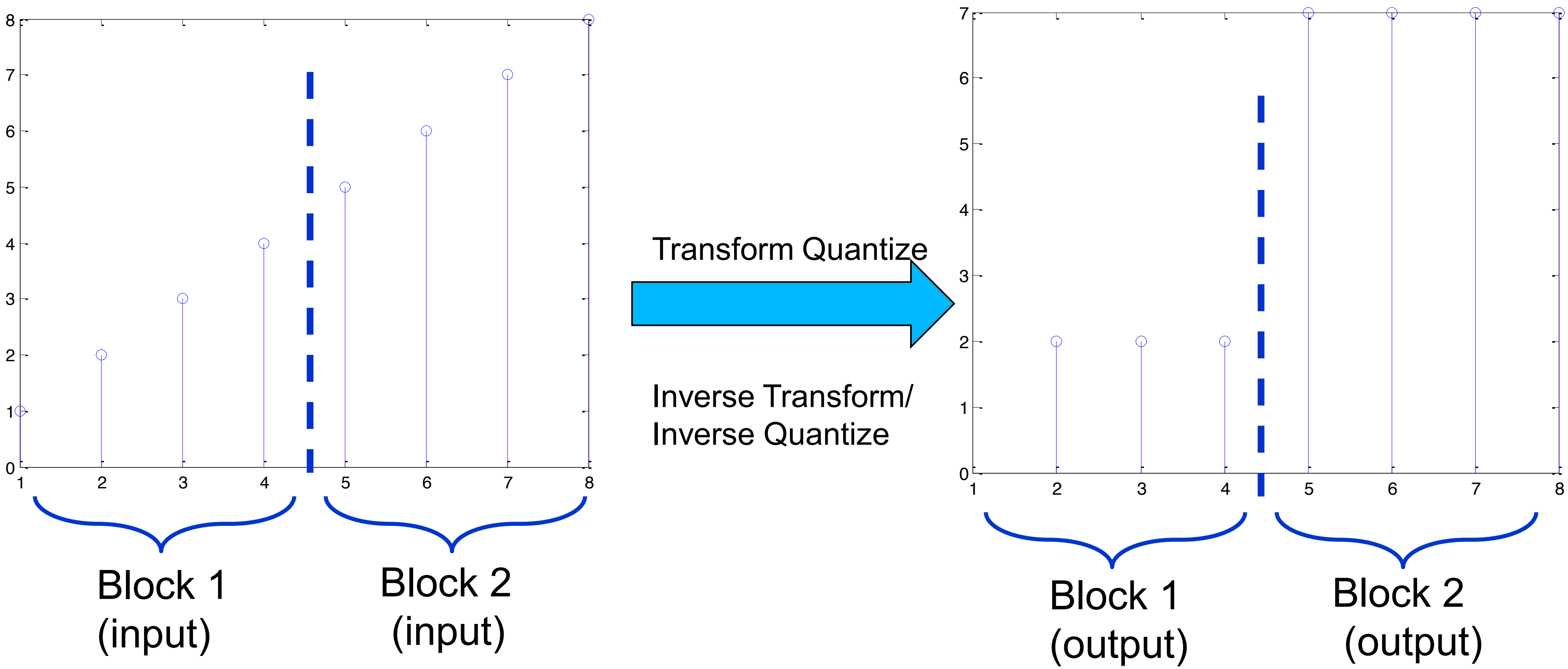
Quantize (Q = 1)	Inverse Transform/Quantize
[5 -2 0 0]	[1 2 3 4]
[13 -2 0 0]	[5 6 7 8]

Decoded surface : [1 2 3 4 5 6 7 8]

Quantize (Q = 5)	Inverse Transform/Quantize
[1 0 0 0]	[2 2 2 2]
[3 0 0 0]	[7 7 7 7]

Decoded surface : [2 2 2 2 7 7 7 7]

Blocking

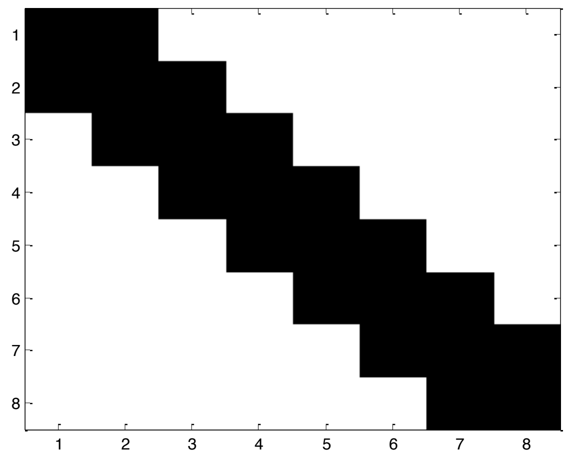

Transform Quantize
Inverse Transform/
Inverse Quantize
Block 1
(input)
Block 2
(input)
Block 1
(output)
Block 2
(output)

Blocking

Source: wikipedia

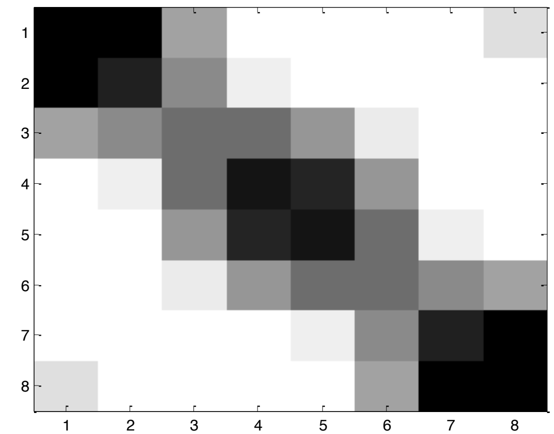
Ringing

- Ringing artefact: Spurious signal signals near sharp edges in the picture
    - Reason: Loss of high frequency harmonics due to quantization



Transform Quantize

Inverse Transform/
Inverse Quantize

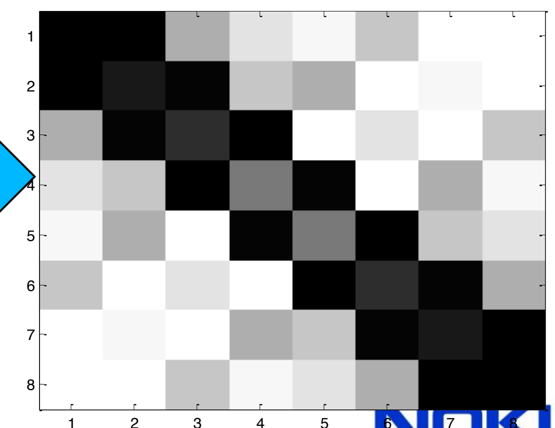


1.34	0.00	0.08	0.00	0.06	0.00	0.03	0.00
0.00	-0.60	0.00	0.10	0.00	0.07	0.00	0.02
0.08	0.00	-0.51	0.00	0.08	0.00	0.05	0.00
0.00	0.10	0.00	-0.36	0.00	0.06	0.00	0.02
0.06	0.00	0.08	0.00	-0.19	0.00	0.03	0.00
0.00	0.07	0.00	0.06	0.00	-0.02	0.00	0.01
0.03	0.00	0.05	0.00	0.03	0.00	0.12	0.00
0.00	0.02	0.00	0.02	0.00	0.01	0.00	0.22

DCT Coefficients (/1000)

Transform Quantize

Inverse Transform/
Inverse Quantize



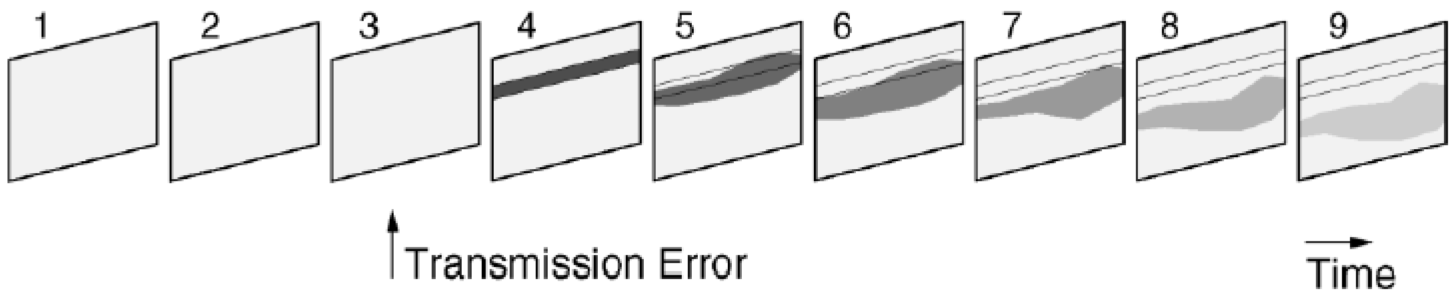
Video Applications: Reliable transfer of video

Review

- Error Resilience Tools
  - Source coding based tools
    - Intra Picture Refresh
    - Picture Segmentation
    - Intra Macro-block refresh
    - Reference Picture Selection
    - Data Partitioning
    - Redundant Pictures
  - Channel Coding based tools
    - Forward Error Correction
    - Interleaving

Error Resilience in Video Coding

- A transmission error gets propagated due to predictive coding



Intra Refresh

- Place INTRA macroblocks to stop error propagation.
- Questions:
  - Why not place an INTRA frame?
  - How to place the INTRA macroblocks?
  - How much INTRA macroblocks?

Reference Picture Selection

- Reference Picture Selection
  - Using correct reference picture according to decoder-side feedback
  - In case there is no error, encoder uses the regular prediction chain

$$0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 12 \rightarrow 13 \rightarrow 14 \rightarrow 15$$

  - When encoder receives a message indicating an error, correct referencing is used

$$0 \rightarrow 1 \rightarrow \cancel{2} \rightarrow 3 \rightarrow 4 \rightarrow 5 \quad 6 \rightarrow 7 \rightarrow 8 \rightarrow \cancel{9} \rightarrow 10 \rightarrow 11 \rightarrow 12 \quad 13 \rightarrow 14 \rightarrow 15$$

Video Applications

- Different applications have different requirements for video coding.
- Three different applications will be reviewed:
  - TV Broadcast
  - Video Streaming vs. Download
  - Video Conferencing

Video Applications

- TV Broadcast can be mainly characterized by:
  - Constant channel rate
    - TV stations rent a fixed channel rate
  - Vast distribution (one to all) without feedback channel
    - Thousands of people watch at the same time

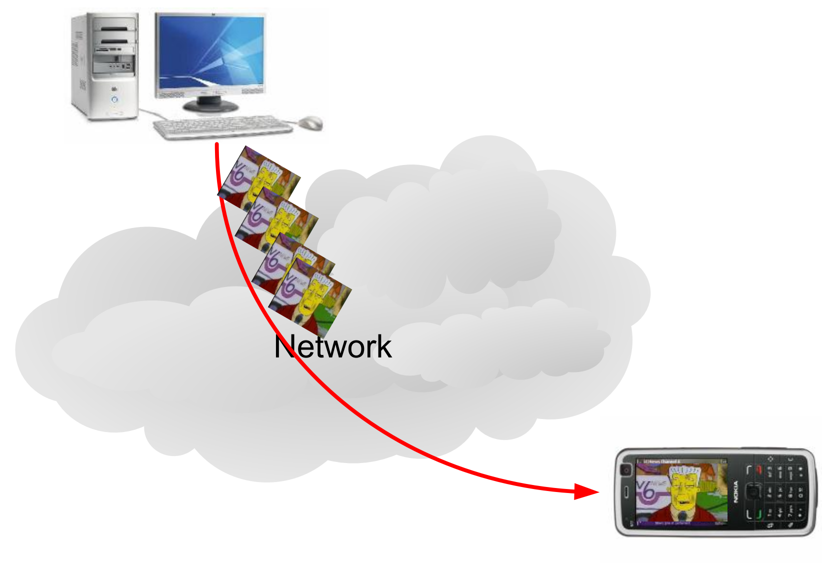
Video Applications

- Constant channel rate
  - Video quality at that bitrate is very important
  - More focus on difficult to code videos as the artifacts will be more visible

- Vast distribution (one to all):
  - Feedback based error resilience is not possible

Streaming vs. Download

- Download:
  - Receive entire content before playback begins
  - High “start-up” delay as media file can be large
    - User might not want to wait for long time
  - 1 GB for a 2 hour CIF resolution movie encoded using H.264/AVC
    - Might not be possible for a mobile device with limited storage capacity
- Streaming:
  - Play the media file while it is being received
  - Reasonable “start-up” delays
- What is YouTube (streaming, downloading)?
  - Progressive Download (you can playback during download)



YouTube

NOKIA
Connecting People

Streaming Live Multimedia

- Example Applications:
  - Internet radio
  - Live sport event
- Streaming
  - playback buffer
  - playback can lag some seconds after transmission
  - still have timing constraint
- Interactivity
  - fast forward not possible
  - rewind, pause still possible

Video Conferencing

- Video Conferencing
  - One of the most difficult video application to realize
  - Main problems are:
    - Need for low delay
    - Error resilience
    - Different end-points with different capabilities