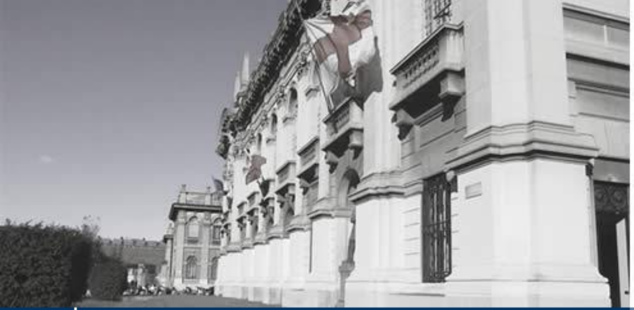


POLITECNICO DI MILANO

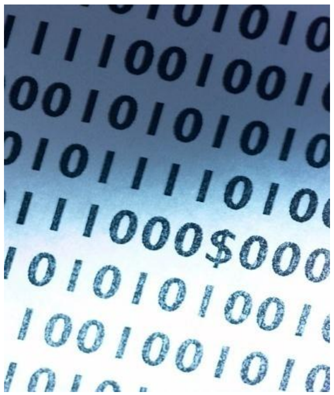
MPEG Encoding

OVERVIEW

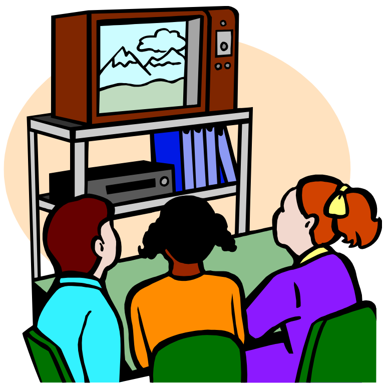
Introduction

- Before the encoding stage:
  - Capture the video content
  - Digitalization of a video
    - Sampling and quantization techniques
- Uncompressed video

Capture



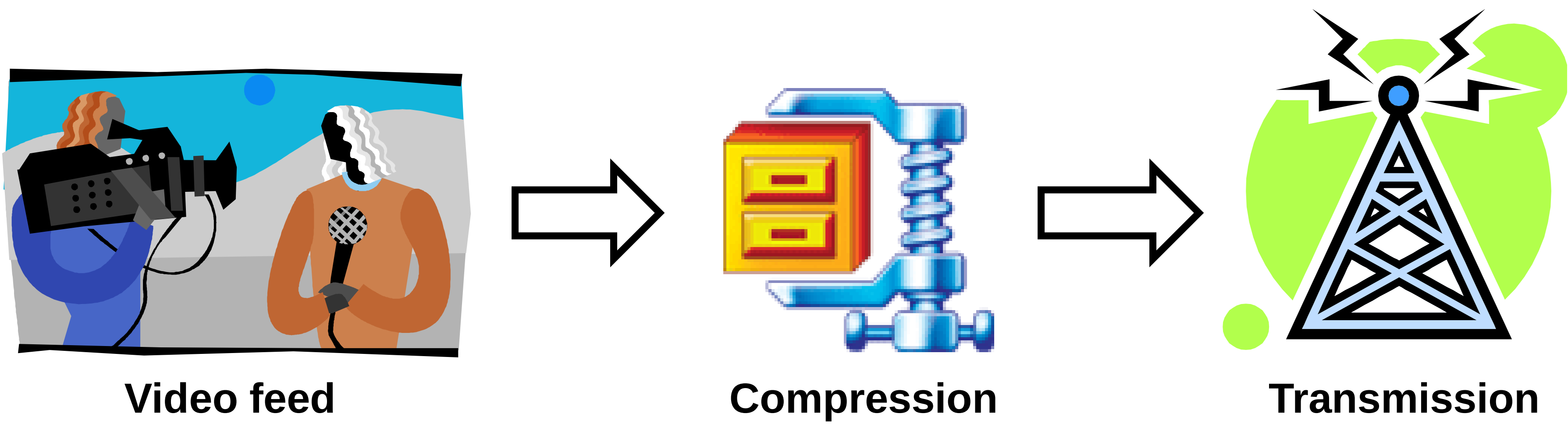
Digitalization



Uncompressed video

Encoding

- Needs of specialized hardware
- 3 steps:
  - Receive a video feed
  - Apply compression
  - Prepare for transmission



Advantages and Disadvantages of encoding

PROS

- Large reduction of the dimension of a file
- Less use of computational power
- Less time to send the video over the network
- Possible use of relatively low capacity broadband connections.

CONS

- Delays
- Less overall quality of the image
- Risk to affect the quality of the signal

Compression

- Compression allows to broadcast high quality video and audio channels over IP broadband network:
  - Takes advantages from human and aural systems deficiencies
  - Reduce the size of the original signal
  - “Compression ratio”: level of compression
    - High compression ratios will often decrease the quality of the resulting video signal

Categories

- Categories:
  - **Lossless**
    - Allows to perfectly recreate the original image
    - Used for still images
  - **Lossy**
    - Some video image information is destroyed
    - Most used
    - Most popular are MPEG and VC-1

THE NEED FOR COMPRESSION

The need for compression

- If digital systems offer so many advantages, why have we not moved to digital television long ago?

Resolution of digital television

Picture rate (Hz)	25	30
Lines per picture	576	480
Luminance samples per line	720	720
Fields per second	50	60
Interlace	2:1	2:1

Bandwidth requirements for uncompressed digital video

- For both 25 and 30 Hz transmission, 14’400 lines per second are transmitted, which means 10’368’000 pixels each second.
- Signal sampled at half the rate of luminance signal, that is 360 samples per line
  - 8 bit accuracy implies an average of 16 bits for luminance samples
- Raw bit rate = 10’368’000 luminance samples per second * 16 bits per sample.
- Data rate = 165,89 Mbit/s
  - Too high!!

Bandwidth requirements for uncompressed digital audio

- Audio channel sampled at 44,1 KHz
- Resolution 16 bits per sample
- It requires 705,6 Kbit/s
- Surround-sound systems (5 channel audio): 705,5 * 5 = 3,5 Mbit/s

Target bandwidth

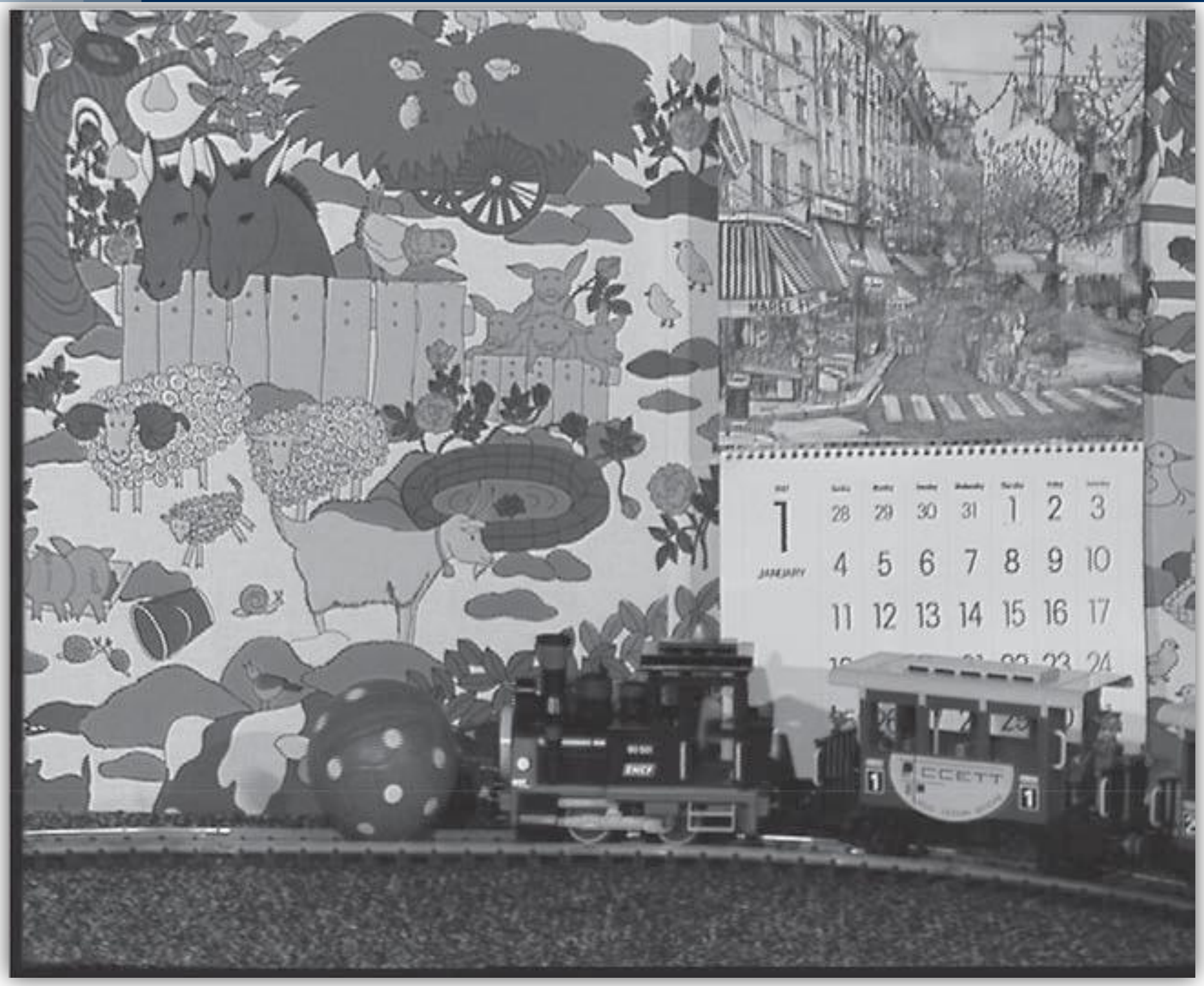
- 5–10 Mbit/s is a reasonable target bit rate for a digital television service, with approximately
  - 10% of the available data rate taken by transmission overheads
  - 10% allocated to audio
  - the remaining 80% to video

CHARACTERISTICS OF VIDEO MATERIAL

Focus points

- Measure of the “sameness”
- Entropy
- Human visual systems

Example image



The "sameness"

- Measured by the autocorrelation function

$$r = \frac{\sum_i \sum_j \left(A(i,j) - \mu_a\right)\left(B(i,j) - \mu_b\right)}{\sqrt{\sum_i \sum_j \left(A(i,j) - \mu_a\right)^2 \left(B(i,j) - \mu_b\right)^2}}$$

Example

130	135
126	122

Subtract Mean (say 128) →

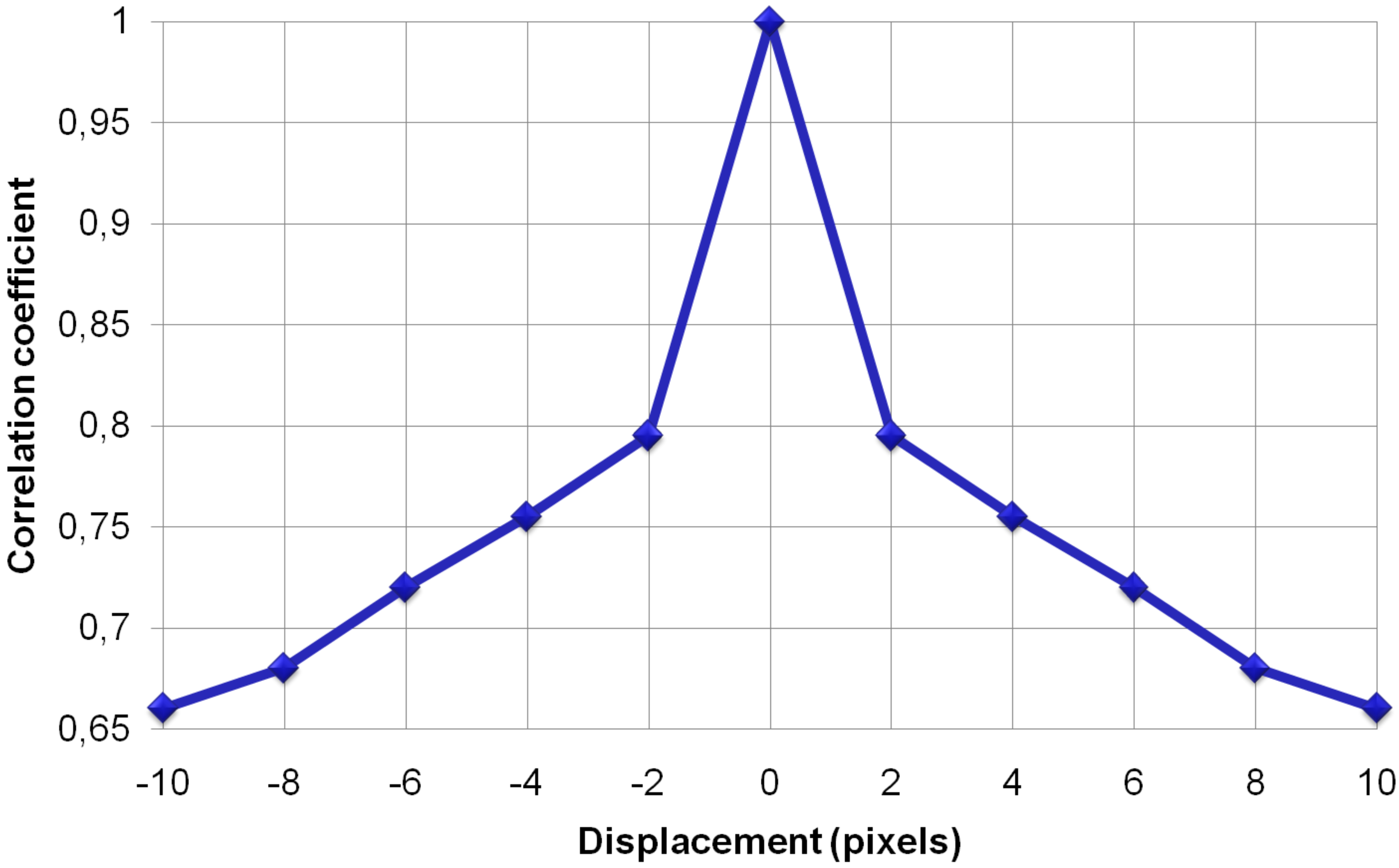
+2	+7
-2	-6

133	131
132	131

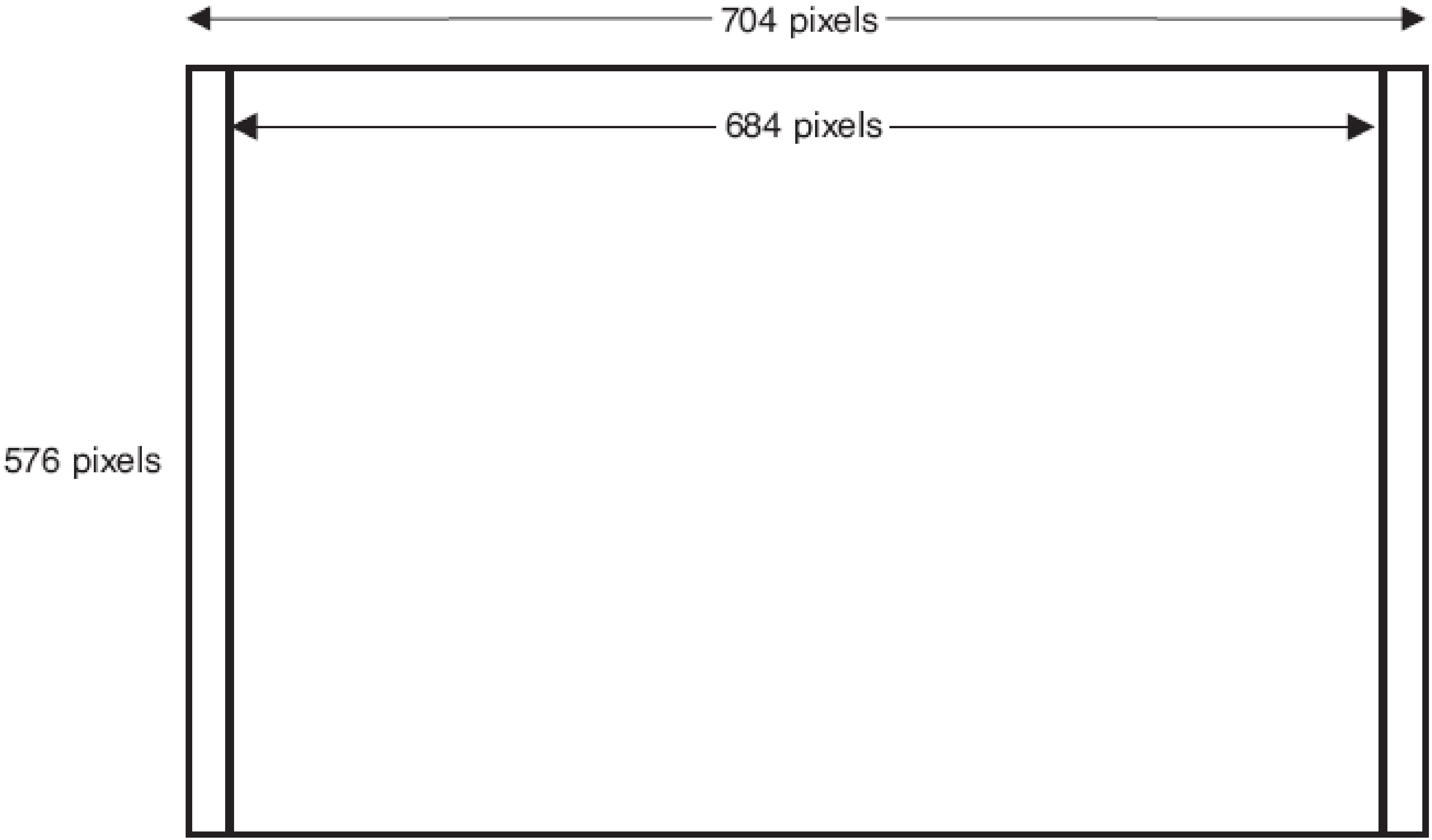
Subtract Mean (say 128) →

+5	+3
+4	+3

Example (2)


1
0,95
0,9
0,85
0,8
0,75
0,7
0,65
Correlation coefficient
-10
-8
-6
-4
-2
0
2
4
6
8
10
Displacement (pixels)

Block area for horizontal correlation coefficient



First conclusions

- The value of a single pixel “tells” us something about its neighbours
- Use of the same technique in a video sequence

Information content

- It is important to be able to quantify in some way the amount of information produced by a source
- p_a: probability that the symbol a is emitted from a source
  - $p_a = 1 \Rightarrow$ the symbol is produced continuously by the source.

$$I_a = \log_2\left(\frac{1}{p_a}\right) = -\log_2(p_a)$$

Information properties

1. $$I_a = 0 \ \mathit{for} \ p_a = 1$$

2. $$I_a \geq 0 \ \mathit{for} \ 0 \leq p_a \leq 1$$

3. $$I_a \geq I_b \ \mathit{if} \ p_a < p_b$$

4. $$I_{ab} = I_a \cap I_b = I_a + I_b$$

Entropy

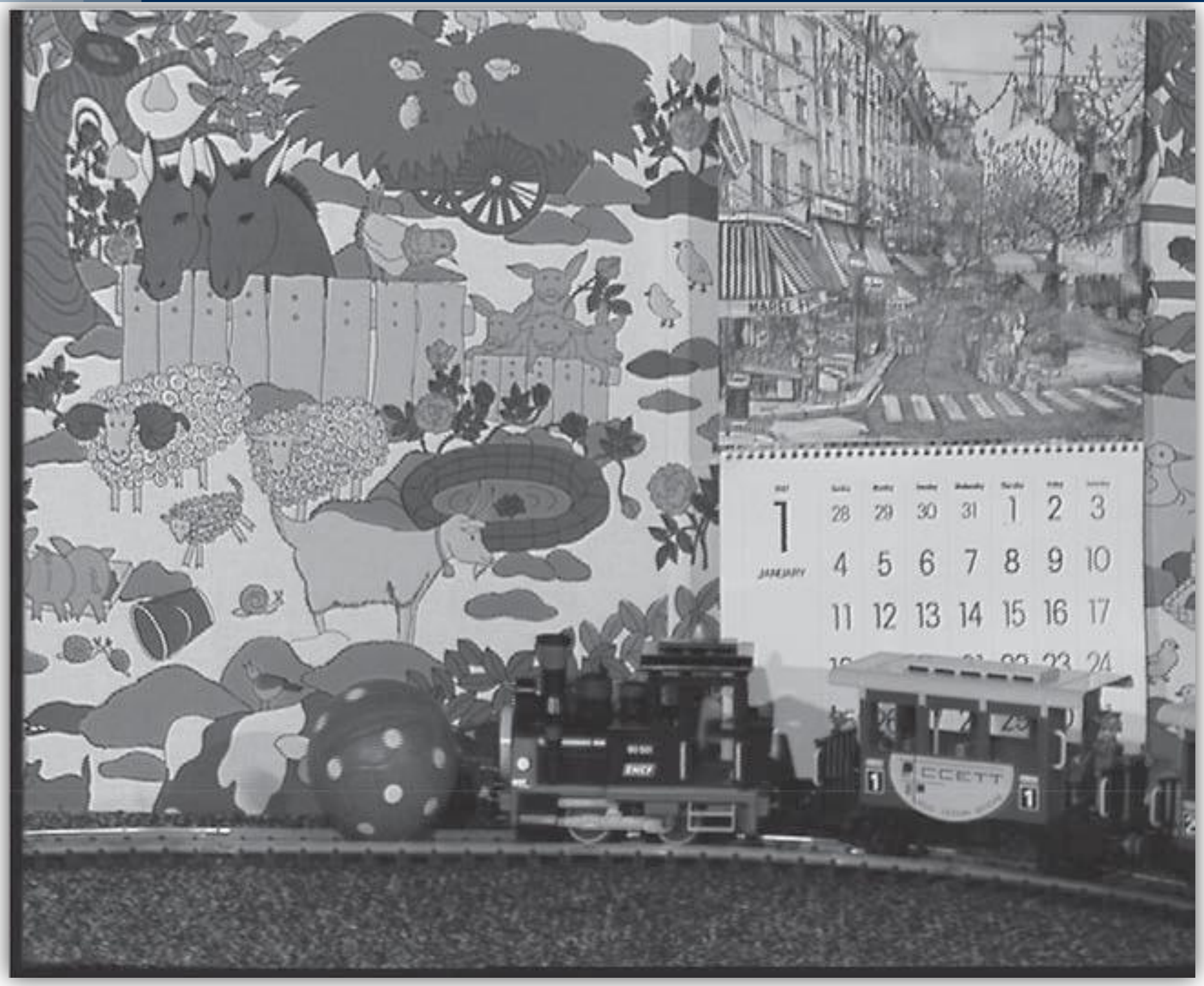
- Consider a source producing m symbols s

$$H = E\left(I_{Sk}\right) = \sum_{k=0}^{m-1} p_k I_{Sk} = \sum_{k=0}^{m-1} p_k \left(\log_2 \left(\frac{1}{p_k} \right) \right)$$

$$= -\sum_{k=0}^{m-1} p_k \log_2 p_k$$

- H is called *entropy* of the source

Entropy in our example



Entropy in our example

- Extend the process to measure the information content of a picture
- In an 8-bit gray scale picture, each pixel takes one of 256 possible values (0–255)
- Calculate the probability and the entropy
  - If all 256 values were equally likely, the entropy would be 8 bits/pixel
  - The entropy in the example picture is 7.61 bits/pixel

The human visual systems

- The aim of the process is to produce a reconstructed picture that, when viewed by a human viewer, is of sufficient quality to meet the needs of the service
- Much of the compression achieved relies upon the introduction of coding artifacts in areas of a picture where they will not be subjectively noticeable

Characteristics of the human visual system

- Perception of change in brightness
- Spatial masking
- Temporal masking
- Tracking of motion

ENTROPY BASED CODING

Entropy Coding

- Idea of using variable length coding to efficiently represent messages
- Huffman coding:
  - CodeWords
  - Most used in current video and audio compression standards

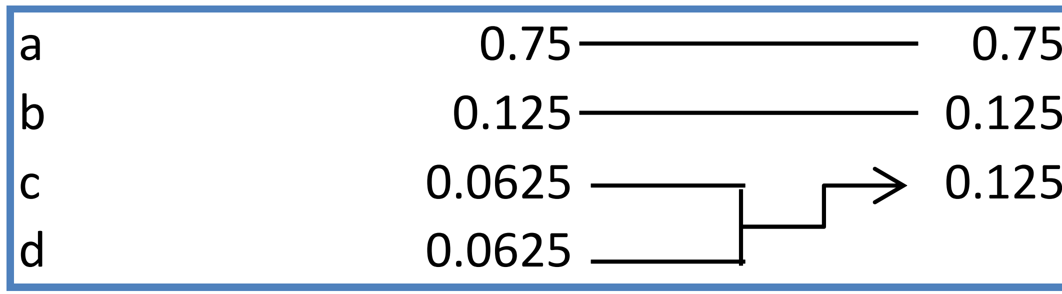
Huffman Coding - Steps

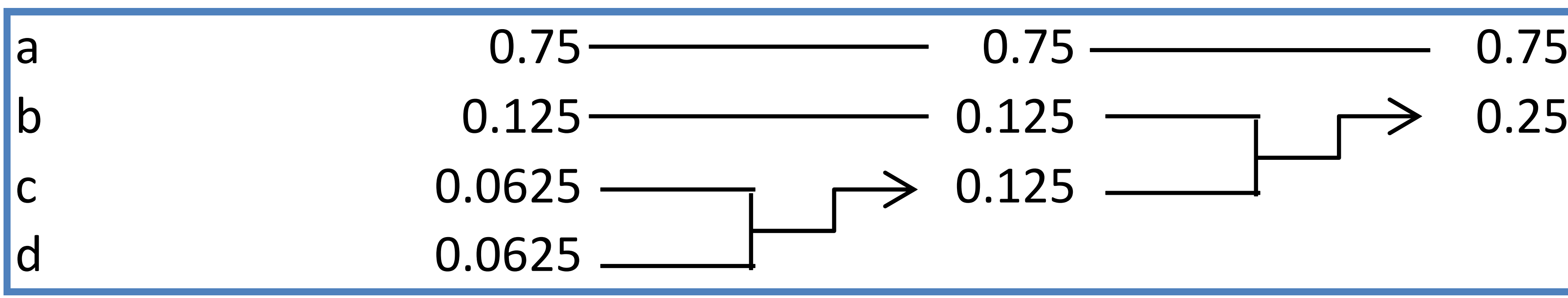
- List the symbols to be transmitted in decreasing order of probability
- Combine the two symbols with the smallest probabilities and reorder the symbols in decreasing order of probability
  - Repeat until only two combined symbols remain.
- Mark each combination consistently with either $\rbrack^{0}_{1}$ or $\rbrack^{1}_{0}$
- Read off the Huffman CodeWords from right to left

Example

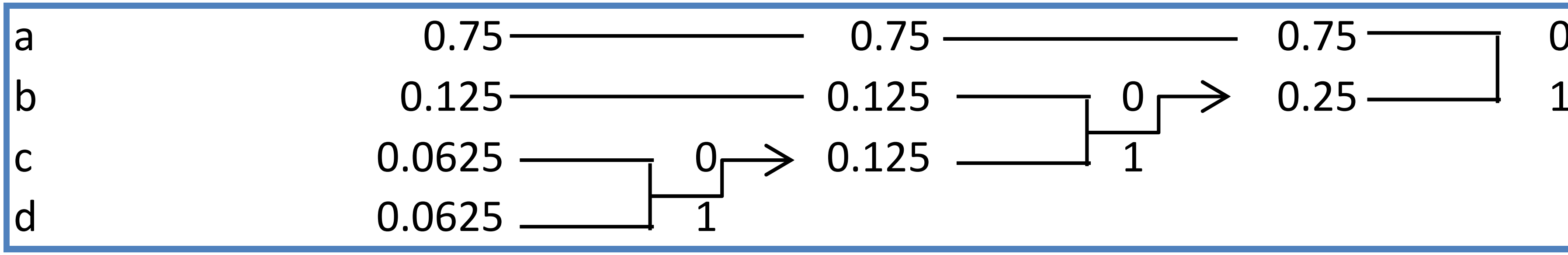
- a, b, c, and d have probabilities 0.75, 0.125, 0.0625, and 0.0625

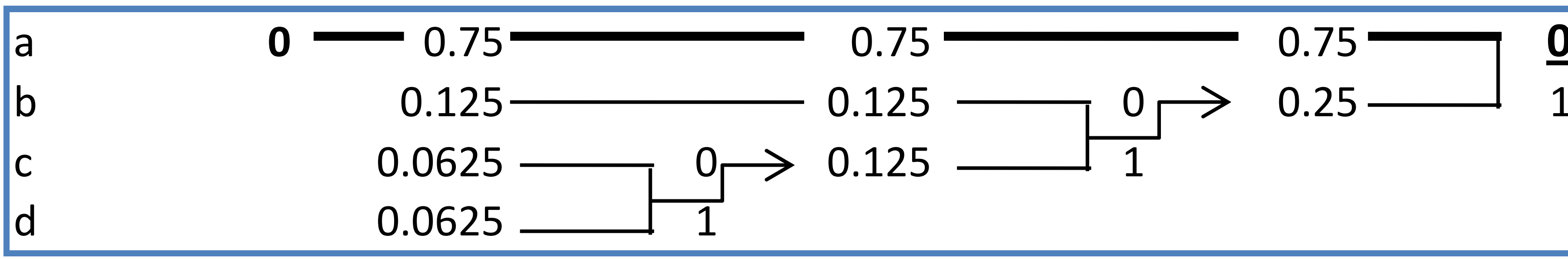
a	0.75
b	0.125
c	0.0625
d	0.0625

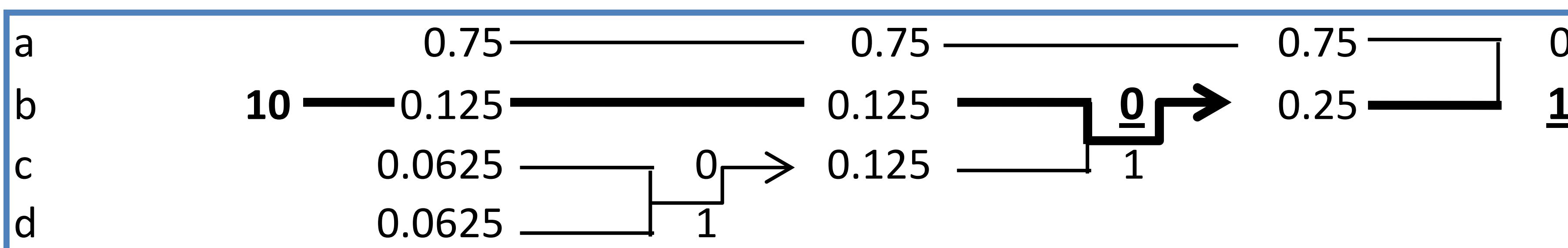




Example (2)

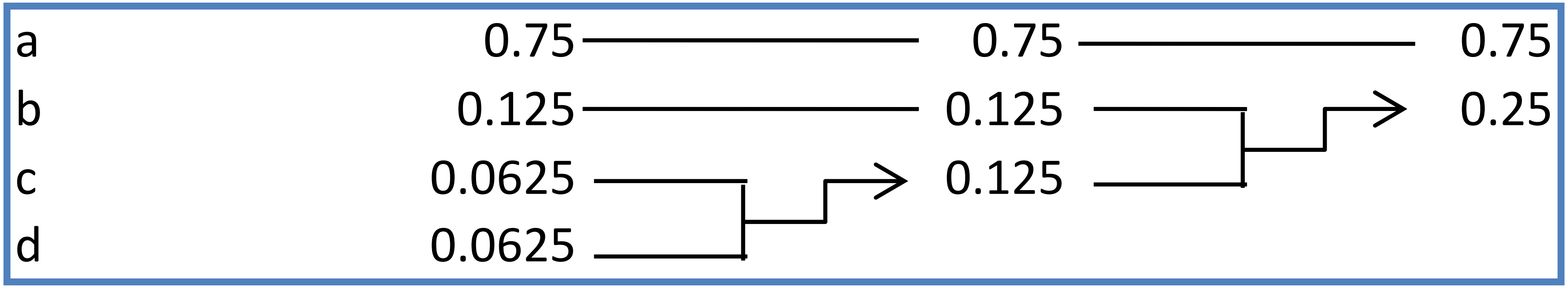

a 0.75 0.75 0.75 0
b 0.125 0.125 0.25 1
c 0.0625 0.125
d 0.0625
0
1
0
1


a 0 0.75 0.75 0.75 0
b 0.125 0.125 0.25 1
c 0.0625 0.125
d 0.0625
0
1
0
1


a 0.75 0.75 0.75 0
b 10 0.125 0.125 0.25 1
c 0.0625 0.125
d 0.0625
0
1
0
1

Considerations

- No short CodeWord ever forms a prefix for a longer CodeWord in a Huffman code.



a CW	0
b CW	10
c CW	110
d CW	111

Considerations (2)

- Errors problem

01011011100010 **Correct**

0	**10**	**110**	**111**	**0**	**0**	**0**	**10**
a	**b**	**c**	**d**	**a**	**a**	**a**	**b**

0**0**011011100010 **Single bit error**

0	**0**	**0**	**110**	**111**	**0**	**0**	**0**	**10**
a	**a**	**a**	**c**	**d**	**a**	**a**	**a**	**b**

Considerations (3)

- Integer approximation limit
  - Huffman codes cannot exactly achieve the entropy of the symbol stream

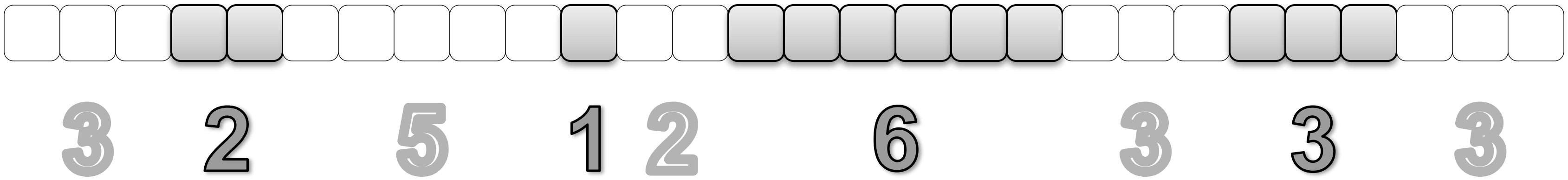
Symbol	Probability
a	1/3
b	1/3
c	1/3

→ **Optimum # of bits = 1.6**

→ **Hoffman assigns 1 or 2 bits**

Run Length Coding

- A run of consecutive identical symbols is represented by a single variable length (e.g., Huffman) CodeWord



PREDICTIVE ENCODING

Predictive Coding

- Exploit the correlation between nearby pixels
  - information that needs to be transmitted can be reduced
    - Use of predictive encoding

- Transmit only the difference between the prediction and the actual pixel value
  - Pixel value prediction is made using the value of the pixel immediately to its left.

Example

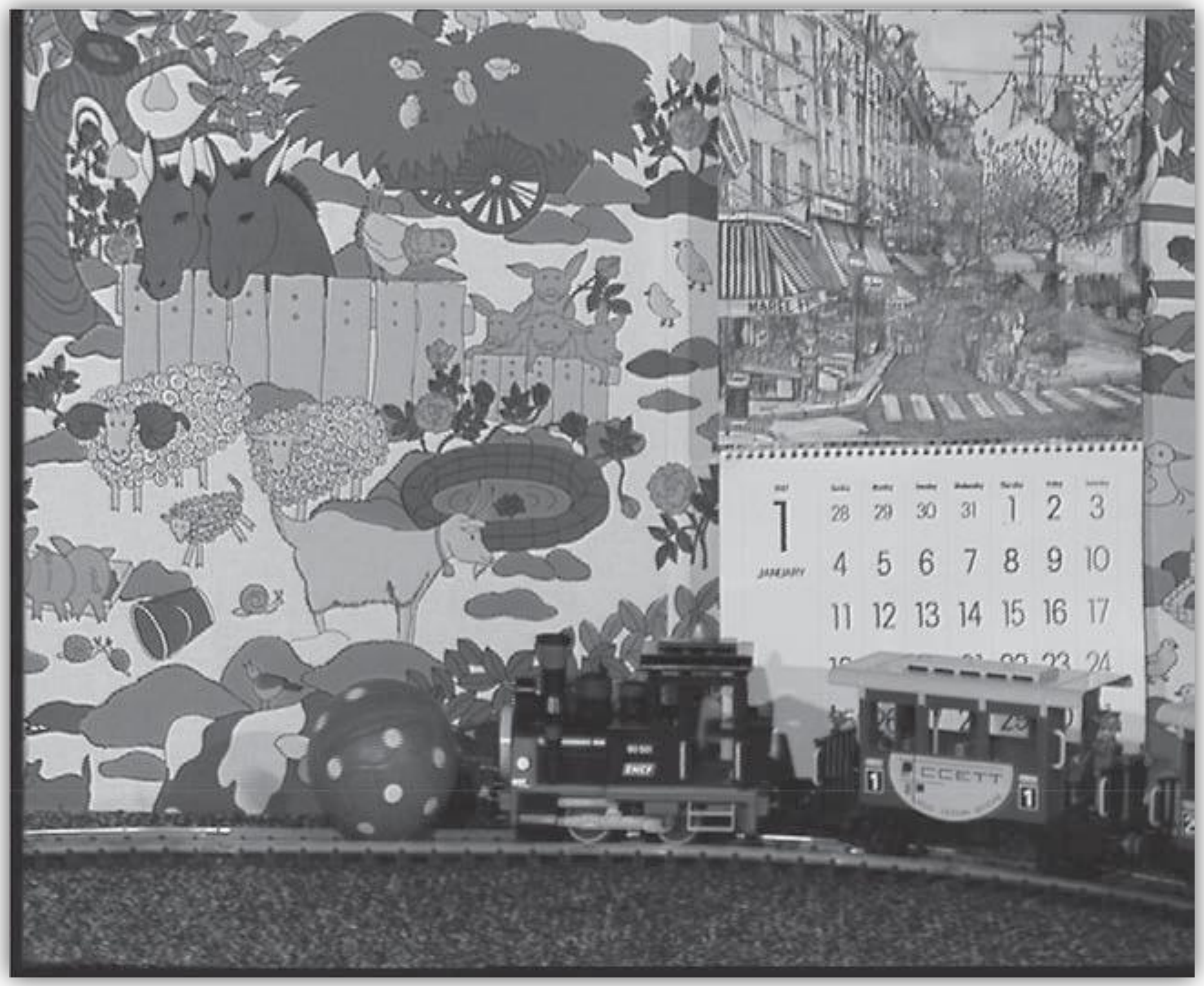
- $\hat{X} = A$
- Transmitted: $X - \hat{X}$

  $X - A$

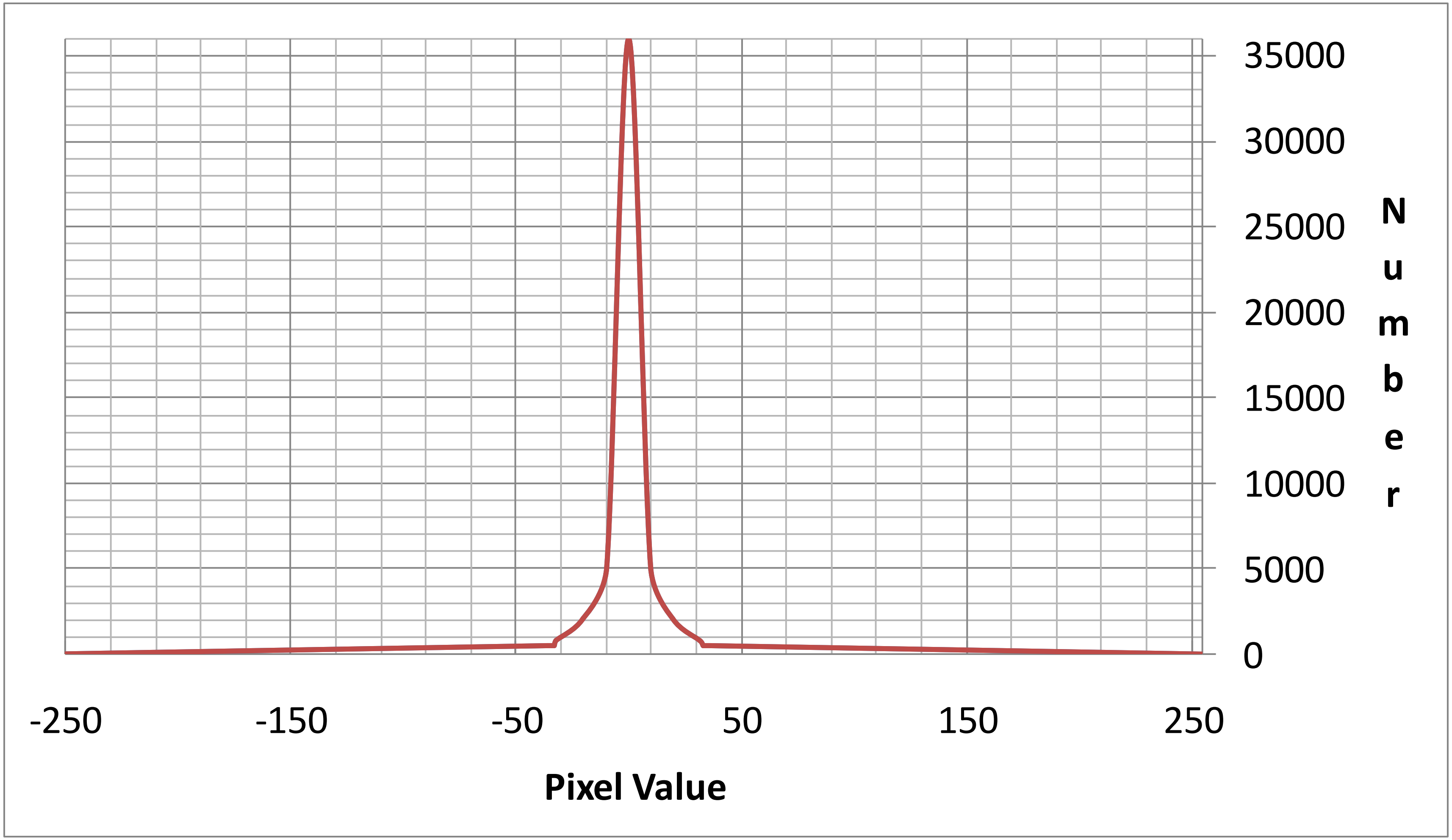
Pixel value	130	135	141	129	151
Prediction	128	130	135	141	129

+2	+5	+6	-12	+22

"Mobile and Calendar" Example



Example: Histogram in “Mobile and Calendar”



One-dimensional DPCM difference picture

Generalized k-dimensional predictor

- Any pixel whose value has already been transmitted to the receiver can be used as the predictor
- More than 1 pixel can be used

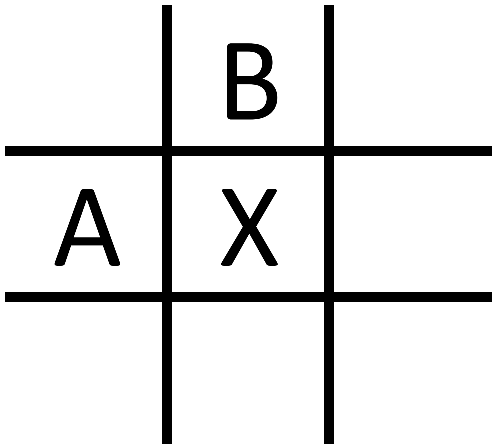
$$\hat{X} = k_1 A + k_2 B + k_3 C + k_4 D$$

B	C	D
A	X	

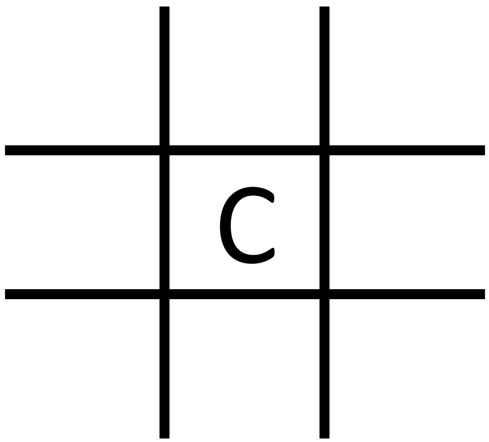
Three-dimensional predictor

$$\hat{X} = k_1 A + k_2 B + k_3 C$$

Current Frame

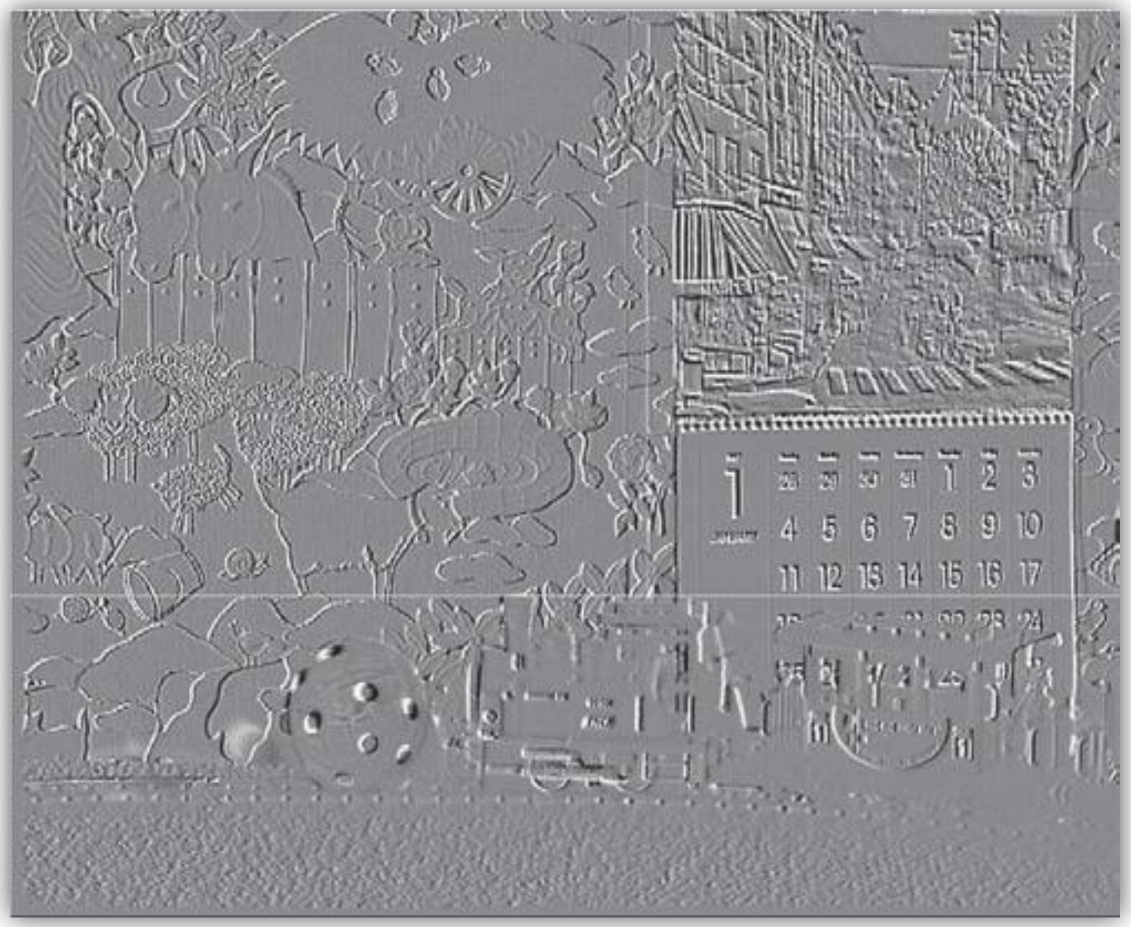


Prediction Frame



Interpicture prediction

- It’s also possible to predict from a previous picture or pictures



Motion-compensated prediction

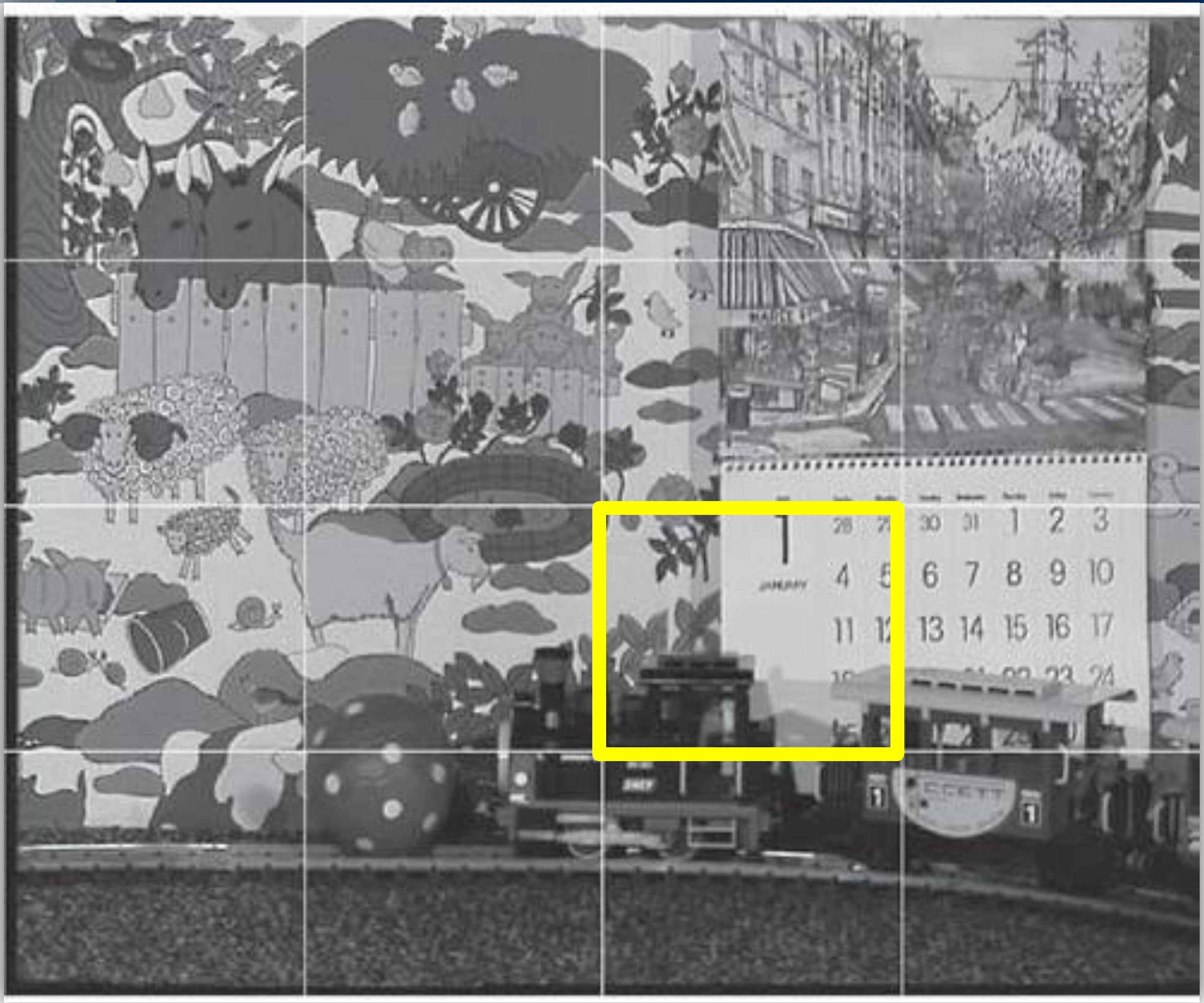
- To take account of the motion that occurs between pictures would be interesting
- High complexity
- Solution: approximate all motion as translational motion in a plane normal to the camera axis
  - Can be made for sufficiently small objects.
- Two steps:
  - **Motion estimation**
  - **Motion compensation**

Motion Estimation

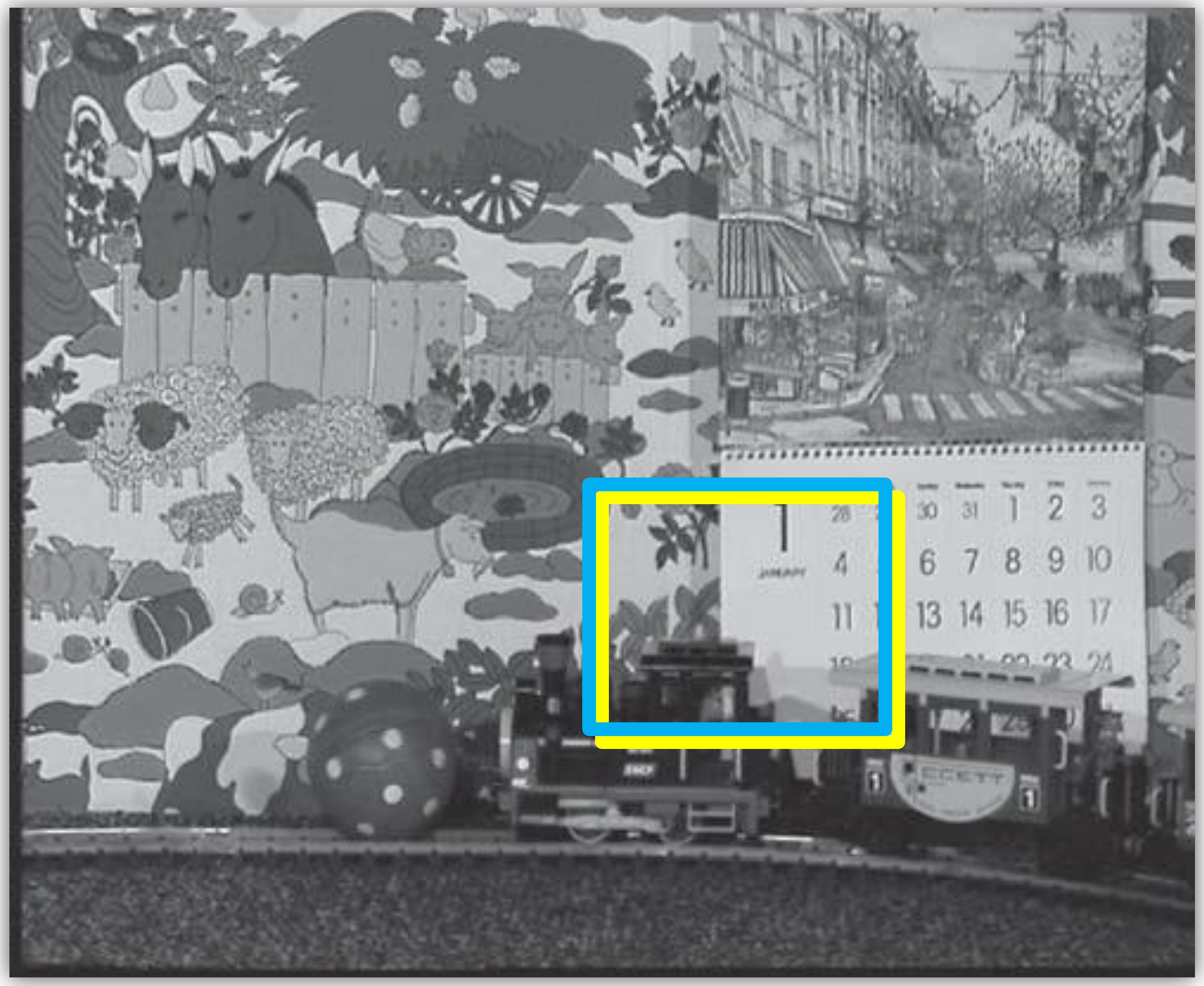
- Estimate the transitional motion
- Two approaches:
  - Partition into **individual objects**
    - Many problems
  - Partition into a **number of blocks**
    - Most used

Example of prediction: first frame

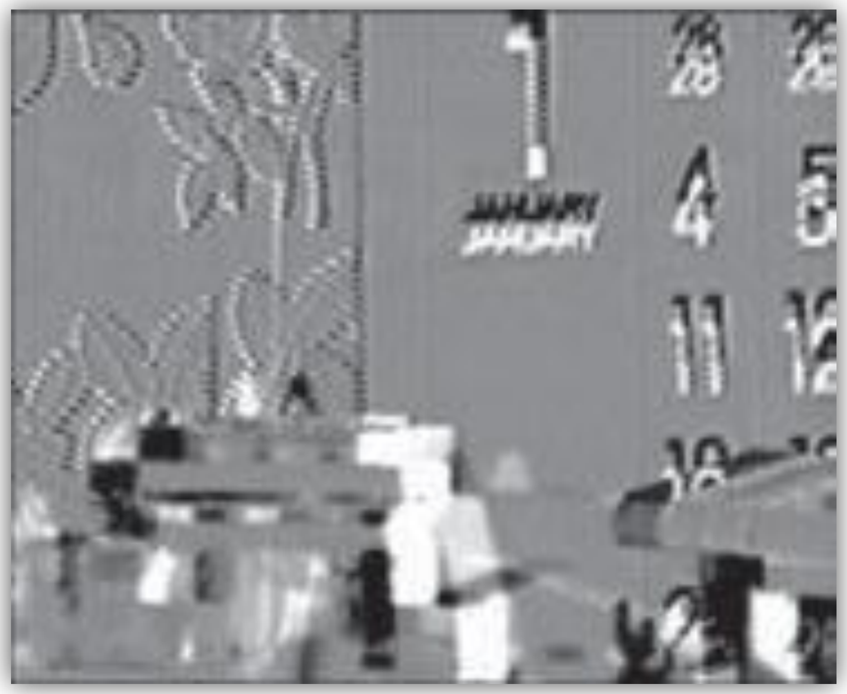
Example of prediction: second frame



Best matching block in the prediction picture



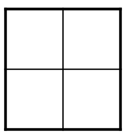
Interpicture difference and Motion compensated difference



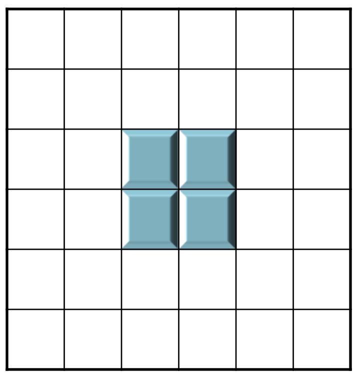
Matching blocks

- In general the best matching block is usually not identical to the block in the current picture
- Matching criteria:
  - Summed absolute difference
    - Most used, more simple
  - Summed squared difference
    - Produces a slightly superior result
- How the search is to be performed?
  - Fixed search area

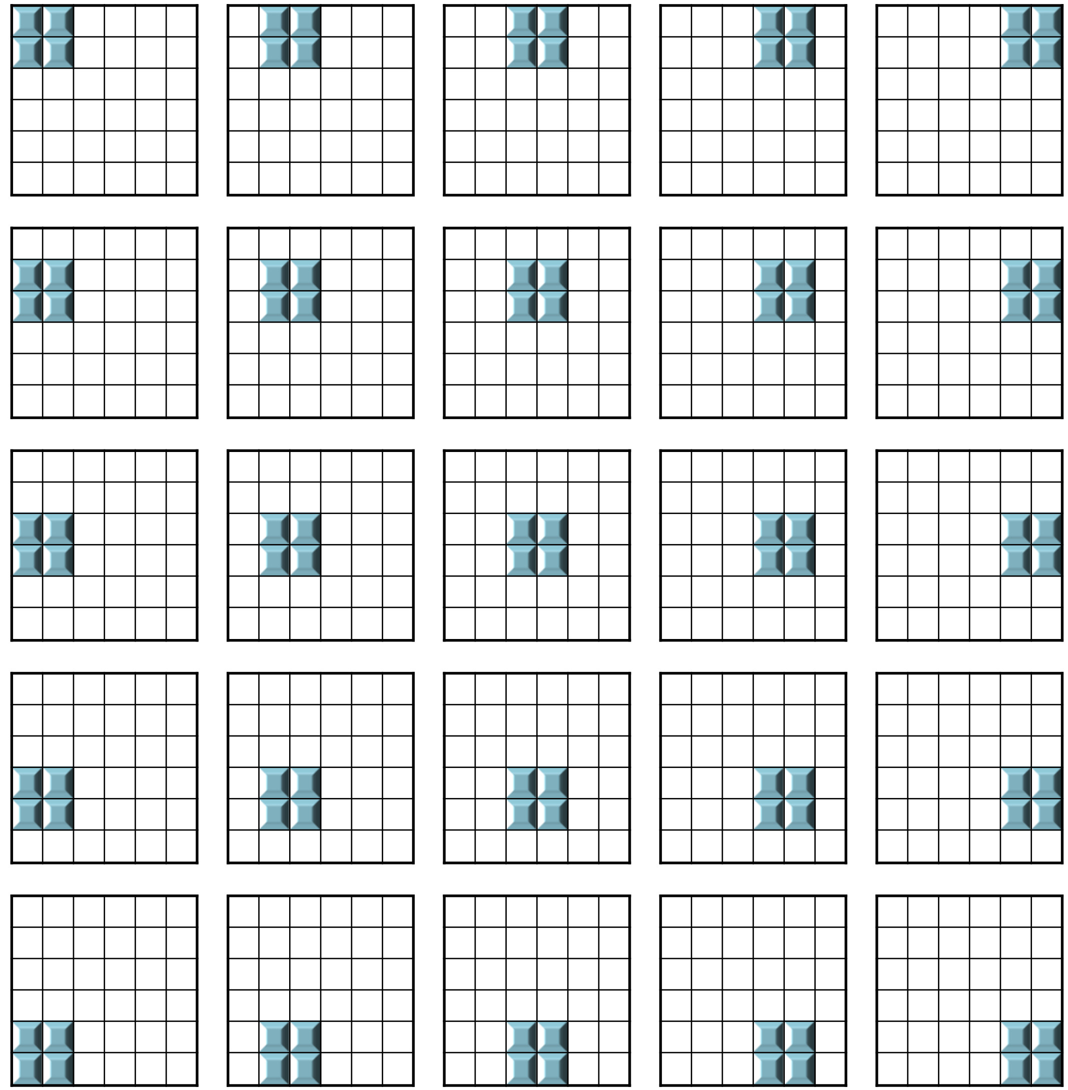
Example 51



Block from current picture

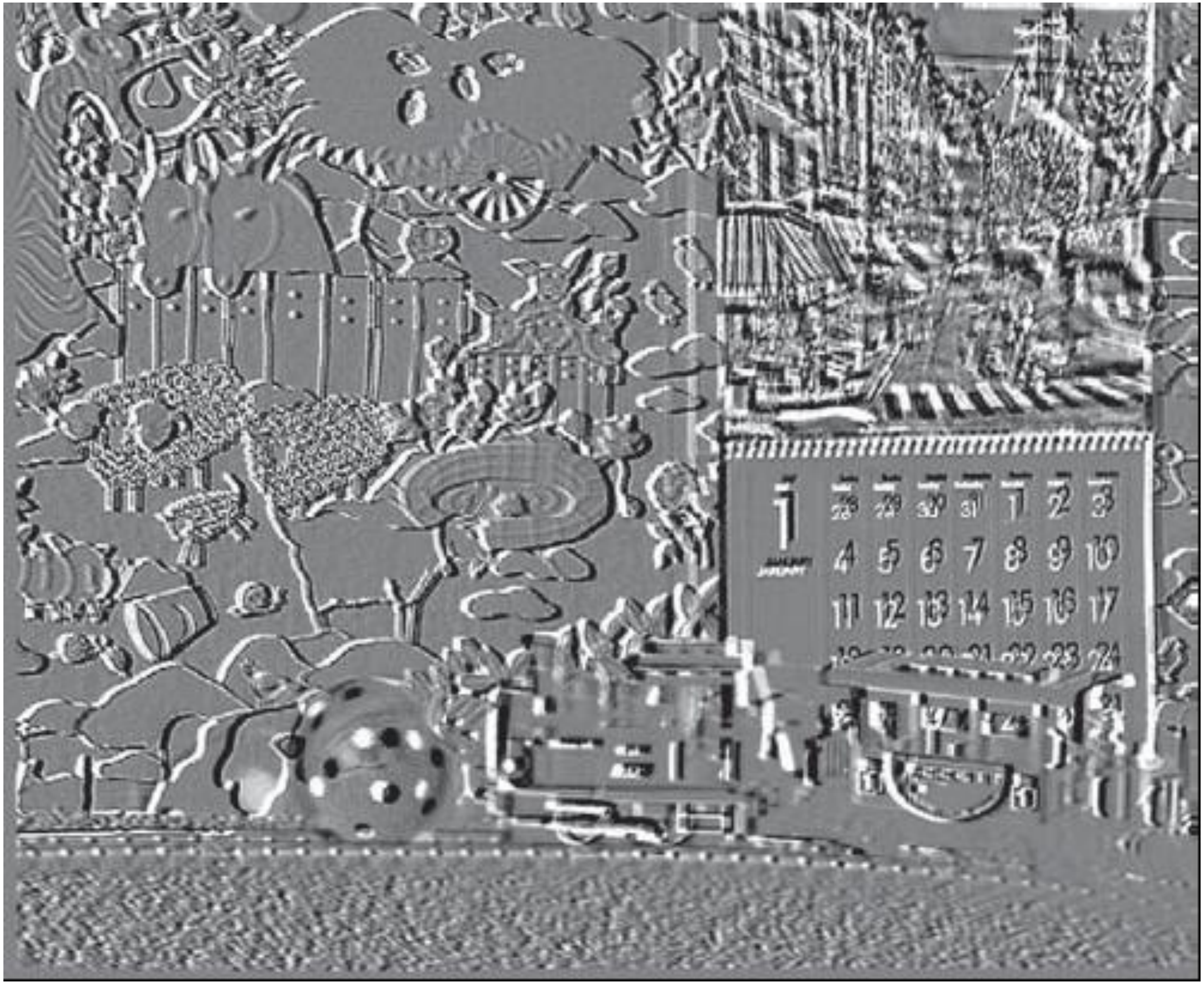


Search area from prediction picture

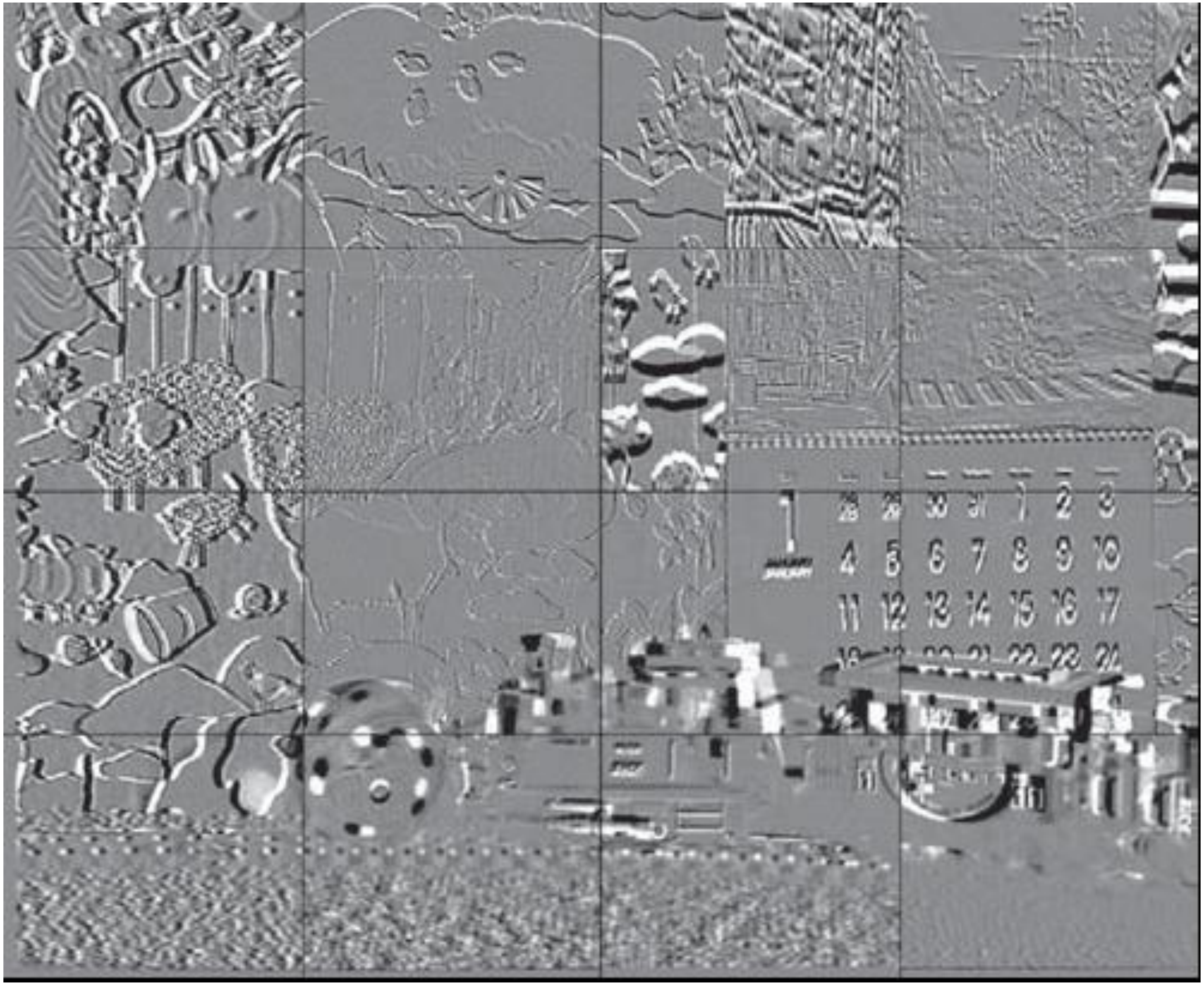


Possible search position

Difference Image



Motion Compensated Image: 96 x 96 pixel blocks



Problem

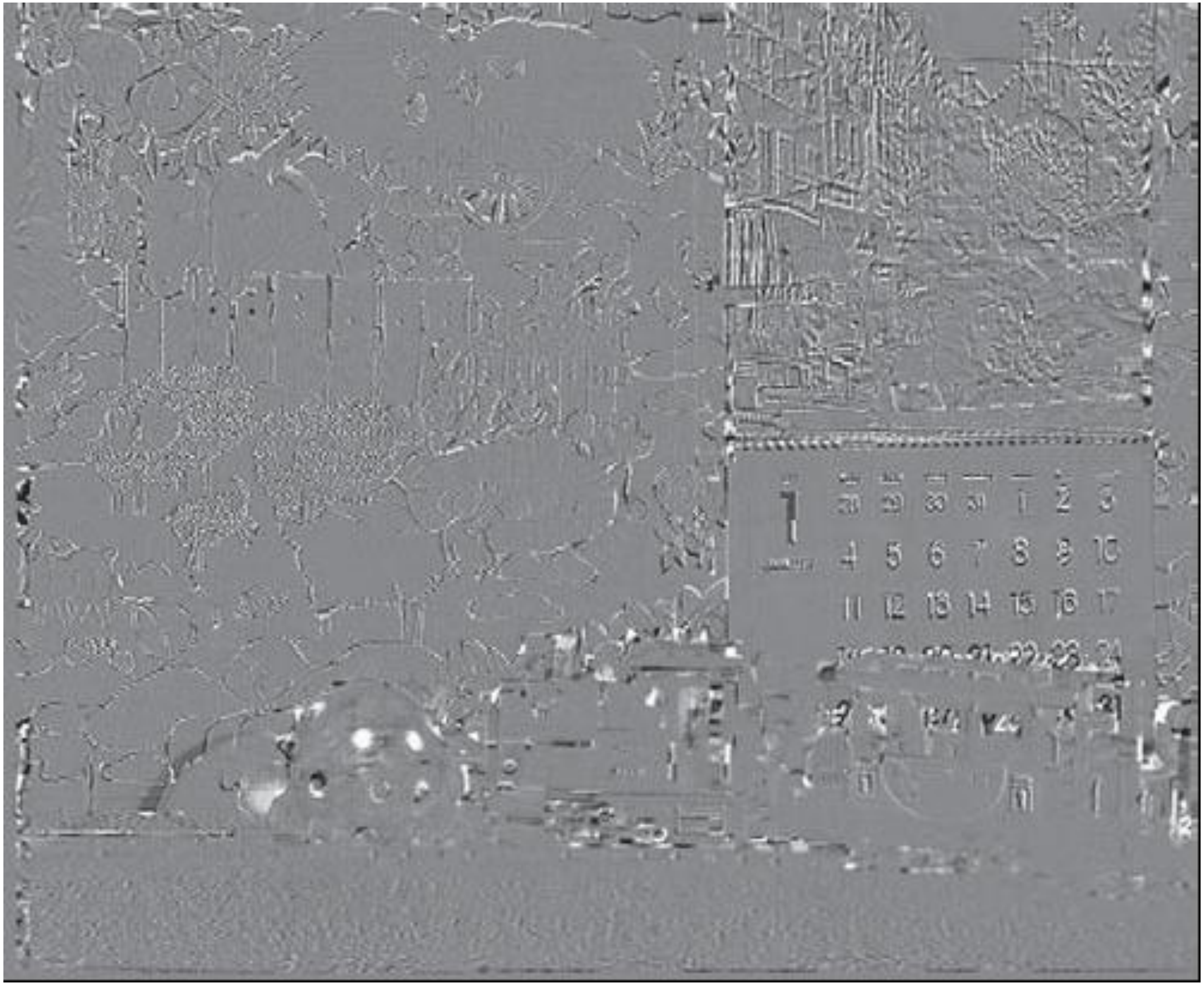
- The block size should be sufficiently small
  - Do not contains two or more different motions
- The block should be sufficiently large
  - Overhead associated with the motion vector not excessive

Example

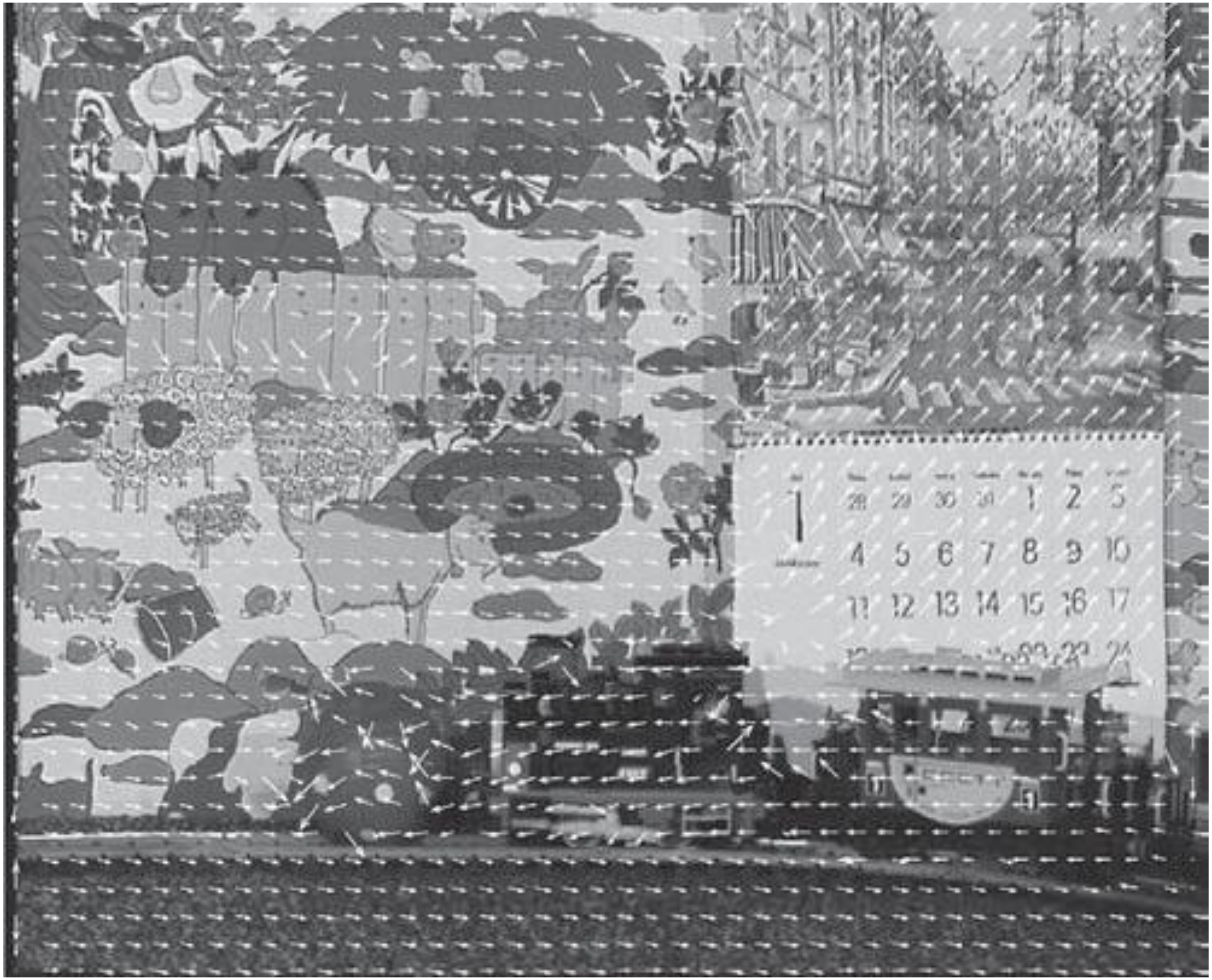
- Search range ± 7 pixels → 8 bits to represent components motion
- Motion vector overhead: (8 bits/ block)/block size

Blocks size (pixels)	Motion vector overhead (bits/pixel)
1x1	**8.0000** = 8/(1x1)
2x2	**2.0000** = 8/(2x2)
4x4	**0.5000** = 8/(4x4)
8x8	**0.1250** = 8/(8x8)
16x16	**0.0313** = 8/(16x16)
32x32	**0.0078** = 8/(32x32)

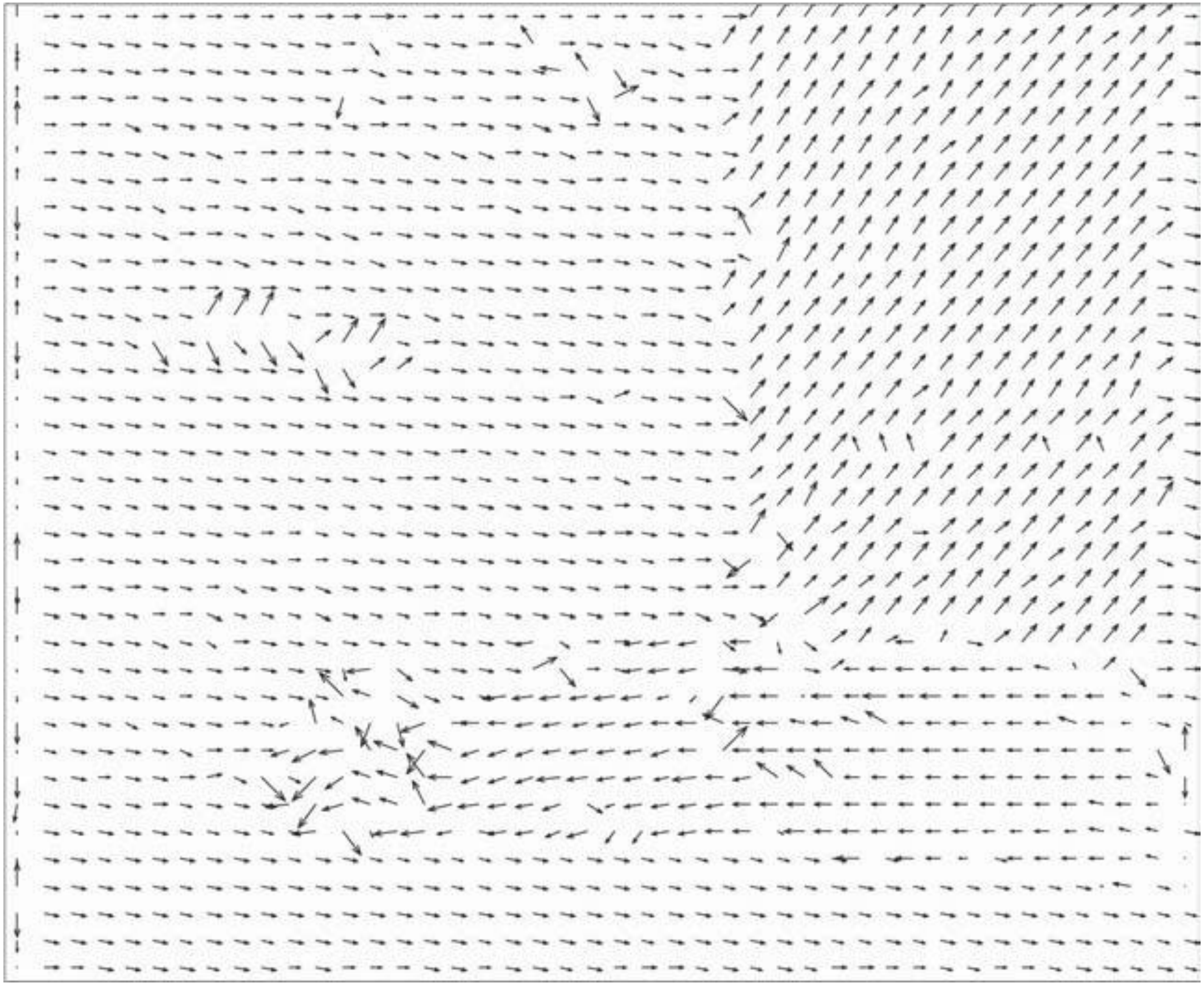
Motion Compensated Image: 16 x 16 pixel blocks



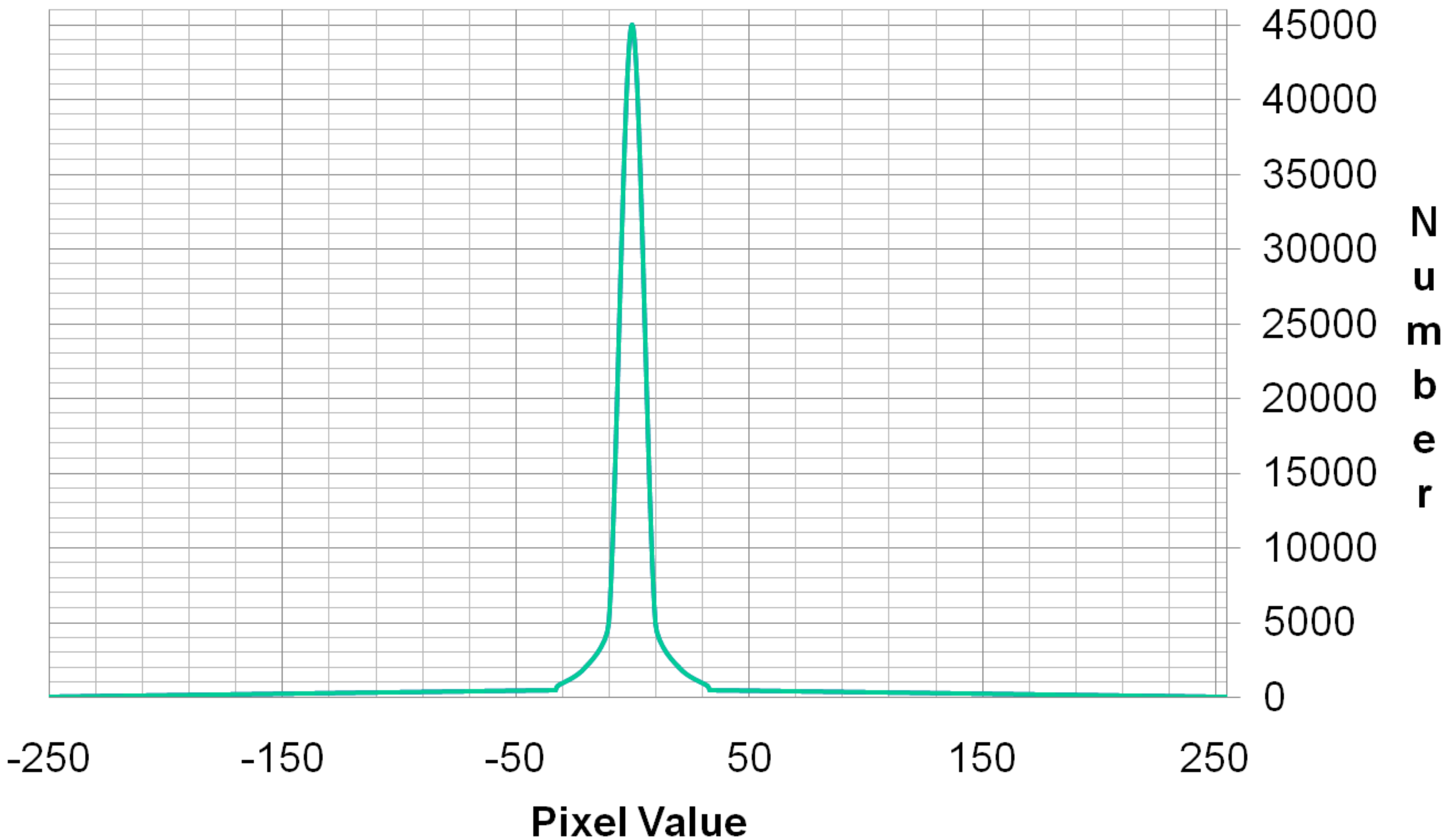
Motion vectors: 16 x 16 blocks



Motion vectors only: 16 x 16 blocks



Histogram for the motion-compensated difference picture (16 x 16 blocks)

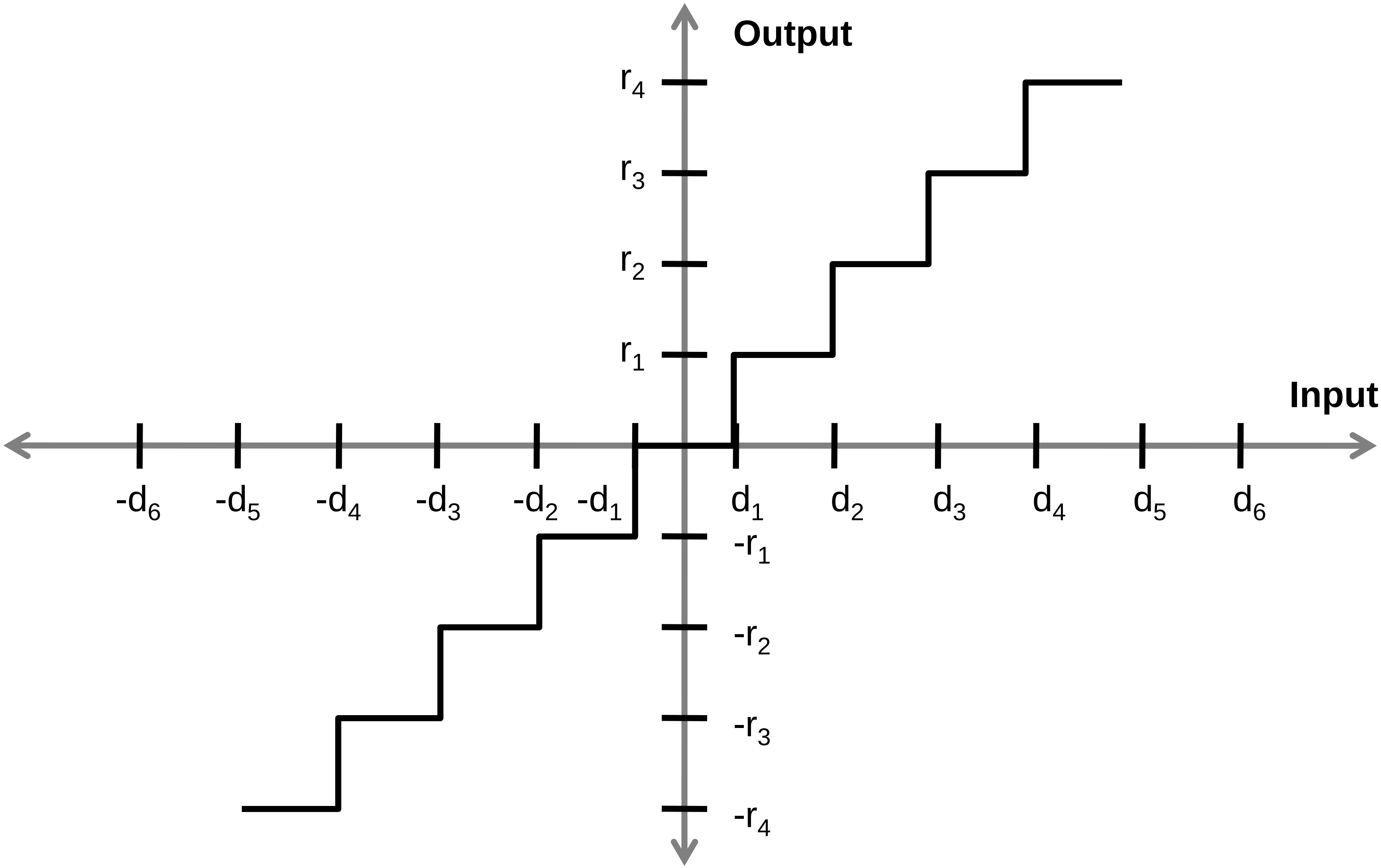


QUANTIZATION

Quantization

- All of the techniques discussed up to this point have been lossless
  - amount of compression that can be achieved is limited
- Solution: remove information that is not important to the subjective quality of the video material
- Use of a **quantizer**

Linear Quantizer transfer function



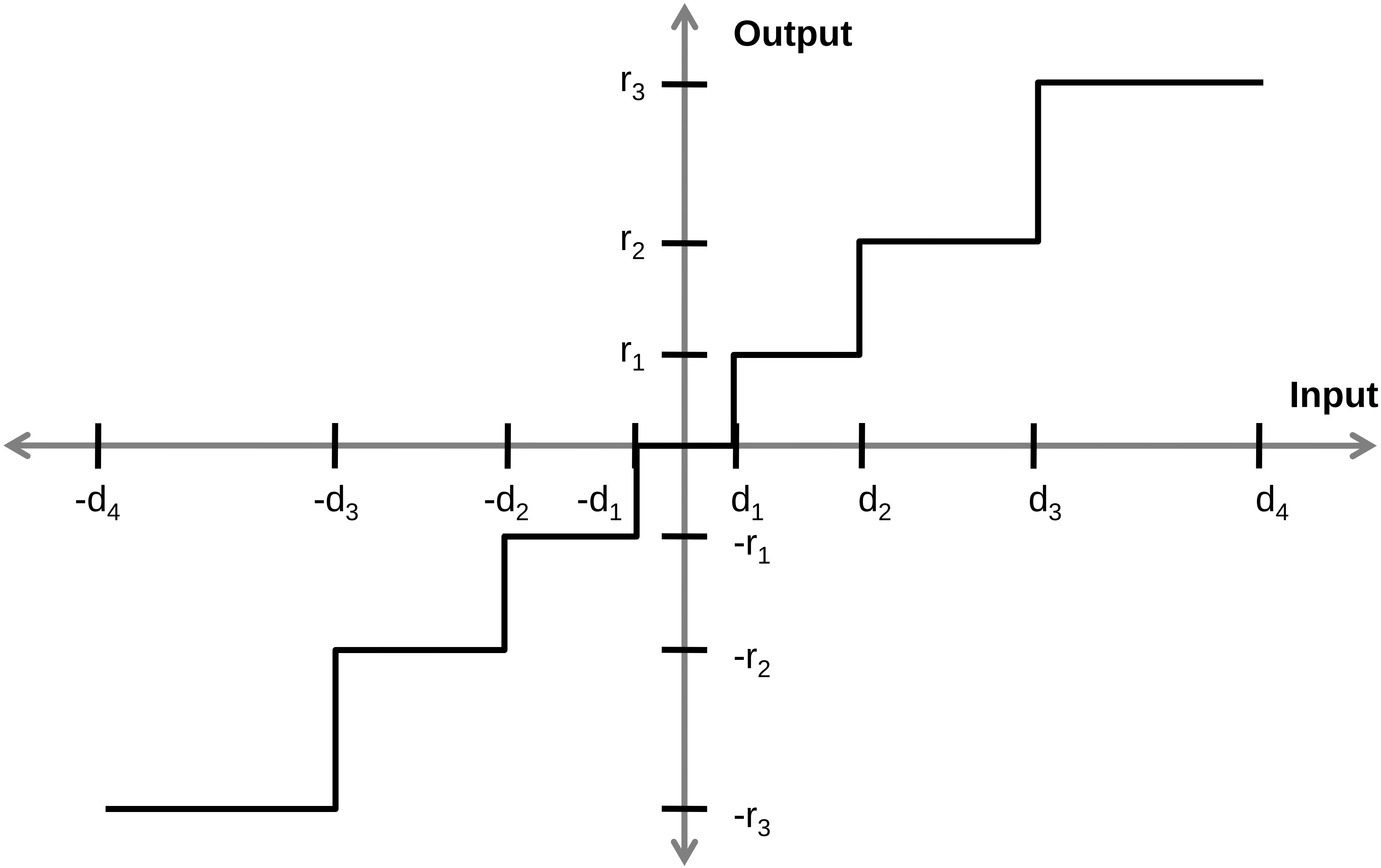
Non-linear Quantizer

- Decision level *d*
- Reconstruction level *r*

$$d_k = \frac{r_k + r_{k+1}}{2} \qquad r_k = \frac{\int_{d_{k-1}}^{d_k} x * p(x)dx}{\int_{d_{k-1}}^{d_k} p(x)dx}$$

$$p(x) = \frac{\lambda}{2} e^{-\lambda|x|}$$

Non-linear Quantizer transfer function

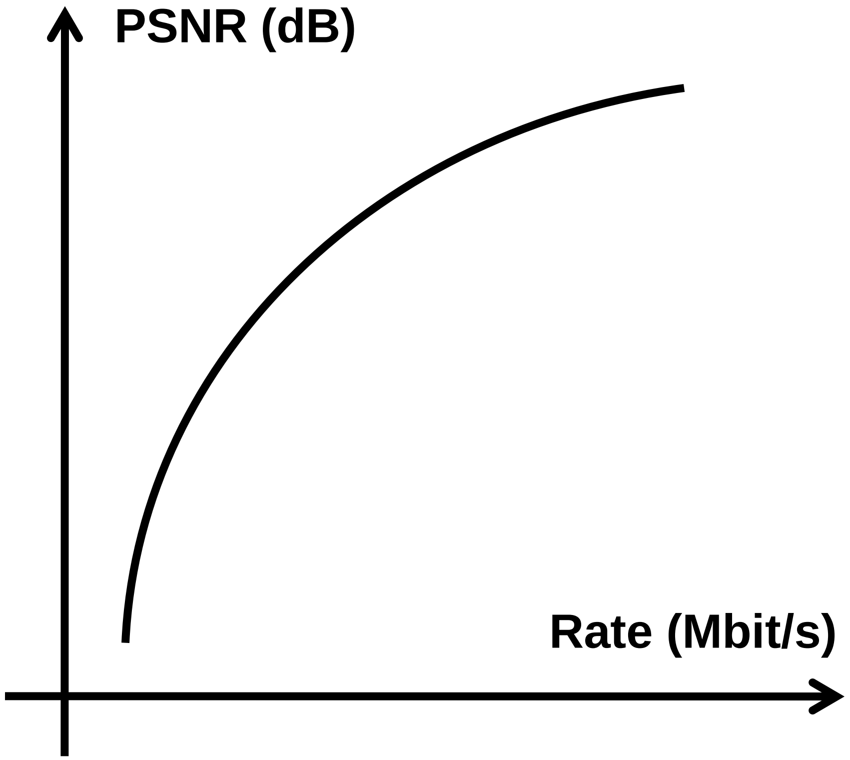


Rate-distortion curves

- Peak signal-to-noise ratio (PSNR)
  - measure the quality of a reconstructed video service

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{i=1}^{M}\left(x_{i,j} - \hat{x}_{i,j}\right)^2$$

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



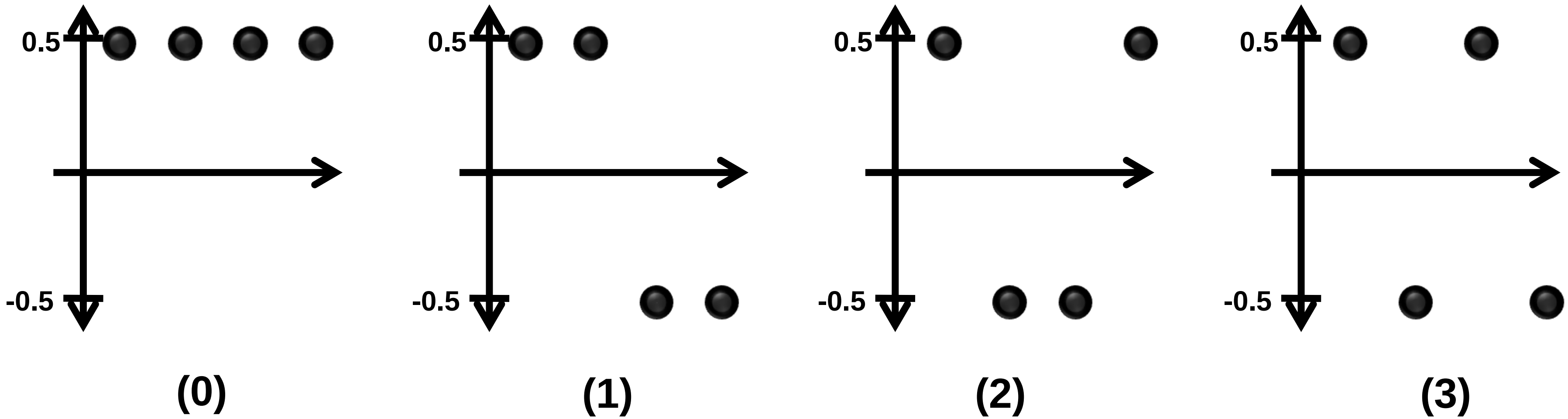
TRANSFORM CODING

The DCT (discrete cosine transform)

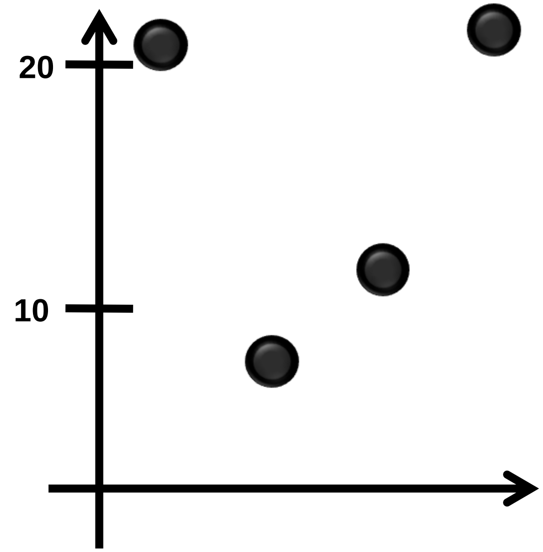
- The **DCT** (discrete cosine transform) is a particular case of the Fourier transform
- It decomposes the signal into only one series of harmonic cosine functions in phase with the signal, which reduces by half the number of coefficients necessary to describe the signal compared to a Fourier transform.

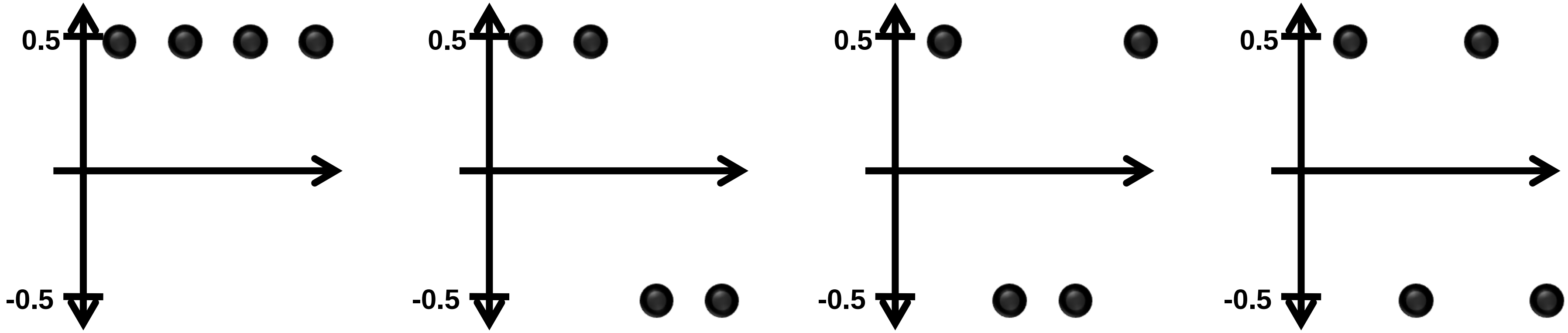
DCT coefficients 65

- (0) : The first coefficient gives the average value of the pixels and it is called the **DC coefficient**
- (1-3) : The remaining coefficients give information about the variation of the pixel values and are referred to as **AC coefficients**



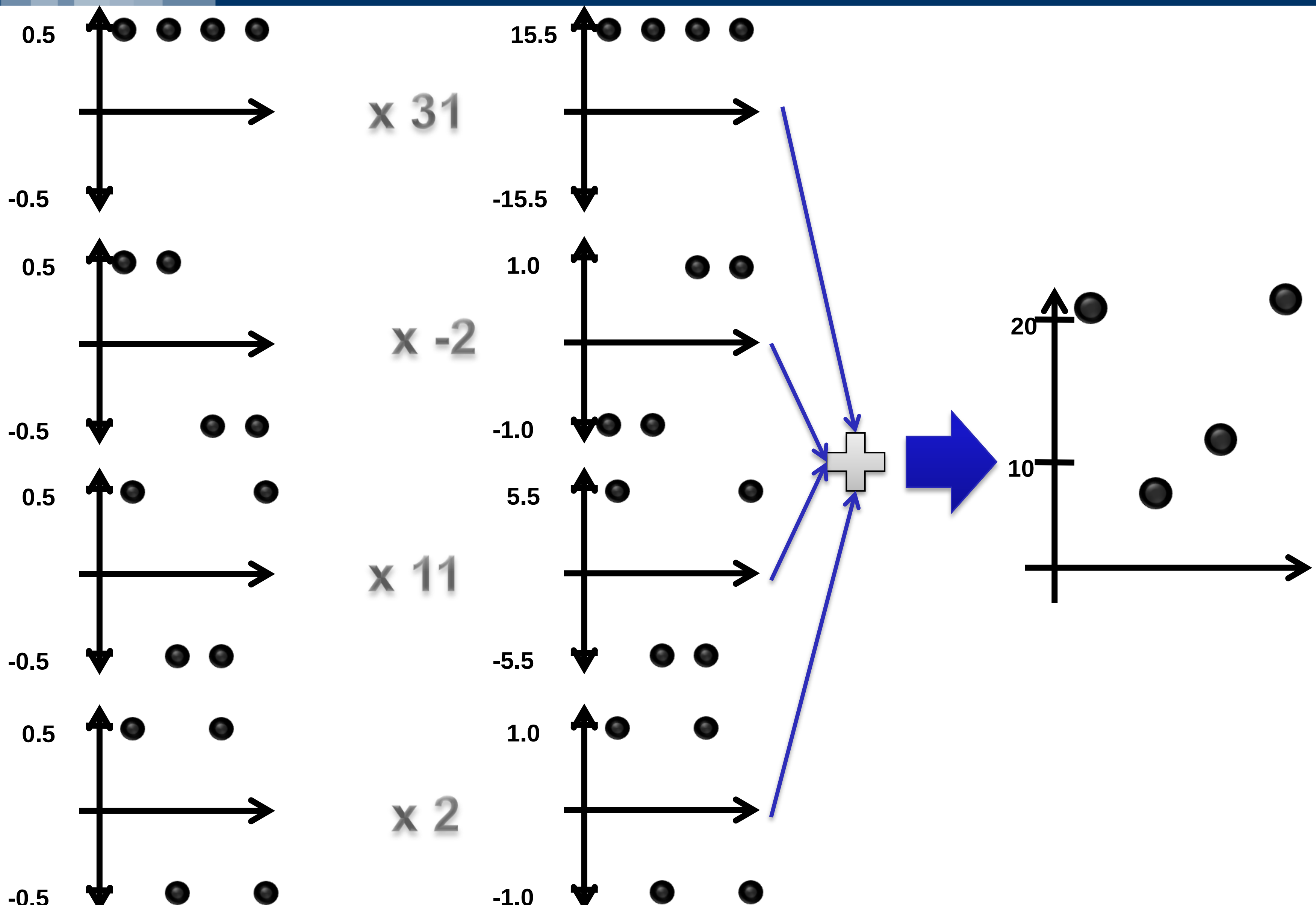
Example





(0) (1) (2) (3)

Example (2)



DCT and pictures

- **Pictures**
  - the original signal is a sampled **bidimensional signal → bidimensional DCT** (horizontal and vertical directions)
    - transform the luminance (or chrominance) discrete values of a block of N ×N pixels into another block (or matrix) of N ×N coefficients representing the amplitude of each of the cosine harmonic functions.

Example: matrix of coefficient

143	118	136	114	120	112	129	134
131	121	126	143	134	123	124	133
147	127	134	136	113	111	140	129
151	129	132	139	115	116	134	131
150	124	112	134	126	129	132	125
134	93	116	136	124	137	138	131
144	142	119	147	162	149	123	132
143	117	103	125	122	109	128	132

8x8 Pixel Block

DCT

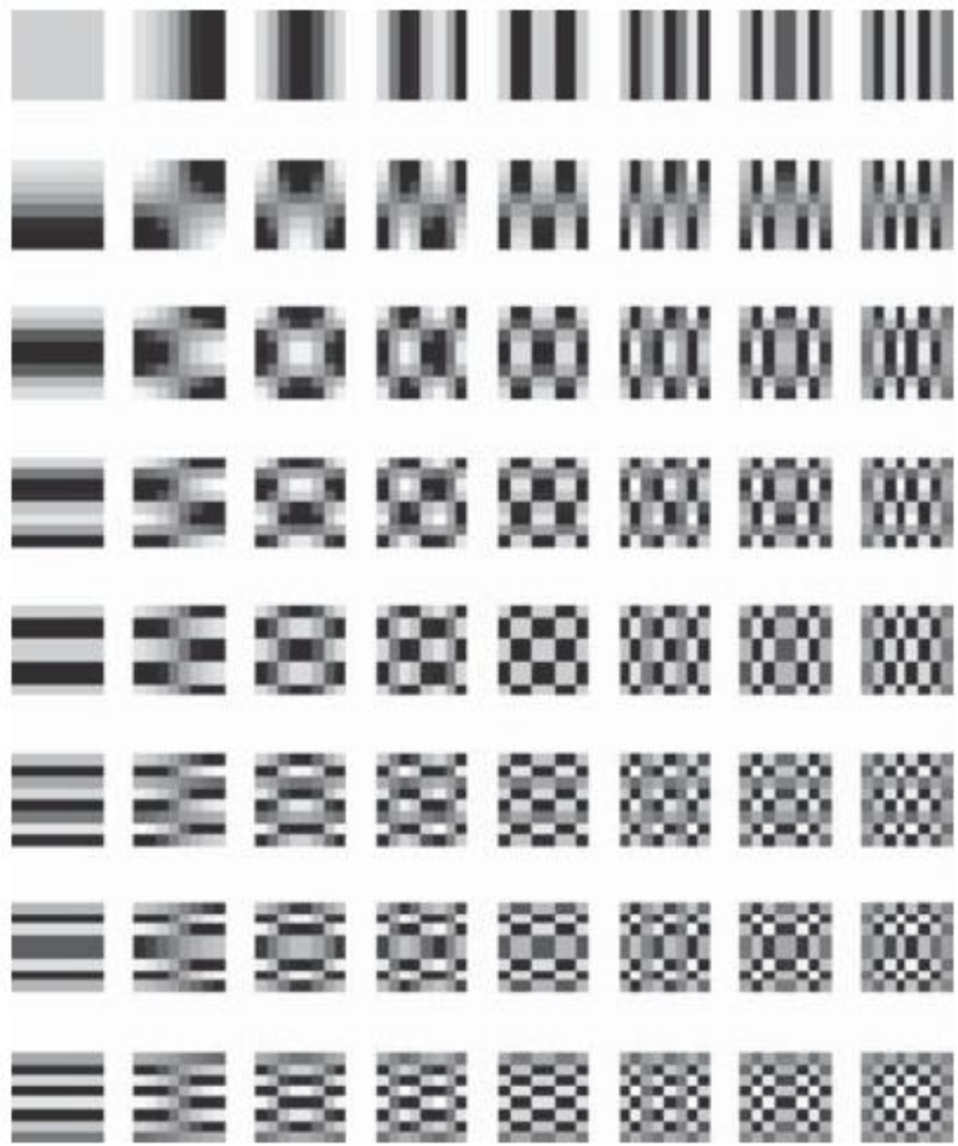
Horizontal Frequency

Vertical Frequency

0	1	2	3	4	5	6	7	
1033	7	20	12	37	31	-2	0	0
-4	12	10	-26	-12	8	7	13	1
-8	-4	-1	-2	6	18	0	9	2
0	-16	-11	14	-1	-3	15	2	3
-17	11	26	-3	-1	4	-3	7	4
20	11	0	-2	-7	-8	3	4	5
24	-9	20	8	-6	24	6	11	6
11	11	2	8	-3	-10	-7	0	7

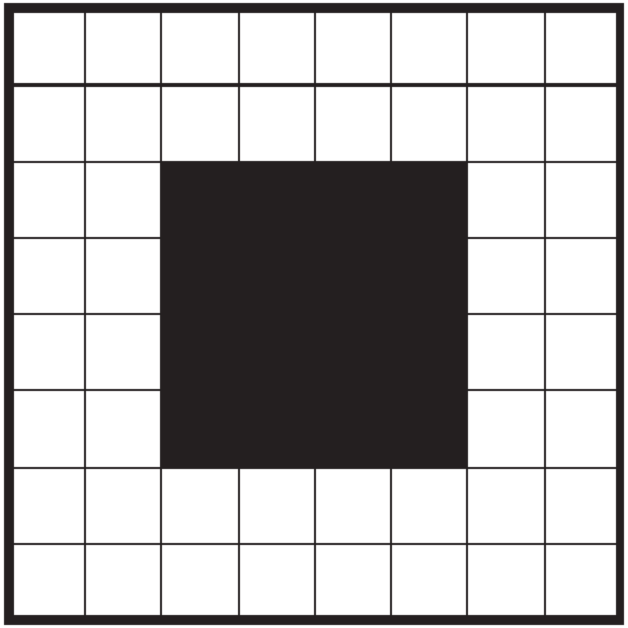
Matrix of 8x8 Coefficients

Basis vectors for two-dimensional DCT



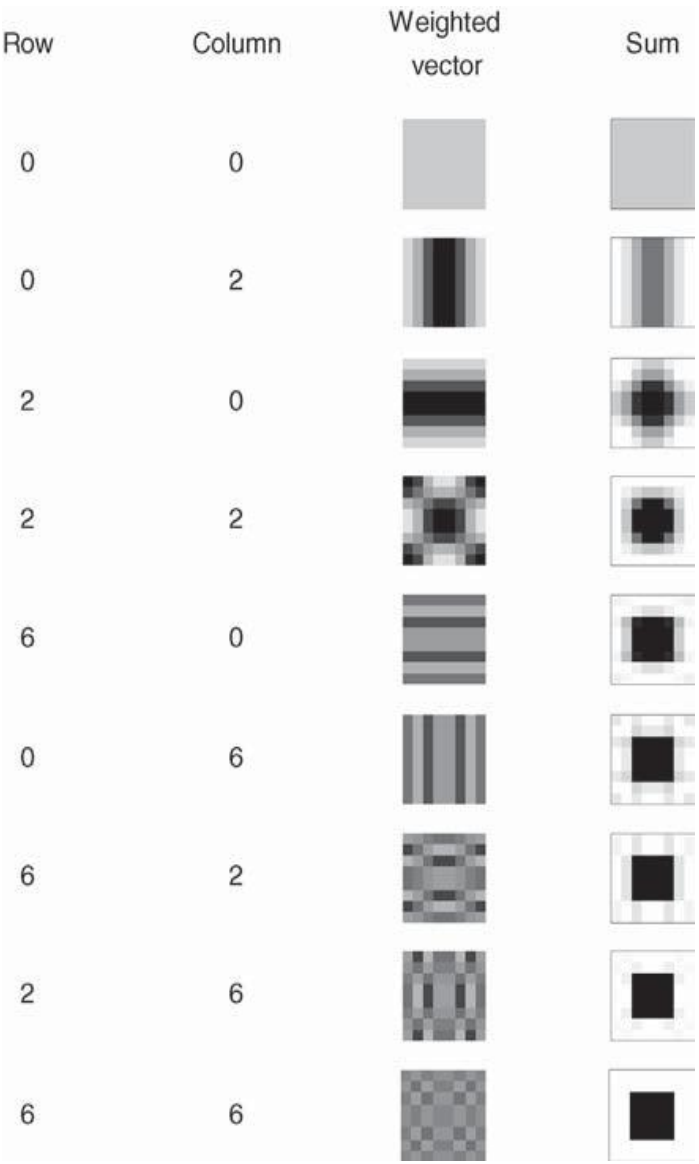
Example

Example



$$\begin{bmatrix} 255 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \\ 255 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \\ 255 & 255 & 0 & 0 & 0 & 0 & 255 & 255 \\ 255 & 255 & 0 & 0 & 0 & 0 & 255 & 255 \\ 255 & 255 & 0 & 0 & 0 & 0 & 255 & 255 \\ 255 & 255 & 0 & 0 & 0 & 0 & 255 & 255 \\ 255 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \\ 255 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \end{bmatrix}$$

Example (2)



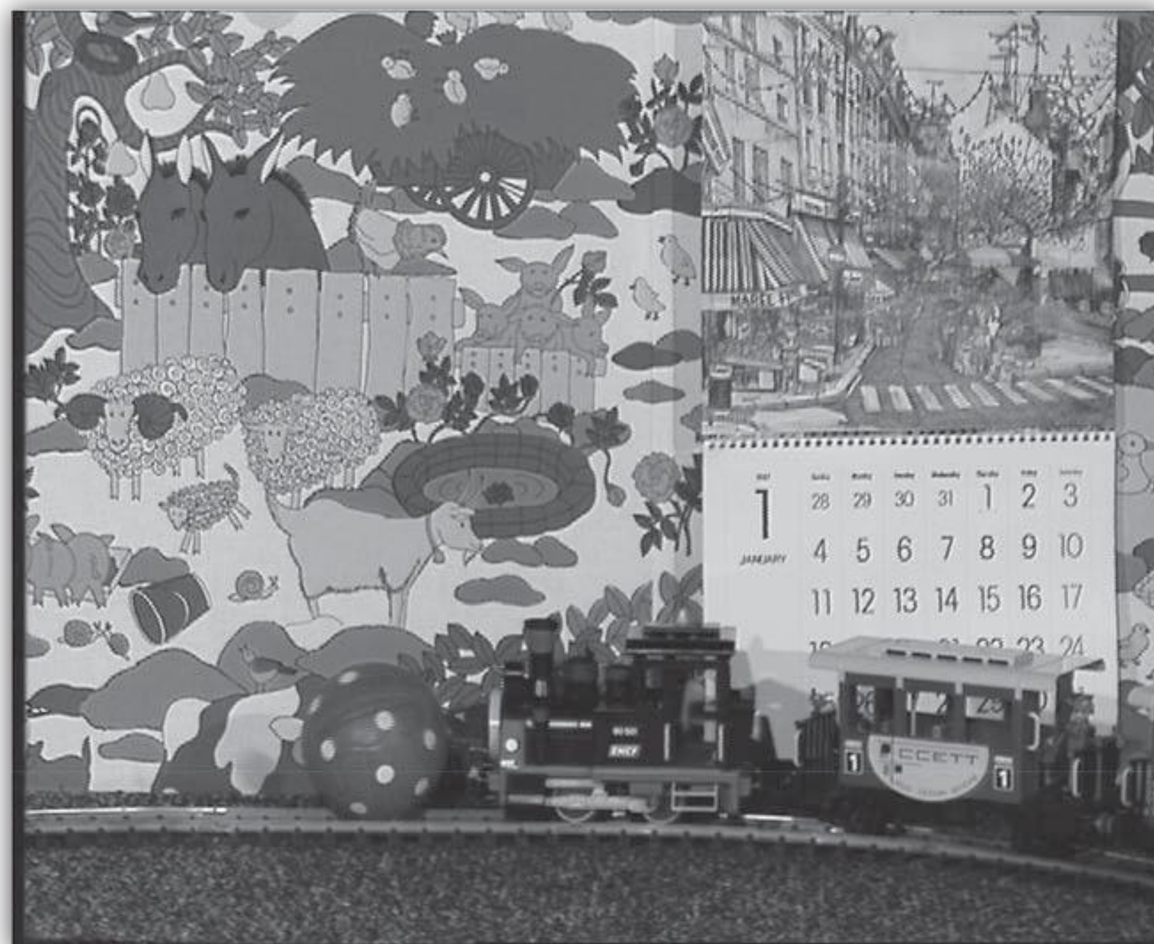
Resulting matrix of coefficients

$$\begin{bmatrix} +1530 & 0 & +471 & 0 & 0 & 0 & -195 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ +471 & 0 & -435 & 0 & 0 & 0 & +180 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -195 & 0 & +180 & 0 & 0 & 0 & -75 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

DCT – Thresholding - Quantization

- **DCT**: calculation of the matrix of coefficients
- **Thresholding**: elimination of the useful value corresponding to the high frequencies
- **Quantization**: quantization of the remaining values

Original image

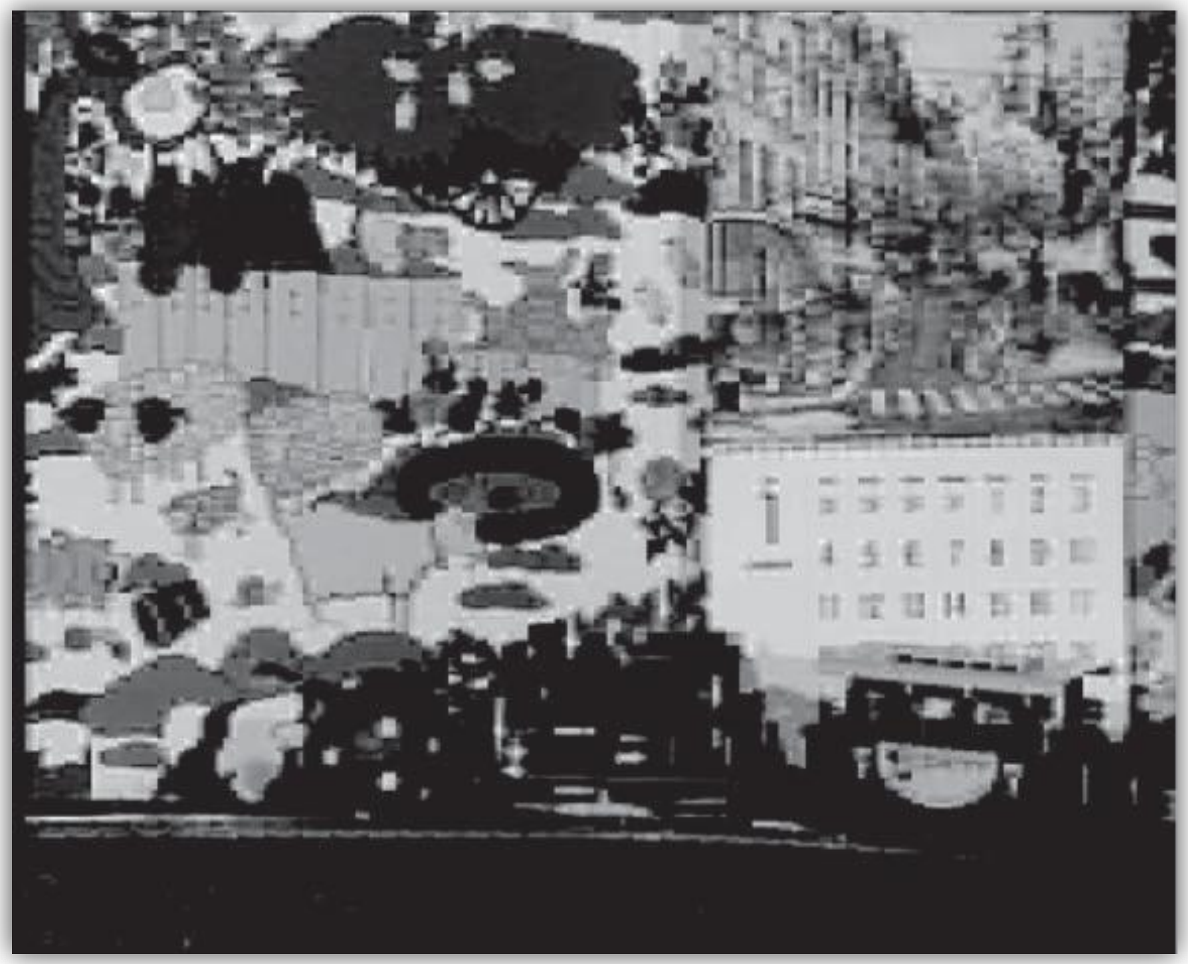


Example (2)

DCT using top 4x4 coefficient

Example (3)

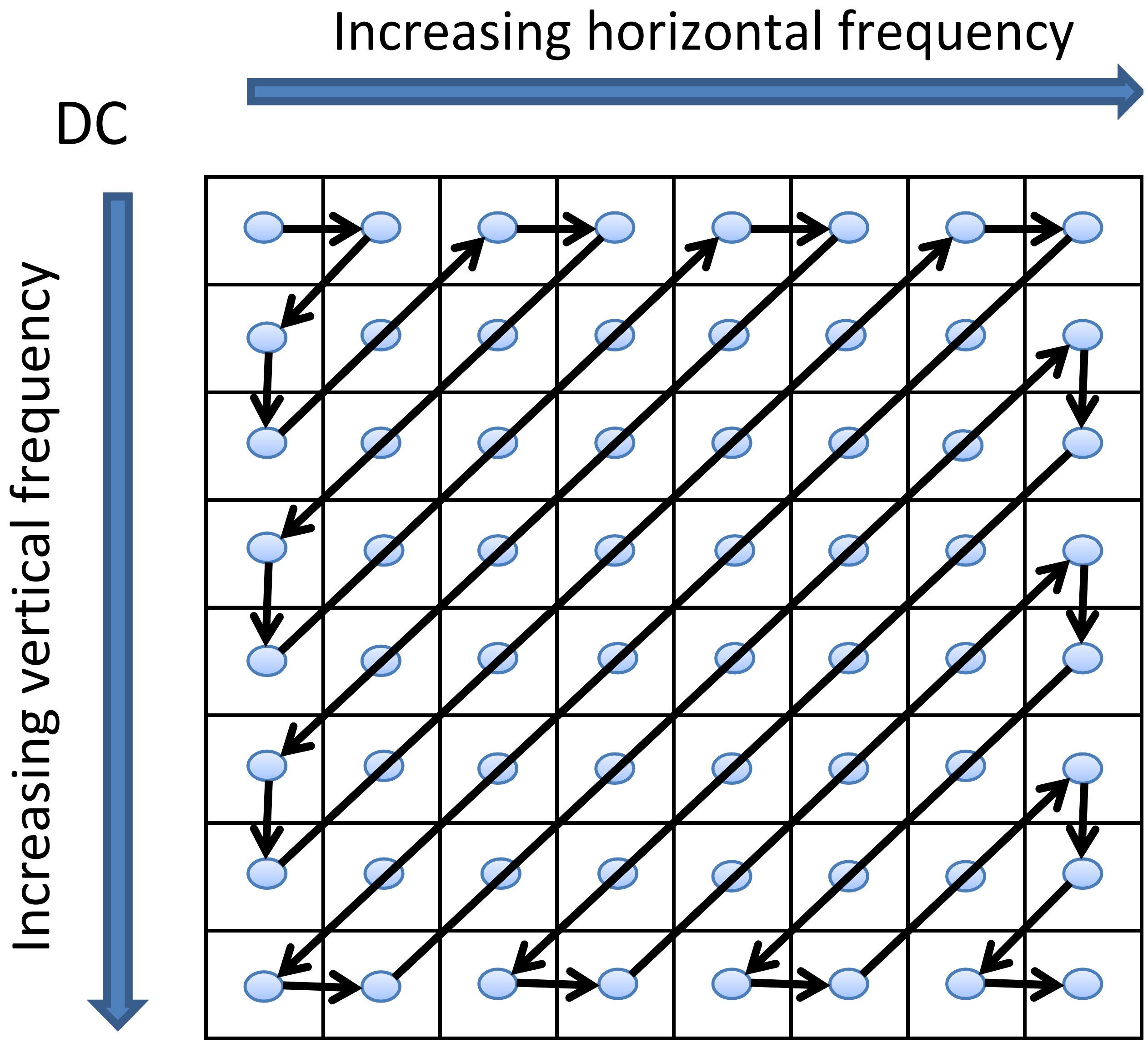
DCT using top 2x2 coefficient



Example (4)

DCT using top 1x1 coefficient

Zig-Zag scanning



Block of quantized DCT coefficients

$$
\begin{bmatrix}
+38 & 0 & +4 & -2 & 0 & 0 & 0 & 0 \\
-5 & +1 & 0 & +1 & 0 & 0 & 0 & 0 \\
+7 & +2 & -2 & +1 & 0 & 0 & 0 & 0 \\
+2 & -1 & +1 & -1 & 0 & 0 & 0 & 0 \\
-2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

Transimitted bitstream

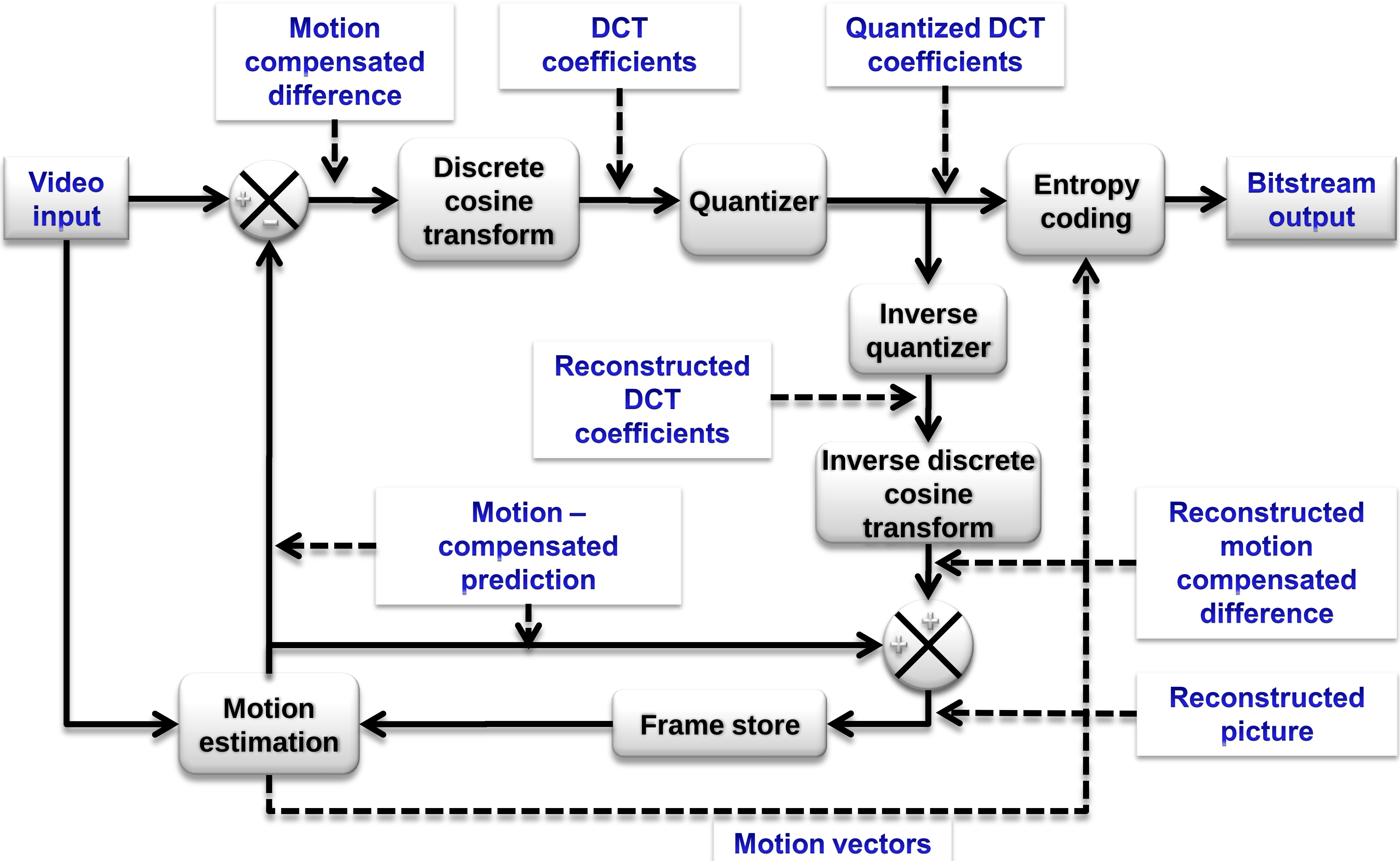
- Coefficients after zig-zag scanning:
  +38, 0, –5, +7, +1, +4, –2, 0, +2, +2, –2, –1, –2, +1, 0, 0, 0, +1, +1, 0, 0, 0, 0, 0, –1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
- Coefficients after coding into (run,coefficient) pairs:
  (0,+38) (1,–5) (0,+7) (0,+1) (0,+4) (0,–2) (1,+2) (0,+2) (0,–2) (0,–1) (0,–2) (0,+1) (3,+1) (0,+1) (5,–1)
- Huffman CodeWords used to represent the quantized DCT coefficients:
  CodeWord(0,+38), CodeWord(1,-5), CodeWord(0,+7), CodeWord(0,+1), CodeWord(0,+4), CodeWord(0,-2), CodeWord(1,+2), CodeWord(0,+2), CodeWord(0,-2), CodeWord(0,-1), CodeWord(0,-2), CodeWord(0,+1), CodeWord(3,+1), CodeWord(0,+1), CodeWord(5,-1), CodeWordEOB.

DCT and video

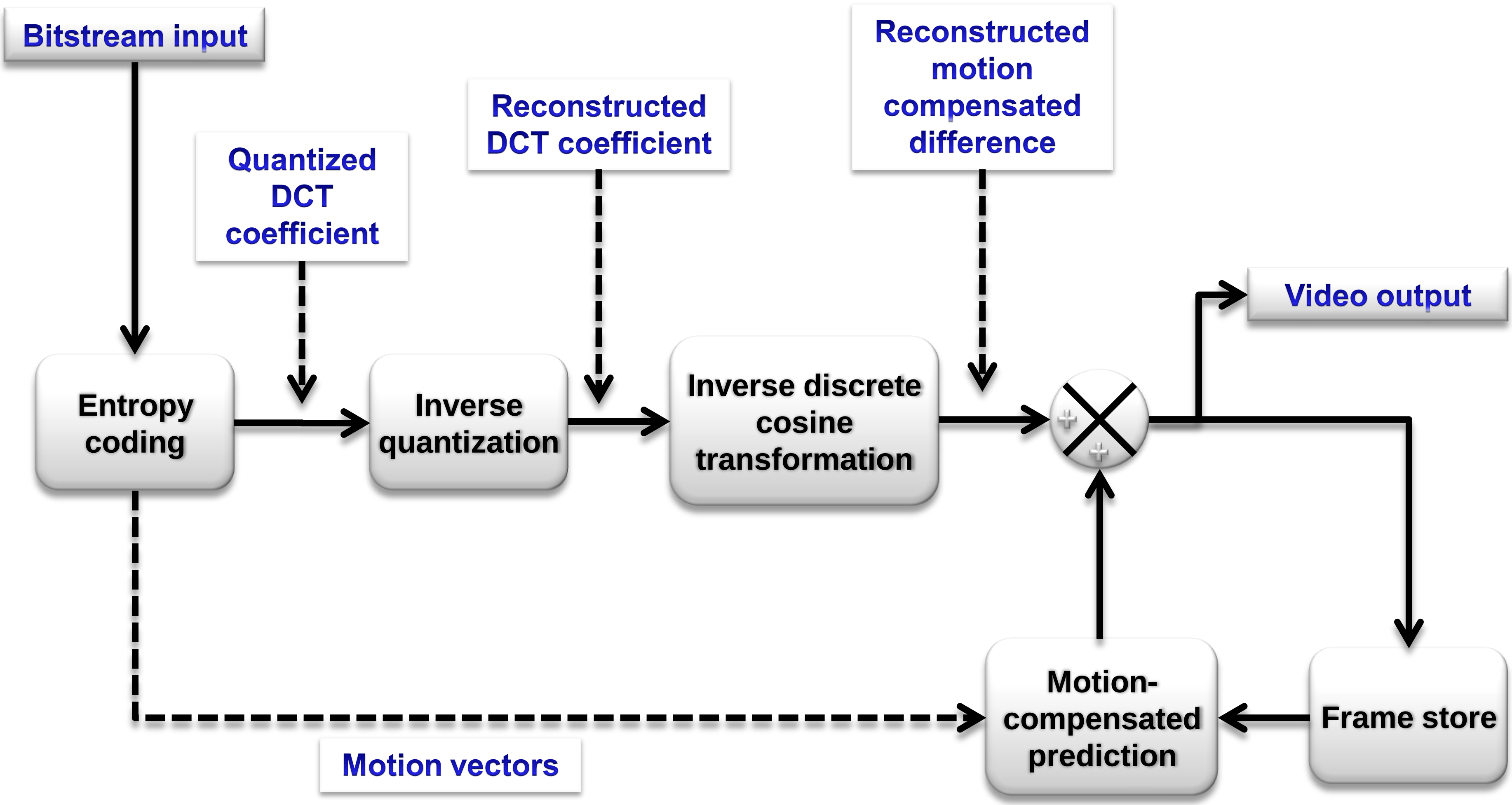
- **Videos**

  The DCT can also be used to code the prediction difference after motion-compensated prediction that we have seen previously.

Motion-compensated DCT enconder



Motion-compensated DCT decoder

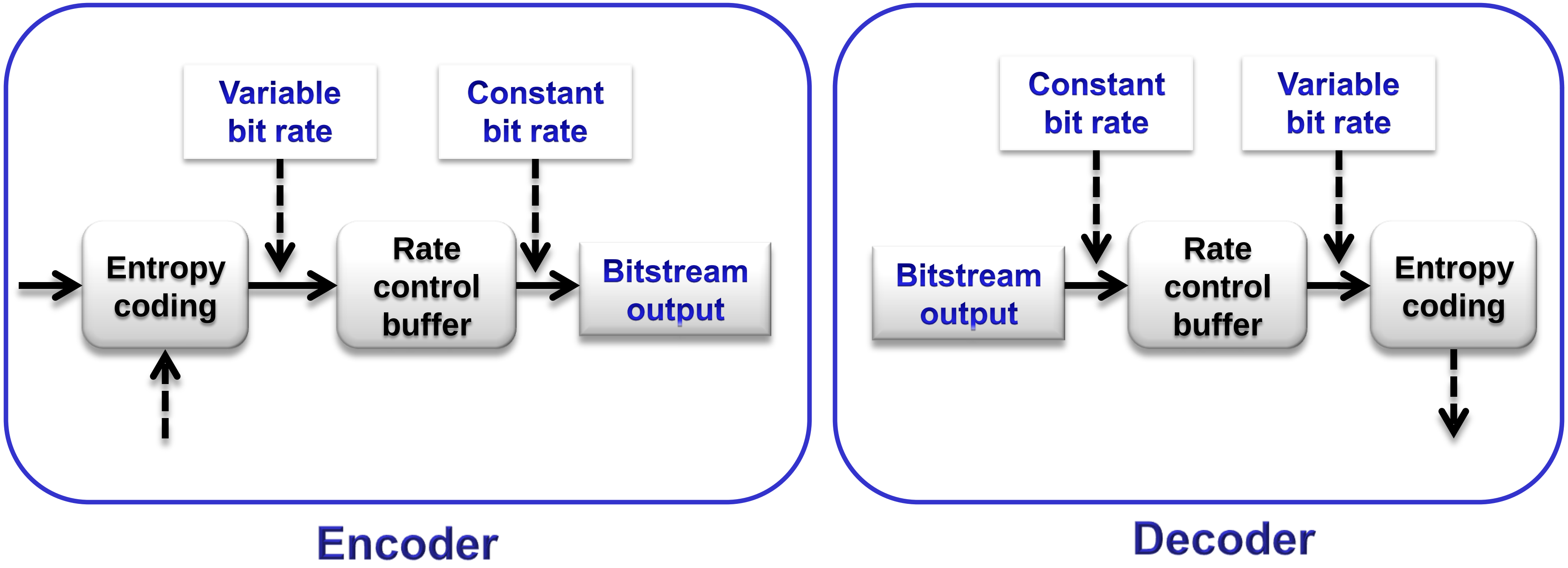


Rate control

- The output of the encoder is a variable rate bit stream (VBR).
  - different number of transformation from block to block
  - different number of bits required to encode each transform coefficient

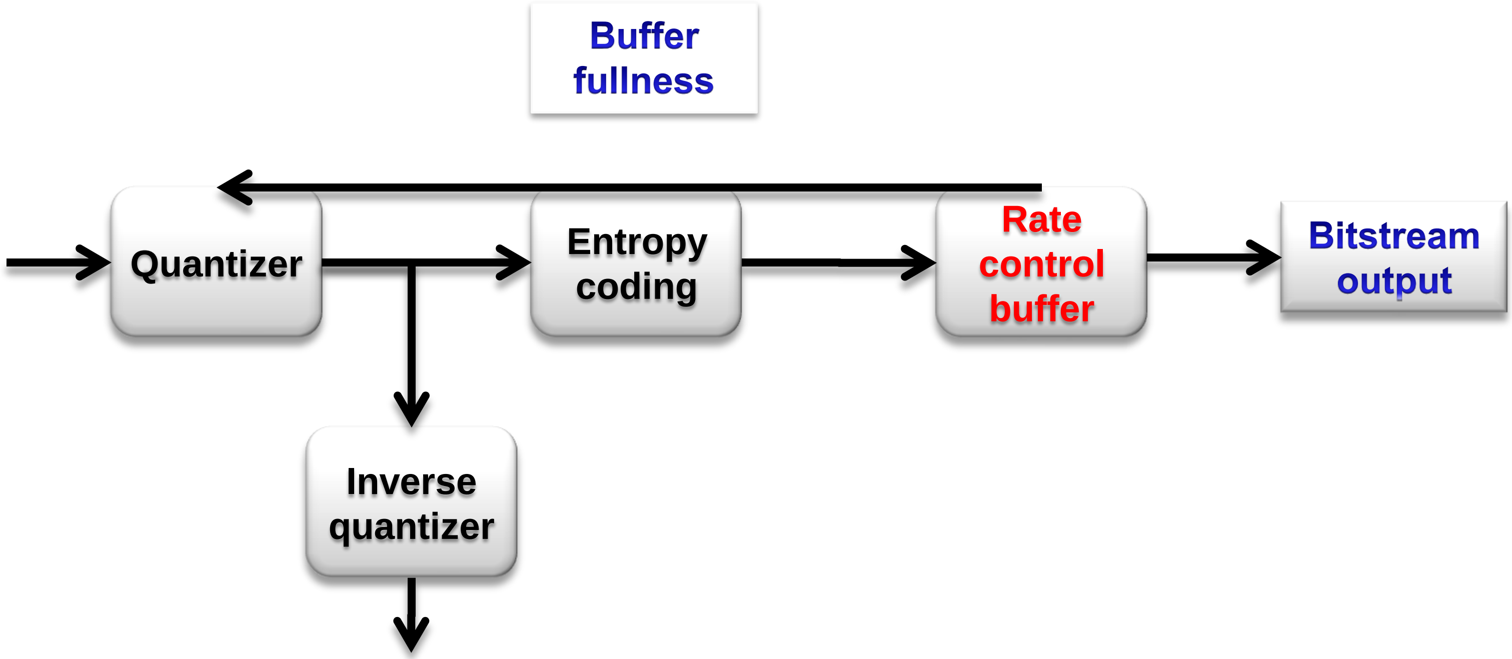
Rate control buffer

- The simplest way to achieve rate control is to introduce a buffer between the output of the entropy encoder and the transmission channel



Buffer overflow

- The rate control buffer needs to be sufficiently large to avoid becoming full (called buffer overflow).
- One simplistic approach would be to use **buffer fullness** in determining the **quantizer step size** to use at the encoder.



Rate controller and human visual system

- The effective exploitation of the characteristics of the human visual system is essential in a good rate controller.
  - *Frequency sensitivity*
  - *Visibility threshold*
  - *Spatial masking*
  - *Temporal masking-fast moving objects*
  - *Temporal masking-scene changes*
  - *Luminance masking of chrominance*

DIGITALIZATION

Introduction: digitize video signal

PROS

- Allows multiple copies without degradation in quality
- Special effect and better editing
- Permitting international exchange independent of the broadcast standard to be used for diffusion

CONS

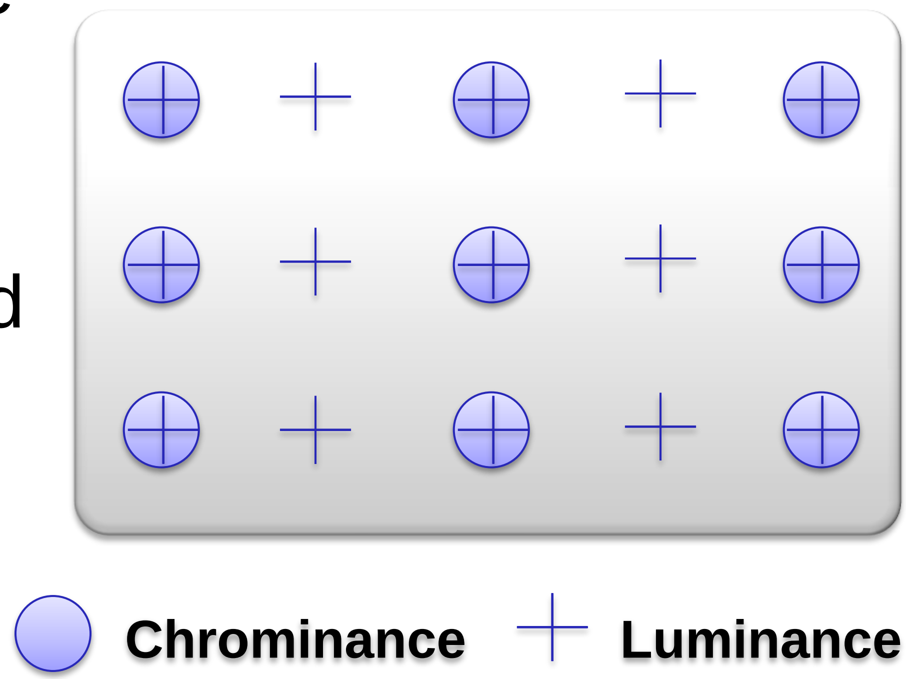
- Bit rate

Digitalization formats

- Shannon sampling theorem
  - $F_{max} = F_s/2$
  - CCIR prescribes a sampling frequency of 13,5 MHz
- 8 bits minimum required
- Bit Rate: 13,5x8 = 108 Mb/s

4:2:2 Format

- Active lines on image:
    - 480 for 525 line-systems
    - 576 for 625 line-systems
- 720 samples per line
- Y, C_b, C_r
    - Four Y samples for two C_b samples and two C_r samples
- High bit rate



Chroma subsampling notation

Luma horizontal sampling reference (originally, luma f_S *as multiple of 3 3⁄8 MHz*)

CB and CR horizontal factor **(relative to first digit)**

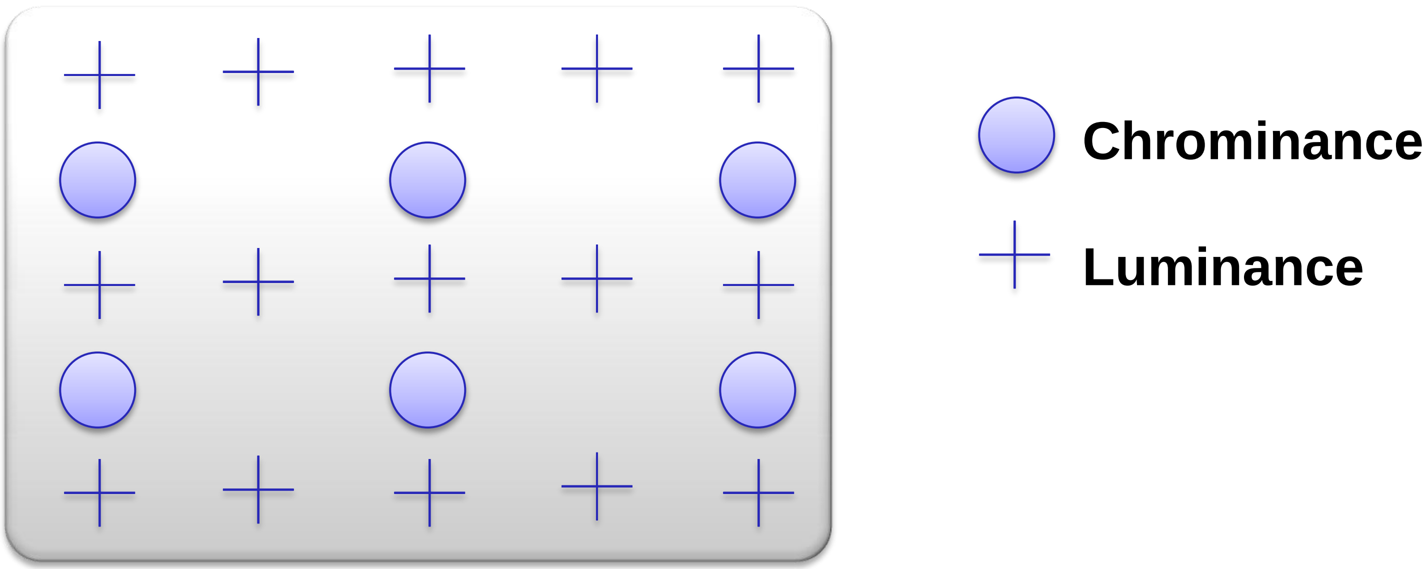
Same as second digit; or zero, indicating ***CB and CR*** **are subsampled 2:1 vertically**

If present, same as luma digit; indicates ***alpha (key) component***

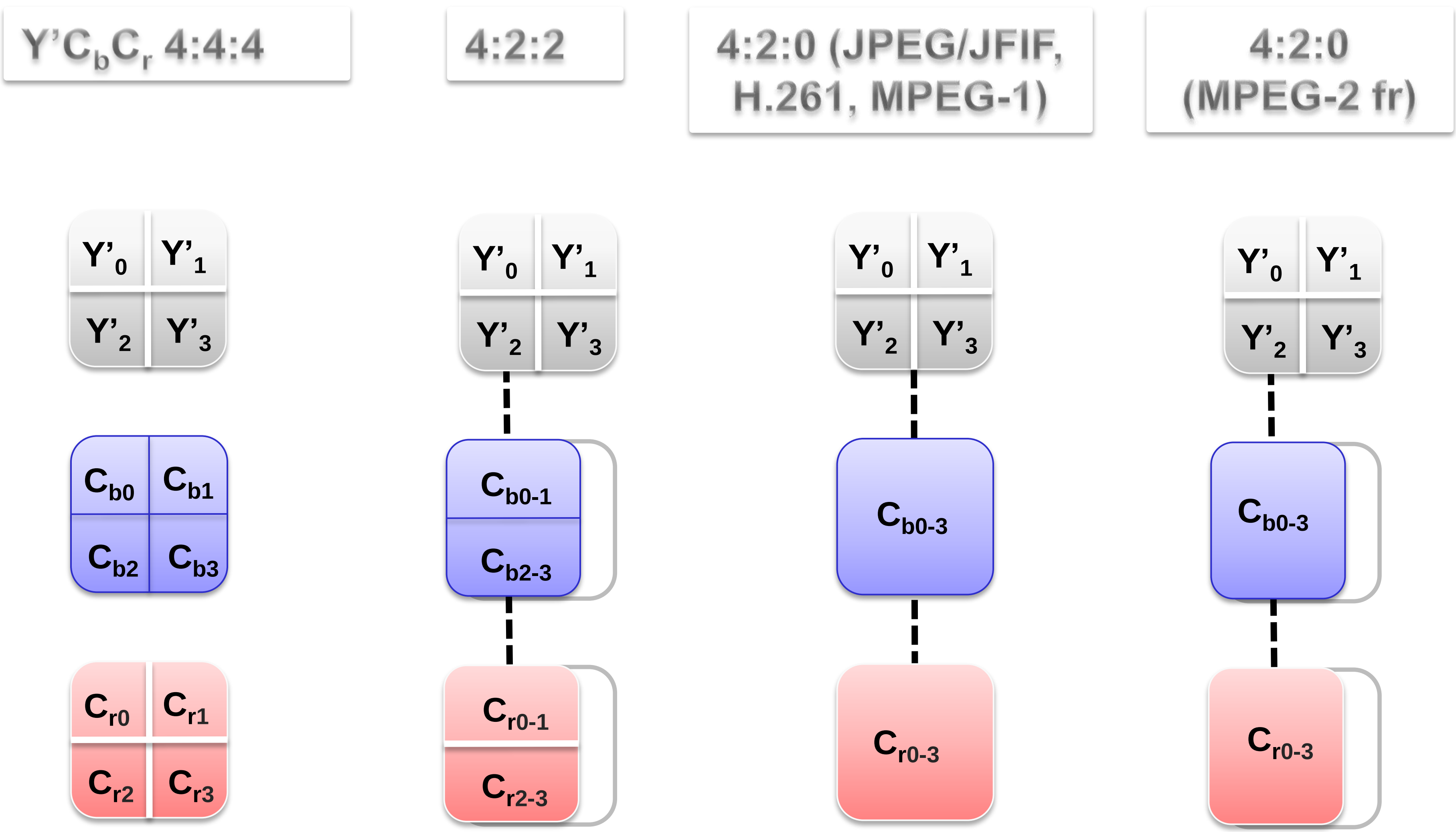
4:2:2:4

4:2:0

- Use the same chroma samples for two successive lines
- Same resolution of 4:2:2
  - luminance resolution: 720×576 (625 lines) or 720×480 (525 lines)
  - chrominance resolution: 360×288 (625 lines) or 360×240 (525 lines).



Chroma subsampling




4:2:0
4:2:0
4:2:0

MPEG - 1

MPEG Compression

- Moving Pictures Experts Group
  - Industry association
- MPEG technology is a compression standard
  - Used by satellite, cable and terrestrial TV systems

Summary of MPEG formats

Mpeg Format	Description
MPEG-1	The MPEG-1 file format was originally developed in 1988 and was primarily used to compress video data at bit rates of 1.5 Mbps. MPEG-1 content is used for such services as DAB (Digital Audio Broadcasting). MPEG-1 is also the basis of the MP3 standard, which is widely used for music on the Internet.
MPEG-2	MPEG-2 builds on the powerful compression capabilities of the MPEG-1 standard. MPEG-2 is widely used in the delivery of broadcast-quality television and storing video content on DVDs. A number of international television standards are based on this compression format.
MPEG-4 (part 2)	MPEG-4, whose formal ISO/IEC designation is ISO/IEC 14496, was finalized in October 1998 and became an international standard in 2000. Part 2 of the standard is divided into a number of profiles that address the requirements of various video applications ranging from mobile phones to surveillance cameras.
MPEG-4 (part 10)	MPEG-4 Part 10 also called H.264/AVC is designed to deliver broadcast and DVD-quality video at minimum data rates.

MPEG-1

- Goal: allow storage on CD-ROM of live video and stereo sound
  - Maximum bitrate: 1.5 Mb/s
- Picture format: SIF
  - Resolution of a consumer video recorder
- Sound compression algorithm: MUSICAM
- **Not suitable for broadcast applications**

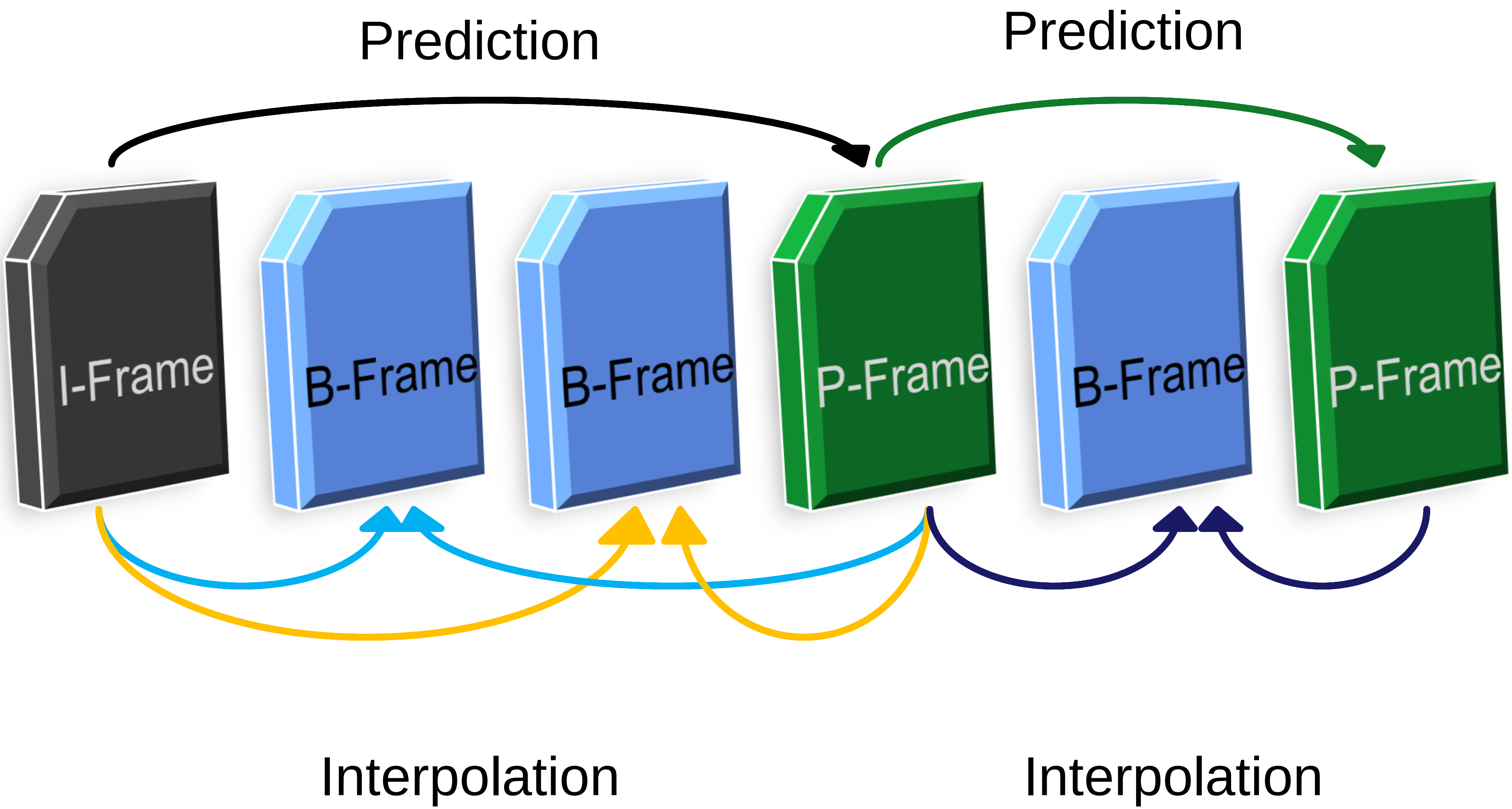
Parts of MPEG-1 standard

Part	Description
1	Addresses the problem of combining one or more data streams from the video and audio parts of the standard with timing information to form a single data stream.
2	Defines a coded representation that can be used for compressing both 525 and 625 lines to bit rates approximately 1.5 Mbps
3	Defines a coded representation that can be used for compressing both mono and stereo audio sequences
4	Defines how tests can be used to verify if bitstreams and decoders meet the requirements of the prior parts of the standard
5	A technical report that provides a full software implementation of the first three parts of the standard

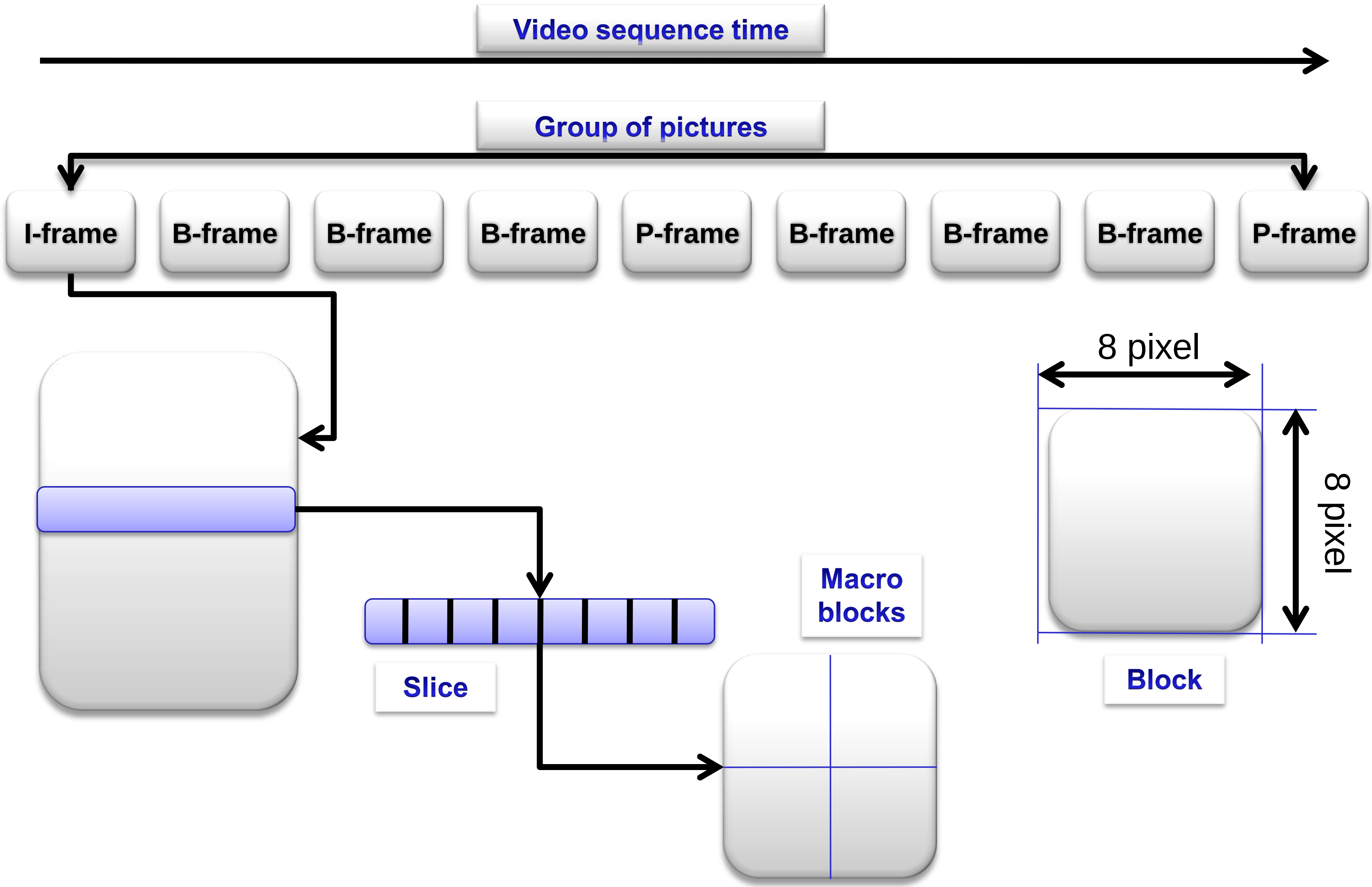
Video Compression Stages

- Subsampling
- Division of a picture frame into 8x8 pixel blocks
- DCT
- Quantization
- Macroblocks division (16x16 pixel)
- Slices
- Group of Pictures (GOP)

Types of frame (pictures)

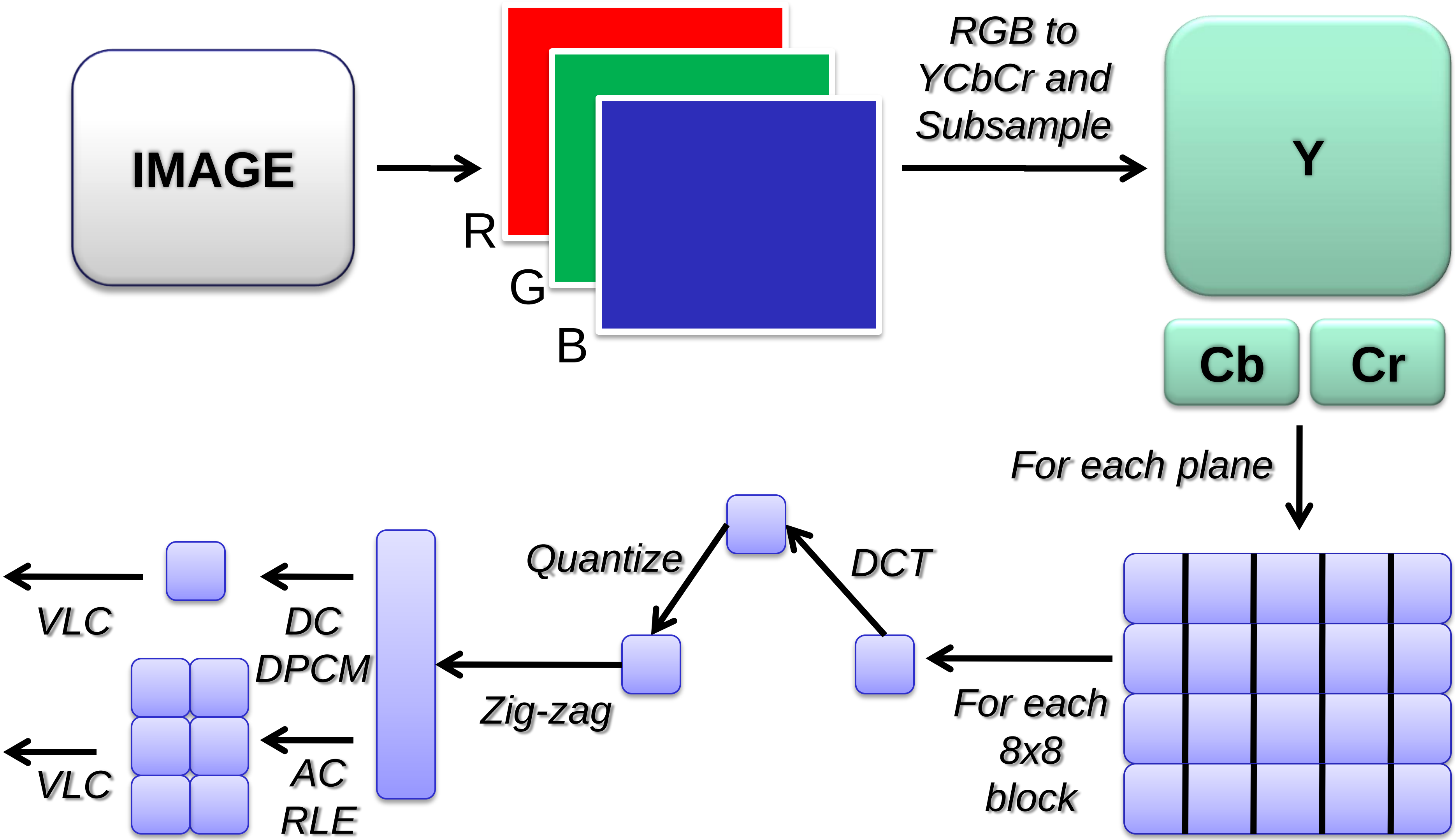


Summary

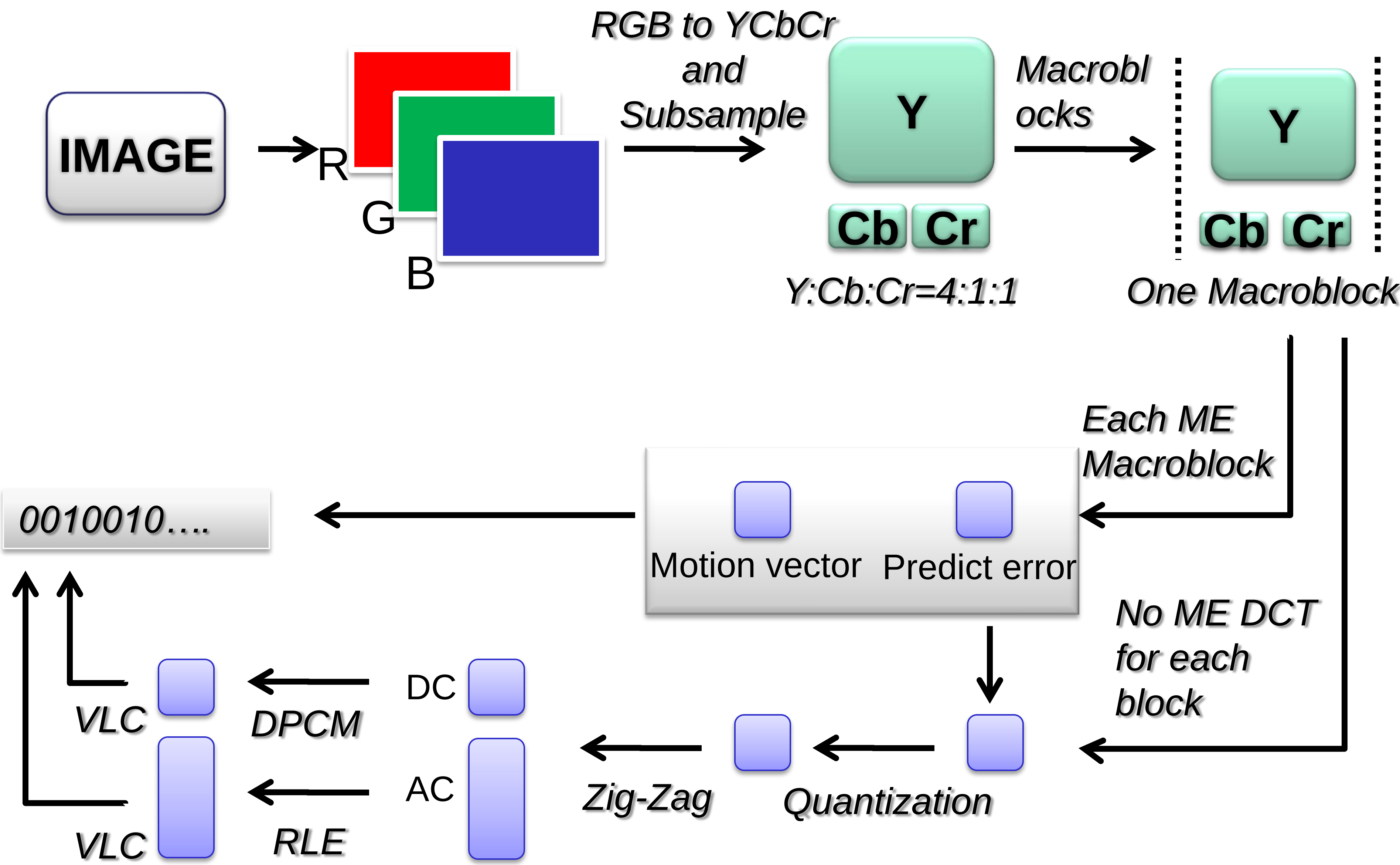


Video sequence time
Group of pictures
I-frame
B-frame
B-frame
B-frame
P-frame
B-frame
B-frame
B-frame
P-frame
8 pixel
8 pixel
Macro blocks
Slice
Block

I-Frame encoding flow chart



I-Frame encoding flow chart



MPEG - 2

MPEG-2

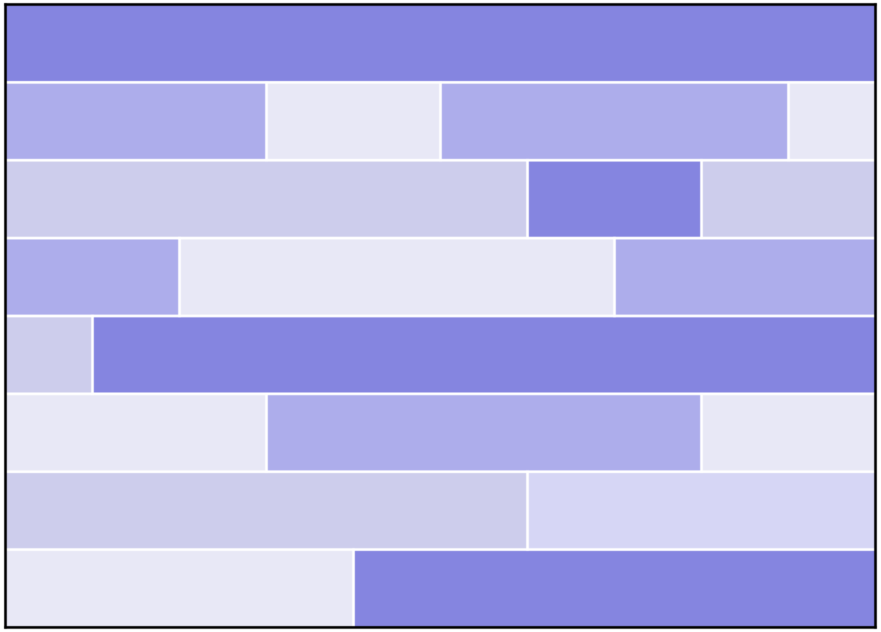
- MPEG-2 is build on the powerful compression capabilities of the MPEG-1 standard
- Dominant transport and compression standard for Digital TV
- Widely used to storing video content on DVDs
- Two categories:
  - Video Compression
  - Audio Compression

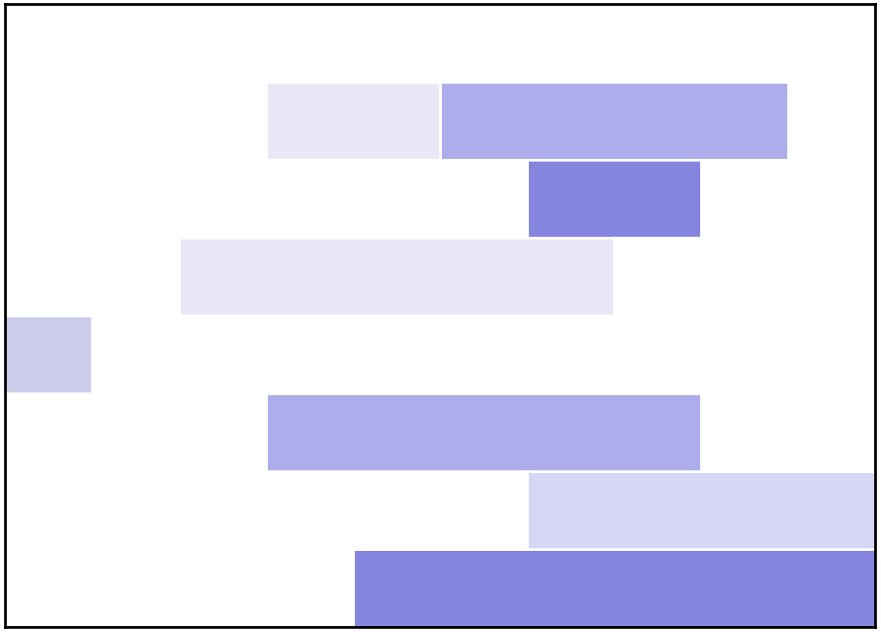
MPEG-2 Additions

- Different definition of the slices
- Defines nonscalable and scalable profiles
- Defines four levels of coding parameters
- Supports interlaced or progressive video sequences
- Changes meaning of aspect ratio information variable

The MPEG-2 slices in the most general case

- There is a small difference in the definition of the slices, as they do not necessarily cover the whole picture and are only made up of contiguous blocks of the same horizontal row





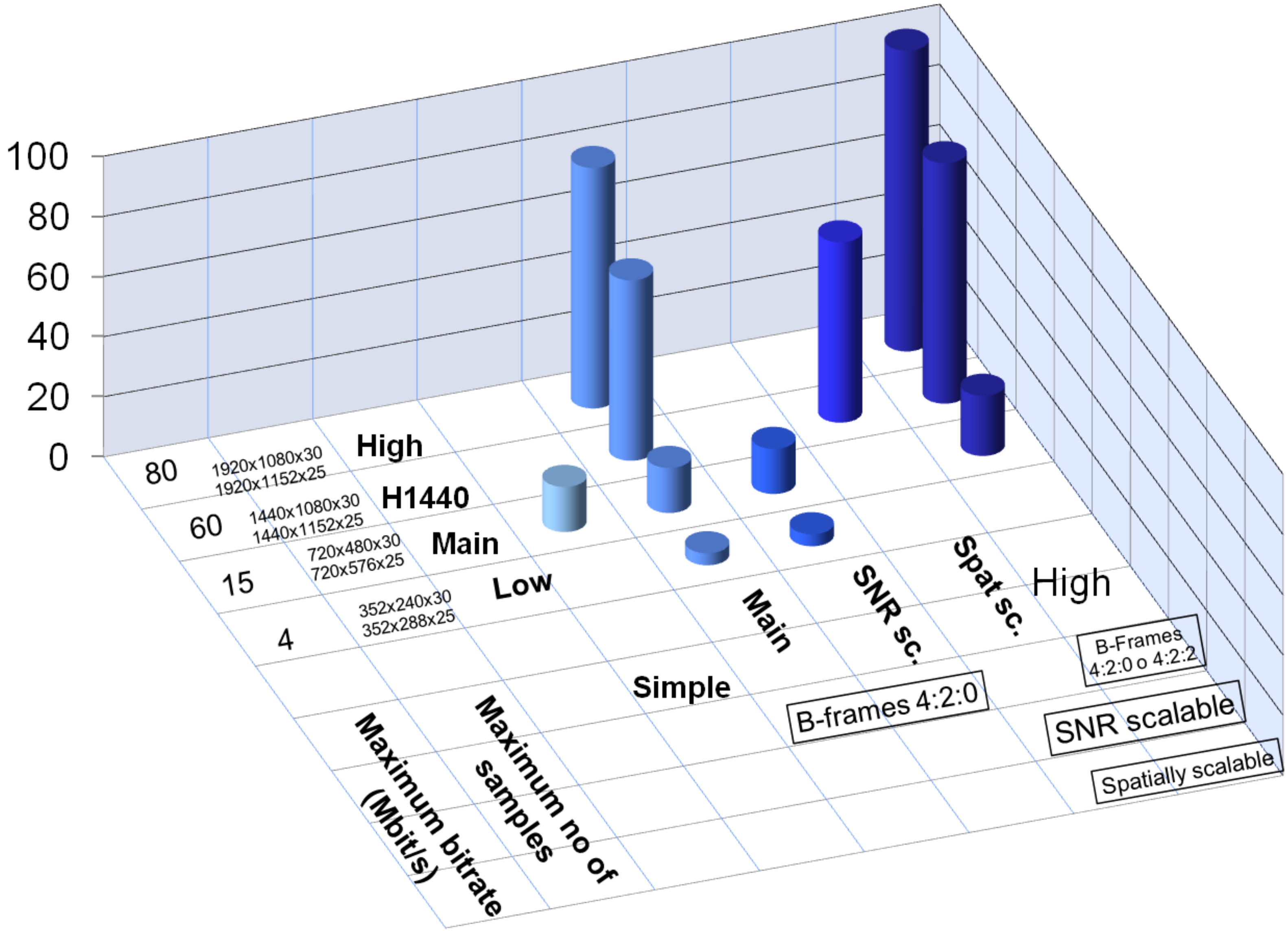
MPEG-2 Profiles

- A profile defines a subset of the overall specification
  - Simple
  - Main profile
  - SNR
  - Spatial
  - High profile
- Profiles are divided in levels

MPEG-2 Levels

Parameter	Low level	Main Level	High-1440 level	High-level
Horizontal size (pixels)	352	720	1440	1920
Vertical size (pixels)	288	576	1152	1152
Picture rate (Hz)	30	30	60	60
Bit rate (Mbit/s)	4	15	60	80
Bits/picture (kbits)	167	626	2503	3337

MPEG-2 Levels and profiles



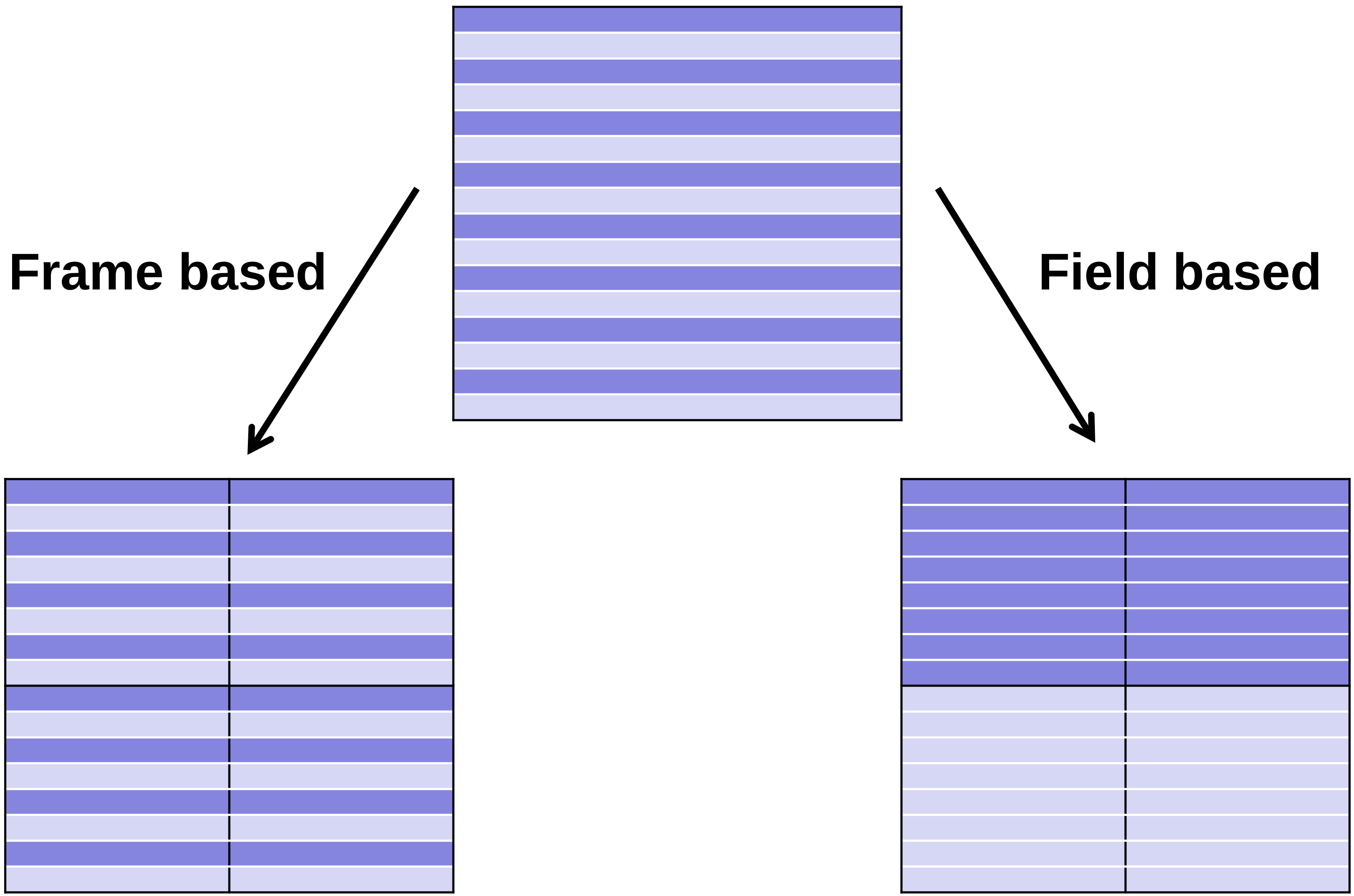
Parts of MPEG-2

- Part 1. Multiplexing
- Part 2. Profiles
- Part 3. Multi-channel extension of audio, compatible with MPEG-1
- Part 4. Test
- Part 5. Sample source code
- Part 6. VCR control (pause, rewind, fast forward, ...)
- Part 7. Multi-channel extension of audio, not compatible with MPEG-1
- Part 8. Dismissed
- Part 9. Control signals between set-top boxes and headend servers

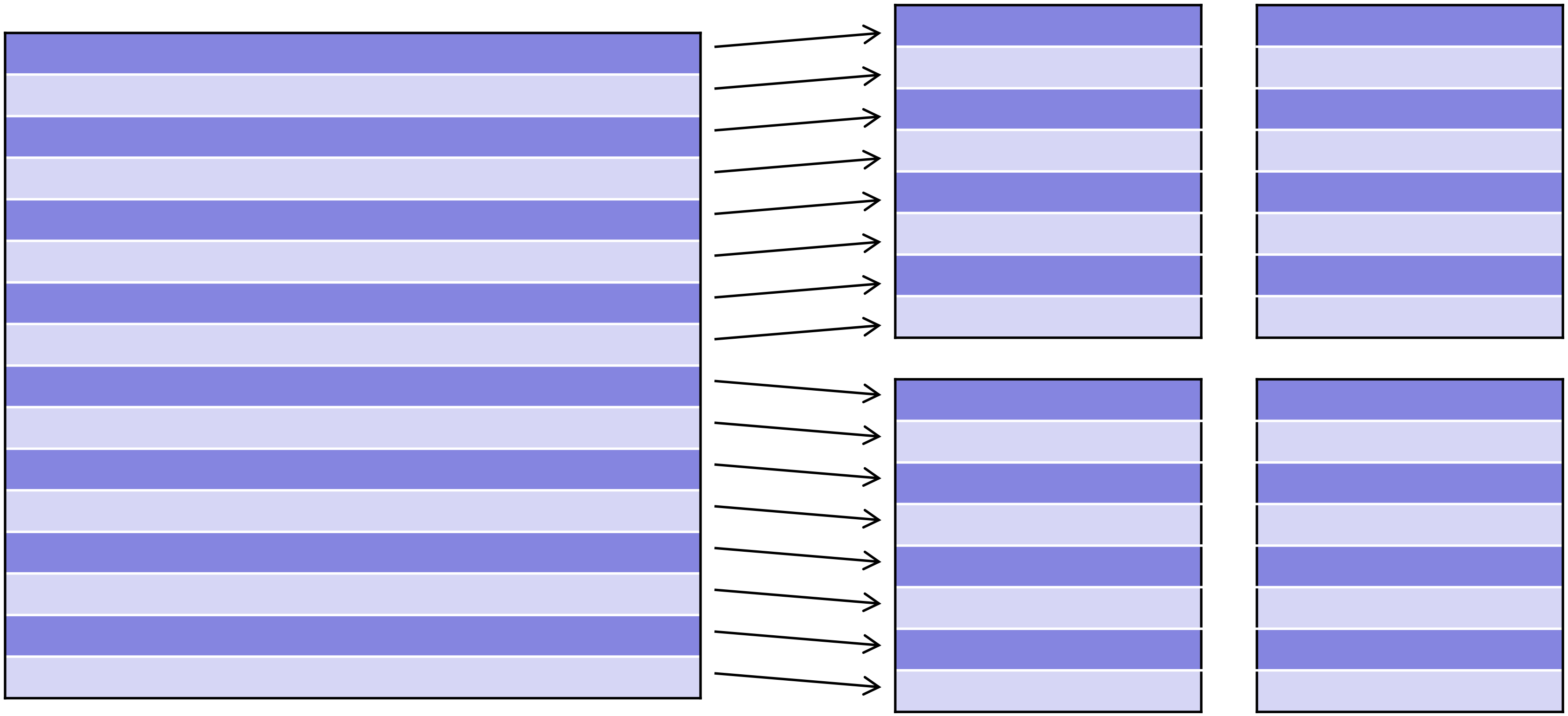
MPEG-2 Processing of interlaced pictures

- For the intra coding of interlaced pictures, MPEG-2 permits one to choose between two image structures:
  - Frame Based
  - Field Based

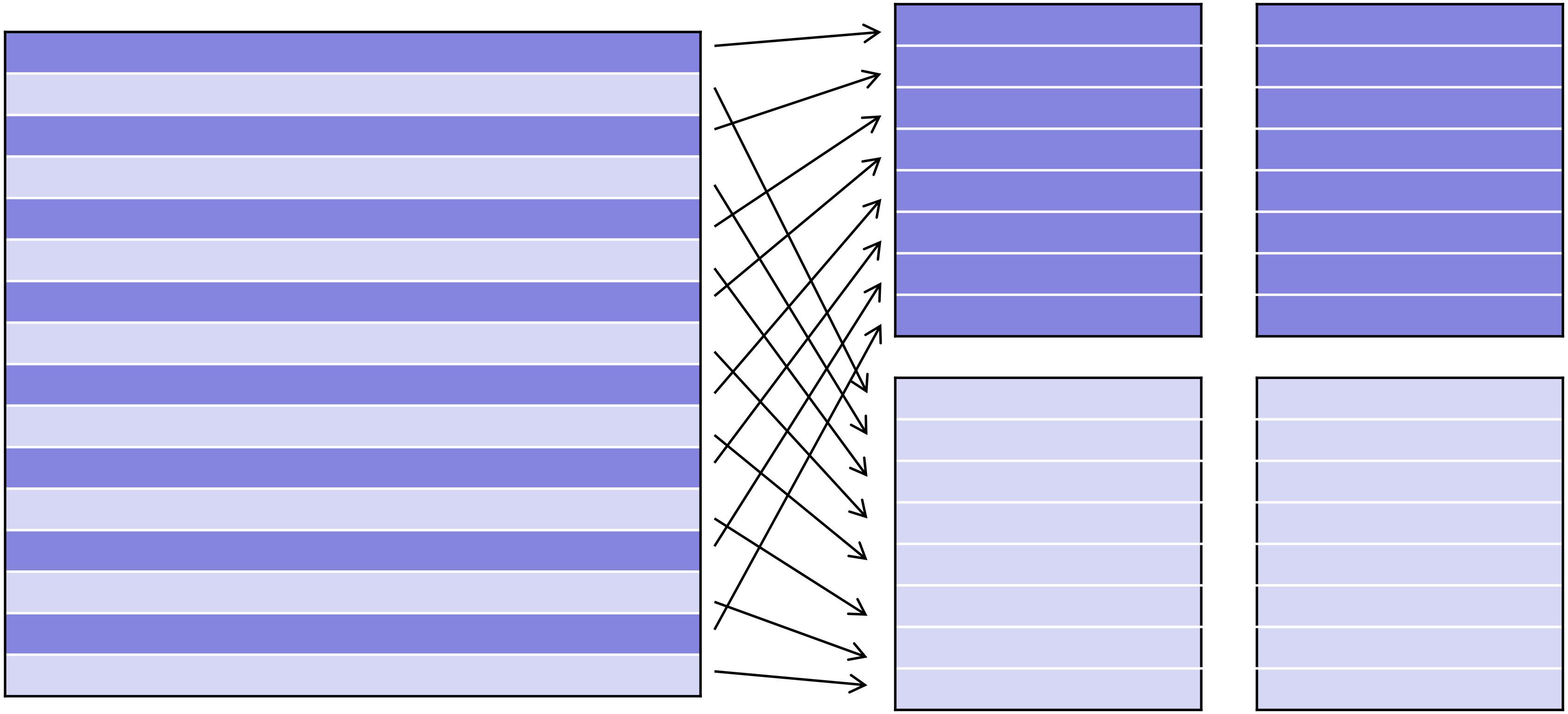
Frame-based and field-based coding



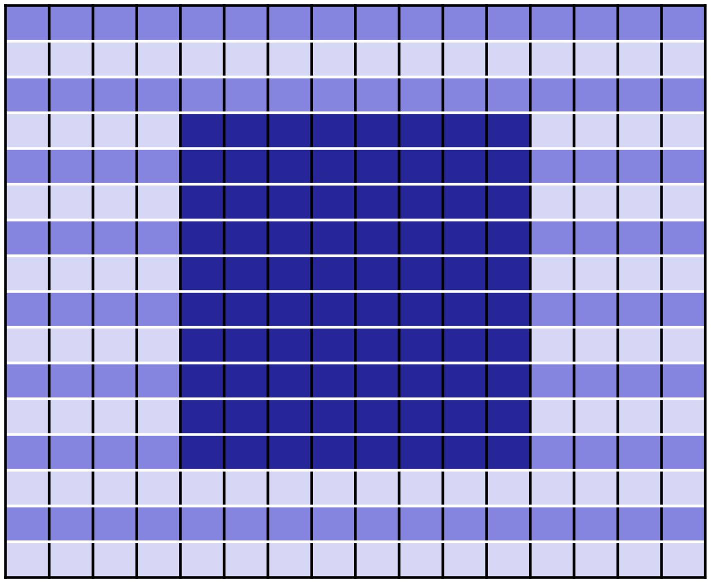
Cutting blocks out of macroblocks (frame mode)



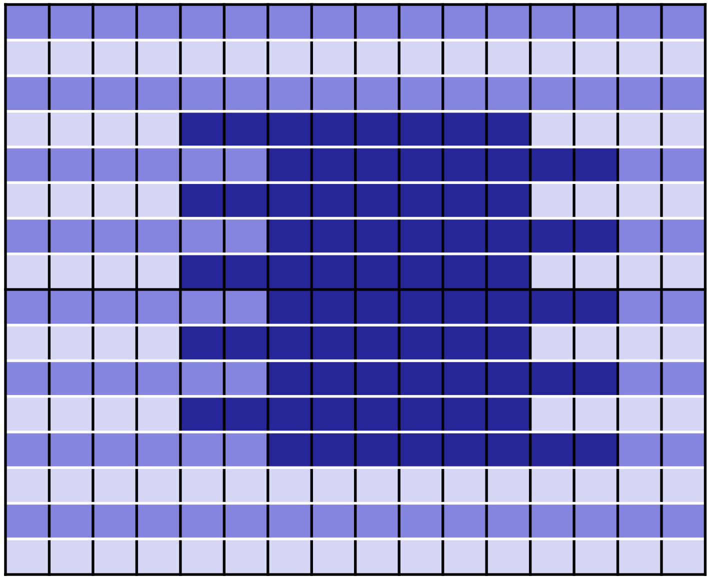
Cutting blocks out of macroblocks (field mode)



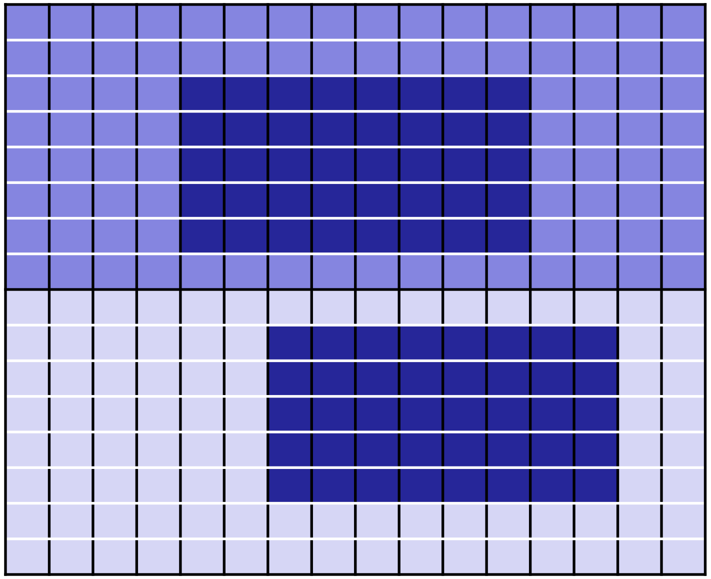
Example



- We assume that the block starts to move to the right at the rate of 2 pixels/field



Frame mode



Field mode

MPEG - 4

MPEG-4

- The MPEG-4 standard (ISO/IEC 14496) is the successor to MPEG-2. The success of this new standard is due to:
  - Development of advanced video service (i.e. HDTV & video-on-demand)
  - Discover of new high-performance tecnologies that brings more power and memory availability
- This compression technology is also know as H.264/AVC.

Benefits of H.264/AVC

- The main benefits of H.264/AVC are as follows:
  - Good performance
  - Low bandwidth requirements
  - Interoperable with existing video processing infrastructure
  - Support for HDTV
  - Selected by a wide range of organizations
  - Reduced storage space
  - Support for multiple applications
  - Transport independent
  - Adapts easily to poor quality networks
  - Used in a wide variety of applications

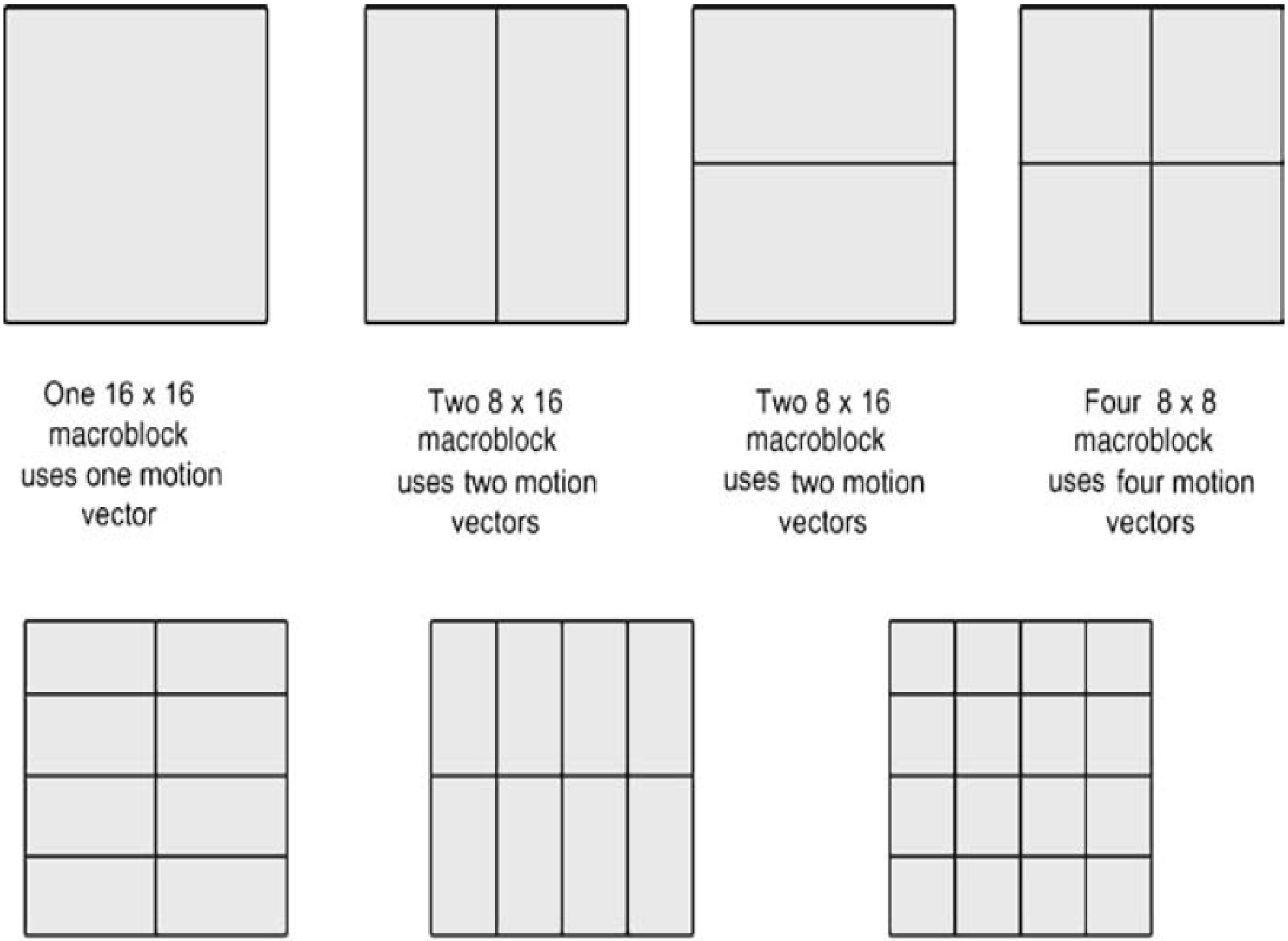
H.264/AVC encoding modules and technical characteristic

- The H.264/AVC coding comprises of a number of underlying encoding modules and technical characteristics.
  - Intraprediction and Coding
    - By using the intraprediction and coding mechanism, H.264/AVC is able to exploit the spatial redundancies that are part of a video picture.
  - Interprediction and Coding
    - The interprediction and coding component of H.264/AVC relies on motion estimation to exploit various temporal redundancies that exist between video frames in a sequence.

H.264/AVC encoding modules and technical characteristic (2)

- Multiple Frame Types
  - H.264/AVC has also added support for two additional frame types Switching I (SI) and Switching P (SP).
- Sophisticated Frame Referencing System
  - H.264/ACV allows B-frame access any frame in any part of the video timeline for motion-compensated predictions.
- Macroblock Partitioning
  - H.264/ACV, unlike previous MPEG version, allow to divide maroblock into partitions. This improves prediction and allows encoders to handle video motion at a very detailed level.

Block sizes supported by H.264/AVC / MPEG4-10



H.264/AVC encoding modules and technical characteristic (3)

- Advanced Transform and Quantization Process
  - This module compresses the bulk of the video data.
- Deblocking Loop Filter
  - H.264 specifies the use of a filter to prevent the appearance of artifacts on the TV screen.
- Specific Profiles
  - H.264/AVC standard specifies seven different “entertainment centric” profiles or feature sets

VC - 1

VC-1 (Video Codec 1)

- Next generation compression technology
- Adopted by:
  - WMV 9
  - HD-DVD
  - Blu-ray

VC-1 Characteristics (1)

- Deployable across a range of platforms
- Support for three separate profiles
  - Simple, main and advanced
  - Main is divided in low,medium and high level

Level	Maximum bitrate (Mbps)	Resolutions supported
Low	2	320 x240 operating at a frequency of 24 Hertz. Typically used by computer displays
Medium	10	Supports two resolutions: 720x480 at 30 Hertz and 720 576 at 25 Hertz. This level is well suited to the delivery of SD IPTV
High	20	This level is capable of processing high resolution video content such as frames that contain 1920 1080 pixel densities

VC-1 Characteristics (2)

- ASF support
- Support for a range of block sizes
- Additional frame type
- Operates across a range of network transport technologies