Kurs INF5080

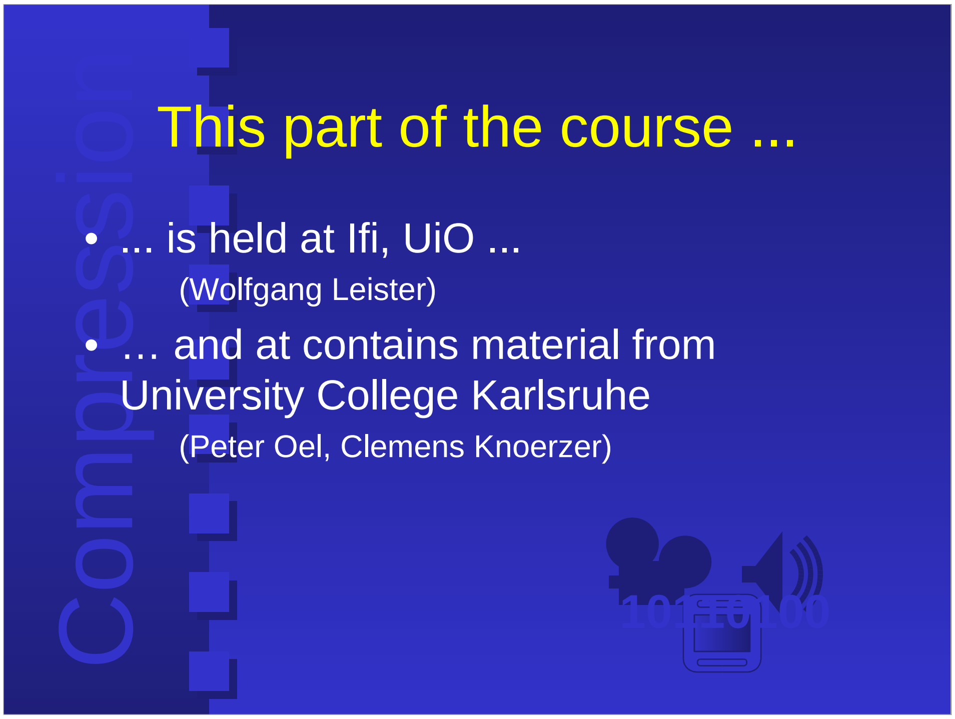
Compression

Ifi, UiO

Norsk Regnesentral

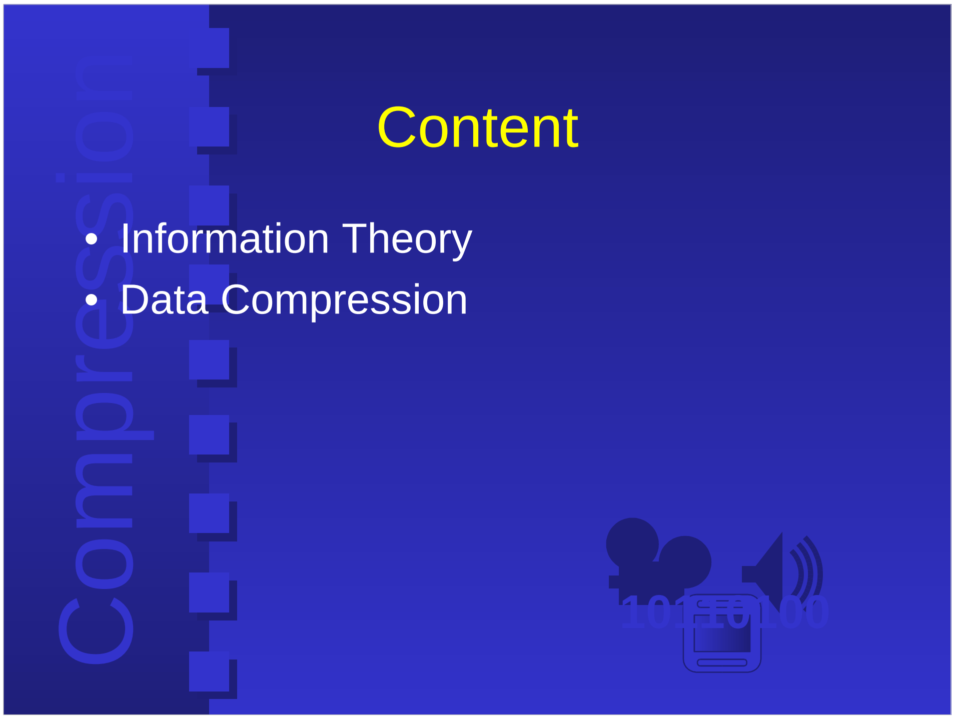
Vårsemester 2005

Wolfgang Leister



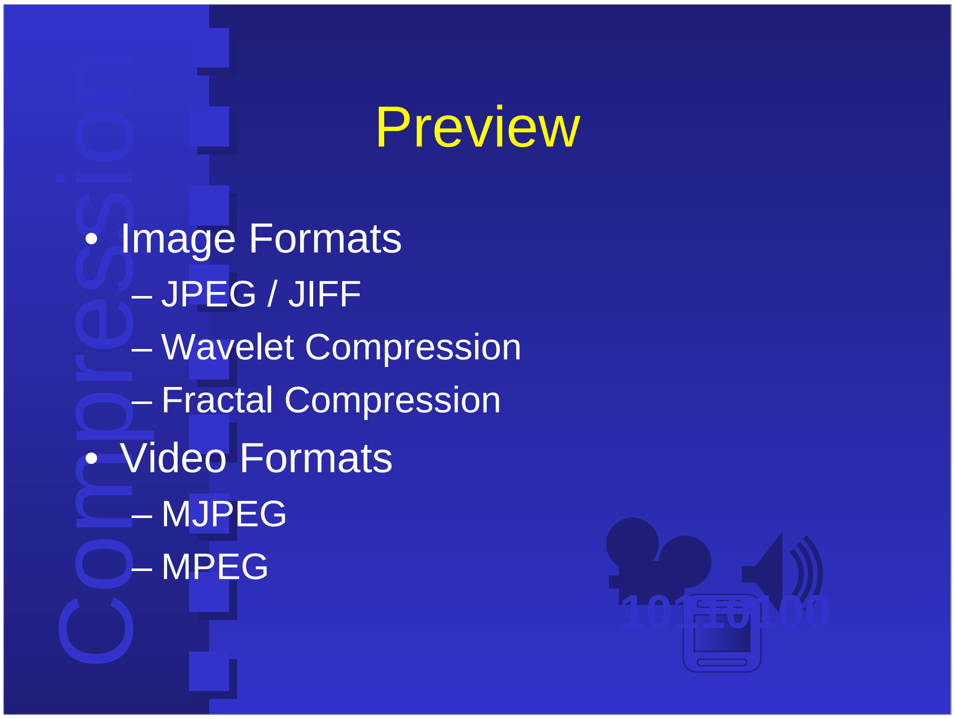
This part of the course ...

- ... is held at Ifi, UiO ...
  (Wolfgang Leister)
- … and at contains material from University College Karlsruhe
  (Peter Oel, Clemens Knoerzer)


Compression
10110100

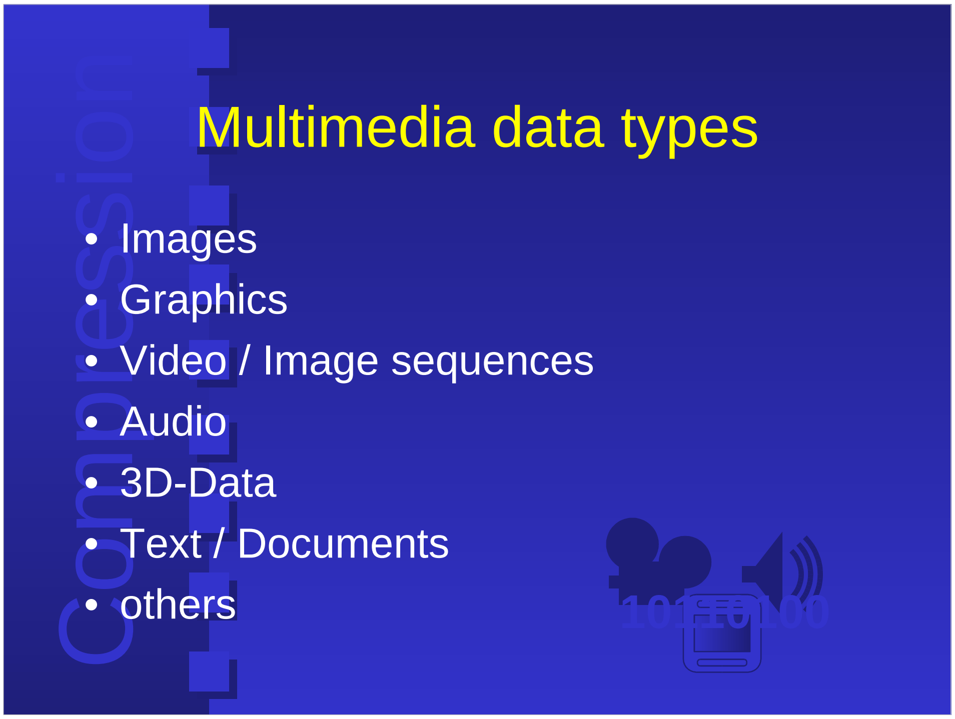
Content

- Information Theory
- Data Compression


Compression
10110100

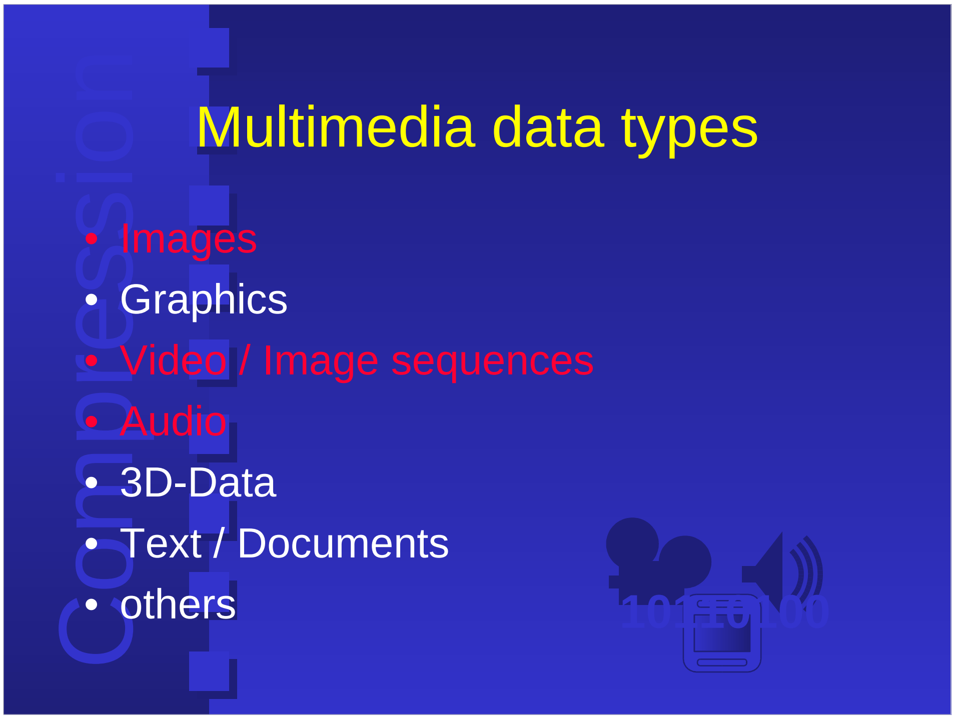
Preview

- Image Formats
  - JPEG / JIFF
  - Wavelet Compression
  - Fractal Compression
- Video Formats
  - MJPEG
  - MPEG



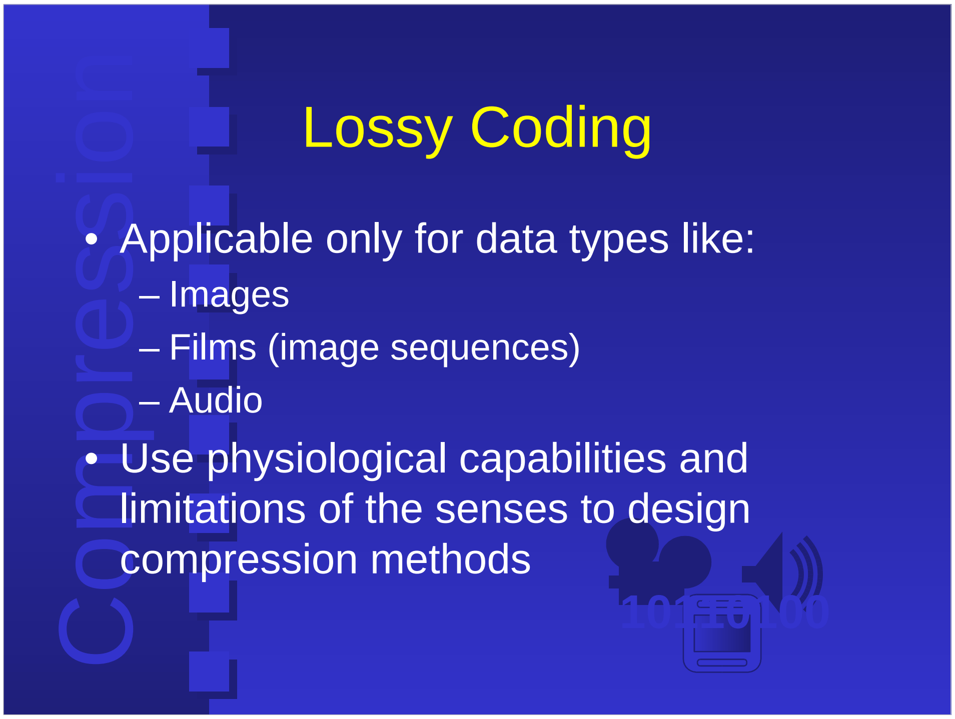
Multimedia data types

- Images
- Graphics
- Video / Image sequences
- Audio
- 3D-Data
- Text / Documents
- others



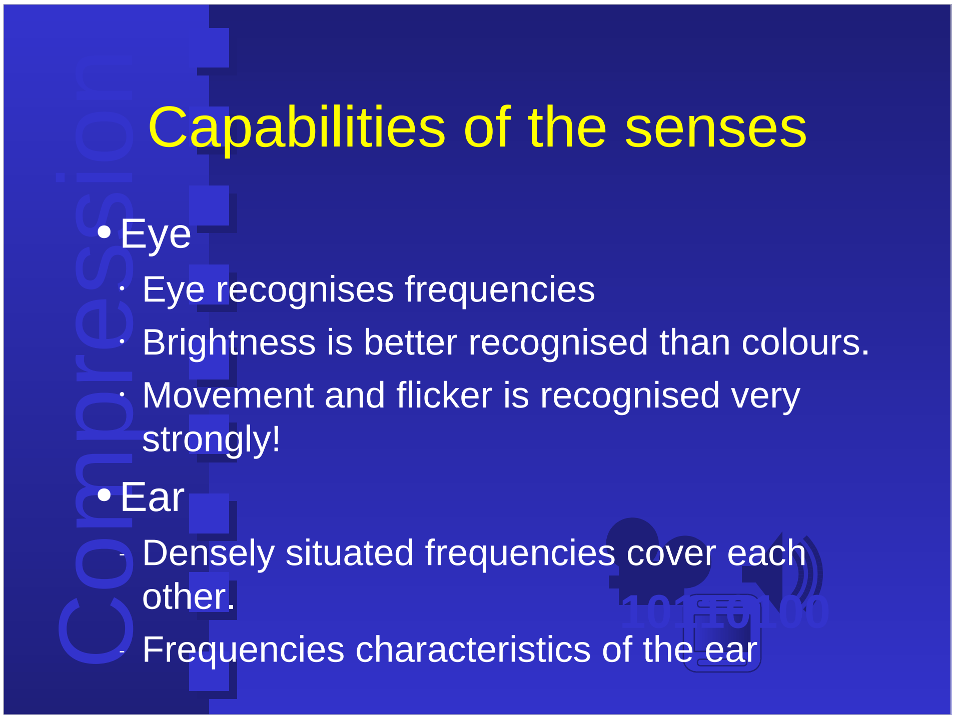
Multimedia data types

- Images
- Graphics
- Video / Image sequences
- Audio
- 3D-Data
- Text / Documents
- others



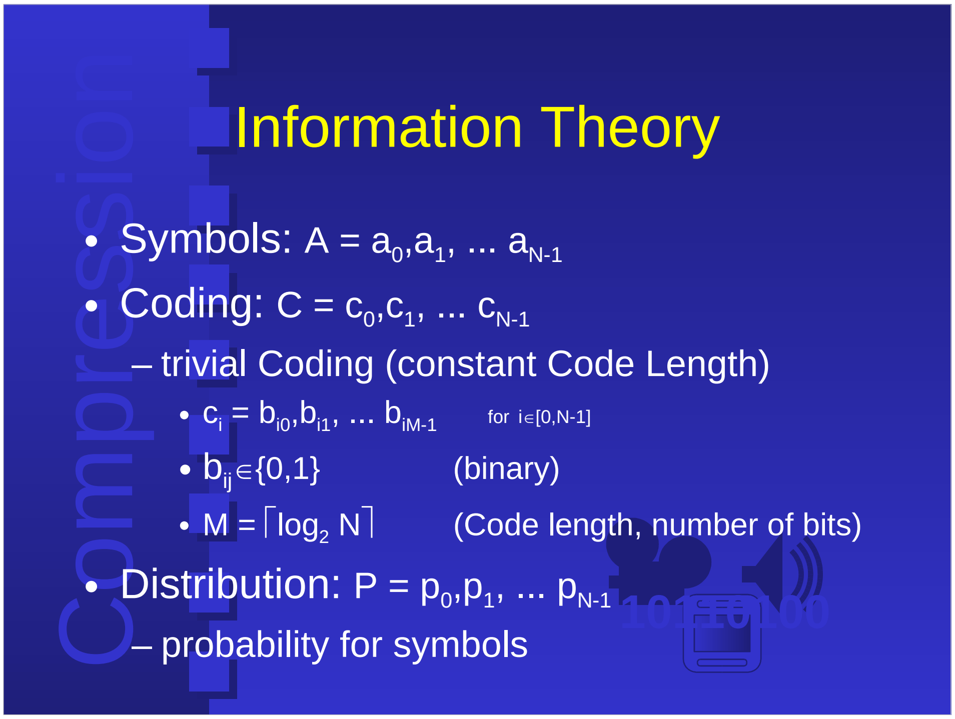
Lossy Coding

- Applicable only for data types like:
  - Images
  - Films (image sequences)
  - Audio
- Use physiological capabilities and limitations of the senses to design compression methods



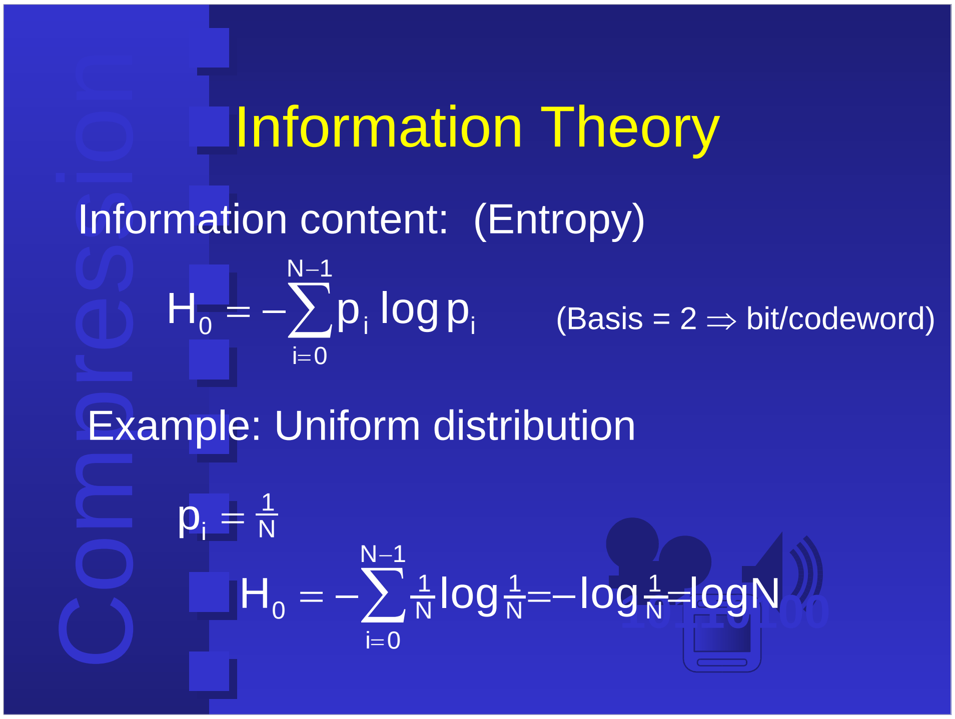
Capabilities of the senses

- Eye
  - Eye recognises frequencies
  - Brightness is better recognised than colours.
  - Movement and flicker is recognised very strongly!
- Ear
  - Densely situated frequencies cover each other.
  - Frequencies characteristics of the ear



Information Theory

- Symbols: $A = a_0, a_1, \ldots a_{N-1}$
- Coding: $C = c_0, c_1, \ldots c_{N-1}$
  - – trivial Coding (constant Code Length)
    - $c_i = b_{i0}, b_{i1}, \ldots b_{iM-1}$ for $i \in [0,N-1]$
    - $b_{ij} \in \{0,1\}$ (binary)
    - $M = \lceil \log_2 N \rceil$ (Code length, number of bits)
- Distribution: $P = p_0, p_1, \ldots p_{N-1}$
  - – probability for symbols



Information Theory

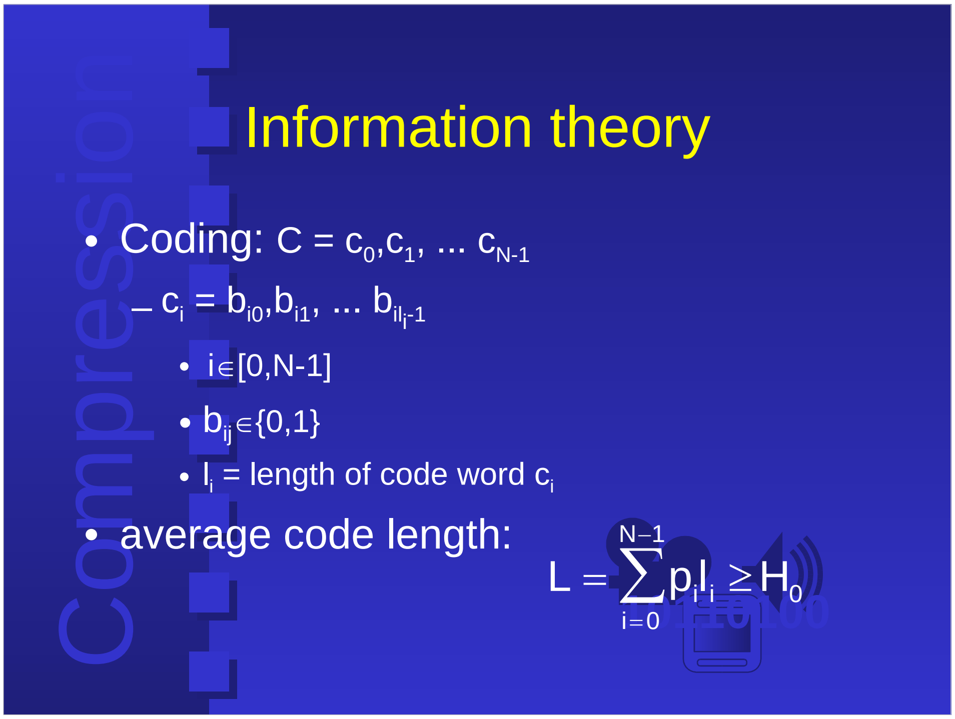
Information content: (Entropy)

$$H_0 = -\sum_{i=0}^{N-1} p_i \log p_i$$ (Basis = 2 $\Rightarrow$ bit/codeword)

Example: Uniform distribution

$$p_i = \tfrac{1}{N}$$

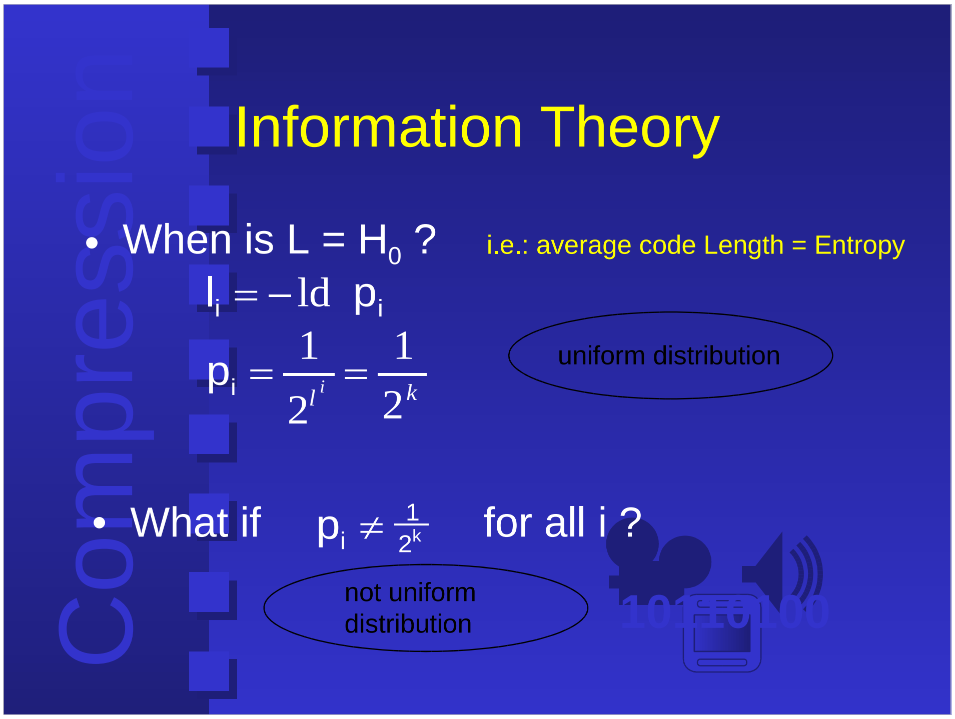
$$H_0 = -\sum_{i=0}^{N-1} \tfrac{1}{N} \log \tfrac{1}{N} = -\log \tfrac{1}{N} = \log N$$



Information theory

- Coding: $C = c_0, c_1, \ldots c_{N-1}$
  - $c_i = b_{i0}, b_{i1}, \ldots b_{il_i-1}$
    - $i \in [0,N-1]$
    - $b_{ij} \in \{0,1\}$
    - l_i = length of code word c_i
- average code length:

$$L = \sum_{i=0}^{N-1} p_i l_i \ge H_0$$



Information Theory

- When is $L = H_0$? i.e.: average code Length = Entropy

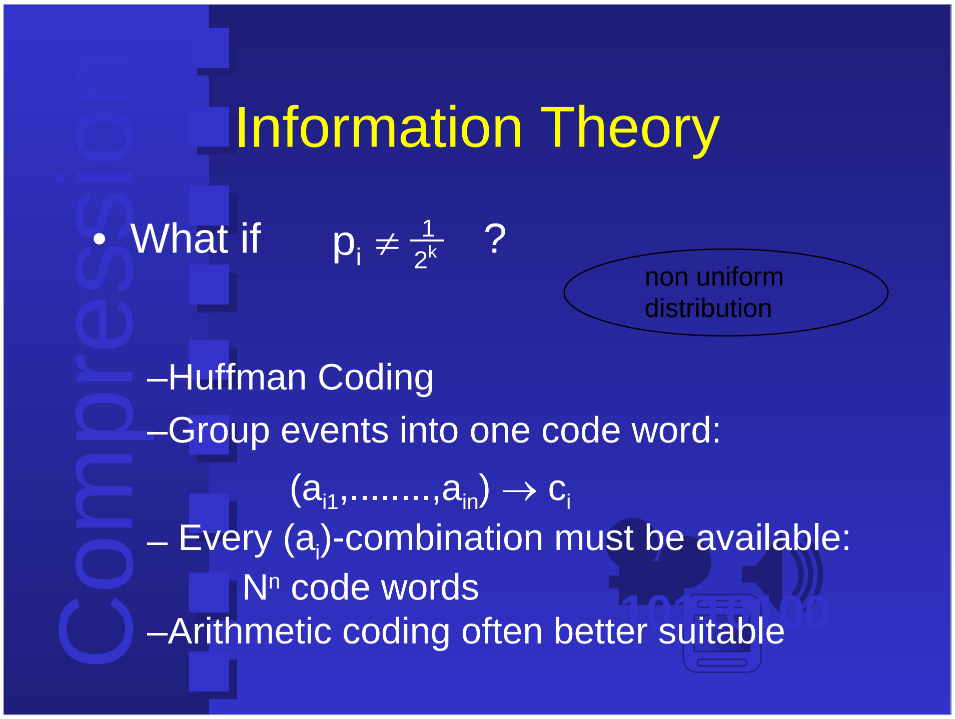
$$l_i = -\mathrm{ld}\ p_i$$

$$p_i = \frac{1}{2^{l_i}} = \frac{1}{2^k}$$

uniform distribution

- What if $p_i \neq \frac{1}{2^k}$ for all i ?

not uniform distribution



Information Theory

- What if $p_i \neq \frac{1}{2^k}$?

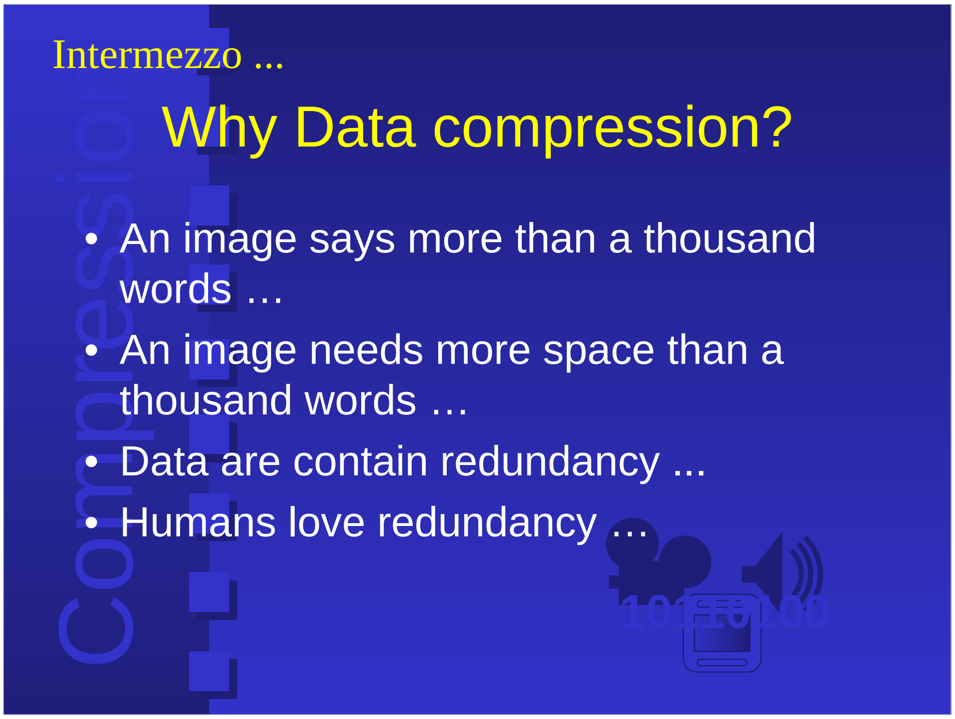
non uniform distribution

- –Huffman Coding
- –Group events into one code word:

$$(a_{i1},........,a_{in}) \rightarrow c_i$$

- – Every (a_i)-combination must be available:

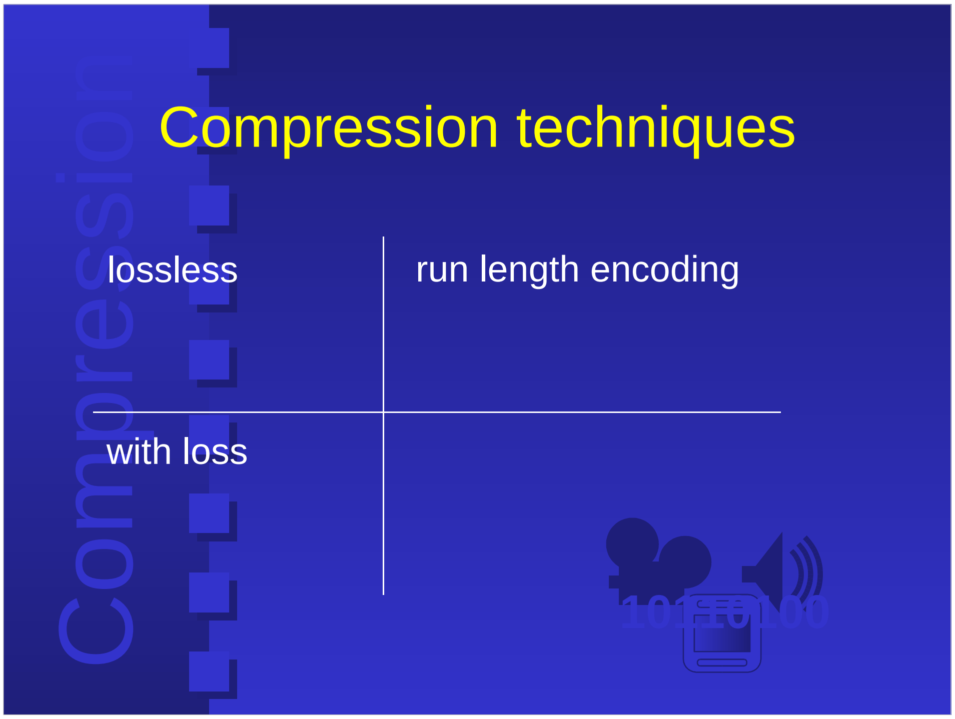
  N^n code words
- –Arithmetic coding often better suitable



Intermezzo ...

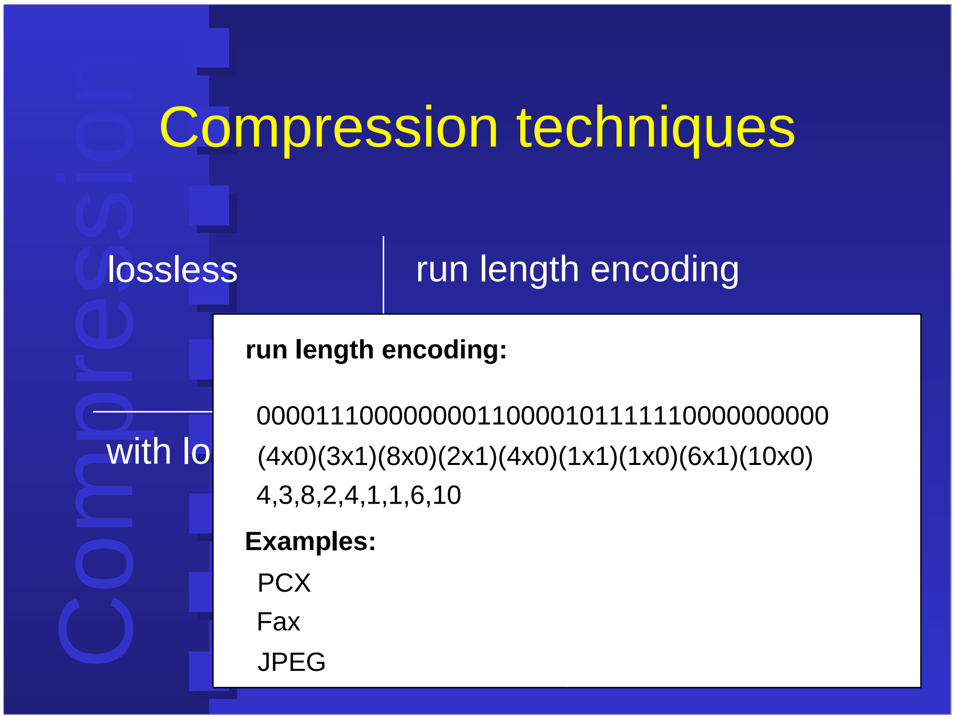
Why Data compression?

- An image says more than a thousand words …
- An image needs more space than a thousand words …
- Data are contain redundancy ...
- Humans love redundancy …


Compression
10110100

Compression techniques

lossless	run length encoding
with loss	


Compression

Compression techniques

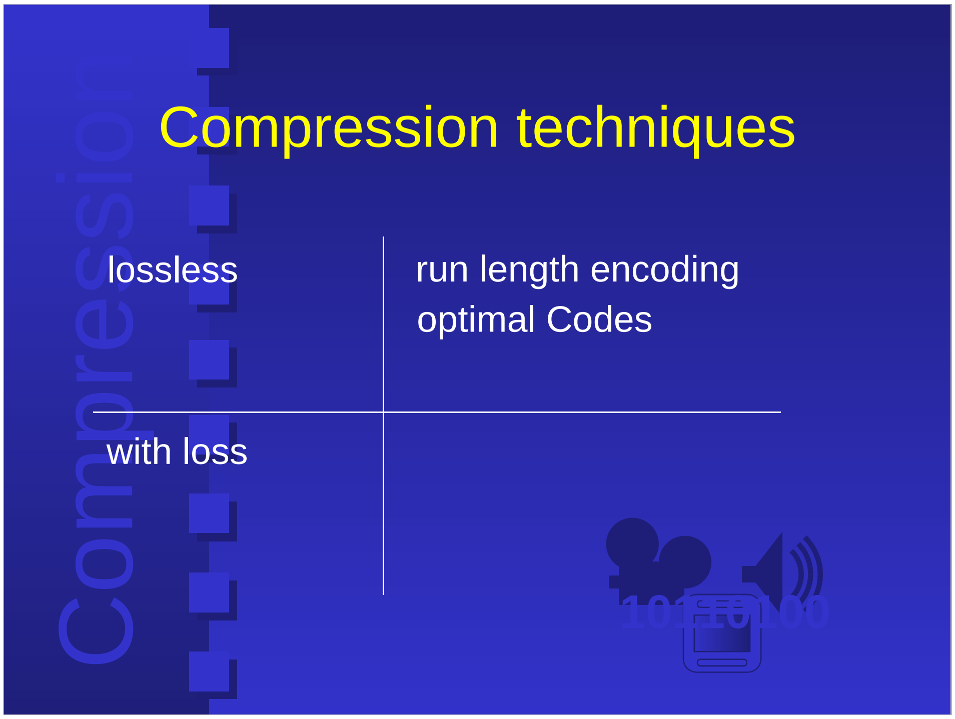
lossless	run length encoding
with lo	

run length encoding:

0000111000000000110000101111110000000000
(4x0)(3x1)(8x0)(2x1)(4x0)(1x1)(1x0)(6x1)(10x0)
4,3,8,2,4,1,1,6,10

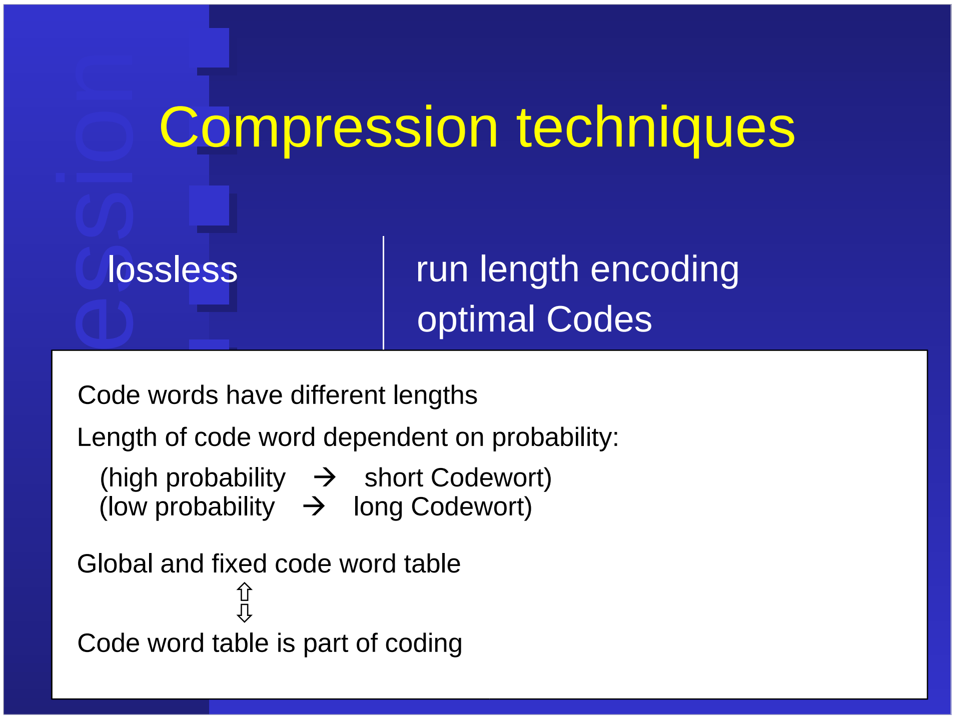
Examples:

PCX
Fax
JPEG



Compression techniques

lossless	run length encoding optimal Codes
with loss	



Compression techniques

lossless	run length encoding optimal Codes

Code words have different lengths

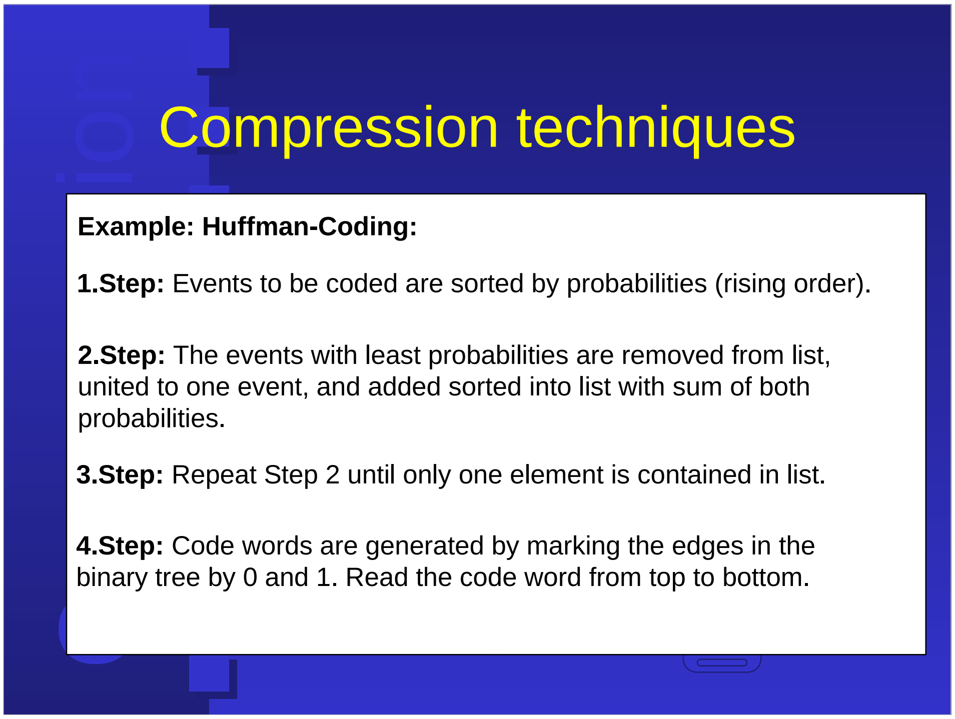
Length of code word dependent on probability:

(high probability → short Codewort)
(low probability → long Codewort)

Global and fixed code word table

⇳

Code word table is part of coding



Compression techniques

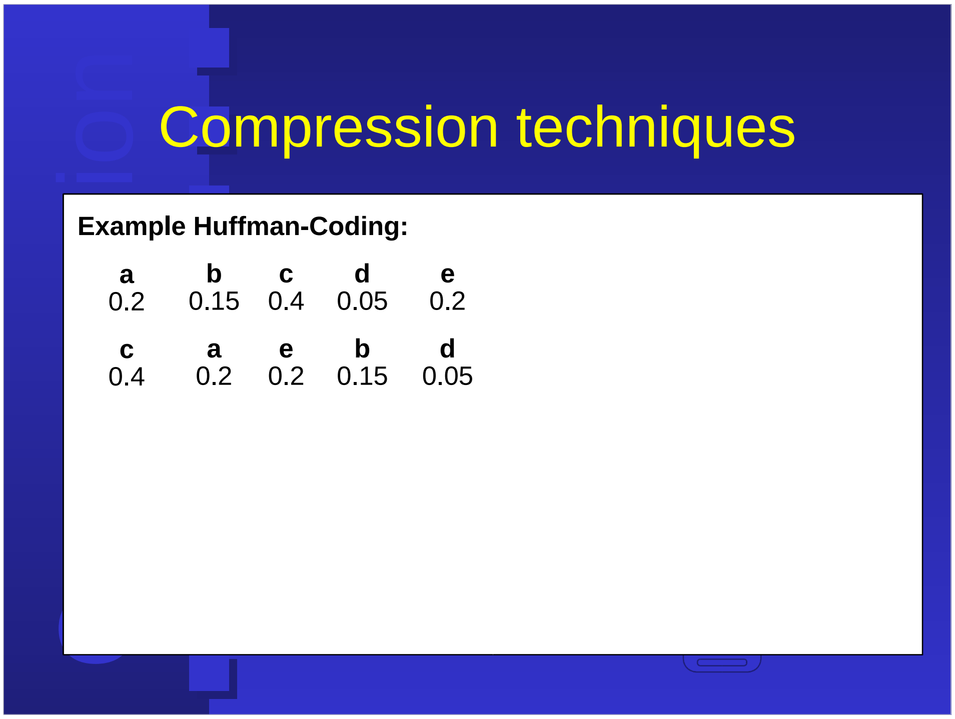
Example: Huffman-Coding:

1.Step: Events to be coded are sorted by probabilities (rising order).

2.Step: The events with least probabilities are removed from list, united to one event, and added sorted into list with sum of both probabilities.

3.Step: Repeat Step 2 until only one element is contained in list.

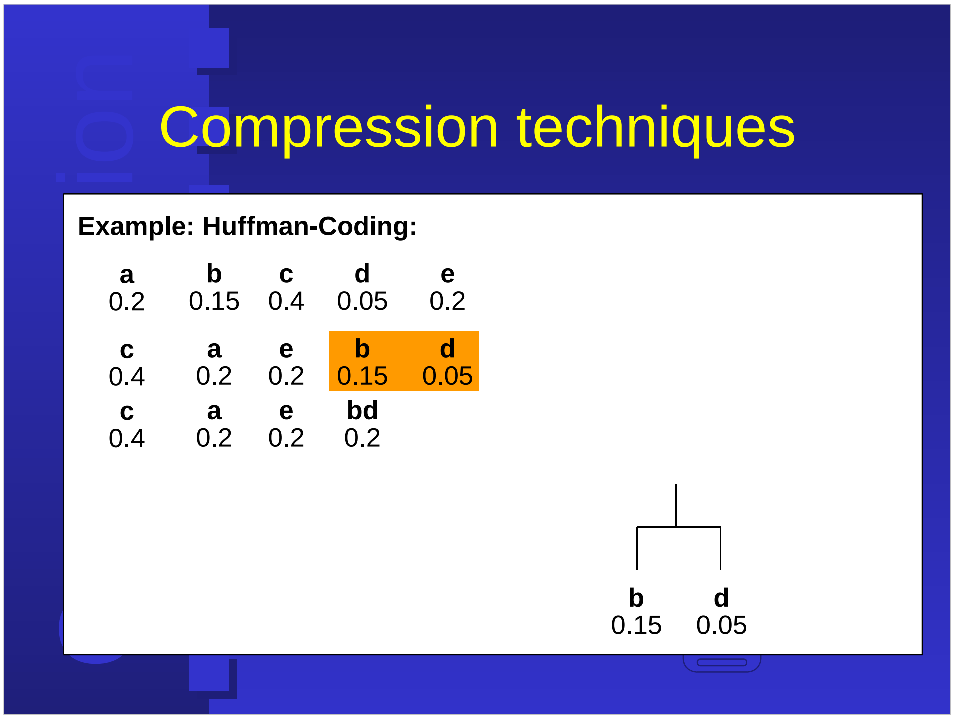
4.Step: Code words are generated by marking the edges in the binary tree by 0 and 1. Read the code word from top to bottom.



Compression techniques

Example Huffman-Coding:

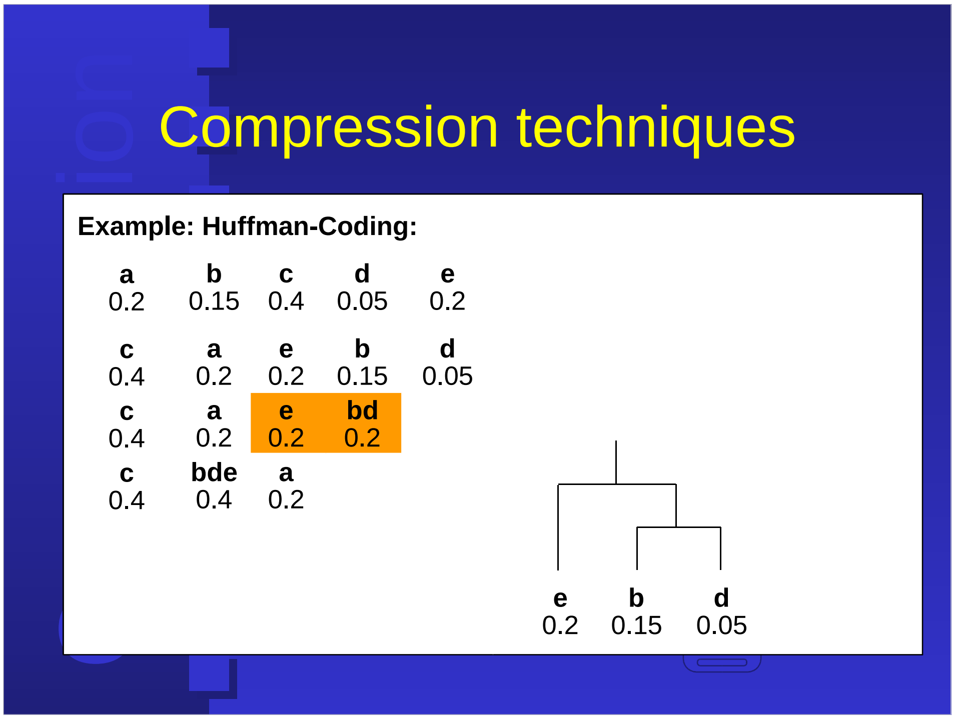
a	**b**	**c**	**d**	**e**
0.2	0.15	0.4	0.05	0.2
c	**a**	**e**	**b**	**d**
0.4	0.2	0.2	0.15	0.05



Compression techniques

Example: Huffman-Coding:

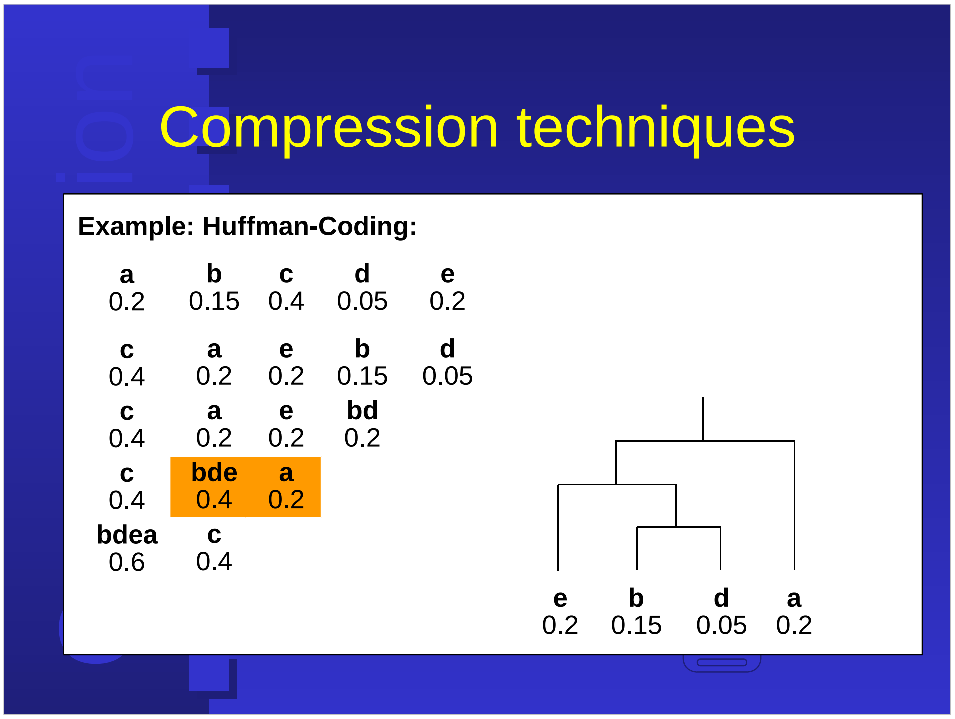
a	**b**	**c**	**d**	**e**
0.2	0.15	0.4	0.05	0.2
c	**a**	**e**	**b**	**d**
0.4	0.2	0.2	0.15	0.05
c	**a**	**e**	**bd**	
0.4	0.2	0.2	0.2	



Compression techniques

Example: Huffman-Coding:

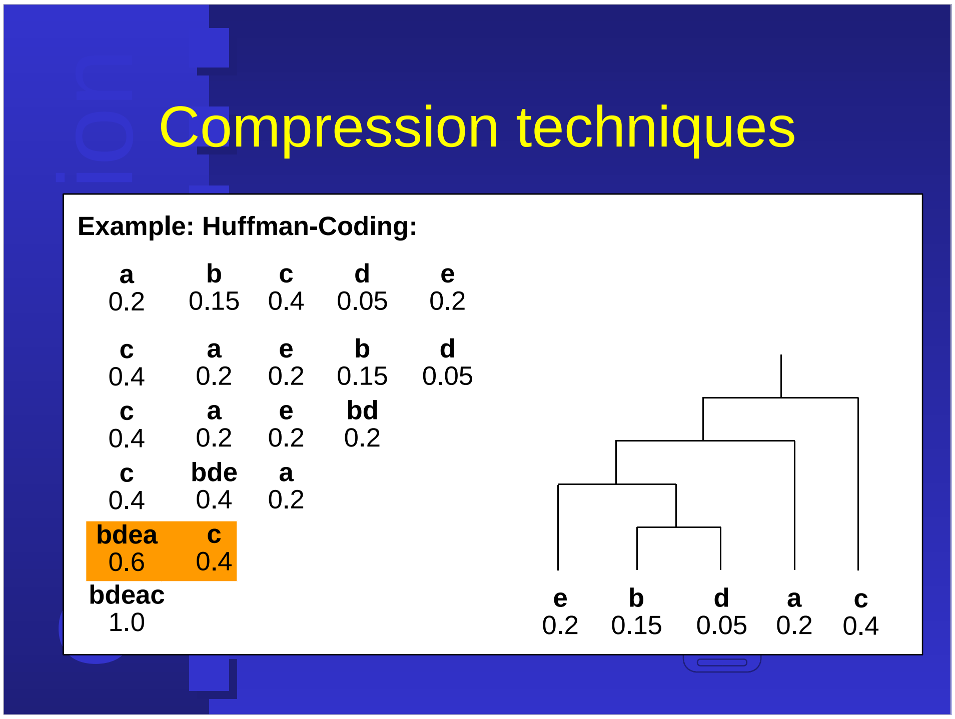
a	**b**	**c**	**d**	**e**
0.2	0.15	0.4	0.05	0.2
c	**a**	**e**	**b**	**d**
0.4	0.2	0.2	0.15	0.05
c	**a**	**e**	**bd**	
0.4	0.2	0.2	0.2	
c	**bde**	**a**		
0.4	0.4	0.2		



Compression techniques

Example: Huffman-Coding:

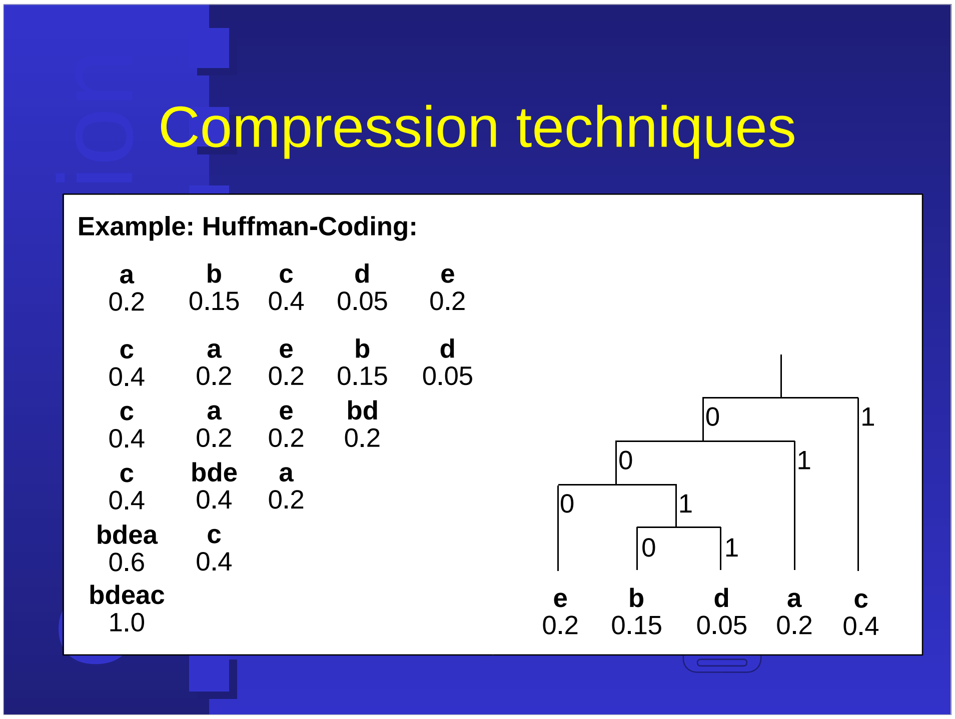
a 0.2	**b** 0.15	**c** 0.4	**d** 0.05	**e** 0.2
c 0.4	**a** 0.2	**e** 0.2	**b** 0.15	**d** 0.05
c 0.4	**a** 0.2	**e** 0.2	**bd** 0.2	
c 0.4	**bde** 0.4	**a** 0.2		
bdea 0.6	**c** 0.4			



Compression techniques

Example: Huffman-Coding:

a 0.2	**b** 0.15	**c** 0.4	**d** 0.05	**e** 0.2
c 0.4	**a** 0.2	**e** 0.2	**b** 0.15	**d** 0.05
c 0.4	**a** 0.2	**e** 0.2	**bd** 0.2	
c 0.4	**bde** 0.4	**a** 0.2		
bdea 0.6	**c** 0.4			
bdeac 1.0				



Compression techniques

Example: Huffman-Coding:

a	**b**	**c**	**d**	**e**
0.2	0.15	0.4	0.05	0.2

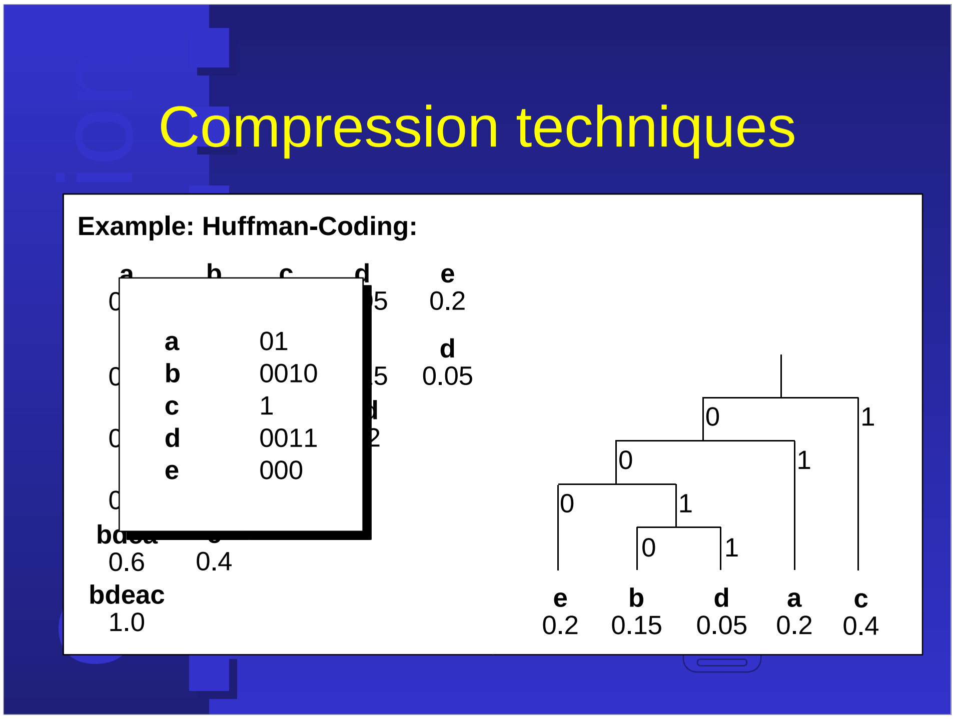
c	**a**	**e**	**b**	**d**
0.4	0.2	0.2	0.15	0.05

c	**a**	**e**	**bd**
0.4	0.2	0.2	0.2

c	**bde**	**a**
0.4	0.4	0.2

bdea	**c**
0.6	0.4

bdeac
1.0



Compression techniques

Example: Huffman-Coding:

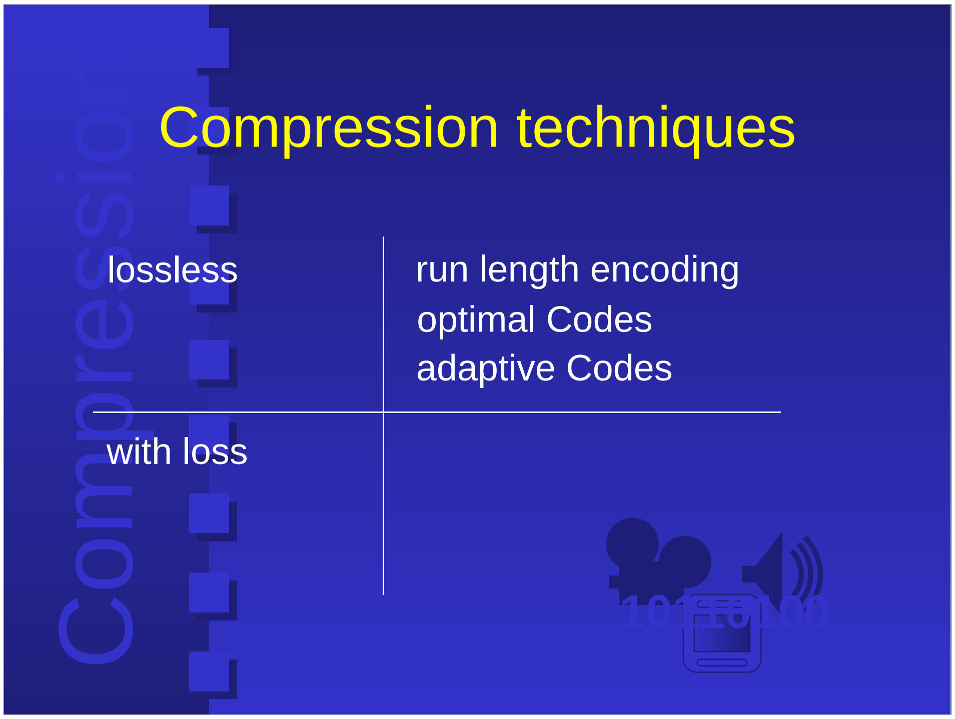
e
0.2

d
0.05

0.6 0.4

bdeac
1.0

a	01
b	0010
c	1
d	0011
e	000

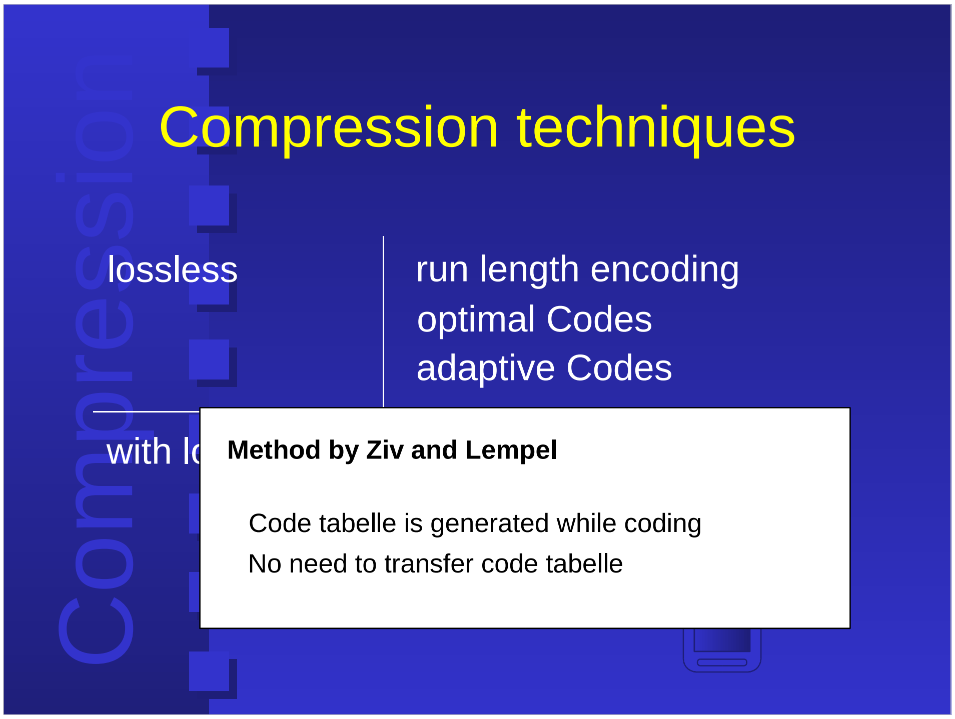


Compression techniques

lossless

- run length encoding
- optimal Codes
- adaptive Codes

with loss



Compression techniques

lossless

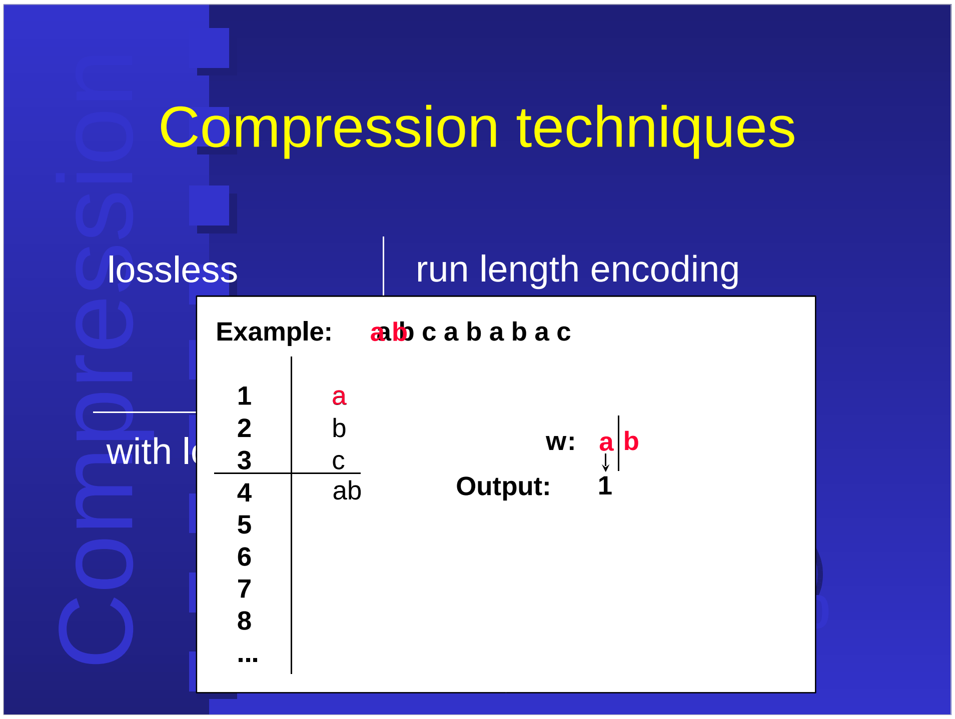
- run length encoding
- optimal Codes
- adaptive Codes

with lo

Method by Ziv and Lempel

Code tabelle is generated while coding

No need to transfer code tabelle



Compression techniques

lossless | run length encoding

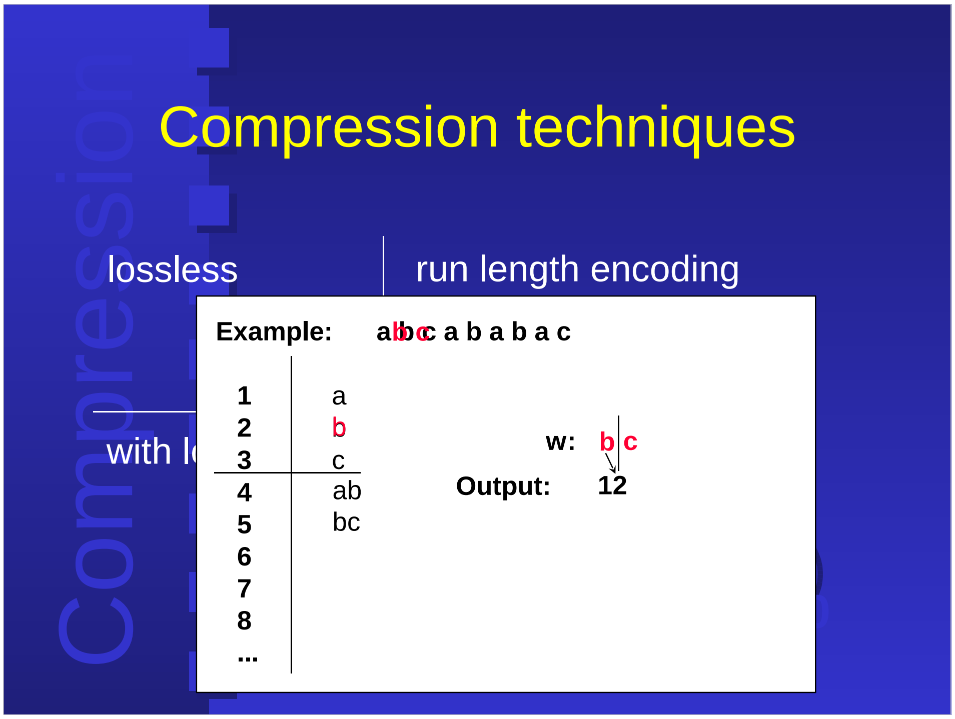
with l

Example: a b c a b a b a c

1	a
2	b
3	c
4	ab
5	
6	
7	
8	
...	

w: a | b

Output: 1



Compression techniques

lossless | run length encoding

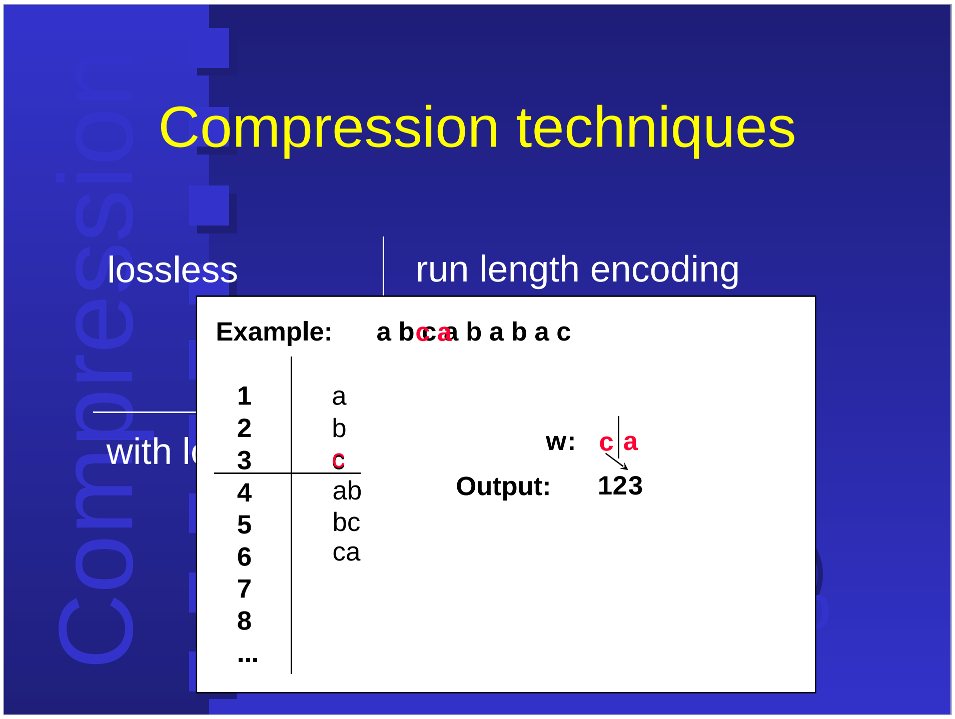
with l

Example: a b c a b a b a c

1	a
2	b
3	c
4	ab
5	bc
6	
7	
8	
...	

w: b | c

Output: 12



Compression techniques

lossless | run length encoding

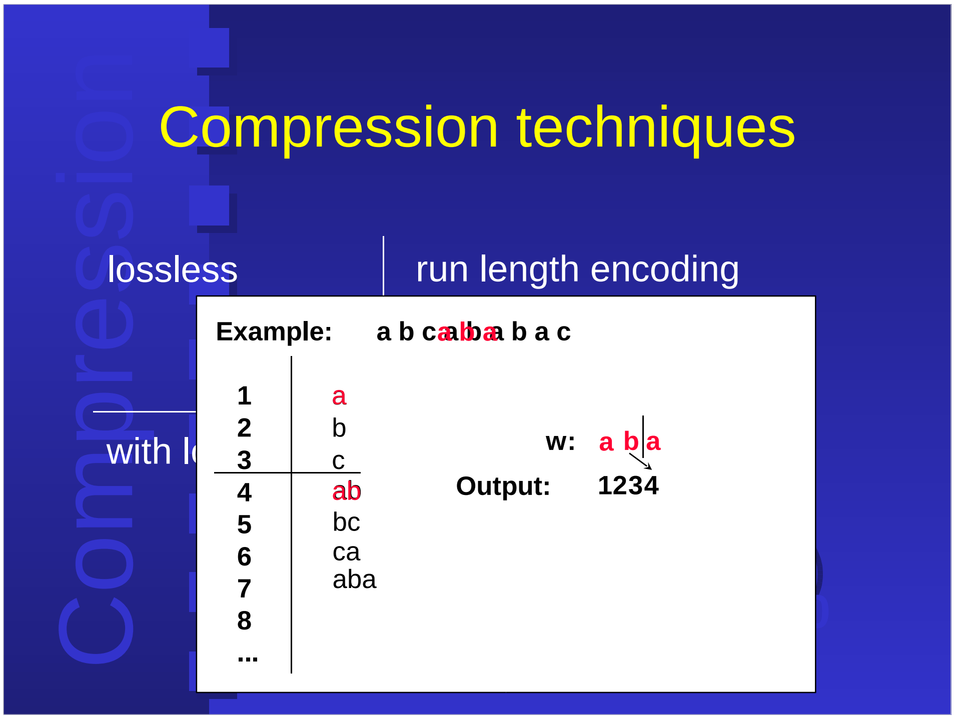
with l

Example: a b c a b a b a c

1	a
2	b
3	c
4	ab
5	bc
6	ca
7	
8	
...	

w: c a

Output: 123



Compression techniques

lossless | run length encoding

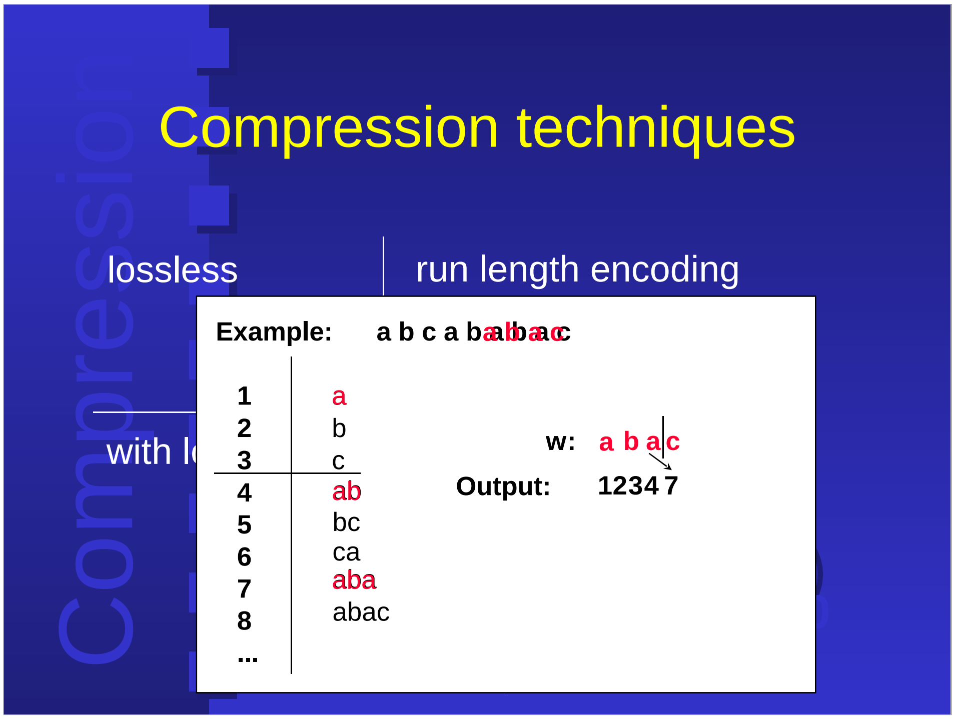
with l

Example: a b c a b a b a c

1	a
2	b
3	c
4	ab
5	bc
6	ca
7	aba
8	
...	

w: a b a

Output: 1234



Compression techniques

lossless

run length encoding

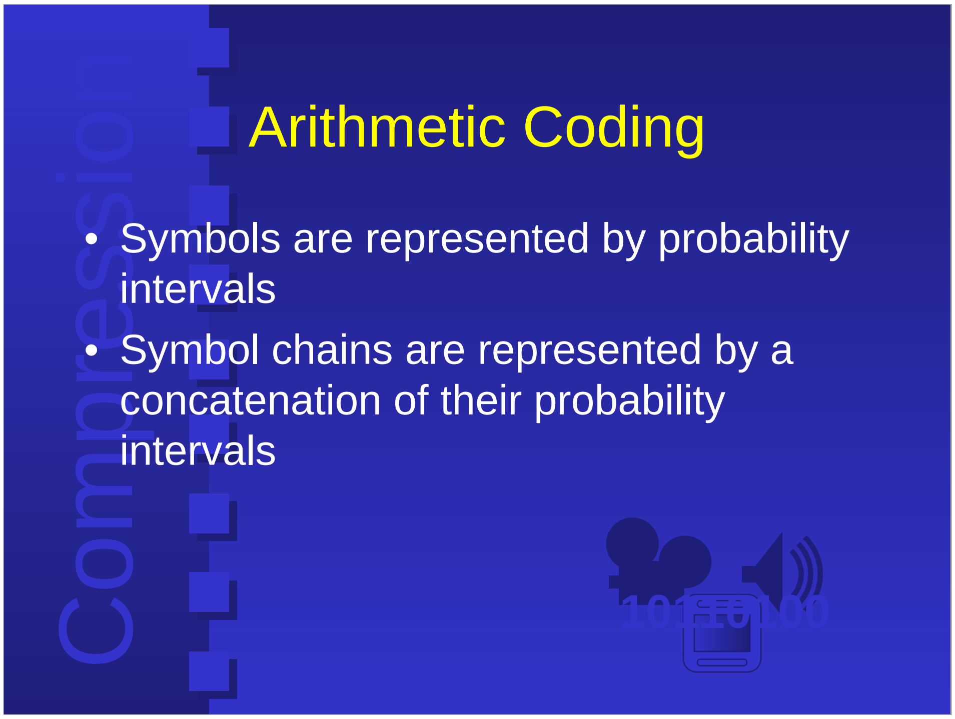
with l

Example: a b c a b a b a c

1	a
2	b
3	c
4	ab
5	bc
6	ca
7	aba
8	abac
...	

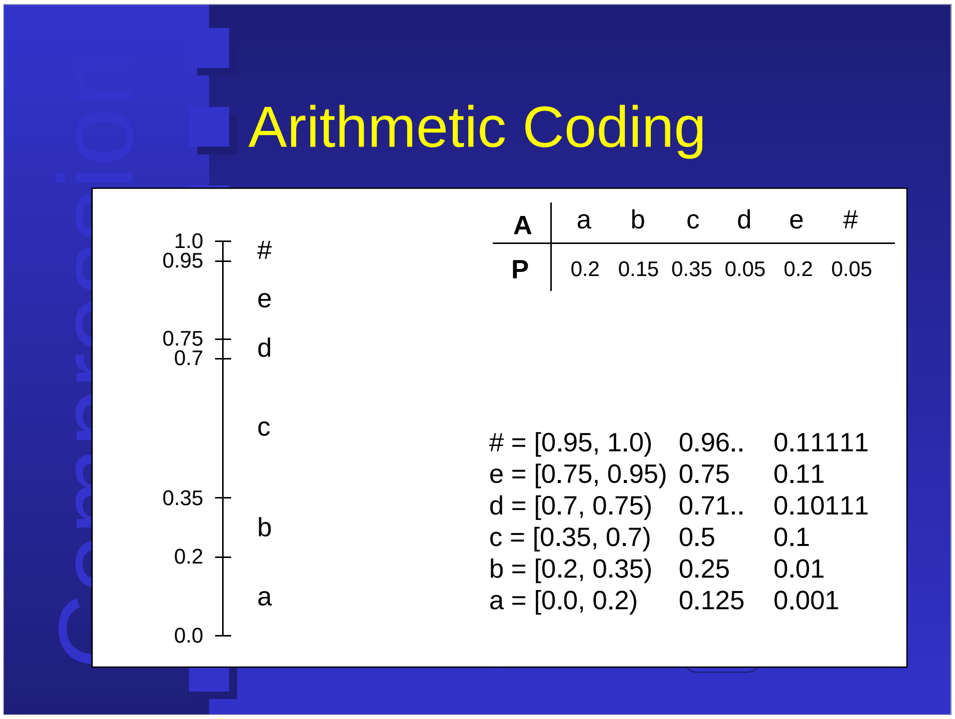
w: a b a c

Output: 1234 7



Arithmetic Coding

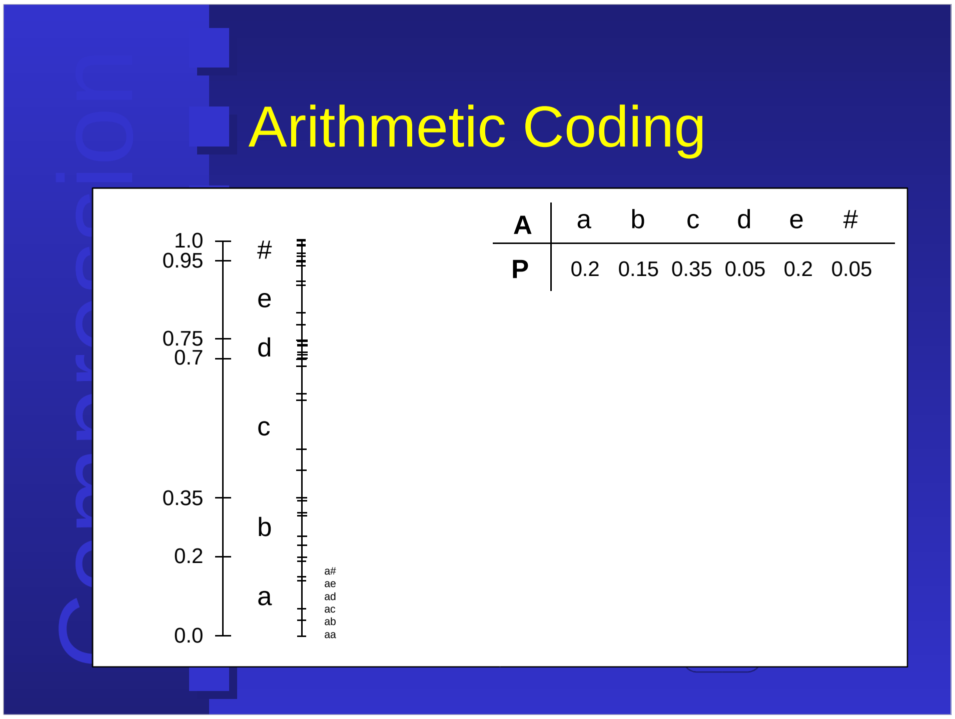
- Symbols are represented by probability intervals
- Symbol chains are represented by a concatenation of their probability intervals



Arithmetic Coding

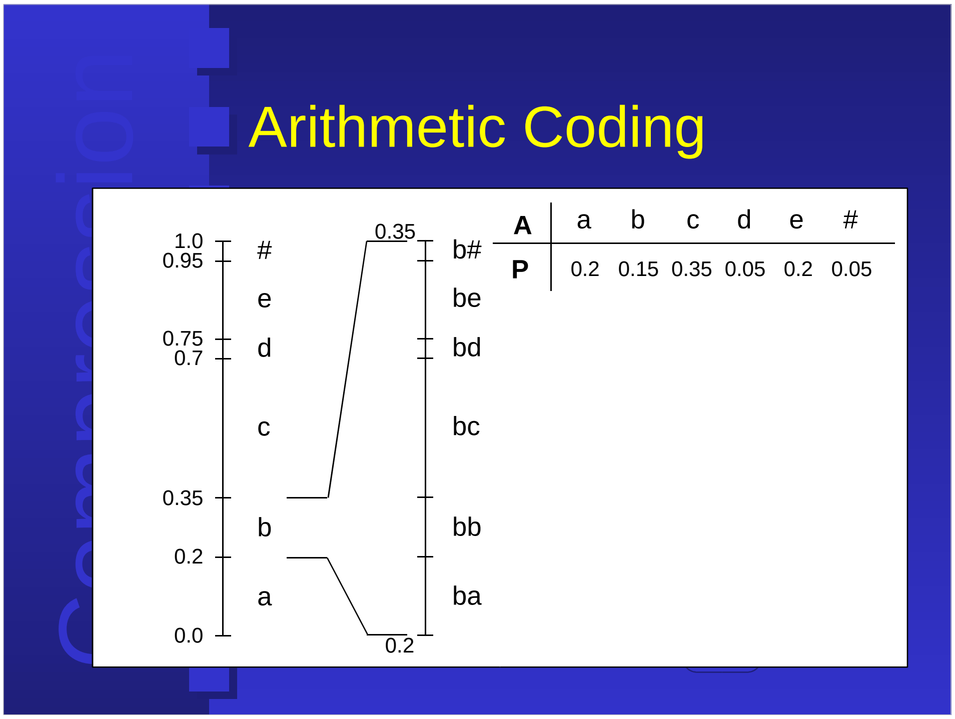
A	a	b	c	d	e	#
P	0.2	0.15	0.35	0.05	0.2	0.05

# = [0.95, 1.0)	0.96..	0.11111
e = [0.75, 0.95)	0.75	0.11
d = [0.7, 0.75)	0.71..	0.10111
c = [0.35, 0.7)	0.5	0.1
b = [0.2, 0.35)	0.25	0.01
a = [0.0, 0.2)	0.125	0.001



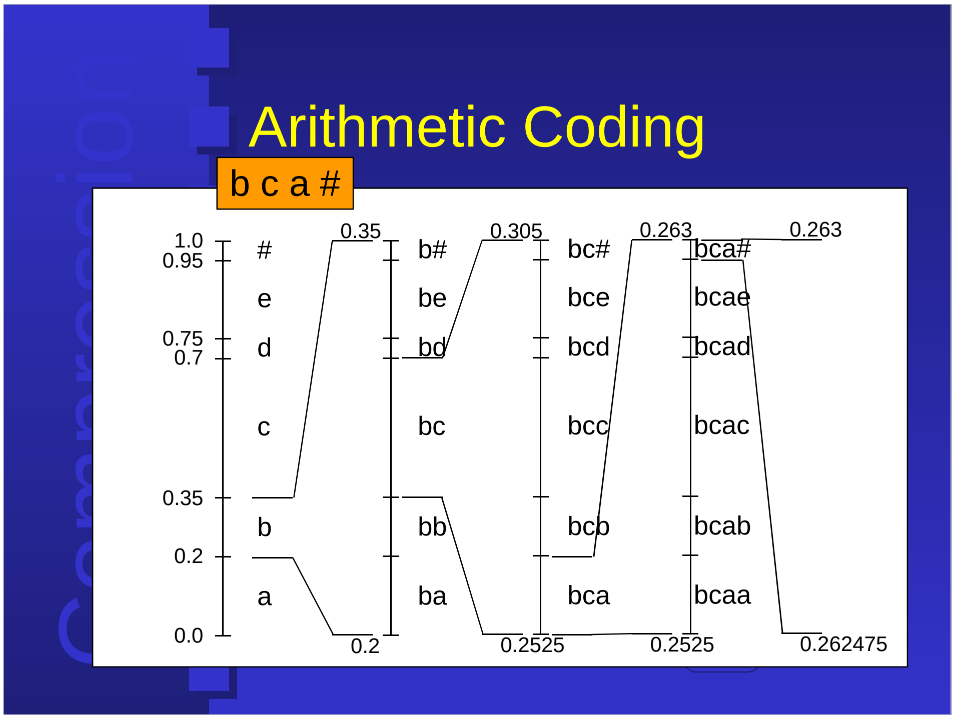
Arithmetic Coding

A	a	b	c	d	e	#
P	0.2	0.15	0.35	0.05	0.2	0.05



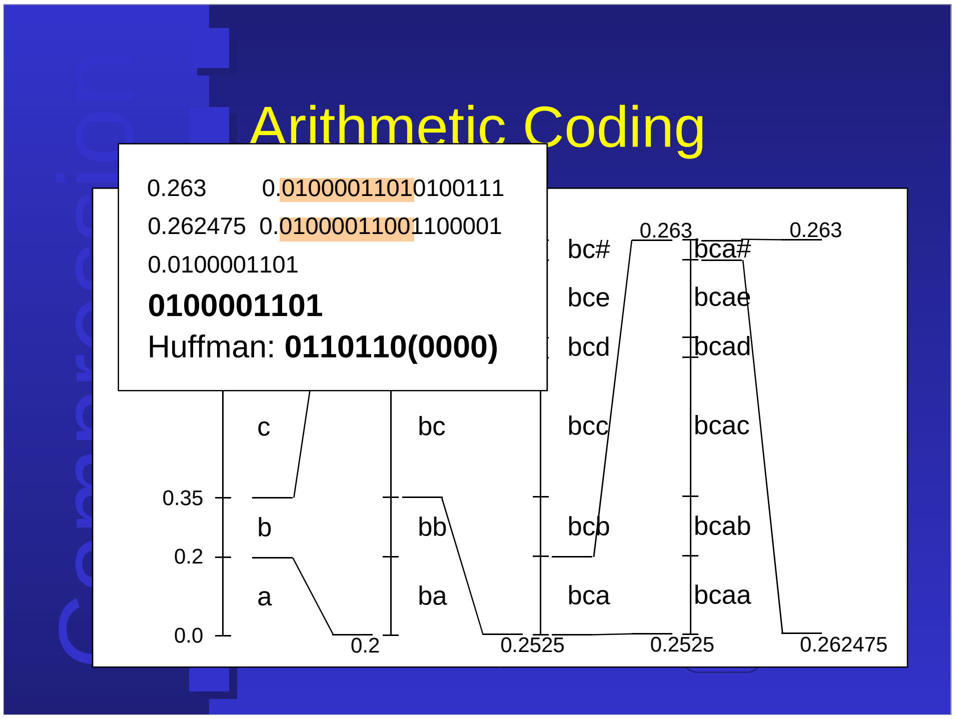
Arithmetic Coding

A	a	b	c	d	e	#
P	0.2	0.15	0.35	0.05	0.2	0.05



Arithmetic Coding

b c a #



Arithmetic Coding

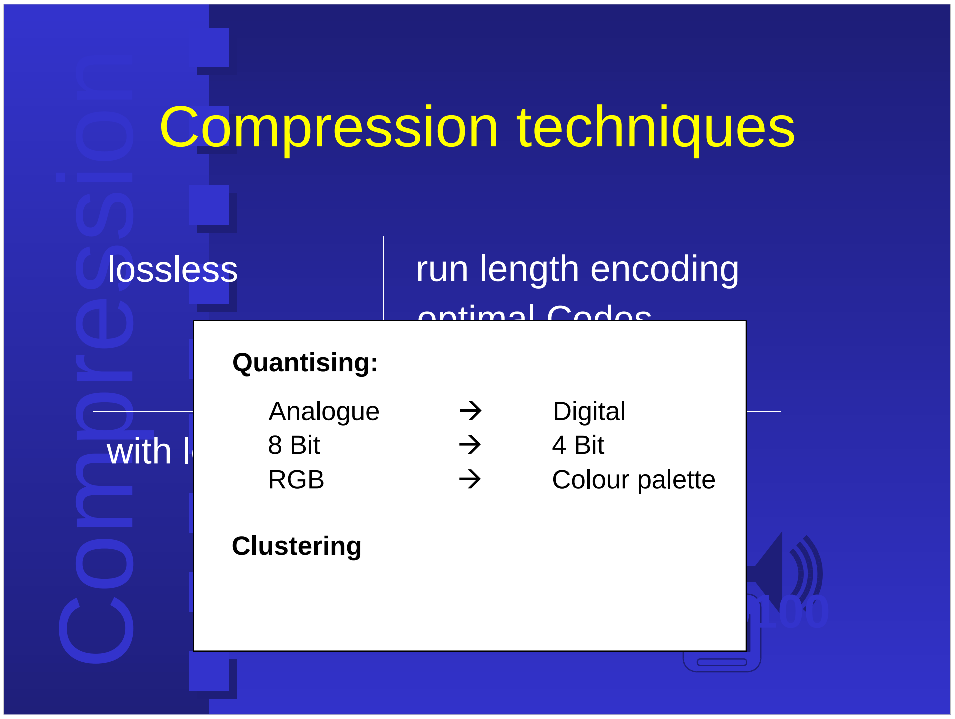
0.263 0.0100001101 0100111

0.262475 0.0100001100 1100001

0.0100001101

0100001101

Huffman: **0110110(0000)**



Compression techniques

lossless

run length encoding

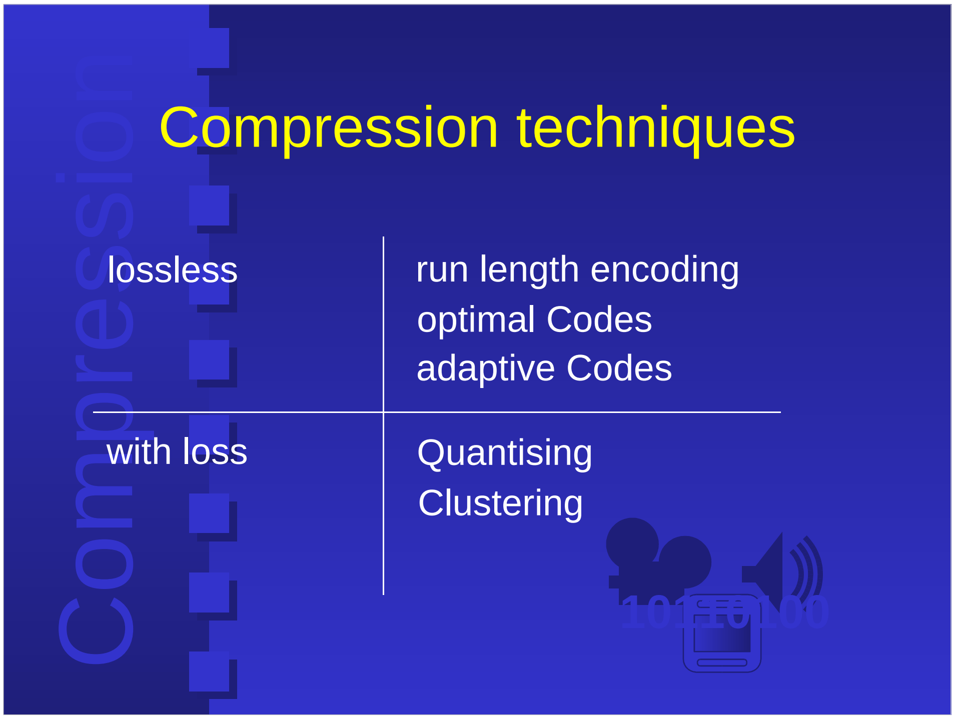
optimal Codes

with l

Quantising:

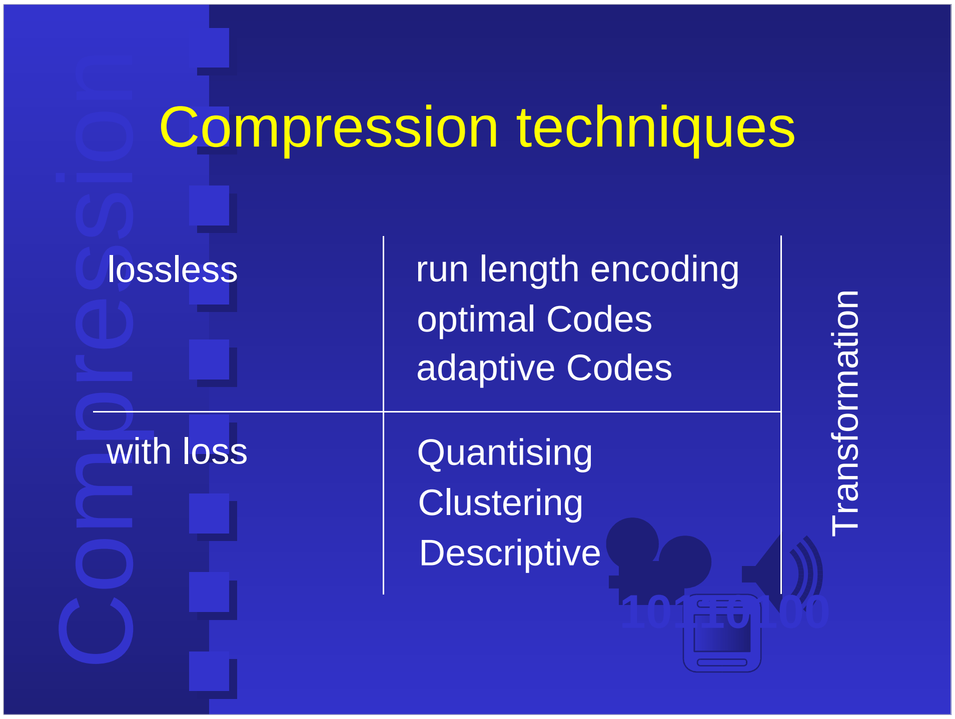
Analogue	→	Digital
8 Bit	→	4 Bit
RGB	→	Colour palette

Clustering



Compression techniques

lossless	run length encoding optimal Codes adaptive Codes
with loss	Quantising Clustering



Compression techniques

lossless	run length encoding optimal Codes adaptive Codes	Transformation
with loss	Quantising Clustering Descriptive	

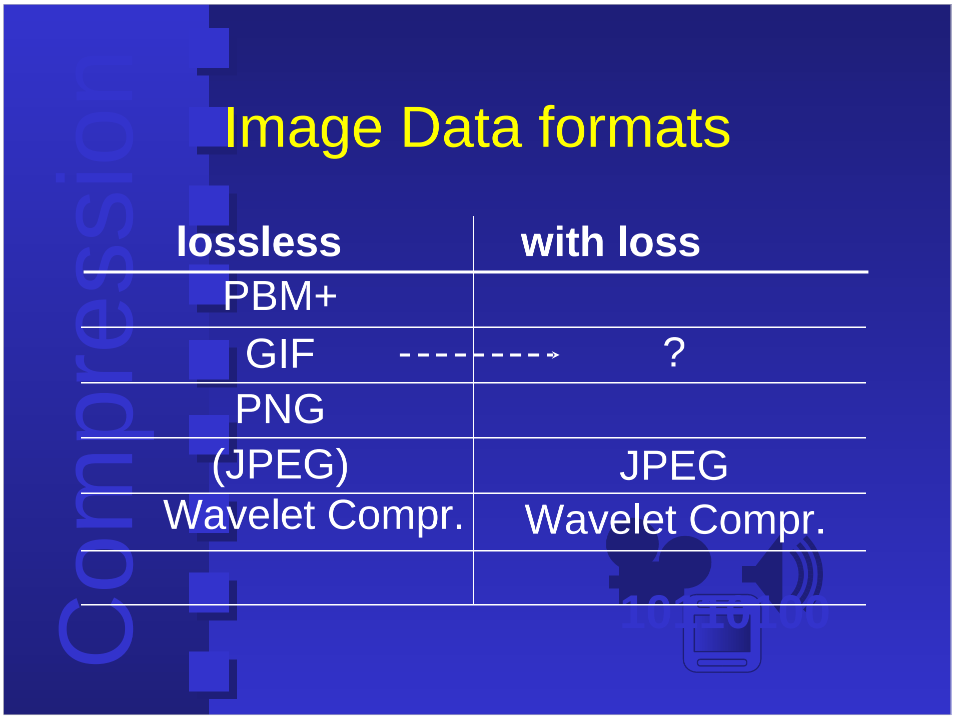
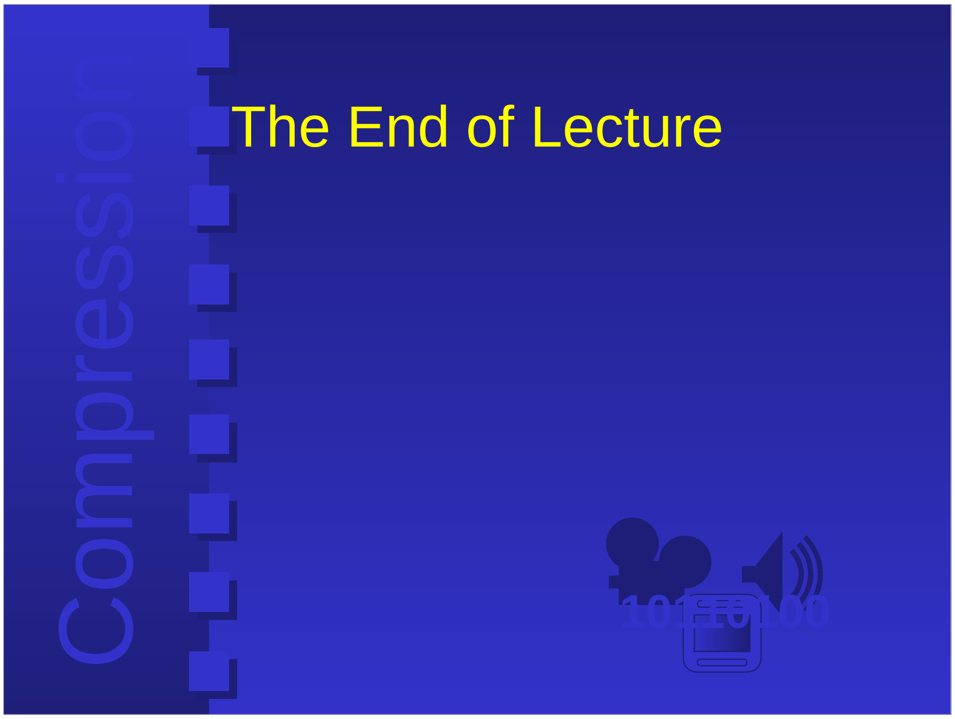

Compression
10110100

Image Data formats

lossless	with loss
PBM+	
GIF ---------→	?
PNG	
(JPEG)	JPEG
Wavelet Compr.	Wavelet Compr.


Compression
10110100

The End of Lecture