Digital Image Processing

Lesson 6: Image Compression

Master Course

Fall Semester 2023

Prof. Rolf Ingold

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Content

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Definition of Image Compression

- **Image compression** is data compression applied to digital images

- The goal is to reduce the amount of information (number of bytes) used to represent an image
  - Needing less storage
  - Allowing faster transmission

- The principle consists of reducing two forms of redundancies
  - **Data redundancy**, due to local correlations
  - **Coding redundancy**, due to non-optimal encoding

- **Compression** : the original image is transformed and encoded into a compressed file
- **Decompression** : the compressed file is decoded and the original image is reconstructed

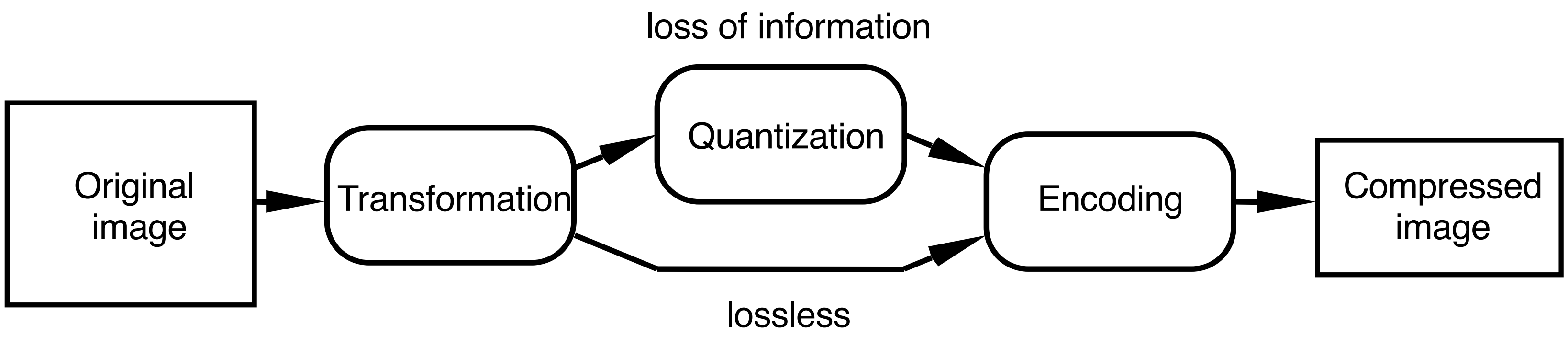
Two Families of Compression Methods

- **Lossless compression**
  - All information is preserved
  - The transformation is reversible
  - Suitable for binary and indexed color images

- **Lossy compression**
  - Trade-off between (visual) image quality and data size
  - Some information is lost
  - The transformation is not reversible
  - Small discrete distortions are introduced
  - Suitable for natural grayscale and color images such as photographs

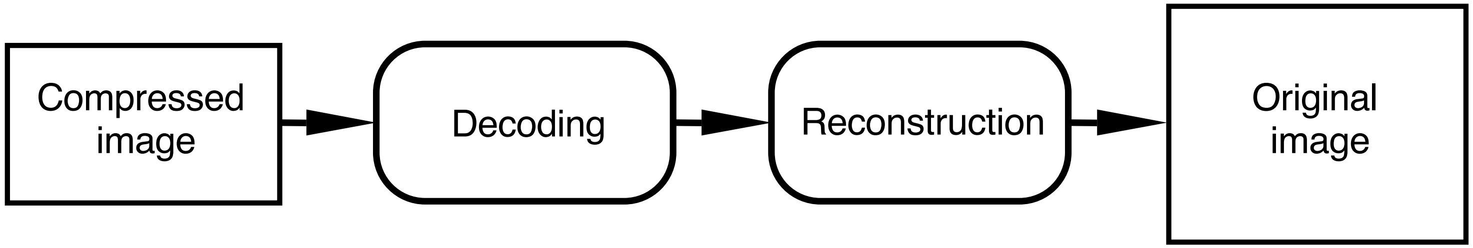
Compression Steps

- **Transformation** performs data decorrelation to **reduce data redundancy**
- **Quantization** performs approximation by reducing the set of possible values
- **Encoding** assigns optimal codes to **eliminate coding redundancy**



Decompression Steps

- **Decoding** restores the values that represent the data in the transformed space
- **Reconstruction** recomputes the data of the original image space



- For some applications, special requirements may be requested
  - **Progressive display** to preview a low quality image while downloading (stepwise) a better quality version

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Encoding

- **Goal of encoding**: represent a sequence of symbols (or values), by minimizing the length of data
  - Assign a binary code to each symbol
  - Must be reversible (can be decoded univocally)

- Types of encoding schemes:
  - **Entropic** encoding
    - **Huffman coding**
    - Arithmetic coding (encumbered by patents!)
  - **Dictionary based** encoding

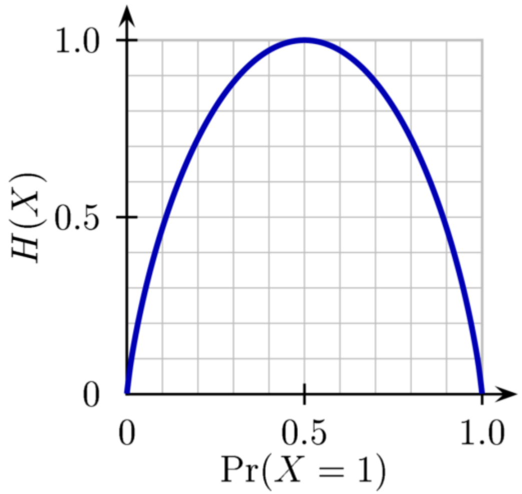
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Information Entropy

- **Entropy** is a basic concept of information theory
- Formal definition (by Shannon): the entropy of a discrete random event X, with possible states $x_1, x_2, ..., x_n$, is defined as

$$H(X) = -\sum_{i=1}^{n} p(x_i)\, log_2\, p(x_i)$$

- The entropy corresponds to a lower bound of the average amount of bits used to represent one event
  - Entropy is maximal and equal to $\log_2 n$ if all states have the same probability
  - Entropy decreases with the differences of the states' probabilities
  - Entropy is minimal and equal to 0 if only one state can occur
  - Example: Bernoulli trial with two states



UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Example: Entropy Computation

- Let us consider an image with 8 grey levels Z in the range $0..7$ with known distributions P_Z; the table below computes its entropy

Z	P_Z	$-\log_2(P_Z)$	$-P_Z \log_2 (P_Z)$
0	0.19	2.396	0.455
1	0.25	2.000	0.500
2	0.21	2.252	0.473
3	0.16	2.644	0.423
4	0.08	3.644	0.292
5	0.06	4.059	0.244
6	0.03	5.059	0.152
7	0.02	5.644	0.113
		Entropy :	2.652

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Information Coding Principles

- The **encoder** translates a sequence of symbols (events) into a bit string
- The **decoder** interprets this bit string to reconstruct the sequence of symbols

- Optimal compression is achieved by using codes with variable lengths
  - Short codes are used for frequent symbols
  - Long codes are used for scarce symbols

- Decoding must be univocal
  - It is guaranteed with codes that are **prefix-free:** no code is a prefix of another code
  - Each bit string must be interpretable: it is either a code or a prefix of a code
- Sequences of **Horn addresses** of binary trees meet these requirements

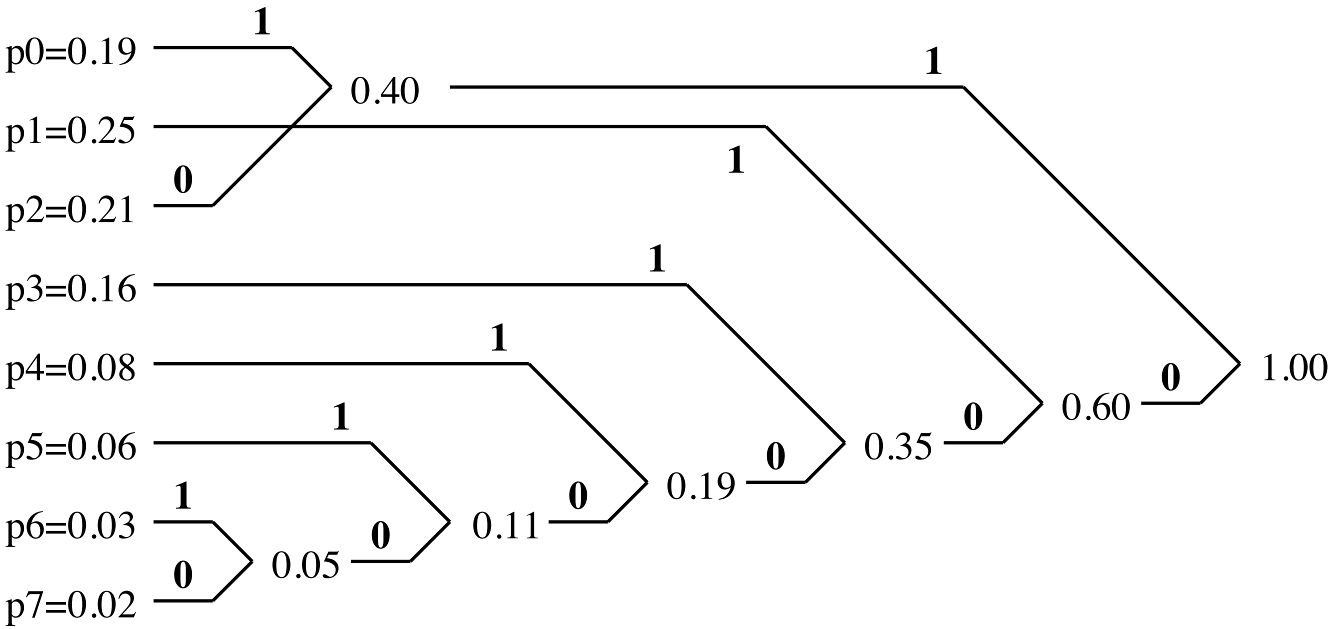
Example: Information Coding

- Evaluation of a variable length code used to represent the gray levels of the previous example of a

Z	$Freq_Z$	code 1	l_1	$Freq_Z \cdot l_1$	code 2	l_2	$Freq_Z \cdot l_2$
0	19	000	3	57	11	2	38
1	25	001	3	75	01	2	50
2	21	010	3	63	10	2	42
3	16	011	3	48	001	3	48
4	8	100	3	24	0001	4	43
5	6	101	3	18	00001	5	30
6	3	110	3	9	000001	6	18
7	2	111	3	6	000000	6	12
		size :		300	size :		281

Example: Building of Huffman Codes

- The **Huffman codes** for the previous example is built by using the binary tree below



$z_0 \to$ **11** $z_1 \to$ **01** $z_2 \to$ **10** $z_3 \to$ **001**

$z_4 \to$ **0001** $z_5 \to$ **00001** $z_6 \to$ **000001** $z_7 \to$ **000000**

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Dictionary Based Coding

- **Dictionary based coding** has initially been developed to compress text
- Principles
  - Sequences of symbols are encoded according to entries in a dictionary
  - To avoid transmitting the dictionary, it is possible to automatically build a dictionary of previously seen strings
- Performance is similar to entropy based coding

- LZW compression schema
  - 8-bits codes (0-255) are used to transmit single byte values
  - 12-bits codes (256-4095) refer to most frequent byte sequences
  - Dictionaries are build dynamically and transmitted implicitly

Example of LZW Compression Principle

Symbols: A B B C A B D A B B C A B B B C A B B A B B A B D

Code: 0 1 1 2 4 3 4 6 10 11 5 10 8

Dictionary:

0	A	4	AB	8	ABD	12	ABBB
1	B	5	BB	9	DA	13	BCAB
2	C	6	BC	10	ABB	14	BBA
3	D	7	CA	11	BCA	15	ABBA

Limits of Entropic Encoding

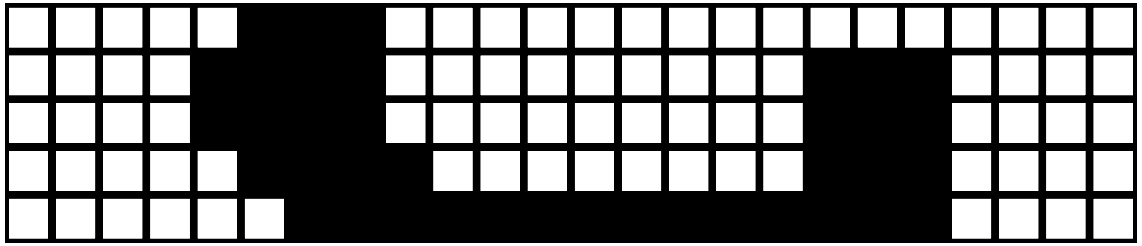
- Entropic encoding is suitable to remove **coding redundancy**

- Images generally don't have much coding redundancy
  - Entropy encoding alone does not allow to compress image data significantly!

- The role of preliminary transformations is to reduce data redundancy by transforming it into coding redundancy, which can later be eliminated by using entropic coding

Run-Length Encoding (RLE)

- RLE is a very simple form of lossless data compression
- Lengths of horizontal runs (sequences of same pixel values) are stored as a single number
- Example :
  - Original image **using 120 bits**

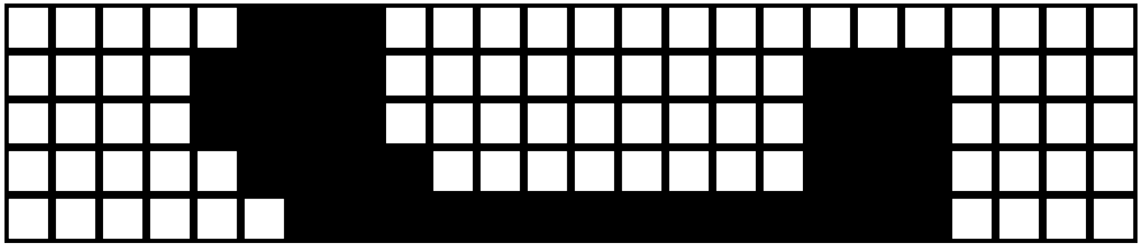


  - RLE: **5, 3, 16, 4, 4, 9, 3, 4, 4, 4, 9, 3, 4, 5, 4, 8, 3, 4, 6, 14, 4** using 21x5 = **105 bits** (assuming five bits are used for each number)

UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

2D-Run-Length Encoding

- Principle : apply RLE on scan line differences



ABS, 5, 3, ABS, FL,	0 1010 011 0 0000
REL, -1, 0, ABS, 9, 3, ABS, FL,	1 10 0 0 11011 011 0 0000
REL, 0, 0, REL, 0, 0, ABS, FL,	1 0 0 1 0 0 0 0000
REL, +1, +1, REL, 0, 0, ABS, FL,	1 10 10 1 0 0 0 0000
ABS, 6, 14, ABS, FL	0 1011 1111010 0 0000

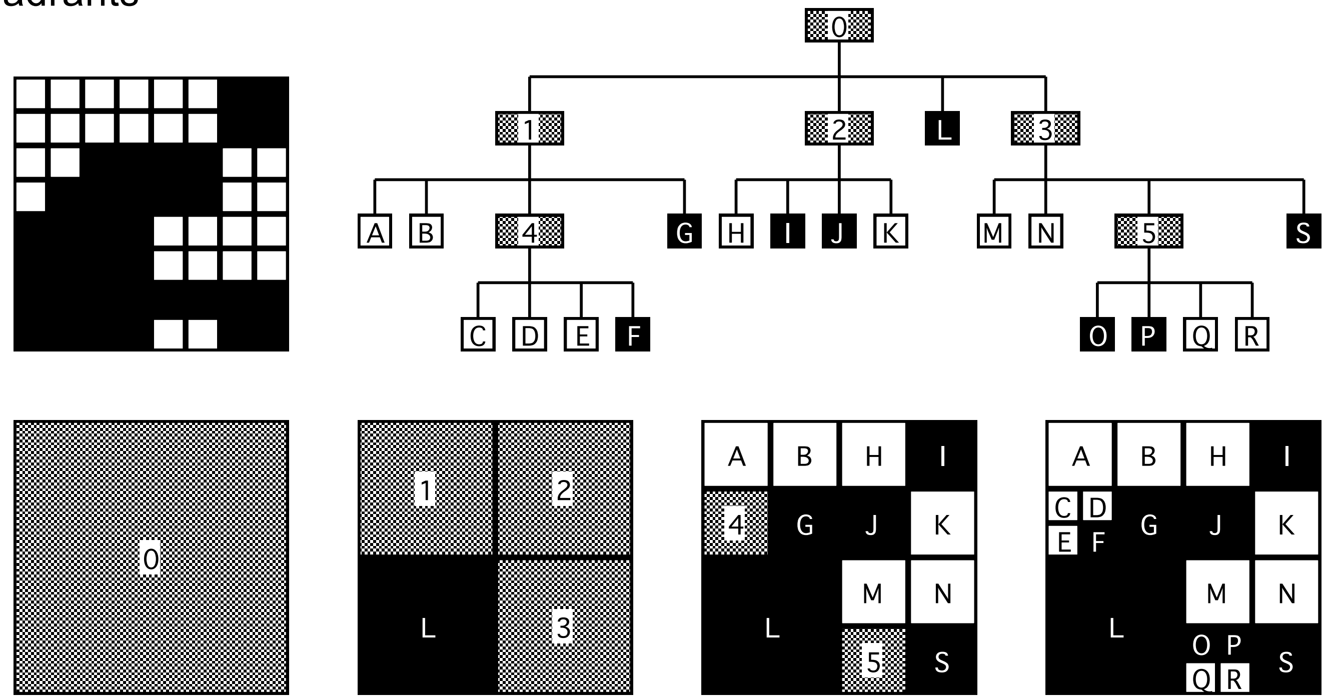
code table

ABS	0
REL	1
FL	0000
0	0
-1	10
+1	11
0	0001
1	001
2	010
3	011
4	100
5	1010
6	1011
6+...	11...

- Size of the compressed encoding: 13 + 18 + 11 + 13 + 17 = **72 bits**

Quadtree Data Structure

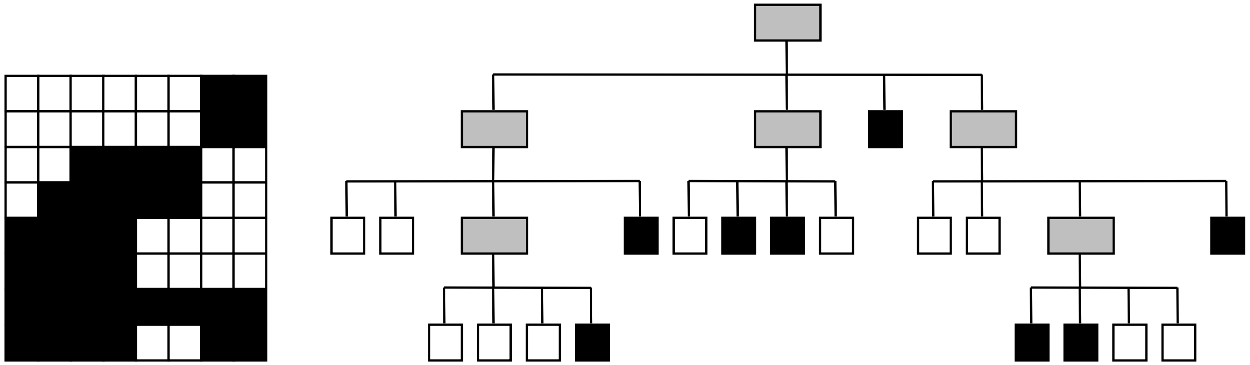
- A **quadtree** is a data structure used to partition an image by recursively subdividing it into four quadrants



- The process stops when a node represents a homogeneous area

Quadtree compression

- Quadtrees can be used for lossless data compression
  - The quadtree is represented as a list of nodes in
  - Example: Original image uses **64 bits**



  - Quadtree sequence: **XXWWXWWWBBXWBBWBXWWXBBWWB** uses 25x2 = **50 bits** (assuming two bits are used for each symbol)

UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Lossless Compression of Grayscale Images

- Lossless grayscale image compression can be achieved
  - By using binary image compression applied to a bit-plane decomposition
  - By decorrelating adjacent pixels and using entropy coding

- Performance is limited!

Bit-plane Decomposition

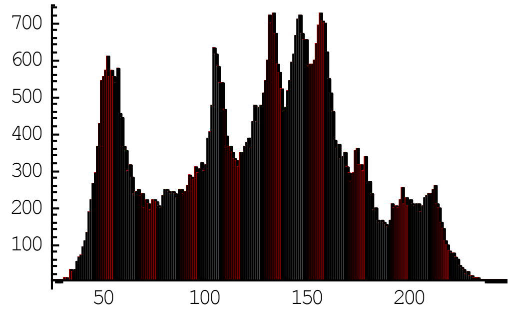
- A binary image with 256 gray levels can be represented by 8 bit-planes a_0, a_1, … a_7.
    - The higher planes (a_7, …) are more compressible than the lower

- A higher compression is achieved with the Gray planes defined as $g_i = a_i \oplus a_{i+1}$

Entropy Encoding for Gray-Level Images

- Usually, gray-level images don't have much coding redundancy

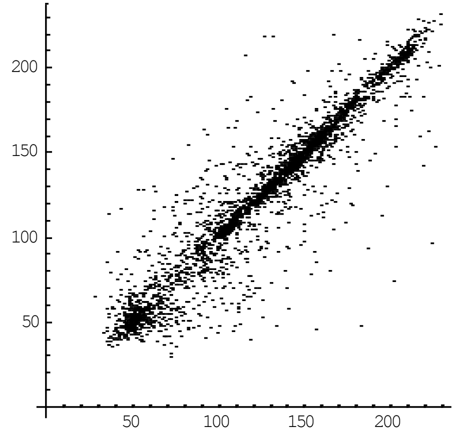
- Example:
  - The original Lena picture with **65'536** pixels and 256 gray levels has an entropy of 7.394
  - The theoretical minimal size is **60'572 bytes**
  - Using Huffman codes, the picture can be compressed into **60'825 bytes** (plus 481 bytes for the codebook)



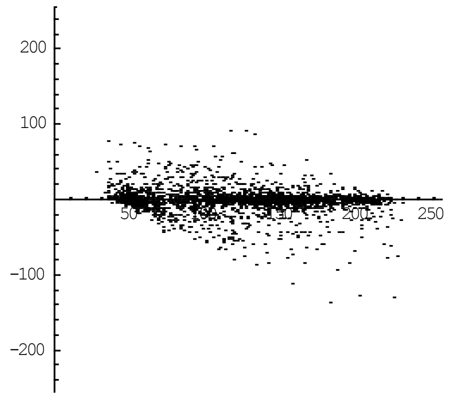
UNI
FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Data Correlation for Adjacent Pixels

- In natural images, adjacent pixels are highly correlated
- Example (from the Lena picture) with $z_1 = f(2x, y)$, $z_2 = f(2x+1, y)$



distribution of (z_1, z_2)



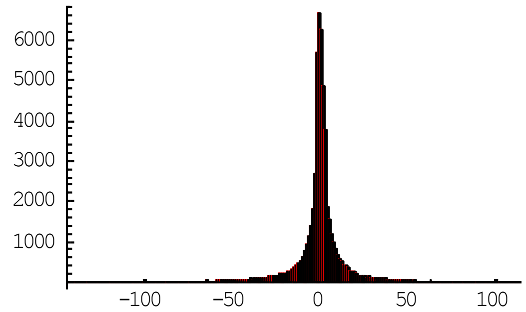
distribution of $(z_1, z_2\text{-}z_1)$

- The distribution of $z_2\text{-}z_1$ is much narrower than the distribution of z_2
  - It allows a much higher compression

Decorrelation by Adjacent Pixel Differences

- Replacing pixel values by pixel differences
  - Narrows the distribution
  - Reduces the entropy

- Example:
  - In the Lena picture pixel differences have an entropy of 5.277
  - They require globally **43'321 bytes**



UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Lossy Compression of Gray-Level Images

- Lossy compression consists of
  - Transforming spatial data in order to reduce data correlation
  - Quantization of the values in transformed space
  - Encoding the data by eliminating code redundancy

- Type of transformations used
  - Hotelling transform, based on principal component analysis (PCA)
  - Fourier Transform
  - Discrete Cosine Transform (DCT)
  - Walsh-Hadamard Transform
  - Haar Transform, Wavelets
  - ...

Hotelling Transform

- The **Hotelling transform** is a linear transform that optimally decorrelates a data bloc
  - It is calculated as follows
    - Estimation of the covariance matrix

$$\mathbf{C} = E\{(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^t\} = \frac{1}{n}\sum_{k=1}^{n} x_k x_k{}^t - \mathbf{m}\mathbf{m}^t \qquad \mathbf{m} = E\{\mathbf{x}\} = \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_k$$

    - Finding the eigenvectors (i.e., an orthonormal matrix) $\mathbf{A}$ such as

$$\mathbf{A}\mathbf{C}\mathbf{A}^t = \begin{pmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_d \end{pmatrix}$$

    - Then, the **Hotelling transform** is defined as $\mathbf{y} = \mathbf{A}^t(\mathbf{x}-\mathbf{m})$ is

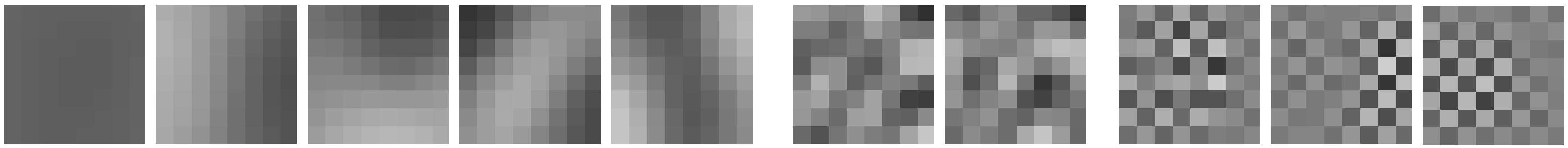
- The **reverse Hotelling transform** transform is obtained by $\mathbf{x} = \mathbf{A}\mathbf{y}+\mathbf{m}$

UNI
FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Hotelling Transform (cont.)

- The Hotelling transform is applied on blocks of pixels (8x8 or 16x16)
- The Hotelling transform can be understood as a linear transform using a set of data specific basis functions
- These functions are ordered according to decreasing eigenvalues

- Example :
  - the 64 8x8 basis functions of the Lena Pictures are listed below with their index



1 2 3 4 5 ... 16 17 ... 62 63 64

UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Quantization of Hotteling Transform

- The Hotteling Transform is a reversible transform: the original image can be rebuilt from the Hotteling Transform coefficients

- For lossy compression, the coefficient of the Hotteling Transform are quantized, using two complementary principles
  - A large subset of of the coefficients are ignored (i.e., put to 0)
  - Possible values are restricted to multiples of a quantization step

$$\hat{y} = q \,\mathrm{Round}(y / q)$$

- Entropy coding used for both
  - Encoding of the basis function
  - Transform coefficients

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Example of Hotelling-Based Compression

- Original image compared to
  - 1/4 of coefficients quantized with a step of 16
    (size = **3'834 bytes**, compression rate = **~1:17**)
  - 1/8 of coefficients quantized with a step of 16
    (size = **2'152 bytes**, compression rate = **~1:30**)



UNI
FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Wavelet Transforms

- Wavelet transforms provide an interesting alternative to Fourier transforms

- Wavelet transforms convert a signal into a series of wavelets
  - Using basis functions, bounded in frequency as well as in space domains

- There exist an infinity of wavelets transforms
  - The first known discrete wavelets transform was invented by Alfred Haar in 1909
  - In the late 1980s and early 1990s, Stéphane Mallat made some fundamental contributions to the development of wavelet theory
  - Orthogonal wavelets with compact support were introduced by Ingrid Daubechies in 1988
  - Since then, many other wavelets have been invented

General Form of Continuous Wavelet Transforms

- Mathematicaly, the continuous wavelet transform (CWT) is defined by

$$\gamma(\tau,\sigma) = \int_{-\infty}^{+\infty} x(t) \frac{1}{\sqrt{\sigma}} \psi_{\sigma,\tau}(t) dt \qquad \psi_{\sigma,\tau}(t) = \frac{1}{\sqrt{\sigma}} \psi\left(\frac{t-\tau}{\sigma}\right)$$

where τ represents translation and σ a scale factor of the mother wavelet $\psi(\tau)$

- Then, the original function can be reconstructed using the inverse transform

$$x(t) = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \gamma(\tau,\sigma) \psi\left(\frac{t-\tau}{\sigma}\right) d\tau \frac{d\sigma}{|\sigma|^2} \qquad C_\psi = \int_{-\infty}^{+\infty} \frac{|\Psi(\zeta)|^2}{|\zeta|} d\zeta$$

where Ψ is the Fourier transform of ψ

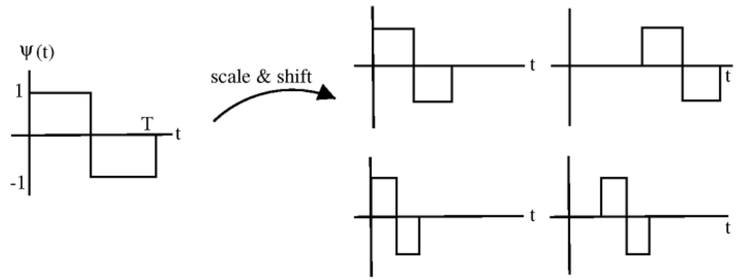
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Haar Wavelets

- The Haar wavelet can also be described as a step function

$$\psi(x) = \begin{cases} 1 & 0 \leq x < \frac{1}{2} \\ -1 & \frac{1}{2} \leq x < 1 \\ 0 & otherwise \end{cases}$$

- Haar transforms can be understood as a combination of Haar wavelets with different scale and shift parameters



UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

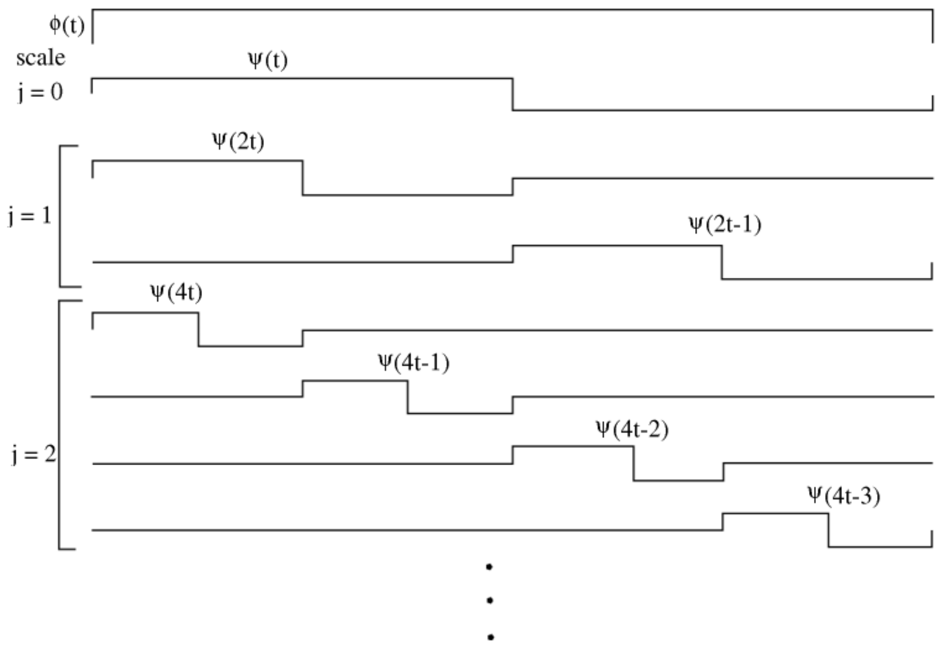
Haar Transform

- The Haar transform can be expressed as a combination of the following basic functions
  - A step function

$$\phi(t) = \begin{cases} 1 & 0 \le t < 1 \\ 0 & otherwise \end{cases}$$

  - Several wavelets (downscaled and shifted)

$$\psi_i^j(t) = \psi(2^j t - i)$$



UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Computation of the Haar Transform

- The Haar Transform can be computed stepwise as follows
  - Separate odd samples and even samples
  - Compute their sum (mean value) and (half of) their differences

$$s_{n-1,l} = (s_{n,2l} + s_{n,2l+1})/2$$
$$d_{n-1,l} = (s_{n,2l+1} - s_{n,2l})/2$$

  - At the end, keep the last sum and all differences

- Example : the Haar transform of **`[8,4,9,7]`** is **`[7,-1,2,1]`**

```
[ 8    4    9    7 ]
  [ 6         8 ]                [ 2          1 ]
      [ 7 ]                          [ -1 ]
```

- The Haar transform of N samples can be computed in O(N) time !

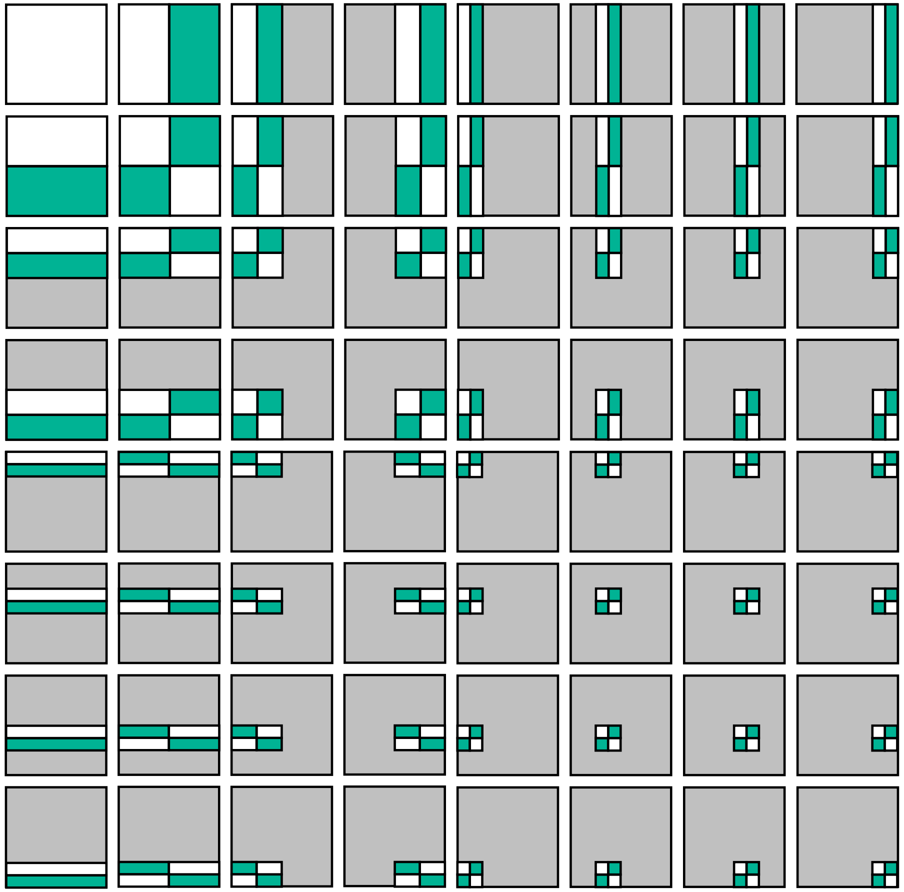
UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Haar Transform in 2D space

- The standard Haar decomposition in 2D space is obtained by computing
  - A 1D Haar transform on each row
  - Followed by a 1D Haar transform on each column

  (or conversely)

- The corresponding basis functions are illustrated on the right



UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Illustration of the Haar Transform

- The wavelet transform has many zero or near zero values !

original Lena picture and
its Haar transform (with adjusted brightness (+128) and contrast (×8))

Wavelet Based Compression

- Simple entropy coding of Haar wavelets can be used for loss-free compression

- Lossy compression is obtained by
  - Ordering coefficients according to their power
  - Removing less significant coefficients
  - Quantizing the other coefficients
  - Applying entropy coding

Illustration of Haar Based Compression

- Illustrartion of the Original Lena picture and the results of its Haar based compression, when removing 7/8 and 15/16 of coefficients

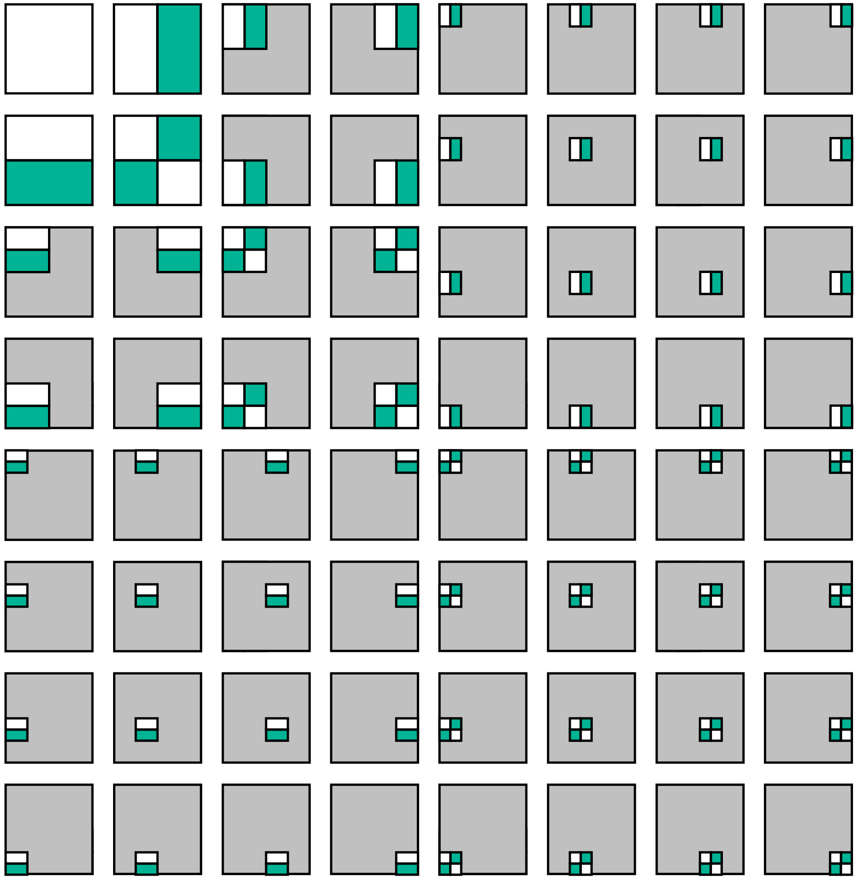
compression 1:7.7
PSNR=31.51dB

compression 1:14
PSNR=28.34dB

UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Dyadic Wavelet Transform

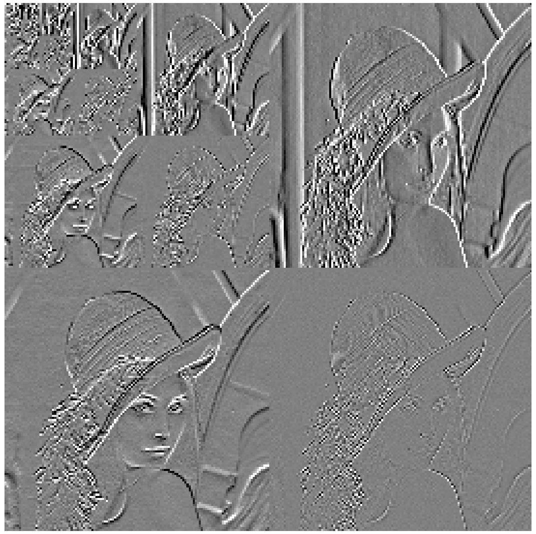
- A more efficient representation for entropy coding can be achieved by using a slightly modified 2D Haar Transform
- Alternating between rows and columns is applied at each decomposition steps
- The resulting basis functions are illustrated on the right



UNI FR
UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Illustration of Dyadic Wavelet Transform

- The resulting transform contains multi-resolution square subimages

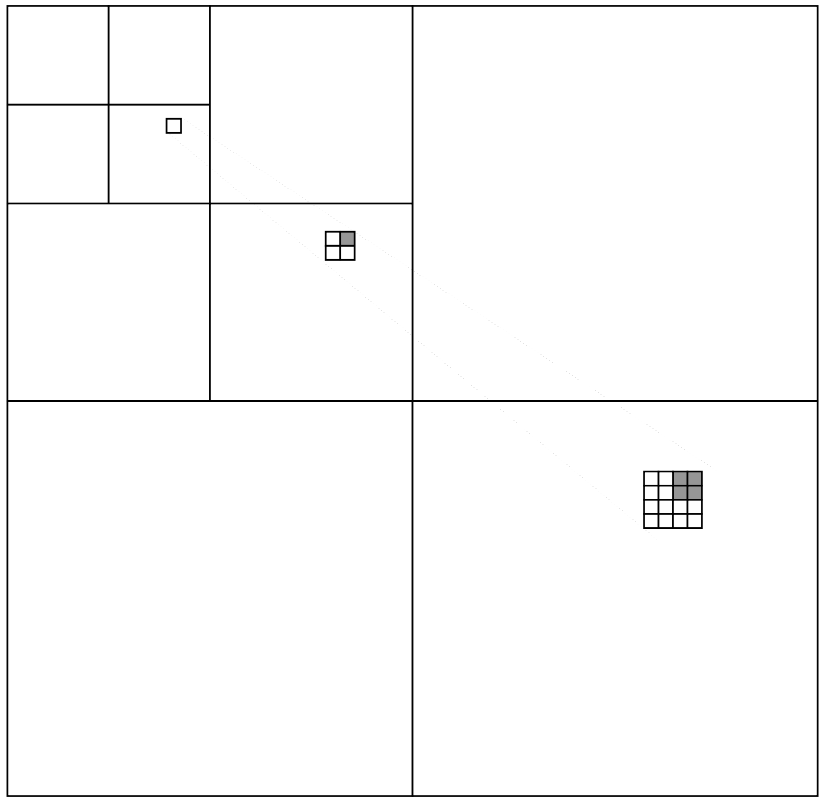


original Lena picture and its Haar transform,
with adjusted brightness (+128) and contrast (×8)

Encoding of Dyadic Wavelet Coefficients

- Optimal coefficient encoding uses the quadtree structure of multi-resolution subimages

- Various encoding schemes have been proposed
  - EZW (embedded zerotree wavelet)
  - SPIHT (set partitioning in hierchical trees)
  - WDR (wavelet difference reduction)
  - ...



UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Color Image Compression

- Color image compression could be done channel-wise
  - Not very efficient because it ignores color correlation

- Principle Component Analysis in RGB space can be used to decorrelate color information

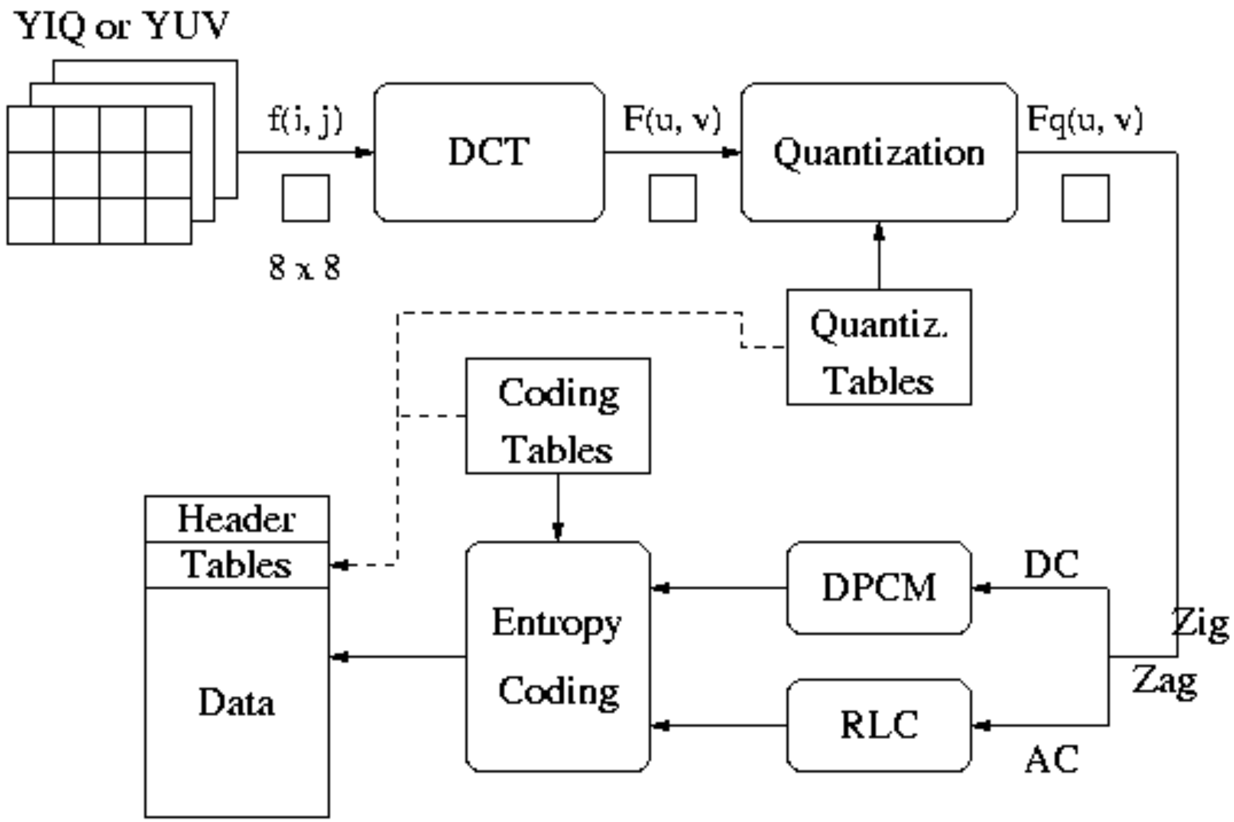
- Other methods work in "physiological" color spaces :
  - Using different sampling rates
  - Using different quantization steps

  for luminance and chrominance

Principles of JPEG Compression

- JPEC Compression uses the following principles
  - Luminance and chrominance are separated
  - Chrominance resolution is reduced by a factor 2
  - Image split into 8x8 blocks compressed individually
  - DCT is computes for each block
  - DCT coefficients are quantized using a special tables
  - 0 values are ignored
  - Remaining values are encoded using entropy coding
  - Depending of the similarity with the previous block, absolute or differential coding is used

Schema of JPEG compression



UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

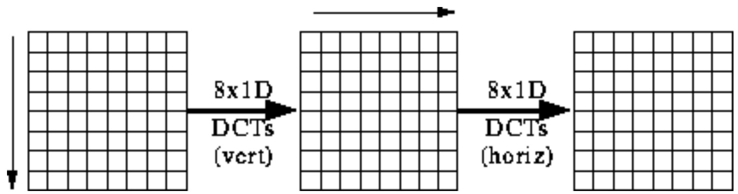
DCT basis functions

- The discrete cosine transform (DCT) is a Fourier-related transform using only real numbers
- The Discrete Cosine Transform is defined as

$$F(u,v)=\alpha(u)a(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

with
$$\alpha(u)=\begin{cases}\sqrt{\frac{1}{N}} & u=0\\ \sqrt{\frac{2}{N}} & u>0\end{cases}$$

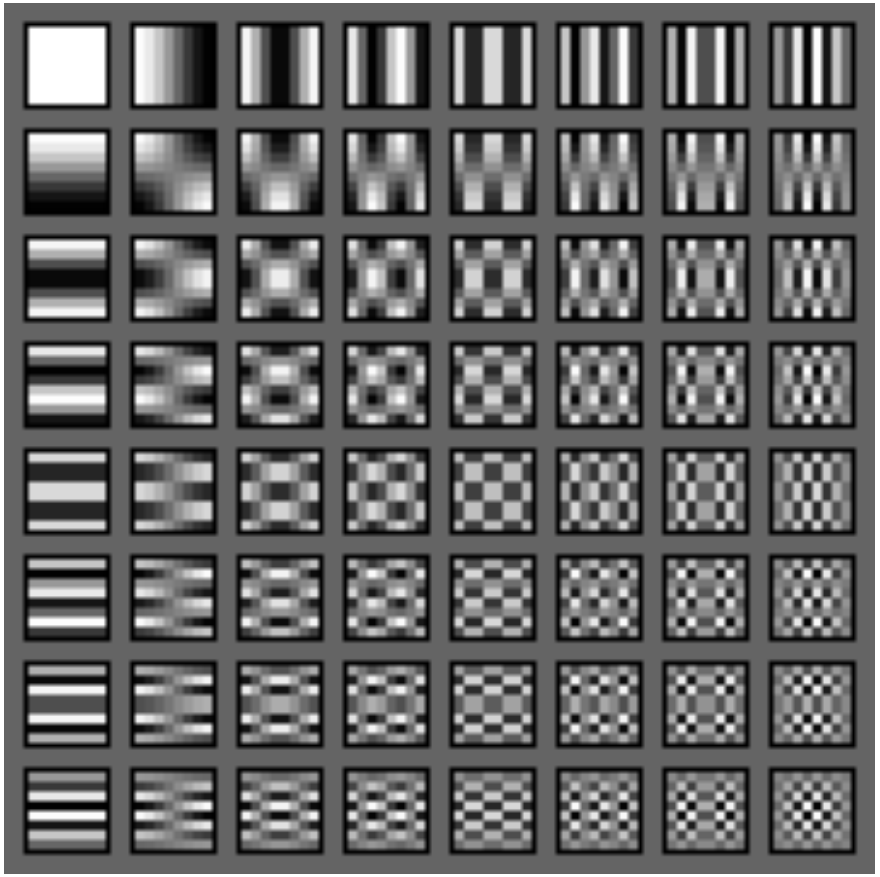
- The DCT is separable and can be computed using a fast algorithm



- The reverse transform is similar to the direct transform (but $\alpha(u)$, $\alpha(v)$ factors being inside the summation expression)

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

DCT basis functions



Quantization Tables in JPEG

- Quantization tables are used to approximate
  - Different quantization is performed on luminance and chrominance

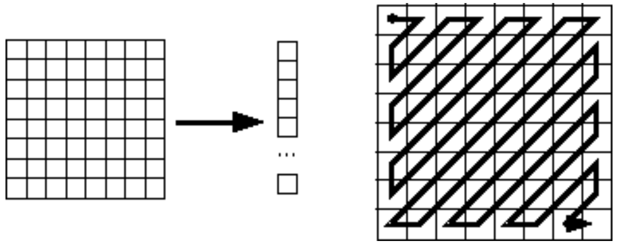
16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

- Quantization error is the main source of the lossy compression.
- Quantization tables can be scaled to adjust the quality factor.

JPEG Block Encoding Scheme

- Encoding is performed in three steps
  1) Zig-zag scan to transformed 8 x 8 blocs into 1 x 64 vectors



  2) Two type of encoders
     a) DPCM : Differential Pulse Code Modulation (encode the difference from previous 8 x 8 blocks), on DC components
     b) RLE : Run Length Encode (RLE) on AC components (supposed to contain a lot of 0)
  3) Entropy based coding of values