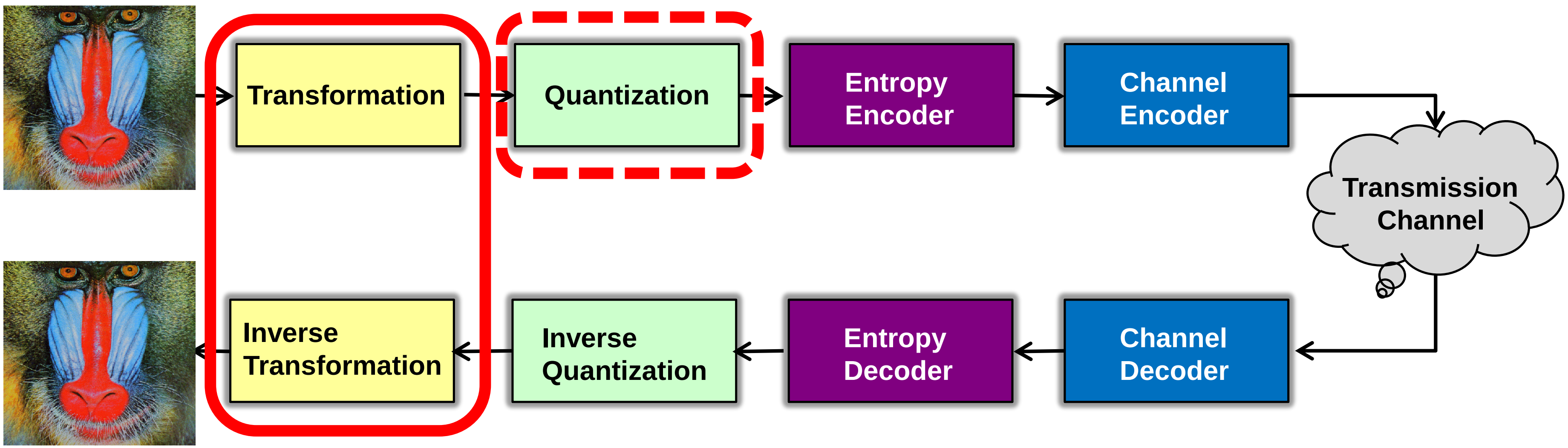
TECHNISCHE UNIVERSITÄT CHEMNITZ

Chapter 7:
Discrete Cosine Transformation (DCT)

Motivation

- Data compression (e.g. JPEG)

Working Principle



General Information

- Multiple different transformations, most common: DCT-I, DCT-II, DCT-III, DCT-IV
- “The DCT” usually means DCT-II
- DCT-II and DCT-III are each others inverse

DCT (one-dimensional)

$$X[k] = c_k \cdot \sum_{n=0}^{N-1} x[n] \cdot \cos\left\{ \frac{\pi}{N} \cdot \left(n + \frac{1}{2} \right) \cdot k \right\}; \qquad k = 0,1,2 \ldots N-1$$

$$c_k = \begin{cases} \sqrt{\frac{1}{N}}; & k = 0 \\ \sqrt{\frac{2}{N}}; & k \neq 0 \end{cases}$$

Inverse DCT (one-dimensional)

$$x[n] = \frac{2}{N} \cdot \sum_{k=0}^{N-1} X[k] \cdot \cos\left\{ \frac{\pi}{N} \cdot \left(n + \frac{1}{2} \right) \cdot k \right\}; \qquad n = 0,1,2 \ldots N-1$$

DCT (two-dimensional)

x, y – local range
u, v – frequency range

$$S[u,v] = \frac{1}{4} * \ c_u\ (u) * c_v(v) * \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} s[x,y] * \cos[\alpha\ (2x+1)\ u] * \cos[\alpha\ (2y+1)\ v]$$

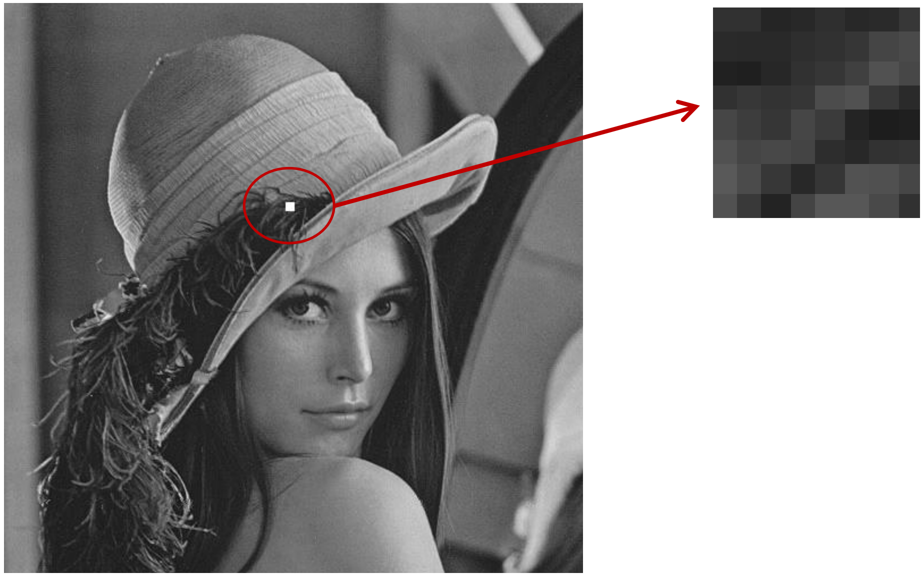
Inverse DCT (two-dimensional)

$$s[x,y] = \frac{1}{4} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} S[u,v] * cos[\alpha\ (2u+1)x] * cos[\alpha\ (2v+1)\ y] * \ c_u\ (u) * c_v(v)$$

$$with \quad c_u(u) = \begin{cases} \frac{1}{\sqrt{2}}, & u = 0 \\ 1, & u \neq 0 \end{cases} \qquad c_v(v) = \begin{cases} \frac{1}{\sqrt{2}}, & v = 0 \\ 1, & v \neq 0 \end{cases} \quad and \quad \alpha = \frac{\pi}{2N}$$

Transformation of an image (1)

- Image is divided into blocks of size NxN
- Each block is transformed independently
- Trade-off between image quality and compression



Transformation of an image (2)

Original image, size 512x512

DCT

DCT coefficients, 8x8 blocks

IDCT

Image, reconstructed from the DCT coefficients

Quantization

- In case of 8 bit image → range of S_Q is [-128…127]

105	115	125	133	137	137	137	137
115	129	133	139	145	139	139	139
127	137	147	153	143	139	139	139
145	149	151	147	147	145	145	145
145	147	149	151	151	137	137	137
149	149	149	149	147	141	141	141
151	151	149	153	151	141	141	141
151	151	149	149	153	143	143	143

8x8 pixel block s(x,y)

DCT

1135	-2	-24	-10	4	-3	-5	3
-45	-35	-12	-6	-6	0	1	-2
-22	-19	-3	3	0	-2	-1	0
-14	-4	0	3	2	0	0	1
-1	-2	3	3	0	-1	1	3
4	0	3	-1	-2	3	2	-2
-3	-1	-1	-3	-1	3	2	-2
-5	3	-8	-4	4	2	-1	-1

DCT coefficients S(u,v)

K

8	16	19	22	26	27	29	34
16	16	22	24	27	29	34	37
19	22	26	27	29	34	34	38
22	22	26	27	29	34	37	40
22	26	27	29	32	35	40	48
26	27	29	32	35	40	48	58
26	29	29	34	38	46	56	69
27	29	35	38	46	56	69	83

Quantization coefficients Q(u,v)

142	0	-1	0	0	0	0	0
-3	-1	0	0	0	0	0	0
-1	-1	0	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

Quantized DCT coefficients S_Q(u,v)

Data compression based on quantization

(basic idea, current JPEG compression is much more complex)

- A lot of quantized coefficients are very small.
- These small coefficients will not be stored.
  → reduction of the image size
- Loss of information
  → visible, if too much information was lost

142	0	-1	0	0	0	0	0
-3	-1	0	0	0	0	0	0
-1	-1	0	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

Quantized DCT coefficients $S_Q(u,v)$

Image,
reconstructed with the
non-quantized DCT coefficients

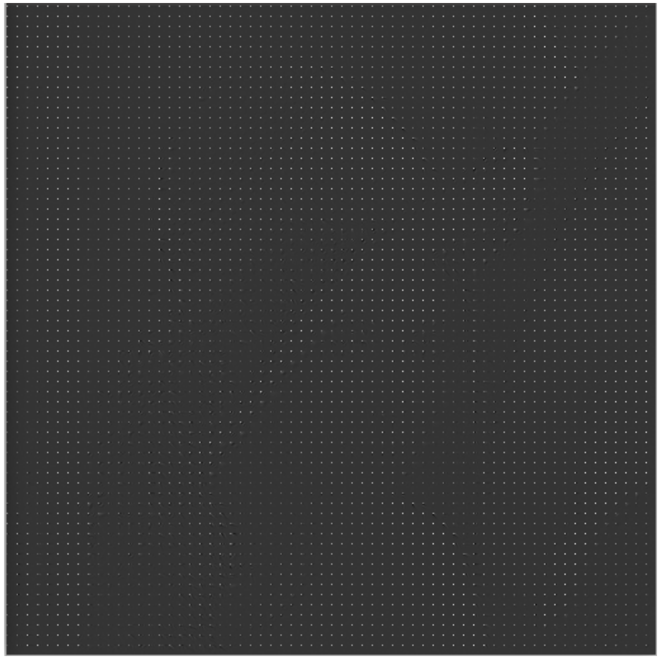
Image,
reconstructed with the
quantized DCT coefficients

Seventh Exercise

- Implement the Discrete Cosine Transform and its inverse.

Expected Output (1)

Original image



DCT coefficients

Expected Output (2)

Reconstructed image (examples for 3 different quantization levels)