

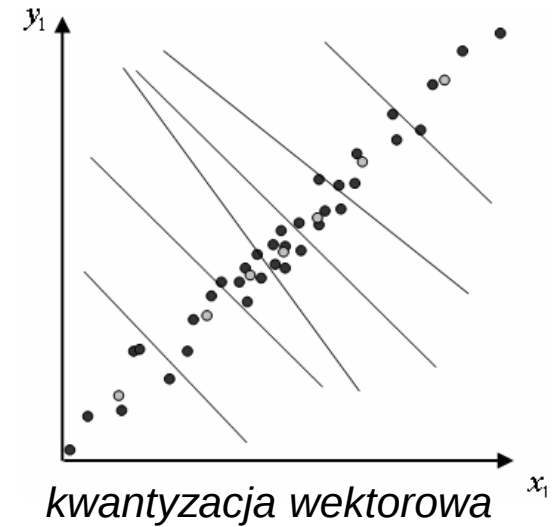
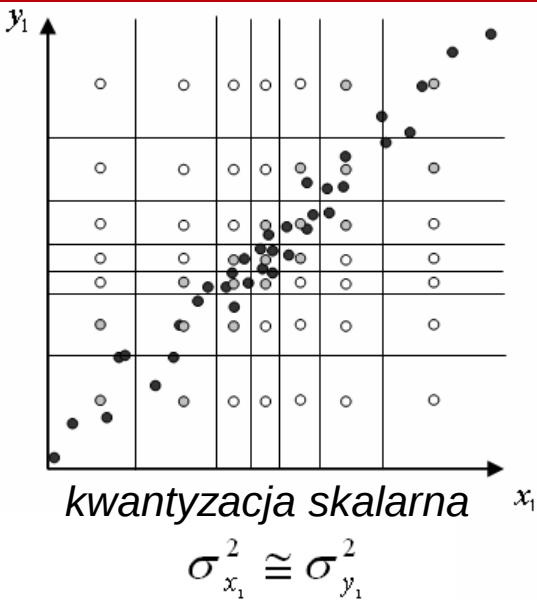
KODOWANIE TRANSFORMACYJNE

Materiały KODA, A,Przelaskowski

- Dane: obrazy, dźwięk
 - Uproszczenie kwantyzacji przez transformację
 - Dobór postaci transformacji
 - Metoda kompresji
 - Kwantyzacja
 - Testy
-

NARODZINY KONCEPCJI

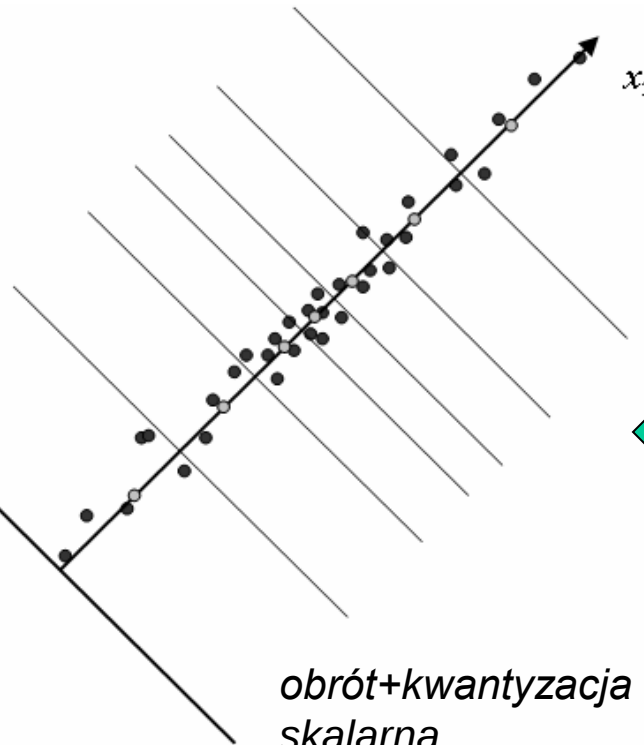
Koncepcja kodowania: uproszczenie kwantyzacji



Upakowanie →

$$\sigma_{x_2}^2 \gg \sigma_{y_2}^2$$
$$\sigma_{x_1}^2 + \sigma_{y_1}^2 = \sigma_{x_2}^2 + \sigma_{y_2}^2$$

← Transformacja



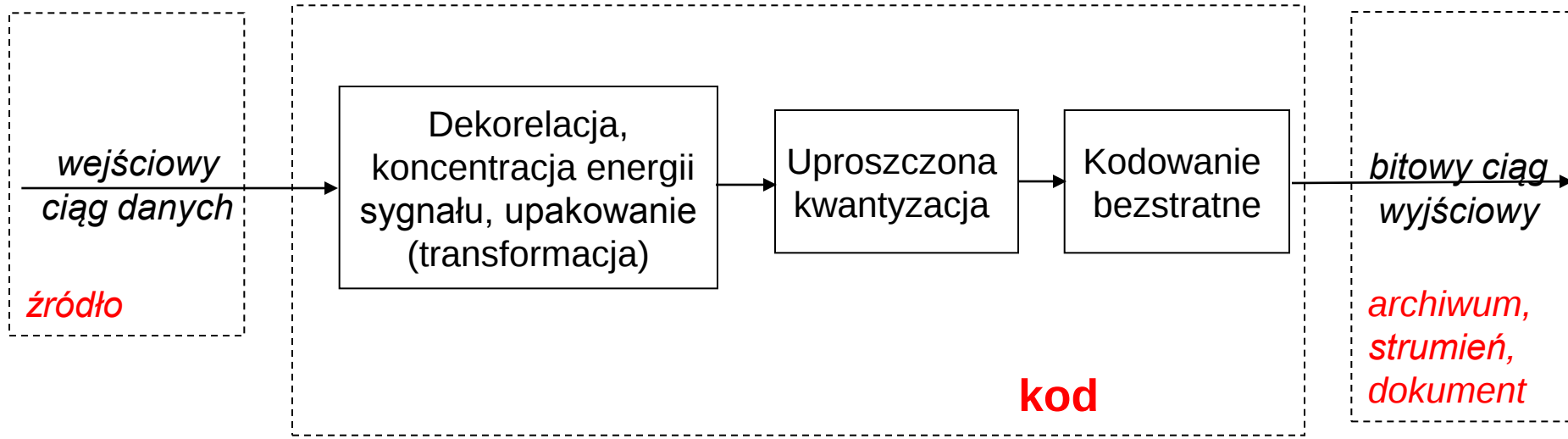
KLT (T, Karhunen-Loeve) – quasi-optymalna, ale niekorzystna w implementacji

- Wyznaczamy macierz kowariancji
- Diagonalizujemy macierz kowariancji
- Utrzymujemy zbiór wartości własnych i wektorów własnych
- Ustalamy zbiór wektorów własnych jako bazę transformacji
- Uzyskujemy zdekorelowany sygnał

$$\mathbf{R}_{\mathbf{xx}} = \mathbf{X}\mathbf{X}^T = \boldsymbol{\Psi}(\mathbf{xx}^T)\boldsymbol{\Psi}^T = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$$

PARADYGMATY

Paradygmat klasyczny



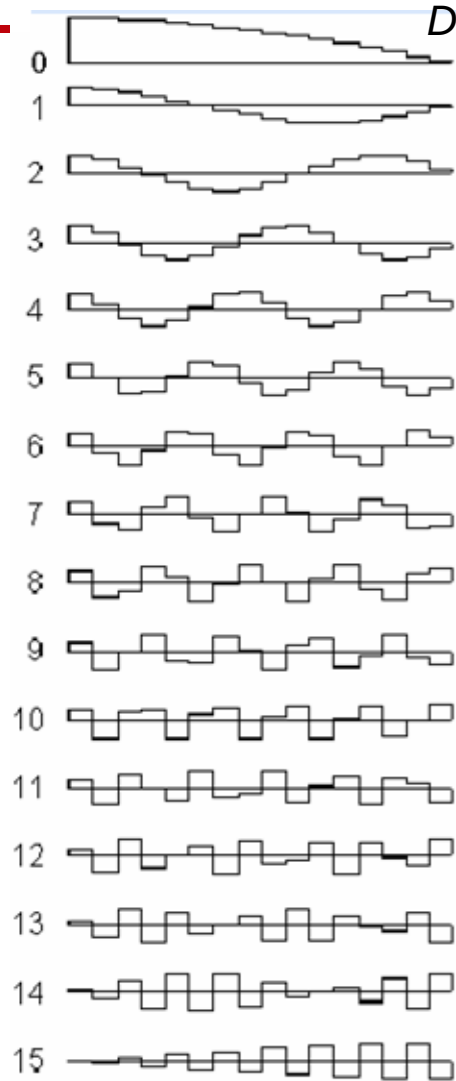
Transformacja

Pożądane cechy:

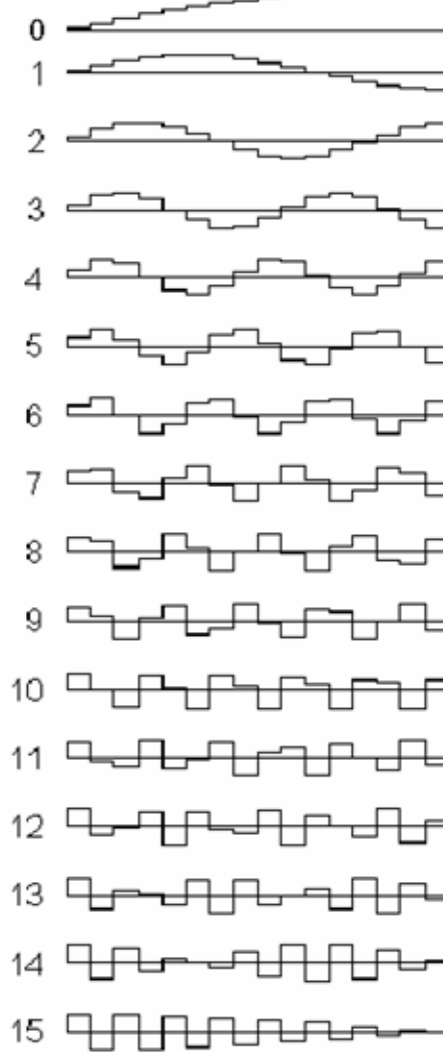
- Dopasowana baza transformacji (dopasowana lokalnie do sygnału, elastyczna, zupełna, najlepiej niezależna od sygnału)
 - Kryterium: dekorelacja danych (usuwanie zależności statystycznych), upakowanie energii wzdłuż wybranych kierunków (wokół elementów bazy), reprezentacja rzadka
 - Szybka implementacja (samej transformacji jak i procedury doboru transformacji)
-

Porównanie baz DCT, DST i KLT

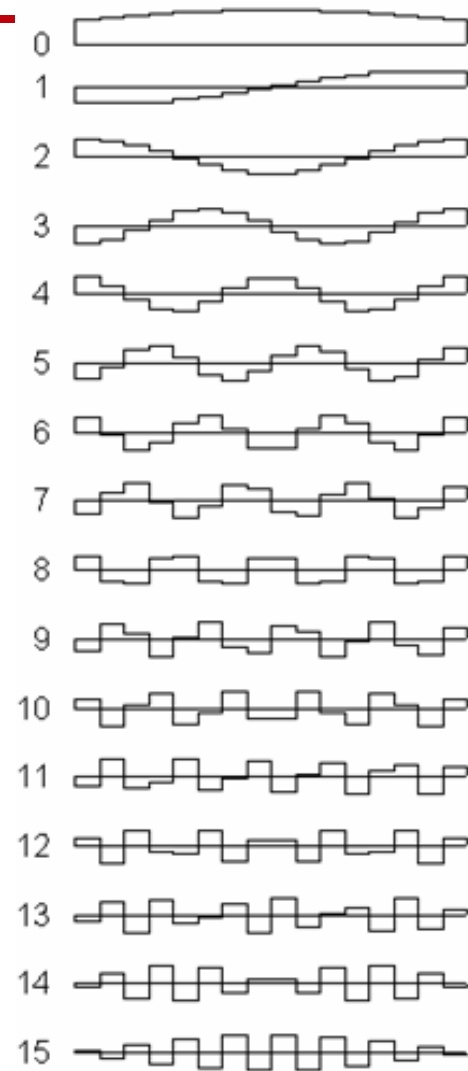
DCT



DST



KLT, Markowa 1 rzędu, $\rho=0,9$



$$C_{i,k}^{\text{IV}} = \sqrt{\frac{2}{n}} \left[\cos \left(\left(i + \frac{1}{2} \right) \frac{\left(k + \frac{1}{2} \right) \pi}{n} \right) \right]_{i,k=0,\dots,n-1}$$

$$S_{i,k}^{\text{IV}} = \sqrt{\frac{2}{n}} \left[\sin \left(\left(i - \frac{1}{2} \right) \frac{\left(k - \frac{1}{2} \right) \pi}{n} \right) \right]_{i,k=1,\dots,n}$$

$$c_i = \begin{cases} \frac{1}{\sqrt{2}} & \text{dla } i = 0 \text{ lub } n \\ 1 & \text{w p. p.} \end{cases}$$

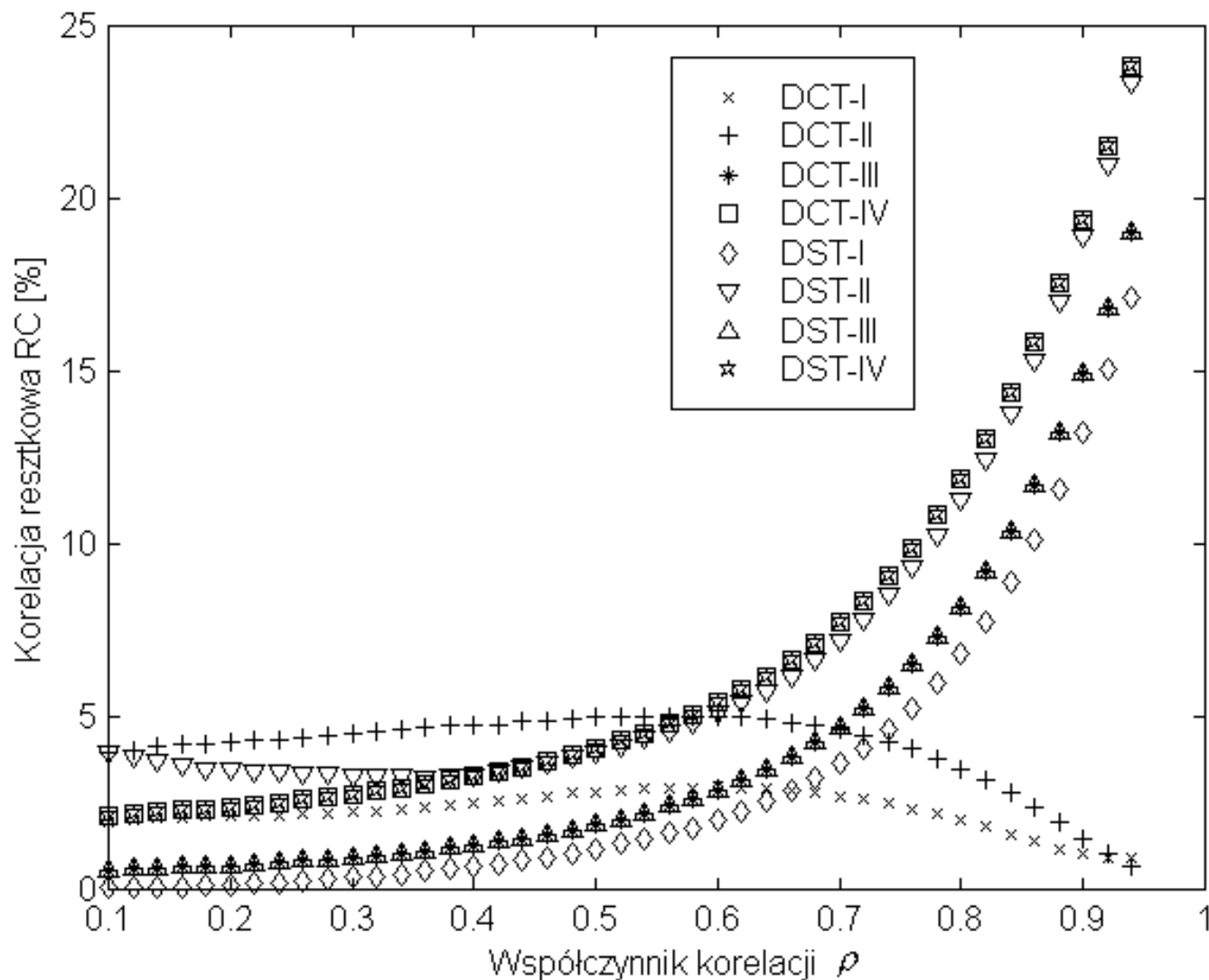
Porównanie: rozkład wariancji współczynników transformat

i	Transformata wektora $n=16$, model Markowa 1 rzędu, $\rho=0,9$										
	KLT	DFT	WHT	DCT				DST			
				I	II	III	IV	I	II	III	IV
1	9,9268	9,8346	9,8346	9,8678	9,8346	8,9324	8,6494	9,2183	8,9142	8,9324	8,6494
2	2,9487	1,8342	2,5360	2,9244	2,9327	3,3212	3,3195	2,6418	2,4576	3,3212	3,3195
3	1,1278	1,8342	1,0200	1,1416	1,2108	1,2934	1,2788	1,4678	1,4326	1,2934	1,2788
4	0,5682	0,5189	0,7060	0,5587	0,5815	0,6984	0,7256	0,7091	0,6895	0,6984	0,7256
5	0,3411	0,5189	0,3070	0,3361	0,3483	0,4333	0,4418	0,5312	0,5607	0,4333	0,4418
6	0,2294	0,2502	0,3030	0,2283	0,2314	0,2879	0,3166	0,3139	0,3320	0,2879	0,3166
7	0,1671	0,2502	0,2830	0,1708	0,1685	0,2161	0,2318	0,2628	0,3123	0,2161	0,2318
8	0,1290	0,1553	0,1060	0,1329	0,1295	0,1581	0,1862	0,1738	0,2060	0,1581	0,1862
9	0,1043	0,1553	0,1050	0,1126	0,1047	0,1319	0,1510	0,1526	0,2111	0,1319	0,1510
10	0,0875	0,1126	0,1050	0,0942	0,0877	0,1029	0,1302	0,1102	0,1493	0,1029	0,1302
11	0,0758	0,1126	0,1040	0,0867	0,0759	0,0920	0,1132	0,0988	0,1624	0,0920	0,1132
12	0,0675	0,0913	0,1040	0,0760	0,0675	0,0763	0,1028	0,0777	0,1211	0,0763	0,1028
13	0,0616	0,0913	0,1030	0,0743	0,0616	0,0716	0,0942	0,0707	0,1381	0,0716	0,0942
14	0,0574	0,0811	0,1020	0,0675	0,0574	0,0635	0,0892	0,0608	0,1075	0,0635	0,0892
15	0,0547	0,0811	0,0980	0,0692	0,0547	0,0618	0,0855	0,0569	0,1277	0,0618	0,0855
16	0,0531	0,0780	0,0780	0,0588	0,0531	0,0593	0,0840	0,0535	0,0780	0,0593	0,0840

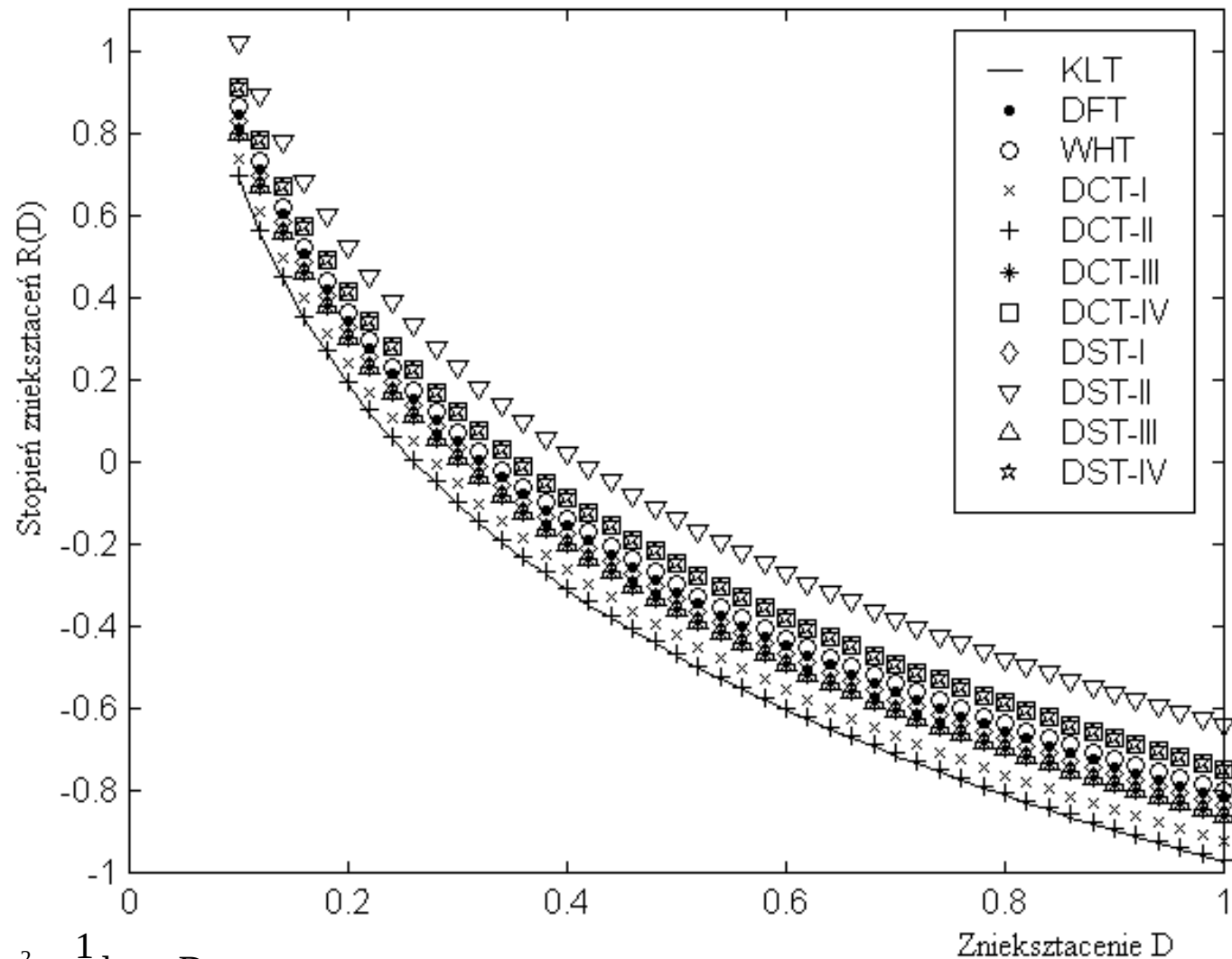
Porównanie: upakowanie energii

[illegible]

Porównanie: korelacja resztkowa



Porównanie: stopień zniekształceń



$$R(T, D) = \frac{1}{2n} \sum_{i=1}^n \log_2 \sigma_i^2 - \frac{1}{2} \log_2 D$$

Porównanie: maksymalna zdolność redukcji bitów

Transformacja	<i>MRB</i>
KLT	1,1231
DFT	0,9485
WHT	0,9374
DCT-I	1,0734
DCT-II	1,1172
DCT-III	1,0113
DCT-IV	0,8983
DST-I	0,9848
DST-II	0,7915
DST-III	1,0113
DST-IV	0,8983

$$MRB = -\frac{1}{2n} \sum_{i=1}^n \log_2 \sigma_i^2$$

DCT $\approx$ KLT

(na potrzeby kompresji)

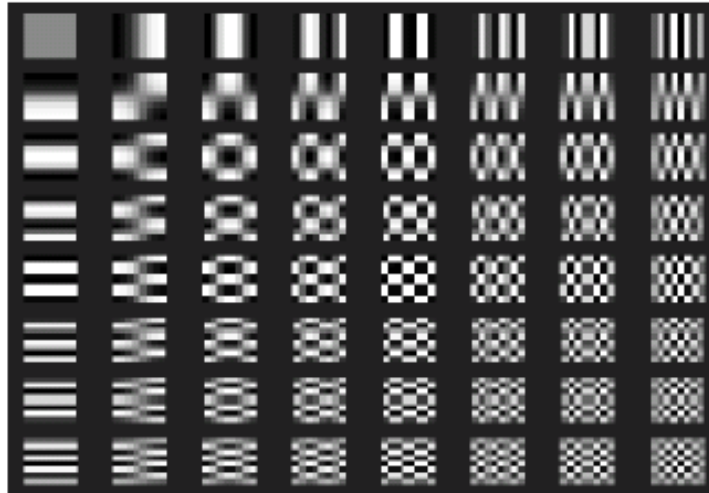
Transformacja DCT (2W) – efektywna, łatwa w implementacji, ‘lokalna’ poprzez bloki

prosta

$$k(u, v) = \frac{1}{\sqrt{2n}} c_u c_v \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} f(x, y) \cos \frac{(x + \frac{1}{2})u\pi}{n} \cos \frac{(y + \frac{1}{2})v\pi}{n}$$
$$c_u, c_v = \frac{1}{\sqrt{2}} \quad \text{dla } u, v = 0; \quad c_u, c_v = 1 \quad \text{w p.p.}$$

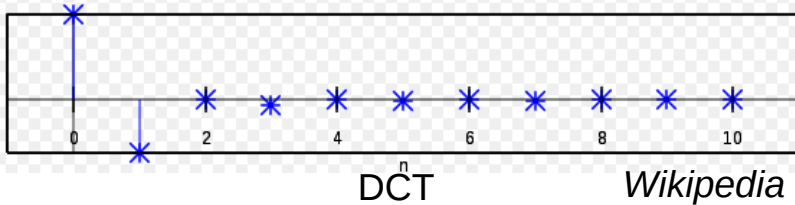
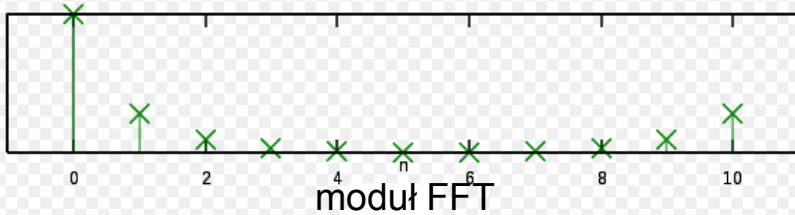
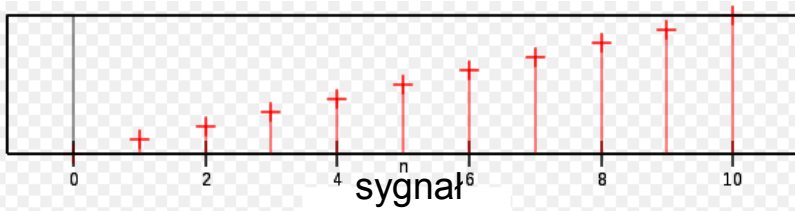
odwrotna

$$f(x, y) = \frac{1}{\sqrt{2n}} \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} c_u c_v k(u, v) \cos \frac{(x + \frac{1}{2})u\pi}{n} \cos \frac{(y + \frac{1}{2})v\pi}{n}$$

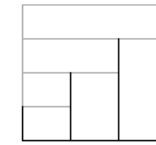


Stosowana m.in. w JPEG

DCT czy FFT

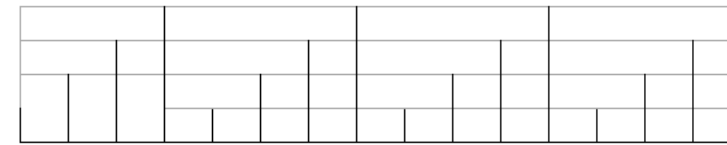


Wikipedia



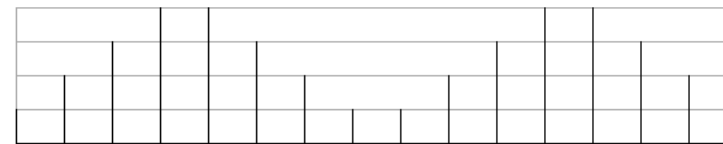
0 3

oryginał



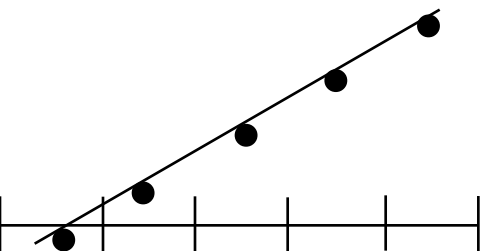
0 4 8 12

*rozwiniecie
sygnału w DFT*

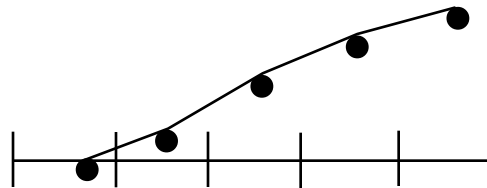


0 4 8 12

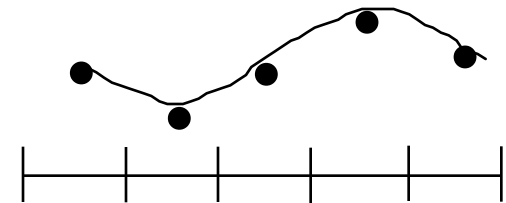
*rozwiniecie
sygnału w DCT*



oryginał



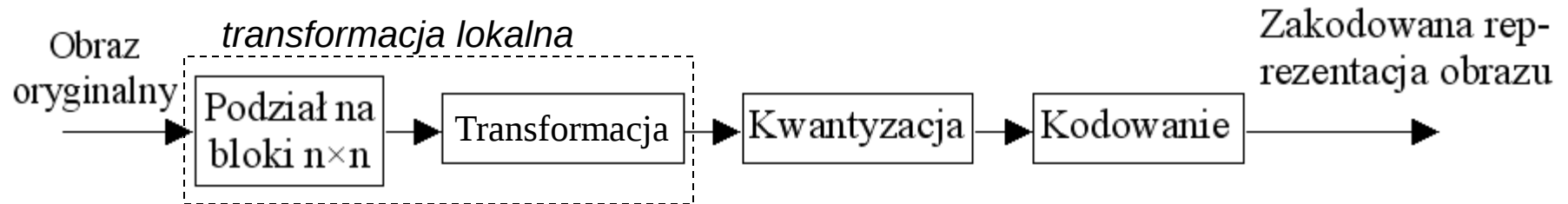
rekonstrukcja z kilku współczynników DCT



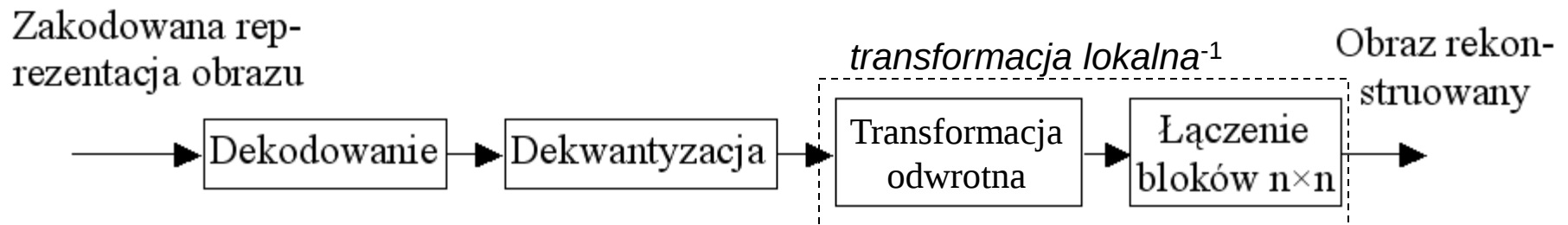
rekonstrukcja z kilku współczynników FFT

Paradygmat TK

KOMPRESJA



DEKOMPRESJA



Selekcja próbek

blok obrazu

109	101	91	79	80	85	84	89
87	87	89	99	98	91	91	93
100	102	109	127	112	99	87	84
84	89	87	88	83	90	100	94
100	101	108	112	110	114	113	109
103	103	95	89	89	89	82	86
84	97	100	100	108	103	106	112
111	115	111	113	107	100	109	107

784.2	6.630	-6.45	-5.46	2.25	0.000	-3.82	-2.80
-34.9	10.89	-0.46	5.318	4.209	3.641	3.270	-2.62
0.726	10.11	12.68	-0.72	2.956	-5.22	-0.91	3.229
-19.8	2.843	21.02	7.299	-5.43	-7.31	7.198	2.446
9.000	-2.50	20.10	1.802	-8.00	5.061	-0.48	1.331
-21.7	-3.61	8.453	-1.76	-5.13	-3.66	0.128	0.899
2.405	37.31	-0.16	3.307	2.564	0.964	2.072	-3.61
40.72	-3.57	-18.2	-1.70	3.892	0.977	2.199	-0.53

DCT

*strefowa
selekcja
próbek*

784	7	-6	-5	2	0	0	0
-35	11	0	5	0	0	0	0
1	10	13	0	0	0	0	0
-20	3	0	0	0	0	0	0
9	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

784	0	0	0	0	0	0	0
-35	11	0	0	0	0	0	0
0	10	13	0	0	0	0	0
-20	0	21	0	0	0	0	0
0	0	20	0	0	0	0	0
-22	0	0	0	0	0	0	0
0	37	0	0	0	0	0	0
41	0	-18	0	0	0	0	0

*progowa
selekcja
próbek
(próg=10)*

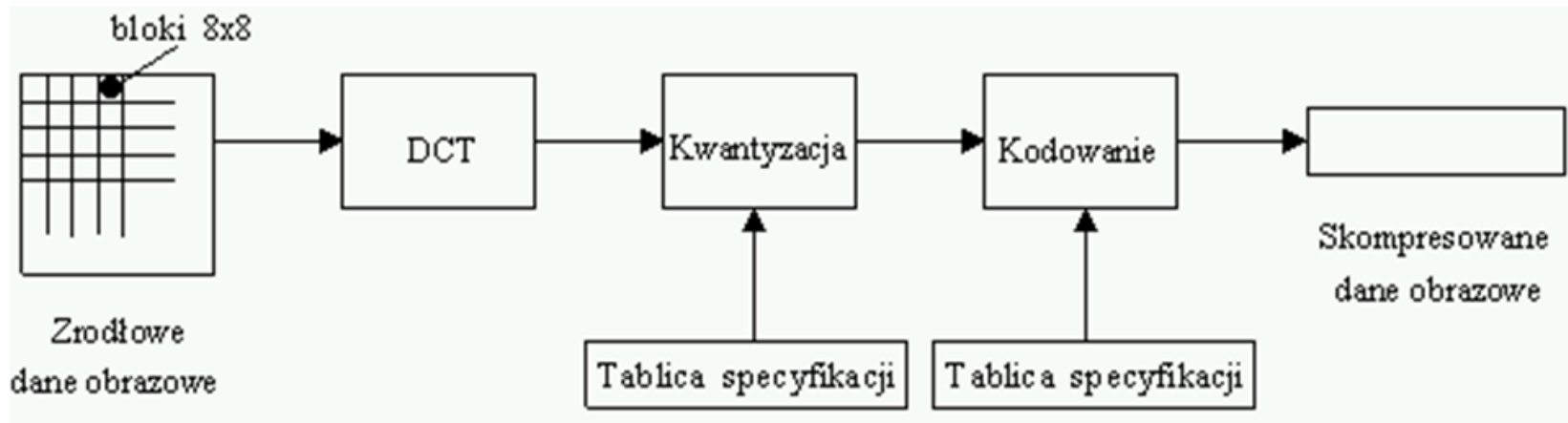
*rekonstrukcje
bloku*

99	96	93	90	88	86	85	85
97	96	94	93	92	90	89	89
96	97	98	100	99	97	95	94
95	97	101	104	104	101	99	98
93	96	100	103	102	99	97	97
93	96	99	100	98	96	96	97
100	102	103	103	100	99	101	103
109	109	109	108	106	105	108	112

106	99	89	81	77	80	85	89
87	89	91	93	95	97	97	98
100	106	114	118	114	103	90	82
85	85	85	86	88	91	94	96
100	102	104	107	110	112	113	113
102	99	94	90	88	88	89	91
90	93	99	104	108	110	109	109
114	112	108	106	104	105	106	107

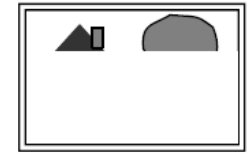
Standard JPEG

- Podstawowy proces kodowania (ang. baseline process).
- Rozszerzony , bazujący na DCT proces kodowania (ang. extended DCT- based process).
- Bezstratny proces kodowania (ang. lossless process).
- Hierarchiczny proces kodowania (ang. hierarchical process).



Właściwości kodeka JPEG

■ Progresja kodowania



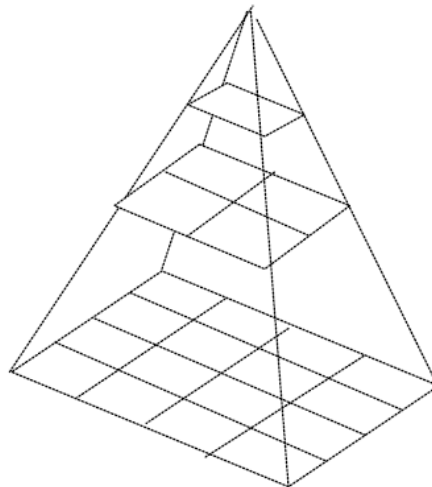
sekwencyjnie



z progresją



■ Kodowanie hierarchiczne (z podpróbkowaniem)



Kwantyzacja i kodowanie w JPEG

Diagram illustrating the quantization process. A blue arrow labeled u points to the first row, and a blue arrow labeled v points to the first column. The text "składowa stała" (component constant) is written next to the first row.

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

Percepcyjna tablica kwantyzacji (luminancja)

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

tablica
kwantyzacji dla
chrominancji

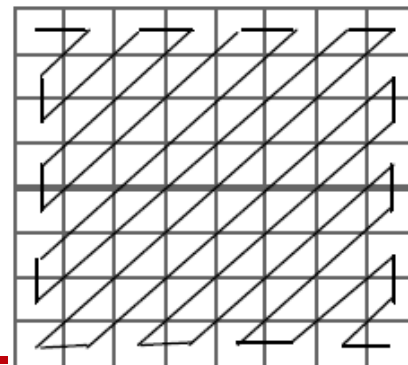
Kwantyzacja/selekcja:

dzielenie współczynników DCT $k(u,v)$ przez element tablicy kwantyzacji q o współrzędnych (u,v) i zaokrąglanie do liczby całkowitej

$$\tilde{k}(u,v) = \left\lfloor \frac{k(u,v)}{q(u,v)} \right\rfloor$$

Kodowanie:

przeglądanie zygzak, różnicowo składowa stała bloków (predykcja), kod Huffmana-RLE lub kod arytmetyczny



Efekty JPEGowe



Oryginał



12:1



43:1



barb2.bmp 263222 Bytes



Efekty JPEGowe

10% 26584 Bytes



5% 14695 Bytes



1% 3482 Bytes



Polecam zabawę z JPEG

<http://www.sfu.ca/~cjenning/toybox/hjpeg/index.html>

Transform of each channel. Notice that you can make out the patterns of little 8x8 boxes. Remember that the upper-left corner of those boxes is the low-frequency part, the other corner is high-frequency. Brighter pixels mean more information, *i.e.* more of the image was of that kind of frequency.

- Now choose the next drop down item (Quantized DCTs). Whoosh! The lower corners of each of the 8x8 boxes just went dark. The values in each box were just divided by the values in the corresponding cells (throwing away any remainders) of the quantization tables (Step 3), so they now have less information. The next drop down box shows the reconstructed DCTs, *i.e.* what we get back when we decompress the image by multiplying by the values in the tables. Compare the original DCT of the colour planes to the reconstructed versions. A *lot* of colour information has been lost, but it is not very noticeable in the final decompressed picture.

JPEG and Hierarchical JPEG Demo

1. Choose a sample image:
Lena (128 × 128)

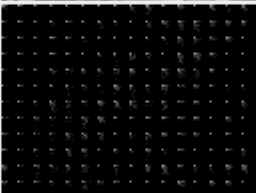
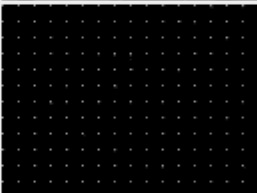
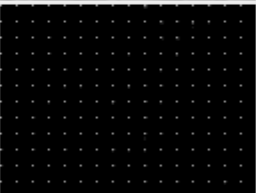
2. Choose a chroma subsampling format:
☒ None (4:4:4) ☐ Quartered (4:2:0)

3. Choose a quality setting or...
Low High

...create custom quantization tables:

Luminance								Chrominance							
28	21	24	24	32	42	87	129	28	21	24	24	32	42	87	129
19	21	23	30	39	62	114	165	19	21	23	30	39	62	114	165
17	24	28	39	66	98	140	170	17	24	28	39	66	98	140	170
28	33	42	51	100	114	156	176	28	33	42	51	100	114	156	176
42	46	71	91	122	145	185	201	42	46	71	91	122	145	185	201
71	104	102	156	195	186	217	179	71	104	102	156	195	186	217	179
91	107	123	143	185	203	215	185	91	107	123	143	185	203	215	185
109	98	100	111	138	165	181	177	109	98	100	111	138	165	181	177

Done

RGB / RGB-Output	Y / Y-Quant.	Cb / Cb-Quant.	Cr / Cr-Quant.
			
			

The first row of monitors shows the input image. The second row shows: Quantized DCTs

Zoom Level

x	y	
126	75	
R	G	B
232	191	136
Y	Cb	Cr
0	0	0

40	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
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

: applet requires that your browser support at least the Java 2 runtime. If the applet above doesn't work properly, this is almost certainly the blem. You can download the latest Java runtime from Sun to correct this.