

# Komprese obrazů

Václav Hlaváč

České vysoké učení technické v Praze  
Fakulta elektrotechnická, katedra kybernetiky  
Centrum strojového vnímání

<http://cmp.felk.cvut.cz/~hlavac>, [hlavac@fel.cvut.cz](mailto:hlavac@fel.cvut.cz)

*Poděkování: Tomáš Svoboda, Jan Kybic*

## Osnova přednášky:

- ◆ Komprese 1D dat vs. obrazů.
- ◆ Redundance a irrelevance.
- ◆ Schéma komprese obrazu.
- ◆ Optimální kódování.
- ◆ Entropie a komprese.
- ◆ Prediktivní a transformační komprese.

# Kompresa obrazů, úvod

- ◆ **Cíl:** Zmenšení množství dat potřebných k reprezentaci obrazu. Spotřebované množství paměti se měří například v bitech.
  - ◆ **Použití:** Pro přenos a uchování dat.
  - ◆ **Proč se liší komprese 2D obrazů od komprese 1D dat?**
- 
- ◆ Digitalizovaný obraz se chápe jako dvojrozměrná struktura (matice) náhodných vzorků.
  - ◆ Matematicky řečeno: Cílem je převést matici jasů (nebo 3 matice s barevnými složkami) do jiné reprezentace, v níž jsou obrazová data méně statisticky závislá (zhruba řečeno, méně korelovaná).

## Proč je možná komprese obrazů?

- ◆ Obrazy nejsou náhodné. Nejsou většinou pouze náhodným šumem.
- ◆ V obrazech je nadbytečnost (redundance). Hodnotu obrazové funkce lze z hodnot v okolí předpovídat.
- ◆ Hodnoty obrazové funkce nejsou rozděleny rovnoměrně.
- ◆ Mezi hodnotami obrazové jsou plošné závislosti (korelace), velmi často se projevují lokálně, někdy i globálně (např. u některých textur).
- ◆ U barevných obrazů jsou mezi jejich jednotlivými barevnými kanály statistické závislosti.
- ◆ Ne všechny detaily jsou v obrazech potřebné, když je bude pozorovat člověk. Vizualní subsystém člověka mnohé detaily není schopen vnímat.

## Možné čtení

- ◆ Anil Jain: “Fundamentals of Digital Image Processing”, 1989.
- ◆ M. Sonka, V. Hlaváč, R. Boyle R.: “Image Processing, Analysis, and Machine Vision”, 2007.
- ◆ T. Svoboda, J. Kybic, V. Hlaváč: “Image Processing, Analysis, and Machine Vision, A MATLAB Companion”, 2007.  
<http://visionbook.felk.cvut.cz>

# Snížení rozlišení

- ◆ Snížení prostorového rozlišení obrazu.
- ◆ Jednoduchá, ztrátová metoda komprese, která se obvykle hodí pro web nebo zobrazování na monitoru s omezeným prostorovým rozlišením.
- ◆ Lze vylepšit pokročilejší interpolací, např. pomocí B-splajnů.

## Snížení rozlišení, příklad (1)



Original size, 3456 × 5184, 859 kB (stored as JPEG with quality 75).

## Snížení rozlišení, příklad (2)



Downsampled  $2\times$ ,  $1728 \times 2592$ , 237 kB.

## Snížení rozlišení, příklad (3)



Downsampled 4×, 864 × 1296, 75 kB.



## Snížení rozlišení, příklad (4)



Downsampled 8×, 432 × 648, 27 kB.

## Snížení rozlišení, příklad (5)



Downsampled  $8\times$ ,  $216 \times 324$ , 10 kB.

## Snížení rozlišení, příklad (6)



Downsampled  $8\times$ ,  $216 \times 324$ , 10 kB, bicubic interpolation.

## Snížení rozlišení, příklad (7)



Downsampled  $16\times$ ,  $108 \times 162$ , 4.2 kB.

## Snížení rozlišení, příklad (8)



Downsampled  $16\times$ ,  $108 \times 162$ , 4.2 kB, bicubic interpolation.

# Redundance a irelevance

## ◆ Redundance v kódování

- Základní princip: častěji se opakující symbol se kóduje kratším kódovým slovem.
- Optimální kódování: Huffmanovo a aritmetické kódování.

## ◆ Redundance mezi pixely

- Pomocí lineárních integrálních transformací obrazu, např. Fourierovou, kosinovou či vlnkovou transformací.
- Prediktivní komprese.
- Úsporné algoritmy generování obrazu, např. fraktální.

## ◆ Irelevance z hlediska vnímání člověkem

- Nezobrazit např. všechny jasové úrovně nebo vysoké frekvence.

# Rozdělení metod komprese obrazů

## 1. Segmentace objektů v obraze, tj. s interpretací dat.

- ◆ Metody jsou závislé na sémantice dat.
- ◆ Dosahuje se nejvyšších kompresních poměrů.
- ◆ Není možná zpětná rekonstrukce výchozího obrazu.

## 2. Odstranění redundantní a irelevantní informace, tj. bez interpretace dat.

- ◆ Lze použít na libovolná obrazová data.
- ◆ Využívá se statistických nadbytečností v obraze (sekvenci obrazů).

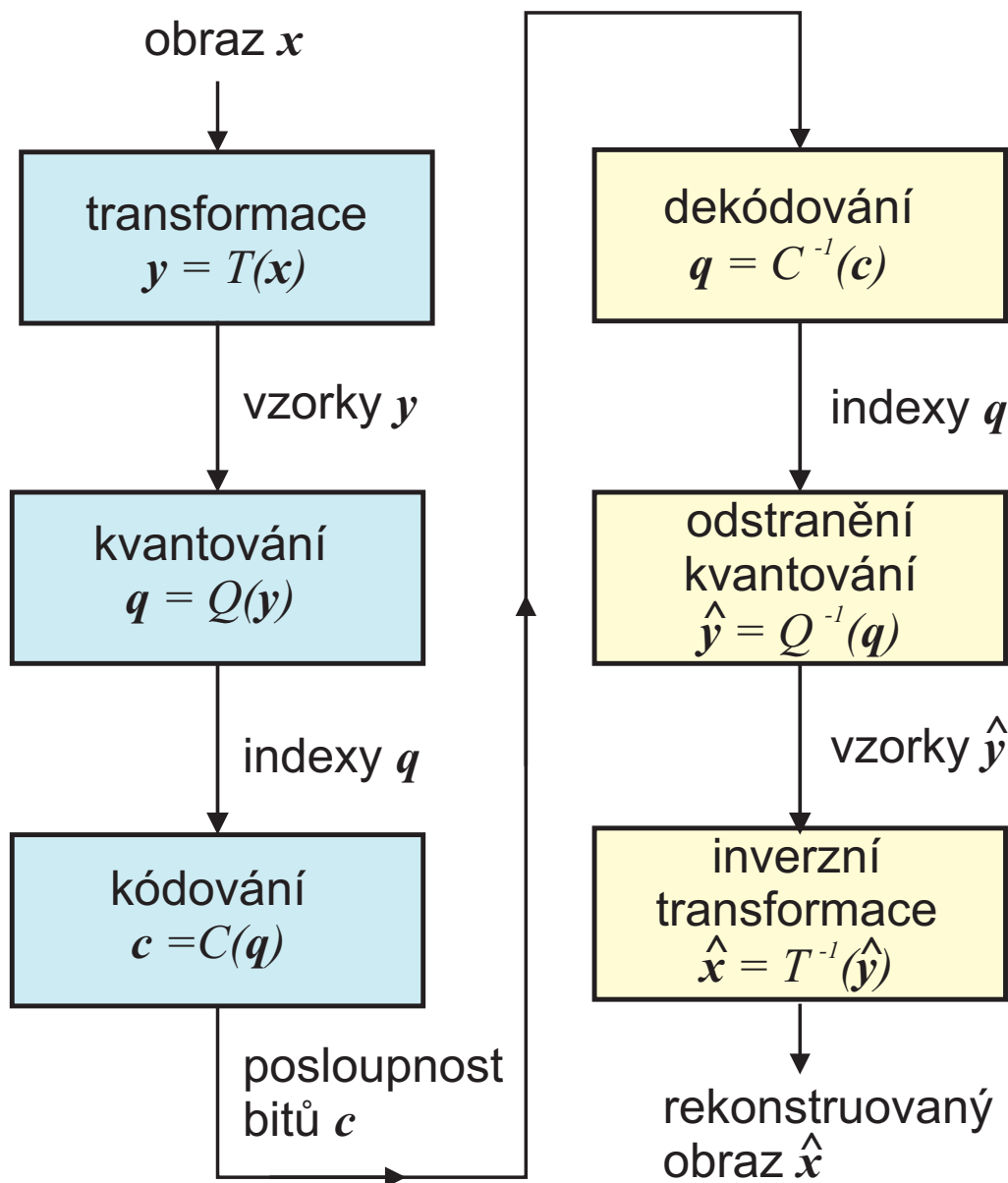
# Redundance versus irrelevance

Dvě velké třídy používaných postupů:

1. **Bezeztrátové metody**, odstraňují redundanci.
  - ◆ Odstraňují pouze statistickou nadbytečnost.
  - ◆ Umožňují úplnou rekonstrukci výchozího signálu.
2. **Ztrátové metody**, odstraňují irelevantní informaci.
  - ◆ Nevratně odstraňují část informace, která má v daném kontextu malý význam (např. vysoké frekvence, jasové detaily nepostřehnutelné lidským okem).
  - ◆ Umožňují pouze částečnou rekonstrukci výchozího signálu.



# Kompresa obrazu a jeho rekonstrukce



- ◆ Transformace  $T$  odstraňuje redundanci a je obvykle invertovatelná.  
*Např.: kosínová transformace, kódování úseky řádků (RLE).*
- ◆ Kvantování  $Q$  odstraňuje irelevanci a není invertovatelné.  
*Např.: Zanedbání koeficientů kosínové transformace odpovídajících vysokým frekvencím.*
- ◆ Kódování  $C$  a dekódování  $C^{-1}$  je invertovatelné a bezztrátové.

## Huffmanovo kódování, z roku 1952

- ◆ Vstup: symboly s pravděpodobností jejich výskytu; zpráva.
- ◆ Výstup: optimálně zakódovaná zpráva.
- ◆ Postup: podle pravděpodobností výskytu symbolů se zdola nahoru se vytváří binární (Huffmanův) strom. Tento strom potom slouží ke generování zakódované zprávy.
- ◆ Prefixový kód, tj. žádné kódové slovo nemůže být prefixem žádného jiného kódového slova. Umožňuje dekodování, aniž by se znala délka jednotlivých slov.
- ◆ Celočíselný počet bitů na symbol.
- ◆ Nechť  $b$  je průměrný počet bitů na symbol. Potom průměrná délka kódového slova je  $L$ ,

$$H(b) \leq L \leq H(b) + 1$$

# Huffmanovo kódování, příklad

Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02

$$s_2 = 2 \quad \underline{0.3}$$

$$s_1 = 1 \quad \underline{0.26}$$

$$s_3 = 3 \quad \underline{0.15}$$

$$s_0 = 0 \quad \underline{0.12}$$

$$s_4 = 4 \quad \underline{0.1}$$

$$s_5 = 5 \quad \underline{0.03}$$

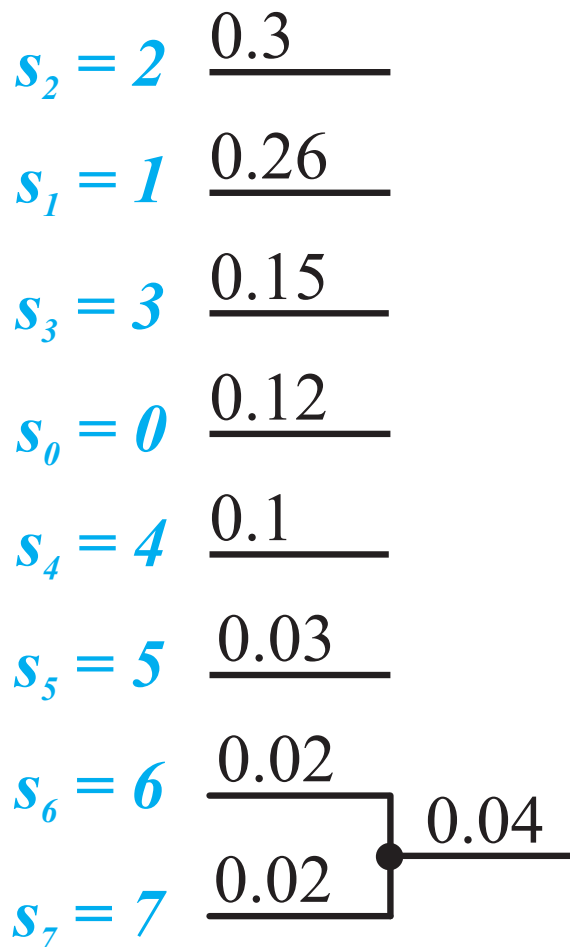
$$s_6 = 6 \quad \underline{0.02}$$

$$s_7 = 7 \quad \underline{0.02}$$

# Huffmanovo kódování, příklad

Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

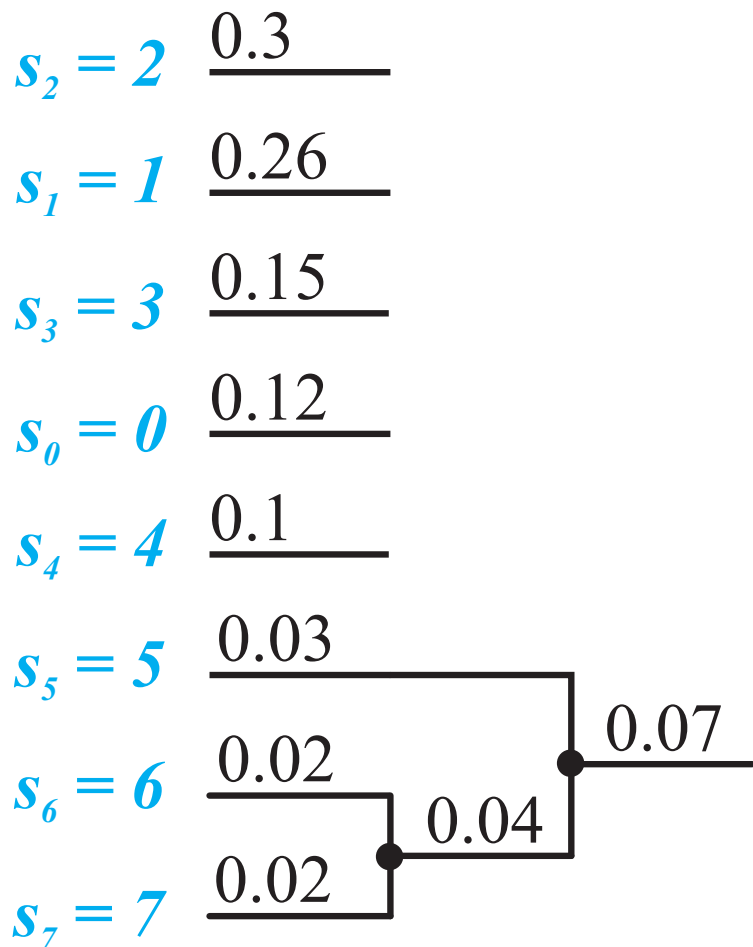
$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02



# Huffmanovo kódování, příklad

Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

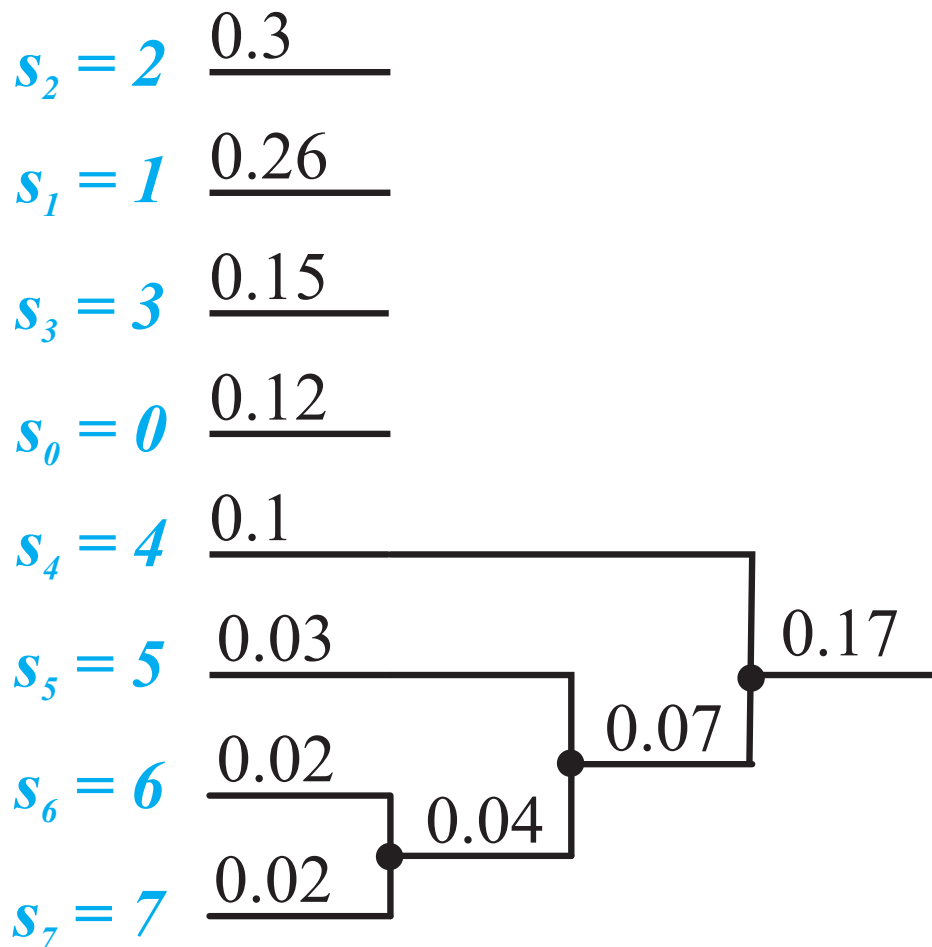
$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02



# Huffmanovo kódování, příklad

Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

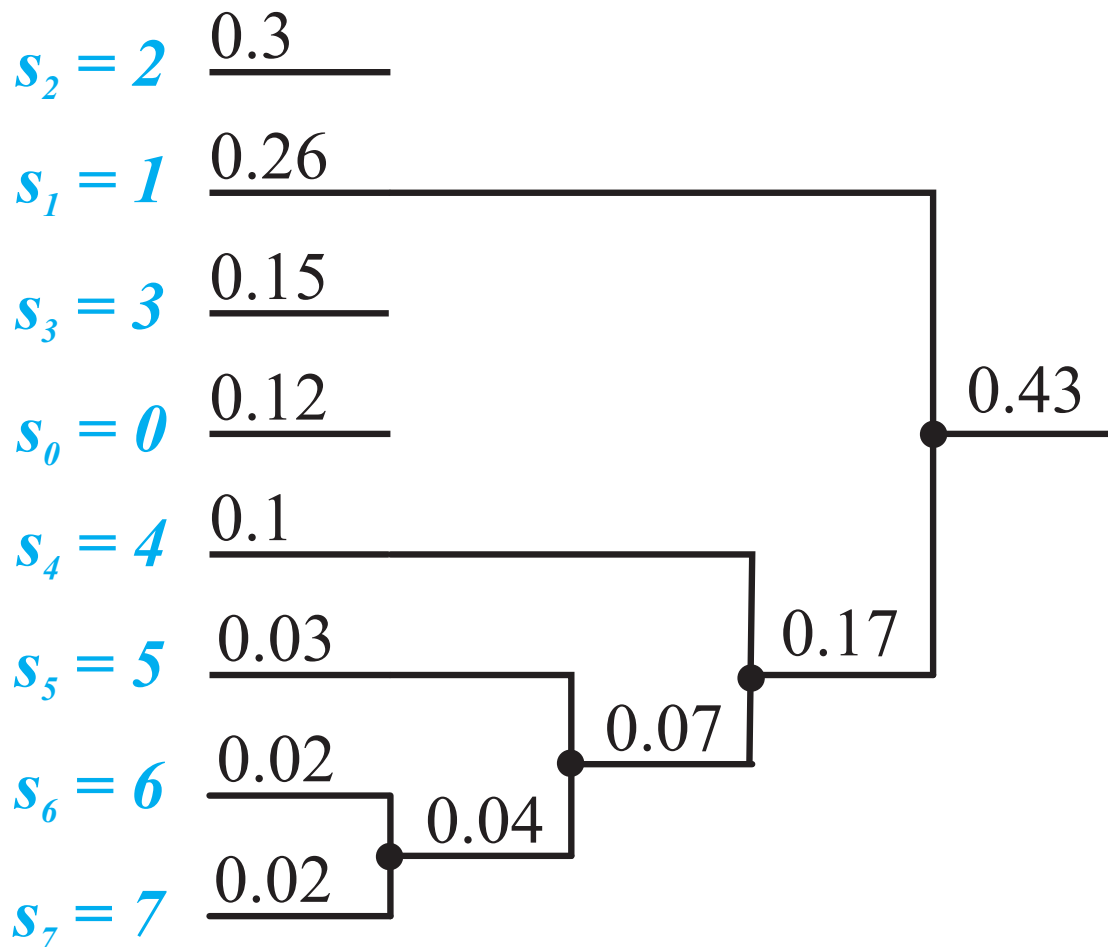
$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02



# Huffmanovo kódování, příklad

Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

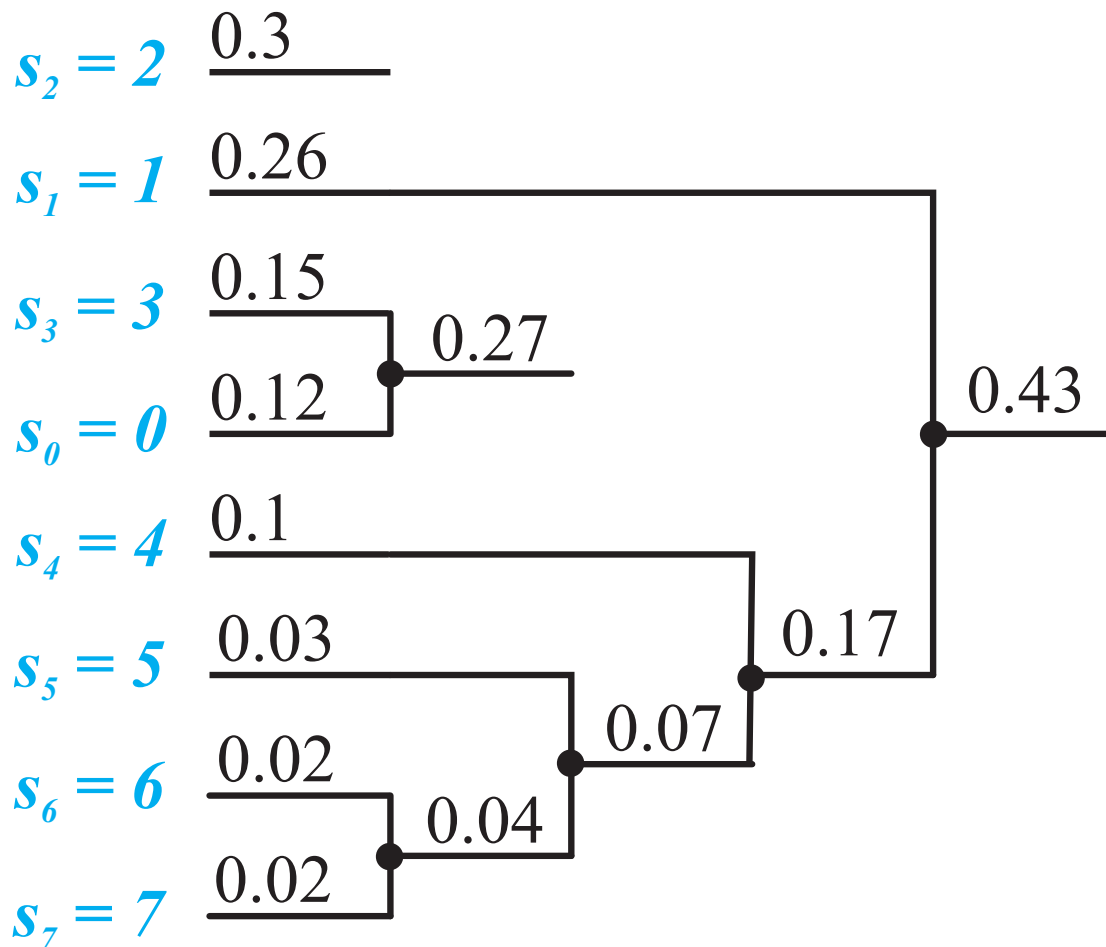
$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02



# Huffmanovo kódování, příklad

Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02

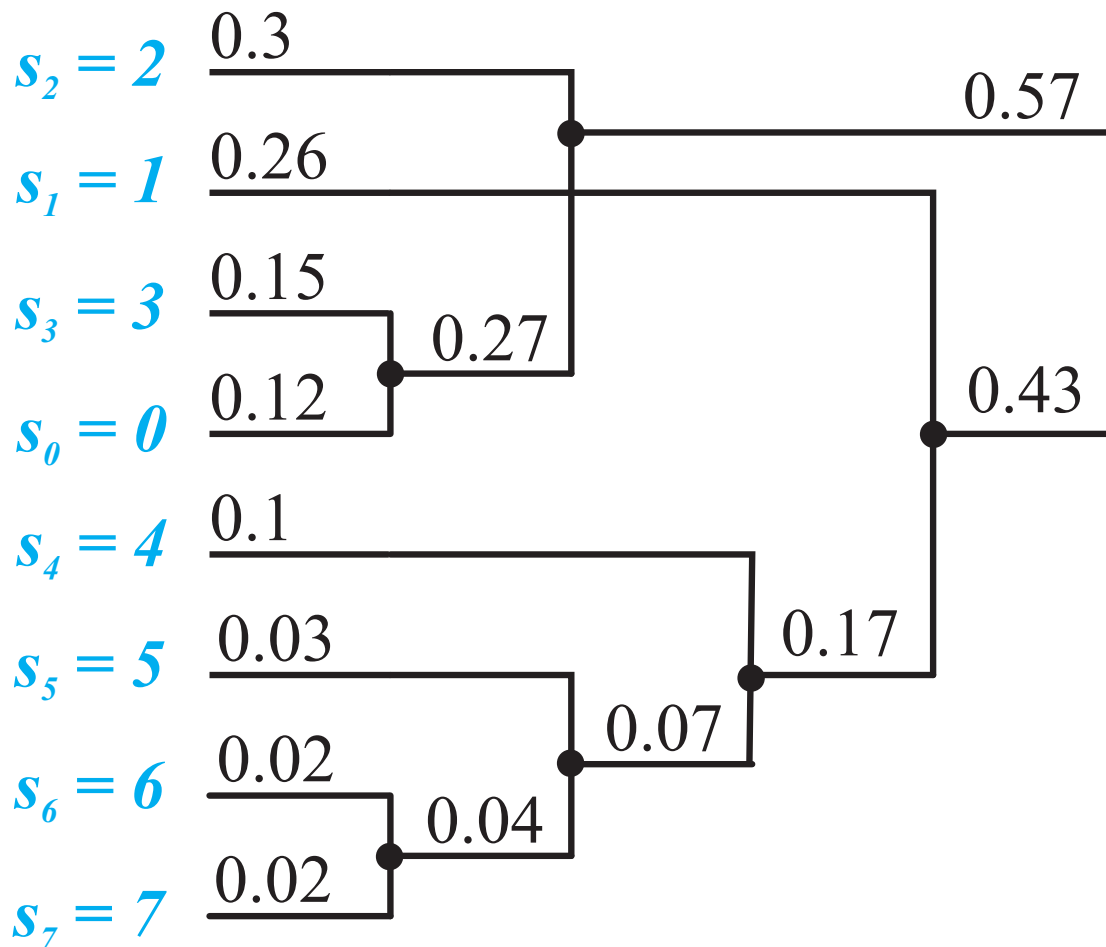




# Huffmanovo kódování, příklad

Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

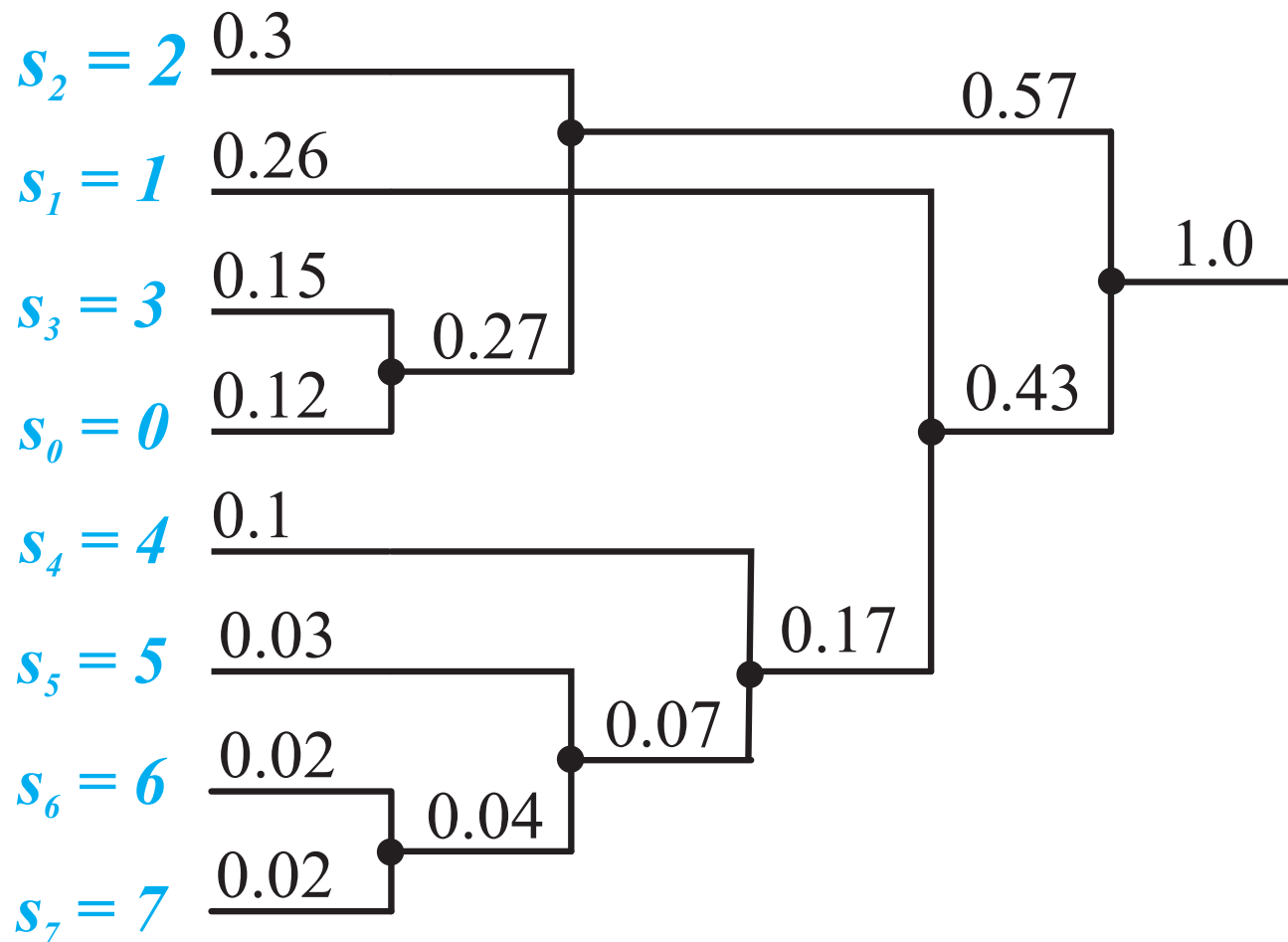
$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02



# Huffmanovo kódování, příklad

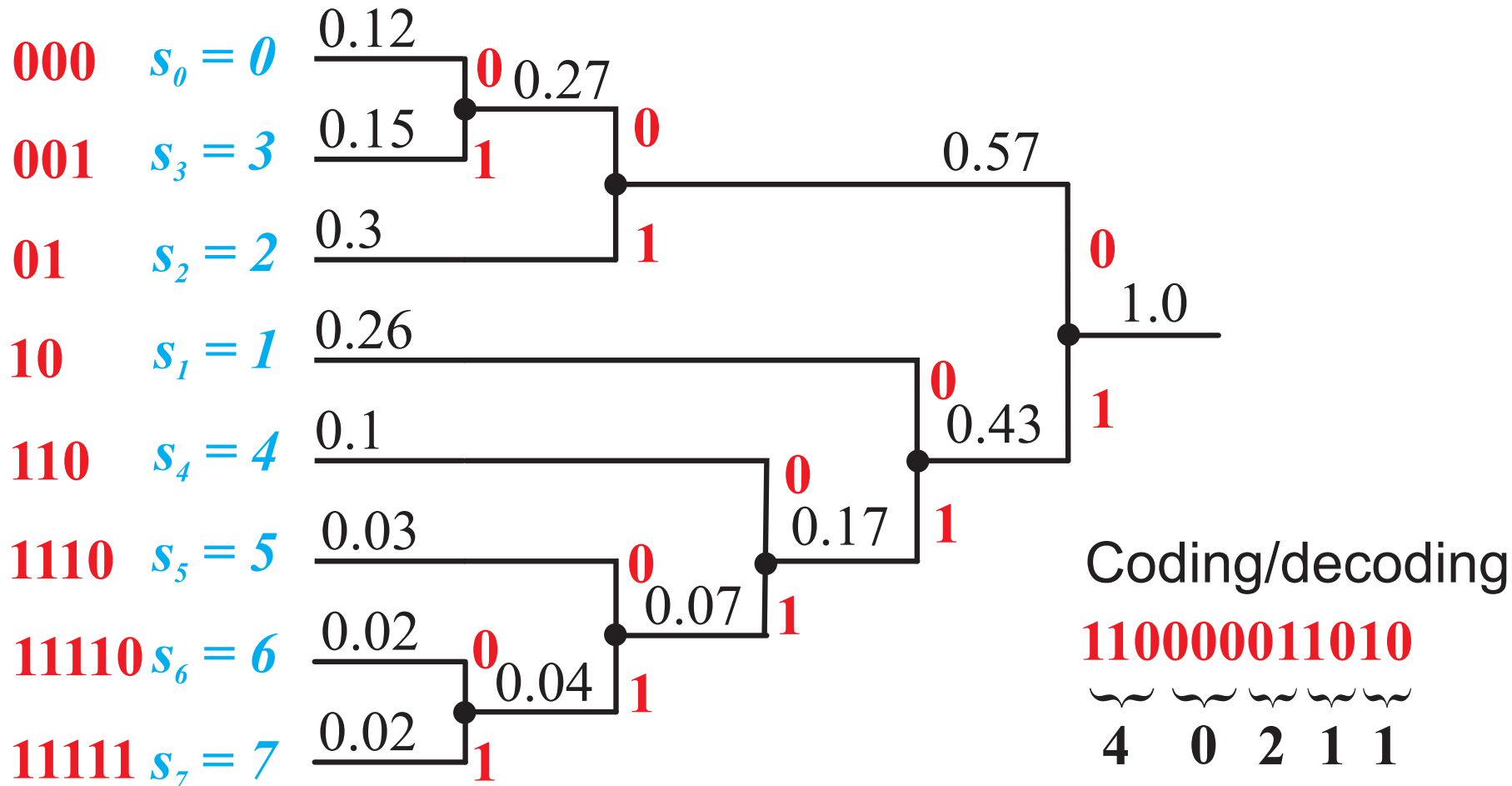
Vstup: zprávy obsahují osm symbolů s pravděpodobnostmi výskytu

$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$
0.12	0.26	0.3	0.15	0.1	0.03	0.02	0.02



# Huffmanovo kódování, příklad přeuspořádání stromu

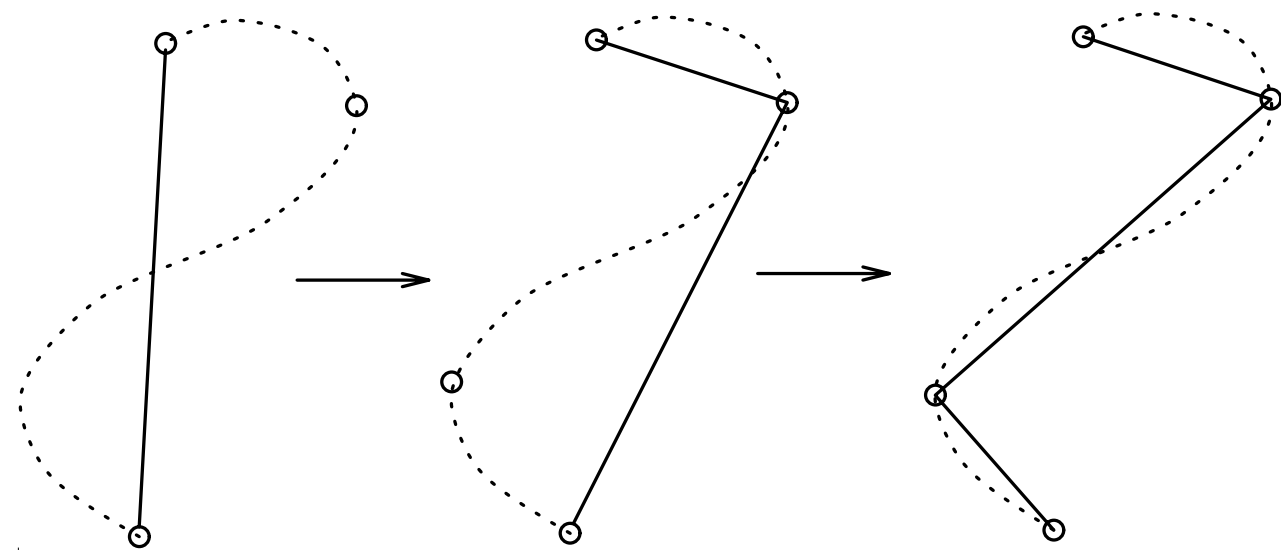
- ◆ Důvod: aby se větve stromu nekřížily.
- ◆ Kódování: u větvení stromu 0, 1.



# Kódování segmentovaných dat (1)

## Kódování hranic oblastí

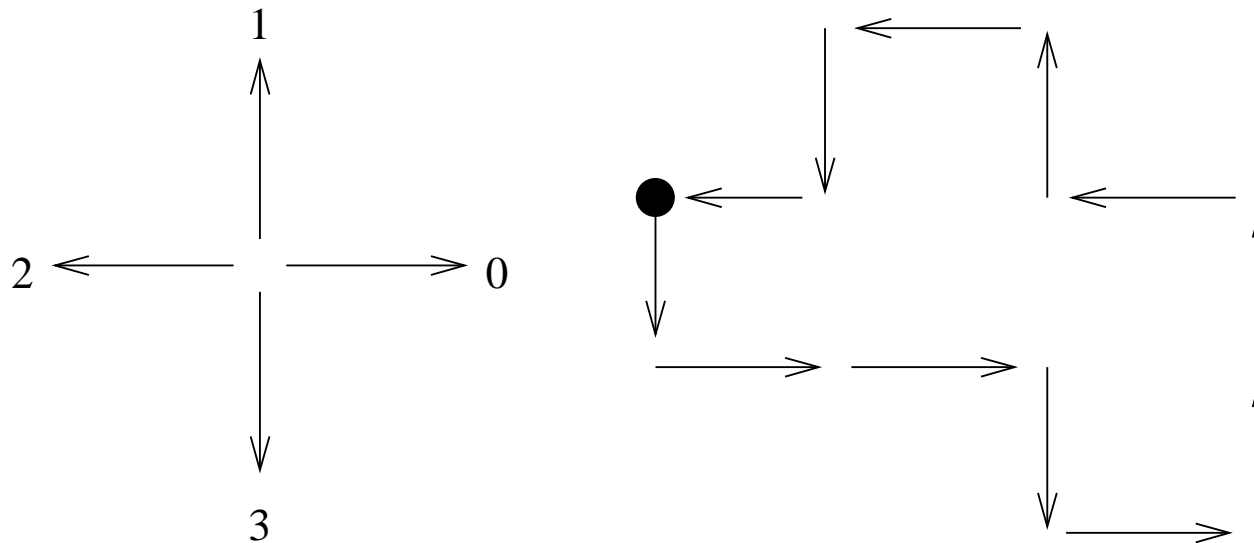
Polygonální aproximace hranice



# Kódování segmentovaných dat (2)

## Kódování hranic oblastí

Řetězový (též Freemanův, 1961) kód, 4-okolí



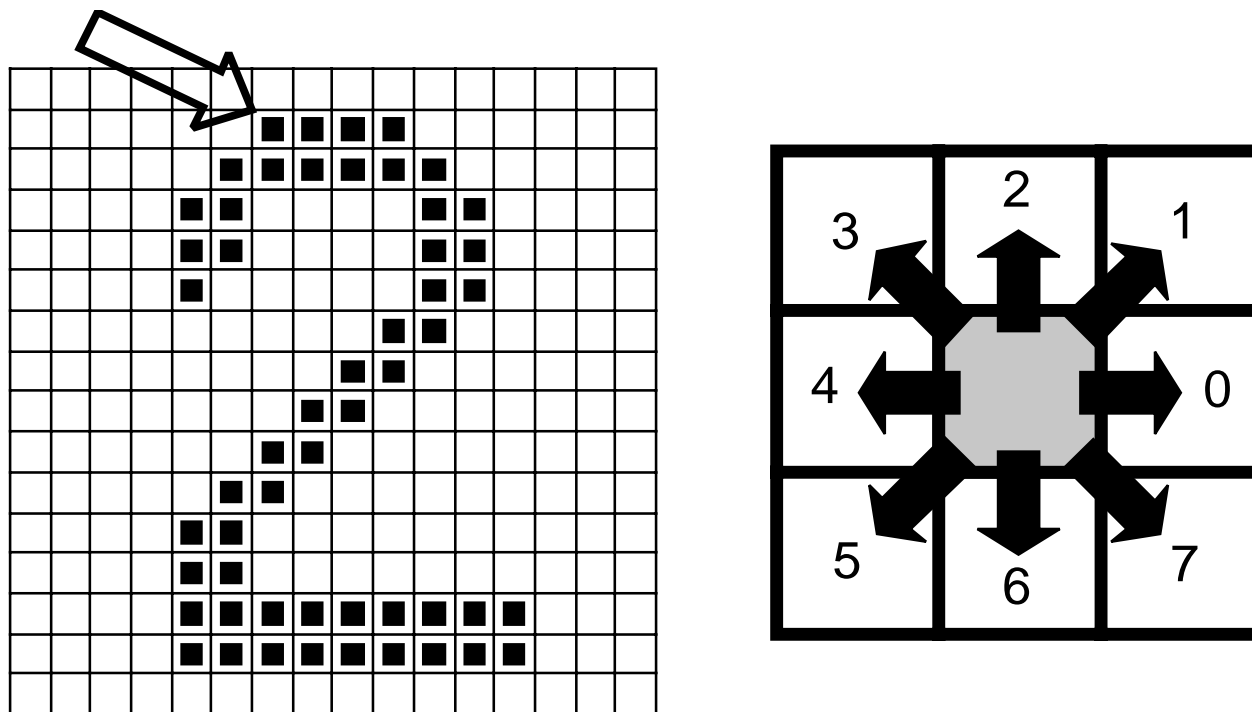
Řetězový kód: 3, 0, 0, 3, 0, 1, 1, 2, 1, 2, 3, 2.

Derivace kódu: 1, 0, 3, 1, 1, 0, 1, 3, 1, 1, 3, 1.

# Kódování segmentovaných dat (3)

## Kódování hranic oblastí

Řetězový (též Freemanův) kód, 8-okolí



Kód: 000776655555660000000644444444222111112234445652211

# Kódování segmentovaných dat (4)

## Kódování oblastí

Kódování úseky řádků (angl. Run Length Encoding, RLE)

	0	1	2	3	4	5	6
0							
1		■			■		
2		■	■	■	■		
3							
4							
5			■	■		■	
6							

Kódem je seznam seznamů. Každý seznam popisuje situaci v jednom řádku. Používá FAX (CCITT Group 3).

((11144)(214)(52355))

# Teorie informace a redundance

**Entropie ve fyzice** je měrou energie soustavy, která není k dispozici k vykonání práce. Jelikož práci lze získat “z řádu soustavy”, je entropie měrou neuspořádanosti soustavy. Souvisí s druhou termodynamickou větou.

Pojem zavedl v roce 1850 německý fyzik Rudolf Clausius (1822-1888, jeden ze zakladatelů termodynamiky).

---

**Entropie v teorii informace**, Claude Shannon, 1948

$$H_e = - \sum_i p_i \log_2 p_i \quad [\text{bitů}],$$

kde  $p_i$  je pravděpodobnost  $i$ -tého symbolu ve zprávě.



## Entropie, dva příklady

Nechť jsou ve zprávě jen dva znaky a, b.

### Příklad 1

$$p(\mathbf{a}) = p(\mathbf{b}) = \frac{1}{2}$$

$$H = - \left( \frac{1}{2} \log_2 \frac{1}{2} + \frac{1}{2} \log_2 \frac{1}{2} \right) = \left( \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 1 \right) = 1$$

### Příklad 2

$$p(\mathbf{a}) = 0,99; p(\mathbf{b}) = 0,01$$

$$\begin{aligned} H &= - (0,99 \log_2 0,99 + 0,01 \log_2 0,01) \\ &= - (0,99 \cdot (-0,0145) + 0,01 \cdot (-6,6439)) \\ &= 0,0144 + 0,0664 = 0,0808 \end{aligned}$$

# Entropie pro šedotónový obraz

Nechť obraz má  $G$  jasových úrovní,  $k = 0 \dots G - 1$  s pravděpodobnostmi  $P(k)$ .

Entropie  $H_e = - \sum_k P(k) \log_2 P(k)$  [bitů] ,

Nechť  $b$  je nejmenší počet bitů, kterým lze reprezentovat počet kvantizačních úrovní.

Informační redundance  $r = b - H_e$  .

# Odhad entropie z histogramu obrazu

Nechť  $h(k)$ ,  $0 \leq k \leq 2^b - 1$  a  $M$ ,  $N$  jsou rozměry obrazu.

$$\text{Odhad pravděpodobnosti } \hat{P} = \frac{h(k)}{M N}.$$

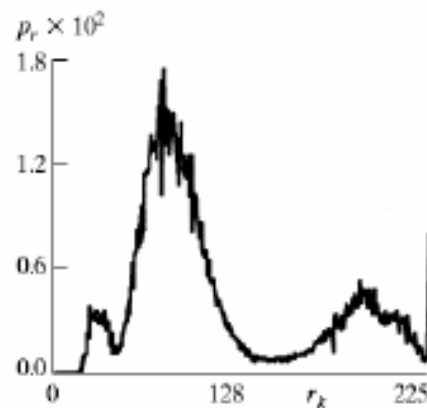
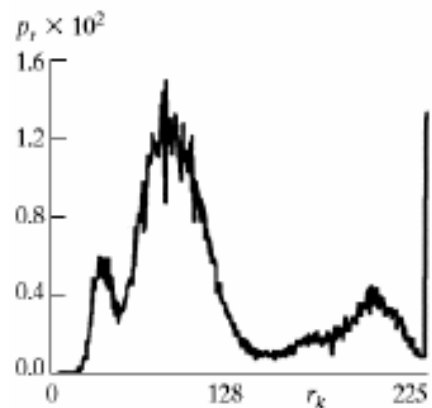
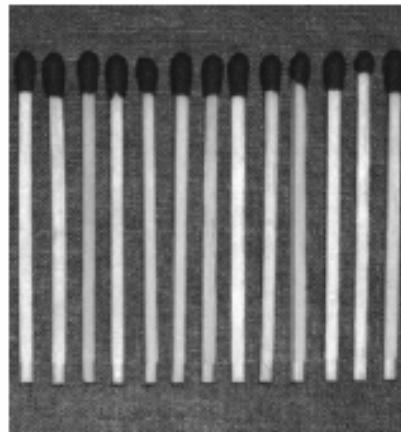
$$\text{Odhad entropie } \hat{H}_e = - \sum_k^{2^b-1} \hat{P}(k) \log_2 \hat{P}(k) \quad [\text{bitů}]$$

---

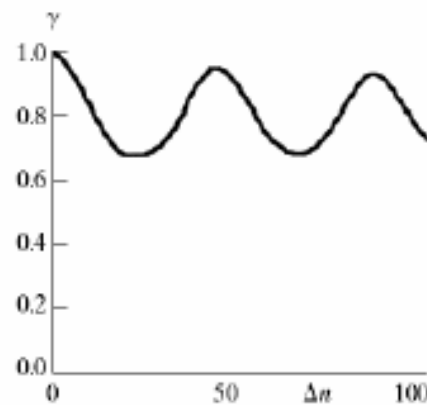
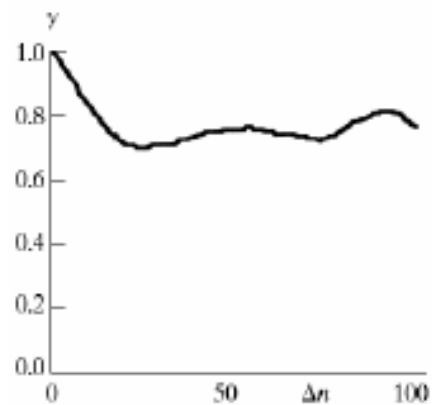
Poznámka:

Odhad entropie je příliš optimistický. Ve skutečnosti je entropie nižší, protože mezi jasy jednotlivých pixelů obrazu existují statistické závislosti.

# Ilustrace redundance mezi pixely



Histogram



Normalizovaná autokorelace

# Tři definice kompresního poměru

1. Na základě **redundance** (měřené entropií)  $K = \frac{b}{\hat{H}_e}$

2. Na základě **úspory paměti**

$$\kappa = \frac{n_1}{n_2} = \frac{\text{délka zprávy před kompresí}}{\text{délka zprávy po kompresi}}$$

3. **Relativní úspora paměti**  $R = 1 - \frac{1}{\kappa}$

---

Příklad 1:  $n_1 = n_2 \Rightarrow \kappa = 1, R = 0$ .

Příklad 2:  $n_1 : n_2 = 10 : 1 \Rightarrow \kappa = 10, R = 0,9 = 90\%$ .

# Měření ztrátovosti komprese

Nechť  $u_1, \dots, u_n$  je výchozí posloupnost a  $u'_1, \dots, u'_n$  ztrátově komprimovaná posloupnost

- ◆ Středně kvadratická chyba (Mean Square Error, MSE)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (u_i - u'_i)^2$$

- ◆ Poměr signálu k šumu (Signal to noise ration, SNR)

$$\text{SNR} = 10 \log_{10} \frac{P^2}{\text{MSE}^2} [\text{dB}],$$

kde  $P$  je interval hodnot vstupní posloupnosti,

$$P = \max\{u_1, \dots, u_n\} - \min\{u_1, \dots, u_n\}.$$

## Měření ztrátovosti komprese (2)

- ◆ Špičkový poměr signálu k šumu (peak-signal to noise ration, PSNR)

$$\text{PSNR} = 10 \log_{10} \frac{M^2}{\text{MSE}^2},$$

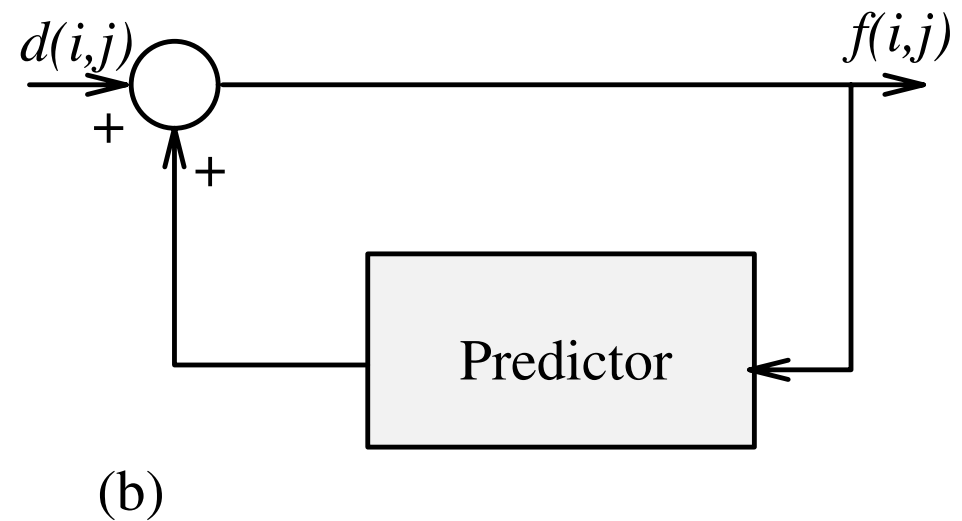
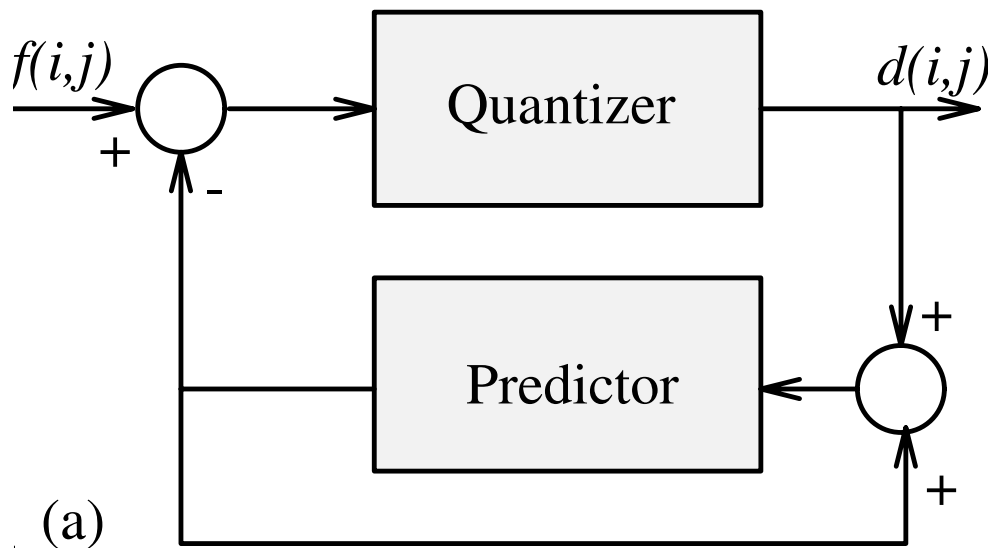
kde  $M$  je maximální interval hodnot vstupní posloupnosti, tj. 256 pro osmibitový vstup a 65356 pro šestnáctibitový vstup.

---

V praxi se používá zejména SNR a PSNR. Výraz MSE slouží jako pomocný pro jejich definici.

# Prediktivní komprese – myšlenka

- ◆ Najít matematický model, který dokáže predikovat na základě předchozích hodnot další hodnotu.
- ◆ Přenášet pouze rozdíl mezi skutečnou a predikovanou hodnotou.
- ◆ Ke kompresi dochází, protože rozdílová data mají menší statistickou variaci (např. rozptyl) než původní data.





# Digitální pulsně kódová modulace (1)

- ◆ Mějme obraz  $f(i, j)$ . Odhadněme statistické závislosti v obraze pomocí autokorelační funkce  $R(i, j, k, l) = \mathcal{E}(f(i, j) f(k, l)) = f f^\top$ .
- ◆ Hledáme matematický model prediktoru  $\hat{f}(i, j)$ .
- ◆ Rozdíl  $d(i, j) = \hat{f}(i, j) - f(i, j)$ .
- ◆ Předpokládejme např. lineární prediktor 3. řádu

$$\hat{f}(i, j) = a_1 f(i, j - 1) + a_2 f(i - 1, j - 1) + a_3 f(i - 1, j) ,$$

kde  $a_1, a_2, a_3$  jsou parametry prediktivního modelu.

$f(i, j-1)$	$f(i, j)$
$f(i-1, j-1)$	$f(i-1, j)$

## Digitální pulsně kódová modulace (2)

- ◆ Jak se odhadnou parametry prediktivního modelu  $a_1, a_2, a_3$ ?
- ◆ Vyřešením statistické optimalizační úlohy. Předpokládá se stacionární náhodný proces  $f$  s nulovou střední hodnotou.

$$e = \mathcal{E} \left( [\tilde{f}(i, j) - f(i, j)]^2 \right) .$$

a prediktor třetího řádu

$$a_1 R(0, 0) + a_2 R(0, 1) + a_3 R(1, 1) = R(1, 0)$$

$$a_1 R(0, 1) + a_2 R(0, 0) + a_3 R(1, 0) = R(1, 1)$$

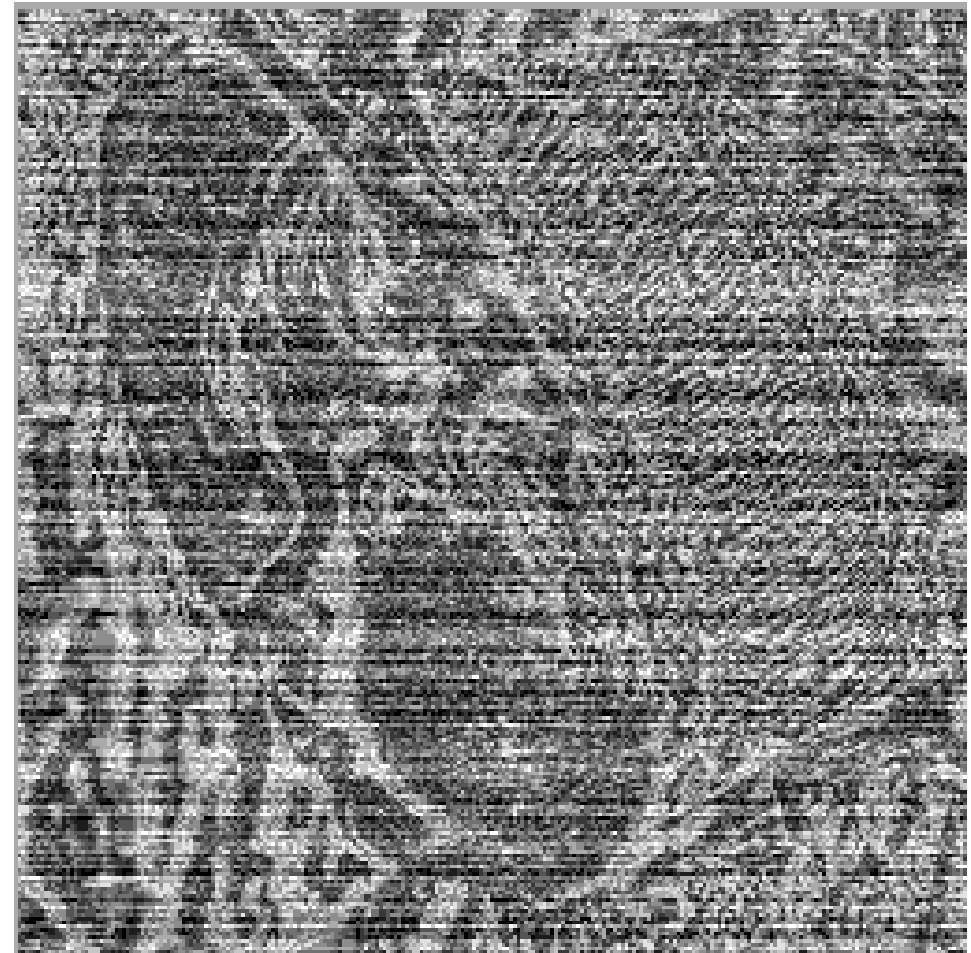
$$a_1 R(1, 1) + a_2 R(1, 0) + a_3 R(0, 0) = R(0, 1)$$

kde  $R(m, n)$  je autokorelační funkce speciálního tvaru  
 $R(\alpha, \beta) = R(0, 0) \exp(-c_1\alpha - c_2\beta)$ .

# DPCM – Příklad, $K = 3.8$



Po rekonstrukci  $K = 3.8$ .

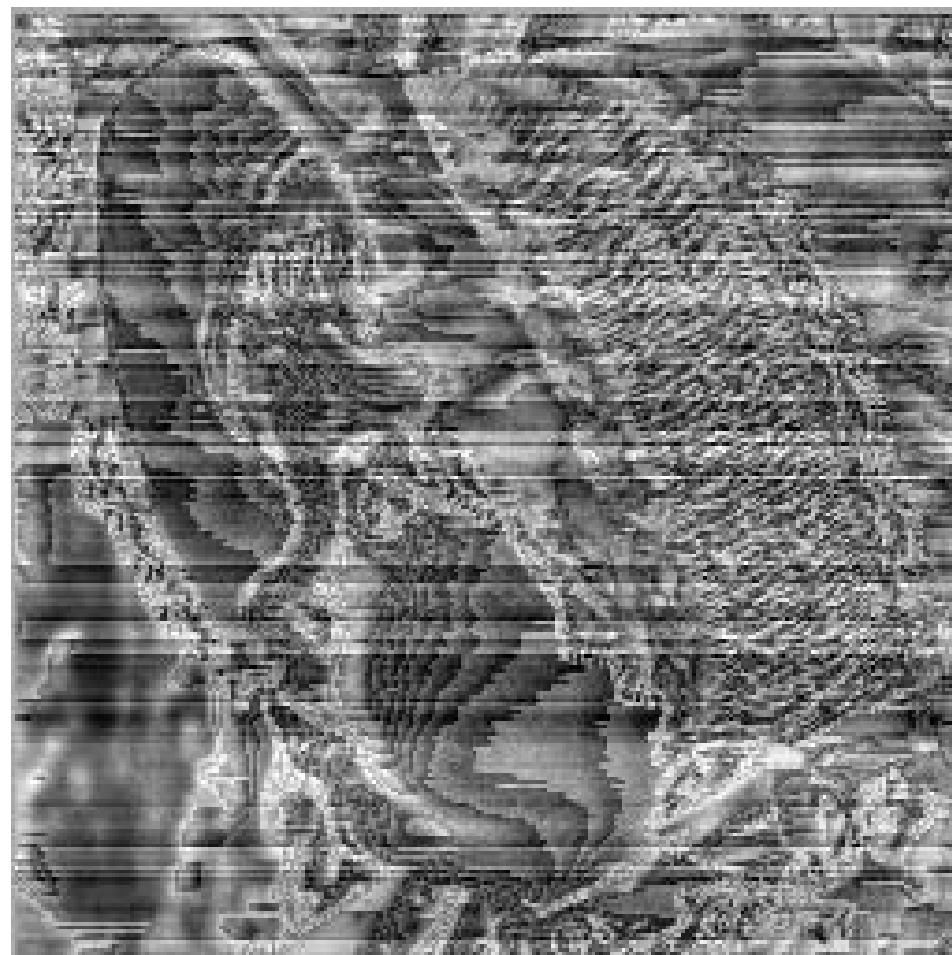


Rozdílový snímek.

# DPCM – Příklad, $K = 6.2$



Po rekonstrukci  $K = 6.2$ .



Rozdílový snímek.

## JPEG, úvod

- ◆ Joint Photographic Expert Group. Standartizováno v roce 1992.
- ◆ Používá se na šedotónové i barevné obrázky. Barevné se nejdříve převedou z barevného prostoru RGB do prostoru YUV, kde lze UV matice reprezentovat v polovičním rozlišení než matici Y ( $\approx$  intenzita).
- ◆ Existuje jako ztrátová i bezztrátová komprese.
- ◆ První generace (.jpg) z 1992 používá DCT (diskrétní kosínovou transformaci) pro odstranění redundance a irelevance. Pro optimální kódování se použije převod koeficientů DCT do 1D vektoru, kódování úseky řádků a symboly kóduje Huffmanovým kódováním.
- ◆ Druhá generace JPEG2000 (.jp2) z roku 2000 odstraňuje redundanci a irelevanci pomocí vlnkové transformace. Potom kóduje v jednotlivých bitových rovinách a symboly kóduje aritmetickým kódováním.

## Proč JPEG zvolilo DCT?

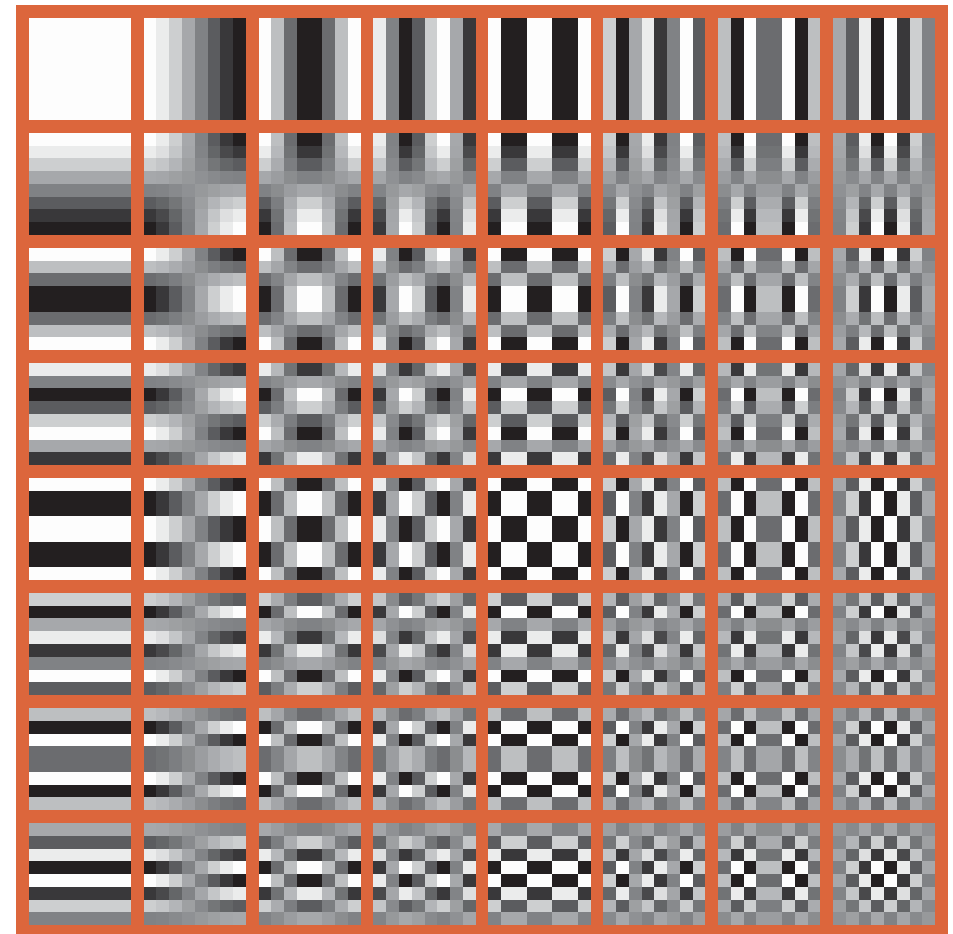
- ◆ DCT je implicitně periodická. Nejsou takové problémy s nespojitostí.
- ◆ DCT dobře aproximuje PCA (metoda hlavních směrů, Principal Component Analysis, Karhunen-Loeve rozvoj), která je optimální z hlediska středněkvadratické chyby (energie).
- ◆ Na rozdíl od PCA, DCT má pevně zvolené bázové funkce.
- ◆ Aby se ušetřil výpočetní čas, je obraz rozdělen na bloky  $8 \times 8$ , které jsou komprimovány nezávisle na sobě

# DCT, bázové funkce

Pevně zvolených 64 bázových funkcí.

- ◆ Každý blok obrazu  $8 \times 8$  lze vyjádřit jako lineární kombinaci bázových funkcí.
- ◆ Výpočet DCT slouží k nalezení vah lineární kombinace.
- ◆ Váhy jsou prahovány. Velikost prahu ovlivňuje míru komprese, tj. volí se irelevance.

DCT2 base for [8x8] block



# Příklad, kameraman

image block



image intensities

1	185	187	184	183	189	186	185	186
2	185	184	186	190	187	186	189	191
3	186	187	187	188	190	185	189	191
4	186	189	189	189	193	193	193	195
5	185	190	188	193	199	198	189	184
6	191	187	162	156	116	30	15	14
7	168	102	49	22	15	11	10	10
8	25	19	19	26	17	11	10	10
	1	2	3	4	5	6	7	8



# Příklad, kameraman, DCT

image intensities

1	185	187	184	183	189	186	185	186
2	185	184	186	190	187	186	189	191
3	186	187	187	188	190	185	189	191
4	186	189	189	189	193	193	193	195
5	185	190	188	193	199	198	189	184
6	191	187	162	156	116	30	15	14
7	168	102	49	22	15	11	10	10
8	25	19	19	26	17	11	10	10
	1	2	3	4	5	6	7	8

coefficients of the DCT2

1	1117	114	10	7	19	-2	-7	2
2	459	-119	-20	-11	-16	-4	3	0
3	-267	-3	24	8	1	6	4	-1
4	50	107	-9	-1	11	-6	-7	3
5	52	-111	-22	-2	-16	-2	5	-3
6	-38	39	46	19	2	0	4	3
7	-17	39	-46	-26	8	-5	-10	2
8	30	-46	28	22	-9	2	7	-1
	1	2	3	4	5	6	7	8

# Příklad, kameraman, 100 % a 50 %

100 % of most significant DCT2 coeffs



50 % of most significant DCT2 coeffs



# Příklad, kameraman, 20 % a 5 %

20 % of most significant DCT2 coeffs



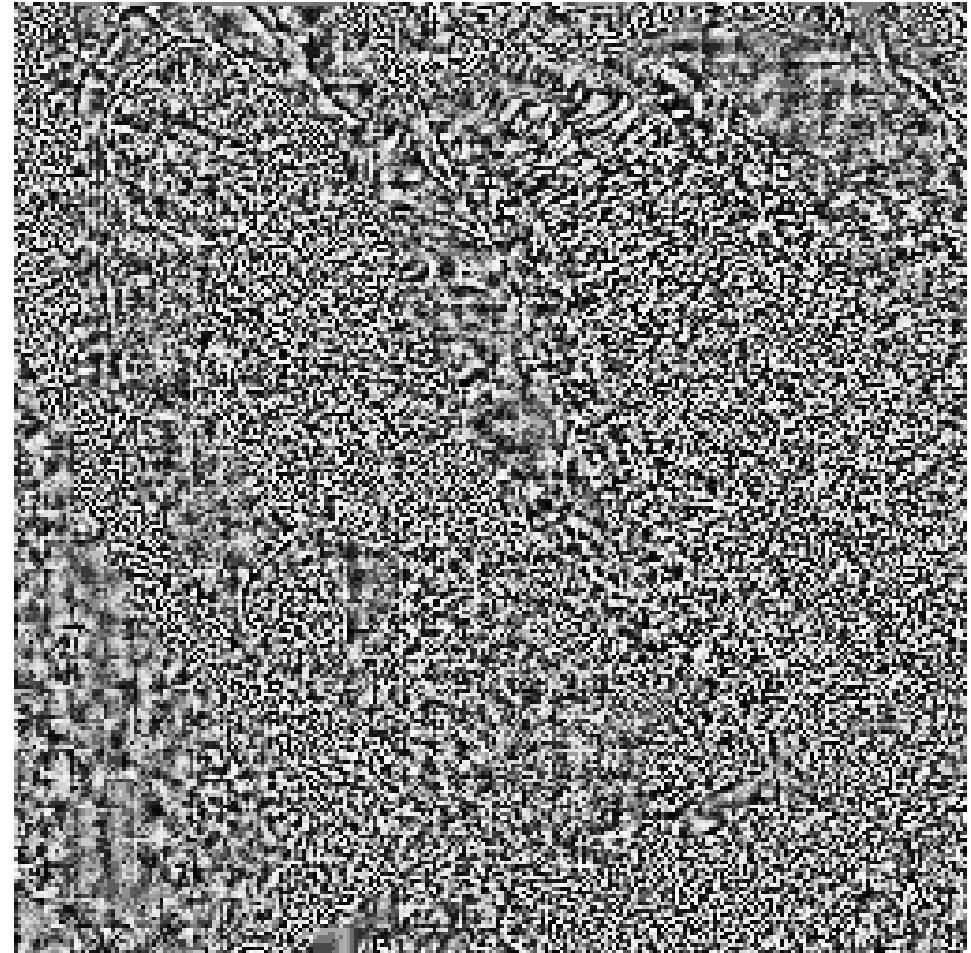
5 % of most significant DCT2 coeffs



# JPEG – příklad, $K = 3.8$



Po rekonstrukci  $K = 3.8$ .

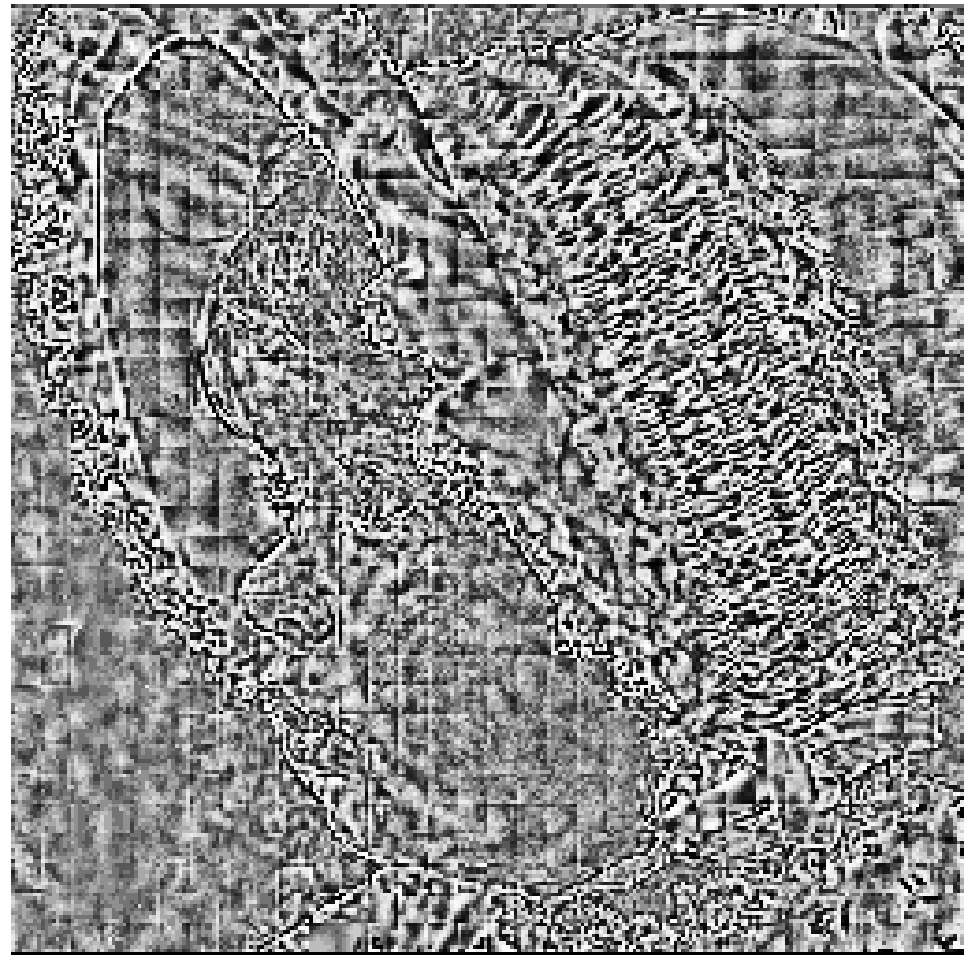


Rozdílový snímek.

# JPEG – příklad, $K = 4.2$



Po rekonstrukci  $K = 4.2$ .

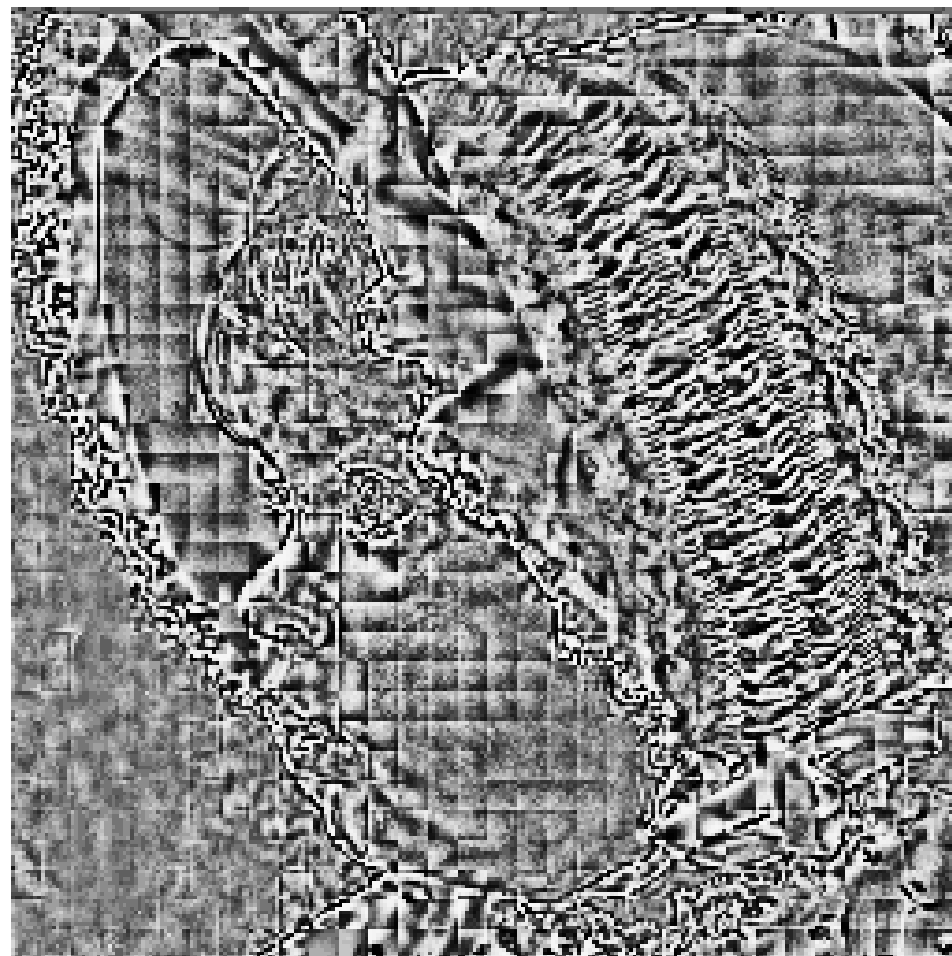


Rozdílový snímek.

# JPEG – Příklad, $K = 5.6$



Po rekonstrukci  $K = 5.6$ .

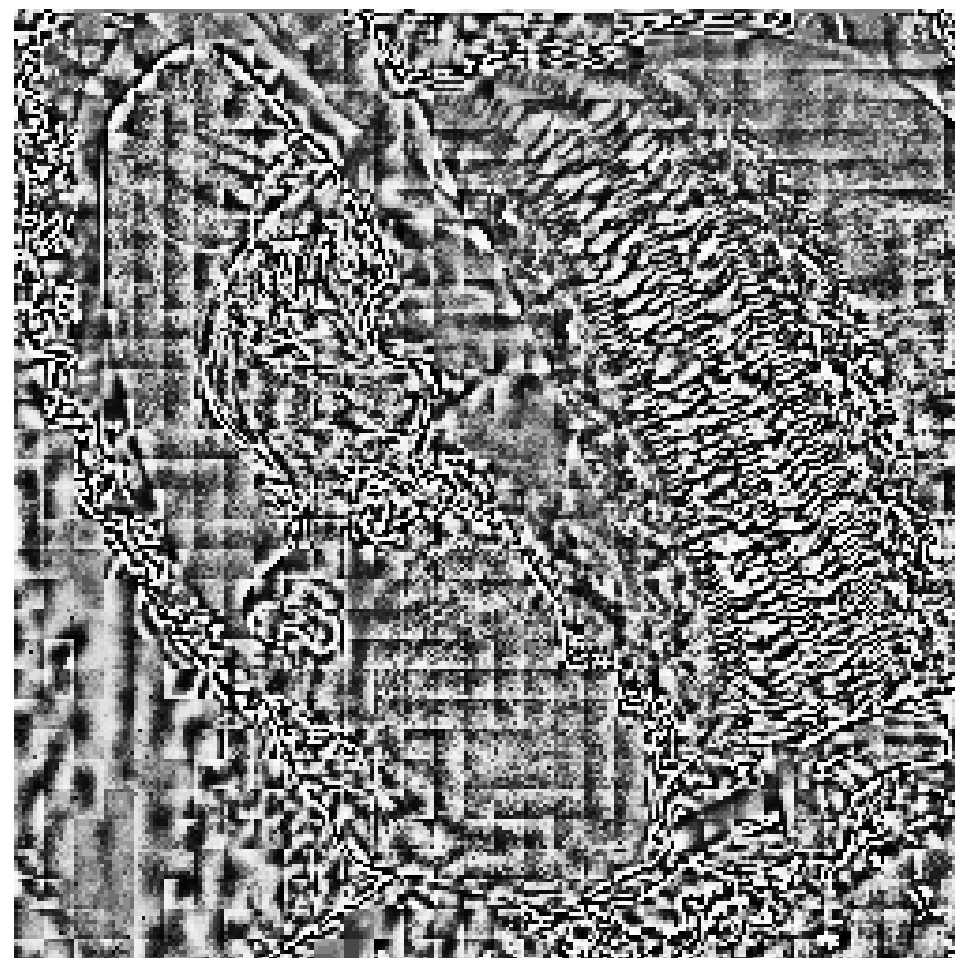


Rozdílový snímek.

# JPEG – příklad, $K = 10.2$



Po rekonstrukci  $K = 10.2$ .



Rozdílový snímek.