

Multimediální systémy

Jan Outrata



KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLMOUCI

přednášky



Authoring

- = **vytvoření (*interaktivního*) multimediálního obsahu** – získáním, úpravami a složením více typů médií dohromady
- vyžaduje znalost typů médií, manipulace s nimi a jejich kombinace (včetně programování/skriptování), umělecké cítění aj.
- náročné na čas: množství, velikost médií, požadavky na layout/strukturu (prostorové, časové), způsoby interakce aj.
- **authoring nástroje** – pro úpravu konkrétních typů médií nebo pro kombinaci různých typů, přidání interaktivity, přizpůsobení pro distribuci a zobrazení (formát, velikost, layout/struktura – profily, dle např. reprodukčního zařízení, typu přenosu aj.)
- **pasivní obsah** – uživatelská akce nemá efekt na (lineární) průběh, např. film/TV (video), hudba/rádio, kniha
- **aktivní (interaktivní) obsah** – (nelineární) průběh zvolený uživatelem, např. hra, DVD/Blue-ray disk, video-on-demand (streamovací) aplikace, fotoalbum, hypertextový dokument(s odkazy, např. webový) aj.



- získání (capturing), úpravy ((post-)processing) a uložení v různých formátech (formatting) pro jednotlivé typy médií – **intramedia (media) authoring**
- prostorová a časová organizace prvků různých typů médií = **layout** – **intermedia (multimedia) authoring**
- specifikace způsobů **interakce** mezi prvky = vytvoření (nelineárního) průběhu s možností volby – **intermedia (multimedia) authoring**
- ideálně podpora různých platform a způsobů interakce, obsah nezávislý na platformě, v praxi pro konkrétní platformu(y)

- nástroje pro jednotlivé typy médií, např. video-, audio-**editory**, editory obrázků, grafiky (animace), textový editor apod.
- získání a úpravy médií:
 - obraz: kreslení, ořezání, transformace, kompozice, filtrace, (prostorové) efekty, ...
 - video/animace: obraz/grafika + střih, změna rychlosti, skenování, (časoprostorové) efekty, ...
 - audio: generování zvuku, střih, změna rychlosti, kompozice filtrace, (časové) efekty, ...
 - grafika: tvorba objektů, ořezání, transformace (i barev, např. osvětlení), kompozice, **renderování** = převod z vektorové do rastrové grafiky (obrazu), ...
- nastavení parametrů formátu pro uložení, např. komprese

- **specifické nástroje** pro specifické formáty a použití, např. audio-video (kontejner, stream), DVD/Blue-ray, herní enginy
- kategorizace na základě prostorové a časové organizace prvků médií a specifikace interakce:
 - **prostorová organizace (layout)**: rozmístění prvků, kompozice v prostoru, např. vložení grafiky do videa – šablony, graf scény
 - **časová organizace**: rozmístění prvků, kompozice v čase, např. synchronizace videa a audia, dynamická kompozice – časové rozvrhy
 - **specifikace interakce**: umístění uživatelských ovládacích prvků pro události a specifikace jejich akce (navigace, např. „přehrávač“), programovatelné složené akce, např. interaktivní dynamické prostorové a časové organizace – multimediální aplikace, DVD/Blue-ray disk, interaktivní TV, hry, . . . , webová stránka

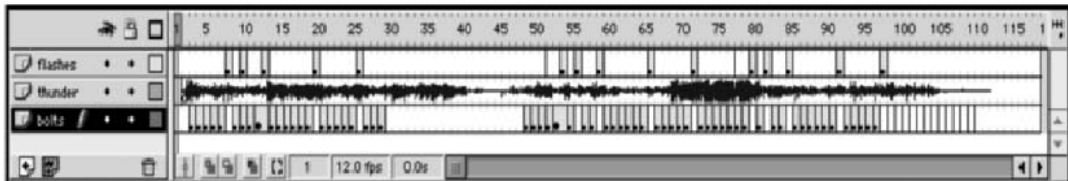
- **časová osa (timeline)** – osnovy, kanály, „přehrávač“, pro úpravy video a audio médií a časovou organizaci („režie“)

Obrázek: Timeline

- **skriptování** – skriptovací jazyk pro programování akcí „za hranicemi UI“ (dynamické organizace, interaktivita), automatizace, např. JavaScript, SMIL (streaming), VRML (3D grafika), XMT (MPEG4)

Obrázek: SMIL skript

- další, např. flow control (diagram akcí mezi prvky médií), karty (prostorová organizace, např. u prezentací) aj.



```
<smil>
  <head>
    <layout>
      <region id="Region1" top="5" left="20" />
      <region id="Region2" top="100" left="20" />
    </layout>
  </head>
  <body>
    <image region="Region1"
      src="http://www.mysite.org/logo.jpg"
      dur="10s" />
    <video region="Region2"
      src="http://www.mysite.org/myvideo.avi"
      clipBegin="10" clipEnd="20">
    </video>
  </body>
</smil>
```




- jednoduchá, komfortní interakce pro jednotlivé typy médií a multimédií
- vizuální/grafická média, ale nejen ⇒ **grafické UI** (z PC desktopu, pro mobilní zařízení), např. tlačítka (pro „**přehrávač**“), ikony, dialogy (výběr TV kanálu/streamu)
- prostorová omezení zařízení (nejen displeje), jiné (dotykové) ovládání: např. mobilu, tabletu → upravené UI, responzivita
- spolupráce (přenosných) propojených a synchronizovaných zařízení sdílejících obsah



- **nezávislost na zařízení** = stejně nebo podobně koncipovaný obsah na různých zařízeních, „**univerzální přístup**“ (sít, HW/SW platforma, lokace, jazyk apod.) – univerzální formáty definující možnosti zařízení, např. HTML5, více různě velkých forem obsahu pro distribuci → **profily** (MPEG4+)
- **distribuovaný authoring** = současná spolupráce více lidí na obsahu, použití upravených systémů na správu verzí z vývoje SW (využívající mediální prostoročasové sémantiky dat) – nároky na síť (velký objem, rychlá odezva), řešení konfliktů (zamykání objektů), náročná organizace (např. real-time), např. WebDAV pro web
- přidružené služby – např. **(automatická) správa obsahu** = prohlížení, vyhledávání, výběr, vytváření přizpůsobeného, aktualizace, sdílení – databáze, indexace, metadata, archivy, různé distribuce, např. Google News (?)



Kompresa

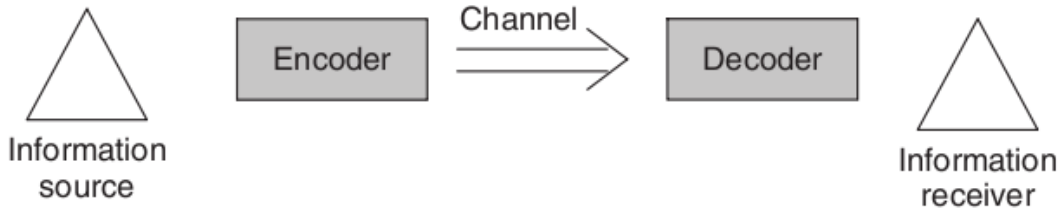


- velké objemy dat, potřeba uložit a distribuovat po různých sítích
- dříve omezené kapacity a rychlosti, dnes větší objemy – např. video HDTV 1080p 30 fps 12 bpp (YUV 4:2:0) = 712 Mb/s, obraz 10 Mpx 24bpp (RGB) ~ 30 MB
- komprese = **redukce dat při zachování (přijatelné úrovně) obsažené informace** – využití uspořádání a vzorů (a redundancí na přijatelnou úroveň) v multimediálních datech

- Claude Shannon, 1940s
- = **efektivita a spolehlivost přenosu informace ze zdroje k cíli daným (chybovým) komunikačním kanálem** – **minimální objem dat** (pomocí komprese) a zabezpečení správnosti dat (pomocí přidání kontrolních dat pro detekci a opravy chyb)
- **enkoder** a **dekoder**, produkované informační **symboly** = pravděpodobnostní proměnné, bez sémantiky

Obrázek: Shannonův komunikační model

- data = posloupnost bitů s hodnotami 0 a 1 (digitální binární forma)



- **abeceda** = neprázdná konečná množina symbolů, $S = \{s_1, \dots, s_n\}$, např. 256 hodnot v 8 b, 26 znaků abecedy aj.
- **kódování (kód)** = zobrazení symbolů na posloupnosti bitů (kódová slova pro symboly), $C = \{\langle s_1, c_1 \rangle, \dots, \langle s_n, c_n \rangle\}$, např. ASCII tabulka
- **zpráva** = konečná posloupnost symbolů z abecedy, např. $s_1 s_2 s_1 s_4$, reprezentována proudem bitů
- **informace (informační obsah)** zprávy \sim počet a uspořádání symbolů ve zprávě – důležitá frekvence výskytů symbolů
- **pravděpodobnost (výskytu) symbolu**: $p_i = \frac{f_i}{|M|}$, kde f_i je počet (frekvence) výskytů symbolu s_i ve zprávě M délky $|M|$ – použita pro optimální kódování symbolů/zprávy s méně bity pro pravděpodobnější symboly – kódy **proměnné délky** l_i

Obrázek: Kódování symbolů a zprávy

- **jednoznačně dekódovatelný kód** = v proudu bitů je možné jednoznačně rozpoznat kódová slova symbolů, např. první 2 kódy z předchozího obrázku
- **prefixový kód** = žádné kódové slovo není prefixem jiného
- **Kraftova-McMillanova věta** – prefixové kódy jsou dostačující

Symbol	Code	Number of occurrences (probability)	Bits used by each symbol
s_1	00	70 (0.7)	$70 \times 2 = 140$
s_2	01	5 (0.05)	$5 \times 2 = 10$
s_3	10	20 (0.2)	$20 \times 2 = 40$
s_4	11	5 (0.05)	$5 \times 2 = 10$

200

Symbol	Code	Number of occurrences (probability)	Bits used by each symbol
s_1	1	70 (0.7)	$70 \times 1 = 70$
s_2	001	5 (0.05)	$5 \times 3 = 15$
s_3	01	20 (0.2)	$20 \times 2 = 40$
s_4	000	5 (0.05)	$5 \times 3 = 15$

140

Symbol	Code	Number of occurrences (probability)	Bits used by each symbol
s_1	0	70 (0.7)	$70 \times 1 = 70$
s_2	01	5 (0.05)	$5 \times 2 = 10$
s_3	1	20 (0.2)	$20 \times 1 = 20$
s_4	10	5 (0.05)	$5 \times 2 = 10$

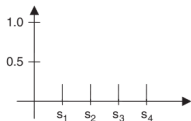
110

- délka zprávy v počtu symbolů: $|M| = \sum_{i=1}^n f_i$
- délka zprávy v bitech: $l_M = \sum_{i=1}^n f_i l_i$
- **průměrná délka symbolu** v bitech: $\lambda_M = \frac{l_M}{|M|} = \sum_{i=1}^n p_i l_i$
- **cíl komprese = nalézt prefixový kód takový, že λ_M je minimální** – tj. když p_i roste, l_i musí klesat
- **entropie**: $E_M = -\sum_{i=1}^n p_i \log_2 p_i$, teoreticky minimální λ_M (Shannon), slouží k **vyjádření míry informace ve zprávě M** , nejvyšší při stejné pravděpodobnosti symbolů – např. při náhodném výskytu symbolů
 - $\log_2 \frac{1}{p_i} =$ vlastní informace symbolu s_i , reprezentuje l_i
 - teoretický limit komprese

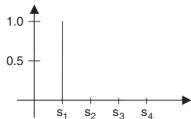
Obrázek: Entropie

- **efektivita** (kódu): $\frac{E_M}{\lambda_M}$, míra „účinnosti“ komprese, nejlepší = 1

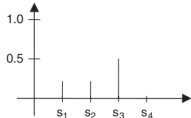
$$P_i = \{0.25, 0.25, 0.25, 0.25\}$$
$$H = -(4 \times 0.25 \times \log_2 0.25)$$
$$H = 2$$



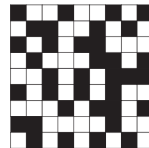
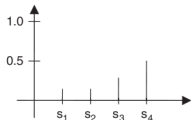
$$P_i = \{1.0, 0.0, 0.0, 0.0\}$$
$$H = -(1 \times \log_2 1)$$
$$H = 0$$



$$P_i = \{0.25, 0.25, 0.5, 0.0\}$$
$$H = -(2 \times 0.25 \times \log_2 0.25 + 0.5 \times \log_2 0.5)$$
$$H = 1.5$$



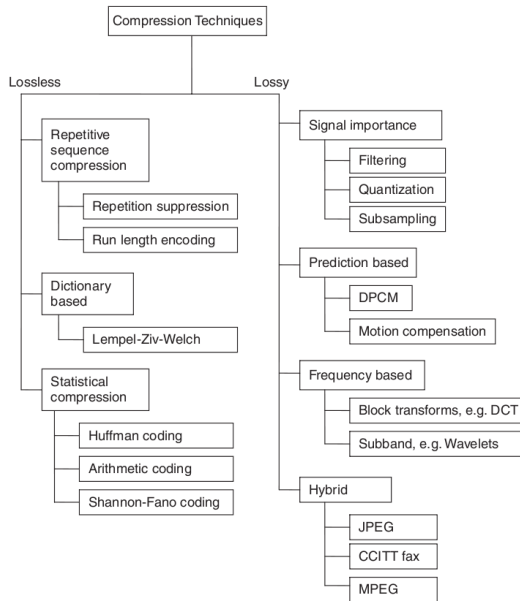
$$P_i = \{0.125, 0.125, 0.25, 0.5\}$$
$$H = -(2 \times 0.125 \times \log_2 0.125 + 0.25 \times \log_2 0.25 + 0.5 \times \log_2 0.5)$$
$$H = 1.75$$



- **bezeztrátová (lossless)** = po dekompresi stejná data, bez ztráty informace
- **ztrátová (lossy)** = po dekompresi jiná data (podobný signál), změna informace, **zkreslení dat (signálu)**
- používané samostatně i v kombinaci v metodách komprese multimediálních dat

Obrázek: Taxonomie kompresních metod

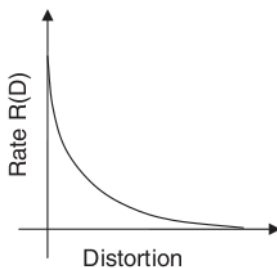
- **compression (bit) rate** = komprimovaná velikost dat v bitech na symbol/vzorek/pixel/sekundu, použití u přenosu dat
 - **proměnná** u bezztrátové komprese (proměnná délka kódových slov symbolů), udržována **konstantní** u ztrátové komprese (proměnným zkreslením dat) – výhodnější pro úpravy a (real-time) přenos
- **compression ratio (kompresní poměr)** = poměr velikosti (rate) původních a komprimovaných dat, použití u uložení dat



- = vztah mezi compression (bit) rate a mírou zkreslení (změny informace) u ztrátové komprese – nepřímá úměra, použití pro hodnocení efektivity metod
- **míra zkreslení** $f(y - \hat{y})$: typicky měření rozdílů mezi původními (y) a dekomprimovanými (\hat{y}) daty
 - např. součet rozdílů hodnot pixelů, populární střední kvadratická chyba (**mean square error**) nebo poměr (špičkového) signálu k šumu (**(peak) signal-to-noise ratio, (P)SNR**)
 - čím vyšší (a vyšší kompresní poměr), tím menší bit rate a naopak → komprese = kompromis

Obrázek: Rate distortion a míry zkreslení

- nemusí odpovídat **vnímanému zkreslení**, např. posunutí obrazu o pixel



$$\sigma_d^2 = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$SNR = \frac{\sigma_{\hat{y}}^2}{\sigma_d^2}, \quad SNR(db) = 10 \times \log\left(\frac{\sigma_{\hat{y}}^2}{\sigma_d^2}\right)$$

Where $\sigma_{\hat{y}}$ is the averaged squared value of the output

$$PSNR(db) = 10 \times \log\left(\frac{\sigma_{peak}^2}{\sigma_d^2}\right)$$

Where σ_{peak} is the squared value of the peak output

- = odstranění **statistické redundance** v datech
- kratší kódová slova pro častější (pravděpodobnější) symboly a naopak
- **statistické metody** – použití **pravděpodobnostních modelů** = (odhadů) pravděpodobnosti výskytů symbolů dat spočítaných před (**semi-adaptivní**) nebo během (**adaptivní**) komprese nebo empiricky zjištěných (**statický**)
 - semi-adaptivní model je potřeba uložit s komprimovanými daty
- **slovníkové metody** – použití **slovníku** = seznamu dříve se v datech vyskytujících řetězců symbolů vytvořeného typicky během komprese

Run Length Encoding (RLE)

- = kódování posloupností opakujících se symbolů dvojicemi počet opakování a symbol (nebo novými symboly)
- detaily: kódování počtu opakování, nekódování posloupností kratších než dvojice, aj.
- použití ve standardech pro kompresi obrazu/grafiky a audia

- = kódování dříve se vyskytujících řetězců symbolů, vzorů, ukládaných do **slovníku**, pozicemi ve slovníku
- vytváření slovníku při kompresi i dekompresi
- (explicitně) nevyužívají pravděpodobností výskytu symbolů
- Lempel, Ziv, 1970s: **metody LZ***

LZW

- uložení samostatných symbolů do slovníku
- komprese = opakování kódování nejdelšího řetězce na vstupu, který je ve slovníku, a uložení spojení řetězce a dalšího symbolu na vstupu do slovníku
- dekomprese = opakování uložení spojení předchozího dekódovaného řetězce a prvního znaku aktuálně, popř. předchozího, dekódovaného řetězce do slovníku a dekódování kódu

Obrázek: LZW komprese

- detaily: omezená velikost slovníku (velký = lepší komprese, ale delší kódy a doba vyhledávání, v praxi 4096), vyprázdnění slovníku
- použití např. v obrazovém formátu GIF

	x	x	y	y	x	y	x	y	x	x	y	y	x	y	x	x	y	x	x	y	y	x
	0	0	1	1	3	6	3	4	7	5	8	0										
Iteration number	Current length symbol(s)	Dictionary code used	Next symbol in string	New dictionary code generated	New dictionary index generated																	
				x	0																	
				y	1																	
1	x	0	x	xx	2																	
2	x	0	y	xy	3																	
3	y	1	y	yy	4																	
4	y	1	x	yx	5																	
5	xy	3	x	yx	6																	
6	xyx	6	x	xyxx	7																	
7	xy	3	y	xyy	8																	
8	yy	4	x	yyx	9																	
9	xyxx	7	y	xyxxy	10																	
10	yx	5	x	yxx	11																	
11	xyy	8	x	xyyx	12																	
12	x	0	END																			

Index	Entry	Index	Entry
0	x	7	xyxx
1	y	8	xyy
2	xx	9	yyx
3	xy	10	xyxxy
4	yy	11	yxx
5	yx	12	xyyx
6	xyx		

- = vytvoření optimálního prefixového kódu s kratšími kódovými slovy pro pravděpodobnější symboly a naopak
- Huffman, 1952
- (Huffmanovy) kódy symbolů = posloupnosti označení 0 a 1 hran na cestách binárním stromem od kořene k listům označených symboly
- konstrukce **(Huffmanova) stromu** – pravděpodobnější symboly blíže kořenu:
 - 1 listy (setříděně) označeny pravděpodobnostmi výskytu symbolů
 - 2 opakování vytvoření rodiče pro 2 uzly označené minimálními pravděpodobnostmi a označení rodiče součtem pravděpodobností a hran k rodiči 0 a 1

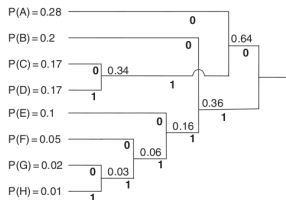
Obrázek: Huffmanovo kódování

Symbol	Probability	Binary code	Code length
A	0.28	000	3
B	0.2	001	3
C	0.17	010	3
D	0.17	011	3
E	0.1	100	3
F	0.05	101	3
G	0.02	110	3
H	0.01	111	3

Average symbol length = 3

Entropy = 2.574

Efficiency = 0.858



Symbol	Probability	Huffman code	Code length
A	0.28	00	1
B	0.2	10	3
C	0.17	010	3
D	0.17	011	3
E	0.1	110	3
F	0.05	1110	4
G	0.02	11110	5
H	0.01	11111	5

Average symbol length = 2.63

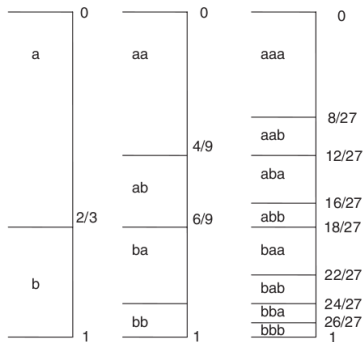
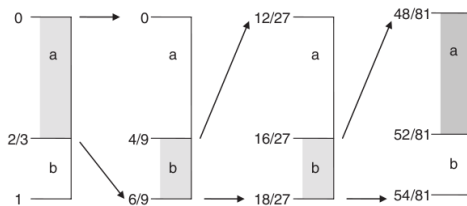
Entropy = 2.574

Efficiency = 0.9792

- **kódování řetězce symbolů** jako celku místo jednotlivých symbolů (kdy je potřeba celý počet bitů pro kódové slovo symbolu)
- = kódování řetězce symbolů do dvojice minimální binární reprezentace racionálního čísla v **podintervalu intervalu** $[0, 1)$ a délky řetězce
- konstrukce podintervalu:
 - 1 interval nastaven na $[0, 1)$
 - 2 opakování rozkladu intervalu na podintervaly s délkami proporčně odpovídajícími pravděpodobnostem výskytu pevně uspořádaných symbolů a nastavení intervalu na podinterval pro symbol na vstupu

Obrázek: Aritmetické kódování, $[0.10010111, 0.10100100]$, $0.101 = \frac{50}{80}$, 101

- detaily: zvětšení a posunutí malého podintervalu kolem 0.5, celočíselná aritmetika, rozdělení dat na bloky a kódování bloků



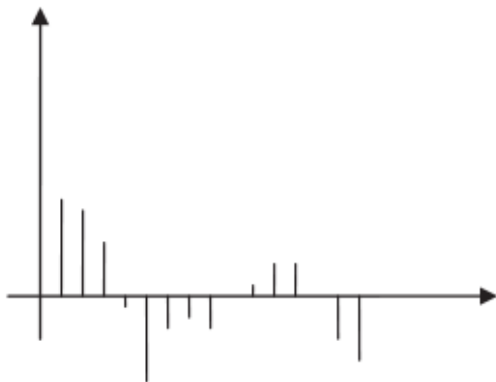
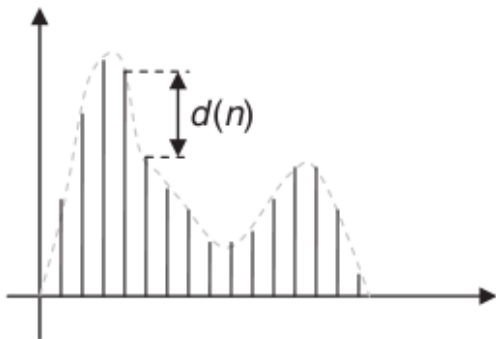


- = **změna („ztráta“)** informace a **zkreslení** (při vnímání) dekomprimovaných dat pro dosažení vyšší komprese, za teoretický limit bezztrátové
- určitá míra zkreslení je lidskými smysly nerozpoznatelná a dále pak (v závislosti na aplikaci) akceptovatelná
- kvantizace je ztrátová sama o sobě

- signál o nízké frekvenci se mění (v čase) pomalu → vzorky mají blízké hodnoty ⇒ mohou být **predikovány z předchozích hodnot** – metody založené na predikci (**prediction-based**)
- nejjednodušší predikce = přímo předchozí hodnota
- = kódování **chyby predikce (prediction error)** = rozdílu mezi predikovanou a skutečnou hodnotou – má menší rozsah než signál samotný a tedy menší entropii = méně bitů pro reprezentaci – kvantizaci

Obrázek: DPCM

Diferenční PCM (DPCM)



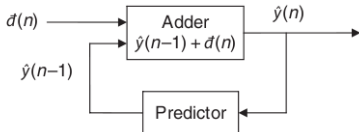
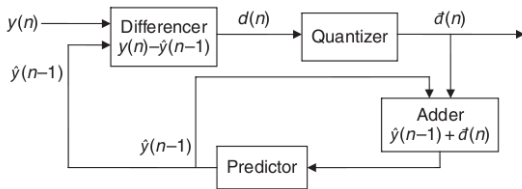
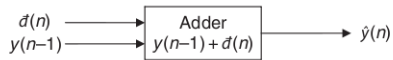
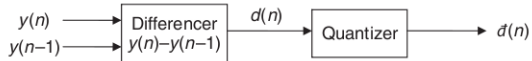
- **open loop případ** = na výstupu enkoderu kvantizovaný rozdíl mezi z předchozích hodnot predikovanou a skutečnou hodnotou signálu – **problém**: dekoder nemá k dispozici přesné předchozí hodnoty, ale pouze zkršené, o kvantizační chybu, která se ovšem akumuluje

Obrázek: Open loop případ

- **closed loop případ** = na výstupu enkoderu kvantizovaný rozdíl mezi z **dekódovaných** (zkršených) předchozích hodnot predikovanou a skutečnou hodnotou signálu – redukce akumulace kvantizační chyby, dekoder jako součást enkoderu

Obrázek: Closed loop případ

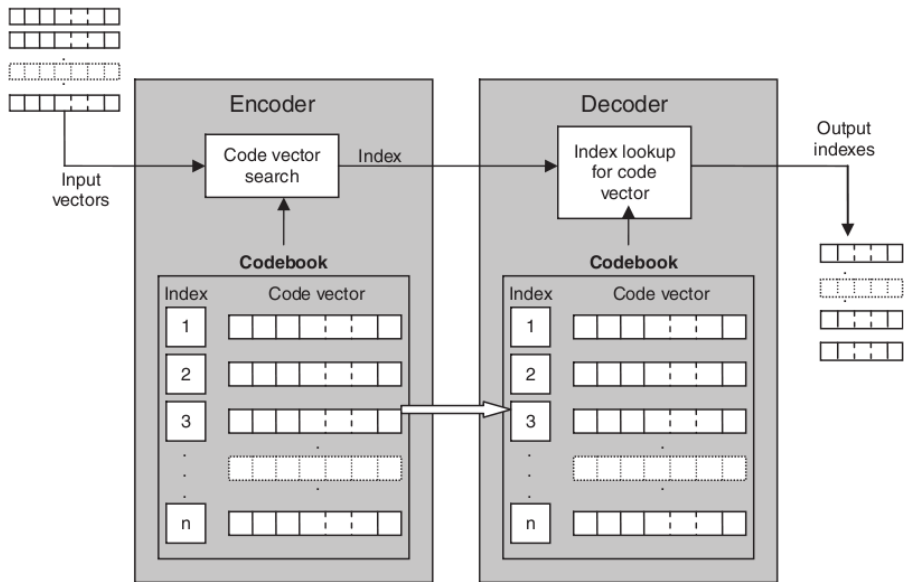
Diferenční PCM (DPCM)



- skalární kvantizace = na jednotlivých vzorcích – jednoduchá, rychlá, ale nezohledňuje korelace mezi (sousedními) vzorky
- = kvantizace **skupin (vektorů) vzorků** – zohlednění korelací → lepší komprese
 - vytvoření **slovníku vektorů**: statistická analýza signálu nebo empiricky (trénováním), nejlepší reprezentanti s maximálním tolerovaným zkreslením, otázka velikosti slovníku (větší = menší zkreslení, ale více místa, např. pro obraz 128–2048) a vektorů (větší = více korelací)
 - kódování vektorů na vstupu pozicemi vektorů ve slovníku s minimálním zkreslením (měřeným např. střední kvadratickou chybou)

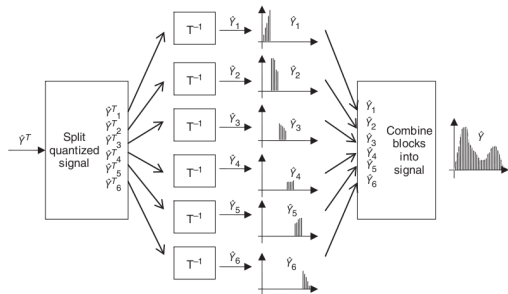
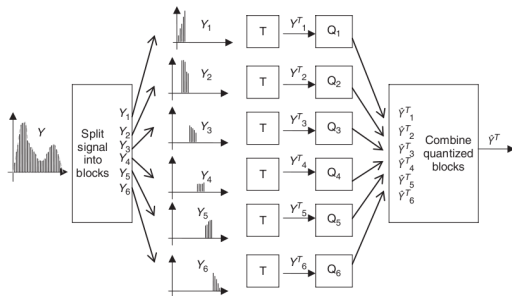
Obrázek: Vektorová kvantizace

- např. barevné obrazy – pixel má barvu reprezentovanou hodnotami v barevných kanálech = vektor, výběr barev = slovník (paleta barev) – tzv. **indexované barvy**



- = (invertibilní) transformace signálu z domény vzorků do jiné domény s nižší entropií signálu a kvantizace (ve vhodném pořadí prvků domény)
- **frekvenční** – do frekvenční domény, reprezentace koeficienty základních frekvencí, např. diskrétní **Fourierova/kosinová**, Hadamard, Lapped Orthogonal aj.
- **statistické** – do (méně nebo) nekorelovaných dimenzí, např. pomocí **analýzy hlavních komponent (PCA)**, např. Karhunen-Loeve
- **podpásmové (subband)** – do vícespektrální frekvenční domény
- transformace **bloků signálu**, např. 8x8 pixelů u JPEG komprese, a nezávislé kvantizace (s různými rozsahy) – pro kontrolu bit rate v závislosti na entropii

Obrázek: Blokově transformační kódování



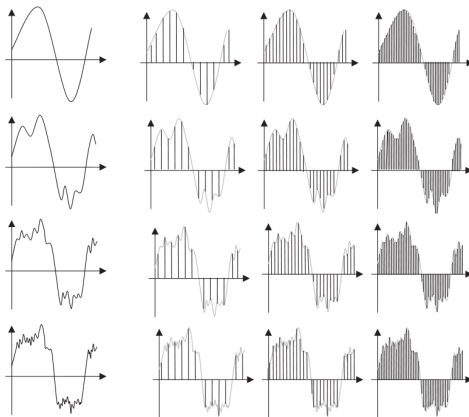
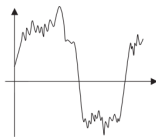
- **postupné jemnější aproximace/rozlišení** (spektrálního) popisu **signálu** rozděleného do **(pod)pásem**, s přidáváním vyšších frekvencí (tj. detailů) v rámci aproximace

Obrázek: Pásmový popis signálu

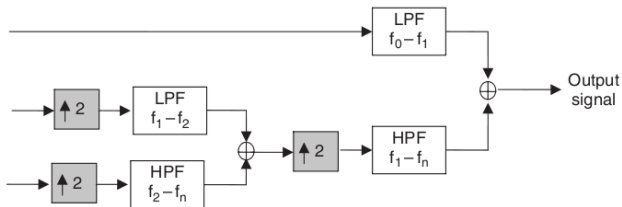
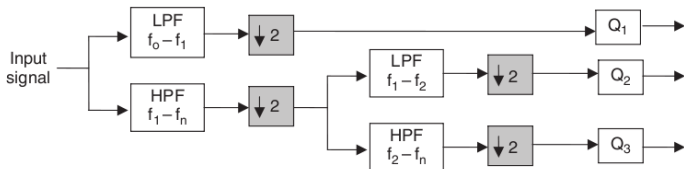
- = nezávislé (transformační) kódování signálů v pásmech v závislosti na významu pásma (ve vnímání) a výšce frekvencí pásma (určuje frekvenci vzorkování), např. u audia
- rozložení signálu do pásem – fitrace, např. band-pass pomocí **filtrových bank**
- **waveletová (víceúrovňová pásmová)** transformace/komprese = rozložení signálu do nízkofrekvenčního a vysokofrekvenčního pásma, poloviční podvzorkování, rekurzivní opakování na vysokofrekvenční, poté (transformační) kódování signálů (frekvenčních koeficientů) v pásmech

Obrázek: Waveletová transformace, LPF \leftrightarrow HPF

Podpásmové (subband) kódování



Podpásmové (subband) kódování



 Downsampling by 2

 Upsampling by 2





- = kombinace ztrátových a bezztrátových metod (v tomto pořadí!) – **ztrátové** (kvantizace) **snižují entropii**
- např. JPEG komprese: DCT bloků obrazu a DPCM kvantizovaných frekvenčních koeficientů, pak RLE a Huffmanovo kódování



- **rychlost a (časová/prostorová) složitost** kódování – real-time (horší) vs. offline (lepší komprese, rychlost obráceně), vliv množství signálu dostupného před kompresí (lepší komprese, vyšší složitost)
- **omezení bit rate** – variabilní pro uložení, konstantní pro úpravy a přenos → spolupráce ztrátové a bezztrátové části pro minimální zkreslení při omezené bit rate (**rate control**, rate distortion teorie – aktivní výzkum)
- **symetrické vs. asymetrické** komprese – enkoder a dekoder zhruba stejně vs. různé složitě/rychlé, u multimédií přirozeně asymetrické se složitějším (a lepším) enkoderem (offline) a rychlejším dekoderem (real-time), opačně např. u zálohování
- **adaptivní vs. neadaptivní** komprese – adaptace kompresních technik (modelu dat) podle předchozích dat, neadaptivní s empiricky zjištěným modelem dat, semi-adaptivní s analytickým statistickým průchodem před kompresí (**víceprůchodové komprese**)