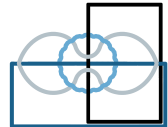


Byl jednou jeden JPEG ... a je tu s námi stále

Pavel Rajmic

Vysoké učení technické v Brně
Signal Processing Laboratory

25. října 2023
Seminář MPN, MFF UK, Praha



„To je nějaká problematika, ten JPEG?“

Úvod

Počítačové formáty pro rastrové obrázky

- bmp
- tiff
- gif
- png
- jpg
- j2k
- bpg
- webp
- avif
- ...

Úvod

Co znamená „JPEG“

- JPEG Joint Photographic Experts Group
 - skupina ustanovená 1986
 - algoritmus
 - přípona souboru (.jpeg, .jpg)
- JFIF JPEG File Interchange Format
- Základní mód (algoritmus) standardizován v r. 1992
- Existuje několik dalších módů, ale jsou málo využívané

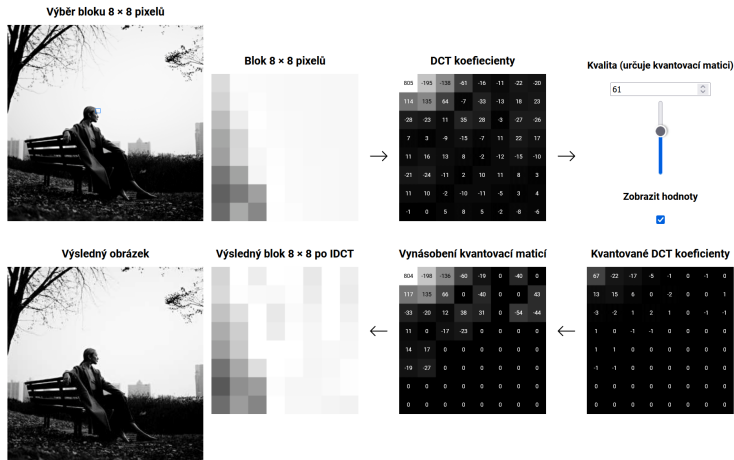


JPEG

Kvalita a kvantita I

- Interaktivní ukázka kódování obrázku:

<https://www.utko.fekt.vut.cz/~rajmic/applets/jpeg/templates/jpeg.html>



JPEG

Kvalita a kvantita II

- Digitální obrázky jsou posloupnosti čísel
- Např. obrázek velikosti 2304×4096 pixelů, každý pixel obsahuje R, G, B hodnoty, každá z nich vyjádřena 8 bity, celkově **27 MB**



JPEG

Kvalita a kvantita II

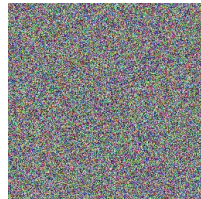
- Digitální obrázky jsou posloupnosti čísel
- Např. obrázek velikosti 2304×4096 pixelů, každý pixel obsahuje R, G, B hodnoty, každá z nich vyjádřena 8 bity, celkově **27 MB**
- JPEG umí kompresní poměr až 25:1 bez postřehnutelné ztráty kvality



JPEG

Kvalita a kvantita II

- Digitální obrázky jsou posloupnosti čísel
- Např. obrázek velikosti 2304×4096 pixelů, každý pixel obsahuje R, G, B hodnoty, každá z nich vyjádřena 8 bity, celkově **27 MB**
- JPEG umí kompresní poměr až 25:1 bez postřehnutelné ztráty kvality
- Jak úspěšná komprese bude ovšem záleží na dvou věcech:
 - na tom, co je na obrázku



- na tom, jak nastavíme parametr „kvality“ Q (0–100 nebo 1–100)
<https://youtu.be/ENJSVrYN6v4>



JPEG

Kvalita a kvantita III



originál 560×519 px



Q = 15, 15 kB



Q = 10, 12 kB



Q = 5, 8 kB



Q = 0, 6 kB

JPEG

Kvalita a kvantita III



originál 560×519 px



Q = 15, 15 kB



Q = 10, 12 kB



Q = 5, 8 kB



Q = 0, 6 kB



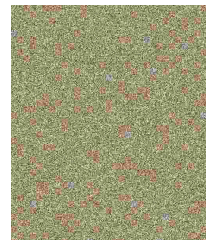
permut. 560×519 px



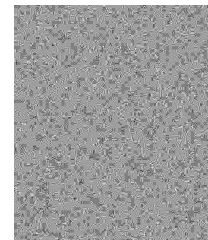
Q = 15, 62 kB



Q = 10, 50 kB

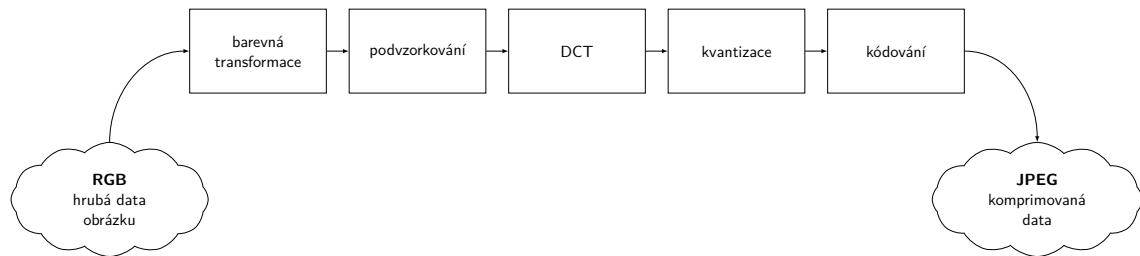


Q = 5, 38 kB

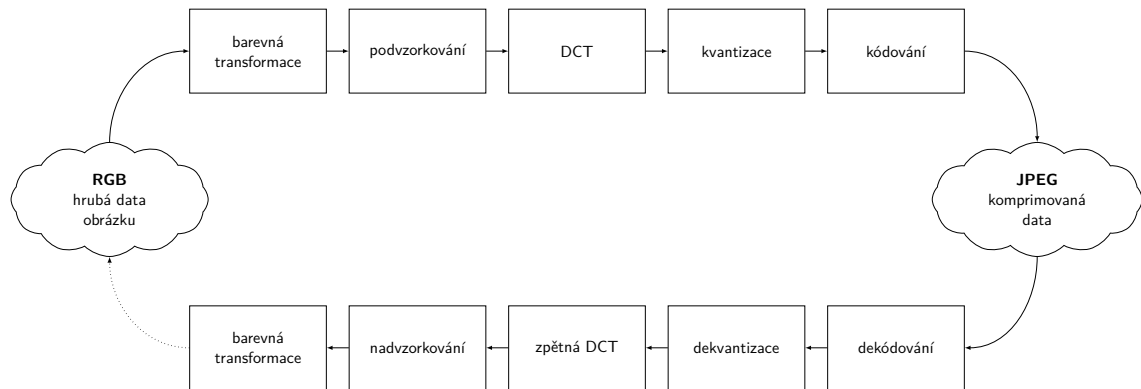


Q = 0, 32 kB

Základní kroky/komponenty kodéru a dekodéru JPEG

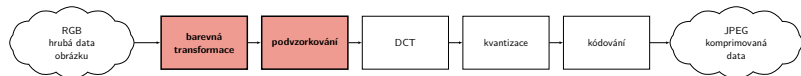


Základní kroky/komponenty kodéru a dekodéru JPEG



Představení komponent JPEG

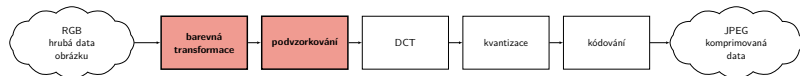
Převod z RGB do YCbCr + podvzorkování



- RGB je barevný model vhodný pro technická zařízení
- Vizuálnímu systému člověka je bližší model YCbCr
 - Y... jas, Cb... „modrost“, Cr... „červenost“
- Mezi RGB a YCbCr funguje lineární bijekce $\mathbb{R}^3 \rightarrow \mathbb{R}^3$

Představení komponent JPEG

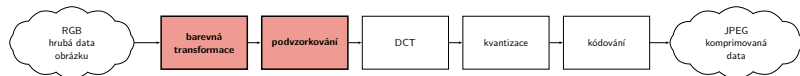
Převod z RGB do YCbCr + podvzorkování



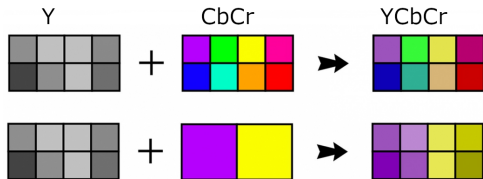
- RGB je barevný model vhodný pro technická zařízení
- Vizualnímu systému člověka je bližší model YCbCr
 - Y... jas, Cb... „modrost“, Cr... „červenost“
- Mezi RGB a YCbCr funguje lineární bijekce $\mathbb{R}^3 \rightarrow \mathbb{R}^3$
- Na změnu jasu je člověk citlivější
 - \Rightarrow možnost chovat se hruběji k barvosným složkám (v RGB nelze aplikovat)
 - Po převedení RGB do YCbCr se složky Cb, Cr decimují po blocích 2×2 (schéma 4:2:0)

Představení komponent JPEG

Převod z RGB do YCbCr + podvzorkování



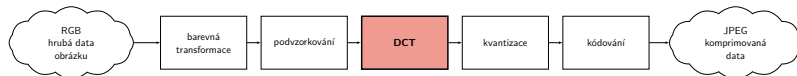
- RGB je barevný model vhodný pro technická zařízení
- Vizuálnímu systém člověka je bližší model YCbCr
 - Y...jas, Cb...„modrost“, Cr...„červenost“
- Mezi RGB a YCbCr funguje lineární bijekce $\mathbb{R}^3 \rightarrow \mathbb{R}^3$
- Na změnu jasu je člověk citlivější
 - \Rightarrow možnost chovat se hruběji k barvosným složkám (v RGB nelze aplikovat)
 - Po převedení RGB do YCbCr se složky Cb, Cr decimují po blocích 2×2 (schéma 4:2:0)



Aplet: <https://www.utko.fekt.vut.cz/~rajmic/applets/jpeg/templates/subsampling.html>

Představení komponent JPEG

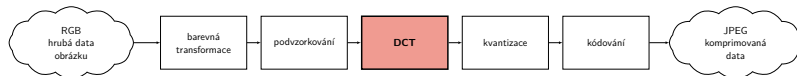
Diskrétní kosinová transformace (DCT)



- DCT (resp. DCT-II) pochází z roku 1972 (N. Ahmed)
- Vznikla se záměrem komprimovat obrazy
- Je podobná Fourierově transformaci (DFT), která také pracuje s harmonickými funkcemi
- Má oproti DFT dvě hlavní výhody:
 - Převádí reálná čísla na reálná
 - Na běžných obrázcích **dosahuje vyšší úrovně komprese (lépe aproximuje)**

Představení komponent JPEG

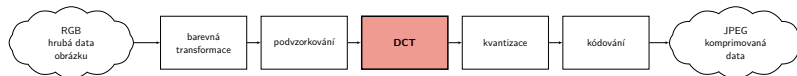
Diskrétní kosinová transformace (DCT)



- DCT (resp. DCT-II) pochází z roku 1972 (N. Ahmed)
- Vznikla se záměrem komprimovat obrazy
- Je podobná Fourierově transformaci (DFT), která také pracuje s harmonickými funkcemi
- Má oproti DFT dvě hlavní výhody:
 - Převádí reálná čísla na reálná
 - Na běžných obrázcích **dosahuje vyšší úrovně komprese (lépe aproximuje)**
- DCT je klíčový krok JPEG

Představení komponent JPEG

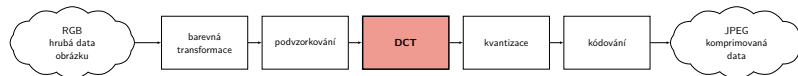
Diskrétní kosinová transformace (DCT)



- Je to opět lineární bijekce, $\mathbb{R}^N \rightarrow \mathbb{R}^N$
- Aplet: <https://www.utko.fekt.vut.cz/~rajmic/applets/dct1d/>

Představení komponent JPEG

Diskrétní kosinová transformace (DCT)



- Je to opět lineární bijekce, $\mathbb{R}^N \rightarrow \mathbb{R}^N$
- Aplet: <https://www.utko.fekt.vut.cz/~rajmic/applets/dct1d/>
- V jednorozměrném případě je vzorec pro „analýzu“ vektoru $[x_1, \dots, x_N]$:

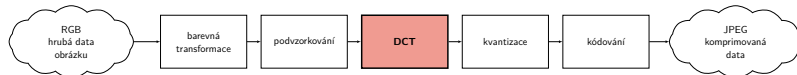
$$X_k = c_k \cdot \sum_{n=0}^{N-1} x_n \cdot \cos\left(\frac{\pi(n+1/2)k}{N}\right), \quad k = 1, \dots, N$$

- V „syntéze“ vyjadřujeme vektor $[x_1, \dots, x_N]$ jako lineární kombinaci báзовých kosinových kmitů o různých kmitočtech:

$$x_n = d_k \cdot \sum_{k=0}^{N-1} X_k \cdot \cos\left(\frac{\pi(k+1/2)n}{N}\right), \quad n = 1, \dots, N$$

Představení komponent JPEG

Diskrétní kosinová transformace (DCT)



- Je to opět lineární bijekce, $\mathbb{R}^N \rightarrow \mathbb{R}^N$
- Aplet: <https://www.utko.fekt.vut.cz/~rajmic/applets/dct1d/>
- V jednorozměrném případě je vzorec pro „analýzu“ vektoru $[x_1, \dots, x_N]$:

$$X_k = c_k \cdot \sum_{n=0}^{N-1} x_n \cdot \cos\left(\frac{\pi(n+1/2)k}{N}\right), \quad k = 1, \dots, N$$

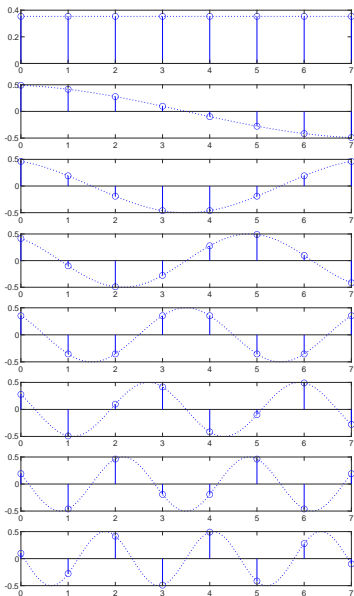
- V „syntéze“ vyjadřujeme vektor $[x_1, \dots, x_N]$ jako lineární kombinaci báзовých kosinových kmitů o různých kmitočtech:

$$x_n = d_k \cdot \sum_{k=0}^{N-1} X_k \cdot \cos\left(\frac{\pi(k+1/2)n}{N}\right), \quad n = 1, \dots, N$$

- Jsme schopni takto vyjádřit **jakýkoliv** vektor!
- Unitarita DCT

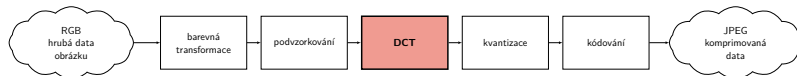
Představení komponent JPEG

Diskrétní kosinová transformace (DCT) – ukázka báze \mathbb{R}^8

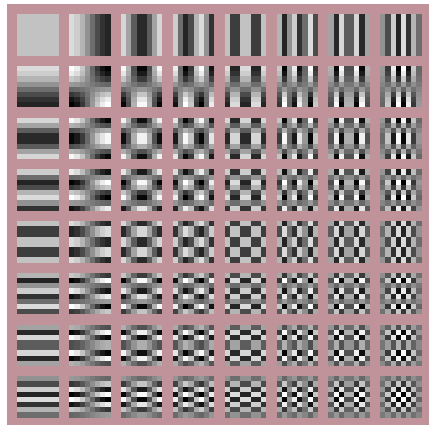


Představení komponent JPEG

Diskrétní kosinová transformace (DCT)



- V JPEG se používá dvojrozměrná DCT
- Ta je separabilní
- DCT se počítá **zvlášť** pro bloky 8×8 pixelů
 - Tedy DCT je zde $\mathbb{R}^{64} \rightarrow \mathbb{R}^{64}$
 - Takové DCT odpovídá báze prostoru \mathbb{R}^{64} , viz obr.
 - Ovšem Cb, Cr jsou podvzorkované 4:2:0
Takže kanály Cb, Cr ve skutečnosti odpovídají blokům 16×16 pixelů



Představení komponent JPEG

Diskrétní kosinová transformace (DCT) – proč zrovna DCT?

- Po transformaci je informace mnohem méně rozprostřená; naopak je **koncentrovanější**, což umožní **efektivní kompresi**
- Navíc typicky **DCT dosahuje vyšší úrovně komprese (lépe aproximuje) než DFT**

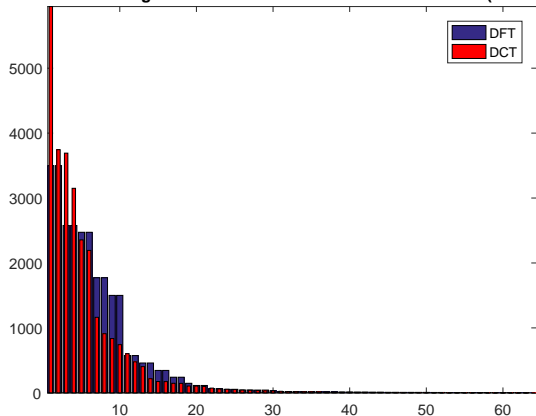
Představení komponent JPEG

Diskrétní kosinová transformace (DCT) – proč zrovna DCT?

- Po transformaci je informace mnohem méně rozprostřená; naopak je **koncentrovanější**, což umožní **efektivní kompresi**
- Navíc typicky **DCT dosahuje vyšší úrovně komprese (lépe aproximuje) než DFT**

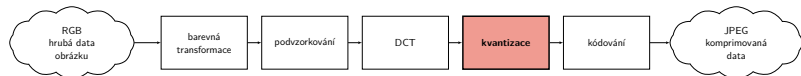


Koncentrace energie obrazu v transformacních koeficientech (mimo DC)



Představení komponent JPEG

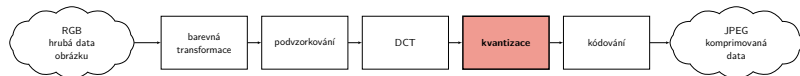
Kvantizace I



- z DCT vypadávají reálná čísla
- $\text{round}(x)$... zaokrouhlování (tj. kvantizace na celá čísla)
Příklad: $\text{round}(38,85) = 39$, $\text{round}(-0,412) = 0$

Představení komponent JPEG

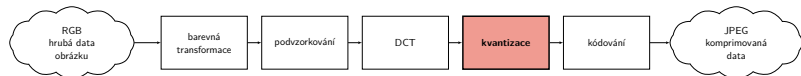
Kvantizace I



- z DCT vypadávají reálná čísla
- $\text{round}(x)$... zaokrouhlování (tj. kvantizace na celá čísla)
Příklad: $\text{round}(38,85) = 39$, $\text{round}(-0,412) = 0$
- $\text{round}(a) = 39$; $a = ?$

Představení komponent JPEG

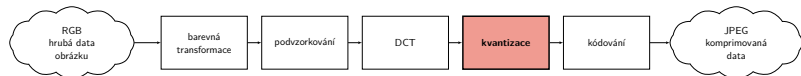
Kvantizace I



- z DCT vypadávají reálná čísla
- $\text{round}(x)$... zaokrouhlování (tj. kvantizace na celá čísla)
Příklad: $\text{round}(38,85) = 39$, $\text{round}(-0,412) = 0$
- $\text{round}(a) = 39$; $a = ?$
- $\text{round}(x/w) \cdot w$... kvantizace na w -násobky celých čísel
Příklad: $\text{round}(38,85/4) \cdot 4 = \text{round}(9,7125) \cdot 4 = 10 \cdot 4 = 40$
Speciální případ $w = 1$
- Čím větší w , tím větší šance, že **zaokrouhlíme na nulu**

Představení komponent JPEG

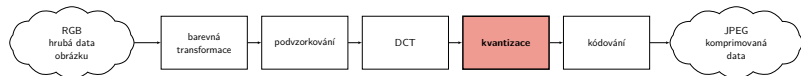
Kvantizace I



- z DCT vypadávají reálná čísla
- $\text{round}(x)$... zaokrouhlování (tj. kvantizace na celá čísla)
Příklad: $\text{round}(38,85) = 39$, $\text{round}(-0,412) = 0$
- $\text{round}(a) = 39$; $a = ?$
- $\text{round}(x/w) \cdot w$... kvantizace na w -násobky celých čísel
Příklad: $\text{round}(38,85/4) \cdot 4 = \text{round}(9,7125) \cdot 4 = 10 \cdot 4 = 40$
Speciální případ $w = 1$
- Čím větší w , tím větší šance, že **zaokrouhlíme na nulu**
- Kodér dělá: $\hat{x} = \text{round}(x/w)$
Dekodér dělá: $\hat{x} \cdot w$

Představení komponent JPEG

Kvantizace II



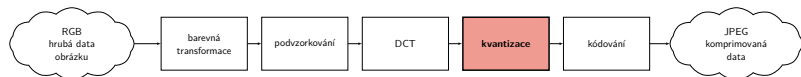
- V JPEG z DCT každého bloku padá 8×8 matice reálných čísel, X

- Kodér dělá: $\hat{x}_{ij} = \text{round}(x_{ij}/w_{ij})$

Dekodér dělá: $\hat{x}_{ij} \cdot w_{ij}$ pro všechna i, j

Představení komponent JPEG

Kvantizace II



- V JPEG z DCT každého bloku padá 8×8 matice reálných čísel, X

- Kodér dělá: $\hat{x}_{ij} = \text{round}(x_{ij}/w_{ij})$

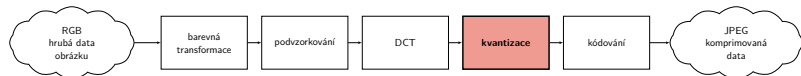
Dekodér dělá: $\hat{x}_{ij} \cdot w_{ij}$ pro všechna i, j

- Tzv. kvantizační tabulka je matice W složená z w_{ij}
- Pro kvalitu $Q = 50$ standard specifikuje kvantizační tabulku jasové složky takto:

$$W(50) = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Představení komponent JPEG

Kvantizace II



- V JPEG z DCT každého bloku padá 8×8 matice reálných čísel, X

- Kodér dělá: $\hat{x}_{ij} = \text{round}(x_{ij}/w_{ij})$

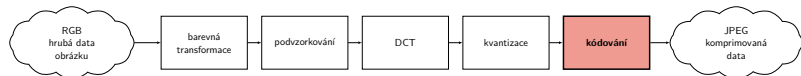
Dekodér dělá: $\hat{x}_{ij} \cdot w_{ij}$ pro všechna i, j

- Tzv. kvantizační tabulka je matice W složená z w_{ij}
- Pro kvalitu $Q = 50$ standard specifikuje kvantizační tabulku jasové složky
- Kvantizovaná matice DCT koeficientů může vypadat takto (nízké Q):

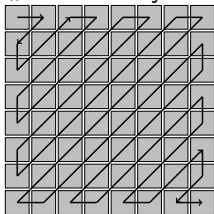
$$\hat{X} = \begin{bmatrix} 12 & -5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 3 & 1 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & -4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Představení komponent JPEG

Kódování délkou běhu (RLE)



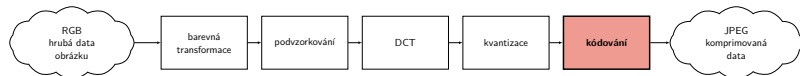
- V 8×8 matici \hat{X} celých čísel jsou pozice řazeny podle frekvence: nízké frekvence vlevo nahoře, vysoké vpravo dole
- Fotografie typicky obsahují málo vysokých frekvencí
- V kombinaci s kvantizací to znamená, že směrem dolů doprava roste výskyt nul (aplet) „Cik-cak“ vyčítání řadí nuly k sobě, tj. do řady za sebe:



- Jak efektivně zakódovat posloupnost
12, -5, 3, 3, 3, 0, 0, 1, 0, 0, 0, 0, -4, 0, 0, 0, 0, 0, 0, 0, -2, 1, 0, 0, 0, 0, 0, ... ? **RLE!**

Představení komponent JPEG

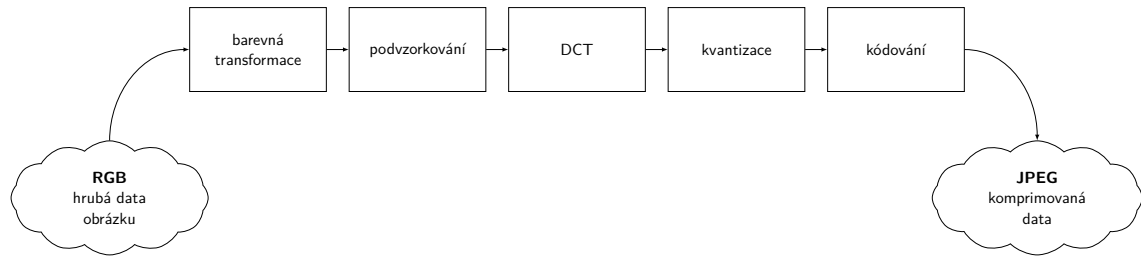
Huffmanovo kódování



- Huffmanovo kódování (1952) úsporně ukládá posloupnost symbolů do binární podoby
- Příklad (obecný):
 - V posloupnosti EERDBEEREERDEBEBRE jsou symboly podle četnosti: E, R, B, D
 - Huffmanovým algoritmem je symbolům přiděleno bitové vyjádření:
 - E: 1
 - R: 01
 - B: 001
 - D: 000
 - Zpráva pak je 1|1|01|000|001|1|1|01|1|1|01|000|1|001|1|001|01|1
- Prefixový kód
- JPEG: Huffmanova tabulka pro 162 různých kombinací, které mohou vzejít z RLE
- JPEG: Univerzální tabulka vs. tabulka na míru obrázku

Aplet: https://www.utko.fekt.vut.cz/~rajmic/applets/huffman_kod/

Shrnutí kroků



<https://www.utko.fekt.vut.cz/~rajmic/applets/jpeg/templates/jpeg.html>

JPEG artefakty



- Viditelné bloky

JPEG artefakty



- Viditelné bloky
- Posterizace

JPEG artefakty



- Viditelné bloky
- Posterizace
- Ztráta barevnosti

JPEG artefakty

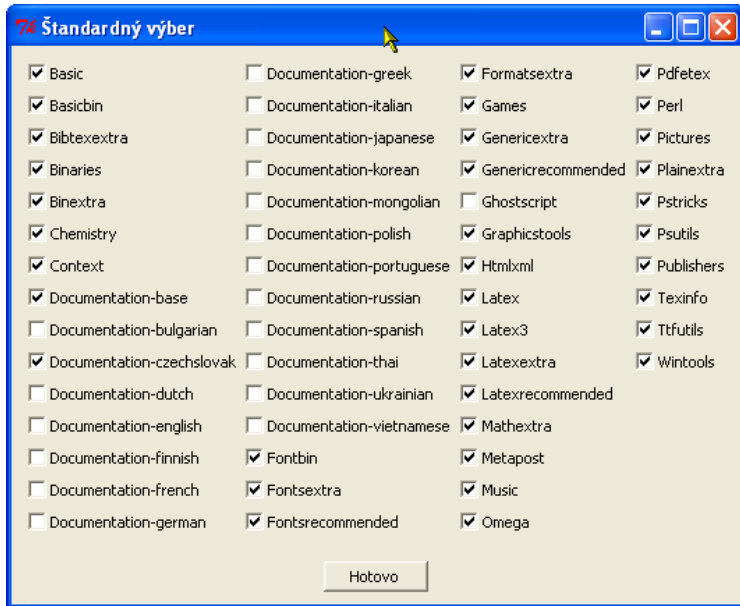


- Viditelné bloky
- Posterizace
- Ztráta barevnosti
- „Smetí“ okolo ostrých hran

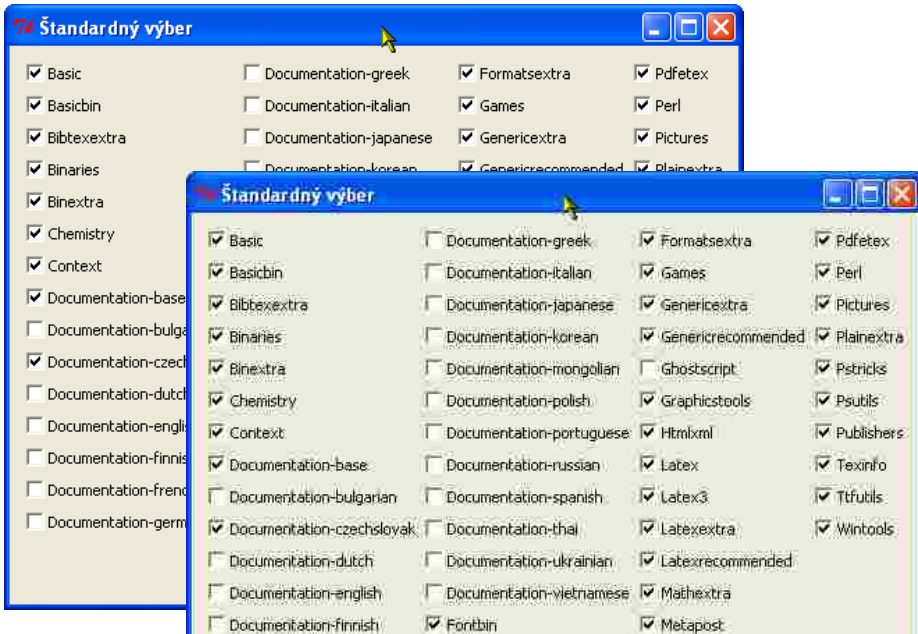


- Viditelné bloky
- Posterizace
- Ztráta barevnosti
- „Smetí“ okolo ostrých hran
- Případně pruhy

JPEG není univerzální: PNG8 vs. JPEG (oba 19 kB)



JPEG není univerzální: PNG8 vs. JPEG (oba 19 kB)



Potlačování JPEG artefaktů

Data z kodéru použijeme jinak než to dělá standardní dekodér.

Stojí nás to výpočetní čas vyšší než běžný JPEG dekodér.

- Rozmazání obrázku (dolní propust; klouzavý průměr)
- Optimalizační úloha (M. Šorel & M. Bartoš, ÚTIA AVČR), např.

$$\arg \min_Y p(Y) \quad \text{vzhledem k} \quad \text{round}(\text{DCT}(Y)/W) = \hat{X} \quad \text{a} \quad 0 \leq y_{ij} \leq 255$$

- Neuronové sítě

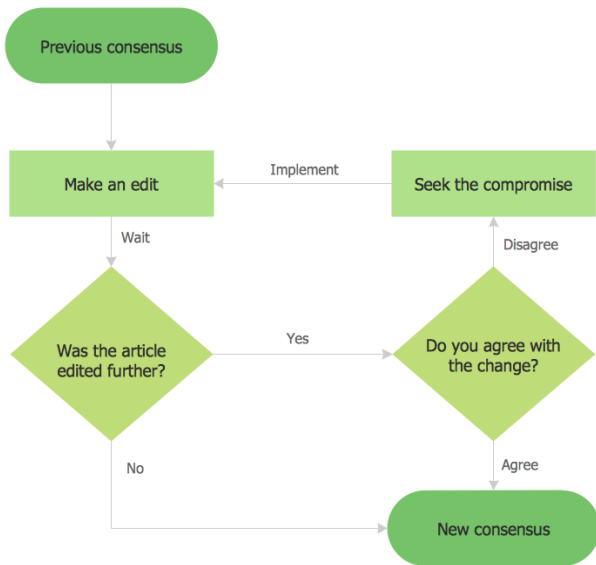
- Přímočarý přístup:

- 1 učení na datasetu „originál \longleftrightarrow JPEG“
- 2 inference

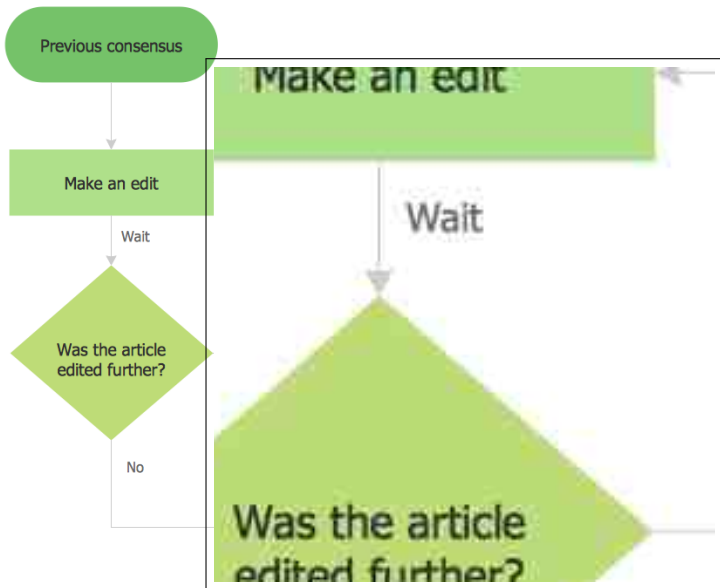
- Diferencovatelná aproximace JPEG řetězce (M. Elad, Technion)
(ale nemají perceptuální ztrátovou funkci)
- Deep image prior (bez učení)

https://dmitryulyanov.github.io/deep_image_prior

Příklad artefaktů, kde by se hodila totální variace $p(Y) = \|\nabla Y\|_1$



Příklad artefaktů, kde by se hodila totální variace $p(Y) = \|\nabla Y\|_1$



- Standard stejné skupiny, okolo r. 2000
- Založený na vlnkové (waveletové) transformaci místo DCT
 - transformace je parametrizovatelná
 - biortogonální... a mnoho dalších hezkých vlastností
 - především ale vysoká schopnost komprese

JPEG2000

- Standard stejné skupiny, okolo r. 2000
- Založený na vlnkové (waveletové) transformaci místo DCT
 - transformace je parametrizovatelná
 - biortogonální . . . a mnoho dalších hezkých vlastností
 - především ale vysoká schopnost komprese



originál

JPEG2000

- Standard stejné skupiny, okolo r. 2000
- Založený na vlnkové (waveletové) transformaci místo DCT
 - transformace je parametrizovatelná
 - biortogonální... a mnoho dalších hezkých vlastností
 - především ale vysoká schopnost komprese



originál



JPEG, 6 kB

JPEG2000

- Standard stejné skupiny, okolo r. 2000
- Založený na vlnkové (waveletové) transformaci místo DCT
 - transformace je parametrizovatelná
 - biortogonální . . . a mnoho dalších hezkých vlastností
 - především ale vysoká schopnost komprese



originál



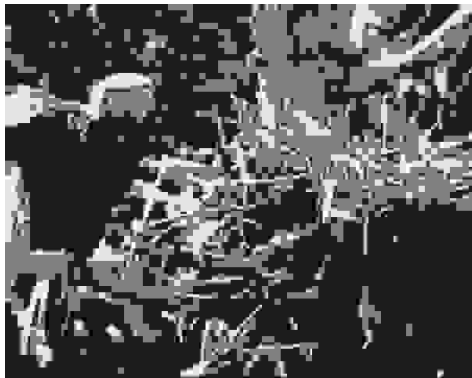
JPEG2000, 6 kB

JPEG2000

- Standard stejné skupiny, okolo r. 2000
- Založený na vlnkové (waveletové) transformaci místo DCT
 - transformace je parametrizovatelná
 - biortogonální ... a mnoho dalších hezkých vlastností
 - především ale vysoká schopnost komprese



originál



JPEG, 751 B

JPEG2000

- Standard stejné skupiny, okolo r. 2000
- Založený na vlnkové (waveletové) transformaci místo DCT
 - transformace je parametrizovatelná
 - biortogonální . . . a mnoho dalších hezkých vlastností
 - především ale vysoká schopnost komprese



originál



JPEG2000, 781 B

JPEG2000

- Standard stejné skupiny, okolo r. 2000
- Založený na vlnkové (waveletové) transformaci místo DCT
 - transformace je parametrizovatelná
 - biortogonální . . . a mnoho dalších hezkých vlastností
 - především ale vysoká schopnost komprese
- V praxi se používá omezeně (proč?!)



originál



JPEG2000, 781 B

- Kvantizační tabulky mohou být proprietární
⇒ forenzní analýza
„Ano, vyfotil jsem to já foťákem Canon 5D Mark II.“

- Kvantizační tabulky mohou být proprietární
⇒ forenzní analýza
„Ano, vyfotil jsem to já foťákem Canon 5D Mark II.“
- Některé kodéry nepoužívají `round()` ale `floor()`
⇒ forenzní analýza

- JPEG je relativně přímočarý algoritmus (de)komprese obrázků
- Funguje díky propojení komponent z různých oblastí lidského bádání
- JPEG s námi patrně ještě zůstane :)
- JPEG je **vhodný** na fotografie
- JPEG je **nevhodný** pro snímky obrazovky, loga, vývojové diagramy apod.

Děkuji za pozornost.

- https://en.wikipedia.org/wiki/Gibbs_phenomenon