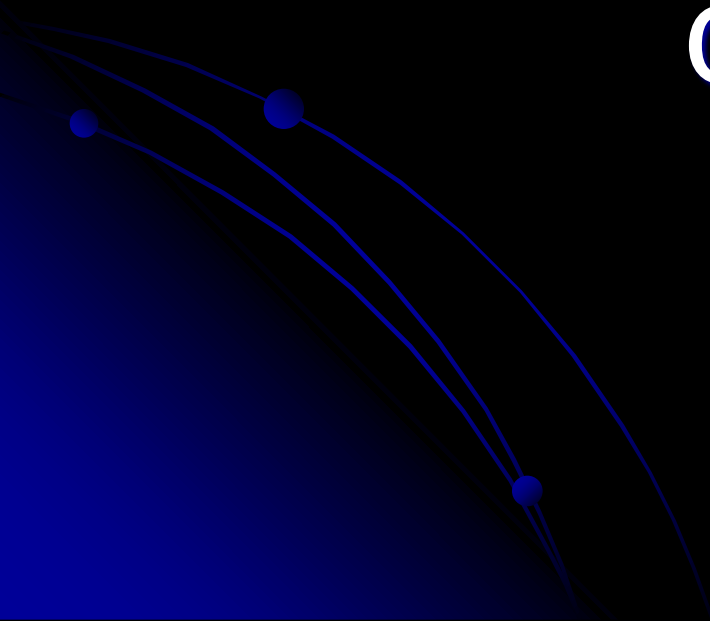


Kvantovanie farieb dithering



Kvantovanie

Obrazovo nezávislé metódy

- najvýznamnejšie bity (~ 2)
- rozdelenie priestoru farieb (partitioning)
- referenčné farby, indexovanie
- prahovanie

Obrazovo závislé metódy

- zhlukovanie priestoru farieb (clustering)
- segmentácia obrazu – rozdelenie obrazu na útvary, ktoré majú rovnaké charakteristiky (farba, textúra, ...)
- prahovanie

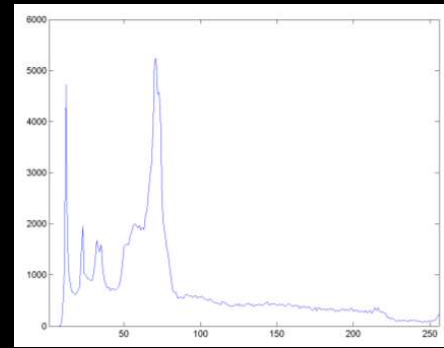
Šedotónové obrazy

Jednoduchý prípad – jeden kanál

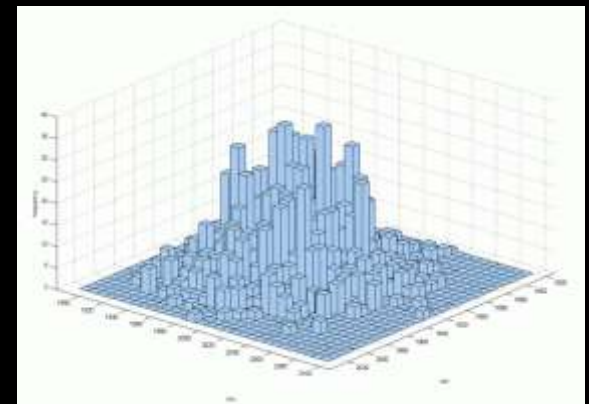
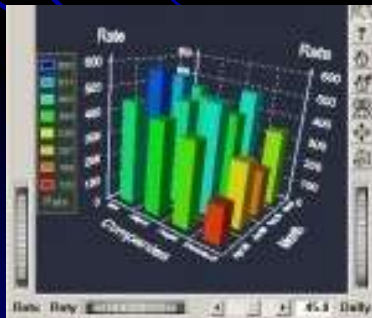
Zmenšenie počtu farieb (úrovní šedej)



Histogram



- Každéj farbe $R_i G_j B_k$ priradí počet obrazových bodov s touto farbou
- Pre každú farbu 256 odtieňov – 256^3 pamäť
- Veľkosť obrazu oveľa menšia ako veľkosť histogramu
- Väčšina binov histogramu prázdna



Histogram - variácie

Xiang

Iba RG pole a list s hodnotami B a frekvenciu výskytu

Zložitosť $O(S/2)$ kde S je priemerná veľkosť B listov

Balasubramanian

Vylepšenie B list -> binárny strom

Zložitosť $O(\log(S))$

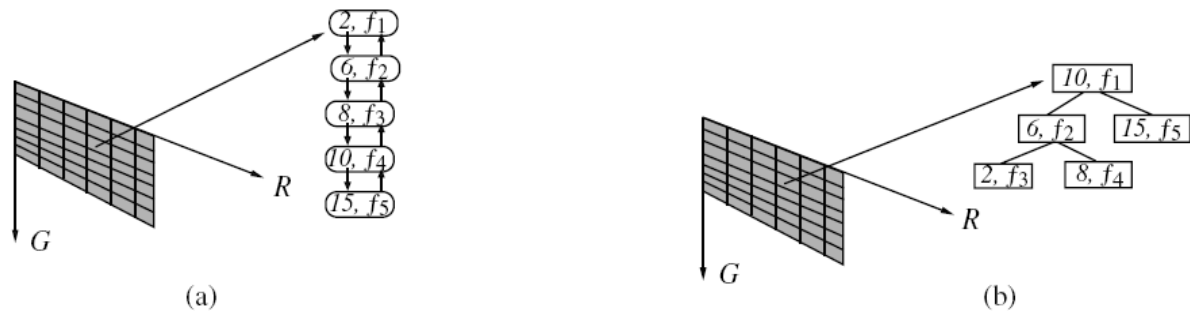
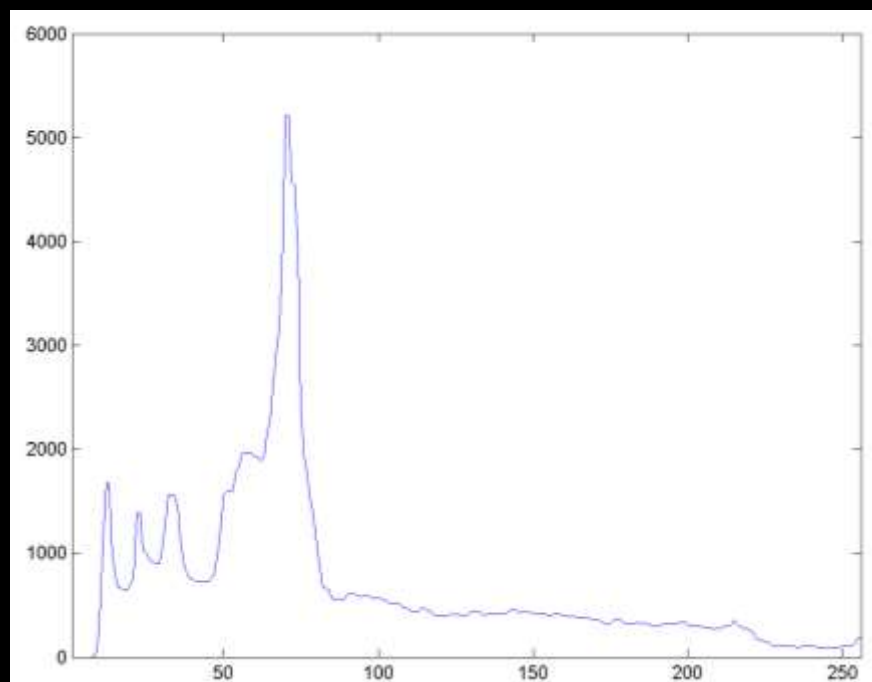
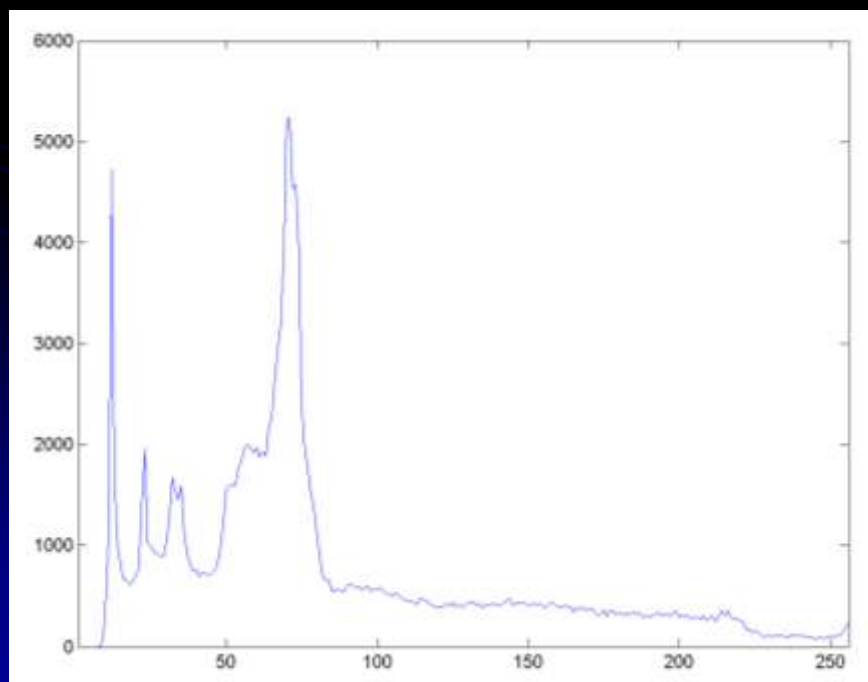


Figure 9.3 The histograms of (a) Xiang⁷⁷ and (b) Balasubramanian.¹⁰

Histogram

každéj úrovni jasů priradí zodpovedajúcu početnosť v obraze

vyhladenie histogramu



Prahovanie histogramu

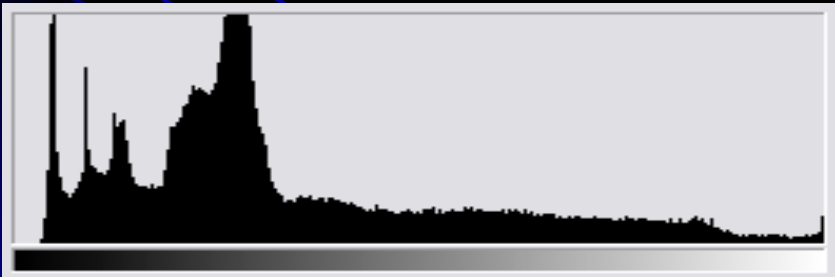
prahová (hraničná) hodnota: $p(i)$, $i=0,..n$

reprezentatívna farba: $f(i)$, $i=1,..n$ - voľba

\forall body obrazu s intenzitou $I(x,y)$

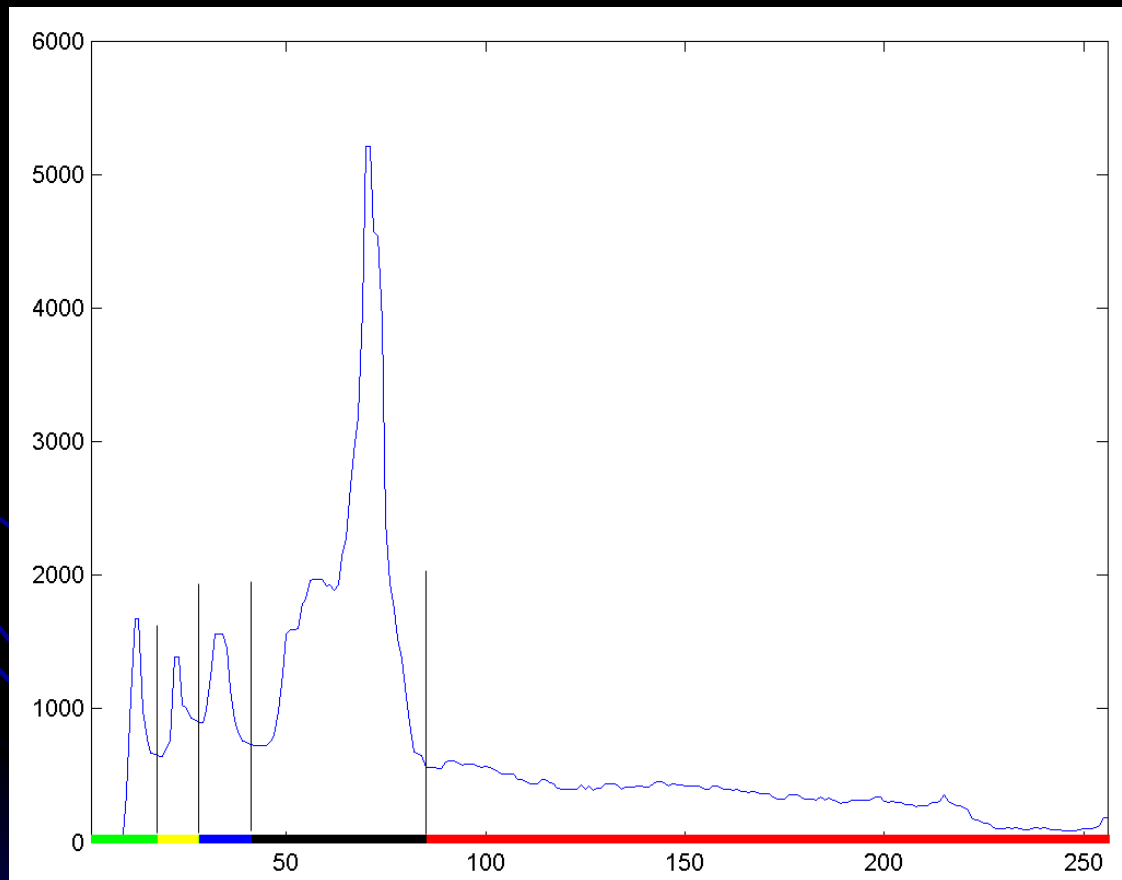
$\forall i=1,..n$

ak $p(i-1) < I(x,y) \leq p(i)$ tak $I(x,y) = f(i)$

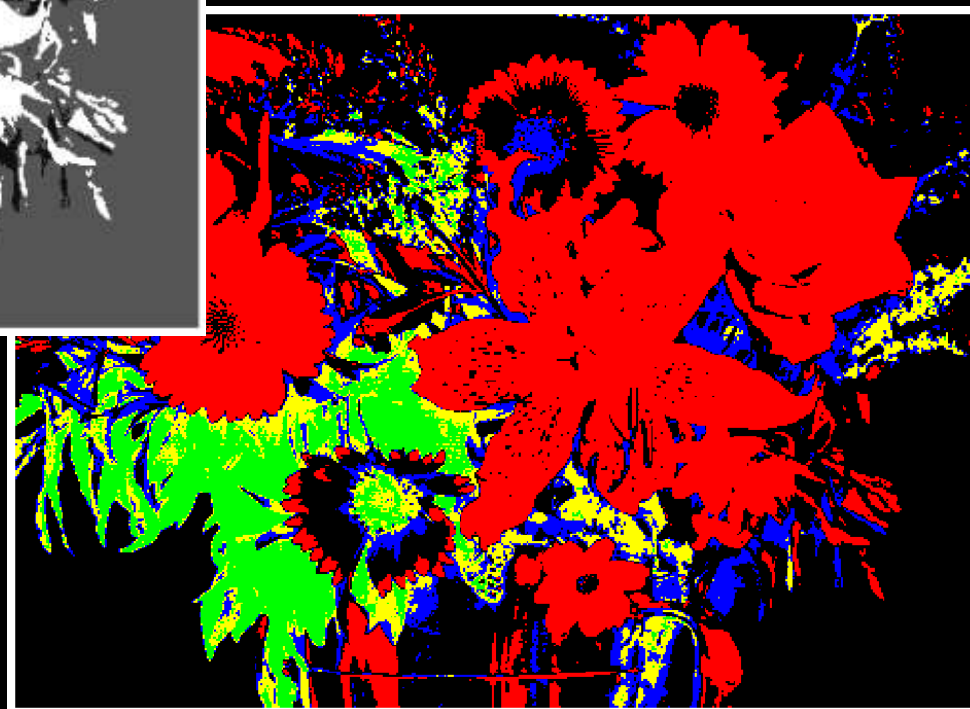
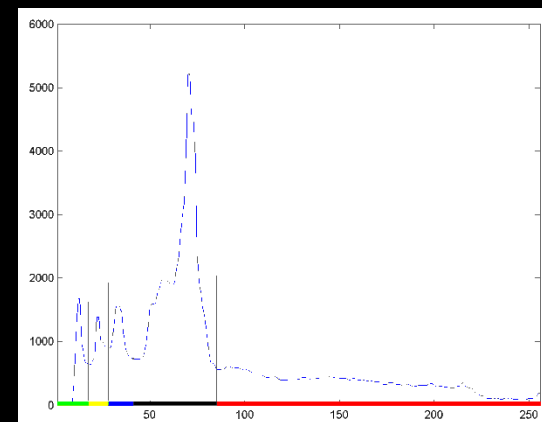


Prahovanie

adaptívne – hľadáme lokálne minimá

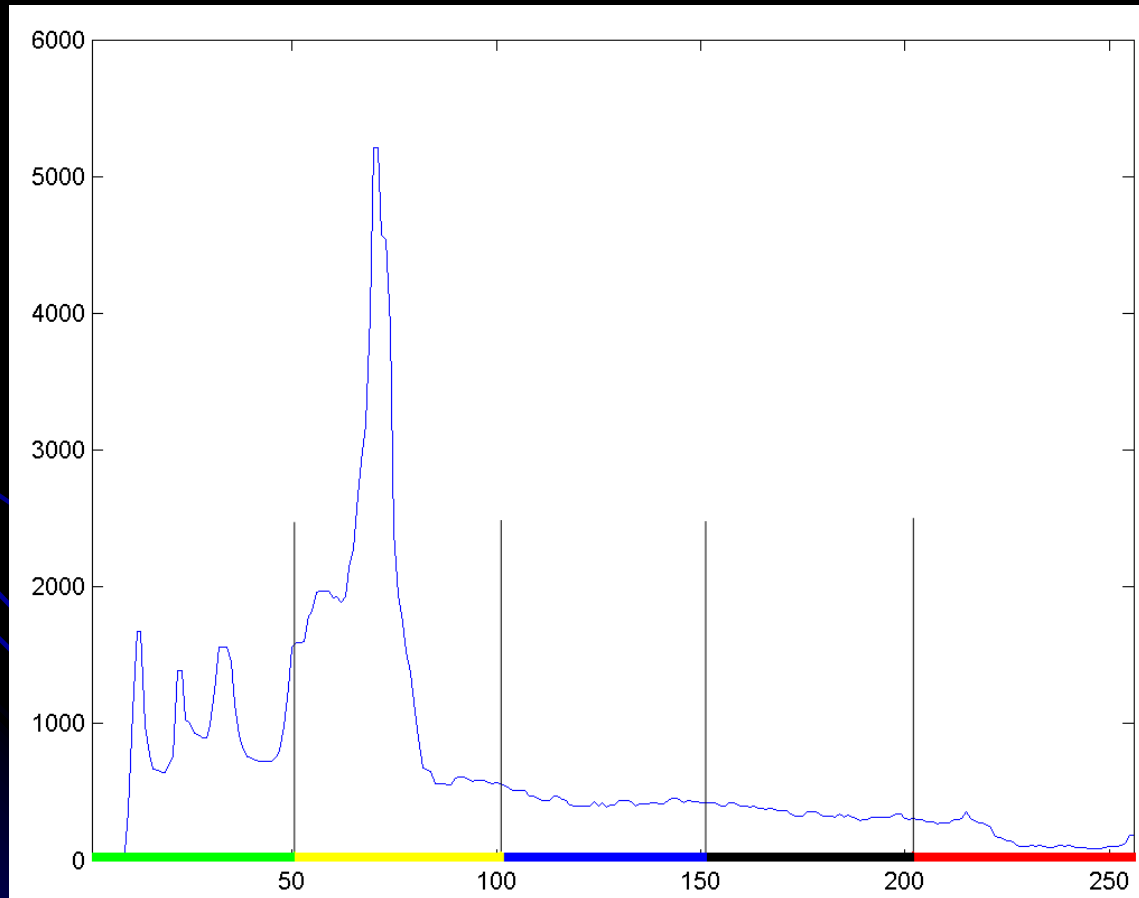


Výsledok

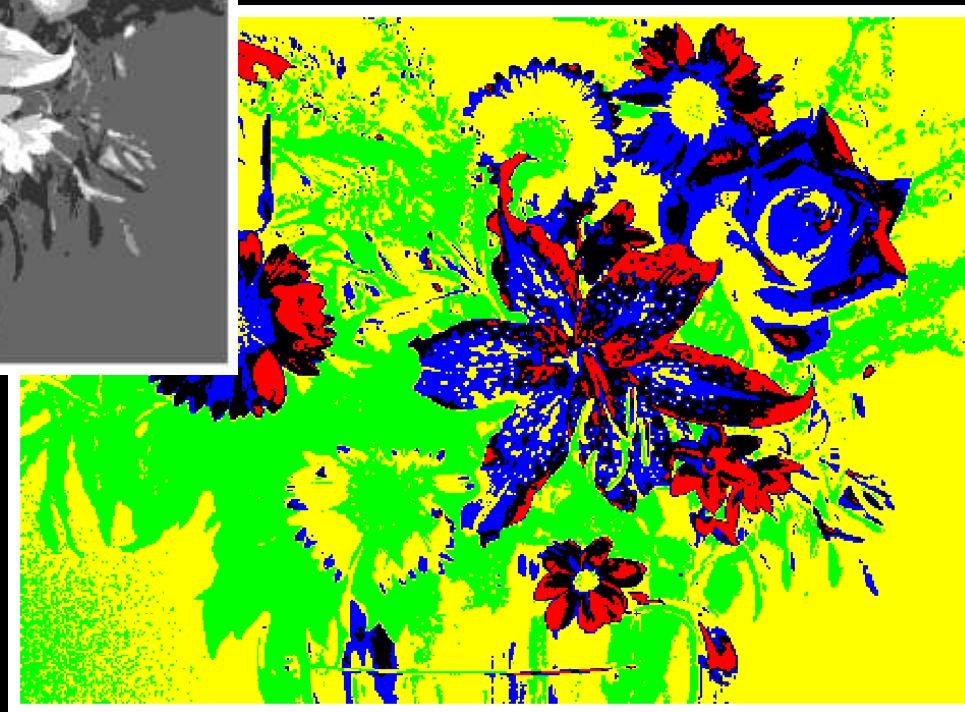
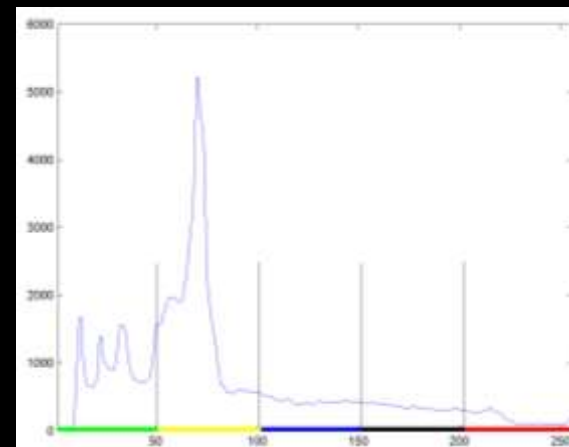


Prahovanie

pevný prah – pravidelné intervaly



Výsledek



Porovnanie

adaptívne vs. pevné



účel použitia

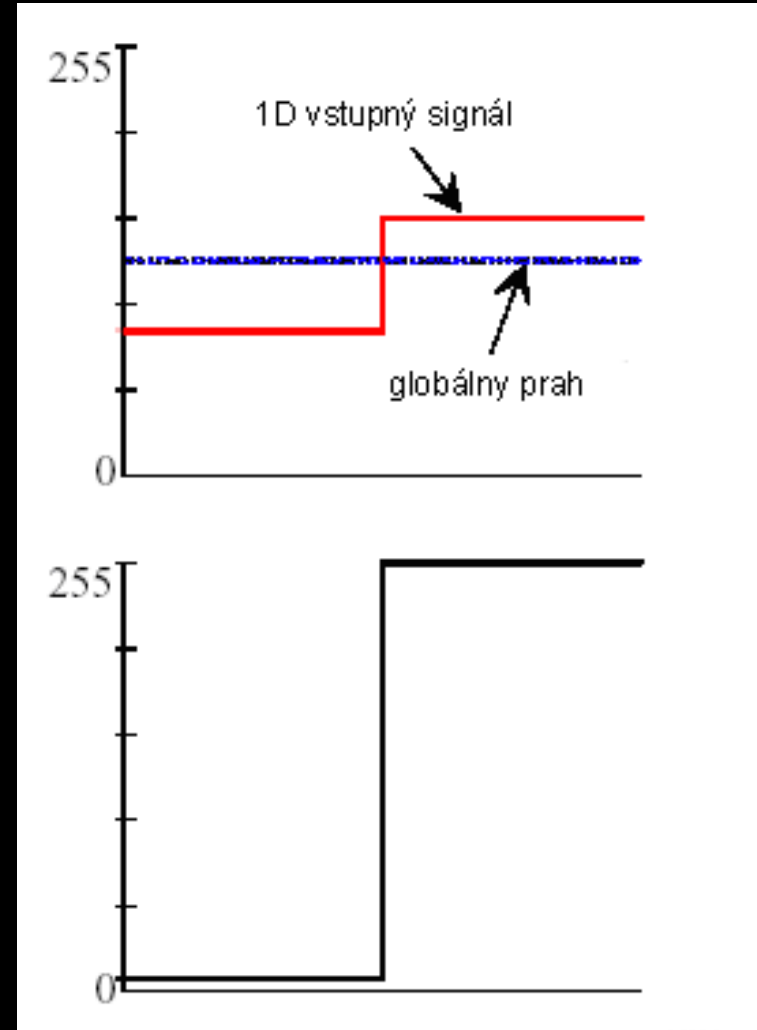
Binarizácia

2 farby (biela, čierna)

triviálne pomocou globálneho prahovania

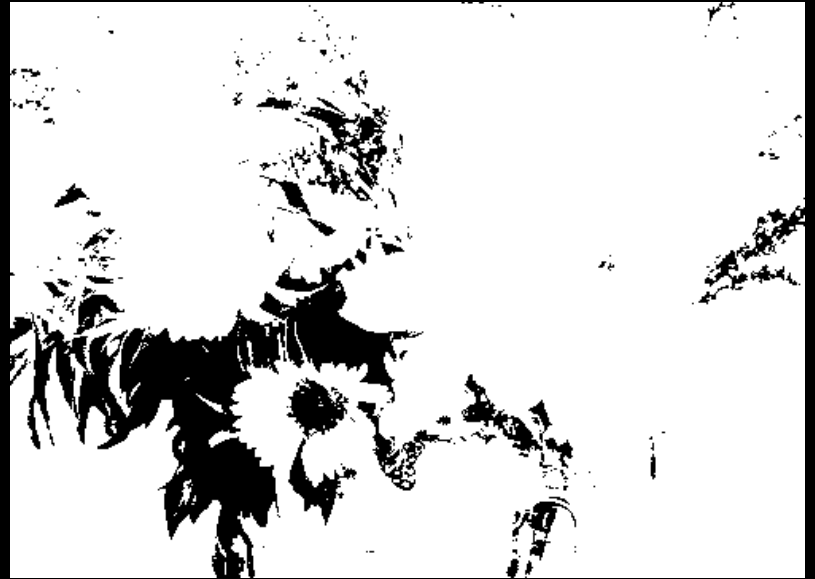
stráca sa nám informácia

naivný algoritmus
prah = 1/2





Problém ?



Vylepšenie

prah taký, aby sa (približne) zachovala
priemerná intenzita

Intenzita

originál : 0.3297

prah 0.5 : 0.2048

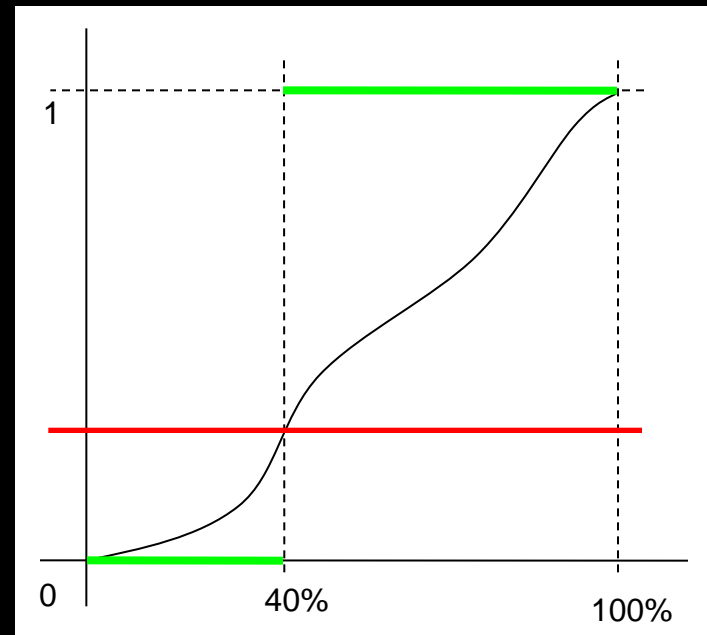


Príklad

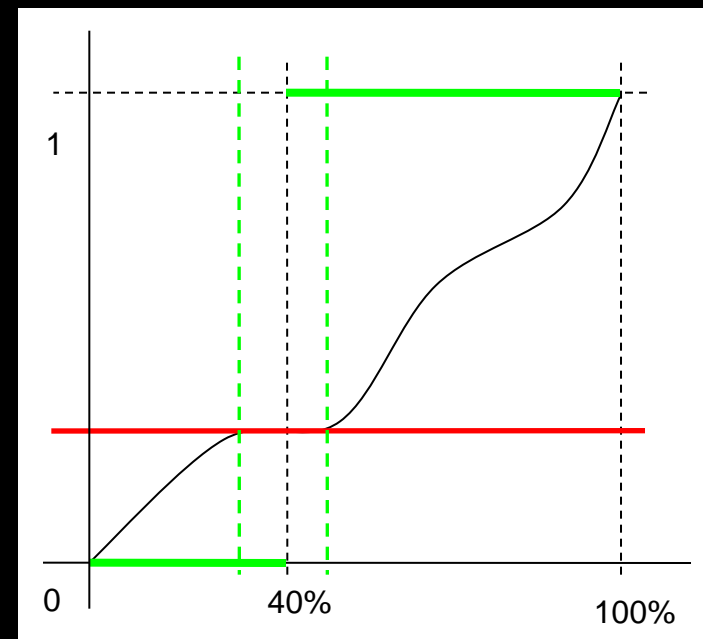
priemerná intenzita = 0.6

prah taký, aby 60% bodov malo vyššiu hodnotu a 40% nižšiu hodnotu

Usporiadame „body“
podľa intenzity
Určíme prah



Nie vždy zachováme intenzitu presne





originál : 0.3297



prah 0.5 : 0.2048

prah 0.3216

intenzita 0.3326



Náhodná modulácia

Intenzita 0.3297

Pred prahovaním
pridáme šum



Každému pixlu sa pridá náhodná hodnota
Rovnomerne rozloženie z $[-a, a]$

Náhodná modulácia

Gaussovský šum

Intenzita 0.3297

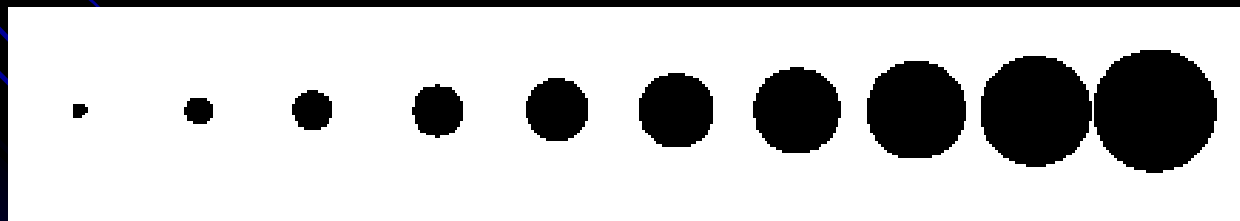


Halftoning

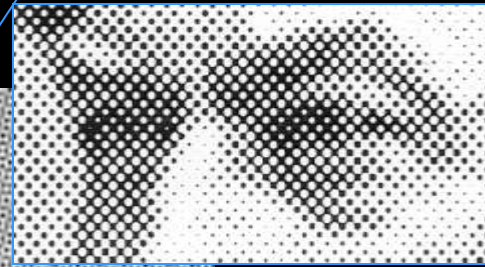
používa sa pri tlači novín

Šedé obrazové body sa reprodukovujú ako rôzne veľké tlačové body.

Čím tmavší obrazový bod, tým väčší tlačový bod



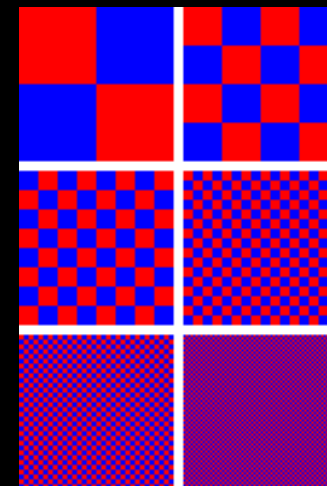
Halftoning



Dithering

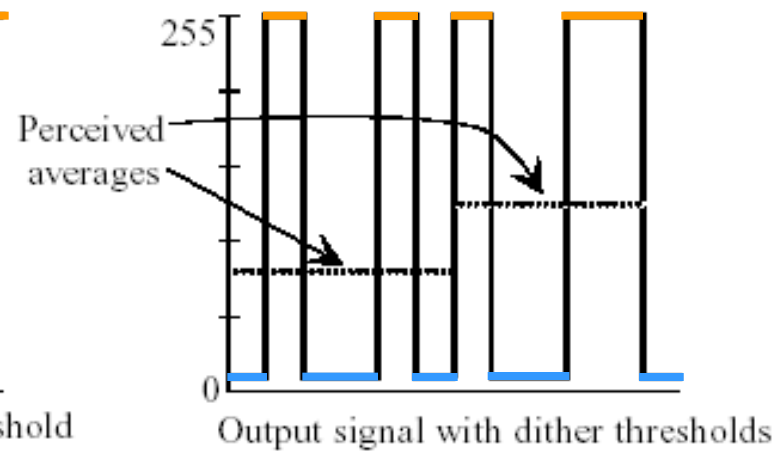
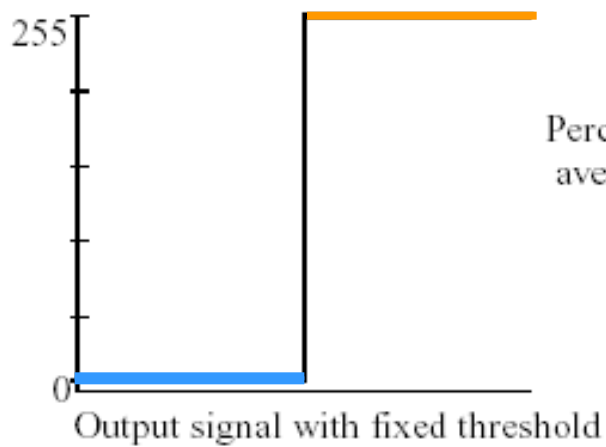
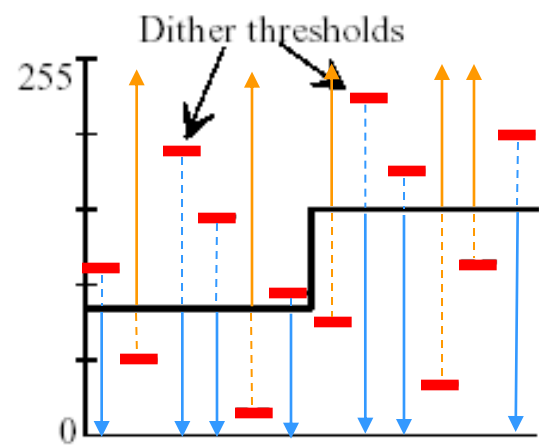
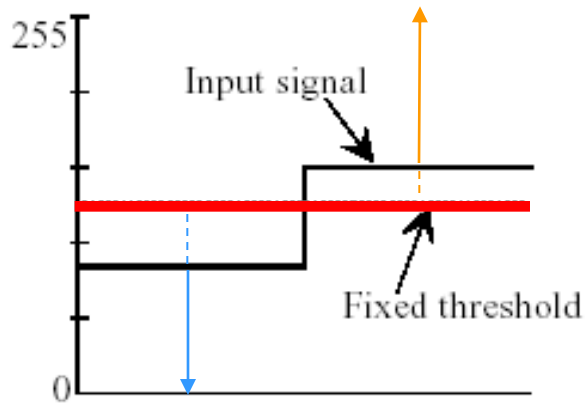
Využíva vlastnosť oka priemerovať body v malom okolí

Dither coding (alebo *dithering*) – zmena šedotónového obrazu na binárny, pri zachovaní priemernej intenzity v oblastiach obrazu



Zmenšovaním veľkosti červeného a modrého obrazového bodu vytvorím fialovú farbu

Dithering



Dithering

Najrozšírenejšie metódy

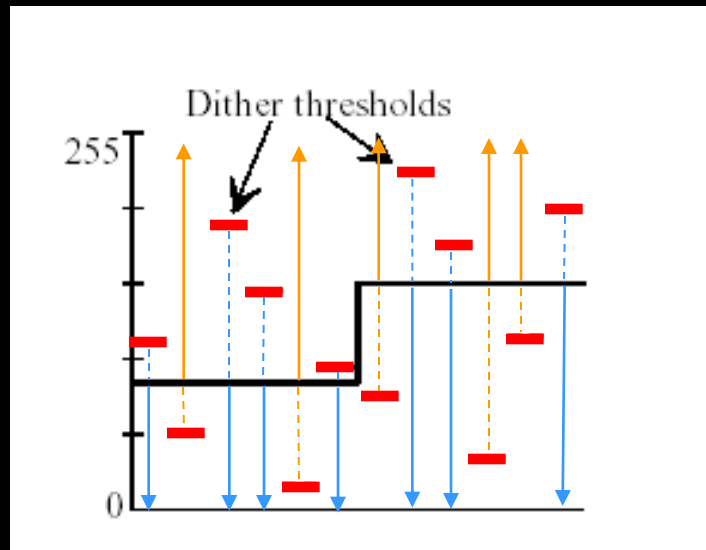
ordered dithering

error diffusion



Ordered dithering

- Aplikovanie **prahovej mapy** na obrazové body, čo spôsobuje že niektoré body sú zobrazované inou farbou



Ordered dithering

- obraz rozdelíme na bloky veľkosti $n \times n$
- každý blok sa spracuje samostatne
- každý pixel bloku sa porovná s určeným prahom
- prahové hodnoty sú dané generátorom pseudonáhodných čísel

$$th_{i,j} = \frac{x_{\max} + 1}{n^2} \cdot (k_{i,j} + 0.5)$$

n – veľkosť matice

$k_{i,j}$ – zodpovedajúca pseudonáhodná hodnota daná maticou

x_{\max} – maximálna intenzita

Ordered dithering – maticice

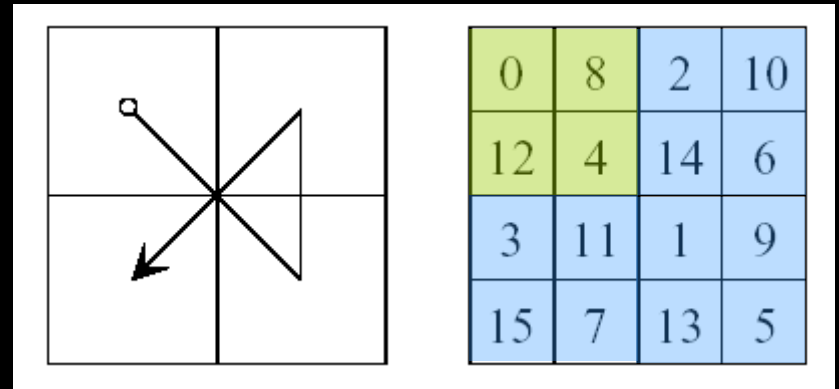
Ako vyrobiť maticu?

$$D_2 = \begin{bmatrix} 0 & 2 \\ 3 & 1 \end{bmatrix}$$

$$D_n = \begin{bmatrix} 4D_{n/2} + D_{00}I^{n/2} & 4D_{n/2} + D_{01}I^{n/2} \\ 4D_{n/2} + D_{10}I^{n/2} & 4D_{n/2} + D_{11}I^{n/2} \end{bmatrix}$$

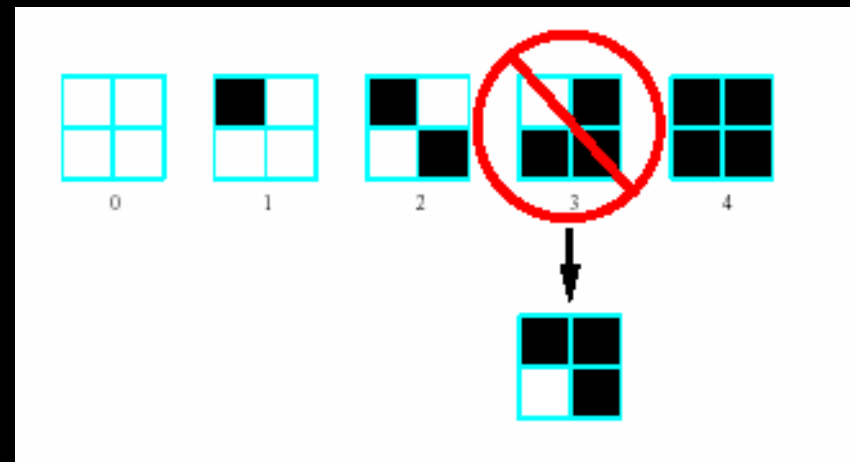
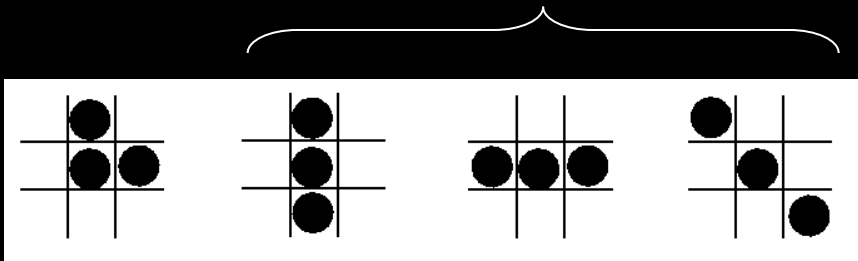
$$D_3 = \begin{bmatrix} 7 & 2 & 6 \\ 4 & 0 & 1 \\ 3 & 8 & 5 \end{bmatrix}$$

D_{ij} - i, j element matice D_2
 I^n - $n \times n$ jednotková matica

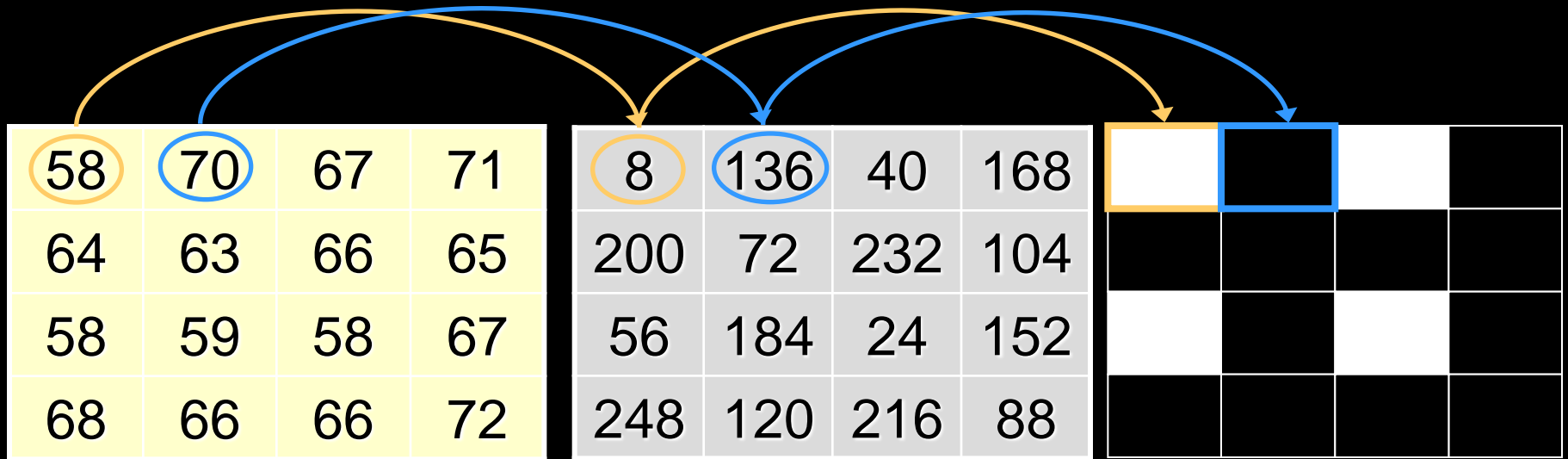


Malice

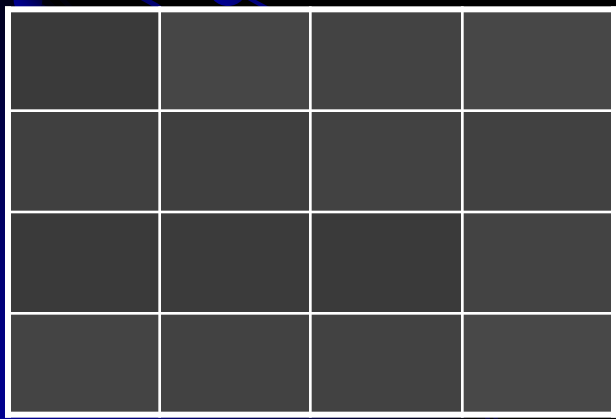
treba vyberať starostlivo,
aby nevytvárali neželané efekty



Ordered dithering – příklad



Vstupný obrázok



Zodpovedajúce prahy

$$th_{i,j} = \frac{x_{\max} + 1}{n^2} \cdot (k_{i,j} + 0.5)$$

Výsledný obrázok

Vstupná intenzita: 1038 (64.875)

Výstupná intenzita: 1020 (63.75)

blok, kde všetky intenzity sú v intervale 56 – 72:
výsledný obraz: 4 biele a 12 čiernych pixlov

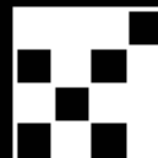
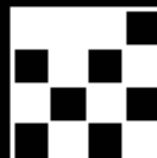
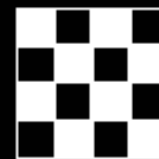
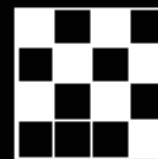
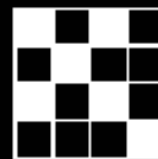
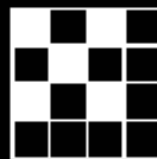
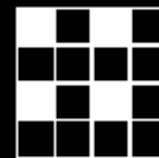
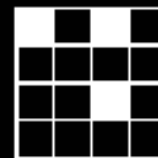
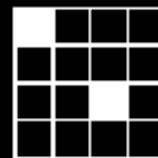
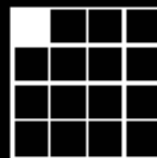
intenzity okolo

16

32

48

64







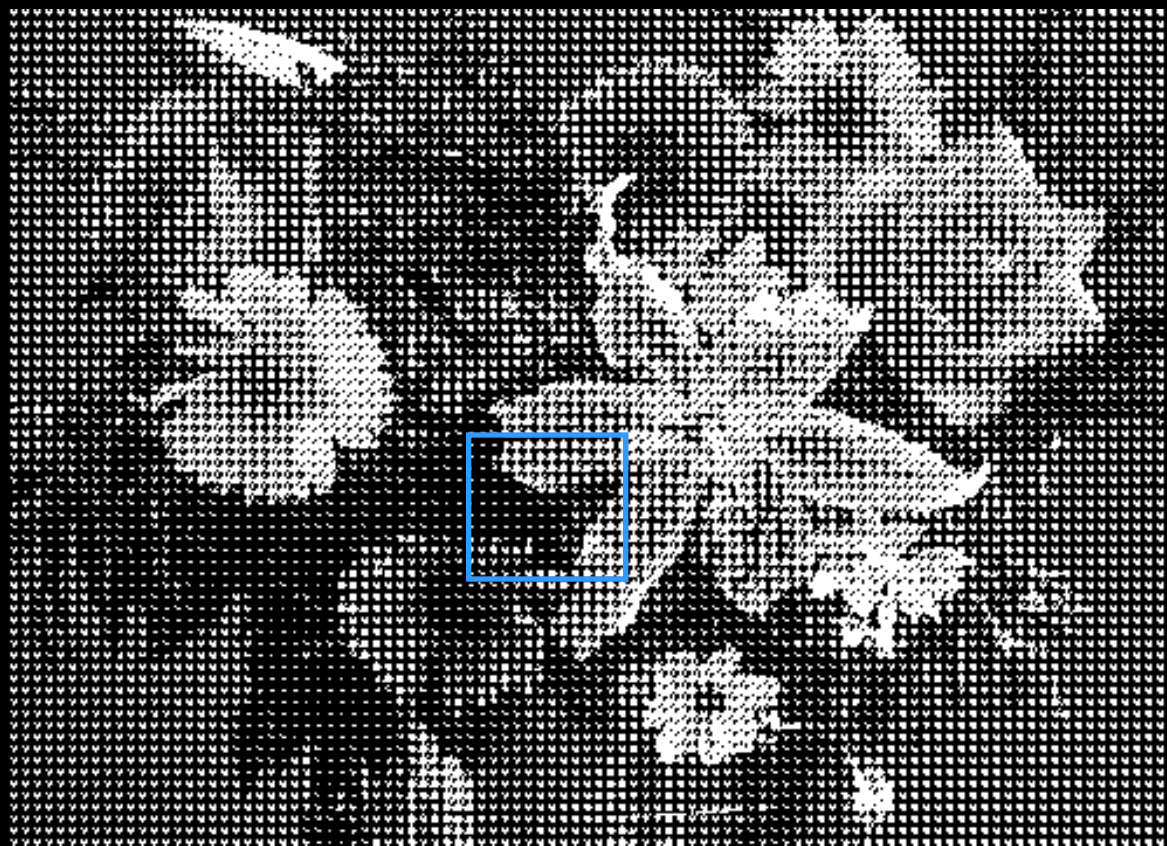
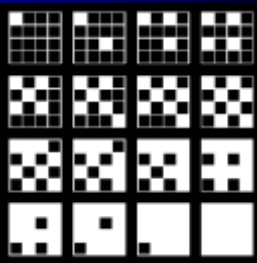
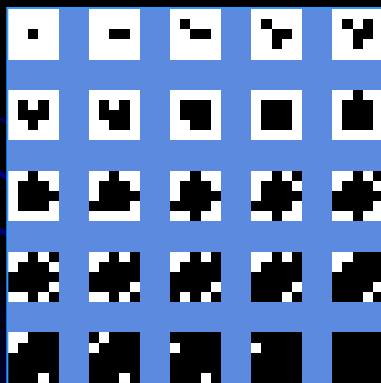
Ordered dithering - nedostatky

- Vytvára nežiaduce rekurzívne textúry, šrafovanie



Clustered dithering

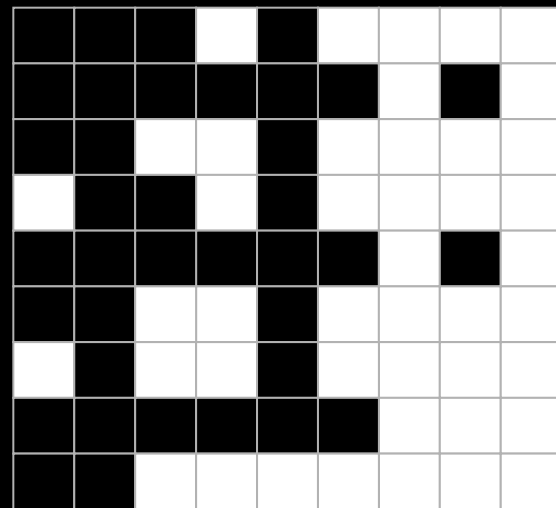
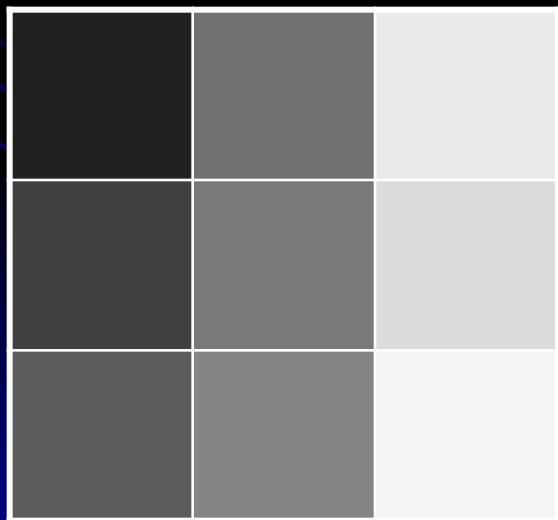
napodobuje novinovou tlač

$$D = \begin{bmatrix} 21 & 22 & 9 & 19 & 13 \\ 24 & 2 & 7 & 4 & 16 \\ 14 & 5 & 0 & 1 & 10 \\ 11 & 8 & 3 & 6 & 20 \\ 17 & 18 & 12 & 23 & 15 \end{bmatrix}$$


Patterning

pixel nahradíme blokom
výstupné zariadenie má vyššie rozlíšenie

Matice - podobne ako pre dithering



Patterning

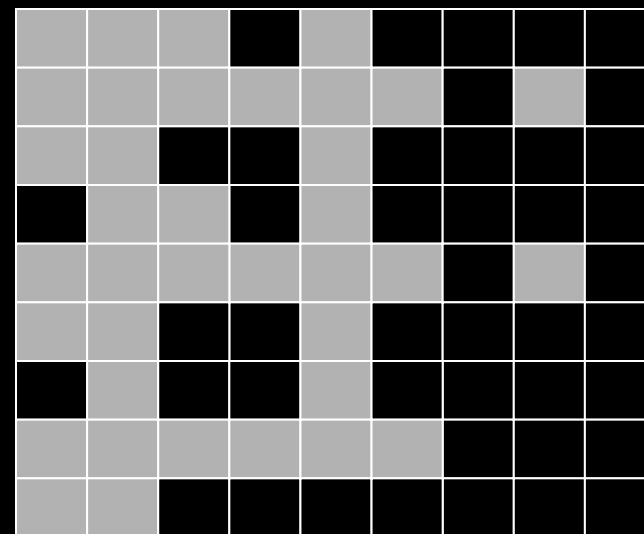
Vstupný obrázok

33	113	234
64	121	219
92	133	245

1	4	8
2	4	8
3	5	9

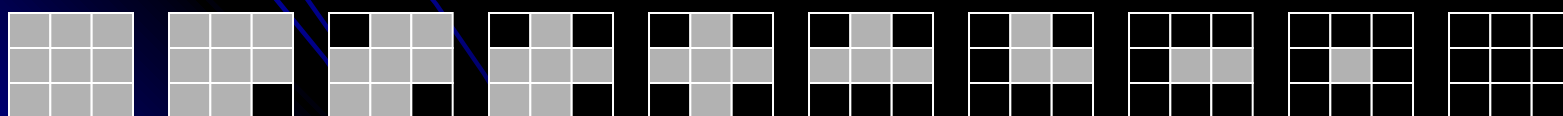
Zodpovedajúca matica

Výsledný obrázok



prahy

14 42 71 99 128 156 184 213 241



0 1 2 3 4 5 6 7 8 9

Patterning



Error diffusion

- chyba z kvantizačného procesu sa prenáša na susedné obrazové body aby sa následne negovala.

metóda *Floyd-Steinberg dithering*



Error diffusion

Nech $I(x,y)$ je hodnota obrazového bodu získaná pseudonáhodným procesom a $\Delta(x,y) = I(x,y) - I_Q(x,y)$ je kvantizačná chyba v obrazovom bode (x,y) , potom $I'(x,y)$ je vyjadrené nasledovne

$$I'(x,y) = I(x,y) + \delta(x,y)$$

kde

$$\delta(x,y) = \sum_{i=0}^m \sum_{j=0}^n C_{ij} \Delta(x-i, y-j)$$

a

$$C_{00} = 0 \text{ and } \sum_{i=0}^m \sum_{j=0}^n C_{ij} = 1$$

i a j určujú okolie na ktorom sa kvantizačná chyba akumuluje na základe koeficientov C_{ij} ,

Podmienka na C_{ij} zaručuje, že sa lokálne kvantizačná chyba priemeruje na nulu.

$\delta(x,y)$ reprezentuje dvoj-rozmerný filter

Error diffusion

- Spracúva obraz po riadkoch zhora dolu zľava doprava.
- Každý bod je zaokrúhlený k 0 alebo 1 (255).
- Chyba pri zaokrúhľovaní je potom rozdelená medzi susedné body podľa masky.

Napr. ak hodnota intenzity je 191, zaokrúhlením na 255 máme chybu 64.

=> moc svetlý bod, preto susedné body stmavíme, aby sa suma intenzít bodov (veľmi) nezmenila.

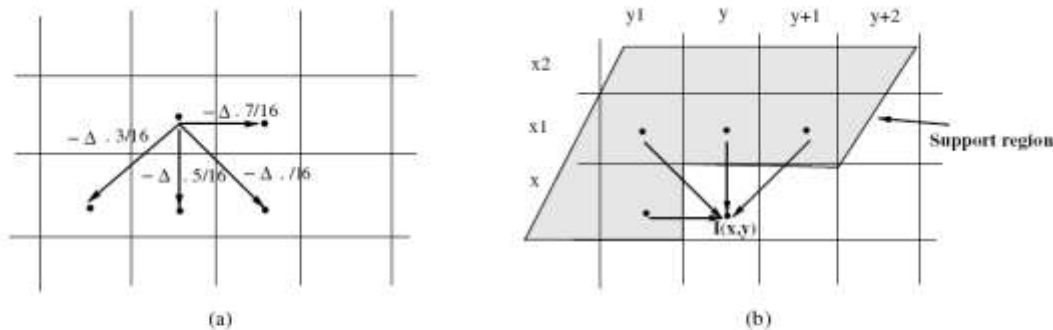


Figure 9.15 The Floyd-Steinberg algorithm: (a) the propagation of the quantization error to neighboring pixels and (b) the contribution of neighboring pixels to the noise vector.

Floyd-Steinberg

	●	7/16
3/16	5/16	1/16

35	89	95	132
68	112	100	150
51	45	98	127

0			

$$35/16 = 2.1875$$

●	15
11	2

35	104	95	132
79	114	100	150
51	45	98	127

Floyd-Steinberg

	●	7/16
3/16	5/16	1/16

35	104	95	132
79	114	100	150
51	45	98	127

0	0		

$$104/16 = 6.5$$

	●	46
20	33	6

35	104	141	132
99	147	106	150
51	45	98	127

Floyd-Steinberg

	●	7/16
3/16	5/16	1/16

35	104	141	132
99	147	106	150
51	45	98	127

0	0	255	

$$-114/16 = -7.125$$

	●	-50
-21	-36	-7

35	104	141	82
99	126	70	143
51	45	98	127

Floyd-Steinberg

vstup

35	89	95	132
68	112	100	150
51	45	98	127

výstup

0	0	255	0
0	255	0	255
0	0	0	255

Suma intenzít v originálnom obraze: 1102.
Suma intenzít vo výslednom obraze: 1020.
Priemerná chyba na pixel: -6.83

Error diffusion - nedostatky

- Výber farebnej palety - treba zaručiť aby každá farba pôvodného obrazu sa dala vyjadriť ako lineárna kombinácia farieb z farebnej palety
- Niekedy vzniknú „ghosts“ v obraze
- Vytvára sa zrnitý obraz



24-bit RGB image dithered to 3-bit
RGB using Floyd-Steinberg
dithering

Porovnanie rôznych metód



Pôvodný obraz



threshold



random



halftone



Ordered dithering



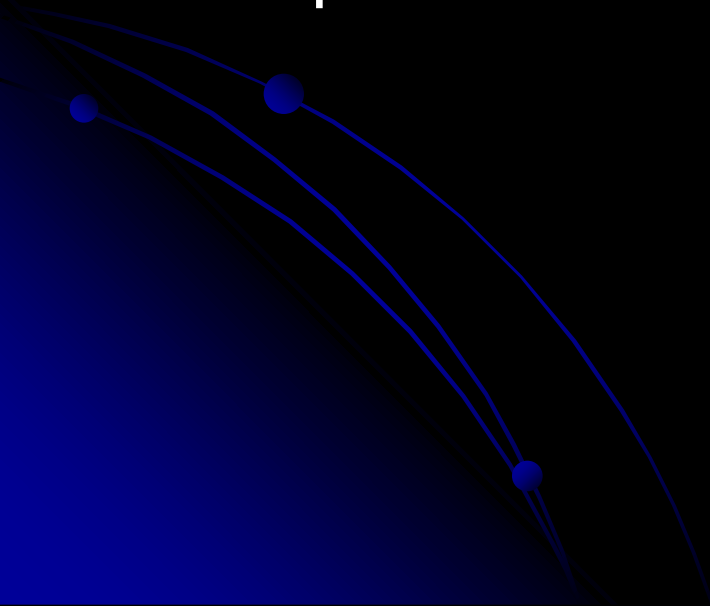
Floyd-Steinberg

Farebné obrázky

3 kanály – RGB, HSV, Lab, ...

2 problémy:

- Rozdelenie 3D priestoru
- Mapovanie farieb



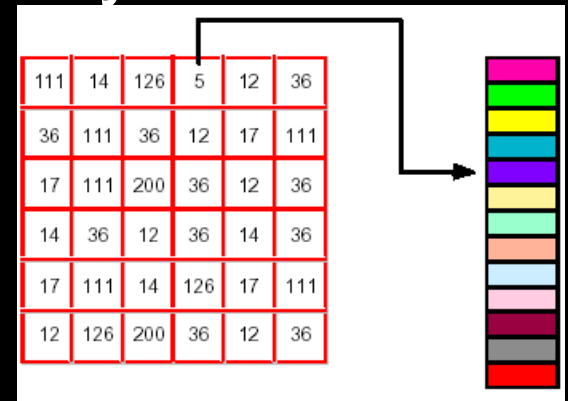
Indexovanie

Paleta farieb

Každému pixlu priradíme index do palety

Počet farieb v palette = 256

Index – 8 bitov

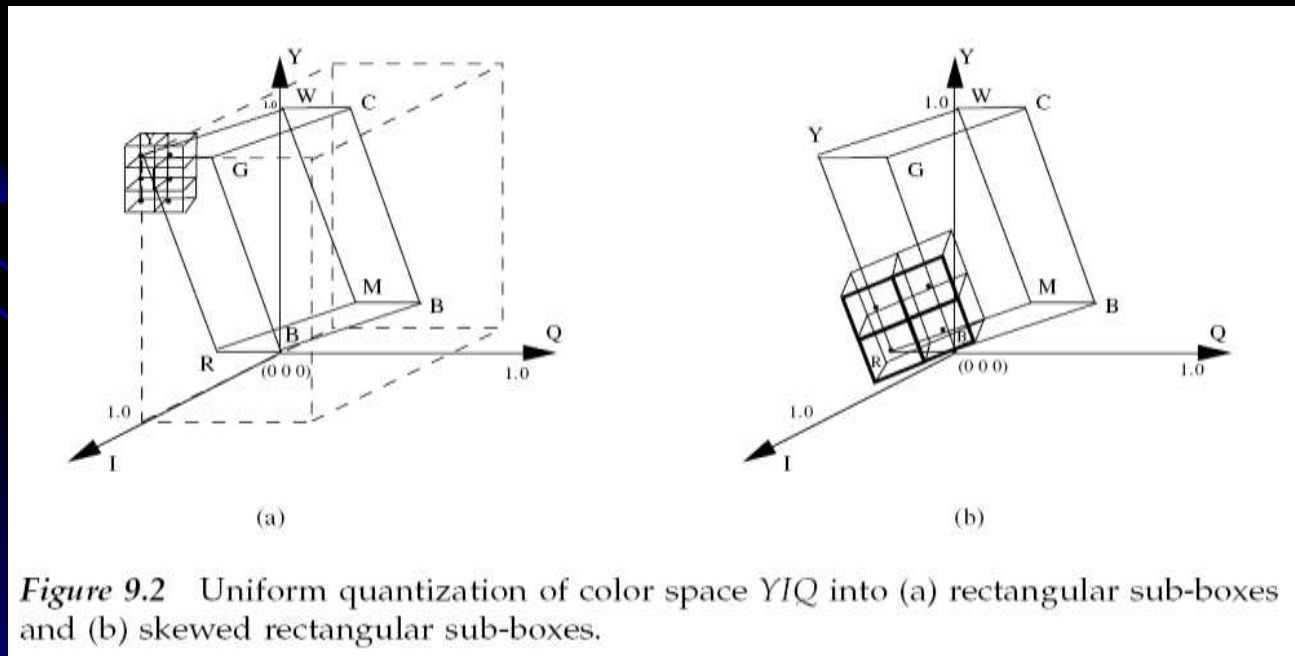


Zmenšíme potrebný priestor 3x
z pôvodného R,G,B à 8 bitov



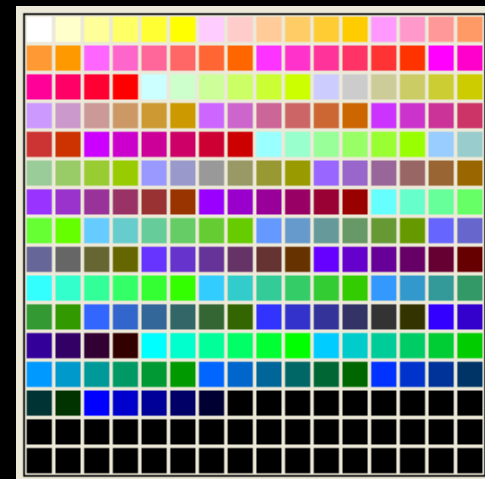
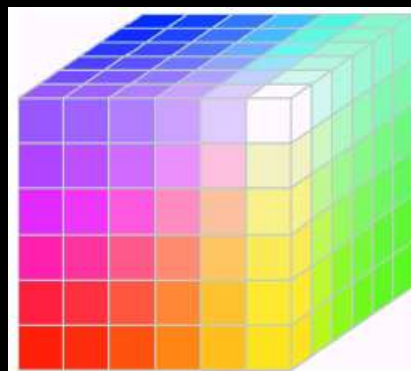
Obrazovo nezávislé kvantovanie

- Vyberieme K reprezentatívnych farieb z farebného priestoru nezávisle od frekvencie výskytu v konkrétnom obraze
- Rozdelenie priestoru farieb na K rovnakých subkociek.
- Reprezentatívne farby – centroid subkocky
- Techniky sa líšia podľa geometrie priestoru farieb

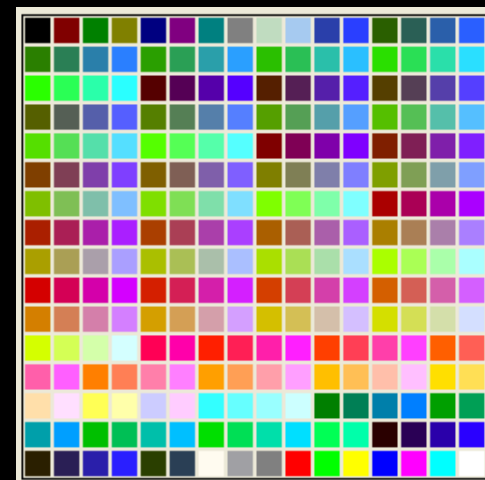


Pevná paleta - príklady

Pravidelné rozdelenie RGB kocky
na 6x6x6 – 216 farieb
Web-save color palette



Rozdelenie 3-3-2:
rozdelenie RGB kocky na 8x8x4 –
256 farieb





216

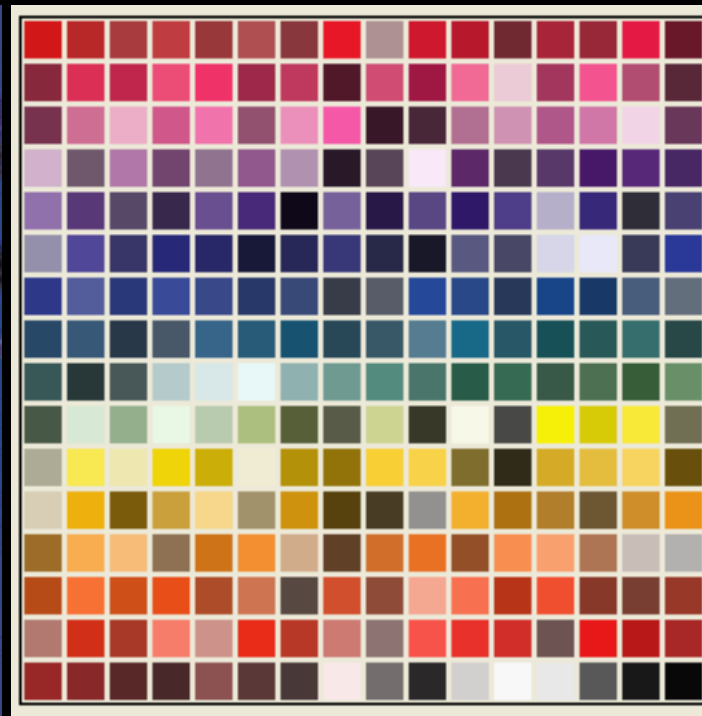


3-3-2

Adaptívna paleta

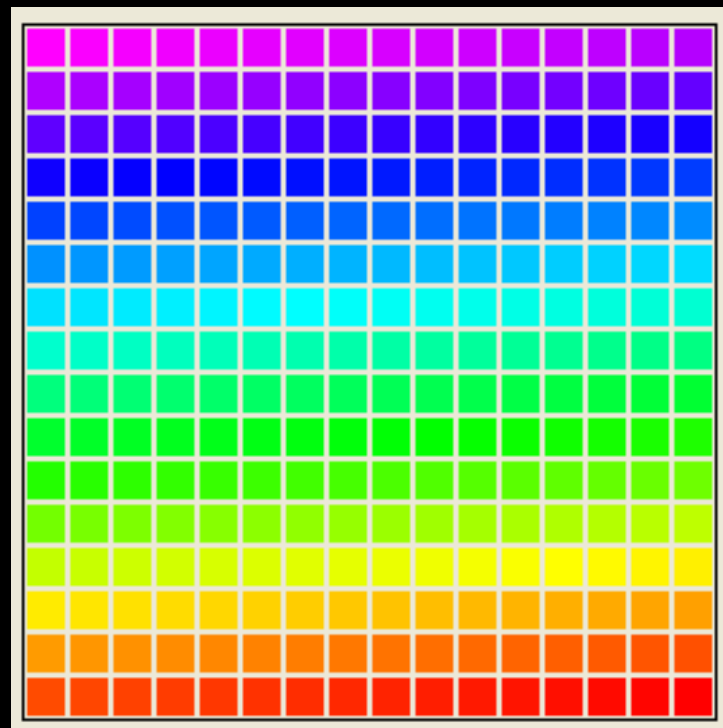


Farby v palette – vyberú sa z farieb v obraze



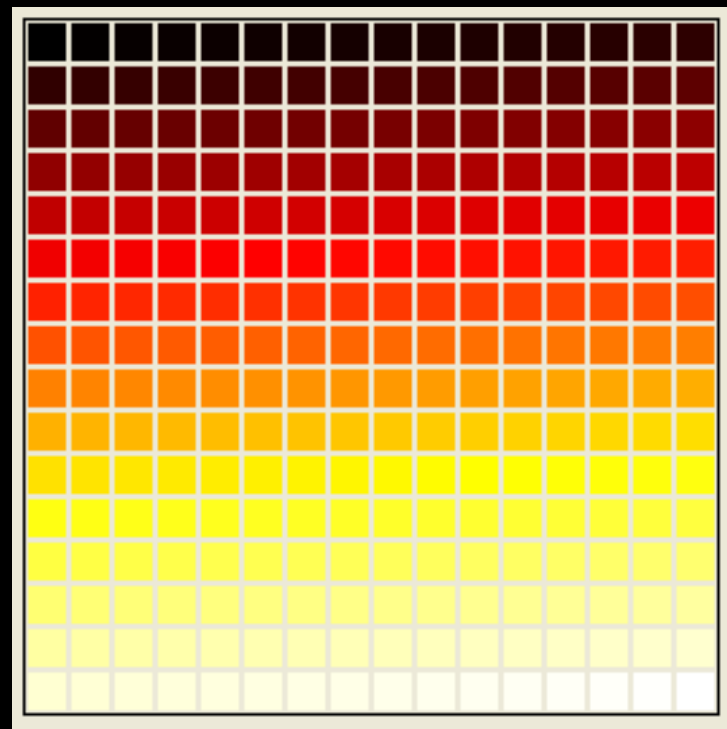
Ak zvolíme „nesprávnu“ paletu

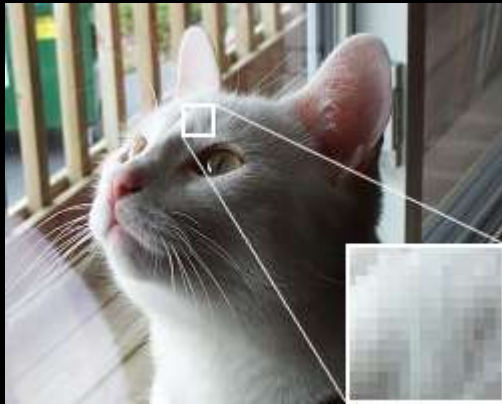
spektrum



Ak zvolíme „nesprávnu“ paletu

Black body

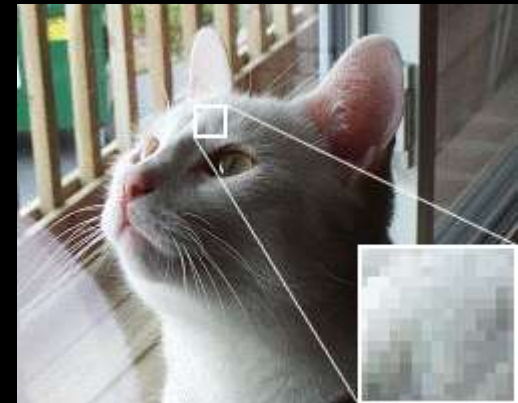




originál



Original image using the web-safe color palette with Floyd-Steinberg dithering.



Here, the original has been reduced to a 256-color optimized palette with Floyd-Steinberg dithering applied. The use of an optimized palette, rather than a fixed palette, allows the result to better represent the colors in the original image.