

Spracovanie obrazu (Image Processing)

Michal Fano, Gábor Blázsovits a Marek Zimányi

1	Matematický model obrazu a snímanie obrazu	2
1.1	Susednosť	2
1.2	Teselácia roviny	3
1.3	Digitalizácia	3
1.4	Kvantovanie	4
1.5	Vzorkovanie	4
1.6	Techniky vzorkovania	5
1.6.1	Bodové	5
1.6.2	Plošné	6
2	Geometrické transformácie a transformácie hodôt jasu	6
2.1	Histogram	6
2.2	Zvýšenie kontrastu	7
2.3	Ekvalizácia histogramu	7
2.4	Binárne operácie	9
2.4.1	Bodové jasové transformácie	9
3	Filtrácia šumu, detekcia hrán	10
3.1	Konvolúcia	10
3.2	Diskrétna konvolúcia:	11
3.3	Metóda rotujúcej masky	11
3.4	Vyhladzovanie obrazu	12
3.4.1	Vyhladzovacie filtre - Mean	12
3.4.2	Vyhladzovacie filtre - Gaussian	13
3.4.3	Vyhladzovacie filtre - Median	14
3.5	Hľadanie hrán	15
3.5.1	Robertsov hr. detector	16
3.5.2	Sobelov hr. Detector a ine	16
3.5.3	Prechod nulou	17
3.5.4	LoG operátor (Laplacian of Gaussian)	18
3.5.5	Canny detektor	20
4	Segmentácia	21
4.1	Prahovanie (Thresholding)	22
4.1.1	Jednoduché prahovanie	22
4.2	Adaptívne prahovanie	23
4.3	Sledovanie hranice	23
4.4	Techniky založené na oblastiach	25
4.4.1	Selekcia podľa farby – Narastanie oblasti spájaním bodov do oblasti	25
4.4.2	Spájanie oblastí	25
4.4.3	Rozdeľovanie oblasti	26
4.4.4	Rozdeľovanie a spájanie oblastí	26
5	Morfológia	26
5.1	Dilatácia	28
5.2	Erózia	28
5.3	Otvorenie	29
5.4	Uzavretie	29
5.5	Hit&Miss	30
5.6	Thinning	30
5.7	Thickening	31

5.8	Skelet.....	31
6	Reprezentácia Tvaru	32
6.1	Run-Length Encoding (RLE).....	33
6.2	Quadtree.....	33
6.3	Chain Code.....	33

1 Matematický model obrazu a snímanie obrazu

obraz

- spojitý - funkcia $f(x,y)$, jas - funkčná hodnota
- digitálny - diskretizácia priestorových a jasových hodnôt (matica čísel, 1 element - **pixel**)

zdroj obrazu

- syntetický - metódy počítačovej grafiky
- prirodzený - kamera, skener

Obrazová funkcia (OF) – zobrazenie $f: I(R^2) \rightarrow (R^n)$

$I = f(x,y)$

$0 \leq f(x,y) < \infty$

Komponenty OF

jas, svietivosť: $0 \leq i(x,y) < \infty$

reflectance: $0 \leq r(x,y) \leq 1$

$f(x,y) = i(x,y) \cdot r(x,y)$

Obor hodnôt môže predstavovať jas (TV, kamera, scanner), intenzita RTG, ultrazvuk, tepelné, žiarenie, viacej spektrálnych pásiem

My sa budeme zaoberať len skalou úrovne šedej (grayscale)

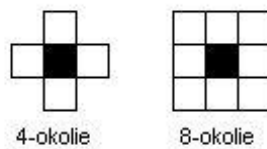
Oblasť je súvislá množina obrazových elementov, pre ktorú platí, že medzi každými dvoma bodmi existuje cesta patriaca celá do tejto množiny. Predpokladajme, že R_i sú oblasti obrazu. Nech R je oblasť ktorá vznikne zjednotením všetkých oblastí R_i . Potom R^C je množinovým doplnkom oblasti R , nazývame ho pozadím.

Objekty sú oblasti, ktoré obvykle odpovedajú entitám zobrazovaného sveta. V jednoduchom praktickom prípade, keď má bod jas väčší ako určitý prah, priradíme ho k objektu.

1.1 Susednosť

Ďalším dôležitým pojmom je **susednosť**. Rozlišujeme **4-susednosť** a **8-susednosť**. 4-susedia daného obrazového elementu sú body s jednotkovou vzdialenosťou v metrike D_4 . Podobne 8-susedia daného bodu sú body s jednotkovou vzdialenosťou v D_8 . Tiež sa hovorí **4-okolie**, alebo **8-okolie**

Súvislosť a susednosť definovaná na diskkrétnej štvorcovej mriežke nás privedie k určitým paradoxom. Predstavme si úsečku s 45 stupňovým sklonom v digitálnom obraze. Ak uvažujeme 4-susednosť, potom táto úsečka je v každom svojom bode nesúvislá. Ďalším paradoxom je, že dve pretínajúce sa úsečky v digitálnom obraze sa len dotýkajú.

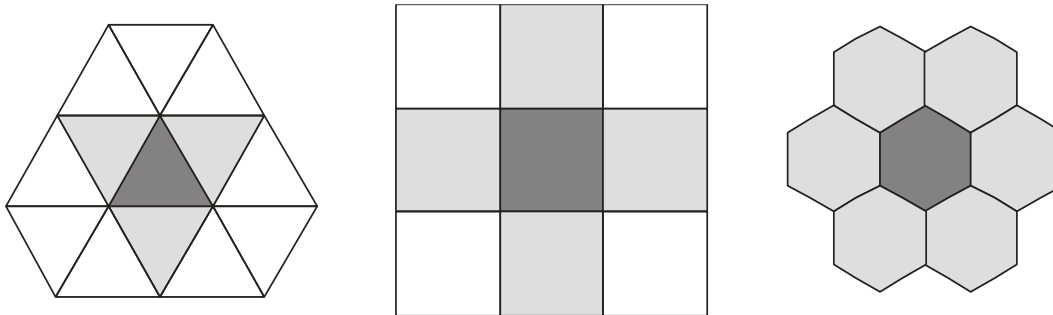


Jedným riešením pre tieto paradoxy je použiť 8-susednosť pre objekty a 4-susednosť pre pozadie. Je to ale nepraktické riešenie. Ďalšou možnosťou je použiť hexagonálnu mriežku, v ktorej paradoxy nevznikajú. Narazíme ale na realizačný problém, pretože väčšina grafických zariadení podporuje štvorcový raster.

Súvislosť a susednosť definovaná na diskretnej štvorcovej mriežke nás privedie k určitým paradoxom. Predstavme si úsečku s 45 stupňovým sklonom v digitálnom obraze. Ak uvažujeme 4-susednosť, potom táto úsečka je v každom svojom bode nesúvislá. Ďalším paradoxom je, že dve pretínajúce sa úsečky v digitálnom obraze sa len dotýkajú.

Pod pojmom **hranica oblasti** rozumieme množinu všetkých bodov, ktoré majú aspoň jedného suseda, ktorý nepatrí do oblasti. V digitálnom obraze rozlišujeme **vonkajšiu** a **vnútornú** hranicu. Pre vnútornú hranicu potom platí predošlá definícia. Vonkajšia hranica je hranicou pozadia.

1.2 Teselácia roviny



Teselácia

- pokrytie pravidelnými polygónmi
- trojuholníková, štvoruholníková, hexagonálna
- pixel (Picture Element)

štvorhuloníky

- anizotropia a pridružené problémy
- metriky
 - euklidovská
 - Manhetenská
 - chessboard
- pojem susednosti
- pojem súvislosti

1.3 Digitalizácia

Pri získavaní digitálneho obrazu dochádza k prechodu od spojitej funkcie k diskretnej funkcii. Tento proces sa nazýva **digitalizácia** a odohráva sa v dvoch nezávislých krokoch, ktoré sú **kvantovanie** a **vzorkovanie**. Prechod od spojitej funkcie $f(x,y)$ k diskretnej funkcii $I(x,y)$, a nezávislé kroky

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & K & f(0,N-1) \\ f(1,0) & f(1,1) & \Lambda & f(1,N-1) \\ & M & O & \\ f(M-1,0) & f(M-1,1) & K & f(M-1,N-1) \end{bmatrix}$$

$X = 0, 1, \dots, N-1$

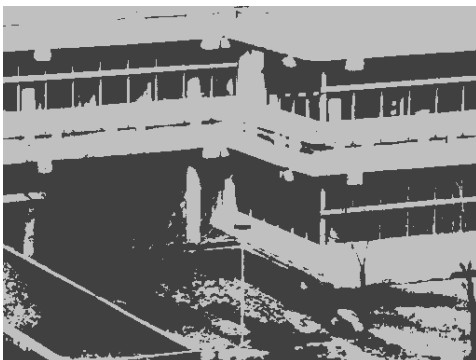
$y = 0, 1, \dots, M-1$

$F(x,y) = 0, 1, \dots, G-1$

1.4 Kvantovanie

Princípom kvantovania je diskretizácia oboru hodnôt obrazovej funkcie. Obor hodnôt funkcie sa rozdelí na intervaly, ku ktorým je potom pridelená jediná, zástupná hodnota. Podľa spôsobu rozdelenia kvantovacej veličiny hovoríme o *uniformnom* a *neuniformnom* kvantovaní. Uniformné kvantovanie používa konštantnú dĺžku intervalu, kým neuniformné kvantovanie premenlivú dĺžku intervalu.

Na výber zástupnej hodnoty môžeme použiť rôzne techniky. Obvykle sa používa priemer hodnôt z celého intervalu, váhovaný priemer, medián, priemer z hodnôt na okraji intervalu.

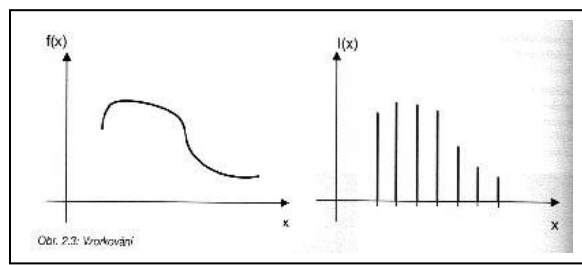


Kvantovanie

- lineárne vs. logaritmické
- Neuniformné
- riedko okolo hrán

Pozor! chyba v kvantovaní

1.5 Vzorkovanie



odoberanie hodnôt - vzoriek v rovnakých intervaloch

Vzorkovaním spojitej funkcie $f(x,y)$, rozumieme odoberanie hodnôt - vzoriek, v rovnakých intervaloch. Spojitý obraz sa rozdelí na N riadkov a M stĺpcov. Prienik riadku a stĺpca potom tvorí obrazový bod. V súvislosti so vzorkovaním spojitej obrazovej funkcie sa vyskytujú dve otázky. Aký má byť interval vzorkovania? Odpoveď nám dá **Shannonova veta** o vzorkovaní z teórie signálov, ktorá hovorí, že vzorkovacia frekvencia musí byť aspoň dvakrát väčšia ako najvyššia frekvencia vo vzorkovanom signáli. V našom prípade to znamená, že interval vzorkovania musí byť takej veľkosti, aby bol menší alebo rovný polovici rozmeru najmenších detailov v obraze. Druhou otázkou je, že aká má byť vzorkovacia mriežka? Musí byť pravidelná a musí pokryť celú rovinu. Štvoruholníková a hexagonálna mriežka vyhovuje týmto podmienkam. V praxi sa najčastejšie používa štvoruholníková mriežka, lebo je jednoduchšie realizovateľná. Jednému vzorkovaciemu bodu odpovedá v digitálnom obraze jeden obrazový element, pixel. Proces vzorkovania teda určuje definičný obor obrazovej funkcie

Komplikácie pri nevhodnom postupe

- Moiré
- Aliasing
- antialiasová čiara (nakreslit)
- rotujúce koleso
- plot (Foley / 819 / 17.2)
- pásikové spektrum (Foley / 628 / 14.18)
- šachovnica (Foley / 828 / 17.12)



1.6 Techniky vzorkovania

1.6.1 Bodové

- supersampling
- stochastické

- regulárne

Presná reprezentácia nezaručená,
Doslédok: miznúce objekty pri animácii

1.6.2 Plošné

- nevážené
- vážené
- vážené s prekrývaním

Snaha zdokonaľiť bodové vzorkovanie

Vážený priemer

- bez prekrývania --> blikanie objektov
- prekrývanie --> sampling theory

Poznámky zo „sampling theory“

2 Geometrické transformácie a transformácie hodôt jasu

- Bodové jasové transformácie
- Binárne operácie
- Histogram

Z angl. Image Enhancements

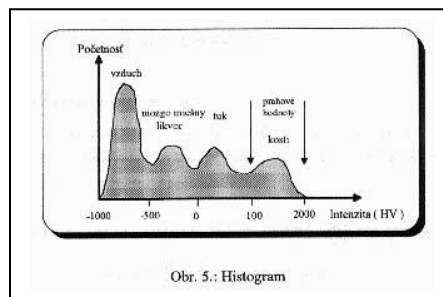
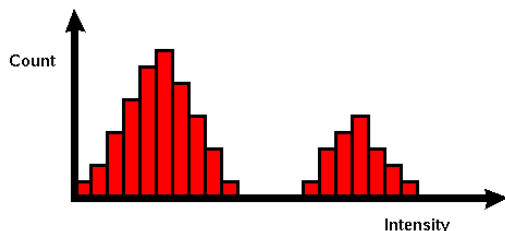
Predpoklad“

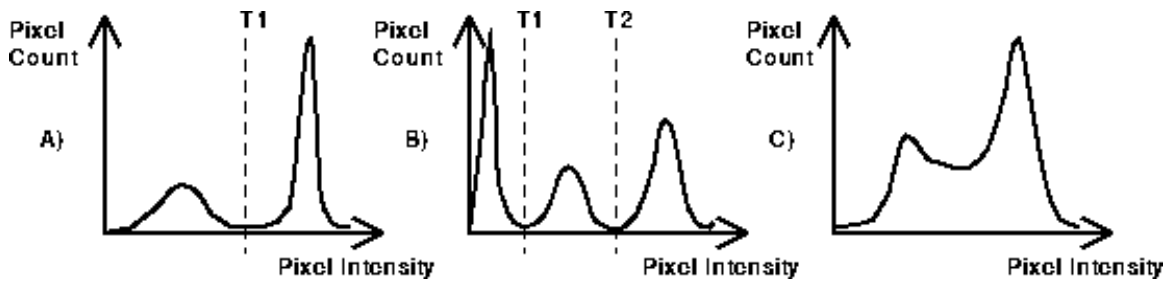
- práca s obrazom po diskretizácii
- k dispozícii máme už len diskretizovanú formu signálu

2.1 Histogram

Histogram jasu je funkcia ktorá priradí každej úrovni jasu zodpovedajúcu početnosť príslušného jasu v programe. Histogram v spojitom prípade vyjadruje funkciu hustoty rozdelenia, ktorá je škálovaná na jednotkový integrál.

- diskrétna reprezentácia pravdepodobnosti hustoty intenzít



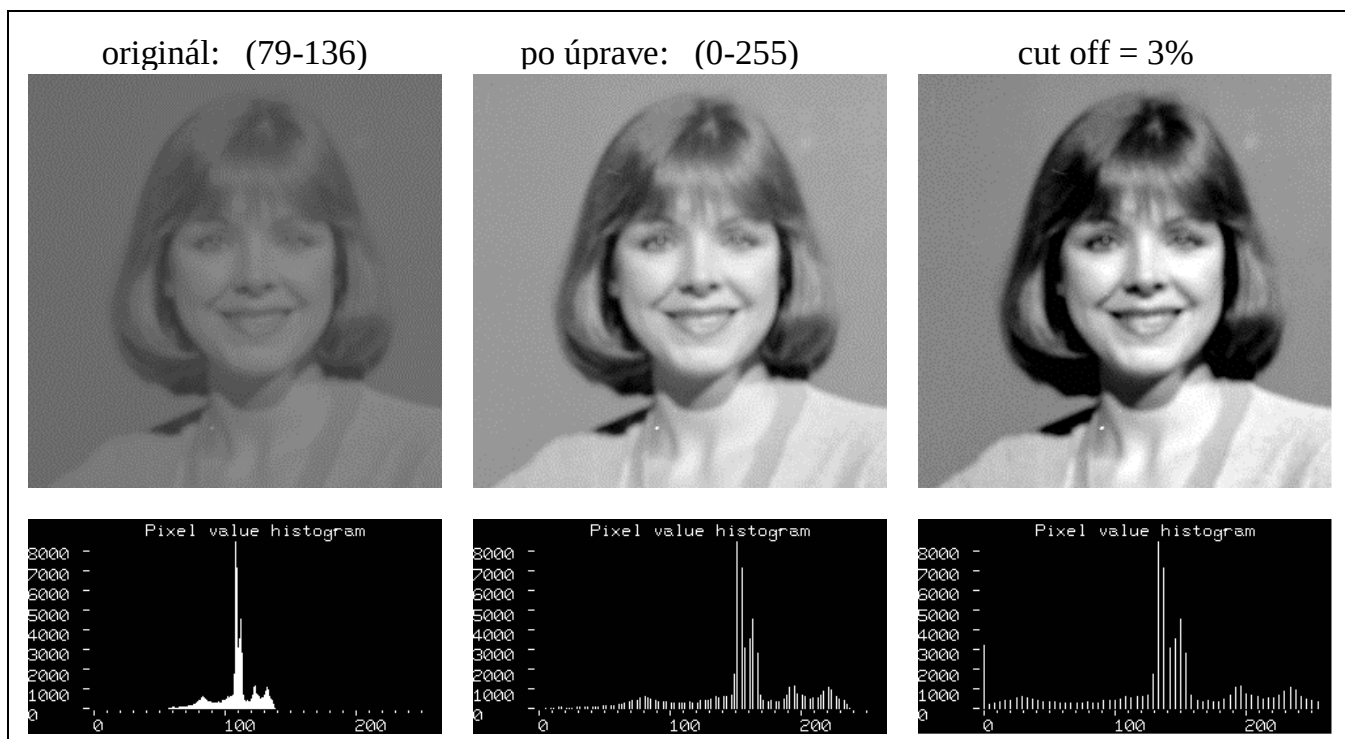


2.2 Zvýšenie kontrastu

Contrast stretching (Normalization)

Inou možnosťou na dosiahnutie lepšej kvality obrazu je **normalizácia histogramu**. Použitím transformácie

$$q = \tau(p) = (p - p_0) \frac{q_k - q_0}{p_k - p_0} + q_0 \text{ rozťahujeme histogram na novú jasovú stupnicu } \langle q_0, q_k \rangle.$$



2.3 E
kv
ali
zá
ci
a
hi
st
og
ra
m

u

Na zvýšenie kontrastu v obraze sa veľmi často používa **ekvalizácia histogramu**. Ekvalizáciou sa snažíme dosiahnuť ideálny histogram, ktorý obsahuje rovnaký počet z každej zastúpenej jasovej hodnoty. Majme histogram pôvodného obrazu $H(p)$ s jasovou stupnicou $p = \langle p_0, p_k \rangle$. Transformáciou jasovej stupnice dostávame ekvalizovaný histogram $G(q)$ s jasovou stupnicou $q = \langle q_0, q_k \rangle$. Požadujeme, aby zobrazenie $\tau : p \rightarrow q$ bola monotónna

$$\sum_{i=0}^k G(q_i) = \sum_{i=0}^k H(p_i) \quad (1)$$

Sumy v rovnici (1) sú diskkrétne distribučné funkcie. Ekvalizovaný histogram $G(q)$ zodpovedá rovnomernému rozdeleniu f , ktorá ma konštantnú hustotu pravdepodobnosti.

$$f = \frac{N^2}{q_k - q_0} \quad (2)$$

Po dosadení (2) za ľavú stranu (1) v spojitom prípade dostaneme

$$N^2 \int_{q_0}^q \frac{1}{q_k - q_0} ds = \frac{N^2 (q - q_0)}{q_k - q_0} = \int_{p_0}^p H(s) ds \quad (3)$$

Teda transformácia τ má tvar:

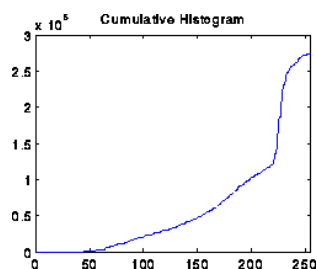
$$q = \tau(p) = \frac{q_k - q_0}{N^2} \int_{p_0}^p H(s) ds + q_0 \quad (4)$$

Diskrétna aproximácia transformácie τ je

$$q = \tau(p) = \frac{q_k - q_0}{N^2} \sum_{i=0}^p H(i) + q_0$$

Techniky pracujúce s tvarom histogramu sú vo všeobecnosti nelineárne a nemonotónne
Ekvalizácia

- monotónna
- nelineárna
- snaha o rovnomernú distribúciu intenzít

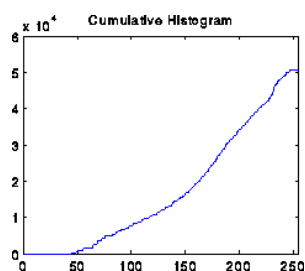


Zlepšenie

- kumulatívny histogram z regiónu s rovnomernejším zastúpením jasových hodnot
- prirodzenejší výsledný kontrast

Zovšeobecnenie

- Priama špecifikácia histogramu



2.4 Binárne operácie

2.4.1 Bodové jasové transformácie

Zdraznenie detailov reprezentovaných úzkym rozsahom vyšších hodnôt

Mocninový nezávisí v takej miere na rozsahu vstupných intenzít

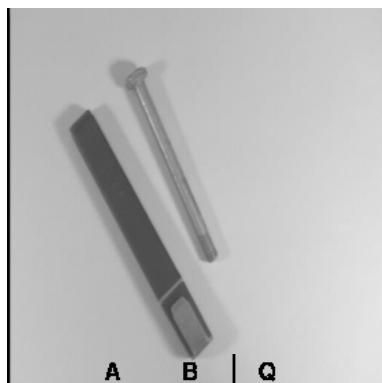
Mocninový operátor známy aj ako gamma korekcia

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

AND

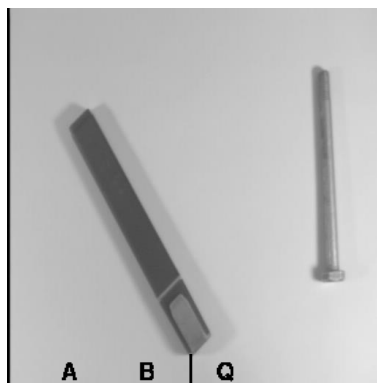
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

NAND



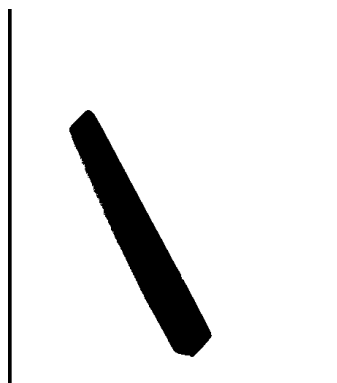
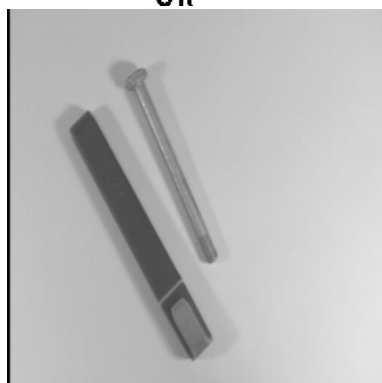
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

OR



A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

NOR

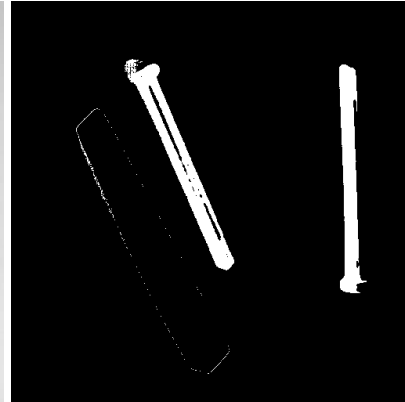
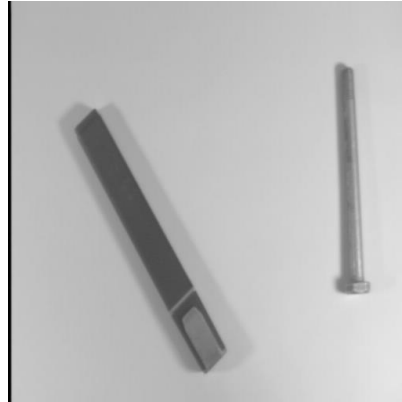
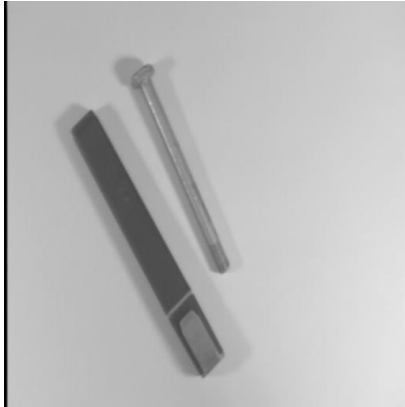


A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

XOR

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

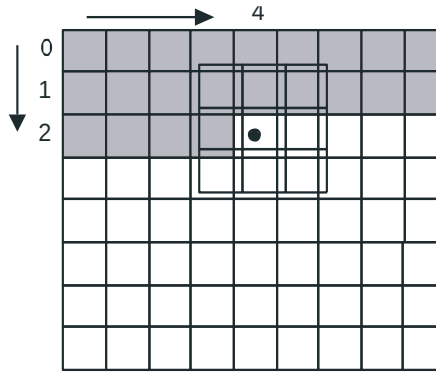
XNOR



3 Filtrácia šumu, detekcia hrán

3.1 Konvolúcia

- Výpočet výstupu na základe okolia vo vstupe
- Výpočtová náročnosť
 - závislosť od veľkosti okolia
 - hardwerová implementácia
- Použitie
 - 1) vyhladenie šumu, bohužiaľ aj hrán
 - 2) gradientné: zvýraznenie vyšších frekvencií, bohužiaľ aj šumu
- metódy sú “slepé”
- voľba filtrov a ich veľkostí metódou pokusov a omylov



3.2 Diskrétna konvolúcia:

Zopakovanie spojitaj

- prechod k diskkrétnej
- konvolučné jadro
 - filter $\Leftrightarrow$ konvolučné jadro
 - stredovo symetrické (+ / - v integráli sa neprevia)
 - veľkosť $N \times N$ (N zvykne byť nepárne)

$$g(x, y) = \sum_{(m,n) \in O} h(x-m, y-n) f(m, n)$$

Algoritmus v pseudokóde:

for each pixel (x,y)

begin

G(x,y) = 0;

for each element (a,b)

begin

G(x,y) += K(a,b) * F(x-a,y-b)

end

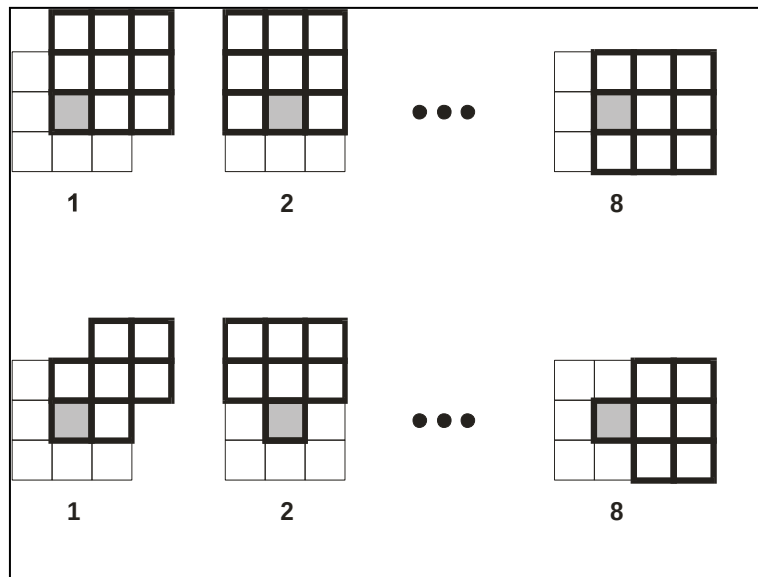
end

Transformácie v lokálnom okolí bodu sa delia na dve skupiny:

- Vyhľadovanie – tieto metódy sa snažia znížiť šum v obraze, ale rozostávajú hrany.
- Ostrenie – detekcia hrán a čiar, ale zvyšuje šum

3.3 Metóda rotujúcej masky

- Kombinácia protichodných techník
 - priemeruje len homogénne časti
 - má dokonca mierne ostriaci charakter
- Princíp
 - momogenita ako rozptyl jasu
 - threshold
 - “rotácia“ masky
 - ak je oblasť homogénna filtruj, inak ignoruj



- Iteratívne použitie
 - rýchla konvergencia do stabilného stavu

3.4 Vyhľadzovanie obrazu

- nový jas <-- priemer jasov okolia
- problém: rozmazanie hrán

cieľ - potlačiť šum vzniknutý pri snímaní, prípadne pri prenose obrazu

- priestorový a frekvenčný prístup
- transformácia obrazu pomocou operátora v nasledujúcej forme:

$$g(x, y) = T[f(x, y)]$$

- Riešenia
 - z viacerých predloh
 - väčšia váha streového bodu
- priemerovacie
- priemerovanie s rotujúcou maskou
- Gaussian
- Medián

Vyjadrenie operátora pomocou **masky okolia**, veľkosť masky 3x3, 5x3, 9x9, ošetrenie okrajov

3.4.1 Vyhľadzovacie filtre - Mean



originál + Gaussov šum



mean: 3 x 3

$$h_1 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

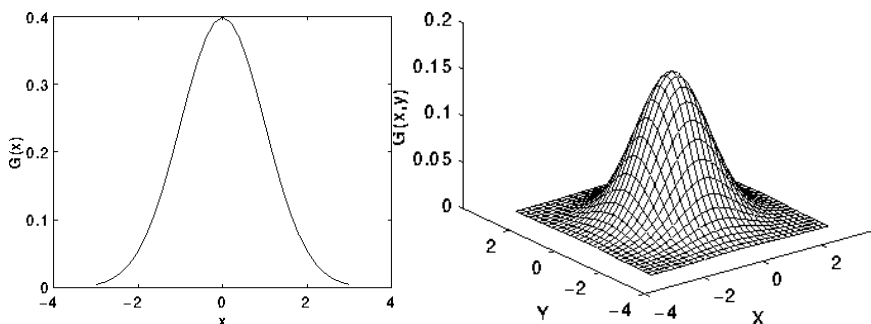
$$h_2 = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

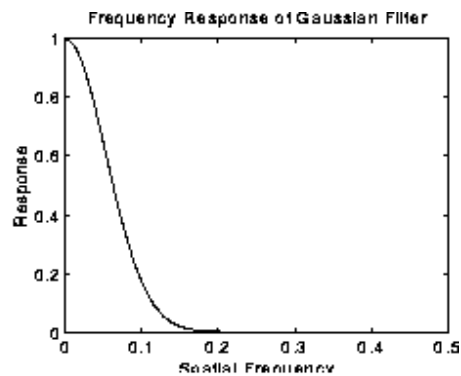
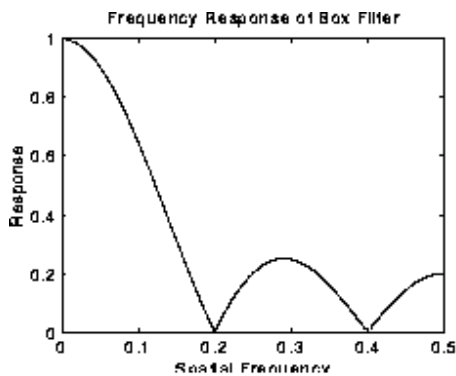
$$h_3 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Maska pre priemerovanie a gausov filter

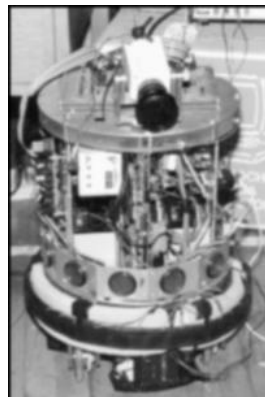
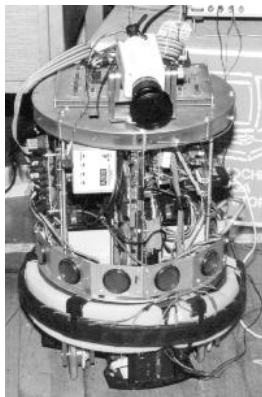
3.4.2 Vyhľadovacie filtre - Gaussian

- Gaussova distribúcia
 - Sigma - štandardná odchýlka - jediný parameter !
 - Nenulová --> nekonečný nosič --> windowing
- Prečo je Gauss lepší ako Mean ?
 - Frequency response
 - mean prepustí vyššie frekvencie
- implementácia
 - separabilne pre x a y
 - kernel = (1 / 10.7) x [1.3, 3.2, 3.8, 3.2, 1.3]





Example1:



3.4.3 Vyhľadovacie filtre - Median

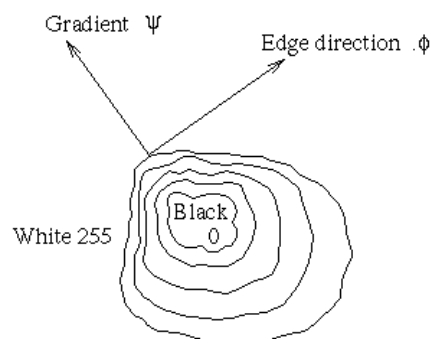
- x - pravdepodobnostná veličina
- medián M : $P(M < x) = 1/2$
 - diskrétno: stredný prvok postupnosti
 - zrýchlenie, lebo netreba triediť celú postupnosť
- redukuje stupeň rozmazania
- potlačuje impulzivný šum
- iteratívne použitie
- problém: porušuje tenké čiary a rohy
- riešenie - iný tvar masky (napr. kríž)
- varianty: (order statistic filtration)
 - vybrať iný prvok postupnosti (min, max)



3.5 Hľadanie hrán

Hranou sa nazývajú miesta v obraze, kde sa prudko mení jasová hodnota obrazovej funkcie $f(x,y)$. Z matematickej analýzy vieme, že na sledovanie priebehu funkcie dvoch premenných slúži **gradientový operátor** ∇ . Gradient je vektorová veličina určujúca smer rastu funkcie a veľkosť rastu. Body s veľkou hodnotou gradientu sa považujú za hrany.

- Hrana
 - hrana definovaná gradientom jasu
 - veľkosť a smer gradientu
 - prudká zmena intenzít -> gradient
 - hrana vs. hranica
- Detektory
 - konvolučné
 - detekcia: $f''(x) = 0$
 - parametrický model obrazu
- gradient
 - $\nabla f(x,y) = (G_x, G_y) = (\partial f / \partial x, \partial f / \partial y)$
- veľkosť a jej aproximácia
 - $|\nabla f(x,y)| = \text{euclidean}$
 - $|\nabla f(x,y)| = \text{manhattan} = |G_x| + |G_y|$
- definícia smeru
 - $\alpha = \arctan (G_y / G_x)$



V spojitom prípade, gradient funkcie dvoch premenných je:

$$\nabla f(x,y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$$

Veľkosť gradientu:

$$|\nabla f(x,y)| = \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2}$$

Smer hrany sa obvykle definuje ako kolmý na smer gradientu. Smer gradientu je:

$$\Psi = \arctg\left(\frac{\partial f}{\partial x} / \frac{\partial f}{\partial y}\right)$$

V diskretnom prípade parciálne derivácie aproximujeme diferenciami

$$\frac{\partial f(x, y)}{\partial x} \approx \Delta_x f(x, y) = f(x, y) - f(x-1, y)$$

$$\frac{\partial f(x, y)}{\partial y} \approx \Delta_y f(x, y) = f(x, y) - f(x, y-1)$$

V prípade, že hľadáme silné hrany, bez ohľadu na smer, môžeme použiť **Laplacian** ∇^2 . Tento operátor vychádza z druhých parciálnych derivácií. Pre monotónne rastúcu obrazovú funkciu f je Laplacian ∇^2 nulový, ak gradient dosahuje lokálne maximum

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}$$

3.5.1 Robertsov hr. detector

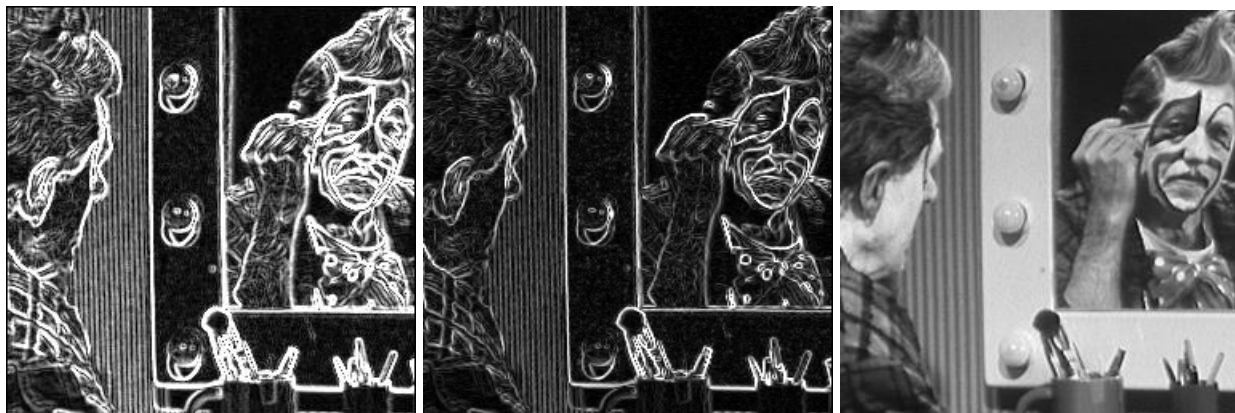
Robertsov (pre 2 smery)

$$h_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad h_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Veľkosť gradientu sa počíta podľa vzťahu

$$|\nabla f(x, y)| = |f(x, y) - f(x+1, y+1)| + |f(x, y+1) - f(x+1, y)|$$

Operátor je citlivý na šum, lebo používa malé okolie na aproximáciu



3.5.2 Sobelov hr. Detector a ine

Sobelov

$$h_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad h_2 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \quad h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \dots$$

Prewitov

$$h_1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad h_2 = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix} \quad h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \dots$$

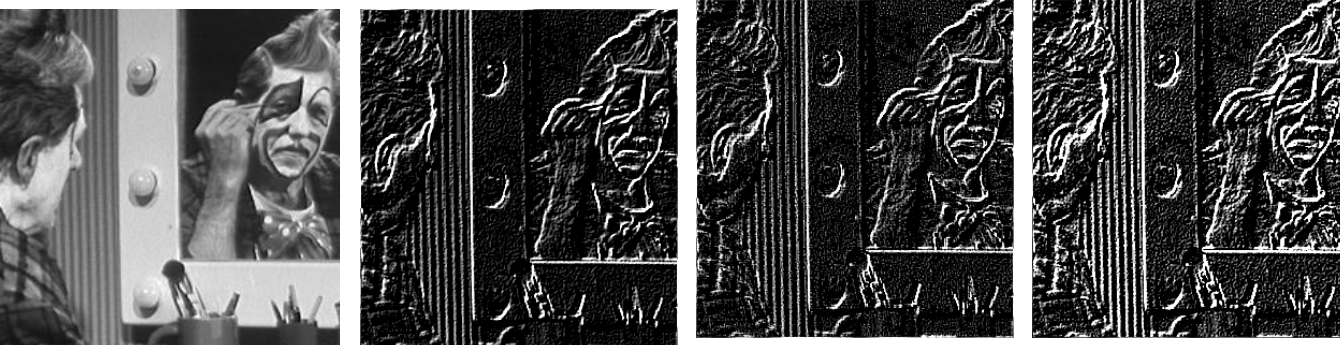


Robinsonov

$$h_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix} \quad h_2 = \begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{bmatrix} \quad h_3 = \begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix} \dots$$

Kirschov

$$h_1 = \begin{bmatrix} 3 & 3 & 3 \\ 3 & 0 & 3 \\ -5 & -5 & -5 \end{bmatrix} \quad h_2 = \begin{bmatrix} 3 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & -5 & 3 \end{bmatrix} \quad h_3 = \begin{bmatrix} -5 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & 3 & 3 \end{bmatrix} \dots$$

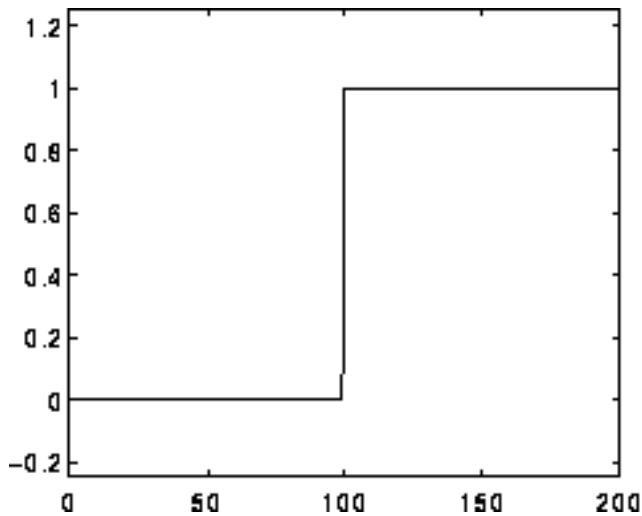


3.5.3 Prechod nulou

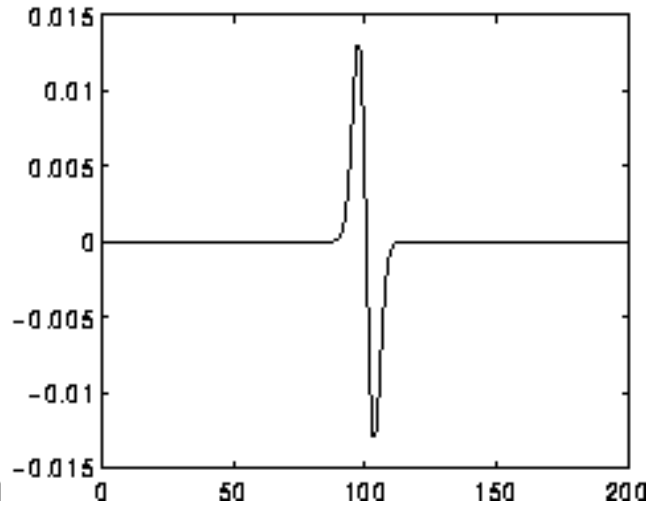
- z angl. Zero Crossing
- pracuje na väčšom okolí
 - eliminuje závislosť na šume
- f je na hrane $\Rightarrow f'' = 0$ (f' mení znamienko)
 - ľahšie sa hľadajú prechody nulou ako maximá f'
- Postup
 1. LoG = vyhladenie (gauss) + odhad 2. Derivácie (Laplace)
 2. Detekcia prechodu nulou

a) prahovanie LoG s prahom $T = 0$

b) uvažovať obe strany prahu vybrať pozíciu s menšou absolútnou hodnotou



Signál v okolí 1D hrany



Odozva na LoG

3.5.4 LoG operátor (Laplacian of Gaussian)

- súčet druhých parc. derivácií
- aproximácia 2.derivácie
- náhla zmena v obraze -> hrany
- 3 rozne diskkrétne aproximácie
 - isotropné - aproximujú kruh --> smerovo nezávislé
- malé jadro --> citlivosť na šum --> vyhladenie
 - Gauss (G) a potom Laplace (L) - výpočtovo náročné
 - asociativita konvolúcie $L * (G * I) = (L * G) * I$
 - menej výpočtov
 - $(L * G)$ si pre dané sigma v G predpočítame

Základom prístupu je hľadanie miesta v obraze, kde druhá derivácia obrazovej funkcie prechádza nulou. Na tomto mieste prvá derivácia dosahuje lokálne maximum.

Problémom je, že druhá derivácia je citlivá na šum. Preto sa najprv odstráni šum a až potom sa derivuje. Na odstránenie šumu sa používa lineárny filter, ktorého koeficienty v konvolučnej maske zodpovedajú dvojrozmernému gaussovskému rozdeleniu

$$G(x, y, \sigma) = e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

kde σ je stredná kvadratická odchýlka, ktorá určuje okolie. Body bližšie k stredu majú väčšiu váhu ako vzdialenejšie. Na odhad druhej derivácie z filtrovanej obrazovej funkcie $G(x, y, \sigma) * f(x, y)$, kde $*$ je konvolúcia, môžeme použiť Laplacian ∇^2 . Dostaneme LoG operátor, $\nabla^2[G(x, y, \sigma) * f(x, y)]$ v ktorom vďaka linearite operácií môžeme zameniť poradie konvolúcie a derivácie.

$$\nabla^2(G(x, y, \sigma) * f(x, y)) = (\nabla^2 G(x, y, \sigma)) * f(x, y)$$

Hodnota $\nabla^2 G(x, y, \sigma)$ nezávisí od obrazu, preto sa to dá vypočítať dopredu. Použitím substitúcie $x^2 + y^2 = r^2$ dostávame

$$G(x, y, \sigma) = e^{-\frac{r^2}{2\sigma^2}}$$

kde r je Euklidovská vzdialenosť od stredu. Gaussian je stredovo symetrický, preto je substitúcia prípustná. Prvá derivácia je

$$G'(x, y, \sigma) = -\frac{1}{\sigma^2} r e^{-\frac{r^2}{2\sigma^2}}$$

Druhá derivácia, Laplacian Gaussianu je

$$G''(x, y, \sigma) = \frac{1}{\sigma^2} \left(\frac{r^2}{\sigma^2} - 1 \right) e^{-\frac{r^2}{2\sigma^2}}$$

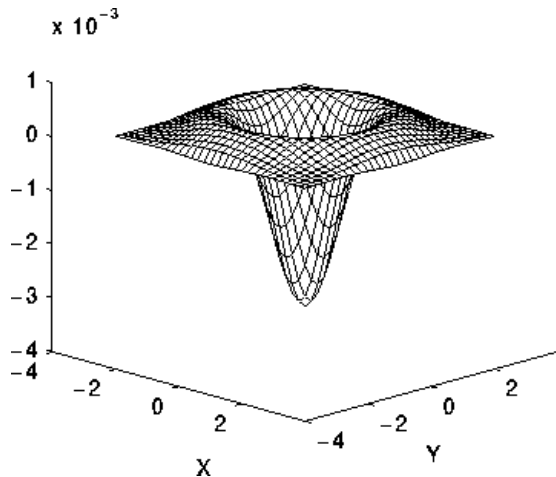
Po spätnom dosadení dostaneme vzťah na výpočet konvolučnej masky:

$$h(x, y) = c \left(\frac{x^2 + y^2 - \sigma^2}{\sigma^4} \right) e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

kde c je normalizovaný koeficient.

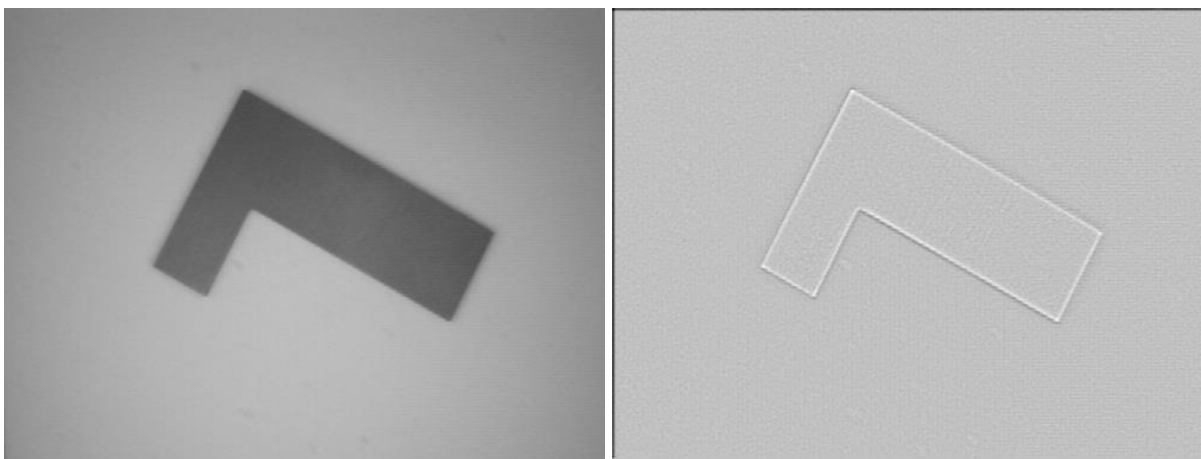
Aproximácia operátora LoG v maske 5x5:

$$h = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$



0	0	3	2	2	2	3	0	0
0	2	3	5	5	5	3	2	0
3	3	5	3	0	3	5	3	3
2	5	3	-12	-23	-12	3	5	2
2	5	0	-23	-40	-23	0	5	2
2	5	3	-12	-23	-12	3	5	2
3	3	5	3	0	3	5	3	3
0	2	3	5	5	5	3	2	0
0	0	3	2	2	2	3	0	0

Example:



Vplyv veľkosti jadra Gaussovho operátora na LoG a teda aj na výsledok prechodu nulou:



$\sigma = 1.0$

$\sigma = 2.0$

$\sigma = 3.0$

3.5.5 Canny detektor

Canny určil tri kritéria, ktoré by mal optimálny hranový detektor spĺňať:

1. Detekčné kritérium, detektor nesmie zabudnúť významnú hranu a na jednu hranu môže byť maximálne jedna odozva.
2. Lokalizačné kritérium, rozdiel medzi skutočnou a nájdenou hranou má byť minimálny.
3. Kritérium jednej odozvy.

Cannyho detektor využíva konvolúciu s dvojrozmerným Gaussianom a deriváciu v smere gradientu. Poskytuje informácie o smere a veľkosti hrany. Nech G je dvojrozmerný Gaussian, (8). Nech G_n je prvá derivácia G v smere gradientu

$$G_n = \frac{\partial G}{\partial n} = n \nabla G$$

kde n je smer gradientu, ktorý dostaneme nasledovne:

$$n = \frac{\nabla(G * f)}{|\nabla(G * f)|}$$

Hranu dostaneme v bode, kde funkcia $G_n * f$ dosiahne lokálne maximum, a druhá derivácia sa rovná nule.

$$\frac{\partial^2}{\partial n^2} G * f = 0$$

Pre silu hrany platí:

$$|G_n * f| = |\nabla(G * f)|$$

Kritérium jednej odozvy sa dosahuje následne prahovaním.

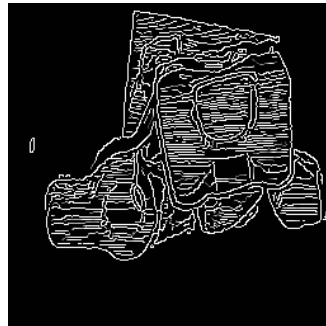
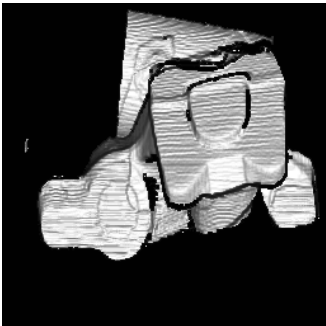
Sumarum:

- varianta zero - crossing, pracuje s maximami f
- navrhnutý ako optimálny hranový detektor
 - tri parametre (δ , $T1$, $T2$)
- V etapách
 - 1 vyhladenie: Gaussov filter (sigma)
 - 2 1. derivácie: Roberts | Sobel --> ridge intensity image
 - 3 v ridge image
- chodíme po hrebeni ($T1$) a mažeme všetko čo je pod hranicou kosodreviny ($T1 - T2$)
- Problémy s “Y” spojeniami

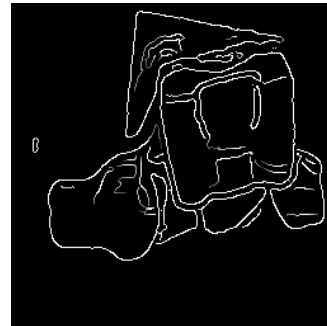
Kritéria hranového detektora:

- detekčné kritérium
- lokalizačné kritérium
- požiadavka jednej odozvy

$$\frac{\partial^2}{\partial n^2} G * g = 0 \quad (4.63)$$



$\sigma = 1.0$



$\sigma = 1.0$

4 Segmentácia

Ďalším krokom je pospájanie extrahovaných bodov do ohraničujúcej oblasti.

Segmentácia - metóda, ktorá rozdelí obraz do častí podľa súvislosti objektov či predmetov v obraze.

podľa použitej techniky - globálna a lokálna

segmentácia prahovaním - body objektu a body pozadia

Metódy segmentácie môžeme rozdeliť na:

- metódy založené na obrazových bodoch, sem patria techniky založené na hranách a prahovanie.
- metódy založené na oblastiach. Ich myšlienka je založená na hľadaní homogénnych oblastí v obraze.

4.1 Prahovanie (Thresholding)

výstup: binárny obraz

T - prahová hodnota (intensity threshold)

$$g(x,y) = \begin{cases} \text{BACKGROUND} & \text{ak } f(x,y) \leq T \\ \text{FOREGROUND} & \text{ak } f(x,y) > T \end{cases}$$

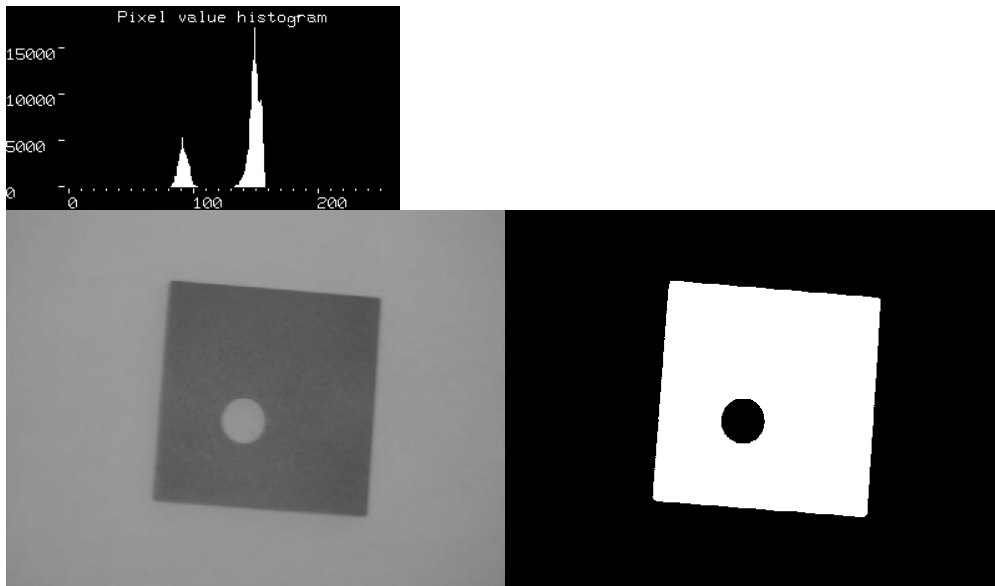
4.1.1 Jednoduché prahovanie

Opiera sa o analýzu histogramu

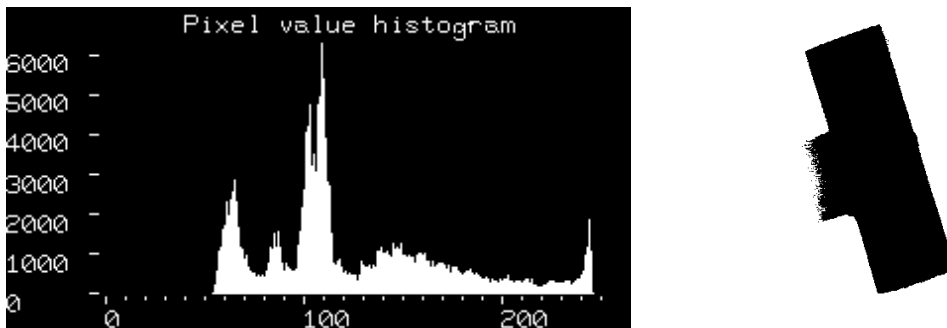
Prahovacie metódy predpokladajú, že objekty v obraze sa dajú odlíšiť od pozadia na základe jasovej hodnoty jednotlivých obrazových bodov. Najprv sa zvolí **prah T**. Podľa tejto prahovej hodnoty sa rozdelia obrazové body na body objektu a body pozadia. Táto metóda sa nazýva **globálne prahovanie**. Prahovaný obraz $g(x,y)$ získame z pôvodného obrazu $f(x,y)$ pomocou vzťahu:

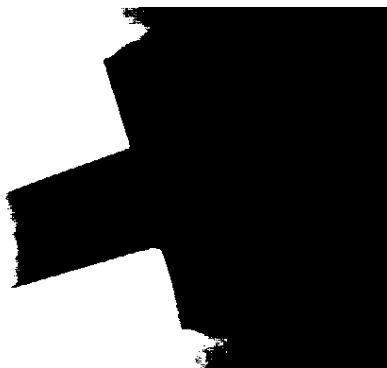
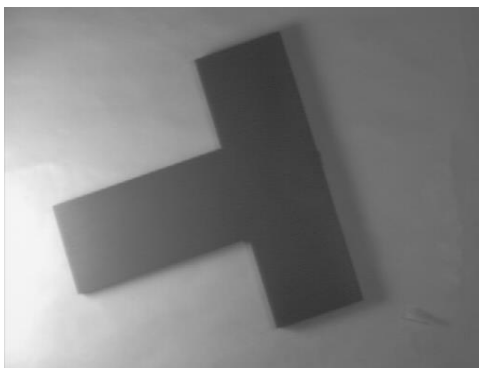
$$g(x,y) = \begin{cases} 255 & \text{ak } f(x,y) > T \\ 0 & \text{ak } f(x,y) \leq T \end{cases}$$

Prikald1:



Prikald2:

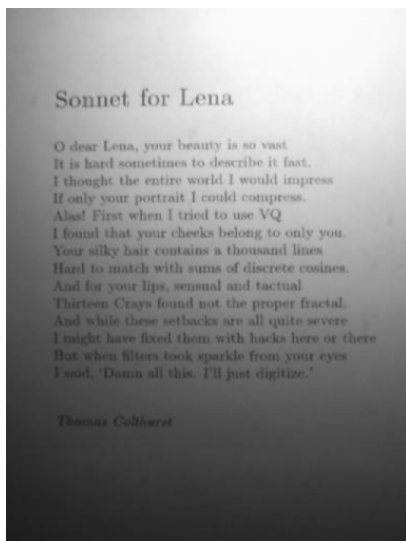




Problém ... a preto použijeme adaptívne prahovanie =>

4.2 Adaptívne prahovanie

- vychádza sa z faktu, že menšie oblasti skor vykazujú rovnomerné osvetlenie ako väčšie
- prah
 - dynamický
 - závislý od pozície spracúvaného pixla
- Chow & Kanenko
 - rozdelenie na prekrývajúce sa časti I
 - výpočet lokálnych prahov T
 - pre každý pixel
 - prah = interpolácia prahov podobrazov
 - výpočtovo náročné
- Local Thresholding
 - v malých okoliach bodu
 - $T = \text{mean} \mid \text{median} \mid (\text{max} - \text{min}) / 2$
 - rýchlejšie ako Chow & Kanenko

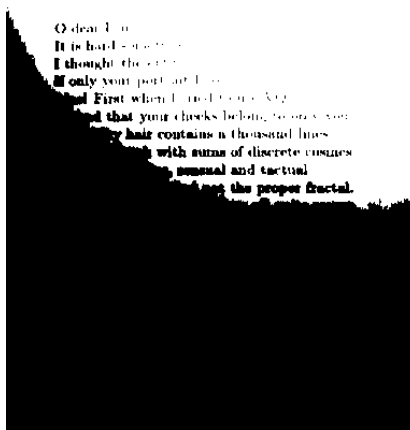


Sonnet for Lena

O dear Lena, your beauty is so vast
 It is hard sometimes to describe it fast.
 I thought the entire world I would impress
 If only your portrait I could compress.
 Alas! First when I tried to use VQ
 I found that your cheeks belong to only you.
 Your silky hair contains a thousand lines
 Hard to match with sums of discrete cosines.
 And for your lips, sensual and tactual
 Thirteen Crays found not the proper fractal.
 And while these setbacks are all quite severe
 I might have fixed them with licks here or there
 But when filters took sparkle from your eyes
 I said, 'Damn all this. I'll just digitize.'

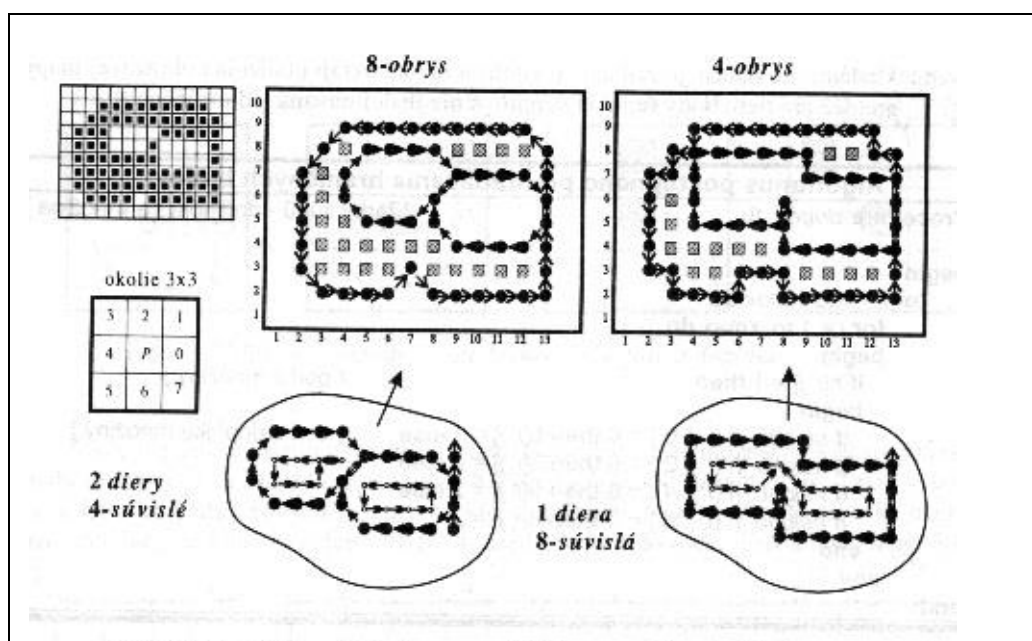
Thomas Calhoun

Sonnet for Lena



4.3 Sledovanie hranice

Predpokladajme, že vstupný obraz je binárny, alebo že oblasti boli identifikované farbením.

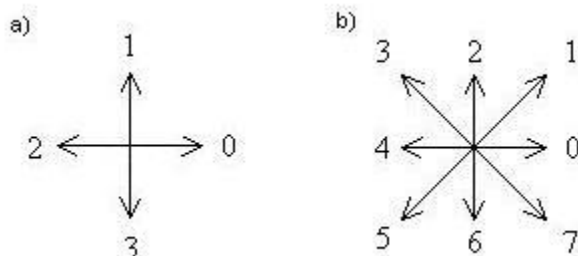


Obr. 8.5 Rozdiel medzi 8-obrysom a 4-obrysom tej istej oblasti

Potom vnútornú hranicu oblasti môžeme získať pomocou nasledujúceho algoritmu:

- 1) Prechádzajme obraz po riadkoch, až kým nenarazíme na obrazový element, ktorý nepatrí pozadiu. Tento bod označíme P, je to prvý bod hranice.
Nech premenná *Smer* uchováva smer od posledného bodu do aktuálneho bodu.
 - i. Pre 4-susednosť: $Smer := 3$
 - ii. Pre 8-susednosť: $smer := 7$.
- 2) Nasledujúci hraničný bod je ten, ktorý je nájdený ako prvý pri prehľadávaní okolia bodu, pričom prehľadávanie začneme v smere
 - i. Pre 4-susednosť
 $(smer + 3) \bmod 4$
 - ii. Pre 8-susednosť:
 $(smer + 7) \bmod 8$ pre párny *smer*
 $(smer + 6) \bmod 8$ pre nepárny *smer*

Nasledujúca hodnota premennej *smer* je určená na základe obr. 4.2.
- 3) Ak $P_n = P_1$ a $P_{n-1} = P_0$ potom krok (4)
Inak opakuj krok (2)
- 4) Opakujeme kroky (1) až (3) dovtedy, kým neprejdeme celý obraz.



Obr. Poradie susedov pri sledovaní hranice.
a) pre 4-susednosť. b) pre 8-susednosť.

Tento algoritmus nájde hranice všetkých oblastí v obraze, ale nenájde diery v oblastiach. Diery môžeme najst' jednoduchou úpravou predchádzajúceho algoritmu.

4.4 Techniky založené na oblastiach

V predošlej kapitole sme hľadali hranice regiónov. Teraz budeme hľadať samotné regióny na základe určitých vlastností.

Definujme **kritérium homogenity P**, ako logický predikát na množine bodov, ktorý je pravdivý, ak maximálny jasový rozptyl

l bodov je menší ako prahová hodnota T. Nech R je množina bodov, potom pre P platí

$$P(R) = \left(\max_{\{x,y\}, \{x1,y1\} \in R} \{ |f(x,y) - f(x1,y1)| \} \leq T \right)$$

4.4.1 Selekcia podľa farby – Narastanie oblasti spájaním bodov do oblasti

Myšlienka metódy je spájanie bodov do oblasti. Na začiatku musíme zvoliť **základné body**, ktoré budú vytvárať jadrá jednotlivých oblastí. Pre každý základný bod budeme testovať jeho susedov, či spolu spĺňajú kritérium homogenity. Ak áno, potom ich spojíme do jednej oblasti. Výsledok závisí na voľbe základných bodov a na voľbe T pre kritérium homogenity.

4.4.2 Spájanie oblastí

Nech R je región obsahujúci celý obraz. Cieľom je rozdeliť R na n podregiónov $R_1, \dots, R_n$ s vlastnosťami:

a) $\bigcup_{i=1}^n R_i = R$

b) R_i je spojitý región, pre každé $i=1, \dots, n$. Región je **spojitý**, ak pre každú dvojicu bodov z regiónu existuje cesta spájajúca tieto body, patriaca celá do regiónu.

c) $R_i \cap R_j = \emptyset$ pre $\forall i, j \quad i \neq j$

d) $P(R_i) = TRUE \quad \forall i$

e) $P(R_i \cup R_j) = FALSE$ pre $i \neq j$

Obraz najprv rozdelíme na menšie disjunktné časti veľkosti 2×2 , 4×4 alebo 8×8 . Susedné oblasti, ktoré spolu spĺňajú kritérium homogenity spojíme do jednej oblasti. Výsledkom tohto postupu je rozdelenie spĺňajúce podmienky a) – e).

Na spájanie oblastí môžeme použiť aj ďalšie heuristiky:

- dve susedné oblasti R_i, R_j spojíme, ak dostatočne veľká časť ich vzájomnej hranice je zložená z nevýznamných hrán. Či je časť vzájomnej hranice dostatočne veľká sa určuje podľa celkovej dĺžky kratšej uzavretej hranice.

$$\frac{W}{\min(l_i, l_j)} > T$$

kde W je počet slabých hrán spoločnej hranice, l_i, l_j sú dĺžky hranice oblastí R_i, R_j , T je prahová hodnota.

- Dve susedné oblasti R_i, R_j spojíme do jednej oblasti, ak dostatočne veľká časť ich vzájomnej hranice je zložená z nevýznamných hrán.

$$\frac{W}{l} > T$$

kde W je počet slabých hrán vzájomnej hranice, l je počet spoločných hrán, T je príslušná prahová hodnota.

- Dve susedné oblasti R_i, R_j spojíme, ak počet nevýznamných hrán ich vzájomnej hranice presahuje prahovú hodnotu T.

$$W > T$$

Hrana medzi dvoma bodmi obrazu je **nevýznamná hrana**, ak rozdiel v jasových hodnotách je dostatočne malý.

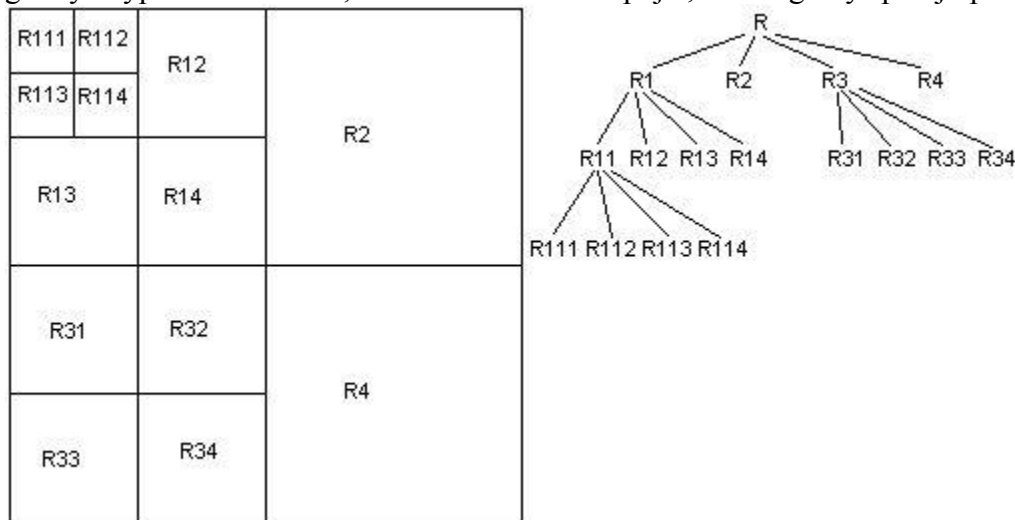
$$|f(A) - f(B)| < T \quad \text{kde } A, B \text{ sú susedné body.}$$

4.4.3 Rozdeľovanie oblasti

Rozdeľovanie oblasti je principiálne opačný prístup k segmentácii ako spájanie oblastí. Obraz tvorí jeden región, ktorý zvyčajne nespĺňa podmienky a) – e). Preto ho rozdelíme na menšie časti, až kým podmienky nebudú splnené.

4.4.4 Rozdeľovanie a spájanie oblastí

Metóda rozdeľovania a spájania oblastí zachováva dobré vlastnosti predošlých techník. Pri tejto metóde využívame pyramídovú dátovú štruktúru. Ak je oblasť v danej úrovni pyramídy nehomogénna, tak sa rozdelí na štyri podoblasti, na elementy nižšej úrovne. Výsledkom rozdeľovania bude kvadrantový strom, v ktorom každý list reprezentuje homogénnu oblasť. V druhej časti algoritmu spojíme dve susedné oblasti, ak spolu spĺňajú kritérium homogenity. Výpočet ukončíme, keď už nemáme čo spojiť, teda regióny spĺňajú podmienky a) – e).



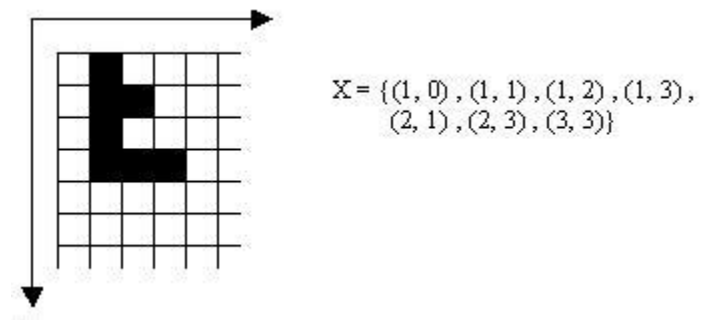
Kvadrantový strom.

Poznámka: Pri implementácii týchto algoritmov výsledok závisí aj od toho, kde sme začali spájanie. Napríklad výsledok spájania začínajúce v ľavom hornom rohu sa môže líšiť od výsledku spájania začínajúce v pravom dolnom rohu.

5 Morfológia

Matematická morfológia predpokladá, že obraz sa dá modelovať pomocou bodových množín. Definičným oborom pre popis dvojrozmerných útvarov je dvojrozmerný euklidovský priestor E^2 a systém jeho podmnožín. Základným prvkom pre binárnu matematickú morfológiu je usporiadaná dvojica celých čísel, pre šedotónové obrazy usporiadaná trojica.

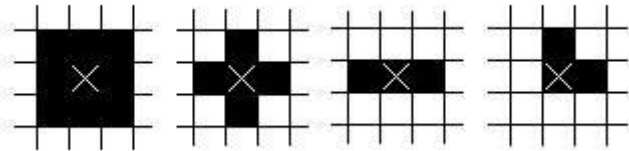
Binárny obraz môžeme vyjadriť ako 2D bodovú množinu. Množina X bude obsahovať body s hodnotou 1 – body objektu, X^c body s hodnotou 0 – body pozadia.



Príklad bodovej množiny.

Morfologická transformácia Ψ je relácia medzi bodovou množinou X a štruktúrnym elementom B . **Štruktúrny element** je tiež bodová množina, ale skladá sa z menšieho počtu bodov. Obsahuje jeden význačný bod O , ktorý sa nazýva reprezentatívny bod.

Typické štruktúrne elementy:

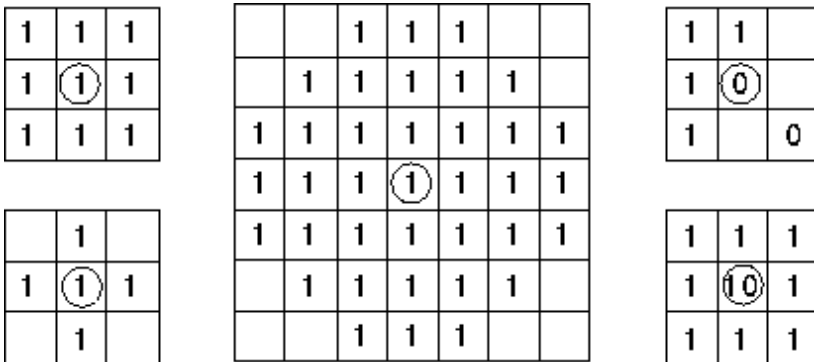


Typické štruktúrne elementy.

Translácia bodovej množiny X o vektor h sa označuje X_h a je definovaná vzťahom:

$$X_h = \{p \in E^2, p = x + h, x \in X\}$$

- detekcia hrán
- odstránenie šumu
- zvýrazňovanie
- segmentácia

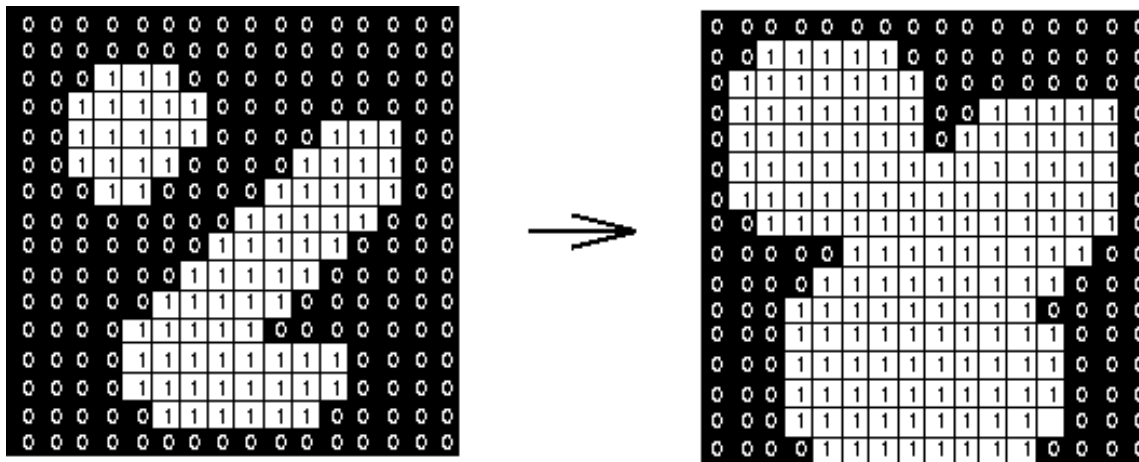


Typy morfologických operácií:

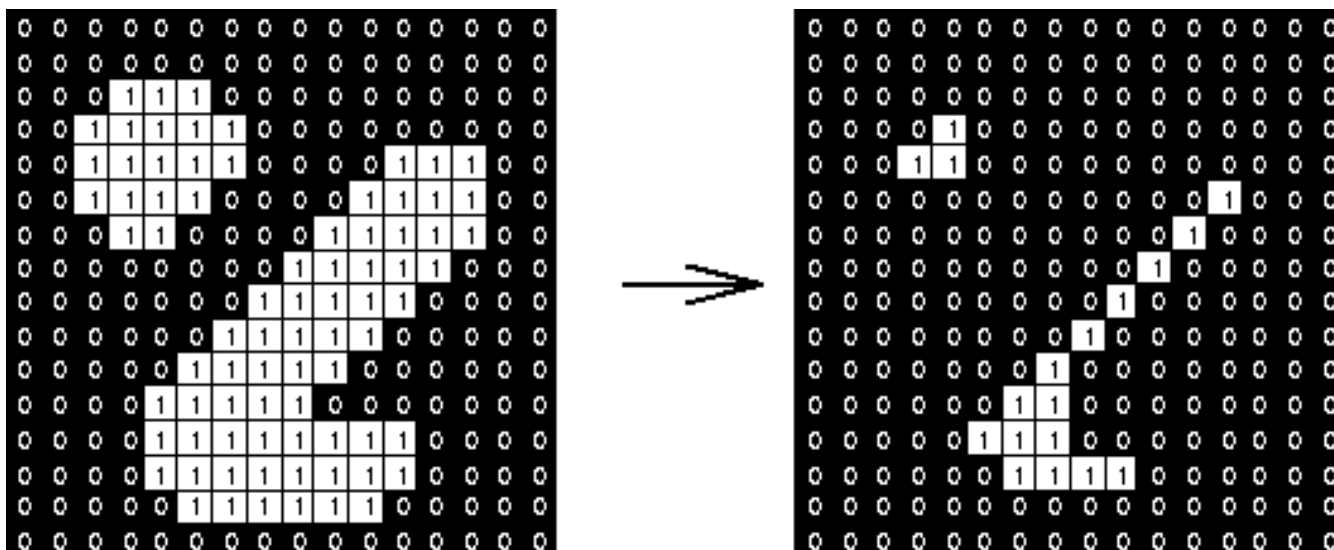
- Dilatácia
- Erózia
- Otvorenie
- Uzavretie
- Hit&Miss
- Thinning
- Thickening
- Skelet

5.1 Dilatácia

$$X \oplus B = \bigcup X_b = \{x \mid B_x \cap X \neq \emptyset\}$$



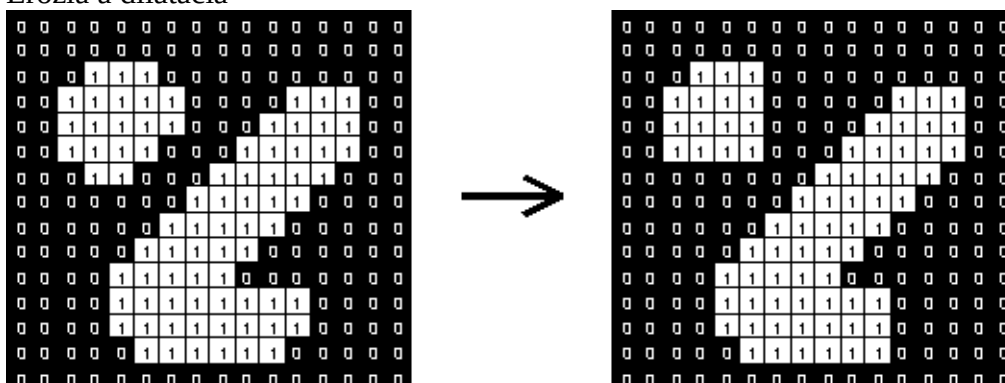
5.2 Erózia





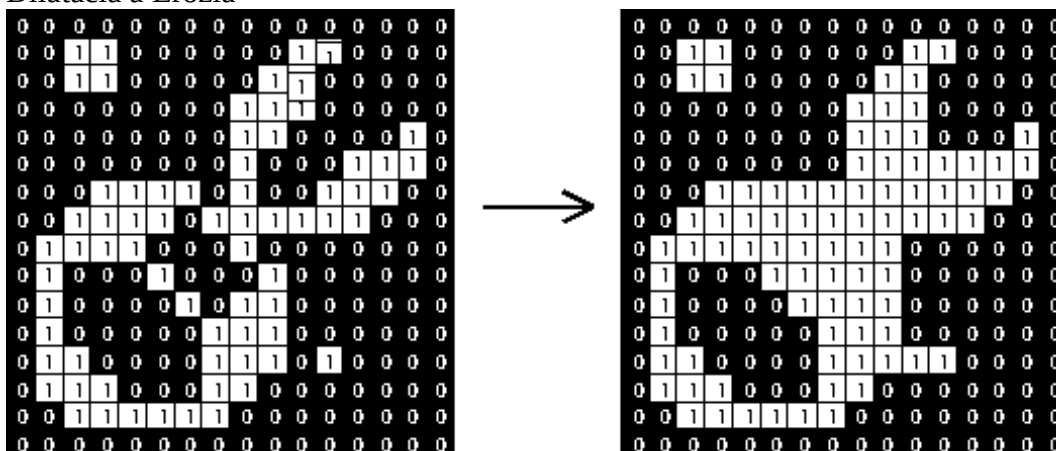
5.3 Otvorenie

Erozia a dilatácia

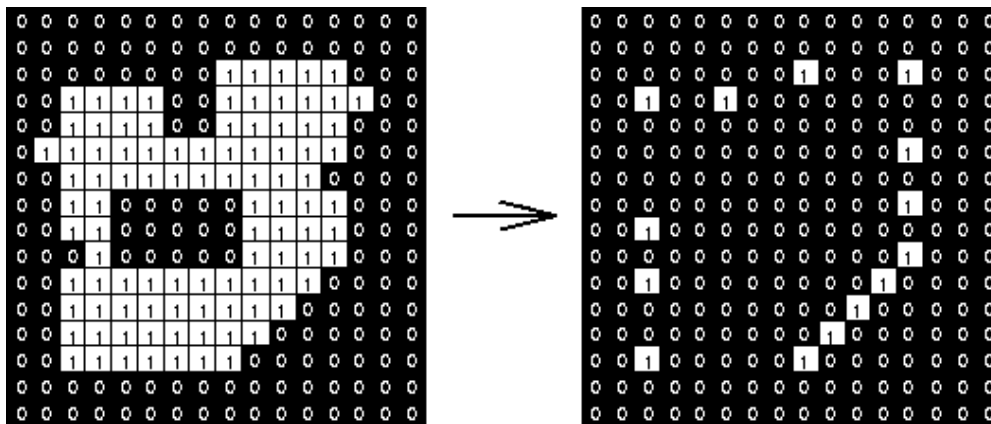


5.4 Uzavretie

Dilatácia a Erozia



5.5 Hit&Miss



	1	
0	1	1
0	0	

	1	
1	1	0
	0	0

	0	0
1	1	0
	1	

0	0	
0	1	1
	1	

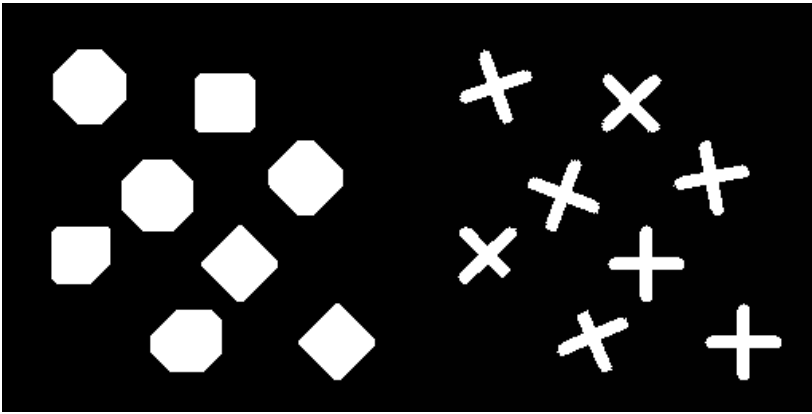
5.6 Thinning



$$\text{thin}(I, J) = I - \text{hit-and-miss}(I, J)$$

$$X - Y = X \cap \text{NOT } Y$$

5.7 Thickening

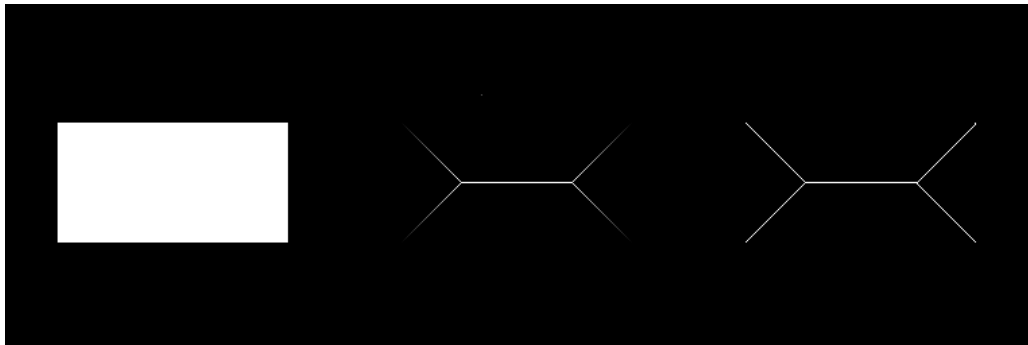


$$\text{thicken}(I, J) = I \cup \text{hit-and-miss}(I, J)$$

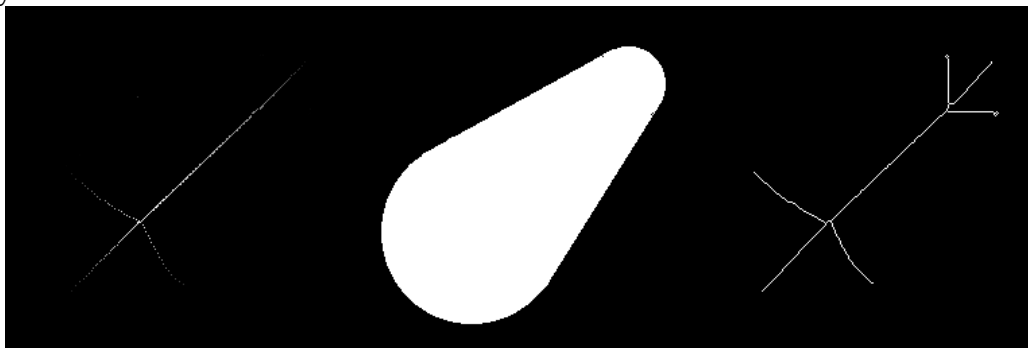
5.8 Skelet

Skelet = kostra množiny.

Skeletom množiny rozumieme jednorozmerné čiary, topologicky ekvivalentné so zadanou množinou a zachovávajúce jej geometrickú podstatu. Všeobecný postup týchto algoritmov je podobný ako pri morfologických transformáciách a spočíva v **stenčovaní zadanej množiny**.



Praktické využitie skeletu nachádzame hlavne pri rozpoznávaní obrazcov a pri kódovaní a prenose veľkého objemu grafických dát.



Základné pojmy:

1. **Zachovanie súvislej množiny.** Ak je množina súvislá, potom aj skelet musí byť súvislý.
2. **Zachovanie koncových bodov.** V množine treba čo najskôr detekovať koncové body tak, aby sa neskrátili charakteristické línie.
3. **Zachovanie strednej línie.** Pri stenčovaní je treba zachovávať približne stred t.j. stenčovať symetricky.

Len prvá požiadavka je topologického charakteru, ostatné dve sú geometrické.

Skeletový bod:

AK platia nasledujúce podmienky:

1. Bod A sa vyskytuje v postupnosti obrysu dva razy, čiže je viacnásobný
2. Bod A nemá žiadnych susedov z vnútra množiny t.j. má len obrysových susedov a susedov z doplnku
3. Má aspoň jedného priameho suseda, ktorý je obrysovým bodom, ale nie je jeho obrysovým susedom.

Picture y Ferko-Ruyickž str. 133

Algoritmy skeletovania môžeme rozdeliť do dvoch skupín.

1. Algoritmy využívajúce obrys môžeme sledovať dve etapy: v prvej etape sa zostrojí obrys a v druhej etape sa hľadajú vhodné podmienky na obrysové body, ktoré je možné vylúčiť. (Klasický alg. Alebo Kwocka)
2. Do druhej skupiny algoritmov patria tie, ktoré sa snažia priamo vyhľadať skeletované body, pričom sa ostatné body postupne vylučujú. (Napr. Alg. Zhanga-Suena)

5.8.1 Klasický algoritmus skeletovania

5.8.2 Algoritmus Zhang-Suena

5.8.3 Algoritmus Kwocka

6 Reprezentácia Tvaru

- Run-Length Encoding (RLE)
- Quadtree
- Chain Code

6.1 Run-Length Encoding (RLE)

a) Gray value image

Original line (hex): 12 12 12 20 20 20 20 25 27 25 20 20 20 20 20

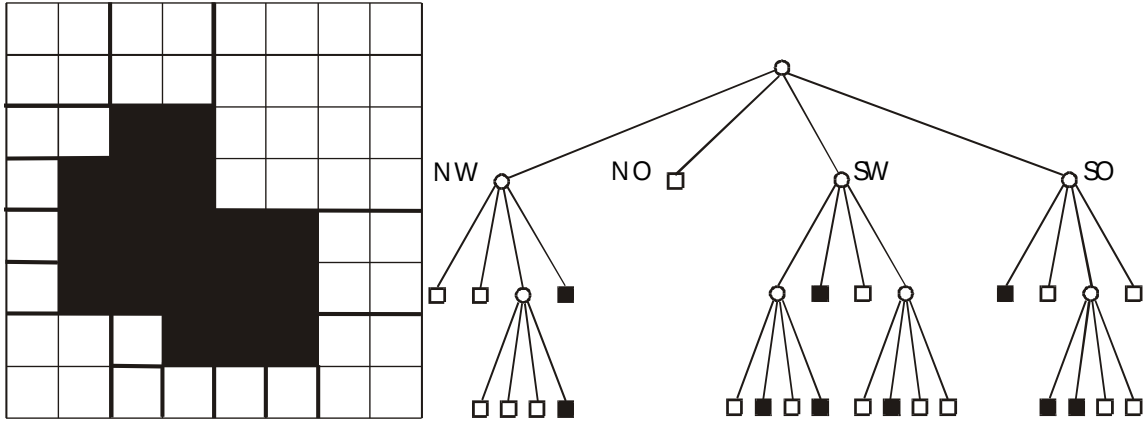
Code (hex): 82 12 83 20 2 25 27 25 85 20

b) Binary image

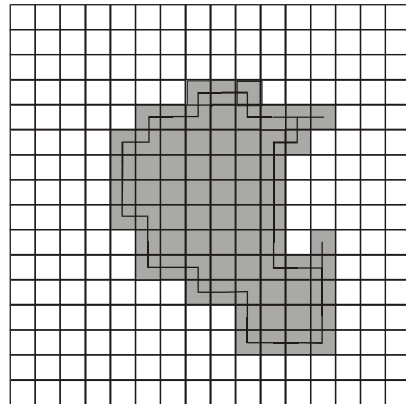
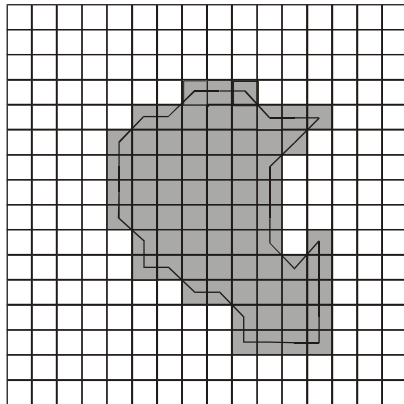
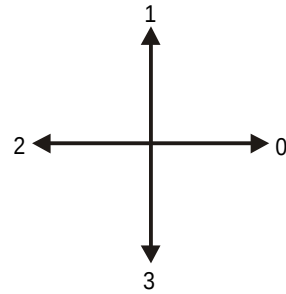
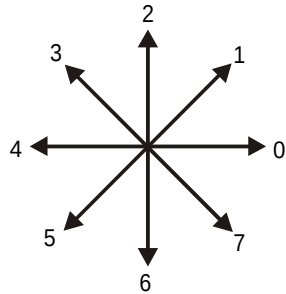
Original line (hex): 1 1 1 1 1 1 0 0 0 1 1 1 0 0 1 0 0 0 0 0 1 1 1 1 1 1 1

Code (hex) 0 6 3 3 2 1 5 8

6.2 Quadtree



6.3 Chain Code



Počítačová grafika 1, príklady otázok na písomnú skúšku, Andrej Ferko

Pozn.

Ak je to vhodné, v každej odpovedi načrtnite ilustračný obrázok, má cenu 0-3 body.

1. Charakterizujte referenčný model počítačovej grafiky.
2. Aké poznáte jednoduché/ľahké metódy modelovania v počítačovej grafike?
3. Zadaťte niekoľko vhodných primitívnych objektov (guľa?, torus?, kužeľ?...) a vytvorte z nich geometrický model lavičky v reprezentácii CSG. Nezabudnite konkrétne uviesť všetky súradnice význačných bodov modelu.
4. Charakterizujte reprezentáciu farby v počítačovej grafike.
5. Zadaťte okno a záber parametrami funkcií SET WINDOW a SET VIEWPORT tak, aby matica transformácie mala na hlavnej diagonále hodnoty (3, 2, 1) a v spodnom riadku (128, 64, 1). Určite inverznú maticu a uplatnite ju na súradnice bodu v strede záberu.
6. Vymenujte a špecifikujte 6 logických vstupných zariadení (lokátor...), najmä aké poskytujú vstupné dátové záznamy.
7. Vymenujte a špecifikujte parametre a atribúty troch grafických výstupných funkcií, dvoch štandardných a jednej Vašej, novej, celkom originálnej, no akademicky prijateľnej.
8. Architektúra multimediálneho systému.
9. Opíšte slovom i obrazom princípy troch vizualizácií (vizualizačných metafor) na príklade resp. príkladoch Vami zvolených netriviálnych dát (počasie, medicína, známky zo skúšok...). Hodnotenie: 6 minút/bodov.
10. Maticová otázka: Zo svojho mena určite tri body v rovine a trojuholník ABC takto: súčet počtu znakov prvého, súčet počtu znakov posledného mena, celkový súčet, napr. Eva Nováková >> 3, 8, 11 >> A[3,8], B[8,11], C[11,3]. Okolo tohto trojuholníka zvolte ako oblasť záujmu obdĺžnikové okno (funkcia SET WINDOW) tak, aby v ňom ležal celý trojuholník. Potom nastavte záber (funkcia SET VIEWPORT) ako obdĺžnik v jednotkovom štvorci, do ktorého sa má okno transformovať. Určite maticu transformácie a jej inverznú maticu. Tú aplikujte na ťažisko transformovaného trojuholníka. Skúška správnosti: ťažisko zo súradníc záberu sa má inverzne transformovať na ťažisko trojuholníka ABC vo svetových súradniciach okna. Hodnotenie: 14 minút/bodov.
11. Otázka z portálu pg.netgraphics.sk číslo 5 (id=17 typ=1 náročnosť=3)

: Aké je analytické vyjadrenie priamky, na ktorej ležia body [15,4], [11,16]?

- $y=5x-16$
- $y=-3x+49$
- $y=-x+49$
- $y=x+32$

12. Výpočet: Vyčísľte dva rôzne (nie krajné) body na Bézierovej krivke pomocou algoritmu de Casteljau. Riadiace body zvolte tak, aby krivka aproximovala hornú polkružnicu, ktorej stred a polomer zadajte pomocou prvých troch písmen svojho krstného mena, prepočítaných na ich poradové čísla v rámci abecedy. Napr. ABC >> 123, stred [1, 2], polomer 3.

13. Zadajte okno a záber parametrami funkcií SET WINDOW a SET VIEWPORT. Určite maticu transformácie okna na záber a k nej inverznú maticu. Transformujte súradnice pravého horného rohu okna (window) zo svetových súradníc a inverzne transformujte súradnice pravého dolného rohu záberu (viewport) do svetových súradníc.

14. Stručne opíšte technológie, ktoré používate vo svojom projekte (napr. HTML, XML, VRML...)

15. Zadajte okno a záber parametrami funkcií SET WINDOW a SET VIEWPORT tak, že ľavý dolný roh okna je v počiatku a ľavý dolný roh záberu je v bode, danom dňom a mesiacom Vášho narodenia. Zvolte pravé horné rohy okna a záberu ľubovoľne. Nakreslite obrázok a určite homogénnu maticu transformácie okna na záber a k nej inverznú maticu. Vypočítajte, kam sa transformuje začiatok súradnicovej sústavy záberu. Obmena: Transformujte súradnice pravého horného rohu okna (window) zo svetových súradníc a inverzne transformujte súradnice pravého dolného rohu záberu (viewport) do svetových súradníc.

16. Vedomostná otázka. Grafické výstupné prvky, parametre a atribúty, vstup a výstup, možné chyby. Uvedte aspoň tri. Každá špecifikácia sa hodnotí 5/3 bodu.

17. Určite maticu transformácie, ktorá zobrazí jednotkový štvorec (0.0,0.0) ... (1.0,1.0) na obdĺžnik (a, b) ... (c,d), kde reálne čísla a, b, c, d si zadajte sami, ale nesmú byť ani nulové ani po dvojiciach rovnaké. Určite aj inverznú maticu, napr. použitím rozloženia na základné afinné transformácie. Každá z matic sa hodnotí 5/2 bodu.

18. Otázka z časti TESTY na pg.netgraphics.sk. Odpoveď nielen zakrúžkujte, ale aj vysvetlite výpočtom, úvahou alebo graficky. Správna voľba samotná je za 5 bodov.

19. Opíšte niektorý farebný priestor a vyjadrite v ňom sivú farbu.

20. Vymodelujte čo najjednoduchšou metódou objekt topologicky ekvivalentný toroidu („pneumatike, kúpaciemu kolesu, cédéčku“). Aby bol objekt korektný, nesmie byť nulovej hrúbky (elipsa, kružnica, hranica obdĺžnika). Vo výslednej reprezentácii Vášho modelu zvoleného objektu musia byť explicitne hodnotami určené nielen súradnice význačných bodov (koncové body, stredy objektov... (x,y,z)), ale aj počiatok a osi súradnicového systému, pomocou ktorého súradnice bodov vyjadrujete. Nestrácajte čas s farbou ani osvetlením, iba čo najpresnejšie určite geometrický model a jeho dátovú reprezentáciu.

21. Otázka z časti TESTY na pg.netgraphics.sk. Pri konštrukcii Voronoiovhho diagramu

- hľadáme extrémálne body v danej množine.
- hľadáme body, ktoré ležia v konvexnom obale všetkých bodov z množiny.
- hľadáme množinu bodov, ktoré sú od P ďalej ako od ľubovoľného iného bodu.
- hľadáme množinu bodov, ktoré nie sú od P ďalej ako od ľubovoľného iného bodu.

22. Zadaťte okno a záber parametrami funkcií SET WINDOW a SET VIEWPORT tak, aby nimi určená matica transformácie mala na hlavnej diagonále hodnoty $(3.0, 2.0, 1.0)$ a v spodnom riadku aspoň dve nenulové hodnoty, napr. $(128.7, 64.33, 1.0)$. Určite inverznú maticu, napr. použitím rozloženia na matice pre základné afinné transformácie alebo výpočtom a uplatnite ju na súradnice bodu v strede záberu.

23. Otázka z časti TESTY na pg.netgraphics.sk. Prvky nasledujúcej matice sú buď nuly (0) alebo nenulové reálne čísla. Návod: odpovedajte buď Vám známej použitím teórie alebo výpočtom: uplatnite transformáciu s vhodne zvolenými (rôznymi) konkrétnymi hodnotami na jeden až tri body a odpozorujte jej účinok.

11. Matica na obrázku reprezentuje transformáciu:

- otočenie so stredom v ľubovoľnom pevnom bode (x,y)
- zmenu mierky so stredom v ľubovoľnom pevnom bode (x,y)
- otočenie okolo počiatku
- posunutie so stredom v ľubovoľnom pevnom bode (x,y)

24. Zapište algoritmus de Casteljeau na vyčíslenie Bézierovej krivky.

25. Vyčísľte algoritmom de Casteljeau bod na Bézierovej krivke.

26. Bilineárna interpolácia.

27. Výpočet bilineárnej interpolácie pre šedotónový obrázok, napr. ťažisko rastrového trojuholníka, ktorého intenzity pre vrcholy sú dané dĺžkami, rozdielmi a súčtami počtu znakov Vášho mena.

28. Zo svojho mena určite tri body v rovine a trojuholník ABC takto: súčet počtu znakov prvého, súčet počtu znakov posledného mena, celkový súčet, napr. Eva Novakova >> 3, 8, 11 >> A[3,8], B[8,11], C[11,3]. Určite maticu transformácie, ktorá zobrazí jednotkový štvorec (0.0,0.0) ... (1.0,1.0), t.j. okno na záber, t.j. obdĺžnik (a, b) ... (c,d), kde čísla a, b, c, d si zadajte sami, ale nesmú byť ani nulové ani po dvojiciach rovnaké a záber musí obsahovať trojuholník ABC. Určite aj inverznú maticu, napr. použitím rozloženia na základné afinné transformácie a transformáciu niektorého z bodov A, B, C do okna. Každá z matic znamená 3 body, obrázok 3 body, výpočet a správny výsledok 6 bodov.
29. Lokálne osvetľovacie modely. Za menej bodov pre difúzny odraz, za viac aj pre zrkadlou zložku (Phong).
30. Grafický systém, cieľ, požiadavky, skupiny funkcií.
31. Základné funkcie na hierarchiu obrazu.
32. Napíšte všetko, čo viete o použití matíc v počítačovej grafike.
33. Počítačová animácia. Definícia a jeden konkrétny príklad v 2D, napr. empirický model pre po schodoch skackajúcu loptičku, ponárajúce sa kyvadlo, do vetra rozsypané pukance... Alebo: Dané sú koncové body úsečky v euklidovskej rovine (x1, y1), (x2, y2), dané dĺžkami, rozdielmi a súčtami počtu znakov Vášho mena a treba nájsť jej zobrazenie pomocou aspoň dvoch zložených matíc. Uvedte princíp práce algoritmu, nemusíte ho rozpisovať do elementárnych krokov (násobenie, sčítanie). Bonus: napíšte pseudokód pre animáciu v reálnom čase.
34. Algoritmus vyplnania oblasti.
35. Vyplňte oblasť z časti 5.3. algoritmom Flood Fill.
35. Vyplňte oblasť z časti 5.3. algoritmom Parity Fill.
36. Postup návrhu loga podľa Glassnera.
37. Navrh stránky podľa J.J.Garretta.

Odkazy

<http://www.sccg.sk/ferko/PGASO2012-bookmarks.pdf>
<http://dip.sccg.sk/>

Aplety a testy, z ktorých je v každom teste vždy jedna otázka.
<http://www.netgraphics.sk/>