Számítógépes képfeldolgozás

Képszűrés

Tófalvi Tamás

Eötvös Loránd Tudományegyetem, Informatikai Kar
tofi@inf.elte.hu

Tartalom

1. Képszűrés
2. Gyakran használt szűrők
3. Gyors és adaptív szűrők

Tartalom

1 Képszűrés
- Korreláció és konvolúció
- A zajszűrés alapjai

2 Gyakran használt szűrők

3 Gyors és adaptív szűrők

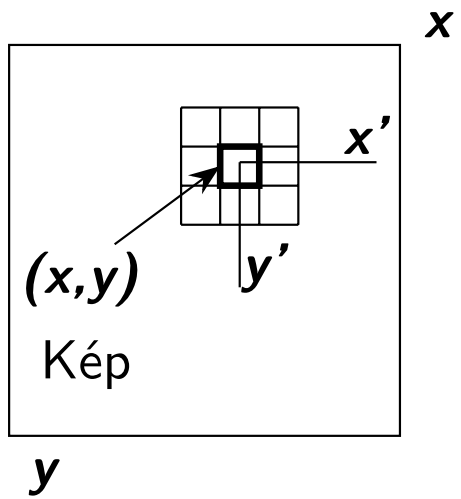
Szomszédsági operátorok

- Az (x, y) pontbeli kimenetet (x, y) szomszédsága határozza meg:

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

  - $f(x, y)$ a bemeneti, $g(x, y)$ a kimeneti kép
  - T az f-en ható operátor, (x, y) szomszédságán értelmezve

- Ablakos mintavételezés: feltesszük a pixelek közötti *lokális* összefüggést
  - a korreláció a távolsággal csökken
  - periodikus mintázatokra ez nem igaz



3×3-as ablak az (x, y) pontban
x', y': lokális koord.

Nem rekurzív és rekurzív operátorok

- **Nem rekurzív** szomszédsági operátor
  - a kimenet csak a bemeneti képtől függ
  - a kimenet elkülönül a bemenettől: a bemenet a művelet során *nem* módosul
  - a hatás a szomszédságra korlátozódik
- **Rekurzív** szomszédsági operátor
  - a kimenet részben a korábban előállított kimeneti értékektől függ
  - a kimenet *nem* különül el a bemenettől: a bemenet a művelet során módosul
  - a hatás a szomszédságon túlra terjed
  - hasznos, de sokkal bonyolultabb
- Csak nem rekurzív operátorokkal foglalkozunk

Általános nem rekurzív szomszédsági operátor

$$g(x,y) = \phi[x, y, f(x',y') : (x',y') \in N(x,y)]$$

- $f(x,y)$ a bemeneti, $g(x,y)$ a kimeneti kép
- $N(x,y)$ az (x,y) pont szomszédsága
- (x',y') a $N(x,y)$-beli *lokális koordináták*
- $f(x',y') : (x',y') \in N(x,y)$ az $N(x,y)$-beli pixelértékek listája
  - $N(x,y)$-t adott sorrendben járjuk be
  - minden $(x',y') \in N(x,y)$-re vesszük $f(x',y')$-t, és a listába tesszük
- ϕ függhet a bemeneti képbeli (x,y) pozíciótól
  - az $N(x,y)$ szomszédság függhet (x,y)-tól
  - a kimeneti értéket számoló eljárás is függhet (x,y)-tól
- ϕ lehet nemlineáris
  - A lineáris operátor: $A(\alpha p + \beta q) = \alpha Ap + \beta Aq$

Korreláció

A lineáris, eltolásinvariáns operátor a bemeneti pixelek lineáris kombinációja: az f kép és a w maszk **keresztkorrelációja**

$$g(x,y) = (f \otimes w)(x,y) \doteq \sum_{\substack{(x',y')\in W \\ (x+x',y+y')\in F}} f(x+x',y+y') \cdot w(x',y')$$

- W az ablakbeli, F a képbeli pozíciók halmaza
- a W szomszédság és a $w(x',y')$ súlyok eltolásinvariánsak
- w neve: a súlyok *kernele* vagy *maszkja*

Konvolúció

Az f kép **konvolúciója** a w kernellel:

$$g(x,y) = (f * w)(x,y) \doteq \sum_{\substack{(x',y')\in W \\ (x-x',y-y')\in F}} f(x-x',y-y') \cdot w(x',y')$$

- A W ablakot fordított sorrendben járjuk be.
- Szimmetrikus maszkokkal fogunk dolgozni.
  $\Rightarrow$ nincs különbség korreláció és konvolúció között

A konvolúció alaptulajdonságai

1. A korreláció a tükrözött maszkkal vett konvolúció: $f \otimes w = f * w^{\sim}$
   - $w^{\sim}(x, y) \doteq w(-x, -y)$ a w tükrözöttje
2. Kommutatív: $w * v = v * w$ (a sorrend tetszőleges)
3. Asszociatív: $(f * w) * v = f * (w * v)$
4. Disztributív: $(f + g) * w = f * w + g * w$
5. Homogén: $(\alpha f) * w = \alpha(f * w)$ tetszőleges α konstansra
6. A kompozíció tükrözése: $(w * v)^{\sim} = w^{\sim} * v^{\sim}$

- f és g képek, w és v maszkok
- $w * v$: a w maszkot képként kezeljük, és v-vel konvolváljuk
  - az eredmény egy nagyobb maszk
  - az asszociativitással gyorsítható a szűrés

Példák: 3×3-as átlagoló szűrők

$$\frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix} \quad \frac{1}{6}\begin{bmatrix}0 & 1 & 0\\ 1 & 2 & 1\\ 0 & 1 & 0\end{bmatrix} \quad \frac{1}{8}\begin{bmatrix}0 & 1 & 0\\ 1 & 4 & 1\\ 0 & 1 & 0\end{bmatrix} \quad \frac{1}{16}\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}$$

doboz szűrő | átlagoló 1 | átlagoló 2 | átlagoló 3

- *Doboz szűrő*: egyenletes súlyú átlagoló szűrő
- Egyébként a súlyok a középponttól való távolsággal csökkennek
  - az eredményhez való hozzájárulás a távolsággal csökken
- A normáló tényezők a maszk együtthatóinak összegei
  - kimeneti tartomány: [minérték, maxérték]
- A szűrő mérete általában páratlan

5×5-ös átlagoló szűrők 1/2

$$\begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 6 & 4 & 2 \\ 3 & 6 & 9 & 6 & 3 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- Ez az 5×5-ös szűrő két 3×3-as szűrő konvolúciója.
- Gyorsabb megvalósítást tesz lehetővé:
  - 5×5-ös szűrő: $5 \times 5 = 25$ szorzás, 24 összeadás
  - két 3×3-as szűrő: $2 \times 3 \times 3 = 18$ szorzás, $2 \times 8 = 16$ összeadás

5×5-ös átlagoló szűrők 2/2

$$\frac{1}{100}\begin{bmatrix} 0 & 3 & 4 & 3 & 0 \\ 3 & 6 & 7 & 6 & 3 \\ 4 & 7 & \mathbf{8} & 7 & 4 \\ 3 & 6 & 7 & 6 & 3 \\ 0 & 3 & 4 & 3 & 0 \end{bmatrix}$$

- Ez a szűrő a

$$w(r) = 8 - r^2$$

függvény diszkrét változata, ahol $r = \sqrt{x^2 + y^2}$ a középponttól (**8**) mért távolság.
  - például $4 = 8 - 2^2$
  - figyeljük meg a forgásszimmetriát

Konvolúciós szűrő alkalmazása: számpélda 1/2

$$\frac{1}{16}\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix} * \begin{array}{|cccccc|}\hline \mathbf{3} & \mathbf{2} & \mathbf{8} & 7 & 8 & 8\\ \mathbf{2} & \mathbf{2} & \mathbf{7} & 8 & 7 & 7\\ \mathbf{2} & \mathbf{3} & \mathbf{9} & 9 & 8 & 8\\ 1 & 2 & 9 & 9 & 7 & 8\\ 2 & 2 & 8 & 8 & 8 & 8\\ 2 & 3 & 7 & 7 & 9 & 7\\ \hline \end{array} = \begin{array}{|cccccc|}\hline - & - & - & - & - & -\\ - & 4 & .. & .. & .. & -\\ - & .. & .. & .. & .. & -\\ - & .. & .. & .. & .. & -\\ - & .. & .. & .. & .. & -\\ - & - & - & - & - & -\\ \hline \end{array}$$

$$\frac{1\cdot 3+2\cdot 2+1\cdot 8+2\cdot 2+4\cdot 2+2\cdot 7+1\cdot 2+2\cdot 3+1\cdot 9}{16} = \frac{58}{16} \approx 4$$

- A szűrő aktuális (kezdő) pozíciója a bemeneti képen félkövéren.
- Az eredmény a kimeneti kép megfelelő pozíciójába kerül.

Konvolúciós szűrő alkalmazása: számpélda 2/2

$$\frac{1}{16}\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix} * \begin{array}{|c|c|c|c|c|c|}\hline 3 & \mathbf{2} & \mathbf{8} & \mathbf{7} & 8 & 8\\ \hline 2 & \mathbf{2} & \mathbf{7} & \mathbf{8} & 7 & 7\\ \hline 2 & \mathbf{3} & \mathbf{9} & \mathbf{9} & 8 & 8\\ \hline 1 & 2 & 9 & 9 & 7 & 8\\ \hline 2 & 2 & 8 & 8 & 8 & 8\\ \hline 2 & 3 & 7 & 7 & 9 & 7\\ \hline\end{array} = \begin{array}{|c|c|c|c|c|c|}\hline - & - & - & - & - & -\\ \hline - & 4 & 6 & .. & .. & -\\ \hline - & .. & .. & .. & .. & -\\ \hline - & .. & .. & .. & .. & -\\ \hline - & .. & .. & .. & .. & -\\ \hline - & - & - & - & - & -\\ \hline\end{array}$$

- A szűrő aktuális (következő) pozíciója félkövéren.
- Az eredmény a kimeneti kép megfelelő pozíciójába kerül.
- A bemenet és a kimenet külön mátrix!

Szegélypixelek kezelése

- $D_W \times D_W$-s maszk esetén a szegélysáv szélessége $\lfloor D_W/2 \rfloor$ (páratlan D_W-re)
  - $\Rightarrow$ a sáv a szűrő méretével együtt nő

Lehetőségek:

- Feltöltés nullákkal
  - erős mesterséges éleket okozhat
  - megzavarhatja a szürkeségi normalizálást ($[0, 255]$-re skálázás)
- Feltöltés a kimeneti kép átlagértékével
  - gyengébb mesterséges élek
  - nem befolyásolja a szürkeségi normalizálást
- Feltöltés a legközelebbi kiszámított értékkel
- A bemeneti kép periodikusnak tekintése (mint egy henger), és az eredmény kiszámítása minden pixelre

Zajtípusok

- **Additív, képtől független** (fehér) zaj:

$$g(x,y) = f(x,y) + v(x,y)$$

  - $f(x,y)$ a bemeneti, $g(x,y)$ a kimeneti kép, $v(x,y)$ a zaj
  - tipikus csatorna- (átviteli) zaj
- **Korrelálatlan multiplikatív zaj**:

$$g(x,y) = f(x,y) \cdot v(x,y)$$

  - amplitúdómoduláció (-ingadozás)
  - tipikus a TV-raszter soraira
- **Kvantálási zaj** (hiba):

$$v_{zaj}(x,y) = g_{kvantált}(x,y) - f_{eredeti}(x,y)$$

- **Só-bors, avagy csúcszaj**: pontszerű, korrelálatlan véletlen zaj

Heurisztikus zajszűrés

- A képjavítás gyakran „heurisztikus” képhelyreállítást jelent
  - nem feltételezünk explicit zajmodellt
- Ugyanakkor különböző zajtípusokhoz különböző szűrők illenek a legjobban
  - *átlagoló szűrő* additív, nulla várható értékű zajra
  - *mediánszűrő* só-bors zajra

$\Rightarrow$ A zaj elemzése kívánatos

- A zajos pixelek kis csoportjai könnyebben eltávolíthatók
  - jó becslés a zajmentes értékre, ha a „jó” pixelek vannak többségben az ablakban
  - rossz becslés, ha a zajos értékek dominálnak

Tartalom

1 Képszűrés

2 Gyakran használt szűrők
- Lineáris simító szűrők
- Mediánszűrő
- Laplace-szűrő
- Életlen maszkolás

3 Gyors és adaptív szűrők

Átlagoló szűrő és doboz szűrő

Átlagoló szűrő:

- Térbeli átlagoló (simító) szűrő
- Nemnegatív súlyok, amelyek összege 1

$$0 \leq w_{\text{átl}}(x, y) \leq 1, \quad \sum_{x,y} w_{\text{átl}}(x, y) = 1$$

  - a gyakorlatban egész súlyokat használunk, majd normalizálunk
- A súlyok nem nőnek a szűrő középpontjától való távolsággal:

$$w_{\text{átl}}(x_1, y_1) \leq w_{\text{átl}}(x_2, y_2), \quad \text{ha } x_1^2 + y_1^2 > x_2^2 + y_2^2$$

Doboz szűrő

- Egyenletes súlyú átlagoló szűrő
  - a legegyszerűbb és leggyorsabb átlagoló szűrő
- $(2M+1) \times (2N+1)$ méretű ablakra

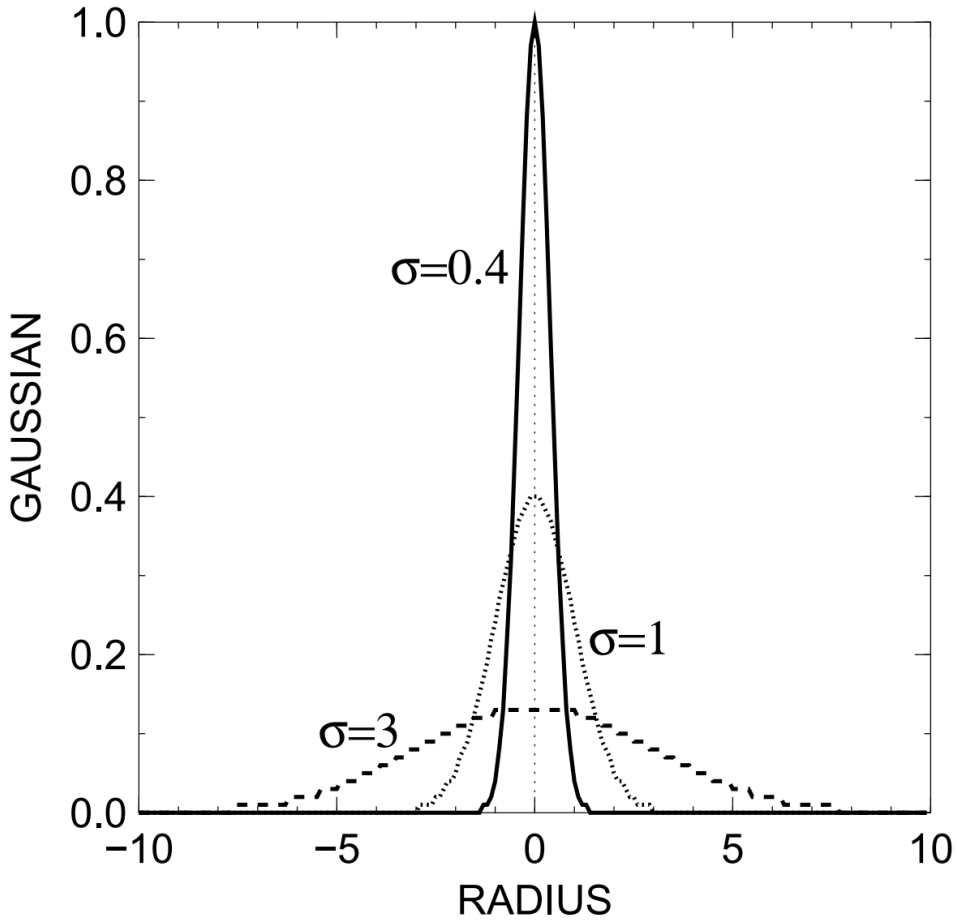
$$g(x,y) = \frac{1}{(2M+1) \times (2N+1)} \sum_{x'=-M}^{M} \sum_{y'=-N}^{N} f(x+x', y+y')$$

Gauss-szűrő 1/2

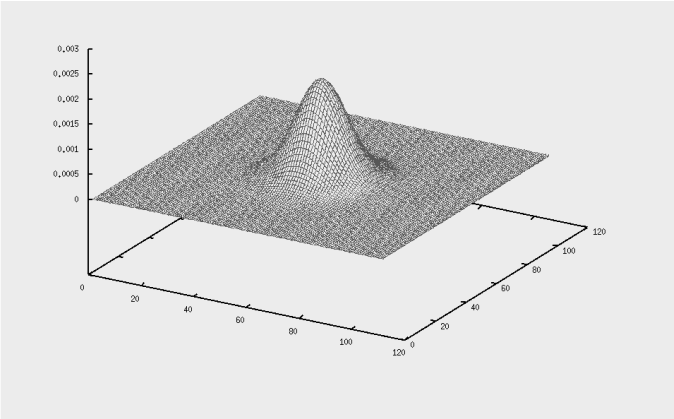
$$w_G(x,y) = \frac{1}{\sum_{(x,y)\in W} e^{-\frac{r^2(x,y)}{2\sigma^2}}} e^{-\frac{r^2(x,y)}{2\sigma^2}}$$

- A súlyokat a 2D Gauss- (normális) eloszlásfüggvény adja.
- $r^2(x,y) = x^2 + y^2$ a maszk középpontjától mért távolság négyzete
  - nem függ a szögtől, csak r-től
  - harang alakú, forgásszimmetrikus
- A σ paraméter szabályozza a szűrő méretét
  - nagyobb $\sigma \Rightarrow$ nagyobb szűrő és erősebb simítás

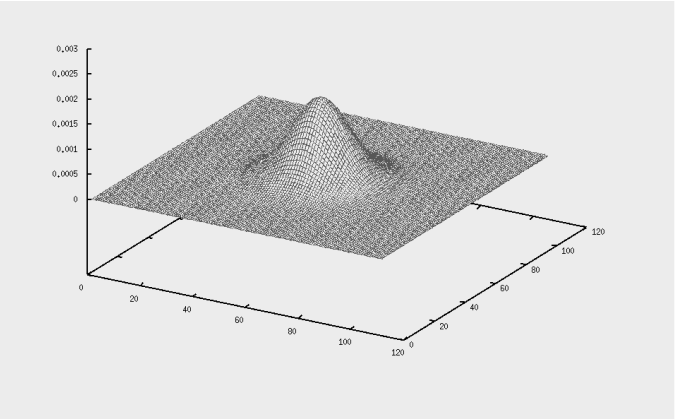
A Gauss-szűrő alakja növekvő σ-ra: 2D



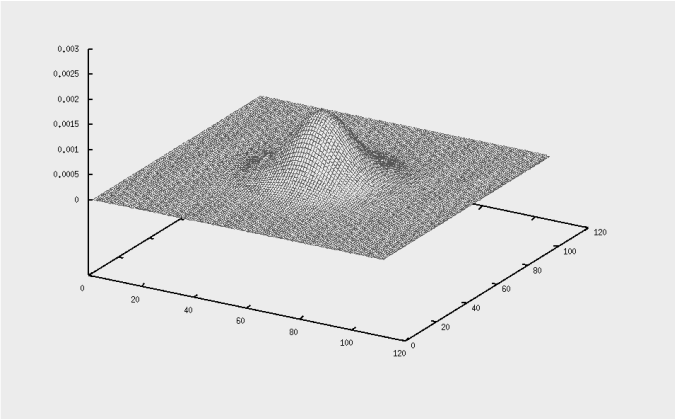
A Gauss-szűrő alakja növekvő σ-ra: 3D



$\sigma = 9$



$\sigma = 10$



$\sigma = 11$

Gauss-szűrő 2/2

$$w_G(x,y) = \frac{1}{\sum_{(x,y)\in W} e^{-\frac{x^2+y^2}{2\sigma^2}}} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Diszkretizáláskor $w_G(r)$-t $r_{max} = k\sigma$-nál vágjuk el.
  - tipikusan $k = 2{,}5$
  - ez a harang térfogatának nagy részét tartalmazza
- A Gauss-szűrő szeparálható:

$$w_G(x,y) = w_G(x) \cdot w_G(y) \Leftarrow \exp(a+b) = \exp(a) \cdot \exp(b)$$

  - gyors megvalósítás: egy 2D szűrő helyett két 1D szűrő
  - $O\left((2r_{max})^2\right)$ művelet 2D-ben, $O(2 \cdot 2r_{max})$ 1D-ben

A simítás felhasználása

- **Zajszűrés**
  - a doboz szűrő csökkenti a nulla várható értékű fehér zajt, mert a pozitív és negatív értékek kioltják egymást
  - nagyobb szűrőméret $\Rightarrow$ nagyobb zajcsökkentés
- Finom részletek eltávolítása
- **Almintavételezés**: áttérés kisebb felbontásra
  - előbb átlagolunk, majd ritkítunk (sorokat/oszlopokat eldobunk)
- A kép **skálatér**-reprezentációjának előállítása
  - Gauss-szűrt képek sorozata növekvő σ-val
  - képelemzés változó részletességgel

A simítás alapvető tulajdonságai

- Csökkenti a kontrasztot és elmossa az éleket
- A kimeneti szürkeségi tartomány a bemenetin belül marad
- Létrehozhat olyan szürkeségi értékeket, amelyek a bemenetben nem szerepeltek
  - bináris kép simítása szürkeárnyalatos képet ad
- A kilógó értékek (outlierek) erősen befolyásolhatják az átlagot

  $\Rightarrow$ az átlag nem robusztus
  - a kilógó értékek hibás értékek, pl. csúcszaj
- A doboz szűrő műveletigénye
  - közvetlen megvalósítás: $O(N \cdot N_W)$
  - futó (run) szűrő megvalósítás: $O(N)$
  - N a kép mérete (területe), N_W az ablak mérete

A nemlineáris mediánszűrő

- A mediánszűrő kimenete az ablakbeli szürkeségi értékek mediánja:
  - rendezzük (rangsoroljuk) a pixeleket szürkeségi érték szerint
  - válasszuk a rendezett sorozat közepén álló értéket
  - az ablak mérete általában páratlan: 3×3, 5×5 stb.
- Példa:
  - a 3×3-as ablak kilenc szürkeségi értéke

$$(1, 1, 3, 2, 5, 4, 4, 12, 11)$$

  - a rendezett sorozat

$$(1, 1, 2, 3, \mathbf{4}, 4, 5, 11, 12)$$

  - a medián értéke **4**

A medián tulajdonságai 1/2

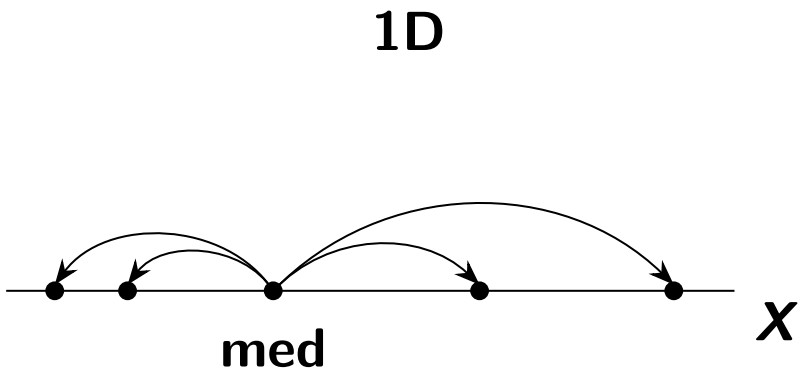
- A medián számítása *nemlineáris* művelet: két P és Q sorozatra

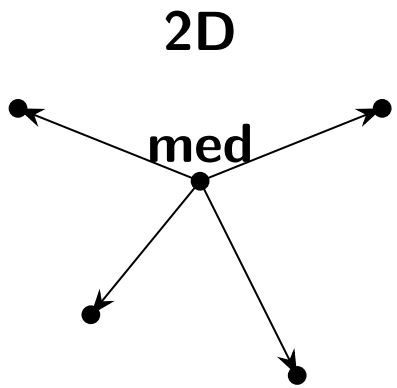
$$Med(\alpha P) = \alpha Med(P) \quad \text{de} \quad Med(P+Q) \neq Med(P)+Med(Q)$$

- A medián kiválasztása *szavazási eljárásnak* tekinthető
  - a rendezés során minden pixel szavaz egy szürkeségi értékre
  - a medián a többségből, a „középről” kerül ki
  - a szélsőséges értékeket elutasítjuk, mint a többséghez nem tartozókat

A medián tulajdonságai 2/2

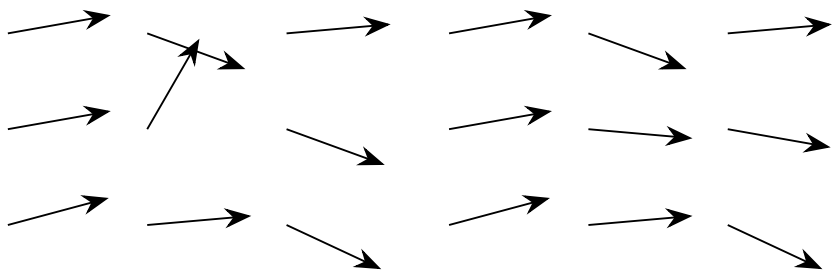
- A medián *robusztus statisztika*
  - a kilógó értékek nem torzítják az eredményt
  - az *összeomlási pont* (breakdown point) ott van, ahol a kilógó értékek aránya eléri az 50%-ot
- Tekintsük a számokat az X tengely pontjainak. A mediántól a többi pontig mért távolságok összege minimális minden 1D ponthalmazra
  - más szóval a medián a halmaz *legbelső* pontja
  - ez a tulajdonság ekvivalens a medián definíciójával
  - ezzel terjeszthető ki a medián magasabb dimenzióra, **vektorokra**





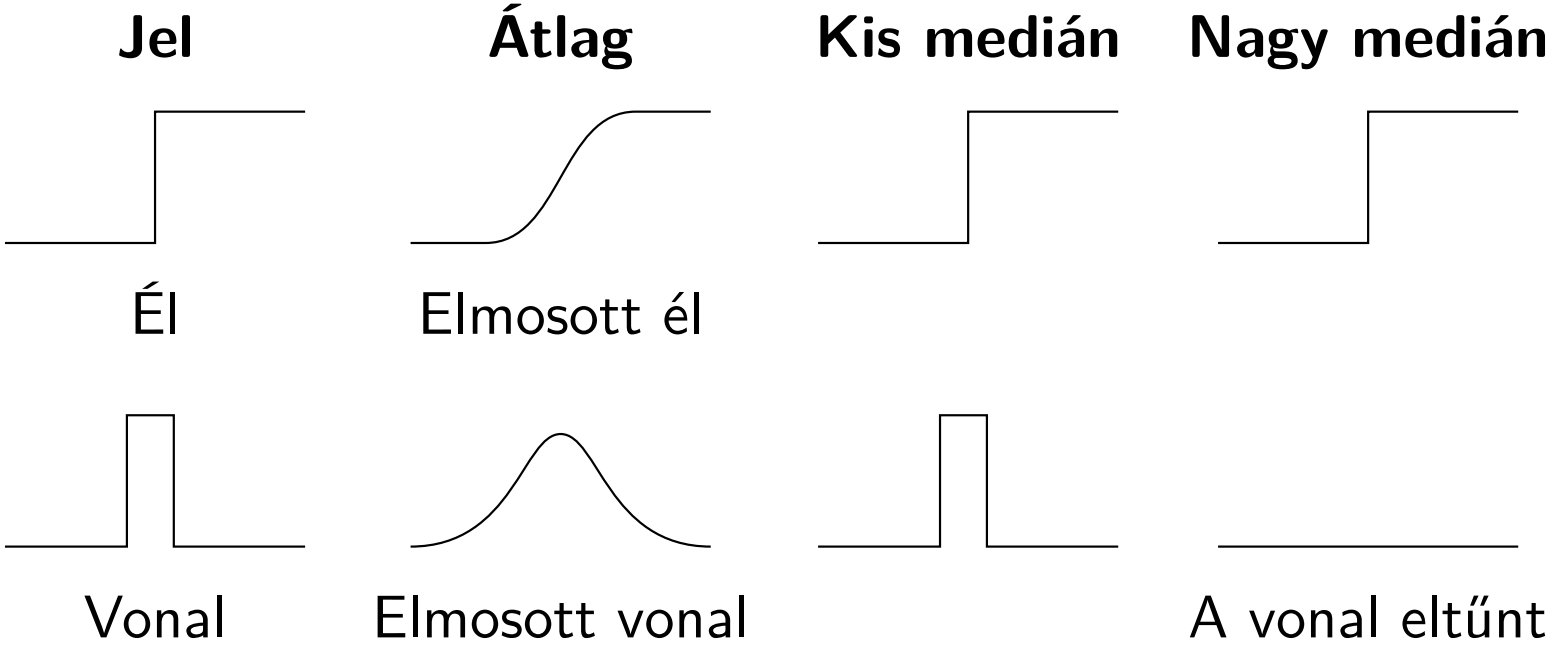
A mediánszűrő tulajdonságai

- Eltávolítja az izolált zajpixeleket
- Nem mossa el a képet, de lekerekíti a sarkokat
- Eltávolítja a vékony vonalakat, ha *szűrőméret* $> 2\times$ *vonalvastagság*
  - a háttérpixelek kerülnek többségbe
- Műveletigény
  - közvetlen megvalósítás: $O(N \cdot N_W \cdot \log N_W)$
  - futó szűrő megvalósítás: $O(N \cdot \log N_W)$
- A vektormedián-szűrő vektormezőket javít
  - eltávolítja a környező vektorokkal összeegyeztethetetlen vektorokat



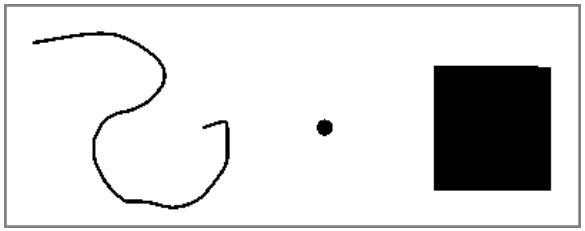
szűrés előtt szűrés után

Lépcsőél és vonal átlagoló és medián szűrése

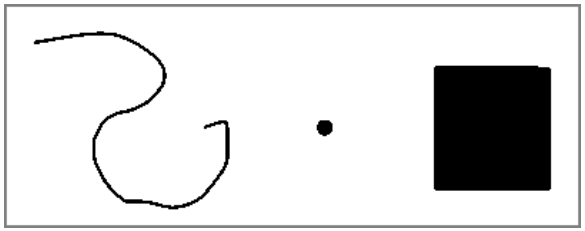


- A vonal eltűnik, ha a mediánszűrő mérete meghaladja a vonalvastagság kétszeresét

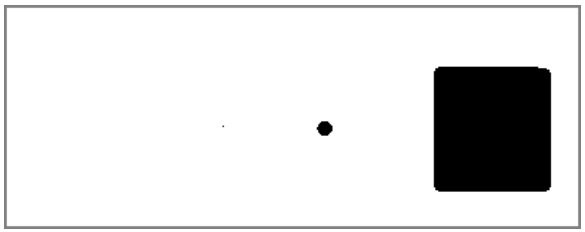
Medián és doboz szűrő összehasonlítása bináris képen



bemeneti kép



medián 3×3



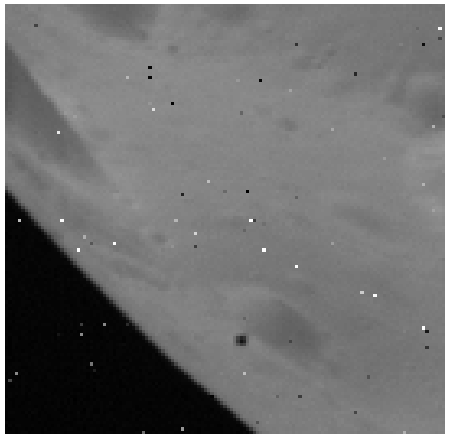
medián 7×7

medián 17×17

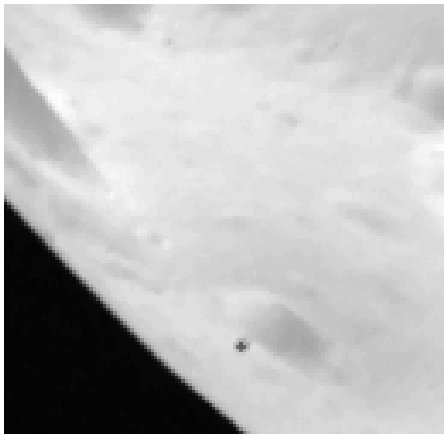
doboz 5×5

doboz 9×9

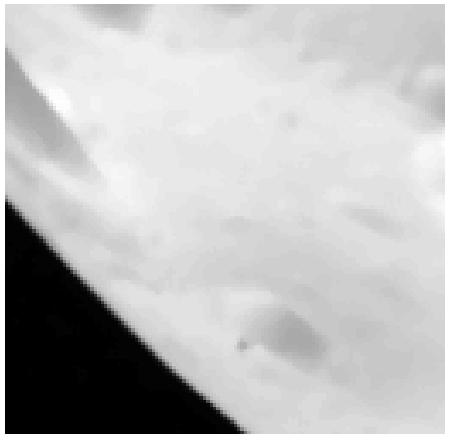
Összehasonlítás só-bors zajjal terhelt képen



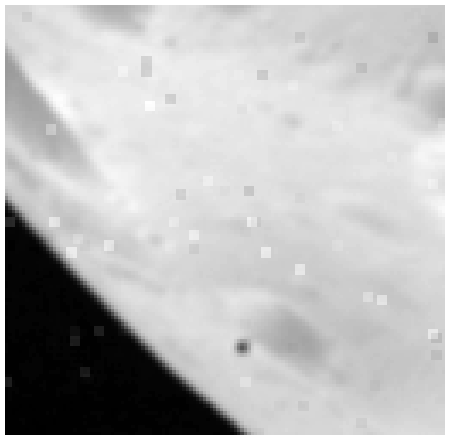
bemeneti kép



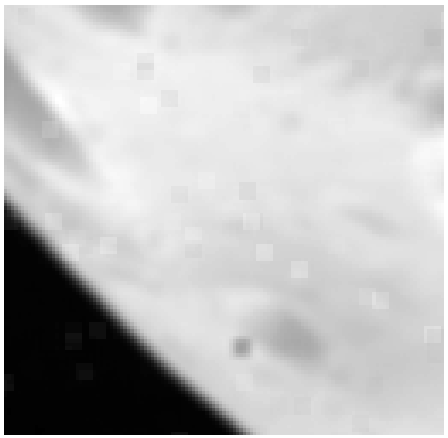
medián 3×3



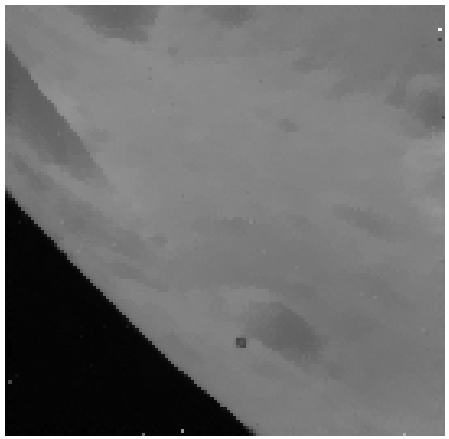
medián 5×5



doboz 3×3



doboz 5×5



szimm. doboz 5×5

A Laplace-operátor és közelítése

- A Laplace-operátor definíciója

$$g(x,y) = \Delta f(x,y) \doteq \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) f$$

- Tervezzünk egyszerű 3×3-as w_L kernelt a Laplace-operátorhoz
  - a deriváltakat differenciákkal közelítjük

$$\frac{\partial f}{\partial x} \longrightarrow \begin{array}{|c|c|c|}\hline -1 & 1 & 0 \\ \hline \end{array}$$

$$\frac{\partial^2 f}{\partial x^2} \longrightarrow \begin{array}{|c|c|c|}\hline -1 & 1 & 0 \\ \hline \end{array} - \begin{array}{|c|c|c|}\hline 0 & -1 & 1 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline -1 & 2 & -1 \\ \hline \end{array}$$

$$\Delta f \longrightarrow \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline -1 & 2 & -1 \\ \hline 0 & 0 & 0 \\ \hline \end{array} + \begin{array}{|c|c|c|}\hline 0 & -1 & 0 \\ \hline 0 & 2 & 0 \\ \hline 0 & -1 & 0 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline 0 & -1 & 0 \\ \hline -1 & 4 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array}$$

A Laplace-szűrő és az átlagolás

A kernelt 4-gyel normálva

$$w_L = \Delta f(x,y) \approx f(x,y) - Av(x,y),$$

ahol Av a négy szomszéd átlaga

$$Av(x,y) \doteq \frac{1}{4}\Big[f(x-1,y) + f(x,y-1) + f(x+1,y) + f(x,y+1)\Big]$$

$\frac{1}{4}$

0	−1	0
−1	4	−1
0	−1	0

4-szomszédos változat

$\frac{1}{8}$

−1	−1	−1
−1	8	−1
−1	−1	−1

8-szomszédos változat

Laplace-szűrésre használt egyszerű maszkok

A Laplace-szűrő tulajdonságai 1/2

- Közel áll az eredeti és a simított kép különbségéhez
  - a fokozatos változások kivonódnak, a finomak megmaradnak
  - a kép nem változó részein nulla a válasz
- Formálisan a kimeneti tartomány $[-255, 255]$
  - egy pixel és szomszédai különbsége kicsi

  $\Rightarrow$ a gyakorlatban a tartomány szűk
- Kiemeli az intenzitásváltozásokat, a finom részleteket
  - kontúrok, foltok, vékony vonalak

A Laplace-szűrő tulajdonságai 2/2

- Zajérzékeny: másodrendű deriváltakat tartalmaz
- Általában simító szűrőkkel kombinálva használjuk
- Gauss-Laplace (Laplacian-of-Gaussian, LoG)

$$w_{LoG} = w_G * w_L$$

  - deriválás előtt sima függvényt állítunk elő
  - kevésbé zajérzékeny, mint a Laplace-szűrő
  - a LoG nullátmenetei az *élek*

Példák Laplace-szűrésre 1/3

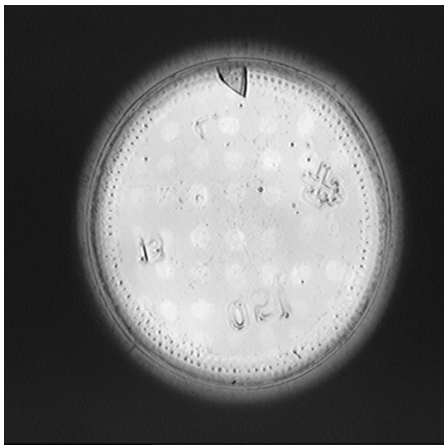
bemenet

Laplace, abszolút érték

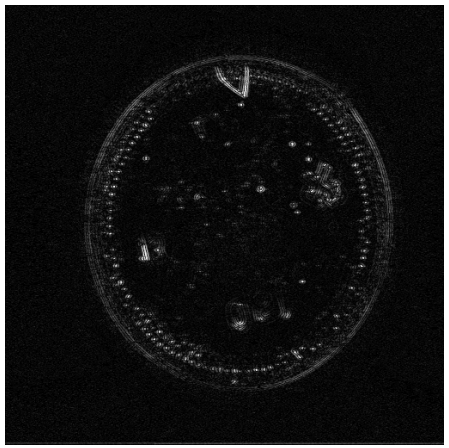
Laplace, eltolt

- A kimenet két különböző megjelenítése látható
  - abszolút érték szerinti leképezés: $-127 \to 127$, $127 \to 127$
  - eltolt leképezés: $-127 \to 0$, $127 \to 254$
- A leképezéstől függően más-más részletek látszanak

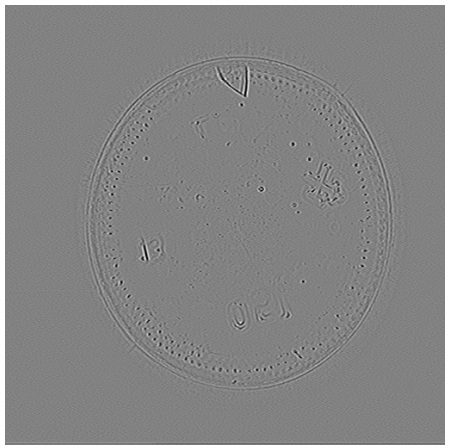
Példák Laplace-szűrésre 2/3



bemenet



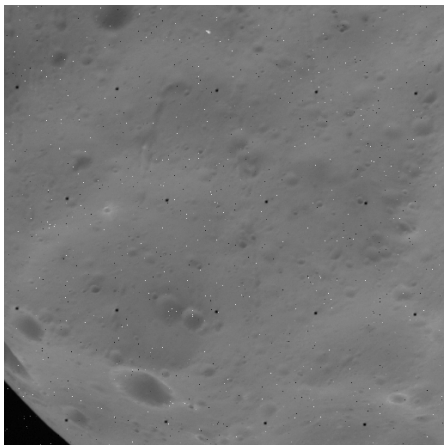
Laplace, abszolút érték



Laplace, eltolt

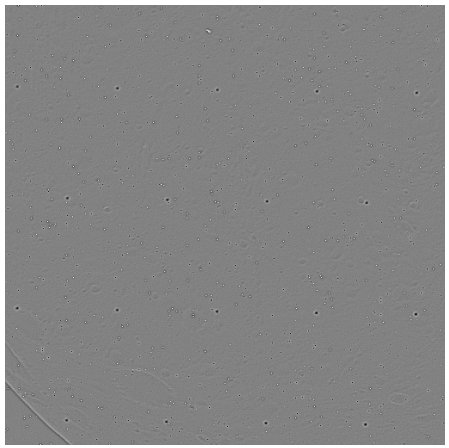
- A finom részletek kiemelődnek, köztük az üvegdarab és a jelek
- A fokozatos változások elnyomódnak

Példák Laplace-szűrésre 3/3



bemenet

Laplace, abszolút érték



Laplace, eltolt

- A Laplace-szűrő zajérzékeny
- Csúcszajos, kontrasztos részleteket nem tartalmazó bemenetre a kimenet nagyrészt zaj

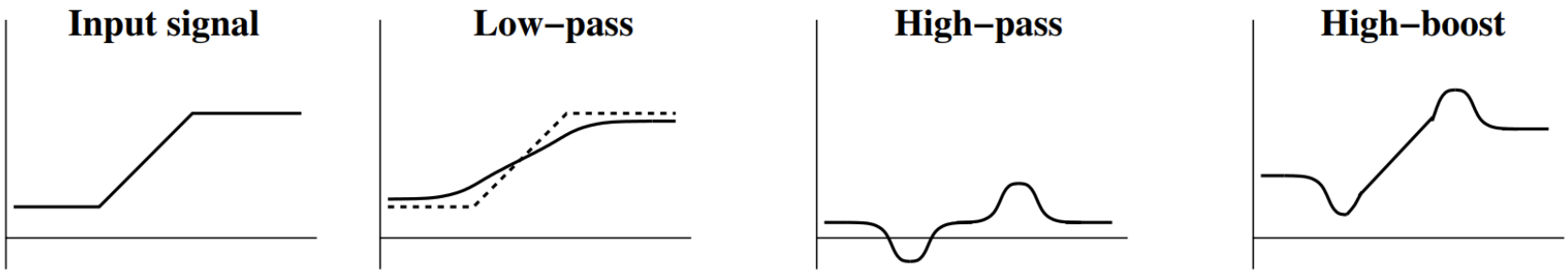
Az életlen maszkolás szűrője

- Cél: a kontúrok és más magas frekvenciás jellemzők kiemelése
- Megoldás: a Laplace-kimenet egy részének hozzáadása a bemeneti képhez
  - a Laplace-szűrő felerősíti a képbeli változásokat
- Definíció:

$$g(x, y) = f(x, y) + \lambda \cdot \Delta f(x, y)$$

- A $\lambda > 0$ paraméter
  - nagyobb $\lambda \Rightarrow$ a magas frekvenciás jellemzők erősebb kiemelése
  - az életlen maszkolás **kiemelő** (high-boost) szűrő

Az életlen maszkolás jelentése



$f(x)$ $\quad f * w_G \quad f * w_L = f - f * w_G \quad f + \lambda \cdot f * w_L$

- *Aluláteresztő* (átlagoló): a kép simítása
- *Felüláteresztő* (Laplace): a kép és az aluláteresztő kimenetének különbsége
- *Kiemelő* (életlen maszkolás): a felüláteresztő egy részét a képhez adjuk
- Az „alul-” és „felül-” a szűrő *frekvenciatartománybeli* hatására utal

Egyszerű konvolúciós kernel az életlen maszkoláshoz

A Laplace-szűrő 8-szomszédos változatával, $f + \lambda\Delta f$-re:

$$\begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} + \lambda \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 8 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} = \lambda \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 8+1/\lambda & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$

A $\beta = 1/\lambda$ paraméter bevezetésével és normálással a kernel:

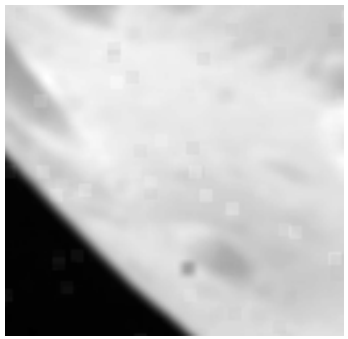
$$w_U = \frac{1}{9} \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 8+\beta & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$

- $0 \leq \beta \leq 1$; tipikus értékek: 0,1–0,2
- Normálás: a lehetséges legnagyobb kimeneti érték G_{max} (255)
  - más normálás is használható

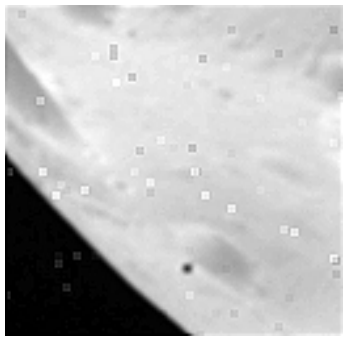
Példák és összefoglalás: életlen maszkolás

1. kép

1. eredmény



2. kép



2. eredmény

- Kontrasztjavításra használják, főként a fényképészetben
- Kiemeli a magas frekvenciás jellemzőket, például az éleket
- Felerősítheti a zajt

Tartalom

1 Képszűrés

2 Gyakran használt szűrők

3 Gyors és adaptív szűrők
- Szeparálható szűrők
- Futó szűrés
- Adaptív zajszűrés

A szűrő szeparálhatósága 1/2

- A szűrő felbontható 1D szűrők szorzatára

$$w(x, y) = u(x) \cdot v(y)$$

- Példa szeparálható szűrőre
  - a 2D szűrőmátrix minden eleme az 1D szűrők megfelelő elemeinek szorzata

$w(x, y)$

1	2	1
2	4	2
1	2	1

$u(x)$

1	2	1

$v(y)$

1
2
1

A szűrő szeparálhatósága 2/2

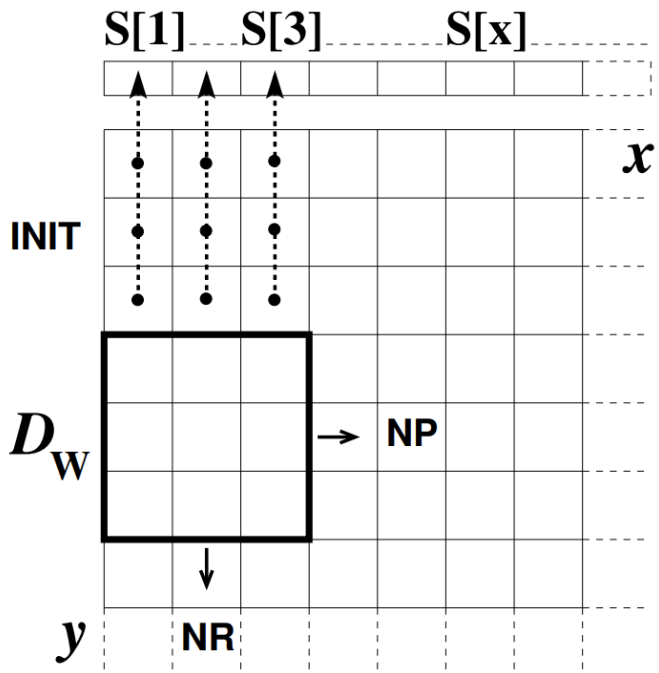
- A $N_{\text{műv}}$ műveletszám pontonként $D_W \times D_W$-s ablakra
  - eredeti szűrő: $N_{\text{műv}} = O(D_W^2)$
  - szeparálható szűrő: $N_{\text{műv}} = 2 \cdot O(D_W)$
- A Gauss-szűrő és a doboz szűrő szeparálható
  - Gauss: $w_G(x, y) = w_G(x) \cdot w_G(y)$, $w_G(x) \propto \exp\left\{-\frac{x^2}{2\sigma^2}\right\}$
  - a doboz szűrő két egységnyi 1D szűrő szorzata
  - a doboz szűrő futó megvalósítása még gyorsabb
- 2D szűrő felbontása 1D szűrők lineáris kombinációjára
  - szinguláris érték-felbontás (SVD) használatával
  - nem feltétlenül gyorsabb: az 1D szűrők számától függ

A futó szűrés fogalma

- Amikor az ablak a következő pozícióba lép,
  - ne *a nulláról* számoljuk a kimeneti értéket,
  - hanem *frissítsük* az előző pozícióban kapott értéket
- Különböző szűrőkre léteznek futó szűrési megoldások
  - doboz szűrő
  - mediánszűrő
- A hatékonyság a frissítés egyszerűségén múlik
  - az additív mennyiségek, mint az átlag, könnyen frissíthetők
  - a nemlineáris mediánt nehezebb frissíteni
- A futó szűrés bonyolultabb alakú ablakokra is kiterjeszthető

Futó szűrés doboz szűrőre

- Adatszerkezet: $S[x]$ tömb
- Inicializálás (INIT)
  - a kezdősorra kiszámítjuk az $S[x]$ oszlopösszegeket
- Első pozíció a sorban
  - az ablakösszeget $S[x]$-ből számoljuk
- Lépés a sorban (NP)
  - az ablakösszeg frissítése: a kilépő S kivonása, a belépő S hozzáadása
- Következő sor (NR):
  - minden $S[x]$ frissítése: a kilépő pixel kivonása, a belépő hozzáadása



N_{ops} független D_W-től, ha a kép sokkal nagyobb az ablaknál

Adaptív szomszédságválasztás

- Az eddig tárgyalt szűrők nem adaptívak (pozíciófüggetlenek):
  - rögzített szomszédságválasztó eljárás
  - rögzített függvény számolja a kimeneti értéket
- Az adaptivitás a **lokális kontextus használatát** jelenti a zajszűrők teljesítményének javítására
  - elkerülni az „éleken átnyúló átlagolást” az átlagoló szűrőnél
  - elkerülni a sarkok lekerekítését a mediánszűrőnél
- E nemkívánatos hatások fő oka
  - a szűrő különböző osztályokba (eloszlásokba) tartozó pixeleket kever össze
  - ha az ablak kontúron áll, az objektum- és háttérpixelek keverednek

Az alapötlet

- Próbáljuk szétválasztani
  - az objektumpixeleket a háttérpixelektől
  - a releváns szürkeségi értékeket a zajtól
- Adaptivitás a szomszédos pixelek kiválasztásában: a *releváns* pixeleket választjuk ki
  - eddig az ablak minden pixelét használtuk
  - mostantól bizonyos pixeleket választunk ki
- Adaptivitás a kimeneti értéket számoló függvényben: nincs
  - eddig rögzített függvényeket használtunk: átlag, medián stb.
  - ezen *nem* változtatunk

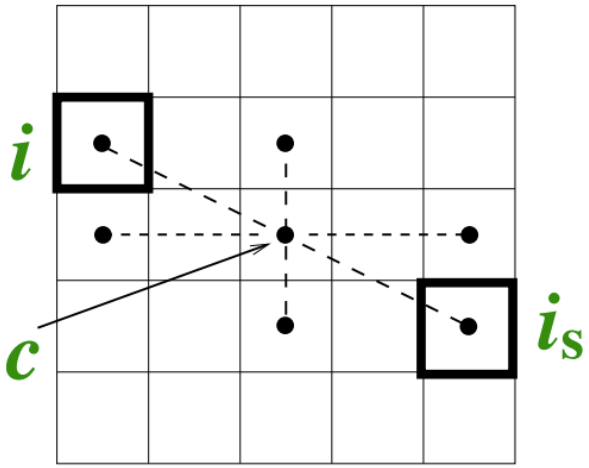
Pixelválasztás $n \times n$-es ablakban

- **Standard** szomszédság
  - mind az n^2 pixelt használjuk
- k **legközelebbi** szomszéd (k-NN)
  - azt a k pixelt választjuk, amelyek szürkeségi értékben a legközelebb vannak a c központi pixelhez
  - lehetséges választás: $k = n \times \left[\frac{n}{2}\right] + (n-1)$
  - például $n = 3$ esetén $k = 5$
- **Szigma-legközelebbi** szomszédok
  - az i pixelt választjuk, ha $|I(i) - I(c)| < k \cdot \sigma_{zaj}$
  - általában $k = 2$
  - σ_{zaj} a zaj szórása

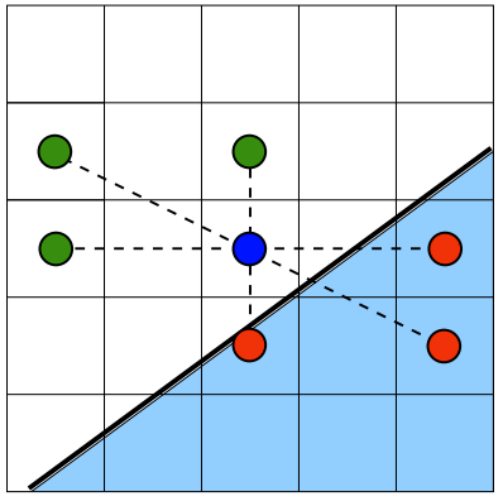
  $\Rightarrow$ a kép egyenletes (nem változó) tartományán becsüljük

Szimmetrikus legközelebbi szomszédok

- Az i pixelt választjuk, ha $|I(i) - I(c)| < |I(i_s) - I(c)|$
  - c a központi pixel, $\{i, i_s\}$ centrálisan szimmetrikus pixelpár
- Lokális kontextus: az intenzitást és a geometriát is figyelembe vesszük
- Élek esetén hasznos
  - az él azonos oldalán lévő pixeleket választja
  - elkerüli az élen átnyúló átlagolást

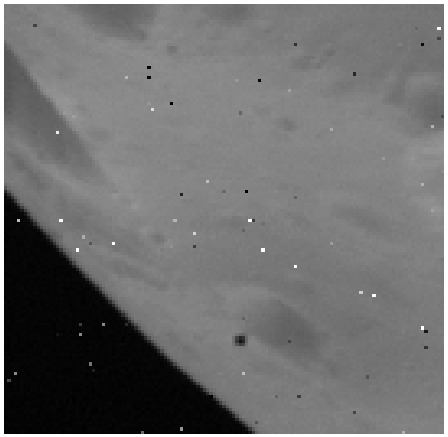


szimmetrikus pixelpár



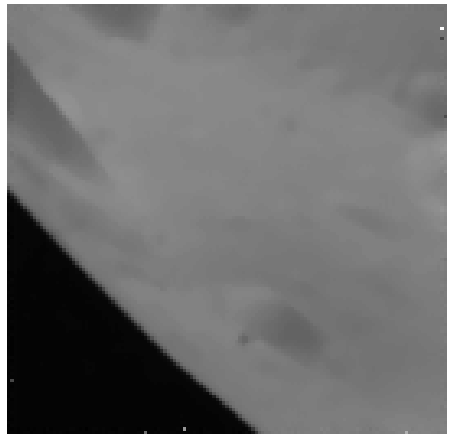
működés élen

Standard és adaptív 5 × 5-ös szűrők összehasonlítása

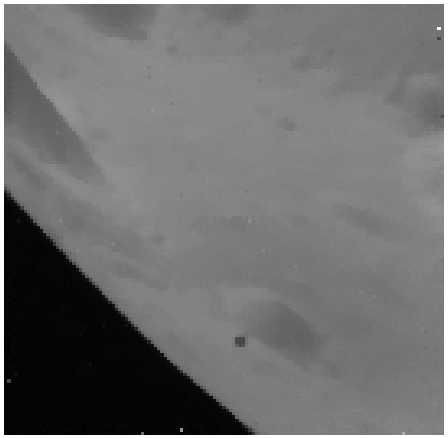


kép

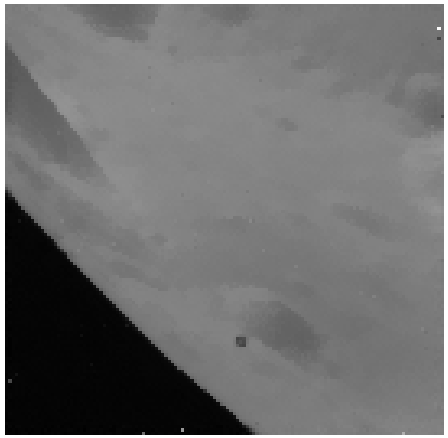
doboz



medián



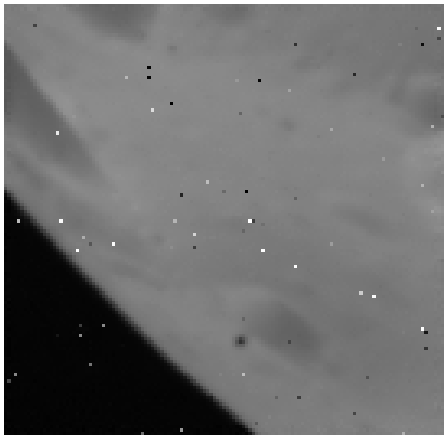
k-NN átlag



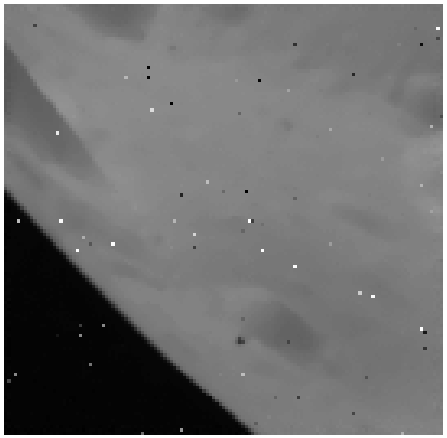
szimm. átlag

szimm. medián

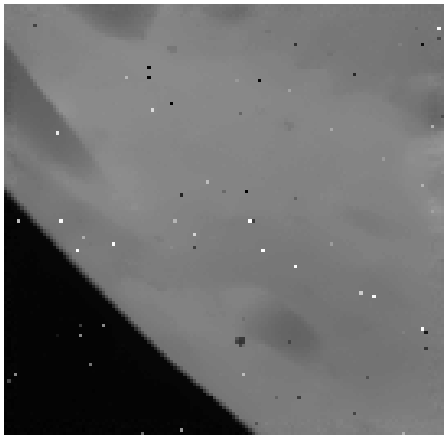
A szigma-szűrő nem távolítja el a csúcszajt



szigma átlag 5×5



szigma medián 5×5



szigma medián 9×9

- Csúcszajos $I_{zajos}(x, y)$ pixelre az $I_{zajos} \pm 2\sigma_{zaj}$ intervallum nem tartalmazza a zajmentes szomszédokat
  - $|I_{zajos} - I_{zajmentes}| > 2\sigma_{zaj}$
- Az I_{zajos} csúcsértéket választjuk ki

  $\Rightarrow$ a zaj nem tűnik el