

Binárne obrazy

Matematická morfológia



Darken ($f - 128$)



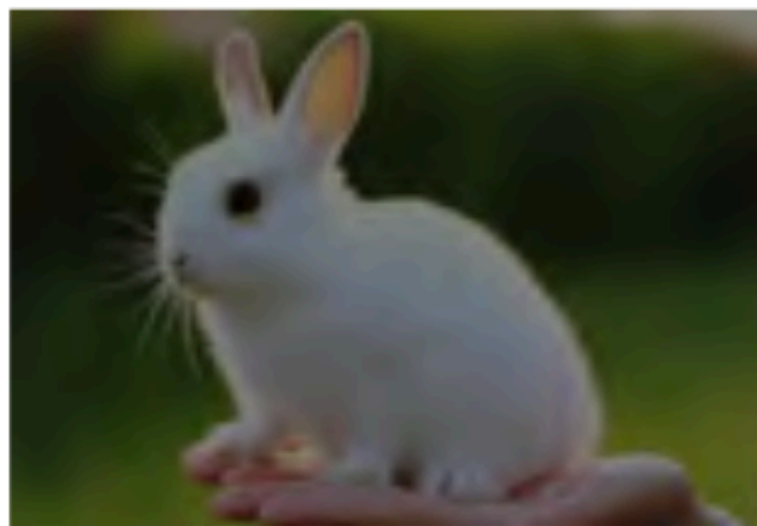
Original (f)



Lighten ($f + 128$)



Invert ($255 - f$)



Low Contrast ($f/2$)



Original (f)



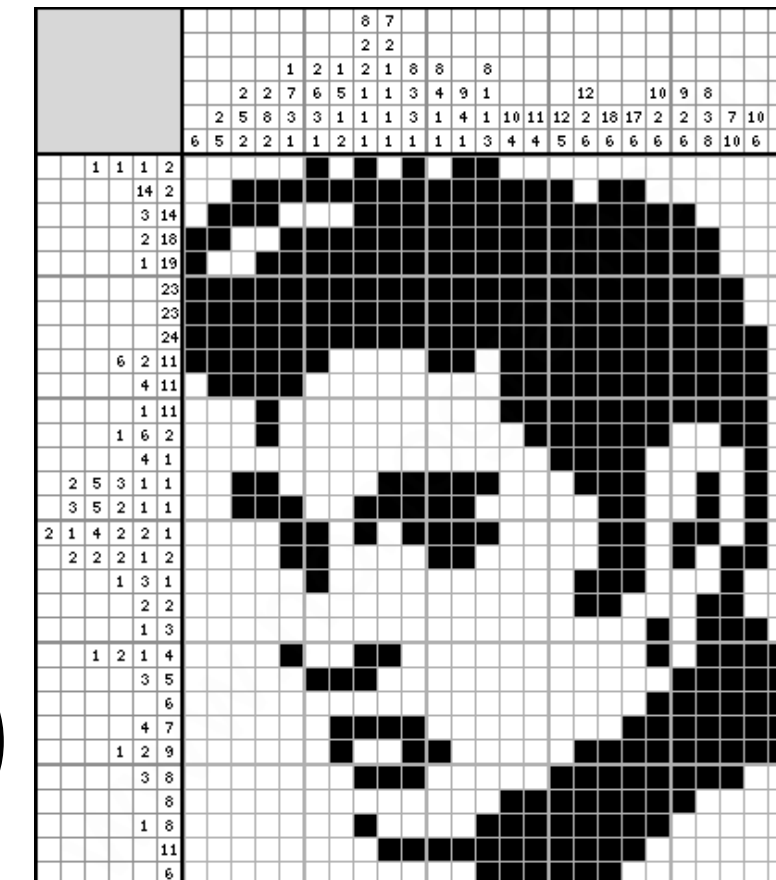
High Contrast ($f * 2$)



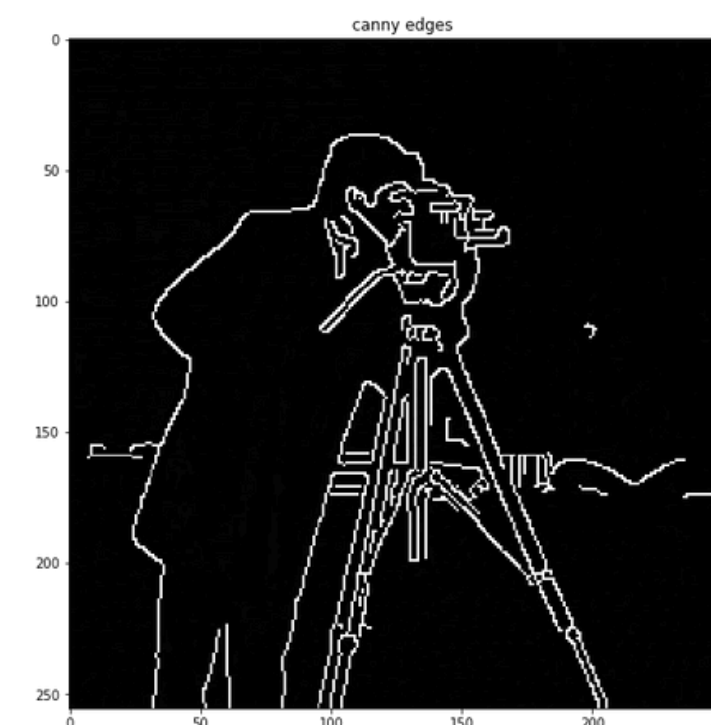
Gray ($0.3f_R + 0.6f_G + 0.1f_B$)

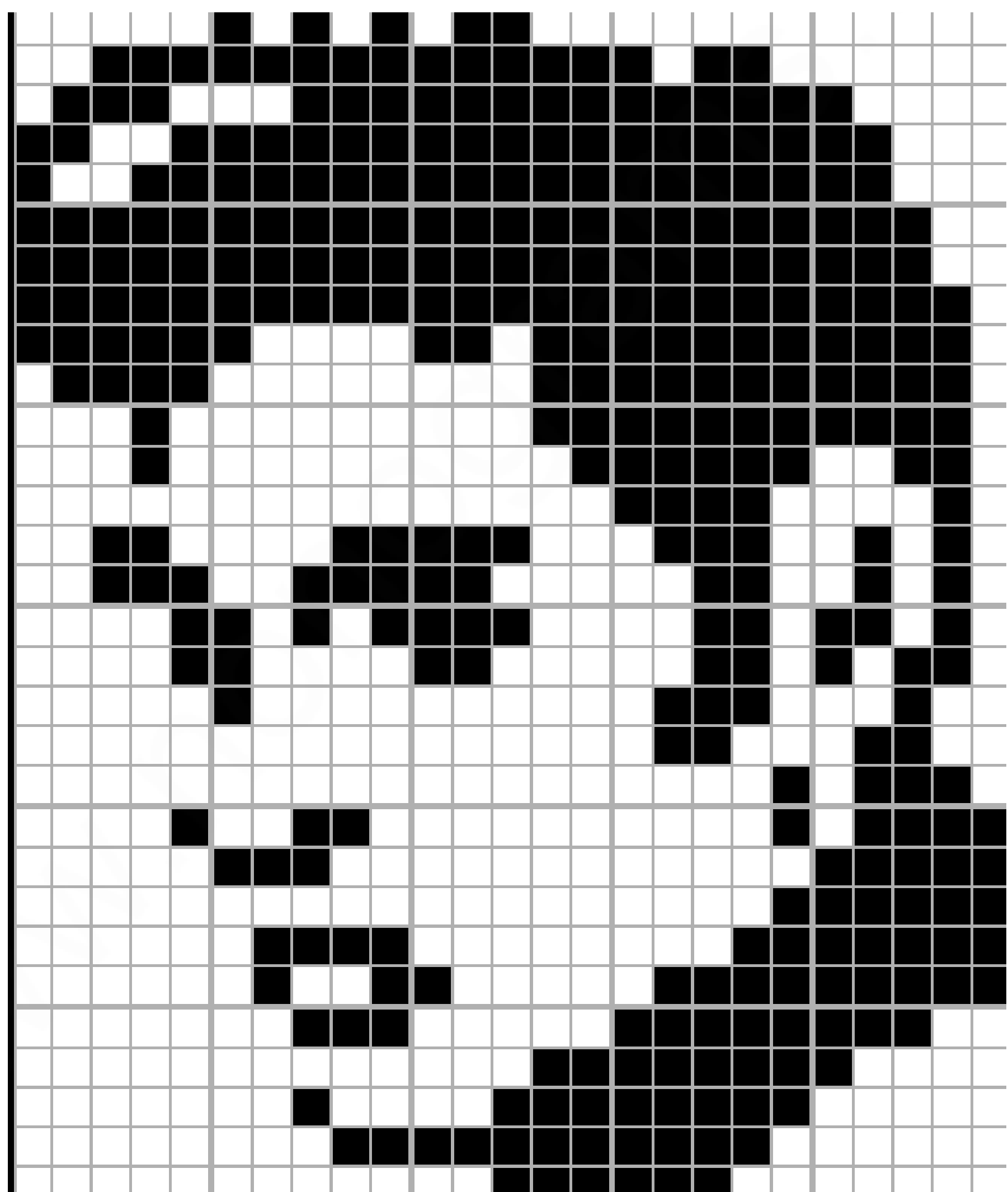
Binárny obraz

- 1-bitová hodnota jasů pre každý pixel
- Malá veľkosť súboru (+ run-length encoding)
- Výstup operácii - napr. detekcie hrán



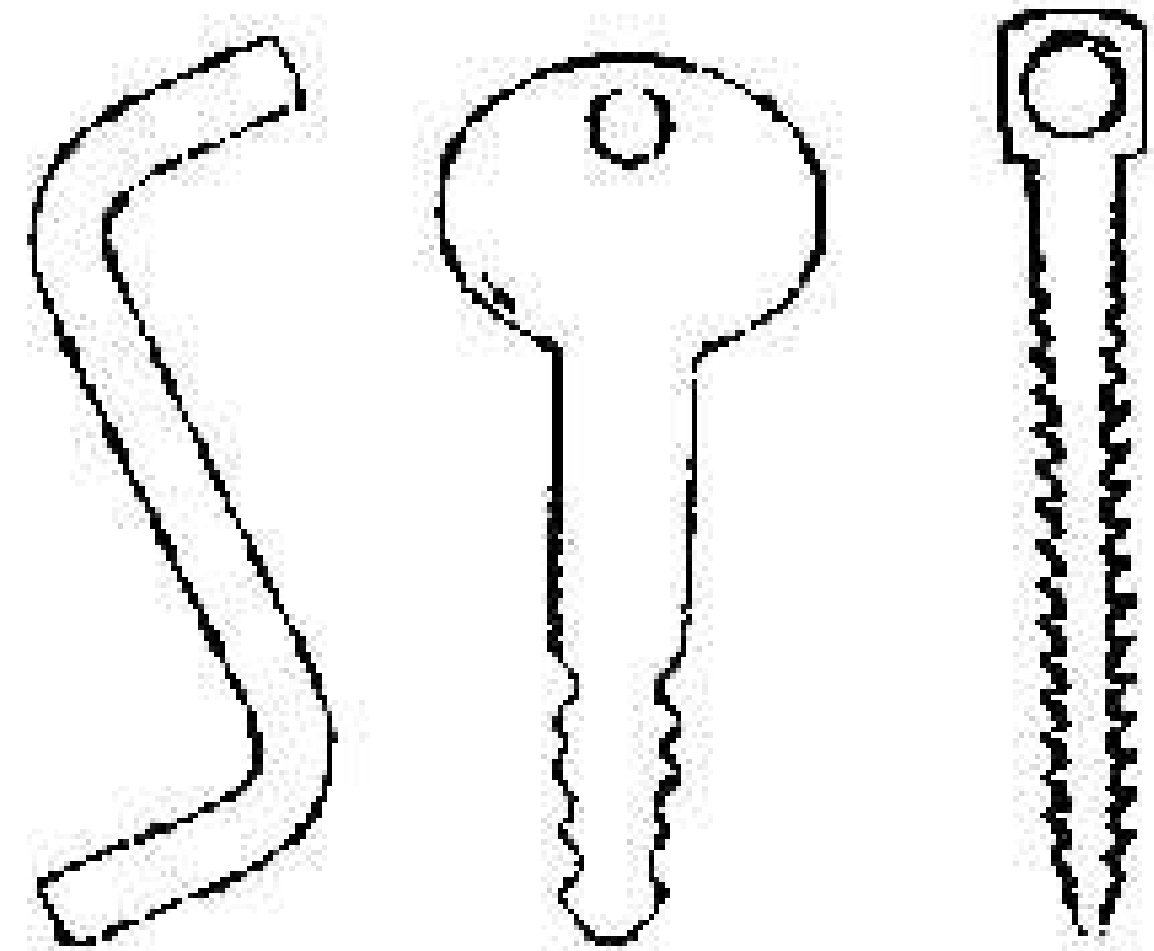
$$\theta(f, t) = \begin{cases} 1 & \text{if } f \geq t, \\ 0 & \text{else,} \end{cases}$$

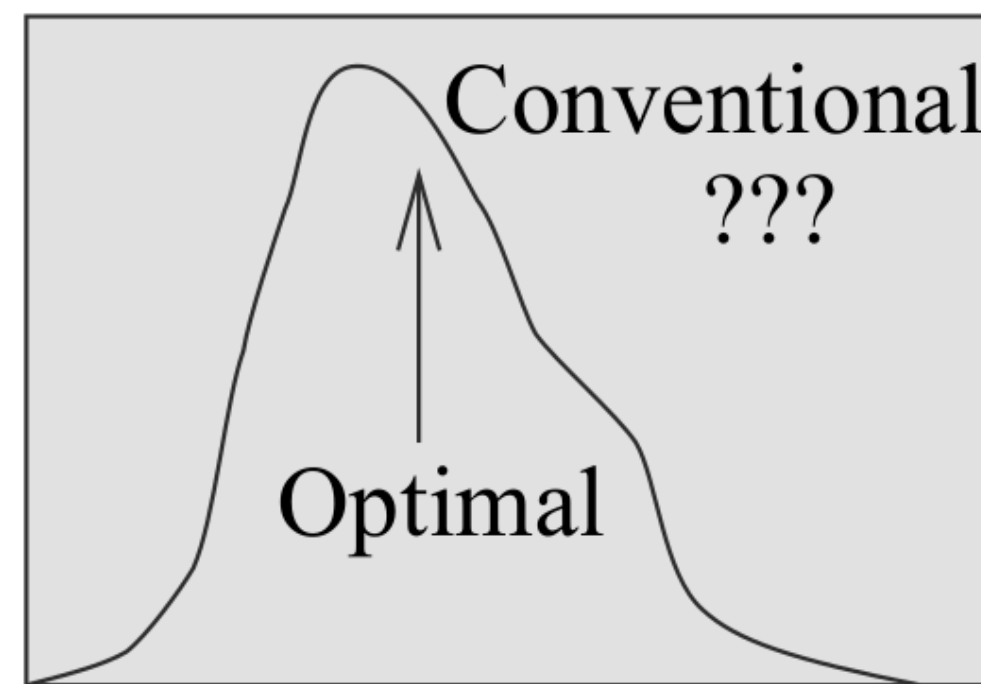
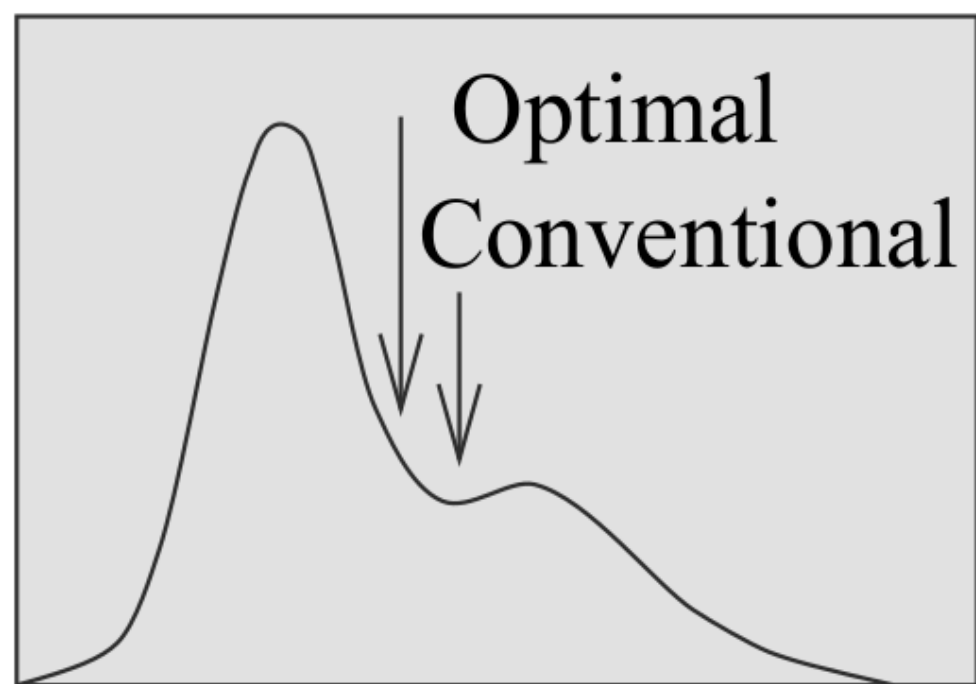
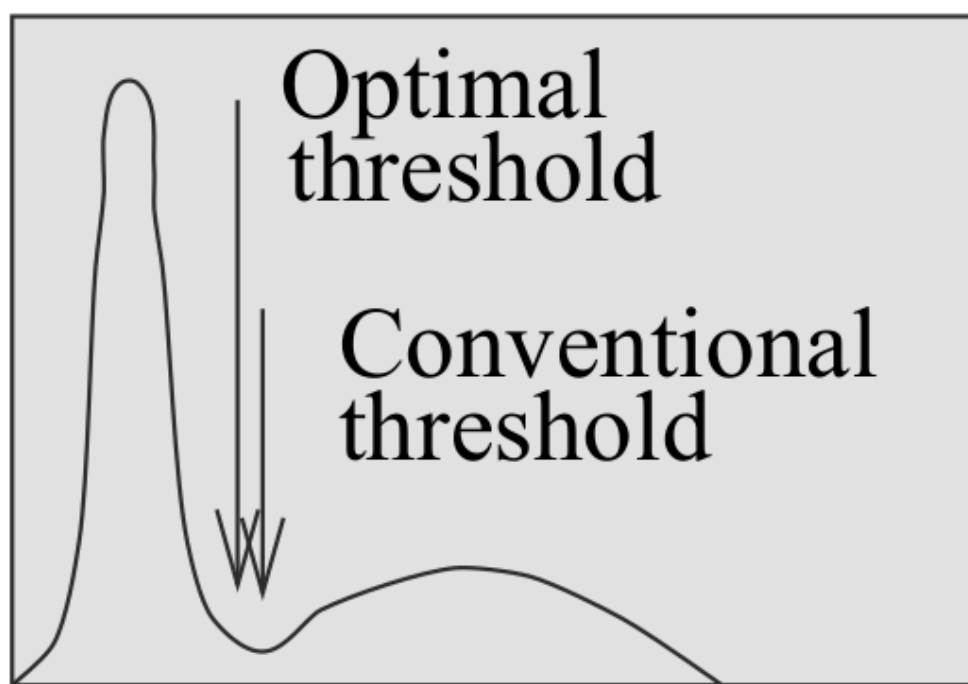
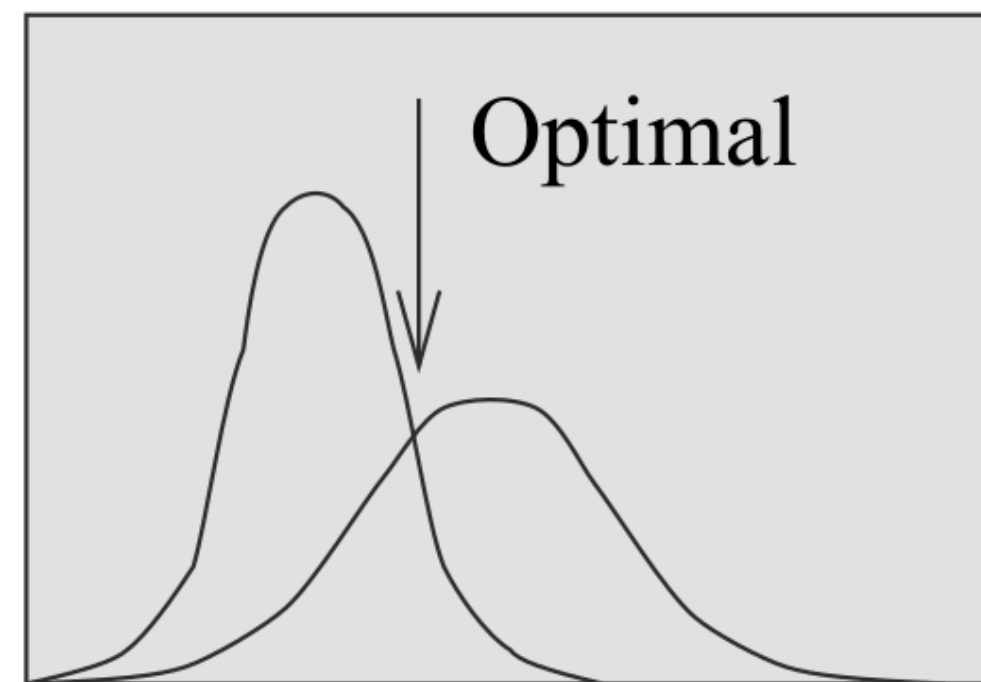
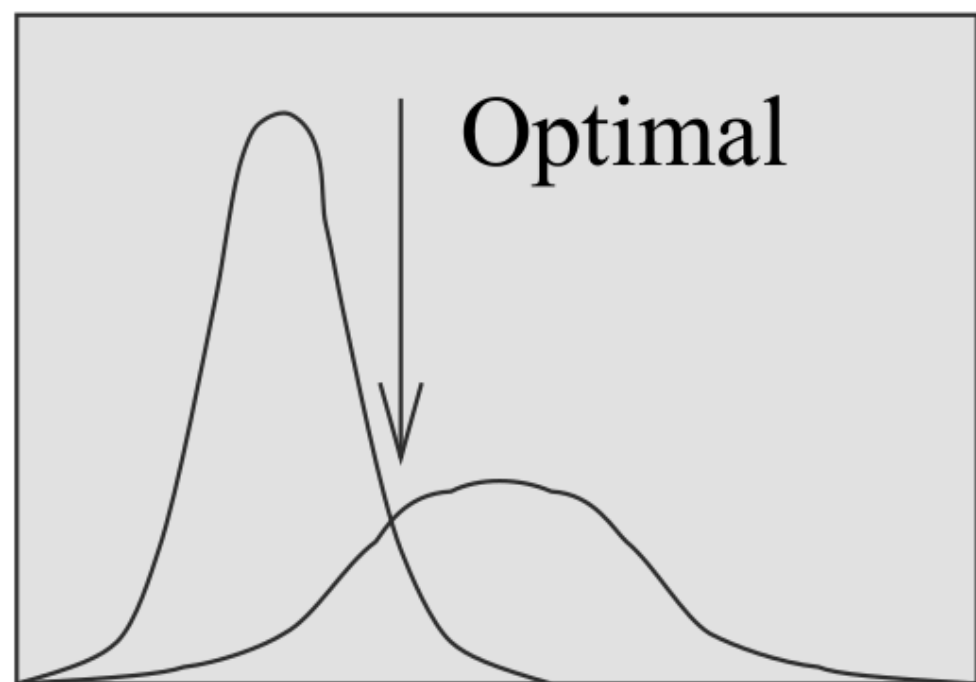
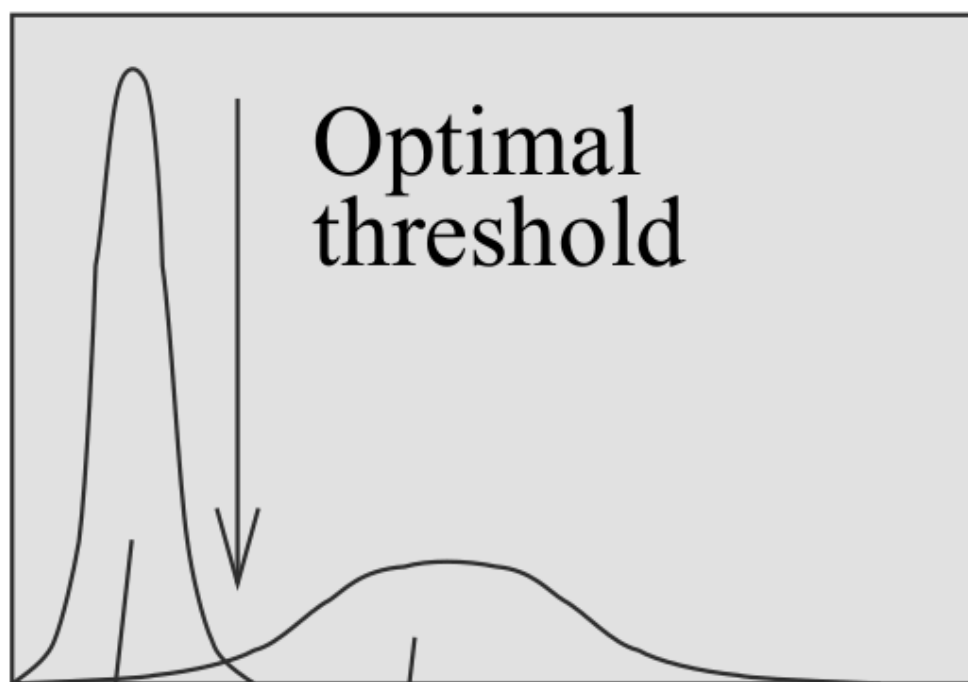




Prahovanie

Prahovanie podľa úrovne jasů





Algorithm 6.2: Otsu's threshold detection

1. For an image I on G gray levels, compute the gray-level histogram $H(0), H(1), \dots, H(G - 1)$. Normalize the histogram by dividing through by the number of pixels in I —the histogram now represents the probability of each gray-level.
2. For each possible threshold $t = 0, \dots, G - 2$ partition the histogram into background B (gray-levels less than or equal to t), and foreground F (gray-levels more than t).
3. Compute $\sigma_B(t), \sigma_F(t)$, the variance of the background and foreground gray-levels. Compute the probability of a pixel being background

$$\omega_B(t) = \sum_{j=0}^t H(j)$$

and $\omega_F(t)$ similarly.

Set

$$\sigma(t) = \omega_B(t)\sigma_B(t) + \omega_F(t)\sigma_F(t)$$

and select as threshold $\hat{t} = \min_t(\sigma(t))$.

Metrika, vzdialenosť, susedstvo

Metrika

Pre ľubovoľnú metriku D platia nasledovné vlastnosti:

- vzdialenosť dvoch bodov je vždy nezáporná

$$D(A, B) \geq 0$$

- vzdialenosť totožných bodov je nulová a ak je vzdialenosť nulová, ide o totožné body

$$D(A, B) = 0 \Leftrightarrow A = B$$

- komutativita

$$D(A, B) = D(B, A)$$

- trojuholníková nerovnosť (pre ľubovoľný bod $C(x_C, y_C)$)

$$D(A, B) \leq D(A, C) + D(C, B)$$

Vzdialenosti

Vzdialenosť dvoch bodov A, B v priestore môže byť definovaná rôznymi spôsobmi, napr.:

- Euklidovská vzdialenosť

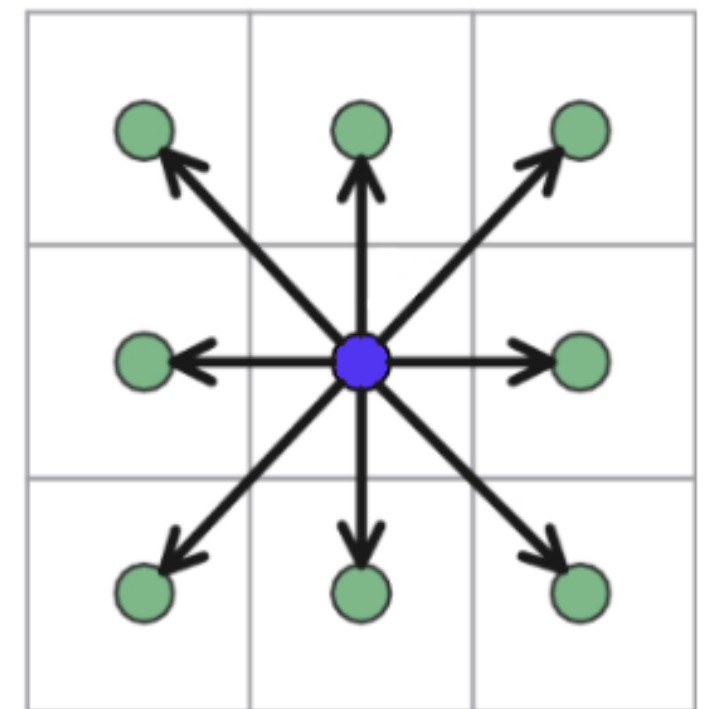
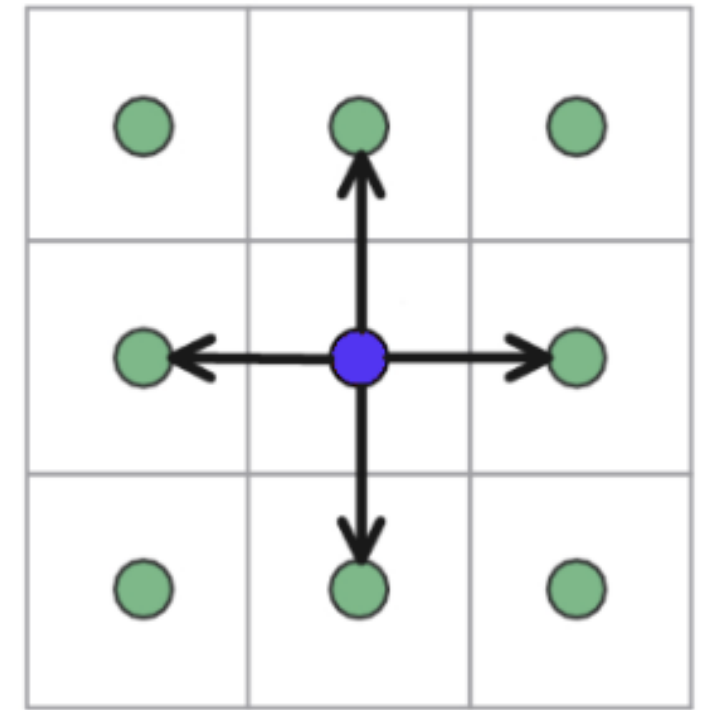
$$D_E(A, B) = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

- City-block distance (Manhattan distance)

$$D_4(A, B) = |x_A - x_B| + |y_A - y_B|$$

- Šachovnicová vzdialenosť (Chebyshev)

$$D_8(A, B) = \max(|x_A - x_B|, |y_A - y_B|)$$



Pojem **4-susedstvo** a **8-susedstvo** je definovaný podľa príslušnej metriky. Dva body (resp. políčka v mriežke) sú susedné ak ich vzdialenosť D_4 resp. D_8 je 1.

Matematická morfológia

Matematická morfológia

Nasledovné metódy slúžia na modifikáciu binárneho obrazu. Pracujú s obrazom ako s množinou bodov (*point set*), môžeme uvažovať množinu čiernych bodov. Aplikujú sa pravidlá štandardnej množinovej algebry.

Morfologická transformácia Ψ je relácia medzi obrazom X a štruktúrnym elementom B , pričom X a B sú bodové množiny.

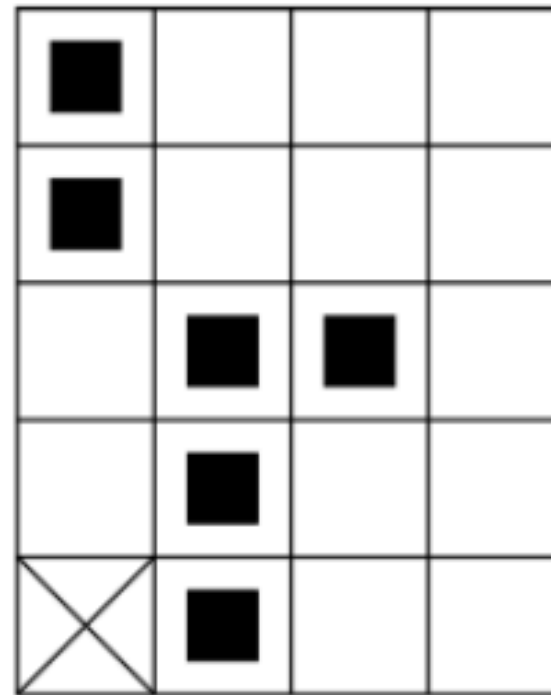


Fig. 5 Znáozornenie bodovej množiny $X = \{(1, 0), (1, 1), (1, 2), (2, 2), (0, 3), (0, 4)\}$. Označený je počiatok $(0, 0)$. Os x rastie vpravo, os y hore.

Štruktúrny element B je bodová množina, ktorá môže byť vnímaná ako vzor, podľa ktorého je upravovaný obraz (details sú uvedené pri jednotlivých metódach). Počiatok, bod $(0, 0)$, môže ale nemusí byť súčasťou štruktúrneho elementu. Najčastejšie sa používajú elementy v tvare obdĺžníka, elipsy alebo kríža (*viď dokumentáciu k opencv*).

Posunutie o vektor

Bodovú množinu X je možné posunúť o vektor h , tak, že každý bod z množiny X je posunutý o zadaný vektor.

$$X_h = \{p \in \varepsilon^2 : p = x + h, \forall x \in X\}$$

ε^2 označuje dvojrozmerný euklidovský priestor.

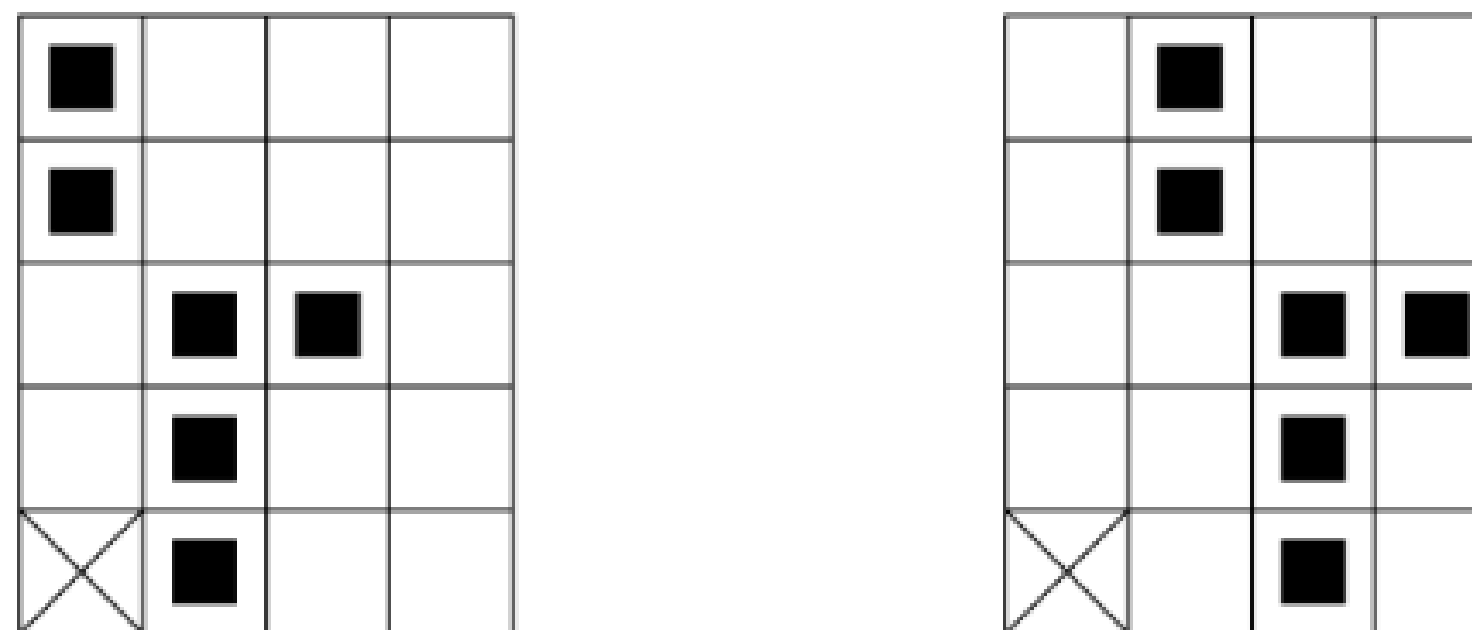


Fig. 6 Obrázok znázorňuje posunutie bodovej množiny $X = \{(1, 0), (1, 1), (1, 2), (2, 2), (0, 3), (0, 4)\}$ o vektor $(1, 0)$.

Dilatácia a Erózia



Dilatácia

Dilatácia je morfológická operácia, ktorá vo všeobecnosti zväčšuje bodovú množinu.

$$X \oplus B = \{p \in \varepsilon^2 : p = x + b, x \in X, b \in B\}$$

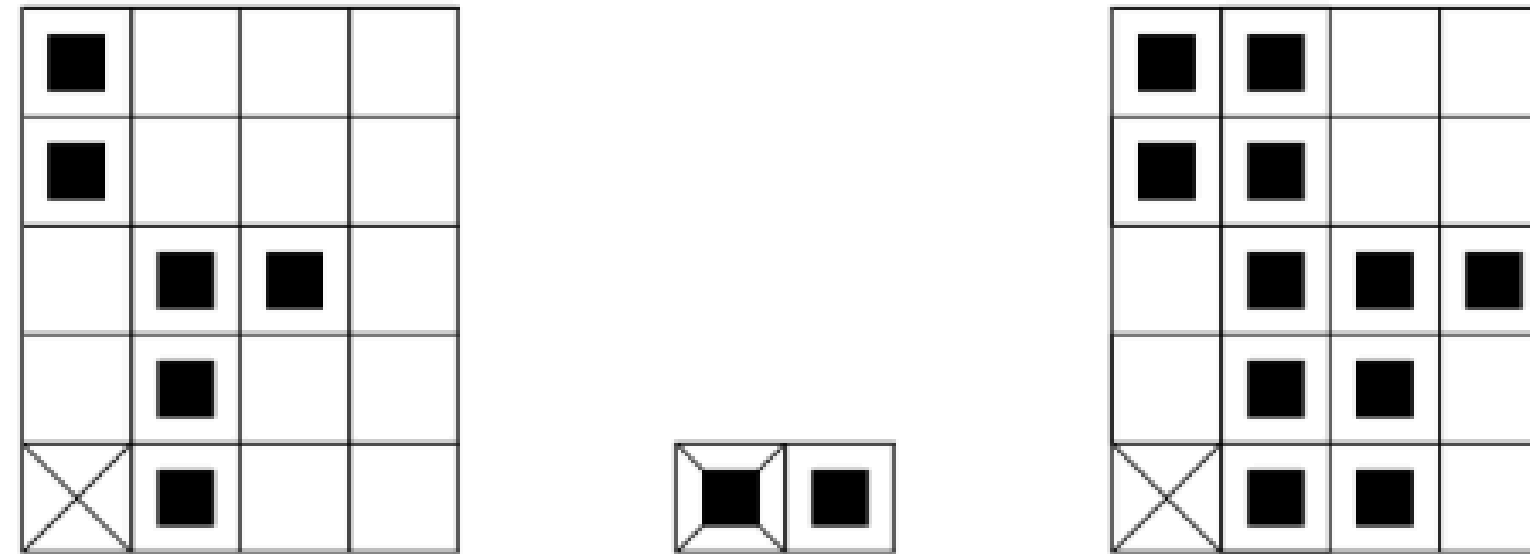


Fig. 7 Ukážka dilatácie. Vľavo je obraz X , v strede štruktúrny element B a vpravo výsledná bodová množina po dilatácii $X \oplus B$.

Erózia

Operácia erózie spôsobuje zmenšenie bodovej množiny.

$$X \ominus B = \{p \in \varepsilon^2 : p + b \in X, \forall b \in B\}$$

$$X \ominus B = \{p \in \varepsilon^2 : B_p \subseteq X\}$$

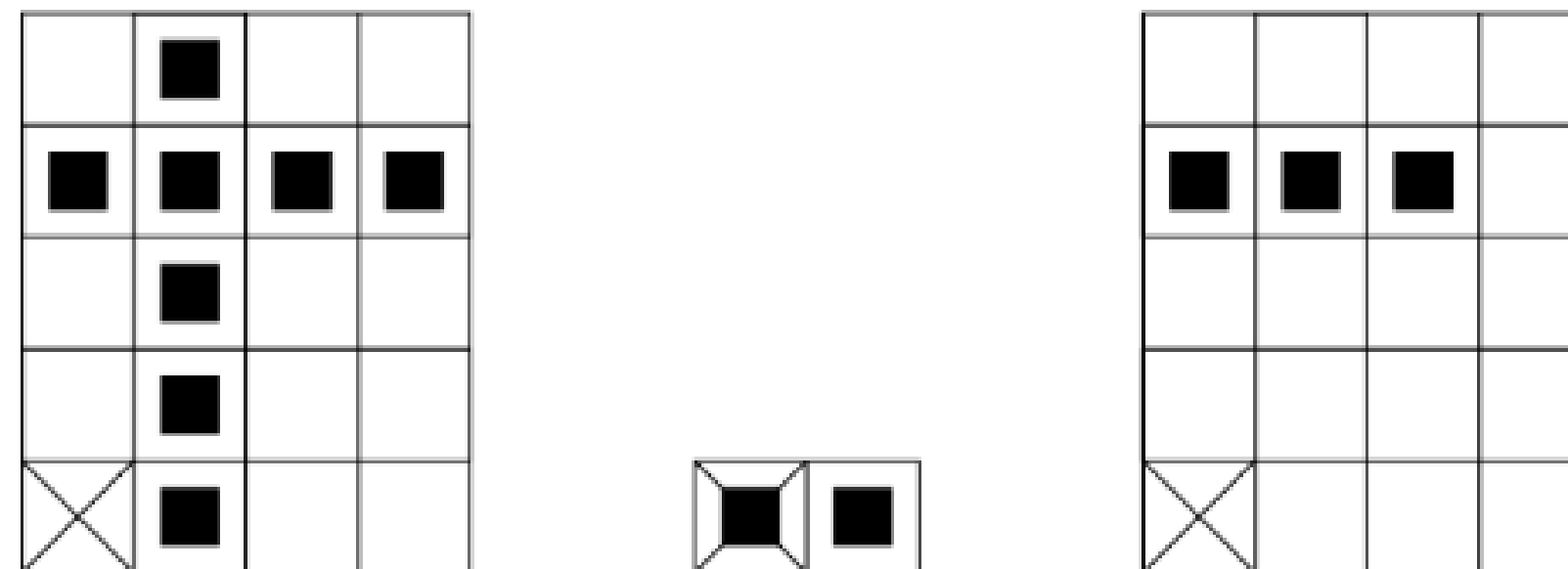


Fig. 8 Ukážka erózie. Zjednodušene povedané: štruktúrny element sa posúva po originálnom obraze. Vo výsledku bude čierny pixel ak po priložení je zhoda (čierne pixely) na všetkých miestach v obraze, ktoré prislúchajú pozíciám v štruktúrnom elemente.

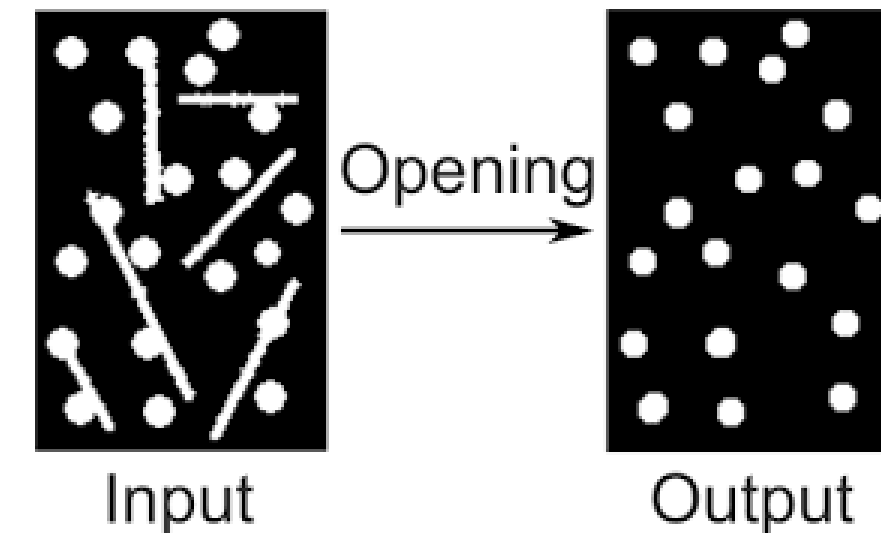
Vlastnosti dilatácie:

- komutatívna: $X \oplus B = B \oplus X$
- asociatívna: $X \oplus (B \oplus D) = (X \oplus B) \oplus D$
- invariantná voči posunutiu: $X_h \oplus B = (X \oplus B)_h$
- rastúca: $X \subseteq Y \implies X \oplus B \subseteq Y \oplus B$

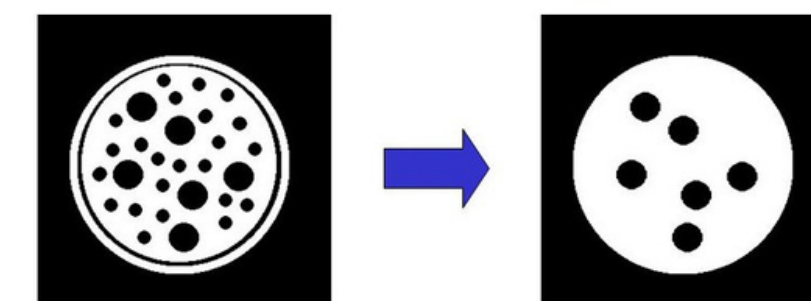
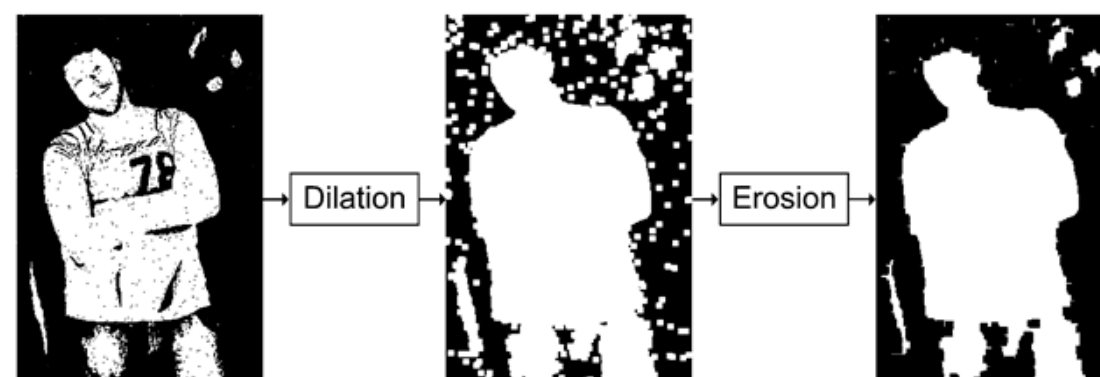
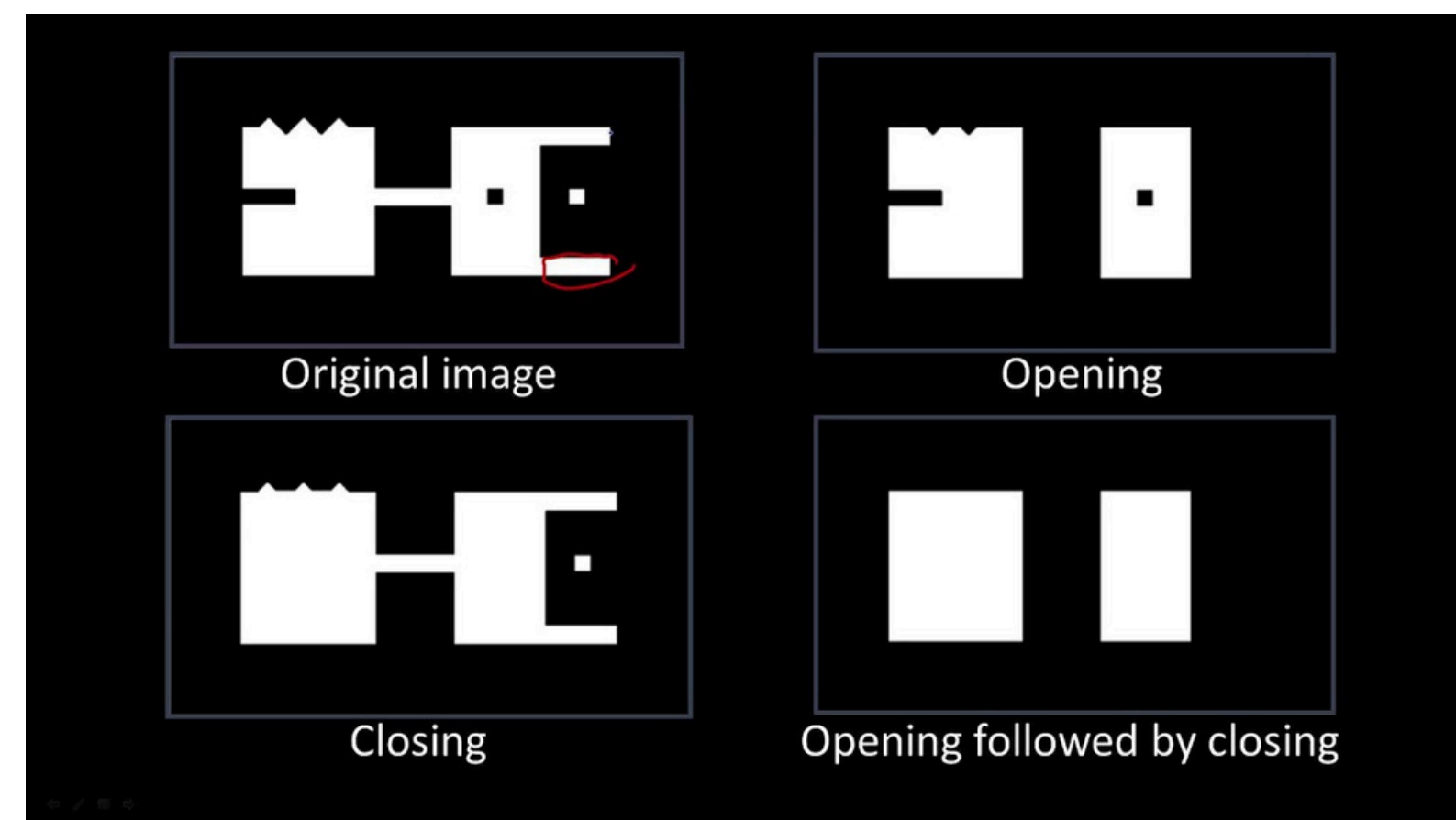
Niektoré vybrané vlastnosti erózie:

- ak je počiatok súčasťou štruktúrneho elementu, tak sa nezväčšuje $(0, 0) \in B \implies X \ominus B \subseteq X$,
- **nie je** komutatívna: $X \ominus B \neq B \ominus X$
- invariantná voči posunutiu: $X_h \ominus B = (X \ominus B)_h$
- rastúca: $X \subseteq Y \implies X \ominus B \subseteq Y \ominus B$

Otvorenie a uzavretie



- Opening = erózia + dilatácia
 - Zväčší diery
 - Odstráni malé objekty
 - Oddelí objekty
- Closing = dilatácia + erózia
 - Zachová malé objekty
 - Odstráni diery
 - Spojí objekty



Distance transform

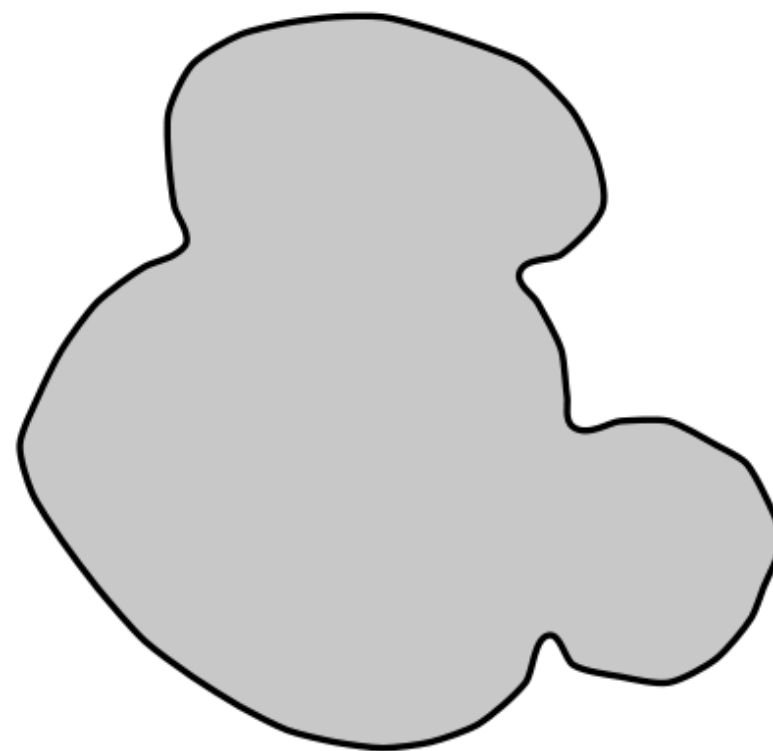
0	0	0	0	1	0	0
0	0	1	1	1	0	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	1	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0

0	0	0	0	1	0	0
0	0	1	1	2	0	0
0	1	2	2	3	1	0
0	1	2	3			

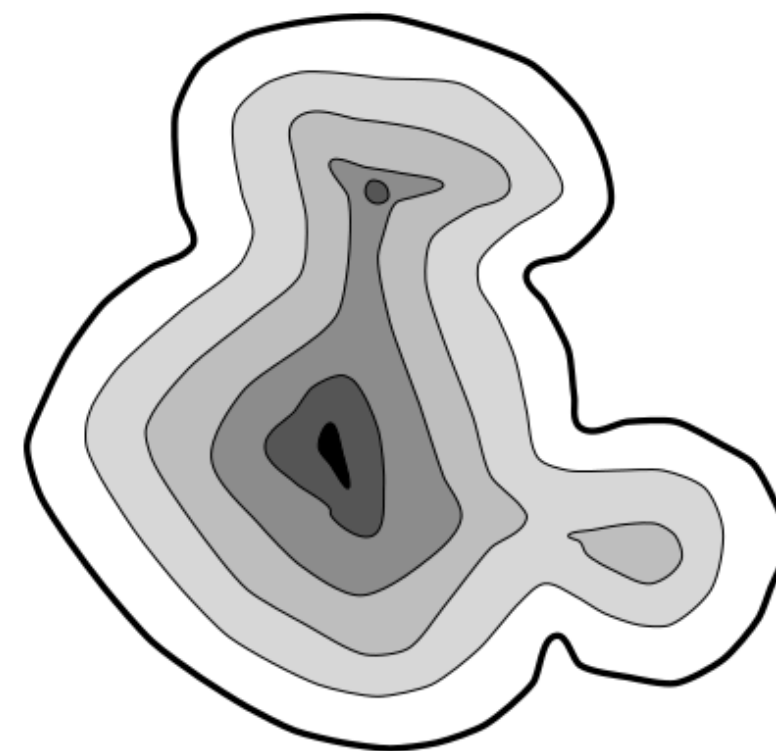
0	0	0	0	1	0	0
0	0	1	1	2	0	0
0	1	2	2	3	1	0
0	1	2	2	1	1	0
0	1	2	1	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0

0	0	0	0	1	0	0
0	0	1	1	1	0	0
0	1	2	2	2	1	0
0	1	2	2	1	1	0
0	1	2	1	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0

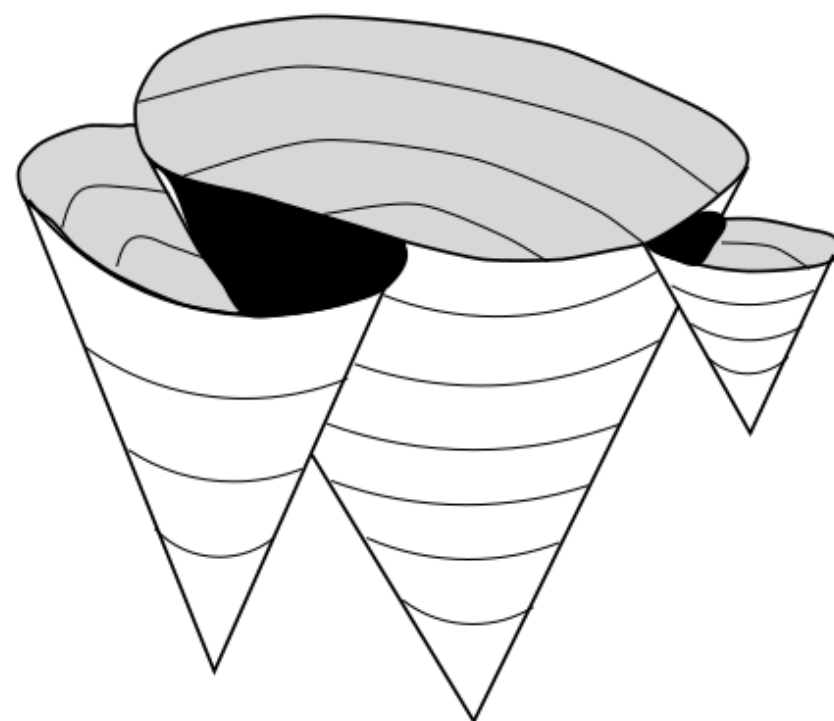
Watershed



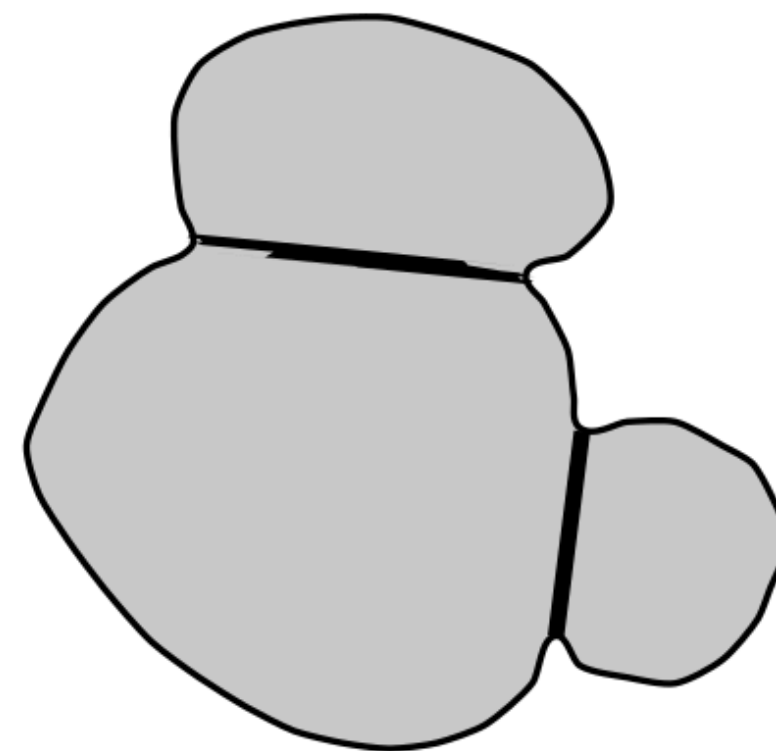
(a)



(b)



(c)



(d)

Hit or miss

$$X \otimes B = \{x : B_1 \subset X \wedge B_2 \subset X^c\}, B = (B_1, B_2)$$

0	0	0
don't care	1	don't care
1	1	1

Structuring element

- The structuring element has pixels with 3 values
 - ➔ 0: corresponding pixel must be 0
 - ➔ 1: corresponding pixel must be 1
 - ➔ don't care

Input pixels

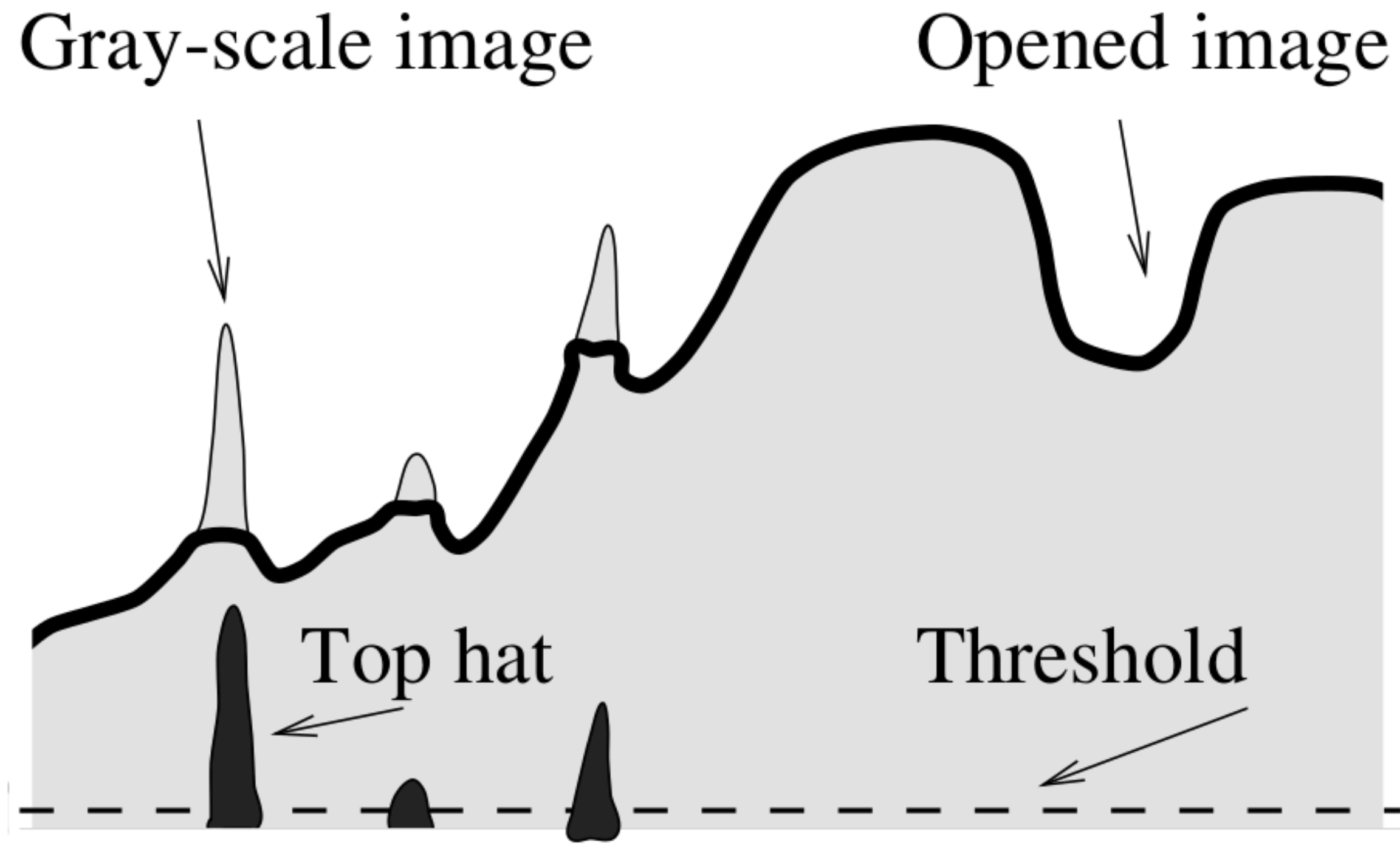
0	0	0
1	1	0
1	1	1

= true

0	0	1
1	1	0
1	1	1

= false

Top hat



Granulometry

