

# Razpoznavanje vzorcev

UN2-1-AV 2024/2025

## Razčlenjevanje vzorcev

Simon Dobrišek

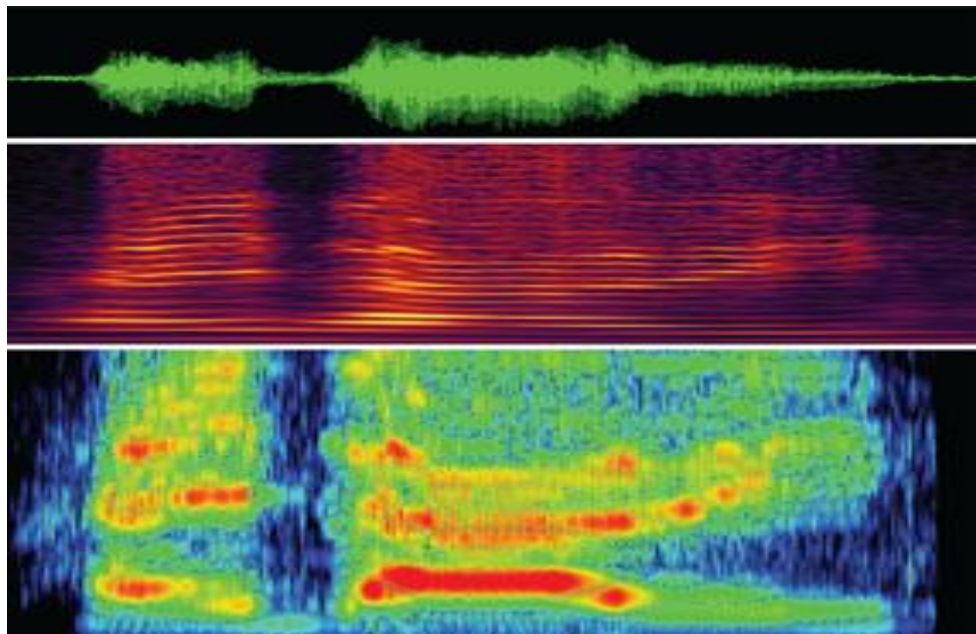
- Uvod
- Razčlenjevanje govornega signala
- Razčlenjevanje slik

# Razčlenjevanje zapletenih vzorcev

- Zapletene vzorce razčlenjujemo na samostojne enote, ki so *enovite* glede na izbrano lastnost vzorca.
- Na primer, sliko lahko razčlenimo na področja glede na svetilnost (sivi nivo) elementov slike, glede na barvo, glede na sestavo površine (teksturo) ipd.
- Vzorce razčlenjujemo na enote le, če v opisu vzorca zasledimo spremembe znotraj vzorca samega.
- Mesto večjih sprememb vrednosti funkcije, ki opisuje vzorce, lahko kaže na meje med enotami, vendar tega ne moremo sprejeti kot splošno pravilo.

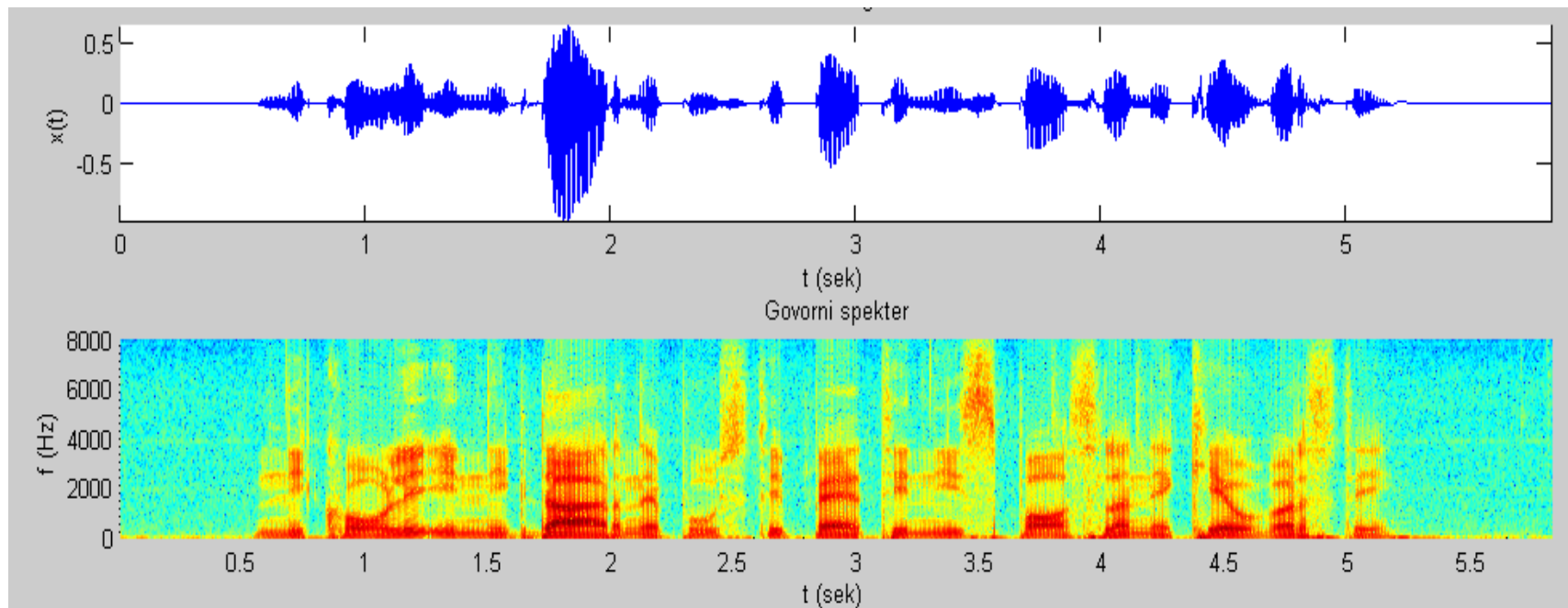
# RAZČLENJEVANJE GOVORNEGA SIGNALA

- Uvod v razčlenjevanje govornega signala
- Enote razpoznavanja govornega signala
- Postopki razčlenjevanja govornega signala



# Uvod v razčlenjevanje govornega signala

- Govorni signal vzorčimo kot **akustični signal**, pri katerem se navadno predpostavlja zgornjo frekvenčno mejo pri 8kHz.
- V govornem signalu ljudje zaznavamo kratko-časovno deloma **enovite vzorce - glasove**, ki se zaporedoma izmenjujejo.



# Enote razpoznavanja govornega signala

- **Besede** so najmanjše pomensko samostojne **enote povedi**, to je najmanjše enote najmanjšega možnega sporočila.
- Razčlenjevanje govora na besede je razmeroma preprosto, če se besede (ukazi) **izgovarjajo ločeno**.
- Besede pa pogosto izgovarjajo **brez premorov**, kar zelo oteži ali celo onemogoči razčlenjevanje govora na besede.
- **Fonem** ali **glas** je najmanjši gradnik govora, po katerem se lahko ena izgovorjena beseda po pomenu loči od druge.
- V besedah se glasovi razvrščajo po določenih načelih: okoli **samoglasnikov** se zbirajo **soglasniki** in skupaj z njimi tvorijo **zloge**.



# Razčlenjevanje govora na zloge

- Navadno je v besedi toliko zlogov, kolikor samoglasnikov vsebuje.
- Razčlenjevanje govora na zloge je razmeroma preprosta naloga - glasnost govora doseže lokalni **maksimum nad jedrom zloga** (nad samoglasnikom), med koncema sosednjih zlogov pa **lokalni minimum**.
- Vendar je razvrščanje zlogov zelo **težka naloga** zaradi velikega števila različnih zlogov.
- Število zlogov je odvisno od slovarja za dano področje uporabe in že pri majhnih slovarjih obsega nekaj tisoč zlogov.

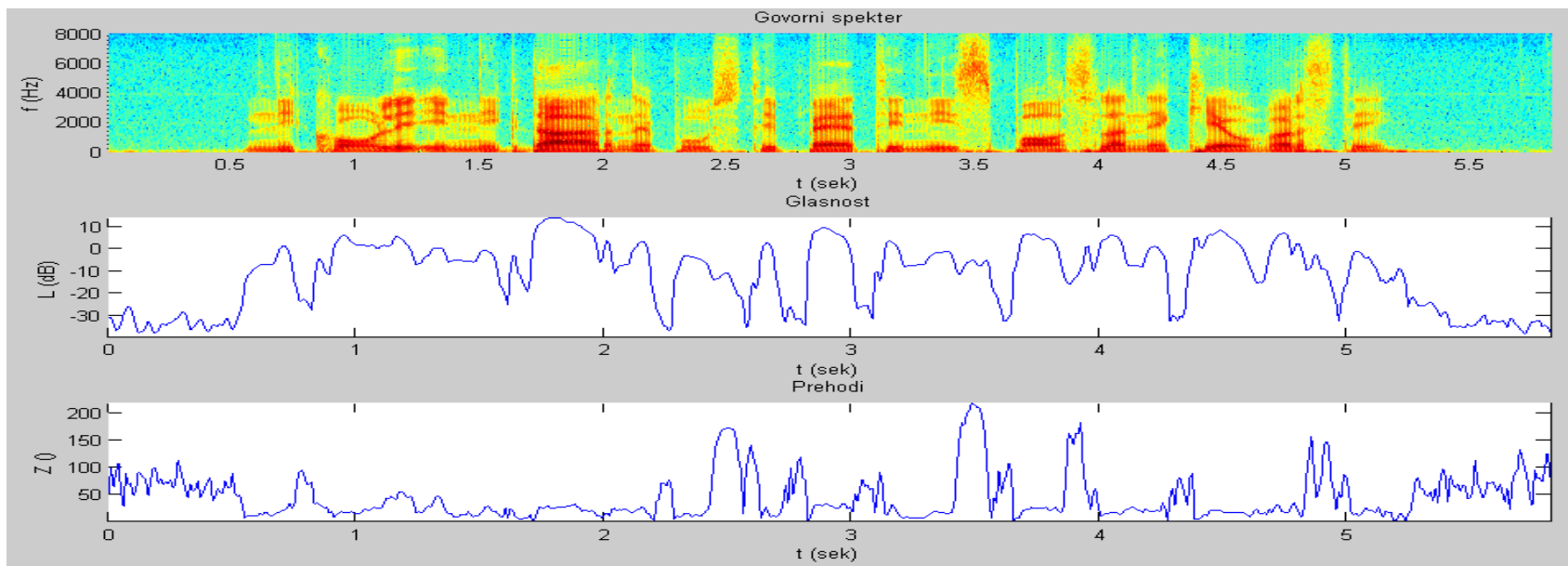
# Razčlenjevanje govora na alofone

- Glede na glasovno okolje, v katerem se nahajajo, dobijo glasovi poseben izgovor, kar zadeva njihovo trajanje, glasnost in zven.
- Različne oblike pojavljanja glasu imenujemo njegove **alofone**.
- Strokovnjaki, ki se ukvarjajo z glasoslovjem, lahko ločijo tudi do 200 alofonov, vendar jih je dejansko nekaj tisoč.
- V splošnem je razčlenjevanje govora na alofone lažje rešljivo od razčlenjevanja govora na glasove.



# Postopki razčlenjevanja govornega signala

- Govorni signal navadno razčlenjujemo na osnovi merjenja kratko-časovne **glasnosti govora**.
- Navadno se že z analizo kratko-časovne glasnosti in števila prehodov skozi ničlo lahko govorni signal razčleni na povedi, med katerimi so zaradi vdihov kratki premori.



# Ročno razčlenjevanje govornega signala

- Za pripravo učne množice govornih vzorcev govorne signale pogosto ročno razčlenjujemo na povedi in govorce ter transkribiramo/prepišemo s preposlušanjem posnetka.

The screenshot displays a speech analysis software interface. The top section shows a transcript of a conversation between a moderator and two speakers (speaker#9 and speaker#8). The transcript is as follows:

moderator  
Torej, v bližini inštitucij. Če je neka inštitucija, potem razmišljajmo, kdo tam vas gleda.  
speaker#9  
Varnostne službe.  
moderator  
..., ki je v teh ištitudiji?  
speaker#9  
... ali pa zunanji, tista ki skrbi za ...  
moderator  
ok.  
speaker#8  
Jaz sem danes naletel na bankomat, ki je imel kamero.  
moderator  
Torej pri bankomatu, kjer stojite. Tudi tam recimo, kdo vas gleda?  
speaker#8  
Lahko je že kakšen nov algoritem za preverjanje, če je upravnik ...  
moderator

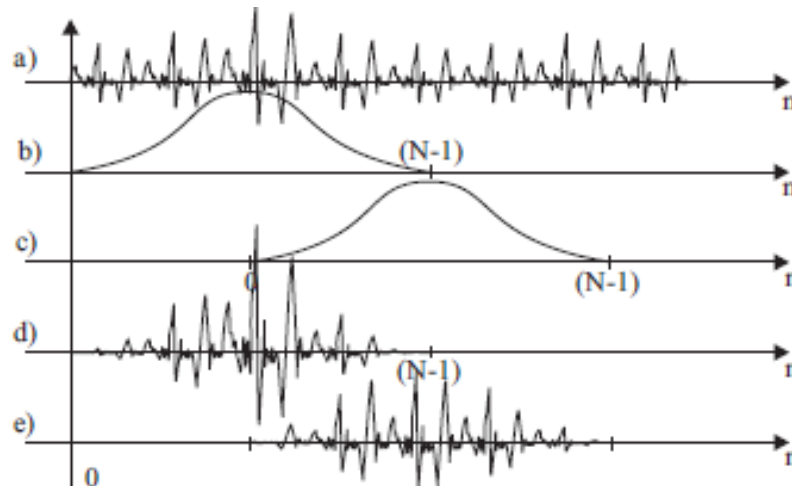
Below the transcript is a waveform visualization of the audio signal. A red vertical line marks a specific point in the audio, corresponding to the start of the segment highlighted in the transcript below. The waveform is labeled "RESPECTWP12FG1\_v9.July.2014\_Philips\_mono".

At the bottom, a table shows the segmentation of the audio into segments, with columns for speaker, start time, end time, and text. The table is as follows:

speaker#9	moderator	.s	moderator	speaker#9	moder.	speaker#8	mo
pomagam, saj je ... ruge okoliščine.	Predvsem v... ... institucij.	Torej, v bližini inštitucij. Če je neka inštitucija, potem razmišljajmo, kdo tam vas gleda.	..., ki je v teh ištitudiji?	... ali pa ... ... ki skrbi za ...	ok.	Jaz sem danes ... ... imel kamero.	Torej pri bankom tam recimo.
13:20	13:25	13:30	13:35	13:40	13:45		

# Razčlenjevanje govornega signala na izseke

- Govorni signal navadno razčlenjujemo z izbrano okensko funkcijo na zaporedne prekrivajoče-se **kratke izseke** z enakim trajanjem (navadno 10-30ms, s prekrivanjem 5-10ms).
- Tako kratki izseki govornega signala so krajši tudi od najkrajšega glasu, a dovolj dolgi, da odražajo kratko-časovne spektralne značilnosti izsekov glasov.



# RAZČLENJEVANJE SLIK

- Razčlenjevanje slik z ugotavljanjem področij
- Razčlenjevanje slik z ugotavljanjem obrisov



# Razčlenjevanje z ugotavljanjem področij

- Slika je sestavljena iz prostorsko **povezanih slikovnih elementov** — področij, ki so enovita glede na izbran kriterij enovitosti.
- **Kriterij enovitosti** področij je navadno svetilnost (sivi nivo) elementov slike, barva, sestava površine (tekstura) ipd.
- Področja na sliki se medsebojno **ne prekrivajo**, njihova unija pa je enaka sliki.
- Področja na sliki ugotavljamo s postopki **upravljanja**, z neposrednim **določanjem področij** ali z **rojenjem slikovnih elementov**.

# Razčlenjevanje z upragovljanjem

- Z upragovljanjem razčlenjujemo slike razmeroma enostavnih prizorov. Na primer, v industrijskih aplikacijah pogosto sestavlja prizor **predmet**, ki leži na določeni **enolični podlagi**.



- Takšne prizore lahko zelo hitro razčlenimo (ločimo predmet od podlage) na podlagi **porazdelitve relativnih frekvenc sivih nivojev** slike, saj je v tem primeru histogram porazdelitve bi-modalen z dvema izrazitima vrhovoma.
- En vrh histograma ustreza svetilnosti podlage, drugi pa svetilnosti predmeta.



# Porazdelitev relativnih frekvenc sivih nivojev slike

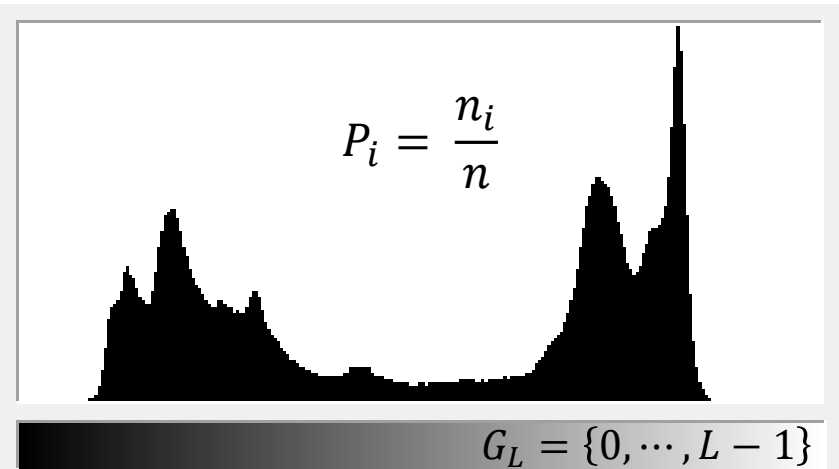
- Porazdelitev (histogram) relativnih frekvenc sivih nivojev slike je določena kot

$$P_i = \frac{n_i}{n}, \quad i = 0, \dots, L - 1$$

kjer sta:

$n_i$  število slikovnih elementov s sivim nivojem  $i \in G_L$  ter

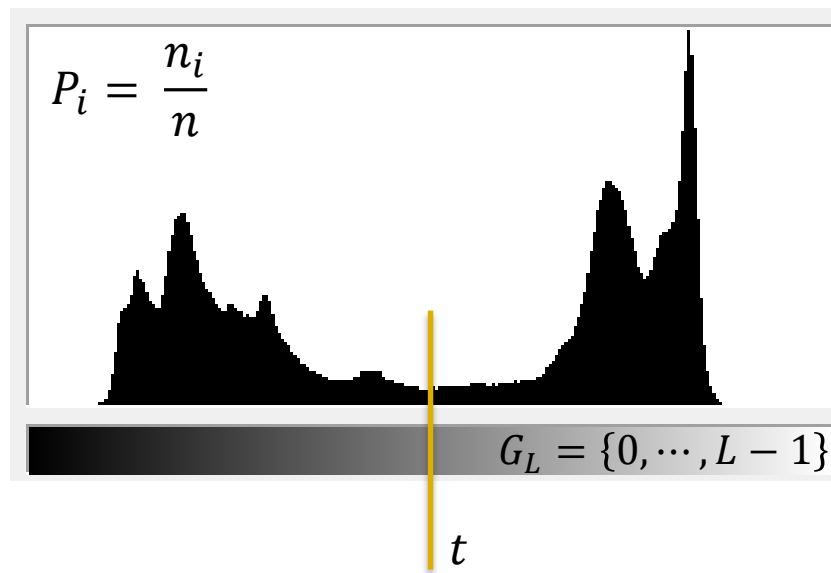
$n$  število vseh slikovnih elementov slike ( $n = M \cdot N$ ).





# Razčlenjevanje z upragovljanjem

- Iz histograma lahko določimo mejo, to je prag med sivimi nivoji podlage in sivimi nivoji predmeta.
- S pragom lahko razvrstimo vse slikovne elemente v dva razreda: v razred  $\omega_1 = \text{"podlaga"}$  in v razred  $\omega_2 = \text{"predmet"}$ .



# Upragovljjanje sivih slik

- Digitalno sivo sliko  $f(x, y)$  matematično zapišemo s preslikavo, ki vsaki točki slike (slikovnemu elementu) priredi neko svetilnost (sivi nivo):

$$f: I_M \times I_N \rightarrow G_L ,$$

- kjer sta  $I_M = \{0, \dots, M - 1\}$  in  $I_N = \{0, \dots, N - 1\}$  podmnožici celih ne-negativnih števil ter  $G_L = \{0, \dots, L - 1\}$  množica diskretnih vrednosti svetilnosti (sivih nivojev).
- Pri t.i. osem-bitni sivi sliki, ker število sivih nivojev  $L$  enako 256, kjer vrednost svetilnosti 0 predstavlja povsem črno in vrednost svetilnosti 255 povsem belo barvo, torej,  $G_L = \{0, \dots, 255\}$ .

# Definicija upragovljanja slike

- Pri upragovljanju sliko  $f(x, y)$  prevedemo v sliko  $f_t(x, y)$ , ki je dana s preslikavo

$$f_t: I_M \times I_N \rightarrow \Omega ,$$

- ki vsaki točki slike priredi eno od (navadno dveh) oznak področja na sliki  $\Omega = \{\omega_1, \omega_2\}$  (npr.  $\omega_1 = \text{"podlaga"}$  in  $\omega_2 = \text{"predmet"}$  ali preprosto  $\omega_1 = \text{"0"}$  in  $\omega_2 = \text{"1"}$  ).
- Preslikavo  $f_t$  tako definiramo z izrazom

$$f_t(x, y) = \begin{cases} \omega_1 & f(x, y) > t \\ \omega_2 & f(x, y) \leq t \end{cases} ,$$

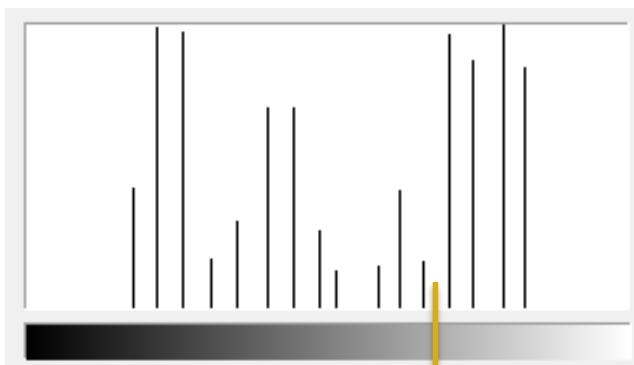
kjer je  $t$  prag svetilnosti, ki ločuje področja na sliki.

# Ponazoritev upragovljanja sive slike (1/2)

$$f(x, y) =$$

192,	198,	209,	189,	182,	209,	208,	209
183,	190,	122,	53,	54,	114,	206,	209
176,	149,	49,	59,	108,	53,	161,	209
175,	105,	101,	53,	60,	57,	114,	207
181,	93,	78,	71,	70,	117,	99,	204
187,	109,	46,	64,	45,	63,	126,	202
185,	152,	61,	107,	47,	66,	176,	200
180,	177,	131,	86,	96,	159,	191,	198

$$P_i = \frac{n_i}{n}$$



$G_L = \{0, \dots, 255\}$

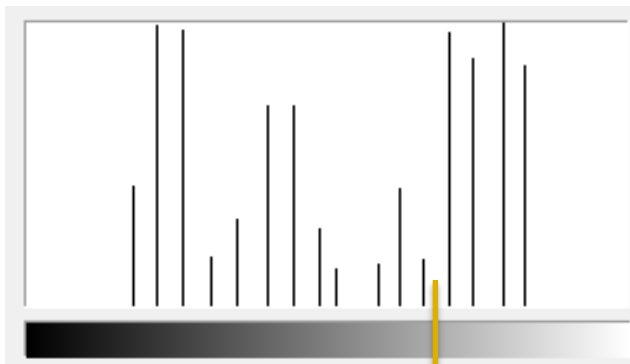
$t = 180$

# Ponazoritev upragovljanja sive slike (2/2)

$$f_t(x, y) =$$

$\omega_1$	$\omega_1$	$\omega_1$	$\omega_1$	$\omega_1$	$\omega_1$	$\omega_1$	$\omega_1$
$\omega_1$	$\omega_1$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_1$	$\omega_1$
$\omega_1$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_1$
$\omega_1$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_1$
$\omega_1$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_1$
$\omega_1$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_1$
$\omega_1$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_1$
$\omega_1$	$\omega_1$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_2$	$\omega_1$	$\omega_1$

$$P_i = \frac{n_i}{n}$$



$G_L = \{0, \dots, 255\}$

$t = 180$

# Razčlenjevanje z upragovljanjem

- Pri upragovljanju slik je prag v splošnem funkcija štirih spremenljivk:

$$t = t(x, y, f(x, y), p(x, y))$$

kjer so:

$x, y$  koordinati slikovnega elementa,

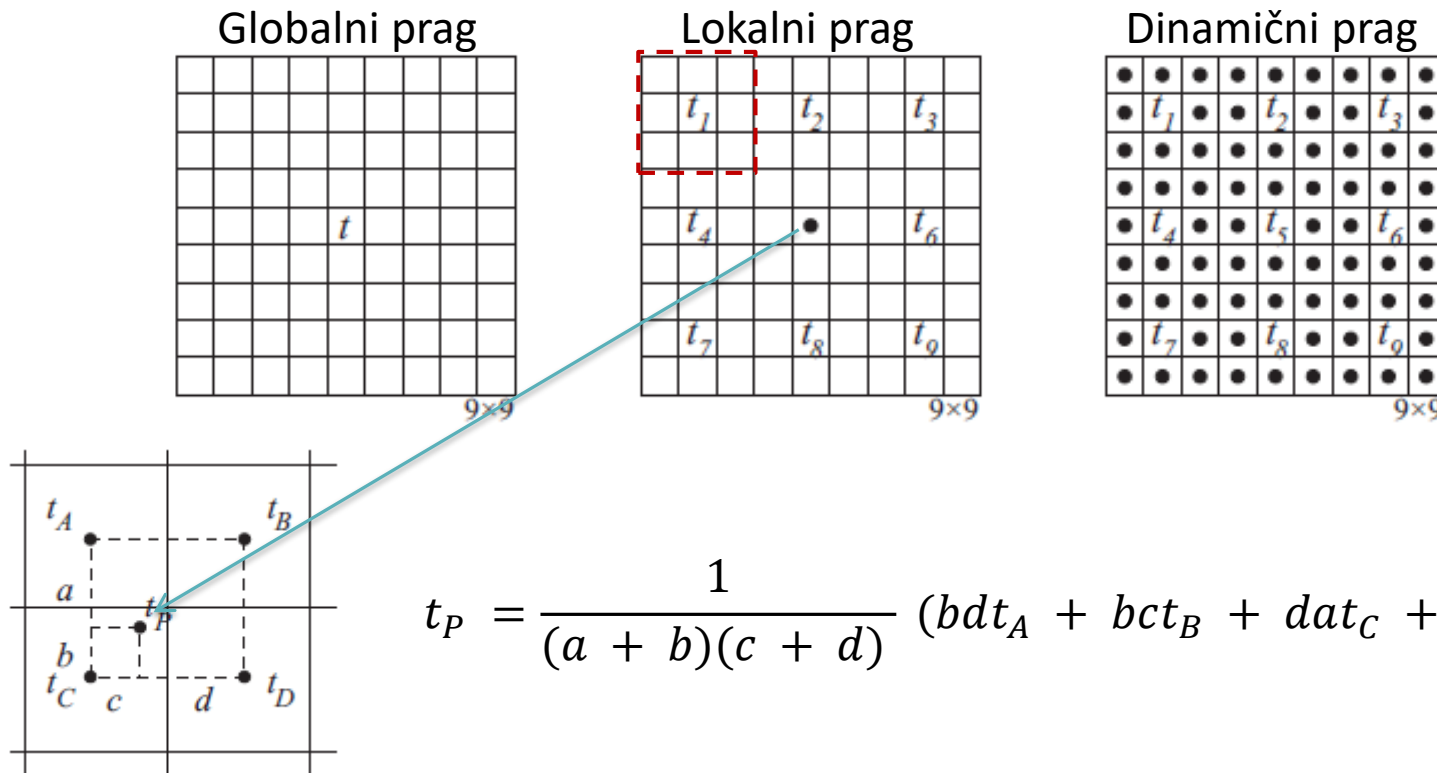
$f(x, y)$  svetilnost slikovnega elementa in

$p(x, y)$  lokalna lastnost, ki jo merimo v okolici slikovnega elementa s koordinatama  $(x, y)$ .

- Prag  $t$ , ki je odvisen samo od svetilnosti slikovnih elementov  $f(x, y)$ , je globalni prag.
- Prag  $t$ , ki je odvisen od  $f(x, y)$  in določene(-ih) lastnosti v okolici slikovnega elementa  $(x, y)$ , je lokalni prag.
- Prag  $t$ , ki je odvisen tudi od prostorskih koordinat  $x$  in  $y$  poleg  $f(x, y)$  in  $p(x, y)$ , je dinamični prag.

# Razčlenjevanje z upragovljanjem

- Upragovljjanje z globalnim, z lokalnim in z dinamičnim pragom.
- V podslikah oziroma v slikovnih elementih, ki so označene s piko, določimo prag z interpolacijo med izračunanimi pragovi.

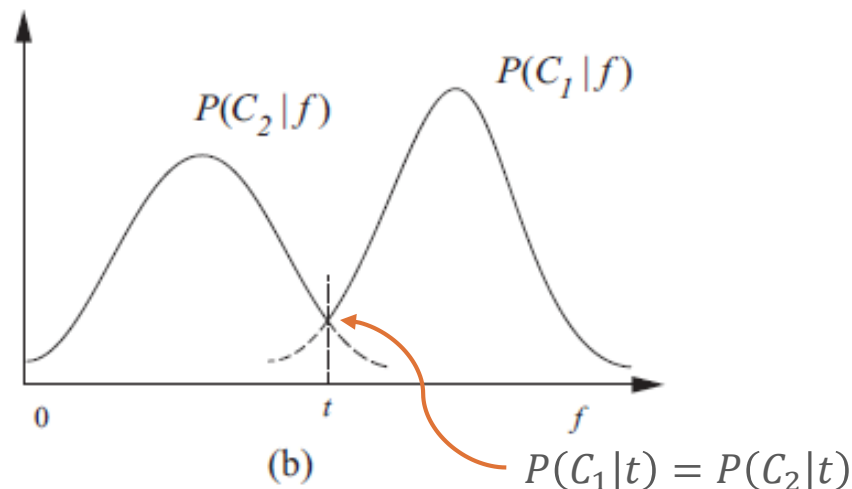
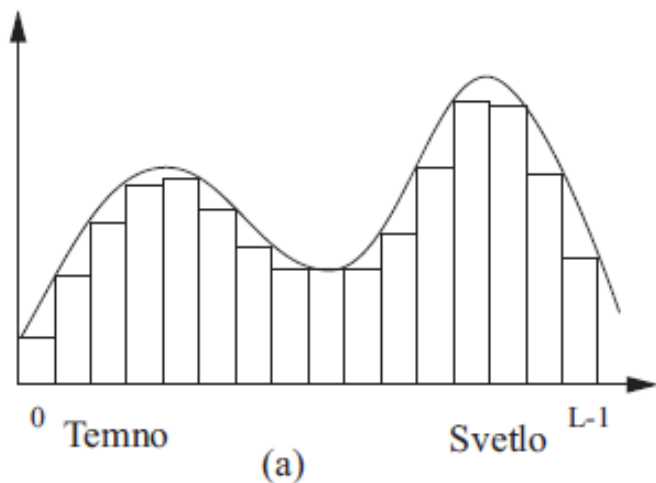


$$t_P = \frac{1}{(a+b)(c+d)} (bdt_A + bct_B + dat_C + cat_D)$$



# Upragovljjanje z najmanjšim tveganjem

- Aproksimacija histograma z vsoto dveh funkcij verjetnostne gostote



$$P(C_i|f) = \frac{P(C_i)P(f|C_i)}{P(f)} \quad \wedge \quad P(C_1|t) = P(C_2|t) \Rightarrow P(C_1)P(t|C_1) = P(C_2)P(t|C_2)$$

# Upragovljajanje z najmanjšim tveganjem

Najboljši prag  $t$  dobimo iz enačbe:

$$P(C_1)P(t \mid C_1) = P(C_2)P(t \mid C_2).$$

V primeru, če sta funkciji pogojnih gostot verjetnosti Gaussovi funkciji  $N(m_1, \sigma_1^2)$  in  $N(m_2, \sigma_2^2)$ , izračunamo najboljši prag iz enačbe:

$$At^2 + Bt + C = 0,$$

kjer so:

$$A = \sigma_1^2 - \sigma_2^2$$

$$B = 2(m_1\sigma_2^2 - m_2\sigma_1^2)$$

$$C = \sigma_1^2 m_2^2 - \sigma_2^2 m_1^2 + 2\sigma_1^2 \sigma_2^2 \ln \frac{\sigma_2 P(C_1)}{\sigma_1 P(C_2)}.$$

# Upragovljjanje z najmanjšim tveganjem

- Določanje  $(m_1, \sigma_1)$  in  $(m_2, \sigma_2)$  iz porazdelitve relativnih frekvenc sivih nivojev slike.

## Postopek

1. korak: Porazdelitev relativnih frekvenc sivih nivojev  $P_i$ ;  $i = 0, 1, \dots, L - 1$  zgladimo z omejenim obteženim povprečjem:

$$P_i^* = \frac{P_{i-2} + 2P_{i-1} + 3P_i + 2P_{i+1} + P_{i+2}}{9}; \quad i = 2, 3, \dots, L - 3.$$

2. korak: Poiščemo dva najbolj pogosta siva nivoja<sup>b</sup>  $v_s, v_z \in G_L$ , ki sta medsebojno oddaljena najmanj  $D_{min}$  sivih nivojev (navadno najmanj 30).
3. korak: Določimo sivi nivo  $v \in G_L$ , ki je najmanj pogost med dvema modusoma porazdelitve  $P_i^*$ .
4. korak:  $m_1, \sigma_1, m_2$  in  $\sigma_2$  ocenimo iz enačb:

$$P_v = \sum_{i=0}^v P_i^*.$$

$$\begin{aligned} m_1 &= \frac{1}{P_v} \sum_{i=0}^v iP_i^*, & \sigma_1 &= \sqrt{\frac{1}{P_v} \sum_{i=0}^v (i - m_1)^2 P_i^*}, \\ m_2 &= \frac{1}{1 - P_v} \sum_{i=v+1}^{L-1} iP_i^*, & \sigma_2 &= \sqrt{\frac{1}{1 - P_v} \sum_{i=v+1}^{L-1} (i - m_2)^2 P_i^*}, \end{aligned}$$

$$P(C_1) = P_v \quad \text{in} \quad P(C_2) = 1 - P_v$$

# Upragovljjanje z maksimizacijo variance

- Slikovne elemente slike glede na vrednost praga razdelimo na dve podmnožici – z elementi pod in nad vrednostjo praga.
- Za optimalni prag izberemo tisto vrednost, ki jo določa:

$$t^* = \arg \operatorname{Max}_{t \in G_L} \left\{ \frac{\sigma_B^2}{\sigma_T^2} \right\}$$

kjer sta  $\sigma_B^2$  varianca sivih nivojev med podmnožicama ter  $\sigma_T^2$  celotna varianca sivih nivojev:

$$\begin{aligned}\sigma_T^2 &= \sum_{i=0}^{L-1} (i - \mu_T)^2 P_i, & \mu_T &= \sum_{i=0}^{L-1} i P_i, \\ \sigma_B^2 &= P^*(t)(1 - P^*(t))(\mu_1 - \mu_0)^2, & P^*(t) &= \sum_{i=0}^t P_i, \\ \mu_1 &= \frac{\mu_T - \mu_t}{1 - P^*(t)}, & \mu_0 &= \frac{\mu_t}{P^*(t)}, & \mu_t &= \sum_{i=0}^t i P_i.\end{aligned}$$

# Upragovljjanje z maksimizacijo informacije (1/2)

- Če sliko upragovljamo s pragom  $t$ , delimo množico sivih nivojev slike  $G_L$  na dve tuji množici  $G_0 \cap G_1 = \emptyset$ , kjer sta  $G_0 = \{0, \dots, t\}$  in  $G_1 = \{t + 1, \dots, L - 1\}$ .
- Množicama  $G_0$  in  $G_1$  pripadata porazdelitvi relativnih frekvenc sivih nivojev

$$\mathbf{P}_0 = \mathbf{P}_0(t) = \left\{ \frac{P_0}{P^*(t)}, \dots, \frac{P_t}{P^*(t)} \right\}$$

in

$$\mathbf{P}_1 = \mathbf{P}_1(t) = \left\{ \frac{P_{t+1}}{1 - P^*(t)}, \dots, \frac{P_{L-1}}{1 - P^*(t)} \right\} ,$$

kjer je

$$P^*(t) = \sum_{i=0}^t P_i \quad .$$

# Upragovljjanje z maksimizacijo informacije (2/2)

- Para  $(G_0, \mathbf{P}_0)$  in  $(G_1, \mathbf{P}_1)$  lahko obravnavamo kot verjetnostni shemi dveh naključnih spremenljivk z informacijama (entropijama)

$$H_0(t) = - \sum_{i=0}^t \frac{P_i}{P^*(t)} \log \left( \frac{P_i}{P^*(t)} \right)$$

in

$$H_1(t) = - \sum_{i=t+1}^{L-1} \frac{P_i}{1 - P^*(t)} \log \left( \frac{P_i}{1 - P^*(t)} \right) .$$

- Optimalni prag svetilnosti  $t^*$ , lahko določimo kot prag, ki maksimizira celotno informacijo upragovljene slike, to je:

$$t^* = \arg \max_{t=0, \dots, L-2} \{H_0(t) + H_1(t)\}$$

- Optimalni prag določimo z računanjem informacije za vse možne vrednosti praga in ugotavljanjem, pri kateri vrednosti praga je vrednost informacije največja.

# Algoritem za določanje vrednosti praga

- Prag, ki maksimizira informacijo upragovljene slike, poiščemo z naslednjim algoritmom:

*Postopek*

$max := 0; t^* = 0;$

**za**  $t := 0$  **do**  $L - 2$  **naredi**

“Izračunaj  $(H_0(t) + H_1(t))$ ”;

**če**  $(H_0(t) + H_1(t)) > max$  **potem**

$max := (H_0(t) + H_1(t));$

$t^* := t;$

**konec-če**

**konec-za**

*Konec postopka.*



# Neposredno razčlenjevanje slike

- Sliko lahko predstavimo s področjem  $R$ , ki vsebuje (pod)področja  $R_1, R_2, \dots, R_K$ , če le-ta zadoščajo naslednjim pogojem:

1.  $\bigcup_{i=1}^K R_i = R$ ,
2.  $R_i$  je področje povezanih slikovnih elementov;  $i = 1, 2, \dots, K$ ,
3.  $R_i \cap R_j = \emptyset$  za  $\forall i \neq j$ ,
4.  $P(R_i) = 1$  za  $i = 1, 2, \dots, K$ ,
5.  $P(R_i \cup R_j) = 0$  za  $i \neq j$  in sosednja  $R_i, R_j$ ,

kjer je  $P(R_i)$  logični predikat, ki je enak 1, če slikovni elementi iz področja  $R_i$  zadoščajo kriteriju enovitosti (homogenosti)  $P$ . Na primer,  $P(R_i) = 1$ , če imajo vsi slikovni elementi iz  $R_i$  enake svetilnosti.

- Že predstavljeni postopki razčlenjevanje z upragovljanjem so poseben primer neposrednega določanja področij, za katere velja:

$$P(R_i) = 1, \text{ če je } t_i - 1 \leq f(x, y) \in R_i \quad \forall t_i, i = 1, 2, \dots, K$$

# Združevanje področij

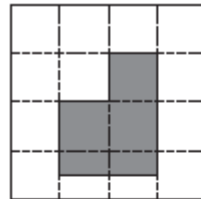
- To je postopek, ki **združuje** manjša področja v večja področja, če je izpolnjen postavljeni kriterij.
- Postopek začnemo z določanjem atomarnih področij — **semen**. Le-teh je lahko največ toliko, kolikor je slikovnih elementov na dani sliki, vendar navadno izberemo okvirje slikovnih elementov razsežnosti  $2 \times 2$ ,  $4 \times 4$  ali  $8 \times 8$ .
- Za vsako atomarno področje izračunamo **statistiko področja** (npr. povprečno svetilnost slikovnih elementov ali varianco slikovnih elementov, frekvenčno porazdelitev, itn.).
- Statistiko vsakega atomarnega področja nato **primerjamo** s statistikami sosednjih atomarnih področij.
- Ko sta si statistiki dveh sosednjih atomarnih področij **podobni**, združimo področji v eno področje ter za novo področje ponovno ovrednotimo statistiko.

# Razdruževanje področij

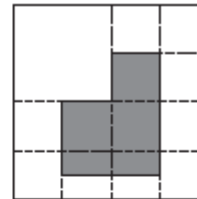
- Opisani postopek lahko začnemo tudi tako, da za začetek vzamemo, da je **celotna slika** samo eno področje  $R$ .
- Nato **razdružimo** področje  $R$  na nekaj podpodročij (navadno štiri enaka), za katere za vsako posebej ugotavljamo, če zadošča pogoju enovitosti.
- V kolikor ne, razdružimo to podpodročje na nova, po obsegu še **manjša** podpodročja.
- Postopek razdruževanja področij nadaljujemo toliko časa, dokler ni zadoščeno **vsem** omenjenim petim **pogojem**.

# Združevanje in razdruževanje področij

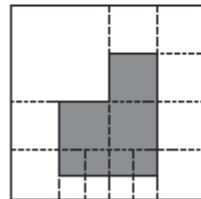
- Postopek razčlenjevanja slike z razdruževanjem in z združevanjem področij združuje dobre lastnosti obeh predhodnih postopkov.



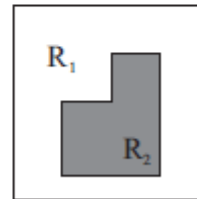
1. korak



2. korak



3. korak

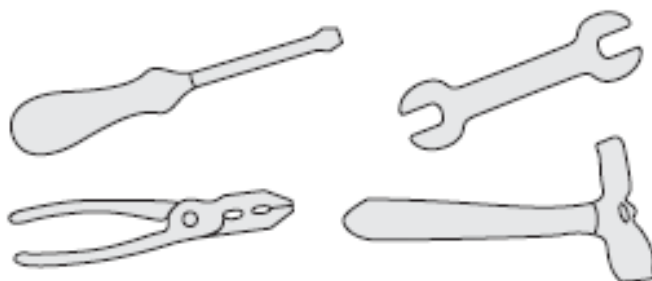


4. korak

Ponazoritev postopka razčlenjevanja slike z razdruževanjem in z združevanjem področij.

# Razčlenjevanje slike z ugotavljanjem obrisov

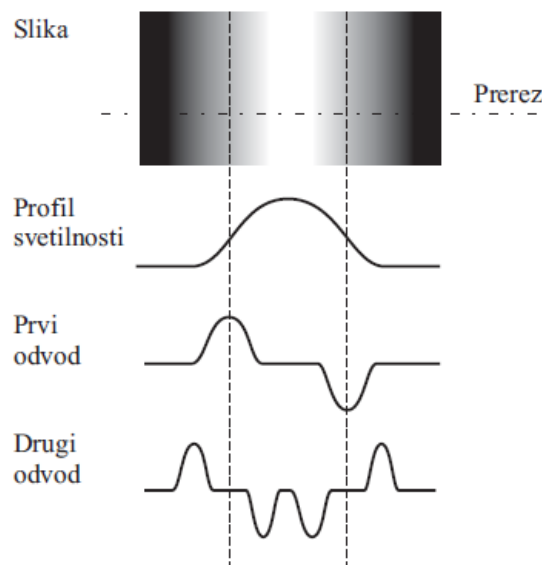
- Poizkusi z biološkimi vidnimi sistemi so pokazali, da so **obrisi objektov**, ki jih vsebuje dan prizor, zelo pomembni za razpoznavanje in za tolmačenje prizora.



- Določevanje obrisov objektov sloni na ugotavljanju naglih sprememb svetilnosti (sivega nivoja) slikovnih elementov, do katerih pride na **robvih objektov**.

# Postopki, ki temeljijo na odvajanju slikovne funkcije

- Robove področij, to je nagle spremembe svetilnosti slikovnih elementov, lahko odkrijemo z odvajanjem slikovne funkcije v vsaki točki slike.
- Algoritmi za ugotavljanje robnih točk temeljijo na ugotavljanju mest hitrih sprememb sivega nivoja slike.



# Ugotavljanje robov z gradientnimi operatorji

- Odvajanjem slike se izvaja z računanjem gradienta slike v vsaki točki  $(x, y)$ .
- Gradient slike  $f(x, y)$  v točki  $(x, y)$  je definiran z dvorazsežnim vektorjem:

$$\nabla(f(x, y)) = \begin{bmatrix} f_x(x, y) \\ f_y(x, y) \end{bmatrix} = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix}$$

- Vse točke slike z dolžino gradienta, večjo od nič, so možne točke roba.
- Gradient digitalne slike lahko računamo s pomočjo **Robertsovega**, **Sobelovega** ali **Prewittovega** operatorja.



# Sobelov operator

- Opisali bomo le Sobelov gradientni operator, ki ga uporabljamo za ugotavljanje robov objektov na slikah z **malo šuma**, ali na slikah, ki smo jim predhodno **odstranili šum**.
- S pomočjo Sobelovega operatorja izračunamo gradient slike v točki  $(x, y)$  z upoštevanjem **osmih** najbližjih sosedov te točke:

$$\begin{aligned}f_x(x, y) &= \frac{\partial f(x, y)}{\partial x} = (f(x+1, y+1) + 2f(x+1, y) + f(x+1, y-1)) - \\&\quad - (f(x-1, y+1) + 2f(x-1, y) + f(x-1, y-1)) \\&= (s_1 + 2s_0 + s_7) - (s_3 + 2s_4 + s_5) \\f_y(x, y) &= \frac{\partial f(x, y)}{\partial y} = (f(x+1, y+1) + 2f(x, y+1) + f(x-1, y+1)) - \\&\quad - (f(x+1, y-1) + 2f(x, y-1) + f(x-1, y-1)) \\&= (s_1 + 2s_2 + s_3) - (s_7 + 2s_6 + s_5),\end{aligned}$$

kjer smo s  $s_0$  do  $s_7$  označili sosede točke  $(x, y)$ .

# Sobelov operator

- Sobelov operator lahko zapišemo tudi s spodnjima “maskama”.
- Z maskama izračunamo odvod slike v točki  $(x, y)$  tako, da osrediščimo masko na točko  $(x, y)$ , zmnožimo svetilnosti slikovnih elementov z istoležnimi koeficienti v maski in zmnožke seštejemo.
- Vsota predstavlja vrednost delnega odvoda  $f_x(x, y)$  oziroma  $f_y(x, y)$  v točki  $(x, y)$ .

$s_3$	$s_2$	$s_1$
$s_4$	$(x, y)$	$s_0$
$s_5$	$s_6$	$s_7$

Sosedje

-1	0	1
-2	0	2
-1	0	1

Maska za  $f_x(x, y)$

1	2	1
0	0	0
-1	-2	-1

Maska za  $f_y(x, y)$

# Postopek ugotavljanja robnih točk

- Za vsak slikovni element  $(x, y)$  izračunamo dolžino gradienta (moč roba):

$$D(x, y) = \sqrt{f_x^2(x, y) + f_y^2(x, y)}$$

ali (računsko preprosteje)

$$D(x, y) = |f_x(x, y)| + |f_y(x, y)|$$

in *smer* gradienta (“usmerjenost” roba):

$$\theta(x, y) = \arctan \left( \frac{f_y(x, y)}{f_x(x, y)} \right)$$

- Možne točke roba so vse tiste točke slike  $f(x, y)$ , za katere je  $D(x, y) > 0$ .



# Obdelava slik robnih točk

- Vse tiste točke  $(x, y)$  slike, za katere je  $D(x, y) > 0$ , so možne točke robov, vendar je dolžina gradienta v številnih točkah slike zelo majhna.
- Prav te točke zelo verjetno niso točke robov področij na sliki, zato jih iz množice možnih robnih točk izločimo.
- Opisali bomo tri postopke manjšanja množice možnih robnih točk.

# Upragovljjanje slike moči robnih točk

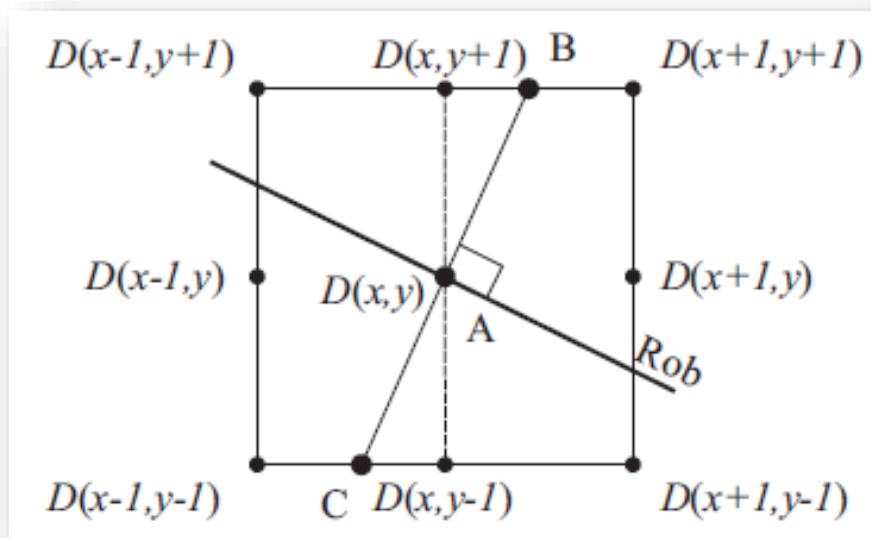
- Z upragovljjanjem slike dolžin gradienta  $D(x, y)$  obdržimo v množici možnih robnih točk le tiste točke slike, katerih dolžina gradienta ni manjša od določenega praga:

$$D'(x, y) = \begin{cases} D(x, y), & \text{če } D(x, y) > \gamma \\ 0, & \text{drugače,} \end{cases}$$

- kjer je  $\gamma$  prag, ki ga določimo iz histograma dolžin gradienta tako, da na sliki  $D'(x, y)$  obdržimo med 60% in 90% dolžin gradienta s slike  $D(x, y)$ .
- Možne točke roba so vse tiste točke slike  $f(x, y)$ , za katere je  $D'(x, y) > 0$ .

# Odpravljanje ne-največjih vrednosti iz slike moči robnih točk (1/2)

- Postopek temelji na primerjanju dolžine gradienta vsake točke gradientne slike (točka A) z dolžinami gradienta v (interpoliranih) točkah B in C, ki ležita na smernici gradienta točke A.



Osem sosedov točke A in interpolirani točki B in C

# Odpravljanje ne-največjih vrednosti iz slike moči robnih točk (2/2)

- Točka gradientne slike (točka A) je možna točka roba, če je dolžina gradienta v točki A,  $D_A$  večja od dolžin gradienta v točkah B,  $D_B$ , in C,  $D_C$ .

$$D'(x, y) = \begin{cases} D(x, y), & \text{če } D(x, y) > \text{Max}\{D_B, D_C\} \\ 0, & \text{drugače.} \end{cases}$$

- Dolžini gradienta v interpoliranih točkah B in C lahko izračunamo iz obrazcev:

$$D_B = \frac{D(x+1, y+1)}{\text{tg } \theta(x, y)} + D(x, y+1) - \frac{D(x, y+1)}{\text{tg } \theta(x, y)}$$
$$D_C = \frac{D(x-1, y-1)}{\text{tg } \theta(x, y)} + D(x, y-1) - \frac{D(x, y-1)}{\text{tg } \theta(x, y)}$$

# Tanjšanje slike robnih točk

- Zaradi zelo gostega odtipavanja slikovne funkcije, neostrih robov objektov razpoznavanja, šuma ipd. so robovi objektov na sliki pogosto **široki več slikovnih elementov**.
- Za določene postopke povezovanja robnih točk v obrise, ki jih bomo obravnavali v nadaljevanju, je zelo pomembno, da robove področij na sliki predhodno **stanjšamo** na širino **enega** slikovnega elementa.
- Predstavljen bo morfološki postopek tanjšanja, ki stanjša robove do širine enega slikovnega elementa, ohrani pa krajišči robov.



# Tanjšanje slike robnih točk

- Iz uprakovljene slike  $D'(x, y)$  najprej naredimo (pomožno) binarno sliko  $B(x, y)$ :

$$B(x, y) = \begin{cases} 1, & \text{če } D'(x, y) > 0 \\ 0, & \text{drugače.} \end{cases}$$

- Nato opravimo tanjšanje robov nad sliko  $B(x, y)$  s postopkom:

*Postopek*

```
dokler "prihaja do sprememb" naredi
  za "vsako masko  $\mathcal{W}_i$ " naredi
    za "vsak  $(x, y)$ , za katerega je  $B(x, y) = 1$ " naredi
      "primerjaj vrednosti 8-ih sosedov slikovnega elementa
       $(x, y)$  z vrednostmi maske  $\mathcal{W}_i$ , ki so položeni nanje";
      če "vse vrednosti maske  $\mathcal{W}_i$  in istoležnih
      slikovnih elementov na sliki  $B$  sovpadajo", potem
         $B(x, y) := 0$ ;
      konec-če
    konec-za
  konec-za
konec-dokler
```

*Konec postopka.*

# Tanjšanje slike robnih točk

- Spodaj sta ponazarja dve izmed osmih mask postopka .
- Preostalih šest mask dobimo tako, da maski  $\mathcal{W}_1$  in  $\mathcal{W}_2$  še trikrat zavrtimo, vsakič za  $90^\circ$  .
- Zvezdice v posameznih poljih mask pomenijo, da sovpadanje teh točk mask z istoležnimi točkami slike  $B(x, y)$  ni zahtevano.

0	0	0
*	1	*
1	1	1

$\mathcal{W}_1$

*	0	0
1	1	0
*	1	*

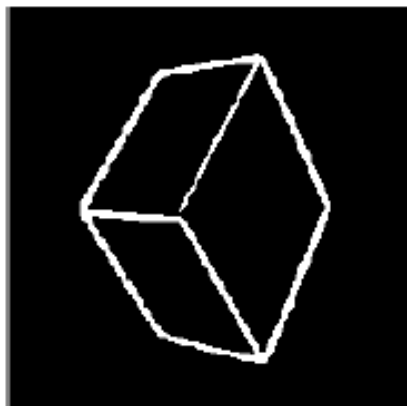
$\mathcal{W}_2$

# Tanjšanje slike robnih točk

- Rezultat postopka tanjšanja slike dolžin gradienta  $D(x, y)$  je nova slika dolžin gradienta:

$$D(x, y) = \begin{cases} D'(x, y), & \text{če } B(x, y) = 1 \\ 0, & \text{drugače.} \end{cases}$$

- Možne točke roba so vse tiste točke slike, za katere je  $D(x, y) > 0$



Upragovljena slika  
dolžin gradienta



Slika s stanjšanimi  
robovi

# Povezovanje robnih točk v obrise področij

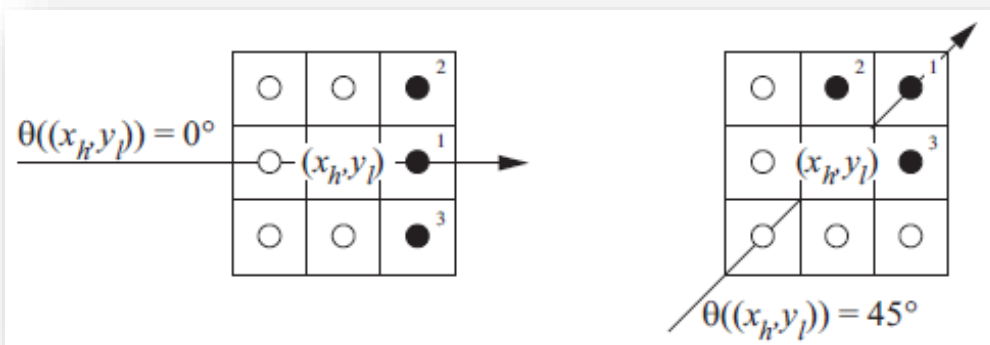
- Zaradi šuma v slikah s postopki ugotavljanja robnih točk **ne moremo natančno** določiti obrisov področij.
- Množice robnih točk, ki jih dobimo z gradientnim operatorjem, predstavljajo samo **napotek**, kje naj iščemo obris.
- Po postopkih za ugotavljanje robnih točk zato izvajamo še postopke za **povezovanje robnih točk** v obrise področij.
- Ti postopki so lahko **lokalni**, če je povezovanje omejeno s sosedstvom robnih elementov, oziroma so **globalni**, če to omejitev opustimo.

# Lokalni postopek povezovanja robnih točk (1/4)

- S postopkom Nevatije in Babuja obdelamo sliko dolžin gradienta  $D(x, y)$ , ki smo jo predhodno “zožili” in “stanjšali”.
- Pri obdelavi si pomagamo tudi z informacijo o robnih točkah, ki jo vsebuje slika smeri gradienta  $\theta(x, y)$ .
- V prvem prehodu slike  $D(x, y)$  preverjamo neposredne sosede robne točke  $(x_k, y_l)$ , ki so najbližje smeri  $\theta((x_k, y_l))$ .
- Izmed teh treh točk izberemo tisto, katere usmerjenost je najbližja usmerjenosti  $\theta((x_k, y_l))$ .
- Če ima več robnih točk enako smer, izberemo tisto, katere moč roba (dolžina gradienta) je večja.

# Lokalni postopek povezovanja robnih točk (2/4)

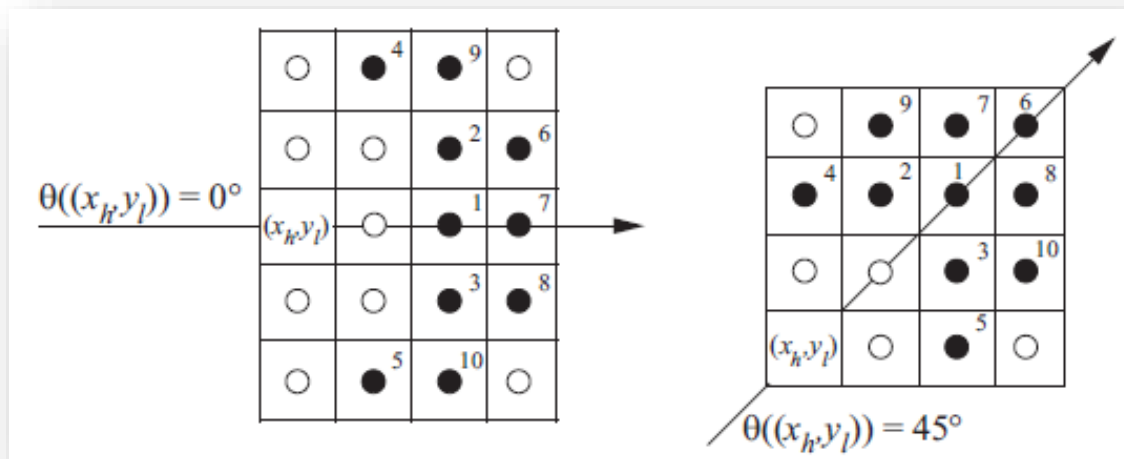
- V drugem prehodu slike  $D(x, y)$  povežemo vse robne točke slike v obrise.
- Sliko  $D(x, y)$  pregledujemo vrstico za vrstico, da bi ugotovili morebiten začetek obrisa.
- Če moč roba trenutne točke presega zahtevani prag in ta točka še ne pripada nobeni krivulji, jo označimo kot začetno točko nove krivulje in ji priredimo novo številko (odseka) obrisa.



Primera morebitnih naslednikov robne točke  $(x_k, y_l)$

# Lokalni postopek povezovanja robnih točk (3/4)

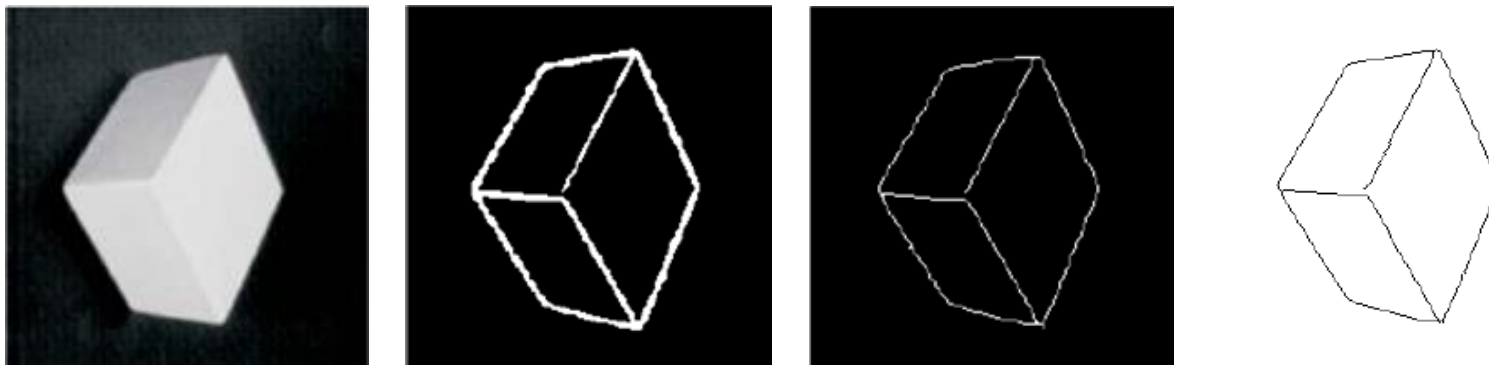
- Pri izvajanju postopka poskušamo premostiti manjše prekinitve obrisov, če menimo, da so le-te posledica šuma.
- Na primer dovolimo, da med dvema robnima točkama sme manjkati največ ena točka.



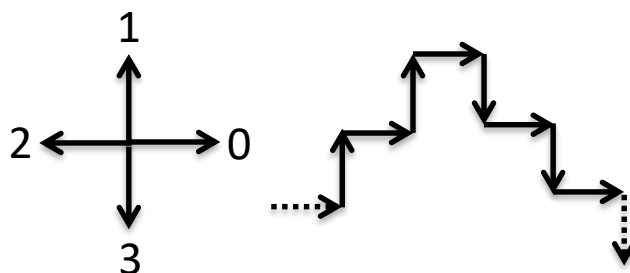
Primer določevanja naslednika robne točke  $(x_k, y_l)$  pri skoku  
čez "luknjo" v obrisu, ki je "široka" en slikovni element

# Lokalni postopek povezovanja robnih točk (4/4)

- Rezultat izvajanja postopka so verižni kodi (denimo štiri-ali osem-smerna) obrisov področij na vhodni sliki.



Rezultat obdelave slike kvadra po postopku Nevatije in Babuja



Primer dela štiri-smerne verižne kode: ... 10103030 ...



# Povezovanje robnih točk s histereznim upragovljajanjem (1/2)

- Histerežno upragovljajanje je globalni postopek povezovanja robnih točk v obrise področij na sliki (izvajan po „stanjšanju“).
- Osnovna zamisel histereznega upragovljajanja je naslednja:
  - med vsemi možnimi točkami roba izberemo tiste točke,
  - katerih dolžina gradienta je **večja od zgornjega praga**  $t_z$  in
  - ležijo na **povezani verigi točk**, ki jo sestavljajo točke z dolžino gradienta večjo od spodnjega praga  $t_s$ .
- Razmerje med spodnjim in zgornjim pragom podaja koeficient  $\delta$ :

$$t_s = \delta t_z,$$

kjer je  $\delta \in [0,3, 0,5]$ , navadno  $\delta = 0,33$ .

# Povezovanje robnih točk s histereznim upravljanjem (2/2)

## *Postopek*

1. korak: Določimo dve pomožni sliki  $h_z(x, y)$  in  $h_s(x, y)$  tako, da velja za vsako točko  $(x, y)$ :

$$h_z(x, y) = 0, \quad \text{če je} \quad D(x, y) < t_z,$$

$$h_z(x, y) = 1, \quad \text{če je} \quad D(x, y) \geq t_z,$$

$$h_s(x, y) = 0, \quad \text{če je} \quad D(x, y) < t_s,$$

$$h_s(x, y) = 1, \quad \text{če je} \quad D(x, y) \geq t_s,$$

kjer je  $D(x, y)$  dolžina gradienta v točki  $(x, y)$ .

2. korak: V prvem prehodu ( $k = 1$ ) slike dolžin gradienta  $D(x, y)$  poiščemo vse tiste točke  $(x, y)$ , za katere velja  $h_z(x, y) = 1$ , jih označimo s  $k$  in pregledamo vseh 8 sosednjih točk. Če katerakoli sosednja točka ustreza pogoju  $h_s(x, y) = 1$ , jo označimo s  $k + 1$ .
3. korak: V naslednjem prehodu se ustavimo pri vseh tistih točkah, ki imajo oznako enako trenutnemu prehodu slike  $k$  in pregledamo vseh 8 sosedov te točke. Če katerakoli sosednja točka spet ustreza pogoju  $h_s(x, y) = 1$ , jo označimo s  $k = k + 1$ .
4. korak: V vseh nadaljnjih prehodih skozi vse točke slike  $D(x, y)$  postopamo na enak način kot v 3. koraku.
5. korak: Postopek ustavimo, ko ne najdemo nobene točke z vrednostjo tekočega prehoda, to je, ko nobena sosednja točka iz  $(k - 1)$  prehoda ne ustreza pogoju  $h_s(x, y) = 1$ .

*Konec postopka.*

Na koncu, vse neoznačene točke  $(x, y)$ , kakor tudi povezane verige točk, krajše od  $I_{min}$ , izločimo iz slike  $D(x, y)$ .

# Cannyjev postopek za ugotavljanje obrisov

- Canny je predlagal postopek za ugotavljanje obrisov, ki hkrati optimizira naslednje kriterije:
  - **moč ugotavljanja robov** (pomembni robovi ne smejo ostati neodkriti),
  - **pravilno lokalizacijo robov** (razdalja med dejanskim in odkritim robom mora biti kar se da majhna) in
  - **število odzivov roba** (večkratnih odzivov na en dejanski rob mora biti kar se da malo),

pri pogoju, da so na sliki robovi idealni in je slika motena le s seštevalno primešanim belim šumom.
- Cannyjev postopek odkriva robne točke objektov na sliki z odvajanjem slike, ki jo **predhodno zgladimo** z Gaussovim sitom.

# Glajenje slike z Gaussovim sitom (1/4)

- Označimo odziv na enotin impulz Gaussovega sita z  $G(x, y)$ . Zglajeno sliko  $g(x, y)$  dobimo z dvorazsežno konvolucijo:

$$g(x, y) = f(x, y) * G(x, y),$$

- Delna odvoda zglajene slike  $g(x, y)$  na  $x$  in na  $y$  sta enaka dvorazsežni konvoluciji slike  $f(x, y)$  in delnega odvoda funkcije  $G(x, y)$  na  $x$

$$\begin{aligned} g_x(x, y) &= \frac{\partial f(x, y) * G(x, y)}{\partial x} = f(x, y) * \frac{\partial G(x, y)}{\partial x} \\ &= f(x, y) * \left( -\frac{x}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \right) \\ &= \sum_{n=-W}^W \sum_{m=-W}^W w_{G_x}(n, m) f(x-n, y-m), \end{aligned}$$

$$\text{za: } x = W, \dots, N_x - (W + 1) \text{ in } y = W, \dots, N_y - (W + 1)$$

# Glajenje slike z Gaussovim sitom (2/4)

- oziroma na  $y$

$$\begin{aligned}g_y(x, y) &= \frac{\partial f(x, y) * G(x, y)}{\partial y} = f(x, y) * \frac{\partial G(x, y)}{\partial y} \\&= f(x, y) * \left( -\frac{y}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \right) \\&= \sum_{n=-W}^W \sum_{m=-W}^W w_{G_y}(n, m) f(x - n, y - m),\end{aligned}$$

za:  $x = W, \dots, N_x - (W + 1)$  in  $y = W, \dots, N_y - (W + 1)$

kjer sta  $w_{G_x}(n, m)$  in  $w_{G_y}(n, m)$  polji utežnih faktorjev normiranega okna razsežnost  $(2W + 1) \times (2W + 1)$

# Glajenje slike z Gaussovim sitom (3/4)

- Polji utežnih faktorjev  $w_{G_x}(n, m)$  in  $w_{G_y}(n, m)$  določimo v dveh korakih.
- Najprej določimo polji nenormiranih faktorjev

$$G_x(n, m) = n e^{-\frac{n^2+m^2}{2\sigma^2}} \quad \text{in} \quad G_y(n, m) = m e^{-\frac{n^2+m^2}{2\sigma^2}}$$

- Pri čemer vrednost parametra  $\sigma$  izberemo tako, da so vrednosti faktorjev na robu polja blizu 0, denimo 0,01

$$\sigma = \frac{W}{\sqrt{2[\ln(W) - \ln(0.01)]}}$$

- Za  $W = 2$  je tako  $\sigma = 0,614$

# Glajenje slike z Gaussovimi sitom (4/4)

- Pri  $W = 2$  in  $\sigma = 0,614$  sta polji nenormiranih faktorjev

$$G_x(n, m) = \begin{bmatrix} 0.000 & 0.001 & 0 & -0.001 & -0.000 \\ 0.003 & 0.071 & 0 & -0.071 & -0.003 \\ 0.010 & 0.266 & 0 & -0.266 & -0.010 \\ 0.003 & 0.071 & 0 & -0.071 & -0.003 \\ 0.000 & 0.001 & 0 & -0.001 & -0.000 \end{bmatrix}$$

$$G_y(n, m) = \begin{bmatrix} -0.000 & -0.003 & -0.010 & -0.003 & -0.000 \\ -0.001 & -0.071 & -0.266 & -0.071 & -0.001 \\ 0 & 0 & 0 & 0 & 0 \\ 0.001 & 0.071 & 0.266 & 0.071 & 0.001 \\ 0.000 & 0.003 & 0.010 & 0.003 & 0.000 \end{bmatrix}$$

- Nenormirani polji utežnih faktorjev nato normiramo tako, da delimo pozitivne utežne faktorje z vsoto pozitivnih utežnih faktorjev, negativne pa z absolutno vrednostjo vsote negativnih utežnih faktorjev

$$w_{G_x}(n, m) = \begin{bmatrix} 0 & 0.002 & 0 & -0.002 & 0 \\ 0.007 & 0.167 & 0 & -0.167 & -0.007 \\ 0.023 & 0.624 & 0 & -0.624 & -0.023 \\ 0.007 & 0.167 & 0 & -0.167 & -0.007 \\ 0 & 0.002 & 0 & -0.002 & 0 \end{bmatrix}$$

$$w_{G_y}(n, m) = \begin{bmatrix} 0 & -0.007 & -0.023 & -0.007 & 0 \\ -0.002 & -0.167 & -0.624 & -0.167 & -0.002 \\ 0 & 0 & 0 & 0 & 0 \\ 0.002 & 0.167 & 0.624 & 0.167 & 0.002 \\ 0 & 0.007 & 0.023 & 0.007 & 0 \end{bmatrix}$$



# Cannyjev postopek za ugotavljanje obrisov

- **Korak 1:** Za dano sliko  $f(x, y)$  izračunamo  $g_x(x, y)$  in  $g_y(x, y)$ .
- **Korak 2:** Za vsak slikovni element  $(x, y)$  izračunamo dolžino in smer gradienta

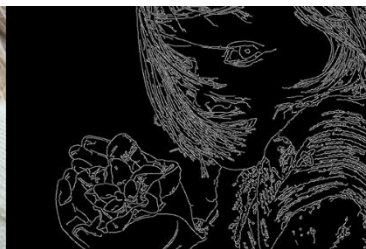
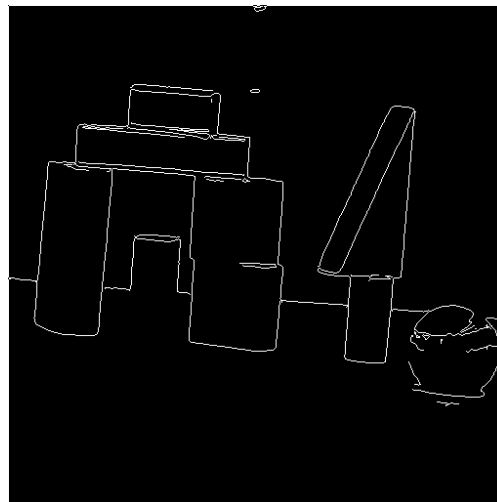
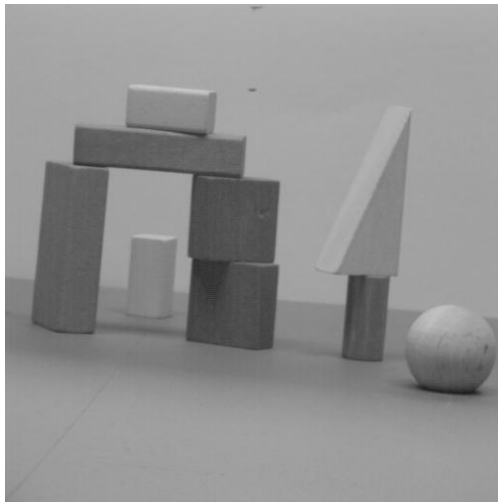
$$D(x, y) = \sqrt{g_x^2(x, y) + g_y^2(x, y)} \quad \theta(x, y) = \arctan\left(\frac{g_y(x, y)}{g_x(x, y)}\right)$$

- **Korak 3:** Zožimo izbor možnih robnih točk s postopkom *Odpravljanje nenajvečjih vrednosti* iz slike dolžin gradienta  $D(x, y)$ .
- **Korak 4:** Sliko možnih točk roba  $D'(x, y)$  obdelamo s postopkom *Histereznega upravljanja*.

Postopek navadno ponovimo za  $W = 1, 2$  in  $3$  in *izberemo najboljši rezultat*.



# Primer rezultatov Cannyjevega postopka



# Povezovanje robnih točk na osnovi modela obrisa

- S postopki, ki temeljijo na modelu obrisa, robne točke povežemo v obris tako, da ugotavljamo, **ali ležijo na krivulji** predpisane oblike, na primer na premici, krožnici, elipsi ipd.
- Ali ležijo točke na krivulji predpisane oblike ali v njeni bližini, lahko učinkovito ugotovimo s Houghovo transformacijo.
- Zasnovano Houghove transformacije bomo pojasnili na primeru povezovanja robnih točk v premice.

# Houghova transformacija (1/5)

- Skozi vsako točko  $(x_i, y_i)$  v slikovni ravnini  $xy$  lahko potegnemo neskončno premic, ki zadoščajo enačbi:

$$y_i = ax_i + b$$

- Če to enačbo zapišemo v obliki:

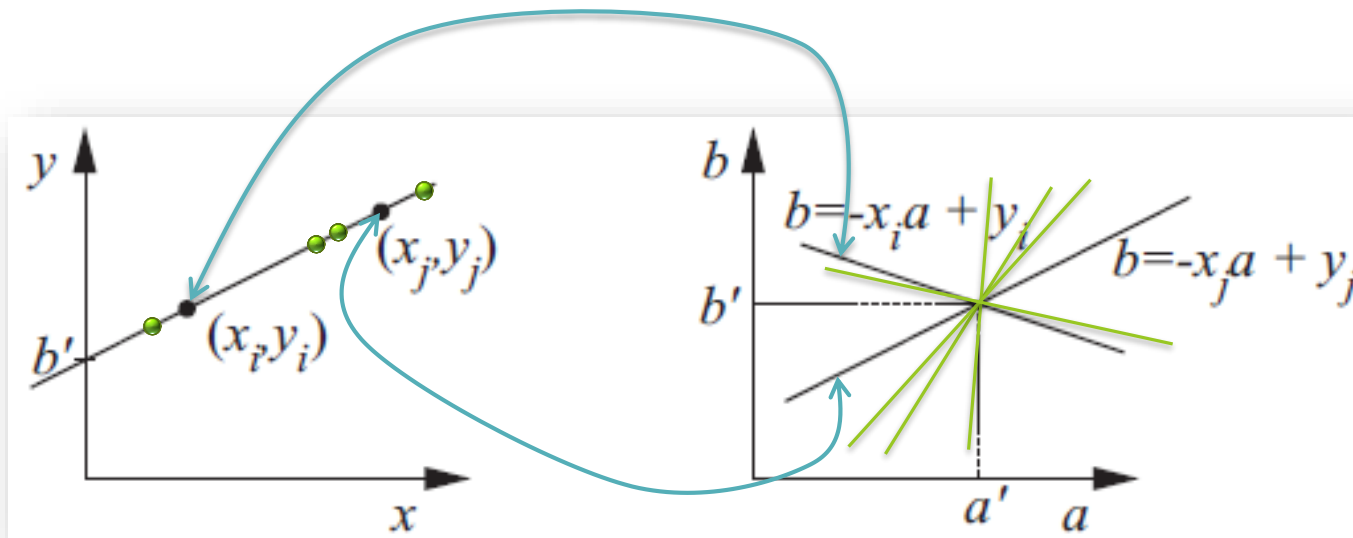
$$b = -x_i a + y_i$$

imamo v ravnini  $ab$ , ki jo imenujemo tudi *prostor parametrov krivulje*, le eno samo premico, ki ustreza vsem premicam, ki jih lahko potegnemo skozi točko  $(x_i, y_i)$ .

- Enako velja za neko drugo točko  $(x_j, y_j)$ , ki ji v prostoru parametrov ustreza premica  $b = -x_j a + y_j$

# Houghova transformacija (2/5)

- Presečišče obeh premic v točki  $(a', b')$  v prostoru parametrov  $ab$  potem določa premico v slikovnem prostoru, ki gre v ravnini  $xy$  skozi obe točki -  $(x_i, y_i)$  in  $(x_j, y_j)$ .
- Še več, vsem točkam, ki ležijo na tej premici, ustrezajo v parametričnem prostoru premice, ki se vse sekajo v točki  $(a', b')$ .



# Houghova transformacija (3/5)

- Vzemimo, da želimo v slikovni ravnini  $xy$  poiskati vse tiste robne točke, ki ležijo na isti premici.
- To dosežemo z iskanjem presečišč med premicami v prostoru parametrov  $ab$ , kjer vsaka premica ustreza eni robni točki v slikovnem prostoru  $xy$ .
- Učinkovitost Houghove transformacije izvira iz razdelitve parametričnega prostora na zbiralne celice, vsaki kjer točki parametričnega prostora  $(a_p, b_q)$  ustreza zbiralna celica  $A_{pq}$ .

# Houghova transformacija (4/5)

- V začetku procesa povezovanja robnih točk v obris objekta, ki ga lahko opišemo s premicami, postavimo vse zbiralne celice na vrednost nič in za vsako robno točko  $(x_k, y_k)$  v slikovni ravnini  $xy$  rešimo enačbo

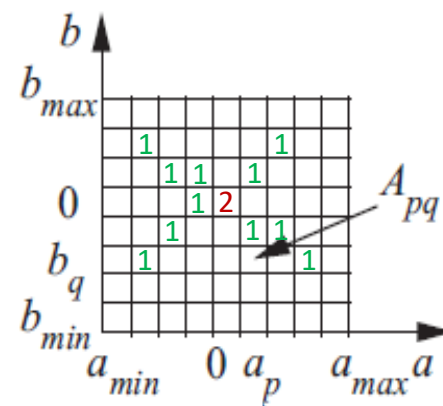
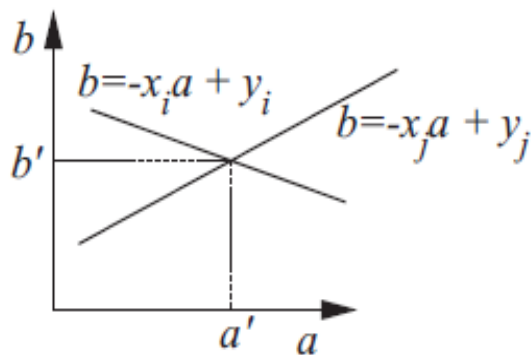
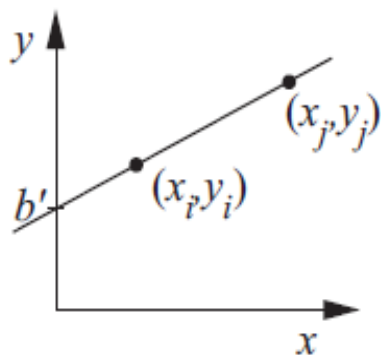
$$b = -x_k a + y_k$$

tako, da parameter  $a$  lahko zavzame vse vrednosti med  $a_{min}$  in  $a_{max}$ , ki jih dopušča razdelitev v parametričnem prostoru.

- Če smo za nek  $a_p$  izračunali  $b_q$ , povečamo vrednost zbiralne celice  $A_{pq}$  za ena.
- Ob koncu postopka, ko smo na opisani način obdelali vseh  $n$  robnih točk, pomeni vrednost  $m$  v zbiralni celici  $A_{pq}$ , da v slikovni ravnini izmed  $n$  robnih točk leži na premici  $y = a_p x + b_q$   $m$  točk.

# Houghova transformacija (5/5)

- Ponazoritev iskanja robnih točk, ki ležijo na premicah



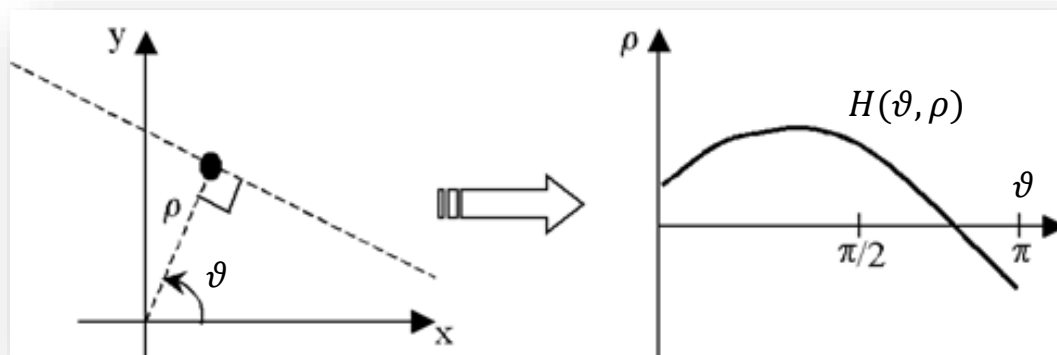
# Zapis premice v normalni obliki

- Zapis premice v obliki  $y = ax + b$  ni primeren za premice, ki so vzporedne ali skoraj vzporedne z osjo  $y$ , ker se tedaj  $a$  in  $b$  približujeta neskončnosti.
- Zato premico, ki gre skozi točko  $(x_i, y_i)$ , raje zapišemo v normalni obliki s parametroma  $\rho$  in  $\vartheta$  namesto  $a$  in  $b$ :

$$y_i = -\frac{\cos \vartheta}{\sin \vartheta} x_i + \frac{\rho}{\sin \vartheta}$$

ki jo v prostoru parametrov zapišemo:

$$\rho = x_i \cos \vartheta + y_i \sin \vartheta$$





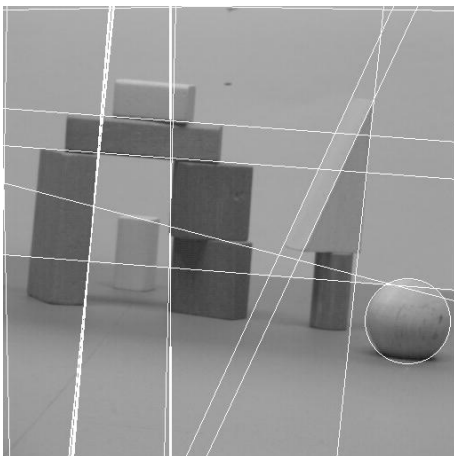
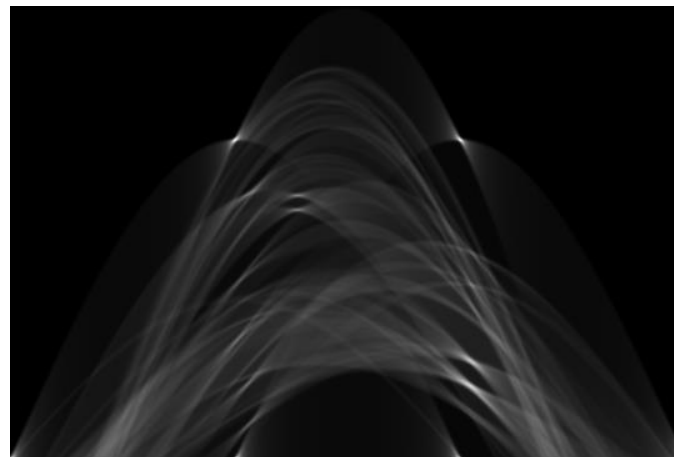
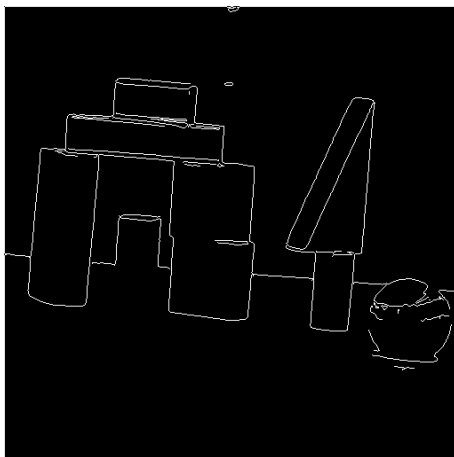
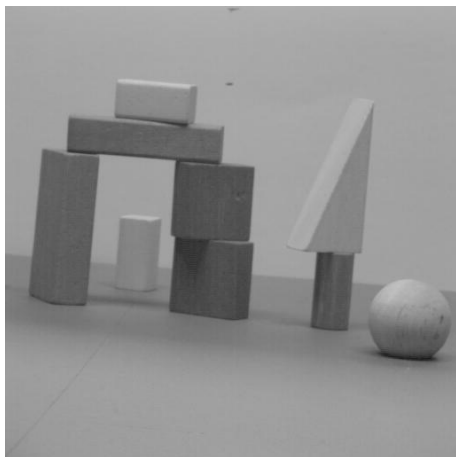
# Obravnava krivulj, ki niso premice

- Povezovanje robnih točk v obris s Houghovo transformacijo lahko opravimo z vsako krivuljo, ki jo splošno zapišemo v obliki  $g(\mathbf{x}, \mathbf{c}) = 0$ , kjer je  $\mathbf{x}$  vektor koordinat točke in  $\mathbf{c}$  vektor parametrov krivulje.
- Na primer, robne točke, ki ležijo na krožnici s polmerom  $R$  in s središčem v točki  $(u, v)$ :

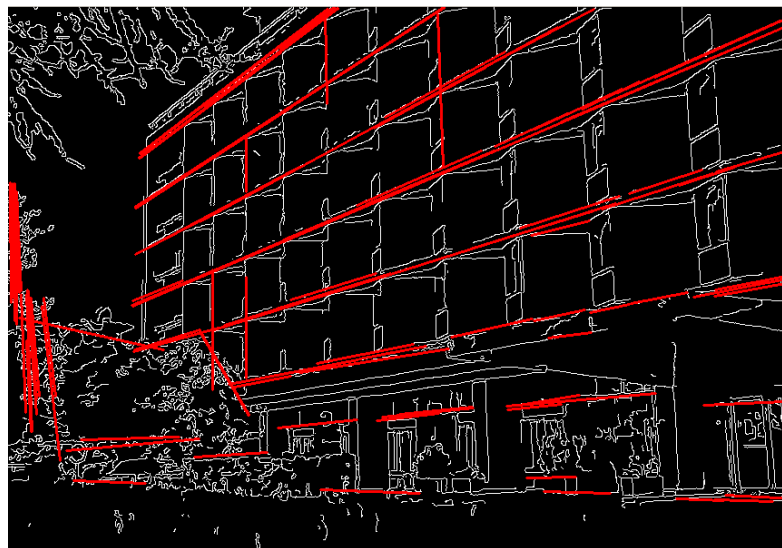
$$R = \sqrt{(x - u)^2 + (y - v)^2},$$

odkrivamo s Houghovo transformacijo ravnine  $xy$  v trirazsežni prostor parametrov:  $u$ ,  $v$  in  $R$  z zbiralniki  $A_{pqr}$ .

# Zgled iskanja obrisov s Houghovo transformacijo



# Zgledi iskanja obrisov s Houghovo transformacijo



# Zaključek

- Najprej smo se spoznali s postopki razčlenjevanja govornih signalov.
- Govorne signale razčlenjujemo predvsem na povedi ter na kratko-časovne izseke z uporabo okenskih funkcij.
- Pri razčlenjevanju slik smo spoznali postopke, ki temeljijo na ugotavljanjem področij, ter postopki, ki temeljijo na ugotavljanju obrisov.
- Spoznali smo se tudi z Haughovo transformacijo, ki omogoča povezovanje robnih točk objektov na slikah na osnovi modela obrisa.