

Robotika in računalniško zaznavanje (RRZ)

Ujemanje slik

Danijel Skočaj

Univerza v Ljubljani

Fakulteta za računalništvo in informatiko

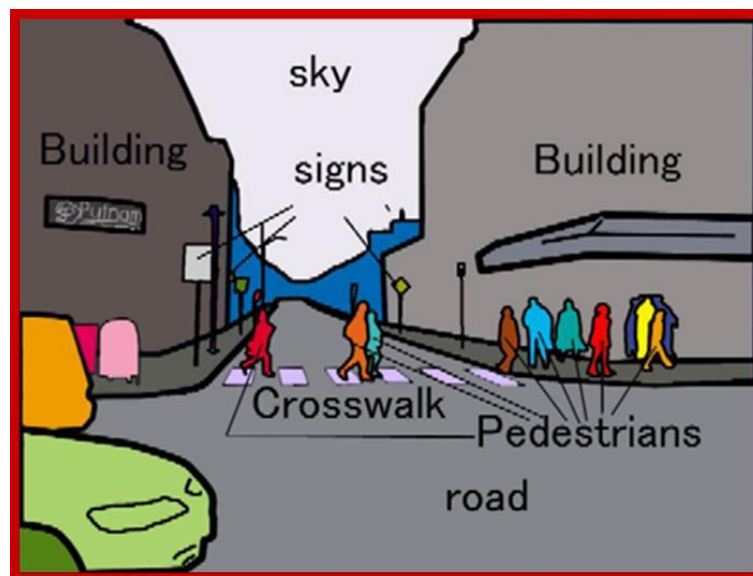
Literatura: W. Burger, M. J. Burge (2008).

Digital Image Processing, poglavje 17

v1.0

Interpretacija slik

- Ultimativni cilj računalniškega (spoznavnega) vida
- Razumeti kaj se na sliki nahaja
 - Zaznati
 - Spoznati



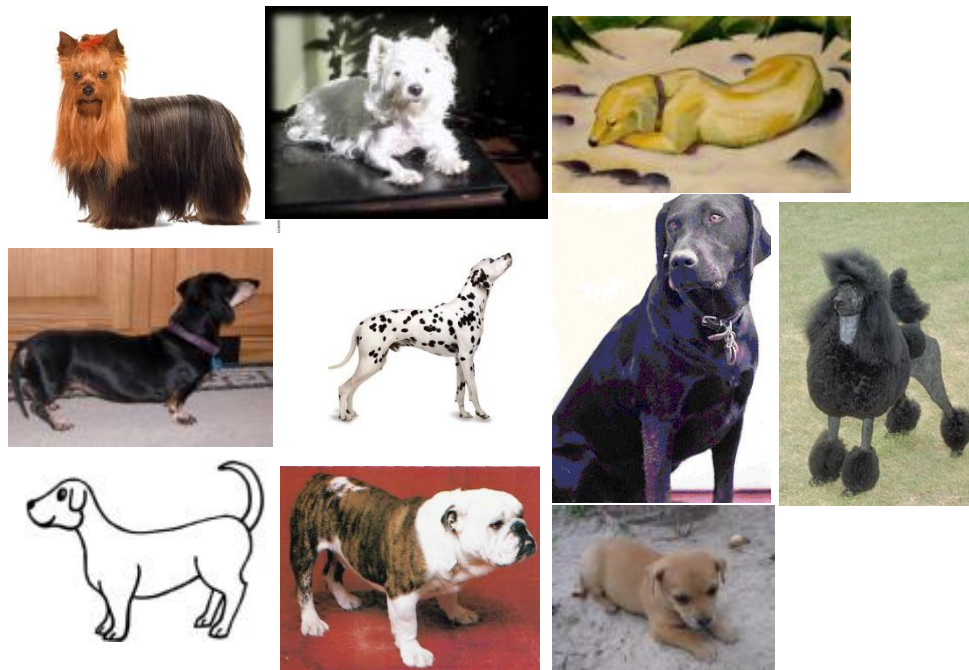
Razpoznavanje objektov

- Različni položaji na sliki
- Različne orientacije
- Različne velikosti
- Različne osvetlitve
- Prekrivanja
- Deformacije



Kategorizacija objekto

- Velika vizualna variabilnost znotraj kategorije



Interpretacija slik

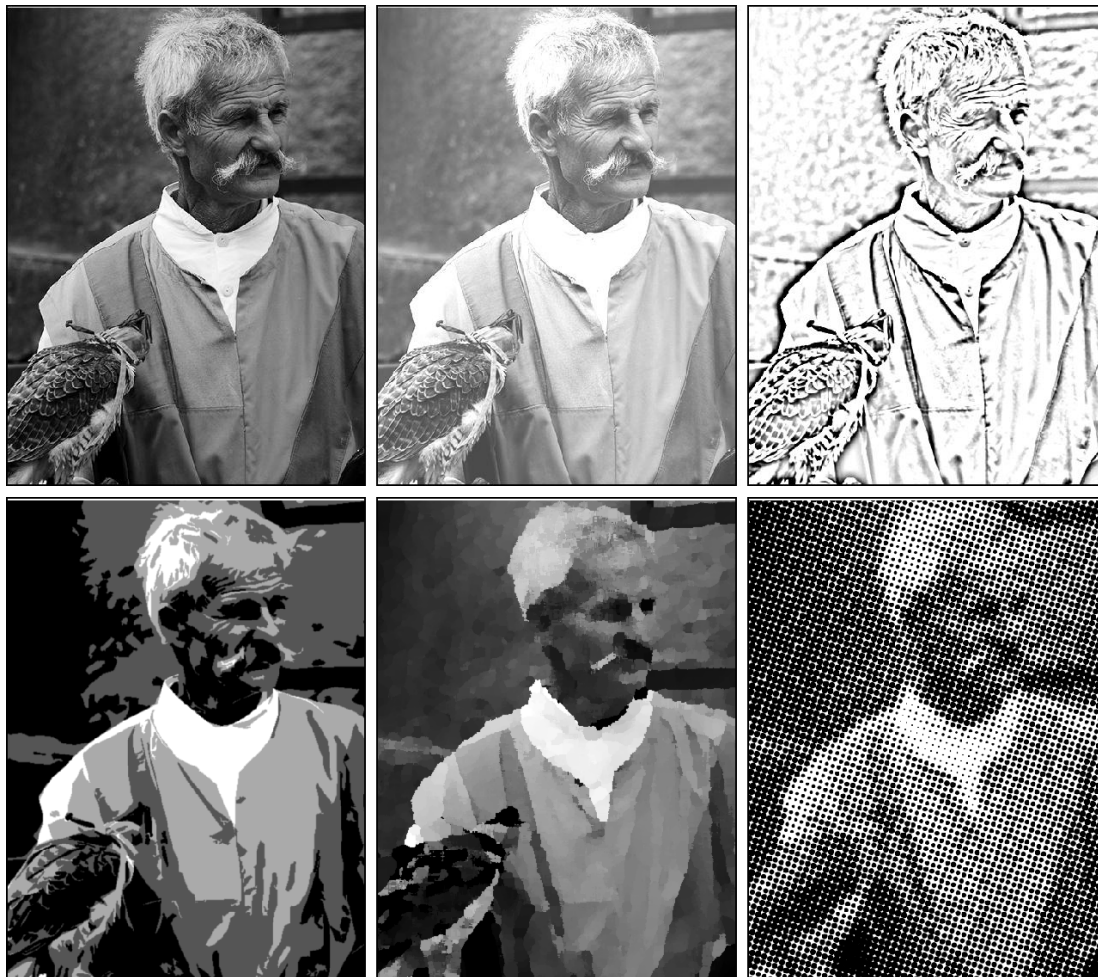
- Koliko stolov je na sliki?



MPIK Tübingen

Ujemanje slik

- Ali sta sliki (ali del slike) „enaki“?

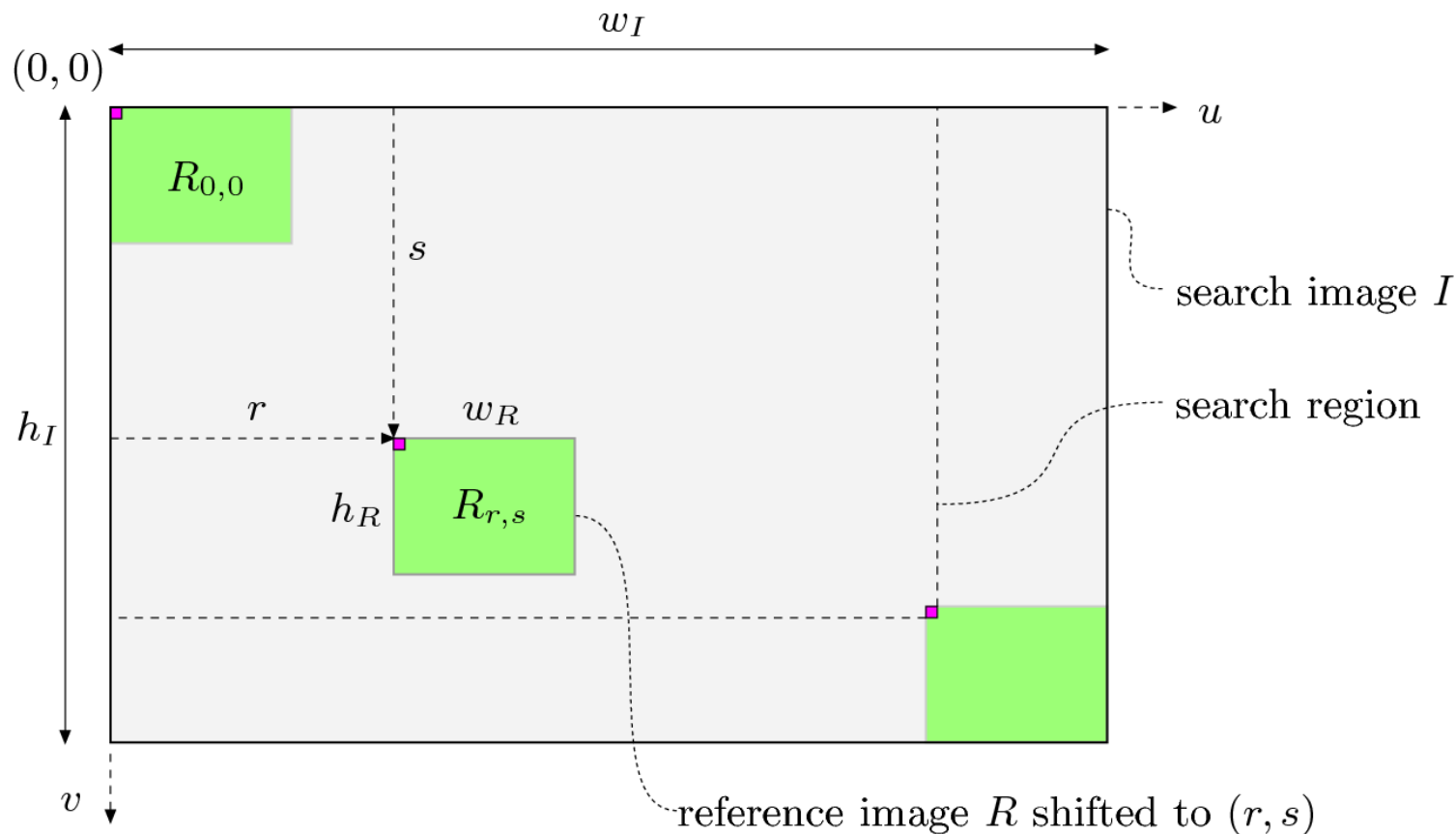


Ujemanje slik

- Precej bolj enostaven problem
- Cilj ni semantična interpretacija slike
- Primerjamo slike na nivoju slikovnih elementov
- Iščemo dele slik na večji sliki
- Zelo uporabno
 - Za iskanje korespondenčnih točk na stereo slikah
 - Iskanje določenih vzorcev na sliki
 - Za sledenje predmetov
 - Za registracijo slik
 - Za razpoznavanje značilnih delov slik
- Kakšno metriko uporabiti
 - Invariance?

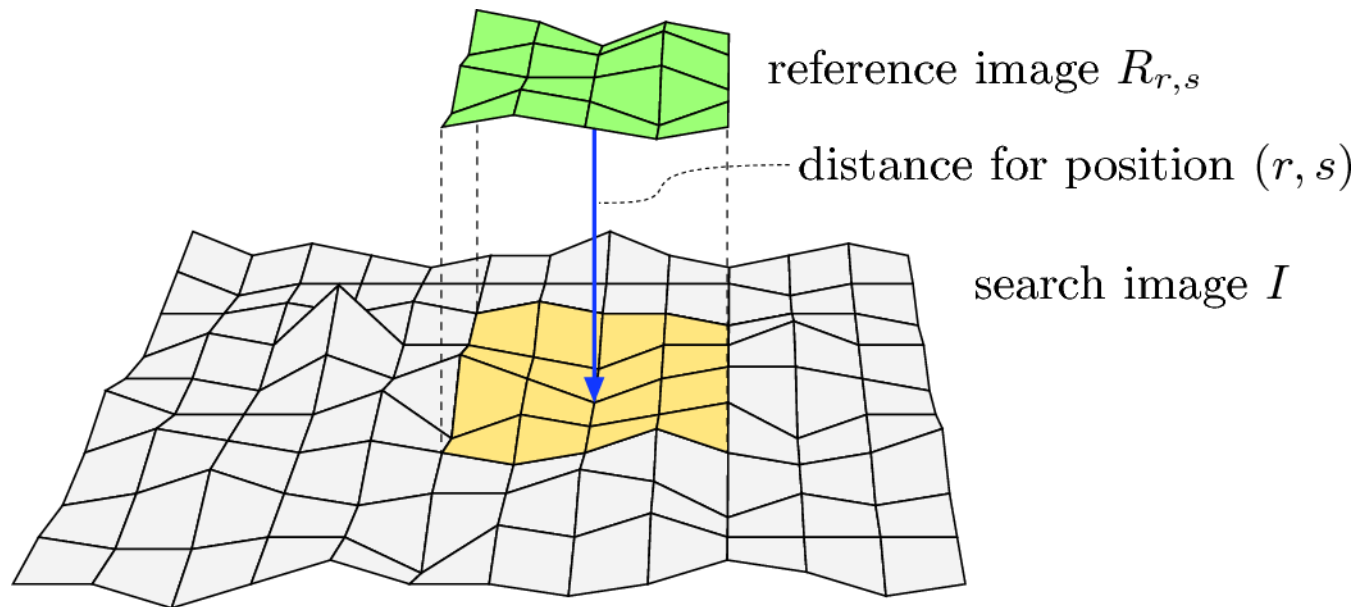
Ujemanje vzorcev v intenzitetnih slikah

- Lokalizacija referenčne slike (vzorca) R v večji intenzitetni (sivinski) sliki I
 - Poišči lokacije (r,s) na sliki I , kjer sta R in ustrezna podslika na sliki I najbolj podobni



Razlika med slikovnimi vzorci

- Referenčno sliko pomikamo po glavni sliki
 - V vsaki točki izračunamo razliko



Mere podobnosti

- Različne mere podobnosti se lahko uporabljajo:

- Vsota absolutnih razlik:

$$d_A(r, s) = \sum_{(i,j) \in R} |I(r+i, s+j) - R(i, j)|$$

- Maksimalna razlika:

$$d_M(r, s) = \max_{(i,j) \in R} |I(r+i, s+j) - R(i, j)|$$

- Vsota kvadratov razlik:

- Evklidska razdalja

$$d_E(r, s) = \left[\sum_{(i,j) \in R} (I(r+i, s+j) - R(i, j))^2 \right]^{1/2}$$

Razdalja in korelacija

- Kvadrat evklidske razdalje:

$$\begin{aligned} d_E^2(r, s) &= \sum_{(i,j) \in R} (I(r+i, s+j) - R(i, j))^2 \\ &= \underbrace{\sum_{(i,j) \in R} I^2(r+i, s+j)}_{A(r, s)} + \underbrace{\sum_{(i,j) \in R} R^2(i, j)}_B - 2 \underbrace{\sum_{(i,j) \in R} I(r+i, s+j) \cdot R(i, j)}_{C(r, s)} \end{aligned}$$

- Linearna prečna korelacija

- Zanemarimo vpliv lokalne intenzitete slike

$$(I \circledast R)(r, s) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} I(r+i, s+j) \cdot R(i, j)$$

$$\sum_{i=0}^{w_R-1} \sum_{j=0}^{h_R-1} I(r+i, s+j) \cdot R(i, j) = \sum_{(i,j) \in R} I(r+i, s+j) \cdot R(i, j)$$

Normalizirana prečna korelacija

- Normaliziramo z vrednostjo lokalne intenzitete slike
 - Postanemo invariantni na lokalne spremembe v intenziteti
 - Lokalna mera

$$\begin{aligned} C_N(r, s) &= \frac{C(r, s)}{\sqrt{A(r, s) \cdot B}} = \frac{C(r, s)}{\sqrt{A(r, s)} \cdot \sqrt{B}} \\ &= \frac{\sum_{(i,j) \in R} I(r+i, s+j) \cdot R(i, j)}{\left[\sum_{(i,j) \in R} I^2(r+i, s+j) \right]^{1/2} \cdot \left[\sum_{(i,j) \in R} R^2(i, j) \right]^{1/2}} \end{aligned}$$

- Vrednost
 - 0 – ni ujemanja
 - 1 – popolno ujemanje
- Ni invariantna na spremembo intenzitete celotne slike

Korelacijski koeficient

- Ne primerjamo originalne vrednosti, ampak razlike do povprečnih vrednosti vzorca in podslike:

$$C_L(r, s) = \frac{\sum_{(i,j) \in R} (I(r+i, s+j) - \bar{I}(r, s)) \cdot (R(i, j) - \bar{R})}{\left[\sum_{(i,j) \in R} (I(r+i, s+j) - \bar{I}_{r,s})^2 \right]^{1/2} \cdot \underbrace{\left[\sum_{(i,j) \in R} (R(i, j) - \bar{R})^2 \right]^{1/2}}_{S_R^2 = K \cdot \sigma_R^2}}$$
$$K = |R|$$

$$\bar{I}_{r,s} = \frac{1}{K} \cdot \sum_{(i,j) \in R} I(r+i, s+j) \quad \text{and} \quad \bar{R} = \frac{1}{K} \cdot \sum_{(i,j) \in R} R(i, j)$$

- Vrednost
 - -1 – ni ujemanja
 - 1 – popolno ujemanje
- Je invariantna na spremembo intenzitete celotne slike

Korelacijski koeficient

- Hitrejše računanje korelacijskega koeficienta
 - \bar{R} in S_R izračunamo samo enkrat za celo sliko

$$S_R^2 = K \cdot \sigma_R^2 = \sum_{(i,j) \in R} (R(i,j) - \bar{R})^2$$

$$\sigma_R = \frac{1}{K} \sum R^2(i,j) - \bar{R}^2$$

$$S_R^2 = \sum_{(i,j) \in R} R^2(i,j) - K \cdot \bar{R}^2$$

$$= \sum_{(i,j) \in R} R^2(i,j) - \frac{1}{K} \cdot \left(\sum_{(i,j) \in R} R(i,j) \right)^2$$

$$C_L(r,s) = \frac{\sum_{(i,j) \in R} (I(r+i, s+j) \cdot R(i,j)) - K \cdot \bar{I}_{r,s} \cdot \bar{R}}{\left[\sum_{(i,j) \in R} I^2(r+i, s+j) - K \cdot \bar{I}_{r,s}^2 \right]^{1/2} \cdot S_R}$$

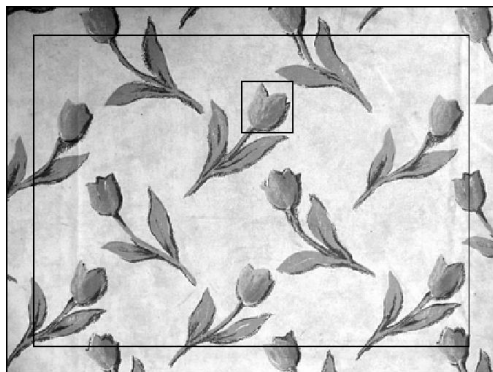
Algoritem

```
1: CORRELATIONCOEFFICIENT ( $I, R$ )
    $I(u, v)$ : search image of size  $w_I \times h_I$ 
    $R(i, j)$ : reference image of size  $w_R \times h_R$ 
   Returns  $C(r, s)$  containing the values of the correlation coefficient
   between  $I$  and  $R$  positioned at  $(r, s)$ .

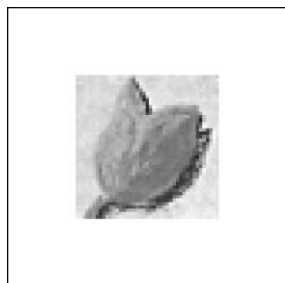
   STEP 1—INITIALIZE:
2:    $K \leftarrow w_R \cdot h_R$ 
3:    $\Sigma_R \leftarrow 0, \Sigma_{R2} \leftarrow 0$ 
4:   for  $i \leftarrow 0 \dots (w_R - 1)$  do
5:     for  $j \leftarrow 0 \dots (h_R - 1)$  do
6:        $\Sigma_R \leftarrow \Sigma_R + R(i, j)$ 
7:        $\Sigma_{R2} \leftarrow \Sigma_{R2} + (R(i, j))^2$ 
8:    $\bar{R} \leftarrow \Sigma_R / K$  ▷ Eqn. (17.8)
9:    $S_R \leftarrow \sqrt{\Sigma_{R2} - K \cdot \bar{R}^2} = \sqrt{\Sigma_{R2} - \Sigma_R^2 / K}$  ▷ Eqn. (17.10)

   STEP 2—COMPUTE THE CORRELATION MAP:
10:   $C \leftarrow$  new map of size  $(w_I - w_R + 1) \times (h_I - h_R + 1)$ ,  $C(r, s) \in \mathbb{R}$ 
11:  for  $r \leftarrow 0 \dots (w_I - w_R)$  do ▷ place  $R$  at position  $(r, s)$ 
12:    for  $s \leftarrow 0 \dots (h_I - h_R)$  do
      Compute correlation coefficient for position  $(r, s)$ :
13:       $\Sigma_I \leftarrow 0, \Sigma_{I2} \leftarrow 0, \Sigma_{IR} \leftarrow 0$ 
14:      for  $i \leftarrow 0 \dots (w_R - 1)$  do
15:        for  $j \leftarrow 0 \dots (h_R - 1)$  do
16:           $a_I \leftarrow I(r + i, s + j)$ 
17:           $a_R \leftarrow R(i, j)$ 
18:           $\Sigma_I \leftarrow \Sigma_I + a_I$ 
19:           $\Sigma_{I2} \leftarrow \Sigma_{I2} + a_I^2$ 
20:           $\Sigma_{IR} \leftarrow \Sigma_{IR} + a_I \cdot a_R$ 
21:       $\bar{I}_{r,s} \leftarrow \Sigma_I / K$  ▷ Eqn. (17.8)
22:       $C(r, s) \leftarrow \frac{\Sigma_{IR} - K \cdot \bar{I}_{r,s} \cdot \bar{R}}{\sqrt{\Sigma_{I2} - K \cdot \bar{I}_{r,s}^2} \cdot S_R} = \frac{\Sigma_{IR} - \Sigma_I \cdot \bar{R}}{\sqrt{\Sigma_{I2} - \Sigma_I^2 / K} \cdot S_R}$ 
23:  return  $C$ . ▷  $C(r, s) \in [-1, 1]$ 
```

Primer



(a) original image I



(b) reference image R



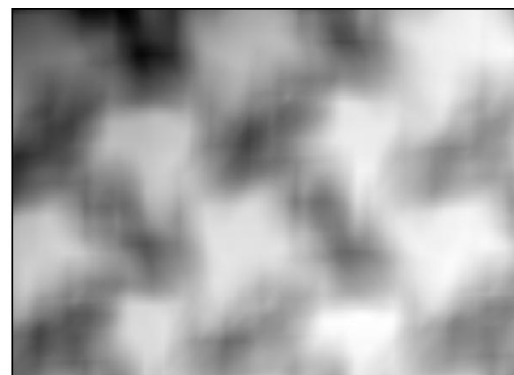
(c) sum of absolute differences



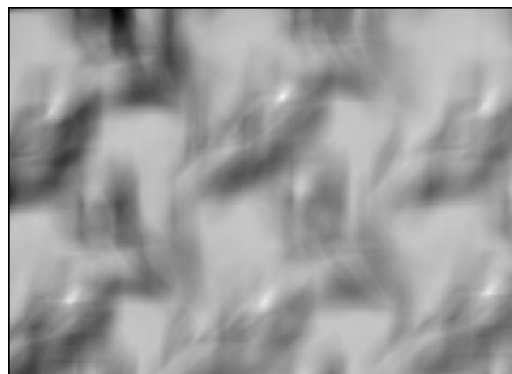
(d) maximum difference



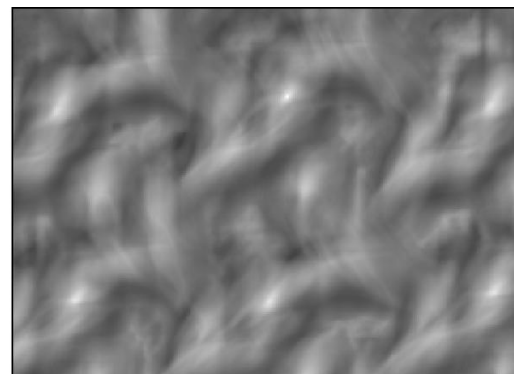
(e) sum of squared distances



(f) global cross correlation



(g) normalized cross correlation

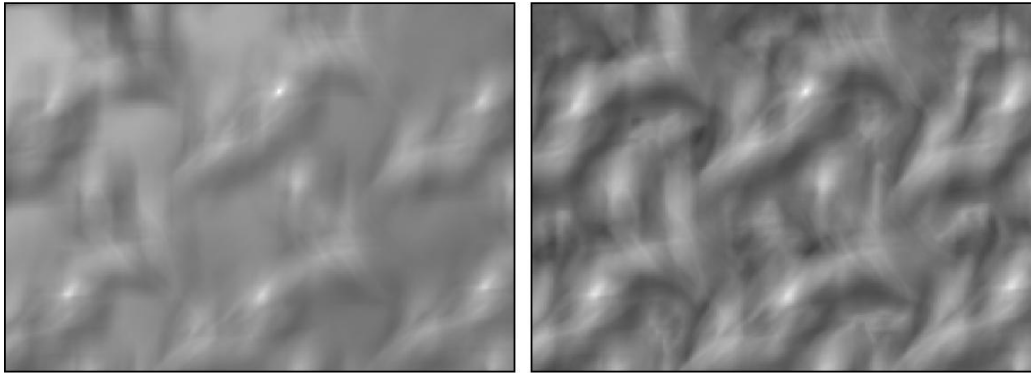


(h) correlation coefficient

Primer

- Spremenjena globalna intenziteta referenčnega vzorca

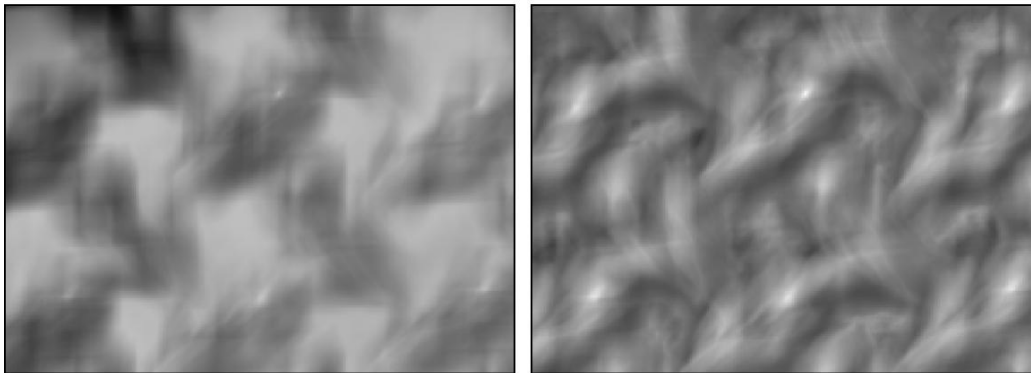
Original reference image: R



(a) Euclidean distance

(b) correlation coefficient

Modified reference image: $R' = R + 50$

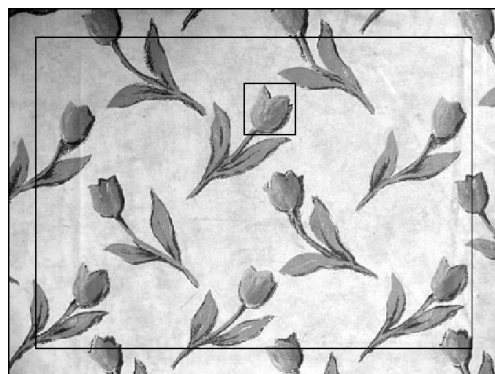


(c) Euclidean distance

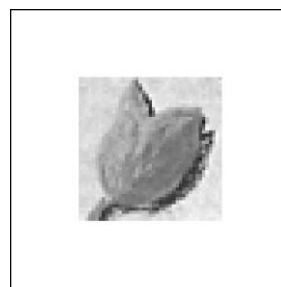
(d) correlation coefficient

Primer

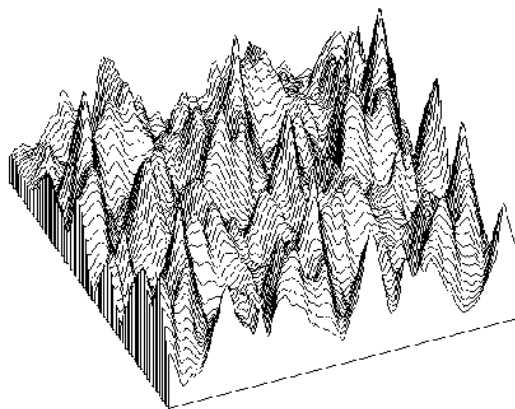
- Iskanje maksimalnih vrednosti korelacijskega koeficienta z uporagovljenjem



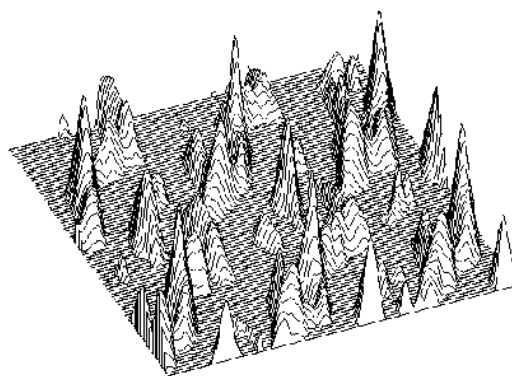
(a) original image I



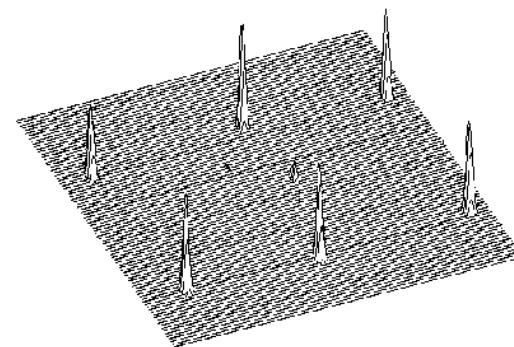
(b) reference image R



(a)



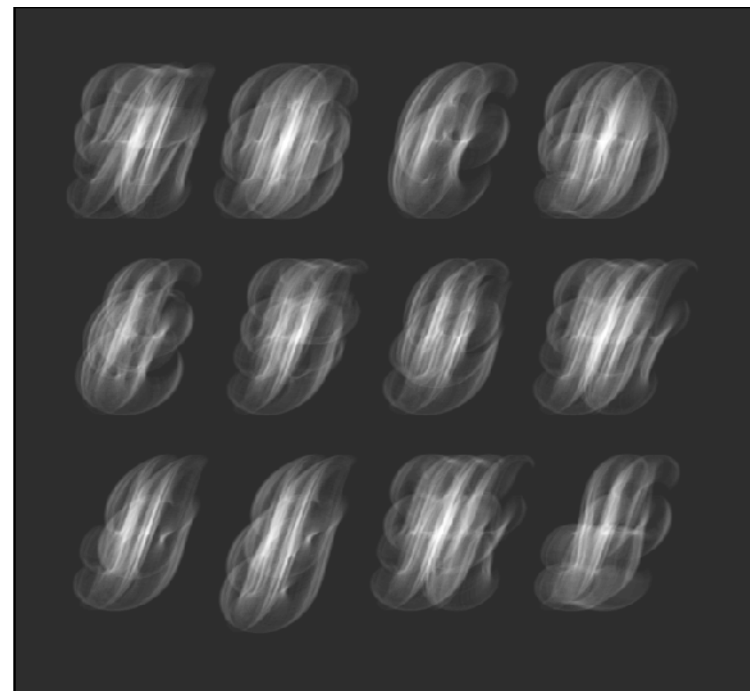
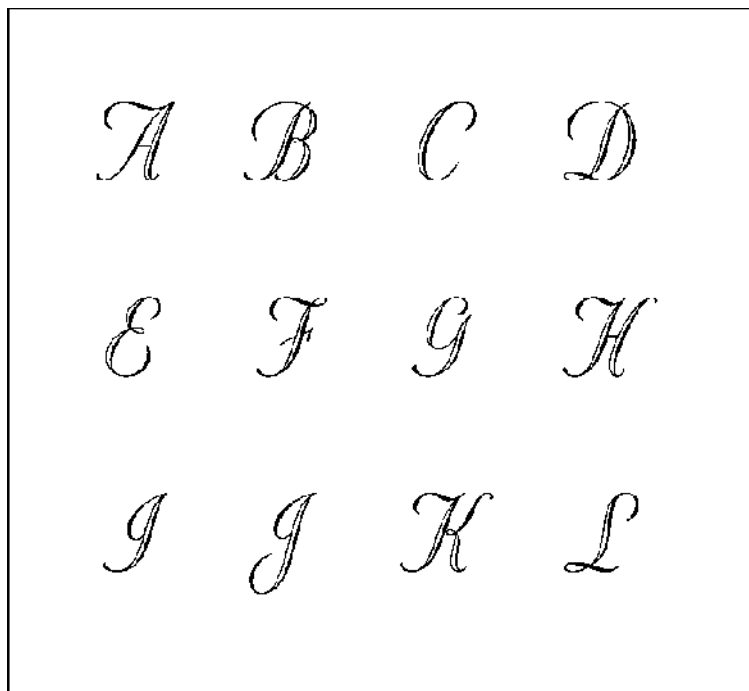
(b)



(c)

Ujemanje binarnih slik

- Neposredno primerjanje binarnih slik ni primerno
 - Imamo samo dve možni vrednosti sl. elementov
 - Že najmanjši premik ima zelo velik vpliv
 - Zelo nezvezno spreminjanje funkcije razlike z veliko lokalnimi maksimumi



Transformacija razdalje

- Distance transform
- Vsak slikovni element ozadja preslika v razdaljo do najbližjega slikovnega elementa ospredja

$$FG(I) = \{\mathbf{p} \mid I(\mathbf{p}) = 1\}$$

$$BG(I) = \{\mathbf{p} \mid I(\mathbf{p}) = 0\}$$

$$D(\mathbf{p}) = \min_{\mathbf{p}' \in FG(I)} \text{dist}(\mathbf{p}, \mathbf{p}')$$

- Uporabimo lahko različne razdalje:
 - Evklidska razdalja:

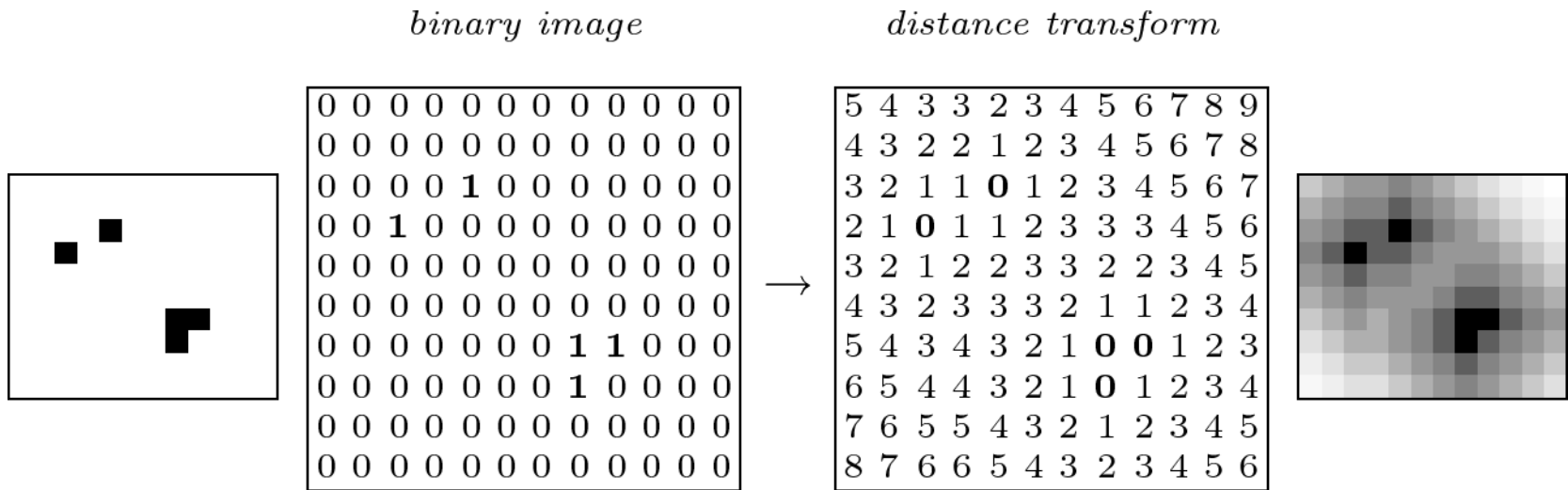
$$d_E(\mathbf{p}, \mathbf{p}') = \|\mathbf{p} - \mathbf{p}'\| = \sqrt{(u - u')^2 + (v - v')^2} \in \mathbb{R}^+$$

- Manhattanska razdalja:

$$d_M(\mathbf{p}, \mathbf{p}') = |u - u'| + |v - v'| \in \mathbb{N}_0$$

Primer

- Pretvorba binarne slike v transformacijo razdalje
- Manhattanska razdalja



- Neposredna implementacija tega algoritma je zelo časovno zahtevna
 - Za vsak slikovni element ozadja moramo poiskati najbližji slikovni element ospredja

Chamferjev algoritem

- Učinkovit algoritem za računanje transformacije razdalje
- Slikovne elemente slike obiskuje sekvenčno v dveh korakih:
 - Prvič od zgornjega levega kota proti spodnem desnemu
 - Drugič od spodnjega desnega kota do zgornjega levega
 - Vrednosti razdalj se propagirajo z uporabo mask razdalj
 - Opisujejo geometrično radaljo med trenutnim slikovnim elementom (x) in ostalimi

$$M^L = \begin{bmatrix} m_2^L & m_3^L & m_4^L \\ m_1^L & \times & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix}$$

$$M^R = \begin{bmatrix} \cdot & \cdot & \cdot \\ \cdot & \times & m_1^R \\ m_4^R & m_3^R & m_2^R \end{bmatrix}$$

Maske razdalj

- Manhattanska razdalja

- Vrne natančno manhattansko razdaljo

$$M_M^L = \begin{bmatrix} 2 & 1 & 2 \\ 1 & \times & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix}$$

$$M_M^R = \begin{bmatrix} \cdot & \cdot & \cdot \\ \cdot & \times & 1 \\ 2 & 1 & 2 \end{bmatrix}$$

- Evklidska razdalja

- Računa približek evklidske razdalje

$$M_E^L = \begin{bmatrix} \sqrt{2} & 1 & \sqrt{2} \\ 1 & \times & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix}$$

$$M_E^R = \begin{bmatrix} \cdot & \cdot & \cdot \\ \cdot & \times & 1 \\ \sqrt{2} & 1 & \sqrt{2} \end{bmatrix}$$

- Celoštevilske maske

- Približek evklidske razdalje

$$M_{E'}^L = \begin{bmatrix} 4 & 3 & 4 \\ 3 & \times & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix}$$

$$M_{E'}^R = \begin{bmatrix} \cdot & \cdot & \cdot \\ \cdot & \times & 3 \\ 4 & 3 & 4 \end{bmatrix}$$

Algoritem

```
1: DISTANCETRANSFORM ( $I$ )
    $I$ : binary image of size  $M \times N$ .
   Returns the distance transform of image  $I$ .

   STEP 1—INITIALIZE:
2:  $D \leftarrow$  new distance map of size  $M \times N$ ,  $D(u, v) \in \mathbb{R}$ 
3: for all image coordinates  $(u, v)$  do
4:   if  $I(u, v) = 1$  then
5:      $D(u, v) \leftarrow 0$  ▷ foreground pixel (zero distance)
6:   else
7:      $D(u, v) \leftarrow \infty$  ▷ background pixel (infinite distance)

   STEP 2—L→R PASS (using distance mask  $M^L = m_i^L$ ):
8: for  $v \leftarrow 1, 2, \dots, N-1$  do ▷ top → bottom
9:   for  $u \leftarrow 1, 2, \dots, M-2$  do ▷ left → right
10:    if  $D(u, v) > 0$  then
11:       $d_1 \leftarrow m_1^L + D(u-1, v)$ 
12:       $d_2 \leftarrow m_2^L + D(u-1, v-1)$ 
13:       $d_3 \leftarrow m_3^L + D(u, v-1)$ 
14:       $d_4 \leftarrow m_4^L + D(u+1, v-1)$ 
15:       $D(u, v) \leftarrow \min(d_1, d_2, d_3, d_4)$ 

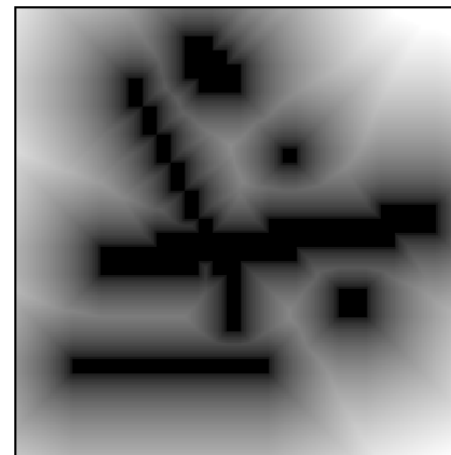
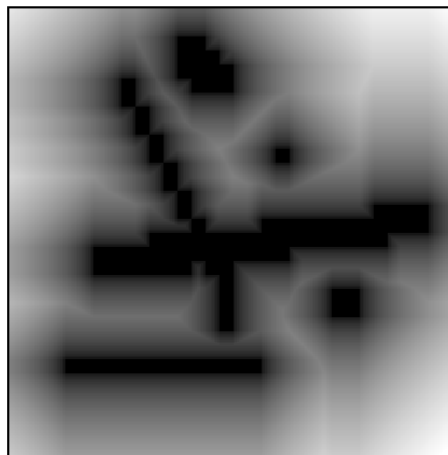
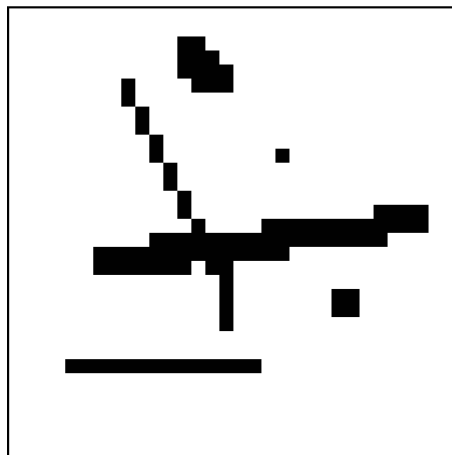
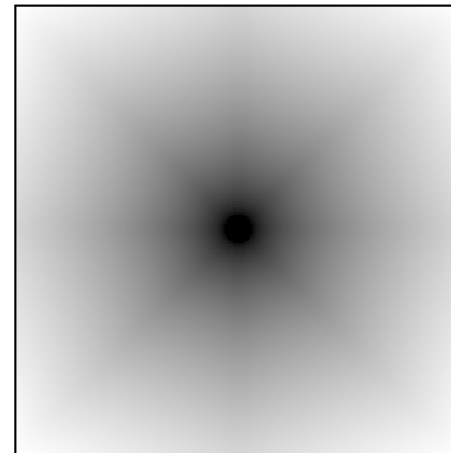
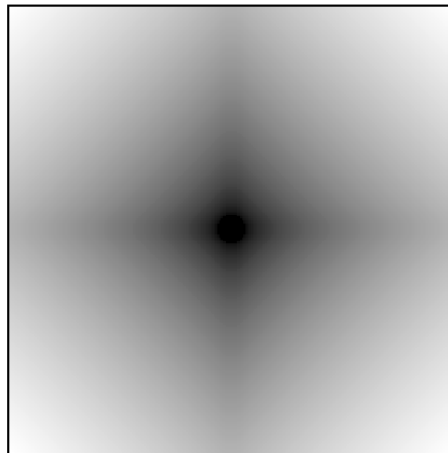
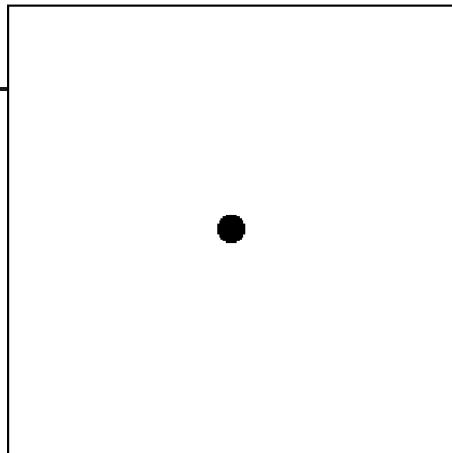
   STEP 3—R→L PASS (using distance mask  $M^R = m_i^R$ ):
16: for  $v \leftarrow N-2, \dots, 1, 0$  do ▷ bottom → top
17:   for  $u \leftarrow M-2, \dots, 2, 1$  do ▷ right → left
18:    if  $D(u, v) > 0$  then
19:       $d_1 \leftarrow m_1^R + D(u+1, v)$ 
20:       $d_2 \leftarrow m_2^R + D(u+1, v+1)$ 
21:       $d_3 \leftarrow m_3^R + D(u, v+1)$ 
22:       $d_4 \leftarrow m_4^R + D(u-1, v+1)$ 
23:       $D(u, v) \leftarrow \min(D(u, v), d_1, d_2, d_3, d_4)$ 
24: return  $D$ .
```

Primer

Original

Manhattan distance

Euclid. dist. (approx.)



Chamferjevo ujemanje

- Uporabimo transformacijo razdalje za oceno podobnosti binarnih slik
- Seštevajo se vse vrednosti s slike transformacije razdalje, ki so istoležne s slikovnimi elementi ospredja na binarni vzorčni sliki

$$Q(r, s) = \frac{1}{K} \cdot \sum_{(i,j) \in FG(R)} D(r+i, s+j) \quad K = |FG(R)|$$

- Najboljša lokacija je tista, kjer je vsota razdalj (Q) najmanjša

$$\mathbf{p}_{\text{opt}} = (r_{\text{opt}}, s_{\text{opt}}) = \underset{(r,s)}{\operatorname{argmin}} Q(r, s)$$

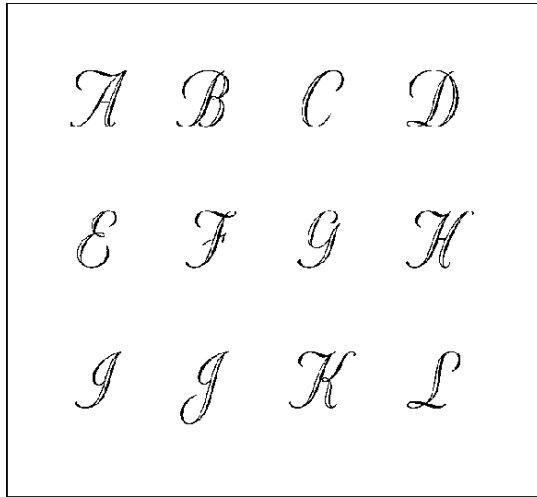
- Namesto linearne vsote lahko uporabimo tudi koren povprečja kvadratov:

$$Q_{rms}(r, s) = \sqrt{\frac{1}{K} \cdot \sum_{(i,j) \in FG(R)} D^2(r+i, s+j)}$$

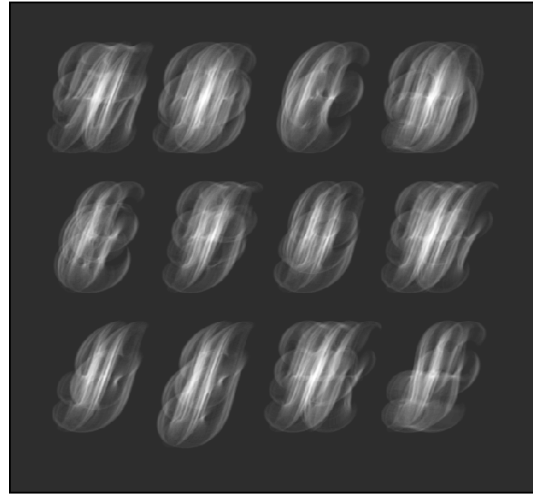
Algoritem

```
1: CHAMFERMATCH ( $I, R$ )  
     $I$ : binary search image of size  $w_I \times h_I$   
     $R$ : binary reference image of size  $w_R \times h_R$   
    Returns a two-dimensional map of match scores.  
  
    STEP 1—INITIALIZE:  
2:    $D \leftarrow \text{DISTANCETRANSFORM}(I)$  ▷ see Alg. 17.2  
3:    $K \leftarrow$  number of foreground pixels in  $R$   
4:    $Q \leftarrow$  new match map of size  $(w_I - w_R + 1) \times (h_I - h_R + 1)$ ,  $Q(r, s) \in \mathbb{R}$   
  
    STEP 2—COMPUTE THE MATCH SCORE:  
5:   for  $r \leftarrow 0 \dots (w_I - w_R)$  do ▷ place  $R$  at  $(r, s)$   
6:     for  $s \leftarrow 0 \dots (h_I - h_R)$  do  
7:       Get match score for template placed at  $(r, s)$ :  
8:        $q \leftarrow 0$   
9:       for  $i \leftarrow 0 \dots (w_R - 1)$  do  
10:        for  $j \leftarrow 0 \dots (h_R - 1)$  do  
11:          if  $R(i, j) = 1$  then ▷ foreground pixel in template  
12:             $q \leftarrow q + D(r + i, s + j)$   
13:           $Q(r, s) \leftarrow q / K$   
  
13:   return  $Q$ .
```

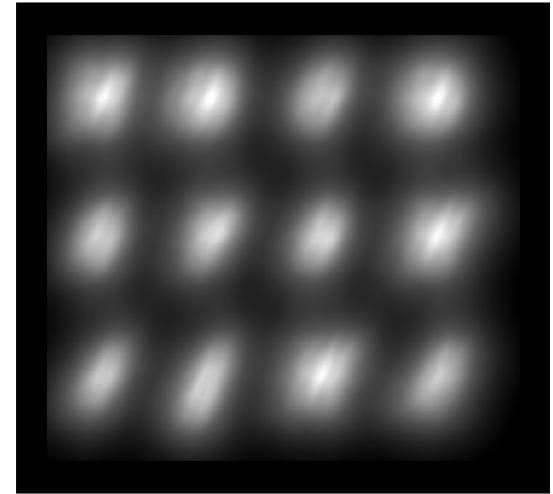
Primer



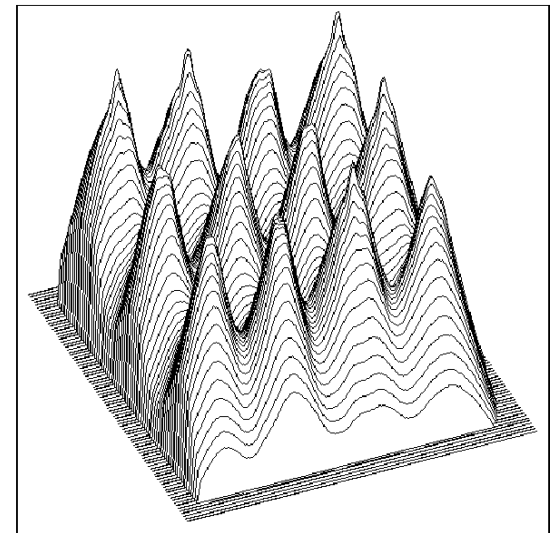
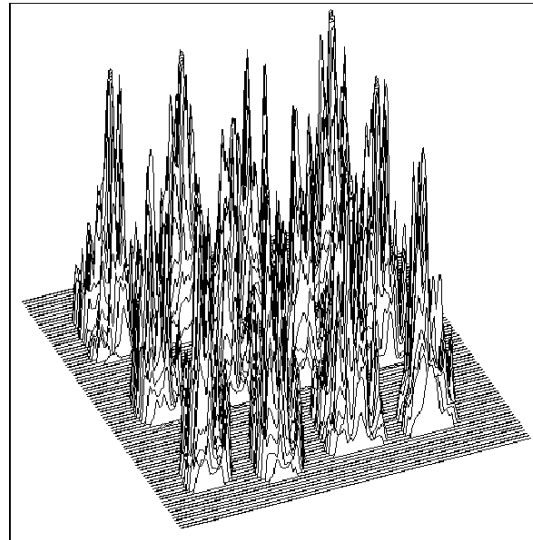
direct comparison



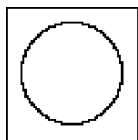
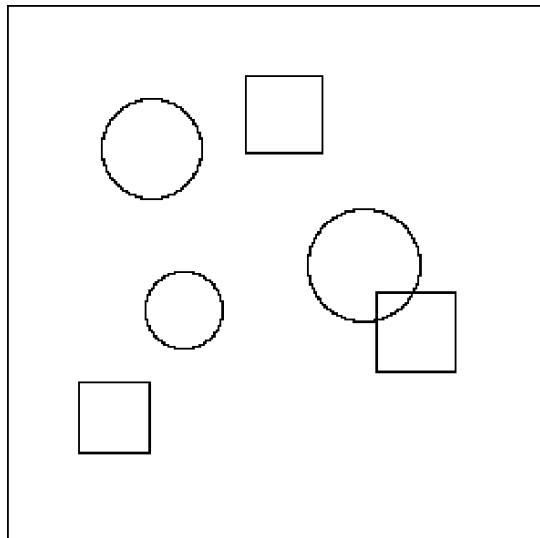
chamfer matching



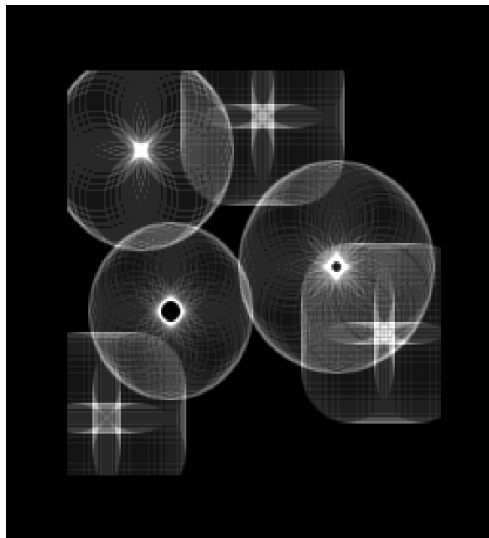
- Gladka funkcija razlik
- Omogoča uporabo lokalnih metod za iskanje maksimumov



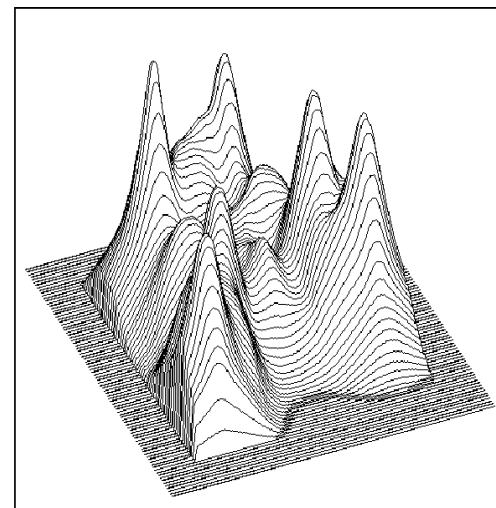
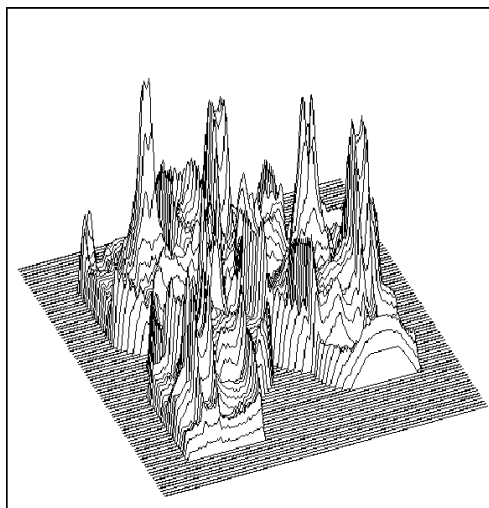
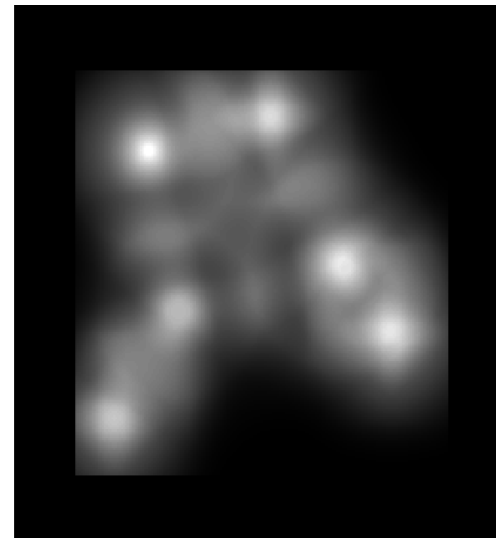
Primer



direct comparison



chamfer matching



- Tolerantno na majhne spremembe velikosti
- Ni tolerantno na premik, rotacijo, skalo