

# Sisteme de recunoastere a formelor – Lab 4

## Transformarea distanta (DT). Potrivirea modelelor folosind DT

### 1. Obiective

In acest laborator vom studia un algoritm care efectueaza transformarea distanta a unei imagini binare (obiect si fundal). Transformarea permite evaluarea unui scor de potrivire a modelului unui obiect cunoscut (de exemplu un contur de pieton) si un obiect necunoscut, pentru a decide daca obiectul necunoscut este sau nu similar cu obiectul model. Cu cat scorul de potrivire este mai mic, cu atat obiectul necunoscut este mai asemanator modelului.

### 2. Fundamente teoretice

#### 2.1. Transformata distanta

O transformare distanta, cunoscuta si ca harta de distante, sau camp de distante, este o reprezentare a unei imagini digitale. Alegerea termenilor depinde de un punct de vedere, si anume daca imaginea initiala este transformata in alta reprezentare, sau este imbunatatita cu o informatie suplimentara (harta sau camp). Harta ofera fiecarui pixel din imagine informatia privitoare la distanta sa fata de cel mai apropiat pixel obiect (sau obstacol). Cel mai des intalnit tip de pixel obiect este un punct de muchie.

Transformata distanta este un operator care se aplica in mod normal doar imaginilor binare. Rezultatul transformarii este o imagine grayscale, care seamana cu imaginea intrare, dar in care intensitatile punctelor arata distanta fata de cel mai apropiat punct de muchie.

In imaginea urmatoare avem un exemplu de aplicare a transformatei distanta pe o imagine ce contine o forma dreptunghiulara. In imaginea din stanga, pixelii cu valoarea "0" sunt pixeli obiect (puncte de muchie), iar cei cu valoarea "1" puncte de fundal. In imaginea din dreapta, se vede rezultatul aplicarii DT folosind metrica "tabla de sah", unde fiecare valoare codifica distanta fata de un punct de muchie.

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

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

Binary Image

Distance transformation

De obicei transformata/harta este denumita pe baza metricii alese. De exemplu, se poate vorbi de transformata distanta Manhattan, daca metrica folosita este distanta Manhattan. Alte metrice sunt:

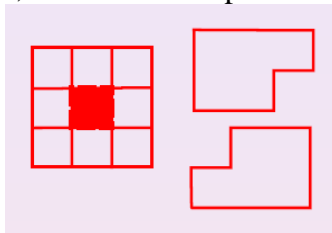
- Distanta Euclidiană;
- Distanta *City block* sau *Manhattan*;
- Distanta „tabla de sah”.

Exista mai multi algoritmi pentru implementarea DT:

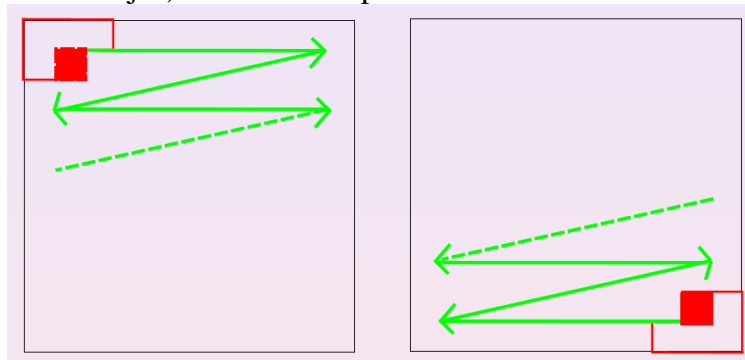
- Chamfer DT;
- Euclidian DT;
- Voronoi diagram DT.

Vom prezenta transformata distanta Chamfer, care este simpla si foarte rapida, necesitand doar doua parcurgeri ale imaginii binare. O vedere de ansamblu a algoritmului este:

- Se alege o masca 3x3, care se descompune in doua parti:



- O dubla parcurgere (prima data sus-jos, stanga-dreapta, a doua oara jos-sus, dreapta-stanga) a imaginii (cu jumatatile corespunzatoare ale mastii, vezi figura de mai jos) este necesara pentru a actualiza distanta minimala:



In timpul parcurgerii imaginii sursa (direct si invers), se face urmatoarea actualizare pe imaginea DT:

$$DT(i,j) = \min_{(k,l) \in Mask} (DT(i+k, j+l) + weight(k,l))$$

- Inicializarea imaginii DT se face in felul urmator:

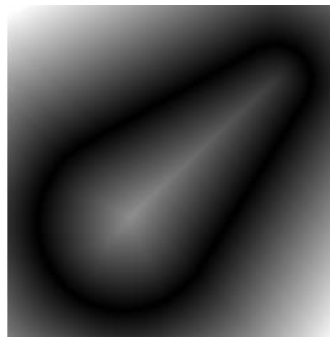
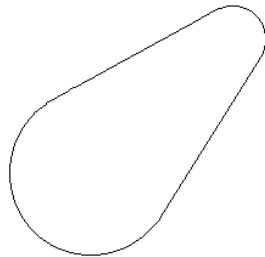
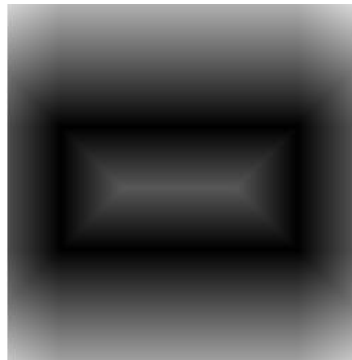
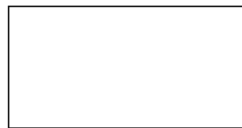
$$DT(i,j) = 0 \quad \text{if } (i,j) \in Object$$

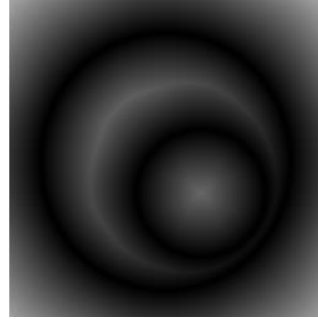
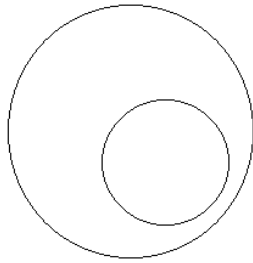
$$DT(i,j) = +\infty \quad \text{if } (i,j) \notin Object$$

- Precizari:

- o Deplasamentul (0,0) cu ponderea 0 apartine mastii;
- o Pentru o imagine pe 8 biti pentru infinit se foloseste valoarea 255;
- o Daca dorim sa obtinem o imagine DT unde fiecare pixel aproximeaza distanta Euclidiana la cel mai apropiat obiect, atunci ponderea  $weight(k,l)$  trebuie sa aiba valoarea:
  - $w_{HV}$  daca directia  $(k,l)$  este orizontala sau verticala fata de originea mastii
  - $w_D$  daca directia  $(k,l)$  este diagonala fata de originea mastii
  - relatia dintre  $w_{HV}$  si  $w_D$  trebuie sa fie  $w_D \approx w_{HV} \cdot \sqrt{2}$
  - o alegere buna este  $(w_{HV}, w_D) = (5, 7)$  deoarece  $7 \approx 5 \cdot \sqrt{2}$

Exemple de imagini DT obtinute cu metoda Chamfer, cu valorile  $(w_{HV}, w_D) = (2,3)$ . In stanga e imaginea originala, in dreapta imaginea DT.





## 2.2. Potrivirea de modele folosind DT

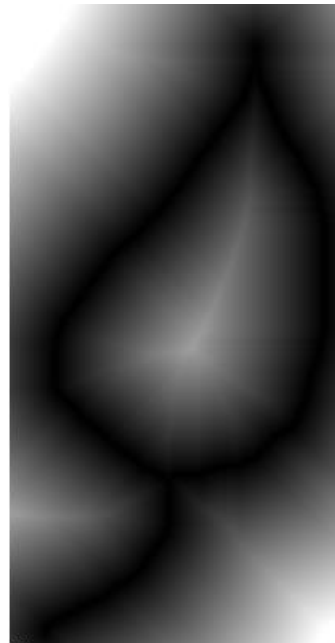
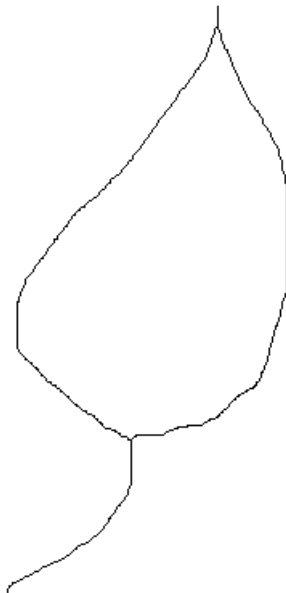
Dorim sa calculam un scor de potrivire intre un model de obiect cunoscut, si un obiect necunoscut. Consideram ca ambele obiecte au aceeasi dimensiune.

Pasii pentru calcularea scorului sunt:

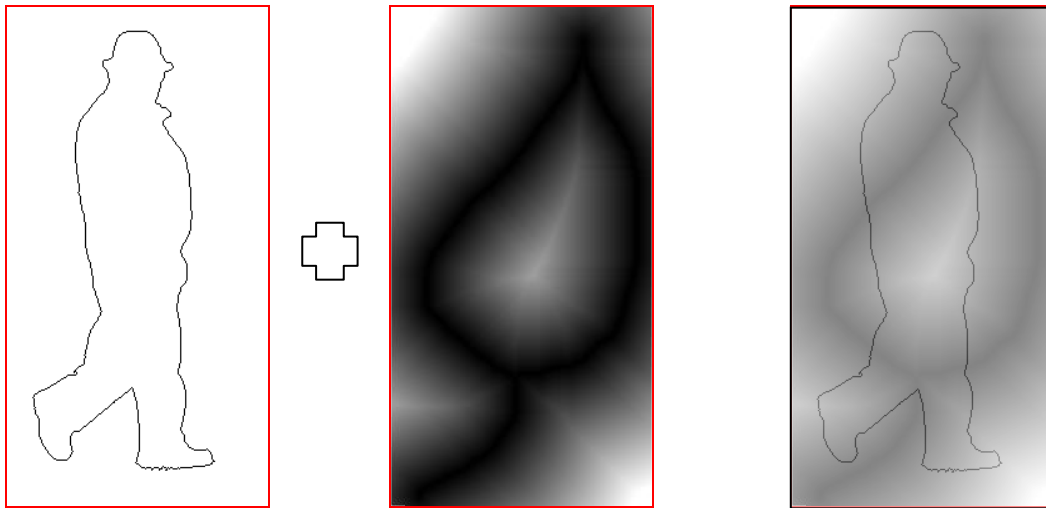
- Calcularea imaginii DT pentru imaginea de model;
- Suprapunerea obiectului necunoscut peste imaginea DT calculata anterior;
- Scorul de potrivire este o media a tuturor valorilor pixelilor din imaginea DT care se afla sub punctele de pe conturul obiectului necunoscut.

Exemplu: Consideram ca obiectul model este o frunza (contur), si obiectul necunoscut este un pieton (contur):

- Calculam imaginea DT a frunzei:



- Suprapunem pietonul peste imaginea DT a frunzei; calculam scorul de potrivire ca media valorii tuturor pixelilor din imaginea DT a frunzei care cad sub pixelii conturului pietonului.



### **3. Detalii de implementare**

Citirea pozei de intrare ca si o poza grayscale:

```
Mat img = imread("filename", IMREAD_GRAYSCALE);
```

Initializarea imaginii DT cu poza de intrare:

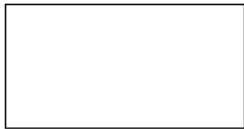
```
Mat dt = src.clone();
```

Accesarea pixelilor dintr-o vecinatate de tip 8:

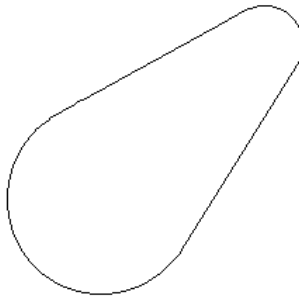
```
int di[8] = {-1,-1,-1,0,0,1,1,1};  
int dj[8] = {-1,0,1,-1,1,-1,0,1};  
int weight[8] = {0,1,0,1,1,0,1,0};  
for(int k=0; k<8; k++)  
    uchar pixel = img.at<uchar>(i+di[k], j+dj[k]);
```

#### 4. Activitate practica

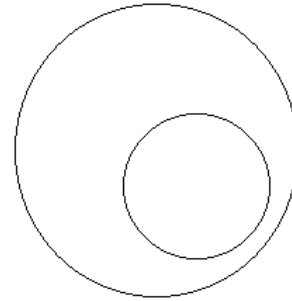
1. Implementati Transformata Distanta Chamfer. Calculati si vizualizati imaginea DT pentru pozele de intrare: *contour1.bmp*, *contour2.bmp*, *contour3.bmp*. Rezultatele trebuie sa coincida cu cele prezentate in text. Pixelii de obiect sunt negri iar fundalul este alb.



*contour1.bmp*

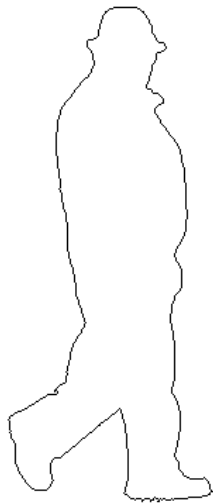


*contour2.bmp*

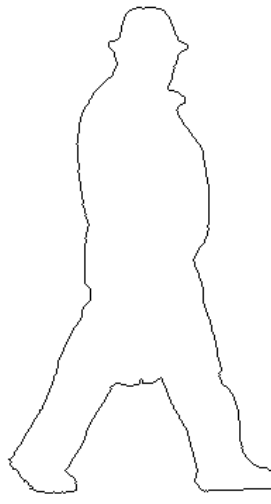


*contour3.bmp*

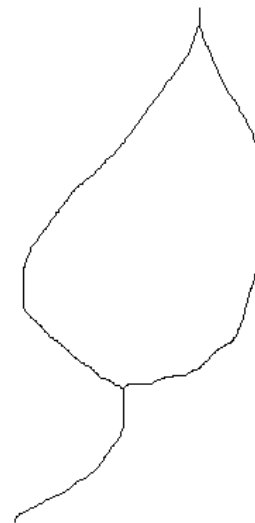
2. Calculati imaginea DT pentru *template.bmp*. Evaluati scorul de potrivire intre imaginea model si cele doua obiecte necunoscute: *unknown\_object1.bmp* – pieton, *unknown\_object2.bmp* – frunza. Scorul de potrivire este media valorilor din imaginea DT de pe pozitiile punctelor de contur ale obiectului necunoscut.



*template.bmp*



*unknown\_object1.bmp*



*unknown\_object2.bmp*

3. Calculati scorul de potrivire prin inversarea rolurilor de obiect necunoscut si obiect model.

4. Inainte sa calculati scorul translatati obiectul necunoscut astfel incat centrul lui de masa sa coincida cu centrul obiectului model. Pentru simplitate, centrele de masa se estimeaza din punctele de pe contur.

5. Optional, implementati Transformata Distanta Euclidiana adevarata. De ce este diferita fata de Transformata Distanta Chamfer?

## **5. Referinte**

[1] Wikipedia The Free Encyclopedia – *Distance Transform*,

[http://en.wikipedia.org/wiki/Distance\\_transform](http://en.wikipedia.org/wiki/Distance_transform)

[2] D. Coeurjolly – *Distance Transform*, University Claude Bernard Lyon

[http://www.cb.uu.se/~tc18/subfields/distance\\_skeletons/DistanceTransform.pdf](http://www.cb.uu.se/~tc18/subfields/distance_skeletons/DistanceTransform.pdf)

[3] Compendium of Computer Vision – *Distance Transform*,

<http://homepages.inf.ed.ac.uk/rbf/HIPR2/distance.htm>