Univerza *v Ljubljani*
Fakulteta *za elektrotehniko*

Computer Vision 10 – Image processing and analysis 2b

doc. dr. Janez Perš
(with contributions by prof. Stanislav Kovačič)

Laboratory for Machine Intelligence
Faculty of Electrical Engineering
University of Ljubljana

Quick recap of the previous lectures

- Image formation
- Color
- Image processing
- Edges, Hough transform
- Segmentation, morphological operations

Outline

- Image analysis, continued

- Morphological operations on grayscale images
- Distance transform
- Region boundaries & active contours
- Region descriptors, moments

Morphological operations on grayscale images

- Until now, we always assumed thresholding *before* applying
  - Erosion
  - Dilation
  - Opening
  - Closing

- Can those operations be performed on grayscale images?
  - Yes, we can expand the definition of erosion and dilation to fit grayscale images as well

Erosion and dilation on grayscale images

- Now the image is not constrained to 0 and 1
  - Instead of forcing the central element to 0 or 1, we substitute it with the min or max pixelwise operation!
- Example:
  - SE – structuring element of ones in 5x5 window
  - Slide SE through the image
- Erode: min(Image, SE)
  - therefore, if at least one of pixels within SE belongs to background (0), the central pixel will become 0
- Dilate: max(Image, SE)
  - therefore, if at least one of pixels within SE belongs to foreground (255), the central pixel will become 255

Distance transform

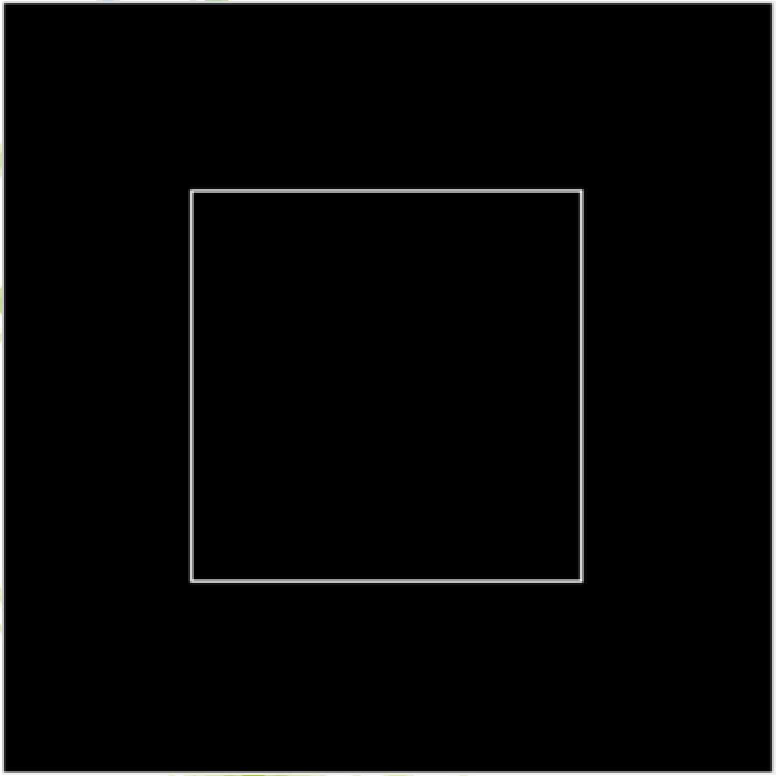
- Mostly used on binary images
  - Pixel values are replaced with the distance to important structures in the image, such as background.

```
. . . . X . . .
. . X X X X X .
. . X X X X X .
. . X X X X X .
. . . . X . . .
```

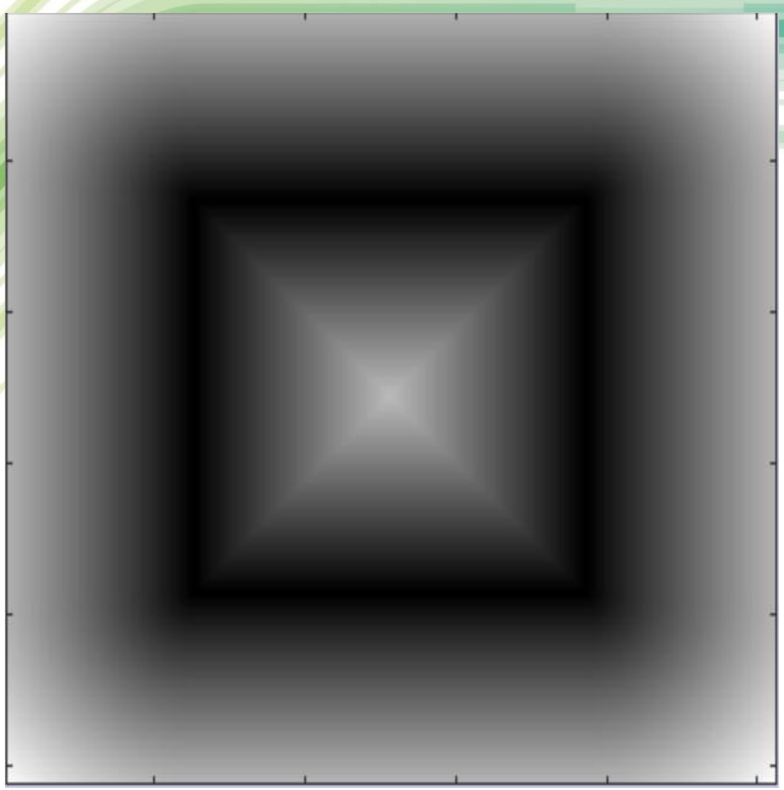
```
. . . . 1 . . .
. . 1 1 2 1 1 .
. . 1 2 3 2 1 .
. . 1 1 2 2 1 .
. . . . 1 . . .
```

- But typical “important structures” are edges, or silhouettes
- Note: dots could be labeled as well.

Distance transform - illustration

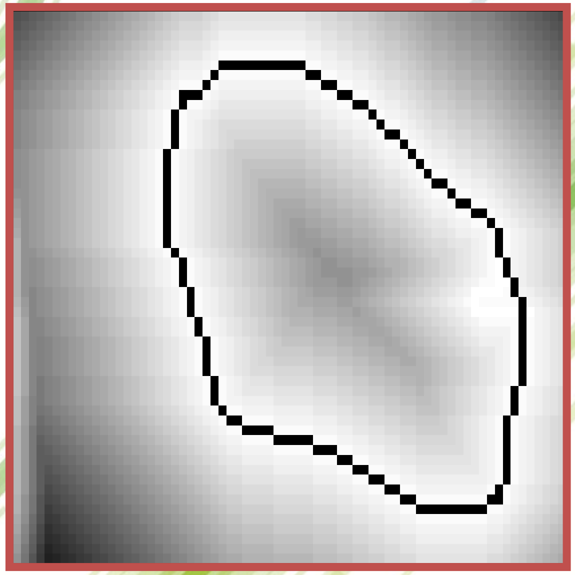


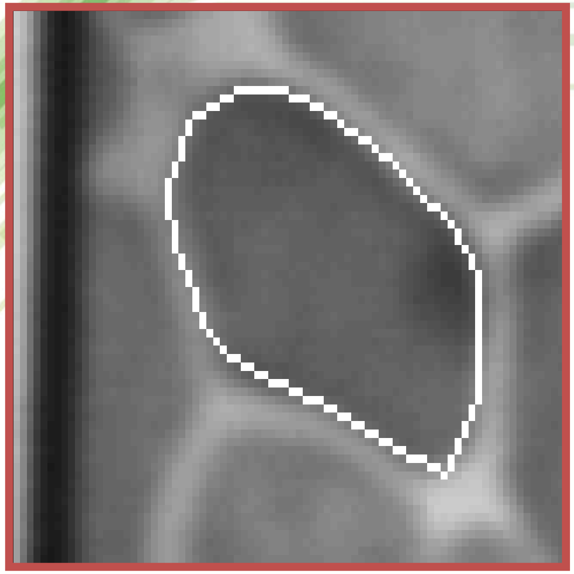
Input image: edge pixels (e.g. after edge detection)



Distance transform. Pixel values are proportional to the pixel distance from the edge

Distance transform - illustration





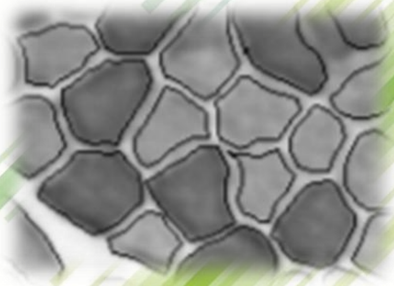
- Many use cases – e.g. template matching
  - How does it work in that case?

Other morphological operations

- Top hat (transform):
  - Result = original image – opening
- Bottom hat (transform):
  - Result = original image – closing
- Watershed transform:
  - Think of an image as a landscape of mountains (light regions) and valleys (dark regions)
  - then flood the valleys.
  - Ridges define solution.

Contour based approaches

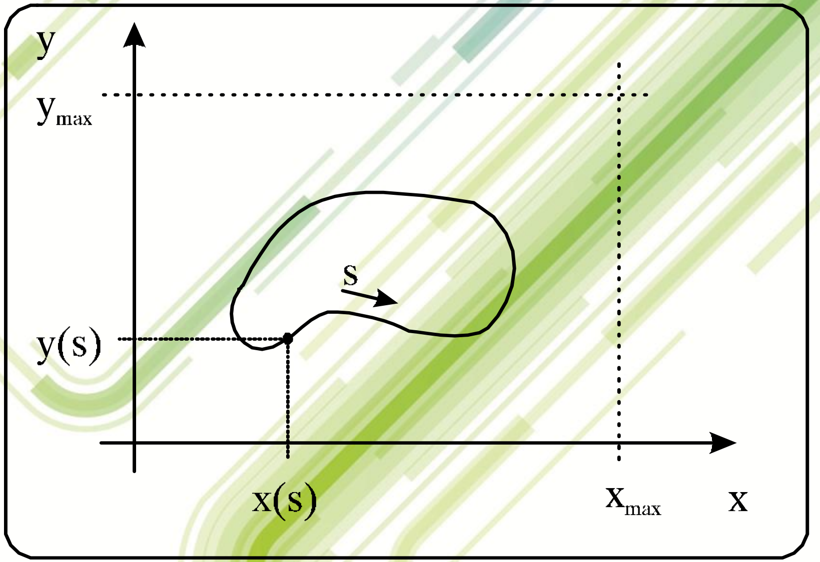
- Often, the best way do describe a region is by specifying the boundary



- Boundary = contour = contour based approaches.

Region description using boundary

- Object region is described by pixels on the boundary



Input: 2D image

Output: parametric curve

I(x, y) → (x(s), y(s))

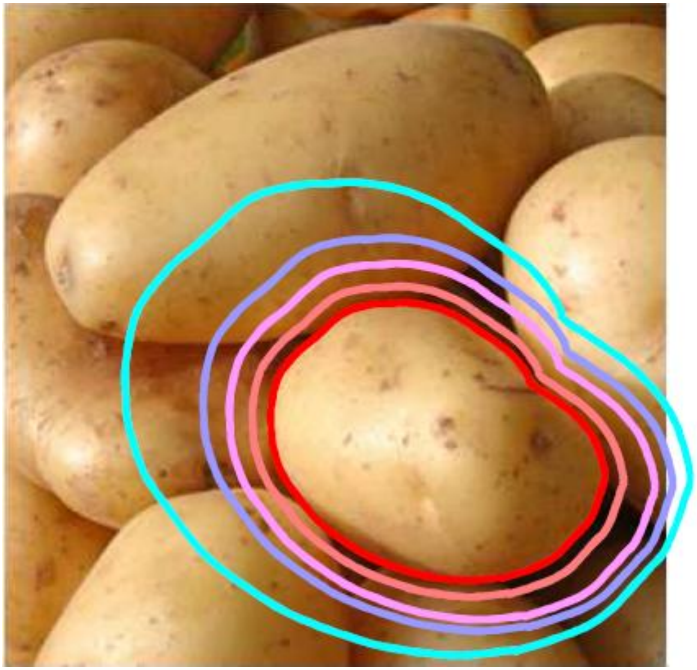
- Various algorithms for contour following (tracing) exist. We will not discuss those.

Active contour models

- Active contour models – “snakes”
  - (Kass, Witkin, Terzopoulos, 1988):
  - Contour is placed onto the image.
  - Contour acts as an elastic string that is moving, therefore “snake”.
- The contour

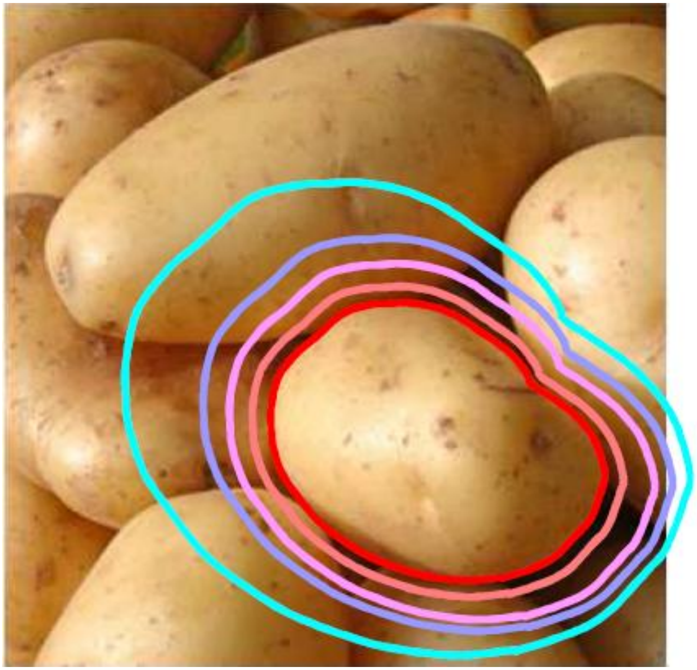
  $v(s)=(x(s),y(s)),\ s \in [0,1]$

  is fitted to the boundary of image region



Active contour models

- $v(s)=(x(s),y(s))$, $s \in [0,1]$
- Two types of forces influence the contour
  - *Image forces* attract snake to distinctive image structures, like edges,
  - *internal snake forces* resist tearing or folding. That is, internal forces keep contour continuous and smooth.



Active contour models

- The contour v(s)=(x(s),y(s)), s $\in$ [0,1] has an energy functional

$$E(\mathbf{v}) = \int_0^1 \left(E_{int}(\mathbf{v}(s)) + E_{ext}(\mathbf{v}(s))\right) ds$$

Internal (snake) energy
Increases if snake is
forced to deform

External (image) energy
Decreases if snake fits the
Image feature (contour) well

- Task: minimize E(v),
  - That is: produce best fit with minimum deformation

Active contour models

- Internal energy: smoothness constraint

$$E_{int}(\mathbf{v}(s)) = \frac{1}{2}\left(\alpha(s)\left|\frac{\partial \mathbf{v}(s)}{\partial s}\right|^2 + \beta(s)\left|\frac{\partial^2 \mathbf{v}(s)}{\partial s^2}\right|^2 \right)$$

Makes snake to behave like elastic band, prevents tearing

Makes snake to behave like a string, prevents bending

Active contour models

- Various implementation approaches exist:
  - direct discretization of energy functional
  - solution of Euler differential equation

$$-\alpha(s)\frac{\partial^2 \mathbf{v}(s)}{\partial s^2}+\beta(s)\frac{\partial^4 \mathbf{v}(s)}{\partial s^4}+\nabla P(\mathbf{v}(s))=0$$

Internal energy terms

External – image energy (that attracts the snake towards image edges)

Snake in the discrete form

- Using the finite differences method to solve the differential equation

$$\frac{1}{h}\left(\frac{\alpha_i}{h}(\mathbf{v}_i-\mathbf{v}_{i-1})-\frac{\alpha_{i+1}}{h}(\mathbf{v}_{i+1}-\mathbf{v}_i)\right)+$$

$$+\frac{1}{h^2}\left(\frac{\beta_{i-1}}{h^2}(\mathbf{v}_{i-2}-2\mathbf{v}_{i-1}+\mathbf{v}_i)-\frac{2\beta_i}{h^2}(\mathbf{v}_{i-1}-2\mathbf{v}_i+\mathbf{v}_{i+1})+\frac{\beta_{i+1}}{h^2}(\mathbf{v}_i-2\mathbf{v}_{i+1}+\mathbf{v}_{i+2})\right)$$

$$+\nabla P(\mathbf{v}_i)=0.$$

$$\mathbf{Av}+\nabla P(\mathbf{v})=0$$

A – stiffness matrix

v – vector of discrete contour points

$\nabla P(\mathbf{v})$ - image “potential field”

Snake in the discrete form

- Using the finite differences method to solve the differential equation

$$\frac{1}{h}\left(\frac{\alpha_i}{h}(\mathbf{v}_i-\mathbf{v}_{i-1})-\frac{\alpha_{i+1}}{h}(\mathbf{v}_{i+1}-\mathbf{v}_i)\right)+$$

$$+\frac{1}{h^2}\left(\frac{\beta_{i-1}}{h^2}(\mathbf{v}_{i-2}-2\mathbf{v}_{i-1}+\mathbf{v}_i)-\frac{2\beta_i}{h^2}(\mathbf{v}_{i-1}-2\mathbf{v}_i+\mathbf{v}_{i+1})+\frac{\beta_{i+1}}{h^2}(\mathbf{v}_i-2\mathbf{v}_{i+1}+\mathbf{v}_{i+2})\right)$$

$$+\nabla P(\mathbf{v}_i)=0.$$

$$\mathbf{Av}+\nabla P(\mathbf{v})=0$$

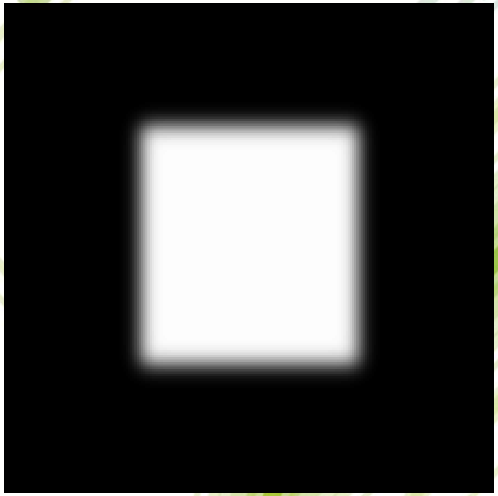
A – stiffness matrix

v – vector of discrete contour points

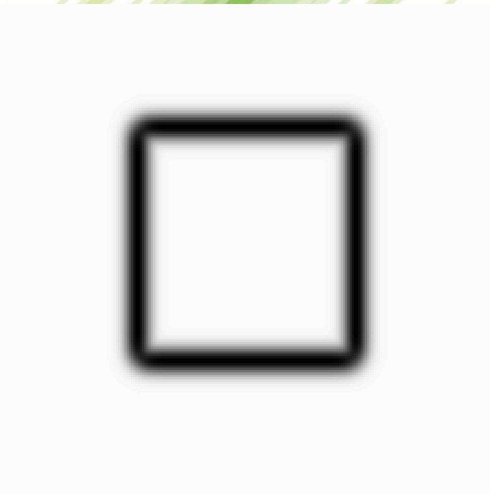
$\nabla P(\mathbf{v})$ - image “potential field”

External energy – an example

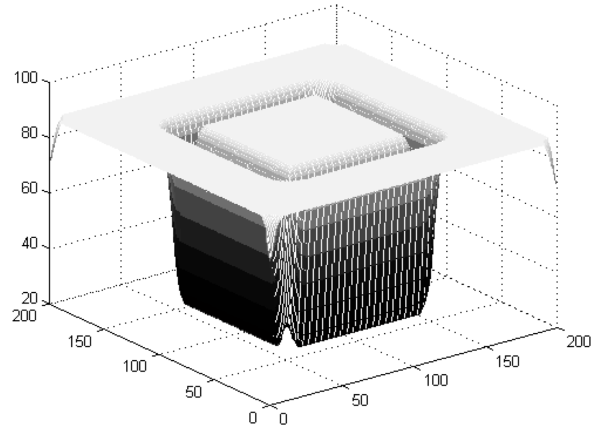
- Snake is attracted to the shape in the first image



image

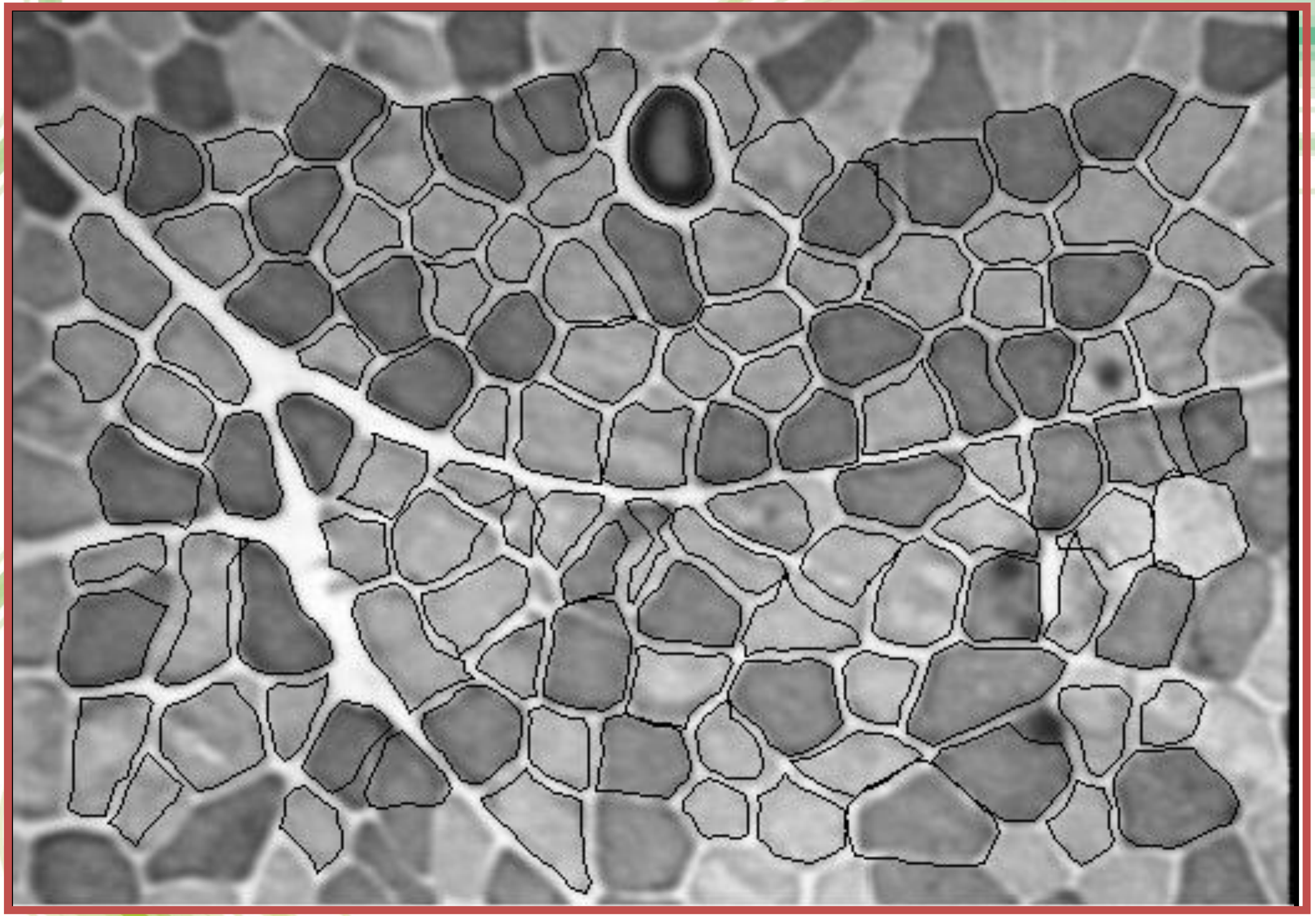


gradient



ext. energy

Snakes – segmentation example



- Initialization has to be provided (e.g. manual)

Snakes - problems

- Snakes are very sensitive to local minima
- Initialization of snake is a serious problem by itself
- All that snake does is smoothing the solution, no shape preferences
- Thus, many improvements are possible, and have been developed:
  - various snake derivatives,
  - and later point distribution models (PDM),
  - active shape models (ASM), ...

Snakes in tracking

Kalman snakes (J. Denzler)

Condensation snakes
Particle filtering approach
(Isard & Blake)

(video!)

Region (2D shape) description

- Once the regions are detected and different regions labeled, we can do many things to describe regions, i.e. compute
  - position,
  - orientation,
  - bounding box,
  - shape factors,
  - area,
  - perimeter,
  - texture, appearance...
- Matlab: regionprops

- But before that, we usually apply various

Region (2D shape) description

- But before calculating region descriptors, we usually apply various "morphological operations", i.e., perform *morphological filtering* as required.

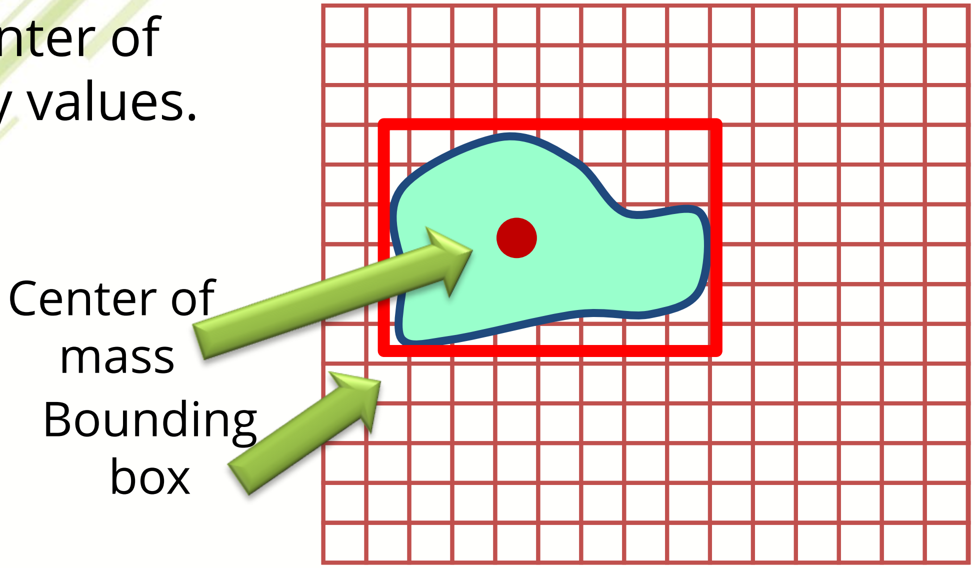
- Usually the whole procedure is as follows:
  - Image filtering/preprocessing
  - Image thresholding
  - Morphological operations (filtering)
  - Region analysis (region descriptors) as high level features

Region (2D shape) descriptors

- Size, area and / or perimeter, bounding box
- simply count pixels that belong to region / boundary
- Position
  - (e.g. x and y of the center of *mass*) = mean x and y values.

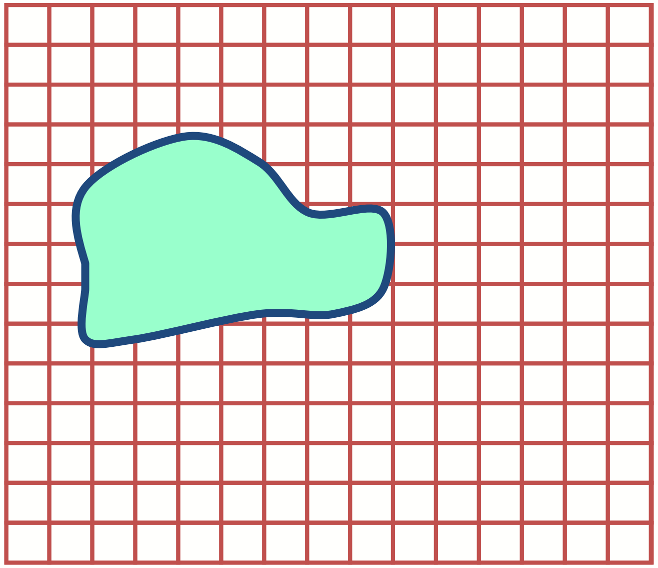
$$x_c = \sum_{x,y} x f(x,y) / \sum_{x,y} f(x,y)$$

$$y_c = \sum_{x,y} y f(x,y) / \sum_{x,y} f(x,y)$$



Region (2D shape) descriptors

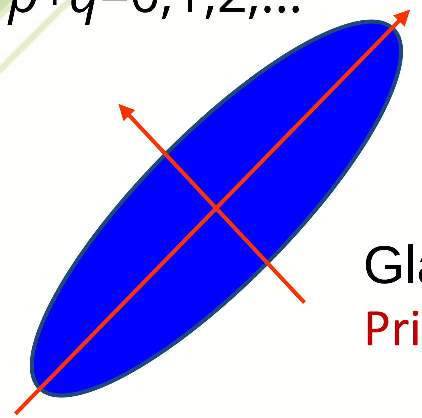
- Orientation
  - (e.g. principal axes, moments of inertia, covariance matrix)
- Shape factors
  - compactness, roundness, eccentricity, moments, …
- Fourier descriptors, bending energy, …



Region (2D shape) descriptors

- x and y are treated as random variables.
  - Raw (Statistical) moments of order *p*+*q*=0,1,2,…

$$m_{pq} = \sum_{x,y} x^p y^q f(x,y)$$



- Central moments

$$\mu_{pq} = \sum_{x,y} (x - x_c)^p (y - y_c)^q f(x,y)$$

  - Used for shape description, orientation estimation, normalization

Raw and central moments

- Image moments are essentially a generalization of “intuitive” descriptors.
  - For example, center of mass, as derived from raw moments:

$$x_c = \frac{m_{10}}{m_{00}}, \quad y_c = \frac{m_{01}}{m_{00}}$$

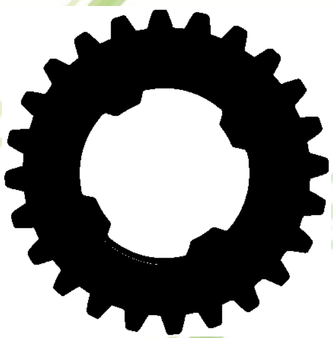
Shape/form factors

- Shape factors – global shape features
  - Compactness
  - Elongateness
  - Eccentricy
  - Moments

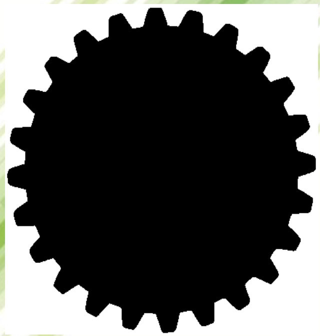
  …
  - Fourier descriptors
  - Bending energy (of an outline)

Compactness

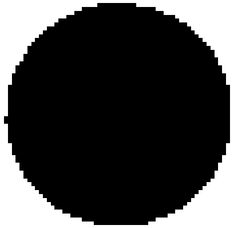
- Compactness = perimeter^2 / area



Compactness =7.4



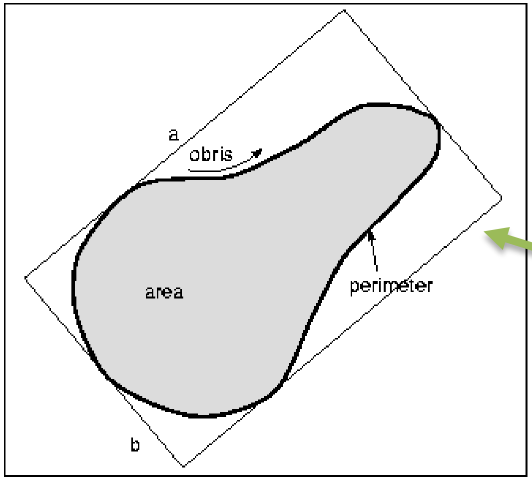
Compactness =2.8



Compactness =1.1
Almost a circle!

Eccentricity

- Eccentricity = major axis / perpendicular axis = (a / b)

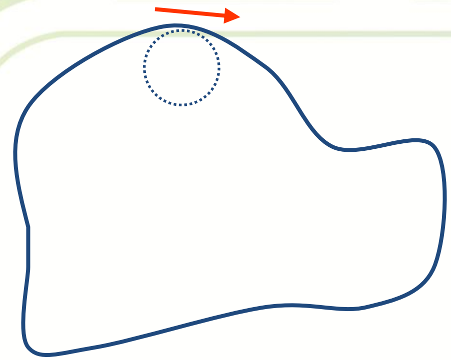


Bounding box

Shape from outline

Bending energy $BEN = \int_s k^2(s)ds$

$(x(s), y(s)), \quad (0 \leq s < L)$



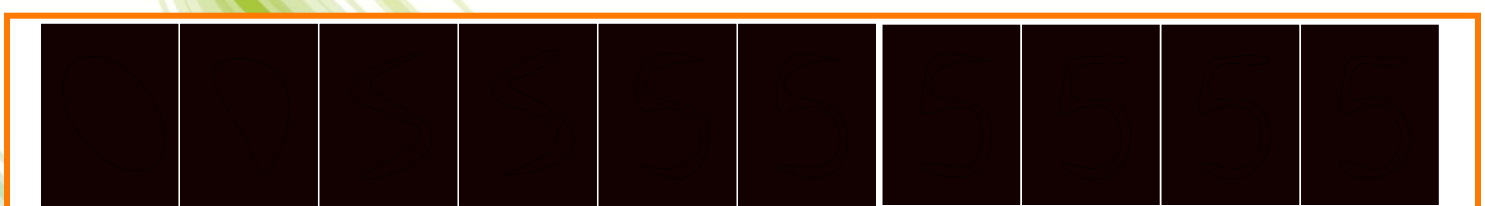
$$k = \frac{1}{r} = \left\| \frac{dT}{ds} \right\| = \frac{\dot{x}\ddot{y} - \ddot{x}\dot{y}}{\sqrt{(\dot{x}^2 + \dot{y}^2)^3}}$$

k : curvature

r : radius of osculating circle

dT : tangent vector

Fourier descriptors



Literature

- P. Corke, Robotics, Vision, and Control, 2013.
- R. Gonzalez, R. Woods, Digital Image Processing, 3rd Ed., 2007.
- R. Szeliski book, http://szeliski.org/Book/
- M. Sonka, V. Hlavac, R. Boyle, Image Processing, Analysis, and Machine Vision, 4th Ed., 2014.
- E. Trucco, A. Verri, Introductory Techniques for 3D Computer Vision, Prentice Hall, 1998.
- Web, journal papers, ...

Questions?