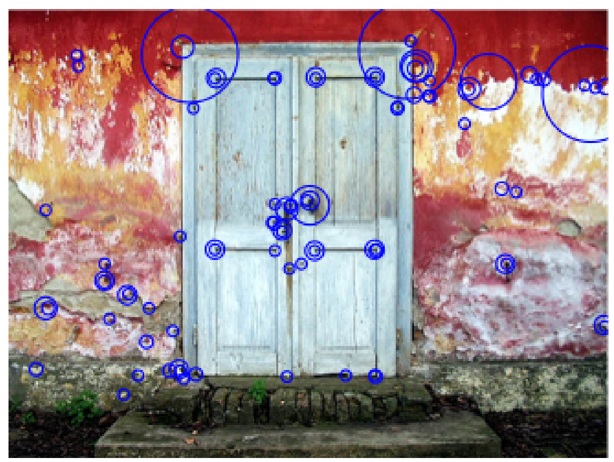
Feature Extraction

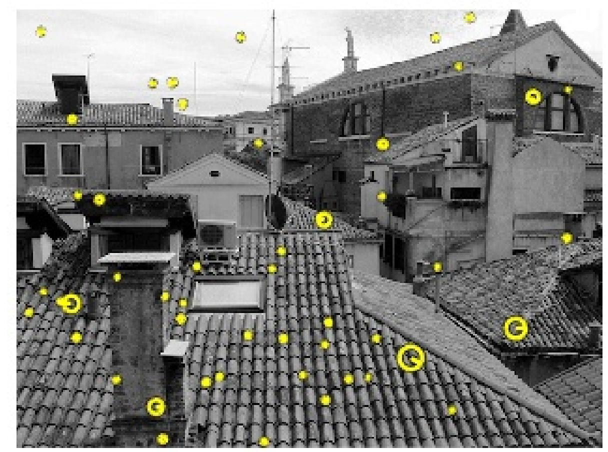
Eric Debreuve
Equipe-Projet Commune **Morpheme**
`http://team.inria.fr/morpheme`
Inria SAM / Lab. I3S / Lab. iBV

First, Some Examples

Harris (& Stephens) detector



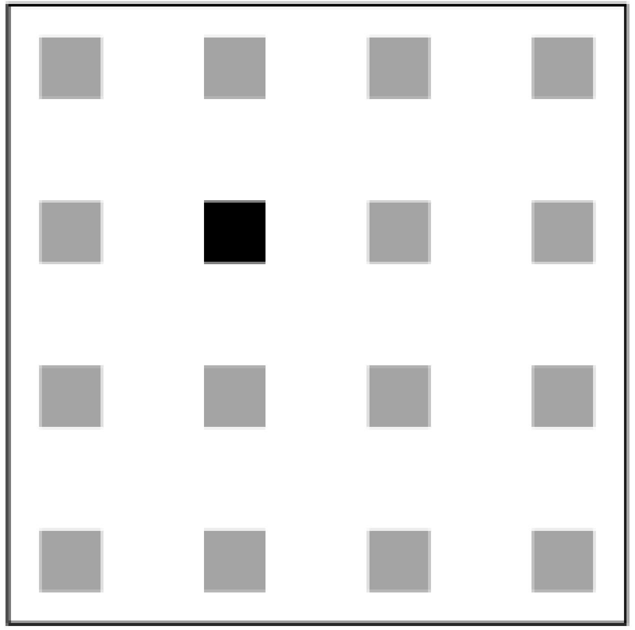
Harris-Laplace detector



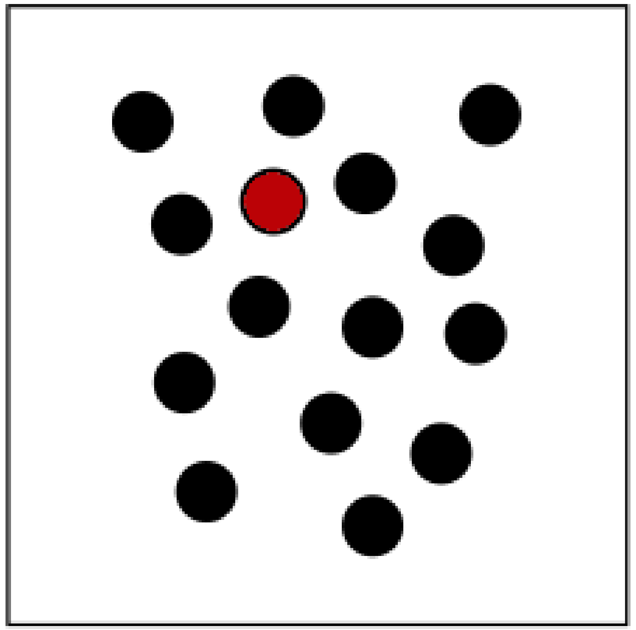
SIFT detector

Lindeberg detector: Laplacian of Gaussian (LoG)

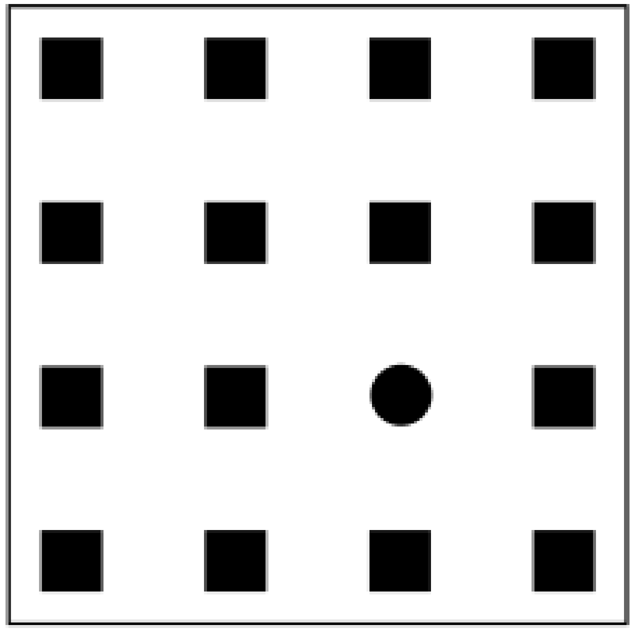
Visual Saliency



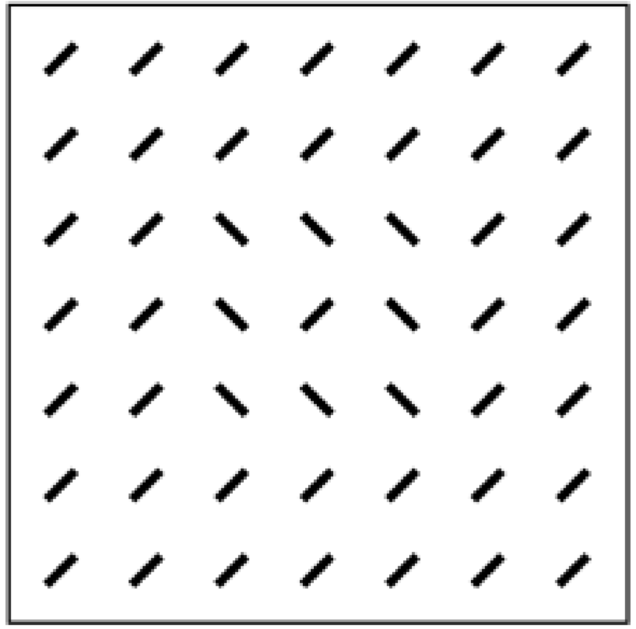
Intensity



Color



Curvature



Orientation

OUTLINE

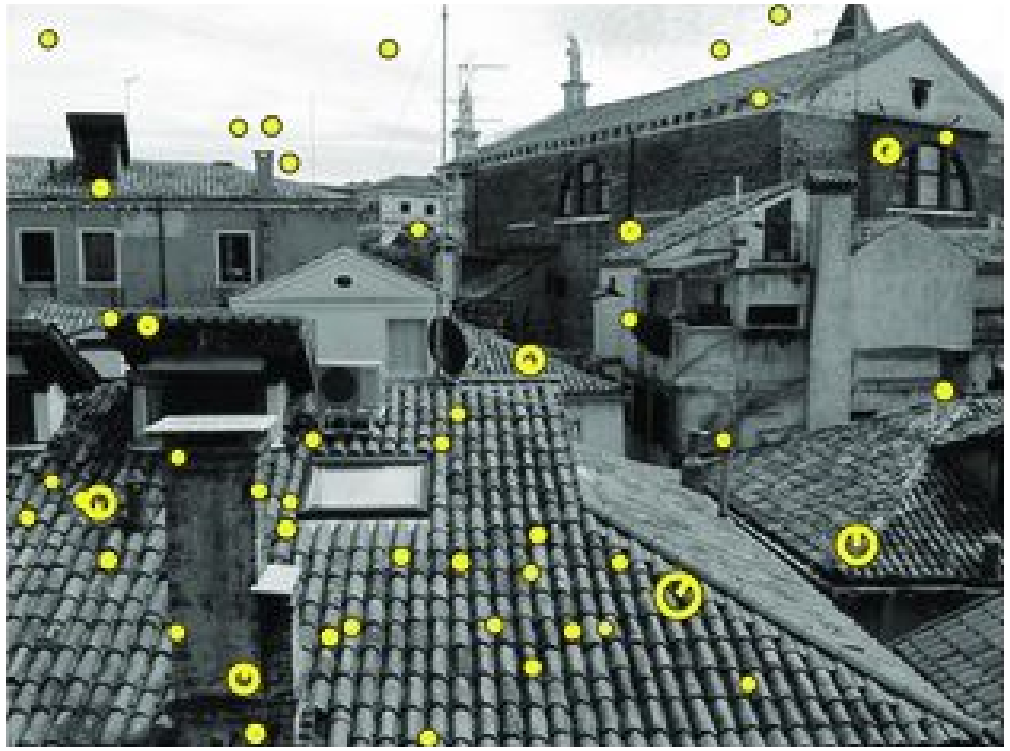
Keypoints (Sparse,[♯1] pixel-wise[♯2] features)

[♯1]As opposed to "image-wide"
[♯2]As opposed to "object-wise"

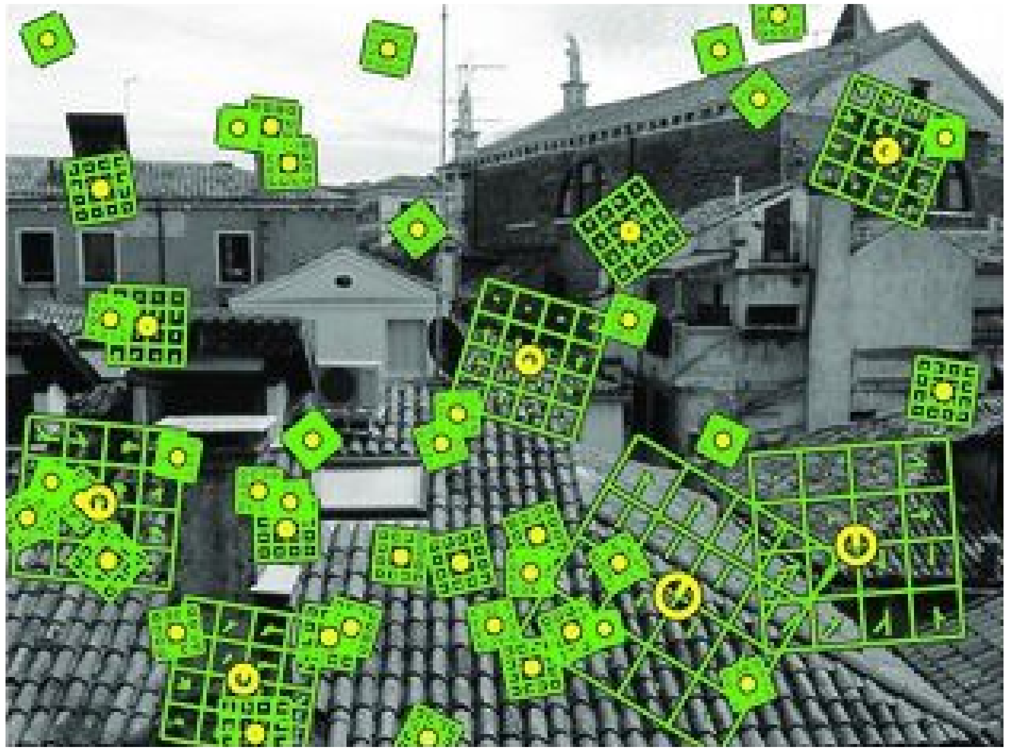
Points of Interest (Keypoints)

2 aspects

- Detection
  - Identify pixels in image which are of interest



- Description
  - Characterization of image in neighborhood of pixels of interest



But detection and description cannot be decoupled

- It is not possible to identify pixels of interest just by looking at individual pixels
- A context is needed

Terminology

- Keypoint = Corner in a wide sense

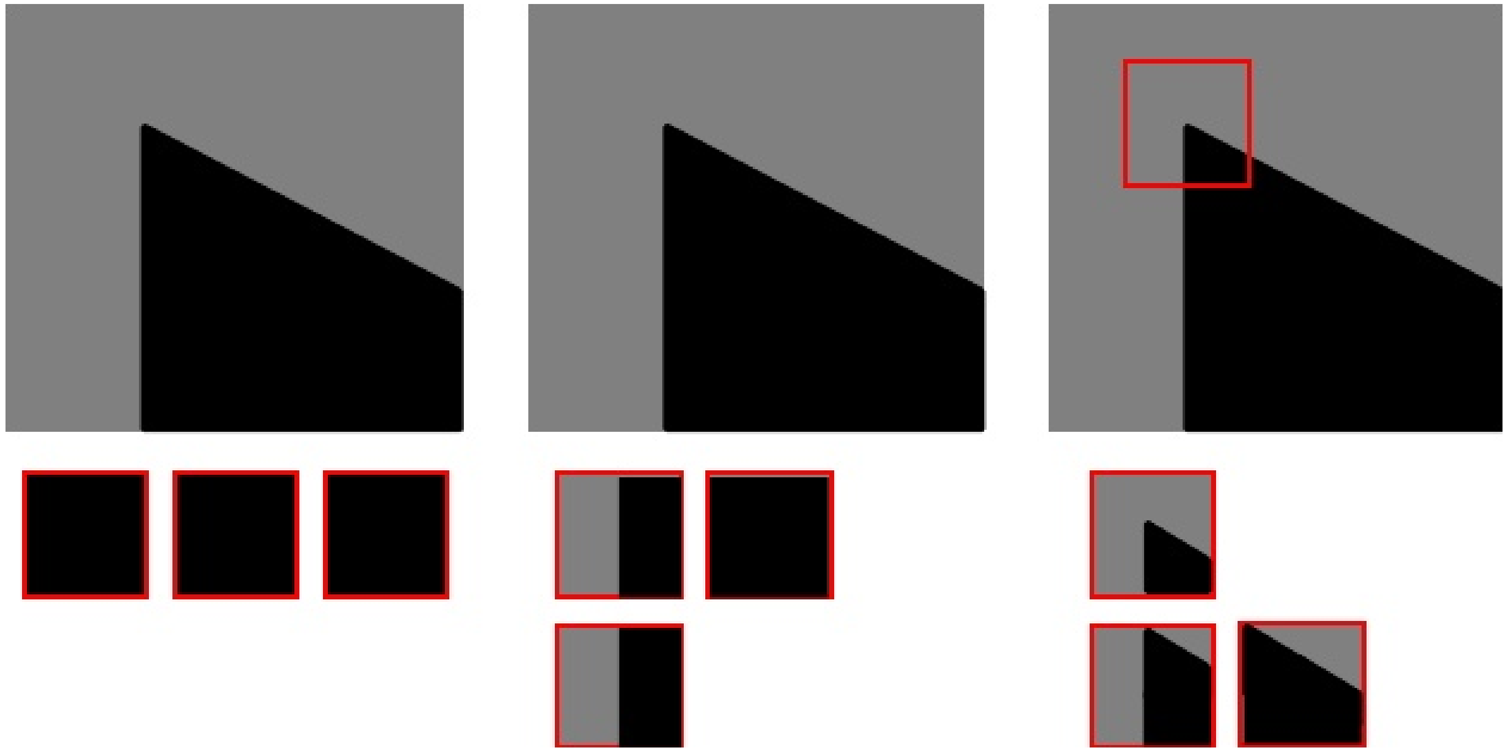
Keypoints Detectors

Keypoint

- Choose a definition
- Find a method to detect pixels following this definition

Example

- Moravec detector [1977]
  - Definition: large intensity variations in certain directions
  - Method: High patch dissimilarity among all 8 directions



Flat area Edge Corner

Harris (& Stephens) Detector

Improvement of Moravec detector

- "By smoothing" Moravec detector response
- Structure tensor (defined at each pixel (x, y) of I): $S = \begin{bmatrix} \overline{\left(\frac{\partial I}{\partial x}\right)^2} & \overline{\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}} \\ \overline{\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}} & \overline{\left(\frac{\partial I}{\partial y}\right)^2} \end{bmatrix}$ where $\bar{A} = \sum_{(x,y)\in W} g(x, y)\, A(x, y)$
- Classical choice: g = Gaussian

Candidates

- Large variations of S in all directions
- Such a behavior can be inferred from the eigenvalues λ_1 and λ_2 of S
  1. λ_1 and λ_2 are small: homogeneous area $\longrightarrow$ pixel is not a keypoint
  2. either λ_1 or λ_2 is large, the other is small: presence of an edge
  3. λ_1 and λ_2 are large: presence of a corner
- Proposed energy: $E(W) = \lambda_1 \lambda_2 - \beta(\lambda_1 + \lambda_2)^2$ where $\beta \in [0.04, 0.15]$
- Avoid costly computation of eigenvalues
  - $E(W) = \det S - \beta \operatorname{tr}^2 S$

Keypoints

- Local maxima of $E(W) >$ some threshold

Harris (& Stephens) Detector

1988

Detector parameters

1. Derivative filters $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$
2. Weighting window g
3. β
4. Neighborhood size for local maxima extraction
5. Threshold on the maxima values

Example

Weakness

- Not multi-scale

What does multi-scale mean?

- Close to image
- A little bit away
- Further away
- ...

Variant: Shi-Tomasi detector

- $E(W) = \min(\lambda_1, \lambda_2)$

Multi-scale Harris detector

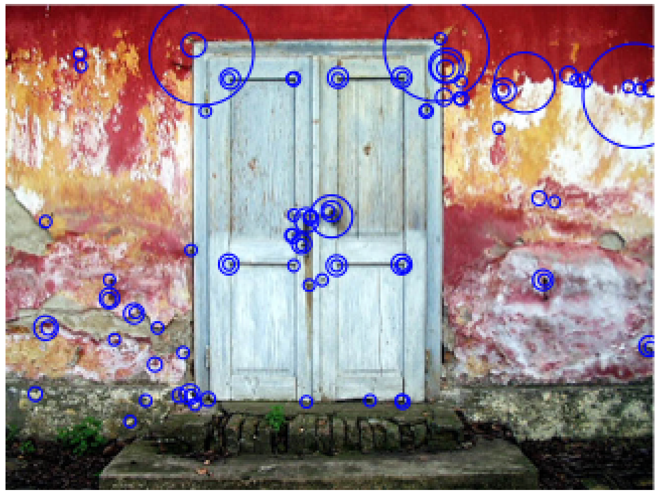
- σ_L: local scale
- σ_I: integration scale (parameter)
  - Common practice: $\sigma_I = \beta\,\sigma_L,\ \beta \in [1, 2]$
- Multi-scale representation: $L_{\sigma_L} = G_{\sigma_L} * I$
- Structure tensor
  - $S(\sigma_L) = \sigma_L^2\, G_{\sigma_I} *^{\sharp 1} \begin{bmatrix} \left(\frac{\partial L_{\sigma_L}}{\partial x}\right)^2 & \frac{\partial L_{\sigma_L}}{\partial x}\frac{\partial L_{\sigma_L}}{\partial y} \\ \frac{\partial L_{\sigma_L}}{\partial x}\frac{\partial L_{\sigma_L}}{\partial y} & \left(\frac{\partial L_{\sigma_L}}{\partial y}\right)^2 \end{bmatrix}$
- Two steps
  1. Candidates at scale σ_L
     - Local maxima of $E(\sigma_L) = \det S(\sigma_L) - \beta\,\mathrm{tr}^2\, S(\sigma_L)$
     - (like the Harris detector)
  2. Keypoints: keep local scale extrema of $\Delta^{\mathrm{norm}} L_{\sigma_L}$
     - (like the Lindeberg detector (see fe_10))

Improvement

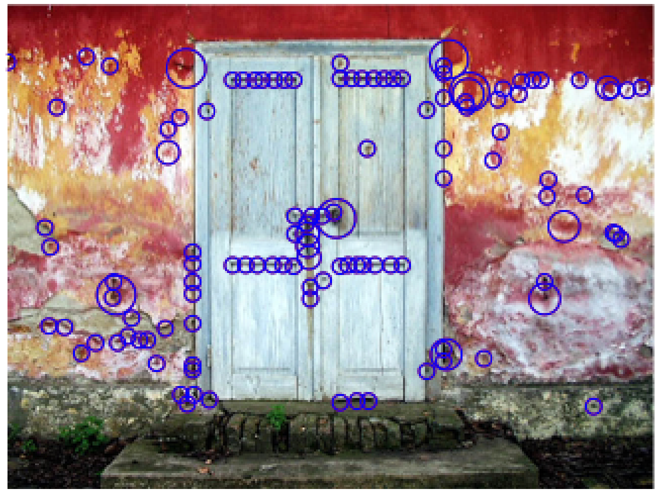
- Anisotropic integration: $\sigma_{I,1}$ and $\sigma_{I,2}$

⇒ Robustness to affine transforms when matching keypoints

Example



Points p_i with there scale σ_i: circles of radius $3\sigma_i$ centered at (x_i, y_i)



Blobs B_i: circles of radius $3\sigma_i$ centered at (x_i, y_i) (see fe_10)

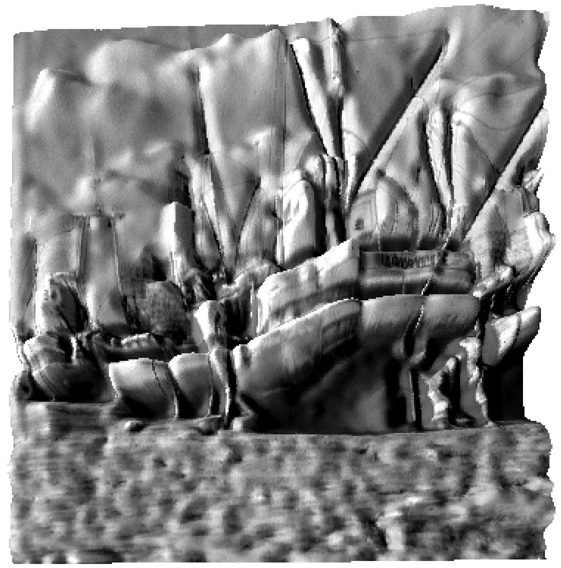
[♯1] Where the convolution applies to each element of the matrix

Kitchen & Rosenfeld

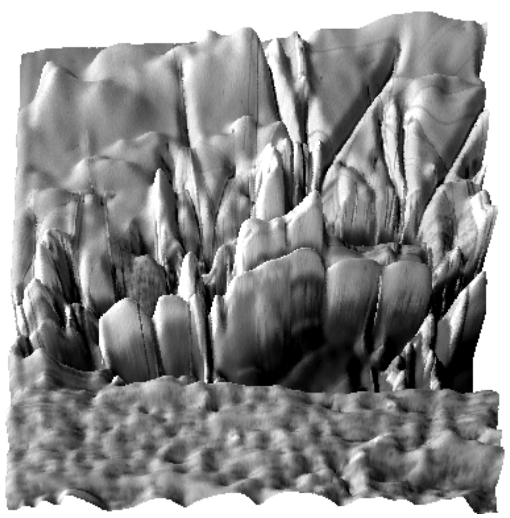
1982

Surface point of view

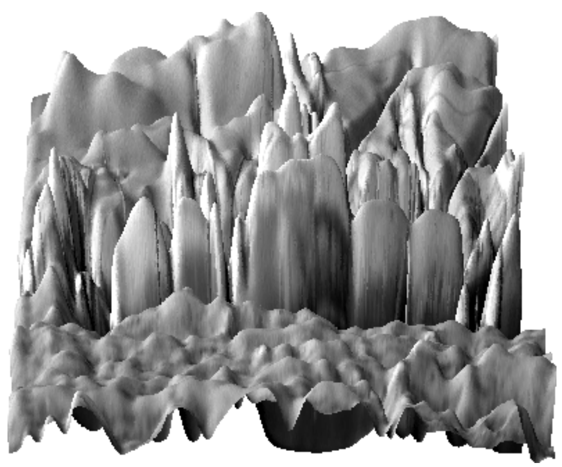
Image I(x,y) with level curves



Surface (x,y,I(x,y)) – from above



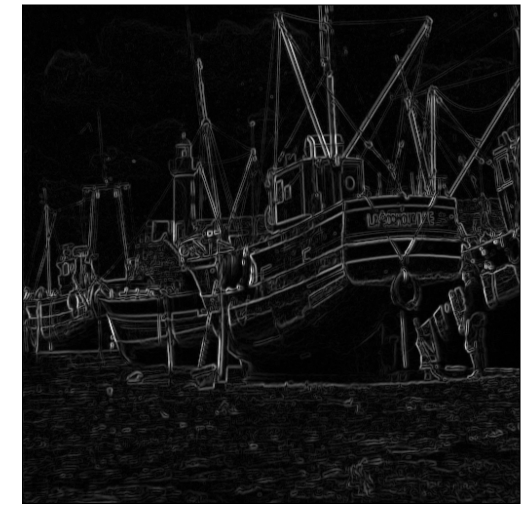
Surf. (x,y,I(x,y)) – from lower



Surf. (x,y,I(x,y)) – from even lower

Keypoint

- High curvature of level curve…
- and high gradient magnitude



Blob Detector

Lindeberg detector [1998]

- Based on Laplacian of Gaussian (LoG)
- Multi-scale

Method

- Multi-scale representation: $L_\sigma = G_\sigma * I$
  - E.g. $\sigma \in \{\sigma_i\}$, $\sigma_i \in [0, 100]$
- Laplacian
  - $$\begin{aligned}\Delta L_\sigma &= \frac{\partial^2 L_\sigma}{\partial x^2} + \frac{\partial^2 L_\sigma}{\partial y^2} \\ &= \underbrace{\left(\frac{\partial^2 G_\sigma}{\partial x^2} + \frac{\partial^2 G_\sigma}{\partial y^2}\right)}_{\mathrm{LoG}_\sigma} * I\end{aligned}$$
- Blobs
  - For a given scale σ, local (spatial) extrema of ΔL_σ
- Scale-independent blobs
  - Replace ΔL_σ with scale-normalized Laplacian $\Delta^{\mathrm{norm}} L_\sigma = \sigma^2 \, \Delta L_\sigma$
  - Blobs: local space-and-scale extrema of $\Delta^{\mathrm{norm}} L_\sigma$

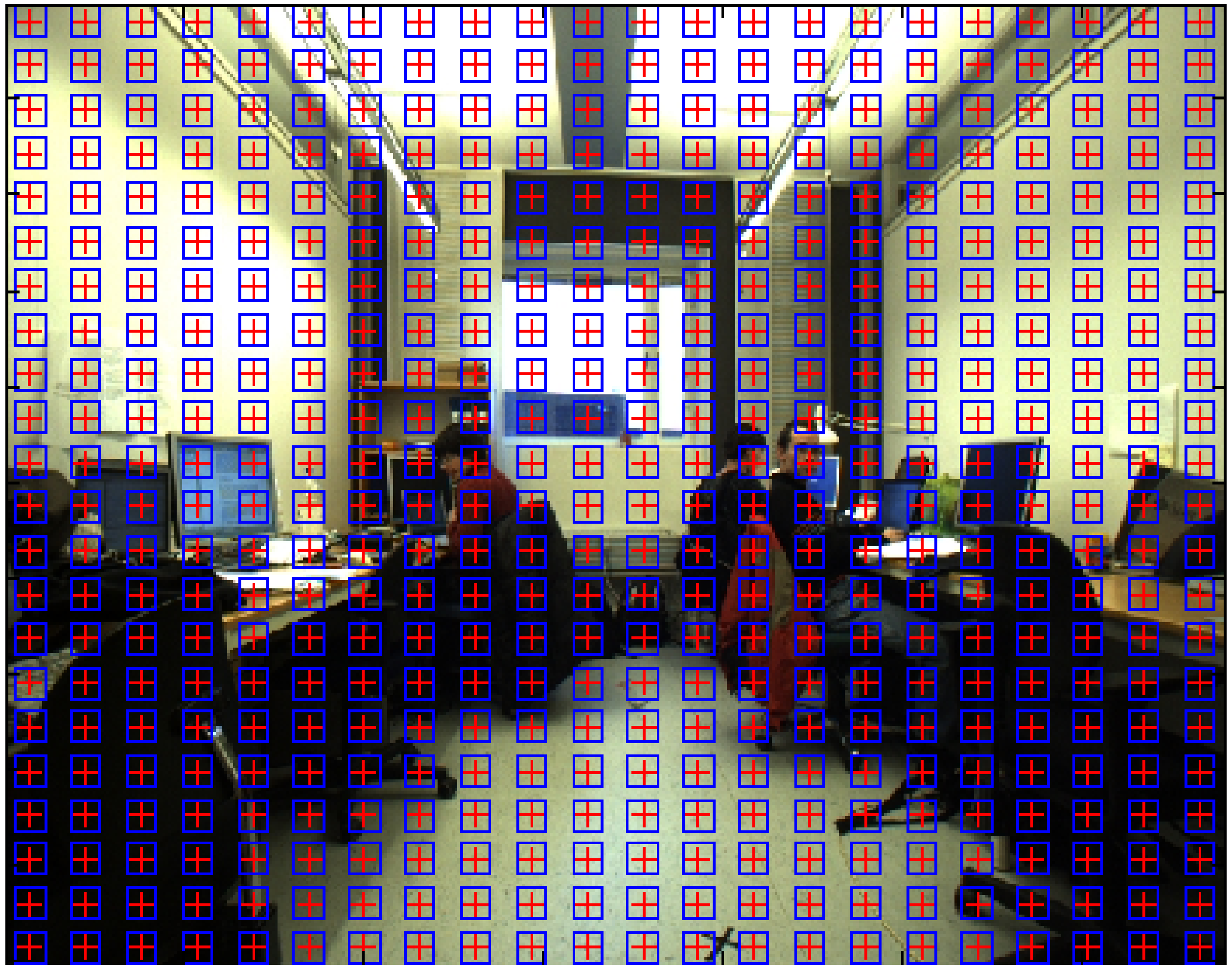
Example

Blobs B_i: circles of radius $3\sigma_i$ centered at (x_i, y_i)

Descriptors

Can be used…

- In conjonction with corresponding detector (e.g. SIFT)
- In conjonction with another detector
- Standalone at any position (e.g. on a regular grid)



Description of pixel alone

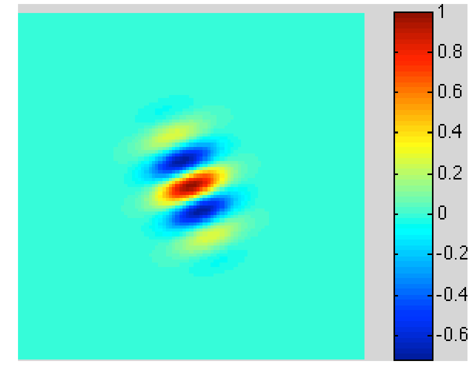
- Color, scale, main orientation…
  - Color: not robust to change of luminance
  - Norm and direction of gradient: robust to change of luminance
- However, regardless of their robustness to several transformations, they are too local to represent a discriminative information

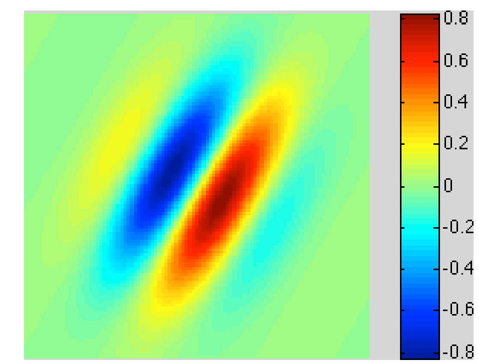
Need to account for neighborhood of keypoint

- If scale is available, size of neighborhood can be defined in accordance
- Examples
  1. Pixel neighborhood: set of pixel `features` in the neighborhood
     - `feature` can be: color, norm/direction of gradient, reponses to Gabor filters…

$$\text{Gabor} = \frac{1}{K} \exp\left(-\frac{x_\theta^2 + \gamma y_\theta^2}{2\sigma^2}\right) \cos\left(\frac{2\pi}{\lambda} x_\theta + \varphi\right)$$

where K: normalization constant,[♯1] x_θ and y_θ: rotated coordinates,[♯2] γ: eccentricity,[♯3] λ: frequency, and φ: phase





$\sigma = 3, \gamma = 1.5, \theta = 20, \lambda = 5, \varphi = 0$ | $\sigma = 3, \gamma = 0.7, \theta = 240, \lambda = 8, \varphi = 90$

⇒ Sensitivity to geometric deformations (rotation, scaling…) due to strict geometric constraint

  2. Histogram/joint histogram of pixel neighborhood `features`

⇒ No geometrical constraint

  3. Sift descriptor

⇒ Some geometrical constraint

[♯1] $\sigma = \sigma_x$, $\sigma_x = \sqrt{\gamma}\sigma_y$, $K = 1/(2/pi\sigma_x\sigma_y = \sigma^2/\sqrt{\gamma})$
[♯2] $x_\theta = x\cos\theta + y\sin\theta$ and $y_\theta = -x\sin\theta + y\cos\theta$
[♯3] $\frac{\sigma_x^2}{\sigma_y^2}$

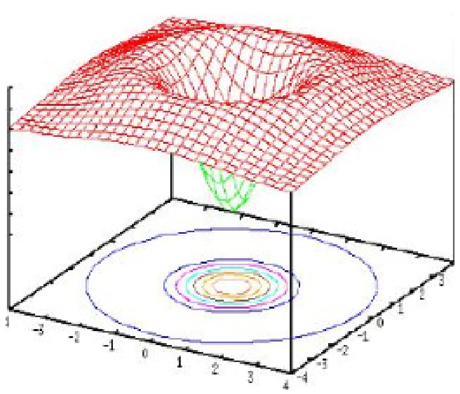
SIFT (Scale-Invariant Feature Transform)

1999

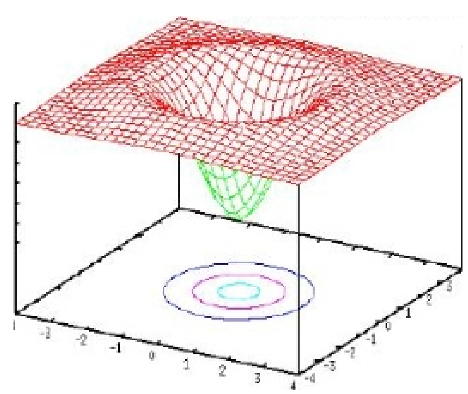
Both a detector and a local descriptor

Coarse keypoint detection

- Multi-scale representation: $L_s = G_s * I$
  - Smoothes out details < than 3 or 4 times s
  - (i.o.w. keeps details larger than that...)
- Difference of Gaussians (DoG)
  - $D_s = L_{ks} - L_s$
    - Contains only objects whose characteristic scale is between s and ks (bandpass filter)
  - DoGs for scale couples from $\{\sigma, k\sigma, k^2\sigma \ldots\}$
  - (approximation of scale-normalized Laplacian $s^2 \Delta L_s$: similarity with Lindeberg detector (see fe_10))



DoG



LoG

- Local scale-and-space extrema of D_s
  - Scale neighborhood of s: $\{k^{-1}s, s, ks\}$

Refinement

- Localization may be inaccurate
  - Especially at coarse scales
- Scale-and-space interpolation for subpixel accuracy

Discarding irrelevant keypoints

- Low-contrast keypoints: $|D_{\hat{s}}| < \text{Thr}$ (typically 0.03)
- Keypoints on edges (low localization accuracy in the direction of the edge)

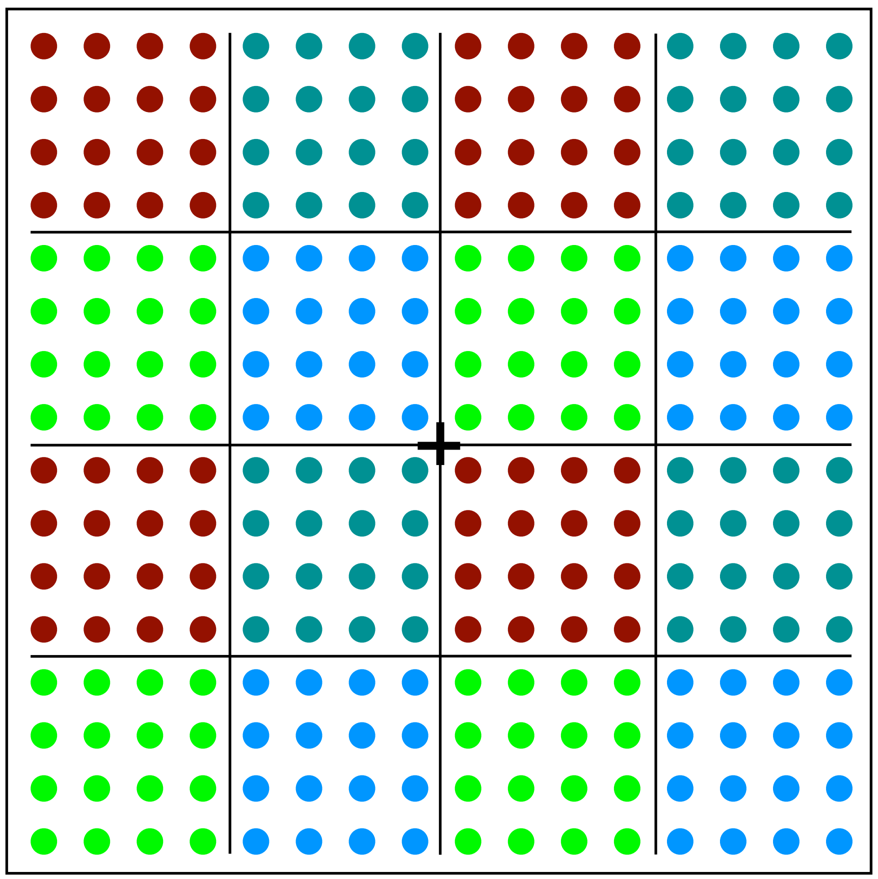
SIFT (Scale-Invariant Feature Transform) *1999*

Detected keypoints: (x_i, y_i, s_i)

Assignment of a local orientation $\rightarrow (x_i, y_i, s_i, \theta_i)$

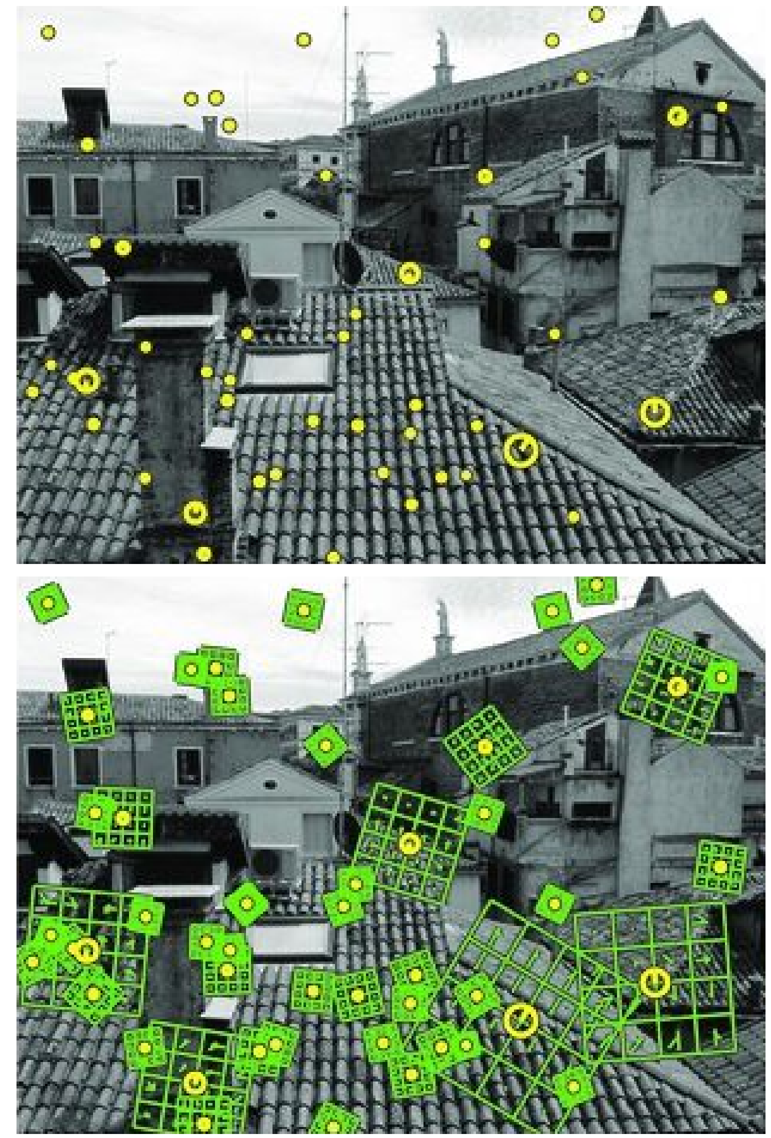
Local description of $(x_i, y_i, s_i, \theta_i)$

- Neighborhood: 16×16-grid rotated by θ_i and stretched by βs_i (typically $\beta = 3$)
- Subdivision into 16 4×4-regions



Grid before rotation

- In each subregions, computation of the histogram of gradient orientations (8 bins of 45°) weighted by a local Gaussian neighborhood (standard deviation: half the subregion side) and the gradient norms
- Concatenation of the 16 8-bin histograms: 128-dimensional SIFT descriptor

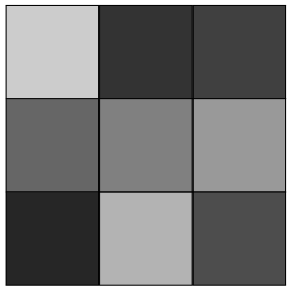


Local Binary Pattern (LBP)

A.k.a. Census Transform

Typical implementation

- Neighborhood of pixel



1	2	3
8	Ref.	4
7	6	5

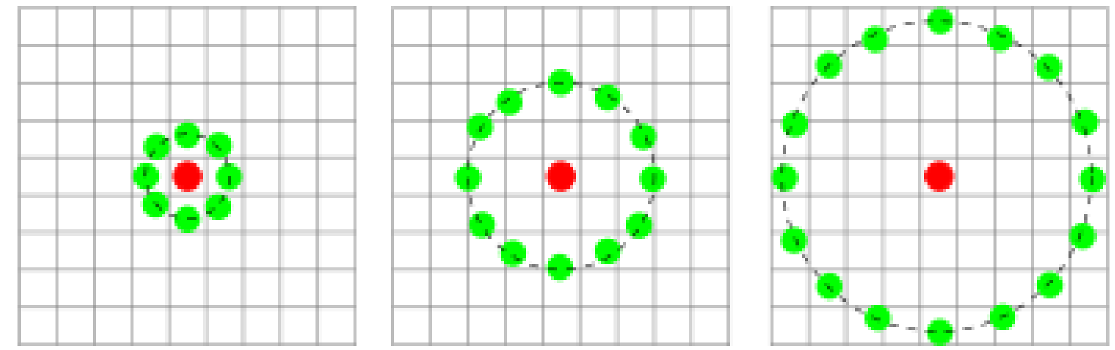
- Rule
  - Pixel p $\geq$ Ref. $\Rightarrow$ 1
  - Pixel p $<$ Ref. $\Rightarrow$ 0

1	0	0
0	Ref.	1
0	1	0

- Coding

1	2	3	4	5	6	7	8
1	0	0	0	1	0	1	0
128	64	32	16	8	4	2	1
Code = 138							

One possible generalization



Properties

- Pros
  - Robustness to variation of luminance and contrast
  - Robustness to outliers[♯1] (pixels with accidentally high or low values)
  - Encodes local spatial structure
- Cons
  - No robustness w/r/t geometrical transformations
  - Loss of information in the encoding

[♯1] Only the comparison with the central value counts so if it is above or largely above, it does not change anything

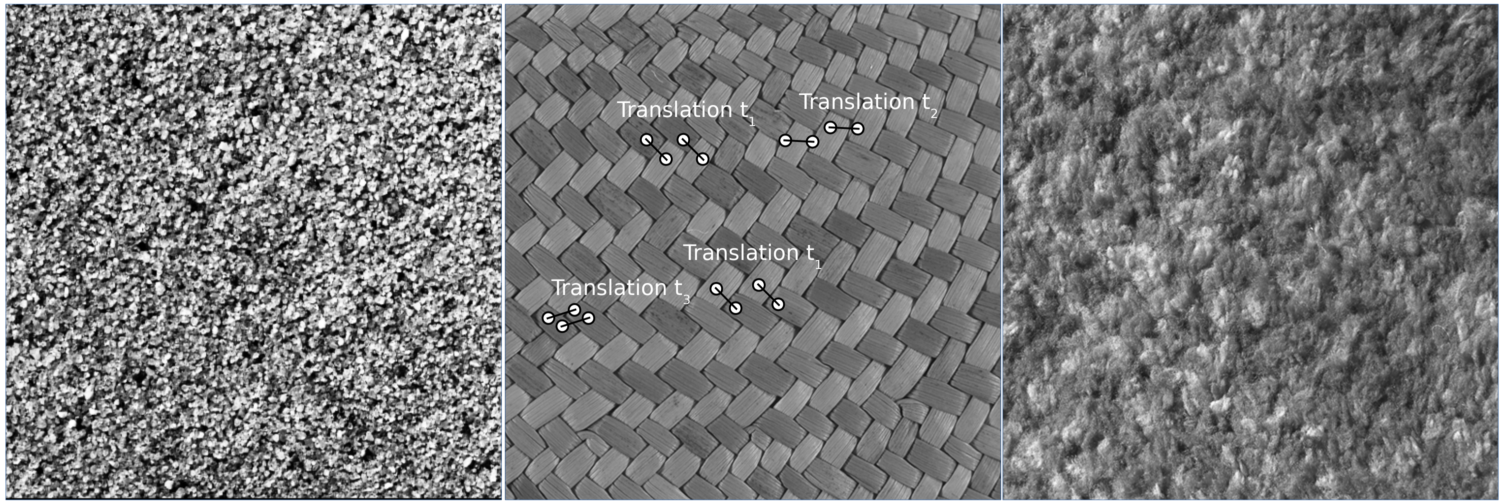
OUTLINE

Image-wide, pixel-wise[♯1] features

[♯1] As opposed to “object-wise”

Co-occurrence Matrix

Illustration



Principle

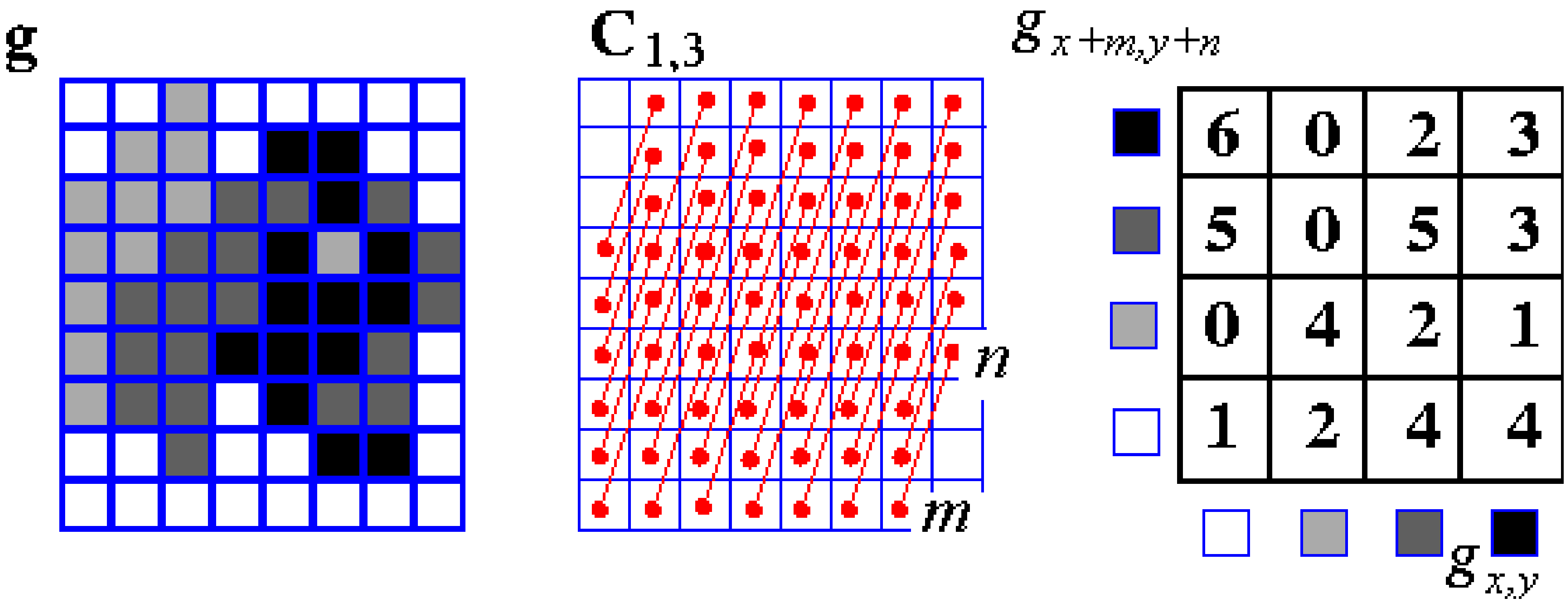
- Histogram: $h(i)$ = number of pixels with intensity i
- Co-occurrence histogram
  - Translation T (e.g. 3 pixels to the right, -1 pixel to the top)
  - $h_T(i_1, i_2)$ = number of pixel couples (p_1, p_2) with intensities i_1 and i_2 resp., and such that $p_2 = p_1 + T$
- Co-occurrence matrix
  - Gray levels $\in [0..255]$
  - 256×256-matrix = $\begin{bmatrix} h_T(0,0) & h_T(0,1) & \cdots & h_T(0,255) \\ h_T(1,0) & \ddots & \cdots & h_T(1,255) \\ \vdots & \vdots & \ddots & \vdots \\ h_T(255,0) & h_T(255,1) & \cdots & h_T(255,255) \end{bmatrix}$
- Classical acronym: GLCM=Gray Level Co-occurrence Matrix

Compute characteristics of matrix to characterize texture

Co-occurrence Matrix

Procedure illustration

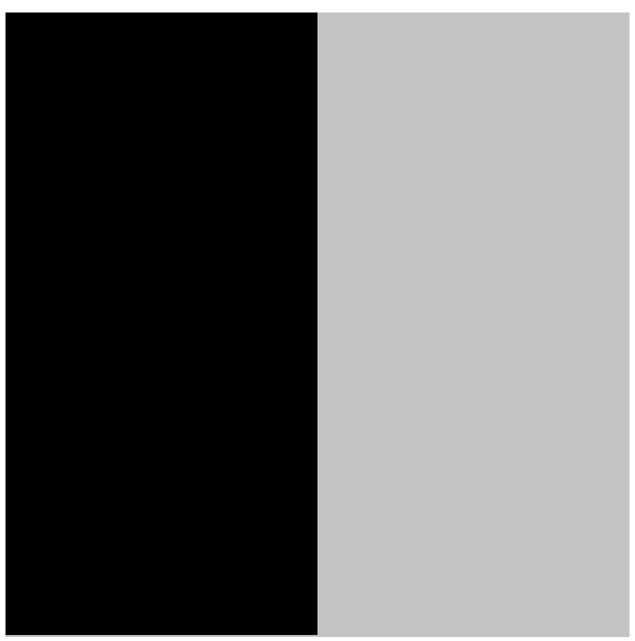
- Only 4 gray levels
- Translation $T = (1, 3)$

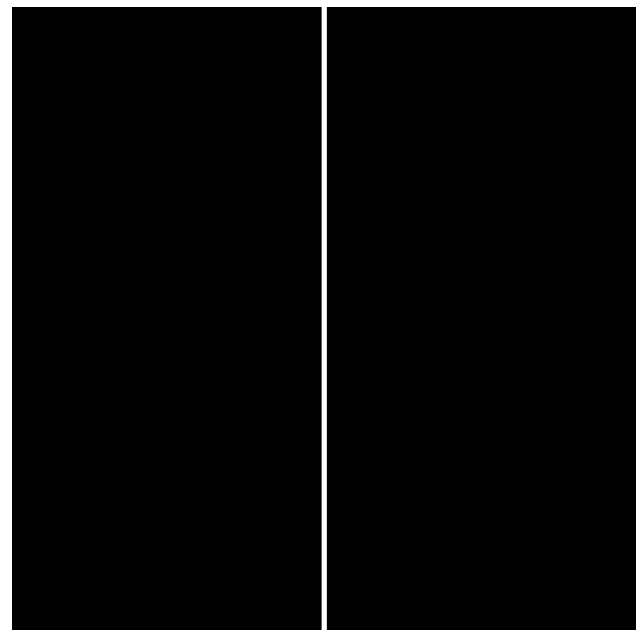


OUTLINE

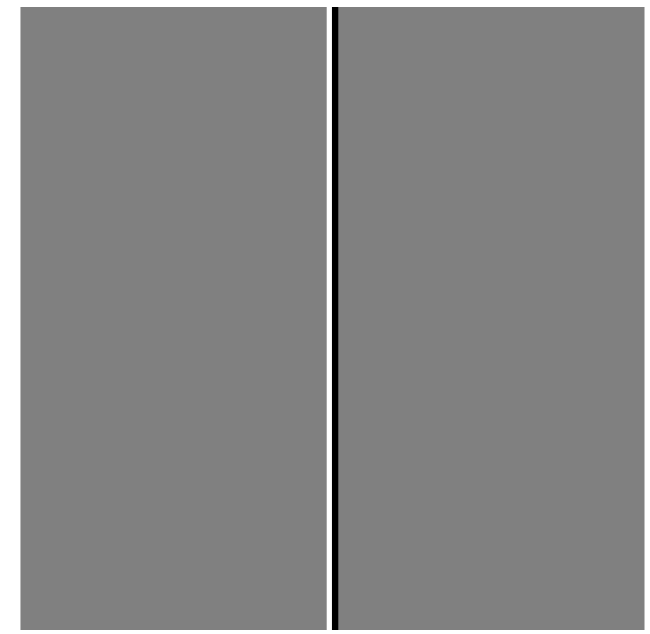
Contour detection **(note: not exactly object detection)**

Contour Detection

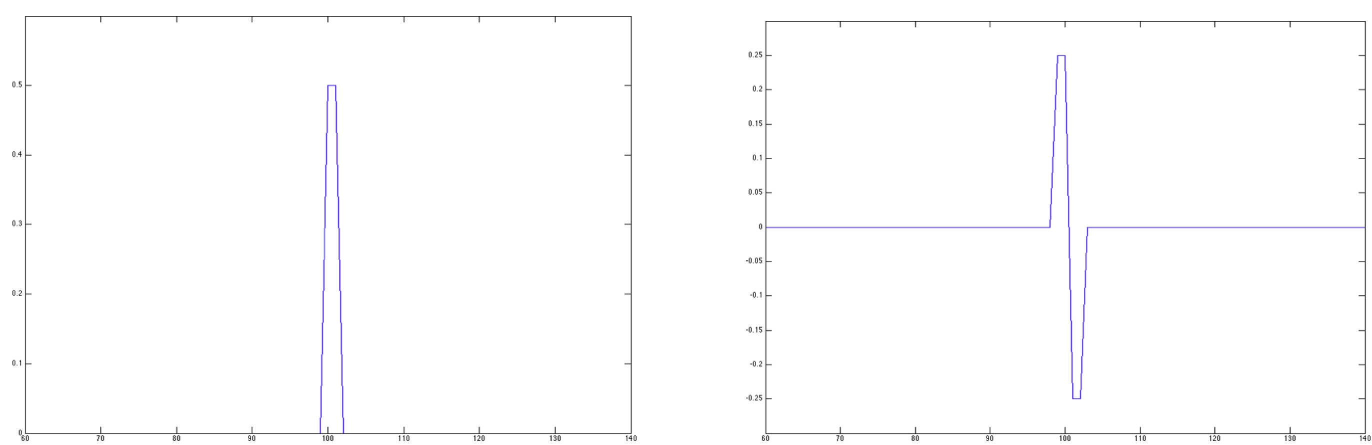




Gradient (1st order derivatives)

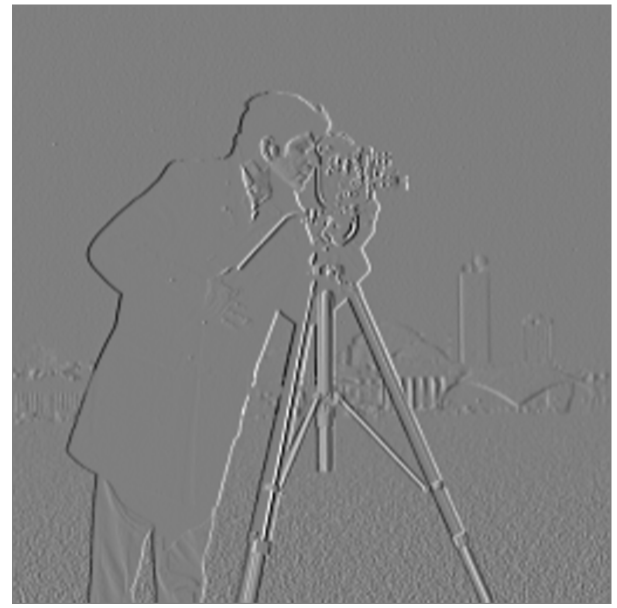


Laplacian (2nd order derivatives)

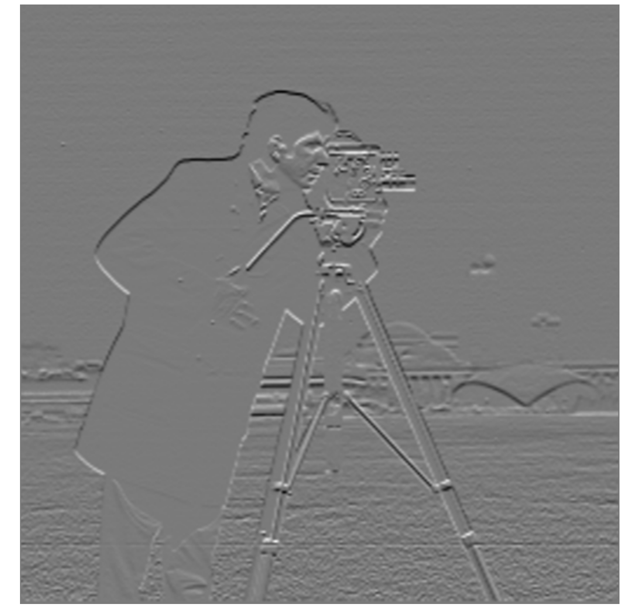


Respective profiles

Contour Detection: 1^{st} order derivatives



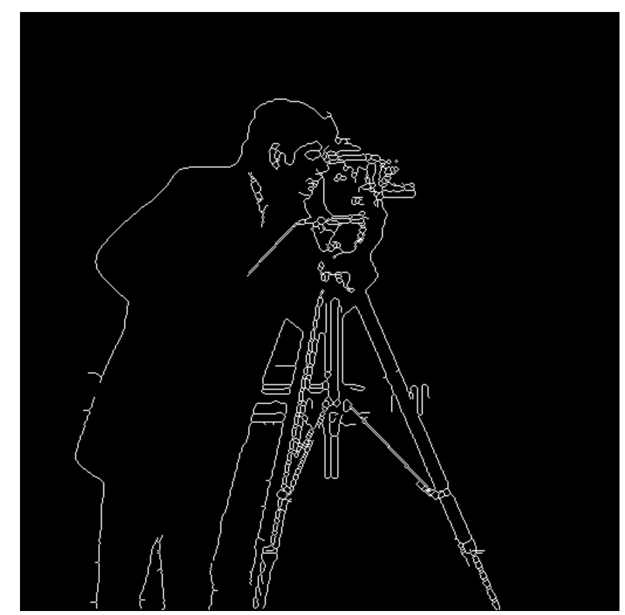
Horiz. gradient



Vert. gradient

Gradient norm

Thresholded gradient



Hysteresis thresholding

About gradient

- Horizontal gradient: smooth vertically, then compute derivative horizontally

About thresholding

- Classical thresholding with threshold τ
  - All pixels with intensity higher than τ are set to a given high value (usually 1 or 255)
  - All pixels with intensity lower than τ are set to a given low value (usually `0`)
- Hysteresis thresholding with thresholds τ_{low} and τ_{high}
  - All pixels with intensity higher than τ_{high} are set to a given high value
  - All pixels with intensity lower than τ_{low} are set to a given low value
  - A chain of pixels in-between is set to the high value if it connects to a pixel above τ_{high}, otherwise it is set to the low value

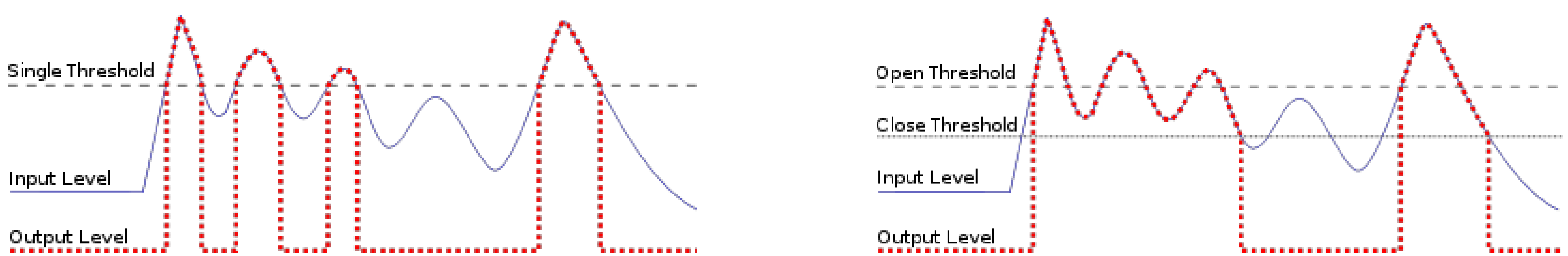
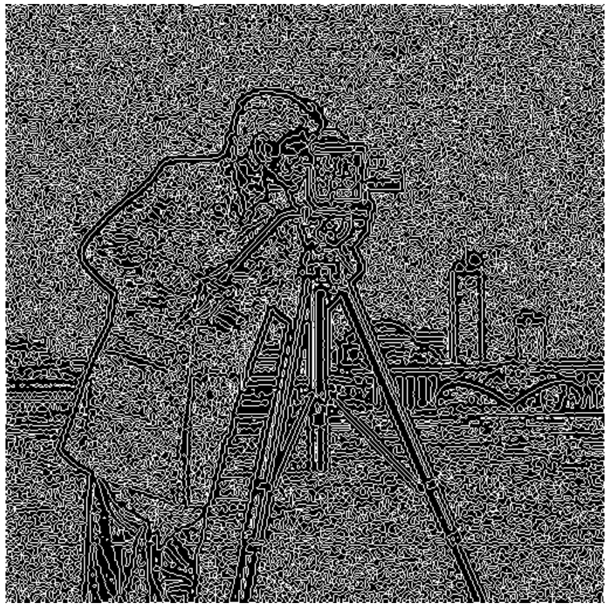


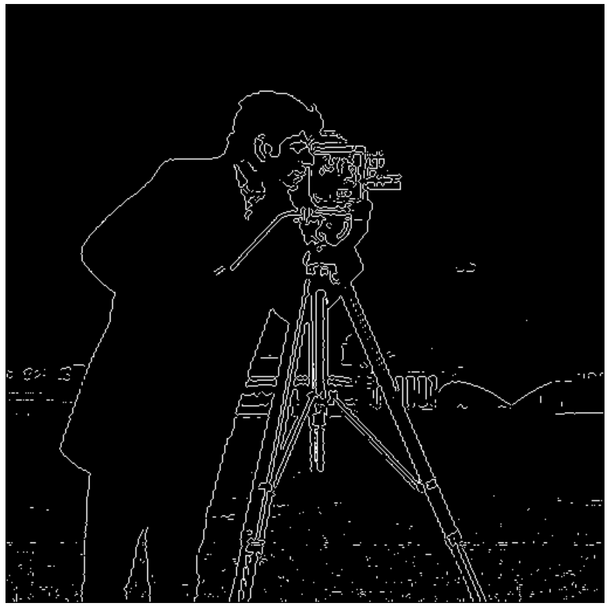
Image from Wikipedia

Contour Detection: 2^{nd} order derivatives

Laplacian



Zero-crossings



Zero-crossings where gradient is high

Active Contour

Illustration

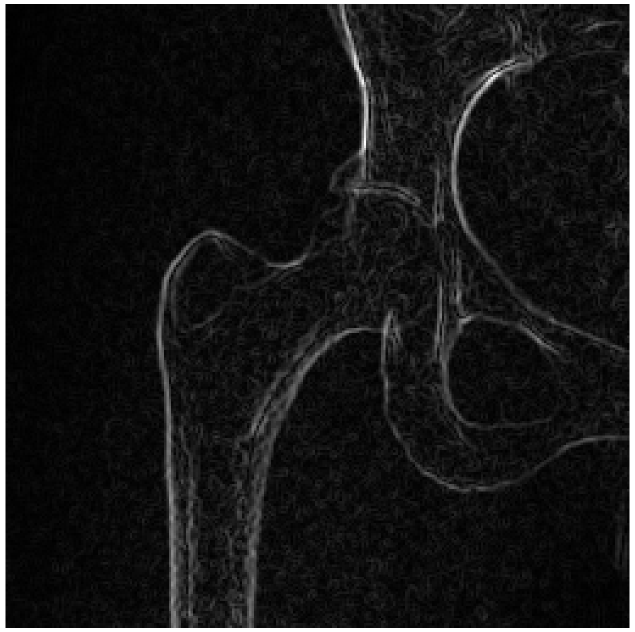
- <One contour/object>
- <Multiple initial contours>
- <X-ray image>

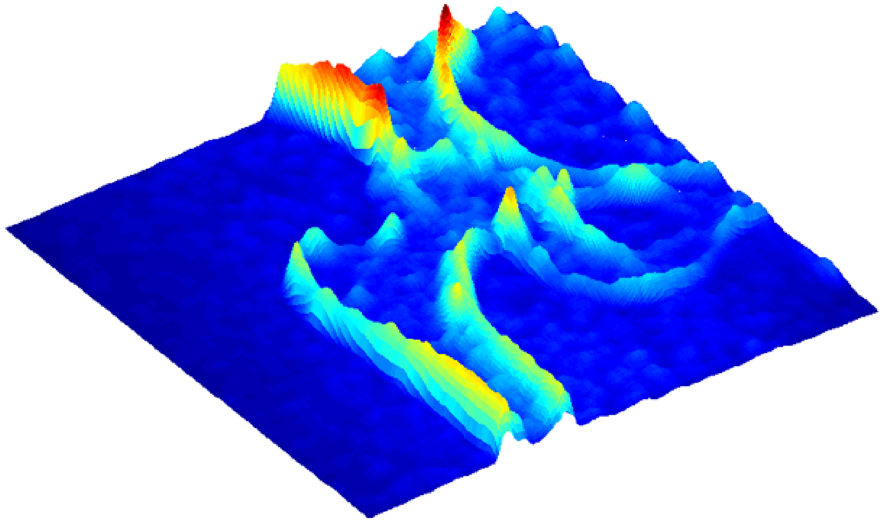
Iterative approach

1. Initial contour (typically chosen manually)
2. Compute contour energy
   - Determine small contour deformation that will decrease energy most efficiently
   - If energy cannot be decreased: active contour convergence
3. Deform contour
4. Go back to 2

Contour-based energy: uses image data at contour location

- Typically image gradient norm: if high, then low energy (and vice versa)

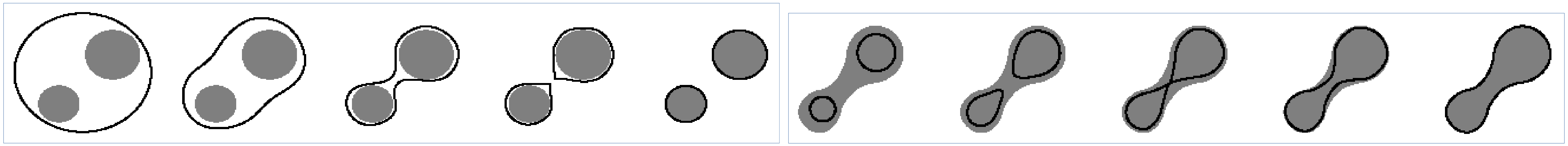




Region-based energy: uses image data inside and/or outside contour

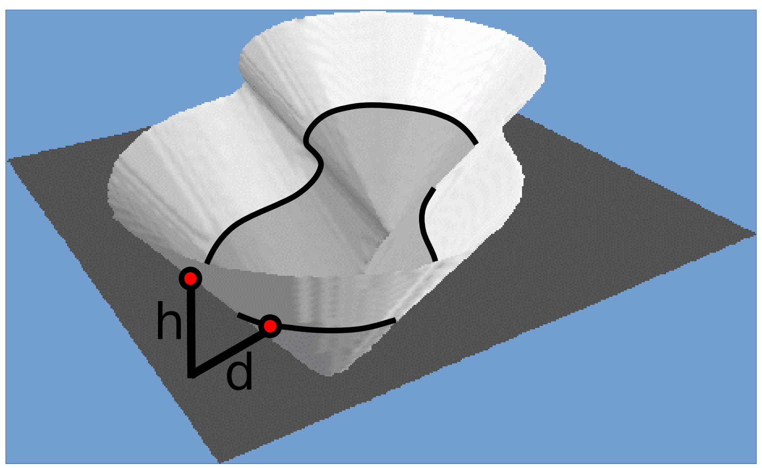
- Typically homogeneity inside, or inside/outside contrast: if high, then low energy (and vice versa)
  <One contour/object>

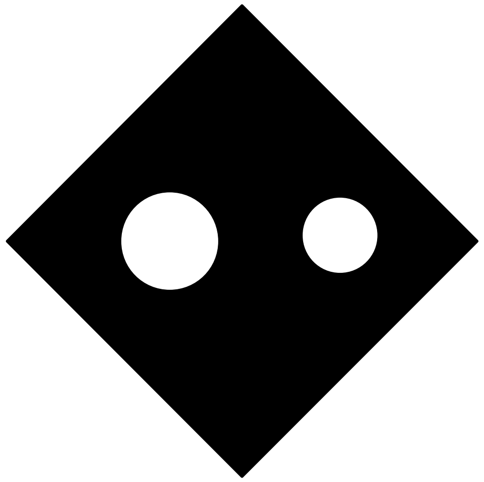
Problems

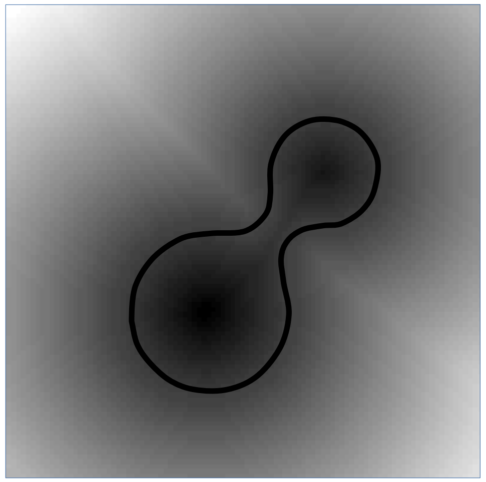


Solution

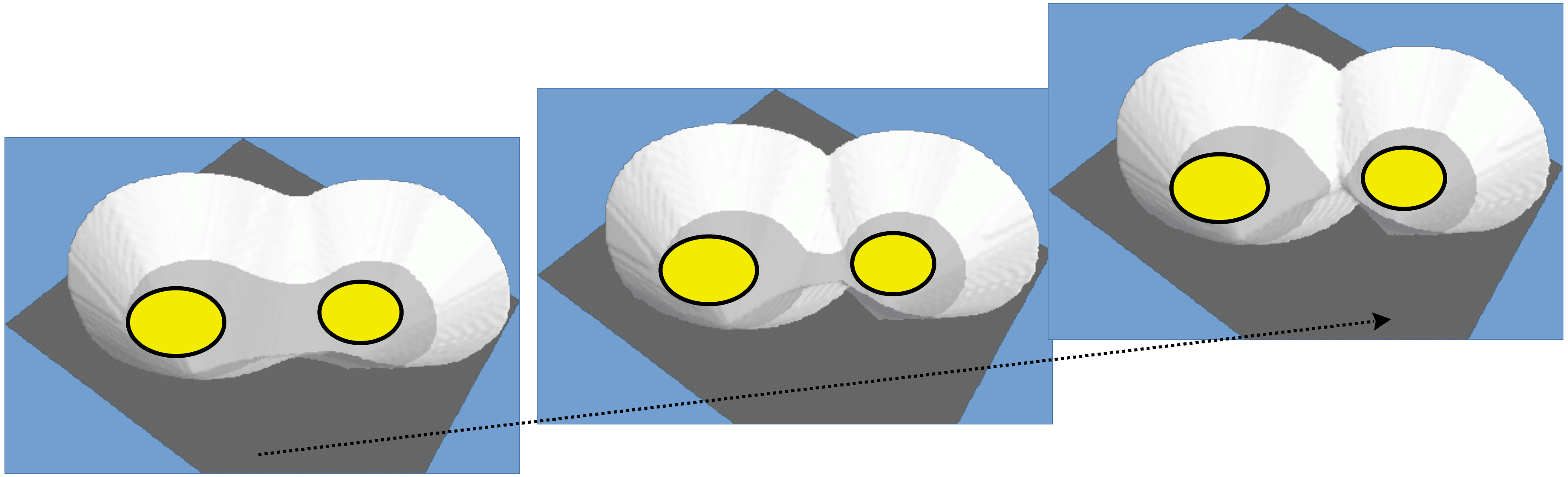
- Explanation







- Evolution

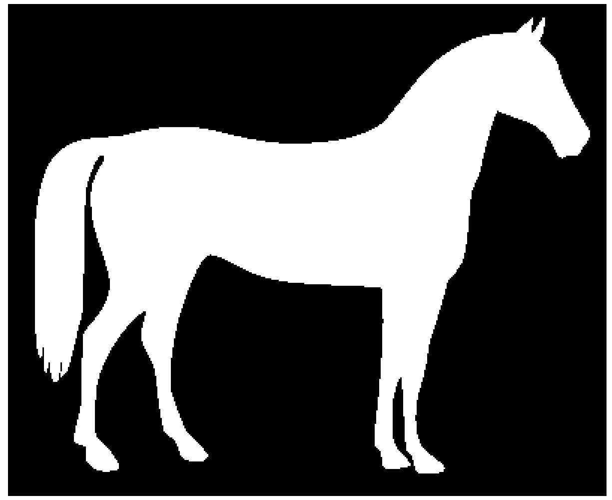
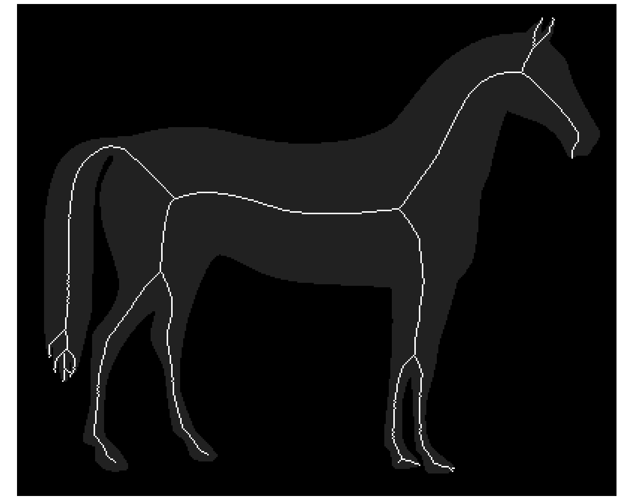


Object skeleton

OUTLINE

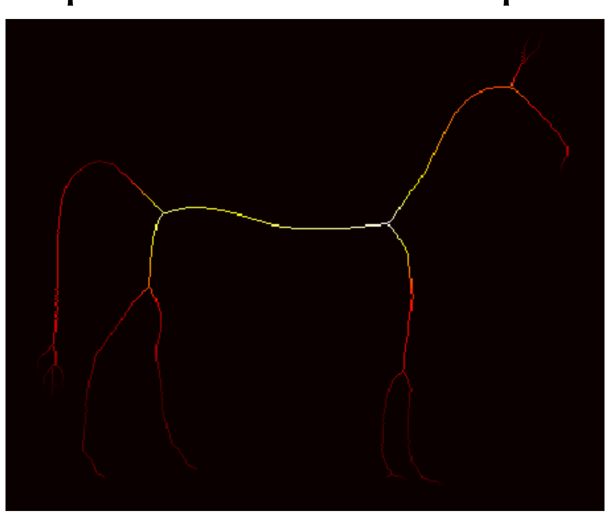
Object Skeleton

Skeleton of object mask

Interpretation

- Geometrical and topological information about object
  - Topology (graph): vertices, edges, cycles, but no geometrical considerations <Mug-Torus>
- Representation of shape: skeleton + distance to object boundary



Object Skeleton

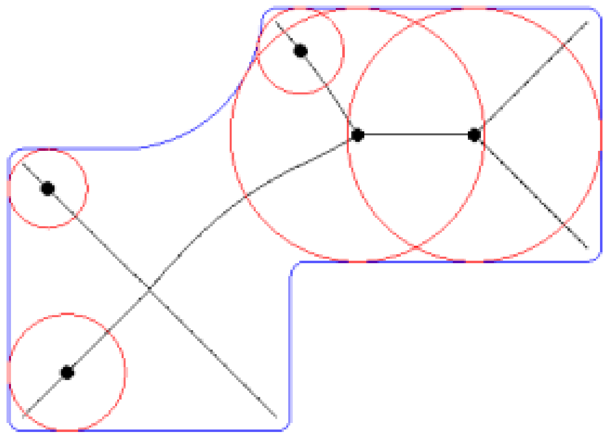
Fire propagation simultaneously started on boundary: where 2 or more wavefront meet

Set of locations equidistant to at least 2 points on object boundary

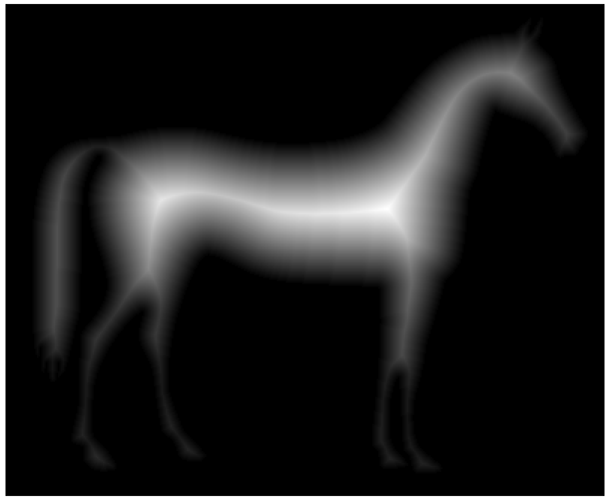
$\Leftrightarrow$ Set of centers of disks tangent to object boundary at least 2 points

= Medial axis

Set of centers of maximal disks

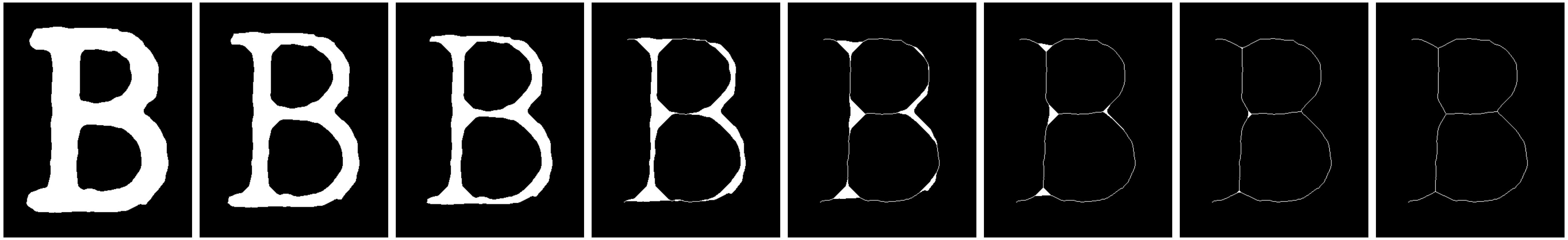


Crest lines of distance function

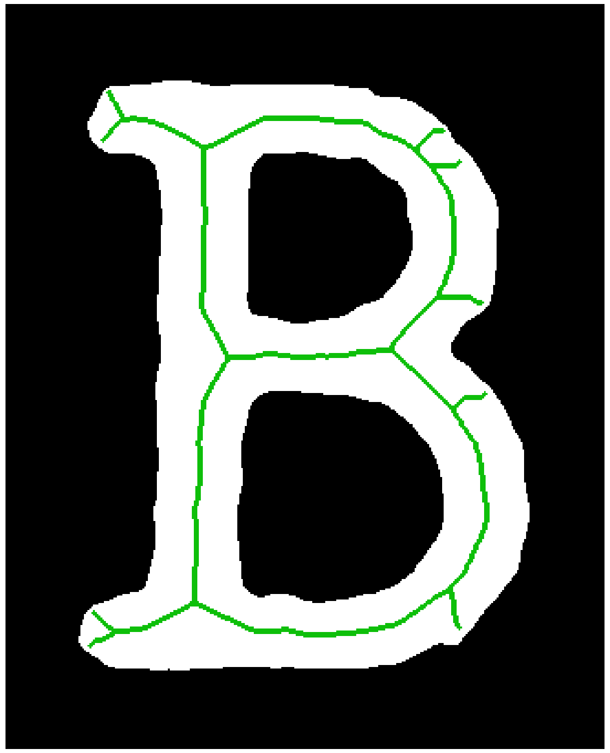


Object Skeleton

E.g. morphological thinning ("successive erosions without changing topology")



Followed by pruning to remove spurious branches

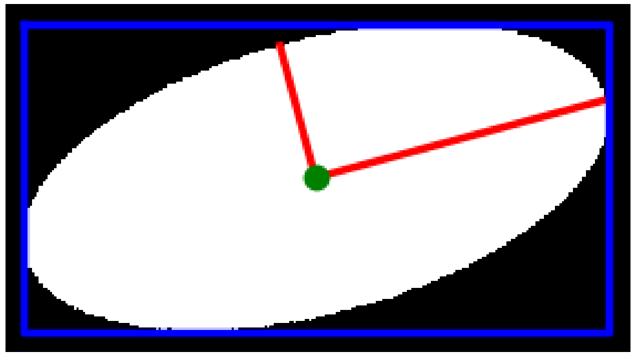


OUTLINE

Object-wise measures

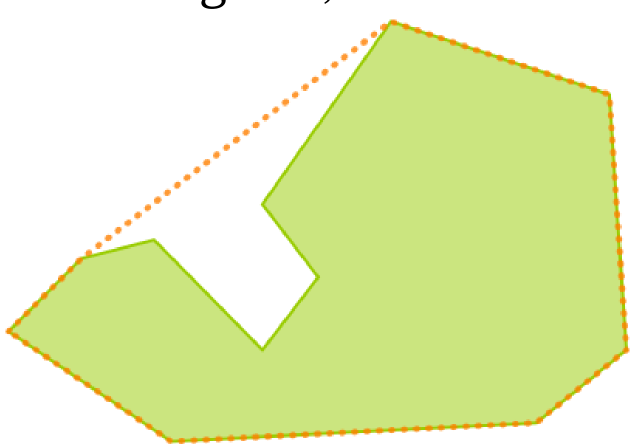
Features on Objects

Previous step: object detection



Typical measures

- Bounding box, convex hull



- Main orientation

- Eccentricity of fitting ellipse[♯1]
- Perimeter
- Area
  - Area / bbox area
  - Area / convex hull area (solidity)
- Moments
  - W/o centering
    $$m_{ij} = \sum_{(x,y) \in W} I(x,y) \times x^i \times y^j$$
  - Central moments
    $$m_{ij} = \sum_{(x,y) \in W} I(x,y) \times (x - x_c)^i \times (y - y_c)^j$$

[♯1] Eccentricity: how ellipse deviates from being a circle; $\in [0, 1[$; 0=circle