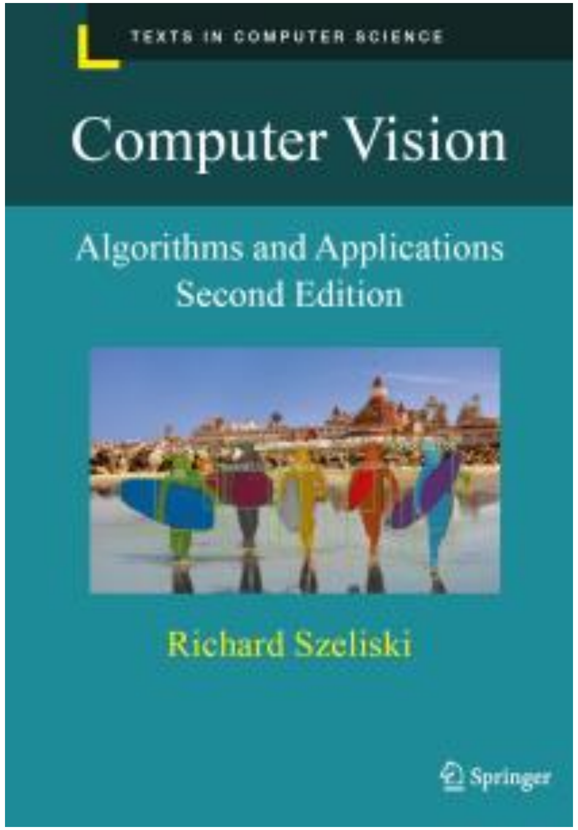
Computer Vision

Feature invariance

Important information



Textbook

Rick Szeliski, *Computer Vision: Algorithms and Applications* online at: http://szeliski.org/Book/

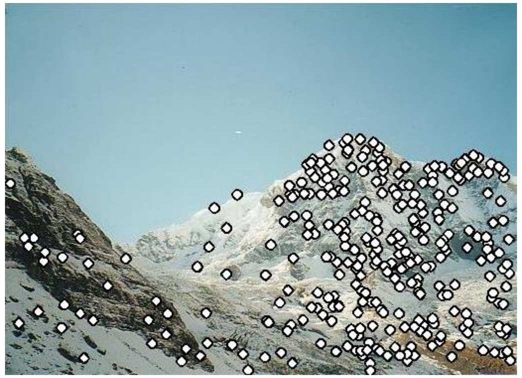
Many of the slides in this course are modified from the excellent class notes of similar courses offered in other schools by Noah Snavely, Prof Yung-Yu Chuang, Fredo Durand, Alyosha Efros, Bill Freeman, James Hays, Svetlana Lazebnik, Andrej Karpathy, Fei-Fei Li, Srinivasa Narasimhan, Silvio Savarese, Steve Seitz, Richard Szeliski, and Li Zhang. The instructor is extremely thankful to the researchers for making their notes available online. Please feel free to use and modify any of the slides, but acknowledge the original sources where appropriate.
All readings are from Richard Szeliski, Computer Vision: Algorithms and Applications, 2nd Edition, unless otherwise noted.

Reading

- Szeliski (2nd edition): 7.1

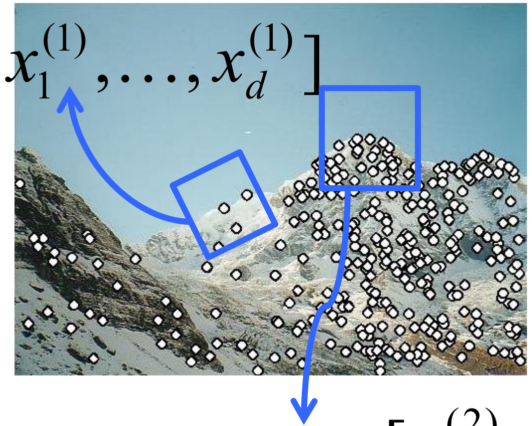
Local features: main components

1) **Detection**: Identify the interest points



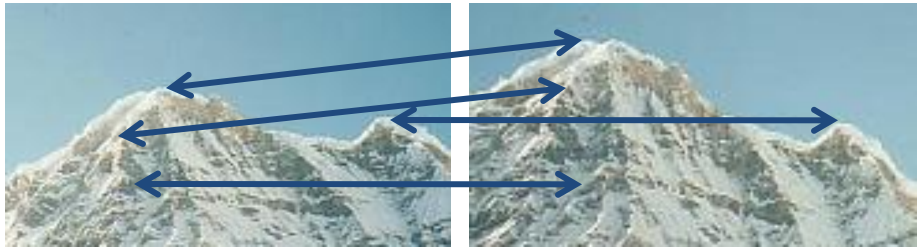
2) **Description**: Extract vector feature descriptor surrounding each interest point.

$$\mathbf{x}_1 = [x_1^{(1)}, \ldots, x_d^{(1)}]$$



$$\mathbf{x}_2 = [x_1^{(2)}, \ldots, x_d^{(2)}]$$

3) **Matching**: Determine correspondence between descriptors in two views



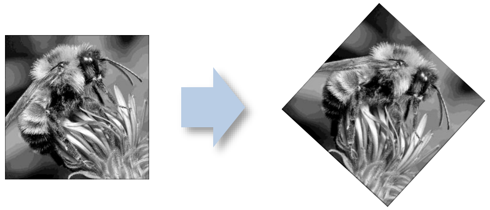
Kristen Grauman

Harris features (in red)

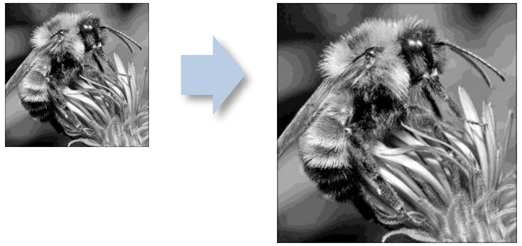
Image transformations

- Geometric

Rotation

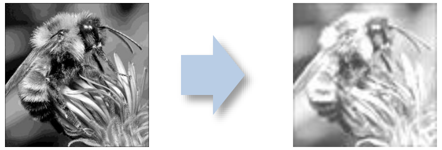


Scale



- Photometric

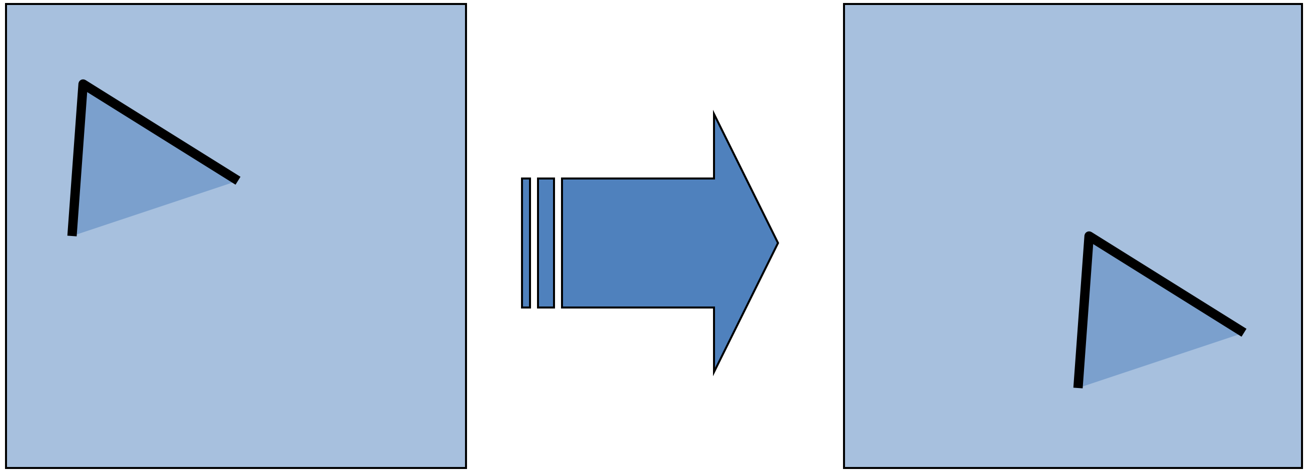
Intensity change



Invariance and equivariance

- We want corner locations to be *invariant* to photometric transformations and *equivariant* to geometric transformations
  - **Invariance:** image is transformed and corner locations do not change
  - **Equivariance:** if we have two transformed versions of the same image, features should be detected in corresponding locations
  - (Sometimes “invariant” and “equivariant” are both referred to as “invariant”)
  - (Sometimes “equivariant” is called “covariant”)

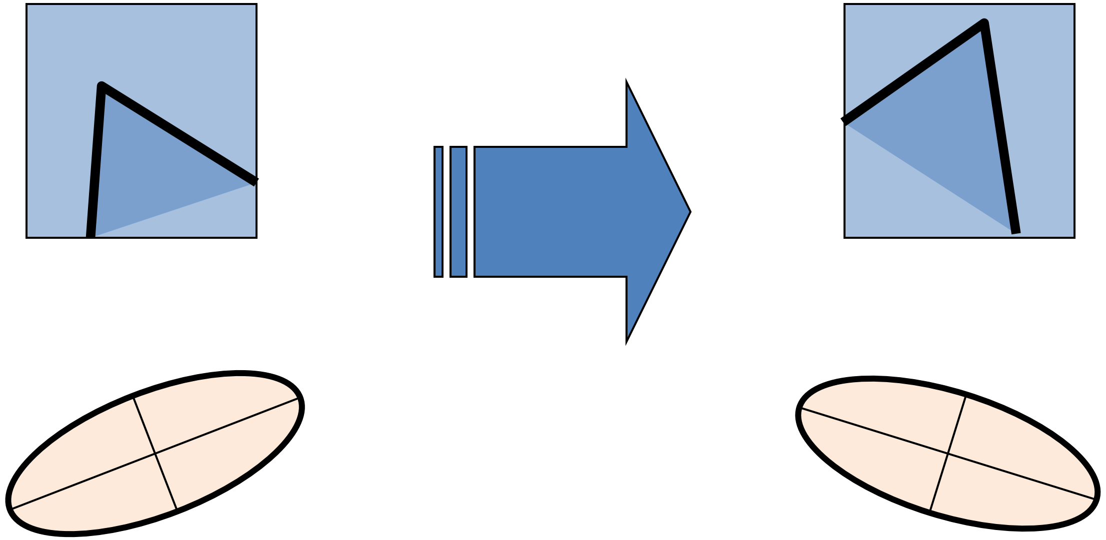
Harris detector invariance properties: image translation



- Derivatives and window function are equivariant

Corner location is equivariant w.r.t. translation

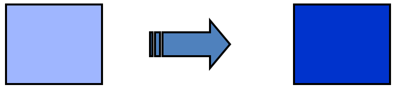
Harris detector invariance properties: image rotation



Second moment ellipse rotates but its shape (i.e. eigenvalues) remains the same

Corner location is equivariant w.r.t. image rotation

Harris detector invariance properties: Affine intensity change

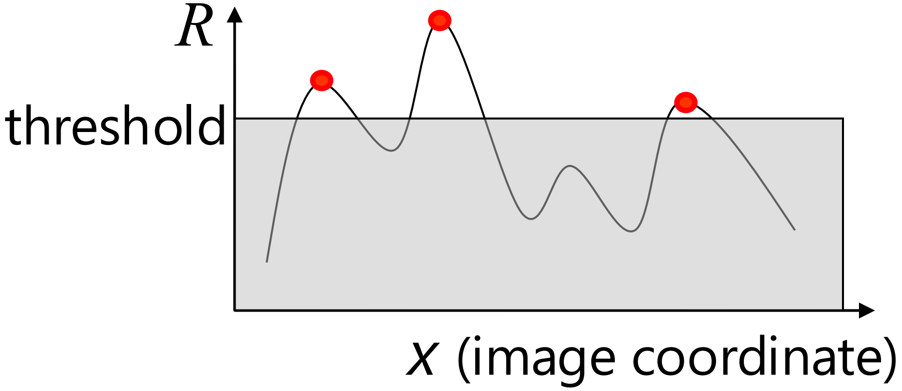


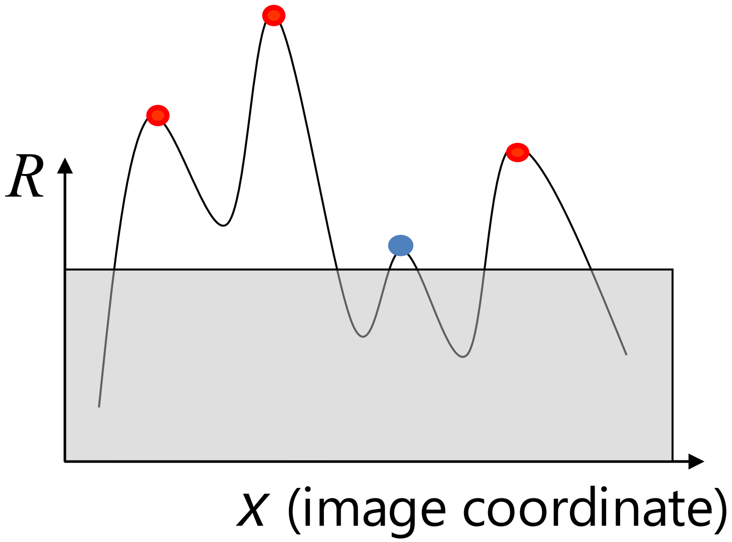
$I \rightarrow a\,I + b$

- Only derivatives are used to compute Harris scores

→ *invariance* to intensity shift $I \rightarrow I + b$

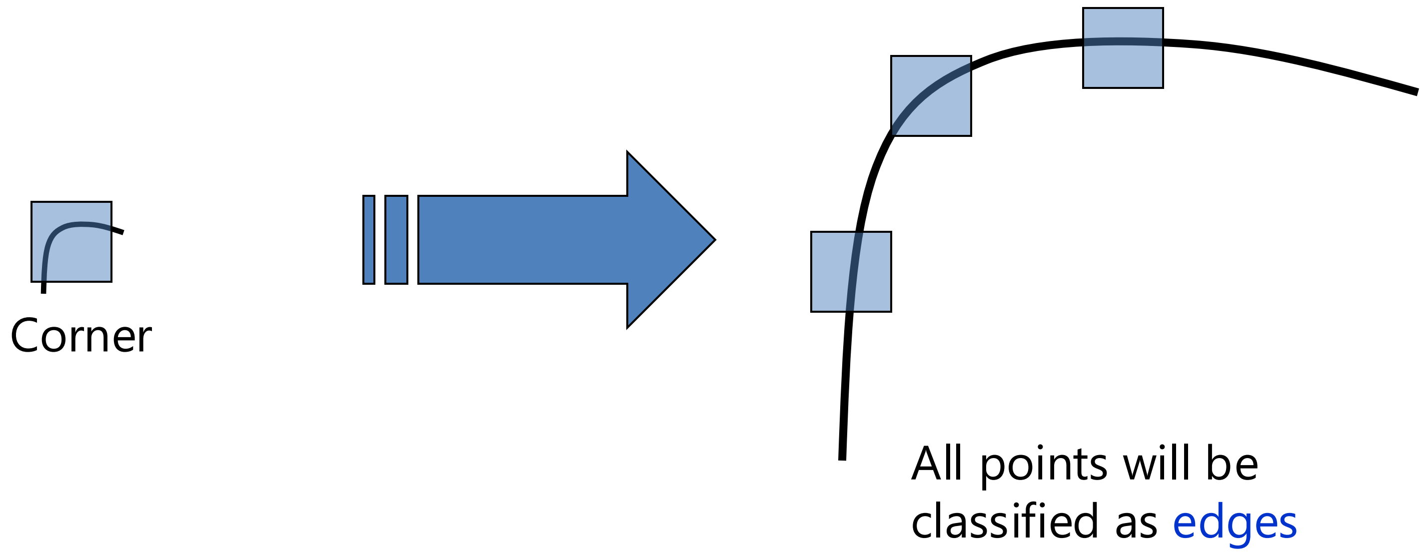
- Intensity scaling: $I \rightarrow a\,I$





Partially invariant to affine intensity change

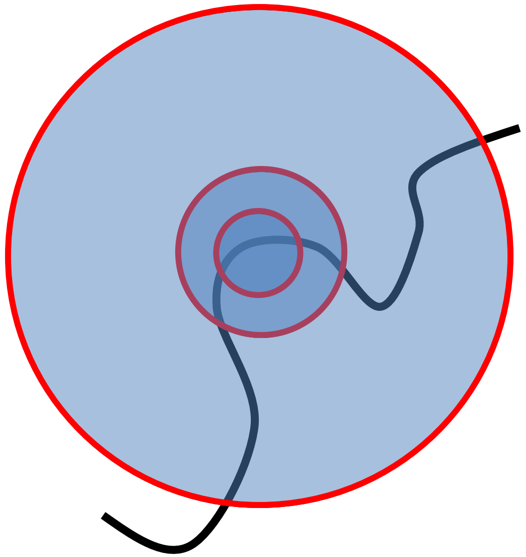
Harris detector invariance properties: scaling



Neither invariant nor equivariant to scaling

Scale invariant detection

Suppose you're looking for corners

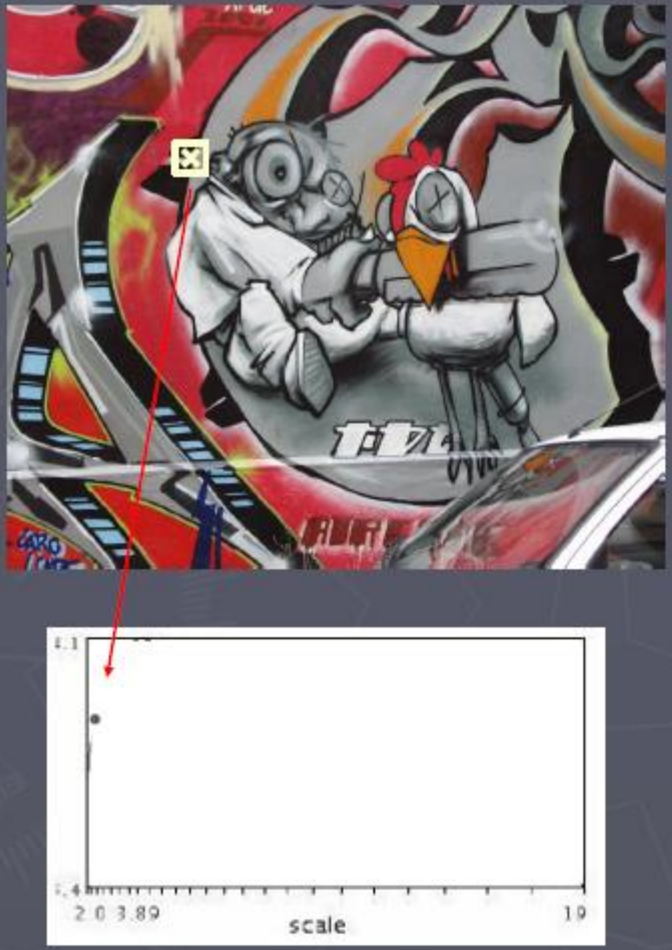


Key idea: find scale that gives local maximum of *f*

- in both position and scale
- One definition of *f*: the Harris operator

Automatic scale selection

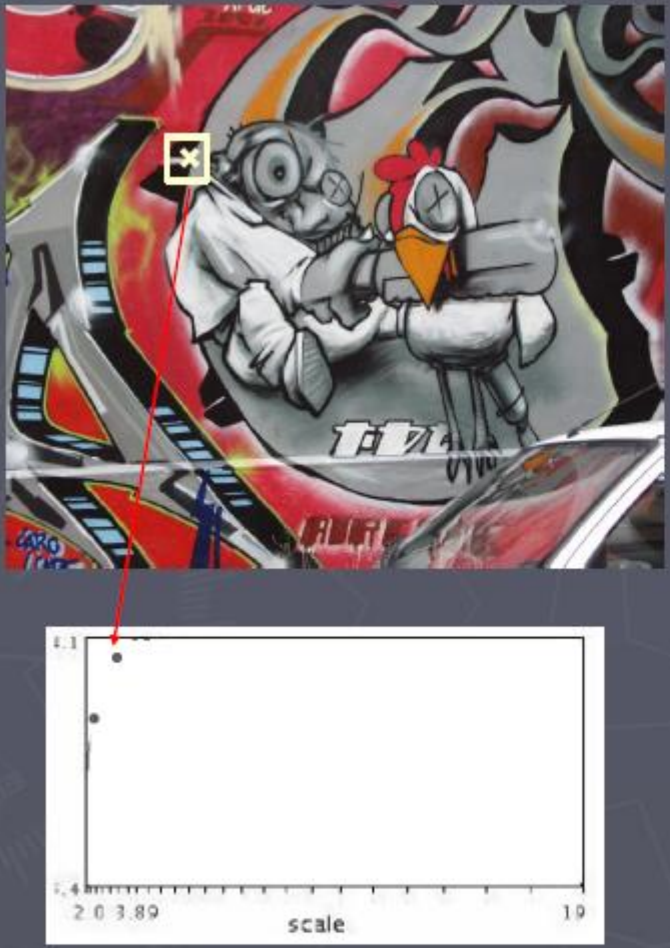
Lindeberg et al., 1996



$f(I_{i_1 \dots i_m}(x,\sigma))$

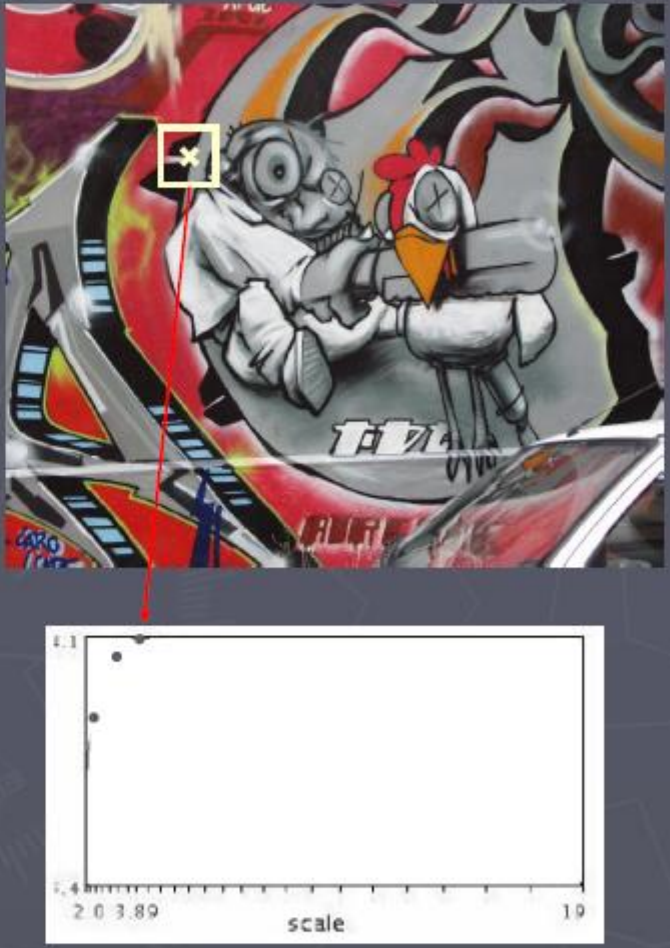
Slide from Tinne Tuytelaars

Automatic scale selection



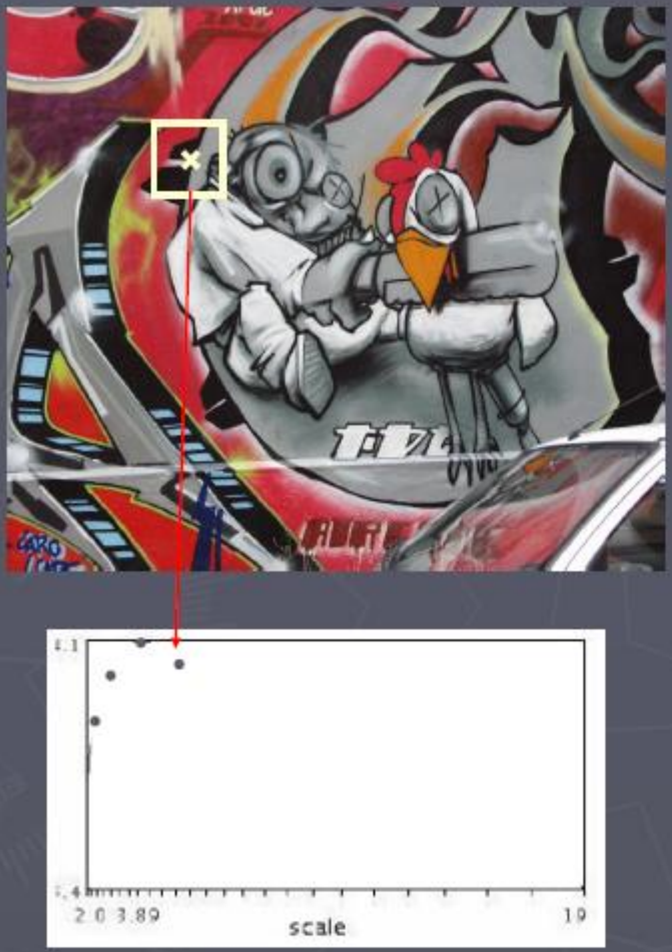
$f(I_{i_1 \ldots i_m}(x,\sigma))$

Automatic scale selection



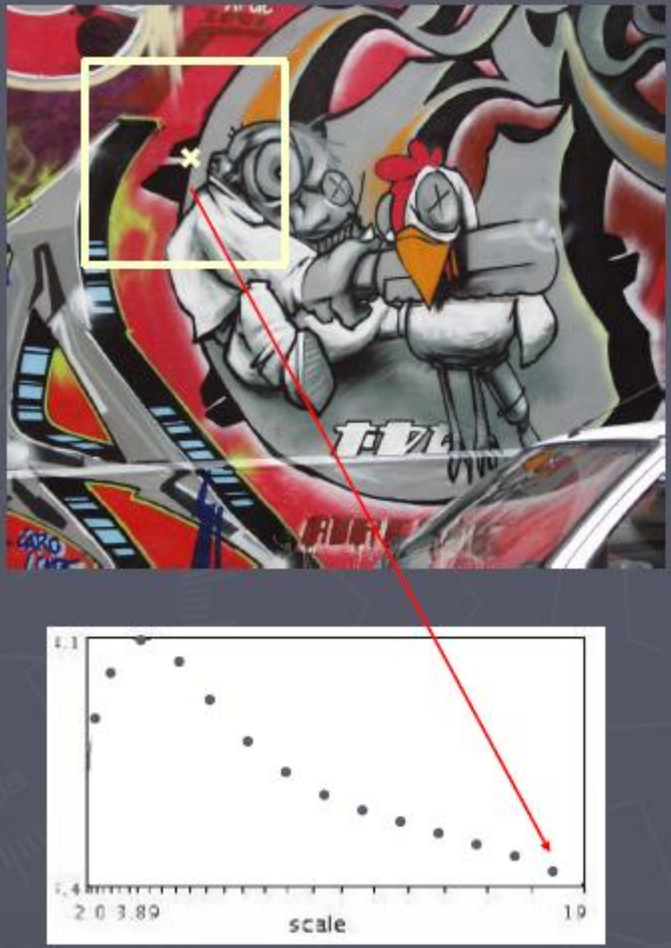
$f(I_{i_1 \ldots i_m}(x,\sigma))$

Automatic scale selection



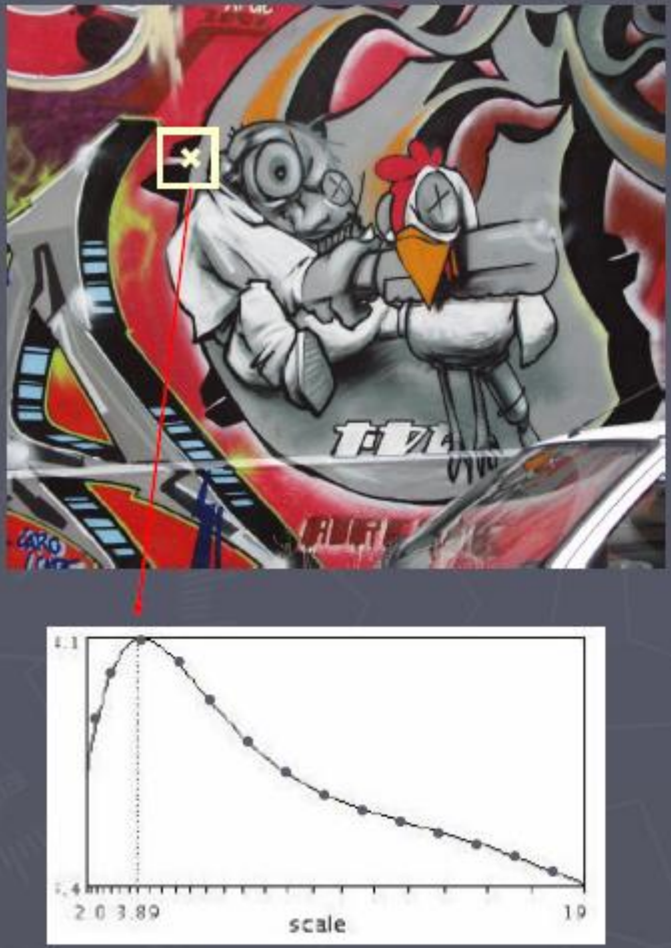
$f(I_{i_1 \ldots i_m}(x,\sigma))$

Automatic scale selection



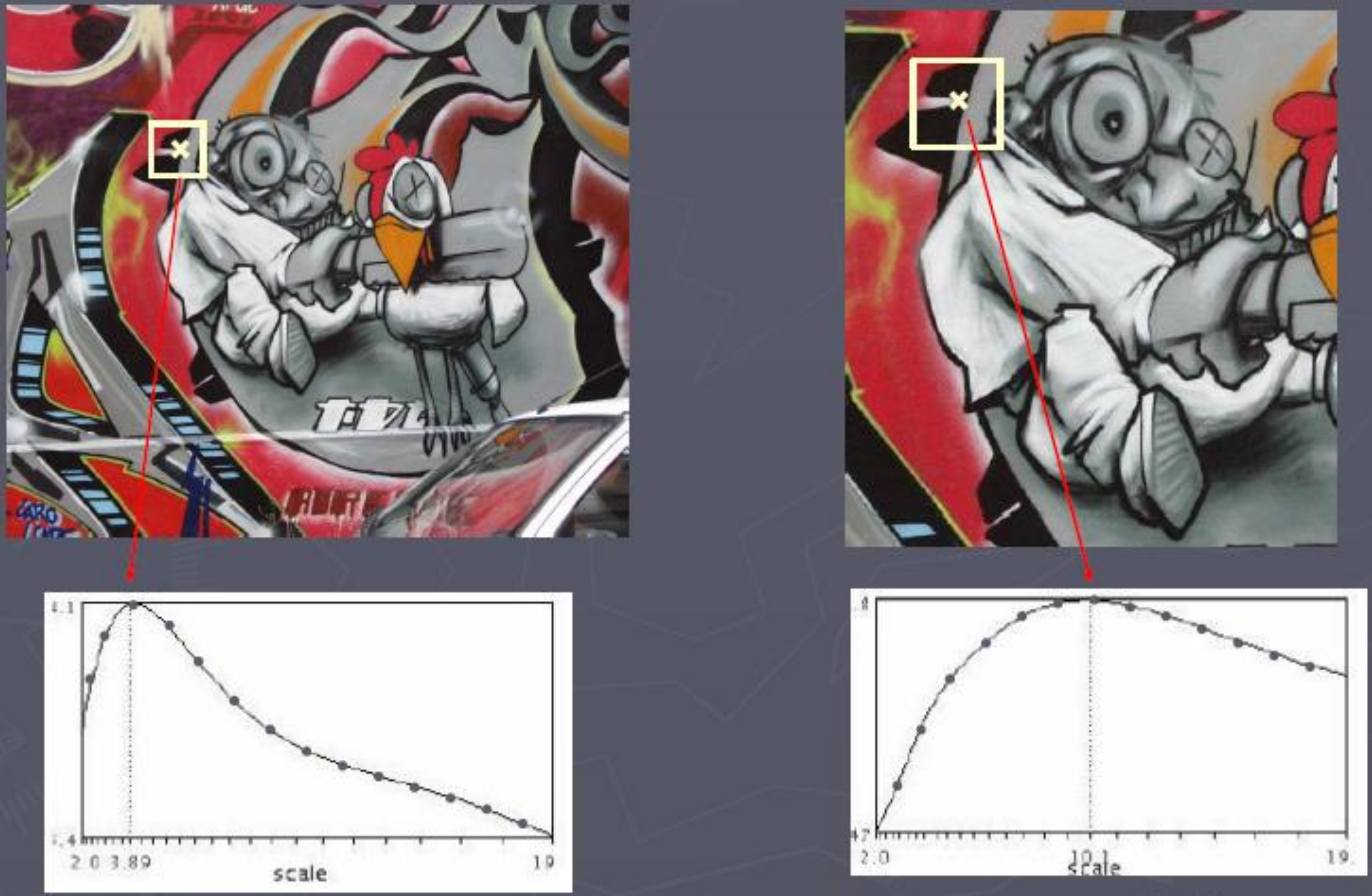
$f(I_{i_1 \ldots i_m}(x,\sigma))$

Automatic scale selection



$$f(I_{i_1 \ldots i_m}(x,\sigma))$$

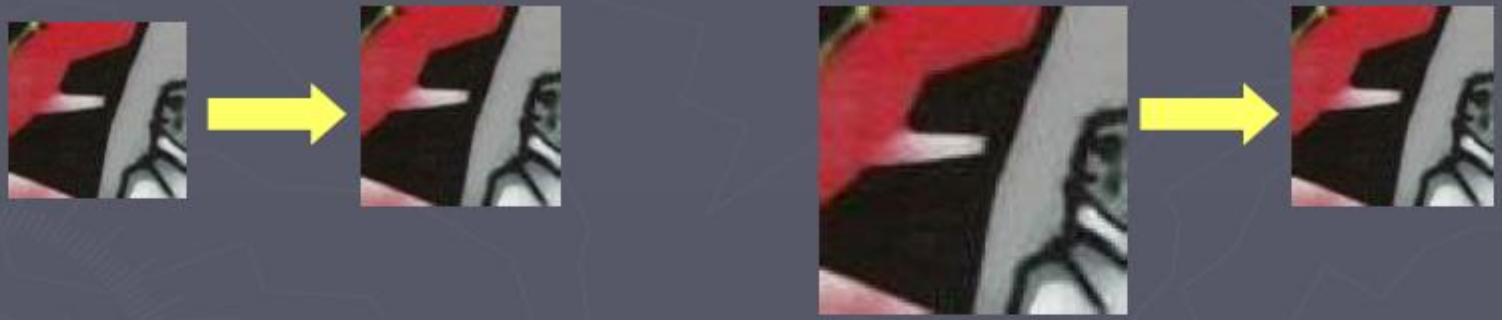
Automatic scale selection



$f(I_{i_1 \ldots i_m}(x, \sigma))$ $\quad$ $f(I_{i_1 \ldots i_m}(x', \sigma'))$

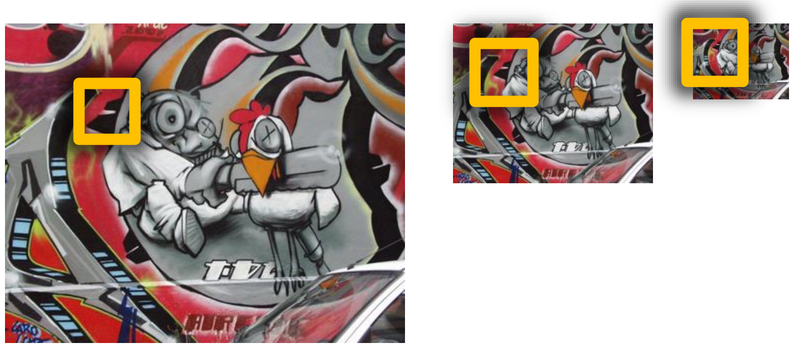
Automatic scale selection

Normalize: rescale to fixed size



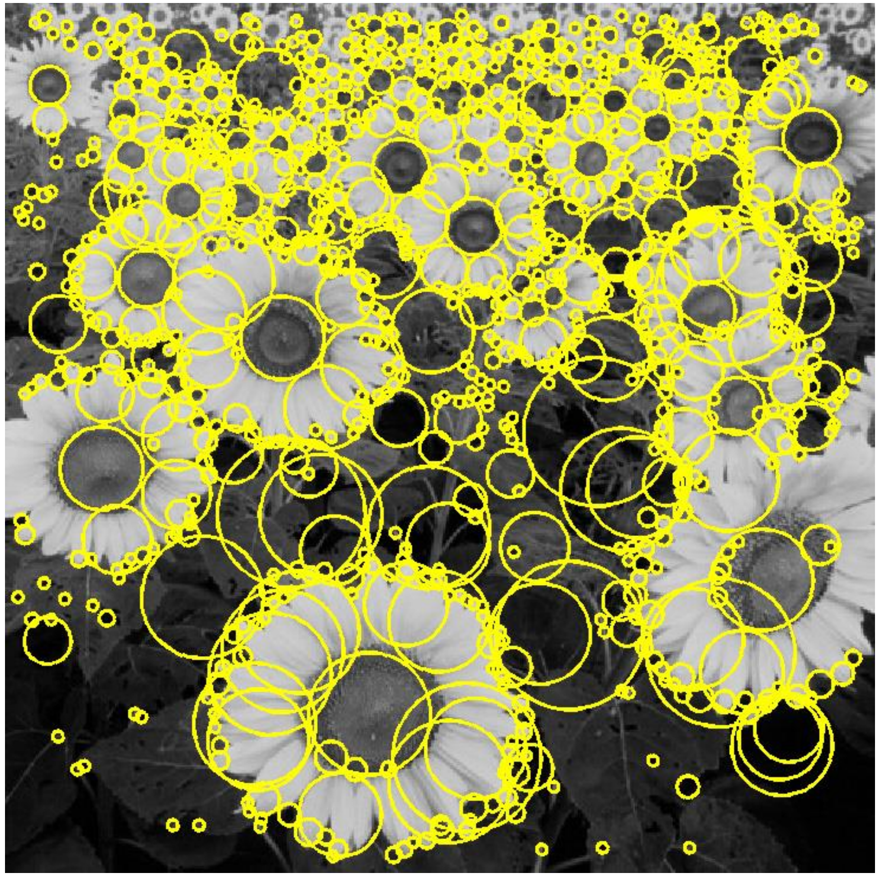
Implementation

- Instead of computing *f* for larger and larger windows, we can implement using a fixed window size with a Gaussian pyramid



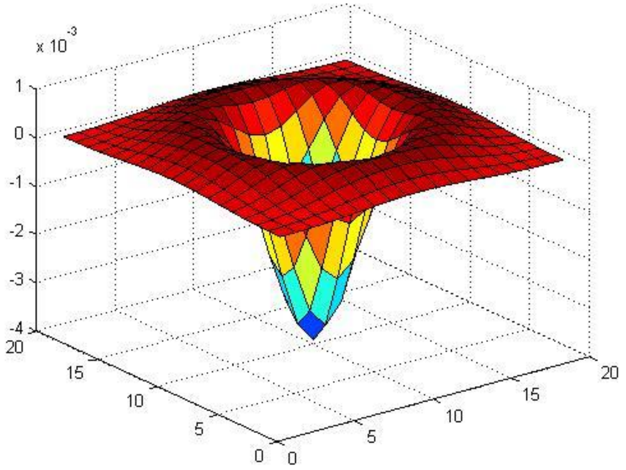
(sometimes need to create in-between levels, e.g. a ¾-size image)

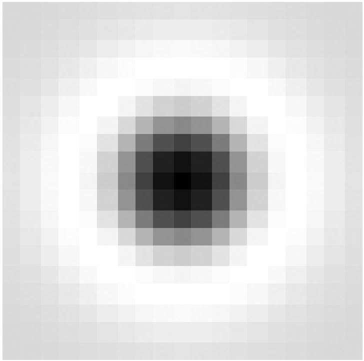
Feature extraction: Corners and blobs



Another common definition of *f*

- The *Laplacian of Gaussian* (*LoG*)



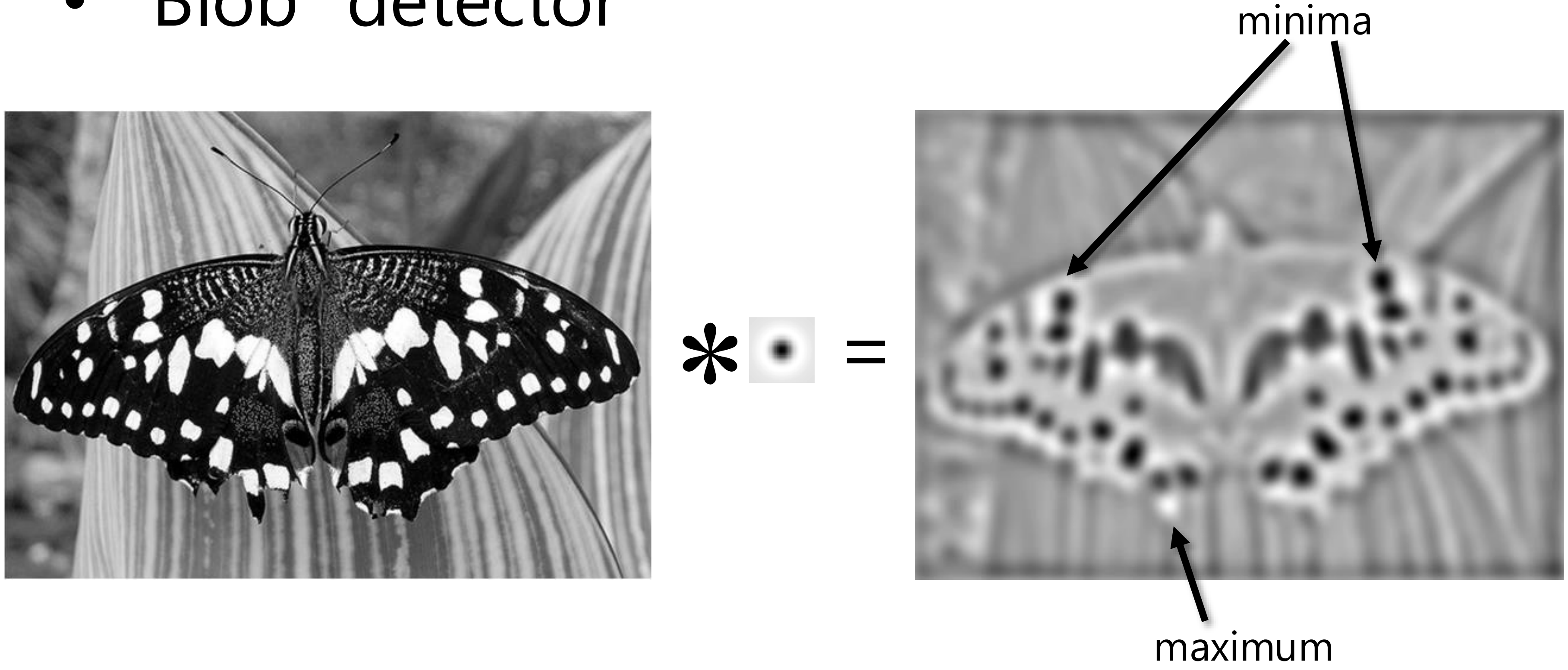


$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

(very similar to a Difference of Gaussians (DoG) – i.e. a Gaussian minus a slightly smaller Gaussian)

Laplacian of Gaussian

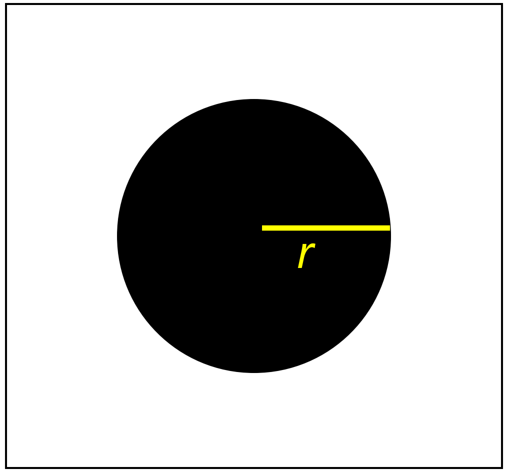
- “Blob” detector



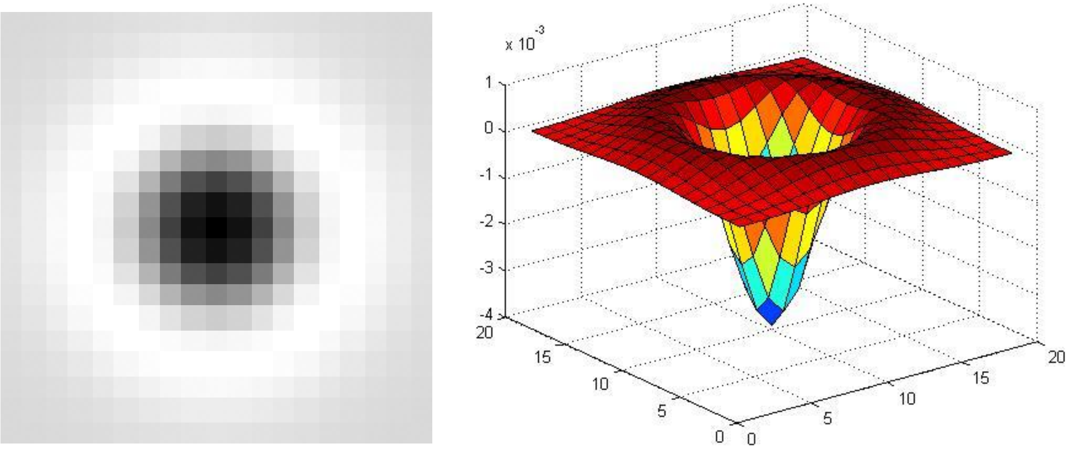
- Find maxima *and minima* of LoG operator in space and scale

Scale selection

- At what scale does the Laplacian achieve a maximum response for a binary circle of radius r?



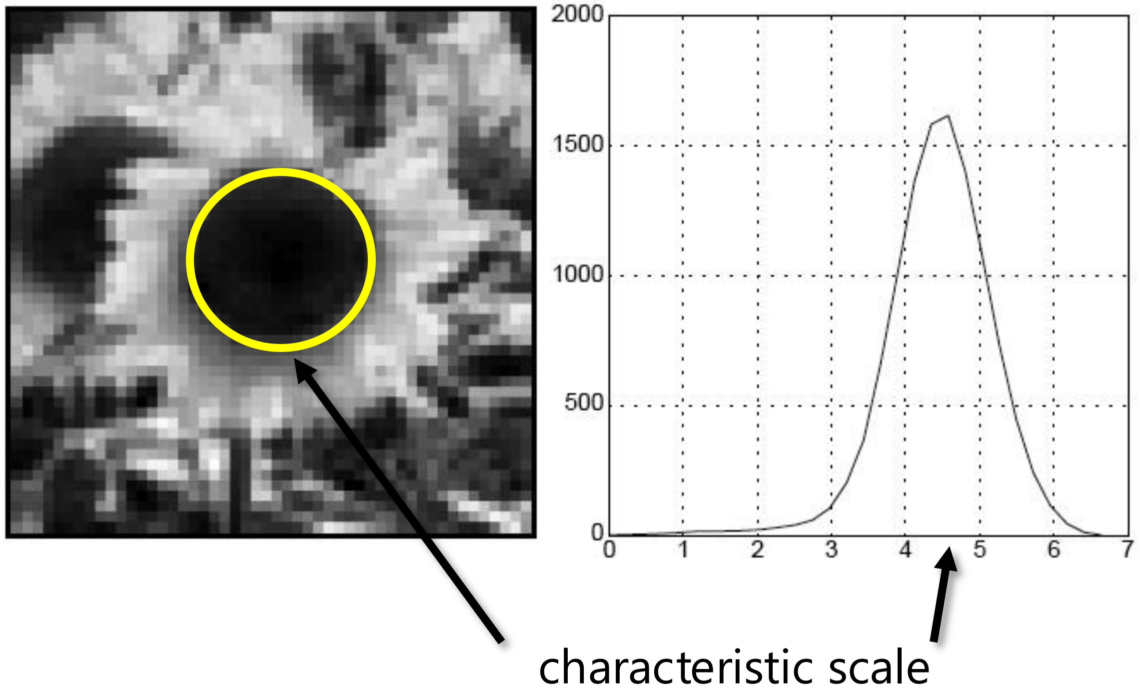
image



Laplacian

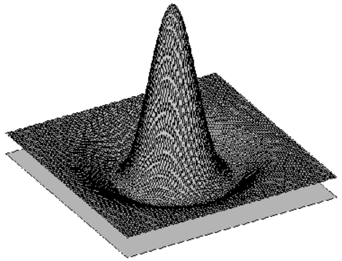
Characteristic scale

- We define the characteristic scale as the scale that produces peak of Laplacian response



T. Lindeberg (1998). ["Feature detection with automatic scale selection."](#) *International Journal of Computer Vision* **30** (2): pp 77--116.

Find local maxima in 3D position-scale space



$$L_{xx}(\sigma) + L_{yy}(\sigma)$$

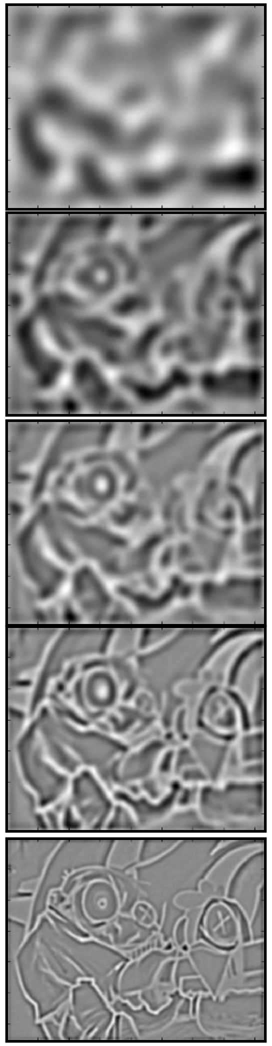
σ^5

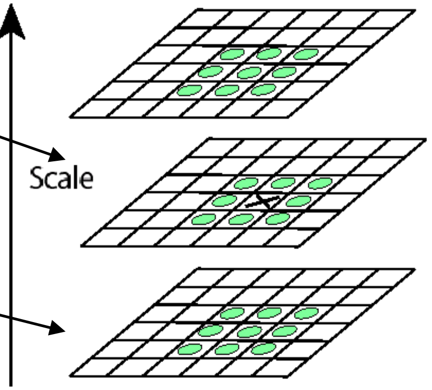
σ^4

σ^3

σ^2

σ





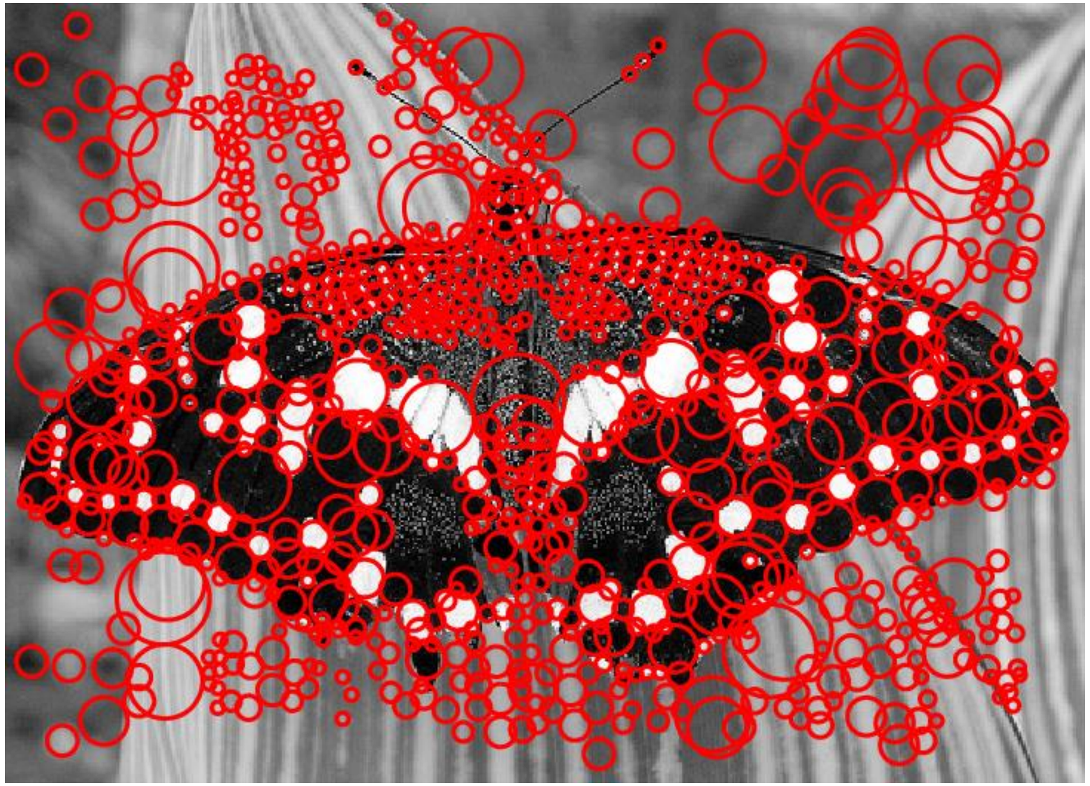
$\Rightarrow$ **List of *(x, y, s)***

Scale-space blob detector: Example

Scale-space blob detector: Example

sigma = 11.9912

Scale-space blob detector: Example



Scale Invariant Detection

- Functions for determining scale $f = \text{Kernel} * \text{Image}$

Kernels:

$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

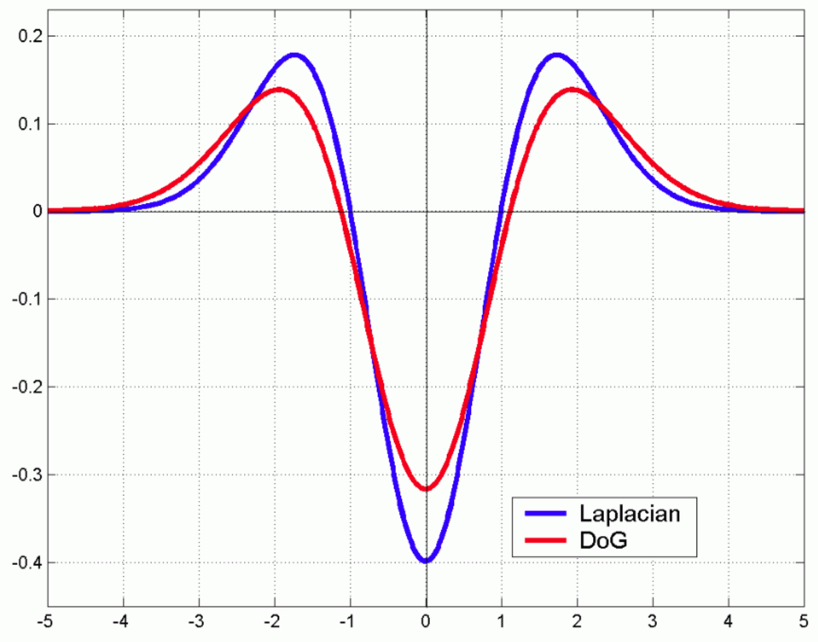
(Laplacian)

$$DoG = G(x, y, k\sigma) - G(x, y, \sigma)$$

(Difference of Gaussians)

where Gaussian

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

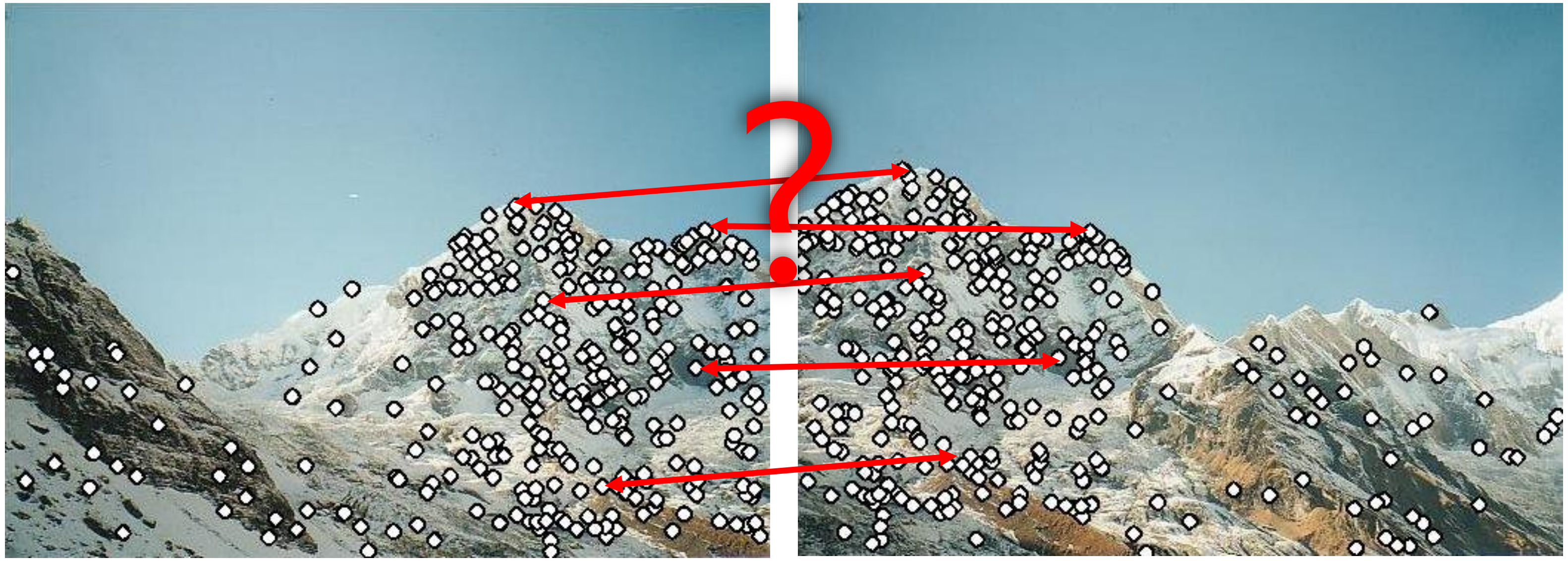


Note: The LoG and DoG operators are both rotation equivariant

Questions?

Feature descriptors

We know how to detect good points
Next question: **How to match them?**



Answer: Come up with a *descriptor* for each point,
find similar descriptors between the two images