Feature Detection and Matching

Feature Point

Motivation

Requirements

Image Matching

Slide credit: Ranjay Krishna

Image Matching

by Diva Sian

by swashford

Slide credit: Steve Seitz

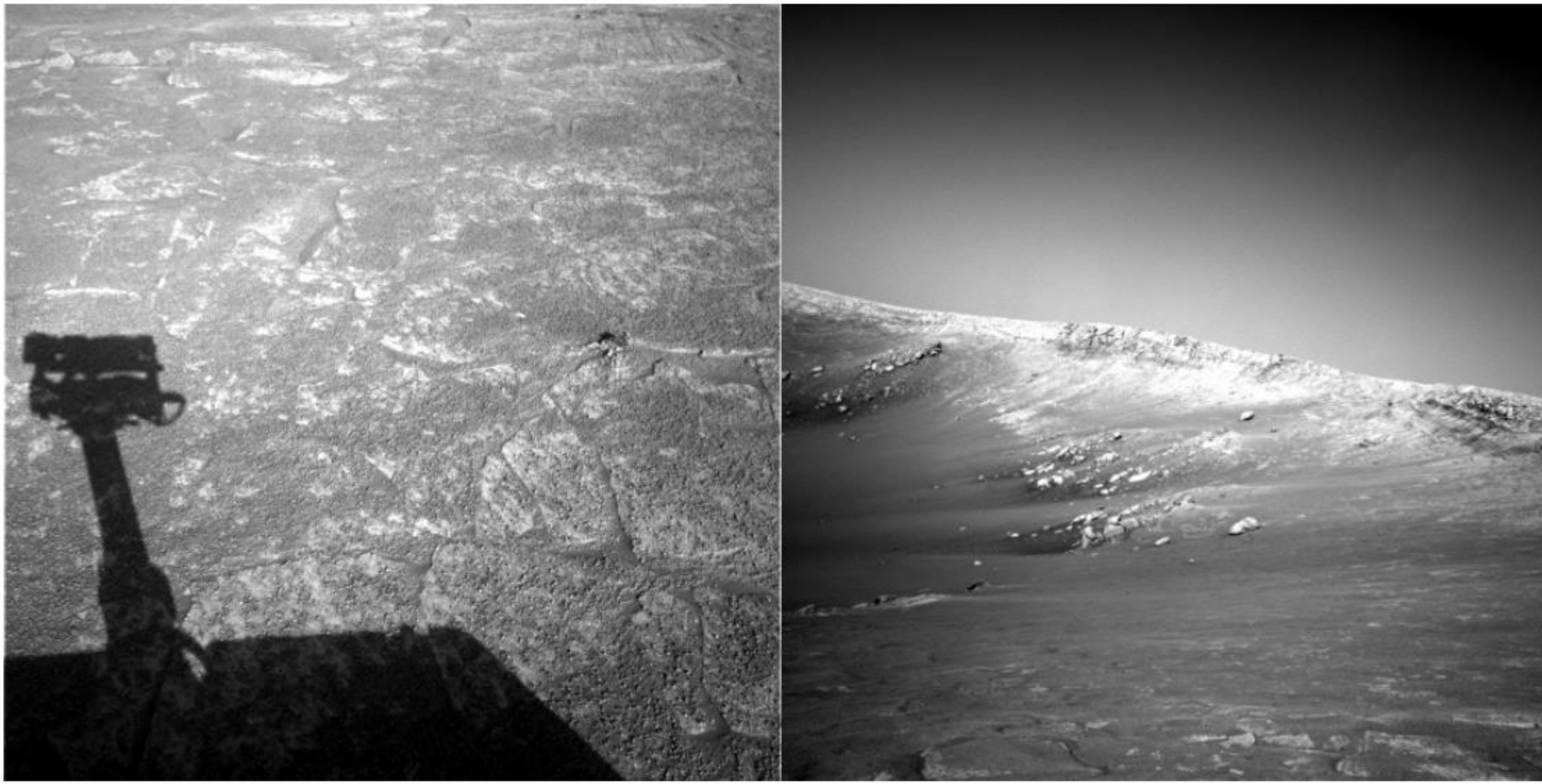
Image Matching

by Diva Sian

by scgbt

Slide credit: Steve Seitz

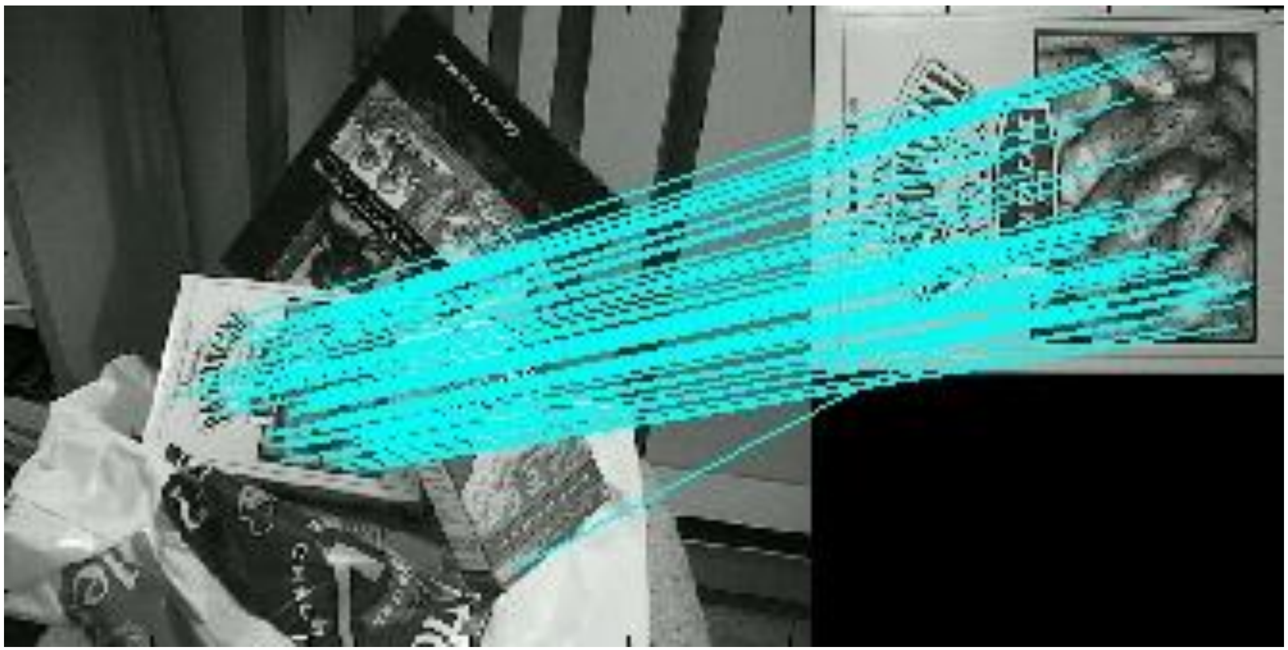
Can you match two images?



NASA Mars Rover images

Slide credit: Steve Seitz

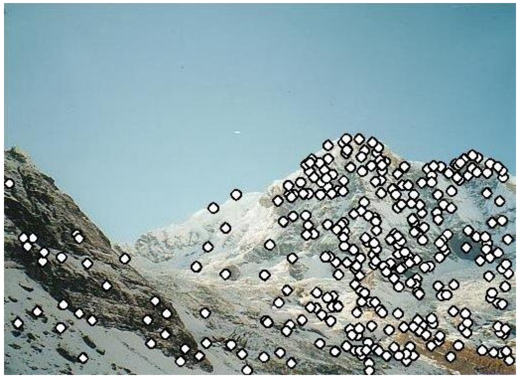
Feature Points



Slide credit: Kristen Grauman

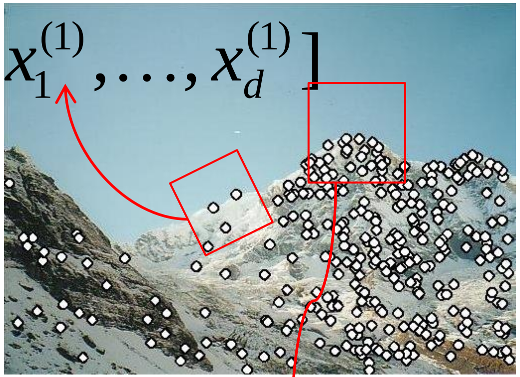
Local Features

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



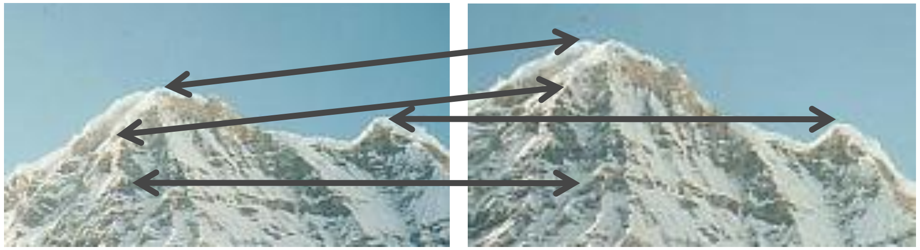
2) **Description:** Extract vector feature descriptor surrounding each interest point.

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



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

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



Slide credit: Kristen Grauman

Properties for Local Features

- Repeatability
  - The same feature can be found in several images despite geometric and photometric transformations
- Saliency
  - Each feature has a distinctive description
- Compactness and efficiency
  - Many fewer features than image pixels
- Locality
  - A feature occupies a relatively small area of the image; robust to clutter and occlusion

Slide credit: Kristen Grauman

Repeatability (1/2)

- We want to detect (at least some of) the same points in both images.

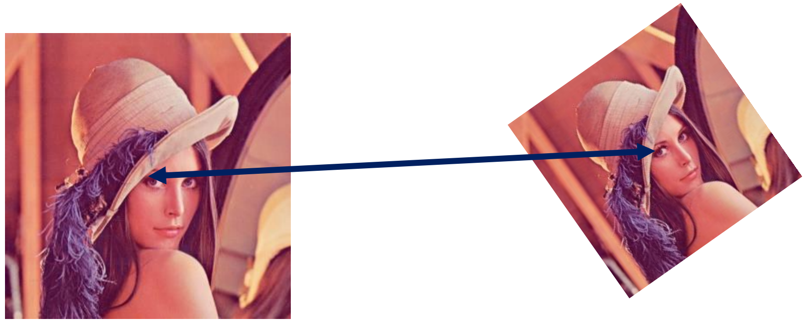
No chance to find true matches!

- Yet we have to be able to run the detection procedure *independently* per image.

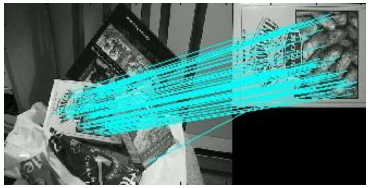
Slide credit: Kristen Grauman

Repeatability (2/2)

- Invariant to translation, rotation, and scale



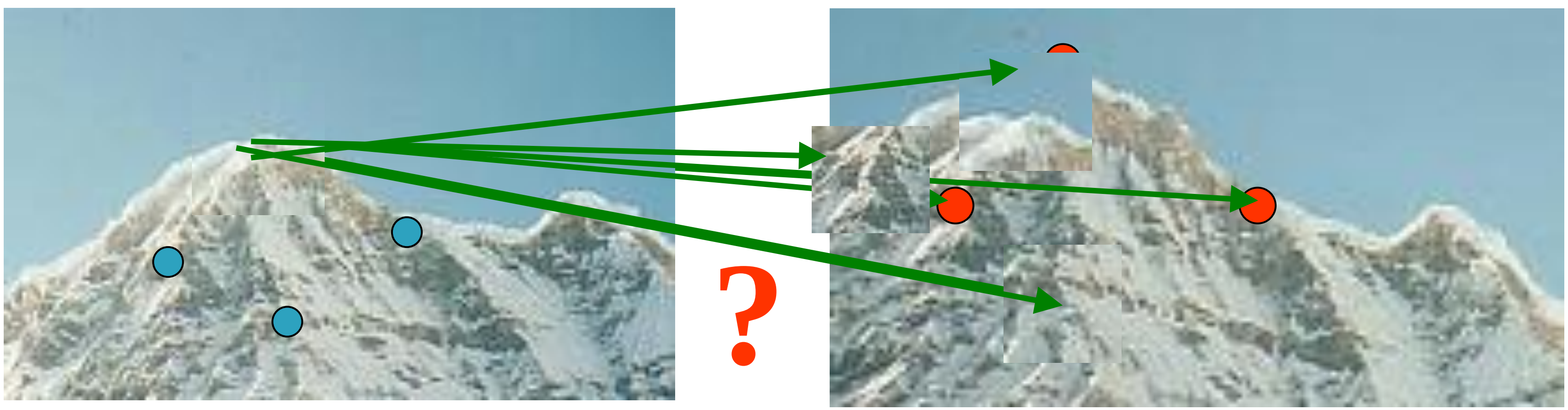
- Robust to transformation



- Robust to lighting and noise

Distinctiveness

- We want to be able to reliably determine which point goes with which.



- Must provide some invariance to geometric and photometric differences between the two views.

Slide credit: Kristen Grauman

Local Invariant Features

- Hessian & **Harris** [Beaudet ‘78], [Harris ‘88]
- Laplacian, DoG [Lindeberg ‘98], [Lowe ‘99]
- Harris-/Hessian-Laplace [Mikolajczyk & Schmid ‘01]
- Harris-/Hessian-Affine [Mikolajczyk & Schmid ‘04]
- EBR and IBR [Tuytelaars & Van Gool ‘04]
- MSER [Matas ‘02]
- Salient Regions [Kadir & Brady ‘01]
- Others…

- Those detectors have become a basic building block for many applications in Computer Vision.

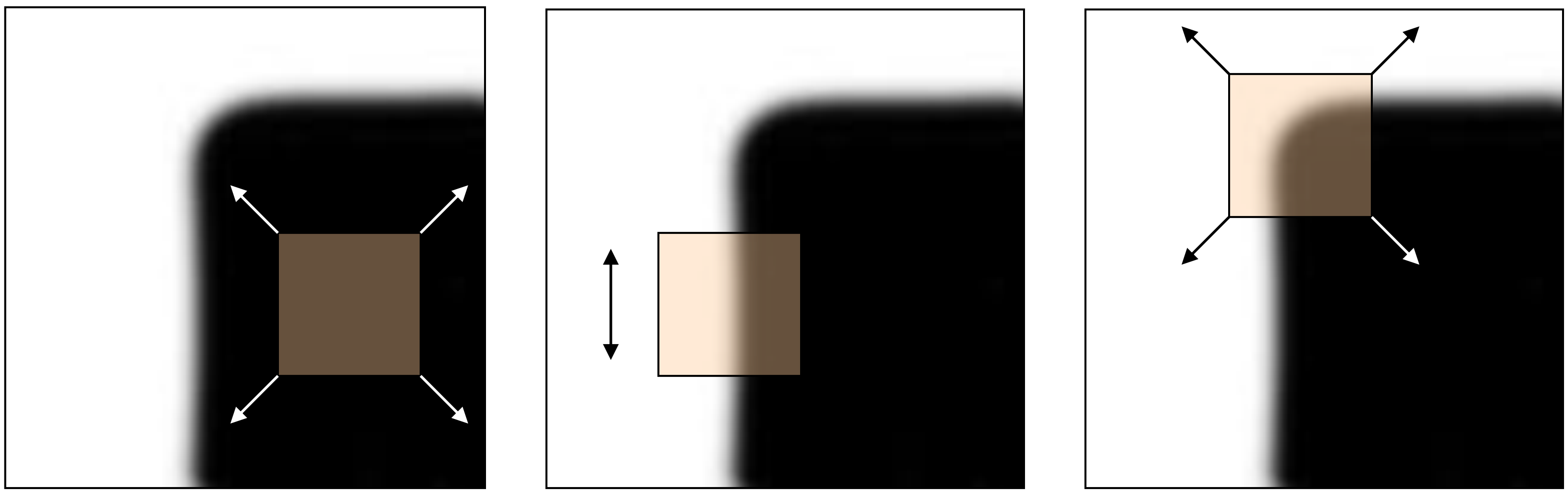
Slide credit: Bastian Leibe

Feature Point Detection

Harris Corner

Corners as distinctive interest points

- Shifting a window in *any direction* should give *a large change* in intensity



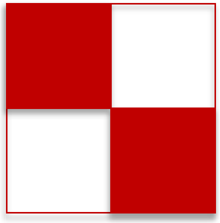
"flat" region:
no change in all directions

"edge":
no change along th
e edge direction

"corner":
significant change in all directions

Slide credit: Alyosha Efros, Darya Frolova, Denis Simakov

Corners versus edges



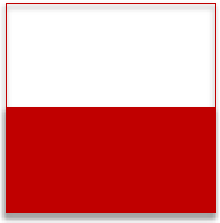
$$\sum I_x^2 \longrightarrow \text{Large}$$

$$\sum I_y^2 \longrightarrow \text{Large}$$

Corner

$$I_x \Leftrightarrow \frac{\partial I}{\partial x}$$

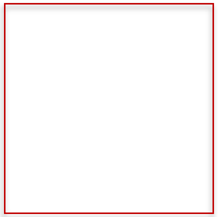
$$I_y \Leftrightarrow \frac{\partial I}{\partial y}$$



$$\sum I_x^2 \longrightarrow \text{Small}$$

$$\sum I_y^2 \longrightarrow \text{Large}$$

Edge



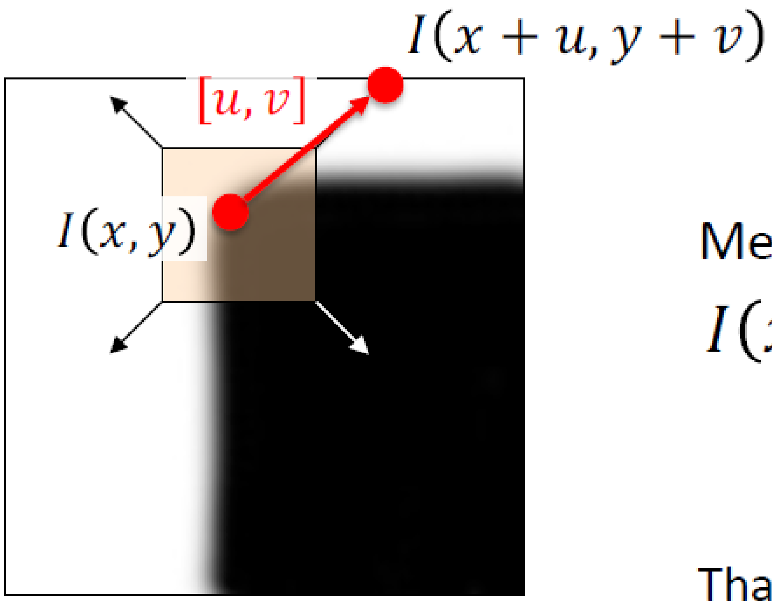
$$\sum I_x^2 \longrightarrow \text{Small}$$

$$\sum I_y^2 \longrightarrow \text{Small}$$

Nothing

Slide credit: Ranjay Krishna

Measure change in all directions



"corner": significant change in all directions

Measure change as intensity difference:

$$I(x+u, y+v) - I(x, y)$$

That's for a single point, but we have to accumulate over a "small window" around that point…

Slide credit: Ranjay Krishna

Harris Detector Formulation

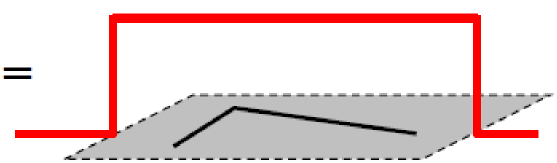
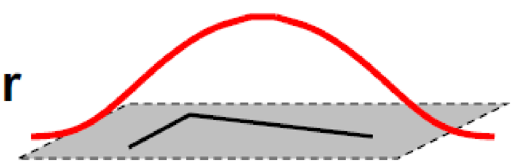
- Change of intensity for the shift [u,v]:

$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u, y+v) - I(x,y)\right]^2$$

Window function

Shifted intensity

Intensity

Window function $w(x,y) =$  or 

1 in window, 0 outside

Gaussian

Slide credit: Rick Szeliski

Harris Detector Formulation

- This measure of change can be approximated by (Taylor expansion):

$$E(u,v) \approx [u \;\; v] \; M \begin{bmatrix} u \\ v \end{bmatrix}$$

where M is a 2×2 matrix computed from image derivatives:

$$M = \underbrace{\sum_{x,y}}_{\text{Sum over image region – the area we are checking for corner}} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Gradient with respect to x, times gradient with respect to y

Sum over image region – the area we are checking for corner

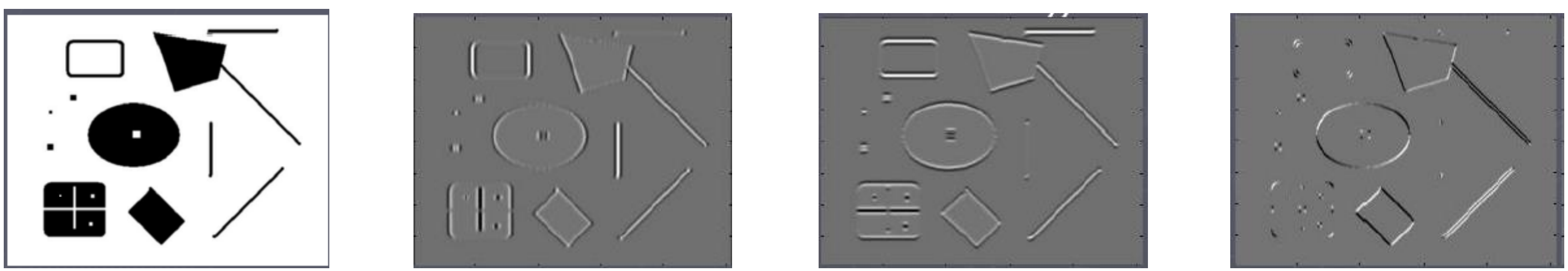
$$M = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \; I_y]$$

Slide credit: Rick Szeliski

Harris Detector Formulation

$$M = \sum w(x,y) \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix}$$

2 x 2 matrix of image derivatives



Notation:

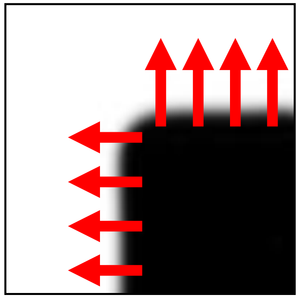
$$I_x \Leftrightarrow \frac{\partial I}{\partial x} \qquad I_y \Leftrightarrow \frac{\partial I}{\partial y} \qquad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$$

Slide credit: Kristen Grauman

What does this matrix reveal?

- First, consider an axis-aligned corner:

$$M = \sum \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$



- This means dominant gradient directions align with x or y axis
- Look for locations where **both** λ's are large.
- If either λ is close to 0, then this is **not** corner-like.

What if we have a corner that is not aligned with the image axes?

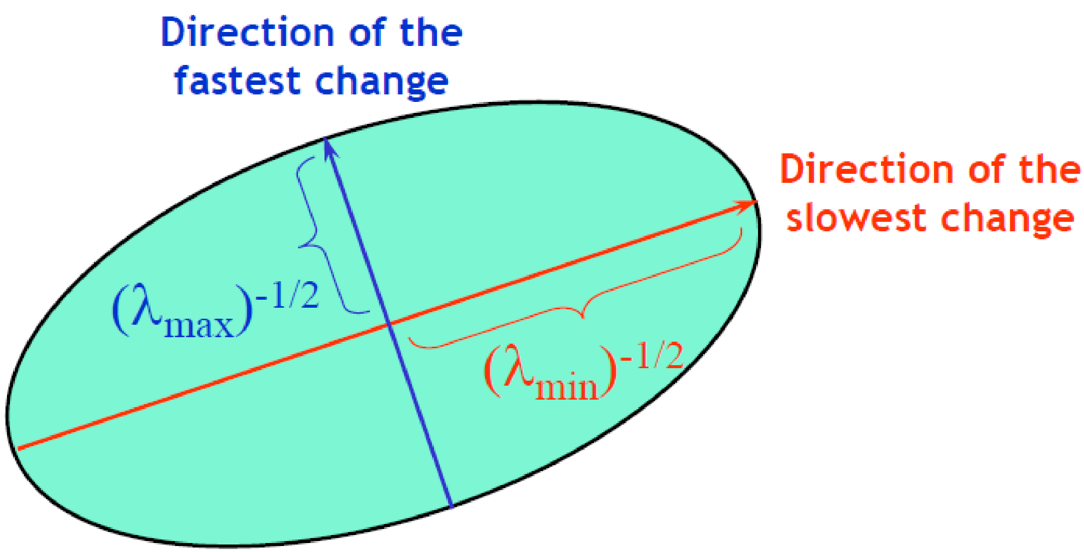
Slide credit: Kristen Grauman

General Case

- Since M is symmetric, we have $M = R^{-1}\begin{bmatrix}\lambda_1 & 0 \\ 0 & \lambda_2\end{bmatrix}R$

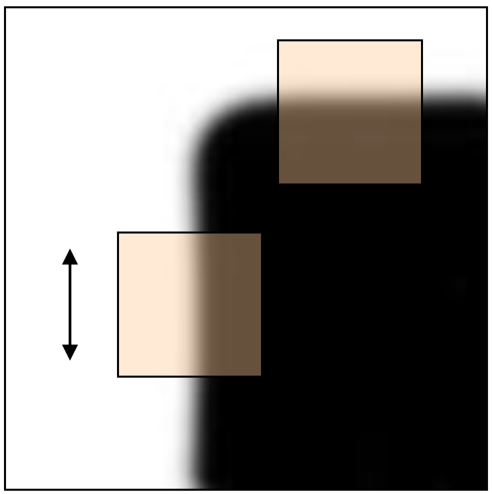
  (Eigenvalue decomposition)

- We can think of M as an ellipse with axis lengths determined by the eigenvalues and orientation determined by R



Slide credit: Ranjay Krishna

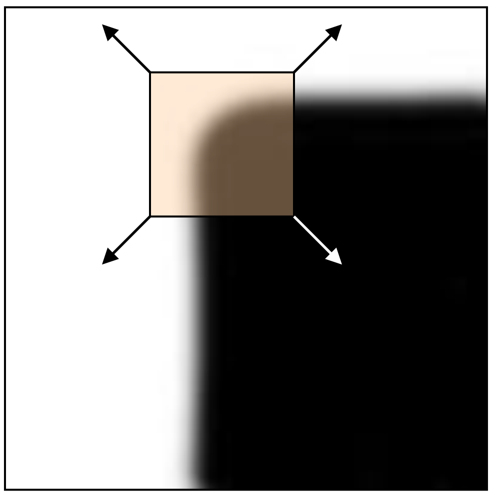
Example



“edge”:

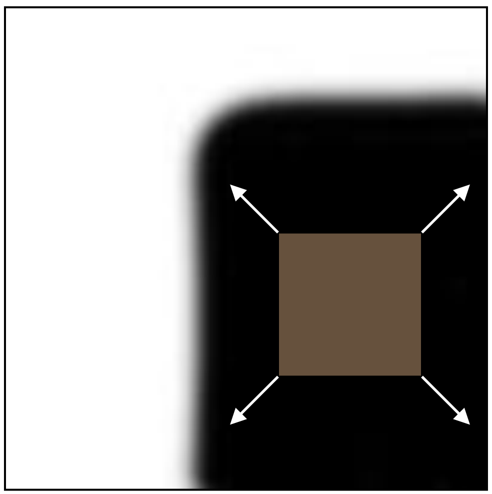
$\lambda_1 >> \lambda_2$

$\lambda_2 >> \lambda_1$



“corner”:

λ_1 and λ_2 are large, $\lambda_1 \sim \lambda_2$;



“flat” region

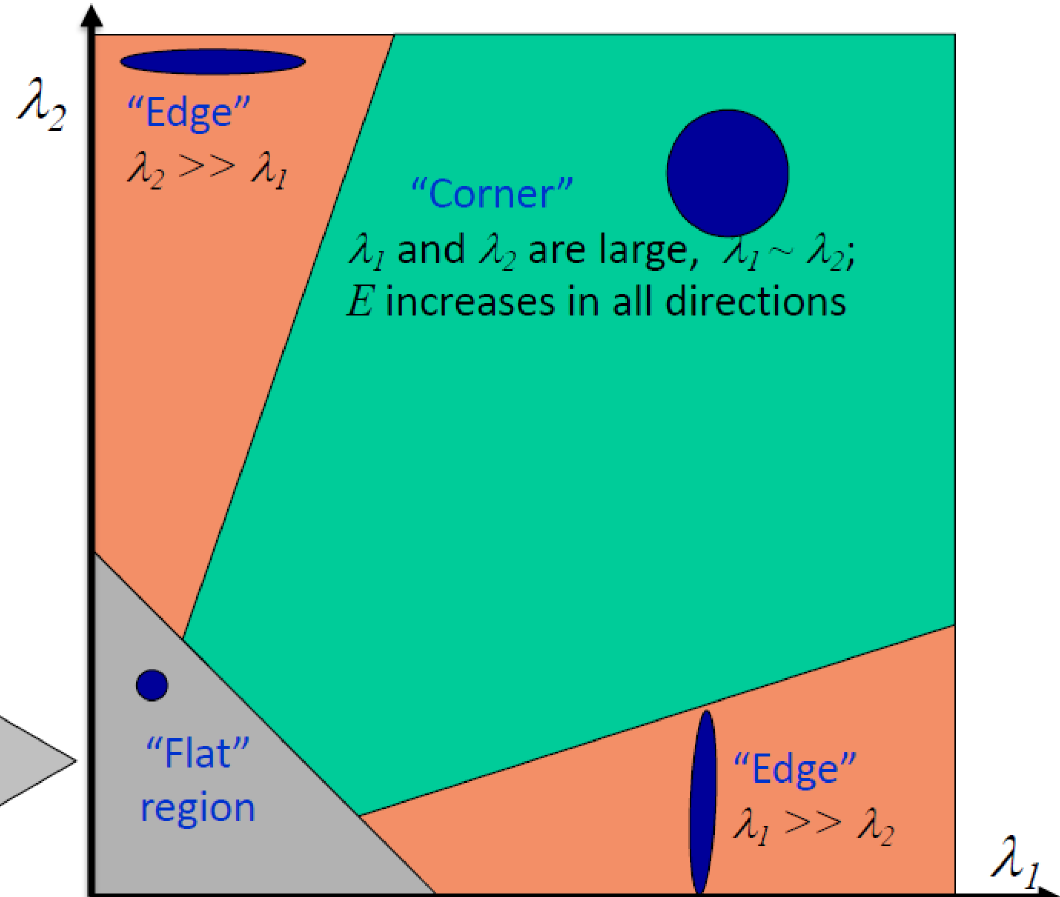
λ_1 and λ_2 are small;

Slide credit: Kristen Grauman

Interpreting the Eigenvalues

- Classification of image points using eigenvalues of M:

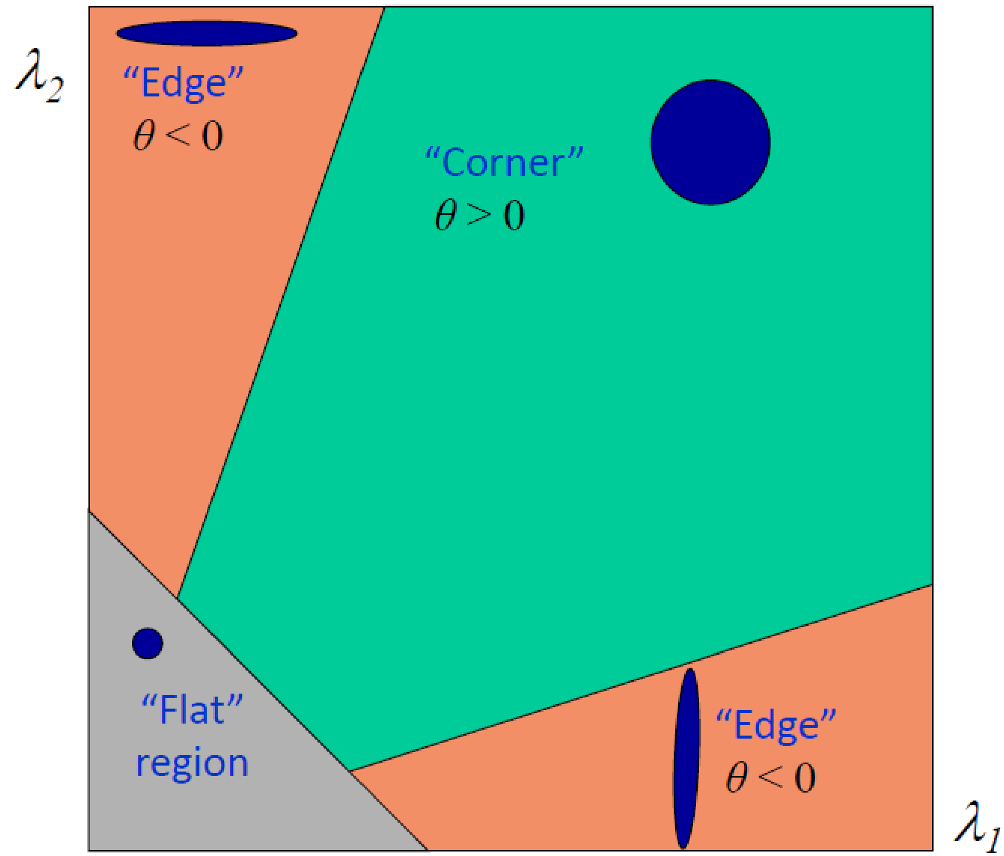
λ_1 and λ_2 are small;
E is almost constant in all directions



Slide credit: Kristen Grauman

Corner Response Function

$$\theta = \det(M) - \alpha \,\mathrm{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$



- Fast approximation
  - Avoid computing the eigenvalues
  - α: constant (0.04 to 0.06)

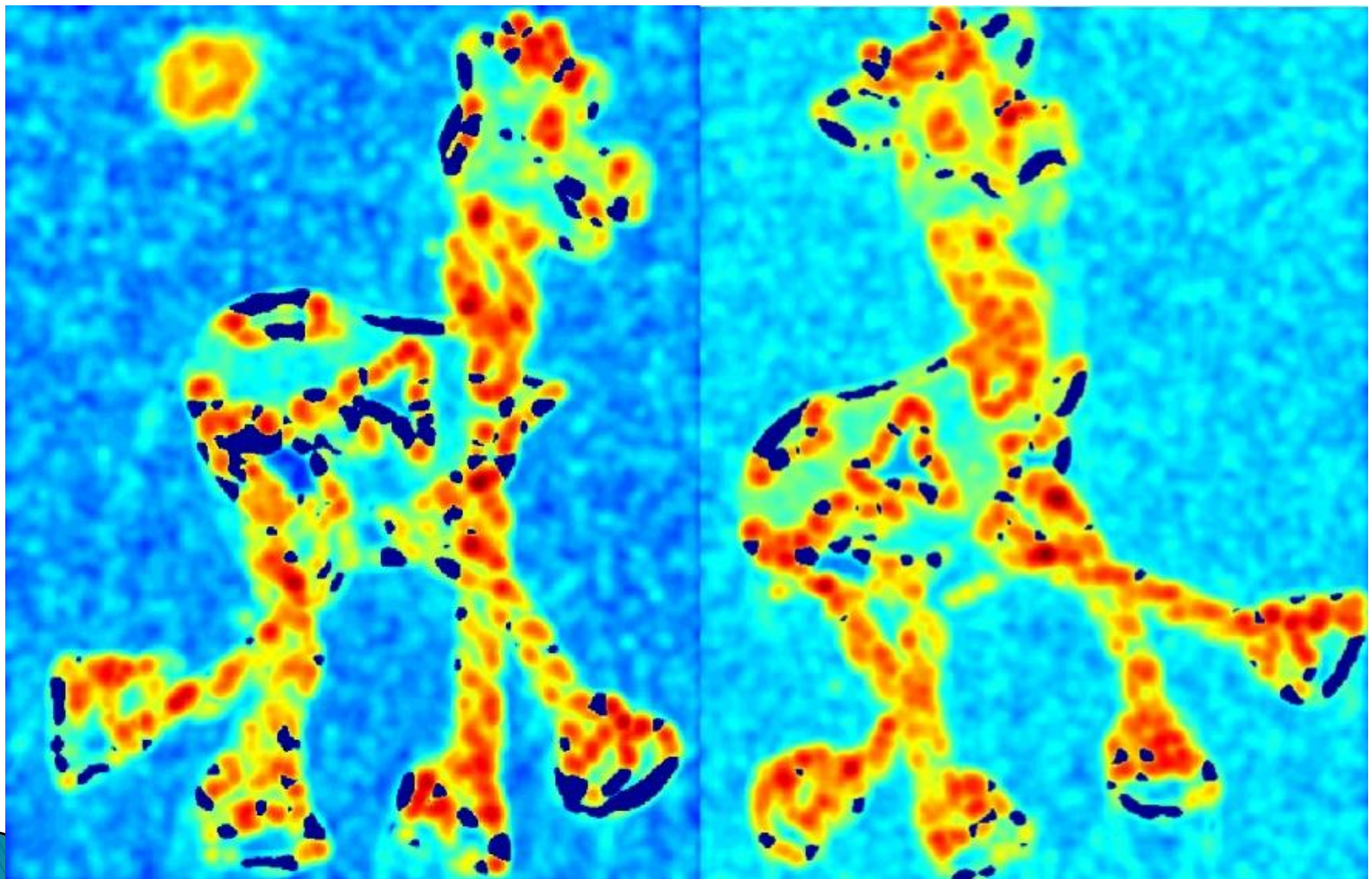
Slide credit: Kristen Grauman

Harris Detector: Steps

Slide credit: Kristen Grauman

Harris Detector: Steps

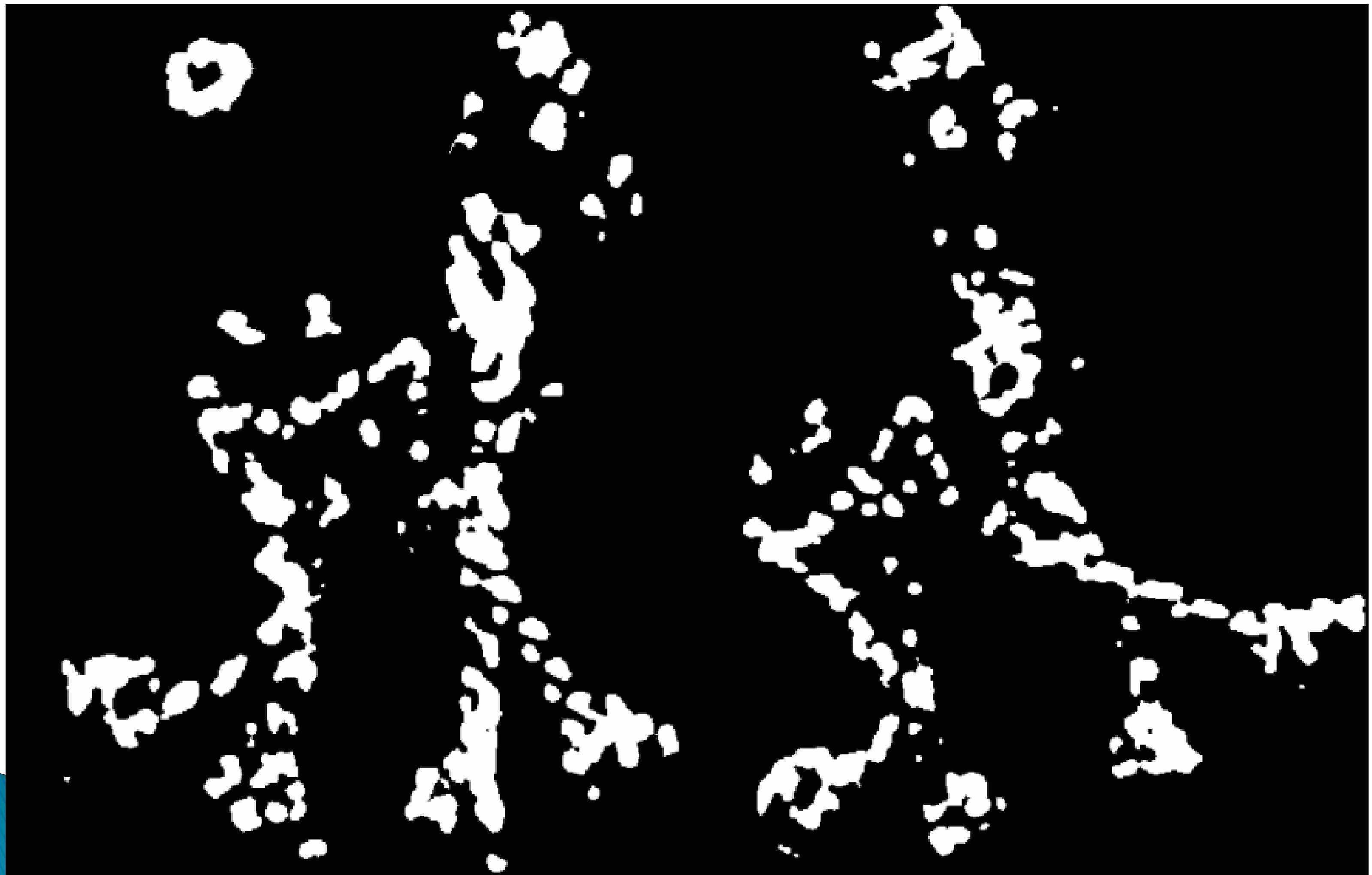
Compute corner response f



Slide credit: Kristen Grauman

Harris Detector: Steps

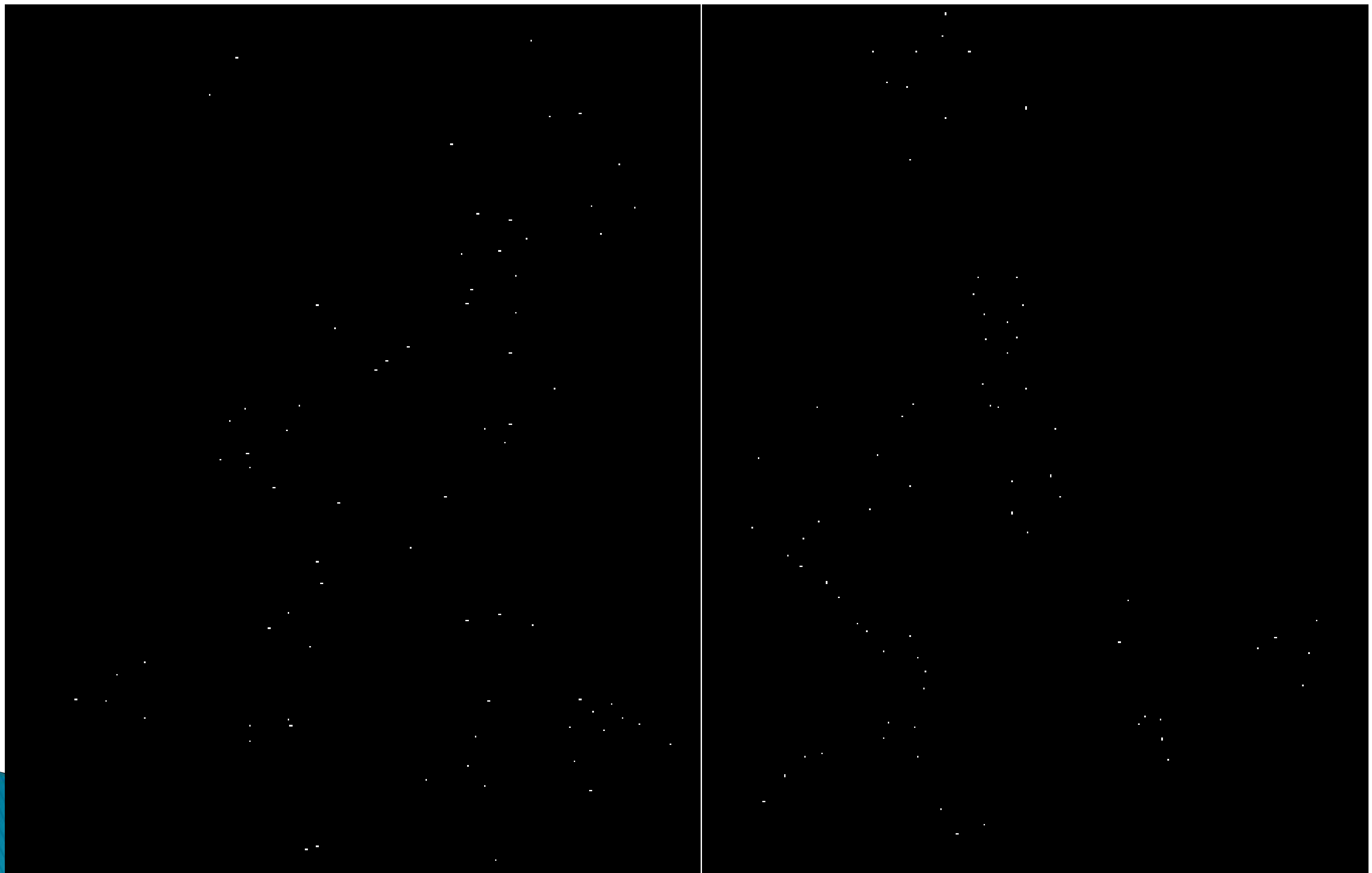
Find points with large corner response: $f >$ threshold



Slide credit: Kristen Grauman

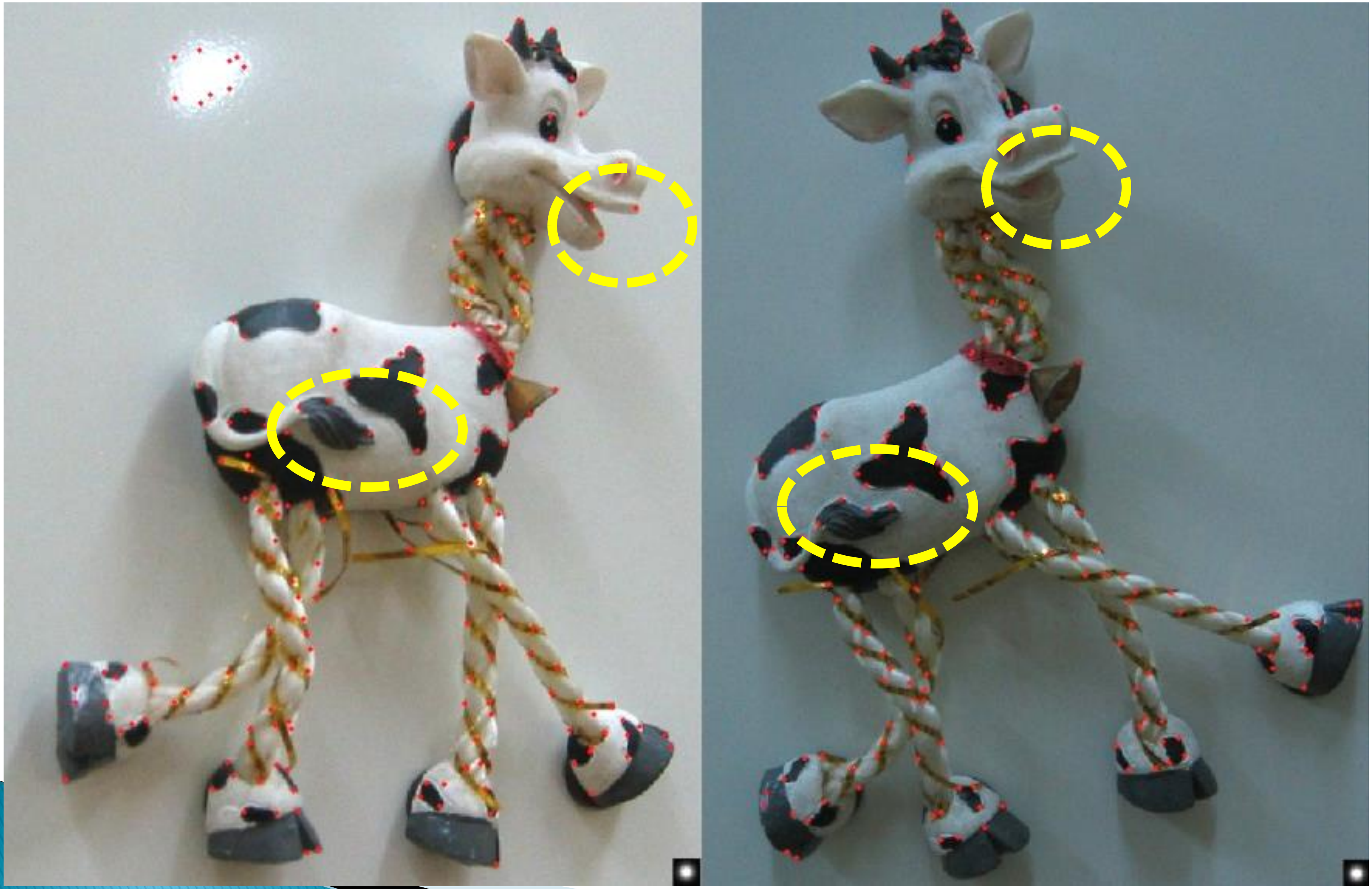
Harris Detector: Steps

Take only the points of local maxima of f



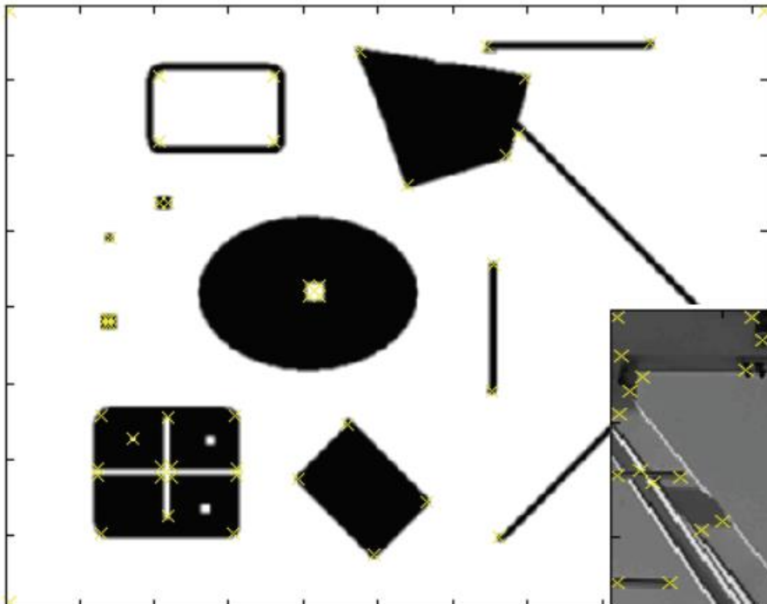
Slide credit: Kristen Grauman

Harris Detector: Steps

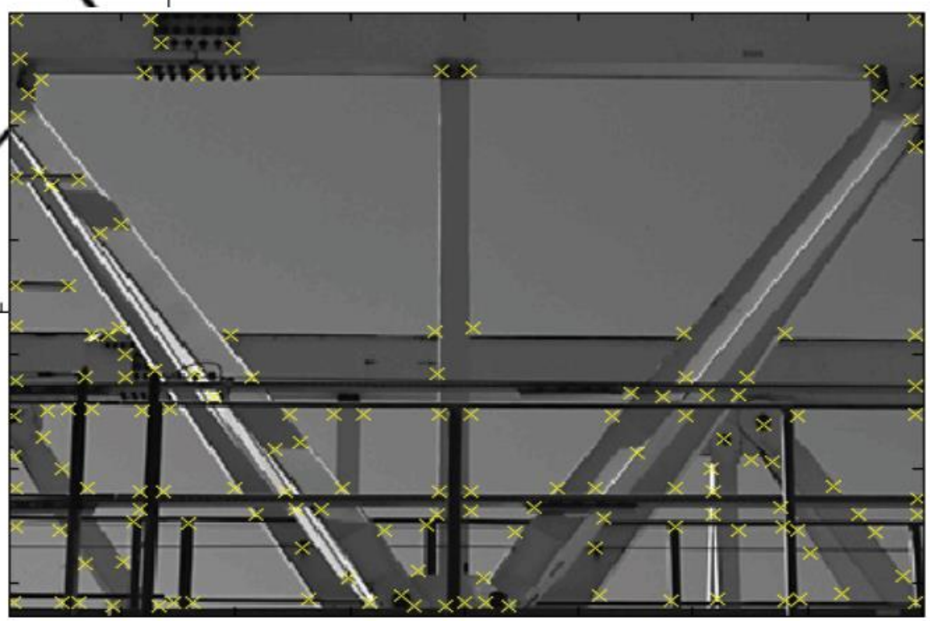


Slide credit: Kristen Grauman

Harris Detector – Responses



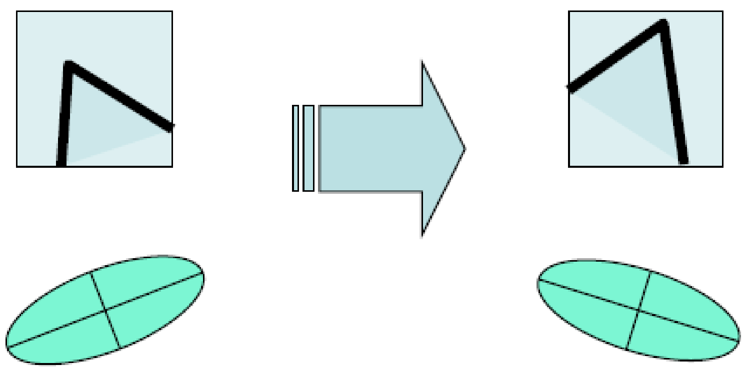
Effect: A very precise corner detector.



Slide credit: Krystian Mikolajczyk

Properties of Harris Corner Detector

- Translation invariant?
- Rotation Invariant?

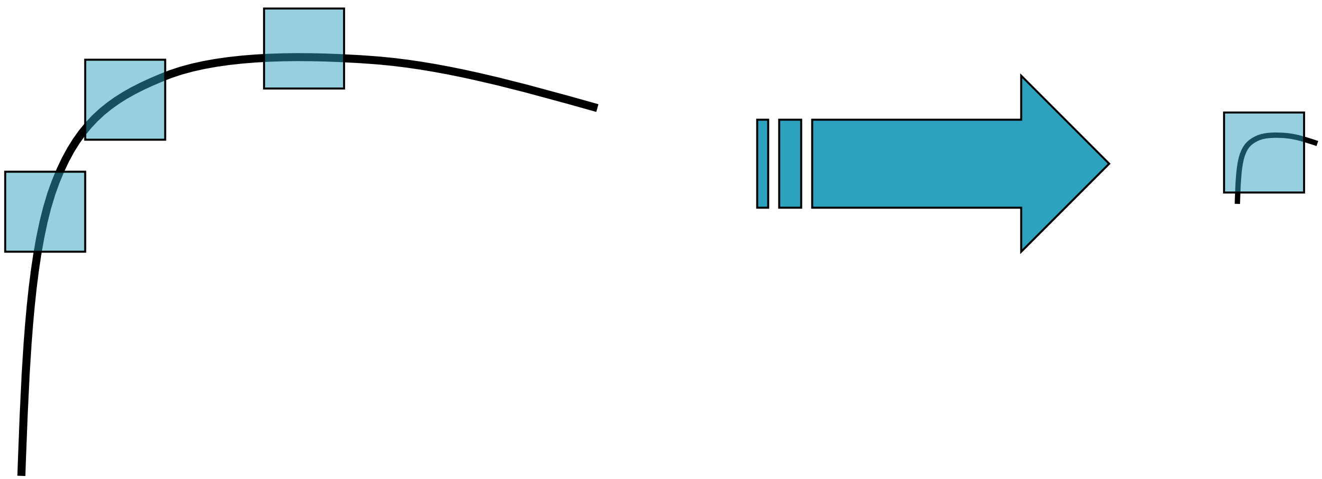


$$M = X \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} X^T$$

Slide credit: Kristen Grauman

Properties of Harris Corner Detector

- Scale invariant?



All points will be cl
assified as edges

Corner
!

Scale Invariant Keypoint Detection

Automatic Scale Selection

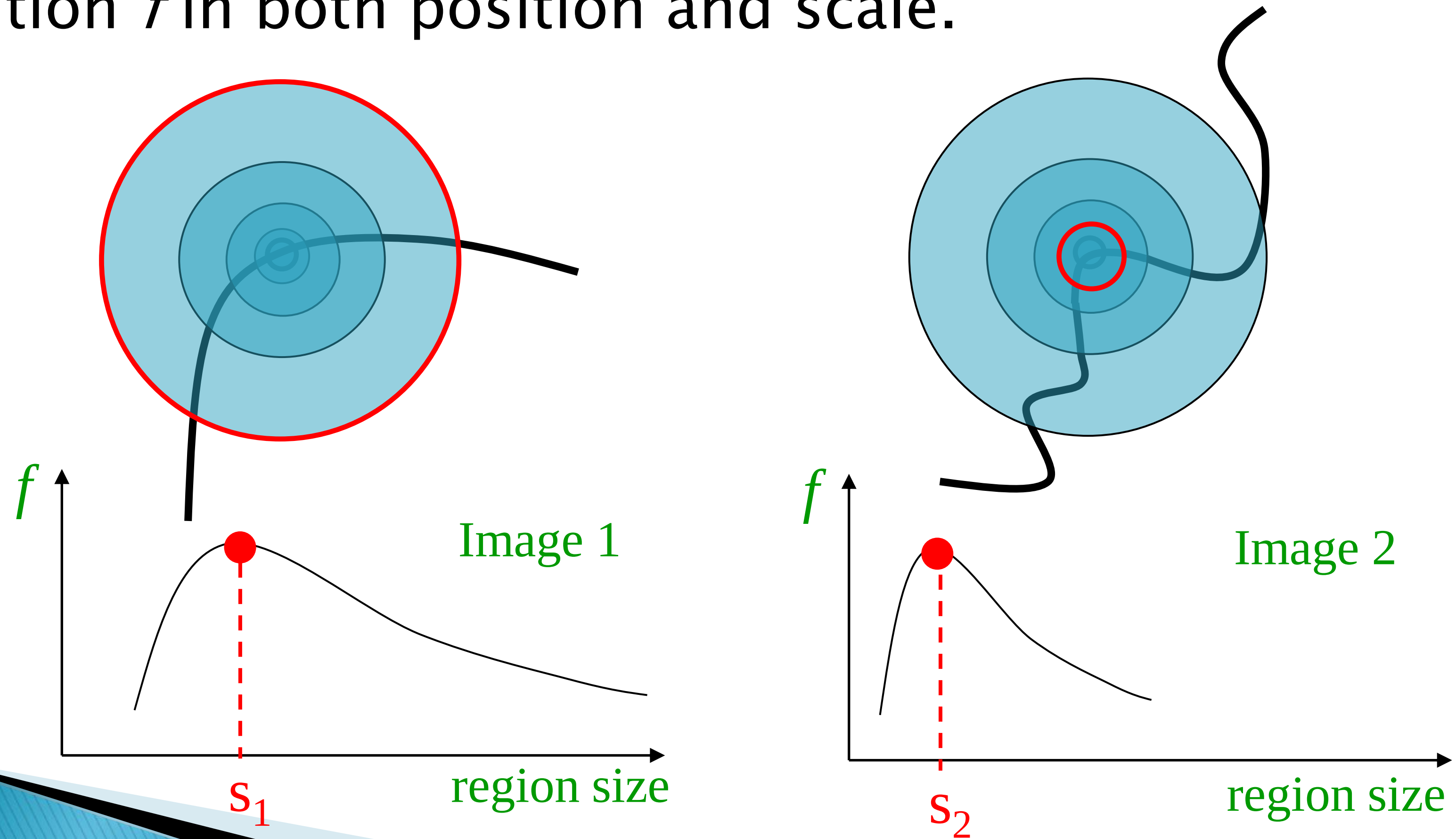
- How do we find corresponding patch sizes?

Slide credit: Kristen Grauman

Automatic scale selection

Intuition:

- Find scale that gives local maxima of some function f in both position and scale.



Slide credit: Kristen Grauman

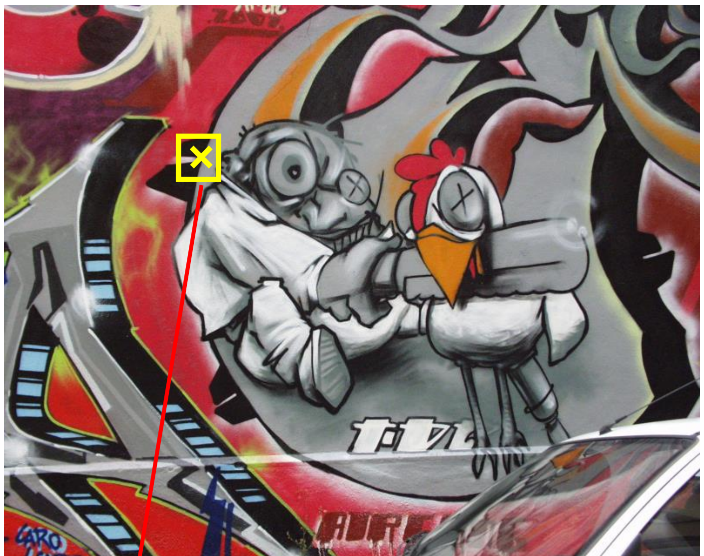
Automatic Scale Selection

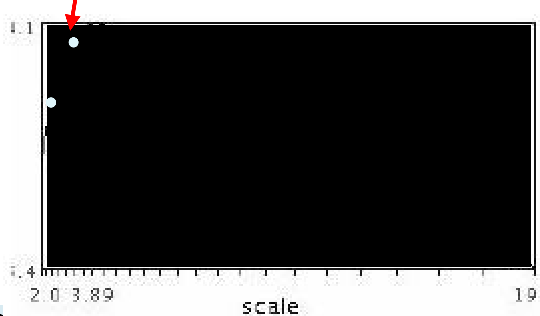
$f(I_{i_1 \dots i_m}(x,\sigma))$

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

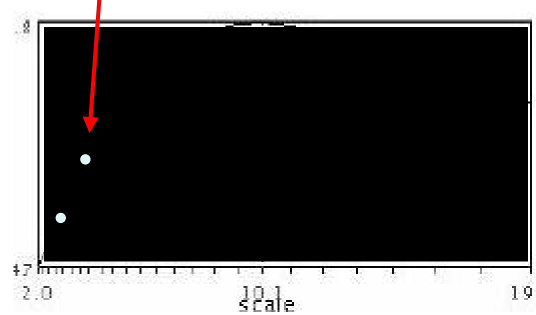
Slide credit: Kristen Grauman

Automatic Scale Selection





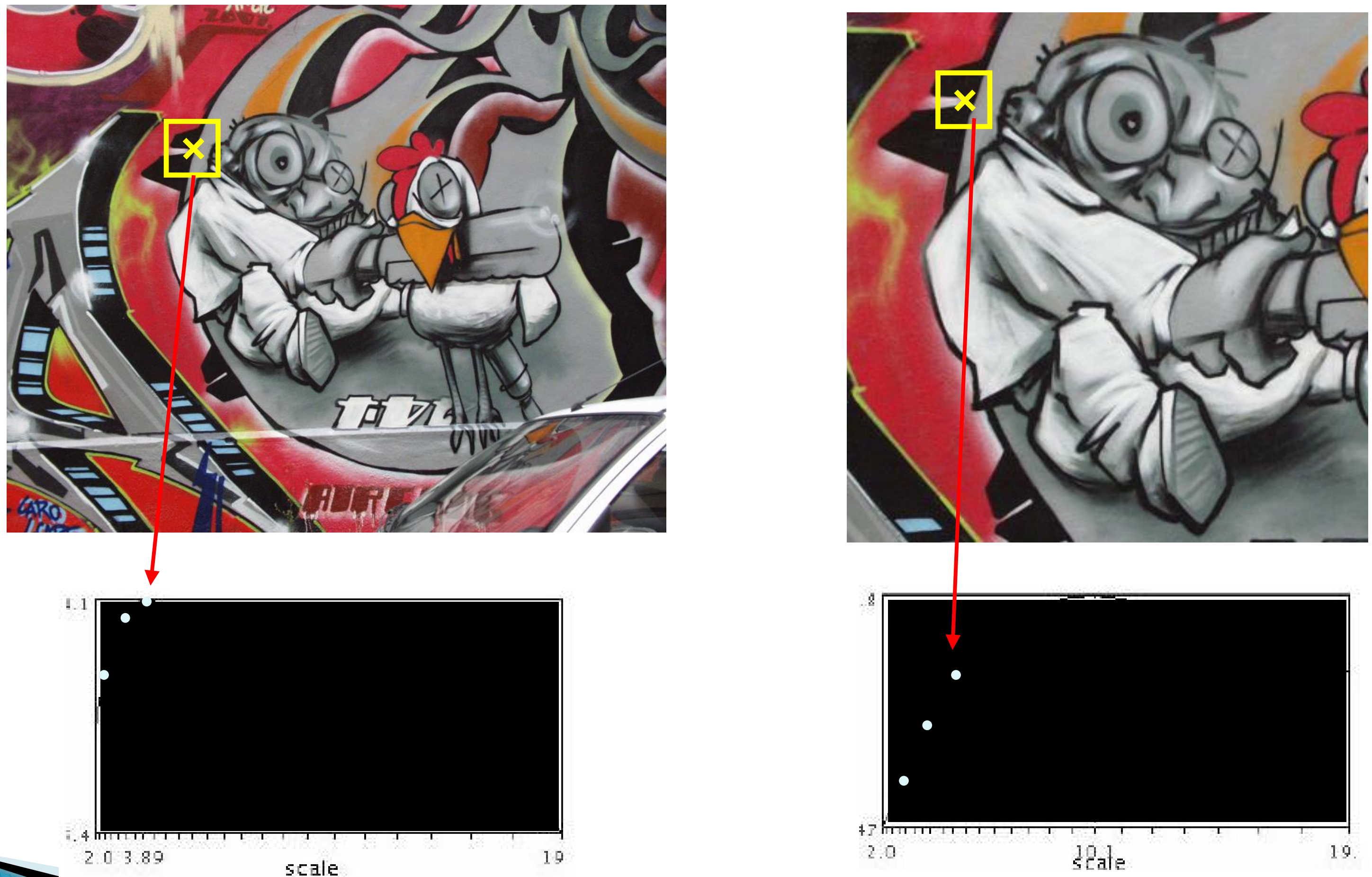
$$f(I_{i_1 \dots i_m}(x, \sigma))$$



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

Slide credit: Kristen Grauman

Automatic Scale Selection

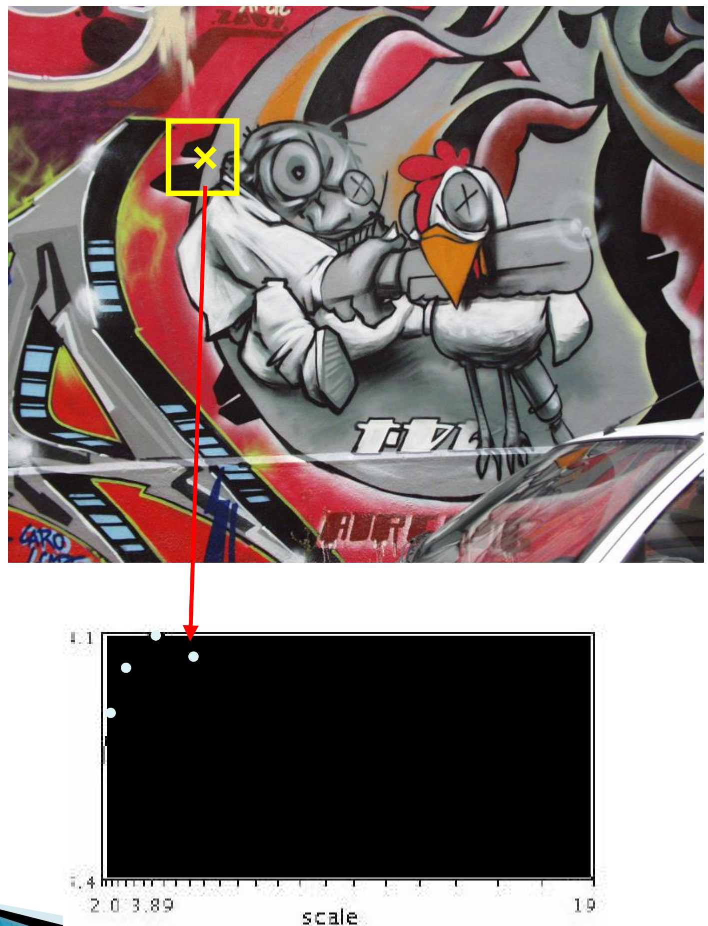


$f(I_{i_1 \ldots i_m}(x, \sigma))$

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

Slide credit: Kristen Grauman

Automatic Scale Selection

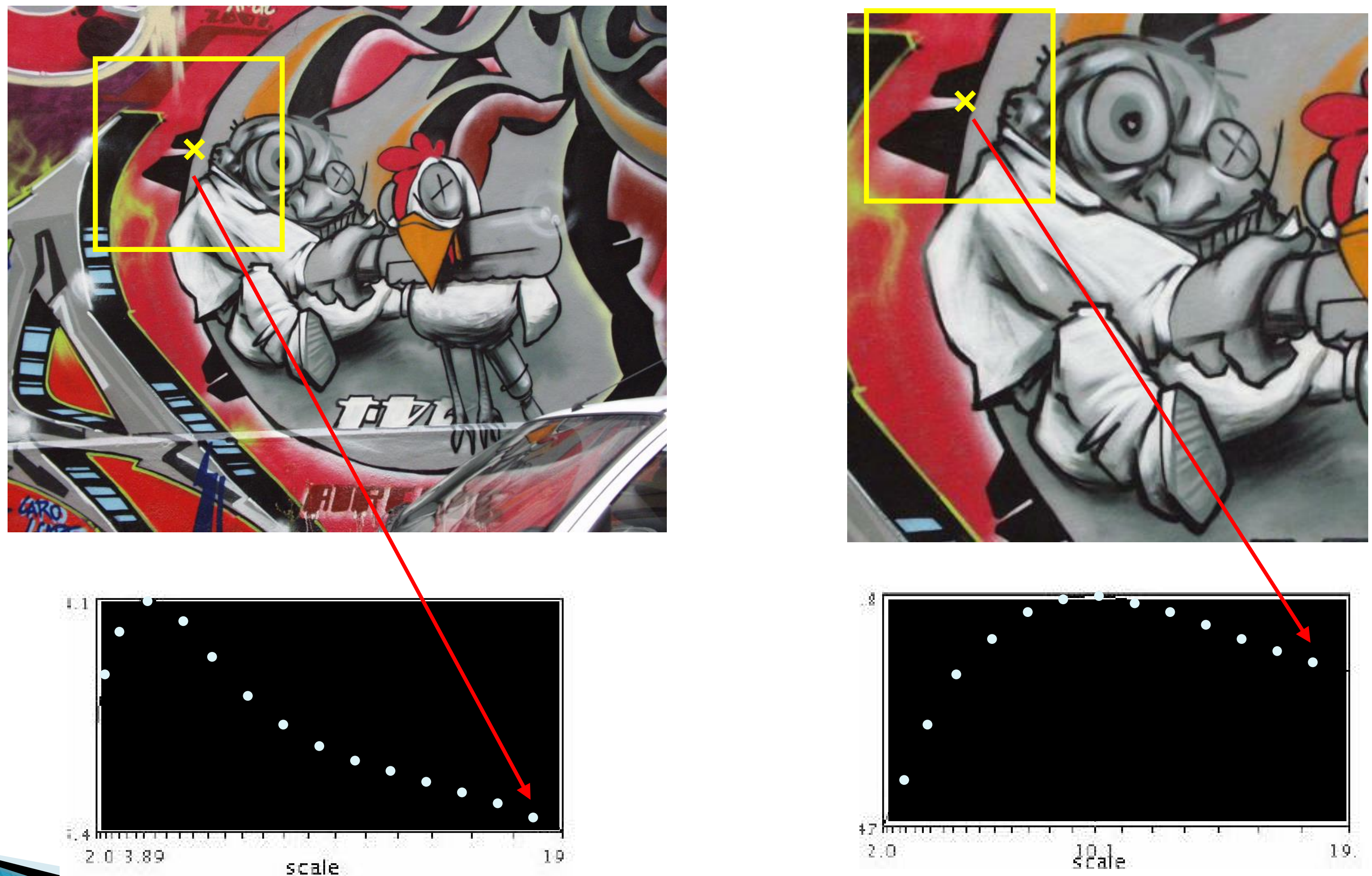


$f(I_{i_1 \ldots i_m}(x,\sigma))$

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

Slide credit: Kristen Grauman

Automatic Scale Selection

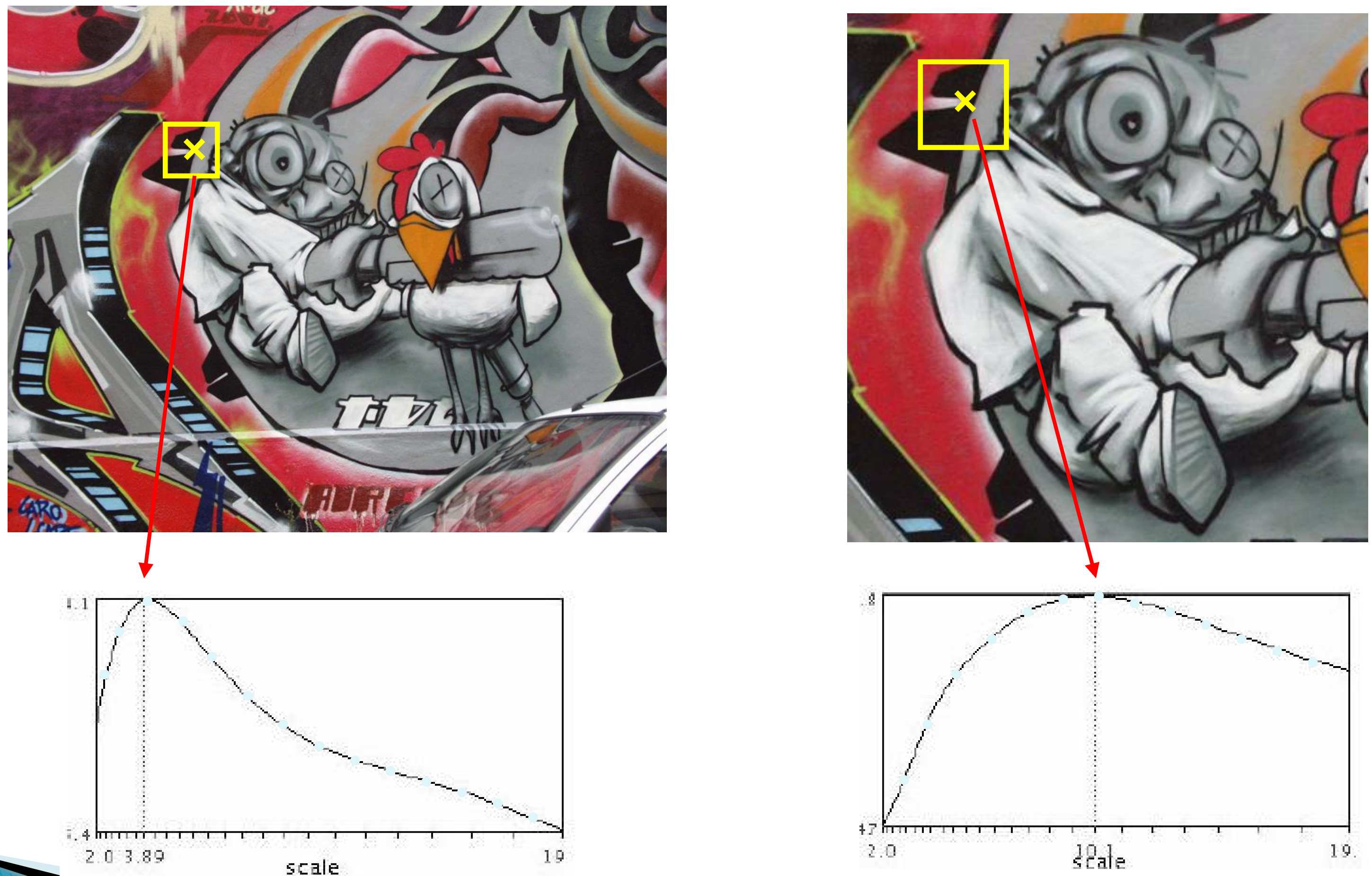


$$f(I_{i_1 \ldots i_m}(x,\sigma))$$

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

Slide credit: Kristen Grauman

Automatic Scale Selection

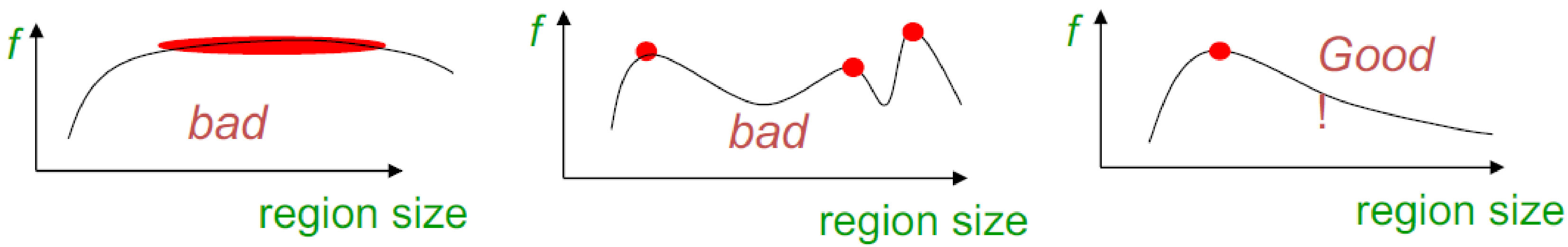


$$f(I_{i_1 \ldots i_m}(x, \sigma)) \qquad f(I_{i_1 \ldots i_m}(x', \sigma'))$$

Slide credit: Kristen Grauman

What is a good function for scale invariant detection?

- A "good" function for scale detection:
  has one stable sharp peak

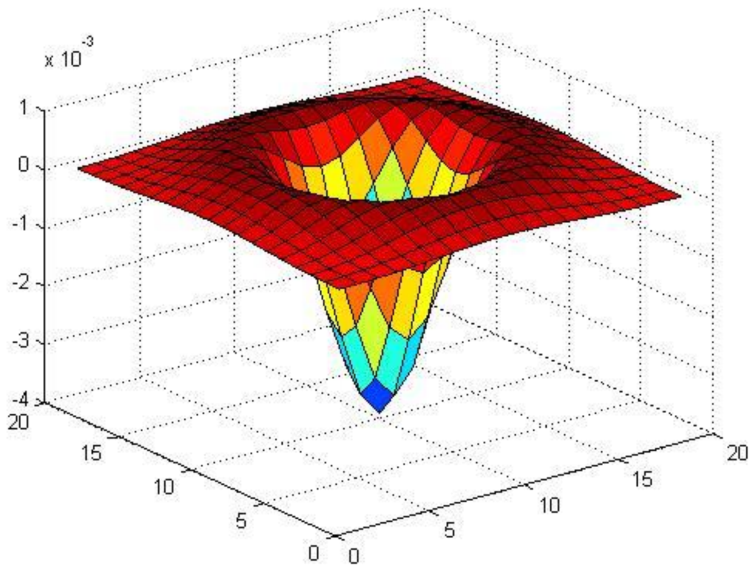


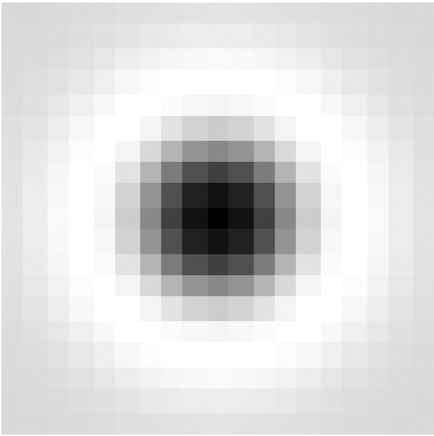
- For usual images: a good function would be one which responds to contrast (sharp local intensity change)

Slide credit: Ranjay Krishna

Blob detection in 2D

- Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D





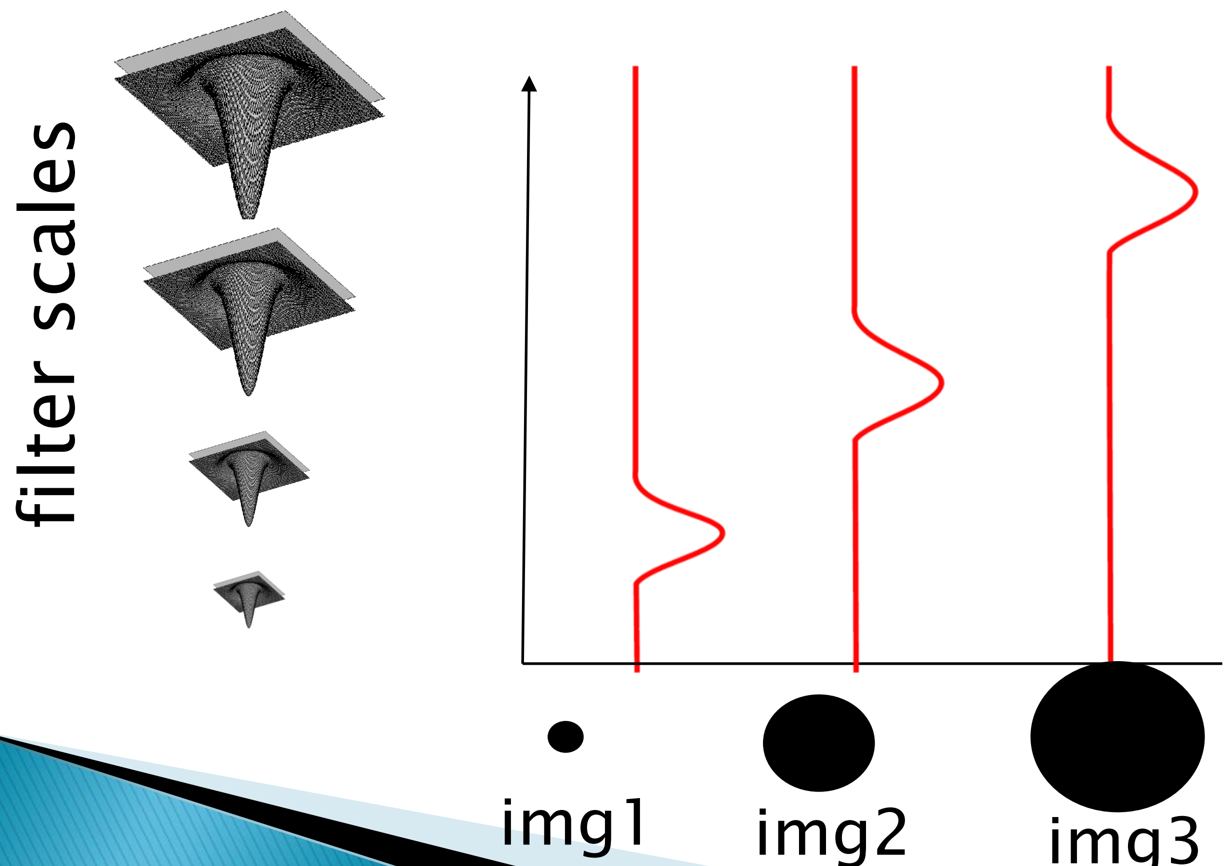
$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

Slide credit: Kristen Grauman

Blob detection in 2D: scale selection

- Laplacian-of-Gaussian = "blob" detector

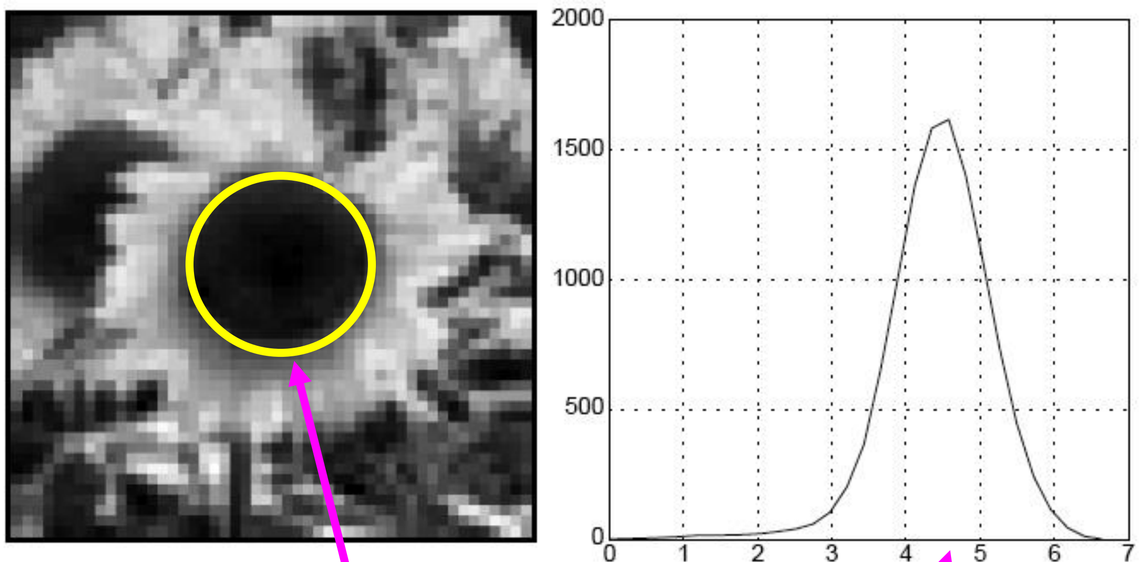
$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$



Slide credit: Kristen Grauman

Blob detection in 2D

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



characteristic scale

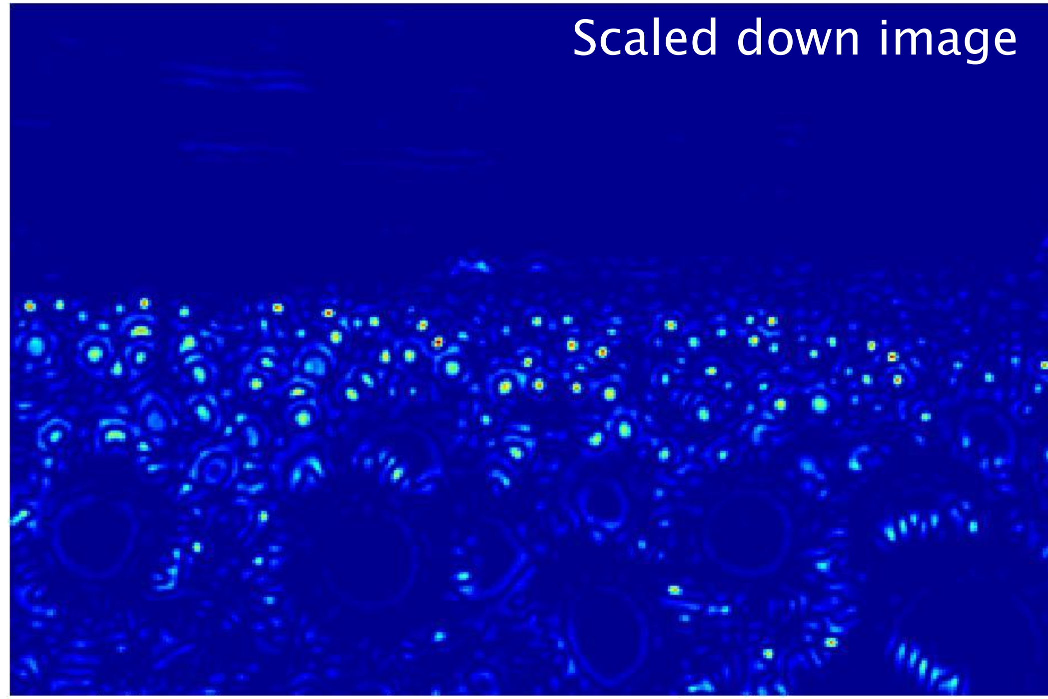
Slide credit: Lana Lazebnik

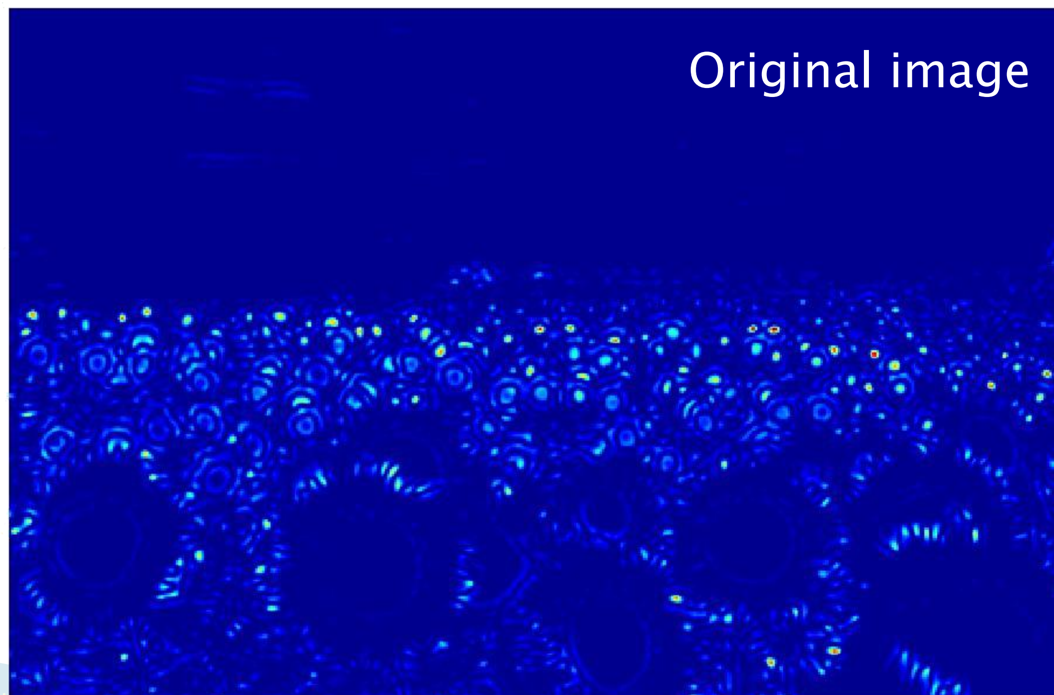
Example

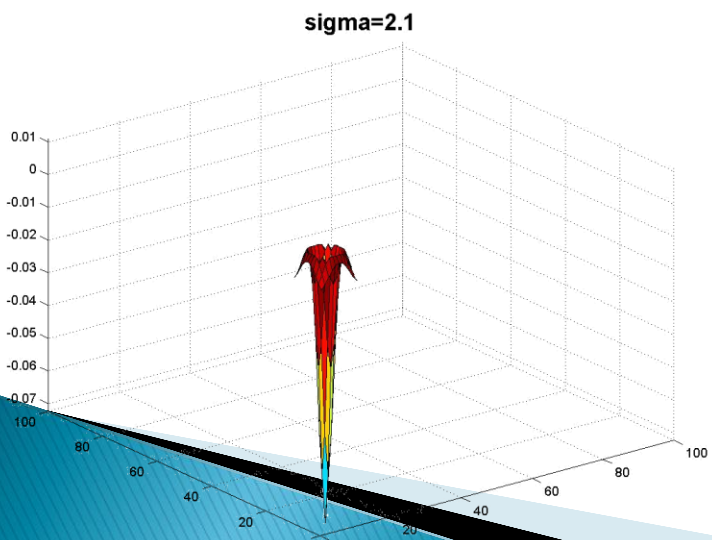
Original image at ¾ the size

Slide credit: Kristen Grauman

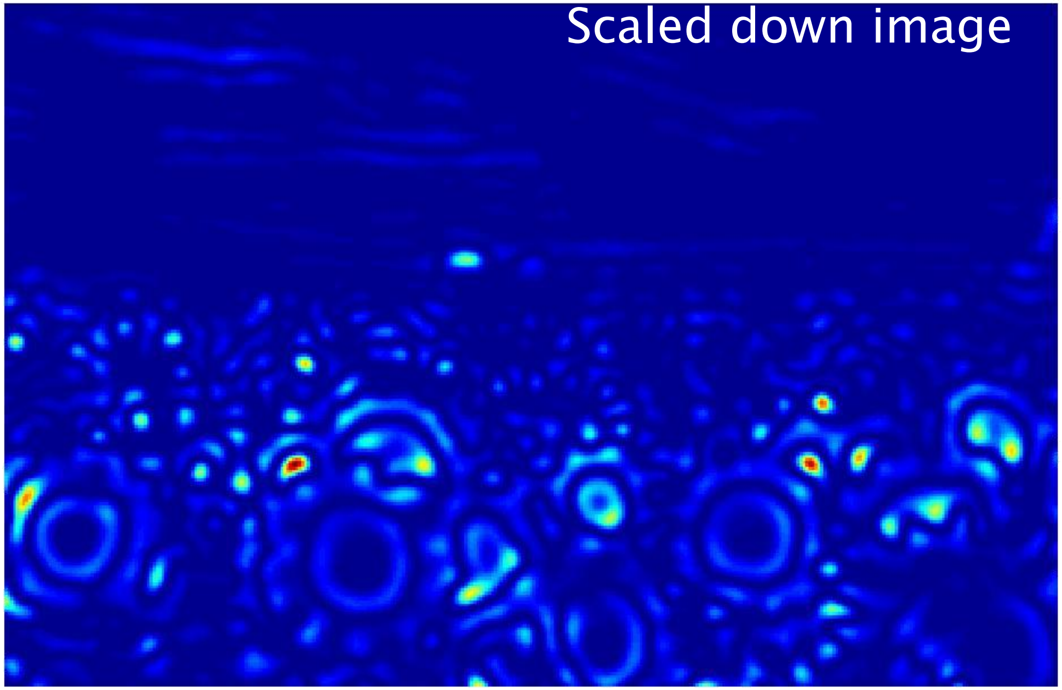
Original image
at ¾ the size

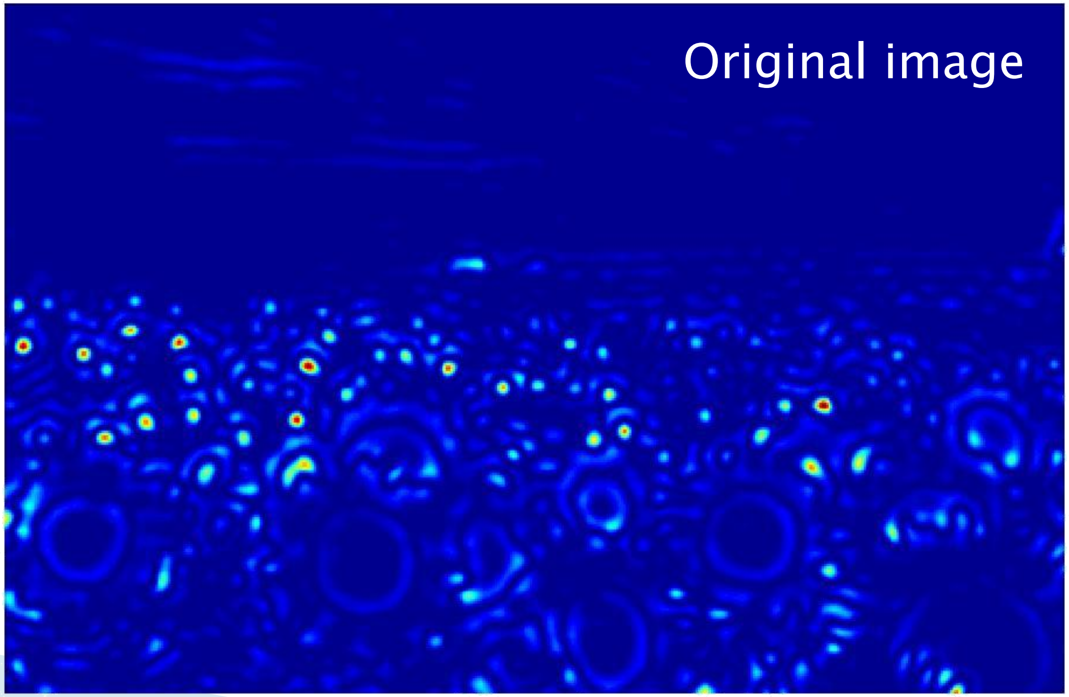


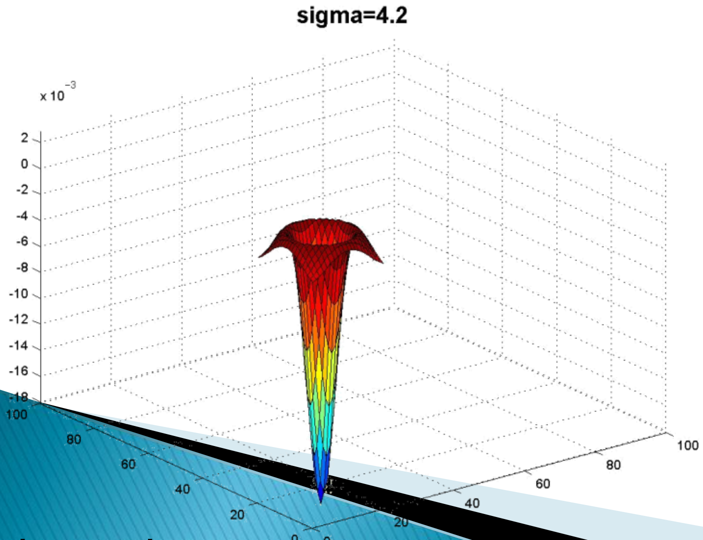




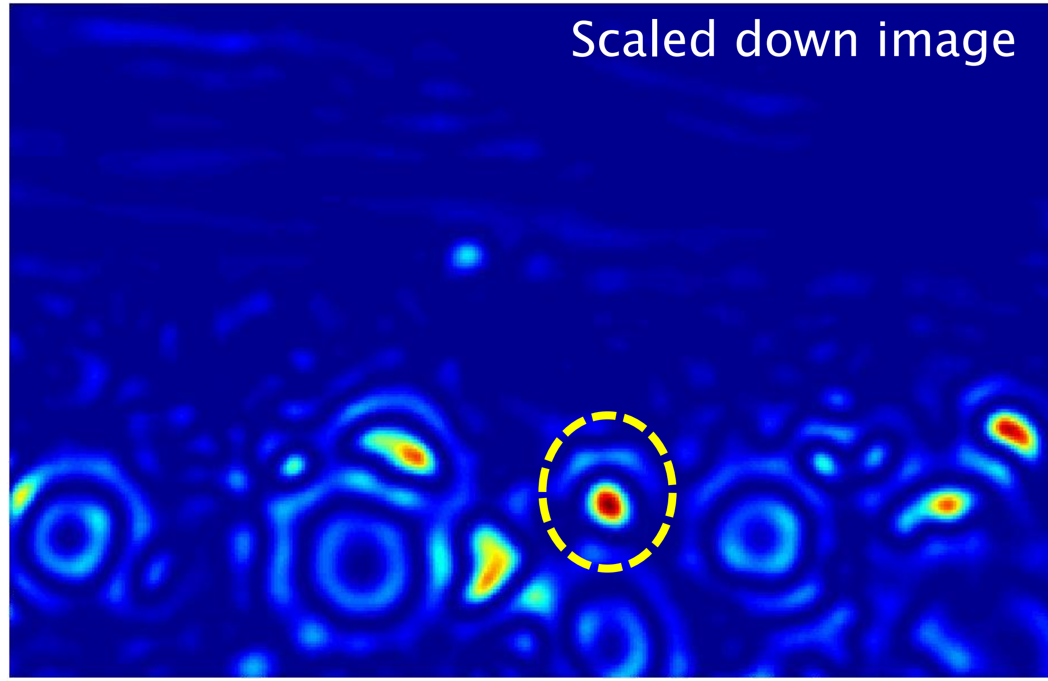
Slide credit: Kristen Grauman

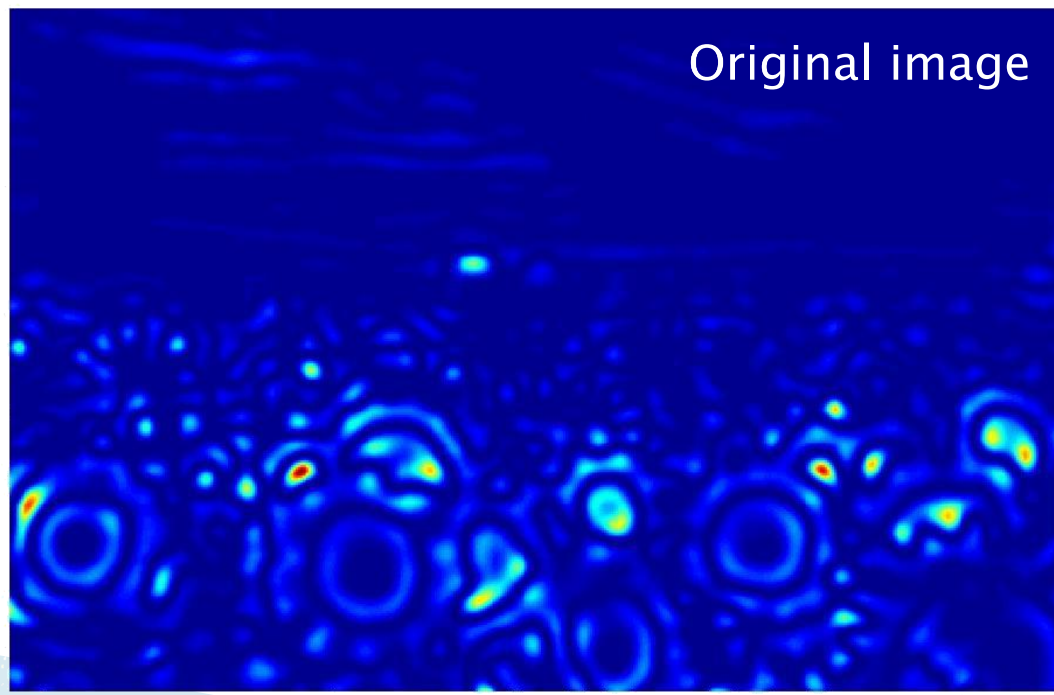


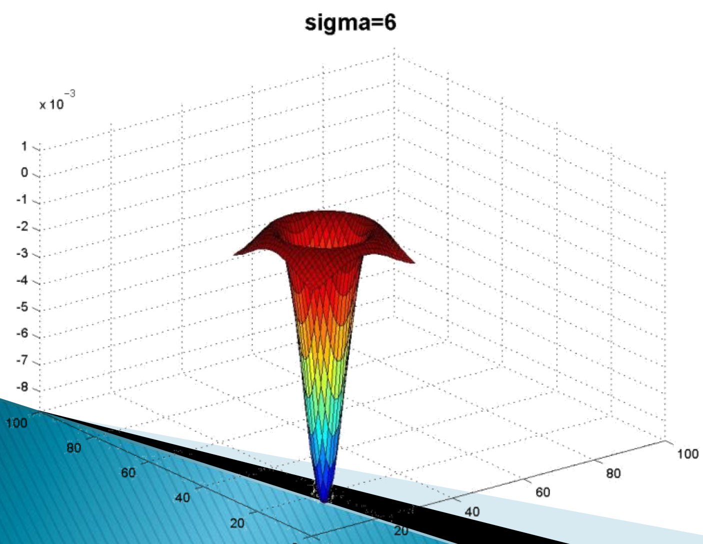




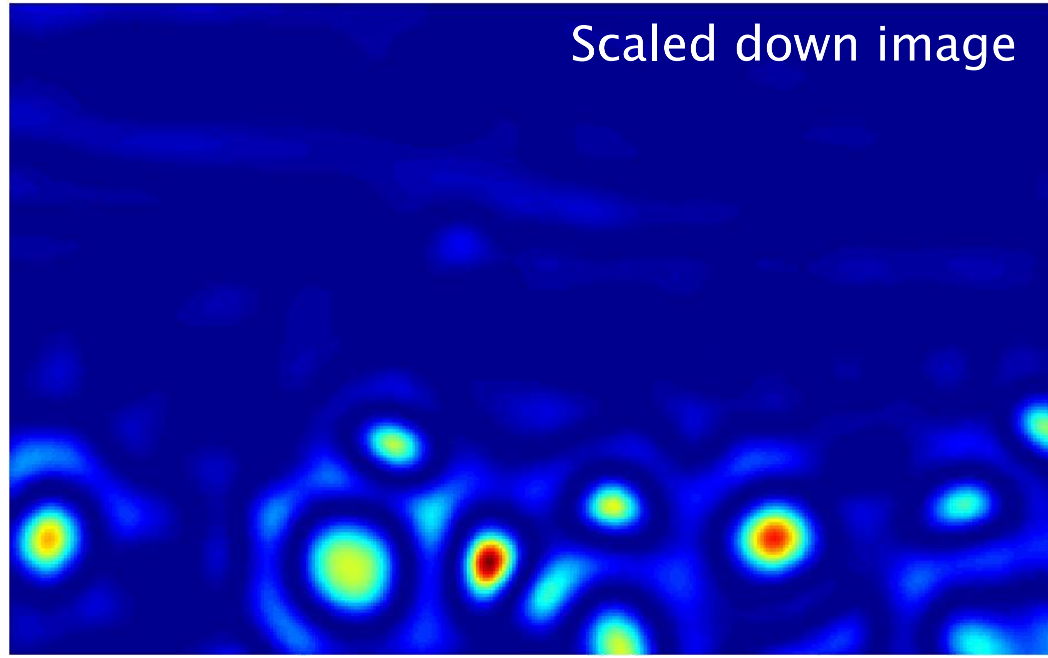
Slide credit: Kristen Grauman

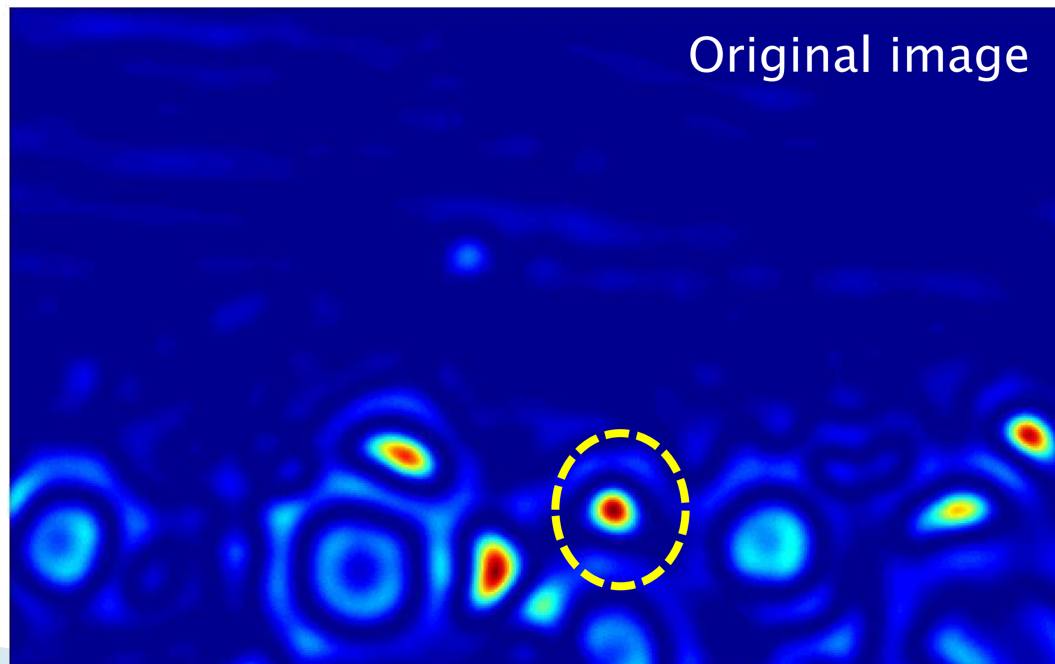


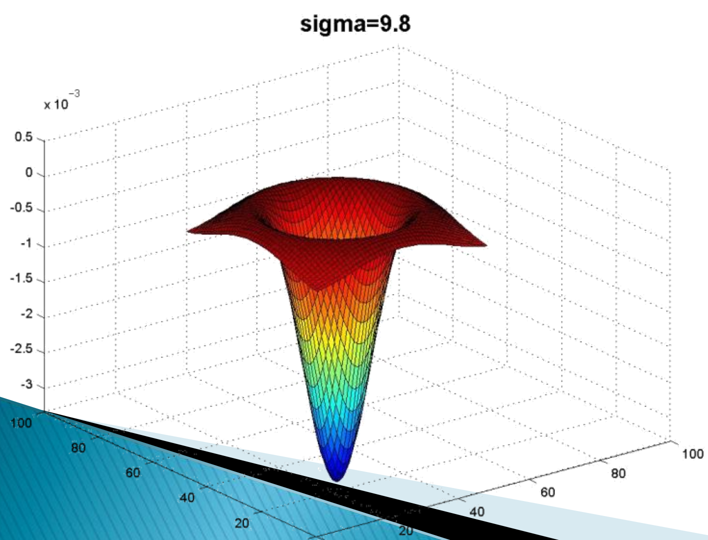




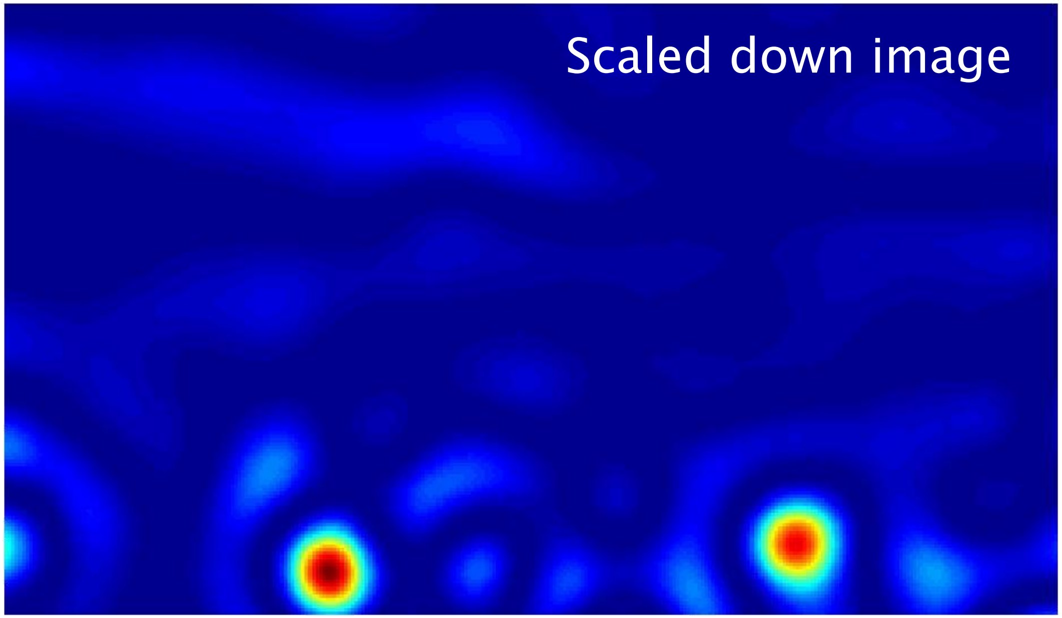
Slide credit: Kristen Grauman

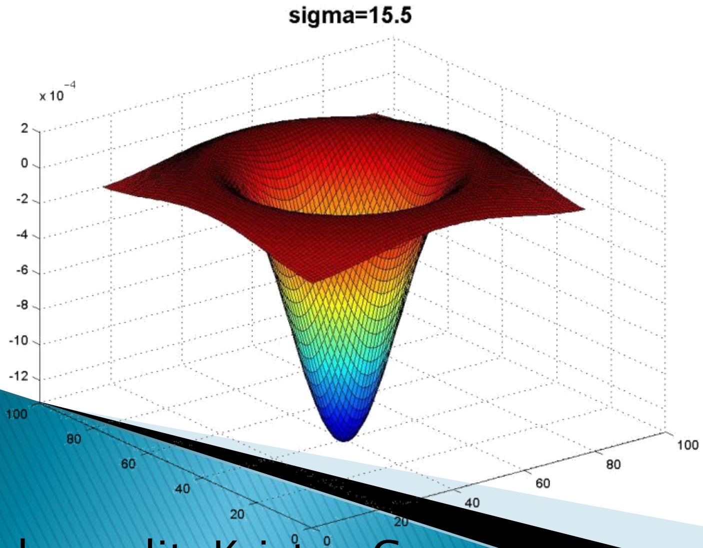


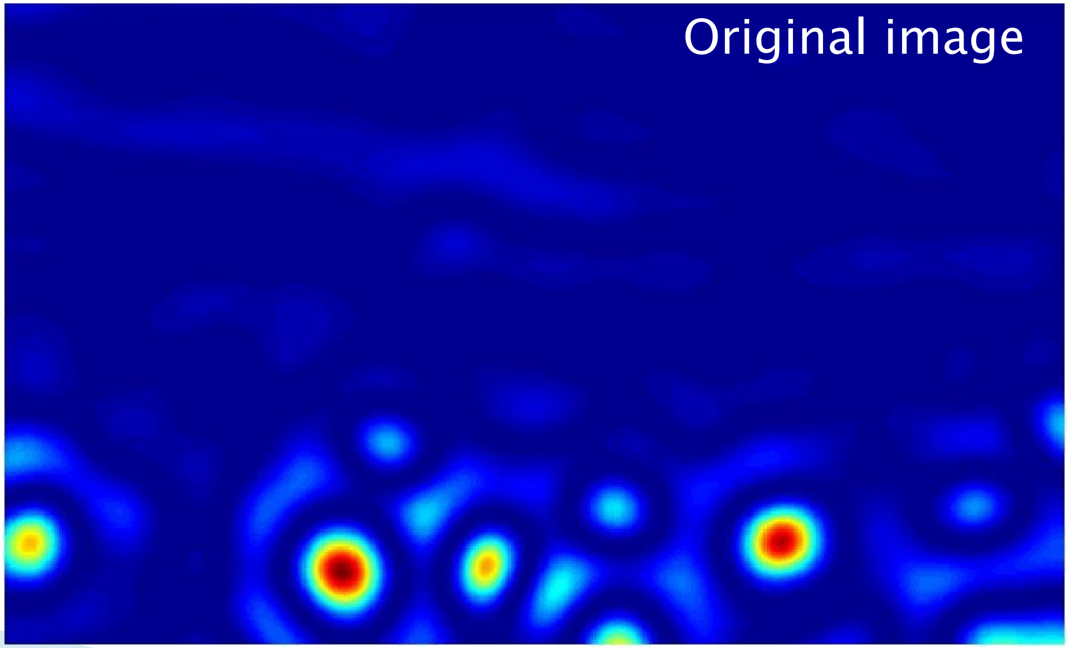




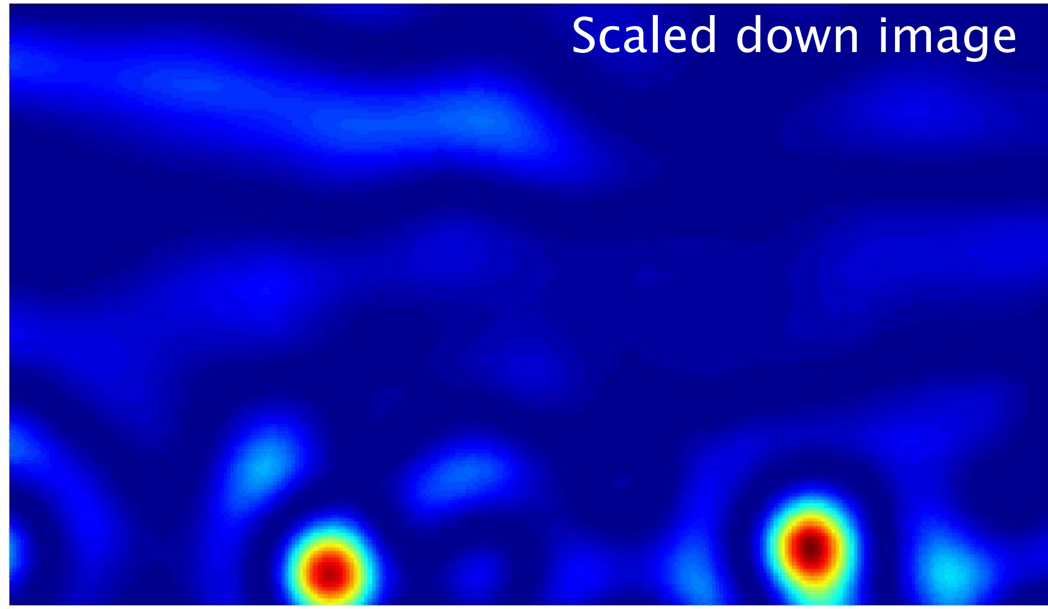
Slide credit: Kristen Grauman

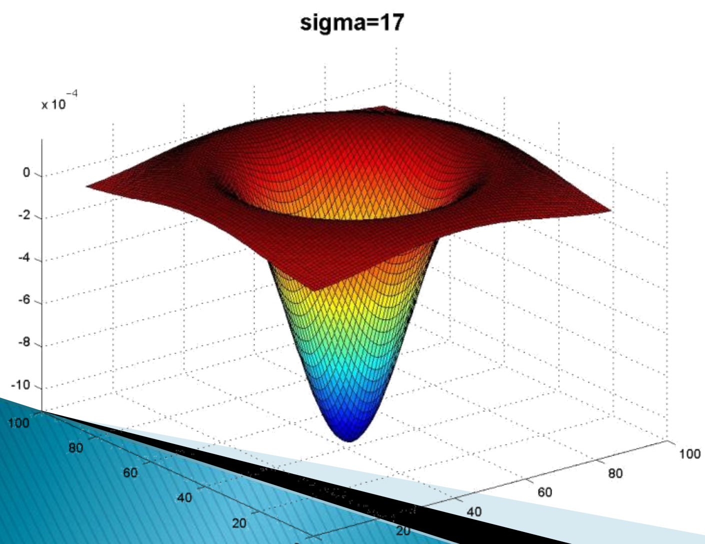


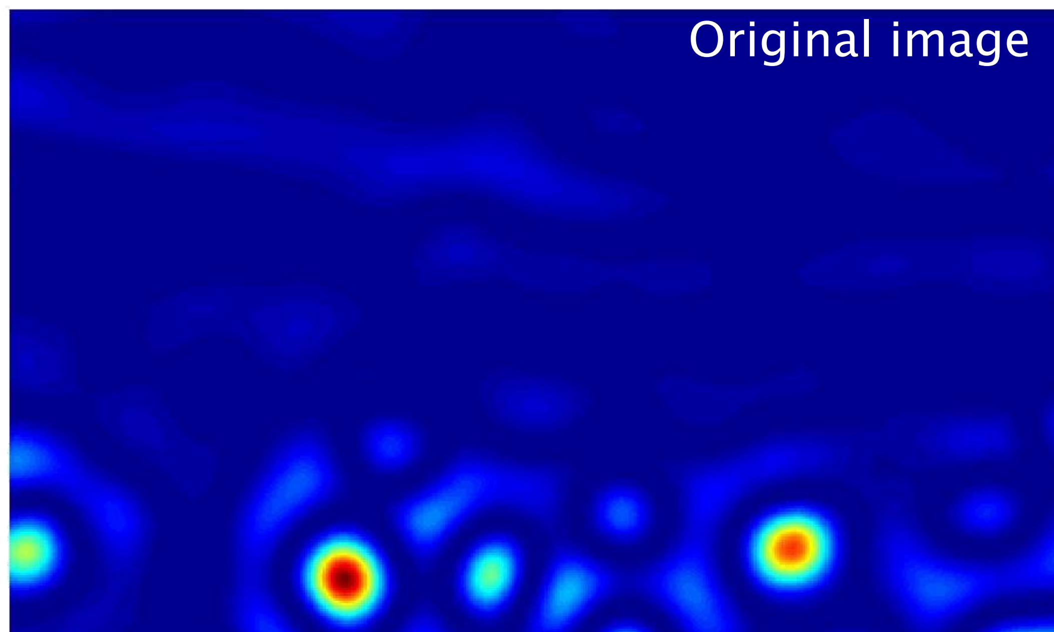




Slide credit: Kristen Grauman





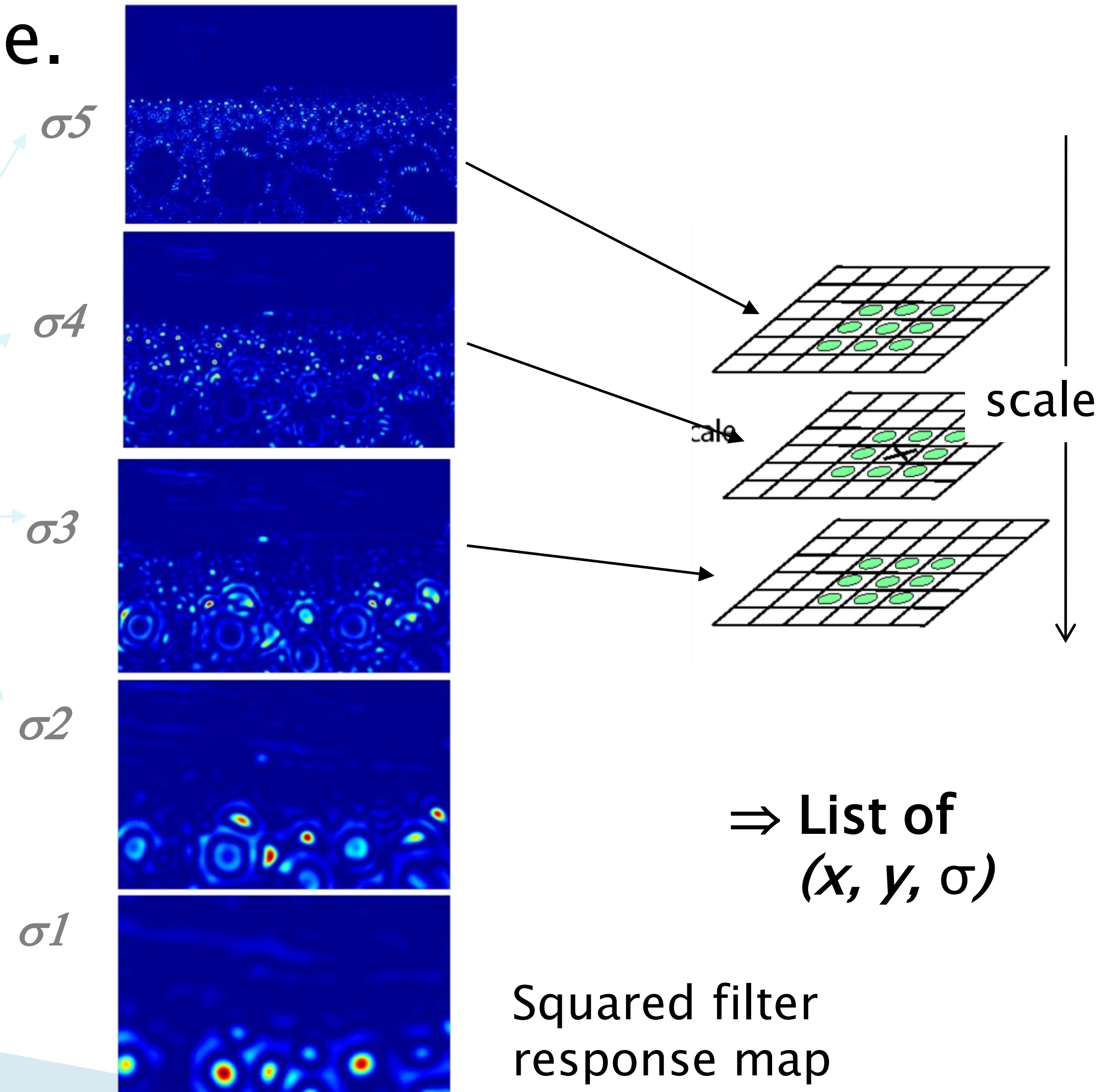


Slide credit: Kristen Grauman

Scale invariant interest points

Interest points are local maxima in both po
sition and scale.

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



$\Rightarrow$ **List of *(x, y, σ)***

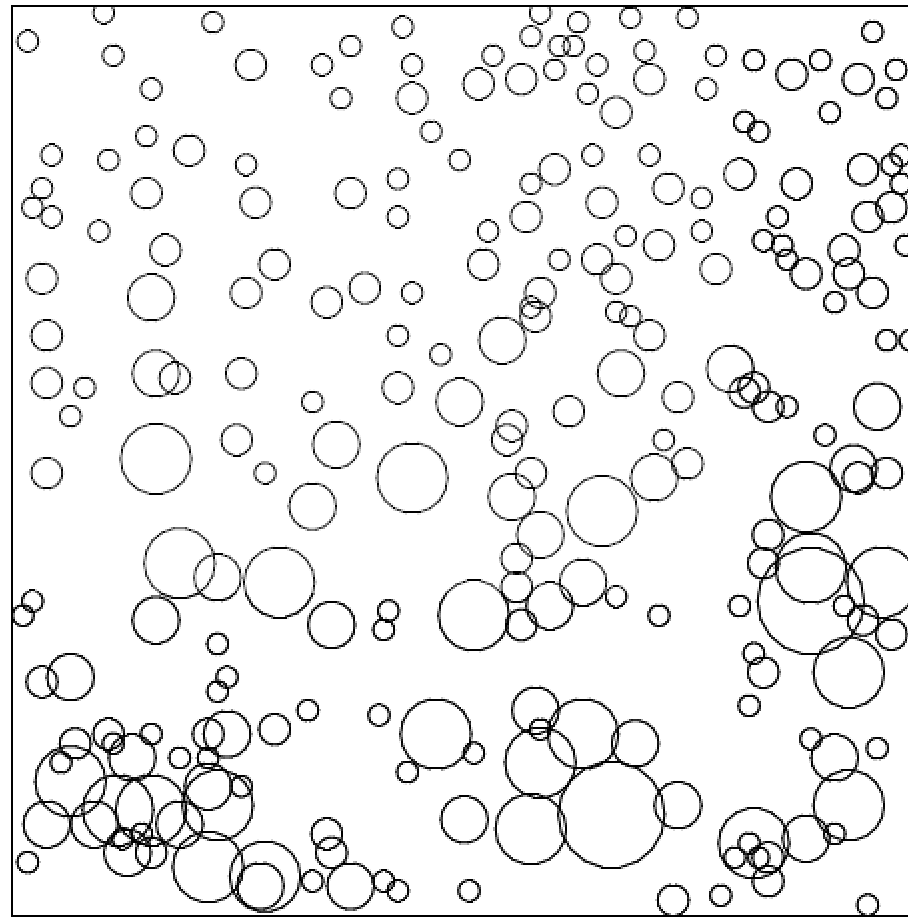
Squared filter response map
s

Slide credit: Kristen Grauman

Scale-space blob detector: Example

original image

scale-space maxima of $(\nabla^2_{norm} L)^2$



T. Lindeberg. Feature detection with automatic scale selection. IJCV 1998.

Scale-space blob detector: Example

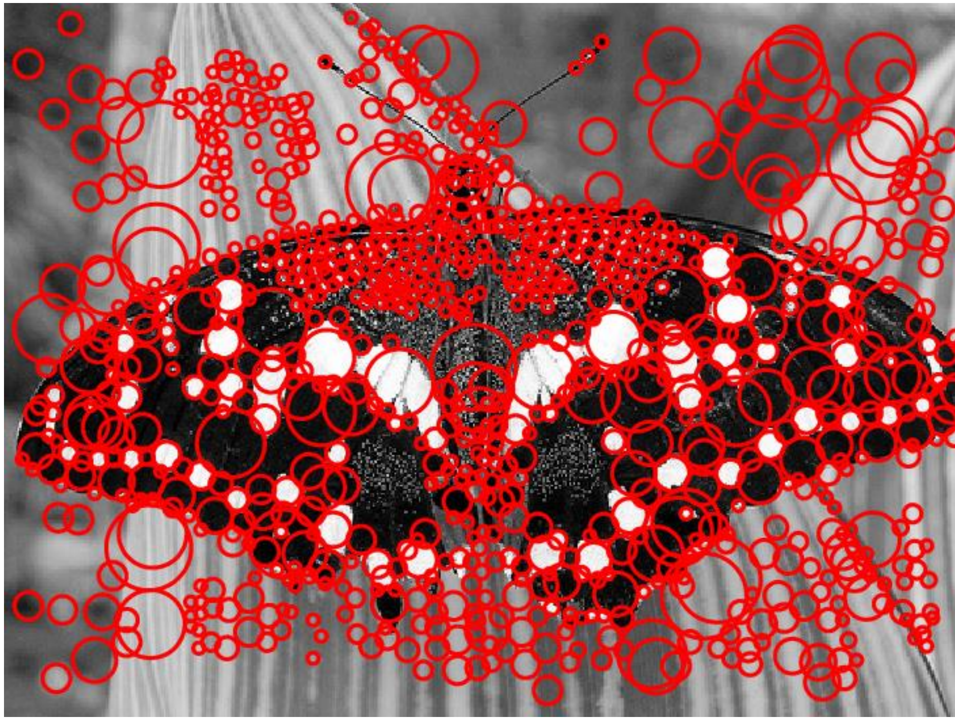


Image credit: Lana Lazebnik

Technical detail

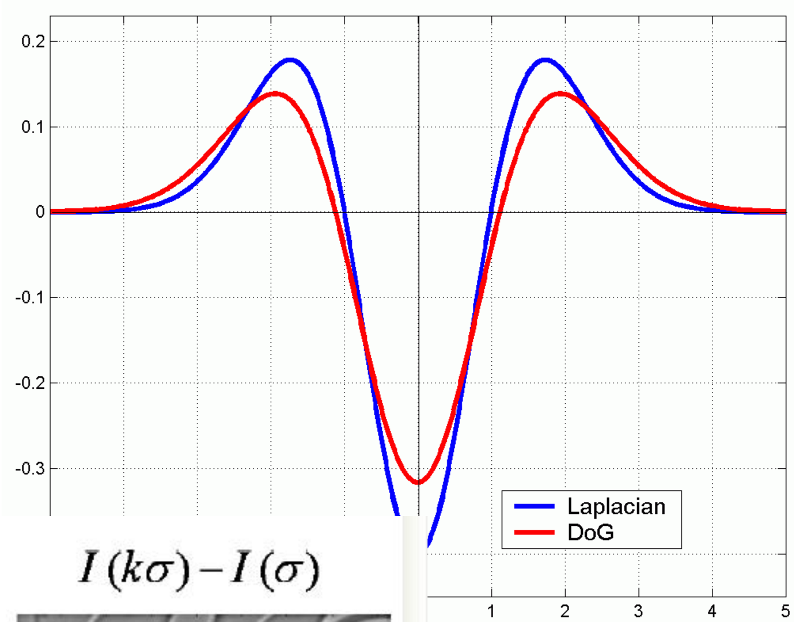
- We can approximate the Laplacian with a differen ce of Gaussians; more efficient to implement.

$$L = \sigma^2 \left(G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma) \right)$$

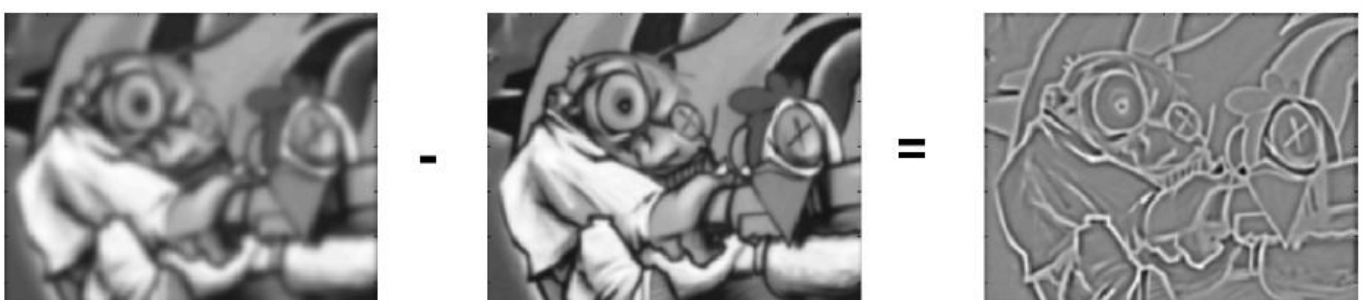
(Laplacian)

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

(Difference of Gaussians)



$I(k\sigma)$ $\quad$ $I(\sigma)$ $\quad$ $I(k\sigma) - I(\sigma)$



Slide credit: Kristen Grauman

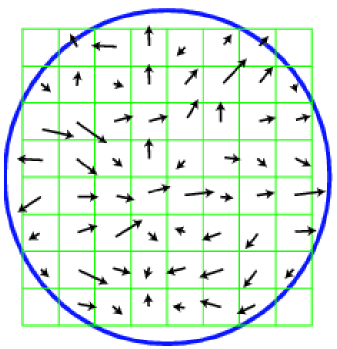
Feature Point Description

- Now we have feature points.
- How do we describe feature points for matching?

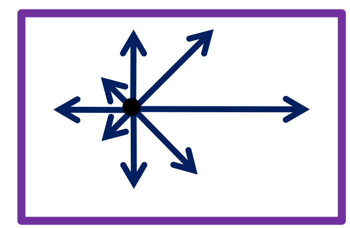
Descriptor

Scale Invariant Feature Transform (SIFT) **descriptor**

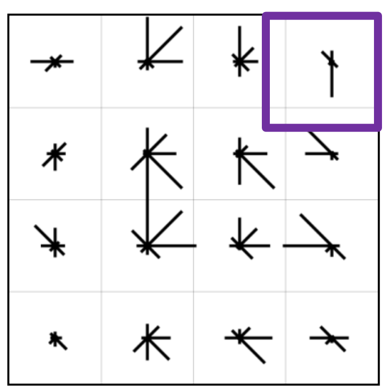
- Use histograms to bin pixels within sub-patches according to their orientation.



gradients



subdivided local patch

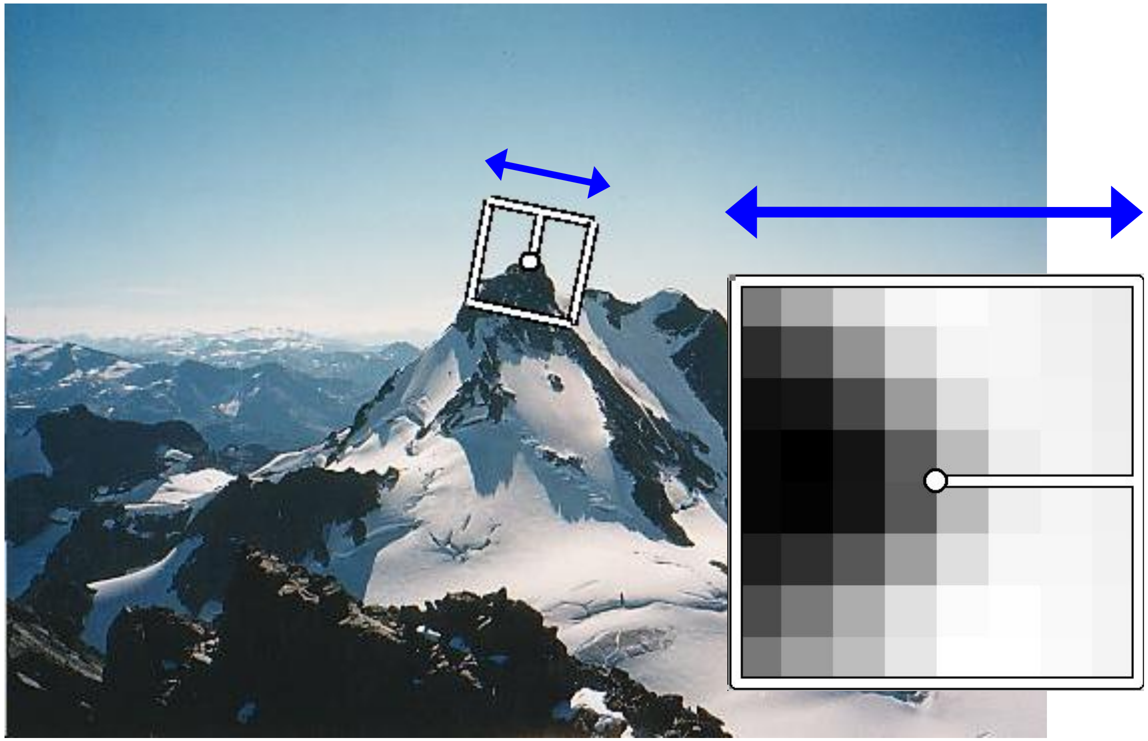


histogram per grid cell

Final descriptor = concatenation of all histograms, normalize

Slide credit: Kristen Grauman

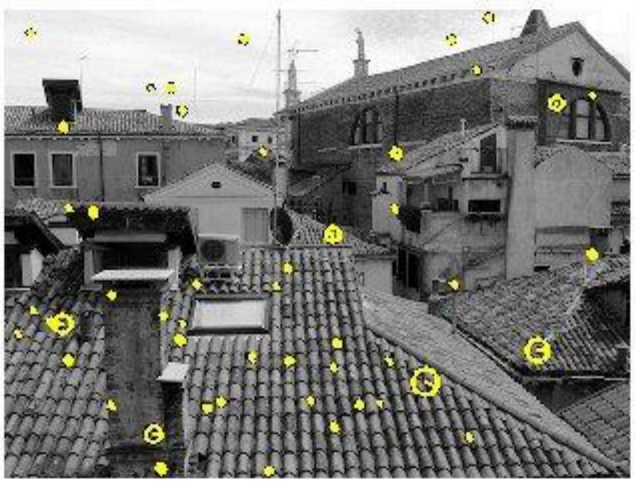
Making descriptor rotation invariant



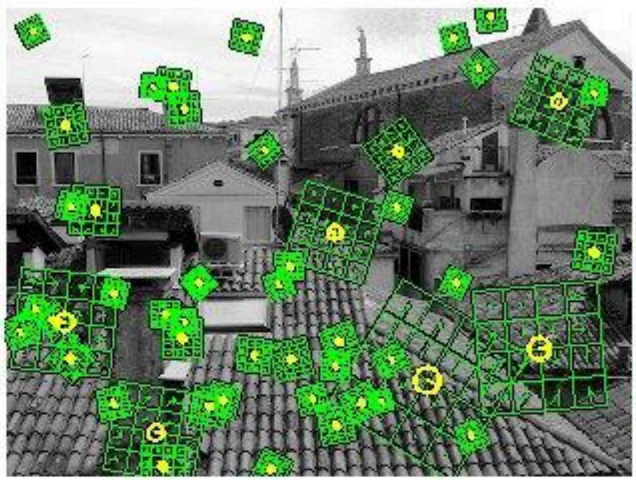
- Rotate patch according to its dominant gradient orientation
- This puts the patches into a canonical orientation.

Image from Matthew Brown

Scale Invariant Feature Transform (SIFT) descriptor [Lowe 2004]



Interest points and their
scales and orientations
(random subset of 50)



SIFT descriptors

http://www.vlfeat.org/overview/sift.html

Feature Points Matching

Matching local features

Slide credit: Kristen Grauman

Matching local features

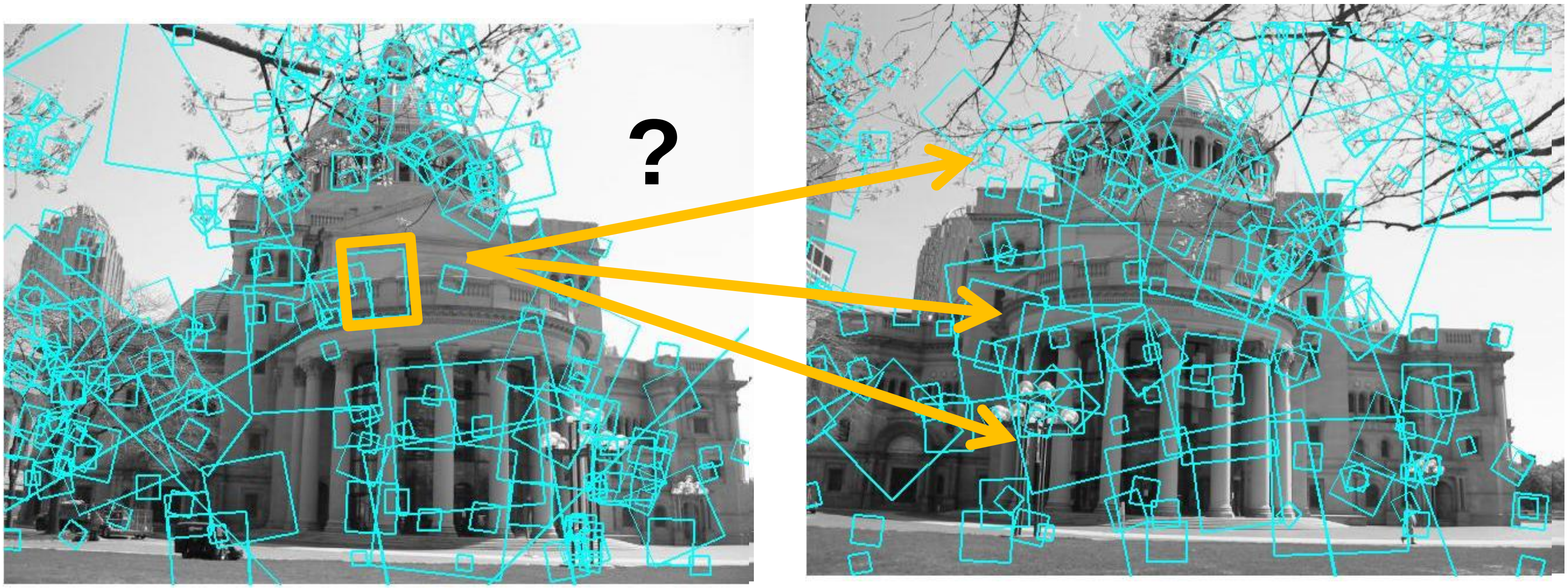


Image 1

Image 2

- To generate **candidate matches**, find patches that have the most similar appearance
- Simplest approach: compare them all, take the closest (or closest k, or within a thresholded distance)

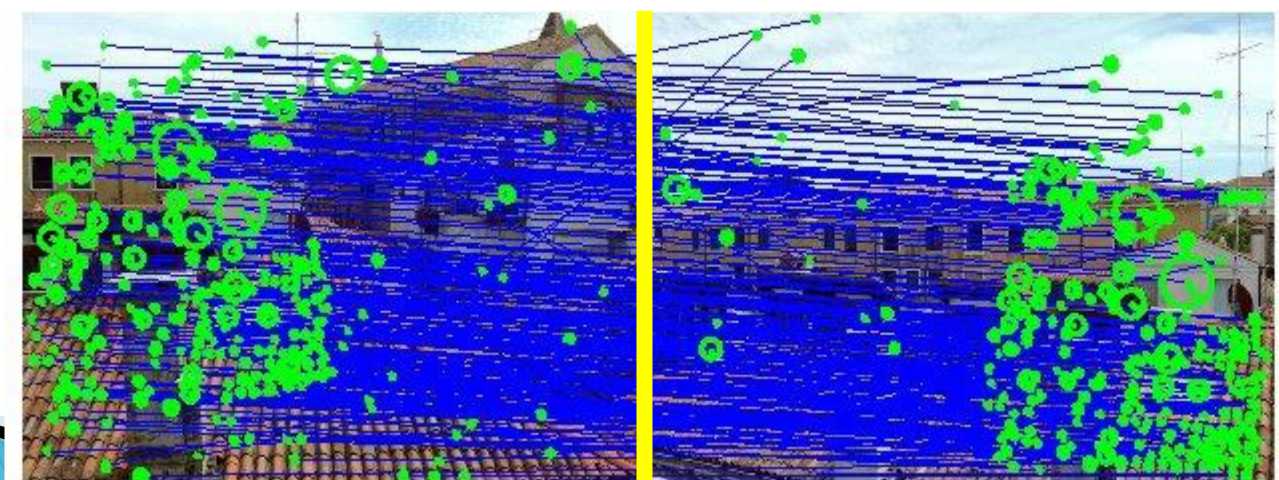
Slide credit: Kristen Grauman

SIFT (preliminary) matches

img1

img2

img1



img2

http://www.vlfeat.org/overview/sift.html