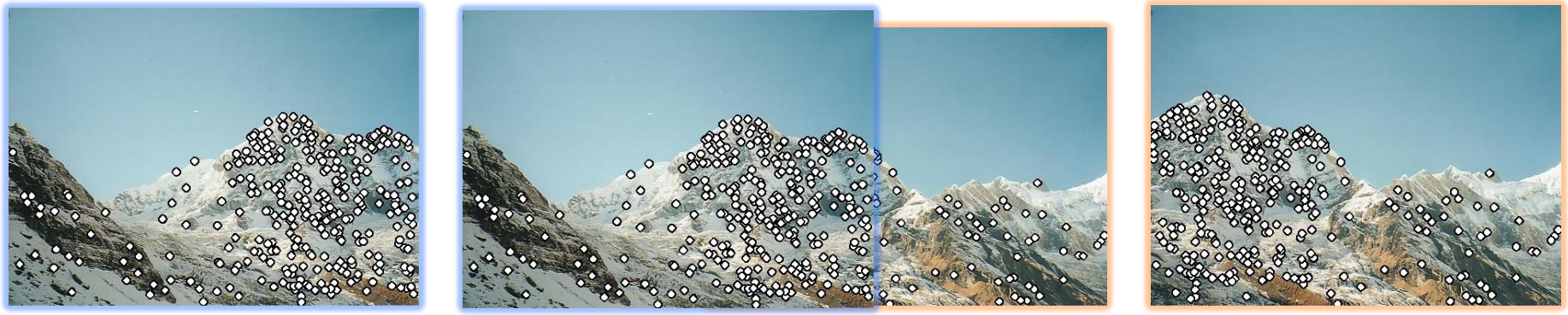
CS 4476 Computer Vision:

5. Local Feature Detection

Humphrey Shi

9/3/2024



The follow slides are mostly based on the lecture slides of Judy Hoffman and Svetlana Lazebnik, with credits to Devi, Kristen, etc.

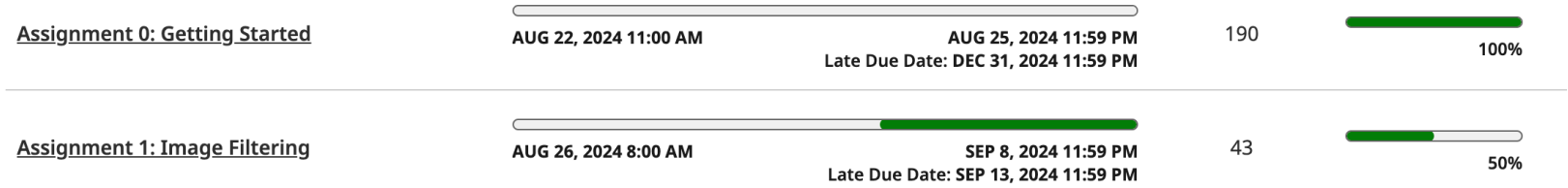
Outline

- Logistics & Course Recap
- Local Feature Detection

Logistics & Course Recap

Logistics

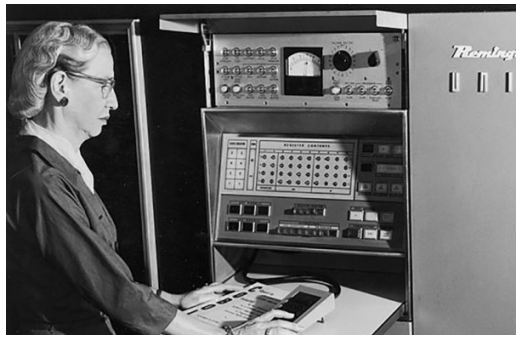
- Assignment 1 – Due this Sunday.
- Estimated weekly effort ~ 10-15hrs.



Logistics

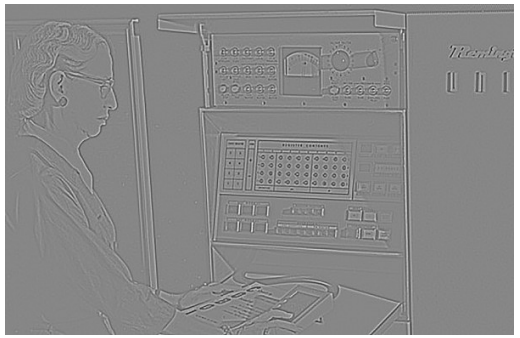
- Schedule for TA office hours
  - Monday 10 am to 12 pm: Anh, Fengzhe
  - Tuesday 9 am to 11 am: Dhruv,
  - Wednesday 3pm to 5pm: Chieh, Yue
  - Thursday 9am to 11am: Aditya, Shubham
  - Friday 9am to 11 am: Manushree, Bogi

Recap: Image Filtering (continued)



Image

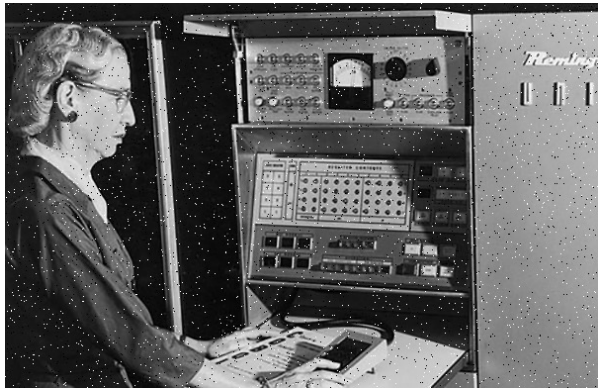
+α



Details

=

“Sharpened” α=2

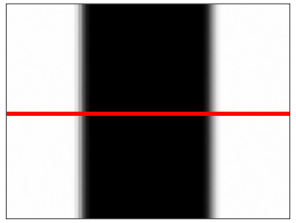


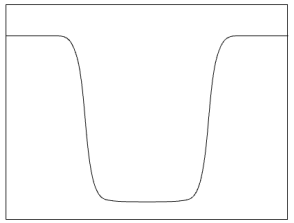
Noisy image

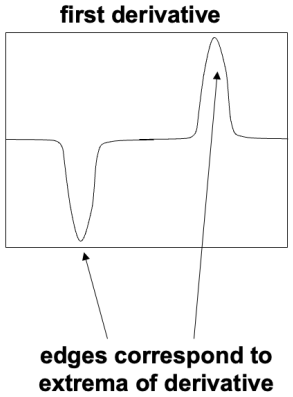
Gaussian filter, σ = 1

Median filter, size = 3

Recap: Gradients & Edges

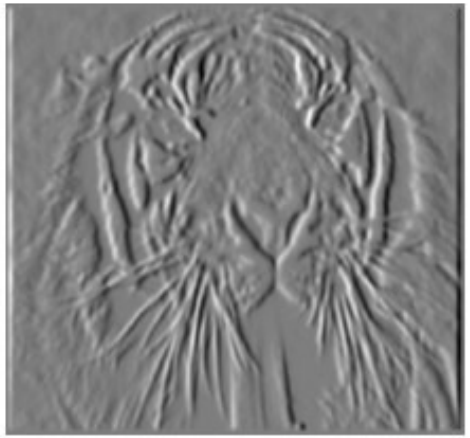


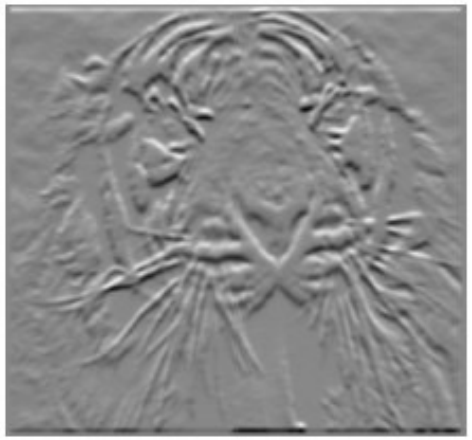




$$\frac{\partial f(x,y)}{\partial x}$$

-1	1





$$\frac{\partial f(x,y)}{\partial y}$$

1
-1

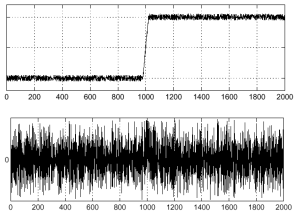
$$M_x = \begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

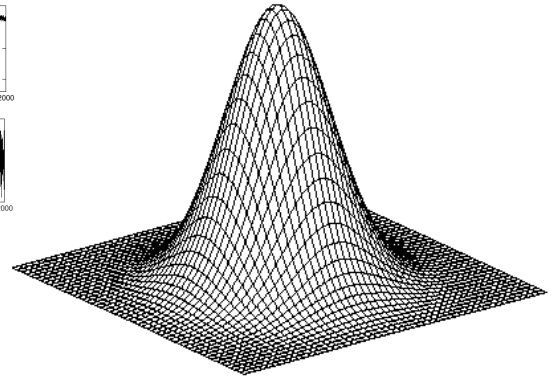
$$M_y = \begin{array}{|c|c|c|}\hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

Sobel Filter

Recap: Edge Detection

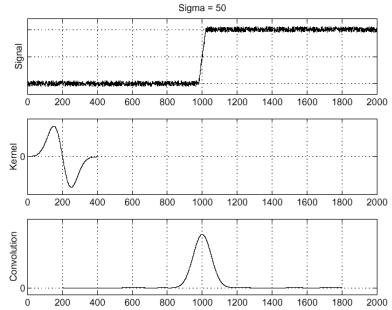
- 2D Edge Detection Filters

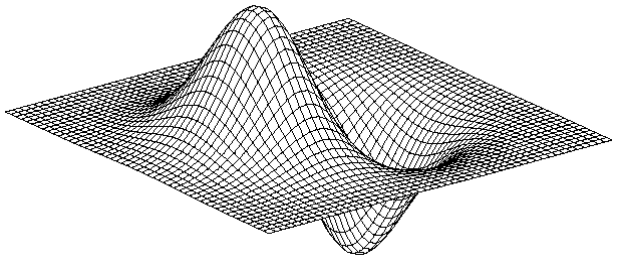




Gaussian

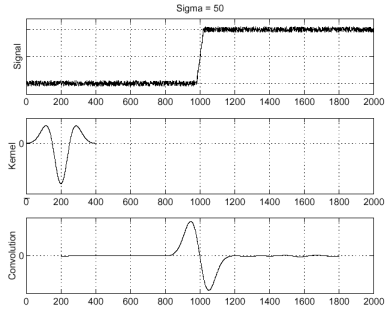
$$h_\sigma(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$



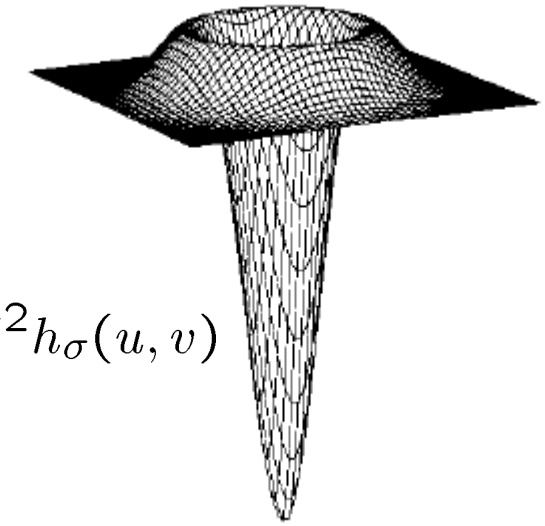


derivative of Gaussian

$$\frac{\partial}{\partial x} h_\sigma(u,v)$$



Laplacian of Gaussian



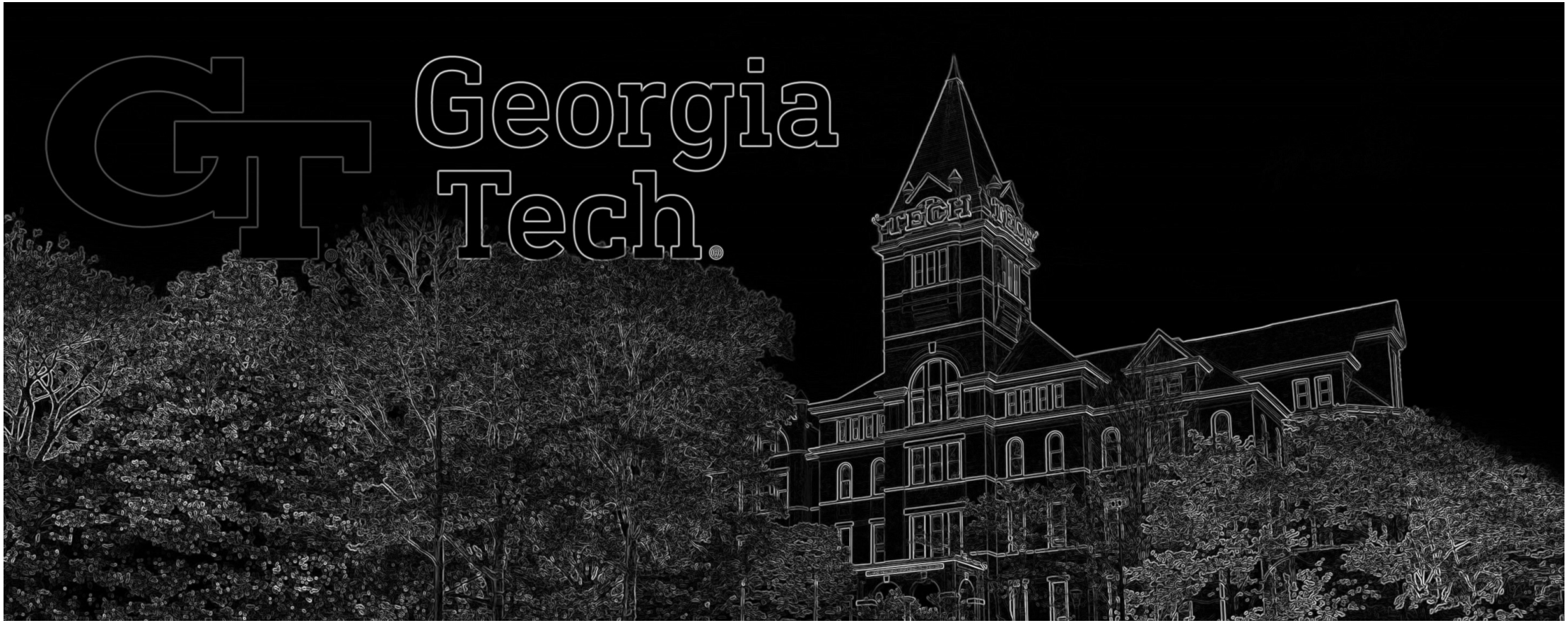
$$\nabla^2 h_\sigma(u,v)$$

- Canny Edge Detector
  - How?

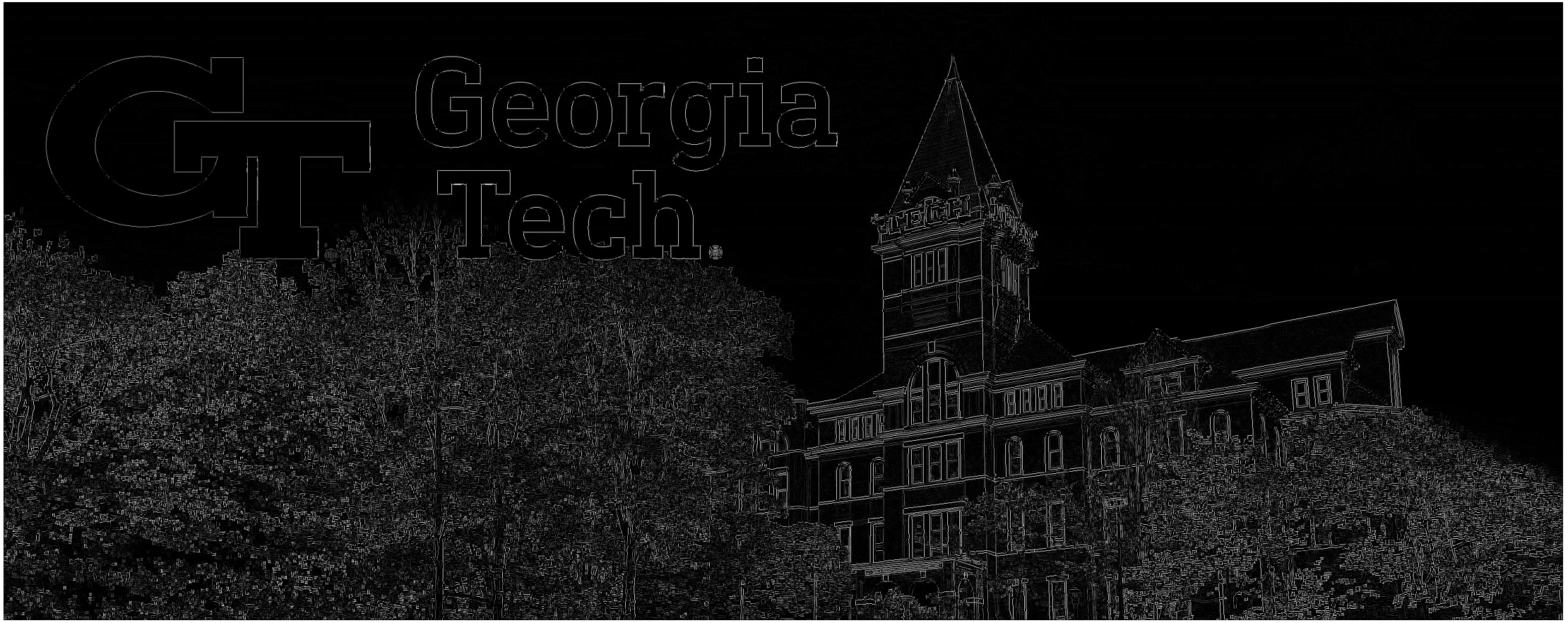
GT

Noise Reduction by Smoothing

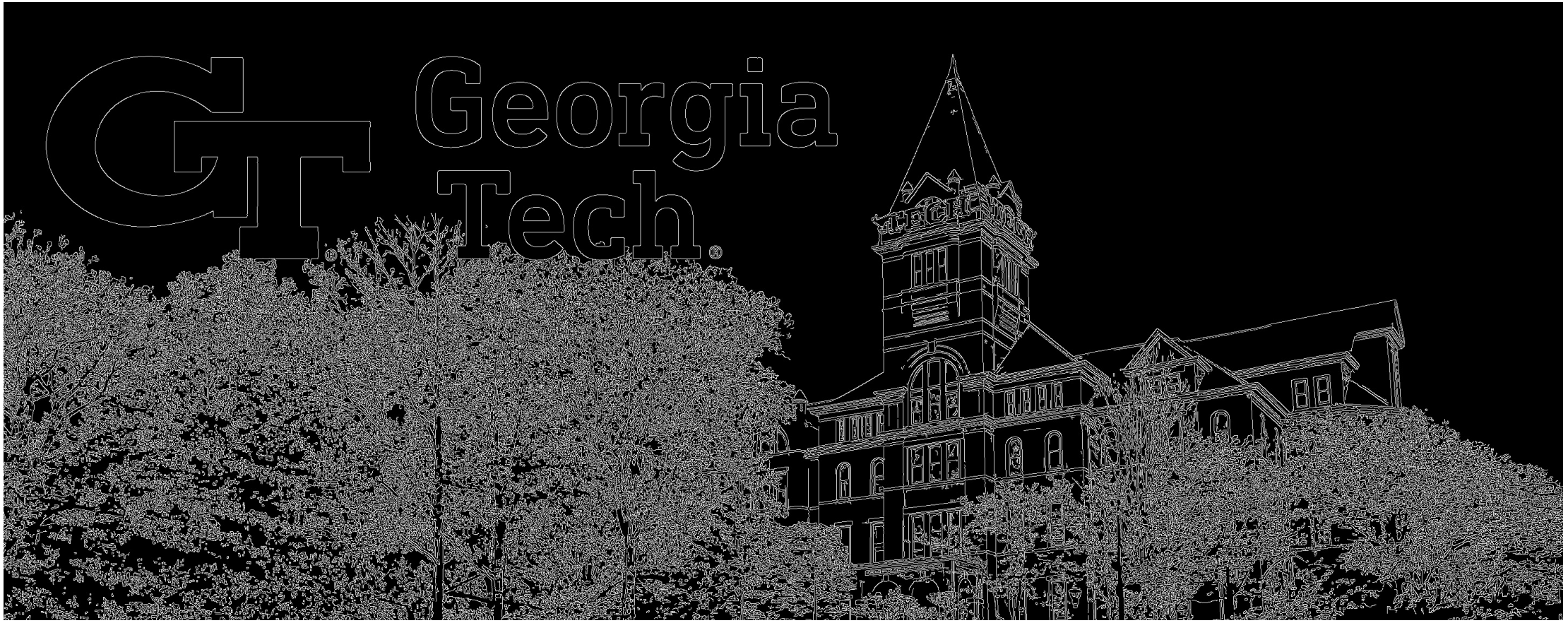
Finding Image Gradients



Non-Maximum Suppression to Thin Edges



Edge Tracking by Hysteresis Thresholding



GT

Local Feature Detection

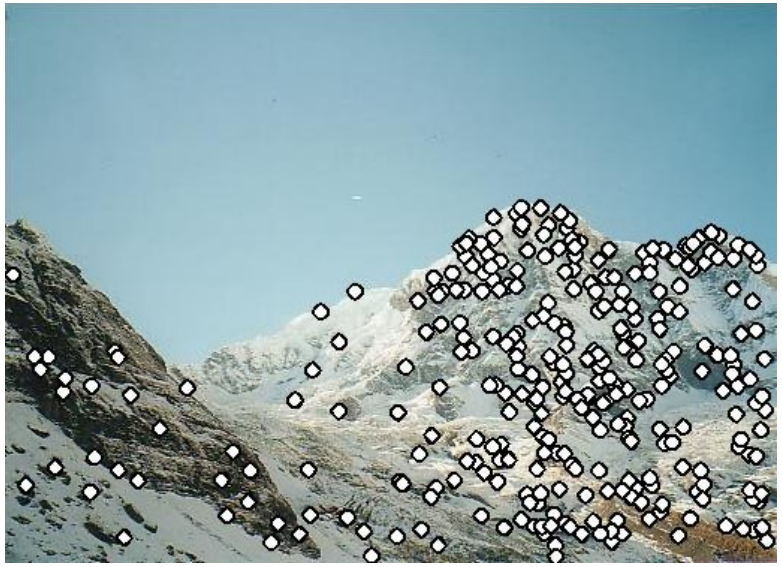
Image Alignment & Local Features

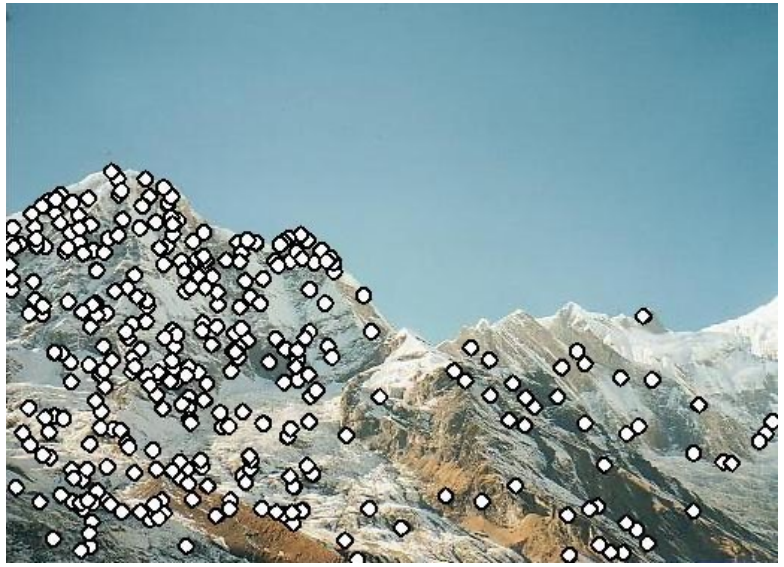
Image Alignment

Question: How do we combine the two images?

Slide credit: Judy Hoffman adapted from Lana Lazebnik

Robust feature-based alignment

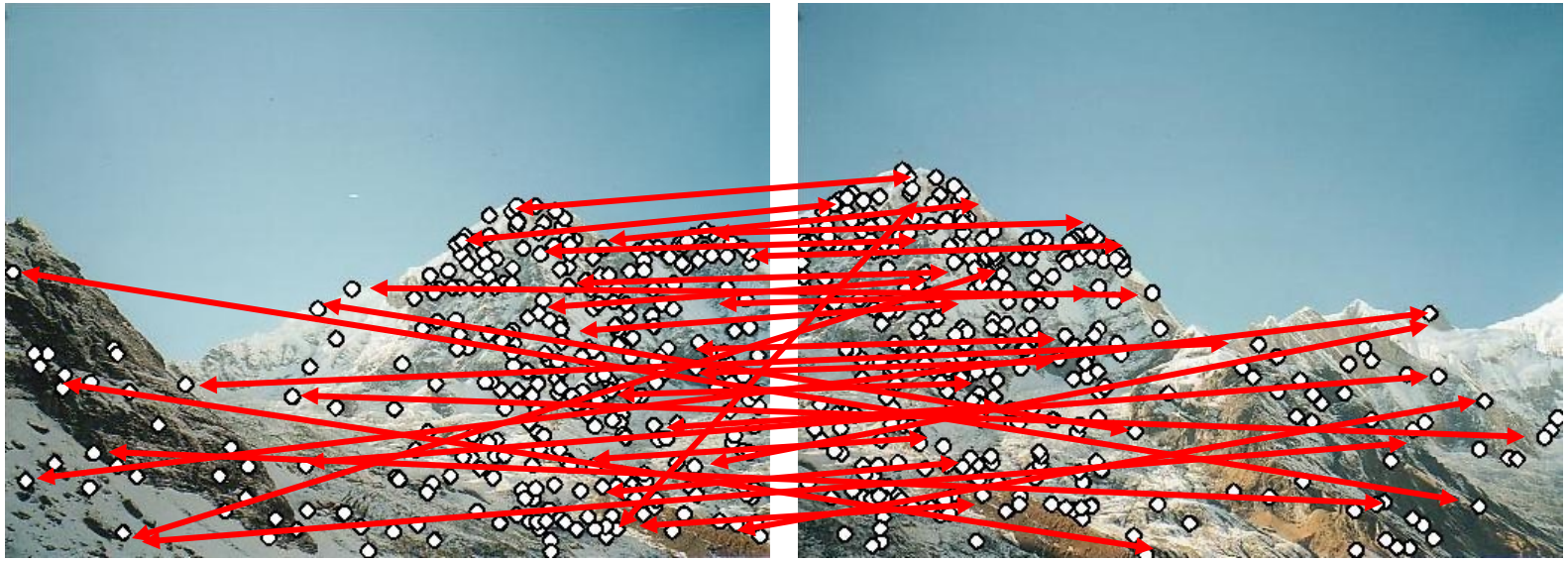




- Extract keypoint features

Slide credit: Judy Hoffman adapted from Lana Lazebnik

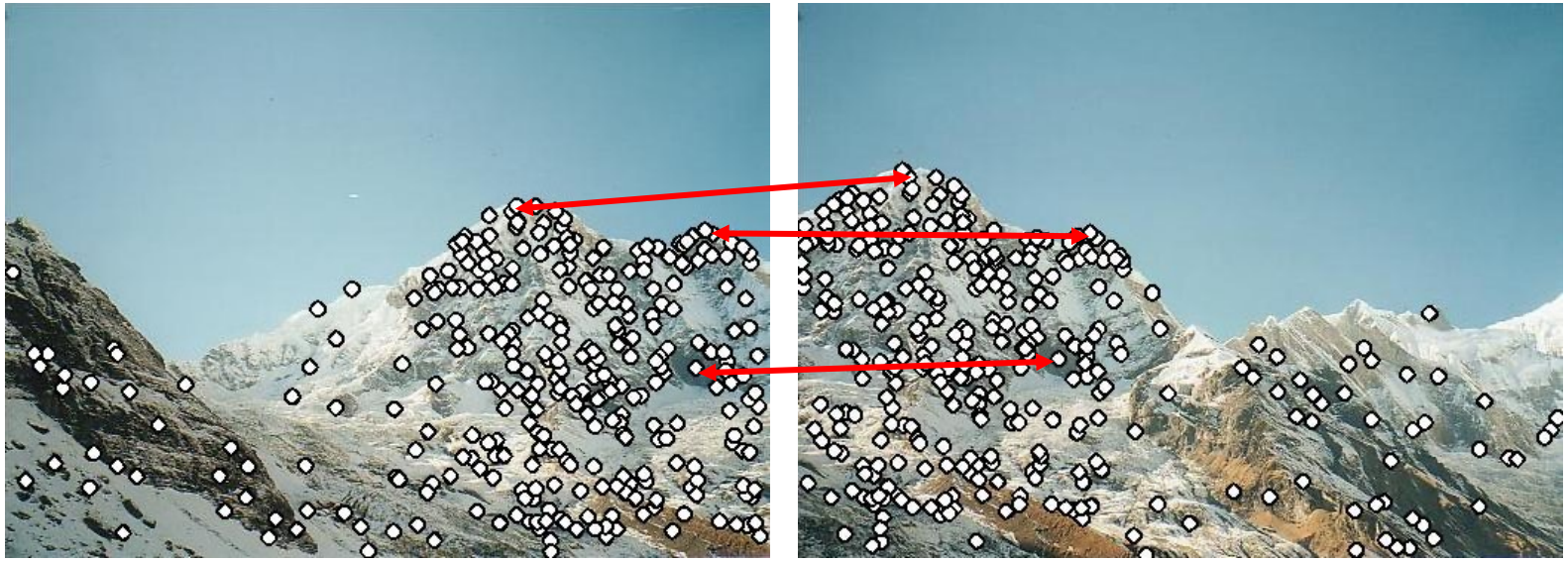
Robust feature-based alignment



- Extract keypoint features
- Compute *potential matches*

Slide credit: Judy Hoffman adapted from Lana Lazebnik

Robust feature-based alignment



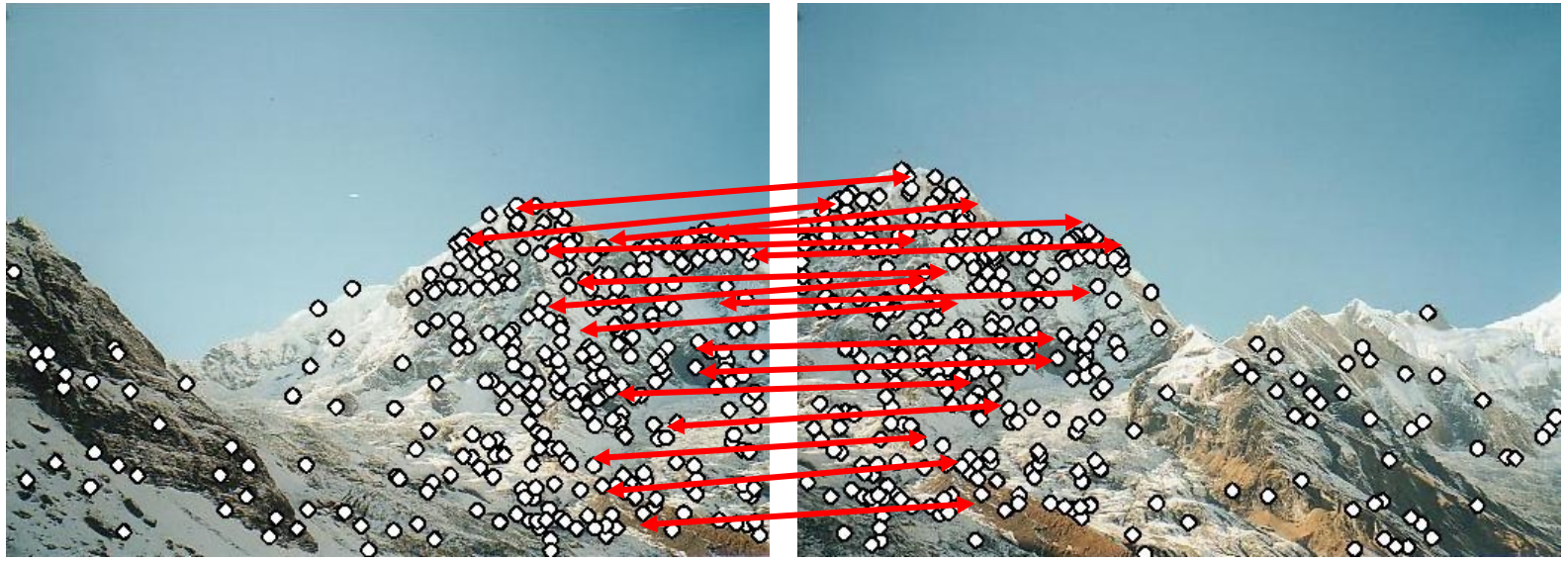
- Extract keypoint features
- Compute *potential matches*

Loop:

Hypothesize transformation *T* (small group of potential matches that are related by *T*)

Slide credit: Judy Hoffman adapted from Lana Lazebnik

Robust feature-based alignment



- Extract keypoint features
- Compute *potential matches*

Loop:

Hypothesize transformation *T* (small group of potential matches that are related by *T*)

Verify transformation (search for other matches consistent with *T*)

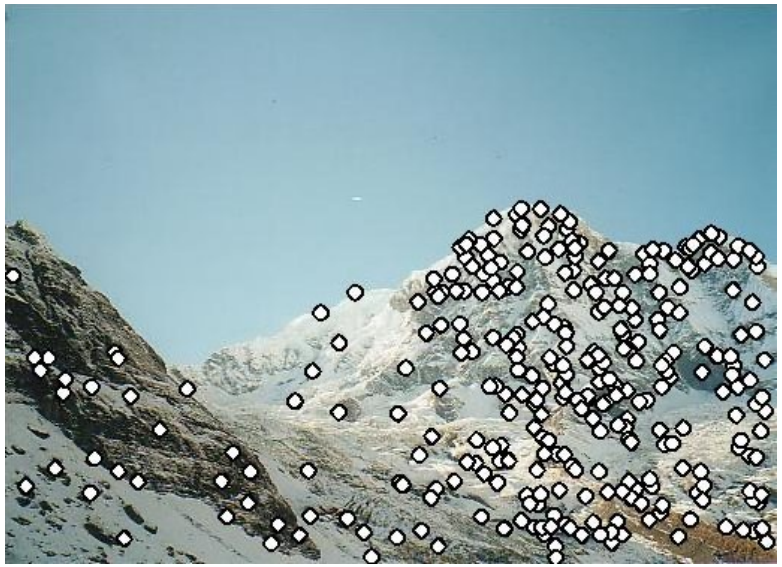
Slide credit: Judy Hoffman adapted from Lana Lazebnik

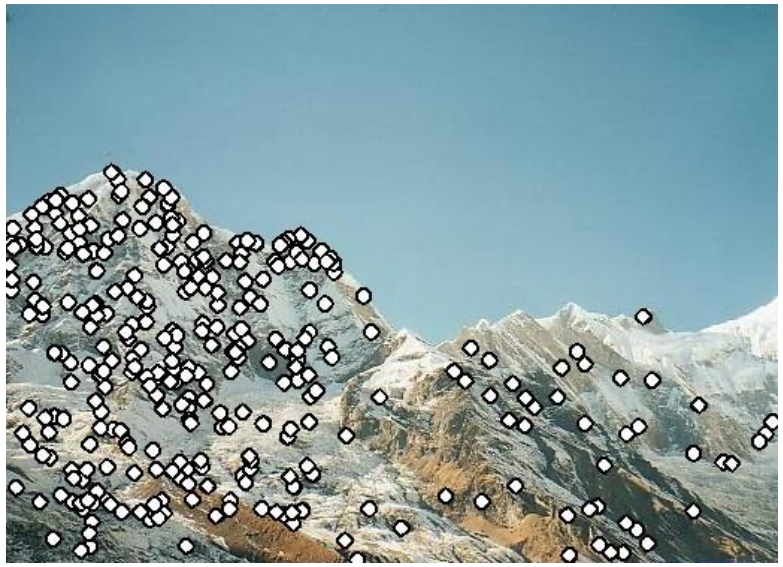
Robust feature-based alignment

- Extract keypoint features
- Compute *potential matches*
- Align the images

Slide credit: Judy Hoffman adapted from Lana Lazebnik

Today: Detecting Features





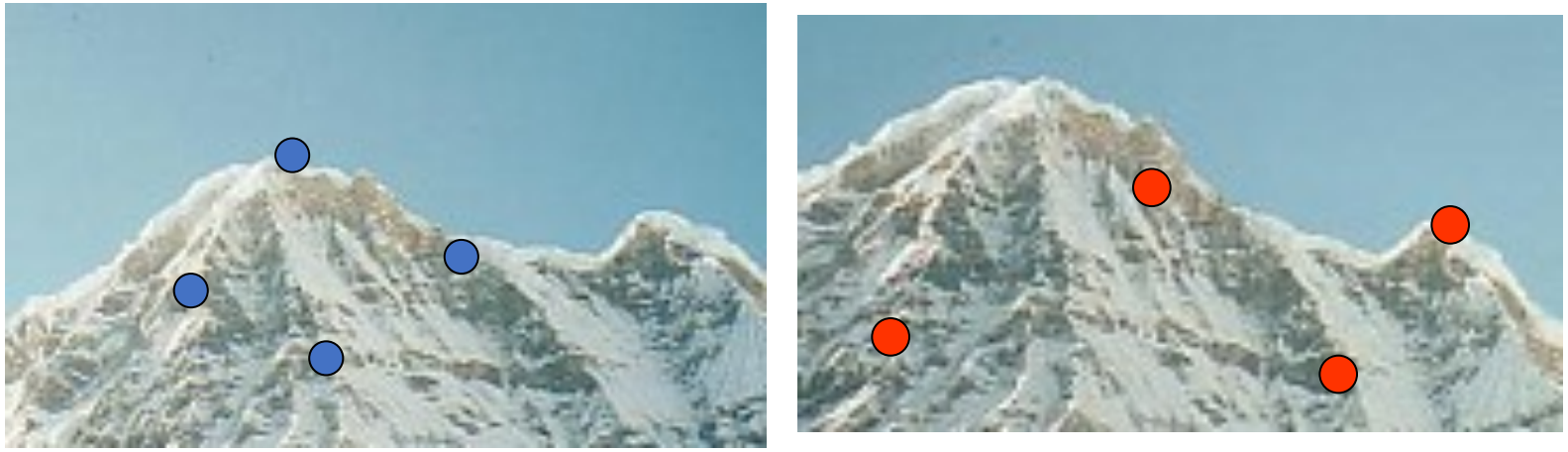
How to detect *which features* to match?

Local features: desired properties

- 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

Goal: interest operator repeatability

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

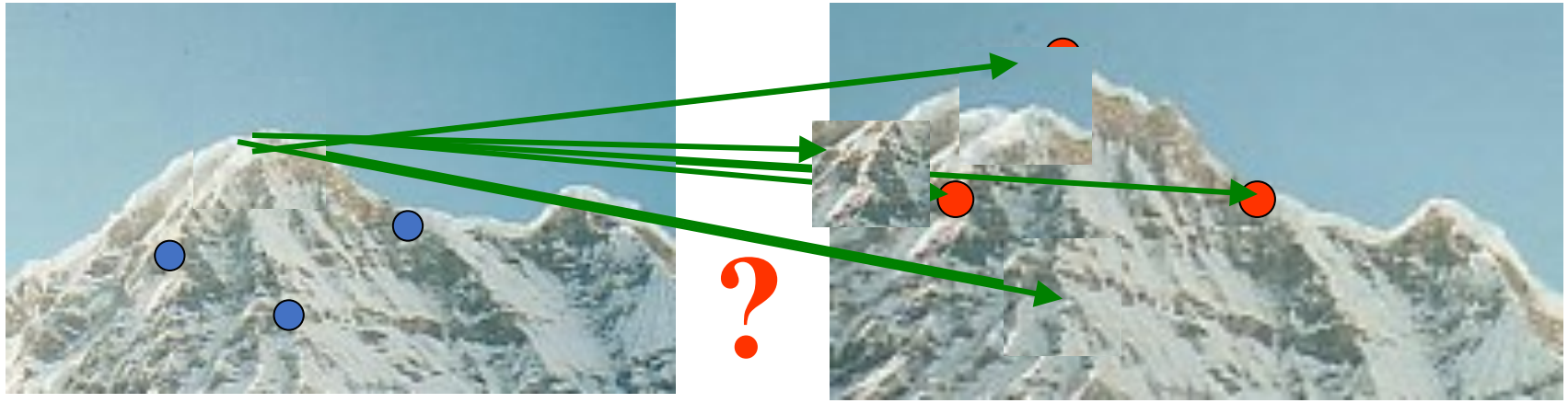


No chance to find true matches!

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

Goal: descriptor 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.

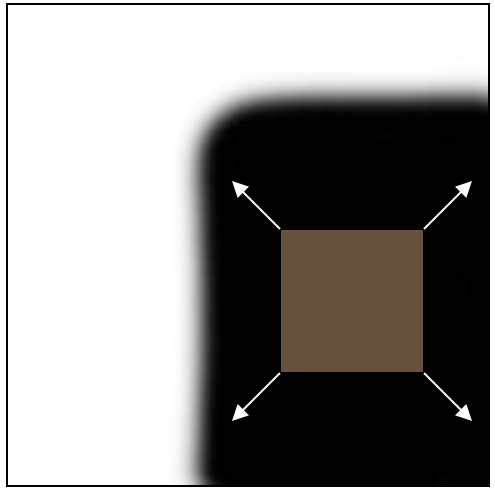
- Question: What keypoints should we choose?

Local Feature Detection

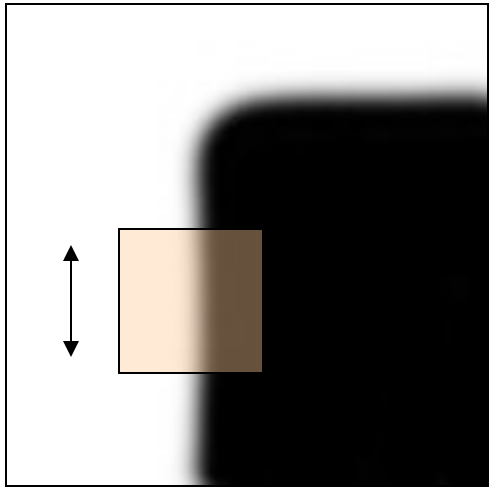
Corners as Interest Points

Corners as distinctive interest points

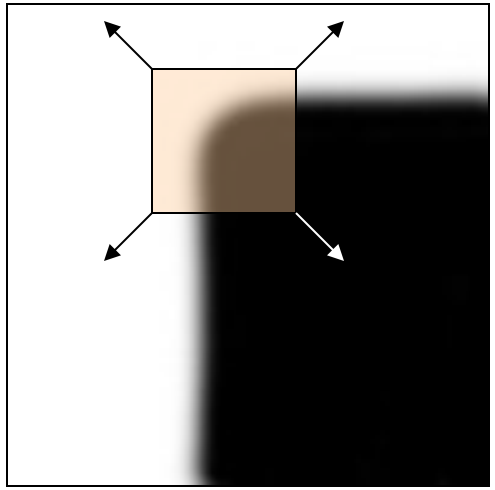
- We should easily recognize the point by looking through a small window
- Shifting a window in *any direction* should give *a large change* in intensity



"flat" region:
no change in
all directions



"edge":
no change
along the edge
direction

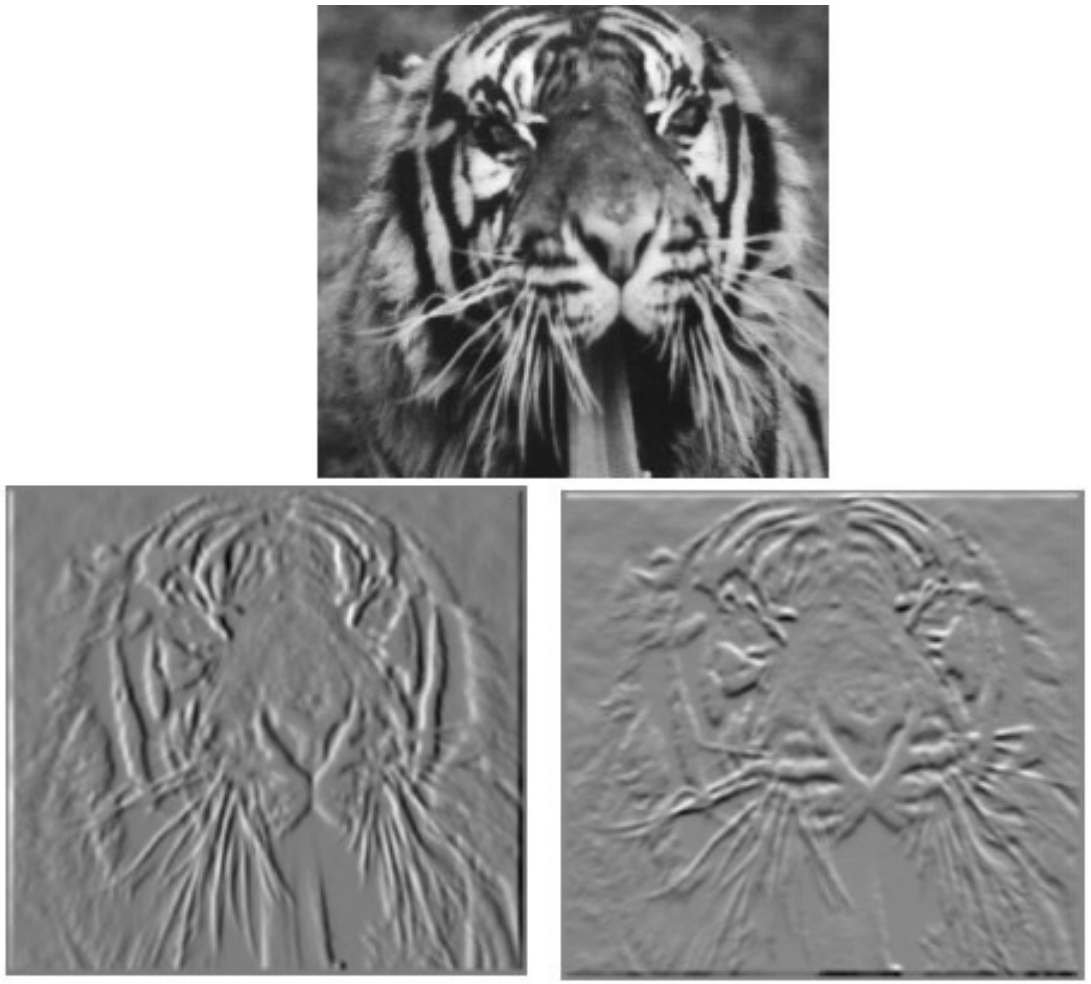


"corner":
significant
change in all
directions

Slide credit: Alyosha Efros, Darya Frolova, Denis Simakov

Recall: Partial derivatives measure changes along direction

$$I_x = \frac{dI}{dx} = \frac{\partial f(x,y)}{\partial x}$$



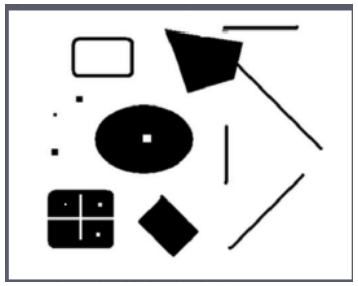
$$I_y = \frac{dI}{dy} = \frac{\partial f(x,y)}{\partial y}$$

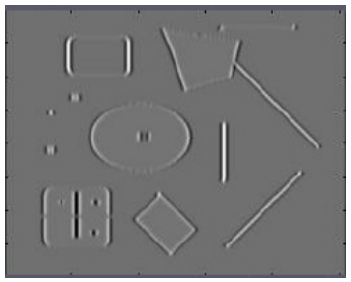
Slide credit: Judy Hoffman

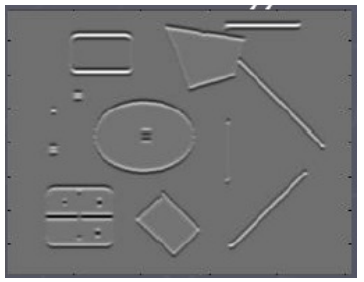
Corners as distinctive interest points

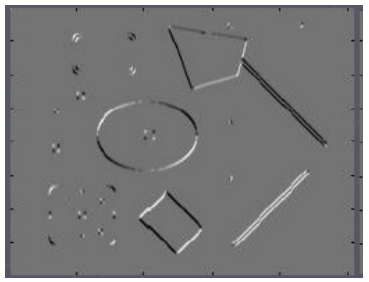
$$M = \sum_{(i,j)\in W} \begin{bmatrix} I_x(i,j)I_x(i,j) & I_x(i,j)I_y(i,j) \\ I_x(i,j)I_y(i,j) & I_y(i,j)I_y(i,j) \end{bmatrix}$$

2 x 2 matrix of image derivatives (averaged in neighborhood of a point (*i,j*)).







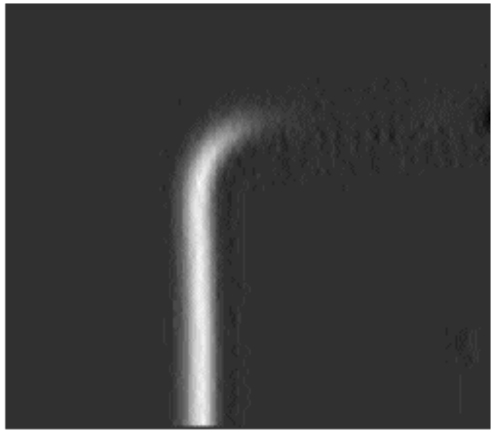


Notation:

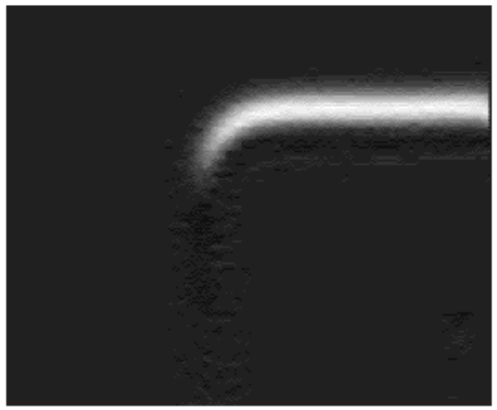
$$I_x \Leftrightarrow \frac{\partial I}{\partial x} \qquad I_y \Leftrightarrow \frac{\partial I}{\partial y} \qquad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$$

What does this matrix reveal?

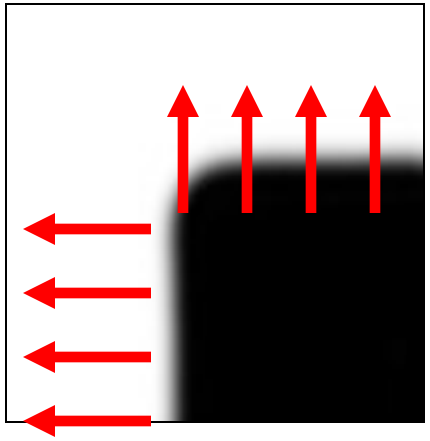
First, consider an axis-aligned corner:



I_x

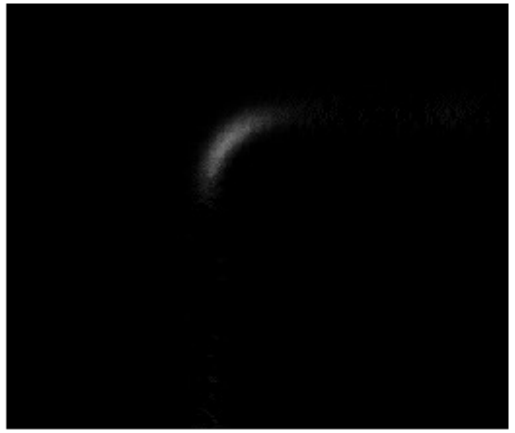


I_y



$$M = \sum_{(i,j)\in W} \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix}$$

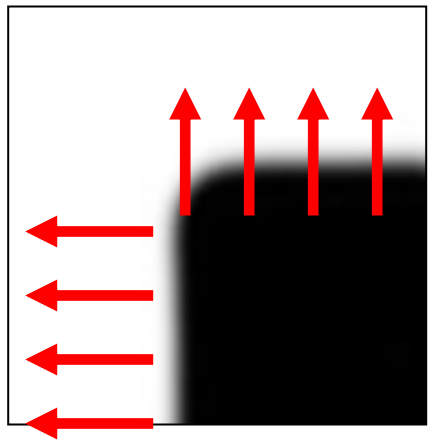
What will $I_x I_y$ look like?



Slide credit: Judy Hoffman

What does this matrix reveal?

First, consider an axis-aligned corner:



$$M = \sum_{(i,j)\in W} \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix} \approx \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

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?

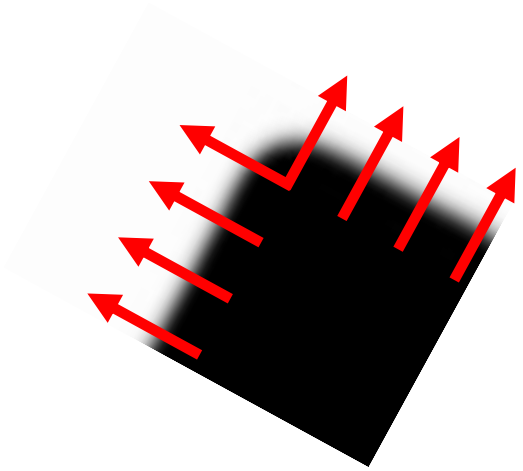
Slide credit: Judy Hoffman

What does this matrix reveal?

Since *M* is symmetric, we have

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

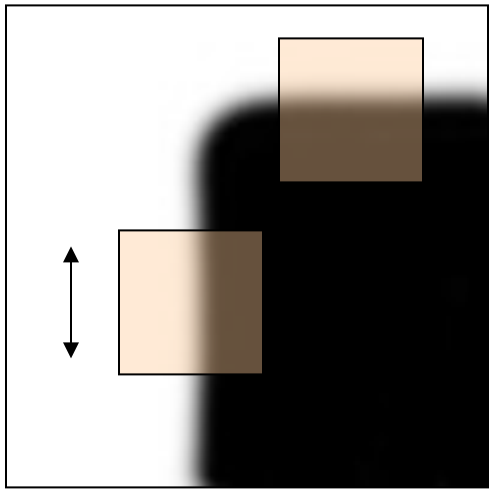
$$Mx_i = \lambda_i x_i$$



The *eigenvalues* of *M* reveal the amount of intensity change in the two principal orthogonal gradient directions in the window.

Slide credit: Judy Hoffman

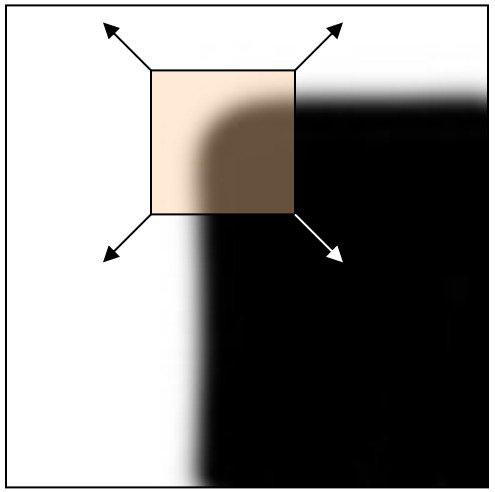
Corner response function



"edge":

$\lambda_1 >> \lambda_2$

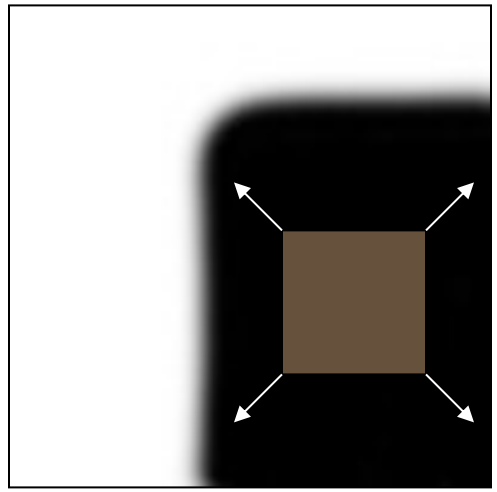
$\lambda_2 >> \lambda_1$



"corner":

λ_1 and λ_2 are large,

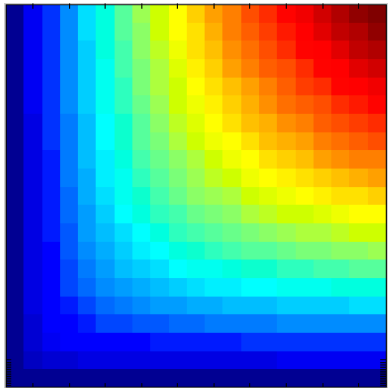
$\lambda_1 \sim \lambda_2$;



"flat" region

λ_1 and λ_2 are small;

$$f = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}$$



Slide credit: Judy Hoffman

Harris corner detector

1. Cornerness scores
   - Compute M matrix for each image window
2. Threshold scores
   - Find points whose surrounding window gave large corner response
     (*f*> *threshold*)
3. Local filter for most cornerlike region
   - Take the points of local maxima
     i.e., perform non-maximum suppression

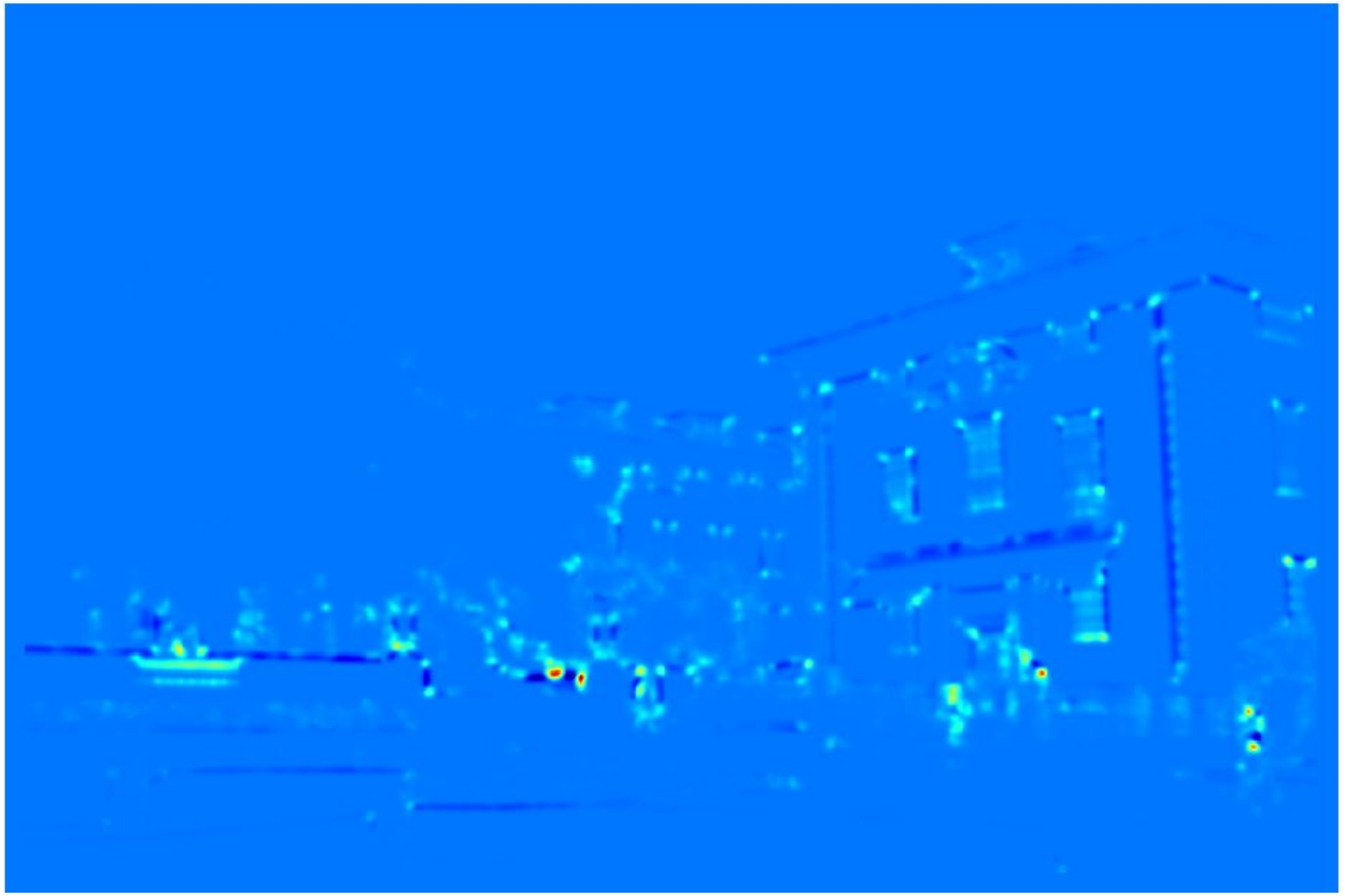
Slide credit: Judy Hoffman

Example of Harris application

Slide credit: Judy Hoffman

Example of Harris application

Compute corner response at every pixel.



Slide credit: Judy Hoffman

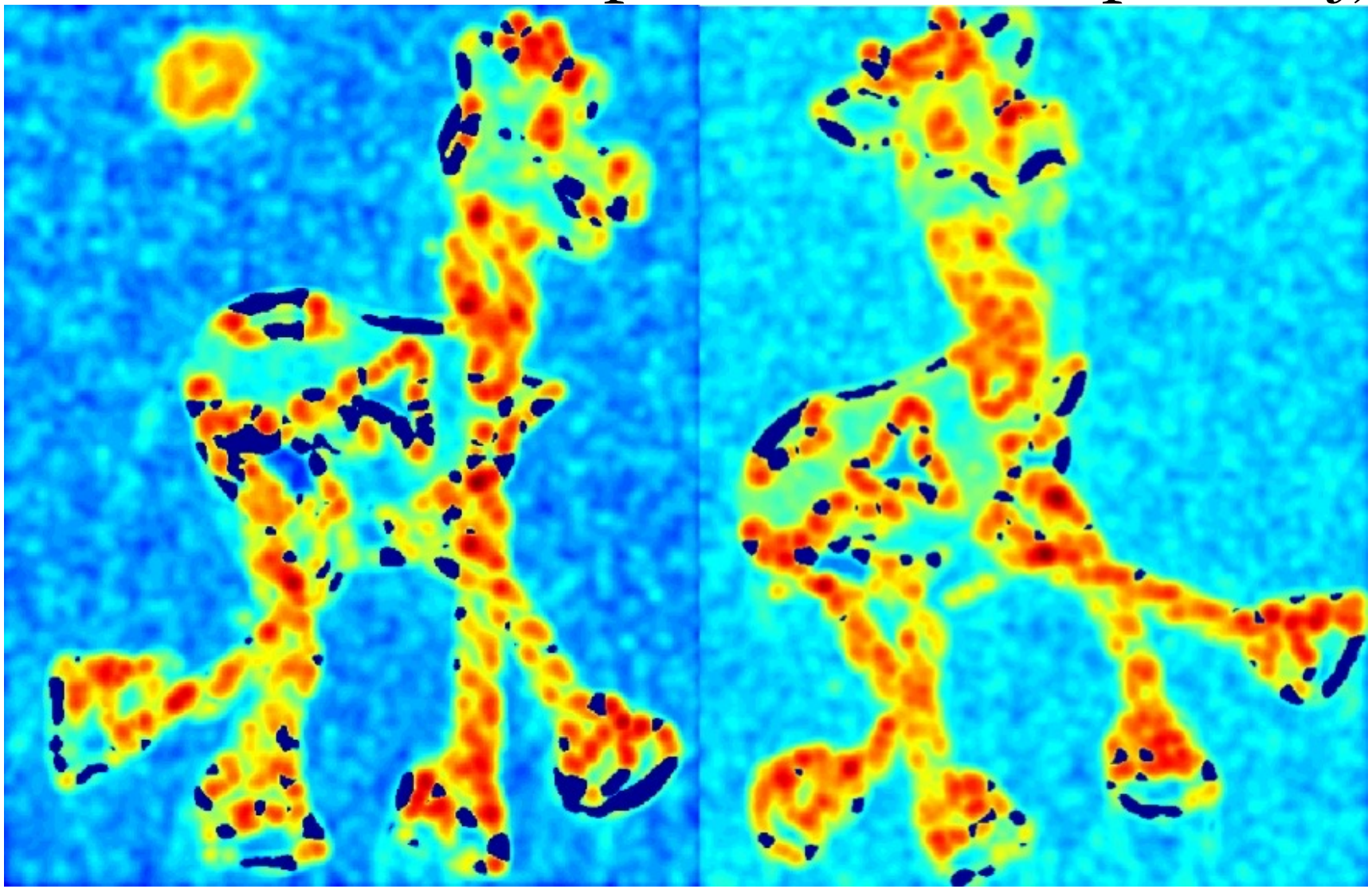
Example of Harris application

Slide credit: Judy Hoffman

Harris Detector: Steps

Slide credit: Kristen Grauman

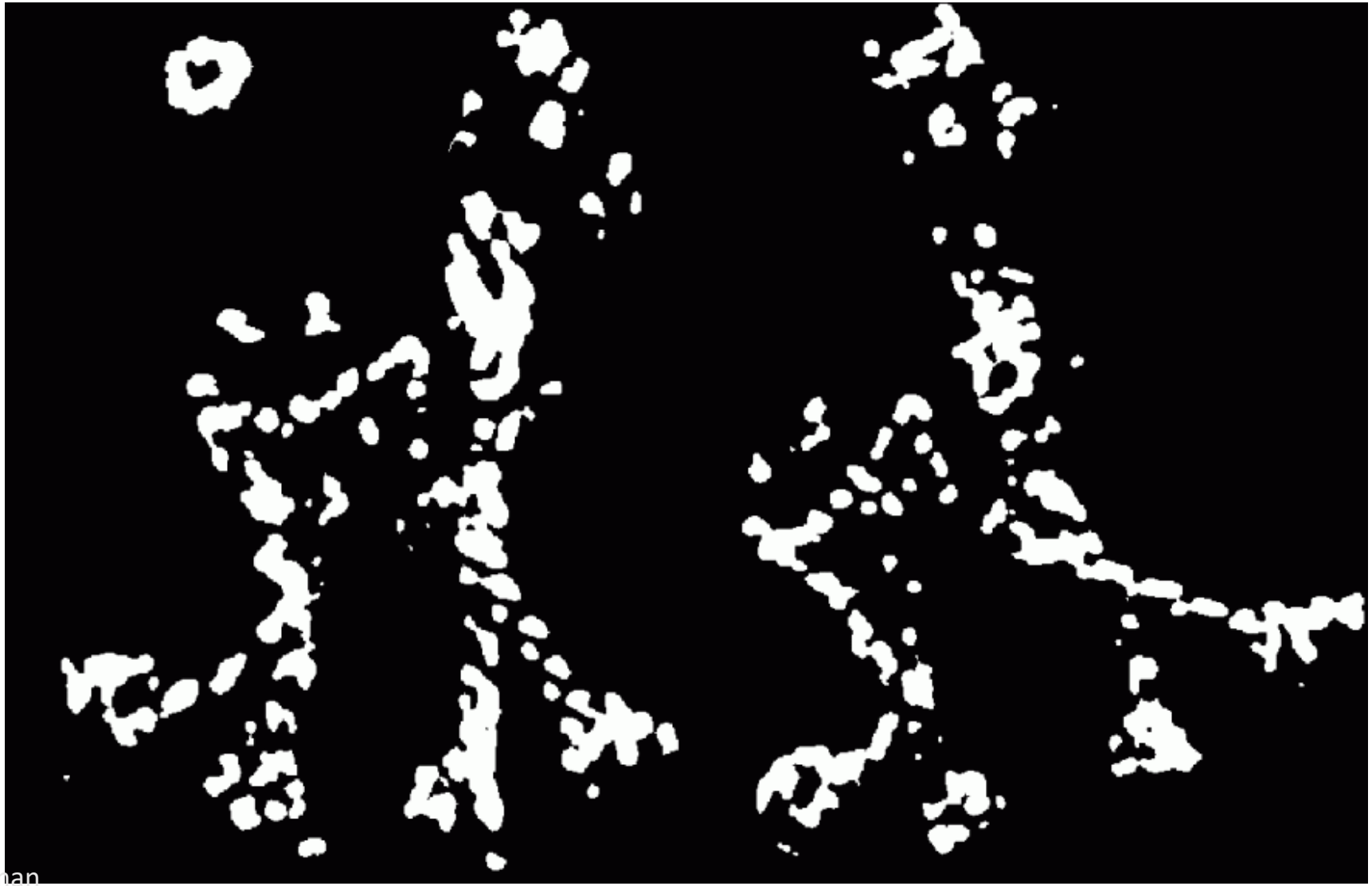
1. Harris Detector: compute corner response *(f)*



Slide credit: Kristen Grauman

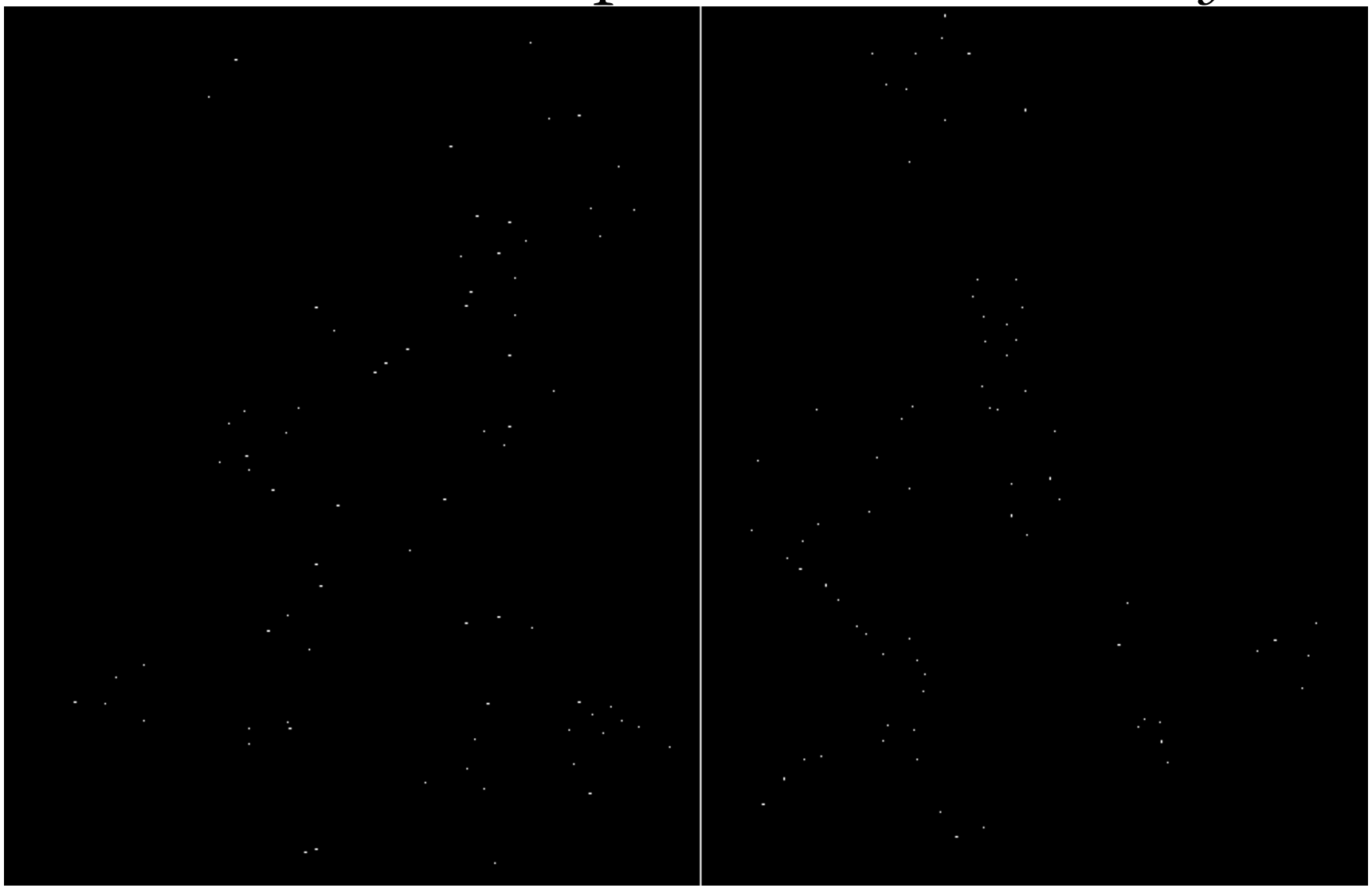
2. Harris: Filter points with large corner response

$f >$ threshold



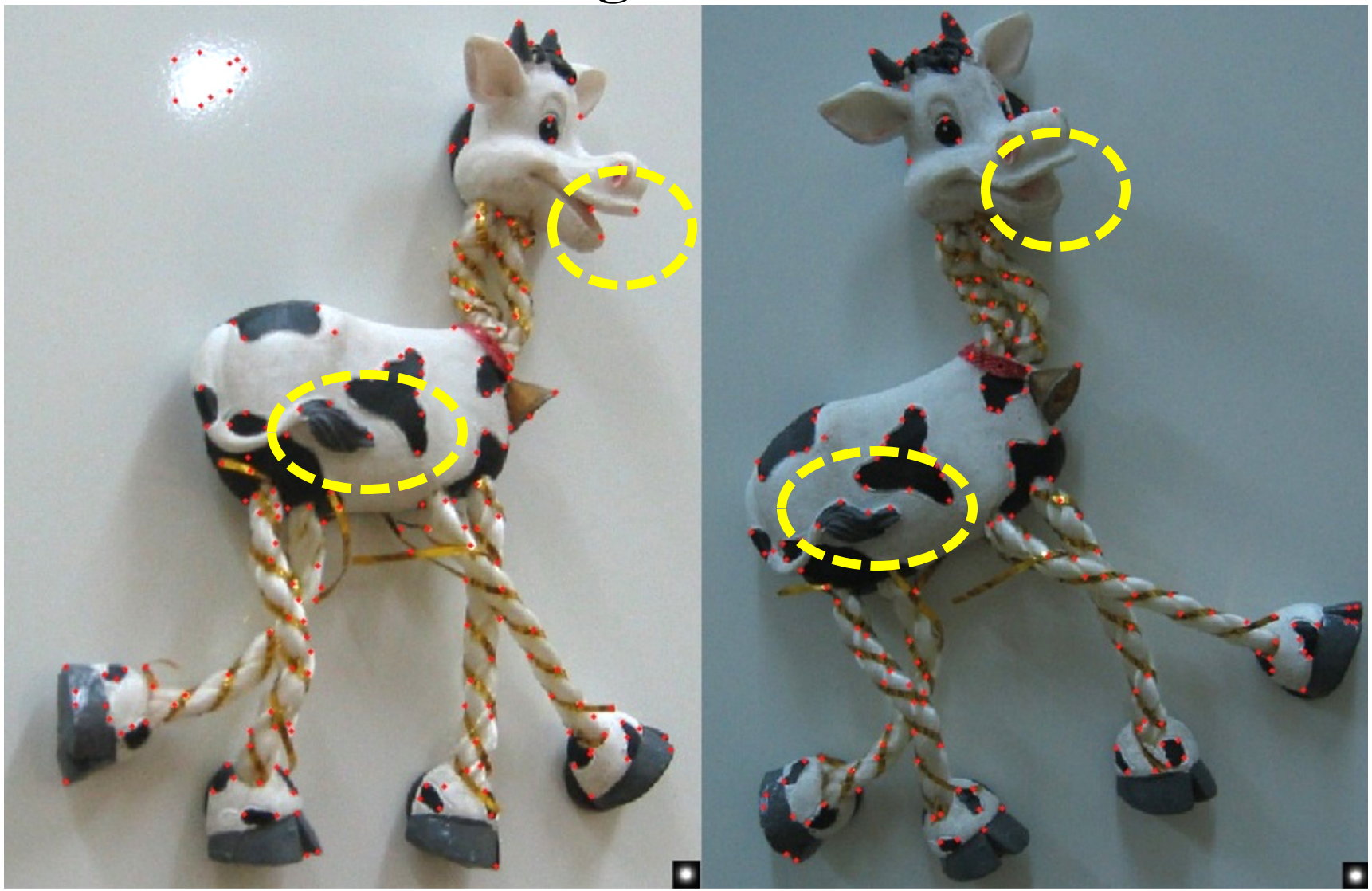
Slide credit: Kristen Grauman

3. Harris Detector: Keep local maxima of f



Slide credit: Kristen Grauman

Harris Detector: Finding Matches



Slide credit: Kristen Grauman

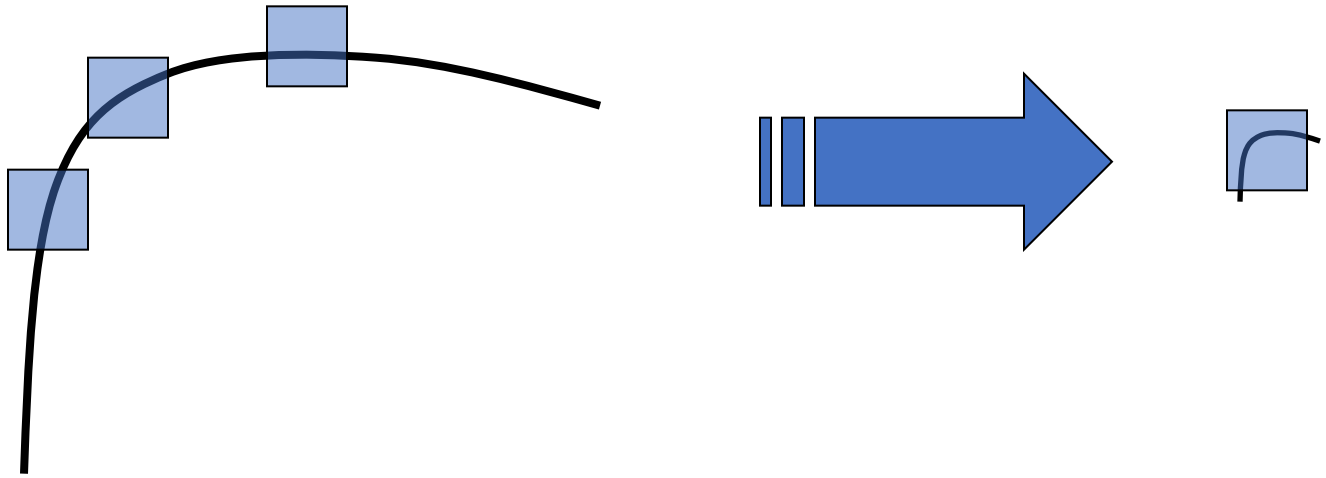
Keypoints by Harris Corner Detector

Keypoints by Harris Corner Detector

Properties of the Harris corner detector

- Rotation invariant? Yes

- Scale invariant? No



All points will be classified as edges

Corner !

Local Feature Detection

Blob Detector with Automatic Scale Selection

Scale invariant interest points

How can we independently select interest points in each image, such that the detections are repeatable across different scales?

Slide credit: Kristen Grauman

Scale invariant interest points

How can we independently select interest points in each image, such that the detections are repeatable across different scales?

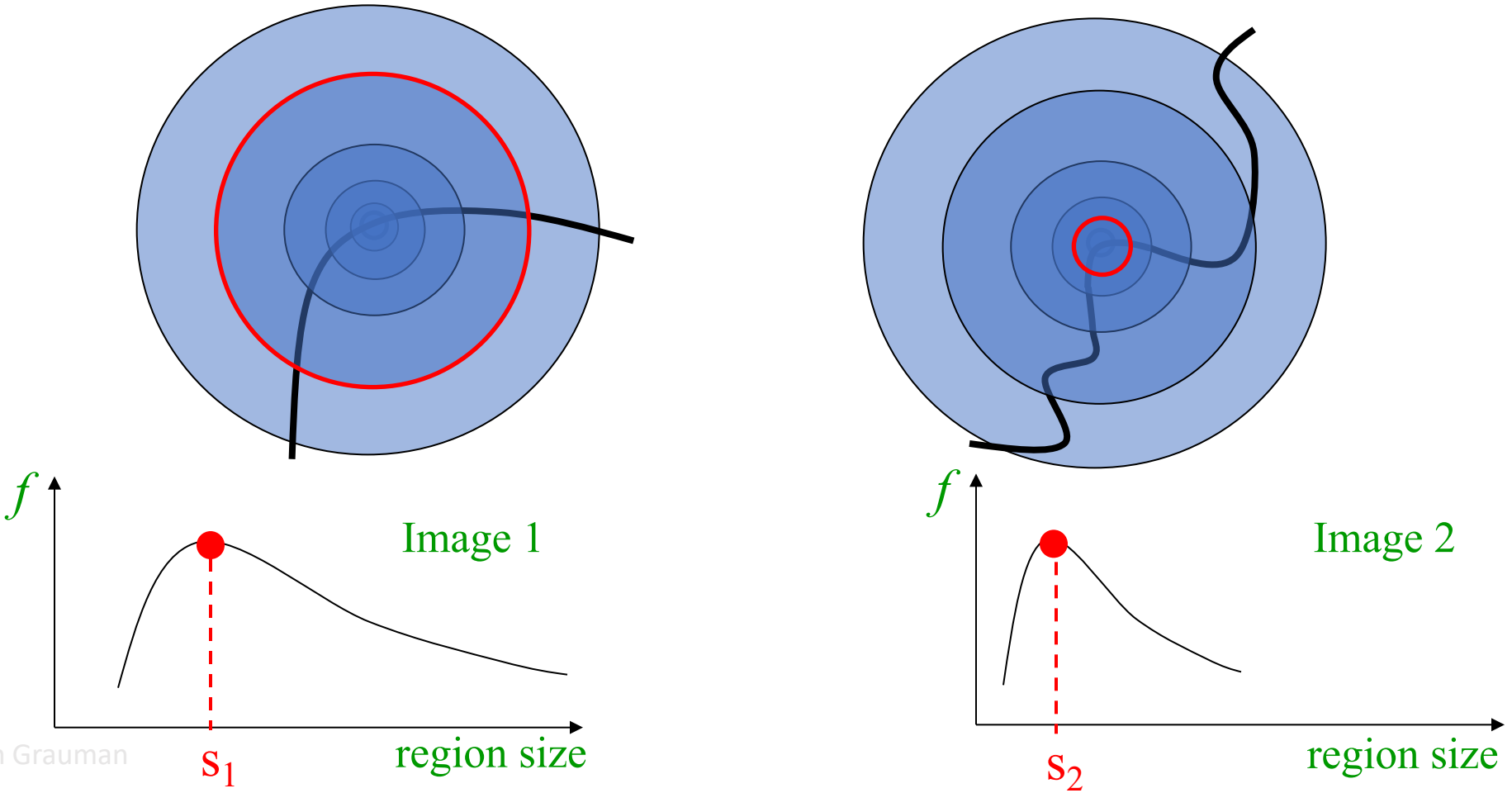
Scale invariant interest points

How can we independently select interest points in each image, such that the detections are repeatable across different scales?

Automatic scale selection

Intuition:

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



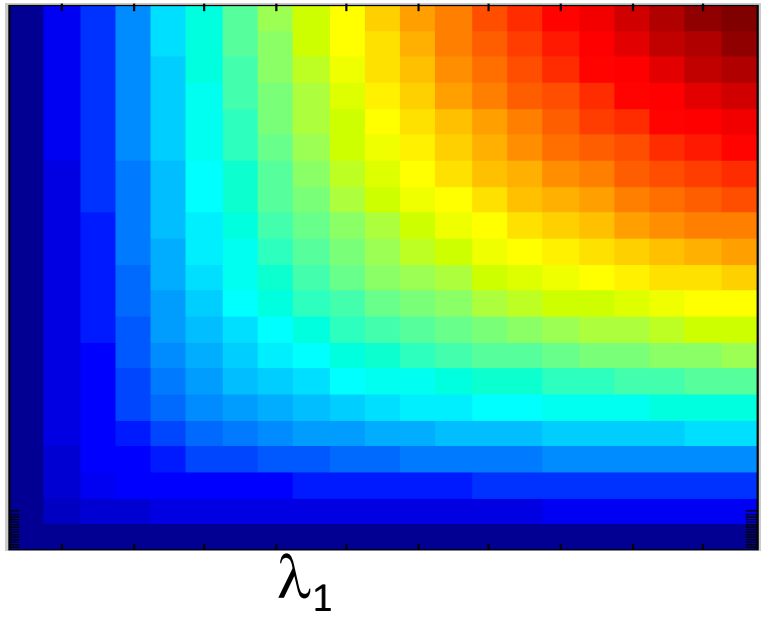
Slide credit: Kristen Grauman

Choosing f

- What can be the "signature" function?

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

$$f = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}$$



Slide credit: Kristen Grauman

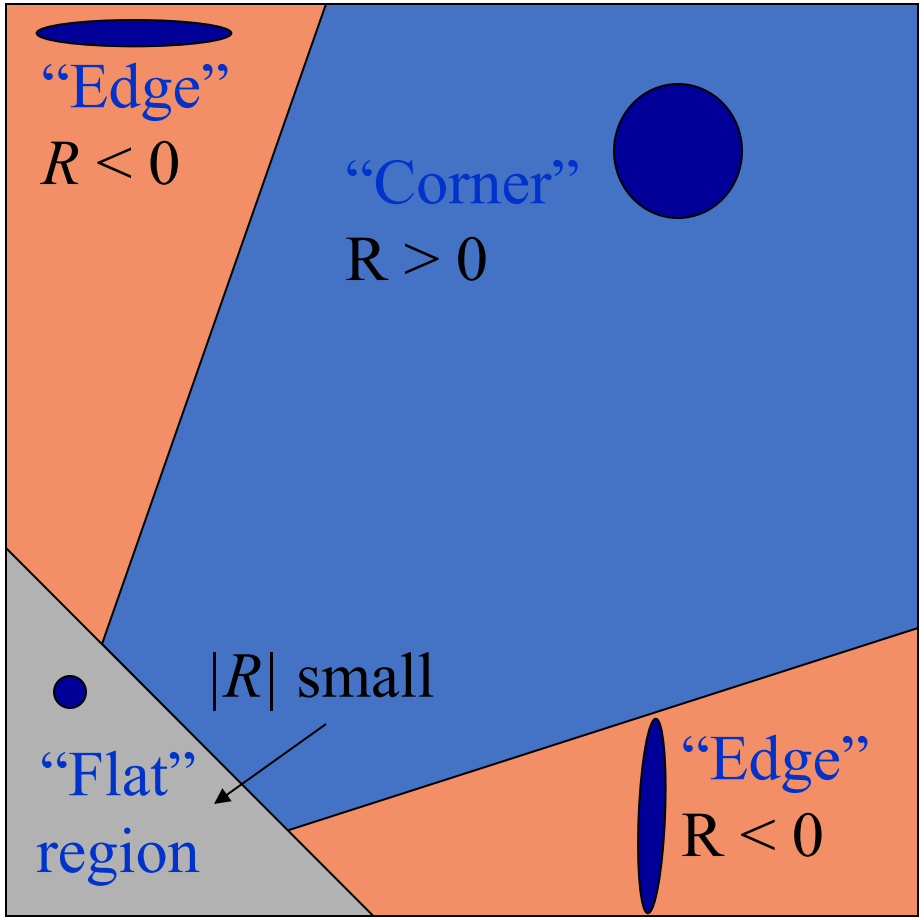
Choosing f

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

α: constant (0.04 to 0.06)

Calculating eigenvalues is usually slow!

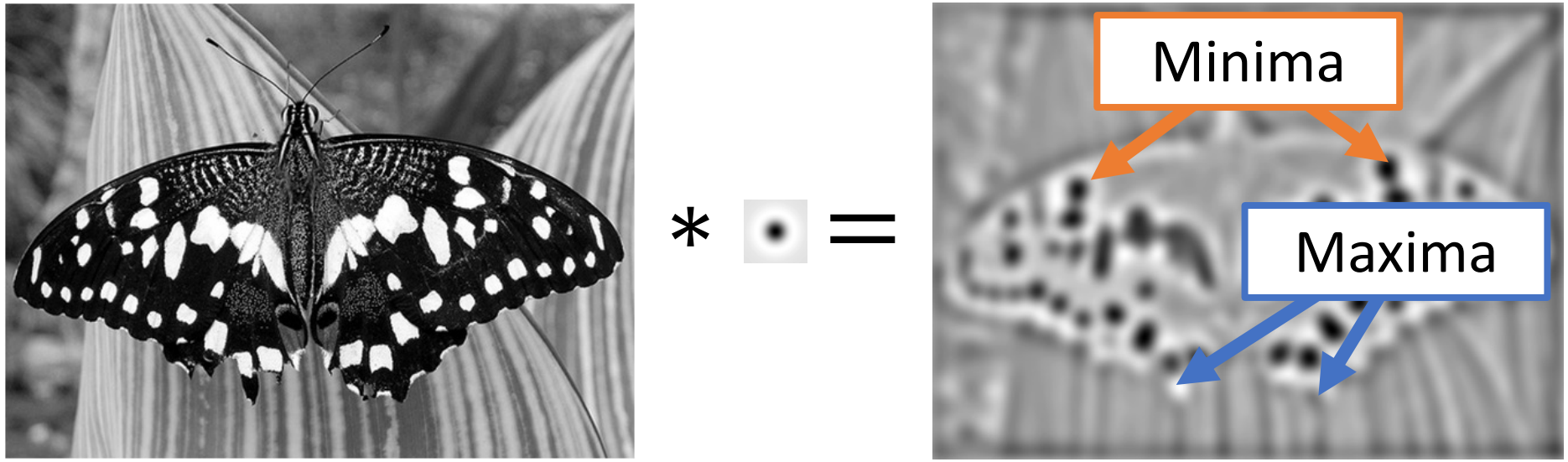
Determinant, trace have closed form solution



Slide credit: S. Lazebnik; Note: this refers to visualization ellipses, not original M ellipse. Other slides on the internet may vary

Blob Detection

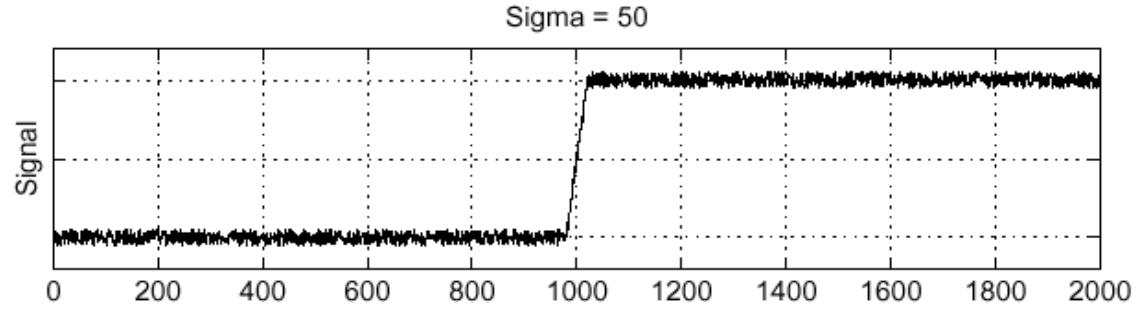
Another detector (has some nice properties)



Find maxima ***and minima*** of blob filter response in scale ***and space***

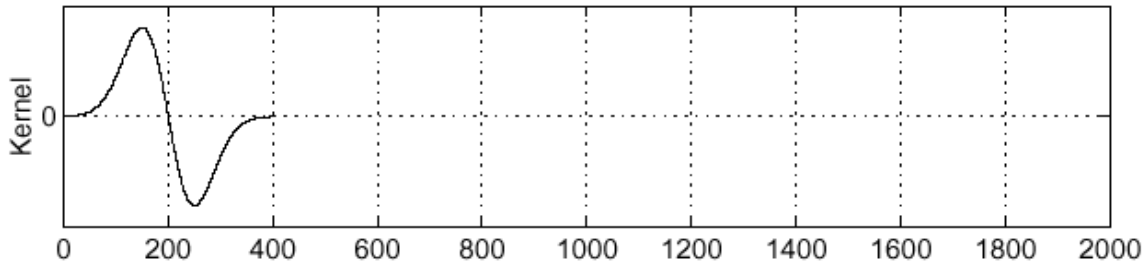
Recall: Edge Detection

f



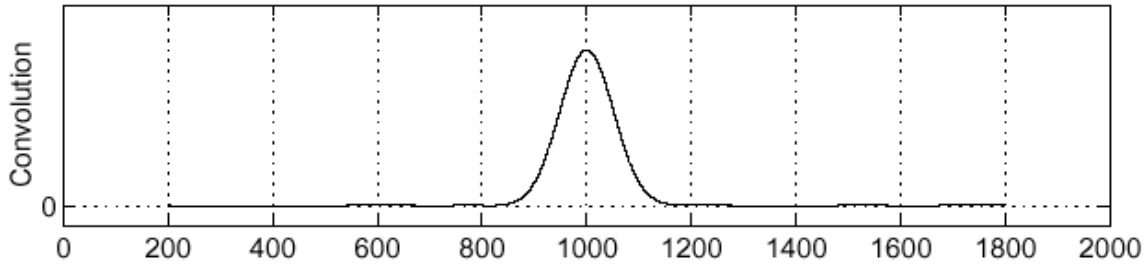
Edge

$\frac{d}{dx} g$



Derivative of Gaussian

$f * \frac{d}{dx} g$

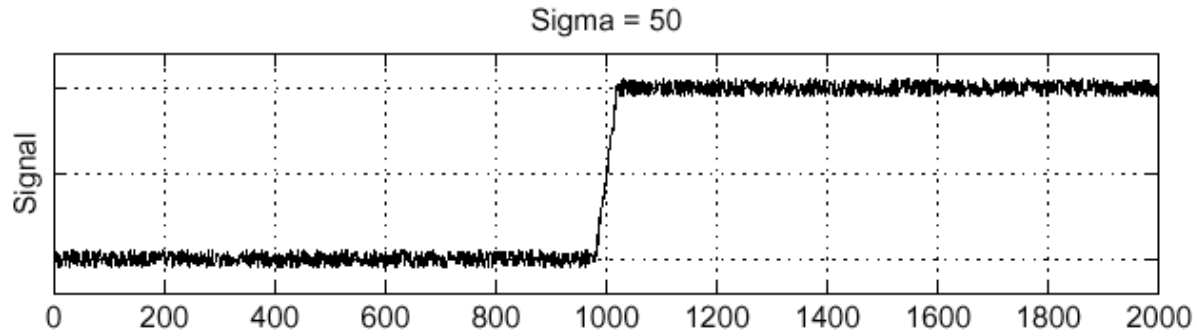


Edge = maximum of derivative

Source: S. Seitz

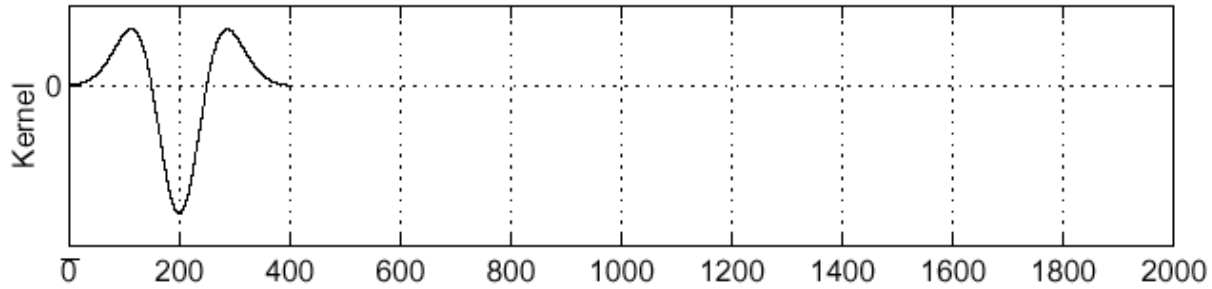
Recall: Edge Detection

f



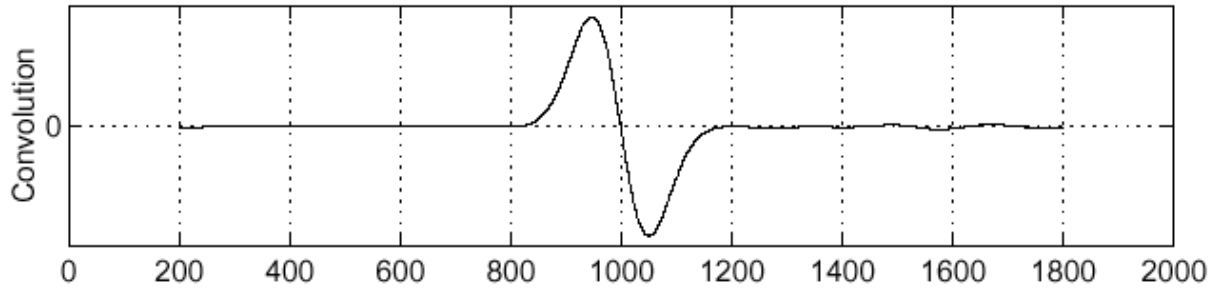
Edge

$\frac{d^2}{dx^2} g$



Second derivative of Gaussian (Laplacian)

$f * \frac{d^2}{dx^2} g$

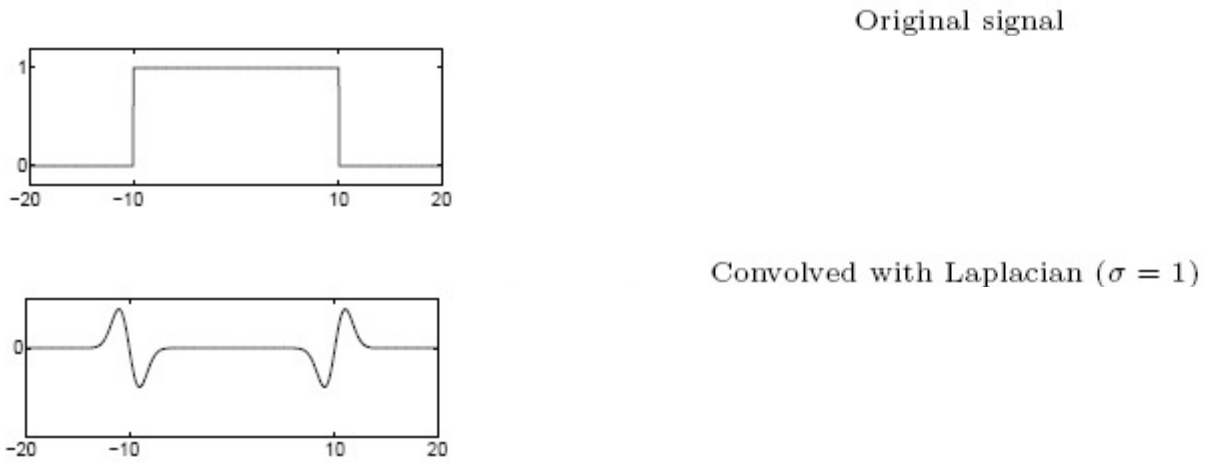


Edge = zero crossing of second derivative

Source: S. Seitz

From edges to blobs

- Edge = ripple
- Blob = superposition of two ripples



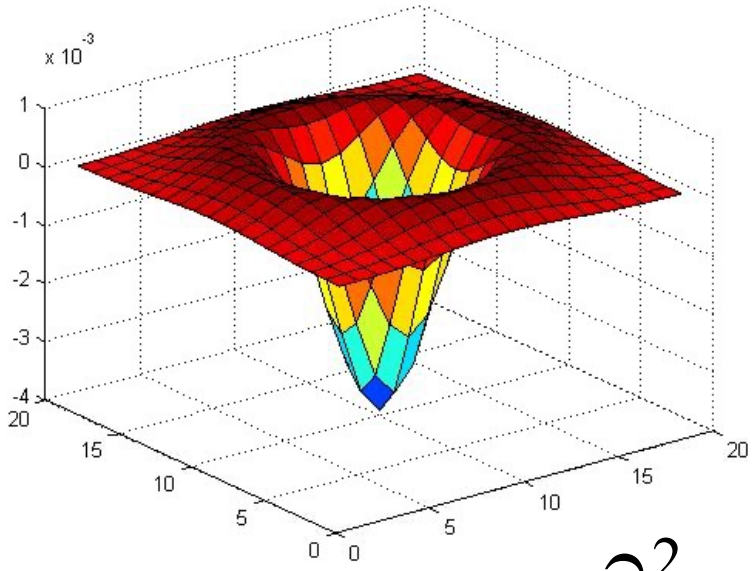
maximum

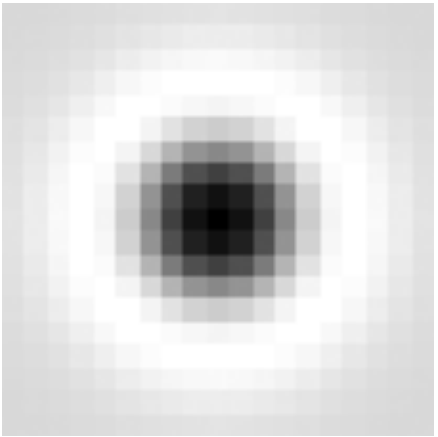
Spatial selection: the **magnitude** of the Laplacian response will achieve a maximum at the center of the blob, provided the scale of the Laplacian is "matched" to the scale of the blob

Slide credit: Lana Lazebnik

Blob detection in 2D

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





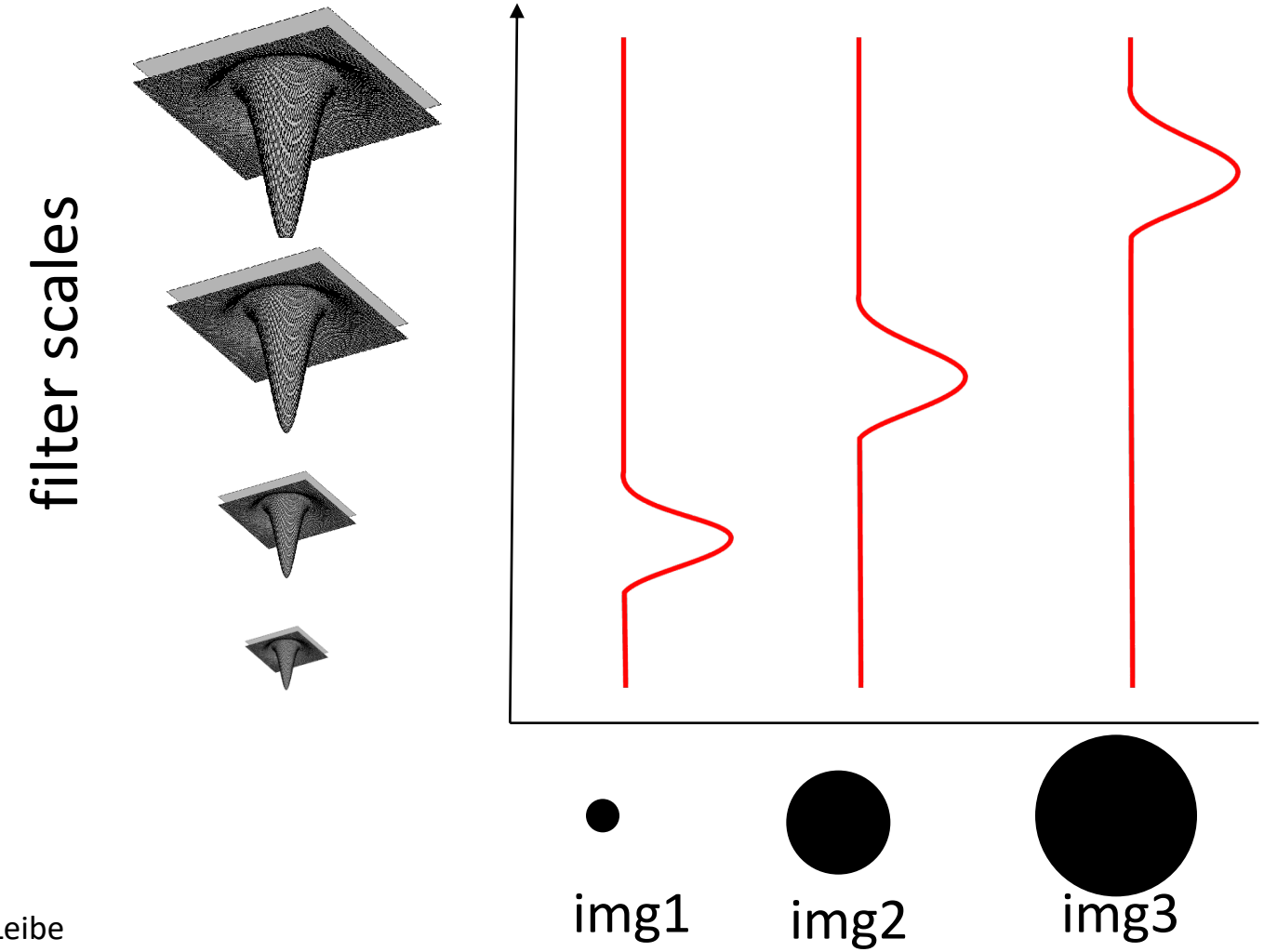
$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

Slide credit: Kristen Grauman

Blob detection in 2D: scale selection

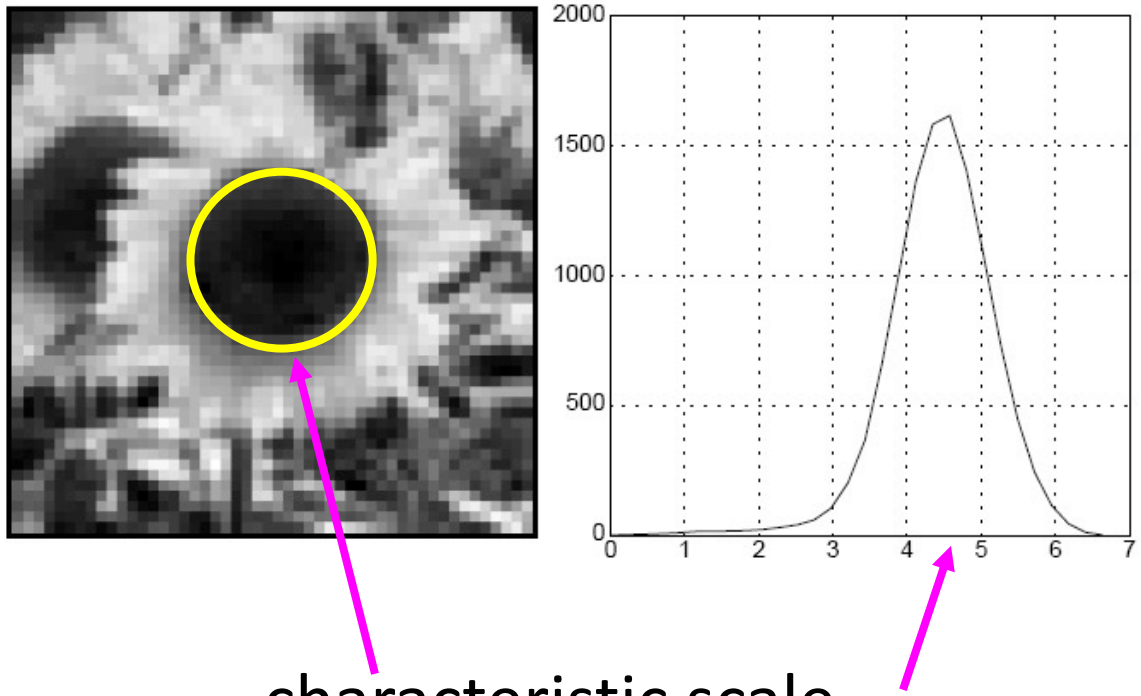
Laplacian-of-Gaussian = "blob" detector

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



Blob detection in 2D

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

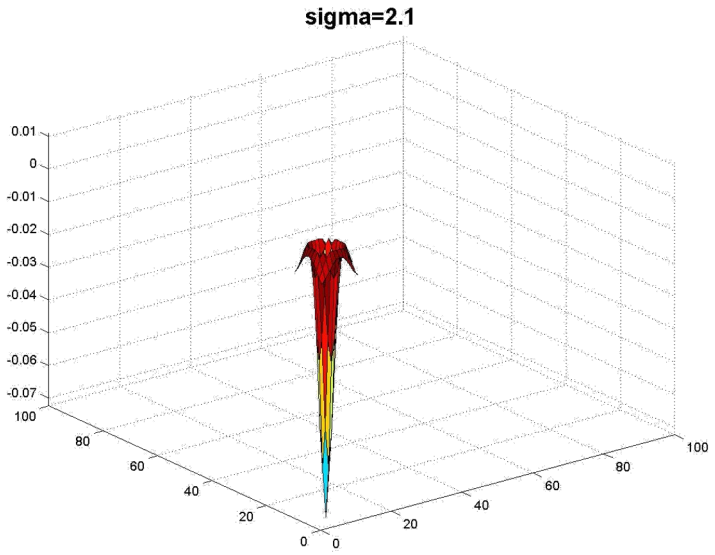


Slide credit: Lana Lazebnik

Example

Original image at
¾ the size

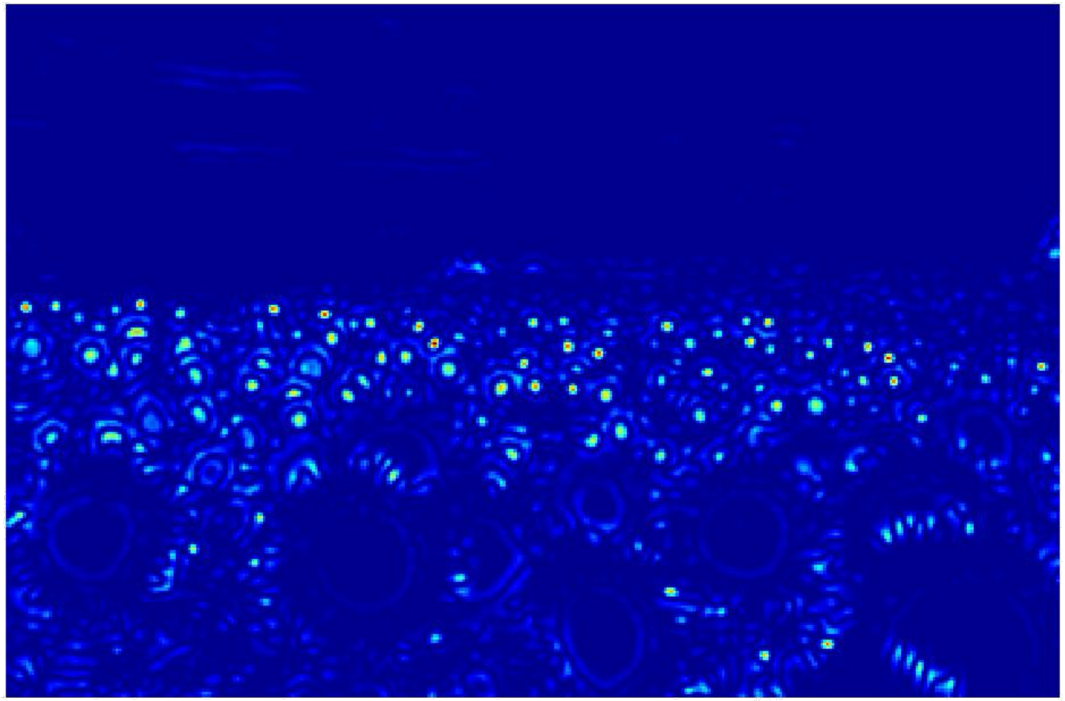
What will we recognize?

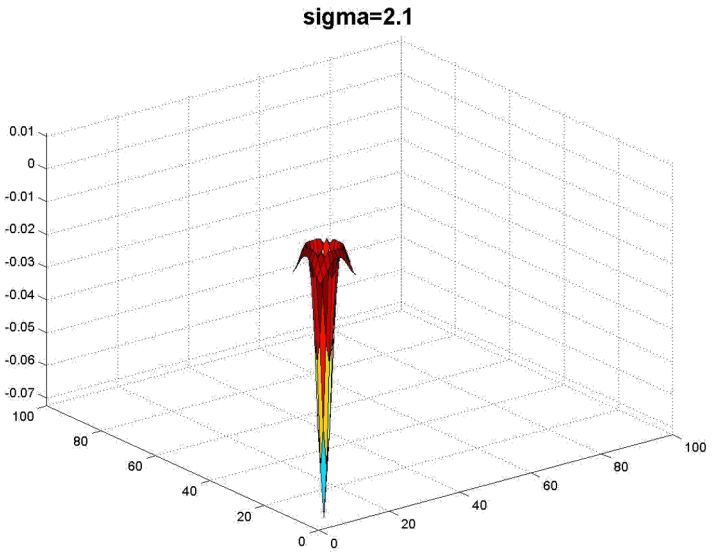


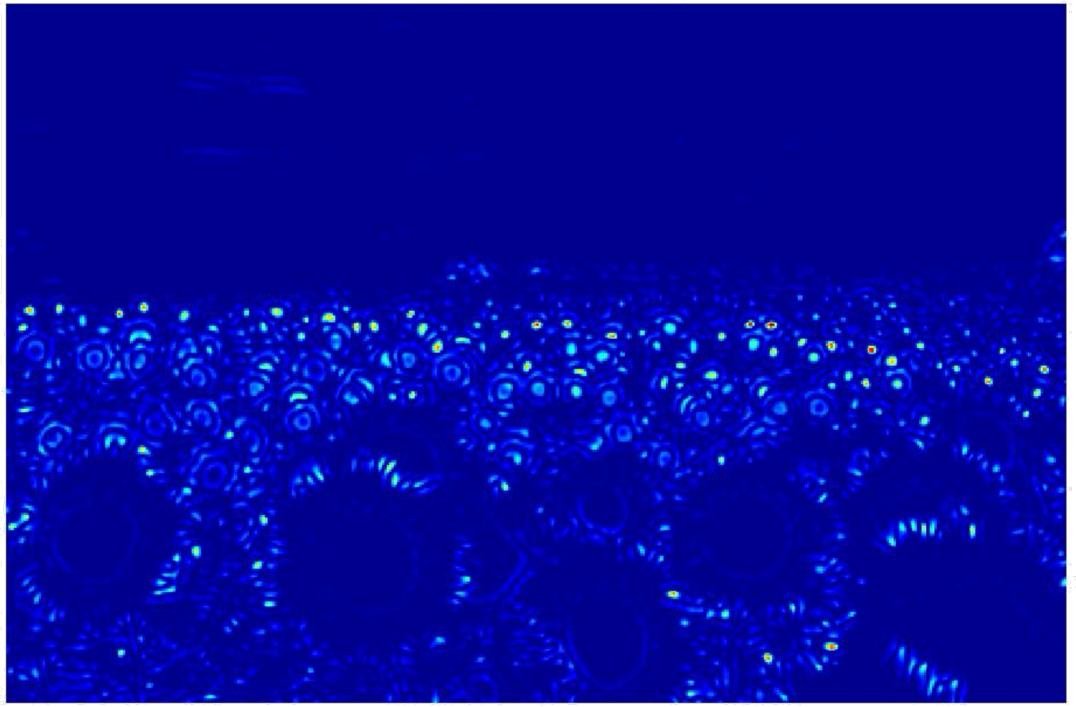
Slide credit: Kristen Grauman

Example

Original image at
¾ the size

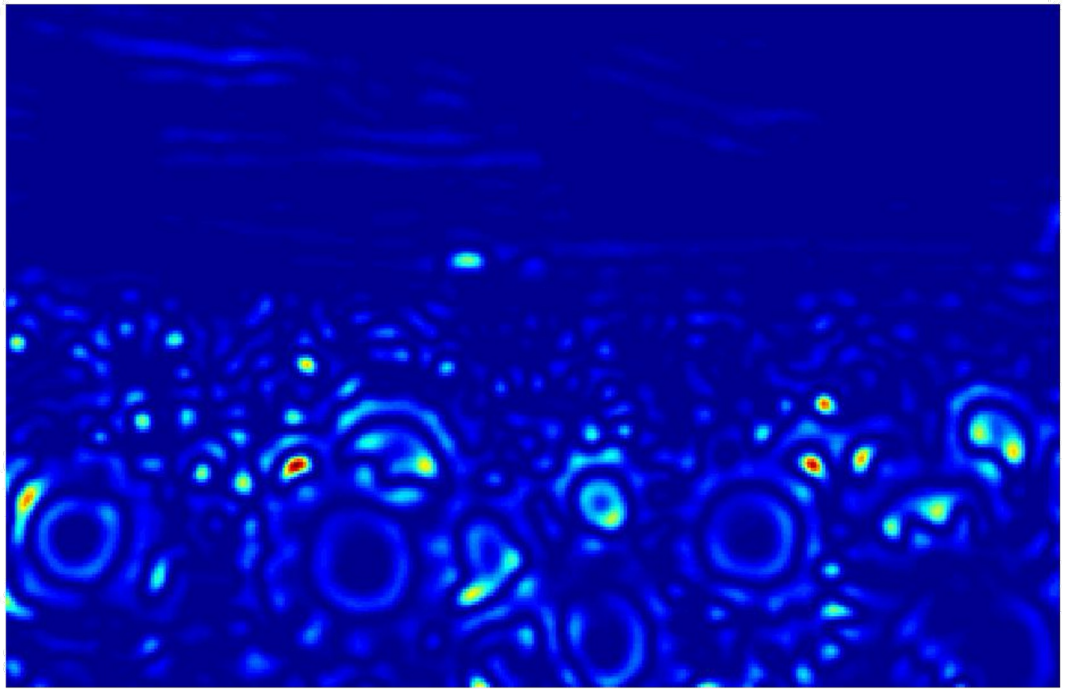


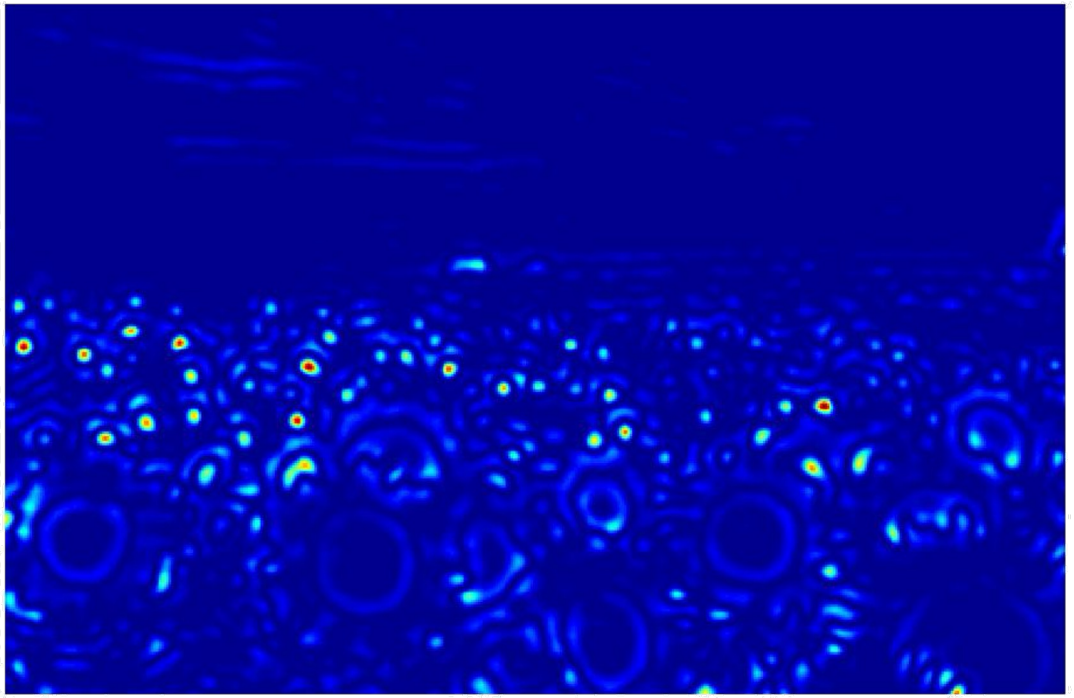


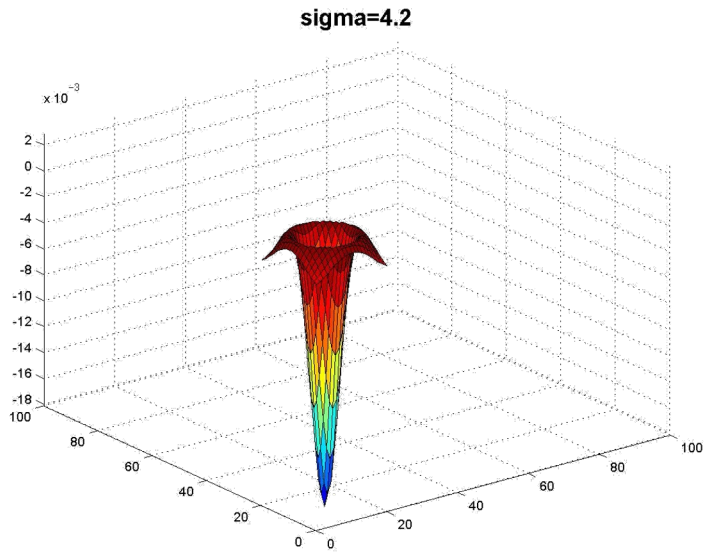


Slide credit: Kristen Grauman

Example



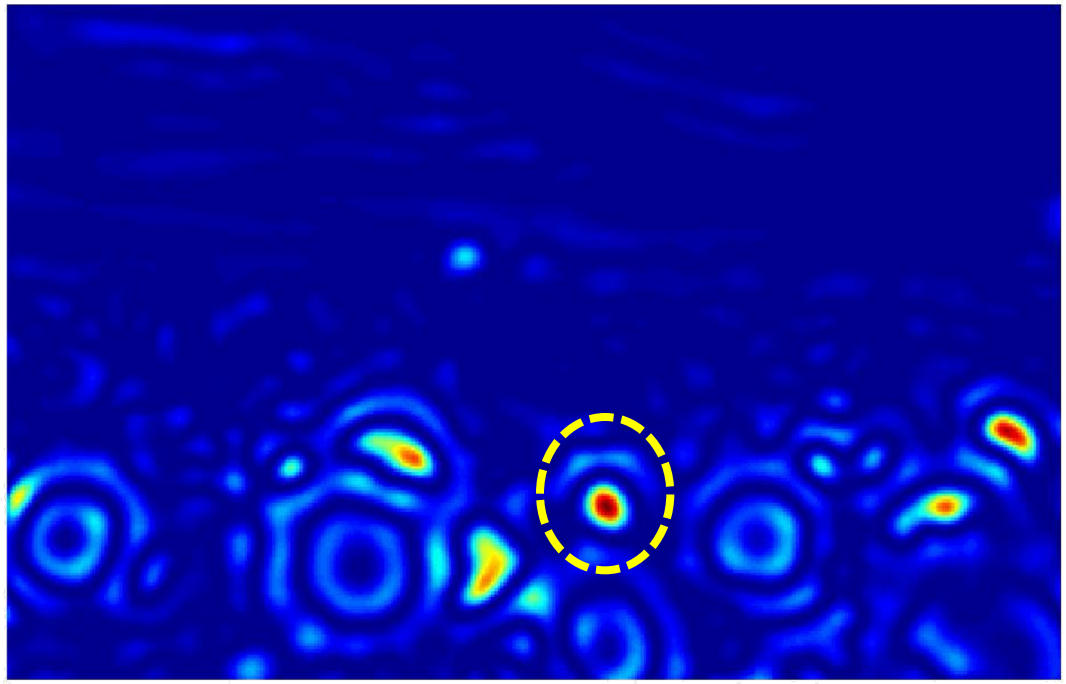


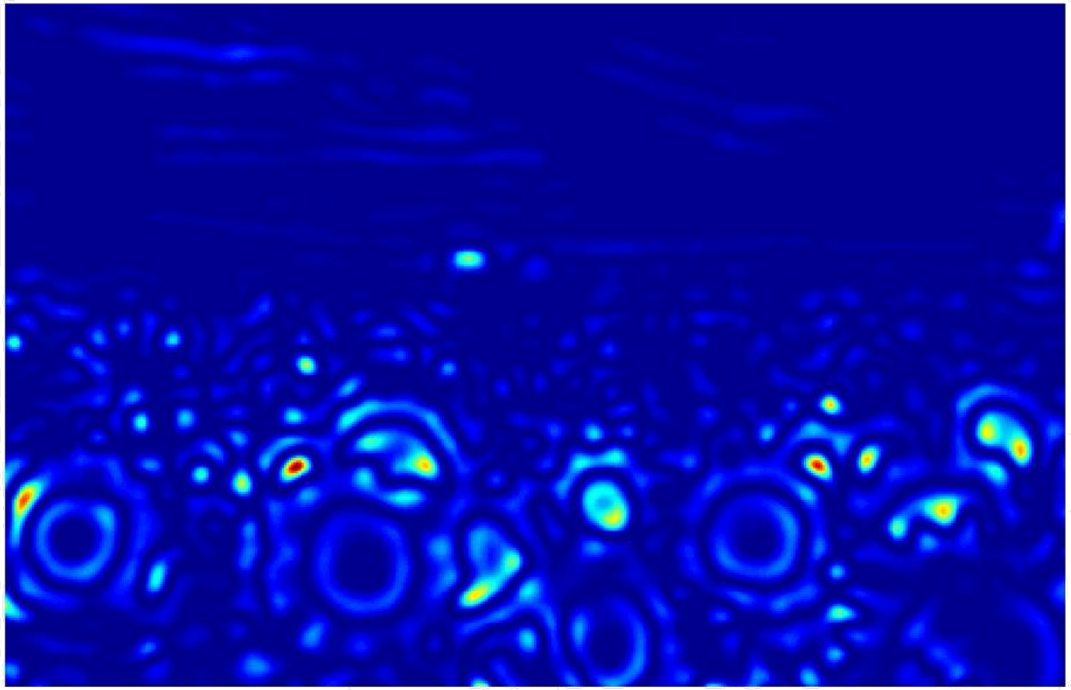


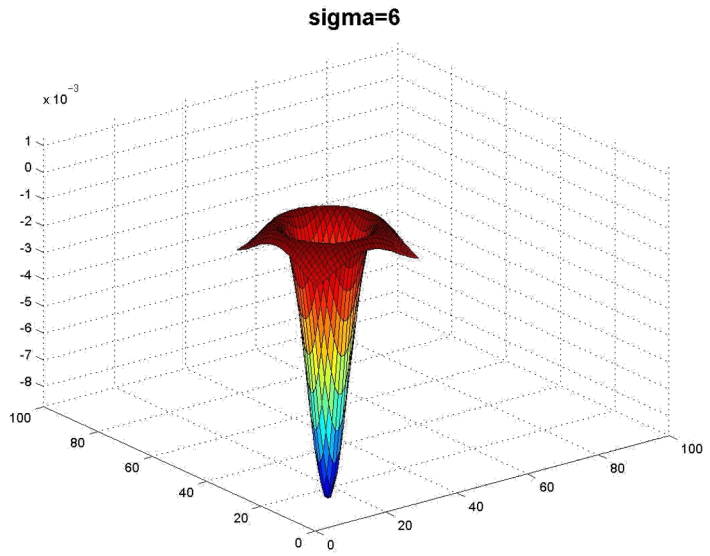
Slide credit: Kristen Grauman

Example

Do we recognize the same points in the two images?

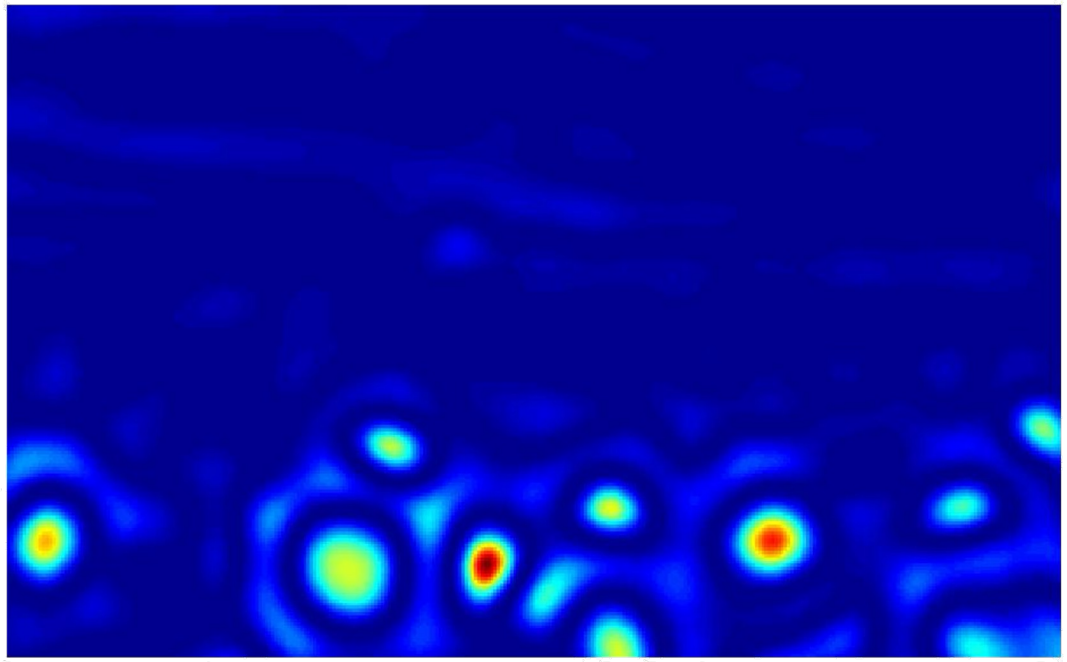


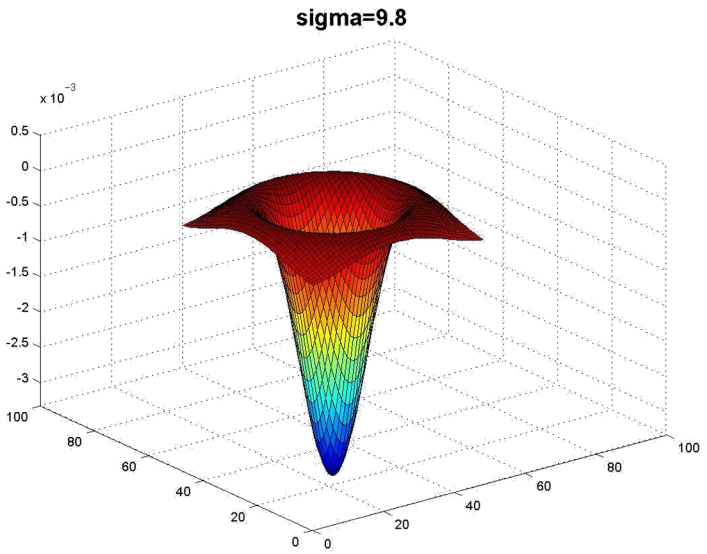


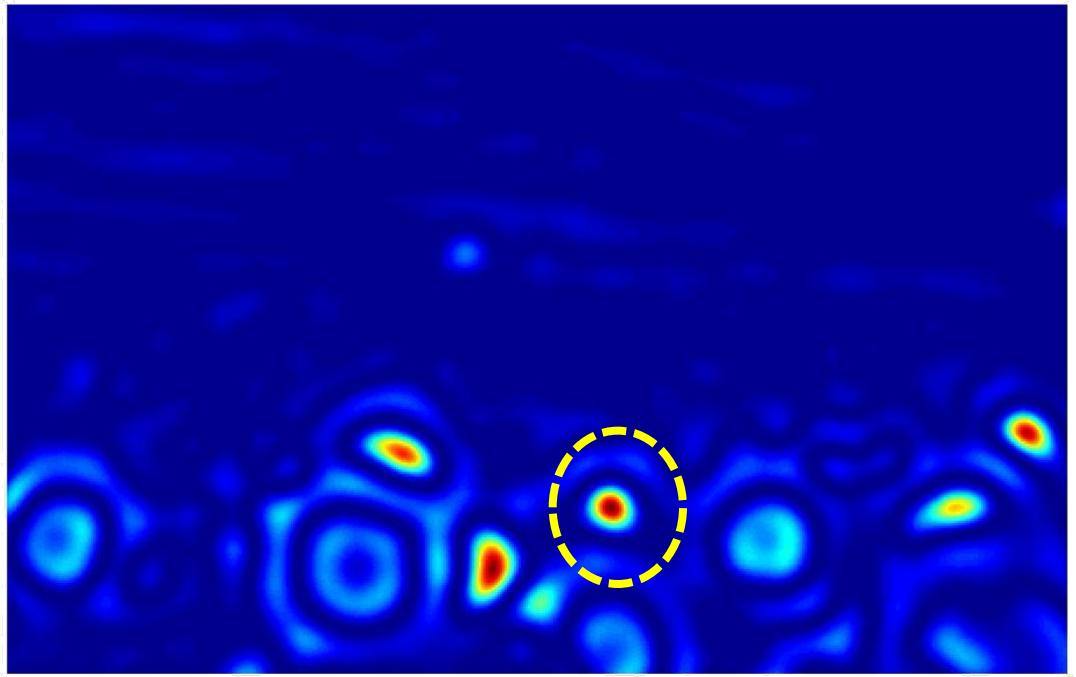


Slide credit: Kristen Grauman

Example

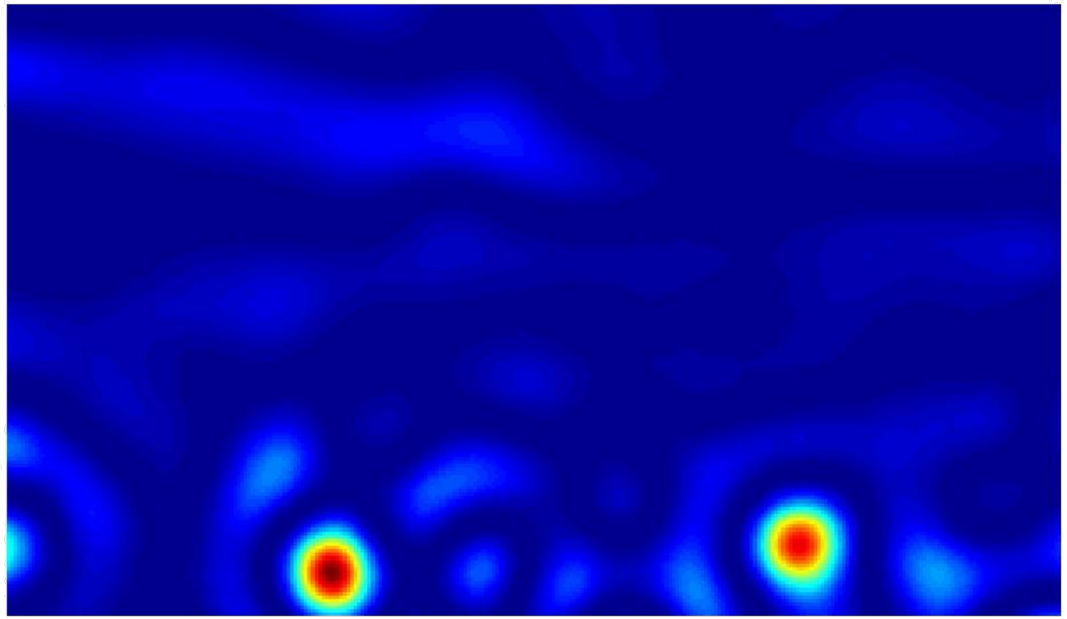


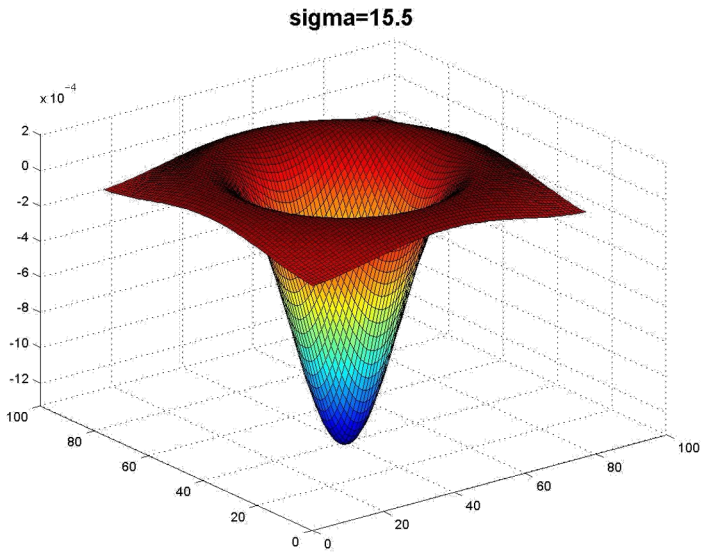


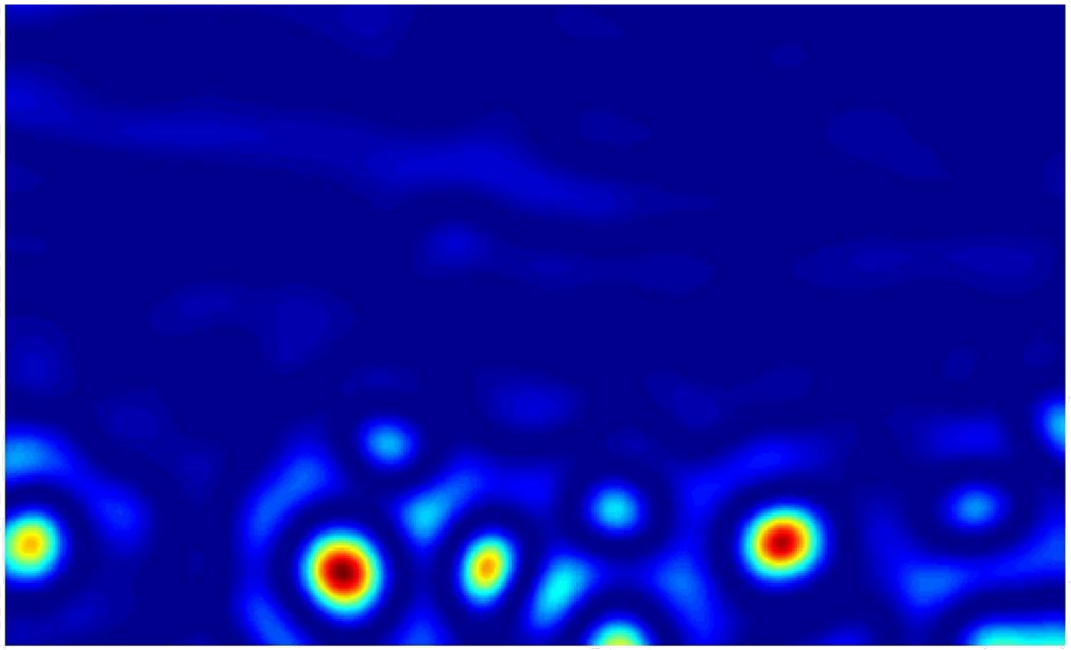


Slide credit: Kristen Grauman

Example

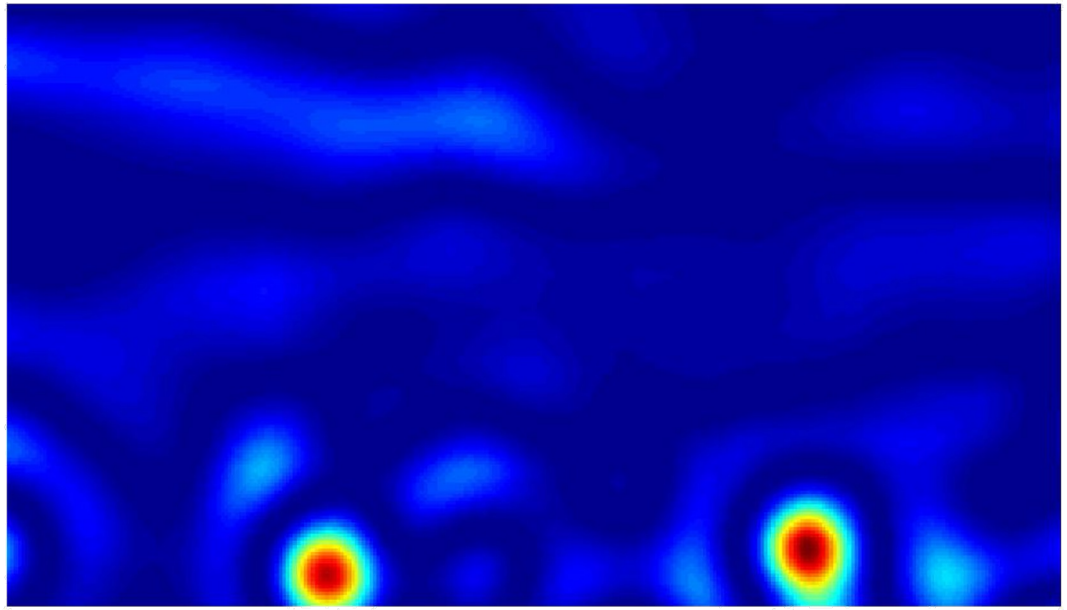


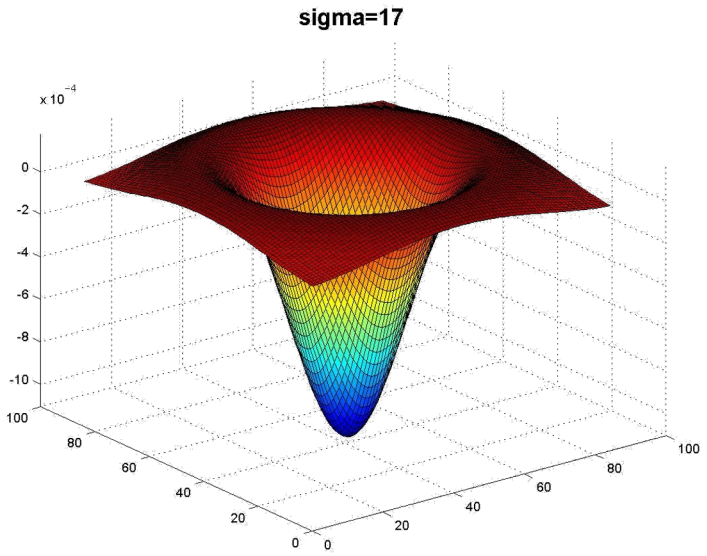


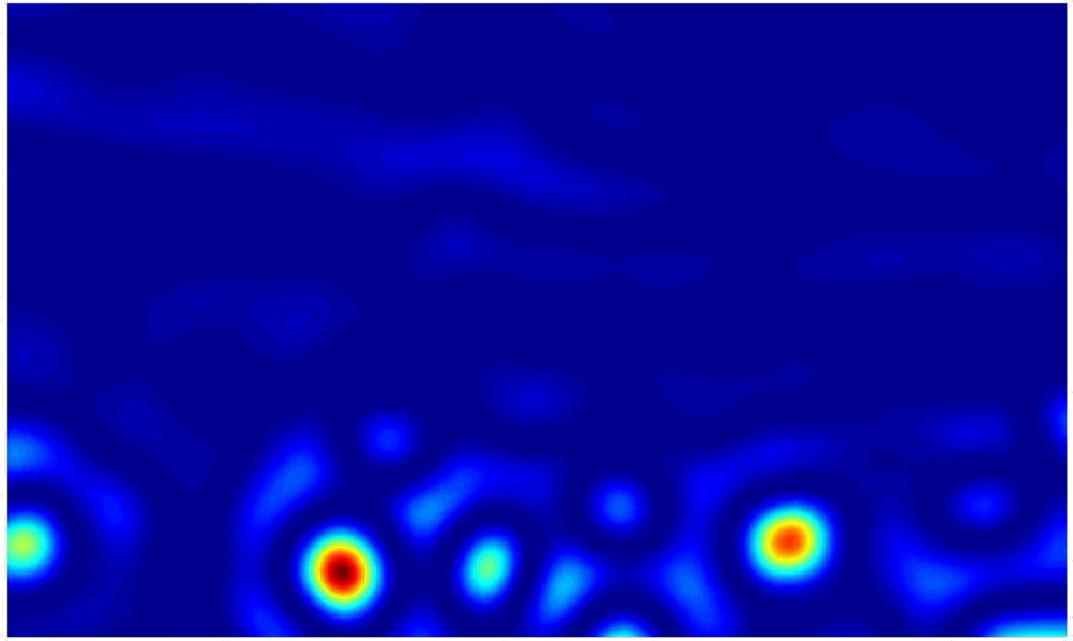


Slide credit: Kristen Grauman

Example





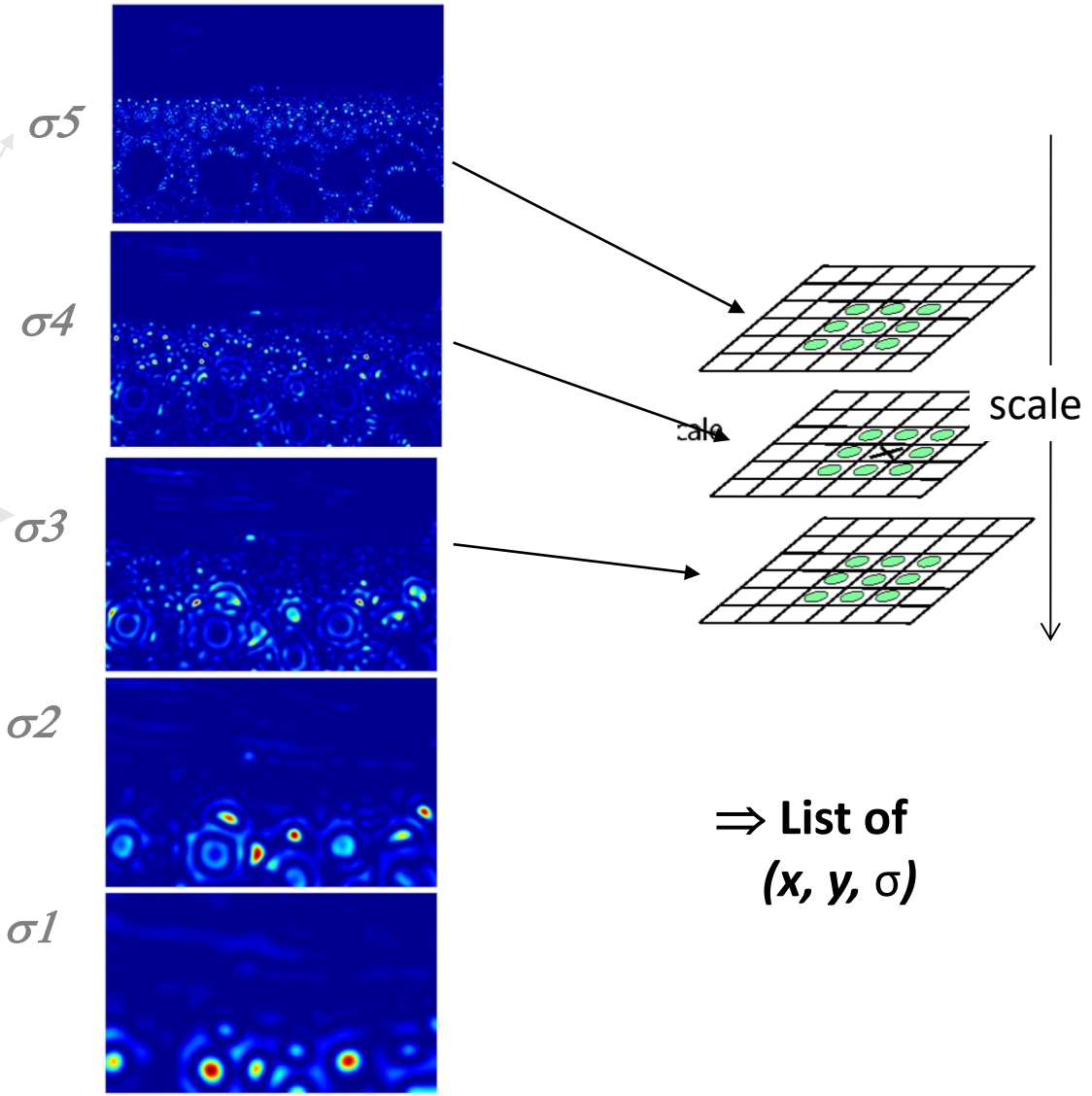


Slide credit: Kristen Grauman

Scale invariant interest points

Interest points are local maxima in both position and scale.

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

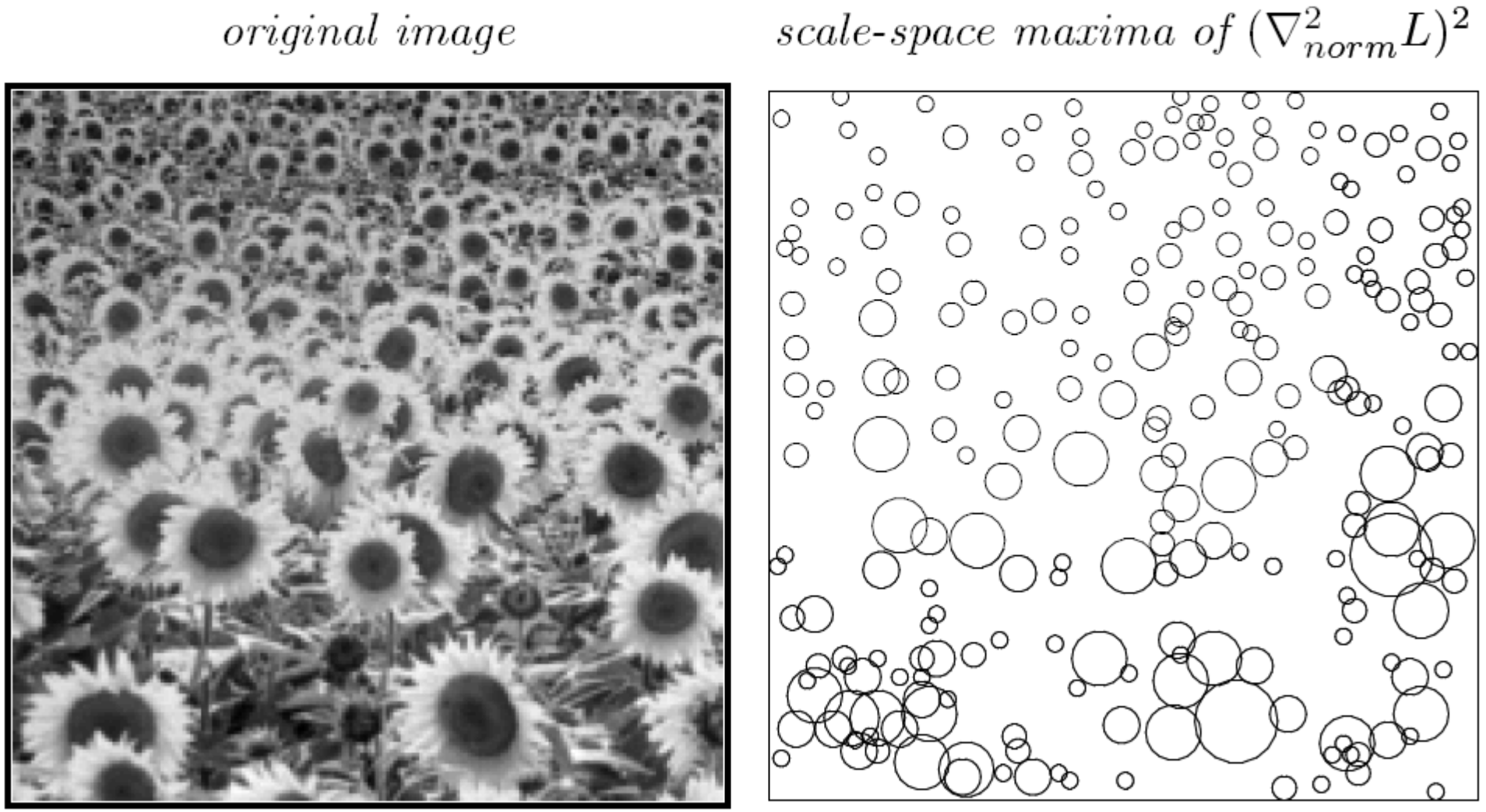


Squared filter response maps

⇒ **List of** ***(x, y, σ)***

Slide credit: Kristen Grauman

Scale-space blob detector: Example



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

Slide credit: Kristen Grauman

Scale-space blob detector: Example

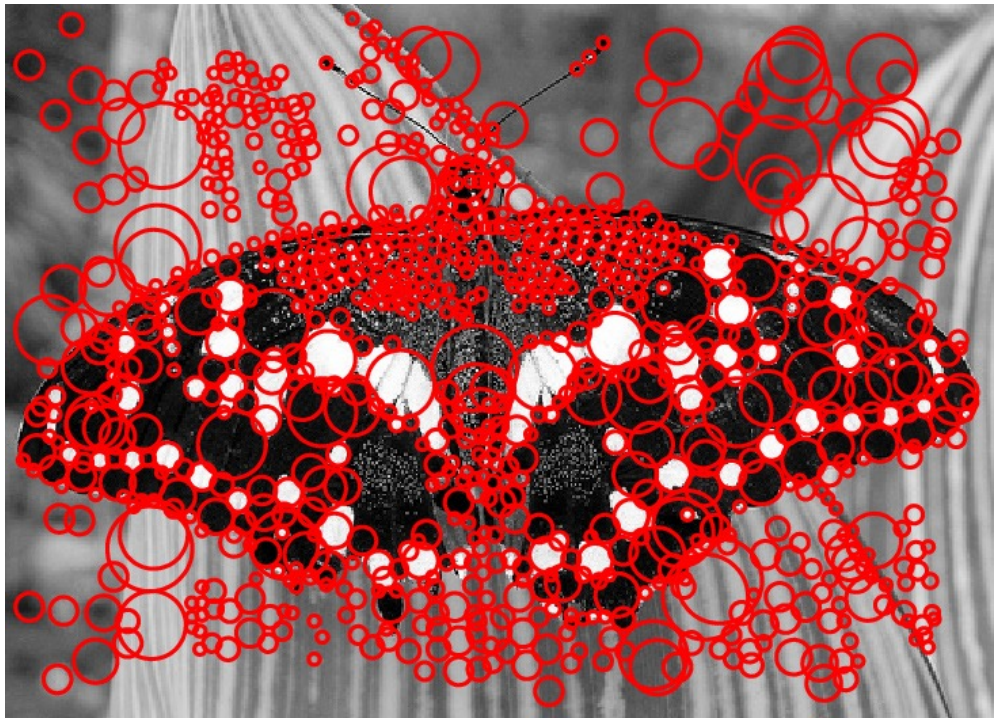


Image credit: Lana Lazebnik

Questions?

See you next time!