3D Vision and Machine Perception

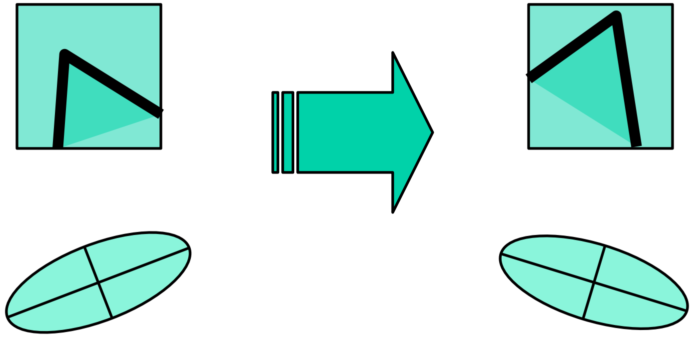
Prof. Kyungdon Joo

3D Vision & Robotics Lab.

AI Graduate School (AIGS) & Computer Science and Engineering (CSE)

Recap: Harris corner response is invariant to rotation

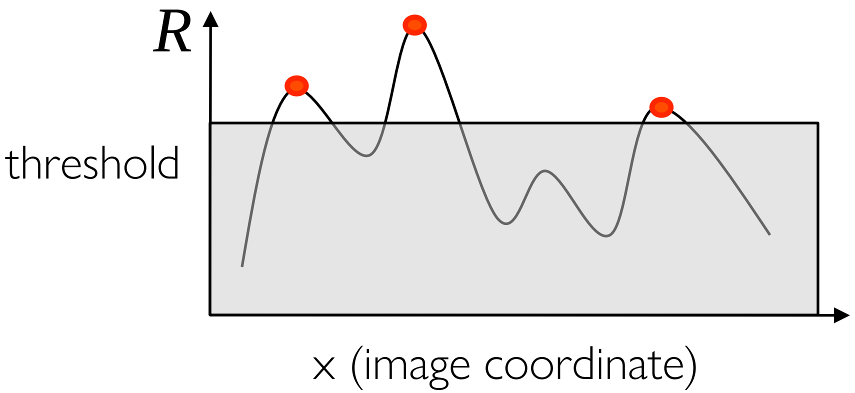
- Ellipse rotates but its shape (eigenvalues) remains the same

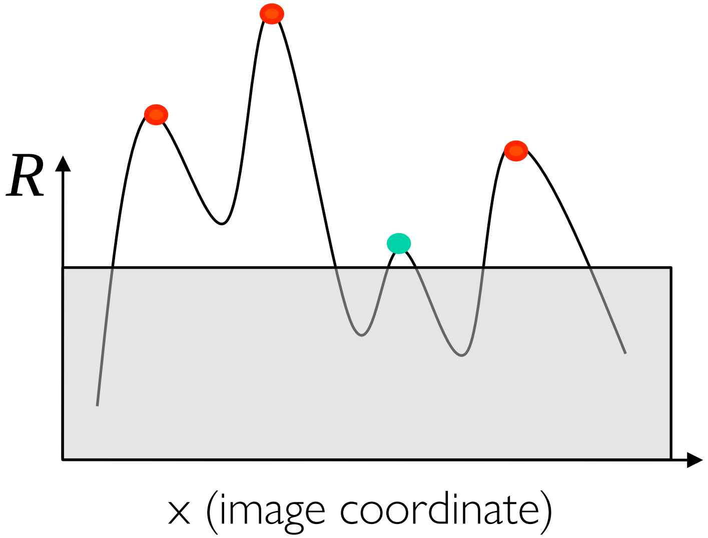


- Corner response R is invariant to image rotation

Recap: Harris corner response is invariant to intensity changes

- Partial invariance to **affine intensity** change
  - Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$
  - Intensity scale : $I \rightarrow a\, I$





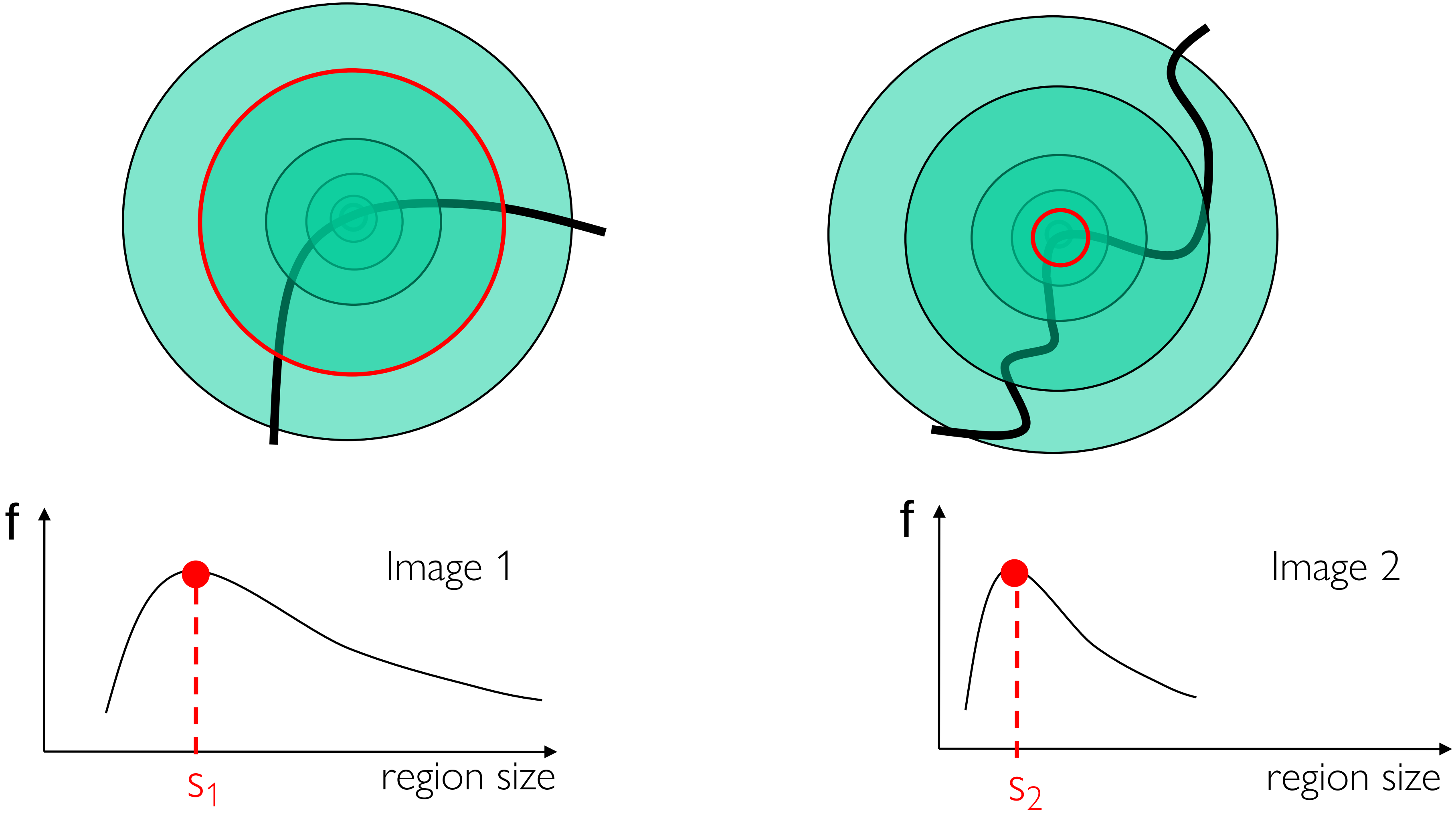
Multi-scale detection

- The Harris detector is not invariant to scale change

- How can we make a feature detector scale-invariant?
- How can we automatically select the scale?

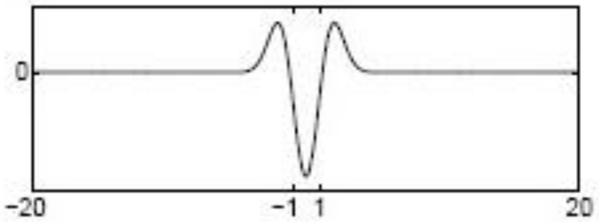
Multi-scale keypoint detection

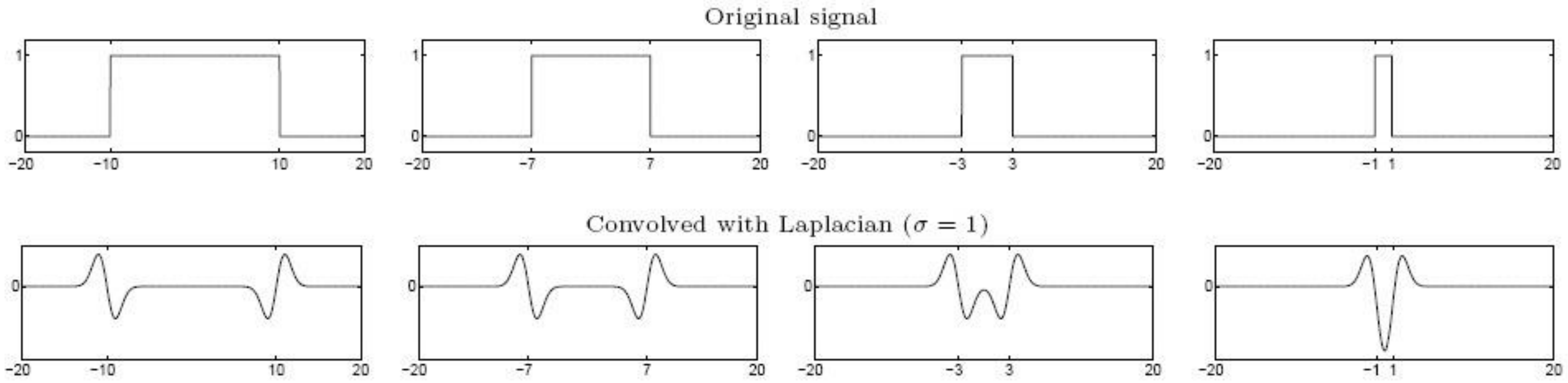
- Find local maxima in both **position** and **scale**



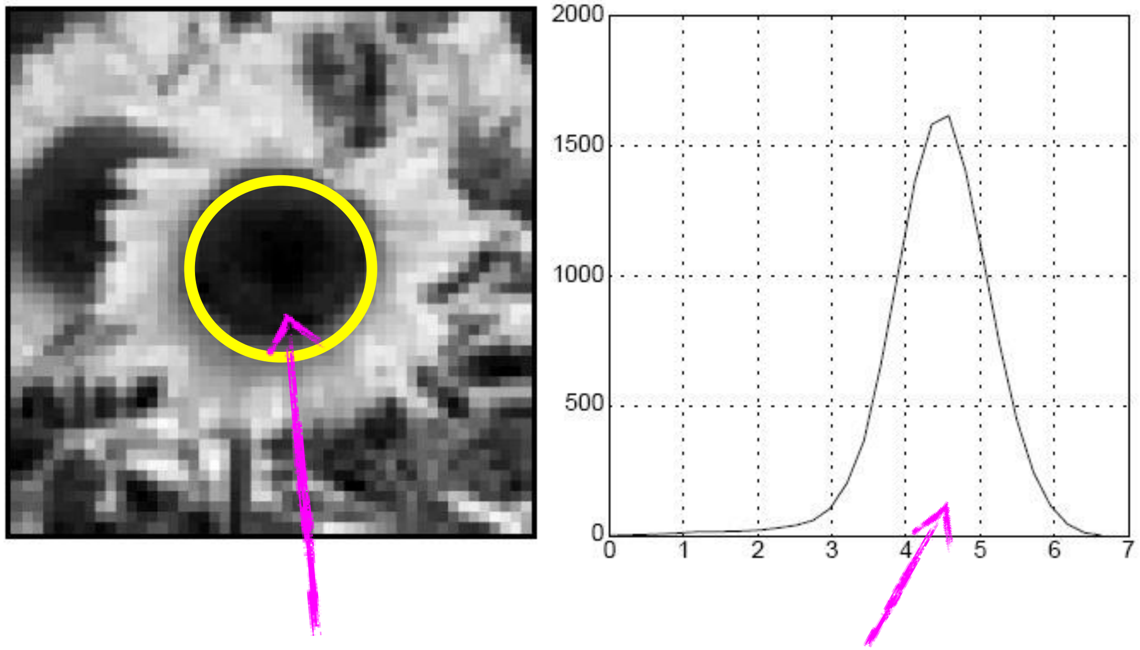
- Formaly highest response
  when the signal has the same **characteristic scale** as the filter

Laplacian filter





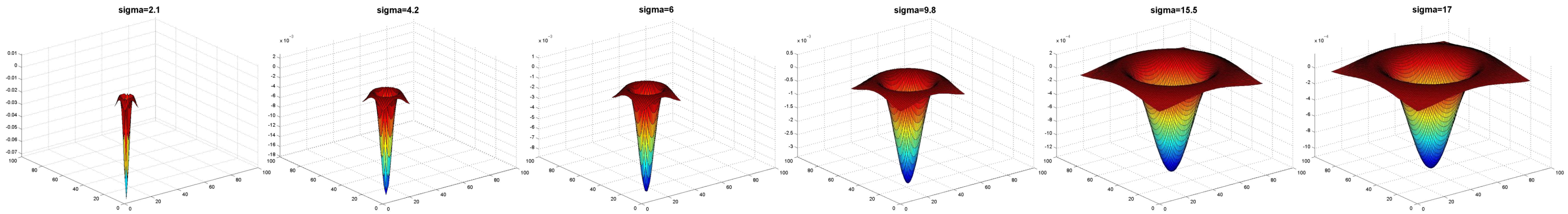
- characteristic scale - the scale that produces peak filter response



characteristic scale

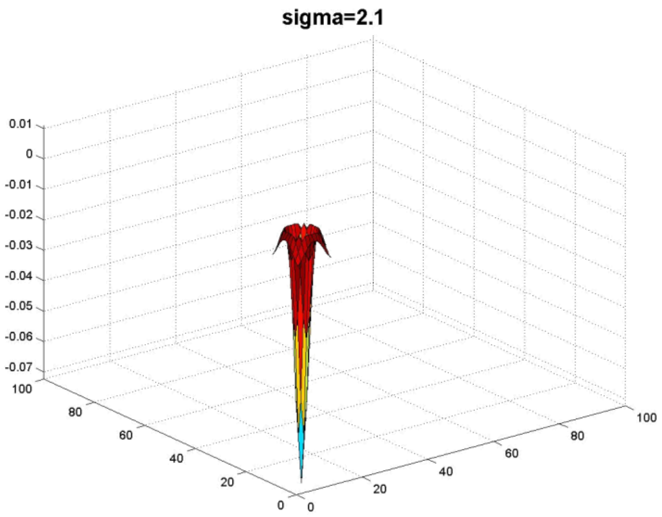
- we need to search over characteristic scales

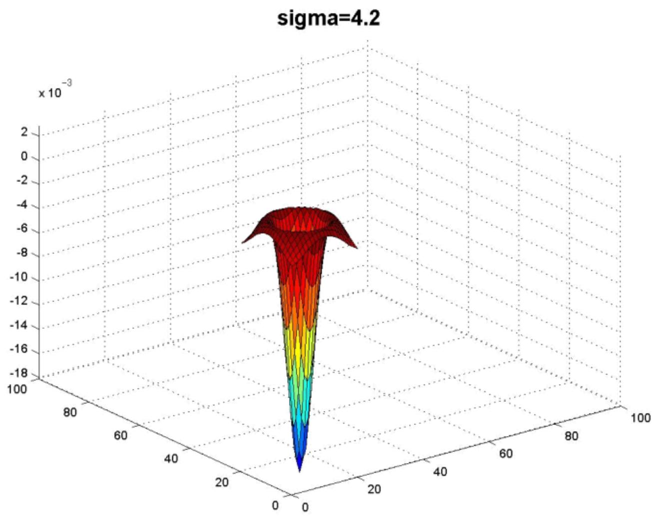
- What happens if you apply different Laplacian filters?

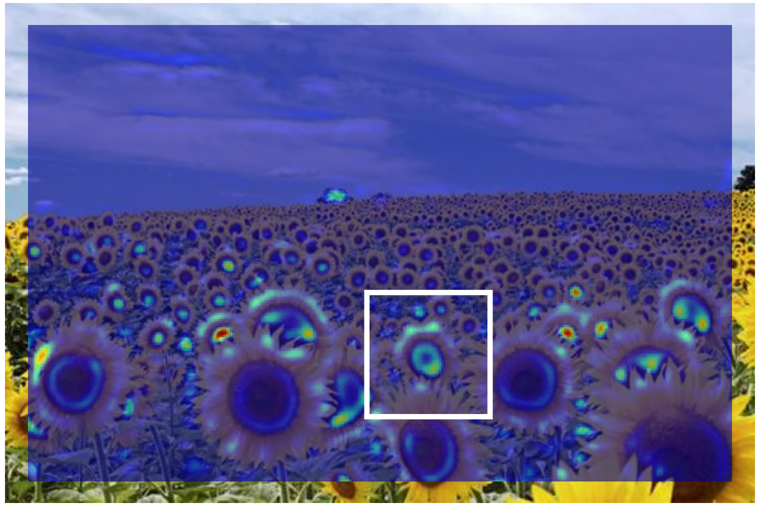


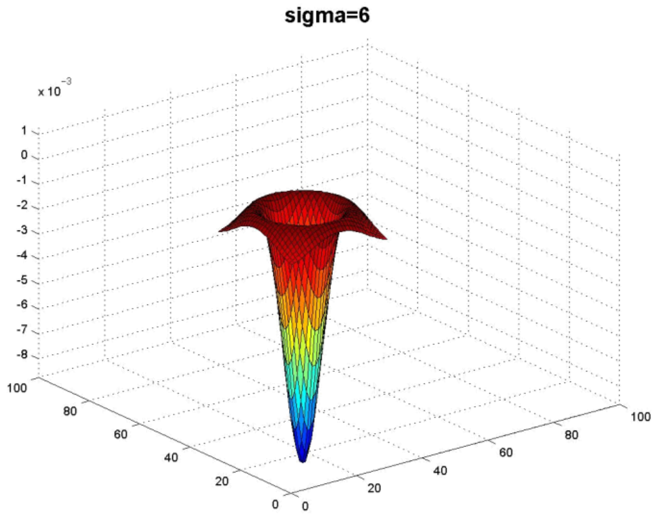
Full size

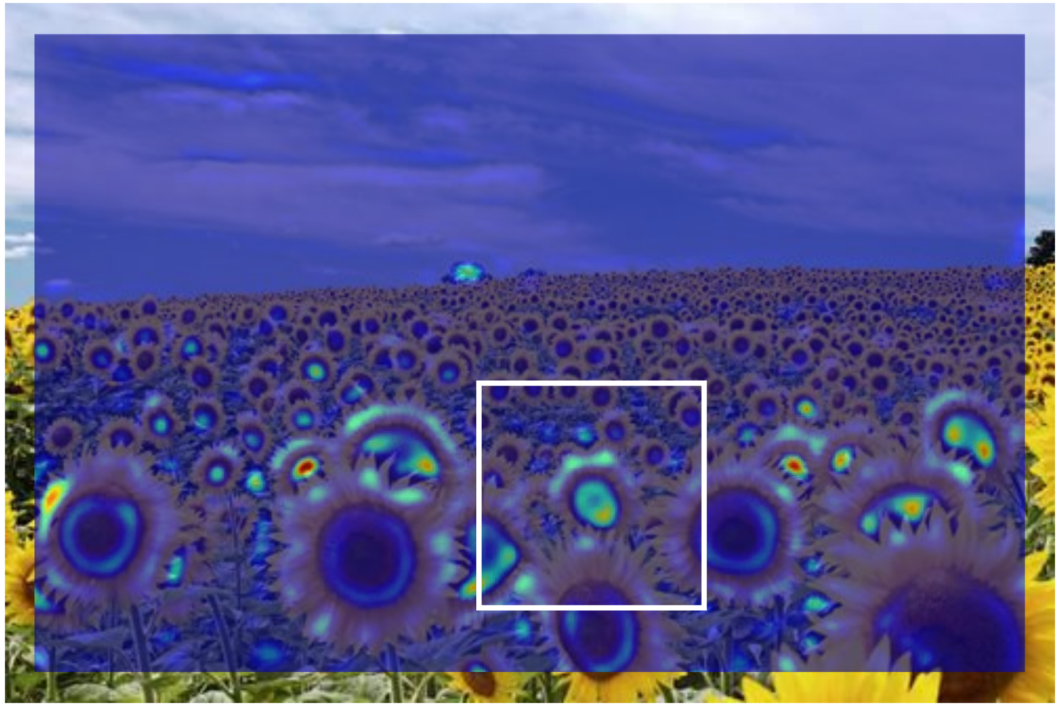
3/4 size


sigma=2.1
0.01
0
-0.01
-0.02
-0.03
-0.04
-0.05
-0.06
-0.07
100
80
60
40
20
0
0
20
40
60
80
100

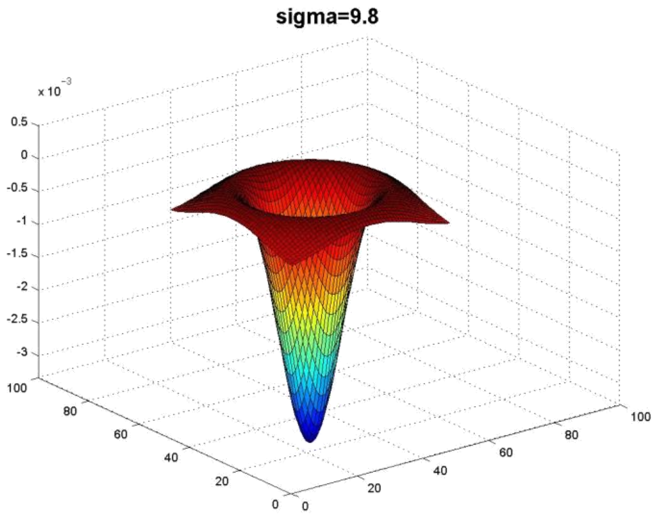

sigma=4.2
x 10^-3
2
0
-2
-4
-6
-8
-10
-12
-14
-16
-18
100
80
60
40
20
0
0
20
40
60
80
100



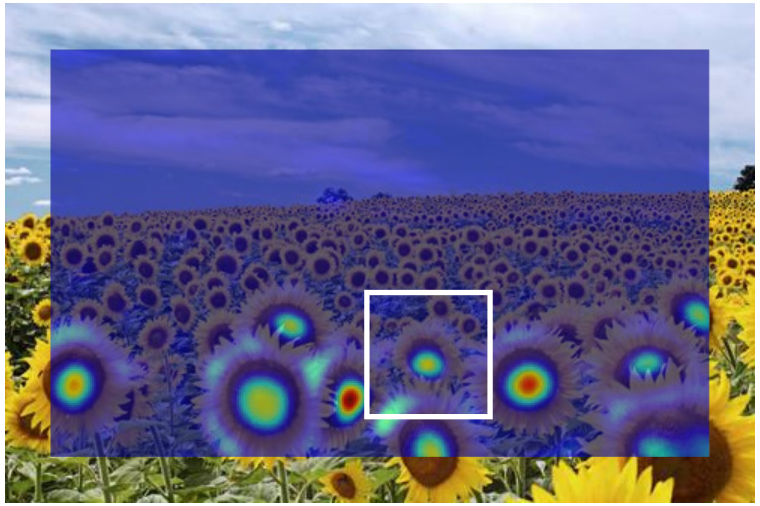

sigma=6
x 10^-3
1
0
-1
-2
-3
-4
-5
-6
-7
-8
100
80
60
40
20
0
0
20
40
60
80
100

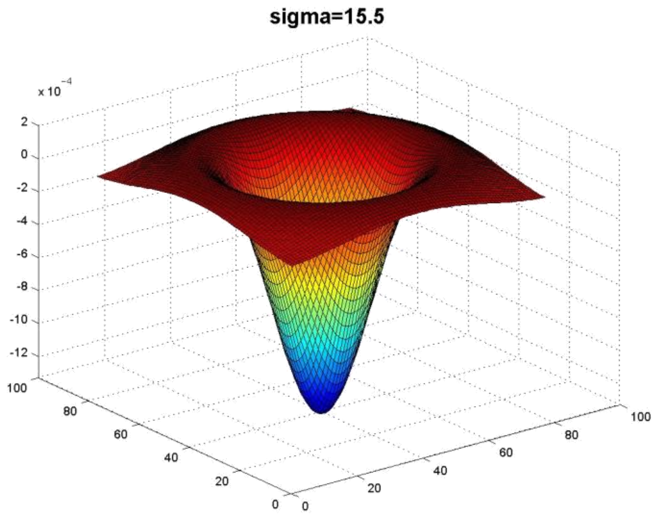


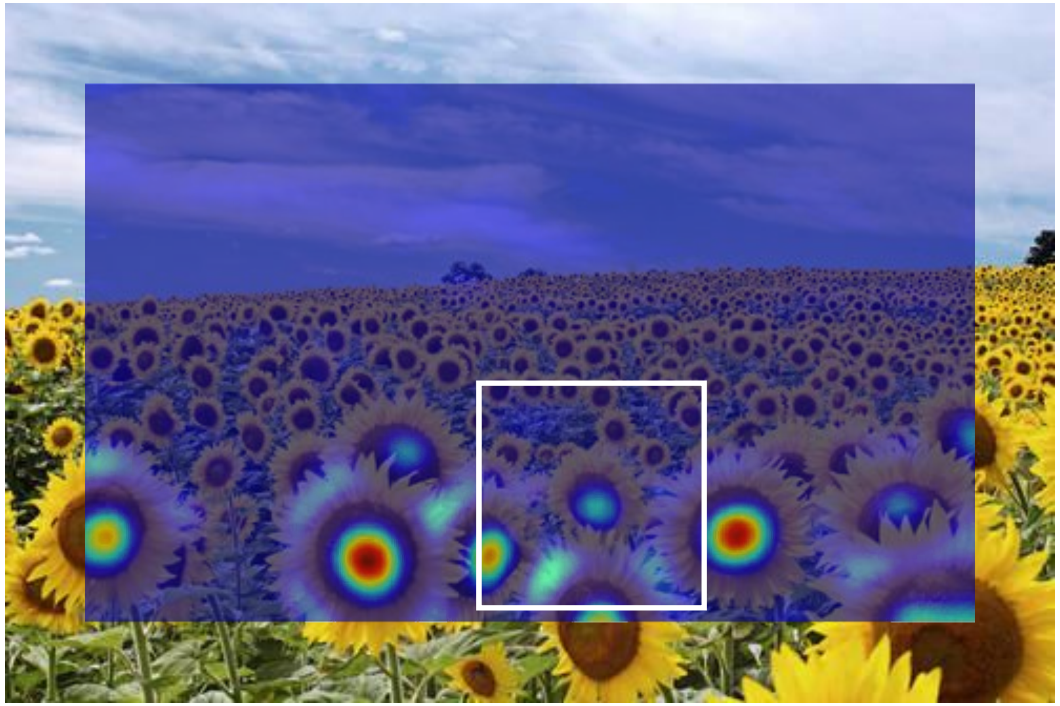

peak!

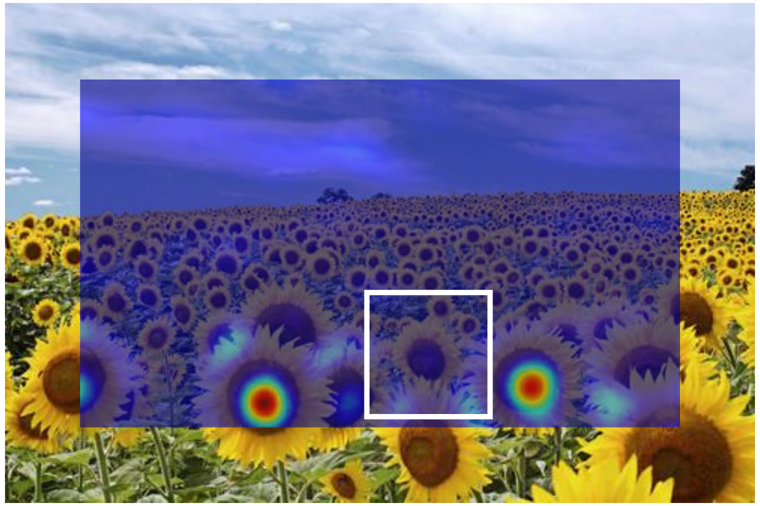

sigma=9.8

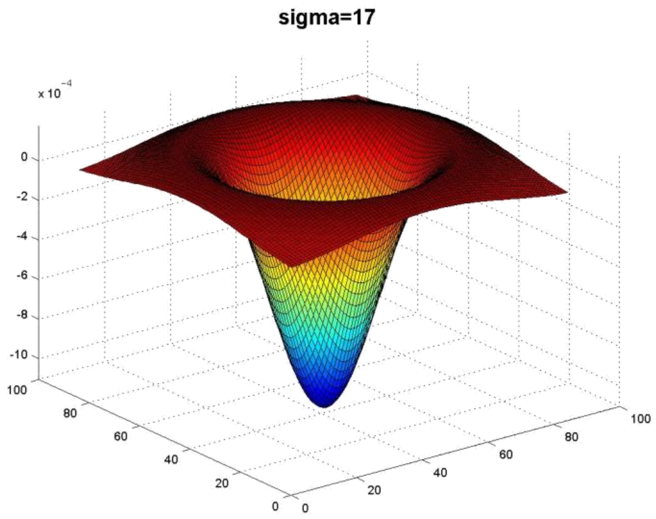

peak!

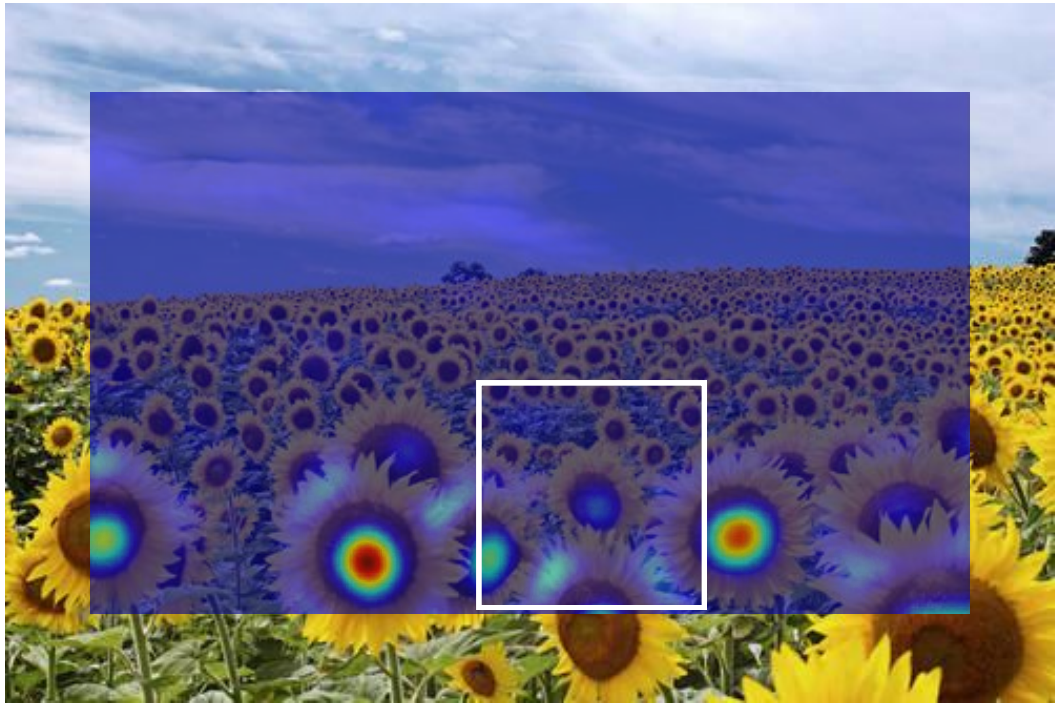


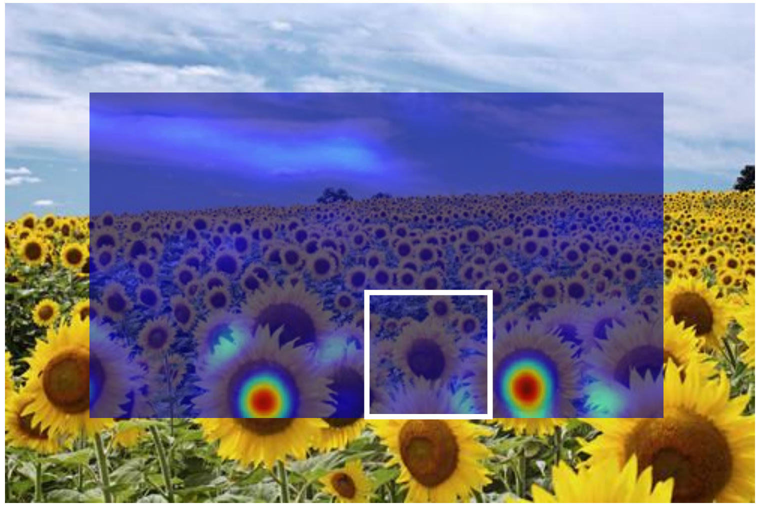

sigma=15.5
x 10^-4
2
0
-2
-4
-6
-8
-10
-12
100
80
60
40
20
0
0
20
40
60
80
100

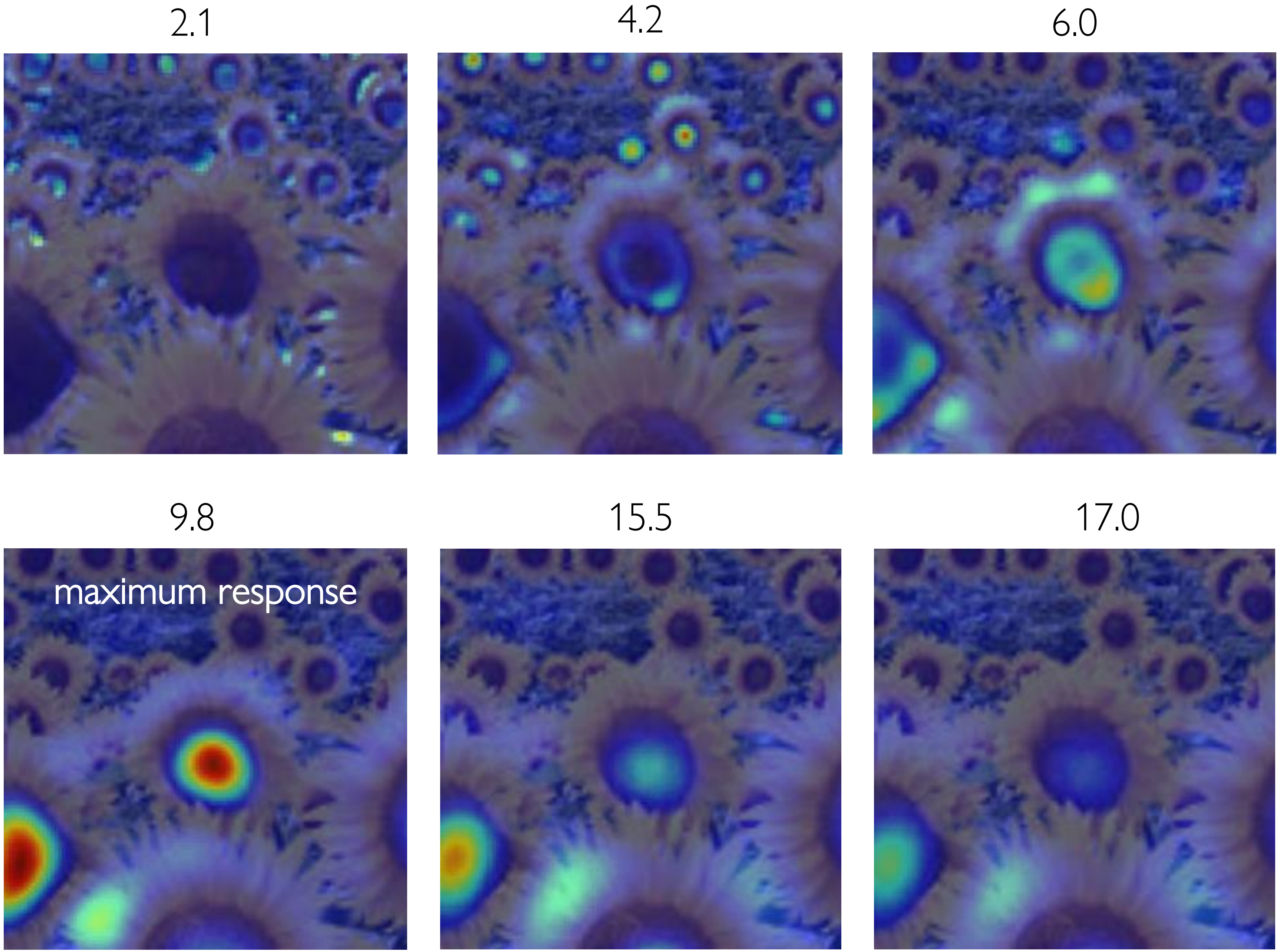




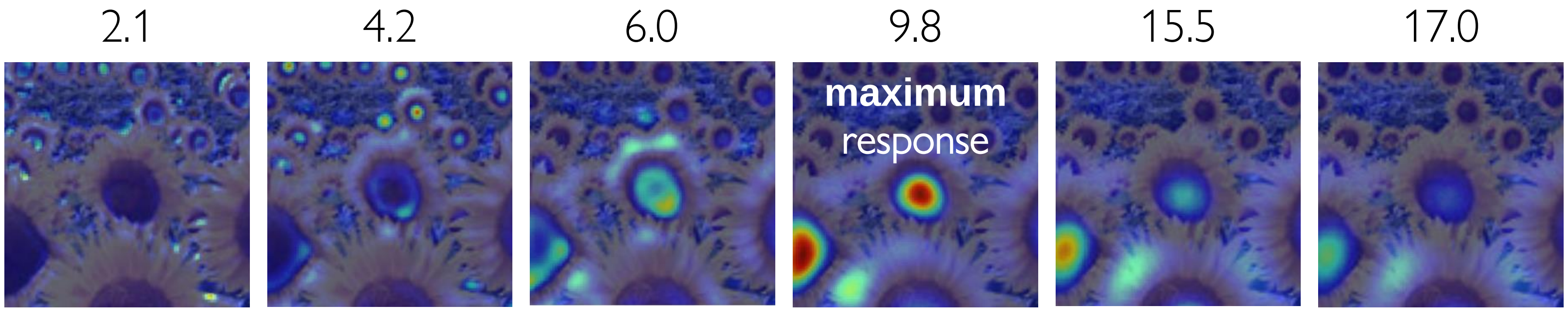

sigma=17
x 10^-4
0
-2
-4
-6
-8
-10
100
80
60
40
20
0
0
20
40
60
80
100



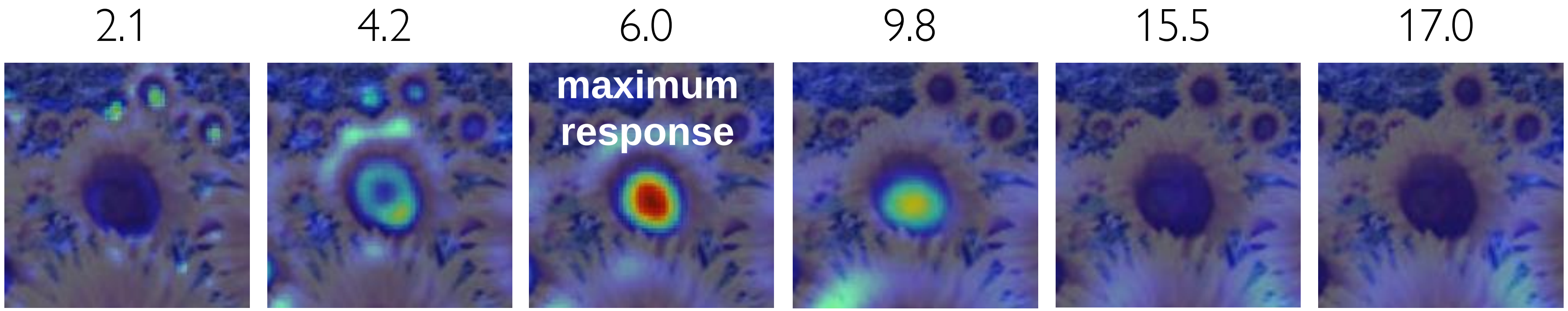



2.1
4.2
6.0
9.8
maximum response
15.5
17.0

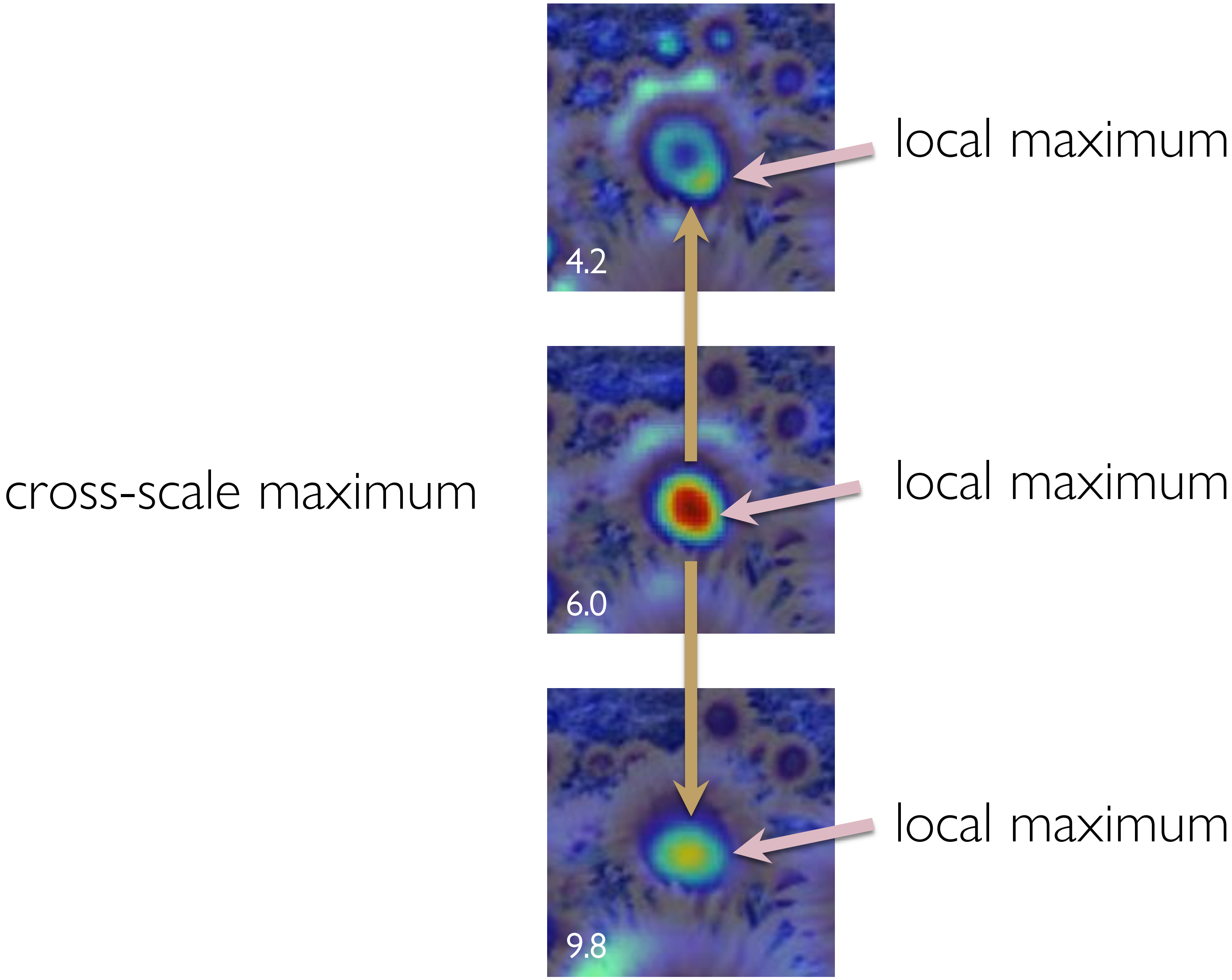
Optimal scale



Full size image



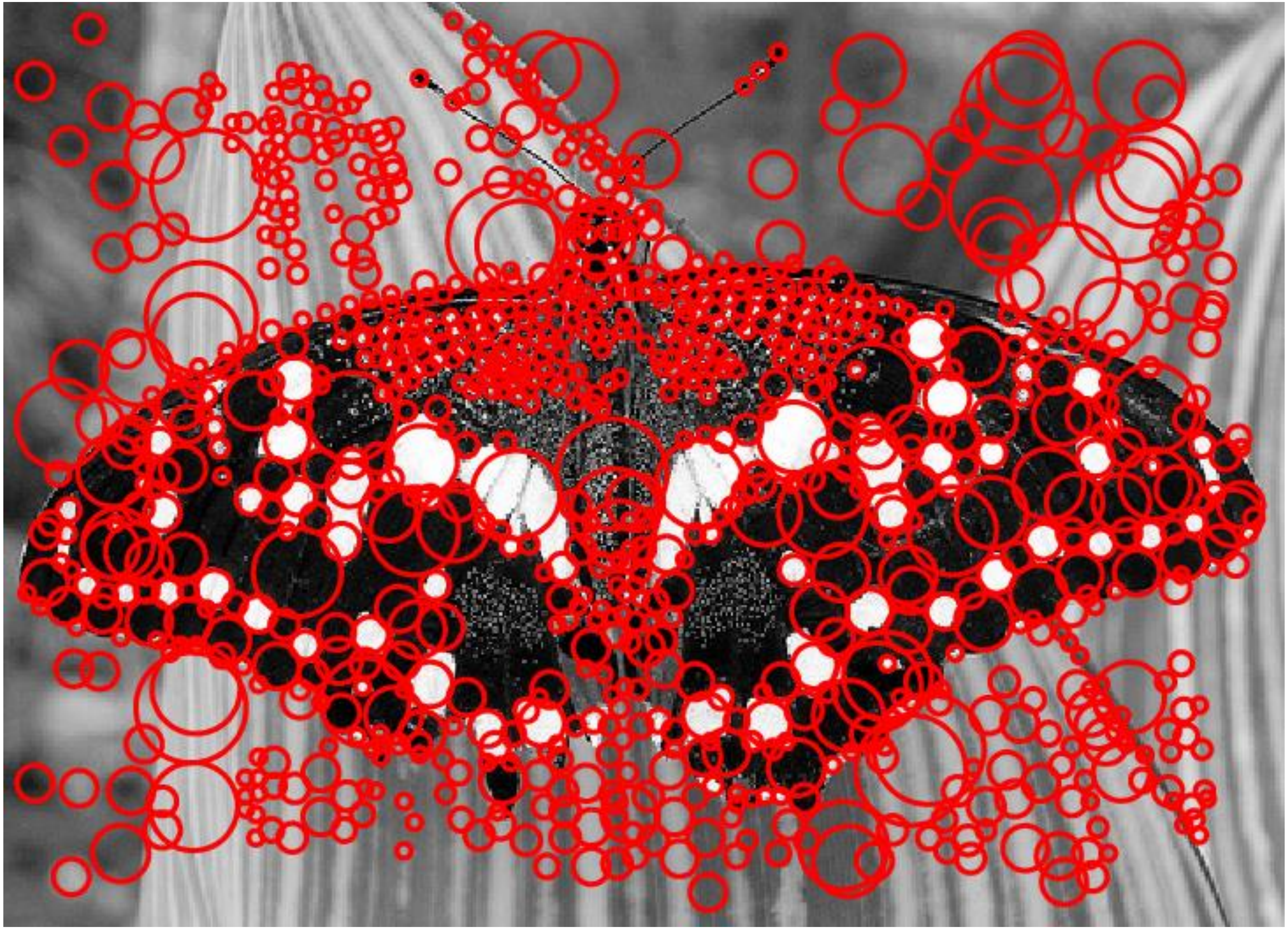
3/4 size image


local maximum
4.2
cross-scale maximum
local maximum
6.0
local maximum
9.8

How would you implement scale selection?

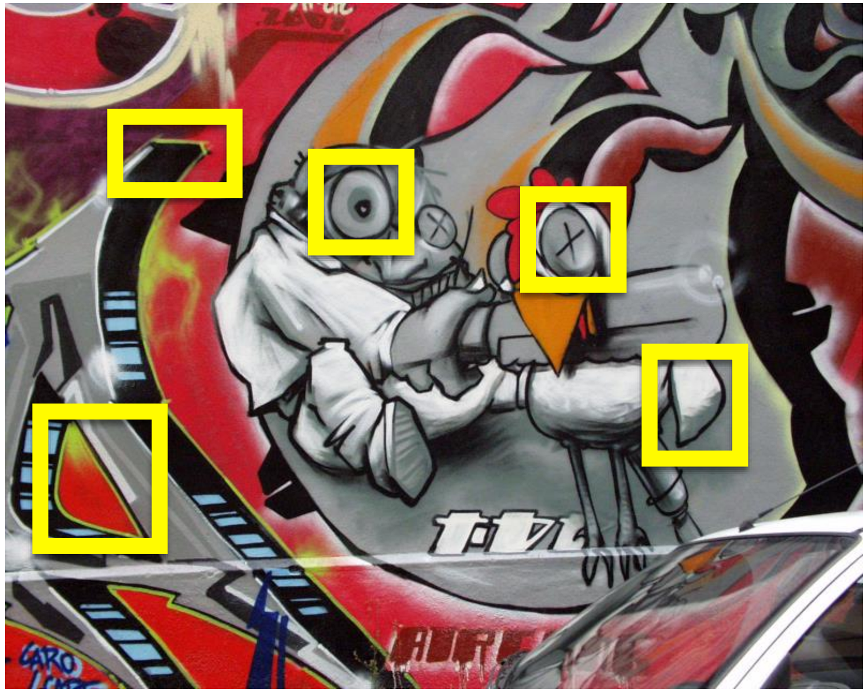
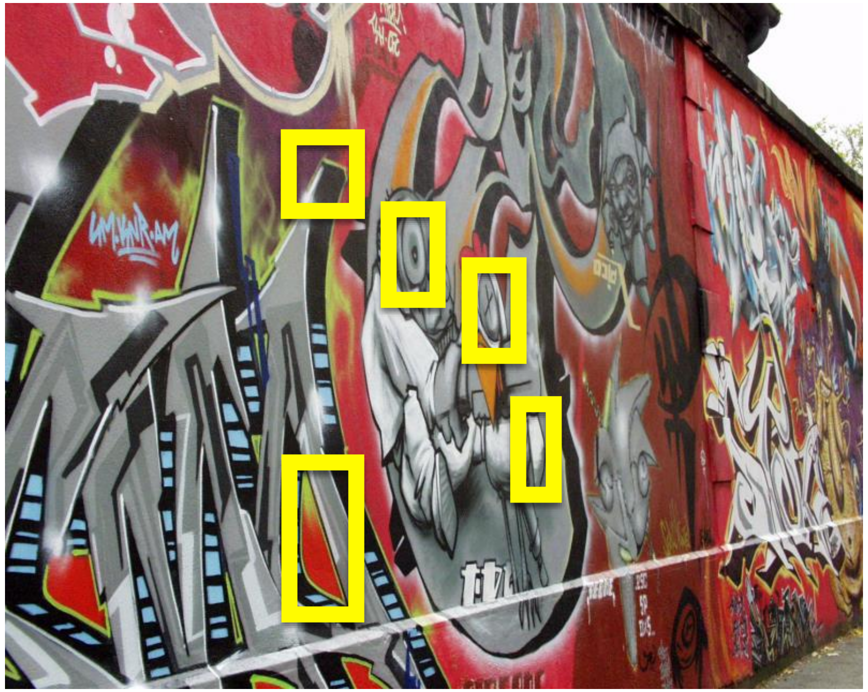
- For each level of the Gaussian pyramid
  - compute feature response (e.g. Harris, Laplacian)

- For each level of the Gaussian pyramid
  - if local maximum and cross-scale
  - **save** scale and location of feature (x, y, s)



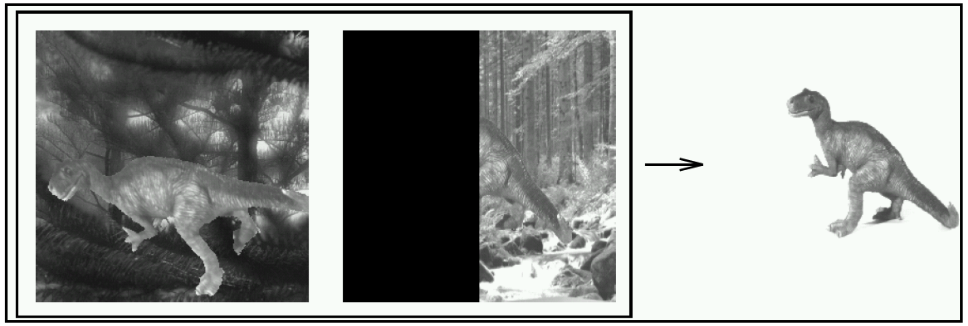
Feature Descriptors

Why do we need feature descriptors?

If we know where the good features are,
how do we match them?

Object instance recognition



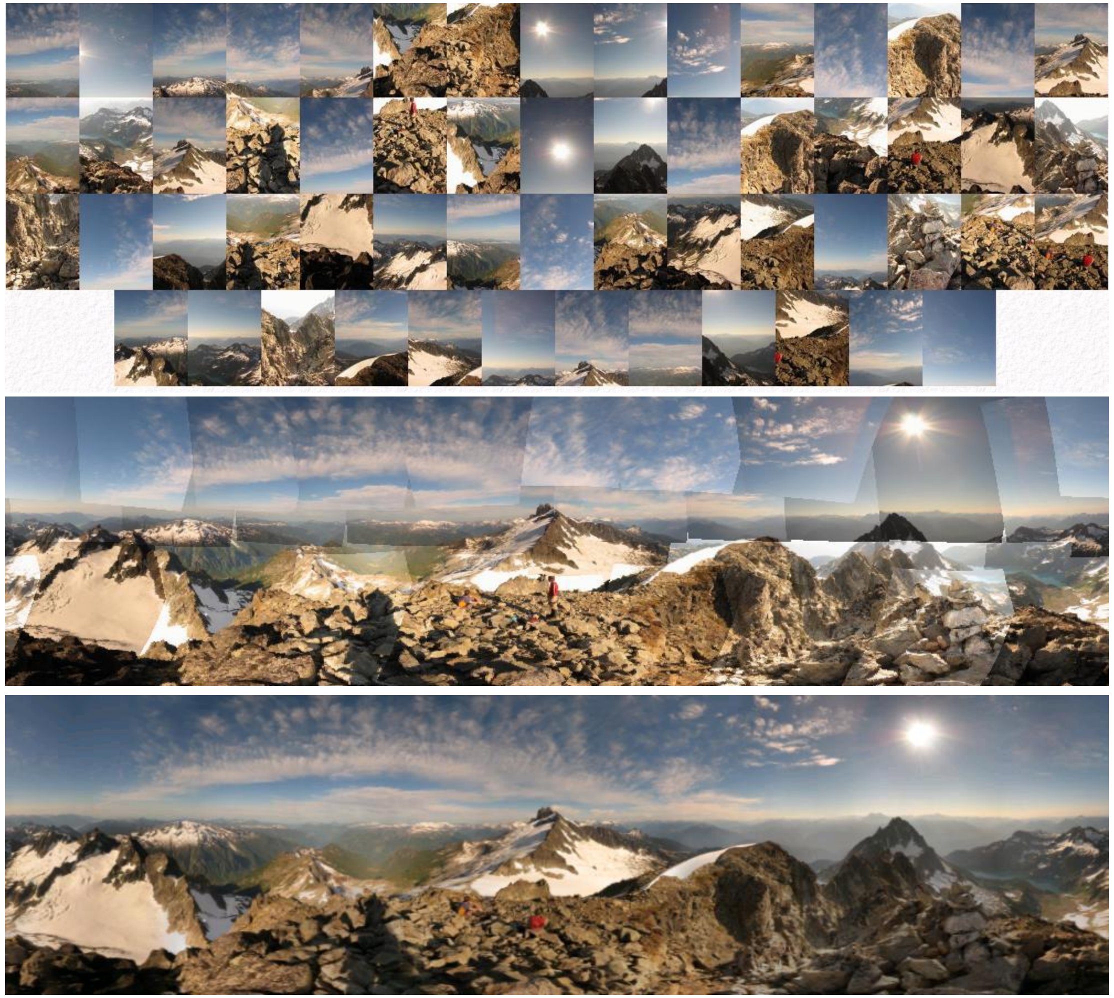
Schmid and Mohr 1997

Sivic and Zisserman, 2003

Rothganger et al. 2003

Lowe 2002

Image mosaicing



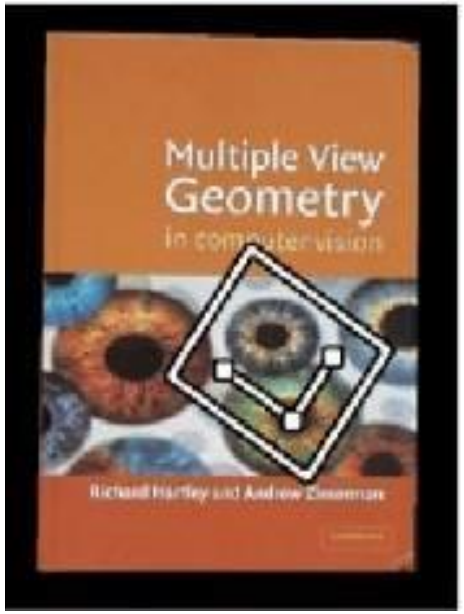
How do we describe an image patch?

Patches with similar content should have similar descriptors.

Designing Feature Descriptors

Photometric transformations

Geometric transformations



Objects will appear at different scales, translation and rotation

What is the best descriptor for an image feature?

Image patch

- Just use the pixel values of the patch

- Perfectly fine if geometry and appearance is unchanged (a.k.a. template matching)

Image patch

- Just use the pixel values of the patch

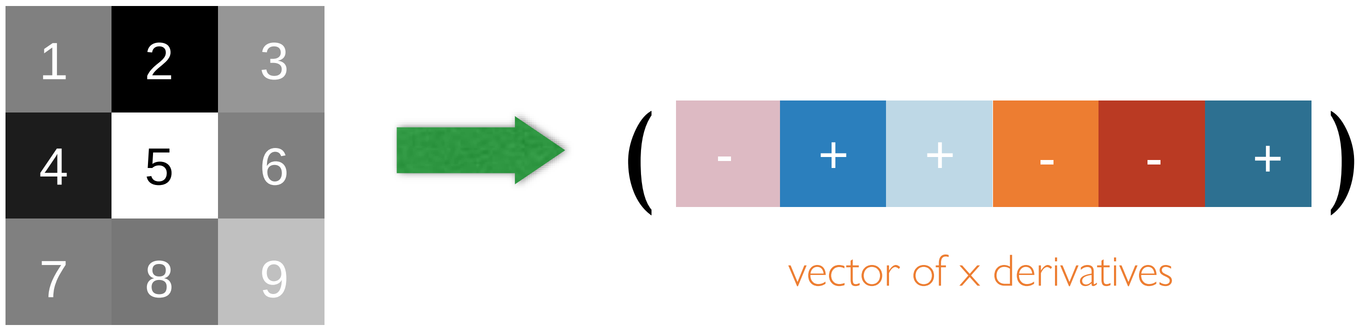
- Perfectly fine if geometry and appearance is unchanged (a.k.a. template matching)

What are the problems?

How can you be less sensitive to absolute intensity values?

Image gradients

- Use pixel differences



vector of x derivatives

'binary descriptor'

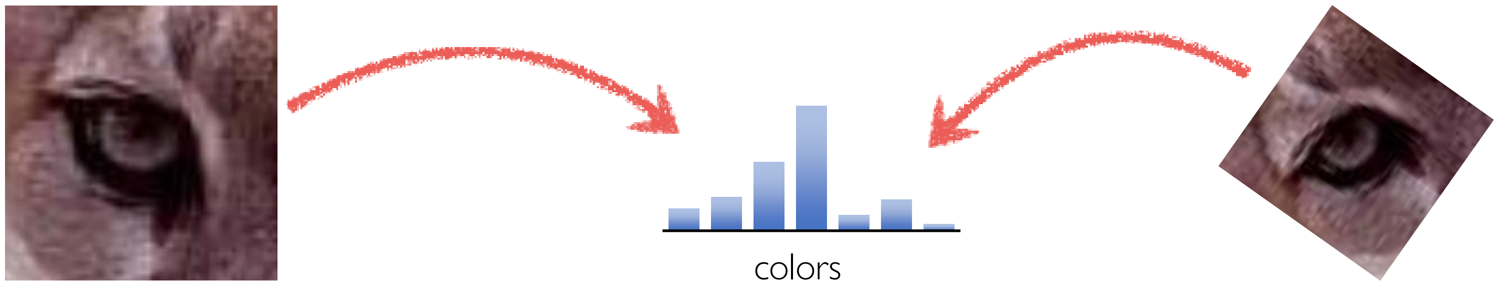
- Feature is invariant to absolute intensity values

What are the problems?

How can you be less sensitive to deformations?

Color histogram

- Count the colors in the image using a histogram

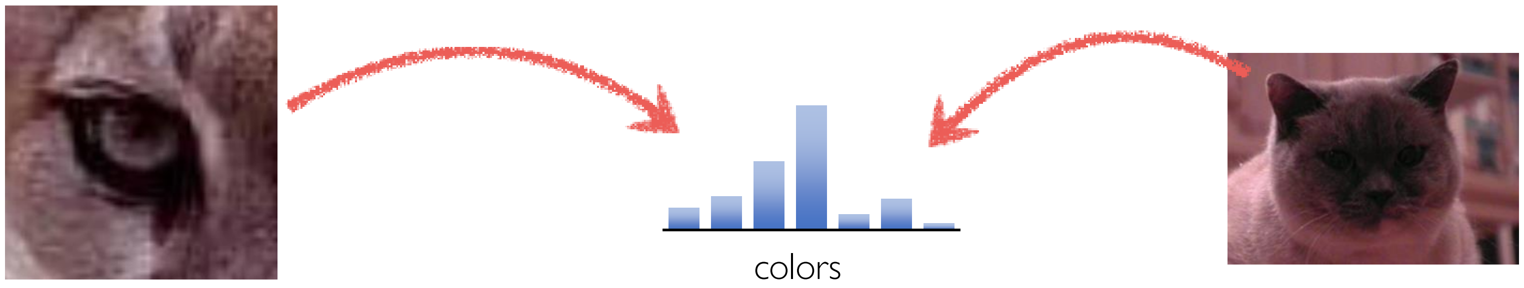


- Invariant to changes in scale and rotation

What are the problems?

Color histogram

- Count the colors in the image using a histogram



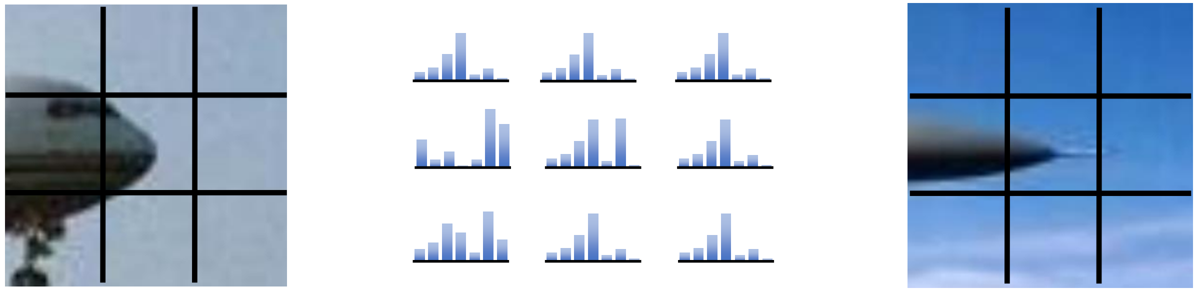
- Invariant to changes in scale and rotation

What are the problems?

How can you be more sensitive to spatial layout?

Spatial histograms

- Compute histograms over spatial ‘cells’

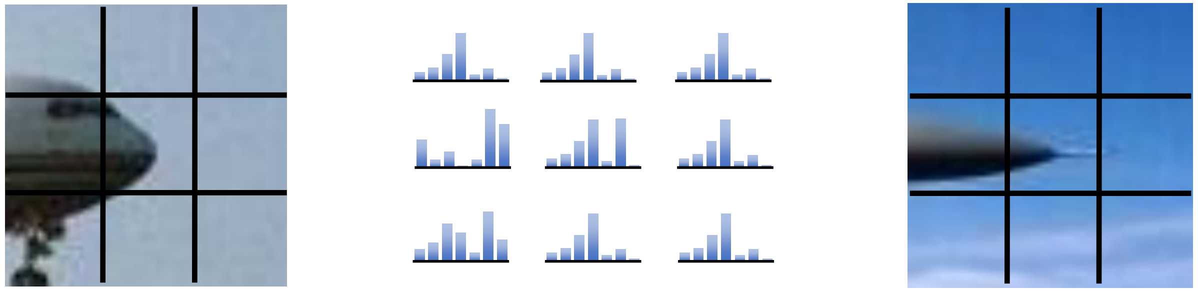


- Retains rough spatial layout
- Some invariance to deformations

What are the problems?

Spatial histograms

- Compute histograms over spatial 'cells'



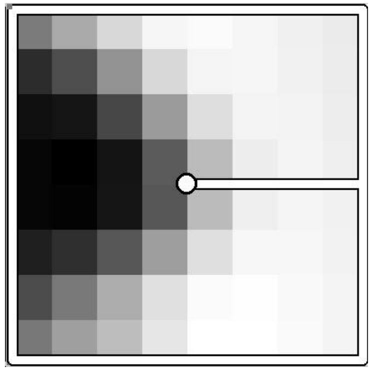
- Retains rough spatial layout
- Some invariance to deformations

What are the problems?

How can you be completely invariant to rotation?

Orientation normalization

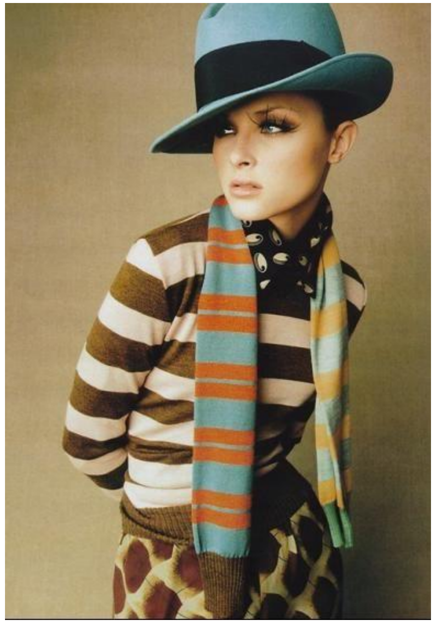
- Use the dominant image gradient direction to normalize the orientation of the patch



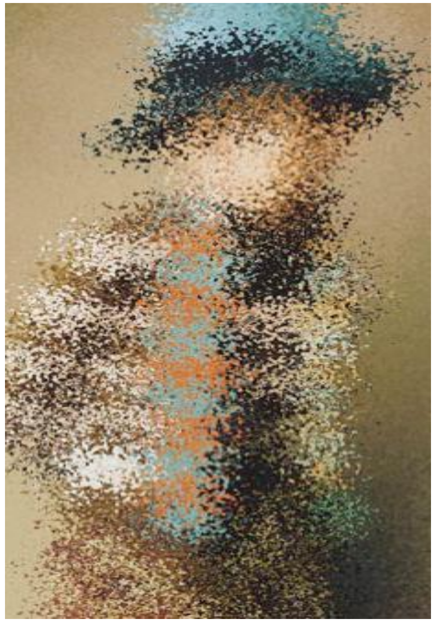
Save the orientation angle θ along with (x, y, s)

What are the problems?

Discriminative power



Raw pixels



Locally orderless

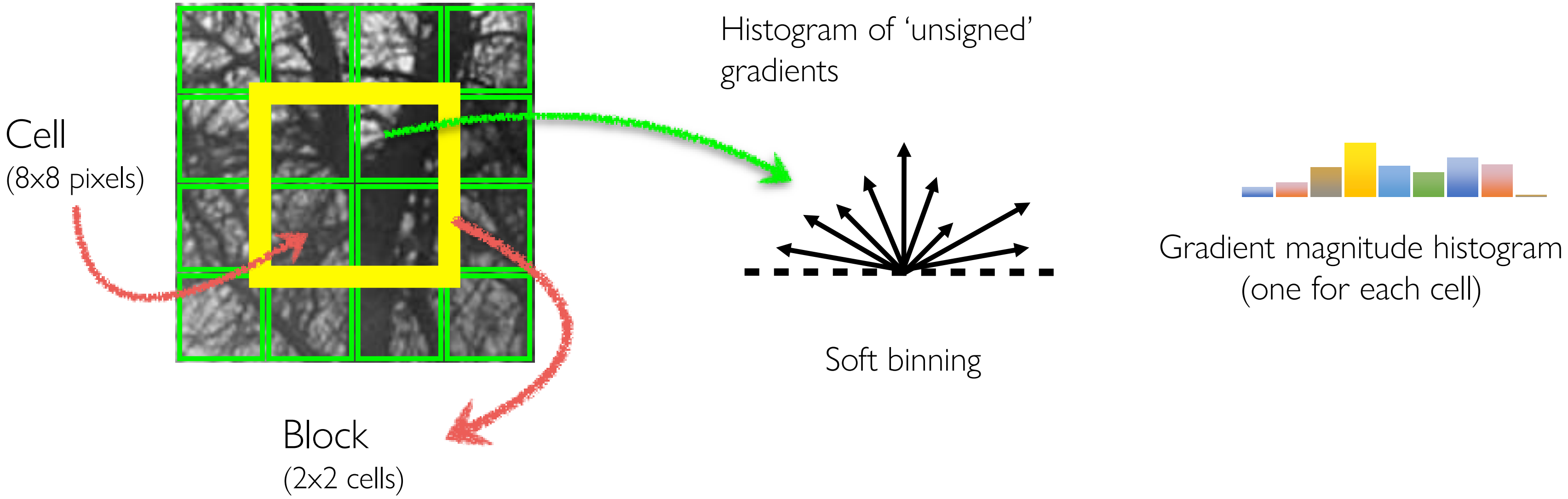
Global histogram

Generalization power

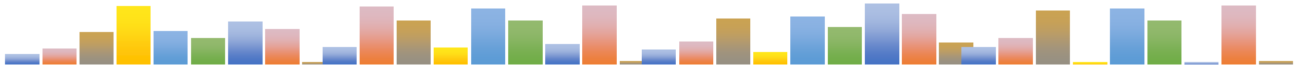
HOG descriptor

HOG

- Dalal, Triggs. **Histograms of Oriented Gradients** for Human Detection. CVPR, 2005



Concatenate and L-2 normalization



Single scale, no dominant orientation

Pedestrian detection

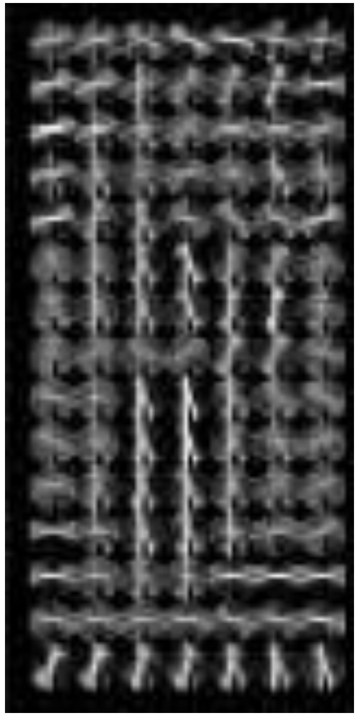
1 cell step size

128 pixels
16 cells
15 blocks

15 x 7 x 4 x 36
= 3780

64 pixels 8 cells 7 blocks

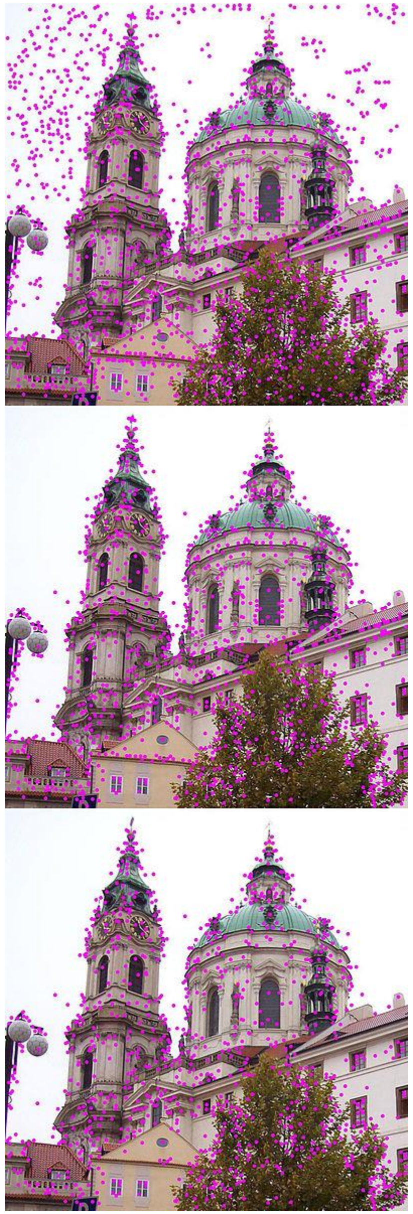
Visualization



http://chrisjmccormick.wordpress.com/2013/05/09/hog-person-detector-tutorial/

SIFT

SIFT (Scale Invariant Feature Transform)

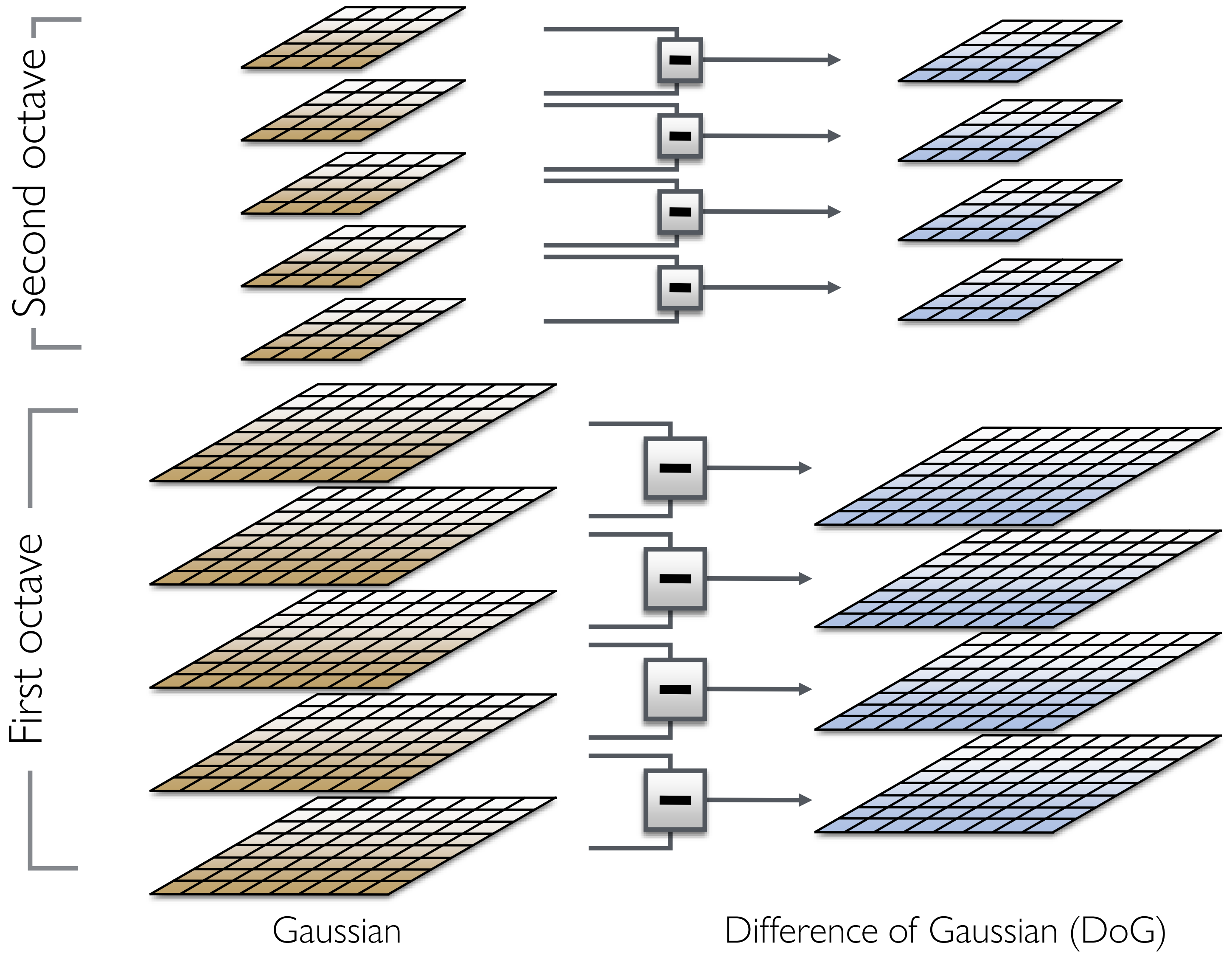


SIFT describes both a **detector** and **descriptor**

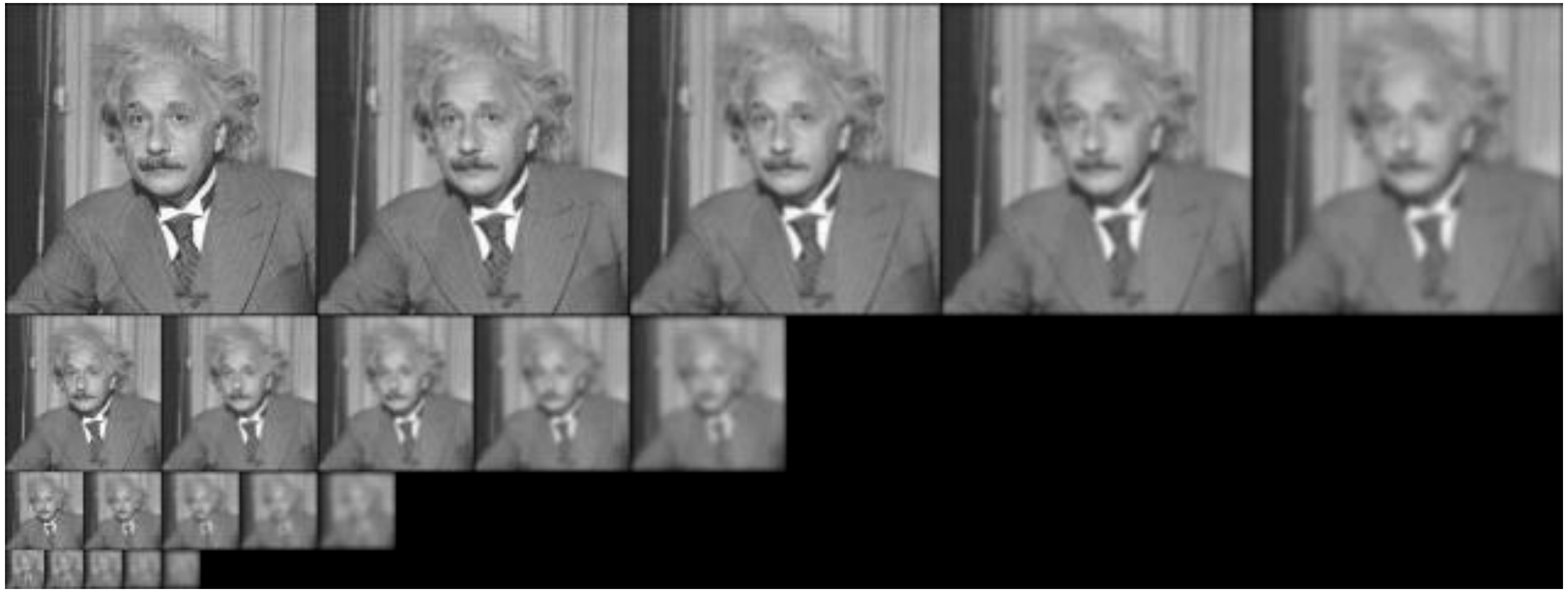
1. Multi-scale extrema detection
2. Keypoint localization
3. Orientation assignment
4. Keypoint descriptor

1. Multi-scale extrema detection

- Scale space



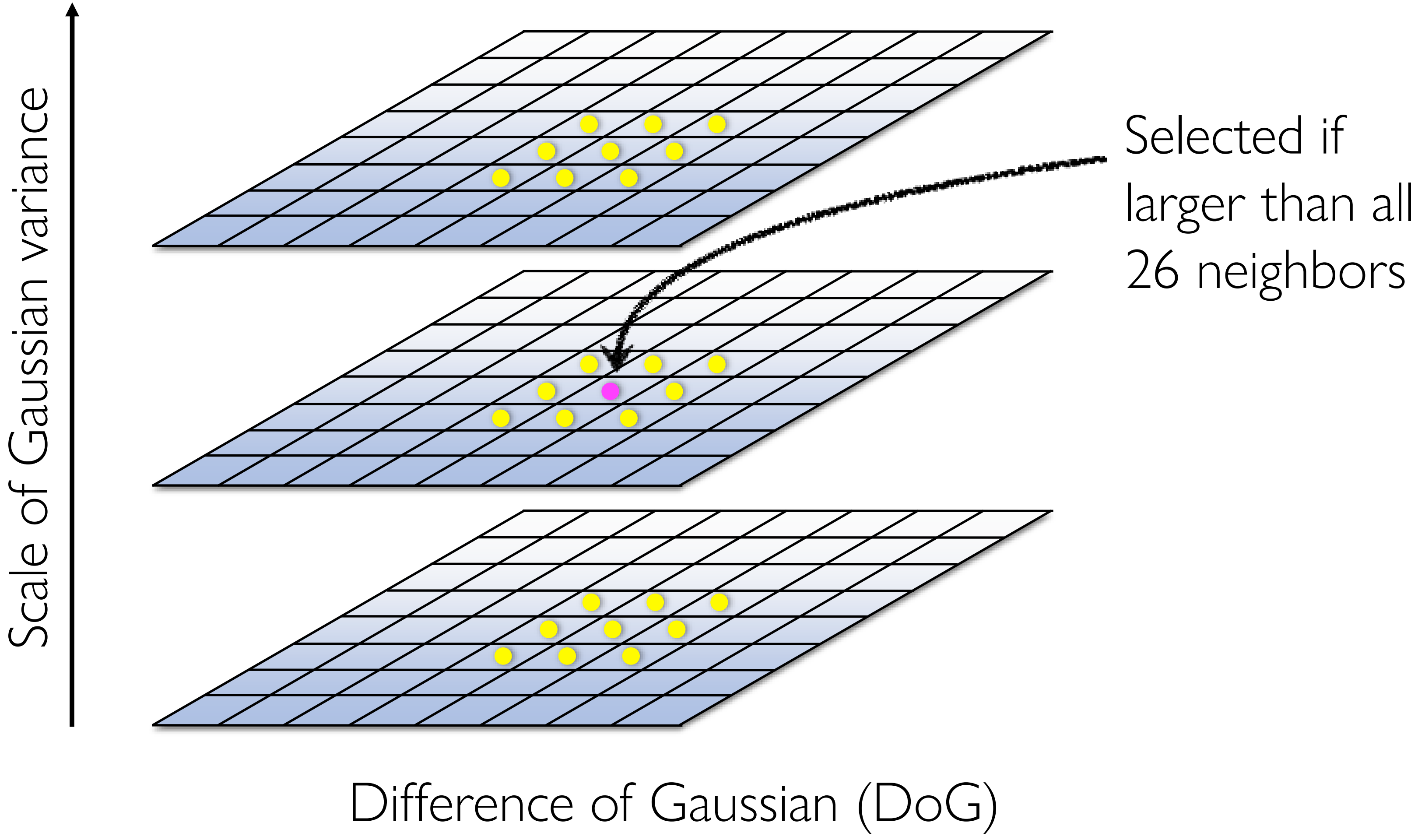
Why DoG?



Gaussian

Laplacian

Scale-space extrema



2. Keypoint localization

- 2nd order Taylor series approximation of DoG scale-space

$$f(\mathbf{x}) = f + \frac{\partial f}{\partial \mathbf{x}}^T \mathbf{x} + \frac{1}{2}\mathbf{x}^T \frac{\partial^2 f}{\partial \mathbf{x}^2}\mathbf{x}$$

$$\mathbf{x} = \{x, y, \sigma\}$$

- Take the derivative and solve for extrema

$$\mathbf{x}_m = -\frac{\partial^2 f}{\partial \mathbf{x}^2}^{-1} \frac{\partial f}{\partial \mathbf{x}}$$

- Additional tests to retain only strong features

3. Orientation assignment

- For a keypoint, L is the **Gaussian-smoothed** image with the closest scale,

$$m(x,y) = \sqrt{\underbrace{(L(x+1,y) - L(x-1,y))^2}_{\text{x-derivative}} + \underbrace{(L(x,y+1) - L(x,y-1))^2}_{\text{y-derivative}}}$$

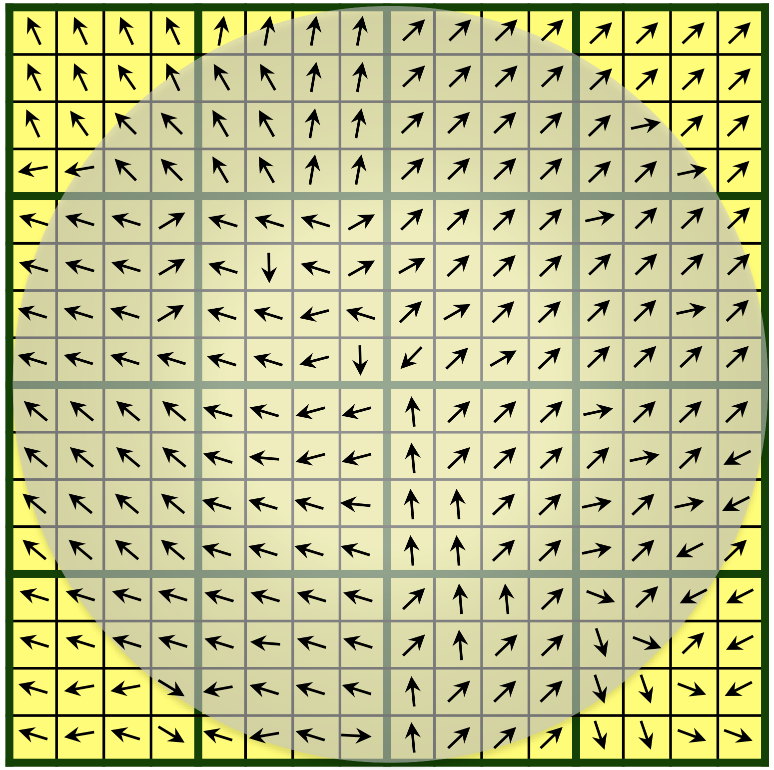
$$\theta(x,y) = \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y)))$$

- Detection process returns

$$\{\underbrace{x, y}_{\text{Location}}, \underbrace{\sigma}_{\text{Scale}}, \underbrace{\theta}_{\text{Orientation}}\}$$

4. Keypoint descriptor

Image Gradients
(4 x 4 pixel per cell, 4 x 4 cells)



Gaussian weighting
(sigma = half width)

SIFT descriptor
(16 cells x 8 directions = 128 dims)