Week 8

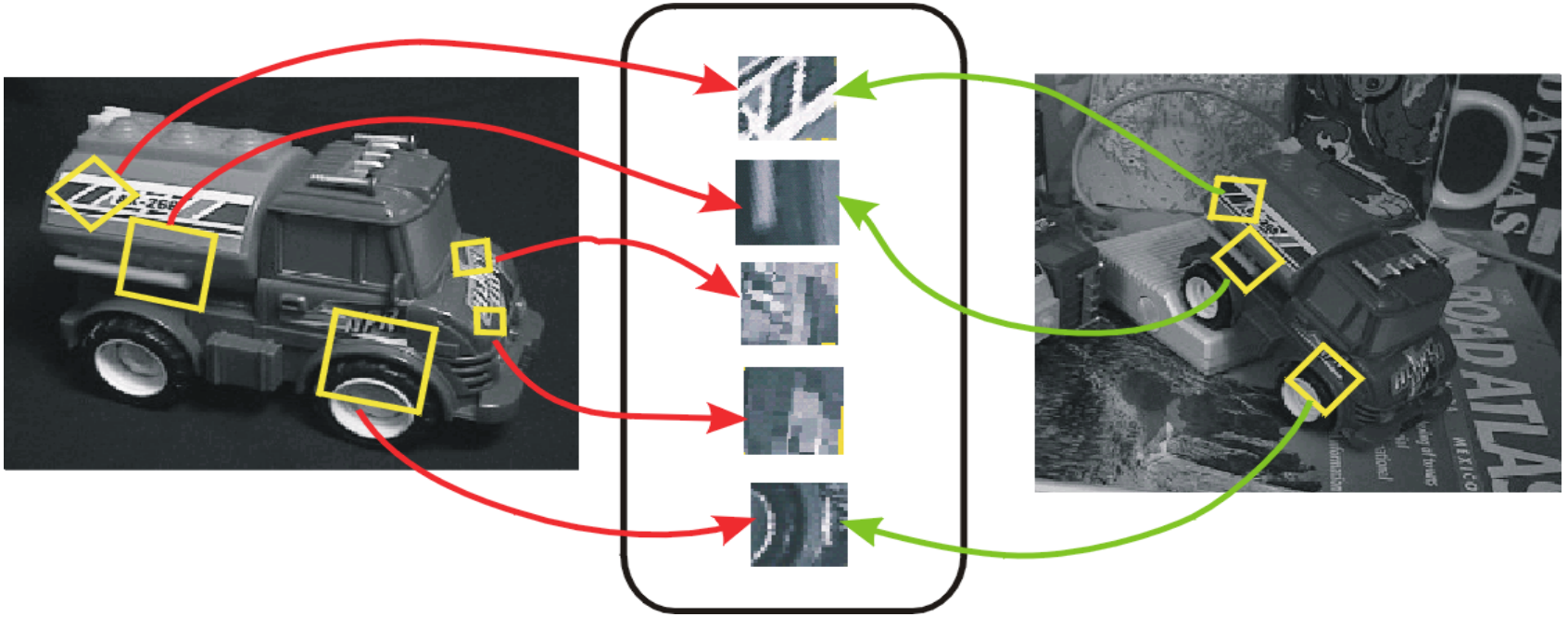
SIFT Features

Given a single object image, search images with that object!

Challenges

- Illumination change
- Orientation change
- Scale change

The Main Idea



SIFT – Scale Invariant Feature Transform

David Lowe
University of British Columbia

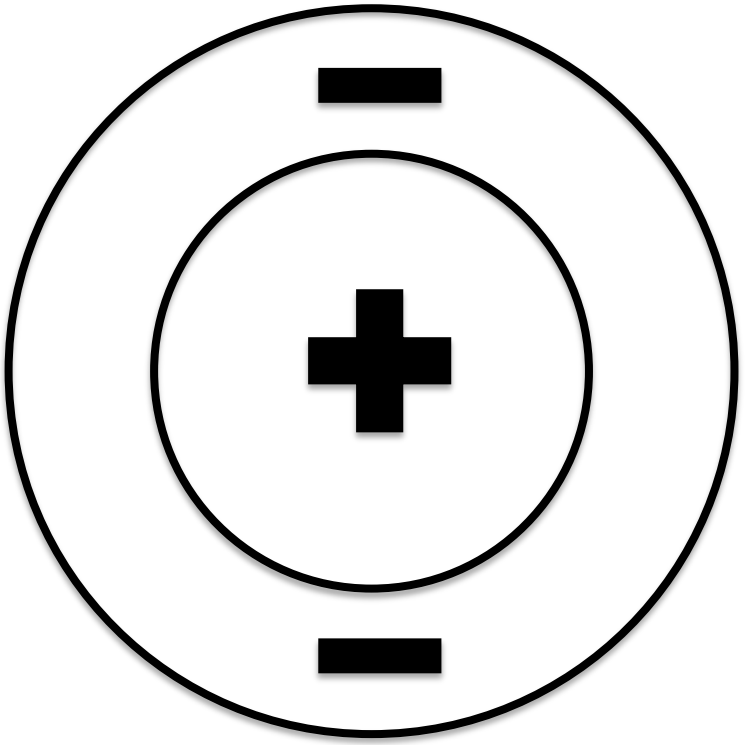
Main Steps

- Extract keypoints in example image
  - key point descriptors
- Find keypoints in the test image
  - keypoint descriptors
- Match keypoints across images
- If enough number of good matches are found, content is similar

Finding Keypoints

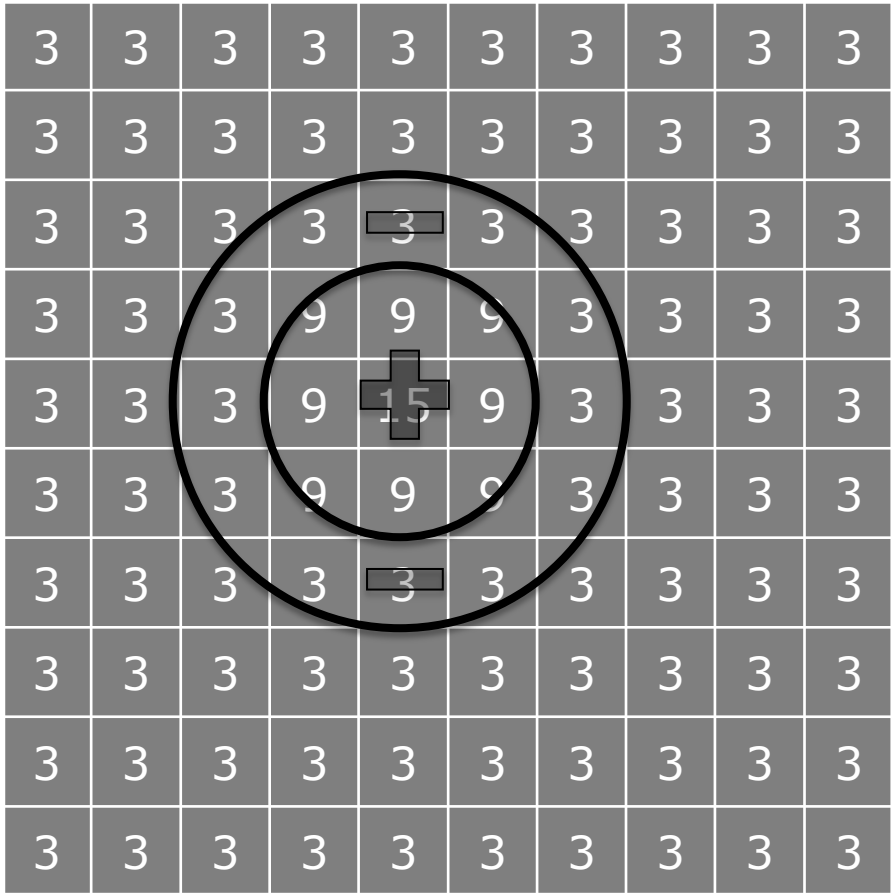
3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3
3	3	3	9	9	9	3	3	3	3
3	3	3	9	15	9	3	3	3	3
3	3	3	9	9	9	3	3	3	3
3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3

Center-Surround Approach



D=(a1+a2+a3…)
– (b1+b2+b3)

Finding Keypoints



D=(9+9+15..)
– (3+3+3)

Differential of Gaussian

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$$

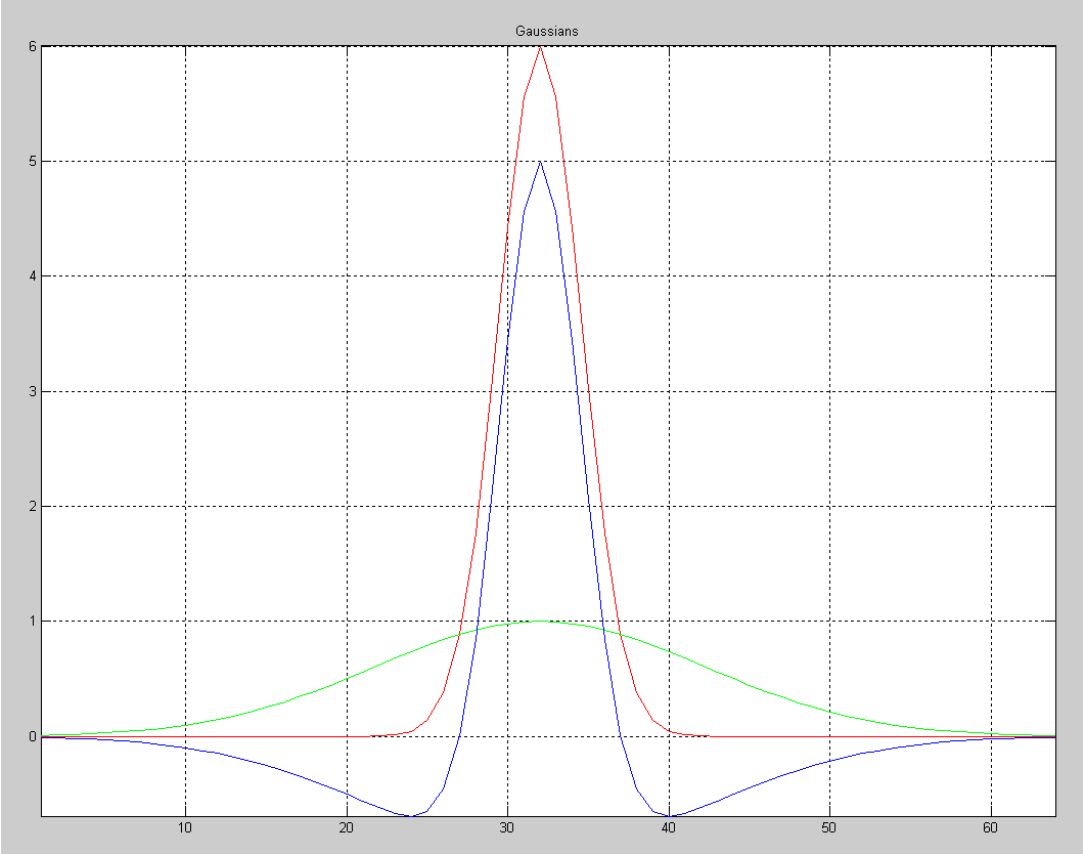
where

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

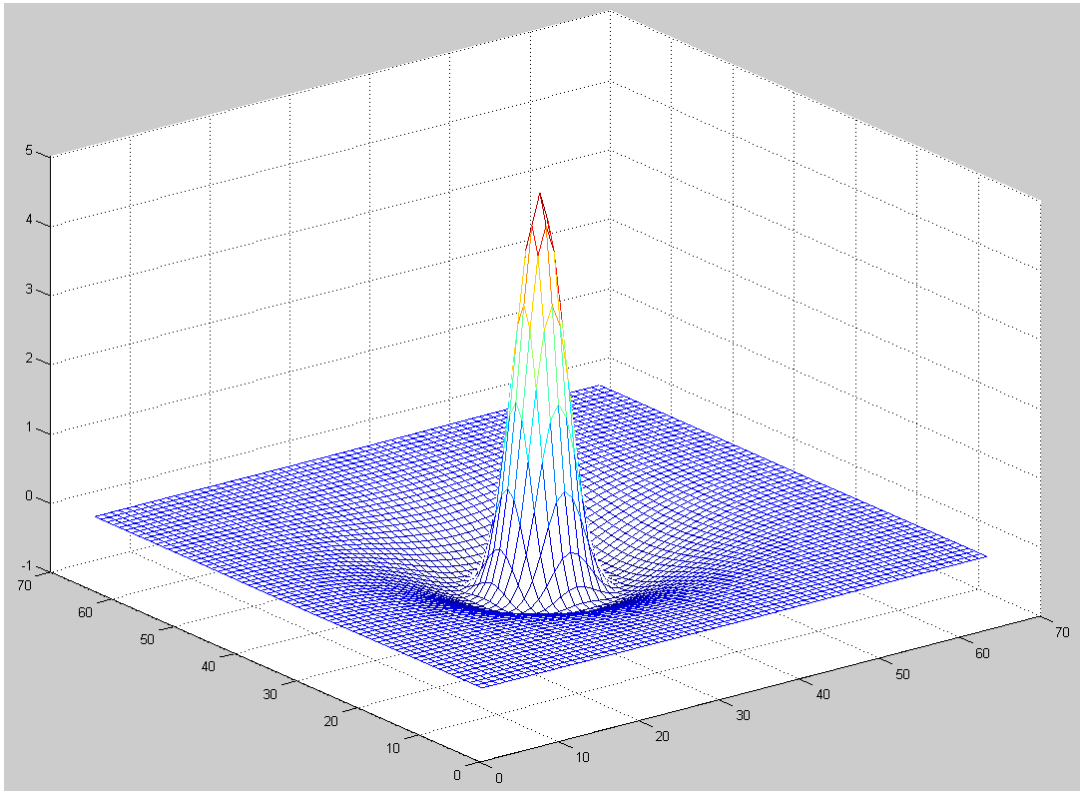
Example

- G = [-1 4 1]
- A = [1 2 3 3 5 3 3 2 1]

Differential of Gaussian



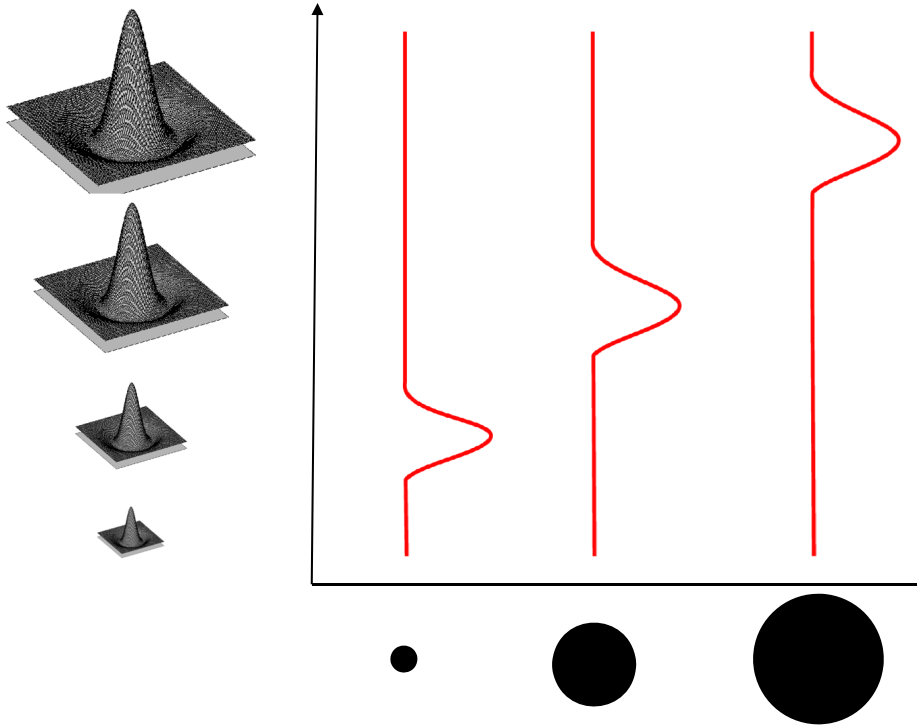
DoG in 3D

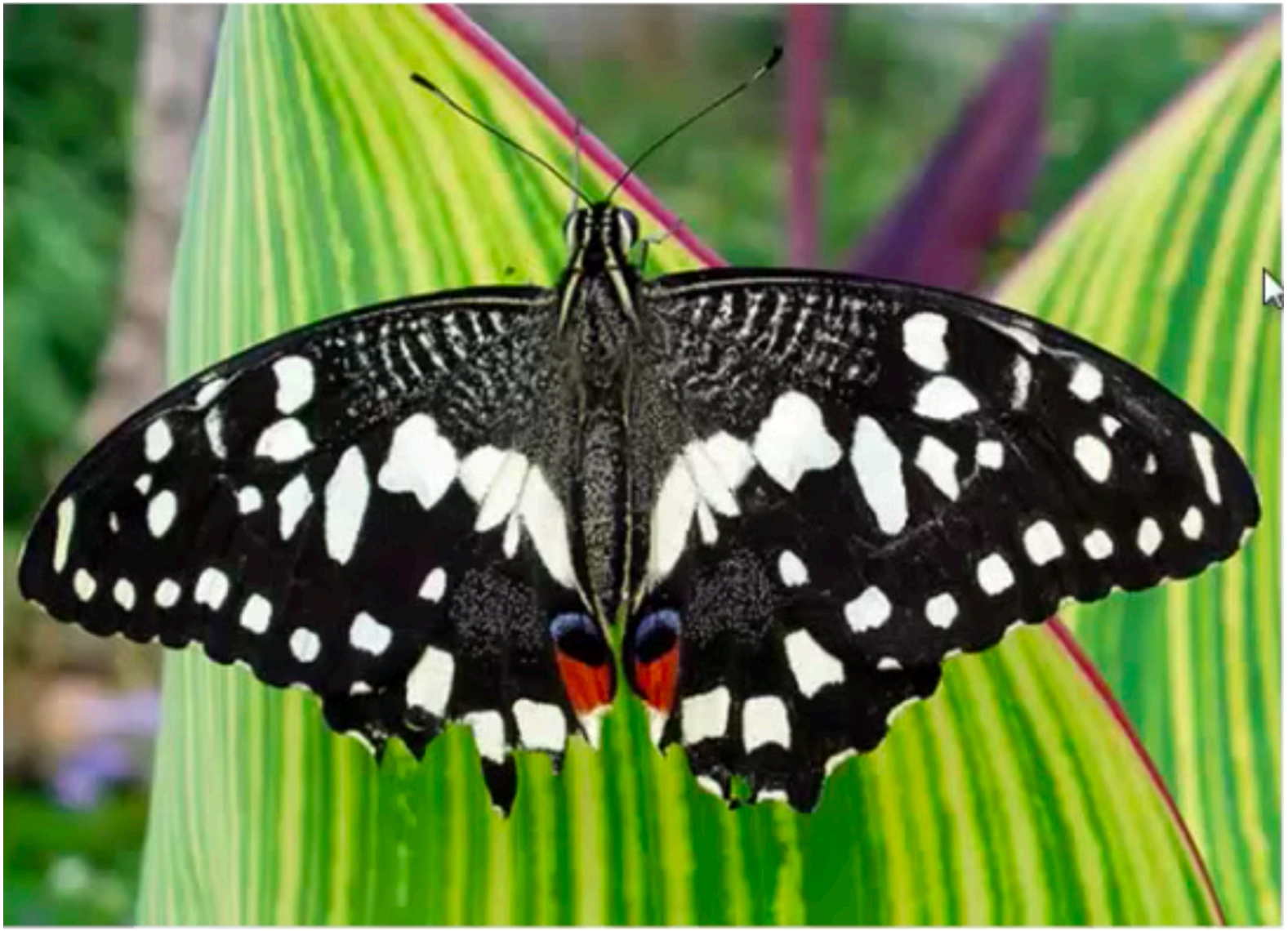


LoG for Peak Detection

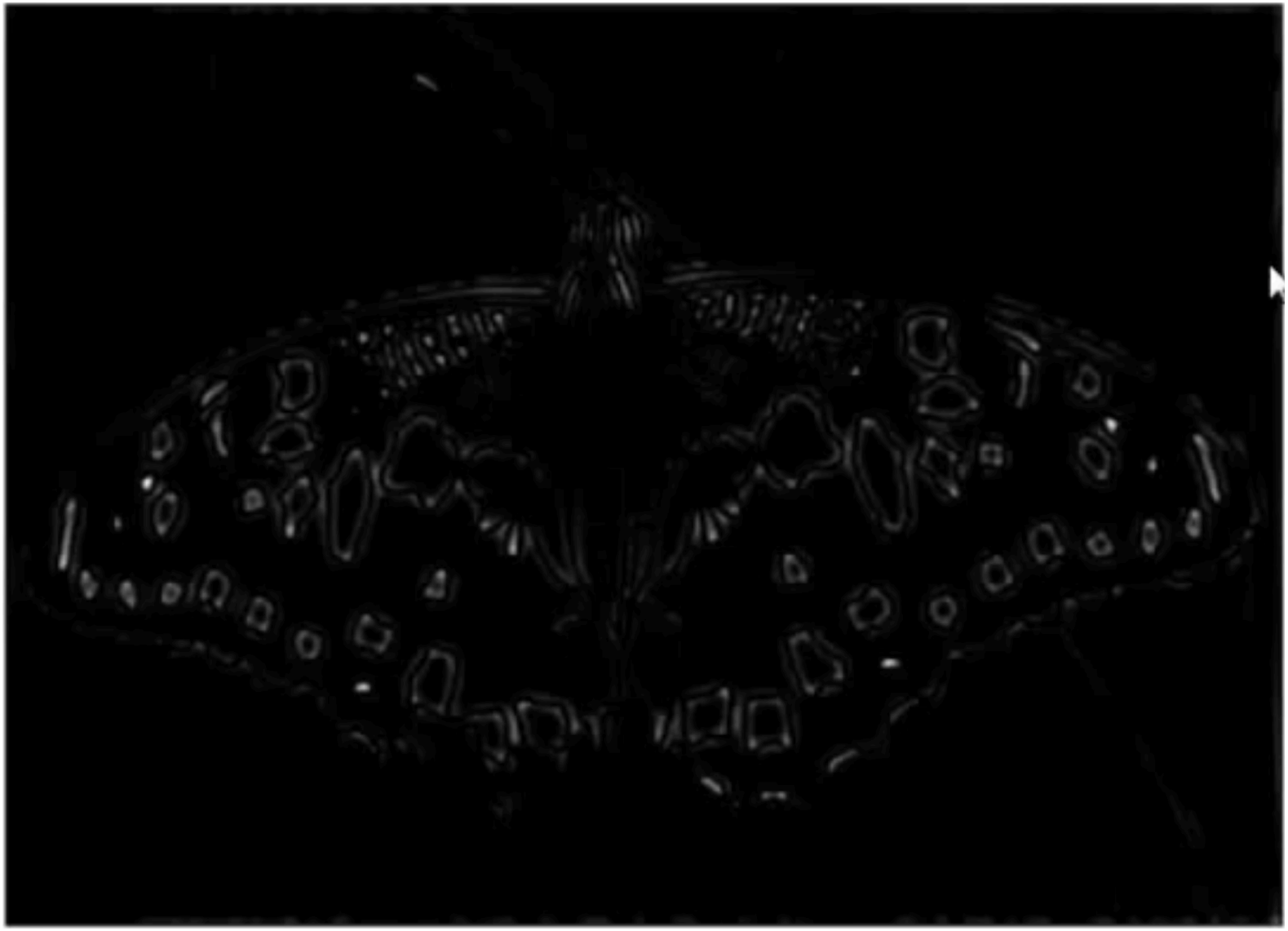
- Apply Gaussian filter
- Take second order derivative
- Second order derivative is also called Laplacian
- Hence, Laplacian of Gaussian

What should be the width of DoG?





Source: Lana Lazebnik



$\sigma = 2$

Source: Lana Lazebnik

$\sigma = 2.5$

Source: Lana Lazebnik

σ= 3.1

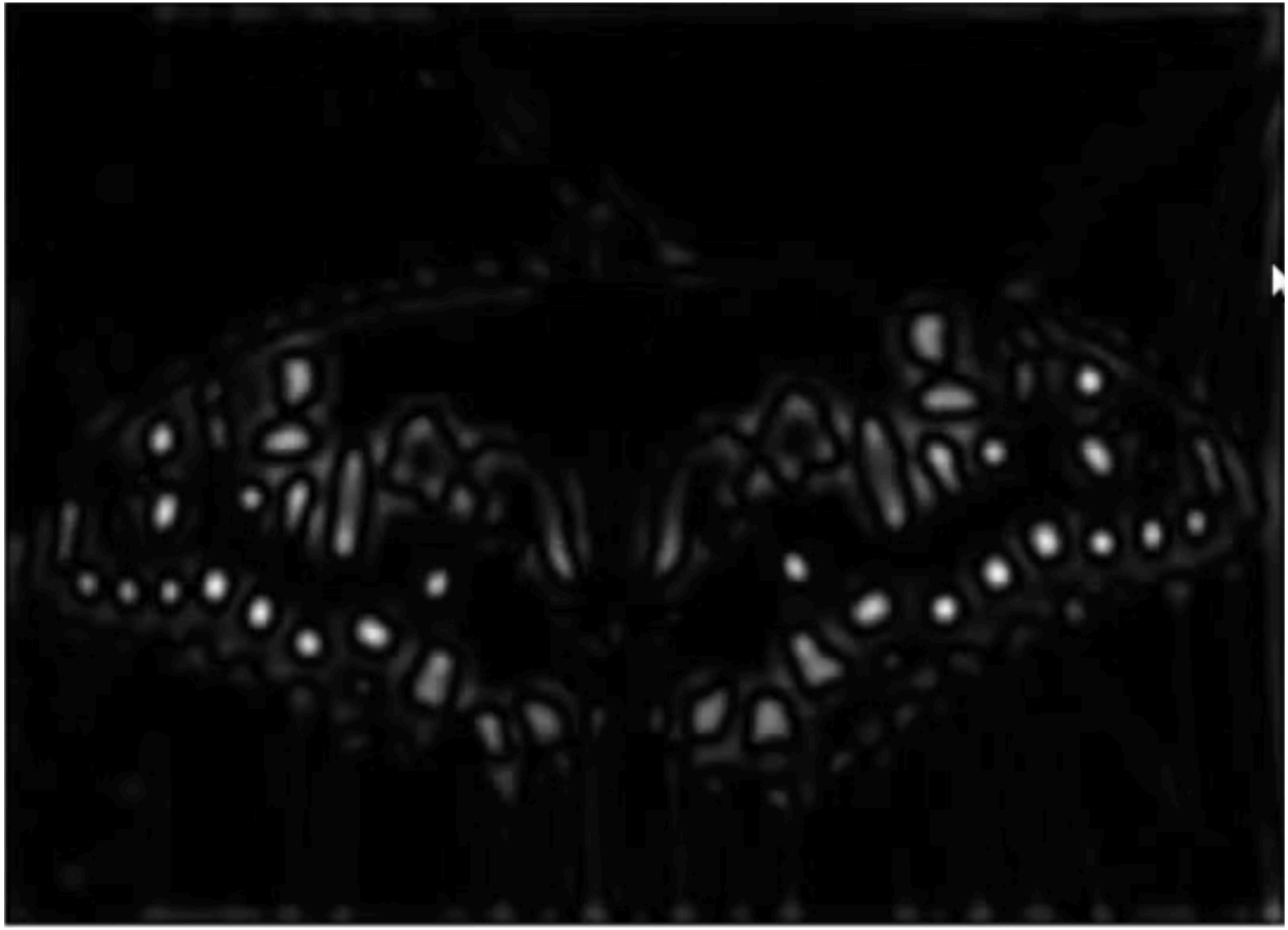
Source: Lana Lazebnik

$\sigma = 3.9$

Source: Lana Lazebnik

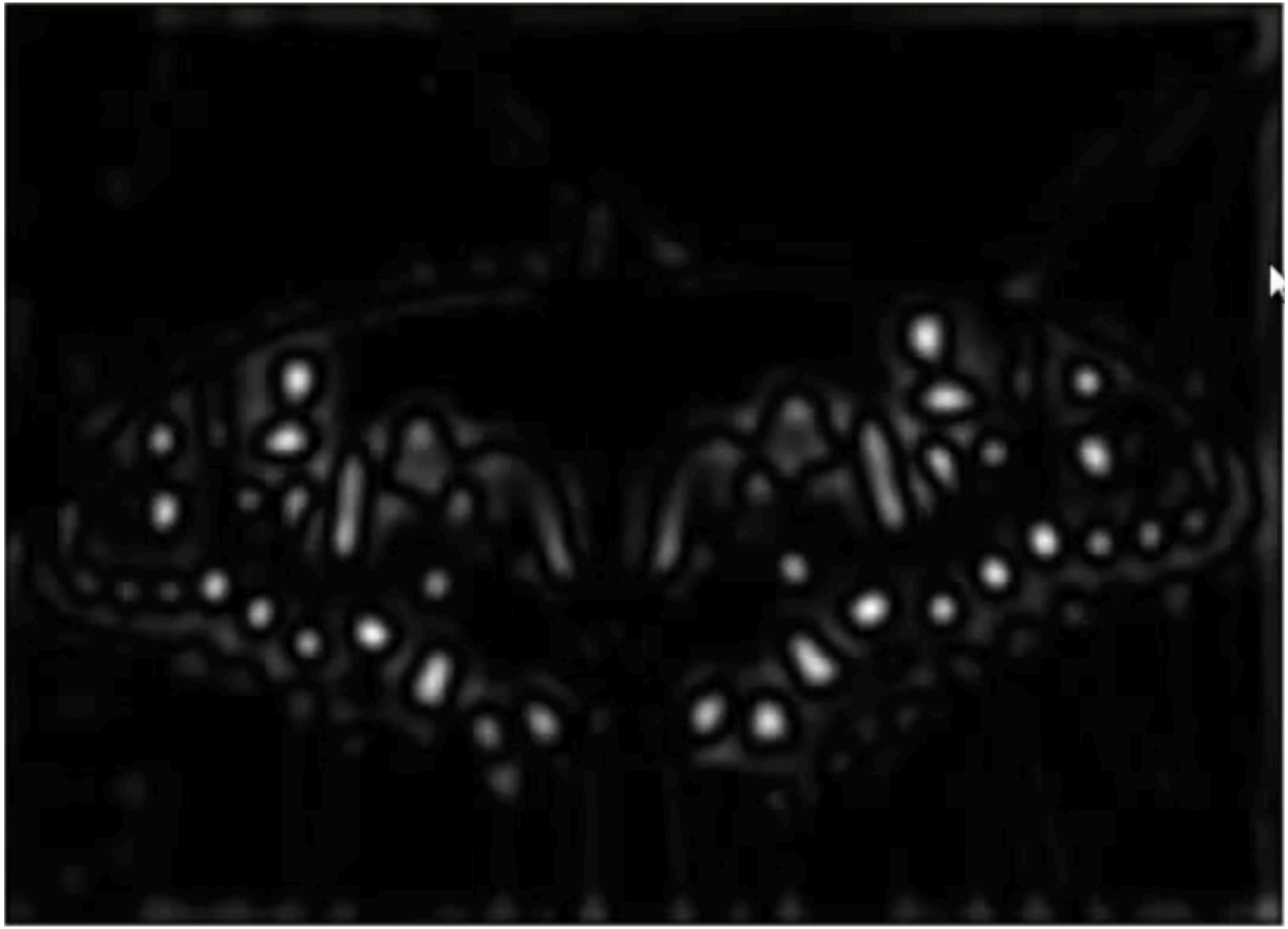
$\sigma = 4.8$

Source: Lana Lazebnik



σ= 6.1

Source: Lana Lazebnik

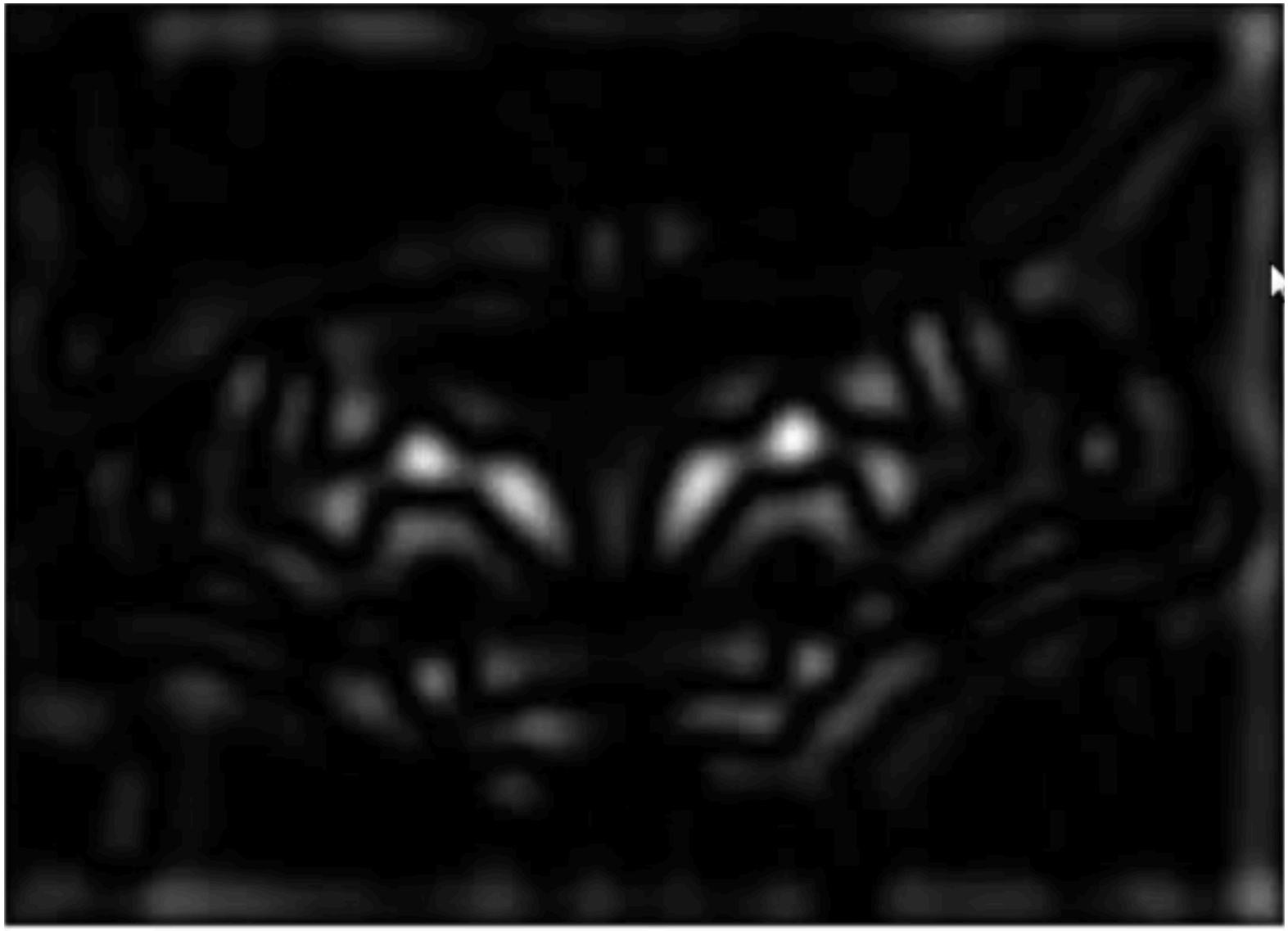


$\sigma = 7.6$

Source: Lana Lazebnik

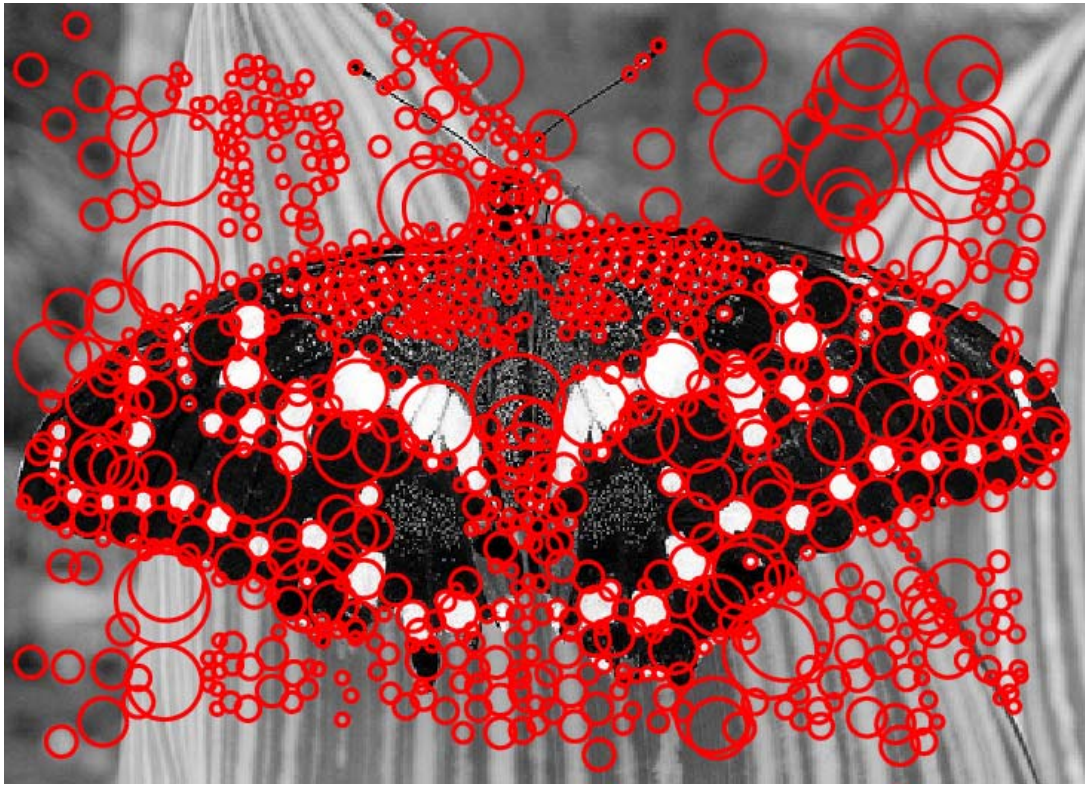
σ= 9.5

Source: Lana Lazebnik



σ= 11.9

Source: Lana Lazebnik

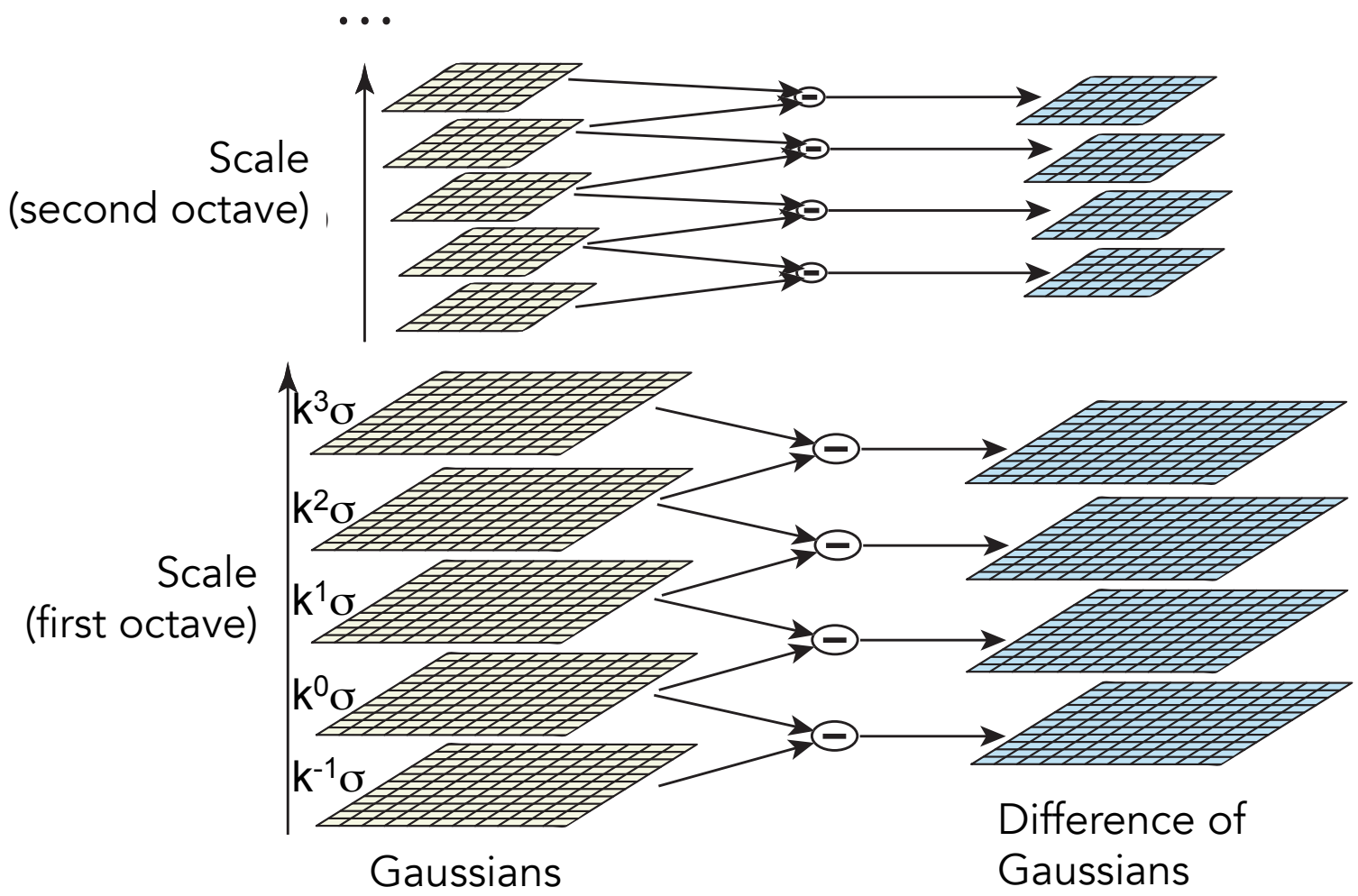


Source: Lana Lazebnik

Stable feature/keypoints

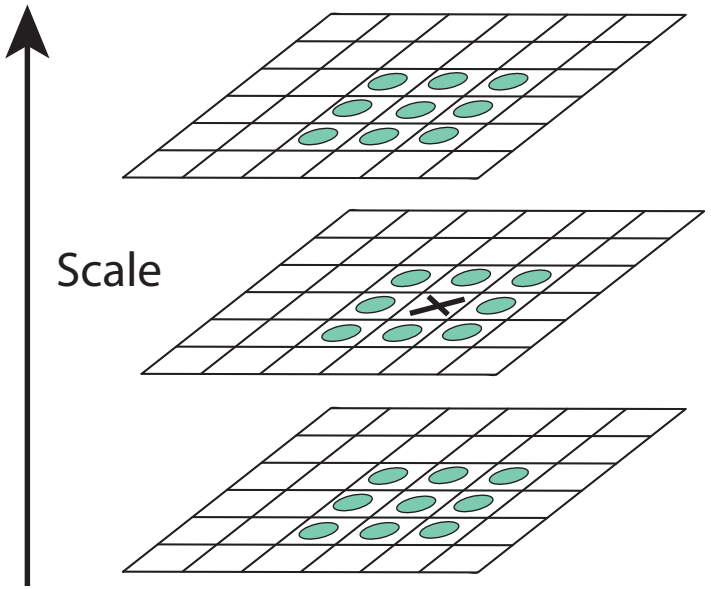
- Features that are stable across scales
- Features that can be assigned across different views of the same object

Scale-Space Extrema Detection



Key point localization

- Detect maxima and minima of difference-of-Gaussian in scale space
- Each point is compared to its 8 neighbors in the current image and 9 neighbors each in the scales above and below



For each max or min found, record the **location** and the **scale**.

Refining Keypoints

- There are still a lot of points, some of them are not stable.
- Keypoint along edges are not stable!
- Eliminating edge points. Edge points have large principal curvature across the edge but a small one in the perpendicular direction.

Orientation Assignment

- Calculate gradient of the image with nearest scale or blur (σ) to the Kaypoint DoG
- Divide histogram into 36 bins and obtain histogram for 16*16 patch around keypoint
- The bin votes are gradient magnitudes weighted by a circular Gaussian window with σ that is 1.5 times of scale of keypoint
- The peak in the histogram in the orientation of the keypoint

Obtain feature vector for each keypoint using gradient orientation!

Possible Features

- Store intensity values in the neighborhood
  - –Sensitive to lighting changes
- Gradient Orientation Histogram

Feature descriptor

- Based on 16*16 patches
- 4*4 sub regions
- 8 bins in each sub region
- 4*4*8=128 dimensions in total

Histogram Calculation

- Find bin corresponding to the angle of the pixel
- Votes are Gaussian-weighted gradient magnitudes
- The Gaussian has σ equal to the half of the descriptor window, i.e. 8

Feature descriptor

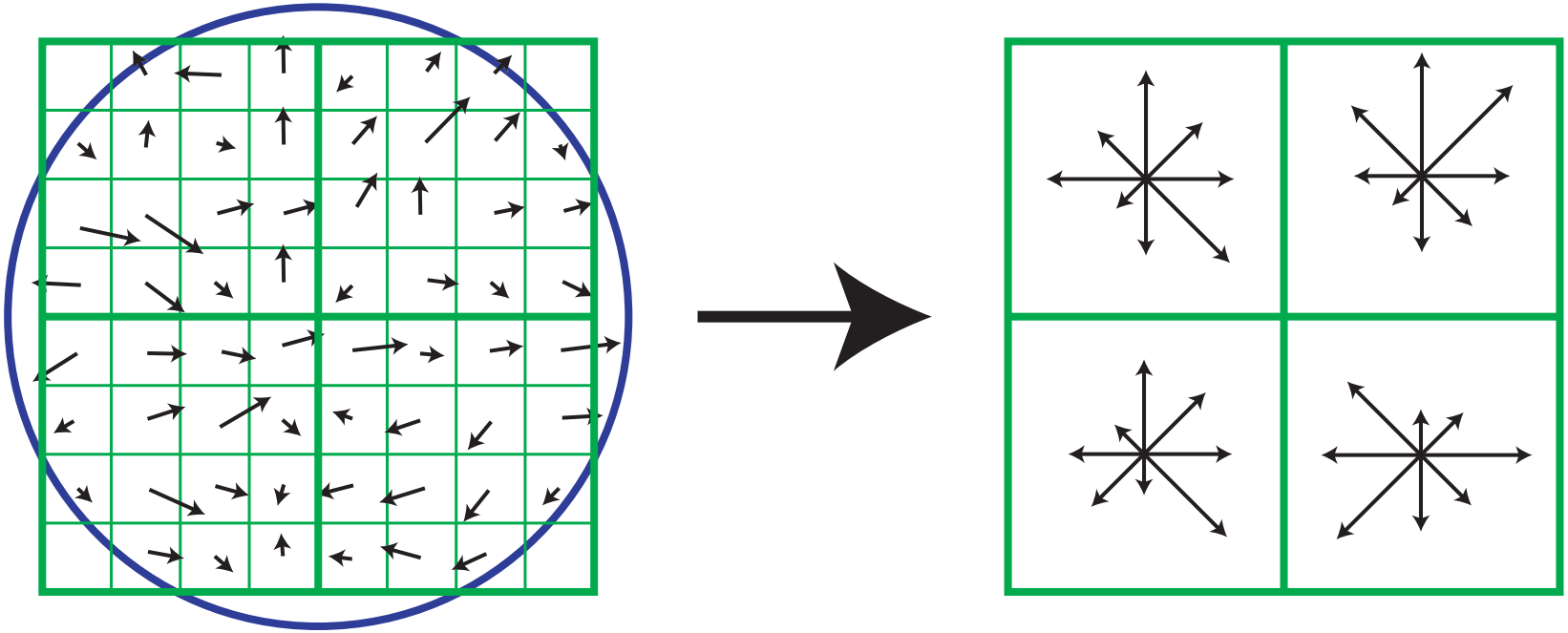
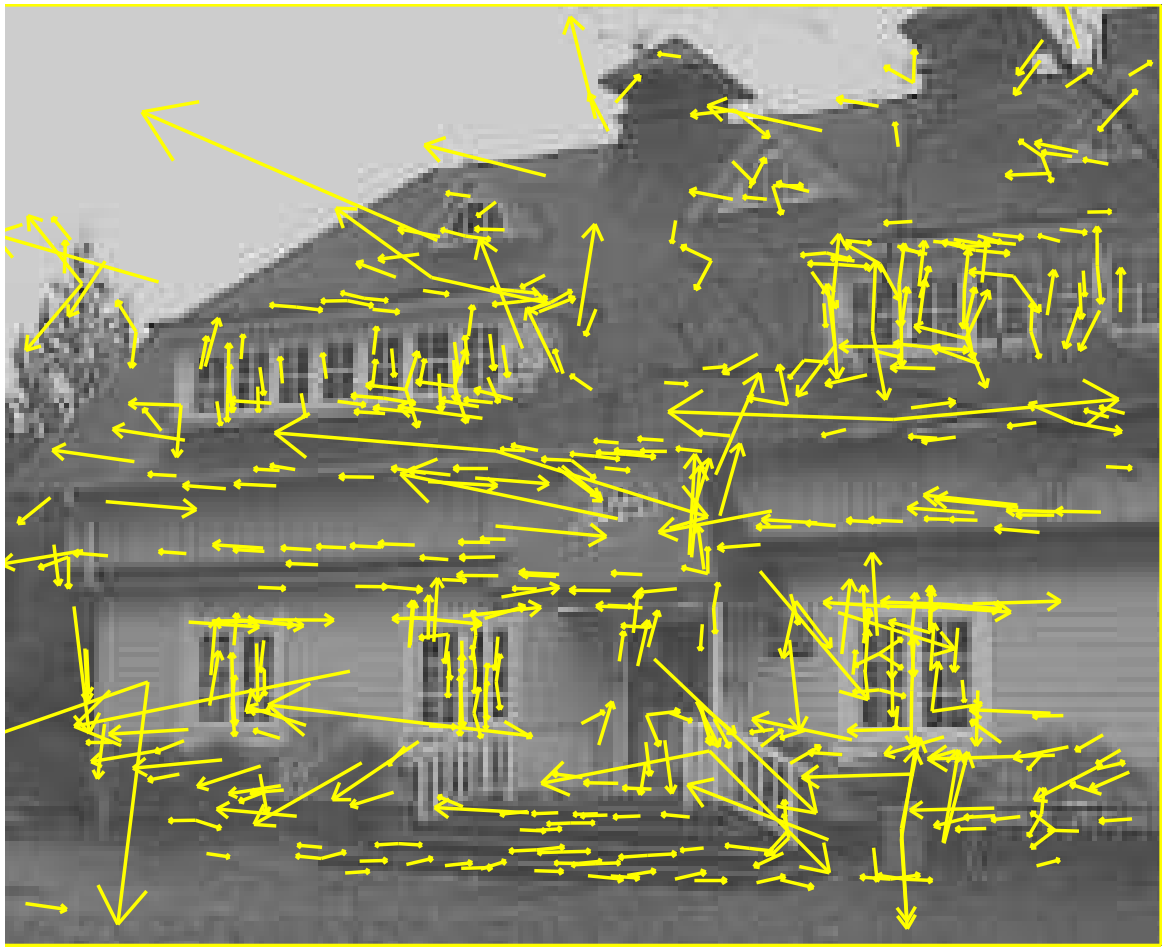


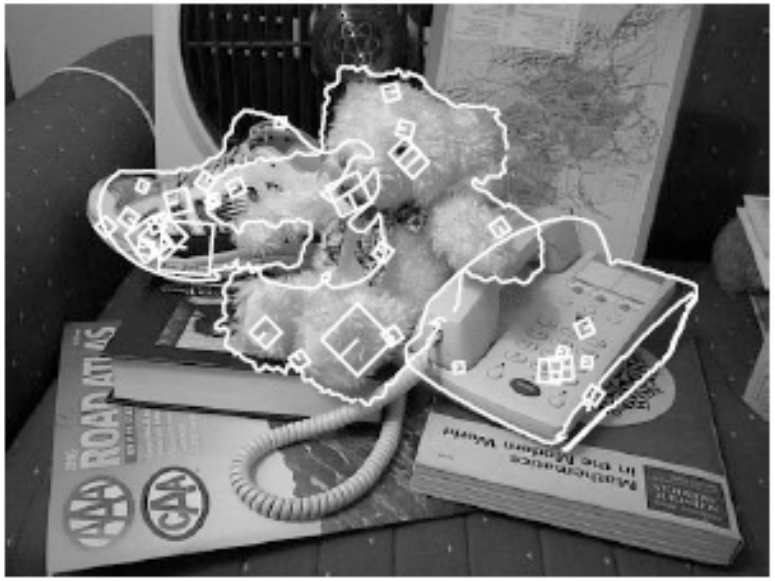
Image gradients

Keypoint descriptor

Example




AAA
CAA


AAA
CAA

Object Recognition Overview

- Store SIFT vectors for each keypoint for each model object in database
- Generate keypoints in test image
- Use nearest neighbor to find feature matches
- Cluster features that agree on object pose
- Estimate final location, scale, and orientation of the object

SURF

1. A faster implementation of SIFT with few minor modifications!
2. Uses 64 features instead of 128 features!

How would you use SIFT for classification?

Steps

- Collect all SIFT keypoint feature vectors from training images
- Obtain a codebook using k-means clustering
- Quantize feature vectors and calculate frequency of each centroid
  - Bag of Words
- From each image you will get a fixed size vector