

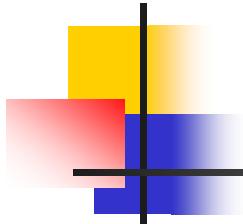
# CAP 5415 Computer Vision

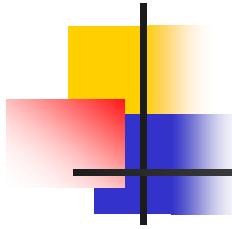
Dr. Mubarak Shah

Univ. of Central Florida

**Lecture-6**

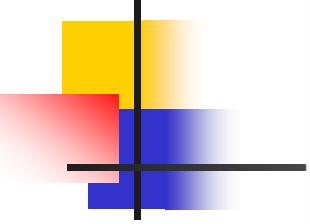
# SIFT: Detector and Descriptor





# SIFT: David Lowe, UBC





# Distinctive Image Features from Scale-Invariant Keypoints

DAVID G. LOWE

Computer Science Department, University of British Columbia, Vancouver, B.C., Canada

Lowe@cs.ubc.ca

Received January 10, 2003; Revised January 7, 2004; Accepted January 22, 2004

**Abstract.** This paper presents a method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene. The features are invariant to image scale and rotation, and are shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. The features are highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images. This paper also describes an approach to using these features for object recognition. The recognition proceeds by matching individual features to a database of features from known objects using a fast nearest-neighbor algorithm, followed by a Hough transform to identify clusters belonging to a single object, and finally performing verification through least-squares solution for consistent pose parameters. This approach to recognition can robustly identify objects among clutter and occlusion while achieving near real-time performance.

**Keywords:** invariant features, object recognition, scale invariance, image matching

## 1. Introduction

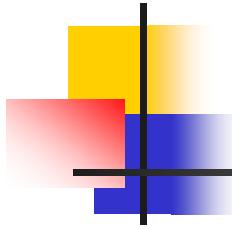
Image matching is a fundamental aspect of many problems in computer vision, including object or scene recognition, solving for 3D structure from multiple images, stereo correspondence, and motion tracking. This paper describes image features that have many properties that make them suitable for matching differing images of an object or scene. The features are invariant to image scaling and rotation, and partially invariant to change in illumination and 3D camera viewpoint. They are well localized in both the spatial and frequency domains, reducing the probability of disruption by occlusion, clutter, or noise. Large numbers of features can be extracted from typical images with efficient algorithms. In addition, the features are highly distinctive, which allows a single feature to be correctly matched with high probability against a large database of features, providing a basis for object and scene recognition.

The cost of extracting these features is minimized by taking a cascade filtering approach, in which the more

expensive operations are applied only at locations that pass an initial test. Following are the major stages of computation used to generate the set of image features:

1. *Scale-space extrema detection:* The first stage of computation searches over all scales and image locations. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation.
2. *Keypoint localization:* At each candidate location, a detailed model is fit to determine location and scale. Keypoints are selected based on measures of their stability.
3. *Orientation assignment:* One or more orientations are assigned to each keypoint location based on local image gradient directions. All future operations are performed on image data that has been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations.

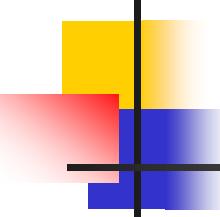
Cited by [25688](#)



# SIFT - Key Point Extraction

- Stands for **S**cale **I**nvariant **F**eature **T**ransform
- Patented by university of British Columbia
- Similar to the one used in primate visual system (human, ape, monkey, etc.)
- Transforms image data into scale-invariant coordinates

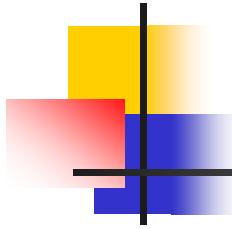
D. Lowe. *Distinctive image features from scale-invariant key points.*, International Journal of Computer Vision 2004.



# Goal

---

- Extract distinctive invariant features
  - Correctly matched against a large database of features from many images
- Invariance to image scale and rotation
- Robustness to
  - Affine (rotation, scale, shear) distortion,
  - Change in 3D viewpoint,
  - Addition of noise,
  - Change in illumination.

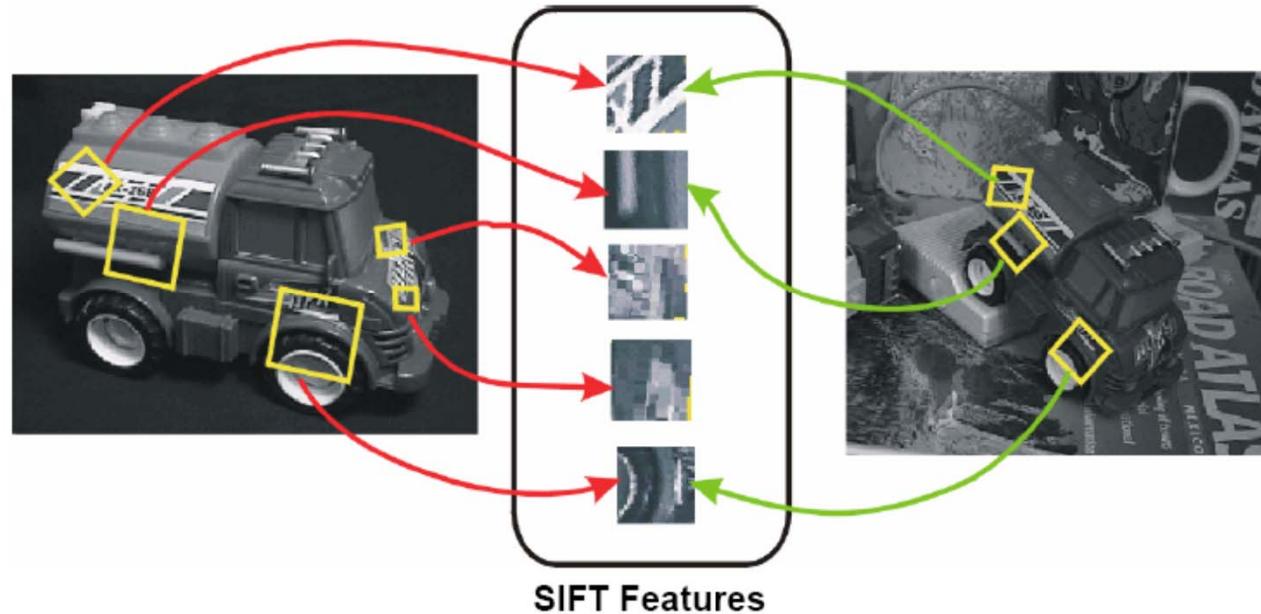


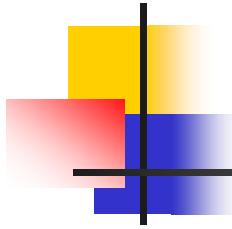
# Advantages

---

- **Locality:** features are local, so robust to occlusion and clutter
- **Distinctiveness:** individual features can be matched to a large database of objects
- **Quantity:** many features can be generated for even small objects
- **Efficiency:** close to real-time performance

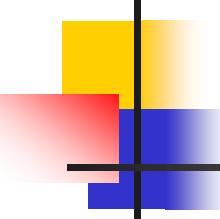
# Invariant Local Features





# Steps for Extracting Key Points (SIFT Points)

- Scale space peak selection
  - Potential locations for finding features
- Key point localization
  - Accurately locating the feature key points
- Orientation Assignment
  - Assigning orientation to the key points
- Key point descriptor
  - Describing the key point as a high dimensional vector (128) (SIFT Descriptor)

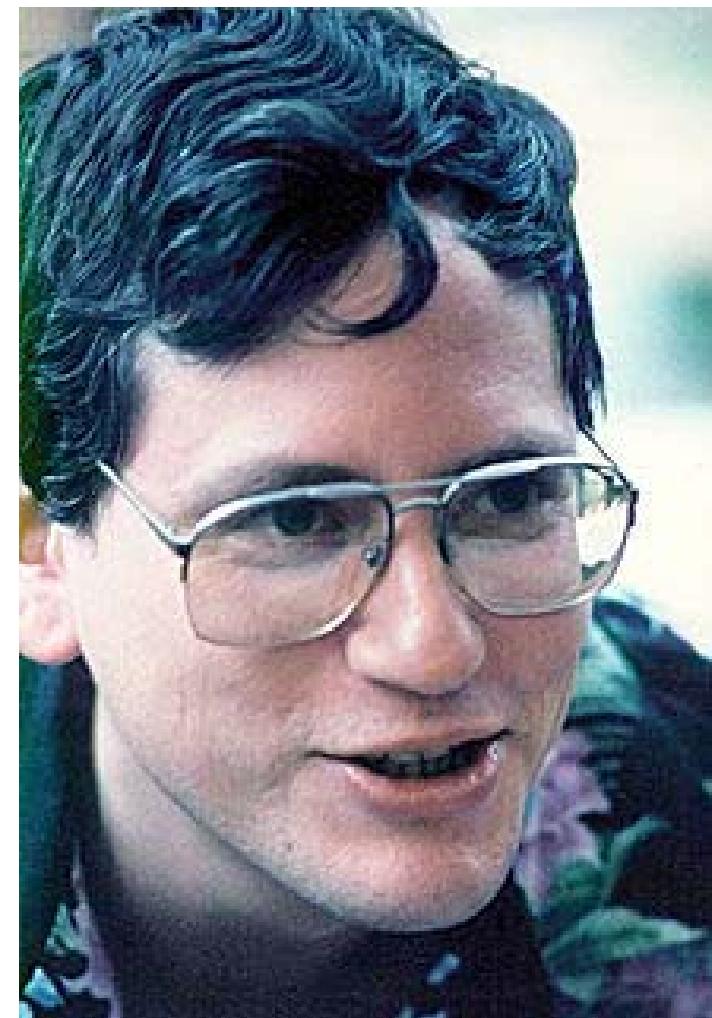


# Scales

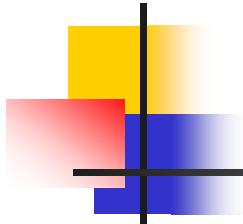
- What should be sigma value for Canny and LoG edge detection?
- If use multiple sigma values (scales), how do you combine multiple edge maps?
- Marr-Hildreth:
  - *Spatial Coincidence* assumption:
    - Zerocrossings that coincide over several scales are physically significant.

# **Andrew Witkin**

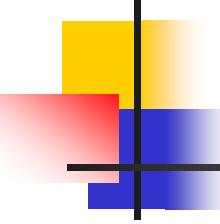
**(July 22, 1952 – September 12, 2010)**



# Andrew Witkin

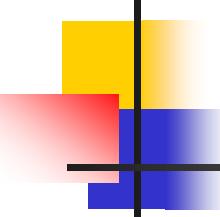


- He did his BS at Columbia University in Psychology
- Ph.D. at MIT
- Worked at
  - SRI International
  - Schlumberger's Fairchild Laboratory for Artificial Intelligence Research
  - CMU CS Professor (1988-1998)
  - Pixar (1998-2010)
- Died in a scuba diving accident off the coast of Monterey, California on September 12th 2010.



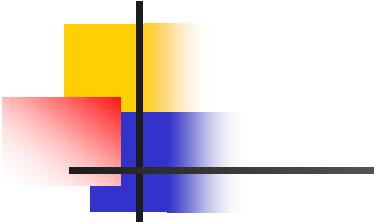
# Andrew Witkin

- The paper "Snakes: Active Contour Models" (by Kass, Witkin, and Terzopoulos) achieved an honorable mention for the Marr Prize in 1987.
  - According to CiteSeer, this paper is the 11th most cited paper ever in computer science.
- The 1987 paper "Constraints on deformable models: Recovering 3D shape and nonrigid motion" (by Terzopoulos, Witkin, and Kass) was also a prize winner.
- In 1992, Witkin and Kass were awarded the Prix Ars Electronica computer graphics award for "Reaction–Diffusion Texture Buttons."



# Andrew Witkin

- Witkin received the ACM SIGGRAPH Computer Graphics Achievement Award in 2001 "for his pioneering work in bringing a physics based approach to computer graphics."
- As senior scientist at Pixar Animation Studios, Witkin received a technical academy award in 2006 for "pioneering work in physically based computer-generated techniques used to simulate realistic cloth in motion pictures."
- In the PIXAR movie Cars 2 in the credits, exit a message: In loving memory Japhet Pieper 1965-2010 Andy Witkin 1952-2010. This message can be see at minutes:1:43:09.



## SCALE-SPACE FILTERING:

A New Approach To Multi-Scale Description

Andrew P. Witkin

Fairchild Laboratory for Artificial Intelligence Research

### ABSTRACT

The extrema in a signal and its first few derivatives provide a useful general purpose qualitative description for many kinds of signals. A fundamental problem in computing such descriptions is scale: a derivative must be taken over some neighborhood, but there is seldom a principled basis for choosing its size. Scale-space filtering is a method that describes signals qualitatively, managing the ambiguity of scale in an organized and natural way. The signal is first expanded by convolution with gaussian masks over a continuum of sizes. This "scale-space" image is then collapsed, using its qualitative structure, into a tree providing a concise but complete qualitative description covering all scales of observation. The description is further refined by applying a stability criterion, to identify events that persist of large changes in scale.

### 1. Introduction

Hardly any sophisticated signal understanding task can be performed using the raw numerical signal values directly; some description of the signal must first be obtained. An initial description ought to be as compact as possible, and its elements should correspond as closely as possible to meaningful objects or events in the signal-forming process. Frequently, local extrema in the signal and its derivatives—and intervals bounded by extrema—are particularly appropriate descriptive primitives: although local and closely tied to the signal data, these events often have direct semantic interpretations, e.g. as edges in images. A description that characterizes a signal by its extrema and those of its first few derivatives is a *qualitative* description of exactly the kind we were taught to use in elementary calculus to "sketch" a function.

A great deal of effort has been expended to obtain this kind of primitive qualitative description (for overviews of this literature, see [11],[2],[10],) and the problem has proved extremely difficult. The problem of *scale* has emerged consistently as a fundamental source of difficulty, because the

events we perceive and find meaningful vary enormously in size and extent. The problem is not so much to eliminate fine-scale noise, as to separate events at different scales arising from distinct physical processes.[7] It is possible to introduce a *parameter of scale* by smoothing the signal with a mask of variable size, but with the introduction of scale-dependence comes ambiguity: every setting of the scale parameter yields a different description; new extremal points may appear, and existing ones may move or disappear. How can we decide which if any of this continuum of descriptions is "right"?

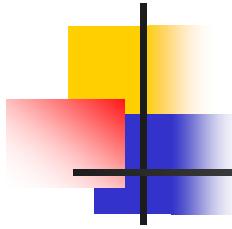
There is rarely a sound basis for setting the scale parameter. In fact, it has become apparent that for many tasks no one scale of description is categorically correct: the physical processes that generate signals such as images act at a variety of scales, none intrinsically more interesting or important than another. Thus the ambiguity introduced by scale is inherent and inescapable, so the goal of scale-dependent description cannot be to eliminate this ambiguity, but rather to manage it effectively, and reduce it where possible.

This line of thinking has led to considerable interest in multi-scale descriptions [12],[2],[9], [8]. However, merely computing descriptions at multiple scales does not solve the problem; if anything, it exacerbates it by increasing the volume of data. Some means must be found to organize or simplify the description, by relating one scale to another. Some work has been done in this area aimed at obtaining "edge pyramids" (e.g. [6]), but no clear-cut criteria for constructing them have been put forward. Marr [7] suggested that zero-crossings that coincide over several scales are "physically significant," but this idea was neither justified nor tested.

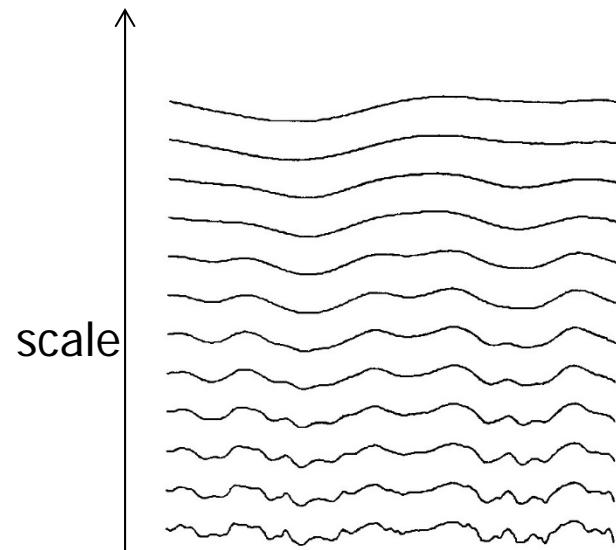
How, then, can descriptions at different scales be related to each other in an organized, natural, and compact way? Our solution, which we call *scale-space filtering*, begins by continuously varying the scale parameter, sweeping out a surface that we call the *scale-space image*. In this representation, it is possible to track extrema as they move continuously with scale changes, and to identify the singu-

# Scale Space (Witkin, IJCAI 1983)

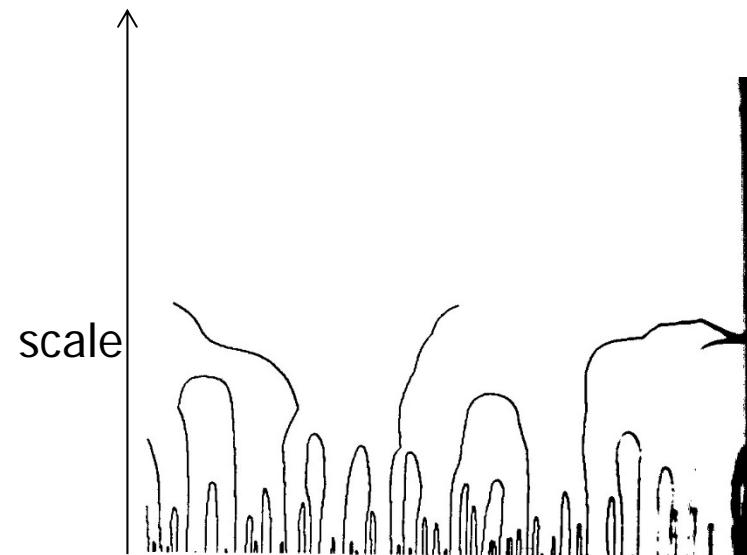
- Apply whole spectrum of scales (sigma in Gaussian)
- Plot zero-crossings vs scales in a scale-space



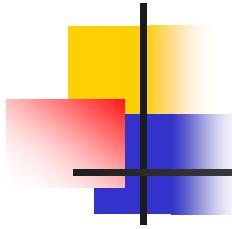
# Scale Space



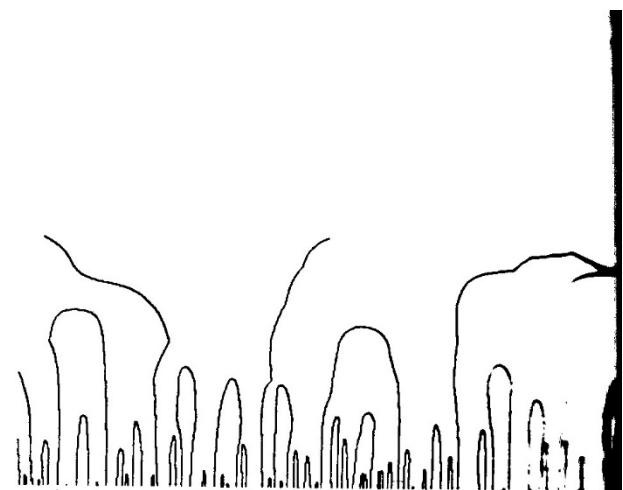
Multiple smooth versions of a signal



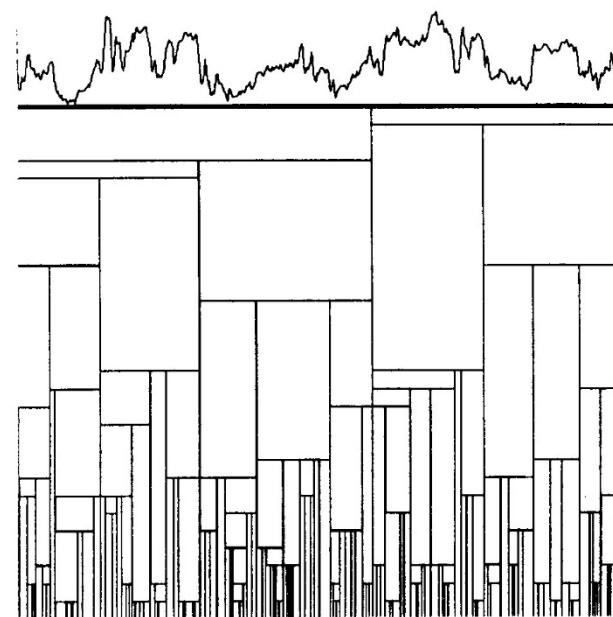
Zerocrossings at multiple scale



# Scale Space



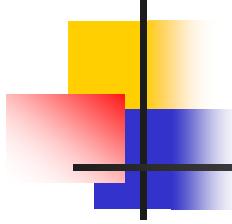
Scale Space



Interval Tree

# Scale Space (Witkin, IJCAI 1983)

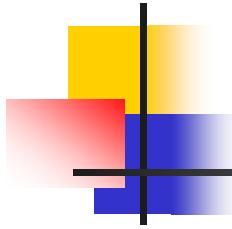
- Apply whole spectrum of scales
- Plot zerocrossings vs scales in a scale-space
- Interpret scale space contours
  - Contours are arches, open at the bottom, closed at the top
  - Interval tree
    - Each interval corresponds to a node in a tree,
      - whose parent node represents larger interval, from which interval emerged, and
      - whose off springs represent smaller intervals.
    - Stability of a node is a scale range over which the interval exists.



# Scale Space

---

- Top level description
  - Iteratively remove nodes from the tree,
    - splicing out nodes that are less stable than any of their parents and off springs



# Scale Space

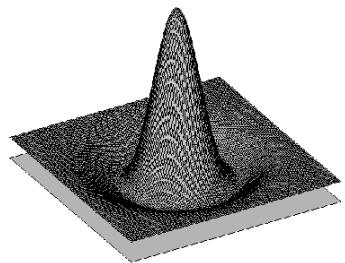
A top level description of several signals using stability criterion.



# Laplacian-of-Gaussian (LoG)

- Interest points:

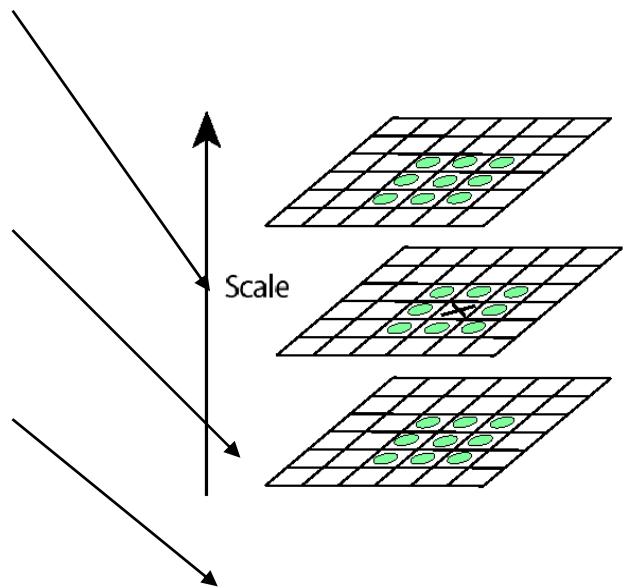
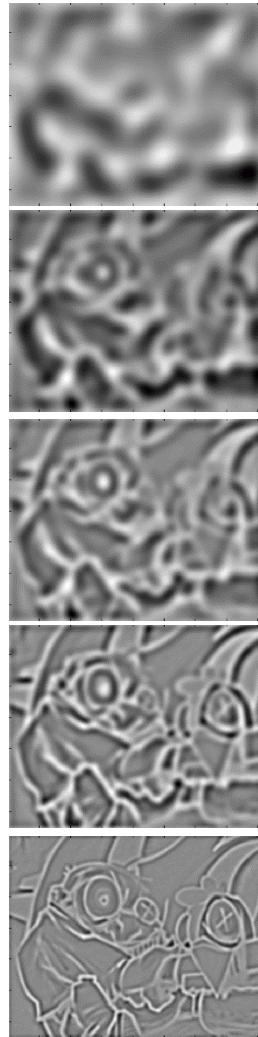
Local maxima in scale space of Laplacian-of-Gaussian



$$L_{xx}(\sigma) + L_{yy}(\sigma) \rightarrow \sigma^3$$

$\sigma^5$   
 $\sigma^4$   
 $\sigma^3$   
 $\sigma^2$   
 $\sigma$

A series of five arrows pointing downwards from left to right, each labeled with a different power of sigma ( $\sigma^5, \sigma^4, \sigma^3, \sigma^2, \sigma$ ). These arrows indicate the progression of the LoG operator across increasing scales.



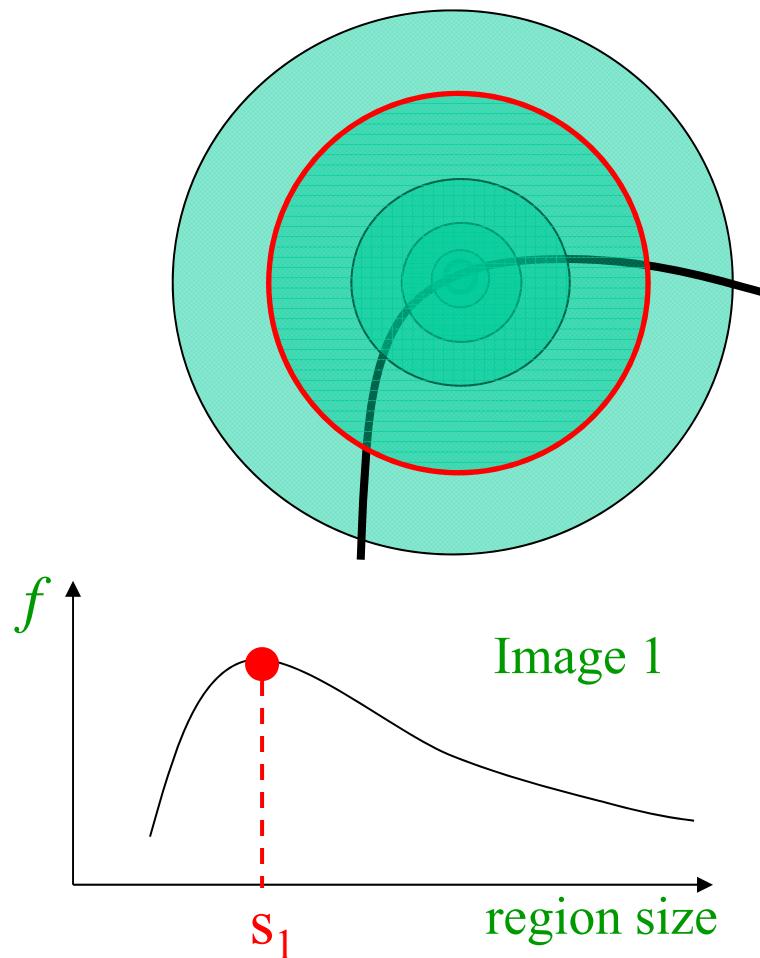
$\Rightarrow$  List of  
 $(x, y, \sigma)$

# Automatic scale selection

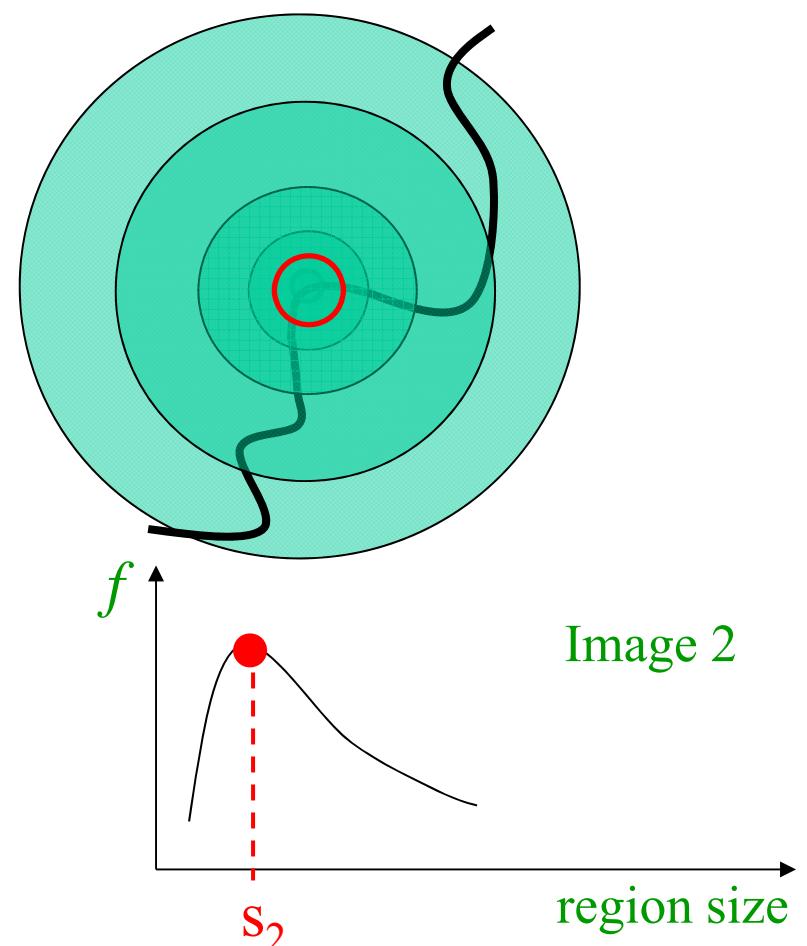
---

## Intuition:

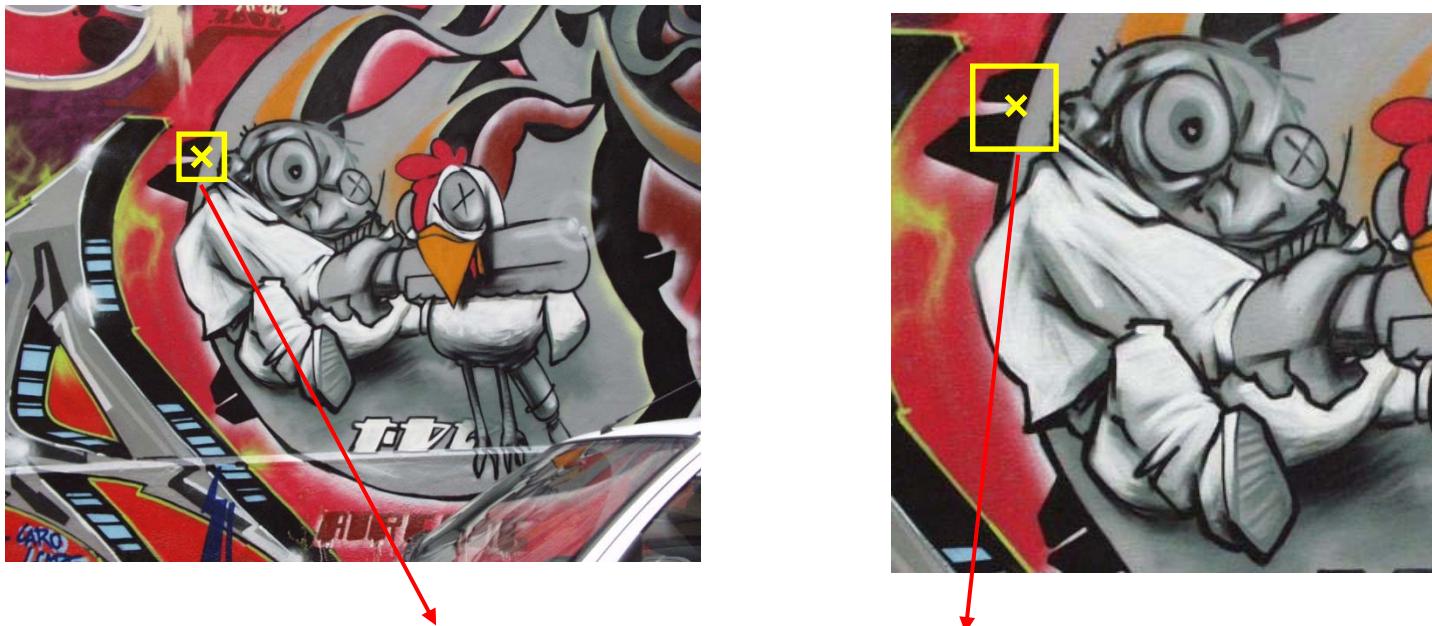
- Find scale that gives local maxima of some function  $f$  in both position and scale.



K. Grauman,



# Automatic Scale Selection

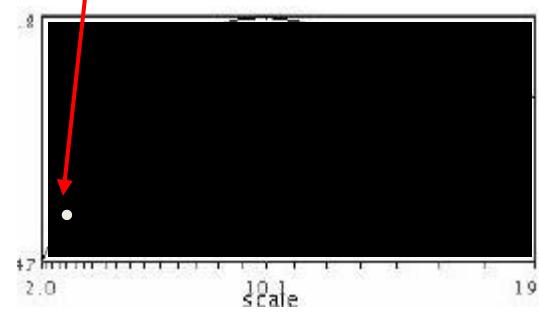
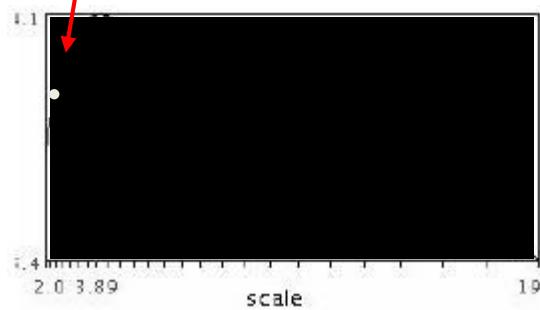


$$f(I_{i_1 \dots i_m}(x, \sigma)) = f(I_{i_1 \dots i_m}(x', \sigma'))$$

How to find corresponding patch sizes?

# Automatic Scale Selection

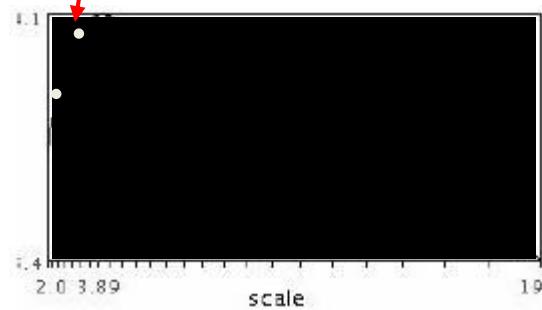
- Function responses for increasing scale (scale signature)



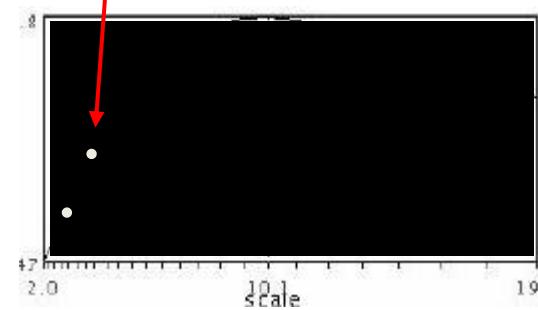
K. Grauman, B. Leibe

# Automatic Scale Selection

- Function responses for increasing scale (scale signature)



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

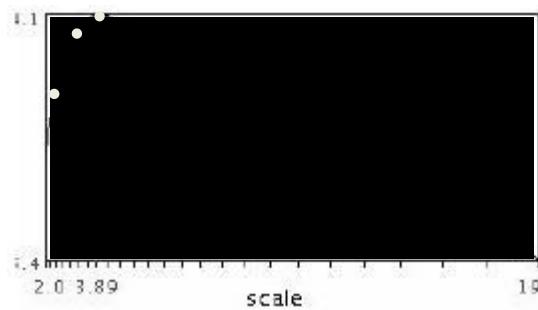
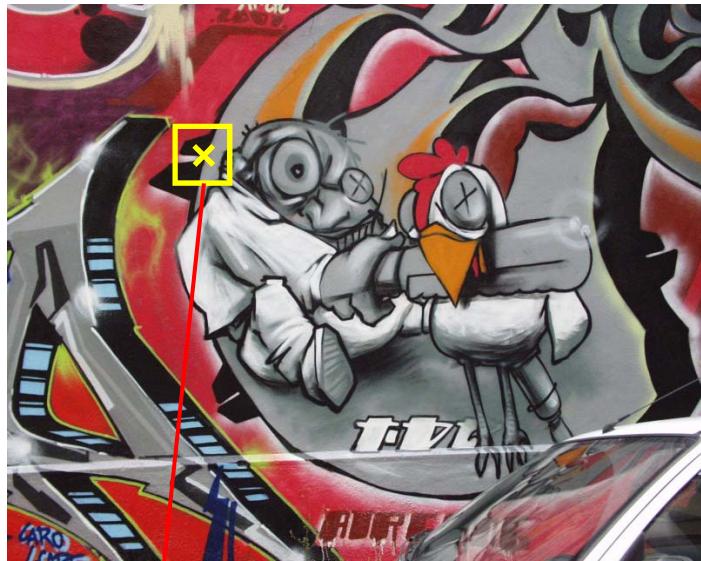


$$f(I_{i_1 \dots i_m}(x', \sigma))$$

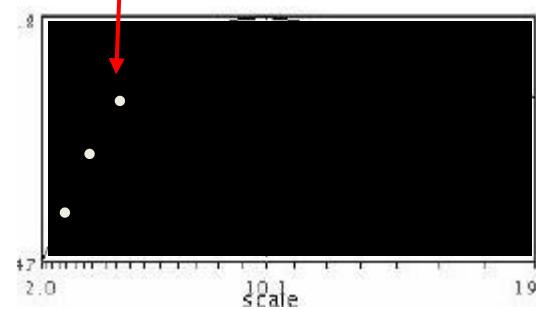
K. Grauman, B. Leibe

# Automatic Scale Selection

- Function responses for increasing scale (scale signature)



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

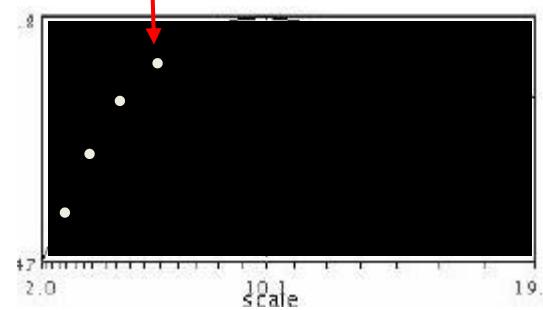
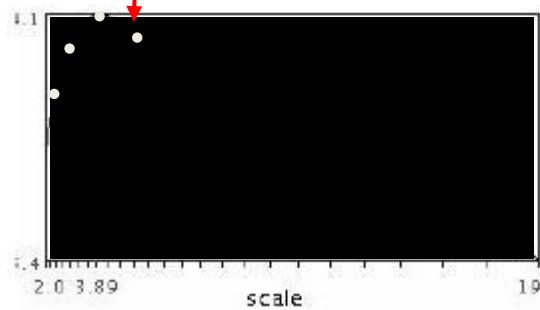
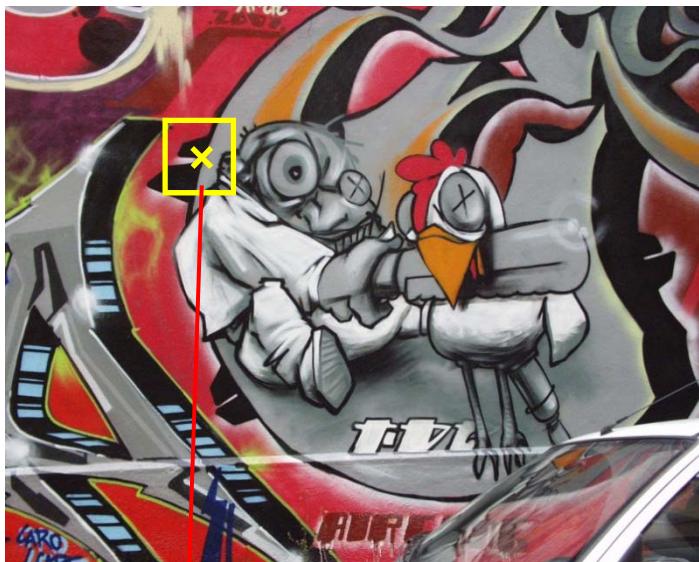


$$f(I_{i_1 \dots i_m}(x', \sigma))$$

K. Grauman, B. Leibe

# Automatic Scale Selection

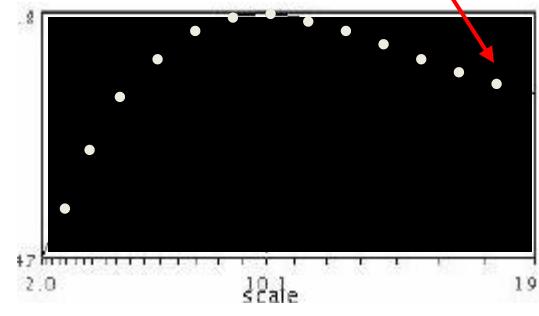
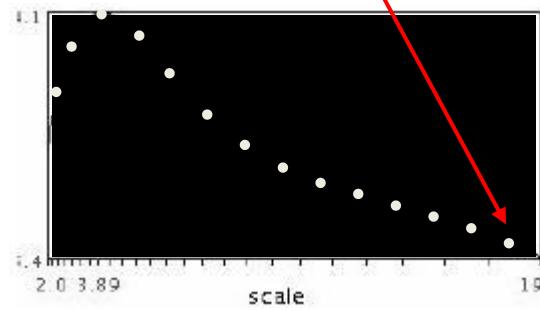
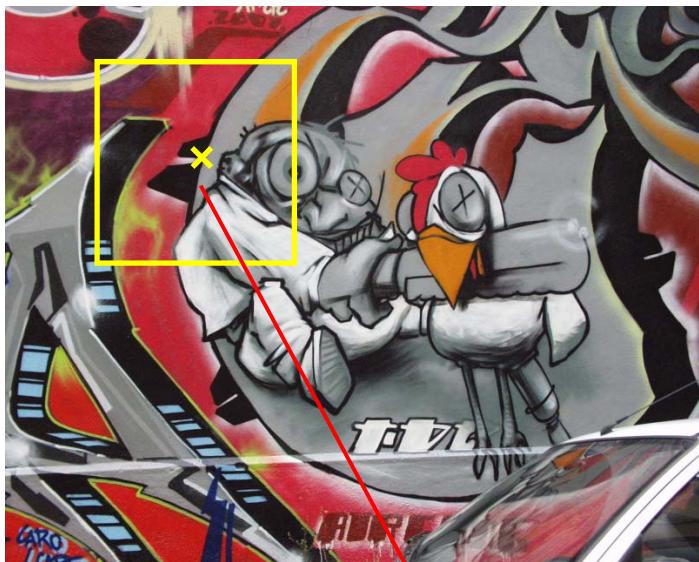
- Function responses for increasing scale (scale signature)



K. Grauman, B. Leibe

# Automatic Scale Selection

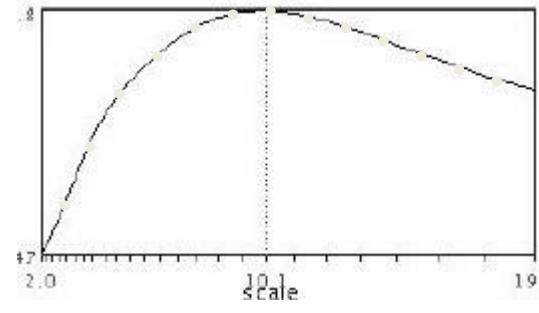
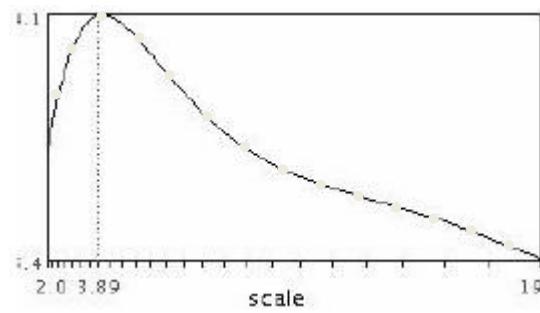
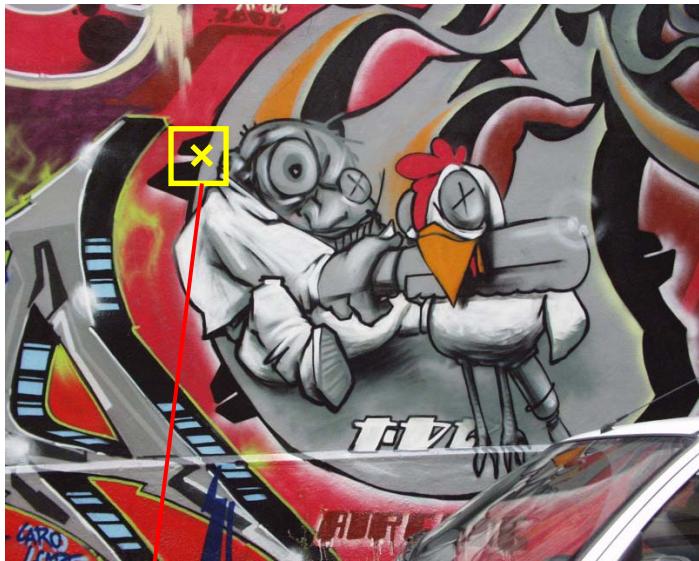
- Function responses for increasing scale (scale signature)



K. Grauman, B. Leibe

# Automatic Scale Selection

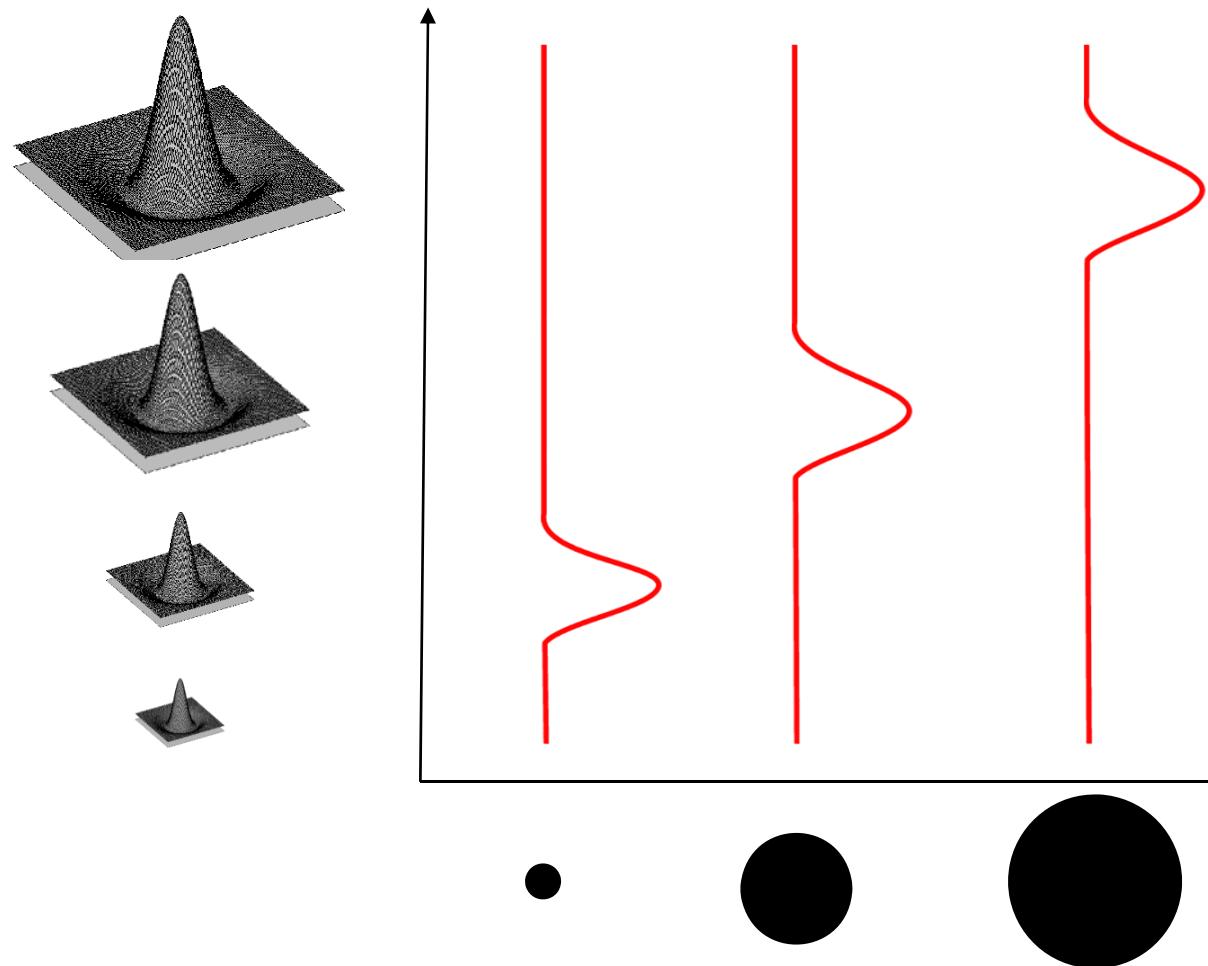
- Function responses for increasing scale (scale signature)



K. Grauman, B. Leibe

# What Is A Useful Signature Function?

- Laplacian-of-Gaussian = “blob” detector



# Scale-space blob detector: Example

---



Source: Lana Lazebnik

# Scale-space blob detector: Example

---

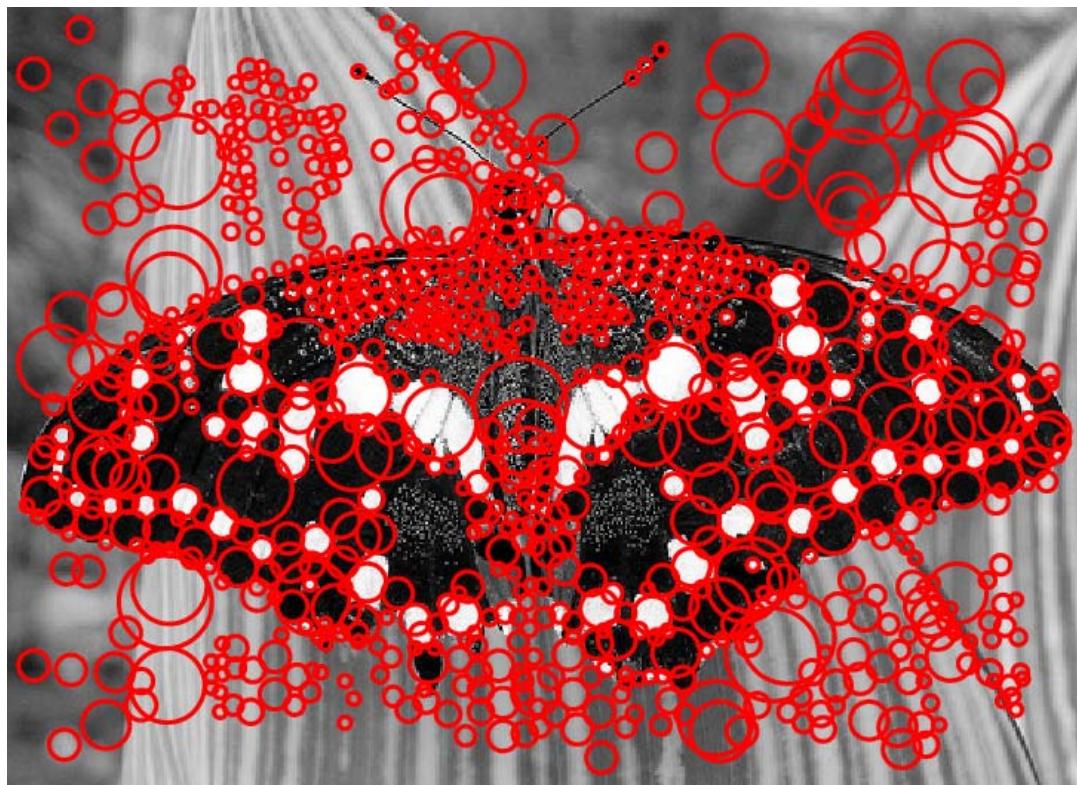


sigma = 11.9912

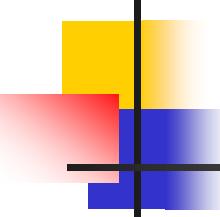
Source: Lana Lazebnik

# Scale-space blob detector: Example

---

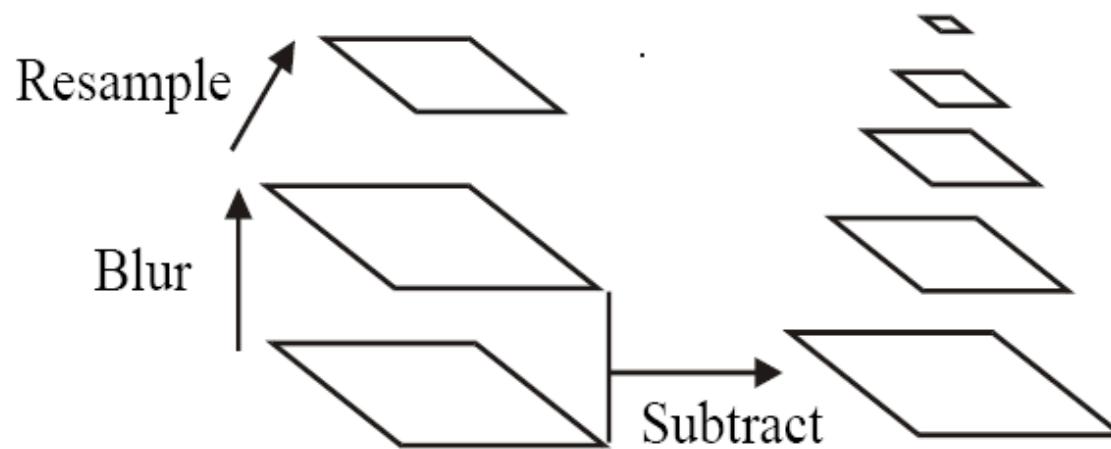


Source: Lana Lazebnik

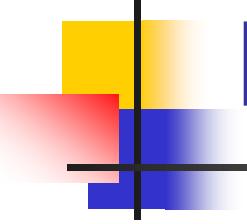


# Building a Scale Space

- All scales must be examined to identify scale-invariant features
- An efficient function is to compute the Laplacian Pyramid (Difference of Gaussian) (Burt & Adelson, 1983)



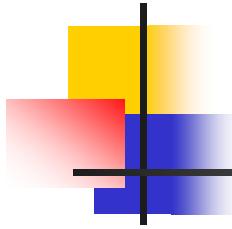
# Approximation of LoG by Difference of Gaussians


$$\frac{\partial G}{\partial \sigma} = \sigma \Delta^2 G \quad \text{Heat Equation}$$

$$\sigma \Delta^2 G = \frac{\partial G}{\partial \sigma} = \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma}$$

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \Delta^2 G$$

Typical values:  $\sigma = 1.6$ ;  $k = \sqrt{2}$



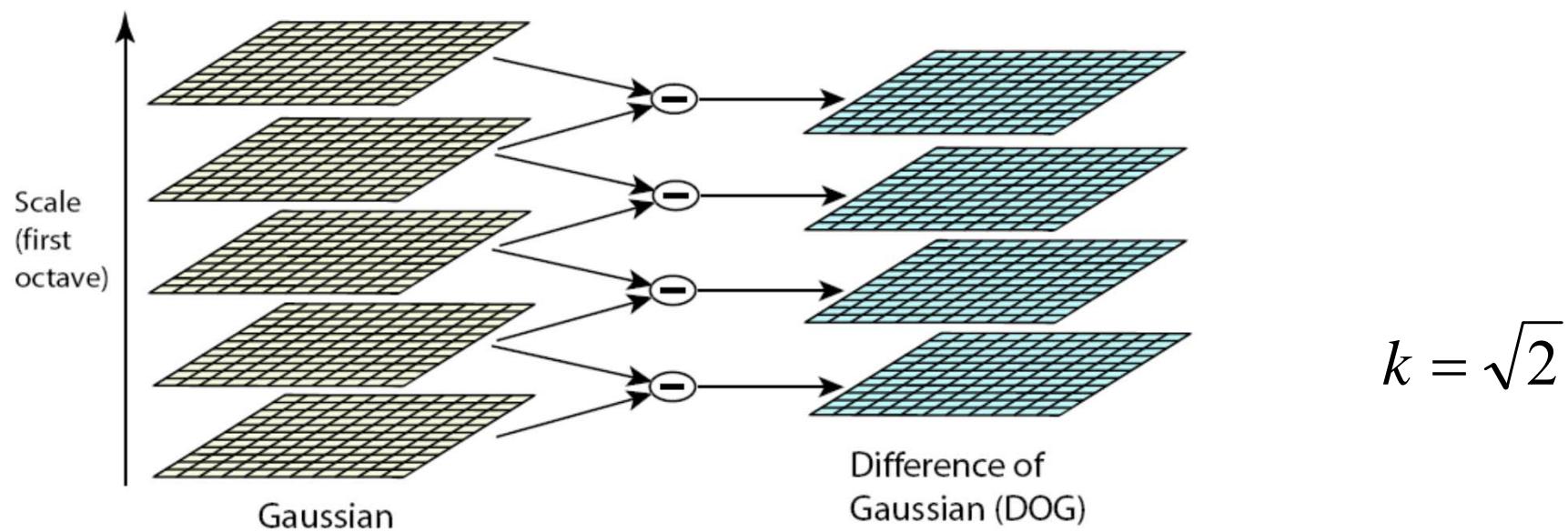
# Steps for Extracting Key Points (SIFT Points)

- Scale space peak selection
  - Potential locations for finding features
- Key point localization
  - Accurately locating the feature key points
- Orientation Assignment
  - Assigning orientation to the key points
- Key point descriptor
  - Describing the key point as a high dimensional vector (128) (SIFT Descriptor)

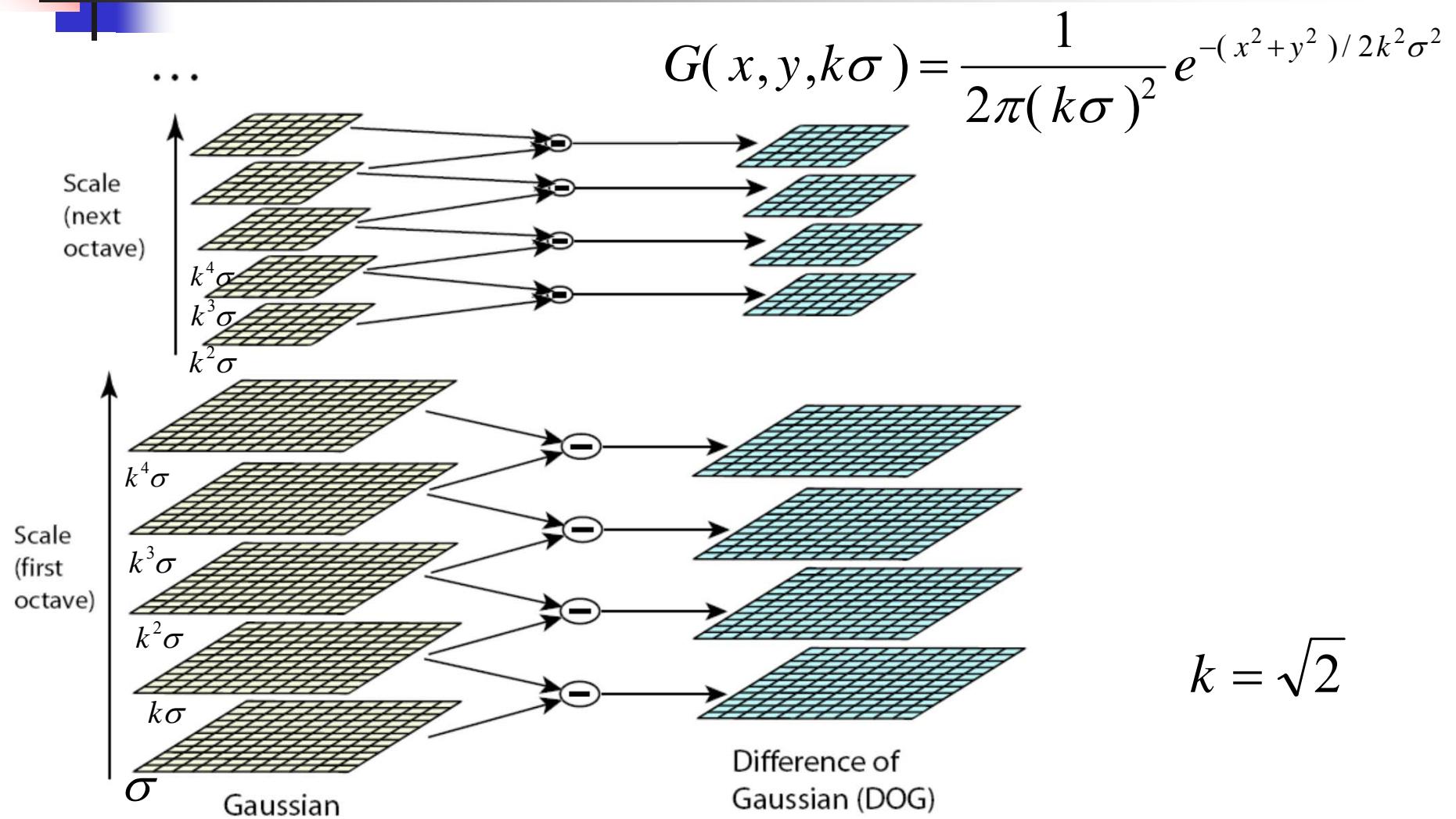
# Building a Scale Space

...

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



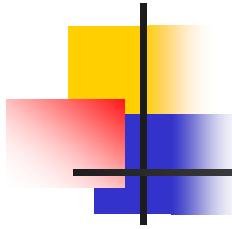
# Building a Scale Space





	scale →			
octave	0.707107	1.000000	1.414214	2.000000
	1.414214	2.000000	2.828427	4.000000
	2.828427	4.000000	5.656854	8.000000
	5.656854	8.000000	11.313708	16.000000
				22.627417

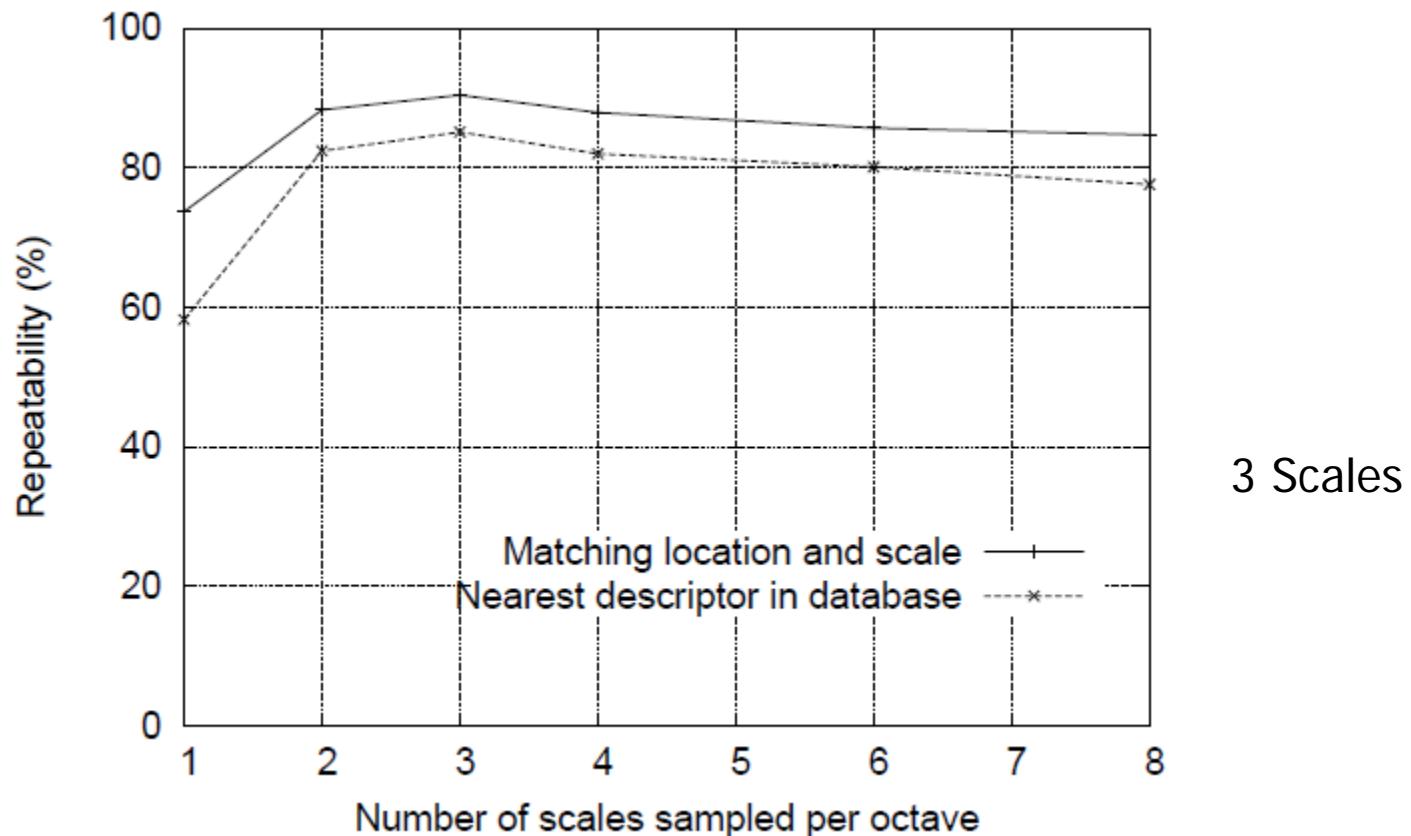
$$\sigma = .707187.6; k = \sqrt{2}$$



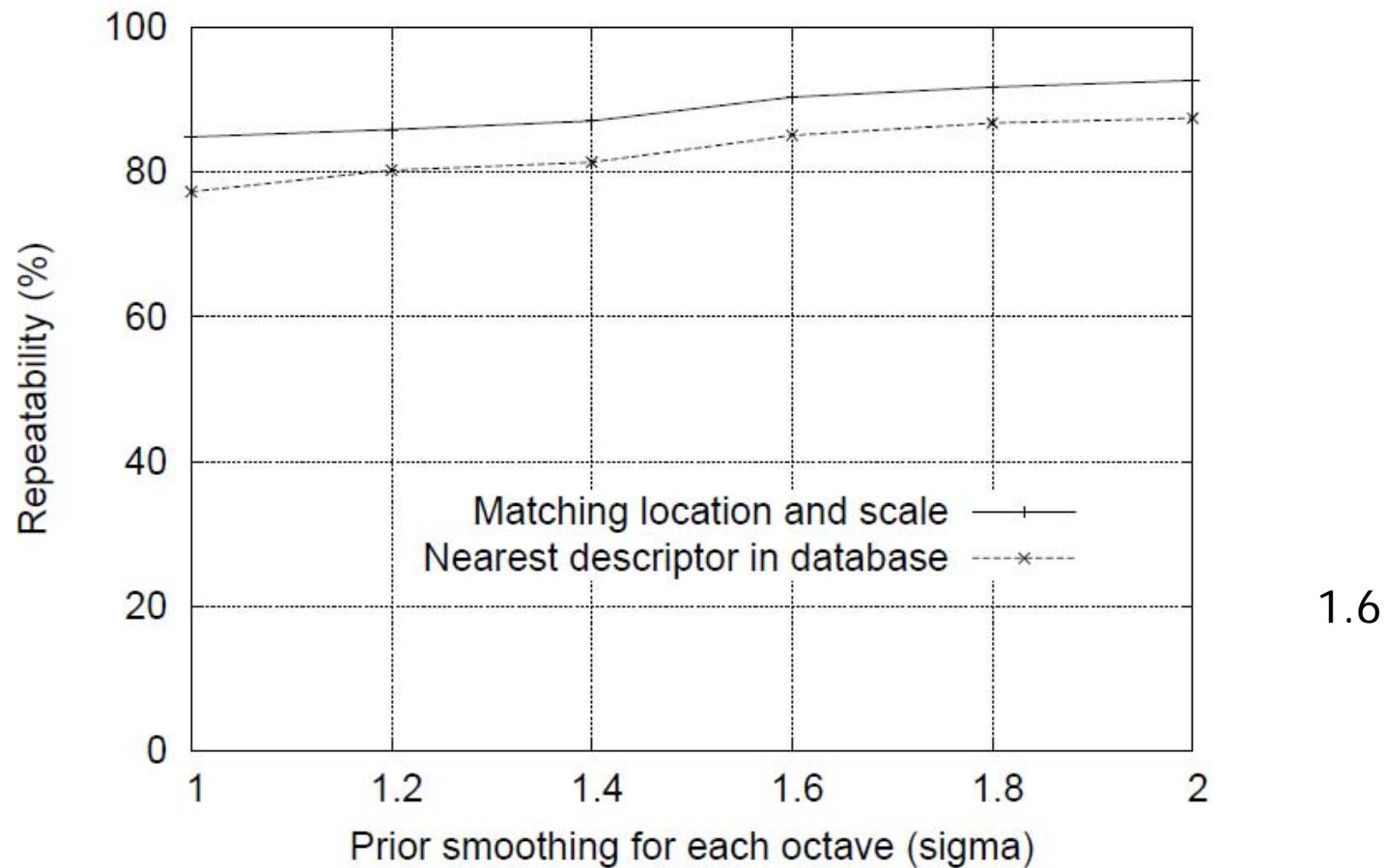
# How many scales per octave?

- A collection of 32 real images drawn from a diverse range, including
  - outdoor scenes, human faces, aerial photographs, and industrial
- **Each image was then subject to a range of transformations:**
  - rotation, scaling, affine stretch, change in brightness and contrast, and addition of image noise.

# How many scales per octave?

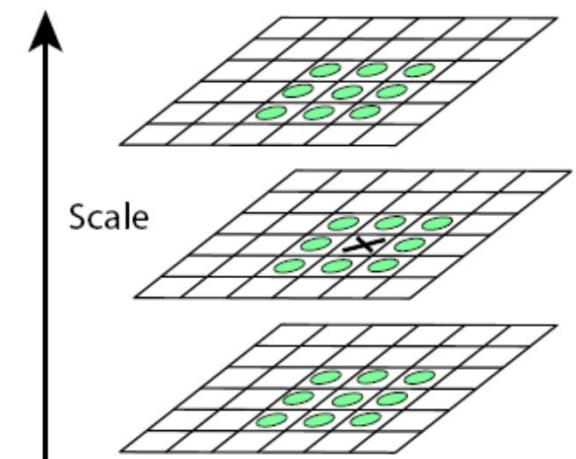


# Initial value of sigma



# Scale Space Peak Detection

- Compare a pixel (**X**) with 26 pixels in current and adjacent scales (**Green Circles**)
- Select a pixel (**X**) if larger/smaller than all 26 pixels
- Large number of extrema, computationally expensive
  - Detect the most stable subset with a coarse sampling of scales

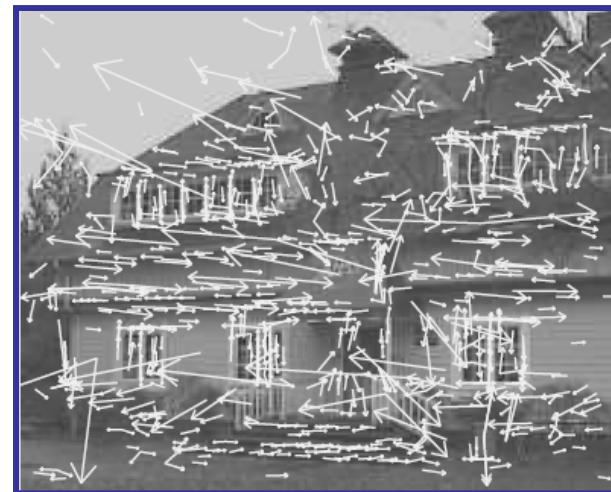


# Key Point Localization

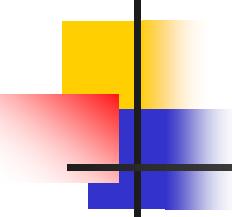
- Candidates are chosen from extrema detection



original image



extrema locations



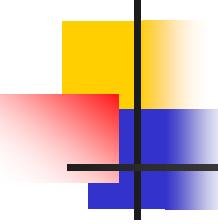
# Initial Outlier Rejection

- 1. Low contrast candidates
- 2. Poorly localized candidates along an edge
- Taylor series expansion of DOG,  $D$ .

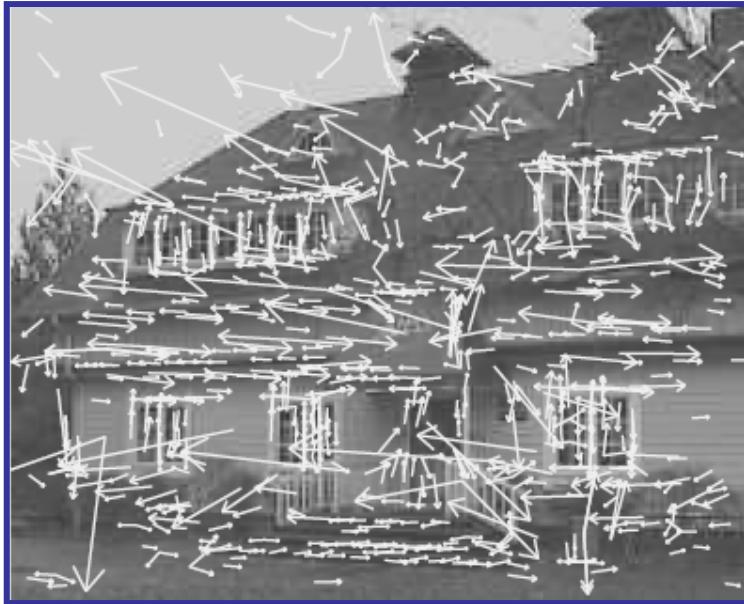
$$D(\mathbf{x}) = D + \frac{\partial D}{\partial \mathbf{x}}^T \mathbf{x} + \frac{1}{2} \mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2} \mathbf{x}$$

$\mathbf{x} = (x, y, \sigma)^T$  Homework

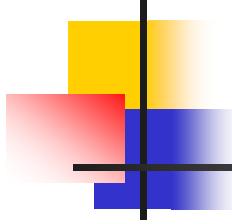
- Minima or maxima is located at  $\hat{\mathbf{x}} = -\frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1} \frac{\partial D}{\partial \mathbf{x}}$
- Value of  $D(\mathbf{x})$  at minima/maxima must be large,  $|D(x)| > th$ .



# Initial Outlier Rejection

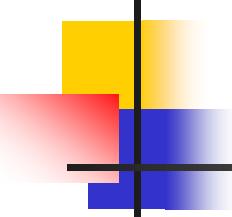


from 832 key points to 729 key points,  $\text{th}=0.03$ .



# Further Outlier Rejection

- DOG has strong response along edge
- Assume DOG as a surface
  - Compute principal curvatures (PC)
  - Poorly defined peak will have very low curvature along the edge, high across the edge



# Further Outlier Rejection

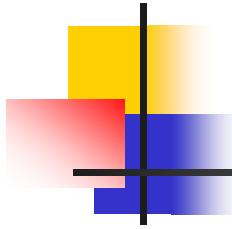
- Use Trace and det similar to Harris corner detector
- Compute Hessian of D (principal curvature)

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad Tr(H) = D_{xx} + D_{yy} = \lambda_1 + \lambda_2$$
$$Det(H) = D_{xx}D_{yy} - D_{xy}^2 = \lambda_1 \lambda_2$$

- Remove outliers by evaluating

$$\frac{Tr(H)^2}{Det(H)} = \frac{(r+1)^2}{r} \quad r = \frac{\lambda_1}{\lambda_2}$$

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\lambda_1 + \lambda_2)^2}{\lambda_1 \lambda_2} = \frac{(r\lambda_2 + \lambda_2)^2}{r\lambda_2^2} = \frac{(r+1)^2}{r}$$



# Further Outlier Rejection

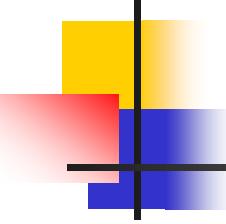
- Following quantity is minimum (eigen values are equal) when  $r=1$
- It increases with  $r$

$$r = \frac{\lambda_1}{\lambda_2}$$

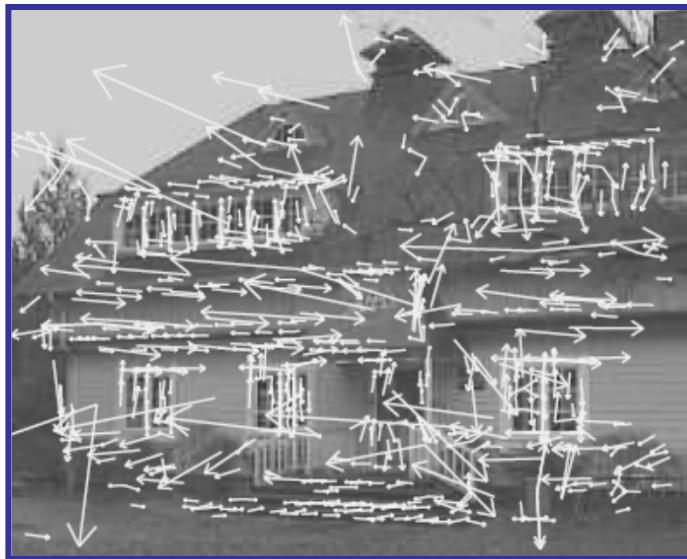
$$\frac{Tr(H)^2}{Det(H)} = \frac{(r+1)^2}{r}$$

- Eliminate key points if

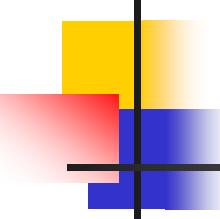
$$\frac{Tr(H)^2}{Det(H)} \prec \frac{(r+1)^2}{r} \quad r > 10$$



# Further Outlier Rejection



from 729 key points to 536 key points.



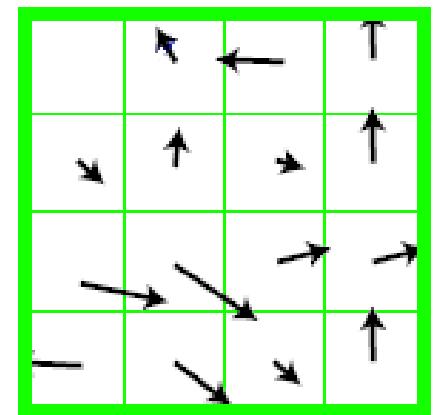
# Orientation Assignment

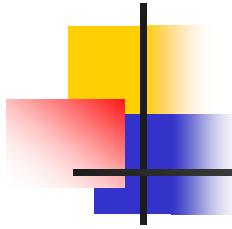
- To achieve rotation invariance
- Compute central derivatives, gradient magnitude and direction of  $L$  (smooth image) at the scale of key point  $(x,y)$

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

# Orientation Assignment

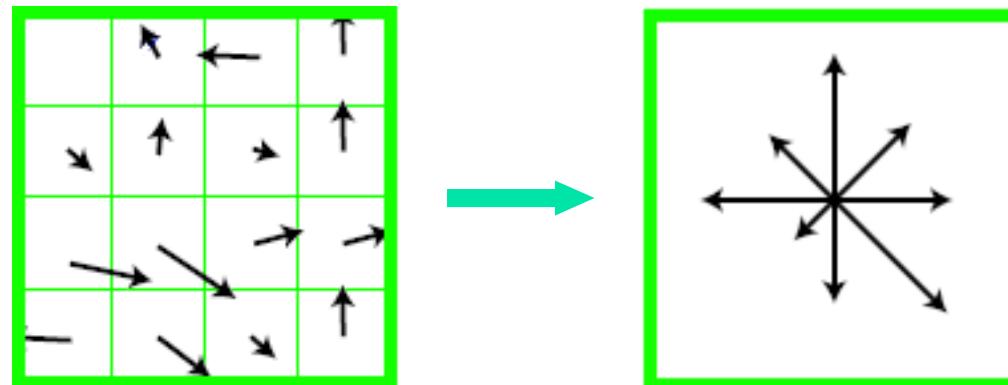
- Create a weighted direction histogram in a neighborhood of a key point (36 bins)
- Weights are
  - Gradient magnitudes
  - Spatial gaussian filter with  $\sigma=1.5 \times \text{scale of key point}$

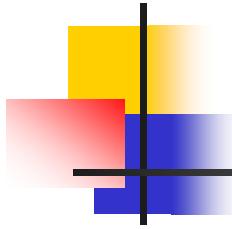




# Orientation Assignment

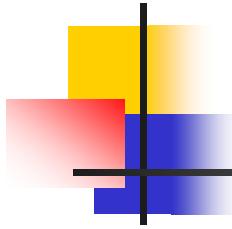
- Select the peak as direction of the key point
- Introduce additional key points at same location
  - if another peak is within 80% of max peak of the histogram with different directions





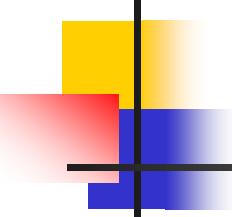
# Local Image Descriptors at Key Points

- Possible descriptor
  - Store intensity samples in the neighborhood
  - Sensitive to lighting changes, 3D object transformation
- Use of gradient orientation histograms
  - Robust representation



# Similarity to IT cortex

- Complex neurons respond to a gradient at a particular orientation.
- Location of the feature can shift over a small receptive field.



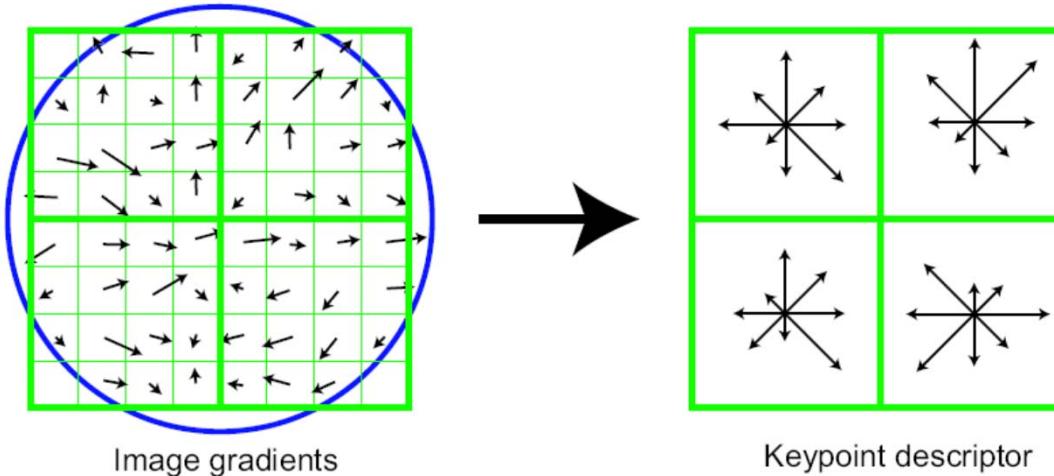
# Tomaso Poggio, MIT

- Edelman, Intrator, and Poggio (1997)
  - The function of the cells allow for matching and recognition of 3D objects from a range of view points.
- Experiments show better recognition accuracy for 3D objects rotated in depth by up to 20 degrees

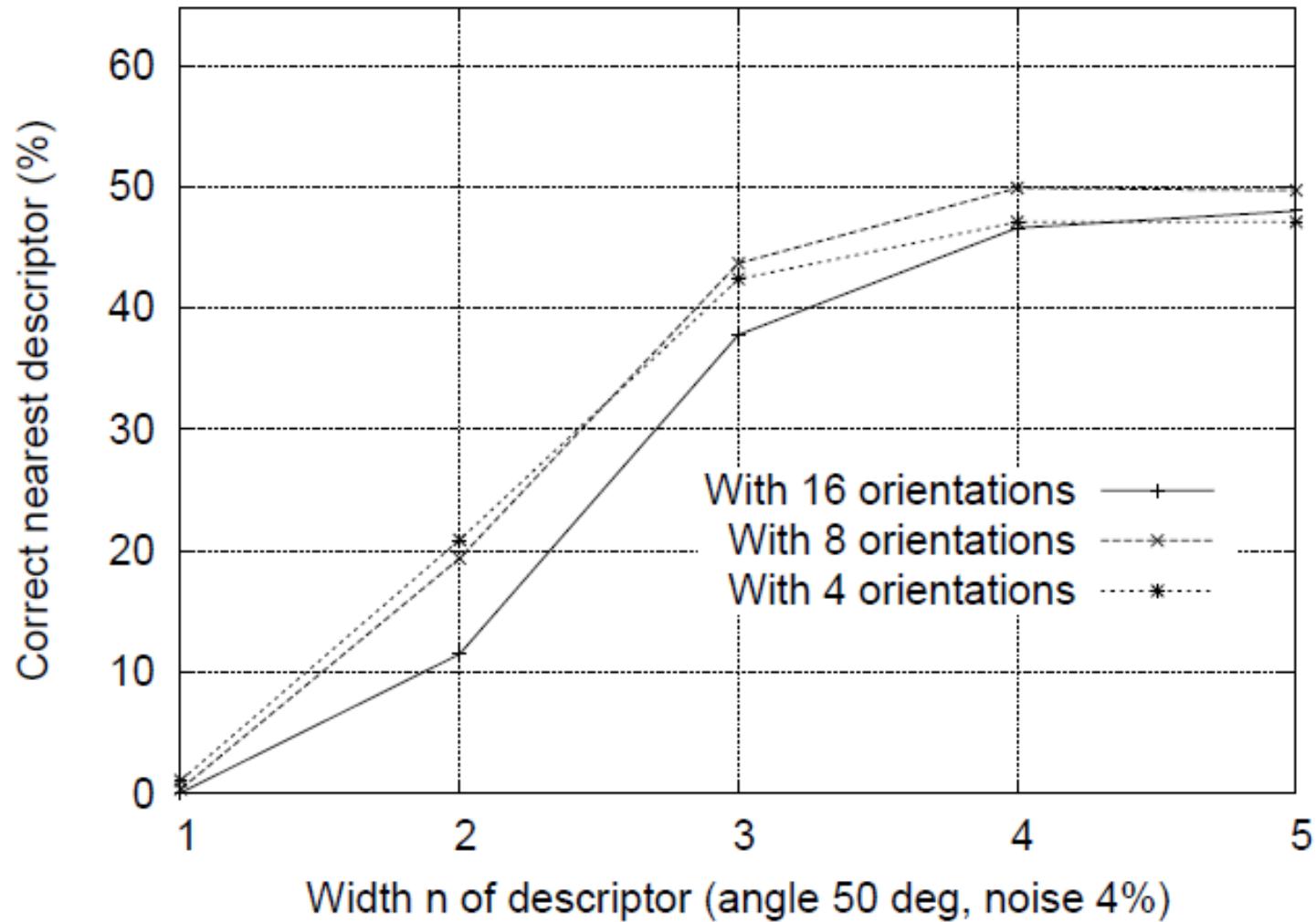


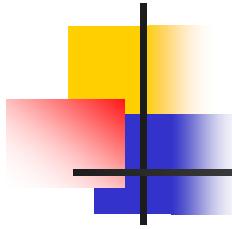
# Extraction of Local Image Descriptors at Key Points

- Compute relative orientation and magnitude in a  $16 \times 16$  neighborhood at key point
- Form weighted histogram (8 bin) for  $4 \times 4$  regions
  - Weight by magnitude and spatial Gaussian
  - Concatenate 16 histograms in one long vector of 128 dimensions
- Example for  $8 \times 8$  to  $2 \times 2$  descriptors



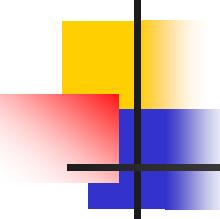
# Descriptor Regions (n by n)





# Extraction of Local Image Descriptors at Key Points

- Store numbers in a vector
- Normalize to unit vector (**UN**)
  - Illumination invariance (affine changes)
- For non-linear intensity transforms
  - Bound **Unit Vector** items to maximum 0.2 (remove gradients larger than 0.2)
  - Renormalize to unit vector



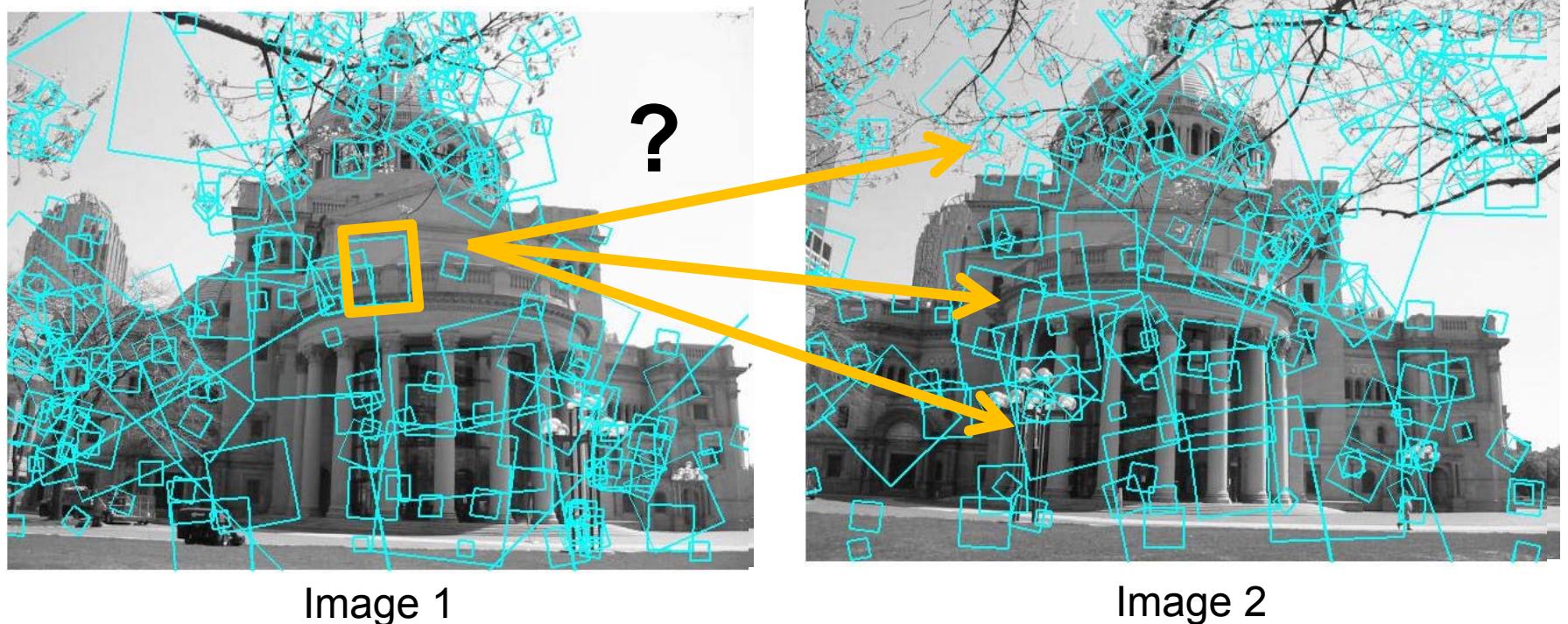
# Key point matching

- Match the key points against a database of that obtained from training images.
- Find the nearest neighbor i.e. a key point with minimum Euclidean distance.
  - Efficient Nearest Neighbor matching
    - Looks at ratio of distance between best and 2<sup>nd</sup> best match (.8)

# Matching local features



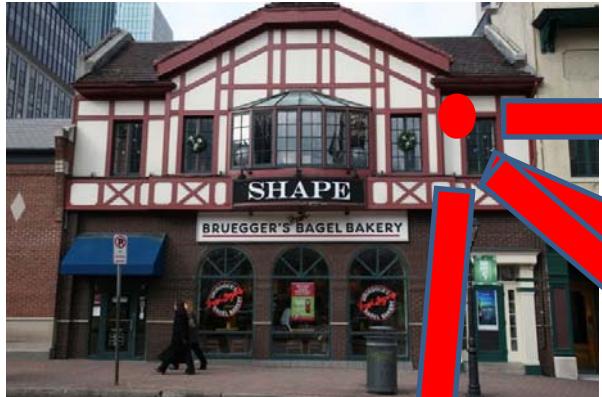
# Matching local features



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

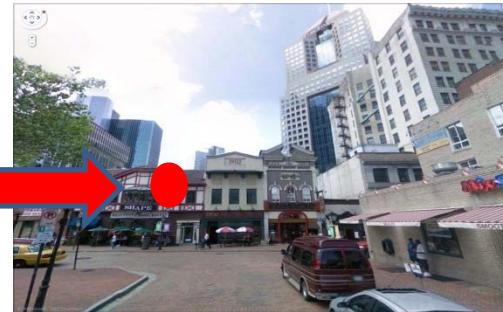


Query Image



Query Image

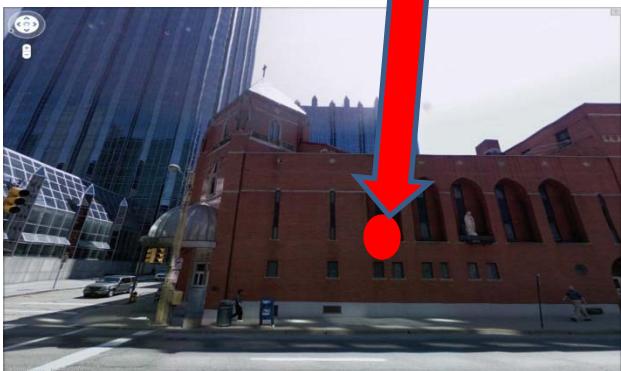
1<sup>st</sup> NN



2<sup>nd</sup> NN



3<sup>rd</sup> NN



4<sup>th</sup> NN



if  $\frac{\text{Distance to first match}}{\text{Distance to second match}} < .8$  Goodmatch

# Ambiguous matches



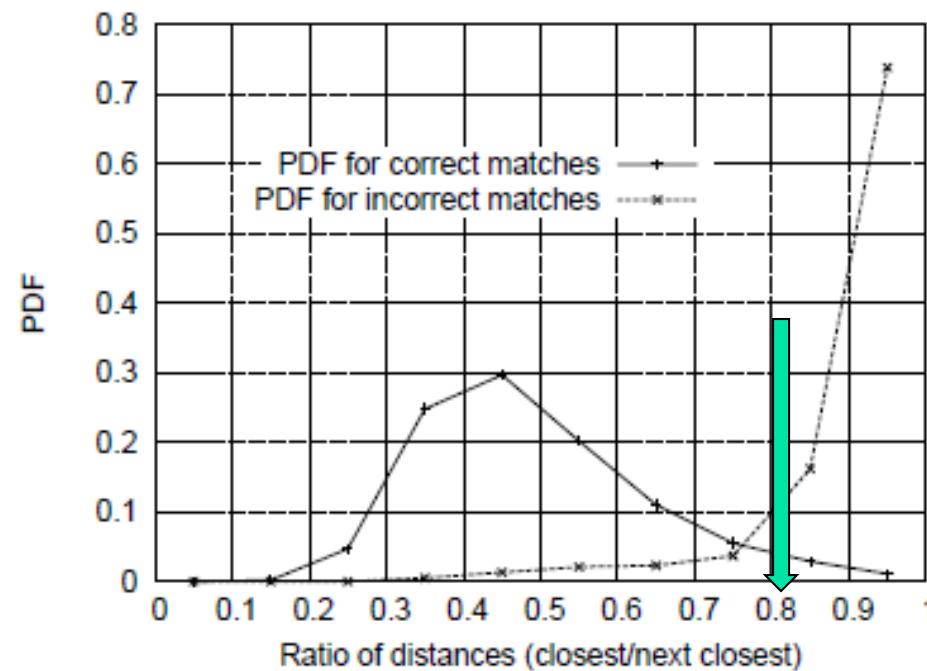
Image 1

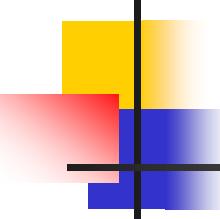


Image 2

- At what distance do we have a good match?
- To add robustness to matching, can consider **ratio** : distance to best match / distance to second best match
- If low, first match looks good.
- If high, could be ambiguous match.

# The ratio of distance from the closest to the distance of the second closest





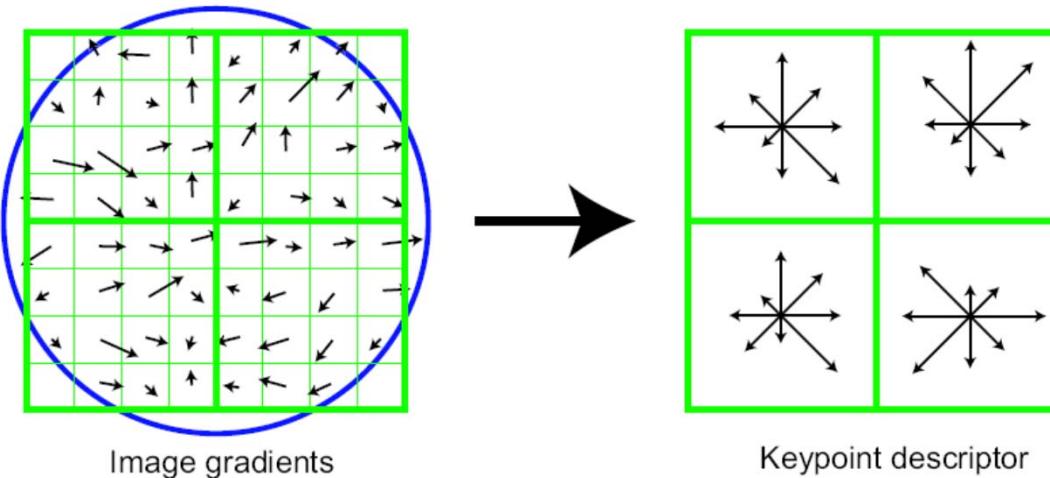
# SIFT Detector

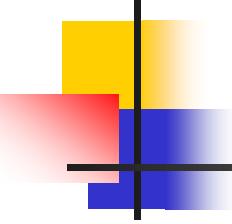
---

- Generate Scale Space of an Image
- Detect Peaks in Scale Space (extrema)
- Localize Interest Points (Taylor Series)
- Remove outliers (remove response along edges)
- Assign Orientation

# SIFT Descriptor

- Compute relative orientation and magnitude in a  $16 \times 16$  neighborhood at key point
- Form weighted histogram (8 bin) for  $4 \times 4$  regions
  - Weight by magnitude and spatial Gaussian
  - Concatenate 16 histograms in one long vector of 128 dimensions
- Example for  $8 \times 8$  to  $2 \times 2$  descriptors





# Reference

---

D. Lowe. *Distinctive image features from scale-invariant key points.*, International Journal of Computer Vision 2004.

# Histograms of Oriented Gradients for Human Detection **(HOG)**

N. Dalal and B. Triggs  
CVPR 2005

Cited by 8908

# Histograms of Oriented Gradients for Human Detection

Navneet Dalal and Bill Triggs

INRIA Rhône-Alps, 655 avenue de l'Europe, Montbonnot 38334, France  
 [{Navneet.Dalal,Bill.Triggs}@inrialpes.fr](mailto:{Navneet.Dalal,Bill.Triggs}@inrialpes.fr), <http://lear.inrialpes.fr>

## Abstract

We study the question of feature sets for robust visual object recognition, adopting linear SVM based human detection as a test case. After reviewing existing edge and gradient based descriptors, we show experimentally that grids of Histograms of Oriented Gradient (HOG) descriptors significantly outperform existing feature sets for human detection. We study the influence of each stage of the computation on performance, concluding that fine-scale gradients, fine orientation binning, relatively coarse spatial binning, and high-quality local contrast normalization in overlapping descriptor blocks are all important for good results. The new approach gives near-perfect separation on the original MIT pedestrian database, so we introduce a more challenging dataset containing over 1800 annotated human images with a large range of pose variations and backgrounds.

## 1 Introduction

Detecting humans in images is a challenging task owing to their variable appearance and the wide range of poses that they can adopt. The first need is a robust feature set that allows the human form to be discriminated cleanly, even in cluttered backgrounds under difficult illumination. We study the issue of feature sets for human detection, showing that locally normalized Histogram of Oriented Gradient (HOG) descriptors provide excellent performance relative to other existing feature sets including wavelets [17, 22]. The proposed descriptors are reminiscent of edge orientation histograms [4,5], SIFT descriptors [12] and shape contexts [1], but they are computed on a dense grid of uniformly spaced cells and they use overlapping local contrast normalizations for improved performance. We make a detailed study of the effects of various implementation choices on detector performance, taking “pedestrian detection” (the detection of mostly visible people in more or less upright poses) as a test case. For simplicity and speed, we use linear SVM as a baseline classifier throughout the study. The new detectors give essentially perfect results on the MIT pedestrian test set [18, 17], so we have created a more challenging set containing over 1800 pedestrian images with a large range of poses and backgrounds. Ongoing work suggests that our feature set performs equally well for other shape-based object classes.

We briefly discuss previous work on human detection in §2, give an overview of our method §3, describe our data sets in §4 and give a detailed description and experimental evaluation of each stage of the process in §§5–6. The main conclusions are summarized in §7.

## 2 Previous Work

There is an extensive literature on object detection, but here we mention just a few relevant papers on human detection [18, 17, 22, 16, 20]. See [6] for a survey. Papageorgiou *et al* [18] describe a pedestrian detector based on a polynomial SVM using rectified Haar wavelets as input descriptors, with a parts (subwindow) based variant in [17]. Depontet *et al* give an optimized version of this [2]. Gavrila & Philomin [8] take a more direct approach, extracting edge images and matching them to a set of learned exemplars using chamfer distance. This has been used in a practical real-time pedestrian detection system [7]. Viola *et al* [22] build an efficient moving person detector, using AdaBoost to train a chain of progressively more complex region rejection rules based on Haar-like wavelets and space-time differences. Ronfard *et al* [19] build an articulated body detector by incorporating SVM based limb classifiers over 1<sup>st</sup> and 2<sup>nd</sup> order Gaussian filters in a dynamic programming framework similar to those of Felzenszwalb & Huttenlocher [3] and Ioffe & Forsyth [9]. Mikolajczyk *et al* [16] use combinations of orientation-position histograms with binary-thresholded gradient magnitudes to build a parts based method containing detectors for faces, heads, and front and side profiles of upper and lower body parts. In contrast, our detector uses a simpler architecture with a single detection window, but appears to give significantly higher performance on pedestrian images.

## 3 Overview of the Method

This section gives an overview of our feature extraction chain, which is summarized in fig. 1. Implementation details are postponed until §6. The method is based on evaluating well-normalized local histograms of image gradient orientations in a dense grid. Similar features have seen increasing use over the past decade [4, 5, 12, 15]. The basic idea is that local object appearance and shape can often be characterized rather well by the distribution of local intensity gradients or



Dr. Edgar Seemann

# HOG Steps

- HOG feature extraction
  - Compute centered horizontal and vertical gradients with no smoothing
  - Compute gradient orientation and magnitudes
    - For color image, pick the color channel with the highest gradient magnitude for each pixel.
  - For a 64x128 image,
  - Divide the image into 16x16 blocks of 50% overlap.
    - $7 \times 15 = 105$  blocks in total
  - Each block should consist of 2x2 cells with size 8x8.
  - Quantize the gradient orientation into 9 bins
    - The vote is the gradient magnitude
    - Interpolate votes between neighboring bin center.
    - The vote can also be weighted with Gaussian to downweight the pixels near the edges of the block.
  - Concatenate histograms (Feature dimension:  $105 \times 4 \times 9 = 3,780$ )

# Computing Gradients

- Centered:  $f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x-h)}{2h}$

- Filter masks in x and y directions

- Centered:

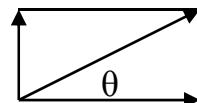
$$\begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline\end{array}$$

$$\begin{array}{|c|}\hline -1 \\ \hline 0 \\ \hline 1 \\ \hline\end{array}$$

- Gradient

- Magnitude:

$$s = \sqrt{s_x^2 + s_y^2}$$

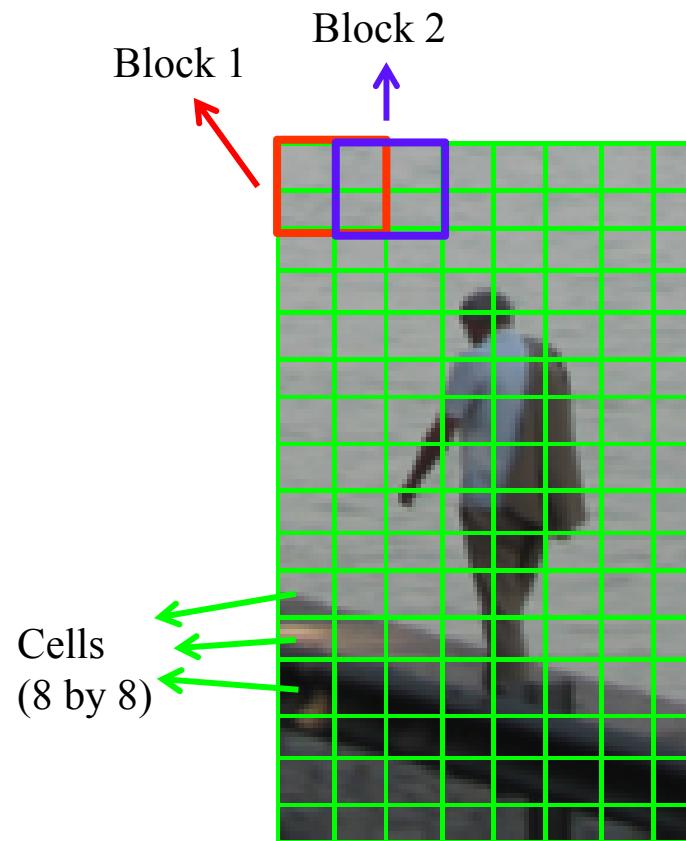


- Orientation:

$$\theta = \arctan\left(\frac{s_y}{s_x}\right)$$

# Blocks, Cells

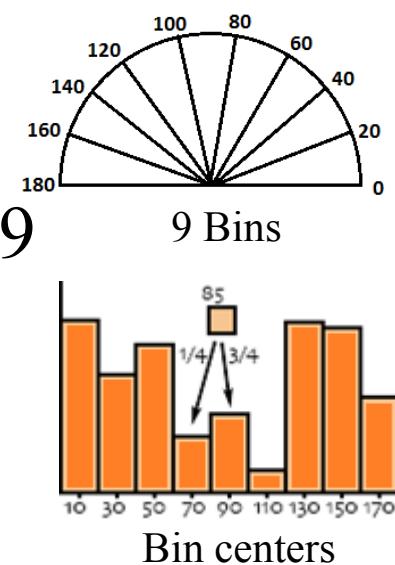
- 16x16 blocks of 50% overlap.
  - $7 \times 15 = 105$  blocks in total
- Each block should consist of 2x2 cells with size 8x8.



# Votes

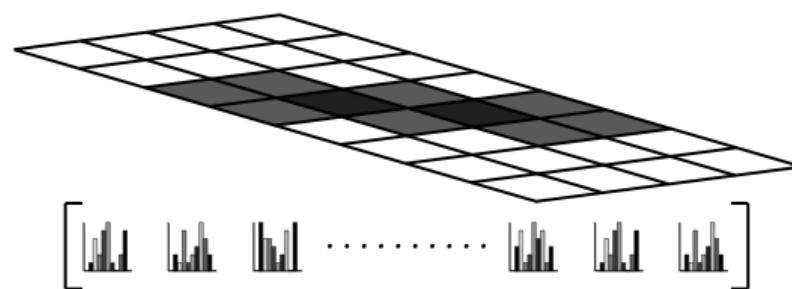
- Each block consists of 2x2 cells with size 8x8
- Quantize the gradient orientation into 9 bins (0-180)

- The vote is the gradient magnitude
- Interpolate votes linearly between neighboring bin centers.
  - Example: if  $\theta=85$  degrees.
  - Distance to the bin center Bin 70 and Bin 90 are 15 and 5 degrees, respectively.
  - Hence, ratios are  $5/20=1/4$ ,  $15/20=3/4$ .
- The vote can also be weighted with Gaussian to down weight the pixels near the edges of the block.

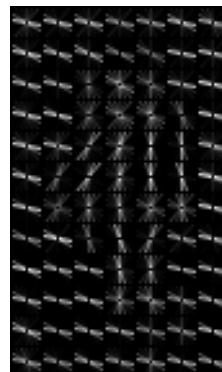


# Final Feature Vector

- Concatenate histograms
  - Make it a 1D vector of length 3780.



- Visualization



# Results

Navneet Dalal and Bill Triggs "Histograms of Oriented Gradients for Human Detection" CVPR05



# SIFT Vs HOG

## SIFT

- 128 dimensional vector
- 16 by 16 window
- 4x4 sub-window (16 total)
- 8 bin histogram

## HOG

- 3,780 dimensional vector
- 64 by 128 window
- 16 by 16 blocks with overlap
- Each block consists of 2 by 2 cells each of 8 by 8
- Overlapping
- 9 bin histogram