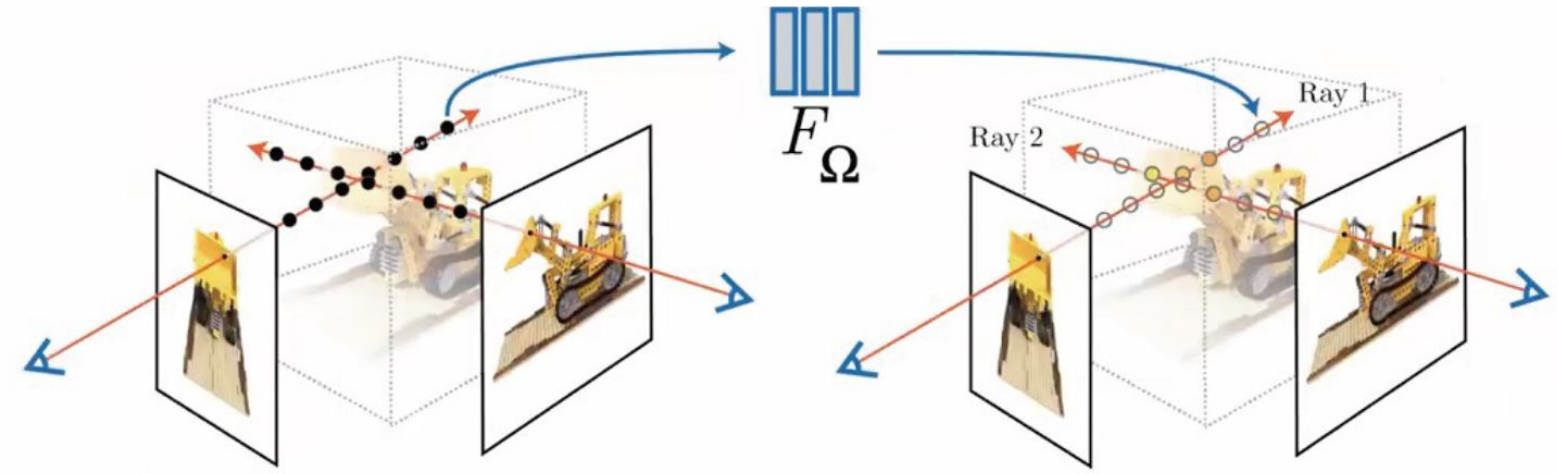
Vision Computacional
par imagenes y videos

Modulo 3.2: Extracción de Descriptores



Gilberto Ochoa Ruiz, PhD
Associate Professor
Researcher in Computer Vision

Computer Science Dept.
Advanced AI Group
gilberto.ochoa@tec.mx

Tecnológico de Monterrey

TODAY'S AGENDA

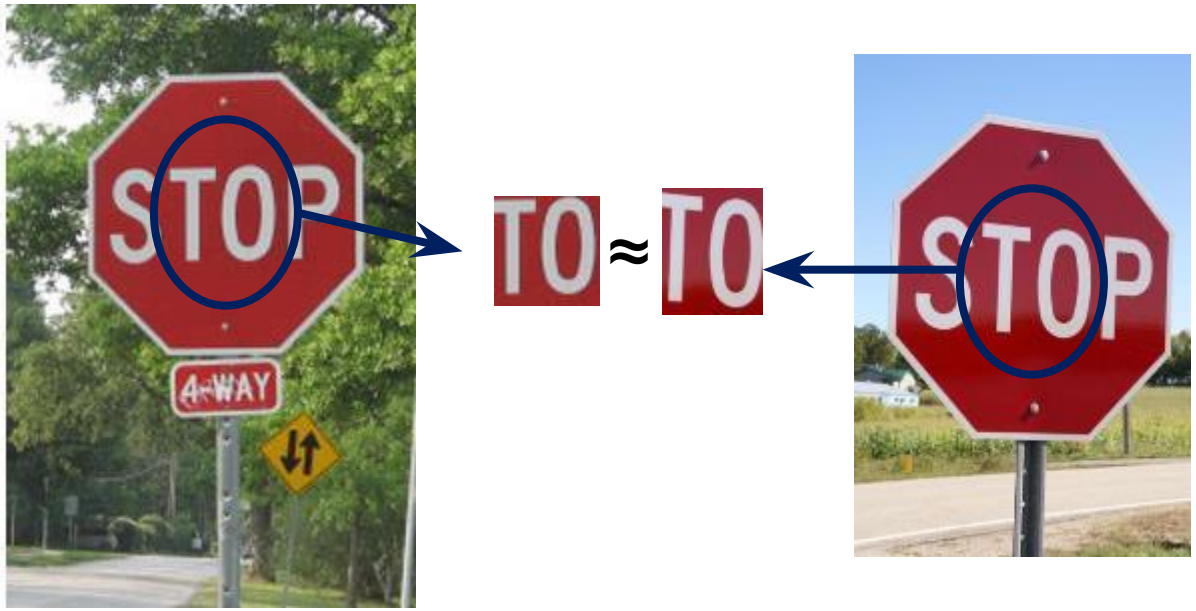
introduction to our course

1. Today class goal
2. About us
3. Course policies
4. Evaluation schema
5. Summary and conclusions

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Matching images
A challenge for computer vision systems!



Encontrar correspondencias entre múltiples imágenes: puntos de interés
Correspondencias: puntos, parches, líneas o regiones en las imágenes

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

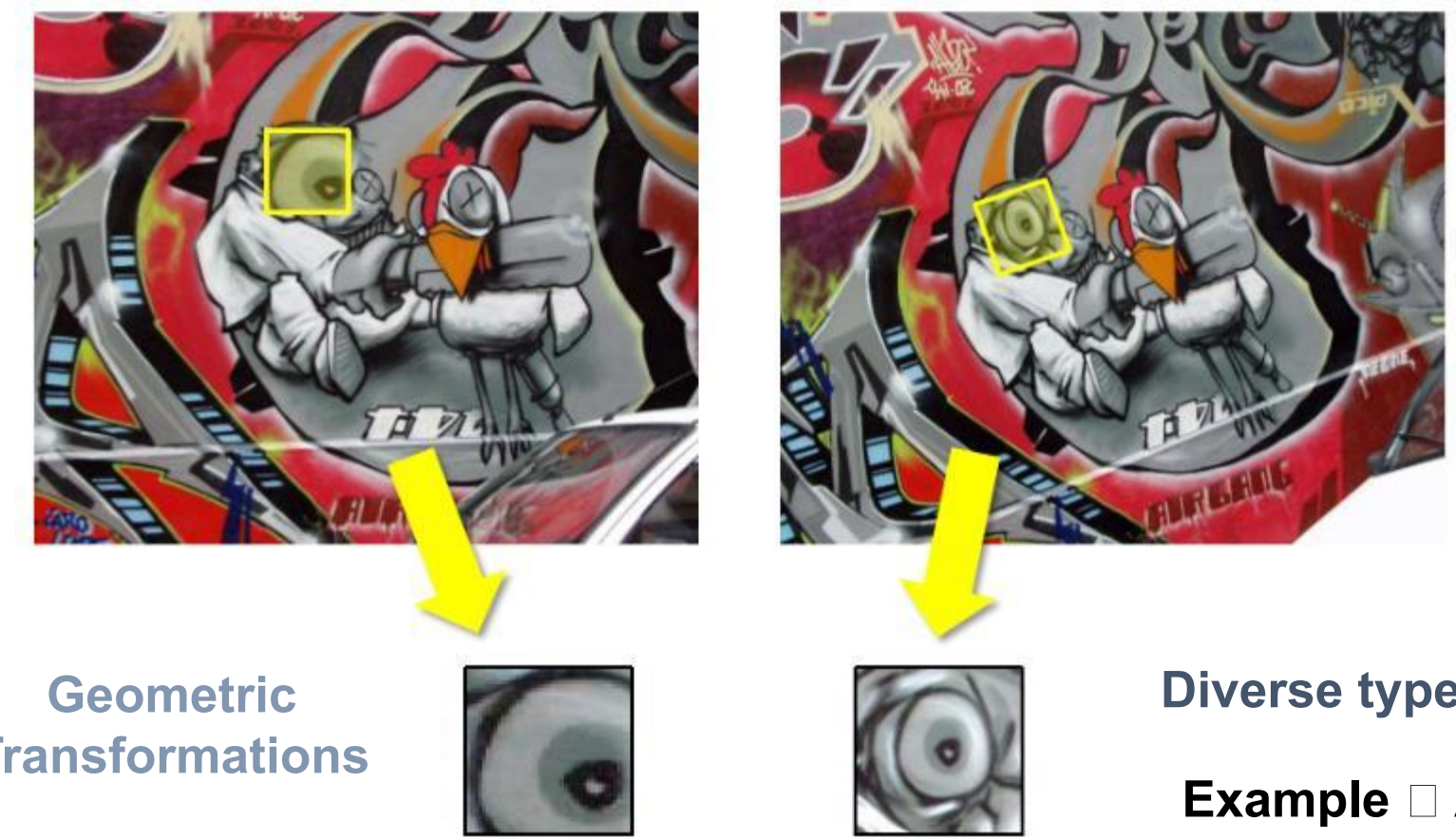
Computer Vision Challenge

Non-trivial problem: changes in perspective between views
It requires then: invariance in traslation, rotation, scale

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Computer Vision Challenge



Geometric Transformations

Diverse types exist

Example □ Affine

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Motivation

It enables to map 3D points to images

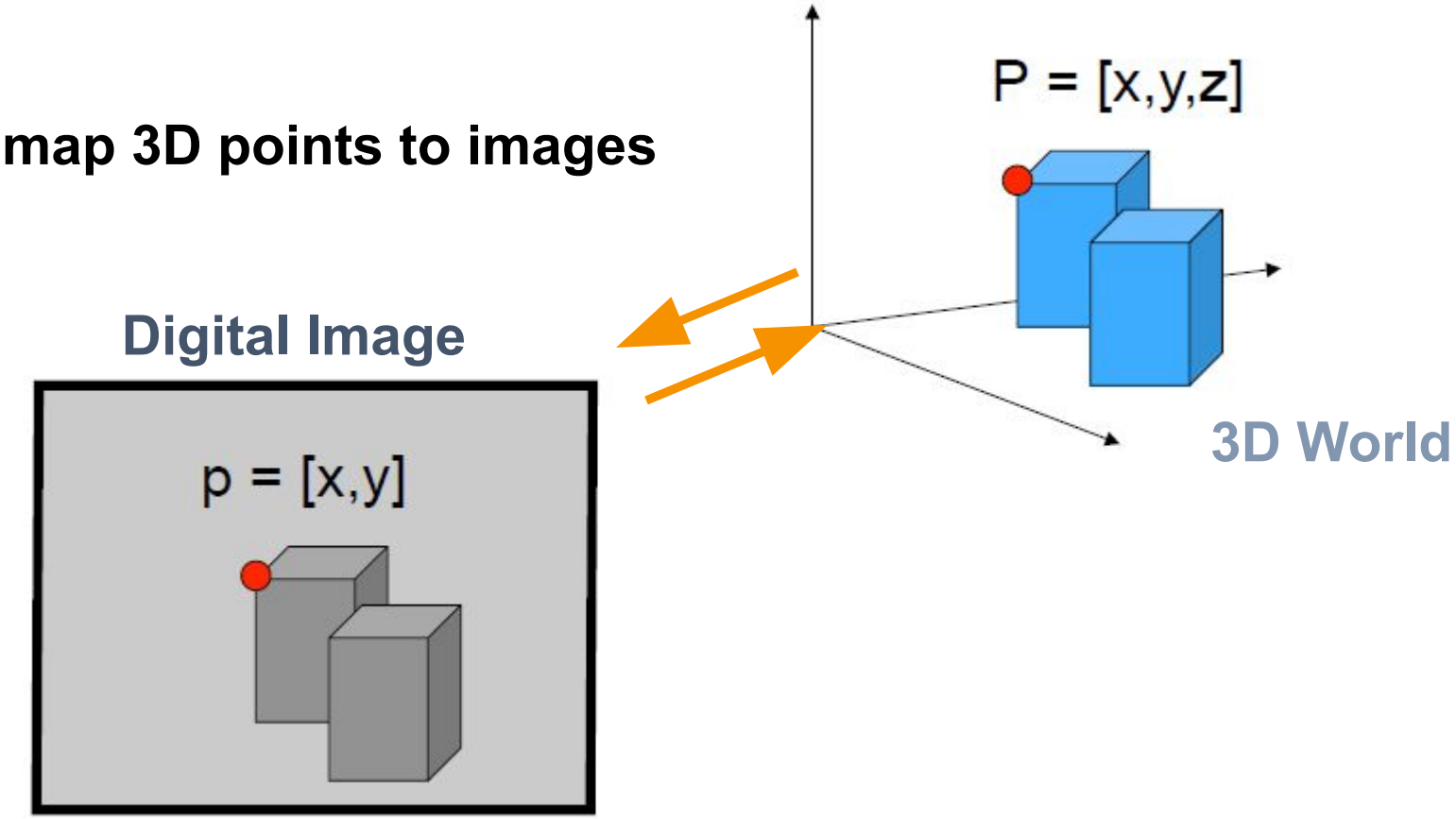


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Motivation

Image alignment

Image Stitching

3D Reconstruction

Object tracking in video

Drone and Robot Navigation

Image Retrieval

Object Recognition

Biometrics Applications

Note: “interest points” = “keypoints”, aka “features points”

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Extreme example! □ NASAs Mars Rover

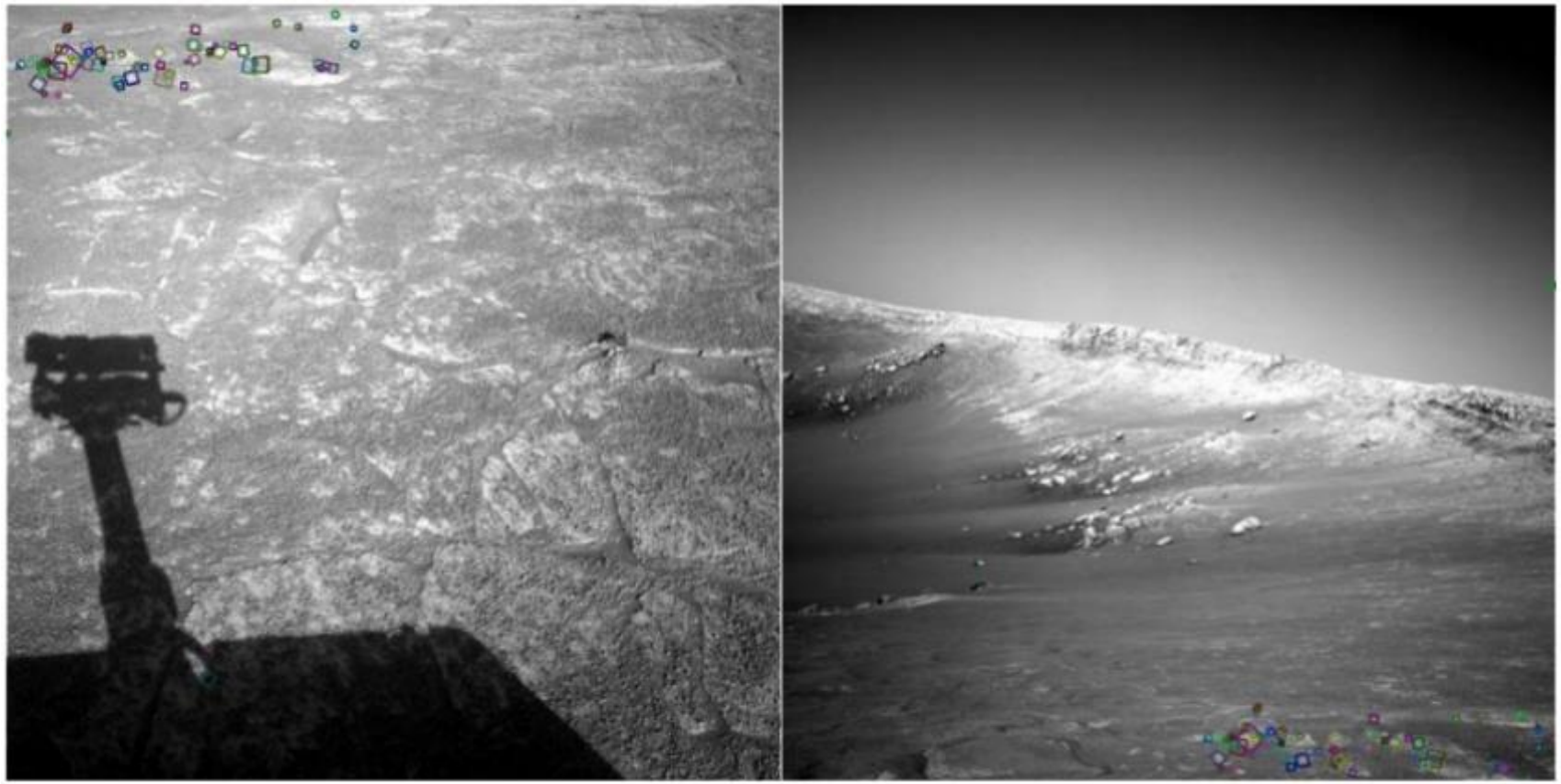
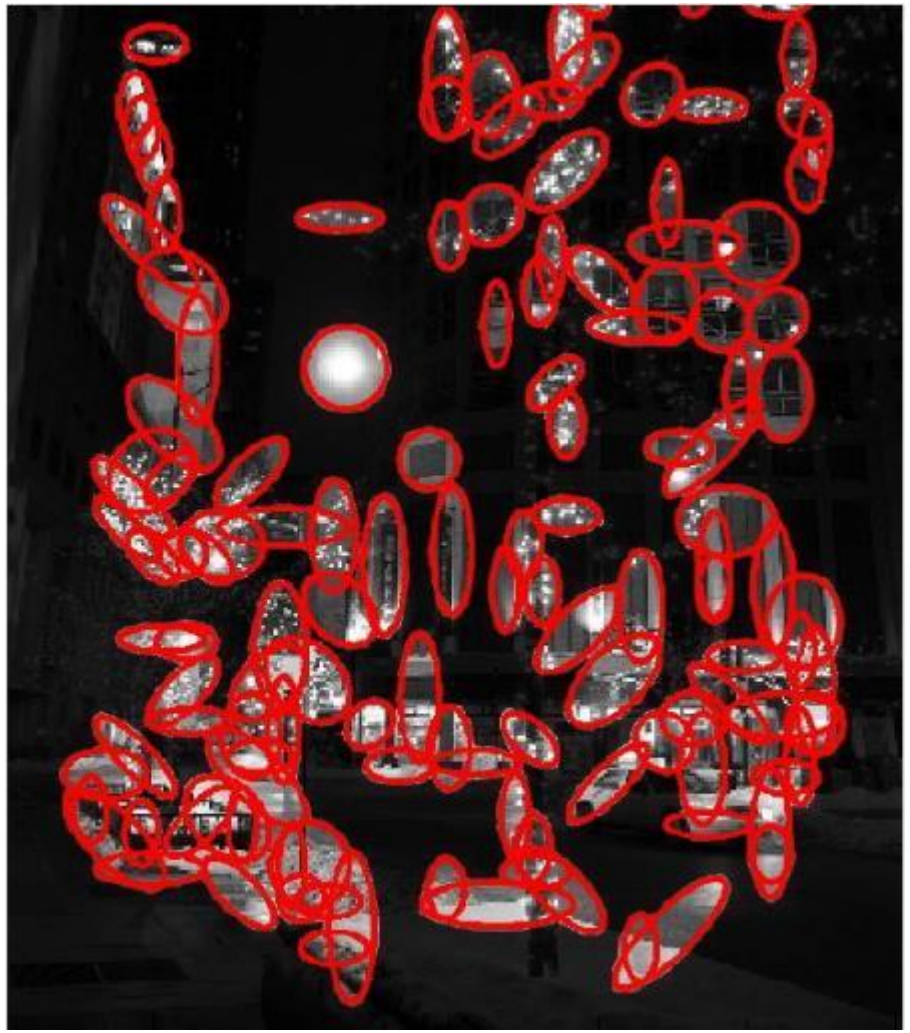


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

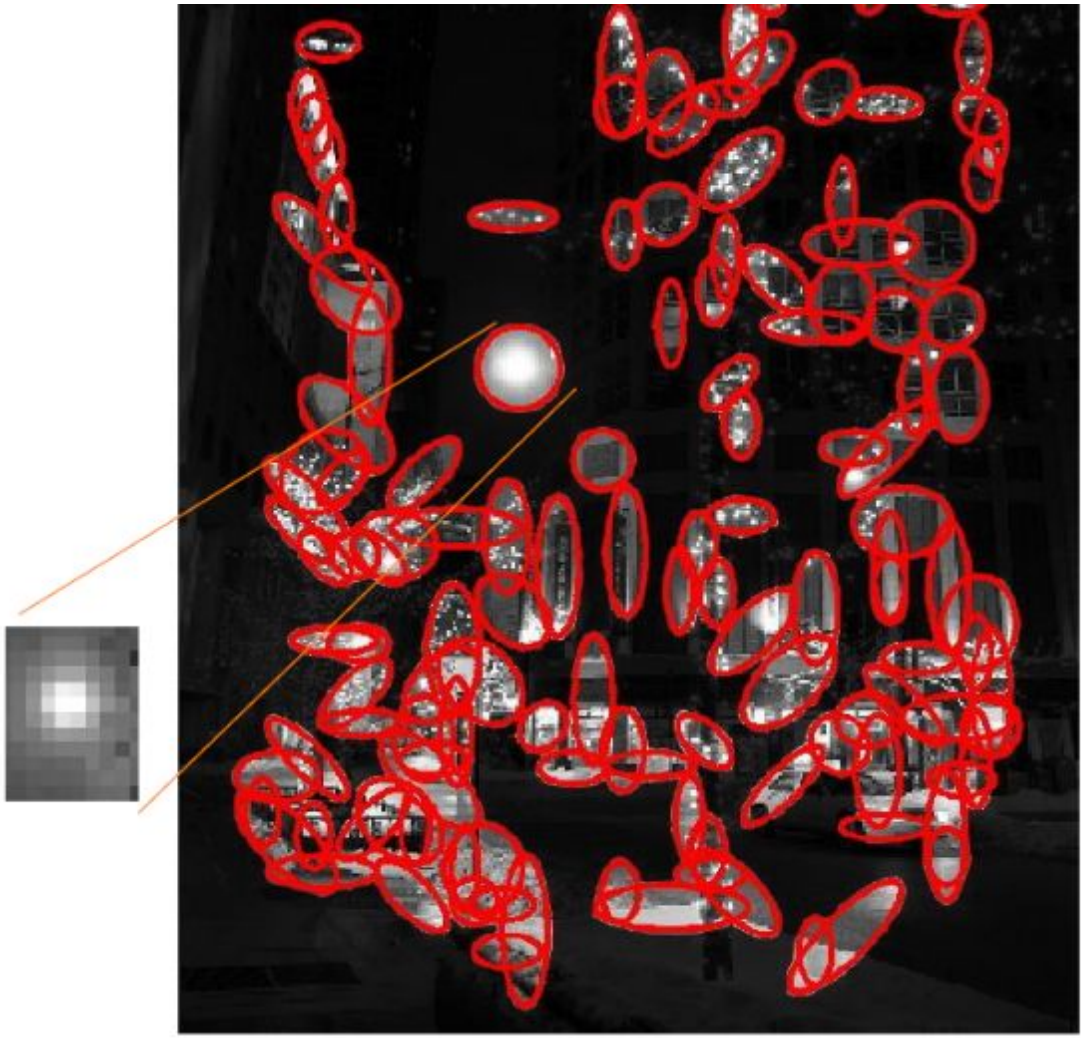


Feature Detection

e.g. DoG

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



Feature Detection

e.g. DoG

Feature Description

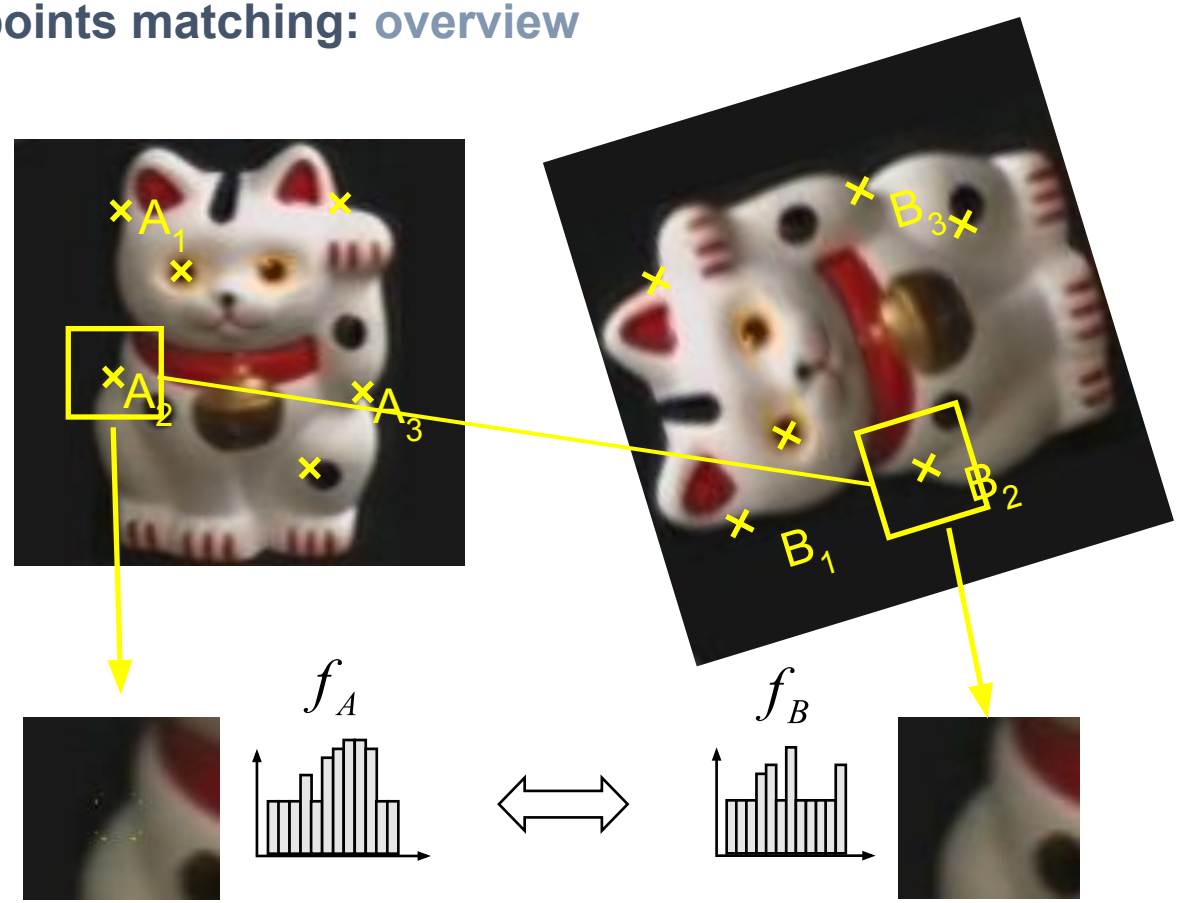
e.g. SIFT

- Estimation
- Matching
- Indexing
- Detection

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Keypoints matching: overview



$$d(f_A, f_B) < T$$

1. Find a set of **distinctive keypoints**

2. Define a **region** around each keypoint

3. Calculate a **local descriptor** from the normalized region

4. Find **correspondences** between descriptors

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

The content of an image is transfotmed into keypoints coordinates, which are **invariant to traslation, roration and scale change**

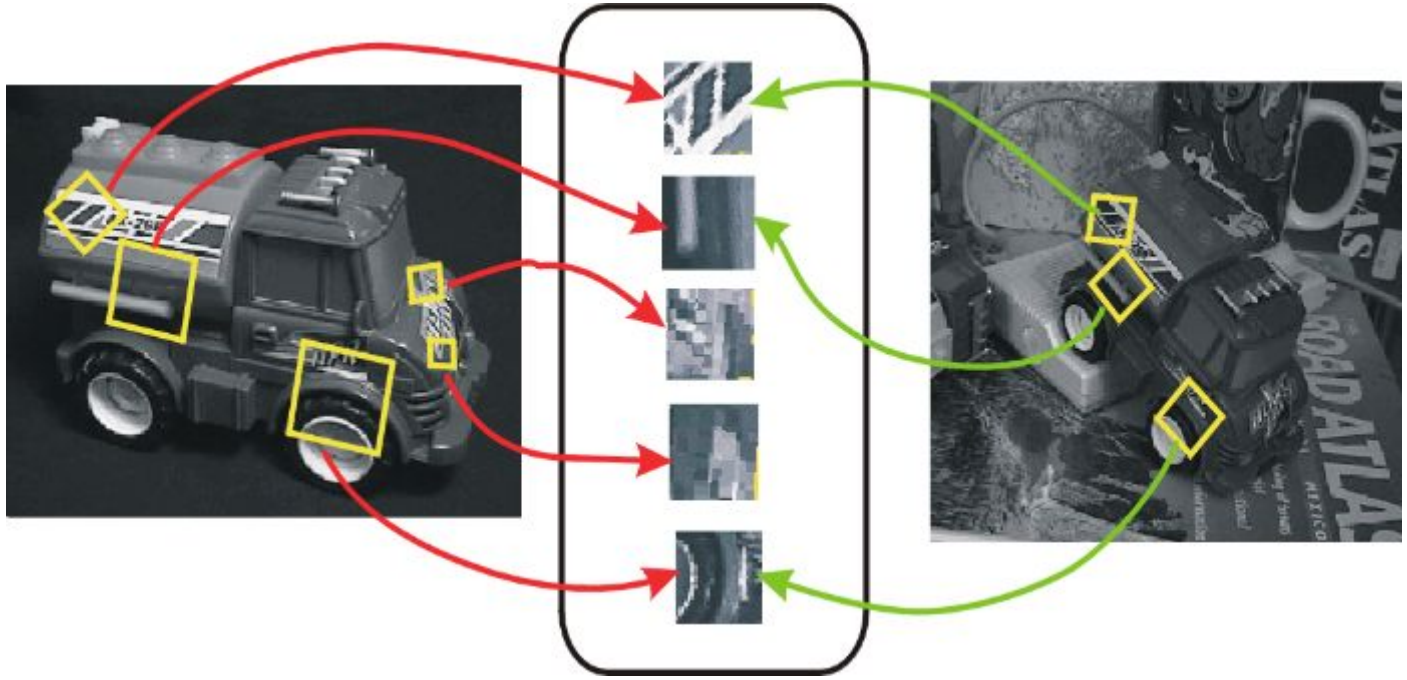
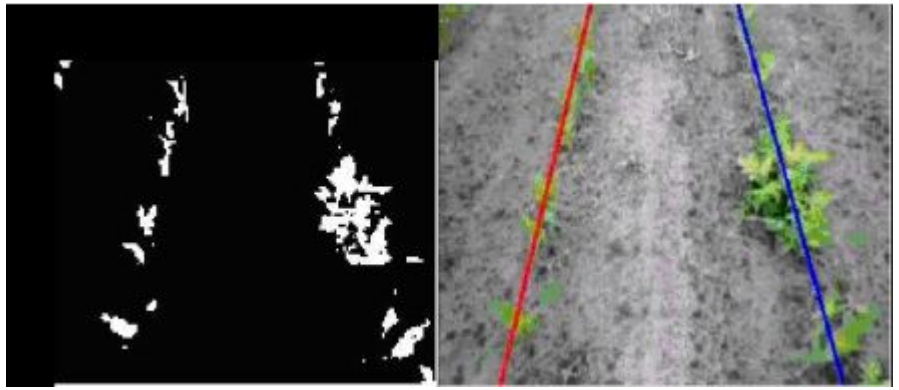
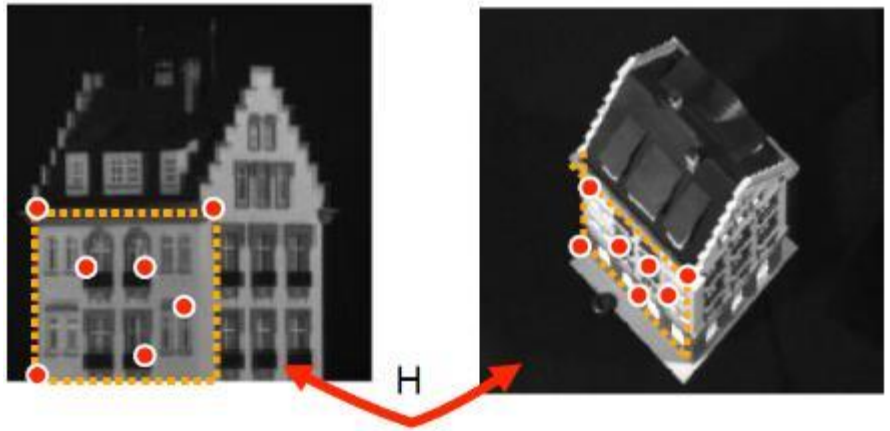
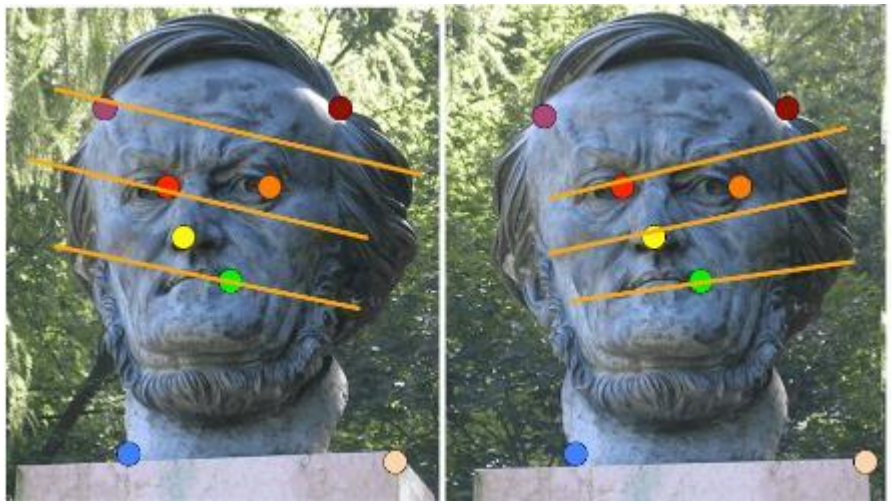


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors







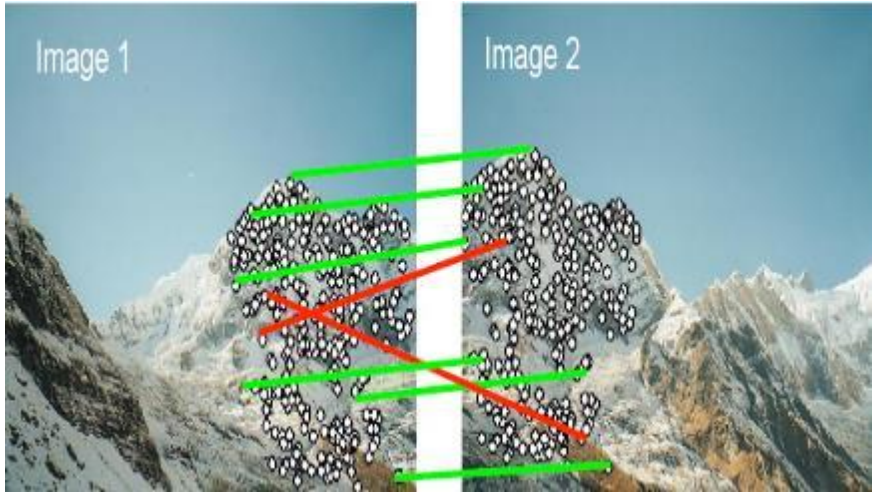


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

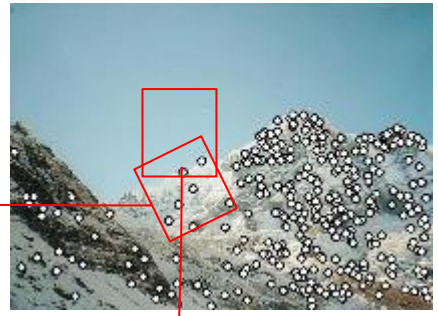
Keypoint detection

1) **Detection:** Identify keypoints or features

2) **Description:** Extract a vector or descriptor around each keypoint

3) **Matching:** Find corresondences between each of the views

4) **Readjust:** Mix images



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

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

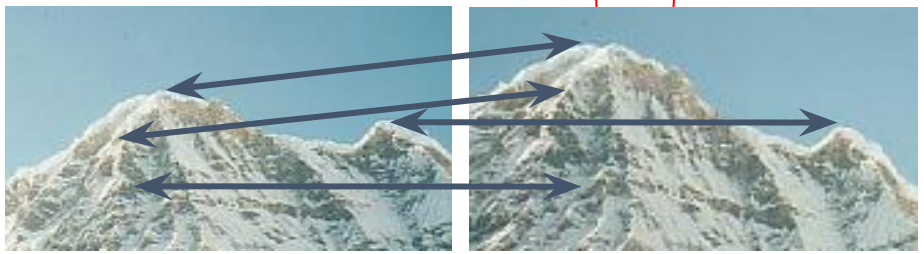


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

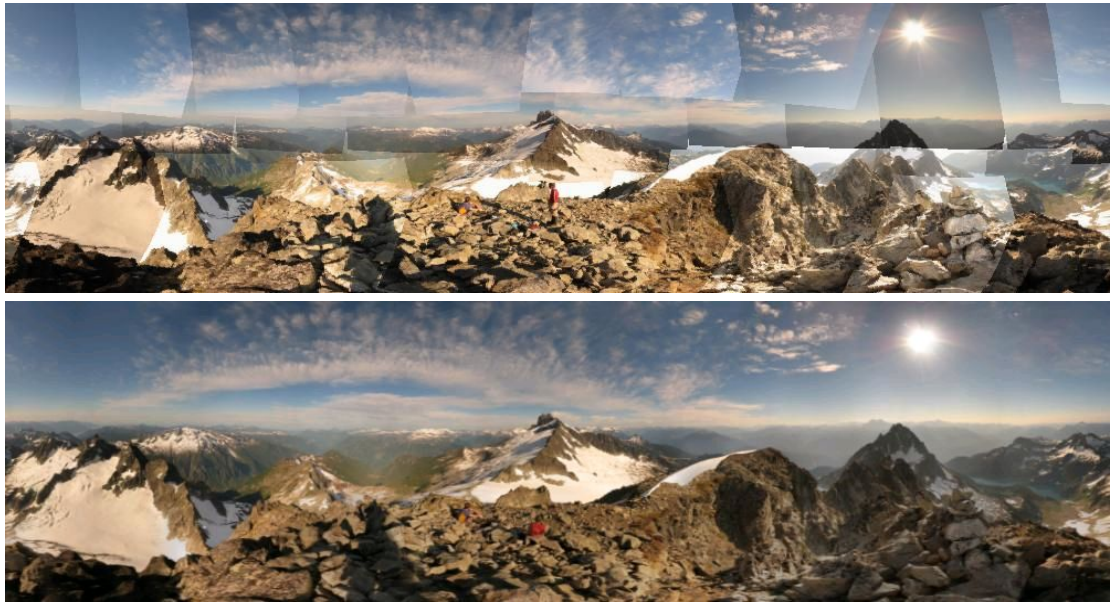


Image Analysis for Computer vision

Keypoint Detection + Local Descriptorsc

Several detectors have been proposed through the years

Hessian & Harris	[Beaudet '78], [Harris '88]
Laplacian, DoG	[Lindeberg '98], [Lowe 1999]
Harris-/Hessian-Laplace	[Mikolajczyk & Schmid '01]
Harris-/Hessian-Affine	[Mikolajczyk & Schmid '04]
EBR and IBR	[Tuytelaars & Van Gool '04]
MSER	[Matas '02]
Salient Regions	[Kadir & Brady '01]

Tuytelaars, T, and Mikolajczyk, **Local Invariant Feature Detection: A Survey**, Foundations and Trends in Computer Vision, Vol. 3. Num. 3, pag. 177-280, 2007.

Several descriprots have been proposed through the years too

Tuytelaars, T, and Mikolajczyk, **Local Invariant Feature Detection: A Survey**, Foundations and Trends in Computer Vision, Vol. 3. Num. 3, pag. 177-280, 2007.

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

- **Repeatibility**

– The same features should be found in various images despite photometric or geometric transformations

Invariant to traslation, rotation, etc

- **Prominency (saliency)**

– Each “feature” should be found within “regions of interest” of the image

- **Localization**

– Each feature occupies a “relatively small área” of the image

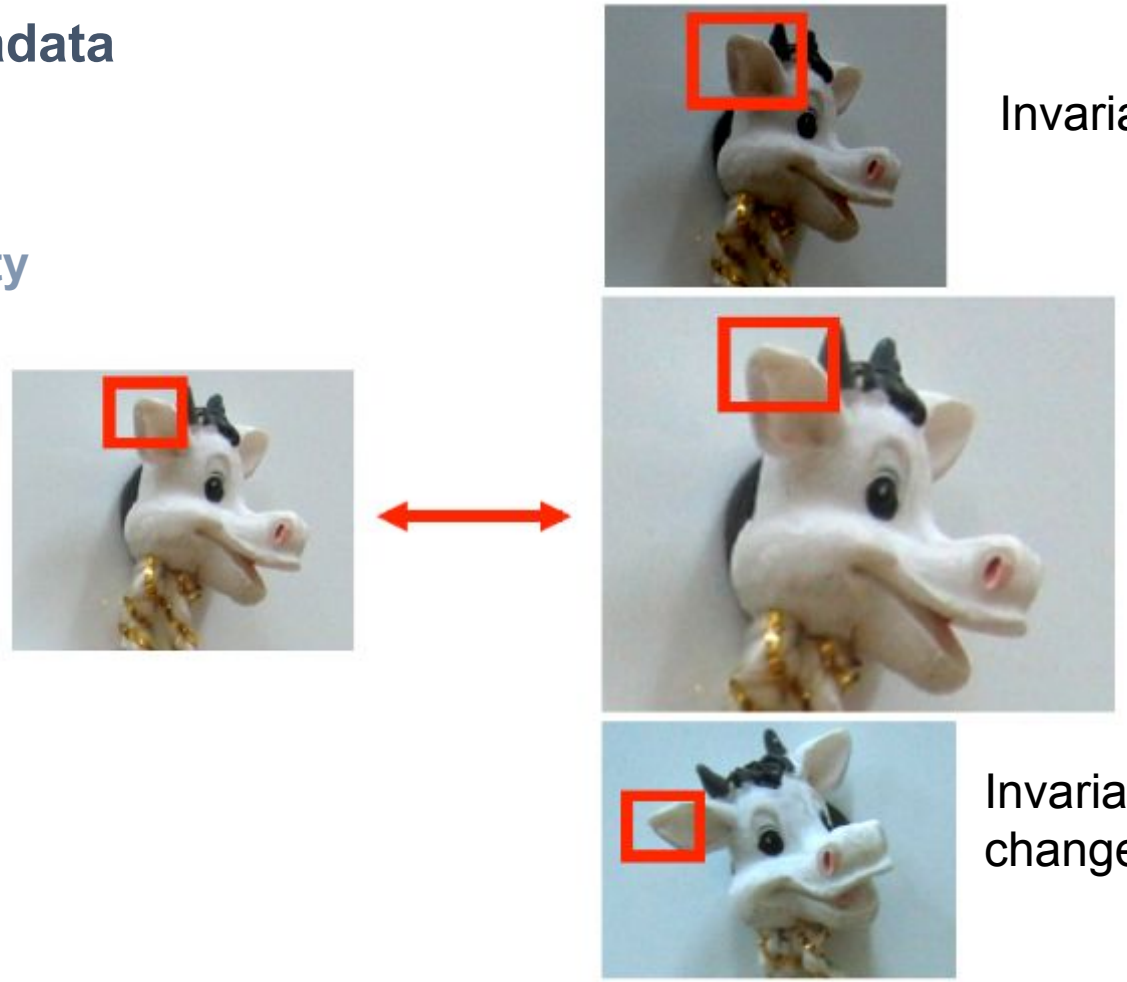
Desiradata

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Desiradata

Repeatibility



Invariance to **ilumination**

Invariance to **scale changes**

Invariancie to **position/pose** changes: **rotation / affine**

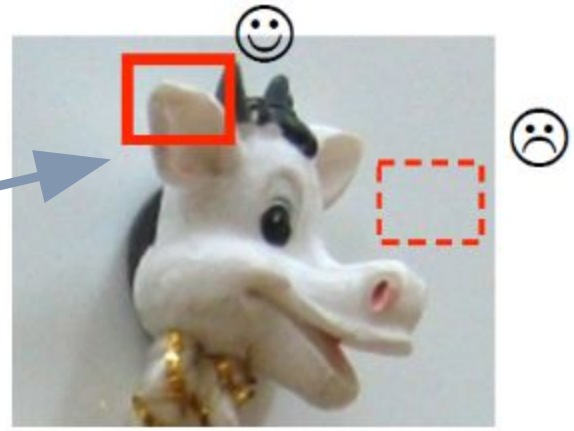
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Desiradata

Prominency (saliency)

Each “feature” should be found within “regions of interest” of the image



Localización

Each feature occupies a “relatively small área” of the image

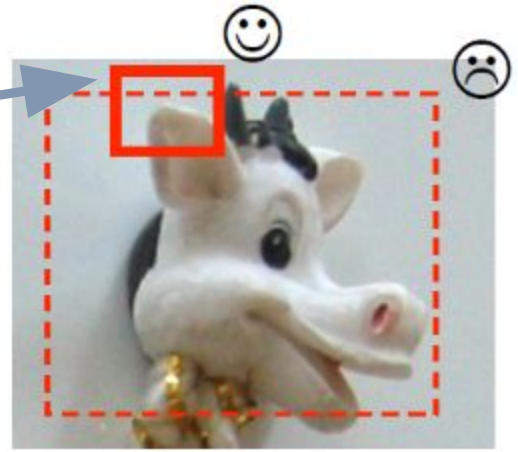


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Illumination	Scale	Rotation	Affine (viewpoint)	Perspective

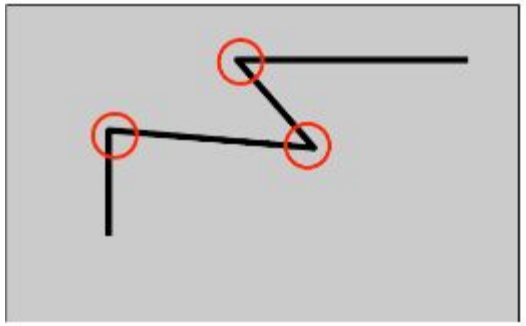
Image Analysis for Computer vision

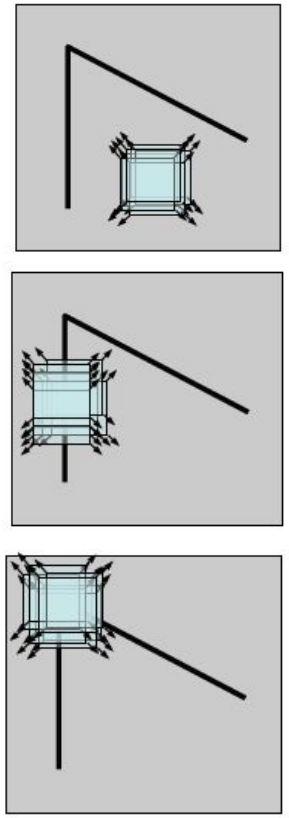
Keypoint Detection + Local Descriptors

Harris corner detector

Basic idea:

Explore changes in intensity within a windo,w varying the location the window





"flat" region: no changes in any direction

"edge": no changes in the line direction

"corner": high variations or significative in all directions

C.Harris and M.Stephens. **"A Combined Corner and Edge Detector."** *Proceedings of the 4th Alvey Vision Conference*: pages 147--151.

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

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

α: constant (0.04 a 0.06)

$$det\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = ad - bc$$

$$trace\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = a + d$$

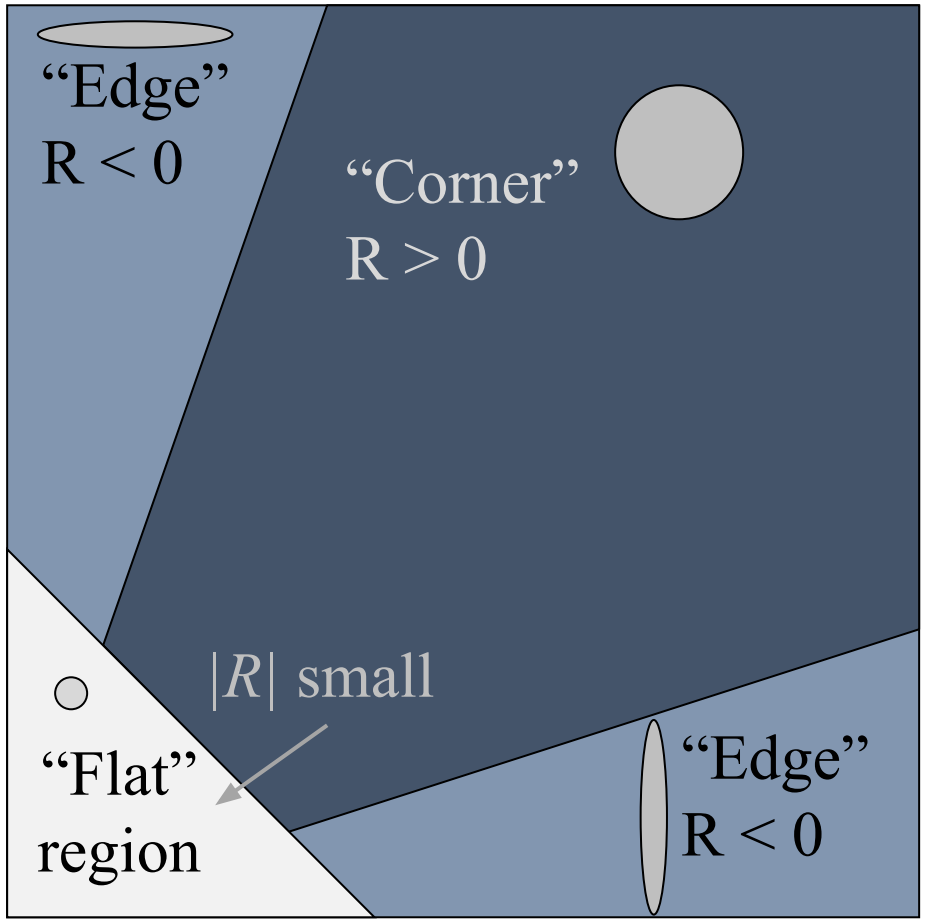
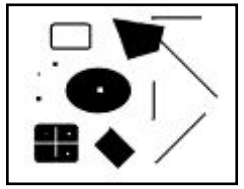


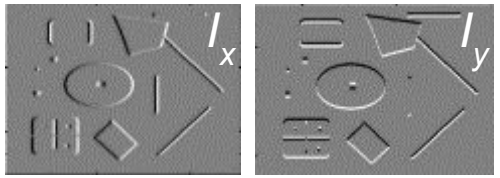
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

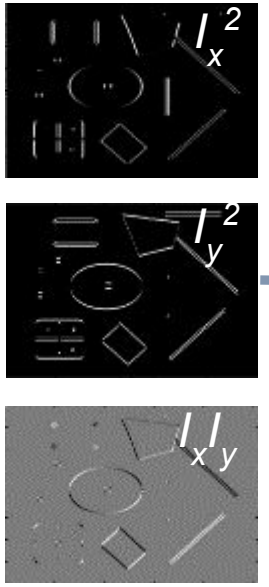
Harris corner detector



1. Image derivatives



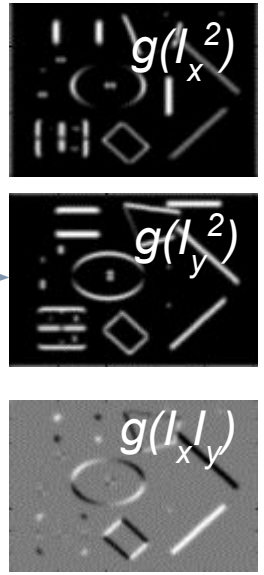
2. Squares of the derivatives



$$\det M = \lambda_1 \lambda_2$$

$$\operatorname{trace} M = \lambda_1 + \lambda_2$$

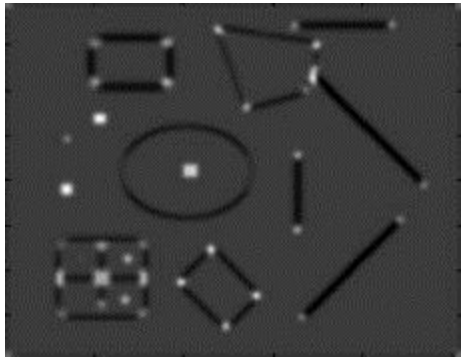
3. Gaussian filter $g(\sigma_I)$



4. "Cornerness" function

$$har = \det[M(\sigma_I, \sigma_D)] - \alpha[\operatorname{trace}(M(\sigma_I, \sigma_D))^2]$$

$$= g(I_x^2)g(I_y^2) - [g(I_x I_y)]^2 - \alpha[g(I_x^2) + g(I_y^2)]^2$$



5. Non-maxima supression (NMS)

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Harris corner detector → Input Image

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Harris corner detector → Cornerness R

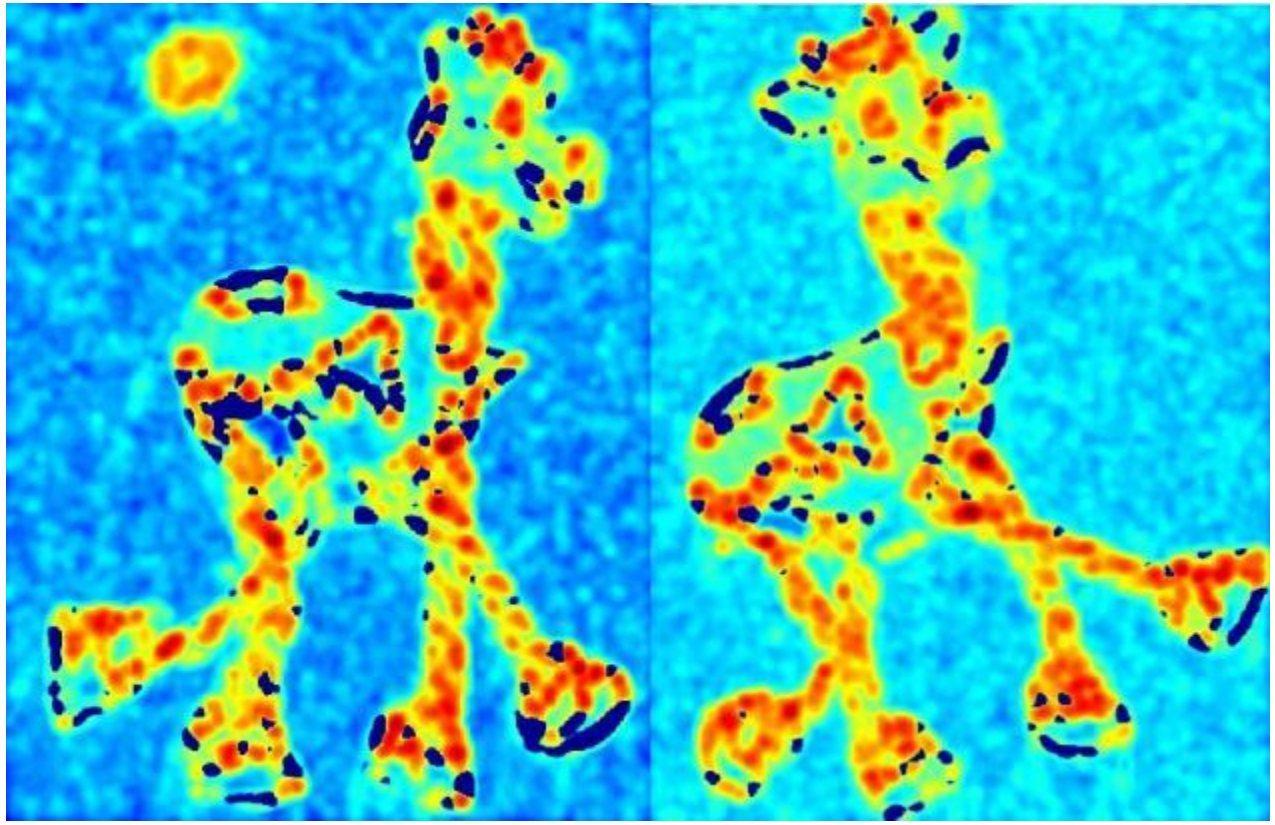


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Harris corner detector → Points with R > threshold

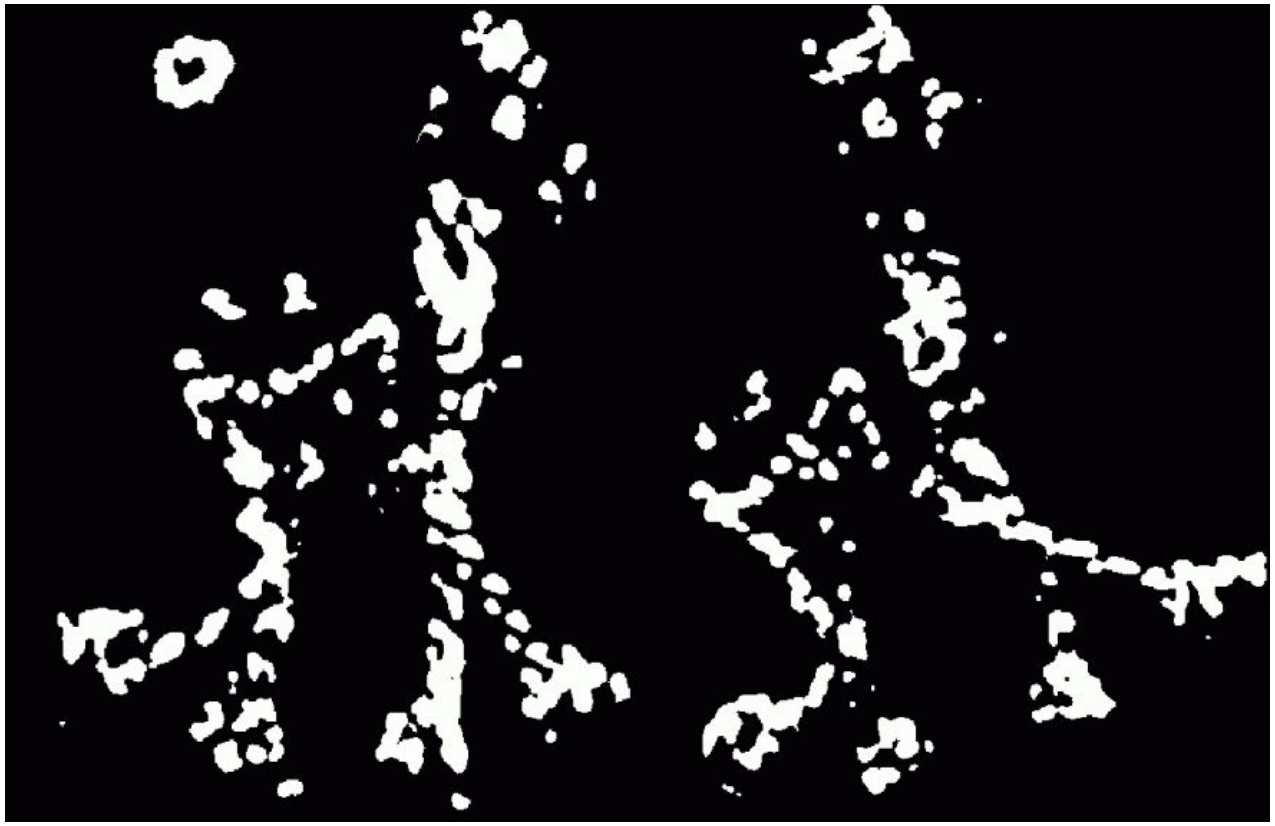


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Harris corner detector → Apply NMS

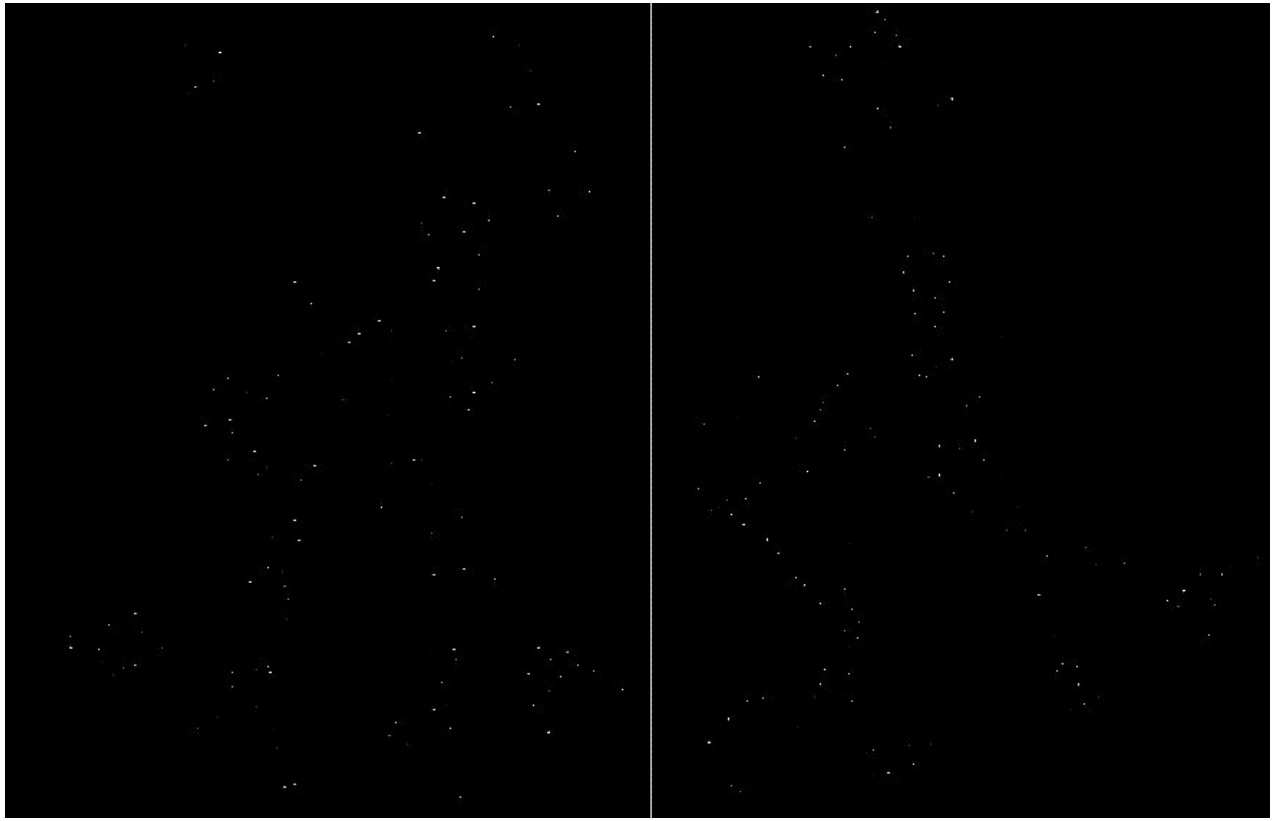


Image Analysis for Computer vision

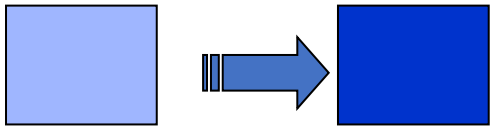
Keypoint Detection + Local Descriptors

Harris corner detector → Super/imposed points

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

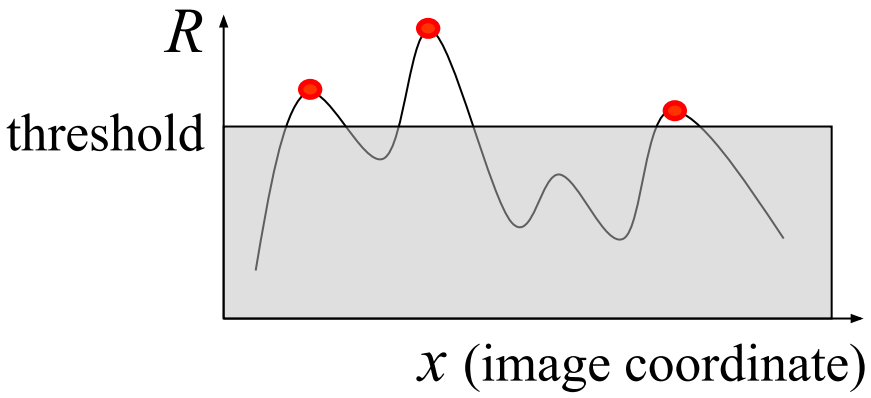
Harris corner detector → Partially invariant to changes in light **intensity**



$$I \rightarrow a\,I + b$$

Derivatives are used, this □ **Invariance to shifts in intensity**

I* → *I* + *b *or in scales of intensity*: ***I* → *a I***



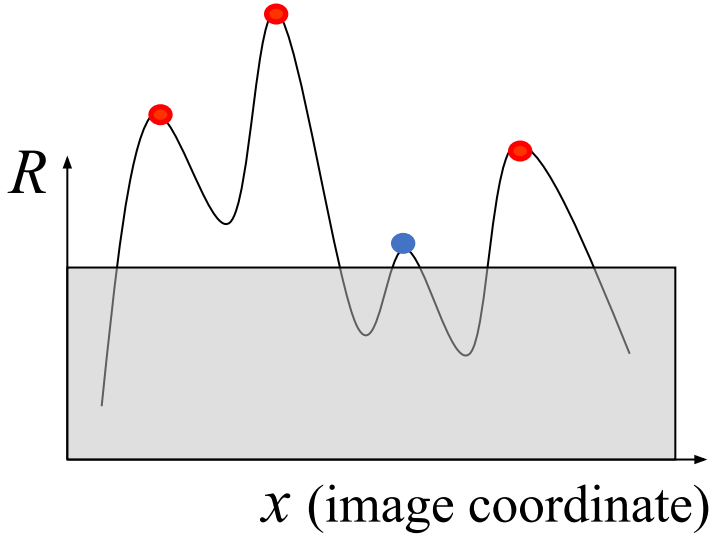


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Harris corner detector

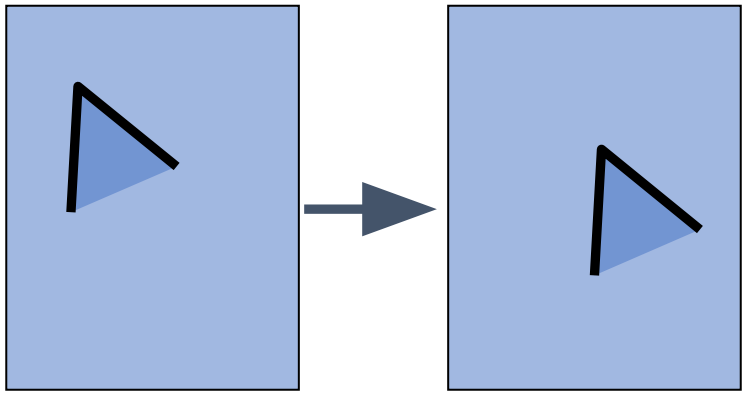


Image Translation

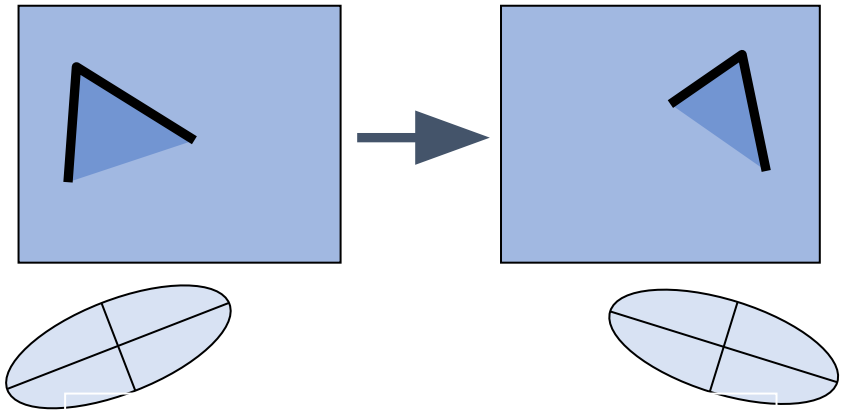


Image Rotation

Derivatives and widndow kernerls are “shift-invariant”

La location of a corner is covariant w.r.t. traslation

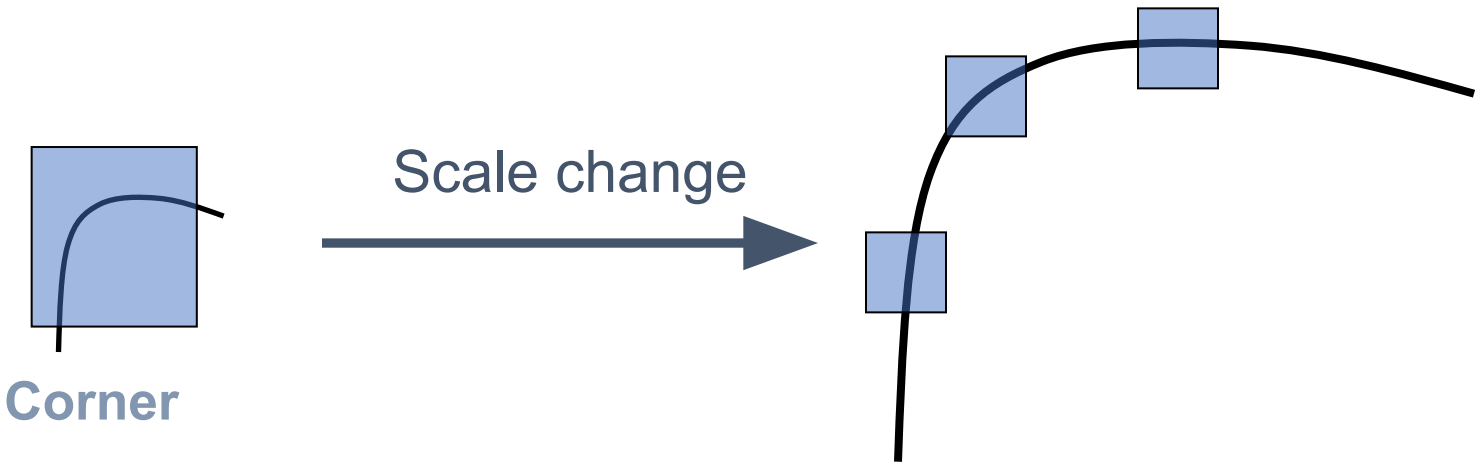
The slops of moment M rotates but its shape (i.e. eigenvalues) are connstant

La location of a corner is covariant w.r.t. to rotation

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Harris corner detector



Every corner or **keypoint** us classified as an **edge**

The location of a corner is not invariant to scale!

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Requirements

- Repeatibility
- Prominence (saliency)
- Localization

But also

Invariance rotation, traslation and scale changes (by change of **perspective or depth**)

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

- Consider regions of different size
- Select regions to subtend the same content
- How to choose the size of the region independently

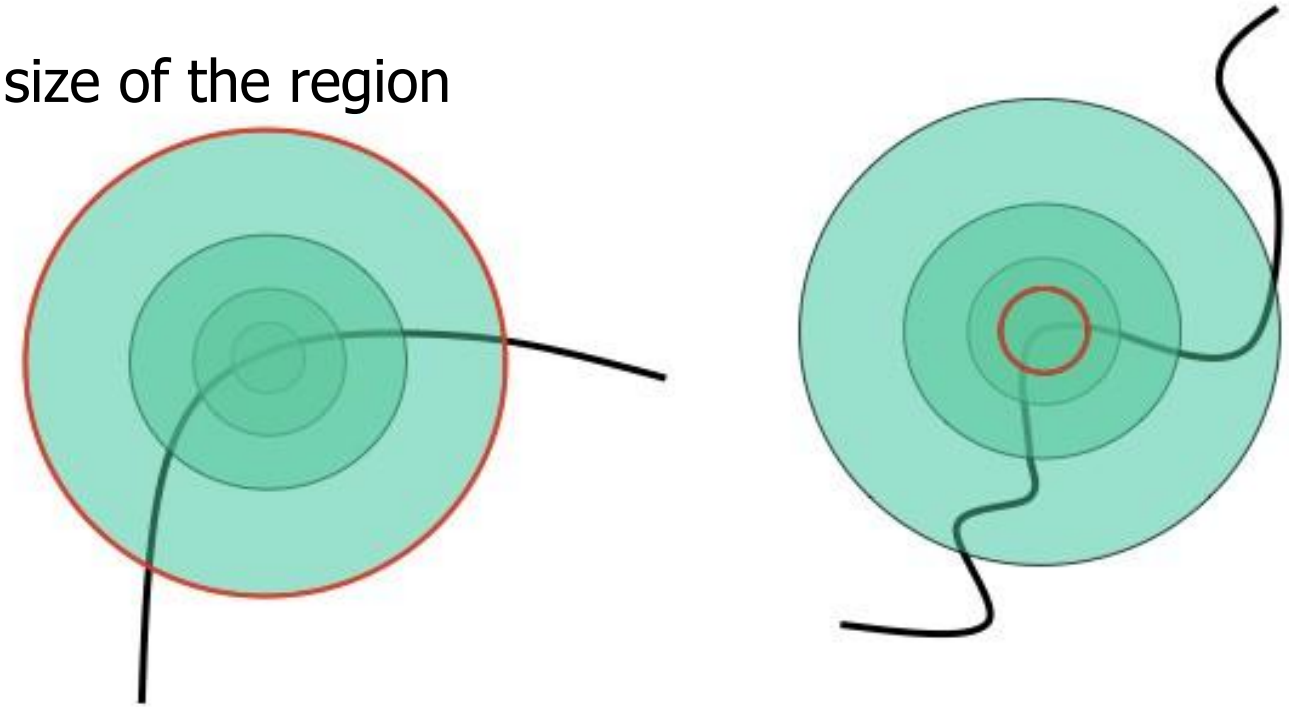


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Solution: blob detectors

T. Lindeberg (1998). "Feature detection with automatic scale selection." *International Journal of Computer Vision* **30** (2): pp 77--116.

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

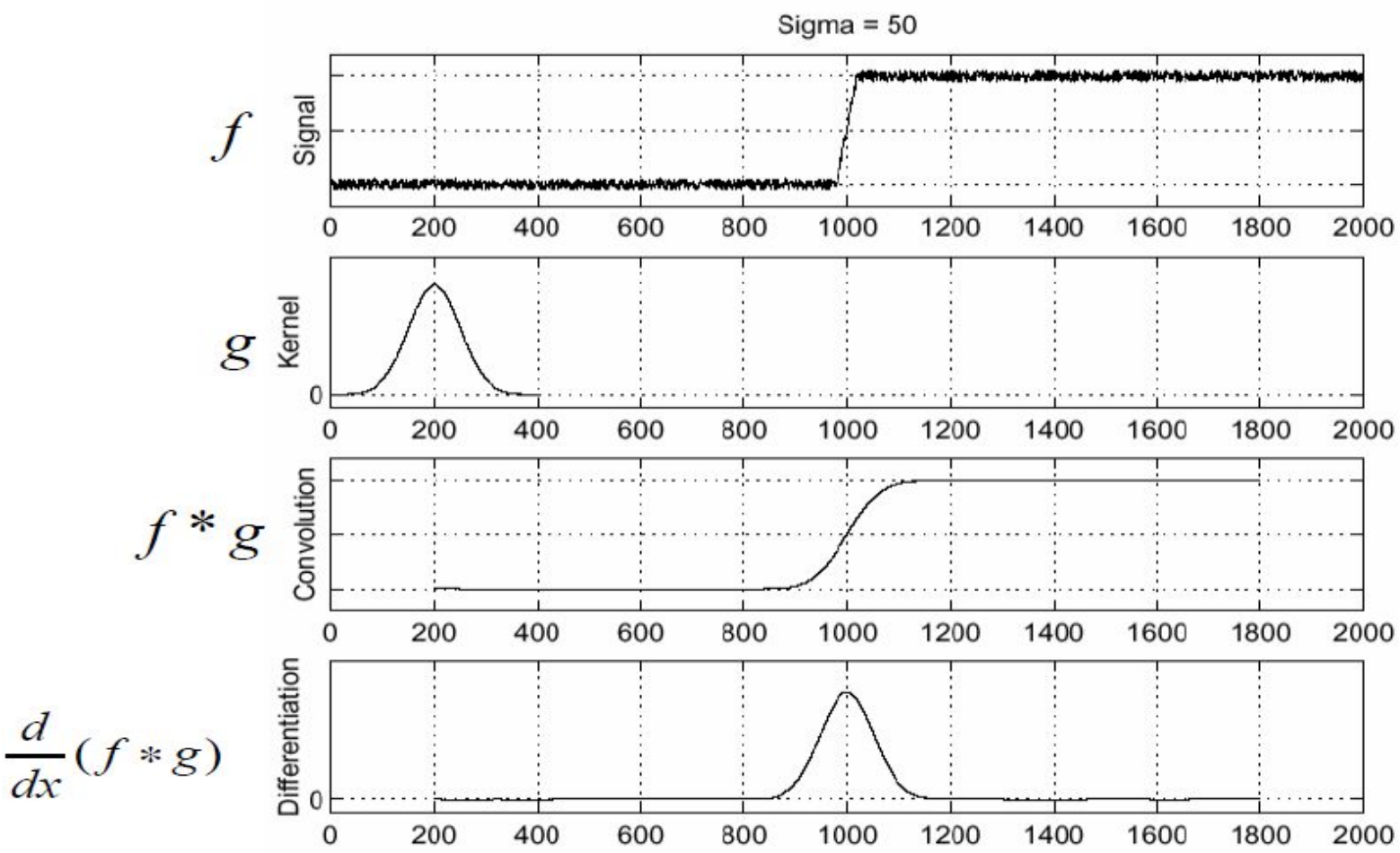
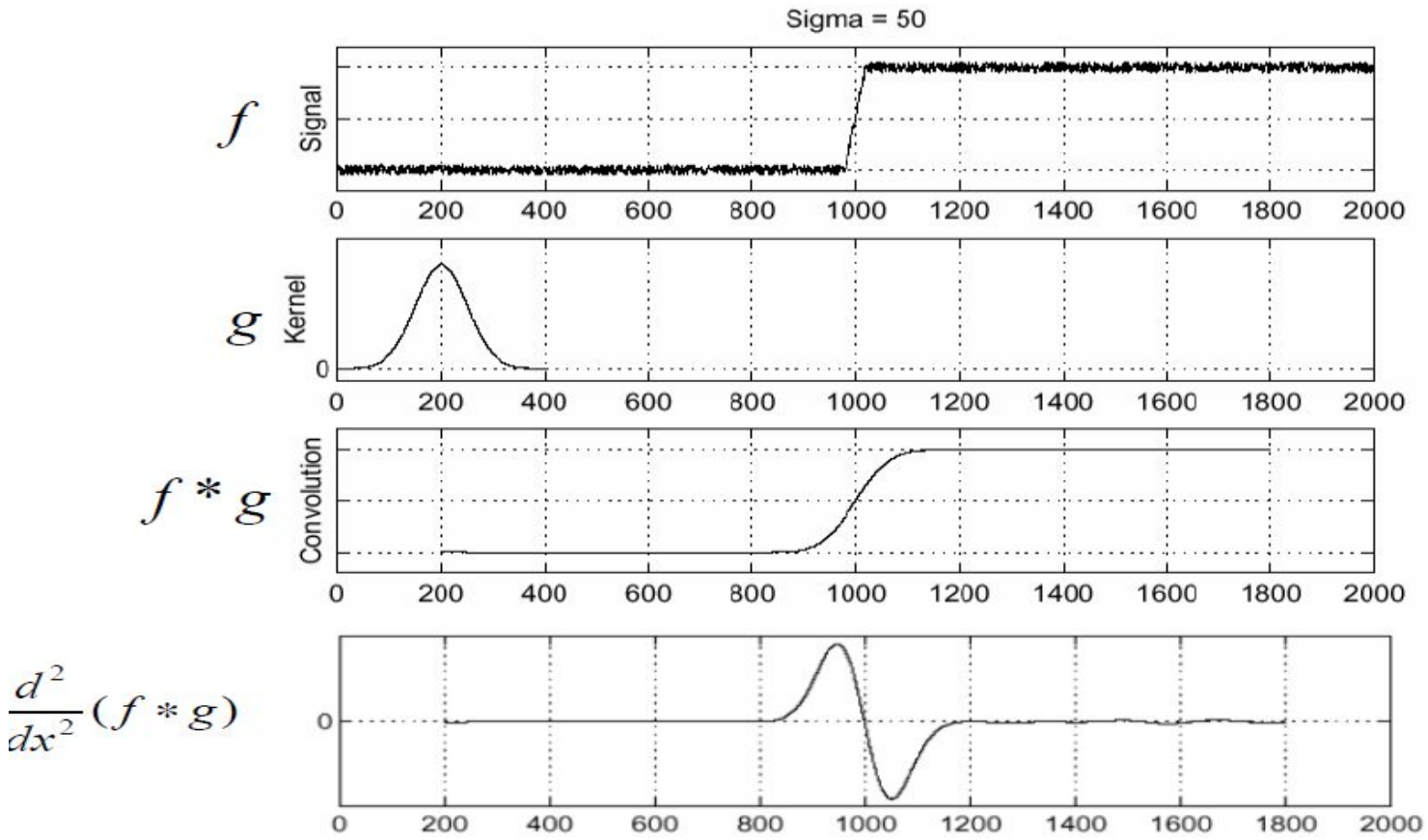


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

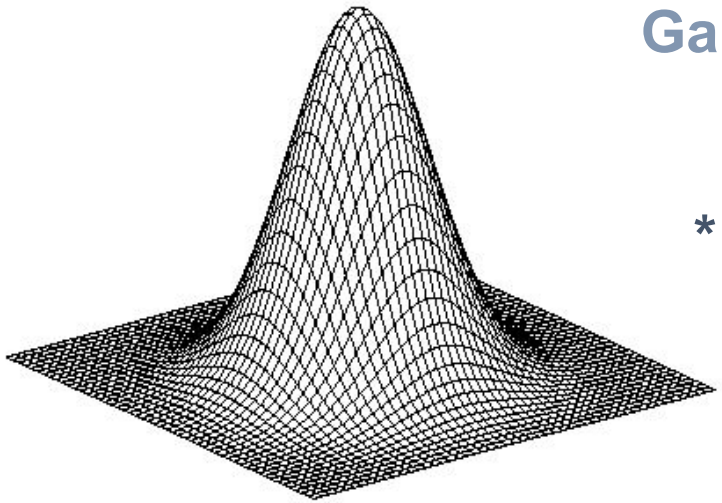


$f * \frac{d^2}{dx^2} g$ = “second derivative of Gaussian” filter = **Laplacian of the Gaussian**

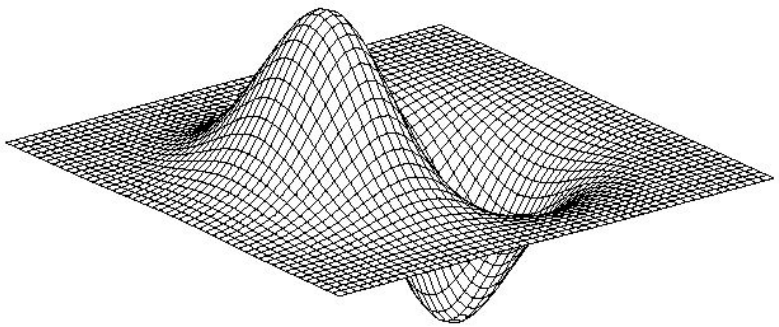
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Derivative of Gaussian filter



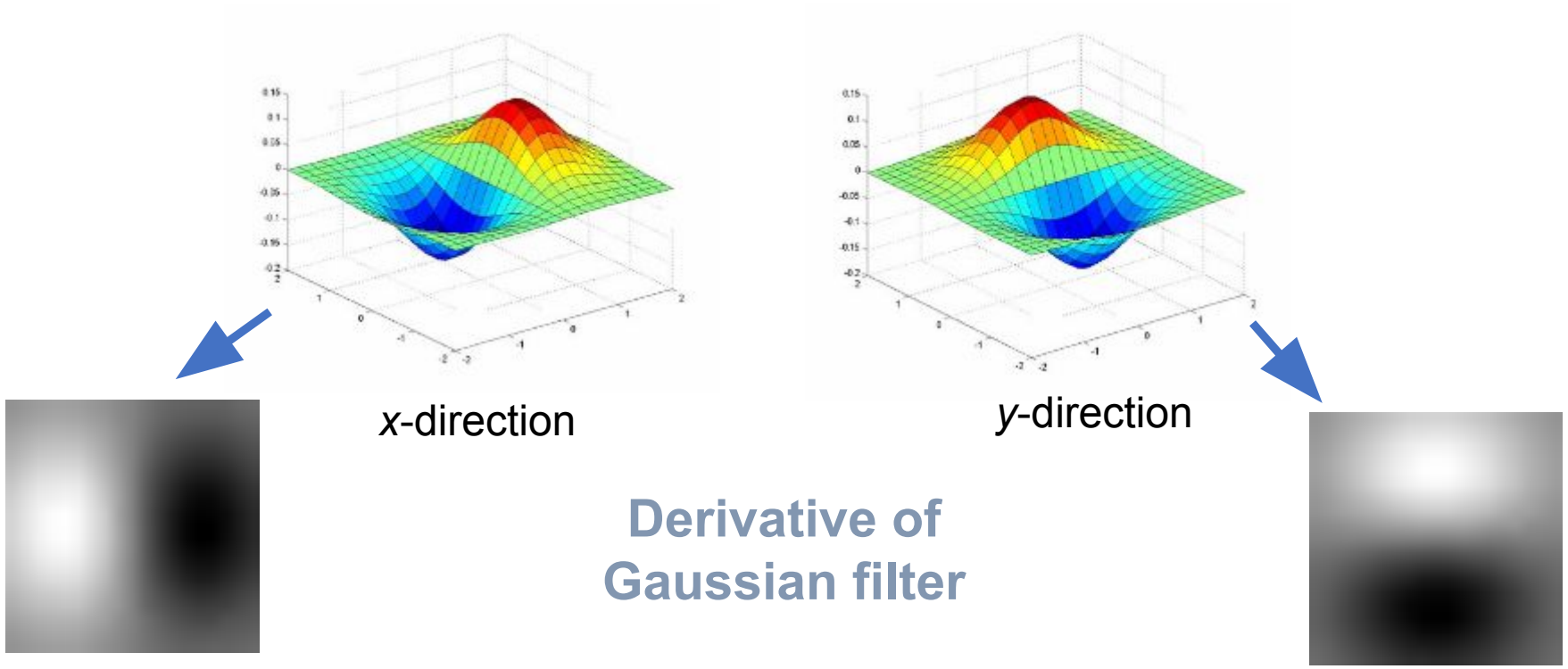
* [1 -1] =



J. Canny, A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



J. Canny, ***A Computational Approach To Edge Detection***, IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.

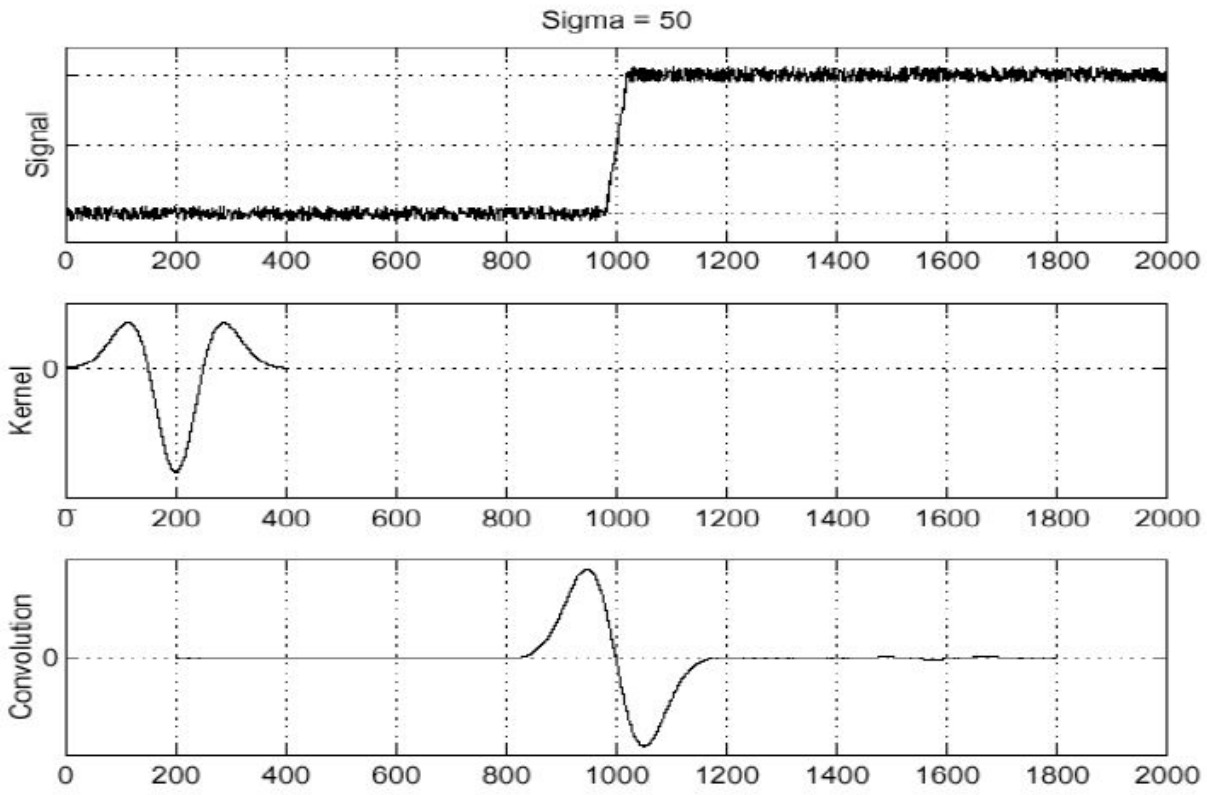
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

f

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

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



Edge

Laplacian

Edge
zero crossing

Edge = zero crossing of the second derivative

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Podemos el Laplaciano para encontrar un “blob”(function rectangular)?

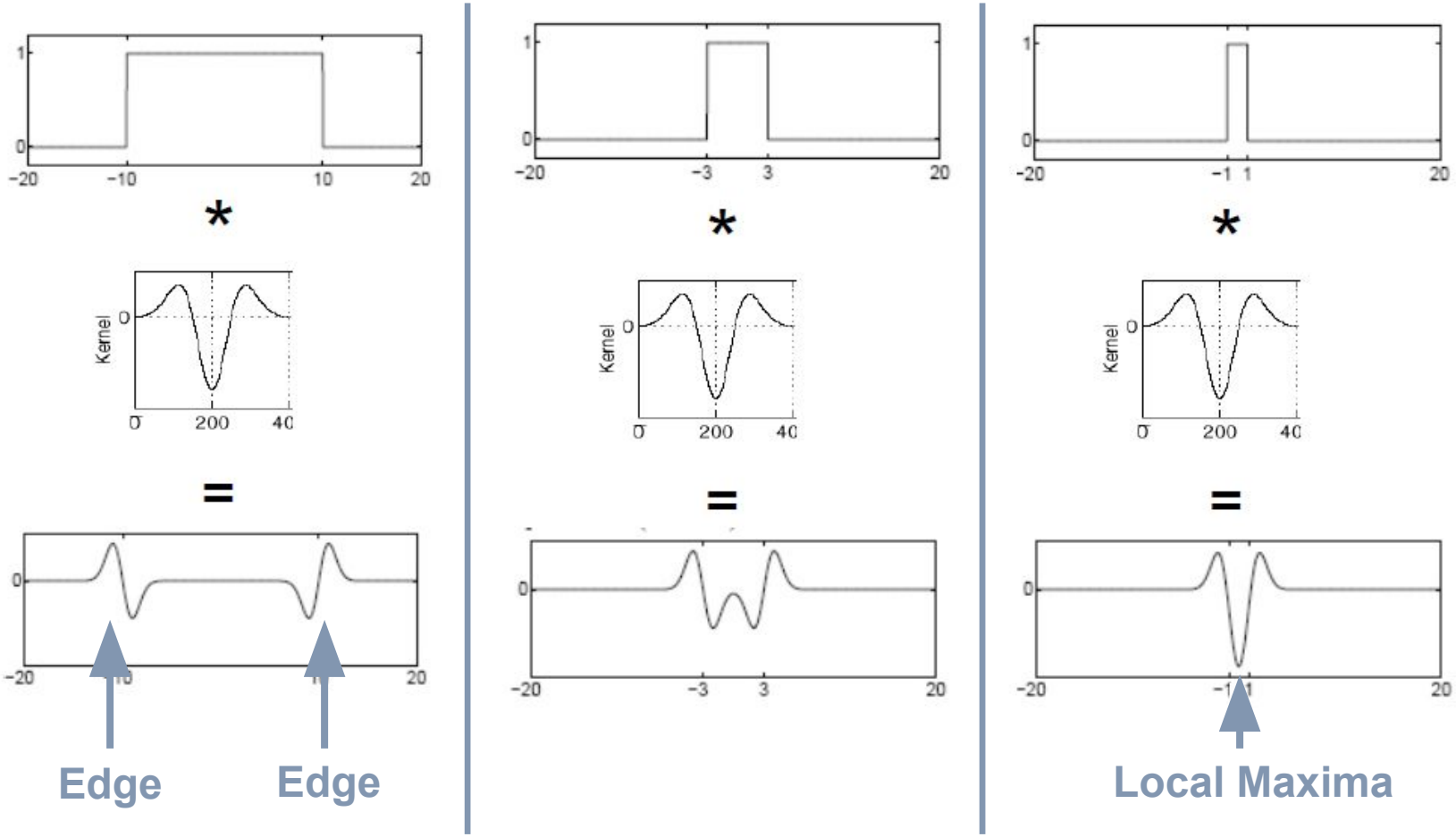
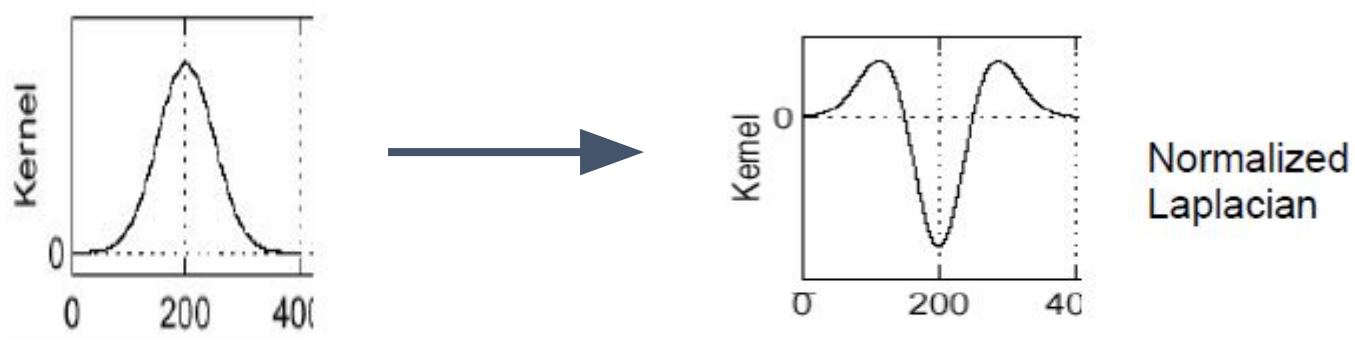


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

What happens as we change the width of the function? □ **Scale selection**

Idea: Convolve the signal with Laplacians at different scales and look for the máxima, how we do that? **Incrementing σ**



$$g(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

$$\sigma^2 \frac{d^2}{dx^2} g_n$$

To maintain the energy response constant, the kernel must be multiplied by σ □ **Scale normalization**

- Laplacian is the second derivative of the Gaussian, it must be multiplied by σ^2

Image Analysis for Computer vision

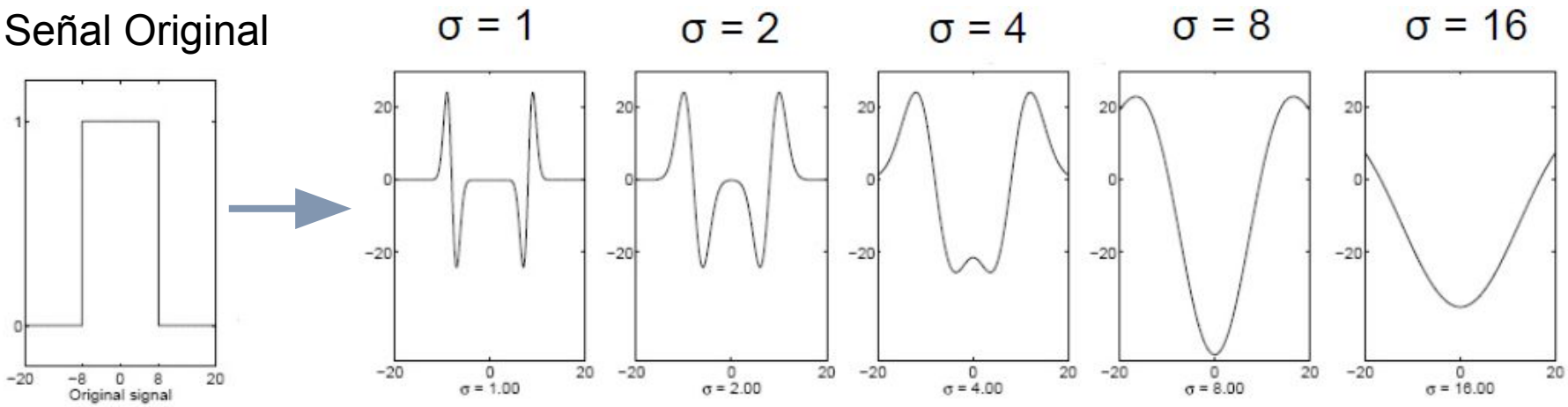
Keypoint Detection + Local Descriptors

Concept: characteristic scale

Local maxima

Response of the **"Scale-normalized Laplacian"**

Señal Original



The **characteristic scale** is that the produces a max in the responses of the Laplacian.
This procedure allows us to

1) Detect un **blob**
2) Estimate the size of a **blob**!

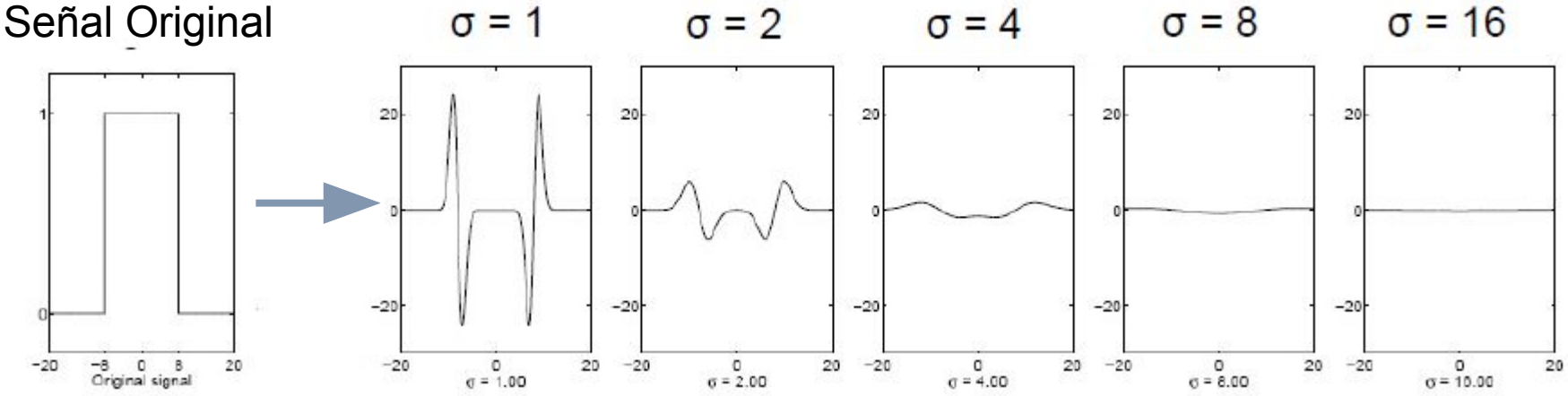
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Concept: Charactestisc scale

What happens when the Laplacian is not normalized?

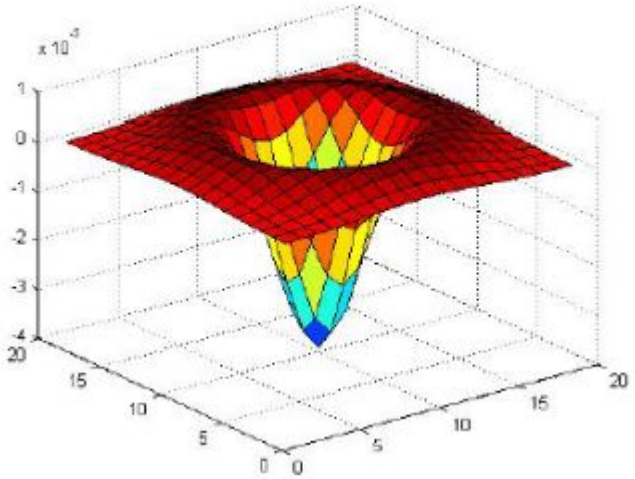
Señal Original

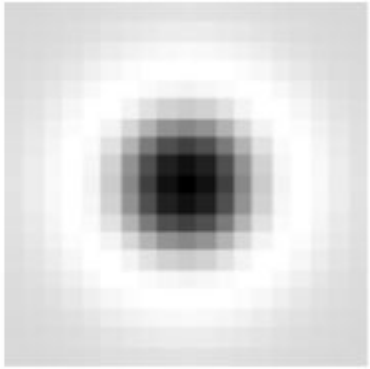


No local maxima

Blob detection in 2D

Laplacian of Gaussian: Symetric Circular Operator, it is used to detect blobs in 2D





Normalized in scale →

$$\nabla^2_{\text{norm}} g = \sigma^2 \left(\frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \right) = \sigma^2 (g_{xx} + g_{yy})$$

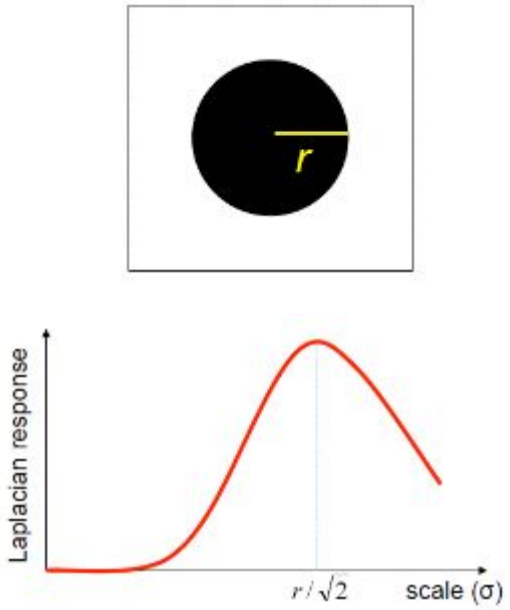
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Blob detection in 2D → **Scale-space blob detector**

For a binary circle of ratio r, the Laplacian attains a maxima at

$\sigma = r / \mathrm{sqrt}(2)$



1. Convolving the image with the Laplacian at different scales, modifying **σ**
2. Find a maxima of the Laplacian response in the space scale

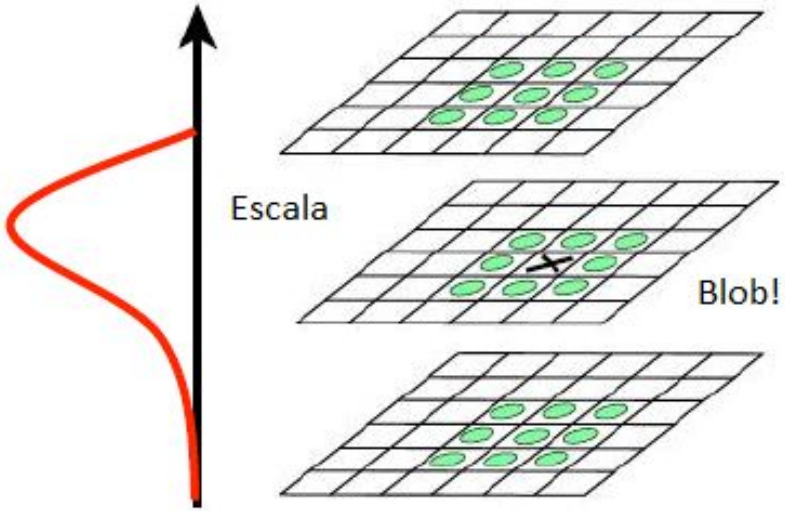


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Blob detection en 2D → **Scale-space blob detector**

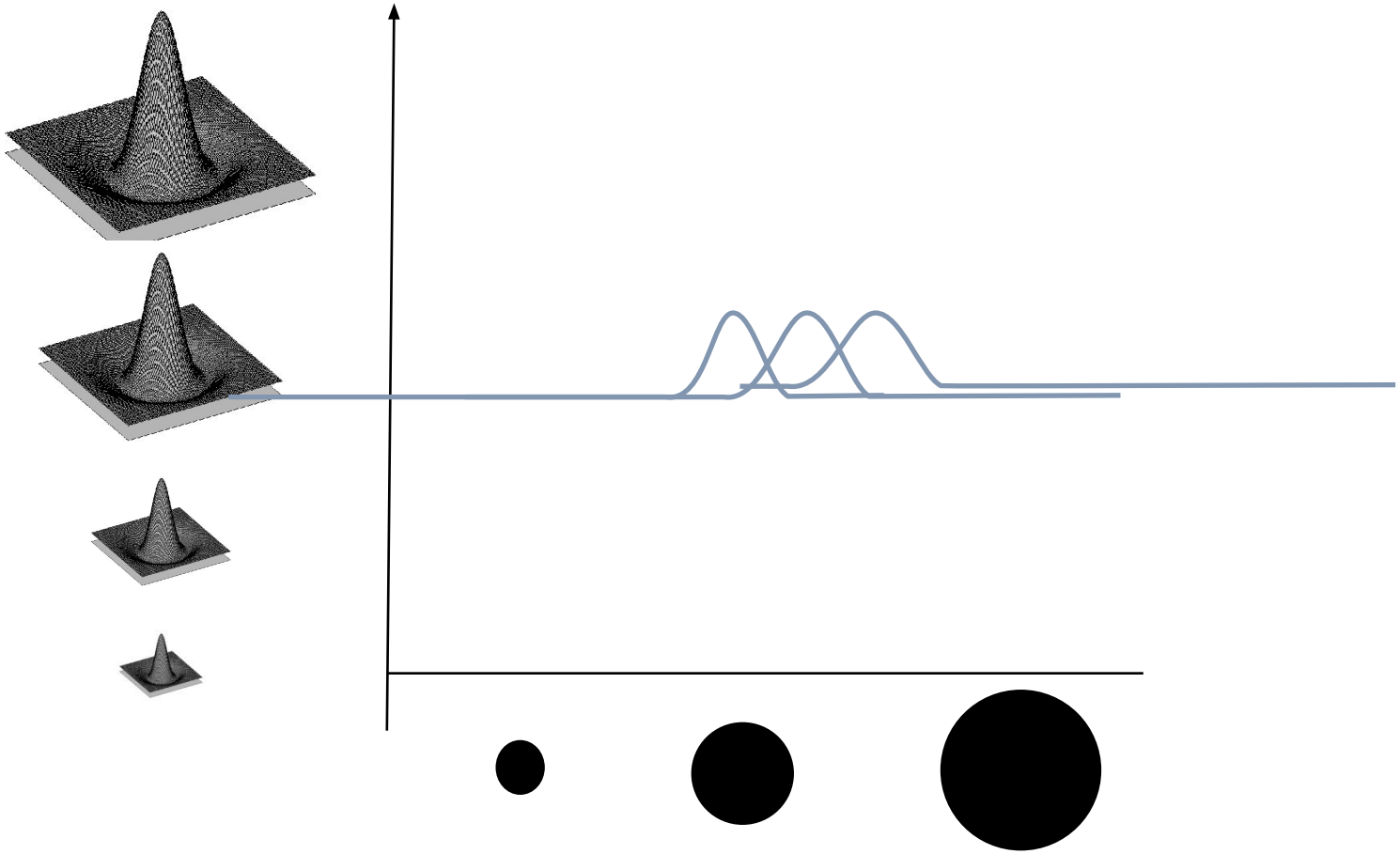
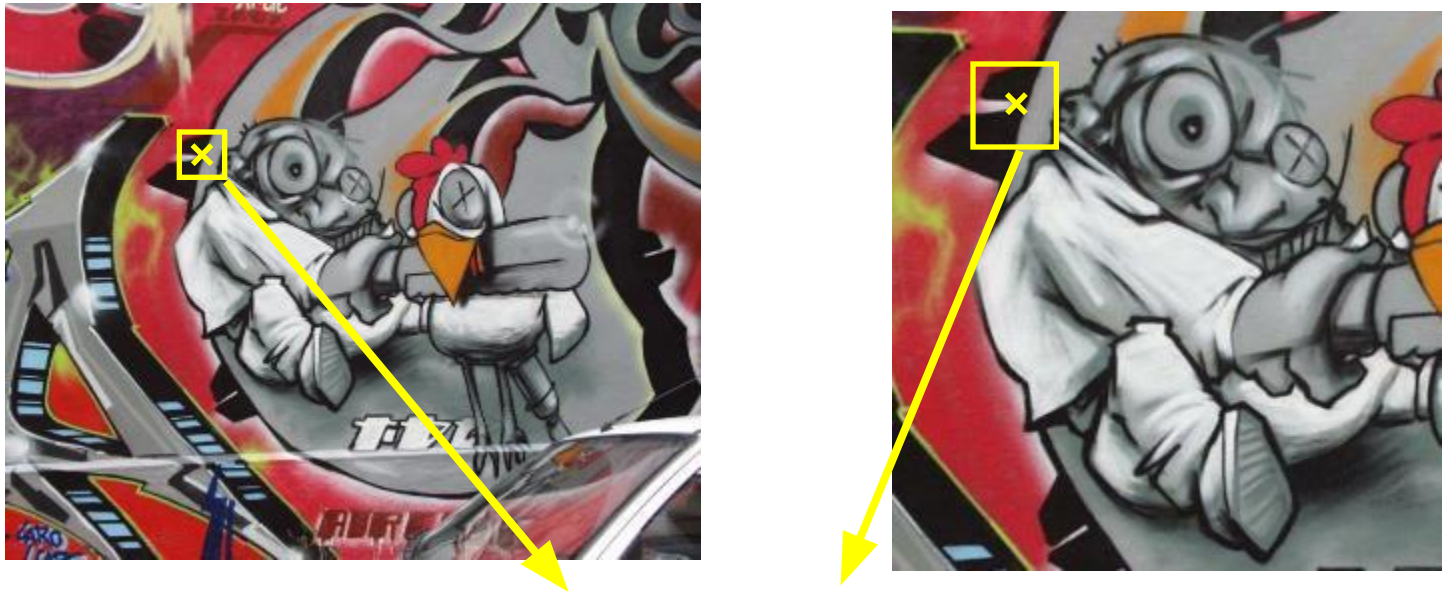


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

How do we find similar patches in two different images?



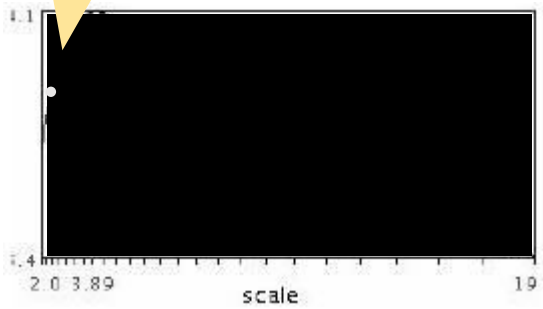
$$f(I_{i_1 \ldots i_m}(x,\sigma)) = f(I_{i_1 \ldots i_m}(x',\sigma'))$$

T. Lindeberg (1998). "Feature detection with automatic scale selection." *International Journal of Computer Vision* **30** (2): pp 77--116.

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

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



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

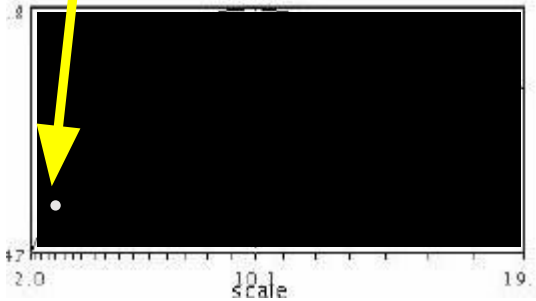
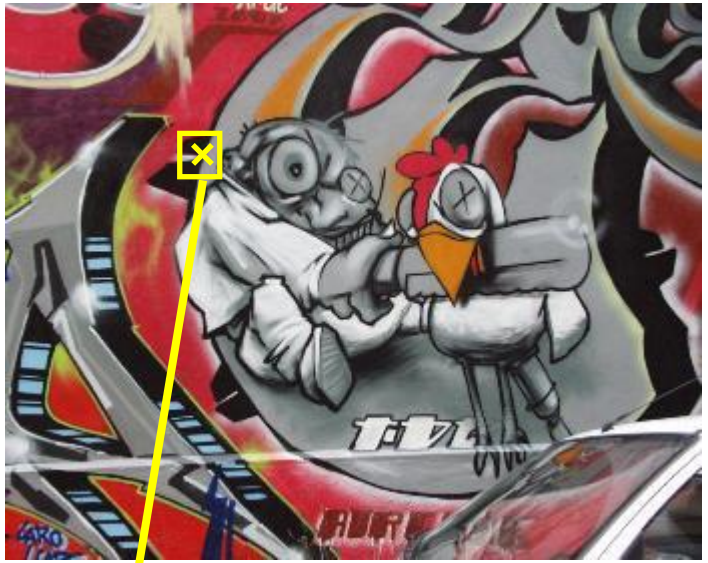
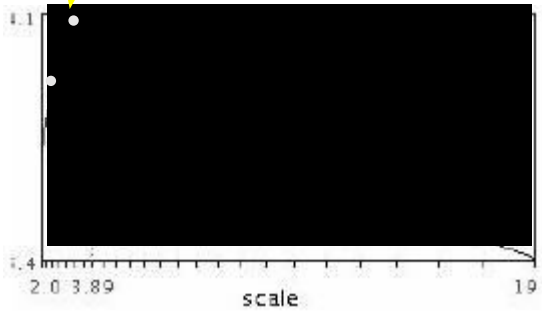


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



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



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

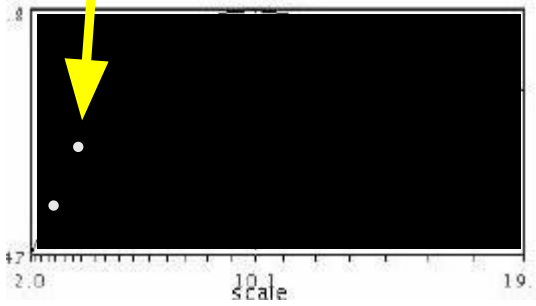
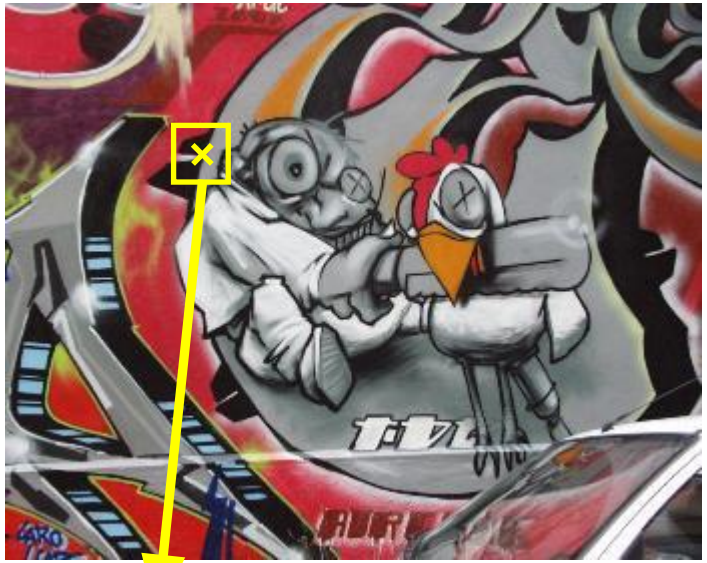
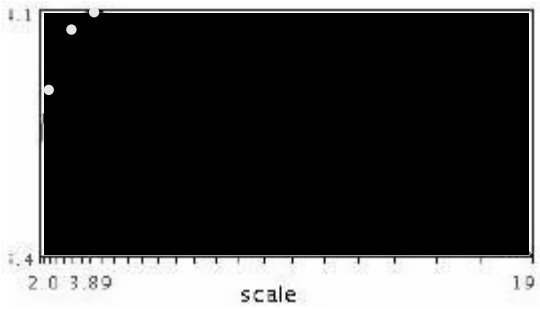


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



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



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

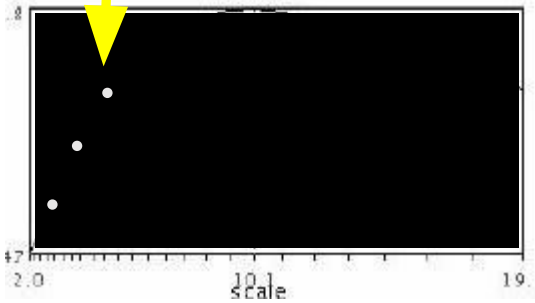
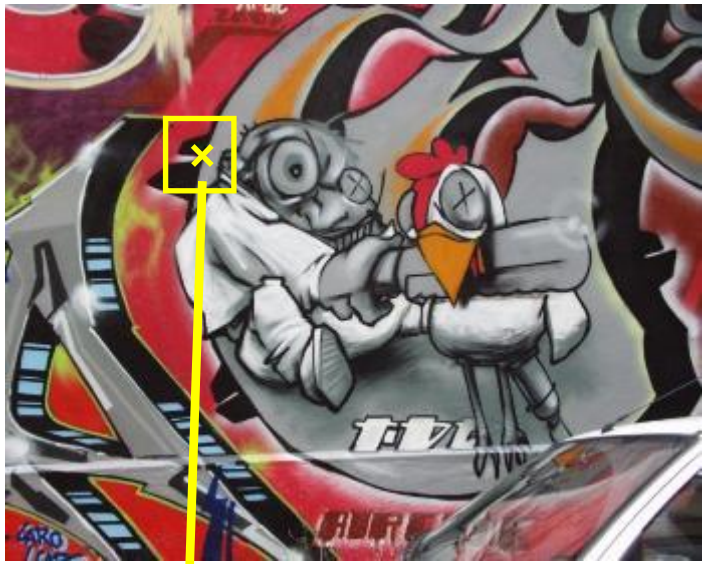
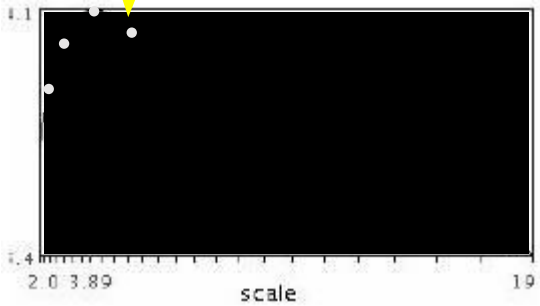


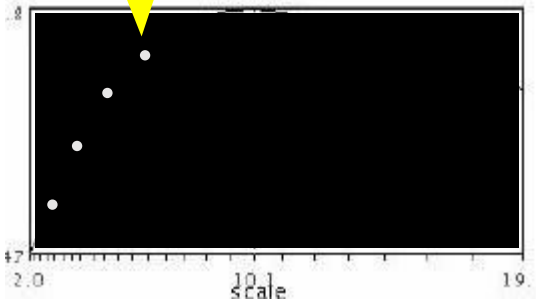
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors





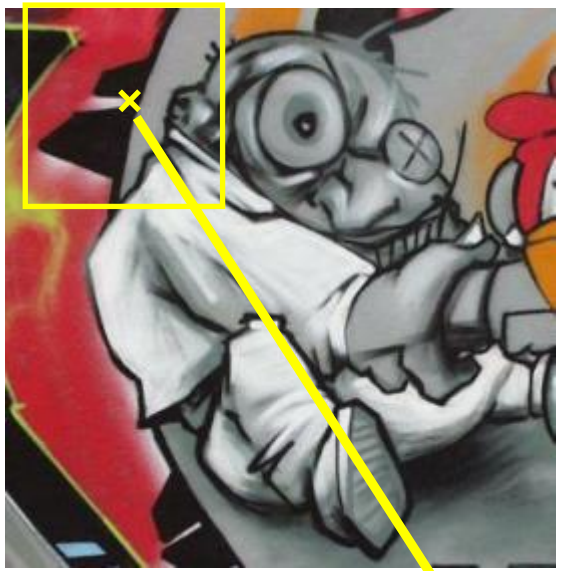
$f(I_{i_1 \ldots i_m}(x,\sigma))$



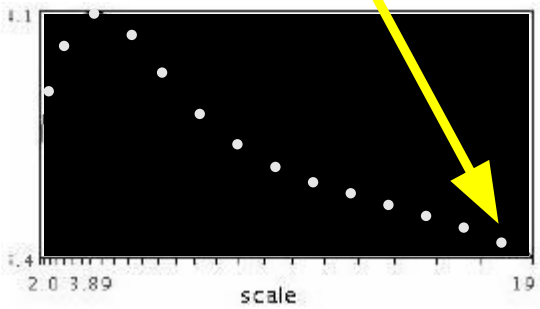
$f(I_{i_1 \ldots i_m}(x',\sigma))$

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



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



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

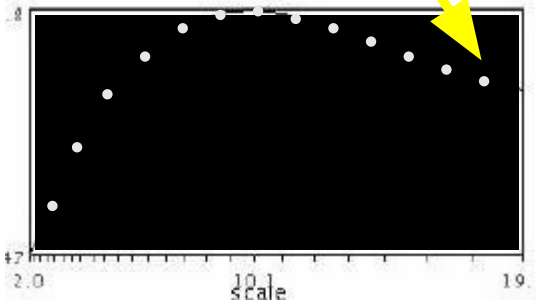
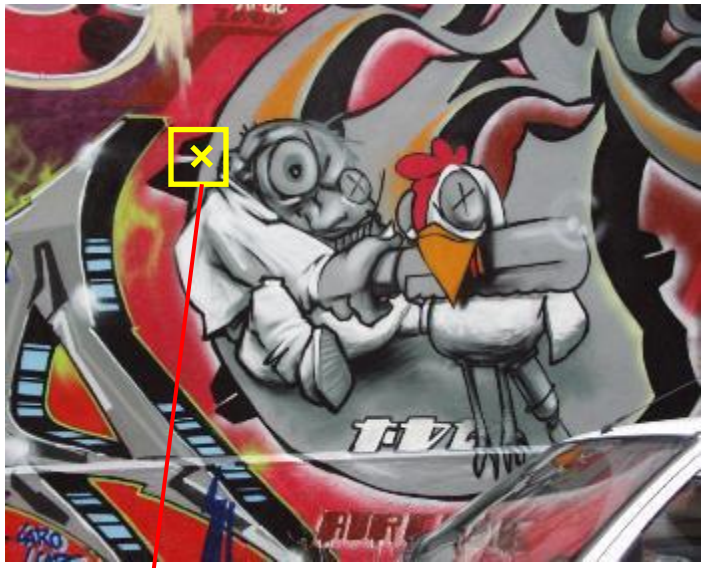
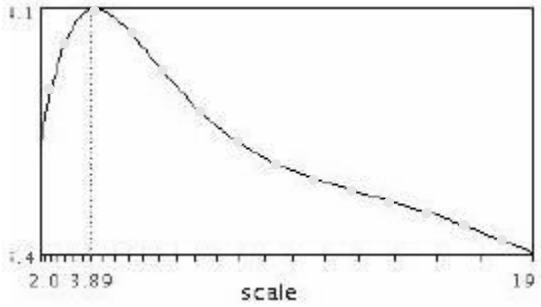


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



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



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

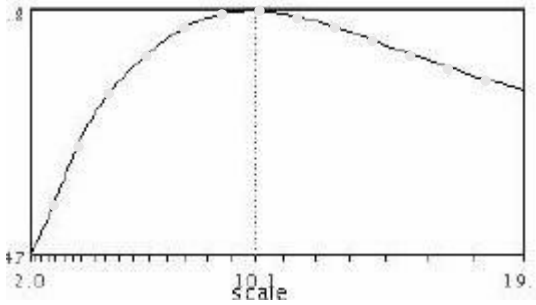
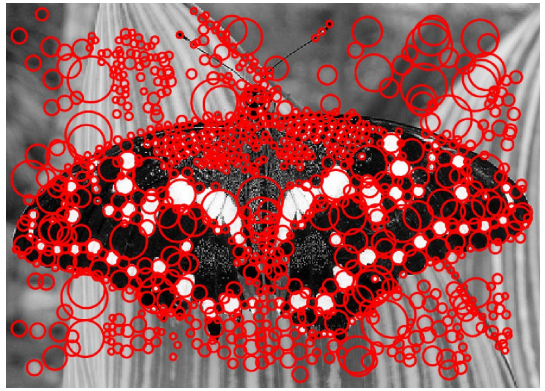


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Blob detection in 2D → **Scale-space blob detector**

sigma = 11.9912



After covolving with various Gaussian kernels of various sizez □

A threshold can be applied to see how many blobs were detected

Image in the rigy shows blobs detected at different scales

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

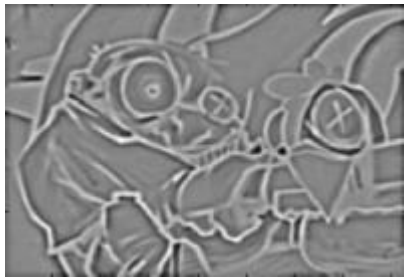
Difference of Gaussians (DoG)

David G. Lowe. "Distinctive image features from scale-invariant keypoints." *IJCV* 60 (2), 04

1st basic idea: Aproximatte the Laplacian with a Difference of Gaussians

-

=



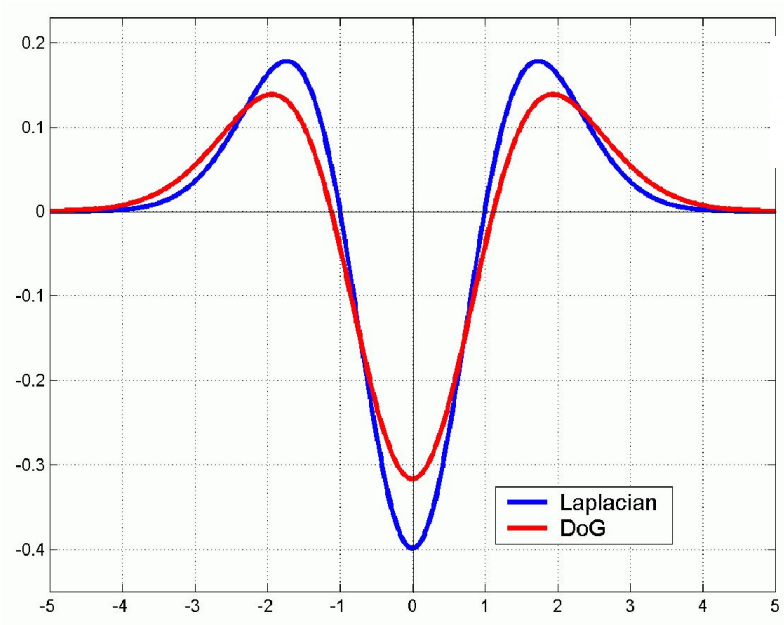
2nd basic idea: Find lines in two images at the fundamental scale, apply a detector to extract keypoints (based on the Hessian matrix)

3rd basic idea: These keypoints are invariant to transformations, use **SIFT descriptor**

Difference of Gaussians (DoG)

David G. **Lowe**. "Distinctive image features from scale-invariant keypoints." *IJCV* 60 (2), 04

Basic Idea: Approximating the Laplacian with Difference of Gaussians



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

Laplacian

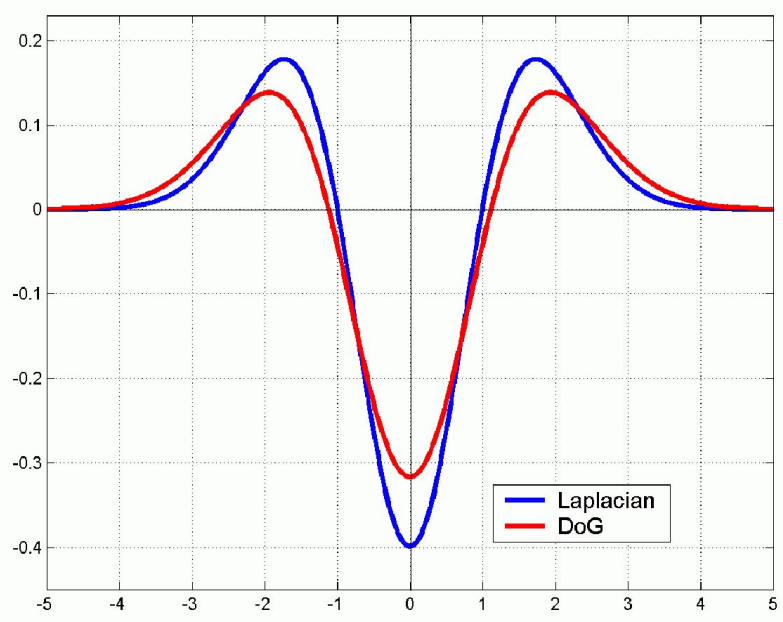
$$DoG = g(x,y,2\sigma) - g(x,y,\sigma)$$

Difference of Gaussians with scales 2σ y σ

Difference of Gaussians (DoG)

David G. **Lowe**. **"Distinctive image features from scale-invariant keypoints."**
IJCV 60 (2), 04

Basic Idea: Approximating the Laplacian with a Difference de Gaussians



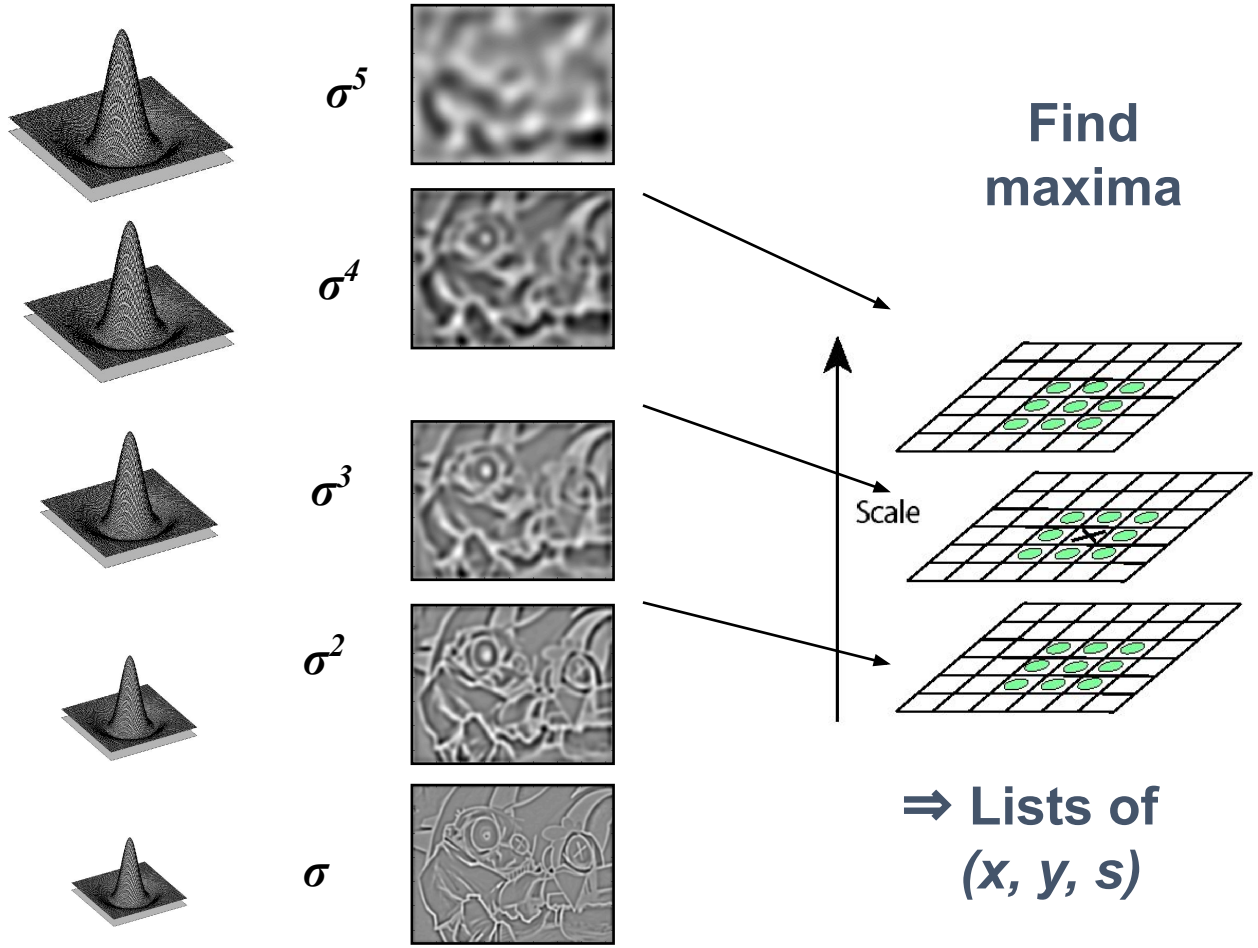
$$DoG = g(x, y, k\sigma) - g(x, y, \sigma)$$

$$\approx (k-1)\sigma^2 \mathbf{L}$$

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Convolution with Gaussians



Find maxima

⇒ Lists of *(x, y, s)*

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

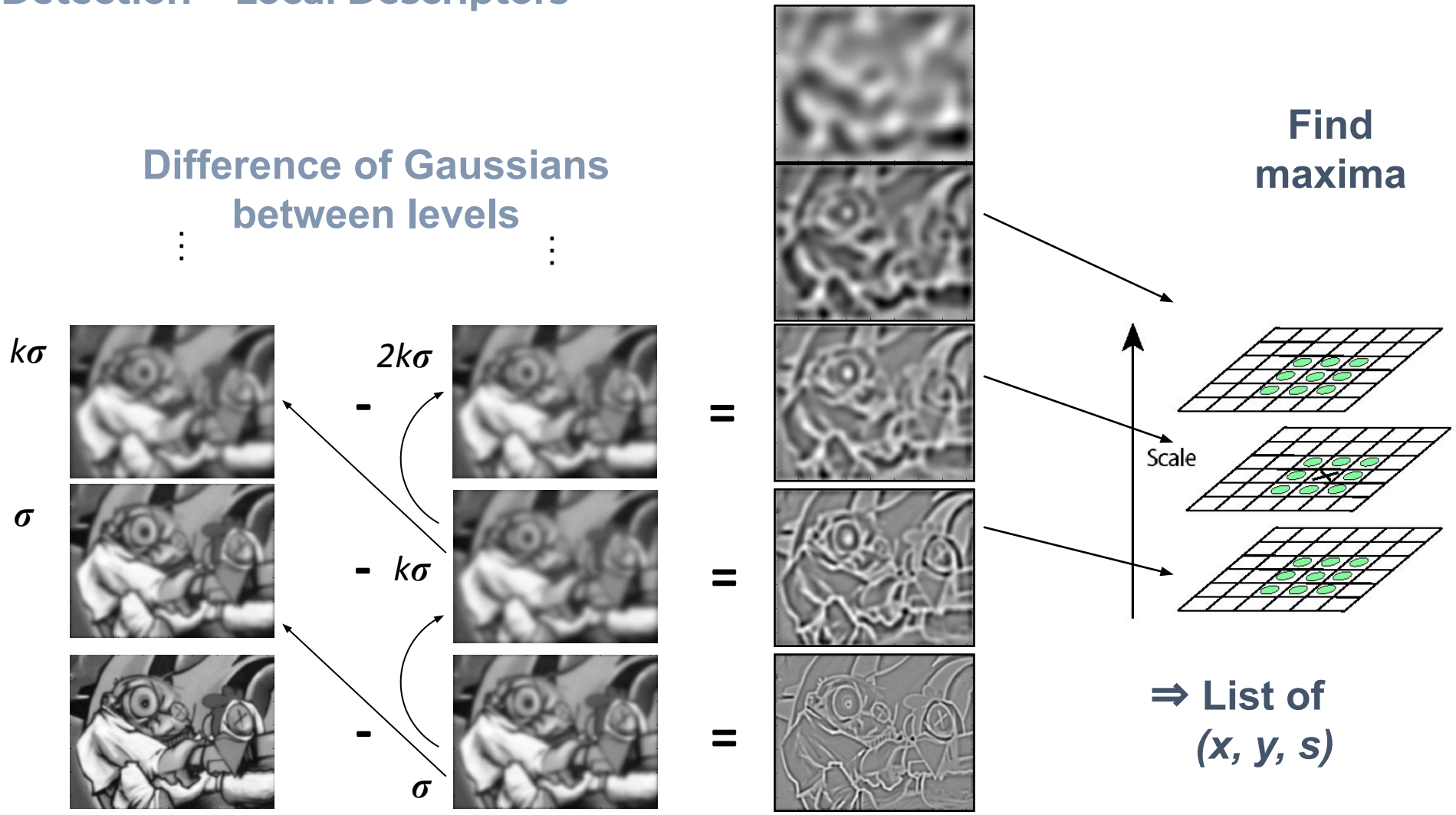


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Difference of Gaussians (DoG)

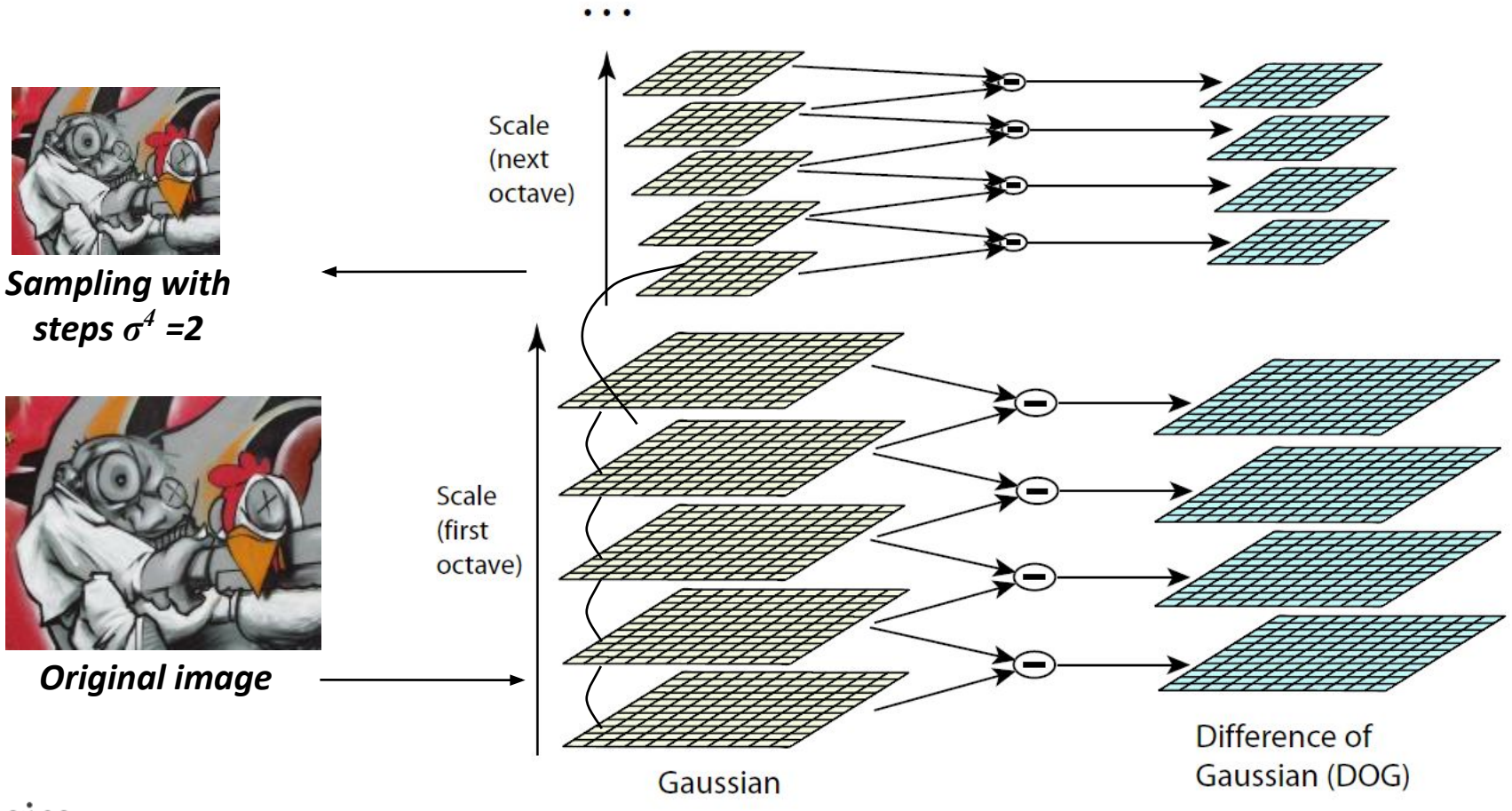
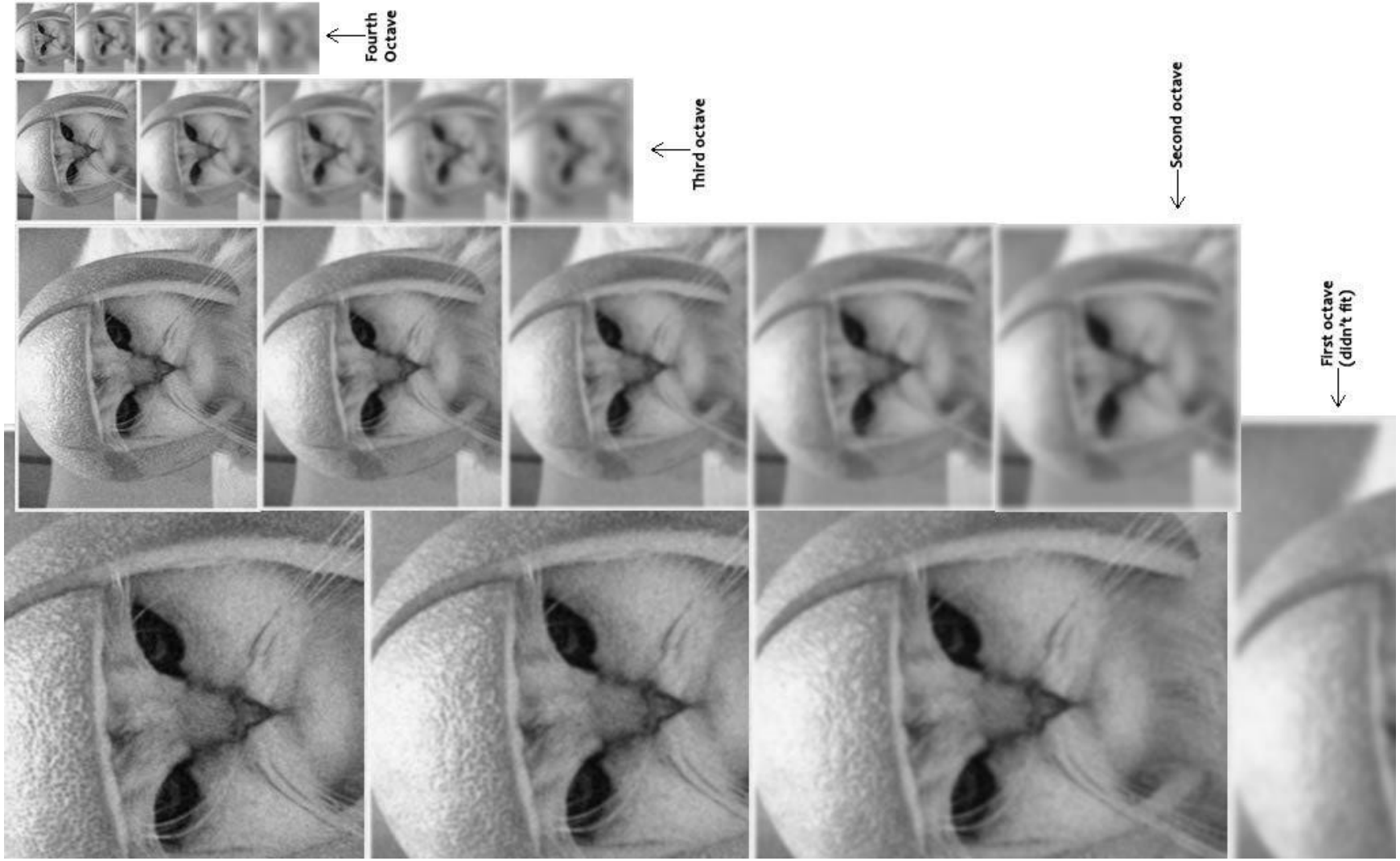


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



Advantages of Local Features

Locality

- features are local, so robust to occlusion and clutter

Distinctiveness:

- can differentiate a large database of objects

Quantity

- hundreds or thousands in a single image

Efficiency

- real-time performance achievable

Generality

- exploit different types of features in different situations

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

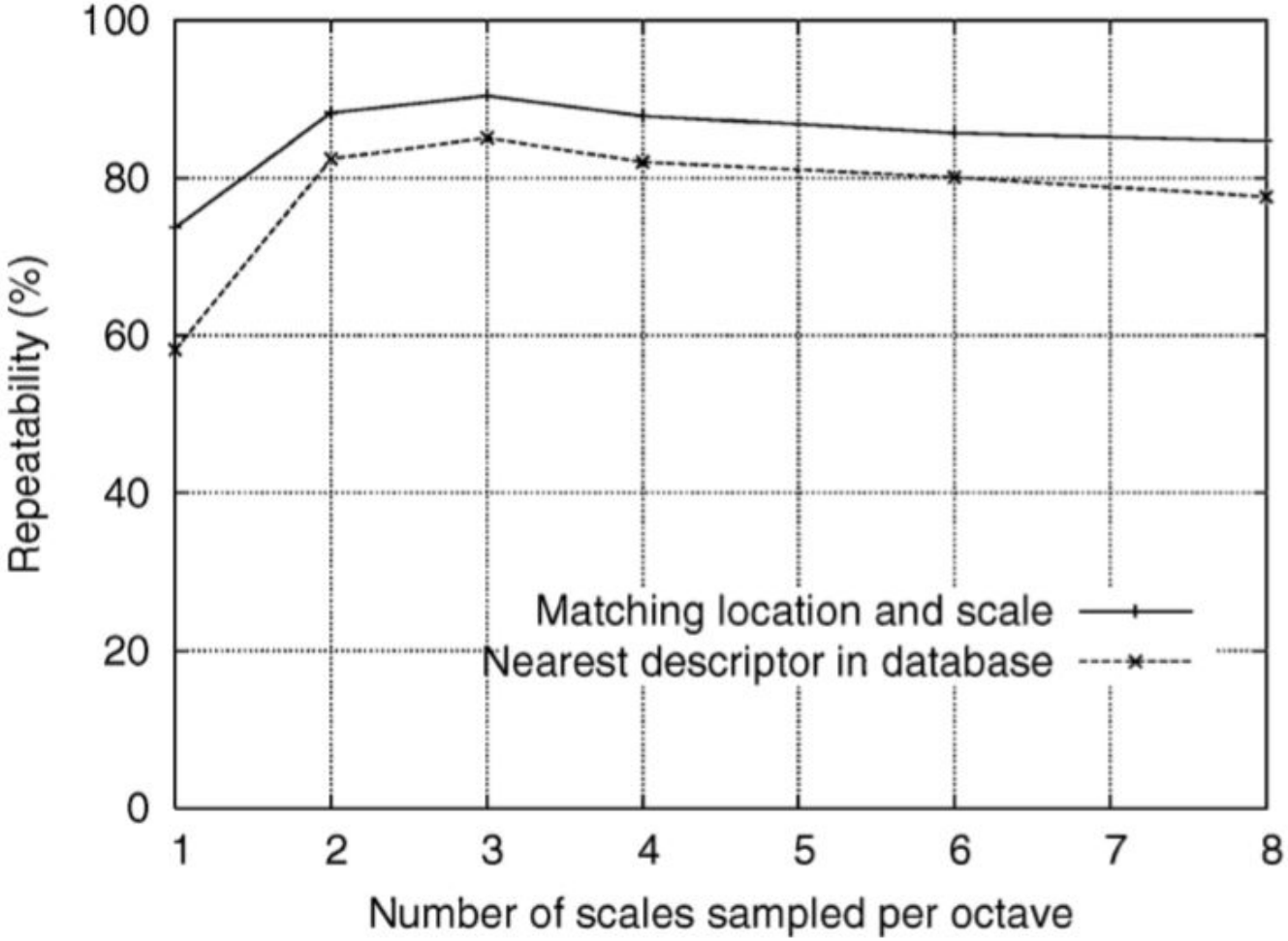


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

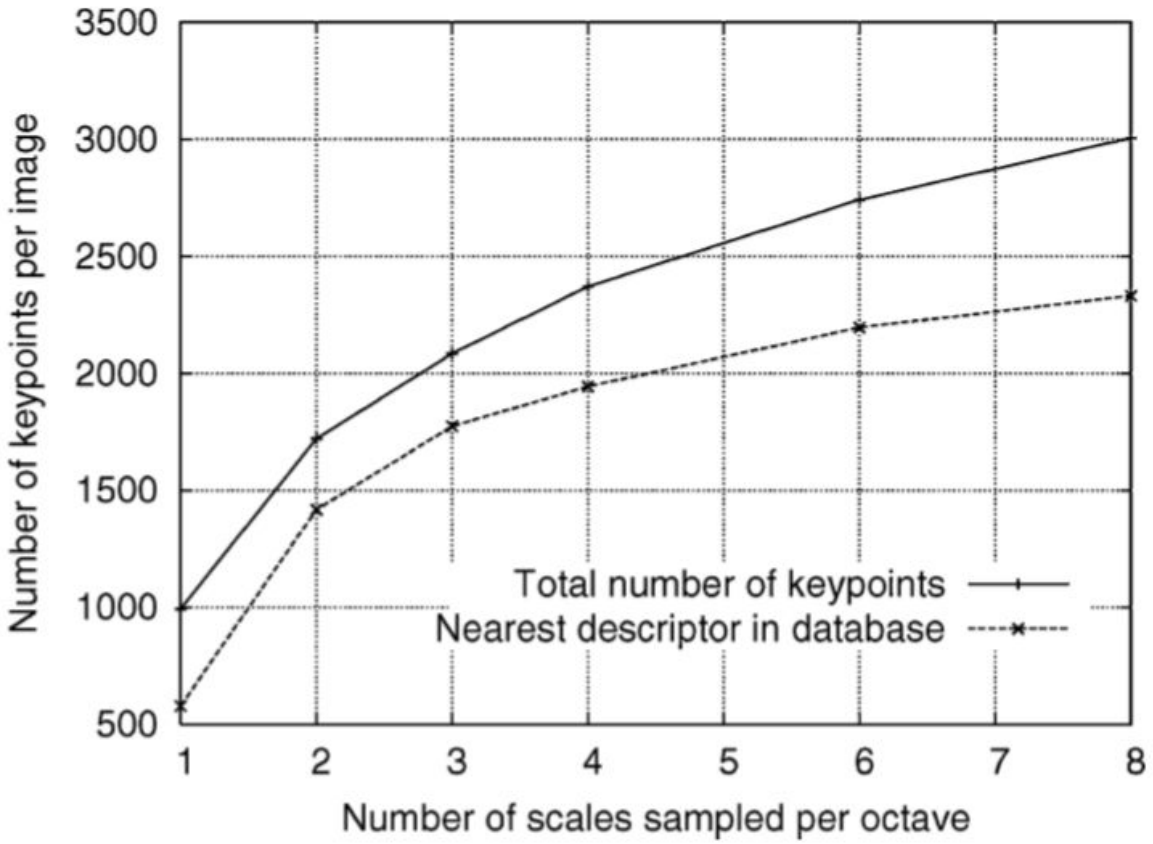


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Difference-of-Gaussian (DoG)

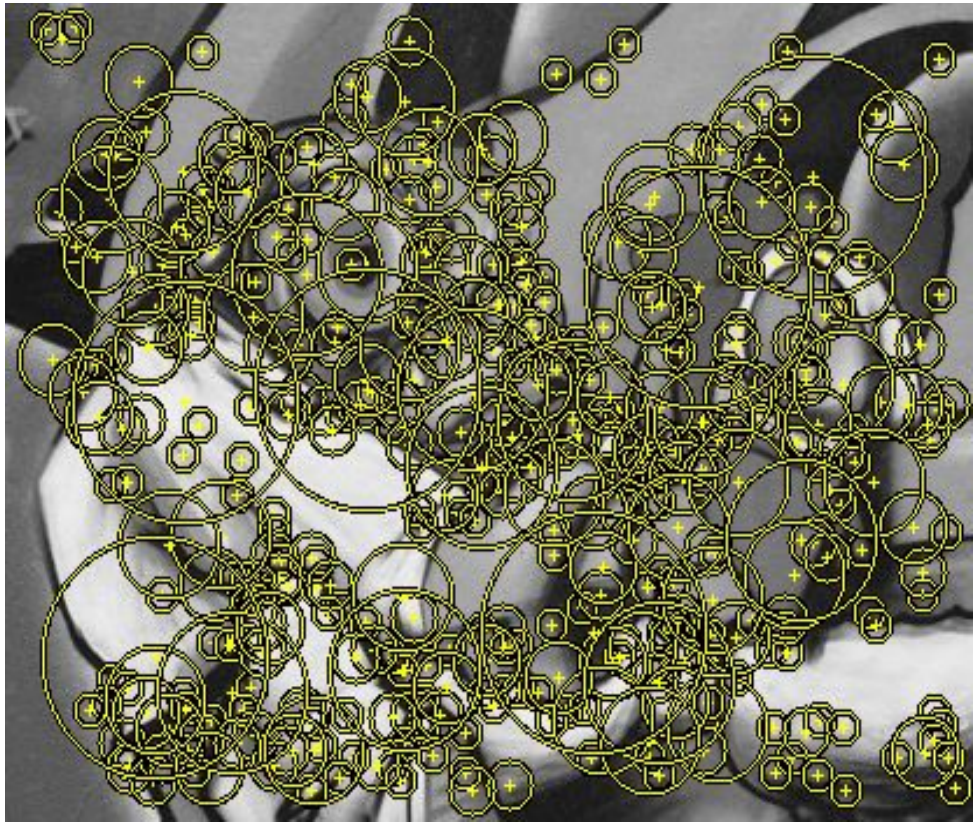
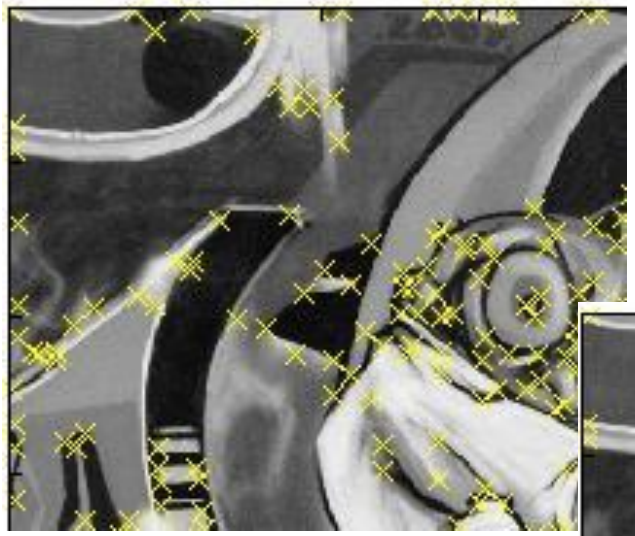


Image Analysis for Computer vision

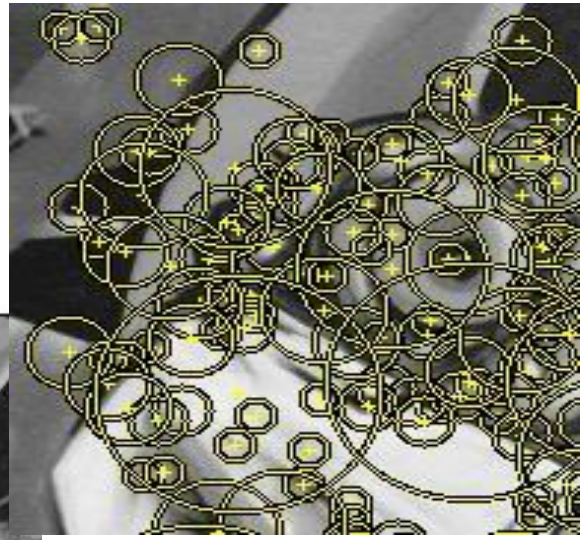
Keypoint Detection + Local Descriptors

Comparison



Harris

LoG
Laplacian of Gaussian



DOG

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Local descriptors how are they built?

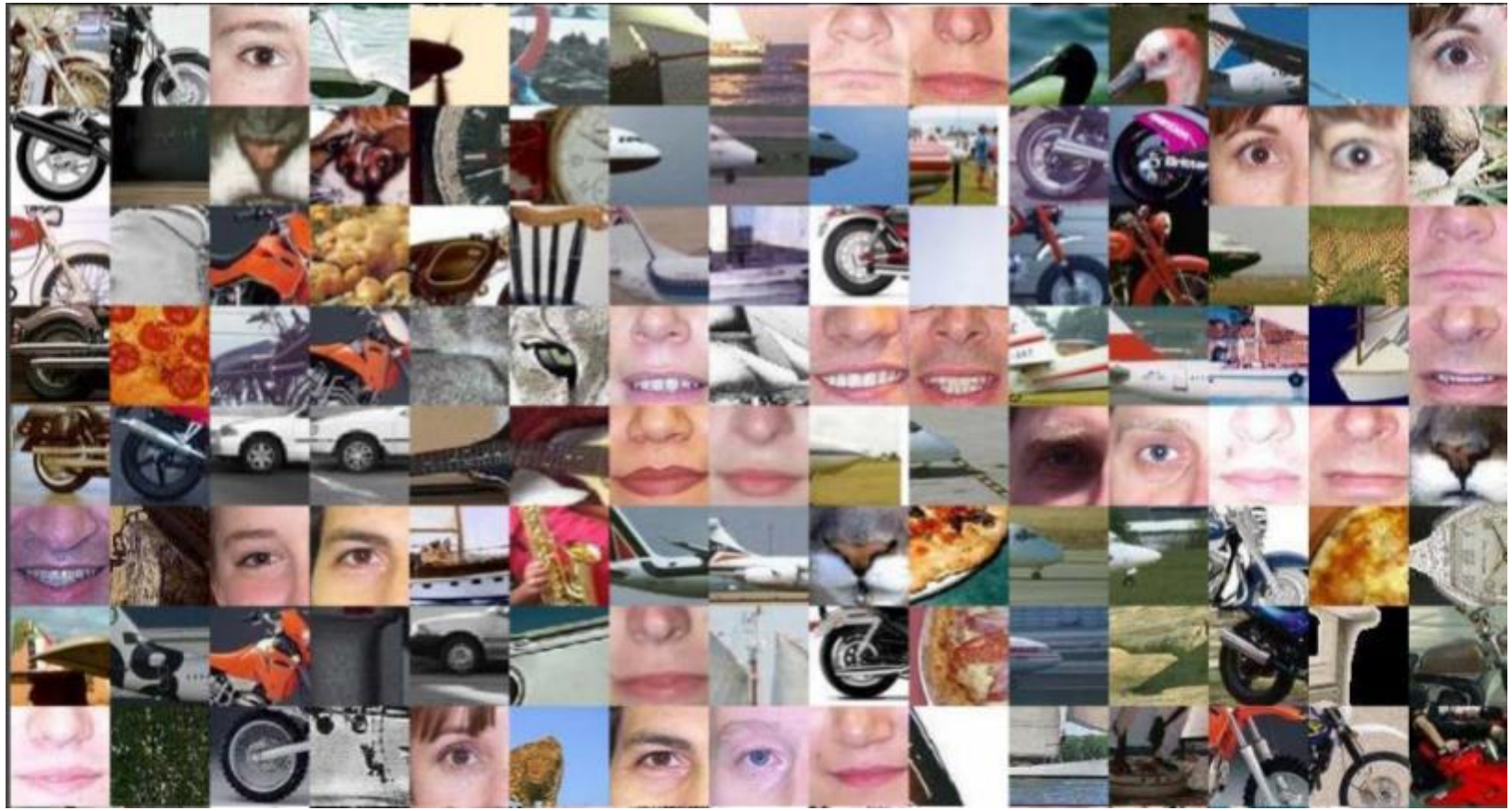


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Local descriptors how are they built?

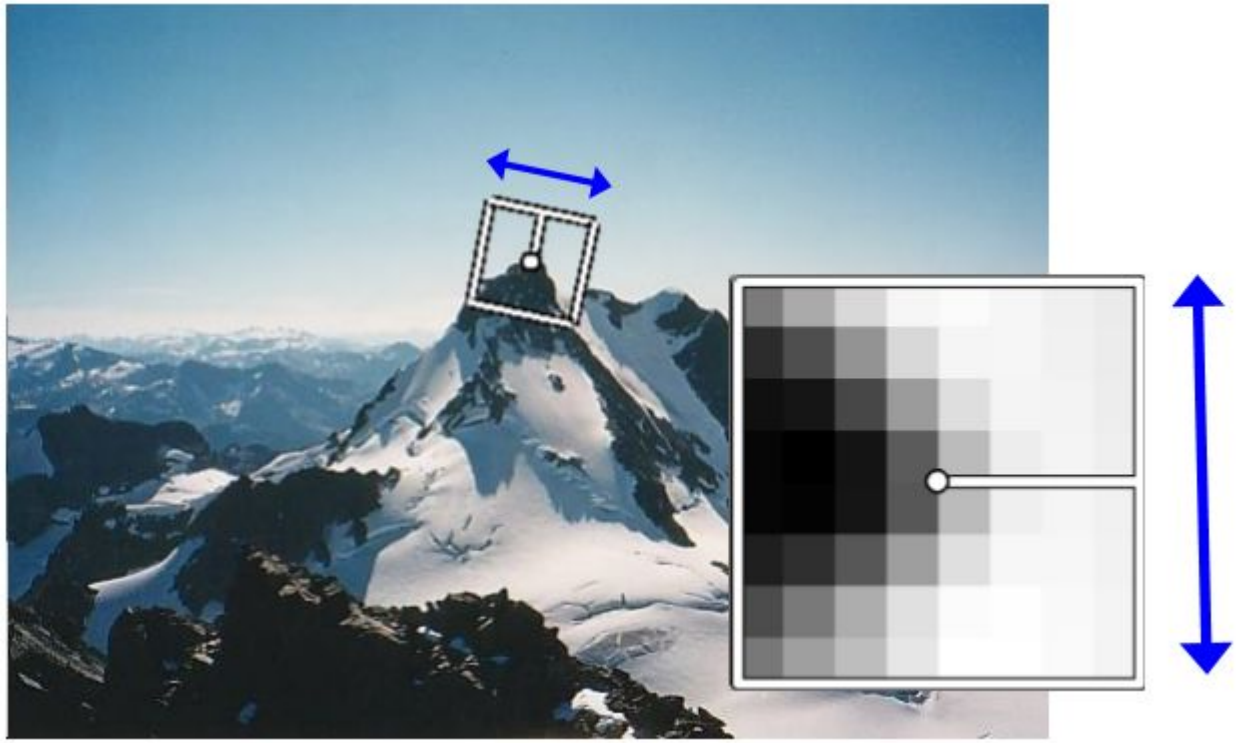
Patches are usually similar

Thus, descriptors must me alike!

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Local descriptors more than a patch!!!

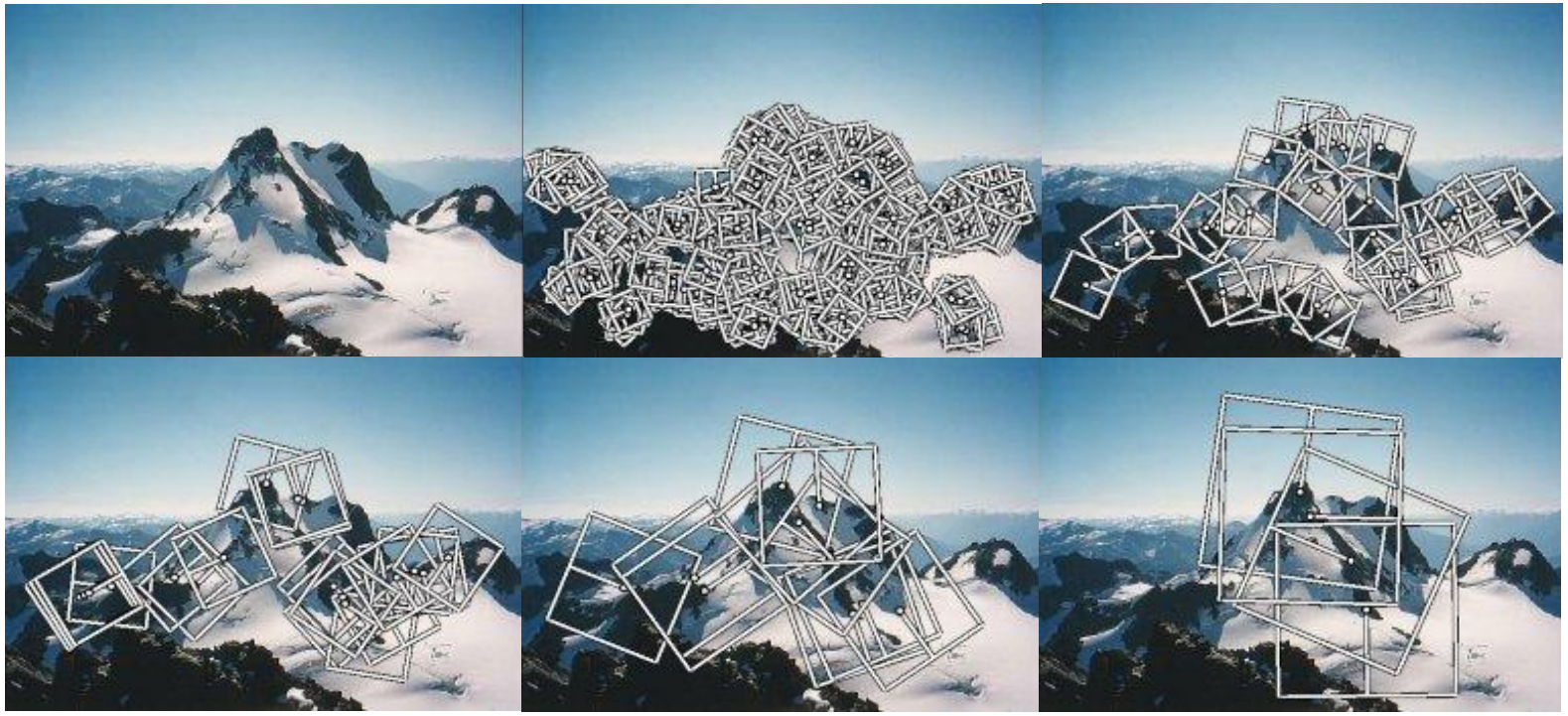


A simple patch has serious limitations, **which are they?**

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Local descriptors detecting patches at various scales



Advantages? Invariance to rotation, traslation and scale

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Applications image stitching

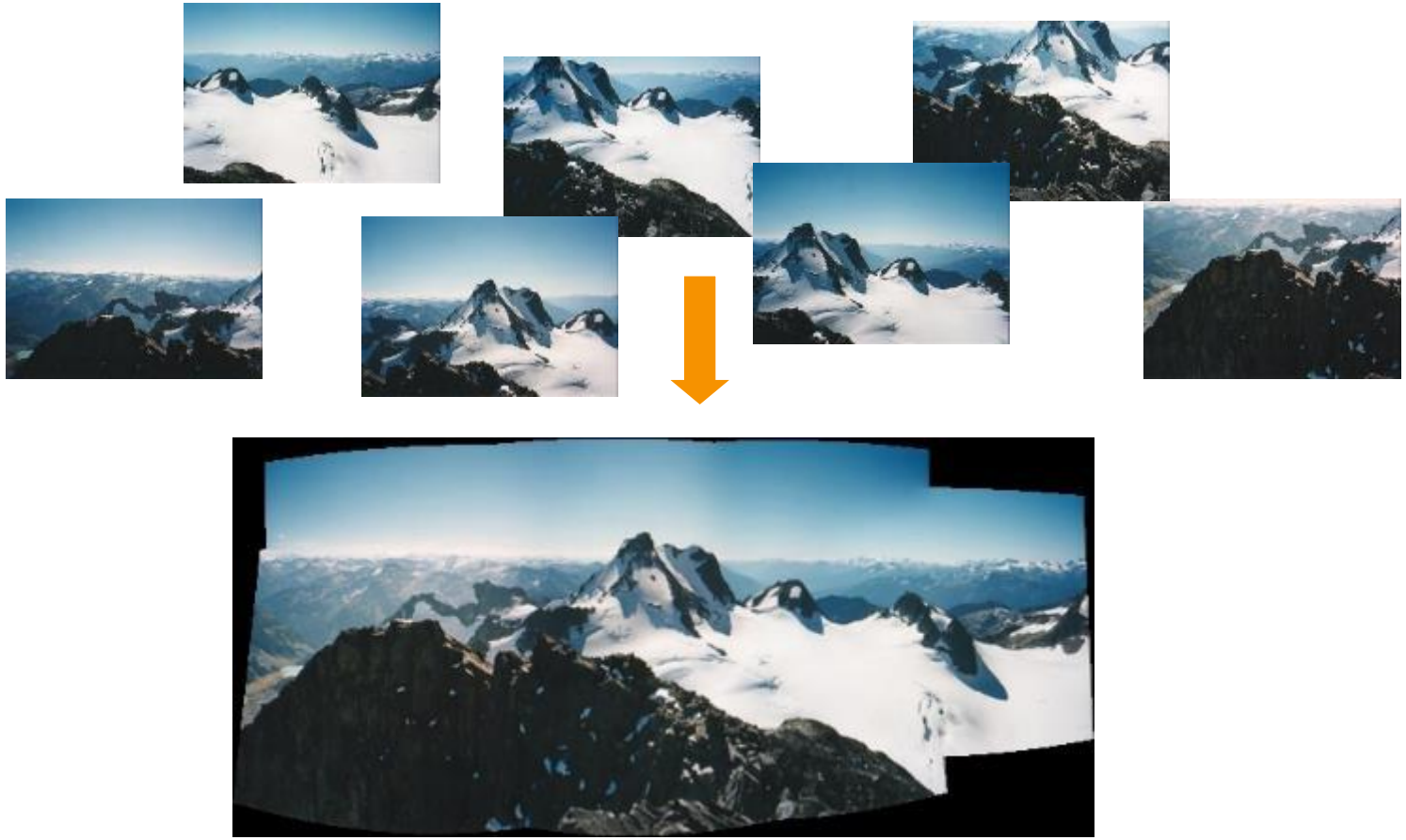


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Atributos locales visuales otras aplicaciones de los “descriptors”

Aquellas donde relaciones espacio-temporales entre imágenes sean importantes

Alineación de imágenes (homografías)

Reconstrucción 3D (e.g. Photo Tourism)

Aplicación de seguimiento de objetos

Motilón tracking/ flujo óptico

Navegación de robots(drones, sondas, UAVs)

Reconocimiento de objetos

Indexing y recuperación de bases de datos

(en estas dos ultimas pierde relevacion Deep Learning)

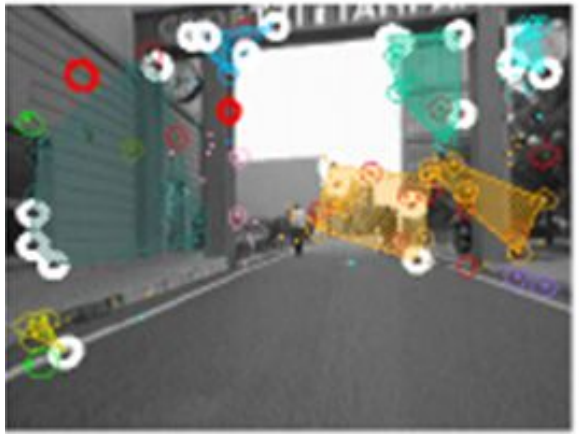


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

SIFT Descriptor: Scale Invariant Feature Transform

- Dive a window into cells of 4x4 (2x2 shwn below)
- Calculate a histogram of gradients for each cell
- 16 cells * 8 orientations □ **a descriptor of 128 dimensions**

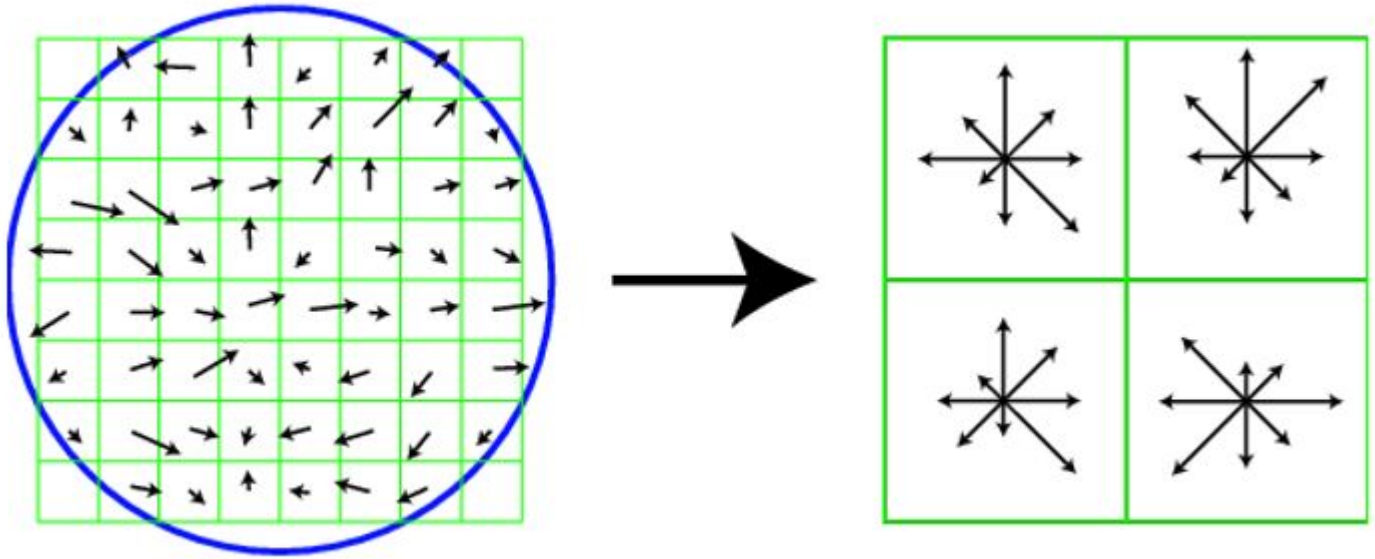


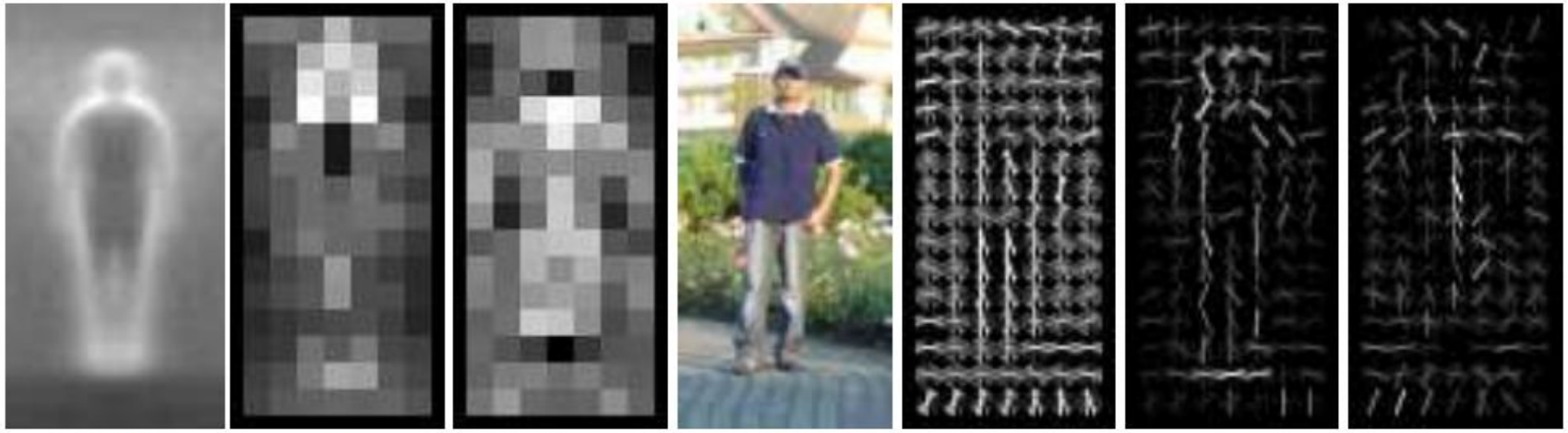
Image gradients

Keypoint descriptor

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

HOG



(a) (b) (c) (d) (e) (f) (g)

Histograms of Oriented Gradients for Human Detection, Dalal and Triggs, CVPR 2005

AP ~77%

More sophisticated methods: AP ~90%

(a) average gradient image over training examples
(b) each “pixel” shows max positive SVM weight in the block centered on that pixel
(c) same as (b) for negative SVM weights
(d) test image
(e) its R-HOG descriptor

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

HOG

1. Compute gradients in the region to be described
2. Put them in bins according to orientation
3. Group the cells into large blocks
4. Normalize each block
5. Train classifiers to decide if these are parts of a human

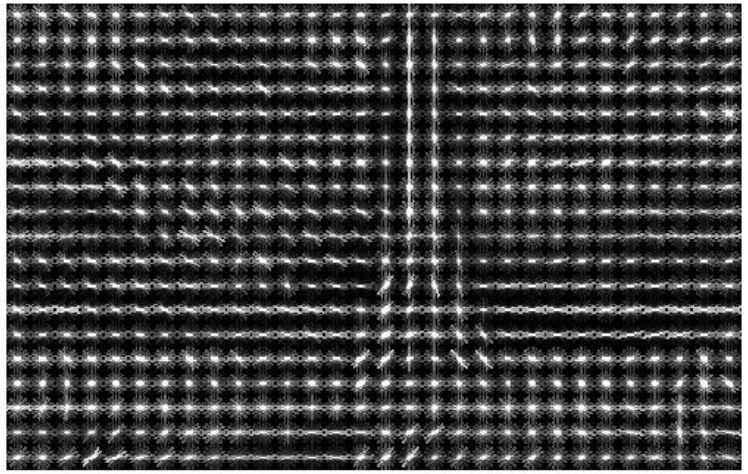


Image credit: N. Snavely

HOG Details

- Gradients

  [-1 0 1] and $[-1\ 0\ 1]^T$ were good enough filters.

- Cell Histograms

  Each pixel within the cell casts a weighted vote for an orientation-based histogram channel based on the values found in the gradient computation. (9 channels worked)

- Blocks

  Group the cells together into larger blocks, either R-HOG blocks (rectangular) or **C-HOG** blocks (circular).

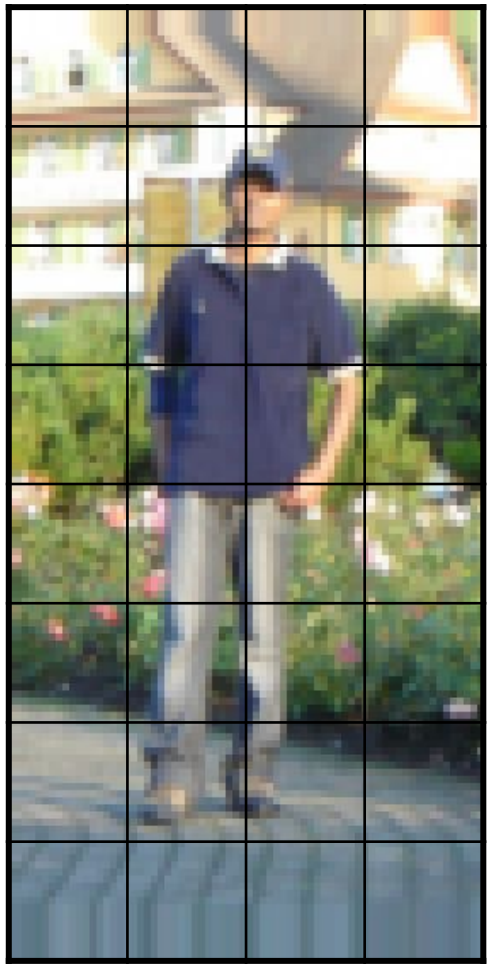
Example: Dalal-Triggs pedestrian detector

1. Extract fixed-sized (64x128 pixel) window at each position and scale
2. Compute HOG (histogram of gradient) features within each window
3. Score the window with a linear SVM classifier
4. Perform non-maxima suppression to remove overlapping detections with lower scores

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

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



-1	0	1

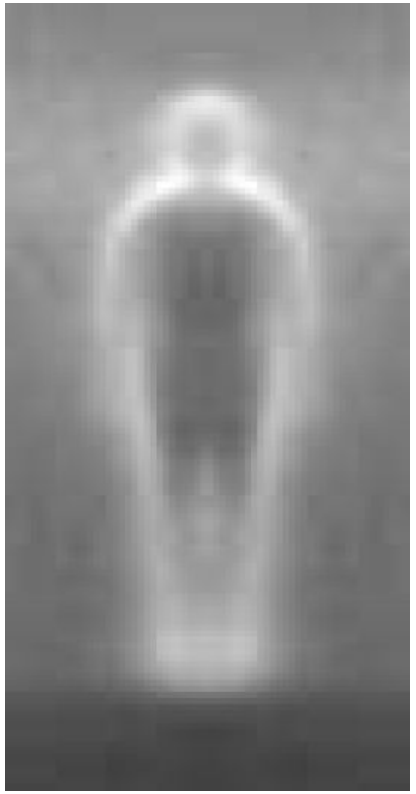
centered

-1	1

uncentered

1	-8	0	8	-1

cubic-corrected



0	1
-1	0

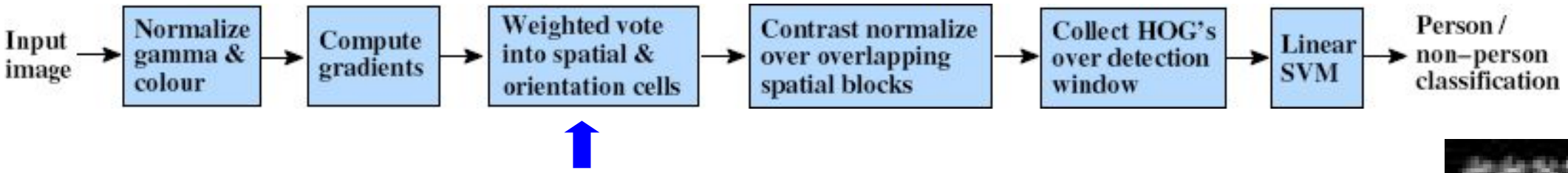
diagonal

-1	0	1
-2	0	2
-1	0	1

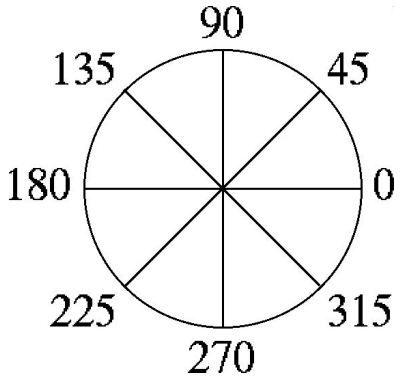
Sobel

Image Analysis for Computer vision

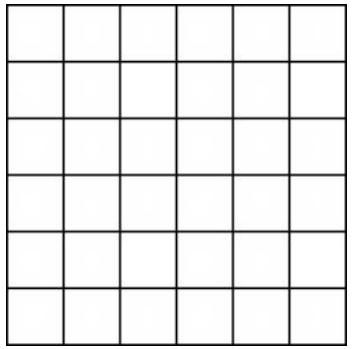
Keypoint Detection + Local Descriptors



Orientation: 9 bins (for unsigned angles)



Histograms in 8x8 pixel cells



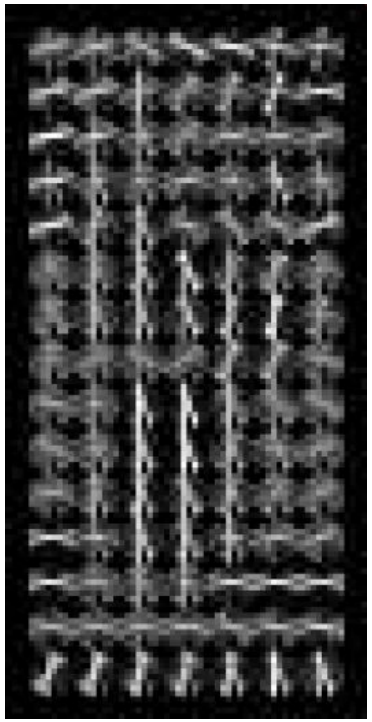
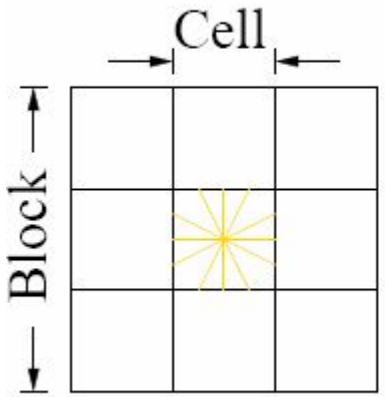


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

R-HOG

Normalize with respect to surrounding cells

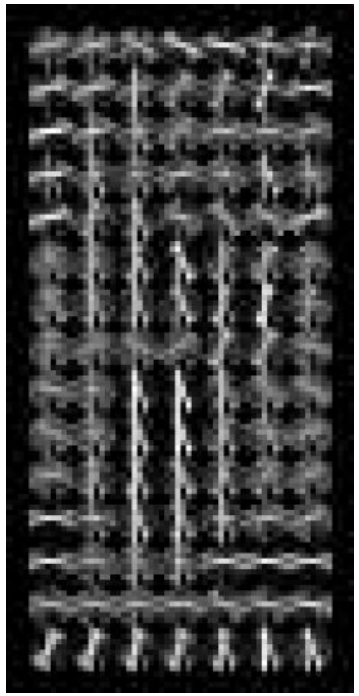


$$L2-norm: v \longrightarrow v/\sqrt{||v||_2^2+\epsilon^2}$$

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

X=



orientations

features = 15 x 7 x 9 x 4 = 3780

cells

normalizations by neighboring cells

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

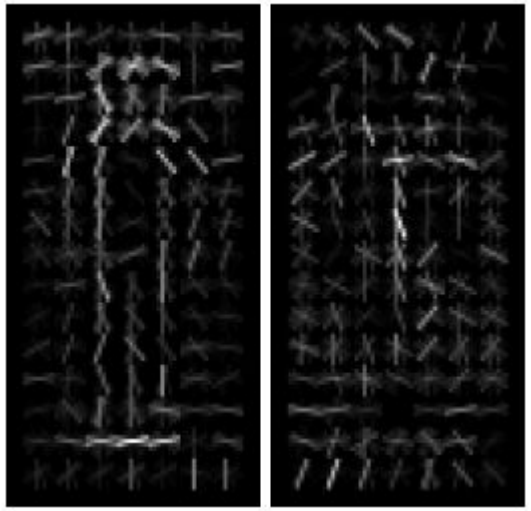
Training set

+

−

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



pos w neg w

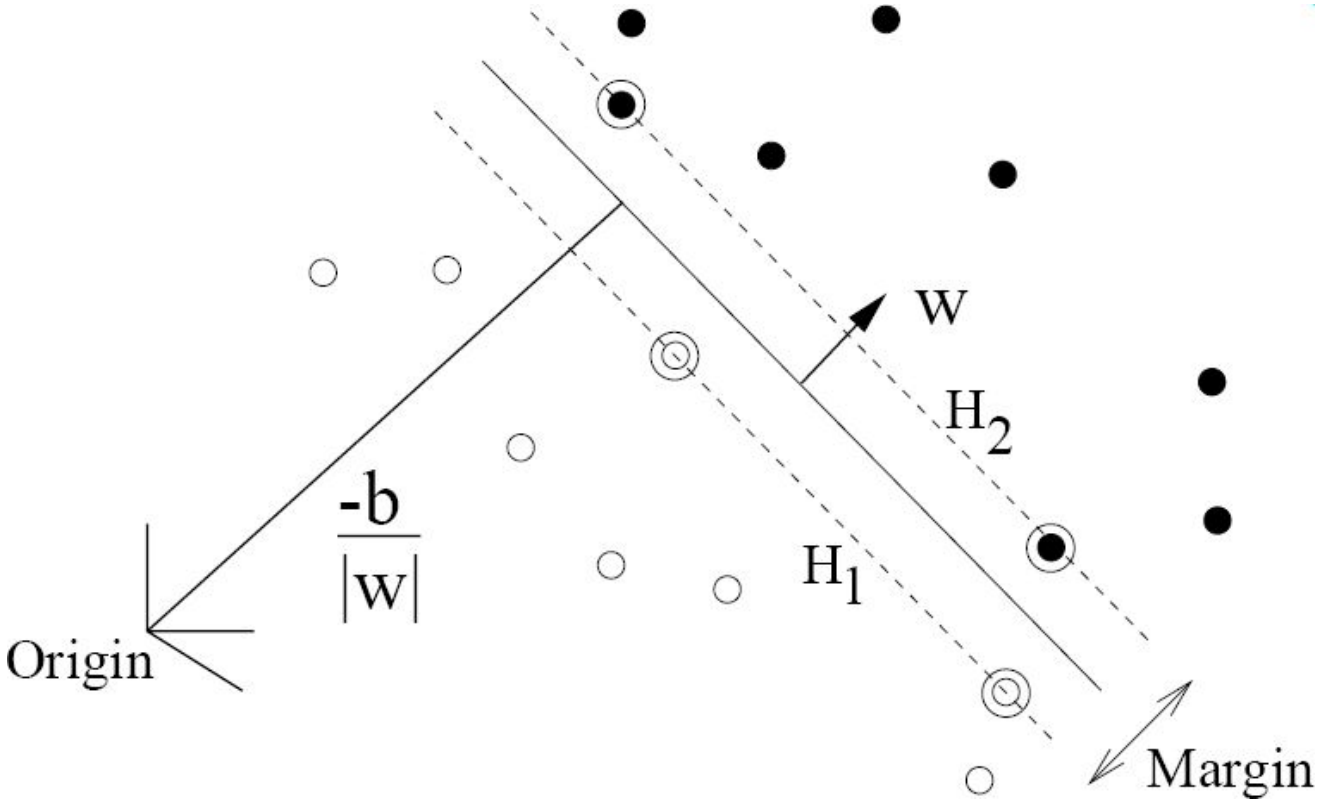
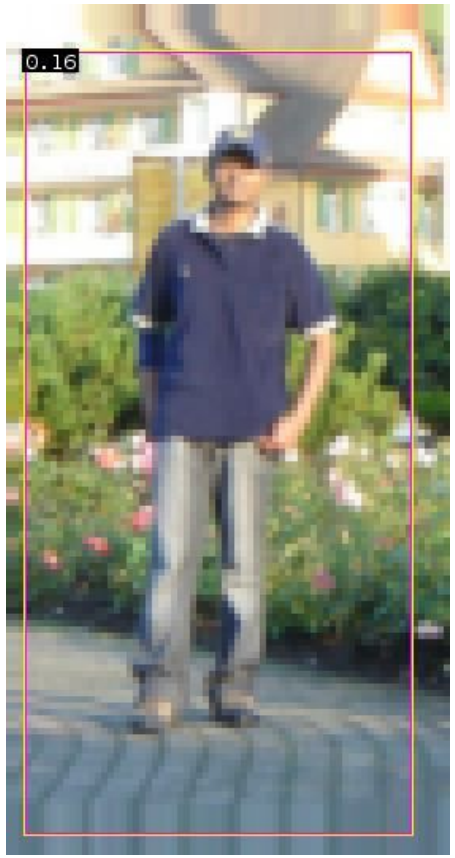


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



$$0.16 = w^T x - b$$

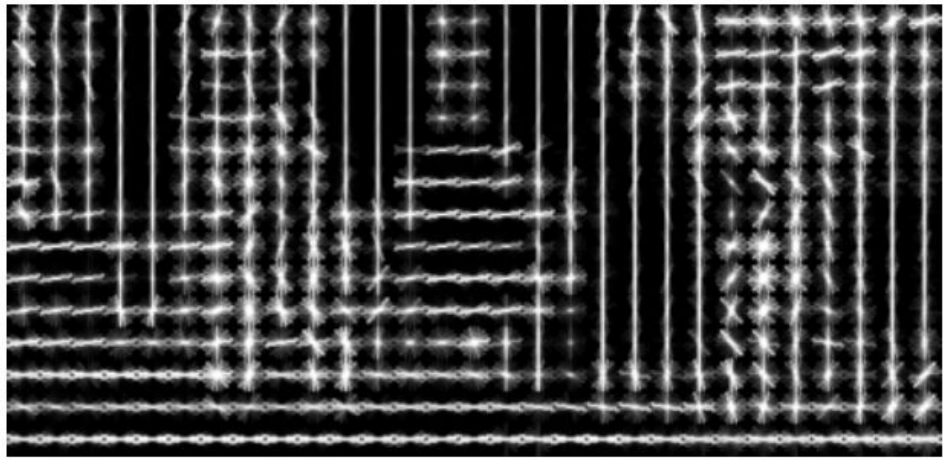
$$sign(0.16) = 1$$

pedestrian

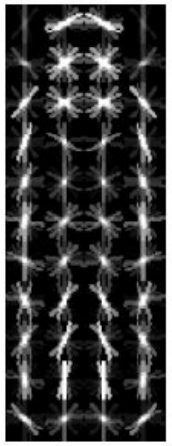
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

HOG feature map



Template



Detector response map

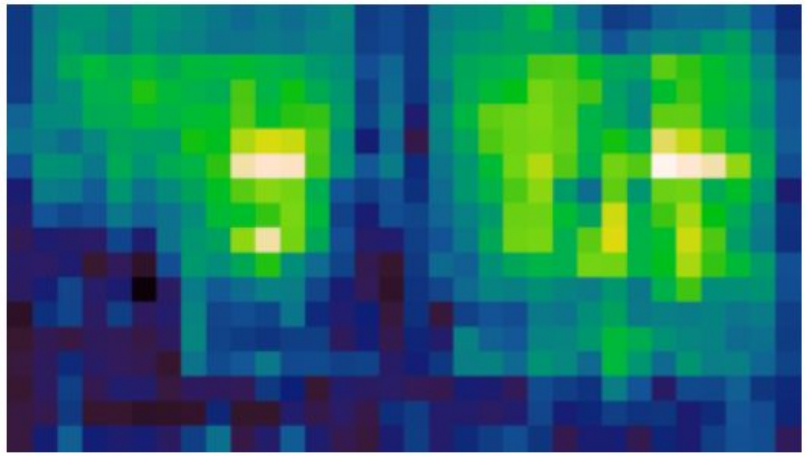
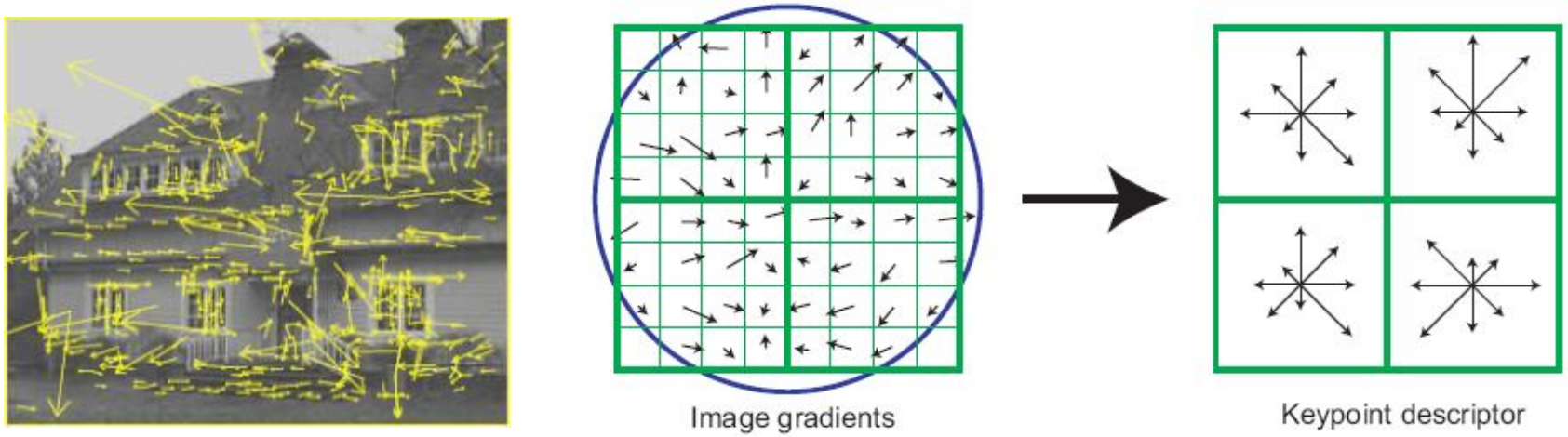


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

SIFT Descriptor: Scale Invariant Feature Transform

- Dive a window into cells of 4x4 (2x2 shwn below)
- Calculate a histogram of gradients for each cell
- 16 cells * 8 orientations □ **a descriptor of 128 dimensions**



SIFT Descriptor: Scale Invariant Feature Transform

- Dive a window into cells of 4x4 (2x2 shwn below)
- Calculate a histogram of gradients for each cell
- 16 cells * 8 orientations □ **a descriptor of 128 dimensions**

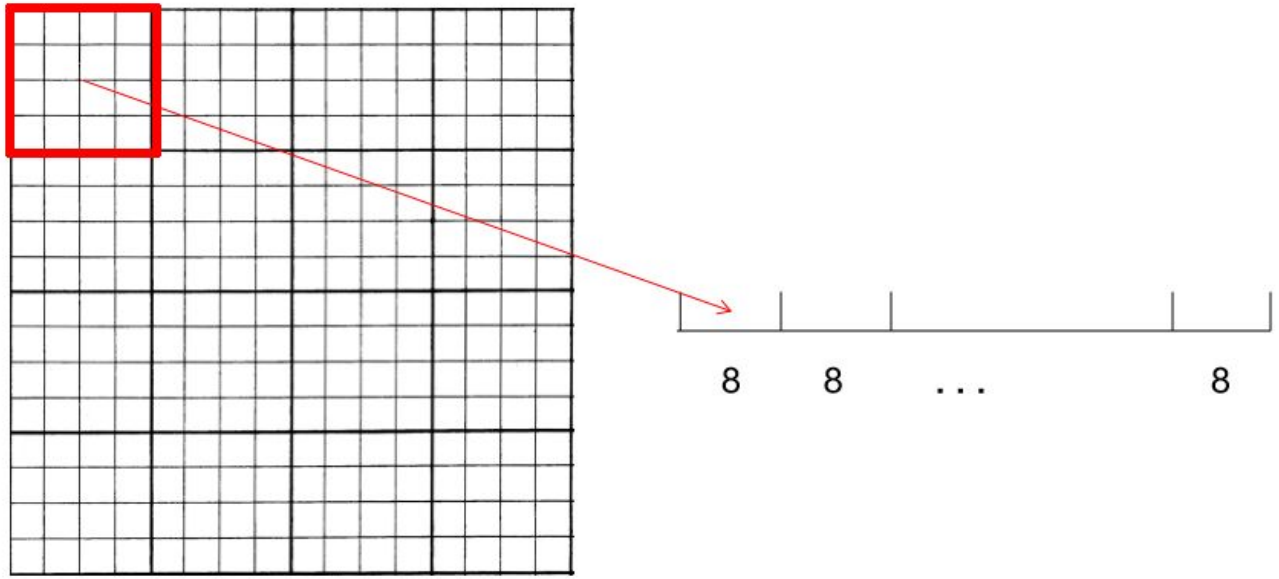
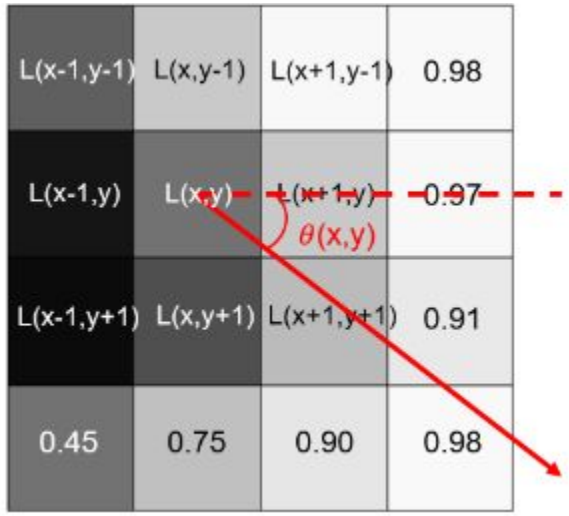


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

0.37	0.79	0.97	0.98
0.08	0.45	0.79	0.97
0.04	0.31	0.73	0.91
0.45	0.75	0.90	0.98

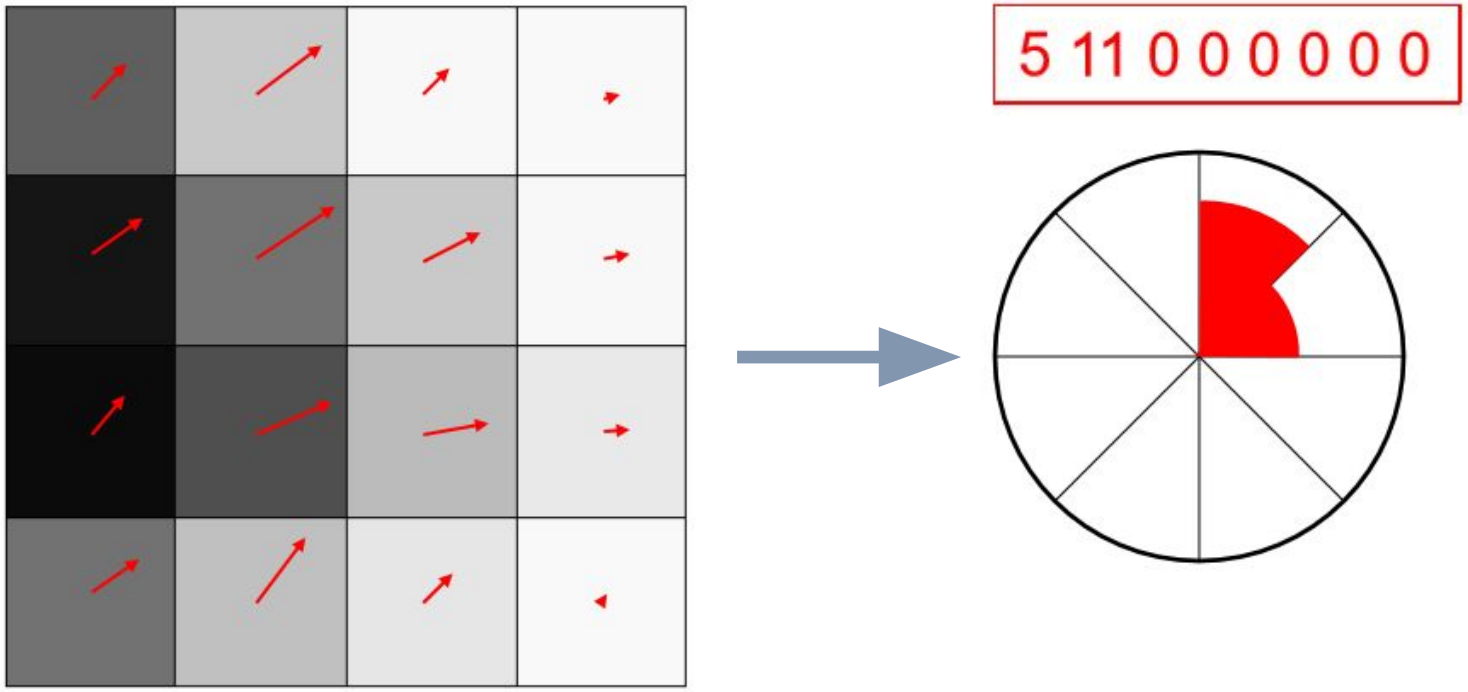


$$\text{magnitude}(x,y)=\sqrt{\left(L(x+1,y)-L(x-1,y)\right)^2+\left(L(x,y+1)-L(x,y-1)\right)^2}$$

$$\theta(x,y)=atan\left(\left(\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)}\right)\right)$$

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors



Orientations for each of the 16 pixels in the cell

In this exampe, the orientations are two “bins”: 5 & 11, respectively

SIFT Descriptor: Scale Invariant Feature Transform

- Dive a window into cells of 4x4 (2x2 shwn below)
- Calculate a histogram of gradients for each cell
- 16 cells * 8 orientations □ **a descriptor of 128 dimensions**
- Apply a treshold to normalize the vector

$$\sum_i d_i^2 = 1 \quad \text{such that: } d_i < 0.2$$

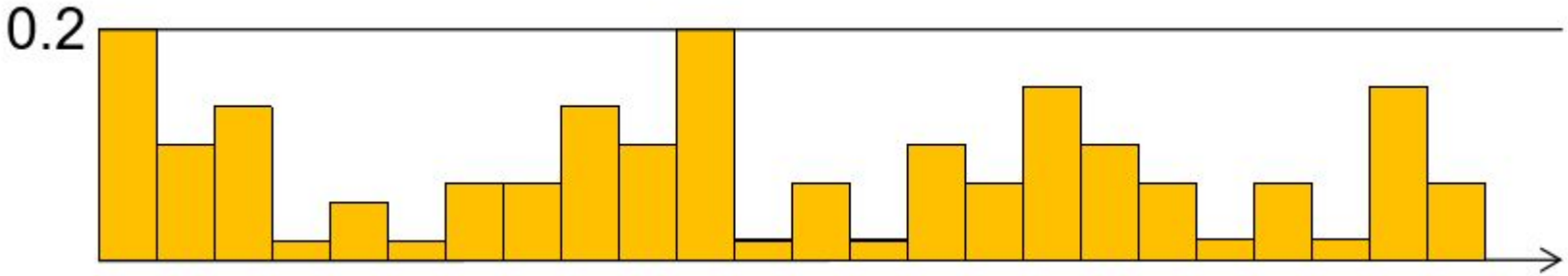


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

SIFT Descriptor - Properties

- Extraordinarilly robust technique for finding “matches”
- It can stand changes in “viewpoint” – rotactions of up to 30°
- It can handle changes in illumination – Day vs. Night (below)
- It reduces the computational load of many algorithms sparce computing

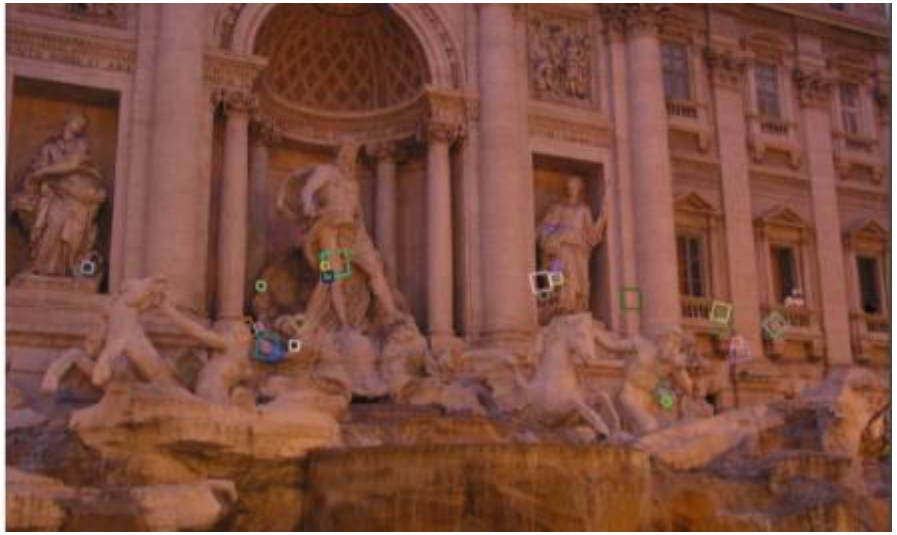


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

1. **Scale-space extrema detection**

   Search over multiple scales and image locations.

2. **Keypoint localization**

   Fit a model to detrmine location and scale.
   Select keypoints based on a measure of stability.

3. **Orientation assignment**

   Compute best orientation(s) for each keypoint region.

4. **Keypoint description**

   Use local image gradients at selected scale and rotation to describe each keypoint region.

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

SIFT Descriptor – Properties and applications

Very powerful for recognizing object instances
Particularly in scenarios with Good textures
Now surpassed by Deep Learning methods

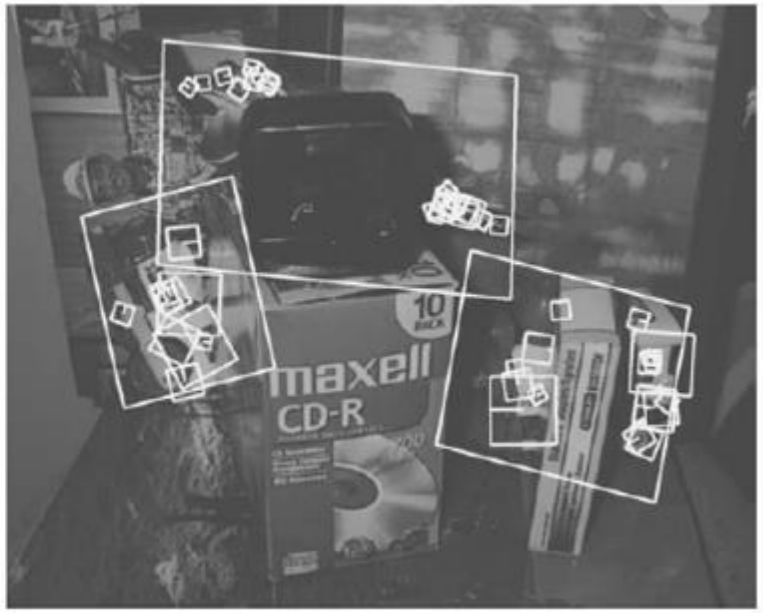


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

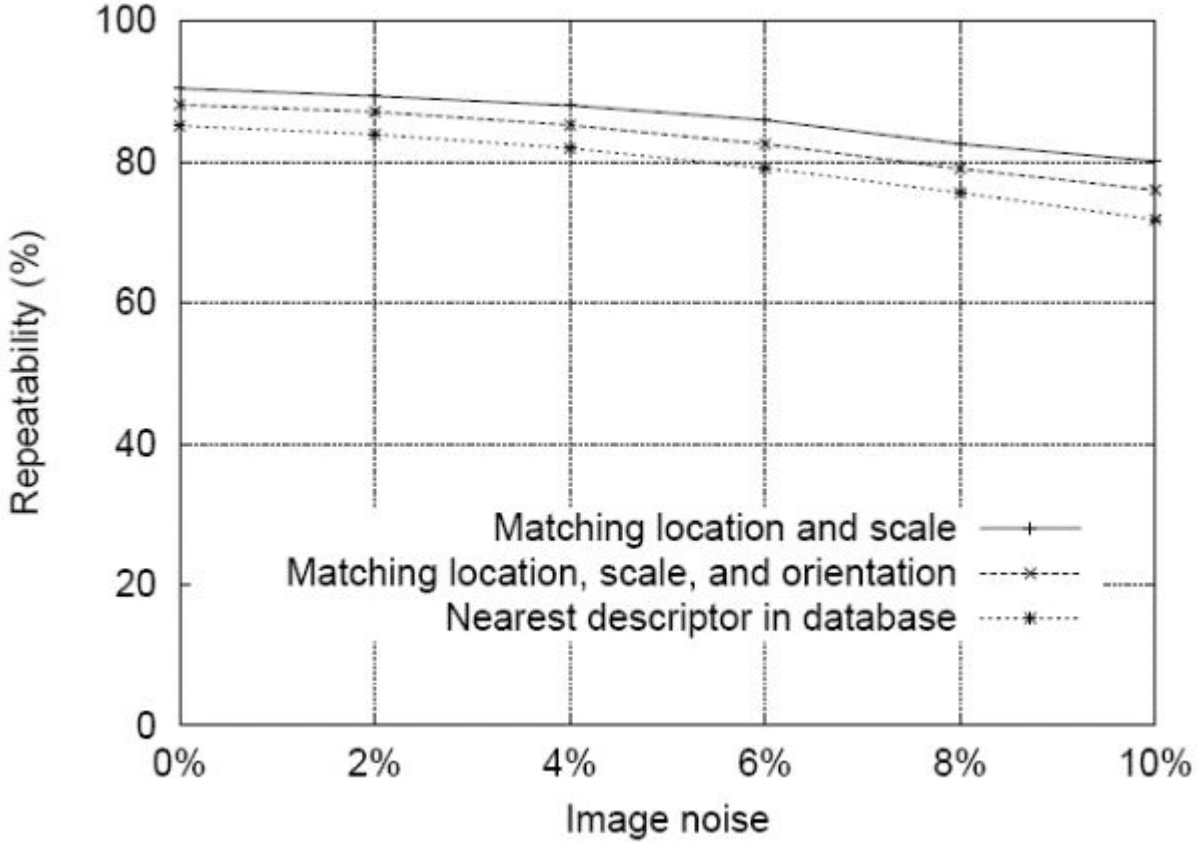
The SIFT descriptor is powerful for recognizing object instances

It works even in environment wiht strong changes in illumination

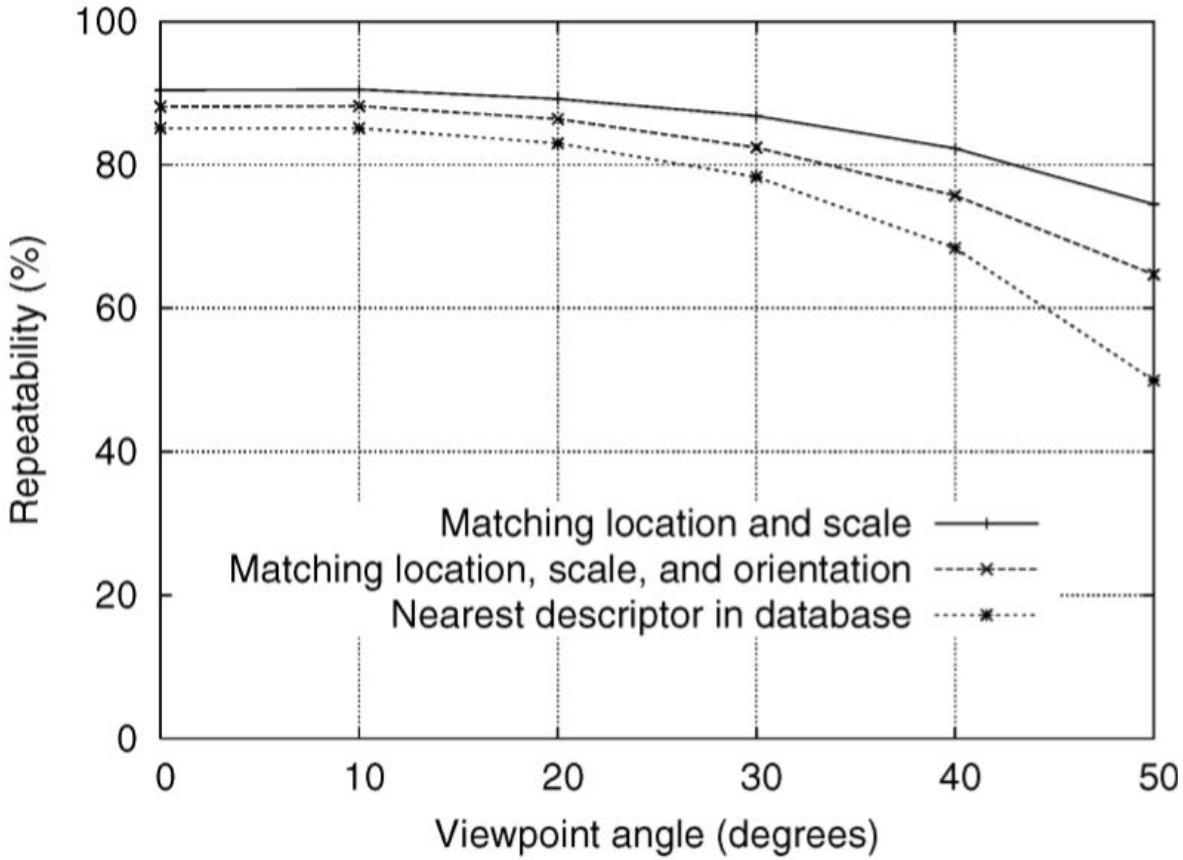
This makes it usefull for apps in

Robotics
UAVs (drones)
Augmented Reality
Virtual Reality

SIFT Descriptor – Properties – Repeatability



SIFT Descriptor – Properties – Repeatability



SIFT Descriptor – Properties – Repeatability

Table 7.1 Overview of feature detectors.

Feature Detector	Corner	Blob	Region	Rotation invariant	Scale invariant	Affine invariant	Repeatability	Localization accuracy	Robustness	Efficiency
Harris	√			√			+++	+++	+++	++
Hessian		√		√			++	++	++	+
SUSAN	√			√			++	++	++	+++
Harris-Laplace	√	(√)		√	√		+++	+++	++	+
Hessian-Laplace	(√)	√		√	√		+++	+++	+++	+
DoG	(√)	√		√	√		++	++	++	++
SURF	(√)	√		√	√		++	++	++	+++
Harris-Affine	√	(√)		√	√	√	+++	+++	++	++
Hessian-Affine	(√)	√		√	√	√	+++	+++	+++	++
Salient Regions	(√)	√		√	√	(√)	+	+	++	+
Edge-based	√			√	√	√	+++	+++	+	+
MSER			√	√	√	√	+++	+++	++	+++
Intensity-based			√	√	√	√	++	++	++	++
Superpixels			√	√	(√)	(√)	+	+	+	+

Aunque sumamente eficiente, otros algoritmos han sido propuestos para ciertas aplicaciones (sistemas embebidos, moviles), como SURF y ORB, **algoritmos a ser presentados por ustedes en la próxima clase**

Image Analysis for Computer vision

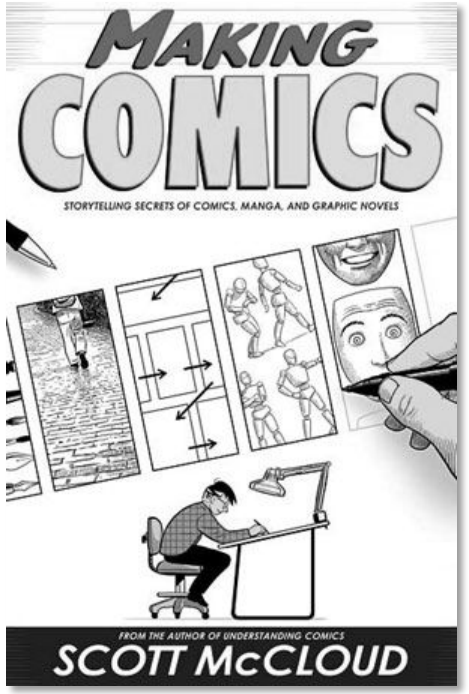
Keypoint Detection + Local Descriptors

Detector	Illumination	Rotation	Scale	View point
Lowe '99 (DoG)	Yes	Yes	Yes	No
Harris corner	Yes	Yes	No	No
Mikolajczyk & Schmid '01, '02	Yes	Yes	Yes	Yes
Tuytelaars, '00	Yes	Yes	No (Yes '04)	Yes
Kadir & Brady, 01	Yes	Yes	Yes	no
Matas, '02	Yes	Yes	Yes	no

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

What is the **geometric relation** between this two images?



¿?

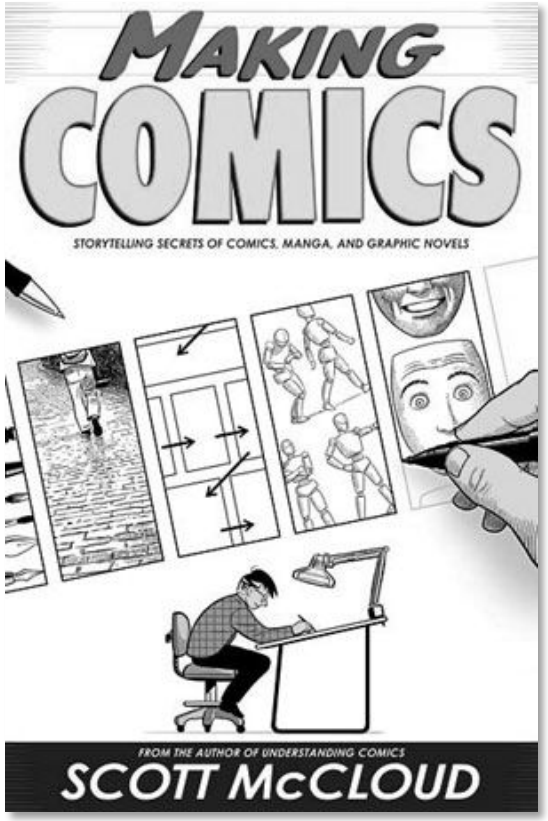
Asnwer: **Similarity transformation** (translation, rotation, uniform scale)

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

What is the **geometric relation** between this two images?

How do we find information about this correspondence?



¡Usando SIFT!

+

Calibración de cámaras

(next class)

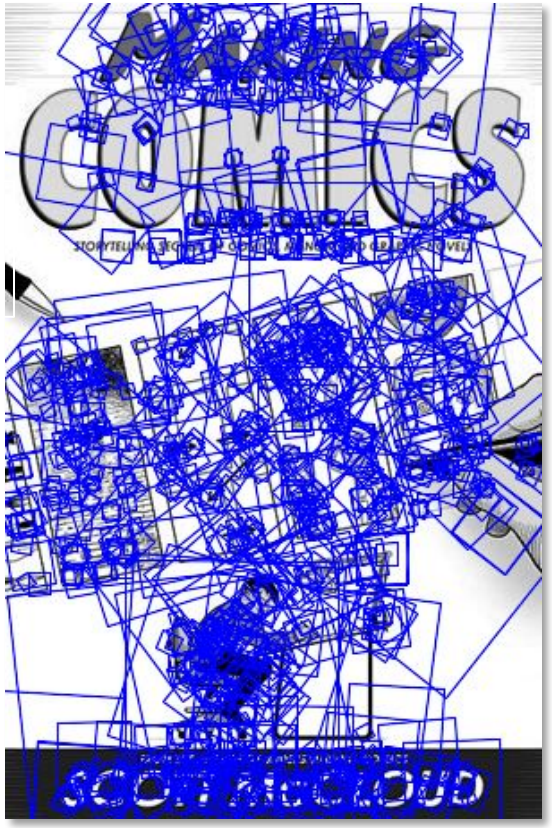
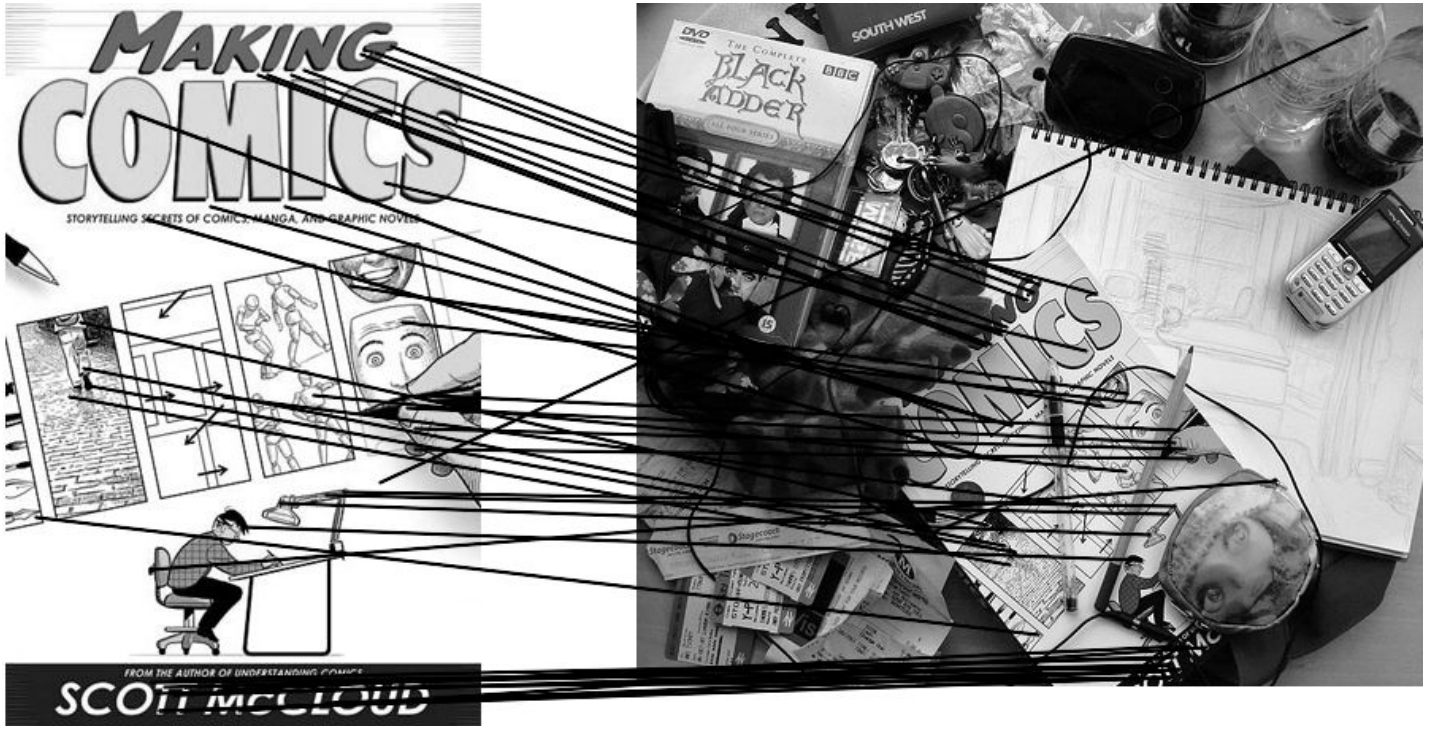


Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

What is the **geometric relation** between this two images?

How do we find information about this correspondence?



We have to do what is known as **feature matching**

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

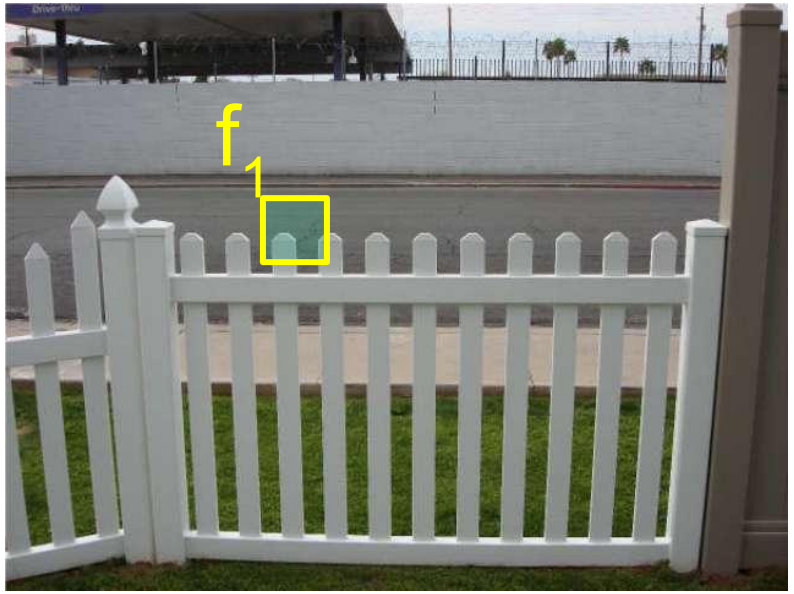
Tenemos que hacer los que se conoce como **feature matching**

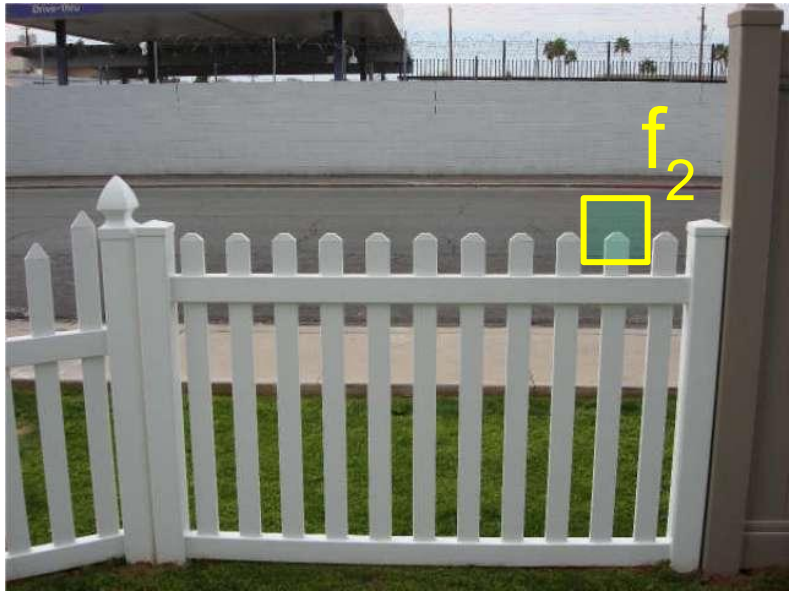
¿Como definimos la **distancia** entre dos **features** f_1, f_2? □

Acercamiento sencillo: Distancia L_2, $||f_1 - f_2||$

Puede dar buenos "scores" a matches ambiguos □ incorrectos

I_1





I_2

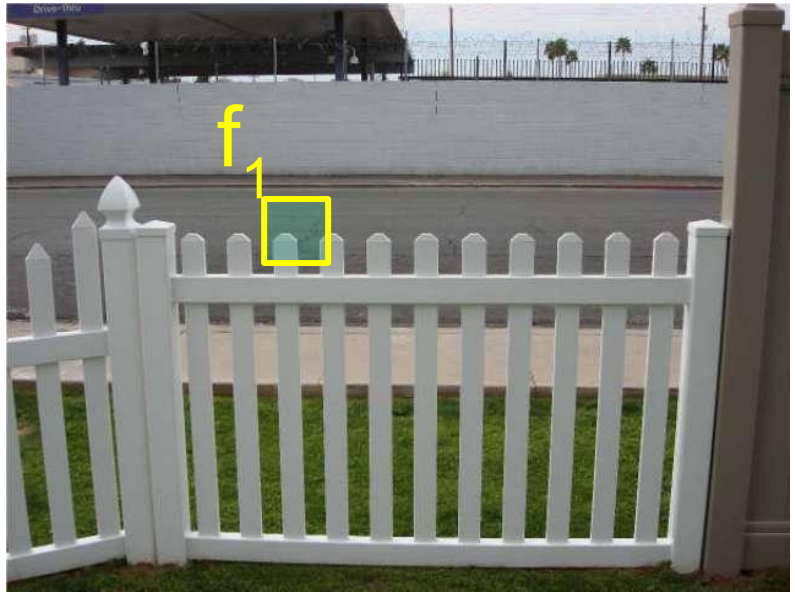
Image Analysis for Computer vision

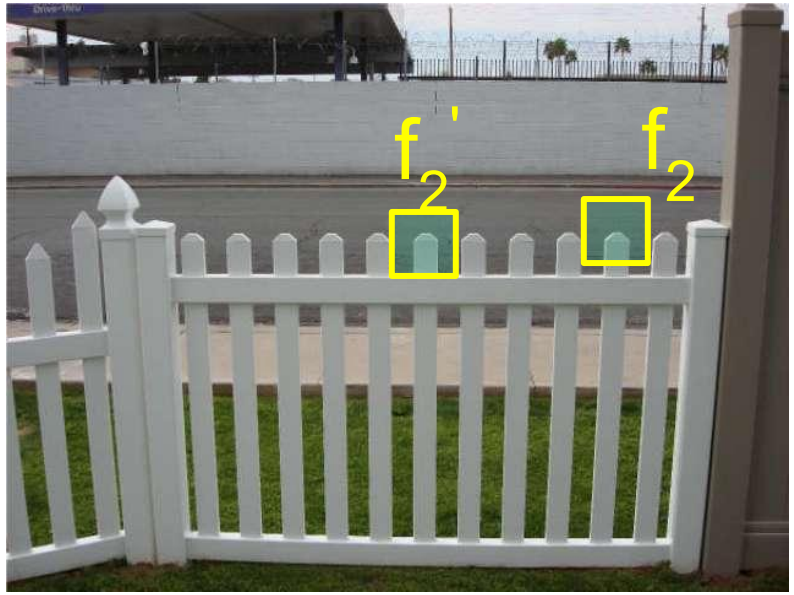
Keypoint Detection + Local Descriptors

Tenemos que hacer los que se conoce como **feature matching**

- Mejor acercamiento: distancia de los radios= $||f_1 - f_2|| / ||f_1 - f_2'||$
- **Sum of Squared Differences (SSD)**
  - f_2 es el 2do SSD match para f_1 en I_2 y f_2' es el 2do
  - Da valores altos para mathes ambibuos

I_1



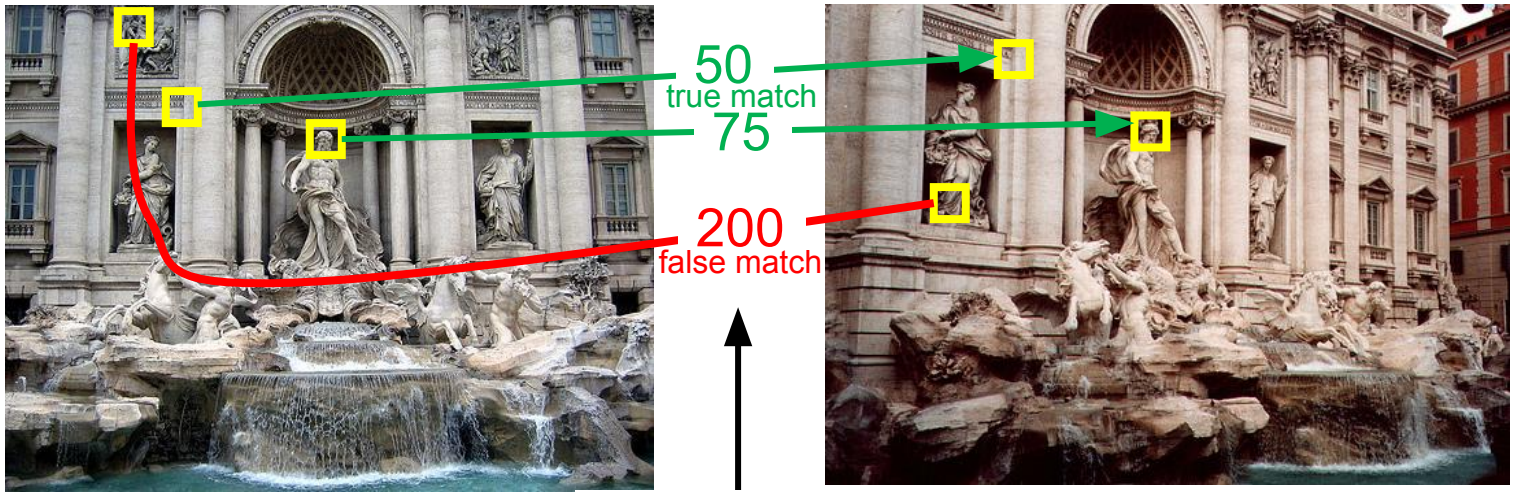


I_2

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

¿Como podemos medir el **performance** de un feature matcher?



feature distance

Aplicando un **threshold** afecta el performance

- **True positives** = # matched detectados que son correctos
- **False positives** = # maches detectados que son incorrectos

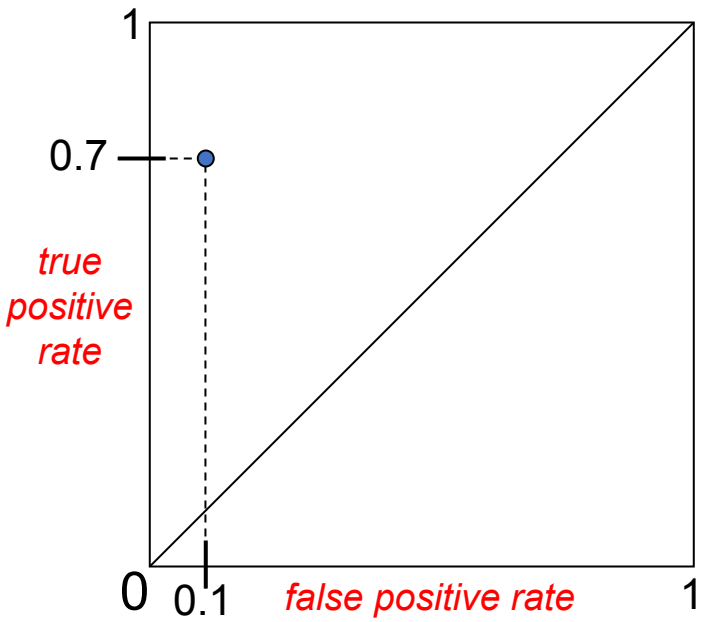
Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

¿Como podemos medir el **performance** de un feature matcher?

$$\frac{\text{\# true positives}}{\text{\# correctly matched features (positives)}}$$

"recall"



$$\frac{\text{\# false positives}}{\text{\# incorrectly matched features (negatives)}}$$

1 - "precision"

Sum of Squared Differences

Otras opciones existen

Normalized Cross-Correlation

Ver ejemplo en libro
Computer Vision with Python (pag 32)

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

¿Como podemos medir el **performance** de un feature matcher?

$$\frac{\text{\# true positives}}{\text{\# correctly matched features (positives)}}$$

"recall"

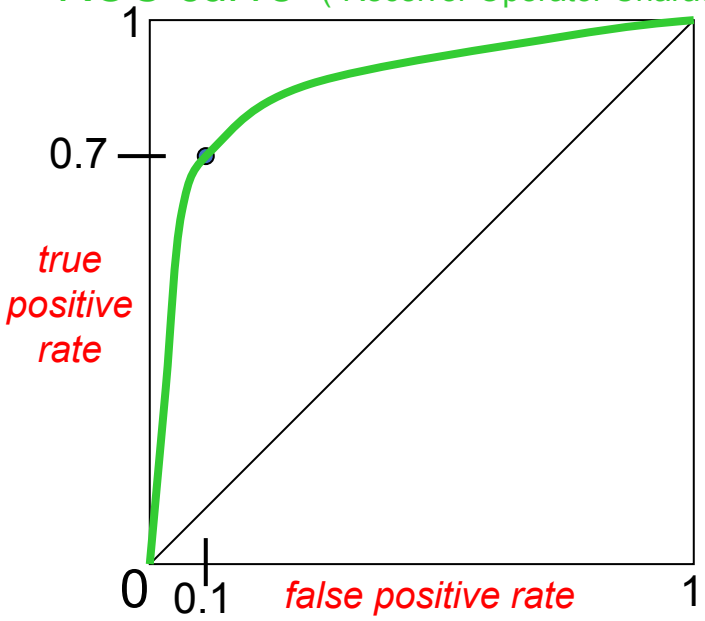
Sum of Squared Differences

Otras opciones existen

Normalized Cross-Correlation

Ver ejemplo en libro
Computer Vision with Python (pag 32)

ROC curve ("Receiver Operator Characteristic")



$$\frac{\text{\# false positives}}{\text{\# incorrectly matched features (negatives)}}$$

1 - "precision"

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

Detector	Illumination	Rotation	Scale	View point
Lowe '99 (DoG)	Yes	Yes	Yes	No
Harris corner	Yes	Yes	No	No
Mikolajczyk & Schmid '01, '02	Yes	Yes	Yes	Yes
Tuytelaars, '00	Yes	Yes	No (Yes '04)	Yes
Kadir & Brady, 01	Yes	Yes	Yes	no
Matas, '02	Yes	Yes	Yes	no

Los detectores son importantes para encontrar zonas de interes pero no son informativos se necesitan descriptores locales (proxima clase

Image Analysis for Computer vision

Keypoint Detection + Local Descriptors

SUSAN: Smallest Univalue Segment Assimilating Nucleus, Smith and Brady, 1997

SURF: Speeded Up Robust Features, Tuytelars & Van Gool, 2005

GLOH: Gradient Location and Orientation Histogram, K. Mikolajczyk, 2005

FAST (corner detector): A machine learning approach to corner detection, 2006

ORB: An efficient alternative to SIFT or SURF, Rublee, 2011

FREAK: Fast Retina Keypoint, Alahi, 2012

BRISK: Binary Robust Invariant Keypoint, 2013

¡Presentacion 5ta clase!

LIFT: Learned Invariant Feature Transform, Fua 2016

SuperPoint: Self Supervised Interest Point Detection, 2018

Copyright 2020. Tecnologico de Monterrey.
The total or partial reproduction of this work is
prohibited without explicit authorization of Tecnologico
de Monterrey