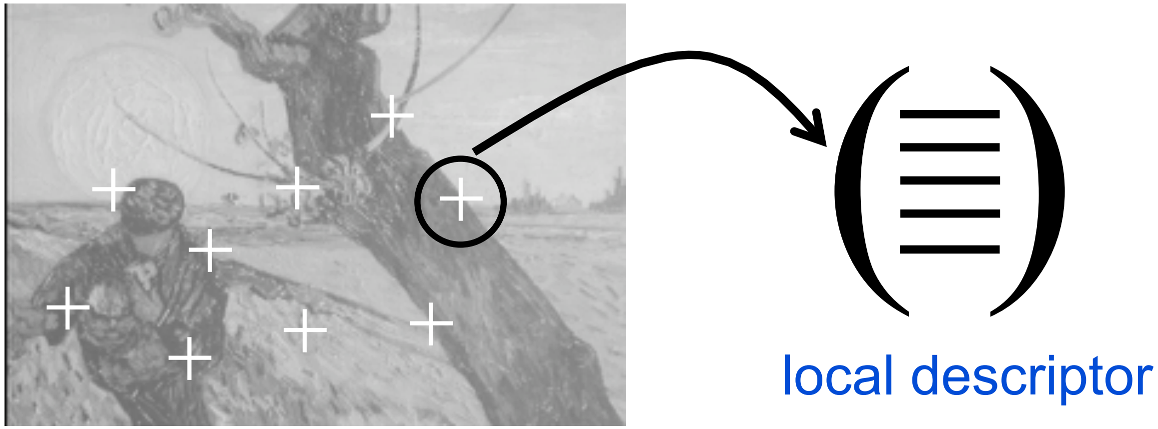
Local invariant features

Cordelia Schmid
INRIA, Grenoble

Local features



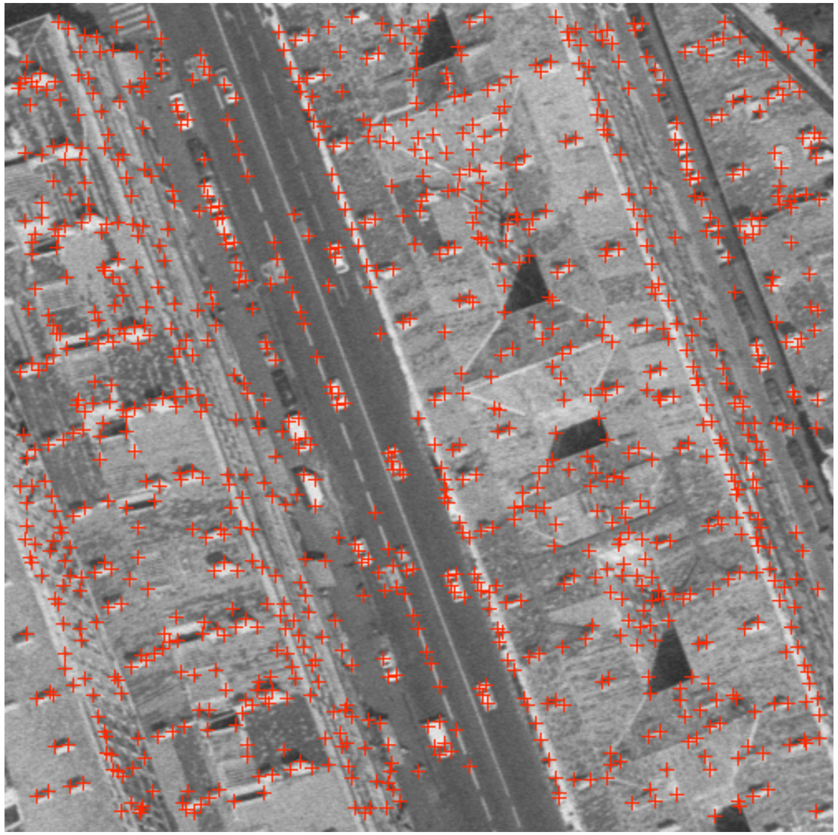
Several / many local descriptors per image

Robust to occlusion/clutter + no object segmentation required

Photometric : distinctive

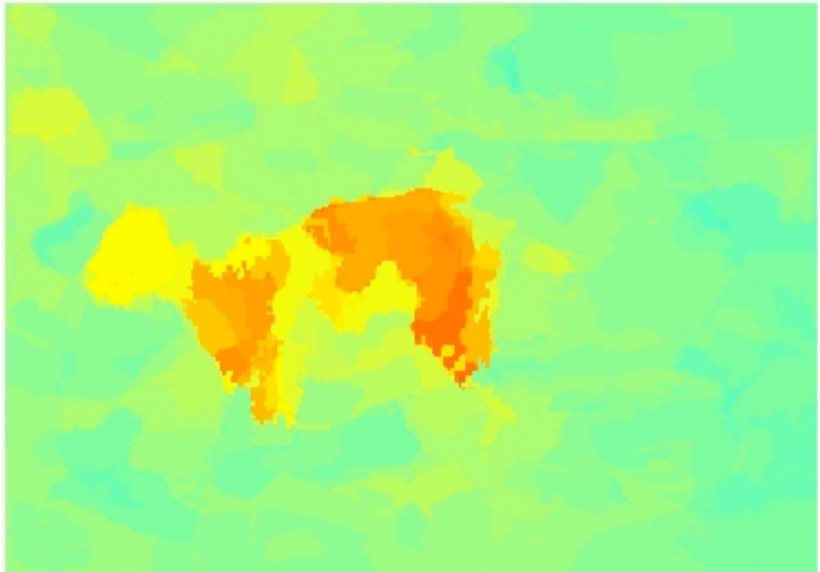
Invariant : to image transformations + illumination changes

Local features: interest points

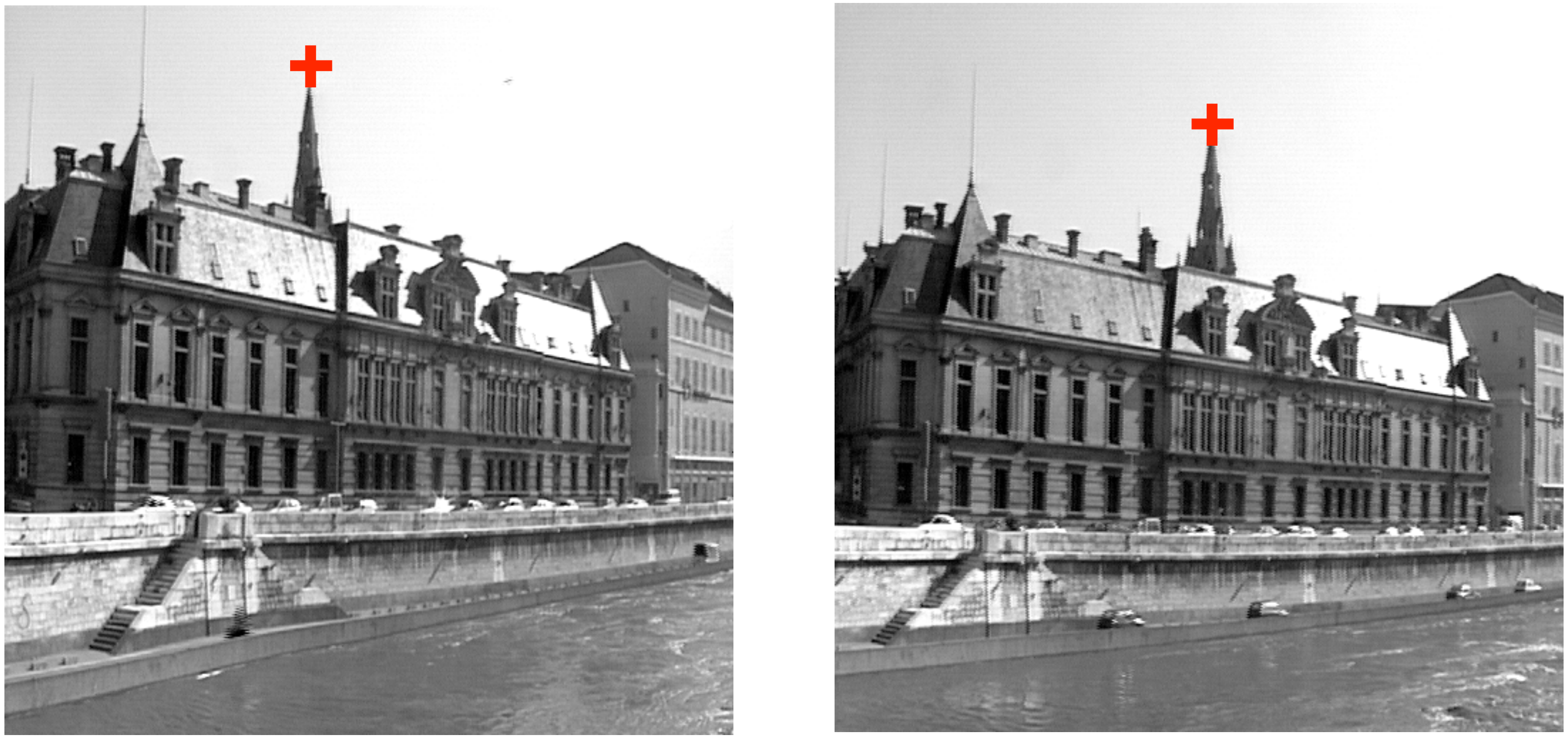


Local features: Contours/segments

Local features: segmentation

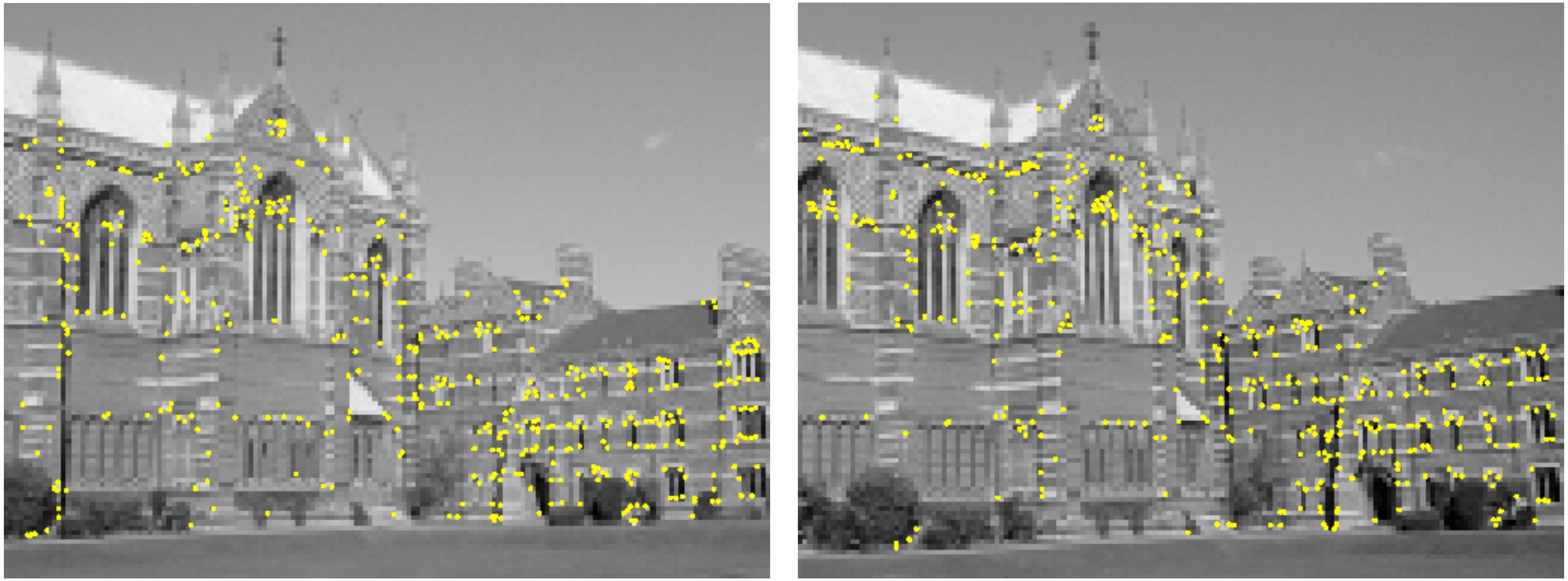


Application: Matching



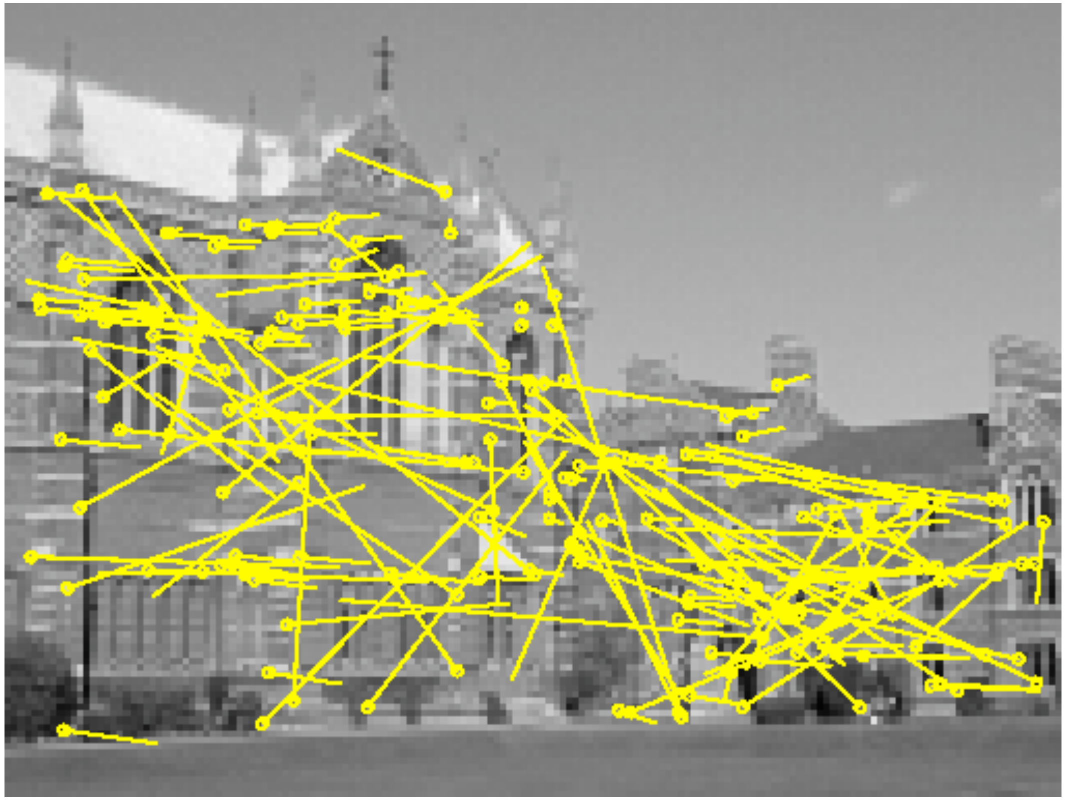
Find corresponding locations in the image

Illustration – Matching



Interest points extracted with Harris detector (~ 500 points)

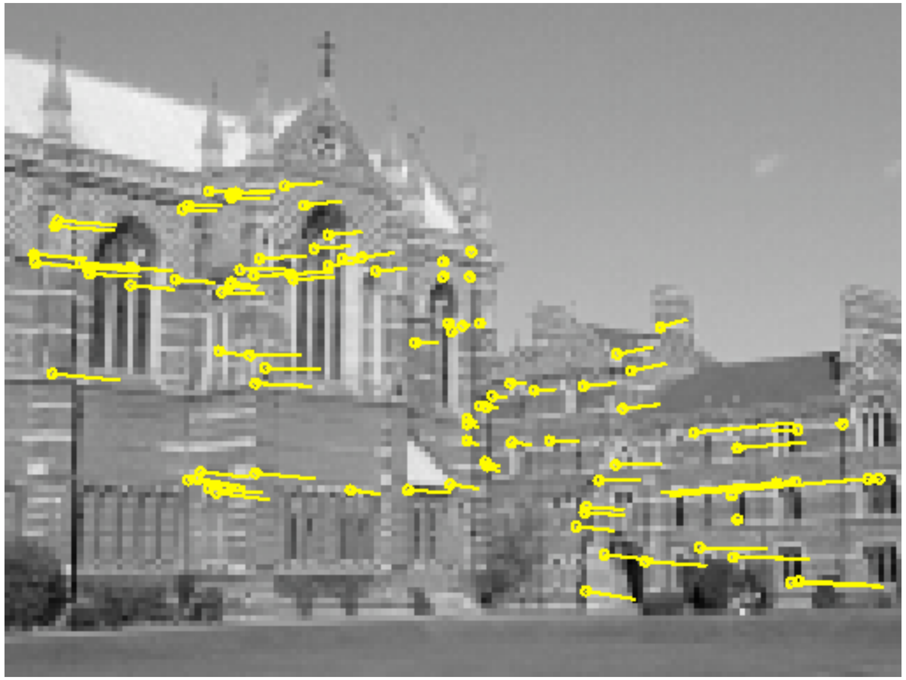
Illustration – Matching



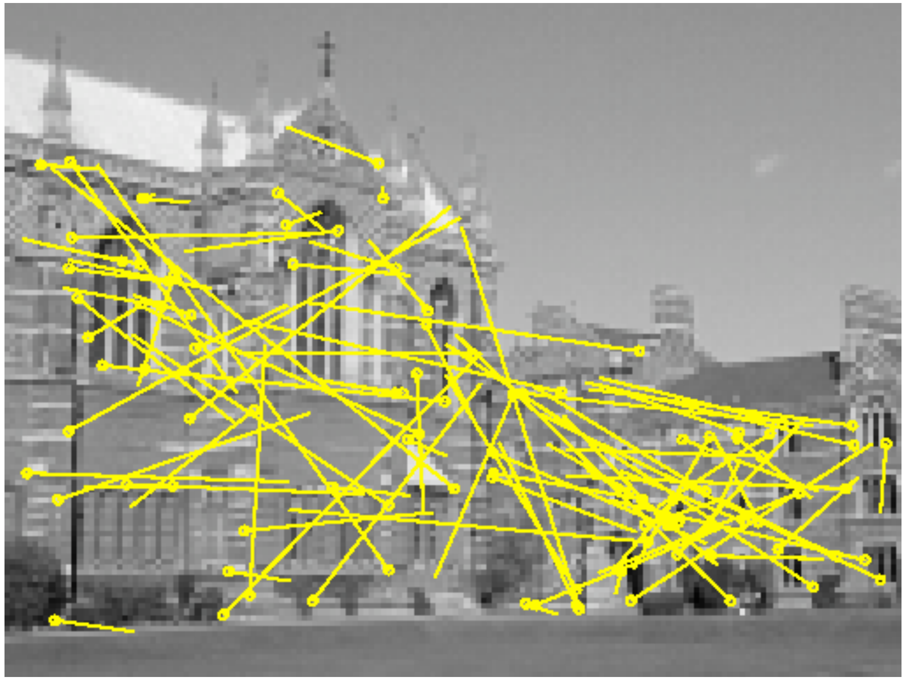
Interest points matched based on cross-correlation (188 pairs)

Illustration – Matching

Global constraint - Robust estimation of the fundamental matrix

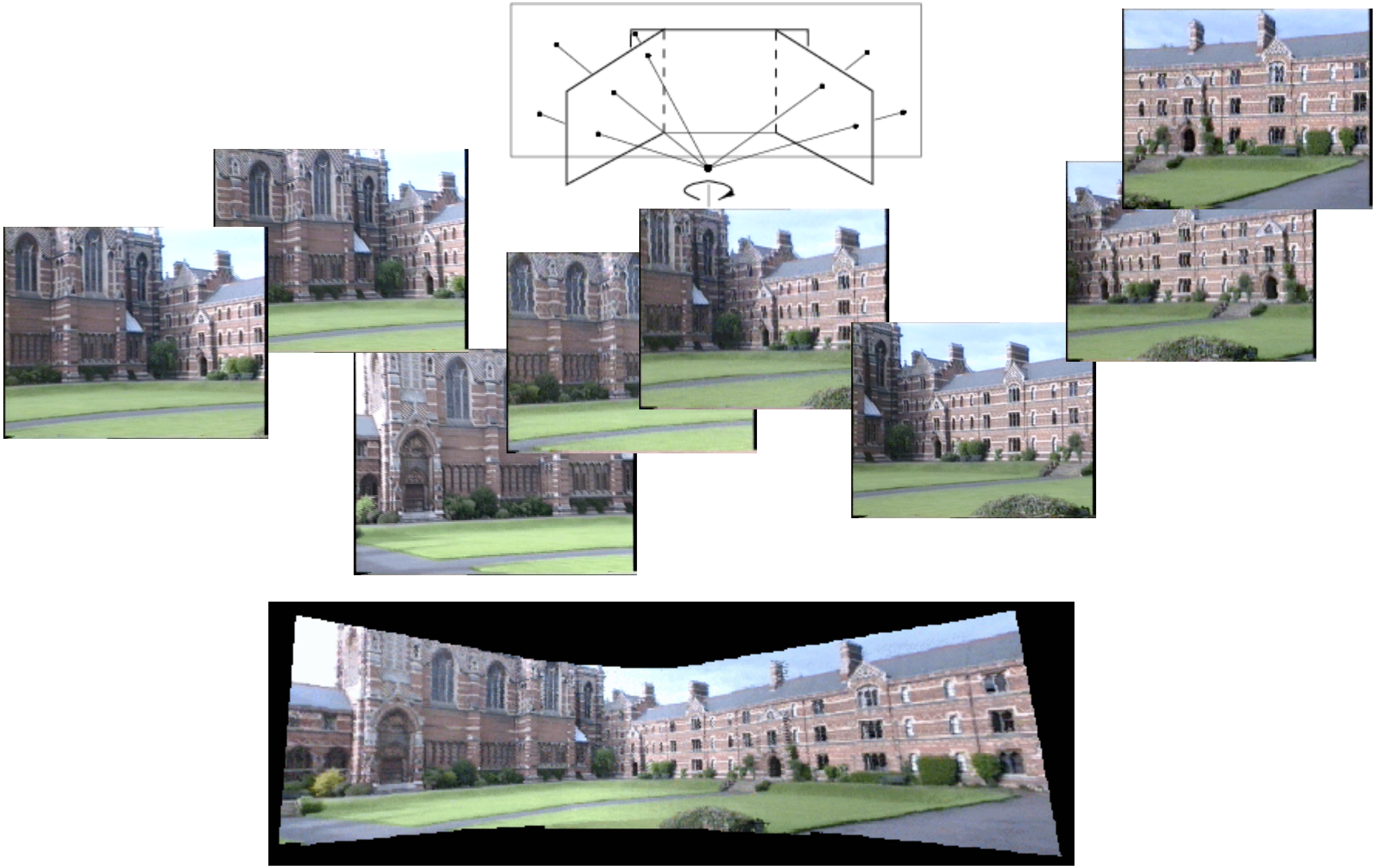


99 inliers



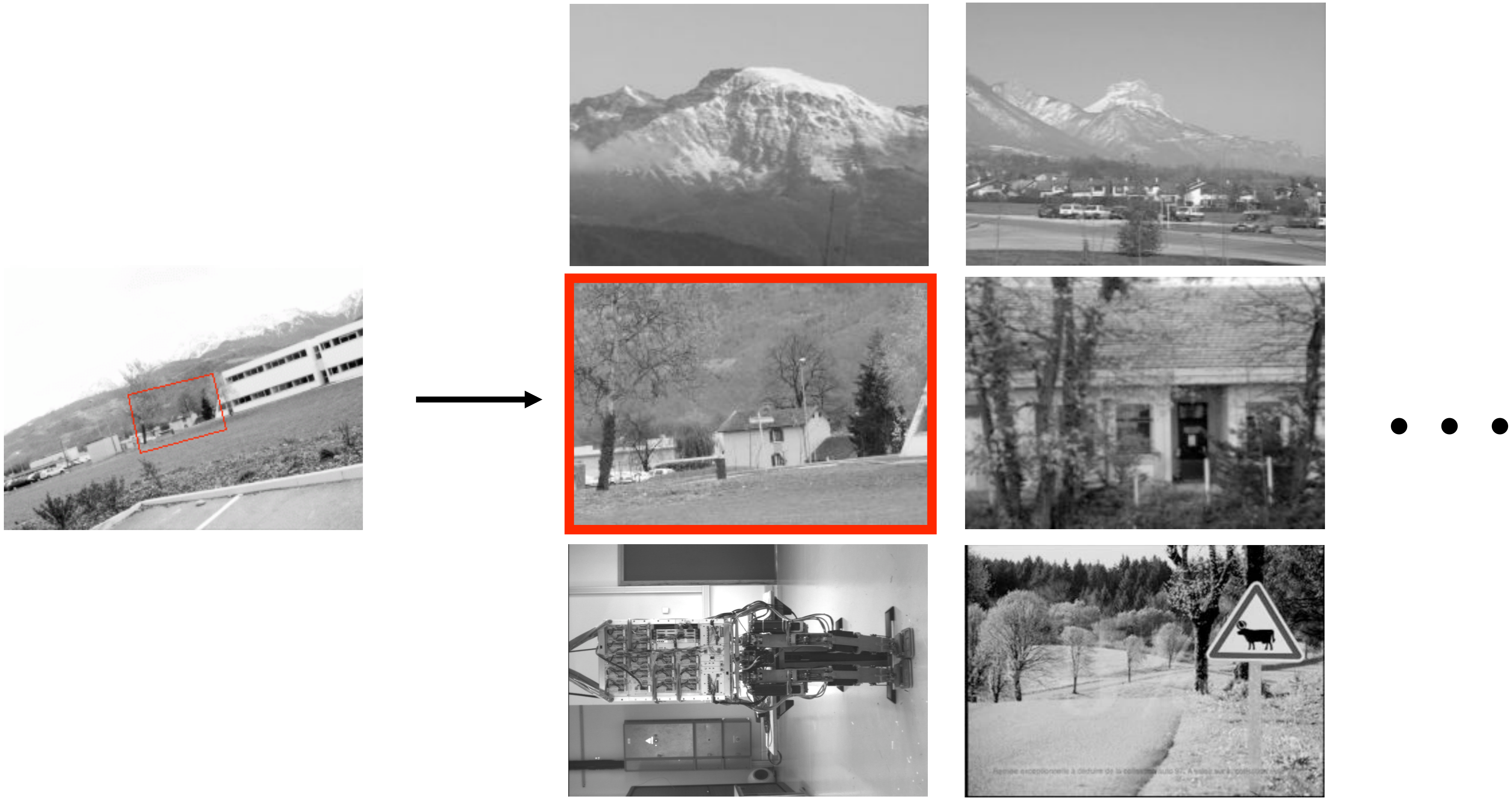
89 outliers

Application: Panorama stitching



Application: Instance-level recognition

Search for particular objects and scenes in large databases

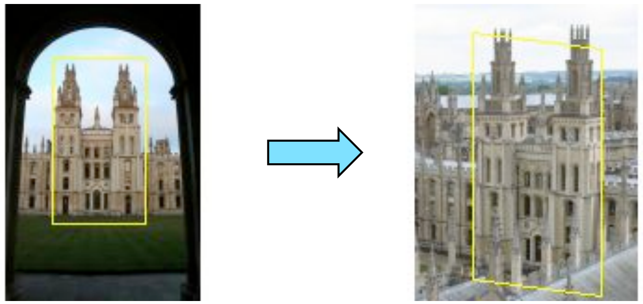


Difficulties

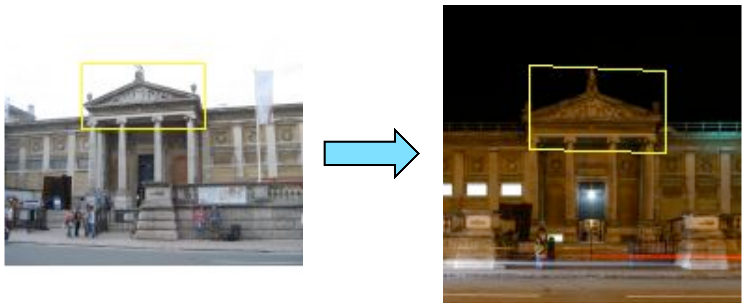
Finding the object despite possibly large changes in scale, viewpoint, lighting and partial occlusion

→ **requires invariant description**

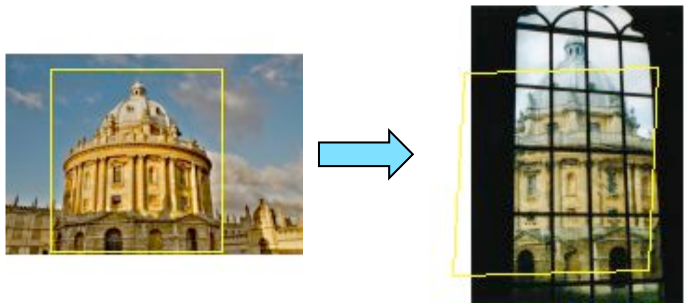
Scale



Viewpoint



Lighting



Occlusion

Difficulties

- Very large images collection → need for efficient indexing
  - Flickr has 2 billion photographs, more than 1 million added daily
  - Facebook has 15 billion images (~27 million added daily)
  - Large personal collections
  - Video collections, i.e., YouTube

Applications

Search photos on the web for particular places

Find these landmarks ...in these images and 1M more

Applications

- Take a picture of a product or advertisement
  → find relevant information on the web

[Pixee – Milpix]

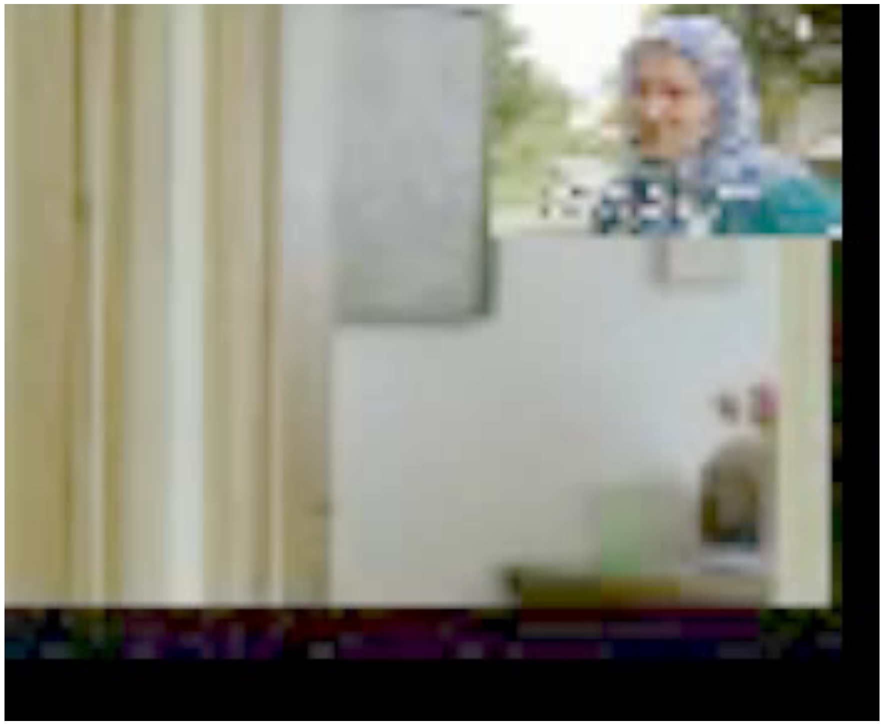
Applications

- Finding stolen/missing objects in a large collection

Applications

- Copy detection for images and videos

Query video



Search in 200h of video

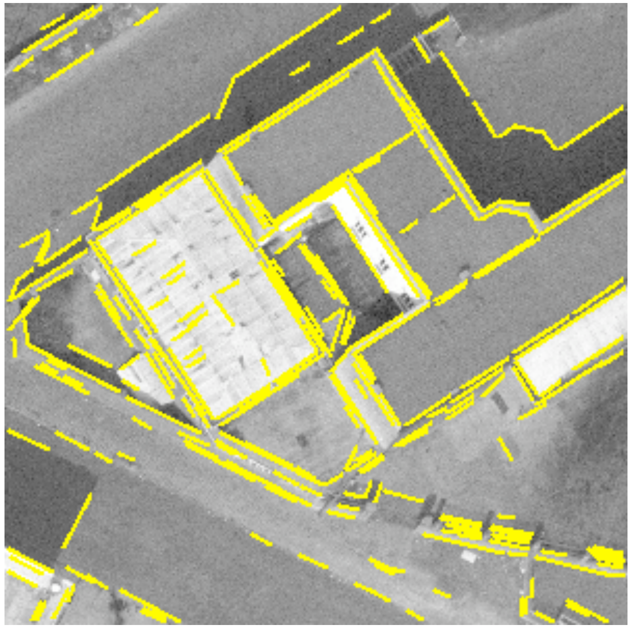
Local features - history

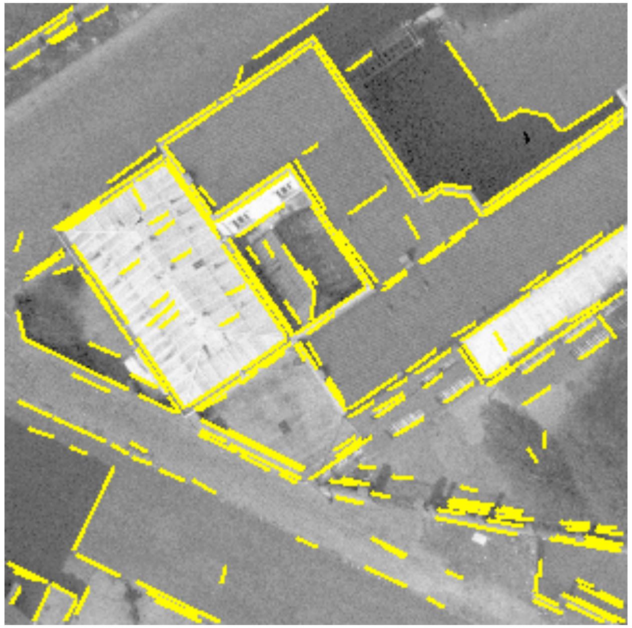
- Line segments [Lowe'87, Ayache'90]
- Interest points & cross correlation [Z. Zhang et al. 95]
- Rotation invariance with differential invariants [Schmid&Mohr'96]
- Scale & affine invariant detectors [Lindeberg'98, Lowe'99, Tuytelaars&VanGool'00, Mikolajczyk&Schmid'02, Matas et al.'02]
- Dense detectors and descriptors [Leung&Malik'99, Fei-Fei& Perona'05, Lazebnik et al.'06]
- Contour and region (segmentation) descriptors [Shotton et al.'05, Opelt et al.'06, Ferrari et al.'06, Leordeanu et al.'07]

Example for line segments

images 600 x 600

Example for line segments



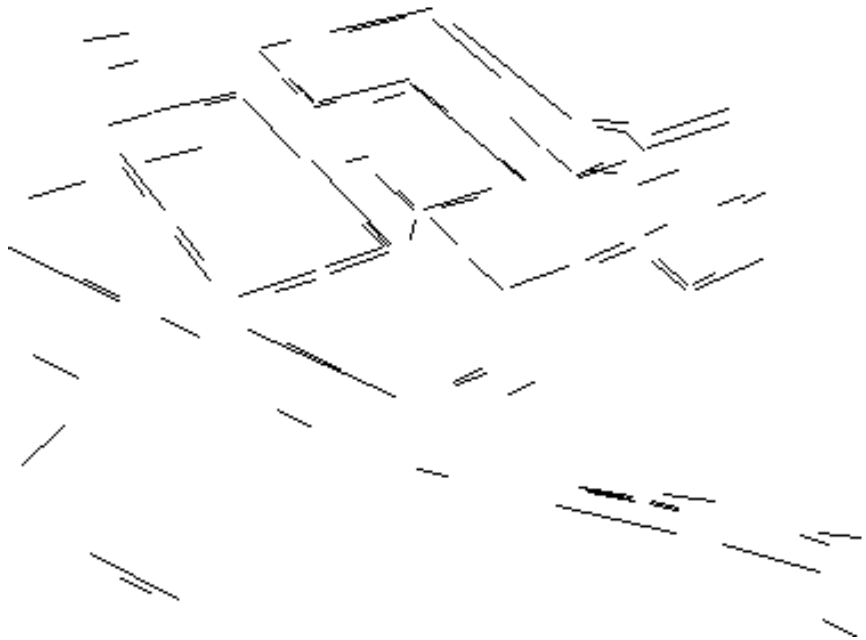


248 / 212 line segments extracted

Matched line segments


89 matched line segments - 100% correct

3D reconstruction of line segments



Problems of line segments

- Often only partial extraction
  - Line segments broken into parts
  - Missing parts
- Information not very discriminative
  - 1D information
  - Similar for many segments
- Potential solutions
  - Pairs and triplets of segments
  - **Interest points**

Overview

- **Harris interest points**
- Comparing interest points
- Scale & affine invariant interest points
- Evaluation and comparison of different detectors
- Region descriptors and their performance

Harris detector [Harris & Stephens'88]

Based on the idea of auto-correlation

Important difference in all directions => interest point

Images

- We can think of an **image** as a function, *f*, from R^2 to R:
  - $f(x, y)$ gives the **intensity** at position (x, y)
  - the image is defined over a rectangle with a finite range
- A color image is just three functions pasted together. We can write this as a "vector-valued" function:

$$f(x,y) = \begin{bmatrix} r(x,y) \\ g(x,y) \\ b(x,y) \end{bmatrix}$$

Digital images

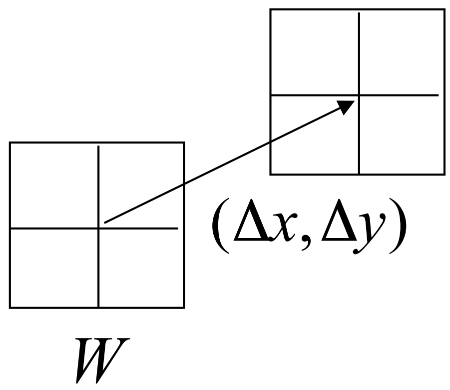
- In computer vision we operate on **digital** images:
  - **Sample** the 2D space on a regular grid
  - **Quantize** each sample (round to nearest integer)
- The image can now be represented as a matrix of integer values (pixels)

62	79	23	119	120	105	4	0
10	10	9	62	12	78	34	0
10	58	197	46	46	0	0	48
176	135	5	188	191	68	0	49
2	1	1	29	26	37	0	77
0	89	144	147	187	102	62	208
255	252	0	166	123	62	0	31
166	63	127	17	1	0	99	30

Harris detector

Auto-correlation function for a point (x, y) and a shift $(\Delta x, \Delta y)$

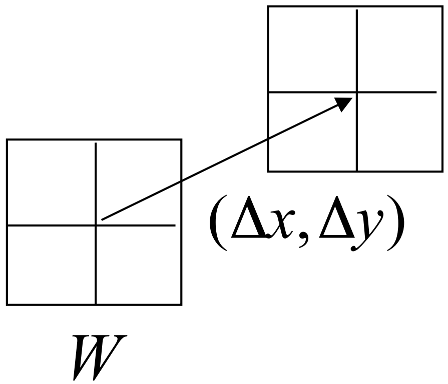
$$a(x, y) = \sum_{(x_k, y_k) \in W(x, y)} \left(I(x_k, y_k) - I(x_k + \Delta x, y_k + \Delta y) \right)^2$$



Harris detector

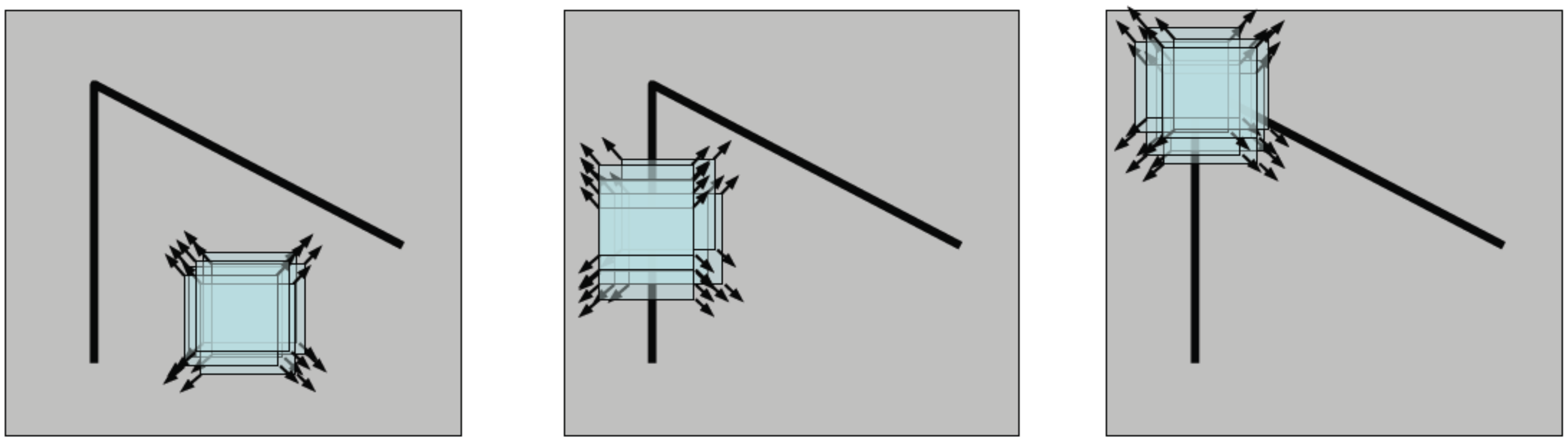
Auto-correlation function for a point (x, y) and a shift $(\Delta x, \Delta y)$

$$a(x,y) = \sum_{(x_k, y_k) \in W(x,y)} \left(I(x_k, y_k) - I(x_k + \Delta x, y_k + \Delta y) \right)^2$$



$a(x,y)$
- small in all directions → uniform region
- large in one directions → contour
- large in all directions → interest point

Harris detector



"flat" region:
no change in
all directions

"edge":
no change along
the edge direction

"corner":
significant change
in all directions

Harris detector

Discret shifts are avoided based on the auto-correlation matrix

with first order approximation

$$I(x_k + \Delta x, y_k + \Delta y) = I(x_k, y_k) + (I_x(x_k, y_k) \quad I_y(x_k, y_k))\begin{pmatrix}\Delta x \\ \Delta y\end{pmatrix}$$

$$a(x, y) = \sum_{(x_k, y_k) \in W(x,y)} (I(x_k, y_k) - I(x_k + \Delta x, y_k + \Delta y))^2$$

$$= \sum_{(x_k, y_k) \in W} \left(\left(I_x(x_k, y_k) \quad I_y(x_k, y_k)\right)\begin{pmatrix}\Delta x \\ \Delta y\end{pmatrix} \right)^2$$

Harris detector

$$= \begin{pmatrix}\Delta x & \Delta y\end{pmatrix} \begin{bmatrix} \sum_{(x_k,y_k)\in W} (I_x(x_k,y_k))^2 & \sum_{(x_k,y_k)\in W} I_x(x_k,y_k) I_y(x_k,y_k) \\ \sum_{(x_k,y_k)\in W} I_x(x_k,y_k) I_y(x_k,y_k) & \sum_{(x_k,y_k)\in W} (I_y(x_k,y_k))^2 \end{bmatrix} \begin{pmatrix}\Delta x \\ \Delta y\end{pmatrix}$$

Auto-correlation matrix

the sum can be smoothed with a Gaussian

$$= \begin{pmatrix}\Delta x & \Delta y\end{pmatrix} G \otimes \begin{bmatrix} I_x^{\,2} & I_x I_y \\ I_x I_y & I_y^{\,2} \end{bmatrix} \begin{pmatrix}\Delta x \\ \Delta y\end{pmatrix}$$

Harris detector

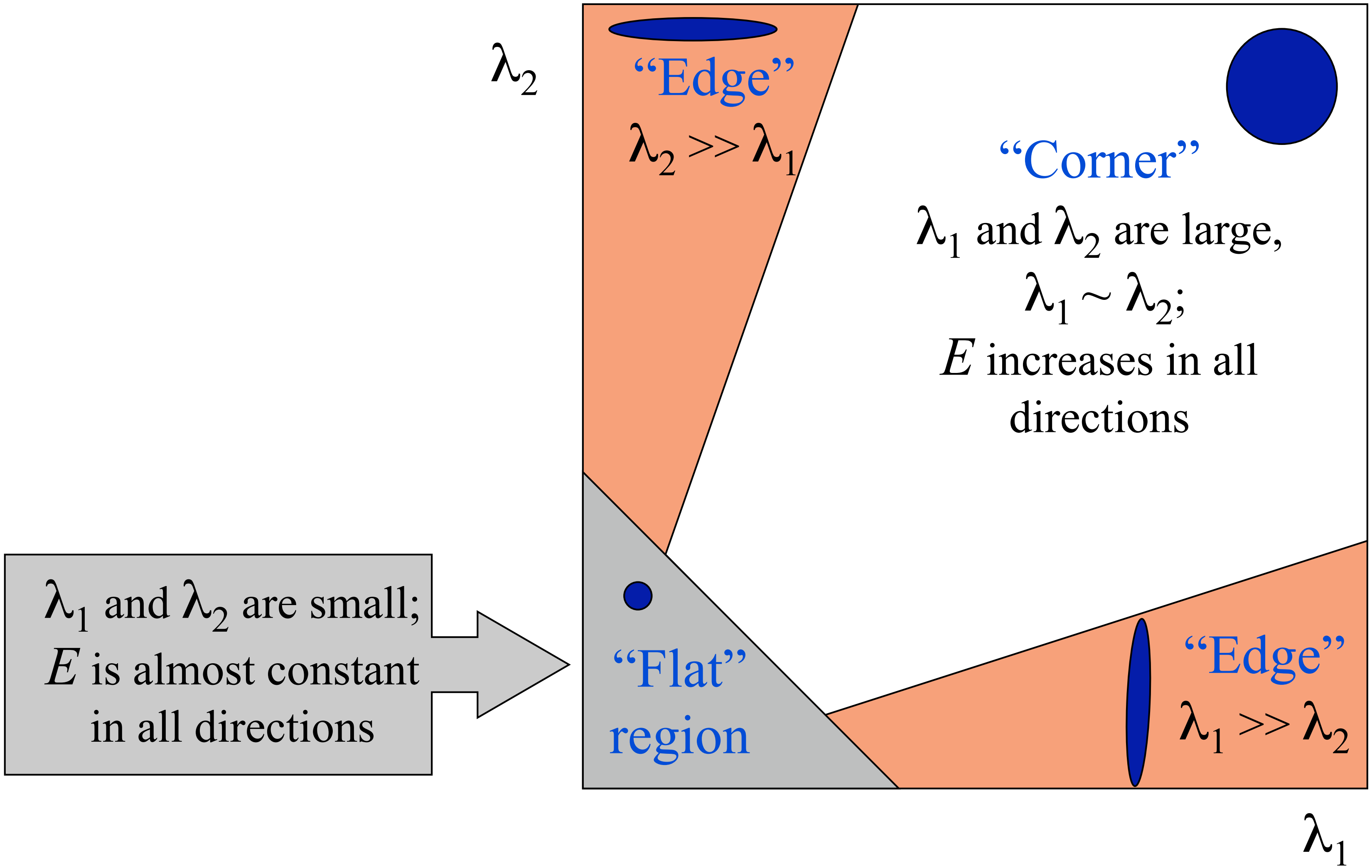
- Auto-correlation matrix

$$G \otimes \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

  - captures the structure of the local neighborhood
  - measure based on eigenvalues of this matrix
    - 2 strong eigenvalues => interest point
    - 1 strong eigenvalue => contour
    - 0 eigenvalue => uniform region

Interpreting the eigenvalues

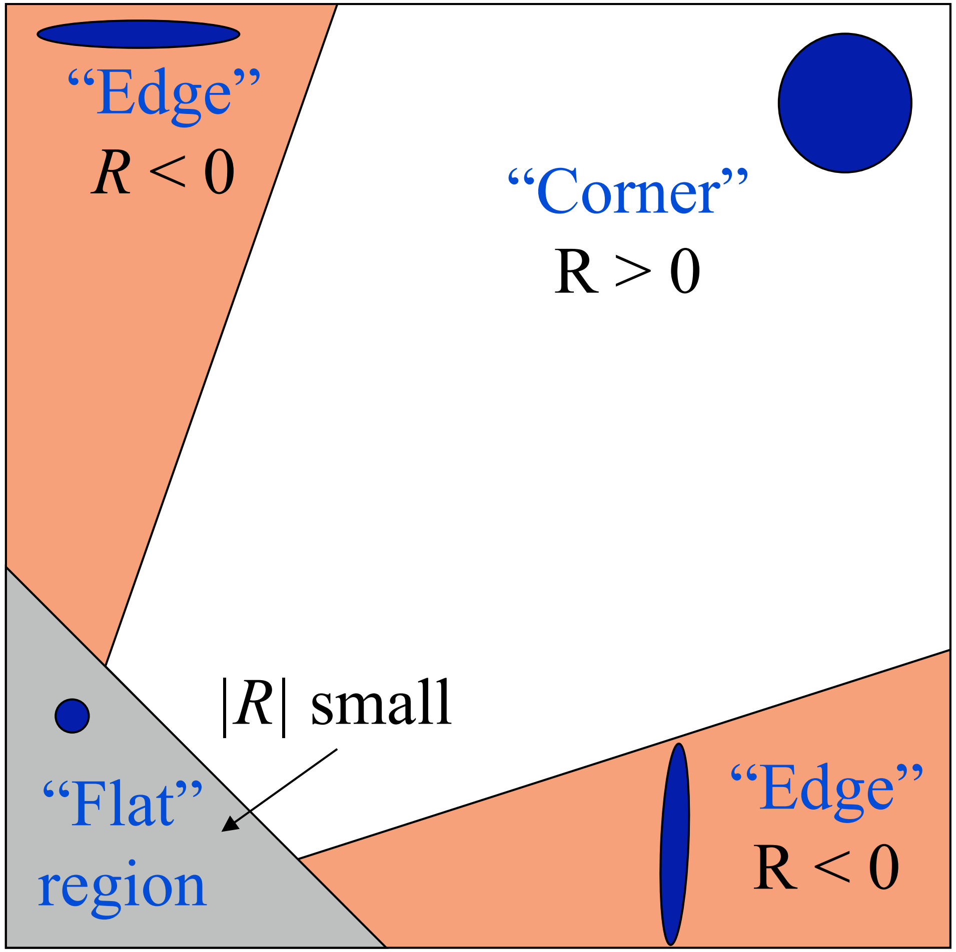
Classification of image points using eigenvalues of *M*:



Corner response function

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

α: constant (0.04 to 0.06)



Harris detector

- Cornerness function

$$f = \det(a) - k(trace(a))^2 = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2$$

Reduces the effect of a strong contour

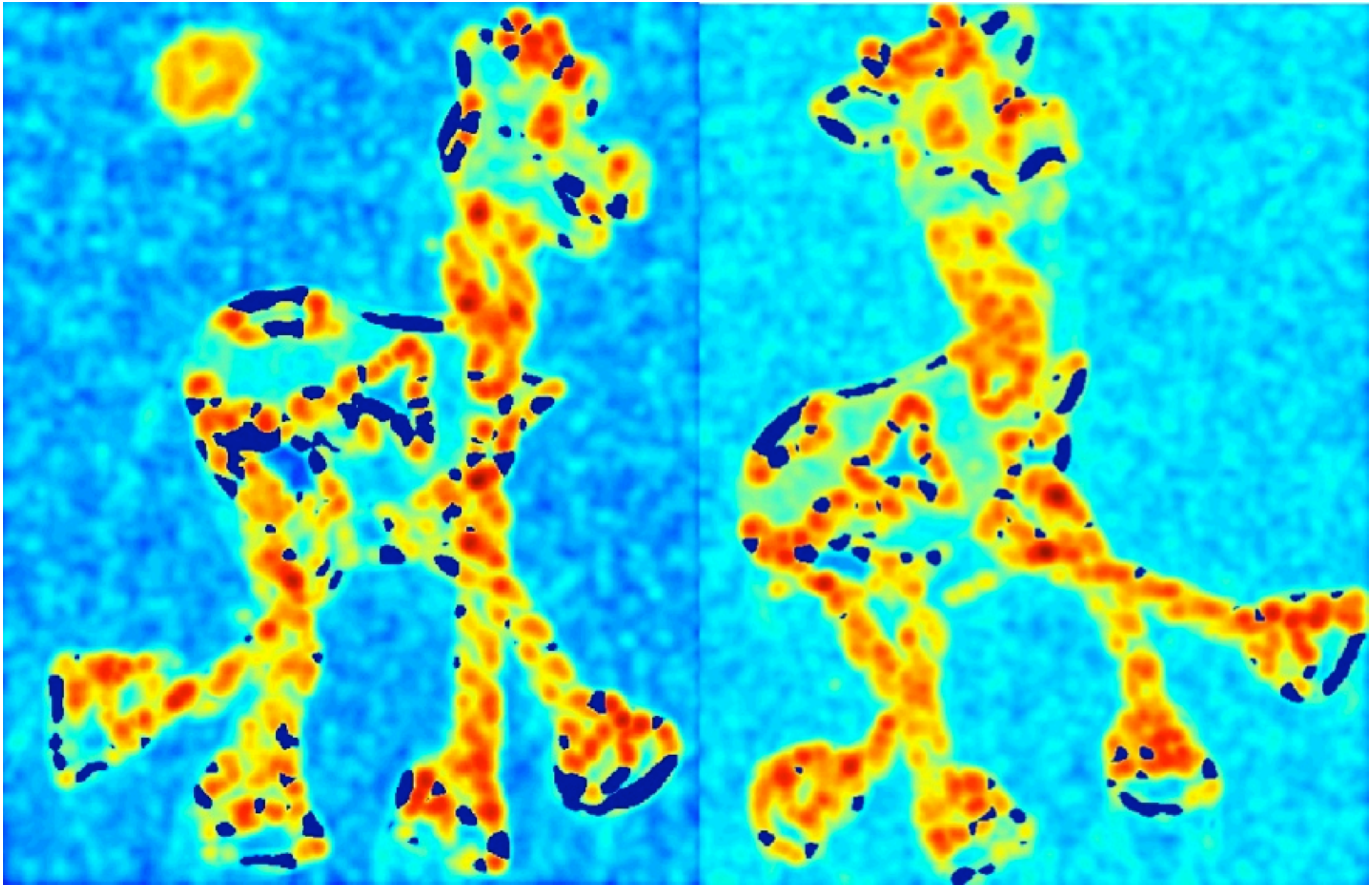
- Interest point detection
  - Treshold (absolut, relatif, number of corners)
  - Local maxima

$$f > thresh \;\wedge\; \forall x, y \in 8-neighbourhood \;\; f(x, y) \geq f(x', y')$$

Harris Detector: Steps

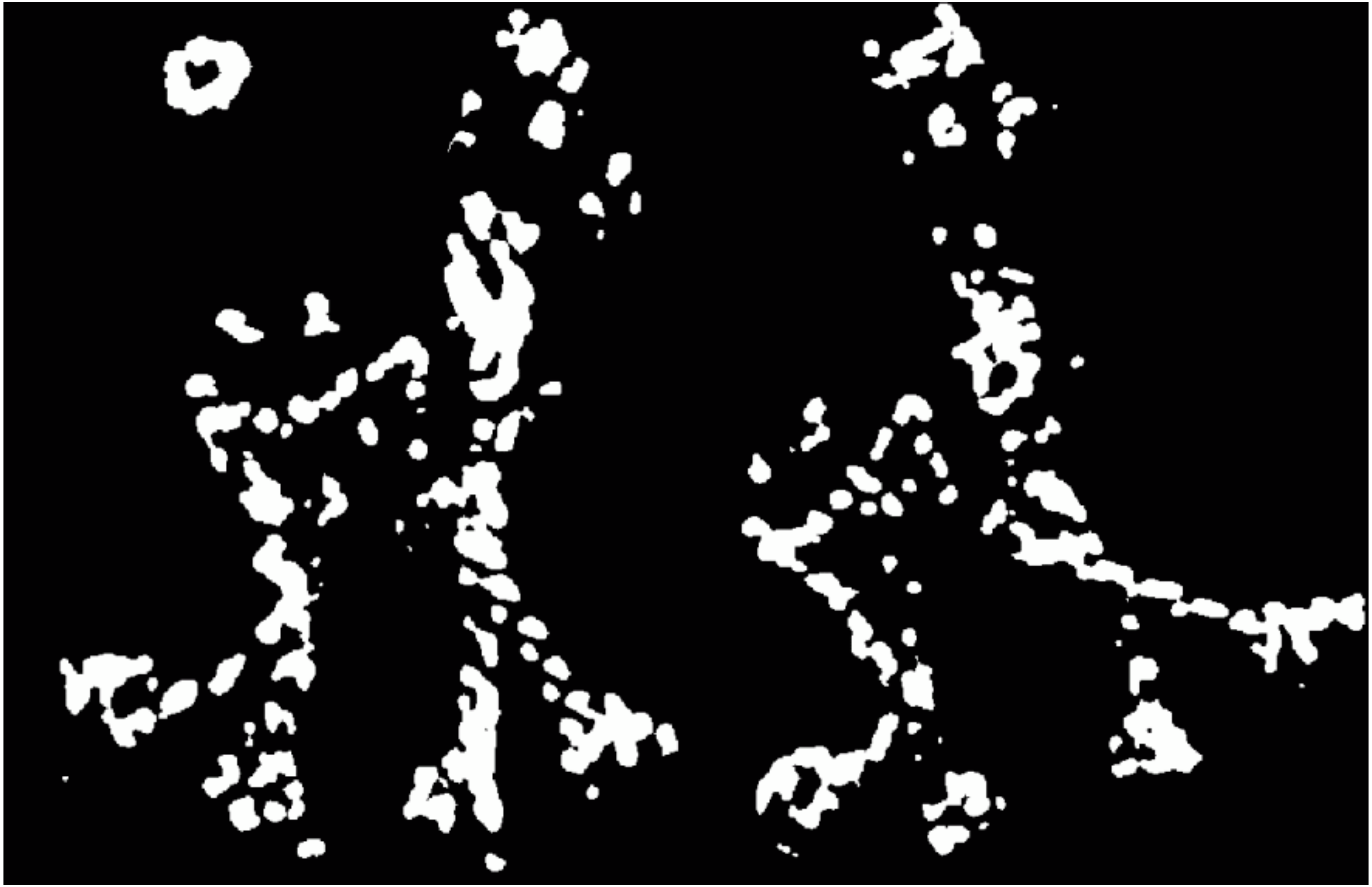
Harris Detector: Steps

Compute corner response *R*



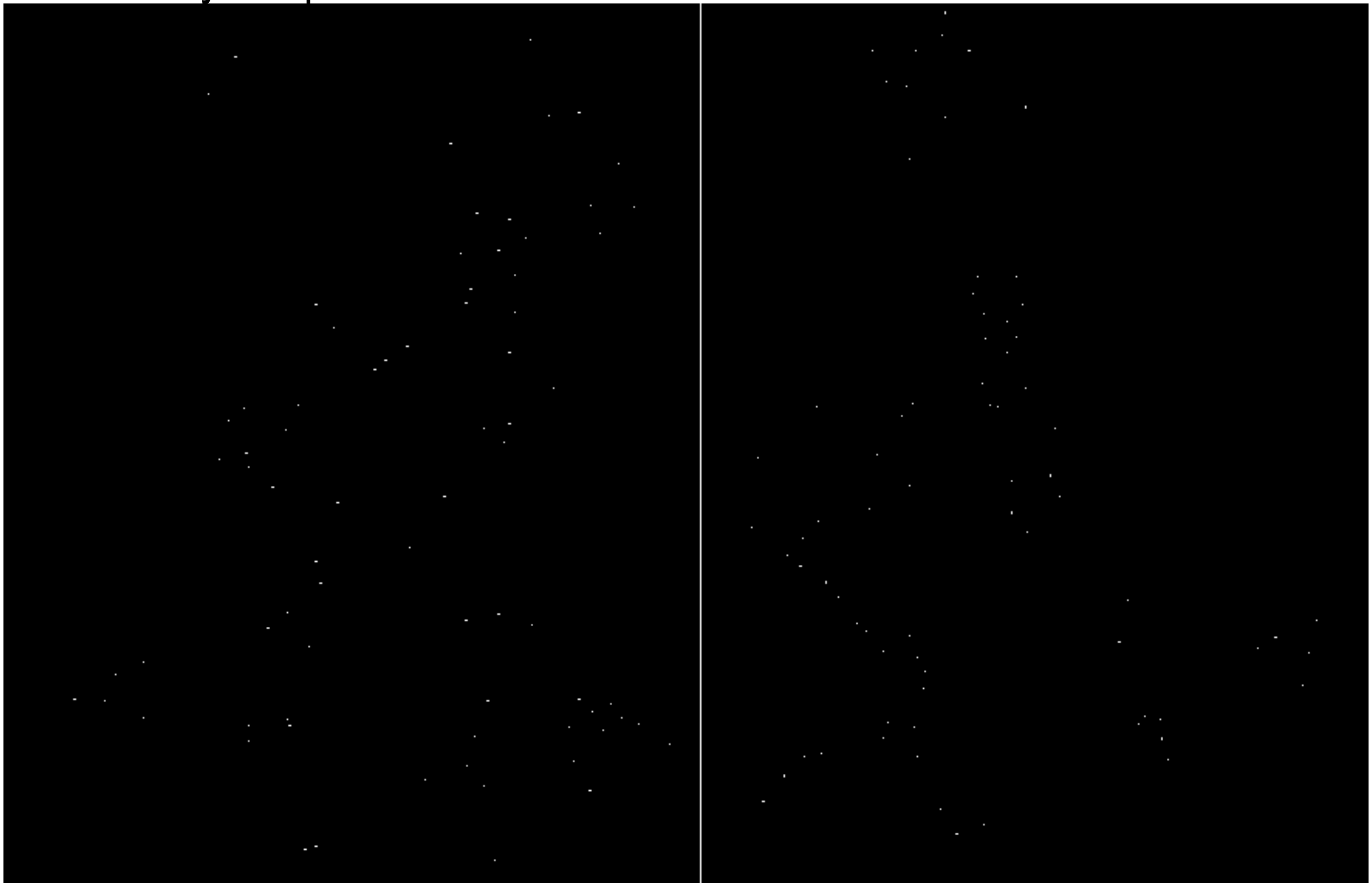
Harris Detector: Steps

Find points with large corner response: R>threshold

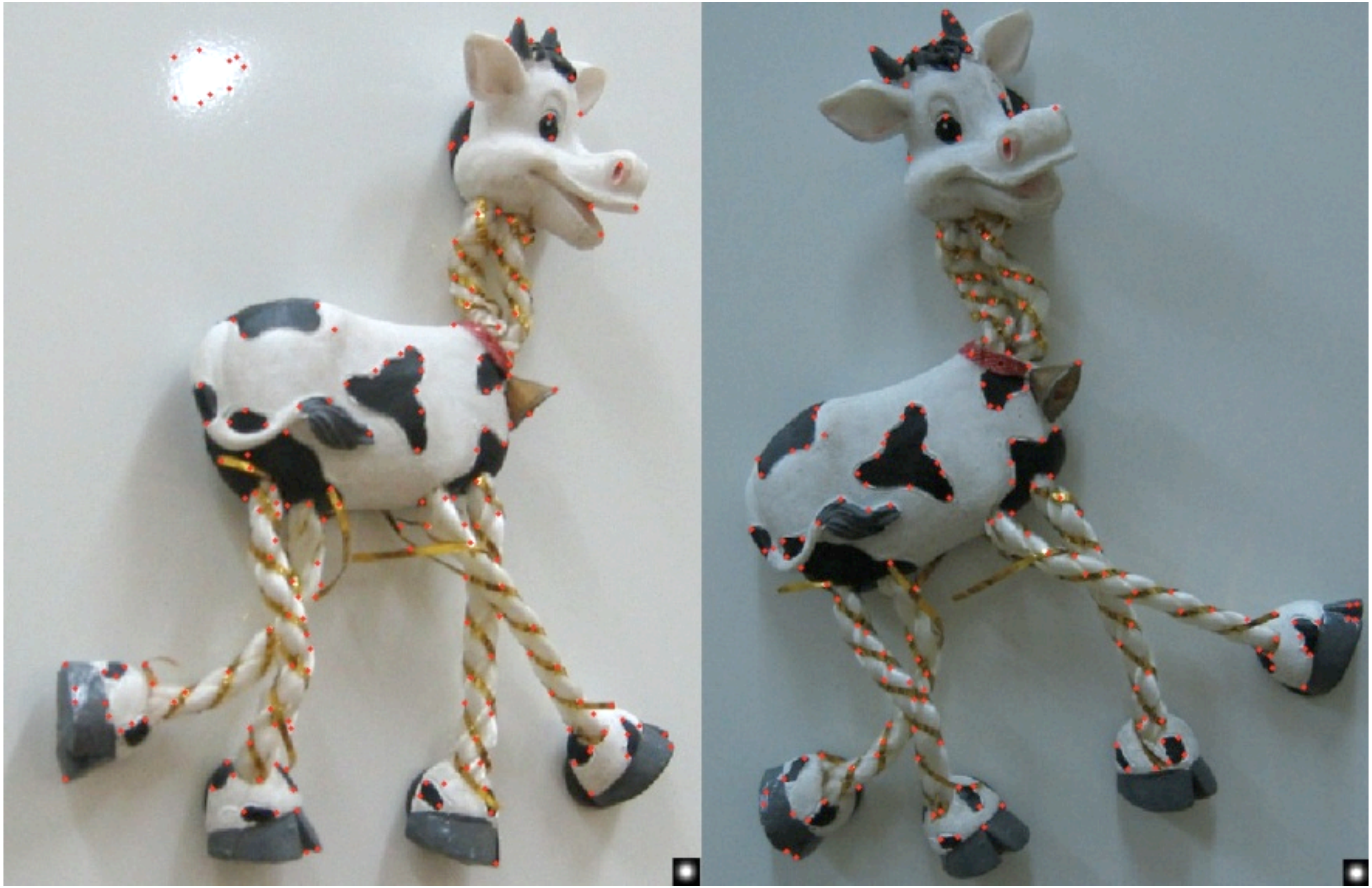


Harris Detector: Steps

Take only the points of local maxima of *R*



Harris Detector: Steps

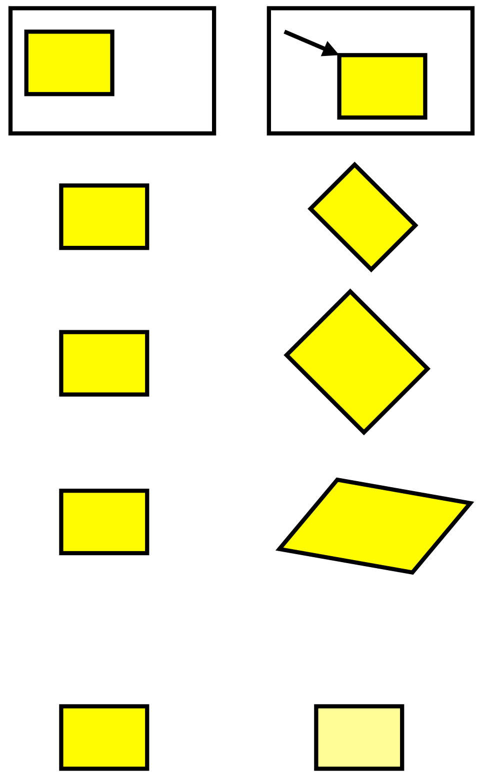


Harris detector: Summary of steps

1. Compute Gaussian derivatives at each pixel
2. Compute second moment matrix M in a Gaussian window around each pixel
3. Compute corner response function R
4. Threshold R
5. Find local maxima of response function (non-maximum suppression)

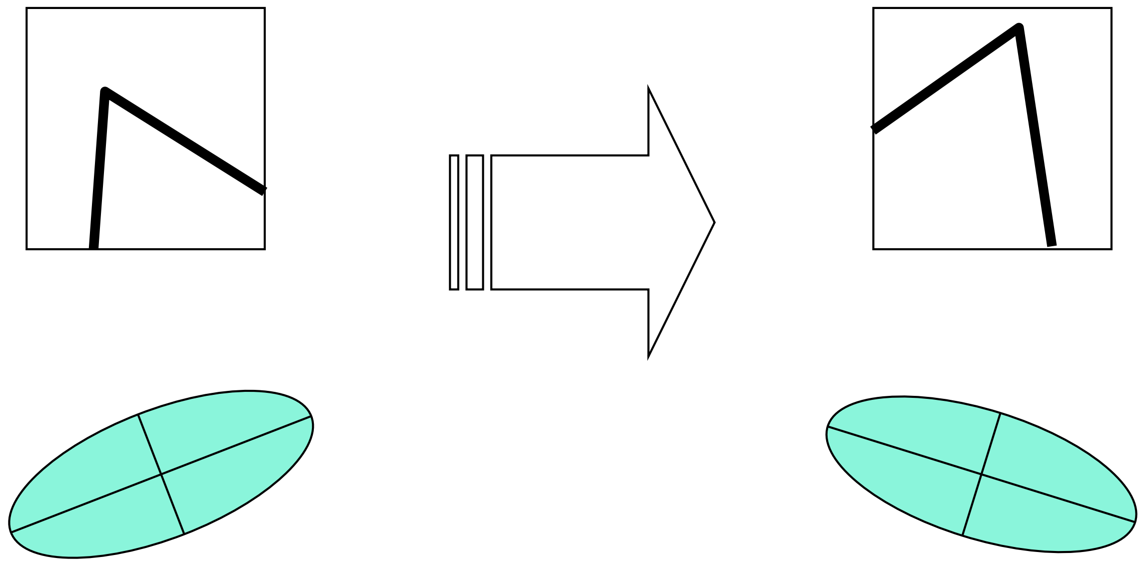
Harris - invariance to transformations

- Geometric transformations
  - translation
  - rotation
  - similitude (rotation + scale change)
  - affine (valide for local planar objects)
- Photometric transformations
  - Affine intensity changes ($I \rightarrow a\,I + b$)



Harris Detector: Invariance Properties

- Rotation



Ellipse rotates but its shape (i.e. eigenvalues) remains the same

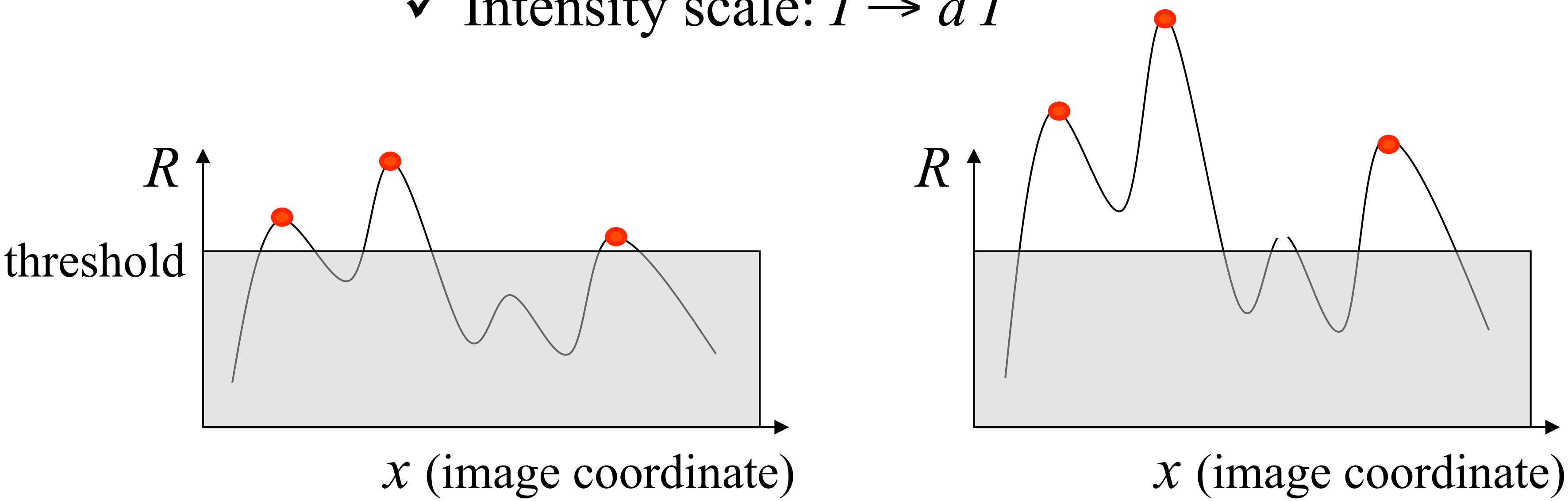
Corner response R is invariant to image rotation

Harris Detector: Invariance Properties

- Affine intensity change

  ✓ Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$

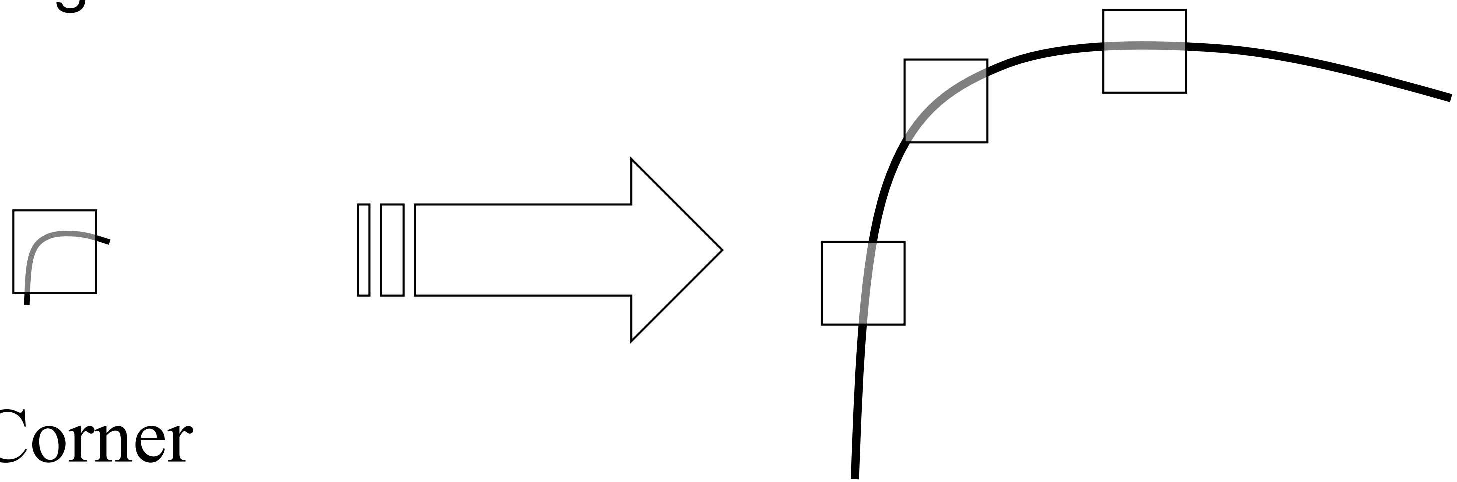
  ✓ Intensity scale: $I \rightarrow a\, I$



Partially invariant to affine intensity change, dependent on type of threshold

Harris Detector: Invariance Properties

- Scaling



Corner

All points will be classified as edges

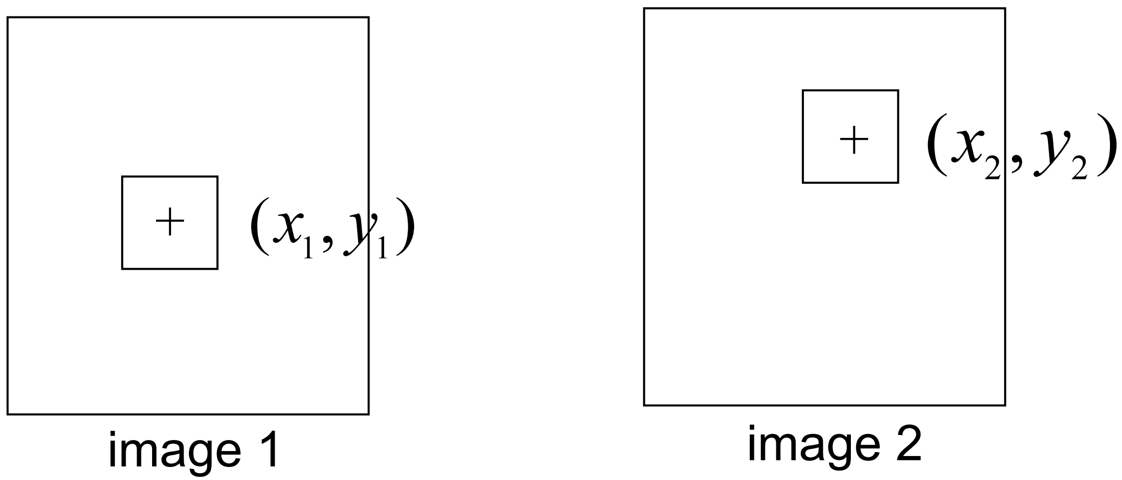
Not invariant to scaling

Overview

- Harris interest points
- **Comparing interest points (SSD, ZNCC, Derivatives, SIFT)**
- Scale & affine invariant interest points
- Evaluation and comparison of different detectors
- Region descriptors and their performance

Comparison of patches - SSD

Comparison of the intensities in the neighborhood of two interest points



SSD : sum of square difference

$$\frac{1}{(2N+1)^2}\sum_{i=-N}^{N}\sum_{j=-N}^{N}\left(I_1(x_1+i, y_1+j) - I_2(x_2+i, y_2+j)\right)^2$$

Small difference values → similar patches

Comparison of patches

SSD : $\frac{1}{(2N+1)^2}\sum_{i=-N}^{N}\sum_{j=-N}^{N}(I_1(x_1+i,y_1+j)-I_2(x_2+i,y_2+j))^2$

Invariance to photometric transformations?

Intensity changes (I → I + b)

=> Normalizing with the mean of each patch

$$\frac{1}{(2N+1)^2}\sum_{i=-N}^{N}\sum_{j=-N}^{N}((I_1(x_1+i,y_1+j)-m_1)-(I_2(x_2+i,y_2+j)-m_2))^2$$

Intensity changes (I → aI + b)

=> Normalizing with the mean and standard deviation of each patch

$$\frac{1}{(2N+1)^2}\sum_{i=-N}^{N}\sum_{j=-N}^{N}\left(\frac{I_1(x_1+i,y_1+j)-m_1}{\sigma_1}-\frac{I_2(x_2+i,y_2+j)-m_2}{\sigma_2}\right)^2$$

Cross-correlation ZNCC

zero normalized SSD

$$\frac{1}{(2N+1)^2}\sum_{i=-N}^{N}\sum_{j=-N}^{N}\left(\frac{I_1(x_1+i,y_1+j)-m_1}{\sigma_1}-\frac{I_2(x_2+i,y_2+j)-m_2}{\sigma_2}\right)^2$$

⬇

ZNCC: zero normalized cross correlation

$$\frac{1}{(2N+1)^2}\sum_{i=-N}^{N}\sum_{j=-N}^{N}\left(\frac{I_1(x_1+i,y_1+j)-m_1}{\sigma_1}\right)\cdot\left(\frac{I_2(x_2+i,y_2+j)-m_2}{\sigma_2}\right)$$

ZNCC values between -1 and 1, 1 when identical patches

in practice threshold around 0.5

Local descriptors

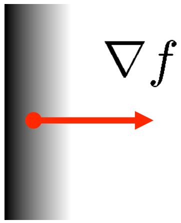
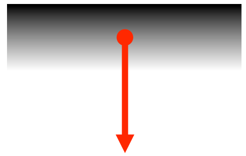
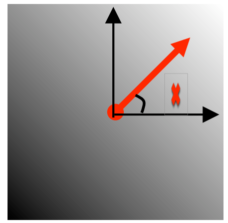
- Greyvalue derivatives

- Differential invariants [Koenderink'87]
  - combinations of derivatives

- SIFT descriptor [Lowe'99]

Greyvalue derivatives: Image gradient

- The gradient of an image: $$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

-  $\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$  $\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$  $\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$

- The gradient points in the direction of most rapid increase in intensity

- The gradient direction is given by $$\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$
  - how does this relate to the direction of the edge?

- The *edge strength* is given by the gradient magnitude

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Source: Steve Seitz

Differentiation and convolution

- Recall, for 2D function, f(x,y): $$\frac{\partial f}{\partial x} = \lim_{\varepsilon \to 0} \left(\frac{f(x+\varepsilon, y)}{\varepsilon} - \frac{f(x,y)}{\varepsilon} \right)$$
- We could approximate this as $$\frac{\partial f}{\partial x} \approx \frac{f(x_{n+1}, y) - f(x_n, y)}{\Delta x}$$
- Convolution with the filter

-1	**1**

Source: D. Forsyth, D. Lowe

Finite difference filters

- Other approximations of derivative filters exist:

Prewitt:

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad ; \quad M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Sobel:

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad ; \quad M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

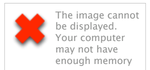
Roberts:

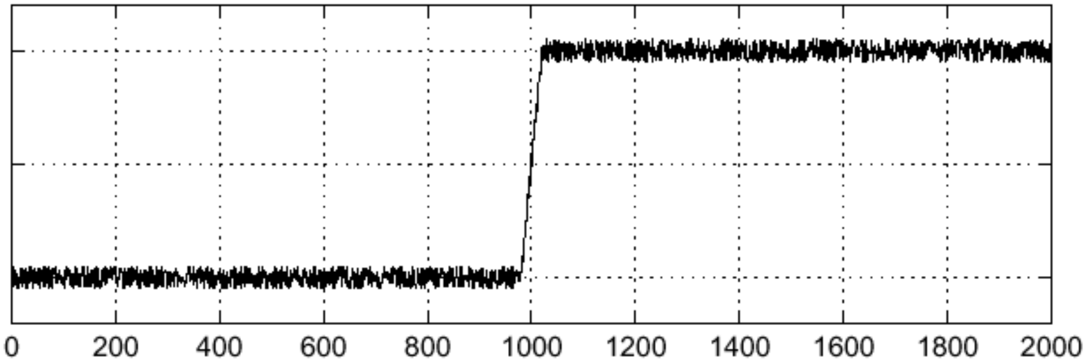
$$M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad ; \quad M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Source: K. Grauman

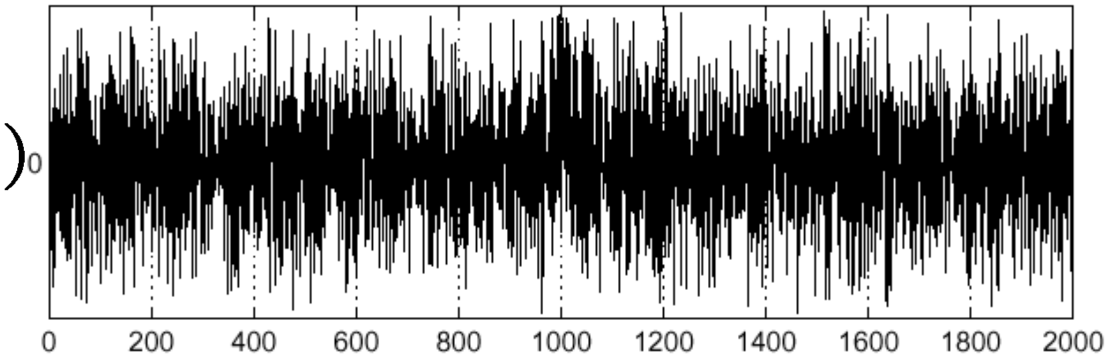
Effects of noise

- Consider a single row or column of the image
  - Plotting intensity as a function of position gives a signal





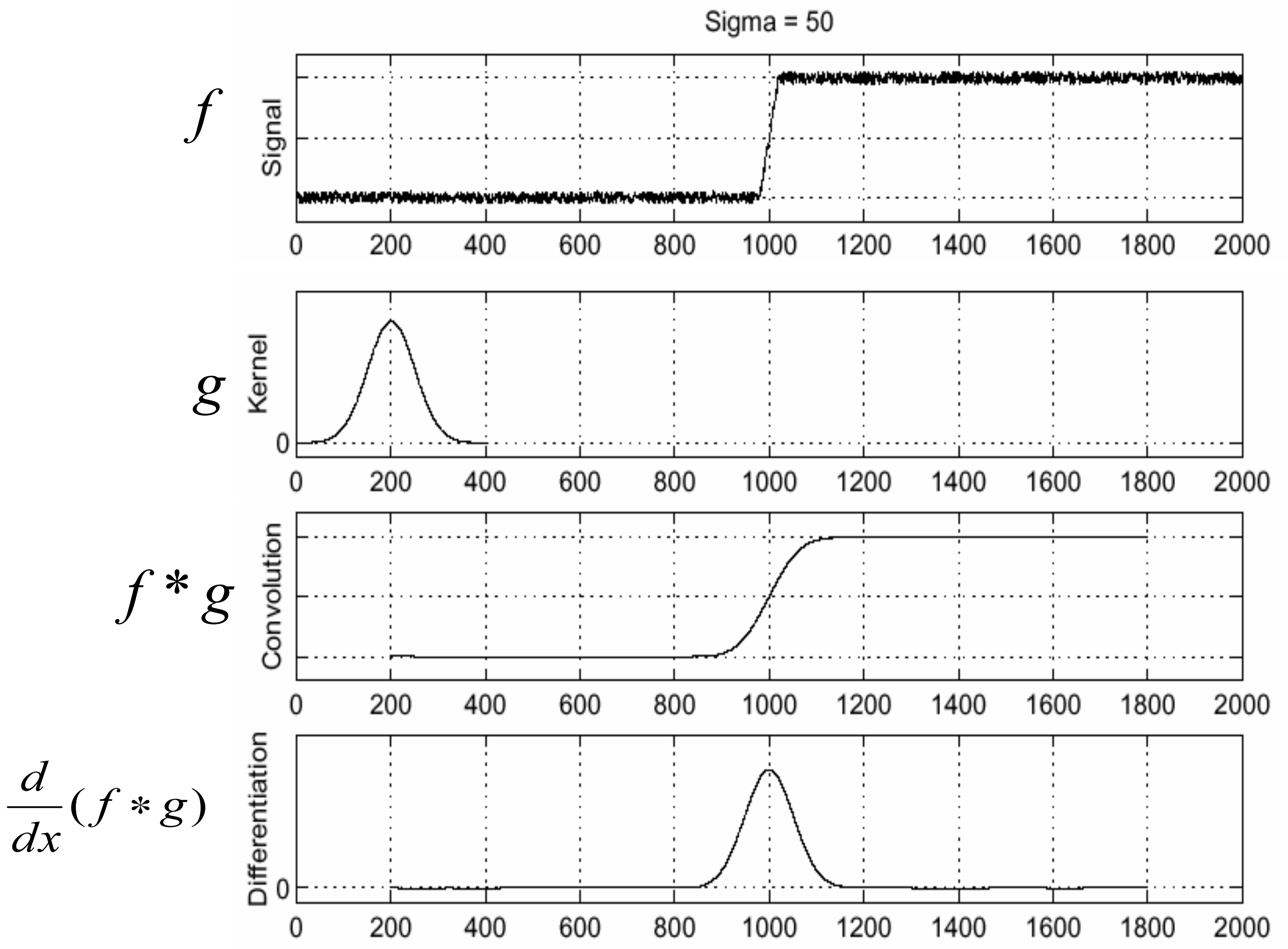
$$\frac{d}{dx}f(x)$$



- Where is the edge?

Source: S. Seitz

Solution: smooth first



- To find edges, look for peaks in $\frac{d}{dx}(f * g)$

Source: S. Seitz

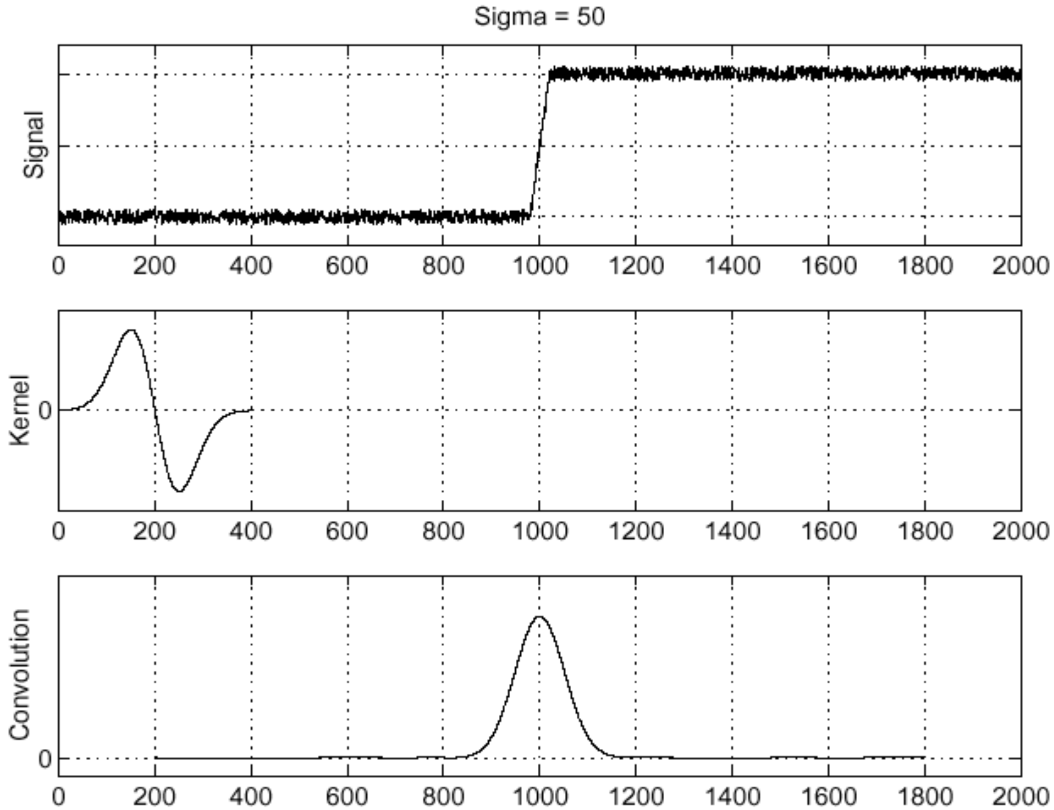
Derivative theorem of convolution

- Differentiation is convolution, and convolution is associative: $\frac{d}{dx}(f * g) = f * \frac{d}{dx} g$
- This saves us one operation:

f

$\frac{d}{dx} g$

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



Source: S. Seitz

Local descriptors

- Greyvalue derivatives
  - Convolution with Gaussian derivatives

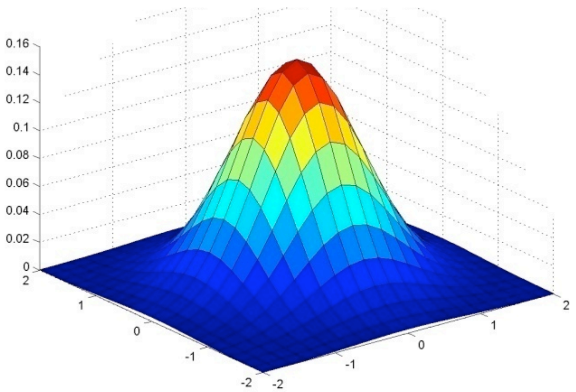
$$\mathbf{v}(x,y) = \begin{pmatrix} I(x,y) * G(\sigma) \\ I(x,y) * G_x(\sigma) \\ I(x,y) * G_y(\sigma) \\ I(x,y) * G_{xx}(\sigma) \\ I(x,y) * G_{xy}(\sigma) \\ I(x,y) * G_{yy}(\sigma) \\ \vdots \end{pmatrix} = \begin{pmatrix} L(x,y) \\ L_x(x,y) \\ L_y(x,y) \\ L_{xx}(x,y) \\ L_{xy}(x,y) \\ L_{yy}(x,y) \\ \vdots \end{pmatrix}$$

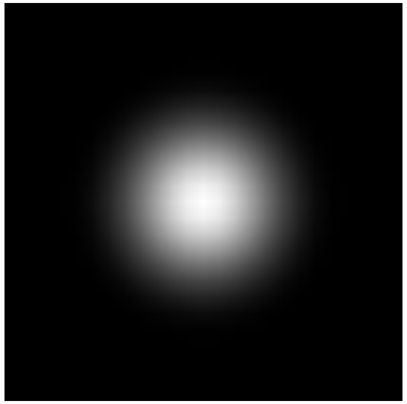
$$I(x,y) * G(\sigma) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x',y',\sigma) I(x-x',y-y') dx'dy'$$

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

Gaussian Kernel

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





- Gaussian filters have infinite support, but discrete filters use finite kernels

- Rule of thumb: set filter half-width to about 3 σ

Local descriptors – rotation invariance

Invariance to image rotation : differential invariants [Koen87]

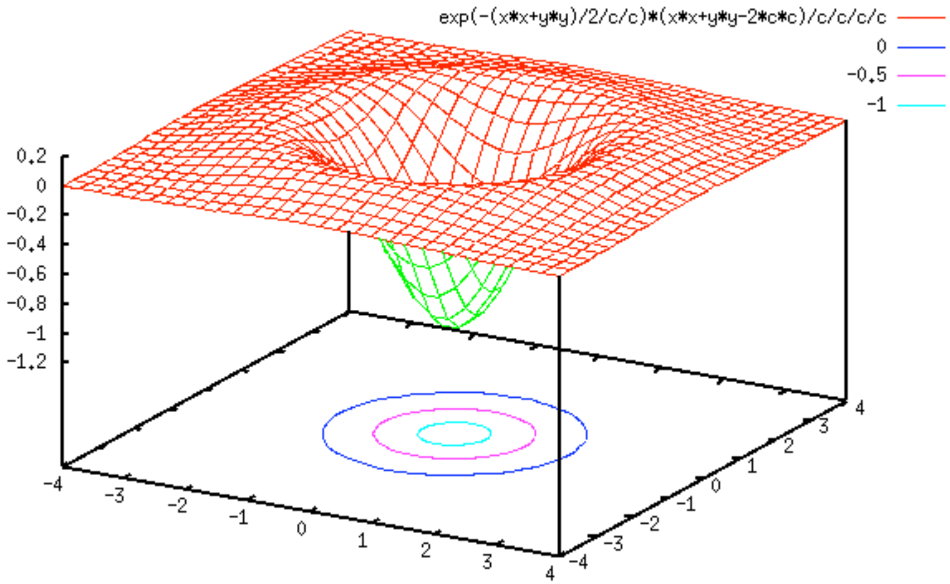
gradient magnitude → $L_x L_x + L_y L_y$

Laplacian → $L_{xx} + L_{yy}$

$$\begin{bmatrix} L \\ L_x L_x + L_y L_y \\ L_{xx} L_x L_x + 2 L_{xy} L_x L_y + L_{yy} L_{yy} \\ L_{xx} + L_{yy} \\ L_{xx} L_{xx} + 2 L_{xy} L_{xy} + L_{yy} L_{yy} \\ \cdots \\ \cdots \\ \cdots \\ \cdots \end{bmatrix}$$

Laplacian of Gaussian (LOG)

$$LOG = G_{xx}(\sigma) + G_{yy}(\sigma)$$



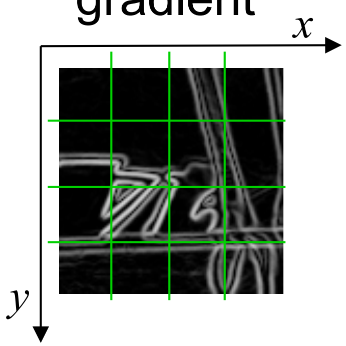
SIFT descriptor [Lowe'99]

- Approach
  - 8 orientations of the gradient
  - 4x4 spatial grid
  - soft-assignment to spatial bins, dimension 128
  - normalization of the descriptor to norm one
  - comparison with Euclidean distance

image patch

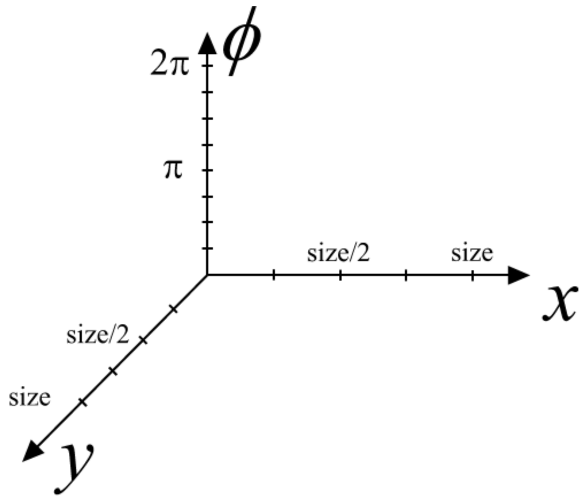
→

gradient



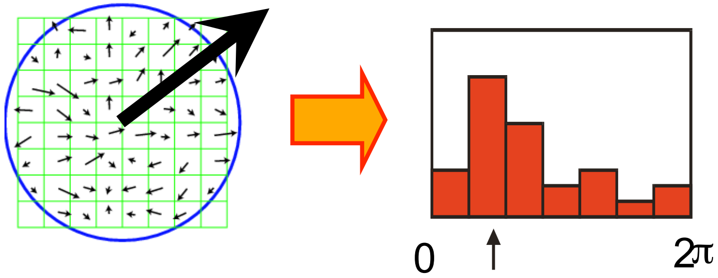
→

3D histogram

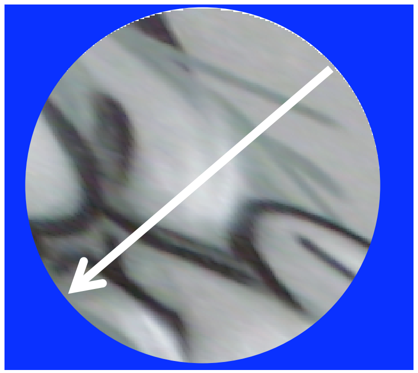


Local descriptors - rotation invariance

- Estimation of the dominant orientation
  - extract gradient orientation
  - histogram over gradient orientation
  - peak in this histogram



- Rotate patch in dominant direction



Local descriptors – illumination change

- Robustness to illumination changes

  in case of an affine transformation $I_1(\mathbf{x}) = aI_2(\mathbf{x}) + b$

- Normalization of the image patch with mean and variance

Invariance to scale changes

- Scale change between two images
- Scale factor s can be eliminated
- Support region for calculation!!
  - In case of a convolution with Gaussian derivatives defined by σ

$$I(x,y) * G(\sigma) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x',y',\sigma) I(x-x', y-y') dx' dy'$$

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$