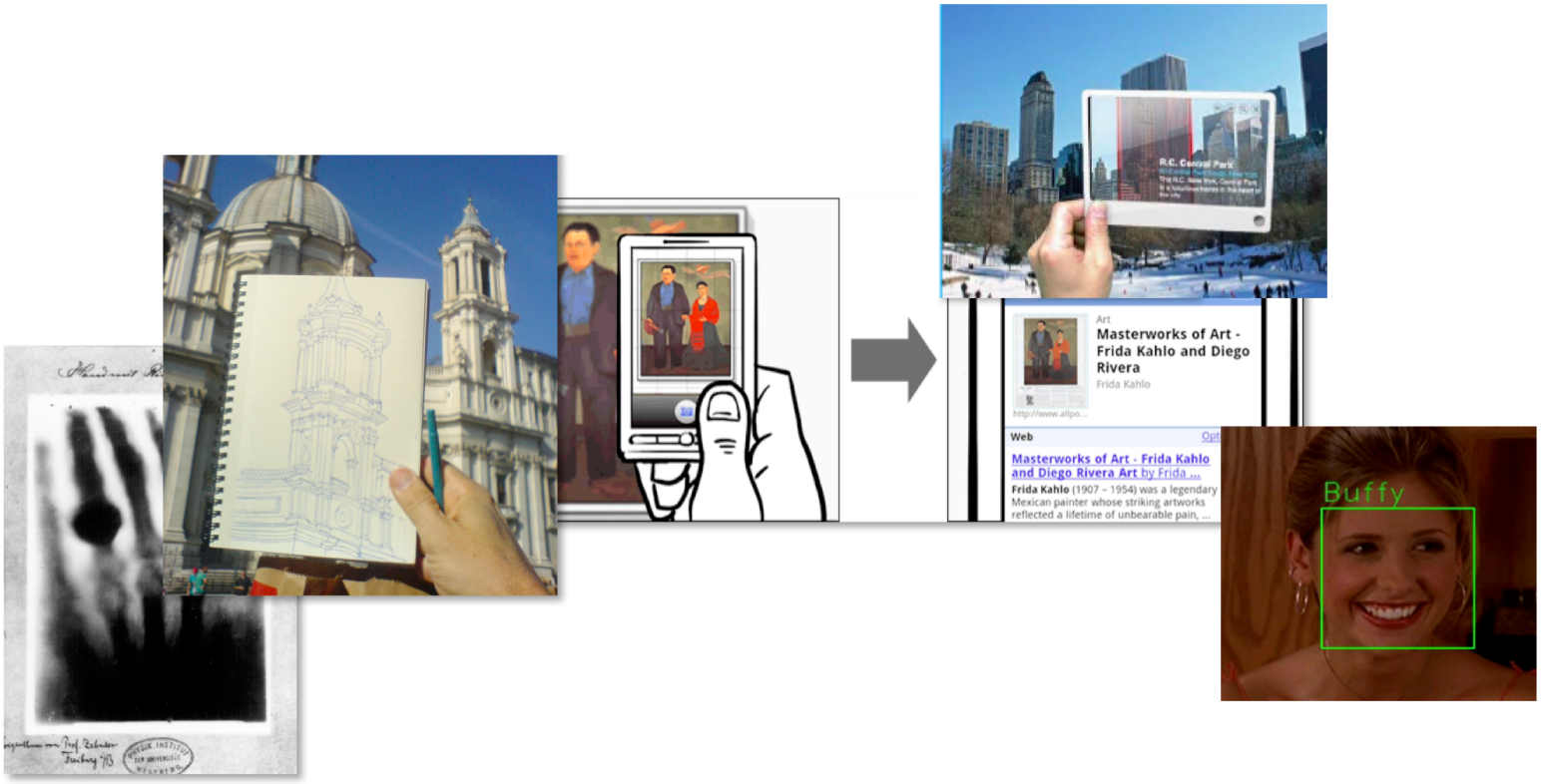


COURS RDFIA deep Image

http://www-poleia.lip6.fr/~cord/teaching-rdfia2020/

Matthieu Cord
Sorbonne University

Course Outline

Part I

1. Computer Vision Introduction (1): Visual (local) feature detection and description (2): Bag of Word Image representation
2. Introduction to Introduction to Machine Learning: Risk, Classification, Datasets, benchmarks and evaluation
3. Neural Nets (NNs), Linear classification (SVM)
4. Convolutional NNs -- Large deep ConvNets

COMPUTER VISION: WHERE ARE WE NOW?

Source (next slides): Cornell CV course

Deployed: depth cameras

https://realsense.intel.com/stereo/

Microsoft Kinect

Deployed: shape capture

The Matrix movies, ESC Entertainment, XYZRGB, NRC

Source: S. Seitz

Deployed: Optical character recognition (OCR)

- If you have a scanner, it probably came with OCR software

Digit recognition, AT&T labs
http://www.research.att.com/~yann/

License plate readers
http://en.wikipedia.org/wiki/Automatic_number_plate_recognition

Automatic check processing

Source: S. Seitz

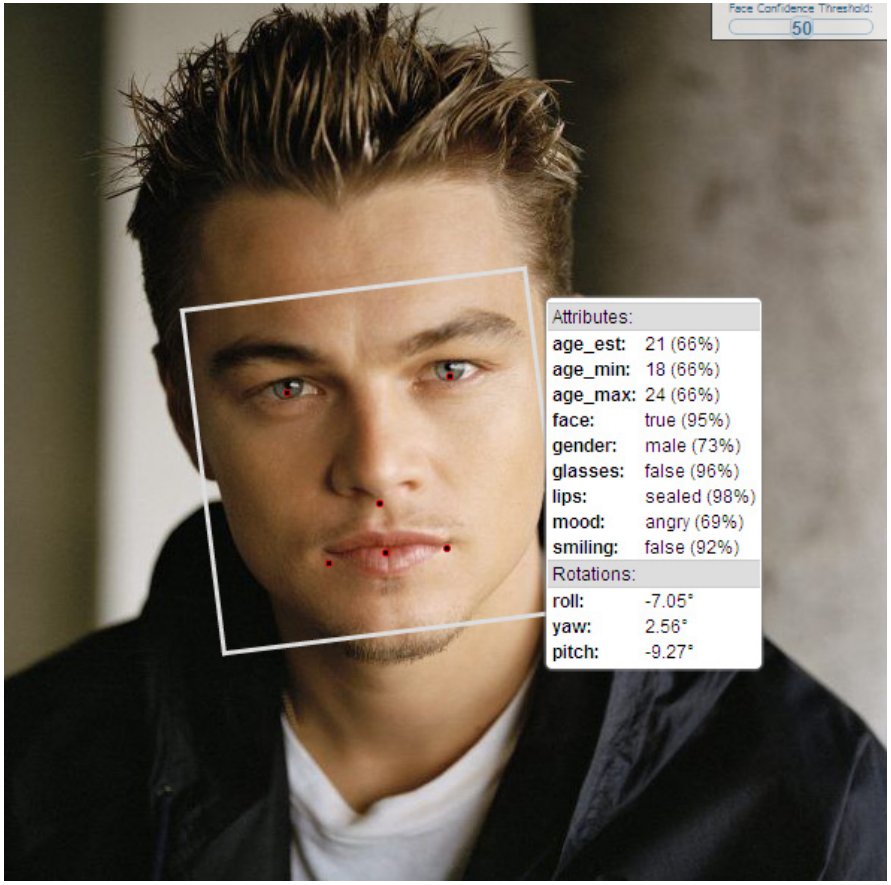
Deployed: Face detection

- Cameras now detect faces
  - Canon, Sony, Fuji, …

Source: S. Seitz

Significant progress: Face Recognition

FACE++



Significant progress: Recognizing objects

Mask R-CNN. Kaiming He, Georgia Gkioxari, Piotr Dollar, Ross Girshick. ICCV 2017

Recognition-based product search

Challenges: Other imaging domains

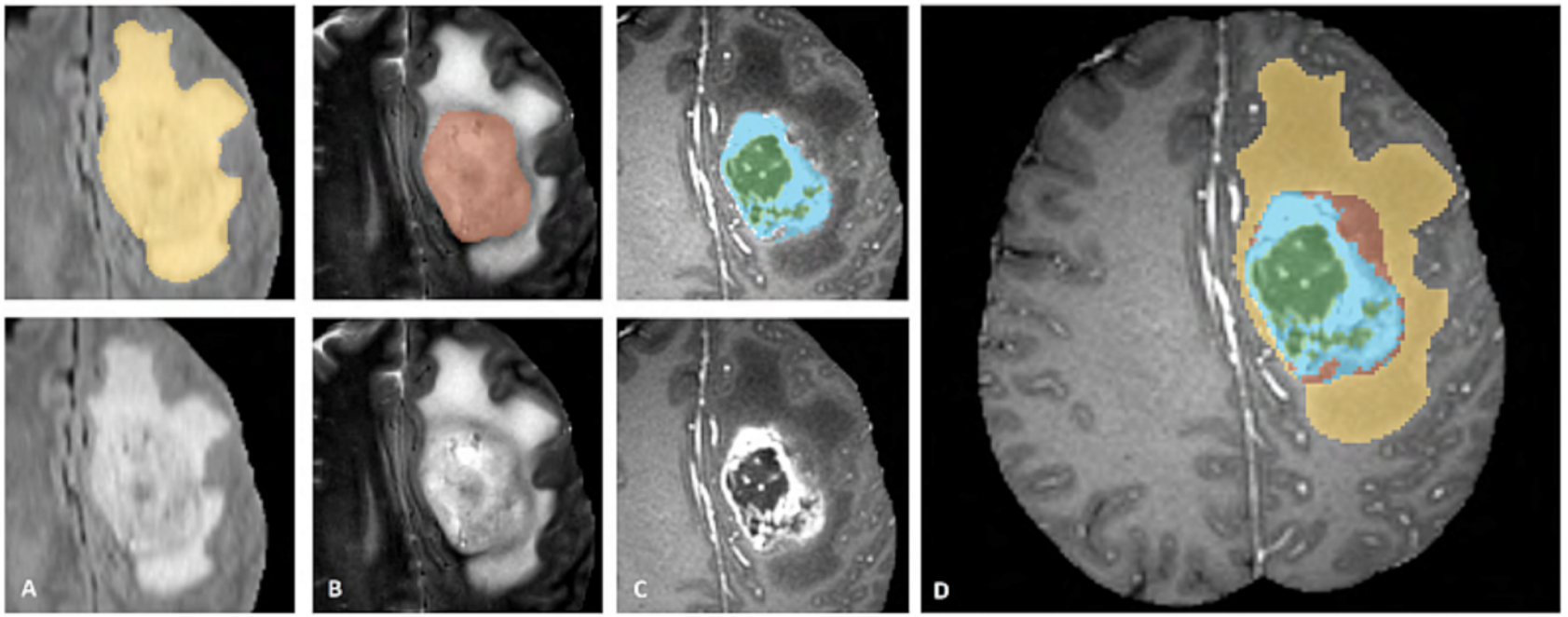
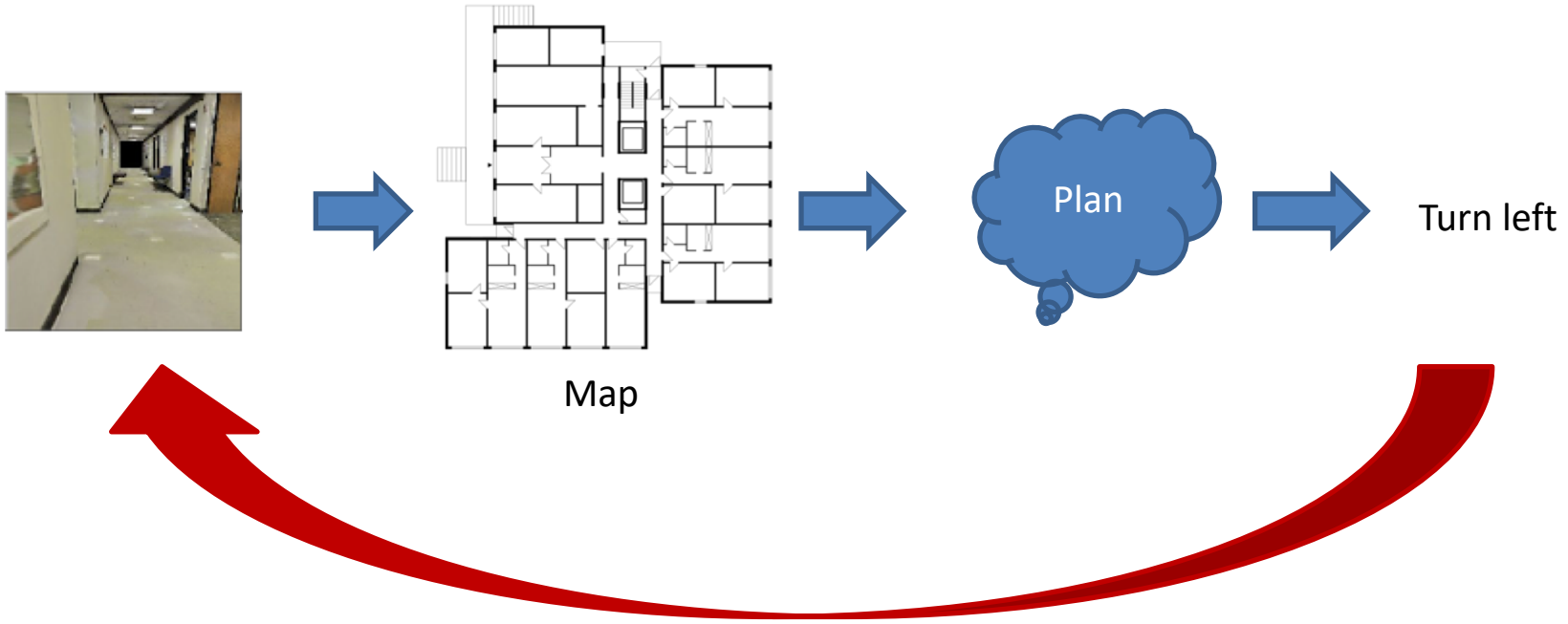


Fig.1: Glioma sub-regions. Shown are image patches with the tumor sub-regions that are annotated in the different modalities (top left) and the final labels for the whole dataset (right). The image patches show from left to right: the whole tumor (yellow) visible in T2-FLAIR (Fig.A), the tumor core (red) visible in T2 (Fig.B), the enhancing tumor structures (light blue) visible in T1Gd, surrounding the cystic/necrotic components of the core (green) (Fig. C). The segmentations are combined to generate the final labels of the tumor sub-regions (Fig.D): edema (yellow), non-enhancing solid core (red), necrotic/cystic core (green), enhancing core (blue). (Figure taken from the BraTS IEEE TMI paper.)

Challenges: Integrating Vision and Action



Cognitive Mapping and Planning for Visual Navigation
Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, Jitendra Malik
CVPR 2017

Challenge: Visual Reasoning
VQA task: Why is this funny?

The picture above is funny.

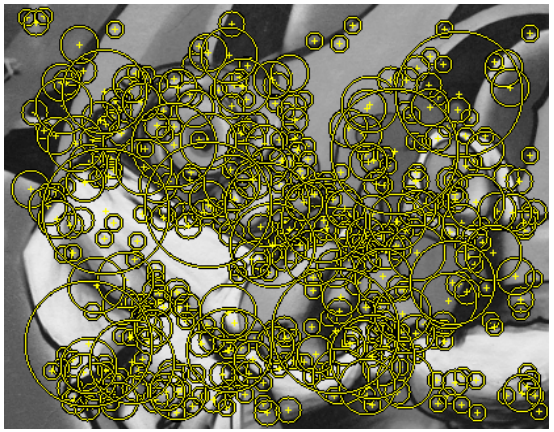
Andrej Karpathy

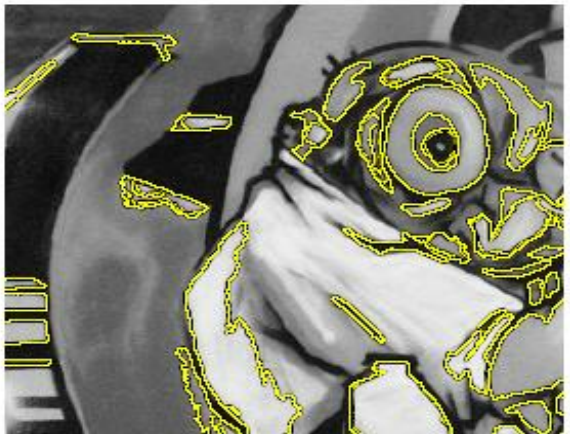
Course Outline

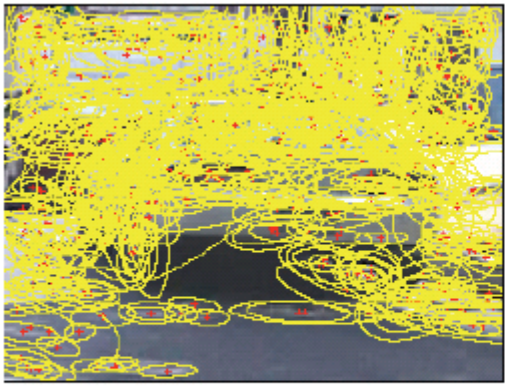
1. **Computer Vision Introduction (1): Visual (local) feature detection and description (2): Bag of Word Image representation**

Local feature detection and description

Points/Regions of Interest detection

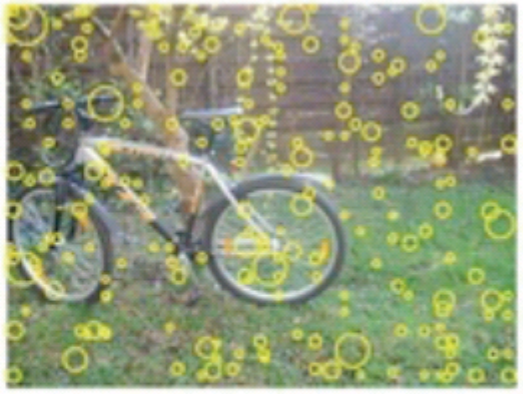






Sparse, at interest points

Dense, uniformly

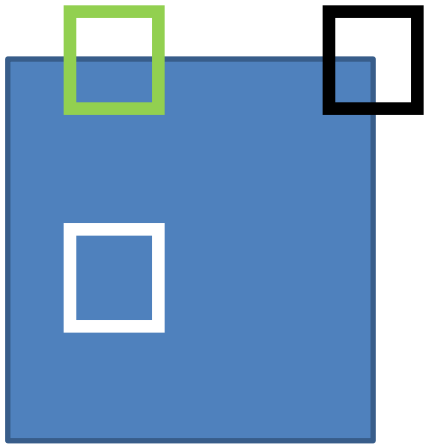


Randomly

One example: Corner detection (Harris corner detector)

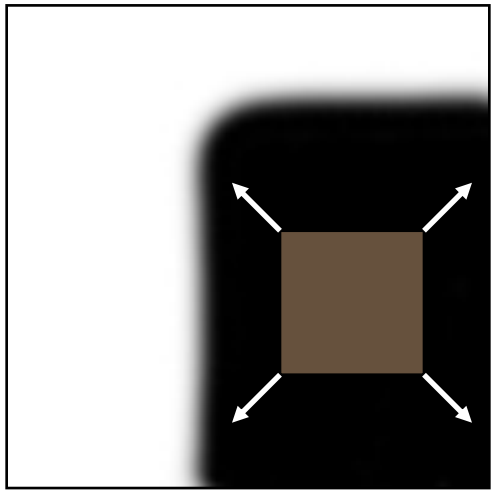
Corner detection

- Main idea: Translating window should cause large differences in patch appearance

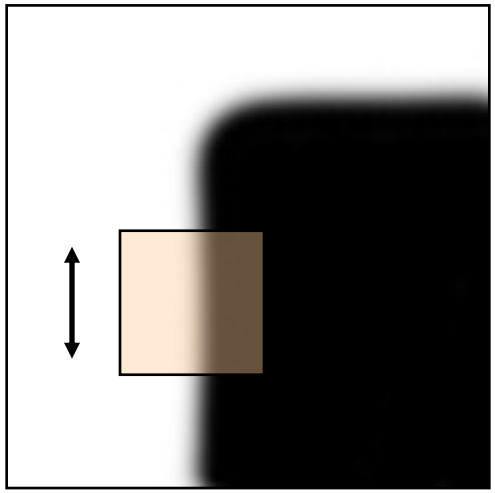


Corner Detection: Basic Idea

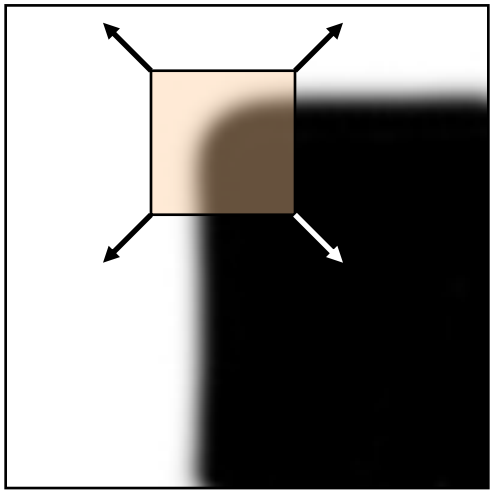
Recognize the type of point (flat, edge, corner) by looking through a small window W



"flat" region:
no change in
all directions



"edge":
no change
along the edge
direction



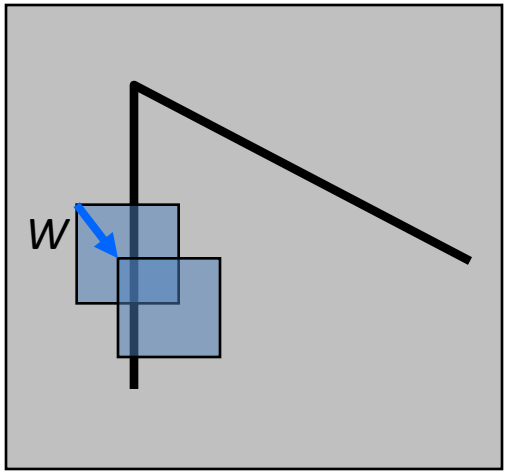
"corner":
significant
change in all
directions

Corner detection op == Shifting a window in *any direction*, keep the ones that give *a large change* in intensity

Harris corner detection: the math

Consider shifting the window *W* by (*u,v*)

- how do the pixels in W change?
- compare each pixel before and after by summing up the squared differences (SSD)
- this defines an SSD "error" *E*(*u,v*):



$$E(u,v) = \sum_{(x,y)\in W} [I(x+u, y+v) - I(x,y)]^2$$

- We want E(u,v) to be *as high as possible for all u, v!*

Small motion assumption

Taylor Series expansion of *I*:

$$I(x+u, y+v) = I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \text{higher order terms}$$

If the motion (u,v) is small, then first order approximation is good

$$\begin{aligned} I(x+u, y+v) &\approx I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v \\ &\approx I(x,y) + \begin{bmatrix} I_x & I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \end{aligned}$$

$$\text{shorthand: } I_x = \frac{\partial I}{\partial x}$$

Plugging this into the formula on the previous slide…

Corner detection: the math

$$E(u,v) = \sum_{(x,y)\in W} [I(x+u,y+v) - I(x,y)]^2$$

$$\approx \sum_{(x,y)\in W} [I(x,y) + I_x u + I_y v - I(x,y)]^2$$

$$E(u,v) \approx \sum_{(x,y)\in W} [I_x u + I_y v]^2$$

$$\approx Au^2 + 2Buv + Cv^2$$

$$A = \sum_{(x,y)\in W} I_x^2 \qquad B = \sum_{(x,y)\in W} I_x I_y \qquad C = \sum_{(x,y)\in W} I_y^2$$

E(*u*,*v*) is locally approximated as a quadratic error function

Interpreting the second moment matrix

$$M = \sum_{x,y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} A & B \\ B & C \end{bmatrix}$$

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

Recall that we want E(u,v) to be as large as possible for all u,v

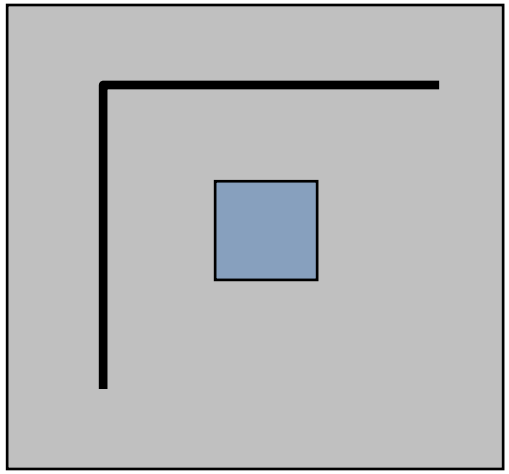
What does this mean in terms of M?

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\begin{bmatrix} A & B \\ B & C \end{bmatrix}}_{M} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y)\in W} I_x^2$$

$$B = \sum_{(x,y)\in W} I_x I_y$$

$$C = \sum_{(x,y)\in W} I_y^2$$



$$M = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$M \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$E(u,v) = 0 \quad \forall u, v$$

Flat patch: $I_x = 0$

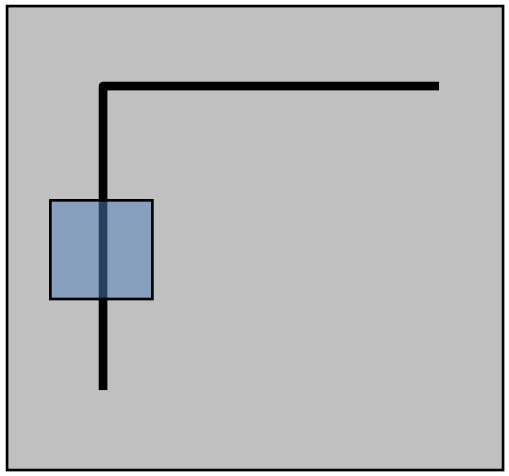
$I_y = 0$

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\begin{bmatrix} A & B \\ B & C \end{bmatrix}}_{M} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y)\in W} I_x^2$$

$$B = \sum_{(x,y)\in W} I_x I_y$$

$$C = \sum_{(x,y)\in W} I_y^2$$



Vertical edge: $I_y = 0$

$$M = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}$$

$$M \begin{bmatrix} 0 \\ v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

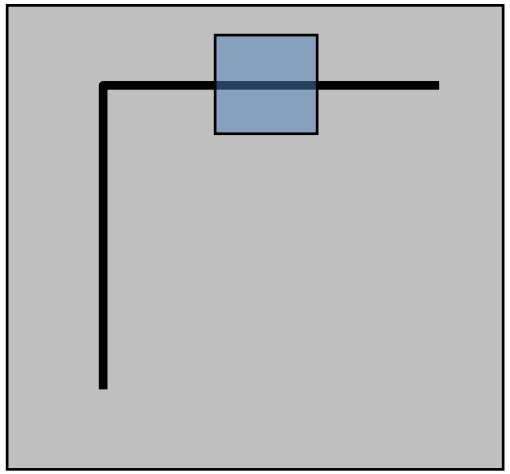
$$E(0,v) = 0 \quad \forall v$$

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\begin{bmatrix} A & B \\ B & C \end{bmatrix}}_{M} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y)\in W} I_x^2$$

$$B = \sum_{(x,y)\in W} I_x I_y$$

$$C = \sum_{(x,y)\in W} I_y^2$$



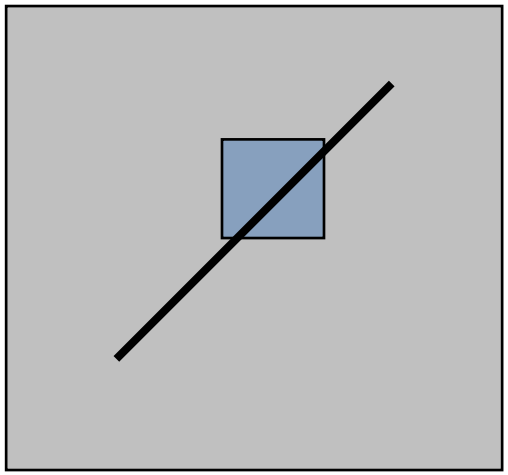
Horizontal edge: $I_x = 0$

$$M = \begin{bmatrix} 0 & 0 \\ 0 & C \end{bmatrix}$$

$$M \begin{bmatrix} u \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$E(u,0) = 0 \; \forall u$$

What about edges in arbitrary orientation?



$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

$$M \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow E(u,v) = 0$$

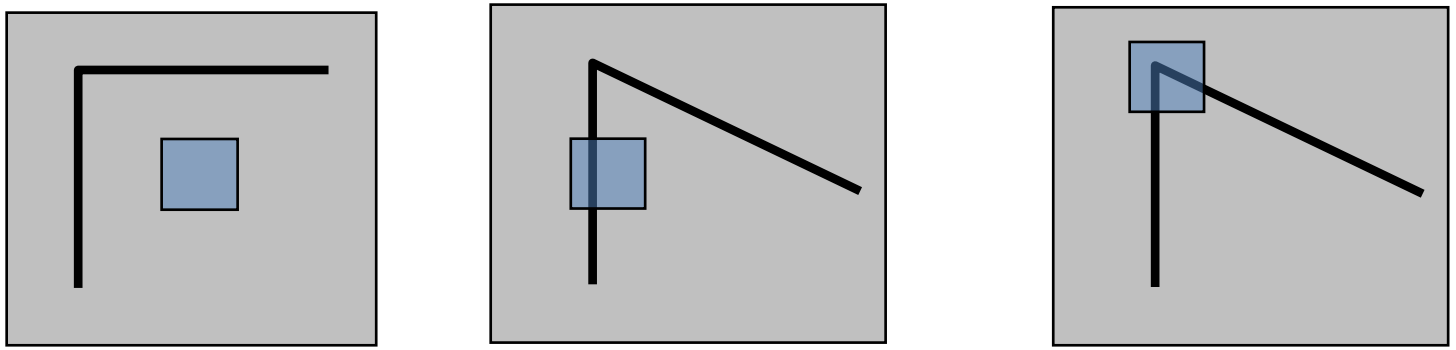
$$M \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Leftrightarrow E(u,v) = 0$$

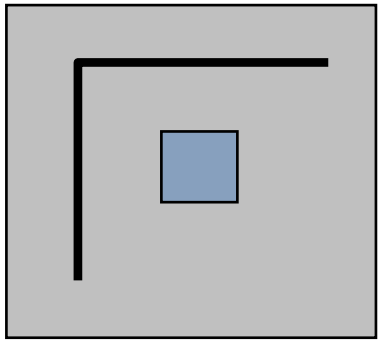
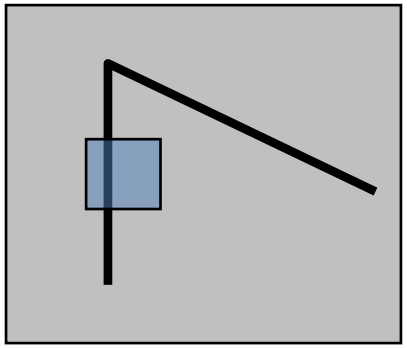
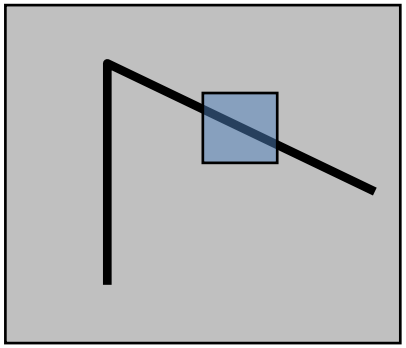
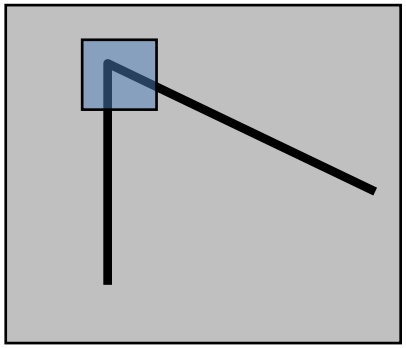
Solutions to Mx = 0 are directions for which E is 0: window can slide in this direction without changing appearance

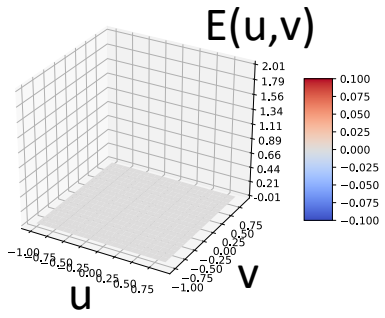
$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

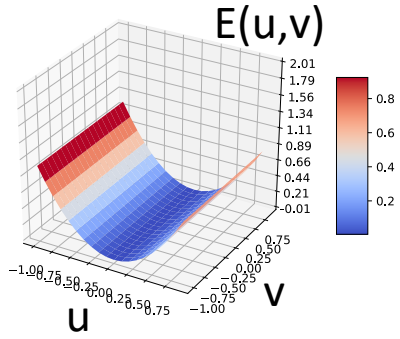
Solutions to Mx = 0 are directions for which E is 0: window can slide in this direction without changing appearance

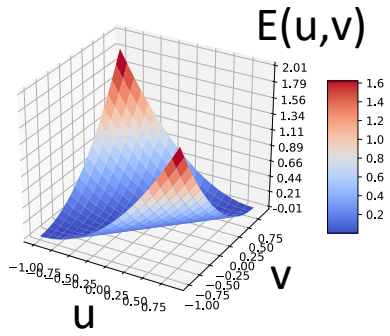
For corners, we want no such directions to exist

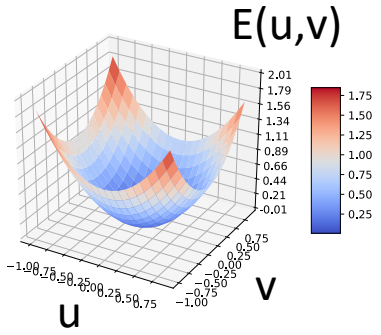



E(u,v)
u
v


E(u,v)
u
v


E(u,v)
u
v


E(u,v)
u
v

Eigenvalues and eigenvectors of M

- $Mx = 0 \Rightarrow Mx = \lambda x$: x is an eigenvector of M with eigenvalue 0
- M is 2 x 2, so it has 2 eigenvalues $(\lambda_{max}, \lambda_{min})$ with eigenvectors (x_{max}, x_{min})
- $E(x_{max}) = \ x_{max}^T M x_{max} = \lambda_{max} ||x_{max}||^2 = \lambda_{max}$ (eigenvectors have unit norm)
- $E(x_{min}) = x_{min}^T M x_{min} = \lambda_{min} ||x_{min}||^2 = \lambda_{min}$

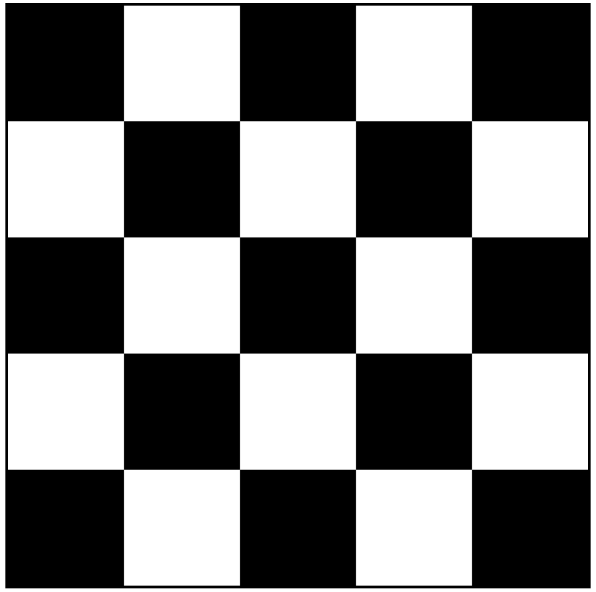
Corner detection: the math

How are λ_{max}, x_{max}, λ_{min}, and x_{min} relevant for feature detection?

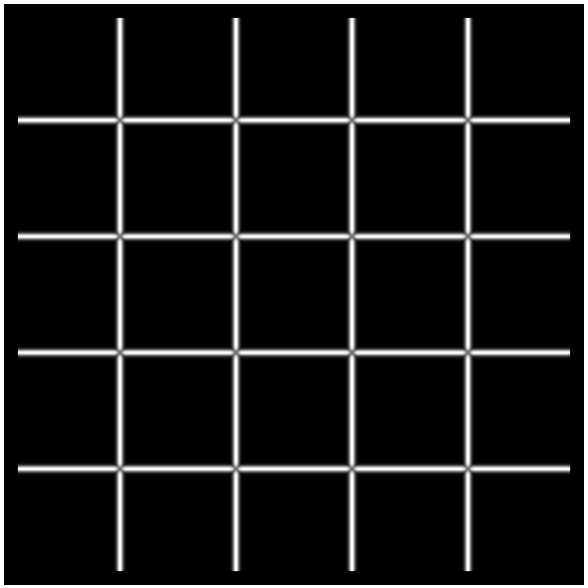
- Need a feature scoring function

Want $E(u,v)$ to be large for small shifts in all directions

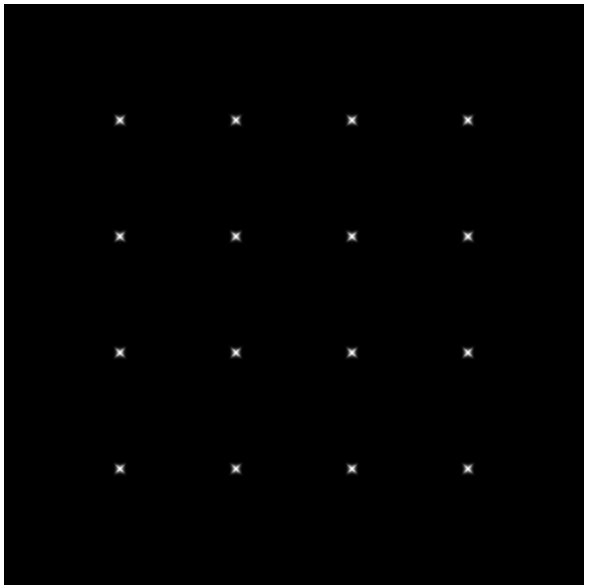
- the minimum of $E(u,v)$ should be large, over all unit vectors $[u\ v]$
- this minimum is given by the smaller eigenvalue (λ_{min}) of M



I



$\lambda_{\max}$

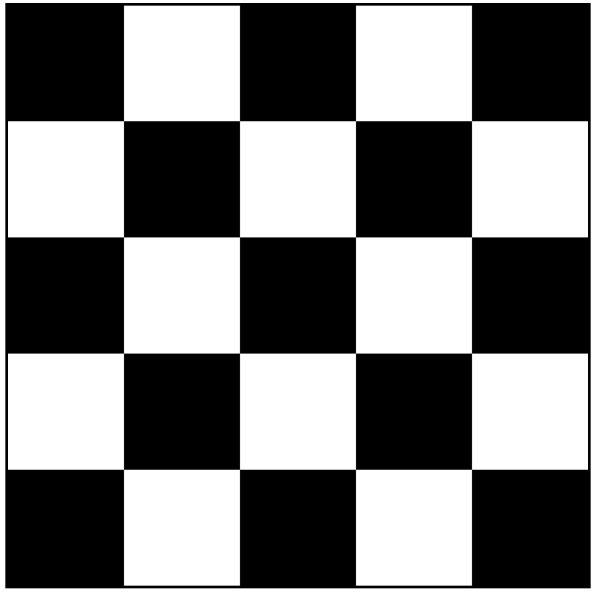


$\lambda_{\min}$

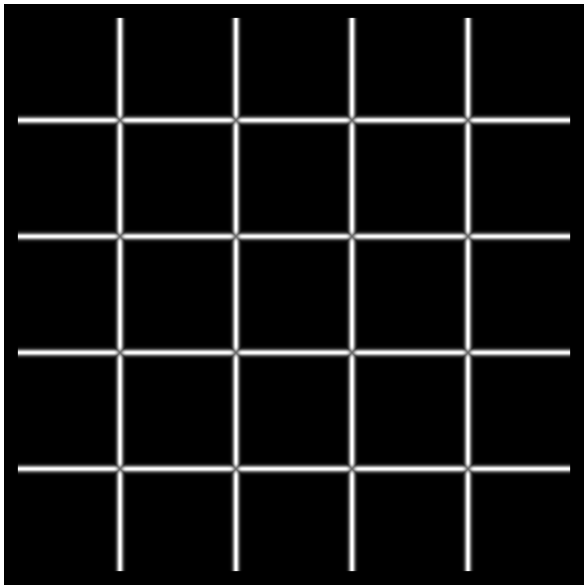
Corner detection summary

Here's what you do

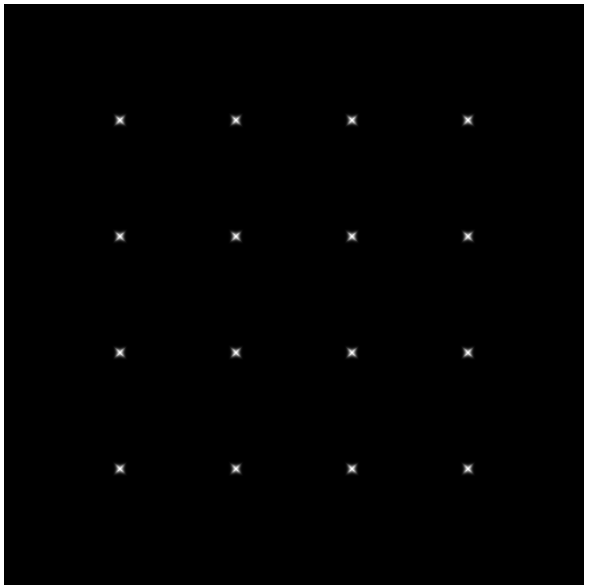
- Compute the gradient at each point in the image
- Create the *M* matrix from the entries in the gradient
- Compute the eigenvalues
- Find points with large response (λ_{min} > threshold)
- Choose those points where λ_{min} is a local maximum as features



I



$\lambda_{\max}$

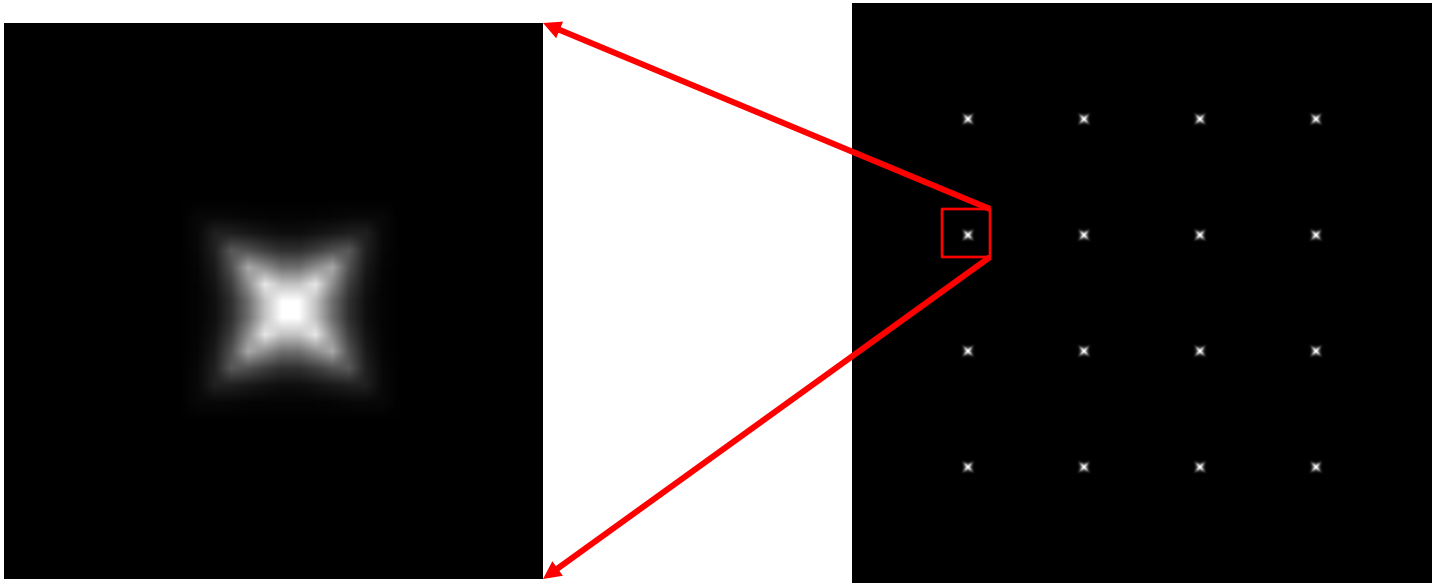


$\lambda_{\min}$

Corner detection summary

Here's what you do

- Compute the gradient at each point in the image
- Create the *H* matrix from the entries in the gradient
- Compute the eigenvalues
- Find points with large response (λ_{min} > threshold)
- Choose those points where λ_{min} is a local maximum as features



λ_{min}

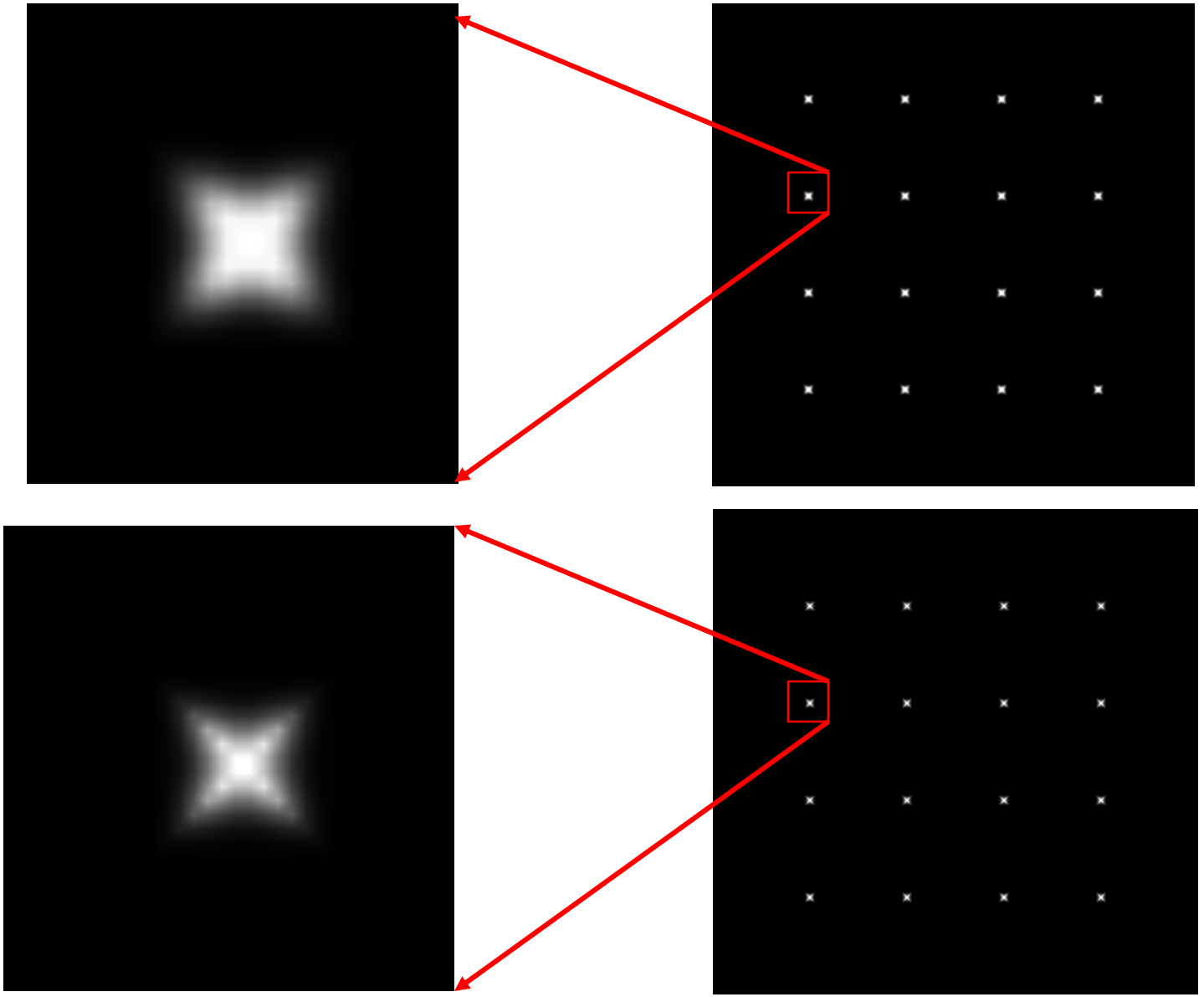
The Harris operator

λ_{min} is a variant of the "Harris operator" R for feature detection:

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

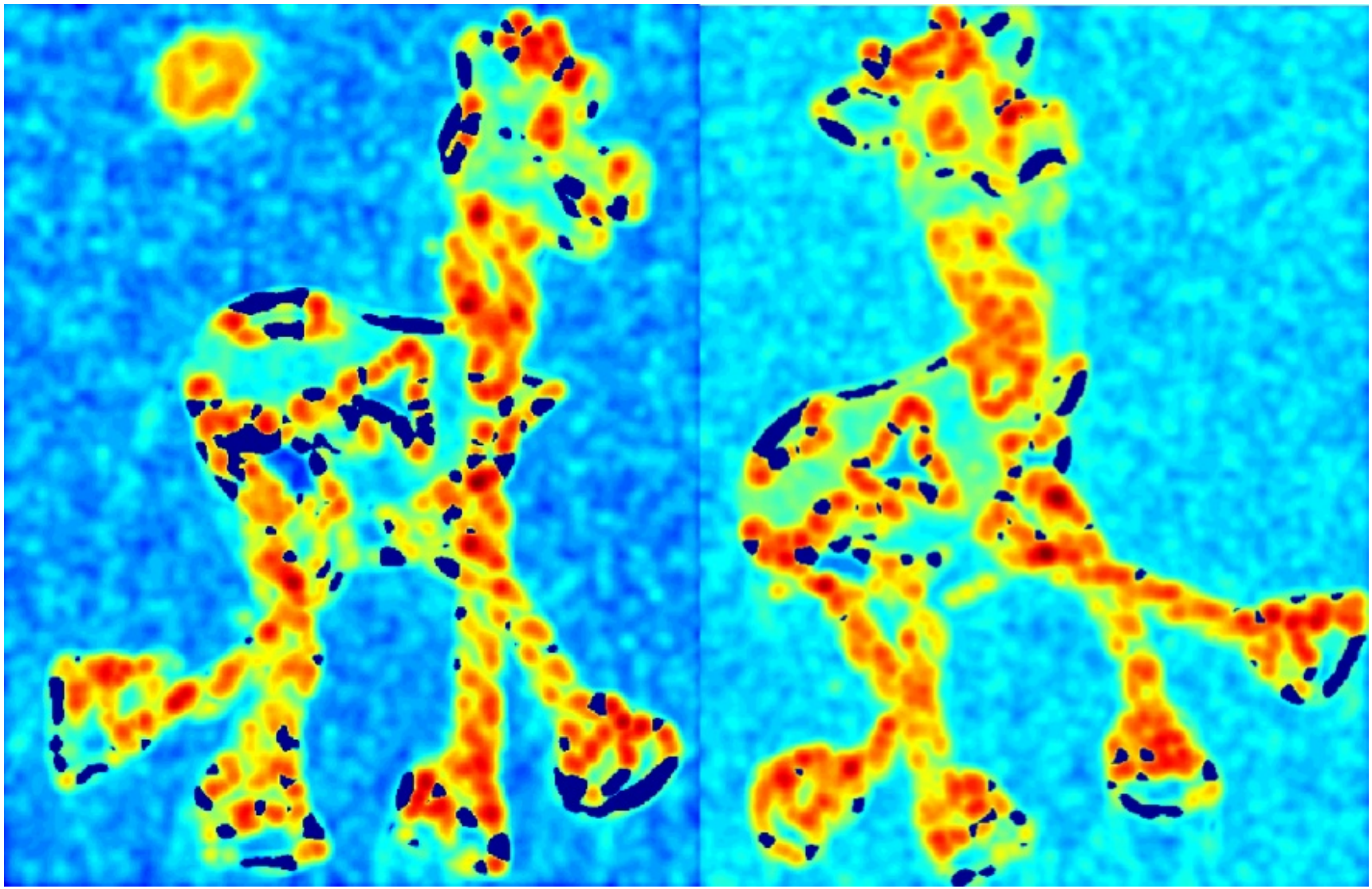
- The *trace* is the sum of the diagonals, i.e., *trace(H) = h_{11} + h_{22}*
- Very similar to λ_{min} but less expensive (no square root)
- Called the "Harris Corner Detector" or "Harris Operator"

The Harris operator

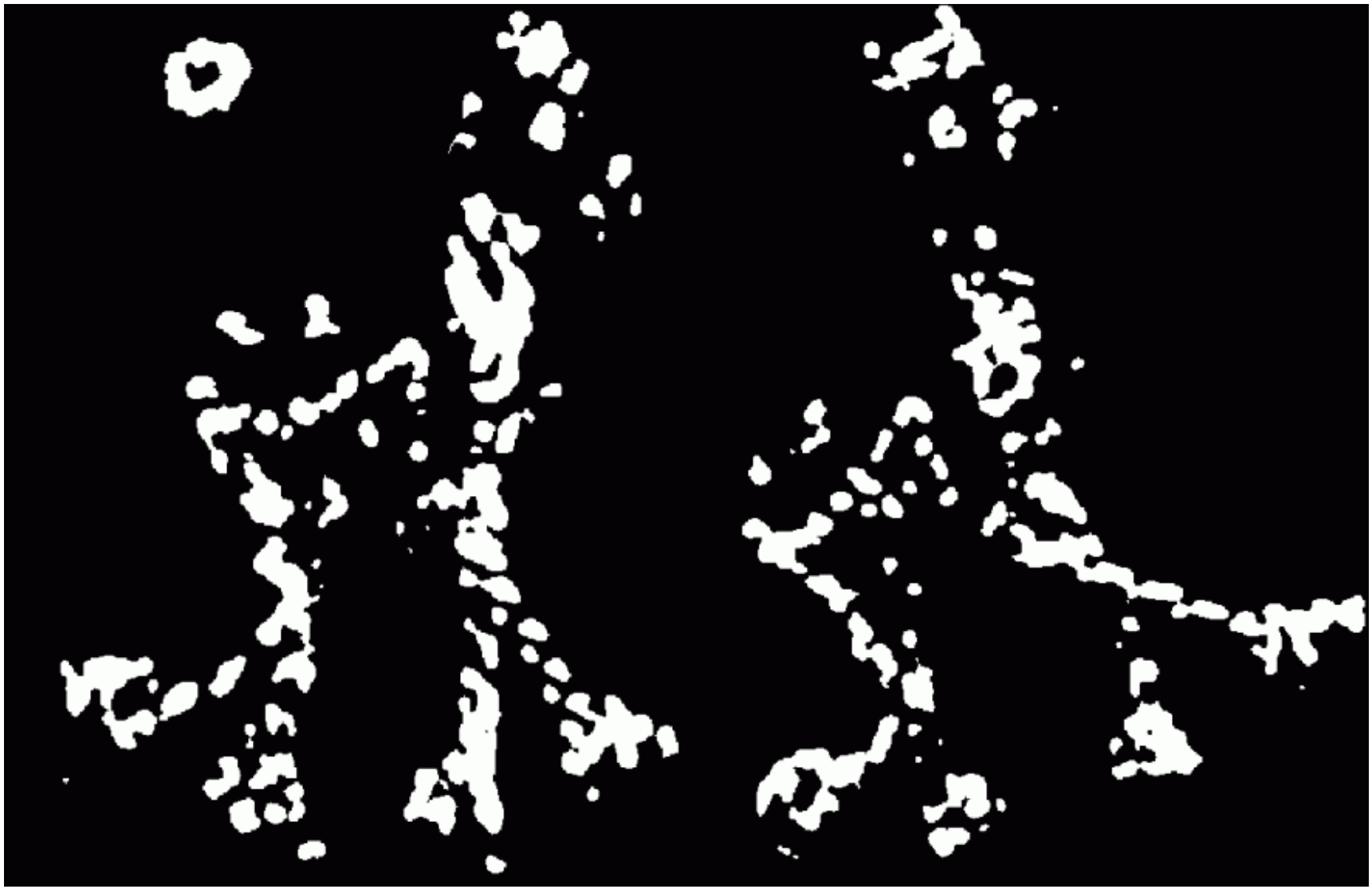


Harris detector example

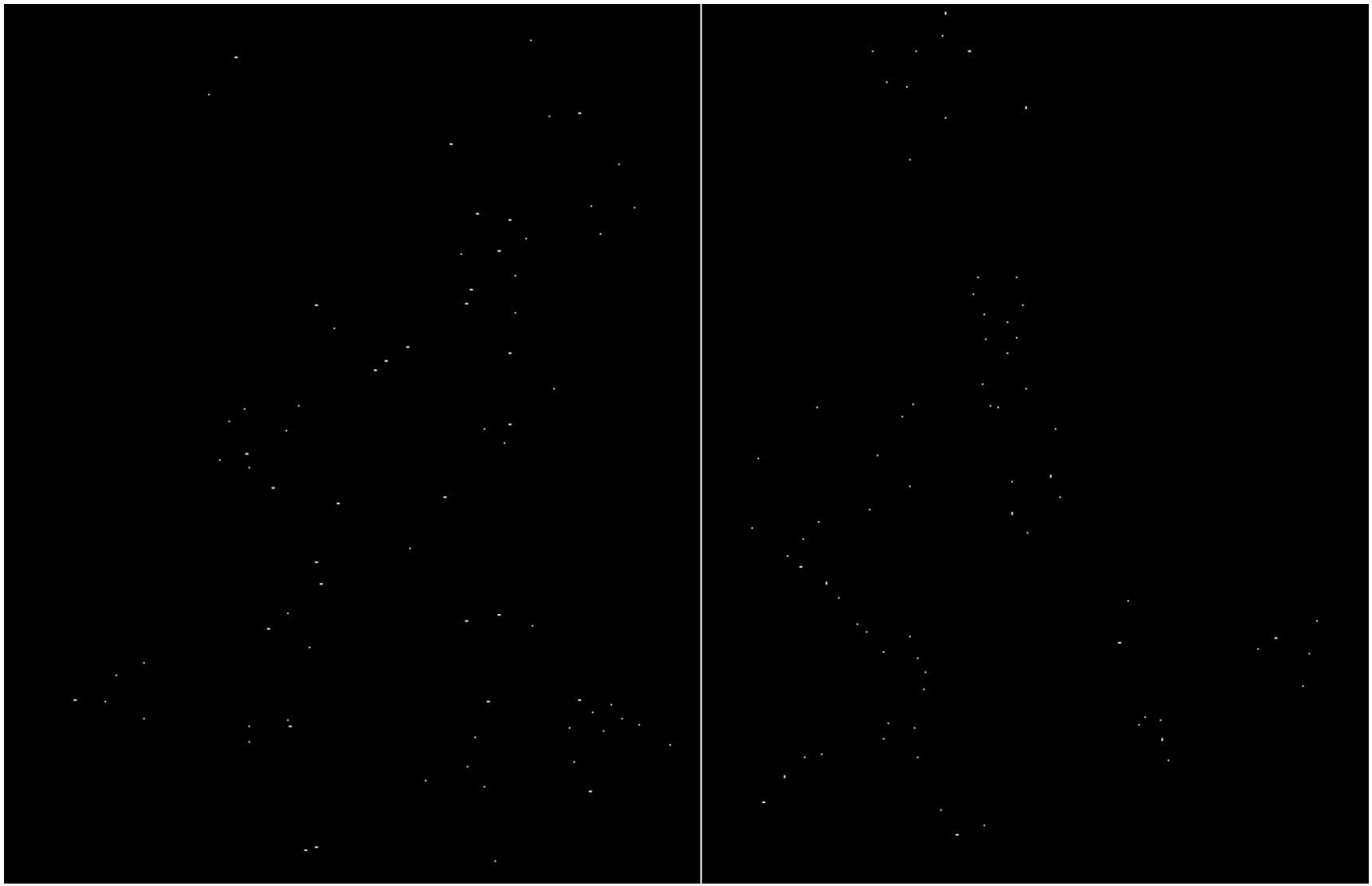
f value (red high, blue low)



Threshold (f > value)



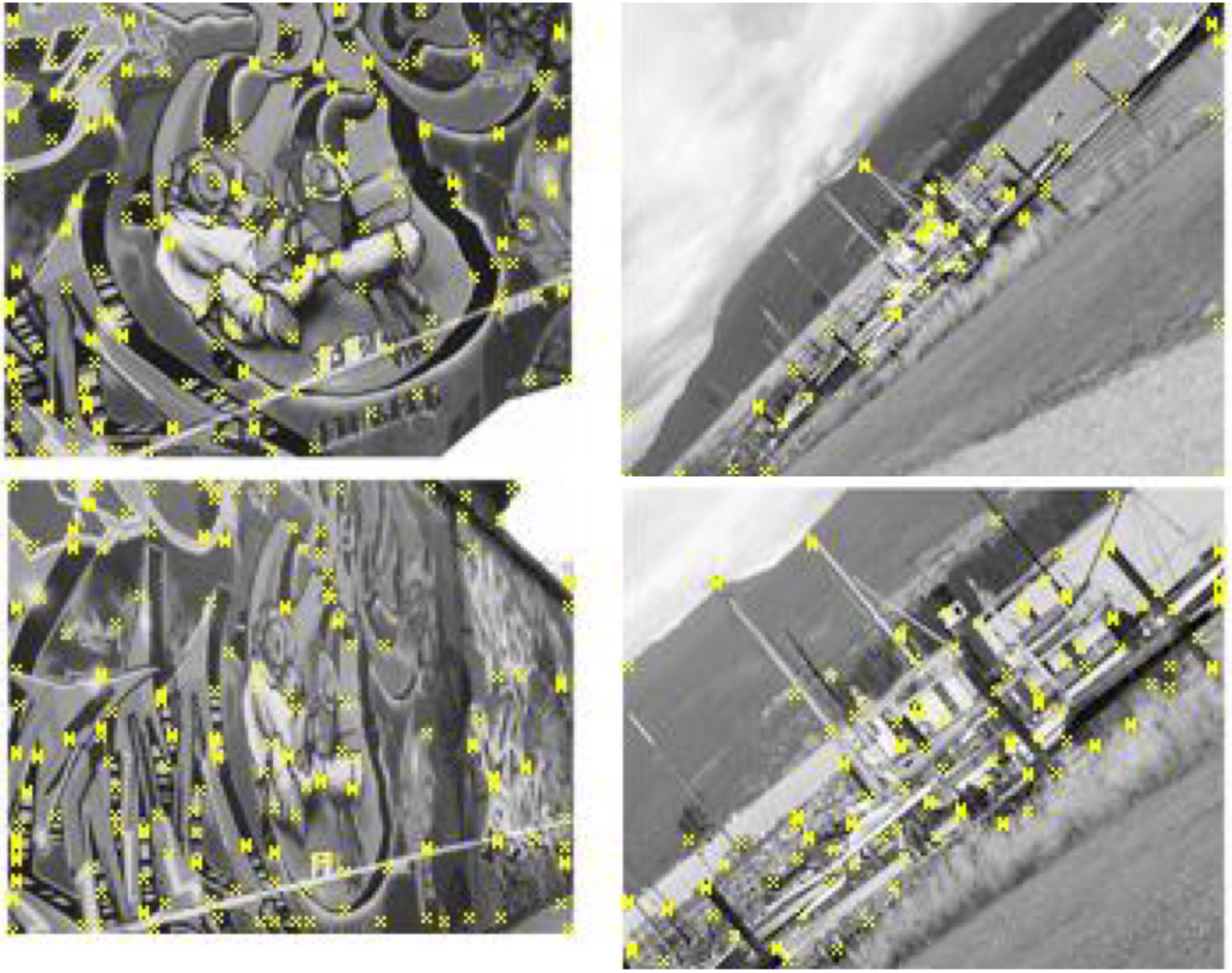
Find local maxima of f



Harris features (in red)

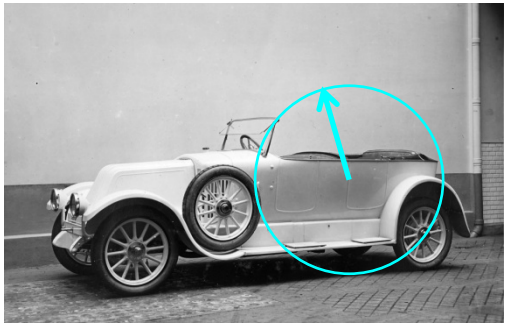
Local feature detection

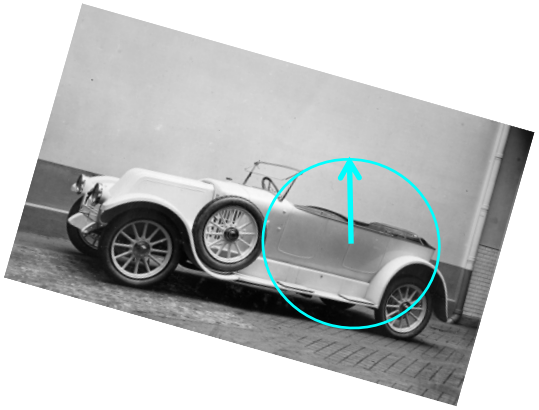
Looking for repeatability



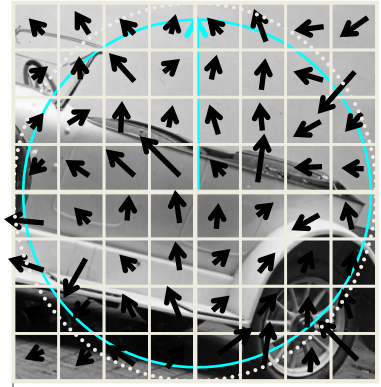
Local feature description

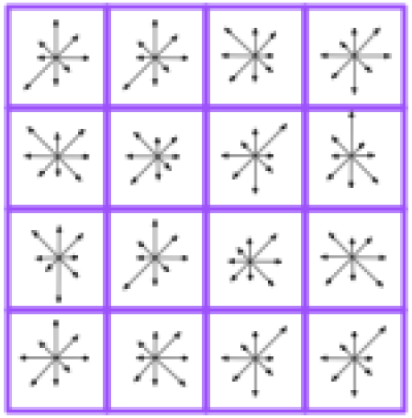
Local description (always looking for invariance)





SIFT descriptors/features

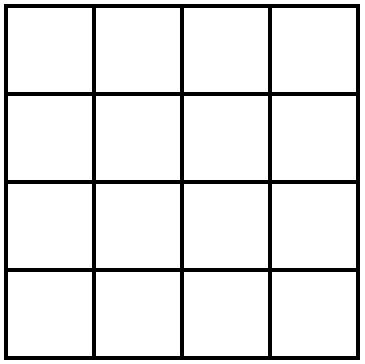




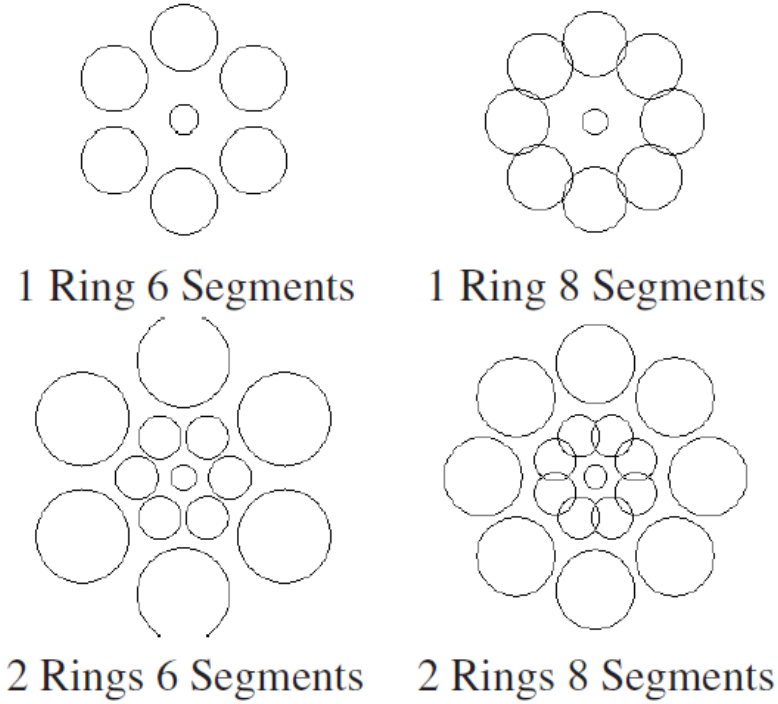
10
17
35
77
35
8
44
3
27
3
0
...

Extension: Daisy

Circular gradient binning



SIFT



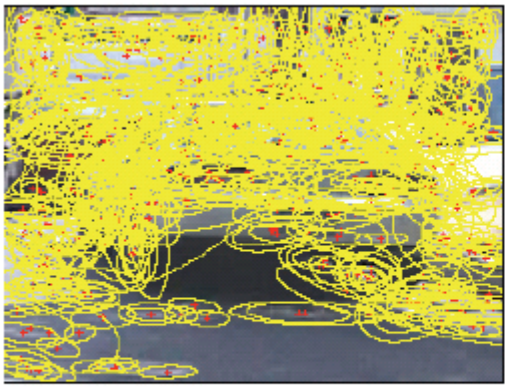
Daisy

Picking the best DAISY, S. Winder, G. Hua, M. Brown, CVPR 09

Feature descriptors

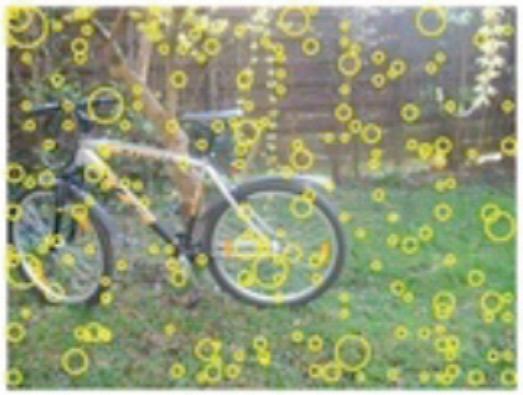
- Expected properties?
  - Similar patches => close descriptors
  - Invariance (robustness) to geom. transformation : rotation, scale, view point, luminance, semantics ? …

BoF: (First) Image representation

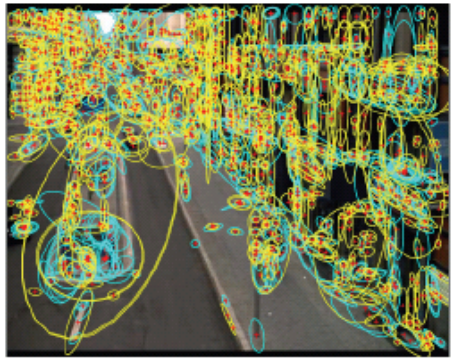


Sparse, at interest points

Dense, uniformly

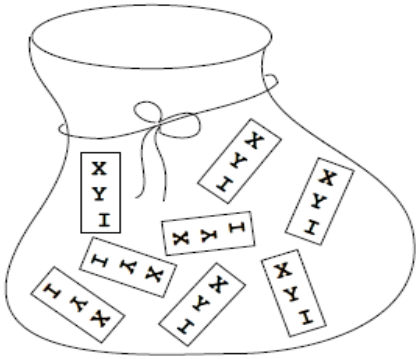


Randomly



Multiple interest operators

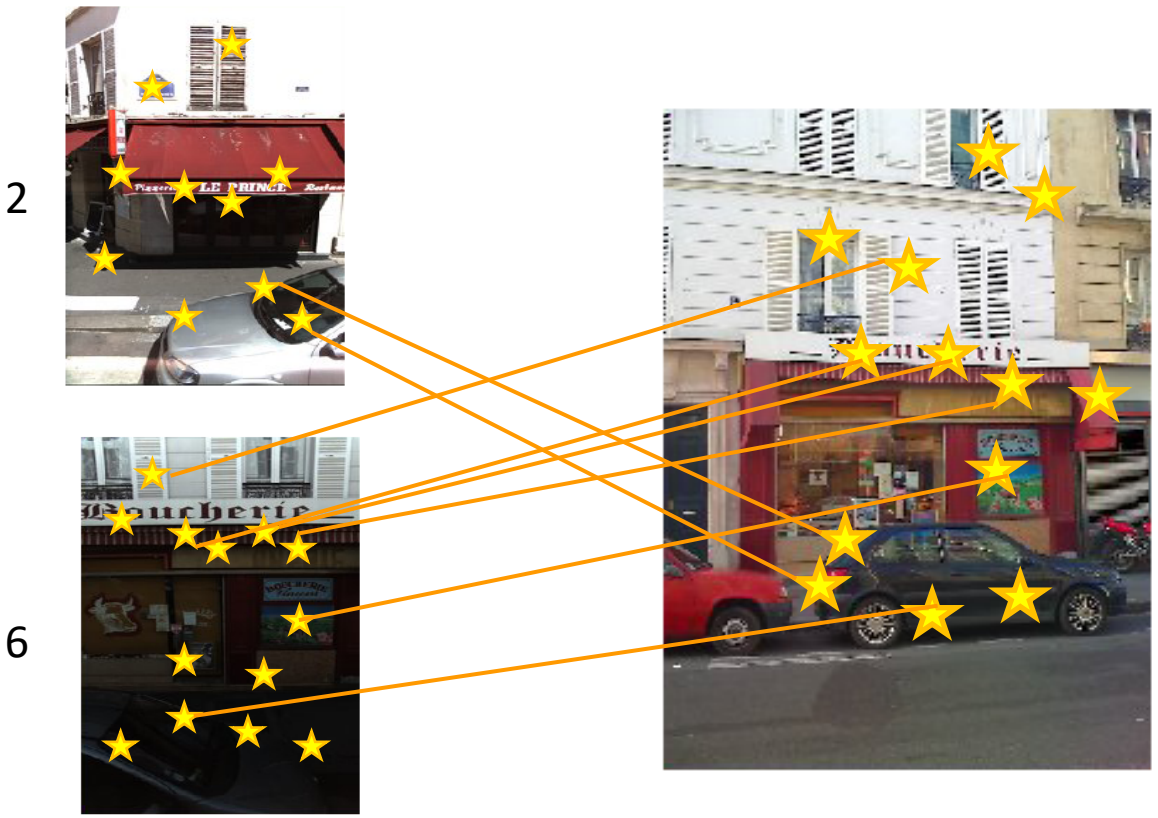
Feature extraction



A bag of features
BoF

BoF -- Image representation

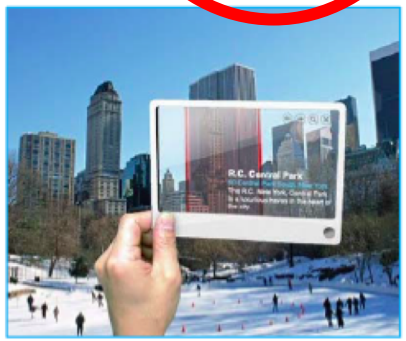
- Image similarity based on matching of local features + voting



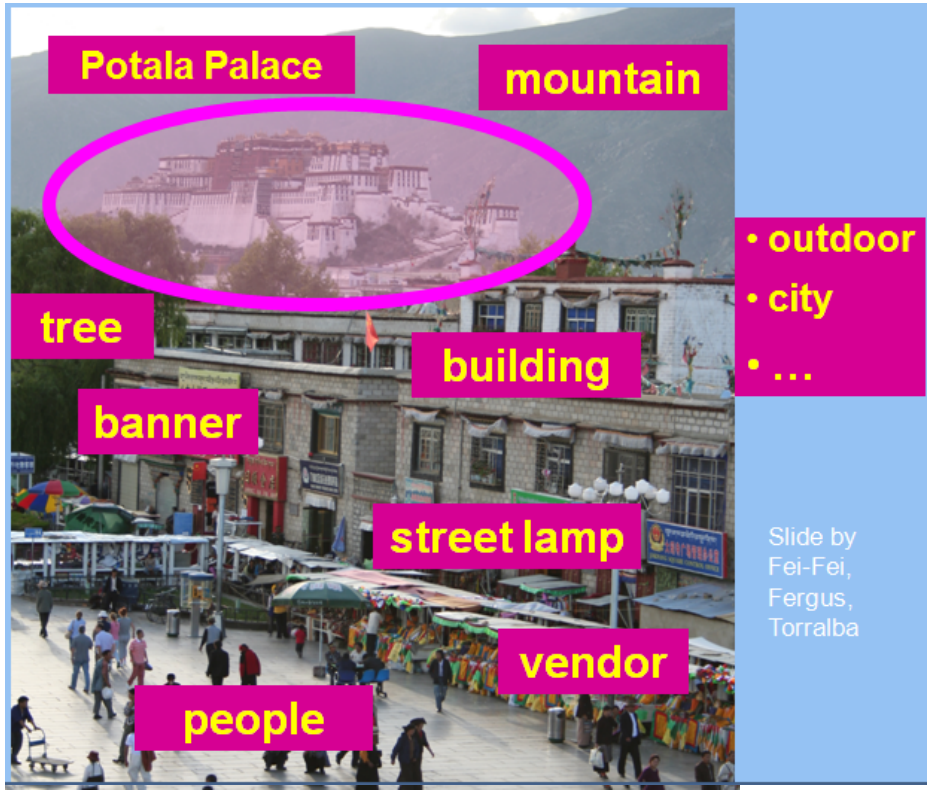
Applications to Image Retrieval

Image Retrieval

- Context: Instance search (second example)



Advanced Understanding



Two pizzas sitting on top of a stove top oven

Image Understanding

- Focus of this course: recognition, classification and understanding
- Fundamental Pbs:
  - Image representation,
  - Data similarity,
  - Decision function
- Examples of applications:
  - face recognition
  - human action recognition in movies
  - Search Engines
  - Pattern recognition in remote sensing,
  - Medical imagery