EIE4512 - Digital Image Processing
Interest Points

香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

Zhen Li
lizhen@cuhk.edu.cn
School of Science and Engineering
The Chinese University of Hong Kong, Shen Zhen
March 5 --7, 2019



How do we build a panorama?

We need to match (align) images

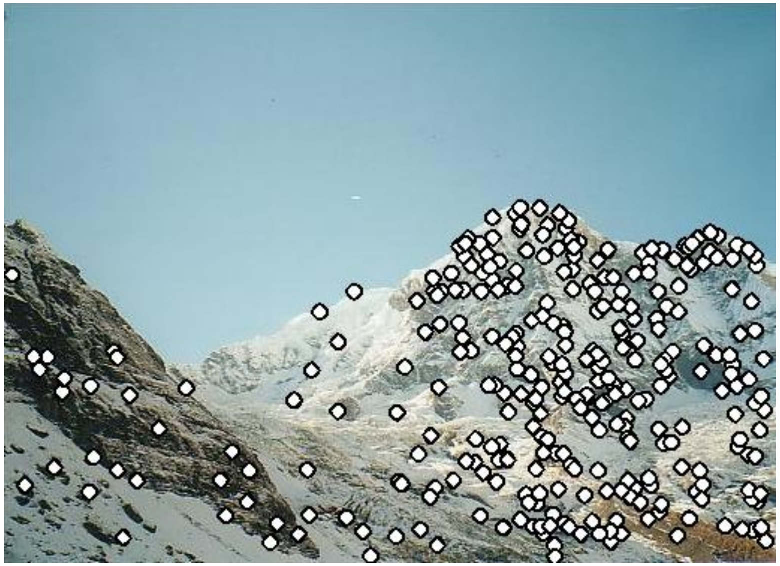
Global methods sensitive to occlusion, lighting, parallax effects. So look for local features that match well.

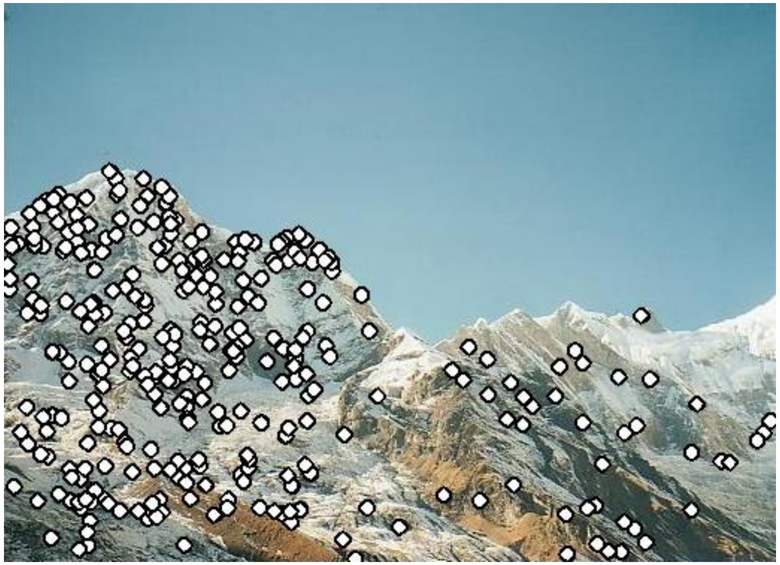
How would you do it by eye?

How do we build a panorama?

Matching with features

- Detect feature points in both images

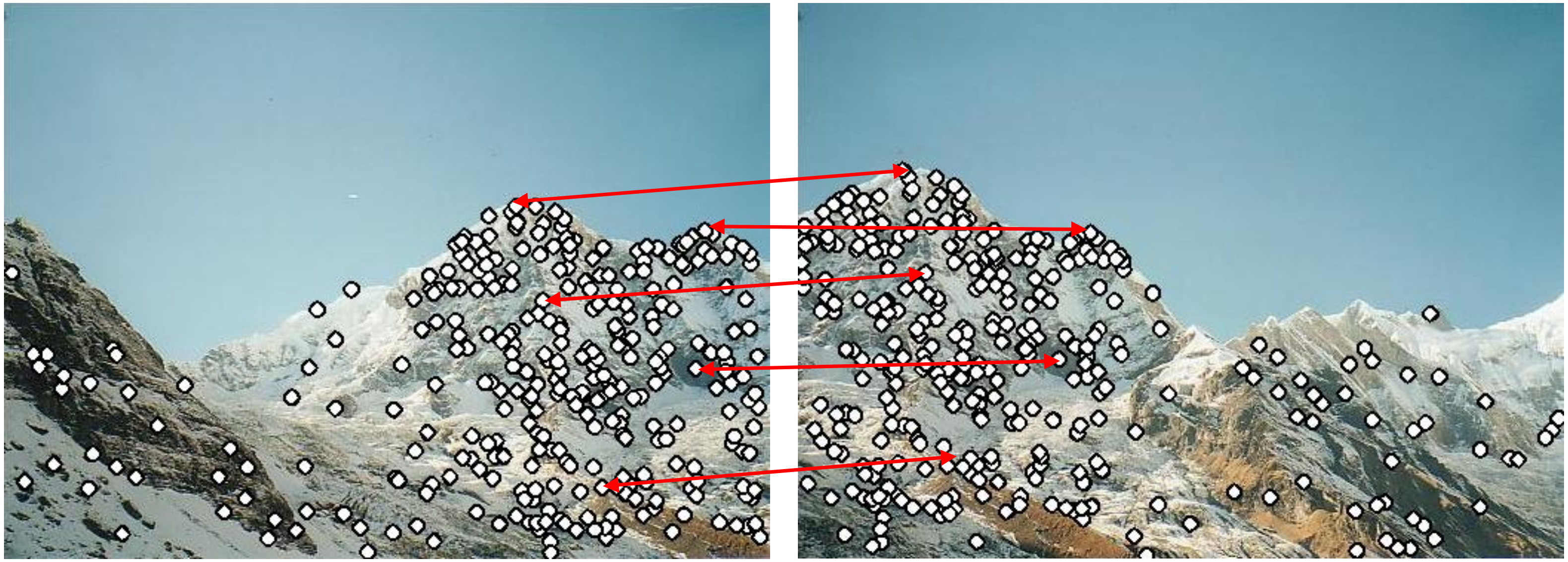




How do we build a panorama?

Matching with features

- Detect feature points in both images
- Find corresponding pairs



How do we build a panorama?

Matching with features

- Detect feature points in both images
- Find corresponding pairs
- Use these matching pairs to align images - the required mapping is called a homography.

What is a good interest point?

Repeatability

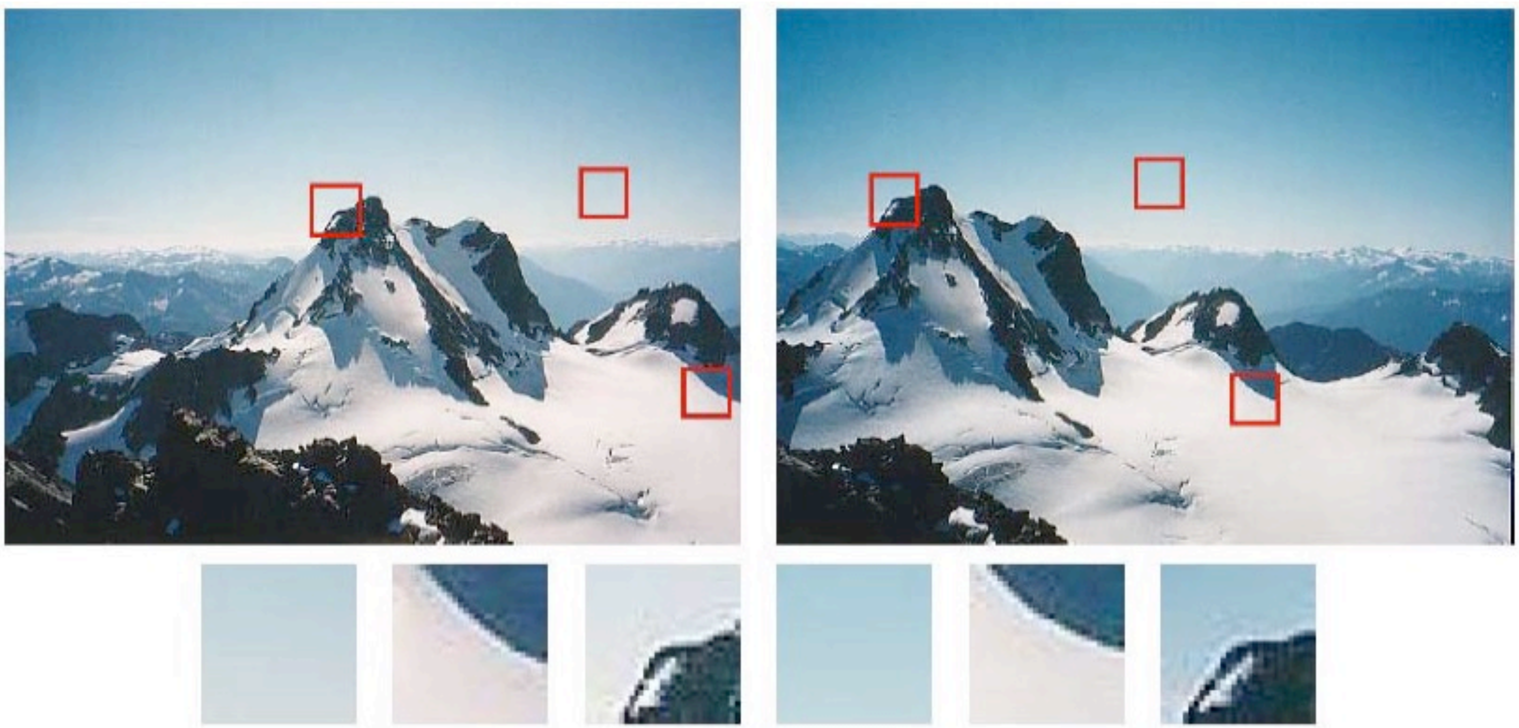
- We want to detect (at least some of) the same points in both images.
- Yet we have to run the detection procedure independently per image

An example of bad repeatability!

What is a good interest point?

Distinctiveness

- We want to be able to reliably determine which point goes with which
- Some points can be localised or matched with higher accuracy than others



More applications …

Feature points are used also for

- Image alignment
- 3D reconstruction
- Motion tracking (e.g. crowd motion)
- Object recognition
- Indexing and database retrieval
- Robot navigation
- etc.

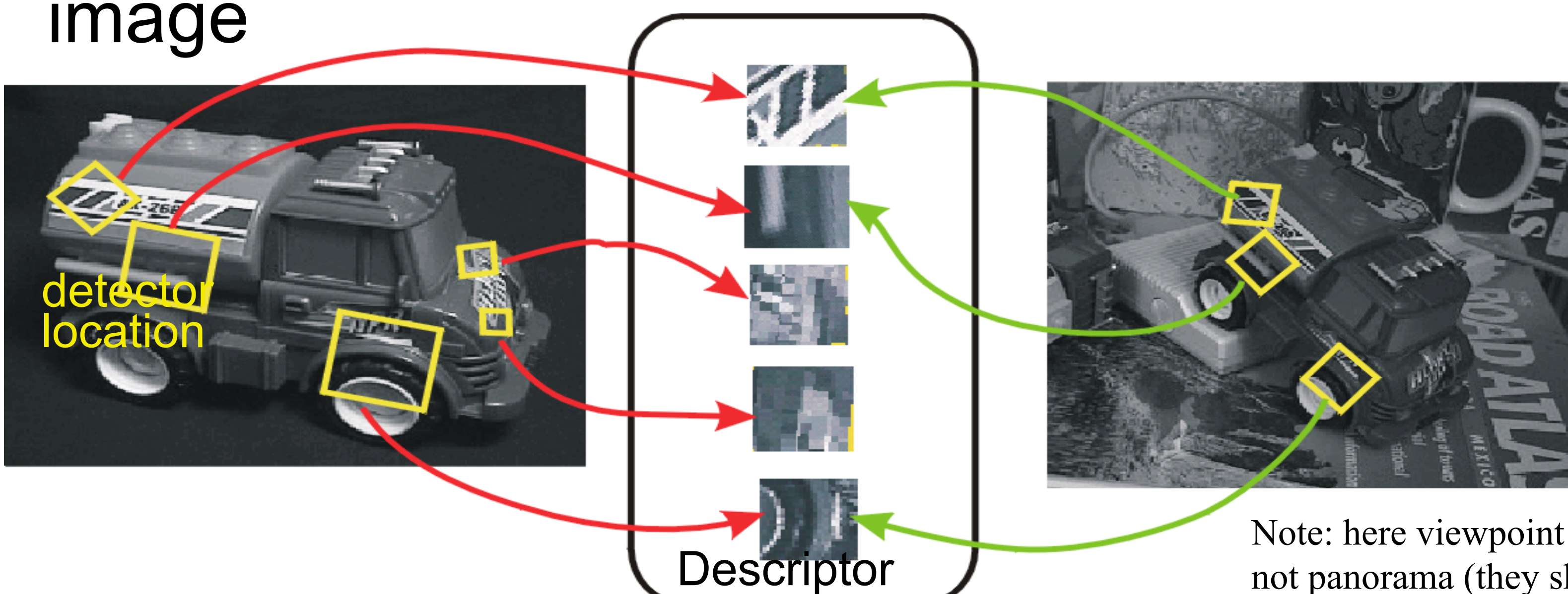
More applications …

Detector: detect same scene points independently in both images

Descriptor: encode local neighboring window

- Note how scale & rotation of window are the same in both image (but computed independently)

Correspondence: find most similar descriptor in other image

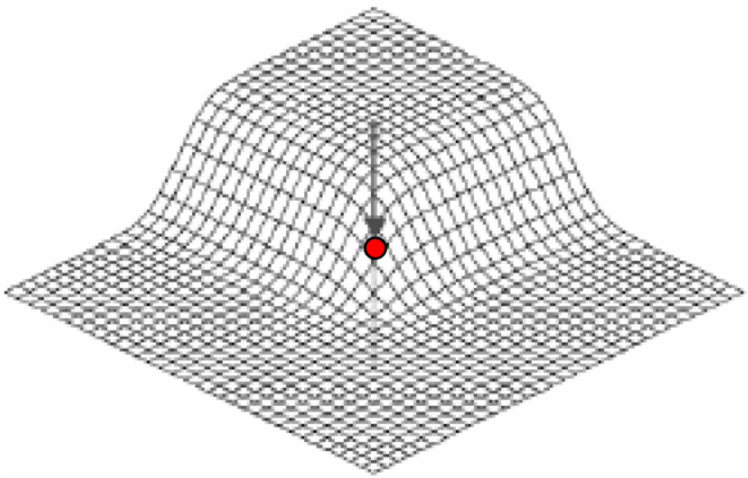


Note: here viewpoint is different, not panorama (they show off)

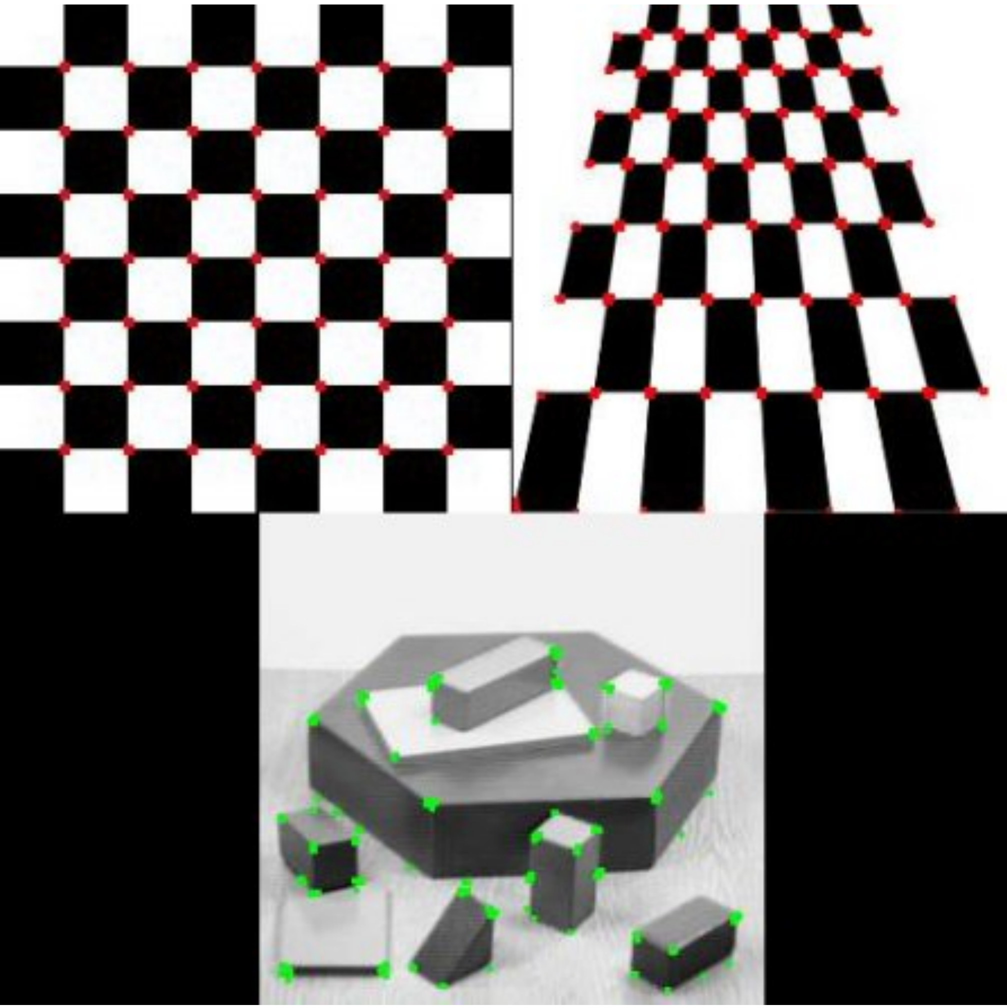
What is an interest point?

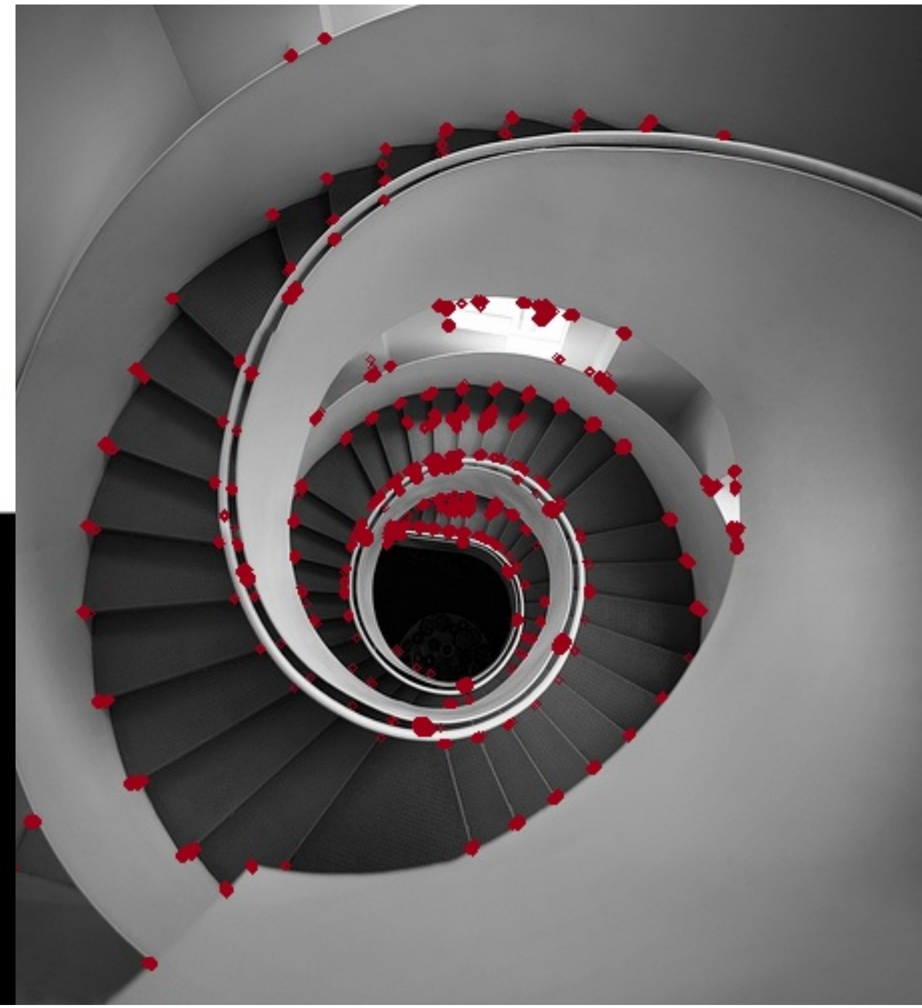
Expressive texture

- The point at which the direction of the boundary of object changes abruptly
- Intersection point between two or more edge segments



Synthetic & real interest points





Properties of interest point detectors

Detect all (or most) true interest points

No false interest points

Well localized

Robust with respect to noise

Efficient detection

Harris corner detector

Proposed by

- Chris Harris and Mike Stephens
- 1988

A COMBINED CORNER AND EDGE DETECTOR

Chris Harris & Mike Stephens

Plessey Research Roke Manor, United Kingdom

Consistency of image edge filtering is of prime importance for 3D interpretation of image sequences using feature tracking algorithms. To cater for image regions containing texture and isolated features, a combined corner and edge detector based on the local auto-correlation function is utilised, and it is shown to perform with good consistency on natural imagery.

INTRODUCTION

The problem we are addressing in Alvey Project MMI149 is that of using computer vision to understand the unconstrained 3D world, in which the viewed scenes will in general contain too wide a diversity of objects for top-down recognition techniques to work. For example, we desire to obtain an understanding of natural scenes, containing roads, buildings, trees, bushes, etc., as typified by the two frames from a sequence illustrated in Figure 1. The solution to this problem that we are pursuing is to use a computer vision system based upon motion analysis of a monocular image sequence from a mobile camera. By extraction and tracking of image features, representations of the 3D analogues of these features can be constructed.

To enable explicit tracking of image features to be performed, the image features must be discrete, and not form a continuum like texture, or edge pixels (edgels). For this reason, our earlier work[1] has concentrated on the extraction and tracking of feature-points or corners, since they are discrete, reliable and meaningful[2]. However, the lack of connectivity of feature-points is a major limitation in our obtaining higher level descriptions, such as surfaces and objects. We need the richer information that is available from edges[3].

THE EDGE TRACKING PROBLEM

Matching between edge images on a pixel-by-pixel basis works for stereo, because of the known epi-polar camera geometry. However for the motion problem, where the camera motion is unknown, the aperture problem prevents us from undertaking explicit edgel matching. This could be overcome by solving for the motion beforehand, but we are still faced with the task of tracking each individual edge pixel and estimating its 3D location from, for example, Kalman Filtering. This approach is unattractive in comparison with assembling the edgels into edge segments, and tracking these segments as the features.

Now, the unconstrained imagery we shall be considering will contain both curved edges and texture of various scales. Representing edges as a set of straight line fragments[4], and using these as our discrete features will be inappropriate, since curved lines and texture edges can be expected to fragment differently on each image of the sequence, and so be untrackable. Because of ill-conditioning, the use of parametrised curves (eg. circular arcs) cannot be expected to provide the solution, especially with real imagery.

a *b*

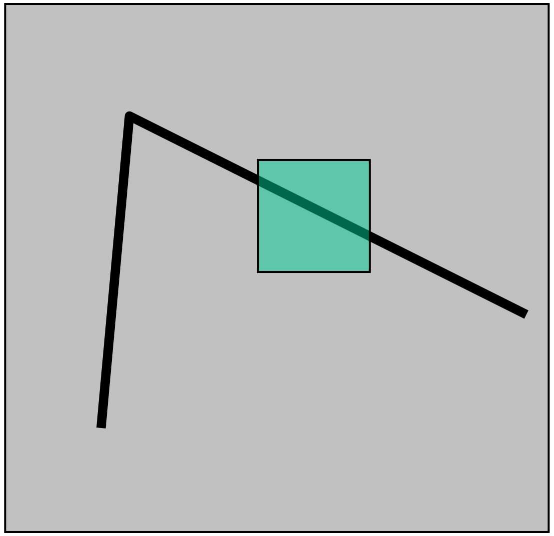
Figure 1. Pair of images from an outdoor sequence.

AVC 1988 doi:10.5244/C.2.23

Harris corner detector

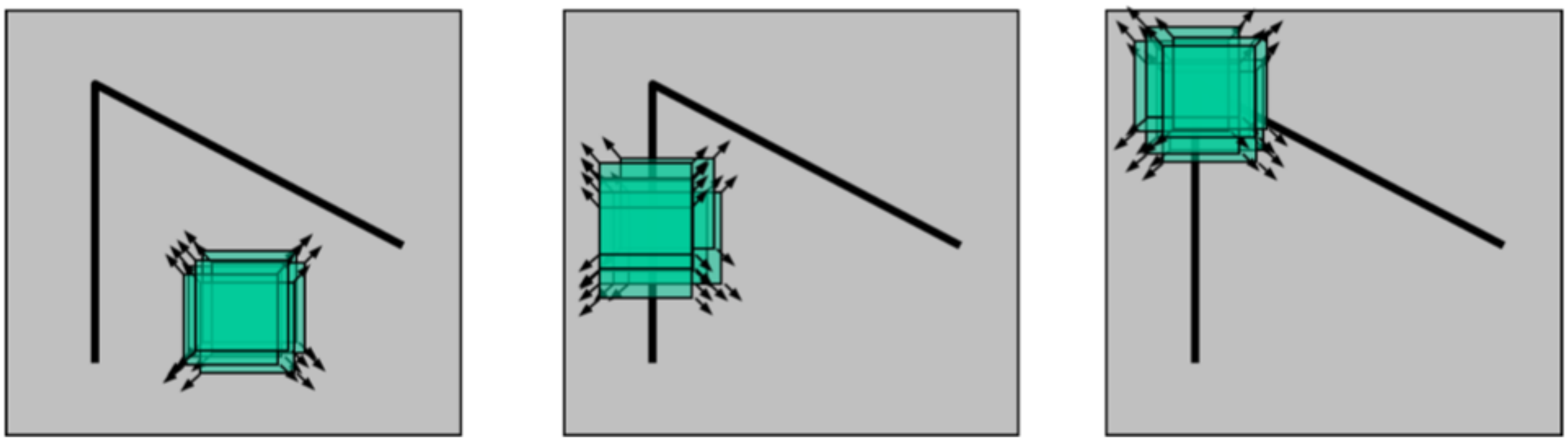
The basic idea

- We should easily localize the point by looking through a small window
- Shifting a window in *any direction* should give *a large change* in intensity



Harris corner detector

The basic idea



"flat" region:
no change as shift window in all directions

"edge":
no change as shift window along the edge direction

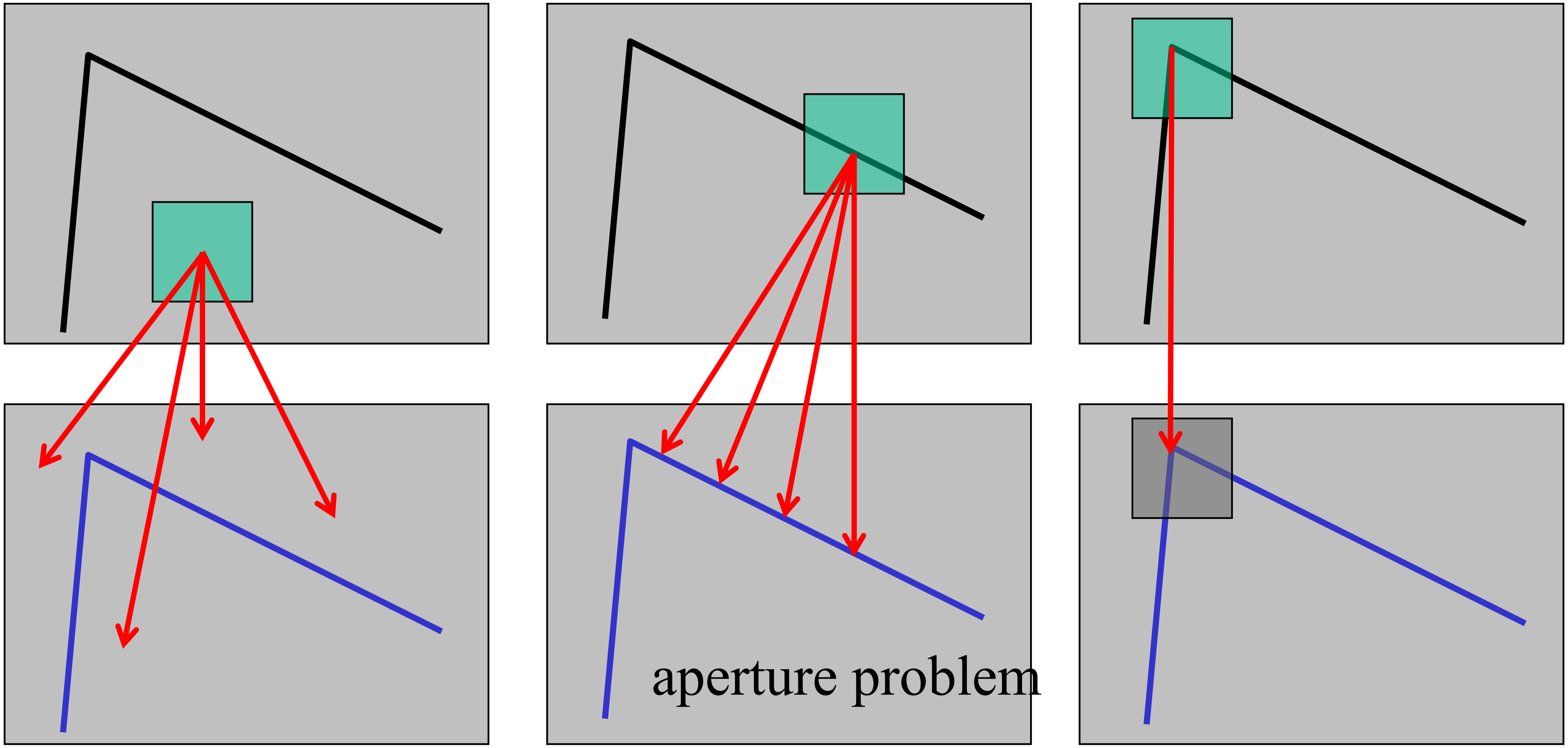
"corner":
significant change as shift window in all directions

Slide adapted from Darya Frolova, Denis Simakov, Weizmann

Harris corner detector

Why do we prefer the corners?

http://persci.mit.edu/demos/Motion&Form/demos/one-square/one-square.html



(a) (b) (c)

Harris corner detector - Mathematics

Window-averaged squared change of intensity induced by shifting the image data by displacement vector [u,v]:

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

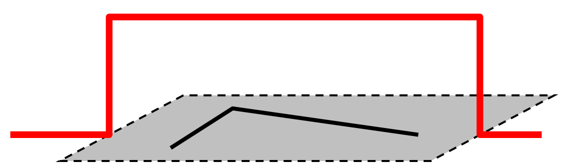
Sum over image region – area we are checking for corner

Window function

Shifted intensity

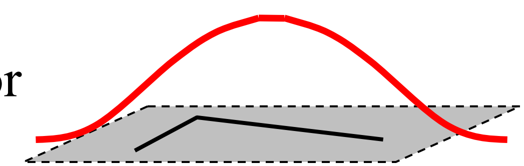
Intensity

Window function $w(x,y) =$



1 in window, 0 outside

or



Gaussian

Harris corner detector - Mathematics

Window-averaged squared change of intensity induced by shifting the image data by [u,v]:

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

For nearly constant patches, this will be near 0.
For very distinctive patches, this will be larger.
Hence... we want patches where $E(u,v)$ is LARGE.

Harris corner detector - Mathematics

Taylor Series for 2D Functions

$$f(x+u,y+v) \quad = \quad f(x,y)+uf_x(x,y)+vf_y(x,y)+\dots$$

First partial derivatives

$$\frac{1}{2!}\left[u^2 f_{xx}(x,y)+2uvf_{xy}(x,y)+v^2 f_{yy}(x,y)\right]+$$

Second partial derivatives

$$\frac{1}{3!}\left[u^3 f_{xxx}(x,y)+3u^2 v f_{xxy}(x,y)+3uv^2 f_{xyy}(x,y)+v^3 f_{yyy}(x,y)\right]$$

Third partial derivatives

+ higher order terms

Harris corner detector - Mathematics

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 approx is good

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

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

Plugging this into the formula on the previous slide…

Harris corner detector - Mathematics

$$
\begin{aligned}
E(u,v) &= \sum_{x,y} w(x,y)[I(x+u,y+v)-I(x,y)]^2 \\
&= \sum_{x,y} w(x,y)[I(x,y)+uI_x+vI_y-I(x,y)]^2 \\
&= \sum_{x,y} w(x,y)[uI_x+vI_y]^2 \\
&= \sum_{x,y} w(x,y)[u^2I_x^2+2uvI_xI_y+v^2I_y^2] \\
&= \sum_{x,y} w(x,y)[u\ v]\begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix} \\
&= [u\ v]\sum_{x,y} w(x,y)\begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}
\end{aligned}
$$

First order approx.

Rewrite in matrix form

Harris corner detector - Mathematics

$$E(u,v) \approx \begin{pmatrix} u & v \end{pmatrix} \sum_{x,y} w(x,y) \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix} \begin{pmatrix} u \\ v \end{pmatrix}$$

$$E(u,v) \cong [u,v] \; M \; \begin{bmatrix} u \\ v \end{bmatrix}$$

where M is a 2×2 matrix computed from image derivatives:

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

M is also called "structure tensor"

Harris corner detector - Mathematics

Intensity change in shifting window: eigenvalue analysis

$$E(u,v) \cong [u,v] \; M \; \begin{bmatrix} u \\ v \end{bmatrix}$$

λ_1, λ_2 – eigenvalues of M

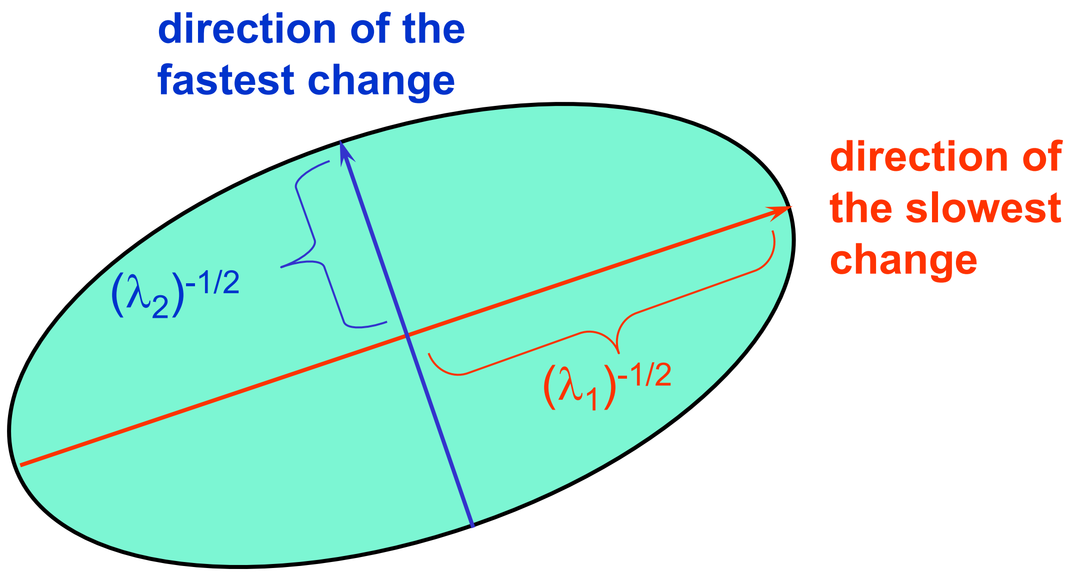
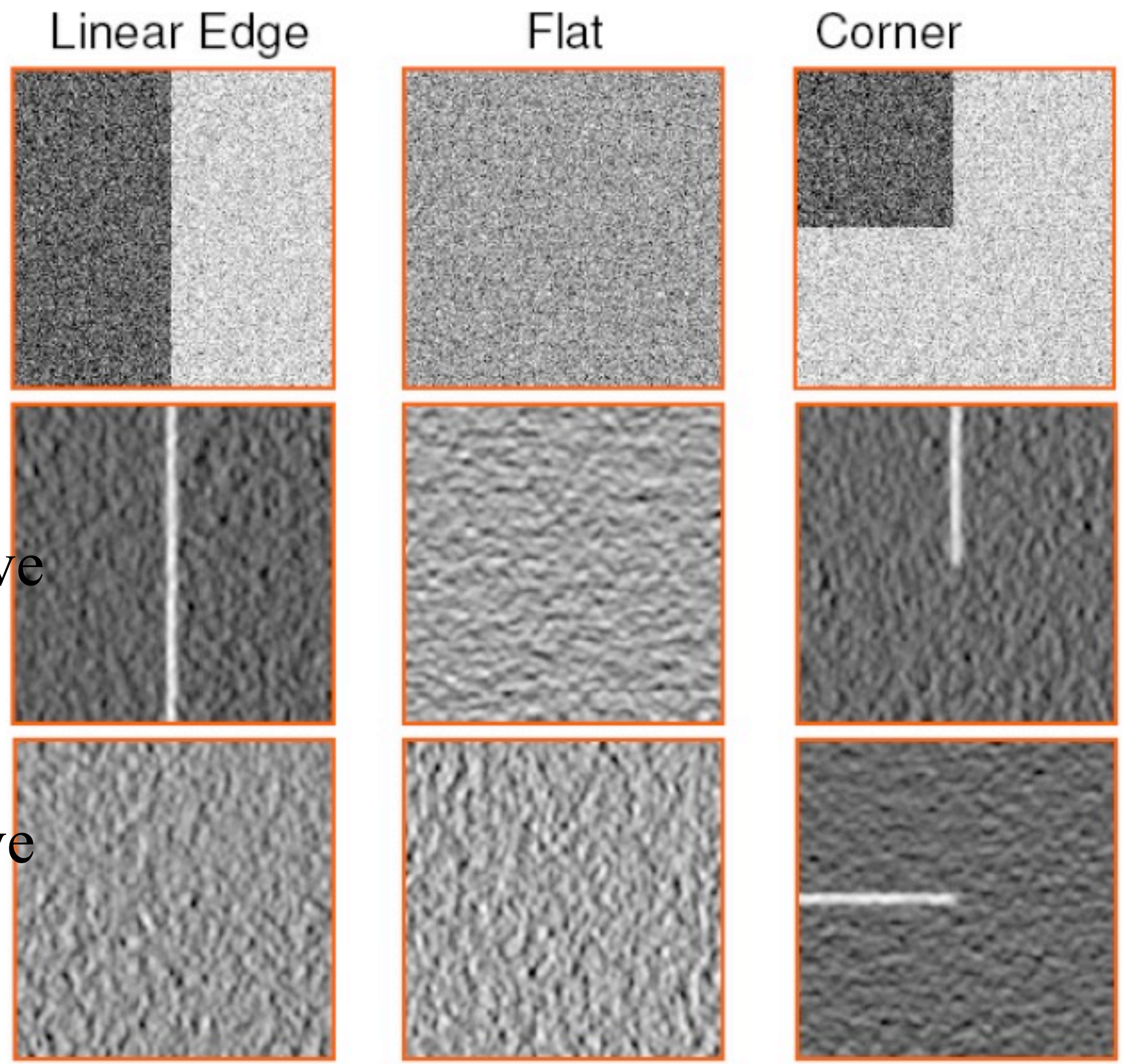


Image derivatives - examples



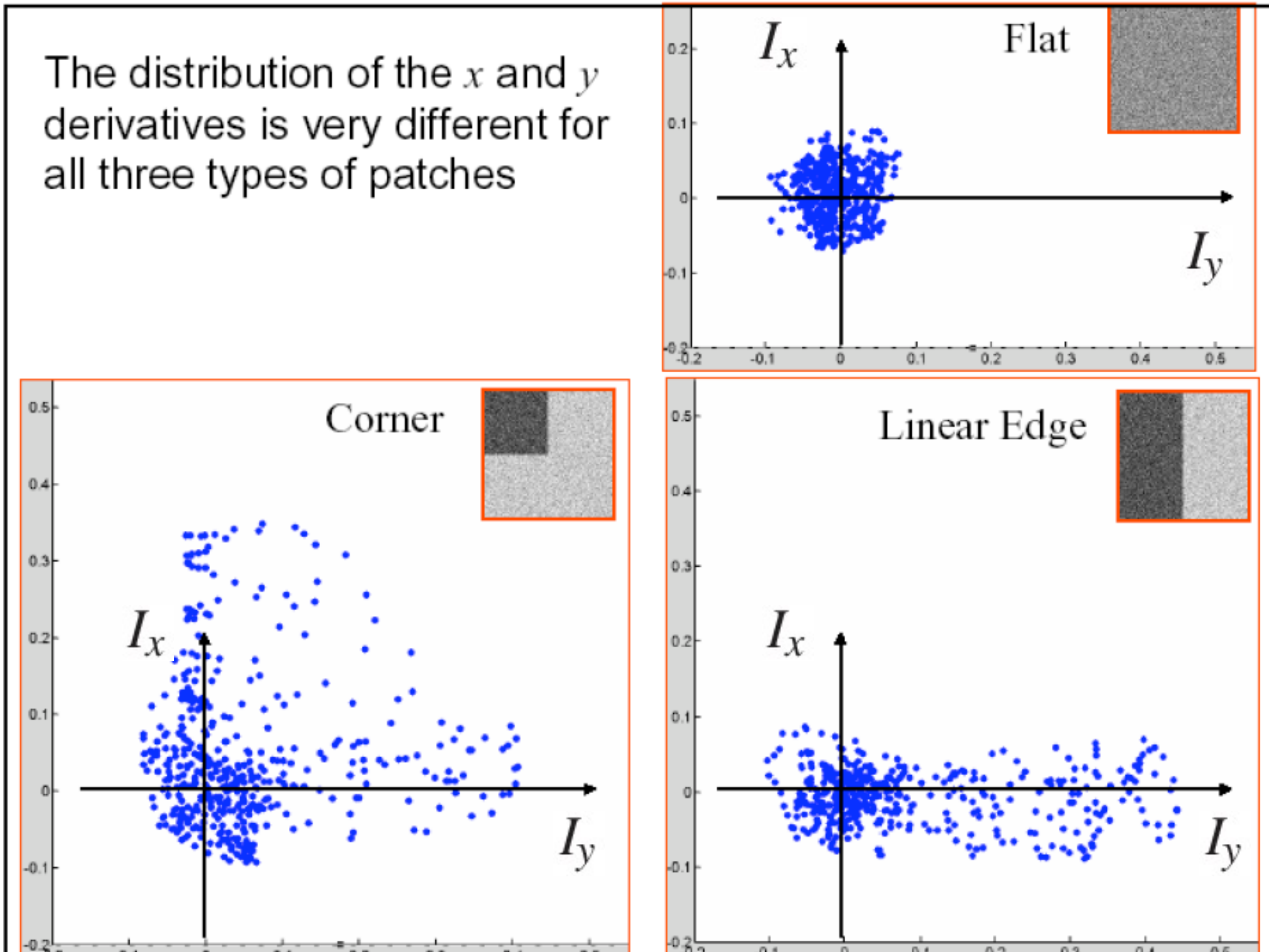
Input

Y derivative

X derivative

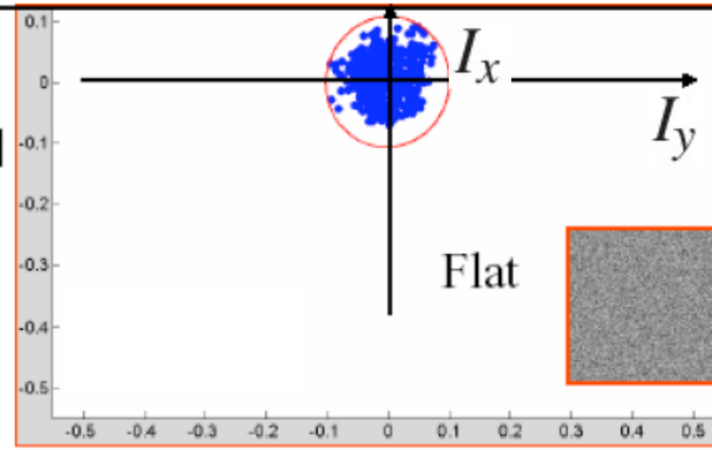
Another view

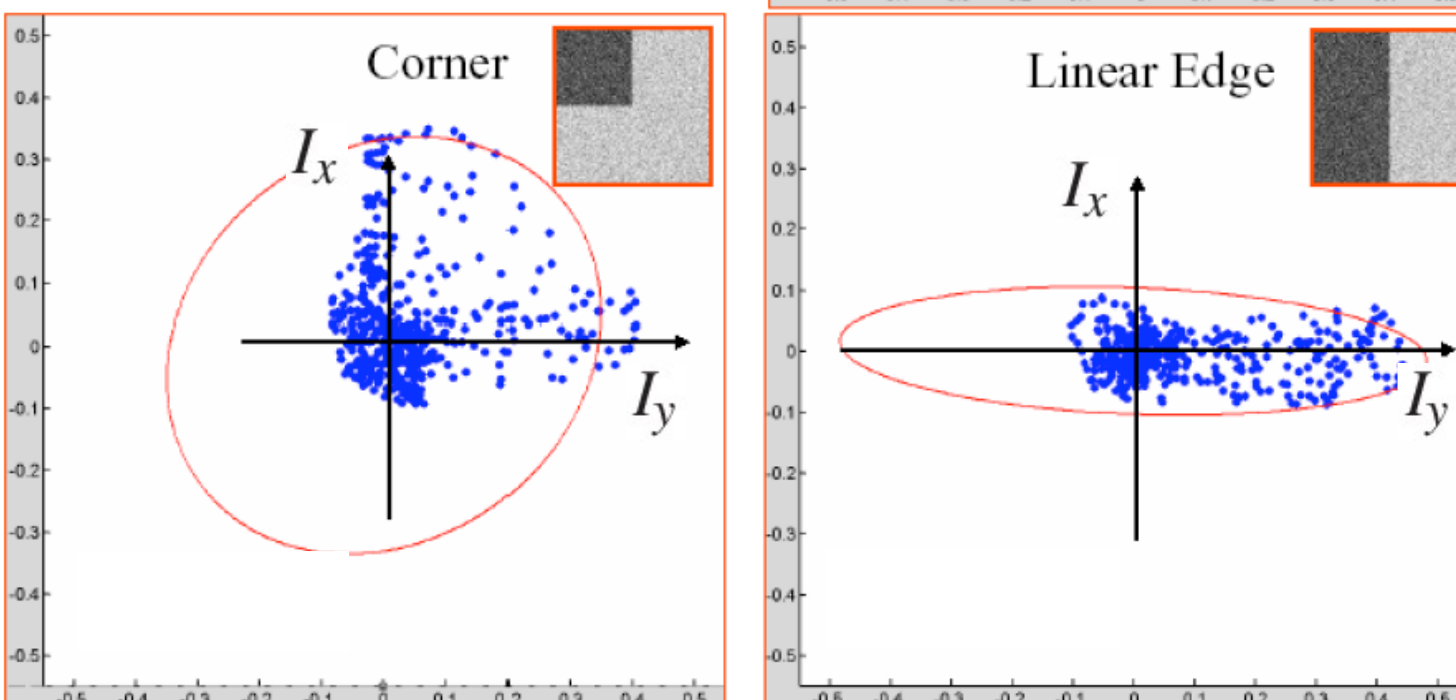
The distribution of the x and y derivatives is very different for all three types of patches



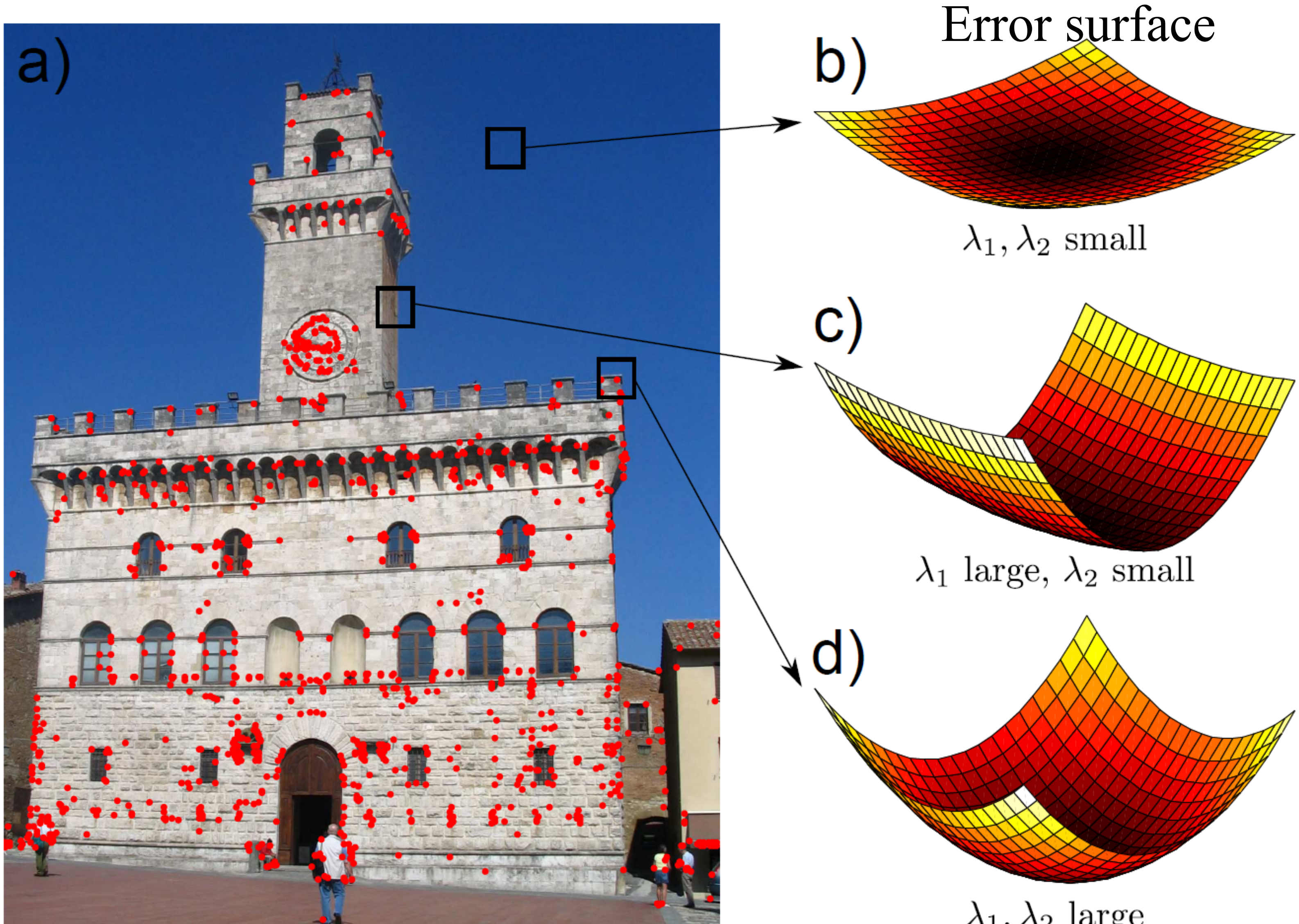
Another view

The distribution of x and y derivatives can be characterized by the shape and size of the principal component ellipse

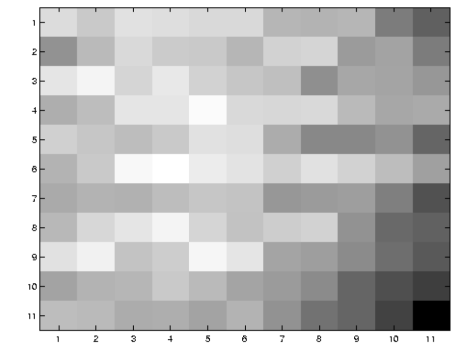




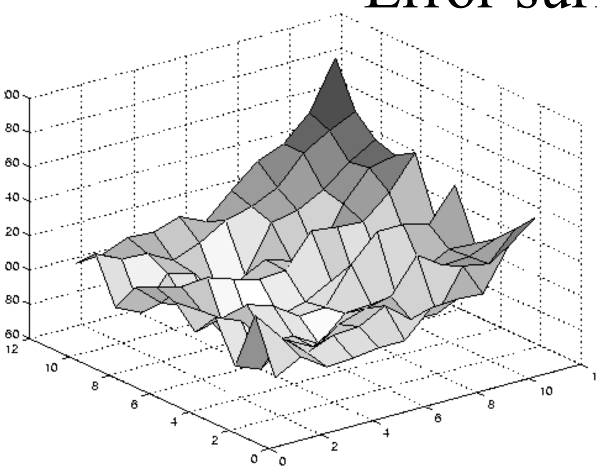
Selecting good features



Selecting good features

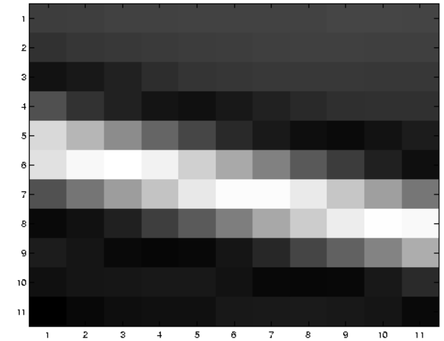


Error surface

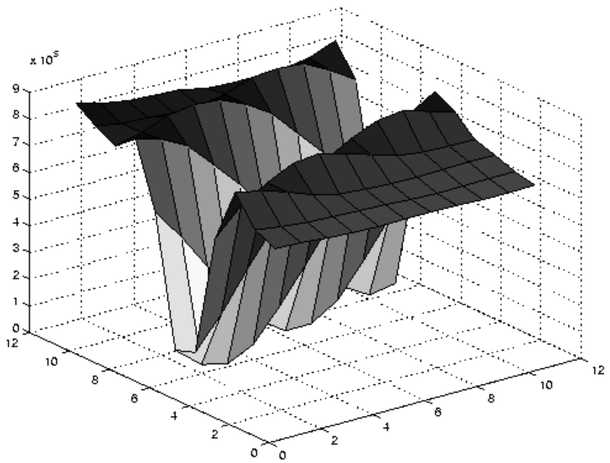


small λ_1, small λ_2

Selecting good features

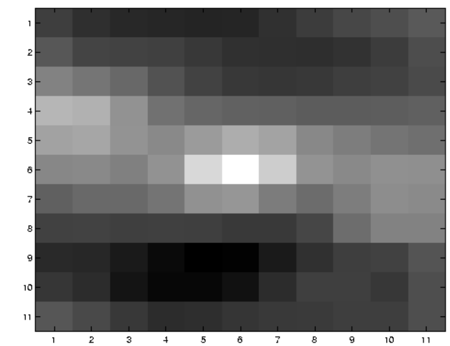


Error surface

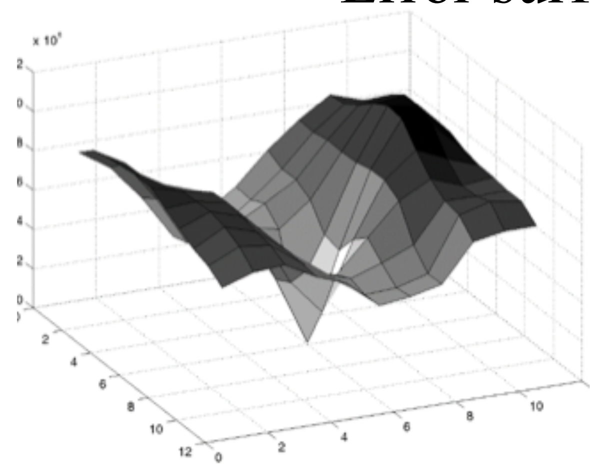


large λ_1, small λ_2

Selecting good features



Error surface

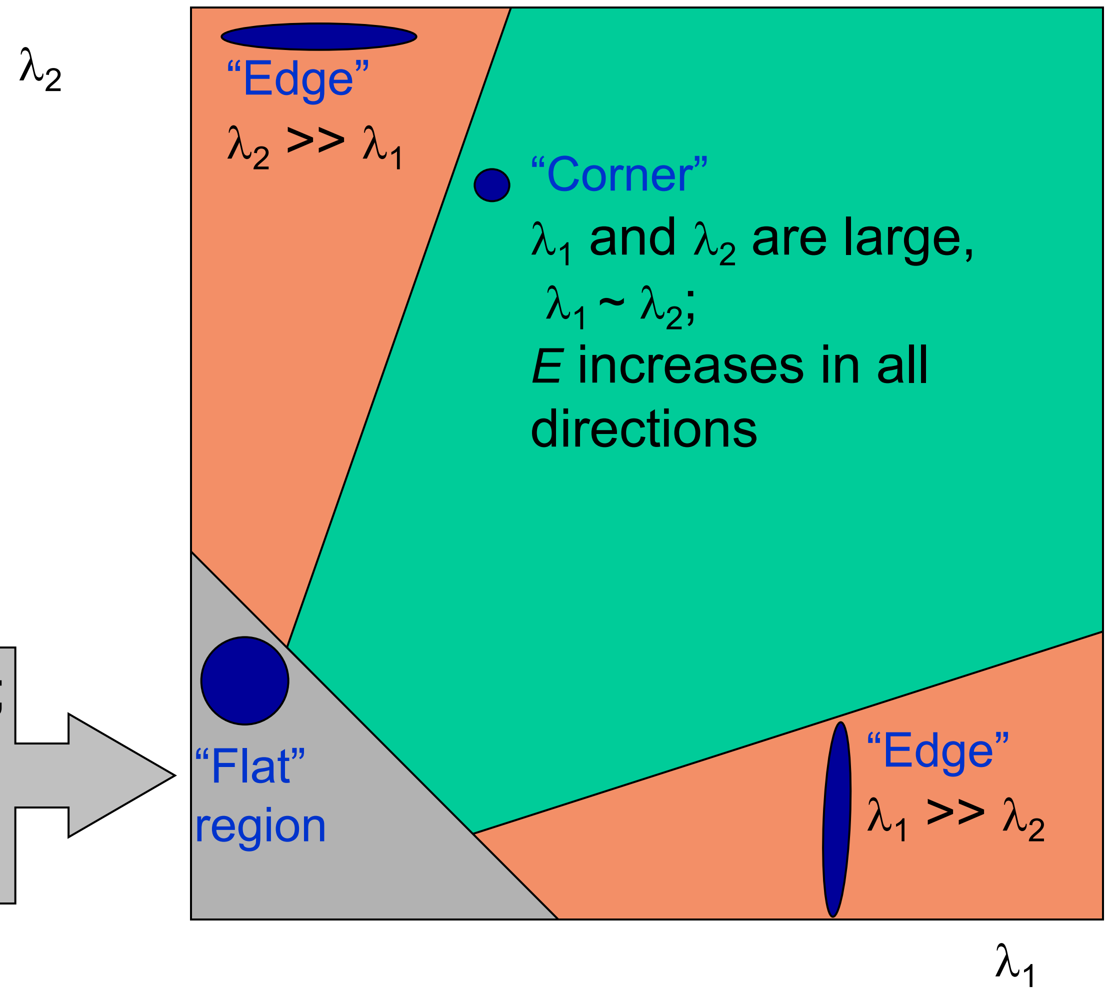


λ_1 and λ_2 are large

Harris corner detector - Mathematics

Classification of image points using eigenvalues of *M*:

λ_1 and λ_2 are small; *E* is almost constant in all directions



Harris corner detector - Mathematics

Measure of corner response:

$$R = \det M - k\left(\operatorname{trace} M\right)^2$$

This expression does not requires computing the eigenvalues.

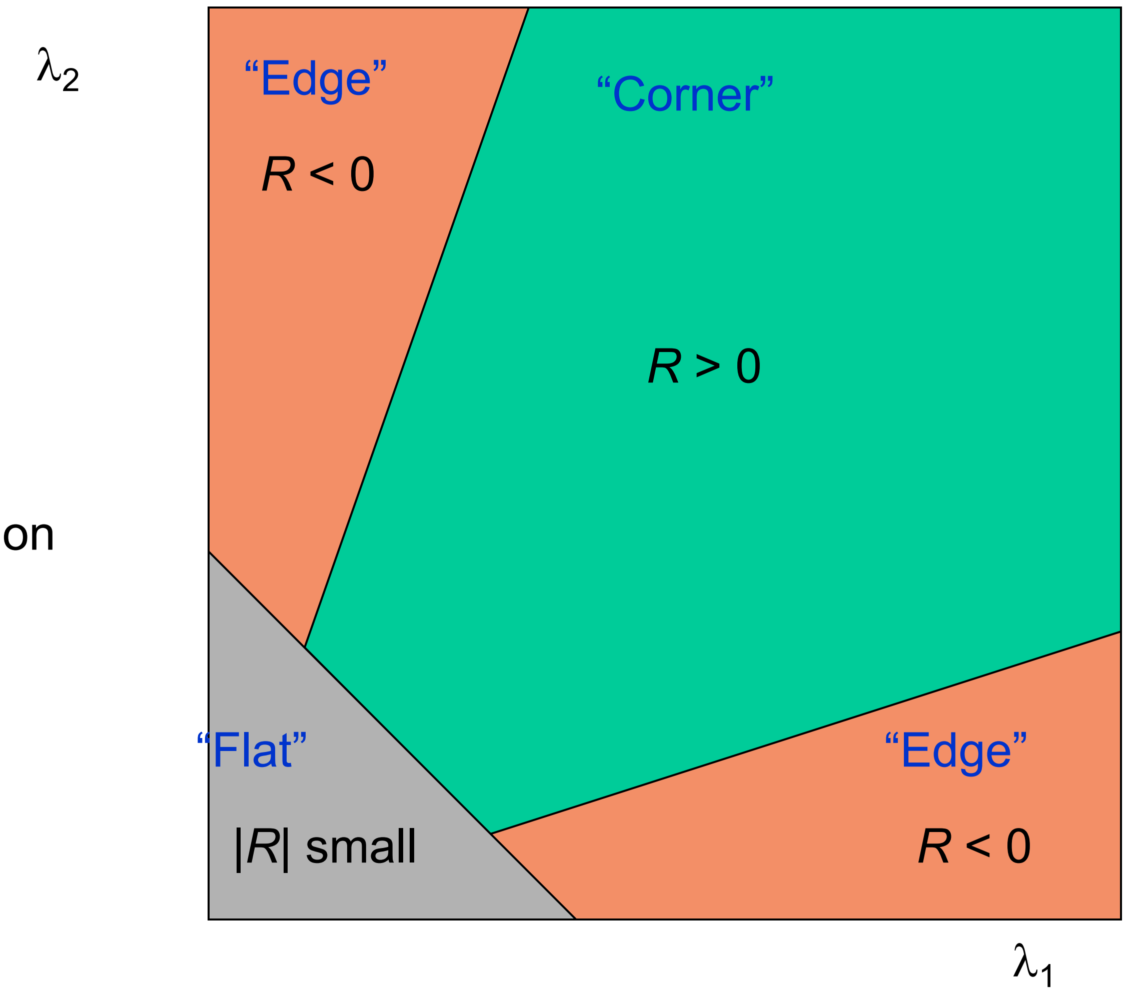
$$\det M = \lambda_1 \lambda_2$$
$$\operatorname{trace} M = \lambda_1 + \lambda_2$$

(*k* – empirical constant, *k* = 0.04-0.06)

In MATLAB corner(*image*)

Harris corner detector - Mathematics

- R depends only on eigenvalues of M
- R is large for a corner
- R is negative with large magnitude for an edge
- $|R|$ is small for a flat region



Harris corner detector – work flow

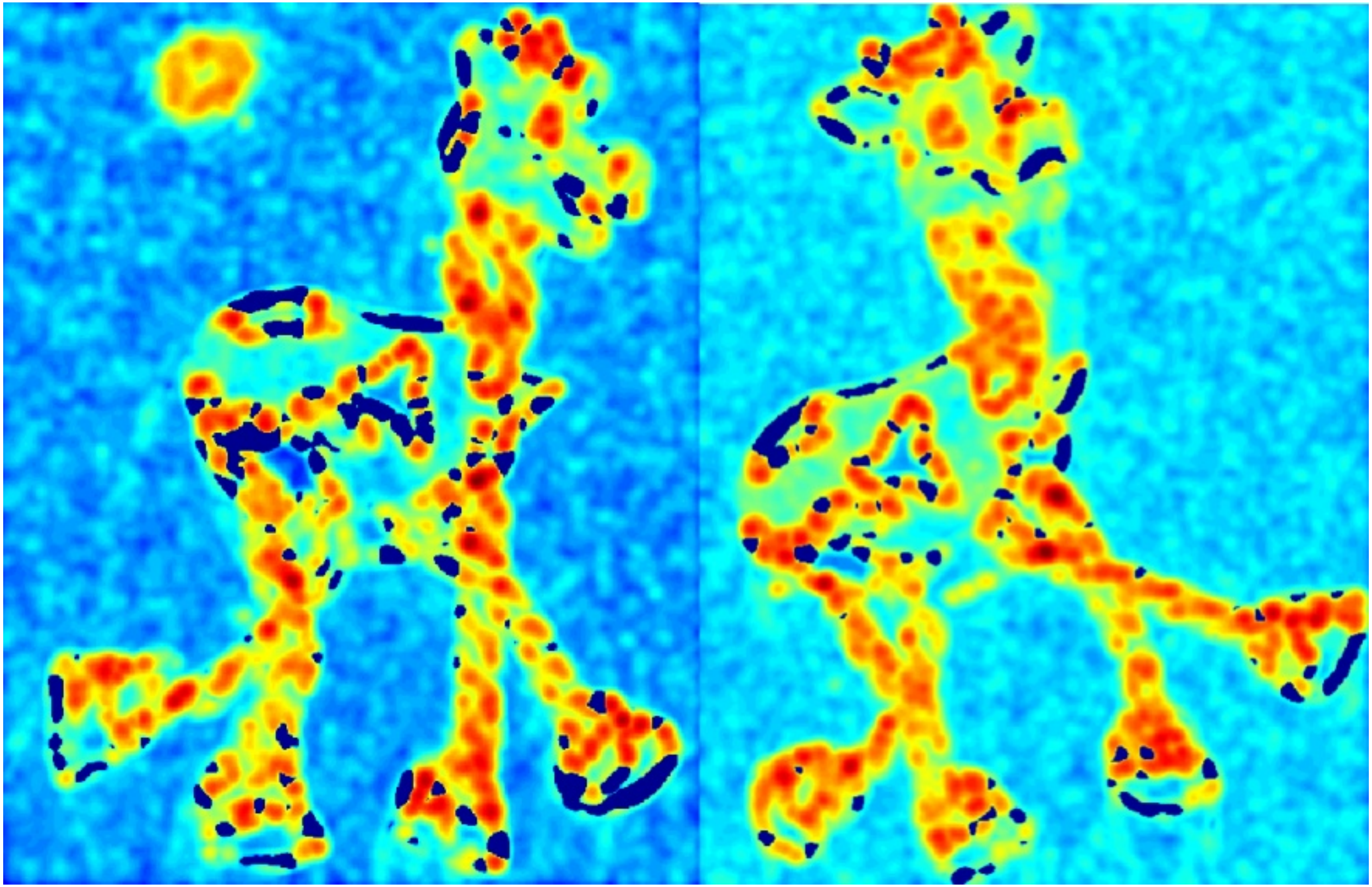
The Algorithm:

- Find points with large corner response function *R* ($R >$ threshold)
- Take the points of local maxima of *R*

Harris corner detector – work flow

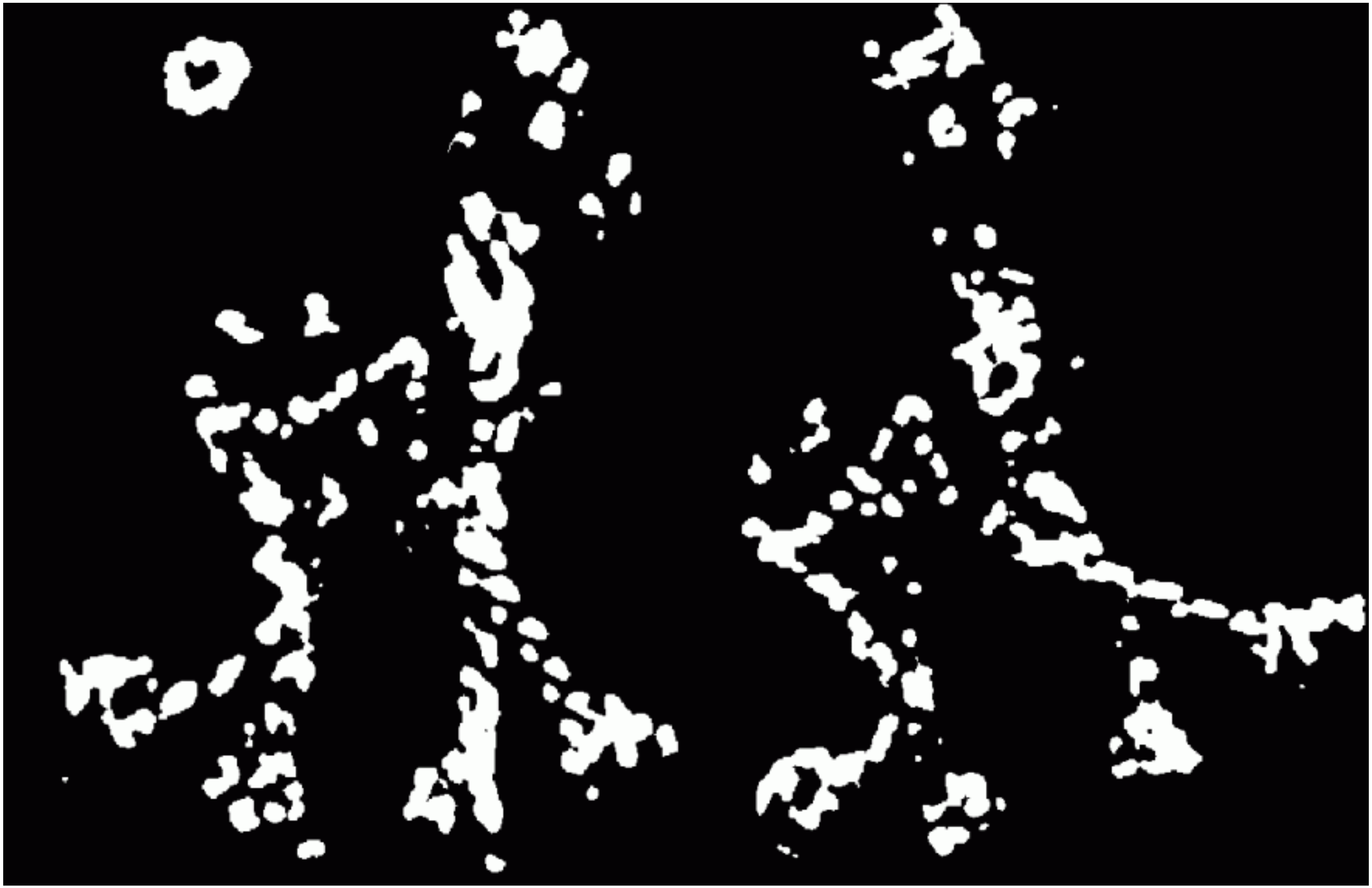
Harris corner detector – work flow

Compute corner response R



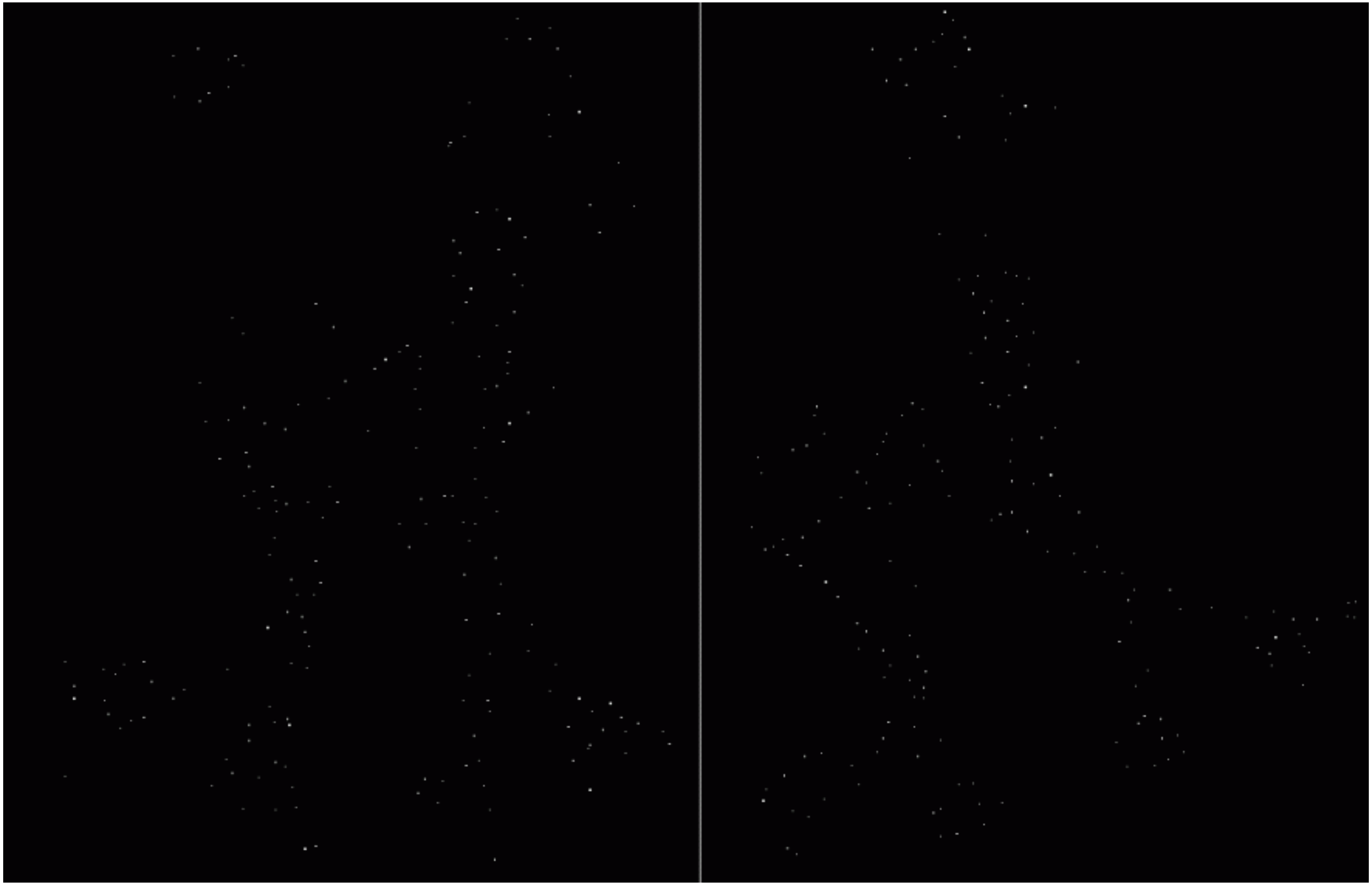
Harris corner detector – work flow

Find points with large corner response: R>threshold



Harris corner detector – work flow

Take only the points of local maxima of R



Harris corner detector – work flow

Harris detector: summary

- Average intensity change in direction $[u,v]$ can be expressed as a bilinear form:

$$E(u,v) \cong [u,v] \; M \begin{bmatrix} u \\ v \end{bmatrix}$$

- Describe a point in terms of eigenvalues of M: *measure of corner response*

$$R = \lambda_1 \lambda_2 - k\left(\lambda_1 + \lambda_2\right)^2$$

- A good (corner) point should have a *large intensity change* in *all directions*, i.e. R should be large positive

References

Reading material

- Richard Szeliski, Computer Vision: Algorithms and Applications, Sec. 4.1.1
- Available online: http://szeliski.org/Book/drafts/SzeliskiBook_20100903_draft.pdf

Some slides are adapted from

- Darya Frolova, Denis Simakov, Weizmann Institute.
- Advances in Computer Vision, MIT, Antonio Torralba
- UCF Computer Vision Lectures, Mubarak Shah