Interest Point Detection

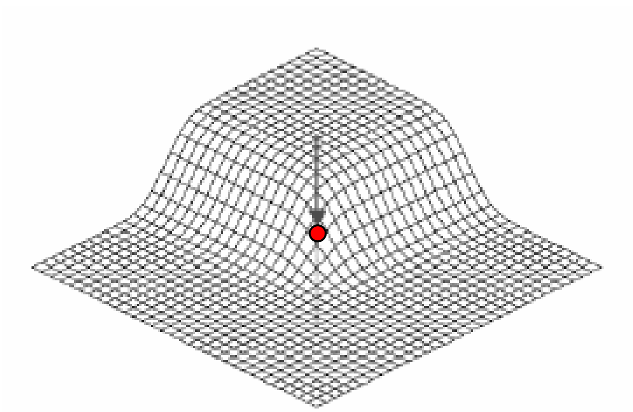
Lecture-4

Contents

- Harris Corner Detector
- Sum of Squares Differences (SSD)
  - Corrleation
- Taylor Series
- Eigen Vectors and Eigen Values
- Invariance and co-variance

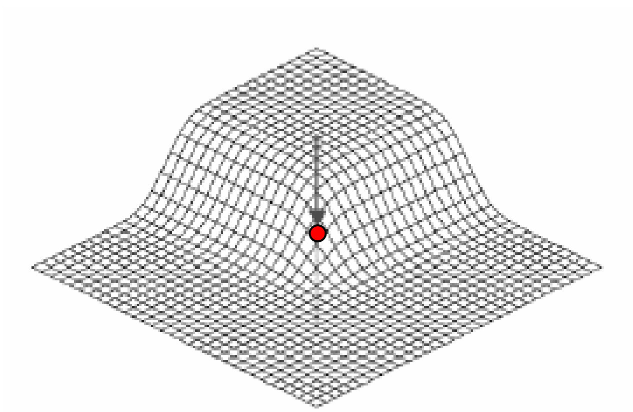
What is an interest point

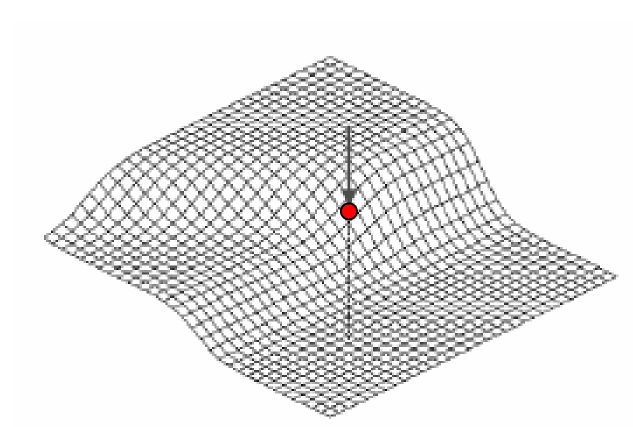
- Expressive texture
  - The point at which the direction of the boundary of object changes abruptly
  - Intersection point between two or more edge segments



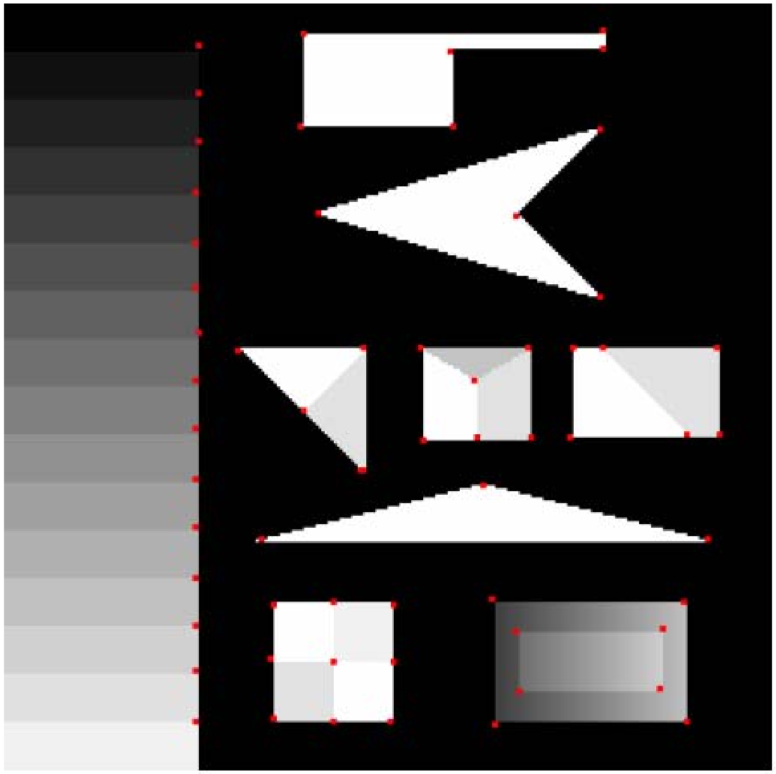
What is an interest point

- 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



Corners are indicated in red

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

Possible Approaches to Corner Detection

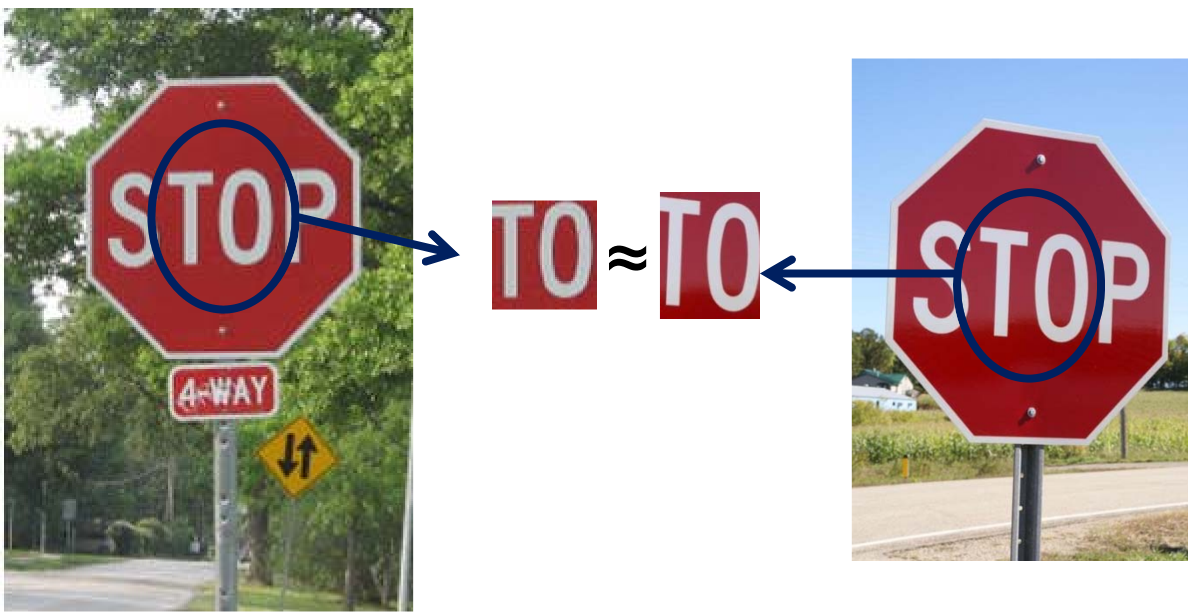
- Based on brightness of images
  - Usually image derivatives
- Based on boundary extraction
  - First step edge detection
  - Curvature analysis of edges

Where can we use it?

- Automate object tracking
- Point matching for computing disparity
- Stereo calibration
  - Estimation of fundamental matrix
- Motion based segmentation
- Recognition
- 3D object reconstruction
- Robot navigation
- Image retrieval and indexing

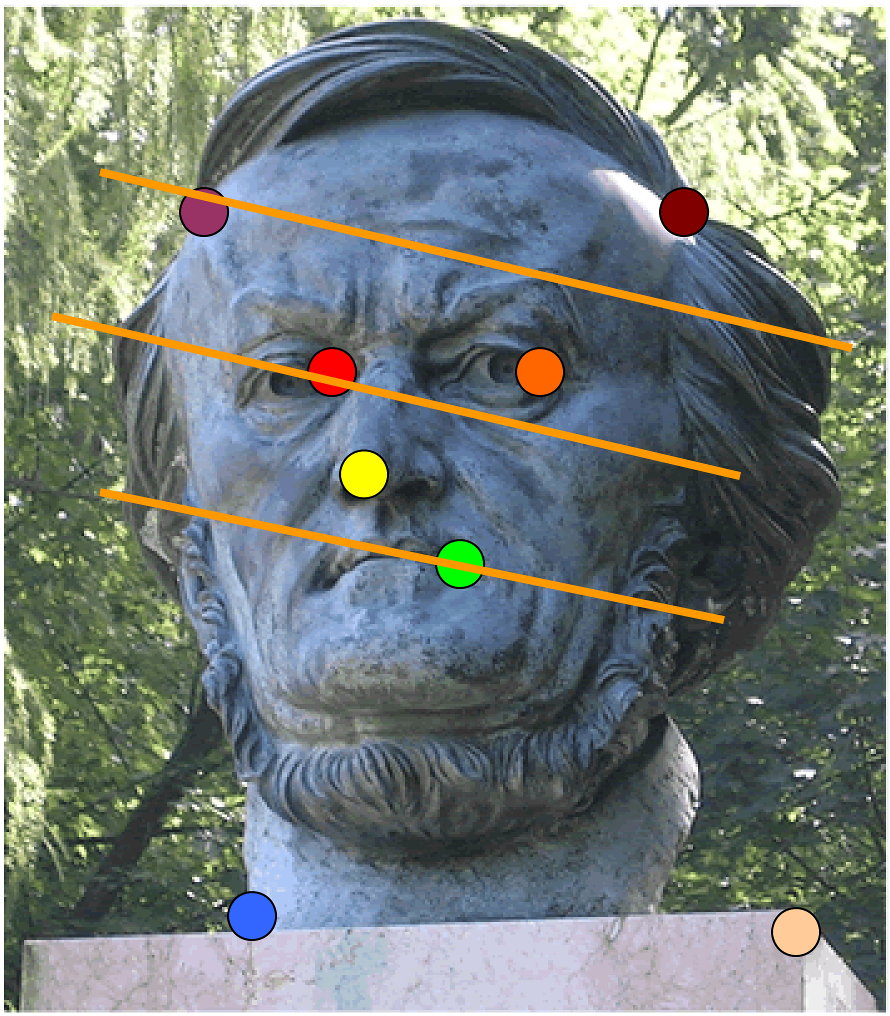
Correspondence across views

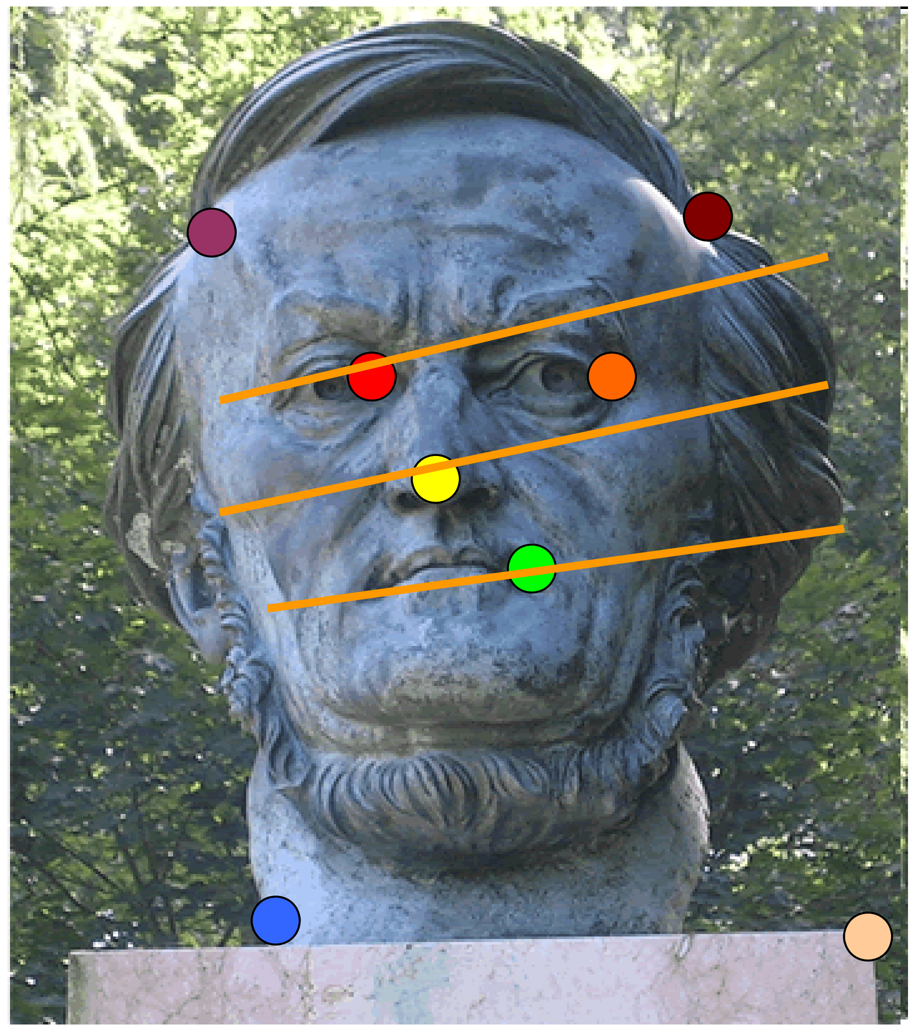
- Correspondence: matching points, patches, edges, or regions across images



Slide Credit: James Hays

Example: estimating "fundamental matrix" that corresponds two views





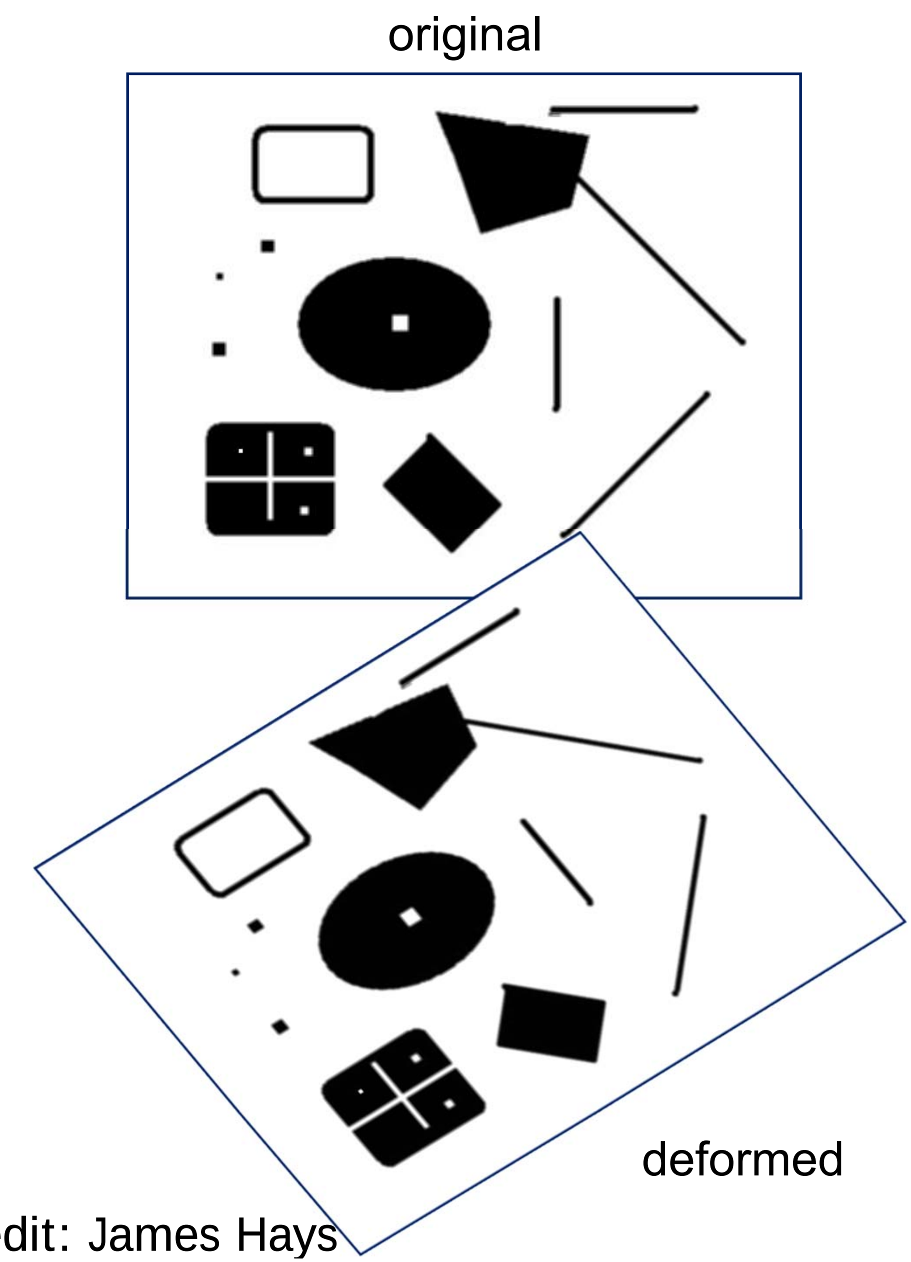
Slide from Silvio Savarese

Example: structure from motion

Slide Credit: James Hays

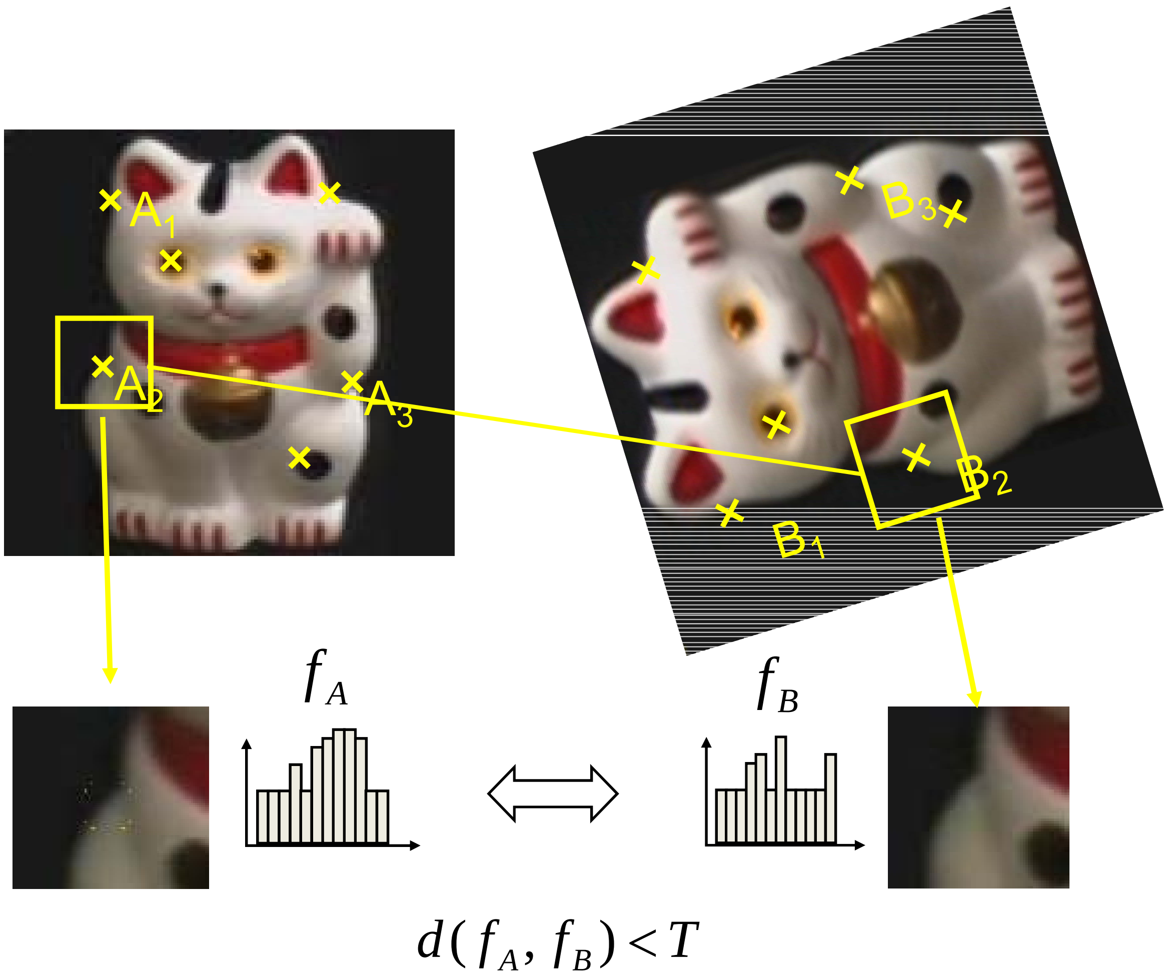
This class: interest points

- Suppose you have to click on some point, go away and come back after I deform the image, and click on the same points again.
  - Which points would you choose?



Slide Credit: James Hays

Overview of Keypoint Matching



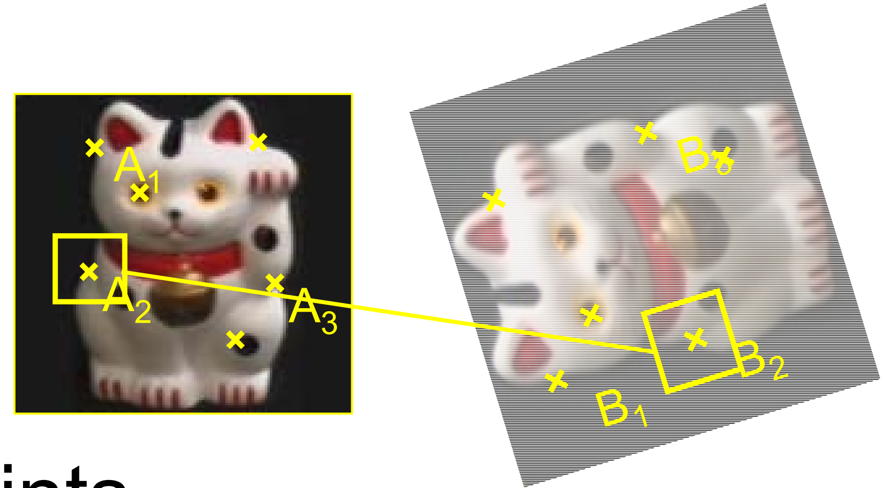
1. **Find a set of distinctive key-points**
2. **Define a region around each keypoint**
3. **Extract and normalize the region content**
4. **Compute a local descriptor from the normalized region**
5. **Match local descriptors**

K. Grauman, B. Leibe

Goals for Keypoints

Detect points that are *repeatable* and *distinctive*

Key trade-offs



Detection of interest points

More Repeatable	More Points
Robust detection	Robust to occlusion
Precise localization	Works with less texture

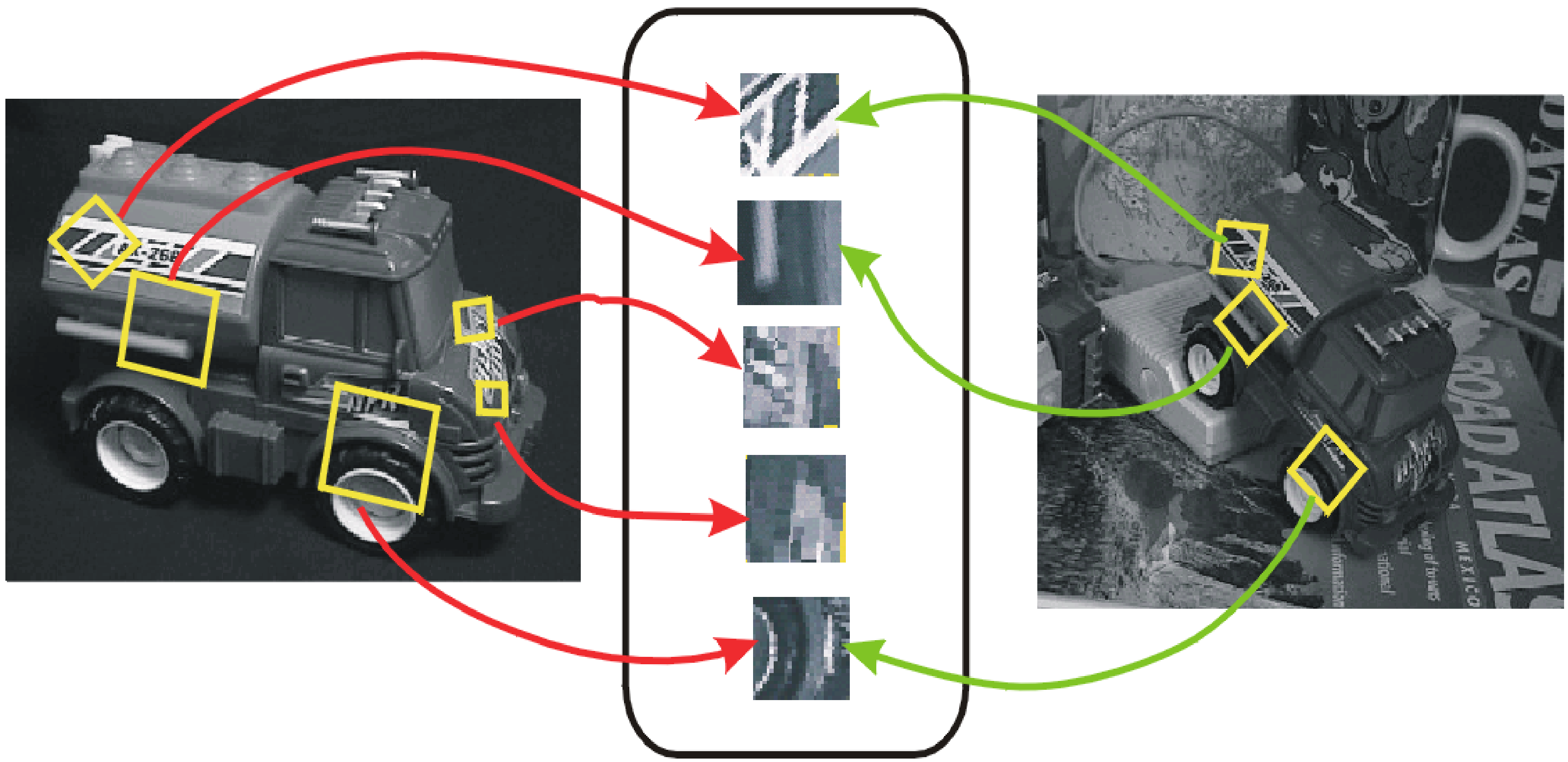
Description of patches

More Distinctive	More Flexible
Minimize wrong matches	Robust to expected variations
	Maximize correct matches

Slide Credit: James Hays

Invariant Local Features

Image content is transformed into local feature coordinates that are invariant to translation, rotation, scale, and other imaging parameters



Features Descriptors

Choosing interest points

Where would you tell your friend to meet you?

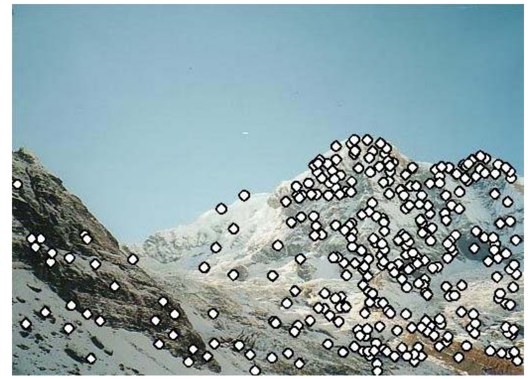
Slide Credit: James Hays

Feature extraction: Corners

Slides from Rick Szeliski, Svetlana Lazebnik, and Kristin Grauman

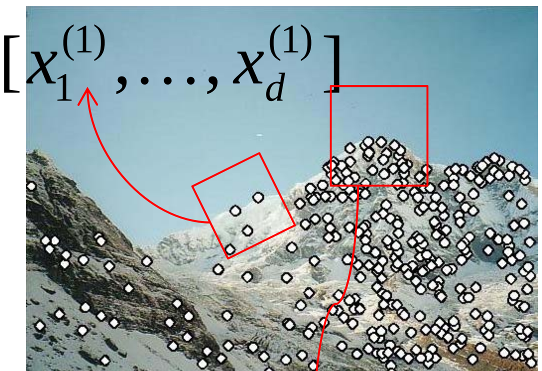
Local features: main components

1) **Detection:** Identify the interest points



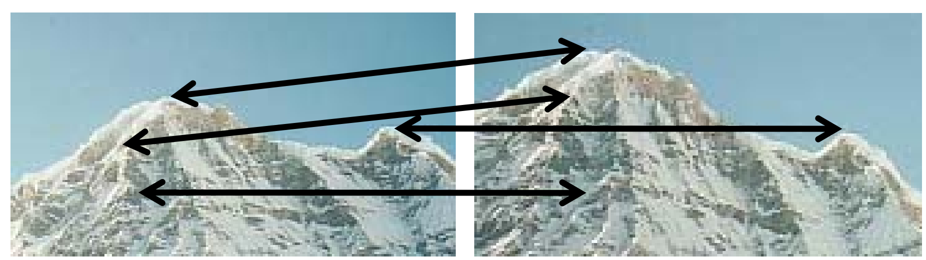
2) **Description** :Extract feature vector descriptor surrounding each interest point.

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



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

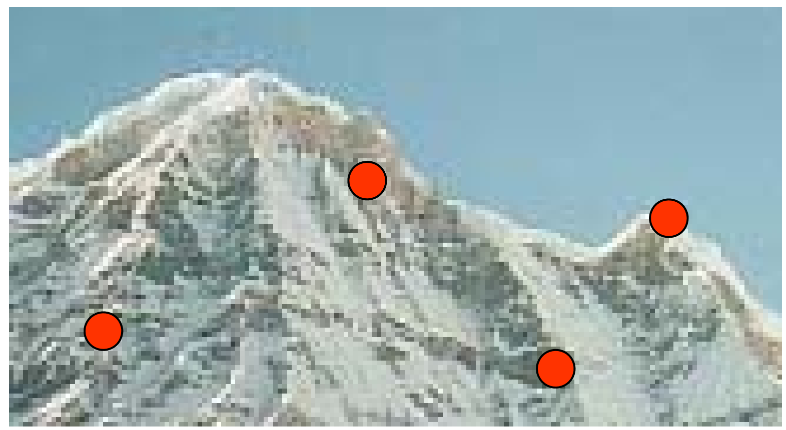
3) **Matching:** Determine correspondence between descriptors in two views



Kristen Grauman

Goal: interest operator repeatability

- We want to detect (at least some of) the same points in both images.

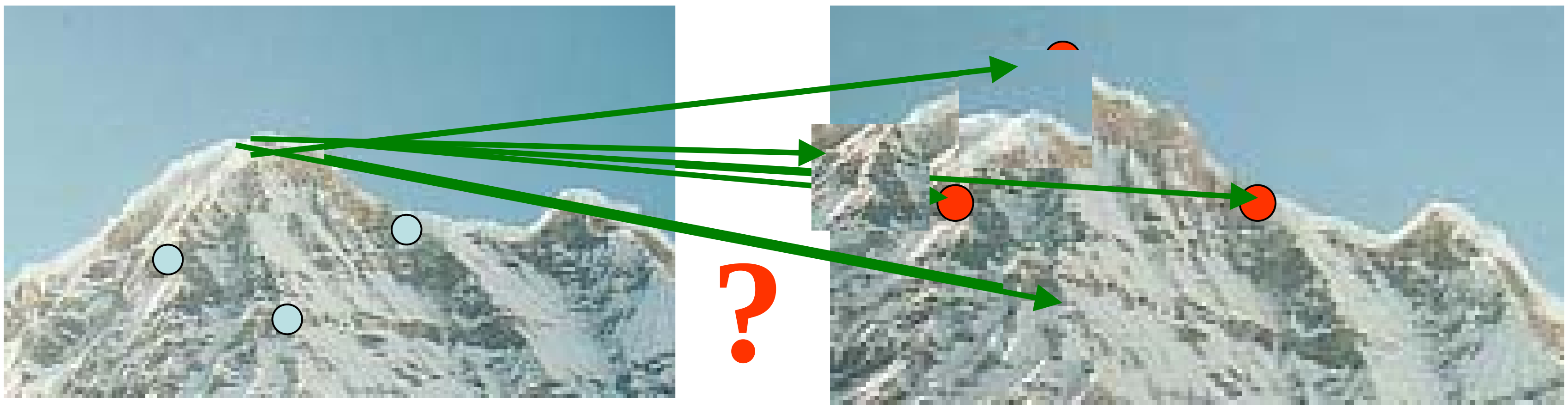
No chance to find true matches!

- Yet we have to be able to run the detection procedure *independently* per image.

Kristen Grauman

Goal: descriptor distinctiveness

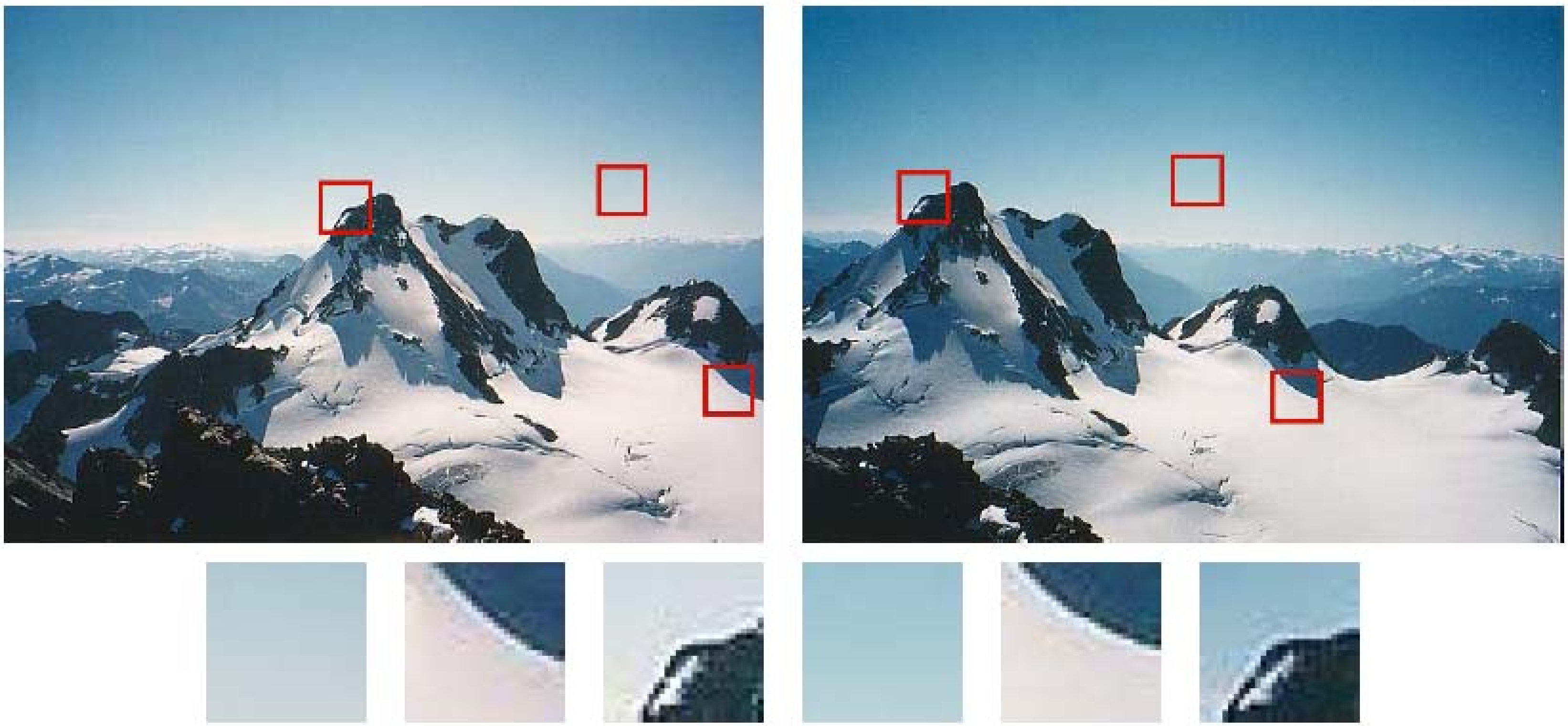
- We want to be able to reliably determine which point goes with which.



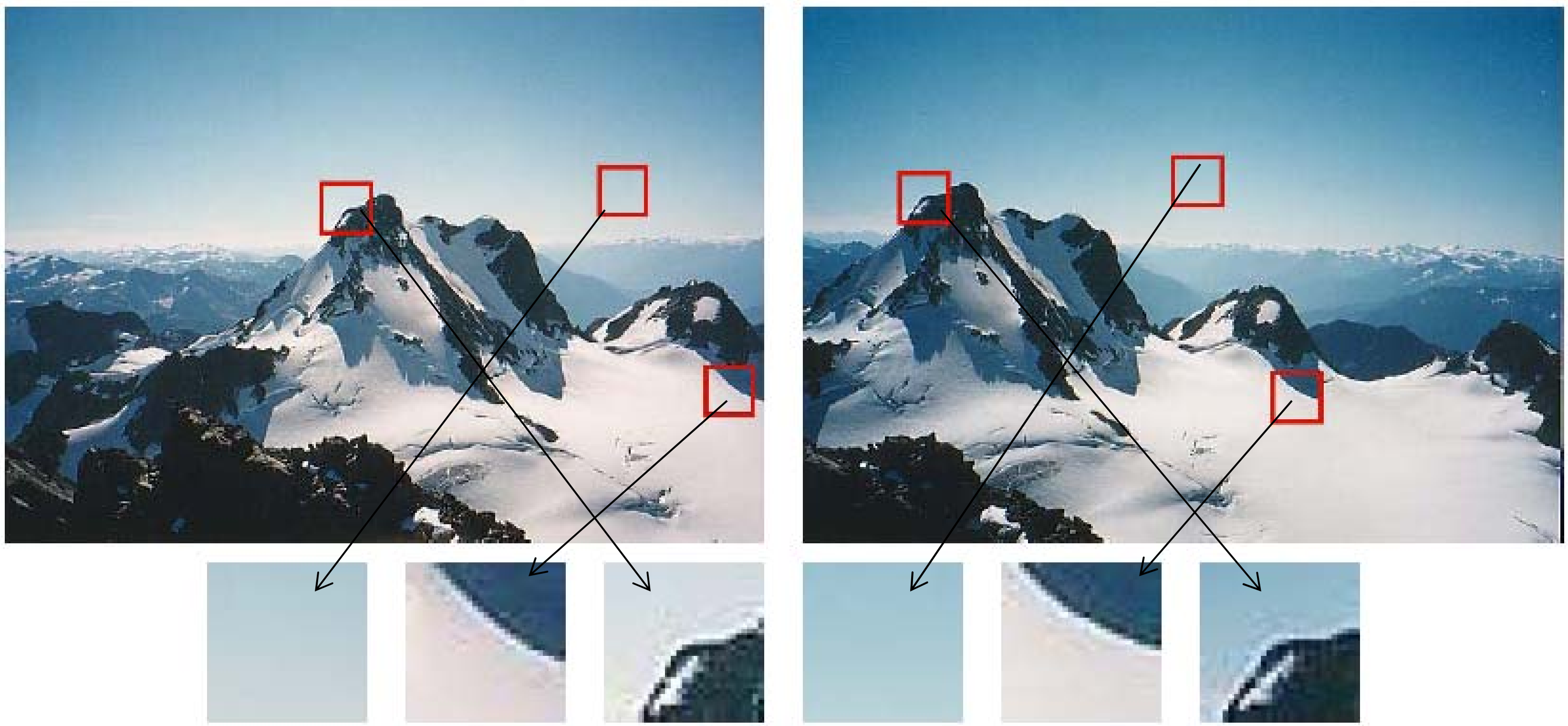
- Must provide some invariance to geometric and photometric differences between the two views.

Kristen Grauman

Some patches can be localized or matched with higher accuracy than others.



Some patches can be localized or matched with higher accuracy than others.



Harris Interest Point Detector

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.

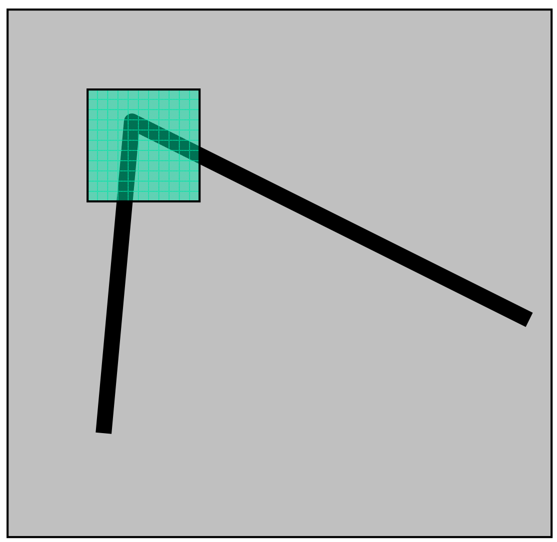
Now, the unconstrained imagery we shall be considering will contain both curved edges and texture of various scales. Representing edges as a set of straight line fragments[4], and using these as our discrete features will be inappropriate, since curved lines and texture edges can be expected to fragment 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.

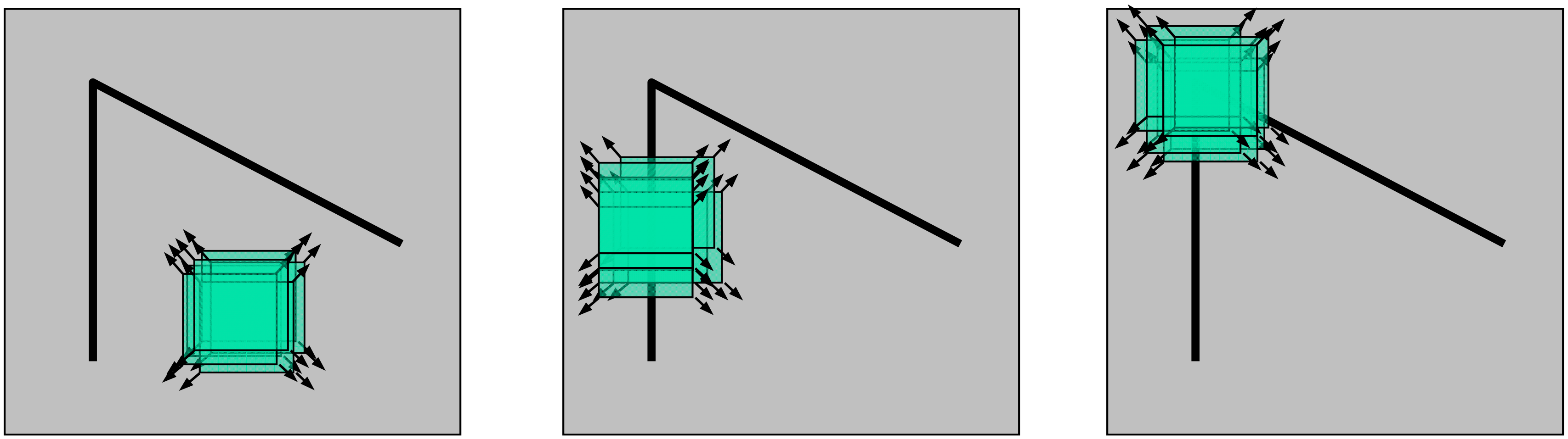
Harris Corner Detector

- Corner point can be recognized in a window
- Shifting a window in any direction should give a large change in intensity



C.Harris, M.Stephens. “A Combined Corner and Edge Detector”. 1988

Basic Idea

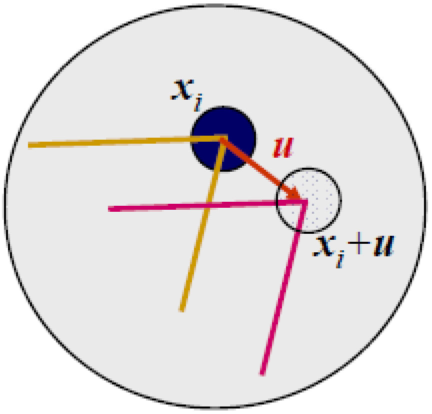


“flat” region:
no change in
all directions

“edge”:
no change along
the edge direction

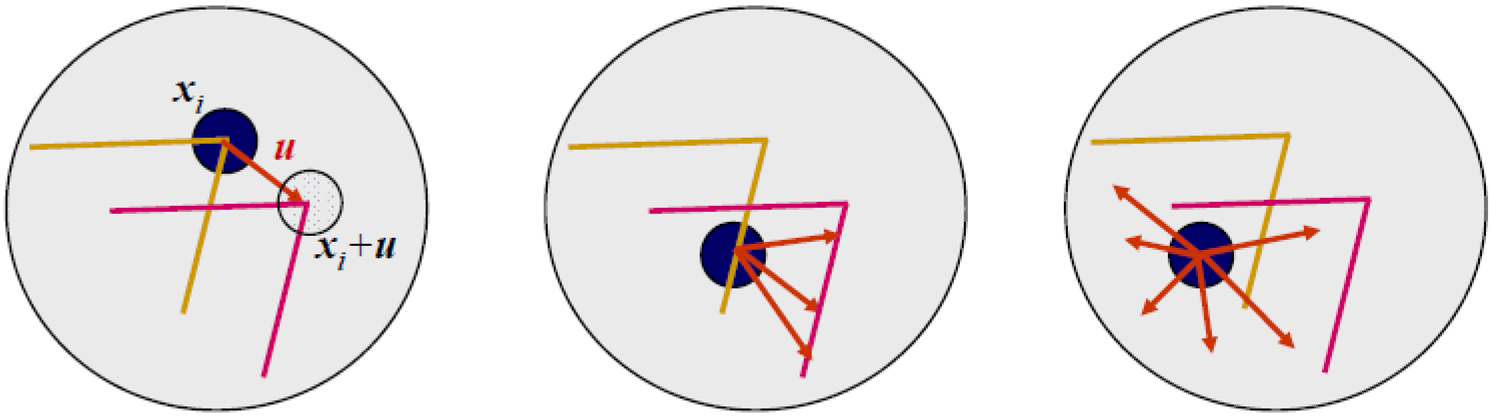
“corner”:
significant change
in all directions

Aperture Problem



(a)

Aperture Problem



(a) (b) (c)

Correlation

$$f \otimes h = \sum_k \sum_l f(k,l)h(k,l)$$

$$f = \text{Image}$$

$$h = \text{Kernel}$$

f

f_1	f_2	f_3
f_4	f_5	f_6
f_7	f_8	f_9

$\otimes$

h

h_1	h_2	h_3
h_4	h_5	h_6
h_7	h_8	h_9

$$\begin{aligned} f * h = {} & f_1h_1 + f_2h_2 + f_3h_3 \\ & + f_4h_4 + f_5h_5 + f_6h_6 \\ & + f_7h_7 + f_8h_8 + f_9h_9 \end{aligned}$$

Correlation

$$f \otimes h = \sum_{k} \sum_{l} f(k,l) h(k,l)$$

Cross correlation

$$f \otimes f = \sum_{k} \sum_{l} f(k,l) f(k,l)$$

Auto correlation

Correlation Vs SSD

$$SSD = \sum_k \sum_l \left(f(k,l) - h(k,l) \right)^2$$

Sum of Squares Difference

minimize

$$SSD = \sum_k \sum_l \left(\cancel{f(k,l)^2} - 2h(k,l) f(k,l) + \cancel{h(k,l)^2} \right)$$

minimize

$$SSD = \sum_k \sum_l \left(-2h(k,l) f(k,l) \right)$$

These terms do not depend on correlation

$$SSD = \sum_k \sum_l \left(\cancel{2} h(k,l) f(k,l) \right)$$

maximize

$$Correlation = \sum_k \sum_l \left(h(k,l) f(k,l) \right)$$

maximize

$$f \otimes f = \sum_k \sum_l f(k,l) f(k,l)$$

Mathematics of Harris Detector

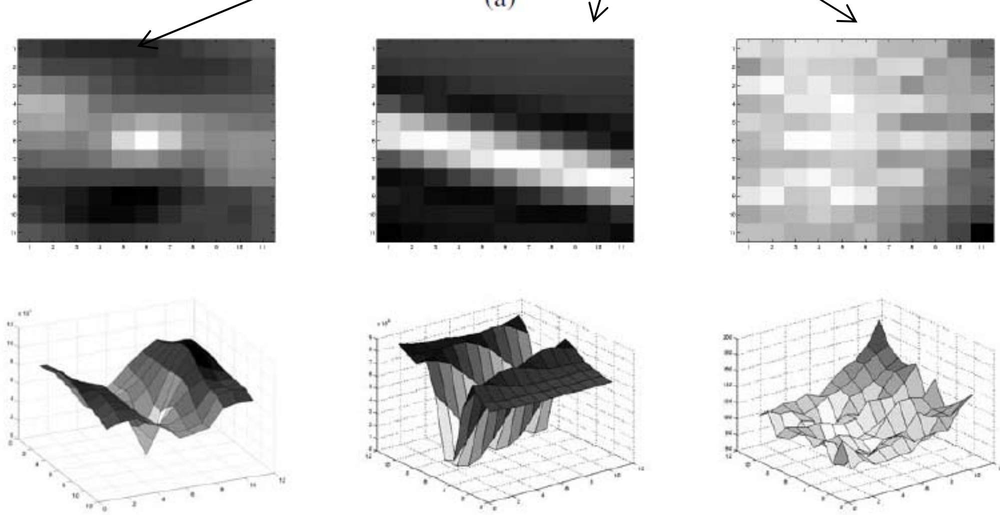
- Change of intensity for the shift (u,v)

$$E(u,v) = \sum_{x,y} [\underbrace{I(x+u, y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

Auto-correlation

Auto-Correlation

(a)



Brook Taylor (1685-1731)

His marriage in 1721 with Miss Brydges of Wallington, Surrey, led to an estrangement from his father, which ended in 1723 after her death in giving birth to a son, who also died.

1725 he married—this time with his father's approval—Sabetta Sawbridge of Olantigh, Kent, who also died in childbirth in 1730 ; in this case, however, his daughter, Elizabeth, survived.

Taylor was elected a fellow of the Royal Society early in 1712, and in the same year sat on the committee for adjudicating the claims of Sir Isaac Newton and Gottfried Leibniz about Calculus.

Taylor Series

$f(x)$ Can be represented at point a in terms of its derivatives

$$f(x) = f(a) + (x-a) f_x + \frac{(x-a)^2}{2!} f_{xx} + \frac{(x-a)^3}{3!} f_{xxx} + \dots$$

Taylor Series

Express $I(x+u, y+v)$ at (x, y):

$$I(x+u, y+v) = I(x, y) + I_x(x+u-x) + I_y(y+v-y)$$

$$I(x+u, y+v) = I(x, y) + I_x u + I_y v$$

Taylor Series

$$f(x,y,t) = f(x+dx, y+dy, t+dt)$$

Taylor Series of right side

$$\cancel{f(x,y,t)} = \cancel{f(x,y,t)} + \frac{\partial f}{\partial x}(x+dx-x) + \frac{\partial f}{\partial y}(y+dy-y) + \frac{\partial f}{\partial t}(t+dt-t)$$

$$0 = f_x dx + f_y dy + f_t dt$$

$$0 = f_x \frac{dx}{dt} + f_y \frac{dy}{dt} + f_t \frac{dt}{dt}$$

$$0 = f_x u + f_y v + f_t \qquad f_x u + f_y v + f_t = 0$$

Optical Flow Constrain Equation

Mathematics of Harris Detector

$$E(u,v)=\sum_{x,y}[\underbrace{I(x+u,y+v)}_{\text{shifted intensity}}-\underbrace{I(x,y)}_{\text{intensity}}]^2$$

$$E(u,v)=\sum_{x,y}[\underbrace{\cancel{I(x,y)}+uI_x+vI_y}_{\text{shifted intensity}}-\underbrace{\cancel{I(x,y)}}_{\text{intensity}}]^2$$

Taylor Series

$$E(u,v)=\sum_{x,y}[uI_x+vI_y]^2$$

$$E(u,v)=\sum_{x,y}\left[\begin{pmatrix}u & v\end{pmatrix}\begin{pmatrix}I_x\\ I_y\end{pmatrix}\right]^2$$

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

$$M=\sum_{x,y}\begin{bmatrix}I_xI_x & I_xI_y\\ I_xI_y & I_yI_y\end{bmatrix}$$

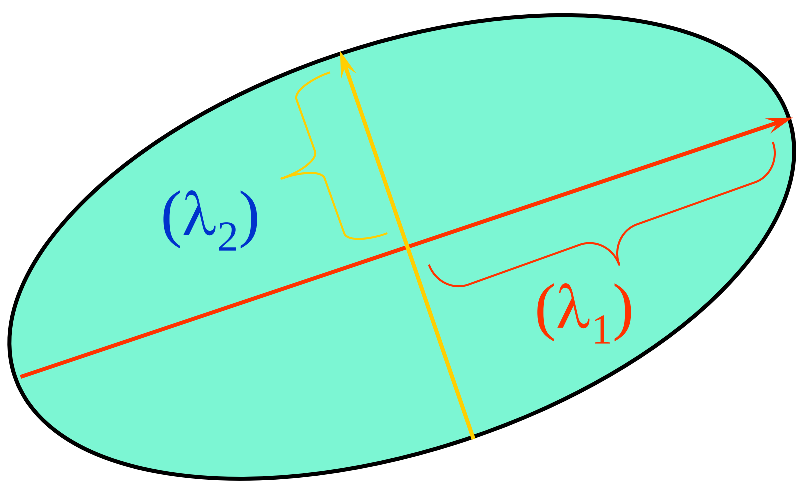
$$E(u,v)=\begin{pmatrix}u & v\end{pmatrix}\left[\sum_{x,y}\begin{pmatrix}I_x\\ I_y\end{pmatrix}\begin{pmatrix}I_x & I_y\end{pmatrix}\right]\begin{pmatrix}u\\ v\end{pmatrix}$$

$$E(u,v)=\begin{pmatrix}u & v\end{pmatrix}M\begin{pmatrix}u\\ v\end{pmatrix}$$

Mathematics of Harris Detector

$$E(u,v) = (u \quad v)\, M \begin{pmatrix} u \\ v \end{pmatrix} \qquad M = \sum_{x,y} \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix}$$

- E(u,v) is an equation of an ellipse.
- Let λ_1 and λ_2 be eigenvalues of M



Eigen Vectors and Eigen Values

The eigen vector, *x*, of a matrix *A* is a special vector, with the following property

$$Ax = \lambda x$$

Where λ is called eigen value

To find eigen values of a matrix *A* first find the roots of:

$$\det(A - \lambda I) = 0$$

Then solve the following linear system for each eigen value to find corresponding eigen vector

$$(A - \lambda I)x = 0$$

Example

$$A = \begin{bmatrix} -1 & 2 & 0 \\ 0 & 3 & 4 \\ 0 & 0 & 7 \end{bmatrix}$$

Eigen Values

$$\lambda_1 = 7,\ \lambda_2 = 3, \lambda_3 = -1$$

$$\mathbf{x_1} = \begin{bmatrix} 1 \\ 4 \\ 4 \end{bmatrix},\ \mathbf{x_2} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix},\ \ \mathbf{x_3} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Eigen Vectors

Eigen Values

$$\det(A - \lambda I) = 0$$

$$\det\left(\begin{bmatrix} -1 & 2 & 0 \\ 0 & 3 & 4 \\ 0 & 0 & 7 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) = 0$$

$$\det\left(\begin{bmatrix} -1-\lambda & 2 & 0 \\ 0 & 3-\lambda & 4 \\ 0 & 0 & 7-\lambda \end{bmatrix}\right) = 0$$

$$(-1-\lambda)((3-\lambda)(7-\lambda)-0) = 0$$

$$(-1-\lambda)(3-\lambda)(7-\lambda) = 0$$

$$\lambda = -1, \quad \lambda = 3, \quad \lambda = 7$$

Eigen Vectors

$$\lambda = -1 \qquad (A - \lambda I)x = 0$$

$$\left(\begin{bmatrix} -1 & 2 & 0 \\ 0 & 3 & 4 \\ 0 & 0 & 7 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right)\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 2 & 0 \\ 0 & 4 & 4 \\ 0 & 0 & 8 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$0 + 2x_2 + 0 \ = 0$$

$$0 + 4x_2 + 4x_3 = 0$$

$$0 + 0 + 8x_3 = 0$$

$$x_1 = 1, \quad x_2 = 0, \quad x_3 = 0$$

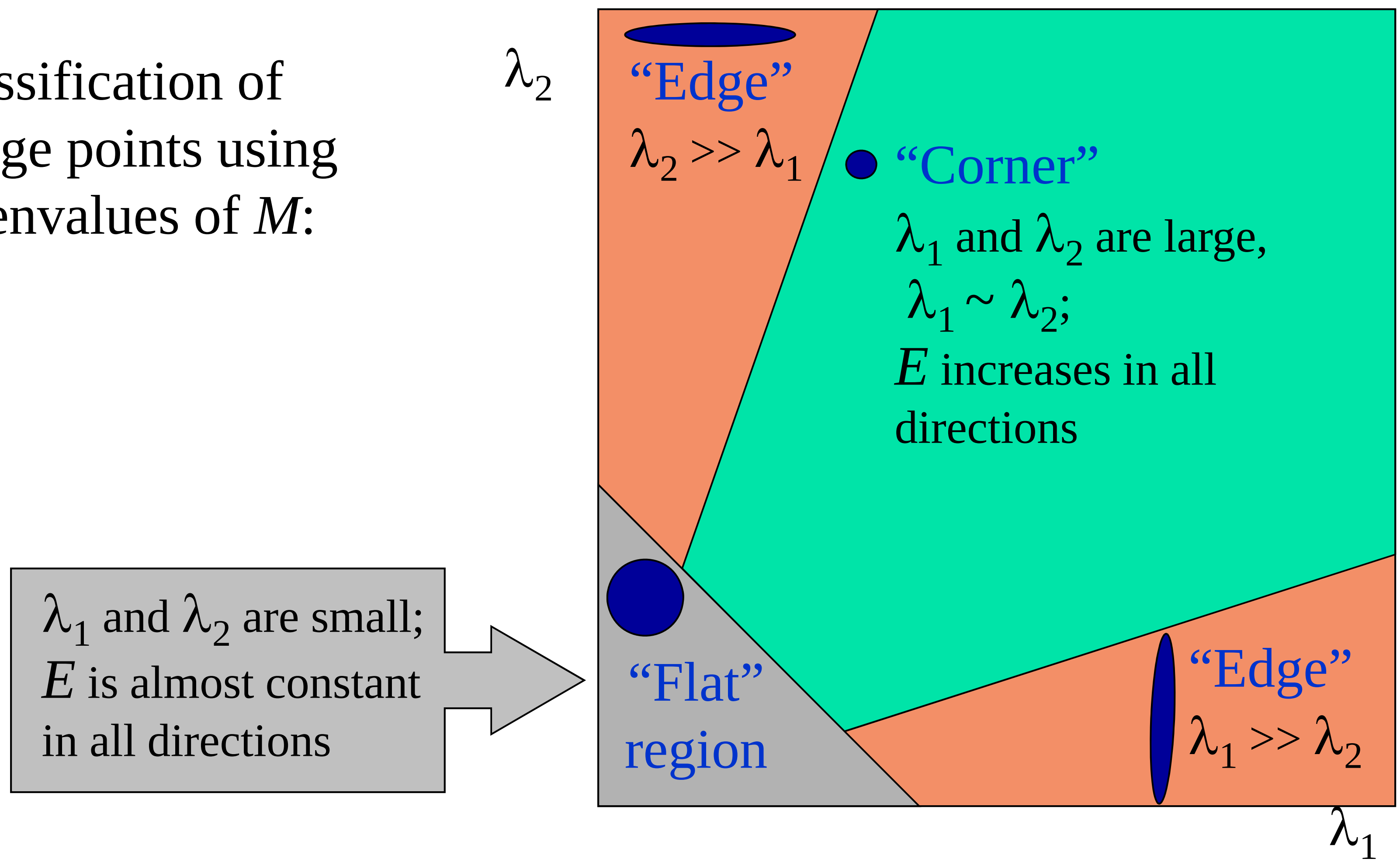
$$\mathbf{x_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

MATLAB Fuction

```
[vectorC,valueC]=eig(C);
```

Mathematics of Harris Detector

Classification of image points using eigenvalues of M:



Mathematics of Harris Detector

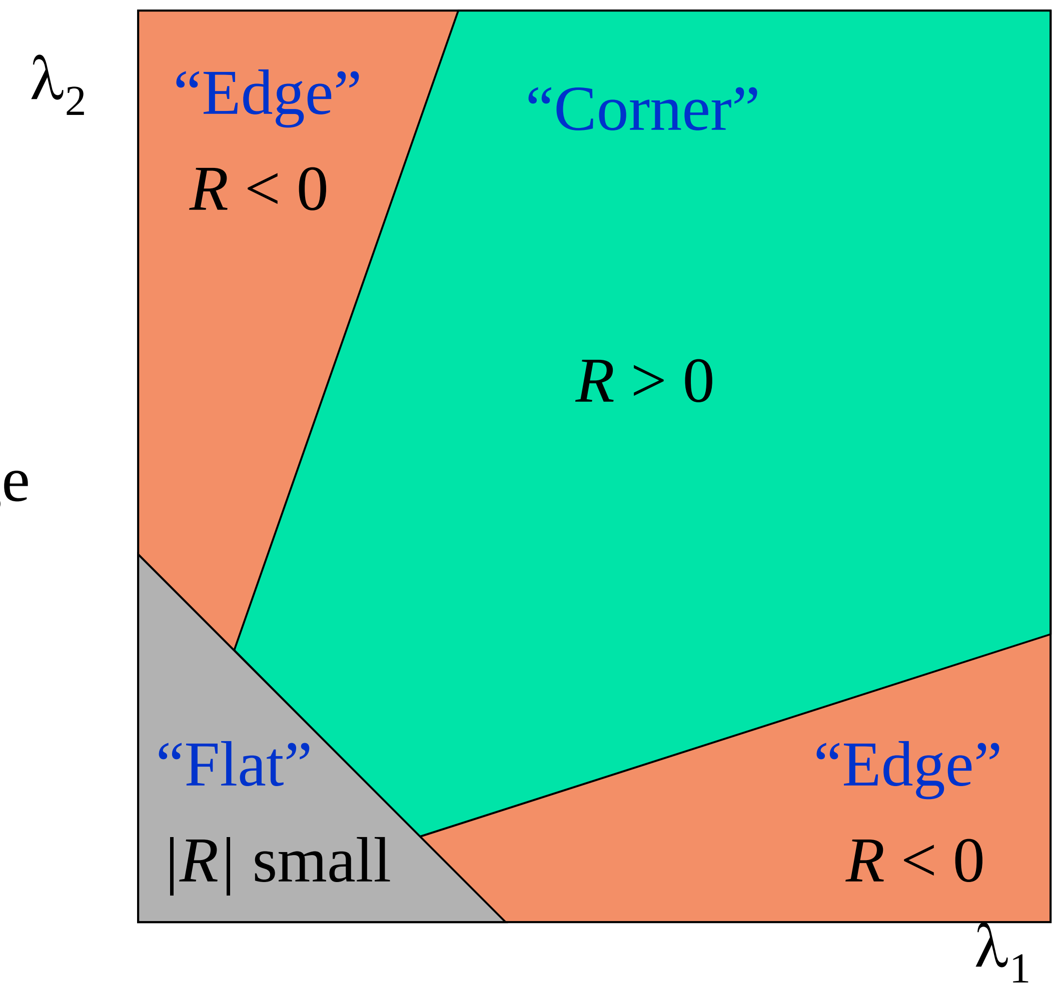
- Measure of cornerness in terms of λ_1, λ_2

$$M = SDS^{-1} \qquad D = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

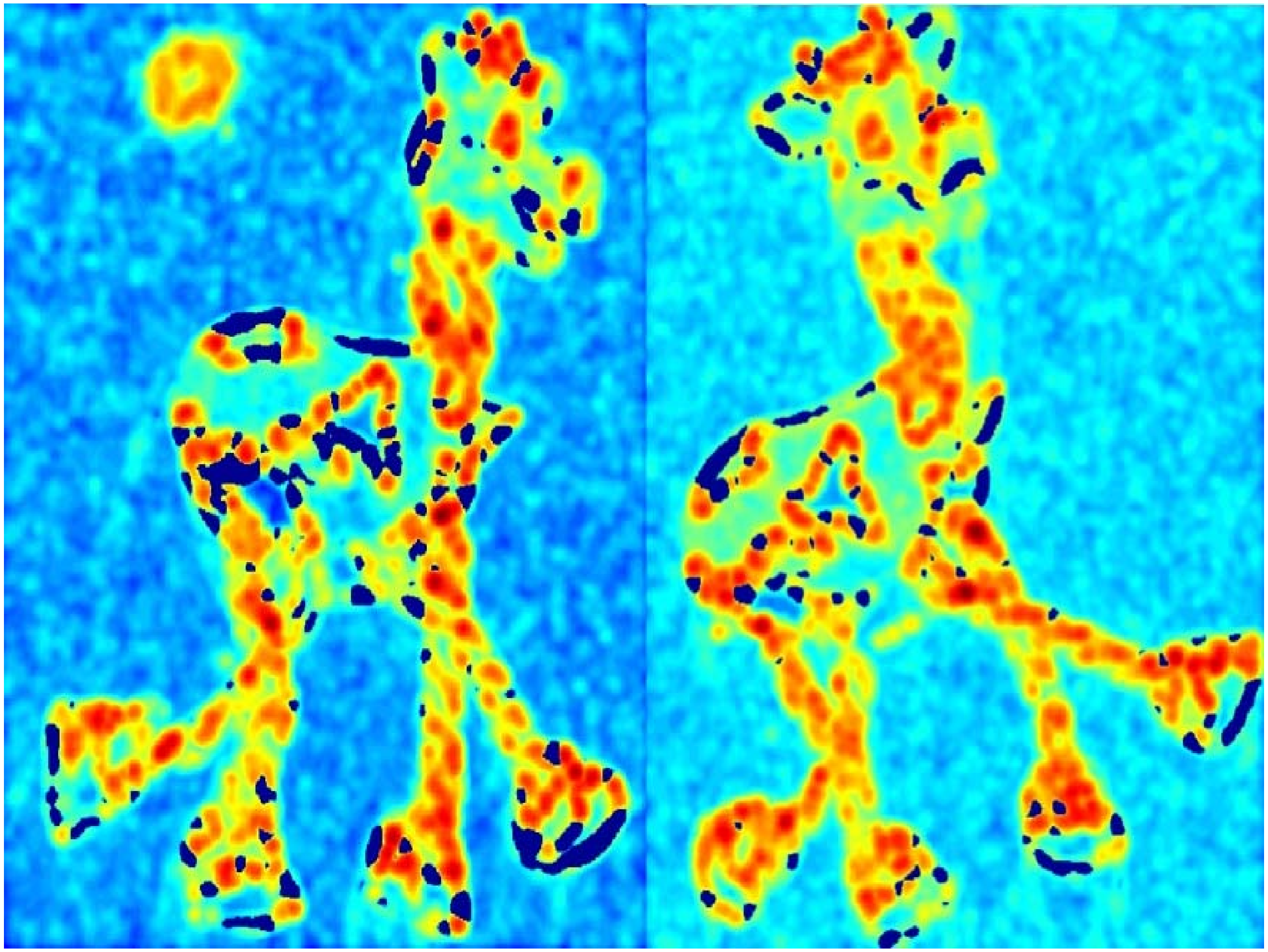
$$R = \det D - k(trace\ D)^2 \qquad R = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2$$

Mathematics of Harris Detector

- *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

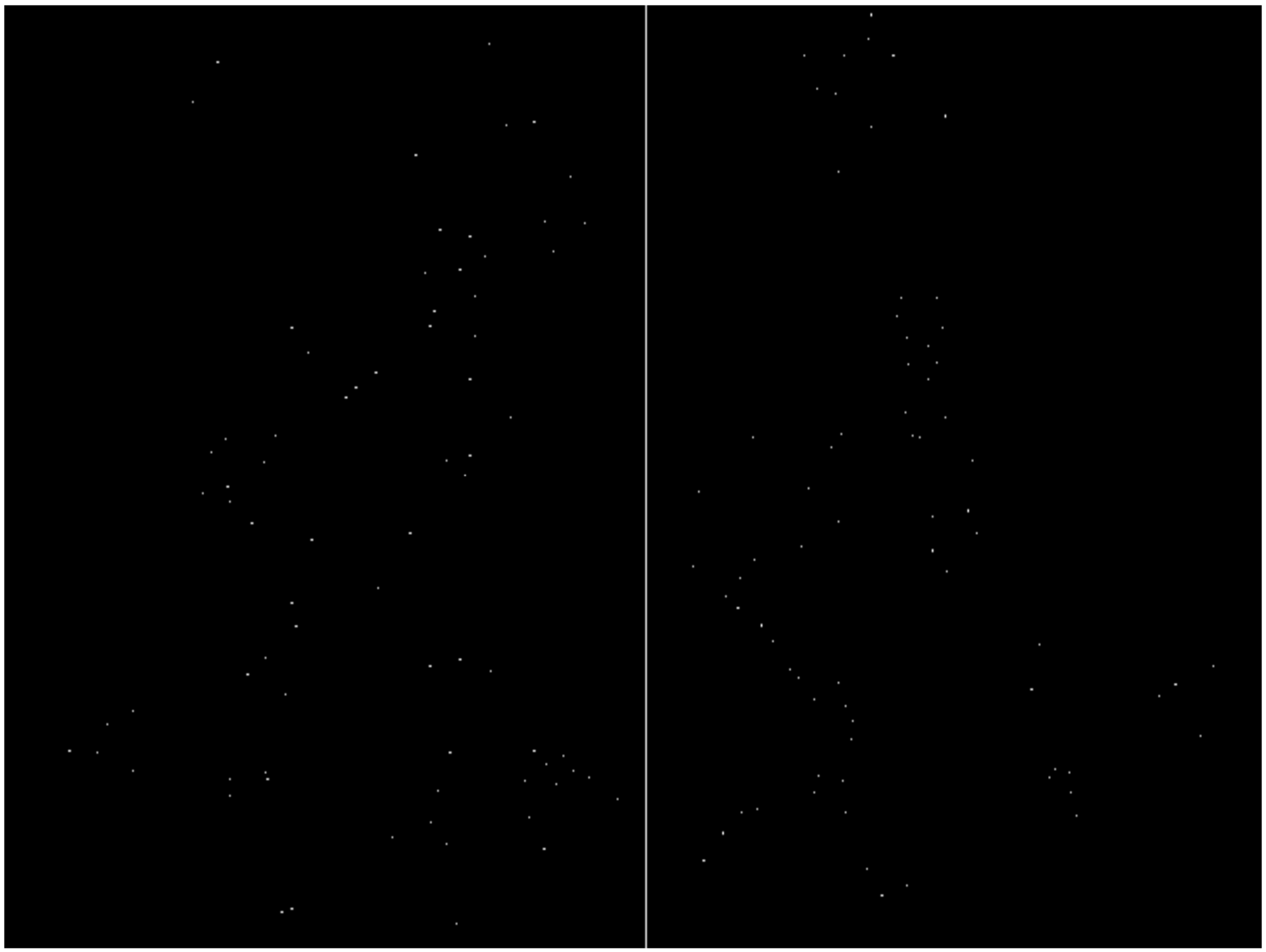


Compute corner response



Find points with large corner response: $R >$ threshold

Take only the points of local maxima of R



If pixel value is greater than its neighbors then it is a local maxima.

Other Version of Harris Detectors

$$R = \lambda_1 - \alpha\lambda_2$$

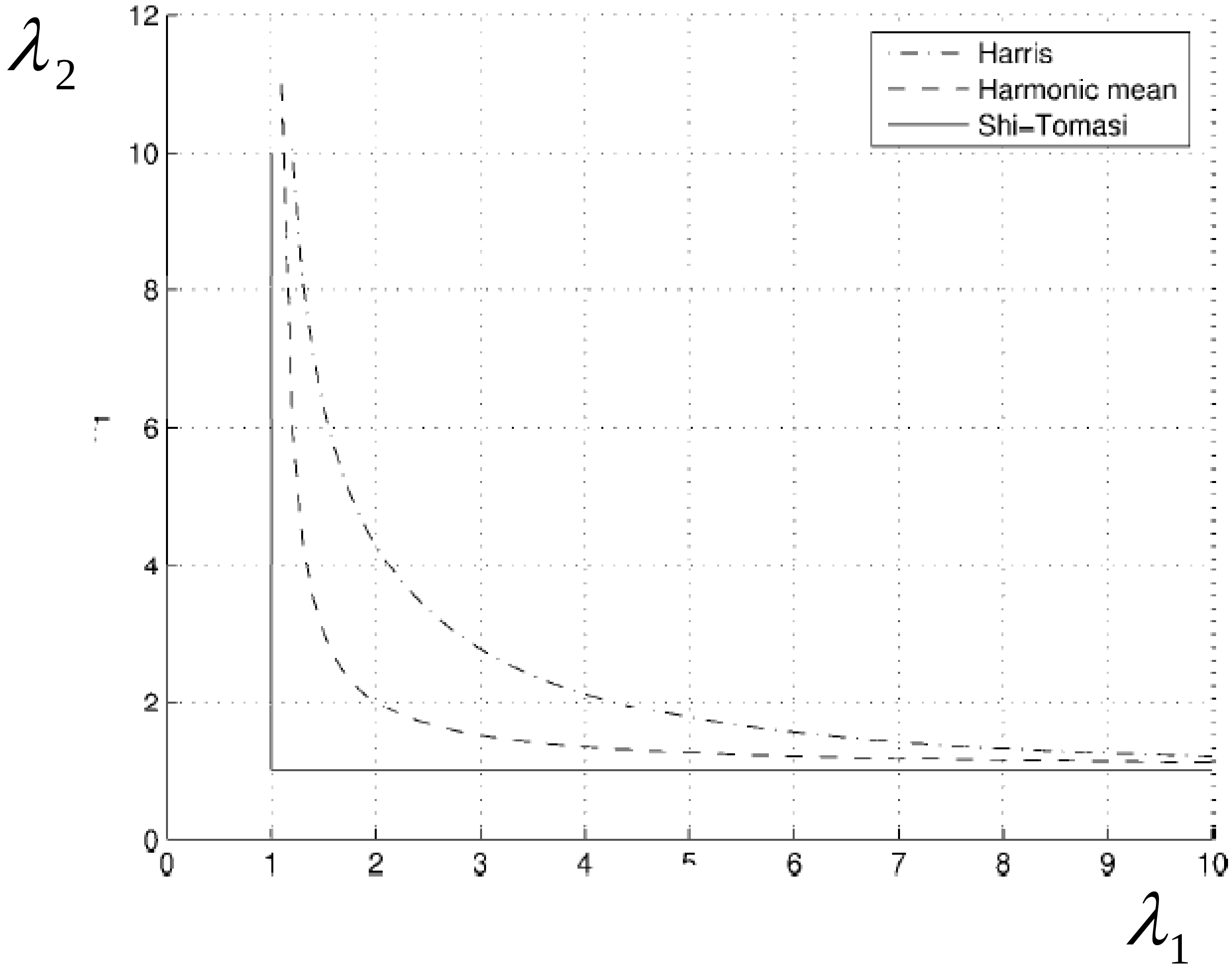
Triggs

$$R = \frac{\det(D)}{trace(D)} = \frac{\lambda_1\lambda_2}{\lambda_1 + \lambda_2}$$

Szeliski (Harmonic mean)

$$R = \lambda_1$$

Shi-Tomasi



λ_2
Harris
Harmonic mean
Shi–Tomasi
λ_1

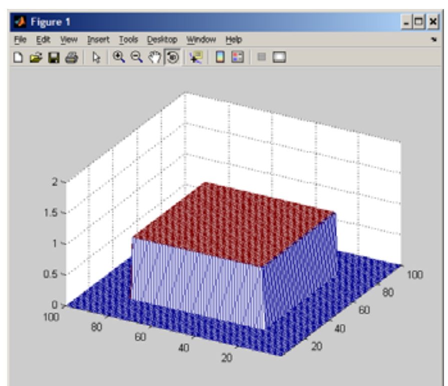
Mathematics of Harris Detector

- Change of intensity for the shift (u,v)

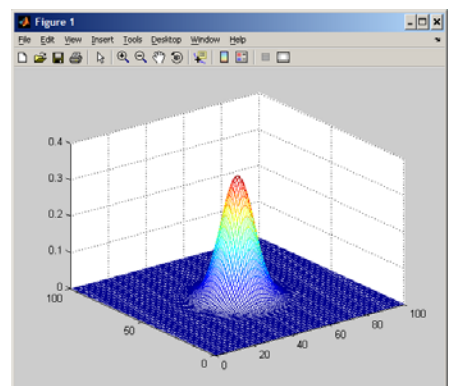
$$E(u,v) = \sum_{x,y} [\underbrace{I(x+u, y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

Auto-correlation

Window functions →



UNIFORM



GAUSSIAN

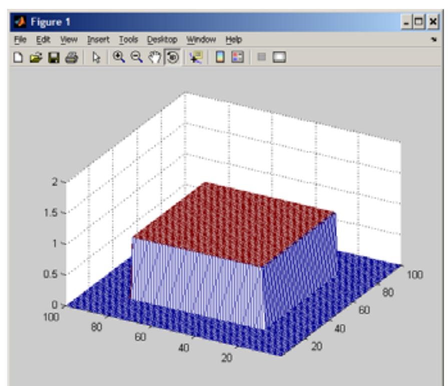
Mathematics of Harris Detector

- Change of intensity for the shift (u,v)

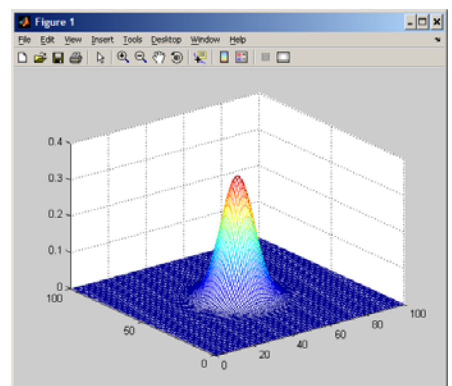
$$E(u,v) = \sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \, [\underbrace{I(x+u,y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

Auto-correlation

Window functions →



UNIFORM



GAUSSIAN

Mathematics of Harris Detector

$$E(u,v)=\sum_{x,y}\underbrace{w(x,y)}_{\text{window function}}\,[\underbrace{I(x+u,y+v)}_{\text{shifted intensity}}-\underbrace{I(x,y)}_{\text{intensity}}]^2$$

$$E(u,v)=\sum_{x,y}\underbrace{w(x,y)}_{\text{window function}}[\underbrace{I(x,y)+uI_x+vI_y}_{\text{shifted intensity}}-\underbrace{I(x,y)}_{\text{intensity}}]^2$$

Taylor Series

$$E(u,v)=\sum_{x,y}w(x,y)[uI_x+vI_y]^2$$

$$E(u,v)=\sum_{x,y}w(x,y)\left[\begin{pmatrix}u & v\end{pmatrix}\begin{pmatrix}I_x\\ I_y\end{pmatrix}\right]^2$$

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

$$M=\sum_{x,y}\begin{bmatrix}I_xI_x & I_xI_y\\ I_xI_y & I_yI_y\end{bmatrix}$$

$$E(u,v)=\begin{pmatrix}u & v\end{pmatrix}\left[\sum_{x,y}w(x,y)\begin{pmatrix}I_x\\ I_y\end{pmatrix}\begin{pmatrix}I_x & I_y\end{pmatrix}\right]\begin{pmatrix}u\\ v\end{pmatrix}$$

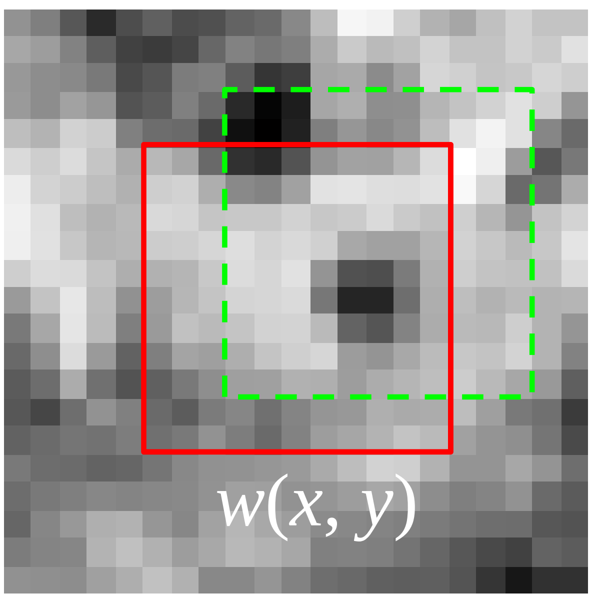
$$E(u,v)=\begin{pmatrix}u & v\end{pmatrix}M\begin{pmatrix}u\\ v\end{pmatrix}$$

Corner Detection: Mathematics

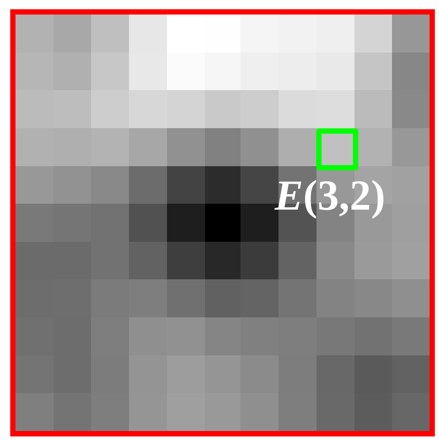
Change in appearance of window $w(x,y)$
for the shift $[u,v]$:

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

$I(x, y)$



$E(u, v)$



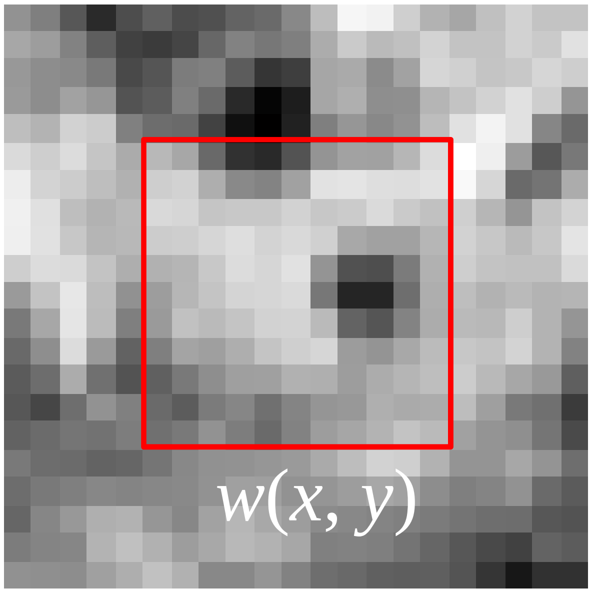
Slide Credit: James Hays

Corner Detection: Mathematics

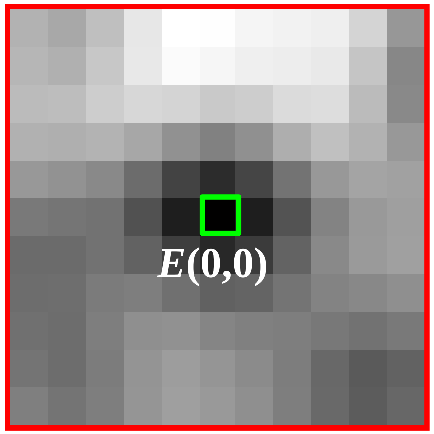
Change in appearance of window $w(x,y)$
for the shift $[u,v]$:

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

$I(x, y)$



$E(u, v)$



Slide Credit: James Hays

Corner Detection: Mathematics

Change in appearance of window $w(x,y)$
for the shift $[u,v]$:

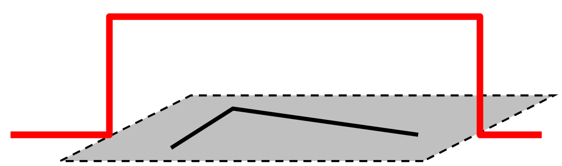
$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

Window function

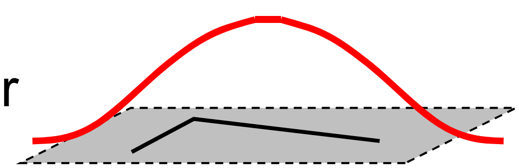
Shifted intensity

Intensity

Window function $w(x,y) =$



or



1 in window, 0 outside

Gaussian

Source: R. Szeliski

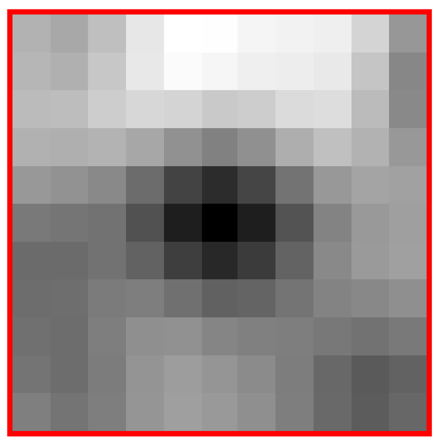
Corner Detection: Mathematics

Change in appearance of window $w(x,y)$
for the shift $[u,v]$:

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

We want to find out how this function behaves for small shifts

$E(u, v)$



Slide Credit: James Hays

Corner Detection: Mathematics

The quadratic approximation simplifies to

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

where *M* is a *second moment matrix* computed from image derivatives:

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

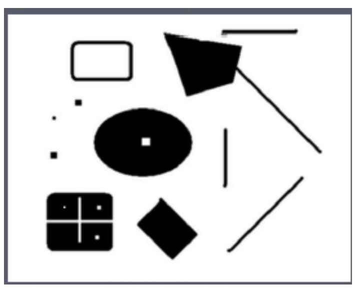
$$M = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \; I_y] = \sum \nabla I (\nabla I)^T$$

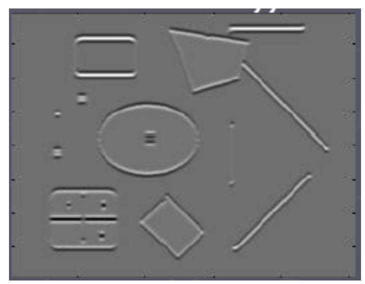
Slide Credit: James Hays

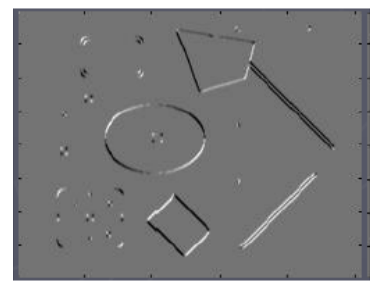
Corners as distinctive interest points

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

2 x 2 matrix of image derivatives (averaged in neighborhood of a point).







Notation:

$$I_x \Leftrightarrow \frac{\partial I}{\partial x} \qquad I_y \Leftrightarrow \frac{\partial I}{\partial y} \qquad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$$

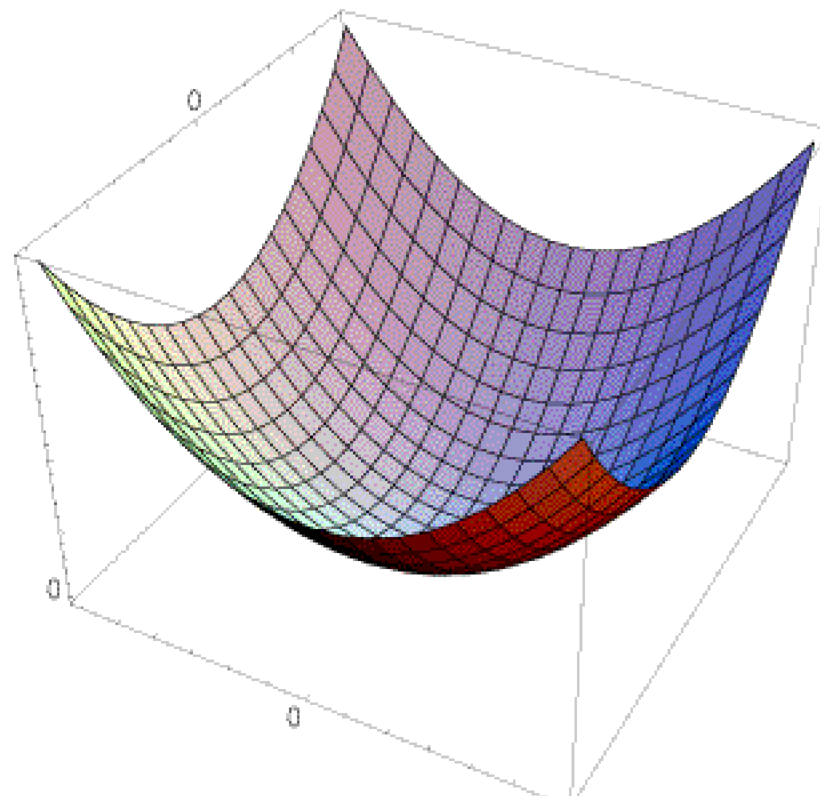
Slide Credit: James Hays

Interpreting the second moment matrix

The surface *E*(*u*,*v*) is locally approximated by a quadratic form. Let's try to understand its shape.

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

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



Slide Credit: James Hays

Interpreting the second moment matrix

First, consider the axis-aligned case (gradients are either horizontal or vertical)

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

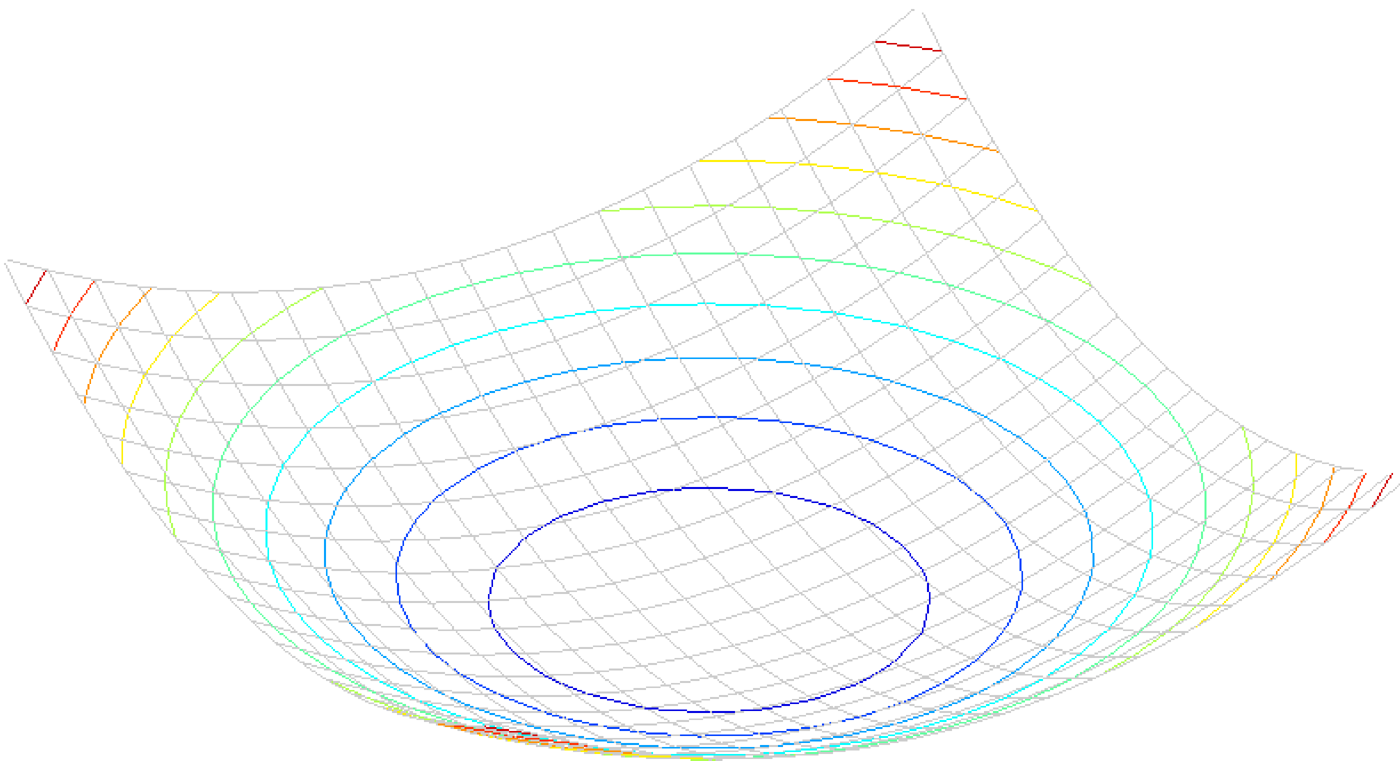
If either λ is close to 0, then this is **not** a corner, so look for locations where both are large.

Slide Credit: James Hays

Interpreting the second moment matrix

Consider a horizontal "slice" of $E(u, v)$: $\quad [u \;\; v] \; M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$

This is the equation of an ellipse.



Slide Credit: James Hays

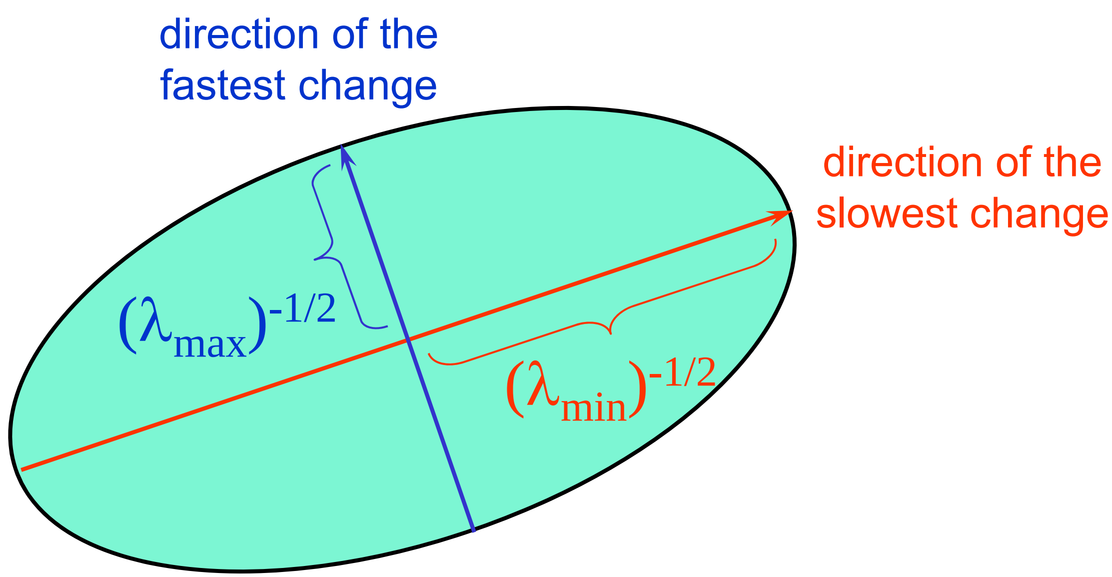
Interpreting the second moment matrix

Consider a horizontal "slice" of $E(u, v)$: $[u \ \ v] \ M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$

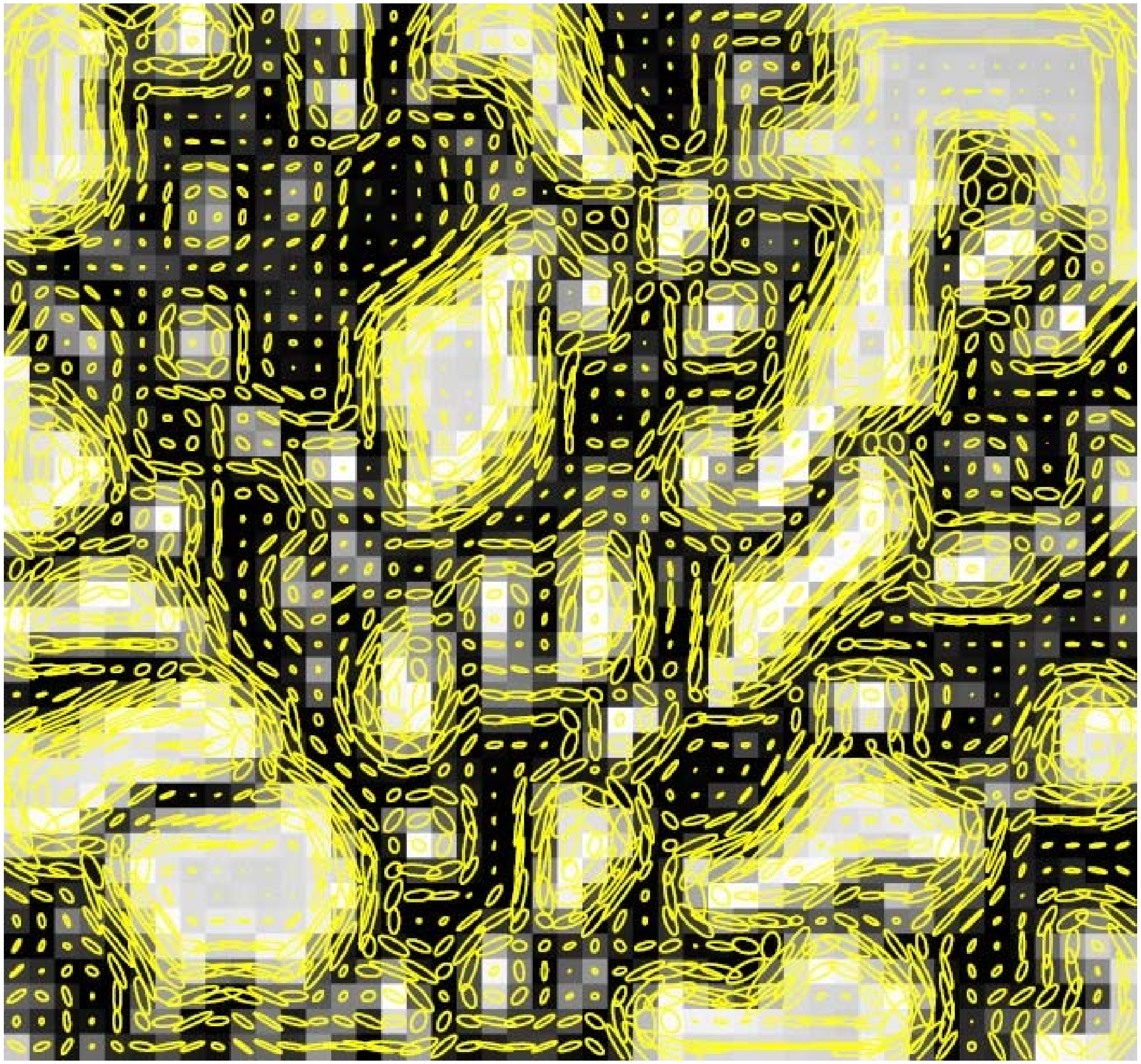
This is the equation of an ellipse.

Diagonalization of M: $M = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$

The axis lengths of the ellipse are determined by the eigenvalues and the orientation is determined by R



Visualization of second moment matrices



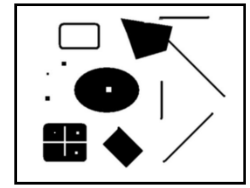
Slide Credit: James Hays

Algorithm

- Compute horizontal and vertical derivatives of image I_x and I_y.
- Compute three images corresponding to three terms in matrix *M*.
- Convolve these three images with a larger Gaussian (window).
- Compute scalar cornerness value using one of the *R* measures.
- Find local maxima above some threshold as detected interest points.

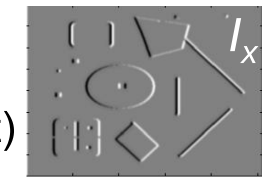
Harris Detector [Harris88]

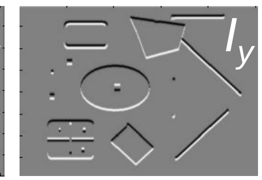
- Second moment matrix



$$\mu(\sigma_I,\sigma_D) = g(\sigma_I) * \begin{bmatrix} I_x^2(\sigma_D) & I_x I_y(\sigma_D) \\ I_x I_y(\sigma_D) & I_y^2(\sigma_D) \end{bmatrix}$$

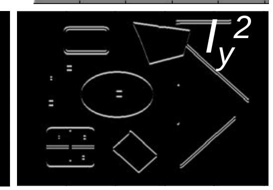
1. Image derivatives (optionally, blur first)

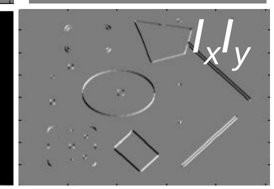




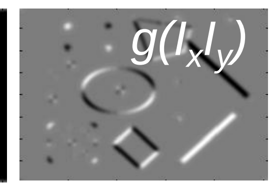
$$\det M = \lambda_1 \lambda_2$$
$$\operatorname{trace} M = \lambda_1 + \lambda_2$$

2. Square of derivatives





3. Gaussian filter $g(\sigma_I)$



4. Cornerness function – both eigenvalues are strong

$$har = \det[\mu(\sigma_I,\sigma_D)] - \alpha[\operatorname{trace}(\mu(\sigma_I,\sigma_D))^2] =$$
$$g(I_x^2)g(I_y^2) - [g(I_xI_y)]^2 - \alpha[g(I_x^2) + g(I_y^2)]^2$$

5. Non-maxima suppression

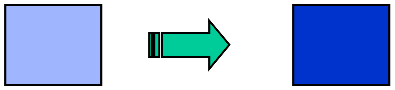
Slide Credit: James Hays

Invariance and covariance

- We want corner locations to be *invariant* to photometric transformations and *covariant* to geometric transformations
  - **Invariance:** image is transformed and corner locations do not change
  - **Covariance:** if we have two transformed versions of the same image, features should be detected in corresponding locations

Slide Credit: James Hays

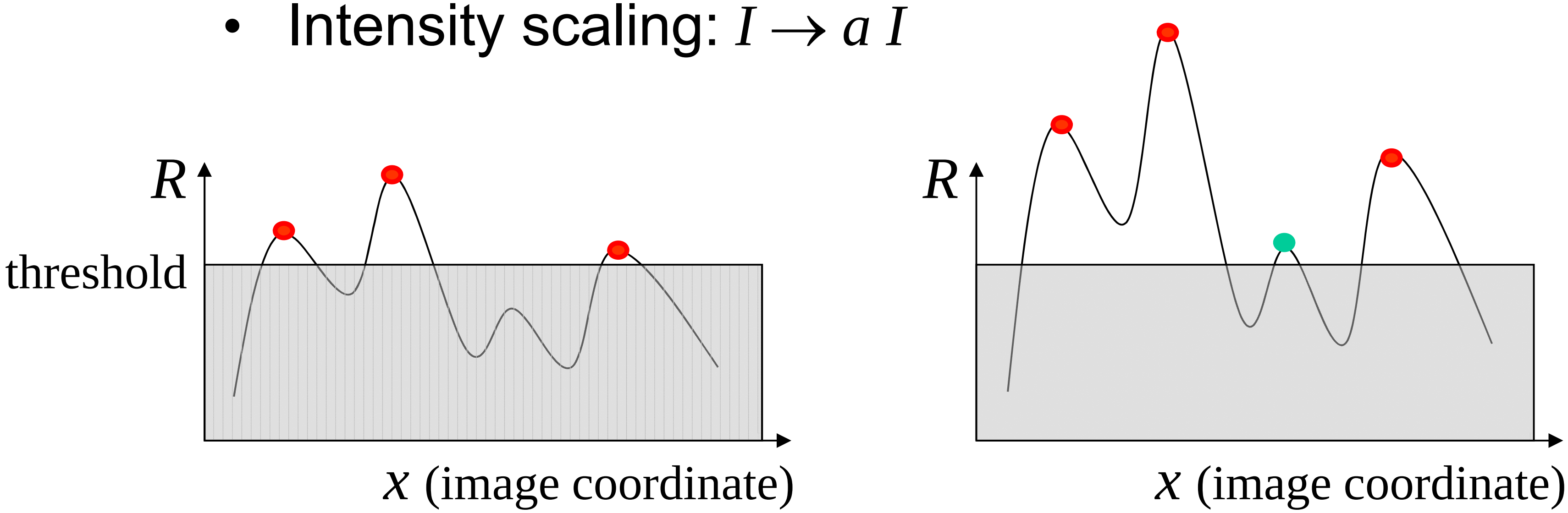
Affine intensity change



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

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

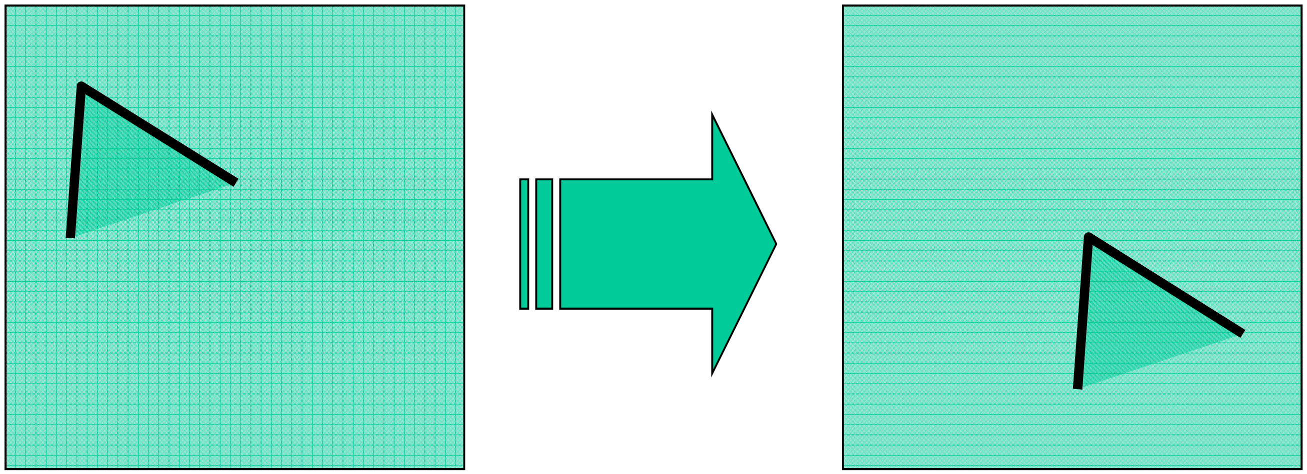
- Intensity scaling: $I \rightarrow a\, I$



Partially invariant to affine intensity change

Slide Credit: James Hays

Image translation

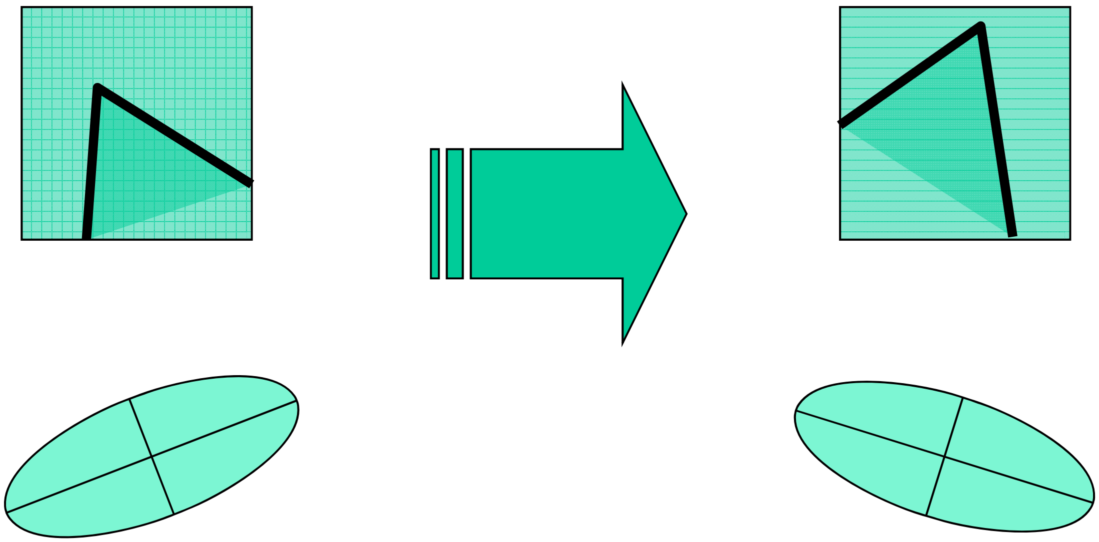


- Derivatives and window function are shift-invariant

Corner location is covariant w.r.t. translation

Slide Credit: James Hays

Image rotation

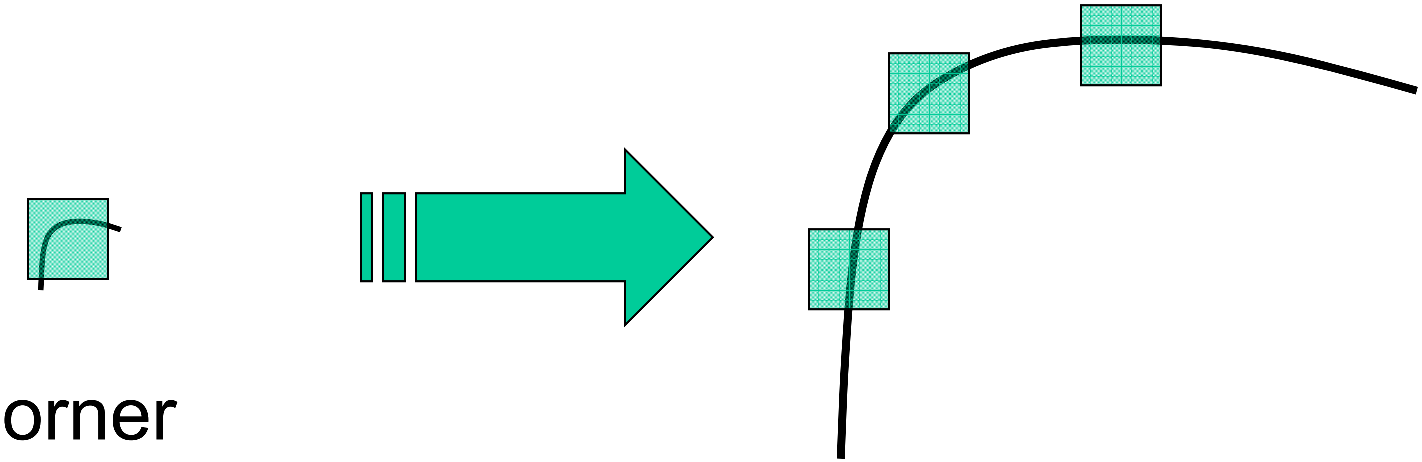


Second moment ellipse rotates but its shape (i.e. eigenvalues) remains the same

Corner location is covariant w.r.t. rotation

Slide Credit: James Hays

Scaling



Corner

All points will be classified as edges

Corner location is not covariant to scaling!

Slide Credit: James Hays

Algorithm

- Compute horizontal and vertical derivatives of image I_x and I_y.
- Compute three images corresponding to three terms in matrix *M*.
- Convolve these three images with a larger Gaussian (window).
- Compute scalar cornerness value using one of the *R* measures.
- Find local maxima above some threshold as detected interest points.

Reading Material

- Section 4.1.1 Feature Detectors
  - Richard Szeliski, "Computer Vision: Algorithms and Application“, Springer.