Local feature: Corners

Lu Peng
School of Computer Science,
Beijing University of Posts and Telecommunications

0

Machine Vision Technology							
Semantic information				**Metric 3D information**			
Pixels	Segments	Images	Videos	Camera		Multi-view Geometry	
Convolutions Edges & Fitting Local features Texture	Segmentation Clustering	Recognition Detection	Motion Tracking	Camera Model	Camera Calibration	Epipolar Geometry	SFM
10	4	4	2	2	2	2	2

1

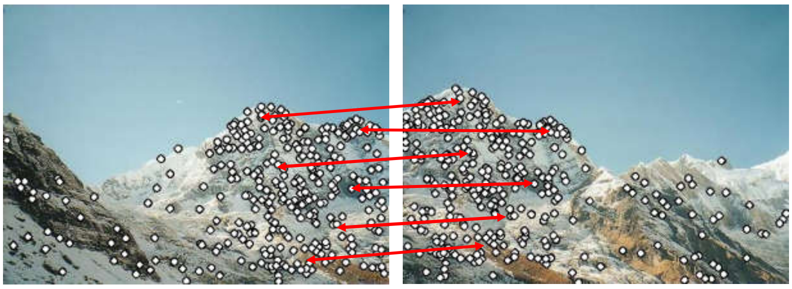
Why extract features?

- Motivation: panorama stitching
  - We have two images – how do we combine them?

Source： S. Lazebnik

Why extract features?

- Motivation: panorama stitching
  - We have two images – how do we combine them?



Step 1: extract features

Step 2: match features

Source： S. Lazebnik

Why extract features?

- Motivation: panorama stitching
  - We have two images – how do we combine them?

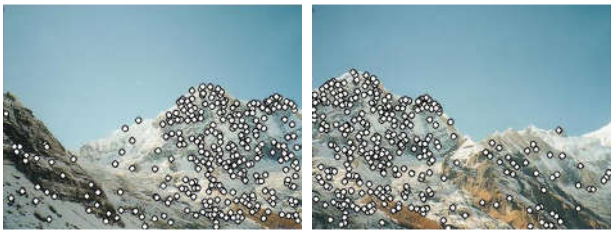
Step 1: extract features

Step 2: match features

Step 3: align images

Source： S. Lazebnik

Characteristics of good features



- Repeatability
  - The same feature can be found in several images despite geometric and photometric transformations
- Saliency
  - Each feature is distinctive
- Compactness and efficiency
  - Many fewer features than image pixels
- Locality
  - A feature occupies a relatively small area of the image; robust to clutter and occlusion

Source： S. Lazebnik

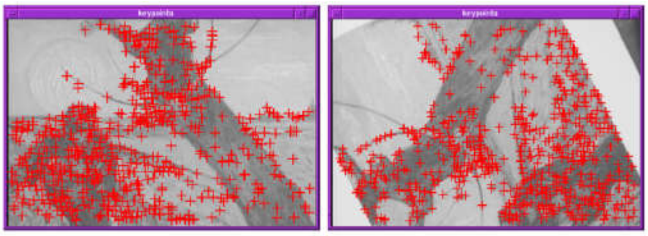
Applications

Feature points are used for:

- Image alignment
- 3D reconstruction
- Motion tracking
- Robot navigation
- Indexing and database retrieval
- Object recognition

Source: S. Lazebnik

Finding Corners



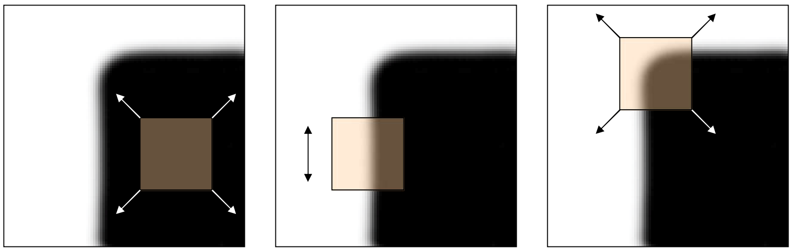
- Key property: in the region around a corner, image gradient has two or more dominant directions
- Corners are repeatable and distinctive

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

Source: S. Lazebnik

Corner Detection: Basic Idea

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



"flat" region:
no change in all directions

"edge":
no change along the edge direction

"corner":
significant change in all directions

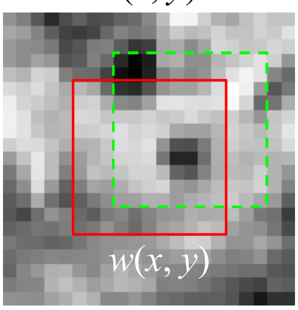
Source: A. Efros

Corner Detection: Mathematics

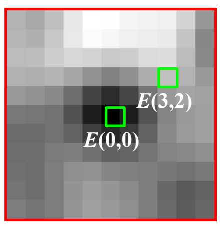
Change in appearance of window $w(x,y)$
for the shift $[u,v]$:

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

$I(x, y)$



$E(u, v)$



Source： S. Lazebnik

Corner Detection: Mathematics

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

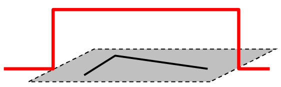
$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u, y+v) - I(x,y)\right]^2$$

Window function

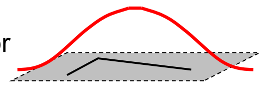
Shifted intensity

Intensity

Window function $w(x,y) =$



or



1 in window, 0 outside

Gaussian

Source: R. Szeliski

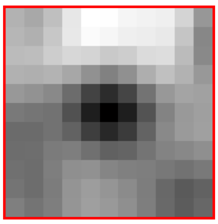
Corner Detection: Mathematics

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

$E(u, v)$



Source: S. Lazebnik

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

Local quadratic approximation of $E(u,v)$ in the neighborhood of (0,0) is given by the *second-order Taylor expansion*:

$$E(u,v) \approx E(0,0) + [u \quad v]\begin{bmatrix} E_u(0,0) \\ E_v(0,0) \end{bmatrix} + \frac{1}{2}[u \quad v]\begin{bmatrix} E_{uu}(0,0) & E_{uv}(0,0) \\ E_{uv}(0,0) & E_{vv}(0,0) \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}$$

Source： S. Lazebnik

Corner Detection: Mathematics

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

Second-order Taylor expansion of $E(u,v)$ about (0,0):

$$E(u,v) \approx E(0,0) + [u \quad v]\begin{bmatrix} E_u(0,0) \\ E_v(0,0) \end{bmatrix} + \frac{1}{2}[u \quad v]\begin{bmatrix} E_{uu}(0,0) & E_{uv}(0,0) \\ E_{uv}(0,0) & E_{vv}(0,0) \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}$$

$$E_u(u,v) = \sum_{x,y} 2w(x,y)\,[I(x+u,y+v) - I(x,y)]I_x(x+u,y+v)$$

$$E_{uu}(u,v) = \sum_{x,y} 2w(x,y)\,I_x(x+u,y+v)I_x(x+u,y+v) + \sum_{x,y} 2w(x,y)\,[I(x+u,y+v) - I(x,y)]I_{xx}(x+u,y+v)$$

$$E_{uv}(u,v) = \sum_{x,y} 2w(x,y)\,I_y(x+u,y+v)I_x(x+u,y+v) + \sum_{x,y} 2w(x,y)\,[I(x+u,y+v) - I(x,y)]I_{xy}(x+u,y+v)$$

Source： S. Lazebnik

Corner Detection: Mathematics

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

Second-order Taylor expansion of *E*(*u*,*v*) about (0,0):

$$E(u,v)\approx E(0,0)+[u\ \ v]\begin{bmatrix}E_u(0,0)\\E_v(0,0)\end{bmatrix}+\frac{1}{2}[u\ \ v]\begin{bmatrix}E_{uu}(0,0) & E_{uv}(0,0)\\E_{uv}(0,0) & E_{vv}(0,0)\end{bmatrix}\begin{bmatrix}u\\v\end{bmatrix}$$

$$E(0,0)=0 \qquad E_{uu}(0,0)=\sum_{x,y} 2w(x,y)\,I_x(x,y)I_x(x,y)$$

$$E_u(0,0)=0 \qquad E_{vv}(0,0)=\sum_{x,y} 2w(x,y)\,I_y(x,y)I_y(x,y)$$

$$E_v(0,0)=0 \qquad E_{uv}(0,0)=\sum_{x,y} 2w(x,y)\,I_x(x,y)I_y(x,y)$$

Source: S. Lazebnik

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:

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

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

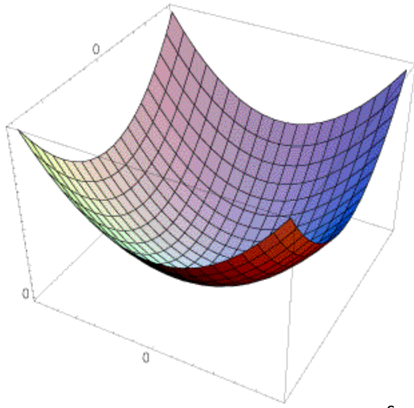
Source: S. Lazebnik

Interpreting the second moment matrix

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

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

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



Source：S. Lazebnik

Interpreting the second moment matrix

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

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

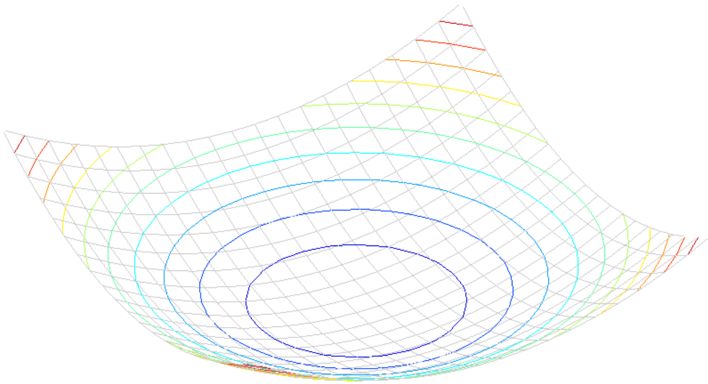
If either λ is close to 0, then this is **not** a corner, so look for locations where both are large.

Source：S. Lazebnik

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.



Source: S. Lazebnik

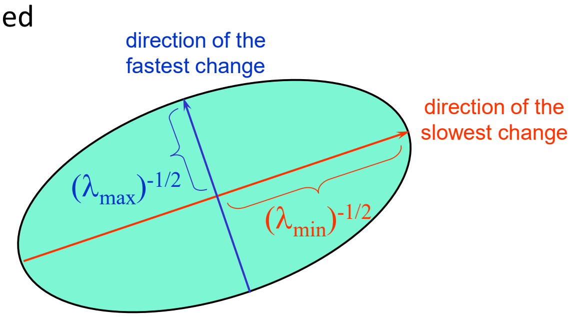
Interpreting the second moment matrix

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.

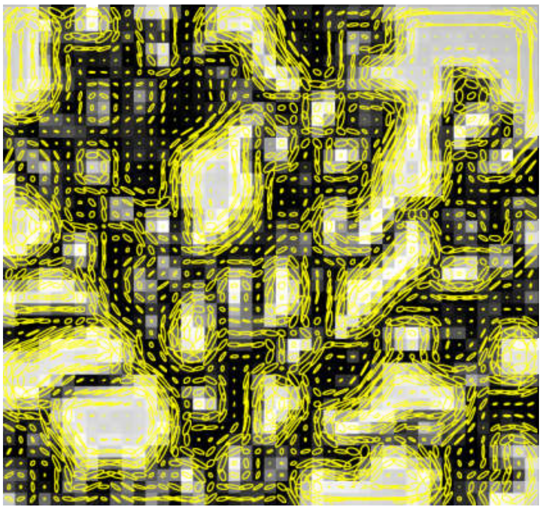
Diagonalization of M: $$M = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

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



Source: S. Lazebnik

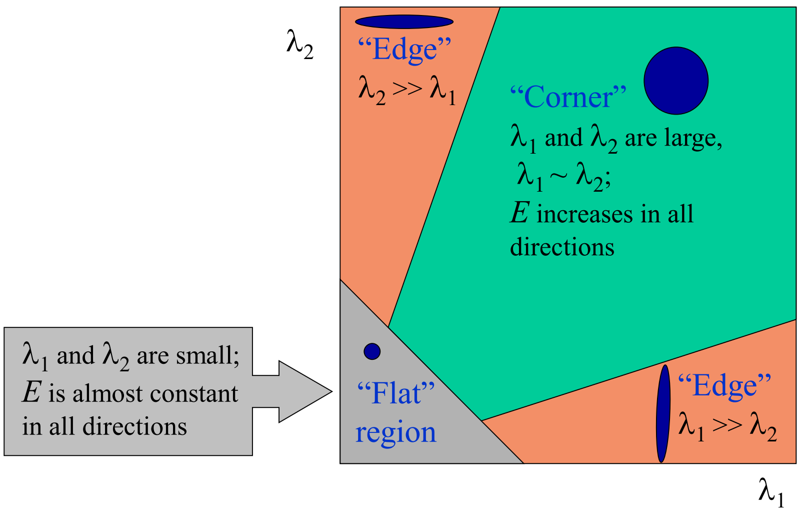
Visualization of second moment matrices



Source： S. Lazebnik

Interpreting the eigenvalues

Classification of image points using eigenvalues of *M*:

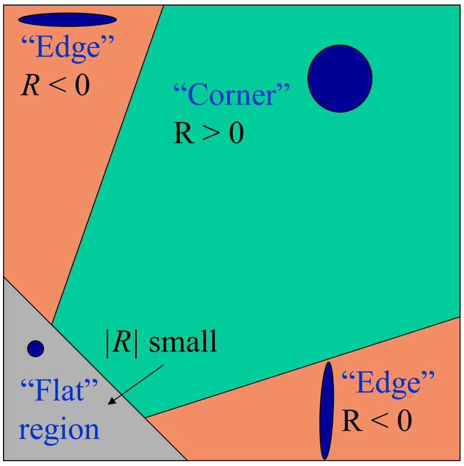


Source： S. Lazebnik

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)



Source：S. Lazebnik

Harris detector: 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 (nonmaximum suppression)

C.Harris and M.Stephens. "A Combined Corner and Edge Detector." *Proceedings of the 4th Alvey Vision Conference*: pages 147—151, 1988.

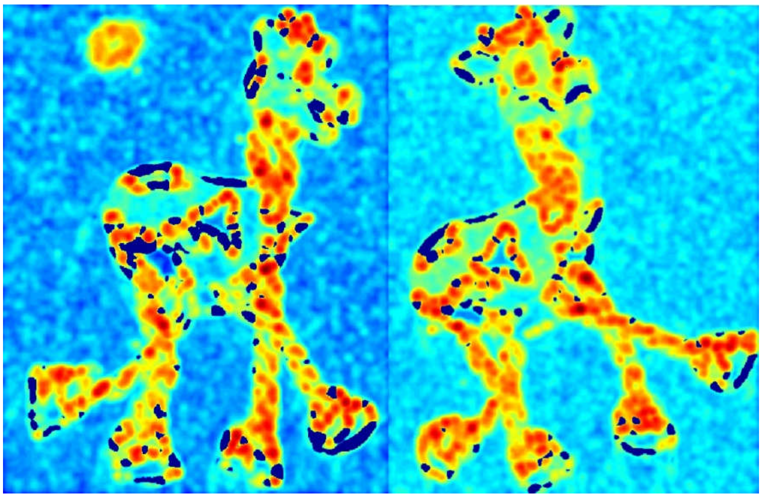
Source：S. Lazebnik

Harris Detector: Steps

Source： S. Lazebnik

Harris Detector: Steps

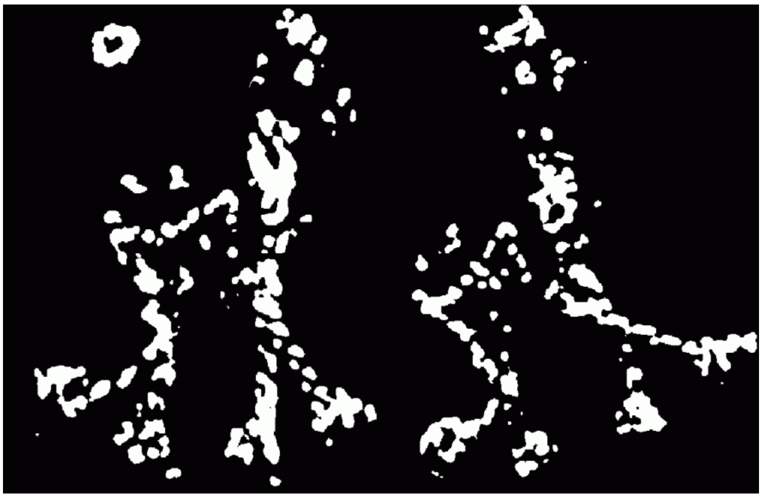
Compute corner response R



Source： S. Lazebnik

Harris Detector: Steps

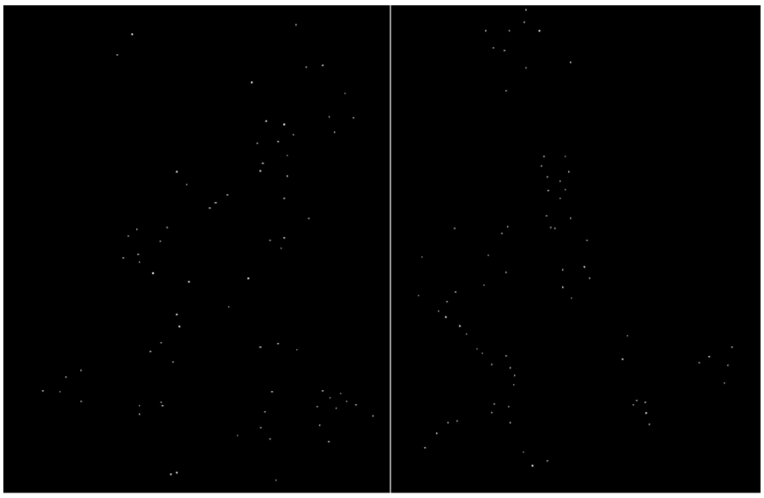
Find points with large corner response: R>threshold



Source： S. Lazebnik

Harris Detector: Steps

Take only the points of local maxima of R



Source： S. Lazebnik

Harris Detector: Steps

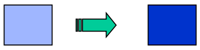
Source： S. Lazebnik

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

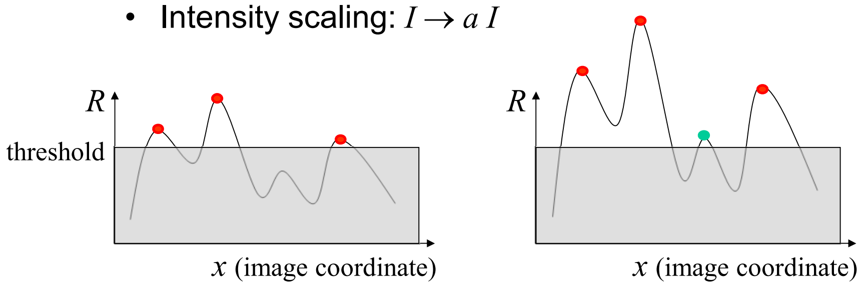
Source： S. Lazebnik

Affine intensity change



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

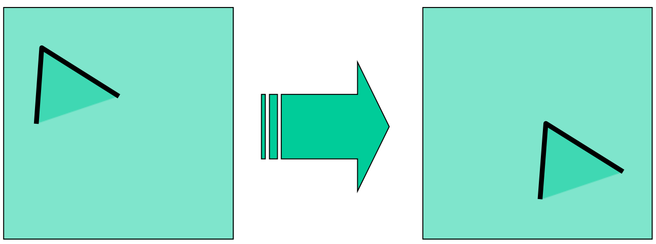
- Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$
- Intensity scaling: $I \rightarrow a\,I$



Partially invariant to affine intensity change

Source：S. Lazebnik

Image translation

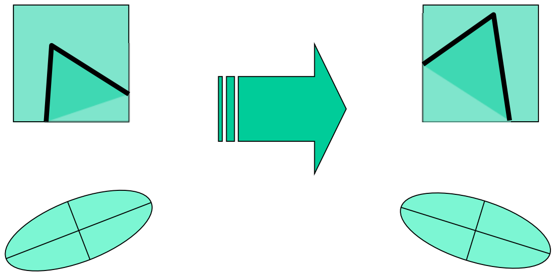


- Derivatives and window function are shift-invariant

Corner location is covariant w.r.t. translation

Source：S. Lazebnik

Image rotation

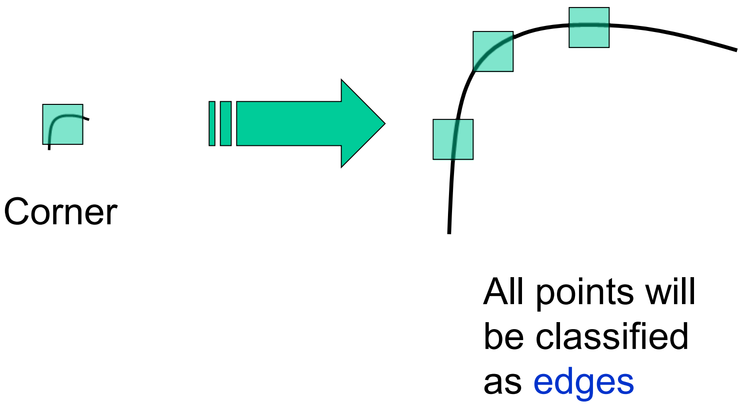


Second moment ellipse rotates but its shape (i.e. eigenvalues) remains the same

Corner location is covariant w.r.t. rotation

Source： S. Lazebnik

Scaling



Corner location is not covariant to scaling!

Source： S. Lazebnik