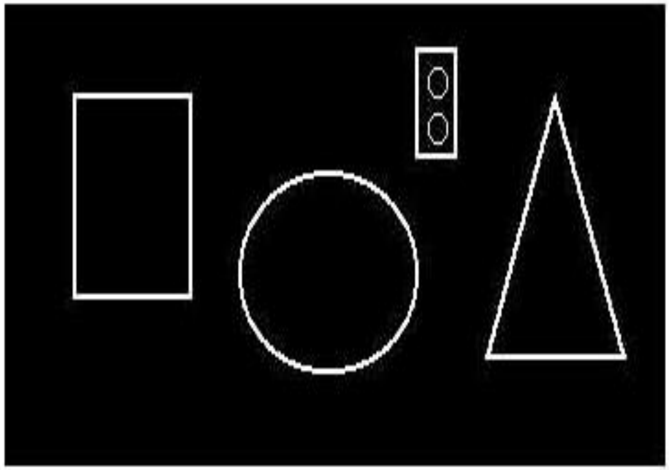
Hough Transform

Hough Transform

- It locates straight lines
- It locates straight l intervals
- It locates circles
- It locates algebraic curves
- It locates arbitrary specific shapes in a image



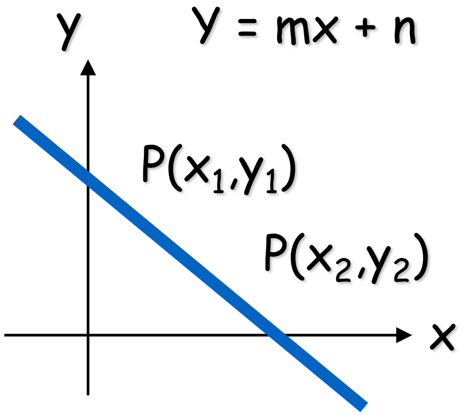
Hough Transform – cont.

- Straight line case
  - Consider a single isolated edge point (x_i, y_i)
    - There are an infinite number of lines that could pass through the points
  - Each of these lines can be characterized by some particular equation

$$y_i = mx_i + c$$

Line detection

- Mathematical model of a line:



$Y = mx + n$

$Y_1 = m\,x_1 + n$

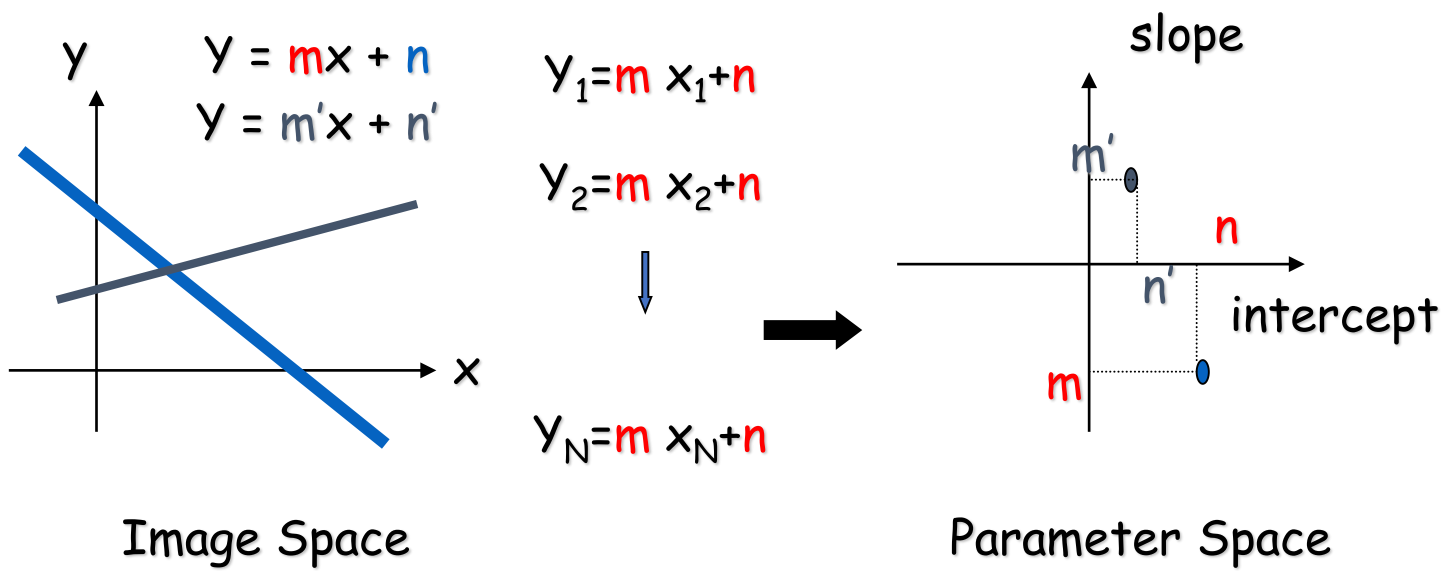
$Y_2 = m\,x_2 + n$

$\downarrow$

$Y_N = m\,x_N + n$

Hough Transform, Harish Corner Detector & SIFT

Image and Parameter Spaces



Line in Img. Space ~ Point in Param. Space

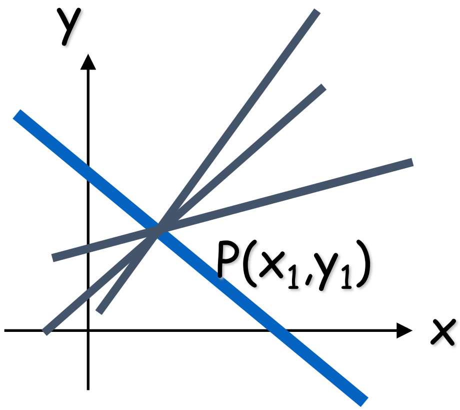
Image space

Fix (m,n), Vary (x,y) - Line

$Y = mx + n$

Fix (x_1,y_1), Vary (m,n) – Lines thru a Point

$Y_1 = m\,x_1 + n$

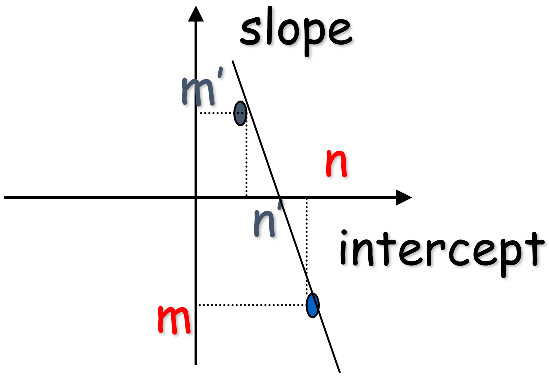


Parameter space

$Y_1 = m\, x_1 + n$ Can be re-written as: $n = -x_1\, m + Y_1$

Fix $(-x_1, y_1)$, Vary (m,n) - Line

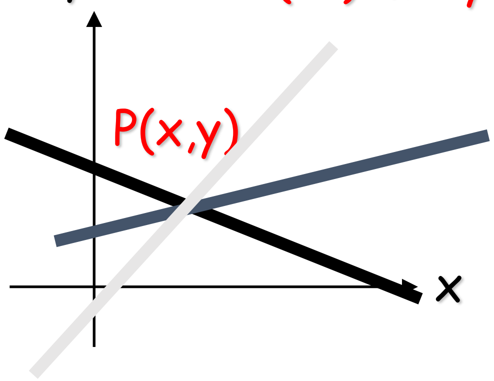
$N' = -x_1\, m' + Y_1$

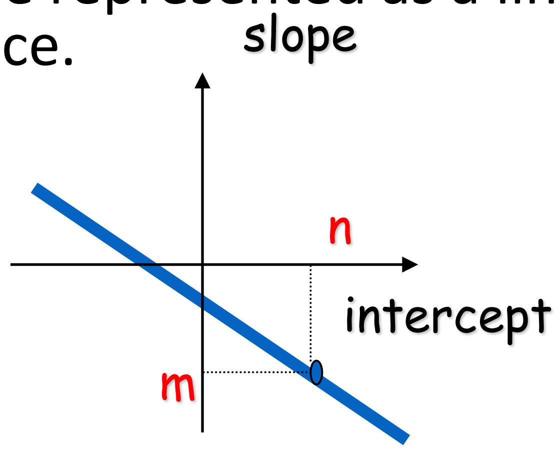


Hough Transform Technique

- Given an edge point, there is an infinite number of lines passing through it (Vary m and n).
  - These lines can be represented as a line in parameter space.

$n = (-x)\,m + y$





Parameter Space

Hough Transform Technique

- Given a set of collinear edge points, each of them have associated a line in parameter space.
  - These lines intersect at the point (m,n) corresponding to the parameters of the line in the image space.

HT - parametric representation

- y = kx + q
  - (x,y) - co-ordinates
  - k - gradient
  - q - y intercept
- Any straight line is characterized by k & q
  - use : ‘slope-intercept’ or (k,q) space not (x,y) space
  - (k,q) - parameter space
  - (x,y) - image space
  - can use (k,q) co-ordinates to represent a line

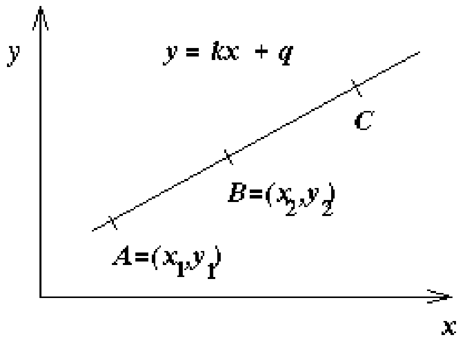
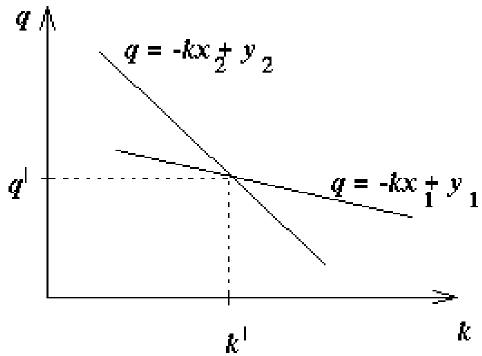


Image space



Parameter space

Hough Transform for lines

- Line equation: $y = mx + c \Longleftrightarrow c = -mx + y$
- Each point in image space corresponds to a line in parameter space.
- An intersection point in parameter space corresponds to a line in image space.

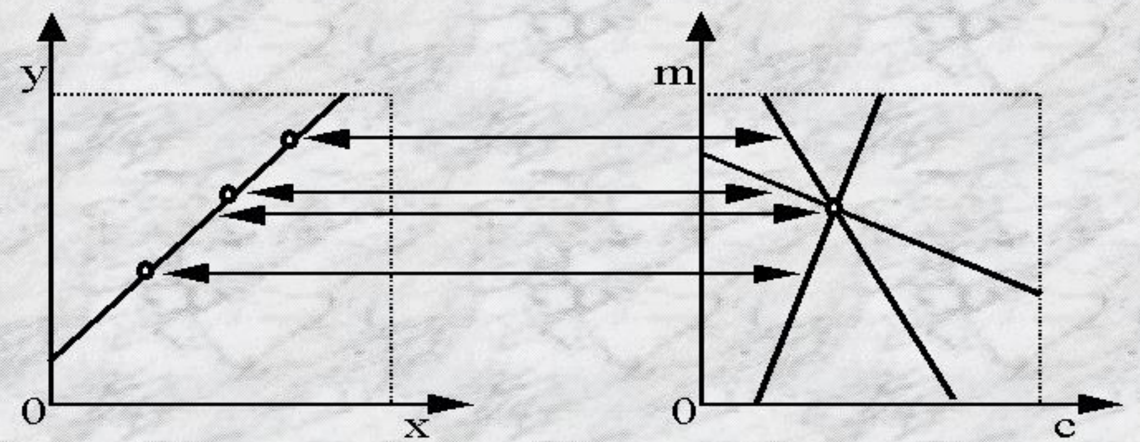


Image Parameter Spaces

- Image Space
  - Lines
  - Points
  - Collinear points

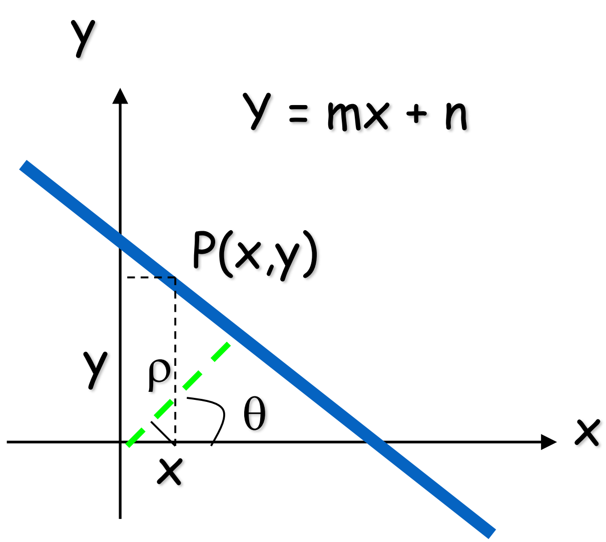
- Parameter Space
  - Points
  - Lines
  - Intersecting lines

Practical Issues with This Hough Parameterization

- The slope of the line is $-\infty<m<\infty$
  - The parameter space is INFINITE
- The representation $y = mx + n$

Solution:

- Use the “Normal” equation of a line:



$\rho = x \cos\theta + y \sin\theta$

θ Is the line orientation

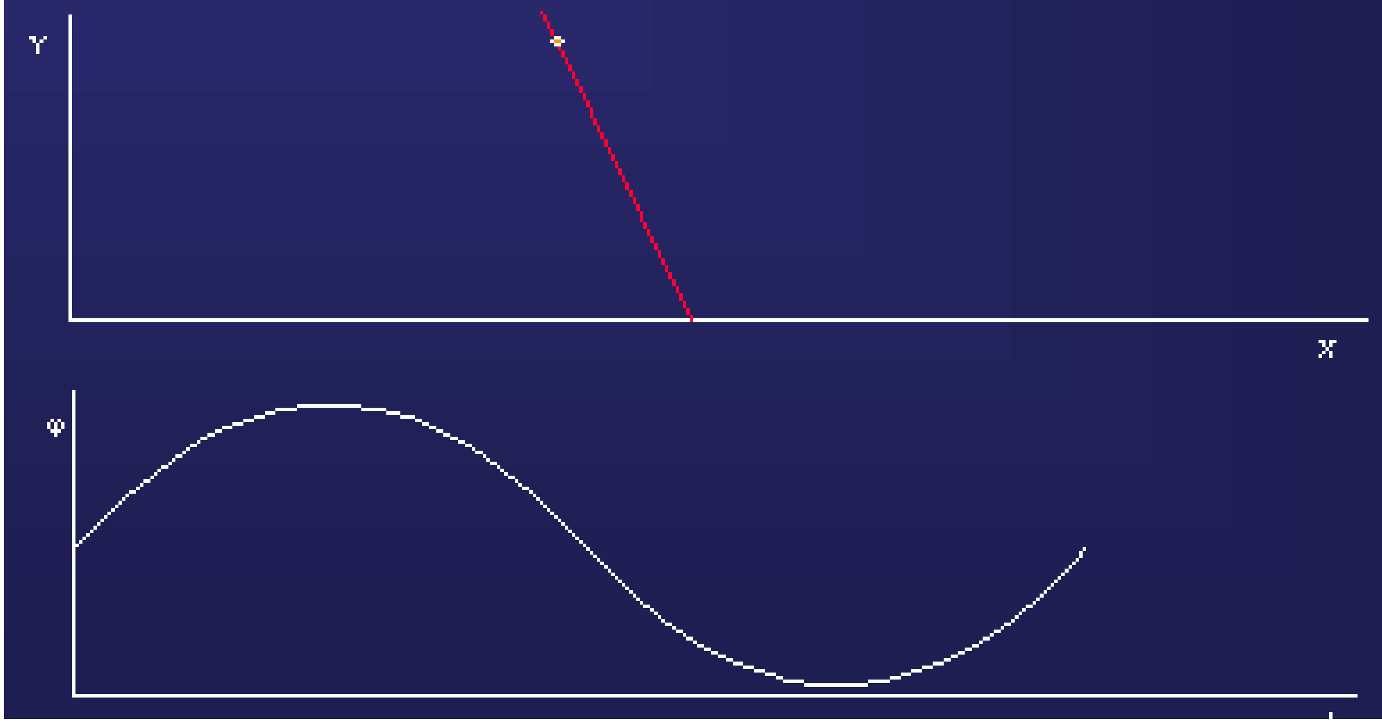
ρ Is the distance between the origin and the line

Hough Transform, Harish Corner Detector & SIFT

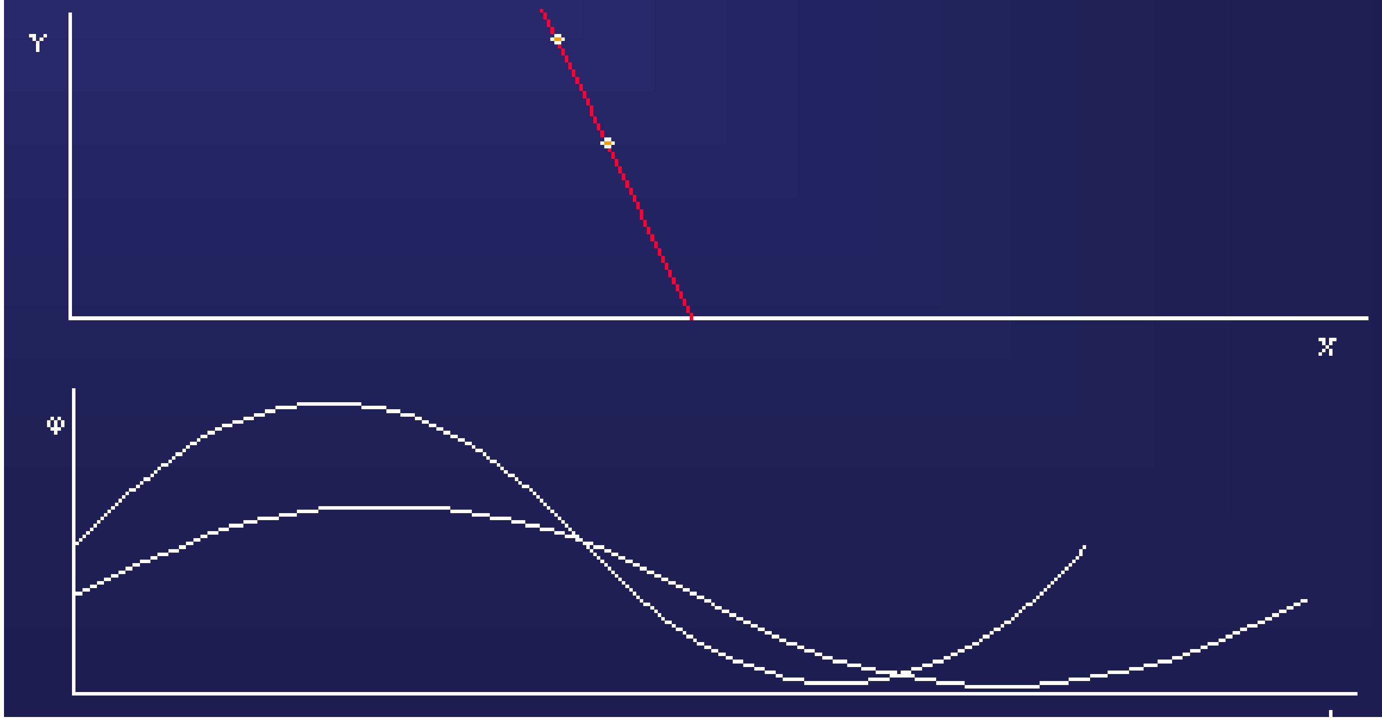
Another Viewpoint

- Since any (r,φ) represents a point in parameter space we may plot all possible (r,φ) points for each (x,y) poi

- Taking any particular (x,y) point its set of (r,φ) points is a sinusoid through parameter space



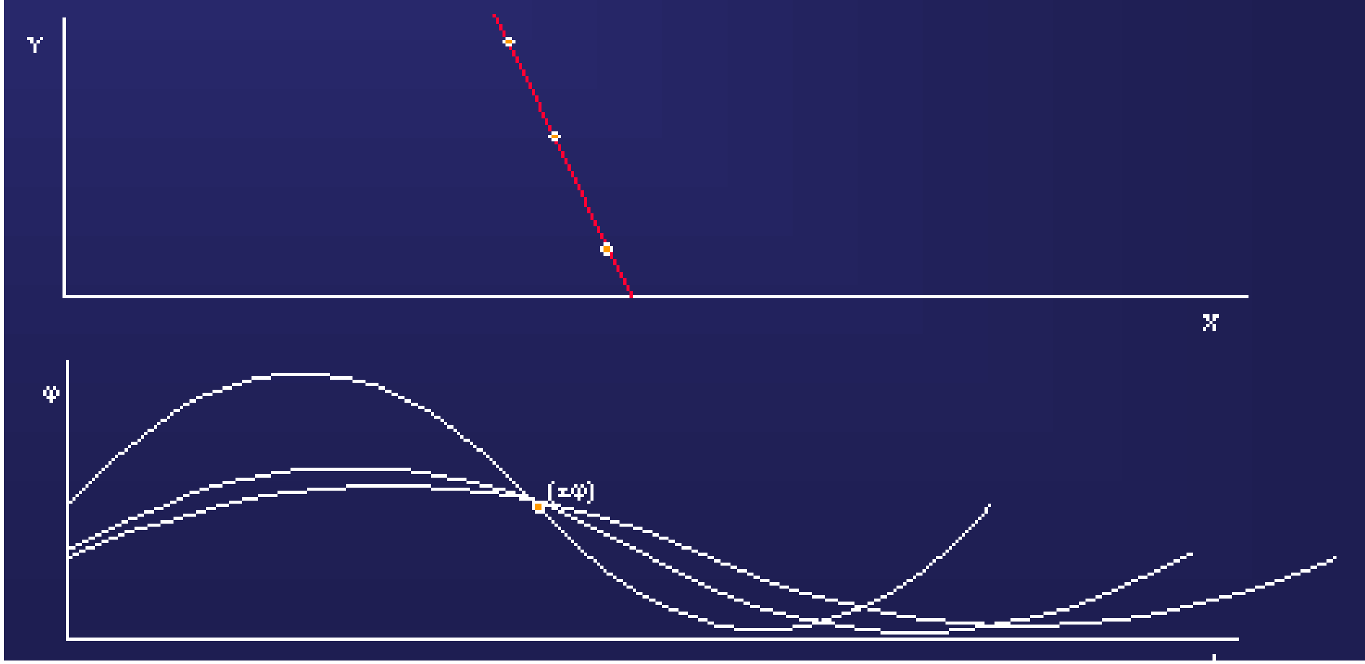
- Each sinusoid tells us every possible line that can pass through the corresponding (x,y) point



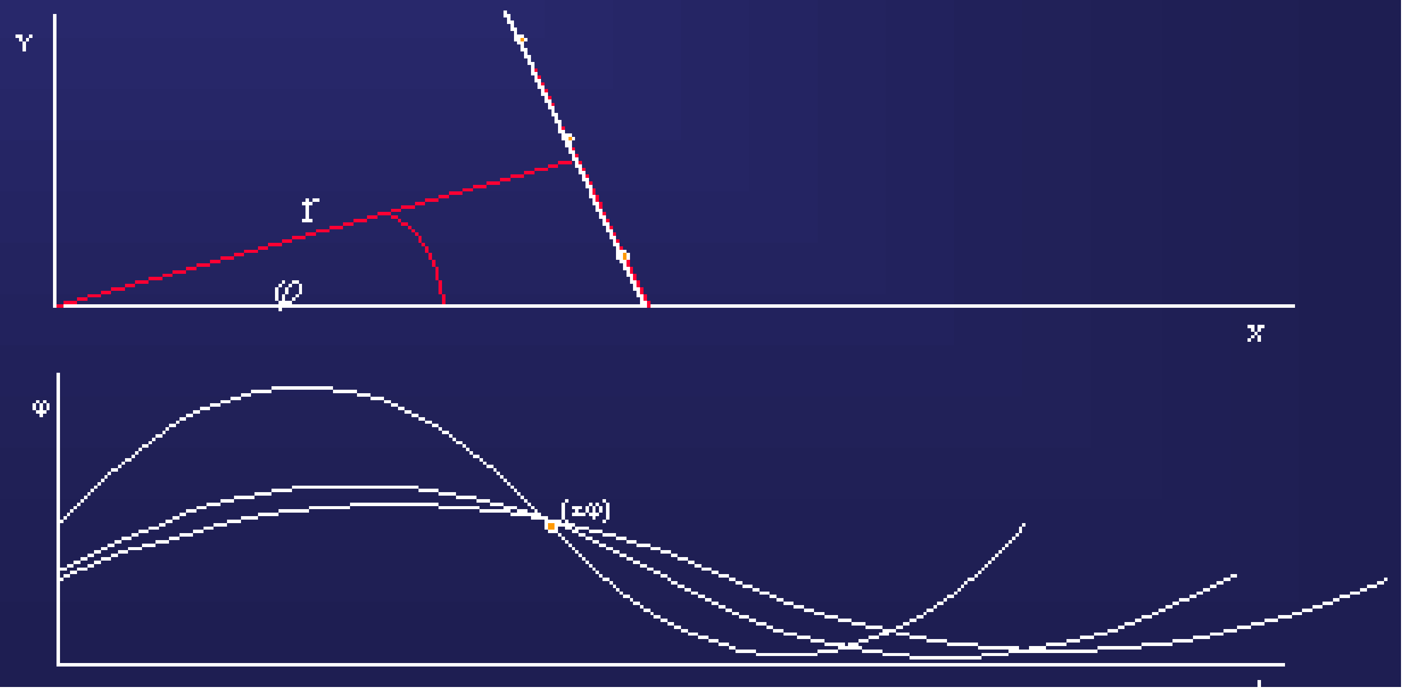
Hough Transform, Harish Corner Detector & SIFT

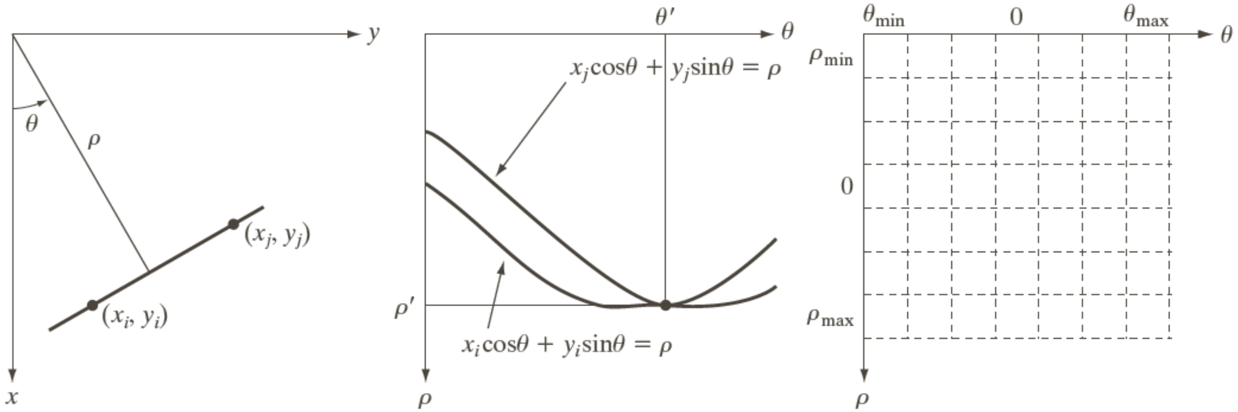
Hough Transform, Harish Corner Detector & SIFT

- The line upon which the (x,y) points lie is then given by the (r,φ) pair on which all the sinusoids *agree* (i.e. intersect)



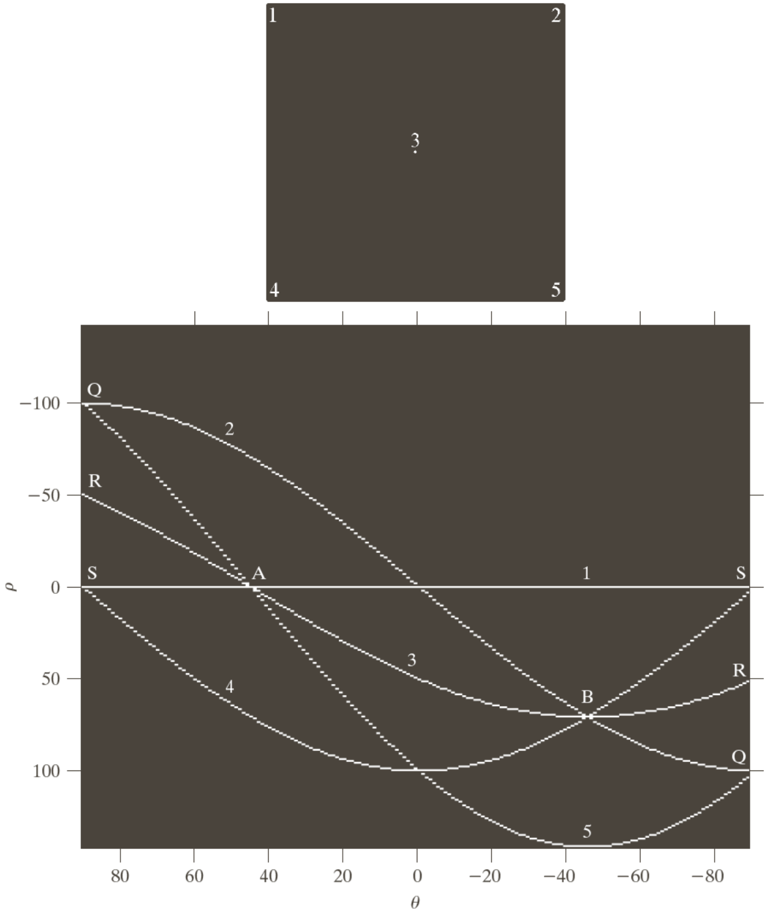
- Finding this intersection we then have the required (r,φ) parameters and therefore the line in the image





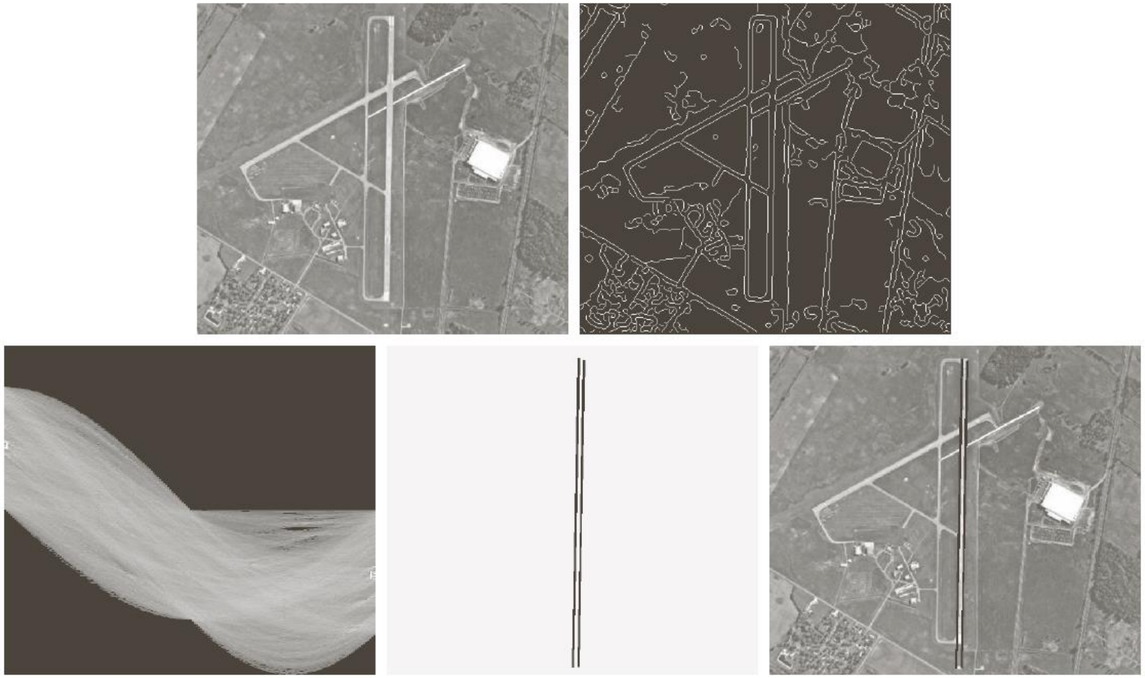
a b c

FIGURE 10.32 (a) (ρ, θ) parameterization of line in the xy-plane. (b) Sinusoidal curves in the $\rho\theta$-plane; the point of intersection (ρ', θ') corresponds to the line passing through points (x_i, y_i) and (x_j, y_j) in the xy-plane. (c) Division of the $\rho\theta$-plane into accumulator cells.



a
b

FIGURE 10.33
(a) Image of size 101 × 101 pixels, containing five points.
(b) Corresponding parameter space. (The points in (a) were enlarged to make them easier to see.)



a b
c d e

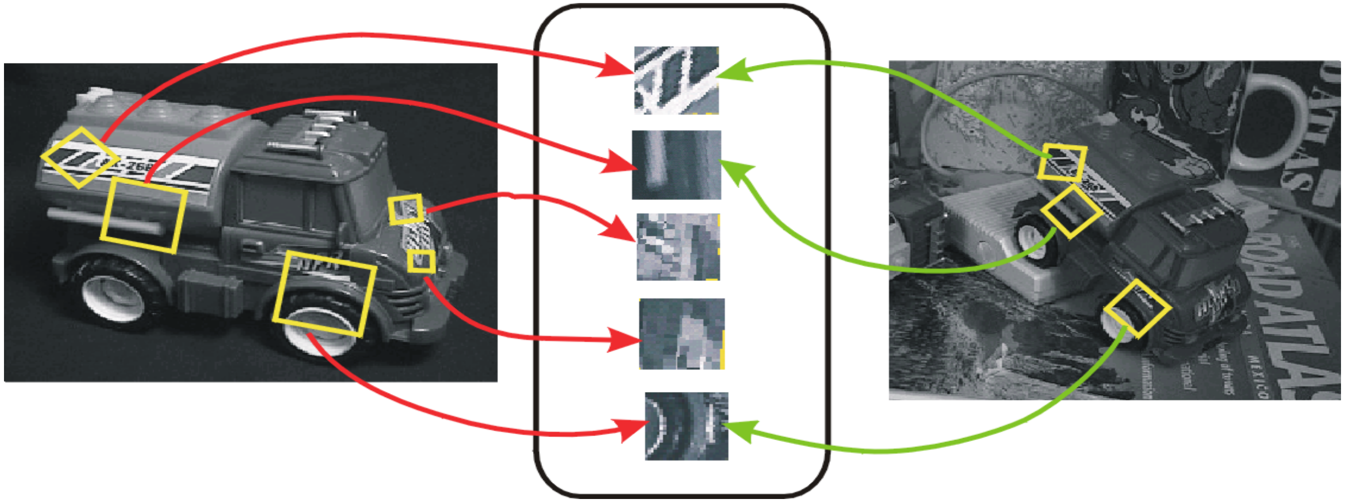
FIGURE 10.34 (a) A 502 × 564 aerial image of an airport. (b) Edge image obtained using Canny's algorithm. (c) Hough parameter space (the boxes highlight the points associated with long vertical lines). (d) Lines in the image plane corresponding to the points highlighted by the boxes). (e) Lines superimposed on the original image.

Hough Transform, Harish Corner Detector & SIFT

Harris Corner Detection

Invariant Local Features

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



Features Descriptors

More motivation…

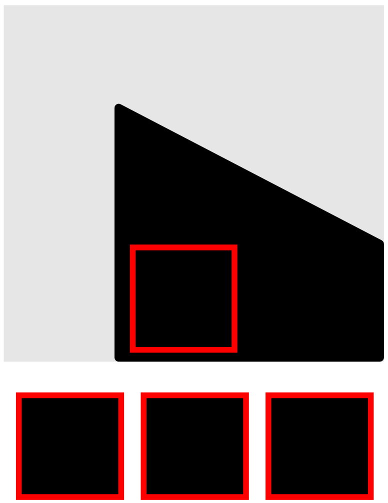
- Feature points are used for:
  - Image alignment (homography, fundamental matrix)
  - 3D reconstruction
  - Motion tracking
  - Object recognition
  - Indexing and database retrieval
  - Robot navigation
  - … other

Hough Transform, Harish Corner Detector & SIFT

Moravec corner detector (1980)

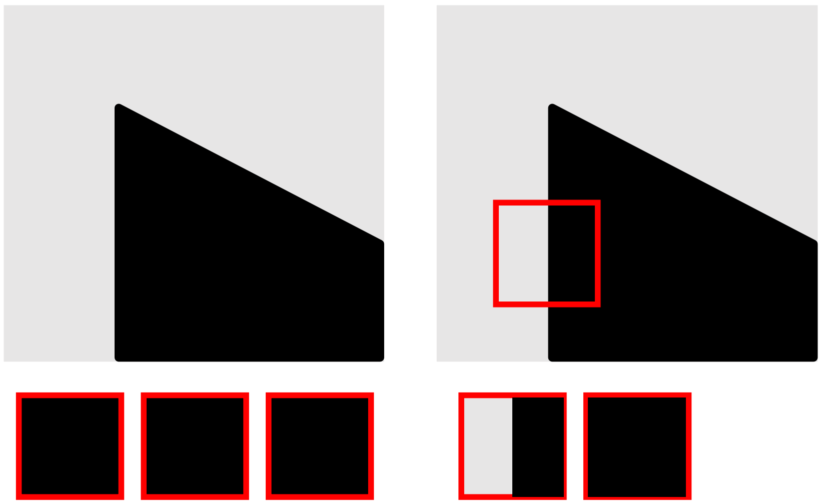
- We should easily recognize the point by looking through a small window
- Shifting a window in *any direction* should give *a large change* in intensity

Moravec corner detector



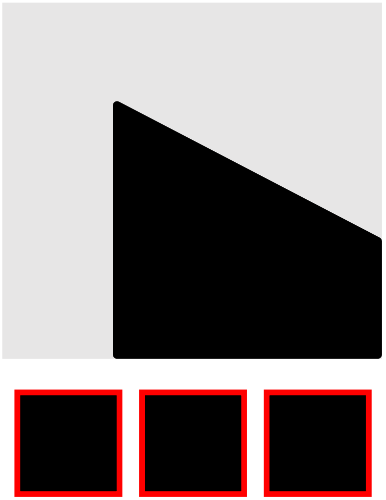
flat

Moravec corner detector

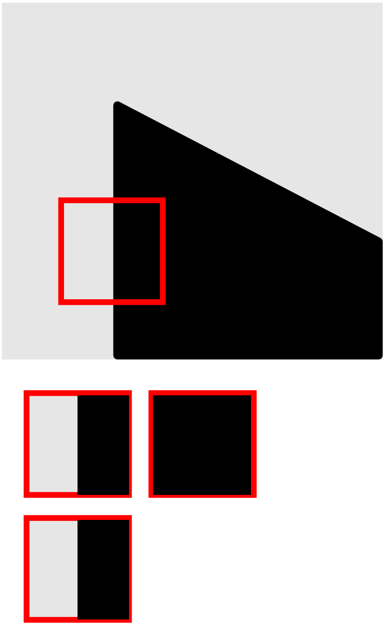


flat

Moravec corner detector

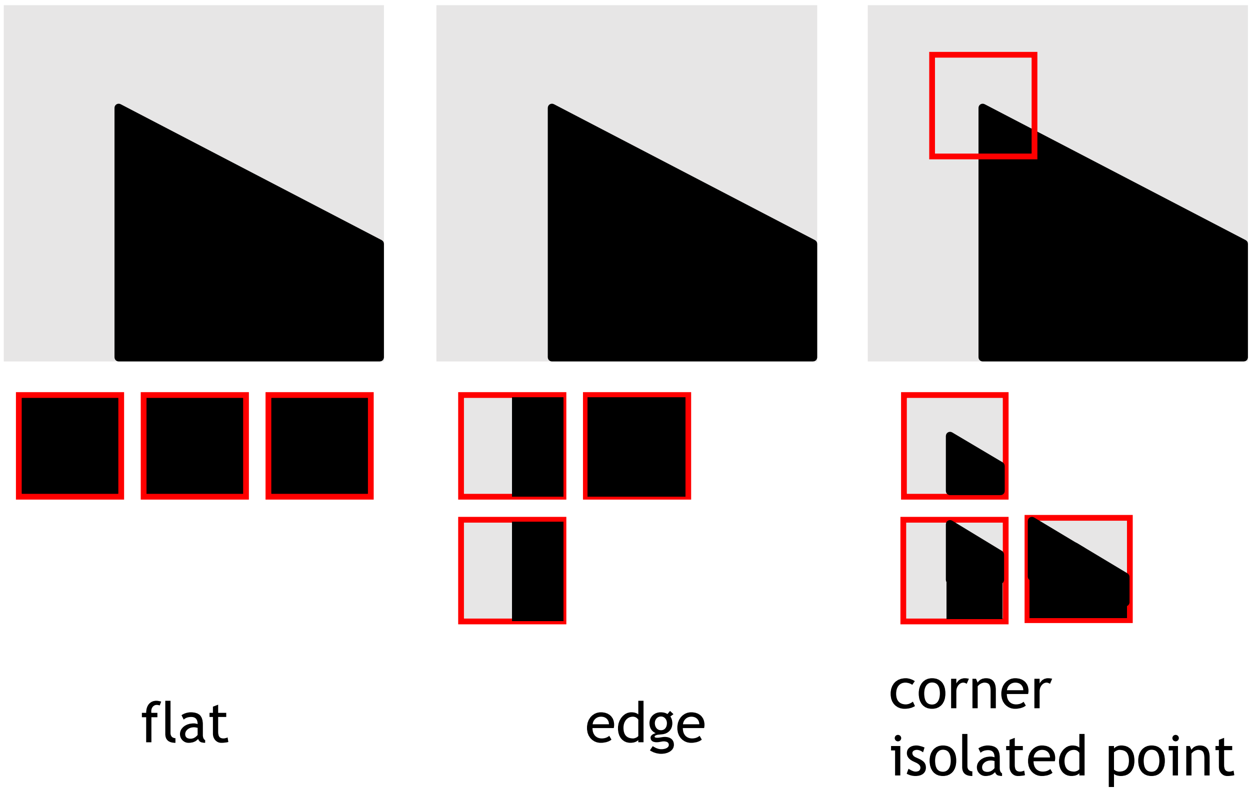


flat



edge

Moravec corner detector



Hough Transform, Harish Corner Detector & SIFT

Hough Transform, Harish Corner Detector & SIFT

Moravec corner detector

Change of intensity for the shift [*u,v*]:

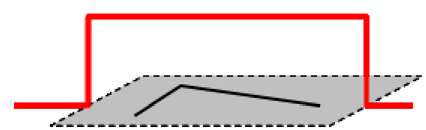
$$E(u,v)=\sum_{x,y} w(x,y)\left[I(x+u,y+v)-I(x,y)\right]^2$$

Window function

Shifted intensity

Intensity

Window function $w(x,y) =$



1 in window, 0 outside

Four shifts: (u,v) = (1,0), (1,1), (0,1), (-1, 1)

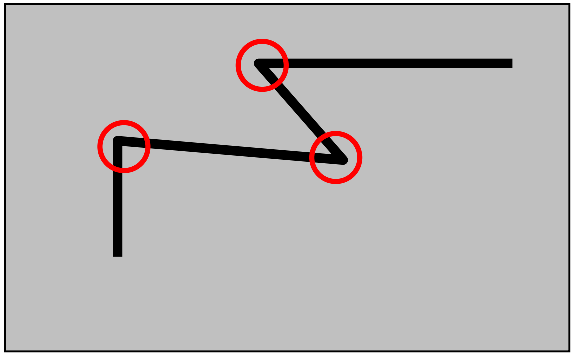
Look for local maxima in *min{E}*

Problems of Moravec detector

- Noisy response due to a binary window function
- Responds too strong for edges because only minimum of E is taken into account

⇨Harris corner detector (1988) solves these problems.

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



Harris Detector: Mathematics

Change of intensity for the shift $[u,v]$:

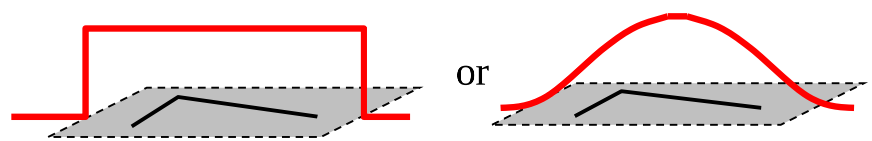
$$E(u,v)=\sum_{x,y} w(x,y)\left[I(x+u,y+v)-I(x,y)\right]^2$$

Window function

Shifted intensity

Intensity

Window function $w(x,y)$ =



or

1 in window, 0 outside

Gaussian

Change of intensity 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

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.

Hough Transform, Harish Corner Detector & SIFT

Taylor Series for 2D Functions

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

First partial derivatives

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

Second partial derivatives

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

Third partial derivatives

$+ \ \ldots$ (Higher order terms)

First order approx

$$f(x+u,y+v) \approx f(x,y) + uf_x(x,y) + vf_y(x,y)$$

$$\sum [I(x+u, y+v) - I(x,y)]^2$$

$$\approx \sum [I(x,y) + uI_x + vI_y - I(x,y)]^2$$ First order approx

$$= \sum u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2$$

$$= \sum \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$ Rewrite as matrix equation

$$= \begin{bmatrix} u & v \end{bmatrix} \left(\sum \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \right) \begin{bmatrix} u \\ v \end{bmatrix}$$

Harris Detector: Mathematics

For small shifts $[u,v]$ we have a *bilinear* approximation:

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

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

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

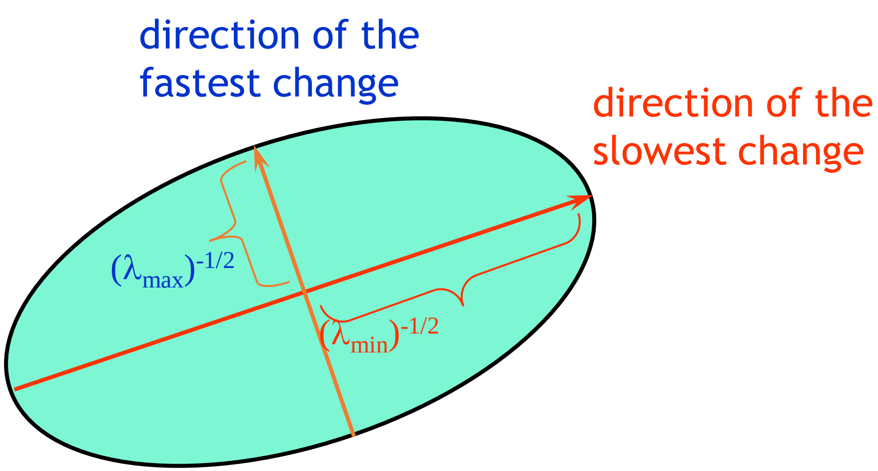
Harris corner detector

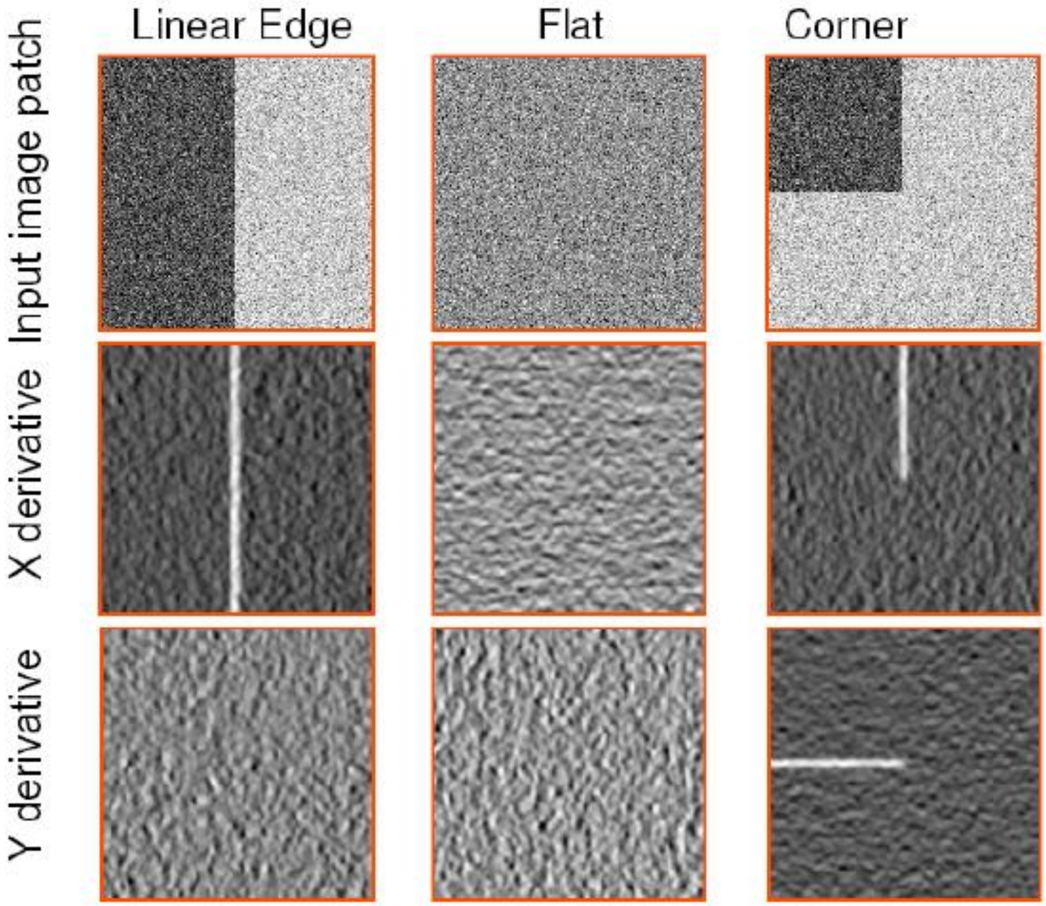
Intensity change in shifting window: eigenvalue analysis

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

λ_1, λ_2 – eigenvalues of M

Ellipse *E(u,v)* = const

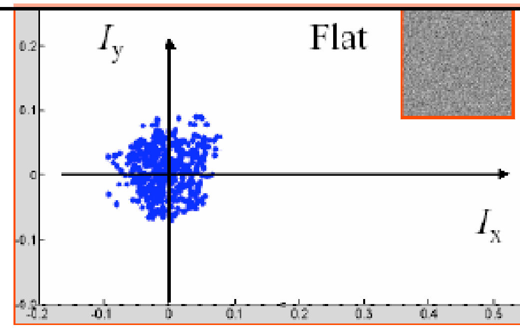


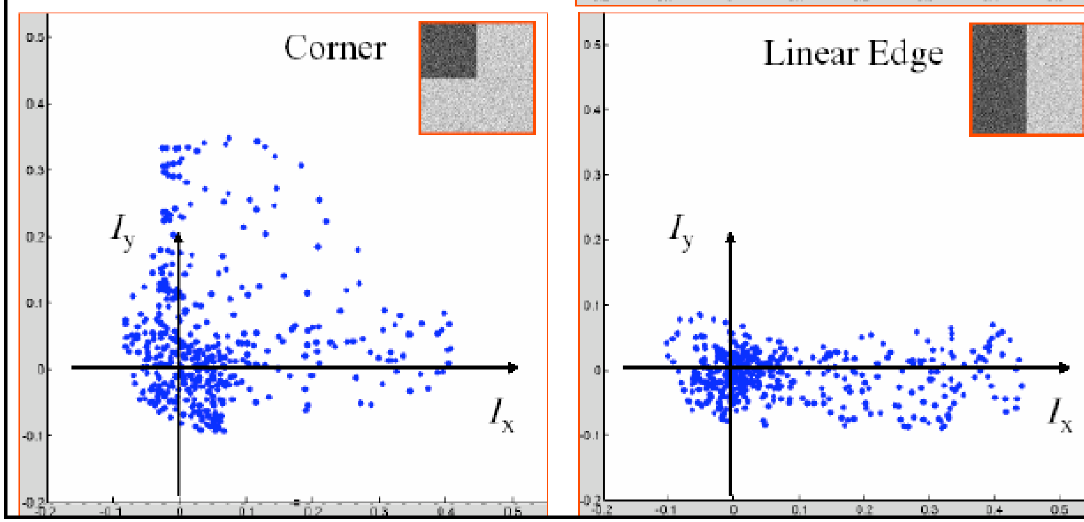


Hough Transform, Harish Corner Detector & SIFT

Plotting Derivatives as 2D Points

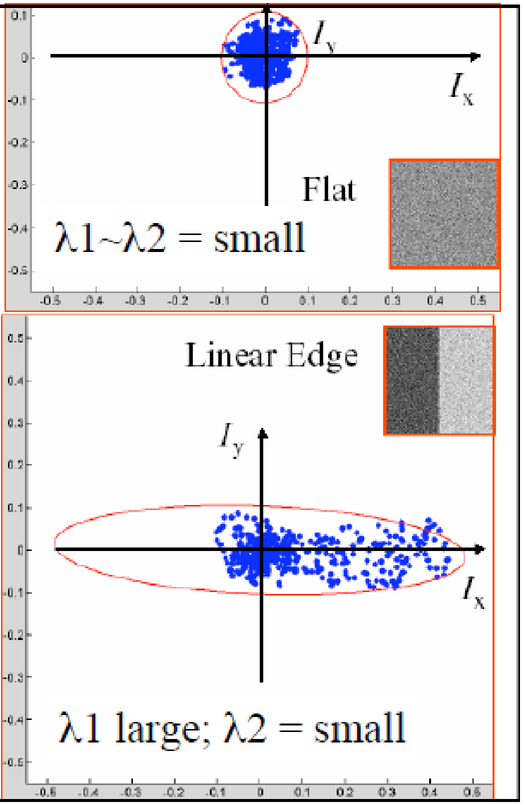
The distribution of the x and y derivatives is very different for all three types of patches





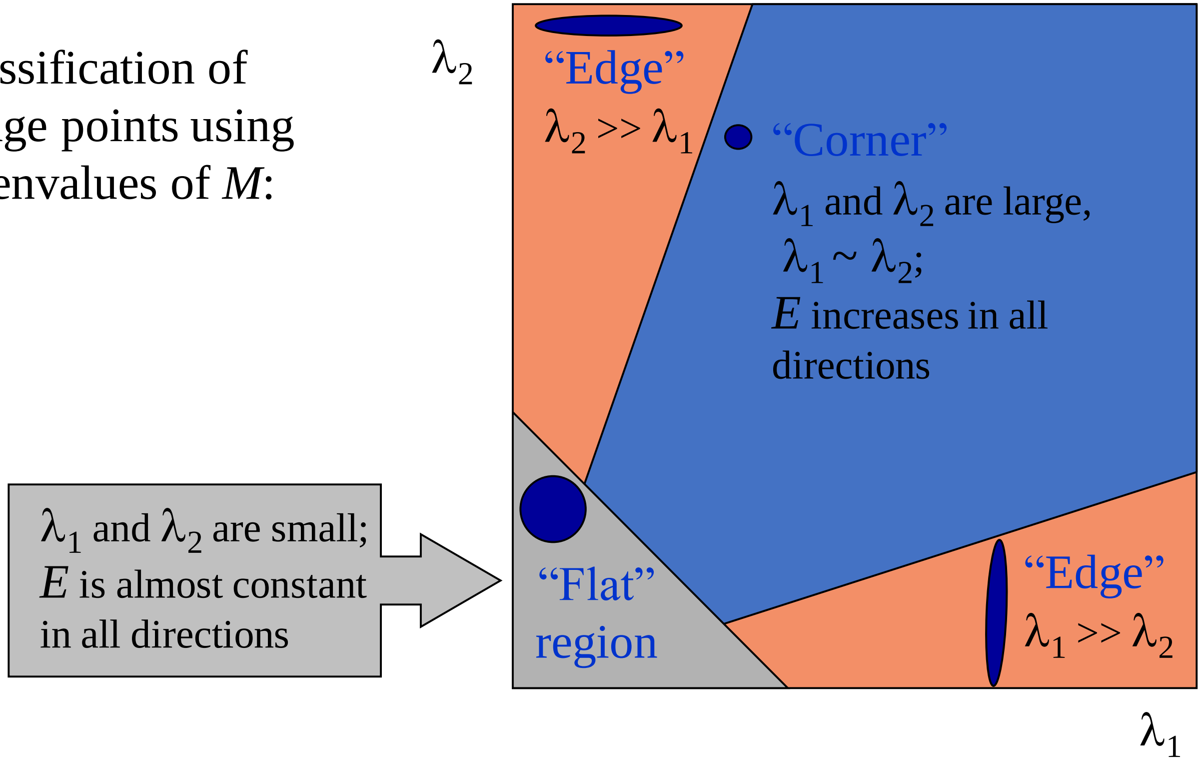
Fitting Ellipse to Each Set of Points

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



Harris Detector: Mathematics

Classification of image points using eigenvalues of M:



Harris Detector: Mathematics

Measure of corner response:

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

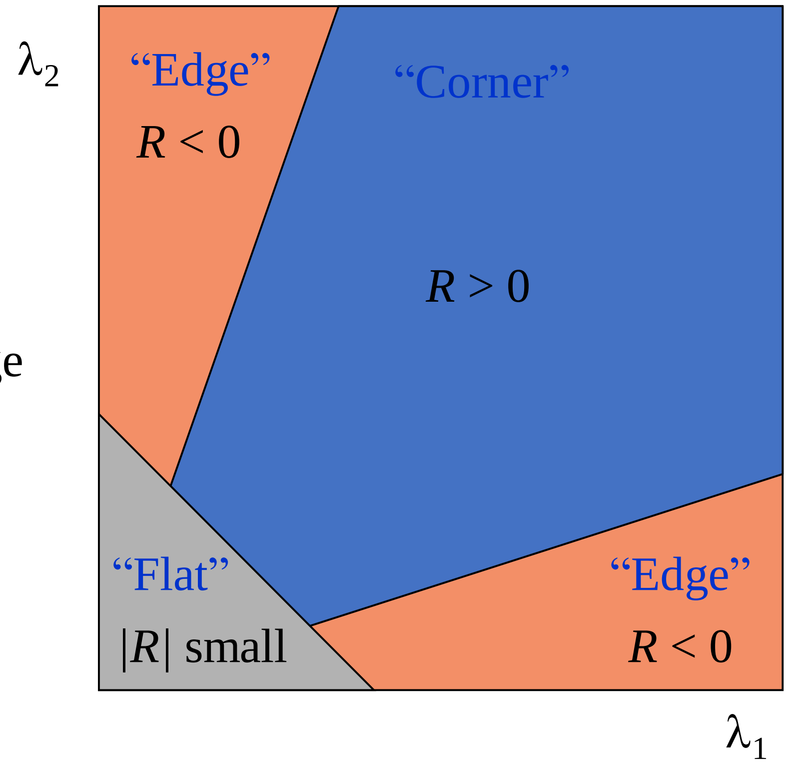
$$\det M = \lambda_1 \lambda_2$$
$$\operatorname{trace} M = \lambda_1 + \lambda_2$$

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

Hough Transform, Harish Corner Detector & SIFT

Harris 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



Summary of Harris detector

1. Compute x and y derivatives of image

$$I_x = G_\sigma^x * I \quad I_y = G_\sigma^y * I$$

2. Compute products of derivatives at every pixel

$$I_{x2} = I_x . I_x \quad I_{y2} = I_y . I_y \quad I_{xy} = I_x . I_y$$

3. Compute the sums of the products of derivatives at each pixel

$$S_{x2} = G_{\sigma'} * I_{x2} \quad S_{y2} = G_{\sigma'} * I_{y2} \quad S_{xy} = G_{\sigma'} * I_{xy}$$

4. Define at each pixel (x, y) the matrix

$$H(x,y) = \begin{bmatrix} S_{x2}(x,y) & S_{xy}(x,y) \\ S_{xy}(x,y) & S_{y2}(x,y) \end{bmatrix}$$

5. Compute the response of the detector at each pixel

$$R = Det(H) - k(Trace(H))^2$$

6. Threshold on value of R. Compute nonmax suppression.

Harris Detector: Summary

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

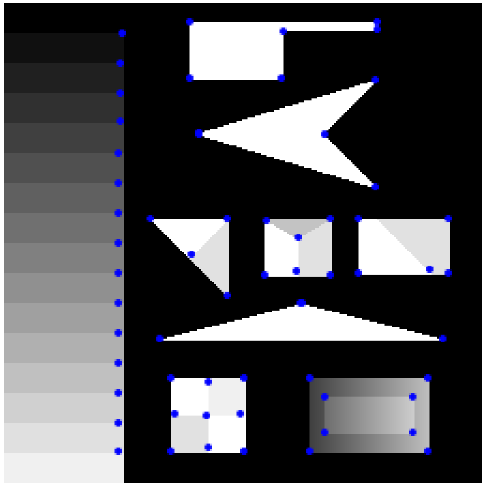
$$E(u,v) \cong [u,v] \ M \begin{bmatrix} u \\ v \end{bmatrix}$$

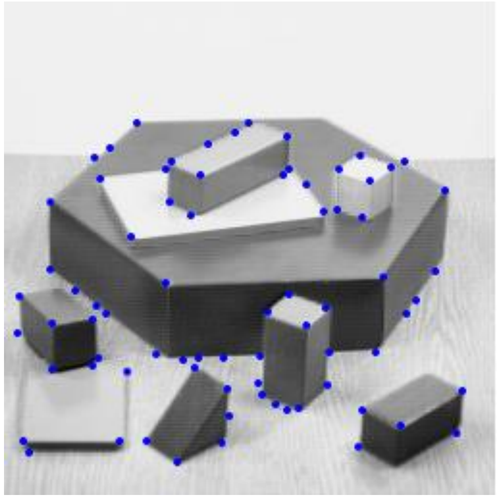
- Describe a point in terms of eigenvalues of *M*: *measure of corner response*

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

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

Examples



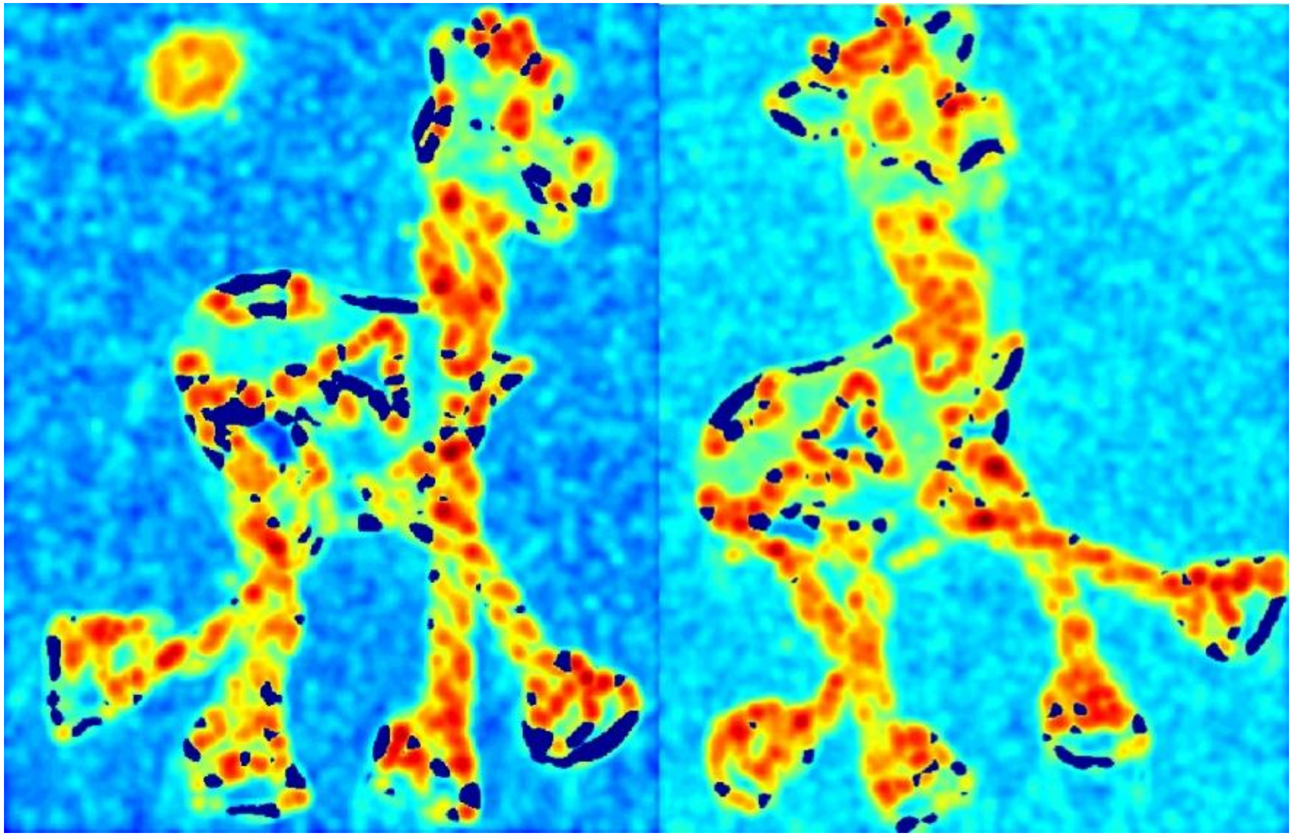


Examples

Harris Detector: Workflow

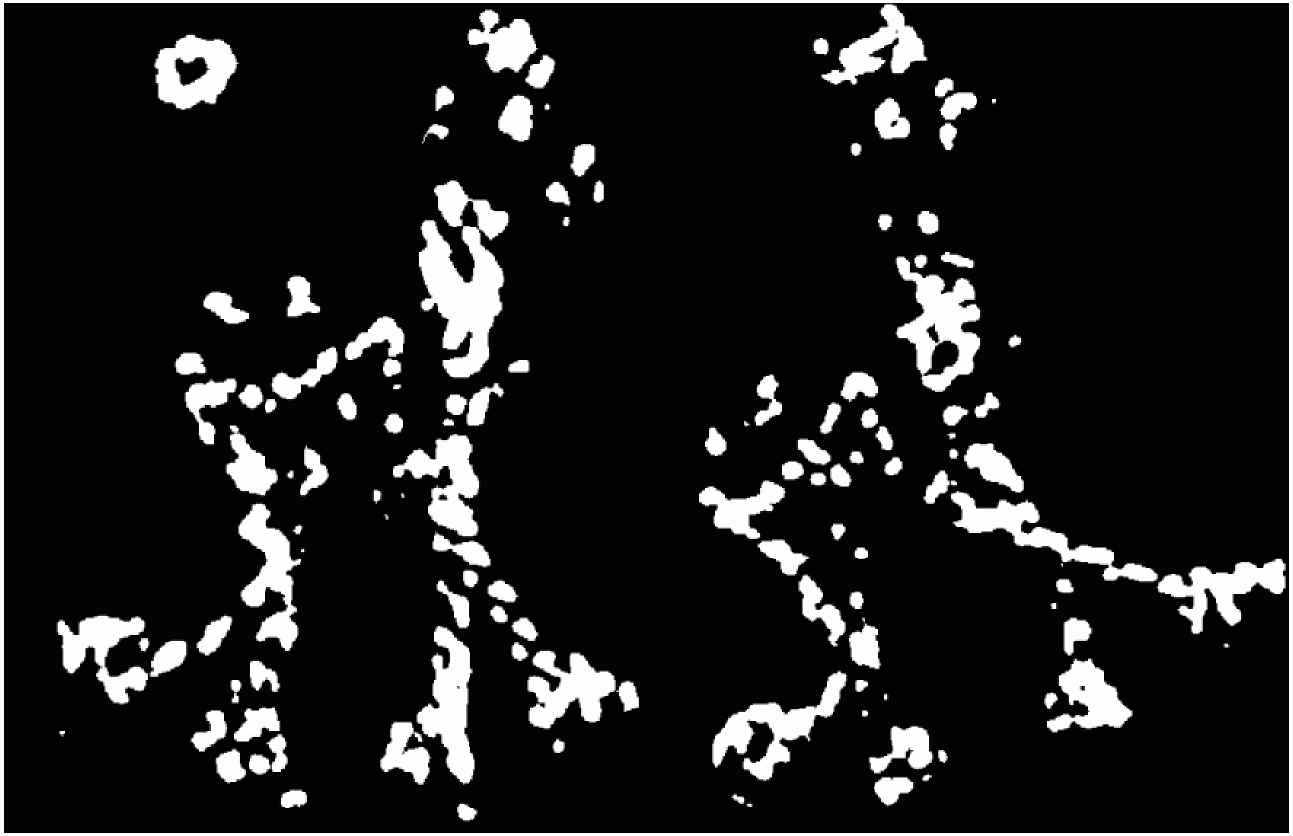
Harris Detector: Workflow

Compute corner response R



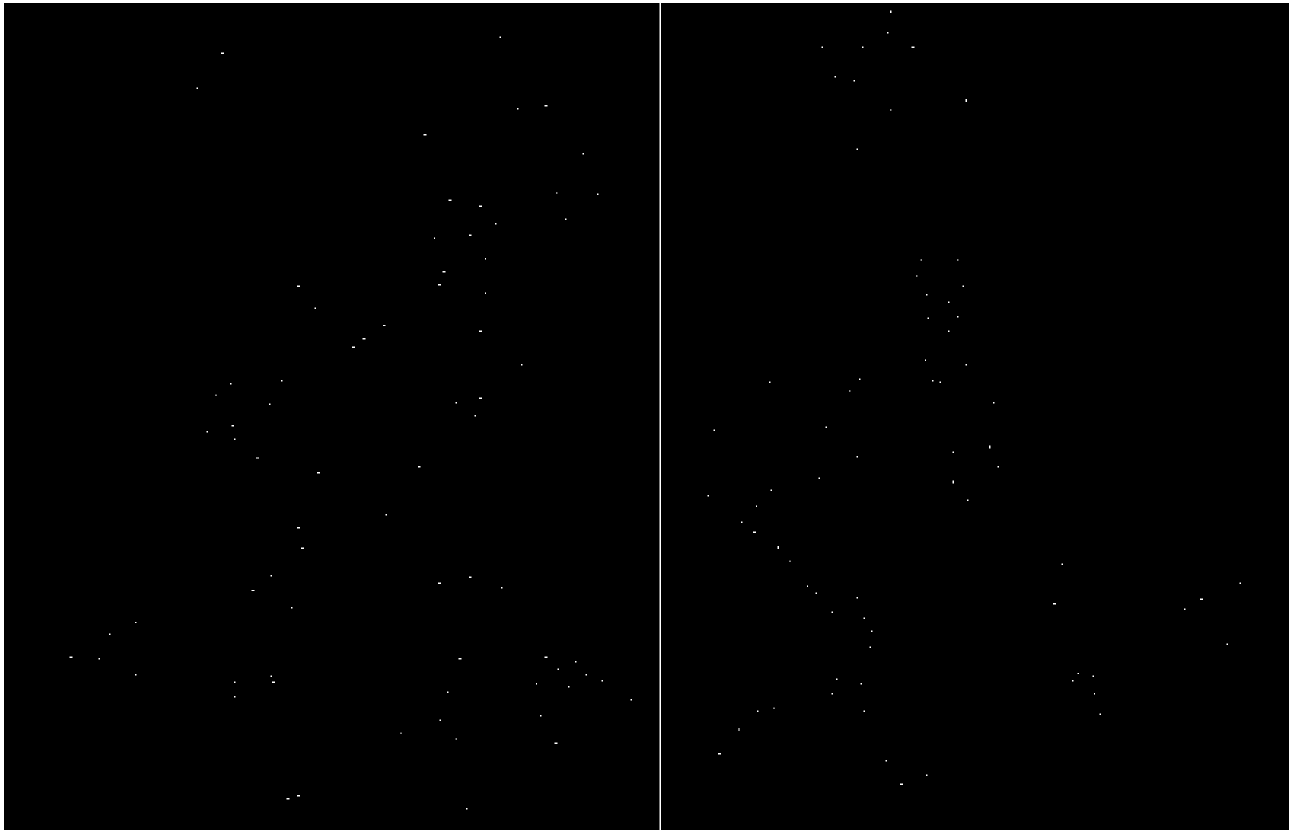
Harris Detector: Workflow

Find points with large corner response: R>threshold



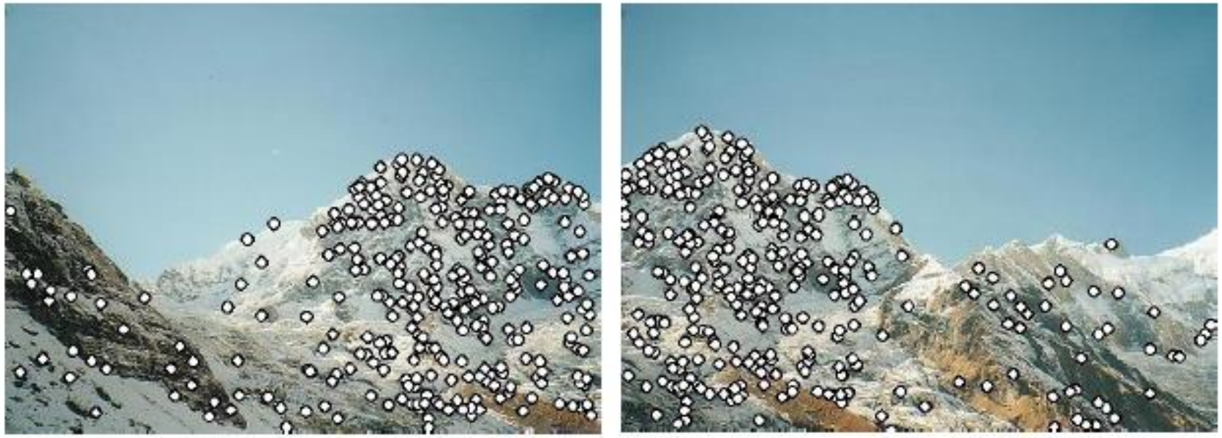
Harris Detector: Workflow

Take only the points of local maxima of R



Hough Transform, Harish Corner Detector & SIFT

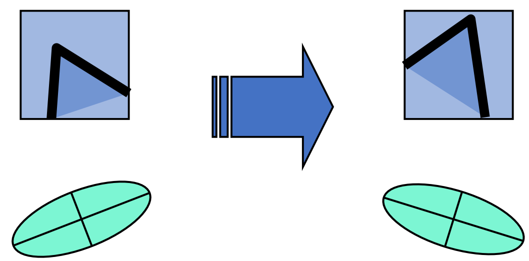
Harris Detector: Workflow



Detected Harris Corner Points

Harris Detector: Some Properties

- Rotation invariance



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

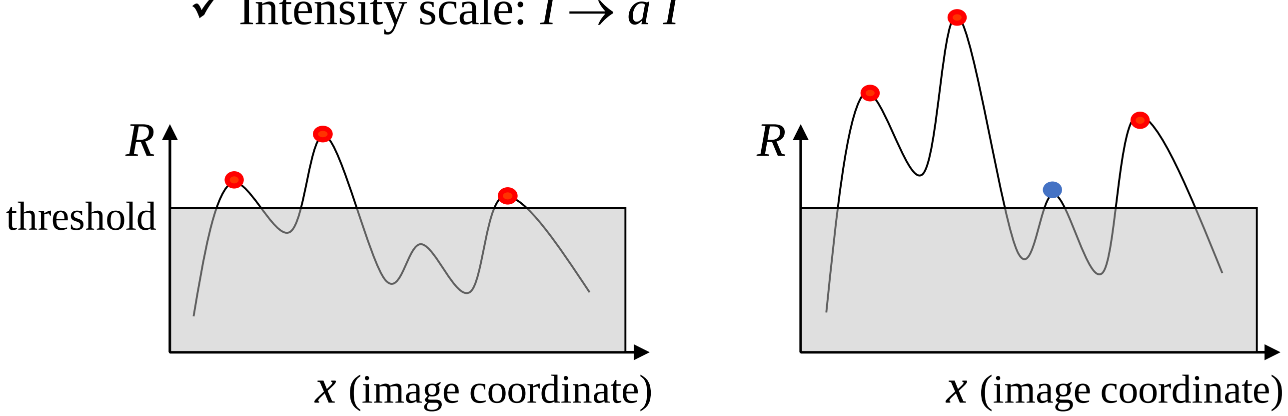
Corner response R is invariant to image rotation

Harris Detector: Some Properties

- Partial invariance to *affine intensity* change

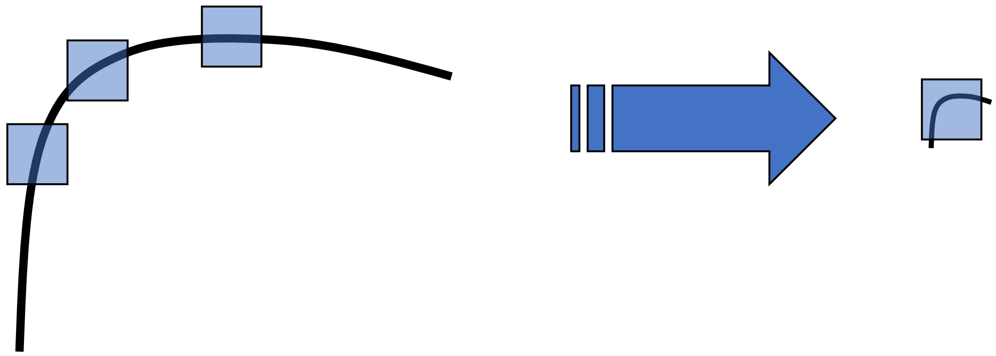
✓ Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$

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



Harris Detector: Some Properties

- But: non-invariant to *image scale*!

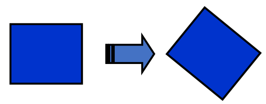


All points will be classified as edges

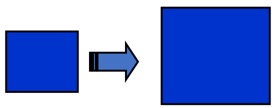
Corner !

Models of Image Change

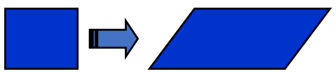
- Geometry
  - Rotation



  - Similarity (rotation + uniform scale)

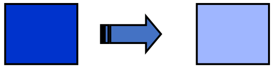


  - Affine (scale dependent on direction)



  - valid for: orthographic camera, locally planar object

- Photometry
  - Affine intensity change ($I \rightarrow a\, I + b$)



Hough Transform, Harish Corner Detector & SIFT

Examples

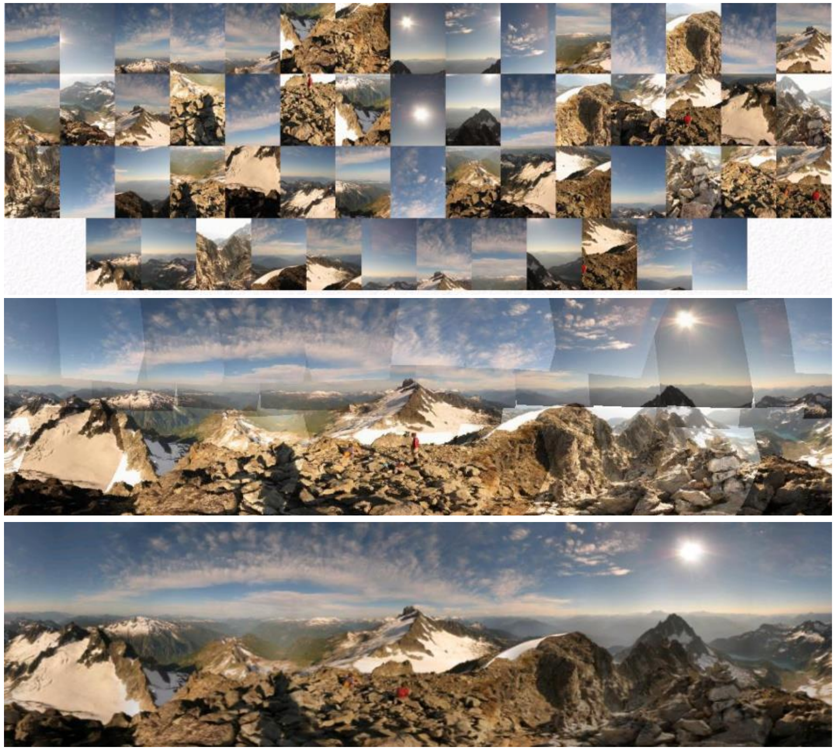
Scale Invariant Feature Transform (SIFT)

Introduction

- Initially proposed for correspondence matching
  - Proven to be the most effective in such cases according to a recent performance study by Mikolajczyk & Schmid (ICCV ’03)

Introduction

- Automatic Mosaicing



- http://www.cs.ubc.ca/~mbrown/autostitch/autostitch.html

Introduction

- Now being used for general object class recognition (e.g. 2005 Pascal challenge)

- Histogram of gradients
  - Human detection, Dalal & Triggs CVPR ’05

Hough Transform, Harish Corner Detector & SIFT

Intro.

- SIFT in one sentence
  - Histogram of gradients @ Harris-corner-like

SIFT

1. Scale-space extrema detection
2. Keypoint localization
3. Orientation assignment
4. Keypoint descriptor

Hough Transform, Harish Corner Detector & SIFT

Detection

Extracting Keypoints

- Extrema detection
- Correct localization
- Choosing robust keypoints only

1st part of the talk

Description

Distinctive Description

- Local invariant orientation
- Building keypoint descriptor

2nd part of the talk

Application

Matching Keypoints

- Nearest-neighbor algorithm
- Finding 3 matches
- Least-square affine approximation

Last part of the talk

Z Z Z z z

Finding Keypoints – Scale, Location

- Scale selection principle (T. Lindeberg ’94)

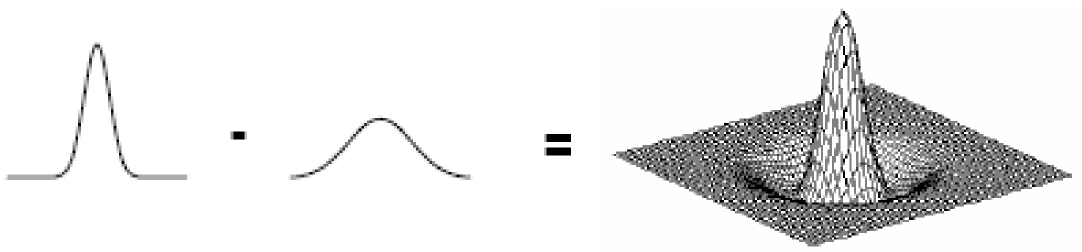
  - In the absence of other evidence, assume that a scale level, at which (possibly non-linear) combination of normalized derivatives assumes a local maximum over scales, can be treated as reflecting a characteristic length of a corresponding structure in the data.

➔ Maxima/minima of Difference of Gaussian

Hough Transform, Harish Corner Detector & SIFT

- Difference-of-Gaussian (DoG):

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y)$$
$$= L(x,y,k\sigma) - L(x,y,\sigma)$$



Low computation time -

Only subtraction of smoothed images!

Detection of Scale-space Extrema

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

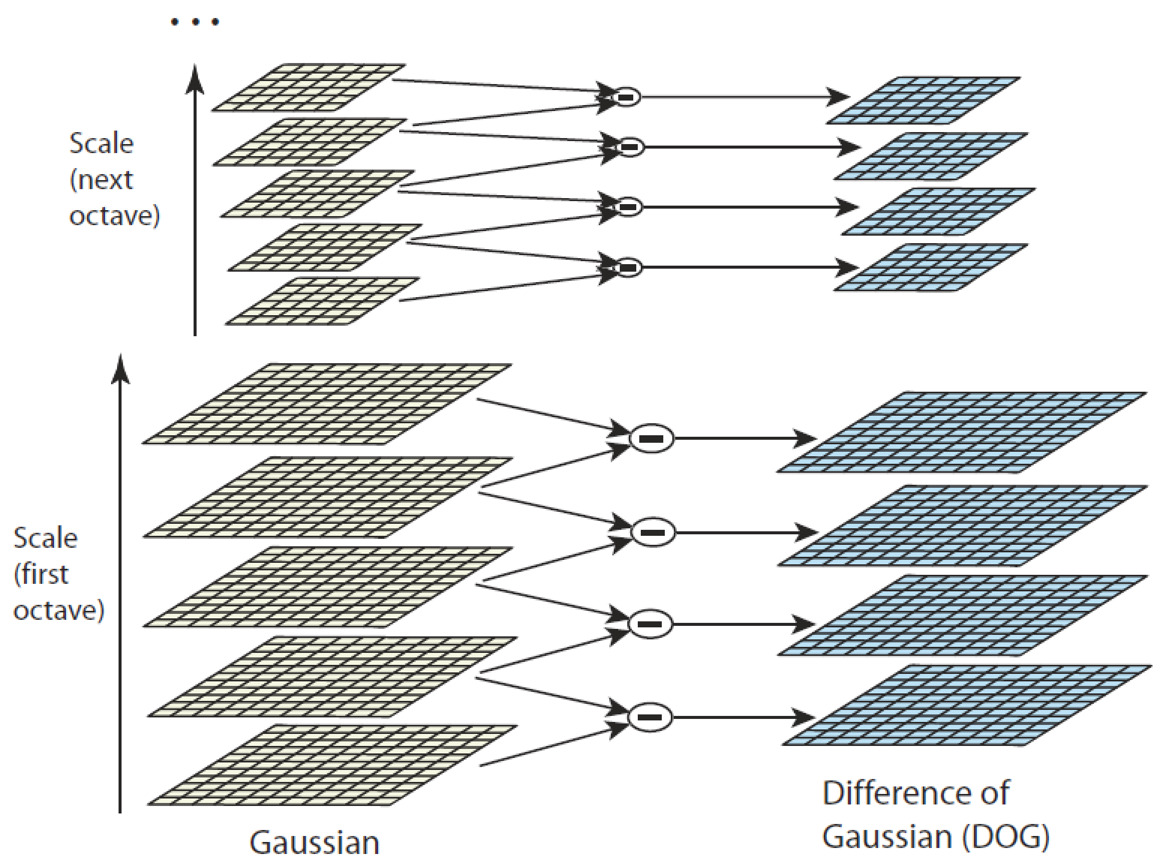
$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned}$$

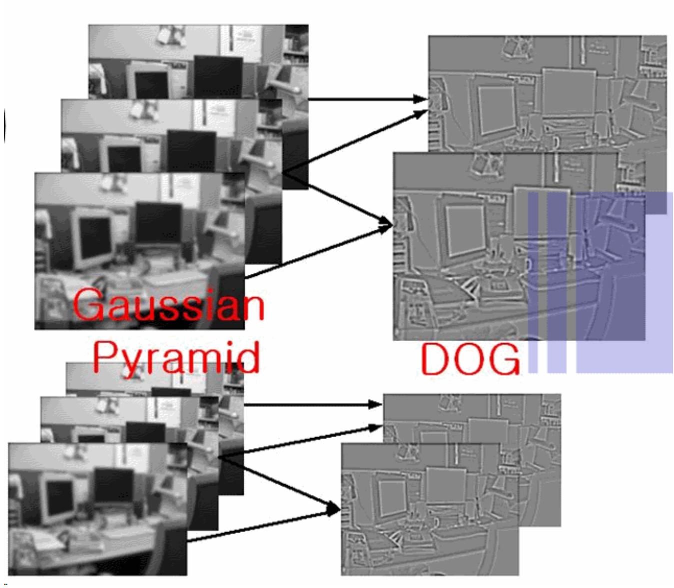
$$\frac{\partial G}{\partial \sigma} = \sigma \nabla^2 G$$

$$\sigma \nabla^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma}$$

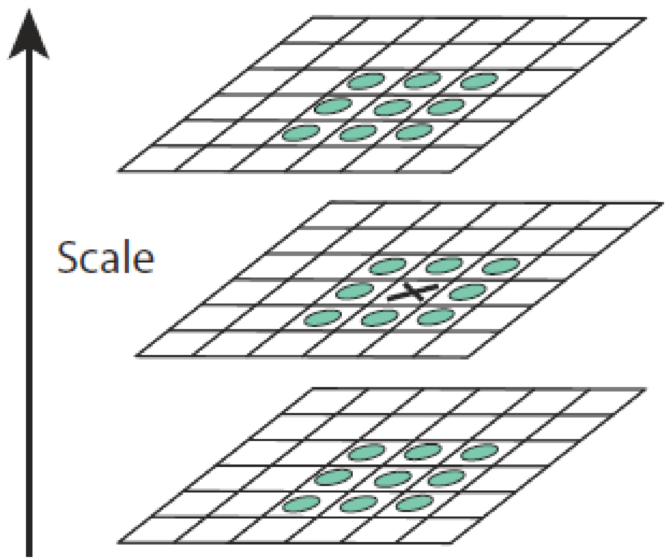
$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\sigma^2 \nabla^2 G$$


. . .
Scale
(next
octave)
Scale
(first
octave)
Gaussian
Difference of
Gaussian (DOG)

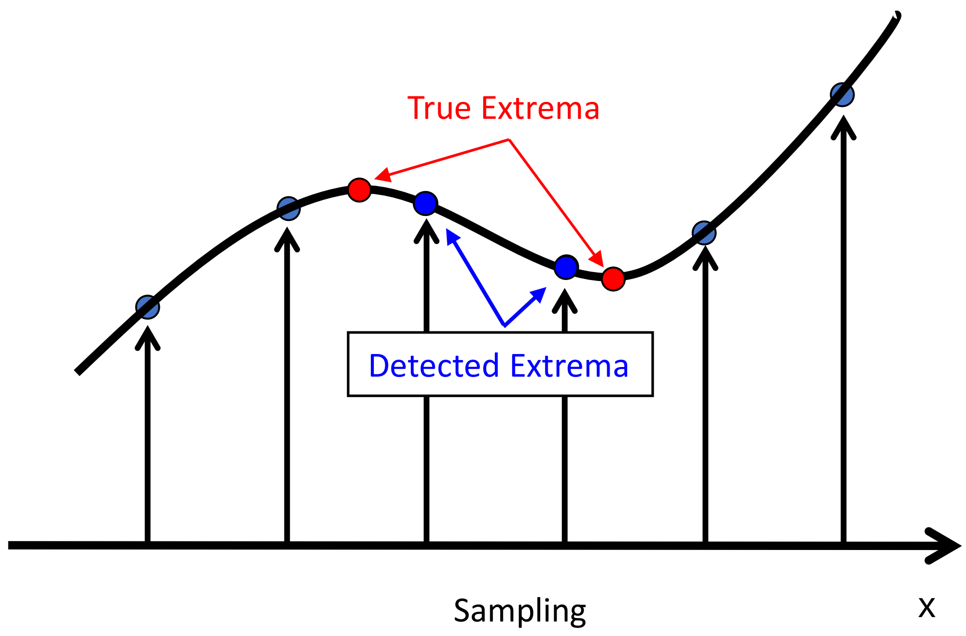
Differences Of Gaussians



Local Extrema Detection



- The Problem:



Keypoint Localization

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}} \mathbf{x} + \frac{1}{2} \mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2} \mathbf{x}$$

$$D(\hat{\mathbf{x}}) = D + \frac{1}{2} \frac{\partial D^T}{\partial \mathbf{x}} \hat{\mathbf{x}}.$$

$$\mathbf{H} = \left[\begin{array}{cc} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{array} \right]$$

$$\mathrm{Tr}(\mathbf{H}) = D_{xx} + D_{yy} = \alpha + \beta,$$

$$\mathrm{Det}(\mathbf{H}) = D_{xx} D_{yy} - (D_{xy})^2 = \alpha\beta.$$

$$\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r+1)^2}{r},$$

$$\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} < \frac{(r+1)^2}{r}.$$

Finding "Cornerness"

- Principal curvature are proportional to eigenvalues $\lambda_{\max}, \lambda_{\min}$ of Hessian matrix:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

- Harris (1988) showed:

$$\frac{\lambda_{\max}}{\lambda_{\min}} < r \Leftrightarrow \frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}$$

- **Threshold:** if $r < 10$ - ratio is too great, keypoint discarded.

Orientation Assignment

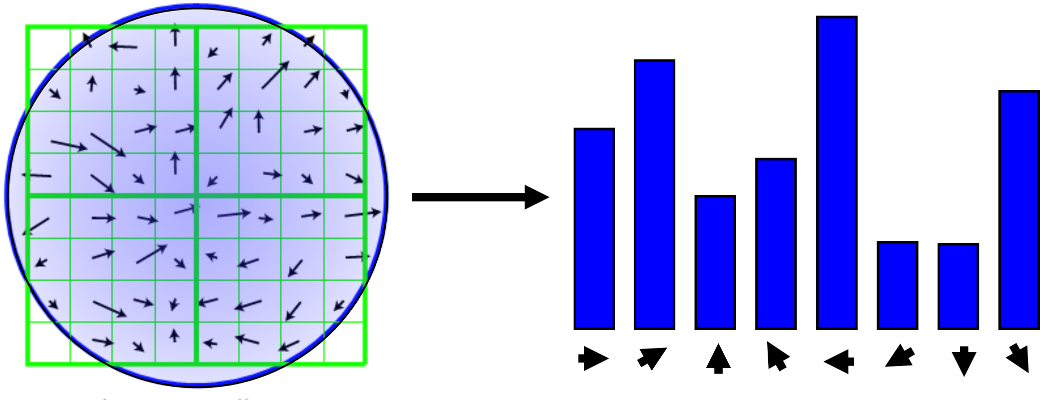
$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$

$$\theta(x,y) = \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y)))$$

- An orientation histogram is formed from the gradient orientations of sample points within a region around the keypoint.
- The orientation histogram has 36 bins covering the 360 degree range of orientations.
- Each sample added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with a that is 1.5 times that of the scale of the keypoint.
- Peaks in the orientation histogram correspond to dominant directions of local gradients.

Keypoints Orientation

- Create gradient histogram (36 bins) weighted by magnitude and Gaussian window (σ is 1.5 times that of the scale of a keypoint)



- Any histogram peak within 80% of highest peak is assigned to keypoint (multiple assignments possible).

The Local Image Descriptor

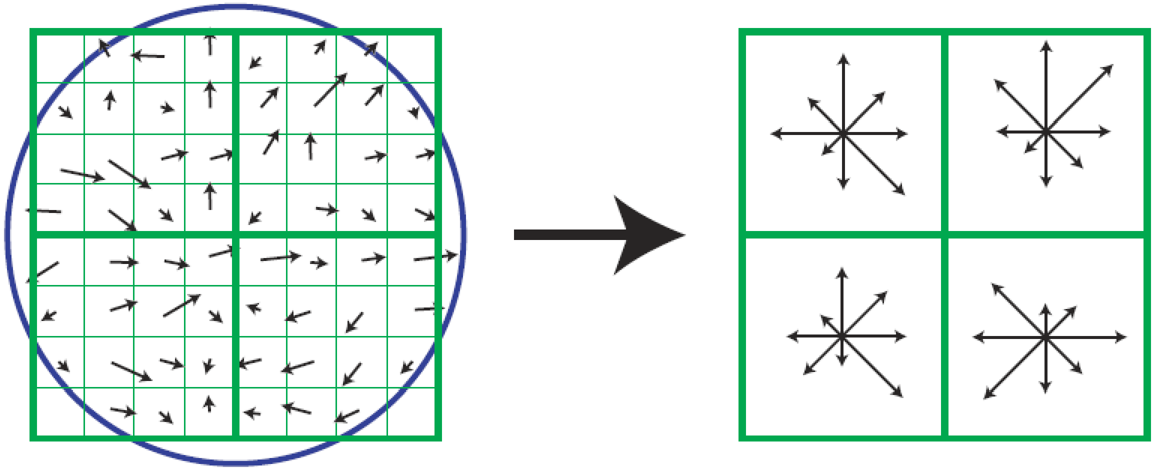


Image gradients

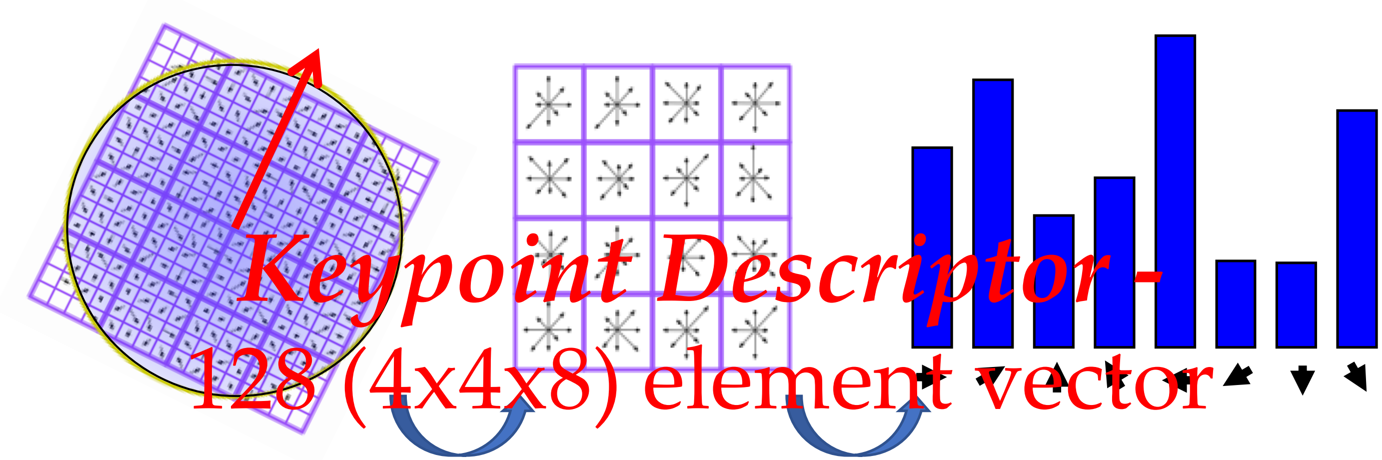
Keypoint descriptor

This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array

Keypoint Descriptor Length: 4x4x8 = 128

Hough Transform, Harish Corner Detector & SIFT

- Create 16 gradient histograms (8 bins) weighted by magnitude and Gaussian window (σ is 0.5 times of the window)



Keypoint Descriptor - 128 (4x4x8) element vector

Hough Transform, Harish Corner Detector & SIFT

Hough Transform, Harish Corner Detector & SIFT