Hough Transform: Grouping of local elements to global structures

P.V.C. Hough 1962 Patent

Duda & Hart 1972 "Detection of collinear points"

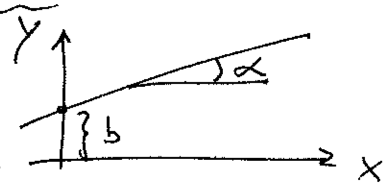
Ballard D.H. 1981 "Generalizing the HT"

Gerig G. 1986/87 "Linking feature space & accumulator space"

① Parametrization of straight line:

- $y = a \cdot x + b$

  $a = \tan \alpha$

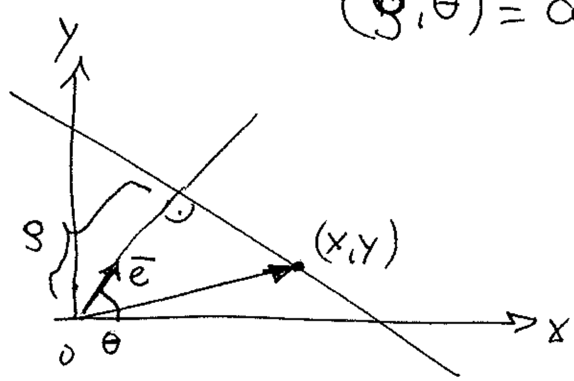


- normal form:

$$\boxed{\rho = x \cdot \cos\theta + y \sin\theta}$$

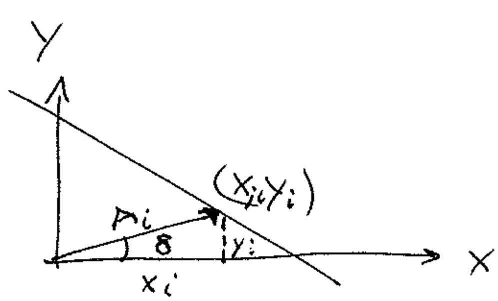
$(x, y) = \bar{x}$: point coordinate

$(\rho, \theta) = \bar{a}$: parameter vector



$$\rho = \begin{pmatrix} x \\ y \end{pmatrix} \cdot \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$$

projection of $\bar{x}$ onto direction of normal $\bar{e}$



$$\cos\delta = \frac{x_i}{A_i} \qquad A_i = \sqrt{x_i^2 + y_i^2}$$

$$\sin\delta = \frac{y_i}{A_i} \qquad \delta = \tan^{-1}\left(\frac{y_i}{x_i}\right)$$

Given point $(x_i, y_i) \Rightarrow (\rho, \theta)$ are parameters

$$\rho = x_i \cos\theta + y_i \sin\theta$$

$$\frac{\rho}{A_i} = \underbrace{\frac{x_i}{A_i}}\cos\theta + \underbrace{\frac{y_i}{A_i}}\sin\theta_i$$

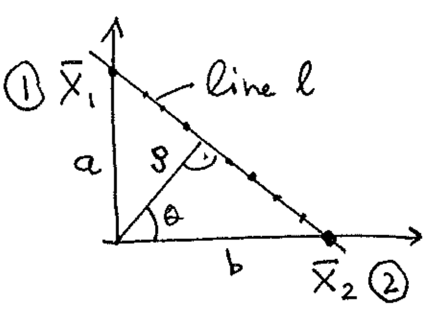
$$\frac{\rho}{A_i} = \underbrace{\cos\delta_i\cos\theta + \sin\delta_i\sin\theta}_{\cos(\theta-\delta_i)}$$

$$\Rightarrow \boxed{\rho = A_i\cos(\theta - \delta_i)} \qquad A_i = \sqrt{x_i^2 + y_i^2}, \quad \delta_i = \tan^{-1}\left(\frac{y_i}{x_i}\right)$$

Given $(x_i, y_i) \rightarrow (A_i, \delta_i) \Rightarrow$ Cos-function in ρ, θ space

Example:

Image Space



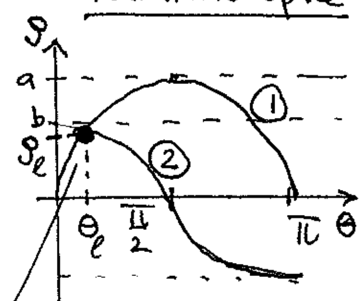
1 $\bar{X}_1$: $\rho_1 = a\cos(\theta - \frac{\pi}{2})$

2 $\bar{X}_2$: $\rho_2 = b\cos(\theta - 0)$

$\vdots$

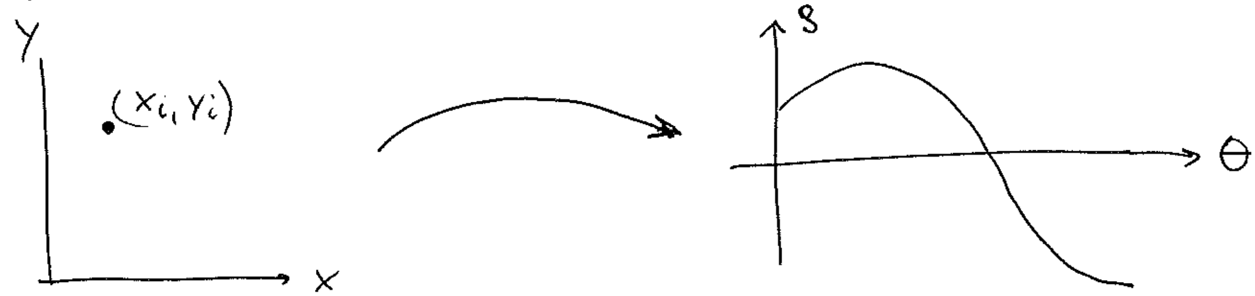
$\bar{X}_n$

Parameter Space

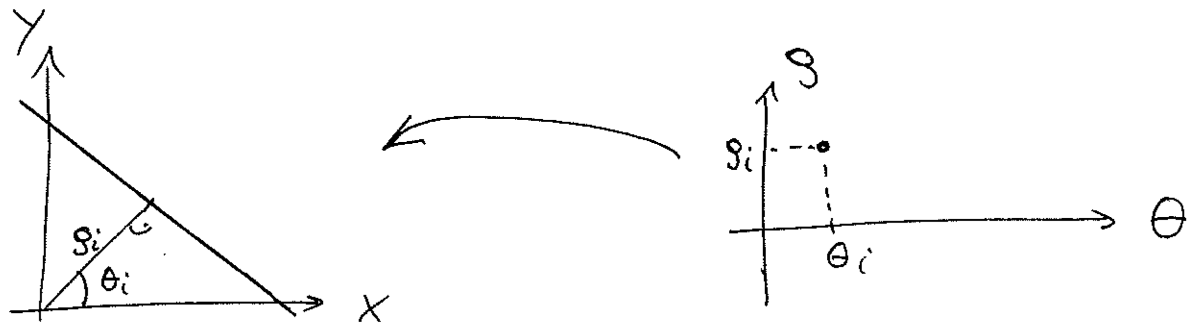


intersection:
(ρ_ℓ, θ_ℓ) = parametrization of line through $\bar{x}_1$ and $\bar{x}_2$

Point to Curve Transformation



$$\rho = A_i \cos(\theta - \delta_i)$$

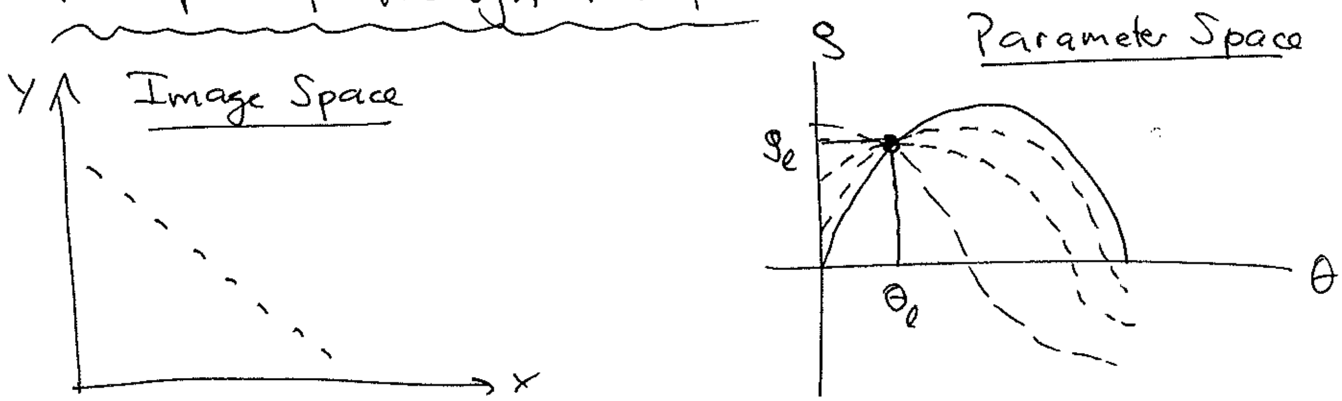


(ρ_i, θ_i)

$$\rho_i = x \cos\theta_i + y \sin\theta_i$$

all (x,y) form straight line

1.1 Principle of Hough Transform



collinear points in image space form set of cosine curves in parameter space with common intersection
→ intersection defines line parametrization

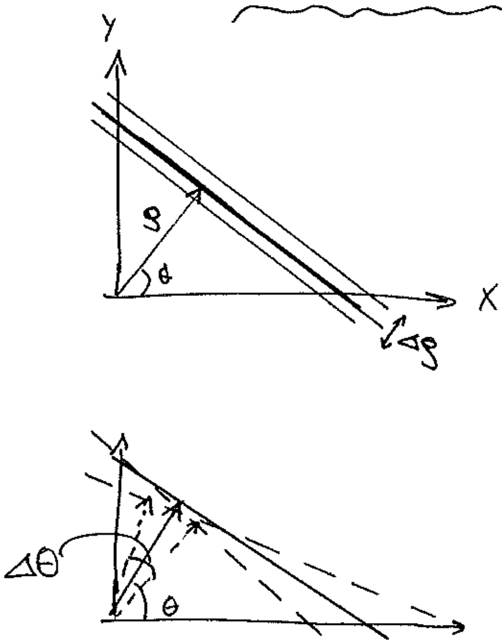
1.2 Numerical implementation

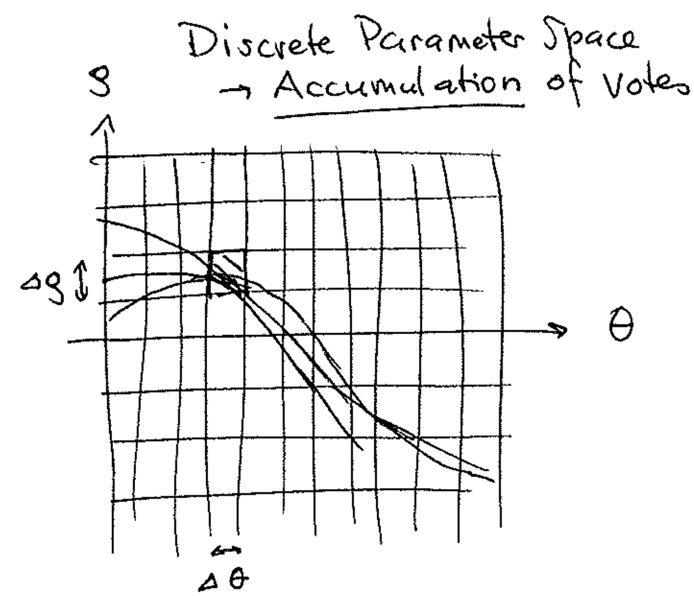
a) Intersection of curves:

- n points $(\vec{x}_i) \Rightarrow n$ cos-curves
- pairwise intersections: $\frac{n(n-1)}{2}$ intersection tests
- find intersection locations in parameter space with high density

$\Rightarrow$ not preferable: - # intersections $\propto n^2 \propto O_n^2$
- no notion of noise and subtle geometric variation

b) Discretization of parameter space





- cos-curves in parameter space intersect $(\Delta\rho, \Delta\theta)$ cells
- cells $(\Delta\rho, \Delta\theta)$ collect # intersecting curves
- each cell can be incremented by curves
  → register # collinear points

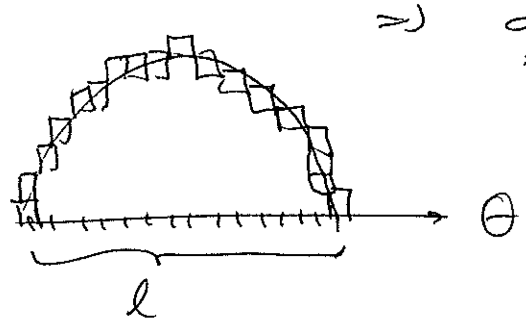
Computational expense:

- Θ divided into l intervals (e.g. 1°)
- ϱ divided into k intervals (e.g. 1 pixel)

$\Rightarrow$ $\varrho = A\cos(\Theta - \delta)$: l elements

n image points : $n \cdot l$ cell increments

$\Rightarrow \propto O_n$



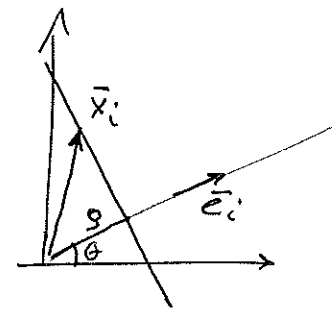
- search for maxima in parameter space:

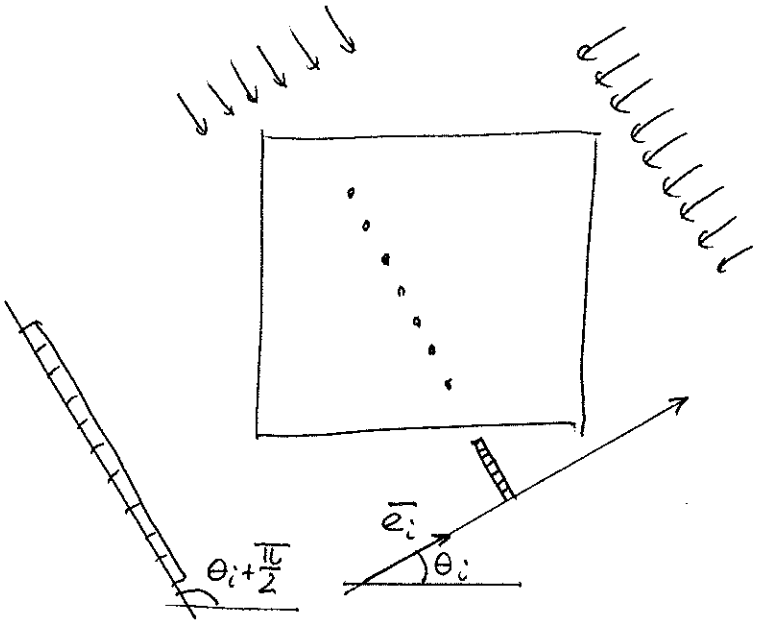
  $l \cdot k$ cells to be visited

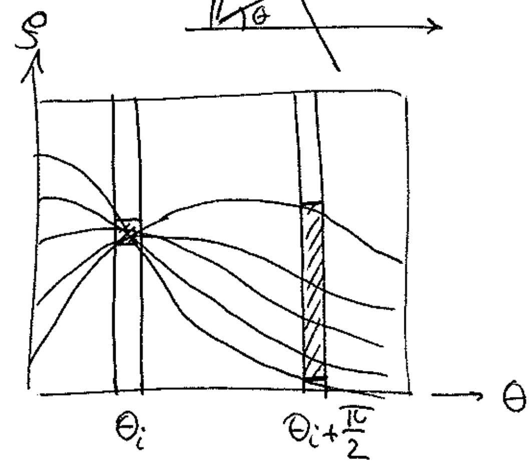
  to find set of cells with maximum density

1.3 Alternative View of Hough Transform

$$\rho = \bar{X}_i \cdot \bar{e}_i = x_i \cos\theta + y_i \sin\theta$$







\# of collinear points

location of points along straight line

- projection onto $\bar{e}_i = \begin{pmatrix} \cos\theta_i \\ \sin\theta_i \end{pmatrix}$
- summation

$\rightarrow$ Projection summation

$\Rightarrow$ Analogy to Radon Transform

$$H(r,\theta) = \int_{-\infty}^{\infty} dx \int_{-\infty}^{\infty} dy \; I(x,y) \, \delta(\rho - x\cos\theta - y\sin\theta)$$

$\delta(0) = 1$: point $x,y \in$ line

$\delta(\;) = 0$: otherwise

Use in computer tomography: CT

1.4 Hough Transform with use of local edge orientation

<u>so far</u>: only localization of image points (x, y)

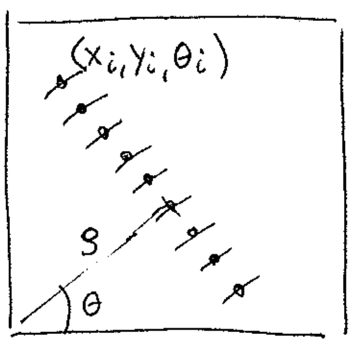
<u>new</u>: object contours also have local orientation

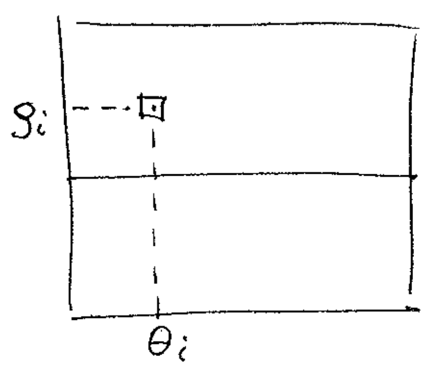
- edges: $\tan(\;) = \left(\dfrac{\frac{\partial}{\partial x}(G \otimes I)}{\frac{\partial}{\partial y}(G \otimes I)} \right)$ Canny

- lines: Hessian: $\begin{pmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{pmatrix} \longrightarrow \begin{pmatrix} \lambda_1 & \\ & \lambda_2 \end{pmatrix}$

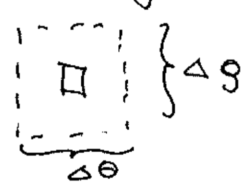
  direction of lines: $\underset{\text{eigenvalue}}{\lambda_1} \leftrightarrow \underset{\text{eigenvector}}{\begin{pmatrix} ev_{1x} \\ ev_{1y} \end{pmatrix}}$

$\Rightarrow$ direction across edges (gradient direction) and lines (direction of maximum 2nd derivative) : correspond to θ ?





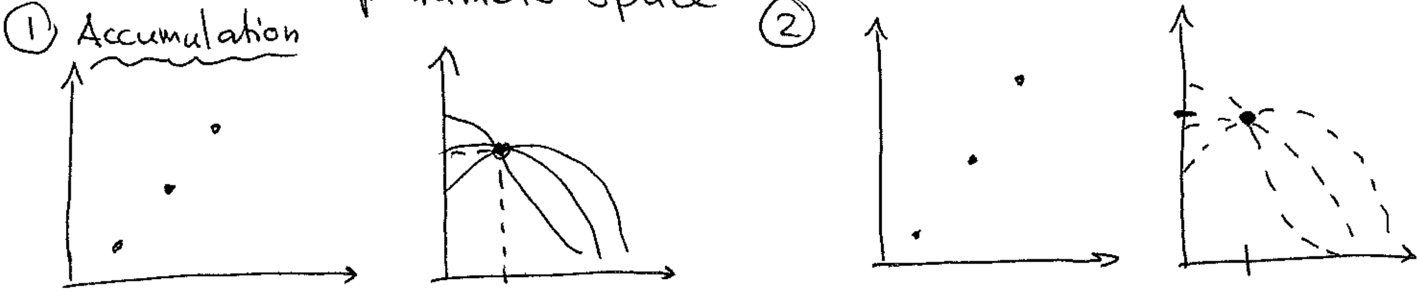
- $\varrho_i = x_i \cos\theta_i + y_i \sin\theta_i$ | θ_i: gradient orientation
- ideally: incrementation of <u>only one cell</u> (ϱ_i, θ_i)
- practically: error in localization and angle due to noise: increment larger region in parameter space



1.5 Simplification of accumulated parameter space: Decrementation and Maximum Search

(Gerig et al., IJCPR, 1986)(ICCV 1987)

Idea:
- Set of collinear image points transforms into set of cosine curves in image space, but only cell of maximum density is finally what we want to find.
- Decremet parameter space in a second pass and only keep a maximum per curve.
- ⇒ Image-guided decrementation of parameter space



Algorithm: 1 – build accumulator by incrementation of cells intersected by cosine curves
- eventually smooth accumulator to reduce discretization artifacts

2 – traverse each cosine curve for each image point again, but only keep cell of maximum vote per curve
→ only cell of maximum density remains

– Additional possibility: keep list of points x_i per remaining parameter cell ⇒ we know location of points forming a straight line.