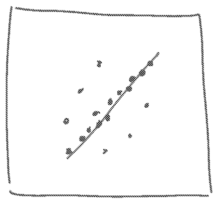
Line detection

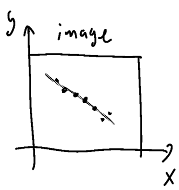
Tasks:

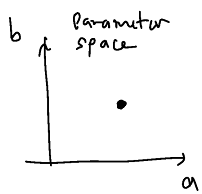
1) Grouping
2) Fitting



* Two simultaneous tasks. solving one makes the other easier

Hough transform



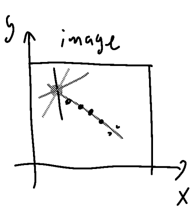


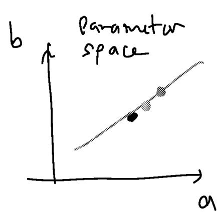
$y = ax + b$

* multiple points on a line in the image correspond to a single point in parameter space (the parameters of the line)

* Detect lines by casting votes in parameter space

Voting in parameter space





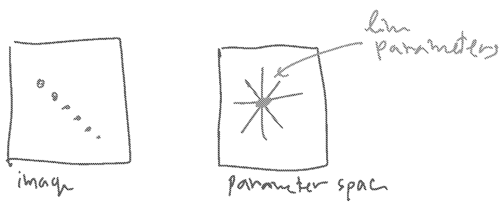
* A single point in the image defines a line in parameter space (describing the parameters of all possible lines through the point)

$y = ax + b \Rightarrow b = y - ax$

given (x,y) scan a and compute b.
cast votes at locations (a,b) - a line

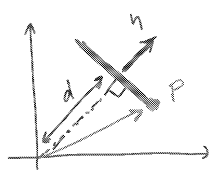
Voting in parameter space

- Each point in the image defines a line in parameter space
- Multiple points define multiple lines in parameter space
- Intersection of lines in parameter space indicate parameters common to multiple points on the same line



Better line representation

* Problem with $y = ax + b$ model:
  - what is the possible range of a?
  - How to represent vertical lines



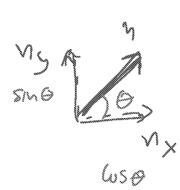
$n = (n_x, n_y)$ normal (orientation)

d = distance from origin

* For point $P = (x, y)$ to be on the line:
  $(n_x, n_y) \cdot (x, y) = d$

Implicit line equation

$n = (n_x, n_y)$ $\begin{cases} n_x = \cos\theta \\ n_y = \sin\theta \end{cases}$



* Implicit line equation:

$$(n_x, n_y) \cdot (x, y) = d \Rightarrow (\cos\theta)x + (\sin\theta)y - d = 0$$

$$(Ax + By + C = 0)$$

* line parameters: θ, d

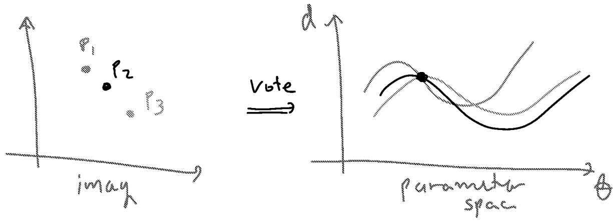
Hough with explicit line equation

* Explicit line equation:

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

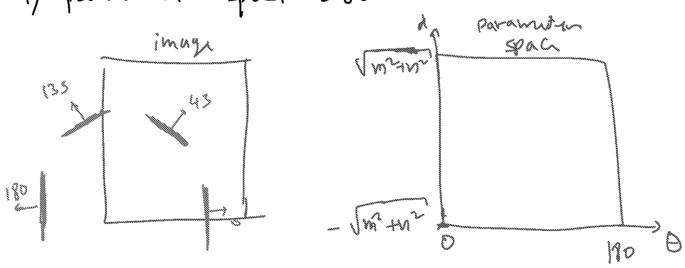
* Vote by point (x,y):

  for each θ_i: vote $d_i = x \cos\theta_i + y \sin\theta_i$



Practical issues

1) parameter space size:



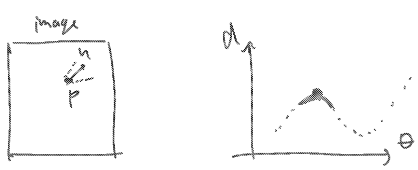
2) Bin size: larger bins are more efficient but provide less localization

3) Peak detection: threshold and suppress close peaks

Hough with edge elements

* Estimate a normal vector at each pixel

  ⇒ each pixel provides a vote which is a small curve section (accounting for in accurate normal direction)



* A point (x,y) with normal θ votes for

  $d = x \cos(\theta + \alpha \Delta\theta) + y \sin(\theta + \alpha \Delta\theta)$

  where $\alpha \in [-1, 1]$

Generalized Hough transform

* objective: $f(\underbrace{x,y}_{\text{point}}, \underbrace{a_1, \ldots, a_m}_{\text{parameters}}) = 0$

* For each (x,y) scan $a_1, \ldots, a_{m-1}$ and vote for a_m

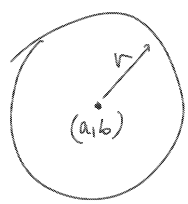
* Problems:
  - sparse space with increased dimensions
  - voting becomes inefficient with increased dimensions

Example - Hough transform for circles

* circle equation:

$$(x-a)^2 + (y-b)^2 = r^2$$

parameters: a, b, r



or

$$\begin{cases} x = a + r\cos\theta \\ y = b + r\sin\theta \end{cases}$$

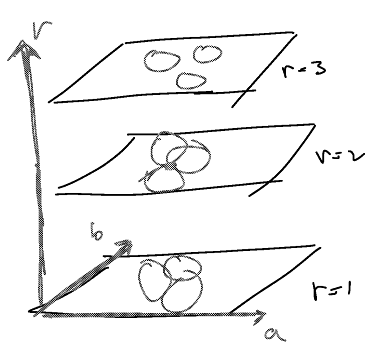
* Vote:

Given (x,y) vote in each r plane using:

$$a = x - r\cos\theta \qquad \text{where } \theta \in [0, 360]$$
$$b = y - r\sin\theta$$

Circle Hough vote

each point (x,y) casts a circle vote in each r plane



Model fitting

* - Given data $\{(x_i, y_i)\}_{i=1}^{m}$ and model $M_p(x_i)$:
  - define objective: $E(p) = \sum_i (M_p(x_i) - y_i)^2$
  - Minimize objective: $p^* = \underset{p}{\arg\min}\ E(p)$

Line fitting

* Given a group of points (e.g. neighboring a line detected using Hough) find a more accurate model.
* Given $\{(x_i, y_i)\}_{i=1}^{n}$ find the parameters of the line $y = ax + b$

$$\begin{cases} E(a,b) = \sum_{i=1}^{n} \left(y_i - (ax_i + b) \right)^2 \\ a^*, b^* = \underset{a,b}{\arg\min}\ E(a,b) \end{cases}$$

$$\Rightarrow \nabla E(a,b) = 0 \Rightarrow a^*, b^*$$

Line fitting

$$E(a,b) = \sum_{i=1}^{n} \left(y_i - (ax_i + b) \right)^2$$

$$\nabla E(a,b) = \begin{bmatrix} \frac{\partial E}{\partial a} \\ \frac{\partial E}{\partial b} \end{bmatrix} = \begin{bmatrix} \sum 2(y_i - ax_i - b)(-x_i) \\ \sum 2(y_i - ax_i - b)(-1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} \sum \cancel{2} x_i^2 & \sum \cancel{2} x_i \\ \sum \cancel{2} x_i & \sum \cancel{2} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} a \\ b \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} \sum \cancel{2} y_i x_i \\ \sum \cancel{2} y_i \end{bmatrix}}_{b}$$

$$\Rightarrow x = A^{-1} b \quad \text{where} \quad A = \begin{bmatrix} \sum x_i^2 & \sum x_i \\ \sum x_i & m \end{bmatrix} \quad x = \begin{bmatrix} a \\ b \end{bmatrix} \quad b = \begin{bmatrix} \sum x_i y_i \\ \sum y_i \end{bmatrix}$$