KECE471 Computer Vision

Segmentation by Fitting a Model

Chang-Su Kim

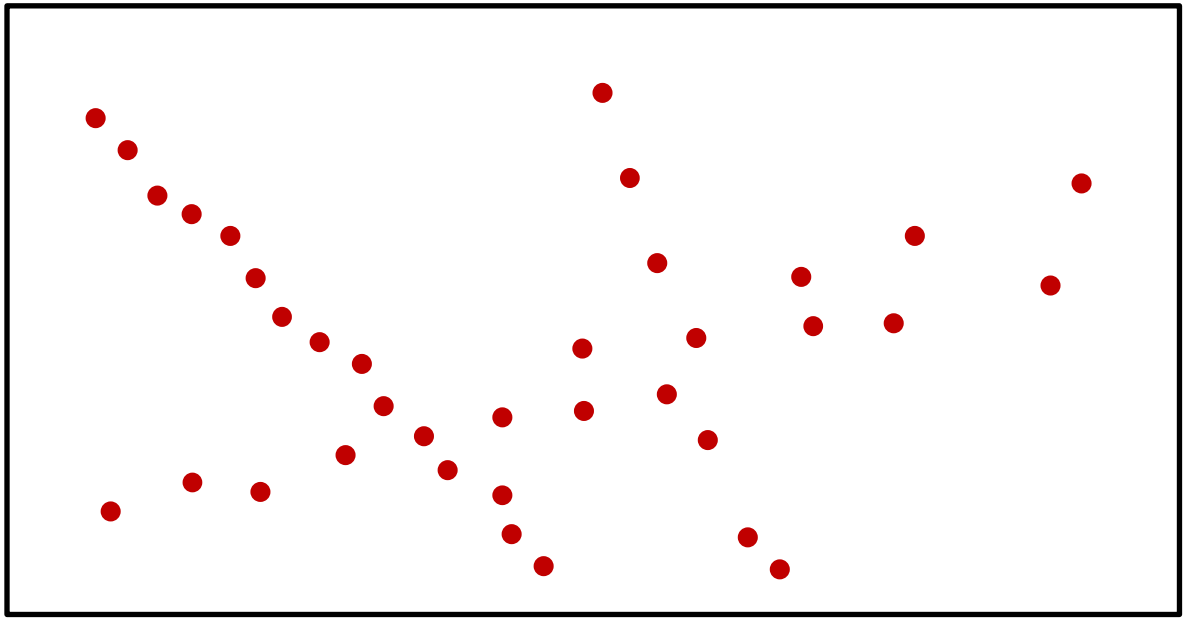
Chapter 15, Computer Vision by Forsyth and Ponce
Note: Dr. Forsyth's notes are partly used.
Jun-Sung Kim in Korea University made the first draft of these slides

Fitting

- Choose an object (or model) to represent a set of tokens
  - Objects : line, circle, ellipse, and etc
  - e.g. Find a line that best describes a set of points

- Three main questions
  - What object represents this set of tokens best?
  - Which objects are associated with which tokens?
  - How many objects are there?

Line Fitting



- Three main questions
  - What line represents this set of points best?
  - Which lines gets which tokens?
  - How many lines are there?

Hough Transform for Line Fitting

Hough Transform for Line Fitting

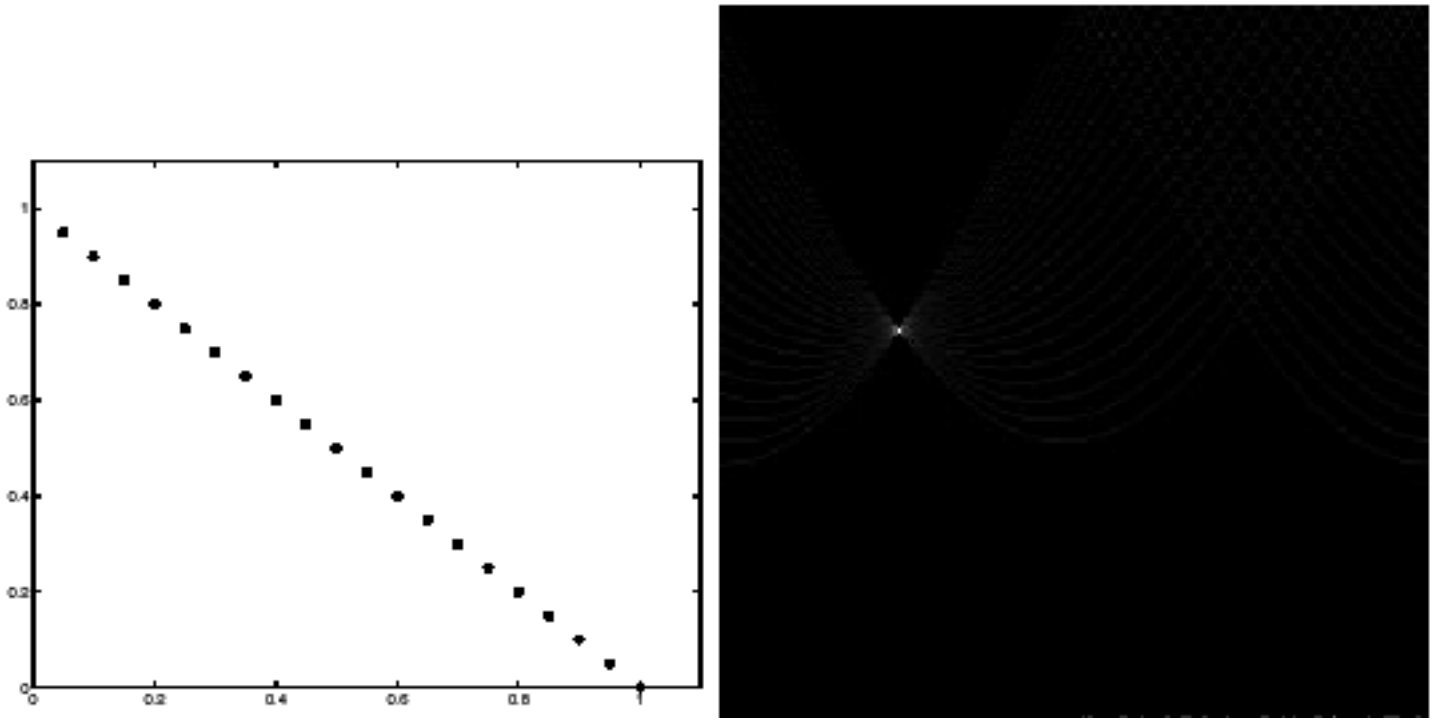
- Hough Transform
  - It may answer all three questions
  - A line is the set of point (x, y) such that

$$x \cos\theta + y \sin\theta - r = 0$$

1. For a point (x_0, y_0), there is a family of lines through the point
   - Different choices of θ give different lines
2. Each point casts a vote for each line in the corresponding family
3. If there is a line that has many votes, that should be the line passing through many points

Hough Transform for Line Fitting

- Example : The Hough transform array
  - form a line

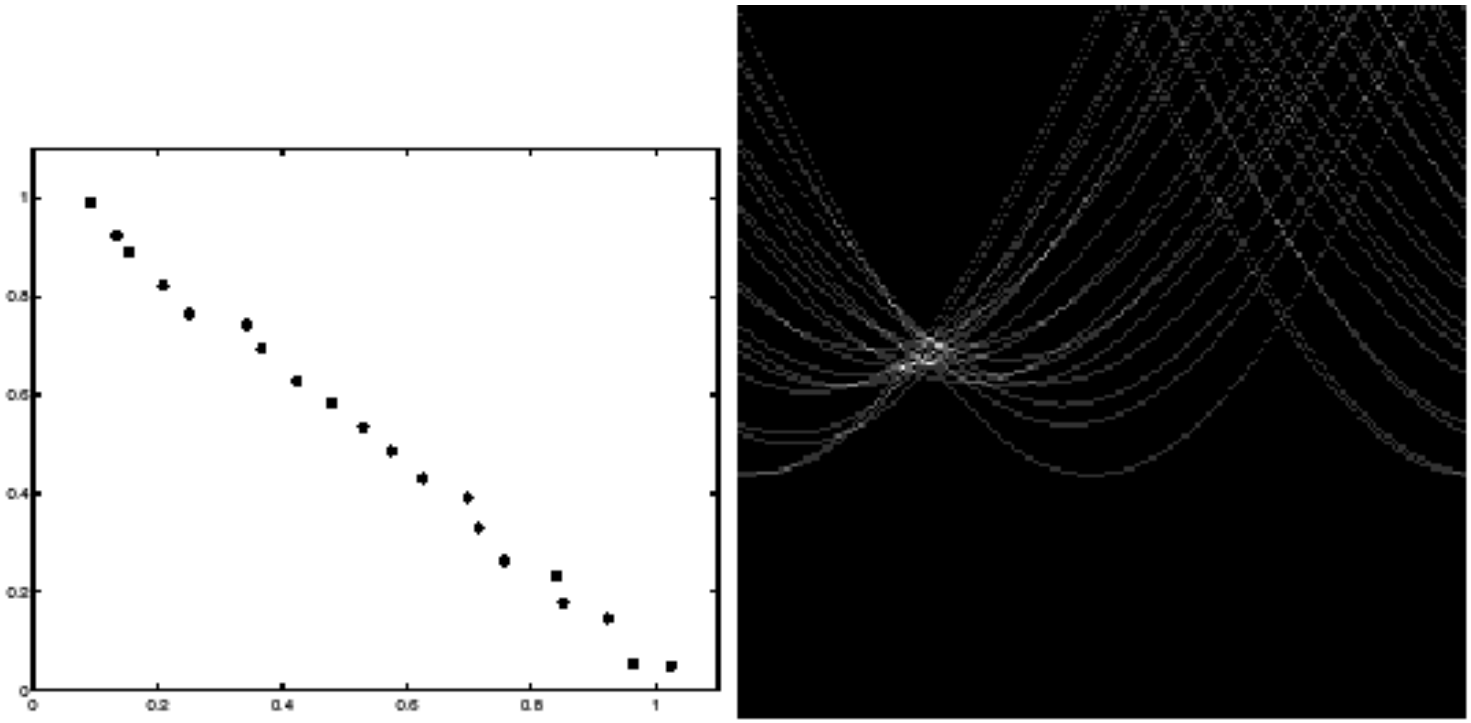


Hough Transform for Line Fitting

- Mechanics of the Hough transform
  - Construct an array representing θ, r
  - For each point, render the curve (θ, r) into this array, casting one vote to each cell
  - Difficulties
    - How big should the cells be?
      - If too big, we cannot distinguish between quite different lines
      - If too small, noise causes lines to be missed
  - How many lines?
    - Count the peaks in the array
  - Which points belong to which lines?
    - Tag the votes

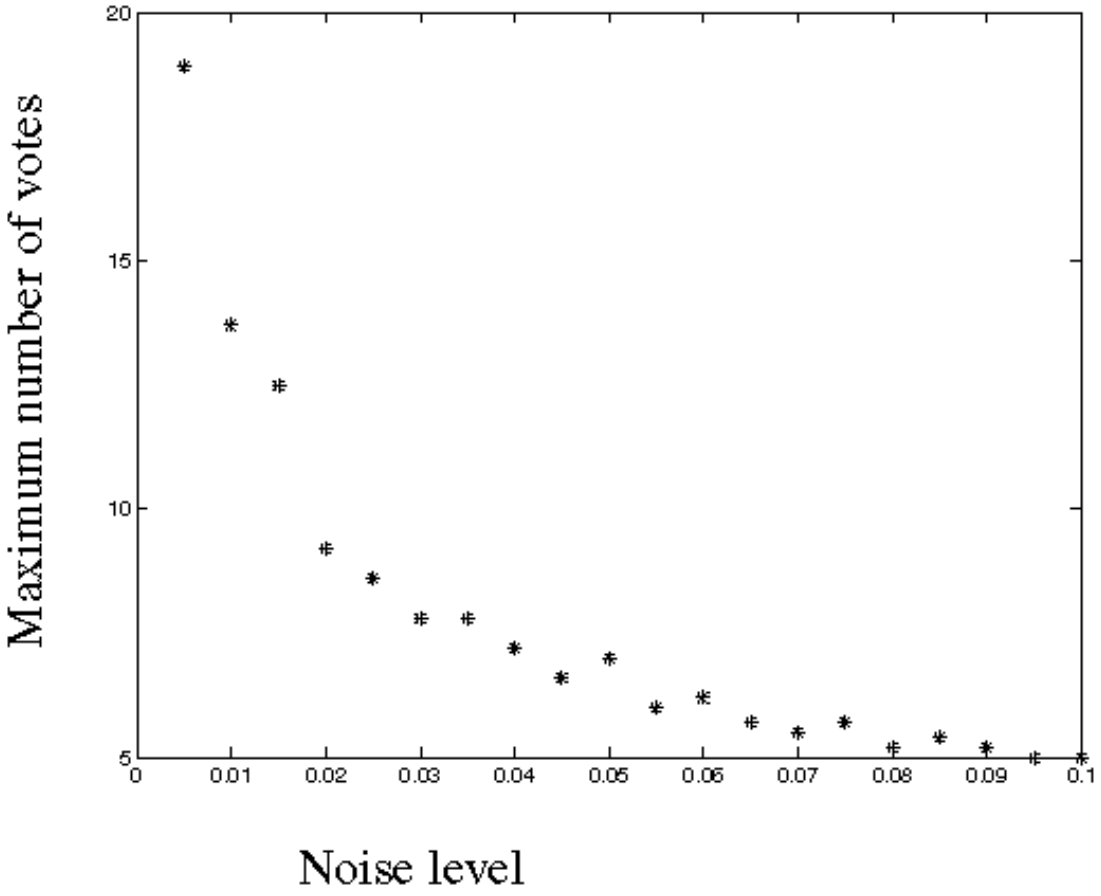
Hough Transform for Line Fitting

- Example : The Hough transform array
  - for a line with noises in the range [0, 0.05]



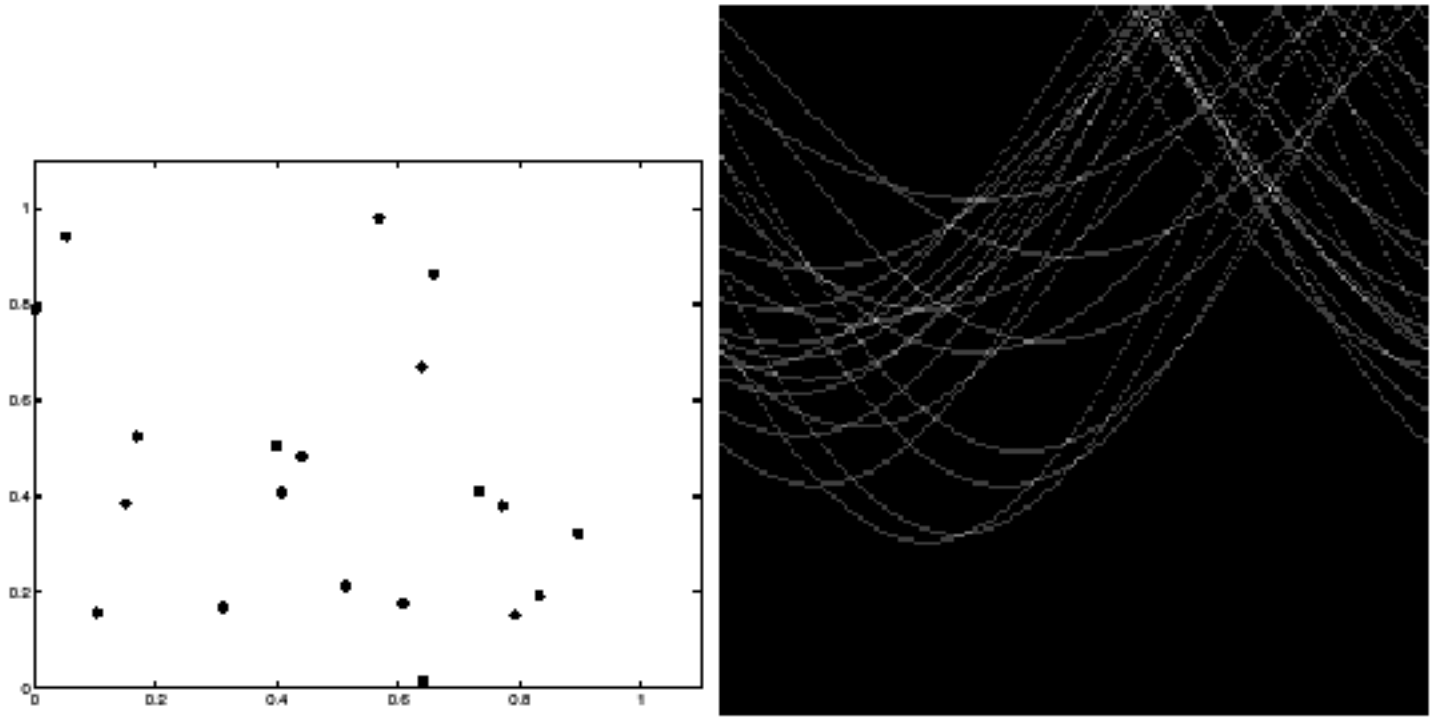
Hough Transform for Line Fitting

- # of votes with increasing noise level



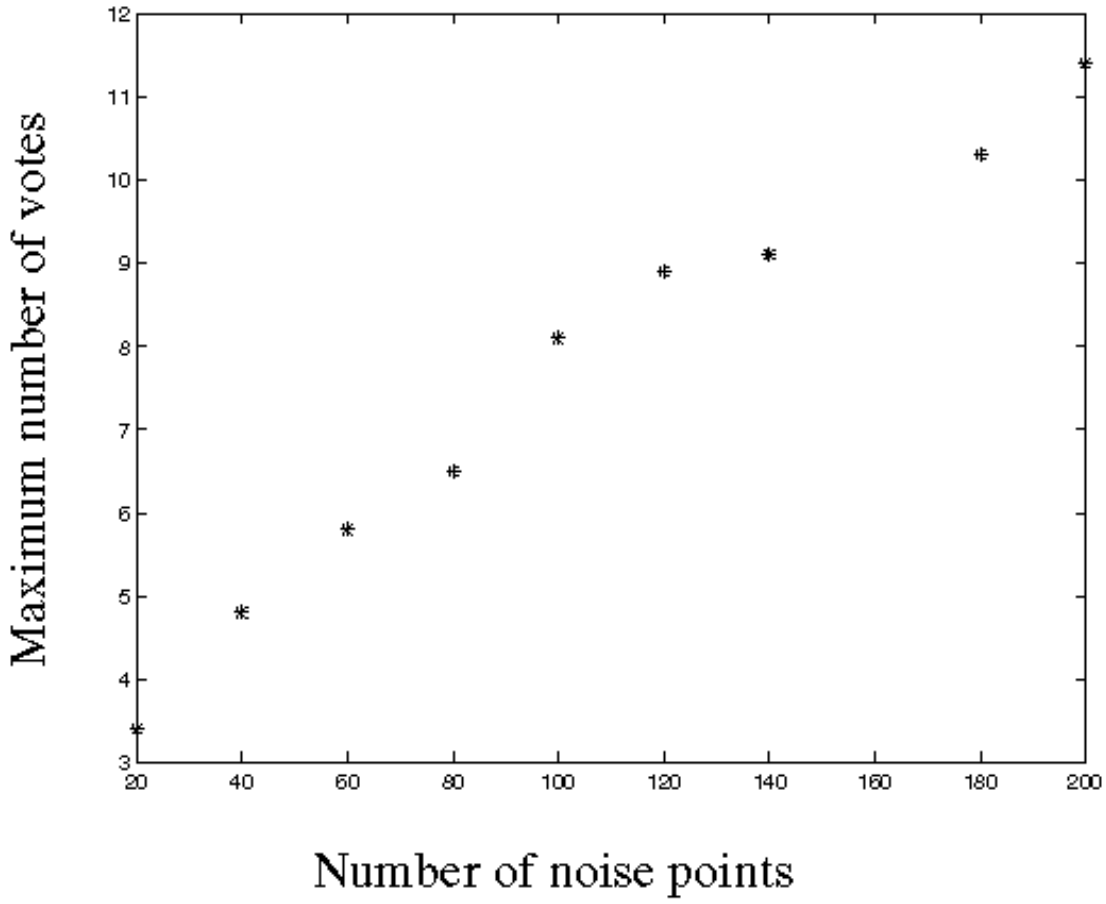
Hough Transform for Line Fitting

- Example : The Hough transform array
  - for random points



Hough Transform for Line Fitting

- # of votes with increasing random points

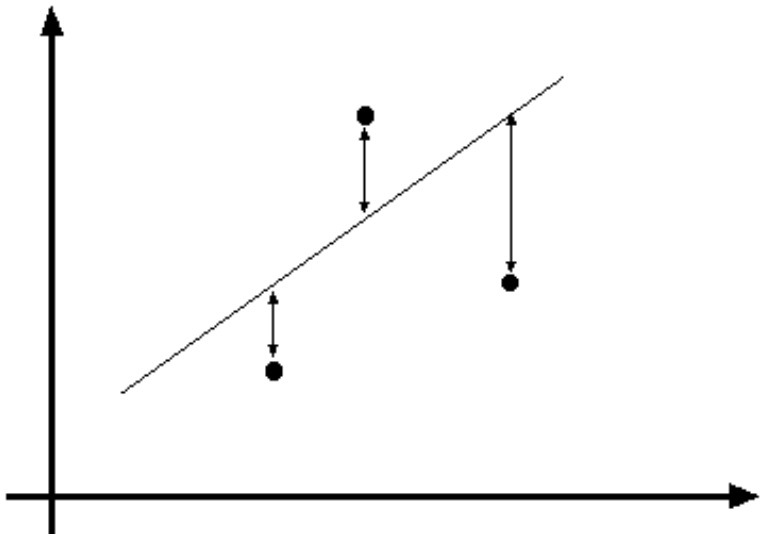


Line Fitting

Line Fitting with Least Squares

- To choose the line that minimizes

$$\sum_i (y_i - ax_i - b)^2$$



$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \overline{x^2} & \overline{x} \\ \overline{x} & 1 \end{pmatrix}^{-1} \begin{pmatrix} \overline{xy} \\ \overline{y} \end{pmatrix}$$

Raphael, 1505

Virgin in the meadow

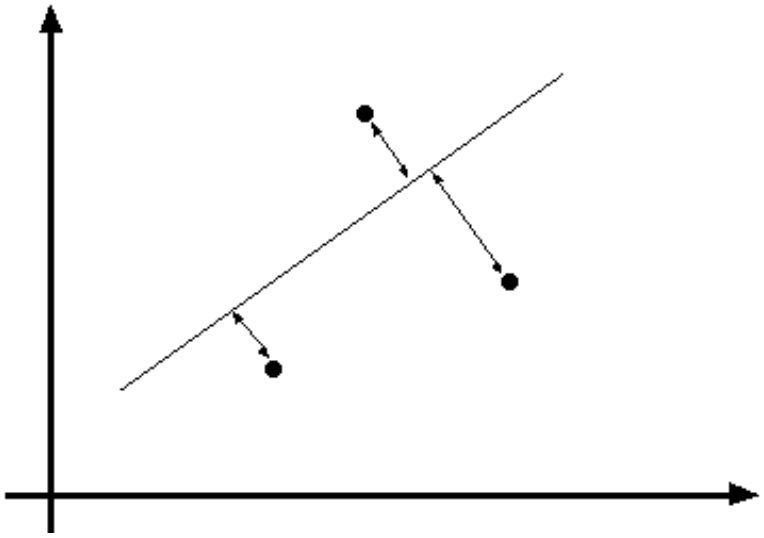
In *The History of Art*, E. H. Gombrich said

What he tried *again and again* to get was the right *balance* between the figures, the right relationship which would make the most harmonious role.

Line Fitting with Total Least Squares

- To choose the line that minimizes

$$\sum_i (ax_i + by_i + c)^2 \qquad \text{where} \quad a^2 + b^2 = 1$$



$$\begin{pmatrix} \overline{x^2} - \bar{x}\cdot\bar{x} & \overline{xy} - \bar{x}\cdot\bar{y} \\ \overline{xy} - \bar{x}\cdot\bar{y} & \overline{y^2} - \bar{y}\cdot\bar{y} \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \mu \begin{pmatrix} a \\ b \end{pmatrix}$$

$$c = -a\bar{x} - b\bar{y}$$

Which points are on which lines?

- If we know the set of points for a line, the line fitting is not difficult
- But, finding the set is difficult
  - One approach is to use Hough transform
- We learn two more strategies
  - Incremental line fitting
  - K-means

Incremental Line Fitting

Algorithm 15.1: Incremental line fitting by walking along a curve, fitting a line to runs of pixels along the curve, and breaking the curve when the residual is too large

```
Put all points on curve list, in order along the curve
Empty the line point list
Empty the line list
Until there are too few points on the curve
   Transfer first few points on the curve to the line point list
   Fit line to line point list
   While fitted line is good enough
      Transfer the next point on the curve
         to the line point list and refit the line
   end
   Transfer last point(s) back to curve
   Refit line
   Attach line to line list
end
```

K-Means Line Fitting

Algorithm 15.2: K-means line fitting by allocating points to the closest line and then refitting.

Hypothesize k lines (perhaps uniformly at random)
or
Hypothesize an assignment of lines to points
  and then fit lines using this assignment

Until convergence
  Allocate each point to the closest line
  Refit lines
end

- Note we are minimizing $\sum_i \sum_{x_j \in P_i} d^2(x_j, l_i)$

Curve Fitting

- In principle, an easy generalization from line fitting

- In practice, rather hard
  - It is generally difficult to compute the distance between a point and a curve

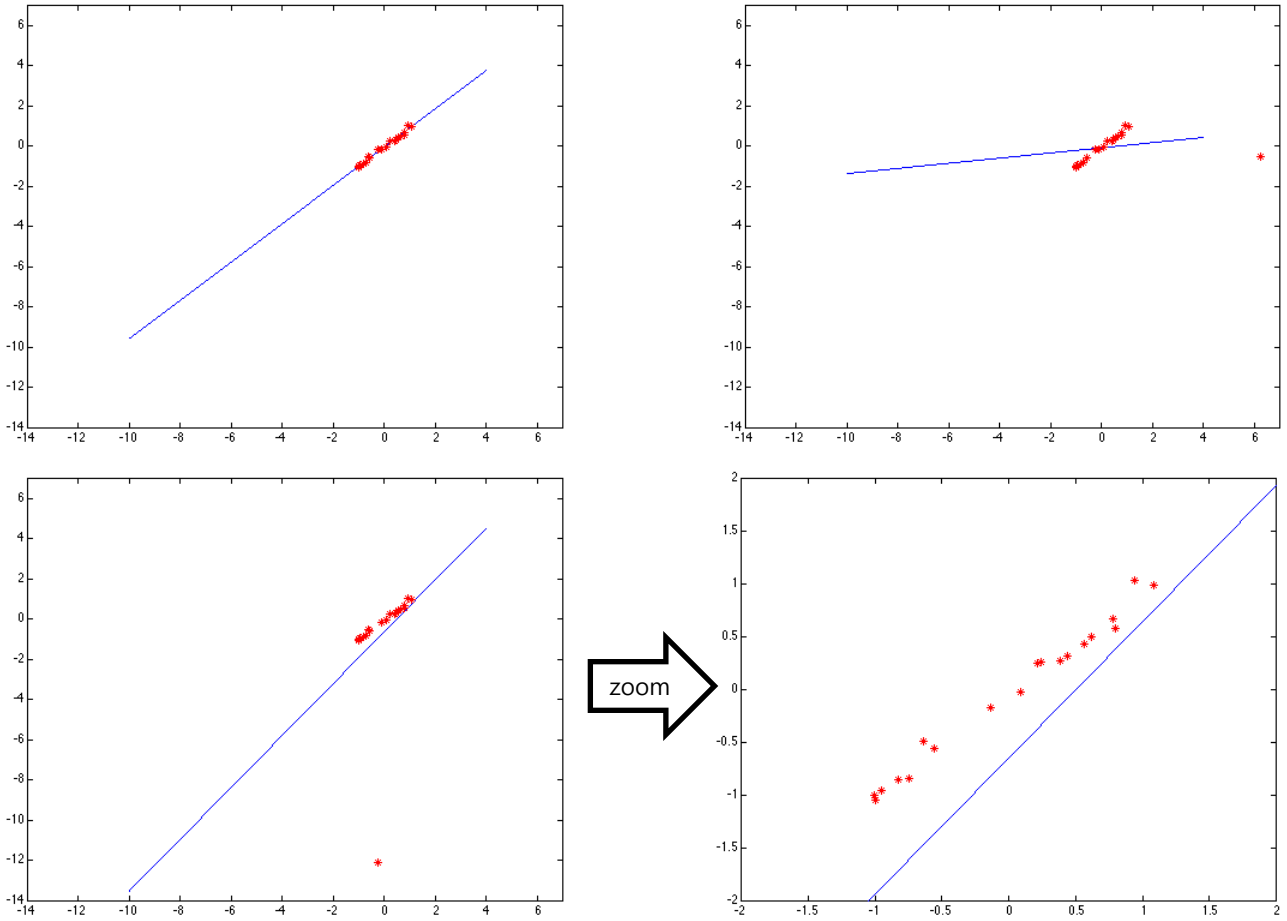
Dealing with Outliers

Robustness

- Poor fits in practice
  - Line fitting methods involve squared error terms
  - Squared errors can cause bias in the presence of noise points
- Robustness
  - M-estimators
    - Square nearby points but threshold far away points
  - RANSAC
    - Search for good points

Robustness

- Least-squares is sensitive to outliers



Robust M-Estimator

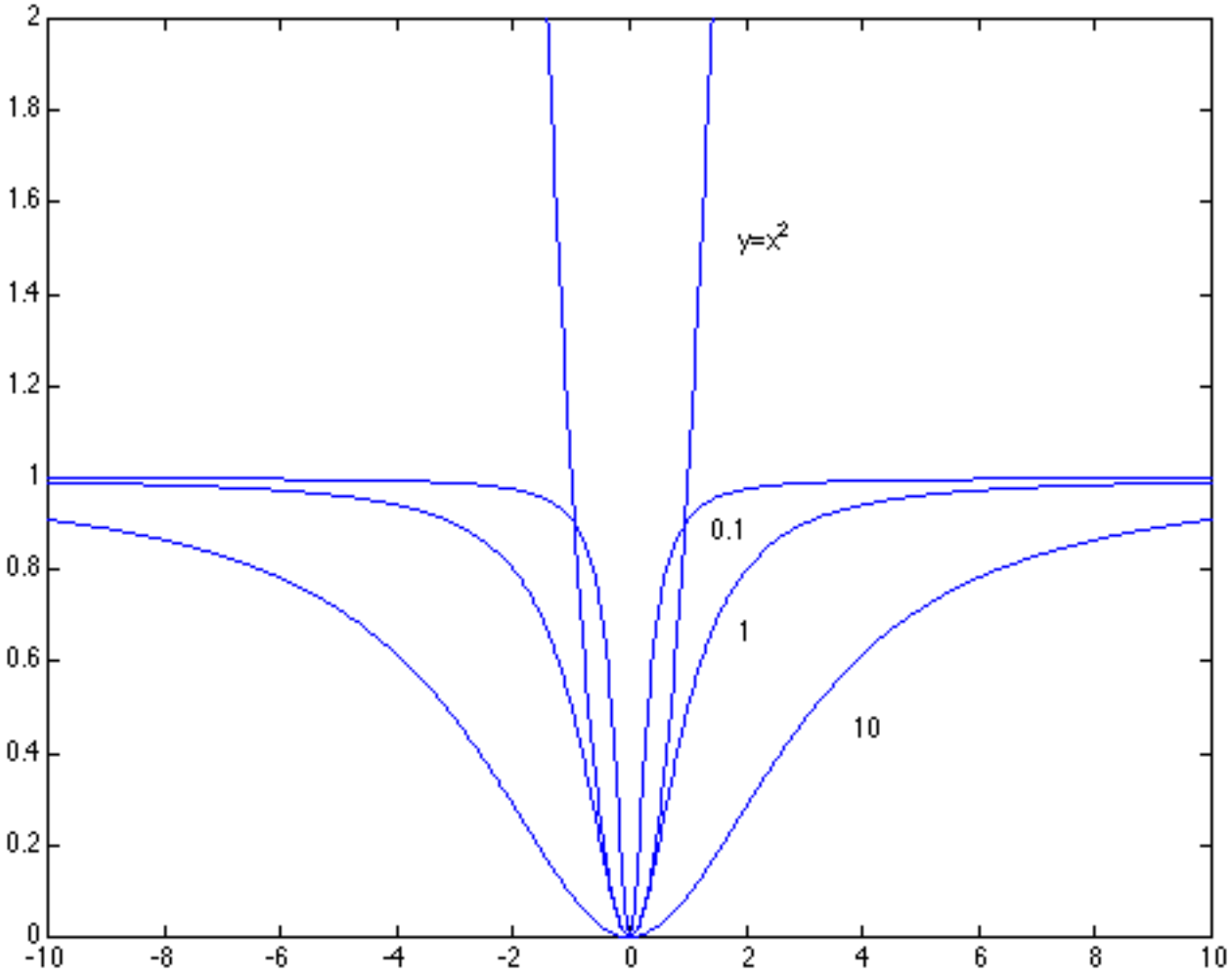
- It estimates parameters by minimizing $\sum_i \rho(d_i;\sigma)$ instead of $\sum_i d_i^2$

- Common choice

$$\rho(d_i;\sigma) = \frac{d_i^2}{\sigma^2 + d_i^2}$$

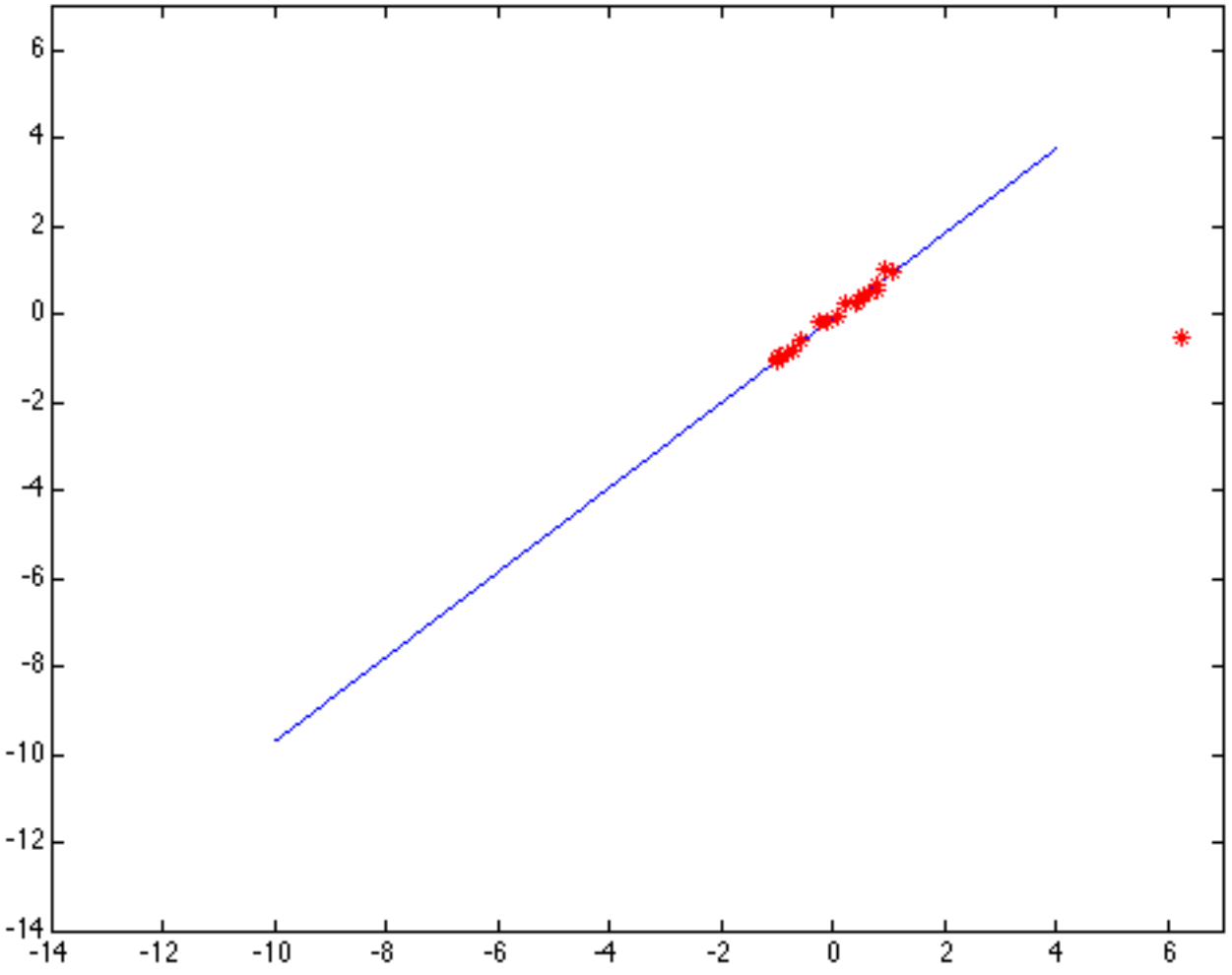
  - The scale parameter σ controls the point at which the function flattens out

Robust M-Estimator



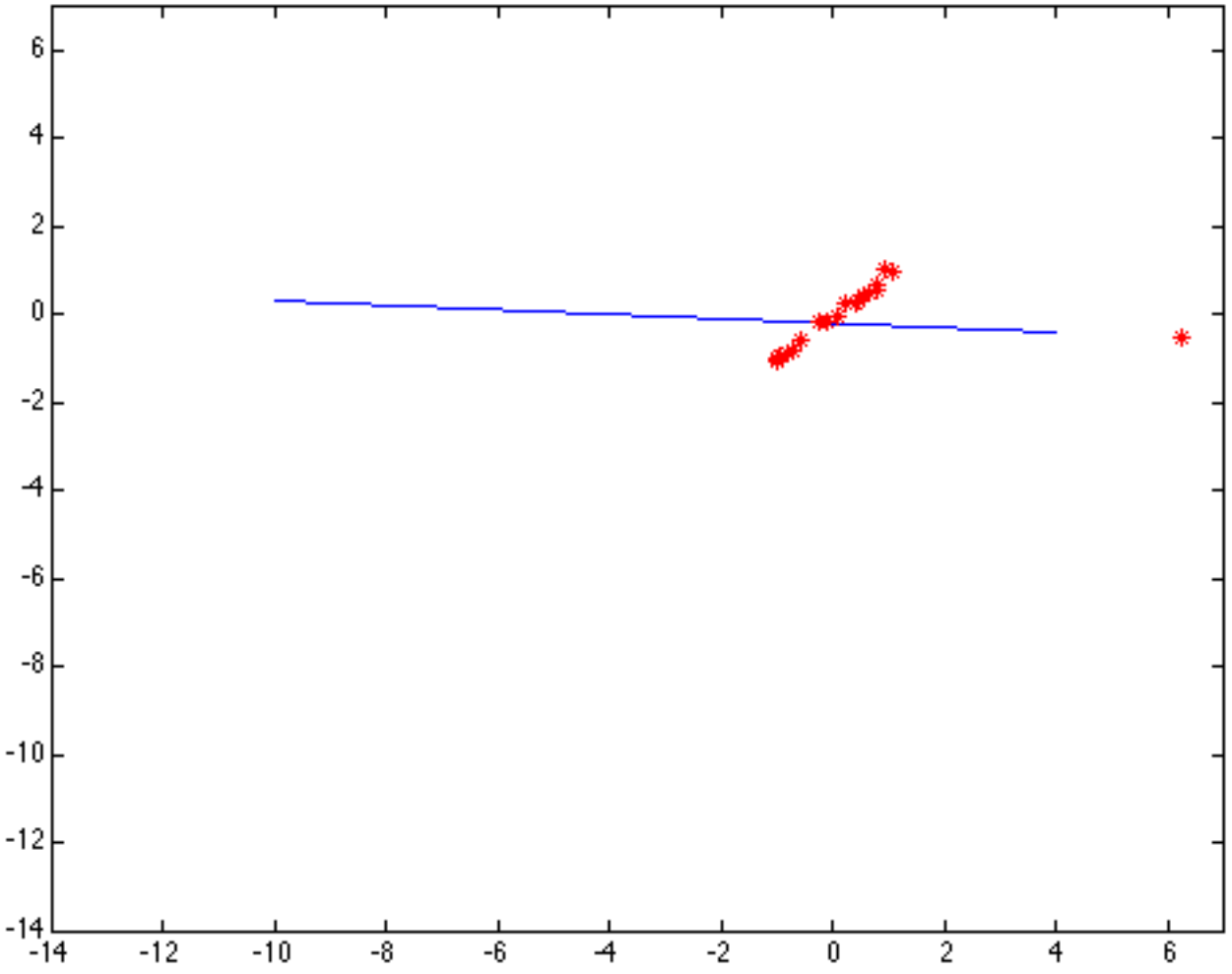
$\sigma^2 = 0.1, 1, 10$

Robustness



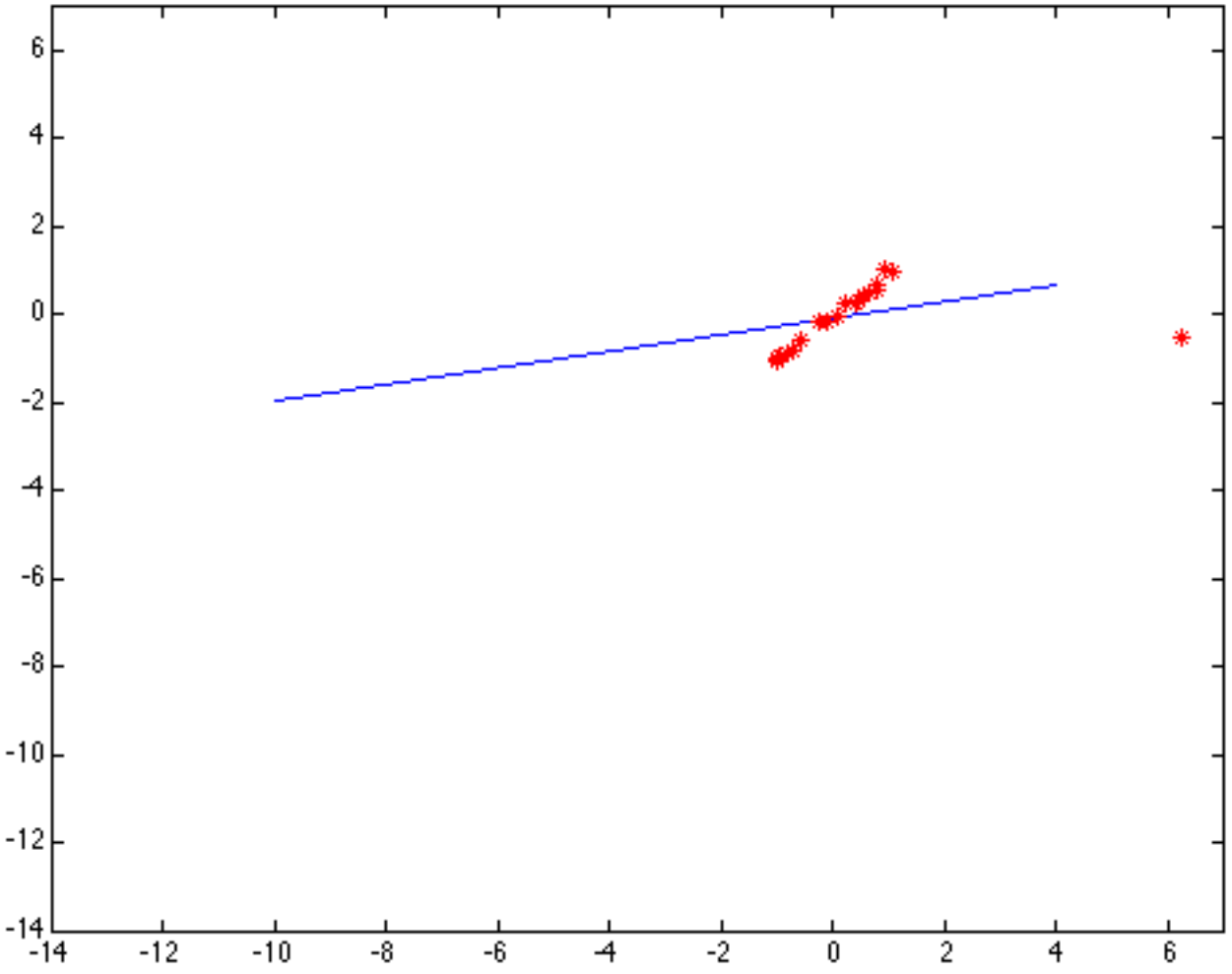
with an appropriate choice of σ

Robustness



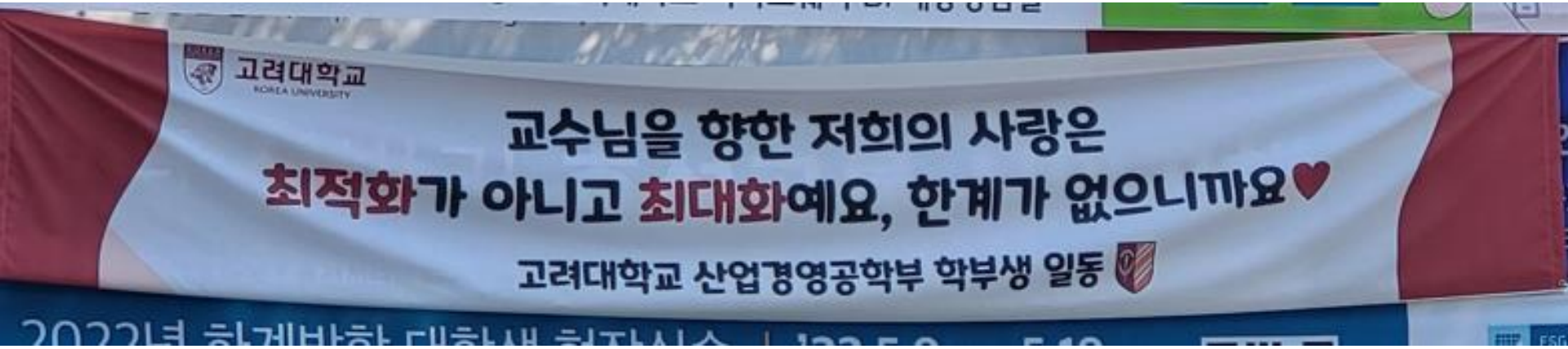
too small σ

Robustness



too large σ


대학전용
기호 2번 바람
제1회 고려대-LG전자 iPBL 지원자 모집
iPBL 주제 : 반려로봇가전 활용 아이디어/시나리오 발굴, 모터 드라이브에 AI기술 접목을 통한 신규가치 발굴, 포터블 냉장고 컨셉 발굴 외 9건
지원자격 : 팀장 – 4단계 BK21 사업 참여 대학원생, 팀원 – 고려대학교 전구성원(학부생 및 인문/사회 분야 한함)
지원금 : 1인당 100만원과 상품 및 상장 수여
지원기간 : 5월 1일 ~ 15일까지
상세내용 확인 및 지원 : https://moaform.com/q/JT9Yjw
LG전자
isu 이수그룹 신입사원 공개채용
모집기간 | 5.16 ~ 6.9
지원방법 | 이수그룹 채용 홈페이지 온라인 지원
신입사원 채용상담회 일정 5.16 ~ 5.20 장소 고려대학교 하나스퀘어 BI 채용상담실
신입 뽑음
고려대학교
교수님을 향한 저희의 사랑은
최적화가 아니고 최대화예요, 한계가 없으니까요♥
고려대학교 산업경영공학부 학부생 일동
posco 2022년 하계방학 대학생 현장실습 | '22.5.9(월) - 5.19(목)
포스코 스틸브릿지(채용연계형 프로그램) 모집
지원자격 | 졸업생 또는 '23.2월 이전 졸업예정자 ('23.1월중 입사가 가능한 자)
https://bit.ly/37v7Cav
공고확인
GREEN TOMORROW with POSCO
상세내용 보러가기
| 서류 접수기간 | 5.4 - 5.15 24:00
HYOSUNG
5월 9일(월) ~ 20일(금) 18시
효성그룹 신입공채


고려대학교
KOREA UNIVERSITY
교수님을 향한 저희의 사랑은
최적화가 아니고 최대화예요, 한계가 없으니까요♥
고려대학교 산업경영공학부 학부생 일동

RANSAC

- Choose a small subset uniformly at random
- Fit to that
- Anything that is close to the result is signal; all others are noises
- Refit
- Do this many times and choose the best

Algorithm 15.4: RANSAC: fitting lines using random sample consensus

Determine:
    n — the smallest number of points required
    k — the number of iterations required
    t — the threshold used to identify a point that fits well
    d — the number of nearby points required
      to assert a model fits well
Until k iterations have occurred
    Draw a sample of n points from the data
      uniformly and at random
    Fit to that set of n points
    For each data point outside the sample
      Test the distance from the point to the line
        against t; if the distance from the point to the line
        is less than t, the point is close
    end
    If there are d or more points close to the line
      then there is a good fit. Refit the line using all
      these points.
end
Use the best fit from this collection, using the
  fitting error as a criterion

RANSAC

- RANdom SAmple Consensus
  - Issues
    - How many times?
      - Often enough that we are likely to have a good line
    - How big a subset?
      - Smallest possible
    - What does ‘close’ mean?
      - Depends on the problem
    - What is a good line?
      - One where the number of nearby points is so big that it is unlikely to be all outliers