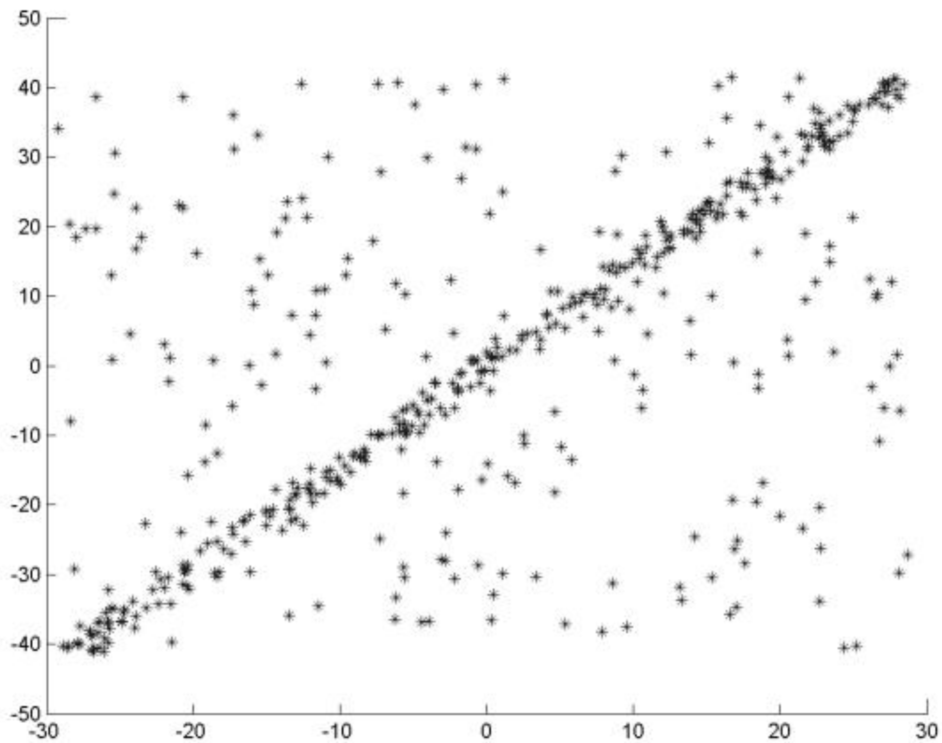
Image Processing

RANSAC

Example Tasks

Search for a straight line in a clutter of points



i.e. search for parameters a and b for the model $ax + by = 1$

given a training set $\big((x^1, y^1), (x^2, y^2) \ldots (x^i, y^i)\big)$

Example Tasks

Estimate the fundamental matrix F

i.e. parameters satisfying

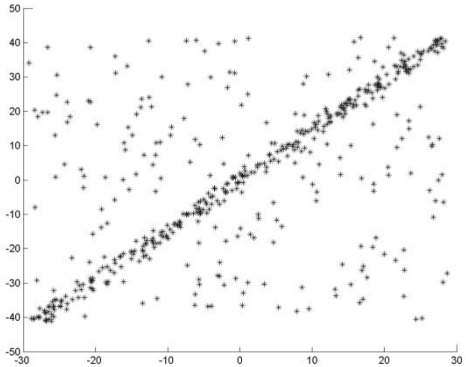
$$[x_{l1}, x_{l2}, 1] \cdot \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{r1} \\ x_{r2} \\ 1 \end{bmatrix} = 0$$

given a training set of correspondent pairs $\left((x_l^1, x_r^1), (x_l^2, x_r^2) \dots (x_l^i, x_r^i)\right)$

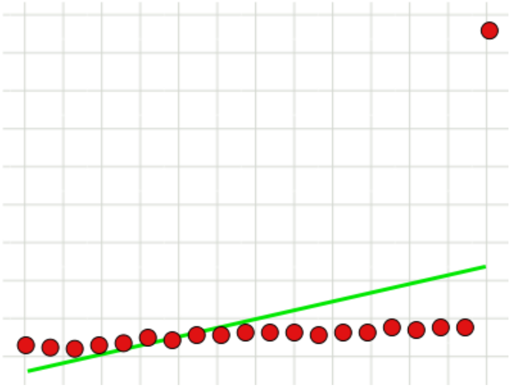
Two sources of errors

1. **Noise**: the coordinates deviate from the true ones according to some “rule” (probability) – the more far the less confident

2. **Outliers**: the data have nothing in common with the model to be estimated



Neglecting the latter can lead to a wrong estimation →



The way out – find outliers explicitly, estimate the model from inliers only

Task formulation

Let $x \in \mathcal{X}$ be the input space and $y \in \mathcal{Y}$ be the parameter space
The training data consist of data points $L = (x^1, x^2 \dots x^i),\ x^i \in \mathcal{X}$

Let an **evaluation function** $f : \mathcal{X} \times \mathcal{Y} \rightarrow \{0, 1\}$ be given that checks the consistency of a point x with a model y .

- straight line

$$f(x, y, a, b) = \begin{cases} 1 & \text{wenn } ax + by = 1 \\ 0 & \text{sonst.} \end{cases}$$

- fundamental matrix

$$f(x_l, x_r, F) = \begin{cases} 1 & \text{wenn } x_l F x_r = 0, \\ 0 & \text{sonst.} \end{cases}$$

The task is to find the parameter that is consistent with the **majority** of the data points:

$$y^* = \arg\max_{y \in \mathcal{Y}} \sum_i f(x^i, y)$$

Ideas

$$y^* = \arg\max_{y \in \mathcal{Y}} \sum_i f(x^i, y)$$

A naïve approach – enumerate all parameter values
→ **Hough Transform** (very time consuming, not possible at all for parameters of high dimension).

Observation: the parameter space is sparsely occupied – most of the parameter values “have no chance” (see board for an illustration).

Idea: do not try all values, but only some of them.

Which ones?

Oracle

Let an **oracle** be given – a function $g : \mathcal{X}^d \rightarrow \mathcal{Y}$ that estimates the model, which is consistent with a given d-tuple of data points.

Examples: a straight line can be estimated from 2 points, the fundamental matrix from 8 (or even 7) points (correspondences) etc.

Do not enumerate all parameter values but all d-tuples of data points, i.e. $L' \subset L, |L'| = d$.

$$L'^* = \arg\max_{L' \subset L, |L'| = d} \sum_i f\big(x^i, g(L')\big), \quad y^* = g(L'^*)$$

Examples: straight line – $|L|^2$ trials, fundamental matrix – $|L|^8$.

The optimization is performed over a **discrete domain**.

RANSAC

Random Sample Consensus, Fischler and Bolles 1981

Do not even try all subsets, but sample them randomly:

Wiederhole oft

  Würfele $L' \subset L,\ |L'| = d$

  Schätze $y = g(L')$

  Bewerte $f(y) = \sum_i f(x^i, y)$

  wenn $f(y) > f(y^*)$

    `setze` $y^* = y$ `und merke` $f(y^*)$

How many time to sample in order to reliable estimate the true model?

Convergence

Assumption: it is necessary to sample **any** d-tuple of inliers just **once** in order to estimate the model correctly.

Let ε be the probability of outliers.

The probability to sample d inliers is $(1-\varepsilon)^d$

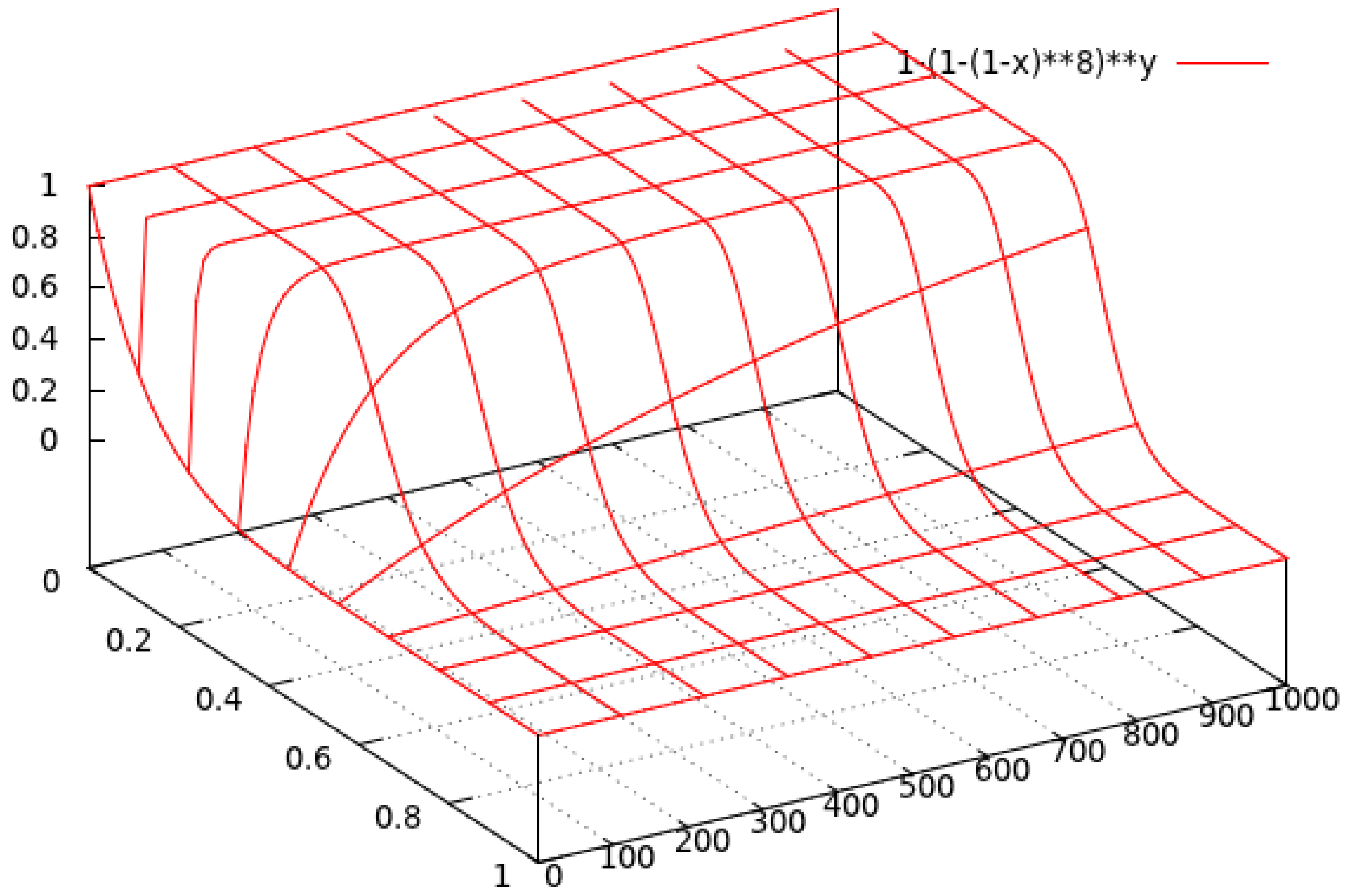
The probability of a “wrong” d-tuple is $1-(1-\varepsilon)^d$

The probability to sample n times only wrong tuples is $(1-(1-\varepsilon)^d)^n$

The probability to sample the “right” tuple at least once during the process (i.e. to estimate the correct model according to assumptions)

$$1-(1-(1-\varepsilon)^d)^n$$

Convergence



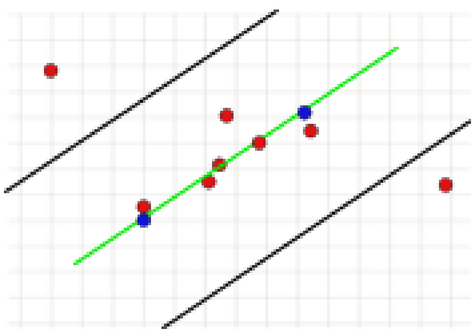
$$1-(1-(1-\varepsilon)^d)^n,\ d=8,\ \varepsilon\in[0,1],\ n=1\ldots1000$$

Problems, extensions

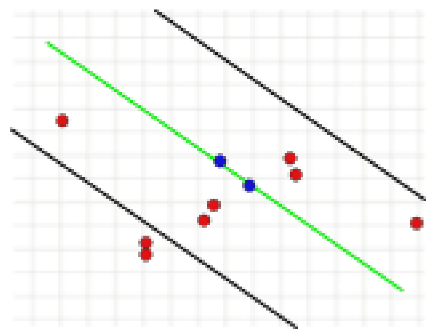
The evaluation functions as considered before are "too strict". They are almost newer satisfied in presence of noise (e.g. $x_l F x_r = 0$ newer holds exactly). When is a data point "good"?

Introduce **confidence intervals**, e.g. $|x_l F x_r| < \nu$, $|ax + by - 1| < \nu$
The choice of a right confidence interval is crucial.

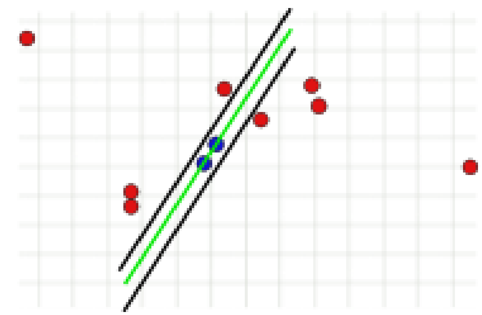
Examples:



Large confidence,
"right" model,
2 outliers



Large confidence,
"wrong" model,
2 outliers again



Small confidence,
Almost all points are
outliers (independent
on the model)

MSAC

Inliers are evaluated **quantitatively**

The evaluation function $f : \mathcal{X} \times \mathcal{Y} \rightarrow \{0, 1\}$ becomes $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$.

It assigns a **penalty** to each pair "(data point, model instance)".

If the data point is outside the confidence interval, the penalty is constant, otherwise it depends on the "distance" to the model.

Example for the fundamental matrix:

$$f(x_l, x_r, F) = \begin{cases} 1 & \text{wenn } |x_l F x_r| < \nu, \\ 0 & \text{sonst.} \end{cases}$$

becomes

$$f(x_l, x_r, F) = \min(|x_l F x_r|, \nu)$$

→ the task is to find the model of the minimum average penalty

$$y^* = \arg\min_y \sum_i f(x^i, y)$$

Problems, extensions

Evaluation of a hypothesis y, i.e. $\sum_i f(x^i, y)$ is often time consuming

$T_{d,d}$ -test (**Randomized** RANSAC):

instead to check all data points $x^i \in L$

1. Sample m points from L
2. If all of them are good, check all others as before
3. If there is at least one bad point among m, reject the hypothesis

It is of course possible that good hypotheses are rejected. However it saves time (bad hypotheses are recognized fast) → one can sample more often → the right hypothesis is caught sometime anyway → all-in-all often profitable (depending on application).

Problems, extensions

The choice of the **Oracle** is crucial

Example – the fundamental matrix:

a) 8-point algorithm – less precise, fast and simple, harder to catch ($d = 8$)
b) 7-point algorithm – more precise, more complex, easier to catch ($d = 7$)

→ compromises are thinkable, e.g. "sample + optimize" etc.

All the stuff is easy to **parallelize**.

Other applications

Panorama stitching: find the homographies.

Puzzles:
S. Winkelbach, M. Rilk, C. Schönfelder, F. Wahl:
„Fast Random Sample Matching of 3d Fragments“