3D Computer Vision

Radim Šára   Martin Matoušek

Center for Machine Perception
Department of Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague

https://cw.fel.cvut.cz/wiki/courses/tdv/start

http://cmp.felk.cvut.cz
mailto:sara@cmp.felk.cvut.cz
phone ext. 7203

rev. November 7, 2023

Open Informatics Master's Course

►The Least-Squares Triangulation by SVD

- if $\mathbf{D}$ is full-rank we may minimize the algebraic least-squares error

$$\boldsymbol{\varepsilon}^2(\underline{\mathbf{X}}) = \|\mathbf{D}\underline{\mathbf{X}}\|^2 \quad \text{s.t.} \quad \|\underline{\mathbf{X}}\| = 1, \qquad \underline{\mathbf{X}} \in \mathbb{R}^4$$

- let $\mathbf{d}_i$ be the i-th row of $\mathbf{D}$ reshaped as a column vector, then

$$\|\mathbf{D}\underline{\mathbf{X}}\|^2 = \sum_{i=1}^{4} (\mathbf{d}_i^\top \underline{\mathbf{X}})^2 = \sum_{i=1}^{4} \underline{\mathbf{X}}^\top \mathbf{d}_i \mathbf{d}_i^\top \underline{\mathbf{X}} = \underline{\mathbf{X}}^\top \mathbf{Q}\, \underline{\mathbf{X}}, \text{ where } \mathbf{Q} = \sum_{i=1}^{4} \mathbf{d}_i \mathbf{d}_i^\top = \mathbf{D}^\top \mathbf{D} \in \mathbb{R}^{4,4}$$

- we write the SVD of $\mathbf{Q}$ as $\mathbf{Q} = \sum_{j=1}^{4} \sigma_j^2\, \mathbf{u}_j \mathbf{u}_j^\top$, in which [Golub & van Loan 2013, Sec. 2.5]

$$\sigma_1^2 \geq \cdots \geq \sigma_4^2 \geq 0 \quad \text{and} \quad \mathbf{u}_l^\top \mathbf{u}_m = \begin{cases} 0 & \text{if } l \neq m \\ 1 & \text{otherwise} \end{cases}$$

- then $\min_{\mathbf{q}, \|\mathbf{q}\|=1} \mathbf{q}^\top \mathbf{Q}\, \mathbf{q} = \sigma_4^2$ and $\underline{\mathbf{X}} = \arg\min_{\mathbf{q}, \|\mathbf{q}\|=1} \mathbf{q}^\top \mathbf{Q}\, \mathbf{q} = \mathbf{u}_4$ $\mathbf{u}_4$ – the last column of $\mathbf{U}$ from $\mathrm{SVD}(\mathbf{Q})$

Proof (by contradiction).

Let $\bar{\mathbf{q}} = \sum_{i=1}^{4} a_i \mathbf{u}_i$ s.t. $\sum_{i=1}^{4} a_i^2 = 1$, then $\|\bar{\mathbf{q}}\| = 1$, as desired, and

$$\bar{\mathbf{q}}^\top \mathbf{Q}\, \bar{\mathbf{q}} = \sum_{j=1}^{4} \sigma_j^2\, \bar{\mathbf{q}}^\top \mathbf{u}_j\, \mathbf{u}_j^\top \bar{\mathbf{q}} = \sum_{j=1}^{4} \sigma_j^2\, (\mathbf{u}_j^\top \bar{\mathbf{q}})^2 = \cdots = \sum_{j=1}^{4} a_j^2 \sigma_j^2 \;\geq\; \sum_{j=1}^{4} a_j^2 \sigma_4^2 = \left(\sum_{j=1}^{4} a_j^2\right) \sigma_4^2 = \sigma_4^2$$

since $\sigma_j \geq \sigma_4$ □

- if $\sigma_4 \ll \sigma_3$, there is a unique solution $\underline{\mathbf{X}} = \mathbf{u}_4$ with residual error $(\mathbf{D}\,\underline{\mathbf{X}})^2 = \sigma_4^2$

  the quality (conditioning) of the solution may be expressed as $q = \sigma_3/\sigma_4$ (greater is better)

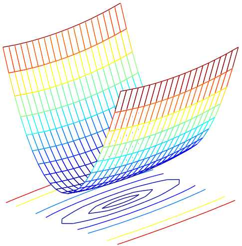
Matlab code for the least-squares solver:

```
[U,O,V] = svd(D);
X = V(:,end);
q = sqrt(O(end-1,end-1)/O(end,end));
```

⊛ P1; 1pt: Why did we decompose $\mathbf{D}$ here, and not $\mathbf{Q} = \mathbf{D}^\top\mathbf{D}$?

▶Numerical Conditioning

- The equation $\mathbf{D}\underline{\mathbf{X}} = \mathbf{0}$ in (16) may be ill-conditioned for numerical computation, which results in a poor estimate for $\underline{\mathbf{X}}$.



Why: on a row of $\mathbf{D}$ there are big entries together with small entries, e.g. of orders projection centers in mm, image points in px

$$\begin{bmatrix} 10^3 & 0 & 10^3 & 10^6 \\ 0 & 10^3 & 10^3 & 10^6 \\ 10^3 & 0 & 10^3 & 10^6 \\ 0 & 10^3 & 10^3 & 10^6 \end{bmatrix}$$

Quick fix:

1. re-scale the problem by a regular diagonal conditioning matrix $\mathbf{S} \in \mathbb{R}^{4,4}$

$$\mathbf{0} = \mathbf{D}\,\underline{\mathbf{X}} = \mathbf{D}\,\mathbf{S}\,\mathbf{S}^{-1}\underline{\mathbf{X}} = \bar{\mathbf{D}}\,\underline{\bar{\mathbf{X}}}$$

   choose $\mathbf{S}$ to make the entries in $\hat{\mathbf{D}}$ all smaller than unity in absolute value, e.g.:

$$\mathbf{S} = \mathrm{diag}(10^{-3}, 10^{-3}, 10^{-3}, 10^{-6})$$ `S = diag(1./max(abs(D), [], 1))`

2. solve for $\underline{\bar{\mathbf{X}}}$ as before
3. get the final solution as $\underline{\mathbf{X}} = \mathbf{S}\,\underline{\bar{\mathbf{X}}}$

- when SVD is used in camera resection from six points →62, conditioning is essential for success

▶We Have Added to The ZOO (cont'd from →69)

problem	given	unknown	slide
camera resection	6 world–img correspondences $\{(X_i,\, m_i)\}_{i=1}^{6}$	$\mathbf{P}$	62
exterior orientation	$\mathbf{K}$, 3 world–img correspondences $\{(X_i,\, m_i)\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{t}$	66
relative pointcloud orientation	3 world-world correspondences $\{(X_i,\, Y_i)\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{t}$	70
fundamental matrix	7 img–img correspondences $\{(m_i,\, m_i')\}_{i=1}^{7}$	$\mathbf{F}$	85
relative camera orientation	$\mathbf{K}$, 5 img–img correspondences $\{(m_i,\, m_i')\}_{i=1}^{5}$	$\mathbf{R}$, $\mathbf{t}$	89
triangulation	$\mathbf{P}_1$, $\mathbf{P}_2$, 1 img–img correspondence $(m,\, m')$	$\underline{\mathbf{X}}$	90

A bigger ZOO at http://aag.ciirc.cvut.cz/minimal/

calibrated problems

- have fewer degenerate configurations
- can do with fewer points (good for geometry proposal generators →123)

- algebraic error optimization (SVD) makes sense in camera resection and triangulation only
- but it is not the best method; we will now focus on 'optimizing optimally'

Module V

Optimization for 3D Vision

covered by

[1] [H&Z] Secs: 11.4, 11.6, 4.7

[2] Fischler, M.A. and Bolles, R.C . Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM* 24(6):381–395, 1981

additional references

P. D. Sampson. Fitting conic sections to 'very scattered' data: An iterative refinement of the Bookstein algorithm. *Computer Vision, Graphics, and Image Processing*, 18:97–108, 1982.

O. Chum, J. Matas, and J. Kittler. Locally optimized RANSAC. In *Proc DAGM*, LNCS 2781:236–243. Springer-Verlag, 2003.

O. Chum, T. Werner, and J. Matas. Epipolar geometry estimation via RANSAC benefits from the oriented epipolar constraint. In *Proc ICPR*, vol 1:112–115, 2004.

▶Algebraic Error vs Reprojection Error

- algebraic error c – camera index, (u^c, v^c) – image coordinates →91

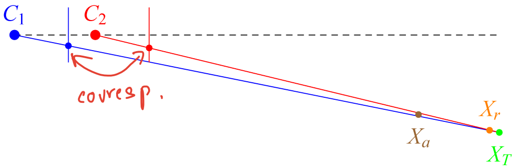
$$\varepsilon^2(\underline{\mathbf{X}}) = \|\mathbf{D}\underline{\mathbf{X}}\|^2 = \sum_{c=1}^{2} \left[\left(u^c (\mathbf{p}_3^c)^\top \underline{\mathbf{X}} - (\mathbf{p}_1^c)^\top \underline{\mathbf{X}} \right)^2 + \left(v^c (\mathbf{p}_3^c)^\top \underline{\mathbf{X}} - (\mathbf{p}_2^c)^\top \underline{\mathbf{X}} \right)^2 \right]$$

- reprojection error

$$e^2(\underline{\mathbf{X}}) = \sum_{c=1}^{2} \left[\left(u^c - \frac{(\mathbf{p}_1^c)^\top \underline{\mathbf{X}}}{(\mathbf{p}_3^c)^\top \underline{\mathbf{X}}} \right)^2 + \left(v^c - \frac{(\mathbf{p}_2^c)^\top \underline{\mathbf{X}}}{(\mathbf{p}_3^c)^\top \underline{\mathbf{X}}} \right)^2 \right]$$

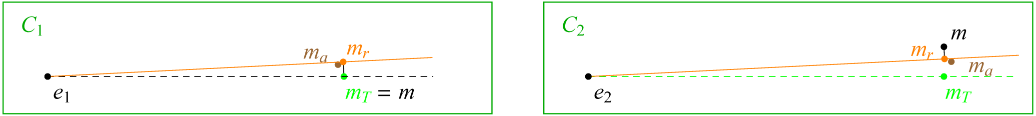
- algebraic error zero ⇔ reprojection error zero $\qquad \sigma_4 = 0 \Rightarrow$ non-trivial null space
- epipolar constraint satisfied ⇒ equivalent results
- in general: minimizing algebraic error is cheap but it gives inferior results
- minimizing reprojection error is expensive but it gives good results
- the midpoint of the common perpendicular to both optical rays gives about 50% greater error in 3D
- the golden standard method – deferred to →108

Algebraic Error vs Reprojection Error: Example



- forward camera motion
- error $f/50$ in image 2, orthogonal to epipolar plane

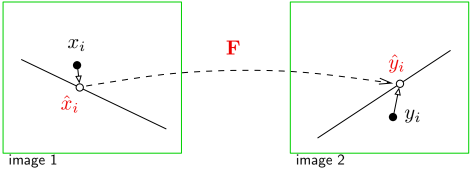
  X_T – noiseless ground truth position
  X_r – reprojection error minimizer
  X_a – algebraic error minimizer
  m – measurement (m_T with noise in v^2)



- this demonstrates a difficult configuration (forward camera motion) and a random correspondence
- noise-free ground-truth triangulation from m_T is X_T
- reprojection error minimizer X_r has an error due to simulated noise in image detections (black m)
- algebraic error minimizer X_a essentially failed

Background problems: (1) Given at least 8 matched points $x_i \leftrightarrow y_j$ in a general position, estimate the most 'likely' fundamental matrix $\mathbf{F}$; (2) given $\mathbf{F}$ triangulate 3D point from $x_i \leftrightarrow y_j$.

$$\mathbf{x}_i = (u_i^1,\, v_i^1), \quad \mathbf{y}_i = (u_i^2,\, v_i^2), \qquad i = 1, 2, \dots, k, \quad k \geq 8 \text{ for (1) or } k = 1 \text{ for (2)}$$



- detected points (measurements) x_i, y_i
- we introduce matches $\mathbf{Z}_i = (\mathbf{x}_i, \mathbf{y}_i) = (u_i^1, v_i^1, u_i^2, v_i^2) \in \mathbb{R}^4$; and the set $Z = \{\mathbf{Z}_i\}_{i=1}^k$
- corrected points $\hat{x}_i$, $\hat{y}_i$; $\hat{\mathbf{Z}}_i = (\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_i) = (\hat{u}_i^1, \hat{v}_i^1, \hat{u}_i^2, \hat{v}_i^2)$; $\hat{Z} = \{\hat{\mathbf{Z}}_i\}_{i=1}^k$ are correspondences
- correspondences satisfy the epipolar geometry exactly $\underline{\hat{\mathbf{y}}}_i^\top \mathbf{F}\, \underline{\hat{\mathbf{x}}}_i = 0$, $i = 1, \dots, k$
- small correction is more probable
- let $\mathbf{e}_i(\cdot)$ be the 'reprojection error' (vector) per match i,

$$\mathbf{e}_i(x_i, y_i \mid \hat{x}_i, \hat{y}_i, \mathbf{F}) = \begin{bmatrix} \mathbf{x}_i - \hat{\mathbf{x}}_i \\ \mathbf{y}_i - \hat{\mathbf{y}}_i \end{bmatrix} = \mathbf{e}_i(\mathbf{Z}_i \mid \hat{\mathbf{Z}}_i, \mathbf{F}) = \mathbf{Z}_i - \hat{\mathbf{Z}}_i(\mathbf{F})$$
$$\|\mathbf{e}_i(\cdot)\|^2 \stackrel{\text{def}}{=} \mathbf{e}_i^2(\cdot) = \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2 + \|\mathbf{y}_i - \hat{\mathbf{y}}_i\|^2 = \|\mathbf{Z}_i - \hat{\mathbf{Z}}_i(\mathbf{F})\|^2 \quad \in \mathbb{R}^4 \tag{17}$$

Consider the estimation of $\mathbf{F}$

- the total reprojection error (of all data) is

$$L(Z \mid \hat{Z}, \mathbf{F}) = \sum_{i=1}^{k} \mathbf{e}_i^2(x_i, y_i \mid \hat{x}_i, \hat{y}_i, \mathbf{F}) \quad = \sum_{i=1}^{k} \mathbf{e}_i^2(\mathbf{Z}_i \mid \hat{\mathbf{Z}}_i, \mathbf{F})$$

- and the optimization problem is

$$(\hat{Z}^*, \mathbf{F}^*) = \arg\min_{\mathbf{F}, \hat{Z}} L(Z \mid \hat{Z}, \mathbf{F}) \quad \text{s.t.} \quad \operatorname{rank} \mathbf{F} = 2, \quad \underline{\hat{\mathbf{y}}}_i^\top \mathbf{F}\, \underline{\hat{\mathbf{x}}}_i = 0, \quad (\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i) \in \hat{\mathbf{Z}}_i \tag{18}$$

$\hat{x}_i \simeq P_1 X_i \; ; \; \hat{y}_i \simeq P_2 X_i$

Possible approaches

- they differ in how the correspondences $\hat{x}_i$, $\hat{y}_i$ are obtained:
  1. direct optimization of reprojection error over all variables $\hat{Z}$, $\mathbf{F}$ — needs a good parameterization for $\mathbf{F}$ →100
  2. Sampson optimal correction = partial correction of $\mathbf{Z}_i$ towards $\hat{\mathbf{Z}}_i$ used in an iterative minimization over $\mathbf{F}$ →102

Method 1: Reprojection Error Optimization: Idea

- we need to encode the constraints $\hat{\underline{\mathbf{y}}}_i \mathbf{F} \hat{\underline{\mathbf{x}}}_i = 0$, $\operatorname{rank} \mathbf{F} = 2$
- idea: reconstruct 3D point via equivalent projection matrices and use reprojection error
- the equivalent projection matrices are see [H&Z,Sec. 9.5] for a complete characterization

$$\mathbf{P}_1 = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}, \quad \mathbf{P}_2(\mathbf{F}) = \begin{bmatrix} \underbrace{[\underline{\mathbf{e}}_2]_\times \mathbf{F} + \underline{\mathbf{e}}_2 \underline{\mathbf{e}}_1^\top}_{Q} & \underbrace{\underline{\mathbf{e}}_2}_{q} \end{bmatrix}, \quad \text{s.t.} \quad \mathbf{F}\mathbf{e}_1 = \mathbf{0},\ \mathbf{e}_2^\top \mathbf{F} = \mathbf{0} \tag{19}$$

$Q e_1 \neq 0$

due CD + 3W

⊛ H3; 2pt: Given rank-2 matrix $\mathbf{F}$, let $\underline{\mathbf{e}}_1$, $\underline{\mathbf{e}}_2$ be the right and left nullspace basis vectors of $\mathbf{F}$, respectively. Verify that such $\mathbf{F}$ is a fundamental matrix of $\mathbf{P}_1$, $\mathbf{P}_2$ from (19).

Hints:

(1) consider $\hat{\underline{\mathbf{x}}}_i = \mathbf{P}_1 \underline{\mathbf{X}}_i$ and $\hat{\underline{\mathbf{y}}}_i = \mathbf{P}_2 \underline{\mathbf{X}}_i$

(2) $\mathbf{A}$ is skew symmetric iff $\mathbf{x}^\top \mathbf{A} \mathbf{x} = 0$ for all vectors $\mathbf{x}$.

(cont'd) Reprojection Error Optimization: Algorithm

1. compute $\mathbf{F}^{(0)}$ by the 7-point algorithm →85; construct camera $\mathbf{P}_2^{(0)}$ from $\mathbf{F}^{(0)}$ using (19)
2. triangulate 3D points $\hat{\mathbf{X}}_i^{(0)}$ from matches (x_i, y_i) for all $i = 1, \ldots, k$ by the SVD alg. →90
3. starting from $\mathbf{P}_2^{(0)}$, $\hat{\mathbf{X}}_{1:k}^{(0)}$ minimize the reprojection error (17)

7

$$(\hat{\mathbf{X}}^*_{1:k}, \mathbf{F}^*) = \arg \min_{\mathbf{F},\, \hat{\mathbf{X}}_{1:k}} \sum_{i=1}^{k} \mathbf{e}_i^2(\mathbf{Z}_i \mid \hat{\mathbf{Z}}_i(\hat{\mathbf{X}}_i, \mathbf{P}_2(\mathbf{F})))$$

where

$$\hat{\mathbf{Z}}_i = (\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i) \ \text{(Cartesian)}, \quad \hat{\underline{\mathbf{x}}}_i \simeq \mathbf{P}_1 \hat{\underline{\mathbf{X}}}_i, \ \ \hat{\underline{\mathbf{y}}}_i \simeq \mathbf{P}_2(\mathbf{F})\, \hat{\underline{\mathbf{X}}}_i \ \text{(homogeneous)}$$

- non-linear, non-convex problem
- solves $\mathbf{F}$ estimation and triangulation of all k points jointly
- the solver would be quite slow
- $3k + 7$ parameters to be found: latent: $\hat{\mathbf{X}}_i$, for all i (correspondences!), non-latent: $\mathbf{F}$
- we need minimal representations for $\hat{\mathbf{X}}_i$ and $\mathbf{F}$ →153 or introduce constraints
- there are other pitfalls; this is essentially bundle adjustment; we will return to this later →141

▶Method 2: First-Order Error Approximation

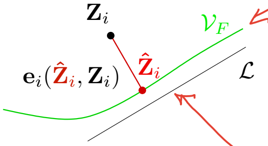
An elegant method for solving problems like (18):

- we will get rid of the latent parameters $\hat{X}$ needed for obtaining the correction

[H&Z, p. 287], [Sampson 1982]

- we will recycle the algebraic error $\boldsymbol{\varepsilon} = \underline{\mathbf{y}}^\top \mathbf{F}\, \underline{\mathbf{x}}$ from →85

- consider matches $\mathbf{Z}_i$, correspondences $\hat{\mathbf{Z}}_i$, and reprojection error $\mathbf{e}_i = \|\mathbf{Z}_i - \hat{\mathbf{Z}}_i\|^2$
- correspondences satisfy $\hat{\underline{\mathbf{y}}}_i^\top \mathbf{F}\, \hat{\underline{\mathbf{x}}}_i = 0$, $\quad \hat{\underline{\mathbf{x}}}_i = (\hat{u}^1, \hat{v}^1, 1)$, $\hat{\underline{\mathbf{y}}}_i = (\hat{u}^2, \hat{v}^2, 1)$
- this is a manifold $\mathcal{V}_F \in \mathbb{R}^4$: a set of points $\hat{\mathbf{Z}} = (\hat{u}^1, \hat{v}^1, \hat{u}^2, \hat{v}^2) \in \mathbb{R}^4$ consistent with $\mathbf{F}$
- algebraic error vanishes for $\hat{\mathbf{Z}}_i$: $\boxed{\mathbf{0} = \boldsymbol{\varepsilon}_i(\hat{\mathbf{Z}}_i) = \hat{\underline{\mathbf{y}}}_i^\top \mathbf{F}\, \hat{\underline{\mathbf{x}}}_i}$ $\qquad$ $\boldsymbol{\varepsilon}(\mathbf{Z})$ is a function of $\mathbf{Z}$



Sampson's idea: Linearize the algebraic error $\boldsymbol{\varepsilon}(\mathbf{Z})$ at $\mathbf{Z}_i$ (where it is non-zero) and evaluate the resulting linear function at $\hat{\mathbf{Z}}_i$ (where it is zero). The zero-crossing replaces $\mathcal{V}_F$ by a linear manifold $\mathcal{L}$. The point on $\mathcal{V}_F$ closest to $\mathbf{Z}_i$ is replaced by the closest point on $\mathcal{L}$.

$$\mathcal{L}: \quad 0 = \boldsymbol{\varepsilon}_i(\hat{\mathbf{Z}}_i) \approx \boldsymbol{\varepsilon}_i(\mathbf{Z}_i) + \frac{\partial \boldsymbol{\varepsilon}_i(\mathbf{Z}_i)}{\partial \mathbf{Z}_i}(\hat{\mathbf{Z}}_i - \mathbf{Z}_i) \qquad \text{linear in } \hat{\mathbf{Z}}_i$$

$\varepsilon_i(\hat{z}_i \mid F)$

▶Sampson's Approximation of Reprojection Error

translates $\varepsilon \to e$

- linearize $\boldsymbol{\varepsilon}(\mathbf{Z})$ at match $\mathbf{Z}_i$, evaluate it at correspondence $\hat{\mathbf{Z}}_i$

$$\boldsymbol{\varepsilon}_i(\mathbf{Z}_i) + \underbrace{\frac{\partial \boldsymbol{\varepsilon}_i(\mathbf{Z}_i)}{\partial \mathbf{Z}_i}}_{\mathbf{J}_i(\mathbf{Z}_i)} \underbrace{(\hat{\mathbf{Z}}_i - \mathbf{Z}_i)}_{\mathbf{e}_i(\hat{\mathbf{Z}}_i, \mathbf{Z}_i)} \stackrel{\text{def}}{=} \underbrace{\boldsymbol{\varepsilon}_i(\mathbf{Z}_i)}_{\text{given}} + \mathbf{J}_i(\mathbf{Z}_i)\, \underbrace{\mathbf{e}_i(\hat{\mathbf{Z}}_i, \mathbf{Z}_i)}_{\text{wanted}} = \boldsymbol{\varepsilon}_i(\hat{\mathbf{Z}}_i) \stackrel{!}{=} 0$$

- goal: compute function $\mathbf{e}_i(\cdot)$ from $\boldsymbol{\varepsilon}_i(\cdot)$, where $\mathbf{e}_i(\cdot)$ is the distance of $\hat{\mathbf{Z}}_i$ from $\mathbf{Z}_i$
- we have a linear underconstrained equation for $\mathbf{e}_i(\cdot)$ — e.g. $\boldsymbol{\varepsilon}_i \in \mathbb{R}$, $\mathbf{e}_i \in \mathbb{R}^4$
- we look for a minimal $\mathbf{e}_i(\cdot)$ per match i

$$\mathbf{e}_i(\cdot)^* = \arg\min_{\mathbf{e}_i(\cdot)} \|\mathbf{e}_i(\cdot)\|^2 \quad \text{subject to} \quad \boldsymbol{\varepsilon}_i(\cdot) + \mathbf{J}_i(\cdot)\,\mathbf{e}_i(\cdot) = 0$$

- which has a closed-form solution note that $\mathbf{J}_i(\cdot)$ is not invertible! — ⊛ P1; 1pt: derive $\mathbf{e}_i^*(\cdot)$

$$\mathbf{e}_i^*(\cdot) = -\mathbf{J}_i^\top (\mathbf{J}_i \mathbf{J}_i^\top)^{-1} \boldsymbol{\varepsilon}_i(\cdot) \qquad \text{pseudo-inverse}$$
$$\|\mathbf{e}_i^*(\cdot)\|^2 = \boldsymbol{\varepsilon}_i^\top(\cdot)(\mathbf{J}_i \mathbf{J}_i^\top)^{-1} \boldsymbol{\varepsilon}_i(\cdot) \qquad (20)$$

- this maps $\boldsymbol{\varepsilon}_i(\cdot)$ to an estimate of $\mathbf{e}_i(\cdot)$ per correspondence
- we need $\|\mathbf{e}_i\|^2$ for the $\mathbf{F}$ estimation, we will need $\mathbf{e}_i$ for triangulation in the golden-standard alg. →108
- the unknown parameters $\mathbf{F}$ are inside: $\mathbf{e}_i = \mathbf{e}_i(\mathbf{F})$, $\boldsymbol{\varepsilon}_i = \boldsymbol{\varepsilon}_i(\mathbf{F})$, $\mathbf{J}_i = \mathbf{J}_i(\mathbf{F})$

▶Example: Fitting A Circle To Scattered Points

Problem: Fit an origin-centered circle $\mathcal{C}$: $\|\mathbf{x}\|^2 - r^2 = 0$ to a set of 2D points $Z = \{x_i\}_{i=1}^{k}$

1. consider radial error as the 'algebraic error' $\boldsymbol{\varepsilon}(\mathbf{x}) = \|\mathbf{x}\|^2 - r^2$ — 'arbitrary' choice
2. linearize it at $\hat{\mathbf{x}}$ — we are dropping i in $\boldsymbol{\varepsilon}_i$, $\mathbf{e}_i$ etc for clarity

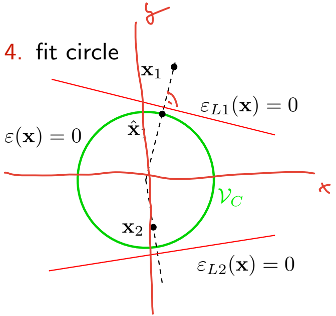
$$\boldsymbol{\varepsilon}(\hat{\mathbf{x}}) \approx \boldsymbol{\varepsilon}(\mathbf{x}) + \underbrace{\frac{\partial \boldsymbol{\varepsilon}(\mathbf{x})}{\partial \mathbf{x}}}_{\mathbf{J}(\mathbf{x})=2\mathbf{x}^\top} \underbrace{(\hat{\mathbf{x}} - \mathbf{x})}_{\mathbf{e}(\hat{\mathbf{x}},\mathbf{x})} = \cdots = 2\,\mathbf{x}^\top \hat{\mathbf{x}} - (r^2 + \|\mathbf{x}\|^2) \stackrel{\text{def}}{=} \boldsymbol{\varepsilon}_L(\hat{\mathbf{x}})$$

$\boldsymbol{\varepsilon}_L(\hat{\mathbf{x}}) = 0$ is a line with normal $\frac{\mathbf{x}}{\|\mathbf{x}\|}$ and intercept $\frac{r^2+\|\mathbf{x}\|^2}{2\|\mathbf{x}\|}$ — not tangent to $\mathcal{C}$, outside!

3. using (20), express error approximation $\mathbf{e}^*$ as

$$\|\mathbf{e}^*\|^2 = \boldsymbol{\varepsilon}^\top (\mathbf{J}\mathbf{J}^\top)^{-1} \boldsymbol{\varepsilon} = \frac{(\|\mathbf{x}\|^2 - r^2)^2}{4\|\mathbf{x}\|^2}$$

4. fit circle



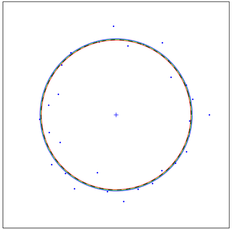
$$r^* = \arg\min_r \sum_{i=1}^{k} \frac{(\|\mathbf{x}_i\|^2 - r^2)^2}{4\|\mathbf{x}_i\|^2} = \cdots = \left(\frac{1}{k}\sum_{i=1}^{k} \frac{1}{\|\mathbf{x}_i\|^2}\right)^{-\frac{1}{2}}$$

- this example results in a convex quadratic optimization problem
- note that the 'algebraic error' minimizer is different:

$$\arg\min_r \sum_{i=1}^{k} (\|\mathbf{x}_i\|^2 - r^2)^2 = \left(\frac{1}{k}\sum_{i=1}^{k} \|\mathbf{x}_i\|^2\right)^{\frac{1}{2}}$$

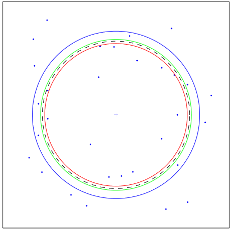
Circle Fitting: Some Results

medium radial noise



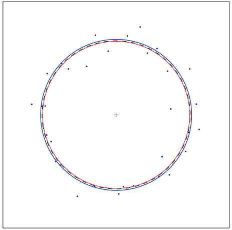
opt: 1.8, Smp: 1.9, dir: 2.3

big radial noise



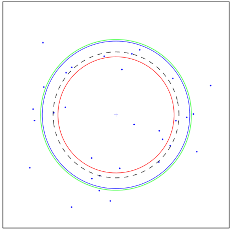
1.6, 1.8, 2.6

medium isotropic noise



1.8, 2.0, 2.2

big isotropic noise



1.6, 2.0, 2.4

mean ranks over 10 000 random trials with $k = 32$ samples; smaller is better

solid green – ground truth
solid red – Sampson error $\mathbf{e}$ minimizer
solid blue – direct algebraic radial error $\boldsymbol{\varepsilon}$ minimizer
dashed black – optimal estimator for isotropic error

optimal estimator for isotropic error (black, dashed):

$$r \approx \frac{3}{4k}\sum_{i=1}^{k}\|\mathbf{x}_i\| + \sqrt{\left(\frac{3}{4k}\sum_{i=1}^{k}\|\mathbf{x}_i\|\right)^2 - \frac{1}{2k}\sum_{i=1}^{k}\|\mathbf{x}_i\|^2}$$

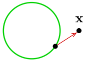
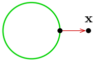
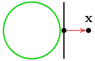
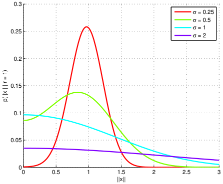
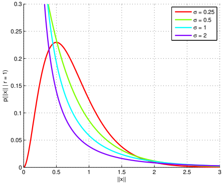
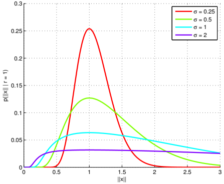
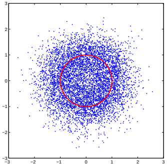
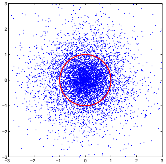
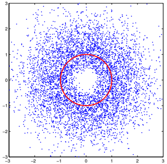
which method is better?

- error should model noise, radial noise and isotropic noise behave differently
- ground truth: Normally distributed isotropic error, Gamma-distributed radial error (!) the devil is hiding there
- Sampson: better for the radial distribution model; Direct: better for the isotropic model
- no matter how corrected, the algebraic error minimizer is not an unbiased parameter estimator

Cramér-Rao bound tells us how close one can get with unbiased estimator and given k

Discussion: On The Art of Probabilistic Model Design. . .

- a few probabilistic models for fitting zero-centered circle C of radius r to points in $\mathbb{R}^2$

	marginalized over C	orthogonal deviation from C	Sampson approximation
error model	 $N(\mathbf{0}, \sigma^2\mathbf{I})$	 $\Gamma(\cdot, \cdot)$	 $N(\mathbf{0}, \sigma^2\mathbf{I})$
radial p.d.f.			
random sample			
$p(\mathbf{x} \mid r)$	$\approx \frac{1}{\sigma\sqrt{(2\pi)^3 r\,\|\mathbf{x}\|}}\, e^{-\frac{(\|\mathbf{x}\|-r)^2}{2\sigma^2}}$	$\frac{1}{2\pi\Gamma(\frac{r^2}{\sigma})}\,\frac{1}{\|\mathbf{x}\|^2}\left(\frac{r\|\mathbf{x}\|}{\sigma}\right)^{\frac{r^2}{\sigma}} e^{-\frac{r\|\mathbf{x}\|}{\sigma}}$	$\frac{1}{r\sigma\sqrt{(2\pi)^3}}\, e^{-\frac{e^2(\mathbf{x};r)}{2\sigma^2}}$
	• mode inside the circle • models the inside well • tends to normal distribution	• peak at the center • unusable for small radii • tends to Dirac distribution	• mode at the circle • hole at the center • tends to normal distribution

▸Sampson Error for Fundamental Matrix Manifold

The (signed) epipolar algebraic error is — assuming finite points

$$\varepsilon_i(\mathbf{F}) = \underline{\mathbf{y}}_i^\top \mathbf{F}\, \underline{\mathbf{x}}_i, \quad \underline{\mathbf{x}}_i = (u_i^1,\, v_i^1,\, 1), \quad \underline{\mathbf{y}}_i = (u_i^2,\, v_i^2,\, 1), \qquad \varepsilon_i \in \mathbb{R}$$

Let $\mathbf{F} = \begin{bmatrix}\mathbf{F}_1 & \mathbf{F}_2 & \mathbf{F}_3\end{bmatrix}$ (per columns) $= \begin{bmatrix}(\mathbf{F}^1)^\top \\ (\mathbf{F}^2)^\top \\ (\mathbf{F}^3)^\top\end{bmatrix}$ (per rows), $\mathbf{S} = \begin{bmatrix}1 & 0 & 0 \\ 0 & 1 & 0\end{bmatrix}$, then

Sampson

$$\mathbf{J}_i(\mathbf{F}) = \left[\frac{\partial \varepsilon_i(\mathbf{F})}{\partial u_i^1}, \frac{\partial \varepsilon_i(\mathbf{F})}{\partial v_i^1}, \frac{\partial \varepsilon_i(\mathbf{F})}{\partial u_i^2}, \frac{\partial \varepsilon_i(\mathbf{F})}{\partial v_i^2}\right] \qquad \mathbf{J}_i \in \mathbb{R}^{1,4}$$

derivatives over point coordinates

$$= \left[(\mathbf{F}_1)^\top \underline{\mathbf{y}}_i,\ (\mathbf{F}_2)^\top \underline{\mathbf{y}}_i,\ (\mathbf{F}^1)^\top \underline{\mathbf{x}}_i,\ (\mathbf{F}^2)^\top \underline{\mathbf{x}}_i\right] = \begin{bmatrix}\mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i \\ \mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i\end{bmatrix}^\top$$

$$\boxed{\mathbf{e}_i(\mathbf{F}) = -\frac{\mathbf{J}_i^\top(\mathbf{F})\, \varepsilon_i(\mathbf{F})}{\|\mathbf{J}_i(\mathbf{F})\|^2}} \qquad \mathbf{e}_i(\mathbf{F}) \in \mathbb{R}^4$$

Sampson error vector

$$e_i(\mathbf{F}) \overset{\text{def}}{=} \|\mathbf{e}_i(\mathbf{F})\| = \frac{\varepsilon_i(\mathbf{F})}{\|\mathbf{J}_i(\mathbf{F})\|} = \frac{\left|\underline{\mathbf{y}}_i^\top \mathbf{F} \underline{\mathbf{x}}_i\right|}{\sqrt{\|\mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i\|^2 + \|\mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i\|^2}} \qquad e_i(\mathbf{F}) \in \mathbb{R}$$

scalar Sampson error

$F \to \lambda F \quad \lambda \neq 0$

$L = \sum_{i=1}^{k} e_i^2(F)$

- generalization for infinite points is easy
- Sampson error 'normalizes' the algebraic error
- automatically copes with multiplicative factors $\mathbf{F} \mapsto \lambda\mathbf{F}$
- the actual optimization not yet covered →112

▶Sampson Error for Triangulation: The Golden Standard Triangulation Method

Given $\mathbf{P}_1$, $\mathbf{P}_2$ and a correspondence $x \leftrightarrow y$, look for 3D point $\mathbf{X}$ projecting to x and y →90

Idea:

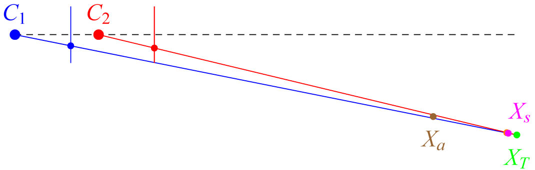
1. if not given, compute $\mathbf{F}$ from $\mathbf{P}_1$, $\mathbf{P}_2$, e.g. $\mathbf{F} = (\mathbf{Q}_1\mathbf{Q}_2^{-1})^\top[\mathbf{q}_1 - (\mathbf{Q}_1\mathbf{Q}_2^{-1})\mathbf{q}_2]_\times$ →77
2. correct the measurement by the linear estimate of the correction vector $\mathbf{e}_i(\mathbf{F})$ →103

$$\begin{bmatrix}\hat{u}^1\\ \hat{v}^1\\ \hat{u}^2\\ \hat{v}^2\end{bmatrix} \approx \begin{bmatrix}u^1\\ v^1\\ u^2\\ v^2\end{bmatrix} \underbrace{-\frac{\varepsilon}{\|\mathbf{J}\|^2}\mathbf{J}^\top}_{\mathbf{e}_i(\mathbf{F})} = \begin{bmatrix}u^1\\ v^1\\ u^2\\ v^2\end{bmatrix} - \frac{\underline{\mathbf{y}}^\top \mathbf{F}\underline{\mathbf{x}}}{\|\mathbf{S}\mathbf{F}\underline{\mathbf{x}}\|^2 + \|\mathbf{S}\mathbf{F}^\top\underline{\mathbf{y}}\|^2} \begin{bmatrix}(\mathbf{F}_1)^\top\underline{\mathbf{y}}\\ (\mathbf{F}_2)^\top\underline{\mathbf{y}}\\ (\mathbf{F}^1)^\top\underline{\mathbf{x}}\\ (\mathbf{F}^2)^\top\underline{\mathbf{x}}\end{bmatrix}$$

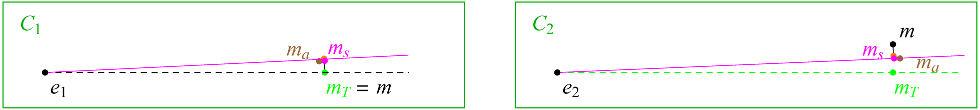
3. use the SVD triangulation algorithm with numerical conditioning →91

THM

Ex (cont'd from →97):



X_T – noiseless ground truth position
● – reprojection error minimizer
X_s – Sampson-corrected algebraic error minimizer
X_a – algebraic error minimizer
m – measurement (m_T with noise in v^2)



▶Back to Fundamental Matrix Estimation

Goal: Given a set $X = \{(x_i, y_i)\}_{i=1}^k$ of $k \gg 7$ inlier correspondences, compute a statistically efficient estimate for fundamental matrix $\mathbf{F}$ (or essential matrix $\mathbf{E}$).

What we have so far

- 7-point algorithm for $\mathbf{F}$ (5-point algorithm for $\mathbf{E}$) $\rightarrow$85
- definition of Sampson error per correspondence $e_i(\mathbf{F} \mid x_i, y_i)$ $\rightarrow$107
- triangulation requiring an optimal $\mathbf{F}$

What we need

- correspondence recognition see later $\rightarrow$116
- an optimization algorithm for many $(k \gg 7)$ correspondences comes next

$$\mathbf{F}^* = \arg\min_{\mathbf{F}} \sum_{i=1}^{k} e_i^2(\mathbf{F} \mid X)$$

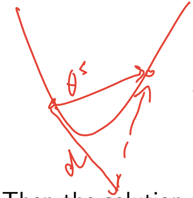
- the 7-point estimate is a good starting point $\mathbf{F}_0$

Consider error function $\mathbf{e}_i(\boldsymbol{\theta}) = f(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\theta}) \in \mathbb{R}^m$, with $\mathbf{x}_i, \mathbf{y}_i$ given, $\boldsymbol{\theta} \in \mathbb{R}^q$ unknown

$\theta = \mathbf{F}$, $q = 9$, $m = 1$ for f.m. estimation

Our goal: $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{k} \|\mathbf{e}_i(\boldsymbol{\theta})\|^2$

Idea 1 (Gauss-Newton approximation): proceed iteratively for $s = 0, 1, 2, \ldots$



$$\boldsymbol{\theta}^{s+1} := \boldsymbol{\theta}^s + \lambda\mathbf{d}_s, \quad \text{where} \quad \mathbf{d}_s = \arg\min_{\mathbf{d}} \sum_{i=1}^{k} \|\mathbf{e}_i(\boldsymbol{\theta}^s + \mathbf{d})\|^2 \tag{21}$$

$$\mathbf{e}_i(\boldsymbol{\theta}^s + \mathbf{d}) \approx \mathbf{e}_i(\boldsymbol{\theta}^s) + \mathbf{L}_i\,\mathbf{d},$$

$$(\mathbf{L}_i)_{jl} = \frac{\partial\big(\mathbf{e}_i(\boldsymbol{\theta})\big)_j}{\partial(\boldsymbol{\theta})_l}, \qquad \mathbf{L}_i \in \mathbb{R}^{m,q}$$

typically a 'long' matrix, $m \ll q$

Then the solution to Problem (21) is a set of 'normal eqs'

$$-\underbrace{\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{e}_i(\boldsymbol{\theta}^s)}_{\mathbf{e}\in\mathbb{R}^{q,1}} = \underbrace{\left(\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{L}_i\right)}_{\mathbf{L}\in\mathbb{R}^{q,q}} \mathbf{d}_s, \tag{22}$$

- $\mathbf{d}_s$ can be solved for by Gaussian elimination using Choleski decomposition of $\mathbf{L}$

  $\mathbf{L}$ (large) symmetric PSD $\Rightarrow$ use Choleski, almost $2\times$ faster than Gauss-Seidel, see bundle adjustment $\rightarrow$144
- beware of rank defficiency in $\mathbf{L}$ when k is small
- such updates do not lead to stable convergence $\longrightarrow$ ideas of Levenberg and Marquardt

LM (cont'd)

Idea 2 (Levenberg): replace $\sum_i \mathbf{L}_i^\top \mathbf{L}_i$ with $\sum_i \mathbf{L}_i^\top \mathbf{L}_i + \lambda\,\mathbf{I}$ for some damping factor $\lambda \geq 0$

Idea 3 (Marquardt): replace $\lambda\,\mathbf{I}$ with $\lambda \sum_i \operatorname{diag}(\mathbf{L}_i^\top \mathbf{L}_i)$ to adapt to local curvature:

$$-\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{e}_i(\boldsymbol{\theta}^s) = \left(\sum_{i=1}^{k} \left(\mathbf{L}_i^\top \mathbf{L}_i + \lambda \operatorname{diag}(\mathbf{L}_i^\top \mathbf{L}_i) \right) \right) \mathbf{d}_s$$

Idea 4 (Marquardt): adaptive λ small $\lambda \to$ Gauss-Newton, large $\lambda \to$ gradient descend

1. choose $\lambda \approx 10^{-3}$ and compute $\mathbf{d}_s$
2. if $\sum_i \|\mathbf{e}_i(\boldsymbol{\theta}^s + \mathbf{d}_s)\|^2 < \sum_i \|\mathbf{e}_i(\boldsymbol{\theta}^s)\|^2$ then accept $\mathbf{d}_s$ and set $\lambda := \lambda/10$, $s := s+1$ better: Armijo's rule
3. otherwise set $\lambda := 10\lambda$ and recompute $\mathbf{d}_s$

- sometimes different constants are needed for the 10 and 10^{-3}
- note that $\mathbf{L}_i \in \mathbb{R}^{m,q}$ (long matrix) but each contribution $\mathbf{L}_i^\top \mathbf{L}_i$ is a square singular $q \times q$ matrix (always singular for $k < q$)
- λ helps avoid the consequences of gauge freedom →146
- the error function can be made robust to outliers →117
- we have approximated the least squares Hessian by ignoring second derivatives of the error function (Gauss-Newton approximation) See [Triggs et al. 1999, Sec. 4.3]
- a good book on convex optimization: [Boyd and Vandenberghe(2009)]

LM with Sampson Error for Fundamental Matrix Estimation

Sampson (derived by linearization over point coordinates u^1, v^1, u^2, v^2)

$$e_i(\mathbf{F}) = \frac{\varepsilon_i}{\|\mathbf{J}_i\|} = \frac{\underline{\mathbf{y}}_i^\top \mathbf{F} \underline{\mathbf{x}}_i}{\sqrt{\|\mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i\|^2 + \|\mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i\|^2}} \quad \text{where} \quad \mathbf{S} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

LM (by linearization over parameters $\mathbf{F}$)

$$\mathbf{L}_i = \frac{\partial e_i(\mathbf{F})}{\partial \mathbf{F}} = \cdots = \frac{1}{2\|\mathbf{J}_i\|} \left[\left(\underline{\mathbf{y}}_i - \frac{2e_i(\mathbf{F})}{\|\mathbf{J}_i\|} \mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i \right) \underline{\mathbf{x}}_i^\top + \underline{\mathbf{y}}_i \left(\underline{\mathbf{x}}_i - \frac{2e_i(\mathbf{F})}{\|\mathbf{J}_i\|} \mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i \right)^\top \right] \tag{23}$$

- $\mathbf{L}_i$ in (23) is a 3×3 matrix, must be reshaped to dimension-9 vector $\mathrm{vec}(\mathbf{L}_i)$ to be used in LM
- $\underline{\mathbf{x}}_i$ and $\underline{\mathbf{y}}_i$ in Sampson error are normalized to unit homogeneous coordinate (23) relies on this
- reinforce $\operatorname{rank} \mathbf{F} = 2$ after each LM update to stay on the fundamental matrix manifold and $\|\mathbf{F}\| = 1$ to avoid gauge freedom by SVD →113
- LM linearization could be done by numerical differentiation (we can afford it, we have a small dimension here)

▶Local Optimization for Fundamental Matrix Estimation

Summary so far

- Given a set $X = \{(x_i, y_i)\}_{i=1}^k$ of $k \gg 7$ inlier correspondences, compute a statistically efficient estimate for fundamental matrix $\mathbf{F}$.
  1. Find the conditioned (→93) 7-point $\mathbf{F}_0$ (→85) from a suitable 7-tuple
  2. Improve the $\mathbf{F}_0^*$ using the LM optimization (→110–111) and the Sampson error (→112) on all inliers, reinforce rank-2, unit-norm $\mathbf{F}_k^*$ after each LM iteration using SVD

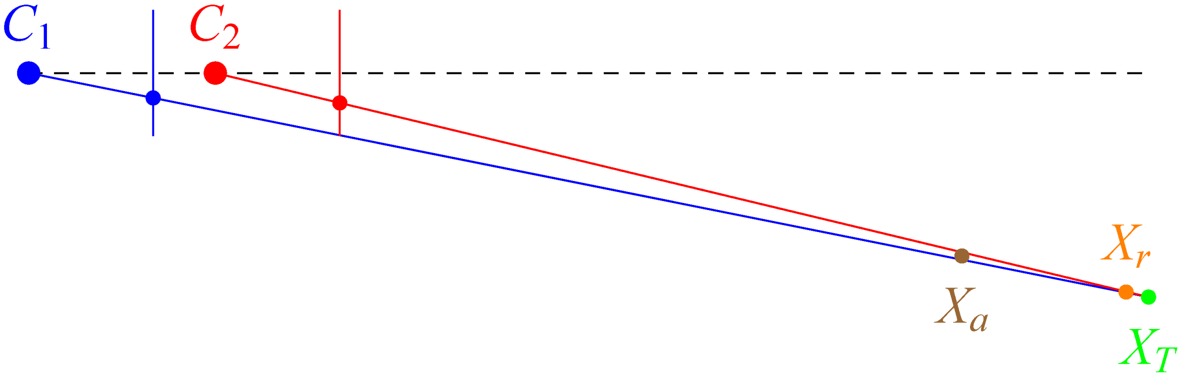
Partial conceptualization

- inlier = a correspondence (a true match)
- outlier = a non-correspondence
- binary inlier/outlier labels are hidden
- we can get their likely estimate only, with respect to a model

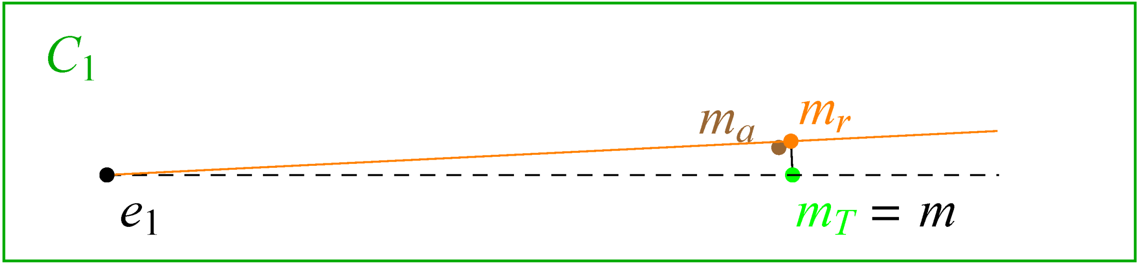
We are not yet done

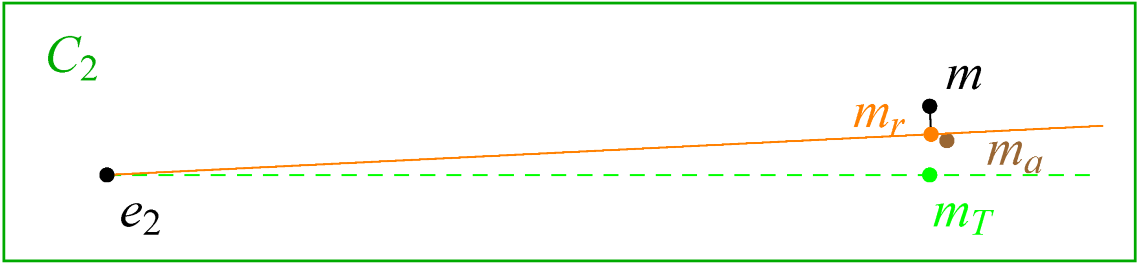
- if there are no wrong correspondences (mismatches, outliers), this gives a local optimum given the 7-point initial estimate
- the algorithm breaks under contamination of (inlier) correspondences by outliers
- the full problem involves finding the inliers!
- in addition, we need a mechanism for jumping out of local minima (and exploring the space of all fundamental matrices)

Thank You

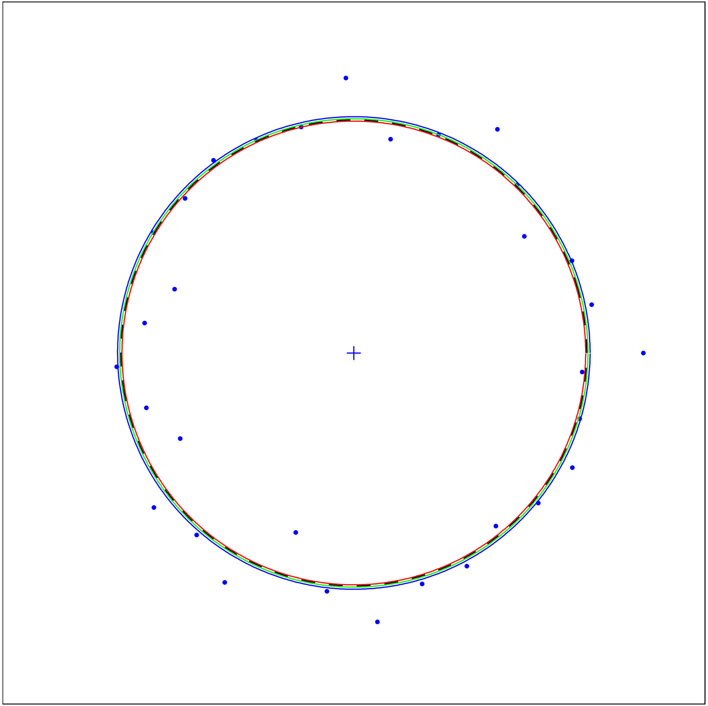


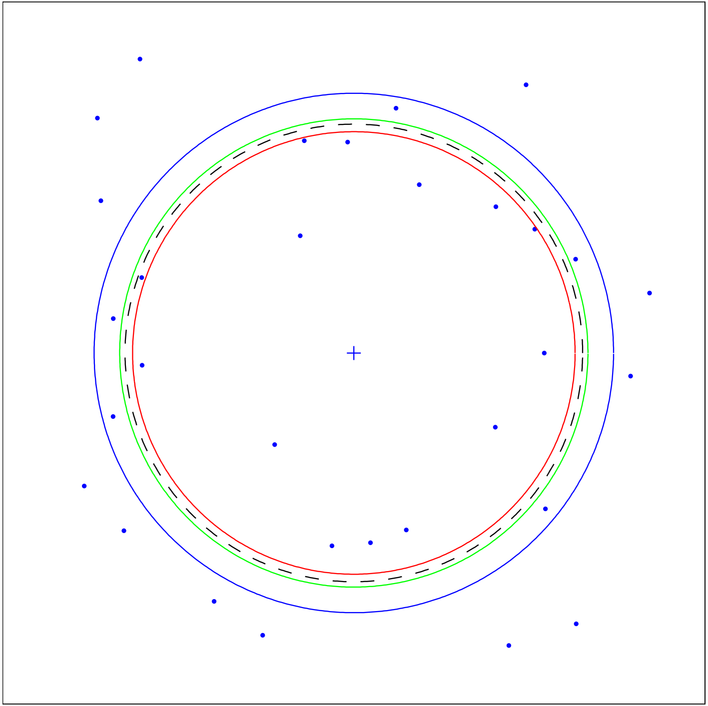
C_1
C_2
X_r
X_a
X_T

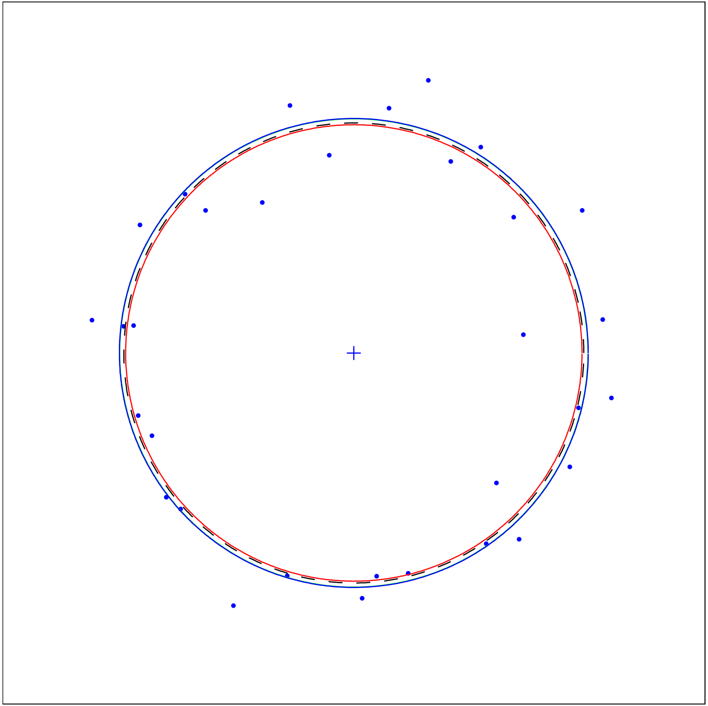

C_1
m_a
m_r
e_1
$m_T = m$

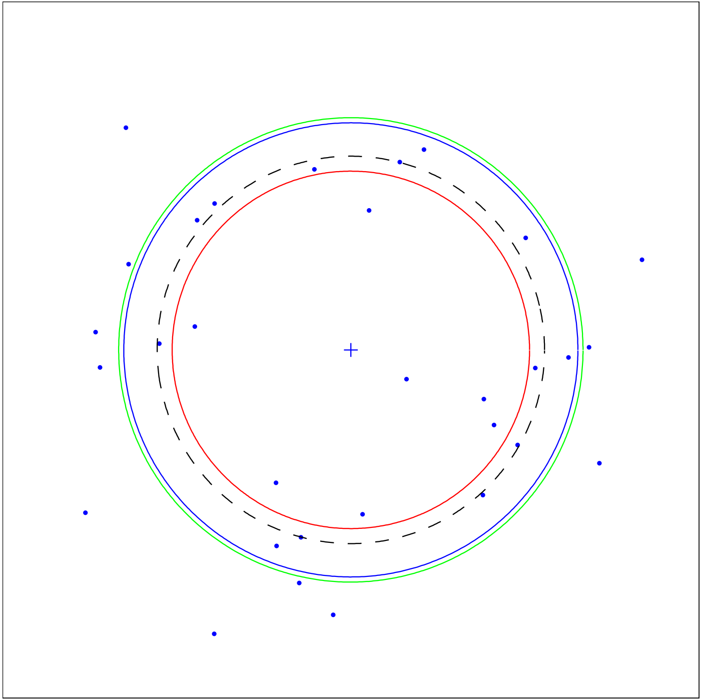


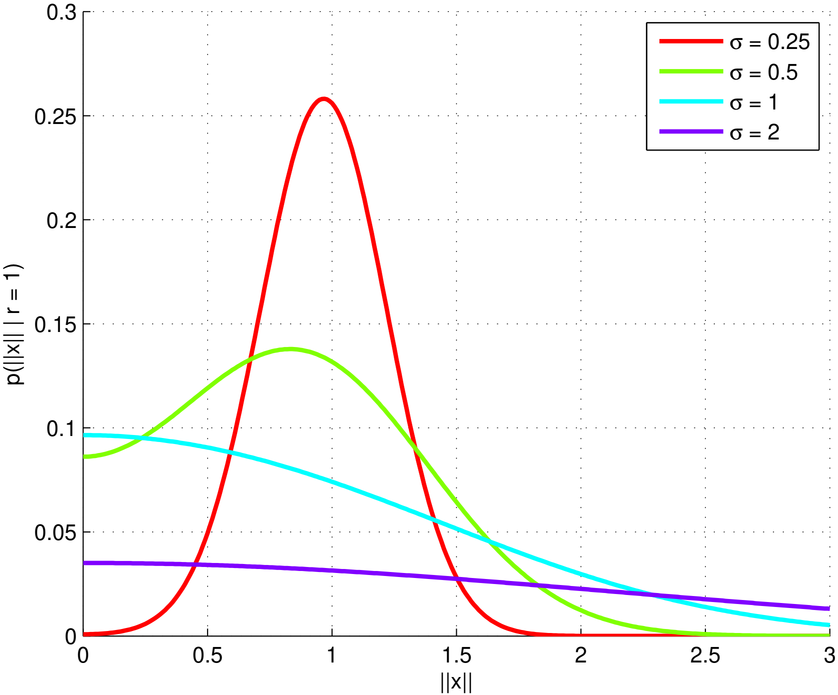
C_2
m
m_r
m_a
e_2
m_T

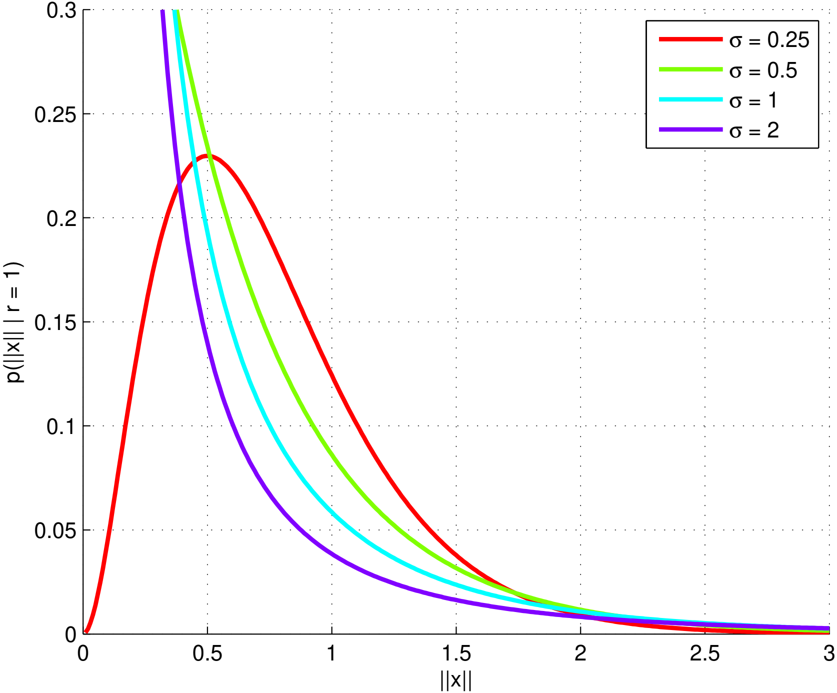


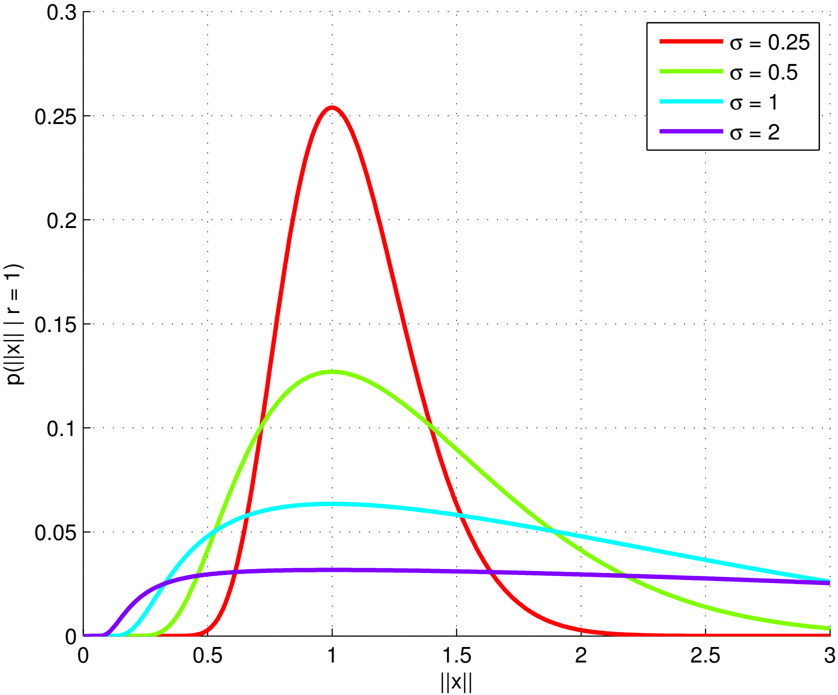


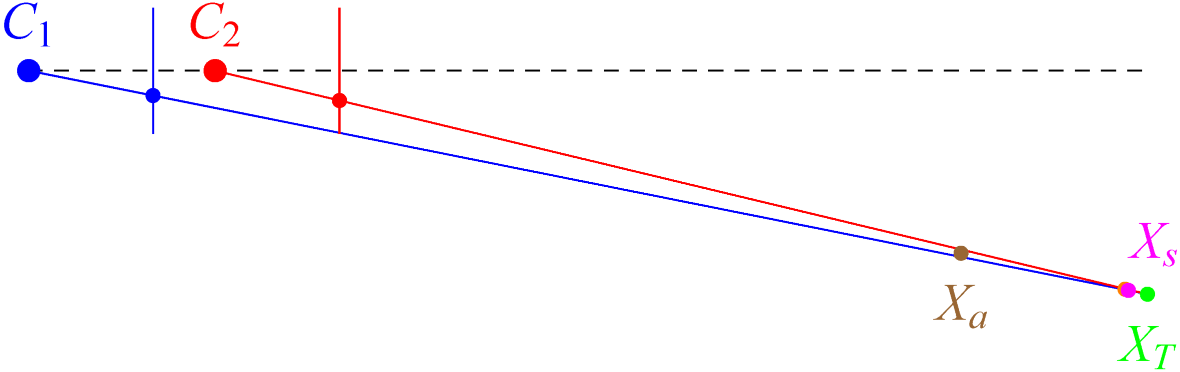


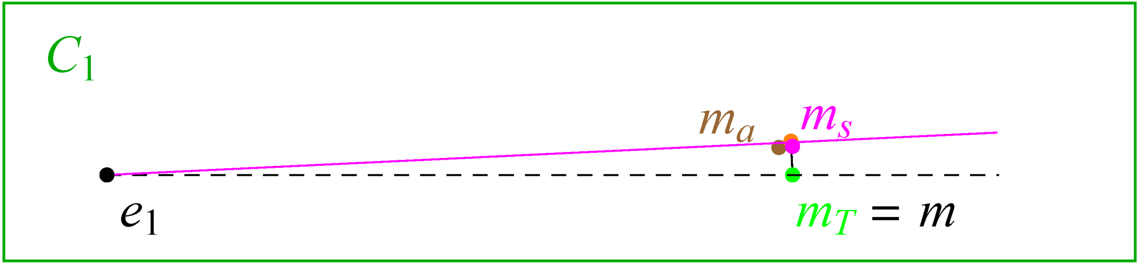



σ = 0.25
σ = 0.5
σ = 1
σ = 2
p(||x|| | r = 1)
||x||
0
0.05
0.1
0.15
0.2
0.25
0.3
0
0.5
1
1.5
2
2.5
3

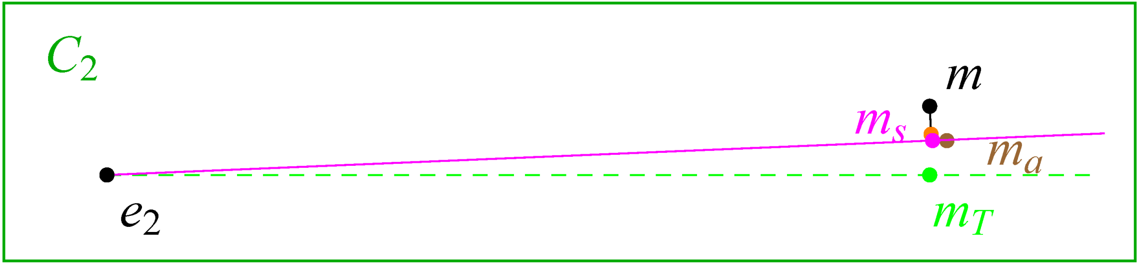

σ = 0.25
σ = 0.5
σ = 1
σ = 2
0.3
0.25
0.2
0.15
0.1
0.05
0
p(||x|| | r = 1)
0
0.5
1
1.5
2
2.5
3
||x||


σ = 0.25
σ = 0.5
σ = 1
σ = 2
p(||x|| | r = 1)
||x||
0
0.05
0.1
0.15
0.2
0.25
0.3
0.5
1
1.5
2
2.5
3


C_1
C_2
X_s
X_a
X_T



C_1
m_a
m_s
e_1
$m_T = m$


C_2
m
m_s
m_a
e_2
m_T