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 3, 2020

Open Informatics Master's Course

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.

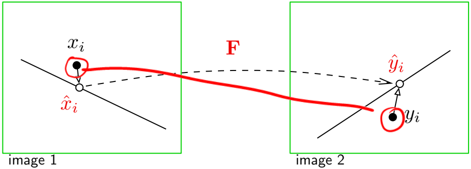
O. Chum, J. Matas, and J. Kittler. Locally optimized RANSAC. In *Proc DAGM*, LNCS 2781:236–243. Springer-Verlag, 2003.

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.

▶The Concept of Error for Epipolar Geometry

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 $x_i \leftrightarrow y_j$.

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



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

$$\mathbf{e}_i(x_i, y_i \mid \hat{x}_i, \hat{y}_i, \mathbf{F}) = \begin{bmatrix} \mathbf{x}_i - \hat{\mathbf{x}}_i \\ \mathbf{y}_i - \hat{\mathbf{y}}_i \end{bmatrix} = \mathbf{e}_i(\mathbf{Z}_i \mid \hat{\mathbf{Z}}_i, \mathbf{F}) = \mathbf{Z}_i - \hat{\mathbf{Z}}_i(\mathbf{F}) \tag{15}$$

$$\|\mathbf{e}_i(\cdot)\|^2 \stackrel{\text{def}}{=} \mathbf{e}_i^2(\cdot) = \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2 + \|\mathbf{y}_i - \hat{\mathbf{y}}_i\|^2 = \|\mathbf{Z}_i - \hat{\mathbf{Z}}_i(\mathbf{F})\|^2$$

►cont'd

- the total reprojection error (of all data) then is

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

- and the optimization problem is

$$(\hat{Z}^*, \mathbf{F}^*) = \arg \min_{\substack{\mathbf{F} \\ \operatorname{rank} \mathbf{F} = 2 \\ \text{homo}}} \; \min_{\substack{\hat{Z} \\ \underline{\hat{\mathbf{y}}}_i^\top \mathbf{F}\, \underline{\hat{\mathbf{x}}}_i = 0}} \sum_{i=1}^{k} \mathbf{e}_i^2(x_i, y_i \mid \underbrace{\hat{x}_i, \hat{y}_i}_{\hat{z}_i}, \mathbf{F}) \tag{16}$$

Three 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 $\hat{Z}$, $\mathbf{F}$ →98
  2. Sampson optimal correction = partial correction of $\mathbf{Z}_i$ towards $\hat{\mathbf{Z}}_i$ used in an iterative minimization over $\mathbf{F}$ →99

  ~~3.~~ removing $\hat{x}_i$, $\hat{y}_i$ altogether = marginalization of $L(Z, \hat{Z} \mid \mathbf{F})$ over $\hat{Z}$ followed by minimization over $\mathbf{F}$ not covered, the marginalization is difficult

Method 1: Reprojection Error Optimization

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

$$\mathbf{P}_1 = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} [\underline{\mathbf{e}}_2]_\times \mathbf{F} + \underline{\mathbf{e}}_2 \underline{\mathbf{e}}_1^\top & \underline{\mathbf{e}}_2 \end{bmatrix} \tag{17}$$

⊛ H3; 2pt: Given $\mathbf{F}$, let $\underline{\mathbf{e}}_1$, $\underline{\mathbf{e}}_2$ be the left and right nullspace basis vectors of $\mathbf{F}$ (i.e. the epipoles). Verify that $\mathbf{F}$ is a fundamental matrix of $\mathbf{P}_1$, $\mathbf{P}_2$ from (17). Hint: $\mathbf{A}$ is skew symmetric iff $\mathbf{x}^\top \mathbf{A} \mathbf{x} = 0$ for all vectors $\mathbf{x}$.

---

1. compute $\mathbf{F}^{(0)}$ by the 7-point algorithm →84; construct camera $\mathbf{P}_2^{(0)}$ from $\mathbf{F}^{(0)}$ using (17)
2. triangulate 3D points $\hat{\mathbf{X}}_i^{(0)}$ from matches (x_i, y_i) for all $i = 1, \dots, k$ →89
3. starting from $\mathbf{P}_2^{(0)}$, $\hat{\mathbf{X}}^{(0)}$ minimize the reprojection error (15)

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

where

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

Non-linear, non-convex problem

4. compute $\mathbf{F}$ from $\mathbf{P}_1$, $\mathbf{P}_2^*$

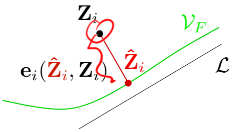
- $3k + 12$ parameters to be found: latent: $\hat{\mathbf{X}}_i$, for all i (correspondences!), non-latent: $\mathbf{P}_2$
- minimal representation: $3k + 7$ parameters, $\mathbf{P}_2 = \mathbf{P}_2(\mathbf{F})$ →145
- there are pitfalls; this is essentially bundle adjustment; we will return to this later →136

▶Method 2: First-Order Error Approximation

An elegant method for solving problems like (16):

- 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 $\boxed{\boldsymbol{\varepsilon}_i = \underline{\mathbf{y}}_i^\top \mathbf{F}\, \underline{\mathbf{x}}_i}$ from →84

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



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 $\mathcal{L}$.

$$0 = \underline{\boldsymbol{\varepsilon}_i(\hat{\mathbf{Z}}_i)} \approx \underbrace{\boldsymbol{\varepsilon}_i(\mathbf{Z}_i) + \frac{\partial \boldsymbol{\varepsilon}_i(\mathbf{Z}_i)}{\partial \mathbf{Z}_i} \overbrace{(\hat{\mathbf{Z}}_i - \mathbf{Z}_i)}^{e_i}}$$

▸Sampson's Approximation of Reprojection Error

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

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

- 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 $\mathbf{e}_i(\cdot)$
- 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 $\mathbf{e}_i^*(\cdot)$

$$\begin{aligned} \mathbf{e}_i^*(\cdot) &= -\mathbf{J}_i^\top (\mathbf{J}_i \mathbf{J}_i^\top)^{-1} \boldsymbol{\varepsilon}_i(\cdot) \qquad \text{pseudo-inverse} \\ \|\mathbf{e}_i^*(\cdot)\|^2 &= \boldsymbol{\varepsilon}_i^\top(\cdot) (\mathbf{J}_i \mathbf{J}_i^\top)^{-1} \boldsymbol{\varepsilon}_i(\cdot) \end{aligned} \qquad (18)$$

- this maps $\boldsymbol{\varepsilon}_i(\cdot)$ to an estimate of $\mathbf{e}_i(\cdot)$ per correspondence
- we often do not need $\mathbf{e}_i$, just $\|\mathbf{e}_i\|^2$ exception: triangulation →105
- 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*

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

$\|x\|^2 - r^2$

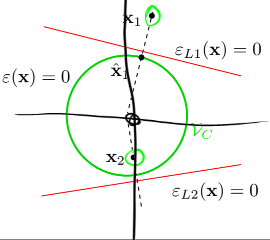
$\boldsymbol{\varepsilon}_L(\hat{\mathbf{x}}) = 0$ is a line with normal $\frac{\mathbf{x}}{\|\mathbf{x}\|}$ and intercept $\frac{r^2 + \|\mathbf{x}\|^2}{2\|\mathbf{x}\|}$ — not tangent to $\mathcal{C}$, outside!

3. using (18), express error approximation $\mathbf{e}^*$ as

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

1

4. fit circle



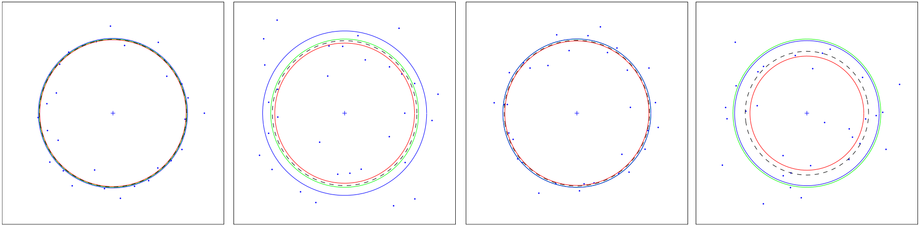
$$r^* = \arg\min_r \sum_{i=1}^k \frac{(\|\mathbf{x}_i\|^2 - r^2)^2}{4\|\mathbf{x}_i\|^2} = \cdots = \left(\frac{1}{k} \sum_{i=1}^k \frac{1}{\|\mathbf{x}_i\|^2}\right)^{\frac{1}{2}}$$

- this example results in a convex quadratic optimization problem
- note that

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

Circle Fitting: Some Results

medium radial noise | big radial noise | medium isotropic noise | big isotropic noise



opt: 1.8, Smp: 1.9, dir: 2.3 | 1.6, 1.8, 2.6 | 1.8, 2.0, 2.2 | 1.6, 2.0, 2.4

mean ranks over 10 000 random trials with $k = 32$ samples

green – ground truth

red – Sampson error $\mathbf{e}$ minimizer

blue – direct radial error $\boldsymbol{\varepsilon}$ minimizer

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
- 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...

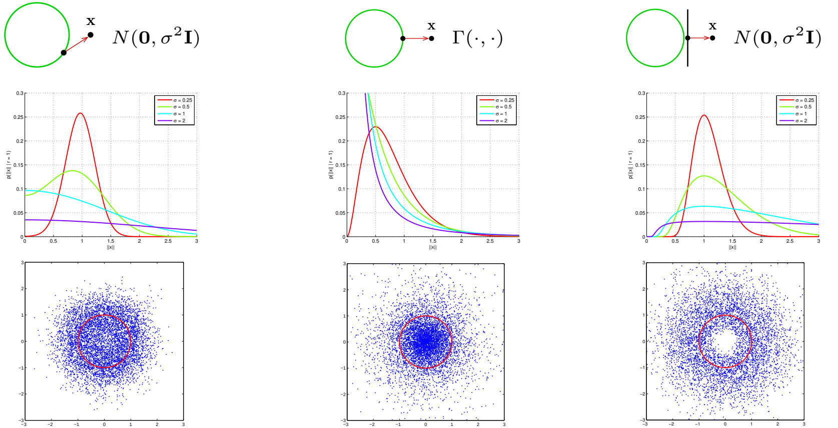
- a few models for fitting zero-centered circle C of radius r to points in $\mathbb{R}^2$

marginalized over C | orthogonal deviation from C | Sampson approximation

error model

radial p.d.f.

random sample



$p(\mathbf{x} \mid r)$

marginalized over C:

$$\approx \frac{1}{\sigma\sqrt{(2\pi)^3 r\,\|\mathbf{x}\|}}\, e^{-\frac{(\|\mathbf{x}\|-r)^2}{2\sigma^2}}$$

- mode inside the circle
- models the inside well
- tends to normal distrib.

orthogonal deviation from C:

$$\frac{1}{2\pi\Gamma(\frac{r^2}{\sigma})}\,\frac{1}{\|\mathbf{x}\|^2}\left(\frac{r\|\mathbf{x}\|}{\sigma}\right)^{\frac{r^2}{\sigma}} e^{-\frac{r\|\mathbf{x}\|}{\sigma}}$$

- peak at the center
- unusable for small radii
- tends to Dirac distrib.

Sampson approximation:

$$\frac{1}{r\sigma\sqrt{(2\pi)^3}}\, e^{-\frac{e^2(\mathbf{x};r)}{2\sigma^2}}$$

- mode at the circle
- hole at the center
- tends to normal distrib.

▶Sampson Error for Fundamental Matrix Manifold

The epipolar algebraic error is

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

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

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

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

Sampson error vector

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

scalar Sampson error

- Sampson error 'normalizes' the algebraic error
- automatically copes with multiplicative factors $\mathbf{F} \mapsto \lambda \mathbf{F}$
- actual optimization not yet covered →109

▶Back to Triangulation: The Golden Standard 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 →89

Idea:

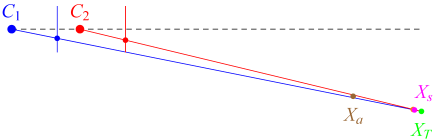
1. if not given, compute $\mathbf{F}$ from $\mathbf{P}_1$, $\mathbf{P}_2$, e.g. $\mathbf{F} = (\mathbf{Q}_1\mathbf{Q}_2^{-1})^\top [\mathbf{q}_1 - (\mathbf{Q}_1\mathbf{Q}_2^{-1})\mathbf{q}_2]_\times$
2. correct the measurement by the linear estimate of the correction vector →100

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

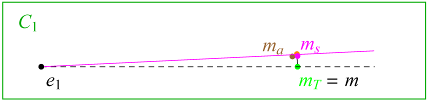
$\hat{z}$ $\qquad z$

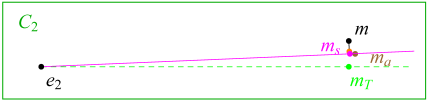
3. use the SVD triangulation algorithm with numerical conditioning →90

Ex (cont'd from →93):



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 $\mathbf{F}$.

What we have so far

- 7-point algorithm for $\mathbf{F}$ (5-point algorithm for $\mathbf{E}$) →84
- definition of Sampson error per correspondence $e_i(\mathbf{F} \mid x_i, y_i)$ →104
- triangulation requiring an optimal $\mathbf{F}$

What we need

- an optimization algorithm for

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

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

Levenberg-Marquardt (LM) Iterative Estimation in a Nutshell

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 $s = 0, 1, 2, \ldots$

$$\boldsymbol{\theta}^{s+1} := \boldsymbol{\theta}^s + \mathbf{d}_s\,, \quad \text{where} \quad \mathbf{d}_s = \arg\min_{\mathbf{d}} \sum_{i=1}^{k} \|\mathbf{e}_i(\boldsymbol{\theta}^s + \mathbf{d})\|^2 \tag{19}$$

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

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

Then the solution to Problem (19) is a set of 'normal' eqs

$$-\underbrace{\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{e}_i(\boldsymbol{\theta}^s)}_{\mathbf{e}\in\mathbb{R}^{q,1}} = \underbrace{\left(\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{L}_i\right)}_{\mathbf{L}\in\mathbb{R}^{q,q}} \mathbf{d}_s, \tag{20}$$

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

  $\mathbf{L}$ symmetric PSD $\Rightarrow$ use Choleski, almost $2\times$ faster than Gauss-Seidel, see bundle adjustment →139
- such updates do not lead to stable convergence ⟶ ideas of Levenberg and Marquardt

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 $\lambda := \lambda/10$, $s := s + 1$
3. otherwise set $\lambda := 10\lambda$ and recompute $\mathbf{d}_s$

$F \mapsto F + D_s \xmapsto{\text{rank 2 projection}}$

- 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 $k < q$)
- error can be made robust to outliers, see the trick $\to$112
- 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]
- λ helps avoid the consequences of gauge freedom $\to$141
- modern variants of LM are Trust Region methods

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 $\mathbf{F}$)

$$\mathbf{L}_i = \frac{\partial e_i(\mathbf{F})}{\partial \mathbf{F}} = \cdots = \frac{1}{2\|\mathbf{J}_i\|} \left[\left(\underline{\mathbf{y}}_i - \frac{2e_i}{\|\mathbf{J}_i\|} \mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i \right) \underline{\mathbf{x}}_i^\top + \underline{\mathbf{y}}_i \left(\underline{\mathbf{x}}_i - \frac{2e_i}{\|\mathbf{J}_i\|} \mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i \right)^\top \right] \tag{21}$$

- $\mathbf{L}_i$ in (21) is a 3×3 matrix, must be reshaped to dimension-9 vector $\operatorname{vec}(\mathbf{L}_i)$ to be used in LM
- $\underline{\mathbf{x}}_i$ and $\underline{\mathbf{y}}_i$ in Sampson error are normalized to unit homogeneous coordinate (21) relies on this
- reinforce $\operatorname{rank}\mathbf{F} = 2$ after each LM update to stay in the fundamental matrix manifold and $\|\mathbf{F}\| = 1$ to avoid gauge freedom by SVD →110
- LM linearization could be done by numerical differentiation (we have a small dimension here)

►Local Optimization for Fundamental Matrix Estimation

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

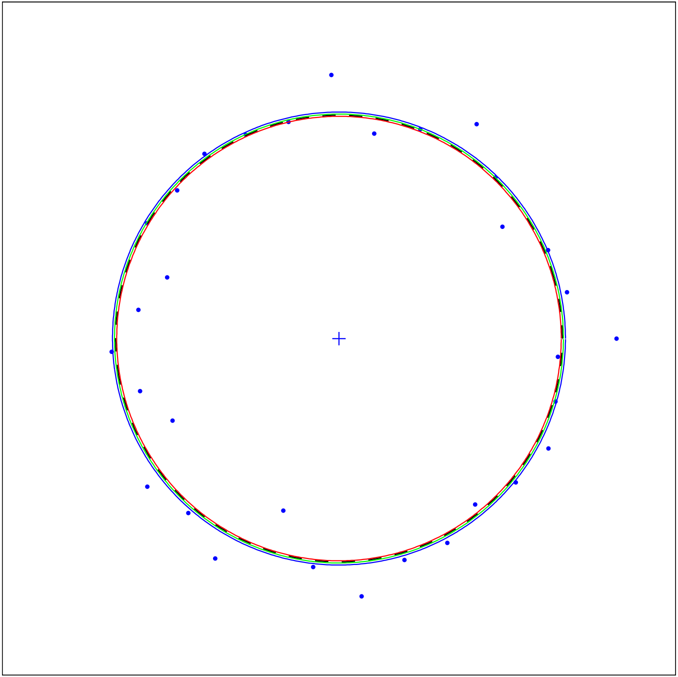
Summary so far

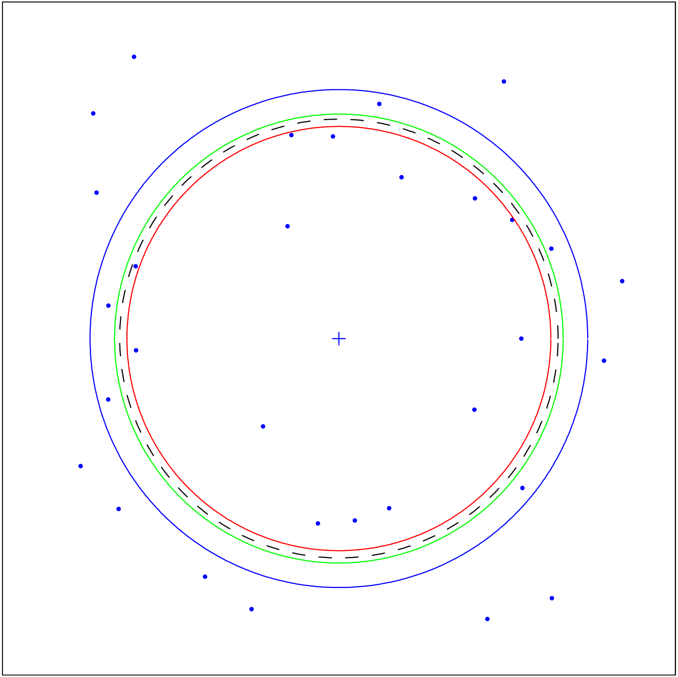
1. Find the conditioned (→92) 7-point $\mathbf{F}_0$ (→84) from a suitable 7-tuple
2. Improve the $\mathbf{F}_0^*$ using the LM optimization (→107–108) and the Sampson error (→109) on all inliers, reinforce rank-2, unit-norm $\mathbf{F}_k^*$ after each LM iteration using SVD

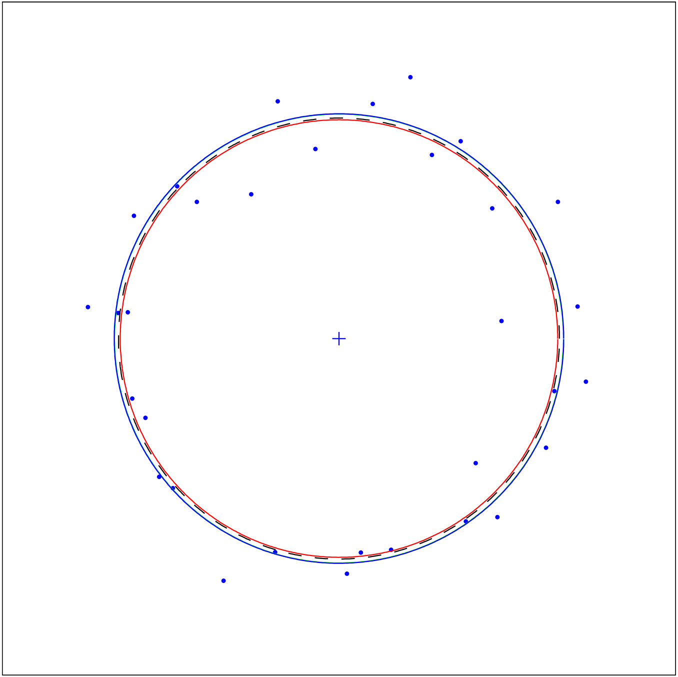
We are not yet done

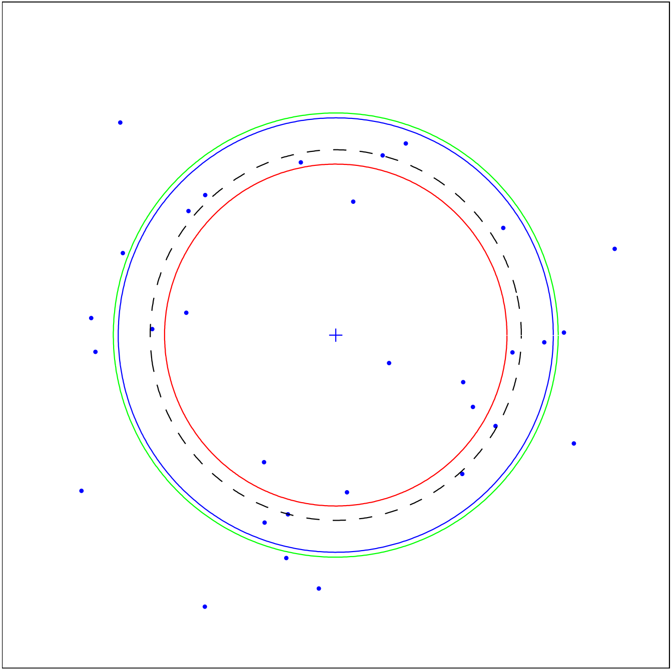
- if there are no wrong correspondences (mismatches, outliers), this gives a local optimum given the 7-point initial estimate
- the algorithm breaks under contamination of (inlier) correspondences by outliers
- the full problem involves finding the inliers!
- in addition, we need a mechanism for jumping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

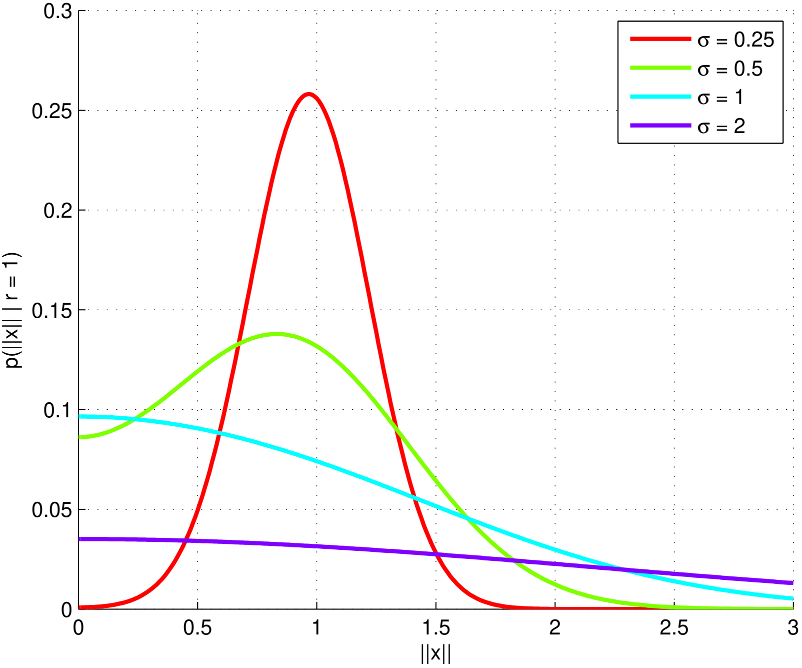
Thank You



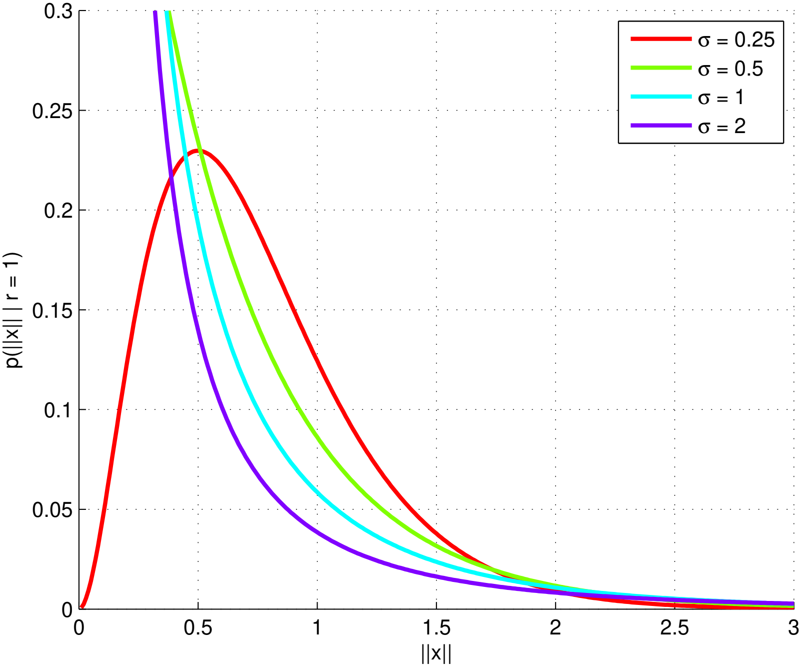


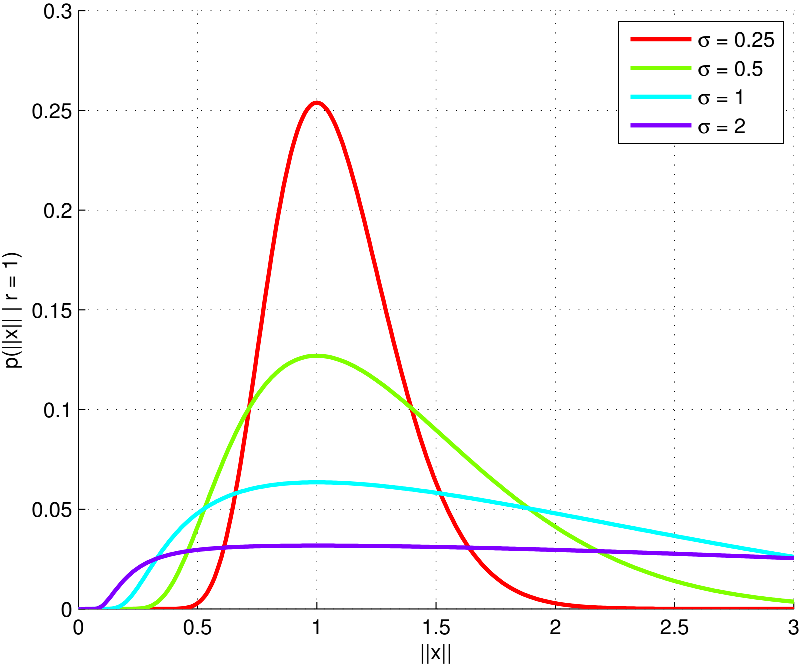


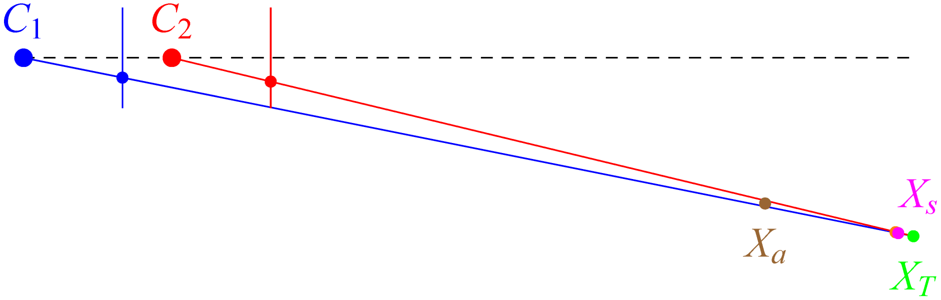


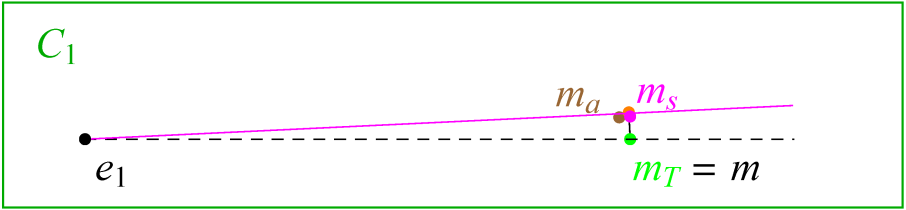


σ = 0.25
σ = 0.5
σ = 1
σ = 2
p(||x|| | r = 1)
||x||
0
0.05
0.1
0.15
0.2
0.25
0.3
0
0.5
1
1.5
2
2.5
3

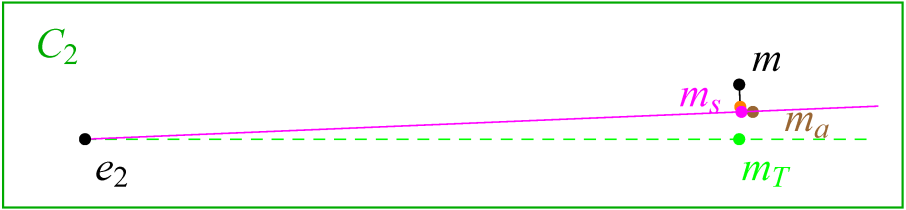

0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.5
1
1.5
2
2.5
3
||x||
p(||x|| | r = 1)
σ = 0.25
σ = 0.5
σ = 1
σ = 2


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


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