

Revisiting Sampson Approximations for Geometric Estimation Problems

Felix Rydell
KTH Royal Institute of Technology
felixry@kth.se

Angélica Torres
Max Planck Institute for Mathematics in the Sciences
angelica.torres@mis.mpg.de

Viktor Larsson
Lund University
viktor.larsson@math.lth.se

Abstract

Many problems in computer vision can be formulated as geometric estimation problems, i.e. given a collection of measurements (e.g. point correspondences) we wish to fit a model (e.g. an essential matrix) that agrees with our observations. This necessitates some measure of how much an observation “agrees” with a given model. A natural choice is to consider the smallest perturbation that makes the observation exactly satisfy the constraints. However, for many problems, this metric is expensive or otherwise intractable to compute. The so-called Sampson error approximates this geometric error through a linearization scheme. For epipolar geometry, the Sampson error is a popular choice and in practice known to yield very tight approximations of the corresponding geometric residual (the reprojection error).

In this paper we revisit the Sampson approximation and provide new theoretical insights as to why and when this approximation works, as well as provide explicit bounds on the tightness under some mild assumptions. Our theoretical results are validated in several experiments on real data and in the context of different geometric estimation tasks.

1. Introduction

Estimating a geometric model from a collection of measurements is a common task in computer vision pipelines. Prerequisite for any such estimation is the ability to check whether a measurement is consistent with a given model. This can be used both to filter outlier measurements or as a loss for non-linear model refinement. For generative models, i.e. models that can produce the idealized measurements, it is straight-forward to check this consistency, e.g. computing the difference between the projection and observed 2D keypoint. However, in many cases the relation between model and data is implicitly encoded through a set of geometric constraints. One such example is the epipolar

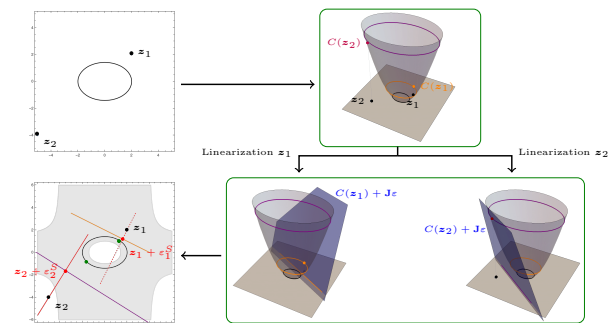


Figure 1. The model $C(x, y) = x^2 + 2y^2 - 4 = 0$ is the ellipse on the top left, and the data points are z_1 and z_2 . On the top right, the gray surface is the graph $(x, y, C(x, y))$, the orange and purple curves are the level sets $C(x, y) = C(z_1)$ and $C(x, y) = C(z_2)$ respectively. In the bottom right the blue planes are tangent to the surface at $(z_1, C(z_1))$ and $(z_2, C(z_2))$ respectively. The orange and purple lines are the linearized constraints for z_1 and z_2 respectively. We keep this color convention for the linearized constraints on the bottom left, and represent their normals in red. The Sampson approximations for z_1 and z_2 are the red points $z_1 + \epsilon_1^S$ and $z_2 + \epsilon_2^S$, and the minimizers of the geometric error are depicted in green. Since z_1 is in the gray region (obtained from Proposition 3.2), its Sampson approximation is better than the approximation for z_2 .

constraint in two-view geometry,

$$C(\mathbf{x}_1, \mathbf{x}_2, \mathbf{E}) = (\mathbf{x}_2; 1)^\top \mathbf{E} (\mathbf{x}_1; 1) = 0 \quad (1)$$

relating the essential (or fundamental) matrix E with the point correspondence $(\mathbf{x}_1, \mathbf{x}_2) \in \mathbb{R}^2 \times \mathbb{R}^2$. For a noisy correspondence the constraint will not be satisfied exactly, so it becomes natural to ask: How close is this match to agreeing with the model? Mathematically, this can be formulated as

$$\min_{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2} \|\hat{\mathbf{x}}_1 - \mathbf{x}_1\|^2 + \|\hat{\mathbf{x}}_2 - \mathbf{x}_2\|^2 \quad (2)$$

$$\text{s.t. } C(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \mathbf{E}) = 0 \quad (3)$$

i.e., what is the smallest perturbation to the points such that they exactly satisfy the constraint. This minimum distance gives a measure of the consistency between the model (the essential matrix E) and the measurements $(\mathbf{x}_1$ and $\mathbf{x}_2)$.

For the specific case of the epipolar constraint, the optimal solution to (2) can be computed [8, 14] by finding the roots to degree 6 univariate polynomial. As this procedure is relatively expensive, and difficult to integrate into non-linear refinement methods, it is common in practice to instead use some approximation of the error in (2). One example of this is the Sampson error [15, 20], defined as

$$\mathcal{E}_S^2 = \frac{((\mathbf{x}_2; 1)^\top \mathbf{E}(\mathbf{x}_1; 1))^2}{\|\mathbf{E}_{12}(\mathbf{x}_1; 1)\|^2 + \|(\mathbf{E}^\top)_{12}(\mathbf{x}_2; 1)\|^2}. \quad (4)$$

This expression is derived by linearizing the constraint in (2), i.e. by replacing $C(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, E) = 0$ by the constant and linear terms of its Taylor expansion around the data point $(\mathbf{x}_1, \mathbf{x}_2)$. It provides a good approximation of (2) if the curvature of the constraint at the data point is small enough in relation to the size of $C(\mathbf{x}_1, \mathbf{x}_2, E)$, while being cheap to compute and easy to optimize in non-linear refinement. This approach was originally used by Sampson [20] to approximate distances between conics and points, but has since been applied to many other geometric models.

In this paper we revisit the Sampson approximation and provide new theoretical insights into why (and when) the approximation works well. Under relatively mild assumptions we derive explicit bounds on the tightness of the approximation. These bounds are experimentally validated on real data and showcased in multiple applications from geometric computer vision (two- and three-view estimation, vanishing point estimation and resectioning).

The paper is organized as follows: Section 1.1 discusses the related work. In Section 2 we first present the classical derivation of the Sampson error, along with some geometrical interpretations of the approximation. In Section 3 we present our bounds on the approximation. Finally, in Section 4 we provide some experimental evaluation.

1.1. Related Work

The Geometric Error. In applied algebraic geometry, fitting noisy data points to a mathematical model defined by polynomials has recently seen a lot of interest [6], and more specifically for 3D reconstruction [16, 19]. One of the main contributions is the development and computation of the so called Euclidean distance degree, which is the number of critical points to the closest point optimization problem given generic data. It expresses the algebraic complexity of fitting data to a model; the higher the Euclidean distance degree is, the more computationally expensive this optimization is. It is used to implement efficient solvers in Homotopy Continuation [2] or for solving the associated polynomial systems via specialized symbolic solvers [11].

The Sampson Error. The Sampson approximation was first proposed in [20] to approximate the point-conic distance. It was also later derived independently by

Taubin [27]. Since then it has appeared in numerous papers for different problems. Luong and Faugeras [15] introduced it for approximating the reprojection error in epipolar geometry. The corresponding geometric error for homographies was first introduced by Sturm [24] and later revisited by Chum et al. [4] who also derived the Sampson approximation in this setting. Leonardos et al. [12] used the Sampson error for point-line-line constraint from the trifocal tensor. Chojnacki et al. [3] considered how to integrate known measurement covariances, and recently Terekhov et al. [28] generalized the Sampson error in epipolar geometry to handle arbitrary central camera model. In [26] it was used in an optimization method for conic fitting that guarantees convergence to an ellipse. This extensive use of the Sampson approximation for geometric problems shows its versatility, which motivates a deeper theoretical study in a more general setting. This is precisely our goal in this work.

2. The Sampson Approximation

We consider geometric residuals analogous to (2) for general models, i.e. problems of the form

$$\begin{aligned} \mathcal{E}_G^2(\mathbf{z}, \boldsymbol{\theta}) &= \min_{\boldsymbol{\varepsilon}} \|\boldsymbol{\varepsilon}\|^2 & (5) \\ \text{s.t. } & C(\mathbf{z} + \boldsymbol{\varepsilon}, \boldsymbol{\theta}) = 0 & (6) \end{aligned}$$

where $\mathbf{z} \in \mathbb{R}^n$ is our measurement, $\boldsymbol{\theta}$ our model parameters, and $C(\mathbf{z}, \boldsymbol{\theta})$ our geometric constraint. In the special case of epipolar geometry, we have that $\mathbf{z} = [\mathbf{x}_1, \mathbf{x}_2]$ consists of the two matching image points, the model parameters are the entries of the essential matrix \mathbf{E} , and the constraint is,

$$C(\mathbf{z}, \mathbf{E}) = (\mathbf{x}_2; 1)^\top \mathbf{E}(\mathbf{x}_1; 1) = 0. \quad (7)$$

In the rest of the paper we consider a general polynomial constraint $C(\mathbf{z}, \boldsymbol{\theta})$ and will for notational convenience drop the dependence on the model parameters $\boldsymbol{\theta}$ in most places.

Since (5) often does not admit a simple closed form solution, the idea in [20] is to linearize the constraint, $C(\mathbf{z} + \boldsymbol{\varepsilon}) = 0$, at the original measurement \mathbf{z} , i.e.

$$\begin{aligned} \mathcal{E}_S^2(\mathbf{z}, \boldsymbol{\theta}) &= \min_{\boldsymbol{\varepsilon}} \|\boldsymbol{\varepsilon}\|^2 & (8) \\ \text{s.t. } & C(\mathbf{z}) + \mathbf{J}\boldsymbol{\varepsilon} = 0 & (9) \end{aligned}$$

where $\mathbf{J} = \partial C(\mathbf{z})/\partial \mathbf{z}$ is the Jacobian of the constraint, evaluated at \mathbf{z} . Introducing a Lagrangian for (8),¹

$$\mathcal{L}(\boldsymbol{\varepsilon}, \alpha) = \frac{1}{2} \|\boldsymbol{\varepsilon}\|^2 + \alpha (C(\mathbf{z}) + \mathbf{J}\boldsymbol{\varepsilon}) \quad (10)$$

we get the first-order constraints as

$$\boldsymbol{\varepsilon} + \alpha \mathbf{J}^\top = 0, \quad C(\mathbf{z}) + \mathbf{J}\boldsymbol{\varepsilon} = 0. \quad (11)$$

¹For convenience we introduce 1/2 here; it does not affect the optimum.

Inserting the first equation into the second yields

$$C(z) - \alpha \|J\|^2 = 0 \implies \varepsilon = -\frac{C(z)}{\|J\|^2} J^\top. \quad (12)$$

Thus the minimum in (8) is given by

$$\mathcal{E}_S^2(z, \theta) = \left\| \frac{C(z)}{\|J\|^2} J^\top \right\|^2 = \frac{C(z)^2}{\|J\|^2}. \quad (13)$$

In the special case of epipolar geometry, replacing $C(z)$ with the epipolar constraint, we arrive at the classical formula for the Sampson error (4).

2.1. Multiple Constraints and Covariances

The Sampson error for a single constraint, discussed previously, can easily be generalized to the case where we measure the deviation in the Mahalanobis distance from a model defined by multiple constraints. See [7, p128] for more details. Assume that the N constraints are given by

$$C(z) = (C_1(z), C_2(z), \dots, C_N(z))^\top = \mathbf{0}, \quad (14)$$

and we want to measure the deviation ε in Mahalanobis distance for a given positive definite covariance Σ . This can be formulated as

$$\mathcal{E}_G^2(z, \theta) = \min_{\varepsilon} \|\varepsilon\|_{\Sigma}^2 \quad (15)$$

$$\text{s.t. } C(z + \varepsilon) = \mathbf{0}, \quad (16)$$

where $\|\varepsilon\|_{\Sigma}^2 = \varepsilon^\top \Sigma^{-1} \varepsilon$. The Lagrangian of the linearized constraint then becomes

$$\mathcal{L}(\varepsilon, \alpha) = \frac{1}{2} \|\varepsilon\|_{\Sigma}^2 + \alpha^\top (C(z) + J\varepsilon), \quad (17)$$

where $J \in \mathbb{R}^{N \times n}$ is the Jacobian of C evaluated at $z \in \mathbb{R}^n$. First order conditions again yield

$$\Sigma^{-1} \varepsilon + J^\top \alpha = \mathbf{0}, \quad C(z) + J\varepsilon = \mathbf{0}. \quad (18)$$

We get $\varepsilon = -\Sigma J^\top \alpha$, allowing us to solve for α as

$$C(z) - J\Sigma J^\top \alpha = \mathbf{0} \implies \alpha = (J\Sigma J^\top)^{-1} C(z), \quad (19)$$

when $J\Sigma J^\top$ is invertible (in Section 2.2 we consider the general case). Let ε^S denote the argmin of the linearized optimization problem. From this we get

$$\varepsilon^S = -\Sigma J^\top (J\Sigma J^\top)^{-1} C(z), \quad (20)$$

and finally, $\|\varepsilon^S\|_{\Sigma}^2 = \varepsilon^S \Sigma^{-1} \varepsilon^S$, which simplifies to

$$C(z)^\top (J\Sigma J^\top)^{-1} C(z) = \|C(z)\|_{J\Sigma J^\top}^2. \quad (21)$$

2.2. General Case

Let $(\cdot)^\dagger$ denote the Moore-Penrose pseudo-inverse of a matrix. Assuming that the linearized equation $C(z) + J\varepsilon = \mathbf{0}$ has at least one solution, meaning that $C(z) \in \text{Im } J = \text{Im } J\Sigma^{1/2}$, then $(J\Sigma^{1/2})(J\Sigma^{1/2})^\dagger C(z) = C(z)$. In this case, all solutions to the linearized equation can be written

$$\varepsilon(\mu) = -\Sigma^{1/2} (J\Sigma^{1/2})^\dagger C(z) + M\mu, \quad \mu \in \mathbb{R}^N \quad (22)$$

where the columns of M is a basis for the nullspace of J . In order to find ε^S we consider

$$\|\varepsilon^S\|_{\Sigma} = \min_{\mu} \|\Sigma^{-1/2} (-\Sigma^{1/2} (J\Sigma^{1/2})^\dagger C(z) + M\mu)\| \quad (23)$$

$$= \min_{\mu} \| - (J\Sigma^{1/2})^\dagger C(z) + \Sigma^{-1/2} M\mu \|. \quad (24)$$

Here we note that $M^\top \Sigma^{-1/2} (J\Sigma^{1/2})^\dagger = \mathbf{0}$, which implies that the optimal choice is $\mu = \mathbf{0}$. To be precise, we get

$$\varepsilon^S = -\Sigma^{1/2} (J\Sigma^{1/2})^\dagger C(z). \quad (25)$$

and thus

$$\|\varepsilon^S\|_{\Sigma}^2 = \|(J\Sigma^{1/2})^\dagger C(z)\|^2. \quad (26)$$

Remark 2.1. For the case of $\Sigma = I$, we note that $\|\varepsilon^S\| = \|J^\dagger C(z_0)\|$ is the length of the Gauss-Newton step,

$$z_1 = z_0 - (J^\top J)^{-1} J^\top C(z_0) \quad (27)$$

applied to solving $C(x) = 0$ starting from the point z_0 . Figure 1 visualizes this for one constraint in two variables.

3. Bounding the Approximation Error

In this section we investigate the Sampson approximation (8) in terms of how well it approximates the geometric error (5). We do this by constructing explicit bounds. We start by studying one quadratic constraint and then we generalize our methods by considering constraints of higher degrees. Finally, we discuss the case of multiple constraints, but leave the details in the Supplementary Material.

3.1. One Quadratic Constraint

Consider the case where we have one quadratic constraint $C(z)$. The classical Sampson error (4) falls into this category as the epipolar constraint is a quadratic polynomial in terms of the image points.

We write ε^G for the argmin of (5) and ε^S for the argmin of (8). Given a fixed data point $z \in \mathbb{R}^n$, we write H for the Hessian of C at z . Since C is a quadratic polynomial,

$$C(z + \varepsilon) = C(z) + J\varepsilon + \frac{1}{2} \varepsilon^\top H \varepsilon. \quad (28)$$

We use that any vector ε satisfies the inequality

$$|\varepsilon^\top \mathbf{H} \varepsilon| \leq \rho \|\varepsilon\|^2 \quad (29)$$

where ρ is the spectral radius of the Hessian of $C(\mathbf{z})$, that is, the maximum of the absolute values of its eigenvalues.

First we present an upper bound on the Sampson approximation $\|\varepsilon^S\|$ in terms of the geometric residual $\|\varepsilon^G\|$.

Proposition 3.1. *When the optimization problem (5) only has one quadratic constraint and $\mathbf{J} \neq 0$, then*

$$\|\varepsilon^S\| \leq \|\varepsilon^G\| + \frac{\rho}{2\|\mathbf{J}\|} \|\varepsilon^G\|^2. \quad (30)$$

Under the assumption that $\rho/2\|\mathbf{J}\|$ is reasonably small (i.e. function is approximately linear), we can interpret this proposition as saying that if ε^S is big, then ε^G is also big.

The condition $\mathbf{J} \neq 0$ is reasonable, as the approximation is undefined if the linearized constraint set is empty.

Proof. Since ε^G satisfies the constraint (28), we have that

$$0 = C(\mathbf{z}) + \mathbf{J}\varepsilon^G + \frac{1}{2}(\varepsilon^G)^\top \mathbf{H} \varepsilon^G \quad (31)$$

$$= C(\mathbf{z}) + \mathbf{J} \left(\varepsilon^G + \frac{\mathbf{J}^\top}{\|\mathbf{J}\|^2} \frac{1}{2} (\varepsilon^G)^\top \mathbf{H} \varepsilon^G \right). \quad (32)$$

In other words, $\varepsilon^G + \frac{\mathbf{J}^\top}{\|\mathbf{J}\|^2} \frac{1}{2} (\varepsilon^G)^\top \mathbf{H} \varepsilon^G$ satisfies the linearized constraint. Since ε^S is by definition the smallest vector that satisfies the linearized constraint, we get that

$$\|\varepsilon^S\| \leq \|\varepsilon^G\| + \frac{1}{2\|\mathbf{J}\|} |(\varepsilon^G)^\top \mathbf{H} \varepsilon^G|. \quad (33)$$

Finally, using the inequality of the spectral radius ρ mentioned above, we get the upper bound. \square

Next we give an upper bound of $\|\varepsilon^G\|$ in terms of $\|\varepsilon^S\|$.

Proposition 3.2. *If the minimization problem (5) only has one quadratic constraint,*

$$\mathbf{J} \neq 0 \text{ and } \|\mathbf{J}\|^4 \geq 2|C(\mathbf{z})\|\mathbf{J}\mathbf{H}\mathbf{J}^\top|, \quad (34)$$

then

$$\|\varepsilon^G\| \leq 2\|\varepsilon^S\|. \quad (35)$$

The assumption should be intuitively understood as the model being close enough to linear in the direction of \mathbf{J} locally around the input \mathbf{z} , relative to the size of $C(\mathbf{z})$. Note that from the Proposition, we also have

$$\|\varepsilon^S - \varepsilon^G\| \leq \|\varepsilon^S\| + \|\varepsilon^G\| \leq 3\|\varepsilon^S\|. \quad (36)$$

Proof. Define $\varepsilon(\lambda) := \frac{\lambda}{\|\mathbf{J}\|} \mathbf{J}^\top$ for $\lambda \in \mathbb{R}$ and consider

$$C(\mathbf{z} + \varepsilon(\lambda)) = C(\mathbf{z}) + \|\mathbf{J}\|\lambda + \frac{\mathbf{J}\mathbf{H}\mathbf{J}^\top}{2\|\mathbf{J}\|^2} \lambda^2, \quad (37)$$

which is a quadratic polynomial in λ . Evaluating this polynomial at $\lambda = -2C(\mathbf{z})/\|\mathbf{J}\|$, we get

$$-C(\mathbf{z}) + 2\frac{\mathbf{J}\mathbf{H}\mathbf{J}^\top}{\|\mathbf{J}\|^4} C(\mathbf{z})^2. \quad (38)$$

The absolute value of the second term can be bounded from above by $|C(\mathbf{z})|$. This means that either $C(\mathbf{z}) \geq 0$ and $C(\mathbf{z} + \varepsilon(-2C(\mathbf{z})/\|\mathbf{J}\|)) \leq 0$ or the other way around. By continuity of polynomials, there must exist a solution λ^* to (37) in the interval $[0, -2|C(\mathbf{z})|/\|\mathbf{J}\|]$. Since ε^G is the smallest vector satisfying this, we have

$$\|\varepsilon^G\| \leq \|\varepsilon(\lambda^*)\| = |\lambda^*| \leq 2|C(\mathbf{z})|/\|\mathbf{J}\| = 2\|\varepsilon^S\|. \quad (39)$$

\square

In order to get a sharper bound in the case that $\mathbf{J}\mathbf{H}\mathbf{J}^\top \neq 0$, we may instead solve the quadratic equation (37) directly:

$$\lambda^* = \frac{-\|\mathbf{J}\|^3 \pm \|\mathbf{J}\| \sqrt{\|\mathbf{J}\|^4 - 2C(\mathbf{z})\mathbf{J}\mathbf{H}\mathbf{J}^\top}}{\mathbf{J}\mathbf{H}\mathbf{J}^\top}. \quad (40)$$

Let λ^* be the solution with smallest absolute value, then $\|\varepsilon^G\| \leq |\lambda^*|$ and, following from the proof of Proposition 3.2, $|\lambda^*| \leq 2\|\varepsilon^S\|$. Note that if $\mathbf{J}\mathbf{H}\mathbf{J}^\top = 0$, then $\lambda^* = -C(\mathbf{z})/\|\mathbf{J}\|$ and we directly get $\|\varepsilon^G\| \leq \|\varepsilon^S\|$.

To summarize we have the following inequalities

$$\frac{1}{2}\|\varepsilon^G\| \leq \frac{1}{\tau}\|\varepsilon^G\| \leq \|\varepsilon^S\| \leq \|\varepsilon^G\| + \frac{\rho}{2\|\mathbf{J}\|} \|\varepsilon^G\|^2, \quad (41)$$

where $\tau = |\lambda^*/C(\mathbf{z})|\|\mathbf{J}\|$ and λ^* is given by (40).

The hypothesis in Proposition 3.2 defines a region where the geometric error is bounded linearly by the Sampson error. We highlight that this region contains the region defined by

$$\|\varepsilon^S\| = \frac{|C(\mathbf{z})|}{\|\mathbf{J}\|} \leq \frac{\|\mathbf{J}\|}{2\rho}. \quad (42)$$

Example 3.3. Consider as a constraint the conic $C(\mathbf{x}) = x_1^2 + 2x_2^2 - 4$ in \mathbb{R}^2 . For fixed $\mathbf{z} = (z_1, z_2)$, the minimization problem is

$$\min_{\varepsilon} \|\varepsilon\|^2 \quad (43)$$

$$\text{s.t. } (z_1 + \varepsilon_1)^2 + 2(z_2 + \varepsilon_2)^2 - 4 = 0 \quad (44)$$

and the linearized constraint is

$$z_1^2 + 2z_2^2 - 4 + [2z_1 \quad 4z_2] \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \end{bmatrix} = 0. \quad (45)$$

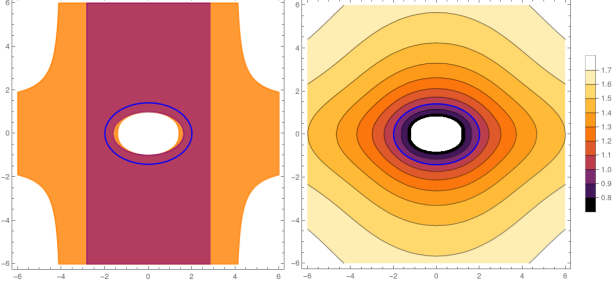


Figure 2. Regions obtained for Example 3.3. On the left, the constraint $x_1^2 + 2x_2^2 - 4 = 0$ is depicted in blue, the purple region is obtained from Equation (42) and it is contained in the orange region coming from Equation (35). On the right, the level sets for the ratio $\|\varepsilon^G\|/\|\varepsilon^S\|$, and the constraint depicted in blue. Observe that, although the ratio changes, it is bounded by 2 for every point in the colored regions.

From Proposition 3.2, we have that $\|\varepsilon^G\| \leq 2\|\varepsilon^S\|$ if \mathbf{z} is in the colored region of Figure 2 (both the orange and purple region). Finally, the relaxed condition from (42) gives a smaller region, potentially easier to work with, where the previous bounds also hold (purple region in Figure 2)

3.2. One Polynomial Constraint

To understand the Sampson error for a model defined by a single polynomial constraint of any degree $C : \mathbb{R}^n \rightarrow \mathbb{R}$, we extend the method presented previously and consider Taylor approximations of degree d :

$$C(\mathbf{z} + \varepsilon) = C(\mathbf{z}) + \sum_{i=1}^d \frac{1}{i!} \varepsilon \times \mathcal{T}_i, \quad (46)$$

where \mathcal{T}_i is a symmetric $n \times \dots \times n$ tensor of order i , and

$$\varepsilon \times \mathcal{T}_i = \sum_{j_1, \dots, j_i \in [n]} (\mathcal{T}_i)_{j_1, \dots, j_i} \varepsilon_{j_1} \dots \varepsilon_{j_i}. \quad (47)$$

For example, $\mathcal{T}_1 = \mathbf{J}$ is the Jacobian and $\mathcal{T}_2 = \mathbf{H}$ the Hessian.

Proposition 3.4. *When the optimization problem (5) only has one polynomial constraint of degree d ,*

$$\mathbf{J} \neq 0 \text{ and } |C(\mathbf{z})| \geq \left| \sum_{i=2}^d (-1)^i \frac{2^i C(\mathbf{z})^i}{i! \|\mathbf{J}\|^i} \mathbf{J} \times \mathcal{T}_i \right|, \quad (48)$$

then $\|\varepsilon^G\| \leq 2\|\varepsilon^S\|$.

Proof. The proof is exactly the same as for Proposition 3.2; one checks that $C(\mathbf{z}) \leq 0$ and $C(\mathbf{z} + \varepsilon(\lambda)) \geq 0$ (or the other way around) for $\lambda = -2C(\mathbf{z})/\|\mathbf{J}\|$. \square

3.3. Multiple Polynomial Constraints

For mathematical models defined by multiple constraints, the Sampson error and its relation to the geometric error are more involved. Here, we give an overview of

our approach to computing Sampson errors in practice and studying its relation to the geometric error for polynomial constraints, but leave mathematical statements and proofs in the Supplementary Material.

Assume that $C(\mathbf{z}) = (C_1(\mathbf{z}), \dots, C_N(\mathbf{z}))$ are N polynomial constraints. The model $C(\mathbf{z}) = \mathbf{0}$, for $\mathbf{z} \in \mathbb{R}^n$, is an algebraic variety X (a zero set of a system of polynomials) of some dimension $m \leq n$, which depends on the constraints. For generic constraints of fixed degrees $m = n - N$, i.e. each constraint lowers the dimension of the model by one, however, for specific systems of polynomials this equality does not necessarily hold. In both cases, the intuitive behaviour is that the more constraints we have, the smaller m is and our data is harder to fit to the model. The same phenomenon occurs with the linearized constraints for the Sampson error: The more constraints we have, the smaller is the affine linear space defined by $C(\mathbf{z}) + \mathbf{J}\varepsilon = 0$. Therefore, if the number of constraints N is much greater than the codimension $n - m$, the Sampson approximation is likely to be poor. To remedy this, we propose to use the fact that locally around a generic point of X , the model is described by precisely $n - m$ polynomials whose Jacobian has full rank. This is a result coming from algebraic geometry. Our proposal is to perform Sampson approximation by choosing a subset of $n - m$ constraints whose Jacobian is full-rank and linearizing those constraints. In Section 4.2 we try this approach for the Three-View Sampson error and the results suggest that it is also beneficial to choose these constraints with smallest degree possible.

In the Supplementary Material, we generalize Proposition 3.1 and find an upper bound for $\|\varepsilon^S\|$ in terms of $\|\varepsilon^G\|$ for multiple constraints of any degrees. We also generalize Proposition 3.2 for quadratic constraints and data points \mathbf{z} such that the Jacobian \mathbf{J} at \mathbf{z} has linearly independent rows. In this case, under appropriate conditions, we find a $\boldsymbol{\lambda}^*$ such that $C_i(\mathbf{z} + \|\mathbf{J}\| \mathbf{J}^\dagger \boldsymbol{\lambda}^*) = 0$ for $i = 1, \dots, n$. We get

$$\|\varepsilon^G\| \leq \|\|\mathbf{J}\| \mathbf{J}^\dagger \boldsymbol{\lambda}^*\| \leq \|\mathbf{J}\| \|\mathbf{J}^\dagger\| \|\boldsymbol{\lambda}^*\|, \quad (49)$$

and by construction we get an upper bound for this $\|\boldsymbol{\lambda}^*\|$ expressed in terms of C , its Jacobian and its Hessian.

4. Experimental Evaluation

In the following sections we evaluate the Sampson approximation for different geometric estimation problems. First, in Section 4.1 we evaluate the classical Sampson error for two-view geometry. Next, in Section 4.2 we consider the analogous error in the three-view setting. Section 4.3 considers line segment to vanishing point errors. Finally, Section 4.4 show an application from absolute pose estimation with uncertainties applied in both 2D and 3D.

4.1. Application: Two-view Relative Pose

We first consider the classical setting where the Sampson approximation is applied, two-view relative pose estimation. For the experiments we use image pairs from the IMC Phototourism 2021 [9] (SIFT), MegaDepth-1500 [25] (SP+LG [13]) and ScanNet-1500 [22] (SP+SG [22]). For each image pair we estimate an initial essential matrix using DLT [7] applied to the ground truth inliers (from [14]).

Results and Discussion. Figure 3 shows the distribution of the difference between the Sampson approximation and the true error, and in Table 1 shows the Area-Under-Curve² up to 1 pixel, on the *British Museum* scene from IMC-PT. For comparison we also include the symmetric epipolar error (distances to the epipolar lines computed in both images) which is another popular choice in practice. The Sampson error provides a very accurate approximation of the true reprojection error. As discussed in Section 3 the quality of the approximation depends on the how close the initial point is to the constraint set (small $C(\mathbf{z})$) and the curvature (small Hessian). In Figure 4 we plot the approximation error $|\mathcal{E}_S - \mathcal{E}_G|$ against $\rho|C(\mathbf{z})|/\|J\|^2$, where ρ is the spectral norm of the Hessian. Consistent with the theory, the figure shows a clear trend where correspondences with smaller $\rho|C(\mathbf{z})|/\|J\|^2$ have smaller errors.

Finally we also evaluate the error in the context of pose refinement on all three datasets. Table 2 shows the resulting pose errors (max of rotation and translation error) after non-linear refinement of the initial essential matrix using different error functions. *Algebraic* is the squared epipolar constraints. *Cosine* is the squared cosine of the angles between the normals of the epipolar planes and the point correspondences. The Sampson error provides the most accurate camera poses after refinement.

AUC@10° pose error	IMC-PT	MD1.5k	SN1.5k
Initial estimate (DLT)	0.361	0.321	0.274
↳ Algebraic	0.580	0.515	0.485
↳ Cosine	0.654	0.689	0.600
↳ Sym. Epipolar	0.673	0.728	0.654
↳ Sampson	0.678	0.732	0.657

Table 2. Comparison of losses for two-view relative pose refinement on IMC-PT, MegaDepth-1500 and ScanNet-1500. The initial pose is found by linear estimate (DLT) applied to the inlier correspondences (w.r.t. ground truth pose), followed by non-linear refinement.

4.2. Application: Three-view Sampson Error

In this section we evaluate different error formulations for 3-view point matches. The naive baseline is to simply average the two-view Sampson errors (4), $\mathcal{E}_{pair} =$

$$\mathcal{E}_S(\mathbf{x}, \mathbf{x}', E_{12}) + \mathcal{E}_S(\mathbf{x}, \mathbf{x}'', E_{13}) + \mathcal{E}_S(\mathbf{x}', \mathbf{x}'', E_{23}) \quad (50)$$

²Area under the CDF up to 1 px error as a ratio of the complete square.

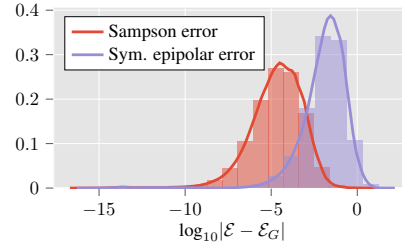


Figure 3. Approximation gap for two-view relative pose. Comparison with optimal triangulation [14]. The unit is in pixels. Here \mathcal{E} refers to either the Sampson or the symmetric epipolar error.

Error	$ \mathcal{E} - \mathcal{E}_G $, AUC @ τ px (\uparrow)		
	$\tau = 0.1$	$\tau = 0.5$	$\tau = 1$
Sampson	0.991	0.998	0.999
Symmetric Epipolar	0.620	0.839	0.902

Table 1. The table shows the Area-Under-Curve of the approximation error $|\mathcal{E} - \mathcal{E}_G|$ up to different thresholds.

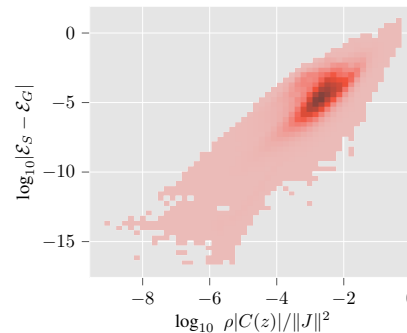


Figure 4. Approximation error against $\rho|C(\mathbf{z})|/\|J\|^2$ for two-view relative pose. Figure shows a heatmap built from the inlier correspondences of $\sim 5k$ image pairs from the *British Museum* scene. Points that are either close to satisfying the epipolar constraint ($C(\mathbf{z}) \approx 0$) or have low curvature (ρ) have smaller errors.

where $(\mathbf{x}, \mathbf{x}', \mathbf{x}'')$ is the correspondence and E_{ij} are the essential matrices. For a given a trifocal tensor $\mathcal{T} \in \mathbb{R}^{3 \times 3 \times 3}$ with slices $T_1, T_2, T_3 \in \mathbb{R}^{3 \times 3}$, a consistent three-view point correspondence $\mathbf{x}, \mathbf{x}', \mathbf{x}''$ satisfies

$$C_9(\mathbf{x}, \mathbf{x}', \mathbf{x}'') = [\hat{\mathbf{x}}']_{\times} \left(\sum_k \hat{\mathbf{x}}_k T_k \right) [\hat{\mathbf{x}}'']_{\times} = \mathbf{0} \quad (51)$$

where $\hat{\mathbf{x}} = [\mathbf{x}; 1]$ is the homogenization of the 2D point $\mathbf{x} \in \mathbb{R}^2$. While we have nine equations, only four are linearly independent. These can be obtained by multiplying with two 3×2 matrices,

$$C_4(\mathbf{x}, \mathbf{x}', \mathbf{x}'') = S_1^T [\hat{\mathbf{x}}']_{\times} \left(\sum_k \hat{\mathbf{x}}_k T_k \right) [\hat{\mathbf{x}}'']_{\times} S_2 \quad (52)$$

where $S_1, S_2 \in \mathbb{R}^{3 \times 2}$ contains a basis for the complement of the left and right nullspace of C_9 respectively. Another

alternative is to only consider the three pairwise constraints,

$$\mathbf{C}_3(\mathbf{x}, \mathbf{x}', \mathbf{x}'') = [\mathbf{x}'^\top E_{12} \mathbf{x}, \mathbf{x}''^\top E_{13} \mathbf{x}, \mathbf{x}''^\top E_{23} \mathbf{x}'] \quad (53)$$

Note that applying the Sampson approximation to \mathbf{C}_3 yields a different error compared to \mathcal{E}_{pair} (50), as the constraints are considered jointly.

In this section we evaluate the following errors

- \mathcal{E}_{pair} - averaging the pairwise Sampson errors (50)
- \mathcal{E}_9 - applying Sampson approximation to \mathbf{C}_9 (51)
- \mathcal{E}_4 - applying Sampson approximation to \mathbf{C}_4 (52)
- \mathcal{E}_3 - applying Sampson approximation to \mathbf{C}_3 (53)

We also include two combinations of the above. First taking 3 out of the 4 constraints from \mathbf{C}_4 , denoted $\mathbf{C}_{4:3}$, and one where we combine \mathbf{C}_3 and \mathbf{C}_4 by taking two quadratic and one cubic constraint, denoted $\mathbf{C}_{4:1,3:2}$. We also consider a set of pseudo-Sampson approximations which take the form $\|\mathbf{C}\|^2/\|\mathbf{J}\|^2$, and thus avoid computing matrix inverses as in Section 2.2. This can be seen as a naive extension of the 1-dimensional Sampson approximation (8) to the multi-dimensional case. We denote these as $\hat{\mathcal{E}}_9, \hat{\mathcal{E}}_4, \hat{\mathcal{E}}_3$.

Experiment setup. To compare the approximations we generate synthetic camera triplets (70° field-of-view, 1000x1000 pixel images) observing a 3D point. To the three projections we add normally distributed noise with standard deviation $\sigma \in \{1, 5, 10\}$ px. We obtain the reference ground-truth reprojection error by directly optimizing over the 3D point. For each of the evaluated approximated error functions we compute the difference to the ground-truth.

Results and Discussion. Table 3 shows the errors for 100k synthetic instances. We compute the Area-Under-Curve up to 1 pixel deviation from the reference error. The Sampson approximation of the epipolar constraints \mathbf{C}_3 (53) yields the best approximation. In particular, we can see that the naive approximation \mathcal{E}_{pair} that averages the pairwise Sampson errors, is significantly worse. Interestingly, we can also see that \mathbf{C}_4 performs much worse compared to both \mathbf{C}_3 and the mixed variants $\mathbf{C}_{4:3}$ and $\mathbf{C}_{4:1,3:2}$. This is consistent with the discussion in Sec. 3.3.

4.3. Application: Vanishing Point Estimation

We now show another example with a 1-dimensional quadratic constraint. Consider a line-segment $(\mathbf{x}_1, \mathbf{x}_2) \in \mathbb{R}^2 \times \mathbb{R}^2$ and a vanishing point $\mathbf{v} \in \mathbb{S}^2$. Assuming we want to refine \mathbf{v} , it is reasonable to consider what is the smallest perturbation of the line endpoints such that the line passes through the vanishing point, i.e. satisfy the constraint

$$C(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{v}^\top \left(\begin{pmatrix} \mathbf{x}_1 \\ 1 \end{pmatrix} \times \begin{pmatrix} \mathbf{x}_2 \\ 1 \end{pmatrix} \right) \quad (54)$$

	$ \mathcal{E} - \mathcal{E}_{gt} , \text{ AUC @ 1px } (\uparrow)$		
	$\sigma = 1\text{px}$	$\sigma = 5\text{px}$	$\sigma = 10\text{px}$
$\mathcal{E}_3 = \ J_3^\dagger \mathbf{C}_3\ $	0.998	0.961	0.882
$\mathcal{E}_{4:1,3:2} = \ J_{4:1,3:2}^\dagger \mathbf{C}_{4:1,3:2}\ $	0.995	0.937	0.830
$\mathcal{E}_{4:3} = \ J_{4:3}^\dagger \mathbf{C}_{4:3}\ $	0.994	0.912	0.768
$\mathcal{E}_4 = \ J_4^\dagger \mathbf{C}_4\ $	0.765	0.481	0.353
$\mathcal{E}_{pair} = \dots$ (50)	0.764	0.312	0.172
$\hat{\mathcal{E}}_3 = \ \mathbf{C}_3\ /\ J_3\ $	0.361	0.016	0.003
$\hat{\mathcal{E}}_4 = \ \mathbf{C}_4\ /\ J_4\ $	0.356	0.009	0.001
$\hat{\mathcal{E}}_9 = \ \mathbf{C}_9\ /\ J_9\ $	0.356	0.009	0.001
$\mathcal{E}_9 = \ J_9^\dagger \mathbf{C}_9\ $	0.348	0.075	0.034

Table 3. Comparison of three-view reprojection error approximations. Each error metric is compared to the ground truth error (found by non-linear refinement of the 3D point). To each keypoint we add noise with standard deviation σ pixels.

Differentiating with respect to the image points we get,

$$J = \left[\left(\begin{pmatrix} \mathbf{x}_2 \\ 1 \end{pmatrix} \times \mathbf{v} \right) S, \left(\mathbf{v} \times \begin{pmatrix} \mathbf{x}_1 \\ 1 \end{pmatrix} \right) S \right] \quad (55)$$

where $S = [I_2 \quad \mathbf{0}]^\top$, and we can directly setup a Sampson approximation of the line-segment to vanishing point distance as $|C(\mathbf{x}_1, \mathbf{x}_2)|/\|\mathbf{J}\|$. For this problem the ground-truth error \mathcal{E}_G can be computed in closed form using SVD.

In Section 3.1 we derived bounds that relate the true geometric error ϵ^G and the Sampson approximation ϵ^S ,

$$B_l := \frac{1}{\tau} \leq \frac{\|\epsilon^S\|}{\|\epsilon^G\|} \leq 1 + \frac{\rho}{2\|\mathbf{J}\|} \|\epsilon^G\| =: B_u \quad (56)$$

Next we evaluate how tight these bounds are on real data.

Experiment Setup. For the experiment we consider circa 350k pairs of line segments and vanishing points collected from the YUB+ [5] and NYU VP [10, 17]. The line segments are detected using DeepLSD [18] and using Progressive-X [1] we estimate a set of vanishing points.

Results and Discussion. Figure 5 shows B_u, B_l for each of line-vanishing point pairs and in Figure 6 we show the distribution of the difference between the bounds. As can be seen in the figures the approximation works extremely well for this setting. We also experimented with refining the vanishing points using the Sampson error but found that the results are very similar to minimizing the mid-point error (as was done in [18]). The full results and details can be found in the Supplementary Material.

4.4. Application: 2D/3D Reprojection Error

Minimizing the square reprojection error, i.e. deviation between the observed 2D point and the projection of the 3D point, assumes a Gaussian noise model on the 2D observations. However, in many scenarios, we also have noise in the 3D points. In this section we consider the case where we have a known covariance for both the 2D and 3D-points.

	$\tau = 5$ px			$\tau = 10$ px			$\tau = 20$ px			RT
	AUC (\uparrow)	ε_R (\downarrow)	ε_t (\downarrow)	AUC (\uparrow)	ε_R (\downarrow)	ε_t (\downarrow)	AUC (\uparrow)	ε_R (\downarrow)	ε_t (\downarrow)	
Reprojection error (60)	0.358	1.03	3.20	0.350	1.06	3.24	0.311	1.21	3.63	0.7 ms
Reprojection error + Cov. (61)	0.367	1.01	3.10	0.379	0.99	2.99	0.378	0.99	3.00	10.1 ms
↳ Sampson approximation	0.367	1.01	3.10	0.379	0.98	2.99	0.375	1.00	3.02	2.4 ms

Table 4. **Pose refinement on 7Scenes.** Table shows the Area-Under-Curve (AUC) @ 5cm position errors and the median errors in rotation ε_R (deg.) and translation ε_t (cm). Results are reported for different inlier thresholds τ . For large thresholds, more uncertain points are included and the improvement from the covariance weighting is more significant. However, larger errors also make the linearization point (the original correspondence) worse, degrading the Sampson approximation. For the two lower thresholds, the Sampson approximation of 2D-3D covariance weighted reprojection error gives almost identical results as performing the full (expensive) optimization.

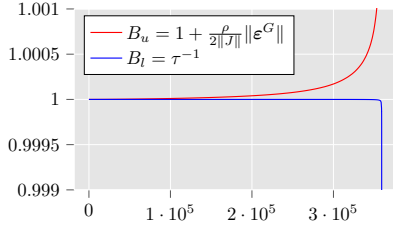


Figure 5. **Evaluation of the bounds for VP-line error.** The lower bound B_l and upper bound B_u for the $\approx 350k$ VP-line pairs in the combined dataset. For illustration the pairs are sorted w.r.t. the tightness of the bound.

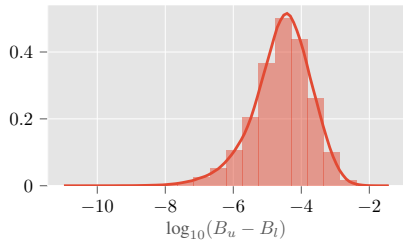


Figure 6. **Evaluation of the bounds for VP-line error.** Distribution of the \log_{10} differences between the upper and lower bounds.

Given a point correspondence $(\mathbf{x}, \mathbf{X}) \in \mathbb{R}^2 \times \mathbb{R}^3$, together with covariances $\Sigma_{2D} \in \mathbb{R}^{2 \times 2}$ and $\Sigma_{3D} \in \mathbb{R}^{3 \times 3}$, the maximum likelihood estimate is then given by

$$\min_{\varepsilon_2, \varepsilon_3} \|\varepsilon_2\|_{\Sigma_{2D}}^2 + \|\varepsilon_3\|_{\Sigma_{3D}}^2 \quad (57)$$

$$\text{s.t. } C(\mathbf{x} + \varepsilon_2, \mathbf{X} + \varepsilon_3) = 0 \quad (58)$$

where $C(\mathbf{x}, \mathbf{X})$ encodes the reprojection equations, i.e.

$$C(\mathbf{x}, \mathbf{X}) = [I_{2 \times 2}, -\mathbf{x}] (R\mathbf{X} + \mathbf{t}) = 0. \quad (59)$$

Experiment Setup. For the experiment we consider the visual localization benchmark setup on 7Scenes [23] dataset. Using HLoc [21] we establish 2D-3D matches and estimate an initial camera pose for each query image. For the experiment we then refine this camera pose, including the uncertainty in both the 2D and 3D points. The 2D covariances are assumed to be unit gaussians and to obtain the 3D covariances we propagate the 2D covariances from the mapping images used to triangulate the 3D point.

Results and Discussion. Table 4 shows the average pose error across all scenes (per-scene results are available in the

Supplementary Material). We compare only minimizing the reprojection error (only 2D noise)

$$\min_{R, \mathbf{t}} \sum_k \|\mathbf{x}_k - \pi(R\mathbf{X}_k + \mathbf{t})\|_{\Sigma_2}^2 \quad (60)$$

with optimizing over the 3D points as well (2D/3D noise),

$$\min_{R, \mathbf{t}, \{\hat{\mathbf{X}}_k\}} \sum_k \|\mathbf{x}_k - \pi(R\hat{\mathbf{X}}_k + \mathbf{t})\|_{\Sigma_2}^2 + \|\hat{\mathbf{X}}_k - \mathbf{X}_k\|_{\Sigma_3}^2 \quad (61)$$

and applying the Sampson approximation to (59). As shown in Table 4, including the uncertainty of the 3D point can greatly improve the pose accuracy. Further, the Sampson approximation works well in this setting and it is only when we include matches with very large errors (20 pixels) that performance degrades.

Note that the optimization problem in (61) requires parameterizing each individual 3D point, potentially leading to hundreds or thousands of extra parameters compared to (60) that only optimize over the 6-DoF in the camera pose. Since the Sampson approximation eliminates the extra unknowns, it also allows us to only optimize over the camera pose while modelling the 3D uncertainty. Table 4 also shows the average runtime in milliseconds for the query images. Minimizing the Sampson approximation is significantly faster compared to (61).

5. Conclusions

The Sampson approximation, originally applied to compute conic-point distances, has shown itself to surprisingly versatile in the context of robust model fitting. While it has been known that it works extremely well in practice, we provide the first theoretical bounds on the approximation error. In multiple experiments on real data in different application contexts we have validated our theory and highlighted the usefulness of the approximation.

Acknowledgments: Viktor Larsson was supported by the strategic research project ELLIIT and the Swedish Research Council (grant no. 2023-05424). Felix Rydell was supported by the Knut and Alice Wallenberg Foundation within their WASP (Wallenberg AI, Autonomous Systems and Software Program) AI/Math initiative. Angélica Torres was supported by DFG grant 464109215 within the priority programme SPP 2298 ‘‘Theoretical Foundations of Deep Learning’’.

References

- [1] Daniel Barath and Jiri Matas. Progressive-X: Efficient, any-time, multi-model fitting algorithm. In *International Conference on Computer Vision (ICCV)*, 2019. 7
- [2] Paul Breiding and Sascha Timme. Homotopycontinuation.jl: A package for homotopy continuation in julia. In James H. Davenport, Manuel Kauers, George Labahn, and Josef Urban, editors, *Mathematical Software – ICMS 2018*, pages 458–465, Cham, 2018. Springer International Publishing. 2
- [3] Wojciech Chojnacki, Michael J. Brooks, Anton Van Den Hengel, and Darren Gawley. On the fitting of surfaces to data with covariances. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2000. 2
- [4] Ondřej Chum, Tomáš Pajdla, and Peter Sturm. The geometric error for homographies. *Computer Vision and Image Understanding (CVIU)*, 2005. 2
- [5] Patrick Denis, James H Elder, and Francisco J Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. In *European Conference on Computer Vision (ECCV)*, 2008. 7
- [6] Jan Draisma, Emil Horobeț, Giorgio Ottaviani, Bernd Sturmfels, and Rekha R Thomas. The euclidean distance degree of an algebraic variety. *Foundations of computational mathematics*, 16:99–149, 2016. 2
- [7] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 3, 6
- [8] Richard I Hartley and Peter Sturm. Triangulation. *Computer Vision and Image Understanding (CVIU)*, 1997. 2
- [9] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image Matching Across Wide Baselines: From Paper to Practice. *International Journal of Computer Vision (IJCV)*, 129(2):517–547, Feb. 2021. 6
- [10] Florian Kluger, Eric Brachmann, Hanno Ackermann, Carsten Rother, Michael Ying Yang, and Bodo Rosenhahn. CONSAC: Robust multi-model fitting by conditional sample consensus. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 7
- [11] Viktor Larsson, Kalle Astrom, and Magnus Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 2
- [12] Spyridon Leonardos, Roberto Tron, and Kostas Daniilidis. A metric parametrization for trifocal tensors with non-colinear pinholes. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. 2
- [13] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *International Conference on Computer Vision (ICCV)*, 2023. 6
- [14] Peter Lindstrom. Triangulation made easy. In *Computer Vision and Pattern Recognition (CVPR)*, 2010. 2, 6
- [15] Quan-Tuan Luong and Olivier D Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision (IJCV)*, 1996. 2
- [16] Laurentiu G Maxim, Jose I Rodriguez, and Botong Wang. Euclidean distance degree of the multiview variety. *SIAM Journal on Applied Algebra and Geometry*, 4(1):28–48, 2020. 2
- [17] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgb-d images. In *European Conference on Computer Vision (ECCV)*, 2012. 7
- [18] Rémi Pautrat, Daniel Barath, Viktor Larsson, Martin R Oswald, and Marc Pollefeys. Deeplsd: Line segment detection and refinement with deep image gradients. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 7
- [19] Felix Rydell, Elima Shehu, and Angelica Torres. Theoretical and numerical analysis of 3d reconstruction using point and line incidences. *arXiv preprint arXiv:2303.13593*, 2023. 2
- [20] Paul D Sampson. Fitting conic sections to “very scattered” data: An iterative refinement of the bookstein algorithm. *Computer graphics and image processing*, 1982. 2
- [21] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 8
- [22] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 6
- [23] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *Computer Vision and Pattern Recognition (CVPR)*, 2013. 8
- [24] Peter Sturm. *Vision 3D non calibrée: contributions à la reconstruction projective et étude des mouvements critiques pour l’auto-calibrage*. PhD thesis, Institut National Polytechnique de Grenoble-INPG, 1997. 2
- [25] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Computer Vision and Pattern Recognition (CVPR)*, 2021. 6
- [26] Zygmunt L Szymak, Wojciech Chojnacki, and Anton Van Den Hengel. Guaranteed ellipse fitting with the sampson distance. In *European Conference on Computer Vision (ECCV)*, 2012. 2
- [27] Gabriel Taubin. Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 1991. 2
- [28] Mikhail Terekhov and Viktor Larsson. Tangent sampson error: Fast approximate two-view reprojection error for central camera models. In *International Conference on Computer Vision (ICCV)*, 2023. 2