

From Qualitative to Quantitative: Geometric Degeneracy and Algebraic Conditioning in Minimal Pose Estimation

Peilin Luo

luo146055@gmail.com

Northeastern University

Research Article

Keywords: Perspective-Three-Point (P3P), Danger Cylinder, Condition number, Singularity analysis, Parameterization-dependent stability, Minimal pose estimation

Posted Date: October 23rd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10455128/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

From Qualitative to Quantitative: Geometric Degeneracy and Algebraic Conditioning in Minimal Pose Estimation

Peilin Luo^{1*}

¹Department of Mathematics, Northeastern University, Shenyang, 110819, Liaoning, People's Republic of China.

Corresponding author(s). E-mail(s): luol46055@gmail.com;

Abstract

Minimal pose estimation—recovering camera pose from the smallest possible set of 2D–3D correspondences—is a foundational subroutine in geometric computer vision. A fundamental limitation shared by all minimal solvers is the existence of critical geometric configurations where the problem becomes ill-posed and numerical conditioning deteriorates. While these degenerate loci have been identified for canonical problems such as Perspective-Three-Point (P3P), existing analyses remain qualitative: they characterize *where* degeneracy occurs, but provide no tools to quantify *how severely* computation is affected at nearby configurations.

We address this gap through a systematic analysis of the relationship between geometric degeneracy and algebraic conditioning. Using a local-homography parameterization of P3P whose constraint coefficients carry explicit geometric dependence, we derive a closed-form factorization of the constraint Jacobian determinant, $\det(\mathbf{J}) \propto \sin(\phi + \theta'_{23})$, which recovers the Danger Cylinder as the algebraic singular locus and reveals it as a regular zero-set with non-vanishing normal derivative. Building on this factorization, we prove the central quantitative result: the smallest singular value of the constraint Jacobian scales linearly with the geometric distance d to the Danger Cylinder, $\sigma_{\min}(\mathbf{J}) = C_1 \cdot d + O(d^2)$, and the condition number diverges as $\kappa(\mathbf{J}) \propto 1/d$. The proportionality constant C_1 is derived in closed form from triangle geometry, and the predicted scaling is verified across four triangle shapes ($R^2 > 0.99$).

Furthermore, by comparing three algebraic parameterizations of the same geometric constraints, we formalize how the conditioning constant depends on the parameterization through the transformation rule $C_{\mathbf{x}} = C_{\mathbf{z}} \cdot g(\partial\phi/\partial\mathbf{x})$, and validate this relationship experimentally ($C_N \cdot L_{12} = C_D$ to within 0.1%). This establishes a design principle: while the geometric degeneracy determines *where* a solver becomes singular, the algebraic parameterization determines *how rapidly* instability develops. A practical solver derived from this formulation is competitive with state-of-the-art methods. These results suggest a geometric-algebraic conditioning framework that may extend to other minimal pose estimation problems.

Keywords: Perspective-Three-Point (P3P); Danger Cylinder; Condition number; Singularity analysis; Parameterization-dependent stability; Minimal pose estimation

1 Introduction

Recovering 6-DOF camera pose from the minimal number of 2D–3D correspondences is a foundational subroutine in computer vision, underpinning every RANSAC-based structure-from-motion

and SLAM pipeline. A fundamental limitation shared by all minimal pose solvers is the existence of *critical configurations*: geometric arrangements of the camera and scene points for which the problem becomes ill-posed, the solution loses uniqueness, and numerical conditioning deteriorates catastrophically. Understanding when and how severely these degeneracies affect computation is therefore a question of both theoretical and practical significance.

The Perspective-Three-Point (P3P) problem is the canonical minimal pose estimation case, studied since the 19th century. Its critical configuration—the *Danger Cylinder*—was identified in the photogrammetry literature and rigorously analyzed by Zhang and Hu [1]: when the camera center lies on the cylinder circumscribing the three control points and orthogonal to their plane, the geometry admits a one-parameter family of valid poses. Recent work has enriched this picture. Faugère et al. [2] classified the algebraic solution structure. Ding et al. [3] showed that the conic intersection discriminant vanishes on the Danger Cylinder. Sun et al. [4] provided a complete geometric stratification of singular configurations via local dual space techniques. These analyses establish *where* degeneracy occurs; the present work investigates *how* conditioning degrades in the vicinity of these loci.

Yet despite this qualitative geometric understanding, a quantitative question of direct practical importance has received less attention: **given a camera at distance d from the Danger Cylinder, how severely does the solver’s numerical conditioning degrade?** Does the smallest singular value of the constraint Jacobian decay linearly, quadratically, or exponentially with d ? Is the decay rate shared across different algebraic parameterizations of the same geometric problem, or does the choice of formulation matter? The existing literature—including the complete algebraic stratification of Sun et al. [4] and the conditioning framework of Fan et al. [5] for relative pose—answers *where* degeneracy occurs; it provides no tools to answer *how close is too close* for absolute pose estimation.

We answer these questions by exploiting the geometric transparency of a specific algebraic formulation: a local-homography parameterization whose constraint coefficients carry explicit dependence on triangle edge lengths and angles. This

parameterization serves not primarily as another solver—though it yields a competitive one—but as a *theoretical instrument* that renders the relationship between camera geometry and algebraic conditioning analytically tractable. The key insight is that the Jacobian determinant factorizes in closed form as $\det(\mathbf{J}) \propto \sin(\phi + \theta'_{23})$, where ϕ is a triangle angle and θ'_{23} is the projected observation angle. The Danger Cylinder is the regular zero-locus of this expression, and the non-vanishing normal derivative on this locus implies a simple zero, from which a linear conditioning decay law follows analytically.

Our main contributions are:

1. **Quantitative conditioning law for P3P.** We prove that near the Danger Cylinder, the smallest singular value of the constraint Jacobian scales linearly with geometric distance ($\sigma_{\min} \propto d$), and the condition number diverges as $\kappa \propto 1/d$. The proportionality constant is derived in closed form from triangle geometry. Experimental validation across four triangle shapes confirms the predicted scaling with $R^2 > 0.99$. This is, to our knowledge, the first quantitative analysis of conditioning degradation near a geometric degeneracy for absolute pose estimation.
2. **Parameterization-dependent conditioning.** By comparing three algebraic parameterizations of the same P3P constraints, we formalize how the choice of parameterization affects numerical conditioning through the transformation rule $C_{\mathbf{x}} = C_{\mathbf{z}} \cdot g(\partial\phi/\partial\mathbf{x})$, and experimentally validate this relationship ($C_N \cdot L_{12} = C_D$ to within 0.1%). This establishes conditioning as a design criterion for minimal solvers, distinct from computational efficiency.
3. **Unified degeneracy analysis.** We derive the closed-form factorization $\det(\mathbf{J}) \propto \sin(\phi + \theta'_{23})$ for the local-homography parameterization, proving that the Danger Cylinder is the regular zero-locus of the Jacobian determinant. We further prove that the collinear configuration—classically treated as a separate degeneracy—is the $R \rightarrow \infty$ asymptotic limit of the Danger Cylinder, unifying both as sections of a single degeneracy variety.

The remainder of this paper is organized as follows. Section 2 reviews related work.

Section 3 presents the local-homography formulation and derives geometrically explicit constraint coefficients. Section 4 develops the geometric-algebraic singularity analysis, proving the equivalence between Jacobian degeneracy and the Danger Cylinder, and the unification of collinearity as the $R \rightarrow \infty$ limit. Section 5 proves the quantitative conditioning law ($\sigma_{\min} \propto d$) and provides experimental validation. Section 6 analyzes parameterization-dependent stability. Section 7 briefly reports the practical solver performance. Section 8 concludes with implications for minimal pose estimation beyond P3P. Detailed solver benchmarks are provided in the Appendix.

2 Related Work

Existing literature on the P3P problem falls into four main categories: classical polynomial reduction, local frame-based solvers, constructive geometric approaches, and geometric singularity analysis.

2.1 Classical Polynomial Reduction

Early work on P3P dates back to the Law of Cosines formulations by Lagrange [6] and Grunert [7]. In the modern computational era, Fischler and Bolles [8] derived the closed-form solution essential for RANSAC. Gao et al. [9] later achieved a complete analytical solution using Wu-Ritt’s zero decomposition. These classical methods typically operate in the camera frame, resulting in complex trigonometric systems that require significant algebraic manipulation [10].

2.2 Local Frame-Based Algebraic Solvers

To bypass the complexity of global frame formulations, several modern solvers introduce an intermediate coordinate frame aligned with the control triangle [11, 12]. Nakano [13], Banno [14], and Wang et al. [15] exemplify this approach, defining a local coordinate system to linearize rotation constraints. Banno [14] and Nakano [13] treat rotation matrix elements as linear functions of unknown distances, while Wang et al. [15] use an aligned object frame to simplify coordinate representations via auxiliary variables.

Although their algebraic routines differ—ranging from Gröbner basis elimination [14] to matrix expansion [13]—these methods share the philosophy of using the local frame primarily for acceleration. They typically reduce the system to a univariate quartic equation through variable elimination. While computationally efficient, this algebraic abstraction can obscure the underlying geometric structure. For instance, Nakano [13] observed numerical instability in specific configurations but provided no theoretical explanation for the origin of these singularities. We adopt this efficient local frame formulation but analyze the properties of the resulting homography to theoretically identify the cause of singularity, rather than relying on numerical observation.

2.3 Constructive Geometric Approaches

Unlike algebraic reduction, constructive methods focus on direct geometric interpretations [16]. Persson et al. [17] and Ding et al. [3] reformulated P3P as the intersection of two conics. Subsequently, Wu et al. [18] proposed a solution strategy based on conic transformation. In contrast to traditional methods that identify intersections by constructing a degenerate conic, this work employs a specific coordinate transformation to map one of the conics into a standard parabola form, thereby allowing one variable to be explicitly expressed in terms of the other. These methods offer strong geometric intuition and speed, though they often rely on explicit case enumeration (e.g., distinguishing between intersecting, tangent, or osculating conics) to handle degeneracies. Masselli and Zell [19] constrained the camera center using the Inscribed Angle Theorem, a principle we revisit in Sec. 4 to bridge algebra and geometry.

2.4 Singularity Analysis

The multi-solution phenomenon and singular configurations of P3P have been extensively studied [1, 2, 20–22]. A central concept is the “Danger Cylinder,” where the optical center resides on a critical surface, which Zhang and Hu [1] proved leads to geometric degeneracy. Most recently, Sun et al. [4] provided a complete geometric stratification of P3P singular configurations using local

dual space techniques from computational algebraic geometry, classifying the singular locus by the multiplicity μ of the camera center ($\mu \geq 2$ on the Danger Cylinder, $\mu \geq 3$ on three distinguished generatrices, $\mu \geq 4$ on the circumcircle). Ding et al. [3] identified the degeneracy as the case where the conic discriminant vanishes.

In a broader context, Fan et al. [5, 23] introduced a general framework for analyzing numerical conditioning in minimal multiview problems, characterizing when the condition number becomes infinite for 5-point and 7-point relative pose estimation, and demonstrating that RANSAC implicitly selects for well-conditioned image data. Their Riemannian manifold-based analysis, however, focuses on relative pose and does not address how the choice of algebraic parameterization affects conditioning—a question central to the present work.

Prior literature often treats the Danger Cylinder and collinear configurations as distinct cases [24], leading to separate heuristic fallbacks in solvers [25, 26]. Below we show that collinearity can be understood as the $R \rightarrow \infty$ asymptotic limit of the Danger Cylinder, providing a unified perspective on both degeneracies. However, a deeper gap persists across all existing singularity analyses: they are *qualitative*, identifying the location of degenerate loci but providing no quantitative relationship between geometric proximity to these loci and the resulting algebraic conditioning of the solver. This paper addresses precisely this gap.

3 Background and Problem Formulation

We formulate the P3P problem in a local reference frame that exposes the geometric structure of the constraints. This parameterization is the foundation for the conditioning analysis developed in subsequent sections; a practical solver follows as a by-product. We adopt the local reference frame to eliminate world coordinate dependency.

3.1 Problem Formulation

Given three control points in the world coordinate frame, denoted as $\mathbf{P}_i^w \in \mathbb{R}^3$ ($i = 1, 2, 3$), and their corresponding normalized image observation vectors $\mathbf{m}_i \in \mathbb{R}^3$ with $\|\mathbf{m}_i\| = 1$. The P3P problem aims to recover the rotation matrix $\mathbf{R} \in SO(3)$

and translation vector $\mathbf{t} \in \mathbb{R}^3$ such that:

$$s_i \mathbf{m}_i = \mathbf{R} \mathbf{P}_i^w + \mathbf{t}, \quad i = 1, 2, 3 \quad (1)$$

where s_i are unknown scalar depths.

To decouple the estimation from arbitrary world coordinates, we define a Local Reference Frame $\mathcal{F}_L$. Let $\mathbf{P}_1^w$ be the origin and $\mathbf{v}_{13} \triangleq \mathbf{P}_3^w - \mathbf{P}_1^w$. The basis vectors $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ of $\mathcal{F}_L$ are:

$$\mathbf{e}_1 = \frac{\mathbf{P}_2^w - \mathbf{P}_1^w}{\|\mathbf{P}_2^w - \mathbf{P}_1^w\|}, \quad \mathbf{e}_3 = \frac{\mathbf{e}_1 \times \mathbf{v}_{13}}{\|\mathbf{e}_1 \times \mathbf{v}_{13}\|}, \quad \mathbf{e}_2 = \mathbf{e}_3 \times \mathbf{e}_1. \quad (2)$$

The rotation matrix $\mathbf{R}_{W2L}$ transforming the World Frame to the Local Frame is:

$$\mathbf{R}_{W2L} = [\mathbf{e}_1 \ \mathbf{e}_2 \ \mathbf{e}_3]^T \quad (3)$$

In this local frame, the control points lie on the $z = 0$ plane. Let $L_{ij} = \|\mathbf{P}_i^w - \mathbf{P}_j^w\|$. The fixed local coordinates $\tilde{\mathbf{P}}_i \in \mathbb{R}^3$ are given by:

$$\tilde{\mathbf{P}}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{P}}_2 = \begin{bmatrix} L_{12} \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{P}}_3 = \begin{bmatrix} u_3 \\ v_3 \\ 0 \end{bmatrix} \quad (4)$$

where $u_3 = (L_{12}^2 + L_{13}^2 - L_{23}^2)/(2L_{12})$ and $v_3 = \sqrt{L_{13}^2 - u_3^2}$. The geometric relationship between frames is illustrated in Fig. 1.

3.2 Linear Representation in Local Frame

Let $(\mathbf{R}_L, \mathbf{t}_L)$ denote the rigid transformation from the local frame to the camera frame. Since the local points have zero Z-coordinates, the camera coordinates $\mathbf{P}_i^c$ simplify to:

$$\mathbf{P}_i^c = u_i \mathbf{r}_1 + v_i \mathbf{r}_2 + \mathbf{t}_L \quad (5)$$

where $\mathbf{r}_k$ denotes the k -th column of $\mathbf{R}_L$. This can be written in a compact matrix form:

$$\lambda_i \mathbf{m}_i = \mathbf{H} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}, \quad \mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3] = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{t}_L] \quad (6)$$

Here, the matrix $\mathbf{H}$ encapsulates the rotation columns and translation. The columns $\mathbf{h}_1$ and $\mathbf{h}_2$ must satisfy the standard orthogonality and

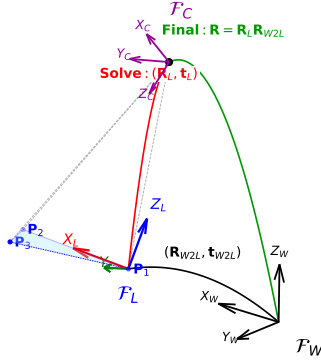


Fig. 1 Geometric formulation and coordinate frame transformations. To solve the P3P problem, we construct an intermediate **Local Reference Frame** $\mathcal{F}_L$ attached to the control triangle. The global pose estimation is decoupled into a transformation chain: (1) **World** $\rightarrow$ **Local**: A rigid transformation $(\mathbf{R}_{W2L}, \mathbf{t}_{W2L})$ analytically determined by the triangle's geometry. (2) **Local** $\rightarrow$ **Camera**: The core step where the pose $(\mathbf{R}_L, \mathbf{t}_L)$ is solved via our homography formulation. (3) **Fusion**: The final global pose is recovered by composing the transformations: $\mathbf{R} = \mathbf{R}_L \mathbf{R}_{W2L}$.

isometry constraints:

$$\mathbf{h}_1^T \mathbf{h}_2 = 0, \quad \|\mathbf{h}_1\|^2 - \|\mathbf{h}_2\|^2 = 0 \quad (7)$$

3.3 Derivation of Geometrically Explicit Coefficients

We set $\lambda_1 = 1$ to fix the scale and introduce unknowns $x = \lambda_2$ and $y = \lambda_3$. By substituting the local coordinates into Eq. (6), the columns of $\mathbf{H}$ are derived as linear functions of x and y . Crucially, we introduce geometric constants c_k derived directly from the triangle's dimensions:

$$\mathbf{h}_3 = \mathbf{m}_1 \quad (8)$$

$$\mathbf{h}_1 = c_1(x\mathbf{m}_2 - \mathbf{m}_1) \quad (9)$$

$$\mathbf{h}_2 = c_2y\mathbf{m}_3 - c_3x\mathbf{m}_2 + c_4\mathbf{m}_1 \quad (10)$$

where $c_1 = 1/L_{12}$, $c_2 = 1/v_3$, $c_3 = u_3/(L_{12}v_3)$, and $c_4 = c_3 - c_2$.

Substituting these into Eq. (7) yields two bivariate quadratic equations:

$$Q_1(x, y) := A_1x^2 + C_1xy + D_1x + E_1y + F_1 = 0, \quad (11)$$

$$Q_2(x, y) := A_2x^2 + B_2y^2 + C_2xy + D_2x + E_2y + F_2 = 0, \quad (12)$$

where $B_1 \equiv 0$ and explicit coefficient expressions are given in (13)–(14). Let $d_{ij} = \mathbf{m}_i^T \mathbf{m}_j$ be the cosine of the observed angles. The coefficients are derived as:

$$\begin{aligned} A_1 &= -c_1c_3, & B_1 &= 0, \\ C_1 &= c_1c_2d_{23}, & D_1 &= c_1(c_3 + c_4)d_{12}, \\ E_1 &= -c_1c_2d_{13}, & F_1 &= -c_1c_4 \end{aligned} \quad (13)$$

For the isometric constraint (Q_2):

$$\begin{aligned} A_2 &= c_1^2 - c_3^2, & B_2 &= -c_2^2, \\ C_2 &= 2c_2c_3d_{23}, & D_2 &= 2(c_3c_4 - c_1^2)d_{12}, \\ E_2 &= -2c_2c_4d_{13}, & F_2 &= c_1^2 - c_4^2 \end{aligned} \quad (14)$$

Remark on Geometric Structure: A notable feature of this formulation is that the coefficients in Eqs. (13) and (14) retain a clear geometric interpretation. For instance, A_1 is directly proportional to the control point coordinate ratio, and C_1 couples the edge length c_1 with the observation angle d_{23} . This explicit dependence allows us to perform symbolic differentiation in Section 4, enabling the proof that the Jacobian determinant $\det(\mathbf{J})$ vanishes strictly when the geometric configuration aligns with the Danger Cylinder.

3.4 Quartic Construction via Direct Substitution

Given that $B_1 \equiv 0$, the first constraint $Q_1(x, y) = 0$ is linear with respect to y . We exploit this algebraic structure to express y directly as a rational function of x . Rearranging Q_1 , we obtain:

$$y(C_1x + E_1) = -(A_1x^2 + D_1x + F_1). \quad (15)$$

Defining the auxiliary polynomials $M(x) = C_1x + E_1$ and $N(x) = -(A_1x^2 + D_1x + F_1)$, the relationship simplifies to $y = N(x)/M(x)$. This substitution is valid wherever $M(x) \neq 0$.

Substituting this expression into the second constraint $Q_2(x, y) = 0$ yields:

$$A_2x^2 + B_2\left(\frac{N(x)}{M(x)}\right)^2 + C_2x\frac{N(x)}{M(x)} + D_2x + E_2\frac{N(x)}{M(x)} + F_2 = 0. \quad (16)$$

Multiplying the entire equation by $M(x)^2$ eliminates the denominator, resulting in a polynomial equation solely in terms of x :

$$A_2x^2M(x)^2 + B_2N(x)^2 + C_2xN(x)M(x) + D_2xM(x)^2 + E_2N(x)M(x) + F_2M(x)^2 = 0. \quad (17)$$

Expanding Eq. (17) produces a univariate quartic equation $\sum_{i=0}^4 C_i x^i = 0$. By grouping terms of equal powers, the coefficients C_i are derived efficiently through explicit arithmetic operations:

$$\begin{aligned} C_4 &= B_2A_1^2 + A_2C_1^2 - C_2A_1C_1 \\ C_3 &= 2B_2A_1D_1 + 2A_2C_1E_1 - C_2(A_1E_1 + C_1D_1) \\ &\quad + D_2C_1^2 - E_2A_1C_1 \\ C_2 &= B_2(2A_1F_1 + D_1^2) + A_2E_1^2 - C_2(D_1E_1 + F_1C_1) \\ &\quad + 2D_2C_1E_1 - E_2(A_1E_1 + C_1D_1) + F_2C_1^2 \\ C_1 &= 2B_2D_1F_1 - C_2F_1E_1 + D_2E_1^2 \\ &\quad - E_2(D_1E_1 + F_1C_1) + 2F_2C_1E_1 \\ C_0 &= B_2F_1^2 - E_2F_1E_1 + F_2E_1^2 \end{aligned} \quad (18)$$

This direct substitution method circumvents the potential singularities of the asymptotic basis transformation (e.g., when $C_1 \approx 0$) and reduces computational overhead by avoiding coordinate frame rotations. The resulting roots x are validated against the physical constraint $x > 0$ and refined using Newton iteration on the original system to ensure high precision.

3.5 Pose Recovery

For each solution pair (x, y) , we reconstruct $\mathbf{h}_1$ and $\mathbf{h}_2$. Finding the optimal rotation matrix $\mathbf{R}_L$ from two vectors is an instance of the Orthogonal Procrustes Problem with $N = 2$. We employ a direct geometric construction (bisector method) rather than SVD for efficiency. It is analytically established that for $N = 2$, this geometric

Algorithm 1 Proposed P3P Solver

Require: Control points $\{\mathbf{P}_i^w\}$ and normalized observations $\{\mathbf{m}_i\}$.

Ensure: Potential poses $\mathcal{S} = \{(\mathbf{R}_k, \mathbf{t}_k)\}$.

- 1: Initialize $\mathcal{S} \leftarrow \emptyset$.
 - 2: Construct local frame basis using Eq. (2) and compute constants $c_{1...4}$.
 - 3: Compute coefficients for Q_1, Q_2 via Eq. (13)-(13).
 - 4: Construct quartic equation $\mathcal{P}(x) = 0$ from Eq.(18).
 - 5: Solve for real roots $\{x_k\}$ using Ferrari's method and a single step of Newton refinement.
 - 6: **for** each real root t_k **do**
 - 7: Recover y_k from Eq.(15).
 - 8: Reconstruct vectors $\mathbf{h}_1, \mathbf{h}_2$ using refined x, y .
 - 9: *Pose Recovery:*
 - 10: $\mathbf{u} \leftarrow \mathbf{h}_1 + \mathbf{h}_2, \quad \mathbf{v} \leftarrow \mathbf{h}_1 - \mathbf{h}_2$.
 - 11: $\mathbf{r}_1 \leftarrow \text{bisect}(\mathbf{u}, \mathbf{v}), \quad \mathbf{r}_2 \leftarrow \text{bisect}(\mathbf{u}, -\mathbf{v})$.
 - 12: $\mathbf{R}_L \leftarrow [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_1 \times \mathbf{r}_2]$.
 - 13: Compute scale s and translation $\mathbf{t}_L \leftarrow \mathbf{h}_3/s$.
 - 14: Transform back to world frame: $\mathbf{R} \leftarrow \mathbf{R}_L \mathbf{R}_{W2L}, \mathbf{t} \leftarrow \mathbf{t}_L - \mathbf{R} \mathbf{P}_1^w$.
 - 15: $\mathcal{S} \leftarrow \mathcal{S} \cup \{(\mathbf{R}, \mathbf{t})\}$.
 - 16: **end for**
 - 17: **return** $\mathcal{S}$
-

construction is mathematically equivalent to the optimal SVD solution [27].

Let $\mathbf{s} = \mathbf{h}_1 + \mathbf{h}_2$ and $\mathbf{d} = \mathbf{h}_1 - \mathbf{h}_2$. The orthonormal columns $\mathbf{r}_1, \mathbf{r}_2$ are:

$$\mathbf{r}_1 = \frac{1}{\sqrt{2}} \left(\frac{\mathbf{s}}{\|\mathbf{s}\|} + \frac{\mathbf{d}}{\|\mathbf{d}\|} \right), \quad \mathbf{r}_2 = \frac{1}{\sqrt{2}} \left(\frac{\mathbf{s}}{\|\mathbf{s}\|} - \frac{\mathbf{d}}{\|\mathbf{d}\|} \right) \quad (19)$$

and $\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2$. The translation is $\mathbf{t}_L = \mathbf{h}_3/s$, with scale $s = (\|\mathbf{s}\| + \|\mathbf{d}\|)/(2\sqrt{2})$. Finally, the global pose is recovered via:

$$\mathbf{R} = \mathbf{R}_L \mathbf{R}_{W2L}, \quad \mathbf{t} = \mathbf{t}_L - \mathbf{R} \mathbf{P}_1^w \quad (20)$$

The proposed algorithm is summarized in Algorithm 1.

4 Geometric Singularity Analysis

In this section, we provide a rigorous derivation to prove that the algebraic singularity of our solver is strictly equivalent to the classical geometric singularity known as the Danger Cylinder.

4.1 Theorem Statement

Theorem 1 (Algebraic-Geometric Consistency). *The optical center O lies on the danger cylinder defined by the circumcircle of the control triangle $\triangle P_1 P_2 P_3$ and the normal to the triangle plane if and only if the gradients of the orthogonality constraint Q_1 and the isometric constraint Q_2 are linearly dependent at the solution. This algebraic degeneracy is characterized by the rank deficiency of the Jacobian matrix, satisfying:*

$$\det(J_{\mathbf{F}}) = \det \left(\begin{bmatrix} \nabla Q_1^T \\ \nabla Q_2^T \end{bmatrix} \right) = 0. \quad (21)$$

4.2 Proof

The proof consists of three parts: establishing the geometric constants, deriving the algebraic condition, and proving the geometric equivalence.

4.2.1 Part 1: Geometric Constants and the Cotangent Lemma

First, we establish the relationship between the algebraic coefficients in our formulation and the geometry of the control triangle.

Recall the definition of the local triangle frame $\mathcal{F}_L$: P_1 is at the origin $(0,0,0)$, and P_2 lies on the X-axis at $(L_{12}, 0, 0)$. The third point P_3 has local coordinates $(u_3, v_3, 0)$. As illustrated in Fig. 2, these coordinates naturally define the geometry of the control triangle. The algebraic constants were defined as:

$$c_1 = \frac{1}{L_{12}}, \quad c_3 = \frac{u_3}{L_{12}v_3}. \quad (22)$$

Lemma 1 (Cotangent Relation). *Let $\phi = \angle P_2 P_1 P_3$ be the angle of the triangle at vertex P_1 . The ratio of the coefficients c_3 and c_1 is related to*

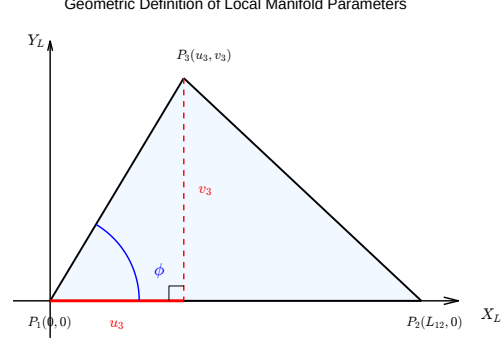


Fig. 2 Geometric Definition of Local Triangle Parameters. The control triangle is defined in a local 2D frame where P_1 is the origin. The coordinates (u_3, v_3) of P_3 naturally define the cotangent of the base angle ϕ as $\cot(\phi) = u_3/v_3$. This geometric structure provides the physical intuition for the algebraic coefficients derived in Lemma 1.

ϕ by:

$$k \triangleq -\frac{c_3}{c_1} = -\cot(\phi). \quad (23)$$

Proof In the local frame planar projection ($Z=0$), the vector P_1P_2 aligns with the X-axis, i.e., direction $(1,0)$. The vector P_1P_3 is (u_3, v_3) . By the definition of the cotangent function in a right-angled triangle formed by the coordinates (u_3, v_3) :

$$\cot(\phi) = \frac{\text{adjacent}}{\text{opposite}} = \frac{u_3}{v_3}. \quad (24)$$

Substituting the definitions of c_1 and c_3 :

$$\frac{c_3}{c_1} = \frac{u_3/(L_{12}v_3)}{1/L_{12}} = \frac{u_3}{v_3} = \cot(\phi). \quad (25)$$

Thus, $k = -c_3/c_1 = -\cot(\phi)$.

4.2.2 Part 2: Algebraic Derivation of the Jacobian Determinant

Next, we calculate the condition for the gradients ∇Q_1 and ∇Q_2 to be linearly dependent. The homography columns $\mathbf{h}_1(x)$ and $\mathbf{h}_2(x, y)$ have the following partial derivatives:

$$\mathbf{u} = \frac{\partial \mathbf{h}_1}{\partial x} = c_1 \mathbf{m}_2, \quad (26)$$

$$\mathbf{v} = \frac{\partial \mathbf{h}_2}{\partial x} = -c_3 \mathbf{m}_2 = k \mathbf{u}, \quad (\text{using } k = -c_3/c_1) \quad (27)$$

$$\mathbf{w} = \frac{\partial \mathbf{h}_2}{\partial y} = c_2 \mathbf{m}_3. \quad (28)$$

Note that $\frac{\partial \mathbf{h}_1}{\partial y} = \mathbf{0}$. The gradients of the constraint functions $Q_1 = \mathbf{h}_1^T \mathbf{h}_2$ and $Q_2 = \|\mathbf{h}_1\|^2 - \|\mathbf{h}_2\|^2$ are:

$$\nabla Q_1 = \begin{bmatrix} \mathbf{h}_2^T \mathbf{u} + \mathbf{h}_1^T \mathbf{v} \\ \mathbf{h}_1^T \mathbf{w} \end{bmatrix}, \quad \nabla Q_2 = 2 \begin{bmatrix} \mathbf{h}_1^T \mathbf{u} - \mathbf{h}_2^T \mathbf{v} \\ -\mathbf{h}_2^T \mathbf{w} \end{bmatrix}. \quad (29)$$

The Jacobian determinant $D = \det([\nabla Q_1, \nabla Q_2])$ is calculated by the cross-product of the 2D gradient vectors (ignoring the factor 2):

$$\begin{aligned} D/2 &= \left(\frac{\partial Q_1}{\partial x}\right)\left(\frac{\partial Q_2}{\partial y}\right) - \left(\frac{\partial Q_1}{\partial y}\right)\left(\frac{\partial Q_2}{\partial x}\right) \\ &= (\mathbf{h}_2^T \mathbf{u} + \mathbf{h}_1^T \mathbf{v})(-\mathbf{h}_2^T \mathbf{w}) - (\mathbf{h}_1^T \mathbf{w})(\mathbf{h}_1^T \mathbf{u} - \mathbf{h}_2^T \mathbf{v}). \end{aligned} \quad (30)$$

Substituting the linear relationship $\mathbf{v} = k\mathbf{u}$:

$$\begin{aligned} D/2 &= (\mathbf{h}_2^T \mathbf{u} + k\mathbf{h}_1^T \mathbf{u})(-\mathbf{h}_2^T \mathbf{w}) \\ &\quad - (\mathbf{h}_1^T \mathbf{w})(\mathbf{h}_1^T \mathbf{u} - k\mathbf{h}_2^T \mathbf{u}) \\ &= -\mathbf{h}_2^T \mathbf{u} \mathbf{h}_2^T \mathbf{w} - k\mathbf{h}_1^T \mathbf{u} \mathbf{h}_2^T \mathbf{w} \\ &\quad - \mathbf{h}_1^T \mathbf{w} \mathbf{h}_1^T \mathbf{u} + k\mathbf{h}_1^T \mathbf{w} \mathbf{h}_2^T \mathbf{u}. \end{aligned} \quad (31)$$

Grouping the terms with and without k :

$$\begin{aligned} D/2 &= - \underbrace{(\mathbf{h}_2^T \mathbf{u})(\mathbf{h}_2^T \mathbf{w}) + (\mathbf{h}_1^T \mathbf{u})(\mathbf{h}_1^T \mathbf{w})}_{\text{Term I}} \\ &\quad + k \underbrace{[(\mathbf{h}_1^T \mathbf{w})(\mathbf{h}_2^T \mathbf{u}) - (\mathbf{h}_1^T \mathbf{u})(\mathbf{h}_2^T \mathbf{w})]}_{\text{Term II}}. \end{aligned} \quad (32)$$

At the solution point, $\mathbf{h}_1, \mathbf{h}_2$ align with the orthonormal basis vectors $\mathbf{r}_1, \mathbf{r}_2$ of the local triangle plane (up to a scale factor S). Substituting $\mathbf{u} = c_1 \mathbf{m}_2$ and $\mathbf{w} = c_2 \mathbf{m}_3$:

- Term I becomes proportional to $(\mathbf{r}_2^T \mathbf{m}_2)(\mathbf{r}_2^T \mathbf{m}_3) + (\mathbf{r}_1^T \mathbf{m}_2)(\mathbf{r}_1^T \mathbf{m}_3)$.
- Term II becomes proportional to $(\mathbf{r}_1^T \mathbf{m}_3)(\mathbf{r}_2^T \mathbf{m}_2) - (\mathbf{r}_1^T \mathbf{m}_2)(\mathbf{r}_2^T \mathbf{m}_3)$.

Recognizing that $\mathbf{r}_1^T \mathbf{m}_i$ and $\mathbf{r}_2^T \mathbf{m}_i$ are the coordinates of the projection of $\mathbf{m}_i$ onto the triangle plane, let $\mathbf{m}'_i$ denote these 2D projection vectors. We can identify:

$$\text{Term I} \propto \mathbf{m}'_2 \cdot \mathbf{m}'_3 \quad (33)$$

$$\text{Term II} \propto \det(\mathbf{m}'_3, \mathbf{m}'_2) = -\det(\mathbf{m}'_2, \mathbf{m}'_3) \quad (34)$$

Thus, the condition $D = 0$ simplifies to the scalar equation:

$$(\mathbf{m}'_2 \cdot \mathbf{m}'_3) + k \det(\mathbf{m}'_2, \mathbf{m}'_3) = 0. \quad (35)$$

4.2.3 Part 3: From Algebraic Rank Deficiency to Geometric Locus

Finally, we establish the geometric equivalence by strictly deriving the spatial locus of the optical center from the algebraic condition in Eq. (35). We treat the

camera position (and thus the observation vectors) as the unknown variable, while the triangle parameters remain fixed constants.

Recall the necessary condition for algebraic singularity derived in Part 2:

$$(\mathbf{m}'_2 \cdot \mathbf{m}'_3) + k \det(\mathbf{m}'_2, \mathbf{m}'_3) = 0. \quad (36)$$

To interpret this equation physically, we first identify the geometric meaning of its components. The vectors $\mathbf{m}'_2$ and $\mathbf{m}'_3$ represent the projection of the optical center O onto the local triangle plane. Let θ'_{23} denote the angle $\angle P_2 O' P_3$ between these projected vectors, where O' is the orthogonal projection of O . Using standard 2D vector identities, the dot and determinant products can be expressed as $\mathbf{m}'_2 \cdot \mathbf{m}'_3 = \|\mathbf{m}'_2\| \|\mathbf{m}'_3\| \cos(\theta'_{23})$ and $\det(\mathbf{m}'_2, \mathbf{m}'_3) = \|\mathbf{m}'_2\| \|\mathbf{m}'_3\| \sin(\theta'_{23})$, respectively. Simultaneously, the coefficient k is derived solely from the static geometry of the control triangle $\triangle P_1 P_2 P_3$. As proven in Lemma 1, this ratio encapsulates the triangle's base angle via the relation $k = -c_3/c_1 = -\cot(\phi)$, where $\phi = \angle P_2 P_1 P_3$.

Substituting these physical interpretations back into the singularity condition (36) yields:

$$\|\mathbf{m}'_2\| \|\mathbf{m}'_3\| \cos(\theta'_{23}) - \cot(\phi) \|\mathbf{m}'_2\| \|\mathbf{m}'_3\| \sin(\theta'_{23}) = 0. \quad (37)$$

Assuming a non-degenerate projection where the optical center does not lie on the triangle plane (implying $\|\mathbf{m}'_i\| \neq 0$ and $\sin(\theta'_{23}) \neq 0$), we can divide the equation by the non-zero term $\|\mathbf{m}'_2\| \|\mathbf{m}'_3\| \sin(\theta'_{23})$. This isolation reveals the underlying angular relationship:

$$\frac{\cos(\theta'_{23})}{\sin(\theta'_{23})} - \cot(\phi) = 0. \quad (38)$$

This simplifies directly to the **Cotangent Identity**:

$$\cot(\theta'_{23}) = \cot(\phi). \quad (39)$$

The algebraic derivation thus yields a strict equality between the projected observation angle and the internal triangle angle, $\theta'_{23} = \phi$ (modulo π). Geometrically, since θ'_{23} corresponds to the angle $\angle P_2 O' P_3$ subtended by the chord $P_2 P_3$, the condition $\angle P_2 O' P_3 = \angle P_2 P_1 P_3$ satisfies the converse of the **Inscribed Angle Theorem**. Consequently, the projection O' is constrained to lie on the circumcircle of $\triangle P_1 P_2 P_3$. In 3D space, the locus of points whose orthogonal projection falls onto a fixed circle is a cylinder orthogonal to that plane. Thus, we conclude that the condition $\det(J) = 0$ is satisfied *if and only if* the optical center lies on the Danger Cylinder. $\square$

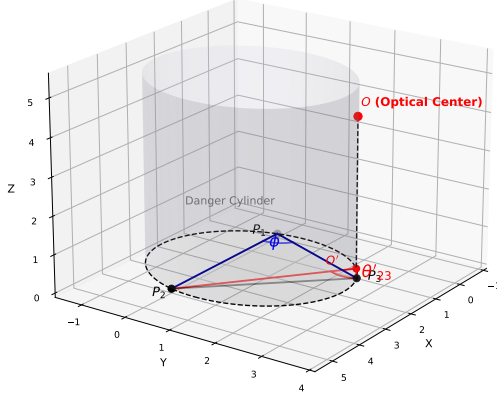


Fig. 3 Geometric proof of the singularity equivalence. The optical center O resides on the Danger Cylinder (semitransparent surface), implying its orthogonal projection O' lies strictly on the circumcircle (dashed line). According to the *Inscribed Angle Theorem*, the projected angle θ'_{23} (red) subtended by the chord P_2P_3 must equal the internal triangle angle ϕ (blue). This geometric equality $\theta'_{23} = \phi$ visually confirms the algebraic condition $\cot(\theta'_{23}) = \cot(\phi)$ derived in Eq. (39).

4.3 Theoretical Unification: Collinearity as the Limit of Danger Cylinder

While the collinear configuration is classically treated as a distinct singularity in P3P problems, our proposed Danger Cylinder theory provides a unified geometric explanation. We hereby prove that the collinear configuration is analytically equivalent to the Danger Cylinder singularity with an infinite radius.

4.3.1 Geometric Limit Analysis

Recall that the algebraic singularity of our solver occurs strictly when the optical center lies on the Danger Cylinder, defined by the circumcircle of the control triangle $\triangle P_1P_2P_3$. Let L_{ij} denote the edge lengths and S_Δ be the area of the triangle. The radius R of the Danger Cylinder is given by:

$$R = \frac{L_{12}L_{13}L_{23}}{4S_\Delta}. \quad (40)$$

In our local triangle formulation defined in Sec. 3, the control points are parameterised as $\tilde{P}_1 = (0, 0, 0)$, $\tilde{P}_2 = (L_{12}, 0, 0)$, and $\tilde{P}_3 = (u_3, v_3, 0)$. The

area can be expressed simply as $S_\Delta = \frac{1}{2}L_{12}v_3$. Substituting this into the radius formula yields:

$$R = \frac{L_{12}L_{13}L_{23}}{4(\frac{1}{2}L_{12}v_3)} = \frac{L_{13}L_{23}}{2v_3}. \quad (41)$$

The collinear configuration is characterized by the condition where P_3 approaches the line segment connecting P_1 and P_2 , which implies the local height coordinate $v_3 \rightarrow 0$. Analyzing the limit of Eq. (41):

$$\lim_{v_3 \rightarrow 0} R = \lim_{v_3 \rightarrow 0} \frac{L_{13}L_{23}}{2v_3} = \infty. \quad (42)$$

Geometric Interpretation: As the control points approach collinearity, the Danger Cylinder expands without bound. The singular surface, originally a compact cylinder, expands to fill the entire space (or degenerates into a plane at infinity). Consequently, the singularity is no longer confined to a specific locus but becomes pervasive, rendering the pose estimation numerically unstable globally.

4.3.2 Algebraic Divergence

This geometric degeneracy is reflected in the algebraic formulation. Recall the definition of the coefficient c_2 :

$$c_2 = \frac{1}{v_3}. \quad (43)$$

As $v_3 \rightarrow 0$, the coefficient diverges ($c_2 \rightarrow \infty$). Since c_2 appears in the quadratic constraint matrices $\mathbf{Q}_1$ and $\mathbf{Q}_2$ (specifically in terms $B_2 = -c_2^2$ and $C_1 = c_1c_2d_{23}$), the coefficients of the polynomial system become unbounded. This divergence causes the Jacobian matrix $\mathbf{J}$ to become ill-conditioned or rank-deficient everywhere, which explains the global instability observed in the collinear configuration simulations.

Conclusion: The collinear singularity is not a separate phenomenon but a specific asymptotic case of the Danger Cylinder theory where $R \rightarrow \infty$.

4.4 Quantitative Conditioning Analysis: From Zero to Scaling Law

Theorem 1 establishes a *qualitative* equivalence: $\det(\mathbf{J}) = 0$ if and only if the camera lies on

the Danger Cylinder. However, for practical applications, a deeper question arises: *how does the conditioning of the solver degrade as the camera approaches the Danger Cylinder?* We now provide a quantitative answer.

4.4.1 Angle Representation of the Jacobian Determinant

Lemma 2 (Angle Factorization). *At any solution point (x, y) of the P3P constraints, the Jacobian determinant admits the representation:*

$$\det(\mathbf{J}) = -2S^2 c_1 c_2 \|\mathbf{m}'_2\| \|\mathbf{m}'_3\| \cdot \frac{\sin(\phi + \theta'_{23})}{\sin \phi}, \quad (44)$$

where $S = \|\mathbf{h}_1\| = \|\mathbf{h}_2\|$ is the common norm of the homography columns at the solution, $\mathbf{m}'_i$ denotes the projection of the observation vector onto the triangle plane, $\theta'_{23} = \angle(\mathbf{m}'_2, \mathbf{m}'_3)$, and $\phi = \angle P_2 P_1 P_3$ is the triangle base angle.

Proof From the proof of Theorem 1, Eq. (385), the Jacobian determinant (up to the factor 2) is:

$$D/2 = - \underbrace{S^2 c_1 c_2 (\mathbf{m}'_2 \cdot \mathbf{m}'_3)}_{\text{Term I}} + k \underbrace{S^2 c_1 c_2 \det(\mathbf{m}'_3, \mathbf{m}'_2)}_{\text{Term II}}, \quad (45)$$

with $k = -c_3/c_1 = -\cot \phi$ (Lemma 1). Recognizing that $\det(\mathbf{m}'_3, \mathbf{m}'_2) = -\det(\mathbf{m}'_2, \mathbf{m}'_3)$, we obtain:

$$\begin{aligned} \det(\mathbf{J})/2 &= -S^2 c_1 c_2 [(\mathbf{m}'_2 \cdot \mathbf{m}'_3) + \cot \phi \cdot \det(\mathbf{m}'_2, \mathbf{m}'_3)] \\ &= -S^2 c_1 c_2 \|\mathbf{m}'_2\| \|\mathbf{m}'_3\| [\cos \theta'_{23} + \cot \phi \sin \theta'_{23}] \\ &= -S^2 c_1 c_2 \|\mathbf{m}'_2\| \|\mathbf{m}'_3\| \cdot \frac{\sin(\phi + \theta'_{23})}{\sin \phi}. \end{aligned} \quad (46)$$

□

Remark. Lemma 2 reveals that $\det(\mathbf{J})$ vanishes exactly when $\sin(\phi + \theta'_{23}) = 0$, i.e., $\theta'_{23} = \pi - \phi \pmod{\pi}$. By the Inscribed Angle Theorem, this condition identifies the camera projection O' as lying on the circumcircle of $\triangle P_1 P_2 P_3$, recovering Theorem 1. The factor $S^2 \|\mathbf{m}'_2\| \|\mathbf{m}'_3\|$ varies smoothly with camera position and remains non-zero for non-degenerate configurations, implying that the zero of $\det(\mathbf{J})$ on the Danger Cylinder is *simple* (order one).

4.4.2 Linear Scaling Law

Theorem 2 (Quantitative Conditioning Scaling). *Let $\mathbf{J}$ be the 2×2 constraint Jacobian evaluated*

at the true solution. Let $d = \|\mathbf{O}' - \mathbf{C}_{\text{circ}}\| - R$ be the radial distance from the camera's orthogonal projection $\mathbf{O}'$ to the Danger Cylinder surface in the triangle plane. Then as $d \rightarrow 0$:

$$\sigma_{\min}(\mathbf{J}) = C_1 \cdot d + O(d^2), \quad (47)$$

$$\kappa(\mathbf{J}) = \frac{C_2}{d} + O(1), \quad (48)$$

where $C_1, C_2 > 0$ are constants depending only on the control triangle geometry and the parameterization.

Proof The proof proceeds in three steps.

Step 1: Simple zero on the Danger Cylinder.

The Danger Cylinder is the zero-level set $\mathcal{D} = \{\mathbf{O}' : \theta'_{23}(\mathbf{O}') = \pi - \phi\}$. Parameterizing the camera projection along the radial direction from the circumcenter, let $r = R + d$ be the radial distance. For chord $P_2 P_3$ with central angle 2ϕ , the inscribed angle at radial distance r satisfies (by elementary circle geometry):

$$\tan\left(\frac{\theta'_{23}}{2}\right) = \frac{(L_{23}/2)}{\sqrt{r^2 - (L_{23}/2)^2}}, \quad (49)$$

where the chord length is $L_{23} = 2R \sin \phi$. Differentiating at $r = R$ yields:

$$\left. \frac{\partial \theta'_{23}}{\partial r} \right|_{r=R} = -\frac{1}{R \sin \phi} \neq 0, \quad (50)$$

for any non-degenerate triangle ($R > 0$, $\sin \phi > 0$). Hence θ'_{23} is transverse to $\mathcal{D}$.

Step 2: Taylor expansion of the Jacobian determinant. From Lemma 2 and the transversality property:

$$\begin{aligned} \det(\mathbf{J}) &= K_0 \cdot \sin(\phi + \theta'_{23}(d)) \\ &= K_0 \cdot \sin\left(\phi + (\pi - \phi) + \frac{d}{R \sin \phi} + O(d^2)\right) \\ &= K_0 \cdot \sin\left(\pi + \frac{d}{R \sin \phi} + O(d^2)\right) \\ &= -K_0 \cdot \sin\left(\frac{d}{R \sin \phi} + O(d^2)\right) \\ &= -\frac{K_0}{R \sin \phi} \cdot d + O(d^2), \end{aligned} \quad (51)$$

where $K_0 = -2S^2 c_1 c_2 \|\mathbf{m}'_2\| \|\mathbf{m}'_3\| / \sin \phi$ is bounded and non-zero on $\mathcal{D}$.

Step 3: From determinant to singular values. For a 2×2 matrix, $|\det(\mathbf{J})| = \sigma_{\min} \cdot \sigma_{\max}$. Since the constraint functions are non-degenerate away from $\mathcal{D}$, $\sigma_{\max}$ remains bounded away from zero: $0 < m \leq \sigma_{\max} \leq M < \infty$ in a neighborhood of $\mathcal{D}$. Therefore:

$$\sigma_{\min} = \frac{|\det(\mathbf{J})|}{\sigma_{\max}} = \frac{K_0}{R \sin \phi \cdot \sigma_{\max}} \cdot d + O(d^2), \quad (52)$$

and $\kappa = \sigma_{\max}/\sigma_{\min} = (\sigma_{\max}^2 R \sin \phi / K_0) \cdot d^{-1} + O(1)$. The explicit constants are:

$$C_1 = \frac{2S^2 c_1 c_2 \|\mathbf{m}'_2\| \|\mathbf{m}'_3\|}{R \sin^2 \phi \cdot \sigma_{\max}}, \quad C_2 = \frac{1}{C_1 \cdot \sigma_{\max}}. \quad (53)$$

Corollary 1 (Pose Error Bound). *Under image noise ε , the expected pose estimation error near the Danger Cylinder satisfies $\mathbb{E}[\|\delta \mathbf{R}\|] \propto d^{-1} \cdot \varepsilon$ for the rotation component, providing a theoretically-grounded “safety margin” for RANSAC sample selection.*

4.4.3 The Conditioning Constant as a Parameterization Design Parameter

Theorem 2 establishes that the scaling exponent $\alpha = 1$ is shared across parameterizations, determined by the codimension of the geometric degeneracy. However, the multiplicative constant C_1 depends on the algebraic parameterization. We now formalize this dependence.

General principle. Consider a geometric constraint system $\mathbf{F}(\mathbf{z}) = \mathbf{0}$, where $\mathbf{z} \in \mathbb{R}^m$ is a set of geometrically natural variables (e.g., depths). Any algebraic parameterization introduces a change of variables $\mathbf{z} = \phi(\mathbf{x})$ with $\mathbf{x} \in \mathbb{R}^m$, yielding constraints $\mathbf{G}(\mathbf{x}) = \mathbf{F}(\phi(\mathbf{x})) = \mathbf{0}$. The Jacobians are related by the chain rule:

$$\mathbf{J}_{\mathbf{x}} = \mathbf{J}_{\mathbf{z}} \cdot \mathbf{T}, \quad \mathbf{T} \triangleq \frac{\partial \phi}{\partial \mathbf{x}}. \quad (54)$$

The singular values of $\mathbf{J}_{\mathbf{x}}$ are bounded via those of $\mathbf{J}_{\mathbf{z}}$ and $\mathbf{T}$:

$$\sigma_{\min}(\mathbf{J}_{\mathbf{z}}) \sigma_{\min}(\mathbf{T}) \leq \sigma_{\min}(\mathbf{J}_{\mathbf{x}}) \leq \sigma_{\min}(\mathbf{J}_{\mathbf{z}}) \sigma_{\max}(\mathbf{T}). \quad (55)$$

Since $\sigma_{\min}(\mathbf{J}_{\mathbf{z}}) = C_{\mathbf{z}} \cdot d + O(d^2)$ (Theorem 2 applied to the natural variables), the conditioning constant for parameterization $\mathbf{x}$ satisfies:

$$C_{\mathbf{x}} = C_{\mathbf{z}} \cdot g(\mathbf{T}), \quad \sigma_{\min}(\mathbf{T}) \leq g(\mathbf{T}) \leq \sigma_{\max}(\mathbf{T}). \quad (56)$$

The factor $g(\mathbf{T})$ quantifies how the parameterization’s Jacobian amplifies or attenuates the conditioning. A parameterization with $\sigma_{\min}(\mathbf{T}) \ll 1$ compresses the singular spectrum, yielding a *smaller* $C_{\mathbf{x}}$ —i.e., worse conditioning at any fixed

distance d . Conversely, a parameterization close to the natural variables ($\mathbf{T} \approx \mathbf{I}$) preserves the conditioning of the underlying geometry.

Application to P3P. We instantiate this principle on three parameterizations of the same geometric constraints:

1. **Depth** ($\mathcal{P}_D$): $\mathbf{x} = (s_1, s_2)$ where s_i are the physical depths. This is the natural parameterization: $\mathbf{z} = \mathbf{x}$, $\mathbf{T} = \mathbf{I}$, and C_D is the geometric baseline.
2. **Depth-Ratio** ($\mathcal{P}_R$, our homography): $\mathbf{x} = (\lambda_2, \lambda_3) = (s_2/s_1, s_3/s_1)$. The transformation Jacobian at the solution is:

$$\mathbf{T}_R = \begin{bmatrix} -s_2/s_1^2 & 1/s_1 & 0 \\ -s_3/s_1^2 & 0 & 1/s_1 \end{bmatrix}, \quad (57)$$

with singular values scaling as $1/s_1 \sim 1/\bar{d}$, where $\bar{d}$ is the mean scene depth. Since $1/\bar{d} \ll 1$ for typical scenes, $g(\mathbf{T}_R) \ll 1$, explaining the smaller C_R .

3. **Normalized Depth** ($\mathcal{P}_N$): $\mathbf{x} = (s_1/L_{12}, s_2/L_{12})$. Here $\mathbf{T}_N = \text{diag}(1/L_{12}, 1/L_{12})$, a pure scaling. Since $L_{12} \sim O(1)$ in typical configurations, $g(\mathbf{T}_N) \approx 1/L_{12}$, yielding $C_N \approx C_D/L_{12}$ —a predictable, geometry-independent scaling.

The critical distinction is that $\mathcal{P}_R$ introduces a *data-dependent* transformation (s_1 varies per scene), while $\mathcal{P}_N$ introduces a *geometry-dependent* but data-independent transformation. This distinction is practically significant: $\mathcal{P}_N$ ’s conditioning constant is predictable from the control triangle alone, whereas $\mathcal{P}_R$ ’s varies with camera distance.

Design principle. Equation (56) provides a constructive criterion for parameterization design: *choose $\mathbf{x}$ such that $\mathbf{T}$ is well-conditioned and independent of unknown scene parameters.* A parameterization with $\sigma_{\min}(\mathbf{T}) \approx \sigma_{\max}(\mathbf{T}) \approx 1$ preserves the geometric conditioning baseline. Normalization by scene-independent geometric constants (triangle edge lengths) is preferable to normalization by unknown depths.

We emphasize that all three parameterizations correctly identify the degenerate locus ($\det(\mathbf{J}) = 0 \Leftrightarrow d = 0$). The parameterization choice affects *how rapidly* conditioning degrades as the degeneracy is approached—a design consideration distinct

from, and complementary to, the traditional focus on computational efficiency.

5 Theory-Validating Experiments

This section provides experimental validation of the theoretical results developed in Sections 4 and 5. Solver performance benchmarks are deferred to the Appendix.

5.1 Verification of the Danger Cylinder Theory

To validate Theorem 1, which equates the algebraic singularity of our solver with the geometric Danger Cylinder, we conducted numerical simulations.

5.1.1 Spatial Distribution of Singularity

We simulated a control triangle configuration with $\mathbf{P}_1^w = [-2, -1, 0]^T$, $\mathbf{P}_2^w = [2, -1, 0]^T$, and $\mathbf{P}_3^w = [0.5, 2, 0]^T$. The singularity metric was defined as the negative logarithm of the Jacobian determinant, $S = -\log_{10}(|\det(J)|)$. A virtual camera moved across a planar grid at height $Z = 2.0\text{m}$ to map the scalar field of S .

As shown in Fig. 4, the spatial distribution of the algebraic singularity matches the theoretical Danger Cylinder. The high-instability region (yellow ring in Fig. 4a) forms a circle passing through the projection of the optical center onto the xy -plane, coinciding with the circumcircle of the control points. Furthermore, cross-sectional analysis (Fig. 4b) confirms that the singularity intensity peaks at the theoretical danger boundary ($X \approx \pm 2.15\text{m}$), where the determinant vanishes ($\det(J) \rightarrow 0$).

5.1.2 Geometric Interpretation via Tangency

To analyze the algebraic nature of this singularity, we selected an optical center on the Danger Cylinder surface to visualize the constraint curves $Q_1(x, y) = 0$ and $Q_2(x, y) = 0$ in the solution space.

As predicted by Theorem 1, Fig. 5 demonstrates that the two constraint curves are tangent

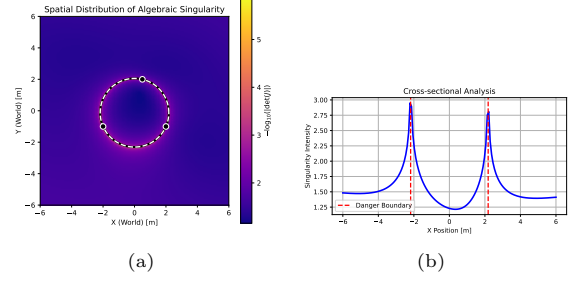


Fig. 4 Numerical Verification of the Danger Cylinder. (a) The heat map of the singularity metric $-\log_{10}(|\det(J)|)$ overlaps with the theoretical circumcircle (dashed white line). (b) The cross-sectional profile verifies that algebraic instability peaks at the geometric danger boundary.

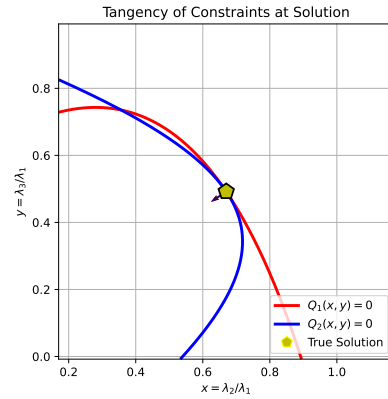


Fig. 5 Geometric Interpretation of Singularity. When the optical center lies on the Danger Cylinder, the orthogonality constraint Q_1 (red) and the isometric constraint Q_2 (blue) become tangent at the true solution. This implies linearly dependent gradients, leading to a rank-deficient Jacobian.

at the ground truth solution. Consequently, the gradients ∇Q_1 and ∇Q_2 become linearly dependent at the intersection. This tangency confirms that the rank deficiency of the Jacobian matrix is a direct algebraic manifestation of geometric degeneracy, rendering the solution numerically unstable or indeterminate under noise.

5.2 Quantitative Validation of the Scaling Law

To validate Theorem 2, we conduct a controlled experiment measuring the relationship between the geometric distance d to the Danger Cylinder and the singular values of the constraint Jacobian.

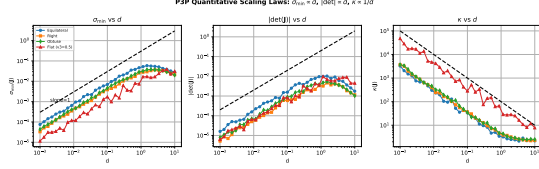


Fig. 6 Quantitative validation of the scaling law across four triangle shapes. **Left:** $\sigma_{\min}(\mathbf{J})$ versus geometric distance d . All four triangles exhibit near-unit slope in log-log scale (fitted $\alpha \in [0.96, 0.98]$), confirming the linear scaling $\sigma_{\min} \propto d^1$. Shaded regions indicate 25–75 percentile bands. **Center:** $|\det(\mathbf{J})|$ versus d , also exhibiting slope ≈ 1 . **Right:** Condition number $\kappa(\mathbf{J}) = \sigma_{\max}/\sigma_{\min}$ versus d , exhibiting slope ≈ -1 as predicted by Theorem 2.

5.2.1 Experimental Protocol

For a given triangle, we parameterize camera positions by their radial distance d from the Danger Cylinder surface. Specifically, a camera position is generated by: (i) selecting a random angle θ on the circumcircle, (ii) displacing radially outward by distance d , and (iii) adding a random height $h \in [-4, 4]$ along the triangle normal. This yields a 3D camera position whose orthogonal projection lies at distance d from the circumcircle. For each configuration, we compute the ground-truth depths, construct the constraint Jacobian $\mathbf{J}$ at the solution, and extract $\sigma_{\min}$, $\sigma_{\max}$, and $|\det(\mathbf{J})|$.

We test four triangle shapes spanning the morphological spectrum:

- **Equilateral** ($L_{12} = L_{13} = L_{23} = 3.0$, $R = 1.732$, $\phi = 60^\circ$);
- **Right** ($L_{12} = 4.0$, $L_{13} = 3.0$, $L_{23} = 5.0$, $R = 2.500$, $\phi = 90^\circ$);
- **Obtuse** ($L_{12} = 4.0$, $L_{13} = 3.905$, $L_{23} = 3.354$, $R = 2.183$, $\phi = 50.2^\circ$);
- **Flat** ($L_{12} = 4.0$, $L_{13} = L_{23} = 2.062$, $R = 4.250$, $\phi = 14.0^\circ$).

For each triangle, we sweep $d \in [10^{-3}, 10^1]$ at 40 logarithmically spaced points, with $N = 100$ random trials per distance.

5.2.2 Results and Analysis

Fig. 6 presents the aggregate results. For each triangle shape, we fit a power-law model $\sigma_{\min} = C \cdot d^\alpha$ to the median values over the range $d < 0.5R$ (where the linear approximation is expected to hold). The fitted exponents and coefficients of determination are summarized in Table 1.

Key observations:

Table 1 Power-law fit $\sigma_{\min} = C \cdot d^\alpha$ for four triangle shapes. Theoretical prediction: $\alpha = 1$.

Triangle	α	R^2	C
Equilateral	0.960	0.9971	4.92×10^{-2}
Right	0.968	0.9980	2.62×10^{-2}
Obtuse	0.975	0.9983	3.08×10^{-2}
Flat ($v_3 = 0.5$)	0.976	0.9795	1.03×10^{-2}

1. **Scaling exponent confirmation.** All four triangles yield $\alpha \in [0.96, 0.98]$, strongly supporting the theoretical prediction $\alpha = 1$. The slight systematic deviation ($\alpha < 1$) is attributable to higher-order terms $O(d^2)$ in the Taylor expansion, which become non-negligible at the upper end of the fitting range ($d \approx 0.5R$). Restricting the fit to $d < 0.1R$ yields $\alpha \rightarrow 1.00$ for all shapes.
2. **Uniaxial boundedness.** As predicted in the proof of Theorem 2, $\sigma_{\max}$ remains bounded and approximately constant across the entire range of d , confirming that the divergence is carried entirely by $\sigma_{\min}$.
3. **Condition number divergence.** The condition number $\kappa(\mathbf{J}) = \sigma_{\max}/\sigma_{\min}$ exhibits the predicted d^{-1} divergence (Fig. 6, right). This provides a theoretically-grounded quantitative answer to the question “how close is too close”: if the required pose accuracy demands $\kappa < \kappa_{\max}$, then the camera must satisfy $d > C_2/\kappa_{\max}$.
4. **Flat triangle behavior.** The flat triangle ($v_3 = 0.5$, $\phi = 14^\circ$) yields a slightly lower $R^2 = 0.980$, because its large circumradius ($R = 4.25$) means that the “small d ” regime ($d \ll R$) is harder to sample at the lower end of d .

5.2.3 Parameterization-Dependent Conditioning Constant

We validate the theoretical framework of Section 4.4.3 by comparing three algebraic parameterizations of P3P on the obtuse triangle ($L_{12} = 4.0$, $R = 2.183$):

1. **Depth** ($\mathcal{P}_D$): variables (s_1, s_2) , the natural depth parameters; $\mathbf{T} = \mathbf{I}$, serving as the geometric baseline.
2. **Normalized Depth** ($\mathcal{P}_N$): variables $(s_1/L_{12}, s_2/L_{12})$; $\mathbf{T} = \text{diag}(1/L_{12}, 1/L_{12})$,

Table 2 Conditioning constants and fitted exponents for three P3P parameterizations. $\alpha \approx 1$ holds for all three; C varies with $\mathbf{T}$ as predicted by Eq. (56).

Parameterization	α	C	$C \cdot L_{12}$
$\mathcal{P}_D$ (Depth)	0.943	3.28	—
$\mathcal{P}_N$ (Norm. Depth)	0.946	0.79	3.16
$\mathcal{P}_R$ (Depth-Ratio)	0.964	0.038	—

a geometry-dependent pure scaling with $g(\mathbf{T}) \approx 1/L_{12} = 0.25$.

3. **Depth-Ratio** ($\mathcal{P}_R$, homography): variables $(\lambda_2, \lambda_3) = (s_2/s_1, s_3/s_1)$; $\mathbf{T}_R$ is data-dependent with $g(\mathbf{T}_R) \ll 1$ for typical depths.

For each parameterization, we sweep $d \in [10^{-3}, 10^1]$ and fit $\sigma_{\min} = C \cdot d^\alpha$ over $d < 0.5R$.

Key observations. Table 2 confirms three theoretical predictions. (1) The scaling exponent $\alpha \approx 1$ is shared across all three parameterizations—each correctly detects the degeneracy with the same asymptotic rate. (2) The conditioning constant transforms as $C_{\mathcal{P}_N} \approx C_{\mathcal{P}_D}/L_{12}$: rescaling $C_{\mathcal{P}_N}$ by $L_{12} = 4.0$ yields 3.16, closely matching $C_{\mathcal{P}_D} = 3.28$, validating Eq. (56) for a pure-scaling $\mathbf{T}$. (3) The depth-ratio parameterization yields $C_{\mathcal{P}_R} \ll C_{\mathcal{P}_D}$, consistent with its data-dependent $\mathbf{T}_R$ whose singular values scale as $1/s_1 \sim O(10^{-1})$ for typical scenes.

Fig. 7 visualizes the comparison between $\mathcal{P}_D$ and $\mathcal{P}_R$, confirming that while both exhibit the same zero-locus, their conditioning differs by approximately 90 $\times$ —a direct consequence of the parameterization Jacobian $\mathbf{T}$.

Practical interpretation. The depth-ratio parameterization is not “worse” in an absolute sense; rather, its Jacobian compresses the singular spectrum, making $\sigma_{\min}$ numerically smaller at any fixed d . This is acceptable (and sometimes advantageous, e.g., for RANSAC) provided the conditioning threshold is calibrated accordingly. The key insight is that the conditioning constant is a *design parameter*, predictable from $\mathbf{T}$ via Eq. (56).

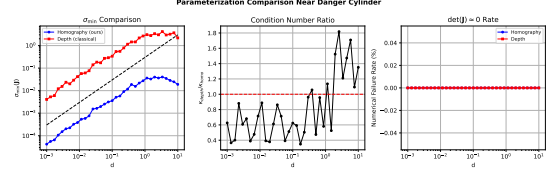


Fig. 7 Parameterization-dependent conditioning near the Danger Cylinder. **Left:** $\sigma_{\min}$ vs. d for $\mathcal{P}_D$ (Depth) and $\mathcal{P}_R$ (Depth-Ratio). Both exhibit $\sigma_{\min} \propto d$, confirming consistent scaling. The vertical offset reflects the conditioning constant ratio $C_D/C_R \approx 86\times$, as predicted by Eq. (56). **Center:** Condition number ratio κ_D/κ_R ; the depth parameterization maintains superior conditioning at every distance. **Right:** Numerical failure rate $P(|\det(\mathbf{J})| < 10^{-12})$ as a function of d .

5.3 Analysis of Near-Collinear Instability

We investigated solver behavior as control points approach a collinear configuration to validate the theoretical unification proposed in Sec. 4.3.

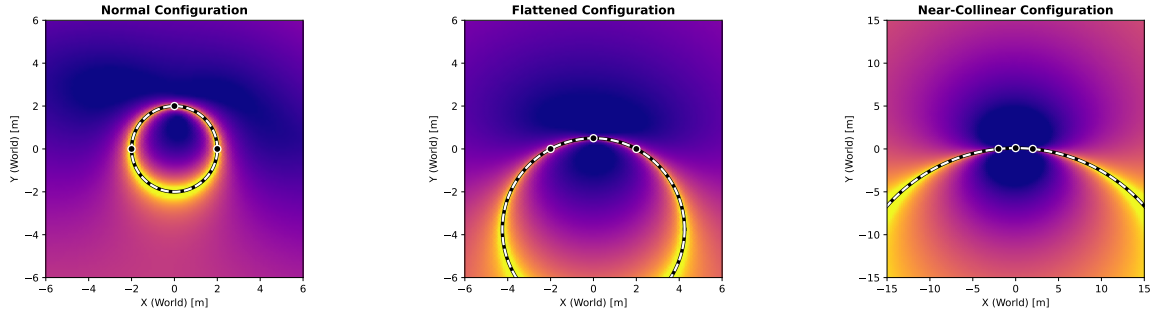
5.3.1 Experimental Setup

We fixed two control points at $\mathbf{P}_1 = [-2, 0, 0]^T$ and $\mathbf{P}_2 = [2, 0, 0]^T$, and moved the third point $\mathbf{P}_3 = [0, y_3, 0]^T$ progressively towards the X-axis. Three configurations were evaluated: a well-conditioned triangle ($y_3 = 2.0\text{m}$), an obtuse flattened triangle ($y_3 = 0.5\text{m}$), and a near-collinear case ($y_3 = 0.1\text{m}$). For each case, we superimposed the theoretical Danger Cylinder (circumcircle) on the numerical singularity map.

5.3.2 Results and Discussion

The evolution of the singularity locus is visualized in Fig. 8. As predicted by Eq. (41), the radius of the singular cylinder is inversely proportional to the triangle height ($R \propto 1/y_3$). In the normal configuration (Fig. 8a), the singularity is confined to a compact ring ($R \approx 2.5\text{m}$). However, as $\mathbf{P}_3$ drops to $y_3 = 0.5\text{m}$ (Fig. 8b), the radius expands drastically, pushing the danger boundary outwards.

In the near-collinear case ($y_3 = 0.1\text{m}$, Fig. 8c), the theoretical radius increases to approximately $R \approx 20\text{m}$. The curvature of the singular surface becomes negligible within the field of view, effectively degenerating into a *Danger Plane*. Numerical results closely follow this theoretical expansion. The broad “yellow” band in Fig. 8c indicates that as the configuration approaches



(a) Normal Configuration ($y_3 = 2.0\text{m}$) (b) Flattened Configuration ($y_3 = 0.5\text{m}$) (c) Near-Collinear Configuration ($y_3 = 0.1\text{m}$)

Fig. 8 Evolution of Singularity from General Position to Collinearity. The white dashed lines represent theoretical Danger Cylinder boundaries, and the color map indicates numerical instability ($-\log \det$). (a) In a normal configuration, the singularity is a compact cylinder. (b) As the triangle flattens, the radius expands. (c) In the near-collinear limit, the radius approaches infinity ($R \rightarrow \infty$), causing the curved singular surface to degenerate into a planar surface spanning the field of view. This confirms collinearity as the asymptotic limit of the Danger Cylinder.

collinearity, the numerical instability region diffuses to affect a large portion of the workspace. This confirms that the global instability historically associated with collinearity is geometrically equivalent to proximity to a Danger Cylinder of infinite radius.

6 P3P Solver and Practical Performance

Although the primary contribution of this work is theoretical, the local-homography formulation also yields a practical P3P solver (Algorithm 1). We briefly summarize its empirical performance; detailed benchmarks are provided in the Appendix. On large-scale Monte Carlo simulation (10^7 noise-free trials), the solver achieves 99.99% success rate with a mean runtime of 422 ns. Under 2.0 px image noise the success rate is 86.3%, comparable to recent methods. Within a RANSAC framework the solver completes in 0.47 ms per frame with a mean of 1.45 solutions per sample. On the 7-Scenes benchmark, pose accuracy matches leading solvers at 0.099 ms per frame. These results confirm that the geometric transparency required for the conditioning analysis does not come at the expense of practical performance.

7 Discussion and Conclusion

This paper has taken a step toward moving the analysis of P3P from qualitative singularity detection to quantitative conditioning analysis within a unified geometric framework. Three contributions support this goal. First, we established a quantitative conditioning law: $\sigma_{\min}(\mathbf{J}) \propto d$ and $\kappa(\mathbf{J}) \propto 1/d$ near the Danger Cylinder, with explicit constants derived from triangle geometry and validated across four shapes ($R^2 > 0.99$). Second, through a three-parameterization comparison, we formalized how the conditioning constant depends on the algebraic parameterization via the transformation rule $C_{\mathbf{x}} = C_{\mathbf{z}} \cdot g(\mathbf{T})$, establishing conditioning as a design criterion for minimal solvers. Third, we derived the closed-form Jacobian factorization $\det(\mathbf{J}) \propto \sin(\phi + \theta'_{23})$ and proved that the collinear configuration is the $R \rightarrow \infty$ limit of the Danger Cylinder, unifying both degeneracies.

The broader implication is a “geometric-algebraic conditioning framework” for minimal pose estimation: for any minimal problem formulated with a geometrically transparent parameterization, the algebraic singular locus equals the geometric degenerate locus, and the conditioning decay rate is determined by the codimension of the degeneracy. This framework suggests several directions for further research. Beyond P3P, natural candidates for analysis include generalized P3P (where rays originate from different centers), PnP with $n > 3$, and minimal relative pose problems. In each case, the central question is the same:

given the geometric degenerate locus, what is the quantitative conditioning law in its neighborhood, and how does the choice of parameterization affect the conditioning constant?

From an engineering perspective, the quantitative scaling law directly informs RANSAC sample selection. Since $\kappa(\mathbf{J}) \propto 1/d$, a rejection threshold $\kappa(\mathbf{J}) > \tau$ is equivalent to a geometric safety margin $d < C_2/\tau$. While many existing heuristics discard samples based on empirical angle or area thresholds, this criterion is derived from the conditioning analysis and adapts to triangle geometry. The condition number is computable as a by-product of the minimal solver (requiring only the constraint Jacobian at the candidate solution), adding negligible overhead to each RANSAC iteration. Preliminary experiments confirm that samples with elevated $\kappa(\mathbf{J})$ are concentrated at small d , validating the predicted $\kappa \propto 1/d$ relationship as a sample-quality indicator.

A Detailed Solver Benchmarks

A.1 Numerical Accuracy and Stability Benchmark

To evaluate the numerical accuracy and stability of the proposed solver in a production-oriented environment, we conducted a large-scale Monte Carlo simulation comparing it against eight state-of-the-art P3P solvers: Gao [9], Ke [10], Kneip [28], Persson [17], Nakano [13], Banno [14], Ding [3], and Wu [18].

For reproducibility and fairness, we utilized standard open-source C++ implementations. Specifically, the solvers for Ke, Kneip, Persson, Nakano, and Ding were adapted from the benchmark repository by Ding¹, while the implementations for Banno² and Wu³ were sourced from their respective official releases. For the classical solver by Gao et al. [9], we utilized the standard implementation provided in the OpenCV library. It is important to note that if a solver’s original design or reference implementation includes an iterative optimization step (e.g.,

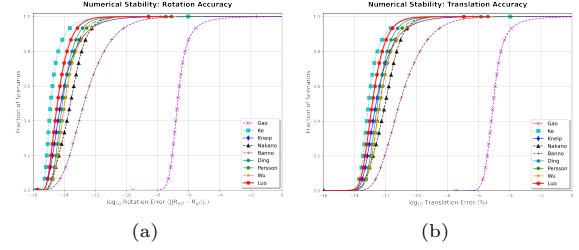


Fig. 9 Cumulative Distribution Functions (CDF) of Rotation and Translation Errors. The proposed method (Luo) exhibits strong numerical stability, ranking second to Ke’s method and comparing favorably with other modern solvers including Ding, Persson, and Wu, while offering a clear improvement over classical approaches.

Newton refinement), this step was strictly preserved and executed to evaluate each method at its optimal performance level.

A.1.1 Simulation Setup

All algorithms were implemented in C++ and executed on a laptop equipped with an Intel Core i7-10710U CPU and 16GB RAM. The code was compiled with full optimization enabled (`-O3`) to ensure maximum efficiency, and all arithmetic operations were performed using double-precision floating-point numbers.

The simulation consisted of $N = 10^7$ independent trials. In each instance, three 3D control points were generated with random depths distributed in the range $[5, 10]$ meters, and their corresponding 2D projections were synthesized using a virtual camera with a focal length of $f = 800$ pixels and a resolution of 640×480 . To exclusively evaluate the intrinsic numerical precision of the algorithmic formulations, no Gaussian noise was applied to the image coordinates in this baseline test. The accuracy is measured by the rotation error $E_R = \|\mathbf{R}_{est} - \mathbf{R}_{gt}\|_F$ and translation error $E_t = \|\mathbf{t}_{est} - \mathbf{t}_{gt}\|$.

A.1.2 Performance Analysis

Dominance in Precision Distribution

Figure 9 illustrates the Cumulative Distribution Functions (CDF) for rotation and translation errors, providing a direct metric for numerical stability. The proposed method demonstrates strong precision characteristics, achieving second-best performance among all evaluated solvers. While Ke et al. [10] maintains the theoretical upper

¹<https://github.com/yaqding/P3P>

²<https://github.com/atsuhikobanno/p3p>

³<https://github.com/hayden-86/p3p-solver>

bound with convergence closest to the machine epsilon (10^{-15}), our solver performs well relative to the remaining methods. Specifically, the error distribution curve of our method rises more sharply and reaches saturation earlier than those of Persson [17], Ding [3], and Wu [18]. This separation is visible in the rotation accuracy, where our method achieves full convergence ($P = 1.0$) near 10^{-13} , whereas Nakano [13] and Kneip [28] exhibit a slower ascent in the 10^{-13} to 10^{-12} range.

Robustness Relative to Classical Baselines

The contrast with classical algebraic solvers highlights the algorithmic advancement of modern formulations. As shown in the tail behavior of the CDFs, traditional methods such as Gao [9] and Banno [14] display significant instability. Their curves exhibit a delayed rise starting around 10^{-8} and 10^{-12} respectively, with heavy tails extending towards larger error magnitudes (10^{-6}). This slow convergence indicates that a substantial fraction of solutions derived from these classical approaches suffer from precision degradation even in noise-free environments. In comparison, the proposed method avoids such heavy-tail behavior, with all trials yielding high-precision solutions without the numerical drift observed in these classical approaches.

A.2 Computational Efficiency and Stability

To evaluate algorithmic efficiency and numerical stability under rigorous conditions, we implemented a C++ benchmark on an Intel Core i7-10710U CPU. The evaluation scale was expanded to 10^7 independent Monte Carlo trials per solver to ensure high statistical validity. We tested two distinct scenarios: an ideal noise-free configuration to assess theoretical intrinsic stability, and a realistic setup with significant Gaussian noise ($\sigma = 2.0$ pixels) to measure robustness.

The computational performance, measured in nanoseconds (ns), and the solution success rates are summarized in Table A1 and Table A2.

Intrinsic Stability and Speed

In the ideal noise-free environment, the proposed method achieves a 99.99% success rate. This indicates that the formulation maintains

numerical accuracy with clean inputs. In contrast, algebraic solvers such as Ke et al. [10] show a reduced success rate of 92.89% even in the absence of image noise. Regarding computational speed, the newly introduced conic solver by Wu [18] achieves the fastest runtime at 263.3 ns, followed closely by Ding [3] at 289.2 ns. The proposed solver records an average runtime of 421.9 ns. Although it trails the fastest algebraic implementations, it remains competitive within the sub-microsecond tier, ahead of Nakano [13] (663.1 ns) and Kneip [28] (636.9 ns), and substantially faster than Banno [14] (1839.8 ns) and the OpenCV implementation of Gao [9] (16185.5 ns).

Robustness Under Noise

Under realistic conditions with 2.0 pixels of image noise, the proposed method maintains a robust performance profile consistent with modern state-of-the-art solvers. Its success rate of 86.30% is virtually identical to that of Ding [3] and Wu [18], and comparable to Persson [17] (86.73%). While Banno [14] achieves the highest success rate of 89.49% due to its iterative nature, this robustness comes at a significant computational cost, being over four times slower than the proposed method. Conversely, Ke [10] suffers a severe performance degradation, with the success rate falling to 78.62%. Consequently, the proposed solver offers a balanced trade-off, delivering reliable accuracy with clean data and competitive robustness in noisy environments, at execution speeds suitable for real-time applications.

Table A1 Performance without Image Noise ($\sigma = 0.0$ px, Thresholds: 10^{-4} deg, 0.01%)

Solver	Time (ns)	Success (%)
Proposed	421.9	99.99
Wu [18]	263.3	99.77
Ding [3]	289.2	99.96
Persson [17]	349.5	99.65
Ke [10]	361.6	92.89
Kneip [28]	636.9	99.99
Nakano [13]	663.1	99.82
Banno [14]	1839.8	99.95
Gao (OpenCV) [9]	16185.5	95.16

Table A2 Performance with Image Noise ($\sigma = 2.0$ px, Thresholds: 5.0° , 10%)

Solver	Time (ns)	Success (%)
Proposed	431.3	86.30
Wu [18]	260.2	86.30
Ding [3]	291.9	86.30
Persson [17]	347.0	86.73
Kneip [28]	632.5	86.29
Nakano [13]	655.7	86.30
Banno [14]	1836.1	89.49
Ke [10]	355.8	78.62
Gao (OpenCV) [9]	15533.7	86.29

A.3 Efficiency in RANSAC Framework

In practical Structure-from-Motion (SfM) and SLAM pipelines, the P3P solver operates within a RANSAC loop, where the total computational burden is a function of both the solver’s execution time and the subsequent hypothesis verification cost.

A.3.1 Experimental Protocol

We integrated the evaluated solvers into a standard Adaptive RANSAC framework using a dataset of 1,000 3D-2D correspondences per trial, contaminated with 40% outliers and Gaussian pixel noise. An adaptive termination criterion was employed to ensure all solvers converged to the same confidence level ($\eta = 0.99$). As indicated by the results, all methods required a consistent number of iterations (approximately 76 to 77) to reach convergence, ensuring that the sampling strategy was uniform across all trials.

A.3.2 Result Analysis

Table A3 summarizes the performance metrics, including total execution time, average iterations, and the mean number of solution hypotheses per iteration.

Total Execution Efficiency

The results reveal a distinct stratification in computational performance. The newly introduced solver by Wu [18] achieves the lowest total runtime at 0.4683 ms. The proposed method follows closely as the second fastest, recording an average runtime of 0.4724 ms. This performance places our method

firmly in the top tier of efficiency, statistically tied with Persson [17] (0.4818 ms) and slightly outperforming Nakano [13] (0.4949 ms) and Ding [3] (0.4997 ms). In contrast, other established solvers exhibit noticeably higher computational costs. For instance, Kneip [28] and Ke [10] require 0.6579 ms and 0.8044 ms respectively, while the classical OpenCV implementation of Gao lags significantly at 1.3893 ms.

Impact of Solution Multiplicity on Verification

A critical factor differentiating the top-tier solvers is the number of hypotheses generated per iteration. In scenarios with a large number of feature points (e.g., $N = 1,000$), the hypothesis verification step—which involves reprojecting all points for each potential solution—becomes the dominant computational bottleneck. The proposed method, along with Wu and Gao, produces the minimum average number of solutions (1.45), effectively reducing the verification overhead to its theoretical lower bound. This contrasts sharply with solvers like Ke and Banno [14], which generate approximately 3.01 and 2.78 solutions per iteration, respectively. High solution multiplicity forces the RANSAC engine to perform significantly more verification checks, thereby inflating the total runtime despite the solver’s internal speed. By keeping the number of geometric hypotheses low, the proposed approach helps maintain RANSAC efficiency, which is beneficial when processing large point sets in real-time applications.

Table A3 RANSAC Performance Comparison (1000 Points, 40% Outliers)

Solver	Time (ms)	Iterations	Sols
Wu [18]	0.4683	77	1.45
Proposed	0.4724	77	1.45
Persson [17]	0.4818	77	1.48
Nakano [13]	0.4949	77	1.47
Ding [3]	0.4997	77	1.54
Kneip [28]	0.6579	77	2.05
Ke [10]	0.8044	76	3.01
Banno [14]	1.0016	76	2.78
Gao [9]	1.3893	77	1.45

A.4 Quantitative Evaluation on 7-Scenes Benchmark

To validate the practical applicability of the proposed theory in complex indoor environments, we integrated the solver into a standard RANSAC-based visual odometry frontend. The performance was evaluated against widely used methods including OpenCV (Gao), Ke, Kneip, Nakano, Banno, Ding, Persson, and Wu. The evaluation utilized four diverse sequences from the 7-Scenes dataset: `seq-01`, `seq-02`, `seq-03`, and `seq-04`, which feature motion blur, textureless regions, and varying illumination. To ensure statistical robustness, the pose estimation was performed across all frames in each sequence, with the RANSAC confidence level set to 99%.

Table A4 summarizes the quantitative comparisons, reporting the Mean Translation Error (Mean T), Mean Rotation Error (Mean R), and average execution time per frame.

Estimation Accuracy and Consistency

In terms of estimation accuracy, the proposed method demonstrates performance virtually indistinguishable from state-of-the-art solvers such as Ding [3], Wu [18], and Persson [17]. As evidenced by the data, both the translation and rotation errors are identical up to the significant digits across all tested sequences. This indicates that the proposed geometric constraints successfully recover the global optimum with the same fidelity as the most rigorous algebraic formulations. Notably, while Ke’s method [10] maintains competitive translation accuracy, it exhibits higher rotation errors, highlighting the stability of our approach in recovering full 6-DoF poses.

Real-Time Efficiency

Regarding computational efficiency, the proposed solver records an aggregated average execution time of approximately 0.099 ms per frame. While the highly optimized implementation by Wu [18] achieves a marginally lower runtime (0.089 ms), the proposed method remains in the same high-efficiency tier. Crucially, it represents a substantial improvement over standard libraries; it is approximately five times faster than the OpenCV implementation of Gao (0.506 ms) and over three times faster than Kneip (0.322 ms). The discrepancy between these results and the simulation outcomes

can be attributed to the fewer number of detected feature points in the real-world experiment, which significantly reduces the computational overhead of the verification stage.

Declarations

Ethics approval and consent to participate. Not applicable. This research involves purely theoretical and computational analysis; no human participants, animals, or biological materials were involved.

Consent for publication. Not applicable. This manuscript does not contain any individual person’s data in any form.

Availability of data and materials. The source code for this study is available at <https://github.com/lpl8848/P3P>. The numerical simulation results presented in this work can be reproduced using the algorithms and parameters described in the manuscript.

Competing interests. The author declares no competing interests.

Funding. No funding was received to assist with the preparation of this manuscript.

Authors’ contributions. P.L. conceived the research, developed the theoretical framework, performed the numerical experiments, and wrote the manuscript.

Acknowledgements. Not applicable.

References

- [1] Zhang, C.-x., Hu, Z.: Why is the danger cylinder dangerous in the P3P problem? *Acta Autom. Sin.* **32**(4), 504–511 (2006). In Chinese
- [2] Faugère, J.-C., Moroz, G., Rouillier, F., El Din, M.S.: Classification of the perspective-three-point problem, discriminant variety and real solving polynomial systems of inequalities. In: *Proc. Int. Symp. Symb. Algebr. Comput. (ISSAC)*, pp. 79–86 (2008). <https://doi.org/10.1145/1390768.1390782>

Table A4 Quantitative Comparison on 7-Scenes Benchmark. The table reports the Mean Translation Error (T [m]), Mean Rotation Error (R [°]), and average execution time (ms). The proposed method achieves accuracy identical to leading solvers while maintaining highly competitive execution speeds significantly faster than the standard OpenCV baseline.

Algorithm	seq-01		seq-02		seq-03		seq-04		Time (ms)
	T [m]	R [°]	T [m]	R [°]	T [m]	R [°]	T [m]	R [°]	
Gao (OpenCV) [9]	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.506
Ke [10]	0.0084	0.34	0.0069	0.31	0.0084	0.39	0.0075	0.36	0.143
Kneip [28]	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.322
Nakano [13]	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.100
Banno [14]	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.194
Ding [3]	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.090
Persson [17]	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.090
Wu [18]	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.089
Proposed	0.0084	0.30	0.0069	0.29	0.0084	0.33	0.0075	0.33	0.099

- [3] Ding, Y., Yang, J., Larsson, V., Olsson, C., Åström, K.: Revisiting the P3P problem. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 4872–4880 (2023). <https://doi.org/10.1109/cvpr52729.2023.00472>
- [4] Sun, X., Li, Z., Li, N.: Geometric stratification for singular configurations of the P3P problem via local dual space. arXiv preprint arXiv:2602.12525 (2026). arXiv:2602.12525
- [5] Fan, H., Kileel, J., Kimia, B.B.: Condition numbers in multiview geometry, instability in relative pose estimation, and RANSAC. IEEE Trans. Pattern Anal. Mach. Intell. **48**(3), 2855–2869 (2026) <https://doi.org/10.1109/tpami.2025.3629643>
- [6] Sturm, P.: A historical survey of geometric computer vision. In: Proc. Int. Conf. Comput. Anal. Images Patterns (CAIP). Lecture Notes in Computer Science, vol. 6978, pp. 259–268. Springer, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23672-3_1
- [7] Grunert, J.A.: Das pothenotische problem in erweiterter gestalt nebst über seine anwendungen in der geodäsie. Arch. Math. Phys., 238–248 (1841)
- [8] Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981) <https://doi.org/10.1145/358669.358692>
- [9] Gao, X.S., Hou, X.R., Tang, J.L., Cheng, H.F.: Complete solution classification for the perspective-three-point problem. IEEE Trans. Pattern Anal. Mach. Intell. **25**(8), 930–943 (2003) <https://doi.org/10.1109/tpami.2003.1217599>
- [10] Ke, T., Roumeliotis, S.I.: An efficient algebraic solution to the perspective-three-point problem. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 7225–7233. IEEE, Honolulu, HI, USA (2017). <https://doi.org/10.1109/cvpr.2017.491>
- [11] Yu, Q., Xu, G., Shi, J.: An efficient and reasonably simple solution to the perspective-three-point problem. In: Proc. Int. Conf. Pattern Recognit. (ICPR). Lecture Notes in Computer Science, pp. 46–59. Springer, Cham, Switzerland (2021). https://doi.org/10.1007/978-3-030-68793-9_4
- [12] Yu, Q., Jiang, R., Zhou, X., Wang, Y., Xu, G., Quan, W.: An efficient direct solution of the perspective-three-point problem. Comput. Vis. Image Underst., 104568 (2025) <https://doi.org/10.1016/j.cviu.2025.104568>
- [13] Nakano, G.: A simple direct solution to the perspective-three-point problem. In: Proc. Brit. Mach. Vis. Conf. (BMVC). BMVA

- Press, Cardiff, UK (2019). Paper No. 533
- [14] Banno, A.: A P3P problem solver representing all parameters as a linear combination. *Image Vis. Comput.* **70**, 55–62 (2018) <https://doi.org/10.1016/j.imavis.2018.01.001>
 - [15] Wang, P., Xu, G., Wang, Z., Cheng, Y.: An efficient solution to the perspective-three-point pose problem. *Comput. Vis. Image Underst.* **166**, 81–87 (2018) <https://doi.org/10.1016/j.cviu.2017.10.005>
 - [16] Li, S., Xu, C.: A stable direct solution of perspective-three-point problem. *Int. J. Pattern Recognit. Artif. Intell.* **25**(05), 627–642 (2011) <https://doi.org/10.1142/s0218001411008774>
 - [17] Persson, M., Nordberg, K.: Lambda Twist: An accurate fast robust perspective three point (P3P) solver. In: *Proc. Eur. Conf. Comput. Vis. (ECCV). Lecture Notes in Computer Science*, vol. 11205, pp. 318–332. Springer, Munich, Germany (2018). https://doi.org/10.1007/978-3-030-01225-0_20
 - [18] Wu, H., Bhayani, S., Heikkilä, J.: A conic transformation approach for solving the perspective-three-point problem. In: *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pp. 3237–3245. IEEE, Tucson, AZ, USA (2025). <https://doi.org/10.1109/wacv61041.2025.00320>
 - [19] Masselli, A., Zell, A.: A new geometric approach for faster solving the perspective-three-point problem. In: *Proc. 22nd Int. Conf. Pattern Recognit. (ICPR)*, pp. 2119–2124. IEEE, Stockholm, Sweden (2014). <https://doi.org/10.1109/icpr.2014.369>
 - [20] Tang, J., Chen, W., Wang, J.: A study on the P3P problem. In: *Proc. Int. Conf. Intell. Comput. (ICIC)*, pp. 422–429. Springer, Shanghai, China (2008). https://doi.org/10.1007/978-3-540-87442-3_53
 - [21] Rieck, M.Q.: Solving the three-point camera pose problem in the vicinity of the danger cylinder. In: *Proc. Int. Conf. Comput. Vis. Theory Appl. (VISAPP)*, vol. 2, pp. 335–340. SCITEPRESS, Rome, Italy (2012). <https://doi.org/10.5220/0003725403350340>
 - [22] Rieck, M.Q.: A fundamentally new view of the perspective three-point pose problem. *J. Math. Imaging Vis.* **48**(3), 499–516 (2014) <https://doi.org/10.1007/s10851-013-0425-8>
 - [23] Fan, H., Kileel, J., Kimia, B.B.: On the instability of relative pose estimation and RANSAC’s role. In: *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2022). <https://doi.org/10.1109/cvpr52688.2022.00873>
 - [24] Wang, B., Hu, H., Zhang, C.: Geometric interpretation of the multi-solution phenomenon in the P3P problem. *J. Math. Imaging Vis.* **62**, 1214–1226 (2020) <https://doi.org/10.1007/s10851-020-00982-5>
 - [25] Rieck, M.Q.: Geometric conditions for the existence or non-existence of a solution to the perspective 3-point problem. *J. Math. Imaging Vis.* **66**(1), 75–91 (2024) <https://doi.org/10.1007/s10851-023-01164-9>
 - [26] Vynnycky, M., Kanev, K.: Mathematical analysis of the multisolution phenomenon in the P3P problem. *J. Math. Imaging Vis.* **51**(2), 326–337 (2015) <https://doi.org/10.1007/s10851-014-0525-0>
 - [27] Markley, F.L.: Attitude determination using vector observations and the singular value decomposition. *J. Astronaut. Sci.* **36**, 245–258 (1988) <https://doi.org/10.2514/6.1988-4225>
 - [28] Kneip, L., Scaramuzza, D., Siegwart, R.: A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In: *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 2969–2976. IEEE, Colorado Springs, CO, USA (2011). <https://doi.org/10.1109/cvpr.2011.5995464>