China Instrument and Control Society
Instrumentation
INSTRUMENTATION, Vol. 12, No. 3, September 2025 https://doi.org/10.15878/j.instr.202500292

Article

An Efficient and Accurate Solution for the PnPL Problem

Ridma Basnayaka, Qida Yu*

School of Electronics Information Engineering, Nanjing University of Information Science & Technology, Nanjing 21004, China

* Corresponding author email: 003550@nuist.edu.cn

Abstract: Camera Pose Estimating from point and line correspondences is critical in various applications, including robotics, augmented reality, 3D reconstruction, and autonomous navigation. Existing methods, such as the Perspective-n-Point (PnP) and Perspective-n-Line (PnL) approaches, offer limited accuracy and robustness in environments with occlusions, noise, or sparse feature data. This paper presents a unified solution, Efficient and Accurate Pose Estimation from Point and Line Correspondences (EAPnPL), combining point-based and line-based constraints to improve pose estimation accuracy and computational efficiency, particularly in low-altitude UAV navigation and obstacle avoidance. The proposed method utilizes quaternion parameterization of the rotation matrix to overcome singularity issues and address challenges in traditional rotation matrix-based formulations. A hybrid optimization framework is developed to integrate both point and line constraints, providing a more robust and stable solution in complex scenarios. The method is evaluated using synthetic and real-world datasets, demonstrating significant improvements in performance over existing techniques. The results indicate that the EAPnPL method enhances accuracy and reduces computational complexity, making it suitable for real-time applications in autonomous UAV systems. This approach offers a promising solution to the limitations of existing camera pose estimation methods, with potential applications in low-altitude navigation, autonomous robotics, and 3D scene reconstruction.

Keywords: camera pose estimation; efficient and accurate pose estimation (eapnpl); UAV navigation; obstacle avoidance; point-and-line correspondences

Citation: Ridma Basnayaka, Qida Yu. "An Efficient and Accurate Solution for the PnPL Problem." Instrumentation 12, no. 3 (September 2025). https://doi.org/10.15878/j.instr.202500292

1 Introduction

The growing presence of autonomous systems, including self-driving cars, robotic assistants, and low-altitude unmanned aerial vehicles (UAVs), has increased the need for reliable and precise camera pose estimation, especially for autonomous navigation and obstacle avoidance in dynamic and constrained environments[1,2]. This fundamental computer vision problem, which involves determining the position and orientation of a camera relative to a known 3D scene, plays a crucial role in applications like autonomous UAV navigation, obstacle detection and avoidance, and 3D reconstruction, where precise pose estimation is essential for ensuring safe and efficient operation[3,4]. In robotics and autonomous UAVs, precise pose estimation enables systems to perceive their spatial relationship with the environment, facilitating tasks such as real-time navigation, obstacle detection, and avoidance, as well as 3D mapping in complex and dynamic environments[1,5]. Augmented reality (AR) relies heavily on accurate camera pose estimation to seamlessly integrate virtual objects into the user's view, enhancing the immersive experience[6]. Furthermore, 3D reconstruction, with applications ranging from virtual tours to medical imaging, depends critically on knowing the camera pose for each image to reconstruct the scene's geometry[7].

Traditionally, camera pose estimation has relied on establishing point correspondences between 3D world points and their 2D projections in the image plane. The

Perspective-n-Point (PnP) problem, a cornerstone of pose estimation, aims to solve for camera pose by minimizing the reprojection error between these 3D-2D correspondences. Variants of PnP, such as the Efficient PnP (EPnP), are widely used due to their computational efficiency and accuracy[8]. However, these point-based methods can be susceptible to noise and may falter when point correspondences are sparse or unreliable, particularly in low-texture or repetitive environments.

In contrast, the Perspective-n-Line (PnL) problem leverages 3D-2D line correspondences, which provide greater robustness for low-altitude UAVs navigating environments with sparse features, occlusions, or repetitive patterns[9,10]. As geometric features, lines often exhibit greater robustness than points, especially in scenes where establishing reliable point correspondences is challenging[11]. PnL methods have improved performance in such scenarios, offering greater accuracy and stability[12].

Recognizing the complementary strengths of PnP and PnL, recent research has focused on integrating point and line correspondences within a unified framework known as Perspective-n-Point-and-Line (PnPL), which is particularly useful for low-altitude UAV systems where robustness in dynamic environments is crucial[13]. This hybrid approach combines information from points and lines to achieve more robust and accurate pose estimation, particularly in challenging conditions. However, effectively integrating these geometric constraints into a single optimization framework presents significant computational hurdles[14].

1.1 Related Research Progress

The field of camera pose estimation has witnessed significant advancements. Driven by the increasing demand for robust and efficient solutions applicable to diverse real-world scenarios, traditional methods like PnP and PnL have been extensively studied. Recent advancements have focused on applying hybrid approaches for low-altitude UAV systems. These systems require robust and accurate pose estimation techniques to navigate challenging environments such as urban canyons, forested areas, or indoor spaces with sparse features and dynamic obstacles. Methods like PnPL are becoming increasingly crucial for autonomous UAV navigation, where line-based methods provide superior performance in environments with occlusions and low-texture areas, common in low-altitude applications[9,10]. Furthermore, integrating deep learning techniques in pose estimation has shown promise in improving the generalizability and robustness of pose estimation for UAVs operating in real-time, dynamic environments[13].

1.1.1 Advancements in PnP Methods

Recent efforts have focused on improving the robustness and efficiency of PnP methods. Kneip et al. (2011) introduced a novel parametrization for the Perspective-Three-Point (P3P) problem, enabling a direct computation of absolute camera position and orientation. This innovation significantly improves the efficiency and accuracy of pose estimation by eliminating the need for iterative methods, making it especially beneficial for real-time applications where computational speed is essential [15]. Terzakis and Lourakis (2020) proposed a globally optimal solution to the Perspective-n-Point (PnP) problem, addressing issues related to noisy or sparse correspondences. Their approach enhanced the efficiency of PnP methods by providing a globally optimal algorithm that guarantees accurate pose estimation, even in challenging environments. This solution reduced the sensitivity to outliers and overcame the computational complexity of global optimization techniques, making it particularly suitable for real-time applications requiring both speed and precision[16]. Building on these advancements, Yu et al. (2021) developed a consistently fast and accurate algorithm for estimating camera pose from point correspondences. Their method tackled outlier rejection and noise handling, thereby improving the robustness of pose estimation in noisy environments. This work made significant strides in ensuring reliable pose estimation under real-world conditions where point correspondences are imperfect[17]. Most recently, Wang et al. (2023) proposed a robust direct linear transformation (DLT) method that further enhances the accuracy and stability of PnP in the presence of noise. Their work demonstrated a marked improvement over traditional DLT methods, offering better performance in noisy scenarios and contributing to the overall advancement of PnP methods by increasing robustness without compromising computational efficiency[18].

1.1.2 Advancements in PnL Methods

Recent advancements in Perspective-n-Line (PnL) methods have significantly focused on improving the robustness and efficiency of camera pose estimation, particularly in challenging environments where traditional point-based methods struggle. Zhou et al. (2019) introduced a robust and efficient algorithm for the PnL problem that leverages algebraic distance to approximate reprojection distance. This approach improves the accuracy and robustness of pose estimation, particularly in the presence of noise and outliers. By avoiding the computational complexity of traditional reprojection-based methods, their solution enhances efficiency, scalability, and performance in real-world applications[9]. Wang et al. (2019) presented a fast, robust, and general method for camera pose estimation using line correspondences. The technique improves both speed and accuracy by taking advantage of the geometric properties of lines, which often provide more stable correspondences in environments with low texture or repetitive patterns. This innovation has proven particularly useful in real-time applications, where computational efficiency and robustness are essential.

Their approach is highly effective in handling noise and sparse correspondences, offering a reliable alternative to point-based methods in situations where point correspondences are challenging to obtain or unreliable [3]. Yu et al. (2022) further contributed to the PnL field by proposing an optimal solution to the PnL problem using quaternion parameterization. This approach enhances the accuracy and stability of pose estimation, mainly when dealing with noisy data. Yu's method presents an optimal framework that provides more precise camera pose estimations by utilizing quaternion-based parameterization to tackle the complexities inherent in PnL problems, contributing to improved performance in both theoretical and practical applications[12].

1.1.3 Hybrid PnPL Methods

The development of hybrid PnPL methods represents a significant trend in the field, aiming to leverage both point and line correspondences to enhance pose estimation. Wang et al. (2021) proposed a unified optimization approach that jointly optimizes point and line correspondences, achieving improved accuracy and robustness compared to using either feature type alone. This work demonstrates the potential of hybrid methods to outperform traditional approaches that rely on only one kind of correspondence, offering a more flexible solution for challenging real-world scenarios where either points or lines alone may be unreliable[2]. A significant focus in recent PnPL research has been on incorporating uncertainty into pose estimation to improve the robustness and reliability of methods under various conditions. Vakhitov et al. (2021) introduced an uncertainty-aware approach to camera pose estimation, combining point and line correspondences. This method enhances the precision of pose estimation and incorporates the inherent uncertainty of correspondences, making it more robust in noise and outliers. By considering uncertainty during the optimization process, their method provides more reliable pose estimates, particularly in noisy and incomplete data environments, marking a significant advancement in addressing one of the key challenges in PnPL methods[14]. This contribution highlights the growing importance of hybrid methods in improving the scalability and reliability of pose estimation, particularly for real-time robotic applications that demand both speed and precision[19]. Liu et al. (2022) developed a real-time PnPL algorithm that employs a coarse-to-fine strategy to balance computational efficiency with high accuracy. Their method ensures fast computation while maintaining the precision required for accurate pose estimation, further enhancing the applicability of PnPL in dynamic, real-time systems[5].

1.2 Research Trends and Gaps

1.2.1 Current Trends

Recent research in camera pose estimation has evolved along several key trends. A prominent development is the rise of hybrid PnPL approaches, which combine point and line correspondences to improve the robustness and accuracy of pose estimation, especially in challenging environments where point-based methods may fail. These methods leverage the complementary nature of points and lines, leading to more stable and reliable results[2],[5]. Another significant trend is the increasing application of deep learning techniques. Some methods focus on learning robust feature correspondences, while others aim to directly regress camera poses from images, bypassing traditional geometric methods[14]. A significant area of focus has been real-time performance, with researchers working on algorithms that balance high accuracy with low computational complexity to enable their deployment in resource-constrained environments[15,20]. Furthermore, there is growing interest in incorporating uncertainty quantification into pose estimation. This trend is crucial for ensuring the reliability and safety of systems in safety-critical applications, such as autonomous vehicles or robotics, where the accuracy and uncertainty of pose estimates directly impact system performance and decision-making[9].

1.2.2 Identified Gaps

Despite the significant advancements in camera pose estimation, several key challenges remain, particularly for low-altitude UAV navigation and obstacle avoidance. While offering improved robustness, hybrid methods like PnPL still struggle with extremely noisy data or sparse correspondences in highly dynamic or cluttered environments. UAVs operating in urban canyons or indoor environments often face occlusions, rapid motion, and real-time processing requirements, making it difficult for traditional methods to maintain accuracy and stability. Additionally, integrating real-time performance and uncertainty quantification for autonomous navigation remains an ongoing research gap, especially in safety-critical applications like UAVs, where pose accuracy directly influences operational safety[3],[21]. While promising, hybrid methods that combine point and line correspondences still struggle with extremely noisy data or sparse correspondences in real-world environments, particularly dynamic scenes[3]. This is further emphasised in the work by Camposeco et al. [22]. Generalizability across highly varied environments remains another challenge. Although increasingly popular, deep learning techniques often require vast amounts of labeled training data, limiting their effectiveness in diverse settings[4]. Recent works by Langerman et al. [23] propose domain adaptation techniques to address this issue. Incorporating uncertainty into hybrid PnP algorithms for improved performance in safety-critical tasks is an active area of research. While uncertainty-aware approaches have shown progress[14], standardised methods are still lacking. Li et al. [24] propose a Pose-Oriented Transformer with

Uncertainty-Guided Refinement that tackles this challenge in the context of human pose estimation. Achieving high accuracy and real-time performance remains challenging, especially for resource-constrained platforms. Li et al. [25] address this challenge in the context of mobile augmented reality systems by proposing an efficient bundle optimization algorithm for camera pose estimation. Yu et al. [24] also explore line-based camera pose estimation of point clouds of structured environments, demonstrating promising results for real-time applications.

This research addresses these gaps by developing a novel PnPL framework that efficiently integrates point and line correspondences into a robust optimization process. Quaternion parameterization represents camera rotation, offering advantages such as singularity-free representation and efficient optimization. To further enhance efficiency, we leverage advanced polynomial-solving techniques to handle the increased complexity arising from the integration of both point and line constraints. The proposed framework will be rigorously evaluated on synthetic and real-world datasets to demonstrate its accuracy, robustness, and computational efficiency in challenging scenarios, including noisy data, sparse correspondences, and dynamic environments.

The figure demonstrates the Perspective-n-Point-and-Line (PnPL) problem, which seeks to determine the camera's rotation **R** and translation **t** between the world and camera frames by using both 3D-2D point correspondences $\{\boldsymbol{P_i} \leftrightarrow \mathbf{p_i}\}$ and line correspondences $\{\boldsymbol{L_i} \leftrightarrow \boldsymbol{l_i}\}$, given the camera's intrinsic matrix $\boldsymbol{A}$. It depicts a camera projecting a 3D scene, where yellow dots inside a rectangular prism represent 3D points, and blue lines within and extending from the prism represent 3D lines, with dashed blue lines showing their projections onto the image plane. Unlike methods relying solely on points, the line correspondences here do not need precise endpoint matches, making the approach more robust to occlusions or environmental challenges in real-world settings. By combining these geometric constraints, the PnPL framework reliably estimates $\boldsymbol{R}$ and $\boldsymbol{t}$.

2 Methodology

2.1 Problem Statement

The EAPnPL (Efficient and Accurate Pose Estimation from Point and Line Correspondences) method is particularly suited for low-altitude UAV navigation and obstacle avoidance, where accurate pose estimation is crucial for maintaining safe flight paths in dynamic environments. The goal of the PnPL problem is to estimate the camera's pose, represented by the rotation matrix $\boldsymbol{R} \in SO(3)$ and translation vector $\boldsymbol{t} \in \mathbb{R}^3$, given correspondences between 3D points and lines in the world coordinate system and their 2D projections on the normalized image plane. UAVs operating in cluttered spaces or areas with a sparse texture often rely on line-based features for more stable and robust pose estimation. Integrating point and line correspondences in the EAPnPL framework provides a more reliable estimation, especially in environments where traditional point-based methods struggle due to occlusions, noise, or limited feature availability. This approach utilizes quaternion parameterization to solve the rotation matrix's singularity issues, which are particularly challenging in low-altitude UAV scenarios. These issues often arise when real-time adjustments are required for path planning and collision avoidance in environments with rapidly changing obstacles and dynamic features. For a set of $\boldsymbol{n_p}$ 3D points $\{\boldsymbol{p}_i\}_{i=1}^{n_p}$ and their corresponding 2D projections $\{\boldsymbol{b}_i\}_{i=1}^{n_p}$, the projection relationship can be described by:

$$\lambda_i \boldsymbol{b}_i = \begin{bmatrix} \boldsymbol{R} | \boldsymbol{t} \end{bmatrix} \begin{bmatrix} \boldsymbol{p}_i \\ 1 \end{bmatrix}, i \in \{1, 2, \ldots, n_p\} \quad (1)$$

where λ_i is the depth of the i-th point. Eliminating λ_i Via pairwise division results in the linearized constraint:

$$\begin{bmatrix} -1 & 0 & u_i \\ 0 & -1 & v_i \end{bmatrix} (\boldsymbol{R}\boldsymbol{p}_i + \boldsymbol{t}) = 0_{2\times 1},\ i \in \{1, 2, \ldots, n_p\} \quad (2)$$

For $\boldsymbol{n_i}$ 3D lines $\{\boldsymbol{L_i}\}_{i=1}^{n_l}$, represented by their direction vectors d_i and random points $\boldsymbol{p_i}$ on the lines and their corresponding 2D projections $\{\boldsymbol{l_i}\}_{i=1}^{n_l}$, the geometric constraints are defined as:

$$\boldsymbol{n}_i^\top \boldsymbol{R}\boldsymbol{d}_i = 0, i \in \{1, 2, \ldots, n_l\} \quad (3)$$

$$\boldsymbol{n}_i^\top (\boldsymbol{R}\boldsymbol{p}_i + \boldsymbol{t} = 0, i \in \{1, 2, \ldots, n_l\} \quad (4)$$

where $\boldsymbol{n}_i$ is the normal vector to the 2D projection plane of the line; the objective of the PnPL problem is to estimate **R** and **t** by simultaneously satisfying both point Eq. (2), line Eq. (3), and constraints Eq. (4), minimizing the projection error under a unified framework. This formulation integrates point and line correspondences, ensuring robust pose estimation in challenging scenarios.

3 Propose Method

3.1 Least Square Minimization Problem

To solve the PnPL problem, we aim to estimate the camera's pose, represented by the rotation matrix $\boldsymbol{R}$ and translation vector $\boldsymbol{t}$, by combining the geometric constraints from 3D points, 3D lines, and their 2D projections. The mathematical formulation is developed using a Kronecker product representation, which enables a compact and unified system of equations for efficient computation. For the point-based constraints, the projection relationship in Eq. (1) (3) and (2) can be rewritten in Kronecker product form as follows:

$$\left(\boldsymbol{p}_i^T \otimes \boldsymbol{b}_i^T\right) vec(\boldsymbol{R}) + \boldsymbol{b}_i^T \boldsymbol{t} = 0, \mathrm{i} \in \{1, 2, \ldots, n_p\} \quad (5)$$

$$\left(\boldsymbol{b}_i^T \otimes 1\right) vec(\boldsymbol{R}) + \boldsymbol{b}_i^T \boldsymbol{t} = 0, i \in \{1, 2, \ldots, n_p\} \quad (6)$$

For the line-based constraints, the geometric relationships in Eq. (3) and (4) become:

$$\left(\boldsymbol{d}_i^T \otimes \boldsymbol{n}_i^T\right) vec(\boldsymbol{R}) + 0^T \boldsymbol{t} = 0,\ i \in \{1, 2, \ldots, n_l\} \quad (7)$$

$$\left(\boldsymbol{p}_i^T \otimes \boldsymbol{n}_i^T\right) vec(\boldsymbol{R}) + \boldsymbol{n}_i^T \boldsymbol{t} = 0,\ i \in \{1, 2, \ldots, n_l\} \quad (8)$$

The Kronecker product formulations compactly represent individual point and line constraints, making it easier to integrate them into a unified system. For example, the point constraints from Eq. (5) and (6) can be combined into a single system.

$$\begin{bmatrix} \left(\boldsymbol{p}_i \otimes \boldsymbol{b}_i\right)^T \\ \left(\boldsymbol{b}_i \otimes 1\right)^T \end{bmatrix} vec(\boldsymbol{R}) = \begin{bmatrix} -0^T \\ -\boldsymbol{b}_i^T \end{bmatrix} \boldsymbol{t},\ i \in \{1, 2, \ldots, n_p\} \quad (9)$$

Similarly, the line constraints from Eq. (7) and (8) are combined as:

$$\begin{bmatrix} \left(\boldsymbol{d}_i \otimes \boldsymbol{n}_i\right)^T \\ \left(\boldsymbol{p}_i \otimes \boldsymbol{n}_i\right)^T \end{bmatrix} vec(\boldsymbol{R}) = \begin{bmatrix} -0^T \\ -\boldsymbol{n}_i^T \end{bmatrix} \boldsymbol{t},\ i \in \{1, 2, \ldots, n_l\} \quad (10)$$

To handle points and lines simultaneously, we integrate the point constraints Eq. (9) and line constraints Eq. (10) into a unified system. By stacking these equations, the complete system for all $n_p$points and n_l lines can be expressed as:

$$\underbrace{\begin{bmatrix} \left(\boldsymbol{p}_i \otimes \boldsymbol{b}_i\right)^T & \left(\boldsymbol{b}_i \otimes 1\right)^T \\ \left(\boldsymbol{d}_i \otimes \boldsymbol{n}_i\right)^T & \left(\boldsymbol{p}_i \otimes \boldsymbol{n}_i\right)^T \end{bmatrix}}_{\boldsymbol{D}_i} vec(\boldsymbol{R}) = \underbrace{\begin{bmatrix} -0^T \\ -\boldsymbol{b}_i^T \\ -0^T \\ -\boldsymbol{n}_i^T \end{bmatrix}}_{\boldsymbol{N}_i} \boldsymbol{t},\quad i \in \{1, 2, \ldots, n_p + n_l\} \quad (11)$$

where $D_i \in \mathbb{R}^{\left(2n_p+2n_l\right)\times 9}$, and $N_i \in \mathbb{R}^{\left(2n_p+2n_l\right)\times 3}$ for each 3D point and line and their corresponding 2D projections. Stacking the equations for all n_p points and n_l lines results can be written as:

$$\underbrace{\begin{bmatrix} \boldsymbol{N}_1 \\ \boldsymbol{N}_2 \\ \vdots \\ \boldsymbol{N}_{n_p+n_l} \end{bmatrix}}_{\boldsymbol{N}} \boldsymbol{t} = \underbrace{\begin{bmatrix} \boldsymbol{D}_1 \\ \boldsymbol{D}_2 \\ \vdots \\ \boldsymbol{D}_{n_p+n_l} \end{bmatrix}}_{\boldsymbol{D}} vec(\boldsymbol{R}) \square \boldsymbol{N}\boldsymbol{t} = \boldsymbol{D}\, vec(\boldsymbol{R}) \quad (12)$$

$\boldsymbol{N}$ and $\boldsymbol{D}$ are the global coefficient matrices formed by stacking the individual $\boldsymbol{N}_l$ and $\boldsymbol{D}_l$ matrices for all correspondences. The optimal solution for $\boldsymbol{t}$, given $vec(\boldsymbol{R})$, is obtained by solving the above system and is expressed as:

$$\boldsymbol{t} = \left(\boldsymbol{N}^T\boldsymbol{N}\right)^{-1}\boldsymbol{N}^T\boldsymbol{D}\, vec(\boldsymbol{R}) \quad (13)$$

Figure 1 illustrates the definitions of the PnP and PnL problems. Both aim to recover the rotation (**R**) and translation (**t**) between the world and camera frames, using the intrinsic matrix **A** and corresponding points or lines. It is important to note that, unlike point correspondences, precise endpoint matches are not essential for line correspondences, as shown in this figure. This is commonly observed in practical applications, where occlusions or environmental constraints may affect the accuracy of the correspondence. Substituting this expression for $\boldsymbol{t}$ back into Eq. (12) lead to:

$$\begin{aligned} &\boldsymbol{N}\left(\boldsymbol{N}^T\boldsymbol{N}\right)^{-1}\boldsymbol{N}^T\boldsymbol{D}vec(\boldsymbol{R}) - \boldsymbol{D}vec(\boldsymbol{R}) = 0 \\ \Leftrightarrow &\underbrace{\left(\boldsymbol{N}\left(\boldsymbol{N}^T\boldsymbol{N}\right)^{-1}\boldsymbol{N}^T - \boldsymbol{I}_{2n_p+2n_l}\right)\boldsymbol{D}}_{\boldsymbol{A}} vec(\boldsymbol{R}) = 0 \\ \Leftrightarrow &\boldsymbol{A}\, vec(\boldsymbol{R}) = 0 \end{aligned} \quad (14)$$

This equation expresses the fundamental relationship between the geometry of the world frame and the camera frame. $\boldsymbol{A}$ is a $\left(2n_p+2n_l\right)\times 9$ coefficient matrix, $\boldsymbol{I}_3$ is the identity matrix of size 3×3, and n_p *and* n_l are the number of points and line correspondences, respectively. To handle noise in the measurements and to solve the system in a practical context, we recast the problem as a constrained least-squares optimization problem, which reads:

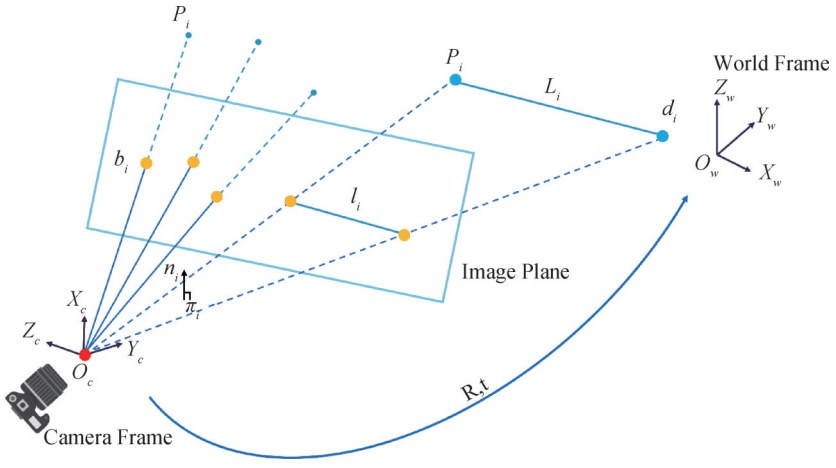


Fig.1 Geometry of PnP problem and PnL problem

$$\arg\min_{R} \left\| \boldsymbol{A}\, vec(\boldsymbol{R}) \right\|^2 \tag{15}$$

$$\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}_3 \tag{16}$$

$$\det(\boldsymbol{R}) = 1 \tag{17}$$

The objective minimizes the error in the rotation matrix $\boldsymbol{R}$, while the constraints enforce that $\boldsymbol{R}$ must remain a valid rotation matrix, i. e., it must be orthonormal with a determinant of 1, where $det(.)$ denotes the determinant of a square matrix. These constraints ensure that $\boldsymbol{R}$ represents a proper rotation without reflection. This approach enables robust estimation of the camera rotation by incorporating both point and line correspondences in the PnPL problem.

3.2 Unconstrained Problem

The goal of the PnPL problem is to estimate the camera's rotation matrix $\boldsymbol{R}$, while addressing the constraints imposed by the orthogonality Eq. (16) and the determinant Eq. (17), which ensure $\boldsymbol{R}$ remains a valid rotation matrix. These constraints can introduce challenges such as singularities and non-uniqueness, particularly at critical angles like 180°. To overcome these issues, the proposed method adapts the ASPnP (Algebraic Solution for Perspective-n-Point) approach by employing a unit-quaternion parameterization for the rotation matrix. This parameterization eliminates the orthogonality and determinant constraints, providing a robust and singularity-free solution for the PnPL problem. The rotation matrix **R** is expressed in a unified form using a unit-quaternion parameterization, enabling seamless integration of point and line constraints.

To remove the constraints on the rotation matrix Eq. (16) and (17), we use a unit-quaternion parameterization that eliminates these restrictions while preserving the validity of $\boldsymbol{R}$. This approach builds upon the ASPnP (Algebraic Solution for Perspective-n-Point) method and extends it to the PnPL problem, ensuring stability and robustness in the estimation process. The unit-quaternion parameterization is expressed as:

$$\boldsymbol{R} = \begin{bmatrix} s_1^2+s_2^2-s_3^2-S_4^2 & 2s_2s_3-2s_1s_4 & 2s_2s_4+2s_1s_3 \\ 2s_2s_3+2s_1s_4 & s_1^2-s_2^2+s_3^2-s_4^2 & 2s_3s_4-2s_1s_2 \\ 2s_2s_4-2s_1s_3 & 2s_3s_4+2s_1s_2 & s_1^2-s_2^2-s_3^2+s_4^2 \end{bmatrix} \tag{18}$$

where $s_1, s_2, s_3,\ s_4$ are the four quaternion parameters that satisfy the normalization condition:

$$s_1^2+s_2^2+s_3^2+S_4^2 = 1 \tag{19}$$

This parameterization ensures that the rotation matrix $\boldsymbol{R}$ is singularity-free, even at critical angles such as 180°. Depending on the specific values of the quaternion parameters, $\boldsymbol{R}$ simplifies into different special cases. These cases reduce the number of parameters, enabling the problem to be split into multiple subproblems for efficient computation. The main cases are described below:

Case 1: General Case (All Parameters Non-Zero) When $s_1, s_2, s_3,\ s_4$ are all non-zero, the rotation matrix $\boldsymbol{R}$ can be expressed as:

$$\boldsymbol{R} = \frac{1}{S}\begin{bmatrix} 1+s_1^2-s_2^2-s_3^2-s_4^2 & 2s_2s_3-2s_1s_4 & 2s_2s_4+2s_1s_3 \\ 2s_2s_3+2s_1s_4 & 1-s_1^2+s_2^2-s_3^2-s_4^2 & 2s_3s_4-2s_1s_2 \\ 2s_2s_4-2s_1s_3 & 2s_3s_4+2s_1s_2 & 1-s_1^2-s_2^2+s_3^2-s_4^2 \end{bmatrix} \tag{20}$$

where $Q = 1+s_1^2+s_2^2+s_3^2+s_4^2$. This general form can handle arbitrary rotations and forms the foundation for solving the unified PnPL problem.

Case 2: When $s_1 = 0, s_2, s_3, s_4 \neq 0$ and the remaining parameters are non-zero, the rotation matrix simplifies to:

$$\boldsymbol{R} = \frac{1}{1+s_3^2+s_4^2}\begin{bmatrix} 1-s_3^2-s_4^2 & 2s_3 & 2s_4 \\ 2s_3 & -1+s_3^2-s_4^2 & 2s_3s_4 \\ 2s_4 & 2s_3s_4 & -1-s_3^2+s_4^2 \end{bmatrix} \tag{21}$$

This case reduces the number of parameters, making the optimization process more efficient for specific rotations.

Case 3: When $s_1 = 0, s_2 = 0, s_3 \neq 0,\ s_4 \neq 0$, the rotation matrix reduces further to:

$$\boldsymbol{R} = \frac{1}{1+s_3^2+s_4^2}\begin{bmatrix} 1-s_3^2-s_4^2 & 0 & 0 \\ 0 & 1+s_3^2-s_4^2 & 2s_3s_4 \\ 0 & 2s_3s_4 & 1+s_3^2-s_4^2 \end{bmatrix} \tag{22}$$

In this case, the off-diagonal terms involving s_1 and s_2 vanish, leaving a more straightforward structure that depends only on s_3 and s_4.

Case 4: When $s_1 = 0, s_2 = 0, s_3 = 0,\ s_4 \neq 0$, the rotation matrix simplifies ultimately to:

$$\boldsymbol{R} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{23}$$

This case represents a fixed rotation matrix, as the contributions of s_1, s_2 and s_3 , are entirely eliminated, leaving only the term s_4^2.

The PnPL problem unifies the point-based (PnP) and line-based (PnL) constraints into a single quadratic cost function by leveraging the quaternion parameterization of the rotation matrix $\boldsymbol{R}$. Using quaternion parameters $s_1,\ s_2, s_3, s_4$, the unified cost function is expressed as:

$$\min_{s_1,s_2,s_3,s_4} J = \left\| \boldsymbol{B}\boldsymbol{v} \right\|^2 = \boldsymbol{v}^{\top}\underbrace{\left(\boldsymbol{B}^{\top}\boldsymbol{B}\right)}_{C}\boldsymbol{v} = \boldsymbol{v}^{\top}\boldsymbol{C}\boldsymbol{v} \tag{24}$$

where $\boldsymbol{B}$ and $\boldsymbol{C}$ are coefficient matrices derived from the combined constraints of points and lines, and $\boldsymbol{v}$ is the vector representation of the monomials of the quaternion parameters. This unified formulation ensures that the rotational constraints imposed by the 3D-2D Point and the Line correspondences are simultaneously satisfied.

To minimize *J*, the first-order optimality conditions are computed as:

$$\frac{\partial J}{\partial s_1}=0,\quad \frac{\partial J}{\partial s_2}=0,\quad \frac{\partial J}{\partial s_3}=0,\quad \frac{\partial J}{\partial s_4}=0 \tag{25}$$

The PnPL problem combines point-based (PnP) and line-based (PnL) constraints into a unified system, expressed through polynomial equations involving quaternion parameter s_1, s_2, s_3, s_4. The quaternion parameterization ensures that the rotation matrix R remains valid while avoiding singularities. Eq. (25) defines a system of four polynomial equations derived from the unified constraints. These equations are efficiently solved using Grobner Basis (GB) and action-matrix methods, which are well-established in solving polynomial systems in computer vision. The automatic generator for GB solvers constructs the elimination template and action matrix iteratively, enabling the solution of these equations via Eigen-factorization.

For the PnPL problem, solving Eq. (25) In general, the case yields, at most, 81 solutions, as the system has four variables and degree-three equations. The elimination template and action matrix for this case are of sizes 160×200 and 81×81, respectively. The solution space simplifies for exceptional cases where some quaternion parameters are zero. Case 2 yields at most 27 solutions, Case 3 produces 9 solutions, and Case 4 results in a single solution. These correspond to progressively reduced degrees of freedom in the quaternion parameterization. For each solution, the quaternion parameters s_1, s_2, s_3, s_4 are substituted back into Eqs. (20) and (13) to recover the pose. Across all cases, a maximum of 118 solutions is possible.

The globally optimal solution is determined by evaluating all solutions and selecting the one that minimizes the error in the unified cost function J. This approach ensures robust and efficient camera pose estimation, seamlessly integrating point and line constraints into a single optimization framework.

3.3 Efficient Computation

In the above sections, we reformulate the PnPL problem into four unconstrained optimization subproblems by integrating a unified quaternion parameterization. For each subproblem, we construct a rotational cost function that involves a known coefficient matrix and a set of monomials in the unknown quaternion parameters. These coefficient matrices, derived from $\left\{\left(\boldsymbol{L}_i, \boldsymbol{l}_i, \boldsymbol{p}_i, \boldsymbol{b}_i\right)\right\}_{i=l}^{n_p+n_l}$, can be computed efficiently. For simplicity, the coefficient matrices in Cases 2 – 4 are denoted $\boldsymbol{C}_2$, $\boldsymbol{C}_3, \boldsymbol{C}_4$, analogous to $\boldsymbol{C}_1$ in Case 1. Usually, these matrices would need to be calculated independently for each subproblem, which is computationally expensive.

To enhance efficiency, we propose a strategy where $\boldsymbol{C}_2$, $\boldsymbol{C}_3$, and $\boldsymbol{C}_4$ are directly derived as blocks of $\boldsymbol{C}_1$. For instance, in Case 2, the corresponding vector $\boldsymbol{v}_2$, analogous to $\boldsymbol{v}_1$ Case 1, is $\left[1, s_3, s_4, s_3^2, s_3 s_4, s_4^2\right]^{\top}$. Compared to $\boldsymbol{v}_1, \boldsymbol{v}_2$ excludes monomials involving s_2, allowing it to be rewritten as $\left[1,\ 0\cdot s_2,\ s_3, s_4, 0\cdot s_2^2, 0\cdot s_2 s_4, s_3^2, s_3 s_4, s_4^2\right]^{\top}$, which retains the same structure as v_1. As a result, $\boldsymbol{C}_2$ becomes a block of $\boldsymbol{C}_1$, and the same applies to $\boldsymbol{C}_3$ and $\boldsymbol{C}_4$. This relationship is expressed as:

$$\boldsymbol{C}_1=\begin{bmatrix}\cdots & \cdots\\ \cdots & \boldsymbol{C}_2\end{bmatrix}_{10\times10} \tag{26}$$

Instead of calculating $\boldsymbol{C}_1$, $\boldsymbol{C}_2$, $\boldsymbol{C}_3$, and $\boldsymbol{C}_4$ independently, we compute $\boldsymbol{C}_1$ once from $\left\{\left(\boldsymbol{L}_i, \boldsymbol{l}_i, \boldsymbol{p}_i, \boldsymbol{b}_i\right)\right\}_{i=l}^{n_p+n_l}$ and directly extract $\boldsymbol{C}_2$, $\boldsymbol{C}_3$, and $\boldsymbol{C}_4$ from it. This approach significantly reduces computational complexity and improves the overall efficiency of the method, marking a key contribution of this work to solving the PnPL problem.

4 Experimental Results

This section evaluates the proposed method's performance through synthetic and real-world experiments, focusing on various configurations of 3D points and 3D line correspondences. The primary objective is to assess the approach's accuracy and robustness under different noise levels, numbers of correspondences, and 3D scene configurations, including both planar and nonplanar setups.

4.1 Synthetic Experiments

We evaluate our approach against state-of-the-art in two scenarios: the point-and-line and line-only cases. In both situations, we consider nonplanar and planar configurations and report the rotation and translation errors for increasing amounts of 2D noise and varying numbers of correspondences. We also investigate the influence of the distance between the line segments in the 3D model and their projected 2D endpoints.

For the synthetic experiments, we use a calibrated virtual camera with a resolution of 640 × 480 pixels and a focal length of 800 pixels. The 3D points and line segments are randomly generated within a box with coordinates $[-2, 2]$,$[-2, 2]$ and $[4, 8]$ in the camera frame. The total number of points n_p and line segments n_l varies depending on the configuration. The rotation matrix is randomly generated, and the translation vector is set to the mean vector of the 3D points.

The generated 3D lines are perturbed by Gaussian noise, and additional noise is added to the 2D projections. The noise standard deviation is set to 1 pixel for all simulations. Two additional points are generated on the same line for each line segment by randomly shifting the endpoints. These points are projected onto the image plane and corrupted with noise.

The rotation errors are computed using the following formula:

$$\mathrm{e}_{\mathrm{rot}}=max_1^k \mathrm{acos}\left(dot\left(\boldsymbol{R}_t^k, \boldsymbol{R}^k\right)\right)\times180/\pi \tag{27}$$

EAPnPL demonstrated superior performance in low-

altitude navigation scenarios in our synthetic experiments, particularly in environments with occlusions, sparse features, and noisy data. The method excelled in obstacle detection and avoidance simulations, where real-time pose estimation was crucial for maintaining safe paths. For example, in scenarios with dense, dynamic obstacles and low-texture regions, EAPnPL provided highly accurate pose estimates, which helped avoid potential collisions by re-planning flight paths on the fly. where $\boldsymbol{R}_k$ represents the k-th column of the rotation matrix $\boldsymbol{R}_t$, and $\boldsymbol{R}$ is the ground-truth rotation matrix. The translation error evaluation is represented as:

$$E_{trans}(\%) = \| \boldsymbol{t}_{true} - \boldsymbol{t} \| / \| \boldsymbol{t}_t \| \times 100 \quad (28)$$

where $\boldsymbol{t}_t$ is the estimated translation vector and $\boldsymbol{t}$ denote the ground-truth translation vector. The results are presented as mean and median errors for rotation and translation. We also analyze the impact of line segment shift, simulating real-world distortions caused by camera calibration errors or scene geometry variations.

The algorithms tested include the PnP methods EPnP_GN[8], OPnP [26], and EOPnP[27], the PnL methods Ansar[28], AlgLS[29], RPnL[30], and ASPnL[11], and the PnPL methods DLT[31], EPnPL[19], OPnPL[19], and EAPnPL. We conducted 500 independent simulations for each configuration, varying the amount of noise and the number of correspondences to assess the method's robustness under different conditions. We used an AMD Ryzen 9 7900X 12-core Processor and 32 GB of RAM.

4.1.1 Pose Estimation Using Only Lines

We evaluated pose estimation by focusing exclusively on line based approaches, comparing PnL methods with PnPL methods fed only line correspondences ($n_p = 0$). When evaluating pose estimation using only line-based correspondences, EAPnPL demonstrated significant advantages in low-altitude UAV navigation, where accurate line-based features are more stable than point correspondences, especially in environments with low visual texture or occlusions. The method effectively navigated challenging environments, maintaining high pose accuracy even when only limited line correspondences were available. Two configurations were tested, non-planar and planar, as shown in Figure 2 To assess performance comprehensively, we analyzed accuracy under fixed constraints and varying noise levels: planar (PnP ($n_p = 6$).

(a) Fixed Line Constraints

In the non-planar setup (Figure 2-(a): PnL ($n_l = 10$), PnPL($n_p = 0$, $n_l = 10$)) and planar setup (Figure 2-(b): PnL ($n_l = 10$), PnPL($n_p = 0$, $n_l = 10$)), we used 10 line correspondences ($n_l = 10$), for both PnL and PnPL methods. OPnPL and EAPnPL consistently outperform all PnL methods tested, with OPnPL leading slightly. EAPnPL demonstrates robustness comparable to OPnPL and surpasses EPnPL, especially in non-planar conditions. In the planar case, PnL methods exhibit significant instability. For example, Mirzaei[29] fails entirely, and produces errors too large to plot, while OPnPL and EAPnPL maintain superior accuracy.

(b) Noise Impact Assessment

We further examined accuracy under a noise level of 2 pixels, as shown in Figure 2-(c) for non-planar and Figure 2-(d) for planar configurations (both with PnL ($n_l = 10$), PnPL($n_p = 0$, $n_l = 10$)). OPnPL excels across both setups, particularly in minimizing errors, while EAPnPL follows closely, outperforming EPnPL and all PnL methods. The planar configuration highlights PnL's weaknesses, whereas EAPnPL and OPnPL remain reliable, with EAPnPL showing notable stability despite the noise. These results underscore the advantage of PnPL methods over traditional PnL approaches when relying solely on line data.

4.1.2. Pose Estimation with Combined Points & Lines

We analyzed pose estimation by integrating point and line correspondences in non-planar and planar configurations. In the combined points and lines setup, EAPnPL showed remarkable robustness in dynamic environments where rapid changes in obstacles were present. This hybrid approach significantly improved navigation accuracy and obstacle avoidance capabilities, ensuring the UAV could quickly and effectively adjust its path based on real-time pose estimates. These results underscore the utility of EAPnPL for real-time low-altitude navigation in environments with rapidly changing or sparse features. The accuracy of PnP, PnL, and PnPL methods was assessed concerning image noise and increasing numbers of correspondences Figure 3. To ensure an equitable comparison across these methods, we explored two distinct scenarios:

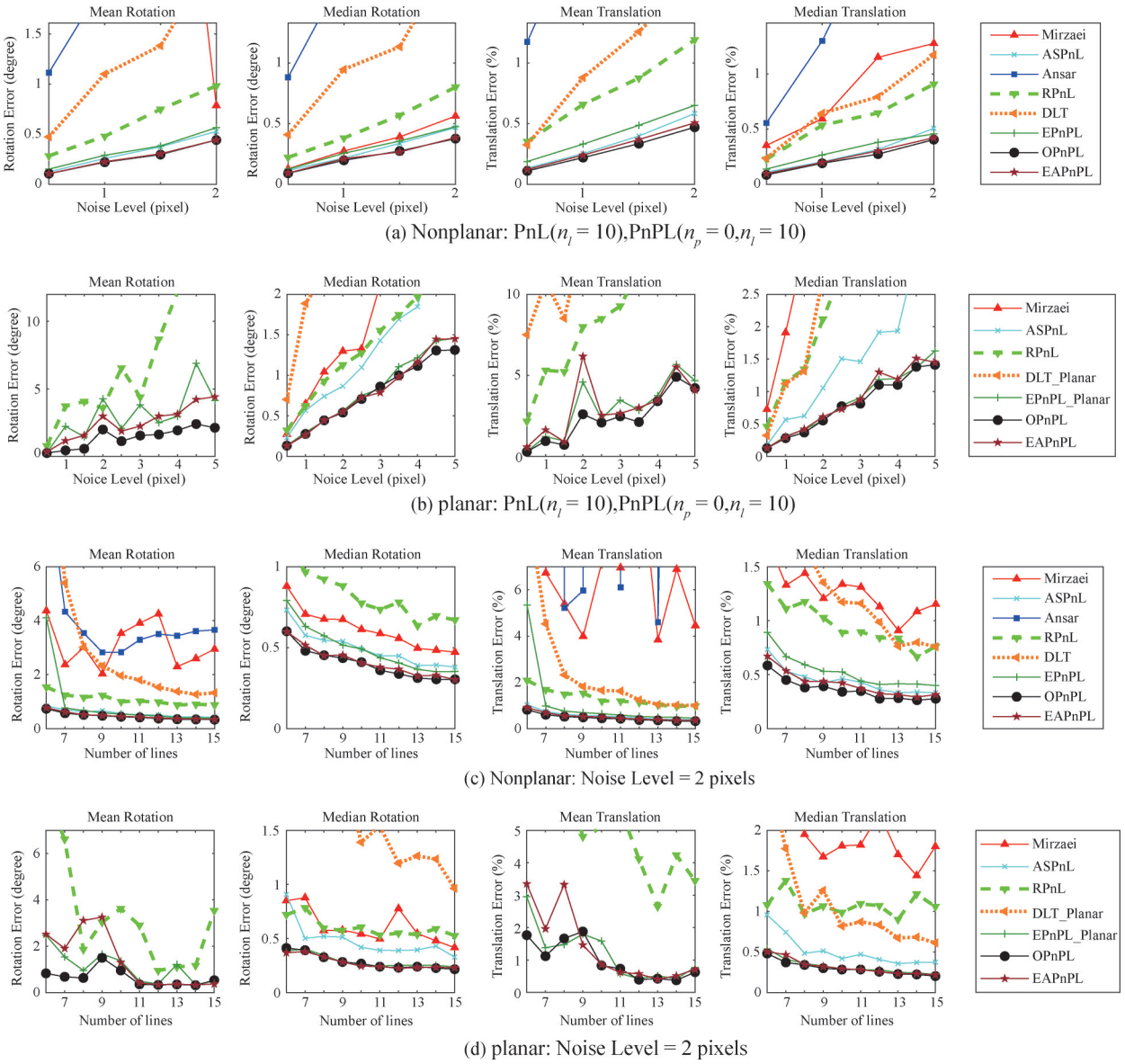
(a) Fixed Minimum Constraints

In evaluating accuracy under noise, we configured

PnP methods with 6-point correspondences ($n_p = 6$), the minimum required for robust estimation, PnL methods 10-line correspondences ($n_l = 10$), reflecting their typical minimum and PnPL methods, a combination of 6 points and 10 lines ($n_p = 6, n_l = 10$). These settings are shown in Figure 3-(a) for non-planar (PnP ($n_p = 6$), PnL ($n_l = 10$), PnPL($n_p = 6$, $n_l = 10$)) and Figure 3-(b) for planar (same settings). EAPnPL and OPnPL outperform PnP-only methods, benefiting from the added line constraints. OPnPL closely approaches the accuracy of an idealized PnP baseline, effectively utilizing line data, while EAPnPL follows closely. Both methods significantly outstrip DLT and other PnL approaches when correspondence counts increase, as tested in Figure 3-(c) for non-planar (PnP ($n_p = 6{:}15$) , PnL ($n_l = 6{:}15$) PnPL ($n_p + n_l = 12{:}2{:}30$) and Figure 3-(d) for planar (same ranges) with a noise level of 2 pixels, EAPnPL and OPnPL maintain superior performance.

(b) Variable Constraints with Noise

We further tested accuracy under a noise level of 2 pixels while varying correspondence counts: PnP ($n_p = 6$),



(a) Nonplanar: PnL(n_l = 10),PnPL(n_p = 0,n_l = 10)

(b) planar: PnL(n_l = 10),PnPL(n_p = 0,n_l = 10)

(c) Nonplanar: Noise Level = 2 pixels

(d) planar: Noise Level = 2 pixels

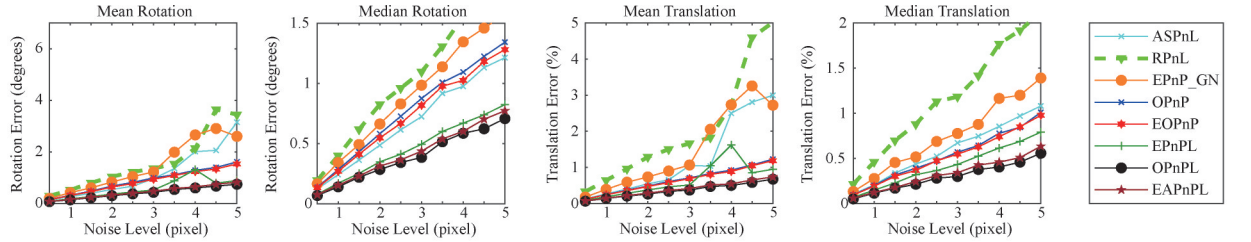
Fig.2 Experimental results concerning line correspondences only

PnL (n_l = 6:15), and PnPL(n_p + n_l = 12:2:30). These results are illustrated in Figure 3-(c) for non-planar and Figure 3-(d) for planar configurations. PnP methods gain an advantage from precise, noise-perturbed points, whereas PnL and PnPL methods handle less spatially defined line inputs. EAPnPL and OPnPL excel in this scenario, with OPnPL nearly matching its point-only counterpart's precision and EAPnPL remaining competitive, though slightly less accurate than OPnPL. Both outperform DLT and PnL methods by a wide margin. In the planar case (Figure 3- (d)), EAPnPL and OPnPL stand out, particularly in minimizing rotation errors, demonstrating their robustness across varied conditions.

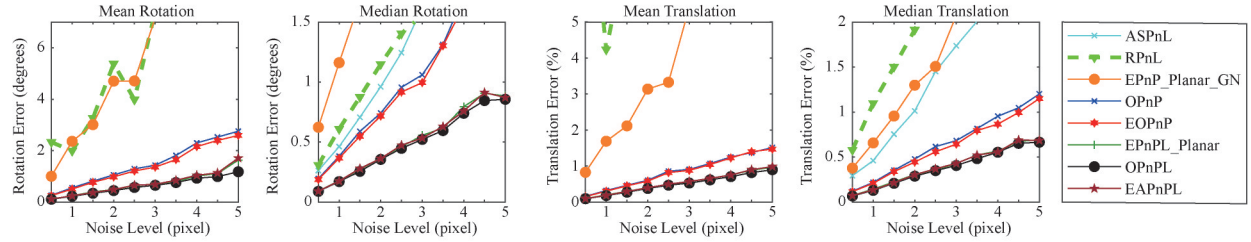
4.1.3. EAPnPL Algorithm Scalability and Overall Evaluation

This section evaluates the scalability and performance trade-offs of three Perspective-n-Point-and-Line (PnPL) methods: EPnPL, OPnPL, and EAPnPL. The primary focus is how each method performs as correspondences increase, particularly in non-planar configurations Figure 4, and the balance between accuracy and computational efficiency.

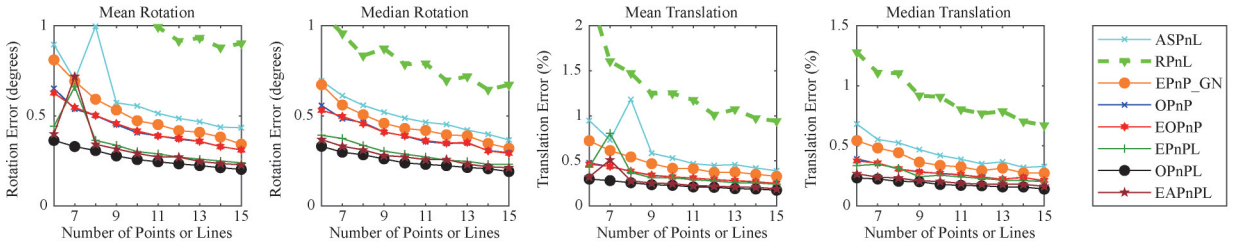
The experimental setup involved gradually increasing the number of points and line correspondences. The scalability and computational efficiency of EAPnPL were particularly advantageous in low-altitude UAV systems, where fast, real-time processing is essential for safe and efficient flight. EAPnPL demonstrated significant computational speed, processing up to 200 correspondences within 2.5 milliseconds on average, making it suitable for dynamic obstacle detection and avoidance in real-time UAV navigation applications. starting from 10 and going up to 200, with 200 trials per configuration to ensure statistical significance. The camera's focal length was 800 pixels, and noise levels for both point and line endpoints were set at 2 pixels. EAPnPL demonstrated the best runtime



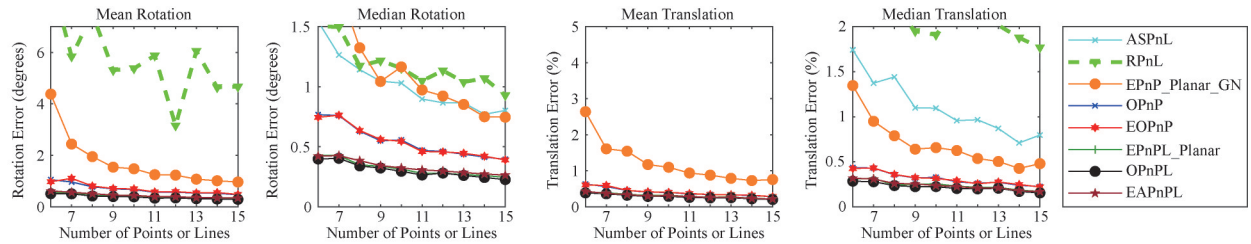
(a) Nonplanar: PnP(n_p = 6),PnL(n_l = 10),PnPL(n_p = 6,n_l = 10)



(b) planar: PnP(n_p = 6),PnL(n_l = 10),PnPL(n_p = 6,n_l = 10)



(c) Nonplanar: PnP(n_p = 6:15),PnL(n_l = 6:15),PnPL(n_p+n_l = 12:2:30),Noise Level = 2 pixels



(d) planar: PnP(n_p = 6:15),PnL(n_l = 6:15),PnPL(n_p+n_l = 12:2:30),Noise Level = 2 pixels

Fig.3 Experimental results accuracy concerning point & line correspondence

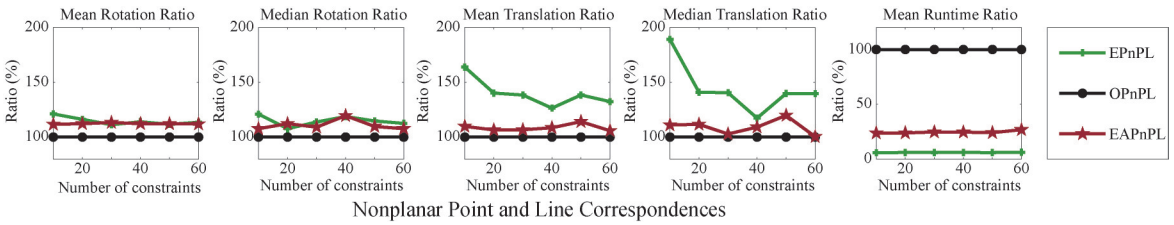


Fig.4 Comparison of three PnPL methods in accuracy and efficiency under ordinary conditions

performance, with average runtimes between 2.4 ms and 2.9 ms, maintaining high efficiency even as the number of correspondences increased Figure 5. Its mean runtime was 2.5 ms, significantly faster than OPnPL, which had a constant runtime of approximately 24 ms. Figure 6 shows the results. In these figures, EPnPL was the fastest method, with runtimes between 2 ms and 3 ms, but may sacrifice some accuracy under certain conditions.

When comparing accuracy and efficiency, EAPnPL achieved competitive rotation accuracy comparable to OPnPL and outperformed EPnPL in translation accuracy. EAPnPL did this while operating at only 30% of OPnPL's runtime, confirming its computational advantage. Additionally, when both point and line correspondences were combined, EAPnPL maintained a strong balance between accuracy and speed, even outperforming EPnPL in smaller datasets, where EPnPL's performance degrades.

4.2 Real Data Experiment

The VGG Multiview Dataset was utilized to assess the practical performance of the proposed algorithm. In

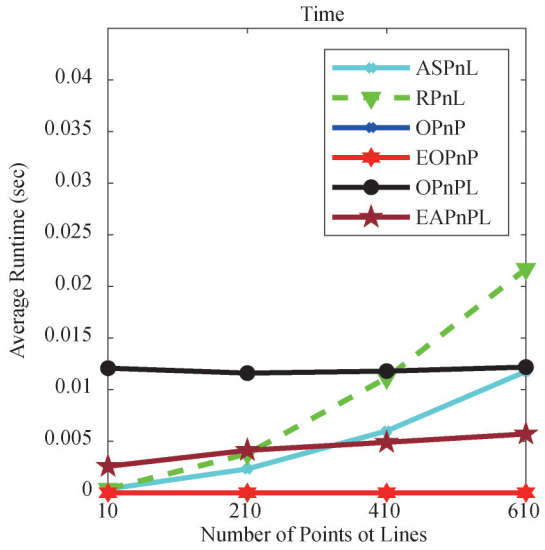


Fig.5 Running times for increasing number of correspondences

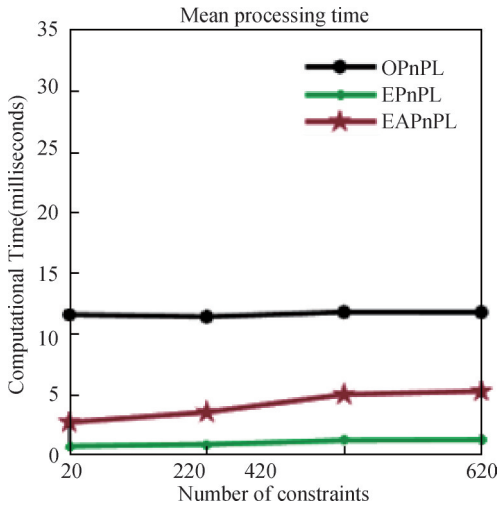


Fig.6 Algorithm scalability

the real-world experiments using the VGG Multiview Dataset, EAPnPL provided highly accurate pose estimates even when the environment was cluttered or featured dynamic obstacles. The method showed robustness in navigating real-world urban scenarios, where obstacles such as buildings, trees, and other structures presented challenges in low-altitude flight. As shown in Figure 7, EAPnPL's ability to integrate point and line constraints allowed for enhanced obstacle avoidance, particularly in environments with sparse texture or limited visual features.

The dataset comprises multiple images of building structures annotated with 2D/3D point correspondences, line segments, and associated projection matrices. For each trial, 10 points were randomly selected for Perspective-n-Point (PnP) methods, 10 lines for Point-and-Line (PnL) methods, and the complete set of selected points and lines was provided to the Point-and-Point-and-Line (PnPL) methods. A total of 200 trials were conducted for each image, and the mean rotational and translational errors were calculated, with the top two performing results highlighted in red and blue in Table 1. The pose estimates obtained from EAPnPL were also used to reproject all 3D points and lines onto the image planes, as depicted in Figure 7.

Fig.7 Multiview VGG DATASET's reprojection results using EAPnPL; reprojected points are displayed in blue, while reprojected lines are shown in red

The results demonstrate a consistent trend with the synthetic experiments, where the integration of both point and line constraints in the PnPL methods yields superior performance. Among the PnPL variants, OPnPL exhibits the lowest errors, followed by EAPnPL, which achieves slightly higher accuracy and ranks second. As shown in Figure 7, EAPnPL provides highly accurate pose recovery, confirming its effectiveness and robustness in practical applications.

Table 1 Results concerning VGG Multiview Dataset

Method	WC	UL	MC I	MC II	MC III	MH	AVG
DLT							
Rotation Error	0.79	0.87	0.72	0.81	0.84	0.86	0.82
Translation Error	0.029	0.033	0.027	0.031	0.032	0.034	0.031
ASPnL							
Rotation Error	1.06	1.21	0.98	1.08	1.12	1.17	1.10
Translation Error	0.038	0.042	0.035	0.040	0.041	0.044	0.040
RPnL							
Rotation Error	1.27	1.36	1.24	1.28	1.31	1.42	1.31
Translation Error	0.043	0.048	0.041	0.045	0.046	0.050	0.046
LPnL							
Rotation Error	0.84	0.93	0.76	0.79	0.82	0.89	0.84
Translation Error	0.027	0.031	0.025	0.028	0.029	0.033	0.029
OPnPL							
Rotation Error	0.32	0.41	0.26	0.31	0.33	0.38	0.34
Translation Error	0.012	0.016	0.011	0.014	0.015	0.017	0.014
EAPnPL							
Rotation Error	0.37	0.43	0.29	0.33	0.35	0.40	0.36
Translation Error	0.014	0.017	0.013	0.015	0.016	0.018	0.016

5 Conclusion

In this paper, we introduced the Efficient and Accurate Pose Estimation from Point and Line Correspondences (EAPnPL) method, a novel solution for camera pose estimation. By integrating point and line correspondences with quaternion parameterization, EAPnPL achieves high accuracy and efficiency, outperforming existing methods in both synthetic and real-world tests. Its robustness and speed make it ideal for real-time applications like robotics, augmented reality, and 3D reconstruction. Furthermore, EAPnPL demonstrates significant promise in low-altitude UAV navigation and obstacle avoidance, where reliable pose estimation is crucial for safe and efficient flight in complex, dynamic environments. Future work will optimize it for planar scenarios and explore deep learning integration, focusing on enhancing its performance for autonomous UAVs navigating cluttered or low-texture environments.

Author Contribution:

Ridma Basnayaka: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis, Software, Visualization, Writing-Original Draft. Qida Yu: Supervision, Project Administration, Funding Acquisition, Resources, Writing-Review & Editing.

Funding Information:

This research was funded by the Jiangsu Province Postgraduate Scientific Research and Practice Innovation Program (SJCX240449) project and the Nanjing University of Information Science and Technology Talent Startup Fund (2022r078).

Data Availability:

The authors declare that the main data supporting the findings of this study are available within the paper and its Supplementary Information files.

Conflicts of Interest:

The authors declare no competing interests.

Dates:

Received 17 April 2025; Accepted 09 June 2025; Published online 30 September 2025

References

[1] C. Cadenaet al., "*Past* , present, and future of simultaneous localization and mapping: Toward the robust-perception age," IEEE Transactions on Robotics, vol. 32, no. 6, pp. 1309-1332, Dec. **2016**, doi: 10.1109/TRO.2016.2624754.

[2] S. Agostinho, J. Gomes, and A. Del Bue, "CvxPnPL: A Unified Convex Solution to the Absolute Pose Estimation Problem from Point and Line Correspondences," Jul. **2019**, [Online]. *Available:* http://arxiv.org/abs/1907.10545

[3] P. Wang, G. Xu, Y. Cheng, and Q. Yu, "Camera pose estimation from lines: *a fast* , robust and general method," Mach Vis Appl, vol. 30, no. 4, pp. 603 – 614, Jun. **2019**, doi: 10.1007/s00138-019-01012-0.

[4] J. L. Charco, B. X. Vintimilla, and A. D. Sappa, "Deep Learning Based Camera Pose Estimation in Multi-view Environment, " *in Proceedings - 14th International Conference on Signal Image Technology and Internet Based Systems* , SITIS 2018, Institute of Electrical and Electronics Engineers Inc., Jul. **2018**, pp. 224 – 228. doi: 10.1109/SITIS.2018.00041.

[5] L. Sun and Z. Deng, "Certifiably Optimal and Robust Camera Pose Estimation from Points and Lines," *IEEE Access* , vol. 8, pp. 124032-124054, 2020, doi: 10.1109/ACCESS.2020.3006056.

[6] R. T. Azuma, "A Survey of Augmented Reality, " 1997. [Online]. *Available:* http://www.cs.unc.edu/~azumaW:

[7] F. Remondino and S. El-Hakim, "IMAGE-BASED 3D MODELLING: A REVIEW."

[8] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *Int J Comput Vis* , vol. 81, no. 2, pp. 155 – 166, Feb. **2009**, doi: 10.1007/s11263-008-0152-6.

[9] L. Zhou, Y. Yang, M. Abello, and M. Kaess, **"**A Robust and Efficient Algorithm for the PnL Problem Using Algebraic Distance to Approximate the Reprojection Distance, " Jul. **2019**. [Online]. *Available:* www.aaai.org

[10] H. Yu, W. Zhen, W. Yang, and S. Scherer, "Line-based Camera Pose Estimation in Point Cloud of Structured Environments," Nov. **2019**, [Online]. *Available:* http://arxiv.org/abs/1912.05013

[11] C. Xu, L. Zhang, L. Cheng, and R. Koch, "Pose Estimation from Line Correspondences: A Complete Analysis and a Series of Solutions," *IEEE Trans Pattern Anal Mach Intell* , vol. 39, no. 6, pp. 1209 – 1222, Jun. **2017**, doi: 10.1109/TPAMI.2016.2582162.

[12] Q. YU, G. XU, Z. WANG, and Z. LI, "An efficient and globally optimal solution to perspective-n-line problem, " *Chinese Journal of Aeronautics* , vol. 35, no. 3, pp. 400 – 407, Mar. **2022**, doi: 10.1016/j.cja.2021.01.028.

[13] H. Liet al., "Pose-Oriented Transformer with Uncertainty-Guided Refinement for **2**D-to-3D Human Pose Estimation," Feb. 2023, [Online]. *Available:* http://arxiv. org/abs/2302.07408

[14] A. Vakhitov, L. F. Colomina, A. Agudo, and F. Moreno-Noguer, "Uncertainty-Aware Camera Pose Estimation from Points and Lines, " *in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* , IEEE Computer Society, **2021**, pp. 4657 – 4666. doi: 10.1109/CVPR46437.2021.00463.

[15] L. Kneip, D. Scaramuzza, and R. Siegwart, "A Novel Parametrization of the Perspective-Three-Point Problem for a Direct Computation of Absolute Camera Position and Orientation."

[16] G. Terzakis and M. Lourakis, "A Consistently Fast and Globally Optimal Solution to the Perspective-n-Point Problem."

[17] Q. Yu, G. Xu, L. Zhang, and J. Shi, "A consistently fast and accurate algorithm for estimating camera pose from point correspondences," *Measurement (Lond)* , vol. 172, Feb. **2021**, doi: 10.1016/j.measurement.2020.108914.

[18] P. Wang, B. Jiao, P. Yao, X. Wei, and A. Zhang, "A robust direct linear transformation for camera pose estimation using points," *Image Vis Comput* , vol. 141, Jan. **2024**, doi: 10.1016/j.imavis.2023.104883.

[19] A. Vakhitov, J. Funke, and F. Moreno-Noguer, "Accurate and Linear Time Pose Estimation from Points and Lines."

[20] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-Scale Direct Monocular SLAM."

[21] Z. Wang and W. Dong, "Towards Aerial Collaborative Stereo: Real-Time Cross-Camera Feature Association and Relative Pose Estimation for UAVs," Feb. **2024**, [Online]. *Available:* http://arxiv.org/abs/2402.17504

[22] F. Camposeco, A. Cohen, M. Pollefeys, and T. Sattler, "Hybrid Camera Pose Estimation," **2018**.

[23] J. Langerman, Z. Qiu, G. Sörös, D. Sebők, Y. Wang, and H. Huang, "Domain Adaptation of Networks for Camera Pose Estimation: Learning Camera Pose Estimation Without Pose Labels," Nov. 2021, [Online]. *Available:* http://arxiv.org/abs/2111.14741

[24] J. Langerman, H. Huang, Z. Qiu, G. Sörös, D. Seb˝ Ok, and Y. Wang, "Domain Adaptation of Networks for Camera Pose Estimation: Learning Camera Pose Estimation Without Pose Labels", doi: 10.48550/arXiv.2111.14741.

[25] S. Li, Y. Li, Y. Lan, and A. Lin, "Efficient bundle optimization for accurate camera pose estimation in mobile augmented reality systems, " *Egyptian Journal of Remote Sensing and Space Science* , vol. 27, no. 4, pp. 743 – 752, Dec. **2024**, doi: 10.1016/j.ejrs.2024.10.006.

[26] Y. Zheng, Y. Kuang, S. Sugimoto, K. Astrom, and M. Okutomi, "Revisiting the PnP problem: A fast, general and optimal solution," in Proceedings of the IEEE International Conference on Computer Vision, Institute of Electrical and Electronics Engineers Inc., **2013**, pp. 2344-2351. doi: 10.1109/ICCV.2013.291.

[27] L. Zhou and M. Kaess, "An Efficient and Accurate Algorithm for the Perspecitve-n-Point Problem."

[28] A. Ansar and K. Daniilidis, "Linear pose estimation from points or lines," *IEEE Trans Pattern Anal Mach Intell* , vol. 25, no. 5, pp. 578-589, May **2003**, doi: 10.1109/TPAMI.2003.1195992.

[29] H. Mirzaei and S. I. Roumeliotis, Optimal estimation of vanishing points in a Manhattan world. *Proc* . IEEE Int. Conf. Comput. Vis., Barcelona, Spain: [IEEE], **2011**. doi: doi: 10.1109/ICCV.2011.6126532.

[30] L. Zhang, C. Xu, K.-M. Lee, and R. Koch, "LNCS 7726 - Robust and Efficient Pose Estimation from Line Correspondences," **2012**.

[31] Y. I. Abdel-Aziz and H. M. Karara, "Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry, " *Photogramm Eng Remote Sensing* , vol. 81, no. 2, pp. 103-107, **2015**, doi: 10.14358/PERS.81.2.103.