

# EPnL: 一种高效且精确的 PnL 问题求解算法

王平<sup>1</sup> 何卫隆<sup>1</sup> 张爱华<sup>1</sup> 姚鹏鹏<sup>2</sup> 徐贵力<sup>3</sup>

**摘要** 现有 Perspective-n-line (PnL) 问题求解算法无法在获得高求解精度的同时保证高求解效率. 为解决这个缺点, 提出了同时兼具求解效率和求解精度算法 EPnL. 该方法首先将 PnL 问题转换为求二次曲面方程组交点的问题, 然后利用单位四元数中变量不同时为零的特性, 分类参数化 PnL 问题中的旋转矩阵. 最后, 为克服常规优化方法可靠性和效率较低的问题, 同时兼具求解效率和求解精度算法利用二次曲面方程组自身的结构信息, 采用低次项参数化高次项的方式将二次曲面方程组的求解问题转换为单变量多项式的求解问题. 实验表明, 相比于现有算法, 该算法在具有高求解精度的同时也兼具高求解效率.

**关键词** 计算机视觉, PnL 问题, 位姿估计, 视觉导航

**引用格式** 王平, 何卫隆, 张爱华, 姚鹏鹏, 徐贵力. EPnL: 一种高效且精确的 PnL 问题求解算法. 自动化学报, 2022, 48(10): 2600–2610

**DOI** 10.16383/j.aas.c200927

## EPnL: An Efficient and Accurate Algorithm to the PnL Problem

WANG Ping<sup>1</sup> HE Wei-Long<sup>1</sup> ZHANG Ai-Hua<sup>1</sup> YAO Peng-Peng<sup>2</sup> XU Gui-Li<sup>3</sup>

**Abstract** The existing algorithms for solving the perspective-n-line (PnL) problem cannot achieve high accuracy as well as maintain high computational efficiency. To solve this disadvantage, the EPnL algorithm is proposed. The EPnL firstly transfers the PnL problem into the problem of finding the intersection of quadratic surface equations, and then uses a classification parametrization, which is derived from the fact that variables are not simultaneously zero in the unit-quaternion, to parameterize the rotation matrix in the PnL problem. At last, to overcome the problem of low reliability and low efficiency of conventional optimization methods, the EPnL uses the structure information of the system of quadratic surface equations, and transfers the problem of finding the intersection of quadratic surface equations into the solving problem of univariate polynomials using the strategy that the higher order terms are parameterized by the lower order terms in the system of quadratic surface equations. Experimental results show that the proposed algorithm has high accuracy as well as high efficiency compared with existing algorithms.

**Key words** Computer vision, PnL problem, pose estimation, vision-based navigation

**Citation** Wang Ping, He Wei-Long, Zhang Ai-Hua, Yao Peng-Peng, Xu Gui-Li. EPnL: An efficient and accurate algorithm to the PnL problem. *Acta Automatica Sinica*, 2022, 48(10): 2600–2610

收稿日期 2020-11-09 录用日期 2021-03-19

Manuscript received November 9, 2020; accepted March 19, 2021

国家自然科学基金 (62001198, 62073161, 61866021), 流程工业综合自动化国家重点实验室开放基金 (PAL-N201808), 甘肃省国际合作科技计划 (18YF1WA068), 甘肃省青年科技基金 (20JR10RA-186) 资助

Supported by National Natural Science Foundation of China (62001198, 62073161, 61866021), State Key Laboratory of Synthetical Automation for Process Industries (PAL-N201808), International Cooperation Science and Technology Program of Gansu Province (18YF1WA068), and Gansu Province Science Foundation for Youths (20JR10RA186)

本文责任编辑 孙秋野

Recommended by Associate Editor SUN Qiu-Ye

1. 兰州理工大学电气工程与信息工程学院 兰州 730050 中国  
2. 香港理工大学纺织与制衣学系 香港 999077 中国 3. 南京航空航天大学自动化学院 南京 211106 中国

1. College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China 2. Institute of Textiles and Clothing, Hong Kong Polytechnic University, Hong Kong 999077, China 3. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

基于直线特征的摄像机绝对位姿估计问题在计算机视觉领域称之为 Perspective-n-line (PnL) 问题, 目的是通过目标物体上的  $n$  条已知直线和其所对应的图像投影来计算相机和目标之间的相对位置和姿态关系. PnL 问题是视觉导航<sup>[1-2]</sup>、机器人视觉定位<sup>[3]</sup>、传感器标定<sup>[4]</sup>、现实增强<sup>[5]</sup> 等领域中的关键核心问题之一. 到目前为止, 现有解决 PnL 问题的方法可以粗略分为迭代法和非迭代法:

迭代法将 PnL 求解问题转换为非线性最优化问题, 并利用 Gauss-Newton 法或 Levenberg-Marquardt 法<sup>[6]</sup> 迭代求解这个非线性最优化问题. 然而, 迭代法对初值的选取比较敏感, 初值选择不合理将导致迭代法收敛速度较慢, 影响算法的实时性. 除此之外, 当使用特征直线较少的时候, 迭代法容易陷入局部最优, 影响 PnL 问题求解的精度和可靠性.

在非迭代法中,最直接的当属直接线性变换(Direct linear transformation, DLT)方法<sup>[7]</sup>. DLT方法简单高效,但抗噪能力不强,在空间参考直线较少的情况下求解精度不高. Přibyl 等<sup>[8]</sup>通过将直线在欧氏空间的坐标表示转换到普吕克(Plücker)空间,提出了解决 PnL 问题的 DLT-Plücker-lines 方法. 相比于 DLT 算法, DLT-Plücker-lines 算法具有更好的抗噪能力和求解精度,但其求解时需要至少 9 条以上的空间参考直线. 随后, Xu 等<sup>[9]</sup>通过借鉴 Perspective-n-point (PnP) 算法原理,提出了一系列线性求解 PnL 的方法,但这些方法仍需要 6 条以上的空间参考直线才可以求解. 最近, Přibyl 等<sup>[10]</sup>基于 DLT 的方法,提出了 DLT-combined-lines 的方法,该方法将线性求解 PnL 问题的直线数目缩减到了 5 条. DLT-combined-lines 方法求解效率高,在直线数目较多的时候,具有很高的求解精度,但是在直线数目较少的情况下,求解精度较差.

虽然直接线性求解 PnL 的方法具有简单、效率高的特点. 但是由于其抗噪能力差、不适合参考直线较少的情况(尤其不适合参考直线少于 6 条以下的情况). 为了克服这些问题, Ansar 等<sup>[11]</sup>将 PnL 问题转换为非迭代优化. 开发了一种通用的 PnL 求解算法,该算法能够处理从 4 到  $n$  条所有的空间参考直线. 然而,文献 [11] 算法在空间直线较少的时候求解精度不高,这主要是因为其最优化求解过程中采用了线性化的策略. 为了提高 PnL 算法的求解精度, Mirzaei 等<sup>[12]</sup>提出了一种直接求解 PnL 问题全局最优解的方法,通过 Cayley 参数的方式参数化旋转矩阵,然后通过矩阵分解和合成的方式,将 PnL 位姿测量问题转换为最优化问题,最后通过矩阵合成技术求解这个最优化问题,得到 PnL 问题的解. 然而,文献 [12] 的方法由于使用了 Cayley 参数表示旋转矩阵,求解过程中容易出现矩阵奇异值,导致求解稳定性不好. 为了克服这个问题, Zhang 等<sup>[13]</sup>提出了 RPnL 方法求解 PnL 问题,该方法将空间参考直线三条线为一组,然后利用 Perspective-3-line (P3L) 约束构建多项式方程组来求解 PnL 问题. 然而, RPnL 是一种次优化的方法,其求解精度还有提升的空间. 2017 年, Xu 等<sup>[9]</sup>改进了 RPnL 方法,提出了 ASPnL 算法. 到目前为止, ASPnL 是求解精度最高,最稳定的 PnL 算法之一. 然而, ASPnL 随着直线数量的增加,其求解消耗时间将快速增长,不利于其应用在实时性较高的任务中. 2019 年, Wang 等<sup>[14]</sup>对 RPnL 算法增加优化步骤,并且使用矩阵化的方式替代 RPnL 算法中的循环步骤,提出了改进的 SRPnL 算法,该算法求解精度和可靠性类似于 ASPnL 算法,但效率远高于 ASPnL 算法.

从以上文献分析可以看出,线性求解 PnL 问题的方法效率高,但精度低,通常无法适用于直线数量小于 6 的情况. 非线性求解 PnL 的方法适应性较广(适应  $4 \sim n$  条直线求解),求解精度高,但是求解过程复杂,效率低下. 针对以上问题,本文提出了一种同时兼具求解效率和求解精度的方法(命名为 EPnL 方法). 本文的主要贡献有:

1) 提出了分类表示旋转的方法:最近的 PnL 问题求解中多使用 Cayley 参数<sup>[13-15]</sup>和四元数表示<sup>[16]</sup>旋转,然而利用 Cayley 参数求解 PnL 问题容易出现奇异值,导致求解结果不稳定. 利用四元数表示旋转则存在解的符号模糊问题,会增加 PnL 问题求解的复杂性,降低求解效率. 针对以上缺点,本文基于四元数参数中变量不同时为零的特性,提出了分类表示旋转的方法,在保证不损失旋转正交约束信息的前提下,避免了 Cayley 参数求解奇异性和四元数参数对解符号模糊的问题,提升了求解 PnL 问题的可靠性和效率.

2) 将 PnL 问题转换为了二次曲面(曲线)方程组求交点的问题:本文首先通过矩阵变化的方式统一求解参数的度量空间,消除求解参数度量空间不同可能引起的算法不稳定因素. 然后,不同于现有文献 [12-14] 直接最优化求解 PnL 问题的方式,本文基于 1) 中分类表示旋转的方法,将 PnL 问题转换为二次曲面(或曲线)求交点的问题来解决,有效地降低了 PnL 问题求解的复杂度.

3) 利用方程组低次项参数化高次项的方式将复杂二次曲面(曲线)方程组求交点问题转换为单变量多项式求解问题:针对迭代法求解耗时且容易陷入局部最优,而 Gröbner 基方法<sup>[17]</sup>求解复杂,无法保障可靠性的问题. 本文利用二次曲面方程组自身的结构信息,将方程组划分为高次项和低次项部分,并通过引入恒等式的方式将高次项用低次项表示,最终将复杂的多变量二次曲面求解问题转换为简单的单变量多项式(最高为 8 次)求解问题来解决. 同时,本文利用少量 Gauss-Newton 法对结果进行精定位,以进一步提升最终的求解精度.

4) 本文提出的算法实现了求解精度和效率的统一. 实验部分选择和主流及最新的 PnL 算法对比,结果表明,EPnL 适用于  $3 \sim n$  条直线的位姿求解,具有通用性. 在所有对比算法中,EPnL 算法求解精度最高,效率仅次于线性 DLT 的方法(排名第 2). 本文中所有方法的源代码已公布<sup>1</sup>,读者可以下载验证.

<sup>1</sup> <https://sites.google.com/view/ping-wang-homepage>

### 1 PnL 问题的描述

基于相机的透视成像模型, PnL 问题如图 1 表示, 其中  $L_i = (v_i, P_i)$  表示 3D 空间中的某条已知直线,  $v_i \in \mathbf{R}^3$  为  $L_i$  的方向向量,  $P_i \in \mathbf{R}^3$  为  $L_i$  上的任意一点.  $l_i = (s_i, p_i)$  为  $L_i$  在图像平面上的投影直线,  $s_i$  和  $p_i$  分别为  $l_i$  的两个端点. 直线  $L_i$ ,  $l_i$  和相机光心  $O$  共同形成平面  $\pi_i$ , 且  $n_i$  为平面  $\pi_i$  的法向量. 当相机内参数标定的情况下,  $n_i$  很容易由光心和投影直线两个端点所形成向量的外积得到, 即:

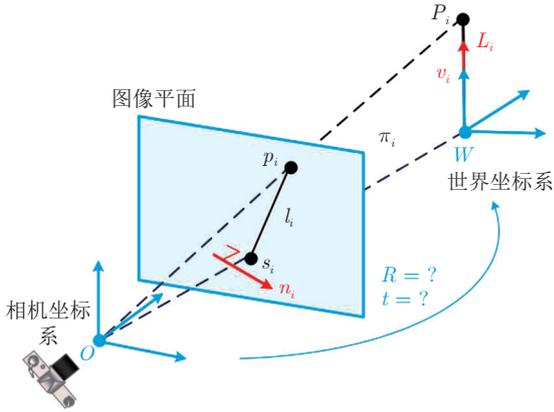


图 1 PnL 问题  
Fig.1 PnL problem

$$n_i = Os_i \times Op_i, \quad i = 1, 2, \dots, n \quad (1)$$

由于直线  $L_i$  在平面  $\pi_i$  内, 因此其和平面法向量  $n_i$  满足垂直的关系, 即:

$$n_i = (RP_i + t) \times (Rv_i), \quad i = 1, 2, \dots, n \quad (2)$$

式中,  $R \in SO(3)$  为  $3 \times 3$  矩阵, 表示相机坐标系和空间直线所在坐标系 (世界坐标系) 之间的旋转关系;  $t \in \mathbf{R}^3$  为  $3 \times 1$  向量, 表示相机坐标系和世界坐标系之间的平移关系. 式 (2) 可进一步展开为:

$$n_i^T Rv_i = 0, \quad i = 1, 2, \dots, n \quad (3)$$

$$n_i^T (RP_i + t) = 0, \quad i = 1, 2, \dots, n \quad (4)$$

PnL 问题的目标就是在空中直线  $L_i$  和其投影  $l_i$  已知的情况下, 利用式 (3) 和式 (4) 计算  $R$  和  $t$ .

### 2 约束方程组的构建

PnL 问题中旋转矩阵  $R$  和平移向量  $t$  的度量空间是不同的, 这在优化求解过程中容易导致系数矩阵中元素数值的差异过大, 最终影响 PnL 问题的求解精度. 由式 (4) 可以看出, 平移  $t$  和旋转  $R$  之间呈现线性关系. 因此, 本文采用矩阵变化的方式, 基于

空间所有的直线信息, 利用旋转  $R$  参数化平移  $t$ , 进而统一 PnL 问题中待求解参数变量的度量空间, 以最终提升求解 PnL 问题的可靠性和精度. 式 (3) 进一步可以表示为:

$$n_i^T Rv_i - 0 \cdot t = 0, \quad i = 1, 2, \dots, n \quad (5)$$

综合式 (4) 和式 (5), 将其写为矩阵形式:

$$\underbrace{\begin{bmatrix} n_i^T & 0 \\ 0 & n_i^T \end{bmatrix}}_{A_i} \underbrace{\begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix}}_C \underbrace{\begin{bmatrix} v_i \\ P_i \end{bmatrix}}_{D_i} = \underbrace{\begin{bmatrix} 0 \\ -n_i^T \end{bmatrix}}_{B_i} t \quad (6)$$

$$\Leftrightarrow A_i C D_i = B_i t, \quad i = 1, 2, \dots, n$$

式 (6) 对空间中的每一条参考直线都满足, 因此有:

$$\underbrace{\begin{bmatrix} A_1 & & \\ & \ddots & \\ & & A_n \end{bmatrix}}_A \underbrace{\begin{bmatrix} C & & \\ & \ddots & \\ & & C \end{bmatrix}}_W \underbrace{\begin{bmatrix} D_1 \\ \vdots \\ D_n \end{bmatrix}}_D = \underbrace{\begin{bmatrix} B_1 \\ \vdots \\ B_n \end{bmatrix}}_B t$$

$$\Leftrightarrow A W D = B t \quad (7)$$

基于式 (7),  $t$  可以表示为:

$$t = B^+ A W D \quad (8)$$

式中,  $B^+ = (B^T B)^{-1} B^T$  为  $B$  的广义逆. 由式 (8) 中可以看出,  $B^+$ ,  $A$  和  $D$  中包含空间所有直线提供的已知信息,  $W$  由旋转  $R$  构成. 因此基于式 (8), 平移向量  $t$  可以表示为旋转  $R$  的参数化形式. 进一步将式 (8) 代入式 (6) 得到:

$$A_i C D_i = B_i B^+ A W D, \quad i = 1, 2, \dots, n \quad (9)$$

式 (9) 依旧对空间的  $n$  条直线都满足, 将式 (9) 变为矩阵形式得到:

$$\begin{bmatrix} A_1 C D_1 - B_1 B^+ A W D \\ A_2 C D_2 - B_2 B^+ A W D \\ \vdots \\ A_n C D_n - B_n B^+ A W D \end{bmatrix} = 0 \quad (10)$$

由式 (10) 可以看出,  $A$ 、 $D$ 、 $B^+$ 、 $A_i$ 、 $D_i$  和  $B_i$  ( $i = 1, 2, \dots, n$ ) 均可以由已知的  $n_i$ 、 $v_i$  和  $P_i$  提前计算得到. 式 (10) 中的未知数仅由  $C$  和  $W$  提供, 而  $C$  和  $W$  由旋转未知变量  $R$  构成. 此时, 如果能够利用式 (10) 求解得到旋转变量  $R$ , 则平移向量  $t$  可以由式 (8) 给出, 即 PnL 问题得到求解.  $R$  和  $t$  的具体求解方法, 本文将在第 4 节重点展开研究.

### 3 基于四元数的旋转分类表示法

由式 (10) 可以看出,  $R$  的表示形式直接影响着

式 (10) 的复杂程度和求解系数矩阵的奇异性, 进而间接影响优化求解 PnL 算法的精度、可靠性和效率.  $R$  的表示形式通常包括: 欧拉角表示形式、旋转矩阵表示形式、Cayley 参数表示形式、四元数表示形式、对偶四元数表示形式、角轴参数表示形式. 其中四元数表示含有正交约束信息, 且其形式不具有奇异性, 因此利用四元数解决 PnL 问题具有精度和可靠性高的特点. 四元数表示  $R$  的形式如下:

$$R = \begin{bmatrix} a^2 + b^2 - c^2 - d^2, 2bc - 2ad, 2bd + 2ac \\ 2bc + 2ad, a^2 - b^2 + c^2 - d^2, 2cd - 2ab \\ 2bd - 2ac, 2cd + 2ab, a^2 - b^2 - c^2 + d^2 \end{bmatrix} \quad (11)$$

式中,  $a$ 、 $b$ 、 $c$  和  $d$  为变量且满足约束条件  $a^2 + b^2 + c^2 + d^2 = 1$ . 由式 (11) 可以看出, 利用四元数求解旋转时,  $[a, b, c, d]^T$  和  $[-a, -b, -c, -d]^T$  表示相同的旋转, 这说明四元数对变量的正、负号无法分辨, 这无疑将扩大求解 PnL 问题时解的搜索空间, 增加求解问题时的复杂性, 降低求解问题的效率.

由四元数约束  $a^2 + b^2 + c^2 + d^2 = 1$  可以看出, 四元数在表示旋转时, 参数  $a$ 、 $b$ 、 $c$  和  $d$  是不同时为 0 的. 基于这个特点, 根据  $a$ 、 $b$ 、 $c$  和  $d$  的取值, 本文提出基于四元数的旋转  $R$  分类表示方法:

1) 形式 1. 当  $a$ 、 $b$ 、 $c$  和  $d$  都不为 0, 任选一个变量 (如  $a$ ), 在  $R$  右端提取出  $1/a^2$  且令  $s_1 = b/a$ ,  $s_2 = c/d$  和  $s_3 = d/a$ , 则  $R$  变为:

$$R = \frac{1}{H} \begin{bmatrix} 1 + s_1^2 - s_2^2 - s_3^2, 2s_1s_2 - 2s_3, 2s_1s_3 + 2s_2 \\ 2s_1s_2 + 2s_3, 1 - s_1^2 + s_2^2 - s_3^2, 2s_2s_3 - 2s_1 \\ 2s_1s_3 - 2s_2, 2s_2s_3 + 2s_1, 1 - s_1^2 - s_2^2 + s_3^2 \end{bmatrix} \quad (12)$$

式中,  $H = 1 + s_1^2 + s_2^2 + s_3^2$ . 式 (12) 本质上就是旋转的 Cayley 参数表示形式, 可以看出, 当变量  $a$  的值为零的时候, Cayley 参数表示矩阵没有意义, 这将导致位姿求解出现奇异性.

2) 形式 2. 当  $a$ 、 $b$ 、 $c$  和  $d$  中有一个变量为 0, 例如  $a = 0$ ,  $b \neq 0$ ,  $c \neq 0$  和  $d \neq 0$ . 则任选一个不为 0 的变量, 例如用  $b$  来化简  $R$ , 则可得:

$$R = \frac{1}{H} \begin{bmatrix} 1 - s_2^2 - s_3^2 & 2s_2 & 2s_3 \\ 2s_2 & -1 + s_2^2 - s_3^2 & 2s_2s_3 \\ 2s_3 & 2s_2s_3 & -1 - s_2^2 + s_3^2 \end{bmatrix} \quad (13)$$

式中,  $s_2 = c/b$ ,  $s_3 = d/b$ , 且  $H = 1 + s_2^2 + s_3^2$ .

3) 形式 3. 当  $a$ 、 $b$ 、 $c$  和  $d$  中有两个变量为 0, 例如  $a = 0$ ,  $b = 0$ ,  $c \neq 0$  和  $d \neq 0$ . 任选  $c$  和  $d$  中的

一个变量, 如  $c$  化简  $R$  得到:

$$R = \frac{1}{H} \begin{bmatrix} -1 - s_3^2 & 0 & 0 \\ 0 & 1 - s_3^2 & 2s_3 \\ 0 & 2s_3 & -1 + s_3^2 \end{bmatrix} \quad (14)$$

式中,  $s_3 = d/c$ , 并且有  $H = 1 + s_3^2$ .

4) 形式 4. 当  $a$ 、 $b$ 、 $c$  和  $d$  中仅有一个变量不为 0, 例如  $a = 0$ ,  $b = 0$ ,  $c = 0$  和  $d \neq 0$ , 则  $R$  直接表示为:

$$R = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

通过以上分类表示  $R$  的方式, 在保证不损失四元数正交约束信息的前提下, 避免了四元数对解符号模糊的问题, 有效降低了求解 PnL 问题时未知变量的个数和解空间的维度, 在理论层面保障了求解 PnL 问题的可靠性和效率.

## 4 旋转矩阵和平移向量的求解

本节将旋转  $R$  的 4 种形式分别代入式 (10) 进行求解.

1) 对于形式 1. 将式 (12) 代入式 (10), 进一步展开, 合并同类项得到:

$$E_1\beta_1 = 0 \quad (16)$$

式中,  $E_1$  为已知的  $2n \times 10$  矩阵, 可以由  $A$ 、 $D$ 、 $B^+$ 、 $A_i$ 、 $D_i$  和  $B_i$  ( $i = 1, 2, \dots, n$ ) 计算得到.  $\beta_1 = [1, s_1, s_2, s_3, s_1^2, s_1s_2, s_1s_3, s_2^2, s_2s_3, s_3^2]^T$  为式 (12) 的向量形式. 对于式 (16), 可以采用 Newton 法和 Gauss-Newton 法<sup>[6]</sup> 等迭代法求解. 然而, 迭代法对初值的选择敏感, 初值选取不好的时候容易导致迭代陷入局部最优. 此外, 迭代法求解过程中需要耗费较多的时间. 为克服此问题, 最近的研究<sup>[15-16]</sup> 中将类似式 (16) 的等式转换为无约束最优化问题, 通过 Gröbner 基方法<sup>[17]</sup> 构建矩阵消去模板来求解. 然而, Gröbner 基方法求解式 (16) 时需要对其两端取平方, 这将引入较多高次未知数, 导致 Gröbner 基方法构建的矩阵消去模板过于复杂, 无法保证求解的可靠性. 除此之外, 矩阵消去模板越复杂, 算法的计算效率也越低, 并且复杂的矩阵消去模板更不利于算法的工程实现.

由式 (16) 可以看出,  $\beta_1$  中含有 3 个未知数, 且所含未知数的最高次数为 2. 因此式 (16) 中每一行都代表着一个二次曲面, 可以将求解式 (16) 看作为求解一组二次曲面交点的问题. 观察式 (16) 结构可以发现, 将变量  $s_1$  看作为常量,  $s_2$  和  $s_3$  为变量, 则

$E_1$  可以被  $[s_2^2, s_3^2, s_2s_3]^T$  和  $[s_2, s_3, 1]^T$  划分为两部分:

$$\begin{aligned}
 & [E_1(8), E_1(10), E_1(9)] \begin{bmatrix} s_2^2 \\ s_3^2 \\ s_2s_3 \end{bmatrix} = -[E_1(3) + \\
 & E_1(6)s_1, E_1(4) + E_1(7)s_1, E_1(1) + E_1(2)s_1 + \\
 & E_1(5)s_1^2] \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} \Leftrightarrow F_1 \begin{bmatrix} s_2^2 \\ s_3^2 \\ s_2s_3 \end{bmatrix} = \\
 & G_1(s_1) \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} \tag{17}
 \end{aligned}$$

式中,  $E_1(i), i = 1, 2, \dots, 10$  表示  $E_1$  的  $i$  列,  $G_1(s)$  中的每一个元素都是变量  $s_1$  的多项式. 式 (17) 进一步可变为:

$$\begin{bmatrix} s_2^2 \\ s_3^2 \\ s_2s_3 \end{bmatrix} = H_1(s_1) \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} \tag{18}$$

式中,  $H_1(s_1) = (F_1^T F_1)^{-1} F_1^T G_1(s_1)$  为  $3 \times 3$  的矩阵, 其中的每一分量都是  $s_1$  的函数, 可以表示为:

$$H_1(s_1) = \begin{bmatrix} H_{11}(s_1) & H_{12}(s_1) & H_{13}(s_1) \\ H_{21}(s_1) & H_{22}(s_1) & H_{23}(s_1) \\ H_{31}(s_1) & H_{32}(s_1) & H_{33}(s_1) \end{bmatrix}$$

通过以上划分方式, 可以将式 (16) 中的高次项部分通过低次项部分来表示. 利用式 (16) 中高次项和低次项之间的关系引入以下恒等式:

$$\begin{aligned}
 & (s_2^2) s_3 = (s_2 s_3) s_2 \\
 & (s_2 s_3) s_3 = (s_3^2) s_2 \\
 & (s_2 s_3) (s_2 s_3) = (s_2^2) (s_3^2) \tag{19}
 \end{aligned}$$

通过式 (19) 的恒等式, 可以建立起式 (16) 中高次项之间的联系. 将式 (18) 再次代入式 (19), 展开可得:

$$\begin{aligned}
 & (H_{11}(s_1)s_2 + H_{12}(s_1)s_3 + H_{13}(s_1)) s_3 = \\
 & (H_{31}(s_1)s_2 + H_{32}(s_1)s_3 + H_{33}(s_1)) s_2 \tag{20}
 \end{aligned}$$

$$\begin{aligned}
 & (H_{31}(s_1)s_2 + H_{32}(s_1)s_3 + H_{33}(s_1)) s_3 = \\
 & (H_{21}(s_1)s_2 + H_{22}(s_1)s_3 + H_{23}(s_1)) s_2 \tag{21}
 \end{aligned}$$

$$\begin{aligned}
 & (H_{31}(s_1)s_2 + H_{32}(s_1)s_3 + H_{33}(s_1))^2 = \\
 & (H_{11}(s_1)s_2 + H_{12}(s_1)s_3 + H_{13}(s_1)) \cdot \\
 & (H_{31}(s_1)s_2 + H_{32}(s_1)s_3 + H_{33}(s_1)) \tag{22}
 \end{aligned}$$

将式 (20)、式 (21) 和式 (22) 展开为:

$$\begin{aligned}
 & e_{11}s_2^2 + e_{12}s_2s_3 + e_{13}s_2 + e_{14}s_3^2 + e_{15}s_3 = 0 \\
 & e_{21}s_2^2 + e_{22}s_2s_3 + e_{23}s_2 + e_{24}s_3^2 + e_{25}s_3 = 0 \\
 & e_{31}s_2^2 + e_{32}s_2s_3 + e_{33}s_2 + e_{34}s_3^2 + e_{35}s_3 + e_{36} = 0 \tag{23}
 \end{aligned}$$

式中,  $e_{ij}, i, j \in \{1, 2, \dots, 6\}$  为  $s_1$  的多项式. 式 (23) 中再次包含  $s_2^2, s_3^2$  和  $s_2s_3$  项, 将式 (18) 再次代入式 (23), 整理可得:

$$\begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} = 0 \Leftrightarrow K_1(s_1) \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} = 0 \tag{24}$$

$K_1(s_1)$  为  $3 \times 3$  的矩阵, 其中每一项是  $s_1$  的多项式. 由式 (24) 可以看出, 要使式 (24) 存在非零解,  $K_1(s_1)$  的行列式  $|K_1(s_1)|$  必须为零. 将  $|K_1(s_1)| = 0$  进一步展开可得一个  $s_1$  的单变量多项式:

$$\begin{aligned}
 & U_8s_1^8 + U_7s_1^7 + U_6s_1^6 + U_5s_1^5 + U_4s_1^4 + \\
 & U_3s_1^3 + U_2s_1^2 + U_1s_1 + U_0 = 0 \tag{25}
 \end{aligned}$$

式中,  $U_i, i = 1, 2, \dots, 8$  为已知的系数. 采用特征值分解<sup>[8]</sup>方法,  $s_1$  可以很容易由式 (25) 求解得到. 将  $s_1$  代入式 (24), 求解其形成的线性方程组, 可得  $s_2$  和  $s_3$ .

实际上, 由于噪声的影响, 式 (16) 的关系不能完全满足, 因此通过以上方式求解得到的  $s_1, s_2$  和  $s_3$  还不是最优解. 为进一步提升  $s_1, s_2$  和  $s_3$  的求解精度, 将式 (16) 转换为不带约束的最优化形式:

$$\{s_1, s_2, s_3\} = \arg \min (\beta_1^T E_1^T E_1 \beta_1) \tag{26}$$

式中,  $E_1^T E_1$  为  $10 \times 10$  的已知对称矩阵. 由于  $s_1, s_2$  和  $s_3$  已经和全局最优值非常接近了, 因此通过单步 Gauss-Newton 法依照如下规则进行精定位:

$$[s_1, s_2, s_3]^T = [s_1, s_2, s_3]^T + \Delta s \tag{27}$$

式中,  $\Delta s = -[J^T J + \lambda I_{3 \times 3}]^{-1} J^T E_1 \beta_1$  表示增量,  $J$  为  $E_1$  的雅克比矩阵,  $\lambda$  为更新时的下降因子,  $I_{3 \times 3}$  为  $3 \times 3$  的单位矩阵. 得到最优的  $s_1, s_2$  和  $s_3$  后, 将其代回式 (12) 和式 (8), 则可得最终需要的  $R$  和  $t$ .

2) 对于形式 2. 类似于步骤 1), 将式 (13) 代入式 (10), 整理可得:

$$E_2 \beta_2 = 0 \tag{28}$$

式中,  $E_2$  为  $2n \times 6$  的对称矩阵, 且  $\beta_2 = [1, s_2, s_3, s_2^2, s_2s_3, s_3^2]^T$ . 同样,  $\beta_2$  中含有 2 个未知数, 且所含未知数的最高次数为 2. 因此式 (28) 中每一行都代表着一个二次曲线, 可以将式 (28) 的求解看作是二次曲线交点的求解. 式 (28) 同样可以表示为两部分:

$$\begin{aligned}
& [E_2(4), E_2(6), E_2(5)] \begin{bmatrix} s_2^2 \\ s_3^2 \\ s_2 s_3 \end{bmatrix} = \\
& - [E_2(2), E_2(3), E_2(1)] \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} \Leftrightarrow F_1 \begin{bmatrix} s_2^2 \\ s_3^2 \\ s_2 s_3 \end{bmatrix} = \\
& G_2 \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} \Leftrightarrow \begin{bmatrix} s_2^2 \\ s_3^2 \\ s_2 s_3 \end{bmatrix} = H_2 \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} \quad (29)
\end{aligned}$$

式中,  $H_2 = (F_2^T F_2)^{-1} F_2^T G_2$  是一个  $3 \times 3$  的系数矩阵. 将式 (29) 再次代入恒等式, 即式 (19) 可得:

$$\begin{aligned}
& m_{11}s_2^2 + m_{12}s_2s_3 + m_{13}s_2 + m_{14}s_3^2 + m_{15}s_3 = 0 \\
& m_{21}s_2^2 + m_{22}s_2s_3 + m_{23}s_2 + m_{24}s_3^2 + m_{25}s_3 = 0 \\
& m_{31}s_2^2 + m_{32}s_2s_3 + m_{33}s_2 + m_{34}s_3^2 + m_{35}s_3 + \\
& m_{36} = 0 \quad (30)
\end{aligned}$$

式中,  $m_{ij}$ ,  $i, j \in \{1, 2, \dots, 6\}$  为已知系数. 可以发现, 式 (30) 的形式和式 (23) 的形式相同, 但其中  $m_{ij}$  和  $e_{ij}$  所表示的具体意义不同. 式 (30) 中同样包含  $s_2^2$ 、 $s_3^2$  和  $s_2s_3$  项, 将式 (29) 再次代入式 (30) 可得:

$$\begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{bmatrix} \begin{bmatrix} s_2 \\ s_3 \\ 1 \end{bmatrix} = 0 \quad (31)$$

式中,  $q_{ij}$ ,  $i, j \in \{1, 2, 3\}$  为常系数, 可以由  $H_2$  和  $m_{ij}$  计算得到, 求解式 (31) 容易得到  $s_2$  和  $s_3$ . 同样, 为进一步提升  $s_2$  和  $s_3$  的求解精度, 首先将式 (28) 转换为无约束优化形式:

$$\{s_2, s_3\} = \arg \min (\beta_2^T E_2^T E_2 \beta_2) \quad (32)$$

式中,  $E_2^T E_2$  为  $6 \times 6$  的已知对称矩阵. 此时  $s_2$  和  $s_3$  可以按照如下式子进行精定位:

$$[s_2, s_3]^T = [s_2, s_3]^T + \Delta s \quad (33)$$

式中,  $\Delta s = -[J^T J + \lambda I_{2 \times 2}]^{-1} J^T E_2 \beta_2$ , 各变量的定义类似于式 (27). 得到最优的  $s_2$  和  $s_3$  以后, 将其代入式 (13) 和式 (8) 则可得需要求解的  $R$  和  $t$ .

3) 对于形式 3. 式 (14) 中仅含有一个变量  $s_3$ , 将式 (14) 代入式 (10), 最终得到一个  $s_3$  的多项式:

$$U_4 s_3^4 + U_3 s_3^3 + U_2 s_3^2 + U_1 s_3 + U_0 = 0 \quad (34)$$

式中,  $U_i$ ,  $i = 0, 1, \dots, 4$  为系数. 采用特征值分解的方法, 式 (34) 很容易求解, 一旦得到  $s_3$ , 将其代入式 (14) 和式 (8) 可得完整的  $R$  和  $t$ .

4) 对于形式 4. 式 (15) 中  $R$  的值已知, 将其直

接代入式 (8) 可得  $t$ .

由以上求解可以看出, 式 (25) 有 8 个解, 式 (31) 有 1 个解, 式 (34) 有 4 个解, 求解形式 4 有 1 个解. 因此, 求解以上形式最多得到 14 个解, 其分别对应着 14 组  $R$  和  $t$ . 将每组  $R$  和  $t$  分别代入式 (10), 选择其误差最小的那组  $R$  和  $t$  作为最终结果输出. 表 1 列出了文献中各非线性求解 PnL 问题算法解的个数. 从表 1 可以看出, 本文提出方法解的个数最少, 有利于提升算法的求解效率和求解可靠性.

表 1 解的个数对比

文献 [9]	文献 [12]	文献 [13]	文献 [14]	本文方法
15	27	15	60	14

## 5 实验结果

本节通过实验验证本文提出的 EPnL 算法性能, 并与现有主流算法进行对比. 对比算法主要分为线性算法和非线性算法两类, 其中线性算法有 DLT-Lines<sup>[7]</sup> 和 LPnL-Bar-LS<sup>[9]</sup> 算法; 非线性算法有 Lift<sup>[11]</sup>、AlgLS<sup>[12]</sup>、RPnL<sup>[13]</sup>、ASPnL<sup>[9]</sup> 和 SRPnL<sup>[14]</sup> 算法.

实验中的源代码可以从作者个人网站 <https://sites.google.com/view/ping-wang-homepage> 下载.

### 5.1 仿真实验

合成分辨率为  $640 \times 480$  像素, 焦距为 800 像素的虚拟相机. 在相机坐标系下产生空间 3D 直线: 在普通情况下, 空间直线随机分布在  $[-2, 2] \times [-2, 2] \times [4, 8]$  的范围内; 在共面情况下, 空间 3D 直线随机分布在  $[-2, 2] \times [-2, 2] \times [0, 0]$  内. 利用随机产生的旋转矩阵  $R_{true}$  和平移向量  $t_{true}$  将相机坐标系下的 3D 直线转换到世界坐标系. 利用合成的虚拟相机, 将相机坐标系下的 3D 直线投影到图像平面上, 并根据仿真实验参数的不同, 给投影直线增加不同等级的噪声  $\delta$ . 为了误差评估的一致性, 本文使用文献 [9–15] 的误差定义形式:

$$\begin{cases} e_{rot}(degrees) = \max_{k \in \{1, 2, 3\}} \arccos(r_{k, true}^T r_k) \times \frac{180}{\pi} \\ e_{trans} = \frac{\|t - t_{true}\|}{\|t_{true}\|} \times 100\% \end{cases} \quad (35)$$

式中,  $r_{k, true}$  和  $r_k$  分别表示  $R_{true}$  和  $R$  的第  $k$  列.

1) 直线数量对算法精度的影响. 首先设置仿真

噪声  $\delta = 5$ , 通过改变输入参考直线数量 4 ~ 20 来验证各 PnL 算法的求解精度, 结果如图 2 所示 (部分算法共面情况下幅度超出显示范围). 由图 2 可以看出, 线性的 DLT-Lines 和 LPnL-Bar-LS 方法在普通和共面情况下求解精度都不高, 且在直线数量少于 6 时无法正常求解. Lift 是一种非迭代的方法, 但由于其求解过程中使用了线性化的策略, 因此在普通情况 (共面情况下无法求解) 下求解精度不高. AlgLS 的中值求解精度很高, 但是其平均求解精度反而较低, 原因是 AlgLS 方法采用了 Cayley 参数表示旋转矩阵, 导致计算的时候容易出现奇异性, 影响算法的整体计算精度. RPnL 是一种次优化方法, 在普通情况下求解精度较高, 在共面情况下求解精度较差. ASPnL 算法作为 RPnL 算法的改进版本, 在普通情况下能取得很高的求解精度, 但在共面情况下平均求解精度较低. SRPnL 方法作为 RPnL 方法的另一种改进版本, 在普通和共面情况下都能取得较高的求解精度. 相比之下, 本文的 EPnL 采用了全局优化的方式求解, 并且在求解过程中分类考虑了旋转的情况, 避免了求解过程中的奇异性, 因此在普通和共面情况下都能保证最好的求解精度.

2) 噪声对算法精度的影响. 固定空间直线数量为 10, 改变噪声等级  $\delta$  为 1 ~ 15 来验证各 PnL 算

法的求解精度, 结果如图 3 所示. 可以看出, 随着噪声等级的增加, 各算法的求解精度都在下降, 且噪声等级和求解误差的增加近似符合线性关系. 由图 2 ~ 3 可以看出, 本文 PnL 算法在普通情况和共面情况下都能取得最好的求解精度.

3) 对比 P3L 算法. 本文算法虽然不是为求解 P3L 问题设计的, 但是可以处理 P3L 问题. 图 4 为本文算法和现有最新 P3L 算法<sup>[13, 19]</sup>的对比结果. 由图 4 可以看出, 本文 EPnL 算法不仅能够处理 P3L 问题, 而且还具有较高的求解精度.

4) 对比运行效率. 图 5 为各 PnL 算法的运行效率曲线图. 测试直线的数量从 4 到 2000 条, 足够覆盖大多数的实际应用场合. 测试时每种算法分别执行 500 次, 并统计其平均运行时间. 由图 5 可以看出, RPnL 和 ASPnL 算法初期运行效率高于 AlgLS 的方法, 但是随着直线数目增加, 其消耗的时间呈指数状增加, 效率反倒低于 AlgLS 方法. DLT-Lines 和 LPnL-Bar-LS 由于是线性求解的方法, 因此求解效率很高. SRPnL 方法相比于 RPnL 和 ASPnL 方法具有较高的求解效率, 但是随着直线数量的增加, 其时间消耗也急剧增加. 相比之下, 本文提出的 EPnL 方法具有很高的求解效率, 当直线数量超过 300 条的时候, 求解效率甚至高于线性的 DLT-Lines 的方法, 仅次于 LPnL-Bar-LS 方法.

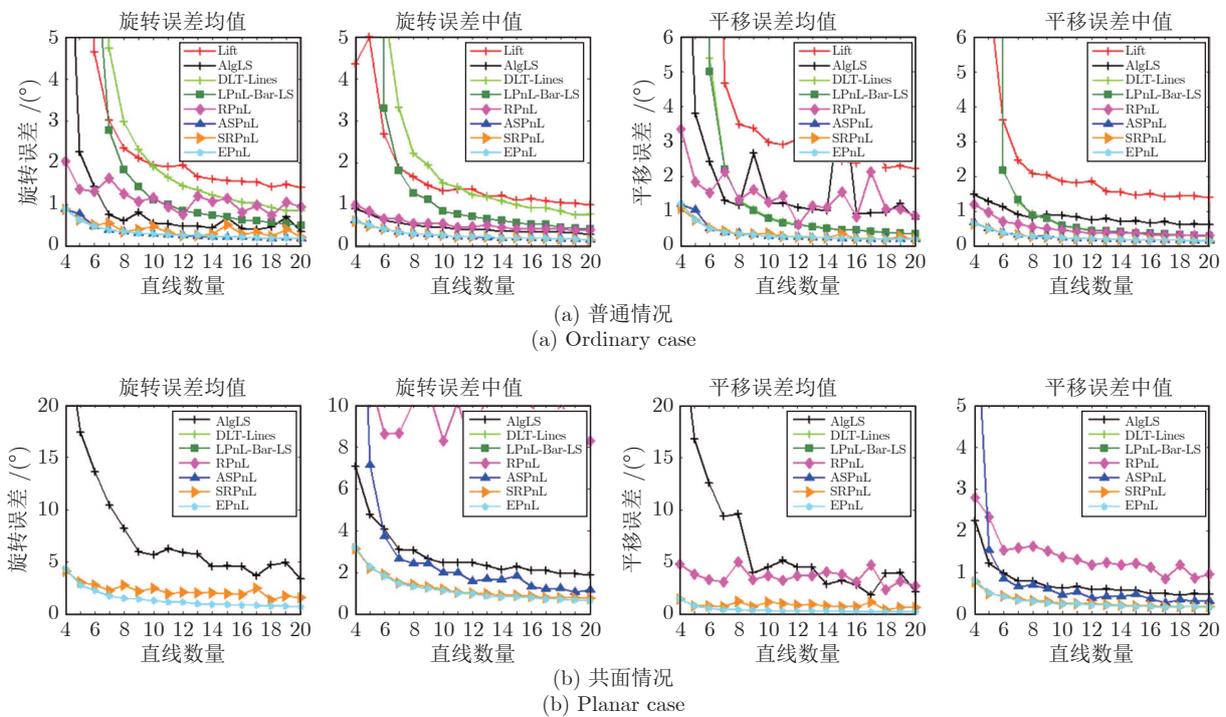


图 2 当直线数目变化时旋转和平移误差的均值和中值

Fig. 2 The mean and median of rotation and translation errors when the number of lines varies

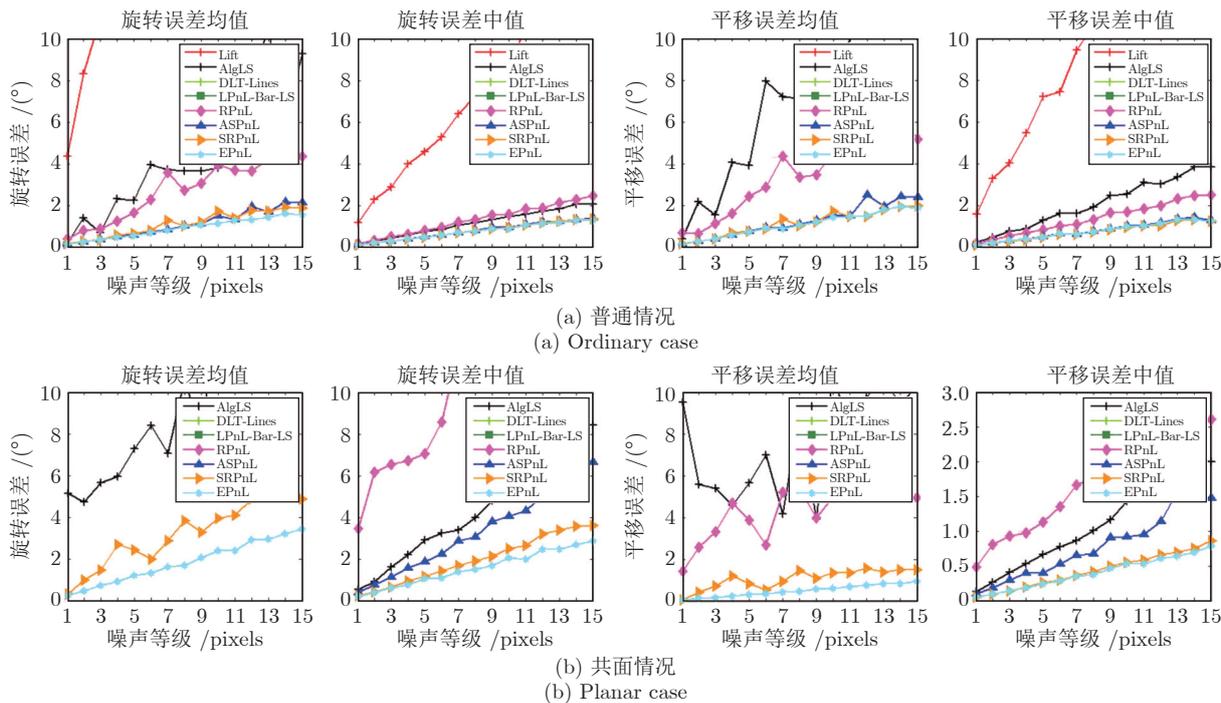


图 3 当噪声等级变化时旋转和平移误差的均值和中值

Fig.3 The mean and median of rotation and translation errors when the noise level varies

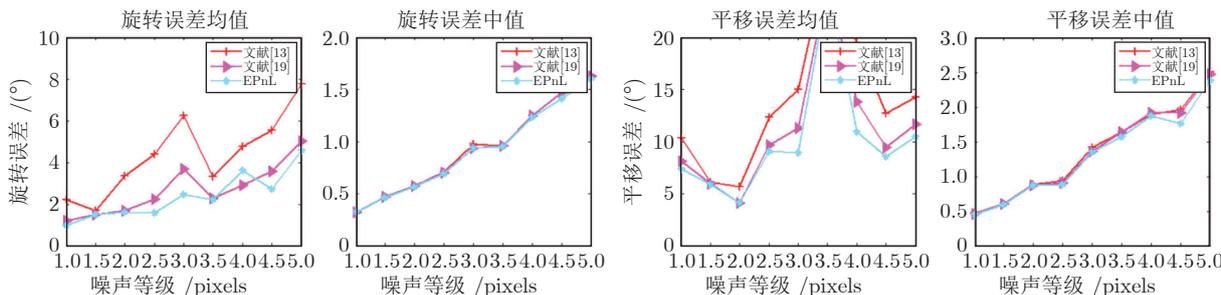
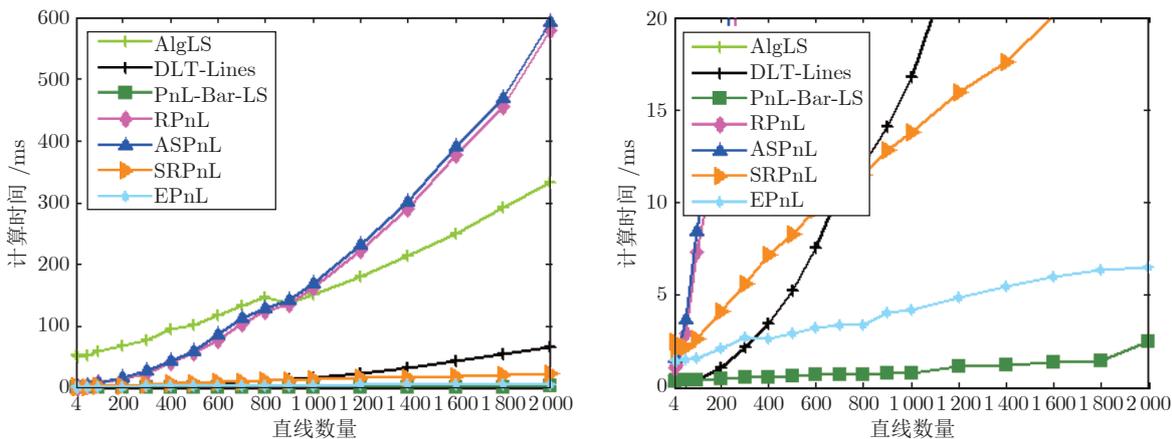


图 4 最小情况下 (n = 3) 旋转和平移误差的均值和中值

Fig.4 The mean and median of rotation and translation errors in the minimal case (n = 3)



(a) 对比算法的平均运行时间分布  
(a) The distribution of average running time for compared methods

(b) 图5 (a) 的局部放大  
(b) A zoomed-in version of Fig. 5 (a)

图 5 对比算法的计算效率

Fig.5 The computational efficiency of compared the methods

综合考虑 EPnL 算法的求解精度、求解效率和求解通用性, 其在所有对比方法中综合性能更优, 非常适合于实际项目应用.

## 5.2 实际实验

牛津大学视觉测量组 (Visual Geometry Group) 建立了以其团队命名的 (VGG) 图像数据集<sup>2</sup>. VGG 数据集中包含 7 组图像数据, 每组数据中包含若干张采集的建筑物图像, 图像中建筑物边缘直线的图像坐标及其对应世界坐标系中的位置、相机相对于世界坐标系之间的位姿关系都是已知的. 因此可以使用 VGG 数据集来测试各 PnL 算法的求解精度. 测试过程中各算法的求解误差定义如下:

$$\begin{cases} \Delta\theta = |Angle(R_{estimation}) - Angle(R_{true})| \\ \Delta T = |t_{estimation} - t_{true}| \end{cases} \quad (36)$$

式中,  $Angle(\cdot)$  表示将旋转矩阵转换为欧拉角,  $R_{estimation}$  和  $t_{estimation}$  表示各算法计算得到的旋转矩阵和平移向量,  $R_{true}$  和  $t_{true}$  为数据集中真实的旋转矩阵和平移向量. 分别利用各 PnL 算法按式 (36) 计算每组图像数据的误差平均值, 结果如表 2 所示, 每组中平均误差最小的值加粗表示. 由表 2 可以看出, 本文提出的 EPnL 算法在能够在 4 组测试集上同时获得最高的角度(旋转) 和位置(平

<sup>2</sup> <http://www.robots.ox.ac.uk/~vgg/>

移) 求解精度, 并且能够在 6 组测试集上获得最高的位置求解精度, 其整体求解精度最好. 为进一步验证 EPnL 算法的求解精度, 利用 EPnL 算法求解的位姿信息将世界坐标系下建筑物的边缘投影到图像上, 结果如图 6 所示, 其中青色直线为投影直线. 由图 6 可见, EPnL 算法能够准确的恢复位姿信息.

## 6 结束语

本文提出了一种精确且高效的 PnL 问题求解算法 (EPnL 算法). EPnL 首先通过矩阵变化的方式统一 PnL 问题的度量空间, 并将 PnL 求解问题转换为求二次曲面 (曲线) 交点的问题. 然后, 针对现有 Cayley 参数和四元数参数表示旋转时存在的问题, 本文提出了基于四元数的旋转分类表示方法, 该表示法在不损失旋转正交约束的前提下, 能够有效提升求解 PnL 问题的可靠性和效率. 最后, 针对现有迭代法和 Gröbner 基法求解问题效率不高且无法保障可靠性的问题. 本文提出利用二次项曲面方程组低次项参数化高次项的方法, 将二次项曲面方程组求交点问题转换为单变量多项式求解问题解决. 仿真和实际实验表明, 相比于现有的 PnL 算法, 本文算法在具有高求解精度的同时兼具高求解效率.

表 2 各算法在 VGG 数据集上的旋转和平移误差均值

Table 2 The mean of rotation and translation errors for each method on the VGG dataset

数据集名称	Model-House	Corridor	Merton-College-I	Merton-College-II	Merton-College-III	University-Library	Wadham-College	
图像数量	10	11	3	3	3	3	5	
AlgLS	$\Delta\theta[^\circ]$	0.4220	0.1983	3.6200	55.8037	3.7495	1.8838	60.0517
	$\Delta T[m]$	0.0384	0.0888	1.1504	14.1879	1.3683	0.9519	9.8801
DLT-Lines	$\Delta\theta[^\circ]$	0.8651	0.1104	0.0869	0.2117	0.1751	0.1736	0.1343
	$\Delta T[m]$	0.0834	0.0415	0.0274	0.1224	0.0625	0.0751	0.0809
LPnL-Bar-LS	$\Delta\theta[^\circ]$	0.4135	0.1178	<b>0.0241</b>	0.0261	0.0652	0.3642	0.1526
	$\Delta T[m]$	0.0403	0.0440	0.0099	0.0149	0.0233	0.1632	0.0909
RPnL	$\Delta\theta[^\circ]$	0.5521	0.3652	1.0870	0.3249	1.7528	2.9731	0.4200
	$\Delta T[m]$	0.0631	0.1150	0.3215	0.1660	0.9121	1.5613	0.1909
ASPnL	$\Delta\theta[^\circ]$	0.2265	<b>0.0911</b>	0.1141	0.1515	1.5584	3.6662	0.4227
	$\Delta T[m]$	0.0162	0.0298	0.0314	0.0600	0.5571	1.6683	0.1955
SRPnL	$\Delta\theta[^\circ]$	<b>0.2258</b>	158.952	0.4381	0.1151	36.4034	4.1848	0.0880
	$\Delta T[m]$	<b>0.0160</b>	17.557	0.1064	0.0495	3.9398	2.0632	0.0407
EPnL	$\Delta\theta[^\circ]$	0.2265	0.0969	0.0306	<b>0.0170</b>	<b>0.0504</b>	<b>0.0871</b>	<b>0.0808</b>
	$\Delta T[m]$	0.0162	<b>0.0252</b>	<b>0.0097</b>	<b>0.0123</b>	<b>0.0147</b>	<b>0.0343</b>	<b>0.0375</b>

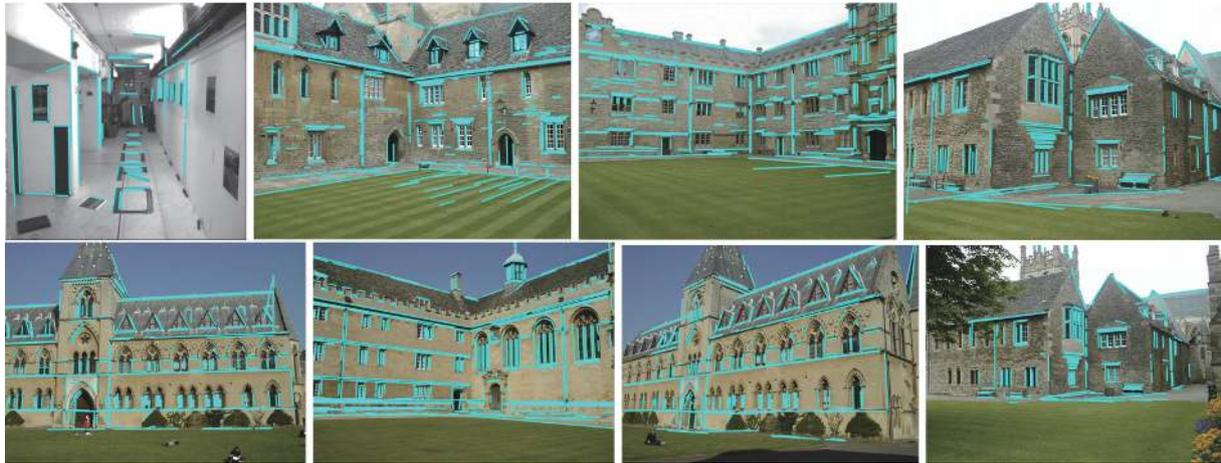
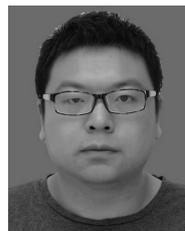


图 6 VGG 数据集中的图片

Fig.6 Images from the VGG dataset

## References

- 1 Yu Kun, Cong Ming-Yu, Duan Jia-Jia, Li Xiang-Yu. Monocular visual navigation method for capture point of docking ring. *Chinese Journal of Scientific Instrument*, 2018, **39**(12): 228–236 (于鲲, 从明煜, 段佳佳, 李向宇. 火箭对接环抓捕点单目视觉导航方法. *仪器仪表学报*, 2018, **39**(12): 228–236)
- 2 Ma Yan-Yang, Ye Zi-Hao, Liu Kun-Hua, Chen Long. Event-based visual localization and mapping algorithms: A survey. *Acta Automatica Sinica*, 2020, **46**(7): 1–11 (马艳阳, 叶梓豪, 刘坤华, 陈龙. 基于事件相机的定位与建图算法: 综述. *自动化学报*, 2020, **46**(7): 1–11)
- 3 Yu Yu-Feng, Zhao Hui-Jing. Off-road localization using monocular camera and nodding LiDAR. *Acta Automatica Sinica*, 2019, **45**(9): 1791–1798 (俞毓锋, 赵卉菁. 基于相机与摇摆激光雷达融合的非结构化环境定位. *自动化学报*, 2019, **45**(9): 1791–1798)
- 4 Hao Jie, Li Gao-Feng, Sun Lei, Lu Xiang, Zhang Sen, Liu Jing-Tai. Relative-pose-of-markers based calibration method for a deformable manipulator. *Acta Automatica Sinica*, 2018, **44**(8): 1413–1424 (郝洁, 李高峰, 孙雷, 卢翔, 张森, 刘景泰. 基于视觉标志间相对位姿的可变形臂标定方法. *自动化学报*, 2018, **44**(8): 1413–1424)
- 5 Alhaija H A, Mustikovela S K, Mescheder L, Geiger A, Rother C. Augmented reality meets computer vision: Efficient data generation for urban driving scenes. *International Journal of Computer Vision*, 2018, **126**(9): 961–972
- 6 Griva I, Nash S G, Sofer A. *Linear and Nonlinear Optimization*. Philadelphia: Siam, 2009. 355–478
- 7 Abdel-Aziz Y I, Karara H M, Hauck M. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. *Photogrammetric Engineering and Remote Sensing*, 2015, **81**(2): 103–107
- 8 Přibyl B, Zemčik P, Čadík M. Camera pose estimation from lines using Plücker coordinates. In: Proceedings of the 2015 British Machine Vision Conference, Swansea, United Kingdom: 2015. 45: 1–12
- 9 Xu C, Zhang L, Cheng L, Koch R. Pose estimation from line correspondences: A complete analysis and a series of solutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, **39**(6): 1209–1222
- 10 Přibyl B, Zemčik P, Čadík M. Absolute pose estimation from line correspondences using direct linear transformation. *Computer Vision and Image Understanding*, 2017, **161**: 130–144
- 11 Ansar K, Daniilidis K. Linear pose estimation from points or lines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, **25**(5): 578–589
- 12 Mirzaei F M, Roumeliotis S I. Globally optimal pose estimation from line correspondences. In: Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China: 2011. 5581–5588
- 13 Zhang L, Xu C, Lee K M, Koch R. Robust and efficient pose estimation from line correspondences. In: Proceedings of the 2012 Asian Conference on Computer Vision, Heidelberg, Berlin: 2012. 217–230
- 14 Wang P, Xu G L, Cheng Y H, Yu Q D. Camera pose estimation from lines: A fast, robust and general method. *Machine Vision and Applications*, 2019, **30**: 603–614
- 15 Yu, Q D, Xu G L, Zhang L M, Shi J C. A consistently fast and accurate algorithm for estimating camera pose from point correspondences. *Measurement*, 2020, **172**: 108914
- 16 Zheng Y Q, Kuang Y B, Sugimoto S, Astrom K, Okutomi M. Revisiting the PnP problem: A Fast, general and optimal solution. In: Proceedings of the 2013 International Conference on Computer Vision, Sydney, Australia: 2013. 2344–2351
- 17 Kukulova Z, Bujnak M, Pajdla T. Automatic generator of minimal problem solvers. In: Proceedings of the 2008 European Conference on Computer Vision, Heidelberg, Berlin: 2008. 302–315
- 18 Press W, Teukolsky S, Vetterling W, Flannery B. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge: Cambridge University Press, 2007.
- 19 Wang P, Xu G L, Cheng Y H. A novel algebraic solution to the perspective-three-line pose problem. *Computer Vision and Image Understanding*, 2020, **191**: 102711



王平 兰州理工大学电气工程与信息工程学院讲师。2019年获南京航空航天大学博士学位。主要研究方向为计算机视觉, 机器学习和信号处理。本文通信作者。

E-mail: pingwangsky@163.com

(WANG Ping Lecturer at the College of Electrical and Information Engineering, Lanzhou University of Technology. He received his Ph.D.

degree from Nanjing University of Aeronautics and Astronautics in 2019. His research interest covers computer vision, machine learning and signal processing. Corresponding author of this paper.)



**何卫隆** 兰州理工大学电气工程与信息工程学院硕士研究生. 主要研究方向为机器视觉和视觉测量.

E-mail: heweilongd@163.com

**(HE Wei-Long** Master student at the College of Electrical and Information Engineering, Lanzhou University of Technology. His research interest covers machine vision and vision measurement.)



**张爱华** 兰州理工大学电气工程与信息工程学院教授. 2005年获西安交通大学博士学位. 主要研究方向为检测技术和模式识别与智能系统.

E-mail: zhangaihua@lut.edu.cn

**(ZHANG Ai-Hua** Professor at the College of Electrical and Information Engineering, Lanzhou University of Technology. She received her Ph.D. degree from Xi'an Jiaotong University in 2005. Her research interest covers detec-

tion technology, pattern recognition and intelligent system.)



**姚鹏鹏** 中国香港理工大学纺织与制衣学系博士研究生. 主要研究方向为多光谱颜色测量, 相机校正和图像检索.

E-mail: p.p.yao@connect.polyu.hk

**(YAO Peng-Peng** Ph.D. candidate at the Institute of Textile and Clothing, Hong Kong Polytechnic University, China. His research interest covers multi-spectral color measurement, camera calibration and image retrieval.)



**徐贵力** 南京航空航天大学自动化学院教授. 2002年获江苏大学博士学位. 主要研究方向为光电检测, 计算机视觉和智能系统.

E-mail: guilixu2002@163.com

**(XU Gui-Li** Professor at the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics. He received his Ph.D. degree from Jiangsu University in 2002. His research interest covers photoelectric detection, computer vision and intelligent system.)