

引用格式:凌寒羽,衣晓,王培元,等.基于混合优化的自适应加速稳健PnP算法[J].电光与控制,2019,26(6):54-59. LING H Y, YI X, WANG P Y, et al. An adaptive accelerating robust PnP algorithm based on hybrid optimization[J]. Electronics Optics & Control, 2019, 26(6): 54-59.

基于混合优化的自适应加速稳健 PnP 算法

凌寒羽¹, 衣晓¹, 王培元¹, 杨卫国²

(1. 海军航空大学, 山东 烟台 264001; 2. 中国人民解放军91206部队, 山东 青岛 266001)

摘要: 根据目前摄像机位姿估计应用的实时性要求, 针对 RPnP 算法在利用最小二乘误差求解时无法得到唯一解的问题, 提出了一种改进自适应加速 RPnP 算法。在计算误差最小二乘时加入更多的限制条件, 确定了输出解的唯一性; 随后计算对应的摄像机外参数, 代替每个极小值重投影误差的计算和比较过程, 节省了大量位姿求解的时间; 最后自适应地将原算法和改进后的算法相结合, 使得输出结果最优化。实验证明, 该方法可以大大降低算法的时间复杂度, 并且运行时间受点数影响极小, 可以较好地应用于实时性要求高的场景。

关键词: 摄像机位姿估计; 自适应加速 RPnP; 解的唯一性; 时间复杂度

中图分类号: TP391.4 **文献标志码:** A **doi:**10.3969/j.issn.1671-637X.2019.06.011

An Adaptive Accelerating Robust PnP Algorithm Based on Hybrid Optimization

LING Han-yu¹, YI Xiao¹, WANG Pei-yuan¹, YANG Wei-guo²

(1. Naval Aviation University, Yantai 264001, China; 2. No. 91206 Unit of PLA, Qingdao 266001, China)

Abstract: In view of the real-time requirement of the application of camera position estimation, and of the defect of the RPnP algorithm that it cannot get a unique solution when using the least square error, an adaptive accelerating RPnP algorithm is proposed. When calculating the error, more constraints are added to the least square, and the uniqueness of the output solution is guaranteed. Afterwards, the corresponding camera parameters are calculated, which are used to replace the calculation and comparison process of each minimum reprojection error, and thus can save a lot of time. Finally, the original algorithm and the improved algorithm are adaptively combined to optimize the output results. The experiment shows that this method can greatly reduce the time complexity of the algorithm, and the running time is hardly influenced by the number of points, which is more adaptable to the scene with high real-time requirement.

Key words: camera position estimation; adaptive accelerating RPnP; the uniqueness of the output solution; time complexity

0 引言

n 点透视 (PnP) 问题是计算机视觉中的经典问题, 它根据图像点特征进行摄像机的位姿估计。一般情况下以摄像机内参数已知为前提, 获得空间 n 个参考点的世界坐标和对应的图像坐标信息, 计算摄像机的位置和姿态。根据特征点数目的不同可将该问题分为两类: 当 $3 \leq n \leq 5$ 时, 选用的特征点数目有限, 该问题抗干

扰性差, 对噪声极其敏感, 特征点位置分布要求严格, 并且存在多解情况; 当 $n \geq 6$ 时, PnP 即为经典 DLT 问题, 能够线性求解。

在早期的研究中得出, P3P^[1] 问题是 PnP 问题的一个最基本情形, 它是产生有效个参考点的最小子集。此后研究人员发现共面的 P4P 问题^[2] 能够产生唯一解, 并对其扩展应用。而近几年在精度和效率方面出现了很多十分优秀的算法: 文献[3] 在传统 PnP 基础上提出高精度快速位姿估计算法 (EPnP), 该方法定位准确, 计算速度快, 拥有 $O(n)$ 的复杂度, 将场景中所有空间参照点用已有的 4 个非共线虚拟控制点逐一表示, 计算全部空间参照点的摄像机坐标, 最终求得摄像机外参和位姿信息; 文献[4] 在此方法的基础上提出

收稿日期: 2018-06-25

修回日期: 2018-07-27

基金项目: 国防科技卓越青年人才基金; 泰山学者工程专项经费 (ts201712072)

作者简介: 凌寒羽 (1993 —), 女, 江苏泰州人, 硕士, 研究方向为计算机视觉。

加权 EPnP 算法,在牺牲时间复杂度的前提下提高了结果的精度;CHEN^[5]为了提高 EPnP 算法的精度和鲁棒性,提出基于虚拟控制点的迭代算法;而后 FERRAZ 和 BINEFA 提出 EPPnP^[6]和 CEPPnP^[7]等系列改进算法;OPnP^[8]位姿估计算法利用非单位四元数参数化旋转矩阵,根据目标函数的一阶最小化条件得到其多项式方程组,将问题转化为求取旋转参数的最优化。除此之外,还包括 UPnP^[9],PPnP^[10],MLPnP^[11],DLS^[12]和 ASPnP^[13]等诸多优秀算法。

文献[14]提出一种稳健的 PnP (RPnP) 的算法,其原理基础是三角约束,参考 P3P 原理将空间参考点分为 3 个子集,利用几何关系,采用多项式求解器代替矩阵的线性求解,避免了噪声影响下的各项解的不一致性,并且保持了 $O(n)$ 的复杂度。为防止迭代方法产生的局部极小值问题,该方法构造代价函数并求导,最终得到摄像机位姿信息的最优值。RPnP 结果精度高,冗余点少,特别是在 5 点以下时仍有很好的作用效果;并且可以灵活地应用于一般三维情况、平面情况和准奇异情况,是目前比较全面的一种 PnP 方法。但是,该方法使用误差最小二乘时无法得到唯一解,需要进行额外的验证。

鉴于此,在原算法的基础上,提出了一种自适应加速 RPnP 算法。计算误差最小二乘时加入更多的限制条件,将输出 x 的结果数量降为 1,随后计算对应的 R 和 t ,代替计算重投影误差的过程,节省了每个极小值的位姿求解过程的时间。最后自适应地将原算法与改进后的算法相结合,并进行仿真和实际图像实验。实验证明,该方法基本维持了原算法的计算精度,可以大大降低算法的时间复杂度,并且运行时间受点数影响极小,可以较好地应用于实时性要求高的场景。

1 自适应加速 RPnP 算法

1.1 RPnP 算法

RPnP 算法^[14]的思想是在给定的空间中新建一个正交的过渡坐标系 $O_a X_a Y_a Z_a$,如图 1 所示。

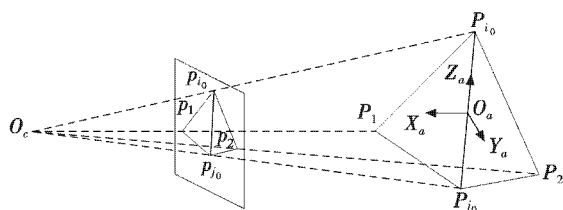


图 1 由空间参考点形成的新的正交坐标系

Fig.1 A new orthogonal coordinate system formed by a space reference point

将空间中 n 个参考点 $P_i (i=1, \dots, n)$ 的世界坐标系坐标值用新建坐标系来表示,故问题最终转化为新建坐标系和摄像机坐标系之间的透视转换矩阵求解。

二者的转换关系可由旋转轴 Z_a 、平移向量 $\overrightarrow{O_c O_a}$ 与绕旋转轴 Z_a 的旋转角 α 来描述,确定这三者后即可得到最终的摄像机位姿估计。

RPnP 算法中最关键的一步是选取合适的边 $\overline{P_i P_j} | i \neq j, i \in \{1, \dots, n\}, j \in \{1, \dots, n\}$ 作为旋转轴,不同的旋转轴对应不同的旋转角和平移向量,并且会直接影响最终的位姿求解精度。为尽量减小噪声影响,选取噪声影响最小的空间边集中投影长度最长的边 $\|p_i p_j\|$ 作为旋转轴。

首先需要求出端点 P_{i_0} 与 P_{j_0} 的深度值,确定 Z_a 轴方向,将所有空间参考点通过 $\{P_{i_0} P_{j_0} P_k | k \neq i_0, k \neq j_0\}$ 方式划分为 $n-2$ 个 3 点子集。根据三角形余弦定理,每个 3 点子集可写为 3 个 P2P 约束,即

$$\begin{cases} x_{i_0}^2 + x_{j_0}^2 - 2x_{i_0}x_{j_0} \cos \theta_{i_0 j_0} - d_{i_0 j_0}^2 = 0 \\ x_{i_0}^2 + x_k^2 - 2x_{i_0}x_k \cos \theta_{i_0 k} - d_{i_0 k}^2 = 0 \\ x_k^2 + x_{j_0}^2 - 2x_k x_{j_0} \cos \theta_{k j_0} - d_{k j_0}^2 = 0 \end{cases} \quad (1)$$

式中: x 代表空间参考点 P_i 的深度值; θ 代表两边之间的夹角; d 代表空间点之间的距离。3 个参考点的深度值为未知变量。可将上述方程组进行消元,化成一个一元四次多项式为

$$f(x) = ax^4 + bx^3 + cx^2 + dx + e = 0 \quad (2)$$

式中: $x = x_{i_0}^2$; a, b, c 和 d 为自变量 x 的系数; e 为任意常数。

由此建立一个 P3P 约束,如图 2 所示。

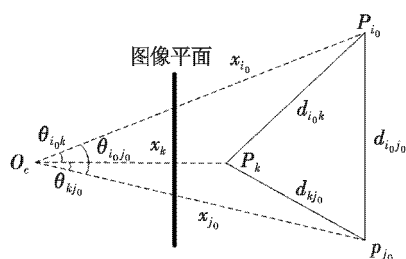


图 2 P3P 约束图示

Fig.2 P3P constraint diagram

$n-2$ 个 3 点子集可得到 $n-2$ 个一元四次多项式,由于每个 3 点子集均以 $\overline{P_{i_0} P_{j_0}}$ 为其中一条边,所以多项式均可使用相同的未知变量 x 来表示, x 为空间参考点 P_{i_0} 深度的平方值,即

$$\begin{cases} f_1(x) = a_1 x^4 + b_1 x^3 + c_1 x^2 + d_1 x + e_1 = 0 \\ f_2(x) = a_2 x^4 + b_2 x^3 + c_2 x^2 + d_2 x + e_2 = 0 \\ \vdots \\ f_{n-3}(x) = a_{n-3} x^4 + b_{n-3} x^3 + c_{n-3} x^2 + d_{n-3} x + e_{n-3} = 0 \\ f_{n-2}(x) = a_{n-2} x^4 + b_{n-2} x^3 + c_{n-2} x^2 + d_{n-2} x + e_{n-2} = 0 \end{cases} \quad (3)$$

考虑到噪声对每个 P3P 约束式的影响,线性方法会引起解的不一致性,本文采用最小二乘的误差处理方法求解多项式,解决方程组局部极小值的问题。设

代价方程为

$$F = \sum_{i=1}^{n-2} f_i^2(x) \quad (4)$$

易知 F 为一元八次方程,对其求导可解得原方程中的极小值为

$$F' = 2 \sum_{i=1}^{n-2} f_i(x) f_i'(x) = 0 \quad (5)$$

由一元七次方程解的性质可知,至多有 4 个根为 F 的极小值。对于每个极小值,可计算出对应的参考点深度,从而推算出其余端点的深度值,最终得到 $O_a X_a Y_a Z_a$ 坐标系下的旋转轴为

$$Z_a = \overline{P_{i_0} P_{j_0}} / \| P_{i_0} P_{j_0} \| \quad (6)$$

参考点的三维坐标投影到二维归一化平面的转换关系为

$$\lambda_i \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} r_1 & r_4 & r_7 \\ r_2 & r_5 & r_8 \\ r_3 & r_6 & r_9 \end{bmatrix} \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (7)$$

式中: $\begin{bmatrix} r_1 & r_4 & r_7 \\ r_2 & r_5 & r_8 \\ r_3 & r_6 & r_9 \end{bmatrix}$ 表示任意一个旋转矩阵; $\text{rot}(Z, \alpha) =$

$$\begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{表示绕 } Z \text{ 轴旋转一个角度 } \alpha。$$

式(7)可写为

$$\lambda_i \mu_i = R_1 R_2 X_i + t \quad (8)$$

将求得的 n 个局部最小值分别按照式(6)和式(7)求出旋转轴、旋转角和平移向量,从而进行摄像头的位姿估计,并按照

$$E = \frac{1}{n} \sum_{i=1}^n (\mu_i - \bar{\mu}_i)^2 \quad (9)$$

选择图像坐标 μ_i 与真实值 $\bar{\mu}_i$ 的平均重投影误差的最小解作为输出结果的最优解。

1.2 T-ARPNP 算法加速原理

在理想无噪声情况下,式(3)可以使用直接解析法求得最优的唯一解 x 。由于不同深度的参考点受噪声的影响不同,每一个 P3P 约束式 $f(x)$ 对应结果的真实值并不为 0,它可正可负且不完全相同。在 RPnP 的方法中,式(4)采用最小二乘误差方法解决此问题,将最终 x 的取值范围缩小至 4 个,而式(8)对每个 x 进行了外参数 R 和 t 的求解,并分别计算出各自的重投影误差进行比较,选出最优解。

计算重投影误差几乎是近年来 PnP 问题研究中必备的一步。在参考了众多国内外研究成果后发现,RPnP 与其他方法存在着很大的不同。绝大多数方法是利用

摄像机成像中的坐标转换方程进行线性求解,对系数方程组进行奇异值分解,求得 R 和 t ,再使用重投影误差的最小二乘法作为目标函数进行迭代运算,寻求最优的摄像机位姿,本文中重投影误差的目的是用来优化。

在 RPnP 算法中,式(4)是利用所有参考点和摄像机点的空间几何关系进行约束的误差最小二乘处理过程,是整个方法的核心计算内容。在 $n-2$ 个 $f(x)$ 多项式中,计算了 $n-2$ 次旋转轴空间距离的平方,而其他“边”仅计算一次,正确选取受噪声影响最小的旋转轴保证了整个算法的高精度,同时可以确定 4 个极小解的计算结果相对而言是比较好的。而重投影误差的计算仅用于从至多 4 个最优解里选出 1 个最合适的解,目的是用来选择。因此,尝试利用其他相对欠精确的计算方法代替重投影误差的选择作用,可大大提高算法速度。

将 RPnP 算法^[14]中使用的几何误差转换为代数误差并分别进行加权优化处理,在相同的环境下进行仿真实验,最终结果的变化幅度极小,如图 3 所示。

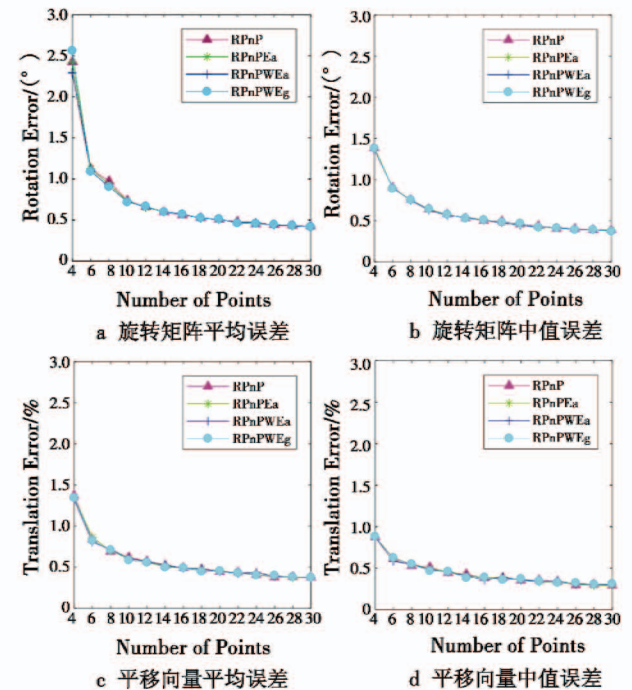


图 3 4 种误差计算方法的比较

Fig. 3 Comparison of four error calculation methods

由图 3 可知,算法总的计算结果主要取决于式(4)的最小二乘处理。考虑相片在式(10)镜头径向畸变影响下误差的分配不均,将误差按照图像上像点到主点坐标 $\mu_0 = (\mu_0, \nu_0)$ 距离的平方进行加权(RPnPWEg)计算,即

$$\begin{cases} \mu_0 = \mu(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ \nu_0 = \nu(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{cases} \quad (10)$$

$$W_{Eg} = \frac{1}{n} \sum_{i=1}^n \frac{(\mu_i - \bar{\mu}_i)^2}{\bar{\mu}_i - \mu_0} \quad (11)$$

将原算法中使用平均代数误差 (RPnPEa) 进行优化可得

$$E_a = \frac{1}{n} \sum_{i=1}^n (\mathbf{X}_i - \bar{\mathbf{X}}_i)^2 \quad (12)$$

根据文献[4]的思想,即:如果几何误差相同,空间点的深度越大,其代数误差越大;深度越小,其代数误差越小,按照深度值 x_i 对式(12)进行加权线性优化 (RPnPWEa) 可得

$$W_{Ea} = \frac{1}{n} \sum_{i=1}^n \left(\frac{\mathbf{X}_i - \bar{\mathbf{X}}_i}{x_i} \right)^2 \quad (13)$$

由图3可以看出,是误差计算的选择作用和选择范围有限,限制了加权方法的有效性。

式(5)最多可求得7个实根,其中极小值点 $x = \{x_i | i \in \{1, \dots, n\}, i \leq 4\}$ 最多有4个。A (accelerated) RPnP 算法的思想是在第一次计算误差最小二乘时加入更多的限制条件,将输出的结果 x 数量降为1,随后计算对应的 \mathbf{R} 和 \mathbf{t} ,无需再计算重投影误差,省略了其他计算结果计算摄像机外参数和重投影误差的比较过程。限制条件及步骤如下所述。

1) 剔除虚根。由于计算出的 x 的物理意义为参考点 P_i 深度的平方,故其一定为实数且大于0。给 x_i 设定一定的虚部误差阈值,保留其因为噪声引起的结果偏差,即

$$T = \left| \frac{\text{Im}(x_i)}{\text{Re}(x)_{\max}} \right| \quad (14)$$

若 $0.999 > T > 0.001$,则舍弃该解,其余结果保留。

2) 去除极大值点及驻点。导数为零表示导函数上点的斜率为零,它包含极大值点、极小值点和驻点,如图4所示。最简单的判断方法是对 F 求二阶导数,即

$$F'' = 2 \sum_{i=1}^{n-2} (f'^2(x_i) + f(x_i)f''(x_i)) = 0 \quad (15)$$

若 $F''(x_i) < 0$,则 x_i 为极小值点;若 $F''(x_i) > 0$,则 x_i 为极大值点。去除极大值点后在其余点中判断驻点的存在性,只需判断 $F'(x_i)$ 左右临近点 $F'(x_i \pm \Delta x)$ 的符号的一致性即可。

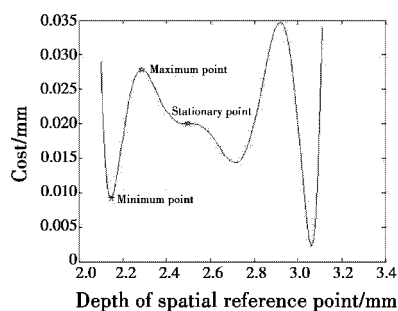


图4 任意一元八次多项式图像

Fig. 4 Image of an arbitrary one-variable, eight-degree polynomial

3) 将符合要求的 x_i 代入原方程 F ,计算每个点引起的误差值大小,并比较得出最终结果为

$$R = F(x_i)_{\min} \quad (16)$$

由式(16)即可得到 x_i 的唯一解,由此计算出新建坐标系的旋转轴,并根据式(7)计算出最终的摄像机位姿信息。

4) 自适应门限函数。

设 \mathbf{R} 为任一实空间, $x \in \mathbf{R}$ 为输入点, Φ 和 Ψ 分别为 \mathbf{R} 上的非线性映射。令 $\Phi(x)$ 为 RPnP 方法函数, $\Psi(x)$ 为 ARnP 方法函数, $N \in \mathbf{R}$ 为非负整数,则基于自适应门限的方法函数为

$$T(x) = \begin{cases} \Phi(x) & L(x) < N \\ \Psi(x) & L(x) > N \end{cases} \quad (17)$$

式中, $L(\cdot)$ 函数为测度函数。

点数是影响最终结果精度的主要原因,仅进行误差最小二乘能够减少很多不必要的计算,但缺少了最终重投影误差的二次验证,会产生个别情况下的不稳定性。鉴于此,提出了一种自适应加速稳健 PnP 算法 (T-ARNP),根据各算法平均点数冗余度的鲁棒性表现,以10点数为界建立门限函数,使得算法在不同条件下能够自主选择是否进行重投影误差的验证,结合高速度和高精度两大优势。

2 实验

2.1 仿真实验

采用与 RPnP 方法^[14]相同的仿真实验配置,选取分辨率为 640 像素 \times 480 像素的虚拟摄像机,其等效焦距为 800 像素。在摄像机坐标系下随机生成三维参考点的坐标,分布在 $[-2, 2] \times [-2, 2] \times [4, 8]$ 范围内,在每个图像点上加入不同水平的高斯噪声,并对每个水平的噪声分别做 1000 次独立实验。为验证算法性能,实验中将本文改进算法与以下几种较著名的方法进行时间和精度上的比较: RPnP 算法^[14]、DLT 算法^[15]、EPnP + GN 算法^[4]、EPPnP 算法^[6]、CEPPnP 算法^[7]、LHM 算法^[16]、OPnP 算法^[8]、DLS 算法^[12]、MLPnP 算法^[11] 和 PPnP 算法^[10]。

算法的平均运行时间 S 、旋转矩阵计算误差 E_R 与平移向量的计算误差 E_t 的算式为

$$S = \frac{1}{N} \sum_{i=1}^N s_i \quad (18)$$

$$E_R = \frac{\|\bar{\mathbf{q}} - \mathbf{q}\|}{\mathbf{q}} \quad (19)$$

$$E_t = \frac{\|\bar{\mathbf{t}} - \mathbf{t}\|}{\mathbf{t}} \quad (20)$$

式中: N 表示实验次数; s_i 表示第 i 次实验的运行时间; $\bar{\mathbf{q}}$ 和 \mathbf{q} 分别表示与旋转矩阵对应的归一化四元数的真

实值与计算值; \bar{t} 和 t 分别表示平移向量的真实值与计算值。

为了视觉效果直观性,图 5 列举了运行速度最快的几种优秀 PnP 算法,并进行对比。

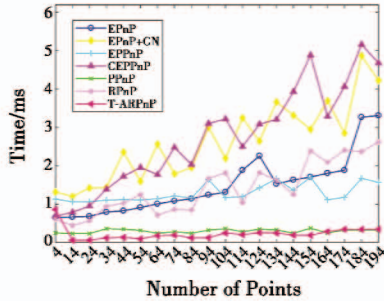


图 5 各算法运行时间对比

Fig. 5 Running time of each algorithm

由图 5 可知,大部分算法运算时间随着点数线性增长。RPnP 算法在非冗余条件下有很高的计算精度,当点数小于 10 时,选用 RPnP 算法进行计算,点数大于 10 时选用 ARPNP 算法进行计算。在点数大于 10 时,T-RPNP 算法可以显著地提高 RPnP 算法的运行速度,并能够保持极低的运算时间,受点数影响极小。与其他算法相比,该算法在时间上有明显优势。

图 6 所示为各算法精度对比。

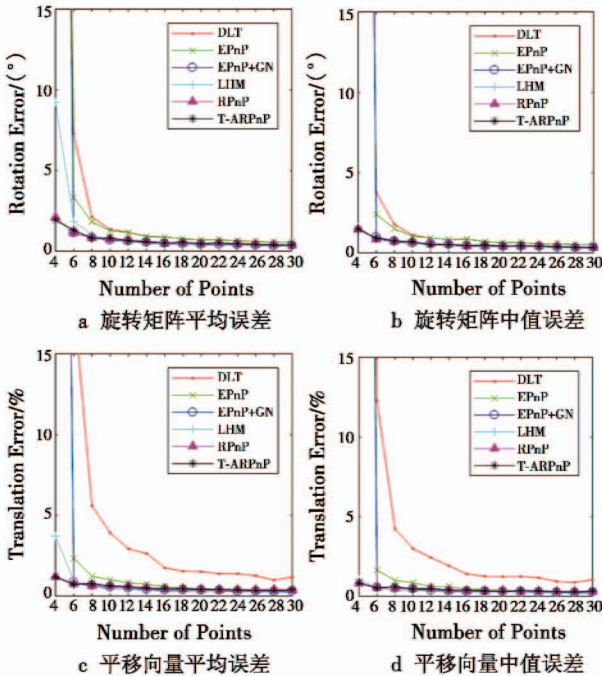


图 6 各算法精度对比

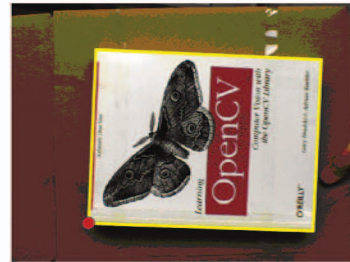
Fig. 6 Accuracy of each algorithm

由图 5 和图 6 可知,基于 T-ARPNP 的算法既保留了原算法中在非冗余情况下高精度的优势,又极大地缩短了算法的运行时间,且可以针对不同实际情况自适应地选择所用的方法,将结果保持在最优的状态。

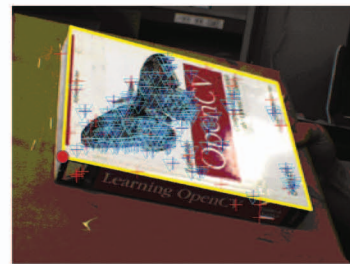
实验结果证明,该算法具有高精度、高时效性、适用性强、稳定性好等优点,在与其他优秀算法的对比中表现十分出色。

2.2 真实图像实验

为验证算法的有效性,进行真实图像实验。图 7 中给定像素大小为 640×480 的图像。使用焦距为 1529.60 像素、主点坐标为 $u_0 = 400.41$ 像素、 $v_0 = 303.68$ 像素的摄像机进行图像拍摄。



a 参考图像



b 输入图像

图 7 真实影像实验结果

Fig. 7 Real image experimental results

图 7a 是参考图像,黄色边框为已知 191 个参考点的模型轮廓。图 7b 是输入图像,将已知对应参考点按照本文方法计算出摄像机位姿,对应模型轮廓和其余内点坐标按照所求位姿信息反投影到输入图像中,计算结果精度越高,模型轮廓与图像边框的吻合度越高。图中模型轮廓与图中物品轮廓刚好重合,可看出改进方法的准确性较好。

用此真实图像对几种 PnP 方法进行测试,实验结果如表 1 所示,可以看出 T-ARPNP 的加速算法是真实有效的。

表 1 真实实验结果

Table 1 Real experimental results

算法	重投影误差/像素	算法耗时/ms
DLT	1.49	1.93
LHM	2.66	46.48
OPnP	1.36	13.67
RPnP	1.35	2.09
T-ARPNP	1.35	0.38

3 结束语

综合国内外优秀的 PnP 算法,具体分析了 RPnP

与其他算法在求解思想上的不同后,提出自适应加速稳健 PnP 算法。根据不同参考点作用能力的大小不同,对主要贡献点增加更多约束条件,将算法中的运算进行重新规整,去除特定情况下不必要的冗余运算步骤,显著提高了原算法的运算速度;最后,根据原算法中非冗余情况下运算结果精度高这一优势,自适应地将非冗余情况下的 RPnP 算法与冗余情况下的 ARPnP 算法进行结合,给出了最优策略。

在各种算法的比较中,本文算法在精度和时效性上均展示了良好的性能,在仿真实验和真实场景实验中表现出了良好的效果,具有较强的实用价值。

参考文献

- [1] KNEIP L, SCARAMUZZA D, SIEGWART R. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation[C]//Conference on Computer Vision and Pattern Recognition (CVPR), 2011. doi:10.1109/CVPR. 2011. 5995464.
- [2] 朱帆,于芳苏,吴易明,等. P4P 法摄像机姿态标定精度分析[J]. 光学学报,2018(11):1-16.
- [3] LEPETIT V, MORENO-NOGUER F, FUA P. EPnP: an accurate $O(n)$ solution to the PnP problem[J]. International Journal of Computer Vision, 2009, 81(2):155-166.
- [4] 杨森,吴福朝. 摄像机位姿的加权线性算法[J]. 软件学报,2011, 22(10):2476-2487.
- [5] CHEN P, HU G D. A simple algorithm for camera pose estimation[C]//Proceedings of International Conference on Mechatronics and Automation, 2012. doi:10.1109/ICMA. 2012. 6285681.
- [6] FERRAZ L, BINEFA X, MORENO-NOGUER F. Very fast solution to the PnP problem with algebraic outlier rejection[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014. doi:10.1109/CVPR. 2014. 71.
- [7] FERRAZ L, BINEFA X, MORENO-NOGUER F. Leveraging feature uncertainty in the PnP problem[C]//Proceedings of the British Machine Vision Conference (BMVC), 2014. doi:10.5244/c. 28. 83.
- [8] ZHENG Y Q, KUANG Y, SUGIMOTO S, et al. Revisiting the PnP problem: a fast, general and optimal solution [C]//Proceedings of the International Conference on Computer Vision (ICCV), 2013. doi:10.1109/ICCV. 2013. 291.
- [9] KNEIP L, LI H, SEO Y. UPnP: an optimal $O(n)$ solution to the absolute pose problem with universal applicability [C]//Proceedings of the European Conference on Computer Vision (ECCV), 2014:127-142.
- [10] GARRO V, CROSILLA F, FUSIELLO A. Solving the PnP problem with anisotropic orthogonal procrustes analysis [C]//The 2nd International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission, 2012. doi:10.1109/3DIMPVT. 2012. 40.
- [11] URBAN S, LEITLOFF J, HINZ S. MLPnP: a real-time maximum likelihood solution to the Perspective-N-Point problem [C]//ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2016. doi:10.5194/isprs-annals-III-3-131-2016.
- [12] HESCH J A, ROUMELIOTIS S I. A direct least-squares (DLS) method for PnP [C]//Proceedings of the International Conference on Computer Vision (ICCV), 2011. doi:10.1109/ICCV. 2011. 6126266.
- [13] ZHENG Y, SUGIMOTO S, OKUTOMI M. ASPnP: an accurate and scalable solution to the perspective-n-point problem [J]. IEICE Transactions on Information and Systems, 2013, 96(7):1525-1535.
- [14] LI S, XU C, XIE M. A robust $O(n)$ solution to the perspective-n-point problem [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 2012, 34(7):1444-1450.
- [15] ABDEL-AZIZ Y I, KARARA H M, HAUCK M. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry [J]. Photogrammetric Engineering & Remote Sensing, 2015, 81(2):103-107.
- [16] 刘延新,刘世良,丁全心,等. 基于数据融合的座舱头部姿态跟踪方法研究[J]. 电光与控制,2016, 23(8): 65-69.