

DOI:10.3969/j.issn.1673-4785.201111016

网络出版地址: http://www.cnki.net/kcms/detail/23.1538.TP.20120301.1652.001.html

自由角度显示系统下的运动向量算法

何雨兰, 顾人舒, 袁杰

(南京大学 电子科学与工程学院, 江苏 南京 210093)

摘要:为了解决三维运动矢量计算的问题,提出一种算法,这种算法是在静态三维重建的基础上用二维图片计算刚体运动的线性方法,利用QR分解求解线性方程的最小均方误差解,迭代来消除离群值,来得到运动矢量.这种算法不需要摄像机标定,实时性好,有利于视频的流畅显示,实验结果表明,此种算法具有一定的鲁棒性.

关键词:运动向量算法;运动估计;点匹配;线性方程组;QR分解

中图分类号:TP301.6 **文献标志码:**A **文章编号:**1673-4785(2012)02-0115-06

A 3-D motion vector algorithm in a free angle system

HE Yulan, GU Renshu, YUAN Jie

(School of Electronic Science and Engineering, Nanjing University, Nanjing 210093, China)

Abstract: In order to solve a 3-D motion vector calculation, a new algorithm was proposed. This algorithm is a linear method that calculates the rigid-body movement from two-dimensional views based on the static 3-D reconstruction. The QR decomposition was adopted to obtain the minimum mean square error of a linear equation set and an iterative process enables the method to eliminate outliers and thus obtain motion vector. This algorithm does not require camera calibration, and it is real-time, which enables smooth video. The experimental results show the algorithm has good robustness.

Keywords: motion vector algorithm; motion estimation; point correspondence; linear equation set; QR decomposition

在计算机视觉领域,从2幅或更多的图片序列中估计三维运动参数刚体已被广泛研究了很长一段时间,可以追踪到1970年左右,早期的研究证明,从平行投影模型来恢复运动,3个视角的图片是必要的;而从透视投影模型恢复运动,2个视角的图片就足够^[1].

1989年, W. Huang等^[2]详细论述了二维完全透视投影的运动估计,提出了经典的8点算法,即利用极几何(epipolar geometry)约束估计基本矩阵(矩阵 $E^{[3]}$)的特征点,然后从基本矩阵得到3-D运动参数.从那时起,为了克服它对噪声的敏感性,大量的改进方法被提出,包括美国研究人员的规范化二维数据的改进8点算法.除了基于特征点的方法外,研究人员也提出了各种基于线特征的方法^[4,5]、点线特征结合的方法、光学流程^[6,7]和使用多帧、迭代算

法等方法^[8-10].

然而,上述三维运动估计方法是从2幅(或2幅以上)图像没有任何先验知识的对象上来综合的.一般来说,如何应用极几何约束是一个关键的问题.至于基于点特征的方法,近期的工作集中在误差分析和控制^[1,11-12]或集中在更好地估计基本矩阵^[13-15]上,所有这一切都不可避免地运用了基础矩阵.最新的相关工作^[16-17]使用无标定摄像机的2帧图像来恢复结构和运动.

1 理论和算法

1.1 相机类型

仿射相机2-D坐标和3-D坐标服从以下规则:

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = P \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix}$$

式中: $\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ 0 & 0 & 0 \end{bmatrix}$, \mathbf{P} 是决定从 3-D 空间到

2-D 平面线性映射的矩阵。

应当指出的是仿射相机是真实相机的近似。只有当某一物体相对于自身的深度来说深度的变化可以忽略的时候,仿射相机是可以适用的。

1.2 刚体运动模型

在 3-D 空间中,2 个位置之间的刚体运动能够被分解为旋转和平移。在 3-D 空间中,一个 3×3 的矩阵 \mathbf{R} 能够描述刚体的旋转,一个 3×1 的矩阵 \mathbf{T} 能够描述翻译。

$$\mathbf{T} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}.$$

在图片 t_0 中的任一特征点的 3-D 坐标为 (X_1, Y_1, Z_1) , 这些点在 t_1 中的坐标为 (X'_1, Y'_1, Z'_1) . 则在 t_0 与 t_1 之间的运动可以表示为

$$\begin{bmatrix} X'_1 \\ Y'_1 \\ Z'_1 \\ 1 \end{bmatrix} = \mathbf{M} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix}.$$

运动矩阵则是:

$$\mathbf{M} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

值得注意的是,在一定的坐标系统下,刚体的任何一点都有相同的运动,即有相同的运动矩阵。任何

一对点都遵守这个等式。尽管旋转矩阵有 9 个元素,它是一个正交矩阵,受 6 个独立的约束:

$$\sum_{j=1}^3 r_{ij}^2 = 1, \sum_{k=1}^3 r_{ik} r_{jk} = 0, i, j = 1, 2, 3.$$

因而,矩阵中只有 3 个独立的元素。在各种不同的用 3 个元素表示 \mathbf{R} 的方法中,以下 2 种方法是经常使用的,如图 1 所示。

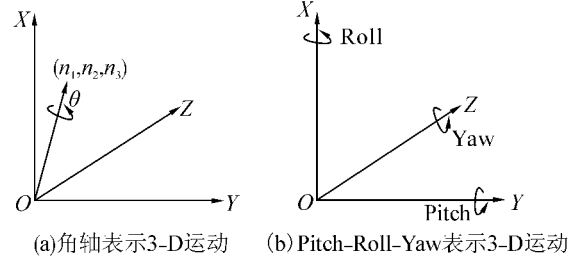


图1 表示 3-D 运动的 2 种常用方法

Fig.1 Two expressions of 3-D motion

1) 角轴表示法。

$$\begin{cases} r_{11} = \cos \theta + (1 - \cos \theta) n_1^2, \\ r_{12} = (1 - \cos \theta) n_1 n_2 - (\sin \theta) n_3, \\ r_{13} = (1 - \cos \theta) n_1 n_3 + (\sin \theta) n_2, \\ r_{21} = (1 - \cos \theta) n_1 n_2 + (\sin \theta) n_3, \\ r_{22} = \cos \theta + (1 - \cos \theta) n_2^2, \\ r_{23} = (1 - \cos \theta) n_2 n_3 - (\sin \theta) n_1, \\ r_{31} = (1 - \cos \theta) n_3 n_1 - (\sin \theta) n_2, \\ r_{32} = (1 - \cos \theta) n_3 n_2 - (\sin \theta) n_1, \\ r_{33} = \cos \theta + (1 - \cos \theta) n_3^2. \end{cases}$$

3-D 矢量从开始的 $(0, 0, 0)$ 到 (n_1, n_2, n_3) 表示刚旋转的坐标轴,它旋转的角度是 θ 。

2) Pitch-Roll-Yaw 表示法:

$$\mathbf{R} = \begin{bmatrix} \cos \theta_y \cos \theta_z & -\cos \theta_y \sin \theta_z & \sin \theta_y \\ \sin \theta_x \sin \theta_y \cos \theta_z + \cos \theta_x \sin \theta_z & -\sin \theta_x \sin \theta_y \sin \theta_z + \cos \theta_x \cos \theta_z & -\sin \theta_x \cos \theta_y \\ -\cos \theta_x \sin \theta_y \cos \theta_z + \sin \theta_x \sin \theta_z & \cos \theta_x \sin \theta_y \sin \theta_z + \sin \theta_x \cos \theta_z & \cos \theta_x \cos \theta_y \end{bmatrix}.$$

1.3 求解运动问题

1.3.1 把运动估计的问题转化为解方程组

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix}, \begin{bmatrix} x'_1 \\ y'_1 \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X'_1 \\ Y'_1 \\ Z'_1 \\ 1 \end{bmatrix}, \begin{bmatrix} X'_1 \\ Y'_1 \\ Z'_1 \\ 1 \end{bmatrix} = \mathbf{M} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix}.$$

$$\text{则有如下已知等式: } \begin{bmatrix} x'_1 \\ y'_1 \\ 1 \end{bmatrix} = \mathbf{PM} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix}.$$

每一对点则给出 2 个等式:

$$\begin{cases} x'_1 = p_{14} + p_{11}t_1 + p_{12}t_2 + p_{13}t_3 + \\ \quad X_1(p_{11}r_{11} + p_{12}r_{21} + p_{13}r_{31}) + \\ \quad Y_1(p_{11}r_{12} + p_{12}r_{22} + p_{13}r_{32}) + \\ \quad Z_1(p_{11}r_{13} + p_{12}r_{23} + p_{13}r_{33}), \\ y'_1 = p_{24} + p_{21}t_1 + p_{22}t_2 + p_{23}t_3 + \\ \quad X_1(p_{21}r_{11} + p_{22}r_{21} + p_{23}r_{31}) + \\ \quad Y_1(p_{21}r_{12} + p_{22}r_{22} + p_{23}r_{32}) + \\ \quad Z_1(p_{21}r_{13} + p_{22}r_{23} + p_{23}r_{33}). \end{cases} \quad (1)$$

对小角度近似来说,如运动足够小,且假设运用

Pitch-Roll-Yaw 表示法,则旋转矩阵可以简化为

$$\mathbf{R} = \begin{bmatrix} 1 & -n_3\theta & n_2\theta \\ n_3\theta & 1 & -n_1\theta \\ -n_2\theta & n_1\theta & 1 \end{bmatrix} = \begin{bmatrix} 1 & -\phi_3 & \phi_2 \\ \phi_3 & 1 & -\phi_1 \\ -\phi_2 & \phi_1 & 1 \end{bmatrix}.$$

式中: $(n_1\theta)^2 + (n_2\theta)^2 + (n_3\theta)^2 = \theta^2\phi_1^2 + \phi_2^2 + \phi_3^2 = \theta^2$.

重写等式组(1):

$$\begin{bmatrix} x'_1 - p_{14} - X_1p_{11} - Y_1p_{12} - Z_1p_{13} \\ y'_1 - p_{24} - X_1p_{21} - Y_1p_{22} - Z_1p_{23} \end{bmatrix} =$$

多个点对点的等式组如下:

$$\begin{bmatrix} x'_1 - p_{14} - X_1p_{11} - Y_1p_{12} - Z_1p_{13} \\ y'_1 - p_{24} - X_1p_{21} - Y_1p_{22} - Z_1p_{23} \\ \vdots \\ x'_n - p_{14} - X_np_{11} - Y_np_{12} - Z_np_{13} \\ y'_n - p_{24} - X_np_{21} - Y_np_{22} - Z_np_{23} \end{bmatrix} = \begin{bmatrix} Y_1p_{13} - Z_1p_{12} & Z_1p_{11} - X_1p_{13} & X_1p_{12} - Y_1p_{11} & p_{11} & p_{12} & p_{13} \\ Y_1p_{23} - Z_1p_{22} & Z_1p_{21} - X_1p_{23} & X_1p_{22} - Y_1p_{21} & p_{21} & p_{22} & p_{23} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_np_{13} - Z_np_{12} & Z_np_{11} - X_np_{13} & X_np_{12} - Y_np_{11} & p_{11} & p_{12} & p_{13} \\ Y_np_{23} - Z_np_{22} & Z_np_{21} - X_np_{23} & X_np_{22} - Y_np_{21} & p_{21} & p_{22} & p_{23} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ t_1 \\ t_2 \\ t_3 \end{bmatrix},$$

$$n = 1, 2, \dots.$$

上述等式的右侧第1个矩阵写作 \mathbf{A} , 它的秩不超过5. 这是单摄像机的限制, 因为摄像机不能给出光轴沿的深度信息. 观察矩阵 \mathbf{A} . 发现右边3列的奇数行数相同, 偶数行也如此. 对矩阵 \mathbf{A} 进行初等变换, 利用首行消除右边3列的 $2i-1$ 行, 用第2行来消除这些行的 $2i$ 行 ($i=2, 3, \dots, n$):

$$\begin{bmatrix} Y_1p_{13} - Z_1p_{12} & Z_1p_{11} - X_1p_{13} & X_1p_{12} - Y_1p_{11} & p_{11} & p_{12} & p_{13} \\ Y_1p_{23} - Z_1p_{22} & Z_1p_{21} - X_1p_{23} & X_1p_{22} - Y_1p_{21} & p_{21} & p_{22} & p_{23} \\ \vdots & \vdots & \vdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & 0 & 0 & 0 \\ Y_np_{13} - Z_np_{12} & Z_np_{11} - X_np_{13} & X_np_{12} - Y_np_{11} & 0 & 0 & 0 \\ Y_np_{23} - Z_np_{22} & Z_np_{21} - X_np_{23} & X_np_{22} - Y_np_{21} & 0 & 0 & 0 \end{bmatrix}. \quad (2)$$

对于得到的矩阵(2), 右边3列作为一个子矩阵 \mathbf{P} , 它的秩最多为2, 因为只有2行非零行; 左边3列作为一个子矩阵 \mathbf{Y} , 它的秩最多为3, 所以矩阵(2)的秩最多为5, $\mathbf{M} = [\mathbf{Y} \mathbf{P}]$, $\text{rank}(\mathbf{Y}) \leq 3$, $\text{rank}(\mathbf{P}) \leq 2$, $\text{rank}(\mathbf{M}) \leq 5$. 初等变换不改变矩阵的秩, 所以矩阵(2)的秩最高为5. 因而知道刚体的运动估计至少需要2台摄像机. 第2台摄像机的投影矩阵为

$$\mathbf{PP} = \begin{bmatrix} PP_{11} & PP_{12} & PP_{13} & PP_{14} \\ PP_{21} & PP_{22} & PP_{23} & PP_{24} \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

已知,

$$\begin{bmatrix} x'_1 - p_{14} - X_1p_{11} - Y_1p_{12} - Z_1p_{13} \\ y'_1 - p_{24} - X_1p_{21} - Y_1p_{22} - Z_1p_{23} \\ \vdots \\ x'_n - PP_{14} - X_nPP_{11} - Y_nPP_{12} - Z_nPP_{13} \\ y'_n - PP_{24} - X_nPP_{21} - Y_nPP_{22} - Z_nPP_{23} \end{bmatrix} =$$

$$\begin{bmatrix} Y_1p_{13} - Z_1p_{12} & Z_1p_{11} - X_1p_{13} & X_1p_{12} - Y_1p_{11} & p_{11} & p_{12} & p_{13} \\ Y_1p_{23} - Z_1p_{22} & Z_1p_{21} - X_1p_{23} & X_1p_{22} - Y_1p_{21} & p_{21} & p_{22} & p_{23} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_np_{13} - Z_np_{12} & Z_np_{11} - X_np_{13} & X_np_{12} - Y_np_{11} & PP_{11} & PP_{12} & PP_{13} \\ Y_np_{23} - Z_np_{22} & Z_np_{21} - X_np_{23} & X_np_{22} - Y_np_{21} & PP_{21} & PP_{22} & PP_{23} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ t_1 \\ t_2 \\ t_3 \end{bmatrix}, n = 1, 2, \dots.$$

左边的矩阵则表示为 \mathbf{b} , 右边的矩阵则表示为 \mathbf{Ax} . 因此, 运动估计问题转化为求解方程组 $\mathbf{b} = \mathbf{Ax}$.

使用从摄像机 1 中得到的 m 个点和对从摄像机 2 中得到的 k 个点, 条件是 $n = m + k$ ($m, n \geq 1$). 理论上, 从 2 台摄像机上得到的 3 个点足够解方程组. 当 $n \geq 3$ 时, 需要解方程组来导出结果. 在利用图片 t_0 和 t_1 进行图像匹配的 3-D 重建后, 可以获得大量的特征点对. 为了利用所有的信息, 首先使用所有的点进行运动估计. 并且为了提高算法的鲁棒性, 利用运动矩阵的初始解来进行二次投影, 在图片 t_1 中, 把结果与真实的 2-D 坐标进行比较, 消除大于阈值 10 的异常值 (实验结果表明误差一般小于 5, 如果错误大于 10, 它可能是一个异常值). 异常值的出现可能是误匹配的结果.

$$\begin{bmatrix} x'_1 - p_{14} \\ y'_1 - p_{24} \\ \vdots \\ x'_n - pp_{14} \\ y'_n - pp_{24} \end{bmatrix} = \begin{bmatrix} X_1 p_{11} & X_1 p_{12} & X_1 p_{13} & Y_1 p_{11} & Y_1 p_{12} & Y_1 p_{13} & Z_1 p_{11} & Z_1 p_{12} & Z_1 p_{13} & p_{11} & p_{12} & p_{13} \\ X_1 p_{21} & X_1 p_{22} & X_1 p_{23} & Y_1 p_{21} & Y_1 p_{22} & Y_1 p_{23} & Z_1 p_{21} & Z_1 p_{22} & Z_1 p_{23} & p_{21} & p_{22} & p_{23} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n pp_{11} & x_n pp_{12} & x_n pp_{13} & y_n pp_{11} & y_n pp_{12} & y_n pp_{13} & z_n pp_{11} & z_n pp_{12} & z_n pp_{13} & pp_{11} & pp_{12} & pp_{13} \\ x_n pp_{21} & x_n pp_{22} & x_n pp_{23} & y_n pp_{21} & y_n pp_{22} & y_n pp_{23} & z_n pp_{21} & z_n pp_{22} & z_n pp_{23} & pp_{21} & pp_{22} & pp_{23} \end{bmatrix} \mathbf{x} \quad (3)$$

使用从摄像机 1 中得到的 m 个点和对从摄像机 2 中得到的 k 个点, 条件是 $n = m + k$ ($m, n \geq 1$). 求解方程组的方法 (3) 与近似算法相同. 对于式 (3) 来说,

$$\mathbf{x} = [r_{11} \ r_{21} \ r_{31} \ r_{12} \ r_{22} \ r_{32} \ r_{13} \ r_{23} \ r_{33} \ t_1 \ t_2 \ t_3]^T.$$

对于通用算法, 为了保证已满秩, 至少需要 $4 + 4 = 8$ 对特征点.

1.3.3 求解线性方程组的最小均方误差解

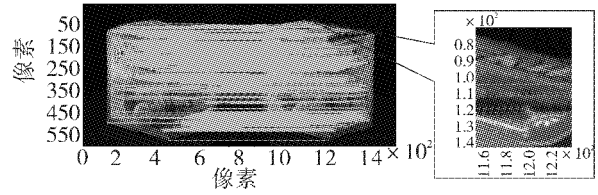
采用 QR (正规正交矩阵—上三角形矩阵) 分解法, 解法如下:

由于误差的存在, $\mathbf{Ax} = \mathbf{b} + \boldsymbol{\varepsilon}$, 问题是求解 \mathbf{x} 使范数 $\|\boldsymbol{\varepsilon}\|_2^2$ 最小.

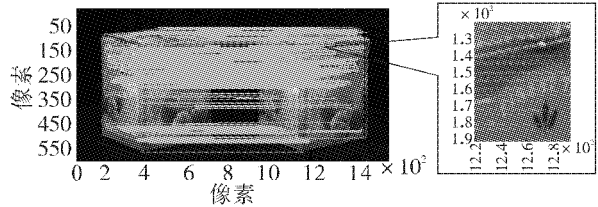
可以找到 \mathbf{Q} 使得 $\mathbf{QA} = \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}$. 其中 \mathbf{Q} 为正交矩阵, \mathbf{R} 为非奇异上三角阵.

$$\begin{aligned} \|\boldsymbol{\varepsilon}\|_2^2 &= \|\mathbf{Ax} - \mathbf{b}\|_2^2 = \|\mathbf{Q}(\mathbf{Ax} - \mathbf{b})\|_2^2 = \\ &= \|\mathbf{QAx} - \mathbf{Qb}\|_2^2 = \left\| \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} \mathbf{x} - \mathbf{Qb} \right\|_2^2. \end{aligned}$$

记 $\mathbf{Qb} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}$, 得 $\|\mathbf{Ax} - \mathbf{b}\|_2^2 = \left\| \begin{bmatrix} \mathbf{Rx} - \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \right\|_2^2$, 由于 $\begin{bmatrix} \mathbf{Rx} - \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}$ 是列向量, 故 $\|\mathbf{Ax} - \mathbf{b}\|_2^2 = \|\mathbf{Rx} - \mathbf{b}\|_2^2$



(a) 来自摄像机 1 的图片 t_0 (左) 和图片 t_1 (右) 的匹配点



(b) 来自摄像机 2 的图片 t_0 (左) 和图片 t_1 (右) 的匹配点

图 2 摄像机 1、2 的图片 t_0 和图片 t_1 的匹配点

Fig. 2 Matched point pairs in image t_0 and t_1 from cameras 1 and 2

1.3.2 通用算法

如果没有小角度近似, 那么可以写出如下方程组:

$$\begin{bmatrix} Y_1 p_{13} & Z_1 p_{11} & Z_2 p_{12} & Z_3 p_{13} & p_{11} & p_{12} & p_{13} \\ Y_1 p_{23} & Z_1 p_{21} & Z_2 p_{22} & Z_3 p_{23} & p_{21} & p_{22} & p_{23} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_n pp_{13} & Z_n pp_{11} & Z_n p_{12} & Z_n pp_{13} & pp_{11} & pp_{12} & pp_{13} \\ Y_n pp_{23} & Z_n pp_{21} & Z_n p_{22} & Z_n pp_{23} & pp_{21} & pp_{22} & pp_{23} \end{bmatrix} \mathbf{x} \quad (3)$$

+ $\|\mathbf{b}_2\|_2^2$.

注意到 $\|\mathbf{b}_2\|_2^2$ 是常数, 所以原问题 $\min \|\mathbf{Ax} - \mathbf{b}\|_2^2$ 转化为 $\min \|\mathbf{Rx} - \mathbf{b}_1\|_2^2$, 即 $\mathbf{Rx} - \mathbf{b}_1 = \mathbf{0}$. 故有

$$\mathbf{x} = \mathbf{R}^{-1} \mathbf{b}_1. \quad (4)$$

至此, \mathbf{x} 向量已经解出. 对于式 (2) 的近似算法, $\mathbf{x} = [\phi_1 \ \phi_2 \ \phi_3 \ t_1 \ t_2 \ t_3]^T$; 对于式 (3) 的非近似算法, $\mathbf{x} = [r_{11} \ r_{21} \ r_{31} \ r_{12} \ r_{22} \ r_{32} \ r_{13} \ r_{23} \ r_{33} \ t_1 \ t_2 \ t_3]^T$.

2 实验验证

利用 Matlab 对提出的算法进行验证, 以下是算法的流程.

- 1) 3-D 重建;
- 2) 匹配图片 t_0 和 t_1 , 获得 t_1 中特征点对的二维坐标;
- 3) 双目视觉: 3-D 重建和 2 台摄像机的图片匹配;
- 4) 初始化: 输入摄像机 1 的 m 对点和摄像机 2 的 k 对点;
- 5) 运动估计转化为解方程组, 每个点对给出 2 个方程;
- 6) 解方程组 $\mathbf{b} = \mathbf{Ax}$ (运用 QR 分解);
- 7) 用导出的 M 投影, 把 2-D 结果与真实的 2-D

坐标做比较:

If (所有点对误差在阈值内) then
 {消除误差超过阈值的点,
 Go to 5), repeat 5)、6)、7)}
 Else 给出最后结果.

2.1 2种算法分别绕 x 轴旋转 5°

已知图中的盒子有 5° 的旋转. 简单起见, 实验中的运动只有旋转. 也就是说, 平移矩阵中的元素全为零, 而旋转矩阵是重点. 首先, 检验近似算法, 然后检验严谨的通用算法. 根据上述算式, $\theta_x = \arctan(-r_{23}/r_{33})$.

实验中采用了图2所示的2幅图片. 在3-D空间中重建图片 t_0 , 然后匹配图片 t_0 和 t_1 , 在摄像机1中得到了318对——同样地, 从摄像机2中得到了246对点. 实验中利用了不同数量的点对, 输入的点对是从所有的点对中得到的双输入采样点. 计算过的 t_1 、 t_2 、 t_3 都近似为零, θ_x 显示在图3和图4上.

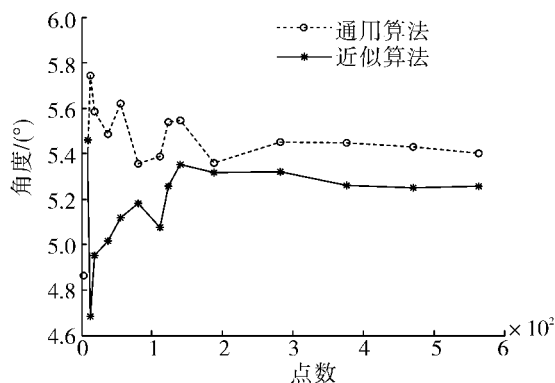


图3 用不同输入数量的点对的实验结果

Fig. 3 Experiments with different number of input point pairs

如图3所示, 当输入点对勉强够计算时, 计算结果深受输入数据的影响, 因而是不可靠的; 然而, 当有足够多的输入数据时, 2种算法都能提供稳定的结果. 当输入点对的数量大于200(2台摄像机), 对于相似算法, 起伏的结果不超过 $0.05/5.4 = 0.9\%$, 通用算法则在 $0.06/5.2559 = 1.2\%$ 以内.

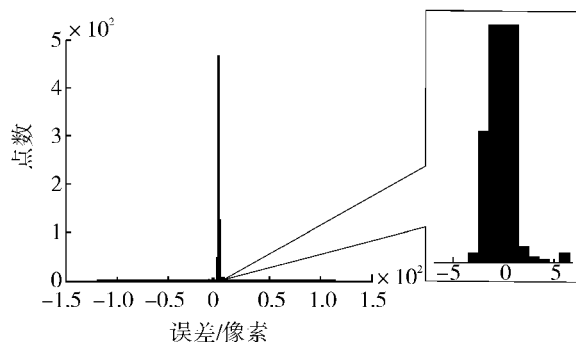


图4 二次投影误差分布(以 x 轴为例)

Fig. 4 Reprojection errors distribution (x axis as example)

这也证实了2次投影误差在图片中几乎是小于2个像素点, 即为 $318 + 246 = 564$ 对点. 此外, 这种算法能够消除离群值点, 这些点对二次投影误差大于阈值10, 因而保证了结果的稳定性和鲁棒性.

2.2 与前人方法的比较和讨论

为了进一步研究提出方法的精度, 从不同的视角进行了对比实验. 采用了Han-Kanade方法, 来重建一个立体和恢复运动, 然后采用所提出的计算方法, 使用相同的输入点计算运动. 如图5所示, 所采用的方法在精度方面与Han-Kanade方法具有可比性.

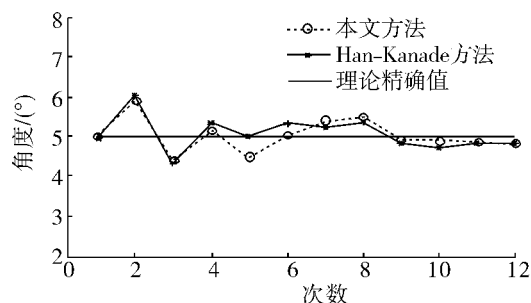


图5 与Hn-Kanade方法的比较(相同的输入点对)

Fig. 5 Comparison with Han-Kanade method (same input point pairs)

对3-D结构和从未校准摄像机上得到的运动进行彻底恢复是很耗时的. 在一台主频是2.4 GHz的电脑上, 对一次3帧重建要耗时5~10 min. 5个视角的全景重建要花费更多的时间, 不能盲目减少全景重建的视角数量(即摄像机数量), 因为这样会给大角度匹配带来困难, 进而给全景拼接带来困难.

而文中所提出的算法在同样的输入点前提下, 耗时少于1 s. 很明显, 在需要恢复长视频序列的条件下, 相比于同时恢复结构和运动的算法, 采用所提出的算法会节省很多时间.

2.3 简单的误差分析

误差包括: 1) 量化误差, 量化误差存在于整个过程, 包括本文中论述的运动计算问题; 2) 3-D重建误差, 包括多视角对应误差等, 作为运动矢量计算的输入, 3-D重建的误差将导致运动矢量的误差; 3) 图片 t_0 与 t_1 匹配上的误差, 有限的像素分辨率可能引入误差, 匹配算法也可能影响图片 t_1 在2-D坐标的精度.

此外, 摄像机的校准和计算精度对文中的算法几乎没有影响. 在文中的方法中, 三维重建是基于无标定摄像机, 运动计算也不涉及校准. 而且Matlab计算精度的超过40位小数, 所以截断误差不影响得到的结果.

3 结论

从实验结果中可以得到如下结论.

1) 提出算法的正确性和有效性得到了验证. 对于相似算法, 需要至少3个点对来进行运动估计, 但是算法的结果很大程度上依赖所取得的3个点对, 因而是很敏感、不稳定的.

2) 当有足够多的数据时, 所提出的算法是稳定的, 且具有鲁棒性, 它可以消除离群值. 最后, 提出的算法是很简洁的. 时间的消耗取决于点对的数量, 通用的算法只需要0.08, 近似算法需要大约0.1 s.

参考文献:

- [1] PAPADIMITRIOU T, STRINTZIS M G, ROUMELIOTIS M. Robust estimation of rigid body 3-D motion parameters based on point correspondences[J]. IEEE Trans on Circuits and Systems for Video Technology, 2000, 10(4): 541-549.
- [2] WENG J, HUANG T S, AHUJA N. Motion and structure from two perspective views: algorithms, error analysis, and error estimation[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1989, 11(5): 451-476.
- [3] HARTLEY R. In defense of the 8-point algorithm[C]//IEEE International Conference on Computer Vision. Cambridge, USA, 1995: 1064-1070.
- [4] EBRAHIMNEZHAD H, ROBUST H G. Motion from space curves and 3D reconstruction from multiviews using perpendicular double stereo rigs[J]. Image and Vision Computing, 2008, 26(10): 1397-1420.
- [5] MONTIEL J M M, TARDOHS J D, MONTANO L. Structure and motion from straight line segments[J]. Pattern Recognition, 2000, 33(8): 1295-1307.
- [6] STEINBACH E G, EISERT P, GIROD B. Model-based 3-D shape and motion estimation using sliding textures[C]//Proc Vision Modeling and Visualization Conference. Berlin, Germany, 2001: 375-382.
- [7] NEUMANN J, FERMULLER C, ALOIMONOS Y. Polydioptric camera design and 3D motion estimation[C]//Proc IEEE Conf Computer Vision and Pattern Recognition. Madison, USA, 2003: 294-301.
- [8] ROACH J W, AGGARWAL J K. Determining the movement of objects from a sequence of images[J]. IEEE Trans on Pattern Anal and Machine Intell, 1980, 7(4): 554-562.
- [9] OLIENSISA J. Multi-frame structure from motion algorithm under perspective projection[J]. International Journal of Computer Vision, 1999, 34(2): 163-192.
- [10] 付永庆, 宋宝森, 吴建芳. 边缘分类 SIFT 算法[J]. 哈尔滨工程大学学报, 2010, 31(5): 163-171.
FU Yongqing, SONG Baosen, WU Jianfang. Edge classification of the SIFT algorithm[J]. Journal of Harbin Engineering University, 2010, 31(5): 163-171.
- [11] SAND P, TELLER S. Particle video: long-range motion estimation using point trajectories[J]. International Journal of Computer Vision, 2008, 80(1): 72-91.
- [12] MATEI B, MEER P. A general method for errors-in-variables problems in computer vision[C]//Proc IEEE Conf Computer Vision and Pattern Recognition. Hilton Head, USA, 2000: 18-25.
- [13] FIROOZFAM P, NEGAHDARIPOUR S. Theoretical accuracy analysis of nocular vision systems for scene reconstruction, motion estimation, and positioning problems in computer vision[C]//Proc 2nd Int Symposium on 3D Data Processing, Visualization, and Transmission. Washington, DC, USA, 2004: 88-895.
- [14] SHEIKH Y, HAKEEM A, SHAH M. On the direct estimation of the fundamental matrix[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Minnesota, USA, 2007: 1-7.
- [15] CHEN P. Why not use the Levenberg-Marquardt method for fundamental matrix estimation[J]. IET Computer Vision, 2010, 4(4): 286-294.
- [16] WU H H P, CHANG S H. Fundamental matrix of planar catadioptric stereo systems[J]. IET Computer Vision, 2010, 4(2): 85-104.
- [17] SZELISKI R. Computer vision: algorithms and applications[M]. Berlin: Springer-Verlag, 2010: 343-374.

作者简介:



何雨兰, 女, 1988 年生, 硕士研究生, 主要研究方向为 3-D 重建算法.



顾人舒, 女, 1989 年生, 硕士研究生, 主要研究方向为 3-D 重建算法.



袁杰, 男, 1975 年生, 副教授, 主要研究方向为 3-D 重建算法、嵌入式系统.