

一种基于产生式系统的空间直线拟合方法

宋玲玲

滨州职业学院计算机信息工程系, 山东滨州 256600

摘要 求解设备转动副的高精度实际轴线是提高设备运行精度的重要措施之一, 当空间测量点密集时, 通常的最小二乘法很难得到正确的拟合直线。研究发现, 最小二乘法易造成 90° 的偏转, 即发生偏转后, 直线位于理想直线的垂直面内; 造成这种偏转的原因是最小二乘法距离函数不是一个单极值函数, 发生偏转的直线实际上是取得了其中一个最小值, 而非全局的极小值。为防止偏转, 提出了一种基于产生式系统的空间直线拟合方法, 该方法首先为拟合直线指定一个基本方向, 然后通过微分变化, 经有限次迭代便可求得理想的拟合直线。该方法能有效避免在使用最小二乘法求取拟合直线时产生的偏转问题, 通过在机器人关节轴线测量中的应用证明, 该方法在工程实际中有较高的应用价值。

关键词 空间直线拟合; 产生式系统; 最小二乘法

中图分类号 TP301.6

文献标识码 A

文章编号 1000-7857(2009)13-0083-05

A Method for Spatial Straight Line Fitting Based on Production System

SONG Lingling

Computer Information Engineering Department, Binzhou Vocational College, Binzhou 256600, Shandong Province, China

Abstract Solving actual high-precision rotation axes of revolute pairs is one of significant initiatives to improve running-precision of a set of equipment. As a classical approach, least square method is difficult to obtain appropriate fitting straight line when measured spatial points are intensive. It was found, through the study on the problem, that least square method causes a 90° deflection easily, that is, straight line after deflection locates in a vertical plane of the ideal straight line, and that the reason resulted in deflection is that distance function of least square method is not a single extreme value function, the occurrence of deflection reaches only one of the minimums rather than the extreme minimum in fact. In order to prevent the deflection, a production system based-on fitting method was proposed for spatial straight line fitting. First, the proposed method specifies a basic direction for the fitting line, and then sets the differential change, at last, through limited iteration, the ideal fitting straight line can be found. The proposed method can effectively avoid the deflection of adopting least square method of straight line fitting, and has been proved with a high value of application in practice through the real joint axes measurement of a robot.

Keywords spatial straight line fitting; production system; least square method

0 引言

一般情况下, 设备的理论回转轴线并非实际的运动副轴线, 实际上, 设备是围绕其平均(实际)回转轴线做一种近似圆周摆动, 如车削机床的主轴回转轴线、关节机器人的各运动关节轴线。在精度要求较低时, 可直接采用理论回转轴线进行计算。随着社会需求的不断提高, 对加工设备的精度要求也越来越高, 采用理论回转轴线作为设备的计算和应用依据, 已不能满足工业实际的要求。因此, 采用精度更高的平均轴线成为提高设备运行精度的重要措施之一。如机器人关节参数精确测定, 除采用一些多点测量法外, 还需测量特定姿态下各关节实际轴线的具体空间位置, 这样才能得到精确的关节参数。

然而, 对于设备回转运动平均轴线的测量, 目前的测量手段只能得到一些离散的空间点集, 不能直接得到一条理想的直线。空间直线拟合就是解决这一问题的重要方法^[1-4], 通常采用最小二乘法。但当空间点密集时, 最小二乘法很难得到正确的拟合直线。针对这种情况, 本文提出了一种基于产生式的空间直线拟合方法, 并给出了具体的 Java 程序实现。

1 最小二乘法空间直线拟合

如图 1 所示, 最小二乘法空间直线拟合就是求解一近似直线, 使得点集中的所有空间点到该直线距离的平方和最小。

收稿日期: 2009-03-10

作者简介: 宋玲玲, 讲师, 研究方向为计算机应用, 电子信箱: sdbzsling@126.com

设所求空间直线的参数方程为^[5]

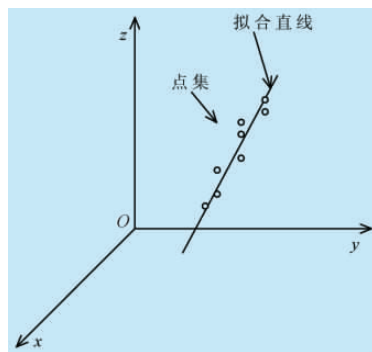


图 1 空间直线拟合

Fig. 1 Spatial straight line fitting

$$\begin{cases} x=mt+x_0 \\ y=nt+y_0 \\ z=pt+z_0 \end{cases} \quad (1)$$

其中, m, n, p 为直线的方向余弦, x_0, y_0, z_0 为直线上一点。

设点集为 $\{p_i\}$, 则点 p_i 到该直线的距离为

$$d_i^2 = \left[(x_i - x_0) + m \frac{m(x_0 - x_i) + n(y_0 - y_i) + p(z_0 - z_i)}{m^2 + n^2 + p^2} \right]^2 + \left[(y_i - y_0) + n \frac{m(x_0 - x_i) + n(y_0 - y_i) + p(z_0 - z_i)}{m^2 + n^2 + p^2} \right]^2 + \left[(z_i - z_0) + p \frac{m(x_0 - x_i) + n(y_0 - y_i) + p(z_0 - z_i)}{m^2 + n^2 + p^2} \right]^2 \quad (2)$$

则点到直线的距离平方和为

$$Q = \sum_{i=0}^k d_i^2 \quad (3)$$

根据最小二乘法原理, 有

$$\frac{\partial Q}{\partial m} = \frac{\partial Q}{\partial n} = \frac{\partial Q}{\partial p} = \frac{\partial Q}{\partial x_0} = \frac{\partial Q}{\partial y_0} = \frac{\partial Q}{\partial z_0} = 0 \quad (4)$$

其中,

$$\frac{\partial Q}{\partial m} = A m^2 + B m + C = 0$$

$$\frac{\partial Q}{\partial n} = D n^2 + E n + F = 0$$

$$\frac{\partial Q}{\partial p} = G p^2 + H p + I = 0$$

$$\frac{\partial Q}{\partial x_0} = J x_0 + K = 0$$

$$\frac{\partial Q}{\partial y_0} = L y_0 + M = 0$$

$$\frac{\partial Q}{\partial z_0} = N z_0 + Q = 0$$

其中, $A, B, \dots, Q$ 由 m, n, p, x_0, y_0, z_0 和点集 $\{p_i\}$ 确定, 由于算式复杂, 从略。

联立上述 6 个偏导方程就可以解出空间拟合直线的 6 个参数, 于是就求出了空间直线。此方法是求解空间点直线拟合的最一般的方法, 但此方法存在一个缺陷, 就是当点集中的点彼此之间距离较近时容易产生直线方向的偏转, 一般

与所求理想直线的夹角为 90° , 如图 2 所示, 虚线为最小二乘法所求出的拟合直线, 而实线为所求理想直线。

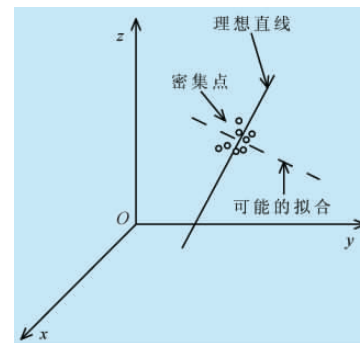


图 2 错误的拟合直线

Fig. 2 Failure fitting straight line

2 基于产生式的直线拟合算法

为了解决最小二乘法容易产生拟合直线偏转的问题, 本文提出了一种采用产生式理论^[6-7]进行空间直线拟合的方法, 该方法既能达到二乘法的拟合效果又能弥补二乘法的不足。因本文主要从工程应用的角度进行算法开发, 所以纯数学上的论证从略。

1) 算法原理。通过对偏转现象的分析得知, 这种偏转一般为 90° 的偏转, 即发生偏转后的直线位于理想直线的垂直面内, 与理想直线相交或者不相交。同时, 通过对最小二乘法公式的研究发现, 距离函数不是一个单极值函数, 存在一个或两个以上最小值。发生偏转的直线实际上是取得了其中的一个最小值, 而非全局的极小值。为防止偏转, 给拟合算法指定一个基本的直线方向是可行的、最简单的一种做法。通常工程领域中测量点偏离回转轴线的误差应当远小于在轴线方向上测量点之间的距离。因此, 拟合直线的初始方向可选择为点集中距离最大两点所确定的方向, 起始点可任选其一。为防止拟合直线方向突变, 对初始拟合直线方向和起始点进行微小变化, 即对 m, n, p, x_0, y_0, z_0 进行微小调节, 使得点集中所有点到该直线的距离之和最小, 再通过多次迭代便可求得精度较高的拟合直线。实际上, 该产生式方法是最小二乘法的一种变形。

2) 算法实现。该算法的求解目标是求得一条空间直线, 使得点集内所有点到该直线距离的平方和最小。易知产生式方法的数据库为所有空间直线的集合, 算法的判定方式是判断两次迭代之间距离平方和的差别大小。类似爬山法求极值, 该算法的控制策略为不可撤回式。算法涉及 6 个参数 (m, n, p, x_0, y_0, z_0) 的计算, 限于篇幅, 对于产生式的规则仅以 m 求解为例介绍如下。

1) IF m 增大后距离平方和减小, m 减小后距离平方和增大, THEN m 相应增大。

2) IF m 增大后距离平方和增大, m 减小后距离平方和减小, THEN m 相应减小。

3) IF m 增大后和减小后距离平方和都减小, THEN m 变



化去减小大的一方。

4) IF m 增大和减小后距离的平和都增大, THEN m 保持不变。

算法伪代码描述如下。

INPUT Point set $\{p\}$; Delta δ ; Absolute deviation d

OUTPUT Line parameters m, n, p, x_0, y_0, z_0

VARIABLE $sum, tmp_d, tmp_d0, tmp_d1$

Step 1 Set $m_i=p[j].x-p[i].x; n_i=p[j].y-p[i].y; p_i=p[j].z-p[i].z$. m, n, p 取 $\max\{m_i^2+n_i^2+p_i^2\}$

Set $x_0=p[i].x; y_0=p[i].y; z_0=p[i].z$

Set $tmp_d=0; sum=0$

Step 2 Set $tmp_d=\sum_i^k d_i^2$ % d_i^2 由式(2)确定%

Step 3 Do Step 4-21 While $ABS(sum-tmp_d)>d$

Step 4 Set $m=m+\delta, tmp_d0=\sum_i^k d_i^2$
% d_i^2 由式(2)确定%

Step 5 Set $m=m-2\delta, tmp_d1=\sum_i^k d_i^2$

Step 6 If $tmp_d0>tmp_d$ And $tmp_d1>tmp_d$
 $m=m+\delta$

Else if $tmp_d0<tmp_d$ And $tmp_d1<tmp_d$
If $tmp_d0<tmp_d1$
 $m=m+2\delta$

$tmp_d=tmp_d0$

Else

$tmp_d=tmp_d1$

End if

Else if $tmp_d0<tmp_d$

$m=m+2\delta$

$tmp_d=tmp_d0$

Else

$tmp_d=tmp_d1$

End if

End if

End if

Step 7 ...

Step 8 ...

$\vdots$

Step 21 ...

Step 22 OUTPUT(m, n, p, x_0, y_0, z_0)

STOP

3 与最小二乘法比较

根据以上提出的基于产生式空间直线拟合算法,在 Windows XP 系统上采用 Java 语言编写了具体的计算程序,得到两组计算数据,并对该方法和最小二乘法进行了比较,如表 1 和表 2 所示。

表 1 一般直线拟合的两种方法比较

Table 1 Comparison between two methods for general straight line fitting

点	x	y	z			
1	0.004 004 771 561 840	0.007 421 122 650 927	0.005 374 824 958 766			
2	0.012 014 314 685 519	0.022 263 367 952 782	0.016 124 474 876 299			
3	0.020 023 857 809 198	0.037 105 613 254 636	0.026 874 124 793 831			
4	0.028 033 400 932 877	0.051 947 858 556 490	0.037 623 774 711 364			
5	0.036 042 944 056 557	0.066 790 103 858 344	0.048 373 424 628 896			
6	0.044 052 487 180 236	0.081 632 349 160 199	0.059 123 074 546 429			
7	0.052 062 030 303 915	0.096 474 594 462 053	0.069 872 724 463 961			
8	0.060 071 573 427 594	0.111 316 839 763 907	0.080 622 374 381 494			
9	0.068 081 116 551 274	0.126 159 085 065 762	0.091 372 024 299 026			
10	0.076 090 659 674 953	0.141 001 330 367 616	0.102 121 674 216 559			
11	0.084 100 202 798 632	0.155 843 575 669 470	0.112 871 324 134 091			
12	0.092 109 745 922 311	0.170 685 820 971 324	0.123 620 974 051 624			
13	0.100 119 289 045 991	0.185 528 066 273 179	0.134 370 623 969 156			
14	0.108 128 832 169 670	0.200 370 311 575 033	0.145 120 273 886 689			
15	0.116 138 375 293 349	0.215 212 556 876 887	0.155 869 923 804 221			
16	0.124 147 918 417 028	0.230 054 802 178 742	0.166 619 573 721 754			
17	0.132 157 461 540 707	0.244 897 047 480 596	0.177 369 223 639 286			
18	0.140 167 004 664 387	0.259 739 292 782 450	0.188 118 873 556 819			
19	0.148 176 547 788 066	0.274 581 538 084 305	0.198 868 523 474 351			
20	0.156 186 090 911 745	0.289 423 783 386 159	0.209 618 173 391 884			
21	0.164 195 634 035 424	0.304 266 028 688 013	0.220 367 823 309 416			
22	0.172 205 177 159 104	0.319 108 273 989 867	0.231 117 473 226 949			
23	0.180 214 720 282 783	0.333 950 519 291 722	0.241 867 123 144 481			
最小二乘法	m	0.400 477 156 183 9	n	0.742 112 265 092 7	p	0.537 482 495 876 6
基于产生式法	m	0.400 477 160 520 5	n	0.742 112 262 245 1	p	0.537 482 496 577 0

Table 2 Comparison between two methods for straight line fitting on spatial intensive points

点	x		y		z	
1	0.000 000 792		0.000 000 576		0.650 000 846	
2	0.000 000 616		0.000 000 448		0.650 000 658	
3	0.000 000 440		0.000 000 320		0.650 000 470	
4	0.000 000 264		0.000 000 192		0.650 000 282	
5	0.000 000 088		0.000 000 064		0.650 000 094	
6	-0.000 000 088		-0.000 000 064		0.649 999 906	
7	-0.000 000 264		-0.000 000 192		0.649 999 718	
8	-0.000 000 440		-0.000 000 320		0.649 999 530	
9	-0.000 000 616		-0.000 000 448		0.649 999 342	
10	-0.000 000 792		-0.000 000 576		0.649 999 154	
11	-0.000 000 968		-0.000 000 704		0.649 998 966	
12	-0.000 001 144		-0.000 000 832		0.649 998 778	
13	-0.000 001 320		-0.000 000 960		0.649 998 590	
14	-0.000 001 496		-0.000 001 088		0.649 998 402	
15	-0.000 001 672		-0.000 001 216		0.649 998 214	
16	-0.000 001 848		-0.000 001 344		0.649 998 026	
17	-0.000 002 024		-0.000 001 472		0.649 997 838	
18	-0.000 002 200		-0.000 001 600		0.649 997 650	
19	-0.000 002 376		-0.000 001 728		0.649 997 462	
最小二乘法	m	-0.446 690 497 534 5	n	0.611 451 884 321 6	p	-0.653 141 785 963 7
基于产生式法	m	-0.611 451 884 496 9	n	-0.446 690 497 663 1	p	-0.653 141 785 711 6

对表 1 中的计算数据进行对比可以看出,基于产生式的空间直线拟合方法的计算精度已经达到 10^{-9} ,因此在工程实践中完全可以替代最小二乘法。从表 2 可以看出,在空间点密集时产生式方法不产生直线偏转,比最小二乘法更可靠。

4 应用实例

采用机器人某一位姿的各关节轴线反解机器人运动学参数是一种直接有效的参数识别方式。然而各关节轴线不能直接得到,必须通过一定的测量和计算才能获得。关节轴线测量原理如图 3 所示。

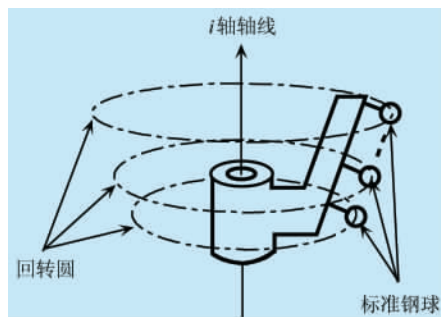


图3 关节轴线测量原理图

Fig. 3 Measure principle of joint axis

图 4 给出了测量方法和工具。在机械本体上粘结一些测量用的钢球,测量工具为 CimCore 的一种关节臂测量仪。通过测量可得到机械臂各关节轴线上的离散点集。采用上述产生式直线拟合算法可得到如表 3 所示轴线参数。

得到这些轴线参数后, 根据识别公式计算出机器人的 DH 参数如表 4 所示。



图4 机械臂及测量

Fig. 4 Robot arm and measure

表 3 轴线参数
Table 3 Parameters of axis

	起始点			方向余弦		
	<i>x</i>	<i>y</i>	<i>z</i>	<i>x</i>	<i>y</i>	<i>z</i>
1 轴	1.231 766 623	0.999 982 726	0.995 793 487	-0.545 777 207	-0.608 366 724	-0.576 209 311
2 轴	-0.003 458 946	0.007 785 237	0.102 733 905	-0.696 016 665	-0.071 043 633	-0.714 502 348
3 轴	1.647 924 723	0.811 265 899	1.923 230 777	0.602 653 533	0.602 452 371	0.523 316 213
4 轴	1.229 536 133	1.010 101 624	1.004 649 956	-0.669 060 452	-0.534 038 259	-0.516 876 436
5 轴	-0.003 744 988	-0.091 639 549	0.003 128 010	-0.635 020 640	-0.487 424 530	-0.599 304 693
6 轴	1.747 899 414	0.807 755 859	1.923 708 892	0.611 838 624	0.610 713 654	0.502 675 175

表 4 机械人 DH 参数
Table 4 DH parameters of the robot arm

	$\theta/(^{\circ})$	$\alpha/(^{\circ})$	<i>a</i> /mm	<i>d</i> /mm
T1	139.507 710 520 539	-91.484 059 527 711	201.367 619 153 294	-389.298 754 966 625
T2	-90.285 650 380 343	-0.211 404 255 910	1 112.566 832 445 129	-46.467 700 082 877
T3	0.452 594 151 557	-90.062 248 733 814	260.332 918 666 243	-1.487 266 599 437
T4	-1.129 726 763 981	86.237 259 428 842	0.741 605 219 105	1 325.240 075 108 269
T5	-86.710 816 624 951	90.011 767 396 621	4.886 929 917 921	-2.748 264 434 855
T6	178.635 591 989 439	89.838 012 255 116	0.226 735 690 241	313.339 428 171 977

根据计算出的 DH 参数，即可建立运动学模型用于运动控制。图 5 为采用上述 DH 参数控制机器人写汉字。从所写字的工整程度上，足以看出这组物理参数精度，也反映了产生式直线拟合算法的计算精度。



图 5 机械人在写字
Fig. 5 The robot is writing

5 结论

在分析最小二乘法空间点集直线拟合原理的基础上提出了一种基于产生式的空间直线拟合方法，给出了部分算法，同时与最小二乘法进行了比较，最后通过在机器人参数

测量进行了实际验证。通过验证证明，该空间直线拟合方法改善了最小二乘法的缺陷，其计算精度可以达到工程实际应用水平，可以在工程中推广应用。

参考文献 (References)

[1] 杜明芳. 空间直线拟合[J]. 北京印刷学院学报, 1996(2): 27-31.
Du Mingfang. *Journal of Beijing Institute of Graphic Communication*, 1996(2): 27-31.

[2] 李立声. 运用改进的 Gause-Newton 优化算法拟合圆柱和空间直线[J]. 现代计量测试, 1999, 7(2): 26-29.
Li Lisheng. *Modern Measurement and Test*, 1999, 7(2): 26-29.

[3] 雷贤卿, 周彦伟, 李言, 等. 机床导轨对主轴回转轴线平行度的精密测量[J]. 制造技术与机床, 2006(5): 68-71.
Lei Xianqing, Zhou Yanwei, Li Yan, et al. *Manufacturing Technology & Machine Tool*, 2006(5): 68-71.

[4] 李增兵, 张晓帆, 李瑞华, 等. 直线拟合快速实现的一种新算法 [J]. 计算机测量与控制, 2006, 14(11): 1524-1525, 1528.
Li Zengbing, Zhang Xiaofan, Li Ruihua, et al. *Computer Measurement & Control*, 2006, 14(11): 1524-1525, 1528.

[5] 孙家广, 陈玉健, 辜凯宁. 计算机辅助几何造型技术[M]. 北京: 清华大学出版社, 1990.
Sun Jiaguang, Chen Yujian, Gu Kaining. *Computer aided geometrical modeling technology*[M]. Beijing: Tsinghua University Press, 1990.

[6] 朱珊珊, 李书琴. 面向对象技术在产生式系统设计中的应用 [J]. 西北农林科技大学学报: 自然科学版, 2006, 34(8): 161-164.
Zhu Shanna, Li Shuqin. *Journal of Northwest A & F University: Natural Science Edition*, 2006, 34(8): 161-164.

[7] 林尧瑞, 马少平. 人工智能导论[M]. 北京: 清华大学出版社, 2000.
Lin Yaorui, Ma Shaoping. *Introduction to artificial intelligence* [M]. Beijing: Tsinghua University Press, 2000.

(责任编辑 代丽)