

文章编号: 2095-2163(2019)01-0122-05

中图分类号: TP249

文献标志码: A

基于手势轨迹的工业机器人轨迹规划

吕亚辉, 严雨灵

(上海工程技术大学 航空运输学院, 上海 201620)

摘要: 本文使用 Leap Motion 手势识别设备获取手势轨迹路径, 使用中值滤波和均值滤波对原路径进行去噪, 获取较平滑的原手势轨迹。在需要工业机器人进行多次重复高效的轨迹运行时, 将手势轨迹相对于关节空间进行时间-冲击最优轨迹规划, 根据需要选取一定数量关键点。本文使用粒子群算法对所给关键点进行轨迹优化, 根据结果对比表明优化后粒子群具有收敛速度快, 寻优效果好的优点, 并使用粒子群算法对手势轨迹进行机器人 6 轴关节空间轨迹优化, 获得时间-冲击最优轨迹。

关键词: 手势识别; 轨迹规划; 粒子群优化

Trajectory planning of industrial robot based on gesture trajectory

LÜ Yahui, YAN Yuling

(School of Air Transportation, Shanghai University of Engineering Science, Shanghai 201620, China)

[Abstract] In order to realize the robot trajectory with gesture trajectory, this paper uses the Leap Motion gesture recognition device to acquire the gesture trajectory path, and uses median filtering and averaging filtering to denoise the original path for obtaining a smoother original gesture trajectory. When the industrial robot is required to perform repeated and efficient trajectory operation, the time-impact optimal trajectory planning of the gesture trajectory relative to the joint space is performed, and a certain number of key points are selected according to the needs. In this paper, the particle group algorithm is used to track the given key points. According to the result comparison, the optimized particle swarm has the advantages of fast convergence speed and good optimization effect. The particle swarm algorithm is used to optimize the six-axis joint space trajectory of the gesture trajectory, and the time-impact optimal trajectory is obtained.

[Key words] gesture recognition; trajectory planning; particle swarm optimization

0 引言

随着机器人人机交互技术的发展, 通过手势来控制机器人运行已经可以实现, 相比于人工编程控制机器人, 使用自然手势对工业机器人进行示教可以降低对操作人员的要求, 不需要机器人编程的知识也可以完成对机器人的控制与示教。如蒋穗峰^[1]等用稀疏自编码网络与 Softmax 分类器结合的方法对手势图像进行识别, 过程包含预训练和微调, 操作员可以在不佩戴任何设备、任意位置和环境下都可分割出手势图像。陈畅^[2]等使用 Leap Motion 传感器获取手势坐标动态, 以使机器人运动, 实现移动、抓取和放置等动作。

为实现手势轨迹控制机器人, 需要处理离散性较大的初始轨迹路径, 降低冲击提高效率 and 稳定性, 对于需要机器人进行重复性的工作时更需要对机器人路径进行优化。如付荣等^[3]使用遗传算法进行

时间最优轨迹规划, 使用 2 个适应度函数寻找最优 3-5-3 插值多项式, 得到了时间最优的轨迹。但是该方法并未考虑冲击, 其结果中可以发现加加速度有很大突变, 具有较大的冲击, 不仅对机器人的工作精度有影响, 还会缩短使用寿命^[4]。为了解决这种问题, 文献[5]中进行最优冲击轨迹规划, 采用 S 型曲线对 B 样条进行插补, 冲击明显减小。除此之外为了在要求运动稳定性的前提下寻找最优时间轨迹, Lin^[6]使用 PSO 算法和类聚合算法在关节空间进行规划, 通过优化轨迹中所安插的节点求得最优冲击轨迹。

本文针对工业机器人的冲击优化问题, 在获取初始手势轨迹后, 进行滤波去噪, 之后根据需要选择关键点, 以 Z 型路径为例, 根据其曲率特征选取 4 个关键点, 使用粒子群算法对原手势轨迹进行冲击优化。获得了在机器人关节空间约束范围内的冲击优化轨迹, 并在 Epson 机器人上进行实验, 结果表明,

基金项目: 上海工程技术大学研究生科研创新项目(E 3-0903-18-01176)。

作者简介: 吕亚辉 (1995-), 男, 硕士研究生, 主要研究方向: 机器人智能控制; 严雨灵 (1994-), 男, 硕士研究生, 主要研究方向: 计算机视觉、虚拟现实。

收稿日期: 2018-10-11

冲击优化轨迹比原手势轨迹运行时间短且冲击力小。

1 手势获取及预处理

完整手势识别包括对手势的初始分割、跟踪、手势分析与理解,为实现手势的人机交互需要对手势进行理解及控制反馈^[7],对手势分割进行识别匹配,根据不同的手势匹配不同的机器人动作,跟踪手势轨迹,并对机器人进行控制。

本文使用手势识别设备跟踪手势轨迹以获取轨迹信息,即使用 Leap Motion 设备的手势提取。为研究工业机器人跟踪手势的轨迹问题,仅提取手势中的食指坐标信息,本文中食指随机走 Z 字型为例(由于手势提取设备由下而上采取,坐标为镜面相反关系),其所提取路径凸显如图 1 所示。图中反 Z 字型与原手势横坐标相反成镜面反射,且该 Z 型轨迹上存在微小偏移,手指运动中即使无意,手势轨迹也会出现颤抖。

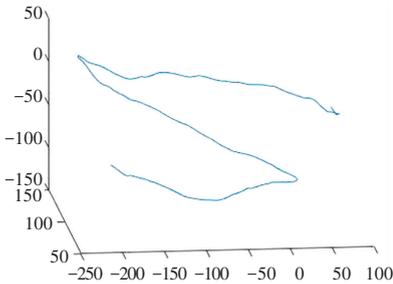


图 1 手势轨迹坐标

Fig. 1 Gesture track coordinates

为了将手势提取轨迹应用于工业机器人,需要对路径进行滤波去噪平滑,根据手势速度及采点速度确定中值及均值滤波的范围空间,文中取各点上下相邻 5 个数的中值或均值。如图 2、图 3 所示,分别为均值滤波和中值滤波去噪后的结果,由图 2 中可知中值滤波取相邻数的中值在大部分位置去噪平滑时能一定程度保留原轨迹信息,但在拐点处处理不如均值滤波。均值滤波在其它位置匹配度不如中值滤波,两者均仅处理相近点的去噪不能对整体 Z 型轨迹平滑。

2 轨迹优化

上节对手势轨迹进行了去噪优化,但若应用在高速且重复运行的工业机器人上,为降低冲击力及其产生的磨损,需要对上节所得轨迹选取关键点,并在 6 轴机器人关节空间进行时间-冲击最优轨迹规划。对于不同的生产方案拥有不同的轨迹路径,选

择冲击优化的关键点也不同,本文以 Z 型轨迹为例进行冲击优化,至少选取起点、终点及 2 处拐点 4 处关键点,实际工业生产中应随必经关键点位置数量决定。

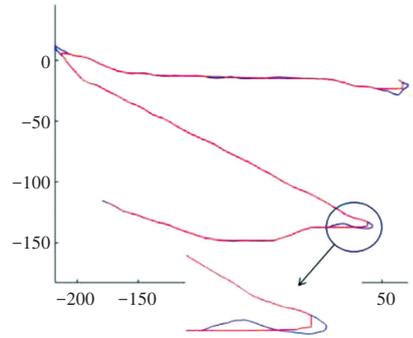


图 2 中值滤波处理

Fig. 2 Median filter processing

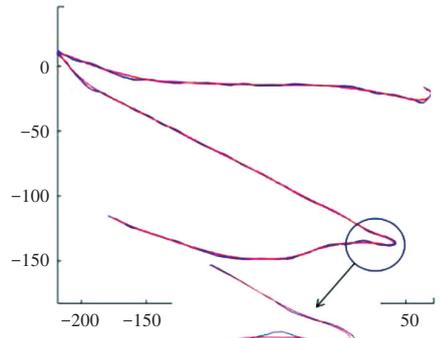


图 3 均值滤波处理

Fig. 3 Mean filter processing

2.1 轨迹特征点提取

为实现关节空间时间-冲击最优轨迹规划,以选取 4 处特征点为例。曲线拟合法和离散结构法是提取离散曲线特征点的常用方法。前者通过对给定点拟合近似函数,再估算给定点的几何特征;后者直接运用微分几何理论,计算离散点几何特征。本文采用第二种方法来提取离散空间曲线的几何特征。对于文中非闭合曲线,首末 2 点必须被选取为特征点,设置曲率阈值,将所有点中曲率大于设定阈值的点选取为特征点。2 拐点由去噪后的手势轨迹曲线为基础,以公式(1)获取曲线中曲率最大的 2 处拐点。

$$k = \frac{|y''|}{(1 + y'^2)^{\frac{3}{2}}} \quad (1)$$

其中, y' 、 y'' 分别为基于离散点序号的各对应坐标关系的一阶导数和二阶导数,其对所有离散点计算得,在 287 个离散点中获取第 84 点和第 191 点 2 处特征点,结合首末 2 点,对此 4 处关键点进行关节空间冲击优化,即取坐标 $(-175, -117, 0)$, $(37, -$

136,0), (-212, 6, 0), (64, -16, 0)。4 点位置以 Epson C3 机器人为例,对其逆运动学求解,获得 4 点处 6 轴角度见表 1。其中,第四轴主要控制绕 Z 轴旋转,Z 型轨迹未涉及,故均为起点 0。对于关节空间冲击最优,即对关节轴的冲击力优化。

表 1 关节空间坐标点

Tab. 1 Joint space coordinate points

关节	路径点 1	路径点 2	路径点 3	路径点 4
J_1	-8.786°	-4.953°	10.739°	5.139°
J_2	0.650°	26.202°	11.566°	-28.647°
J_3	-0.643°	35.271°	-9.631°	39.286°
J_4	0°	0°	0°	0°
J_5	-0.006°	-9.068°	-1.935°	-10.638°
J_6	8.786°	4.953°	-10.739°	-5.139°

2.2 插值多项式

为实现手势轨迹冲击优化,本文采用 5-5-5 插值多项式曲线。付荣^[2]等使用 3-5-3 多项式插值

曲线对时间最优轨迹规划,但其两段时间相连接处的冲击未考虑在内,为对原轨迹进行冲击优化,本文构造 5-5-5 插值多项式,其中 5 次多项式可以保证轨迹连接处的加速度连续,其多项式通式为:

$$X_{i1} = a_{i6} + a_{i5}t_1 + a_{i4}t_1^2 + a_{i3}t_1^3 + a_{i2}t_1^4 + a_{i1}t_1^5 \tag{2}$$

$$X_{i2} = a_{i12} + a_{i11}t_2 + a_{i10}t_2^2 + a_{i9}t_2^3 + a_{i8}t_2^4 + a_{i7}t_2^5 \tag{3}$$

$$X_{i3} = a_{i18} + a_{i17}t_3 + a_{i16}t_3^2 + a_{i15}t_3^3 + a_{i14}t_3^4 + a_{i13}t_3^5 \tag{4}$$

其中 X_{ij} 为第 i 个轴的第 j 段的角度,在点对点的轨迹规划中起点和终点的角度、角速度、角加速度已知,且路径点 X_{i20} 、 X_{i30} 已知,则可根据时间 t_1 、 t_2 、 t_3 获得多项式系数,并由角度位移的 5 次多项式求导计算该段时间内的速度、加速度和加加速度。根据每段时间段的对应关系则可有如下列矩阵与系数 a 的关系式,如公式(5) ~ (7) 所示:

$$A = \begin{bmatrix} t_1^5 & t_1^4 & t_1^3 & t_1^2 & t_1^1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5t_1^4 & 4t_1^3 & 3t_1^2 & 2t_1^1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 20t_1^3 & 12t_1^2 & 6t_1^1 & 2 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 60t_1^2 & 24t_1^1 & 6 & 0 & 0 & 0 & 0 & 0 & -6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & t_2^5 & t_2^4 & t_2^3 & t_2^2 & t_2^1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5t_2^4 & 4t_2^3 & 3t_2^2 & 2t_2^1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 20t_2^3 & 12t_2^2 & 6t_2^1 & 2 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 60t_2^2 & 24t_2^1 & 6 & 0 & 0 & 0 & 0 & -6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & t_3^1 & t_3^1 & t_3^1 & t_3^1 & t_3^1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5t_3^1 & 4t_3^1 & 3t_3^1 & 2t_3^1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 20t_3^1 & 12t_3^1 & 6t_3^1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 60t_3^1 & 24t_3^1 & 6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5}$$

$$b = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ r_3 \ 0 \ 0 \ 0 \ r_0 \ 0 \ 0 \ 0 \ r_2 \ r_1] \tag{6}$$

$$A * a = b \tag{7}$$

其中, a 为 5 次多项式系数, r_0 、 r_1 、 r_2 、 r_3 分别为该轴起点、第一点、第二点、终点的角度,由于矩阵的特性要求每步运算时需要考虑矩阵 A 的可逆性。

3 冲击最优轨迹规划

为寻找最优时间 - 冲击轨迹规划, Gasparetto

等^[3]给予每个优化目标相应的权重,将其转化为单目标。通过调整时间、冲击 2 个优化目标的权重,获得了保证冲击较小的同时也保证了速度的提高,运动时间有较大缩短。文献[8]中将时间、加速度和冲击作为对象进行混合最优轨迹规划。本文对粒子的加加速度与最大加加速度约束的差值加权,并与

总时间求和作为适应度函数, 如公式 (8) 所示:

$$f_1 = W_t(t_1 + t_2 + t_3) + W_v * |V_{max} - \max Vi| + W_a * |a_{max} - \max a| + W_j * |j_{max} - \max j| \quad (8)$$

而当该次迭代的全局最优解不满足约束, 则需对该单维时间段进行修正调整, 以满足约束条件的最短时间为全局最优解而进入下次迭代, 其修正函数如公式 (9) 所示:

$$f_2 = W_t * (t_1 + t_2 + t_3) + W_v * |\max Vi| + W_a * |\max a| + W_j * |\max j| \quad (9)$$

3.1 粒子群优化算法

粒子群优化算法是一种基于群体智能的全局进化优化算法, 模拟鸟群觅食行为, 种群个体之间通过相互协作来寻求最优解, 每个优化问题的潜在解都是搜索空间中的一个粒子。每个粒子具有位置和速度 2 个特征, 粒子的位置坐标对应的目标函数值即可作为该粒子的适应度, 算法通过适应度来衡量粒子的优劣。在每一次迭代中, 粒子通过跟踪 2 个“极值”来更新自己。第一个极值就是粒子本身所找到的最优解, 叫做个体极值 pBest; 另一个极值是整个种群目前找到的最优解, 这个极值是全局极值 gBest。在找到这 2 个最优值时, 粒子根据公式 (10) ~ (11) 更新速度和位置:

$$v_{i+1} = W * v_i + C_1 * rand * (pbest_i - x_i) + C_2 * rand * (gbest_i - x_i) \quad (10)$$

$$x_{i+1} = x_i + v_{i+1} \quad (11)$$

其中, W 为惯性权重, 是上次粒子速度对此次粒子速度的影响程度。Kennedy 提出作为收缩因子, 使其随迭代减小粒子震荡, 从而加快收敛速度。对于惯性权重, 本文采用随迭代次数而自适应变化的时变权重如公式 (12) 所示:

$$W = 0.3 * (\frac{i}{T})^k \quad (12)$$

其中, i 为当前迭代次数, T 为总迭代次数, 当 $k > 1$ 时, 惯性权重随迭代次数由 1 降为 0.3, 且前期降低较慢, 即惯性权重在迭代前期受上次粒子速度影响较大, 而在迭代后期影响程度迅速减小, 可在一定程度防止粒子群过早收敛, 且不影响迭代后期的搜索精度。具体算法流程如下:

(1) 初始化粒子。设定粒子数和其维数, 粒子以 3 段时间 t 组成三维粒子;

(2) 计算粒子的适应度。根据粒子的 3 段时间带入式 (5) ~ (7) 可求出系数向量 a , 并由此带入式 (2) ~ (4) 分别求出 3 段时间内的最大速度、最大加

速度、最大加加速度, 以式 (8) 为适应度函数求取个体极值 pBest 和全局极值 gBest;

(3) 判断全局最优解是否超出最大约束, 若未超出最大约束, 则以式 (8) 为适应度函数对 3 段时间以进行迭代, 求得新的粒子, 并对所得结果与原粒子群所得结果的适应度值进行比较。选取适应度值较低的为当前迭代所得粒子群; 若超出最大约束, 则对超出的该单维时间段进行以式 (9) 为适应度函数迭代选取新的满足约束的全局最优解;

(4) 判断是否达到终止条件, 若否, 则返回 (2);

(5) 以满足 piper 准则的机器人为例, 前 3 轴可确定机械手末端位置, 则根据路径规划是否有姿态变化选择 6 轴角度或 3 轴角度优化, 并分别对所需求解的 6 个轴或 3 个轴求取最优解, 并选取每段时间的最大值作为最终最优时间, 以最终 3 段时间带入各轴轨迹点要求内, 求得最优轨迹。

选取工业机器人其中第一关节轴为例, 使用粒子群算法对该组关键点进行最优时间-冲击规划, 其迭代过程如图 4 所示。就其迭代效果对比可发现, 自适应粒子群算法相比于一般粒子群算法全局搜索更快, 且不影响最终搜索精度。

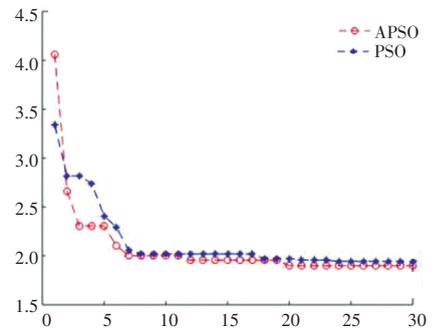


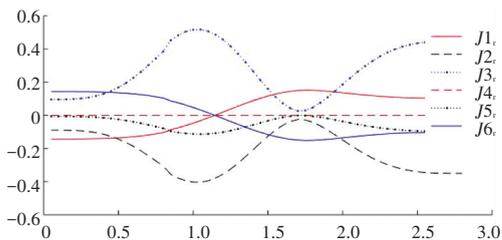
图 4 迭代收敛效果

Fig. 4 Iterative convergence effect

3.2 基于 PSO 算法的插值多项式拟合

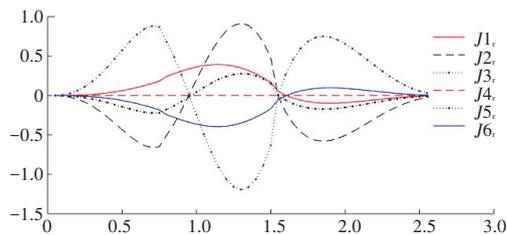
将 PSO 算法应用于机器人 6 轴关节上, 以加速度为 10 m/s^2 、加加速度为 50 m/s^3 为约束, 分别求得 6 轴 3 段最优时间, 每段时间选取 6 轴中最长时间, 获得 3 段时间: 0.842 0 s 、 0.710 1 s 、 1.234 8 s , 总时间 2.786 9 s 。并以公式 (2) ~ (3) 对该 3 段时间对 6 轴求得最终轨迹, 得迭代效果如图 5 所示, 其中角度以弧度为单位表示。

由图中可知, 第三个关键点处加加速度有突变, 但大小均未超过 30 rad/s^3 , 冲击较小。其最终优化结果轨迹如图 6 所示, 其中 Z 轴存在 0.2 mm 波动, 属于空间曲线。



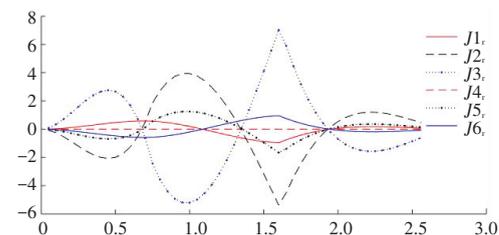
(a) 角度曲线

(a) Angle curve



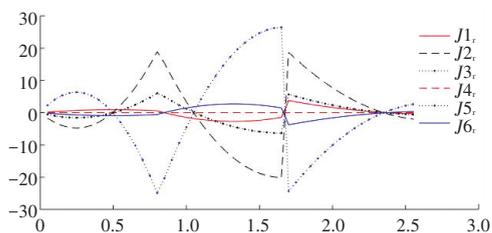
(b) 角速度曲线

(b) Angular velocity curve



(c) 角加速度曲线

(c) Angular acceleration curve



(d) 角加加速度曲线

(d) Angular jerk curve

图5 机器人6轴约束下角度、速度、加速度、加加速度最优曲线

Fig. 5 Optimal curve of angle, velocity, acceleration and jerk under the six-axis constraint of the robot

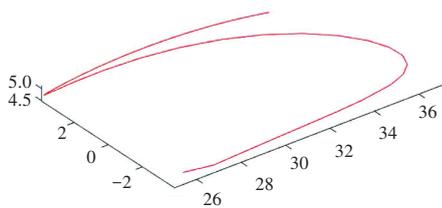


图6 最终冲击优化轨迹

Fig. 6 Final impact optimization trajectory

4 结束语

为降低工业机器人重复生产行为的冲击和磨损、简化机器人冲击优化轨迹的编程,本文通过 Leap Motion 获取手势坐标信息,在使用均值滤波和中值滤波对原离散轨迹进行去噪平滑后,以曲率选取的关键点为例,使用粒子群算法进行关节空间时间-冲击最优轨迹规划。结果表明,优化后的轨迹在不影响时间的情况下,大幅降低了机器人运动的冲击力。对于本文中自适应的粒子群优化算法,相比于一般粒子群优化算法,在不影响搜索精度和全局性的情况下,前期搜索速度更快。

参考文献

[1] 陈畅,陈亮,周雪峰. 基于自然手势交互的工业机器人示教系统设计与实现[J]. 制造业自动化,2018,40(2):21-25.

[2] 付荣,居鹤华. 基于 AGA 的时间最优机械臂轨迹规划算法[J]. 计算机应用研究,2011,28(9):3275-3278.

[3] GASPARETTO A, LANZUTTI A, VIDONI R, et al. Experimental validation and comparative analysis of optimal time-jerk algorithms for trajectory planning[J]. Robotics & Computer Integrated Manufacturing, 2012, 28(2):164-181.

[4] 杨锦涛,姜文刚,林永才. 工业机器人冲击最优的轨迹规划算法[J]. 科学技术与工程,2014,14(28):64-69.

[5] 余阳,林明,林永才. 基于混合遗传算法的工业机器人最优轨迹规划[J]. 计算机工程与设计,2012,33(4):1574-1580.

[6] LIN H I. A fast and unified method to find a minimum-jerk robot joint trajectory using Particle Swarm Optimization[J]. Journal of Intelligent & Robotic Systems, 2014, 75(3-4):379-392.

[7] 沈靖皓. 基于手势引导的工业机器人示教技术研究及应用[D]. 杭州:浙江大学,2017.

[8] DUAN Honggang, ZHANG Rongmin, YU Fei, et al. Optimal trajectory planning for glass-handing robot based on execution time acceleration and jerk[J]. Journal of Robotics, 2016(12):2.