

DOI:10.11784/tdxbz202310020

一种基于 DTW-DP-GMM 的工业机器人轨迹学习策略

肖 洒^{1,2}, 陈旭阳^{1,2}, 叶锦华^{1,2}, 吴海彬^{1,2}

(1. 福州大学机械工程及自动化学院, 福州 350108;

2. 福建省特种智能装备安全与测控重点实验室, 福州 350108)

摘要: 针对机器人示教编程过程中使用高斯混合模型(GMM)规划运动轨迹时存在的高斯分布个数难以选择、复现轨迹精度较低等问题, 提出了一种复合的机器人运动轨迹学习策略. 该策略包含动态时间规整(DTW)算法、高斯混合模型与道格拉斯-普克(DP)算法. 首先, 针对示教过程中采集的多条轨迹在时间长度上存在差异的问题, 采用 DTW 算法来统一示教轨迹在时域上的变化. 其次, 使用 GMM 算法对示教轨迹的特征进行提取, 并利用高斯混合回归(GMR)算法将其重构为复现轨迹. 在这个过程中采用 DP 算法来预估 GMM 算法的关键参数高斯分布的数量, 与传统方法相比, 能够简单直观地得到相对准确的参数值. 利用 DP 算法对复现轨迹的数据点进行稀疏化并优化, 不仅确保了机器人最终运动轨迹的精度, 而且大幅减少了最终轨迹数据点的数量. 最后, 进行了不同形状的模拟焊接轨迹学习规划实验. 结果表明: 经由 DTW 对齐后的示教轨迹具有更加明显的运动特征, 经过 GMM-GMR 学习输出的复现轨迹具有良好的表征结果; 在使用 GMM-GMR 算法学习示教轨迹的过程中, 采用 DP 算法可以有效预估高斯分布个数; 经过 DP 算法稀疏化并优化的最终轨迹的平均位置误差均在 0.500 mm 以内, 其最大误差可以控制在 0.800 mm 以内, 可以满足焊接轨迹规划的精度要求, 验证了该策略的有效性和优越性.

关键词: 工业机器人; 示教编程; 高斯混合模型; 道格拉斯-普克算法; 动态时间规整; 轨迹复现
中图分类号: TP242.2 **文献标志码:** A **文章编号:** 0493-2137(2025)01-0068-13

A Strategy for Industrial Robot Trajectory Learning Based on DTW-DP-GMM

Xiao Sa^{1,2}, Chen Xuyang^{1,2}, Ye Jinhua^{1,2}, Wu Haibin^{1,2}

(1. School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou 350108, China;

2. Fujian Key Laboratory of Special Intelligent Equipment Safety Measurement and Control, Fuzhou 350108, China)

Abstract: The users often encounter issues such as difficulty in selecting the appropriate number of Gaussian distributions and low accuracy in reproducing trajectories when using Gaussian mixture model (GMM) to plan robot trajectories during programming by demonstration. To address these concerns, a composite strategy is proposed, which integrates dynamic time warping (DTW) algorithm, GMM and the Douglas-Peucker (DP) algorithm. First, to address the issue of varying time lengths in multiple trajectories, the DTW algorithm is used to align the variation of the demonstrated trajectories in the time domain. Second, the motion features are learned from the aligned demonstrated trajectories using GMM, which can subsequently be reconstructed into a reproduced trajectory using Gaussian mixture regression (GMR). In this process, the number of Gaussian distributions, a key parameter of GMM, is estimated by DP algorithm, which can derive a relatively precise parameter value simply and intuitively compared with the traditional method. Furthermore, the DP algorithm is employed to sparsify and optimize the data points in the reproduced trajectory, ensuring that the final trajectory maintains high precision while drastically reducing the num-

收稿日期: 2023-10-18; 修回日期: 2023-12-29.

作者简介: 肖 洒 (1984—), 男, 博士研究生, 工程师, xiaosa1014@126.com.

通信作者: 吴海彬, wuhb@fzu.edu.cn.

基金项目: 国家重点研发计划资助项目(2018YFB1308603); 福建省高校产学研合作项目(2022H6016).

Supported by the National Key Research and Development Program of China (No. 2018YFB1308603), the Industry Academia Cooperation Project of Fujian Province, China (No. 2022H6016).

ber of data points in the final trajectory. Finally, experiments conducted on simulated welding trajectories of different shapes are carried out. The experimental results show that the demonstrated trajectories aligned by DTW exhibit more pronounced motion features, and the reproduced trajectory generated using GMM-GMR has great representation result; moreover, the DP algorithm effectively estimates the necessary number of Gaussian distributions for GMM-GMR learning. The average positional errors in final trajectories sparsified by the DP algorithm are within 0.500 mm, and the maximum errors can be controlled within 0.800 mm, meeting the precision requirements of welding trajectory planning. It verifies the effectiveness and the superiority of the proposed strategy.

Keywords: industrial robot; programming by demonstration; Gaussian mixture model (GMM); Douglas-Peucker algorithm; dynamic time warping (DTW); trajectory reproduction

随着“工业 5.0”时代的到来^[1],工业机器人因为具备高速率、高精度以及顺从性等优势,被广泛应用于代替人类大量体力劳动的工作,例如焊接、喷涂、装配等。机器人可以根据专业人员编写的运行程序完成相应的作业。然而,如今的工业产品生产逐渐呈现出多品种、小批量、短周期、个性化等特点,因此如何根据不同的任务需求实现机器人运动的快速编程一直是机器人研究的热点和难点。

近年来,示教编程(programming by demonstration, PbD)^[2]被认为是一种可行的解决方案。该方法又名模仿学习^[3]或人-机器人技能传递^[4],其本质是人类将自己的技能做通用化描述后传递给机器人,使得后者具备相应的作业能力。具体而言,使用特定的数学模型以示教数据为样本进行建模学习并获得描述示教行为运动技能特征的模型参数,从而实现对示教行为的表征。因此,机器人可以根据学习获得的运动控制策略复现或泛化相应的运动轨迹。与传统编程技术相比,示教编程具有成本低、效率高、不依赖机器人本体平台等优点。动态运动基元(dynamic movement primitives, DMP)^[5]以及概率模型法^[6]是典型的示教编程算法。

DMP 利用一系列线性可微方程与可学习的自动强迫项,将运动行为建模为非线性动态吸引子模型,在模仿示教技能时能够确保泛化轨迹收敛到目标点。Yu 等^[7]采用 DMP 同时模拟运动轨迹和阻抗特征,并在抽水任务中验证了所提方法的有效性。Wang 等^[8]利用 DMP 分别对机械手的运动轨迹和力变化进行建模,从而采用混合力/运动控制方法来实现跟踪、复制和泛化所需位置和力的运动技能。虽然 DMP 仅学习单条示教轨迹即可实现点到点和周期运动的泛化,但如果示教轨迹存在噪声,那么轨迹学习的效果就会非常不理想。

基于概率模型的学习方法具有强大的噪声处理能力以及良好的鲁棒性,能够处理高维问题,常被用于机器人复杂运动的模仿学习^[9-10]。目前常用的有隐

马尔可夫模型(hidden Markov model, HMM)^[11]和高斯混合模型(Gaussian mixture model, GMM)^[12]等。Vakanski 等^[13]用 HMM 表征示教轨迹,并通过在状态转移时设置关键点的办法,实现对轨迹的拓展与调整,在刷漆作业中验证了该方法的有效性。Deng 等^[14]利用 GMM 提出了一种从演示中学习人类技能的策略,该策略包括学习和复现阶段,可以在合作操纵过程中为人类提供必要的帮助。基于概率模型的学习方法能够很好地保持示教轨迹的空间形状特征,然而若多条示教轨迹间存在时间长度上的差异,则学习效果欠佳。

虽然示教编程所获得的复现轨迹可以表征人类示教行为,但是由于人类示教的过程存在各种误差,例如人工抖动偏差等,相应的示教轨迹难免存在噪声。因此示教编程更适用于文字书写、物品搬运等动作轨迹精确性要求不高的应用,而无法满焊接、涂胶等轨迹精度要求较高的应用需求。如何通过示教编程获得满足精度需求的运动轨迹是一个迫切值得研究的问题。

针对以上问题,本文将动态时间规整(dynamic time warping, DTW)算法、道格拉斯-普克(Douglas-Peucker, DP)算法与基于 GMM 的示教编程算法结合起来,提出了一种工业机器人轨迹学习策略。在使用 GMM 建模前,先用 DTW 算法对齐示教轨迹并用 DP 算法来预估所需的高斯分布数量;通过训练 GMM 获得相应的参数后,使用高斯混合回归(Gaussian mixture regression, GMR)输出相应的复现轨迹;接下来再次利用 DP 算法对复现轨迹进行稀疏化,提取关键点并输出满足精度需求的最终轨迹。最后,通过模拟焊接轨迹的示教编程实验验证了所提出的轨迹学习策略的有效性与优越性。

1 机器人轨迹学习策略的整体概述

为了实现工业机器人的演示编程,本文设计了机

器人轨迹学习框架,其整体方案如图 1 所示.

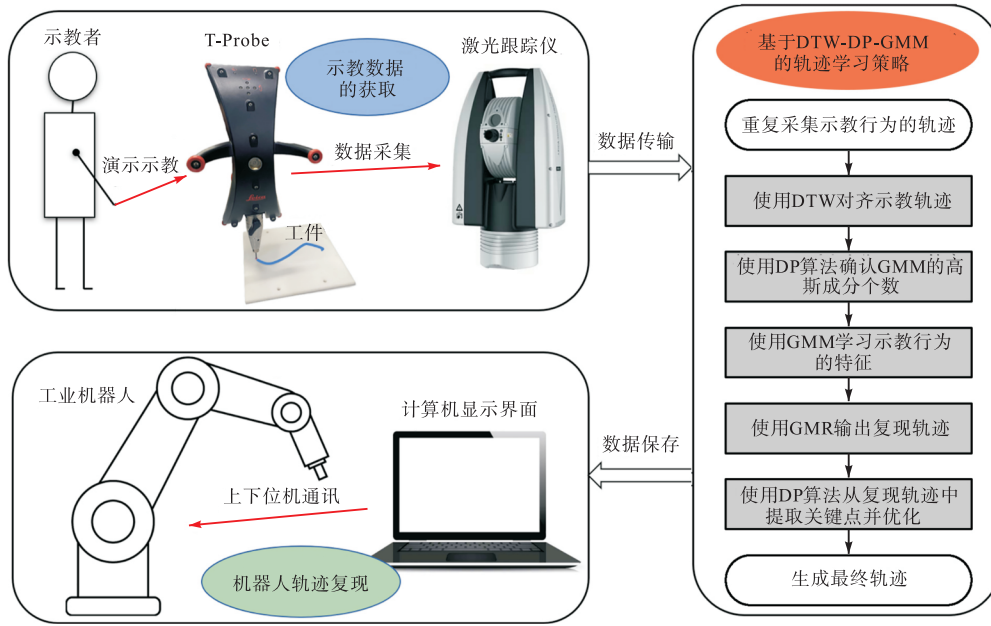


图 1 机器人轨迹学习整体方案
Fig.1 Entire solution for robot trajectory learning

示教阶段:利用激光跟踪仪获取演示轨迹数据.以模拟焊接任务为例,示教者手持 T-Probe 沿着期望的焊接路径移动,同时激光跟踪仪以特定频率采集相应的位置信息.为了能够获取足够的数,需要对特定任务进行多次完整的演示.

学习阶段:首先,针对示教阶段采集的数据在时间长度上存在差异的问题,使用 DTW 算法通过非线性变换将各段示教轨迹在时域上对齐.然后,利用 GMM 算法从对齐后的演示轨迹中学习运动特征,利用 GMR 算法将其重构为复现轨迹.在这个过程中,使用 DP 算法预估高斯分布的个数.最后,再次使用 DP 算法将复现轨迹稀疏化,从中提取关键轨迹点,并进行适当优化,生成满足精度需求的最终轨迹.

复现阶段:上位机(计算机)通过通讯端口将最终轨迹的相关数据发送给下位机(机器人).获得最终轨迹数据后,可将学习到的运动策略控制变量映射到机器人控制器中.于是机器人便可复现出示教者的动作,进而完成相应的作业任务.

2 机器人从示教轨迹中学习的策略

2.1 使用 DTW 算法将多条示教轨迹对齐

人工示教采集的多条轨迹难免存在差异,即在时间序列上长短不一.这种差异会增加示教行为特征的不确定性,进而影响 GMM 从示教中学习的效果,因此有必要将示教轨迹在时间长度上对齐^[15]. DTW

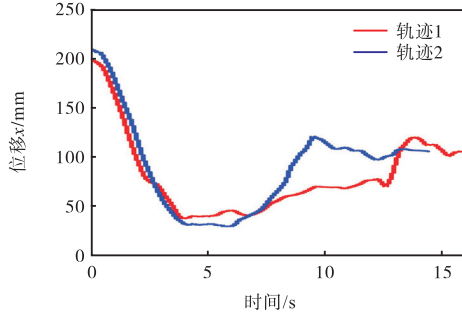
可以解决不同数据序列时间对齐的问题,其通用性较强且受噪声的影响相对较少. DTW 的核心思想是通过非线性变换将两个不同长度的时间信号转换为相同长度,并使变形后两个轨迹之间的距离最小^[16]. 用一个矩阵 V 来描述这种算法,该矩阵由一组示教轨迹中所有数据点与另一组示教轨迹中所有数据点的对应欧氏距离填充. 下面的动态规划方法用于确定从矩阵 V 右下角的元素 $v_{n,m}$ 到 $v_{1,1}$ 的路径,从而使路径元素的累积距离最小,即

$$DTW(i, j) = \begin{cases} \infty & i = 0 \cup j = 0 \\ v_{i,j} & i = 1 \cap j = 1 \\ v_{i,j} + \min(DTW(i, j-1), & \\ DTW(i-1, j), & \text{其他} \\ DTW(i-1, j-1)) & \end{cases} \quad (1)$$

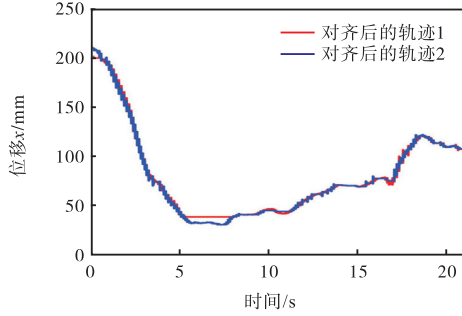
式中 $v_{i,j} \in V$ 是指第 1 组示教轨迹中的元素 i 与第 2 组示教轨迹中的元素 j 之间的加权欧氏距离. 如图 2 所示,使用 DTW 算法对图 2(a)中两条不同时间的示教轨迹进行规整对齐,对齐后的效果如图 2(b)所示.

2.2 使用 GMM-GMR 从示教轨迹中学习

示教编程的关键在于从示教者的演示中获得技能的概括表征,从而使机器人能够复现或泛化所学技能. 本节先介绍使用 GMM 从演示轨迹中学习并利用 GMR 生成复现轨迹的过程,然后再详细描述如何使用 DP 算法预估高斯分布数量,最后介绍如何使用



(a) 不同长时的示教轨迹



(b) 对齐后的轨迹

图2 用 DTW 对齐轨迹的示例

Fig.2 Example of aligning trajectories with DTW

DP 算法对复现轨迹进行稀疏化。

2.2.1 使用 GMM-GMR 学习并复现示教轨迹

记给定的 M 条示教轨迹数据 $\{\{\xi_{n,m}\}_{n=1}^{N_m}\}_{m=1}^M$ 为 $\{\xi_j\}_{j=1}^F$, 其中 N_m 表示第 m 条轨迹的长度(样本点个数), $F = \sum_{m=1}^M N_m$; 样本点 $\xi_j = [\xi_{j,t}, \xi_{j,s}]$, 其中 $\xi_{j,t}$ 表示样本点的输入信息, $\xi_{j,s}$ 表示对应的输出信息. 采用 GMM 对样本建模^[17], 其概率密度函数为

$$p(\xi_j) = \sum_{k=1}^K \pi_k \mathcal{N}(\xi_j; \mu_k, \Sigma_k) = \sum_{k=1}^K \pi_k \frac{1}{\sqrt{(2\pi)^D |\Sigma_k|}} e^{-\frac{1}{2}(\xi_j - \mu_k)^T \Sigma_k^{-1} (\xi_j - \mu_k)} \quad (2)$$

式中: D 表示示教轨迹的维度; K 表示组成该高斯混合模型的高斯分布个数; π_k 、 μ_k 和 Σ_k 则分别表示第 k 个高斯分布的先验概率、期望和方差, 且有

$$\begin{cases} \sum_{k=1}^K \pi_k = 1 \\ \mu_k = \begin{bmatrix} \mu_k^t \\ \mu_k^s \end{bmatrix} \\ \Sigma_k = \begin{bmatrix} \Sigma_k^{tt} & \Sigma_k^{ts} \\ \Sigma_k^{st} & \Sigma_k^{ss} \end{bmatrix} \end{cases} \quad (3)$$

GMM 的参数 $\{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$ 可以使用期望最大值(expectation maximum, EM)算法进行求解. 该算法分为 E-step 和 M-step 两个步骤: E-step 计算联合分

布的条件概率期望 E_k ; M-step 则根据当前 E_k 重新估计 GMM 的相关参数^[18]. 具体而言, 记条件概率 $p(k | \xi_j) = p_{k,j}$, 则有

$$\begin{cases} p_{k,j} = p(k | \xi_j) = \frac{\pi_k p(\xi_j | k)}{\sum_{i=1}^K \pi_i p(\xi_j | i)} \\ E_k = \sum_{j=1}^N p_{k,j} \end{cases} \quad (4)$$

对于第 u 次迭代学习, 计算方法如下.

E-step, 计算后验概率之和, 即

$$\begin{cases} P_{k,j}^{(u)} = \frac{\pi_k^{(u)} \mathcal{N}(\xi_j; \mu_k^{(u)}, \Sigma_k^{(u)})}{\sum_{i=1}^K \pi_i^{(u)} \mathcal{N}(\xi_j; \mu_i^{(u)}, \Sigma_i^{(u)})} \\ E_k^{(u)} = \sum_{j=1}^N P_{k,j}^{(u)} \end{cases} \quad (5)$$

M-step, 更新模型参数, 即

$$\begin{cases} \pi_k^{(u+1)} = \frac{E_k^{(u)}}{N} \\ \mu_k^{(u+1)} = \frac{\sum_{j=1}^N P_{k,j}^{(u)} \xi_j}{E_k^{(u)}} \\ \Sigma_k^{(u+1)} = \frac{\sum_{j=1}^N P_{k,j}^{(u)} (\xi_j - \mu_k^{(u+1)}) (\xi_j - \mu_k^{(u+1)})^T}{E_k^{(u)}} \end{cases} \quad (6)$$

通过 EM 进行不断的迭代学习直到收敛, 便可获得 GMM 参数的最优解 $\{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$. 为防止陷入局部最优, 可用 k 均值聚类算法对 GMM 的参数进行初始化.

在得到 GMM 参数之后, 给定任意输入均可利用 GMR 预测其对应输出的条件概率分布^[19]. 具体而言, 在给定输入 ξ_t 和第 k 个高斯分布的情况下, ξ_s 的条件概率也服从高斯分布, 即 $\xi_s | \xi_t, k \sim \mathcal{N}(\hat{\mu}_k, \hat{\Sigma}_k)$, 其中

$$\begin{cases} \hat{\mu}_k = \mu_k^s + \Sigma_k^{st} (\Sigma_k^{tt})^{-1} (\xi_t - \mu_k^t) \\ \hat{\Sigma}_k = \Sigma_k^{ss} - \Sigma_k^{st} (\Sigma_k^{tt})^{-1} \Sigma_k^{ts} \end{cases} \quad (7)$$

考虑整个 GMM, 则有

$$\xi_s | \xi_t \sim \sum_{k=1}^K h_k \mathcal{N}(\hat{\mu}_k, \hat{\Sigma}_k) \quad (8)$$

其中 h_k 满足

$$h_k = p(k | \xi_t) = \frac{\pi_k \mathcal{N}(\xi_t; \mu_k^t, \Sigma_k^{tt})}{\sum_{i=1}^K \pi_i \mathcal{N}(\xi_t; \mu_i^t, \Sigma_i^{tt})} \quad (9)$$

最后, 基于高斯分布的线性转换特性, 式(8)中 $\xi_s | \xi_t$ 的期望 $\hat{\mu}$ 和协方差 $\hat{\Sigma}$ 为

$$\begin{cases} \hat{\boldsymbol{\mu}} = \sum_{k=1}^K h_k \hat{\boldsymbol{\mu}}_k \\ \hat{\boldsymbol{\Sigma}} = \sum_{k=1}^K h_k^2 \hat{\boldsymbol{\Sigma}}_k \end{cases} \quad (10)$$

式中: $\hat{\boldsymbol{\mu}}$ 为 GMR 重构的轨迹输出点; $\hat{\boldsymbol{\Sigma}}$ 为相应的协方差。

综上所述,示教轨迹的数据经 GMM 编码学习后,再使用 GMR 进行数据重构,可以生成平滑的复现轨迹。该轨迹以均值和协方差的形式封装了示教行为的所有特征,因此可用于表征示教行为。图 3 以实例展示了本文的方法对多次示教行为的学习与复现过程,其中示教轨迹的样本点均以时间为输入, x 轴坐标为输出。首先,使用 DTW 对 3 次示教获得的 x 轴位置轨迹在时域上进行对齐,其结果如图 3(a) 所示。然后,使用 GMM 进行编码学习,并提取示教轨迹的特征,如图 3(b) 所示。这里用椭圆来描述 GMM 提取的示教轨迹特征,椭圆的形状取决于相应高斯分布的协方差矩阵,其面积大小表示轨迹空间位置的不确定度。最后,使用 GMR 输出复现轨迹,如图 3(c) 所示,其中曲线是复现轨迹,阴影部分对应复现轨迹的标准差。

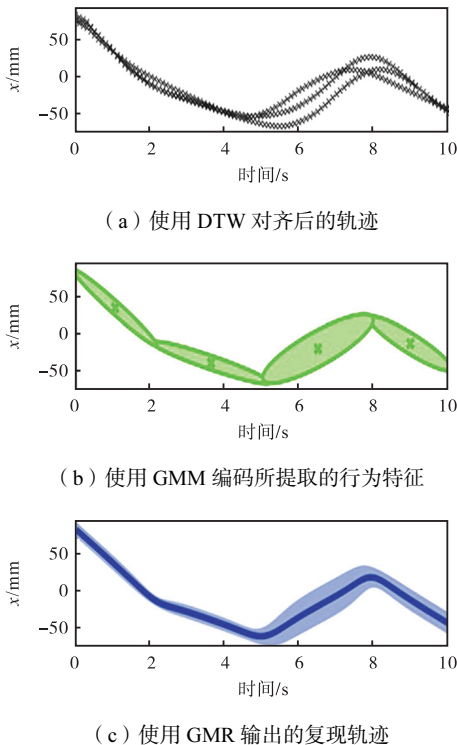


图 3 示教行为的学习与复现

Fig.3 Learning and reproduction of the demonstrated behavior

2.2.2 使用 DP 算法预估高斯分布数量

使用 GMM 模型进行聚类分析需要预先设定高

斯分布的数量 K 。值得注意的是, K 值选取不当可能导致模型欠拟合或过拟合情况的发生。目前常用的解决方式是尝试所有可能聚类个数的模型结构,然后根据一定准则选取其中最优的模型,例如贝叶斯信息准则 (Bayesian information criterion, BIC)^[20]。BIC 采用似然函数作为目标函数,通过计算最优的指标函数 S_{BIC} 来预估 K 值,即

$$S_{\text{BIC}} = -L(\xi_j) + \frac{n(K)}{2} \ln N \quad (11)$$

其中

$$\begin{cases} L(\xi_j) = \sum_{j=1}^N \ln(p(\xi_j)) \\ n(K) = (K-1) + K \left(D + \frac{1}{2} D(D+1) \right) \end{cases} \quad (12)$$

式中: $p(\xi_j)$ 是式 (2) 中定义的点 ξ_j 的概率; N 为示教点总数。

一般而言,用于训练 GMM 的示教轨迹数据点越多, GMR 复现轨迹的拟合效果越好。然而示教数据点越多,通过 BIC 获得的 K 值往往很大。这样不仅会使训练出的 GMM 模型更复杂,而且拟合的准确性欠佳。因此, K 值的选取应该在模型复杂度与模型对数据集表征能力之间寻求最佳平衡。

本文提出使用 DP 算法^[21]来预估训练 GMM 所需的高斯分布数量 K 。DP 算法是一种对曲线进行采样简化的方法,可以在从初始曲线中提取有限的关键点,然后以相邻关键点之间的连线来逼近拟合初始曲线,并且能够在一定程度上保持初始曲线的轮廓形状。本文利用 DP 算法对示教轨迹曲线的分段来确定相应的 K 值。该方法较为简单直观,所获得的 K 值不一定为最优解,但是相对合理。采用 DP 算法确定轨迹相应 K 值的步骤如下。

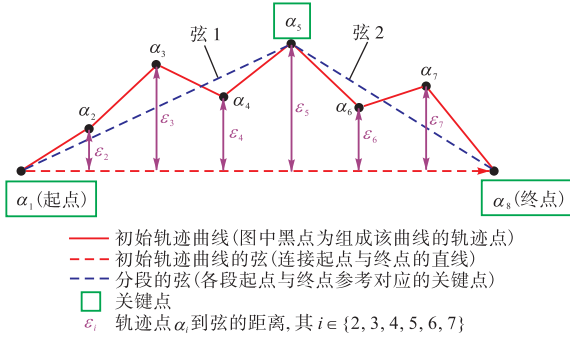
步骤 1 设置弦误差阈值为 ε_{th} 。

步骤 2 将当前轨迹曲线的起点与终点标记为关键点。以一条直线连接相邻的关键点,该直线即为当前曲线的弦,如图 4(a) 中的红色虚线所示。

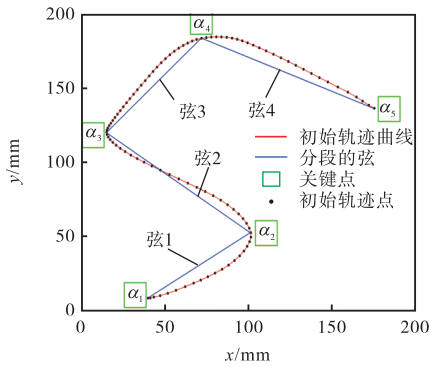
步骤 3 依次计算当前曲线上除了起点与终点外其余各轨迹点到弦的距离,即弦误差 ε ,并找出其中的最大值 ε_{max} 及其对应的点。图 4(a) 中 ε_{max} 为 ε_5 ,即轨迹点 α_5 到弦的距离。

步骤 4 如果 $\varepsilon_{\text{max}} < \varepsilon_{\text{th}}$,则当前曲线上除起点和终点外的轨迹点都可以被忽略;如果 $\varepsilon_{\text{max}} \geq \varepsilon_{\text{th}}$,则将 ε_{max} 对应的轨迹点标记为关键点,并以该点将当前曲线分成两段,返回步骤 2 重新连接各新分段的弦,如图 4(a) 中蓝色虚线所示。

步骤 5 对所有新的分段重复执行步骤 3、4,直



(a) 使用 DP 算法对曲线进行分段的原理



(b) 使用 DP 算法预估 K 值的示例

图 4 用 DP 算法预估 K 值的原理与示例说明

Fig.4 Principle and example illustration of predicting the K-value with the DP algorithm

到所有曲线分段的 ϵ_{\max} 均小于 ϵ_{th} .

步骤 6 曲线当前所有弦的数量即为预估的高斯分布数量 K .

如图 4(b) 所示, 设 $\epsilon_{th} = 10 \text{ mm}$, 由 DP 算法可得图示平面轨迹的弦共有 4 条, 即该轨迹可以用这 4 条直线段来逼近拟合, 因此预估用该轨迹数据点训练 GMM 所需的高斯分布个数 $K = 4$. 值得指出的是, 弦误差阈值 ϵ_{th} 的大小会直接影响 K 的取值. 具体而言, 当轨迹精度的要求不高时, 可以设置较大的 ϵ_{th} , 使 K 值减小.

2.2.3 使用 DP 算法对轨迹进行稀疏化并优化

为了提高示教编程的精度, 本文先采用 DTW 算法将多次示教的轨迹对齐, 再根据 DP 算法预估高斯分布个数, 由 GMM-GMR 从对齐后的轨迹中学习并输出复现轨迹. 虽然复现轨迹的数据点越多越有利于确保轨迹的整体精度, 但却不利于机器人对整条轨迹进行速度规划. 因此本文再次采用 DP 算法对复现轨迹进行稀疏化, 获得必要关键点, 在满足轨迹精度需求的前提下进一步优化复现轨迹.

DP 算法可以在保留初始轨迹轮廓形状的前提下, 尽可能地减少构成曲线的点, 这个过程也被称为轨迹的稀疏化. 假设由 GMR 生成的复现轨迹里表示

空间信息的序列为 $\{\hat{\mu}_n\}_{n=1}^N$, 其中 $\hat{\mu}_n$ 表示第 n 个数据点的空间位置坐标. 设置弦误差阈值为 ϵ_{th} , 按照第 2.2.2 节所述的 DP 算法步骤 1~5 从 $\{\hat{\mu}_n\}_{n=1}^N$ 提取出位置轨迹的关键点 $\{\hat{\mu}_j\}_{j=1}^J$, $J \leq N$.

以图 5 所示的平面复现轨迹为例, 展示了本文所提出的轨迹关键点提取策略. 图中红色曲线表示复现轨迹, 数据点 $\alpha_j, j \in \{1, 2, 3, 4, 5, 6, 7\}$ 是构成该轨迹的离散点, 其中 $\alpha_1, \alpha_4, \alpha_6$ 和 α_7 是通过 DP 算法提取出来的关键点. 最终轨迹由连接相邻关键点的直线组成, 如图 5 中浅蓝色的弦 1、弦 2 和弦 3 所示. 机器人末端执行器按照该轨迹运动从而复现示教行为.

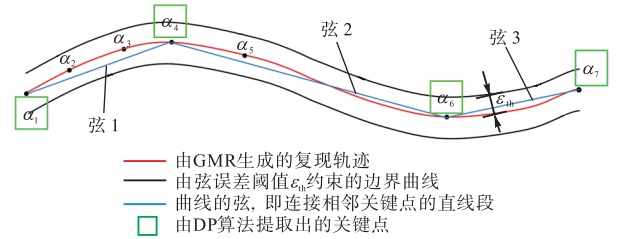


图 5 使用 DP 算法提取轨迹关键点示意

Fig.5 Illustration of extracting key points in the trajectory using the DP algorithm

此外, 考虑到机器人末端在运动到各关键点处其速度方向会发生瞬时改变, 因此为了防止速度突变, 本文通过圆弧过渡模型来对各关键点之间的直线轨迹进行优化.

假设 α_i 为最终轨迹的关键点, 在相邻直线轨迹之间设置半径为 R 的圆弧进行过渡, 由此在 α_i 处产生的位置误差为 β_{i-1} . 如果 β_{i-1} 小于误差阈值 β_{th} , 则对 α_i 处相邻直线轨迹进行圆弧过渡; 反之, 则不进行圆弧过渡. 图 6 显示了该方法对最终轨迹的优化效果, 其中 α_2 处的位置误差 $\beta_1 < \beta_{th}$, 因此在轨迹 1 和轨迹 2 之间设置半径为 R 的圆弧过渡, 同理在 $\alpha_3, \alpha_5, \alpha_6$ 和 α_7 各点处相邻的直线轨迹也是如此处理; α_4 处

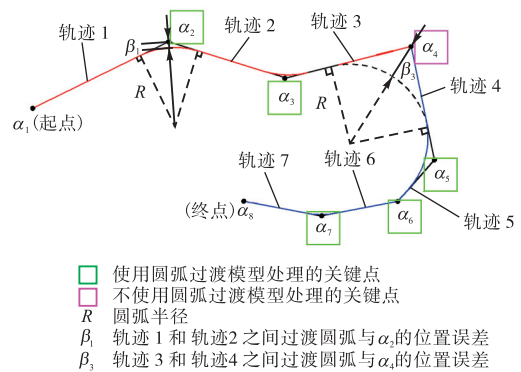


图 6 在轨迹关键点处进行圆弧过渡的示意

Fig.6 Illustration of the arc transition at key points in the trajectory

的位置误差 $\beta_3 > \beta_{th}$, 因此在轨迹 3 和轨迹 4 之间不进行圆弧过渡. 然而为了避免机器人末端的速度在点 α_4 处突变, 因此以该点作为边界点, 对图 6 中由轨迹 1、轨迹 2、轨迹 3 组成的红色轨迹和由轨迹 4、轨迹 5、轨迹 6、轨迹 7 组成的蓝色轨迹分别进行速度规划, 最终完成机器人对示教行为的复现.

3 实验

本文基于模拟焊接的实验场景对所提出学习策略进行验证, 具体执行流程如图 1 所示. 实验中, 使用 Leica-AT960MR 绝对激光跟踪仪采集示教数据; 由 MATLAB 软件完成 DTW 对齐多条示教轨迹、DP 算法确认高斯分布个数、GMM-GMR 编码输出复现轨迹以及 DP 算法提取轨迹关键点等计算工作, 然后将最终轨迹数据发送给 ABB IRB120 工业机器人. 最终由机器人控制其末端的焊枪复现学到的焊接轨迹. 为了验证该学习策略的性能, 分别将其用于模拟平面与空间焊接的场景.

3.1 平面焊接轨迹的学习与复现实验

图 7(a)、(b)、(c) 分别显示了 3 个具有不同形状焊接轨迹的平面工件. 工件被固定在工作平台, 其表

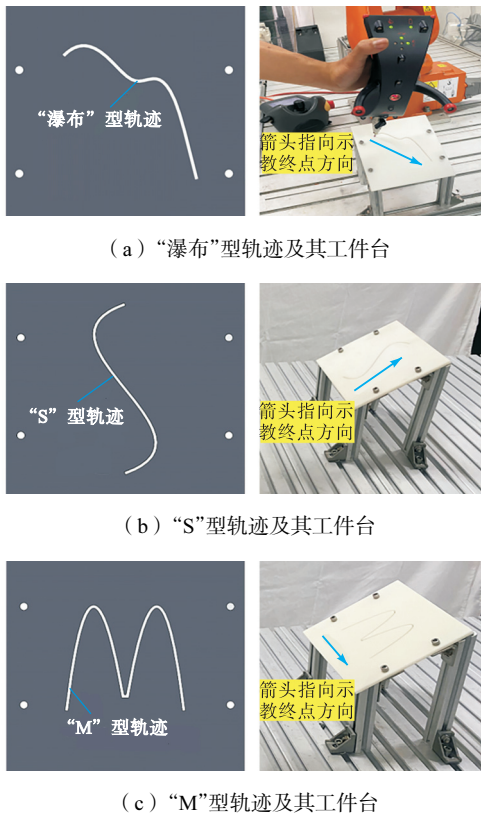


图 7 平面焊接轨迹的模拟实验场景

Fig.7 Simulated experimental scenarios for planar welding trajectories

面的镂空线槽即为对应的模拟焊接轨迹. 示教者手持 T-Probe 模拟焊枪, 使其末端探针沿着线槽轨迹移动, 同时绝对激光跟踪仪以 200 Hz 的频率记录其位置信息. 对 3 种不同的轨迹分别采集了 3 组数据, 如图 8 所示.

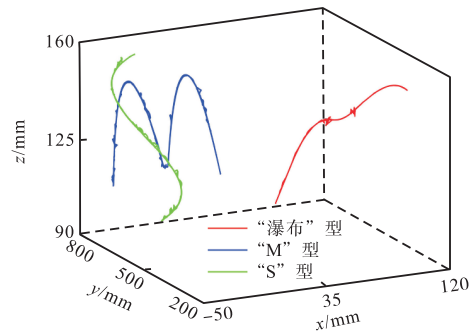


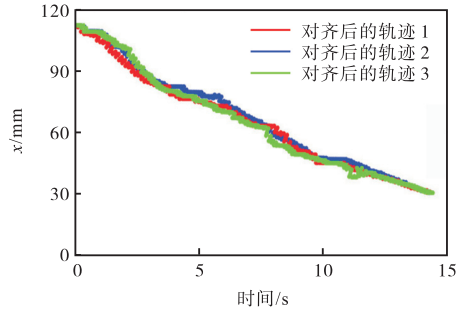
图 8 3 种不同类型的示教轨迹

Fig.8 Three different types of demonstrated trajectories

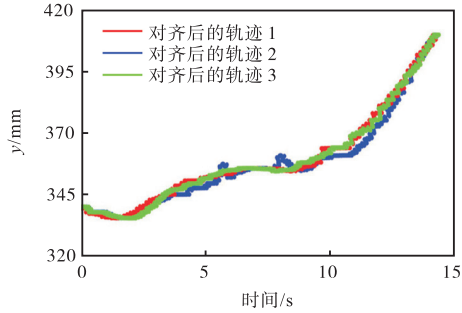
本文以“瀑布”型轨迹为例, 详述使用所提出的轨迹学习策略对其进行学习与复现的过程. 首先通过 DTW 算法将 3 组示教轨迹的数据在各维度的时间序列上进行对齐, 对齐后各组轨迹均包含 2 868 个数据点, 如图 9 所示.

使用 GMM 从对齐后的轨迹中提取示教特征的结果如图 10 中的椭圆所示. 然后, 通过 GMR 按时间间隔 0.01 s 输出组成复现轨迹的离散数据点, 共计 1 434 个. 图 11 显示了 GMR 的输出结果, 其中平滑曲线代表各维度上以时间为输入的位置复现轨迹, 阴影部分代表数据标准差的大小.

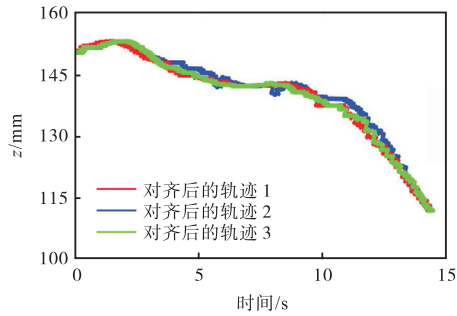
图 12 显示了“瀑布”、“S”和“M”型复现轨迹, 其中各自用于训练 GMM 的 K 值分别为 10、11 和 14. 值得指出的是, 弦误差阈值 ϵ_{th} 是使用 DP 算法预估 K 值的关键参数, 其取值可以参考轨迹的精度需求以及 GMM 的计算效率等因素. 表 1 以“瀑布”型轨迹为样本, 列举了不同的弦误差阈值 ϵ_{th} 与对应的预估 K 值、复现轨迹的精度以及训练 GMM 模型的时间, 其中 $\Delta d_{max} = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$ 表示复现轨迹点坐标与所设计模型轨迹对应点的最大位置误差, t 表示 GMM 迭代学习的时间 (电脑 CPU 为 Core i7-11700 2.5 GHz, 内存为 16 GB). 可以看出, 随着 ϵ_{th} 的减小, 由 DP 算法预估的 K 值以及对应训练 GMM 模型的迭代时间也在不断增加, 相应复现轨迹精度也得到了提升. 因此在轨迹精度需求不高的情况下, 可以通过适当增大弦误差阈值 ϵ_{th} 来预估 K 值. 本文实验中取 ϵ_{th} 为 0.5 mm 即可满足一般的焊接应用需求.



(a) x 轴轨迹对齐



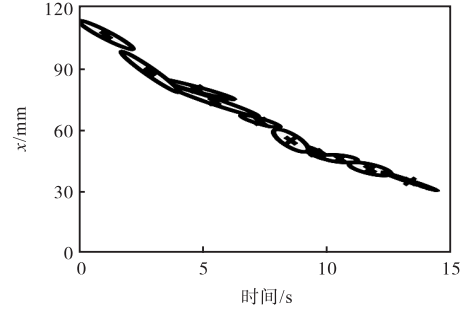
(b) y 轴轨迹对齐



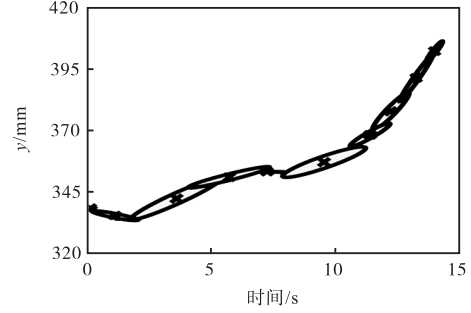
(c) z 轴轨迹对齐

图 9 基于 DTW 的“瀑布”型轨迹对齐

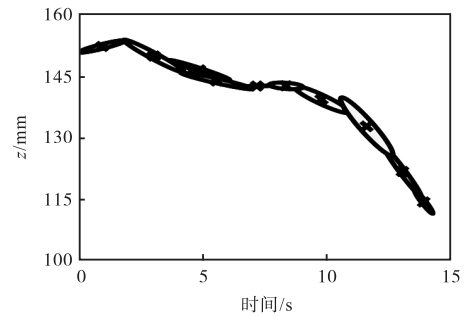
Fig.9 Alignment of the “waterfall” type demonstrated trajectories based on DTW



(a) x 轴轨迹表征



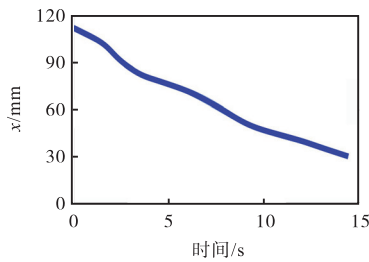
(b) y 轴轨迹表征



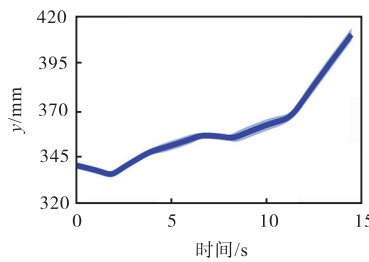
(c) z 轴轨迹表征

图 10 基于 GMM 的“瀑布”型轨迹表征

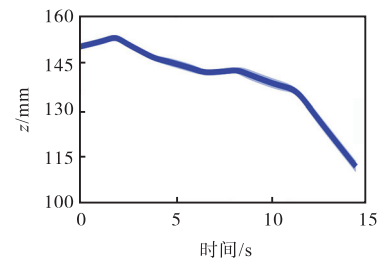
Fig.10 Representation of the “waterfall” type demonstrated trajectories based on GMM



(a) x 轴轨迹复现



(b) y 轴轨迹复现



(c) z 轴轨迹复现

图 11 基于 GMR 的“瀑布”型轨迹复现

Fig.11 Reproduction of the “waterfall” type demonstrated trajectories based on GMR

得到复现轨迹后,再次使用 DP 算法对其进行稀疏化,提取其中的关键点组成最终轨迹.考虑到提取轨迹关键点难免会影响轨迹精度,因此这里的弦误差阈值应比预估高斯分布个数时设置得更小.具体而言,设置弦误差阈值 $\epsilon_{th} = 0.1 \text{ mm}$. 同样以“瀑布”型

轨迹为例,提取的关键点如图 13(a)所示,共计 29 个数据点.从关键点的分布可以看出,DP 算法保留了复现轨迹的轮廓形状,具体表现为:在轮廓曲率较高部位,DP 算法提取的关键点较多,例如图 13(a)中蓝色虚线圈所示;在轮廓曲率较低的部位,所提取的关

键点较少,如图 13(a)中绿色线圈所示. 所有关键点的位置坐标如表 2 所示.

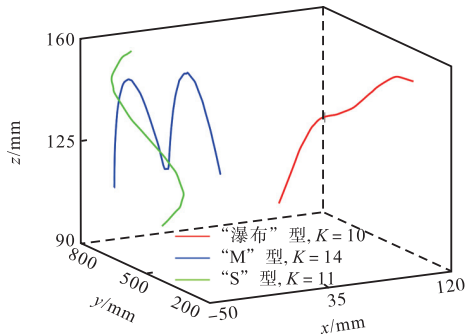


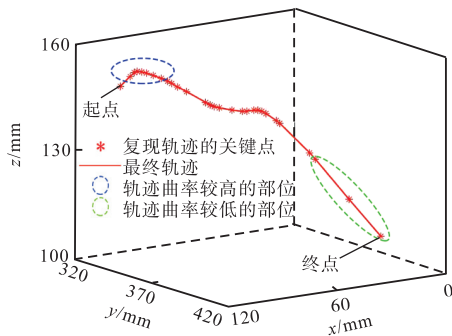
图 12 由 GMR 生成的 3 种不同类型的复现轨迹

Fig.12 Three different types of reproduction trajectories generated using GMR

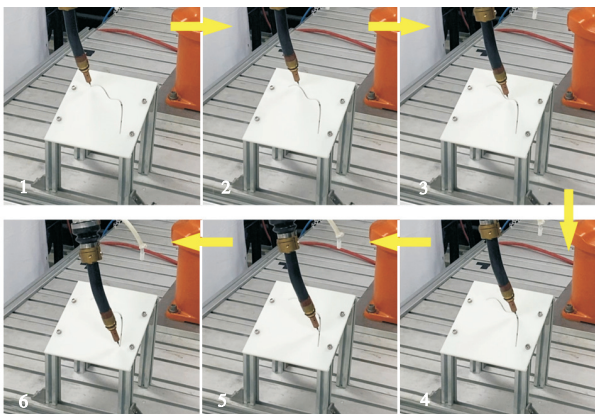
表 1 不同弦误差阈值的实验结果

Tab.1 Experimental results for different chordal error thresholds

$\epsilon_{th} / \text{mm}$	K	$\Delta d_{max} / \text{mm}$	t / s
0.3	22	0.466	6.552
0.4	14	0.504	0.858
0.5	10	0.568	0.439
0.7	8	0.832	0.584
1.0	6	1.156	0.148



(a) 由 DP 算法提取的“瀑布”型轨迹关键点



(b) 机器人执行“瀑布”型轨迹的模拟焊接任务

图 13 机器人根据“瀑布”型最终轨迹执行模拟焊接任务
Fig.13 Robot performs a simulated welding task following the “waterfall” type final trajectory

表 2 “瀑布”型复现轨迹的关键点坐标

Tab.2 Coordinates of the key points in the “waterfall” type reproduction trajectory

序号	x / mm	y / mm	z / mm
1	112.57	340.27	150.29
2	104.70	337.34	152.02
3	101.48	336.28	152.67
4	99.67	336.07	152.80
5	97.67	336.42	152.60
6	95.64	337.16	152.19
7	92.91	338.44	151.48
8	89.86	340.05	150.61
9	87.70	341.38	149.90
10	86.32	342.47	149.33
11	84.67	344.11	148.46
12	81.51	346.55	147.13
13	75.59	351.80	144.42
14	74.08	352.92	143.78
15	72.65	353.69	143.32
16	70.41	354.53	142.85
17	65.04	356.17	142.05
18	60.67	357.91	141.53
19	58.50	358.10	141.49
20	54.30	357.77	141.37
21	52.08	357.76	141.15
22	50.34	358.16	140.78
23	48.77	359.12	140.15
24	45.94	362.27	138.46
25	45.10	363.60	137.76
26	39.47	376.28	130.81
27	38.62	379.01	129.30
28	34.19	395.63	119.97
29	29.86	410.77	111.47

在复现平面轨迹焊接作业时,机器人焊枪末端按直线运动通过各关键点,同时采用第 2.2.3 节所述的圆弧过渡模型进行速度规划. 具体而言,设置圆弧半径 $R = 0.1 \text{mm}$, 误差阈值 $\beta_{th} = 0.1 \text{mm}$. 经计算各关键点圆弧过渡段的位置误差均小于 β_{th} , 因此可以将最终轨迹作为整体进行速度规划,即设置其始末点的速度和加速度均为 0, 机器人末端沿轨迹的运行速度均为 10mm/s . 最终, 机器人执行模拟“瀑布”型轨迹的焊接任务, 其过程如图 13(b) 所示.

此外, 以机器人示教器示教的信息作为标准值与示教编程的最终轨迹进行误差对比来验证精确性. 具体而言, 通过机器人示教器控制机器人末端到达该轨迹的各个位置, 每隔 2mm 记录下机器人末端的位置作为标准值; 将示教编程生成的最终轨迹同样每隔 2mm 计算得到插补点的数据作为比较值. 3 条轨迹的误差统计结果如表 3 和表 4 所示, 其中平均位置误差 $\sigma = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}$ 均在 0.500mm 以内, 最大位置误差 $\sigma^{\max} = \sqrt{(\sigma_x^{\max})^2 + (\sigma_y^{\max})^2 + (\sigma_z^{\max})^2}$ 均小于

0.800 mm.

表 3 3 种不同类型的最终轨迹的平均位置误差统计

Tab.3 Average positional error statistics for the three different types of final trajectories

类型	σ_x / mm	σ_y / mm	σ_z / mm	σ / mm
“瀑布”型	0.224	0.243	0.141	0.359
“S”型	0.212	0.223	0.137	0.337
“M”型	0.298	0.274	0.129	0.425

表 4 3 种不同类型的最终轨迹的最大位置误差统计

Tab.4 Maximum positional error statistics for the three different types of final trajectories

类型	σ_x^{\max} / mm	σ_y^{\max} / mm	σ_z^{\max} / mm	σ^{\max} / mm
“瀑布”型	0.423	0.496	0.347	0.739
“S”型	0.434	0.325	0.384	0.664
“M”型	0.517	0.384	0.326	0.722

3.2 空间焊接轨迹的学习与复现实验

图 14 显示了一个固定在工件台上用于模拟圆管与圆管垂直相贯焊缝的 3D 打印工件,其表面代表焊接轨迹的镂空线槽轮廓呈“马鞍”型. 示教者手持 T-Probe 模拟焊枪,使其末端探针从起点出发沿着线槽轨迹逆时针移动到终点. 注意该轨迹起点与终点位置重合.

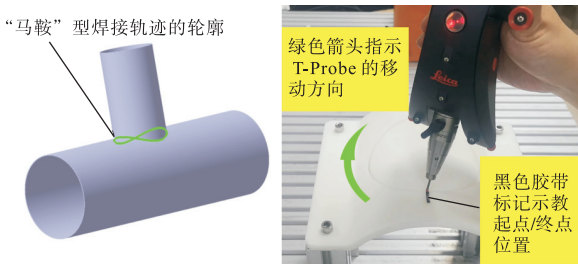
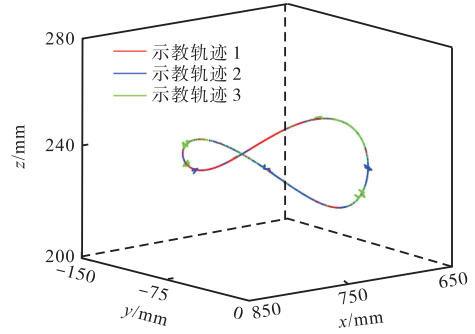


图 14 “马鞍”型焊接轨迹的模拟实验场景

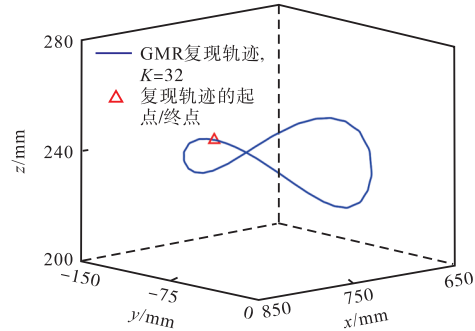
Fig.14 Simulated experimental scenario for the “saddle” type welding trajectories

模拟焊接的动作共执行了 3 遍,其间绝对激光跟踪仪以 200 Hz 的频率记录其位置信息,如图 15(a)所示. 使用基于 DTW-DP-GMM 的轨迹学习策略对示教数据进行处理,所得的“马鞍”型复现轨迹与最终轨迹分别如图 15(b)和 15(c)所示. 在使用 GMM 对示教数据进行建模前,设置弦误差阈值 $\epsilon_{th} = 0.5$ mm, DP 算法预估所需的高斯分布个数 $K = 32$; 在提取复现轨迹关键点时,设置弦误差阈值 $\epsilon_{th} = 0.1$ mm, DP 算法提取出的轨迹关键点共计 59 个,其位置坐标(部分)如表 5 所示.

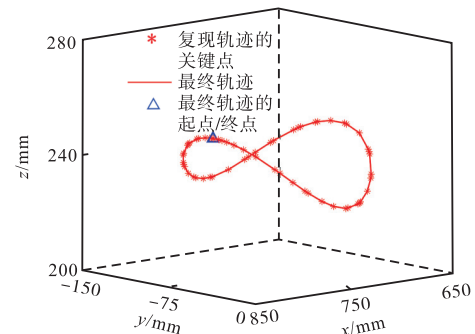
在复现焊接作业时,机器人焊枪末端按直线运动逼近各关键点,并采用第 2.2.3 节所述的圆弧模型过渡对最终轨迹进行速度规划,具体设置与第 3.1 节实验相同. 最终,机器人成功执行模拟“马鞍”型空间



(a) “马鞍”型空间模拟焊接轨迹的示教数据



(b) 使用基于 DTW-DP-GMM 的学习策略生成复现轨迹



(c) 使用基于 DTW-DP-GMM 的学习策略生成最终轨迹

图 15 使用本文所提出的学习策略获得“马鞍”型最终轨迹

Fig.15 Process of obtaining the “saddle” type final trajectory using the learning strategy proposed in this paper

表 5 “马鞍”型复现轨迹的部分关键点坐标

Tab.5 Coordinates of partial key points in the “saddle” type reproduction trajectories

序号	x/mm	y/mm	z/mm
1	809.86	-70.98	249.37
5	806.94	-87.30	247.74
10	792.10	-111.83	239.56
15	762.36	-128.12	229.12
20	729.33	-126.61	230.06
25	703.69	-109.82	239.96
30	689.67	-83.72	247.78
35	698.41	-32.83	242.93
40	724.25	-11.49	231.65
45	751.95	-6.38	227.64
50	788.21	-20.77	237.22
55	808.41	-54.19	248.42
59	809.86	-70.81	249.38

轨迹的焊接任务,其过程如图 16 所示.

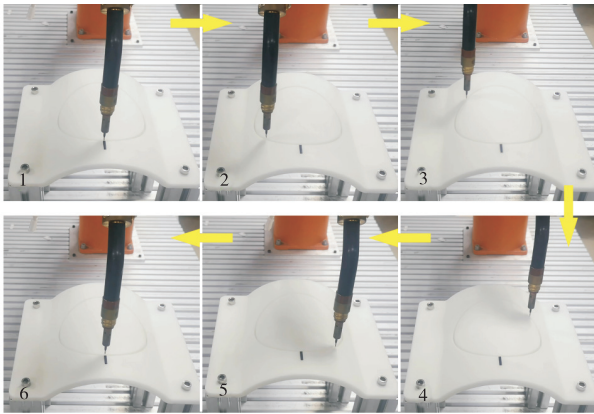


图 16 机器人根据“马鞍”型最终轨迹执行模拟焊接任务
 Fig.16 Robot performs a simulated welding task following the “saddle” type final trajectory

最后,采用与第 3.1 节相同的方法来验证最终轨迹的精确性,统计结果如表 6 和表 7 所示,其中平均位置误差 σ 小于 0.500 mm,最大位置误差 σ^{\max} 小于 0.800 mm.

表 6 “马鞍”型最终轨迹的平均位置误差统计

Tab.6 Average positional error statistics for the “saddle” type final trajectories

σ_x / mm	σ_y / mm	σ_z / mm	σ / mm
0.233	0.257	0.133	0.372

表 7 “马鞍”型最终轨迹的最大位置误差统计

Tab.7 Maximum positional error statistics for the “saddle” type final trajectories

σ_x^{\max} / mm	σ_y^{\max} / mm	σ_z^{\max} / mm	σ^{\max} / mm
0.445	0.542	0.245	0.743

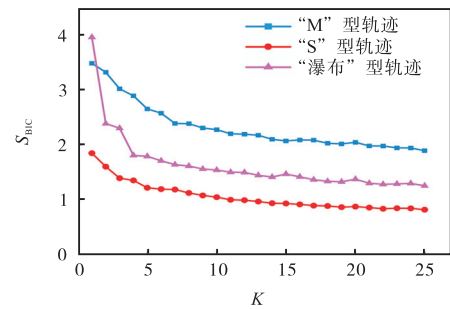
4 讨论

本节从适用性和计算效率等方面对本文提出的基于 DTW-DP-GMM 的工业机器人轨迹学习策略与现有方法进行比较.

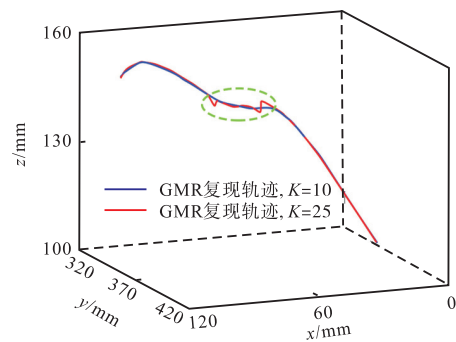
基于 GMM-GMR 的机器人示教编程算法可以有效地从人类演示的行为中学习并复现相应的轨迹.该方法通常需要从多条示教轨迹中学习才能准确地提取相应的行为特征,然而这些示教轨迹序列在时域长度上往往是不一致的,直接从这些示教样本中学习并生成的复现轨迹在精度上往往存在较大的误差.为了解决这个问题,本文提出的学习策略首先使用 DTW 算法将所有示教轨迹在时域上通过非线性变换对齐,尽量减小行为特征的不确定性,然后采用 DP 算法在弦误差阈值的约束下预估用于训练 GMM

模型所需的高斯分布个数 K ,可以使得 GMR 输出的复现轨迹满足位置精度需求.

值得指出的是,使用 BIC 算法预估 K 值时需要为其设置上限 K_{\max} ($K_{\max} \geq 1$),然后按照式 (11) 分别计算从 1 到 K_{\max} 的指标函数 S_{BIC} ,其中最小的 S_{BIC} 对应的 K 值即为最优值.以第 3.1 节实验中所学习的 3 种不同的平面轨迹为例分别计算具体 K 值对应的 S_{BIC} ,结果如图 17(a) 所示.可以看出, S_{BIC} 大致是随着 K 值的增大而减小的.虽然增加 K 值可以有效提升 GMM-GMR 复现轨迹的精度,然而 K 值过大却有可能保留示教行为中因人为抖动产生的轮廓特征,甚至存在过拟合的风险,如图 17(b) 中的绿色虚线圈所示.因此设置合适的 K_{\max} 是使用 BIC 算法预估 K 值的关键,然而这对于非专业的用户而言却是难以判断的.相比而言,本文所采用的 DP 算法无需用户判断参数 K 的取值范围.此外,表 8 列出了分别采用两种算法求解出相同 K 值所需的时间(以第 3.1 节实验的“瀑布”型轨迹为样本),可以看出在计算效率方面 DP 算法也优于 BIC 算法.



(a) 指标函数 S_{BIC} 与 K 值的关系



(b) 由不同 K 值得到的“瀑布”型复现轨迹

图 17 复现的位置轨迹与高斯分布个数 K 的关系

Fig.17 Relation between the reproduction trajectory and the number of Gaussian distributions K

除了被用于预估高斯分布的数量,在本文提出的学习策略中 DP 算法还被用于稀疏化 GMR 输出的复现轨迹以减少输出的数据点.虽然数据点越多越有

利于保证复现轨迹的精度, 但却不利于工业机器人对整条轨迹进行速度规划. 值得注意的是, 如果仅从减少数据点数量的目的考虑, 直接在 GMR 输出复现轨迹时增大相应输入时间的间隔就可以实现. 然而这种方法并没有考虑保留复现轨迹的几何轮廓, 输出的数据点过少可能会导致复现轨迹精度下降. 相比而言, DP 算法在尽可能减少了复现轨迹数据点数量的同时可以保留复现轨迹的轮廓形状, 其拟合精度可以通过弦误差阈值来约束. 第 3.1 节与第 3.2 节的实验均证明了由这些关键点组成的最终轨迹可以满足焊接轨迹的位置精度需求.

表 8 BIC 与 DP 算法在求解出 K 时所需的时间对比

Tab.8 Comparison of the time required for deriving the value of K between BIC and DP algorithms

K	BIC 算法求解时间/s	DP 算法求解时间/s
6	0.469	0.057
8	0.631	0.061
10	0.826	0.065
14	1.448	0.073
22	3.884	0.082

需要指出的是, 在本文的实验里机器人末端在移动过程中焊枪的姿态是不变的. 然而在实际的生产中, 为了保证焊接质量, 机器人经常需要实时调整末端焊枪的姿态以确保与被焊件之间合理的焊接角度. 这种情况下, 在规划末端位置轨迹的同时还应考虑相应末端姿态的变化, 即机器人需要同时学习示教行为的位置和姿态轨迹. 然而本文提出的基于 DTW-DP-GMM 的学习策略并不适用于需要学习并精确复现姿态轨迹的工作场景, 因为 DP 算法无法处理表征末端姿态的单位四元数数据, 因此无法从示教行为的姿态轨迹中求解出所需的高斯分布个数. 在未来的工作中, 需要进一步研究如何从示教行为的姿态数据中获得能够满足用户精度需求的高斯分布数量.

5 结 论

本文研究了使用 GMM-GMR 进行示教编程的方法, 针对高斯分布个数难以选择、复现轨迹精度较低等存在的问题, 提出了将 DTW、DP 算法与 GMM 相结合的机器人轨迹学习策略. 首先, 使用 DTW 将多条示教轨迹对齐, 尽可能地减少相关运动特征的不确定性. 其次, 采用 GMM 对示教行为的特征进行学习, 并用 GMR 输出复现轨迹, 从而减少人工示教带来的误差. 在这个过程中, 提出使用 DP 算法来预估 GMM 中高斯分布的个数. 最后, 对复现轨迹进行优化处理, 即在满足应用精度的前提下再次采用 DP 算

法提取复现轨迹的关键点组成最终轨迹, 并用圆弧过渡模型对其进行优化.

通过模拟平面与空间焊接轨迹的示教编程实验可以得到以下结论: ①在使用 GMM 对示教数据集建模之前, 使用 DTW 将多条示教轨迹进行对齐处理有利于降低复现轨迹的不确定性; ②与传统的 BIC 算法相比, DP 算法可以更加直观高效地预估合适的高斯分布个数; ③实验中机器人复现示教行为的最终轨迹的平均位置误差均在 0.500 mm 以内, 最大位置误差也不超过 0.800 mm, 因此所提出的策略生成的轨迹可以满足焊接实验的精度需求.

参考文献:

- [1] 马南峰, 姚锡凡, 陈飞翔, 等. 面向工业 5.0 的人本智造[J]. 机械工程学报, 2022, 58(18): 88-102.
Ma Nanfeng, Yao Xifan, Chen Feixiang, et al. Human-centric smart manufacturing for industry 5.0[J]. Journal of Mechanical Engineering, 2022, 58(18): 88-102(in Chinese).
- [2] 迟明善, 姚玉峰, 刘亚欣. 基于示教编程的共融机器人技能学习方法的研究进展[J]. 仪器仪表学报, 2020, 41(1): 71-83.
Chi Mingshan, Yao Yufeng, Liu Yaxin. Advances in coexisting-cooperative-cognitive robots skill learning approach based on learning from demonstration[J]. Chinese Journal of Scientific Instrument, 2020, 41(1): 71-83(in Chinese).
- [3] 黄艳龙, 徐 德, 谭 民. 机器人运动轨迹的模仿学习综述[J]. 自动化学报, 2022, 48(2): 315-334.
Huang Yanlong, Xu De, Tan Min. On imitation learning of robot movement trajectories: A survey[J]. Acta Automatica Sinica, 2022, 48(2): 315-334(in Chinese).
- [4] 曾 超, 杨辰光, 李 强, 等. 人-机器人技能传递研究进展[J]. 自动化学报, 2019, 45(10): 1813-1828.
Zeng Chao, Yang Chenguang, Li Qiang, et al. Research progress on human-robot skill transfer[J]. Acta Automatica Sinica, 2019, 45(10): 1813-1828(in Chinese).
- [5] Ijspeert A J, Nakanishi J, Hoffmann H, et al. Dynamical movement primitives: Learning attractor models for motor behaviors[J]. Neural Computation, 2013, 25(2): 328-373.
- [6] Englert P, Paraschos A, Deisenroth M P, et al. Probabilistic model-based imitation learning[J]. Adaptive Behavior, 2013, 21(5): 388-403.
- [7] Yu X, Liu P, He W, et al. Human-robot variable im-

- pedance skills transfer learning based on dynamic movement primitives[J]. *IEEE Robotics and Automation Letters*, 2022, 7(3): 6463-6470.
- [8] Wang N, Chen C, Nuovo A D. A Framework of hybrid force/motion skills learning for robots[J]. *IEEE Transactions on Cognitive and Developmental Systems*, 2021, 13(1): 162-170.
- [9] Mülling K, Kober J, Kroemer O, et al. Learning to select and generalize striking movements in robot table tennis[J]. *The International Journal of Robotics Research*, 2013, 32(3): 263-279.
- [10] Li X, Cheng H, Ji G, et al. Learning complex assembly skills from kinect based human robot interaction[C]//*IEEE International Conference on Robotics and Biomimetics*. Bali, Thailand, 2017: 2646-2651.
- [11] Asfour T, Gyarfas F, Azad P, et al. Imitation learning of dual-arm manipulation tasks in humanoid robots[C]//*2006 6th IEEE-RAS International Conference on Humanoid Robots*. Genova, Italy, 2006: 40-47.
- [12] Wu Y Q, Zhao F, Tao T, et al. A framework for autonomous impedance regulation of robots based on imitation learning and optimal control[J]. *IEEE Robotics and Automation Letters*, 2021, 6(1): 127-134.
- [13] Vakanski A, Mantegh I, Irish A, et al. Trajectory learning for robot programming by demonstration using hidden Markov model and dynamic time warping[J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2012, 42(4): 1039-1052.
- [14] Deng M D, Li Z J, Kang Y, et al. A learning-based hierarchical control scheme for an exoskeleton robot in human-robot cooperative manipulation[J]. *IEEE Transactions on Cybernetics*, 2020, 50(1): 112-125.
- [15] 刘成菊, 林立民, 刘明, 等. 一种基于 DTW-GMM 的机器人多机械臂多任务协同策略[J]. *自动化学报*, 2022, 48(9): 2187-2197.
- Liu Chengju, Lin Limin, Liu Ming, et al. A multi-task collaborative strategy for multi-arm robot based on DTW-GMM[J]. *Acta Automatica Sinica*, 2022, 48(9): 2187-2197 (in Chinese).
- [16] Muhlig M, Gienger M, Hellbach S, et al. Task-level imitation learning using variance-based movement optimization[C]//*2009 IEEE International Conference on Robotics and Automation*. Anchorage, USA, 2009: 1177-1184.
- [17] Huang Y, Caldwell D G. A linearly constrained non-parametric framework for imitation learning[C]//*International Conference on Robotics and Automation*. Paris, France, 2020: 4400-4406.
- [18] Bishop C M. *Pattern Recognition and Machine Learning*[M]. Singapore: Springer, 2006: 423-455.
- [19] Huang Y, Abu-Dakka F J, Silvério J, et al. Generalized orientation learning in robot task space[C]//*2019 International Conference on Robotics and Automation*. Montreal, Canada, 2019: 2531-2537.
- [20] Burnham K P, Anderson D R. Multimodel inference: Understanding AIC and BIC in model selection[J]. *Sociological Methods and Research*, 2004, 33(2): 261-304.
- [21] Li X C, Gao X F, Zhang W, et al. Smooth and collision-free trajectory generation in cluttered environments using cubic B-spline form[J]. *Mechanism and Machine Theory*, 2022, 169(3): 104606.

(责任编辑: 王晓燕)