

# Research on Somatosensory Arm Control System Based on Time Series Similarity

Shiqin Hu, Haiyun Yuan, Jinchi Zou, Yulei Shi, Xuemei Liu, Yuqin Meng

Southwest Petroleum University, Chengdu Sichuan  
Email: 1067459575@qq.com

Received: Sep. 9<sup>th</sup>, 2018; accepted: Sep. 23<sup>rd</sup>, 2018; published: Sep. 30<sup>th</sup>, 2018

---

## Abstract

In order to cope with the complex environment, this paper uses a combination of somatosensory technology and robot control technology and Kinect equipment to design a robotic control system based on somatosensory. In order to get closer to the results of the human arm, we used a six-degree-of-freedom series robot and used a 51 MCU as the main control board, and the Kinect Xbox One as a sensor. In order to make the robot arm imitate and follow the human movement, we used the Kinect sensor to collect human motion information without contact and convert it into control commands. The bone data acquisition module collects the coordinates of the bone joint points, calculates the joint angle and stores it in the Mysql database. As the main communication bridge, Mysql database combines the bone data acquisition module and the robot arm control module through the similarity algorithm. The acquisition platform enters joint angle data into the data inventory, and the control platform takes the joint angle data from the database. The control module calculates the difference between the joint angle of the current moment and the joint angle of the previous moment, and then matches it with the data in the database to identify the motion trajectory and automatically complete the remaining motions. It can also be understood that the robot arm can make a "pre-judgment action". The advantage of the DTW algorithm is that it can reduce the delay so that the robot arm follows the movement of the human arm more quickly. Experimental results show that the method can effectively reduce the time lag, smooth the original data and automatically filter out the abnormal data.

## Keywords

Mysql Database, Mechanical Arm, Somatosensory Control, Similarity Algorithm, DTW Algorithm

---

# 基于时间序列相似性的体感机械手臂控制系统研究

胡诗琴, 袁海云, 邹金池, 石俞磊, 刘雪梅, 孟蓑钦

西南石油大学, 四川 成都

文章引用: 胡诗琴, 袁海云, 邹金池, 石俞磊, 刘雪梅, 孟蓑钦. 基于时间序列相似性的体感机械手臂控制系统研究[J]. 动力系统与控制, 2018, 7(4): 287-297. DOI: 10.12677/dsc.2018.74033

## 摘要

为应对复杂多变的环境, 本文采用了体感技术和机器人控制技术与Kinect设备相结合的方法, 设计了一个基于体感的机械手臂控制系统。为了更接近于人体手臂的结果, 采用了六自由度的串联型机械手臂, 使用51单片机作为主控板, Kinect Xbox One作为传感器, 利用Kinect传感器无接触式采集人体动作信息并转化为控制指令, 让机械手臂模仿并跟随人体运动。骨骼数据采集模块采集骨骼关节点的坐标, 计算出关节角度并存入Mysql数据库中, Mysql数据库作为主要的通信桥梁, 通过相似性算法将骨骼数据采集模块和机械臂控制模块结合起来, 采集平台向数据库存入关节角度数据, 控制平台从数据库中取出关节角度数据。控制模块计算出当前时刻与前一时刻关节角度的差值, 然后与数据库中的数据匹配, 从而识别运动轨迹, 自动完成剩下的动作, 也可以理解为机械手臂能做出“预判动作”。DTW算法的优势在于减小延时, 使得机械手臂更加迅速地跟随人体手臂运动。实验结果证明该方法能有效地降低时滞性、平滑原始数据以及自动滤除异常数据。

## 关键词

Mysql数据库, 机械手臂, 体感控制, 相似性算法, DTW算法

Copyright © 2018 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

在国内, 体感技术正处于发展早期。通过对相关文献分析, 了解到基于光学感测的 Kinect 3D 摄像设备在机器人领域应用广泛, 主要集中在场景建模、视觉分析和姿态识别三个方面。其中, 来自上海交通大学的钱鹤庆学者使用 Kinect 采集人体骨骼关节点坐标, 并采用 DTW 算法进行动态姿势识别, 发明出虚拟地球仪[1]。清华大学的研究者将桌面作为机械臂的工作空间, 使用 Kinect 采集桌面上目标物的标点坐标, 然后根据坐标值, 采用 RRT 算法规划机械手臂的运动路径, 让机械臂自主抓取目标物并扔进垃圾桶, 从而实现简易桌面清理任务[2]。大连理工大学的裴岩明将体感技术与工业机器人技术结合, 设计了一个体感远程控制机械臂的系统[3], 利用 Kinect 捕获人体动作信息, 并将其作为控制指令, 远距离去控制机械手臂。重庆邮电大学的林海波完成了基于 Kinect 的景深数据采集[4], 通过系列计算, 得出操作者的腰、肩、肘和腕关节的转动角度, 利用串口通信, 将这些角度发送给机器人的单片机控制端, 控制四自由度的机械手臂完成抓取任务。中国民族大学的孙强和王文涛研制了一种基于体感遥控的全向移动机器人系统[5], 首先通过 Kinect 传感器获取人体深度图像数据, 然后利用骨骼跟踪技术计算关节点的三维坐标, 从而识别出动作, 并作为控制指令传输给机器人执行。

本文研究的是基于体感的机械手臂控制系统, 使用体感传感器 Kinect 采集人体动作信息, 然后通过 USB3.0 接口传递给计算机, 再让计算机进行一系列复杂运算和处理, 计算出人体手臂每一个关节转动的角度差。之后将这些角度差值转化为字符串型的控制指令, 通过蓝牙串口发射模块, 将控制指令发送给

单片机,让机械臂对应的舵机转动,最终完成体感控制机械臂的任务[6]。利用体感传感器 Kinect 对机械手臂实现直接实时控制,然后建立时间序列模板库,利用时间序列相似性,让采集到的运动序列和模板库中的模板序列进行实时匹配,找到最为相似的模板序列,并将模板序列转化为机械臂的控制指令。这样可以减少误动作的产生,最后使得机械手臂的控制更加平滑、自然。设计分为四个模块:骨骼数据采集模块(Visual C++编写)、机械手臂控制模块(Visual C#编写)、Mysql 数据库通信模块和时间序列相似性算法(MATLAB 编写)。而基于时间序列相似性算法中的一些常用算法有:模板匹配、动态规划、动态时间规整进行比较,实验结果显示,DTW 算法能有效地降低时滞性、平滑原始数据以及自动滤除异常数。

## 2. 相关工作

### 2.1. 骨骼数据采集模块(Visual C++编写)

#### 2.1.1. 骨骼追踪的实现

Kinect 传感器能够追踪人体全身的 25 个骨骼点,识别出人体的骨架模型,找到人体骨骼节点图像,算出骨骼特征点的坐标是骨骼数据采集平台的核心工作。本文在 VS 2013 (Visual Studio 2013)的开发环境下,使用 C++编程语言与 Opencv 开发库来完成骨骼跟踪。

从深度图像信息中提取出骨骼数据,首先要获得深度图像,然后通过计算机的图形视觉处理技术,获得人体图像,最后得出由骨骼关节点构成的人体骨架图,如图 1 所示。

Skeleton Stream (骨骼数据流)、Skeleton Frame (骨骼框架)、Skeleton (骨骼)和 Joint (特征点)是 Kinect 的骨骼追踪系统中的四个主要对象模型。其中 Skeleton Frame 包含了人体的骨架信息。Joint 共有 ME25 个骨骼特征点。每一个 Joint 都具有三种属性,如表 1 所示。

其中,Joint 中的 Joint Tracking State 具有三种跟踪状态属性,如表 2 所示。

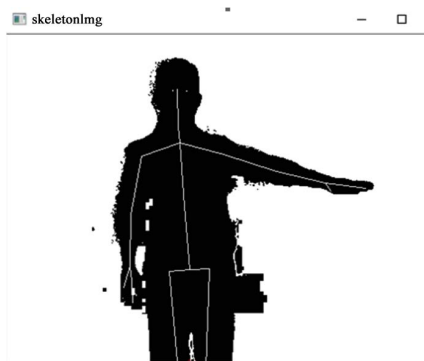


Figure 1. Human skeleton diagram  
图 1. 人体骨架图

Table 1. Three properties of the joint object

表 1. Joint 对象的三种属性

Joint 对象的三种属性	定义
Joint Type 属性	这是 Joint 的一种枚举类型
Position 属性	Position 的 X、Y、Z 表示骨骼点的三维坐标,以米作为单位。使用右手坐标系可以知道 X、Y、Z 的位置关系,以 Kinect 为中心原点, X 轴向左代表正方向,向右为负, Y 轴向上为正方向,向下为负, Z 值表示人距 Kinect 的距离,越靠近 Kinect, Z 值越小。
Joint Tracking State 属性	包含骨骼特征点的跟踪情况。跟踪情况有三种状态: Tracked (已跟踪)、NotTracked (未跟踪)、Inferred (推测而来)。

**Table 2.** Attribute characteristics of Joint Tracking State  
**表 2.** Joint Tracking State 的属性特征

Joint Tracking State 属性字段	含义
Inferred	骨骼追踪在深度图像数据帧中查看不到相应的关节点，但是可以计算出相对应的关节点位置
Not Tracked	关节点的位置无法确定，且位置为 0
Tracked	关节点能够被检测，且一直被追踪

当 Kinect 追踪骨骼特征点时，骨骼的姿势有可能会被遮住，然后导致追踪效果不好。因此，在处理骨骼数据时，首先要进行判断，只有当 Joint Tracking State 等于 Tracked 的骨骼点时才能进行处理，这样就可以除去有干扰的骨骼信息。

通过上述分析，骨骼数据采集流程图如图 2，获取的骨骼数据采集图如图 3。

### 2.1.2. 关节角度的计算

本文采用平面几何计算方法，计算骨骼关节点夹角时，两个关节点的连线与水平面的夹角  $\theta$ ，计算方式如图 4， $\theta$  的结果如公式(2.1)。

因此，角度  $\theta = (2.1)$

本文采用的六自由度机械手臂有六个舵机共同控制机械臂的运动。考虑到人体手臂一只手只能控制机械臂的 5 个自由度，那么需要另外一只手来完成最后一个自由度的控制。人体双臂的控制动作与机械臂动作的对应关系如下表 3。

## 2.2. 机械手臂控制模块(Visual C#编写)

在 Visual Studio 2013 环境下使用 C#搭建机械手臂控制平台并编写上位机软件，如图 5。

机械手臂控制平台将骨骼数据采集平台所采集的角度数据转化为舵机的控制信号，然后控制机械臂运动，舵机编号如图 6。其中上位机发送给 51 单片机的是机械臂的控制指令(即舵机的控制指令)，由于舵机采用 PWM 脉宽调整信号控制，控制指令包括 PWM 值的信息。经过测量计算，关节角度的改变值转换成 PWM 值时，270°舵机每转动 1°，PWM 值改变 7.4；180°舵机每转动 1°，PWM 值改变 11.11。例如原来的 PWM 值为 1400，关节角度变化 10°，则 PWM 变为了  $1500 + 7.4 * 10 = 1474$ 。

综上所述，从 Kinect 传感器到机械手臂控制的设计框图如图 7 所示。

## 2.3. 动态时间规整算法

DTW 算法主要是基于动态规划(Dynamic Programming)的思想，它是通过动态归整来对齐时间轴，使得每个数据点都有着最好的对齐方式，从而消除对不齐造成的误差，以此来获得时间序列间的最优匹配[7]。如果通过时间变化的运动时间序列与时间序列模板库中的模板序列有最大程度的一致性，那么就可以将其中的模板序列去作为一种控制指令去控制机械臂运动。其中 DTW 算法的编写是用 MATLAB 这个软件完成的，并且 MATLAB 还需要完成多种任务：完成 DTW 的编写；实现与 Mysql 数据库之间的通信与联系；不断去访问实时数据表，通过其中的比较，将其用 DTW 算法和时间序列模板库中的模板序列进行相似性判断。

DTW 算法实现的原理如下：

假定存入模板库中的运动序列组为一个参考模板，可以表示为  $R = \{R(1), R(2), R(3), \dots, R(M)\}$ ，其中， $R(1), R(2), R(3), \dots, R(M)$  分别表示一个  $n$  维的时间序列， $M$  为时间序列的个数[8]。实时的动作过程看作是测试模板，表示为  $T = \{T(1), T(2), T(3), \dots, T(N)\}$ ，其中， $T(1), T(2), T(3), \dots, T(N)$  分别表示一个  $n$

维的时间序列， $N$ 为时间序列的个数。

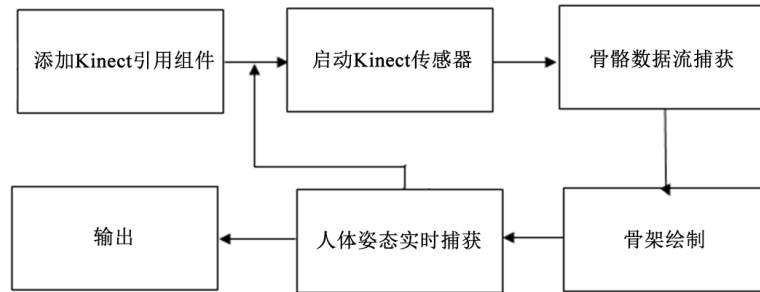


Figure 2. Bone data collection flow chart  
图 2. 骨骼数据采集流程图

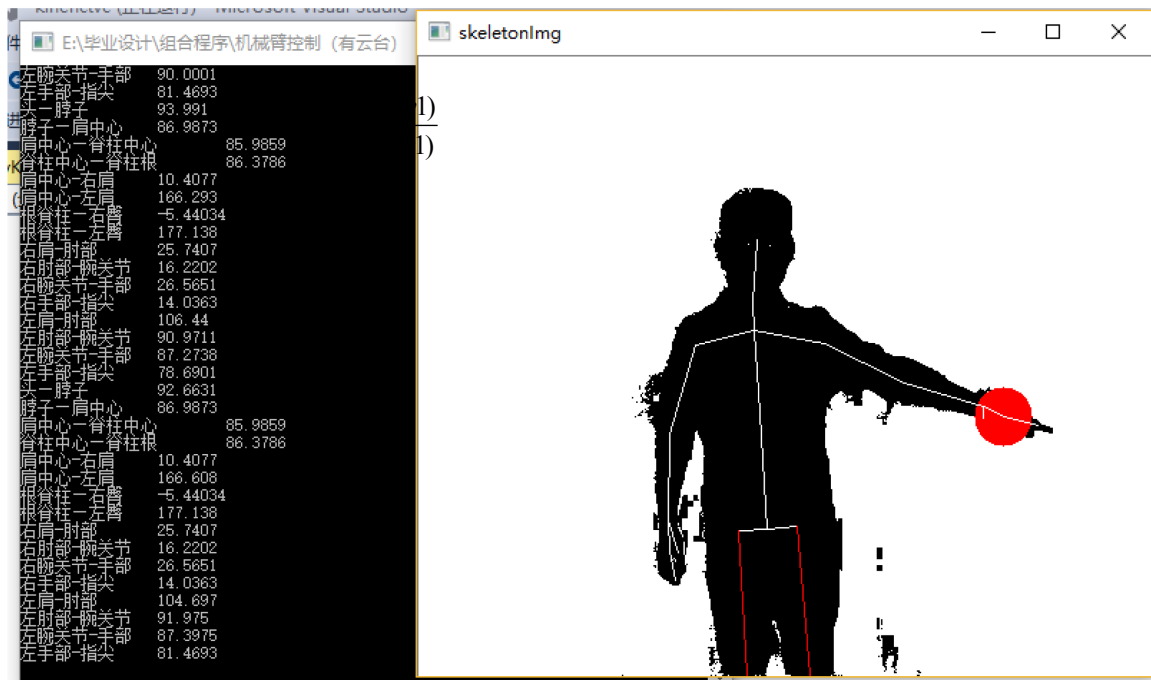


Figure 3. Bone data acquisition diagram  
图 3. 骨骼数据采集图

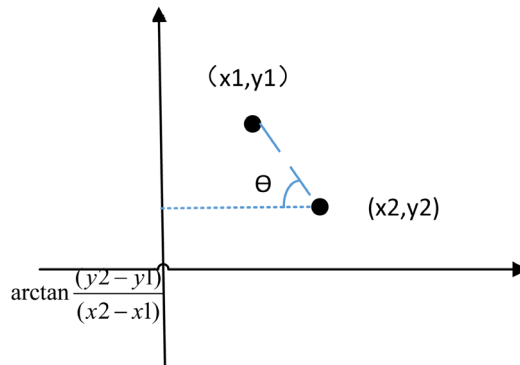


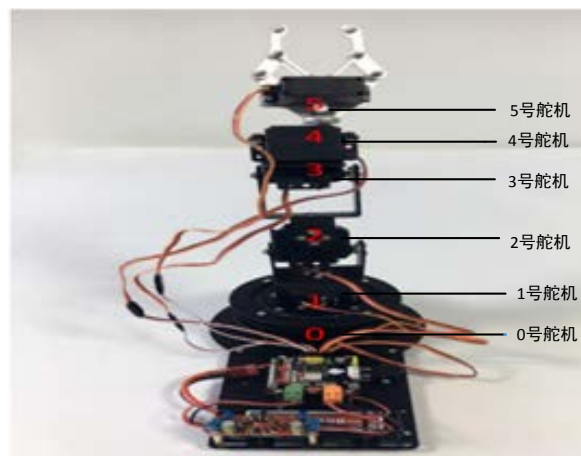
Figure 4. Calculating the angle between two joint points  
图 4. 计算两个关节点之间的夹角  $\theta$

**Table 3.** Two-arm control arm rotation  
**表 3.** 双臂控制机械臂转动

舵机编号	机械臂转动部位	双臂控制动作与机械臂动作的对应关系
0 号	云台的左转	左肘部往上移动——云台左转 左肘部往下移动——云台右转
1 号	大臂的前后移动	脖子到右臂肩关节的夹角变化来控制大臂的前后移动
2 号	小臂的前后移动	左臂肩部往下——小臂往后 左臂肩部往上——小臂往前
3 号	腕关节的前后移动	右臂肘部往上——腕关节往前 右臂肘部往下——腕关节往后
4 号	爪子的左右转	右臂腕关节往下——爪子右转 右臂腕关节往上——爪子左转
5 号	爪子的张合	右手握成拳头——爪子闭合 右手张开——爪子张开
/	机械臂开始动作	举起左手
/	机械臂停止动作	左手握成拳头



**Figure 5.** PC interface  
**图 5.** 上位机界面



**Figure 6.** Numbering diagram of the steering gear  
**图 6.** 舵机的编号图

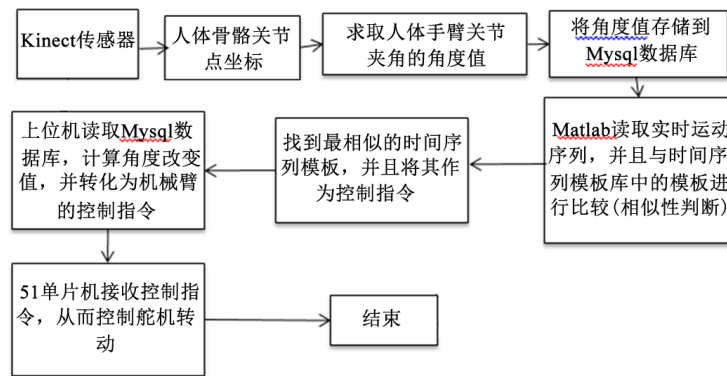


Figure 7. Kinect sensor to robot control design block diagram  
图 7. Kinect 传感器到机械手臂控制的设计框图

计算参考模板和测试模板之间的直接距离, 可以得出它们之间的相似度。两个模板之间的距离越短, 可以得出两个模板之间的相似度也越高。根据 DTW 的边界条件, 最短距离也就是  $d = (0, 0)$  到  $d = (m, n)$  之间的最短路径如图 8。再结合 DTW 算法的单调性和连续性要求, 设定局部路径约束条件为: 当从一个方格  $d(i-1, j-1)$ 、 $d(i-1, j)$  或者  $d(i, j-1)$  中到下一个方格  $d(i, j)$ , 横向或竖向的距离为  $d(i, j)$ , 斜着对角线过来的则是  $2d(i, j)$ 。数学表达式(2.2)可以清楚地解释这种约束条件, 其中  $g(i, j)$  表示 2 个模板都从  $g(0, 0)$  点到  $g(m, n)$  逐次匹配时两个模板之间的距离, 并且要在上一次匹配的基础上加  $d(i, j)$  或者  $2d(i, j)$ , 然后取最小值。通过比较模板之间的直接距离, 可以得出模板之间的相似程度。模板之间的直接距离愈来愈短, 相似程度就愈来愈高。

$$g(ck) = g(i_k, j_k) = g(i, j) = \min \begin{cases} g(i-1, j) + d(i, j) \\ g(i, j-1) + 2d(i, j) \\ g(i, j-1) + d(i, j) \end{cases} \quad (2.2)$$

### 2.4. Uniform Scaling 算法

索引和匹配运动流的一个主要困难(此后可与“时间序列”互换使用)是人体运动速度的可变性[9]。例如, 演员可以执行快速或慢速打击, 而这种可变性可以表现为均匀缩放, 时间序列的全局拉伸或收缩(即相对于时间轴)。DTW 优于欧几里德距离的优势越来越明显, 但是对于均匀缩放不变的相似性搜索的需求还不是很清楚。使用 DTW (图 9 左), 我们无法实现 2 次动作和 3 次动作之间的直观对齐, 即使序列恰好是相同的长度。然而, 通过拉伸底部序列的均匀缩放可以实现有意义的对齐。

为了创建用于均匀缩放的下界限距离度量, 我们将生成边界包络。引入了边界包络以降低 DTW, 虽然这里的原理是相同的, 但是包络的定义是非常不同的。特别是, 我们创建了两个序列 U 和 L:

$$U_i = \max(c \lfloor (i-1) * m/n \rfloor + 1, \dots, c \lfloor i * m/n \rfloor)$$

$$L_i = \max(c \lfloor (i-1) * m/n \rfloor + 1, \dots, c \lfloor i * m/n \rfloor)$$

可以将这些序列可视化为界定时间序列 C 的前 n 个点。再引入下边界函数  $LB\_Keogh$ :

$$LB\_Keogh(Q, C) = \sum_{i=1}^n \begin{cases} (q_i - U_i)^2 & \text{if } q_i > U_i \\ (q_i - L_i)^2 & \text{if } q_i < L_i \\ 0 & \text{otherwise} \end{cases}$$

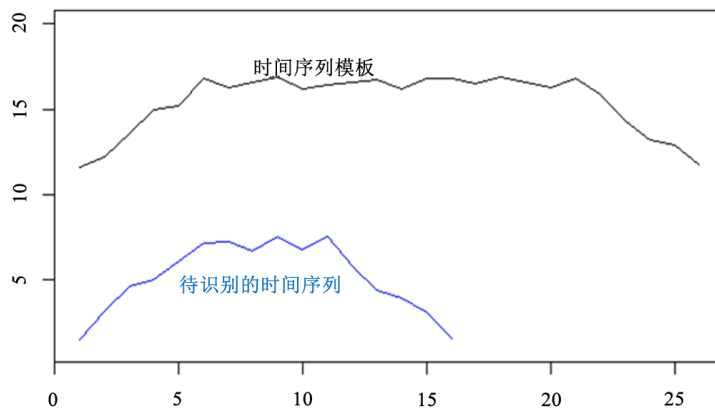


Figure 8. Two time series

图 8. 两条时间序列

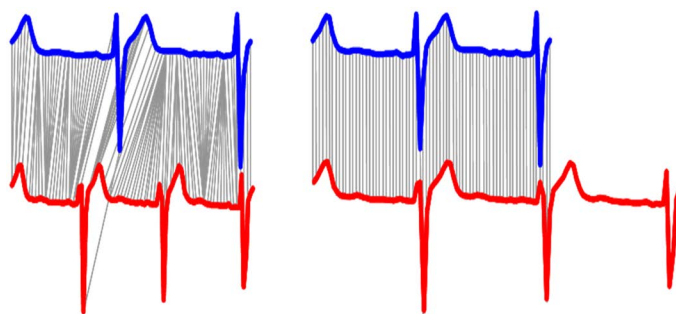


Figure 9. A visual contrast of DTW (left) and Uniform Scaling (right)

图 9. DTW (左)和 Uniform Scaling (右)的视觉对比

它可以降低  $Q$  和  $C$  之间距离的任何比例因子, 将  $U$  和  $L$  可视化为分段常数函数, 它们分别与  $U$  和  $L$  相交而不相交(图 10)。

在这种情况下, 只需计算 DTW 变换的上下边界包络, 以便在均匀缩放找到最佳全局缩放后允许对时间轴进行小的局部调整。

### 3. 实验

#### 3.1. 实验数据集-Mysql 数据库的建立

骨骼数据采集平台和机械手臂控制平台分别在不同的编译环境中完成, 要实现两者的通信, 则需要 Mysql 数据库建立起桥梁。骨骼数据采集平台不断向数据库存入关节角度数据, 机械手臂控制平台也实时从数据库中取出关节角度数据。操纵数据库往往会借助于数据表, Mysql 有一个超强的数据库管理工具 Navicat for Mysql, 数据库管理员可以用其来建立数据表, 完成数据库与其他操作平台间的通信。数据表的各个字段类型如图 11 所示。

使用 Navicat for Mysql 建立的实时运动时间序列表(skeletonangle\_yuntai)如图 12 所示。

使用 Navicat for Mysql 建立的运动时间序列模板库(skeletonangle\_yuntai)如图 13 所示。

#### 3.2. 算法数据对比分析

基于已获得的模板库角度数据, 我们选择两种修改方式对实时动作进行模拟并与标准模板库进行对比。根据每个模板库长度选择 5 组数据进行变动, 变化过程逐渐积累。每变动一次数据, 使用 MATLAB



选择 DTW 和 UNIFORM SCALING 两种算法进行仿真。针对实际活动中可能出现的动作局部延时与动作缺失两种情况进行了模拟。1) 局部动作延时: 逐次修改数据, 使修改的 10 个动作完全相同以模拟动作的延时; 2) 动作缺失: 逐次删除选中的数据, 减少数据序列长度以模拟动作缺失。

实验步骤:

对每一个标准模板库按照实验说明的方案进行改动, 使用 MATLAB 选择实时运动库进行两种不同算法对比。

算法程序流程图如图 14 所示。

针对 12 个标准模板库进行试验, 得到以下数据(表 4, 表 5)。

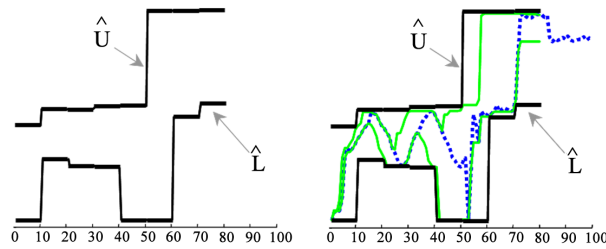


Figure 10. Visualize  $\hat{U}$  and  $\hat{L}$  as the piecewise

图 10. 将  $\hat{U}$  和  $\hat{L}$  视为分段

栏位	索引	外键	触发器	选项	注释	SQL预览
名					类型	长度 小数点 允许空值(
ID					int	6 0 <input type="checkbox"/> 1
movement_Time					datetime	6 0 <input type="checkbox"/>
ShoulderCenter_ShoulferRight					double	6 2 <input type="checkbox"/>
ShoulderRight_ElbowRight					double	6 2 <input type="checkbox"/>
ElbowRight_WristRight					double	6 2 <input type="checkbox"/>
WristRight_HandRight					double	6 2 <input type="checkbox"/>
HandRight_HandTipRight					double	6 2 <input type="checkbox"/>
ElbowLeft_WristLeft					double	6 2 <input type="checkbox"/>

Figure 11. Data field types

图 11. 数据表的各个字段类型

ID	movement_Time	ShoulderCenter	ShoulderRight	ElbowRight	Wristf	WristRight_HandR	HandRight_Hanc	ElbowLeft	Wrist
6192	2018-05-17 19:44:57.000000	8.75	19.54	-82.88	-90	500	0		
6193	2018-05-17 19:44:57.000000	8.75	19.25	-85.33	-90	500	0		
6194	2018-05-17 19:44:57.000000	8.75	18.71	-86.5	-87.88	500	0		
6195	2018-05-17 19:44:58.000000	8.75	17.88	-86.42	-87.88	500	0		
6196	2018-05-17 19:44:58.000000	8.75	17.61	-87.61	-87.88	500	0		
6197	2018-05-17 19:44:58.000000	7.31	17.61	-87.61	-87.8	500	0		
6198	2018-05-17 19:44:58.000000	7.31	17.61	-87.61	-85.6	500	0		
6199	2018-05-17 19:44:58.000000	7.31	17.61	-87.61	-83.42	500	0		
6200	2018-05-17 19:44:58.000000	7.5	18.43	-87.66	-83.42	500	0		
6201	2018-05-17 19:44:58.000000	7.5	18.43	-86.5	-85.6	500	0		
6202	2018-05-17 19:44:58.000000	8.75	17.88	-86.5	-83.42	500	0		
6203	2018-05-17 19:44:59.000000	8.75	17.61	-87.66	-85.6	500	0		
6204	2018-05-17 19:44:59.000000	8.75	17.61	-87.66	-84.81	500	0		
6205	2018-05-17 19:44:59.000000	8.97	17.61	-87.66	-84.81	500	0		
6206	2018-05-17 19:44:59.000000	8.97	17.61	-86.5	-84.81	500	0		
6207	2018-05-17 19:44:59.000000	7.5	18.43	-86.42	-85.03	500	0		
6208	2018-05-17 19:44:59.000000	7.13	19.54	-87.66	-84.81	500	0		
6209	2018-05-17 19:44:59.000000	7.13	19.25	-88.83	-82.23	500	0		
6210	2018-05-17 19:44:59.000000	0	0	0	0	0	0		

Figure 12. Real-time motion time series table

图 12. 实时运动时间序列表

ID	movement	T	ShoulderCenter_ShoulderRi	ShoulderRight_Elbow	ElbowRight_WristRig	WristRight_HandRig	HandRight_HandTipRig	ElbowLeft_WristLeft
24774	2018-05-17	1	13.5	4.51	3.97	-5.44	500	0
24775	2018-05-17	1	12.99	4.64	3.92	-6.01	500	0
24776	2018-05-17	1	15.15	3.67	3.97	-5.71	500	0
24777	2018-05-17	1	14.86	3.76	3.87	-6.34	500	0
24778	2018-05-17	1	14.57	3.81	4.64	-6.34	500	0
24779	2018-05-17	1	15.15	3.72	4.76	-6.01	500	0
24780	2018-05-17	1	13.5	4.57	4.83	-8.53	500	0
24781	2018-05-17	1	15.15	3.76	4.76	-12.53	500	0
24782	2018-05-17	1	15.15	3.76	4.27	-2.49	500	0
24783	2018-05-17	1	14.86	3.81	6.44	10.89	500	0
24784	2018-05-17	1	14.32	5.26	9.46	17.65	500	0
24785	2018-05-17	1	14.62	6.67	12.72	23.2	500	0
24786	2018-05-17	1	14.04	6.84	15.95	25.56	500	0
24787	2018-05-17	1	12.65	8.57	17.82	29.74	500	0
24788	2018-05-17	1	12.41	11	20.67	31.43	500	0
24789	2018-05-17	1	12.17	12.7	22.54	35.84	500	0
24790	2018-05-17	1	13.76	14.04	23.86	40.6	500	0
24791	2018-05-17	1	13.5	16.7	29.25	40.6	500	0
24792	2018-05-17	1	13.76	18.92	29.74	52.13	500	0

Figure 13. Motion time series template library

图 13. 运动时间序列模板库

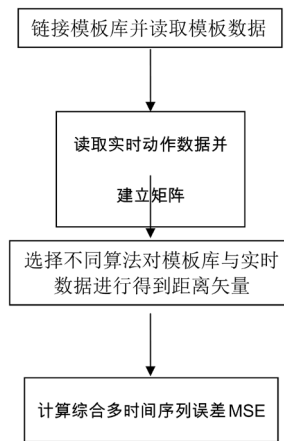


Figure 14. Algorithm program flow chart

图 14. 算法程序流程图

Table 4. Delay time series

表 4. 时间序列延时

	DTW 算法	UNIFORM SCALING 算法
模板匹配正确率 (正确匹配数/总实验次数)	59/60	57/60
误差(同等条件下两种算法所得结果更优 次数/总实验次数)	60/60	0/60

Table 5. The missing action

表 5. 动作缺失

	DTW 算法	UNIFORM SCALING 算法
模板匹配正确率 (正确匹配数/总实验次数)	58/60	57/60
误差(同等条件下两种算法所得结果更优 次数/总实验次数)	51/60	9/60

## 4. 结论

通过上述实验得出以下结论:

1) 在使用同种算法情况下, 实时模板与动作模板库差别越小, 计算所得的综合多时间序列误差 MSE 越小, 实时模板与动作模板库差别越大, 计算所得的综合多时间序列误差 MSE 越大。

2) DTW 算法和 UNIFORM SCALING 算法对时间序列延时的误差计算均要优于对动作缺失的误差计算。

3) 在动作时间序列有延时和模板动作缺失这两种情况下, 模板匹配正确率(正确匹配数/总实验次数)和误差(同等条件下两种算法所得结果更优次数/总实验次数) DTW 算法都得到了比 UNIFORM SCALING 算法更小的 MES 误差。由此可见, 在这两种实时动作下, DTW 算法都表现出了比 UNIFORM SCALING 算法更佳的性能。

## 参考文献

- [1] 钱鹤庆. 应用 Kinect 与手势识别的增强现实教育辅助系统[D]: [硕士学位论文]. 上海: 上海交通大学, 2011.
- [2] 韩峥, 刘华平, 黄文炳, 等. 基于 Kinect 的机械臂目标抓取[J]. 智能系统学报, 2013, 8(2): 149-155.
- [3] 裴岩明. 基于 Kinect 的远程机械臂体感控制系统研究[D]: [硕士学位论文]. 大连: 大连理工大学, 2013: 21.
- [4] 林海波, 梅为林, 张毅, 等. 基于 Kinect 骨骼信息的机械臂体感交互系统的设计与实现[J]. 计算机应用及软件, 2013, 30(2): 157-160.
- [5] 孙强, 王文涛. 基于体感遥控的全向移动机器人的系统设计[A]. 电子技术应用, 2015, 41(6): 157-160.
- [6] 王宇飞. 基于 Kinect 骨骼关节信息的视角不变步态识别方法[D]: [硕士学位论文]. 山东: 山东大学, 2017.
- [7] 何超, 胡章芳, 王艳. 一种基于改进 DTW 算法的动态手势识别方法[A]. 数字通信, 2003, 40(3): 21-25.
- [8] 张众. 小型语音识别系统的研究和开发[D]: [硕士学位论文]. 天津: 天津大学, 2004.
- [9] Keogh, E. (2004) Themistoklis Palpanas, Victor B. Zordan, Dimitrios Gunopulos, Marc Cardle. Indexing Large Human-Motion Databases.

### 知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>  
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2325-677X, 即可查询
2. 打开知网首页 <http://cnki.net/>  
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: [dsc@hanspub.org](mailto:dsc@hanspub.org)