

**本科生毕业设计[论文]**

**基于图像的视觉伺服**

院 系\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

专业班级\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

姓 名\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

学 号\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

指导教师\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

年 月 日

**学位论文原创性声明**

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名： 年 月 日

**学位论文版权使用授权书**

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于 1、保密囗，在 年解密后适用本授权书

2、不保密囗 。

（请在以上相应方框内打“√”）

作者签名： 年 月 日

导师签名： 年 月 日

**摘 要**

相比使用传统传感器作为机器人获取周围信息的方式，由于视觉传感器具有成本低、信息丰富、可靠性高等优点而得到了广泛的应用[1]。视觉伺服，即基于视觉的控制系统，已成为机器人领域的重要发展方向之一。基于图像的视觉伺服不需要对系统参数进行精确的标定，直接从图像中获取信息。本文推导了在具有6个自由度的手眼情况下基于图像视觉伺服中基本的机器人动力学模型、交互矩阵、控制器的设计以及误差函数的设计[2]。并且本文在UR3双臂机器人系统上，基于ROS（机器人操作系统）构建了完整的算法验证平台。使用HOG特征加SVM识别手写体所在位置从而提取关键点来构建交互矩阵，求解正运动学与反运动学，设计位置控制器和基于模糊PID的控制器，实现机器人简单的位姿控制与手写体数字的跟踪。

**关键词：**基于图像视觉伺服；机器人；ROS；运动学；手写体识别；SVM；手眼；位置控制器；基于模糊PID的控制器

**Abstract**

Comparing to using the traditional sensor to gather the information around the robots, the vision sensor is widely used because of its low cost, rich information and high reliability. Visual servo, which is a control system based on the vision data, has become one of the important development direction of the field of robotics. Image-Based Visual Servo can capture information directly from the image without accurate calibration for the system parameters. This paper deduces the basic robotics kinematic dynamics model、interaction matrix、design of the controller and the design of the error function based on Image-Based Visual Servo in the case of hand-in-eye with 6-DOF.In this paper,a complete algorithm validation platform has been built on the UR3 dual-arm robot system based on ROS (Robot Operating System).Using HOG feature and SVM to detect the hand written digit number’center point in image,constrcuting the interaction matrix,solving the forward kinematics and inverse kinematic,designing position controller, realizing simple postion-attitude controller,fuzzy based PID controller and object tracking of hand written digit number.

**Key Words：**Image-Based Visual Servo;Robotics;ROS;Kinematics;Hand written digit number recognition;SVM;Eye-in-hand;position controller;fuzzy based PID controller

**1 绪论**

**1.1 前言**

近年来，机器人技术飞速发展，机器人在工业、农业、服务业以及军事等方向得到了广泛的应用。与此同时，对机器人执行复杂任务的需求也日益增加，具备感知能力的智能机器人被认为是机器人发展的一大方向。

为了使机器人能够在复杂环境下工作，必须提高它的对环境的探测能力与对自身运动控制的能力。使其能在复杂环境下完成自身定位、地图构建、自主搜索等任务。为此，必须为机器人本体装配各种传感器,使它们能够获取外部环境信息进一步完成定位、探查、搜索等功能。

传统的检测手段往往面临检测范围的局限性以及检测手段的单一性，而且可拓展性差，包含信息相对简单，成本过高等缺点。而视觉传感器具有低成本、高信息、非接触式测量、适用范围广和算法简单等优点，成为最重要的机器人传感器之一。采用视觉作为反馈的视觉伺服控制是机器人系统重要的控制手段，可运用如零件的自动检测、智能机器人控制以及生产线的监控等工业自动化生产过程中，同时视觉伺服在运动目标的自动跟踪与识别等方面也有广泛的应用。可以认为视觉伺服控制是真正实现机器人智能控制的关键。

基于位置的视觉伺服，是利用已标定的摄像机，从图像特征中计算物体的三维空间位姿，然后在笛卡尔坐标系中对机器人 进行控制；而基于图像的视觉伺服控制，则是直接在图像空间进行伺服控制，消除当前图像特征与期望的图像特征之间的误差

目前大部分研究都建立在机器状态可解析性描述的基础上，并以某些特定的任务为目标，难以应用于复杂多变的实际应用中。因此如何使机器人真正具有与人一样的视觉能力，是一个需要长期研究的问题。

**1.2 研究现状**

视觉伺服控制涉及计算机视觉、机器人动力学、机器人运动学以及控制理论等多个领域的交叉学科，国内外学者在过去20年间进行了大量的研究。机器人视觉伺服主要是采用视觉传感器采集信息，通过姿态检测算法间接检测机器人的当前位姿或者相对位姿。国外Hutchinson等三篇论文[1-3]对视觉伺服控制进行了开拓性的研究与发展。国内视觉伺服控制的研究开始较晚，方勇纯[4]、薛定宇[5]等对视觉伺服的控制方法进行了综述，总结了经典的视觉伺服的控制算法，陶波[6]等对无标定视觉伺服控制研究进展进行了总结。

相比传统的在标定条件下使用视觉伺服控制系统，无标定视觉伺服控制系统具有更高的灵活性欲适应性，同时对模型的偏差以及图像噪声的等具有较强的鲁棒性，是机器人伺服控制系统未来重要的发展方向和研究热点[6]。虽然视觉伺服的研究已经有20年的历史，但是由于涉及学科众多以及这些学科的发展现状的限制，还有很多问题需要解决。视觉伺服未来研究的方向主要在图像特征的选择、无标定的视觉伺服的研究、视觉系统动态性能的研究、如何结合计算机视觉以及图像处理的最新研究算法、各种智能算法的结合以及多传感器的融合等[5]。

随着科技的不断进步，机器人作业任务将日趋复杂化、多样化。可以预见，视觉伺服控制技术将成为机器人控制的主要手段之一。机器人技术也将从专业技术领域逐步渗透到社会生活的各个方面。因此，在机器人技术的推广过程中，不需要专业标定环节的无标定视觉伺服方法凭借其更高的灵活性与适应性，势必会成为视觉伺服领域最为重要的分支之一。目前无标定视觉伺服技术虽然取得了较为丰硕的成果，但是远未达到成熟可靠的境界[6]。

**2 视觉伺服基础**

**2.1 视觉伺服基本概念**

视觉伺服控制是指使用计算机视觉数据控制机器人的运动。视觉数据可以是从直接安装在机器人上的相机获取操纵器或移动机器人上，在这种情况下的运动机器人引起相机运动，或者相机可以被固定在工作空间中，从而可以观察机器人的运动固定配置。

视觉伺服的控制方法的目的是最小化误差，一般定义如下：

这个公式具有一般性，他具有广泛的求解方法，将会在下面看到。式子(2-1)中的参数定义如下，向量是一组图像测量（例如，兴趣点的图像坐标或对象的质心的图像坐标）。这些图像的测量用于计算个视觉特征的向量，其中是表示的一组参数关于系统的潜在附加知识（例如，粗相机固有参数或对象的3-D模型）。矢量包含所需要的特征的值。而视觉伺服方案的主要区别在于对目标函数的选择与设计上面。

**2.2 基于图像的视觉伺服**

本文主要研究实现基于图像的视觉伺服控制的设计，基于图像的视觉伺服无需对系统参数进行精确的标定、信息可以直接从图像获取等优点得到广泛应用。本文在研究中，构建的目标函数所需要的图像的测量直接是感兴趣点所在图像坐标系中的坐标（例如多边形的定点），所以中可以直接由感兴趣点坐标直接表示，所以目标函数的构建可以直接完成。在本文中，由感兴趣矩形物体的四个顶点的图像坐标构成。

**2.3 摄像机系统**

由于采用的固定在6个自由度的机器手上面的手眼，手眼的运动会造成物体在摄像机成像平面上的运动，所以对摄像机系统的分析以及运动的模型建立十分重要。一般情况下，对于相机系中具有三维坐标点,最后投影在成像平面上的图像具有二维坐标点,我们有：

其中给出以像素为单位表示的图像点的坐标， (，，，)是相机固有参数集合：和是主点，是焦距，是像素尺寸的比率。式子(2-3)为相机的投影方程。

**2.4 欧拉角**

其中其中欧拉角是用于指定三维空间中对象的取向或取向取向改变的三个角度的集合。 欧拉角三角形中的三个角度中的每一个都在三维笛卡尔坐标系中指定围绕一个轴的元素旋转（见图2-1）。 这并不是完整的定义，如果要完全定义欧拉角系统。下面的约定很重要。

1、欧拉角的约定，可以是绕三个轴（z,y,x）Tait-Bryan ，也可以是两个轴（z,y,z）Classic,两者都可以表示所有可能的三维旋转。本文采用Tait-Bryan。

2、旋转的顺序也很重要，本文的欧拉角旋转顺序为z-y-x。即先绕z轴旋转，再绕y轴旋转，最后绕x轴旋转。

**3、内部旋转，外部旋转的区分也很重要，就是每次旋转的轴到底会不会随着你的旋转而发生变化，如果变化就是内部旋转，如果是世界坐标系的轴不会变化就是外部旋转。**

**4、主动被动，就是是点绕轴（系），还是轴（系）绕点。**

**5、坐标系的约定，本文采用的坐标系是右手坐标系。**

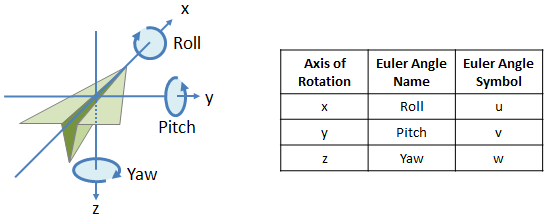


图2-1 欧拉角

**2.5 交互矩阵推导**

为了实现视觉伺服的控制设计，s随时间变换和摄像机速度之间的关系。 让摄像机的空间速度表示为，其中是相机系原点的瞬时线速度，是相机系的瞬时角度。

和之间的关系由下式给出:

其中，叫做s相关的交互矩阵。在一些视觉伺服文献中也叫作雅克比特征矩阵。本文设计的是基于图像的视觉伺服，在2.2中提到直接采用图像坐标作为目标函数，所以直接选取,其中就是特征点或兴趣点在图像中的坐标。所以需要推导s随时间的导数与手眼相机6个自由度的速度的关系。

取投影方程(2-2)关于时间的导数，可以得到：

式子(2-5)得到了,还需要得到其与相机速度之间的关系。这里速度分为两种速度，一个平动（位移）速度以及一个转动速度，所以这里需要用到坐标系变换矩阵。坐标系变换矩阵可以写成,其中是表示旋转矩阵，表示平移矩阵。则坐标系的变换可以写成：

其中分别表示旋转后的在坐标系中的坐标以及旋转前的坐标。

本文使用的旋转矩阵为：

其中分别为绕轴转动的欧拉角。假设在无穷小的时间内，角度的变化也很小,则有：

将(7)带入式子(6)中，可以得到：

而其中：

.

则可以得到下式：

由于在无穷小的时间内，所以则有：

即：

但是所有的改变由手眼的运动而引起的，所以运动速度要改变相应符号。得出在由具有6自由度的手眼运动而产生图像中特征点在图像坐标系中运动的方程为：

**2.5 控制器设计**

一旦选择了s，控制方案的设计可以非常简单。 也许最直接的方法是设计一个速度控制器。为了做到这一点，我们需要s随时间变换和摄像机速度之间的关系。 让摄像机的空间速度表示为，其中是相机系原点的瞬时线速度，是相机系的瞬时角度。

和之间的关系由下式给出:

其中，与s相关的交互矩阵。术语雅可比特征也在某种程度上可互换使用在视觉伺服文献中。

用式子（2-1）和式子（2-4），我们可以立即得到相机速度和误差的动态特性之间的关系：

其中，将设置成机器人控制器的输入，使误差值随时间而呈指数形式下降（即），我们通过（2-10）可以获得:

其中，被选择为的Moore-Penrose伪逆，就是当的秩是6满秩时，，

这个选择方法能够让和达到最小。当k=6时，且时，可以将求逆，然后可以给出控制

在现实的视觉伺服系统中，在实践中你不可能完全知道或者。因此，必须得到这两个矩阵之一的近似或估计。接下来，我们用符号来表示交互矩阵的近似的伪逆和交互矩阵的伪逆的近似。使用这个概念，控制律事实上就是:

**2.6稳定性分析**

使用李雅普罗夫稳定性定理来分析动态系统的稳定性，考虑候选李雅普罗夫函数，如下

该函数的导数为

而之前设计的控制器使得

所以当时，有

这将确保闭环系统的渐进稳定性。

**3 机器人运动学**

**3.1 机器人运动学基本概念**

机器人系统由机器人和作业对象与环境构成。机器人由机器人本体、控制器、软件三部分组成。为了获得作业对象信息与控制器输入，还需要传感器系统。机器人的本体主要有机械臂与移动机构组成，在许多应用中不需要移动机构。机械手由手臂与末端执行器构成。手臂通过多个关节的运动使末端执行器进行预设的运动到达预设的位置。每个手臂拥有一定自由度，自由度指机器人所具有的可进行独立坐标系运动的数目，一般情况下为了使机器人手部能在工作空间内以任意姿态到达任意位置，机器人至少需要六个自由度。

研究机器人运动学主要是要解决下面两个基本问题：

1. 对于一个给定的机器人，已知杆件的几何参数与关节参数，求末端执行器相对于给定坐标系的位置与姿态。给定坐标系一般为世界坐标系。
2. 对于一个给定的机器人，已知杆件的几何参数，给定末端执行器相对于机座坐标系位置与姿态，确定关节参数。

第一个问题常称为正运动学问题，第二个问题常称为反运动学问题。

机器人运动学运用几何学研究构成机器人系统结构的多个自由度运动学链的运动规律，几何强调机器人的连接部分被认为是刚体并且其中假设关节点只能进行转动或者平动。机器人运动学研究机器人系统中运动链的维度连通性与每个连接点的位置，速度和加速度之间的关系。为了规划以及控制运动路径并且输出给执行器的力和力矩，质量和固有内部属性，运动以及相关联的力和力矩需作为机器人动态系统的一部分来研究。

**3.2 运动学方程**

能够表示机器人运动链的运功学方程是机器人运动学中最基本的工具。运动学方程一般是一组非线性方程，描述成关节点相关参数与机器人配置系统之间的映射。运动学方程是机械系统的约束方程，例如机器人操纵器，其定义在一个或多个关节处的输入如何指定装置的配置，以到达任务位置或末端效应器位置。

Denavit-Hartenberg（D-H）方法在描述机器人动力模型中被广泛使用。D-H方法描述构成的串行机器人几何结构已经是研究机器人的基本准则。标准的D-H方法用来建立一个机器人的动力学模型。

在D-H方法表示的系统中，机械手由一串用转动或平移关节连接的刚体组成。每一对关节-杆件组成了一个自由度，每个杆件至多与两个杆件相连即不会构成闭环。

每个杆件可以由两个参数来表示。第一个参数为杆件的长度，即杆件上两个关节轴线的最短距离；第二个参数为杆件旋转角度，即两个相邻关节与的关节轴之间的夹角。

关节也用两个参数来表示。第一个参数是偏移量，即一个杆件到下一个杆件之间沿着关节的轴的距离；第二个参数是关节旋转角度，即一个杆件相对下一个杆件在关节轴上面的旋转角度。

通常这四个参数中三个参数是固定的，而一个叫做关节变量。对于一个旋转关节，关节变量就是，而对于一个移动关节，关节变量就是。D-H的标准相关定义在图3-1中描述。

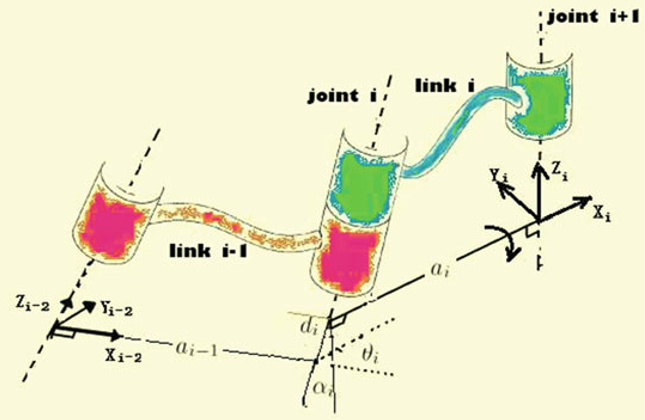


图3-1 标准D-H杆件关节相关定义

为了描述相邻杆件之间平移与转动的关系，可使用D-H方法为每个关节处的杆件建立齐次变换矩阵。这样通过可以将每个自由度的运动逐次传播至末端执行器。

对于坐标系与几何参数做如下定义：

1）轴为沿着关节的运动轴；

2）轴垂直于轴并指向离开轴的方向；

3）轴按右手坐标系建立；

4）为绕轴由轴向轴的关节角；

5）位从第坐标系原点到轴和轴的交点沿轴的距离；

6）为绕轴由轴转向轴的偏角；

7）为与两轴间的最小距离；

通过这些参数，连接部分其次变换矩阵将连接部分的坐标系转到坐标系可以由公式（3-1）表示。



其中缩写成,缩写成,表示绕着轴的旋转，表示沿着轴的平移。对于一个杆件的机器臂，全部的手臂的变换运动叫做正运动学，所有独立的杆件的变换可以写成一个形式，如公式（3-2）所示。在笛卡尔坐标系中末端执行器的位置可以用公式（3-3）来计算。





其中 是笛卡尔坐标系中的一个增广向量。

**3.2.1 正向运动学**

正向运动学是给定了机器人的各个关节的角度和连杆的长度后计算并确定末端执行器的位姿的理论。正向位置运动学（Forward Position Kinematic）解决了给定关节位置计算末端执行器的相应的响应位姿。

在串行链式模型中，正运动学的解通常是只有一个唯一解：一个表示给定关节位置的向量对应于唯一一个末端执行器的位姿。所以正向运动学求解比较简单。一般有两种方法求解——直接计算几何连接关系和使用变换矩阵。

**3.2.2 反向运动学**

反向运动学正好跟正向运动学相反。即为当需要末端执行器达到某一个期望的位置时，需要知道相应的关节角度应该达到的值。反向位置运动学（Inverse Position Kinematics）解决了给定指定的末端执行器的位姿，相应的关节位置的响应是什么的问题。与正向运动学相反的是，反向运动学的解通常不是唯一的;同样的末端执行器的位姿可以用不同的方案来实现。反向运动学的计算比较复杂。

反向运动学中会存在如下的问题：

1. 多组解。
2. 有些情形无解。
3. 冗余问题。

**4 图像处理**

**4.1 图像处理目标与方法**

本文主要工作是基于图像的视觉伺服算法设计，所需要的信息与反馈需要直接从图像中获取，以此构建控制器。本文采用对手写体数字的识别获得对象的中心像素点坐标从而实现对两个自由度的视觉伺服控制，实现物体跟踪。

本文将手写体数字作为识别目标分别为0到9的10个数字，使用图像的HOG特征，通过对MNIST的手写体数据库来SVM分类器作为识别方法，能够准确实时且在一定干扰下提取出手写数字信息，并获得各图像在摄像头中的像素坐标，控制机械臂按照目标一到目标三的顺序依次移动到目标正上方。图4-1为本文图像处理的流程：

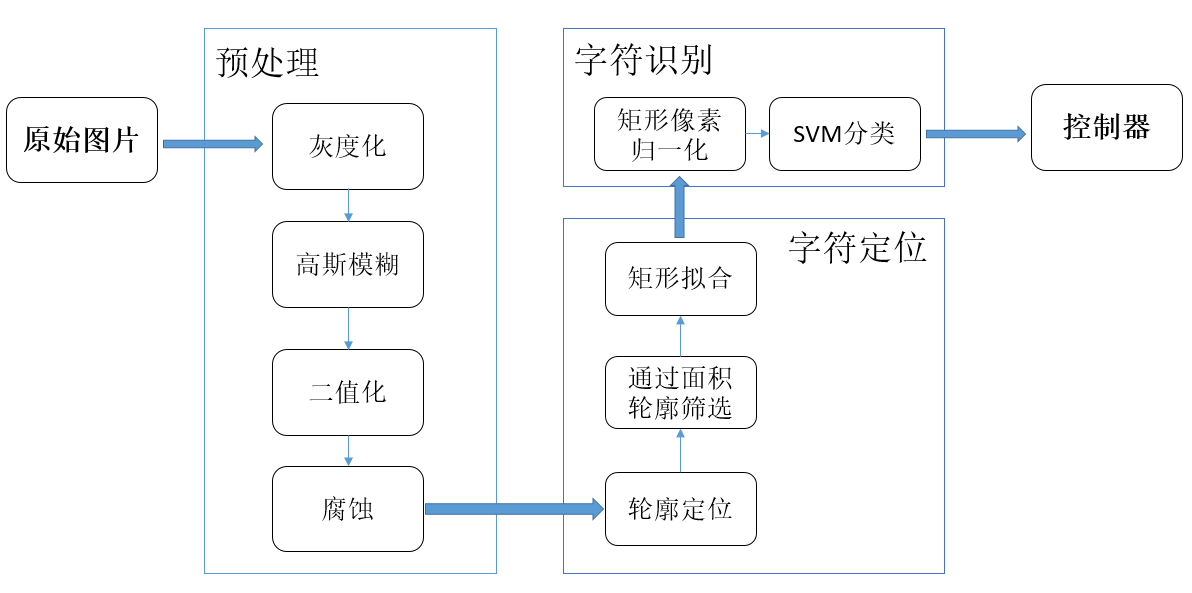


图4-1 手写体识别流程

**4.3 HOG特征**

HOG(方向梯度直方图)是一种在计算机视觉以及图像处理中广泛应用于物体检测的特征描述符。HOG特征最早被应用在SIFT算法中，由于它具有极强的图像特征描述能力，目前被广泛应用于图像处理中[9]。HOG特征通过计算和统计图像中局部区域的梯度方向直方图来构成特征。

HOG特征的提取流程分为5部分，分别是：

1、（可选）全局图像归一化。

2、计算图像的x、y方向的梯度。

3、计算梯度直方图。

4、块区域间归一化。

5、生成特征向量。

第一阶段应用可选的全局图像归一化均衡，旨在减少照明效果的影响。实际上，使用伽玛（幂）压缩，计算每个颜色通道的平方根或对数。图像纹理强度通常与局部表面照明成比例，因此这种压缩有助于减少局部阴影和照明变化的影响。

第二阶段计算一阶图像梯度。这些捕获轮廓，轮廓和一些纹理信息，同时提供进一步的抵抗照明变化。使用局部主导色彩通道，在很大程度上提供了色彩不变性。

第三阶段旨在产生对局部图像内容敏感的编码，同时保持抵抗姿态或外观的小变化。采用的方法以与SIFT功能相同的方式本地地集中梯度取向信息。图像窗口分为小空间区域，称为“单元格”。对于每个单元，我们在单元格中的所有像素上累加梯度或边缘方向的局部一维直方图。这种组合的细胞级一维直方图形成基本的“取向直方图”表示。每个方向直方图将梯度角范围分为固定数量的区间。单元格中的像素的梯度大小用于直方图中单元格的贡献程度。

第四阶段计算标准化，在进入下一阶段之前，将局部的细胞群和对比度归一化。归一化引入更好的照度，阴影和边缘对比度的不变性。通过在本地称为“块”的单元格的本地组上累积局部直方图“能量”的度量来执行。结果用于标准化块中的每个单元格。通常，每个单独的单元在多个块之间共享，但是其标准化是块依赖的，因此是不同的。因此，细胞在具有不同归一化的最终输出向量中出现多次。这似乎是多余的，但它提高了性能。我们将归一化块描述符称为定向梯度（HOG）直方图描述符。

最后一步将所有块的HOG描述符收集形成最后用于分类的特征向量中。

HOG特征结合SVM分类器已经广泛应用于图像识别当中。本文采用手写体数字的HOG特征训练SVM分类器来实现对手写体数字的识别。

**4.3 支持向量机**

支持向量机(SVM)是由Vapnik提出，针对回归于分类问题的一种统计学习理论。它是一种在高维特征空间中使用线性函数假设的学习方法，是具有准则性的最优化学习算法。

SVM的主要思想可以为以下两点：

1. SVM基于线性可分情况进行分析，若问题线性不可分，使用非线性[映射](http://baike.baidu.com/item/%E6%98%A0%E5%B0%84)算法将低维输入空间线性不可分的样本转化为[高维](http://baike.baidu.com/item/%E9%AB%98%E7%BB%B4)特征空间使其线性可分，从而将非线性特征分析转化为线性分析。
2. SVM基于结构风险最小化理论，通过在特征空间中构建最优分割超平面，使得问题得到全局最优解，并且在整个[样本空间](http://baike.baidu.com/item/%E6%A0%B7%E6%9C%AC%E7%A9%BA%E9%97%B4)的期望风险以某个概率满足一定[上界](http://baike.baidu.com/item/%E4%B8%8A%E7%95%8C)

SVM在解决非线性以及高维模式识别中，表现出了克服“维数灾难”以及“过拟合”等传统统计学习方法所遇到困难的特点，目前广泛应用于人脸识别、手写体识别、手势识别等领域。

**4.4 手写体定位与识别**

本文的手写体数字识别为脱机手写体数字识别，该方法分为几个步骤，第一步为预处理，第二步为字体定位，第三部为手写体识别，下面针对这三个方面进行阐述。

**4.4.1 预处理**

为了提高手写体数字识别的精度，排除实际环境中的各种干扰，需要对数据库中的图像做一系列预处理。在预处理方面已经有非常深入且成熟的研究，本文的预处理包括图像灰度化、高斯模糊、二值化、腐蚀。

图像灰度化即将彩色图像转变为灰度图像。彩色图像中的每个像素的颜色有R、G、B三个分量决定，而每个分量有255中值可取，这样一个像素点可以有1600多万（）的颜色的变化范围。而灰度图像是R、G、B三个分量相同的一种特殊的彩色图像，其一个像素点的变化范围为255种，在本文中先将摄像头采集到的彩色图像转变成灰度图像以使后续的图像计算更加简单，提高图像处理的实时性。处理得到的灰度图像与彩色图像一样仍然反映了整幅图像的整体和局部的色度和亮度等级的分布和特征。

高斯模糊是一种线性滤波方法，应用在图像中可以完成锐化、边缘检测等功能。我们可以通过设定一个二维的滤波器矩阵和一个待处理的二维图像，对于图像的每一个像素点，计算它的邻域像素和滤波器矩阵的对应元素的乘积，然后进行求和，作为该像素位置的值，这样就完成了滤波过程。通过合理的设置该滤波器矩阵可以使图像更加强调边缘。

二值化就是将图像上的像素点的[灰度值](http://baike.baidu.com/item/%E7%81%B0%E5%BA%A6%E5%80%BC)设置为0或255，即将整个图像呈现出明显的只有黑和白的视觉效果。常用的方法是设定一个阈值T，用T将图像的数据分成两部分：灰度大于T的像素群灰度设定为255和灰度小于T的像素群灰度设置为0。在很多实际应用场景中，分割的目标仅仅是将图像划分成两个区域：对应于场景中物体的一个区域，以及对应于背景的一个区域。所得到的图像被称为一个二元图像。

腐蚀是图像形态学处理中的一种基础方法。对于二值化以后的图像，由于像素点信息可以用0与1来表示，那么可以进行逻辑操作。定义一个腐蚀矩阵为B，图像为矩阵A。腐蚀首先以腐蚀矩阵B扫描A中每一个像素，然后用该矩阵与其覆盖的二值图像做与操作，如果得到矩阵中所有的值为1则像素输出为1，反之则为0。二值图像腐蚀操作最基本的效果是腐蚀图像中前景色区域的边缘。使得前景图像区域变小，前景图像内部的背景区域被放大。

**4.2.2 字符定位**

图像的预处理中我们得到了排除了无用信号，适用做字符检查的二值化图像。本节将字符从环境中识别并分割出来，方便最优做字符的识别。

首先将二值化的图像进行边缘检查，得到字符图形的边缘信息。该边缘可以看做是多边形，由于图像可能存在一些噪声点儿对最后的字符识别产生影响，所以首先对检测出的图形面积（即包围的像素多少）进行了筛选，将面积过小的噪声点排除。

对该多边形的定位有矩形拟合、椭圆拟合、多边形拟合等方法，由于本文系统需要有较高实时性，要求在保证精度的前提下尽可能简化算法复杂性，故采用了矩形拟合的方法。

拟合得到矩形后取矩形中心点作为该字符的位置。

**4.2.3 字符识别**

进行字符定位以后即可得到图像中被分割出的字符区域，将这些字符区域单独进行识别，即完成了图像处理的过程，得到了字符的数值与信息。

为了方便对所有区域进行统一处理，需要先将识别出的区域的大小转换到相同，本文中将所有的字符块都以2828像素进行处理。

将统一大小的图像输入SVM分类器进行识别，即得到字符对应的数值。至此得到了手写字符的位置与大小。图4-3为识别的结果。

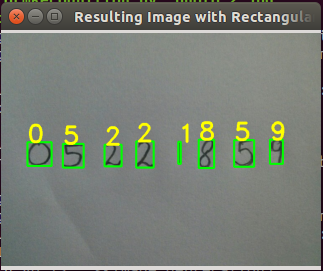


图 4-3 手写体数字识别结果

**5 ROS（机器人操作系统）**

**5.1 ROS介绍**

ROS (Robot Operating System,机器人操作系统)是目前主流的机器人控制框架，它由斯坦福大学人工智能实验室和Willow Garage合作研发，极大的提高了开发效率。

ROS提供了操作系统应有的服务，包括硬件抽象，底层设备控制，常用函数的实现，进程间消息传递，以及包管理。它也提供用于获取、编译、编写、和跨计算机运行代码所需的工具和库函数。提供一系列程序库和工具以帮助软件开发者创建机器人应用软件。ROS遵循BSD开源许可协议。

节点（node）,话题（topic）,消息（message）是ROS的核心概念，三者一同构成了机器人中所有模块的交互方法。图5-1为ROS概念图。

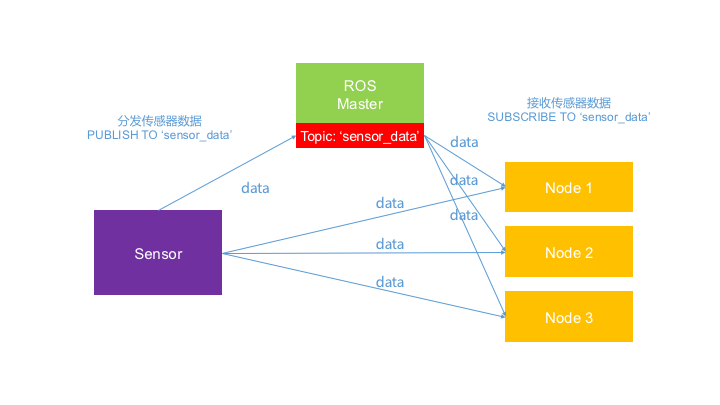


图 5-1 ROS概念图

在ROS中，可以自由地定义节点，一个节点基本对应一个功能或实体，该节点可以与其他节点通信，具体方式是节点在整个系统中广播话题，话题下面包含各类型消息，其他的节点可以订阅消息，可以实现各节点之间通信以及数据的传输。

各节点之间关系是相互独立的，概念图中的节点能够订阅指定消息而不需要知道该消息是哪个节点发送的，这样的结构在项目建设中起到重要的作用，不会因为某些节点的异常而影响整个系统的运行。

**5.2 服务**

节点之间消息的发布/订阅模型是一个非常灵活的通信模式，但是经常需要在分布式系统中使用到的请求/回复的交互机制在多对多的单向传输是不适合的。所以服务相比消息的发布/订阅主要用来解决请求/回复的功能。请求/回复是通过服务来完成的，服务由一对消息定义：一个用于请求，一个用于回复。客户端通过发送请求消息并等待应答来调用服务。这样可以实现客户端与服务器端持久的连接。

**5.3 启动文件**

ROS中另外一个比较重要的概念就是启动文件。启动文件是一个xml格式的文本文件，ROS通过解析启动文件来完成启动节点，设置参数、传递参数和重映射名称等功能，同时能够启动其他的启动文件。这样为我们将多个节点、工程、启动文件以及一些参数的设置与传递提供了很大的方便。同时启动文件可以启动多个，个个节点都是独立运行。通过对启动文件的修改，可以轻松的将需要启动的节点以及一些参数与配置文件等结合起来形成一个完成的工程从而完成特定的任务与目标。

Roslaunch命令是通过SSH本地和远程轻松启动多个ROS节点的工具，以及在参数服务器上设置参数。它包括自动重新生成已经死亡过程的选项。 roslaunch接收一个或多个指定要设置的参数和要启动的节点的XML配置文件（.launch扩展名）以及应该运行在的机器。

**6 算法平台构建**

**6.1计算平台**

本文采用的是Linux内核操作系统的桌面发行版本Ubuntu 14.04，支持ROS的Indigo版本。Linux是一种[自由和开放源代码](https://zh.wikipedia.org/wiki/%E8%87%AA%E7%94%B1%E5%8F%8A%E5%BC%80%E6%94%BE%E6%BA%90%E4%BB%A3%E7%A0%81%E8%BD%AF%E4%BB%B6)的[类UNIX](https://zh.wikipedia.org/wiki/%E7%B1%BBUnix%E7%B3%BB%E7%BB%9F)[操作系统](https://zh.wikipedia.org/wiki/%E4%BD%9C%E6%A5%AD%E7%B3%BB%E7%B5%B1)。该操作系统的[内核](https://zh.wikipedia.org/wiki/%E5%86%85%E6%A0%B8)由[林纳斯·托瓦兹](https://zh.wikipedia.org/wiki/%E6%9E%97%E7%BA%B3%E6%96%AF%C2%B7%E6%89%98%E7%93%A6%E5%85%B9)在1991年10月5日首次发布。，在加上[用户空间](https://zh.wikipedia.org/wiki/%E4%BD%BF%E7%94%A8%E8%80%85%E7%A9%BA%E9%96%93)的[应用程序](https://zh.wikipedia.org/wiki/%E6%87%89%E7%94%A8%E7%A8%8B%E5%BC%8F)之后，成为Linux操作系统。Linux也是[自由软件](https://zh.wikipedia.org/wiki/%E8%87%AA%E7%94%B1%E8%BD%AF%E4%BB%B6)和[开放源代码软件](https://zh.wikipedia.org/wiki/%E5%BC%80%E6%94%BE%E6%BA%90%E4%BB%A3%E7%A0%81%E8%BD%AF%E4%BB%B6)发展中最著名的例子。只要遵循[GNU通用公共许可证](https://zh.wikipedia.org/wiki/GNU%E9%80%9A%E7%94%A8%E5%85%AC%E5%85%B1%E8%AE%B8%E5%8F%AF%E8%AF%81)，任何个人和机构都可以自由地使用Linux的所有底层[源代码](https://zh.wikipedia.org/wiki/%E6%BA%90%E4%BB%A3%E7%A0%81)，也可以自由地修改和再发布。大多数Linux系统还包括像提供[GUI](https://zh.wikipedia.org/wiki/GUI)的[X Window](https://zh.wikipedia.org/wiki/X_Window)之类的程序。除了一部分专家之外，大多数人都是直接使用[Linux发行版](https://zh.wikipedia.org/wiki/Linux%E7%99%BC%E8%A1%8C%E7%89%88)，而不是自己选择每一样组件或自行设置。

Linux最初是作为支持[英特尔](https://zh.wikipedia.org/wiki/%E8%8B%B1%E7%89%B9%E5%B0%94)[x86](https://zh.wikipedia.org/wiki/X86)架构的个人电脑的一个自由操作系统。目前Linux已经被移植到更多的计算机[硬件](https://zh.wikipedia.org/wiki/%E7%A1%AC%E4%BB%B6)平台，远远超出其他任何操作系统。Linux可以运行在[服务器](https://zh.wikipedia.org/wiki/%E6%9C%8D%E5%8A%A1%E5%99%A8)和其他大型平台之上，如[大型主机](https://zh.wikipedia.org/wiki/%E5%A4%A7%E5%9E%8B%E4%B8%BB%E6%A9%9F)和[超级计算机](https://zh.wikipedia.org/wiki/%E8%B6%85%E7%BA%A7%E8%AE%A1%E7%AE%97%E6%9C%BA)。世界上500个最快的超级计算机90％以上运行Linux发行版或变种，包括最快的前10名超级电脑运行的都是基于Linux内核的操作系统。Linux也广泛应用在[嵌入式系统](https://zh.wikipedia.org/wiki/%E5%B5%8C%E5%85%A5%E5%BC%8F%E7%B3%BB%E7%BB%9F)上，如[手机](https://zh.wikipedia.org/wiki/%E6%89%8B%E6%9C%BA)（Mobile Phone）、[平板电脑](https://zh.wikipedia.org/wiki/%E5%B9%B3%E6%9D%BF%E9%9B%BB%E8%85%A6)（Tablet）、[路由器](https://zh.wikipedia.org/wiki/%E8%B7%AF%E7%94%B1%E5%99%A8)（Router）、[电视](https://zh.wikipedia.org/wiki/%E7%94%B5%E8%A7%86)（TV）和[电子游戏机](https://zh.wikipedia.org/wiki/%E7%94%B5%E5%AD%90%E6%B8%B8%E6%88%8F%E6%9C%BA)等。在[移动设备](https://zh.wikipedia.org/wiki/%E7%A7%BB%E5%8A%A8%E8%AE%BE%E5%A4%87)上广泛使用的[Android](https://zh.wikipedia.org/wiki/Android)操作系统就是创建在Linux内核之上。

通常情况下，Linux被打包成供个人计算机和服务器使用的Linux发行版，一些流行的主流Linux发布版，包括[Debian](https://zh.wikipedia.org/wiki/Debian)（及其派生版本[Ubuntu](https://zh.wikipedia.org/wiki/Ubuntu)、[Linux Mint](https://zh.wikipedia.org/wiki/Linux_Mint)）、[Fedora](https://zh.wikipedia.org/wiki/Fedora)（及其相关版本[Red Hat Enterprise Linux](https://zh.wikipedia.org/wiki/Red_Hat_Enterprise_Linux)、[CentOS](https://zh.wikipedia.org/wiki/CentOS)）和[openSUSE](https://zh.wikipedia.org/wiki/OpenSUSE)等。Linux发行版包含Linux内核和支撑内核的实用[程序](https://zh.wikipedia.org/wiki/%E7%A8%8B%E5%BA%8F)和库，通常还带有大量可以满足各类需求的应用程序。个人计算机使用的Linux发行版通常包含[X Window](https://zh.wikipedia.org/wiki/X_Window)和一个相应的桌面环境，如[GNOME](https://zh.wikipedia.org/wiki/GNOME)或[KDE](https://zh.wikipedia.org/wiki/KDE)。桌面Linux操作系统常用的[应用程序](https://zh.wikipedia.org/wiki/%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F)，包括[Firefox](https://zh.wikipedia.org/wiki/Firefox)网页浏览器、[LibreOffice](https://zh.wikipedia.org/wiki/LibreOffice)办公软件、[GIMP](https://zh.wikipedia.org/wiki/GIMP)图像处理工具等。由于Linux是自由软件，任何人都可以创建一个匹配自己需求的Linux发行版。

**6.2 UR3机器臂驱动**

本文采用的算法验证平台的机器臂为Universal Robots公司推出的UR3机器臂。UR3具有6个自由度，最大工作范围19.7英寸，最大负载3公斤。采用的驱动的是Shaun Edwards等人开发的ROS的universal\_robot工具包，提供了操作机器臂各个关节的相关节点、服务、机器人模型以及参数等信息。通过该工具包可以实现对各个关节位置以及速度的控制从而实现对末端执行器的控制。图6-1为UR3机器臂的可视化模型。

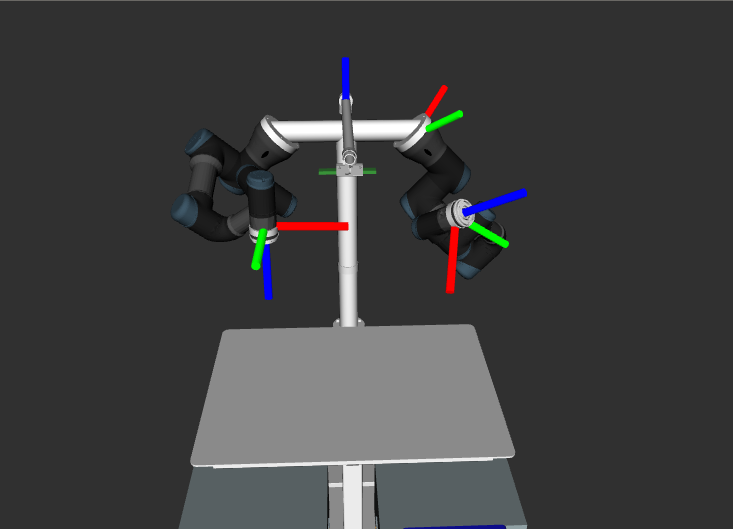


图 6-1 UR3机器臂

**6.3 KDL运动学求解库**

在6.1中提到通过universal\_robot工具包可以控制各个关机的位置以及速度，但是要控制末端执行器的位姿，就需要求解运动学方程。本文使用的是一个运动学求解库KDL(Kinematics and Dynamics Library)，是一个独立的框架可以建模和计算的运动链，如机器人，生物力学人体模型，计算机动画人物，机床等。它提供了各种几何对象（点，系，线，…）类库，各种运动链的集合（串行，仿人，并行，移动…）及其运动规范和插值操作。

所以对UR3机器臂的运动学方程的建立与求解通过KDL来完成、首先加载机器人模型、世界坐标系以及初始安装位置，同时获得当前机器人各个关节的位置，通过建立的正运动学方程计算出末端执行器在世界坐标系中的位置。控制末端执行器的运动就需要将期望的点的世界坐标系的坐标通过建立的反运动学方程求解出期望的各个关节的位置参数等，然后通过插值形成轨迹让机器臂的各个关节跟随轨迹从而实现末端执行器的运动控制。整个对末端执行器的控制流程图如图6-2所示。

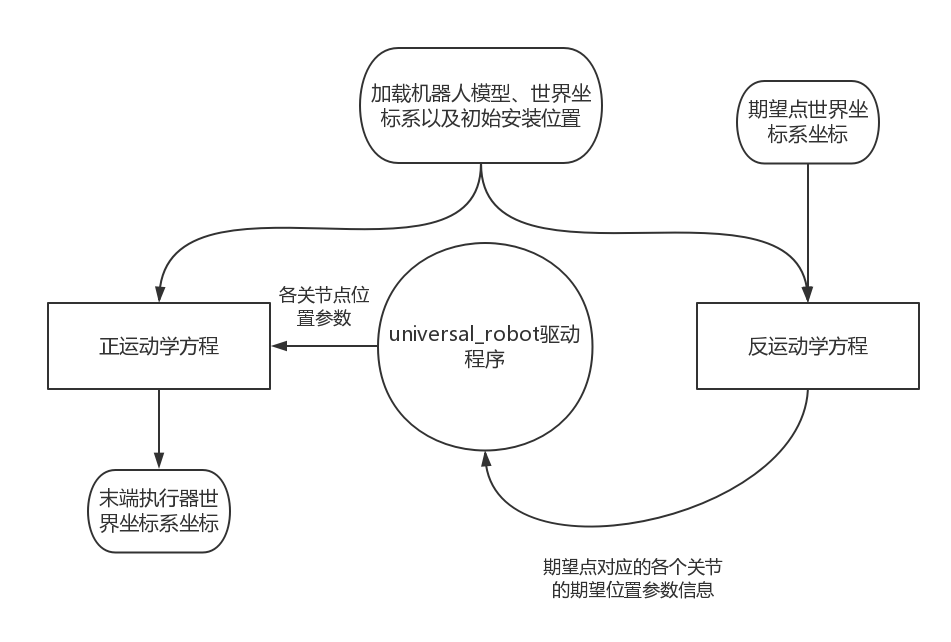


图 6-2 控制流程

**6.4 手眼**

采集图片信息本文使用了通用USB摄像头作为图像捕获设备，将USB摄像头固定在机器臂的末端成为末端执行器。驱动采用ROS下的Benjamin Pitzer等人开发的usb\_cam工具包，将摄像头获取的图像作为话题以30HZ频率发布出去。以供图像处理节点订阅消息进行图像处理后获取所需信息。

**6.5 手写体识别**

手写体识别使用python的机器学习工具包scikit-learn。图像处理部分采用OpenCV和scikit-image。采用HOG特征训练SVM分类器，使用MNIST手写体数据集作为训练数据。

**6.6 控制器设计**

**6.6.1 位置控制器**

设计位置控制器。获取图像中手写体的中心像素坐标点，得到与图像的正中间点的差值即为误差，也是控制器的输入，单位是像素。而根据公式（2-6）来设计控制器。

为了实现对手写体数字的跟跟踪，需要在两个平动自由度，所以这里P就是单位矩阵，得到式子（6-1）。

即得到.所以改变机器人末端执行器所在时间坐标系中的平动可以改变兴趣点在图像中的坐标。所以直接设计P控制器，得到控制律，由式子（6-2）给出。

其中T是控制器输出，直接代表及机器人末端执行器所在世界坐标系中应该运动的平动距离。Kp是控制器的比例环节，e是图像中兴趣点坐标与图像中心点坐标的误差。所以只要误差存在，控制器就会有输出，可以推出在稳态的时候稳态误差一定是0。所以Kp的选择会影响整个控制器，过大会导致系统震荡，过小会导致反应迟缓。最理想的情况Kp的选择应该是一步讲误差调节为0，但是由于在相机平面与世界坐标系中存在投影方程且机器人的运动学模型存在误差以及关节的电机控制存在误差，不可能保证对图像中的每一个兴趣点达到一步到达稳态。但是可以根据投影方程，通过实验，得到末端执行器在世界坐标系中移动的距离与相应图像中兴趣点移动的像素点的比例得到对Kp的估计。所以最后基于图像设计的视觉伺服控制器会对机器人的末端执行器的控制起到修正，哪怕建立的运动学方程以及电机的控制存在误差，最后也能够达到稳态误差为0的稳定状态。

**6.6.2 基于模糊PID控制器**

相比传统的PID控制，模糊控制是模拟人的思维、推理和判断，将先验知识与工程经验用自然语言到的形式表示出来，用来建立一种使用与数字计算机处理的输入输出过程的模型。模糊控制是智能控制的一个十分重要的研究领域[10]。模糊控制能够避开对象的数学模型，力图从成功失败的经验中提取总结，从中提炼出控制规则。模糊控制是一种基于规则的专家系统[11]。

将跟踪目标的像素点误差的绝对值作为模糊控制器的输入语言变量。以其变化的范围来定义模糊集上的域

对应的模糊子集为：

定义输出的语言变量，为速度PID控制器的比例环节Kp。

规则表的建立从系统的稳定性、响应速度、超调和稳态误差等各个方面来指定控制规则。规则如下：

1. 当误差的绝对值大时，输出Kp应该大。
2. 当误差的绝对值小时，输出的Kp应该小。
3. 当误差为0是，输出Kp应该为0。

隶属度函数采用三角隶属度函数，输入语言的范围为[0-2]对应像素点误[0-320](图像平面x轴)或[0-240](图像平面y轴)。输出为归一化的Kp属于[0-1]。图6-3是基于模糊PID

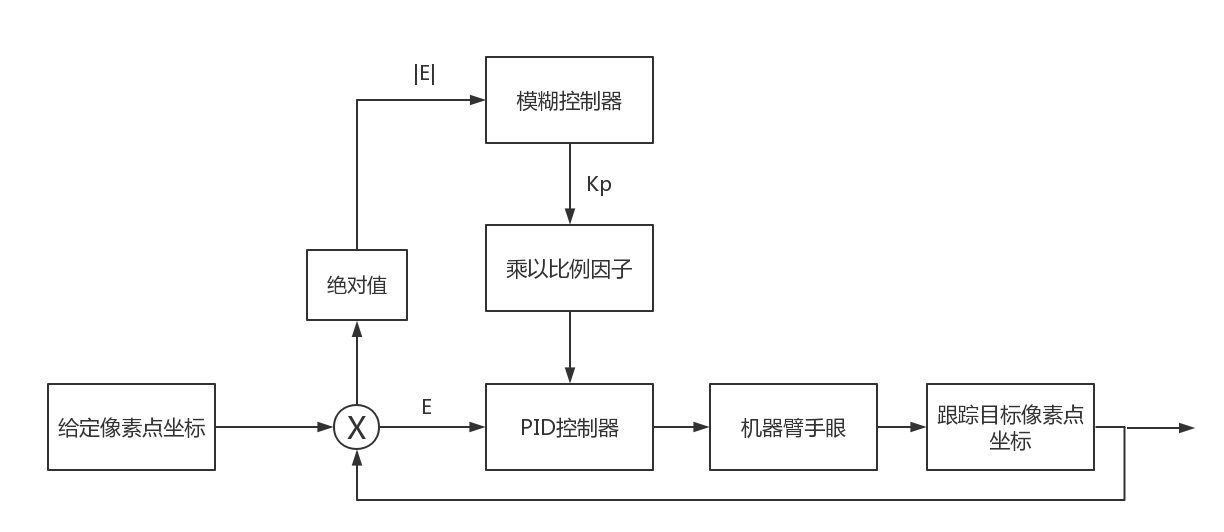


图 6-3 基于模糊PID控制流程

**6.7 跟踪结果分析**

实验中对三个手写体数字进行跟踪，图6-3为跟踪目标，可以看出对一定的噪声的干扰有抵抗，三个数字也正确识别。按照0、1 、2的顺序进行跟踪。

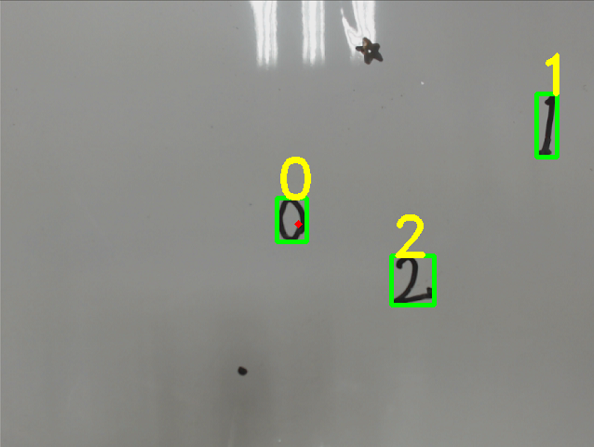


图6-3 跟踪目标

这里将机器人末端执行器对数字0的x轴像素点误差的跟踪曲线绘制出来，如图6-4所示。可以看到在图像坐标系中两个误差均最后达到稳态且稳态误差为0。实现了对目标物体的跟踪。

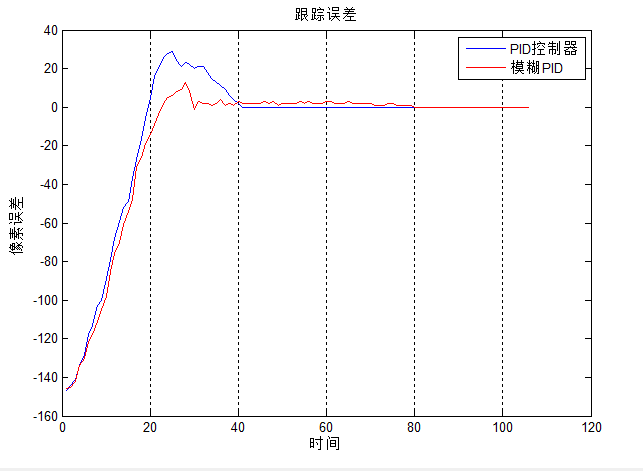


图6-4 跟踪误差

可以看出传统PID与模糊PID控制的效果相比较，模糊PID的控制效果明显要优于传统的PID控制算法，超调下降，，调整时间（3%）基本相当，稳态误差都是0。可以看出模糊控制基于规则和专家经验能够使控制效果得到提升。

**7反思与不足**

本次设计让我学到了很多知识，了解和接触到了很多理论和知识，同时也在实践中完成了很多工作，是自己的能力得到了提升，同时也认识到了自己的不足和差距，在控制理论相关的知识和认知还是相对欠缺，对整体平台架构的设计由于没有经验，换过多次方案，走了很多弯路。但同时也积累了很多经验，认识到实践与理论的不可分割的必要性，理论指导实践，实践验证理论。本次设计最大的收获是将基于图像的视觉伺服的理论成功的应用在了自主搭建的算法验证平台上面并且获取到了真实数据进行分析，在了解到不足后查阅资料，自行设计了模糊PID控制器，改进了系统的动态性能指标，收获颇丰。在此期间，也越来越感到自己的不足。

在编写机器人双臂驱动程序的时候，由于机器臂的整个驱动程序的复杂性，所以采用了ROS的工具包来实现，但由于提供接口等各种原因，并不能设计最常用的速度控制器，能够使误差呈指数下降，如果要设计速度控制器，需要重新建立机器臂的速度运动学方程，驱动程序也需要进行相关部分改写，工作量巨大。所以设计了简单的位置控制器，推导了交互矩阵，使用了比例环节。中间做了很多简化，在跟踪物体时只用到了两个平动自由度，并没有用到转动自由度，因为手眼摄像机的光轴与机器臂末端旋转的中心轴有一定的装配距离，转动会引起平动的耦合，导致整个运动会很复杂。所以没有考虑转动自由度。

**8 致谢**

本研究及学位论文是在我的导师袁烨老师的亲切关怀和悉心指导下完成的。他严肃的科学态度，严谨的治学精神，精益求精的[工作作风](http://www.so.com/s?q=%E5%B7%A5%E4%BD%9C%E4%BD%9C%E9%A3%8E&ie=utf-8&src=wenda_link)，深深地感染和激励着我。袁烨老师不仅在学业上给我以精心指导，同时还在思想、生活上给我以无微不至的关怀，在此谨向袁烨老师致以诚挚的谢意和崇高的敬意。我还要感谢一起在116一起完成毕业设计的同学，正是由于你们的帮助和支持，我才能克服一个一个的困难和疑惑，直至本文的顺利完成。

同时也感谢自动化学院为我提供良好的毕业设计的环境，使得我能够接触和使用到前沿的设备和技术。

最后再一次感谢所有在毕业设计中曾经帮助过我的老师、同学和朋友，以及在论文中被我引用和参考的文献论著的作者。

参考文献

[1]Hutchinson S, Hager G D, Corke P I. A tutorial on visual servo control. IEEE Transactions on Robotics and Automation,1996,12(5): 651-670

[2]Chaumette F, Hutchinson S. Visual servo control I basic approaches. IEEE Robotics & Automation Magazine, 2006,13(4): 82-90

[3]Chaumette F, Hutchinson S. Visual servo control II advanced approaches. IEEE Robotics & Automation Magazine, 2007, 14(1):109-118

[4]方勇纯. 机器人视觉伺服研究综述. 智能系统学报, 2008, 3(2):109-114

[5]薛定宇, 项龙江, 司秉玉, 徐心和. 视觉伺服分类及其动态过程. 东北大学学报(自然科学版), 2003,24(6):543-547

[6]陶波, 龚泽宇, 丁汉. 机器人无标定视觉伺服控制研究进展. 力学学报, 2016, 48(4): 767-783

[7]MALISE,CHAUMETTEF,BODETS.21/2D visual servoing[J]. IEEE Transactions on Robotics and Automation,1999,15(2):238-250

[8]CHAUMET TE F.Potential problems of stability and convergence in image-based and position-based visual servoing[C]The Conference of Vision and Control ,Volume 237 of Lec ture Notes an Co ntrol and Information Sciences .[ S.l.] :Springe r-Ve rlag , 1998.

[9]周柯. 基于HOG特征的图像人体检测技术的研究与实现[D].华中科技大学,2008.

[10]蔚东晓,贾霞彦.模糊控制的现状与发展[J]. 自动化与仪器仪表,2006,(06):4-7.

|  |
| --- |
| [11]葛新成,胡永霞. 模糊控制的现状与发展概述[J]. 现代防御技术,2008,(03):51-55. |

|  |
| --- |
|  |