

# 基于嵌入式 Linux 的视觉伺服系统设计

晁衍凯, 徐昱琳, 杨永焕

(上海大学 机电工程与自动化学院, 上海 200072)

**摘 要:**文中通过分析目前视觉伺服系统的实现方式,建立了一个基于嵌入式 Linux 的视觉伺服系统,用于室内相对简单背景下的运动目标识别与跟踪。文中首先完成了开源视觉库 OpenCV 到嵌入式 Linux 的移植,搭建了硬件平台、软件平台及交叉编译环境。基于 OpenCV 完成图像处理,采用图像的区域特征-不变矩作为特征信息,实现运动目标的形状识别。通过判断目标轮廓在图像中的像素偏移量,计算出机械臂末端的空间坐标值,调节机械臂各关节角度,通过不断逼近的方式,实现对运动目标的跟踪。实验证明,文中设计的视觉伺服系统能够实现设定的目标。

**关键词:**嵌入式视觉伺服;目标识别与跟踪;OpenCV;不变矩

**中图分类号:**TP24

**文献标识码:**A

**文章编号:**1673-629X(2012)05-0006-04

## Embedded Linux Based Visual Servo System Design

CHAO Yan-kai, XU Yu-lin, YANG Yong-huan

(School of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200072, China)

**Abstract:** Through the analysis of the present visual servo system implementation, it concerns the establishment of a visual servo system based on embedded Linux mainly used for recognition and tracking of the moving objects in a relatively simple background of the house. At first, studied the successful transplant of the open source visual library OpenCV to the embedded Linux and building of the hardware/software platform and cross-compiling environment. The image of the moving object is processed via OpenCV. The shape recognition of the moving object is achieved according to the feature information of provincial characteristics of image-moment invariants. The spatial value of manipulator end is calculated by estimating the offset of the moving project's profile, and then the angles of the mechanical arms' joints are adjusted accordingly. These conditions would be approximated gradually to achieve the tracking of the moving object. Experiments show that this visual servo system can realize the target.

**Key words:** embedded visual servo; object recognition and tracking; OpenCV; moment invariants

## 0 引 言

随着科技的飞速发展,诞生于20世纪80年代的机器人视觉伺服系统<sup>[1]</sup>在各行各业得到了广泛的应用,吸引了机器人领域众多学者的普遍关注。相比于目前工业机器人仅能在严格定义的结构化环境中执行预定指令动作<sup>[2]</sup>,基于视觉伺服系统的机器人能提高对环境的感知和应变能力,自动识别并准确快速跟踪目标<sup>[3]</sup>,执行预先指定的动作,极大地推广了机器人的应用。随着各种控制算法<sup>[4]</sup>和智能决策<sup>[5]</sup>在机器人控制中的应用,机器人的功能性和智能性将得到极大提高。根据反馈信息类型的不同,机器人视觉伺服系统分为基于位置的视觉控制<sup>[6]</sup>和基于图像的视觉控制<sup>[7]</sup>,前者的反馈误差在3D Cartesian空间进行计算,

后者的反馈误差在2D图形平面空间进行计算<sup>[8]</sup>。

目前,机器人视觉伺服系统在国内得到了广泛应用,但这些系统大多基于PC或者工控机。随着对机器人小型化、模块化的要求越来越高,体积更小、模块化很高的嵌入式设备逐渐进入工业控制领域<sup>[9]</sup>。而且随着集成电路设计及制造技术的发展,搭载实时操作系统的嵌入式系统成为研究的热点,其应用不断扩展到消费医疗电子、通信、汽车电子等领域。可以预见的是,随着嵌入式技术的发展和需求的扩张,具有体积小、稳定性高、开发周期短、功耗低等优点的嵌入式系统定有广阔的应用前景。

传统的机器人视觉伺服系统大都基于标定技术和雅可比矩阵。采用基于标定的方法实现视觉伺服,需要预先对摄像机的内外参数进行标定<sup>[10]</sup>,由于参数标定结果受摄像机位置及焦距等因素影响较大,基于标定技术的机器人视觉伺服方法受到很大限制。基于雅可比矩阵的机器人视觉伺服方法是利用雅可比矩阵描述了机器人系统中图像空间的微分变化和机器人工作

收稿日期:2011-10-12;修回日期:2012-01-15

基金项目:上海大学研究生创新基金(SHUCX112188)

作者简介:晁衍凯(1988-),男,山东泰安人,主要从事嵌入式系统研究;徐昱琳,博士,副教授,研究方向为智能控制。

空间的微分变化之间的非线性映射关系<sup>[11]</sup>。在机器人运动过程中图像雅可比矩阵是不断变化的,因此需要在线或离线识别图像雅可比矩阵,计算量较大不适合在嵌入式平台上实现。

文中提出一种逐步逼近的方法,通过反复采集摄像机图像,判断目标物体在图像中的位置,计算出下一时刻机械臂末端的空间坐标值,从而建立图像坐标到空间坐标的映射,控制机械臂末端不断逼近目标物体,实现对运动物体的跟踪。

## 1 硬件平台的搭建

文中的控制平台采用天嵌公司的基于 S3C2440 处理器的开发板,并搭载开源 Linux 操作系统。S3C2440 是 Samsung 公司研发的基于 ARM 920T 的 16/32-bit RISC 微处理器,具有 USB、UART、LCD 等丰富的外设接口,最高时钟频率可达 400MHz,在嵌入式领域已得到广泛应用。Linux 是一款开源免费操作系统,具有大量的外设驱动,可方便裁剪移植到多种硬件平台,在 ARM 处理器上搭载 Linux 操作系统可以大大提高系统性能,增加程序设计的灵活性及可移植性。

USB 摄像头采用中星微公司生产的 CMOS 图像传感器 ZC301 芯片组,内置 JPEG 硬压缩器,Linux 2.6 以上版本的内核对该摄像头均有很好的支持。

执行系统采用哈尔滨奥松机器人科技有限公司的四自由度机械臂,机械臂由金属板和 RB-421 型号舵机连接而成,控制板的控制芯片为 AVR 系列单片机 ATMEGA8,输入为串口,输出为 32 路 PWM,32 路 PWM 可同时输出,且输出速度可调。机械臂关节运动通过舵机转动实现,控制命令由开发板通过串口输入到控制板,可实现各关节同时转动,转动角度精度较大,速度可调。

## 2 软件平台的设计

文中首先对 Linux 2.6.30 内核进行裁剪,使其在满足需求的前提下达到最小,并制作 yaffs2 文件系统<sup>[12]</sup>。移植 Linux 内核及文件系统到硬件平台上,利用 u-boot 引导内核使其正常运行。搭建交叉编译环境,在 PC 的虚拟机软件上安装 Linux 系统,并在 Linux 上安装 4.3.2 版本的交叉编译器,应用程序编辑完成经交叉编译生成可执行文件后,经串口工具传输到嵌入式 Linux 的文件系统。

文中的图像处理部分利用视觉库 OpenCV 完成,OpenCV 是由 Intel 建立,实现了图像处理和计算机视觉方面很多通用算法的跨平台计算机开源视觉库,需要预先移植该开源库到嵌入式平台。采用 OpenCV 1.0.0 版本,在该目录下执行 ./configure 命令对编译选项

进行配置,用 --host = arm-linux 选项指定目标平台,用 --enable-static 选项指定生成静态库。此外还要增加对视觉库本身特性的配置选项。配置完成后,分别执行 make、make install 命令进行编译和安装。安装完成后将生成的库文件拷贝到文件系统的 /usr/lib 目录下。

视觉伺服过程程序组成模块及结构如图 1 所示。

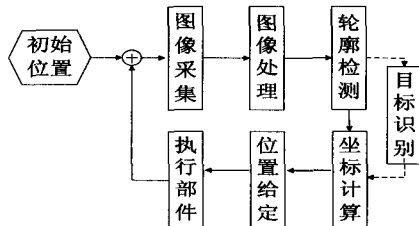


图1 系统总框图

摄像头到达初始位置后开始采集图像,系统对图像进行处理,首先进行形状识别,根据识别结果判断是否是目标物体。如果是目标物体,计算目标物体中心在图像坐标系中的坐标值,判断得到手臂末端需要调整的方向及幅度,结合手臂末端在空间坐标系中的初始坐标值得到下一时刻手臂末端的目标坐标值,通过逆运动学解得到手臂各个关节的转动角度,把转动角度转换成标准命令格式通过串口传输给控制板,手臂末端移动至摄像头光轴与目标物体垂直处。

再次采集摄像头数据,不断重复以上过程,直到机械臂末端逼近目标物体,即目标物体中心在图像坐标系中心 10 像素范围内。如果不是目标物体,调整摄像头初始位置,继续寻找。寻找到目标物体后,反复采集摄像头数据,经图像处理,不经过识别直接计算坐标值,控制手臂运动,实现对目标物体的跟踪。

通过开发板的 LCD 屏实时显示图像采集信息,实现对跟踪过程的监控。

### 2.1 图像处理

图像处理部分的功能通过调用 Intel 的开源图像处理库 OpenCV 的函数完成。通过摄像头得到图像后,首先调用函数 colourfilter 进行颜色滤波,然后依次调用函数 cvCvtColor、cvThreshold 进行颜色空间转换和二值化处理,最后调用函数 cvFindContours 检测轮廓及绘制轮廓。

在图像采集过程中,由于光线等因素影响,在图像中有时会出现斑点,对目标物体中心的计算结果会产生很大影响。因此在图像处理完成后,系统会计算各个轮廓的像素面积值,对小于面积阈值的视为干扰而移除,这样的处理保证了目标物体轮廓中心计算结果的正确性。

目标物体轮廓检测完成后,计算其中心在图像坐标系中的坐标值,并进行标记。图像处理部分流程图

如图 2 所示。

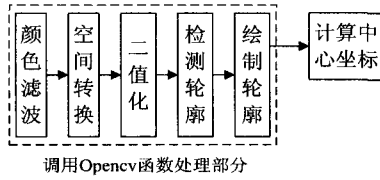


图 2 图像处理流程图

## 2.2 目标识别

矩是一种非常重要的表示目标总体形状的特征量,二维图形的几个关键特征均直接与矩有关,如目标的大小、形心和旋转情况等。由于不变矩概念清晰、识别率稳定,对具有平移、旋转和缩放变化的目标有良好的不变性及抗干扰性,能有效地反映目标的本质特征。所以,文中选用 Hu M. K 提出的 7 个不高于三阶的函数式中的 5 个作为识别目标所依据的特征量。

一幅目标图像可以用二维函数  $f(x, y)$  表示,其二维的阶矩阵  $(p + q)$  定义如下:

$$m_{pq} = \sum_{i=1}^M \sum_{j=1}^N i^p j^q f(i, j) \quad (1)$$

从而中心矩表示为:

$$m_{pq} = \sum_{i=1}^M \sum_{j=1}^N (i - \bar{i})^p (j - \bar{j})^q f(i, j) \quad (2)$$

式中,  $\bar{i} = \frac{m_{10}}{m_{00}}$ ,  $\bar{j} = \frac{m_{01}}{m_{00}}$ 。归一化的中心矩定义为

$$\mu_{pq} = \frac{\eta_{pq}}{\eta_{00}} \quad (3)$$

式中,  $r = (p + q) / 2 + 1$ ,  $p + q = 2, 3, \dots$

Hu M. K 利用规格化中心矩导出了 7 个不高于三阶的函数式,称之为 Hu 不变矩,它们对图像平移、旋转和尺度缩放都具有不变性。其中五个为:

$$\varphi_1 = \mu_{20} + \mu_{02} \quad (4)$$

$$\varphi_3 = (\mu_{30} - 3\mu_{12})^2 + (\mu_{03} - 3\mu_{21})^2 \quad (5)$$

$$\varphi_5 = (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{30} + \mu_{12})^2] + (3\mu_{12} - \mu_{30})(\mu_{21} + \mu_{03})[3(\mu_{12} + \mu_{30})^2 - (\mu_{21} + \mu_{03})] \quad (6)$$

$$\varphi_6 = (\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] + 4\mu_{11}(\mu_{21} + \mu_{03})(\mu_{12} + \mu_{30}) \quad (7)$$

$$\varphi_7 = (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] + (3\mu_{12} - \mu_{30})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \quad (8)$$

实验中发现,5 个不变矩的数值较小且变化范围很大,为了便于对识别目标特征量的比较,对计算结果采用取对数的方法进行数据压缩,并取绝对值,实际采用的不变矩为

$$\varphi_k = |\log_{10} |\varphi_k|| \quad k = 1, 2, \dots, 7 \quad (9)$$

实验中,用不规则图形代表需要识别的目标(如

图 3 所示),表 1 中给出了两个灰度图像旋转和尺度缩放时不变矩提取的部分数据。

表 1 图像的不变矩数据

旋转		三角形的 5 个不变矩			
$10^\circ$	0.598	2.168	5.401	3.260	5.936
$40^\circ$	0.595	2.165	5.437	3.235	5.879
$90^\circ$	0.598	2.157	5.427	3.225	5.863
$180^\circ$	0.606	2.171	5.459	3.246	6.043
尺度		五边形的五个不变矩			
0.8	0.774	3.359	9.749	2.634	9.384
1.0	0.775	3.372	9.751	2.544	9.701
1.2	0.774	3.366	9.841	2.668	9.578
1.4	0.775	3.375	9.859	2.527	9.650

从表中可以看出,选取的目标物体旋转任意角度或者面积缩放任意尺度,文中选取的五个 Hu 不变矩的函数值能够保证很小的变化范围。数据发生波动主要是由于图像分辨率较低,仅在整数坐标处才有精确值,因此计算得到的不变矩不能随原图像的几何变换恒定不变,图像的分辨率越高,这种误差就越小,但计算量也会相应的变大。通过本实验可以得出,在一定范围内,文中选取的不变矩的函数值对具有平移、缩放、旋转的目标能够保持不变性,可以有效地表示目标的形状特征,为目标识别提供一个很好的参考标准。



图 3 目标物体图像

## 2.3 机械臂空间位置确定

由于文中采用的机械臂总臂长较小,实现对目标物体的跟踪必须保持机械臂与目标物体较近的距离,这样会造成摄像机可视范围较小,机械臂末端只能跟踪位于较小范围内的物体。在完成目标物体识别后,根据目标物体所处区域,系统会给定机械臂末端一个空间坐标初始值,机械臂末端在给定初始值基础上进行第一次运动。

文中以图像中心为坐标原点建立图像坐标系,机械臂到达某一位置后采集摄像头数据,在完成目标物体轮廓中心确定后,系统会判断轮廓中心位于图像坐标系的哪个象限,从而确定机械臂末端的移动方向,移动幅度为 1cm。机械臂末端的空间坐标值加上移动方向及幅度即为机械臂下一时刻的位置,将该坐标值转化成标准格式的命令串通过串口发送到机械臂的控制板,控制机械臂末端到达相应位置。在机械臂移动过程中,保持机械臂与目标物体所在平面的距离不变。

摄像头焦点到目标物体表面的距离利用小孔成像原理得到。

## 2.4 机械臂数学模型建立及运动学逆解

首先建立机械臂数学模型,该模型的建立是执行模块的重要组成部分。建立机械臂数学模型大都采用 D-H 方法,但该方法建立过程复杂,计算量较大,适合自由度较多的复杂机械臂建模。鉴于文中采用的机械臂自由度较少,结构简单,文中直接按照机械臂的空间结构建立数学模型,将机械臂各臂抽象为线段,各关节抽象为转动角度。抽象所得机械臂结构示意图如图 4 所示。

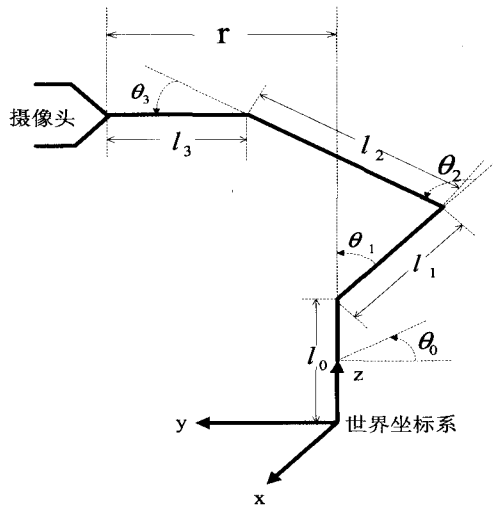


图4 机械臂结构示意图

根据机械臂末端的空间坐标值计算各关节转动角度的计算过程称为机械臂运动学逆解,由于本数学模型建立较简单,因此逆解运算也比较容易。文中建立的空间结构模型,运用三角函数的方法求得各关节的转动角度,完成逆解。

以机械臂第一个关节为坐标原点建立空间坐标系,即世界坐标系。精确测量机械臂各臂长分别为  $l_0$ 、 $l_1$ 、 $l_2$ 、 $l_3$ ,为了简化计算,在文中  $l_0$  取零。经坐标计算得到机械臂末端及摄像机在世界坐标系中的坐标值,经机械臂运动学逆解得到各关节转动角度的计算公式如下:

$$\tan\theta_0 = \frac{y}{x} \quad (10)$$

$$\cos\theta_1 = \frac{l_1^2 + x^2 + y^2 + z^2 - l_2^2}{2 \cdot l_1 \cdot \sqrt{x^2 + y^2 + z^2}} \quad (11)$$

$$\cos\theta_2 = \frac{l_2^2 + l_3^2 - x^2 - y^2 - z^2}{2 \cdot l_2 \cdot l_3} \quad (12)$$

$$\theta_3 = 90 - \theta_2 - \theta_1 \quad (13)$$

经过反余弦计算可得各关节角度值  $\theta_0$ 、 $\theta_1$ 、 $\theta_2$ 、 $\theta_3$ ,舍弃不合理角度后,将角度值转换成标准命令格式“# 1 P S #2 P S #3...”,命令串中数字代表输出的

引脚号,P 代表舵机转动角度,S 代表舵机转动速度,通过串口发送给控制板。控制板分析命令后,在引脚输出与命令相应格式的 PWM,控制舵机转动,实现控制机械臂运动。PWM 的周期为 20ms,占空比为 5% ~ 25%,占空比不同转动角度不同,可控制 PWM 的输出速度控制舵机的转动速度。

## 3 实验

文中搭建了基于嵌入式 Linux 的视觉伺服系统,并提出了从图像坐标到空间坐标的映射方法和运动学逆解方法,为了验证系统功能的正确性,将该系统应用于对室内简单背景下运动物体的跟踪。

本实验设计机械臂自动跟踪,通过分析目标物体与机械臂末端在世界坐标系中坐标值之间的误差,从而证明系统功能的正确性。目标物体为绿色形状不规则物体,在白色墙壁上做自由运动,机械臂末端由复位位置开始跟踪运动物体,保证运动物体始终在机械臂末端可跟踪范围内,并保持摄像头与运动物体相距 100mm。随机选取 5 个时间点做精确测量,得到实验数据如表 2 所示,表中数据均为世界坐标系中的坐标值。

表2 实验数据表

	运动物体坐标(mm)	机械臂末端坐标(mm)	系统计算值(mm)	误差(mm)
1	(213.5, 30.0, 254.5)	(116.5, 35.5, 250.0)	(115.3, 32.3, 252.3)	(3.0, 5.5, -4.5)
2	(212.5, 64.5, 220.0)	(119.5, 60.0, 225.5)	(115.7, 60.2, 222.5)	(7.0, -4.5, 5.5)
3	(212.5, 100.0, 200.5)	(110.5, 96.0, 202.5)	(116.9, 88.3, 210.4)	(-2.0, -4.0, 2.0)
4	(213.0, -131.0, 283.5)	(122.0, -120.5, 272.0)	(114.9, -122.5, 280.4)	(-9.0, -10.5, -10.5)
5	(212.5, -31.0, 292.5)	(115.0, -20.0, 294.0)	(116.1, -30.3, 288.4)	(2.5, -11.0, 1.5)

实验得出,运动物体和机械臂末端在世界坐标系中的坐标值之间的差别在允许范围内,机械臂末端基本实现对运动物体的跟踪。在系统终端下使用命令./genzong &,使命令在后台执行,使用命令 top 查看该程序运行时占用的内存及 cpu 使用率分别为:33.5%, 93.1%。

## 4 结束语

文中基于机器人小型化、模块化的发展趋势,提出了一种基于嵌入式 Linux 的视觉伺服系统的设计方法。首先给出了系统软硬平台的搭建方法,利用图像

(下转第 22 页)

明利用神经网络的方法在预测 miRNA 方面也是行之有效的。并且神经网络更擅长处理高维、大样本的数据集,随着已发现的 miRNA 数量越来越多,在大样本预测的情况下人工神经网络将体现出它独特的优势,准确度也会进一步提高。后续工作可以在两方面进行继续跟进,一是提取更多的 miRNA 相关数据集,更多的训练样本可以拟合出更精确的模型;二是改进人工神经网络,通过增加反馈、集成的方法进一步提高预测的精度,相信人工神经网络的方法在预测 miRNA 的方面会有更大的发挥。

#### 参考文献:

- [1] Bartel D P. MicroRNAs: genomics, biogenesis, mechanism and function[J]. Cell, 2004, 116(2): 281-297.
- [2] Robert J, Johnston S. A microRNA controlling left/right neuronal asymmetry in *Caenorhabditis elegans* [J]. Nature, 2003, 426(6968): 845-849.
- [3] Altuv Y, Landgraf P, Lithwick G, et al. Clustering and conservation patterns of human microRNAs[J]. Nucleic Acids Res, 2005, 33(8): 2697-2706.
- [4] Lim L P, Lau N C, Weinstein E G, et al. The microRNAs of *Caenorhabditis elegans* [J]. Genes Dev, 2003, 17(8): 991-1008.
- [5] Carter R J, Dubchak I, Holbrook S R. A computational approach to identify genes for functional RNAs in genomic sequences[J]. Nucleic Acids Res, 2001, 29(19): 3928-3938.
- [6] Xue C H, Li F, He T, et al. Classification of real and pseudo microRNA precursors using local structure-sequence features and support vector machine[J]. BMC Bioinformatics, 2005, 6(1): 310-310.
- [7] Griffiths-Jones S. The microRNA Registry[J]. Nucleic Acids Research, 2004, 32(1): 109-111.
- [8] Karolchik D, Baertsch R, Diekhans M, et al. The UCSC genome browser database[J]. Nucleic Acids Res, 2003, 31(1): 51-54.
- [9] Hofacker I L. Vienna RNA secondary structure server[J]. Nucleic Acids Res, 2003, 31(13): 3429-3431.
- [10] Pruitt K D, Maglott T D R. RefSeq and LocusLink: NCBI gene-centered resources [J]. Nucleic Acids Res, 2001, 29(1): 137-140.
- [11] 阎平凡, 张长水. 人工神经网络与模拟进化计算[M]. 北京: 清华大学出版社, 2005.
- [12] Bonnet E, Wuyts J, Rouze P, et al. Evidence that micro-RNA precursors, unlike other non-coding RNAs, have lower folding free energies than random sequences [J]. Bioinformatics, 2004, 20(17): 2911-2917.
- [13] Zhao Dongyu, Wang Yan, Luo Di, et al. PMirP: A pre-micro RNA prediction method based on structure-sequence hybrid features [J]. Artificial Intelligence in Medicine, 2010, 49(2): 127-132.
- [14] Hornik K. Multilayer Feedforward Networks are Universal Approximators[J]. Neural Networks, 1989(2): 359-366.
- [15] 谭 营. 计算智能导论[M]. 第 2 版. 北京: 清华大学出版社, 2010.
- [16] Kennedy J, Eberhart R. Particle swarm optimization[C]//Proceedings of the 1995 IEEE International Conference on Neural Networks. [s. l.]: [s. n.], 1995: 1942-1948.

(上接第 9 页)

矩作为特征量识别运动物体,然后提出了一种从图像坐标到空间坐标的映射方法,并结合文中选用的机械臂的结构特点,提出一种运动学逆解方法。最后,在室内简单背景的环境中,实现了机械臂对运动物体的跟踪,验证了本方法。

今后的工作主要在图像处理算法、编程架构上进行改进和提高,使系统实时性更高,提高跟踪速度和精度。

#### 参考文献:

- [1] Hutchinson S, Hager G D, Corke P I. A Tutorial on Visual Servo Control[J]. IEEE Transaction on Robotics and Automation (S1042-296X), 1996, 12(5): 641-670.
- [2] 廖万辉, 李 琳, 陈 祯. 基于嵌入式的智能视觉伺服系统的研发[J]. 机电产品开发和设计, 2009, 22(3): 3-5.
- [3] 刘 伟, 王建平, 张崇巍. 一种移动机器人对运动目标的检测跟踪方法[J]. 计算机技术与发展, 2009, 19(4): 105-108.
- [4] 陈 虹, 梁文彬. 基于机器人的神经网络预测控制算法[J]. 计算机技术与发展, 2008, 18(8): 65-68.
- [5] 魏长水, 贺巧龙, 李东亮. Mirobot 足球机器人决策系统的研究[J]. 计算机技术与发展, 2008, 18(5): 31-33.
- [6] Wilson W, Hulls C, Bell G. Relative end-effector control using Cartesian position-based visual servoing[J]. IEEE Transaction on Robotics and Automation, 1996, 12(10): 684-696.
- [7] Feddema F, Mitchell O. Vision-guided servoing with feature-based trajectory generation[J]. IEEE Transaction on Robotics and Automation, 1989, 5(10): 691-700.
- [8] 赵清杰, 连广宇, 李增圻. 机器人视觉伺服综述[J]. 控制与决策, 2008, 16(6): 849-852.
- [9] 陈友东, 王田苗, 魏洪兴. 工业机器人嵌入式控制系统的研究[J]. 机器人技术与应用, 2010(5): 10-13.
- [10] 刘 欢, 魏立峰, 王 健. 机器人视觉伺服系统的标定[J]. 机器人技术, 2007, 23(4): 278-279.
- [11] 王洪斌, 吕 玲, 李 萍. 在线估计雅可比矩阵的视觉伺服控制算法[J]. 系统仿真学报, 2010, 22(12): 2934-2937.
- [12] 曾福振, 闵联营. 基于 ARM 和 Linux 的嵌入式平台的搭建[J]. 微型机与应用, 2010, 30(12): 51-53.