

基于 DM3730 的车位检测系统的实现

丁元舟 罗小巧 殷齐 严体曾

(华中师范大学 物理科学与技术学院 武汉 430079)

摘要: 以停车场监控视频作为车位检测算法的研究对象,该算法首先通过离线检测获取在不同环境下的车位背景相关的特征信息。接着提取出当前单车位区域作为感兴趣区(ROI),然后分别对其进行3种车位算法检测。该算法运行在ARM+DSP异构双核处理器DM3730平台和嵌入式Linux操作系统下,借用OpenCV中的图像处理函数和DSP运算进行图像处理,并将结果保存到嵌入式数据库中,实现了基于B/S网络模型的多车位检测。测试表明,用户只需通过浏览器访问该系统,就可以获得当前车位的占用情况,能够满足高实时性、高准确性的要求。

关键词: 嵌入式;Linux;车位检测;DM3730

中图分类号: TP302 文献标识码: A 国家标准学科分类代码: 510.4050

Implementation of parking space detection system based on DM3730

Ding Yuanzhou Luo Xiaoqiao Yin Qi Yan Tizeng

(Institute of physical science and technology, Central China Normal University, Wuhan 430079, China)

Abstract: In this paper, we treat the parking lot surveillance video as the research object of parking space detection algorithm. Firstly, the algorithm obtained the feature information of the related parking spaces through off-line testing in different environment background. Secondly, a single parking area is extracted as the region of interest (ROI). It is used three kinds of parking space detection algorithm to them respectively. The algorithm running on ARM + DSP heterogeneous dual-core processor DM3730 platform and embedded Linux operating system. Both OpenCV image processing function and DSP algorithms for image processing are used that the results can be saved to the embedded database, then more parking space detection based on B/S network model could be realized. Test results show that each user may use the browser to access the system, and get the information about the current parking space occupancy, which can satisfy the requirement of high real-time performance and high accuracy.

Keywords: embedded; Linux; parking space detection; DM3730

1 引言

目前,国内外已经建立了相对比较先进的停车场管理系统,其中比较实用的是停车诱导功能。停车诱导利用在交叉路口设置空车位信息显示屏和方向引导标志,能让车主快速找到合适的停车位^[1]。为了获取更为详细的车位占用信息,并实现自动化的管理,车位检测技术便是关键。车位检测方案在国内被大体分为两种,一类是使用物理特征的检测方案,主要是使用传感器类的检测设备^[2];另一类是使用视频、图像处理技术的方案^[3]。

北京紫光百会科技有限公司总经理刘鹏曾说过,停车场管理技术的趋势将是“全视频+电子支付”。当前是一个大数据时代,数据图像就是一种珍贵的资源。那么,基于视频的检测方法,必将会有一个良好的发展前景。

实际中,由于停车场复杂的环境因素,利用视频图像处理技术对停车场的车位进行检测,还存在很多的技术难点。例如,不同的天气会改变采集到的车辆亮度和颜色,行人的走动、车辆间的遮挡也会干扰检测结果;当监控的车位量比较多时,处理车位信息的速度不高,造成检测结果不能及时更新^[4];由于停车场监控摄像头安装的位置不同,摄像头的质量差等原因,采集到的图像会出现变形,对进一步进行图像处理造成十分不便。

本文针对当前停车场车位检测存在的上述问题,对采集到的视频图像进行预处理(不是本文重点介绍的内容),然后进行统一的车位算法检测。运用了嵌入式OpenCV, TI公司DSP的浮点计算优化算法和网络通信技术,实现了停车场的远程车位占用信息查询。该系统运行在嵌入式ARM+DSP平台上,大大的降低了系统的开发

成本和功耗,而且系统稳定可靠。在当前“停车难”问题突出的时代,必然会使停车场远程车位检测系统迅猛发展^[4]。

2 系统总体方案设计

2.1 系统总体设计

本文的硬件电路采用 Embest DevKit8500 开发板,该开发板出厂是用来评估 DM3730 性能,因此引出了大部分的接口,用户可以先在该评估板上完成功能设计,然后裁剪掉不需要的硬件,最后修改成自己的产品。本文用的硬件框图如图 1 所示。其中,MT29C4G96MAZAPCJA-5 芯片集成了 NAND Flash 和 SDRAM,大小分别为 512 MB。TPS65930 主要是用于电源管理的芯片,功能包含电源管理控制器、时钟、嵌入式时钟管理等。总体上说,DM3730 处理器相比前一代 OMAP3530 处理器提高了多处理器性能并减少了延迟,增强了系统实时处理性能^[5]。另外,ARM 与 DSP 对 DM3730 设备外设和内部资源的控制范围不同,主要还是 ARM 占主导地位。

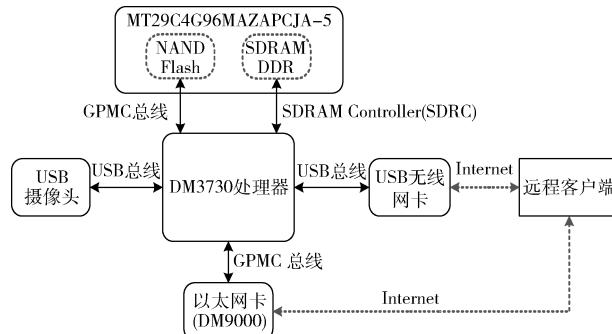


图 1 系统硬件组成框图

本文采用摄像头监控多个车位,提取其中的单车位图像处理。由于摄像头拍摄的角度和高度不同,会导致出现严重的车位遮挡情况。如果选择在合适的位置安装摄像头,可以减小遮挡的面积。实际应用中,除了对安装提出严格要求,建议使用工业级高清摄像头。本文是停车场视频监控系统的雏形,采用的是普通的 USB 摄像头。对于固定的障碍物、行人、天气情况,我们可以通过相应的算法排除^[6]。系统总体功能方案图如图 2 所示。

图 2 中,灰色部分表示车位状态检测算法。图中所示的视频仓库是本文进行车位检测算法的数据来源,ARM 和 DSP 共享这个仓库。车位占用状况的判断选用 3 个标准:1)当前图像帧中的车位信息与预存背景图像的差影方差;2)它们的线性相关度;3)当前图像帧的车位信息的边缘点数。为了提高系统的实时性,本系统将线性相关度检测算法交给 DSP 运行,让 ARM 与 DSP 的共同工作,充分发挥 DSP 的强项性能。两个内核之间通过 TI 提供的 CE (codec engine) 实现通信和交互^[7]。

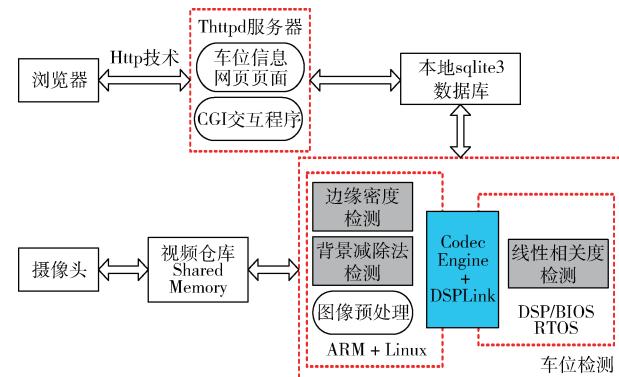


图 2 系统总体功能图

2.2 车位检测算法的确定

为了能够在实际环境中准确的检测出车位的占用信息,本文将这些干扰信息考虑在内,提出了组合 3 种常规检测方法的方案。通过实验分析,总结 3 种检测方法的差异如表 1 所示。

表 1 车位检测方案的比较

车位情况	方差检测	线性相关度检测	边缘点数检测
有车 (有天气干扰)	33.4~51.2	0.07~0.32	300~660
无车 (有天气干扰)	0.4~1.6	~0.99	47~200
行人路过 车位区域	先↑随后↓ 至稳定	先↓随后↑ 至稳定	出现较大波动
车位被阴影 完全覆盖	先↑随后 稳定	存在较高的 线性相关性	出现较 小波动
车位被阴影 部分覆盖	先↑随后稳定	存在一定的 线性相关性	出现一定 波动

注:选取第 1 帧无车情况作为背景,获取的图像缩放成大小为 640×480,↑ 表示值增大,↓ 表示值减小

由表 1 可知,差影图像的方差因天气情况、地面阴影等外界干扰问题不能很好的排除。另外,水迹也会影响检测的准确度。相关度判别法可以排除天气的影响,但是对阴影的检测效果不理想^[8]。车辆拥有比较丰富的边缘信息,在环境条件变化时边缘信息将被保留。尽管有阴影,但阴影的边缘点数相对较少,和有车停留的情形相比,差别比较大。通过对比分析,本文设计的车位检测算法同时基于上面 3 种判据(以下简称特征参数),通过特征参数 1~3 可以排除提及的常见外界干扰,高准确率的判断车位状态变化。

最终确定的车位检测方案框图如图 3 所示。从图中可以看出,最终的检测算法由在线检测和离线检测两部分组

成。在线检测部分将检测结果存入 SQLite 数据库中,供图像显示模块调用。离线检测部分主要是计算出 3 种检测方案中特征阈值,以及图像的偏转角度,供在线检测部分中的算法调用。

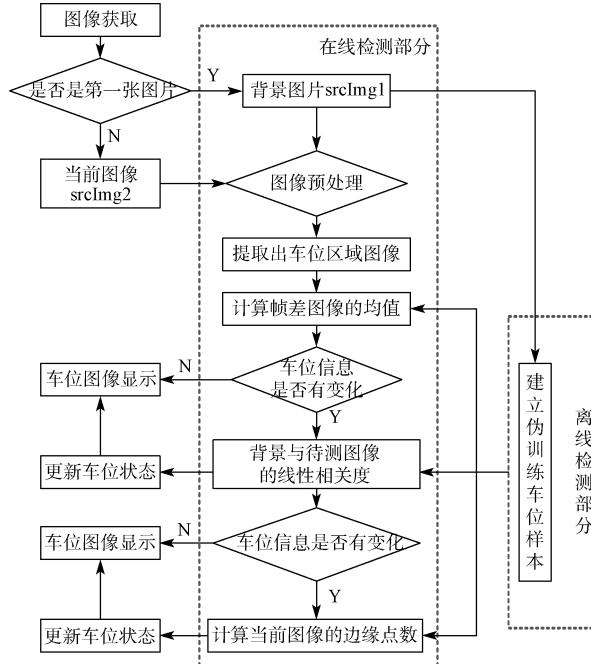


图 3 车位检测方案总体框图

车位在线检测算法按照背景减除法检测→线性相关度检测→边缘检测的顺序执行,后一种的判据是对前一种判据的更进一步的判定,排除掉一些特定的外界干扰。

针对停车场车位检测这一目标,如果只是简单利用现成的或者第三方的算法库,如 OpenCV,可以很方便的实现这一功能。而且,对开发人员来说,只需了解 ARM 端即可。DM3730 可以像普通的 ARM 一样创建这程序,但是在资源有限的嵌入式系统中,如果要保证检测的效率和实时性,那么就不能这么做。DVSDK 考虑到开发人员的不同,要对 DSP 进行开发,可以通过 CE 组件和规范的 DSP 算法接口实现。开发 DSP 算法和一般常见的单核 DSP 类似,可以使用 CCS 开发环境开发,最后变成 xDM 的接口即可^[9]。另外,也可以使用 DVSDK 中提供的 DSP 编译器进行开发,DVSDK 这一方面真是用心良苦。

3 系统实现

3.1 Ubuntu 上开发环境搭建

目前,嵌入式应用开发主要使用交叉开发,即 PC+嵌入式开发板的开发模式。本文采用虚拟机安装 Ubuntu-10.04 作为开发主机的系统。为了开发方便,安装 VMware Tools,并修改/etc/gdm/custom.conf 文件,破解主机的 root 权限。

在开发系统中,除了在 X86 计算机平台上需要安装

OpenCV 外,还需要将其移植到 DM3730 中,让应用程序能在 ARM 上调用经过交叉编译过的 OpenCV 库。考虑到目前针对嵌入式 DSP 的计算机视觉库(EMCV)还不够完善,故不做 DSP 上 OpenCV 移植^[10]。

3.2 DM3730 上 Linux 环境构建

下载 arm-2009q1-203-arm-none-linux-gnueabi. bin, 执行安装,接着在/etc/profile 文件中添加交叉编译工具的路径。

TI 的 DVSDK 把全部的包集合在一起,一次安装就可以将所有的包安装完,方便了许多,相比以前的软件包确实人性化了一点。安装过程如下:进入 <http://www.ti.com/tool/linuxdv-sdk-dm3730>, 下载 dvsdk_dm3730-evm_4_00_00_22_SetupLinux, 安装方法和交叉编译器类似。安装完成后,进入目录/home/mm123456/dm3730/dv-sdk4_00 后执行 1 s 返回内容如图 4 所示。

```

root@ubuntu:/home/mm123456/dm3730/dv-sdk4_00# ls
bin
edma3lld_01_11_00_03
biosutils_1_02_02
framework-components_2_25_03_07
c6accel_1_01_00_01
gstreamer-ti_svnr822
c6run_0_94_04_04
cgt6x_6_1_14
codec-engine_2_26_01_09
codecs-omap3530_4_00_00_00
dmai_2_20_00_10
docs
dspbios_5_41_03_17
dsplink_1_65_00_02
dvsdk_4_00_00_22_dm3730_Release_Notes.pdf
dvsdk-demos_4_00_00_18
dvtb_4_20_12
xdais_6_26_01_03
xdctools_3_16_03_36
  
```

图 4 DVSDK 安装结果

整个 CE 框架包含在 DVSDK 开发包中,该开发包除了包含 CE 框架内容,还包含 DSP 平台编译器、编译相关的工具、板子支持包(PSP)等。因为 TI 的这套框架比较繁琐,故而将很多细节都隐藏起来,比如 Codec、Server、APP 端的编译过程,使用起来很方便。

使用的 OpenCV 版本是 2.4.2,为了让 OpenCV 可以处理 jpeg 格式的图片,需要提前编译生成 libjpeg 的库文件和头文件^[11]。并将这些生成的文件复制到交叉编译器的目录下,以后编译程序就不需要指定图像库的路径。安装成功以后,可以在 prefix 指定的目录找到生成的 libopencv_*.so 库和头文件,将其中的库文件复制到 DM3730 系统中,并使应用程序可以访问到。

移植 Thhttpd 与 SQLite 基本一样,一般依照:make→遇到 error→修改后重新 make 的顺序进行。最后复制所需的依赖库和应用程序到目标板下,可以在 PC 上利用 ldd 查询其依赖的库文件^[12]。如果开发板中存在上述文件,依赖库就不需要复制,缺少的库可以从交叉编译器的路径下查找并添加。另外,在/etc/init.d/rc 添加命令/bin/thhttpd &, 配置 thhttpd 后台开机启动。

3.3 网关服务程序设计

设计一个静态页面给用户提供一个入口,只要用户点击确认按钮之后,DM3730 服务器就会持续发送更新的内

容。本系统采用 CGI 技术实现这些动态页面的输出^[13]。大部分 CGI 程序用作接收来自表单传送来的数据,客户端会把这些信息添加到由 action 标记的网络地址的后面,并用“?”隔开 CGI 程序名。

CGI 是 WEB 服务器和应用程序进行沟通的桥梁,如图 5 所示为 CGI 程序执行流程图。CGI 通过对数据库 SQLite 的读写,动态产生 HTML 源代码输出,这样车位信息页面内容就能够动态的改变。程序的执行流程如图 5 所示。

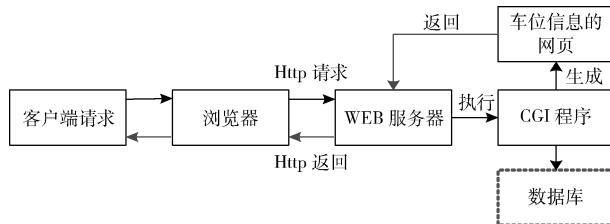


图 5 程序执行流程

3.4 车位检测程序的设计

经过实际测试知,在 ARM 端的线性相关度检测的时间周期相对于其他两种检测方法耗时最长,故将其放入 DSP 核中处理。线性相关度检测是通过计算这 2 帧图像图像的相关系数来进行辨别,相关系数可以反映两组数据或图像(X, Y)的关系密切程度^[14],计算公式如下。

$$r_{xy} = \frac{\sum (x_i - \bar{X})(y_i - \bar{Y})}{\sqrt{\sum (x_i - \bar{X})(x_i - \bar{X})} \sqrt{\sum (y_i - \bar{Y})(y_i - \bar{Y})}}$$

式中: $i=0, 1, 2, \dots, n-1$, \bar{X} 、 \bar{Y} 分别为数组 X 和 Y 的均值,可以在 ARM 端调用 OpenCV 中的 cvAvgSdv 函数得到。

另外,开发 ARM+DSP 平台的车位检测的应用程序还需要编译以下程序:DSP 算法程序(Codec 程序)、DSP 服务器程序(Server 程序)和应用程序。如此复杂的流程,DVSDK 的示例程序是快速开发 ARM+DSP 的好方法,在 codec-engine_2_26_01_09/examples 目录下提供了一些相关的示例程序。具体的设计流程如下。

1) Codec 程序开发。在编译 codec 程序之前,先要配置 xdcpaths.mak。主要是指定 GPP 为 LINUX_GCC,设定 xdc 用到的环境变量,注意使用绝对路径。配置 arm 和 dsp 编译器路径,以及其它的相关组件目录。开发 TI 应用最一般的做法,使用例程框架,然后在此基础上更改。下面将线性相关度检测的 DSP 算法添加到 Codec Engine,并按照标准的格式填充到 codecs 中。复制 codecs 目录下的 videnc_copy,更名为 pkgdection,切换至该目录下,将小写的 copy 更换为 pkgdection,将大写的 COPY 更换为 PKGDECTION。打开 pkgdection.c,将原先的 VIDENCCOPY_TI_IALG 算法部分注释掉,换成线性相关度检测的算法。再执行 make 编译 codec 库,这时在 lib 目

录里可以找到 *.a64P 等库文件。

2) Server 程序开发。Server 端程序可以简单的理解成一个链接调用 Codec 库的可执行文件,同 Codec 程序开发一样,复制例程,然后在例程的基础上修改。步骤如下:复制 servers 目录下的 all_codecs,更名为 pkgdection。切换至该目录下,更改所有的文件名称,与 Codec 程序一致,如 package.bld 中的 all_codecs 全部替代为 pkgdection,改 serverName = “all” 为 serverName = “pkgdection”。进入 pkgdection 的目录下,先执行 make clean,再执行 make 编译 server 库。这时在工程的 bin/ti_platforms_evm3530 目录下可以找到相应的库文件。

3) 应用程序开发。在 CE 应用框架下,通过 VISA 接口调用那些符合 xDM 算法标准的组件。即能支持算法组件在本地(GPP),也可以在远端(DSP)运行^[15]。复制 apps 目录下的 video_copy 目录,更名为 pkgdection。切换至该目录下,更改文件 remote.cfg,local.cfg,app.c,使其同上一步的程序命名相对应,app.c 中主要实现车位检测的两个判别标准。进入 video_pkgdection 的目录下,先清除一遍文件,再执行 make 编译程序,在子目录下可以生成最终的应用程序。

4) 拷贝生成的相应库和应用程序至 DM3730 开发板,启动应用程序前,需要加载好 CMEM 模块和 DSPLINK 模块。CMEM 是 Linux 驱动模块,负责管理一块内存区域。DSPLINK 负责 ARM 端和 DSP 端的通信管理。

4 系统测试

由于现场实验条件的限制,车位检测只能从事先准备好的视频中提取,模拟摄像头采集到的数据源。某一时刻背景与前景图像的截图如图 6 所示。一共对 14 个车位进行轮询处理,处理第 5 个车位时提取出的单车位图像如图 7 所示。

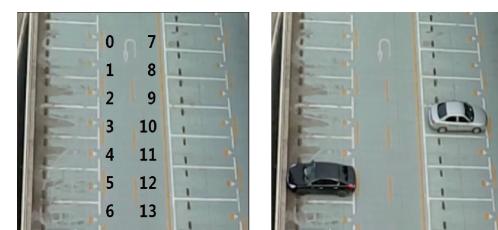


图 6 某时刻的背景与前景图像



图 7 处理中的第 5 号车位图像

另外在程序的执行效率上,通过在 DM3730 上运行 time 命令查看得知,所设计的车位检测算法处理 1 帧高清的 640×480 图像只需 0.5 s 左右的时间,如图 7 所示。该

处理时间与采用的交叉编译器, DSP 算法优化^[15], 以及 OpenCV 算法优化有很大的关系。

```
real    0m 0.54s
user    0m 0.53s
sys     0m 0.00s
root@dm37x-evm:/media/mmcblk0p1/PSRVSS/www/cgi-bin#
```

图 8 算法运行效率

打开车位占用查询页面, 点击“确认选择”按钮, 效果如图 9 所示。该网页由 CGI 生成, 绿色区域表示空车位, 灰色表示车位被占用, 和实际的情况对比, 准确率达到 100%。另外网页也输出了空车位的车位号和其它的车位信息, 方便车主找到空车位。

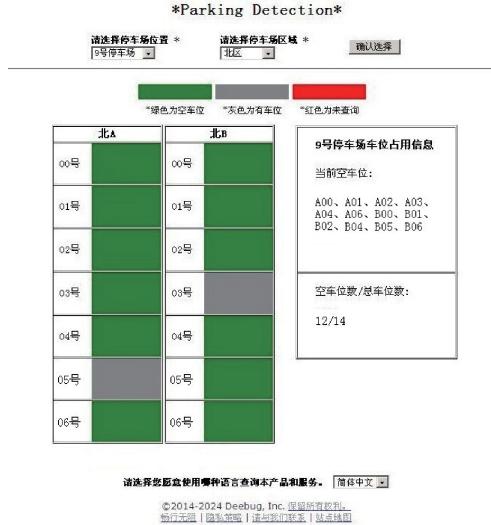


图 9 车位占用查询效果图

5 结 论

如今科学技术正是突飞猛进, 软硬件的配置已经接近 PC 的世界, 作者相信未来的 PC 世界必将会被嵌入式系统取代。本文结合了 OpenCV 技术和 DVSDK 技术, 实现了车位检测的 3 种判据。并且采用了 HTML 网页编程技术和 CGI 技术实现用户与停车场车位检测系统的动态交互。最终实现了车位占用查询的功能, 让用户只需一个浏览器便可以知道结果。经测试, 系统运行高效稳定, 实时性好。

参考文献

- [1] 万婷婷. 智能停车场车位检测算法的研究[D]. 北京: 北京工业大学, 2010.

- [2] 赵中琦, 陈永锐, 易卫东. 基于磁阻传感器的无线车辆检测器[J]. 电子测量技术, 2013, 36(1): 12-17.
- [3] 孙建延, 时伟, 李建峰. 基于图像的嵌入式车位检测系统设计[J]. 制造业自动化, 2012, 34(10): 67-70.
- [4] 李国辉. 空闲车位的嵌入式检测系统设计[D]. 北京: 北方工业大学, 2013.
- [5] Texas Instruments Incorporated. DM3730, DM3725 Digital media processors[Z]. SPRS685B, 2010.
- [6] WANG W J, GAO M. Vehicle detection and counting in traffic video based on OpenCV [J], Applied Mechanics and Materials, 2013 (361-363) : 2232-2235.
- [7] 刘岩俊. 嵌入式 DSP 图像处理系统设计与实现[J]. 国外电子测量技术, 2013, 32(9): 11-14.
- [8] 罗小巧, 姜龙, 瞿少成, 等. 基于视频的停车场车位监控算法研究[J]. 电子测量技术, 2012, 35(2): 33-36.
- [9] 王冬, 李登科. 基于双 DSP 的信息处理机设计[J]. 国外电子测量技术, 2013, 32(6): 89-92.
- [10] 袁敏, 张仁杰, 邵娜. 基于 Linux 的 ARM9 多路视频监控系统设计[J]. 国外电子测量技术, 2009, 28(1): 42-44.
- [11] 赵庆平, 陈得宝, 姜恩华, 等. 一种改进权重的非局部均值图像去噪算法[J]. 电子测量与仪器学报, 2014, 28(3): 334-339.
- [12] 苏慧娟, 于正林, 张桂林. 基于 OpenCV 的图像处理[J]. 科技资讯, 2014, 12(8): 18-19.
- [13] 沈绮. 浅谈 Web 数据库访问技术[J]. 电脑知识与技术, 2009, 5(18): 4648-4649.
- [14] 高晶, 吴育峰, 吴昆, 等. 基于角点检测的图像匹配算法[J]. 仪器仪表学报, 2013, 34(8): 1717-1725.
- [15] 曾蓉, 许永辉. 基于 TMS320DM642 的嵌入式 Web 服务器设计 [J]. 电子测量技术, 2013, 36(5): 72-77.

作者简介

丁元舟(通讯作者), 1988 年出生, 在读硕士, 主要研究方向为嵌入式系统与应用等。

E-mail: dyzok88@126.com

罗小巧, 1972 年出生, 副教授, 主要研究方向为嵌入式系统和智能信息处理等。