

基于树莓派的智能监控系统设计与实现

张 翊¹, 张 琦², 陈梦丹¹, 王 轩¹, 张宏伟¹

(1. 西安工程大学 电子信息学院, 西安 710048; 2. 西安慧金科技有限公司, 西安 710061)

摘要: 随着人们的安全意识越来越强烈, 监控系统在生活中得到广泛的关注和应用; 针对市场上绝大多数监控系统智能化程度低, 视频保存占用内存大, 实时性差, 对闯入目标不能及时报警等缺陷, 设计与开发基于树莓派的智能监控系统; 选用树莓派4B 搭建硬件平台, 搭载 Linux 系统, 基于 Python 开发环境, 结合图像采集模块, 采用关联帧差分运动目标检测算法作为核心算法, 实现运动目标检测; 当系统检测到有目标闯入视野范围内时, 开始录制目标闯入后的 400 帧视频, 并将所录制的视频发送到指定邮箱, 实现报警功能; 在正常光线和昏暗光线下对系统功能进行测试, 实验结果表明在不同光线强度、不同目标闯入情况下, 系统能够准确识别目标闯入, 实现监控、报警功能, 实时性强、稳定性高, 满足设计需求。

关键词: 监控系统; 树莓派 4 B; Python 开发环境; 图像采集; 关联帧差分运动; 目标检测; 报警功能

Design and Implementation of Intelligent Monitoring System Based on Raspberry PI

ZHANG Yue¹, ZHANG Qi², CHEN Mengdan¹, WANG Xuan¹, ZHANG Hongwei¹

(1. School of Electronics and Information, Xi'an Polytechnic University, Xi'an 710048, China;

2. Xi'an Huijin Technology Corporation, Xi'an 710061, China)

Abstract: With the increasing awareness of security, monitoring system has been widely concerned and applied in life. Aiming at the shortcomings of most monitoring systems in the market, such as low intelligence degree, large memory consumption for video preservation, poor real-time performance, and failure to timely alarm the intrusion target, an intelligent monitoring system based on Raspberry PI is designed and developed. Raspberry PI 4B is selected to build the hardware platform, which is equipped with Linux system, based on Python development environment, and combined with the image acquisition module, the associative frame differential motion detection algorithm is used as the core algorithm to achieve the moving target detection. When the system detects the target breaks into the field of the view, it not only starts to record 400 frames of video after the target intruding, but also sends the recorded video to the specified mailbox to realize the alarm function. The system performance is tested under normal light and dim light. The experimental results show that the system can accurately identify the target intrusion in the case of different light intensity and target intrusion, it can realize the monitoring and alarm functions, which has strong real-time and high stability, it satisfies the design requirements.

Keywords: monitoring system; Raspberry PI 4 B; Python development environment; image acquisition; associate frame differential motion; target detection; alarm function

0 引言

随着生活环境和生活质量的提高, 人们的隐私安全保障意识越来越强, 因此监控系统在日常生活中使用的领域越来越广, 例如道路交通监控、刑事侦查、智能家居等领域。但是, 目前市场上绝大多数监控系统仅仅只是简单地将摄像头所拍摄到的内容录制并存储下来。它的优点是能够将所有的视频全部保存, 并且可以随时查阅, 然而其弊端也非常明显, 例如: 保存的视频过多会占据大量的存储空间; 智能化程度不高, 对于有失窃、外来人员闯入等危险行为不能进行及时报警。因此, 自带报警功能的智能监控系统受到越来越多人的青睐。

最初的第一代监控系统是基于闭路电视实现的模拟图像监控设备^[1], 此方案是借助电缆把图像采集设备和监控显示端相连, 从而实现模拟视频数据的发送。20世纪90年代, 数字视频压缩编码技术正式被应用于监控领域, 出现了第二代数字视频监控系统。它可以将模拟数据转化为数字数据, 在突破传送电缆线长度限制的情况下, 视频采集清晰度和系统的抗干扰性能得到有效提高。但由于第二代监控设备采用计算机作为图像数据运算和视频显示端, 传输距离仍然受限, 整套监控系统体积大、成本高, 维护困难^[2]。随着图像采集设备、视频编码器、中央处理器等设备的不断改进与发展, 视频监控领域也进入到第三代智能

收稿日期:2022-11-03; 修回日期:2022-12-12。

基金项目:纺织工业联合会科技指导项目(2020111)。

作者简介:张 翊(1990-),女,河南三门峡人,工学硕士,工程师,主要从事深度学习的图像检测、智能装备开发与应用等方向的研究。

通讯作者:张 琦(1988-),男,河南三门峡人,工学硕士,工程师,主要从事智能装备开发与应用等方向的研究。

引用格式:张 翊,张 琦,陈梦丹,等. 基于树莓派的智能监控系统设计与实现[J]. 计算机测量与控制,2023,31(8):122-127,134.

投稿网址:www.jsjlykz.com

视频监控时代。智能监控系统以图像处理、模式识别和计算机视觉技术为基础, 在监控系统中增加智能视频分析模块, 借助计算机强大的数据处理能力过滤掉视频画面无用或干扰信息、自动识别不同物体, 分析抽取视频源中关键有效信息, 快速准确的定位事故现场, 判断监控画面中的异常情况, 并以最快和最佳的方式发出警报或触发其它动作, 从而有效进行事前预警, 事中处理, 事后及时取证的全自动、全天候、实时监控的智能系统。

智能监控系统市场前景广阔, 国外对其研究的相对较早。卡内基梅隆大学 VSAM 团队在 1997 年研发了一种端到端的多摄像头监控系统, 该系统能够自动收集和传播实时信息^[3]。2005 年, 美国 Tyco 公司为道达尔炼油厂开发的“区域监控器”软件模块, 可以实时显示炼油厂内各站点数据, 且能详细统计现场工作人员出勤情况^[4]。2020 年, AXIS 针对斯洛伐克 Mochovce 核电站研发的智能监控系统, 将摄像头和报警系统互连。该系统一方面能够在发现河中有漂浮物的时候自动报警; 另一方面能够监控现场生产设备的状态, 当发生故障或缺陷时, 及时报警。目前, 除 CMU、麻省理工外, 奥地利 Graz 理工大学的嵌入式智能摄像机研究小组、IBMS3 (smart surveillance system)、Intel IRISNET 项目等, 都是各行业中的佼佼者。

相对于国外, 我国的监控系统研究起步较晚, 早期的一些产品大多用于交通监控、商场监控、保密室监控等。截止到 2021 年, 我国超过 1.76 亿个监控摄像头用于城市监控与报警系统^[5]。近年来, 随着海康威视、大华股份、汉邦高科等公司的陆续上市, 国产的智能监控产品越来越多。例如海康威视, 针对发电、电网、石油化工产业、煤矿产业等研发了安全生产管理平台, 这些监控平台以视频采集为基础, 结合人工智能、物联网技术实现智能监控功能, 在提高工厂效率的同时有效预防生产事故^[6]。这些大企业研发的智能监控系统大多用于大型工厂, 几乎没有针对家庭安防、保密室等小型场所、体积小、价格低廉、实时性高、稳定性强、带自动报警功能的智能监控系统。针对此, 本文研究设计一款小型智能监控系统, 该系统选用树莓派 4 B 搭建硬件平台, 搭载 Linux 操作系统, 基于 Python 开发环境, 结合图像采集模块, 采用关联帧差分运动目标检测算法作为核心算法, 实现运动目标检测。当系统检测到有目标闯入视野范围内时, 开始录制目标闯入后的视频, 并将目标闯入后 400 帧视频发送到指定邮箱, 实现报警功能。

1 系统结构及原理

智能监控系统主要由主控模块、显示模块、图像采集模块、电源模块、邮箱系统共五部分组成, 系统总体结构如图 1 所示。

主控模块是整个控制系统的核心部分, 负责系统控制、数据接收、数据分析和计算、数据传输等工作。图像采集模块作为系统的数据输入, 通过摄像头的拍摄, 把图像的数据传输到主控模块, 完成图像采集功能。显示模块用来

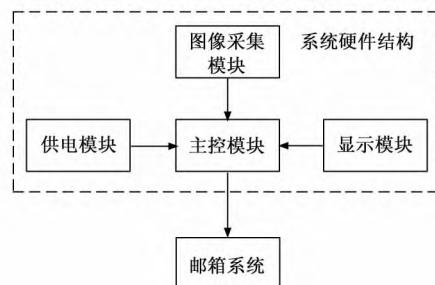


图 1 智能监控系统结构设计框图

显示图像采集模块拍摄到的实时画面。电源模块为系统提供电能, 使系统能够正常工作。邮箱系统用来接收有目标闯入后系统录制的 400 帧视频, 实现系统报警功能。

1.1 系统硬件选型

1) 主控模块。系统主控模块可采用 AT89C51 单片机、FPGA 或 STM32 系列嵌入式微控制器, 其优势在于体积小、运算速度快、价格低廉, 但系统集成度不高, 需要设计多个外部电路和连接多种外部设备, 系统复杂度高, 在日常使用中容易出现故障。而基于 Linux 的树莓派 4 B 控制系统具有 4 GB 的 RAM 和高性能 64 位四核处理器, 能够通过运行相应的操作系统, 实现远程遥控和云端管理, 通过一对 micro-HDMI 端口支持分辨率高达 4 K 的双显示屏, 支持 HEVC 视频 4 Kp60 硬解码, 高达双频 2.4/5.0 GHz 无线局域网, 具有蓝牙 5.0、千兆以太网、USB 3.0 和 PoE 功能, 并能够与 Camera V2 摄像头模块完全相互兼容^[7-8]。同时, 其价格便宜, 集成度高, 运算速度快, 扩展性丰富, 使用方便。因此, 将其作为智能监控系统的主控模块, 能够在降低系统复杂程度的基础上, 降低系统成本。

2) 图像采集模块。树莓派具有良好的扩展功能, 能够与 Camera V2 摄像头模块完全相互兼容, 因此图像采集模块可采用体积较小的 CameraV2 照相机模块。其核心元件是 1 个 500 W 像素的 CMOS 传感器, 最大分辨率为 2 592 * 1 944, 支持每秒高达 30 帧的 1 080 p 视频拍摄, 可以用一条 15 芯的排线与树莓派 4B 控制系统专用的 CSI 接口直连, 能够有效提高系统集成度。

3) 显示模块。系统显示可以使用 VNC 远程连接的方式把树莓派的系统界面远程连接在 PC 端, 也可以使用 IPS 高清显示屏, 通过 HDMI 连接线与核心控制模块进行连接。本研究选用 MP13508 显示屏作为智能监控系统显示模块。该显示模块采用 3.5 英寸电阻触摸 TFT 屏, 分辨率为 480 * 320, 并且包含 SPI + HDMI 接口, 可与树莓派 4 B 控制系统直连, 具有结构简单、体积小、重量轻、便于隐蔽和安装等优势, 能够满足系统设计需求。

4) 电源模块。电源模块选用具有 Type C 接口且能够把 100~240 V、50/60 Hz、0.6 A 的交流电转换为 5 V、3 A 的电源适配器。Type C 接口可与树莓派 4 B 控制系统直连, 结构简单, 使用方便。

2 算法分析

2.1 图像预处理算法分析

图像预处理的主要目的是消除图像中无关的信息，恢复有用的真实信息，增强有关信息的可检测性和最大限度地简化数据，从而提高特征提取、图像分割、匹配和识别的可靠性^[9]。系统对图像预处理包括图像灰度化处理和滤波处理。

灰度化处理是将彩色图像转换为灰度图像的过程，是图像处理的根基。相对于彩色图像，灰度图像占内存更小，运算速度更快，且图像经过灰度化处理后可以在视觉上增加对比，突出目标区域。因此，系统将图像灰度化处理作为图像预处理初始步骤。常见的图像灰度化处理方法有分量法、平均值法和加权平均法。分量法是根据应用需要直接选择彩色图像 R、G、B 三分量中某一个分量，将其作为灰度值进行处理，通常选择 3 个分量中数值最大的分量。平均值法是计算彩色图像 R、G、B 三分量全部数值的平均值，将其作为灰度值，进而获得灰度图像。加权平均法是根据人眼对绿色敏感最高，对蓝色敏感最低，将彩色图像的 R、G、B 三分量以不同的权重进行加权平均，如公式(1) 所示：

$$f(x, y) = 0.299 * R(x, y) + 0.578 * G(x, y) + 0.114 * B(x, y) \quad (1)$$

式中， $R(x, y)$ 、 $G(x, y)$ 、 $B(x, y)$ 分别表示像素点 (x, y) 的 R、G、B 值， $f(x, y)$ 表示图像像素点 (x, y) 处理后的灰度值。

通常分量法转换的灰度图像亮度较高，平均值法产生的灰度图像比较柔和，加权平均法得到的灰度图像效果最好。因此，选择加权平均法作为图像灰度化处理方法。

经过灰度化处理的图像，一般存在噪声、阴影等问题，为了提升图像质量，更好地把目标与背景进行分割，减少图像噪声、阴影对运动目标检测效果的影响，需要对图像进行滤波处理。常用的滤波方法有均值滤波、高斯滤波和中值滤波。

均值滤波是图像处理中常用的手段，从频率域观点来看均值滤波是一种低通滤波器，高频信号将会去掉。理想的均值滤波是用每个像素和它周围像素计算出来的平均值替换图像中每个像素，从而帮助消除图像尖锐噪声，实现图像平滑，模糊等功能。中值滤波是将像素点灰度值设置为相邻窗口范围内各点中值，属于非线性平滑滤波的一种，一定程度上中值更能反映规定范围内的真实值，使得邻域范围内像素值近似于真实值从而消除孤立的噪声点。高斯滤波是对图像进行加权平均的过程，图像中每一个像素点的值由其自身和邻域内其它像素值经过加权平均后得到，类似于卷积。高斯滤波属于线性平滑滤波的一种，滤波后的图像平滑程度取决于标准差，它输出的是领域像素的加权平均值，同时离中心越近的像素权重越高。综合考虑，高斯滤波保留细节的能力相对其余两种方法会更好一点，并且高斯滤波可以通过加权值的改变，调节其性能。因此，

系统选择高斯滤波算法对图像进行滤波处理。

2.2 运动目标检测算法分析

智能监控系统实现的基础是运动目标检测，而运动目标检测的核心在于通过分析视频图像序列，把运动目标从实时变化的背景图像中快速准确地分离出来^[10-11]。在不考虑光照变化、噪声干扰的情况下，能引起序列图像中相邻两帧对应像素的像素值发生变化的因素只有目标的运动。因此，根据序列图像中相邻两帧或者几帧中对应像素点像素值的变化可以得到运动目标的运动信息。经典的运动目标检测算法有：光流法、背景差分法和帧间差分法^[12]。

基于光流法的运动目标检测利用运动目标相对于背景图像运动时存在运动位置矢量场来检测出运动目标的位置^[13]。光流法抗噪性差，容易受噪声、遮挡、光照强度、物体阴影等因素影响，且算法复杂，运算时间长，实时效果差^[14]，难以满足系统实时性要求。

背景差分法是将当前每一帧图像的像素值与之前存好的背景图像的像素值进行差分，差分图像与预先设定的阈值进行比较，如果差分图像像素值大于阈值则为运动目标，如果小于阈值，则为背景区域^[15]。背景差分法能够较快地检测到运动目标，实时性高，但容易受到环境干扰，背景帧的构造和更新将花费大量的时间^[16]，不适用于背景快变化的场景。

帧间差分法又被称时间差分法，是一种利用序列图像相邻帧的相关性，将相邻的两帧或者几帧图像进行差分计算并阈值化以获取运动目标检测的方法^[17]。帧间图像信号的变化往往包含了丰富的目标运动信息，这是视频图像中对运动目标进行分割的基础^[18-20]。由于运动目标与背景在像素值上的差异，且运动目标在不同的图像帧中的位置不同，当视频中存在移动物体时，相邻帧之间对应像素灰度值会有差异，求取相邻帧灰度差的绝对值，如果没有运动目标存在，则差值图像表现出来均为 0，如果有运动目标存在，则差值图像表现为非 0，这样就可判断是否有运动目标存在，同时大致计算得到运动目标的位置、轮廓和移动路径等。帧间差分法的优点是算法简单，程序实现复杂度低，稳定性高，对光线等场景变化不敏感，适应性强，适用场景多。缺点是不能提取运动目标完整图像，只能提取目标边界且边界轮廓粗，比实际目标大，目标内部有“空洞”；对于运动速度较快的目标，容易出现“鬼影”现象或被检测为两个不同的运动目标。对于运动速度缓慢的目标不敏感^[21]，当目标在视频流中前后两帧几乎完全重叠时，会出现漏检现象，系统检测准确性不高，需要对其进行改进，才能满足智能监控系统设计要求。

2.2.1 基于关联帧差分运动目标检测算法

三帧差分法 (TFMD, three-frame difference method) 是帧间差分法的一种，是通过对图像序列连续几帧图像做差分运算从而获取运动目标轮廓的方法。该方法在一定程度上优化了运动物体双边，粗轮廓的现象，相比之下，TFMD 比相邻帧间差法更适应于物体移动速度较快的情况，

且检测精度更高。TFMD 就是取连续的 3 帧图像进行 2 次差分运算, 再将 2 次差分得到的结果进行与运算即可得到运动目标^[22]。相对于其它几种常用的运动目标检测算法而言, 三帧差分法具有计算量小、抗干扰能力强等优点, 能够得到比两帧差分运动目标检测更精确的结果, 但检测结果仍然会出现空洞现象。为了能够提高智能监控系统的准确性, 论文研究的智能监控系统采用基于关联帧差分运动目标检测算法, 该算法的基础是 TFMD, 具体实现原理如图 2 所示。

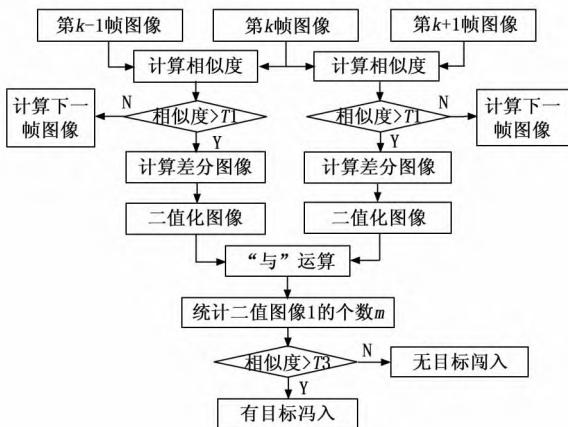


图 2 基于关联帧差分运动目标检测算法原理图

记 $f_k(x, y)$ 为序列图像第 k 帧图像中坐标为 (x, y) 像素点的像素值; $f_{k-1}(x, y)$ 为序列图像第 $k - 1$ 帧图像中坐标为 (x, y) 像素点的像素值; $f_{k+1}(x, y)$ 为序列图像第 $k + 1$ 帧图像中坐标为 (x, y) 像素点的像素值。

1) 通过相邻两帧图像颜色特征的巴氏距离, 根据式(2)算相邻两帧图像之间的相似度:

$$S(\mathbf{F}_k, \mathbf{F}_{k+1}) = \sqrt{1 - \sum_i \frac{\sqrt{\mathbf{F}_k(i) \cdot \mathbf{F}_{k+1}(i)}}{\sum_i \mathbf{F}_k(i) \cdot \sum_i \mathbf{F}_{k+1}(i)}} \quad (2)$$

其中: $\mathbf{F}_k(i)$ 和 $\mathbf{F}_{k+1}(i)$ 为提取出来的第 k 帧和第 $k + 1$ 帧图像的颜色特征向量; S 的值越接近 1, 表明两帧图像越相似。

2) 当第 k 帧和第 $k - 1$ 帧图像的相似度大于设定阈值 T_1 时, 再用式(3)、式(4)算相邻 2 帧图像之间的差分值:

$$D_k(x, y) = |f_{k+1}(x, y) - f_k(x, y)| \quad (3)$$

$$D_{k-1}(x, y) = |f_k(x, y) - f_{k-1}(x, y)| \quad (4)$$

3) 考虑到光照、噪声等外部因素造成的图像像素值的微弱变化, 可利用最大类间方差 (OTSU) 得到自适应阈值 T_2 , 将差分图像和该阈值 T_2 作比较, 当差值大于该阈值时认为该像素点属于运动目标, 否则属于背景。对差分结果进行二值化处理, 如式(5)、式(6)所示:

$$d_k(x, y) = \begin{cases} 1, & D_k(x, y) > T_2 \\ 0, & D_k(x, y) \leqslant T_2 \end{cases} \quad (5)$$

$$d_{k-1}(x, y) = \begin{cases} 1, & D_{k-1}(x, y) > T_2 \\ 0, & D_{k-1}(x, y) \leqslant T_2 \end{cases} \quad (6)$$

4) 对 $d_k(x, y)$ 和 $d_{k-1}(x, y)$ 取与运算, 得到含有运动对象和两帧差分图像重叠部分的二值图像 $M_k(x, y)$, 如式(7)所示:

$$M_k(x, y) = d_k(x, y) \cap d_{k-1}(x, y) \quad (7)$$

5) 统计二值图像 $M_k(x, y)$ 中灰度值为 1 的像素点的个数 m 。若 m 小于设定阈值 T_3 , 则判定当前帧无运动目标, 跳过该帧执行下一帧检测; 若 m 大于设定阈值 T_3 , 则判定当前帧有运动目标, 开始录像, 并开始检测下一帧图像。

3 智能监控系统系统软件设计

3.1 系统软件设计总体思路

智能监控系统总体程序设计主要由初始化程序、运动目标检测程序、视频录制程序和邮件发送程序四部分组成。

初始化程序主要将需要用到的所有模块进行引用, 为后续程序的运行提供初始化和必要的库函数。运动目标检测程序用到的基于关联帧差分运动目标检测算法, 首先对图像进行灰度化和高斯滤波预处理, 然后计算相邻帧的相似度, 若相似度满足设定的阈值, 则将序列图像相邻 3 帧的图像进行差值处理, 将得到差值的绝对值进行与运算后输出到一张新的图片中, 将新的图像进行腐蚀与膨胀, 通过阈值函数处理如果超过阈值则输出有运动目标的信号。当检测到有运动目标时, 视频录制程序开始运行, 录制运动目标闯入监控视野后的 400 帧视频, 并将所录制的视频保存到指定文件夹中。邮件发送程序需要用到子程序中的 SMTP 协议配置, 采用单独的线程来运行。当视频录制完成后, 邮件发送程序将其发送到预先设定好的邮箱内。

3.2 系统软件配置

智能监控系统软件配置包括树莓派操作系统和编程环境两部分。操作系统选择树莓派官方支持的 Raspberry Pi OS 作为操作系统, 只需将系统镜像文件烧录至 TF 卡即可。

树莓派选用 Python 编程语言。Python 是一种面向对象的开源跨平台程序设计语言, 包含各种区域的代码, 如单元测试、线程、图像处理、操作系统接口和协议、Web 服务工具等。这种类型的代码可以消除智能监控系统编写冗长代码的需要, 同时能够把其它语言编写的各种模块, 尤其是 C/C++ 语言编写的模块, 很轻松地联结在一起, 语法清晰简单, 调用方便^[23]。

监控程序主要调用 OpenCV 库函数来实现。OpenCV 是 Intel 的一个开源免费视觉库, 其内部函数库是由 C/C++ 编写的, 具有量清、高效的特点^[24]。OpenCV 库函数内部提供的大量 API 接口可实现图像去噪、边缘检测、角点检测、色彩变换、形态学处理等功能, 同时支持图像文件数据以及摄像头数据的输入, 图像以及视频文件的输出功能。模块功能强大, 调用方便, 能够满足各类图像处理需求。

SMTP 是一种可靠且有效的电子邮件传输协议, 主要用于系统之间的邮件信息传递, 并能够提供有关来信的通知, 操作简单、方便。因此, 系统选用 SMTP 协议实现报

警邮件发送功能。在设计系统程序之前，需要设置用户的邮箱将 SMTP 协议开启，之后可获得邮箱授权码和邮箱服务器地址，这对于之后设计发送邮件报警程序至关重要。

3.3 系统软件实现流程图

智能监控系统软件实现流程如图 3 所示。初始化程序模块首先需要打开摄像头，然后设置帧数和读取当前背景，其次检测是否捕获视频流，如果未捕获视频流则说明未开启摄像头，则重新开启摄像头，直至开启成功。

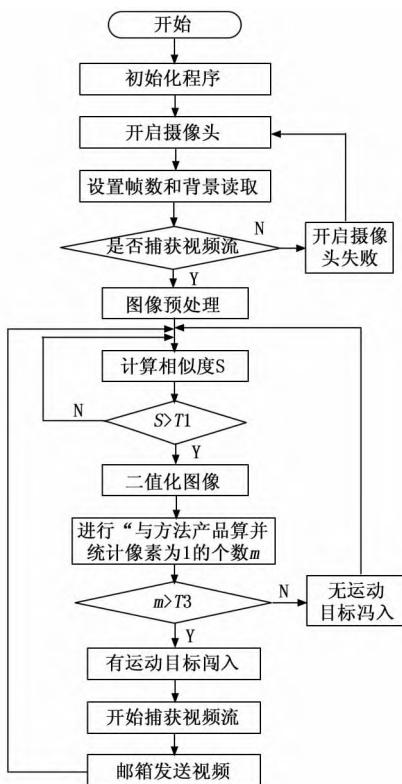


图 3 智能监控系统软件流程图

初始化程序配置完成后，运动目标检测程序首先对捕获的图像进行灰度化和高斯滤波处理。然后，计算相邻两帧图像的相似度，如果前一帧（第 $k-1$ 帧）图像与当前帧（第 k 帧）的相似度、当前帧（第 k 帧）与后一帧（第 $k+1$ 帧）图像的相似度均大于阈值 T_1 ，则对相邻两帧图像进行差分运算得到差分图像 $D_k(x, y)$ 与 $D_{k-1}(x, y)$ ，否则重新计算下一帧图像与相邻帧的相似度。其次考虑到光照、噪声等外部因素造成的图像像素值的微弱变化，将差分图像与利用最大类间方差（OTSU）得到自适应阈值 T_2 进行比较得到二值图像 $d_k(x, y)$ 和 $d_{k-1}(x, y)$ 。最后，对 $d_k(x, y)$ 和 $d_{k-1}(x, y)$ 取与运算，得到含有运动对象和两帧差分图像重叠部分的二值图像 $M_k(x, y)$ ，并统计二值图像 $M_k(x, y)$ 中灰度值为 1 的像素点的个数 m 。若 m 小于设定阈值 T_3 ，则判定当前帧无运动目标，跳过该帧执行下一帧检测；若 m 大于设定阈值 T_3 ，则判定当前帧有运动目标，则开始录像，并开始检测下一帧图像。

当检测到有物体闯入后，视频录制程序开始运行，并

在指定的文件夹保存 400 帧的视频。如果第一次运行没有找到保存的视频的文件夹，则会自动创建一个新的文件夹，后续录制的视频都会保存在这个文件夹，为邮件发送程序提供检索文件。视频录制并保存完整后，显示屏上会显示录像结束。

邮件发送程序需要用到邮箱的 SMTP 协议。在执行将录制的视频发送到指定邮箱之前，需提前开启指定邮箱的 SMTP 协议并获得邮箱服务器地址和密码。邮件发送程序为单独的线程，在程序运行中，不受其它模块的影响。

4 实验结果与分析

本研究设计的智能监控系统以 Raspberry Pi 4B 开发板为硬件平台并以 Python3 语言为基础结合 OpenCV 实现。树莓派出厂时是不具备操作系统的，因此在系统进行实验测试之前，需要给树莓派中烧录系统。在树莓派官网下载烧录系统的工具包，这款工具包可以轻松的为树莓派镜像烧录系统，只需要一步就可以自动的安装好，不需要更多操作。系统烧录完成后，需要对树莓派开发板进行运行环境的搭建，包括 OpenCV 和 SMTP 协议的安装等。系统需要在之后每次启动时都要打开摄像头，因此需要在软件搭建环境中将摄像头设置成开机自启动模式。

4.1 图像采集功能测试

摄像头功能完好与否是智能监控系统成功运行的基础。因此，在软件环境搭建完成并成功登陆系统后，需要对摄像头拍照范围和清晰度进行测试。运行摄像头拍照程序，若摄像头能够正常工作，则屏幕上会显示拍照结果，如图 4 所示，否则摄像头有问题，需进行维修或更换。



图 4 摄像头测试图

4.2 智能监控系统测试

智能监控系统识别率受环境的光线强弱、拍摄角度、拍摄背景等因素的影响，因此为了更好地测试系统的功能是否全面，需要进行正常亮度情况下、昏暗环境下以及不同目标进入监控范围时系统是否都能够正常工作。

4.2.1 正常光线下测试

正常光线下智能监控系统测试结果如图 5 所示，其中图 (a), (c), (e) 为 3 个目标闯入监控范围的测试结果；图 (b), (d), (f) 为发送到指定邮箱的结果显示。

实验结果表明：本研究设计的智能监控系统能够在正常光线下完成目标检测、保存目标闯入视频并将视频发送到指定邮箱报警功能。



图 5 正常光线下智能监控系统测试结果

4.2.2 昏暗光线下测试

昏暗光线下智能监控系统测试结果如图 6 所示, 其中图 (a), (c), (e) 为 3 个目标闯入监控范围的测试结果; 图 (b), (d), (f) 为发送到指定邮箱的结果显示。

实验结果表明: 本研究设计的智能监控系统能够在昏暗光线下完成目标检测、保存目标闯入视频并将视频发送到指定邮箱报警功能。



图 6 昏暗光线下智能监控系统测试结果

正常光线和昏暗光线下测试结果表明: 本研究设计的智能监控系统不仅能够顺利执行监控、报警等任务, 还可以在不同光线强度、不同目标闯入下正常工作, 稳定性高、实时性强, 满足智能监控系统设计需求。

5 结束语

以智能监控系统为实践项目, 设计并开发了一套具备自动报警功能的智能监控系统。该系统选用树莓派 4B 搭建硬件平台, 搭载 Linux 系统, 基于 Python 开发环境, 采用灰度加权平均法及高斯滤波算法对图像进行预处理, 基于关联帧差分运动目标检测算法, 结合图像采集模块, 实现运动目标检测; 当系统检测到有目标闯入时, 开始录制目标闯入后的 400 帧视频; 视频录制完成后发送到指定邮箱, 实现自动报警功能。在正常光线和昏暗光线下对系统功能进行测试, 实验结果表明在不同光线强度、不同目标闯入情况下, 系统能够准确识别目标闯入, 实现监控、报警功能, 实时性强、稳定性高, 具有广阔的应用前景。

参考文献:

- [1] 陈志文, 张国平, 林光源, 等. 基于 V4L2 的嵌入式 DVR 系统设计与实现 [J]. 电子设计工程, 2017 (11): 176–179.
- [2] 赵方鹏. 基于 ARM 和 Linux 的嵌入式网络视频监控系统研究 [D]. 济南: 山东大学, 2007.
- [3] COLLINS R, LIPTON A, KANADE T. A system for video surveillance and monitoring: VSAM final report [R]. Technical Report: CMU – RTR – 00 – 12, Robotics Institute, Carnegie Mellon University, Pittsburgh, Penn, USA, 2000.
- [4] 刘佳琳. 基于 ARM 的嵌入式智能监控系统设计 [D]. 西安: 西安石油大学, 2021.
- [5] 姚国任. 基于 Server 酱的电子客票生成信息推送设计与实现 [J]. 吉林化工学院学报, 2021, 38 (9): 53–59.
- [6] 王淑文. 推进以视频为核心的多维感知技术在能源行业中的深度应用 [J]. 中国安防, 2021 (4): 65–68.
- [7] 吴少宝, 陈庆辉, 黄美璇, 等. 基于树莓派的 Zabbix 监控告警平台设计与实现 [J]. 通化师范学院学报, 2022, 43 (6): 1–7.
- [8] 郭亮, 叶爱民, 林涛, 等. 基于树莓派和 Java 语言的温湿度远程实时测量系统的设计 [J]. 计算机测量与控制, 2017, 25 (6): 4–8.
- [9] 潘燕. 基于改进型决策树 SVM 的图像识别方法 [J]. 新乡学院学报, 2018, 35 (12): 17–20.
- [10] 赵岩, 赵建国. 一种移动车辆遮挡下的车道线检测方法 [J]. 科学技术与工程, 2021, 21 (7): 2782–2787.
- [11] 杜文汉, 李东兴, 王倩楠, 等. 融合改进帧差和边缘提取算法的运动目标检测 [J]. 科学技术与工程, 2022 (5): 1944–1947.
- [12] 吕慷, 张旭秀, 李卫东. 基于人眼识别原理的运动目标检测方法 [J]. 电子测量技术, 2019, 42 (4): 65–69.
- [13] 肖军. 基于光流法的运动目标检测与跟踪算法 [J]. 东北大学学报: 自然科学版, 2016, 37 (6): 770–774.

(下转第 134 页)

度下降，现有的补偿方法基本无法实现在通信负载不增加的前提下补偿通信的延时以及丢帧^[20]。对比测试数据表明，使用标准化 CANopen 通信协议开发的控制系统，在相同的网络节点数量、网络复杂程度、网络拓扑结构情况下，采用同样的硬件平台和通信速率，构建的控制网络可靠性非常高，网络负载率较低，明显优于“自定义协议”系统，尤其是关系到控制系统性能指标的“全系统数据刷新周期”、故障自愈能力、故障自诊断能力，以及网络管理能力、组网难度、新节点加入难易程度、系统开发难度、网络调试难度等方面全面超过“自定义协议”系统。

5 结束语

某高速水下航行器采用了基于 CANopen 网络架构设计出航行器控制系统，进行了大量组部件级测试工作，开展了岸上工房系统级调试，完成了实航试验。经过多航次、多目标、对抗性实航试验验证，航行器控制系统功能正常，满足系统设计指标要求。

经过外部标准仪器设备实际测试，航行器控制系统 CANopen 网络采用 500 kbps 波特率，全系统（包括控制系统、动力系统、姿态系统、智能执行机构伺服系统、地面调试系统、CAN 网关电路、发控接口电路、载荷系统、模拟系统等）运行快速，信息交换流畅，传输滞后小，整个控制网络负载率为 21.9%，明显优于国际 CANopen 现场总线网络协会要求。

即使在干扰信号强烈的高速实航试验过程中，CANopen 控制网络工作正常，能稳定可靠地高效工作，姿态系统响应快速，网络数据刷新率满足控制要求，证明航行器 CANopen 控制网络满足设计要求，该技术适用于通用 UUV、ROV 等多种水下航行器系统，值得推广应用。

参考文献：

- [1] 董权威, 王奥博, 岳才谦, 等. 基于 CAN 总线的自主水下航行器分布式控制系统设计 [J]. 计算机测量与控制, 2020, 28 (12): 104–107.
- [2] 卢绍庆. 现场总线在水下航行体控制系统中的应用 [J]. 水中兵器, 2013 (s): 120–125.
(上接第 127 页)
- [14] 文 丰, 王乐群, 张凯华. 基于 ZYNQ 加速的帧差法运动目标检测 [J]. 单片机与嵌入式系统应用, 2022, 6: 74–78.
- [15] 罗 敏, 刘洞波, 文浩轩, 等. 基于背景差分法和帧间差分法的车辆运动目标检测 [J]. 湖南工程学院学报: 自然科学版, 2019, 29 (4): 58–61.
- [16] 杜建宝, 张祖锋. 基于 FPGA 的运动目标检测系统的设计 [J]. 仪器仪表用户, 2018, 25 (3): 40–42.
- [17] 山 丹, 丛国涛. 基于 FPGA 的动态目标识别与跟踪系统设计 [J]. 电子测量技术, 2019, 42 (10): 132–136.
- [18] 李艳荻, 徐熙平, 陈 江, 等. 动态特征块匹配的背景更新在运动检测的应用 [J]. 仪器仪表学报, 2017, 38 (2): 445–453.
- [19] 贺振东, 王耀南, 刘 洁, 等. 基于背景差分的高铁钢轨表面
- [3] ZELTWANGER H. 现场总线 CANopen 设计与应用 [M]. 周立功, 等译. 北京: 北京航空航天大学出版社, 2011.
- [4] 饶运涛, 邹继军, 郑勇芸. 现场总线 CAN 原理与应用技术 [M]. 北京: 北京航空航天大学出版社, 2003.
- [5] 许东辉. 基于 CANopen EtherNet – IP 的协议转换模块研究 [D]. 天津: 天津理工大学, 2012.
- [6] 刘 杰, 蒋幽君, 等. CAN 总线在石油仪器组合导航系统中的应用 [J]. 计算机测量与控制, 2016, 24 (4): 174–177.
- [7] 王立翔. 基于 CAN 总线智能节点的设计及可靠性分析 [D]. 西安: 西安电子科技大学, 2013.
- [8] 胡 希. 基于 CAN 总线的车载 GPS 上位机综合管理系统的开发 [D]. 辽宁: 东北大学, 2015.
- [9] 袁文燕. 基于 CANLIN 网络结构的汽车网络设计 [D]. 北京: 中国农业大学, 2012.
- [10] 张 义. 基于 DeviceNet 总线协议的从站接口设计 [D]. 北京: 北京工业大学, 2008.
- [11] 许东辉. 基于 CANopen EtherNet – IP 的协议转换模块研究 [D]. 天津: 天津理工大学, 2012.
- [12] 田绍宇. CANopen 网络的实现及实时性研究 [D]. 武汉: 华中科技大学, 2014.
- [13] 刘学伟. DeviceNet – CANopen 集成化从节点通信模块开发设计 [D]. 天津: 天津理工大学, 2013.
- [14] 袁 远, 李红信. CAN 网络实时性分析及改进 [J]. 电子设计工程, 2010 (6): 110–112.
- [15] 刘 怡, 王家乐. CANopen 从站协议的设计与实现 [J]. 电子技术, 2011 (9): 38–40.
- [16] 韩进军, 唐红卫. CAN 总线实时性研究及改进 [J]. 计算机测量与控制, 2005 (11): 1247–1249.
- [17] 禹春来, 许化龙, 刘根旺, 等. CAN 总线冗余方法研究 [J]. 测控技术, 2003 (10): 28–30.
- [18] 苏奎峰, 吕 强. TMS320F2812 原理与开发 [M]. 北京: 电子工业出版社, 2005.
- [19] 苏国火. 嵌入式 CANopen 主站与 IO 从站的功能研发 [D]. 广州: 广东工业大学, 2014.
- [20] 郑宇, 张 健, 施卫科, 等. 实时控制系统中 CAN 总线丢帧及延时的补偿研究 [J]. 计算机测量与控制, 2020, 28 (7): 107–111.
缺陷图像分割 [J]. 仪器仪表学报, 2016, 37 (3): 640–649.
- [20] 屈晶晶, 辛云宏. 连续帧间差分与背景差分相融合的运动目标检测方法 [J]. 光子学报, 2014, 43 (7): 219–226.
- [21] 左风艳, 高胜法, 韩建宇. 基于加权累积差分的运动目标检测与跟踪 [J]. 计算机工程, 2009, 35 (22): 159–161.
- [22] 李搏轩, 沈永良, 胡 月. 混合高斯模型与三帧差分法相结合的建模新算法 [J]. 黑龙江大学工程学报, 2016, 7 (1): 54–59.
- [23] 俞文静, 莫健彬, 黄嘉锵. 基于树莓派与 OpenCV 的智能监控跟踪机器人系统设计与实现 [J]. 现代计算机, 2018, 6 (1): 80–84.
- [24] 高 向. 基于 OpenCV 运动目标检测与跟踪算法研究 [D]. 银川: 宁夏大学, 2019.