

海计算模式下森林火焰识别算法的实现

李 悦, 赵涓涓, 强 彦

(太原理工大学 计算机科学与技术学院, 山西 太原 030024)

摘 要: 针对传统森林防火方法中耗能低效的问题, 讨论了海计算模式下森林防火的意义, 并提出了海计算模式下森林火焰识别算法的实现. 首先根据感知端的特点使用 Qt/Embedded 设计, 再依据森林火焰的特点使用 24 比特加色模型和 Sobel 算子实现色彩和区域判定, 最后把结果反馈到汇聚节点. 该算法能自动对视频捕获的图像进行分析计算并能自动反馈结果, 在传感端就完成了火情判别. 经实验验证, 相比传统的森林防火方法, 该算法不仅实现了森林火灾的即时检测和识别, 同时也实现了高准确率和低耗能.

关键词: 海计算; Qt/Embedded; 24 比特加色模型; Sobel 算子

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2014)02-0067-05

Implementation of Forest Fire Recognition Algorithm Based on Sea Computing

LI Yue, ZHAO Juan-juan, QIANG Yan

(College of Computer Science and Technology, Taiyuan University of Technology, Taiyuan 030024, China)

Abstract: To the problem of low efficiency and high energy consumption for the traditional forest fire prevention methods, this article discussed the significance of forest fire prevention based on Sea computing, and put forward the implementation of forest fire recognition algorithm based on Sea computing. Firstly, used Qt/Embedded to design according to the characteristics of the sensor node. Then, used 24-bits additive color model and Sobel operator to complete the decision of color and region based on the features of forest fire. Finally, the results were fed back to the sink node. This algorithm could automatically analysis image captured by camera and complete the decision of fire in sensor node. The experiments showed that, compared to the traditional methods of forest fire prevention, this algorithm not only completed the instant detection and recognition of forest fire, but also achieved high accuracy rate and low time-consuming at the same time.

Key words: Sea computing; Qt/Embedded; 24-bit additive color model; Sobel operator

1 引言

当前, 物联网研究在感知端(物端)的计算与控制比较薄弱, 很多操作需要通过网络传输在后台完成, 既耗能又低效. 由此, “海计算”概念被引入到了物联网的发展中. 海计算模式^[1]是指通过在物理世界的物体中融入计算与通信设备以及智能算法, 让物体与物体之间能够互连, 在事先无法预知的场景

中进行判断, 使物体具备自组织、自计算、自反馈的计算功能. 海计算的本质是物体与物体之间的智能交流, 实现的是物之间的交互, 强调物理世界的智能连接和物理性质涌现.

在依托物联网, 亦即通过各类无线传感器网络实现森林防火的众多方法中^[2-3], 基本是通过以下两种方法来实现的: (1) 节点处搭载温湿度等传感器, 通过无线网络将温度湿度等数据传回汇聚节点, 然

收稿日期: 2013-03-29; **修回日期:** 2013-05-20

基金项目: 国家自然科学基金(61202163, 61240035, 61373100); 山西省科技攻关项目(20120313032-3); 山西省自然科学基金(2012011015-1)

后汇聚节点根据传回的温度湿度等数据来进行判断区域范围内是否有火情发生;(2)通过节点处的摄像头将采集的图像传回汇聚节点进行图像分析,然后判定是否有火情发生.这些方法都基于传统物联网,即计算与控制都是通过网络传输在汇聚节点完成的.

文献[3]中对不同情况的海计算模式化进行了研究和实现.其中文献[4]将与传统的车辆安全协作系统与海计算模式相结合,对传输控制单元进行调整使其具备智能化,根据无线车载网络的信道情况和状态信息,调整发送的频率和消息长度,根据自身位置估计模型和实际位置决定是否发送自身位置信息,提高了系统的鲁棒性和扩展性.文献[4]将海计算模式与传统的无线传感器网络路由协议相结合,根据情景感知计算提出了一种新的路由协议 CAS-CR(Context-Awareness in Sea Computing Routing Protocol),将物联网网络层中的路由协议运用到传感层上.本文提出了海计算模式下森林火焰识别算法.

2 海计算模式下的森林火焰识别算法思想

海计算模式下的森林火焰识别算法是基于物联网的感知层,是物理世界感知中的智能图像识别及处理算法,如图1所示.该算法的核心是将物联网应用层中的计算在传感层实现,即海计算模式化.它搭载在传感器节点内的微处理器中,当传感器通过外置摄像头截获图像时,火焰识别算法能够自动获取所截获的图像,然后根据图像特征来进行分析,判断这个传感器节点所覆盖的区域是否有火情出现.当一旦判断火情出现时,该算法立即通过串口将代表着“有火情出现”这个内容的信号通过无线网络传送到汇聚节点以及管理节点,来通知相关人员采取措施.由于该算法是基于无线传感器网络实现的,因此算法必须能在传感器上运行,且需要根据传感器中微处理器的执行能力来进行设计.我们选取的平台是常见的嵌入式 Linux 操作系统.

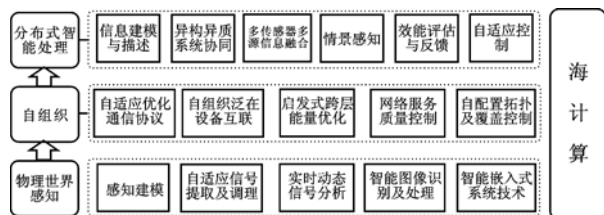


图1 海计算技术结构图^[1]

传统方法之所以无法在传感器端(物端)实现图

像识别,一大问题在于传统的森林火焰识别算法大多复杂,节点处微处理器应付起来相当吃力,因此时间复杂度往往在 $O(n^4)$ 以上^[5-6],耗时又耗电,并不适合森林防火.而本文提出的海计算模式下的森林火焰识别算法,在保证准确率的基础上,根据微处理器的特点将算法体积进行了适当的缩小,时间复杂度控制在了 $O(n^2)$ 以下.该算法的主要思想分为两部分:(1)能够得到摄像头所截获图像的特征;(2)能够针对这些特征来进行分析.我们所选取的算法实现工具是 Qt/Embedded. Qt/Embedded 是图形化界面开发工具 Qt 的嵌入式版本,它通过 QtAPI 与 Linux I/O 以及 Framebuffer 直接交互,拥有较高的运行效率,而且整体采用面向对象编程,拥有良好地体系架构和编程模式.

3 森林火灾的火焰图像特征提取

森林火灾与普通火灾的检测是不同的.因为如果使用机器来断定森林火灾的图像,那会有许多干扰因素,包括树叶、阳光等等.因此,在建立算法时,需要考虑不同因素的干扰.在火灾的初期增长和充分发展阶段,火灾火焰一般具有较为明显的图像视觉特性,如阴燃亮光、火焰颜色、闪烁和外形变化等^[5-7].在我们提出的火焰识别算法中,需要用到图像信息的火焰物理特征有:

(1)颜色变化.由于火灾火焰的颜色与温度具有相关关系,因此在图像处理中,可以根据图像中各点像素的颜色来判别是否出现火焰.如图2所示,根据不同区域的颜色变化进行火焰判定.

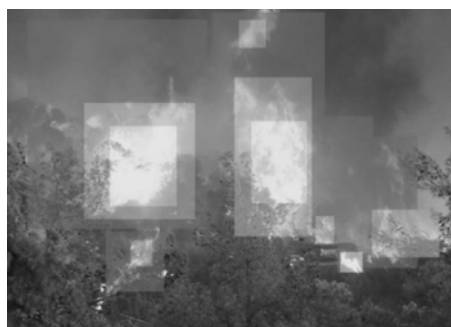


图2 森林火焰的颜色区分

(2)面积变化.着火引燃后,火灾呈不断发展的趋势.从视频图像上可观察到火灾火焰的面积呈现连续的、扩展性的增加趋势.但是,由于森林火灾的特殊性,可能会有当其他类火焰物体(像是红色枫叶)从视野外移入时,此时也会探测到的疑似火焰目标,容易造成干扰.因此,我们依据森林火灾中火焰会迅速蔓延的特点,通过疑似火焰区域的面积变化

来判断是火灾发生还是有干扰物体进入.如图3和图4所示,森林火焰具有迅速蔓延的特点,而森林中疑似火焰的干扰物其区域面积是稳定不变的.

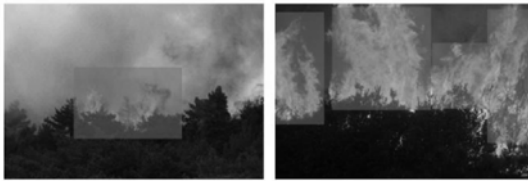


图3 森林火焰区域的急速扩大性

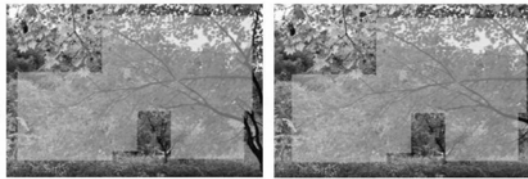


图4 森林干扰区域的稳定不变性

4 算法总体设计

4.1 加色模型

在实际应用中,最常用的彩色判定模型是加色模型,例如通过对红(R)、绿(G)、蓝(B)三个颜色通道的变化以及它们相互之间的叠加来得到各式各样的颜色.本文中所采用的是24比特加色模型.

24比特加色模型是指,在计算机的显示模式中,色彩模型是通过24比特来实现的.例如每像素24位(bits per pixel, bpp)编码的RGB值:使用三个8位无符号整数(0到255)表示红色、绿色和蓝色的强度.

提取摄像头截获图像的RGB值是通过QRgb这个类实现的.

$QRgb\ rgb = image.pixel(i, j),$

式中, image 是通过 Qimage 类读入的图像变量.将图像中坐标为 (i, j) 的像素点的 R 值、B 值和 G 值存到 rgb 三维矩阵中.

$r = qRed(rgb);$

$g = qGreen(rgb);$

$b = qBlue(rgb).$

然后将每个像素点 R 值、G 值、B 值分别赋到 r 、 g 、 b 三个变量中.根据经验数据设定 RGB 阈值 $V(R, G, B)$, 如果某个像素点的 R, G, B 值同时达到阈值范围,就把这个点定义为火焰点(fire point).

4.2 Sobel 算子

传统火焰识别算法中,火焰区域判定一般是通过协方差矩阵等方法提取轮廓来实现的.本文基于的平台是嵌入式 Arm9 处理器与嵌入式 Linux 操作

系统,因此将对传统的面积算法进行简化.

根据火焰图像的特点,我们采用 Sobel 算子方法.

该算子包含两组 3×3 的矩阵,分别为横向及纵向,将之与图像作平面卷积,即可分别得出横向及纵向的亮度差分近似值.如果以 A 代表原始图像, G_x 及 G_y 分别代表经横向及纵向边缘检测的图像,其公式如下:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \quad (1)$$

$$G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A \quad (2)$$

图像中每一个像素的横向及纵向梯度值可以用下面的公式来计算:

$$G = \sqrt{G_x^2 + G_y^2} \quad (3)$$

但通常情况下,一般采用下面的公式来求梯度近似值:

$$|G| = |G_x| + |G_y| \quad (4)$$

然后可用以下公式计算梯度方向.

$$\Theta = \arctan\left(\frac{G_y}{G_x}\right) \quad (5)$$

本文的算法是基于数字图像,采用的方法是

$$T(i, j) = \sqrt{f_x^2 + f_y^2} \quad (6)$$

式(6)中的 f_x 和 f_y 分别是:

$$f_x = f(i-1, j-1) + 2f(i-1, j) + f(i-1, j+1) - f(i+1, j-1) - 2f(i+1, j) - f(i+1, j+1) \quad (7)$$

$$f_y = f(i-1, j-1) + 2f(i, j-1) + f(i+1, j-1) - f(i-1, j+1) - 2f(i, j+1) - f(i+1, j+1) \quad (8)$$

式(7)、(8)是遍历图像中每个像素点,并计算各个像素点(例如 (i, j)) 的上下左右邻点的灰度加权值和.然后根据经验数据设定阈值 τ , 若 $T(i, j) \geq \tau$, 则该像素点 (i, j) 便为边缘点.如果边缘点能构成闭区,则说明有疑似火焰区域.设 $area$ 作为火焰区域值,此时令 $area=1$.

4.3 基于经验数据的阈值确定

选取 50 幅不同火焰图像作为训练样本,如表 1 所示.

表 1 是作为训练样本的 50 幅图像, (R, G, B) 值与灰度加权值和作为特征数据,表中数据表明按照此 (R, G, B) 值和灰度加权值判定火焰情况的准确

程度. 然后依据此数据建立机器学习决策树方法, 使用表 1 中的数据作为训练样本完成智能识别得出决策树模型, 如图 5 所示.

表 1 作为训练样本的 50 幅图像的特征数据

图像顺序	(R,G,B)值	灰度加权值和	类标
1	(180,90,18)	225	1
2	(202,134,55)	258	1
3	(187,55,35)	213	1
4	(228,89,46)	348	1
5	(177,112,8)	250	1
6	(218,68,11)	289	1
...	...	...	...
...	...	...	...
45	(225,47,1)	361	0
46	(198,38,15)	213	0
47	(214,99,23)	344	0
48	(236,108,2)	348	0
49	(176,37,23)	274	0
50	(164,90,11)	267	0

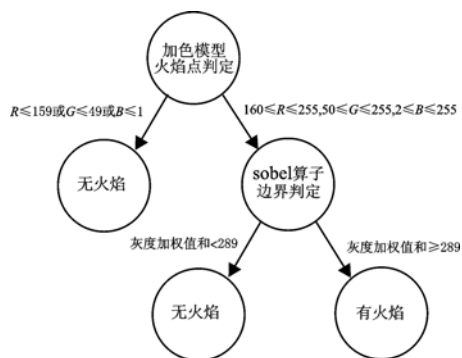


图 5 根据机器学习得出的决策树模型

从图 5 可以看出, 根据决策树模型中可以得到 RGB 阈值以及灰度加权阈值. 其中加色模型火焰点判定的参数阈值为

$$V(R, G, B) = \begin{cases} 160 \leq R \leq 255 \\ 50 \leq G \leq 255 \\ 2 \leq B \leq 255 \end{cases} \quad (9)$$

Sobel 算子边界判定的灰度加权阈值为

$$T(i, j) \geq \tau, \tau = 289 \quad (10)$$

4.4 火焰趋势判定

在之前对 RGB 值进行判定时, 大于阈值的点我们会设为火焰点 (fire point). 这里, 我们定义变量 count 作为火焰点的数量. 算法中, 计算火焰点 count 在整幅图像中的百分比 percent. 设定的火灾怀疑值为 15%, 如果 percent 小于 15%, 或没有完整成型的火焰即 area=0, 认为没有火情发生, 此时向汇聚节点发送数据包“00000000”, 然后执行 shut

down 命令关闭传感器.

如果 percent 大于 15%, 且 area=1, 则再次启动摄像头截获图像继续作分析, 如果百分比没有变化, 则认定为疑似火焰物体的干扰, 也认为没有火情发生. 如果 percent 有变化, 大于 15% 且小于 20%, 则认为是稳定火焰, 向汇聚节点发送数据包“00110011”, 汇聚节点依据这个判定为稳定火焰; 如果百分比大于 20%, 则可以认定是失控火焰, 向汇聚节点连续发送信号“11111111”, 汇聚节点依据这个判定为失控火焰.

5 实验

5.1 实验平台设计

本文实验所采用的平台是单个传感器节点和单台计算机模拟的汇聚节点, 属于简化版的无线传感器网络. 传感器节点由以下几部分组成: 微处理器, Zigbee 无线通信模块, 电源, 摄像头. 具体结构如图 6 所示.

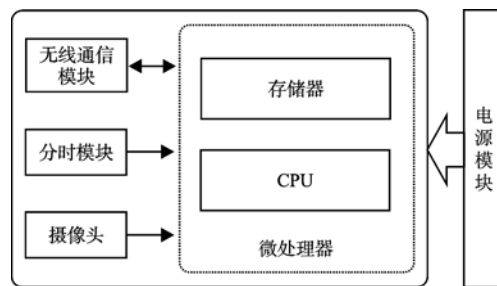


图 6 传感器节点硬件结构图

5.2 实验步骤

(1) 传感器启动时, 摄像头自动获取周遭图片, 通过 ZigBee 无线与汇聚节点通信, 记录结果.

(2) 当传感器自动关闭时, 在一小块区域点燃火苗. 然后待传感器再次启动时, 摄像头获取火灾时的图片, 与汇聚节点通信, 记录结果.

(3) 为了获取与温湿度传感器方式的对比结果, 将有温湿度传感器但无森林火焰识别算法的传感器节点放入同样的环境中, 重复步骤(1)和(2).

(4) 为了获取与传感器节点传送图像到汇聚节点进行处理的方式的对比结果, 将无森林火焰识别算法的传感器节点放入同样的环境中, 重复步骤(1)和(2), 只是当摄像头截获图像后不处理而是通过无线网络传给汇聚节点, 在汇聚节点用相同的森林火焰识别算法进行判别并记录结果.

6 结果讨论

通过上述实验. 我们得到如下结果:

