

引用格式: QIU Liya, CHEN Weilin, LI Fanming, et al. Fast Hash_LBP Moving Target Detection Algorithm Based on Hamming Distance Constraint in Complex Background[J]. Acta Photonica Sinica, 2022, 51(9):0910003

裘莉娅,陈玮琳,李范鸣,等. 复杂背景下基于汉明距离约束的 Hash_LBP 运动目标快速检测算法[J]. 光子学报, 2022, 51(9): 0910003

复杂背景下基于汉明距离约束的 Hash_LBP 运动目标快速检测算法

裘莉娅^{1,2,3}, 陈玮琳^{1,2,3}, 李范鸣^{1,3}, 刘士建^{1,3}, 王晓钰^{1,2,3}, 李严寒^{1,2,3}

(1 中国科学院上海技术物理研究所, 上海 200083)

(2 中国科学院大学, 北京 100049)

(3 中国科学院红外探测与成像技术重点实验室, 上海 200083)

摘要: 自然环境复杂多变, 存在复杂天气如雨雪雾, 草木摇晃和水面波动等大量动态背景并且光线不断变化, 因此对噪声以及背景的抑制一直是复杂场景中运动目标检测的首要难题。为了抑制动态背景、慢速目标被吸收以及图像编码噪声等问题, 在保证实时性的基础上, 提出了一种基于纹理特征的自适应阈值运动目标检测算法。所提算法将感知哈希算法与局部二值模式结合, 提出了一种改进的 Hash_LBP 算法并使用汉明距离进行约束, 得到输入图像的局部二值模式值进行频次统计后, 完成背景建模和前景提取。实验结果表明, 所提算法对于红外和可见光等多种复杂背景, 能够有效地抑制噪声、光照变化和动态背景, 快速准确提取前景目标。

关键词: 计算机视觉; 运动目标检测; 背景建模; 哈希算法; 局部二值模式

中图分类号: TP391.9

文献标识码: A

doi: 10.3788/gzxb20225109.0910003

0 引言

用计算机视觉^[1-3]算法实现运动目标检测一直是计算机技术领域中的一个主要研究方向。如今, 随着经济的发展, 在这个信息化的时代, 运动目标检测在视频监控、无人驾驶、行为研究等诸多领域有着广泛的应用^[4]。但在动态背景下的多运动目标检测仍然是一个具有挑战性的研究领域, 复杂的动态背景^[5]包含多种情况: 水的随机运动, 如雨滴、涟漪或水流等; 背景场景中的高速运动, 如挥舞树叶、移动旗帜或横幅等; 具有相似外观的周围背景的模糊物体; 照明和光照强度的不断变化, 比如, 拨动灯开关引起的突然或逐渐变化, 以及太阳被云覆盖等引起的路面场景的光照变化等。近年涌现出的深度学习^[6-8]算法, 因为具有较强的拟合能力和更高的准确度, 在以上场景中能够提取出准确的运动目标。但深度学习算法性能的好坏极度依赖于数据集的大小, 在缺少样本或者零样本的未知场景中, 其目标检测的准确率往往因样本学习不足而下降, 难以满足实际应用的要求。同时深度学习算法对硬件要求明显高于传统算法, 成本较高, 难以随时部署, 因此传统方法仍然有着广泛的应用范围。

背景建模法^[9-10]因具有算法简单、检测精度高, 运算速度快、满足实时性等优点, 逐渐成为广泛应用的目标检测方法, 目前的算法已经能够很好地实现静态背景下的运动目标检测^[11]。WREN C R 等提出了单高斯背景建模^[12], 由于研究者将每个像素的可能取值看作是一个高斯分布, 不能将背景分离, 因此对光照强度的改变敏感, 具有很高的误报率。STAUFFER C 等引入混合高斯背景模型 (Gaussian Mixed Model,

基金项目: 国家十三五预研基金 (No.104040402)

第一作者: 裘莉娅 (1994-), 女, 博士研究生, 主要研究方向为复杂背景下运动目标检测识别技术。Email: quliyad@163.com

导师 (通讯作者): 刘士建 (1975-), 男, 研究员, 博士, 主要研究方向为红外光电探测系统的总体设计和数字图像处理。Email: shj_liu@ustc.edu

收稿日期: 2022-01-08; 录用日期: 2022-02-24

<http://www.photon.ac.cn>

GMM)^[13],解决了单高斯背景建模不能处理背景中有运动目标的问题,但需要大量的训练帧且模型中的参数需要实时更新使得计算更为复杂,同样也对光照变化极为敏感。ELGAMMAL A等提出核密度估计法(Kernel Density Estimation, KDE),通过计算每个像素点成为背景的概率^[14],但KDE算法同样需要人为设置参数,且参数也需要大量的实验才能得到理想的数值,所以计算复杂度仍较大。LEE D等提出了基于递归高斯混合模型^[15],该方法通过对每一帧进行高斯计算得出一个自适应学习速率来代替全局静态的学习速率保持因子来实现。同年, KYUNGNAM K提出了一种实时的前景-背景分割算法(CodeBook)^[16],该算法将图像中每个像素上采样得到的背景值量化为码本,并在背景更新中对每个码字进行匹配,这使得能够在无参的情况下,捕捉到由于长时间的周期性运动而引起的结构性背景变化,但是前景目标若出现在建立模型期间,会在码本再次更新前出现误检,并且对剧烈的光照环境变化敏感;JUNG C R^[17]使用统计描述符构建了背景模型,基于改进的均值的模型来使噪声最小化并提高像素分类的准确性。这种方法可以检测阴影像素并很好地处理光照变化问题,但会因为背景场景的变化而突然变化。BARNICH O等提出基于ViBe(Visual Background Extractor)^[18]模型的背景建模方法,该方法并没有采用一般模型中先入先出的策略,而是采取随机策略来进行背景更新,计算量小速度快,对相机抖动有一定的鲁棒性,但对更新速率有很高的要求,更新太快会将前景目标吸收,更新太慢,则容易产生鬼影,无法快速地适应环境变化,尤其是在自然环境中受光照变化影响大。OY V提出了局部二进制模式马尔科夫随机场(LBP_MRF, Local Binary Pattern_Markov Random Field)^[19],一种增强的开源TBMOD版本,采用新的LBP^[20]算子和MRF结合,新的LBP使用了比原始LBP更大的邻域,以较低的计算成本有效地估计了前景。对检测光照波动的情况下重建模型有很好的效果。

背景建模的方法^[21-24]通常会有以下问题:环境光照的变化、背景中存在的运动将影响前景目标检测的结果、运动物体的阴影以及图像的噪声等。针对这些问题,提出了一种基于纹理特征的自适应阈值运动目标检测算法,该算法对图像序列中每一个像素点出现的不同灰度值的次数进行统计,做灰度值的直方图,将直方图中出现频率最高的灰度值作为当前背景的灰度值,使得慢速或静止的前景出现时,背景灰度不受影响,并且可以有效地区分静态背景和动态背景;然后结合Hash和LBP,既保留图像的边缘信息,又保留了图像的低频分量;再采用自适应阈值,使得在与参考背景进行比对稳定图像时更为准确,对不同场景具有适应性,减少噪声对LBP背景建模影响的同时有效地提高复杂场景下的召回率和准确率。

1 检测算法原理

算法总体流程如图1,首先将输入的图像序列转换为灰度图,以 3×3 为窗口计算均值,去除高频的细节信息,对图像进行平滑处理;然后基于多态频率分布的原理建立背景模型;接着计算背景模型与每帧图像的差值矩阵,根据差值矩阵计算自适应阈值,来稳定输入图像并且抑制图像序列中的动态背景和噪声等影响;稳定后的图像和背景模型经过LBP算法得到纹理特征,并计算两者之间的汉明距离,通过汉明距离进一步解决动态背景和噪声问题,并且图像纹理特征对光照变化不敏感,可以抑制光照变化的问题;最后再次基于多态频率分布的原理对LBP直方图进行背景建模,检测出运动目标,并且对结果进行形态学操作和孔洞填充提高目标提取的准确性。主要从参考背景建模、与感知哈希算法结合的LBP算法、自适应阈值选取、汉明距离对LBP纹理特征约束四个方面介绍了在动态背景抑制方面的改进。

1.1 基于多态频率分布的背景建模

文献[25]中, LAI A N等将固定长度的数据集序列帧中出现频率最高的灰度值作为真实背景值,该方法在图片序列帧较短的数据集中,容易造成将没有明显位移的运动目标区域或静止的目标区域定义为背景区域;在图片序列帧过长的数据集中,又会造成内存的过度消耗。针对以上问题,记录图像序列中每一个像素点 (x, y) 出现不同灰度值的次数,并进行统计做灰度值的直方图。随着图像序列在时间上的推移,读取的图像帧数 f 增加,即可得到该像素点不断更新的灰度值数组

$$F_{(x,y),f} = \{f_1, f_2, f_3, \dots, f_n\} \quad (1)$$

式中, f_n 为 (x, y) 第 n 帧像素点的灰度值。

图2为参考背景建模过程示意,图2(a)为输入图像序列,随机选取图中一点(100, 150)作为展示,随着

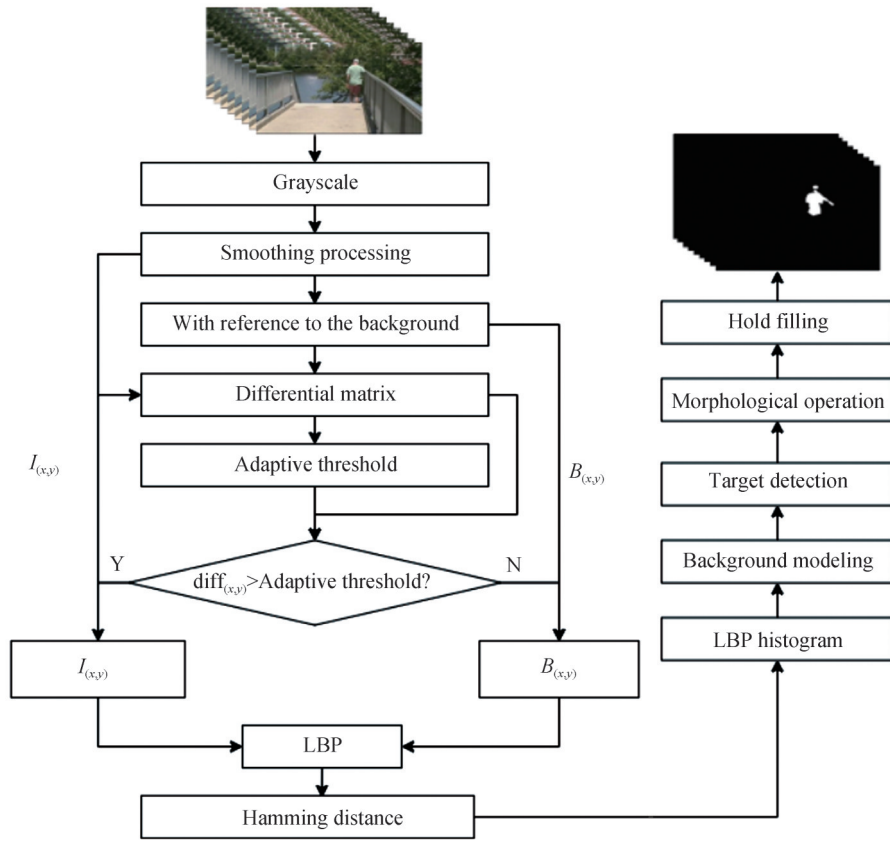


图1 算法流程
Fig. 1 Algorithm flow

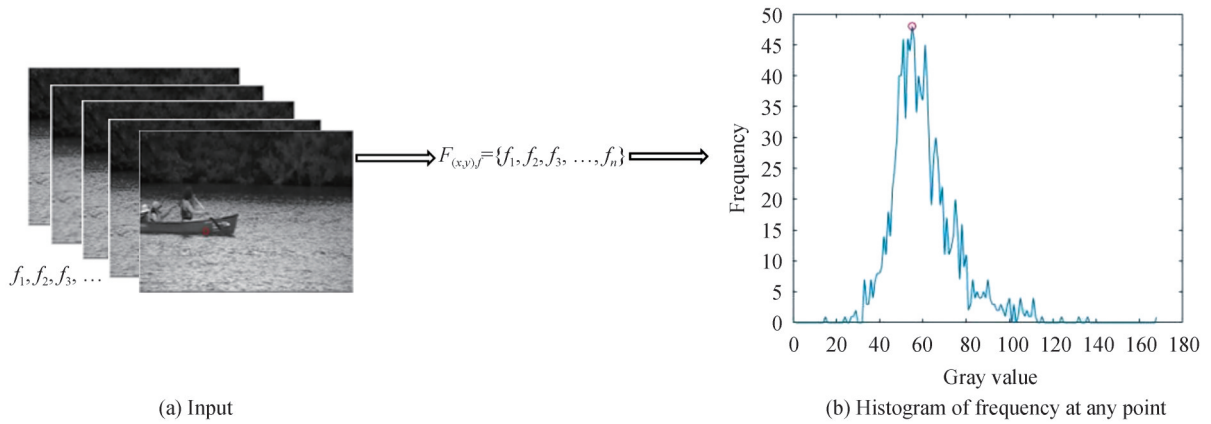


图2 参考背景建模过程
Fig. 2 Refer to the background modeling process

帧数的递增,将每一帧该点的像素值进行频次统计得到图2(b)所示的动态灰度频次直方图。图中红点标记的(55, 45)为第1 180帧时该点像素值为55,出现频次最多为45次,55作为当前参考背景中该点的像素值。

因此,灰度值为 a 所出现的次数 k 可表示为

$$k(a) = \sum_{f(x,y)=a} 1 \quad k \in [0, n] \quad (2)$$

将直方图中出现频率最高的灰度值作为当前背景的灰度值 $B_{(x,y),f}$, 此时表达式为

$$B_{(x,y),f} = \text{Max}[k(a)] = \{f(x,y) = a | \forall b \in [0, 255] \quad k(b) \leq k(a)\} \quad (3)$$

式中, $k(b)$ 为灰度值 b 出现的频次, 假设在当前帧 f_n 及之前帧出现频次最高的灰度值为 a , 频次为 $k(a)$, 下一帧 f_{n+1} 输入的灰度值为 b , 并统计 b 出现的次数为 $k(b)$, 则寻找当前帧该像素点的背景灰度值更新的过程为

$$B_{(x,y),n+1} = \begin{cases} B_{(x,y),n} = a & k(b) \leq k(a) \\ f(x,y)_{n+1} = b & k(b) > k(a) \end{cases} \quad a, b \in [0, 255] \quad (4)$$

1.2 汉明距离约束的 hash_LBP

1.2.1 哈希算法

哈希算法(hash algorithm)^[26]主要是利用图像的低频信息来得到每一张图像的一个字符串, 然后通过比较各自图像的字符串, 求取汉明距离来得到图像之间的相似度。并且哈希值不会随图像的放大、缩小或者改变纵横比而改变; 增加图像的对比度或者改变颜色, 对哈希值都不会造成太大的影响。

每一帧图像包含了低频和高频等各种频率, 其中高频成分主要为物体的边缘等亮度变化剧烈的区域, 而低频成分主要存在于亮度变化较小的区域, 均值哈希算法则主要是利用图像中的这些低频信息, 具体流程如下:

1) 首先将图像进行缩小, 去除高频的细节信息, 得到 8×8 的像素级大小, 这样就可以避免不同尺寸图像带来的差异, 并提高计算速率;

2) 然后将上一步采样后的图像转化为灰度图像;

3) 计算 8×8 个像素的灰度平均值;

4) 将每一个像素的灰度值与上一步求取的平均灰度值比较, 进行二进制编码, 如果小于平均灰度值, 则记为 0, 反之就将该像素点记为 1;

5) 将上面步骤后得到的结果, 按照一定的顺序(每一帧图像采用同样的次序)组合在一起得到一个 64 位的字符串, 该字符串记为当前图像的哈希值。

如果要比较两帧图像的相似度, 就可以根据这两帧图像的哈希值求取它们的汉明距离^[27], 汉明距离的数值越接近于 0 则相似度越高。

1.2.2 结合汉明距离后的 LBP 算法

感知哈希算法和 LBP 算法都可以很好地解决光照变化的问题, 并且运用感知哈希中的汉明距离计算背景模型和输入图像序列的相似度可以通过设置阈值的方式, 很好地抑制动态背景和噪声等对目标检测结果的影响。但由于存在较小的目标, 而哈希感知算法去掉了图像的高频分量, 将图像压缩到 8×8 的大小, 会导致较小尺寸的目标丢失。从均值哈希值的编码方式中, 可以让人联系到经典的 LBP 特征算法, 为了解决这个问题将哈希算法与 LBP 算法进行结合, LBP 可以保留明暗这种过渡的边缘信息, 而 Hash 保留下来的是整幅图像在采样后的低频分量。因此, 为了既保留图像中目标的特征信息, 又保留图像中关键的低频信息, 对传统的 LBP 进行了改进。最原始 LBP 算子采取的是相邻的 8 个像素点与中心像素点的灰度值作差分, 用中心像素的值作为阈值, 如果某相邻像素大于中心像素的值, 则该像素的值标记为 1, 反之则标记为 0, 得到 8 位二进制编码转换为十进制得到 LBP 值。但是易受噪声影响十分的不稳定, 导致最终的计算结果对噪声极其敏感, 并且由于数据集解码恢复后图像边界区域会出现的吉布斯震荡现象, 图像经过 LBP 变换后边界会存在大量条状噪声, 由于红外图像纹理相对于可见光图像更加简单, 且对比度低、信噪比低, 受噪声影响更为严重。

若将图像以 3×3 为窗口计算均值, 去除高频的细节信息, 将这个均值作为当前小区域的灰度值, 再与相邻的小区域灰度进行比较, 进行二进制编码形成新的 LBP 特征。图 3 为参考背景和 991 帧左上角相同无前景区域处, 由于噪声等影响, 导致该位置得到的 LBP 值不同, 造成前景提取时将背景检测为前景的情况。为了抑制动态背景和噪声对目标提取的影响, 通过自适应阈值改进了 LBP 算法, 初步解决复杂背景和噪声的问题, 然后结合感知哈希算法中的汉明距离, 进一步抑制动态背景和噪声, 提出了一种自适应阈值的 Hash_LBP 算法。

首先, 将得到的参考背景与每一帧以 3×3 为窗口, 求取灰度平均值。背景平均值为 \bar{h}_b , 第 i 帧平均值 \bar{h}_i

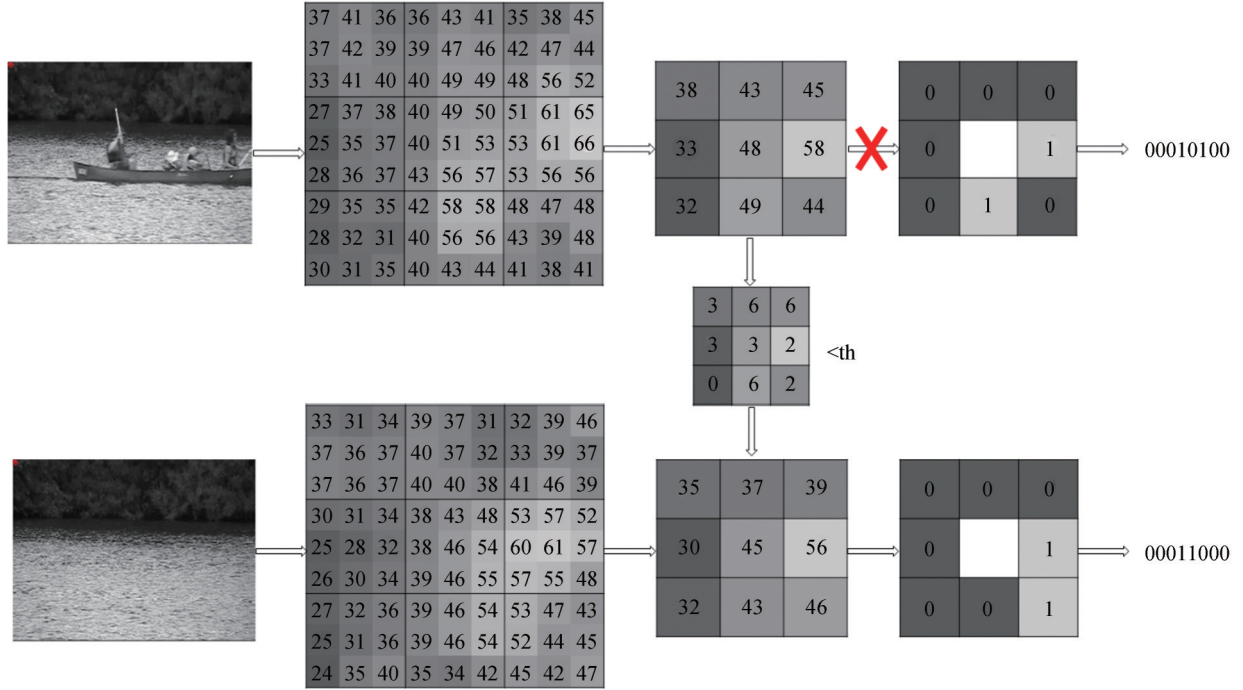


图3 自适应阈值的hash_LBP算子

Fig. 3 Adaptive threshold of hash_LBP operator

与 \bar{h}_b 进行比较,得到差值如果小于阈值 th (th 由原始图像矩阵与背景建模矩阵中的每个像素点一一对应相减得到的差值矩阵经过校正得到)则将 \bar{h}_b 值赋给 \bar{h}_i ,否则不变。最后,与相邻区域灰度平均值进行比较形成LBP特征,得到hash_LBP值,可表示为

$$\text{Hash_LBP}_{N,R}(X, Y) = \sum_{N=0}^{N-1} S(\bar{h}'_i - \bar{h}_c) \times 2^N \quad (5)$$

$$\bar{h}_i = \text{avg}(h) \quad (6)$$

$$\bar{h}'_i = \begin{cases} \bar{h}_i & |\bar{h}_i - \bar{h}_b| > t_h \\ \bar{h}_b & |\bar{h}_i - \bar{h}_b| \leq t_h \end{cases} \quad (7)$$

$$S(\bar{h}'_i - \bar{h}_c) = \begin{cases} 1 & \bar{h}'_i - \bar{h}_c \geq 0 \\ 0 & \bar{h}'_i - \bar{h}_c < 0 \end{cases} \quad (8)$$

式中, \bar{h}_c 表示中心小区域的灰度值,以小区域 (X, Y) 圆心,半径 R 的圆区域称为邻域, N 表示邻域像素点的个数, \bar{h}'_i 为改进后的邻域灰度值, \bar{h}_b 为参考背景模型帧的均值灰度值。

最后对得到LBP值使用汉明距离进行约束,比较当前帧和参考背景每个像素LBP值的8位二进制序列,计算不相同的位数。若不相同位数小于3则该点LBP值为背景该点的LBP值,否则保持当前帧该点的LBP值。当前帧与背景的LBP值差值的绝对值图可以反映动态背景的抑制效果,使用汉明距离约束后可以减少部分动态噪声的影响,如图4。

由图可知,在同一帧的场景中,在使用汉明距离约束以后,湖面的干扰减少了很多,背后的树丛干扰也得到了较好的抑制,说明通过汉明距离的约束后,一定程度上抑制了动态背景。

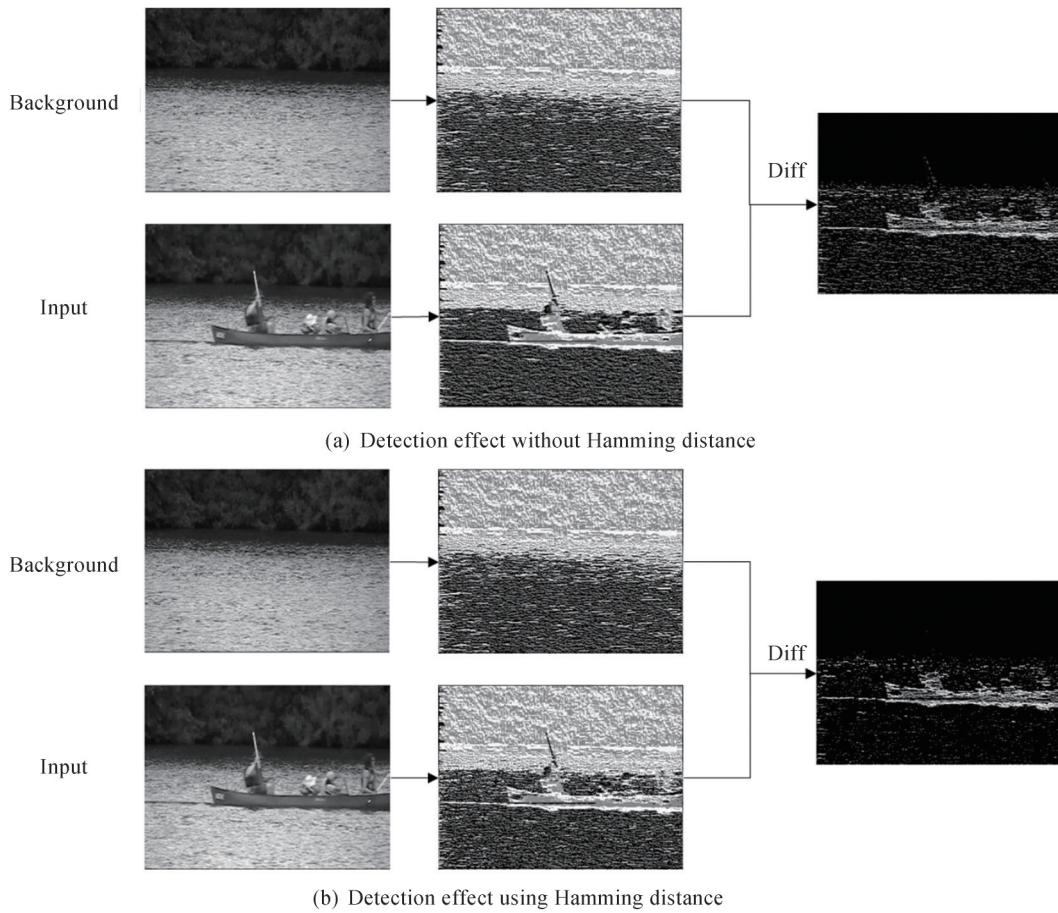


图4 使用汉明距离约束后的检测效果
Fig. 4 Detection effect after using Hamming distance constraint

1.3 自适应阈值的设定

通过对每种动态背景的分析,如图5所示,该数据集为blizzard,场景为下着暴风雪的马路,背景中存在大量雪花,且汽车经过时车灯会造成场景光照变化。随机选取图中一点,如图5(a)中红点标记(400, 400)处,统计连续200帧该点灰度值变化曲线,如图5(b)。图中背景灰度值频繁变化,但波动范围在两条红色虚线区间范围内,而有目标经过后会有更大波动的变化,如图5(b)中两条黑色虚线区间。由于动态背景灰度值波动在一定区间内,可以通过计算参考背景和每帧图像复杂背景波动区间大小作为阈值,对每帧图像进行校正。

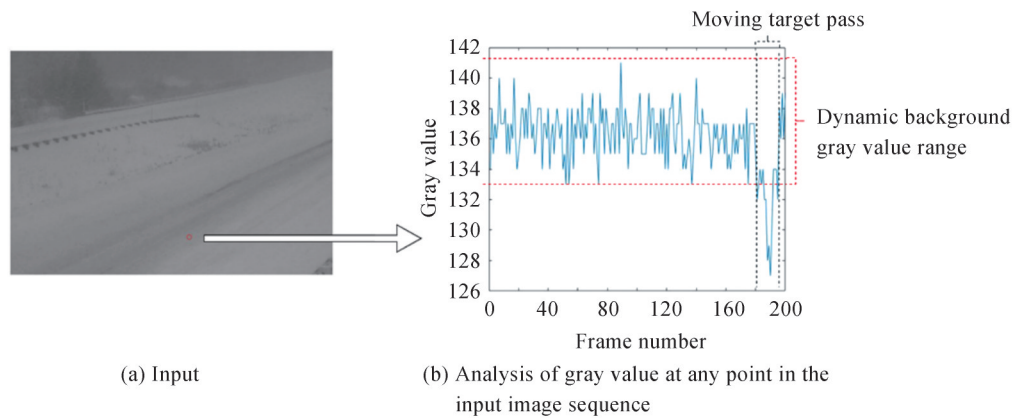


图5 背景分析
Fig.5 Complex background analysis

由于固定的阈值不同数据集的鲁棒性差,因此采用自适应的阈值作补偿差值,采取差值矩阵的标准差进行阈值更新,具体的操作流程如下:

假设目前所使用的数据集尺寸 $M \times N$ 为当前帧的图像矩阵中行和列的大小, $f_{x,y}$ 为图像中每个像素点的灰度值, $x \in [1, M]$, $y \in [1, N]$, 整个当前原始图像矩阵记为 F 。同样大小下的背景建模图像矩阵记为 B , 每个像素点的灰度值为 $B_{x,y}$, 首先将原始图像矩阵与背景建模矩阵中的每个像素点一一对应相减得到差的绝对值, 得到差值矩阵 T , 其表达式为

$$T_{x,y} = \begin{cases} |B_{x,y} - F_{x,y}| & 10 < |B_{x,y} - F_{x,y}| < 100 \\ 0 & \text{else} \end{cases} \quad (9)$$

差值矩阵中, 仅计算矩阵中大于 10 小于 100 的值并求取其差值矩阵平均值 M , 即

$$M = \frac{1}{M \times N} \sum_1^{M \times N} (|B_{x,y} - F_{x,y}|) \quad x \in [1, M], y \in [1, N] \quad (10)$$

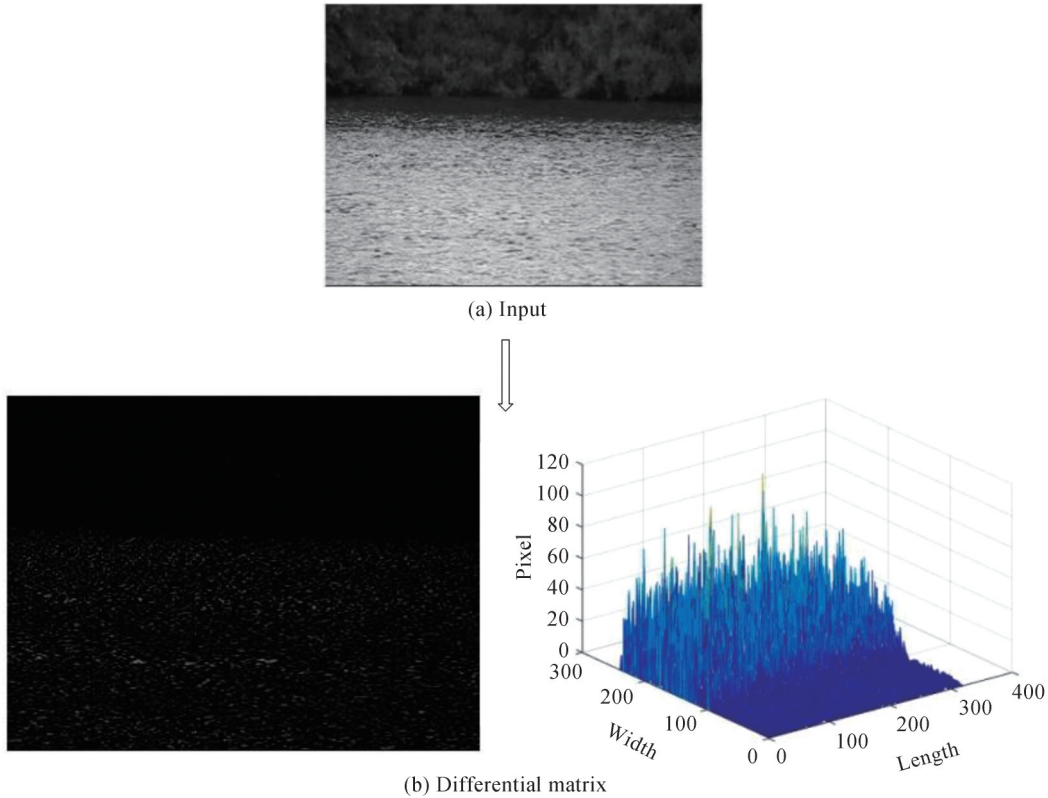
为了更好地衡量差值矩阵中的数据波动, 继续求取差值矩阵的标准差 $\partial(M)$, 表示为

$$\partial(M) = \sqrt{\frac{1}{M \times N} \sum_1^{M \times N} [(T_{x,y} - M)]^2} \quad (11)$$

则阈值可表示为

$$\text{th} = \begin{cases} 20 & \partial(M) \geq 10 \\ \partial(M) + M & \partial(M) < 10 \end{cases} \quad (12)$$

自适应阈值的处理效果如图 6。图 6(a) 为数据集 canoe 中无目标出现, 随机选取的一帧图像, 背景中存在大量闪烁湖面的动态背景。图 6(b) 为该帧图像与参考背景的绝对值差值矩阵图和以图像大小为 X 轴 Y 轴, 差值为 Z 轴的三维差值直方图, 通过三维差值直方图和差值矩阵图反映出存在大量动态背景。图 6(c) 为通过自适应阈值校正后, 该帧图像与参考背景的差值矩阵图与三维差值直方图, 可以看出噪点明显减少。



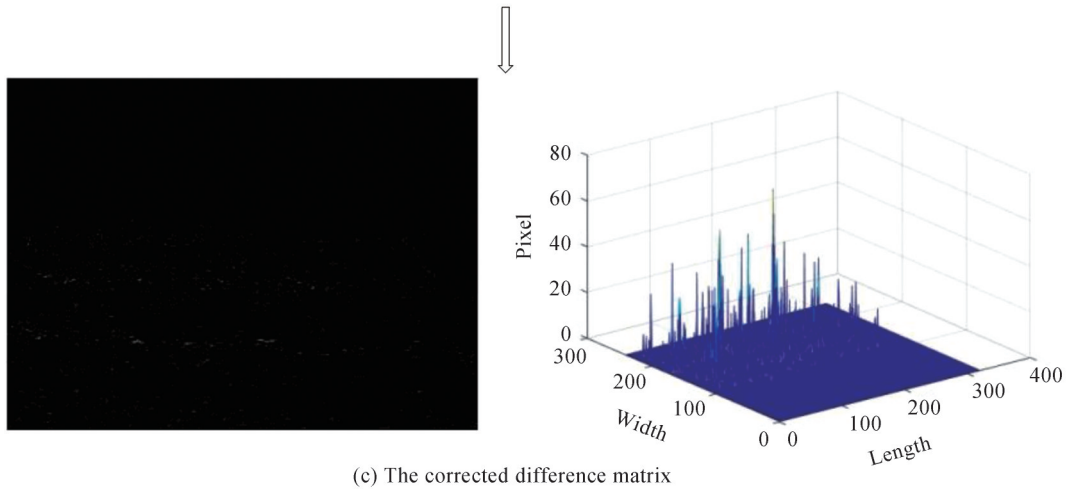


图6 自适应阈值抑制动态背景示意

Fig.6 Schematic of adaptive threshold suppression dynamic background

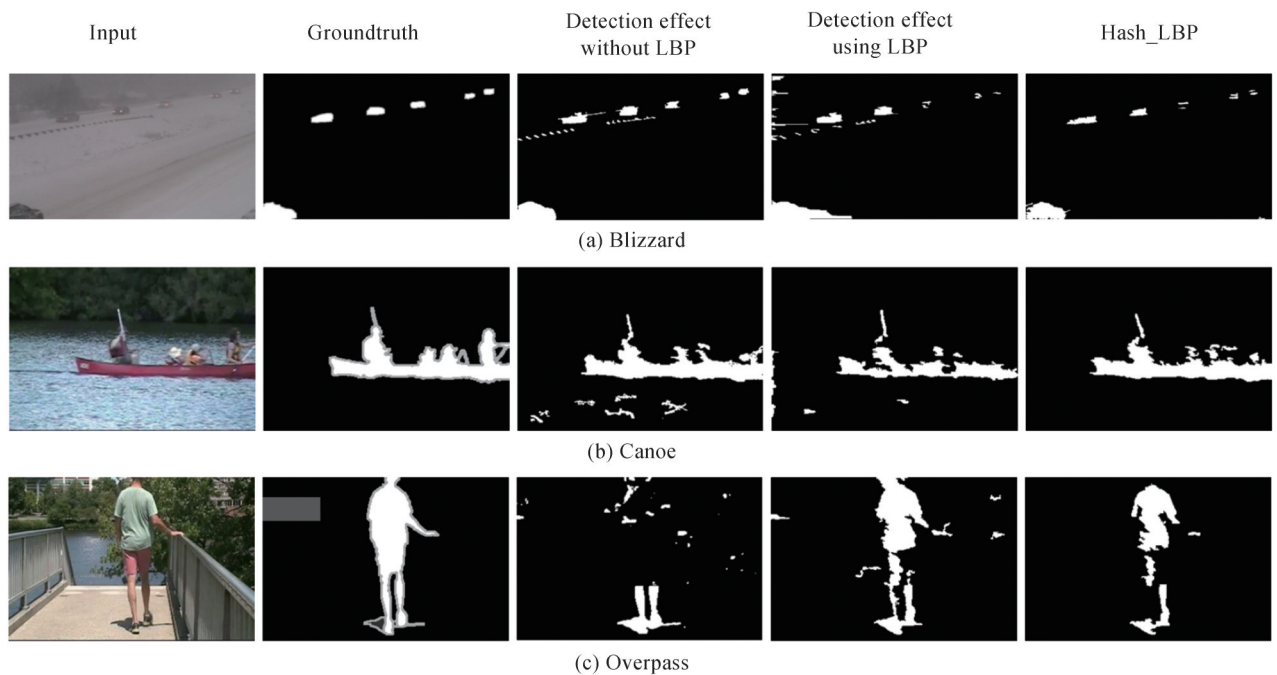
2 实验结果分析

为了能够对所提算法进行定量分析及性能验证,使用了 Change Detection.net2014 公共数据集^[28]中的 6 组复杂可见光场景图像和 5 组红外图像进行对比实验,这 11 组视频分别为 canoe、overpass、highway、skating、blizzard、snow Fall、corridor、dining room、Lake Side、library、park,这些视频包含了生活中常见的复杂场景^[29-30]。

首先对所提算法进行纵向对比,将未结合 LBP 的背景建模算法运动目标检测结果、仅结合 LBP 算法运动目标检测结果与引入汉明距离后的运动目标检测结果进行对比。然后将所提算法与现有背景建模算法 CODE_BOOK^[9]、LBP_MRF^[12]、ViBe^[11]、KDE^[7]、GMM^[6] 的检测结果进行横向对比,对比算法均使用原文默认参数。最后将所提的 Hash_LBP 算法与 ViBe 和 GMM 算法结合,验证 Hash_LBP 的可行性。数据集处理的硬件为 i7-8700K 3.70GHz CPU, 32 GB 内存,显卡 NVIDIA GeForce GTX 1080 Ti, 软件平台使用为: Win10、vs2019 和 OpenCV4.2.0 的环境。

2.1 算法纵向对比实验

如图 7 所示,随机选取 Cdnet2014 数据集中 5 个视频序列中的任意一帧处理结果,在同一个数据集中不



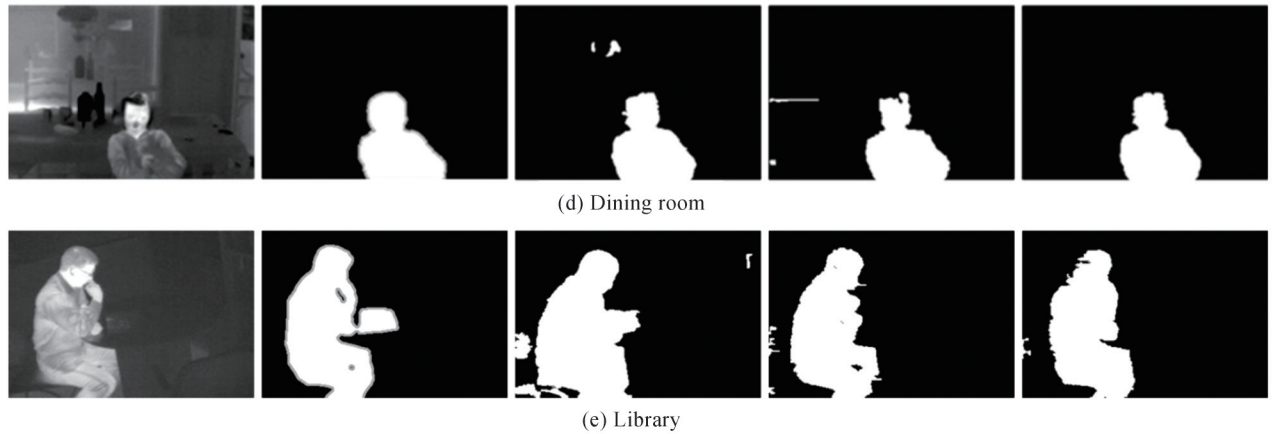


图7 算法纵向对比实验

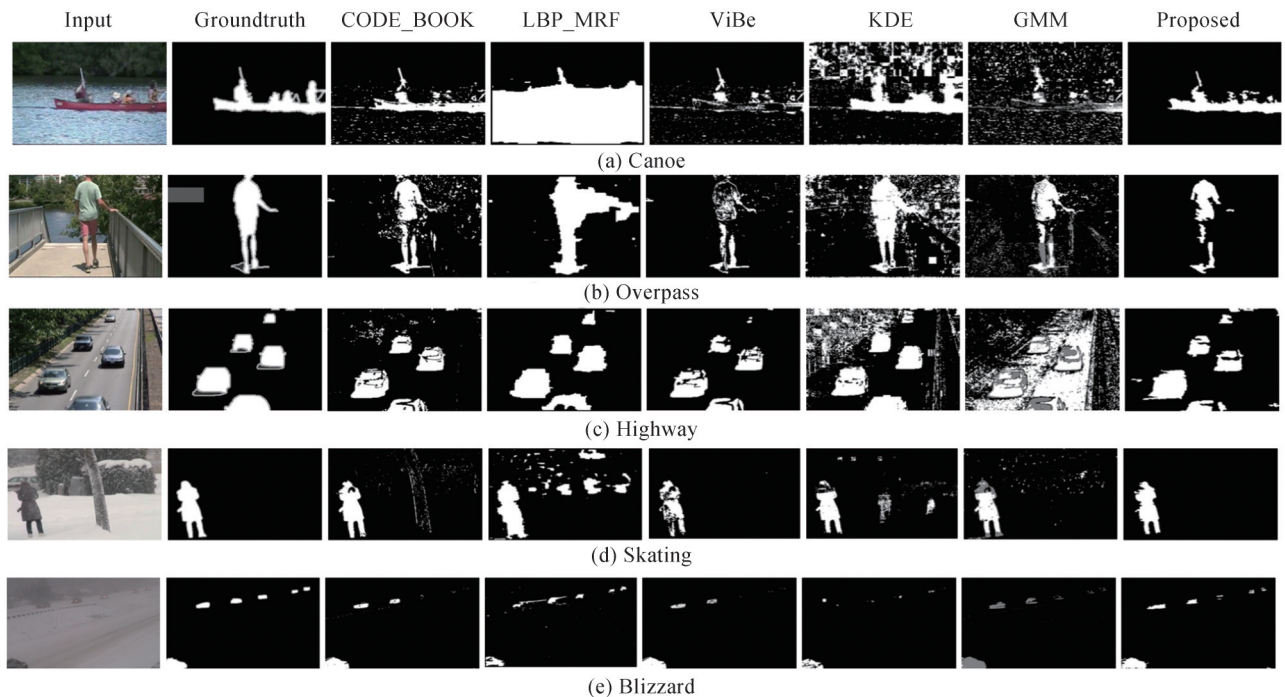
Fig.7 Longitudinal comparison results of the algorithm

同算法下选取同一帧处理结果进行比较,其中图7(a)、(b)、(c)为可见光数据集,图7(d)、(e)为红外数据集。由图7第三列图像可知,处理的数据集中的图像由于对比度较低,且背景变化频繁,导致处理结果从主观上就出现了较多明显的误检,图7(a)中由于光照变化,将路面边缘误检为了前景目标;图7(b)中由于水面闪烁,湖面部分出现了明显的误检;图7(c)中由于背景中树木被风吹动摇摆的影响,背景将部分树木误检测为前景,并且人体结构也并不完整;图7(d)、(e)均是红外图像,图像信噪比较低,但也同样出现了较大的误检。从图7第四列的图像中看到,在引入LBP算法后,将背景误检为前景的区域明显减少,但是由于LBP自身算法的原因导致出现了不同程度的带状噪声,因此又引入了汉明距离对LBP算法进行约束。从图7最后一列可以看到,所提的Hash_LBP算法的处理结果对比前几种方法,目标和背景分割最为准确,很好地解决了动态背景和噪声问题,且运动目标的形状轮廓也较为完整。

2.2 算法检测效果对比实验

2.2.1 可见光复杂场景的对比实验

图8中复杂场景所选取的数据集为Cdnets2014数据集中的Bad weather的canoe、overpass、highway、skating、blizzard和snowfall,各种算法的主观对比结果如图8。



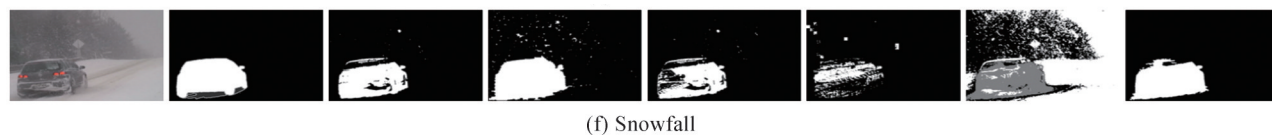


图8 可见光复杂场景下不同算法处理结果
Fig.8 Processing results of different algorithms in complex visible light scenes

从图8可以看到,在六个可见光的复杂场景中,无论是动态背景还是动静结合的背景场景中,所提算法得到的前景目标具有更好的完整性和准确性,尤其是对动态背景的抑制,和其他5个算法相比得到的背景更为干净准确,得到的前景目标误检像素点更少。图8(a)中能够明显看出所提算法对动态背景有很好的抑制作用,CODE_BOOK、LBP_MRF、ViBe三个算法相比KDE和GMM两个算法,对图像上半部分的树丛处理的较好,但是在湖面的抑制仍然不如所提算法,而所提算法对船上人的提取不够完整清晰;图8(b)和图8(c)都是动静结合的场景,在两个场景中,除了LBP_MRF和所提算法,其他算法均在不同程度上将背景判别为前景,但是LBP_MRF算法,在图8(2)中仍然将行人与误判的树叶背景粘连在了一起,因此从主观视觉上所提算法较完整检测出前景的同时很好地抑制了动态背景的影响;图8(d)中除了ViBe算法和所提算法,其余算法都将大部分飘动的雪花以及树叶上的积雪当作了前景,但和ViBe算法相比,所提算法中的前景目标更为完整;图8(e)和图8(f)均为恶劣雪天的室外场景,图8(e)中的信噪比更低,其中,LBP_MRF算法的结果中出现了拖影,其他算法均出现了前景目标提取不完整的现象,而图8(f)中除了所提算法,其他5个算法均受到了动态飘落雪花的影响,但在前景提取上码本和所提算法无法直接判断谁的结果更为准确。

2.2.2 红外复杂场景的对比实验

由于红外数据集中的动态背景影响较小,且红外数据集具有分辨率、对比度和信噪比低,视觉效果模糊等特点,因此,在对比度低的情况下,还需抑制图像解压编码时产生的噪声^[31-32],不同算法的主观对比结果如图9。

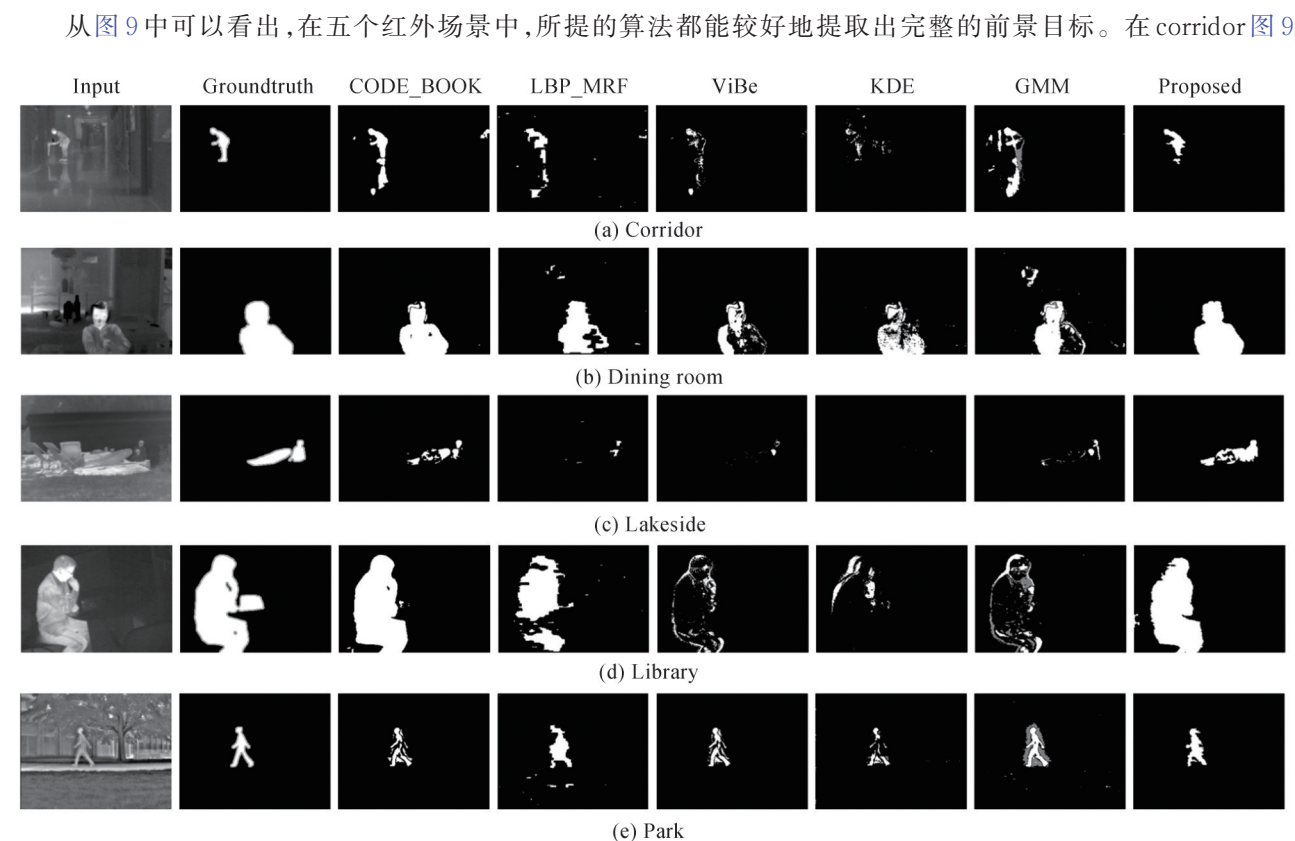


图9 红外复杂场景中各算法的对比实验结果
Fig.9 Comparative experimental results of two algorithms in infrared scene

(a)中,除了码本和所提算法,其他算法均出现了较大的空洞,且码本错误地将人物倒影也当作了前景;在 dining room 图 9(b)中 LBP_MRF 和 GMM 算法受到背景中飘动的旗帜的影响,将其误检成为了前景目标,除了所提算法,其他算法均在不同程度上出现了目标不完整;在 lakeside 图 9(c)中 KDE 算法几乎没有检测到前景目标, LBP_MRF、ViBe、GMM 的前景目标点大量缺失,只有码本和所提算法将前景目标尽可能完整地进行了提取;在 library 图 9(d)中除了所提算法,其他算法在不同程度上都将大部分前景目标当成了背景;在 park 图 9(e)中背景和前景的灰度差异比较小,其他算法提取的前景都在一定程度上不完整,所提算法目标提取较为完整,但目标边缘处还是存在部分条状噪声。

2.3 定量分析

为了对所提算法进行定量分析,采用了 CDet2014 中使用的召回率 R_c (Recall)、准确率 P_r (Precision)、 F 测度 (F-Measure)、误检百分比 P_{PWC} (Percentage of Wrong Classification) 四个评价指标来衡量。

$$P_r = \frac{f_{TP}}{f_{FP} + f_{TP}} \quad (13)$$

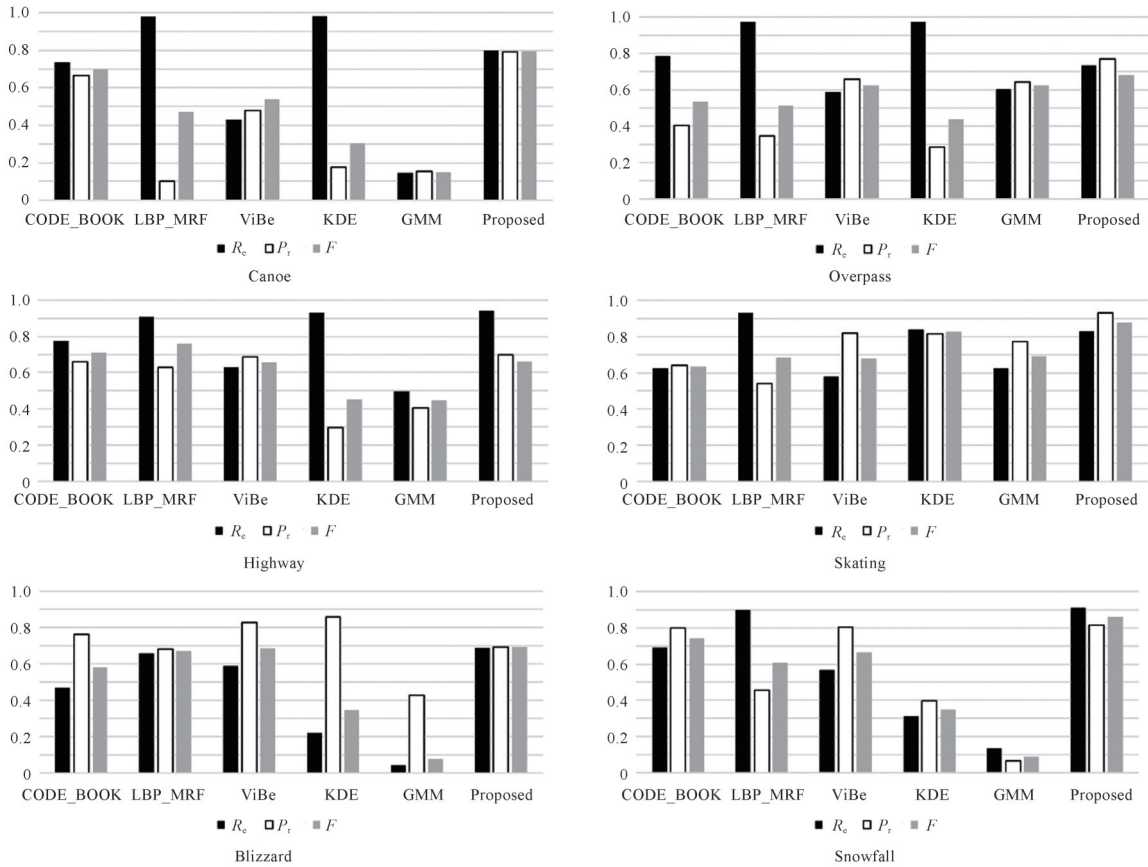
$$P_{PWC} = \frac{f_{FP} + f_{FN}}{f_{TP} + f_{TN} + f_{FP} + f_{FN}} \times 100\% \quad (14)$$

$$R_c = \frac{f_{TP}}{f_{TP} + f_{FN}} \quad (15)$$

式中, P_r 和 R_c 指标有时候会出现矛盾的情况,这就需要综合考虑, F 测度 F-Measure 就是准确率和召回率的加权调和平均,用来衡量算法的综合性能。其表达式为

$$F = \frac{2 \times R_c \times P_r}{R_c + P_r} \quad (16)$$

式(13)~(16)中, f_{TP} 为正确检测为前景的点, f_{TN} 为正确检测为背景的点, f_{FP} 为误检为前景的点, f_{FN} 为误检为背景的点。一般情况下, R_c 、 P_r 、 F 测度均越大越好, P_{PWC} 越小越好。11 种场景的分析结果如图 10 和表 1 和表 2 所示。



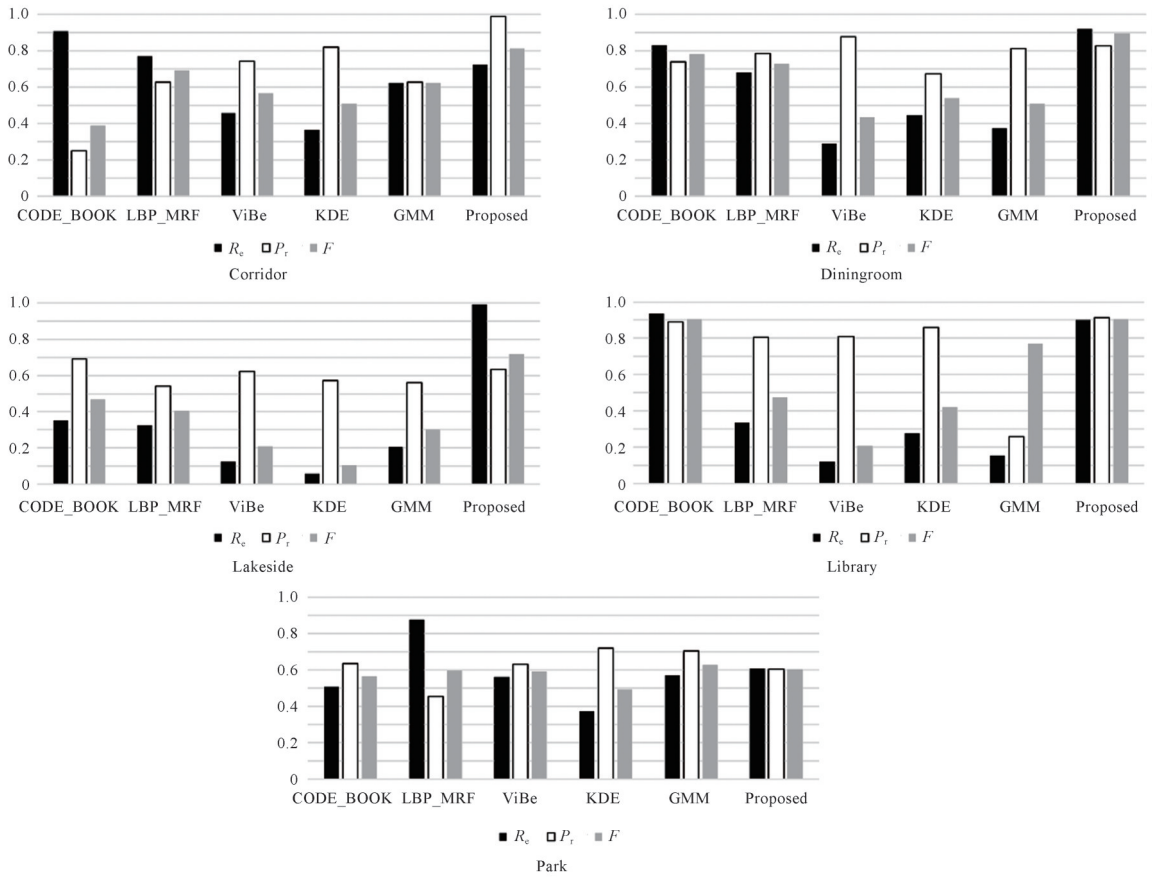


图 10 各场景下 6 个算法的 R_e 、 P_r 、 F 直方图对比
 Fig.10 Comparison of R_e , P_r and F histograms of six algorithms in each scenario

表 1 各算法在不同可见光场景下的误检率

Table 1 False detection rate of each algorithm in different visible scenes

Algorithm	Cano	Overpass	Highway	Skating	Blizzard	Snowfall
CODE_BOOK	0.034 8	0.055 6	0.037 1	0.068 5	0.012 7	0.012 5
LBP_MRF	0.185 5	0.100 5	0.036 1	0.081 3	0.013 3	0.024 4
ViBe	0.051 7	0.038 7	0.039 0	0.048 6	0.010 9	0.011 9
KDE	0.170 8	0.134 0	0.130 9	0.033 4	0.016 7	0.024 2
GMM	0.062 3	0.039 5	0.073 1	0.049 3	0.020 7	0.056 9
Proposed	0.022 5	0.030 8	0.004 7	0.018 4	0.010 4	0.046 5

表 2 各算法在不同红外场景下的误检率

Table 2 False detection rate of each algorithm in different infrared scenes

Algorithm	Corridor	Diningroom	Lakeside	Library	Park
CODE_BOOK	0.099 8	0.039 7	0.016 5	0.038 9	0.018 3
LBP_MRF	0.024 0	0.042 7	0.018 4	0.165 9	0.019 2
ViBe	0.024 1	0.063 3	0.021 0	0.202 2	0.017 9
KDE	0.024 5	0.065 0	0.019 0	0.171 0	0.018 0
GMM	0.025 8	0.060 3	0.021 2	0.198 3	0.015 7
Proposed	0.028 6	0.024 3	0.009 4	0.041 0	0.015 1

图 10 为每个算法 R_e 、 P_r 、 F 数的柱状图,其中黑色柱为 R_e ,白色柱为 P_r ,灰色柱为 F 数,纵坐标范围为 0 到 1,横坐标依次为 CODE_BOOK、LBP_MRF、ViBe、KDE、GMM、所提算法,每个算法都对 R_e 、 P_r 、 F 数分别进行统计。可以看到所提算法在绝大部分场景中都有着较好的指标,11 个场景中所提算法综合性能最优。

从表1~2中可以看到,所提算法在大部分场景中的误检率都是最低的(最低数值在表中加粗和下划线标出),所提算法在综合性能较好的同时也能保持较低的误检率。测试得到的 R_c 、 P_r 、 F 和 P_{PWC} 所有数据如表3所示。

表3 各场景下不同算法的 R_c 、 P_r 、 F 和 P_{PWC} 数值
Table 3 Values of R_c 、 P_r 、 F and P_{PWC} of different algorithms in each scenario

Dataset	Algorithm	R_c	P_r	F	P_{PWC}	Dataset	Algorithm	R_c	P_r	F	P_{PWC}
Cano	CODE_BOOK	0.737 6	0.665 5	0.699 7	0.034 8	Blizzard	CODE_BOOK	0.467 5	0.760 4	0.579 1	0.012 7
	LBP_MRF	0.978 9	0.102 4	0.472 9	0.185 5		LBP_MRF	0.657 4	0.680 7	0.668 8	0.013 3
	ViBe	0.428 3	0.477 0	0.538 2	0.051 7		ViBe	0.587 6	0.826 5	0.686 9	0.010 9
	KDE	0.984 0	0.178 1	0.301 6	0.170 8		KDE	0.216 8	0.857 0	0.346 1	0.016 7
	GMM	0.146 2	0.153 0	0.149 5	0.062 3		GMM	0.042 3	0.426 1	0.076 9	0.020 7
	Proposed	0.801 0	0.791 8	0.796 4	0.022 5		Proposed	0.690 5	0.692 1	0.691 3	0.010 4
Lakeside	CODE_BOOK	0.352 4	0.689 5	0.466 4	0.016 5	Overpass	CODE_BOOK	0.783 5	0.403 4	0.532 6	0.055 6
	LBP_MRF	0.326 2	0.539 4	0.406 6	0.018 4		LBP_MRF	0.972 4	0.345 5	0.509 9	0.100 5
	ViBe	0.124 6	0.621 7	0.207 7	0.021 0		ViBe	0.589 6	0.656 3	0.621 0	0.038 7
	KDE	0.057 4	0.569 9	0.104 3	0.019 0		KDE	0.974 2	0.283 2	0.438 9	0.134 0
	GMM	0.205 2	0.560 0	0.300 3	0.021 2		GMM	0.602 7	0.641 2	0.621 4	0.039 5
	Proposed	0.993 2	0.635 0	0.718 2	0.009 4		Proposed	0.736 0	0.770 7	0.682 3	0.030 8
Snowfall	CODE_BOOK	0.688 9	0.798 3	0.739 6	0.012 5	Library	CODE_BOOK	0.936 0	0.891 1	0.903 7	0.038 9
	LBP_MRF	0.898 3	0.456 8	0.605 6	0.024 4		LBP_MRF	0.333 7	0.806 8	0.472 1	0.165 9
	ViBe	0.567 6	0.803 9	0.665 4	0.011 9		ViBe	0.118 8	0.808 4	0.207 1	0.202 2
	KDE	0.311 5	0.397 1	0.349 2	0.024 2		KDE	0.277 2	0.857 4	0.419 0	0.171 0
	GMM	0.134 8	0.067 5	0.089 9	0.056 9		GMM	0.154 0	0.256 8	0.771 7	0.198 3
	Proposed	0.910 3	0.813 3	0.859 1	0.046 5		Proposed	0.899 9	0.914 5	0.907 2	0.041 0
Highway	CODE_BOOK	0.774 9	0.657 8	0.711 5	0.037 1	Corridor	CODE_BOOK	0.910 6	0.248 6	0.390 2	0.099 8
	LBP_MRF	0.905 9	0.627 5	0.758 6	0.036 1		LBP_MRF	0.769 0	0.628 5	0.691 7	0.024 0
	ViBe	0.628 6	0.685 5	0.655 8	0.039 0		ViBe	0.458 4	0.741 9	0.566 6	0.024 1
	KDE	0.927 8	0.298 8	0.452 0	0.130 9		KDE	0.367 5	0.821 3	0.507 8	0.024 5
	GMM	0.498 7	0.403 8	0.446 3	0.073 1		GMM	0.622 3	0.626 8	0.624 6	0.025 8
	Proposed	0.939 2	0.697 1	0.658 1	0.004 7		Proposed	0.723 6	0.989 2	0.813 4	0.028 6
Park	CODE_BOOK	0.508 6	0.635 7	0.565 1	0.018 3	Skating	CODE_BOOK	0.624 0	0.641 3	0.632 5	0.068 5
	LBP_MRF	0.877 8	0.453 3	0.597 9	0.019 2		LBP_MRF	0.930 7	0.540 5	0.683 9	0.081 3
	ViBe	0.560 7	0.632 8	0.594 6	0.017 9		ViBe	0.579 0	0.816 0	0.677 4	0.048 6
	KDE	0.374 3	0.720 0	0.492 5	0.018 0		KDE	0.838 0	0.814 0	0.825 8	0.033 4
	GMM	0.567 8	0.703 1	0.628 3	0.015 7		GMM	0.626 5	0.770 4	0.691 0	0.049 3
	Proposed	0.607 7	0.603 7	0.605 7	0.015 1		Proposed	0.829 5	0.930 1	0.876 9	0.018 4
Dining room	CODE_BOOK	0.829 2	0.736 7	0.780 2	0.039 7						
	LBP_MRF	0.680 3	0.784 2	0.728 6	0.042 7						
	ViBe	0.288 7	0.878 9	0.434 6	0.063 3						
	KDE	0.447 0	0.671 6	0.536 8	0.065 0						
	GMM	0.370 7	0.810 3	0.508 6	0.060 3						
	Proposed	0.921 4	0.827 0	0.894 9	0.024 3						

2.4 算法实时性对比结果

为体现算法的实时性,在相同的运行环境下,从时间和空间上分析了算法的复杂度。各个算法不同数据集平均每帧处理速度如表4。

可以看到 ViBe算法在多数场景下,都是处理速度最快的;LBP_MRF算法是将LBP和MRF结合起来,

表4 不同算法处理平均时长
Table 4 Average processing time of different algorithms

Scenes	Size	Time/ms					
		CODE_BOOK	LBP_MRF	ViBe	KDE	GMM	Proposed
Canoe	320×240	10.38	63.50	8.46	54.60	11.77	10.69
Overpass	320×240	10.31	20.70	8.41	51.35	8.88	15.47
Highway	320×240	10.41	53.33	8.10	46.43	8.79	13.61
Skating	540×360	12.31	94.28	12.18	52.43	11.37	20.11
Blizzard	720×480	14.86	193.62	16.61	52.54	15.81	31.47
Snowfall	720×480	15.63	229.10	19.36	55.63	15.33	31.54
Corridor	320×240	9.31	55.36	6.63	48.67	7.71	11.99
Diningroom	320×240	9.90	52.80	6.87	46.70	10.48	11.82
Lakeside	320×240	7.44	46.84	7.26	48.60	7.84	11.75
Library	320×240	7.02	52.50	12.29	46.84	15.78	11.77
Park	320×240	10.32	98.85	17.74	48.47	16.94	23.52

因此计算复杂度更高,耗时也更多;所提算法对LBP进行了改进,再结合新的建模算法,因此不如ViBe算法处理快,但相比同样结合了LBP算法的LBP_MRF算法,处理时长明显缩减了很多,可以满足目标检测实时性的需求。

由于所提算法与ViBe、GMM和Code_book检测速率相似,表5中分析了以下几种算法CPU的占用率和占用CPU内存的容量。

表5 CPU占用率和内存对比
Table 5 Comparison between CPU usage and memory

Algorithms	CPU usage	Memory/MB
ViBe	22%	216
GMM	29%	224
CODE_BOOK	24%	229
Proposed	22%	221

由表5可知,所提算法和其他三种算法对CPU的占用率和内存占用相差不大,所以所提算法也非常适合移植到嵌入式平台上工作,满足各种特殊场合下的实时性要求。

2.5 普遍适应性分析

实验表明所提算法对动态背景的抑制对于ViBe和GMM等背景建模算法同样适用。

将所提汉明距离约束的Hash_LBP算法用于ViBe算法和GMM算法,图11(a)为测试数据集canoe,图11(b)为其真值图,图11(c)、(e)为ViBe和GMM背景建模算法目标检测结果,此时前景提取中存在大量动态背景。ViBe和GMM通过汉明距离约束的Hash_LBP算法处理后,抑制了大部分动态背景,如图11(d)、(f)。接着对其进行定量分析,结果如表6。

表6 不同算法下的对比结果
Table 6 Comparison results under different algorithms

Dataset	Cano			
	R_c	P_r	F	P_{pwc}
ViBe	0.428 3	0.477 0	0.538 2	0.051 7
New_ViBe	0.488 0	0.621 7	0.546 8	0.044 2
GMM	0.146 2	0.153 0	0.149 5	0.062 3
New_GMM	0.608 0	0.758 3	0.674 9	0.034 6

可知,通过汉明距离约束的Hash_LBP算法处理过的ViBe算法, R_c 提高了13.9%, P_r 提高了30.3%, F 数提高了1.6%, P_{pwc} 减少了14.5%。通过汉明距离约束的Hash_LBP算法处理过的GMM算法, R_c 提高了315.9%, P_r 提高了395.6%, F 数提高了351.4%, P_{pwc} 减少了44.5%。

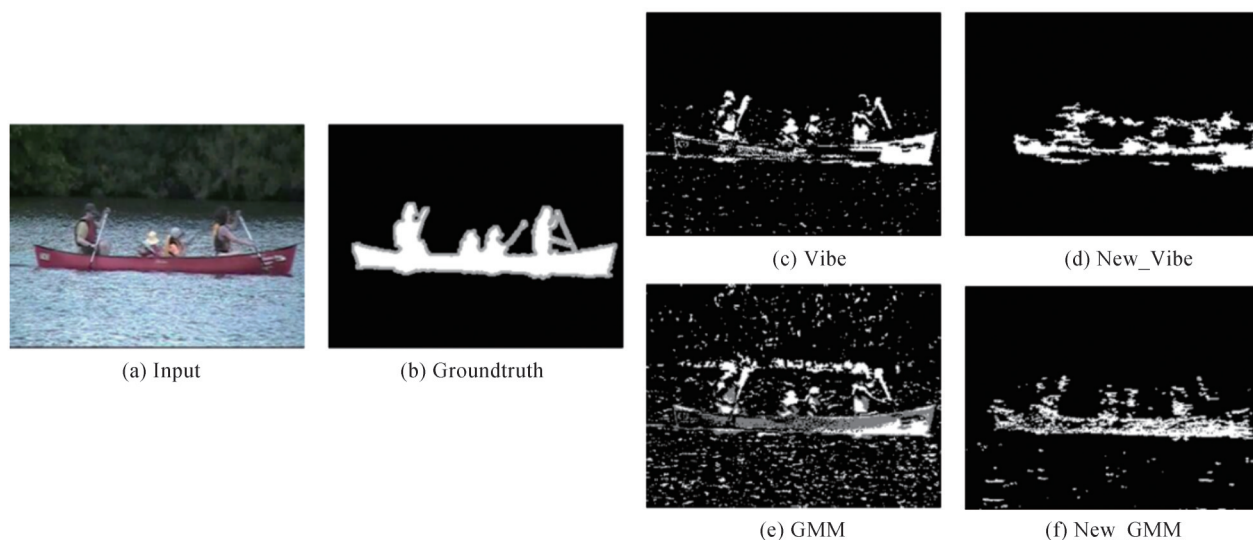


图11 Hash_LBP在ViBe和GMM中检测结果
Fig.11 Hash_LBP detects results in ViBe and GMM

3 结论

本文提出了一种自适应阈值的Hash_LBP算法,主要研究如何抑制动态背景和噪声问题。首先,根据图像序列的频域特性,对每一个像素建立了灰度频次直方图,将频次最高的灰度值作为参考背景,然后参考哈希算法选取 3×3 的窗口求取均值,对图像做平滑处理,将参考背景与每帧比较得到差值矩阵,计算差值矩阵中大于10小于100的值的标准差与均值之和,作为自适应阈值校正当前帧灰度值来抑制动态背景,并对参考背景和校正后的每帧通过LBP算法得到图像的纹理特征。为了更好地抑制动态背景和噪声,将参考背景的LBP值与每帧LBP值的汉明距离,以3为阈值校正每帧LBP值来稳定图像,进一步抑制动态背景和噪声。实验结果表明,所提算法可以很好地抑制动态背景,提高了复杂背景环境下可见光和红外运动目标检测的召回率、准确率、 F 数,降低了误检率,综合指标均优于其他算法。并且有较高的鲁棒性,适应多种背景建模算法,无论是可见光数据集还是红外数据集均适用。但所提算法会吸收部分目标,存在目标检测不完整和鬼影问题,接下来将进一步对算法进行优化,研究孔洞填充算法和快速消除鬼影算法,以及对检测到的目标进行识别和跟踪。

参考文献

- [1] PALERMO M, MOCCIA S, MIGLIORELLI L, et al. Real-time human pose estimation on a smart walker using convolutional neural networks[J]. *Expert Systems with Applications*, 2021, 184:115498.
- [2] KIM H, DAVID O, KSENIA K, et al. Evaluation of a computer vision-based system to analyse behavioral changes in high school classrooms[J]. *International Journal of Information and Communication Technology Education*, 2021, 17(4): 1-12.
- [3] CHEN C, ZHU W, NORTON T. Behaviour recognition of pigs and cattle: Journey from computer vision to deep learning [J]. *Computers and Electronics in Agriculture*, 2021, 187(1-3):106255.
- [4] SINGH K, YADAV D K. Adaptive background modelling technique for moving object detection in video under dynamic environment[J]. *International Journal of Spatio-Temporal Data Science*, 2019, 1(1):4-21.
- [5] WANG Siming, HAN Lele. Moving target detection in complex dynamic background [J]. *Opto-Electronic Engineering*, 2018, 45(10):46-52.
王思明, 韩乐乐. 复杂动态背景下的运动目标检测[J]. *光电工程*, 2018, 45(10):46-52.
- [6] ARVIND K T. Deep learning and its applications[M]. New York: Nova Science Publishers, Inc, 2021.
- [7] SONG Y, CHAO M, WU X, et al. VITAL: visual tracking via adversarial learning[C]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018.

- [8] RIEKERT M, KLEIN A, ADRION F, et al. Automatically detecting pig position and posture by 2D camera imaging and deep learning[J]. Computers and Electronics in Agriculture, 2020, 174:105391.
- [9] ROMERO J D, LADO M J, MENDEZ A J, et al. A background modeling and foreground detection algorithm using scaling coefficients defined with a color model called lightnessred-green-blue[J]. IEEE Transactions on Image Processing, 2018, 27(3): 1243-1258.
- [10] PANDO A G, MURGUIA M I. Analysis and trends on moving object detection algorithm techniques[J]. IEEE Latin America Transactions, 2019, 17(11): 1771-1783.
- [11] ZHAO Z, LU G. Target motion detection algorithm based on dynamic threshold[J]. Journal of Physics: Conference Series, 2021, 1738(1):012085.
- [12] WREN C R, AZARBAYEJANI A. Pfnder: real-time tracking of the human body[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(7): 780-785.
- [13] STAUFFER C, GRIMSON W. Adaptive background mixture models for real-time tracking[C].1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999.
- [14] ELGAMMAL A, HARWOOD D, DAVIS L S, et al. Non-parametric model for background subtraction[C].European Conference on Computer Vision, 2000.
- [15] LEE D. Effective gaussian mixture learning for video background subtraction[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(5): 827-832.
- [16] KIM K, CHALIDABHONGSE T H, HARWOOD D, et al. Real-time foreground - background segmentation using codebook model[J]. Real-Time Imaging, 2004, 11(3): 172-185.
- [17] CLÁUDIO R J. Efficient background subtraction and shadow removal for monochromatic video sequences. [J]. IEEE Transactions on Multimedia, 2009, 11(3): 571-577.
- [18] BARNICH O, VAN DROOGENBROECK M. ViBe: a universal background subtraction algorithm for video sequences [J]. IEEE Transactions on Image Processing, 2011, 20(6): 1709-1724.
- [19] OY V. Texture-based foreground detection[J]. International Journal of Signal Processing, Image Processing and Pattern Recognition, 2011, 4(4):51-62.
- [20] ZHANG Guangshi, GE Guangying, ZHU Ronghua , et al. Facial expression recognition based on LBP feature and deep learning [J]. Computer Measurement and Control. 2020, 28(2):174-178.
张广世, 葛广英, 朱荣华, 等. 结合 LBP 特征和深度学习的人脸表情识别[J]. 计算机测量与控制, 2020, 28(2): 174-178.
- [21] SUBHABRATA A, PRADIPTA K N. Adjacent LBP and LTP based background modeling with mixed-mode learning for foreground detection[J]. Pattern Analysis and Applications, 2021(1):1-28.
- [22] YANG X, LIU T. Moving object detection algorithm based on improved visual background extractor[J]. Journal of Physics: Conference Series, 2021, 1732(1):012078.
- [23] GARG K, RAMAKRISHNAN N, PRAKASH A, et al. Rapid and robust background modeling technique for low-cost road traffic surveillance systems[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, (99):1-12.
- [24] ERICHSON N, BENJAMIN B, STEVEN L, et al .Compressed dynamic mode decomposition for background modeling [J]. Journal of Technology, 2019.16(5):1479-1492.
- [25] LAI A N, YOON H, LEE G. Robust background extraction scheme using histogram-wise for real-time tracking in urban traffic video[C] . Computer and Information Technology, 2008.
- [26] DELP III E J, COSKUN B, MIHAK M K, et al. Perceptual hash based blind geometric synchronization of images for watermarking[C]. SPIE, 2008.
- [27] SHEN Qi, ZHAO Yan, ZHOU Xiaowei, et al. Image hash algorithm combining structure and gradient [J]. Journal of Zhejiang University (Engineering Science), 2020, 54(8):1525-1533.
沈麒, 赵琰, 周晓炜, 等. 结合结构与梯度的图像哈希算法[J]. 浙江大学学报(工学版), 2020, 54(8):1525-1533.
- [28] YI W, JODOIN P M, PORIKLI F, et al. CDnet 2014: an expanded change detection benchmark dataset[C]. Computer Vision & Pattern Recognition Workshops, IEEE, 2014.
- [29] WU Jiawen, WANG Shiyong. Gray video adaptive background modeling algorithm based on statistics [J]. Chinese Journal of Lasers ,2021, 48(3):113-125.
武加文, 王世勇. 基于统计的灰度视频自适应背景建模算法[J]. 中国激光, 2021, 48(3):113-125.
- [30] MIN Weidong, GUO Xiaoguang ,HAN Qing . An improved ViBe algorithm and its application in traffic video processing [J]. Optics and Precision Engineering, 2017, 25(3):806-811.
闵卫东, 郭晓光, 韩清. 改进的 ViBe 算法及其在交通视频处理中的应用[J]. 光学精密工程, 2017, 25(3):806-811.
- [31] LIU Mingwei, WANG Renhua, LI Jing, et al. Infrared and visible image Fusion based on anisotropic guided filtering [J]. Journal of Image and Graphics of China, 2021, 26(10):2421-2432.
刘明葳, 王任华, 李静, 等. 各向异性导向滤波的红外与可见光图像融合[J]. 中国图象图形学报, 2021, 26(10):2421-2432.

- [32] GRASSI D, POLITI R, IGNATIEV N I, et al. The venus nighttime atmosphere as observed by the VIRTIS-M instrument. average fields from the complete infrared data set[J]. *Journal of Geophysical Research: Planets*, 2014, 119(4): 837-849.

Fast Hash_LBP Moving Target Detection Algorithm Based on Hamming Distance Constraint in Complex Background

QIU Liya^{1,2,3}, CHEN Weilin^{1,2,3}, LI Fanming^{1,3}, LIU Shijian^{1,3}, WANG Xiaoyu^{1,2,3},
LI Linhan^{1,2,3}

(1 *Shanghai Institute of Technical Physics, Chinese Academy of Sciences, Shanghai 200083, China*)

(2 *University of Chinese Academy of Sciences, Beijing 100049, China*)

(3 *Key Laboratory of Infrared System Detection and Imaging Technology, Chinese Academy of Sciences, Shanghai 200083, China*)

Abstract: Moving target detection algorithm is to detect the changing region in the input image sequence, so as to extract the target from the background. It is very important for subsequent target recognition and tracking. Due to the complex and changeable natural environment, there is a large number of dynamic backgrounds and changing lights in complex weather, such as rain, snow, fog, vegetation shaking and water surface fluctuation, which has always been the primary problem of moving target detection in complex scenes. In this paper, when the camera has a fixed angle of view, the background modeling algorithm is adopted. To suppress the problems of dynamic background, slow target absorption and image coding noise, based on ensuring real-time performance, the time-frequency domain and frequency-domain characteristics of the input image are analyzed. Because the dynamic background fluctuates in a certain gray range. Through the texture features processed by LBP, the influence of illumination change can be suppressed. An adaptive threshold moving target detection algorithm based on texture features is proposed. First of all, the algorithm converts the input image sequence into a grayscale image, and uses the perceptual hashing algorithm to calculate the average pixel value in the window of 3×3 to remove the high-frequency details in the image part and improve the calculation speed. Then, the dynamic background and noise are processed preliminarily. The frequency domain of the input image is analyzed, and the maximum frequency of the gray frequency distribution histogram of each pixel is counted as the reference background and compared with each frame to obtain the difference matrix. The standard deviation of all values greater than 10 and less than 100 in the difference matrix is calculated as the adaptive threshold, and the gray value of each frame is corrected. Then the dynamic background and noise are processed a second time. Firstly, the Local Binary Pattern (LBP) is used to process the reference background and each frame of the image after preliminary processing to obtain the LBP value. Then, the hamming distance 3 was selected as the threshold to correct the LBP value of each frame. Finally, the LBP texture feature map of each frame is analyzed from the frequency domain, and the background modeling and foreground extraction are carried out according to the polymorphic frequency distribution. To suppress the influence of illumination change, dynamic background and noise on foreground extraction, this paper proposes an improved Hash_LBP algorithm combined with a perceptual hash algorithm and uses Hamming distance to constrain it. Experiments show that the proposed algorithm can effectively suppress noise, illumination change and dynamic background in a variety of complex scenes such as infrared and visible light, quickly and accurately extract foreground targets, and the algorithm is also effective in dynamic background suppression for ViBe and GMM algorithms.

Key words: Computer vision; Moving target detection; Background modeling; Hash algorithm; LBP characteristic operator

OCIS Codes: 100.2000; 150.1135; 100.3008