

# 基于改进视觉背景提取算法的运动目标检测方法

王旭, 刘毅, 李国燕\*

天津城建大学计算机与信息工程学院, 天津 300384

**摘要** 针对视觉背景提取算法在复杂环境下检测出现鬼影现象、动态背景因素形成噪声干扰等问题, 提出一种改进的视觉背景提取算法。通过创建辅助样本集, 对复杂环境中像素点的重要特征信息进行收集; 引入像素点鬼影因子和区域复杂度分析, 自适应调节各像素点的匹配阈值和更新速率; 最后通过基于滑动窗的像素点闪烁程度分析, 将可能被误检为前景的噪声点向辅助样本中依概率更新。多场景下对比实验表明, 该算法可将错分率降低至 1.49%, 且在检测时能快速消除鬼影现象, 有效抑制动态背景产生的噪声干扰, 同时保证前景目标能被完整识别, 在复杂环境下的检测结果更加准确。

**关键词** 图像处理; 运动目标检测; 视觉背景提取算法; 鬼影消除; 自适应阈值; 动态背景

中图分类号 TP391

文献标识码 A

doi: 10.3788/LOP56.011007

## Moving Object Detection Algorithm Based on Improved Visual Background Extractor Algorithm

Wang Xu, Liu Yi, Li Guoyan\*

School of Computer and Information Engineering, Tianjin Chengjian University, Tianjin 300384, China

**Abstract** Aiming at the problems of ghost and the noise interference from dynamic background in classic visual background extraction algorithm, an improved visual background extraction algorithm is proposed. The important feature information of pixels in complex environment can be collected by creating the auxiliary sample set. By introducing analysis of the pixel ghost factor and the region complexity, the matching threshold and updating rate of each pixel can be adaptively adjusted. With the pixel flicker analysis based on sliding window, the points which may be misdetected as foreground can be updated to the auxiliary samples according to probability. The comparative experiments in the multi-scene show that the proposed method can reduce the wrong classifications rate to as low as 1.49%, eliminate the ghost quickly, suppress the noise interference from the dynamic background, and ensure the complete recognition of foreground target. The results of the algorithm are more accurate in the complex environment.

**Key words** image processing; moving object detection; visual background extraction algorithm; ghost elimination; adaptive threshold; dynamic background

**OCIS codes** 100.2000; 100.2960; 100.3008; 150.1135

## 1 引言

运动目标检测技术是计算机视觉和数字图像处理领域的重要研究内容, 近年来国内外学者对该问题进行了大量研究。常见的运动目标检测方法主要有帧差法<sup>[1]</sup>、光流法<sup>[2]</sup>、背景剪除法<sup>[3]</sup>等。其中背景剪除法应用最为广泛, 其特点是通过提取视频中相应数据对视频场景中的背景进行建模。最具代表性

的背景剪除算法有高斯混合模型(GMM)<sup>[4-6]</sup>、视觉背景提取(Vibe)算法<sup>[7-11]</sup>等。Barnich 等<sup>[12]</sup>提出基于像素的非参数化随机样本模型, 即 Vibe 算法, 它通过收集背景样本来构建背景模型, 并将待检测序列中某像素点的特征与其背景模型进行匹配, 通过阈值判断对该点进行背景前景划分, 如果该点被划分为背景, 则将其特征值向该点及其邻域点样本集中, 以一定概率随机更新。该算法具有计算量小、处

收稿日期: 2018-06-04; 修回日期: 2018-07-16; 录用日期: 2018-07-24

基金项目: 国家 863 计划(2015BAF09B02-3)、天津市自然科学基金(17JJCQNJC00500)

\* E-mail: lgy2351076@163.com

理速度快以及应对噪声稳定性好等特点,非常适合嵌入中小型视频设备中。

Vibe 算法在复杂环境中检测运动目标时仍存在一些问題,比如:算法初始化时存在前景目标或静止目标突然运动而出现鬼影的现象;复杂环境中的动态背景因素会使得相应像素点值在时域中频繁变化,导致动态背景被误检为前景,使结果中出现大量噪声。

针对这些问題,本文综合考虑算法在复杂环境下的准确性及有效性,在 Vibe 算法基础上提出通过分析像素点的鬼影程度,动态调整像素点在空间中的传播速率,提高鬼影消除速度;通过分析像素点的区域复杂度信息自适应调整匹配阈值;并根据像素点的闪烁程度更新辅助样本集合,在有效去除动态背景干扰的同时,保证真正前景能被准确检测。

## 2 Vibe 算法描述

### 2.1 采样与匹配

Vibe 算法初始化时为每个像素点创建一个背景样本集合。某像素点位置一定范围内的所有像素点均依概率收集在其背景样本集合中。在检测过程中,将当前视频序列中的像素点与该点位置的背景样本集进行匹配,通过样本集匹配程度来判断该点是否属于背景。设  $M(x)$  为某像素点  $x$  的背景样本集合,则该集合定义为

$$M(x) = \{v_1, v_2, \dots, v_{N-1}, v_N\}, \quad (1)$$

式中  $v_i$  表示对  $x$  一定邻域范围内的像素点概率采样后生成的背景样本集中索引为  $i$  的元素,  $N$  为样本集大小。

在匹配时,  $S_r[v(x)]$  是以  $v(x)$  为球心、 $r$  为半径的球形区域,如果在三维颜色空间中球形区域  $S_r[v(x)]$  与背景样本集合  $M(x)$  的交集满足一定要求,则将像素点  $v(x)$  划分为背景像素,即

$$\{S_r[v(x)] \cap \{v_1, v_2, \dots, v_N\}\} \in R_{\#}, \quad (2)$$

式中  $R_{\#}$  为像素信息在三维颜色空间中匹配时满足欧氏距离阈值的点的集合。算法通过衡量该集合中元素个数来对像素点进行匹配。匹配过程如图 1 所示。

### 2.2 背景模型初始化

Vibe 算法在初始化时为视频序列初始帧中的每个像素点创建其背景样本集合。由于单个像素点包含信息较少,假设像素点的相邻像素点均具有相似的特性分布,Vibe 算法对某像素点所在区域内的所有像素点按一定概率进行采样,并构建该像素点

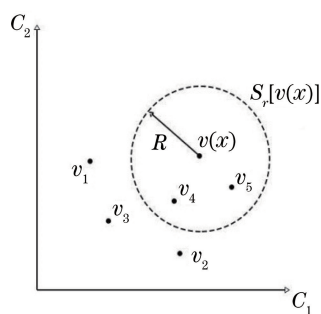


图 1 三维颜色空间匹配示意图

Fig. 1 Matching diagram of three-dimensional color space

的背景样本集合。初始化公式为

$$M^0(x) = \{v^0[y | y \in N_G(x)]\}, \quad (3)$$

式中  $M^0(x)$  为像素点  $x$  的初始化背景样本集合,  $v^0(y)$  为像素点  $x$  在初始化时背景样本集中的采样元素,  $y$  为在像素点  $x$  所在区域内随机选取的一个像素点,  $N_G(x)$  为与像素点  $x$  相邻近且具有相似特征的像素点的集合。

### 2.3 背景模型的更新策略

Vibe 算法采用随机更新的策略,只有在某像素点被划分为背景像素时对其样本集进行更新,时间采样因子为  $T$ ,即被划分为背景的像素点有  $1/T$  的概率代替该点背景样本集中的某一元素。同时,该像素点有一定概率向其相邻像素点样本集合中传播,以保证视频图像序列中像素的空间一致性。

## 3 改进的 Vibe 算法

### 3.1 辅助样本集

在传统 Vibe 算法中,样本集合初始化时对像素点及其邻域的采样并不能满足复杂环境下的检测要求。在复杂场景中,部分像素点的特征信息变化频率较快,不能被背景样本集合及时收集,容易产生误检。

本文提出为每个像素点创建一个辅助样本集合,在一定条件下对可能被误检为前景的像素点进行更新,将其特征信息更新到辅助样本集合中。辅助样本集合保证像素点在特殊情况下的重要特征信息能被有效收集,便于减少动态背景因素产生的噪声误检。

### 3.2 消除鬼影

在视频序列的初始化过程中,由于 Vibe 算法仅采用第一帧图像构建背景模型,因此初始化时如果存在前景目标或者长期静止的目标突然运动,会导致真实背景像素因无法有效匹配而被检测为前景,产生鬼影。Vibe 算法在背景更新时通过空间传播的方法使静止的前景目标逐渐融入到背景中,但该

过程耗时较长。

文献[10]提出利用固定较大的更新参数快速消除鬼影,但运动速度较慢的前景目标容易被同化到背景中,且对于不同场景需要给出不同的先验参数,难以满足多场景环境下的检测要求。由于真实前景区域的像素特征与其邻域背景区域像素特征差距较大,背景像素的信息传播到前景边界像素的样本集合后,该信息很难与前景目标成功匹配,即邻域背景信息很难穿透边界进入到前景目标区域中去。相反,由于背景像素通常具有区域相似的特性,鬼影区域的像素值与其邻域背景区域的像素值差别较小。

在像素点空间传播时进行二次匹配,衡量前景像素点的鬼影程度,并调整疑似鬼影区域的更新速率,实现快速消除鬼影。设背景像素点  $x_1$  的像素值为  $v(x_1)$ ,在背景更新过程中,如果该点的邻域中存在被判断为前景的像素点  $x_2$ ,则将点  $x_1$  的像素值按一定概率向点  $x_2$  的样本集中进行空间传播。传播概率与该背景像素值  $v(x_1)$  和邻域前景点像素值  $v(x_2)$  的匹配程度有关,匹配公式为

$$P_{\text{ghost}}(x_2) = \frac{\lambda \cdot R(x_2)}{\text{dist}[v(x_1), v(x_2)]}, \quad (4)$$

式中  $R(x_2)$  为邻域前景像素点  $x_2$  的当前匹配阈值,  $\lambda$  为鬼影参数。  $P_{\text{ghost}}(x_2)$  为像素点  $x_2$  的鬼影因子,表示该点作为鬼影点的可能性。当某前景点的鬼影因子取值较大时,算法可以提高其邻域背景像素向当前点的空间传播速率,实现快速消除鬼影。

### 3.3 结合区域信息的复杂度分析

文献[7]提出通过像素背景复杂度自适应调节匹配阈值  $R$ ,复杂度参数由统计单个像素信息所得。本文提出一种区域复杂度分析策略。在检测过程中,视频序列中各区域的信息复杂程度不同,部分区域长期显示比较单一的背景信息,部分区域环境信息较为复杂、前景背景切换速度较快。本文算法通过准确衡量当前区域的显著性及环境复杂程度,实现对匹配阈值的自适应动态更新。

通过水平方向与垂直方向上的 Sobel 梯度算子对像素点区域进行边缘检测分析,其梯度特征模板为

$$G_v = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, G_h = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}. \quad (5)$$

使用横向梯度模板算子  $G_v$  与纵向梯度模板算子  $G_h$  分别对某像素点  $x_i$  所在区域进行卷积,得到

结果  $G_v(x_i)$ 、 $G_h(x_i)$ 。该区域梯度参数  $G(x_i)$  的计算公式为

$$G(x_i) = \sqrt{G_v^2(x_i) + G_h^2(x_i)}. \quad (6)$$

通过收集像素点的邻域颜色差值对其区域颜色显著性进行分析,定义像素点  $x_i$  在以自身为中心的某一区域内与该区域内其他像素点颜色的差值总和  $C(x_i)$  为该点区域颜色显著程度为

$$C(x_i) = \frac{1}{N_c} \sum_{x_n \in N_G(x_i)} \sqrt{[v(x_i) - v(x_n)]^2}, \quad (7)$$

式中  $N_c$  为  $x_i$  的邻域像素个数,  $v(x_i)$  为该像素点的颜色特征,包含该点在当前颜色空间中的全部颜色信息。

区域复杂度特征因子  $S(x_i)$  表示为

$$S(x_i) = \alpha G(x_i) + \beta C(x_i), \quad (8)$$

式中参数  $\alpha$ 、 $\beta$  为显著性权重。通过区域复杂度分析对模型的匹配阈值进行自适应动态更新,公式为

$$R(x_i) = \begin{cases} R(x_i) \times (1 - \delta), & R(x_i) > S(x_i) \times R_{\text{scale}} \\ R(x_i) \times (1 + \delta), & \text{else} \end{cases}, \quad (9)$$

式中  $R(x_i)$  为像素点  $x_i$  的匹配阈值,  $\delta$  为单次更新变化幅度参数,  $R_{\text{scale}}$  为调整复杂度的尺度因子。

像素点区域复杂度越高,则该点作为运动目标外边界点或背景物体边缘点的可能性越高,此时适当提高匹配阈值,防止背景因素被误检为前景。像素点区域复杂度越低,该点所在区域背景信息分布越稳定,应适当减小匹配阈值,提高检测精度。

### 3.4 滑动窗模型

滑动窗模型在很多领域都得到了应用,常用于图像、信号等的动态分析处理。该模型通过对某元素在时域中的某一有限区域进行采样,来描述该元素一定时间段内的变化特性。在视频图像处理中,使用滑动窗模型进行采样可有效提高处理结果的实时性。

在某视频序列中,设  $\{B(i)\}_{i=1, \dots, t}$  为从初始化到  $t$  时刻的所有视频序列。在运动目标检测过程中,由于视频环境不断变化,视频某帧在  $t$  时刻的变化特性只与距它最近的  $N_L$  帧视频序列实时相关性较高,滑动窗模型采样即只对某序列  $t$  时刻之前相邻的  $N_L$  帧序列进行采样。在对帧数较多、变化较快的视频序列处理中,可通过调节滑动窗长度  $N_L$  控制采样序列的历史相关性。 $t$  时刻基于滑动窗的视频序列模型如图 2 所示。

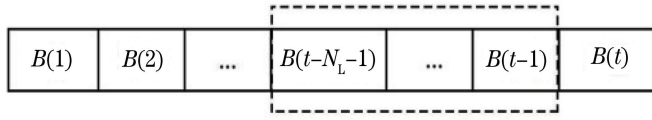


图2 基于滑动窗的背景序列示意图

Fig. 2 Diagram of sequence based on sliding window

### 3.5 像素点闪烁因子

在水波纹、树叶抖动等背景较为复杂的场景中,动态背景区域像素值变化频率较快,因而传统 Vibe 算法在检测时产生大量误检区域。文献[9]提出通过分析像素点在时域中背景与前景之间的切换程度来判断该像素点属于动态背景的可能性。但是当有一个像素点具有较高的闪烁参数值时,不一定表示其属于动态背景。由于闪烁程度是累计计算模式,因此只能大致反映该像素从初始化到当前帧的总体闪烁趋势,且闪烁程度值会因连续前景或背景序列而削减。

本文引入滑动窗模型,统计视频序列中包括当前帧  $t$  在内的  $N_L$  帧最近连续序列,实时反映当前像素点的实时闪烁情况,计算公式为

$$F_t(x_i) =$$

$$\begin{cases} 1, & \text{if } x_i \text{ is divided into the foreground at } t \text{ time} \\ 0, & \text{if } x_i \text{ is divided into the background at } t \text{ time} \end{cases}, \quad (10)$$

$$f_t^{N_L}(x_i) = F_{t-N_L+1}(x_i) + F_{t-N_L+2}(x_i) + \dots + F_t(x_i), \quad (11)$$

式中  $F_t(x_i)$  表示像素  $x_i$  在  $t$  时刻是否属于前景,  $f_t^{N_L}(x_i)$  为像素  $x_i$  通过滑动窗区间取值后的闪烁因子。由滑动短时记忆的特性,像素点最近区间内的闪烁程度值能更好地反映出该像素点可能是被误检为前景的噪声点的概率。闪烁因子越大,则该点作为被误检的动态噪声的可能性越大。本文算法通过一定概率将闪烁因子较大的前景点像素值更新到样本辅助集合中,概率计算公式为

$$P_{\text{flicker}}(x_i) = \begin{cases} \frac{\omega f_t^{N_L}(x_i)}{N_L}, & f_t^{N_L}(x_i) \geq \theta \\ 0, & f_t^{N_L}(x_i) < \theta \end{cases}, \quad (12)$$

式中  $P_{\text{flicker}}(x_i)$  表示某时刻像素点  $x_i$  若被检测为前景则将其更新到辅助样本集中的更新率,  $\theta$  为阈值,  $\omega$  为尺度参数。本文算法通过对闪烁区域的实时分析,快速准确地消除复杂环境下动态背景产生的噪声。

## 4 实验结果

仿真实验硬件平台的 CPU 为 Intel Core i5, 内

存为 8 G, 操作系统为 WIN7 旗舰版, 运行环境为 VS2013。实验数据来源于 highway、boats、overpass、canoe、pedestrians 等视频中的图像序列, 共计 14417 帧。将改进 Vibe 算法、GMM 算法和传统 Vibe 算法<sup>[11]</sup>进行对比, 并对其中部分有真值的视频序列进行定量分析。实验部分参数取值参考文献[9-12], 背景样本集大小  $N=20$ , 初始化距离阈值  $R=20$ , 匹配个数阈值为 2, 初始时间采样因子  $T=16$ 。根据经验给出公式中参数值用于仿真, 选取辅助样本集合大小为 7, 鬼影参数  $\lambda=0.4$ , 区域颜色采样范围为 24 邻域, 显著性权重  $\alpha=0.3, \beta=0.7$ , 单次变化幅度参数  $\delta=0.05$ , 复杂度尺度因子  $R_{\text{scale}}=5$ , 滑动窗口长度  $N_L=100$ , 闪烁阈值  $\theta=22$ , 尺度参数  $\omega=0.35$ 。图 3 为多种算法在视频序列中的检测结果对比。

实验在视频 pedestrians 中以第 330 帧为起始帧开始检测。初始化帧中存在运动目标是产生鬼影的典型场景, 起始帧如图 3(a) 所示, 目标移动后背景信息填入图中行人原始位置, 相应像素点值变化导致算法在该位置检测出鬼影。图 3(b) 和 (c) 分别对应视频序列中第 340 帧和第 430 帧图像, 图 3(d)~(i) 为多种算法的检测结果。实验结果中鬼影像素数目随帧数变化如图 4 所示, 对比传统 Vibe 算法, 改进算法的鬼影点数目随帧数增长而快速下降。实验结果表明, 本文算法能在较短时间内快速消除鬼影。

图 5(a) 为视频序列 overpass 中第 2705 帧, 图 5(b)~(d) 为多种算法检测实验结果。该视频背景中包含大量晃动的树枝作为动态因素, 是具有代表性的复杂场景下的视频序列。由图 5(b)~(d) 可知, 在复杂环境下, GMM 算法更新速率较高, 虽然能够抑制部分动态背景产生的噪声, 但出现的行人目标被快速更新到背景中, 前景误检和背景误检现象均较为严重。传统 Vibe 算法能较为有效地识别前景目标, 但动态背景大量被误检为前景目标, 产生噪声较多。改进 Vibe 算法通过像素闪烁属性分析, 使动态背景被误检为前景现象的发生次数明显减少, 且在动态的背景区域中出现的前景目标也能被准确识别。

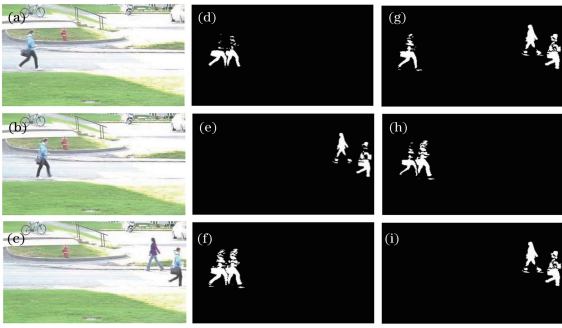


图3 pedestrians 视频序列测试结果比较。(a)起始帧; (b)第340帧;(c)第430帧;(d)第340帧 GMM 算法检测结果;(e)第430帧 GMM 算法检测结果;(f)第340帧 Vibe 算法检测结果;(g)第430帧 Vibe 算法检测结果;(h)第340帧本文算法检测结果;(i)第430帧本文算法检测结果

Fig. 3 Comparison of test results in pedestrians video sequence. (a) Initialization frame; (b) input 340th frame; (c) input 430th frame; (d) GMM result of 340th frame; (e) GMM result of 430th frame; (f) Vibe method result of 340th frame; (g) Vibe method result of 430th frame; (h) proposed method result of 340th frame; (i) proposed method result of 430th frame

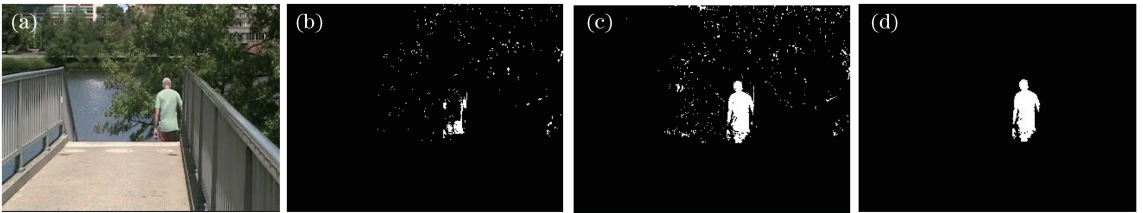


图5 第2705帧 overpass 视频序列测试结果比较。

(a)输入帧;(b) GMM 算法检测结果;(c) Vibe 算法检测结果;(d)本文算法检测结果

Fig. 5 Comparison of test results in overpass video sequence of 2705th frame.

(a) Input frame; (b) GMM result; (c) Vibe method result; (d) proposed method result

表1 算法性能评估指标

Table 1 Comparison of algorithm capability

Performance parameter	GMM	Vibe method	Proposed method
Recall	0.4847	0.6671	0.6557
Specificity	0.9851	0.9671	0.9937
FPR	0.0158	0.0317	0.0087
FNR	0.342	0.1471	0.1624
Precision	0.7390	0.6416	0.8122
PWC / %	2.7982	3.1222	1.4867
FPS / (frame·s <sup>-1</sup> )	19.67	50.12	38.33

GMM 算法通过比较松弛的背景更新策略提升其环境适应能力,降低虚警率,但导致过快将前景目标更新为背景,漏检率大幅升高。传统 Vibe 算法能

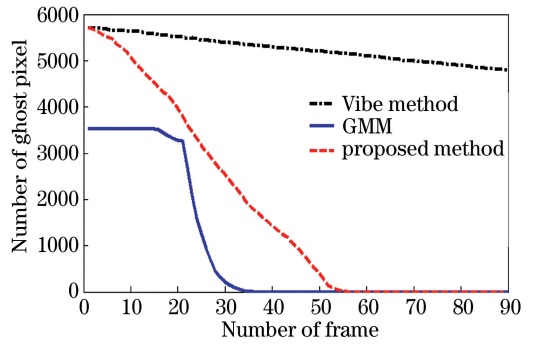


图4 鬼影像素数目随帧数变化

Fig. 4 Number of ghost pixel versus number of frame

为了较为精确地定量分析几种算法的性能,文献[13]中定义多个性能评估指标,即召回率、特异率、虚警率(FPR)、漏检率(FNR)、精度、错分率(PWC)。实验对上述视频序列中部分有真值的序列共计10517帧进行测试与计算,将结果进行平均处理,并通过帧率(FPS)描述算法运行的时间效率,得到数据如表1所示。表中召回率、特异率、精度、帧率指数越高表示算法性能越好,FPR、FNR、PWC指数越低表示算法性能相对越好,PWC指数为较为综合性的评价指标。

较为有效地检测前景目标,但由于其固定的匹配阈值与更新速率导致复杂环境中的动态背景大量被检测为前景,且算法运行中出现鬼影后难以将其消除。本文算法的PWC指数、特异率、虚警率、精度指标均相对最优。对比GMM算法和传统Vibe算法,本文算法的精度分别提高9.9%和26.59%,虚警率分别降低44.9%和72.55%,在综合性能上错分率分别下降46.87%和52.38%。由于改进算法复杂度有所增加,本文算法平均帧率较传统Vibe算法有所下降,但明显优于GMM算法,且平均帧率保持在30 frame/s以上,具有一定实时性<sup>[3]</sup>。本文算法综合检测性能相对最优。

## 5 结 论

提出为像素点创建辅助样本集合,保证像素点重要特征信息不被丢失,并引入区域复杂度信息,结合时域滑动闪烁因子,动态调整匹配阈值与鬼影区域更新速率。实验结果表明,改进算法能快速消除鬼影和复杂背景中的动态噪声,并保证前景目标能被有效检测,提高复杂环境下运动目标检测结果的精确性。算法具有实时性好、抗干扰能力强、检测结果较为准确等优点。

## 参 考 文 献

- [1] Lu G Q. A combined frame difference with a background subtraction algorithm of moving object detection [C] // International Workshop on Communication Technology 2013, International Conference on Consumer Electronics, Communications and Networks, 2013.
- [2] Ding Q, Gu G H, Xu F Y, *et al.* Moving target detection on moving camera with the presence of strong parallax [J]. *Laser & Optoelectronics Progress*, 2015, 52(9): 091501.  
丁祺, 顾国华, 徐富元, 等. 强视差下的移动相机运动目标检测[J]. *激光与光电子学进展*, 2015, 52(9): 091501.
- [3] Xu H W, Chen Q, Qian W X. Target detection algorithm based on improved single Gaussian background model [J]. *Laser & Optoelectronics Progress*, 2016, 53(4): 040401.  
徐鸿伟, 陈钱, 钱惟贤. 基于改进的单高斯背景模型检测算法的研究[J]. *激光与光电子学进展*, 2016, 53(4): 040401.
- [4] Wang X M, Zhao D Q, Sun G M, *et al.* Target detection algorithm based on improved Gaussian mixture model [C] // Proceedings of the 2015 International Conference on Electrical, Computer Engineering and Electronics, 2015.
- [5] Zhou J Y, Wu X P, Zhang C, *et al.* A moving object detection method based on sliding window Gaussian mixture model [J]. *Journal of Electronics & Information Technology*, 2013, 35(7): 1650-1656.  
周建英, 吴小培, 张超, 等. 基于滑动窗的混合高斯模型运动目标检测方法[J]. *电子与信息学报*, 2013, 35(7): 1650-1656.
- [6] Godbehere A B, Matsukawa A, Goldberg K. Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation [C] // 2012 American Control Conference (ACC), 2012: 4305-4312.
- [7] Zhang W Y, Xu H Z, Luo J. Moving objects detection under complex background based on ViBe [J]. *Computer Science*, 2017, 44(9): 304-307.  
张文雅, 徐华中, 罗杰. 基于 ViBe 的复杂背景下的运动目标检测[J]. *计算机科学*, 2017, 44(9): 304-307.
- [8] Xu J Q, Jiang P P, Zhu H B, *et al.* An improved ViBe algorithm for moving object detection [J]. *Journal of Northeastern University (Natural Science)*, 2015, 36(9): 1227-1231.  
徐久强, 江萍萍, 朱宏博, 等. 面向运动目标检测的 ViBe 算法改进[J]. *东北大学学报(自然科学版)*, 2015, 36(9): 1227-1231.
- [9] Mo S W, Deng X P, Wang S, *et al.* Moving object detection algorithm based on improved visual background extractor [J]. *Acta Optica Sinica*, 2016, 36(6): 0615001.  
莫邵文, 邓新蒲, 王帅, 等. 基于改进视觉背景提取的运动目标检测算法[J]. *光学学报*, 2016, 36(6): 0615001.
- [10] Xie S R, Ye S B, Yang B H, *et al.* Moving targets detection based on an improved YUV\_Vibe fusion algorithm [J]. *Laser & Optoelectronics Progress*, 2018, 55(11): 111002.  
谢申汝, 叶生波, 杨宝华, 等. 基于改进的 YUV\_Vibe 融合算法的运动目标检测[J]. *激光与光电子学进展*, 2018, 55(11): 111002.
- [11] van Droogenbroeck M, Paquot O. Background subtraction: experiments and improvements for ViBe [C] // IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2012: 32-37.
- [12] 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.
- [13] Goyette N, Jodoin P M, Porikli F, *et al.* Changedetection. net: a new change detection benchmark dataset [C] // IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2012: 1-8.