

## 基于脉冲图像传感器的高精度高速目标追踪

孙硕<sup>1,2</sup>, 徐江涛<sup>1,2\*</sup>, 高志远<sup>1,2</sup><sup>1</sup>天津大学微电子学院, 天津 300072;<sup>2</sup>天津市成像与感知微电子技术重点实验室, 天津 300072

**摘要** 针对脉冲图像传感器, 提出一种高精度高稳定性的高速目标追踪算法。首先, 介绍脉冲图像传感器的原理; 其次, 结合传感器脉冲密度特性改进传统视觉背景提取(Vibe)算法, 去除传统 Vibe 算法中存在的鬼影和空洞问题, 提高运动检测的完整性; 然后, 结合运动检测, 对传统均值漂移(MS)追踪算法进行改进, 提高目标追踪的精度和稳定性; 最后, 通过图像重构完成场景再现与目标追踪。在 3 个高速场景的实验中: 与直接应用于图像序列的传统 MS 算法相比, 所提算法对高速目标的最大追踪误差分别从 11.0454 降低至 2.2361, 从 14.1421 降低至 5.0000, 从 26.1725 降低至 5.0990; 目标追踪的位置标准差从 7.9879 降低至 2.0393, 从 12.0790 降低至 2.7454, 从 14.4591 降低至 3.5654。综上所述, 所提算法能够有效提高目标的追踪精度和追踪稳定性, 能更好地适用于脉冲图像传感器。

**关键词** 传感器; 脉冲图像传感器; 运动检测; 鬼影去除; 高速目标追踪; 图像重构; 脉冲密度

中图分类号 TP212.9

文献标志码 A

DOI: 10.3788/LOP213286

## Pulse Image Sensor-Based High-Precision and High-Speed Target Tracking

Sun Shuo<sup>1,2</sup>, Xu Jiangtao<sup>1,2\*</sup>, Gao Zhiyuan<sup>1,2</sup><sup>1</sup>School of Microelectronics, Tianjin University, Tianjin 300072, China;<sup>2</sup>Tianjin Key Laboratory of Imaging and Perception Microelectronics Technology, Tianjin 300072, China

**Abstract** A high-speed target tracking algorithm with high accuracy and stability is proposed for pulse image sensors. First, the principle of a pulse image sensor is introduced. Second, the traditional visual background extraction (Vibe) algorithm is improved by combining the pulse density characteristics of the sensor to remove the ghost and hole issues in the traditional Vibe algorithm, this further improves the integrity of motion detection. Subsequently, combined with motion detection, the traditional mean shift (MS) tracking algorithm is enhanced to improve the accuracy and stability of target tracking. Finally, scene reconstruction and target tracking are completed via image reconstruction. In the three high-speed scenes experiments, compared with the traditional MS algorithm, which is directly applied to image sequences, the maximum tracking error of the proposed algorithm for high-speed targets reduced from 11.0454 to 2.2361, from 14.1421 to 5.0000, and from 26.1725 to 5.0990, respectively. The position standard deviation of target tracking decreased from 7.9879 to 2.0393, from 12.0790 to 2.7454, and from 14.4591 to 3.5654, respectively. In summary, the proposed algorithm can effectively improve target tracking accuracy and stability and is more suitable for pulse image sensors than the other algorithms.

**Key words** sensors; pulse image sensor; motion detection; ghost removal; high-speed target tracking; image reconstruction; pulse density

## 1 引言

在目标追踪系统中, 成像系统和分析算法是相互依赖、密不可分的<sup>[1]</sup>。成像系统负责接收场景中的大量信息, 然后通过复杂的分析算法和图像理解技术处

理和利用这些信息<sup>[2]</sup>。图像传感器作为成像系统, 尤其是面对高速场景时, 需要加快视觉信息的接收速度<sup>[3]</sup>, 这意味着图像传感器必须具有更高的帧频<sup>[4]</sup>。然而, 传统的图像传感器在高速场景中存在一定的局限性, 例如, 较低的帧频会导致高速运动物体的图像模

收稿日期: 2021-12-20; 修回日期: 2022-01-06; 录用日期: 2022-01-17; 网络首发日期: 2022-01-27

基金项目: 国家自然科学基金(62134004)

通信作者: \*xujiantao@tju.edu.cn

糊。虽然一些高帧频的图像传感器可以缓解高速场景中的限制,但图像传感器的数据吞吐量也会呈指数增长。因此,最近提出一种基于仿生的脉冲型图像传感器<sup>[5-6]</sup>,它可以在提高帧频的同时降低数据吞吐量,其输出数据为单比特 0 或 1,帧频最高可达 40000 frame/s。由于拥有较少的数据吞吐量和较高的帧频,脉冲图像传感器可以在压缩数据量的同时更清晰地捕获高速目标。

目标追踪算法作为分析算法,在过去的几十年中得到极大的改进,其可靠性和准确性令人满意,如均值漂移(MS)算法<sup>[7-10]</sup>、卡尔曼滤波算法<sup>[11-13]</sup>、粒子滤波算法<sup>[14-15]</sup>、核相关滤波算法<sup>[16]</sup>等。大多数追踪算法可以归纳为以下 4 种:基于对比度分析的追踪、基于匹配的追踪、基于团块的追踪以及基于核函数的追踪<sup>[17]</sup>。其中,大多数都是针对传统相机所拍摄的视频图像序列的研究,而针对脉冲图像传感器数据序列的追踪算法的研究较少。一方面是由于脉冲图像传感器数据为单比特数据,没有直接包含灰度信息,不适合传统的目标追踪算法;另一方面,为了应用传统的目标追踪算法,需要根据单比特数据来重构图像序列。尽管一些研究基于脉冲图像传感器提出一些相关的图像重构算法<sup>[18-21]</sup>,这些算法可以根据单比特的脉冲数据重构出场景的灰度信息,然而如果将传统的目标追踪算法应用于重构后的图像序列,那么图像重构过程中引入的误差和噪声将会严重影响目标追踪的精度和稳定性。

因此,本文提出一种基于脉冲图像传感器的高速目标追踪算法。所提算法不同于传统的以视频图像序列为基础的目标追踪算法,它致力于改进传统的 MS 算法,使其能够应用于脉冲图像传感器数据序列。首先,针对原始的脉冲数据,利用脉冲图像传感器的脉冲密度改进视觉背景提取(Vibe)算法,去除传统算法中存在的鬼影和空洞问题。然后,结合 Vibe 检测结果对

传统的 MS 算法进行改进,提高对场景中的高速目标追踪的精度和稳定性。最后,通过图像重构完成场景再现与目标追踪。

## 2 脉冲图像传感器成像系统及像素原理

脉冲图像传感器成像系统如图 1 所示,主要由传感器模块、柔性电路板(FPC)排线、工业镜头和数据采集器构成,而传感器模块又由脉冲图像传感器芯片、供电板组成。其中,脉冲图像传感器芯片为作者实验室自主设计研发,并通过 0.11  $\mu\text{m}$  工艺进行流片,芯片大小为 9960  $\mu\text{m} \times 7100 \mu\text{m}$ 。芯片整体架构主要包括时序控制、像素阵列、参考电压、串行外设接口(SPI)、列级缓存器和低压差分信号输出(LVDS)。其中:像素阵列为 250 $\times$ 400,是传感器的感光区域,负责将场景的光强信息转化为脉冲信号;时序控制主要为像素阵列提供相应的工作时序以及读出信号,保证图像传感器数据的帧扫描输出;参考电压则为传感器提供相应的偏置电压;SPI 主要负责外界设备与传感器的通信;列级缓存器主要对像素阵列输出的脉冲信号进行缓存,等待读出信号到来后进行输出;LVDS 接口负责通过并串转换将列级缓存器中的脉冲信号输出到芯片外。图像传感器采用帧扫描的方式输出脉冲数据,每次输出一行,时间为 0.1  $\mu\text{s}$ ,故输出一帧需要 25  $\mu\text{s}$ ,因此脉冲图像传感器的帧频最高可达 40000 frame/s。供电板主要为脉冲图像传感器芯片提供相应的参考电压与电源电压,相机镜头采用 8 mm 焦距的日本康标达高清工业相机镜头 M0814-MP2。数据采集器采用度信科技图像传感器摄像模组专用测试盒 ULM928,采集器通过 USB3.0 与 PC 端的上位机之间进行数据传输。传感器模块与数据采集器之间采用 FPC 排线进行数据传输。

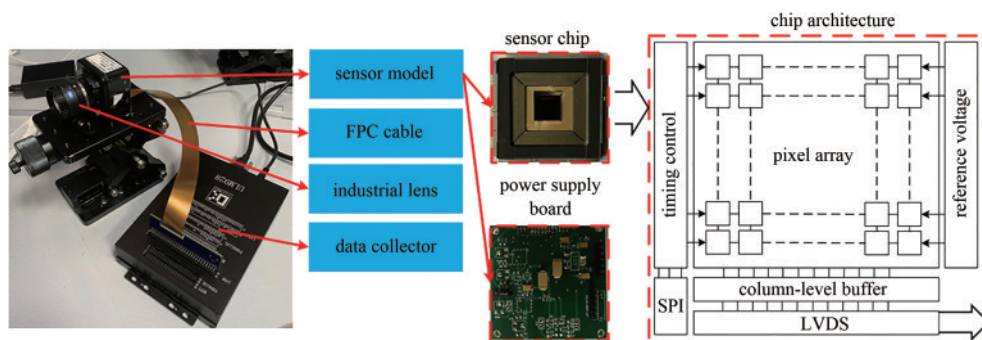


图 1 脉冲图像传感器成像系统

Fig. 1 Imaging system of pulse image sensor

像素单元的模式和工作原理如图 2 和图 3 所示,像素单元主要由感光像素、积分器、复位信号、比较器、读出开关构成。感光像素受到光照时会不断积分光生电流  $I_D$ ,产生积分电荷  $Q_D$ ,当积分电荷  $Q_D$  达到设定的参考电荷  $Q_{ref}$  后使得比较器产生一个脉冲数据 1,同时产生一个复位信号使积分器复位并重新开始积分光生电

流。而产生的脉冲数据 1 在每一帧的读出开关关闭时才会被输出,如果在一帧时间内该像素没有产生脉冲数据 1,则在读出开关关闭时会输出一个脉冲数据 0,且读出开关关闭的时间周期为一帧时间。由此可知,脉冲图像传感器采用的是异步复位同步读出机制,即复位信号到来的时刻与读出开关关闭的时刻并不一

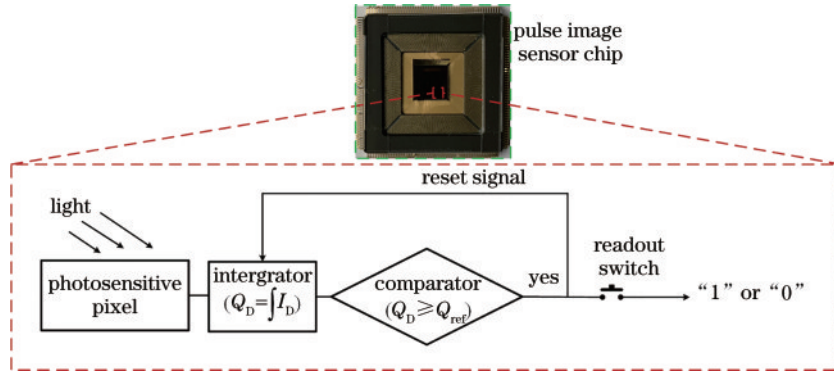


图 2 脉冲图像传感器像素模型

Fig. 2 Pixel model of pulse image sensor

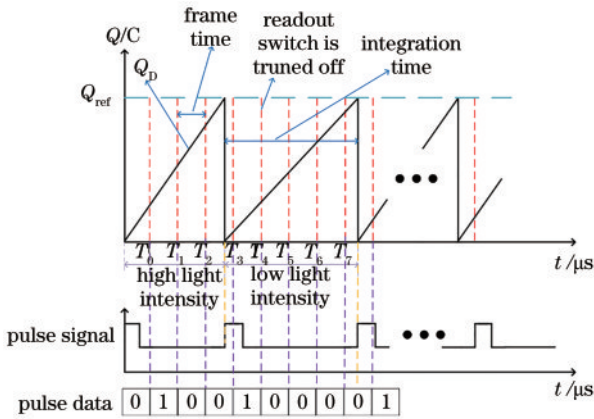


图 3 脉冲图像传感器像素的工作原理

Fig. 3 Working principle of pulse image sensor pixel

致,并且脉冲图像传感器每一帧的输出只有 0 或 1 的单比特数据,如图 4 所示。单比特数据虽然极大减少了数据冗余、缓解了数据接收的负担,但无法表征图像的灰度信息,因此必须根据单比特的数据特性来恢复图像的灰度信息。

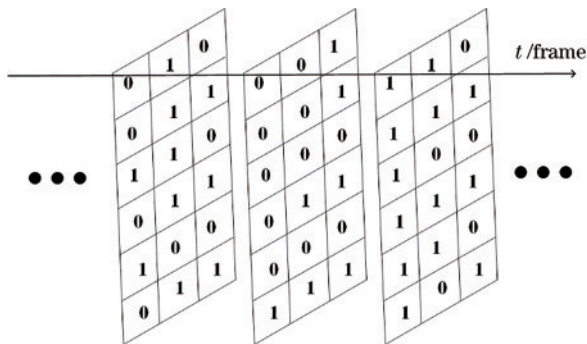


图 4 脉冲图像传感器的输出数据类型

Fig. 4 Output data type of pulse image sensor

根据图 2 和图 3 可知,在理想条件下,像素的实际积分时间  $t_m$  与光电流  $I_D$  的关系可表示为

$$\begin{cases} \int_{t_0}^{t_0+t_m} I_D dt = Q_{ref} \\ I_D = k_L \times L \end{cases} \quad (1)$$

式中: $L$ 为光强大小; $k_L$ 为光电转换系数。光电流  $I_D$  与光强正相关,光强越强, $I_D$  越大,实际积分时间  $t_m$  就会越小;反之,光强越弱, $I_D$  越小,实际积分时间  $t_m$  就会越大。由于脉冲数据是按传统的帧扫描方式输出的,那么每个像素的相邻两个输出脉冲数据 1 之间的时间间隔(简称为脉冲间隔)一定是帧时间的整倍数。而又由于复位信号到来的时刻与读出开关关闭的时刻并不一致,那么实际的积分时间  $t_m$  与脉冲间隔之间可能会存在时间误差,且该时间误差最大不会超过一帧时间,故可近似为

$$\begin{cases} P_t = N \times T \\ t_m \approx P_t \end{cases} \quad (2)$$

式中: $P_t$ 是像素在  $t$  时刻的脉冲间隔; $T$ 是帧时间; $N$ 是帧数。根据式(1)和式(2), $P_t$ 可以表示为

$$P_t \approx Q_{ref} / (k_L \times \bar{L}), \quad (3)$$

式中: $\bar{L}$ 是平均光强。式(3)表明在传感器动态范围内,脉冲间隔与光强成反比。

### 3 基于 Vibe 运动检测算法的改进

Vibe 是一种常用的运动检测算法,该算法虽然具有计算速度快、操作简单等优点,但算法本身存在着较多的缺陷。当 Vibe 算法初始化的背景模型中存在着运动目标或者状态从静止转变为运动的目标时,背景模型的更新速率无法与背景变率不同步,就可能出现虚假检测,即出现鬼影和目标空洞。另一方面,脉冲图像传感器的单比特脉冲数据没有直接包含光强和颜色信息,也无法直接应用于 Vibe 算法,为此需要对脉冲数据进行一定的转换和处理。

根据前面的原理分析可知,脉冲间隔与光强成反比,在一定程度上反映光强的大小,为此需要将脉冲数据转换为脉冲间隔。以  $t$  时刻的像素  $(i, j)$  为例,首先需要找出该像素  $t$  时刻之前最近的一次脉冲数据 1 的输出时刻  $Z_{max}$  和  $t$  时刻之后最近的一次脉冲数据 1 的输出时刻  $Z_{min}$ ,相关计算公式为



$$\begin{cases} Z_{\max} = \max\{z | k_z(i, j) = 1, z < t\} \\ Z_{\min} = \min\{z | k_z(i, j) = 1, z \geq t\} \end{cases}, \quad (4)$$

式中:  $k_z(i, j)$  为像素  $(i, j)$  在  $z$  时刻的脉冲数据。该像素  $t$  时刻的原始脉冲间隔  $P_t(i, j)$  可以通过这两个时刻的差值获得, 计算公式为

$$P_t(i, j) = Z_{\min} - Z_{\max}. \quad (5)$$

图 5 描述了脉冲数据到脉冲间隔的转换过程。根据转换后的脉冲间隔, 可以求出脉冲频率  $f_t(i, j)$ , 即

$$f_t(i, j) = 1/P_t(i, j). \quad (6)$$

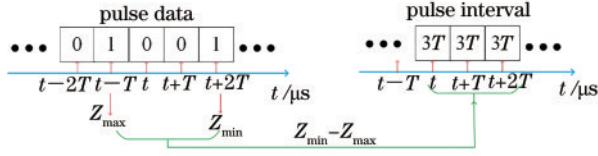


图 5 脉冲数据到脉冲间隔的转换

Fig. 5 Conversion from pulse data to pulse interval

脉冲频率的大小能够在一定程度上代表光强信息。从式(3)可以看出, 脉冲间隔与光强成反比, 那么式(6)中脉冲频率与光强成正比, 即光强越大, 脉冲频率越大。因此, 可以将脉冲频率作为像素的值来进行运动检测。

以像素  $(i, j)$  为例, 不同于传统的单帧建模, 这里使用  $A$  帧像素的值来初始化背景模型  $M_A(i, j)$ , 计算公式为

$$M_A(i, j) = \{f_1(i, j), f_2(i, j), \dots, f_A(i, j)\}. \quad (7)$$

然后计算当前时刻的像素值与背景模型之间的欧氏距离, 并将计算结果与设定的阈值  $R_D$  进行比较, 若低于阈值  $R_D$ , 则比较结果  $r_k$  为 1, 高于阈值  $R_D$  则为 0。然后统计比较结果, 如果统计结果大于给定的阈值  $R_B$ , 则该像素的检测结果  $B_t(i, j)$  为背景, 二值化灰度值为 0, 否则为运动目标, 二值化灰度值为 255, 即

$$\begin{cases} B_t(i, j) = \begin{cases} 0, & \sum_{k=1}^A r_k \geq R_B \\ 255, & \sum_{k=1}^A r_k < R_B \end{cases} \\ r_k = \begin{cases} 1, & |f_t(i, j) - f_k(i, j)| \leq R_D \\ 0, & |f_t(i, j) - f_k(i, j)| > R_D \end{cases} \end{cases}. \quad (8)$$

同时, 如果像素被检测为背景, 需要对背景模型进行更新, 即用其像素值随机替换该像素邻域内任意像素的背景模型中的任意样本, 计算公式为

$$\begin{cases} f_{\text{any}}(i, j) = f_t(i, j) \\ f_{\text{any}}(i_{\text{any}}, j_{\text{any}}) = f_t(i, j) \end{cases}, \quad (9)$$

式中:  $f_{\text{any}}(i, j)$  是像素  $(i, j)$  的背景模型中的任意样本;  $f_{\text{any}}(i_{\text{any}}, j_{\text{any}})$  是像素  $(i, j)$  邻域内任意像素的背景模型中的任意样本。

另外, 为了抑制 Vibe 算法检测过程中可能出现的

鬼影和内部空洞, 需要结合脉冲图像传感器的数据特点引入脉冲密度, 对传统 Vibe 算法进行改进。由于在高光强区域脉冲信号触发频率较高, 脉冲信号比较密集, 低光强区域脉冲信号触发频率较低, 脉冲信号比较稀疏, 因此也可以通过一段时间内脉冲信号的分布来反映场景的光强信息。图 6 为拍摄的高速转盘的脉冲信号, 白色像素点代表该段时间内有脉冲信号触发。从图 6 可以看出, 在不同光强的区域, 脉冲信号的分布是不同的。同理以像素  $(i, j)$  为例, 对像素  $(i, j)$  的邻域  $R_{ij}$  中固定时间段  $t_h$  内的脉冲信号进行统计, 并计算该邻域的脉冲密度  $d_t(i, j)$ , 计算公式为

$$\begin{cases} d_t(i, j) = \frac{\sum_{(x, y) \in R_{ij}} H_t(i, j)}{\sum_{(x, y) \in R_{ij}} 1} \\ H_t(x, y) = \begin{cases} 1, & Z_{\min} \in [t, t + t_h) \\ 0, & Z_{\min} \notin [t, t + t_h) \end{cases} \end{cases}, \quad (10)$$

式中:  $H_t(x, y)$  为邻域  $R_{ij}$  内的像素  $(x, y)$  的脉冲信号。由于鬼影处的光强与实际运动目标处的光强不同, 因此它们的脉冲密度也不相同。假设实际运动目标处的脉冲密度范围为  $R_H$ , 则鬼影处的二值化灰度值虽然为 255, 但其脉冲密度必然不在范围  $R_H$  内, 因此可以据此消除鬼影, 即

$$B_t(i, j) = 0, \text{ if } B_t(i, j) = 255 \text{ and } d_t(i, j) \notin R_H. \quad (11)$$

同理, 目标空洞处的二值化灰度值虽然为 0, 但其脉冲密度与实际目标处一致, 可以据此来补偿空洞, 即

$$B_t(i, j) = 255, \text{ if } B_t(i, j) = 0 \text{ and } d_t(i, j) \in R_H. \quad (12)$$

至此, 便完成了运动检测。

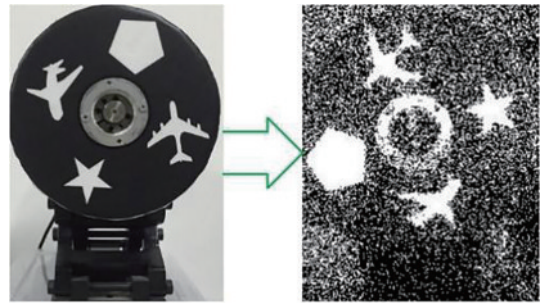


图 6 固定时间内转盘的脉冲信号分布

Fig. 6 Pulse signal distribution of turntable in fixed time

为了验证改进后 Vibe 检测算法的效果, 设立两组对比实验。图 7 和图 8 分别为对高速转盘图案和高速乒乓球的运动检测。图 7(b) 和图 8(b) 为传统 Vibe 算法检测结果, 传统 Vibe 算法虽然可以检测出运动目标的大致轮廓, 但也容易错检出许多鬼影和空洞。图 7(c) 和图 8(c) 为改进后的 Vibe 算法的检测结果, 与传统 Vibe 算法相比, 改进后的检测算法能够有效地避免错检出鬼影和空洞, 更好地保留运动目标的完整性。

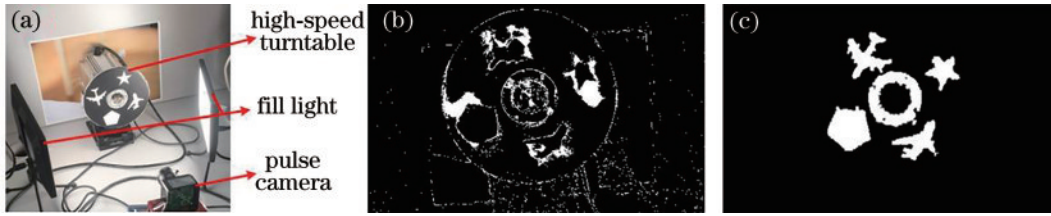


图7 500 r/min 的高速转盘的运动检测。(a)高速转盘拍摄场景;(b)改进前的 Vibe 检测算法;(c)改进后的 Vibe 检测算法

Fig. 7 Motion detection of high-speed turntable at a rotation speed of 500 r/min. (a) High-speed turntable shooting scene; (b) Vibe detection algorithm before improvement; (c) Vibe detection algorithm after improvement

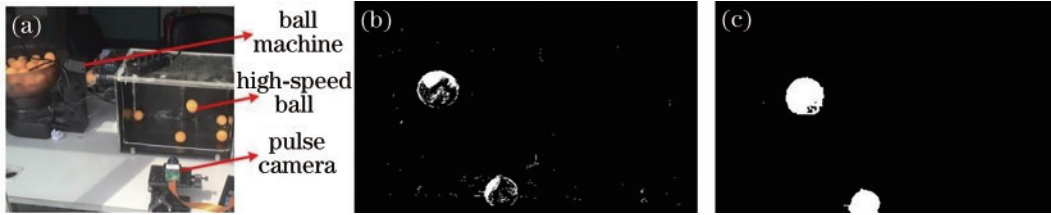


图8 高速乒乓球的运动检测。(a)高速乒乓球拍摄场景;(b)改进前的 Vibe 检测算法;(c)改进后的 Vibe 检测算法

Fig. 8 Motion detection of high-speed table tennis. (a) High-speed table tennis shooting scene; (b) Vibe detection algorithm before improvement; (c) Vibe detection algorithm after improvement

#### 4 基于 MS 目标追踪算法的改进

MS 算法是计算量小、实时性好的无参数密度估计算法,由于采用核函数直方图模型,对边缘遮挡等不敏感。然而 MS 算法缺乏必要的模板更新,追踪窗尺寸总保持不变,直方图特征在目标颜色特征描述方面略显匮乏、缺少空间信息,所以当目标尺寸有所变化时,目标跟踪精度就会降低,甚至跟踪失败。为了更好地适应脉冲图像传感器数据以及弥补传统 MS 算法的缺陷,本研究提出一种结合 Vibe 运动检测的改进算法。

根据前面的传感器原理分析可知,脉冲间隔与光强成反比,在一定程度上反映光强的大小。因此,为了减少还原过程中引入的误差,将像素点的脉冲间隔作为像素的值来描述概率密度分布。首先通过连通域标记对前面的 Vibe 运动检测结果进行前景的边界标记,然后从标记的前景中选择要追踪的运动目标,并以标记区域作为目标模型的像素区域,以图 9 中对转盘图案的追踪为例。

建立目标模型,假设初始  $t_0$  时刻目标所在的像素区域为  $R_Q$ ,区域内的第  $n$  个像素点为  $p_n$ ,其脉冲间隔为  $z_n$ ,中心点像素为  $p_0$ ,其脉冲间隔为  $z_0$ 。为了减少周围

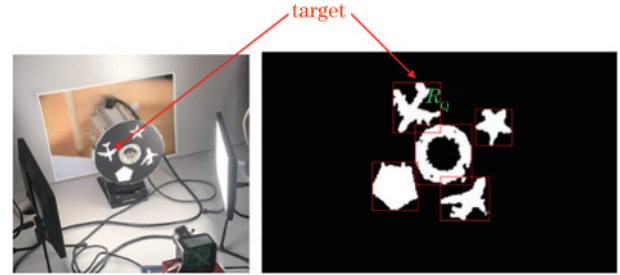


图9 追踪目标的选择

Fig. 9 Selection of tracking target

环境对脉冲间隔概率密度的影响,本实验只计算目标本身脉冲间隔的概率密度,因此利用 Vibe 的检测结果将目标区域内各个像素点及其脉冲间隔调整为

$$\begin{cases} z_n = p_{t_0}(i_n, j_n), B_{t_0}(i_n, j_n) = 255 \\ z_n = 0, \quad \text{others} \\ p_n = (i_n, j_n), p_n \in R_Q \\ z_0 = p_{t_0}(i_0, j_0), B_{t_0}(i_0, j_0) = 255 \\ z_0 = 0, \quad \text{others} \\ p_0 = (i_0, j_0), p_0 \in R_Q \end{cases} \quad (13)$$

然后计算目标模型的概率密度  $Q_u(p_0)$ ,计算公式为

$$\begin{cases} Q_u(p_0) = C \sum_{n=1}^{N_{\text{bin}}} K \left( \left\| \frac{p_n - p_0}{h} \right\|^2 \right) \delta [b(z_n) - u] b(z_n), u \in [1, 2, 3, \dots, N_{\text{Nbin}}] \\ K \left( \left\| \frac{p_n - p_0}{h} \right\|^2 \right) = \frac{2}{\pi} \left( 1 - \left\| \frac{p_n - p_0}{h} \right\|^2 \right), \left\| \frac{p_n - p_0}{h} \right\|^2 \leq 1 \end{cases} \quad (14)$$

式中:  $K$  为 Epanechnikov 核函数;  $h$  为核函数的窗口大小;  $C$  为归一化系数,使得  $\sum_{u=1}^{N_{\text{Nbin}}} Q_u(p_0) = 1$ ;  $b(z_n)$  为区

域内像素点的脉冲间隔  $z_n$  的直方图量化值;  $N_{\text{Nbin}}$  为最大化范围;  $u$  为特征值;  $\delta [b(z_n) - u]$  为 Delta 函数,用于

判断量化值  $b(z_n)$  是否属于特征值  $u$ ;  $N_{um}$  为该区域内像素点的数目。然后假定候选模型, 假设一段时间后, 即  $t$  时刻真实目标所在的区域变为  $R_p$ , 然后利用改进后的 Vibe 算法检测出的所有运动物体作为候选模型, 并将其标记为  $R_{p1}, R_{p2}, R_{p3}, \dots, R_{pm}, \dots$ , 如图 10 所示。

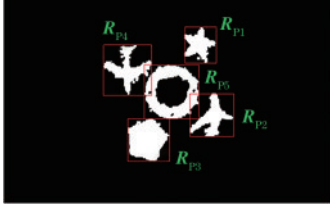


图 10 候选目标的选择。

Fig. 10 Selection of candidate targets

同理假设第  $m$  个候选模型区域  $R_{pm}$  中的第  $n$  个像素点为  $p_{nm}$ , 其脉冲间隔为  $z_{nm}$ , 该区域的中心像素点为  $p_{0m}$ , 其脉冲间隔为  $z_{0m}$ , 计算公式为

$$\begin{cases} z_{nm} = p_t(i_{nm}, j_{nm}), B_t(i_{nm}, j_{nm}) = 255 \\ z_{nm} = 0, \quad \text{others} \\ p_{nm} = (i_{nm}, j_{nm}), p_{nm} \in R_{pm} \\ z_{0m} = p_t(i_{0m}, j_{0m}), B_t(i_{0m}, j_{0m}) = 255 \\ z_{0m} = 0, \quad \text{others} \\ p_{0m} = (i_{0m}, j_{0m}), p_{0m} \in R_{pm} \end{cases}, m \in [1, 2, 3, \dots]. \quad (15)$$

然后计算出所有候选模型的概率密度, 其中第  $m$  个候选模型的概率密度  $Q_{um}(p_{0m})$  为

$$\begin{cases} Q_{um}(p_{0m}) = C \sum_{n=1}^{N_{nm}} K \left( \left\| \frac{p_{nm} - p_{0m}}{h} \right\|^2 \right) \delta[b(z_{nm}) - u] b(z_{nm}), u \in [1, 2, 3, \dots, N_{\text{Nbin}}] \\ K \left( \left\| \frac{p_{nm} - p_{0m}}{h} \right\|^2 \right) = \frac{2}{\pi} \left( 1 - \left\| \frac{p_{nm} - p_{0m}}{h} \right\|^2 \right), \left\| \frac{p_{nm} - p_{0m}}{h} \right\|^2 \leq 1 \end{cases}, m \in [1, 2, 3, \dots]. \quad (16)$$

实际上目标追踪就是从所有检测出的候选模型  $R_{p1}, R_{p2}, R_{p3}, \dots, R_{pm}, \dots$  中找到真实的目标  $R_p$ , 可以通过相似性函数 Bhattacharyya 系数来度量目标模型与所有候选模型的相似度, 其值越大, 代表两个模型越相

似。另外, 由于脉冲图像传感器具有较高的帧频, 所以两帧之间目标的位置变化必定差别不大, 即两帧之间的目标保持在一定距离  $L_d$  内, 因此可通过如下方式对 Bhattacharyya 系数进行调整:

$$\rho_m = \begin{cases} \sum_{u=1}^{N_{\text{Nbin}}} \sqrt{Q_u(p_0) \times Q_{um}(p_{0m})}, \|p_0 - p_{0m}\|^2 \leq L_d \\ 0, \quad \|p_0 - p_{0m}\|^2 > L_d \end{cases}, m = [1, 2, 3, \dots], \quad (17)$$

式中:  $\rho_m$  为调整后的第  $m$  个候选模型的 Bhattacharyya 系数。除此之外,  $L_d$  可以自适应地迭代调整, 首先给  $L_d$  设定一个很小的初始值, 然后根据  $\rho_m$  的值来调整  $L_d$  的值, 如果  $\rho_m$  的最大值为 0, 则  $L_d$  大小增加 1, 直到  $\rho_m$  的最大值不为 0, 即

$$L_d = \begin{cases} L_d + 1, \max(\rho_m) = 0 \\ L_d, \max(\rho_m) \neq 0 \end{cases}, m = [1, 2, 3, \dots]. \quad (18)$$

而真实的目标  $R_p$  则为使  $\rho_m$  最大的候选模型, 即

$$R_p = R_{pm}, \rho_m = \max(\rho_m), m \in [1, 2, 3, \dots]. \quad (19)$$

在找到真实的目标区域之后, 还需要对场景及目标进行图像重构。由式 (2) 可知, 原始脉冲间隔与实际的积分时间之间可能存在时间误差, 因此为了清晰地重构出目标及其所在场景, 还需要对原始的脉冲间隔进行滤波。本实验采用双边滤波, 即

$$\begin{cases} P_t'(i, j) = \frac{1}{W_p} \times \sum_{(x, y, z) \in S} G_{\sigma_s} G_{\sigma_r} P_z(x, y) \\ W_p = \sum_{(x, y, z) \in S} G_{\sigma_s} G_{\sigma_r} \\ G_{\sigma_s} = \exp \left[ -\frac{(i-x)^2 + (j-y)^2 + (t-z)^2}{2\sigma_s^2} \right] \\ G_{\sigma_r} = \exp \left\{ -\frac{[P_t(i, j) - P_z(x, y)]^2}{2\sigma_r^2} \right\} \end{cases}, \quad (20)$$

式中:  $P_t'(i, j)$  为滤波后的像素  $(i, j)$  的脉冲间隔;  $P_z(x, y)$  为像素  $(i, j)$  邻域内的脉冲间隔;  $W_p$  为归一化系数;  $G_{\sigma_s}$  为空域核;  $G_{\sigma_r}$  为像素核;  $S$  为参与像素  $(i, j)$  滤波的邻域;  $\sigma_s$  和  $\sigma_r$  为空域核和像素核的标准差。然后通过如下方式完成场景重构:

$$\begin{cases} G_t(i, j) = K/P_t'(i, j) \\ K = 255 \times P_{\min} \end{cases}, \quad (21)$$

式中:  $P_{\min}$  为最小的脉冲间隔;  $K$  为脉冲间隔-灰度转换系数;  $G_t(i, j)$  为像素  $(i, j)$  的灰度值。在重构的图像中标记出目标, 图 11 显示了重构的高速转盘图案的追踪过程。



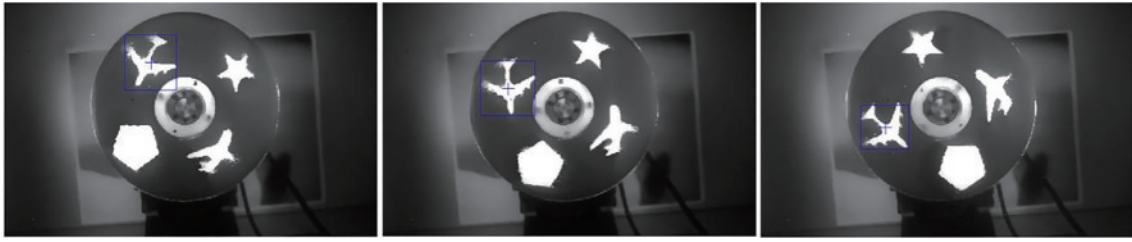


图 11 高速转盘图案的重构与追踪

Fig. 11 Reconstruction and tracking of high-speed turntable patterns

最后还需要对初始的目标模型进行更新以便于下一帧的追踪,即利用找到的真实目标  $R_p$  作为新的目标模型,即

$$R_Q = R_p. \quad (22)$$

图 12 描述了算法的整体流程。

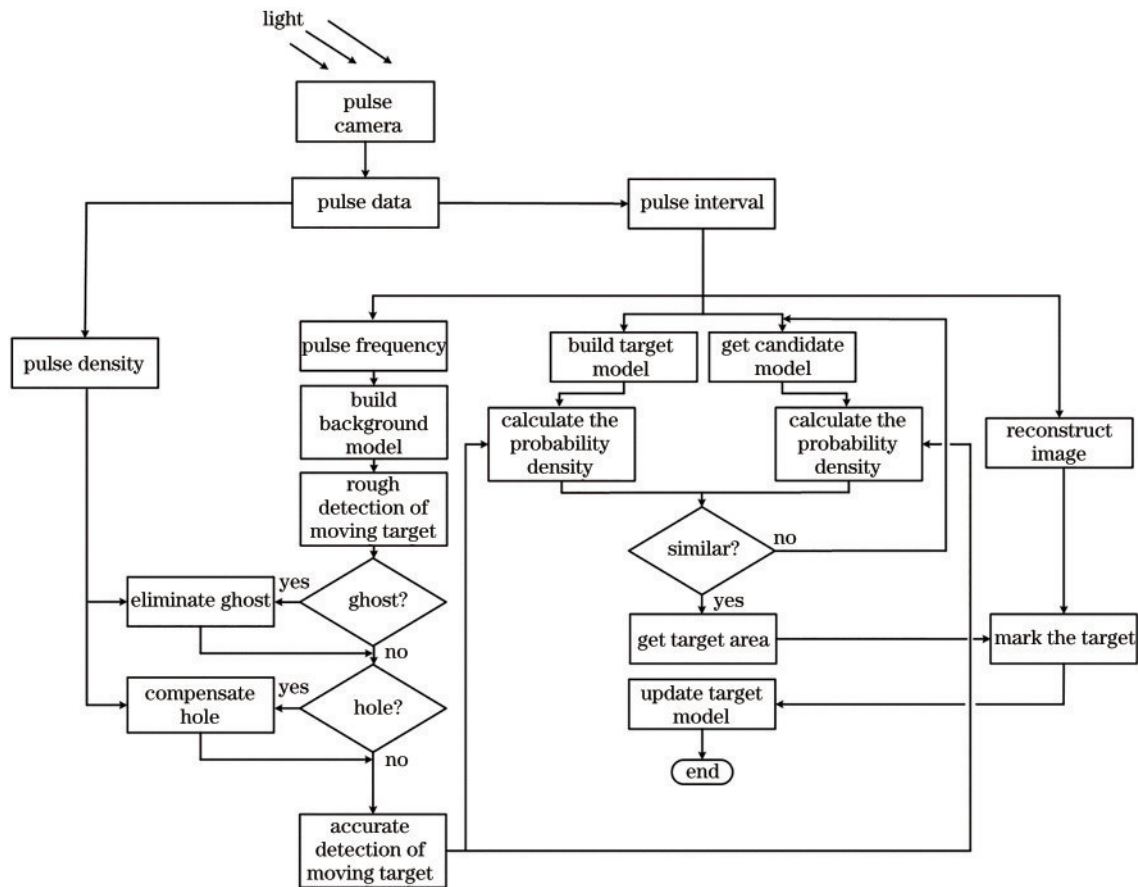


图 12 目标追踪方法的整体流程

Fig. 12 Overall flow of target tracking method

## 5 实验结果与分析

为了有效评估所提高速目标追踪方法的效果,进行了 3 组实验,即使用传统 MS 算法、文献[17]中的改进 MS 算法以及所提算法对以下场景中的目标进行追踪。由于 MS 算法和文献[17]中的改进算法仅适用于视频图像序列,故本实验将这两种算法用于重构后的图像序列。图 13 显示了 3 种算法对室外高速球的追踪过程。从图 13(b)~(e)可以看出,相比于传统 MS 算法的跟踪框,文献[17]中的改进算法的跟踪框

和所提算法的跟踪框能够更准确地锁定目标。表 1 显示了在高速球追踪场景中 3 种算法获得的目标位置及其误差对比。从表中可以看出,所提算法相比于另外两种算法对高速球的追踪误差更低、追踪精度更高,更适合脉冲图像传感器。图 14 显示了 3 种算法的高速球追踪误差随时间的变化,用以验证 3 种算法对目标追踪的稳定性。从图中可以明显看出,传统 MS 算法追踪误差波动较大、追踪过程不稳定,误差最高达到 10 pixel 以上。文献[17]中的改进算法相比于传统 MS 算法,追踪误差波动减小、稳定性得到提升,追

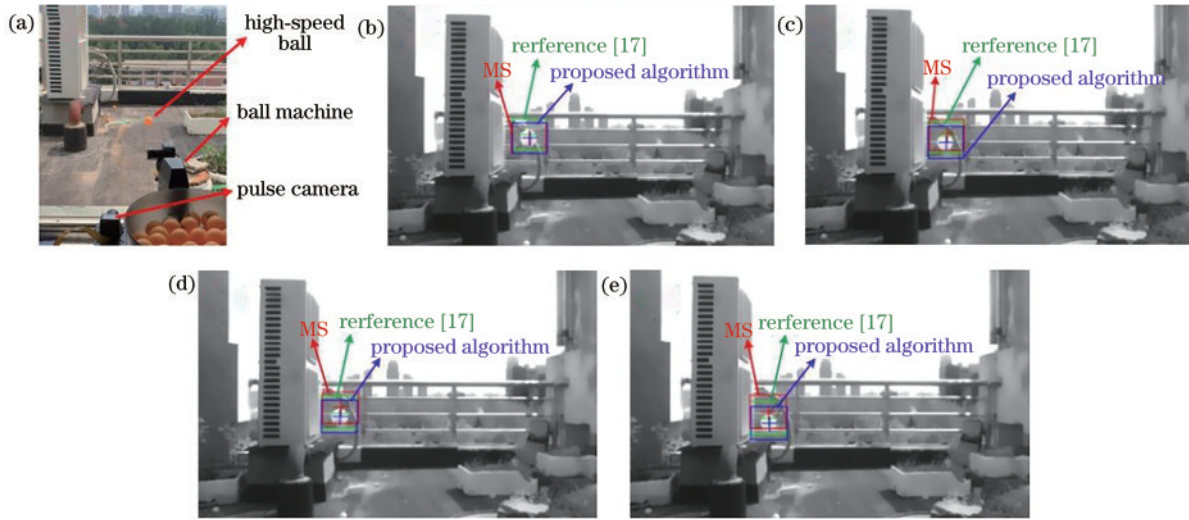


图 13 场景 1: 高速球的重构与追踪。(a) 高速球的拍摄场景; (b) 第 280 帧; (c) 第 550 帧; (d) 第 660 帧; (e) 第 900 帧

Fig. 13 Scene 1: reconstruction and tracking of high-speed ball. (a) High-speed ball shooting scene; (b) 280th frame; (c) 550th frame; (d) 660th frame; (e) 900th frame

表 1 3 种算法对高速球的追踪误差对比

Table 1 Comparison of tracking errors of three algorithms on high-speed ball

Frame number	Target center location				Error /pixel		
	Real location	MS	Reference [17]	Proposed algorithm	MS	Reference [17]	Proposed algorithm
280	(150,137)	(149,138)	(149,134)	(150,138)	1.4142	3.1623	1.0000
550	(145,145)	(146,138)	(145,140)	(145,146)	7.0711	5.0000	1.0000
660	(144,150)	(144,139)	(143,144)	(143,148)	11.0000	6.0828	2.2361
900	(140,155)	(139,144)	(139,150)	(140,155)	11.0454	5.0990	0.0000

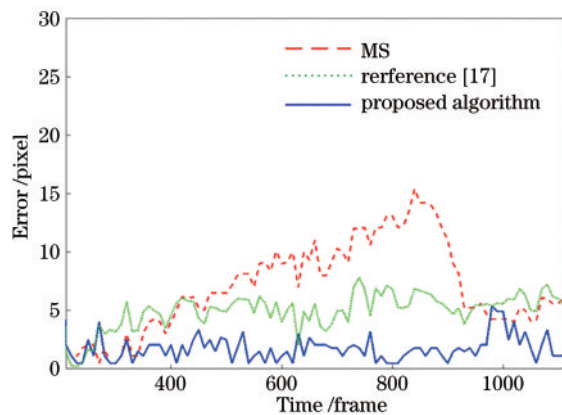


图 14 3 种算法对高速球的追踪稳定性对比

Fig. 14 Comparison of tracking stability of three algorithms on high-speed ball

踪误差稳定在 10 pixel 以下。而所提算法相比于前两种算法,误差波动更小,追踪误差也基本稳定在 5 pixel 以下,即对高速球追踪的稳定性更高。

图 15 显示了发球机产生的球速为 40 m/s 的弹射球的追踪结果。从图 15(a)可以看出,尽管弹射球由于其速度较快而出现重影,但是所提目标追踪方法仍然可以捕获清晰的目标。此外,从图 15(b)~(e)

可以看出,传统 MS 算法的跟踪框在目标被遮挡时逐渐偏离目标中心,而文献[17]的改进算法的跟踪框和所提算法的跟踪框能够牢牢锁定目标。表 2 显示了 3 种算法获得的目标位置及其误差对比。从表中可以看出,所提算法相比于另外两种算法对弹射球的追踪误差更低,追踪精度更高。图 16 显示了 3 种算法的弹射球追踪误差随时间的变化。通过对比可以发现,传统 MS 算法追踪误差波动较大,追踪稳定性较低,误差基本高于 10 pixel。文献[17]中的算法相比于传统 MS 算法,追踪误差波动明显减小,稳定性得到提升,追踪误差稳定在 10 pixel 以下。而所提算法追踪误差基本稳定在 5 pixel,相比于前两种算法,对弹射球追踪的稳定性更高。

图 17 显示了速度为 2000 r/min 的高速转盘图案的追踪过程。从图 17(a)可以看出,由于转盘速度较高,转盘上的图案已经完全模糊,但所提目标追踪方法仍然可以准确地捕捉到转盘的图案细节。另外,从图 17(b)~(e)可以看出,随着目标的旋转,传统 MS 算法的跟踪框随时间逐渐偏离目标,最终导致追踪失败,而文献[17]改进的 MS 算法的跟踪框和所提算法的跟踪框都可以稳定地追踪目标。表 3 显示了在高速转盘图案追踪场景中 3 种算法获得的目标位置及其误差对



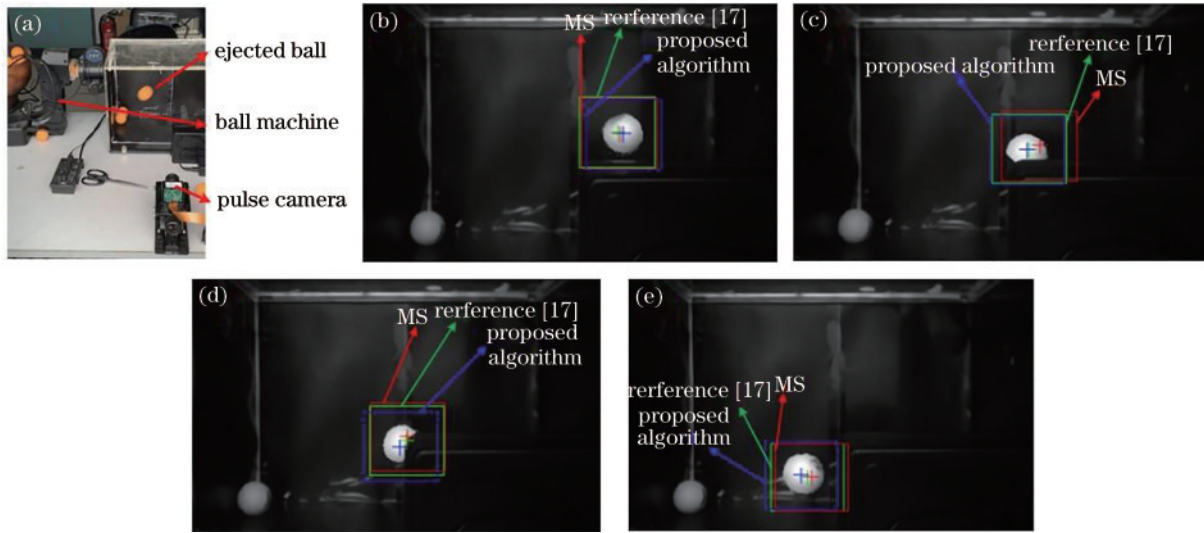


图 15 场景 2: 40 m/s 的弹射球的重构与追踪。(a) 弹射球的拍摄场景; (b) 第 221 帧; (c) 第 397 帧; (d) 第 533 帧; (e) 第 756 帧  
Fig. 15 Scene 2: reconstruction and tracking of ejected ball at a speed of 40 m/s. (a) Ejected ball shooting scene; (b) 221st frame; (c) 397th frame; (d) 533rd frame; (e) 756th frame

表 2 3 种算法对弹射球的追踪误差对比

Table 2 Comparison of tracking errors of three algorithms on ejected ball

Frame number	Target center location				Error /pixel		
	Real location	MS	Reference [17]	Proposed algorithm	MS	Reference [17]	Proposed algorithm
221	(250, 123)	(247, 125)	(245, 125)	(251, 126)	3.6056	5.3851	3.1623
397	(224, 140)	(238, 138)	(228, 141)	(226, 142)	14.1421	4.1231	2.8284
533	(201, 161)	(211, 155)	(209, 159)	(204, 165)	11.6619	8.2462	5.0000
756	(171, 191)	(182, 196)	(178, 196)	(172, 194)	12.0830	8.6023	3.1623

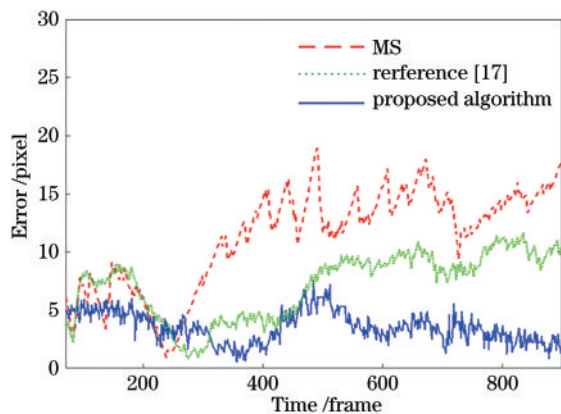


图 16 3 种算法对弹射球的追踪稳定性对比

Fig. 16 Comparison of tracking stability of three algorithms on ejected ball

比。从表 3 可以看出, 所提算法追踪误差更低, 追踪精度更高。图 18 显示了 3 种算法对高速转盘图案的追踪误差随时间的变化。通过对比可以发现, 传统 MS 算法追踪误差波动较大, 追踪过程不稳定, 误差随着时间

开始逐渐高于 10 pixel。虽然文献 [17] 中的算法相比于传统 MS 算法, 减小了追踪误差, 稳定性也得到提升, 但追踪误差同样随着目标旋转, 开始逐渐高于 10 pixel。而所提算法追踪误差基本稳定在 5 pixel, 因此, 对高速转盘图案追踪的稳定性更高。

为了定量地分析所提算法, 本实验采用位置标准差 (LSTD) 来衡量所提算法追踪的精度和稳定性, 公式为

$$D_{LSTD} = \sqrt{\frac{1}{N_{\text{FRAME}}} \sum_{k=1}^{N_{\text{FRAME}}} (\mathbf{X}_k - \mathbf{Y}_k)^2}, \quad (23)$$

式中:  $N_{\text{FRAME}}$  代表拍摄的场景的视频帧数;  $\mathbf{X}_k$  代表第  $k$  帧的目标中心的真实坐标位置;  $\mathbf{Y}_k$  代表算法定位到的目标中心的坐标位置。LSTD 越小, 说明算法定位到的目标位置越接近于目标的真实位置, 追踪的精度和稳定性越高。3 种场景下目标追踪的位置标准差如表 4 所示, 和传统 MS 算法以及文献 [17] 改进的 MS 算法相比, 所提算法的 LSTD 更小, 目标追踪的精度和稳定性更高, 更适用于脉冲图像传感器。

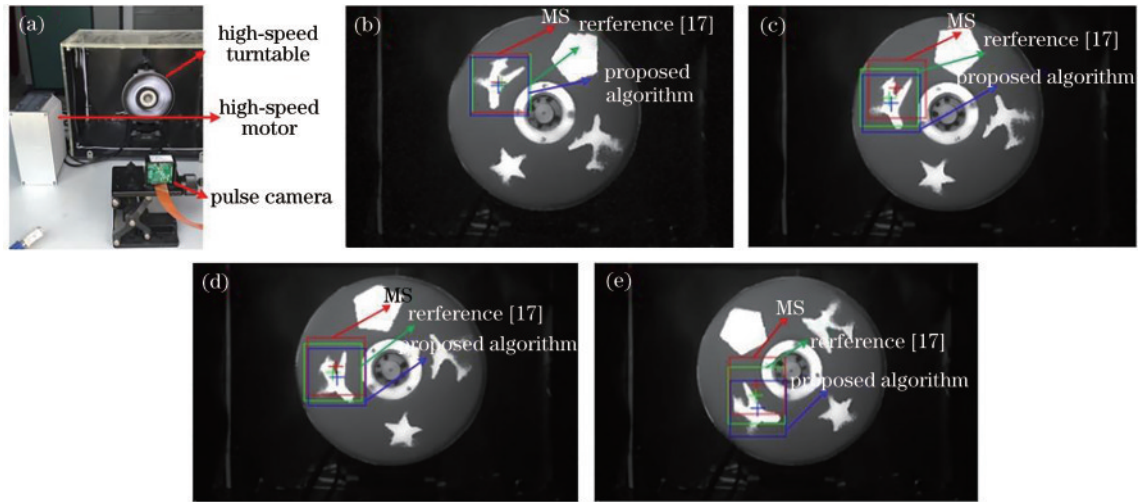


图 17 场景 3: 2000 r/min 的高速转盘图案的重构与追踪。(a) 高速转盘的拍摄场景; (b) 第 70 帧; (c) 第 145 帧; (d) 第 217 帧; (e) 第 340 帧

Fig. 17 Scene 3: reconstruction and tracking of pattern on high-speed turntable at a rotation speed of 2000 r/min. (a) High-speed turntable shooting scene; (b) 70th frame; (c) 145th frame; (d) 217th frame; (e) 340th frame

表 3 3 种算法对高速转盘图案的追踪误差对比

Table 3 Comparison of tracking errors of three algorithms on pattern of high-speed turntable

Frame number	Target center location				Error /pixel		
	Real location	MS	Reference [17]	Proposed algorithm	MS	Reference [17]	Proposed algorithm
70	(159, 80)	(160, 82)	(162, 82)	(160, 85)	2.2361	3.6056	5.0990
145	(150, 101)	(154, 87)	(146, 96)	(148, 102)	14.5602	6.4031	2.2361
217	(150, 122)	(149, 109)	(146, 115)	(150, 120)	13.0384	8.0623	2.0000
340	(167, 154)	(170, 128)	(169, 138)	(171, 152)	26.1725	16.1245	4.4721

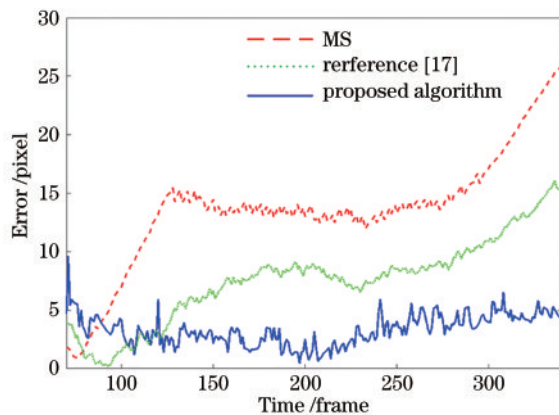


图 18 3 种算法对高速转盘图案的追踪稳定性对比

Fig. 18 Comparison of tracking stability of three algorithms on pattern of high-speed turntable

表 4 3 种场景下位置标准差对比

Table 4 Comparison of location standard deviations in three scenes

Scene	Index	MS	Reference [17]	Proposed algorithm
Scene 1	LSTD	7.9897	5.1999	2.0393
Scene 2	LSTD	12.0790	7.6781	3.7454
Scene 3	LSTD	14.4591	8.3554	3.5654

## 6 结 论

提出一种基于脉冲序图像传感器的高速目标追踪算法。首先,结合脉冲图像传感器的脉冲密度特性对传统 Vibe 运动检测算法进行改进,抑制运动检测过程中出现的鬼影和空洞,提高运动目标在检测过程中的完整性。其次,基于运动检测的结果对传统 MS 追踪算法进行改进,提高目标追踪的精度和稳定性。然后,通过图像重构完成场景再现与目标追踪。最后,通过不同场景下的实验验证所提算法的有效性。

## 参 考 文 献

- [1] Lubana E S, Dick R P. Digital foveation: an energy-aware machine vision framework[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2018, 37(11): 2371-2380.
- [2] Ranft B, Stiller C. The role of machine vision for intelligent vehicles[J]. IEEE Transactions on Intelligent Vehicles, 2016, 1(1): 8-19.
- [3] Liu Y, Sun P, Namiki A. Target tracking of moving and rotating object by high-speed monocular active vision[J]. IEEE Sensors Journal, 2020, 20(12): 6727-6744.

- [4] Wang S B, Xu Y, Zheng Y H, et al. Tracking a golf ball with high-speed stereo vision system[J]. IEEE Transactions on Instrumentation and Measurement, 2019, 68(8): 2742-2754.
- [5] Gao J, Wang Y Z, Nie K M, et al. The analysis and suppressing of non-uniformity in a high-speed spike-based image sensor[J]. Sensors, 2018, 18(12): 4232.
- [6] Dong S W, Zhu L, Xu D Y, et al. An efficient coding method for spike camera using inter-spike intervals[C]//2019 Data Compression Conference (DCC), March 26-29, 2019, Snowbird, UT, USA. New York: IEEE Press, 2019: 568.
- [7] Sun J. A fast MEANSHIFT algorithm-based target tracking system[J]. Sensors, 2012, 12(6): 8218-8235.
- [8] Yao H H. A Survey for target tracking on Meanshift algorithms[C]//2021 IEEE International Conference on Consumer Electronics and Computer Engineering, January 15-17, 2021, Guangzhou, China. New York: IEEE Press, 2021: 476-479.
- [9] Zhang C, Zhang Y F, Gao X P, et al. An improved meanshift tracking algorithm using adaptive quantization step in color space[C]//2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), October 16-18, 2019, Dublin, Ireland. New York: IEEE Press, 2019: 55-59.
- [10] Mehmood R, Nawaz R, Rao N I. Occlusion handling in meanshift tracking using adaptive window Normalized Cross Correlation[C]//Proceedings of 2014 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 14th-18th January, January 14-18, 2014, Islamabad, Pakistan. New York: IEEE Press, 2014: 126-129.
- [11] Gui Q A, Xia Y J. Kalman filter algorithm for sports video moving target tracking[C]//2020 International Conference on Advance in Ambient Computing and Intelligence (ICAACI), September 12-13, 2020, Ottawa, ON, Canada. New York: IEEE Press, 2020: 26-30.
- [12] Sarala B V, Swathi B V, Rani A S N, et al. Object tracking using block motion estimation with adaptive Kalman estimates[C]//2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology, May 19-20, 2017, Bangalore, India. New York: IEEE Press, 2017: 751-754.
- [13] Li H M, Huang L, Zhang R Q, et al. Object Tracking in Video Sequence based on Kalman filter[C]//2020 International Conference on Computer Engineering and Intelligent Control (ICCEIC), November 6-8, 2020, Chongqing, China. New York: IEEE Press, 2020: 106-110.
- [14] Duan Q C, Zhou Q, Duan P. A robust and real-time algorithm for human face tracking using improved particle filtering[C]//2009 Chinese Control and Decision Conference, June 17-19, 2009, Guilin. New York: IEEE Press, 2009: 2421-2425.
- [15] Gao B K, Li W C, Wang S. An improved Particle filter tracking algorithm[C]//Proceedings of the 29th Chinese Control Conference, July 29-31, 2010, Beijing, China. New York: IEEE Press, 2010: 2581-2584.
- [16] 刘欢, 李春庚, 安居白, 等. 基于核相关滤波的多目标追踪[J]. 激光与光电子学进展, 2019, 56(12): 121501.  
Liu H, Li C G, An J B, et al. Multiple object tracking based on kernelized correlation filter[J]. Laser & Optoelectronics Progress, 2019, 56(12): 121501.
- [17] 王凯, 高勇, 吴敏, 等. 改进 ViBe 算法与 Meanshift 结合的目标跟踪方法[J]. 科学技术与工程, 2020, 20(21): 8663-8670.  
Wang K, Gao Y, Wu M, et al. An moving target tracking method based on improved ViBe algorithm and Meanshift[J]. Science Technology and Engineering, 2020, 20(21): 8663-8670.
- [18] Xu J T, Yang Z, Gao Z Y, et al. A method of biomimetic visual perception and image reconstruction based on pulse sequence of events[J]. IEEE Sensors Journal, 2019, 19(3): 1008-1018.
- [19] Xu J T, Xu L, Gao Z Y, et al. A denoising method based on pulse interval compensation for high-speed spike-based image sensor[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2021, 31(8): 2966-2980.
- [20] Xu J T, Zhang P W, Yang Z, et al. Image lag elimination algorithm for pulse-sequence-based high-speed image sensor[J]. IEEE Access, 2019, 7: 159575-159583.
- [21] Zhao J, Xiong R Q, Huang T J. High-speed motion scene reconstruction for spike camera via motion aligned filtering[C]//2020 IEEE International Symposium on Circuits and Systems, October 12-14, 2020, Seville, Spain. New York: IEEE Press, 2020.