

多特征融合的 Camshift 运动目标跟踪算法

吴 玮 郑娟毅 杜 乐

(西安邮电大学通信与信息工程学院 西安 710121)

摘 要 传统的 Camshift 算法以颜色直方图为特征对目标进行跟踪,对刚性目标的跟踪具有较强的鲁棒性。当目标受到颜色相近的干扰物干扰或者部分遮挡时,其跟踪效果和准确度不太理想。为此,提出一种多特征融合的 Camshift 目标跟踪算法。首先,对目标的颜色特征、边缘特征和空间信息进行提取和处理,得到颜色空间直方图和空间边缘方向直方图;然后,分别在 Camshift 算法框架下得到目标匹配中心位置,采用每一帧图像的相似度向量得到权值系数,通过自适应加权融合的方法得到最优中心位置。实验结果表明,相较于传统的 Camshift 目标跟踪算法和改进的复杂特征融合的 Meanshift 算法,所提方法能够更有效地克服颜色干扰、目标重叠遮挡对跟踪效果的影响,避免了目标在跟踪过程中丢失的问题,突破了传统方法的局限性。

关键词 Camshift 算法,多特征,颜色空间直方图,空间边缘方向直方图

中图分类号 TP391.4 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.07.044

Camshift Tracking Moving Target Algorithm Based on Multi-feature Fusion

WU Wei ZHENG Juan-yi DU Le

(School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710121, China)

Abstract Traditional Camshift algorithm tracks target characterized by the color histogram, with strong robustness for rigid target tracking. However, the tracking effects and accuracy may not be ideal when they are interfered by the objects with similar color. In view of this, a Camshift tracking algorithm of multi-feature fusion was proposed. Firstly, by extracting and processing the color feature, edge feature as well as the spatial information of target, the color space histogram and spatial edge orientation histogram are obtained. Then the central locations of target matching are obtained respectively under the framework of Camshift algorithm and the weight coefficient is obtained by using the similarity vectors of each image. The optimal central location is obtained through the method of adaptive weighted fusion. The experimental results show that compared with the traditional Camshift target tracking and the improved Meanshift algorithm based on complex feature fusion, the proposed method can effectively overcome the problems of color interference and overlapping occlusion that hamper tracking effects. It can avoid the target loss in the process of tracking which breaks through the limitations of traditional methods.

Keywords Camshift algorithm, Multi-feature, Color space histogram, Spatial edge orientation histogram

1 引言

运动目标的检测和跟踪是计算机视觉领域中重要的研究课题之一,已被广泛应用在视频监控、智能交通、人机交互、机器人定位、智能车辆等领域^[1],其最终目的是在存在光照变化、环境噪声、物体遮挡等干扰的复杂场景下准确辨别出移动目标并实现有效跟踪。计算机视觉的快速发展催生了大量的目标跟踪算法,主要有基于轮廓、区域和特征的跟踪方法。其中,基于特征匹配的跟踪算法是先提取运动目标的特征信息,通常包括外形轮廓、特征点、颜色、子空间特征^[2]等,再在视频序列中进一步地搜索、判别和匹配该特征区域;然后对视频的

每一帧进行目标识别和特征匹配,从而对运动目标的运动轨迹和姿态实现连续获取,最后实现运动目标的跟踪。

传统的 Camshift (Continuously Adaptive Mean-SHIFT)^[3]目标跟踪算法是依靠目标的颜色特征建立颜色直方图,通过搜索和参考颜色模型最佳匹配的区域来实现对目标的跟踪。它可以有效解决遮挡、变形等问题,并且计算量较小,在目标跟踪领域得到了广泛的应用。但是,其主要利用单一颜色特征信息来实现移动目标的跟踪,当场景中出现颜色相近的物体遮挡或背景干扰时,跟踪效果会受到一定影响,跟踪目标甚至会丢失。鉴于此,文献[4]将自适应背景差分与 Camshift 算法相结合,提高了低对比度、颜色相似干扰条件下

到稿日期:2017-05-05 返修日期:2017-08-14 本文受国家自然科学基金项目(11401469),陕西省工业攻关项目(2013K06-07)资助。

吴 玮(1991—),男,硕士生,主要研究方向为视频图像处理、无线通信,E-mail:tomorrowww13@163.com;郑娟毅(1977—),女,高级工程师,硕士生导师,主要研究方向为宽带无线通信、计算机通信,E-mail:zjyi@xupt.edu.cn(通信作者);杜 乐(1990—),女,硕士生,主要研究方向为宽带无线通信。

运动车辆跟踪的精确度;文献[5]为抑制背景跟踪造成的干扰,采用高斯权函数来选择目标区域,有效消除了背景的干扰,但仍然易受光照变化和遮挡的影响;文献[6]将 Hu 矩和 Camshift 算法的颜色特征相结合,有效克服了单一颜色特征的弊端,抑制了背景杂波对跟踪准确度的影响;文献[7]采用 Kalman 预测和帧间差分来提高 Camshift 算法的跟踪准确性,对相似颜色具有一定的抗干扰性,但并不能有效克服目标遮挡带来的影响;文献[8]针对复杂背景、颜色相近干扰,从特征融合的角度提取 Gabor 小波纹理特征与颜色特征,对 Meanshift 目标跟踪算法进行改进,在光照条件较好时,对于背景颜色与目标颜色相近的情况,其具有较好的适应性,但对于目标遮挡和光照条件较差的情况,跟踪效果依然较差,跟踪准确度和有效性有待提高。

利用多种特征融合的方法能够有效克服仅依靠颜色信息进行目标跟踪存在的不足。为了提高视频目标跟踪效果,针对运动目标跟踪时的颜色干扰、目标重叠和遮挡等问题,本文引入了基于颜色特征的颜色空间直方图和基于边缘特征的空间边缘方向直方图,在 Camshift 目标跟踪算法框架下,采用自适应加权融合策略实现对目标的跟踪,得到最佳的搜索中心位置和窗口,以克服目标遮挡和噪声干扰,改进了传统的 Camshift 跟踪方法。

2 Camshift 目标跟踪算法的基本原理

目标检测是一种基于目标形状和统计特征的图像分割,其目的是将目标区域和背景图像分割开来。运动目标检测是在其基础上对视频流进行处理,提取出运动目标的前景区域,为下一步的目标跟踪、行为判别等环节做准备。目标跟踪算法的执行流程一般为目标预测—特征检测—模板匹配—更新,即以初始帧中的目标位置和运动模型为基础,预测下一帧中运动目标的可能区域,然后将候选区域的特征和初始区域的特征进行比对匹配,通过最优化匹配准则获得最佳匹配对象,将其相应的目标区域作为目标在当前帧的位置,同时将当前帧的目标位置作为下一帧处理的目标初始位置,如此迭代执行。

Camshift 算法即连续自适应的 MeanShift 跟踪算法。该方法首先对视频序列的每一帧做 MeanShift 运算,利用目标的颜色直方图模型将图像转换为颜色概率分布图,通过颜色概率分布图的零阶矩、一阶矩和二阶矩得到每一帧图像中跟踪窗口的质心位置 (x, y) 和方向角 θ ;然后将上一帧的结果(即搜索窗口的大小和中心位置)作为下一帧搜索窗口的初始值,以自适应地调整跟踪窗口,从而有效地解决被跟踪目标在移动过程中的形变问题。其主要步骤如下:

- 1) 读取 RGB(Red, Green, Blue) 视频图像的一帧,将其转换为 HSV(Hue, Saturation, Value) 图像 $I(i, j)$, 并提取 H 分量;
- 2) 初始化搜索窗口,手动选取目标区域(位置和大小),并计算该区域 H 分量的颜色直方图 h_b ;
- 3) 根据 h_b 对当前帧图像做反向投影,得到当前帧的概率分布投影图像,然后以 Meanshift 算法为核心分别定义:

$$\text{零阶矩: } M_{00} = \sum_i \sum_j I(i, j)$$

$$\text{一阶矩: } M_{10} = \sum_i \sum_j i \cdot I(i, j), M_{01} = \sum_i \sum_j j \cdot I(i, j)$$

$$\text{二阶矩: } M_{20} = \sum_i \sum_j i^2 \cdot I(i, j),$$

$$M_{02} = \sum_i \sum_j j^2 \cdot I(i, j),$$

$$M_{11} = \sum_i \sum_j ij \cdot I(i, j)$$

则可以得到质心位置 (x, y) 和目标方向角 θ 的计算公式为:

$$(x, y) = \left(\frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right) \quad (1)$$

$$\theta = \frac{1}{2} \arctan \left[\frac{2 \left(\frac{M_{11}}{M_{00}} - xy \right)}{\left(\frac{M_{20}}{M_{00}} - x^2 \right) - \left(\frac{M_{02}}{M_{00}} - y^2 \right)} \right] \quad (2)$$

4) 将搜索窗口的中心位置 (x_0, y_0) 移动到质心位置 (x, y) , 并判断中心位置是否收敛。若不收敛,则返回步骤 3), 根据新的中心位置继续计算质心位置,直至收敛;如果收敛,则将此时的候选区域当作当前帧中的目标区域,同时返回步骤 1), 重新获取下一帧图像,并利用当前目标区域按步骤 3) 继续执行,在新的图像帧中重新确定候选区域,依次执行。Camshift 算法的基本流程如图 1 所示。

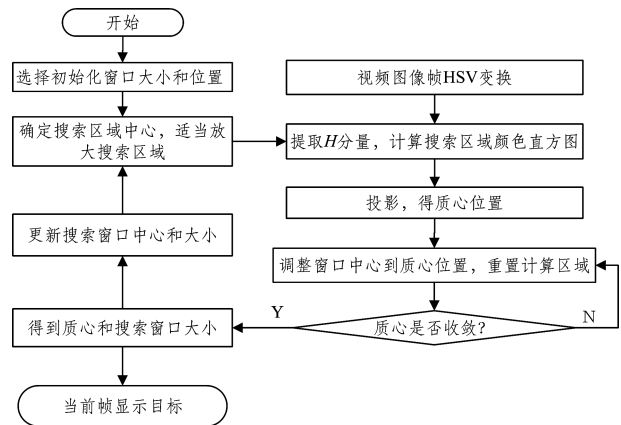


图 1 Camshift 跟踪算法的基本流程图

Fig. 1 Basic flowchart of Camshift tracking algorithm

传统 Camshift 算法仅提取目标的颜色信息特征,容易受到光照变化、颜色相近物体或者遮挡的干扰,从而造成跟踪准确度下降甚至目标丢失的问题。鉴于此,本文利用多种特征融合的方法,结合目标的颜色、边缘方向、空间等信息特征构建基于颜色特征的空间直方图和空间边缘方向直方图。该方法不仅从颜色和边缘方向统计信息进行特征匹配,而且借助空间信息来提高目标跟踪的准确性。

3 特征提取及多特征融合的 Camshift 跟踪算法

3.1 特征提取

空间直方图是一种带高阶矩信息的广义直方图,传统直方图可被视为零阶空间直方图,二阶空间直方图^[9]包含直方图每个区间的空间均值和协方差。一幅图像的二阶空间直方图可定义为 $h(b) = \langle h_b, \mu_b, \sigma_b \rangle$, 其中 h_b, μ_b 和 σ_b 分别是处于第 b 个特征子区间的像素点的概率密度函数、空间分布的均值和协方差。图像的普通直方图即零阶直方图,定义式为:

$$h_b = C \sum_{i=1}^N \delta_{i,b} \quad (3)$$

其中, $\delta(\cdot)$ 是 Kronecker delta 函数,当第 i 个像素点位于第 b

个子区间时, $\delta_{i,b} = 1$, 否则 $\delta_{i,b} = 0$; C 为归一化常数; b 为特征区间数; N 为像素点总数。对于图像的二阶空间直方图, 其图像任一像素点的空间分布均值向量和协方差矩阵分别定义为:

$$\mu_b = \frac{\sum_{i=1}^N x_i \delta_{i,b}}{\sum_{j=1}^L \delta_{j,b}} \quad (4)$$

$$\sigma_b = \frac{\sum_{i=1}^N (x_i - \mu_b)(x_i - \mu_b)^T \delta_{i,b}}{\sum_{j=1}^L \delta_{j,b}} \quad (5)$$

其中, x_i 为像素点的坐标。

为了比较两幅二阶空间直方图, 即对于 $S = \{h_b, \mu_b, \sigma_b\}$ 和 $S' = \{h_b', \mu_b', \sigma_b'\}$, 定义每个 b 区间的相似度^[10] ρ 为:

$$\rho_b = \frac{\sum_{b=1}^B \psi_b \sqrt{h_b h_b'}}{\sqrt{h_b h_b'}} \quad (6)$$

其中, $\psi_b = \eta \exp\{-\frac{1}{2}(\mu_b - \mu_b')^T \bar{\sigma}_b^{-1}(\mu_b - \mu_b')\}$, $\bar{\sigma}_b^{-1} = (\sigma_b^{-1} + \sigma_b'^{-1})^{-1}$, T 表示转置, B 为特征子区间的个数, η 为高斯归一化常数。图像的二阶空间直方图的相似度量函数增加了目标区域的空间位置信息, 丰富了其特征描述, 因此相比于传统直方图, 其相似性度量更加可靠、稳定。

3.1.1 颜色空间分布直方图

由于 RGB 颜色空间对光亮度变化比较敏感, 为减少光照变化对跟踪效果的影响, 并考虑到 HSV 颜色空间比 RGB 颜色空间更符合人的视觉特性^[11] 的特点, 本文采用 HSV 颜色直方图建模的方法来描述目标的颜色特征, 将颜色空间量化为 $B = 8 \times 8 \times 8$ 个子区间, 然后统计目标区域中所有像素点的灰度值处于颜色空间子区间的频数, 并建立颜色直方图。考虑到目标区域的大小以及像素点所处的位置对构建颜色直方图的影响, 本文采用文献[12]提出的加权和的方法, 即根据目标区域中每一个像素点和区域中心的距离关系, 为其分配相应的权值, 进而提高目标区域颜色特征提取的准确性。

对于一个 $h \times w$ 大小的矩形图像区域, 在建立颜色直方图的过程中, 根据不同位置的像素点对颜色直方图的不同贡献, 对距离区域中心越近的像素点赋予越大的权值, 而对距离区域中心越远的像素点赋予越小的权值。其权值函数被定义为:

$$\lambda(d) = \begin{cases} 1-d, & d < 1 \\ 0, & d \geq 1 \end{cases} \quad (7)$$

其中, d 表示任一像素点到区域中心的距离。用 $c(y) = \{c_b(y), b=1, 2, \dots, B\}$ 表示中心点位于 y 处的颜色分布直方图, 为免受图像缩放的影响, 通过归一化得:

$$c_b(y) = \frac{\sum_{i=1}^N \lambda(d) \frac{\|y - x_i\|}{\sqrt{w^2 + h^2}} \delta[\vartheta(x_i) - b]}{\sum_{i=1}^N \lambda(d) \frac{\|y - x_i\|}{\sqrt{w^2 + h^2}}} \quad (8)$$

其中, N 表示大小为 $h \times w$ 的目标区域内的像素点 x_i 的个数, $\vartheta(x_i)$ 表示任一像素点 x_i 位于颜色直方图中的子区间索引 b 。

本文在颜色分布直方图 $c(y)$ 的基础上, 结合二阶空间直方图得到颜色空间直方图模型 $C_b = \{c_b, \mu_b, \sigma_b\}$, 其中 μ_b 和 σ_b 分别为处于第 b 个子区间的像素点坐标的均值向量和协方差矩阵, 分别为:

$$\mu_b(y) = \frac{\sum_{i=1}^N (x_i - y) \delta_{i,b}}{\sum_{j=1}^L \delta_{j,b}} \quad (9)$$

$$\sigma_b(y) = \frac{\sum_{i=1}^N (x_i - \mu_b(y))^T (x_i - \mu_b(y)) \delta_{i,b}}{\sum_{j=1}^L \delta_{j,b}} \quad (10)$$

其中, $\delta_{i,b} = \begin{cases} 1, & x_i \in b \\ 0, & x_i \notin b \end{cases}$ ($i=1, 2, \dots, N$), x_i 为灰度图像像素点的坐标, y 为图像中心点的坐标。为了比较两个颜色空间直方图, 即对于 $C = \{c_b, \mu_b, \sigma_b\}$ 和 $C' = \{c_b', \mu_b', \sigma_b'\}$, 结合式(6)可得每个 b 区间的相似度 $\rho_{c,b}(y)$:

$$\rho_{c,b}(y) = \frac{\sum_{b=1}^{512} \psi_b \sqrt{c_b c_b'}}{\sqrt{c_b c_b'}} \quad (11)$$

其中, $\psi_b = \eta \exp\{-\frac{1}{2}(\mu_b - \mu_b')^T \bar{\sigma}_b^{-1}(\mu_b - \mu_b')\}$, 其中 $\bar{\sigma}_b^{-1} = (\sigma_b^{-1} + \sigma_b'^{-1})^{-1}$, T 表示转置, η 为高斯归一化常数。

3.1.2 空间边缘方向直方图

图像边缘描述的是目标和背景的分界线, 此处图像亮度会突然转变或者中断。图像边缘可以看作是由特定方向和幅值的像素点构成^[13], 表现为局部区域内灰度值的显著变化, 即区域内的灰度值从一个很小的值急剧上升到另一个相差较大的灰度值, 其灰度值可以看作一次跃变。边缘方向直方图将目标的轮廓信息包含进来, 通过边缘统计反映某一区域的边缘方向分布情况和物体的结构信息, 它受光照及目标颜色信息的影响较小。

为了检测图像的边缘, 需要把 GRB 彩色图像转换为 HSV 灰度图像。对于灰度图像 $I(i, j)$, 将其任一像素点 (i, j) 处的梯度矢量定义为:

$$\nabla I = [I_h, I_v]^T$$

其中, I_h 和 I_v 分别表示梯度在水平方向和垂直方向的分量, T 表示向量的转置。该像素点的幅度信息 $M(i, j)$ 和方向信息 $D(i, j)$ 分别为:

$$M(i, j) = \sqrt{I_h^2 + I_v^2} \quad (12)$$

$$D(i, j) = \arctan[I_v / I_h] \quad (13)$$

本文采用 Sobel 算子提取边缘特征, 分别用 Sobel 算子的两个模板与灰度图像 $I(i, j)$ 进行卷积运算, 得到的垂直方向和水平方向的边缘图像 I_h 和 I_v 的计算公式为:

$$\begin{cases} I_h = I * S_h \\ I_v = I * S_v \end{cases} \quad (14)$$

$$\text{其中, } S_h = \begin{bmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -1 & -\sqrt{2} & -1 \end{bmatrix}, S_v = \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix}.$$

为减少噪声影响, 本文预先设定了阈值 $K=95$, 当边缘点的幅度小于阈值时, 将这个边缘点的方向在边缘方向直方图中去除。在建立边缘方向直方图 $e_b(i, j)$ 时, 把梯度方向 $D(i, j) \in [0, 2\pi]$ 等分成 b 个区间, 本文取 $b=16$, 则每个区间的长度为 $\pi/8$, b 值越大, 计算越复杂, 精度越高。 $e_b(i, j)$ 通过统计具有相同量化方向的像素数量得到, 其中 $b=1, 2, \dots, 16$ 。对 $e_b(i, j)$ 进行归一化, 使其免受图像缩放的影响, 得:

$$e_b(i, j) = \frac{n_b}{\sum_{j=0}^{m-1} \sum_{i=0}^{n-1} \delta[M(i, j) - K]} \quad (15)$$

其中, $\delta[M(i, j) - K] = \begin{cases} 1, & M(i, j) \geq K \\ 0, & M(i, j) < K \end{cases}$, m 和 n 分别表示图像 $I(i, j)$ 在水平方向和垂直方向的像素点数目, n_b 表示 $\delta \neq 0$ 且属于第 b 个角度区间的像素点数目。

将边缘方向直方图 e_b 与图像空间信息相结合, 得到空间边缘方向分布直方图 $E_b(y) = \langle e_b, \mu_b, \sigma_b \rangle, b=1, 2, \dots, 16$ 。其中, μ_b 和 σ_b 分别是处于第 b 个子区间的像素点的坐标的均值向量和协方差矩阵, 其计算公式如式(9)~式(10)所示。比较两幅空间边缘方向直方图的匹配程度, 即对于 $E = \langle e_b, \mu_b, \sigma_b \rangle$ 和 $E' = \langle e_b', \mu_b', \sigma_b' \rangle$, 计算每个 b 区间的相似度 $\rho_{e,b}(y)$:

$$\rho_{e,b}(y) = \sum_{b=1}^{16} \psi_b \sqrt{e_b e_b'} \quad (16)$$

3.2 多特征融合的 Camshift 目标跟踪算法

基于特征匹配的跟踪算法通过提取目标区域的显著特征, 将背景和区域分割开来。如果用单一的特征集合来表达目标区域, 那么运动目标的跟踪过程就是在每一帧视频图像中准确搜索到该特征集合。传统 Camshift 算法就是一种通过颜色特征集合来实现跟踪的方法, 其优点在于对运动目标的尺度、形变和姿态变化不敏感^[14]; 但容易受图像模糊、噪声等因素的影响, 特征提取也仅依赖于单一的颜色直方图, 当存在颜色相近的干扰物时, 跟踪效果下降明显。因此, 基于特征匹配跟踪算法的研究逐渐趋向于将多个特征信息融合在一起作为跟踪特征。

常见的信息融合策略有乘性融合和加权和融合, 后者能充分利用各种信息, 且计算量较小, 实现过程简单^[15]。特征加权融合分为固定权值加权融合的方法和自适应权值加权融合的方法。本文将每一帧视频图像的相似度函数作为权值系数, 对基于颜色特征的空间直方图的跟踪结果与基于边缘特征的空间直方图的跟踪结果做自适应权值的加权融合, 权值系数越大, 说明其相似度越接近, 对跟踪效果的贡献就越大。

3.2.1 特征自适应系数

设基于颜色空间直方图的跟踪结果的目标中心坐标为 y_c , 基于空间边缘方向直方图的跟踪结果的目标中心坐标为 y_e 。考虑到融合特征的特性, 对目标位置进行更新处理, 定义两种基于特征的自适应系数分别为 ω_c 和 ω_e , 则自适应跟踪结果的最优目标中心的坐标为:

$$y_{op} = \omega_c y_c + \omega_e y_e \quad (17)$$

其中, ω_c 是基于颜色空间直方图的跟踪结果的权值, ω_e 是基于空间边缘方向直方图的跟踪结果的权值。由颜色空间直方图和空间边缘方向直方图得到相似度向量, 结合式(11)和式(16)得到归一化权值 ω_c 和 ω_e 为:

$$\omega_c = \rho_{e,b}(y) / [\rho_{e,b}(y) + \rho_{e,b}(y)] \quad (18)$$

$$\omega_e = \rho_{e,b}(y) / [\rho_{e,b}(y) + \rho_{e,b}(y)] \quad (19)$$

多特征融合的 Camshift 算法的具体步骤如下。

Step1 对视频图像的一帧进行边缘检测和颜色特征提取, 得到相应的边缘方向信息和颜色信息, 然后结合目标区域的空间信息分别建立空间边缘方向直方图 $e(y)$ 和颜色空间直方图 $c(y)$ 。

Step2 在目标跟踪窗口内, 不断迭代计算质心位置以搜索最佳匹配区域, 分别用相似度 $\rho_{e,b}(y)$ 和 $\rho_{c,b}(y)$ 判断匹配程度, 将对应的最大相似度区域定义为最佳匹配区域 Z_e 和 Z_c ,

并计算出相应的质心位置坐标 y_e 和 y_c 。为得到最大相似度, 分别对 $\rho_{e,b}(y)$ 和 $\rho_{c,b}(y)$ 进行泰勒展开。下面以 $\rho_{e,b}(y)$ 为例进行说明:

$$\rho_{e,b}(y) \approx \rho_{e,b}(y_0) + \Gamma_e(y, y_0) + \Gamma_\mu(y, y_0) \quad (20)$$

$$\Gamma_e(y, y_0) = (e(y) - e(y_0))^T \frac{\partial \rho_{e,b}(y_0)}{\partial e} \quad (21)$$

$$\Gamma_\mu(y, y_0) = (\mu(y) - \mu(y_0))^T \frac{\partial \rho_{e,b}(y_0)}{\partial \mu} \quad (22)$$

其中, y_0 是初始窗口的中心位置。

令 $\frac{\partial \rho_{e,b}(y)}{\partial y} = 0$, 得到最佳匹配中心位置 y_e 的计算公式为:

$$y_e = \frac{\sum_{i=1}^N \alpha_i g\left[\left|\frac{y_0 - x_i}{R}\right|^2\right] x_i - \sum_{b=1}^{16} \gamma_b}{\sum_{i=1}^N \alpha_i g\left[\left|\frac{y_0 - x_i}{R}\right|^2\right]} \quad (23)$$

$$\gamma_b = \eta(y_0) (e_b e_b(y_0))^{\frac{1}{2}} \delta_b^{-1}(y_0) (\mu_b - \hat{\mu}_b(y_0)) \quad (24)$$

$$\alpha_i = \frac{\varepsilon}{R^2} \sum_{b=1}^{16} \eta(y_0) \left(\frac{e_b}{e_b(y_0)}\right)^{\frac{1}{2}} \delta_i(i, k) \quad (25)$$

其中, ε 为归一化系数, R 为搜索窗口的半径; $g(t)$ 是一个核函数, t 值越接近 0 时 $g(t)$ 值愈大。类似地, 可以参照式(23)求得基于颜色模型的最佳匹配中心位置 y_c 。

Step3 采用自适应加权的方式优化 Step2 中得到的最佳匹配中心 y_e 和 y_c , 从而得到最优位置 y_{op} , 并判断阈值是否收敛, 若不收敛, 则返回 Step2, 继续计算最优位置, 直至收敛。

Step4 将 Step3 中求得的最优匹配中心位置 y_{op} 作为下一帧搜索窗口的中心, 并确定此时目标区域的大小, 返回 Step1, 重新获取下一帧图像, 迭代得到下一帧的搜索窗口大小和质心位置, 如此循环。

多特征融合的 Camshift 算法的流程图如图 2 所示。

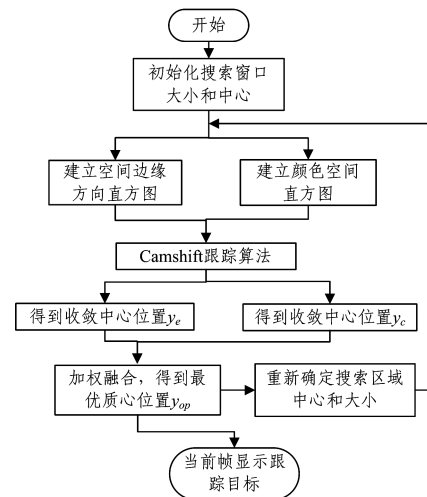


图 2 多特征融合的 Camshift 算法的流程图

Fig. 2 Flowchart of Camshift algorithm based on multi-feature fusion

3.2.2 算法的收敛性

设 y_0 为每帧图像的目标模板像素集合 $\{x_i | i=1, 2, \dots, N\}$ 的中心像素点, N 为像素点个数。目标从当前帧窗口的中心位置移动到下一帧窗口的中心位置的过程中, 跟踪算法的收敛性取决于目标区域内点的权值。受环境噪声的影响, 每个点的可信度是不同的, 通常靠近目标中心位置的像素点的可信度

大于远离中心的像素点的可信用。

引入高斯核函数 $k(y-y_0) = \exp(-\frac{\|x-x_0\|}{2\sigma^2})$, 充分利用像素点的空间位置, 利用空间直方图为目标内不同位置的像素点赋予不同的权值, 使得越靠近中心位置的像素点的权值越大。利用文献[16]提出的数学方法对基于核函数的目标跟踪算法的收敛性进行证明, 其思想是在目标迭代过程中将前一帧得到的搜索窗中心大小和质心作为下一帧搜索窗的初始值, 逐个像素地移动搜索窗, 求出每个搜索区的核密度估计值, 计算其与目标模板模型的相似度量值, 求取最大相似度量值时将目标区域作为搜索窗口位置, 依次迭代, 最后得到的多特征融合的 Camshift 算法亦具有收敛性。

4 实验仿真与对比分析

为验证基于多特征融合的 Camshift 算法的有效性, 以 VS2013+OpenCv2.4.8 为开发平台对本文提出的跟踪算法进行实现, 并在 Intel(R) Core(TM) 2.30 GHz CPU, 4 GB 内存的 PC 机上编程实现。

4.1 实验结果对比

为了直观地检验算法的效果, 采用不同背景条件下的序列视频, 待手动选取起始帧图像中的行人目标后, 对不同背景下行人在室外行走过程中受到颜色干扰和部分遮挡的视频进行实验仿真。仿真结果如图 3—图 8 所示。将传统 Camshift 算法和文献[8]提出的基于 Gabor 小波纹理特征与颜色特征融合的 Meanshift 目标跟踪算法作为参照, 与本文改进的多特征 Camshift 目标跟踪算法进行实验对比分析, 并分别选取某些序列帧的跟踪结果进行对比分析。

场景 1 该组视频序列的背景环境与选取的跟踪目标的颜色信息相近, 光照条件较差, 跟踪过程中复杂背景容易对目标造成干扰, 特别是颜色相近的物体的干扰。图 3—图 5 分别为某一视频序列的第 1, 27, 64, 83 帧, 用矩形框来标定目标。传统 Camshift 目标跟踪算法在此复杂背景下和存在颜色相近物体干扰时极易丢失目标, 文献[8]的目标跟踪算法也丢失了目标。相较而言, 本文改进的多特征融合 Camshift 目标跟踪算法能很好地实现对所标定运动目标的稳定跟踪。

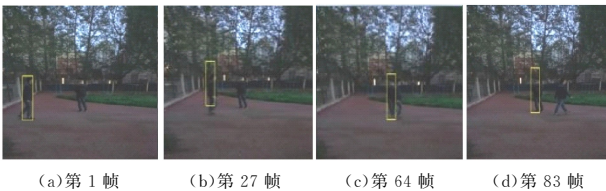


图 3 传统 Camshift 算法的跟踪效果

Fig. 3 Tracking results of traditional Camshift algorithm

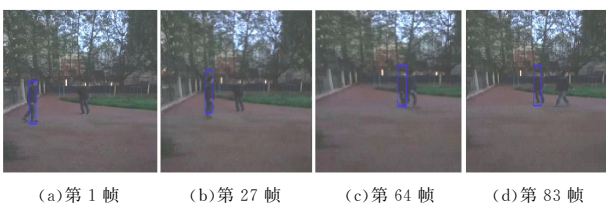


图 4 文献[8]中改进的 Meanshift 算法的跟踪效果

Fig. 4 Tracking results of improved Meanshift algorithm in ref. [8]

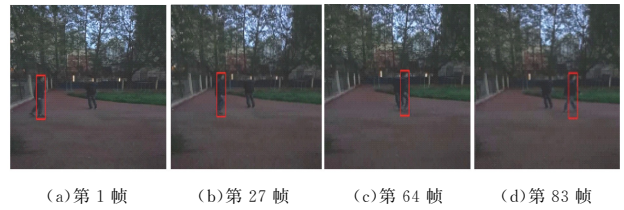


图 5 本文改进算法的跟踪效果

Fig. 5 Tracking results of improved algorithm in this paper

场景 2 该组视频序列的背景环境与选取的跟踪目标的颜色信息差别较大, 光照条件较好, 但跟踪过程中存在遮挡物的干扰。图 6—图 8 分别为另一段视频序列的第 1, 17, 48 和 92 帧, 同样用矩形框来标定目标。传统 Camshift 目标跟踪算法在目标被部分遮挡时容易丢失运动目标, 文献[8]的方法也存在同样的问题, 而本文改进的多特征融合 Camshift 目标跟踪算法能实现对运动目标的稳定跟踪。

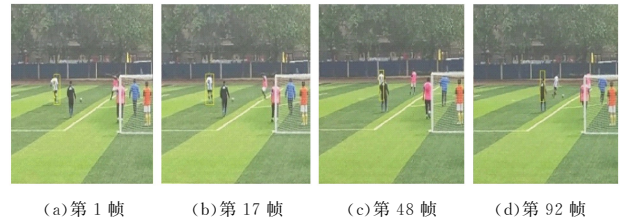


图 6 传统 Camshift 算法的跟踪效果

Fig. 6 Tracking results of traditional Camshift algorithm

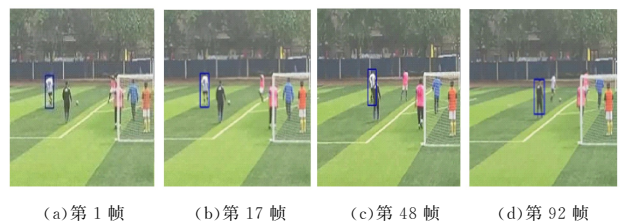


图 7 文献[8]中改进的 Meanshift 算法的跟踪效果

Fig. 7 Tracking results of improved Meanshift algorithm in ref. [8]

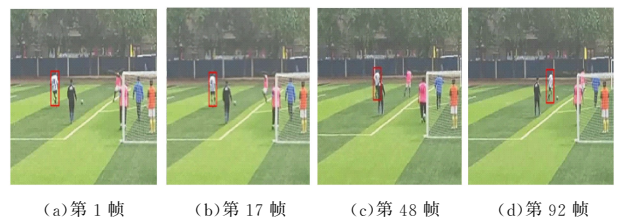


图 8 本文改进算法的跟踪效果

Fig. 8 Tracking results of improved algorithm in this paper

传统 Camshift 算法在不同场景下的目标跟踪效果如图 3 和图 6 所示, 在场景 1 和场景 2 中其对应的跟踪目标分别为行人和白衣运动员。通过观察图 3(c)和图 3(d)可以发现, 当跟踪目标受到旁边衣服颜色相近的人的干扰时, 算法的目标跟踪结果不准确, 质心发生了较大的偏移, 并在图 3(d)时已经彻底丢失了跟踪目标; 同样地, 如图 6 所示, 白衣运动员在图 6(c)所示的第 48 帧时受到黑衣运动员的遮挡影响, 导致其跟踪效果下降, 并在图 6(d)所示的第 92 帧时已经丢失了运动目标。

相较于传统 Camshift 目标跟踪算法, 文献[8]通过提取

目标的 Gabor 小波纹理特征与颜色特征对 Meanshift 目标跟踪算法进行改进。在光照条件较好时,其能有效克服与目标颜色相近的背景物体的干扰,但在场景 1 提供的光照条件较差且存在颜色相近的干扰物影响的复杂背景下(如图 4 所示),跟踪过程中矩形标注的运动目标容易被颜色相近的复杂背景干扰,尤其在第 64 帧(见图 4(c))时由于受到影响而出现了跟踪目标的混淆,最终在第 83 帧(见图 4(d))时丢失了目标;相比于场景 1 中光照较差的复杂环境,场景 2 中光照条件较好,但当运动目标受到遮挡物体干扰,如图 7(c)所示的白衣运动员被黑衣运动员遮挡时,跟踪效果明显下降,最终丢失目标。由此可见,文献[8]提出的目标跟踪算法依然有待改善,特别是对于遮挡问题。

采用多特征融合的 Camshift 跟踪算法在场景 1 和场景 2 下的实验效果分别如图 5 和图 8 所示。场景 1 中,虽然跟踪目标受到颜色相近物体的干扰(见图 5(c)),但是跟踪效果依然稳定,并且持续到第 83 帧(见图 5(d))时跟踪效果依然良好;场景 2 的跟踪过程中,当运动目标受到部分遮挡(见图 8(c))时,跟踪窗口仍能基本收敛到目标区域位置,被跟踪目标并没有丢失,到第 92 帧(见图 8(d))时跟踪效果依然稳定,从而证明了本文算法的优越性。

4.2 量化分析对比

为进一步验证基于多特征融合的 Camshift 目标跟踪算法针对颜色相近干扰和部分遮挡问题的有效性,综合评估其性能,从精确度、实时性的角度将本文改进的 Camshift 算法与传统 Camshift 目标跟踪算法和文献[8]提出的改进的 Meanshift 目标跟踪算法进行量化分析和对比。

4.2.1 精确度对比

为比较算法的精确性,进一步验证本文算法的有效性和稳定性,本文引入中心位置误差(Center Location Error)^[17]来对跟踪结果进行定量分析,求取跟踪结果中每帧图像的目标中心位置真实值 l 与实验值 l' 之间的欧氏距离: $d = \sqrt{(l_x - l_x')^2 + (l_y - l_y')^2}$,以度量算法的跟踪精度。 d 值越小,表示跟踪精度越高,反之精度越低。

基于上述场景 1、场景 2 的视频序列跟踪结果,求取中心位置误差的平均值,对 3 种算法的准确度进行对比,结果如图 9 和图 10 所示。由对比结果可知,相较于传统的 Camshift 跟踪算法和文献[8]提出的改进的 Meanshift 目标跟踪算法,基于多特征融合的 Camshift 算法无论是在光照较差的条件下受到颜色相近物体的干扰,还是跟踪目标被部分遮挡,其中心位置的误差始终保持最低水平,即跟踪窗口能相对较好地收敛到目标区域,其跟踪稳定性和准确度有了很大的改善和提高。

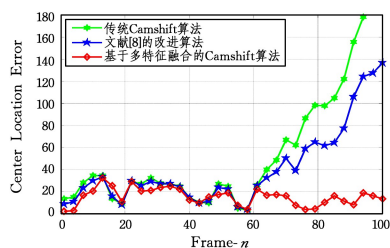


图 9 场景 1 下中心位置误差的对比

Fig. 9 Comparison of center location error in scene 1

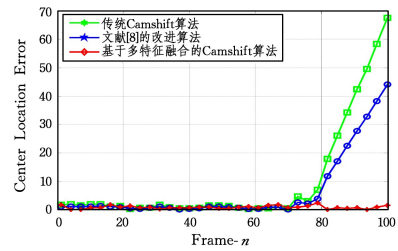


图 10 场景 2 下中心位置误差的对比

Fig. 10 Comparison of center location error in scene 2

4.2.2 实时性对比

在算法实时性方面,针对同一组实验视频序列的同一跟踪目标,在有效跟踪时段内计算出单帧的平均运行时间,并将其与传统 Camshift 目标跟踪算法和文献[8]提出的基于复杂特征融合的 Meanshift 目标跟踪算法进行对比,结果如表 1 所列。

表 1 不同算法的单帧跟踪耗时的对比

Table 1 Tracking time comparison of single frame with different algorithms

(单位:s)	
算法	平均运行耗时
传统 Camshift 算法	0.01032
文献[8]中改进的 Meanshift 算法	0.02601
基于多特征融合的 Camshift 算法	0.03308

由表 1 可知,文献[8]的方法和本文算法均采用了多特征融合策略,算法耗时主要在于特征提取和目标检测两个部分,即提取和处理颜色、边缘特征以及空间位置信息,并进行多种特征信息的融合和收敛性判别,因此相比于传统 Camshift 目标跟踪算法,其复杂度更高,耗时更长,亦即对应的跟踪速度更慢。本文算法在仿真条件下能够达到平均 30.23 帧/s 的运行速度,相比于文献[8]的方法和传统 Camshift 目标跟踪算法,单帧耗时较多,算法的实时性有待提升。

结束语 Camshift 算法的实时性好,但在复杂背景下存在由于颜色相近物体干扰或者遮挡而丢失目标的问题。本文在传统 Camshift 框架下,将颜色特征、边缘特征分别与空间分布信息相结合来构建空间直方图,并采用自适应加权的方式得到最优中心位置,克服了传统 Camshift 算法的不足。实验表明,与文献[8]提出的复杂特征融合的 Meanshift 算法相比,多特征融合的 Camshift 算法在不同情景下能有效克服颜色相近的背景或物体的干扰,提高了运动目标跟踪的稳定性和可靠性;在部分遮挡或者存在目标干扰的情景下,仍能有效实现目标跟踪,避免了跟踪过程中的目标丢失问题,突破了传统 Camshift 算法的局限性,具有更好的鲁棒性和抗干扰性。但是,所提算法的复杂度较高,导致实时性变差,后期将进一步优化算法,以改善算法的跟踪速度和实时性。

参考文献

[1] YILMAZ A, JAVED O, SHAH M. Object tracking: A survey [J]. ACM Computing Surveys, 2006, 38(4): 1-45.
 [2] 杨杰,张翔. 视频目标检测和跟踪及其应用[M]. 上海:上海交通大学出版社,2012.
 [3] BRADSKI G R. Computer vision face tracking for use in a per-

- ceptual user interface[C]//IEEE Workshop on Applications of Computer Vision. 1998:214-219.
- [4] SIRIKUNTAMAT N, SATOH S, CHALIDABHONGSE T H. Vehicle tracking in low hue contrast based on CAMShift and background subtraction[C]//International Joint Conference on Computer Science and Software Engineering. IEEE, 2015:58-62.
- [5] XIU C, BA F. Target tracking based on the improved Camshift method[C]//Chinese Control and Decision Conference. IEEE, 2016:3600-3604.
- [6] ZHU Y, XU B. A robust object tracking method combining shape descriptor and adaptive background camshift[C]//International Conference on Natural Computation. IEEE, 2015:930-933.
- [7] HAN Z, ZHANG R, WEN L, et al. Moving Object Tracking Method Based on Improved Camshift Algorithm[C]//International Conference on Industrial Informatics—Computing Technology, Intelligent Technology, Industrial Information Integration. IEEE, 2016:91-95.
- [8] YANG X, FEI S M, LI G, et al. Improved Mean shift tracking algorithm based on complicated feature fusion[J]. Control and Decision, 2014, 29(7):1297-1300. (in Chinese)
杨欣, 费树岷, 李刚, 等. 基于复杂特征融合的改进 Meanshift 目标跟踪[J]. 控制与决策, 2014, 29(7):1297-1300.
- [9] LIU W J, ZHANG Y J. Edge-color-histogram and Kalman filter-based real-time object tracking [J]. Journal of Tsinghua University Science and Technology, 2008, 48(7):1104-1107. (in Chinese)
刘惟锦, 章毓晋. 基于 Kalman 滤波和边缘直方图的实时目标跟踪[J]. 清华大学学报自然科学版, 2008, 48(7):1104-1107.
- [10] YIMIT A, HAGIHARA Y, MIYOSHI T, et al. 2-D direction histogram based entropic thresholding [J]. Neurocomputing, 2013, 120(10):287-297.
- [11] SHEN X M, ZHOU J X, XU T W. Minority Costume Image Retrieval by Fusion of Color Histogram and Edge Orientation Histogram[C]//International Conference on Computer and Information Science. IEEE, 2016:1-7.
- [12] KATJA N, ESTHER K M. Object tracking with an adaptive color-based particle filter[C]//Proceedings of the 24th DAGM Symposium on Pattern Recognition. 2002:353-360.
- [13] NAVNEET D, BILL T. Histograms of oriented gradients for human detection[C]//Computer Society Conference on Computer Vision & Pattern Recognition. 2005:886-893.
- [14] HAN B, LUO Y. Accurate face detection by combining multiple classifiers using locally assembled histograms of oriented gradients[C]//International Conference on Audio. 2012:106-111.
- [15] ZHANG Y K, LI Y F, SUN J G. Image retrieval based on multi-feature fusion[J]. Journal of Computer Applications, 2015, 35(2):495-498. (in Chinese)
张永库, 李云峰, 孙劲光. 基于多特征融合的图像检索[J]. 计算机应用, 2015, 35(2):495-498.
- [16] QU J B. Convergence analysis of kernel based Camshift algorithm[J]. Journal on Numerical Methods and Computer Applications, 2011, 32(2):148-158. (in Chinese)
曲巨宝. 基于核函数的 Camshift 算法的收敛性分析[J]. 数值计算与计算机应用, 2011, 32(2):148-158.
- [17] XU W J, HOU Z Q, YU W S, et al. Fusing mulit feature object tracking algorithm based on background weighting [J]. Journal of Air Fouce Engineering Universitu(Natrnal Science Edition), 2015, 16(3):71-76. (in Chinese)
许婉君, 侯志强, 余旺盛, 等. 背景加权的多特征融合目标跟踪算法[J]. 空军工程大学学报(自然科学版), 2015, 16(3):71-76.
- (上接第 247 页)
- [11] GOERRES G W, KAMEL E, HEIDELBERG T N, et al. PET-CT image co-registration in the thorax; influence of respiration [J]. European Journal of Nuclear Medicine & Molecular Imaging, 2002, 29(3):351-360.
- [12] ZAIDI H, NAQA I E. PET-guided delineation of radiation therapy treatment volumes: a survey of image segmentation techniques[J]. European Journal of Nuclear Medicine & Molecular Imaging, 2010, 37(11):2165-2187.
- [13] SHAO W. Monte Carlo Methods and Their Applications in Some Statistical Model[D]. Jinan: Shandong University, 2012. (in Chinese)
邵伟. 蒙特卡洛方法及在一些统计模型中的应用[D]. 济南: 山东大学, 2012.
- [14] DREVER L, ROA W, MCEWAN A, et al. Iterative threshold segmentation for PET target volume delineation [J]. Medical Physics, 2007, 34(4):1253-1265.
- [15] LEE Y B, SONG S M, LEE J S, et al. Tumor segmentation from small animal PET using region growing based on gradient magnitude[C]//International Workshop on Enterprise Networking & Computing in Healthcare Industry. Piscataway: IEEE Press, 2005:243-247.
- [16] DICE L R. Measures of the amount of ecologic association between species[J]. Ecology, 1945, 26(3):297-302.
- [17] SONG Q, BAI J, HAN D, et al. Optimal Co-Segmentation of tumor in PET-CT images with context information[J]. IEEE Transactions on Medical Imaging, 2013, 32(9):1685-1697.
- [18] CUI H, WANG X, LIN W, et al. Primary lung tumor segmentation from PET-CT volumes with spatial-topological constraint [J]. International Journal of Computer Assisted Radiology & Surgery, 2016, 11(1):1-11.
- [19] MI H, PETITJEAN C, VERA P, et al. Joint tumor growth prediction and tumor segmentation on therapeutic follow-up PET images[J]. Medical Image Analysis, 2015, 23(1):84-91.