

基于改进型 LTP 的均值漂移目标跟踪算法

邹青志¹ 黄山²

(四川大学电气信息学院 成都 610065)¹ (四川大学计算机学院 成都 610065)²

摘要 提出一种使用改进型 LTP 特征与颜色特征融合的均值漂移(Mean Shift)目标跟踪算法,该算法解决了均值漂移目标跟踪算法在变化的光强场景下跟踪难的问题。首先针对 LTP 模式过多的问题引入旋转不变的 LTP 模式,然后提出动态计算 LTP 算子阈值的方法,之后将改进的 LTP 特征与颜色特征通过自适应函数融合起来并嵌入均值漂移算法中。在变光强场景下与传统目标跟踪算法相比较,此算法跟踪结果明显优于其他算法,且鲁棒性较好。

关键词 Mean Shift 算法, LTP, 自适应, 光照, 鲁棒性

中图分类号 TP391.41 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2017.04.062

Mean Shift Tracking Algorithm Based on Improved LTP Feature Extraction

ZOU Qing-zhi¹ HUANG Shan²

(College of Electrical Engineering and Information, Sichuan University, Chengdu 610065, China)¹

(College of Computer Science, Sichuan University, Chengdu 610065, China)²

Abstract A mean shift target tracking algorithm based on improved LTP feature and color feature fusion was proposed, which can solve the problem of tracking difficult of algorithm under varying light intensity scene. The rotation invariant is introduced aiming at the problem of LTP model, then the dynamic threshold method is put forward, and then the improved LTP feature and color feature are fused to embed into mean shift algorithm through adaptive function. Compared with traditional target tracking algorithm in changing light intensity scenario, tracking result of this algorithm is superior to other algorithms, and has good robustness.

Keywords Mean shift algorithm, LTP, Adaptive, Light intensity, Robustness

1 引言

均值漂移算法自提出以来,就因简单、鲁棒性好而被广泛应用于各个领域,同时很多学者提出了各种改良的方法。例如,文献[1]通过利用像素点权重的协方差矩阵来计算目标的尺度和方向信息,这种方法计算量小而且效果比较好,但是由于其权重像素的值是通过两个直方图相关分量的比值来获得的,因此特别容易受到光照影响;文献[2]则提出了一种最佳候选模型选择法,它是对一段时间内的跟踪效果进行评价,选出相似度最高的候选模型来更新模板,这是一种较为保守的更新方法,所以难以处理显著的变化现象;文献[3-4]为了解决跟踪目标的尺度和方向问题,提出了一种基于自适应窗口的跟踪算法,但是该方法在复杂的光照环境下不能取得良好的效果;还有类似文献[5-6]这一类利用 Kalman 滤波器的模型更新方法,是建立在假设直方图各分量相互独立的情况之下的,然而这类假设在现实应用中往往难以得到满足,因此这类方法也很容易受到这类因素的干扰。光照变化一直是利用颜色特征建立模型的跟踪算法中的难题,它会引起目标与背

景的颜色、明暗变化,导致跟踪失败。

针对光照变化难以跟踪这一问题,本文提出将 LTP 特征融合颜色特征嵌入均值漂移算法中。LTP 是由 Tan 等人^[7]改进 LBP 而来,不仅继承了对光照变化不敏感的特点,而且其本身算子的一个自定义阈值能够在一定程度上均衡剧烈光照变化带来的一系列影响。首先引入旋转不变的 LTP 特征;其次根据图像的全局信息和局部信息来动态计算 LTP 算子的阈值,以达到自适应阈值的目的;最后将改进的 LTP 特征与颜色特征通过自适应函数融合起来并嵌入 Mean Shift 中。实验结果表明,将改进型 LTP 特征嵌入 Mean Shift 中能够较好地解决光照变化下跟踪难的问题。

2 LTP

LTP 由 LBP 发展而来,不仅继承了 LBP 的各种优点,还有其独到之处。LTP 能够非常高效且准确地描绘图像的纹理特征,而且计算简单,对旋转、尺度变化以及光照变化有较强的鲁棒性,更重要的是它还是一种点样本的估计形式,使其能够方便地与其他特征相结合。

到稿日期:2016-03-31 返修日期:2016-06-23

邹青志(1990—),男,硕士生,主要研究方向为图像处理与识别,E-mail:254866789@qq.com;黄山(1969—),男,博士生导师,主要研究方向为智能交通领域的图像识别与处理。

LTP 与 LBP 的计算方法类似,只是 LTP 在 LBP 算子的基础上增加了-1 的阈值模式。LTP 算子根据设置的宽度为 t 的邻域范围,求得其邻域像素 g_p 与其中中心像素 g_c 的差值,若差值大于 t ,则被量化成 1;若差值小于 $-t$,就被量化成-1;否则就被量化成 0。因此,LTP 的定义公式如式(1)所示:

$$s(u) = \begin{cases} 1, & g_p - g_c \geq t \\ 0, & |g_p - g_c| < t \\ -1, & g_p - g_c \leq -t \end{cases} \quad (1)$$

其中, t 为自定义的一个阈值,正是因为该阈值的存在,使得 LTP 能够在剧烈的光强变化下仍然保持良好的鲁棒性;但同时 t 是一个固定阈值的问题也需要解决,这是由于视频图像的光线是时刻改变的,而固定的阈值肯定不能满足这一需求,会影响算法的性能。给定其模板窗口为 3×3 ,权值为 2^i ,LTP 的编码公式如式(2)所示:

$$LTP = \sum_{i=0}^8 2^i \cdot S(u) \quad (2)$$

LTP 的编码与 LBP 的编码过程类似。首先假定这里给定的阈值 t 是 10,经过式(1)的计算,阈值化过程如图 1 所示,从而形成了由 0,-1,1 组成的八位编码。

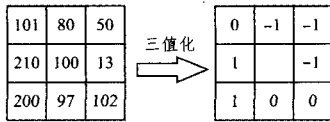


图 1 局部三值化

为了消除负值,Tan 等人又将上述编码过程分解为上半部分(LTUP)和下半部分(LTLP),其中 LTUP 编码中的-1 由 0 代替,LTLP 编码中的 1 由 0 代替、-1 由 1 代替。图 2 给出了 LTUP 和 LTLP 编码的效果。

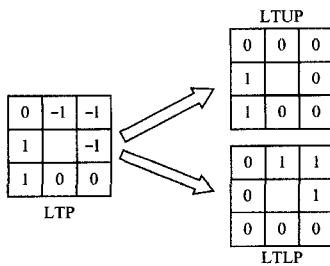


图 2 LTUP 和 LTLP 编码过程

然后,根据 LTUP 和 LTLP 两个部分的 0 和 1 进行单独加权处理。那么 $LTUP = 2^8 + 2^7 = 192$,转换成 8 位二进制码即为 1100000; $LTLP = 2^1 + 2^2 + 2^3 = 14$,同样转换成 8 位二进制码为 0000111。由此得到了该区域的纹理信息的两个特征向量,计算每个像素的 LTP 则可以得到整幅图像的纹理信息。

3 算法改进

3.1 引入旋转不变 LTP 算子

从 LTP 的定义可以看出,LTP 算子虽然是灰度不变的,但是并不是旋转不变的,它的两个纹理特征向量只是按照一个顺序旋转,如果图像不是旋转 360 度,那么 LTP 算子将会改变。

基于以上原因,引入 Maenpaa 等人^[8]在 LBP 算子中的概念,提出旋转不变性的 LTP 算子。通过不断旋转模板邻域得到一系列初始定义的 LTP 值,取最小值作为该邻域的 LTP 值,用公式表示为:

$$LTP = \min(ROR(LTP, i) | i=0, 1, \dots, P-1) \quad (3)$$

其中, $ROR(x, i)$ 为旋转函数,表示将 x 循环右移 i ($i < P$) 位。通过旋转函数,就可以将 8 种 LTP 模式转变为 1 种 LTP 模式,这样就大大减少了 LTP 模式的种类,使得纹理表示更加简单。

3.2 动态计算 LTP 阈值

根据 LTP 的定义以及公式可以知道其阈值 t 的选取会影响整个算法的性能,由于在目标跟踪中场景的情况可能是随时变化的,如果该阈值固定,就会影响算法的鲁棒性,为此提出一个根据实际情况动态计算阈值的方案。

首先要使用最小平方中值算法^[9]来取得最小平方中值,并将其作为对全局图像的一个阈值参考。其原理就是选择 n 个像素点,用最小平方法求出该区域像素的模型参数,然后分别计算每个像素值减去该参数的平方值,把所有平方值按大小顺序排列,其中间值就是最小平方中值。令 σ 为最小平方中值,则用式(4)表达如下:

$$\sigma = \min_s \text{med}(H(i) - s)^2 \quad (4)$$

其中, H 表示像素集, s 为模型参数。通过最小平方中值法就可以得到一个基于全局图像的估计。

然后要考虑到局部突变带来的影响,如果图像某个区域光变化剧烈而其他区域光基本没有变化,那么通过最小平方中值法计算出来的阈值也会相应地改变,这样就相当于让其他区域分担了光强变化带来的影响,而且该区域对于光强变化的抑制效果也会降低,从而影响整个算法的性能。为此,还要对局部图像进行一定的处理。

基于差分图像中像素点的灰度绝对值基本服从 $N(\mu, \sigma^2)$ ^[10] 这一表述,求得图像每个局部区域的均值 μ 和标准差 σ ,然后令 μ' 为 μ 和最小平方中值 σ 的均值,如式(5)所示:

$$\mu' = \frac{\mu + \sigma}{2} \quad (5)$$

然后把阈值替换成区间 $[\mu' - k\sigma, \mu' + k\sigma]$,代入式(1)中,得式(6):

$$s(u) = \begin{cases} 1, & g_p > \mu' + k\sigma \\ 0, & g_p \in [\mu' - k\sigma, \mu' + k\sigma] \\ -1, & g_p < \mu' - k\sigma \end{cases} \quad (6)$$

通过结合的方法,不仅考虑了全局的图像信息,同时也考虑到图像局部突变的影响,可以使得阈值随着环境的变化而变化,提高了算法的性能以及鲁棒性。

3.3 自适应特征融合策略

传统 Mean Shift 算法是用颜色特征来描述目标的,当目标发生形变或者缩放时,其颜色特征相对来说是比较稳定的。然而颜色特征却又容易受到光强变化和背景颜色的影响,因此把纹理特征与颜色特征融合起来作为目标跟踪的判断依据。

有一种直接特征融合的策略^[11],该策略认为纹理特征与

颜色特征的权重输出相互独立,所以把这两个特征进行直接融合,不但它们之间不会互相影响,而且把目标的表现特征与结构特征直接融合能够更为准确地描述目标。但是该策略有一个缺陷,即当一个特征失效时,该特征所占权重不会因此而降低,另外一个特征所占权重也不会因此而升高,即该特征失效时,不但没有抑制其带来的负面影响,而且还会降低目标跟踪的准确度。

因此为了提高融合算法的稳定性,本小节引用一种基于 Sigmoid 函数与巴氏相关系数的自适应融合策略,根据两类特征在当前帧中的可靠性动态调整对应的权重,可靠性高的特征获得更大的权重,反之亦然。

如式(7)所示,每个像素的权重为两类特征权重之和,这里为两种特征分别引入可靠性指数 λ_c 和 λ_l ,分别对应颜色特征的可靠性和 LBP 特征的可靠性,它们之间满足:

$$\lambda_c + \lambda_l = 1 \tag{7}$$

组合特征的巴氏相关系数重新定义为式(8):

$$\rho(y) = \lambda_c \rho_c(y) + \lambda_l \rho_l(y) \tag{8}$$

则权重系数式可以写为式(9):

$$w_i = \lambda_l \frac{w_i^l C_{l,h} \rho_l(y_0)}{2} + \lambda_c \frac{w_i^c C_{c,h} \rho_c(y_0)}{2} \tag{9}$$

显然,通过调整特征可靠性指数可以改变颜色特征与 LTP 特征的影响力。

在目标跟踪过程中,通常有一个假设,即目标状态的变化在短时间内是很缓慢的。在此基础之上,本小节通过比较在上一帧中收敛处两类特征的巴氏系数来调整其各自的可靠性指数;在前一时刻相关系数高的特征在当前帧进行定位时获得更大的权重,反之亦然。

设 rh_{k-1}^c 和 rh_{k-1}^l 分别为 $k-1$ 时刻颜色特征与纹理特征的巴氏系数,令:

$$\gamma_k = \log \frac{rh_{k-1}^c}{rh_{k-1}^l} \tag{10}$$

式(10)的取值反映了两种特征可靠性大小的比值:当颜色特征可靠性与纹理特征相同时, $\gamma_k = 0$;若前者可靠性高于后者, $\gamma_k > 0$;若前者可靠性低于后者, $\gamma_k < 0$ 。但很显然, γ_k 的取值是无界的,而可靠性指数 λ_c 和 λ_l 的取值范围为 $[0, 1]$,为此引入 Sigmoid 函数来对 γ_k 进行加权,其取值即为颜色特征的可靠性指数:

$$\lambda_c = 1 - \frac{1}{1 + e^{-\frac{\gamma_k}{b}}} \tag{11}$$

其中, b 用来控制斜率, Sigmoid 函数随着 b 的减小越发陡峭,能以更快的速度逼近上界极限值 1 和下界极限值 0。如果 b 值取得太大,其逼近速度会变慢;如果 b 值取得太小,由于其迅速逼近极值,会产生类似于阶跃的效果,不利于分析。故本文选取 $b=1$ 。

同样地,可得 λ_l ,将两个可靠性系数代入 Mean Shift 目标函数中,如式(12)所示:

$$\rho(y) = \frac{1}{2} \lambda_c \sum_{u=1}^{u_c} C_{c,h} \sum_{i=1}^{u_l} k \left(\left\| \frac{y-x_i}{h} \right\|^2 \right) \delta(b(x_i^c - u_c))$$

$$\sqrt{\frac{\hat{q}_c}{\hat{p}_{c,u}(y_0)}} + \frac{1}{2} \lambda_l \sum_{u=1}^{u_l} C_{l,h} \sum_{i=1}^{u_c} k \left(\left\| \frac{y-x_i}{h} \right\|^2 \right) \delta(b(x_i^l - u_l)) \sqrt{\frac{\hat{q}_l}{\hat{p}_{l,u}(y_0)}} \tag{12}$$

以式(12)为目标函数进行 Mean Shift 迭代,最终可得到位移序列 y 的收敛位置,在 k 时刻跟踪结束。在跟踪收敛后,重新计算两种特征的相关系数及其可靠性指数,并将其作为下一时刻两种特征的权重代入 Mean Shift 迭代。

4 实验结果与分析

本节主要涉及两个方面的实验:1)各特征在不同光照条件下的对比实验以及应用;2)实际场景中传统 Mean Shift 与本文算法跟踪效果的对比实验。本节实验在 CPU 为 i4590、8G 内存配置下的 PC 上,使用 OPENCV2.4.9 与 VS2010 的开发平台,采用 C/C++ 语言进行编程实现。

4.1 不同光照下特征提取的实验对比

对颜色特征、LBP 特征、原始 LTP 特征以及改进型 LTP 特征对光照的影响进行分析,实验随机选取了一幅彩色图像为对象,对图像进行光照变化,即对图中的所有像素值乘以亮度系数来进行模拟。图 3 是对该图像进行纹理提取的实验结果。

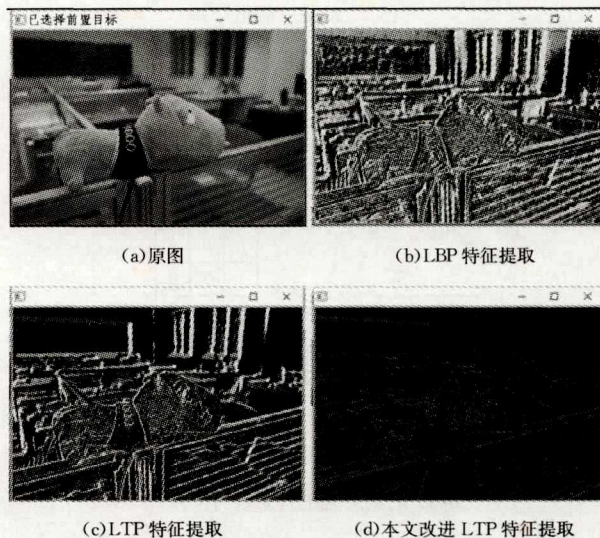


图 3 纹理特征提取

对图像采用各种算子提取特征后得到的仍然是一幅图像。从图 3 中可以看出,经过运算后,所改变的只是各个像素点的提取方式,将经过运算后的图像作为图谱。在图像识别的应用中,一般都不将图谱作为特征向量用于分类识别,而是采用特征谱的统计直方图作为特征向量,如图 4 所示,从左至右分别为使用各种特征对 0.5 倍光照图、原图、1.5 倍光照图进行特征提取的结果。

从图 4 中可以看到,颜色特征对光照的变化特别敏感,其次是 LBP 特征,在光照变化更加剧烈的情况下,这两个特征变得都不可靠;另外,原始 LTP 特征与本文的改进型 LTP 特征在光照变化下能与原图的直方图较好匹配,但可以明显地

看到本文改进型LTP特征在特征维度上明显少于原始特征,这样可以减小算法的复杂度,提升算法的实时性。

为了定量地分析光照变化给4个特征带来的影响,本文统计了光照变化给巴氏相关系数带来的影响,如图5所示。

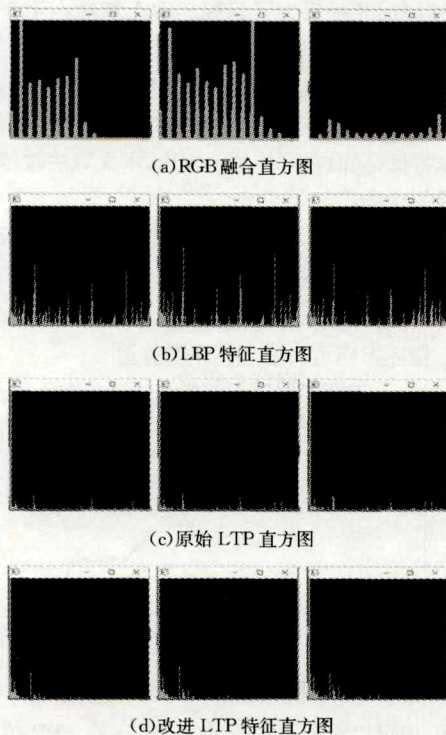


图4 不同光照下特征提取的实验结果

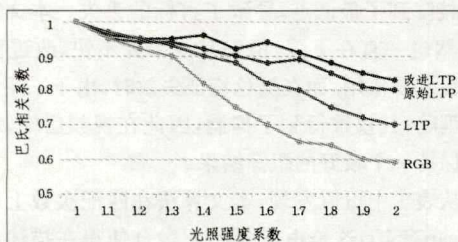


图5 光照变化对不同特征巴氏相关系数的影响

从图5中可以看到,改进型LTP特征的最小值的相关系数为0.90,即在光照强度很大的情况下,改进型LTP特征依然能保持大部分与原始模板相似。原始LTP特征虽然也体现出较好的抗光照变化能力,但是由于其阈值固定,每次变化光强之后都要对其进行大量的实验来取得较好的阈值,这个过程非常繁琐,而且在视频跟踪中场景是不断变化的,不能由单一阈值来贯穿整个视频跟踪过程。最后,LBP特征和颜色特征的匹配程度在开始没有增加光照强度的情况下,能较好地匹配,但是在随着光照强度的变化,巴氏相关系数在不断地减小,巴氏相关系数的匹配程度衰减得越来越严重。

该实验证明,在光照强度变化的情况下,仅仅依靠颜色特征是不可靠的,结合其他特征才能弥补它的不足;另外也体现出改进型LTP特征在光照强度变化下的良好性能,不论针对LBP特征还是原始LTP特征都有很大的优势。

4.2 实际场景下的跟踪实验对比

本实验选取一段飞机飞行表演视频将传统Mean Shift算

法与本文算法进行定量分析。在初始帧时手动选取目标,图中矩形框显示的即为当前帧的跟踪结果。实验结果如图6所示,从上到下分别展示的是20帧、40帧、60帧、80帧以及100帧,其中图6(a)为传统Mean Shift跟踪结果,图6(b)为本文算法跟踪结果。

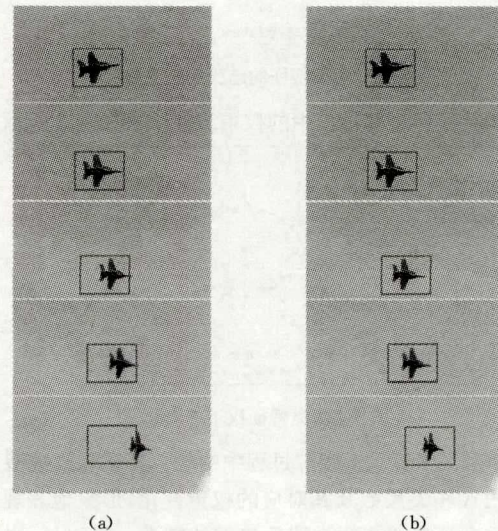


图6 传统Mean Shift与本文算法跟踪对比

由图6可以看出,用传统的Mean Shift算法对光照条件有变化的目标进行跟踪时会出现偏移甚至丢失目标的现象。在前几帧时,由于光照变化较小,且目标前景颜色特征与场景内的颜色区别较大,因此跟踪效果较好。但是在60帧以后,由于摄像机拍摄视角随飞机而转变,导致其采光发生了变化,这时目标和背景颜色相似,使得算法难以提供有鉴别力的目标描述;另外,传统Mean Shift算法采用的是对参考目标和候选目标进行整体线性加权的更新策略,这种方法使得错误的更新累积,以致目标偏移甚至丢失,在该视频中,飞机在飞行过程中有翻转、盘旋等动作以及尺度上的一些变化,造成了一定的更新累积错误;同时,该视频是由摄像机跟随飞机飞行轨迹来拍摄的,在拍摄过程中由于人为因素的影响会出现相邻帧之间目标位移过大的问题,由于传统Mean Shift算法的推导过程需要相似性度量函数在起始位置展开,这就会使得线性展开并不能有效地逼近真实值。基于以上的原因,在视频跟踪的最后出现了目标偏移现象。

用本文算法对该视频进行跟踪时,在光照条件较好的前几帧跟踪效果较好。即使是在60帧、80帧时,目标的颜色空间特征变化较大,也没有影响跟踪结果。在100帧时,前景目标的颜色特征与之前颜色特征有很大的变化,颜色特征几乎失效,本文算法此时的候选模板进行了多特征自适应的特征融合策略,当一种特征失效时,能够自适应地抑制该特征的融合权重,突出另一种有效的特征,并增加另一种特征的融合权重,使得当前帧内的候选目标与初始目标直方图仍能较相似地匹配上,使其跟踪效果较好。

为了定量地观察多特征自适应融合的过程,并验证提出的多特征自适应融合算法的有效性,对两类特征的巴氏相关性系数以及权重分别做了统计,结果如图7所示。

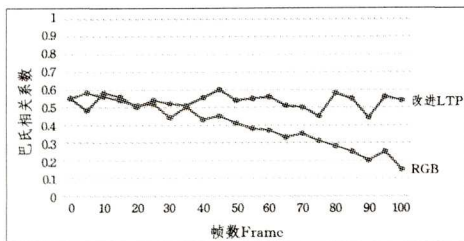


图7 两类特征的巴氏相关系数

两类特征在跟踪过程中的权重统计结果如图8所示。

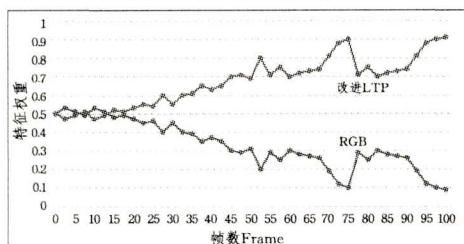


图8 两类特征权重的变化

图7和图8显示了跟踪过程中的颜色特征和改进型LTP特征的巴氏相关系数及其对应的权重变化,形象地表现出了自适应融合算法调节两类特征权重的特性。很明显,当光照强度发生变化时,颜色模型特征的匹配程度越来越低,而改进型LTP纹理特征依然是稳定的。在自适应特征融合权重调节的作用下,改进型LTP纹理特征能随着光照强度的变化获得更大的权重,使得跟踪没有丢失,跟踪效果较好,直接验证了本文所提算法的可行性。

上一个实验是在简单场景下的目标跟踪,本文算法取得了很好的跟踪效果,现选取九州大学数据库的一个室内视频来验证在复杂场景极端光照下本文算法的鲁棒性。该视频是在室内录制的,摄制者通过调整摄像机参数来模拟光照的变化,跟踪效果如图9所示。



图9 传统 Mean Shift 与本文算法跟踪对比

图9中由上到下对视频中第34帧、74帧、108帧以及133帧进行了展示,图9(a)为传统 Mean Shift 跟踪效果,图9(b)为本文跟踪效果。视频从第74帧开始光站变暗,但是由于目

标颜色特征与场景差别较大,可以发现在该帧传统 Mean Shift 跟踪效果仍然不错。随着相机参数进一步变化,在第108帧时,传统 Mean Shift 跟踪目标已经发生了偏移,这是由于光照变低外加背景颜色与目标颜色相似带来了干扰,同样地,在第133帧时几乎丢失了目标。本文算法在108帧仍然可以进行较好的跟踪,但是在133帧时同样发生了一些偏移现象。虽然本文算法在最后的极端情况中发生了偏移现象,但这是处于一种极端的光照环境中,在日常生活中该场景发生的概率非常低,同时在可控的范围内,本文算法较传统算法取得了明显的优势。

两种算法在跟踪过程中的指标比较如图10所示。为了定量地比较两者跟踪误差的大小,本文引入误检率(FPR),其公式如下:

$$FPR = \frac{\text{跟踪窗内被当成目标像素数量}}{\text{跟踪窗内像素数量}} \quad (13)$$

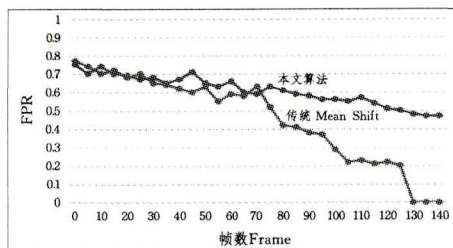


图10 两种算法跟踪误差对比

从图10中可以看到,传统 Mean Shift 的 FPR 值随着光照变化一直下降。当光照变化到使得颜色特征已经失效时, FPR 值直接降到了最低点,导致了目标的丢失。本文算法的 FPR 值虽然也一直在下降,但是其速率并不快,改进型 LTP 特征仍然在发挥作用,而在颜色特征失效时,由于采用了自适应融合的策略,其权重得到了抑制,因此在视频后端,本文算法依然能保持一个较好的跟踪结果。

另外从表1中可以发现,本文算法在匹配次数上明显少于传统 Mean Shift,这是由于多特征融合使得在描述目标时更加准确;在跟踪准确度上本文的优势也很明显,维持在一个较好的水平。唯一的不足之处在于算法的时间,本文算法略逊于传统 Mean Shift,这是由于改进型 LTP 特征提取时维数较高以及 LTP 模式较多导致的。运算时间过长,确实不适用于帧率较快的视频和快速运动目标的视频,这会使得算法的实时性大打折扣。因此在实际的运用中,可以采取隔帧处理的方式,这样就可以满足日常视频处理中的需求;对于大尺寸视频,可以采用降采样的方式,从而在不影响其跟踪精度的基础上保证算法的实时性;也可以采用 PCA 等降维方法。根据实际情况,以上几种方法也可综合使用,以获得较好的跟踪效果。

表1 跟踪性能指标比较

	平均匹配次数/ 次/帧	平均匹配时间/ 秒/帧	跟踪准确度/ %
传统算法	4.633	0.0281	28.93
本文算法	2.775	0.0523	57.82

结束语 本文算法主要处理光强变化场景下的目标跟踪,因此算法都是有针对性地选择目标检测方法、特征以及目标识别方法。本文算法的可靠性以及鲁棒性已经得到了验

证,不过算法也存在一定的不足,例如高维数的特征就会带来较大的计算量;目标识别算法本身的复杂度会影响到整个算法的实时性问题。总而言之,本文算法为光照变化场景下的目标跟踪提供了一个新的方向,而算法的不足也会在以后的研究中进行克服。

参考文献

- [1] Ning J F, ZHANG L, ZHANG D, et al. Scale and Orientation Adaptive Mean Shift Tracking [J]. *IET Computer Vision*, 2012, 6(1): 52-61.
- [2] KHAN Z H, GU Y H, BACKHOUSE A. Robust Visual Object Tracking Using Multi-Mode Anisotropic Mean Shift and Particle Filters [J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2011, 21(1): 74-87.
- [3] CHEN F, HUANG Y J, LI S Y, et al. Meanshift Target Tracking Algorithm based on Adaptive Window [J]. *Electronic Technology*, 2016, 45(5): 1-4. (in Chinese)
陈菲, 黄勇坚, 李淑彦, 等. 自适应窗口的 Meanshift 目标跟踪算法[J]. *电子技术*, 2016, 45(5): 1-4.
- [4] MEHMOOD R, NAWAZ R, RAO N I. Occlusion handling in meanshift tracking using adaptive window Normalized Cross Correlation[C]// *International Bhurban Conference on Applied Sciences & Technology*. 2014: 126-129.
- [5] HE J, YANG Y. Multi-iterative tracking method using mean-shift based on kalman filter[C]// *IEEE International Conference on Signal Processing, Communications & Computing*. 2014: 22-27.
- [6] WANG X, HU Z, FENG J, et al. Mean-Shift tracking algorithm based on Kalman filter using adaptive window and sub-blocking [C]// *World Congress on Intelligent Control & Automation*. 2015: 5438-5443.
- [7] TAN X, TRIGGS B. Enhanced local texture feature sets for face recognition under difficult lighting conditions[C]// *Analysis and Modeling of Faces and Gestures*. 2007: 168-182.
- [8] TOPI M, MATTI P, TIMO O. Texture Classification by Multi-Predicate Local Binary Pattern Operators[J]. *International Conference on Pattern Recognition*, 2000, 3(3): 3951.
- [9] ROSIN P L. Thresholding for change detection [J]. *Computer Vision and Image Understanding*, 2002, 86(2): 79-85.
- [10] TAN X, TRIGGS B. Fusing Gabor and LBP features sets for kernel based face recognition[J]. *International Conference on Analysis & Modeling of Face & Gestures*. 2007, 28(9): 235-249.
- [11] FERNÁNDEZ-CABALLERO A, CASTILLO J C, MARTÍNEZ-CANTOS J, et al. Optical flow or image subtraction in human detection from infrared camera on mobile robot[J]. *Robotics and Autonomous Systems*, 2010, 58(12): 1273-1281.
- [12] JIAO B, YAN L L, LI W. Fast convergent Gaussian Mixture Model in moving objects detection [C]// *IEEE International Conference on Computer Science and Automation Engineering (CSAE)*, 2011. 2011: 422-425.
- (上接第 274 页)
- 章国勇, 伍永刚, 谭宇翔. 一种具有量子行为的细菌觅食优化算法[J]. *电子与信息学报*, 2013, 35(3): 614-621.
- [10] LIU X L, ZHAO K L. Bacteria foraging optimization algorithm based on immune algorithm[J]. *Journal of Computer Applications*, 2012, 32(3): 634-637, 653. (in Chinese)
刘小龙, 赵奎领. 基于免疫算法的细菌觅食优化算法[J]. *计算机应用*, 2012, 32(3): 634-637, 653.
- [11] SABER A Y. Economic dispatch using particle swarm optimization with bacterial foraging effect [J]. *Electrical Power and Energy Systems*, 2012, 34(1): 38-46.
- [12] LI J, DANG J W, BU F. Study on Adaptive Step Length Bacterial Foraging Algorithm[J]. *Journal of Lanzhou Jiaotong University*, 2013, 32(6): 10-14. (in Chinese)
李珺, 党建武, 卜锋. 自适应步长的细菌觅食优化算法研究[J]. *兰州交通大学学报*, 2013, 32(6): 10-14.
- [13] CHEN G, WU X D, ZHU X Q, et al. Efficient string matching with wildcards and length constraints [J]. *Knowledge and Information Systems*, 2006, 10(4): 399-419.
- [14] NAVARRO G. Regular expression searching over Ziv-Lempel compressed text[C]// *Proceedings of the 12th Annual Symposium on Combinatorial Pattern Matching*. Berlin: Springer, 2001: 1-17.
- [15] STORN R, PDEE K. Differential Evolution-A Simple and Efficient Heuristic for Global optimization over Continuous Spaces [J]. *Journal of Global Optimization*, 1997, 11(4): 341-359.
- [16] LI J, DANG J W. Bacterial Foraging Algorithm for Solving High-Dimensional Optimization Problems [J]. *Application research of Computers*, 2016, 33(4): 1024-1027, 1033. (in Chinese)
李珺, 党建武. 细菌觅食算法求解高维优化问题[J]. *计算机应用研究*, 2016, 33(4): 1024-1027, 1033.
- [17] LI J, DANG J W, BU F. Research and Improve on Bacteria Foraging Optimization Algorithm [J]. *Computer Simulation*, 2013, 30(4): 344-347, 415. (in Chinese)
李珺, 党建武, 卜锋. 细菌觅食优化算法的研究与改进[J]. *计算机仿真*, 2013, 30(4): 344-347, 415.
- [18] CEC2005[DB/OL]. http://www3.ntu.edu.sg/home/EPNSugan/index_files/CEC-05/CEC05.htm.
- [19] YAO X, LIU Y, LIN G. Evolutionary programming made faster [J]. *IEEE Transactions on Evolutionary Computation*, 1999, 3(2): 82-102.
- [20] RATNAWEERA A, HALGAMUGE K S. Self organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[J]. *IEEE Trans on Evol Comput*, 2004, 8(3): 240-254.
- [21] BISWAS A, DASGUPTA S, DAS S, et al. Synergy of PSO and bacterial foraging optimization; a comparative study on numerical benchmarks [C]// *Innovations in Hybrid Intelligent Systems*. 2007: 255-263.
- [22] KRINK T, VESTERSTROM J S, RIGET J. Particle swarm optimization with spatial particle extension [C]// *Proceedings of IEEE Congress on Evolutionary Computation*. Honolulu, Hawaii USA, 2002: 1474-1497.