

## 特征融合和自校正的多尺度改进 KCF 目标跟踪算法研究<sup>①</sup>

陈志旺<sup>②</sup> 刘 旺<sup>③</sup>

(燕山大学工业计算机控制工程河北省重点实验室 秦皇岛 066004)

(燕山大学国家冷轧板带装备及工艺工程技术研究中心 秦皇岛 066004)

**摘 要** 传统核相关滤波(KCF)目标跟踪算法使用单一特征,不监控跟踪输出,且跟踪框的大小是固定的,在跟踪对象发生尺度变化、遮挡等异常情况下容易导致跟踪失败。针对这一问题,本文提出一种改进的 KCF 目标跟踪算法。首先,在快速方向梯度直方图(FHOG)特征的基础上级联颜色特征(CN)训练和更新滤波器;其次,利用峰值旁瓣比(PSR)检查跟踪输出,当检测到跟踪输出异常时,启动自校正机制校正跟踪输出,从而准确地重新跟踪到目标。最后,融入尺度滤波器来适应目标尺度的变化。本文对该算法进行了仿真和实物实验,实验结果验证了该算法的有效性。

**关键词** 目标跟踪;相关滤波;异常检测;自校正;尺度自适应;特征融合

## 0 引言

目标跟踪技术是计算机视觉领域研究的重点之一,它在很多领域都有广泛应用,例如视频监控<sup>[1]</sup>、自主导航<sup>[2]</sup>、军事国防<sup>[3]</sup>等。近年来,目标跟踪技术取得很大进展,但是也面临着许多问题,如形态变化、尺度变化、遮挡与消失、图像模糊等情况都会给跟踪过程造成很大的干扰<sup>[4]</sup>,这些问题都可能导致跟踪失败。

基于相关滤波的目标跟踪算法由于其高效性和鲁棒性,具有良好的跟踪效果,成为目标跟踪领域的研究热点之一。2010年,文献[5]提出最小输出平方误差和算法(minimum output sum of squared error filter, MOSSE),将相关滤波应用在目标跟踪技术上,利用离散傅里叶变换,提高了算法的运行速度,获得了较好的跟踪效果。文献[6]在 MOSSE 的基础上,引入了循环矩阵,加入了核函数,提出了基于循环结构的核跟踪算法(circulant structure with kernels, CSK),进一步提高了跟踪效果。2014年,文献[7]

提出核相关滤波(kernel correlation filter, KCF)算法,该算法是 CSK 的改进算法,采用改进的方向梯度直方图(histogram of oriented gradients, HOG)特征<sup>[8]</sup>,即快速方向梯度直方图(fast HOG, FHOG)特征代替原本的灰度特征,将单通道扩展到多通道,成为目标跟踪领域的经典算法。文献[9]按照特征图的可信度对 FHOG 特征和局部二值模式(LBP)特征进行自适应加权融合,实现对目标的定位。文献[10]提出尺度自适应核相关滤波(scale adaptive kernel correlation filter, SAMF)算法,它是 KCF 的改进算法,主要是对候选区域的目标做 7 个尺度的变化,与滤波器进行响应计算完成尺度的检测。文献[11]通过峰值旁瓣比(peak-to-sidelobe ratio, PSR)值判断遮挡类型,在目标被部分遮挡时,对目标外观模型自适应更新;当目标被完全遮挡时,使用扩展的卡尔曼滤波器,利用目标被遮挡前的运动信息来预测目标位置。

受上述文献中对特征选择、尺度变化和输出异常情况处理的启发,本文在 KCF 目标跟踪算法框架

① 国家自然科学基金(61573305)和河北省自然科学基金(F2019203511)资助项目。

② 男,1978年生,博士,副教授;研究方向:多旋翼飞行控制,目标跟踪;E-mail: czwaaron@ysu.edu.cn。

③ 通信作者,E-mail: 1365802458@qq.com。

(收稿日期:2020-12-24)

的基础上,融合了颜色特征(color name, CN)训练和更新滤波器;引入一种自校正机制,在跟踪出现异常时,通过最小化跟踪输出与期望响应之间的差异,重新定义搜索区域,校正跟踪输出;同时,加入自适应尺度滤波器,使目标框能自适应地随目标尺度的变化而变化。

### 1 KCF 目标跟踪算法简介

KCF 目标跟踪算法采用 tracking-by-detection 的跟踪框架<sup>[12]</sup>,跟踪过程中进行模板的学习和更

新,其跟踪框架如图 1 所示。

首先,在初始帧给定被跟踪的目标,提取目标区域特征,经余弦窗处理后循环移位构成训练样本集,结合理想输出训练滤波器。在后续帧中,提取前一帧预测的目标区域特征,处理后循环移位得到待检测样本集,使用训练好的滤波器对待检测样本集进行检测,得到响应图;响应图中最大值所在位置作为目标的偏移量,根据偏移量求得目标的新位置;然后再用新位置区域训练和更新滤波器,用于后续帧的检测和跟踪。

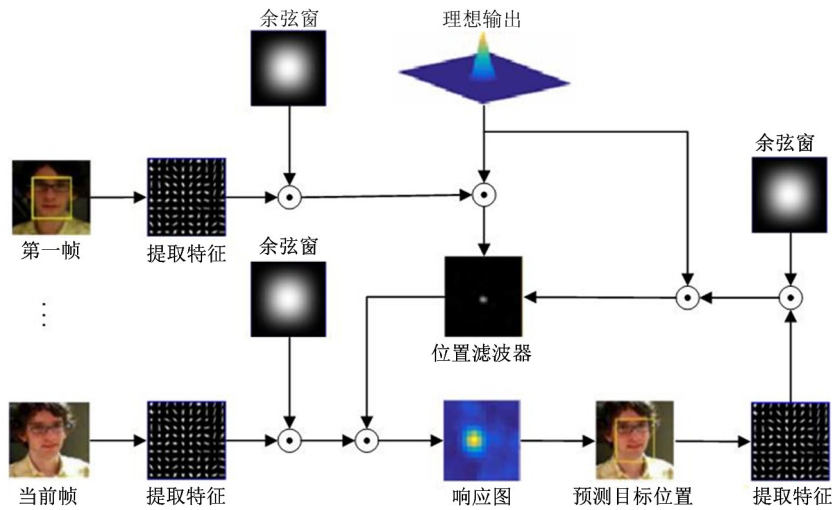


图 1 KCF 目标跟踪算法框架

获取训练样本。KCF 目标跟踪算法加入了循环矩阵的概念,利用循环移位获得了大量的训练样本。具体做法如图 2 所示。

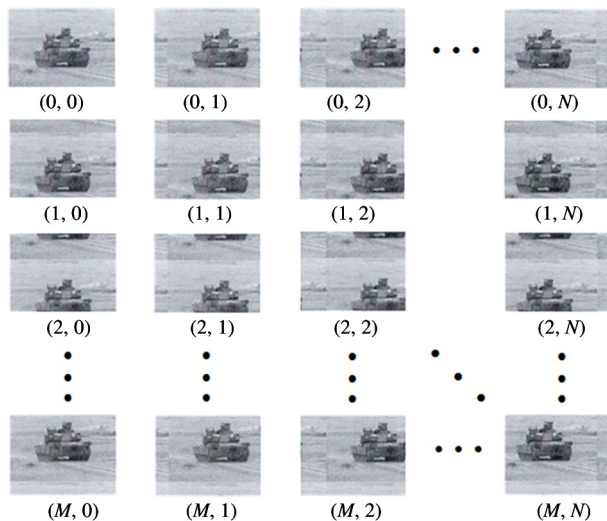


图 2 循环移位示意图

图 2 中,左上角(0,0)是基础样本,其他样本均是由基础样本循环移位得来。其中(M,N)对应的样本是把基础样本沿水平方向移动 N 个像素、沿垂直方向移动 M 个像素得到的样本。把基础样本设为  $x_{00}$ , 则(M,N)处样本表示为  $x_{mn}$ 。训练样本矩阵 X 表示为

$$X = \begin{bmatrix} x_{00} & x_{01} & x_{02} & \cdots & x_{0n} \\ x_{10} & x_{11} & x_{12} & \cdots & x_{1n} \\ x_{20} & x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{m0} & x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \quad (1)$$

然后,训练滤波器。KCF 目标跟踪算法采用岭回归来训练滤波器,滤波器的训练过程是一个岭回归的问题,其目的是使通过滤波器预测到的样本标签与真实标签的差距最小。当样本为一维的情况时,假设训练样本集为  $x = (x_1, x_2, \dots, x_n)$ , 其中  $x_1$

为初始样本,  $x$  由初始样本  $x_1$  循环移位得到, 样本标签为  $y = (y_1, y_2, \dots, y_n)$ , 符合一维高斯分布。目标函数的线性岭回归为

$$\min_{\omega} \sum_i (f(x_i) - y_i)^2 + \lambda \|\omega\|^2 \quad (2)$$

式中,  $f$  为目标函数, 满足  $f(z) = \omega^T z$ ;  $\lambda$  为正则化参数, 防止过拟合。这个回归问题的闭式解为

$$\omega = (x^T x + \lambda I)^{-1} x^T y \quad (3)$$

其中,  $I$  为单位矩阵。

KCF 目标跟踪算法引入了核方法解决样本线性不可分的问题, 重新构造函数  $f(z) = \sum_i \alpha_i k(z, x_i) = \alpha^T k(z)$ , 求得闭式解, 并将其变换到傅里叶域得到滤波器模型:

$$\hat{\alpha} = \frac{\hat{y}}{\hat{k}^{x_1 x_1} + \lambda} \quad (4)$$

其中  $k^{x_1 x_1}$  为训练样本的自相关核, 为高斯核函数, 其表达式为

$$k^{x_1 x_1} = \exp\left(-\frac{1}{\sigma^2}(\|x_1\|^2 + \|x_1\|^2 - 2x_1^T x)\right) \quad (5)$$

式中,  $\sigma$  为高斯带宽,  $x_1$  为训练初始样本。

将一维的样本扩展到二维情况, 得到滤波器在二维情况下的解为

$$\hat{A} = \frac{\hat{Y}}{\hat{K}^{XX} + \lambda} \quad (6)$$

式中,  $\hat{A}$  为滤波器的二维扩展,  $\hat{Y}$  为样本标签的二维扩展,  $\hat{K}^{XX}$  为训练样本的自相关核的二维扩展。

最后, 检测目标位置与更新滤波器。对于每一帧需要被跟踪的目标, 提取目标区域的特征, 处理后对其循环移位构成待检测样本集合, 计算待检测样本和训练样本的核函数, 与滤波器进行响应计算, 得到所有待检测样本的响应, 响应最大值位置作为目标的偏移量, 根据偏移量得到目标的新位置。计算样本响应的公式为

$$R(Z) = F^{-1}(\hat{K}^{XZ} \odot \hat{A}) \quad (7)$$

其中,  $\hat{K}^{XZ}$  为训练样本和检测样本的互相关核,  $F^{-1}$  表示傅里叶逆变换,  $\odot$  表示按位相乘运算。

为了应对跟踪过程中目标外观模型的变化, 需要实时地对滤波器和目标外观模型进行更新, 采用线性插值的方式进行更新。

$$\begin{cases} \hat{A}_i = (1 - \eta) \hat{A}_{i-1} + \eta \hat{A}_i \\ \hat{X}_i = (1 - \eta) \hat{X}_{i-1} + \eta \hat{X}_i \end{cases} \quad (8)$$

式中,  $\hat{A}_i$  为当前帧训练得到的滤波器,  $\hat{A}_{i-1}$  为上一帧检测用的滤波器,  $\hat{X}_i$  为当前帧目标的特征模板,  $\hat{X}_{i-1}$  为上一帧目标区域的特征模板,  $\eta$  为更新率。

## 2 特征融合和自校正的多尺度 KCF 目标跟踪算法

本文在 KCF 目标跟踪算法框架的基础上, 融合了 CN 颜色特征, 加入了自校正机制和尺度滤波器, 其整体框架如图 3 所示, 图中的 HC 特征由 FHOG 特征和 CN 特征级联得到。

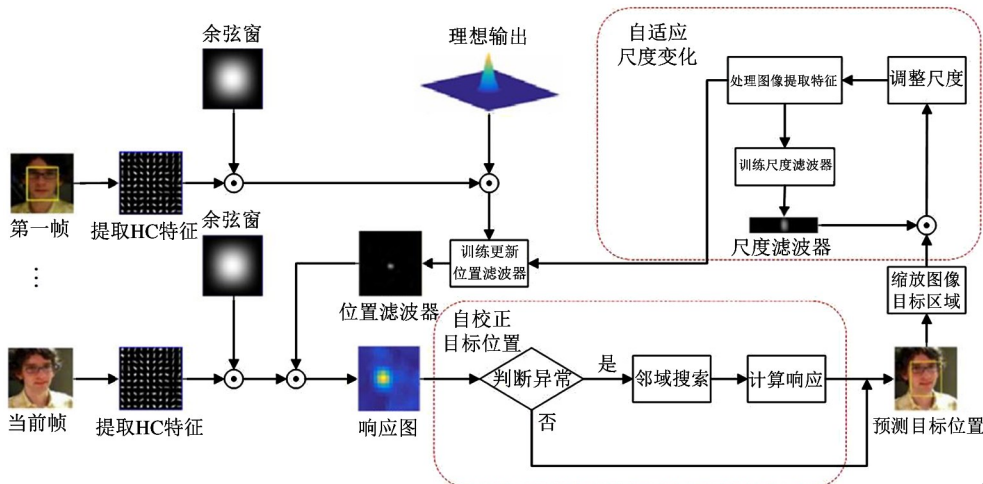


图 3 本文算法框架

### 2.1 FHOG 特征和 CN 特征融合

传统 KCF 目标跟踪算法使用单一的特征训练滤波器来进行跟踪,而单一特征在复杂的跟踪环境中存在局限性。FHOG 特征是 HOG 的改进特征,是梯度特征中很有代表性的局部特征,可以削弱光照和几何变化对跟踪结果的影响,但难以适应目标形变、轮廓模糊、旋转等复杂条件下的跟踪。CN<sup>[13]</sup> 特征将 RGB 空间转化为 11 维度的颜色特征空间,并作归一化处理降为 10 维,虽然对光照变化鲁棒性较低,但对几何变化鲁棒性很好,和 FHOG 特征形成了很好的互补。因此,本文采取融合 FHOG 特征和 CN 特征,得到一个新的特征——HC 特征,使用 HC 特征来训练和更新滤波器与外观模型,使算法能适应更多场景下的目标跟踪,提高算法的性能。

特征融合主要分为串行融合和并行融合。串行融合是最简单的融合方式,它直接将需要融合的两种特征进行简单的拼接。假设特征  $A$  的向量形式为  $(A_1, A_2, \dots, A_p)$ , 维度为  $p$ ; 特征  $B$  的向量形式为  $(B_1, B_2, \dots, B_q)$ , 维度为  $q$ , 则特征  $A$  和特征  $B$  串行融合后的特征向量形式为  $(A_1, A_2, \dots, A_p, B_1, B_2, \dots, B_q)$ , 维度为  $(p + q)$ 。并行融合将待融合特征对应维度相加,维度不够的用 0 补全,以上面的  $A$  特征和  $B$  特征为例,假设  $p > q$ ,  $A$  和  $B$  并行融合后的特征为  $(A_1 + B_1, A_2 + B_2, \dots, A_q + B_q, A_{q+1}, \dots, A_p)$ , 维度为  $p$ 。

经实验测试,本文算法将 FHOG 特征和 CN 特征以串行融合的方式进行融合的效果更好,得到  $31 + 10 = 41$  维特征向量用于滤波器和外观模型的训练与更新,测试数据见后文 4.2.2 节。特征的融合增加了算法的复杂度,但得益于循环矩阵在傅里叶域的性质,可以将多通道进行化简,在保证增强算法性能的同时也保证了算法能以较高的速度运行。

### 2.2 自校正目标位置

传统 KCF 目标跟踪算法在跟踪的过程中不会检查跟踪是否准确,一旦出现跟丢的情况,不能及时校正跟踪框到目标上,后续帧中都会更新错误的信息,从而导致跟踪失败。本文算法在传统 KCF 目标跟踪算法的基础上,加入了一种自校正机制,在目标因发生遮挡、目标丢失等情况而导致跟踪出现异常

时,自校正机制能够通过最小化跟踪输出与期望响应之间的差异,重新定义搜索区域,校正跟踪输出,使得跟踪框重新回到目标上。自校正流程如图 4 所示。

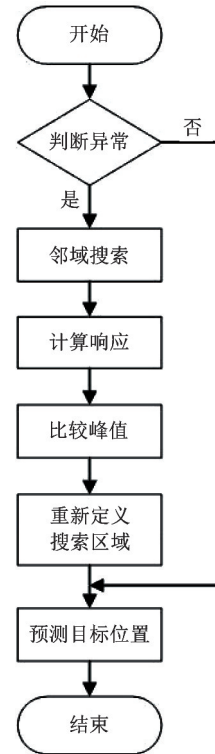


图 4 自校正流程

#### 2.2.1 判断异常

根据在 OTB-2015<sup>[14]</sup> 数据集上做的大量关于 KCF 目标跟踪算法的跟踪实验,发现传统 KCF 目标跟踪算法正常跟踪目标的时候,其响应图接近理想的二维高斯分布,并且峰值明显;而在有干扰的情况下,尤其是遮挡、快速移动、模糊等,目标特征被破坏,响应图会产生波动,不再遵循高斯分布,出现多峰,响应图整体会变得平滑,旁瓣与峰值间的对比度下降,如图 5 所示。

相关响应图显示了候选区域所有循环样本的可靠性,PSR 可以用来评估相关运算的两个目标信号的匹配程度,是一个描述主瓣相对旁瓣突出程度的物理量,可以反映峰值强度,其计算公式为

$$PSR = \frac{\max(R(X)) - u(R(X))}{\sigma(R(X))} \quad (9)$$

其中,  $\max(R(X))$  为响应图的峰值,  $u(R(X))$  为响应图的平均值,  $\sigma(R(X))$  为响应图的标准差。

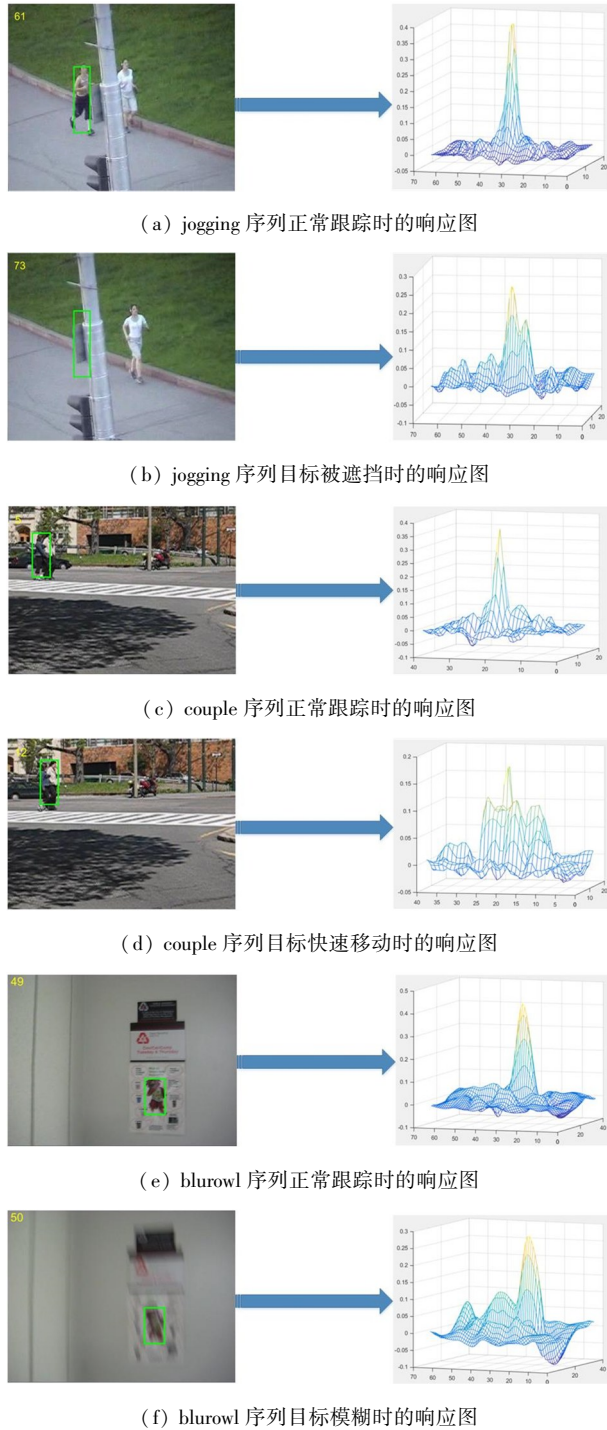


图 5 KCF 正常跟踪和受干扰时的响应图

计算样本响应的公式是  $R(Z) = F^{-1}(\hat{K}^{XZ} \odot \hat{A})$ , 其中  $\hat{A} = \frac{\hat{Y}}{\hat{K}^{XX} + \lambda}$ ,  $X$  和  $Z$  分别为所提取训练样本和检测样本的特征,  $Y$  是符合二维高斯分布的样本标签。由公式可知, 跟踪过程中,  $X$  和  $Z$  差别较小时, 即目标外观模型未发生较大变化时, 其响应结果图  $R(Z)$  应该是接近  $Y$ , 即接近理想二维高斯分

布的, 此时峰值强度较大, PSR 值较大。当有干扰时, 可以看作是目标的外观模型发生了较大变化, 即  $X$  和  $Z$  的差别较大, 此时的计算结果很难接近  $Y$ , 峰值强度变小, PSR 值较小。因此, PSR 值可以反映当前帧和先前帧之间的匹配程度, PSR 值越高, 意味着对应的样本越可靠。当目标受到干扰时, 响应图出现多峰, 响应图整体会变得平滑, 旁瓣与峰值间的对比度下降,  $\max(R(X))$  变小,  $u(R(X))$  变大,  $\sigma(R(X))$  变大, 其 PSR 值会变小。根据式(9)计算图 5 中各响应图的 PSR 值, 结果如表 1 所示。

表 1 图 5 对应的 PSR 值

	PSR	PSR	PSR
(a)	35.4	(c) 21.0	(e) 18.6
(b)	11.6	(d) 5.6	(f) 9.3

根据表 1 数据可以看出, KCF 目标跟踪算法在正常情况下其响应图的 PSR 值远大于受干扰时响应图的 PSR 值, PSR 值越大说明跟踪结果越可靠。根据这一性质, 本文采用 PSR 值来判断跟踪是否出现异常, 当 PSR 值小于某一阈值时, 可判定跟踪出现异常, 此时用自校正机制校正跟踪输出。

为了确定跟踪发生异常的 PSR 阈值, 对 basketball、bird2、blurowl、coke、couple、deer、jogging、soccer、shaking 和 women 等序列进行仿真实验, 发现 KCF 目标跟踪算法正常跟踪时其响应图的 PSR 值大于 20, 当目标受遮挡、快速移动、模糊等因素干扰时, PSR 值在 10 ~ 20 之间, 而当跟踪出现异常时, PSR 值小于 10。经过反复实验, PSR 阈值设为 10 左右最为合适。若阈值太大, 目标正常跟踪时有时也会启动自校正机制, 影响算法的运行速度; 若阈值太小, 当跟踪出现异常时, 不能及时启动自校正机制, 也会导致跟踪失败。

### 2.2.2 邻域搜索

根据 PSR 值检测到跟踪出现异常后, 启动自校正机制, 扩大搜索范围, 在之前确定的目标中心位置附近提取一些候选区域。考虑到相邻帧之间的时间间隔很小, 并且目标的移动速度受到限制, 目标是很有可能会出现在最后估计位置附近的区域中的。因此, 不需要去搜寻整个图像区域来找寻目标, 这样会

大大降低算法的运算速度。以上一帧估计的目标位置为中心,设置一组同心圆,在同心圆上以一定的规律确定一组位置用来生成候选区域,保持高效率计算的同时,也能达到逐步搜索的效果,候选区域位置的计算公式如下:

$$\begin{cases} x_{i,j} = x_0 + i \cdot \Delta T \cdot \cos(j \cdot \Delta \theta + \frac{(-1)^i + 1}{4} \cdot \Delta \theta) \\ y_{i,j} = y_0 + i \cdot \Delta T \cdot \sin(j \cdot \Delta \theta + \frac{(-1)^i + 1}{4} \cdot \Delta \theta) \end{cases} \quad (10)$$

其中,  $(x_0, y_0)$  为前一帧预测目标位置;  $\Delta T$  和  $\Delta \theta$  分别表示距离步长和角度步长;  $i$  的范围为  $[1, m]$ , 其中  $m = R_d/\Delta T, R_d$  是外圆的半径;  $j$  的范围为  $[1, n]$ , 其中  $n = 2\pi/\Delta \theta$ 。根据上述公式便可得到一系列坐标值  $(x_{i,j}, y_{i,j})$ , 根据这些坐标值来提取候选区域。

其中,外圆半径  $R_d$  由目标尺寸和响应图的峰值决定:

$$R_d = 0.8 \times \sqrt{0.025/R_{\max} \times w^2 + 0.25 \times h^2} \quad (11)$$

其中,  $R_{\max}$  为响应图的最大峰值,  $w$  为目标框的宽度,  $h$  为目标框的高度。

邻域搜索示意图如图 6 所示。

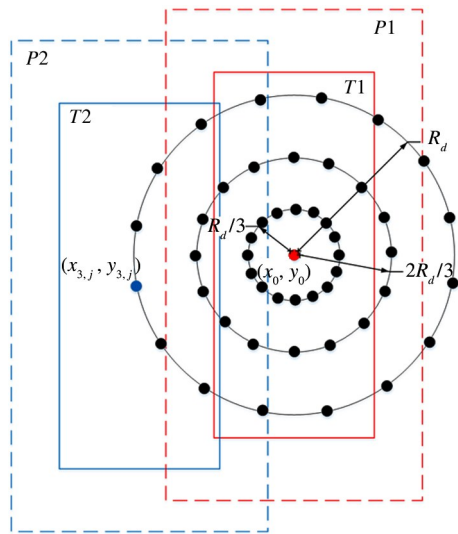


图 6 自校正机制邻域搜索示意图

图 6 中,  $(x_0, y_0)$  为原本的目标中心位置,其对应的实线框  $T1$  为目标框,对应的虚线框  $P1$  为原本的候选区域,  $R_d$  为外圆半径,  $(x_{3,j}, y_{3,j})$  为由

式(11)得出的最外面圆上的某一个点,其对应的实线框  $T2$  为扩展出的新的目标框,对应的虚线框  $P2$  为新的某一个候选区域。本文设定距离步长为  $\Delta T = R_d/3$ , 角度步长为  $\Delta \theta = \pi/8$ 。除了原本的目标中心点外,由式(10)共扩展出 48 个点,根据这 48 个点得到新的 48 个候选区域。

### 2.2.3 预测目标位置

预测目标位置实际上就是最小化实际响应与期望响应之间的差异,若要使差异最小,则需要最大化响应图得分,找出峰值最大的响应图。根据 2.2.2 节的内容确定 48 个候选区域后,将这些候选区域视为待检测样本,由式(7)得到每个待检测样本的响应图,找出每个响应图的峰值;比较这些峰值的大小,最大峰值所对应的候选区域确定为新的搜索区域,该区域对应的坐标值为  $(x_{i,j}, y_{i,j})$ , 将其假设为上一帧目标中心位置,该区域的响应图中的峰值坐标即为目标的偏移量,根据偏移量预测当前帧目标中心位置,从而实现目标位置的预测。

### 2.3 自适应尺度变化

传统 KCF 目标跟踪算法在跟踪过程中跟踪框的大小是固定的,当目标远离或靠近摄像头时,目标占搜索区域的比例会变化,即目标的尺度会发生变化。在尺度较小时,目标区域会包含较多的背景信息,而当尺度较大时,只能提取目标的局部信息,这两种情况都会降低跟踪效果甚至可能导致目标跟踪失败。为了解决跟踪过程中的尺度变化问题,本文在 KCF 目标跟踪算法中加入一个尺度滤波器以适应目标尺度变化。

尺度滤波器的训练采用岭回归方法。假设初始帧目标所在图像块为  $P$ ,以  $P$  的正中间为中心,截取不同尺度的图片。若  $P$  大小为  $M \times N$ , 尺度为  $S$ , 则使用式(12)截取不同尺度大小的图片。

$$a^n M \times a^n N \quad (12)$$

式中,  $a$  为尺度因子,  $n$  的取值范围如式(13)所示。

$$n \in \left\{ -\frac{S-1}{2}, \dots, \frac{S+1}{2} \right\} \quad (13)$$

这样就能够得到一系列不同尺度的图像块,将这些图片缩放至  $M \times N$ , 求它们的特征描述子,构成训练样本集  $f$ , 结合高斯型理想输出响应  $g$  来训练尺度滤波器  $h$ , 如图 7 所示。

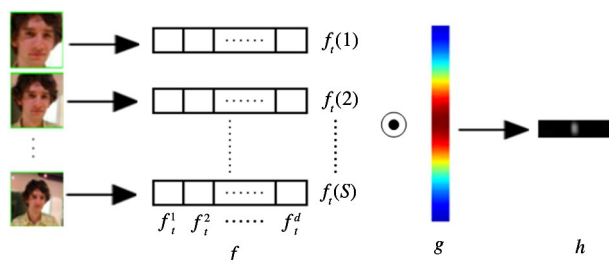


图7 尺度滤波器训练图

在新的一帧中估计出目标位置后,以其为中心,截取不同尺度的图片,再将这些图片缩放至  $M \times N$ , 分别求其特征描述子,与尺度滤波器  $h$  计算响应,响应最大的值对应的尺度即为当前目标尺度,如图 8 所示。

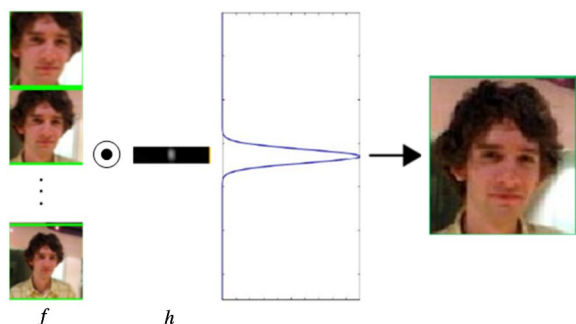


图8 尺度检测图

最后,提取当前帧尺度特征描述子训练和更新尺度滤波器。

### 3 算法整体步骤

本文算法是基于 KCF 目标跟踪算法的改进,在传统 KCF 目标跟踪算法的基础上融合了 CN 颜色特征,加入了尺度滤波器和自校正机制,能够检查跟踪输出,在输出异常时通过自校正机制校正目标位置,然后用尺度滤波器实现目标框尺度自适应变化。算法具体步骤如下。

**步骤 1** 给定初始帧的目标,确定搜索区域,提取 FHOG 特征和 CN 特征,通过串行融合得到 HC 特征,对 HC 特征进行处理,构成位置滤波器的训练样本集,根据式(6)得到位置滤波器  $\hat{A}$ 。

**步骤 2** 以目标正中间为中心,根据式(12)截取不同尺寸的图像块,然后全部缩放至初始目标框

大小,分别求其特征描述子,构成尺度滤波器的训练样本集训练得到尺度滤波器  $h$ 。

**步骤 3** 新的一帧到来后,提取前一帧目标位置区域的 HC 特征,处理后循环移位构成待检测样本,根据式(7)进行响应计算,得到响应结果图。

**步骤 4** 根据式(9)计算响应结果图的 PSR 值,判断 PSR 值是否小于阈值。若是则跳到步骤 5,否则跳到步骤 6。

**步骤 5** 启动自校正机制,根据式(10)在当前目标中心位置附近提取一系列候选区域,根据式(7)得到每个候选区域的响应结果图,峰值最大的候选区域定义为新的搜索区域,其中心位置为上一帧目标中心位置,对应的响应图视为最终响应结果图。

**步骤 6** 选取响应结果图中峰值对应的坐标为偏移量,预测目标位置。

**步骤 7** 得到目标位置后,以该位置为中心,根据式(12)截取不同尺寸的图像块,缩放至初始目标框大小,求其特征描述子构成尺度检测样本集进行响应计算,响应图峰值对应的尺度确定为目标在当前帧的尺度。

**步骤 8** 以新的目标中心位置为中心,将搜索区域图像块缩放至初始帧搜索区域大小,提取特征进行处理,根据式(8)更新位置滤波器。

**步骤 9** 在新的一帧中根据式(12)截取不同尺度的图像块,然后缩放至初始目标框大小,求它们的特征描述子,更新尺度滤波器。

**步骤 10** 判断是否到最后一帧,若是则停止跟踪;若否则跳转到步骤 3。

### 4 实验分析

为了验证本文算法的有效性,不仅在 OTB-2015 数据集上进行了仿真实验,还将该算法移植到 manifold 妙算微型计算机上进行了跟踪实验。

#### 4.1 实验需求

在 OTB-2015 数据集上的仿真实验,个人电脑操作系统为 Windows10 64 bits, CPU Inter Core i3-3240, 主频为 3.20 GHz, 内存为 8 GB。manifold 妙算微型计算机的跟踪实验,需要 Ubuntu 系统,其中配置好

了 ROS 和 OpenCV 环境,准备好高清摄像头。该算法中重要参数取值如下:更新率  $\eta = 0.02$ , 尺度因子  $a = 1.02$ , 尺度大小  $S = 33$ , 异常检测阈值为 10, 距离步长  $\Delta T = R_d/3$ , 角度步长  $\Delta\theta = \pi/8$ , 其他参数和文献[7]保持一致。

#### 4.2 实验仿真

##### 4.2.1 评价标准

在 OTB-2015 数据集上进行仿真实验,该数据集有 100 组视频序列,这些视频序列涵盖了跟踪过程中可能出现的各种干扰因素,包括遮挡、尺度变化、快速移动、光照、模糊等。算法性能以精度图、成功率图和 FPS(每秒处理帧数)的形式显示出来。

精度图显示了中心位置误差在某一特定阈值内的视频序列帧数所占视频序列总帧数的比例。本文实验中选取的阈值为 20 个像素单位。其中,中心位置误差计算公式如下:

$$err_{CLE} = \sqrt{(x - x_0)^2 + (y - y_0)^2} \quad (14)$$

式中,  $(x, y)$  为跟踪算法预测的目标中心的坐标,  $(x_0, y_0)$  为真实的目标中心的坐标。

成功率图显示了目标边界框的重叠率大于某一特定阈值的视频序列帧数占视频序列总帧数的比例。本文实验中选取的阈值为 0.5。重叠率的计算公式如下:

$$S = \frac{|r_t \cap r_a|}{|r_t \cup r_a|} \quad (15)$$

其中,  $r_t$  为跟踪算法估计的目标边界框,  $r_a$  为真实的目标边界框,  $| \cdot |$  代表区域中的像素个数,  $\cap$  和  $\cup$  分别代表交集和并集运算符。

FPS 表示算法每秒处理视频序列的帧数,反映了跟踪算法的运行速度,FPS 越大,跟踪算法运行速度越快,实时性越好。其计算公式如下:

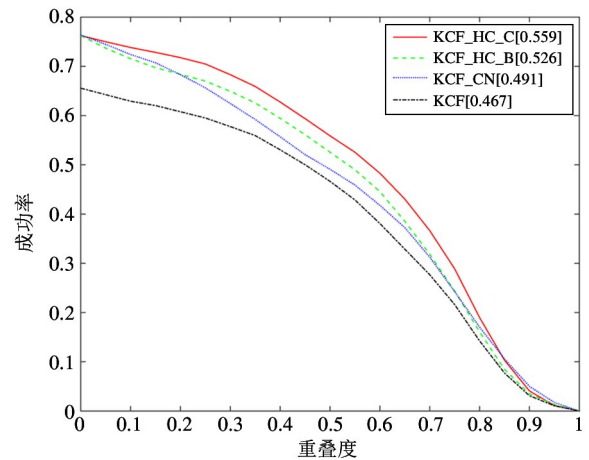
$$FPS = n/t \quad (16)$$

其中,  $n$  代表视频序列的总帧数,  $t$  代表算法运行的总时间。

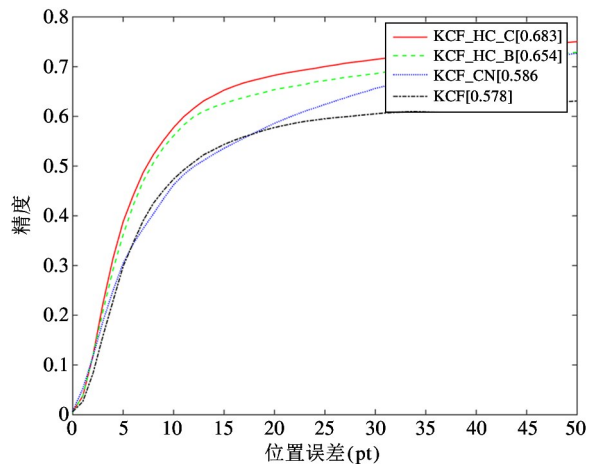
##### 4.2.2 特征融合方式选择实验

为了验证特征融合的有效性,对仅用 FHOG 特征、仅用 CN 特征、将 FHOG 特征和 CN 特征串行融合、将 FHOG 特征和 CN 特征并行融合的 4 种情况进行了对比,测试了 4 种情况在 OTB-2015 数据集上

的 73 组彩色视频序列的整体性能,得到成功率图和精度图如图 9 所示。其中 KCF\_HC\_C 将 FHOG 特征和 CN 特征串行融合,其成功率和精度分别为 55.9% 和 68.3%;KCF\_HC\_B 将 FHOG 特征和 CN 特征并行融合,其成功率和精度分别为 52.6% 和 65.4%;KCF\_CN 仅使用 CN 颜色特征,其成功率和精度分别为 49.1% 和 58.6%;KCF 是传统 KCF 目标跟踪算法,仅使用 HOG 特征,其成功率和精度分别为 46.7% 和 57.8%。从以上成功率和精度可看出,特征融合相较于单一特征效果更好,串行融合的效果也优于并行融合的效果,因此本文算法使用串行融合的方式融合 FHOG 特征和 CN 特征。



(a) 成功率图



(b) 精度图

图 9 KCF 使用不同特征的成功率图和精度图

##### 4.2.3 定量分析

为了对比改进后的算法和原 KCF 目标跟踪算



法的整体性能,测试了两种算法在 OTB-2015 数据集上的 73 组彩色视频序列的整体性能,同时对比了 CSK 和 Struck 两种流行算法及两种改进的 KCF 算法,其中 KCF\_OCC\_CD 是文献[15]中的改进算法,KCF\_Rmix 是文献[16]的改进算法,得到整体的成功率图和精度图如图 10 所示。同时也测试了本文改进算法和原算法在这 73 组彩色视频序列的 11 种不同属性下的精度和成功率,如表 2 所示。

分析图 10 精度图和成功率图可知,相较于传统 KCF 算法和其他流行算法,本文算法在成功率和精度上都是最优的。本文算法成功率为 58.1%,精度为 71.3%;传统 KCF 目标跟踪算法成功率为 46.7%,精度为 57.8%。其中成功率提升了 11.4%,精度提升了 13.5%,由此可见,本文算法的整体性能得到了提升。对比表 2 中传统 KCF 算法和本文改进算法

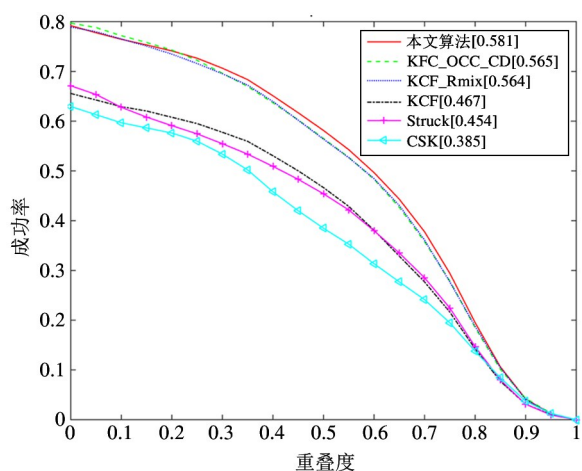
表 2 两种算法在 11 种属性下的精度和成功率

属性	KCF		本文算法	
	成功率	精度	成功率	精度
LR(低分辨率)	0.330	0.522	0.400	0.633
BC(背景混乱)	0.529	0.647	0.628	0.716
OV(出视野)	0.216	0.284	0.439	0.549
IPR(平面内旋转)	0.473	0.606	0.561	0.689
FM(快速移动)	0.332	0.383	0.637	0.772
MB(运动模糊)	0.334	0.385	0.668	0.732
DEF(非刚性形变)	0.399	0.534	0.428	0.609
OCC(遮挡)	0.418	0.540	0.485	0.657
SV(尺度变化)	0.306	0.493	0.447	0.684
OPR(平面外旋转)	0.426	0.552	0.495	0.634
IV(光照变化)	0.508	0.662	0.584	0.745

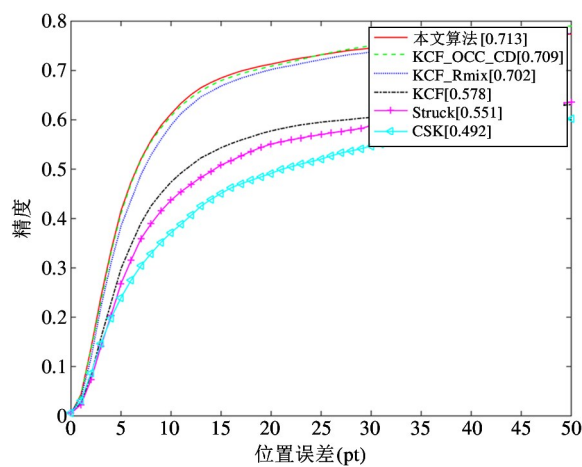
的成功率和精度发现,本文算法在各个属性下的成功率和精度较传统 KCF 目标跟踪算法都有不同程度的提升。由此可知,本文提出的算法能更好地适应各种跟踪场景,其中在尺度变化(SV)、运动模糊(MB)、快速移动(FM)和出视野(OV)的属性下的提升尤为明显。在 SV 的属性下成功率和精度分别提升了 14.1% 和 19.1%,体现了改进算法中尺度滤波器的有效性;在 MB 的属性下成功率和精度分别提升了 33.4% 和 34.7%,体现了改进算法中特征融合的有效性;在 FM 的属性下成功率和精度分别提升了 30.5% 和 38.9%,在 OV 的属性下成功率和精度分别提升了 23.6% 和 26.5%,体现了改进算法中自校正机制的有效性。

为了进一步对比改进算法和原算法在遮挡、尺度变化等单个视频序列的性能,从 OTB-2015 数据集中挑选部分具有挑战性的视频序列,这些序列包含遮挡、尺度变化等干扰因素。得到两种算法在各视频序列的精度和成功率如表 3 所示。

从表 3 可知,在这些具有挑战性的单个视频序列中,改进算法的精度和成功率较传统 KCF 算法而言取得了巨大的提升,其精度和成功率的平均值提升了近一倍,说明本文改进算法在目标遇到遮挡、尺度变化等干扰因素时的跟踪效果明显优于传统 KCF 目标跟踪算法,解决了传统 KCF 目标跟踪算法在这些干扰因素下跟踪不力的问题。



(a) 成功率图



(b) 精度图

图 10 本文算法和原算法的成功率图和精度图

表 3 两种算法在各视频序列具体精度和成功率

视频序列	KCF		本文算法	
	成功率	精度	成功率	精度
Blurowl	0.228	0.228	0.990	0.992
Coke	0.722	0.838	0.918	0.952
Couple	0.243	0.257	0.821	1.000
Human5	0.236	0.265	0.342	1.000
Jogging	0.225	0.235	0.971	0.977
Shaking	0.014	0.025	0.940	0.970
Skating1	0.362	1.000	0.405	1.000
Soccer	0.393	0.793	0.462	0.939
Tiger1	0.857	0.851	0.966	0.960
Walking2	0.380	0.440	0.410	0.674
平均值	0.366	0.493	0.723	0.946

表 4 是原算法与各改进算法的 FPS 数据,从表中可看出,原 KCF 算法处理速度高达 163 FPS,而其他改进算法处理速度都没有达到 100 FPS。本文改进算法在性能上最优,虽然自校正机制的计算量较大,实时性较其他算法较差,只有 40 FPS,但也能基

本满足实时性的要求。

表 4 各改进算法和原算法的 FPS

	KCF	本文算法	KCF_OCC_CD	KCF_Rmix
FPS	163	40	64	77

#### 4.2.4 定性分析

在 OTB-2015 数据集中选取 bird1、jogging、carscale、shaking 几组视频序列对本文算法进行定性分析,并与传统 KCF 目标跟踪算法进行比较。carscale 视频序列中主要包含尺度变化(SV),jogging 视频序列中主要是目标被遮挡(OCC),bird1 视频序列中主要是目标发生非刚性形变(DEF),shaking 视频序列中主要发生了光照变化(IV)。实验结果如图 11 所示。

图 11 中线条较细的框是 KCF 目标跟踪算法的跟踪框,线条较粗的框是本文算法的跟踪框。

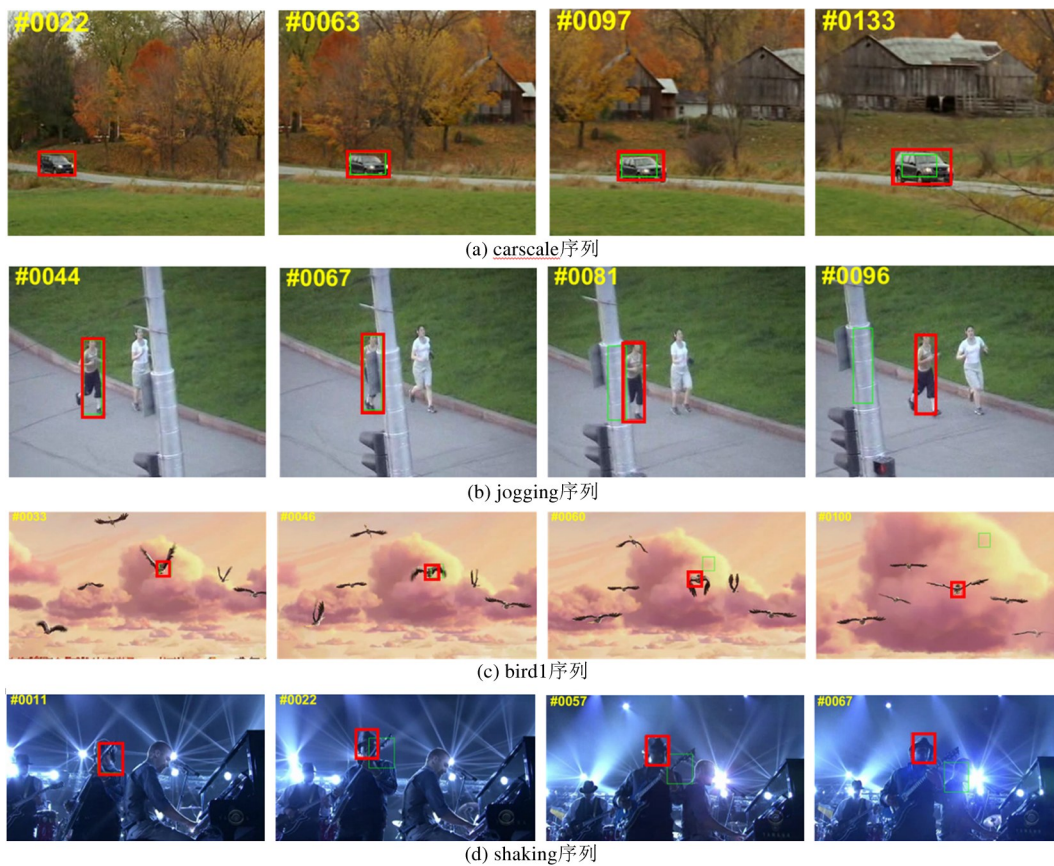


图 11 两种算法在 4 组视频序列跟踪对比图

在图 11(a) 的 carscale 视频序列中,随着镜头的拉近,目标尺度越来越大,传统 KCF 目标跟踪算法在目标尺度变化时不能自适应地改变跟踪框大小,目标框大小始终不变,在后续帧中只能提取局部信息;本文改进算法则能自适应地改变跟踪框大小,随着目标尺度变大,跟踪框也跟着变大,在后续帧中能较为准确地提取全部信息并且能准确跟踪到目标。

在图 11(b) 的 jogging 视频序列中,第 67 帧中目标被电线杆遮挡,第 81 帧目标重新开始出现在视线里,传统 KCF 目标跟踪算法在目标被遮挡后便丢失了目标,当目标重新出现后没有及时跟上,导致跟踪失败。而本文改进算法在目标被遮挡后虽然短暂丢失目标,但在目标重新出现时,跟踪框能及时跟上目标,完成跟踪。

在图 11(c) 的 bird1 视频序列中,目标发生了非刚性形变,第 46 帧中传统 KCF 目标跟踪算法目标框因受到干扰已经偏离目标,第 60 帧以后已经完全跟丢目标,而本文算法在目标发生非刚性形变时仍能较好的跟踪到目标。

在图 11(d) 的 shaking 视频序列中,包含着光照变化和背景杂乱的干扰因素,第 22 帧中,受干扰因素的影响,传统 KCF 目标跟踪算法的目标框已经开始逐渐偏离目标,第 67 帧可以看到已经完全跟丢目标,而本文算法在该过程中一直保持着良好的跟踪效果。

以上定量分析和定性分析的结果证明了本文算法的有效性,解决了传统 KCF 目标跟踪算法在处理目标尺度变化、遮挡等干扰下跟踪不力的问题。本文算法较传统 KCF 目标跟踪算法鲁棒性更好,有着更好的跟踪性能。

### 4.3 manifold 妙算微型计算机跟踪实验

manifold 妙算微型计算机是下一代高性能机器人的“大脑”,手掌大小的机身集成了性能出众的计算能力,其尺寸和接口为 DJI 开发平台所优化,可安装在经纬 M100、M600 飞行平台上,也能无缝兼容 A3、N3 飞行控制器。将本文算法移植到 manifold 妙算微型计算机上进行跟踪实验,利用摄像头获取图像信息并处理,manifold 妙算进行算法的演算,并将跟踪结果显示出来。实验中所用摄像头的参数为

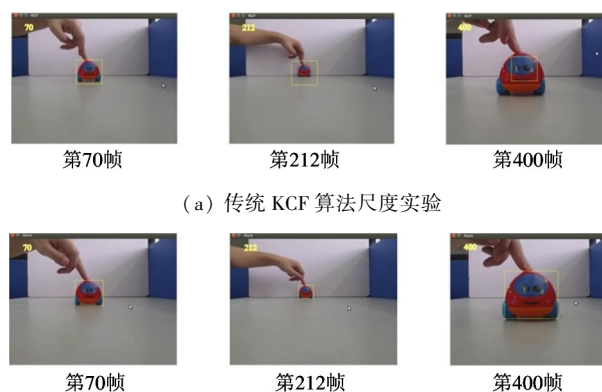
1080P 高清镜头、90°视野、处理帧率 30 FPS、高速 USB 2.0 接口(图 12)。



图 12 manifold 妙算微型计算机及其接口图

#### 4.3.1 尺度变化实验

本实验比较了传统 KCF 目标跟踪算法和本文算法在目标尺度发生变化时的跟踪效果,实验结果如图 13 所示。



(a) 传统 KCF 算法尺度实验

(b) 改进 KCF 算法尺度实验

图 13 两种算法尺度变化实验图

从图中可看出,传统 KCF 目标跟踪算法进行跟踪实验时,第 212 帧目标与摄像头距离拉远,目标尺度变小,此时目标框区域包含过多的背景信息,滤波器学习到太多的背景信息,导致跟踪性能下降;而第 400 帧目标与摄像头的距离拉近,目标尺度变大,由于目标框是固定大小的,此时目标框区域提取的特征只是局部区域的特征,导致特征提取不完全,也会使跟踪性能降低。而本文提出的改进算法在目标尺度变化时,目标框能自适应地变化,提取到较为准确的特征,从而能够精确地跟踪到目标,提高算法性能。尺度自适应实验视频来自 <https://v.qq.com/x/page/b3135k3aoz8.html>

#### 4.3.2 遮挡实验

本实验比较了目标在被遮挡的情况下传统 KCF 目标跟踪算法和本文算法的跟踪效果,实验结果如图 14 所示。

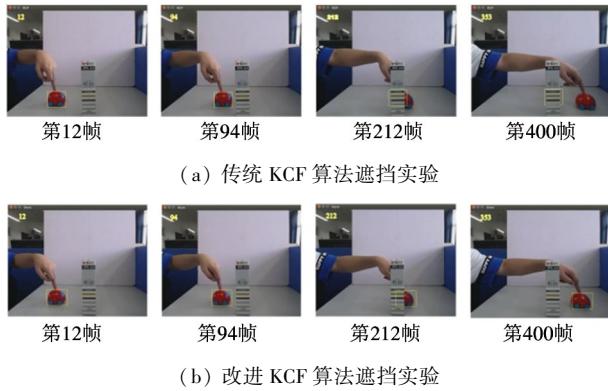


图 14 两种算法遮挡实验图

从图中可看出,第 12 帧到第 94 帧目标未受到干扰时,两种算法均能准确跟踪到目标;当目标被遮挡时,目标框移动到遮挡物上,跟踪目标被遮挡物取代,第 212 帧目标逐渐重新出现时,传统 KCF 目标跟踪算法由于缺乏对遮挡情况的处理机制,使得后续帧学习到的是错误的遮挡物信息,随着错误信息的累积,算法把遮挡物当成目标进行跟踪,导致后续帧的跟踪失败;本文算法在目标被遮挡后,能根据响应图的 PSR 值检测出异常,并及时启动自校正机制,当第 212 帧目标逐渐重新出现时,能及时校正目标框到目标上,继续完成后续帧的跟踪。遮挡实验视频来自 <https://v.qq.com/x/page/i3135y049zn.html>

#### 4.3.3 快速移动实验

本实验比较了传统 KCF 目标跟踪算法和本文算法在目标快速移动时的跟踪效果,实验结果如图 15 所示。

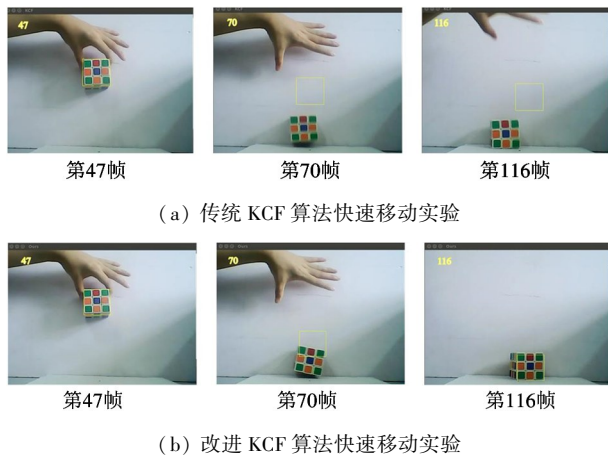


图 15 两种算法快速移动实验图

对于目标自由落体运动,运动过程中速度逐渐变大,第 70 帧速度达到最大值。从图中可看出,传统 KCF 目标跟踪算法实验中,第 70 帧时由于目标快速移动,在当前帧中目标已经移出上一帧的目标区域,在搜索区域中无法搜索到目标,导致后面的跟踪失败;而本文改进算法在目标快速移动导致跟踪发生异常时,能检测出异常,启动自校正机制,扩大搜索范围,重新定位到目标。快速移动实验视频来自 <https://v.qq.com/x/page/s3156qd6d6g.html>

以上实验结果表明,较传统 KCF 目标跟踪算法而言,本文改进算法在目标发生尺度变化、遮挡、快速移动等情况下均能较好地跟踪到目标,提高了算法的性能。

#### 4.3.4 本文算法与其他改进核相关滤波算法对比实验

本实验对比了本文改进算法和文献[15]中改进算法在丢失目标后的实物实验。

图 16 为文献[15]的实物实验,该算法使用窗口滤波进行位置预测,对于运动状态稳定的物体受到遮挡时效果较好,当物体突然改变运动方向或突然加减速时,该算法的效果并不理想。如图 16 所示,当遮挡消失时,目标框已偏离目标且不能重新找回,导致跟踪失败。实验视频来自 <https://v.qq.com/x/page/a3208pm6tgw.html>

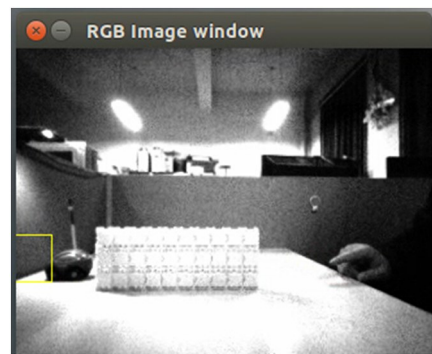


图 16 文献[15]跟踪失败实验图

而本文算法使用的自校正机制实际上相当于扩大了搜索范围,当目标受到干扰时,本文算法会启动自校正机制,通过自校正机制致力于寻找与目标最相似的区域作为目标区域。得利于自校正机制,当跟踪环境正常时,能及时校正目标框到目标上完成

后续帧的跟踪,如图 14 和图 15 所示。相较于文献[15]中的改进算法,本文算法对目标丢失后重新找回的情况效果更好。

## 5 结论

本文在传统 KCF 目标跟踪算法框架的基础上,融合了 CN 颜色特征,引入了自校正机制和尺度滤波器。改进的算法中,多特征融合使得算法在复杂的跟踪场景中有着更好的性能,自校正部分解决了传统 KCF 目标跟踪算法因目标遇到遮挡等干扰因素而导致目标跟踪失败的问题,而尺度滤波器解决了跟踪过程中因目标尺度变化导致跟踪不准确的问题。仿真测试和实物实验结果表明,改进的算法较传统 KCF 目标跟踪算法的跟踪性能有了明显的提升,在目标发生遮挡、尺度变化等情况下,也能准确地跟踪到目标。

### 参考文献

- [ 1 ] 张全,盛妍,吴佐平,等. 公共区域监控视频数据目标特征跟踪定位方法[J]. 自动化与仪器仪表,2020(1):51-54
- [ 2 ] 翟敬梅,刘坤,徐晓. 室内移动机器人自主导航系统设计与方法[J]. 计算机集成制造系统,2020,26(4):890-899
- [ 3 ] 葛宝义,左宪章,胡永江. 视觉目标跟踪方法研究综述[J]. 中国图象图形学报,2018,23(8):1091-1107
- [ 4 ] 孟绿,杨旭. 目标跟踪算法综述[J]. 自动化学报,2019,45(7):1244-1260
- [ 5 ] BOLME D S, BEVERIDGE J R, DRAPER B A, et al. Visual object tracking using adaptive correlation filters [C] // The 23rd IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, USA, 2010: 2544-2550
- [ 6 ] HENRIQUES J F, CASEIRO R, MARTINS P, et al. Exploiting the circulant structure of tracking-by-detection with kernels[C] // Proceedings of the 12th European Conference on Computer Vision, Florence, Italy, 2012: 702-715
- [ 7 ] HENRIQUES J F, CASEIRO R, MARTINS P, et al. High-speed tracking with kernelized correlation filters [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(3): 583-596
- [ 8 ] DALAL N, TRIGGS B. Histograms of oriented gradients for human detection [C] // IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, USA, 2005: 829-836
- [ 9 ] 刘晓悦,王云明,马伟宁. 融合 FHO 和 LBP 特征的尺度自适应相关滤波跟踪算法[J]. 激光与光电子学进展, 2020, 57(4): 330-337
- [ 10 ] Li Y, Zhu J. A scale adaptive kernel correlation filter tracker with feature integration [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(3): 583-596
- [ 11 ] ZOLFAGHARI M, GHANEI-YAKHDAN H, YAZDI M. Real-time object tracking based on an adaptive transition model and extended Kalman filter to handle full occlusion [J]. *The Visual Computer*, 2020, 36(10): 701-715
- [ 12 ] JENKINS M D, BARRIE P, BUGG Y T, et al. Extended fast compressive tracking with weighted multi-frame template matching for fast motion tracking [J]. *Pattern Recognition Letters*, 2016, 69: 82-87
- [ 13 ] DANELLJAN M, KHAN F S, FELSBERG M, et al. Adaptive color attributes for real-time visual tracking [C] // 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, USA, 2014: 1090-1097
- [ 14 ] WU Y, LIM J, YANG M H. Object tracking benchmark [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(9): 1834-1848
- [ 15 ] 陈志旺,王航,刘旺,等. 抗遮挡与尺度自适应的改进 KCF 跟踪算法研究[J]. 控制与决策, 2019, 36(2): 457-462
- [ 16 ] 李国友,纪执安,张凤煦. 一种基于多特征融合的核相关滤波目标跟踪算法[J]. 计算机与应用化学, 2018, 35(12): 1004-1011

## Research on improved multi-scale KCF target tracking algorithm based on features fusion and self-correction

CHEN Zhiwang, LIU Wang

(Key Laboratory of Industrial Computer Control Engineering of Hebei Province,

Yanshan University, Qinhuangdao 066004)

(National Engineering Research Center for Equipment and Technology of Cold Strip Rolling,

Yanshan University, Qinhuangdao 066004)

### Abstract

Traditional kernel correlation filter (KCF) target tracking algorithm uses a single feature, does not monitor the output, and the size of the target box is unchangeable. It can usually cause tracking failures for the target under abnormal conditions such as occluded or scaled. To solve the problems, this paper proposes an improved kernel correlation filter target tracking algorithm. Firstly, the cascade of color name (CN) feature and fast histogram of oriented gradients (FHOG) features are applied to train and update filters. Secondly, the tracking output is monitored by the peak-to-sidelobe ratio (PSR) value. When an abnormal tracking output is detected, a self-correction mechanism is activated to accurately re-track the target. Finally, a scale filter is added to adapt to changes in the target scale. Simulation and physical experimental results verify the effectiveness of the proposed algorithm.

**Key words:** target tracking, correlation filter, abnormal detection, self-correction, scale adaptation, features fusion