

武汉理工大学毕业设计（论文）

基于深度卷积特征的相关滤波算法研究

学院（系）： 自动化学院

专业班级： 自动化 1502 班

学生姓名： 吴 麟

指导教师： 张 健

学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名：

年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于 1、保密口，在 年解密后适用本授权书

2、不保密口

作者签名： 年 月 日

导师签名： 年 月 日

摘 要

为了进一步提高目标跟踪算法在复杂视频场景中的跟踪性能，本文以基于相关滤波的跟踪算法为研究中心，展开了对于核化相关滤波、多层卷积特征、跟踪评价指标等一系列讨论。首先，结合判别式跟踪算法的基本框架，阐述了核化相关滤波算法的基本原理，对算法流程中的 HOG 特征、核回归、模板更新与多通道处理等核心组件做了详细推导。其次，针对核化相关滤波算法在特征提取、尺度评估、模板更新、长时跟踪等方面的缺陷，以多层卷积目标跟踪算法（HCFT）为蓝本，引入了 LMCT 算法中的平均峰值相关能量（APCE），再结合此前的滤波响应峰值，提出了一种高置信度目标检测机制，并将此应用于目标尺度评估、长时跟踪以及目标更新。最后，将多种跟踪算法在标准数据集 OTB50 进行测试，实验结果表明高置信度跟踪算法（HC-HCFT）在跟踪准确率与成功率上较 HCFT、SRDCF、Staple、DSST、KCF 等经典或先进的算法具有更好的性能，且通过 OTB50 全部 11 种属性视频的单项测试，结果表明本文算法具有普遍的适应性。

关键词：计算机视觉；目标跟踪；相关滤波；卷积特征

Abstract

In order to further improve the tracking performance of the visual tracking algorithm in complex video sequences, this paper takes the tracking algorithm based on the correlation filter as the research center, and develops a series of discussions on Kernelized correlation filter, Hierarchical convolution features and tracking evaluation criteria. Firstly, combined with the basic framework of the discriminant tracking algorithm, the basic principles of the Kernelized correlation filter are expounded, and the core components such as HOG feature, kernel regression, template update and multi-channel processing in the algorithm are derived in detail. Secondly, based on the defects of Kernelized correlation filter in feature extraction, scale estimation, template updating and long-term tracking, the High-Confidence Hierarchical Convolutional Features for Visual Tracking (HC-HCFT) is used as the source to introduce the average peak correlation energy (APCE) in the LMCT algorithm. Combined with the previous filter response peak and the average peak correlation energy, we propose a high confidence target detection mechanism, and apply this to the target scale assessment, long-term tracking and target update. Finally, a variety of tracking algorithms are tested in the standard data set OTB50. The experimental results show that the high confidence tracking algorithm (HC-HCFT) is superior to HCFT, SRDCF, Staple, DSST, KCF in tracking accuracy and success rate. The algorithm has better performance, and through the single test of all 11 kinds of attribute videos of OTB50, the results show that the algorithm has universal adaptability.

Key Words: Computer vision; visual tracking; correlation filters; convolution features

目 录

| | |
|----------------------------|----|
| 第 1 章 绪论 | 1 |
| 1.1 研究背景与研究意义 | 1 |
| 1.2 论文研究内容与结构 | 2 |
| 第 2 章 核化相关滤波算法 (KCF) | 4 |
| 2.1 跟踪算法框架 | 4 |
| 2.2 特征表达与提取 | 5 |
| 2.2.1 HOG 特征 | 5 |
| 2.2.2 余弦窗函数 | 7 |
| 2.3 目标搜索策略 | 7 |
| 2.3.1 岭回归 (RLS) | 7 |
| 2.3.2 核方法 (KRLS) | 9 |
| 2.3.3 目标检测 | 10 |
| 2.4 模板更新方法 | 11 |
| 2.5 多通道处理 | 12 |
| 第 3 章 基于卷积特征的相关滤波算法 | 13 |
| 3.1 问题的提出 | 13 |
| 3.2 方案整体设计 | 14 |
| 3.3 多层卷积特征 | 14 |
| 3.3.1 深度卷积网络 | 14 |
| 3.3.2 深度卷积特征 | 16 |
| 3.3.3 多特征融合方法 | 17 |
| 3.4 检测结果置信度 | 18 |
| 3.5 高置信度目标检测 | 19 |
| 3.5.1 目标尺度评估 | 19 |
| 3.5.2 目标丢失再检测 | 20 |
| 3.5.3 高置信度模板更新 | 21 |
| 第 4 章 实验结果及分析 | 23 |
| 4.1 目标跟踪数据集 | 23 |
| 4.2 跟踪算法评价指标 | 23 |
| 4.2.1 准确率 (PR) | 24 |
| 4.2.2 成功率 (SR) | 24 |
| 4.2.3 帧率 (FPS) | 25 |

| | |
|---------------------|----|
| 4.3 实验环境与实验设置 | 26 |
| 4.3.1 软硬件环境 | 26 |
| 4.3.2 实验设置方案 | 26 |
| 4.4 算法仿真与结果分析 | 27 |
| 4.4.1 定性分析 | 27 |
| 4.4.2 整体分析 | 29 |
| 4.4.3 属性分析 | 30 |
| 第 5 章 结论 | 36 |
| 5.1 研究工作总结 | 36 |
| 5.2 未来工作展望 | 36 |
| 致 谢 | 37 |
| 参考文献 | 38 |
| 附 录 | 40 |

第 1 章 绪论

1.1 研究背景与研究意义

作为计算机视觉（Computer Vision, CV）领域的核心问题之一，视频目标跟踪（Visual Object Tracking）自诞生以来饱受研究者的关注。目标跟踪的基本任务可描述为：根据视频第一帧中目标的位置和状态，对视频序列进行分析，进而推断目标在后续帧中的位置和状态。它涉及的内容十分广泛，包括图像处理、模式识别、概率统计、机器学习等多种学科的理论知识，同时在机器人与无人驾驶、智能视频监控、医疗应用与生物研究、远程协作与互动游戏等众多领域中有广阔的应用场景。

目标跟踪问题的研究起源于 20 世纪 50 年代，1955 年由 Wax Nelson^[1]率先提出了目标跟踪的基本概念，此后在 1960 年 Kalman^[2]提出了著名的卡尔曼滤波器算法，进一步促进了目标跟踪技术的发展，至 1964 年，Sitter^[3]等人提出了基于贝叶斯理论的目标轨迹与轨迹数据关联，为目标跟踪技术的成熟奠定了坚实的基础。近年来，随着 TPAMI、CVPR、ICCV、ECCV 等计算机视觉顶级期刊与会议的推动与发展，学者们对视频目标跟踪问题的研究已经相当深入。不论是单目标跟踪抑或是多目标跟踪，不论是生成式模型还是判别式模型，目标跟踪领域百家争鸣，诸多优秀的跟踪算法被提出并不断完善^[4]。

有别于视频目标跟踪的其他分支，本文主要以基于相关滤波^[5]（Correlation Filter, CF）框架的目标跟踪为主要研究对象，这也是当前目标跟踪领域研究与应用的热点问题之一。在近几年的跟踪研究中，基于 CF 的跟踪算法在速度和性能上的优势独占鳌头，其通过极高的效率解决岭回归问题完成了对目标的判断。值得一提的是，在 2014 年由 Henriques^[6]等人提出的核化相关滤波算法（Kernelized Correlation Filter, KCF），利用循环矩阵的原理，通过离散傅里叶变换将 CF 的求解过程转换到复域，提取方向梯度直方图（Histogram of Oriented Gradient, HOG）作为模型特征，同时提出了解决多特征融合的方法，大大提高了目标跟踪的效果，一度成为基于 CF 框架的基准（Baseline），如后来的 DSST^[7]、MKCF^[8]、RPAC^[9]等都是基于 KCF 的改进。

DSST 由林雪平大学 CV 实验室 Martin 等人于 2014 年提出，其针对 KCF 算法中不能实时评估跟踪目标尺寸的缺陷，在 KCF 的位置滤波器基础上引入了尺度滤波器（Scale Filter）的概念，且这两种滤波器互不影响，因此 DSST 具有良好的移植性，可以和任何不具备尺度自适应的目标跟踪算法相结合，进而提高跟踪效果。

MKCF 由中科院自动化所 Ming Tang 等人于 2015 年提出，针对 KCF 或其他基于核相

关滤波算法仅使用单核的局限，引入了一种多核融合策略，利用了各种特征的不变性判别功率谱，进一步提高了跟踪器的性能。另一特色是，论文中还提出了一种基于最优二分搜索和快速特征估计的尺度估计方法，比上述 DSST 算法更为高效。

RPAC 由南洋科技大学 Ting Liu 于 2015 年提出，主要针对 KCF 以及传统的基于 CF 的目标跟踪算法不能很好解决局部遮挡的问题，其首先将目标分解，随之使用多个 CF 来跟踪目标的各个部分。同时，作者提出一种新的权重分配方法，即根据 SCCM 算子^[9]检测目标被遮挡的情况，并据此确定各个 CF 的权重。此外，作者也考虑了多 CF 融合产生的噪声，并制定了相应的抑制策略，因此 RPAC 具有强大的跟踪鲁棒性。

纵观近年来的基于 CF 的目标跟踪技术，在无数研究者的推动下，从各个角度，利用各种方法，提高跟踪的精度和速度，已经达到一个较高的水平。除了上文提到了尺度(Scale)、多核(Multi Kernel)、基于部件(Part-based)、还有颜色(Color)、多特征(Multi Feature)、多模板(Multi template)、长时跟踪(Long-term)等一系列研究方向^①。

尽管目标跟踪技术已有丰富的研究基础，但在实际应用时常常会遇到场景复杂，种类繁多，以及由物体本身运动特性产生的遮挡、形变、位置骤变等情况，使得算法的跟踪精度下降，这也是目前跟踪领域最大的挑战。随着深度学习技术在计算机视觉领域的不断发展，基于深度神经网络(Deep Neural Network, DNN)的目标跟踪算法也逐渐成为研究热点。其中，卷积神经网络(Convolutional Neural Network, CNN)强大的非线性处理能力，可以轻松胜任传统方法难以解决的分类型问题。此外，卷积特征比传统图像特征具有更为丰富的语义信息，故而将 CNN 运用在目标跟踪领域正成为研究者们一个新的思路。本文也将以此为出发点，对 CF 领域的经典之作 KCF 进行学习研究，并针对其采用的 HOG 特征之局限性，重新提取卷积特征来获得更多目标信息，以期在更为复杂的视频场景下取得更好的跟踪效果。

1.2 论文研究内容与结构

本文核心内容主要由三部分组成。首先，引介经典的 KCF 算法，侧重于分析其算法框架及主要贡献；其次从特征提取的角度，将论文中的 HOG 特征用 CNN 特征替代，侧重于阐述 CNN 特征的原理与使用方式；最后，介绍目标跟踪常见的评价指标，并采用当前主流的开源数据集进行算法测试，并根据测试结果分析两种特征的差别。下面按照第一章绪论至终章结语的顺次，详细介绍本文各章节的具体内容。

^① Tracking Benchmark for Correlation Filters: <https://github.com/HEscop/TBCF>.

第一章：对目标跟踪的意义价值进行分析，对目标跟踪领域的历史进行溯源，对目标跟踪的当前发展进行跟进。继而聚焦基于相关滤波的目标跟踪算法，对其中典型的算法进行详细介绍，分析当前研究之不足，并顺势引出深度学习技术。分析了深度学习应用于目标跟踪的广阔前景，进而确立本文的研究路线。

第二章：从判别式跟踪算法框架入手，继而从特征表达与提取、目标搜索策略与模板更新方法三个主要环节详细说明 KCF 的算法流程。其中特征表达与提取包括 HOG 特征分析、余弦窗函数；目标搜索策略包括岭回归、核方法以及目标检测三个方面；模板更新方法自成一节；特别地，对 KCF 的重要贡献——多特征融合方法进行介绍。

第三章：针对 HOG 特征的局限以及 KCF 不具备目标尺度评估能力等缺陷，提出了一种高置信度目标跟踪算法 HC-HCFT，继而分三个步骤对此进行介绍。首先介绍了多层卷积特征提取，包括深度卷积网络、深度卷积特征与多特征融合方法三个部分；其次，介绍了目标检测高置信度指标；最后，将高置信度指标融入目标检测，构成目标尺度评估、目标丢失再检测以及高置信度模板更新三个部分。

第四章：首先介绍了跟踪算法主流的测试数据集以及常用的评价指标，其中数据集包括 OTB 与 VOT、评价指标包括准确率、成功率与帧率；其次，对算法实验的环境与实验的设置进行说明；最后在 OTB 数据集上测试了包括 HC-HCFT 在内的 8 种判别式跟踪算法，并对实验结果进行定性分析、整体分析与属性分析。

第五章：对本文的研究内容与实验结果进行总结，对未来的工作进行展望。

第 2 章 核化相关滤波算法（KCF）

2.1 跟踪算法框架

按照近年来主流的分类方法，当前的目标跟踪算法可以分为生成式和判别式两种类型。生成式方法通过学习跟踪目标的外观模型，以最小匹配误差为优化准则在搜索区域确定目标的位置及状态，如局部无序跟踪 LOT^[10]、增量视觉跟踪 IVT^[11]、基于分解的目标跟踪 VTD^[12]等算法。而判别式方法把跟踪问题转化为一个目标与背景的二分类问题，通过提取目标特征并以此训练一个分类器，从而实现对目标进行跟踪，如上述 KCF、DSST、RPAC 等基于相关滤波的跟踪算法都隶属于判别式跟踪框架。

本文为了深入理解 KCF 的核心思想，首先引介判别式目标跟踪算法的基本框架，其主要由特征提取、目标搜索、模板更新三个环节构成，如图 2.1 所示。

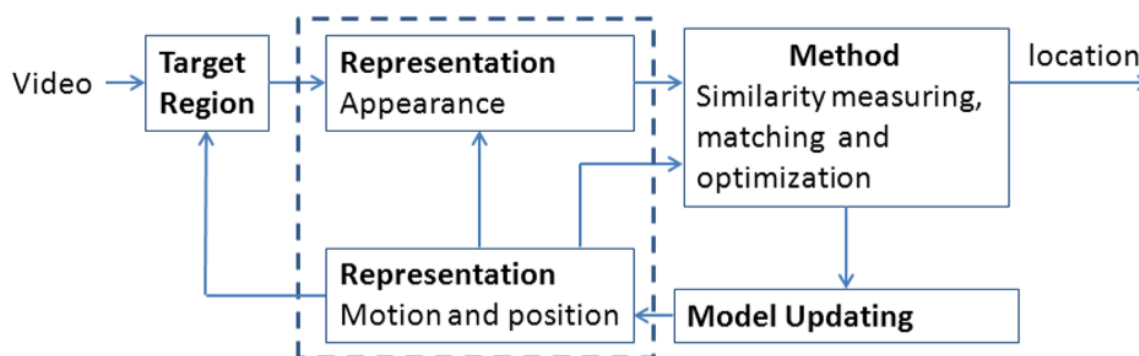


图 2.1 判别式跟踪算法基本框架

其中，特征提取环节负责对目标外观进行抽象的描述，为后续目标搜索环节创造条件。其提取的特征不仅需要具有代表跟踪目标的能力，基于实时性的考虑，也需要具有较好的运算速度。因此，所提取的特征可以是简单的颜色特征、灰度特征，也可以是稍显复杂的形状特征、纹理特征或是多种特征的组合等等。

目标搜索环节作为跟踪算法最为核心的部分，负责根据既往的跟踪目标模板，在新的视频搜索区域中寻找目标最新的位置。对相关滤波类算法，目标跟踪模板一般利用机器学习方法，通过一定的样本训练出分类参数，以此构建目标与背景的分类器。

模板更新环节是根据既往的跟踪目标模板与当前的目标状态，重新计算模板参数的过程。之所以需要历史模板，是因为目标在跟踪过程中可能产生诸多非正常状态的变化，如形变、遮挡等。而结合历史模板之后，可以有效地抑制目标的漂移现象。

这样，当输入一段视频并指定跟踪对象以后，经过这三个环节的处理，则目标跟踪便可以不断迭代下去。同样的，KCF 算法也是判别式方法的一部分。有了上面的基础，所以如图 2.2 所示的 KCF 算法基本框架不难理解。

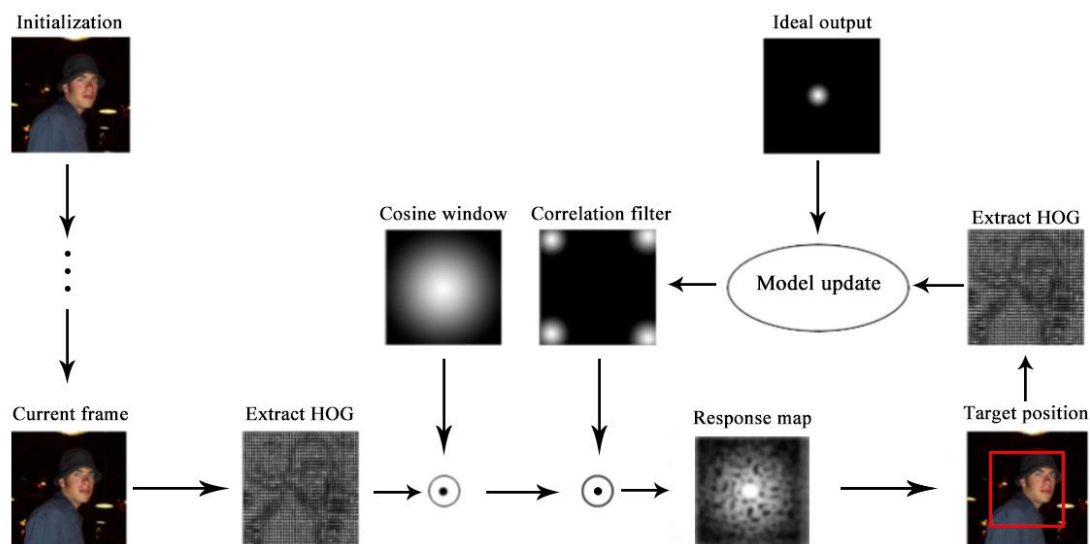


图 2.2 相关滤波跟踪算法基本框架

KCF 的基本框架同样包含三个环节。首先，在初始化的基础上，从当前帧提取 HOG 特征，并利用余弦窗函数对特征进行滤波以淡化搜索空间的边界噪声，达到突出跟踪目标的效果。随后，利用该特征与目标模板参数进行相关滤波运算，得到响应图，进而估计目标的当前位置。此时，目标位置已经变更，则采用模板更新策略再次计算模板参数（包括目标模板与相关滤波器模板），以备下一帧中目标的定位。

本章的后续部分将从这三个环节，详细介绍 KCF 的基本算法。其中，特征表达与提取由 HOG 特征与余弦窗滤波两部分组成；目标搜索策略由岭回归、核方法、目标检测三个部分组成；模板更新方法单独成一节。

2.2 特征表达与提取

2.2.1 HOG 特征

方向梯度直方图特征（Histogram of Oriented Gradient, HOG）是 CV 领域一种常用的形状与边缘描述算子。其核心思想为，一副图形中梯度或边缘的密度分布能够较好地描述检测目标的形状与轮廓信息。如图 2.3 所示，其提取过程是首先将图像划分为小的区块（patch），也称细胞单元（cell），然后计算每个区块的 HOG，最后将所有区块的 HOG 组合起来即可获得目标的 HOG 特征。与其他特征相比，HOG 特征保持了几何与光照不变性，

计算复杂度较低，可满足目标跟踪实时性的要求。

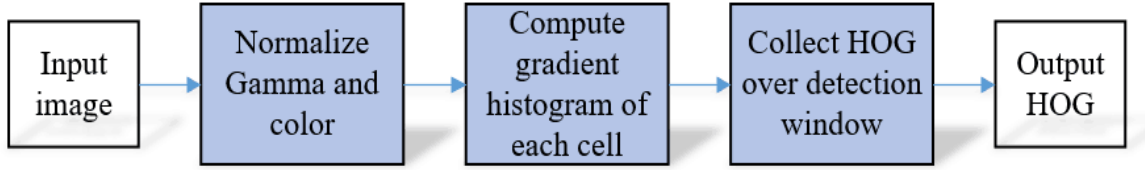


图 2.3 HOG 特征提取的主要流程

（1）标准化 Gamma 与颜色

将检测目标进行灰度化处理，并采用 Gamma 变换对图像进行标准化处理，因为在图像的纹理强度中，局部的表层曝光贡献的比重较大，故而这种压缩处理可以有效地减少图像局部阴影和光照的影响。

（2）统计区块的梯度直方图

每个像素的梯度分为水平和垂直两个部分，设 $I(i, j)$ 为图像中 (i, j) 处像素点的值，则其水平梯度与垂直梯度如下

$$G_x(i, j) = I(i+1, j) - I(i-1, j) \quad (2.1)$$

$$G_y(i, j) = I(i, j+1) - I(i, j-1) \quad (2.2)$$

则 (i, j) 像素点处的梯度幅值与梯度相位分别为

$$G(i, j) = \sqrt{G_x^2(i, j) + G_y^2(i, j)} \quad (2.3)$$

$$\alpha(i, j) = \tan^{-1}\left(\frac{G_y(i, j)}{G_x(i, j)}\right) \quad (2.4)$$

由于图像被我们划分为 $a \times b$ 个区块，设每个区块包含 $n \times n$ 个像素，采用 m 个统计堆栈（Bin）来表达该区块的信息。根据每个像素的梯度相位，进而将该像素归并到某个 Bin 之内。而梯度幅值作为其归并权值，累加到梯度直方图中，形成长度为 m 的一维向量。

（3）收集 HOG 特征

经典的方法是将每个区块提取出的 HOG 特征首尾相连（或称串联），形成长度为 $a \times b \times m$ 的一维向量。但在 KCF 中，并没有将每个区块的 HOG 特征串联，而是将其并联，最终得到 31 通道的 HOG 特征。其提取方法为：首先，将 3 通道的彩色图片分成 $a \times b$ 个区块，然后将梯度方向划分为 9 个，即 $m = 9$ 。其次，统计每个区块在某个梯度方向的信息，得到维度为 $a \times b$ 的特征块。最后，由于梯度方向有 9 个，彩色图片通道数（RGB）有 3 个，

故梯度方向共有 3×9 个，因此整个彩色图片的 HOG 特征共有 27 个通道。此外，由于 KCF 采用的是 fhOG 特征（HOG 特征的改进版本，在 HOG 特征的基础上增加了 4 个纹理特征），故一共需要 31 个通道才能存储所有特征信息。

2.2.2 余弦窗函数

为了降低图像样本的边缘噪声对模板训练的影响，KCF 采用了一种图像滤波方法——余弦窗（又称汉宁窗），其思想是在目标原始像素上乘以余弦窗使得样本边缘的像素接近于零。对长度为 N 的一维余弦窗 $H(N)_{N \times 1}$ ，其各个元素定义为

$$h(i) = \frac{1}{2} (1 - \cos(2\pi \frac{i}{N-1})), i = 0, 1, \dots, N \quad (2.5)$$

而对于 $m \times n$ 的图像样本，余弦窗矩阵可表示为

$$W_{\cos} = H(m) \times H^T(n) \quad (2.6)$$

其中， W_{\cos} 是 $m \times n$ 维的矩阵，余弦窗滤波前后的效果对比如图 2.4 所示。



图 2.4 余弦窗滤波前后效果对比

由图中可见，相比于原始图像，经过余弦窗滤波后的图像边缘都被淡化，使得跟踪目标得以凸显。这种处理方式对于实际的跟踪问题非常有效，尽管它牺牲掉一部分背景信息，但却可以显著地提高跟踪精度。

2.3 目标搜索策略

2.3.1 岭回归（RLS）

MOSSE^[5]是首次运用相关滤波并取得巨大成功的案例。如图 2.5 所示，MOSSE 的基本思路是通过训练得到一个滤波器模板 h ，然后将目标的搜索区域 f （即一幅包含跟踪目标

的图像)，与滤波器模板进行卷积运算，以此获得目标的响应图 g 。

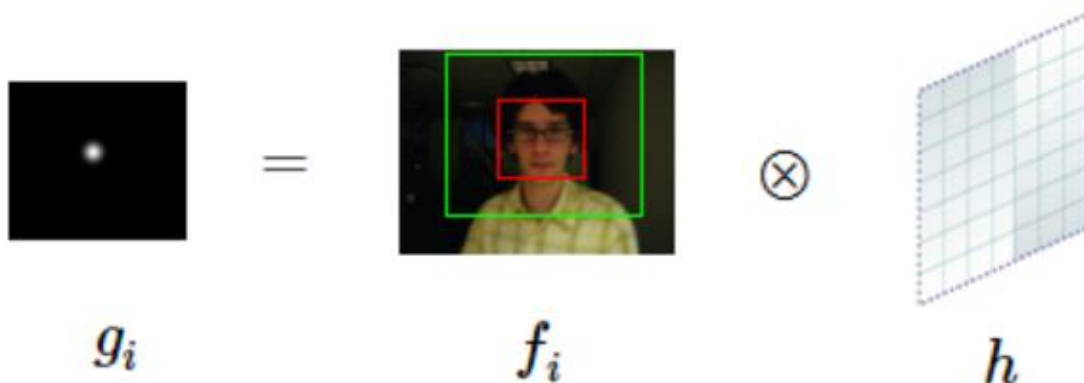


图 2.5 相关滤波器算法基本思路

为了便于计算，这里引入傅里叶变换，根据卷积定理有

$$\mathcal{F}(g) = \mathcal{F}(f \otimes h) = F(f) \cdot F^*(h) \quad (2.7)$$

其中 ‘ $*$ ’ 表示共轭。令 $F(f) \triangleq F, F(g) \triangleq G, F(h) \triangleq H$ ，则有

$$H^* = \frac{G}{F} \quad (2.8)$$

H^* 即是所求滤波器，MOSSE 提出的方法是最小平方和误差，即

$$\min_{H^*} \sum_i |H^* F_i - G_i|^2 \quad (2.9)$$

如果将 $H^* F_i$ 理解为样本 x_i 的函数值 $w^T x_i$ ， G_i 则为 x_i 的理想值 y_i ，此时上式可变为

$$\min_w \sum_i (f(x_i) - y_i)^2 \quad (2.10)$$

其中， $f(x_i) = w^T x_i$ 。可见，MOSSE 采用的核心计算方法是标准的最小二乘。而 KCF 在此基础上加了正则化参数 λ ，这样的好处可以有效防止过拟合，提高模型泛化性能。

$$\min_w \sum_i (f(x_i) - y_i)^2 + \lambda \|w\|^2 \quad (2.11)$$

式 (2.11) 即为岭回归，或称正则化最小二乘 (Regularized Least Squares., RLS) 回归。对上式求偏导，可得 RLS 的闭式解如下

$$w = (X^T X + \lambda I)^{-1} X^T y \quad (2.12)$$

其中，矩阵 X 的每一行为 x_i ， y 为 y_i 构成的列向量， I 为单位矩阵。由于上式包含矩阵求逆的过程，其时间复杂度为 $O(n^3)$ ，这将极大限制目标跟踪的实时性，而 KCF 采用循环矩阵，将时域的矩阵求逆转为至复域中矩阵的点乘。这一思想始现于 Henriques 在 2012 年

提出的 CSK 方法^[13]。具体而言，首先将式（2.12）转换到复域，可得

$$w = (X^H X + \lambda I)^{-1} X^H y \quad (2.13)$$

其中， X^H 为 X 的共轭转置，等价于 $(X^*)^T$ 。由于任何循环矩阵都能在复域使用离散傅里叶矩阵进行对角化，即

$$X = C(x) = F \text{diag}(\hat{x}) F^H \quad (2.14)$$

其中， x 为基向量， $C(\cdot)$ 为循环矩阵， \hat{x} 为 x 的傅里叶变换， $\text{diag}(\cdot)$ 为对角矩阵， F 为与 x 无关的常数矩阵，进一步可得

$$X^H X = F \text{diag}(\hat{x}^* \odot \hat{x}) F^H \quad (2.15)$$

其中，‘ \odot ’表示点乘，将式（2.14）、（2.15），代入式（2.13）可得分类器参数如下

$$\hat{w} = \frac{\hat{x} \odot \hat{y}}{\hat{x}^* \odot \hat{x} + \lambda} \quad (2.16)$$

最终，根据式（2.16）在搜索区域内检测目标的位置。因为矩阵点乘运算和离散傅里叶变换（Discrete Fourier Transform, DFT）的时间复杂度分别为 $O(n)$ 和 $O(n \log n)$ ，所以，当数据量激增后，KCF 的求解复杂度将显著降低，有利于提升跟踪速度。

2.3.2 核方法（KRLS）

KCF 之所以称为“核化”相关滤波，便是其引入了核函数理论。核函数是模式识别领域常用的技巧，它的核心理论为：在低维空间线性不可分的模式，将其映射到高维空间则可以实现线性可分。因此，分类器参数 w 可以表达为样本的线性组合

$$w = \sum_i \alpha_i \varphi(x_i) \quad (2.17)$$

其中， $\varphi(\cdot)$ 为非线性映射函数， α 是样本的组合权重。此时，只要求解出 α ，便可以获得分类器参数 w 。也可以将 α 理解为对偶空间^[14]（Dual space）的参数，而 w 是时域空间的参数。首先，核函数 $\kappa(\cdot)$ 具有如下定义

$$\kappa(x, x') = \varphi^T(x) \varphi(x') \quad (2.18)$$

其中， x' 是与 x 相区别的另一样本。此时，回归函数可以转化为如下形式

$$f(z) = w^T z = \sum_i \alpha_i \kappa(z, x_i) \quad (2.19)$$

引入核函数的岭回归具有如下闭式解

$$\alpha = (K + \lambda I)^{-1} y \quad (2.20)$$

其中， K 为样本核函数值构成的矩阵，即 $K_{ij} = \kappa(x_i, x_j)$ 。显然，若样本数为 n ，则 K 的维度为 $n \times n$ 。

在相关滤波中常用的几种核函数分别为线性核、多项式核、高斯核，分别如式 (2.21)、(2.22) 与 (2.23) 所示

(1) 线性核

$$\kappa^{xx'} = \mathcal{F}^{-1}(\hat{x} \odot \hat{x}'^*) \quad (2.21)$$

(2) 多项式核

$$\kappa^{xx'} = \left(\mathcal{F}^{-1}(\hat{x} \odot \hat{x}'^*) + a \right)^b \quad (2.22)$$

(3) 高斯核

$$\kappa^{xx'} = \exp \left(-\frac{1}{\sigma^2} \left(\|x\|^2 + \|x'\|^2 - 2\mathcal{F}^{-1}(\hat{x} \odot \hat{x}'^*) \right) \right) \quad (2.23)$$

KCF 主要采用高斯核与线性核，其中采用高斯核的算法称为 KCF，而基于线性核的算法称为 DCF。可以证明 K 此时仍然是循环矩阵^[6]，再次利用循环矩阵性质可得 α 的值

$$\hat{\alpha} = \frac{\hat{y}}{\hat{K}^{xx} + \lambda} \quad (2.24)$$

其中， κ^{xx} 是矩阵 K 的第一行向量，即 $K = C(\kappa^{xx})$ 。因为该式是在复域中表达，所以其计算量可得到有效控制。

2.3.3 目标检测

根据式 (2.19) 给出的判别函数，现只需将待判别的样本 z 代入即可。首先，利用核函数定义（见 2.3.2 节），对样本 x 与样本 z 做核运算，则有

$$K^z = C(\kappa^{xz}) \quad (2.25)$$

将其代入判别函数，并将 $f(z)$ 写成矩阵形式

$$f(z) = (K^z)^T \alpha \quad (2.26)$$

其中， K^z 的维度为 $n \times n$ ，每行由 κ^{xz} 的循环移位构成，而 α 为 $n \times 1$ 的列向量，即

$$f(z) = \begin{bmatrix} \kappa_{11} & \kappa_{12} & \cdots & \kappa_{1n} \\ \kappa_{1n} & \kappa_{11} & \cdots & \kappa_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ \kappa_{12} & \kappa_{13} & \cdots & \kappa_{11} \end{bmatrix}^T \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix} \quad (2.27)$$

再根据循环矩阵性质，将求解转移至复域中，最终可得

$$\hat{f}(z) = (\hat{\kappa}^{xz})^* \odot \hat{\alpha} \quad (2.28)$$

上式，即是目标检测函数最终的形式。每个 $f(z)$ 都是 κ^{xz} 各个元素的线性组合，而组合的权重由训练得到的 α 来决定。跟踪的目标，显然应该出现在 $f(z)$ 响应最大的位置。

2.4 模板更新方法

模板更新是跟踪算法不可或缺的环节，主要包含两个部分，一是目标的外观模型 \hat{x} ，二是分类器参数 $\hat{\alpha}$ ，具体更新方法如下

$$\hat{\alpha}_p = \eta \hat{\alpha} + (1 - \eta) \hat{\alpha}_{p-1} \quad (2.29)$$

$$\hat{x}_p = \eta \hat{x} + (1 - \eta) \hat{x}_{p-1} \quad (2.30)$$

其中， p 表示当前帧数（下同）， η 为学习率，其值在 $[0.01, 0.15]$ 范围内最佳^[15]。可见，新一帧的模板参数主要由上一帧的参数决定，而当前帧的模板参数只起到微调作用。这样的好处在于，当跟踪目标的外观发生突变时，利用前一帧的参数可以有效地降低这种噪声带来的影响。然而，这种处理方式也有其缺点。一方面，如果学习率较小，则模板更新缓慢，难以应对目标外观的快速变化；另一方面，如果学习率较大，跟踪器便会倾向于跟踪背景中的目标，而一旦目标发生遮挡，则跟踪器以后再无法识别出目标。

为了解决这个问题，可以将从第一帧到当前帧的所有模板参数考虑进来。首先定义一个关于模型匹配的损失函数，一般仍采用正则化最小二乘法，有

$$\min_{\alpha} \left\| \sum_{i=1}^p f(x_i) - y_i \right\|^2 + \lambda \sum_{i=1}^p \|\alpha_i\|^2 \quad (2.31)$$

此时，可得相关滤波器模板的解为

$$\hat{\alpha}_p = \frac{\sum_{i=1}^p \hat{y}_i \hat{\kappa}^{x_i x_i}}{\sum_{i=1}^p \hat{\kappa}^{x_i x_i} (\hat{\kappa}^{x_i x_i} + \lambda)} \quad (2.32)$$

记 A 、 B 分别为式（2.32）的分子与分母，则其更新方法可表述为

$$A_p = \eta \hat{y}_p \hat{\kappa}^{x_p x_p} + (1 - \eta) A_{p-1} \quad (2.33)$$

$$B_p = \eta \hat{\kappa}^{x_p x_p} (\hat{\kappa}^{x_p x_p} + \lambda) + (1 - \eta) B_{p-1} \quad (2.34)$$

因为该方法累加了全部的历史帧，所以当某一帧出现巨大偏差时，对新的滤波器模板影响很小，这种模板更新方法对于目标短时间局部遮挡的情况具有较好的适应性，已被 DSST、CN^[16]等算法采用。

2.5 多通道处理

与上述过程相比，KCF 的创新更体现在多通道数据处理。其提出者早年提出的 CSK 算法已包括 KCF 所采用的的大多数模块。区别之处是 CSK 使用灰度特征，而 KCF 使用 HOG 特征。本节将阐述 KCF 的多通道处理策略，设待融合的特征有 c 个通道，即

$$x = [x_1, x_2, \dots, x_c] \quad (2.35)$$

KCF 将核函数的计算转移至复域，而复域中的计算以点乘 ‘ \odot ’ 和范数 ‘ $\|\cdot\|$ ’ 为主，因此对于多通道特征，一个简单的处理方案就是将其各个通道的值叠加起来。具体地，对于线性核、多项式核以及高斯核分别有

(1) 线性核

$$\kappa^{xx'} = F^{-1}(\sum_c \hat{x}_c \odot \hat{x}'_c^*) \quad (2.36)$$

(2) 多项式核

$$\kappa^{xx'} = \left(\mathcal{F}^{-1}(\sum_c \hat{x}_c \odot \hat{x}'_c^*) + a \right)^b \quad (2.37)$$

(3) 高斯核

$$\kappa^{xx'} = \exp\left(-\frac{1}{\sigma^2} \left(\|x\|^2 + \|x'\|^2 - 2\mathcal{F}^{-1}(\sum_c x_c \odot \hat{x}'_c^*) \right)\right) \quad (2.38)$$

这种处理方案的好处在于目标特征的获取不仅可以结合多种特征的优势，而且不会显著增加运算复杂度，对于跟踪速度的影响微乎其微。因此，KCF 的后续研究热点之一就是多种特征（例如，灰度、颜色、形状、纹理）快速融合，以期对目标外观更好地描述。

第3章 基于卷积特征的相关滤波算法

3.1 问题的提出

尽管 KCF 创造性地基于 HOG 进行目标特征提取，使用循环矩阵进行加速运算，以及提出了多通道处理等一系列高效快速跟踪策略。然而，KCF 的设计者最初只着眼于跟踪的速度，并没有在尺度自适应、长时跟踪、边界效应等方面深入挖掘，所以给 KCF 留下了很大的改进空间。而随着深度学习的持续发展，将深度学习技术应用在目标跟踪领域已十分普遍，本章将聚焦于此，重点研究深度学习特别是深度特征与相关滤波之间的关系。同时，本章也将对 KCF 中的部分缺陷以及相应的改进方法进行说明。

将深度学习应用的目标跟踪领域，通常有两种思路：深度特征与深度网络。其中，深度特征相较于传统手工特征，在相关滤波中加强了语义信息，从而提高跟踪精度，典型如 HCFT^[17]、DeepSRDCF^[18]。而凭借着深度网络强大的拟合能力，利用多重网络结构实现目标和背景的分类任务也取得了许多先进的成果，如 CFNet^[19]、MDNet^[20]等。由于本文的重点在于相关滤波，因此选择其中典型算法 HCFT，并在此基础上结合 HCFTplus^[21]、HCFTstar^[22]以及 LMCT^[23]的创新点，提出了一个高置信度目标跟踪算法 HC-HCFT^①。

HCFT 在 2015 年的 ICCV 上提出，其基本思想为：卷积神经网络的低层特征具有高分辨率，能够对目标进行精确的定位，而高层特征具有更多的语义信息，能够处理较大的目标变化以及抑制模板漂移。具体来说，在深度卷积神经网络 VGG-NET-19 中，首先利用卷积层 conv3_4、conv4_4 与 conv5_4 提取目标模板的特征（简称“卷积特征”）。然后，分别将不同卷积层提取到的特征送入相关滤波分类器，得到每一个卷积层滤波器的响应图，再通过一定的多特征融合策略，综合判断滤波器的响应情况，最终得到目标的位置。

HCFT 的创新在于采用深度特征代替 HOG 特征，一定程度上提高了跟踪的精度，遗憾的是，和 KCF 一样，HCFT 的模板更新策略过于简单，且没有对目标尺度进行评估，也不能胜任长时跟踪的需求。因此，在 2016 年 SPL 提出的 HCFTplus 与 2017 年 TPAMI 提出的 HCFTstar 中，其作者相继增加了模板更新条件、尺度自适应以及目标丢失再检测三种改进。然而，其目标检测框的置信度仅仅以相关滤波器响应的最大值来衡量，这种准则在 LMCT 中被证明是不可靠的，因此，本文引入 LMCT 中的置信度评价指标 APCE，将其与滤波响应结合，以期在上述三个环节提供高置信度的检测结果。

^① HC-HCFT: High-Confidence Hierarchical Convolutional Features for Visual Tracking.

3.2 方案整体设计

HC-HCFT 算法以相关滤波的框架为基础，只是在特征提取环节以三层卷积网络特征替代，并增加了对检测结果的置信评估，以及尺度自适应、目标丢失再检测、高置信模板更新等环节，HC-HCFT 算法流程如图 3.1 所示。

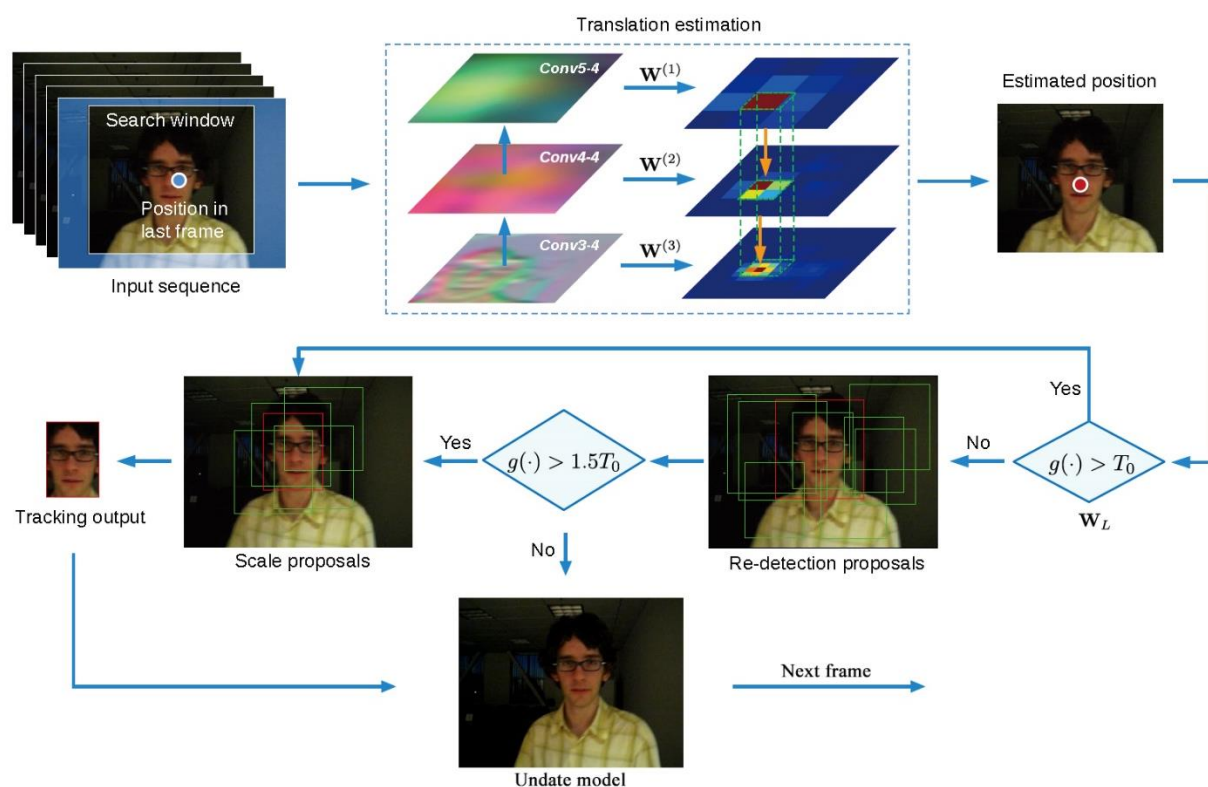


图 3.1 高置信度的多层卷积特征目标跟踪算法框架

因为第 2 章已经系统地分析了相关滤波算法，所以下文只对 HC-HCFT 中的卷积特征、多层特征融合方式、高置信度指标、以及尺度、长时跟踪等模块进行介绍。

3.3 多层卷积特征

3.3.1 深度卷积网络

卷积神经网络^①（CNN）是深度学习的一种重要的算法，它沿用了多层感知器的结构，通过不同的卷积层，不同的卷积核，能对输入图像训练出高维的鲁棒性特征。它最早于上

^① CNN 可视化工具：<https://github.com/HarisIqbal88/PlotNeuralNet>。

世纪 80 年代由日本学者 Fukushima^[24]提出，但由于彼时深度网络理论不完善、计算机硬件水平低下，CNN 在提出后的很多年内都未引起普遍关注。但时至今日，随着 Alex-Net、VGG-Nets、NIN、Res-Net 等深度卷积神经网络(Deep Convolutional Neural Network, DCNN)不断被推出，深度学习理论与应用已经日趋完善。其中，VGG-Nets 是 2014 年 ImageNet 竞赛排名第一的 DNN，由于其良好的泛化性能，故其预训练模型（Pre-trained Model）被广泛应用于图像特征提取，因此本文采用 VGG-Nets 中的 VGG-NET-19 来提取图像特征。

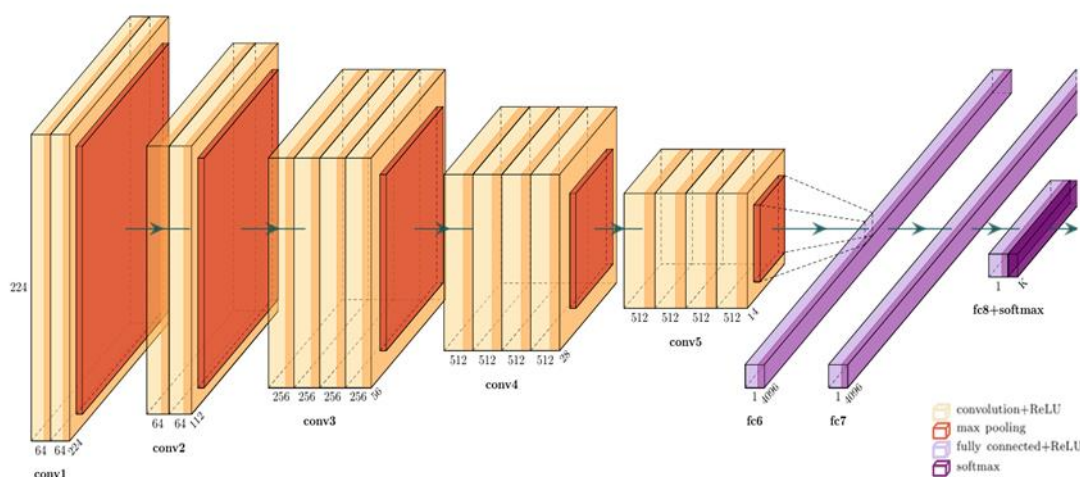


图 3.2 VGG-NET-19 深度网络结构图

如图 3.2 所示，VGG-NET-19 包含了 19 层深度卷积网络，其中 conv1 与 conv2 各有 2 层，conv3 至 conv4 各有 4 层，最后 3 层为全连接层，池化（pooling）为非卷积操作因此不计入层数。VGG-NET-19 深度网络具体参数见表 3.1。

表 3.1 VGG-NET-19 深度网络参数表

| Structure | Filters | Output Size | Memory |
|-------------|---------|-------------|--------|
| Image Input | | 224*224*3 | 150K |
| Conv1_1-2 | 64 | 224*224*64 | 3.2M |
| Pooling1 | | 112*112*64 | 800K |
| Conv2_1-2 | 128 | 112*112*128 | 1.6M |
| Pooling2 | | 56*56*128 | 400K |
| Conv3_1-4 | 256 | 56*56*256 | 800K |
| Pooling3 | | 28*28*256 | 200K |
| Conv4_1-4 | 512 | 28*28*512 | 400K |
| Pooling4 | | 14*14*512 | 100K |
| Conv5_1-4 | 512 | 14*14*512 | 100K |

续表

| | | | |
|----------|------|----------|-----|
| Pooling5 | | 7*7*512 | 25K |
| Fc6 | 4096 | 1*1*4096 | 4k |
| Fc7 | 4096 | 1*1*4096 | 4K |
| Fc8 | 1000 | 1*1*1000 | 1K |

说明：filter 为卷积核的数量，每一个卷积核都将对应一个通道（channel）；Output Size 的前两个参数为每个通道的 Size，而后一个参数即通道数；表中的 Memory 指的是 Image Input 在进行某层网络的运算后所需占用的存储空间。

3.3.2 深度卷积特征

目标的外观模型对于目标跟踪任务而言十分关键，因此从 CSK 中的灰度值至 KCF 的 HOG 特征，研究者对外观模型的探索从未止步。正如上文所述，卷积特征以其丰富的卷积核、强大的表达能力，在目标外观描述方面具有得天独厚的优势。本节，将对 HCFTstar 算法中采用的卷积特征做一定的说明。

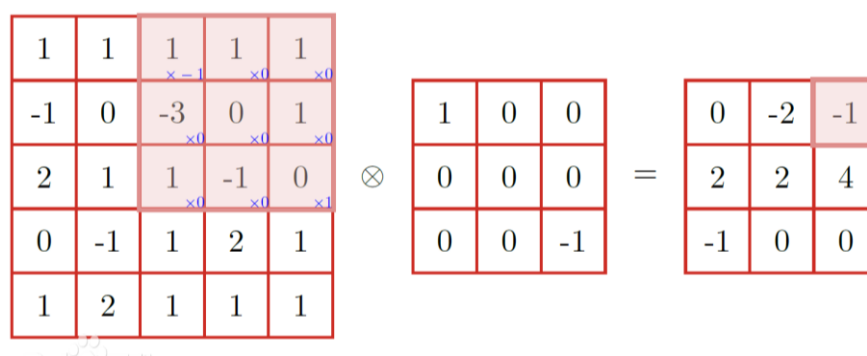


图 3.3 卷积操作与卷积特征示意

如图 3.3 所示，当给定一个卷积模板，则二维卷积操作即意味着对应元素相乘再求和，当卷积核与模板运算完成所得输出即为该通道的卷积特征。限于篇幅，此处不对卷积过程的“填充（Padding）”、“步长（Stride）”等概念一一说明，具体可参见 CS231n 系列课程。在 HCFTstar 中，作者试验性地使用卷积层 conv3_4、conv4_4 与 conv5_4 进行特征提取，由于这三者输出特征图尺寸不同，而在计算相关滤波器时，需要搜索区域尺寸与特征尺寸保持一致，因此对卷积特征进行双线性插值处理，有

$$x_i = \sum_k \alpha_{ik} h_k \quad (3.1)$$

其中， h 为原始特征图， x 为采样后的特征图， i 代表位置， k 代表 K 邻域的特征向量， α 为插值系数，其值由 i 和 k 邻近的特征向量的位置决定。这样，不同尺寸的特征图便可以

统一到相同尺寸，为多特征融合提供便利。

3.3.3 多特征融合方法

HCFTstar 在融合三层卷积网络特征时，首先逐层计算 CNN 特征与相关滤波器的响应，并基于“Coarse-to-Fine”的思想（即“由粗糙到精细”），建立了加权滤波器响应的优化函数，再以此决定跟踪目标当前的位置，其基本框架如图 3.4 所示。

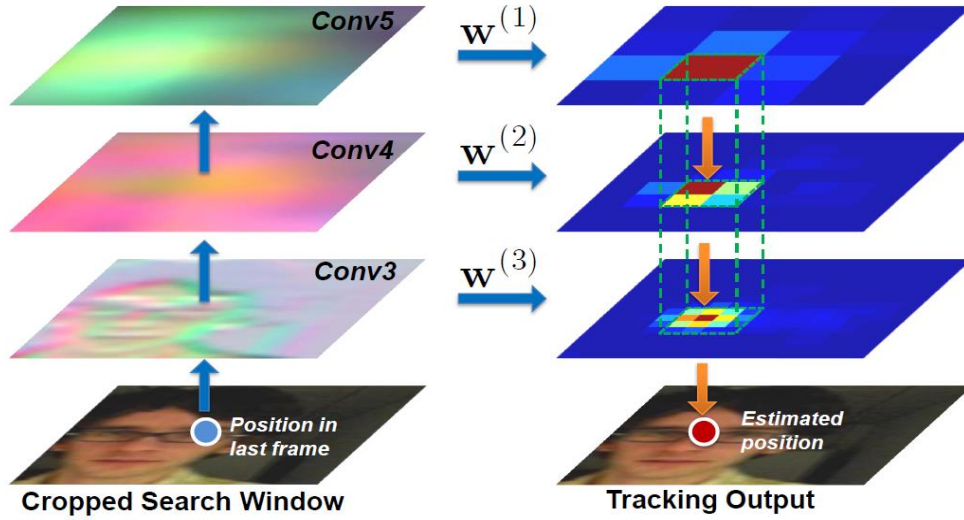


图 3.4 多层卷积特征融合及目标跟踪方法示意

记 conv3_4、conv4_4、conv5_4 三层卷积特征与滤波器在位置 (m,n) 的响应分别为 $f_3(m,n)$, $f_4(m,n)$, $f_5(m,n)$, 卷积层权重为 μ_3, μ_4, μ_5 , 则目标位置优化模型为

$$\max_{m,n} \sum_l \mu_l f_l(m,n) \quad (3.2)$$

而 μ 的确定，在 HCFT 中是由实验测试给出，为 $[0.5 \ 1 \ 0.02]$ ；在 HCFTplus 中，作者将其调整为 $[0.2 \ 0.5 \ 1]$ ；可见，权重的取值具有较大的自由度，因此在 2018 年 PAMI 的论文里作者对权重做了深入的研究，并提出了三种权重设置方案。

(1) Soft mean: $\mu_l = 1 / \max(f_l)$

(2) hard weight: $\mu_l = 2^{l-5}$

(3) soft weight: both(1) and (2)

对于权重设置方案 (3)，目标位置优化模型相应地变为

$$\max_{m,n} \sum_l \frac{\mu_l f_l(m,n)}{\max(f_l)} \quad (3.3)$$

其中， $\mu_l = 2^{l-5}$ ，通过在 OTB100 的测试表现来看，soft weight 这种稍显复杂的权值设置方案，能够带来更高的跟踪准确率（+1%）。

3.4 检测结果置信度

一般而言，目标检测的置信度由滤波响应的最大值决定。然而，LMCT 发现在目标被遮挡时，由于提取的目标特征的缺陷，使得正确跟踪与错误跟踪的滤波响应值差距并不大。如图 3.5 所示，当目标被遮挡时，滤波器响应会振荡剧烈。基于对振荡程度的评估，他们提出了一种称为 APCE（Average peak-to-correlation energy）的置信度评估标准：

$$APCE = \frac{\|f_{\max} - f_{\min}\|^2}{\text{mean}(\sum_{i,j} \|f_{i,j} - f_{\min}\|^2)} \quad (3.4)$$

其中， $f_{i,j}$ 表示 (i, j) 位置上的滤波器响应，不同遮挡情况下的跟踪结果如图 3.5 所示。

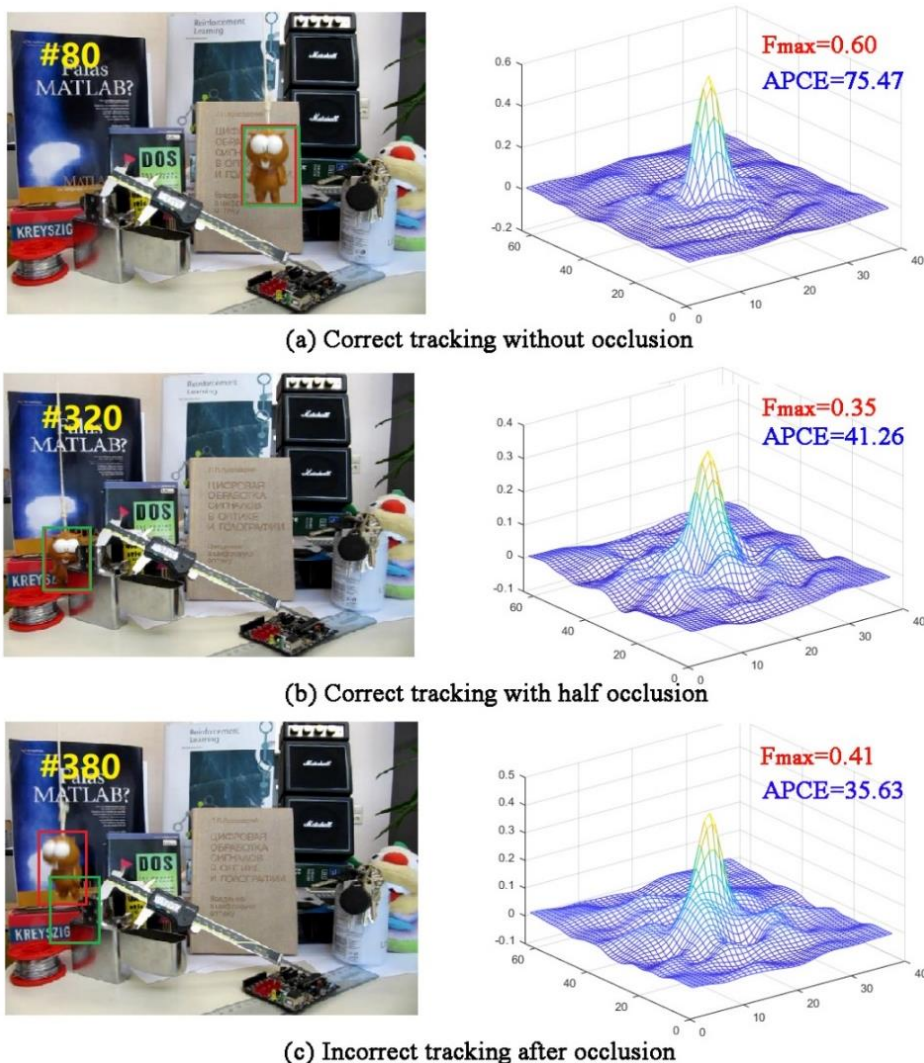


图 3.5 不同遮挡程度下的目标跟踪结果置信度

图中第一列反映了目标的遮挡与跟踪情况，分别是无遮挡（正确跟踪）、半遮挡（正确

跟踪)、无遮挡(模板已漂移,错误跟踪),其中红框表示目标实际位置,滤波表示跟踪器检测位置。第二列反映了滤波器响应情况,无遮挡时目标响应尖锐,其余平缓,滤波器最大响应为 0.60, APCE 为 75.47;目标被半遮挡时,滤波器响应依然明显,但目标周围的振荡幅度增大,最大滤波响应下降至 0.35, APCE 下降至 41.26;而在目标全部遮挡后,模板发生了偏移,跟踪结果如图 3.5 (c) 所示,尽管目标未遮挡,但跟踪已经失败,此时其目标响应尽管比半遮挡时大,为 0.41,但这个数值并不能说明跟踪效果比半遮挡高,而 APCE 值再一次下降,为 35.63,可见用 APCE 反映目标跟踪的置信度比滤波最大响应更加可靠。

综合考虑, LMCT 将 APCE 于滤波最大响应进行赋权累加,得到高置信度指标 $g(b)$

$$g(b) = \beta_1 f_{\max}(b) + \beta_2 APEC(b) \quad (3.5)$$

其中, b 为检测结果对应的图像块, β 为两个置信指标的权重。下一节将介绍高置信度指标在目标尺度评估、目标丢失再检测以及模板更新这三个环节中的应用。

3.5 高置信度目标检测

3.5.1 目标尺度评估

不进行目标尺度的评估(Scale)是 KCF 公认的几个缺陷之一。事实上,若跟踪算法能够有效地评估目标的尺寸,则能大幅度提高跟踪的成功率(定义见 4.2.2 节)。所以,目标尺度评估成为了目标跟踪重要的研究主题。HCFTstar 没有沿用 DSST^[7]、fDSST^[25]这条尺度金字塔线,而是采用国防科技大学 Dafei Huang^[26]等提出的 KCFDP 算法。

用于目标尺度评估的 KCFDP 算法的伪代码如表 3.2 所示,它主要由四个步骤构成:

- (1) 通过 EdgeBox^[27]算法生成多个目标检测框(选取得分最高的前 200 个,每个检测框都具有一定的得分);
- (2) 通过一定的过滤机制,去掉成功率不符合要求的检测框;
- (3) 利用过滤后有效的检测框与滤波器做运算(双线性插值进行维度匹配),并计算检测框的响应。
- (4) 在多个响应中选择具有高置信度的检测框,并与 KCF 获得的检测框的高置信度对比,进而确定该帧目标的尺度与位置。

表 3.2 KCFDP 算法伪代码

Algorithm KCFDP

Inputs:

F_i : image frame

$(I_{i-1}, w_{i-1}, h_{i-1})$: center location, width, height of target in previous frame

α, \bar{x} : previous filter coefficient and target appearance

Outputs:

(I_i, w_i, h_i) : center location, width, height of target in current frame

α, \bar{x} : filter coefficient and target appearance after updating

Preliminary translation estimation:

1. Extract a patch z^d in $(I_{i-1}, s^d w_{i-1}, s^d h_{i-1})$ from F_i
2. Detect a new center location I_i^d with z^d, α, \bar{x}

Scale and aspect ratio estimation:

3. Extract a patch z^p in $(I_{i-1}, s^e w_{i-1}, s^e h_{i-1})$ from F_i
4. Apply *EdgeBoxes* to z^p and get a set of proposal bounding boxes P
5. Use proposal rejection to filter $P[1, \dots, 200]$ and get P^l
6. Evaluate each proposal in P^l and get the box (I_i^p, w_i^p, h_i^p) with the high confidence
7. Get (I_i, w_i, h_i) by updating with damping using $(I_i^p, w_i^p, h_i^p), (w_{i-1}, h_{i-1}), I_i^d$

Model update:

8. Extract a patch x_i in $(I_i, s^d w_i, s^d h_i)$ from F_i
9. Update α and \bar{x} with x_i

3.5.2 目标丢失再检测

在长时跟踪（Long-term）问题中，当目标产生局部或完全遮挡、高速运动，剧烈形变时，极有可能使跟踪器跟踪失败，而这种失败就表现在对目标的检测错误。而利用滤波器响应的置信程度，则可以清晰地反映检测框与目标的匹配情况。匹配度低则响应值小，反之亦然，这构成了目标丢失再检测策略的出发点。

当给定一个目标检测区域后，首先计算其滤波器响应，通过高置信度指标与设定的跟踪失败阈值对比。若不满足，则将生成一系列新的目标检测框，再在这些框中同时考虑滤波响应与运动特性，最终确定目标新的位置。由于滤波响应易于理解，故这里我们只介绍下运动特性，记 b_t^i 为再检测机制在第 t 帧生成的第 i 个框，以 b_{t-1} 的目标框作为参照，可以计算得第 i 个检测框与之前的目标框的距离 D 为

$$D(b_t^i, b_{t-1}) = \exp\left(-\frac{1}{2\sigma^2} \|(x_t^i, y_t^i) - (x_{t-1}, y_{t-1})\|^2\right) \quad (3.6)$$

其中， σ 为初始目标大小的对角线长度。可见，新生成的目标框若与先前的目标框距离稍大，则其距离 D 都将非常之小。该计算公式将新的目标位置限制在距离原来的目标位置很近的范围，这是基于运动过程中相邻帧之间的目标位移不会太大的假设。然后，只

需通过以下优化目标，即可确定最终采纳的目标框。

$$\min_i g(b_t^i) + \alpha D(b_t^i, b_{t-1}) \quad (3.7)$$

$$\text{s.t. } g(b_t^i) > 1.5T_0$$

其中， α 为权重因子， $g(b)$ 为目标检测区域 b 的高置信度， T_0 为跟踪器检测失败所采用的滤波响应阈值。从上式可知，滤波响应与运动特性已融入一个共同的优化目标。

3.5.3 高置信度模板更新

跟踪模板更新（包括滤波器与目标外观）对目标跟踪任务十分重要，但是大部分跟踪算法在更新模板时并没有对检测结果的置信度进行评估。事实上，如果目标出现了遮挡等问题，此时对目标的检测是不准确的，尽管 HC-HCFT 包含了目标丢失再检测环节，然而从目标丢失到再检测的时间内，模板更新依然在继续，这必然会导致模板受到污染。因此，一个直观的处理方案就是当置信度不足时停止模板更新。

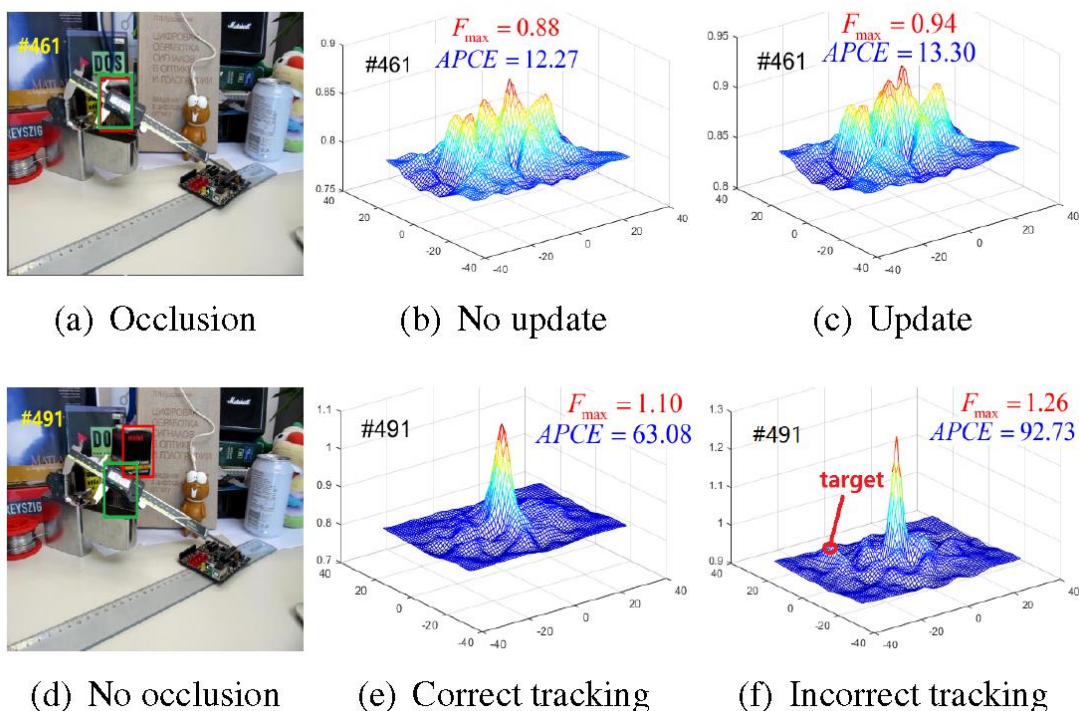


图 3.5 目标遮挡与 APCE 等指标示意

采用高置信度模板更新方案前后的效果对比如图 3.5 所示。其中，第一列表示目标的遮挡与否，红框表示目标的实际位置，绿框表示模板漂移后的跟踪结果。第二列表示目标遮挡后滤波响应振荡严重，停止模板更新，继而能够正确检测到目标的位置。第三列表示

滤波振荡严重，但模板依然更新，使得模板漂移，导致检测错误。此外，该图也提供了最大滤波响应 F_{\max} 与 APCE 的数值。通过对比可以发现，在检测正确的前提下，APCE 比 F_{\max} 的区分度更明显。

HC-HCFT 利用式（3.5）中的高置信度 $g(b)$ 来更新模板。只有在当前帧的 $g(b)$ 高于历史平均综合置信度时，重启模板更新，否则维持原状。这种方式既能有效缓解模板漂移所导致的跟踪失败问题，又能降低模板的更新频率，提高模板更新速度。

第 4 章 实验结果及分析

4.1 目标跟踪数据集

目标跟踪研究领域在 2013 年之前没有权威的视频数据集。研究人员往往需要自行构造数据集来测试其算法的性能，极大妨碍了论文结果的复现以及算法的比较。随着目标跟踪研究的深入，标准视频数据库也顺势推出。与之相关的，还诞生了关于目标跟踪的国际竞赛，这不仅满足了研究需要，也很大程度地促进了跟踪技术的发展。

当前主流的标准数据集有 OTB（Object Tracking Benchmark）与 VOT（Visual Object Tracking）两个系列。前者由 Yi Wu^[28]等于 2013 年 CVPR 上推出，并迅速成为学界的基准。起初 OTB 包含 50 个测试视频，故称为 OTB50。为了防止算法的过拟合现象，Yi Wu^[29]等于 2015 年又将其扩增至 100 个，即 OTB100。

除了 OTB 系列，另一个标准数据集为 VOT。它来源于 VOT 挑战赛，自 2013 年 VOT 组委会每年都会发行一个视频数据集^[30, 31]，其评价准则与数据集编排方式与 OTB 有较大差异，见表 4.1。一般而言，很少有算法能在两个数据集上均有较为理想的表现。此外，OTB 更新较慢，而 VOT 每年一变，因此对挑战者算法的性能要求更为苛刻。

表 4.1 VOT 与 OTB 数据集差异

| Attribute | OTB | VOT |
|------------------------|------------|-----------|
| Color video proportion | 75% | 100% |
| Video resolution | High | Low |
| Evaluation criteria | PR、SR | EAO、EFO |
| Tracking process | Short-term | Long-term |

说明：Short-term 与 Long-term 的区别在于对跟踪过程中目标丢失情况的处理方式上，Short-term 若目标丢失则跟踪结束，而 Long-term 则是在等待数帧后重新检测目标。

4.2 跟踪算法评价指标

一般而言，目标跟踪算法的性能主要可以从三个方面来评价，即准确性、鲁棒性与高效性^[32]。准确性指的是跟踪器在跟踪目标时产生偏移、误检与漏检的程度；鲁棒性指的是跟踪算法对不同的视频跟踪场景的适应程度；高效性主要指跟踪算法的实时性。针对这三个方面的需求，许多具体的评价方法被提出，如准确率、成功率、时间鲁棒性、空间鲁棒性、FPS 等^[28]，下面对其中常见的三种指标做详细说明。

4.2.1 准确率（PR）

由于在目标跟踪任务中，一般是用目标框来代表目标的跟踪位置，而目标的实际位置与算法的跟踪位置往往存在差距。准确率（Precision Rate, PR）的提出正是为了量化这种差距，其定义为目标实际中心与算法跟踪的目标中心之差，在给定的距离阈值之内的帧数占总帧数的百分比。我们将之分为“距离之差”与“帧数百分比”两部分来解读。

（1）距离之差

首先，用中心误差（Center Location Error, CLE）来量化目标实际中心与目标框中心距离。如图 4.1（a）所示，设目标实际中心为 (x, y) ，目标跟踪中心为 (x', y') ，则有

$$CLE = \sqrt{(x' - x)^2 + (y' - y)^2} \quad (4.1)$$

CLE 的单位为像素，其值反映了跟踪算法对目标中心的跟踪性能。

（2）帧数百分比

上述的距离之差，只反映了某一帧图像，所以为了整体评价算法对某个视频每帧的跟踪情况，于是增加了“帧数百分比”这部分。首先，设某个视频共有 N 帧，而中心误差 CLE 的阈值设为 D ，而在 N 帧视频中满足 $CLE < D$ 的共有 M 帧，则此时的准确率 P 为

$$P = \frac{M}{N} \times 100\% \quad (4.2)$$

为了更好的显示，不同的阈值情况下算法的跟踪准确率的变化，将阈值 D 扩展为一个数组，从小到大排列为 $[d_1, d_2, \dots, d_c]$ ，则每一个阈值 d_i ，都将对应一个准确率 p_i ，依此类推，我们可以得到一个关于 p 的集合，即

$$\{p \mid p = \frac{m_i}{N} \times 100\%, i = 1, 2, \dots, c\} \quad (4.3)$$

此时，以 d 为横坐标， p 为纵坐标，即可得到算法的跟踪精度图了。但美中不足的是，当目标的尺度发生变化使尽管 CLE 的值可能很小，但是跟踪框与实际的目标框可能相去甚远。或者当算法跟踪失败时，跟踪框的位置是随机的，此时 CLE 就已经失去置信度。为了弥补 CLE 的这种缺陷，于是提出了成功率这项指标。

4.2.2 成功率（SR）

与 CLE 不同，成功率（Success Rate, SR）以重叠率（Overlap Rate, OR）来量化算法对目标的跟踪精度。其中，重叠率由目标实际区域与算法跟踪区域共同决定，如图 4.1（b）所示。设目标实际区域为 B_t ，算法跟踪区域为 B_s ，则重叠率定义为

$$OR = \frac{B_s \cap B_t}{B_s \cup B_t} \times 100\% \quad (4.4)$$

其中, $B_s \cap B_t$ 表示实际目标区域与算法跟踪目标区域的交集, 而 $B_s \cup B_t$ 为两者的并集, 显然 OR 的范围在 $[0, 1]$ 之间, 当两者完全重合时 $OR = 1$, 两者完全分离时 $OR = 0$ 。与准确率类似, 设置一个阈值 Q , 来衡量一个视频序列中满足 $OR < Q$ 的占其中的百分数。此外, 成功图的定义也与精度图类似, 此处不再赘述。

一般的, 准确率的综合表现采用的参考阈值为 20, 成功率的综合表现采用的参考阈值为 0.5。即在算法的横向比较时, 一般采用该值来描述算法准确度。而在绘制算法精度图与成功图时, CLE 阈值 D 的范围一般为 $[0, 50]$, OR 的阈值 Q 一般为 $[0, 1]$ 。

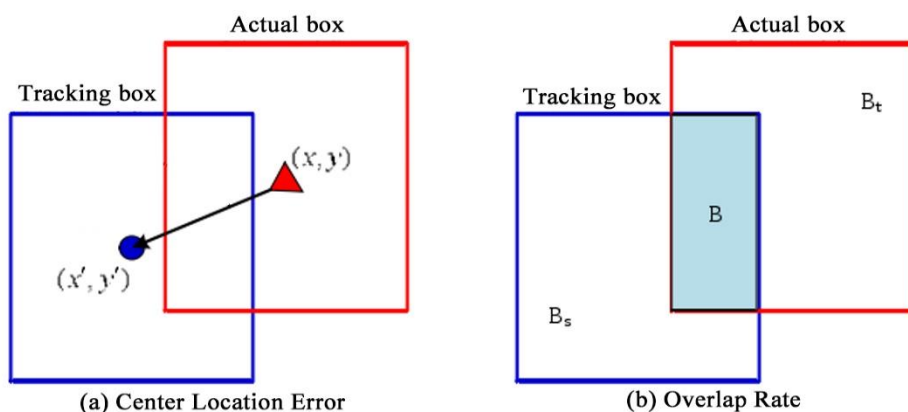


图 4.1 中心位置误差与重叠率

4.2.3 帧率 (FPS)

帧率 (Frame per Second, FPS) 即每秒处理帧数, 这是衡量跟踪算法实时性的重要指标。设某个视频有 N 帧, 而跟踪算法处理完该视频所需时间为 T 秒, 则帧率为

$$FPS = \frac{N}{T} \quad (4.5)$$

特别的, 在某个算法针对特定的数据集而言, 其所展现的帧率为平均帧率, 即

$$\overline{FPS} = \frac{\sum_Z FPS_i}{Z} \quad (4.6)$$

其中, Z 为采用的数据集中视频的数量。目前而言, 采用传统特征的算法, 其帧率都比采用深度特征的跟踪算法要高一到两个数量级。

4.3 实验环境与实验设置

4.3.1 软硬件环境

实验的软硬件主要包括 CPU/GPU/Memory 以及相关软件，具体参数见表 4.2。

表 4.2 实验软硬件环境

| Attribute | Parameter |
|--------------|--|
| CPU | Intel(R) Core(TM) i5-7300HQ CPU 2.5GHz |
| RAM | 8.00G |
| GPU | GeForce GTX 1050 |
| Video Memory | 2.00G |
| MATLAB | MATLAB2018B |
| C++ | Microsoft Visual C++ 2015 (C) |

4.3.2 实验设置方案

本文的算法测试将于 OTB50 上进行，这也是目前最为经典的测试数据集。OTB50 包括 50 个完全标注的测试视频，共分为 11 种属性，见表 4.3。

表 4.3 OTB50 测试视频属性信息

| Video Attributes | Illumination Variation | Scale Variation | Occlusion | Deformation |
|---------------------|---------------------------|------------------------|----------------------|--------------------------|
| | Motion Blur | Fast Motion | In-Plane Rotation | Out-of-Plane Rotation |
| | Out-of-View | Background Clutters | Low Resolution | —— |
| | | | | |

说明：测试视频通常可以被归并到多个属性，属性越多对算法的性能要求越高。

实验部分将选取若干典型算法在 OTB50 上进行测试，测试结果主要包括定性分析与定量分析。其中，定性分析为算法跟踪效果的直观表现，目标与目标检测框将在图片中展示，有利于对算法定性的把握；定量分析为算法跟踪性能的指标评测，如精确度、成功率、时空鲁棒性等，本文主要对准确率与成功率进行展示，定量分析便于评价算法的性能，定位算法的优势与劣势，有助于促进算法的进一步研究。

本文选择 8 种具有代表性的判别式跟踪算法进行测试，各种算法的特性见表 4.4。

表 4.4 典型判别式跟踪算法特性表

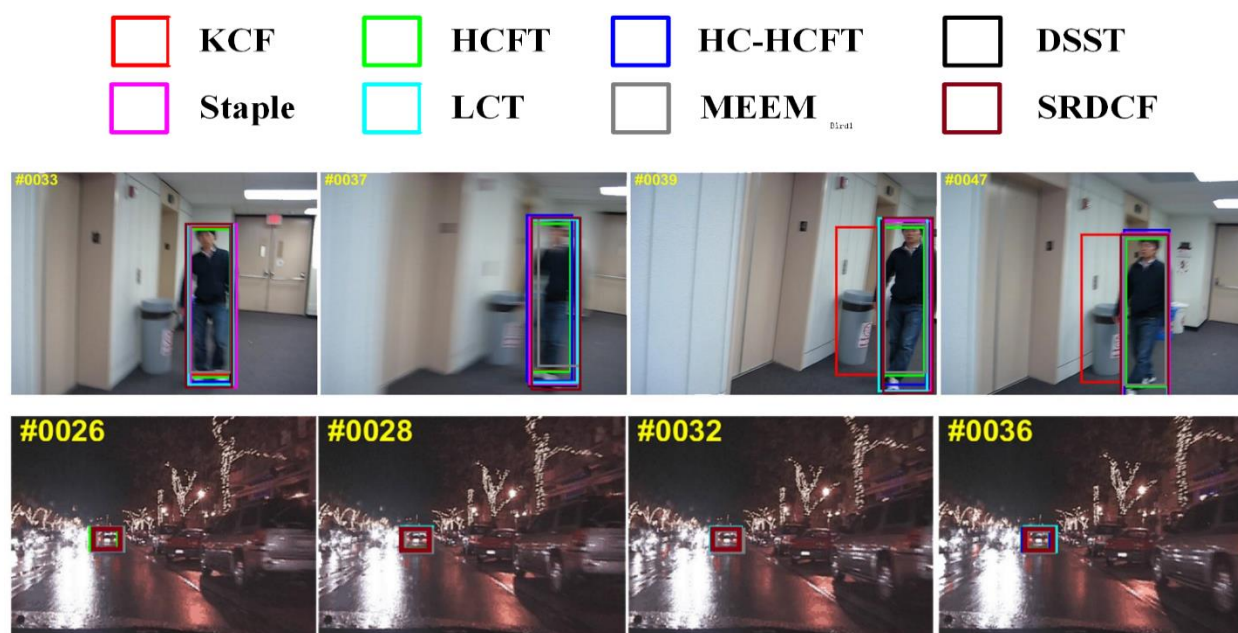
| Name | Description |
|---------|---|
| KCF | CF+ HOG Feature+ Gaussian kernel |
| HCFT | CF+ CNN Features |
| HC-HCFT | CF+ CNN Features+ Scale+ Re-detection+ APCE update |
| DSST | CF+ HOG Feature+ Scale |
| Staple | CF+ (HOG-Color) Features+ Scale |
| LCT | CF+ HOG Features+ Scale+ Re-detection+ F_{\max} undate |
| MEEM | SVM+ CIE Lab Features+ Multi-expert tracking framework update |
| SRDCF | CF+ (HOG-Color-Greyscale) Features+ Bounding+ Scale |

说明：CF 表示相关滤波；Features 表示目标外观使用的特征；Scale 表示算法具有目标尺度评估；Bounding 表示算法对检测区域的边界噪声采取了特别的处理方式；Update 表示算法采用改进的模板更新方法；Kernel 表示算法使用的核函数类型。

4.4 算法仿真与结果分析

4.4.1 定性分析

对上述 8 种不同的判别式跟踪算法在 OTB50 视频数据集进行测试，为了便于分析不同算法的跟踪性能，部分实验效果如下图 4.2 所示。



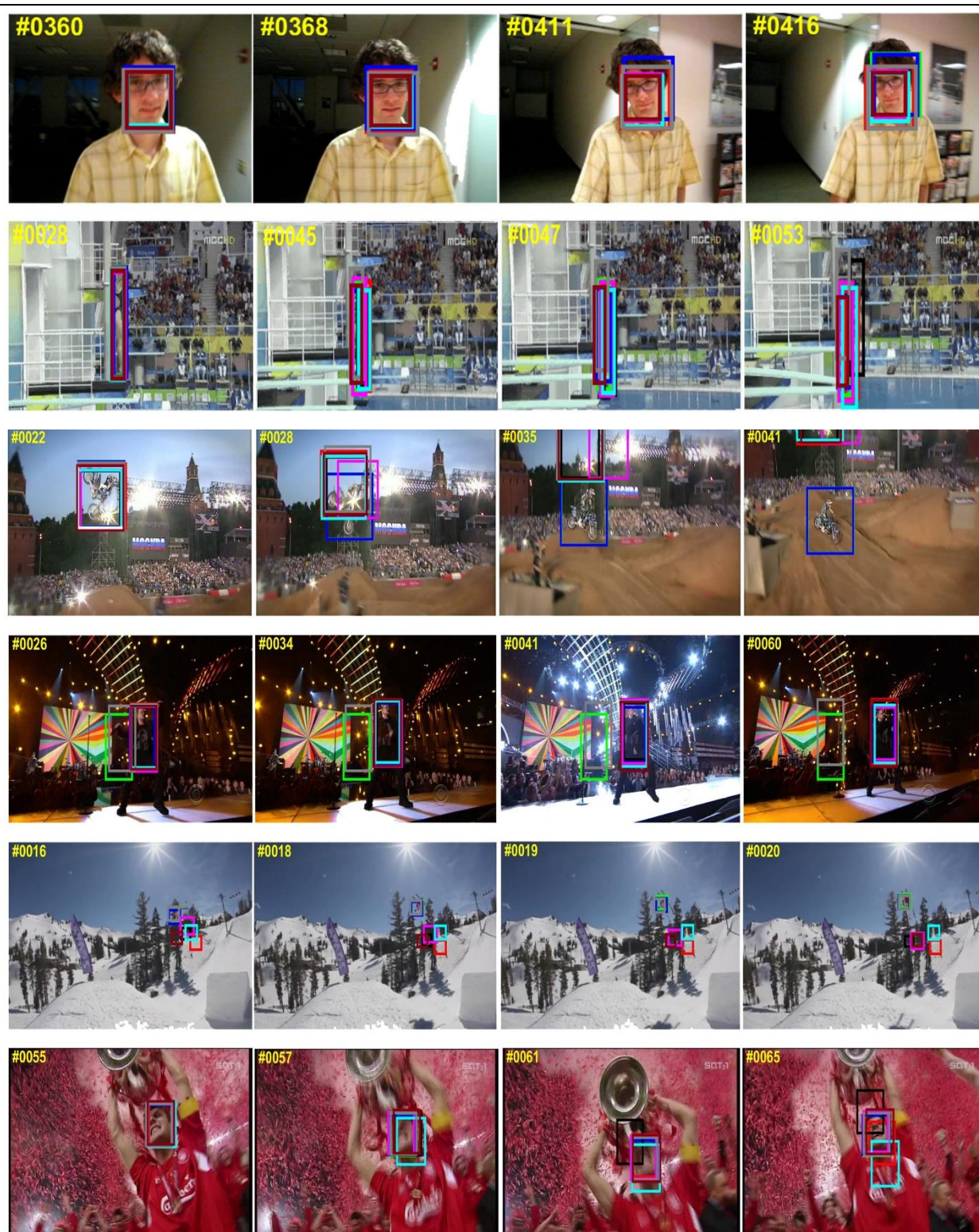


图 4.2 不同算法跟踪效果示意

整体上，8 种算法各有优劣。因为 KCF 采用 HOG 特征，所以在目标快速运动以及驳

杂背景的情况下表现较差。HCFT 的多层卷积网络特征比 HOG 特征更精确，但在驳杂背景和图像分辨率低的条件下依然表现不佳。因为 DSST 增加了尺度评估，所以对尺度变化的目标识别更为准确，比 KCF 具有更高的准确率。Staple 使用 HOG 和颜色两种特征，有效解决了 HOG 特征在目标快速运动以及背景驳杂情况下的缺陷，比 KCF 和 DSST 表现更佳。LCT 与 MEEM 都采用改进的模板更新方法，能够处理目标漂移的问题，有利于长时跟踪。SRDCF 采用了 HOG、颜色和灰度三种特征的组合，在低分辨率与驳杂背景情况下表现较好。此外，几乎所有的跟踪算法都难以解决目标超出边界的问题。HC-HCFT 算法独具高置信目标检测能力，能较处理好目标尺度变化和模板漂移问题，具有更高的精度与成功率。

4.4.2 整体分析

在全部 50 个视频的 One-pass Evaluation (OPE) 测试下，8 种算法的整体精度图与成功图如图 4.3 所示。在跟踪精度方面，HCFT 和 HC-HCFT 近乎一致，这是由于两者均采用多层卷积网络来提取特征，目标检测方式也基本相同，都利用基于循环矩阵特性的加速版相关滤波跟踪器。准确率方面，跟踪算法的排名依次是 SRDCF、MEEM、Staple、LCT、DSST、KCF。HC-HCFT 比之都有较为明显的提升，提升幅度分别为 8.82%、13.40%、14.83%、16.48%、28.62%、29.31%。

而 HC-HCFT 在成功率上面表现出应有的高性能，因为其特有的高置信度目标检测，包括尺度、长时跟踪、模板更新三个方面，在很大程度上提升了 HC-HCFT 对目标的框定精度，比 SRDCF、HCFT、Staple、LCT、MEEM、DSST、KCF 分别提升了 1.86%、6.81%、7.86%、11.59%、16.07%、21.46%、36.23%。可见 HC-HCFT 在整体表现上都处于较高的水平，在不同跟踪场景下具有普遍的适应性。

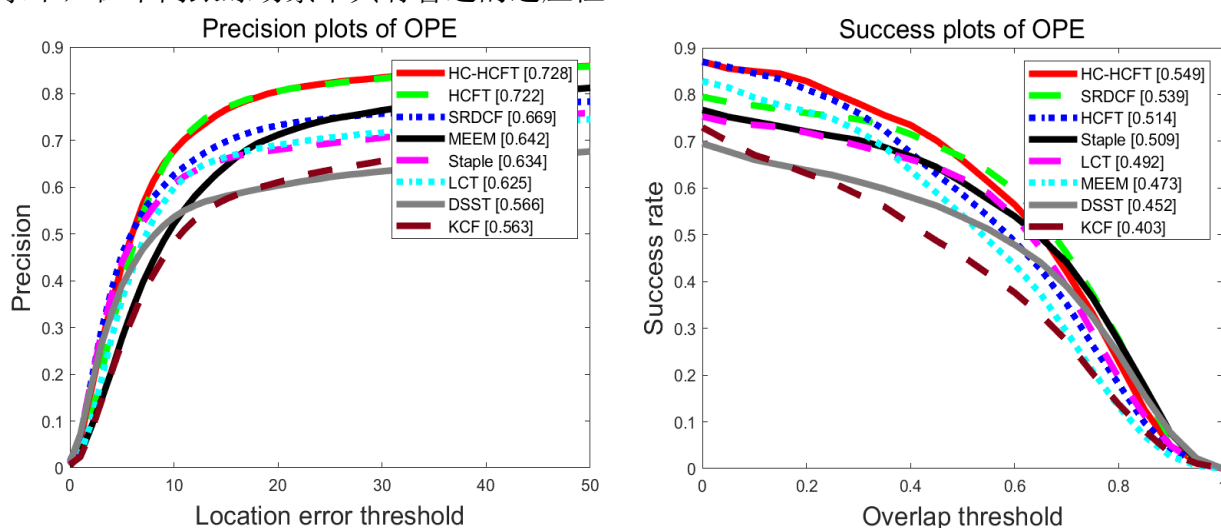


图 4.3 跟踪算法整体精度图与成功图

4.4.3 属性分析

为了更好地分析不同算法在不同属性的测试视频上的表现，在实验过程中还增加了属性实验。属性实验即是 8 种判别式跟踪算法按照 OTB50 划分的 11 种属性分别进行测试，测试结果的精度图与成功图如图 4.4-4.14 所示。

（1）光照变化

在光照变化的情况下，HC-HCFT、HCFT 和 SRDCF 在准确率上位列三甲，所以深度特征比传统特征具有更强的目标表达能力。此外，虽然 SRDCF 所采用的 HOG、颜色和灰度都是传统特征，但是它们的组合可体现多特征融合的优势。成功率上的前三名依次是 HC-HCFT、SRDCF 和 Staple。一方面，它们都具有尺度自适应的能力。另一方面，HC-HCFT 采用深度特征，而 SRDCF 与 Staple 均使用多种传统特征的组合。综上所述，采用深度特征或者多特征融合的算法在光照变化的情况下表现出明显优势。

（2）尺度变化

在目标尺度变化的情况下，准确率上的前三名依次是 HCFT、HC-HCFT 和 MEEM。HC-HCFT 与 HCFT 的差距较小，这是因为深度特征对尺度变化的目标依然有较好的表达能力。MEEM 采用 CIE 特征与改进的模板更新方式，减少了目标漂移，所以其跟踪能力较强。成功率上的前三名为 HC-HCFT、SRDCF 和 HCFT，它们在光照变化方面也位居前三。因此，高置信指标有助于提升 HC-HCFT 的尺度自适应能力。

（3）目标遮挡

在目标被遮挡的情况下，准确率上的前三名依次是 HCFT、HC-HCFT 和 SRDCF，主要原因和上面的分析一样。值得注意的是，高置信度目标检测的 HC-HCFT 需要调节更多的参数，而本文的实验并未对每个视频单独进行调参，所以其表现略逊于 HCFT。成功率上的前三名依次是 Staple、SRDCF 和 HCFT，HC-HCFT 排名第四，Staple 与 SRDCF 的排名体现出多特征融合的优越性。

（4）目标形变

目标发生形变时，准确率上的前三名依次是 HC-HCFT、HCFT 和 Staple。前两者使用深度特征，而 Staple 采用 HOG 和颜色特征的组合。因为颜色特征在目标形变时比较稳定，所以 Staple 的排名靠前。Staple、LCT 和 HC-HCFT 在成功率上排名前三。Staple 的表现依旧出色，LCT 与 HC-HCFT 改进的模板更新策略也缓解了目标漂移问题。

（5）运动模糊与快速运动

目标的运动模糊与快速运动效果类似，高准确率的有 HC-HCFT、HCFT、SRDCF；成功率上，排名前列的依然是 SRDCF、HC-HCFT、HCFT。一是因为深度特征与多特征融合

能更清晰地表达目标，二是因为 HC-HCFT 的高置信模板更新能缓解模型漂移问题。

（6）平面内旋转与平面外旋转

平面内旋转即二维旋转，平面外旋转即三维空间内旋转。在准确率与成功率表现上前三的算法依然是 HC-HCFT、HCFT、SRDCF。因为这三者都没有增加对目标旋转角度的学习，因此对比的重点依然是特征与目标更新方式。深度特征无疑是高精度与高成功率的主要贡献者。SRDCF 的三种特征与边界噪声抑制，也使其性能优异。

（7）目标越界

在目标越界方面表现较好的算法依次是 MEEM、HC-HCFT、HCFT 和 Staple。凭借独特的 Multi-expert 跟踪框架，MEEM 能够定位越界目标，因此在准确率上的优势突出。HCFT 和 HC-HCFT 的深度特征在反映目标外观上的优势依然明显。此外，HC-HCFT 采用目标丢失再检测技术，即使目标完全丢失，也能重新对其定位。

（8）背景驳杂

在背景驳杂方面，准确率上的前三名依次是 HC-HCFT、HCFT 和 SRDCF，该结果再次展现了特征提取的重要性。成功率上的前三名分别为 HC-HCFT、SRDCF 和 HCFT。其中，HC-HCFT、SRDCF 具有尺度自适应能力，而 HCFT 中没有加入尺度自适应环节。

（9）低分辨率

低分辨视频对跟踪算法的影响更大，因为低分辨率意味着目标模板信息有限，可利用的目标与背景信息都明显减少。尤其在分辨率低且目标小的情况下，跟踪算法通常无能为力。在低分辨率的情况下，准确率上排名前三的依次是 HCFT、HC-HCFT 和 SRDCF，这再次体现了特征提取的重要性。在成功率上排名前三的分别是 HC-HCFT、SRDCF 和 HCFT，这也再次证明了尺度自适应的重要性。

（10）属性总览

经过对 OTB50 共 11 种属性的分析，在准确率方面，发现 HC-HCFT、HCFT、SRDCF 三种算法表现最佳，这与 4.4.2 节中的整体分析结论相同。可见，深度特征与多特征结合可以显著地提高算法的跟踪准确率。而具有高置信度目标检测（包括尺度自适应、长时跟踪策略、改进的模板更新）的 HC-HCFT 算法在多个属性中准确率都位列第一，在光照变化、目标形变、运动模糊、目标旋转、背景驳杂等情况下比 HCFT 具有更先进的效果。

成功率方面，表现前三依然是 HC-HCFT、SRDCF、HCFT。注意到 SRDCF 由于具有尺度自适应，因而在一些属性中具有比 HCFT 更好的表现。因为 HC-HCFT 采用高置信目标尺度评估，且由于深度特征，故而是比 SRDCF 整体表现更好。而 DSST、KCF 等 Baseline 算法则长期处于排名劣势，这也证明了各种改进算法的性能优越性。

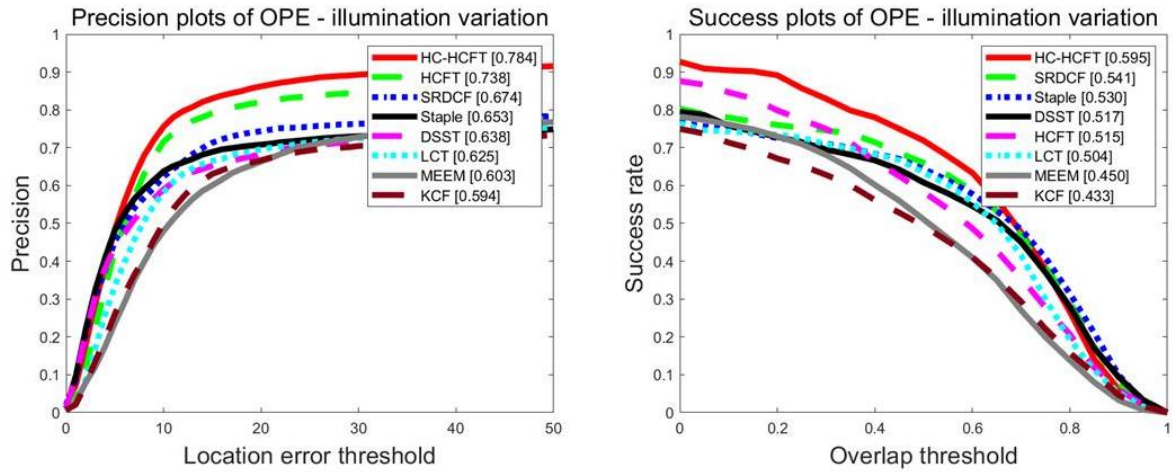


图 4.4 光照变化——跟踪算法精度图与成功图

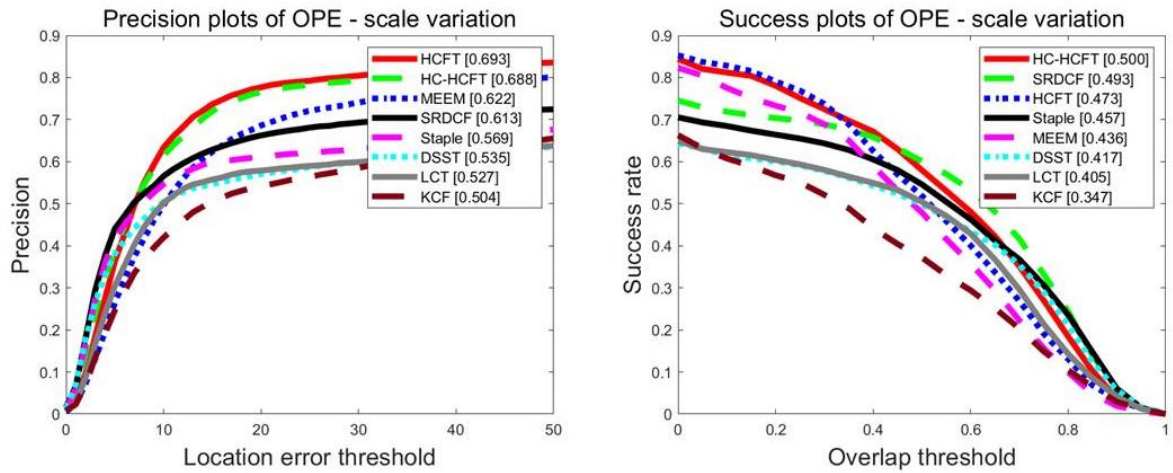


图 4.5 尺度变化——跟踪算法精度图与成功图

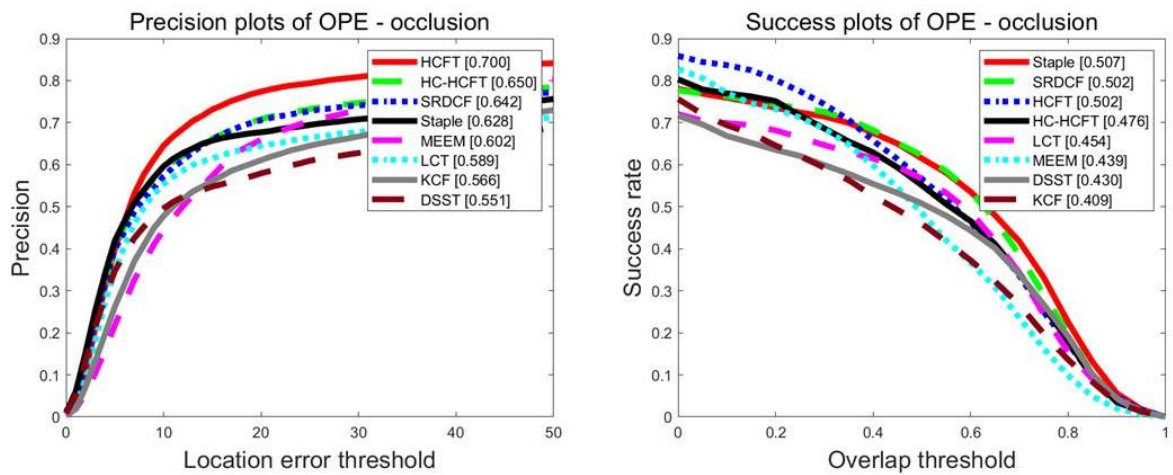


图 4.6 目标遮挡——跟踪算法精度图与成功图

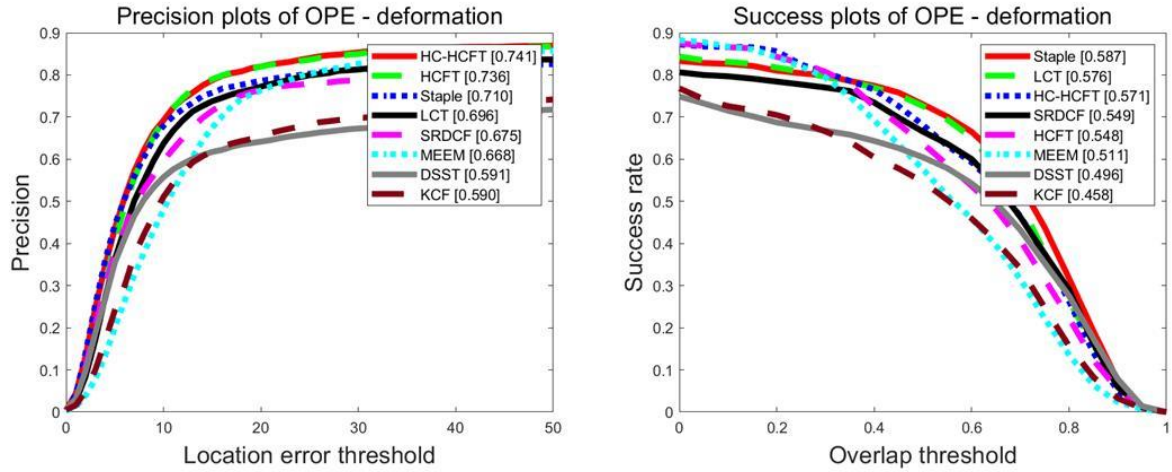


图 4.7 目标形变——跟踪算法精度图与成功图

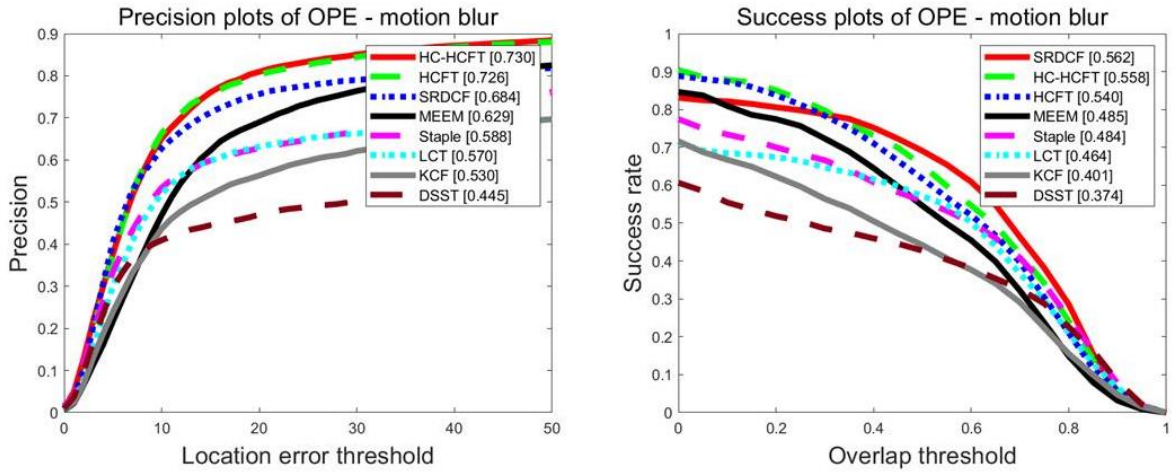


图 4.8 运动模糊——跟踪算法精度图与成功图

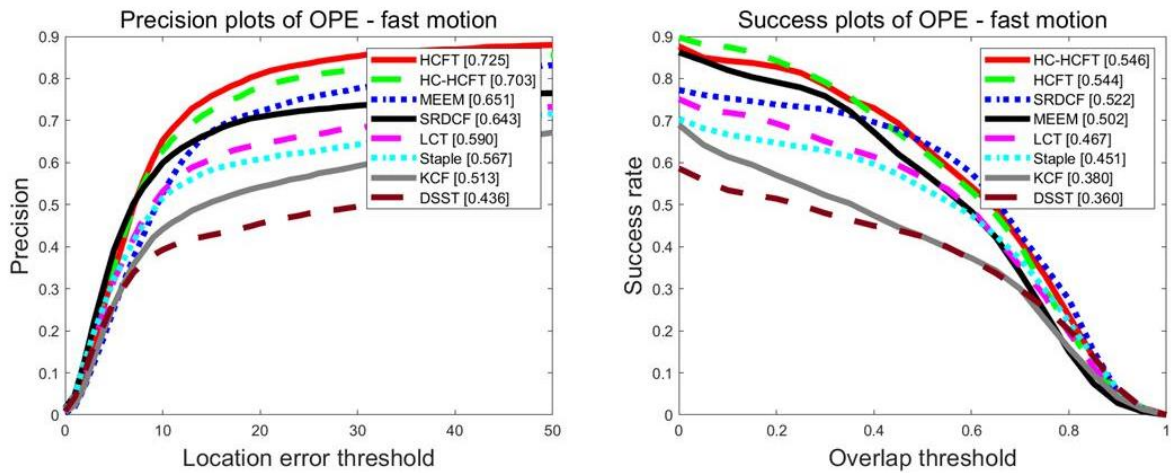


图 4.9 快速运动——跟踪算法精度图与成功图

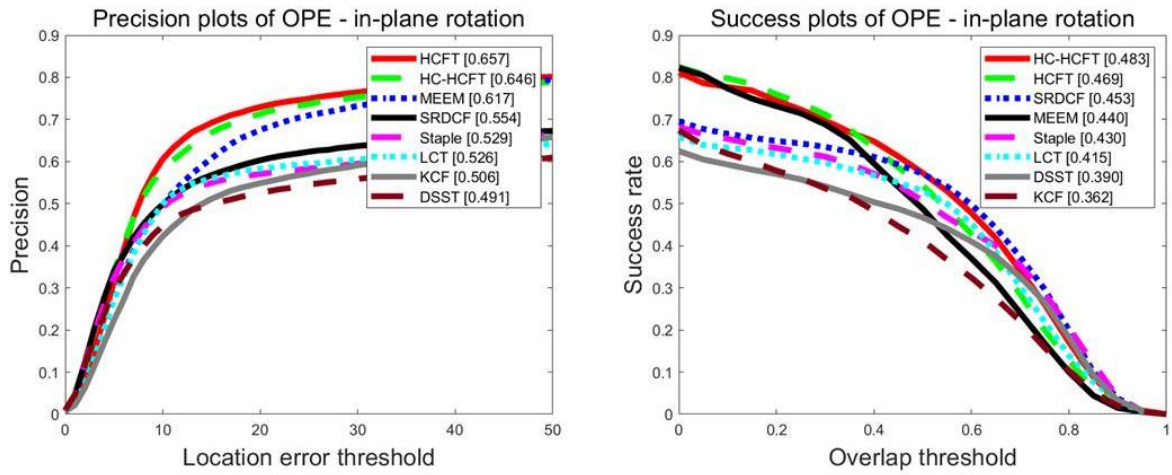


图 4.10 平面内旋转——跟踪算法精度图与成功图

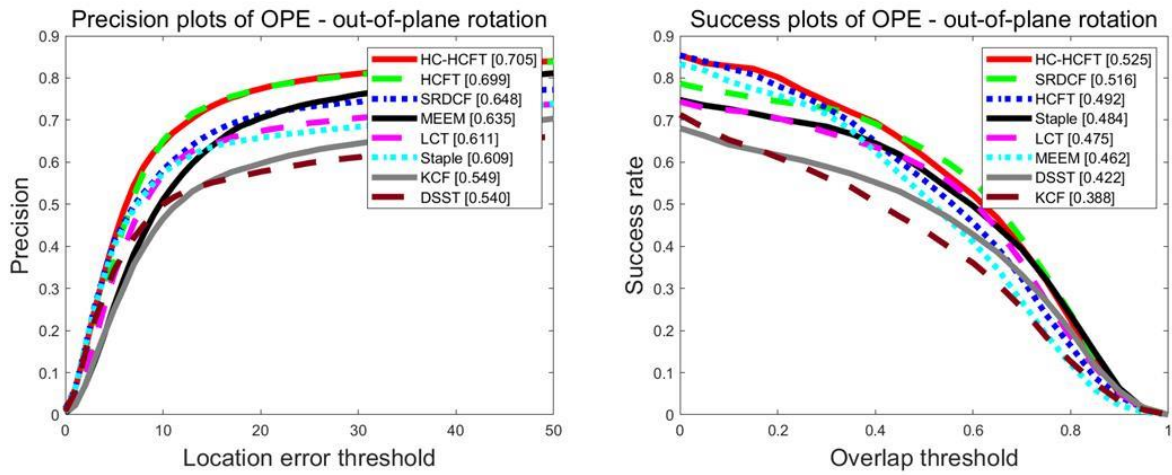


图 4.11 平面外旋转——跟踪算法精度图与成功图

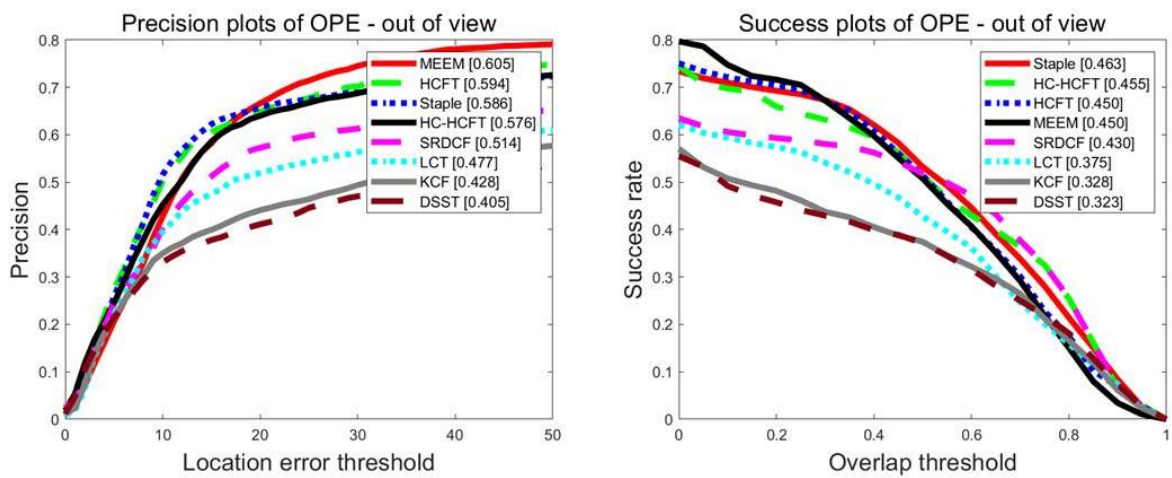


图 4.12 目标越界——跟踪算法精度图与成功图

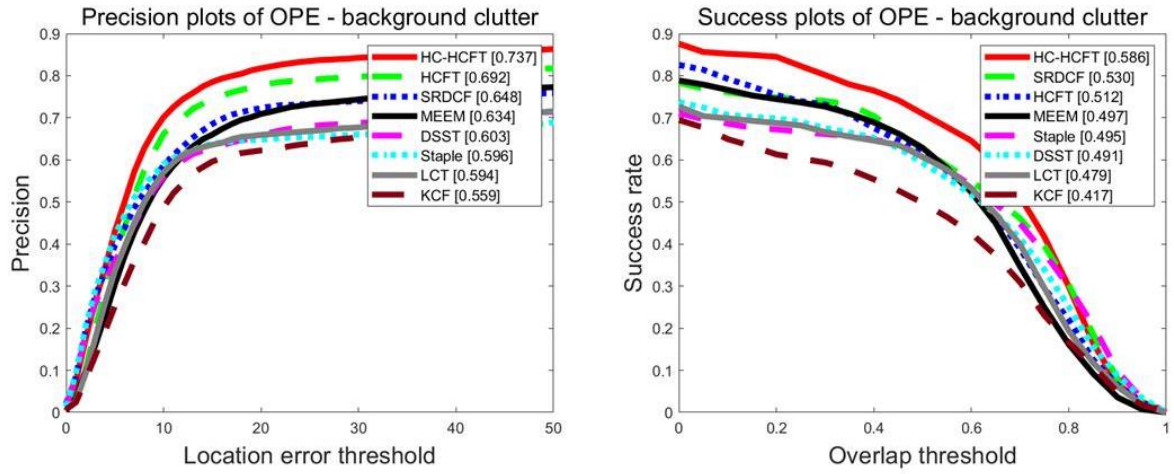


图 4.13 背景驳杂——跟踪算法精度图与成功图

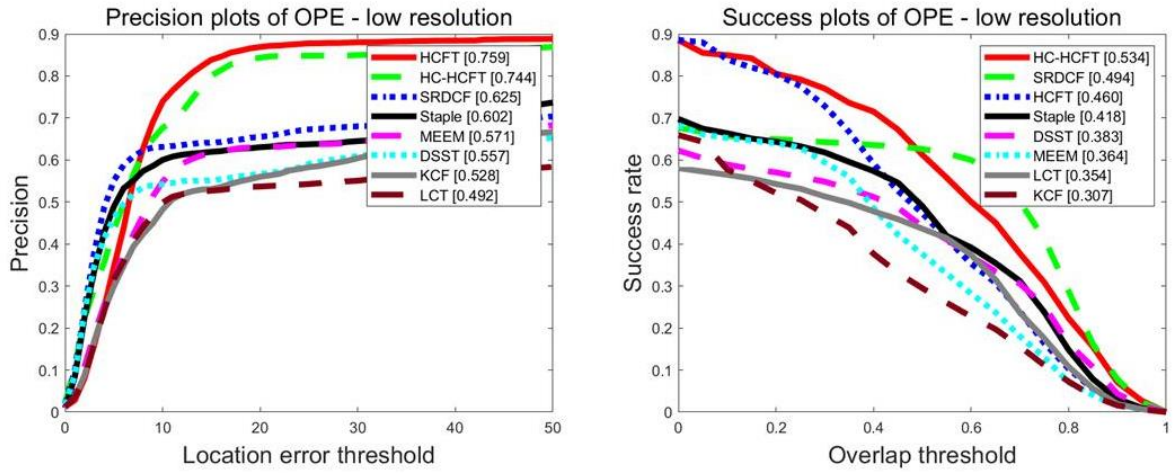


图 4.14 低分辨率——跟踪算法精度图与成功图

第 5 章 结论

5.1 研究工作总结

目标跟踪是计算机视觉的重要研究课题之一，得到了工业界和学术界的高度关注。本文在阐述目标跟踪的基本概念以及梳理其历史发展脉络的基础上，选择了精度高、速度快的相关滤波类跟踪算法为研究对象，研究成果包括以下几个部分：

（1）将相关滤波的基准算法 MOSSE、以及运用循环矩阵加速的 CSK、KCF 算法统一起来，对相关滤波类算法框架进行了深入的分析，对重要的算法原理进行详尽的推导。并在此基础上，总结出 KCF 包括尺度、特征、模板更新等方面的缺陷。

（2）针对 KCF 的上述缺陷，受 HCFT 算法的启发，提出了一种高置信度目标跟踪算法 HC-HCFT。该算法引入深度特征来克服 HOG 特征在目标表达方面的不足，引入 LMCT 的高置信度指标 APCE 来增强其在目标尺度自适应、长时跟踪、模板更新方面的能力。

（3）选取了若干先进或典型的判别式跟踪算法，在 OTB50 标准数据集上进行了对比实验，实验结果表明改进的算法在跟踪准确率与成功率上相较于其他判别式算法更具竞争力，且通过 OTB50 全部 11 种属性视频的测试，也证明了改进算法具有普遍的适应性。

5.2 未来工作展望

诚然，深度特征对目标模板强大的表达有效地提高了跟踪的准确率与成功率，尤其是对于 KCF 难以胜任的目标形变、遮挡、光照等跟踪场景。但由于深度网络往往体量庞大，在算法特征提取环节占用了太多的时间，使得算法难以达到实时性的要求。因此，在实验结果分析时，并没有对 FPS 进行讨论。因为深度特征与传统特征提取的速度往往天差地别，这正是深度特征在目标跟踪的应用过程中最大的阻碍。

基于实时性的考虑，越来越多的研究者已经把注意力转移至小体量深度网络和微调的深度网络上。在尽可能少的牺牲跟踪精度的基础上，尽可能高的提升特征提取的速度成为了重要的研究思路。此外，深度特征仅仅是深度网络的冰山一角，利用大量的正负样本直接训练一个跟踪网络也成为了热门的研究方向^[33]。限于本文篇幅与研究基础，对相关滤波类跟踪更多的内容不再扩展，有兴趣的读者可以持续跟进牛津大学 Joao F. Henriques、林雪平大学 Martin Danelljan 等课题组的最新研究^①。

^① Joao F. Henriques & Martin Danelljan: <http://www.robots.ox.ac.uk/~joao/>. & <http://users.isy.liu.se/en/cvl/marda26/>.

致 谢

感谢岁月山河，众生万象。

吴 麟

2019 年 6 月 10 日

于小米武汉总部

参考文献

- [1] Wax N. Signal - to - Noise Improvement and the Statistics of Track Populations[J]. Journal of Applied Physics, 1955,26(5):586-595.
- [2] Kalman R E. A New Approach to Linear Filtering and Prediction Problems[J]. Journal of Basic Engineering Transactions, 1960,82:35-45.
- [3] Sittler R W. An Optimal Data Association Problem in Surveillance Theory[J]. IEEE Transactions on Military Electronics, 2007,8(2):125-139.
- [4] Smeulders A W M, Chu D M, Cucchiara R, et al. Visual Tracking: An Experimental Survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014,36(7):1442-1468.
- [5] Bolme D S, Beveridge J R, Draper B A, et al. Visual Object Tracking using Adaptive Correlation Filters[M]//IEEE Conference on Computer Vision and Pattern Recognition. 2010:2544-2550.
- [6] 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.
- [7] Danelljan M A H G. Accurate Scale Estimation for Robust Visual Tracking[M]//British Machine Vision Conference, 2014.
- [8] Tang M, Feng J. Multi-kernel Correlation Filter for Visual Tracking[M]//IEEE International Conference on Computer Vision. 2015:3038-3046.
- [9] Ting L, Gang W, Qingxiong Y. Real-time part-based visual tracking via adaptive correlation filters[M]//IEEE Conference on Computer Vision and Pattern Recognition. 2015:4902-4912.
- [10] Oron S, Bar-Hillel A, Levi D, et al. Locally Orderless Tracking[J]. International Journal of Computer Vision, 2015,111(2):213-228.
- [11] Ross D A, Lim J, Lin R, et al. Incremental learning for robust visual tracking[J]. International Journal of Computer Vision, 2008,77(1-3):125-141.
- [12] Kwon J, Lee K M. Visual Tracking Decomposition[M]//IEEE Conference on Computer Vision and Pattern Recognition. 2010:1269-1276.
- [13] Henriques J F, Caseiro R, Martins P, et al. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels[M]// European Conference on Computer Vision. 2012:702-715.
- [14] B. Schölkopf A S. Learning with kernels: Support vector machines, regularization, optimization, and beyond [M]. USA: MIT Press, 2002:88-90.
- [15] Asha C S, Narasimhadhan A V. Adaptive Learning Rate for Visual Tracking using Correlation Filters[M]// European Conference on Computer Vision 2016:614-622.
- [16] Danelljan M, Khan F S, Felsberg M, et al. Adaptive Color Attributes for Real-Time Visual Tracking[M]//IEEE Conference on Computer Vision and Pattern Recognition. 2014:1090-1097.
- [17] Ma C, Huang J, Yang X, et al. Hierarchical Convolutional Features for Visual Tracking[M]//IEEE International Conference on Computer Vision. 2015:3074-3082.
- [18] Danelljan M, Hager G, Khan F S, et al. Convolutional Features for Correlation Filter Based Visual Tracking[M]// IEEE International Conference on Computer Vision. 2015:621-629.

-
- [19] Valmadre J, Bertinetto L, Henriques J, et al. End-to-end representation learning for Correlation Filter based tracking[M]//IEEE Conference on Computer Vision and Pattern Recognition. 2017:5000-5008.
 - [20] Nam H, Han B. Learning Multi-Domain Convolutional Neural Networks for Visual Tracking[M]//IEEE Conference on Computer Vision and Pattern Recognition. 2016:4293-4302.
 - [21] Chao M, Yi X, Ni B, et al. When Correlation Filters Meet Convolutional Neural Networks for Visual Tracking[J]. IEEE Signal Processing Letters, 2016,23(10):1454-1458.
 - [22] Chao M, Huang J B, Yang X, et al. Robust Visual Tracking via Hierarchical Convolutional Features[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017,PP(99):1.
 - [23] Wang M, Yong L, Huang Z. Large Margin Object Tracking with Circulant Feature Maps[M]//IEEE Conference on Computer Vision and Pattern Recognition. 2017.
 - [24] Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position[J]. Biological Cybernetics, 1980,36(4):193-202.
 - [25] Danelljan M, Hager G, Khan F S, et al. Discriminative Scale Space Tracking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017,39(8):1561-1575.
 - [26] Huang D. Enable Scale and Aspect Ratio Adaptability in Visual Tracking with Detection Proposals[M]//British Machine Vision Conference, 2015.
 - [27] Zitnick C L, Dollár P. Edge Boxes: Locating Object Proposals from Edges[M]//European Conference on Computer Vision, 2014.
 - [28] Yi W, Lim J, Yang M H. Online Object Tracking: A Benchmark[M]//IEEE Conference on Computer Vision and Pattern Recognition, 2013.
 - [29] Wu Y, Lim J, Yang M. Object Tracking Benchmark[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015,37(9):1834-1848.
 - [30] Kristan M, Pflugfelder R, Leonardis A, et al. The Visual Object Tracking VOT2014 Challenge Results[M]//European Conference on Computer Vision, Zurich[C].
 - [31] Kristan, Matej, Pflugfelder, et al. The Visual Object Tracking VOT2013 Challenge Results[M]//IEEE International Conference on Computer Vision Workshops, 2013.
 - [32] 管皓, 薛向阳, 安志勇. 深度学习在视频目标跟踪中的应用进展与展望[J]. 自动化学报, 2016,42(06):834-847.
 - [33] Bertinetto L, Valmadre J, Henriques J F, et al. Fully-Convolutional Siamese Networks for Object Tracking[M]//European Conference on Computer Vision, 2016.

附 录

HC-HCFT 的 OTB50 配置程序如下，完整程序见本人 Github^①。

```

58 lines (38 sloc) | 1.61 KB
Raw Blame History
1 function results=run_HC_HCFT(seq, res_path, bSaveImage)
2
3 close all
4
5 s_frame = seq.s_frames;
6
7 dbstop if error;
8
9 addpath('utility','train');
10
11 % Path to MatConvNet. Please run external/matconvnet/vl_compilenn.m to
12 % set up the MatConvNet
13
14 addpath './matconvnet/matlab'
15 vl_setupnn();
16
17 % Where the 'imagenet-vgg-verydeep-19.mat' file is
18 addpath './vgg_model'
19
20 addpath(genpath('edgesbox'));
21 addpath(genpath('piotr_toolbox'));
22 addpath(genpath('Diagnose'))
23
24 % Extra area surrounding the target
25 padding = struct('generic', 1.8, 'large', 1, 'height', 0.4);
26
27 lambda = 1e-4; % Regularization parameter (see Eqn 3 in our paper)
28 output_sigma_factor = 0.1; % Spatial bandwidth (proportional to the target size)
29
30 interp_factor = 0.01; % Model learning rate (see Eqn 6a, 6b)
31 config.kernel_sigma = 1;
32 config.motion_thresh= 0.181; %0.25 for singer2 0.32;%0.15
33 config.appearance_thresh=0.38; %0.38
34 config.features.hog_orientations = 9;
35 config.features.cell_size = 4; % size of hog grid cell
36 config.features.window_size = 6; % size of local region for intensity histogram
37 config.features.nbins=8;
38 global enableGPU;
39 enableGPU = false;
40
41
42 target_sz = [seq.init_rect(1,4), seq.init_rect(1,3)];
43 pos = [seq.init_rect(1,2), seq.init_rect(1,1)] + floor(target_sz/2);
44
45 [rect_position, time] = tracker_HC_HCFT(s_frame, pos, target_sz, ...
46 padding, lambda, output_sigma_factor, interp_factor,config); %tracker_ensemble_RPnew1
47
48 fps = numel(s_frame)/time;
49
50 disp(['fps: ' num2str(fps)])
51
52 results.type = 'rect';
53 results.res = rect_position;%each row is a rectangle
54 results.fps = fps;
55
56 end

```

^① 程序下载地址: <https://github.com/FiftyWu/OTB50>.