文章编号:1007-5321(2020)02-0129-06 **DOI**:10.13190/j.jbupt.2019-071

一种基于 ResNet 网络特征的视觉目标跟踪算法

马素刚[1,2], 赵祥模[1], 侯志强[2], 王忠民[2,3], 孙韩林[2]

(1. 长安大学 信息工程学院, 西安 710064; 2. 西安邮电大学 计算机学院, 西安 710121;
3. 西安邮电大学 陕西省网络数据分析与智能处理重点实验室, 西安 710121)

摘要: 针对复杂场景下目标容易丢失的问题,提出了一种基于深度残差网络(ResNet)特征的尺度自适应视觉目标跟踪算法. 首先,通过 ResNet 提取图像感兴趣区域的多层深度特征,考虑到修正线性单元(ReLU)激活函数对目标特征的抑制作用,在 ReLU 函数之前选取用于提取目标特征的卷积层;然后,在提取的多层特征上分别构建基于核相关滤波的位置滤波器,并对得到的多个响应图进行加权融合,选取响应值最大的点即为目标中心位置. 目标位置确定后,对目标进行多个尺度采样,分别提取不同尺度图像的方向梯度直方图(fHOG)特征,在此基础上构建尺度相关滤波器,从而实现对目标尺度的准确估计. 在视频集 OTB100 中与其他 6 种相关算法进行了比较,实验结果表明,所提算法取得了较高的跟踪成功率和精确度,能够较好地适应目标的尺度变化、背景干扰等复杂场景.

关 键 词: 视觉目标跟踪; 深度残差网络; 核相关滤波; 深度学习; 尺度估计

中图分类号: TP391.4 **文献标志码**: A

A Visual Object Tracking Algorithm Based on Features Extracted by Deep Residual Network

MA Su-gang[1,2], ZHAO Xiang-mo[1], HOU Zhi-qiang[2], WANG Zhong-min[2,3], SUN Han-lin[2]

(1. School of Information Engineering, Chang'an University, Xi'an 710064, China;
2. School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an 710121, China;
3. Shanxi Key Laboratory of Network Data Analysis and Intelligent Processing, Xi'an University of Posts and Telecommunications, Xi'an 710121, China)

Abstract: Because the objects are easy to be lost in complex scenes, a scale adaptive visual object tracking algorithm based on deep residual network (ResNet) features is proposed. Firstly, the ResNet is used to extract the multi-layer deep features of the image region of interest. Considering the restraining effect of rectified linear units (ReLU) activation function on target features, only the convolutional layers before ReLU function are selected. Secondly, the translation filters based on kernelized correlation filter are constructed in the extracted multi-layer features, and then the weighted fusion of the multiple response maps is carried out to obtain the target position with the largest response value. After the target location is determined, the target is sampled at multiple scales, and the felzenszwalb histogram of oriented gradients (fHOG) features of different scale images are extracted separately. On this basis, a scale correlation filter is constructed to estimate the target scale accurately. Comparing with six related algorithms in OTB100, an experiment is carried. It is shown that the proposed algorithm achieves high tracking success

收稿日期: 2019-04-30

基金项目: 国家自然科学基金项目(61571458, 61473309); 陕西省重点研发计划项目(2018ZDCXL-GY-04-02); 陕西省教育厅专项科研计划项目(17JK0696); 西安市科技计划项目(GXYD17.17)

作者简介: 马素刚(1982—), 男, 高级工程师, E-mail: msg@xupt.edu.cn.

rate and accuracy, and can adapt to scale variation, background clutter and other complex scenes.

Key words: visual object tracking; deep residual network; kernelized correlation filter; deep learning; scale estimation

近年来,目标跟踪问题一直是计算机视觉领域研究的热点[1]. 根据观测模型不同,目标跟踪算法可分为生成式模型算法和判别式模型算法2类. 典型的生成式模型算法有均值漂移、粒子滤波等. 判别式模型算法. 如深度学习跟踪(DLT, deep learning tracker)、核相关滤波(KCF, kernelized correlation filter)、卷积神经网络—支持向量机(CNN-SVM, convolutional neural network-support vector machine)等,已经成为解决目标跟踪问题的主流方法.

Henriques 等[2]提出的 KCF 算法利用目标及其周围区域,通过构建循环矩阵采集正负样本,提取多通道方向梯度直方图(fHOG, felzenszwalb histogram of oriented gradients)特征,采用岭回归训练目标滤波器,并成功地利用循环矩阵在频域可对角化的性质,将矩阵的运算转化为向量元素的点乘,大大降低了运算量,提高了运算速度,使算法能够满足实时性要求. 但是,KCF 算法没有考虑尺度变化,如果目标缩小,滤波器就会学习到大量背景信息;如果目标扩大,滤波器只学习了目标的局部纹理. 这2种情况都很可能导致跟踪漂移. Ma 等[3]提出的分层卷积特征(HCF, hierarchical convolutional features)算法,其基本框架就是 KCF,不同的是,HCF 算法使用深度卷积网络特征.

判别式尺度空间跟踪(DSST, discriminative scale space tracker)算法[4]将目标跟踪看成目标中心平移和目标尺度变化2个独立问题,依据最小输出均方误差和(MOSSE, minimum output sum of squared error)算法中提出的相关滤波器(CF, correlation filter),进而设计了2个相互独立的 CF,即位置滤波器和尺度滤波器,分别用于实现目标定位和尺度估计. Li 等[5]提出的尺度自适应多特征(SAMF, scale adaptive multiple feature)算法,仍然基于 CF 框架,但是将单一特征扩展为多个特征,实现了灰度特征、方向梯度直方图(HOG)特征和颜色特征融合.

结合深度学习技术是目前目标跟踪算法的主要方向,如 HCF[3]、CNN-SVM、全卷积孪生网络(SiamFC, fully-convolutional siamese networks)算法等. HCF 算法通过预训练的网络模型 VGG-19 提取被跟踪目标的深度特征,利用 CF 确定目标位置. Hong 等[6]提出的 CNN-SVM 算法,以卷积神经网络全连接层的输出为目标特征,结合在线支持向量机构建了目标跟踪模型. Bertinetto 等[7]提出的 SiamFC 算法原理与 CF 很相似,都是通过比较搜索区域与目标模板的相似度,把相似度值最大的点作为新的目标中心,不同的是,SiamFC 算法采用卷积操作计算相似度值. SiamFC 算法采用5个尺度,SiamFC-3s 算法采用了3个尺度.

笔者受文献[3-4]的启发,同时考虑到深度残差网络(ResNet, deep residual network)[8]较强的特征表示能力,提出了一种基于 ResNet 深度特征的尺度自适应目标跟踪算法.

1 相关技术基础

1.1 ResNet

卷积神经网络(CNN, convolutional neural network)是一种典型的深度神经网络,2012 年以来先后出现了多个 CNN 模型,如 AlexNet、ZFNet、GoogLeNet、VGGNet、ResNet、DenseNet 等. He 等[8]提出了 ResNet 结构,通过在网络中增加恒等映射,很好地解决了深度网络中梯度消失问题,使得网络性能能够在网络深度增加时得到进一步提升. ResNet 层数可以选择18、34、50、101、152 等. 在综合考虑网络各层的特征表示能力以及计算量等因素基础上,笔者选取了 ResNet-50 结构(见图1),该网络的卷积层分为5个部分,即 Conv1、Conv2、Conv3、Conv4 和 Conv5,每部分包含数量不等的子部分(用小圆圈表示),子部分的数量依次为1、3、

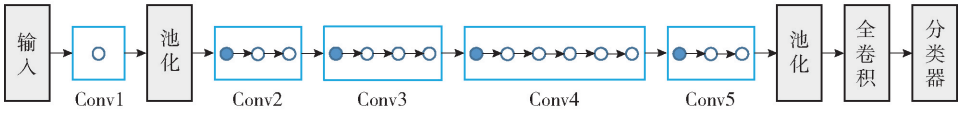


图1 ResNet-50 结构

4、6、3. 子部分进一步表示多个卷积操作，其中实心小圆圈包含 4 个卷积操作，空心小圆圈包含 3 个卷积操作.

1.2 KCF

CF 的基本思想是：设计一个滤波模板，下一帧中多个候选区域与该模板进行卷积运算，输出响应最大的区域即为目标区域. KCF 通过引入核函数，将训练的滤波模板变为一个非线性二分类器，以判别候选区域是目标还是背景.

核相关滤波器 α 可以表示为[2]

$$\hat{\boldsymbol{\alpha}} = \frac{\hat{\boldsymbol{y}}}{\hat{\boldsymbol{k}}^{xx} + \lambda} \tag{1}$$

其中：$\boldsymbol{y}$ 为通过高斯函数构建的响应值，$\hat{\boldsymbol{y}}$ 为 $\boldsymbol{y}$ 的傅里叶变换，$\boldsymbol{k}^{xx}$ 的取值由核函数确定，λ 为正则项. $\boldsymbol{y}$、$\boldsymbol{k}^{xx}$ 和 $\boldsymbol{\alpha}$ 均为二维矩阵，λ 为常数.

如果选择高斯核，则 $\boldsymbol{k}^{xx}$ 为

$$\boldsymbol{k}^{xx} = \exp\left(-\frac{1}{\sigma^2}(\|\boldsymbol{x}\|^2 + \|\boldsymbol{x}\|^2 - 2\mathscr{F}^{-1}(\hat{\boldsymbol{x}}^* \odot \hat{\boldsymbol{x}}))\right) \tag{2}$$

其中：exp 表示指数函数，如果变量为矩阵，则对矩阵每个元素进行指数运算；$\|\cdot\|$ 表示向量的 2-范数；$\boldsymbol{x}$ 为样本特征；$\hat{\boldsymbol{x}}^*$ 表示 $\hat{\boldsymbol{x}}$ 的共轭；σ 为常数；$\mathscr{F}^{-1}$ 表示傅里叶逆变换；⊙表示点乘运算.

2 基于 ResNet 深度特征的尺度自适应目标跟踪算法

2.1 算法总体结构

所提算法的总体结构如图 2 所示，分为目标定位和尺度估计 2 个阶段，并利用跟踪结果对位置滤波器和尺度滤波器进行模型更新.

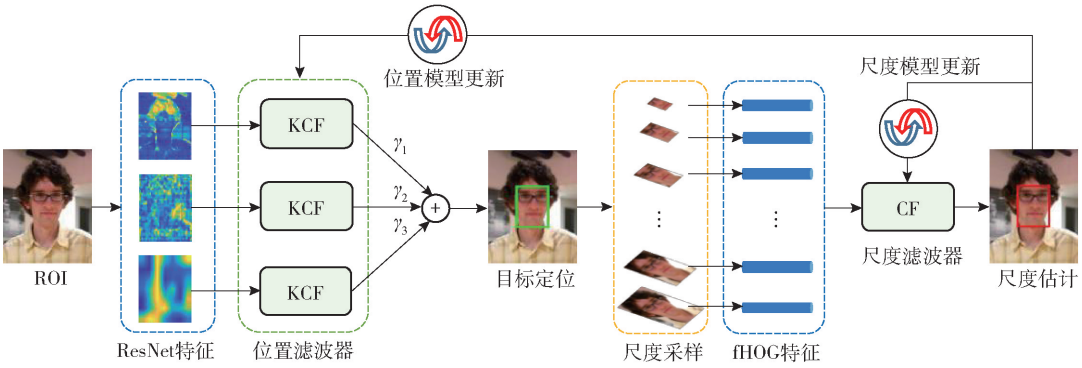


图 2　所提算法总体结构

2.2 目标定位

假设输入图像中的感兴趣区域(ROI, region of interest)大小为 $m \times n \times 3$，经过 ResNet 提取后，得到的特征大小分别为 $m_1 \times n_1 \times l_1$、$m_2 \times n_2 \times l_2$ 和 $m_3 \times n_3 \times l_3$，$l_i$ 表示第 $i(i=1,2,3)$ 个特征通道数. 每个特征分别经过 KCF，得到的响应图大小均为 $s \times t$，然后对各个响应图加权求和，对应权值为 γ_i，在融合后的响应图中值最大的点对应目标的中心位置.

所提算法在目标定位中，通过逐步增加特征层的数量，并经过大量实验验证，最终选择 3 层融合对目标特征进行表示，取得了较好的跟踪效果.

2.2.1 单层特征提取

在 ResNet-50 中由低至高选择 5 层，分别为 Conv1、Conv2-1、Conv3-1、Conv4-1 和 Conv5-1. 在 OTB100[9] 上的测试结果显示，利用 Conv4-1 层提取的特征进行跟踪时，成功率和精确度均为最高.

考虑到激活函数 ReLU 对目标特征的抑制作用，继续在 Conv4-1 之前的层 Conv4-1-p1、Conv4-1-p2 和 Conv4-1-p3 上进行测试，如图 3 所示. 综合考虑成功率和精确度可以看出，Conv4-1-p3 取得了更好的跟踪效果，如表 1 所示.

表 1　ResNet 不同层特征的跟踪效果对比

ResNet 层	成功率	精确度
Conv4-1	0.551	0.776
Conv4-1-p1	0.555	0.780
Conv4-1-p2	0.551	0.766
Conv4-1-p3	**0.555**	**0.784**

2.2.2 两层特征融合

在 Conv4 内部，利用 Conv4-1-p3 与其他层融合进行测试(见图 3).

在 OTB100 上的测试结果(见表 2)表明，融合 ResNet 中 Conv4-1-p3 与 Conv4-5-p1 两层特征时，跟

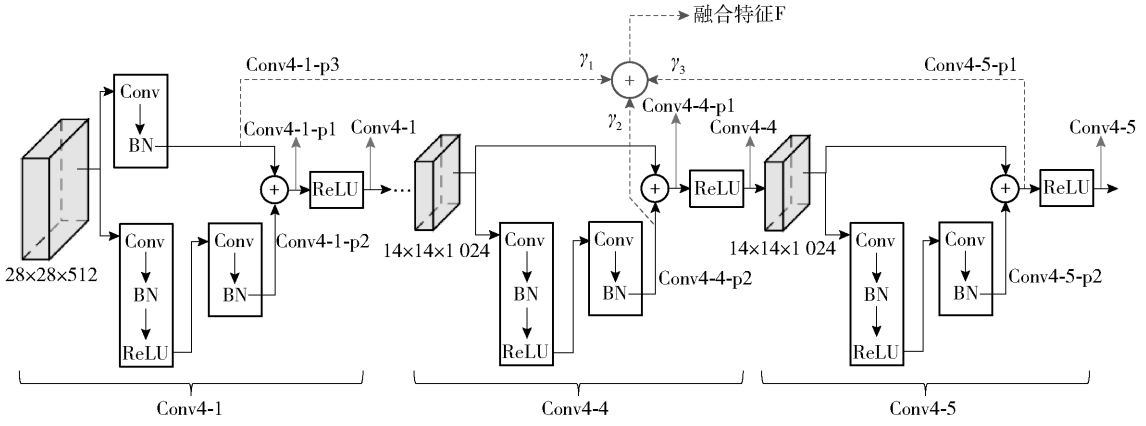


图 3 用于特征提取的 ResNet 层

踪成功率和精确度均高于只提取 1 层特征 Conv4-1-p3 的跟踪结果. 而且,与 Conv4-5-p1 融合的跟踪效果优于经过激活函数 ReLU 的 Conv4-5 层.

表 2 融合 ResNet 两个特征层的跟踪效果对比

ResNet 层	成功率	精确度
Conv4-1-p3 + Conv4-2	0. 560	0. 785
Conv4-1-p3 + Conv4-3	0. 566	0. 805
Conv4-1-p3 + Conv4-4	0. 576	0. 816
Conv4-1-p3 + Conv4-5	0. 579	0. 814
Conv4-1-p3 + Conv4-6	0. 578	0. 813
Conv4-1-p3 + Conv4-5-p1	**0. 591**	**0. 829**
Conv4-1-p3 + Conv4-5-p2	0. 569	0. 800

2. 2. 3 3 层特征融合

为进一步提高深度特征的表示能力,在融合 Conv4-1-p3 和 Conv4-5-p1 的基础上,尝试继续增加其他层. 对跟踪结果中精确度比较低的视频进行了分析,并经过大量的实验验证,选定 Conv4-1-p3、Conv4-4-p2 和 Conv4-5-p1 三层(图 3 中虚线标注)进行融合,跟踪效果得到进一步提升. 图中“BN”表示批量归一化操作.

实验过程中发现,3 层特征融合时,采用不同的权重,跟踪效果不同,如表 3 所示. 因此,通过对跟踪权重进行优化,能获得更好的跟踪性能. 实验中权重的取值被限制在 0 ~ 1 之间,通过大量实验得出的所提算法融合权重分别为 $\gamma_1 = 1, \gamma_2 = 0.2, \gamma_3 = 1$.

权重取值相同时,对 ReLU 前后的层进行比较,如表 4 所示. 可以看出,Conv4-1-p3、Conv4-4-p2 和 Conv4-5-p1 三层融合时,取得了最好的跟踪效果,且这 3 层的位置均位于激活函数 ReLU 之前.

表 3 不同权重下的跟踪效果对比

权重			成功率	精确度
Conv4-1-p3	Conv4-4-p2	Conv4-5-p1		
1	0. 1	1	0. 583	0. 819
1	**0. 2**	**1**	**0. 596**	**0. 840**
1	0. 5	1	0. 574	0. 815
1	0. 7	1	0. 591	0. 834
0. 8	0. 2	1	0. 579	0. 818
1	0. 2	0. 8	0. 591	0. 828

表 4 融合 ResNet 3 个特征层的跟踪效果对比

ResNet 层	成功率	精确度
Conv4-1-p3 + Conv4-4-p2 + Conv4-5-p1	**0. 596**	**0. 840**
Conv4-1-p3 + Conv4-4-p2 + Conv4-5	0. 588	0. 819
Conv4-1-p3 + Conv4-4 + Conv4-5-p1	0. 580	0. 811
Conv4-1-p3 + Conv4-4 + Conv4-5	0. 569	0. 804
Conv4-1 + Conv4-4-p2 + Conv4-5-p1	0. 583	0. 825
Conv4-1 + Conv4-4-p2 + Conv4-5	0. 577	0. 801
Conv4-1 + Conv4-4 + Conv4-5-p1	0. 563	0. 799
Conv4-1 + Conv4-4 + Conv4-5	0. 564	0. 793

2. 3 尺度估计

依据目标中心位置进行尺度采样,得到 33 个不同尺度的样本,把所有样本变换成相同大小,分别提取每个样本的 fHOG 特征(共有 d 维). 33 个特征向量经过尺度滤波器,响应值最大的点对应当前帧中目标的精确尺度.

尺度滤波器表示为[4]

$$\boldsymbol{H}^l = \frac{\boldsymbol{G}^* \odot \boldsymbol{F}^l}{\sum_{k=1}^{d} ((\boldsymbol{F}^k)^* \odot \boldsymbol{F}^k) + \lambda} \tag{3}$$

其中:$\boldsymbol{G}$ 为利用高斯函数构建的响应值,$\boldsymbol{G}^*$ 表示 $\boldsymbol{G}$ 的共轭,$\boldsymbol{F}^l$ 表示第 l 维特征的傅里叶变换,d 为特征

维数.

尺度滤波器的分子项、部分分母项分别为

$$\boldsymbol{A}^l = \boldsymbol{G}^* \odot \boldsymbol{F}^l \tag{4}$$

$$\boldsymbol{B} = \sum_{k=1}^{d} ((\boldsymbol{F}^k)^* \odot \boldsymbol{F}^k) \tag{5}$$

响应值 $\boldsymbol{y}$ 为

$$\boldsymbol{y} = \mathscr{F}^{-1}\left(\frac{\sum_{l=1}^{d}((\boldsymbol{A}^l)^* \odot \boldsymbol{Z}^l)}{\boldsymbol{B}+\lambda}\right) \tag{6}$$

$\boldsymbol{y}$ 中最大值位置对应目标最佳尺度. $\boldsymbol{Z}^l$ 表示输入图像第 l 维特征的傅里叶变换.

2.4　模型更新

确定第 t 帧图像中目标的位置和尺度后,为了使得跟踪算法更加鲁棒,需要在第 $t+1$ 帧跟踪前对位置滤波器和尺度滤波器分别进行更新.

参照式(1),位置模型更新策略为

$$\boldsymbol{\alpha}_t = (1-\eta)\boldsymbol{\alpha}_{t-1} + \eta\boldsymbol{\alpha}(t) \tag{7}$$

其中:$\boldsymbol{\alpha}_{t-1}$为对第 t 帧图像跟踪前求得的滤波器模板,$\boldsymbol{\alpha}(t)$为根据第 t 帧图像求得的滤波器模板,η 为位置滤波器的学习率.

参照式(3),对第 t 帧图像跟踪后,尺度模型更新策略为

$$\boldsymbol{A}_t^l = (1-\eta')\boldsymbol{A}_{t-1}^l + \eta'\boldsymbol{G}_t^* \odot \boldsymbol{F}_t^l \tag{8}$$

$$\boldsymbol{B}_t = (1-\eta')\boldsymbol{B}_{t-1} + \eta' \sum_{k=1}^{d} (\boldsymbol{F}_t^k)^* \odot \boldsymbol{F}_t^k \tag{9}$$

其中 η'为尺度滤波器的学习率.

3　实验结果分析

3.1　实验环境与参数选择

在 Windows 7 系统下,采用 Matlab 和 C++混合编程实现所提算法. 硬件平台配置为:2 块 2.4 GHz CPU,64 GB 内存,1 块 Nvidia GTX 1080Ti GPU. 位置滤波器的学习率 $\eta=0.01$,正则项 $\lambda=10^{-4}$. 尺度滤波器的尺度因子 $a=1.02$,采样个数 $S=33$,学习率 $\eta'=0.025$,正则项 $\lambda=10^{-4}$.

3.2　算法性能分析

在视频集 OTB100[9] 中,选取具有尺度变化属性的 64 个视频进行测试. 表 5、表 6 详细列出了 11 种属性下算法的成功率和精确度,最优结果用粗体标注,次优结果用斜体表示. 表头的缩写字母表示视频的不同属性,括号内的值表示 64 个视频中具有对应属性的视频个数. 可以看出,除了具有 OV 属性的视频集外,所提算法的成功率和精确度均为最优或次优. 这主要是由于所提算法采用了具有较强特征表示能力的 ResNet 提取目标特征,同时考虑了目标尺度的变化.

综合以上分析,所提出的算法在具有尺度变化属性的视频集上表现出较好的综合跟踪性能,同时在光照变化、目标遮挡等复杂环境下仍然具有较好的鲁棒性.

在 OTB100 的所有视频上对相关算法进行测

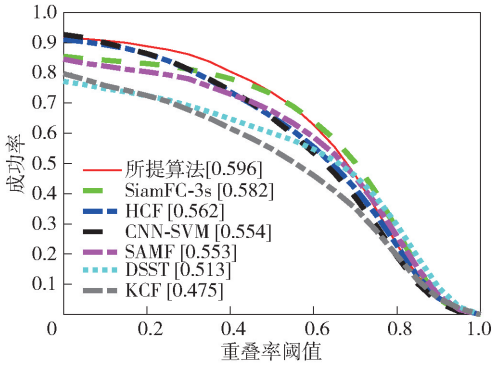
表 5　不同属性下跟踪成功率对比

算法	SV (64)	IV (24)	OCC (33)	DEF (29)	MB (21)	FM (28)	IPR (35)	OPR (45)	OV (11)	BC (17)	LR (8)
所提算法	**0.568**	**0.601**	*0.520*	**0.563**	**0.573**	**0.560**	**0.558**	**0.553**	0.456	**0.580**	*0.465*
HCF	0.485	0.481	0.434	0.465	*0.530*	0.526	0.517	0.482	0.460	*0.491*	0.402
CNN-SVM	0.490	0.470	0.454	*0.504*	0.526	0.501	0.484	0.495	*0.483*	0.476	0.363
SiamFC-3s	*0.552*	*0.529*	**0.522**	0.498	0.485	*0.533*	*0.534*	*0.546*	**0.510**	0.465	**0.574**
SAMF	0.495	0.471	0.470	0.474	0.468	0.470	0.476	0.499	0.477	0.429	0.398
DSST	0.468	0.499	0.425	0.408	0.419	0.411	0.451	0.444	0.377	0.449	0.352
KCF	0.394	0.374	0.369	0.392	0.394	0.418	0.388	0.399	0.384	0.363	0.268

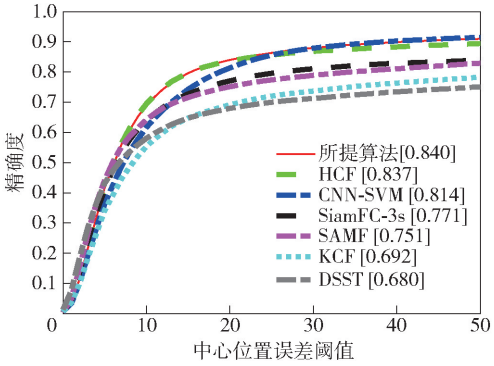
表 6　不同属性下跟踪精确度对比

算法	SV (64)	IV (24)	OCC (33)	DEF (29)	MB (21)	FM (28)	IPR (35)	OPR (45)	OV (11)	BC (17)	LR (8)
所提算法	**0.819**	**0.856**	**0.735**	**0.810**	**0.766**	*0.780*	*0.811*	**0.791**	0.621	**0.845**	*0.813*
HCF	*0.799*	*0.806*	*0.705*	0.762	*0.759*	**0.788**	**0.854**	*0.783*	**0.686**	*0.773*	0.809
CNN-SVM	0.787	0.768	0.704	*0.785*	0.700	0.705	0.779	0.771	0.656	0.716	0.788
SiamFC-3s	0.735	0.691	0.694	0.676	0.623	0.689	0.718	0.743	*0.674*	0.605	**0.828**
SAMF	0.705	0.654	0.661	0.660	0.596	0.620	0.690	0.707	0.645	0.572	0.659
DSST	0.638	0.653	0.569	0.525	0.511	0.508	0.638	0.614	0.478	0.623	0.539
KCF	0.633	0.619	0.570	0.590	0.526	0.588	0.634	0.634	0.506	0.566	0.527

试,成功率曲线和精确度曲线如图4所示. 与成功率位于第2的SiamFC-3s算法相比,所提算法的成功率和精确度分别提高了1.4%、6.9%;与精确度位于第2的HCF算法相比,所提算法的成功率和精确度分别提高了3.4%、0.3%.



(a) 成功率曲线



(b) 精确度曲线

图4 OTB100中所有视频的测试结果

3.3 算法速度

对于不同的视频,目标的尺度不同,从而导致ROI大小不同. 所提算法需要对ROI图像进行卷积等操作,因此目标尺度越大,算法速度越小.

在GPU加速下,所提算法在OTB100的所有视频上的平均速度为3.2 f/s,与其他4种CF类算法速度的比较如表7所示. 与HCF相比,导致所提算法速度较低的因素有2个:一是考虑了尺度变化,二是选用了层次更深的ResNet提取目标特征.

表7 5种CF类算法跟踪速度比较

算法	速度/(f·s^{-1})	算法	速度/(f·s^{-1})
所提算法	3.2	DSST	21.9
HCF	10.4	KCF	243.4
SAMF	16.9		

4 结束语

笔者提出了一种基于ResNet特征融合的尺度自适应目标跟踪算法. 针对多层ResNet特征,利用KCF算法获得目标中心位置,然后对目标进行多尺度采样,在提取fHOG特征的基础上利用CF对目标尺度进行精确估计. 在视频集OTB100中对7种相关算法进行了测试,实验结果表明,所提算法能够准确提取目标特征,同时较好地适应目标尺度变化,取得较高的跟踪成功率和精确度. 但是,由于ResNet层数较多,降低了所提算法的跟踪速度.

在实验过程中发现,利用ResNet提取图像特征时,为了保证每个视频的跟踪结果均为最优,应该为不同视频选择不同的提取特征的层. 如何做到针对每个视频进行自适应特征层选择,是下一步研究的方向.

参考文献:

[1] 卢湖川, 李佩霞, 王栋. 目标跟踪算法综述[J]. 模式识别与人工智能, 2018, 31(1): 61-76.
Lu Huchuan, Li Peixia, Wang Dong. Visual object tracking: a surey[J]. Pattern Recognition and Artificial Intelligence, 2018, 31(1): 61-76.

[2] 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.

[3] Ma C, Huang J B, Yang X K, et al. Hierarchical convolutional features for visual tracking[C]// ICCV 2015. Santiago: Institute of Electrical and Electronics Engineers Inc, 2015: 3074-3082.

[4] Danelljan M, Höger G, Khan F S, et al. Accurate scale estimation for robust visual tracking[C]// BMVC 2014. Nottingham: BMVA, 2014.

[5] Li Y, Zhu J. A scale adaptive kernel correlation filter tracker with feature integration[C]// ECCV 2014. Zurich: Springer Verlag, 2014: 254-265.

[6] Hong S, You T, Kwak S, et al. Online tracking by learning discriminative saliency map with convolutional neural network[C]// ICML 2015. Lile: IMLS, 2015: 597-606.

[7] Bertinetto L, Valmadre J, Henriques J F, et al. Fully-convolutional siamese networks for object tracking[C]// ECCV 2016. Amsterdam: Springer Verlag, 2016: 850-865.

[8] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition[C]// CVPR 2016. Las Vegas: IEEE Computer Society, 2016: 770-778.

[9] 田朗, 黄平牧, 吕铁军. SA-Siam++: 基于双分支孪生网络的目标跟踪算法[J]. 北京邮电大学学报, 2019, 42(6): 105-110.
Tian Lang, Huang Pingmu, Lü Tiejun. SA-Siam++: the two-branch siamese network-based object tracking algorithm[J]. Journal of Beijing University of Posts and Telecommunications, 2019, 42(6): 105-110.