基于难样本挖掘的孪生网络目标跟踪 *

亢　洁[1]，孙　阳[1]，沈钧戈[2†]

(1. 陕西科技大学 电气与控制工程学院, 西安 710021; 2. 西北工业大学 无人系统技术研究院, 西安 710072)

摘　要：为了解决全卷积孪生网络目标跟踪算法(SiamFC)在复杂环境下容易出现跟踪漂移甚至跟踪失败的问题，提出了一种基于难样本挖掘的孪生网络目标跟踪方法。该方法在 SiamFC 算法的基础上，首先利用特征融合模块进行特征融合，以提高特征表征的鲁棒性，然后引入一个新的损失函数，加强网络对难样本的学习能力并缓解正负样本不平衡的问题。为验证该方法的有效性，在 OTB2015 和 GOT10k 数据集上对算法进行测试实验。实验结果表明，在 OTB2015 数据集上该方法比 SiamFC 算法在成功率上提高 2.6%，精度上提高 2%在 GOT10k 数据集上该方法的 mAO 为 0.429，相比 SiamFC 算法提高了 3.7%，在光照变化、目标形变、相似背景干扰情况下具有更好的表现。

关键词：孪生网络；目标跟踪；特征融合；损失函数；难样本挖掘

中图分类号：TP391.4　　doi: 10.19734/j.issn.1001-3695.2020.03.0084

Siamese network object tracking based on hard sample mining

Kang Jie[1], Sun Yang[1], Shen Junge[2†]

(1. *College of Electrical & Control Engineering*, *Shaanxi University of Science & Technology*, *Xi'an* 710021, *China*; 2. *Unmanned System Research Institute*, *Northwestern Polytechnical University*, *Xi'an* 710072, *China*)

Abstract: In complex environment, the object tracking algorithm of fully-convolutional siamese network is prone to track drift or even track failure. In order to solve the problem, this paper proposed a siamese network tracking algorithm based on hard sample mining. On the basis of SiamFC, this method first used a feature fusion module for feature fusion to enhance the robustness of feature representation, and then proposed a novel loss function to strengthen the learning ability of network to hard samples and alleviate the problem of imbalance between positive and negative samples. To verify the validity, this method was tested on OTB2015 benchmark and GOT10k dataset. The results of OTB2015 show that this method increases the success rate by 2.6% and the accuracy by 2% compared with SiamFC. On the GOT10k dataset, the mAO of this method is 0.429, which is 3.7% higher than the SiamFC. It illustrates that this method has a better performance in the case of illumination variation, object deformation, and similar background interference.

Key words: siamese network; object tracking; feature Fusion; loss function; hard sample mining

0　引言

视觉目标跟踪是计算机视觉领域的研究热点之一[1]，广泛应用于智能视频监控、智能交通等领域。但是由于存在目标姿态变化、形状变化等内在因素以及光照变化、背景混杂、遮挡等外在因素的干扰，视觉目标跟踪仍然面临着巨大的挑战。为解决目标跟踪的难题，由于深度特征强大的表征能力，研究者开始将深度学习用于目标跟踪领域。HCF(hierarchical convolutional features for visual tracking)算法[2]、HDT(hedged deep tracking)算法[3]等用深度学习的卷积特征代替传统相关滤波跟踪算法的人工特征，大幅度提高了目标跟踪的成功率和精度。多域卷积神经网络目标跟踪算法[4](MDNet)采用离线训练和在线微调相结合的方式，充分发挥了深度学习端到端的优势，在跟踪性能上获得了显著提高。但是这些使用深度学习的目标跟踪算法的推理过程计算复杂度高，没有很好地平衡准确性和实时性。而基于孪生网络的目标跟踪算法，将目标跟踪问题看做是一个目标模板和候选区域的相似度度量问题，在目标跟踪的实时性上有着很大的优势。因此，基于孪生网络的目标跟踪算法[5~8]逐渐成为目标跟踪的主流算法。其中，全卷积孪生网络目标跟踪算法[6](SiamFC)同时考虑了速度和精度，但是由于其只使用了网络提取的最后一层特征来表征目标，当跟踪的目标外观发生变化时，跟踪鲁棒性差。此外，在训练过程中没有考虑正负样本不平衡以及大量简单负样本的问题，这些简单负样本会对损失函数起主要作用，使得网络不能很好地学习到具有判别力的信息，当出现相似背景干扰时，很容易跟踪错误。

因此，本文以 SiamFC 算法为基础，针对其目标表征能力的欠缺，提出特征融合模块，将浅层和深层的特征进行融合得到更为鲁棒的特征来表征目标。同时，为解决正负样本不平衡以及大量简单负样本的问题，提出改进的损失函数替换 Logistic 损失函数来提高网络的学习能力和判别力。最后，本文提出基于难样本挖掘的孪生网络目标跟踪方法，主要创新点为：a) 针对目标形变、光照变化导致目标外观发生变化时目标表征的鲁棒性欠缺，利用提出的特征融合模块提高特征表征的鲁棒性；b) 针对相似背景干扰问题，提出一种改进的损失函数加强对难样本的学习，提高跟踪算法的判别力。

1　本文方法

本节将详细描述提出的基于难样本挖掘的孪生网络目标跟踪方法，其整体结构如图 1 所示。其核心思想为通过权重共享的两支 Alexnet 网络提取目标模板 z 和搜索区域 x 的特征，将提取得到的三层特征通过设计的特征融合模块进行融

收稿日期：2020-03-23；修回日期：2020-05-01　基金项目：国家自然科学基金资助项目(61603233)；西安市科技计划资助项目(2019216514GXRC001CG002-GXYD1.7)

作者简介：亢洁(1973-)，女，陕西潼关人，副教授，硕导，博士，主要研究方向为模式识别、机器视觉；孙阳(1995-)，男，浙江嘉兴人，硕士研究生，主要研究方向为计算机视觉；沈钧戈(1987-)，女(通信作者)，黑龙江大庆人，助理教授，硕导，博士，主要研究方向为机器学习、计算机视觉(shenjunge@nwpu.edu.cn).

合，利用融合后更为鲁棒的特征来表征目标，然后经过一个相似度计算的卷积操作得到尺寸为 17×17×1 的目标位置响应图，并通过改进 Logistic 损失函数来加强相似背景干扰时的目标定位。

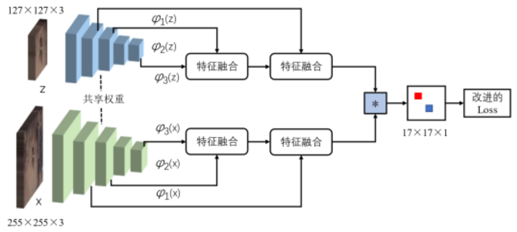


图 1　本文方法的跟踪框架

Fig. 1　Framework of the proposed method

1.1　多特征融合

由于语义特征具有丰富的语义信息，利于目标和背景的判别，因此常常采用网络提取的最后一层的特征来表征目标，然而语义特征的分辨率低，不能很好地捕捉到空间位置等细节信息，这些细节信息也是目标准确定位的关键。卷积神经网络提取的浅层特征含有丰富的细节信息，深层特征含有丰富的语义信息。因此，本文提出一个特征融合模块来有效结合来自深层和浅层的特征信息。其具体结构如图 2 所示。本文采用经过修改的 Alexnet 作为权重共享的孪生网络用于特征的提取，其网络结构如表 1 所示。

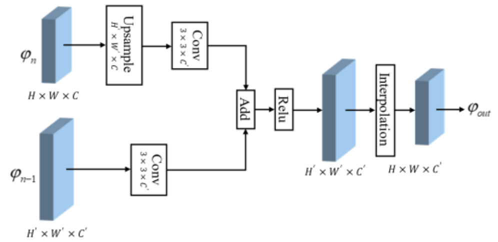


图 2　特征融合模块结构

Fig. 2　Structure of feature fusion module

表 1　孪生网络结构

Tab. 1　Structure of siamese network

网络层	尺寸	通道	步长	输出尺寸	
				模板分支	搜索分支
input	/	/	/	127×127×3	255×255×3
conv1	11×11	192	2	59×59×192	123×123×192
maxpool1	3×3	/	2	29×29×192	61×61×192
conv2	5×5	512	1	25×25×512	57×57×512
maxpool2	3×3	/	2	12×12×512	28×28×512
conv3	3×3	768	1	10×10×768	26×26×768
conv4	3×3	768	1	8×8×768	24×24×768
conv5	3×3	512	1	6×6×512	22×22×512

在网络提取的不同层的特征中，conv2 层的步长为 4，其特征分辨率较高，包含更多位置细节信息，利于目标位置的定位，conv3 层的步长为 8，特征更为稀疏，含有语义信息，conv5 经过两层卷积进一步提取更为复杂的语义信息。因此，本文选择 conv2、conv3、conv5 这三层特征输入到特征融合模块实现浅层特征和深层特征的融合，获得更为鲁棒的特征来表征目标，提高跟踪器应对目标外观变化的跟踪鲁棒性和跟踪精度。

特征融合模块的输入是 $H \times W \times C$ 的深层特征 φ_n 和 $H' \times W' \times C'$ 的浅层的特征 φ_{n-1}，融合过程描述如下：a) 首先对深层特征 φ_n 进行上采样 upsample 操作，本文使用双线性插值实现特征图的上采样，提高深层特征的分辨率。b) 然后对浅层特征 φ_{n-1} 和上采样后的深层特征 φ_n 进行一个 conv 卷积操作，将深层特征的通道数降维至与浅层特征一致，保证特征维度的一致性。c) 接着进行 add 求和和 ReLU 激活操作，将两层的特征做一个元素级融合，把深层特征的语义判别信息和浅层特征的空间位置信息结合到一起。d) 最后采取 interpolation 插值操作，保证输出特征的感受野适用于目标跟踪的精确定位问题。最终，整个特征融合的公式为

$$\varphi_{out} = \sigma(\varphi_n, \varphi_{n-1}) \tag{1}$$

其中 $\sigma(\cdot)$ 代表特征融合操作。

1.2　难样本挖掘损失函数

进一步地，针对目标跟踪中相似背景干扰问题，本文提出一个难样本挖掘损失函数。在 SiamFC 算法中使用的是 logistic 损失函数用于模型的训练：

$$l(y,v) = \log(1+\exp(-yv)) \tag{2}$$

其中 v 是目标模板与搜索区域候选框的相似度分数，$y \in \{+1,-1\}$ 是正、负样本的标签值。

由于搜索区域的目标候选框大多数属于背景即负样本，少数包含目标区域即正样本，带来了训练过程中正负样本的不平衡问题。这种正负样本的不平衡进一步造成了大量简单负样本的存在。但是在 logistic 损失函数中赋予正、负样本相同的权重，没有平衡好正样本和负样本对模型训练的影响，导致跟踪的性能受限；没有考虑训练样本中存在的大量简单负样本，这些简单的负样本会对损失函数的计算起主要作用，使得网络不能够很好地学习到具有判别力的信息，当出现相似背景干扰时，更容易出现跟踪漂移和失败。

因此，针对以上问题，基于 logistic 损失函数提出了一种改进的损失函数：

$$\begin{aligned} l(y,v) = {} & \frac{1+y}{2}\alpha(1-p)^{\gamma}\log(1+\exp(-yv)) + \\ & \frac{1-y}{2}(1-\alpha)p^{\gamma}\log(1+\exp(-yv)) \end{aligned} \tag{3}$$

即

$$l(y,v) = \begin{cases} \alpha(1-p)^{\gamma}\log(1+\exp(v)) & y=1 \\ (1-p)^{\gamma}\log(1+\exp(v)) & y=-1 \end{cases} \tag{4}$$

其中 p 是目标模板与搜索区域候选块的相似度概率值，由 v 通过 sigmoid 函数计算得到；γ 是关注困难样本的放缩因子；α 为平衡正负样本的权重。进一步地将改进的损失函数表示为

$$\alpha_T = \begin{cases} \alpha, y=1 \\ 1-\alpha, y=-1 \end{cases}; p_T = \begin{cases} p, y=1 \\ 1-p, y=-1 \end{cases} \tag{5}$$

$$l(y,v) = \alpha_T(1-p_T)^{\gamma}\log(1+\exp(-yv)) \tag{6}$$

在该改进的损失函数中，引入了 α_T 权重项，设置正样本的权重大于负样本的权重，拉近正负样本对损失值计算的贡献；此外，还引入了 $(1-p_T)^{\gamma}$ 这一动态项，γ 设置为大于 0 的值，对于简单负样本，p_T 更易趋于 1，$(1-p_T)^{\gamma}$ 趋于 0，需要叠加更多的简单负样本的计算值才能对损失函数起作用。对于困难负样本，p_T 更易趋于 0，$(1-p_T)^{\gamma}$ 趋于 1，这样对于困难负样本的权重相对加大了，更加注重困难样本。两者同时作用时，该新的损失函数能够更好地指导模型的学习，使得跟踪模型的性能提高，在目标跟踪时能够更好地应对相似背景干扰的情况。

由于目标模板的尺寸比搜索区域的尺寸小，会得到一个相似度响应图 $\boldsymbol{D}$，其是由目标模板与搜索区域中所有候选块的相似度分数 v 组成。因此，整个相似度响应图的损失函数定义为每一对目标模板和搜索区域候选块损失的平均值：

$$L(y,v) = \frac{1}{|D|}\sum_{u \in D} l(y[u], v[u]) \tag{7}$$

其中 $y[u] \in \{+1,-1\}$ 是每一个位置 $u \in \boldsymbol{D}$ 的标签值，正样本为 $y[u]=1$，负样本为 $y[u]=-1$。

1.3　基于难样本挖掘的孪生网络目标跟踪

SiamFC 算法将视觉目标跟踪看做是一种模板匹配的问题，即通过在每一帧中搜索与目标模板相似的区域来对目标

进行定位。通过学习相似度度量函数 $f(\boldsymbol{z},\boldsymbol{x})$，根据学习的结果计算目标模板特征和搜索区域候选块特征之间的相似度分数：

$$f(z,x)=g(\varphi(z),\varphi(x)) \tag{8}$$

其中：$g(\cdot)$ 是一个相似度度量；$\varphi(\boldsymbol{z})$ 是目标模板的特征；$\varphi(\boldsymbol{x})$ 是搜索区域候选块的特征。

本文方法通过离线训练来学习用于模板匹配的相似度函数。首先从同一个视频序列中获取目标模板和搜索区域作为跟踪器的输入，然后经过相同的两个共享权重的卷积神经网络分支进行特征提取，本文采用的卷积神经网络是修改的的 AlexNet。然后通过级联的特征融合模块对选取的 conv2、conv3、conv5 这三层的特征 φ_1、φ_2、φ_3 进行融合：

$$\varphi=\sigma(\varphi_1,\sigma(\varphi_2,\varphi_3)) \tag{9}$$

接着对融合得到的目标模板分支特征 $\varphi(\boldsymbol{z})$ 和搜索区域分支特征 $\varphi(\boldsymbol{x})$ 做一个交叉相关操作，计算目标模板分支的特征 $\varphi(\boldsymbol{z})$ 和搜索区域候选块的特征 $\varphi(\boldsymbol{x})$ 之间的相似度分数：

$$f(z,x)=\varphi(z)*\varphi(x)+b \tag{10}$$

其中 $*$ 代表交叉相关操作，$\boldsymbol{b}$ 为偏置项。输出为相似度响应图，相似度得分最大的位置即为目标的位置，以此来确定目标在当前帧中的位置。相似度响应图的维度与目标模板和搜索区域大小有关。

最后，在训练的过程中使用改进的损失函数来对网络的权重进行更新，使得网络学到更有用的信息，具有更强的判别力。

跟踪时根据上一帧目标位置的中心来获得当前帧的搜索区域用于计算相似度响应得分图，得分最大的位置即为当前帧中目标的位置。

2 实验和实验结果分析

2.1 实验细节

a) 训练数据准备：为了提高通用目标跟踪器的泛化能力，避免过度拟合跟踪测试的数据集，本文方法在 ILSVRC15[9] 数据集上从头离线训练。这个数据集包含超过 4000 个序列，超过 100 万的视频帧。在同一个视频序列中随机选择 2 帧，将其作为目标模板和搜索区域的训练数据对，并对其进行进一步裁剪和填充，使得目标位于每一帧的中心：

$$(w+2p)\times(h+2p)=A^2 \tag{11}$$

其中 w 和 h 分别为目标边界框的宽、高，A 为目标模板的大小，p 为 $(w+h)/4$。在训练中采用的目标模板大小为 127×127，搜索区域大小为 255×255。

b) 训练设置：在训练中设置了 50 轮，在 2 个 GPU 上训练，Batch 设置为 8。使用 SGD 对网络参数进行更新，动量设置为 0.9，学习率以几何退火的方式从 10^{-2} 到 10^{-5} 自动调整，权重衰减设置为 5×10^{-4}。

c) 测试设置：初始目标的外观只计算一次。使用双线性插值将 17×17 的相似度得分图转换成 255×255 获得更精确的位置。使用 3 个尺度 $1.0375^{\{-1,0,1\}}$ 对目标进行搜索，尺度惩罚因子设置为 0.9745，衰减因子为 0.35。

d) 实验环境和设备：本文方法使用 python 在 Pytorch 中实现，在显存为 11GB 的 NVIDIA GeForce RTX2080Ti GPU，CPU 为 3.3GHz 的 Intel Core i9-7900X，内存为 64GB 的设备上进行的。

2.2 评价标准

在 OTB2015 数据集[10]中选用 OPE(one pass evaluation)的测试方法，计算两个评价指标：精确度图(precision plot)和成功率图(success plot)。精确度表示跟踪器预测位置的中心 C_t 与真实位置的中心 C_{gt} 距离小于 20 像素的帧数占总帧数的百分比：

$$P(\sigma_{SUCC})=\|C_{gt}-C_t\|\le\sigma_{SUCC} \tag{12}$$

其中 σ_{succ} 是中心位置误差的阈值，单位为 pixel。

成功率为每个成功率图曲线下的面积(AUC)，即不同重叠阈值对应成功率的平均值。其中每个重叠阈值的成功率计算公式为

$$S(\sigma_{over})=\frac{R_{gt}\cap R_t}{R_{gt}\cup R_t}\ge\sigma_{over} \tag{13}$$

其中 R_{gt} 是真实位置的目标框，R_t 是预测的目标框，σ_{over} 是目标重叠率的一个阈值，范围为 0~1。

在 GOT10k[11]数据集中，选用 AO(average overlap)作为指标来计算预测目标框和实际目标框之间的平均重叠度。考虑到在评估过程中存在的类别不平衡问题，进一步使用了计算 mAO 这种类别平衡的度量方式来评估：

$$\text{mAO}=\frac{1}{C}\sum_{c=1}^{C}\left(\frac{1}{|S_c|}\sum_{i\in S_c}\text{AO}_i\right) \tag{14}$$

其中 C 表示类别的数量，S_c 表示属于第 C 类的一个子集，$|S_c|$ 表示子集的数量。

2.3 参数确定

a) 参数 γ 作用于 $(1-p_T)^\gamma$ 这一权重项，来关注困难样本。当 γ =0 时，权重项不起作用，相当于原来的 logistic 损失函数；当 γ 取值大于 0，权重项发挥作用。不同 γ 取值下，本文方法的测试实验结果如表 2 所示。当 γ =2 时，本文方法的实验结果最好。

表 2 不同 γ 取值下本文方法的 AUC

Tab. 2 AUC of the proposed method under different γ

γ	0	1	2	3	4
AUC	0.535	0.539	0.558	0.468	0.473

b) 参数 α 是调节正负样本不平衡的重要参数，通过权重因子 α 的调节，拉近正、负样本的损失值。不同 α 取值下，本文方法的测试实验结果如表 3 所示。当 α =0.6 时，本文方法的实验结果最好。

表 3 不同 α 取值下本文方法的 AUC

Tab. 3 AUC of the proposed method under different α

α	0.5	0.6	0.7	0.8	0.9
AUC	0.556	0.589	0.532	0.496	0.512

2.4 实验结果分析

OTB2015 数据集一共含有 100 个跟踪序列，包含了 11 种具有挑战的序列，在这个数据集上将本文方法和 SiamFC[6] 算法进行了比较，并且还和 GOTURN[12]、CFNet[13]、SINT[5]、MEEM[14]、Staple[15]、DSiam[16]、DSST[17]、KCF[18]、SAMF[19]、StructSiam[20]、SiamTri[21]这些主流的目标跟踪算法也作了比较。图 3、4 分别为这 13 种目标跟踪算法在 OTB2015 数据集上的成功率图和精确度图。

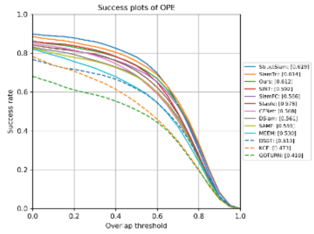


图 3 OTB2015 数据集上算法的成功率

Fig. 3 Success plot of algorithm on OTB2015 benchmark

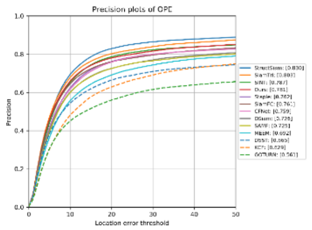


图 4 OTB2015 数据集上算法的精确度

Fig. 4 Precision plot of algorithm on OTB2015 benchmark

从图 3、4 的测试结果可以看到，本文方法相比于 SiamFC 算法，在成功率上提高了 2.6%，在精确度上提高了 2%，表明了本文方法引入特征融合和新的损失函数的有效性。与同样通过改进损失函数的跟踪算法相比，由于 SiamTri 算法通过引入 Triplet 损失，更为充分地利用正样本和负样本之间的

联系，但是 SiamTri 算法对目标外观变化效果不好，因此在精度上本文方法不如 SiamTri 算法，但是在成功率上本文方法和 SiamTri 算法接近。StructSiam 算法充分利用局部特征，但本文方法深度挖掘难样本的信息，在跟踪器性能上本文方法与 StructSiam 算法同样有竞争性。此外，相比于其他基于孪生网络的目标跟踪算法，本文方法取得了更优异的性能，特别是本文方法比 SINT 算法在成功率上高 2%，精度基本持平。但是 SINT 算法通过大量的匹配计算非常耗时，而本文方法的速度达到 71FPS，远超 SINT 算法的 4FPS，在保证跟踪准确度的同时，速度也达到了实时性的要求。本文方法采用卷积操作代替滑动窗口检测，以解决边界效应，因此相比于 KCF 算法在成功率上大幅度提高了 13.9%，在精度上提高了 15.2%。

GOT10k 数据集包括超过 10000 个视频，目标框超过 150 万个，可细分为 563 个目标类别，此数据集还有一个动作类别，分为 87 种动作。用于测试的数据集包含 180 个视频序列，包括 84 个目标类别，32 个动作类别。因此，进一步在难度更大的 GOT10k 数据集上将本文方法和 SiamFC[6]、SiamRes[22]、SiamFC2[13]、DSiam[16]、GOTURN[12]这 5 种算法作了比较，测试结果如表 4 所示。

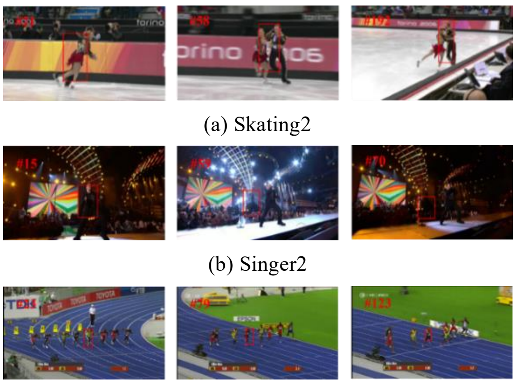
表 4　GOT10k 数据集上的测试结果

Tab. 4　Test result of the GOT10k

指标	SiamFC	SiamRes	SiamFC2	DSiam	GOTURN	本文
mAO	0.392	0.385	0.434	0.417	0.418	0.429

从表 4 的测试结果可以看到，本文方法相比于 SiamFC 算法 mAO 提高了 3.7%。SiamFC2 在 SiamFC 的基础上加入相关滤波，并且实现端到端的训练，和本文方法的性能接近。SiamRes 算法采用更深的主干网络，但没有利用浅层的特征，缺乏位置细节信息，本文方法的 mAO 比 SiamRes 算法提高了 4.4%，进一步证明了本文方法的有效性和较好的泛化能力。

以上是对不同跟踪器的一个定量评价，为了进一步定性地对本文方法进行评估，在 OTB2015 数据集中选取了 Skating2(目标形变)、Singer2(光照变化)、Bolt(背景杂乱)这 3 个挑战性的视频序列对本文方法和 SiamFC 算法进行了进一步的测试实验。图 5、6 分别为 SiamFC 算法和本文方法在这三个视频序列上的跟踪结果。



(a) Skating2

(b) Singer2

(c) Bolt

图 5　SiamFC 算法的跟踪结果

Fig. 5　Tracking result of siamfc

通过对 SiamFC 算法和本文方法在三个视频序列上的跟踪结果的分析，本文方法相比于 SiamFC 算法具有更强的抗形变能力，当目标发生较大的形变时，SiamFC 算法的目标定位有着明显的偏差，而本文方法可以较为准确地定位目标，例如图 5(a)的 192 帧和图 6(a)的 192 帧；本文方法在目标发生严重的光照变化时也表现出了一定的抵抗能力，而 SiamFC 算法当目标受到光照变化的影响时跟踪失败了，例如图 5(b)的 59 帧、70 帧和图 6(b)的 59 帧、70 帧；本文方法在相似目标干扰的情况下的表现比 SiamFC 算法更好，当出现相似目标干扰时，本文方法依然跟踪成功，而 SiamFC 算法完全失败了，例如图 5(c)的 79 帧、123 帧和图 6(c)的 79 帧、123 帧。

(a) Skating2

(b) Singer2

(c) Bolt

图 6　本文方法的跟踪结果

Fig. 6　Tracking result of the proposed method

3　结束语

本文提出了基于难样本挖掘的孪生网络目标跟踪算法，主要解决在光照变化、目标形变情况下的特征表征问题以及相似背景干扰情况下的难样本学习问题。首先，在全卷积孪生网络目标跟踪算法的基础上引入了一个特征融合模块，提高了特征表征的鲁棒性。然后，基于 logistic 损失函数提出改进的损失函数，加强了网络对难样本的学习能力。实验结果表明，在 OTB2015 数据集上相比于 SiamFC 算法，在成功率上提升了 2.6%，在精度上提升了 2%，在 GOT10k 数据集上相比于 SiamFC 算法的 mAO 提高了 3.7%，验证了本文方法改进的有效性。本文方法和其他一些主流的目标跟踪算法相比在性能上也有很大的竞争力。但是本文方法没有考虑目标遮挡问题，在今后的工作中将考虑目标遮挡问题，来进一步提高目标跟踪算法的性能。

参考文献：

[1] 孟琭，杨旭. 目标跟踪算法综述 [J]. 自动化学报, 2019, 45 (7): 1244-1260. (Meng Lu, Yang Xyu. A survey of object tracking algorithms [J]. Acta Automatica Sinica, 2019, 45 (7): 1244-1260.)

[2] Ma Chao, Huang Jiabin, Yang Xiaokang, *et al.* Hierarchical convolutional features for visual tracking [C]// Proceedings of the IEEE International Conference on Computer Vision, 2015: 3074-3082.

[3] Qi Yuankai, Zhang Shengping, Qin Lei, *et al.* Hedged deep tracking [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 4303-4311.

[4] Nam H, Han B. Learning multi-domain convolutional neural networks for visual tracking [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 4293-4302.

[5] Tao Ran, Gavves E, Smeulders A W M. Siamese instance search for tracking [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 1420-1429.

[6] Bertinetto L, Valmadre J, Henriques J F, *et al.* Fully-convolutional siamese networks for object tracking [C]// European Conference on Computer Vision. Springer, Cham, 2016: 850-865.

[7] 仇祝令，查宇飞，朱鹏，等. 基于孪生神经网络在线判别特征的视觉跟踪算法 [J]. 光学学报, 2019, 39 (9): 0915003. (Qiu Zhuling, Zha Yufei, Zhu Peng, *et al.* Visual tracking algorithm based on online feature discrimination with siamese network [J]. Acta Optica Sinica, 2019, 39 (9): 0915003.)

[8] 任珈民，宫宁生，韩镇阳. 一种改进的基于孪生卷积神经网络的目

标跟踪算法 [J]. 小型微型计算机系统, 2019, 40 (12): 2686-2690. (Ren Jiamin, Gong Ningsheng, Han Zhenyang. Improved target tracking algorithm based on siamese convolution neural network [J]. Journal of Chinese Computer Systems, 2019, 40 (12): 2686-2690.)

[9] Russakovsky O, Deng Jia, Su Hao, *et al.* Imagenet large scale visual recognition challenge [J]. International Journal of Computer Vision, 2015, 115 (3): 211-252.

[10] Wu Yi, Lim J, Yang M H. Object tracking benchmark [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2015, 37 (9): 1834-1848.

[11] Huang Lianghua, Zhao Xin, Huang Kaiqi. Got-10k: A large high-diversity benchmark for generic object tracking in the wild [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019: 1-1.

[12] Held D, Thrun S, Savarese S. Learning to track at 100 fps with deep regression networks [C]// European Conference on Computer Vision. Springer, Cham, 2016: 749-765.

[13] Valmadre J, Bertinetto L, Henriques J F, *et al.* End-to-end representation learning for correlation filter based tracking [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 2805-2813.

[14] Zhang Jianming, Ma Shugao, Sclaroff S. MEEM: Robust tracking via multiple experts using entropy minimization [C]// European Conference on Computer Vision. Springer, Cham, 2014: 188-203.

[15] Bertinetto L, Valmadre J, Golodetz S, *et al.* Staple: Complementary learners for real-time tracking [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 1401-1409.

[16] Guo Qing, Feng Wei, Zhou Ce, *et al.* Learning dynamic siamese network for visual object tracking [C]// Proceedings of the IEEE International Conference on Computer Vision, 2017: 1763-1771.

[17] Danelljan M, Häger G, Khan F S, *et al.* Accurate scale estimation for robust visual tracking [C]// British Machine Vision Conference, Nottingham, September 1-5, 2014. BMVA Press, 2014.

[18] Henriques J F, Caseiro R, Martins P, *et al.* High-speed tracking with kernelized correlation filters [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2014, 37 (3): 583-596.

[19] Li Yang, Zhu Jianke. A scale adaptive kernel correlation filter tracker with feature integration [C]// European Conference on Computer Vision. Springer, Cham, 2014: 254-265.

[20] Zhang Yunhua, Wang Lijun, Qi Jinqing, *et al.* Structured siamese network for real-time visual tracking [C]// Proceedings of the European conference on computer vision, 2018: 351-366.

[21] Dong Xingping, Shen Jianbing. Triplet loss in siamese network for object tracking [C]// Proceedings of the European Conference on Computer Vision, 2018: 459-474.

[22] Zhang Zhipeng, Peng Houwen. Deeper and wider siamese networks for real-time visual tracking [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019: 4591-4600.