计算机科学
COMPUTER SCIENCE

基于SiameseFC的双模板异步更新追踪方法

马汉达, 殷达

引用本文

马汉达, 殷达. 基于SiameseFC的双模板异步更新追踪方法[J]. 计算机科学, 2022, 49(11A): 211200133-7.

MA Han-da, YIN Da. Dual Template and Asynchronous Update Tracking Method Based on SiameseFC [J]. Computer Science, 2022, 49(11A): 211200133-7.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.211200133

基于 SiameseFC 的双模板异步更新追踪方法

马汉达　殷　达
江苏大学计算机科学与通信工程学院　江苏 镇江 212013

摘　要　全卷积孪生神经网络 SiameseFC 有着追踪速度快、精度高等优势,但在较为复杂的场景下仍然存在一定的缺陷,并且模板不更新的追踪模式也会在快速变化下的场景中出现较大的误差。因此,提出了一种基于全卷积孪生神经网络的双模板异步更新的追踪算法。首先基于 VGG-16 网络提取深层与浅层两种特征,分别使用两套对应的模板,两套模板独立且异步地更新,从而节约计算资源。然后对于模板的更新,同时考虑初始模板、前一次追踪所用模板,以及前一帧追踪结果提取的模板,并且使用了基于 APCE 的判断机制,更新时动态地分配三者的比例。所提算法在 OTB100 的基准测试结果上优于 SiamRPN 和 SiamDW 等主流算法,成功率与精确度均提升了约 4%～5%,并且速度达到了 44 fps 左右,可以满足实时追踪的要求。

关键词:SiameseFC;VGG-16 网络;模板更新;双模板;APCE
中图法分类号　TP391.4

Dual Template and Asynchronous Update Tracking Method Based on SiameseFC

MA Han-da and YIN Da
School of Computer Science and Communication Engineering,Jiangsu University,Zhenjiang,Jiangsu 212013,China

Abstract　SiameseFC has the advantages of fast tracking speed and high accuracy,but it still has some defects in complex scenes,and the tracking mode without updating the template will also cause large errors in the scene that changes rapidly. Therefore,this paper proposes a new tracking method,the dual-template asynchronous update based on SiameseFC. Firstly,both the deep and shallow features are extracted from the VGG-16 network,and two sets of corresponding templates are used respectively,the two sets of templates are updated independently and asynchronously to save computing resources. Then,for the update of the template,the initial template,the template used in the previous tracking,and the template extracted from the tracking result of the previous frame are considered at the same time. And it uses an APCE-based judgment mechanism to dynamically allocate the proportions of the three templatets when updating. This algorithm is superior to mainstream algorithms such as SiamRPN in the benchmark results of OTB100,the success rate and accuracy improve by about 4%～5%,and reaches about 44 fps,which is sufficient to meet real-time tracking requirements.

Keywords　SiameseFC,VGG-16,Template update,Dual-template,APCE

目标跟踪问题是计算机视觉领域中的一个重要方向。目标跟踪指对图像序列中的运动目标进行检测、提取、识别和跟踪[1]。目前,目标跟踪的难点主要在以下几个方面:目标、背景复杂多变,多目标重合,视频模糊以及目标被遮挡[2]。当下的目标跟踪算法研究主要就是为了解决这些问题。

在深度学习被广泛应用前,采用相关滤波(Correlation Filter)的目标跟踪算法有着较好的跟踪能力。算法使用图像灰度特征,训练滤波器对图像进行滤波处理得到响应图,速度可以达到 600 fps[3]。近年来,深度学习的发展使目标跟踪的研究取得了更大的进步,因此诞生了结合神经网络和相关滤波的跟踪算法,如 HCFT[4] 和 HDT[5] 等,利用 CNN(Convolutional Neural Network)网络强大的特征提取能力取得了优秀的表现。

随着深度学习的继续发展,Bertinetto 等[6] 提出了全卷积孪生神经网络(Fully-Convolutional Siamese network,SiameseFC),使用孪生网络为框架实现了高精度和高效率的目标追踪。为了更大程度地利用神经网络提取特征,Li 等[7] 提出了 SiamVGG,将 VGG-16 网络经过一定修改后以代替原有的网络,实现了更好的追踪效果。在此基础上,文献[8]提出了 SiamRPN,使用候选区生成网络,进一步提升了跟踪器的追踪精度。SiamRPN++[9] 则在 SiamRPN 的基础上解决了平移不变性的问题,进一步提升了性能。SiamDW[10] 提出了一种 CIR 残差网络,解决了 padding 带来的问题,同样提升了跟踪性能。

尽管这些算法都能实现较好的追踪效果,但仍然存在以下问题。首先,这些跟踪器采用的骨干网络 AlexNet 特征提取能力不足,或没有完全利用网络的深度,追踪性能仍有提升空间。其次,上述算法都没有模板更新功能,模板固定为第一帧,虽然可以保证算法的优异性能,但遇到目标遮挡、画面模糊时鲁棒性较差。

为了解决上述问题,文献[11-14]提出了各自的解决方案。文献[11]提出了基于初始和动态双模板的跟踪算法,以减少跟踪中目标的飘移。文献[12]在 SiamFC 框架上加入了生成对抗网络模块,解决低分辨率图像问题。文献[13]在 SiamFC 算法中引入了遮挡机制,对遮挡程度不同的目标实行不同的追踪策略。文献[14]提出高置信的双模板嵌套更新的

通信作者:马汉达(mahd@ujs.edu.cn)

机制。文献[15]提出了 AD-LSTM 这一改进的 Siamese 网络，在 LSTM 网络中增加了一个 cell 来缓解目标特征衰减。

为了更好地解决以上问题，本文提出了基于 SiamFC 的双模板异步更新的追踪算法。首先，基于 VGG-16 网络同时提取深层和浅层两种特征，最终得到深浅两种响应图并按比例结合，从而更好地对网络深度进行利用。其次，本文算法加入了模板更新机制，模板有初始、前一帧、当前 3 种分支，并使平均峰值相关能量 APCE[16] 判断更新与否和 3 个分支的占比，以动态更新的方式提高模板更新的鲁棒性，降低目标被遮挡、模糊情况下的跟丢风险。最后，将深层模板更新频率定为 3 帧。考虑到浅层提取的大多为目标的轮廓、位置等信息，相较于深层的特征随时间变化幅度较小，因此浅层模板更新定为 5 帧，以降低计算量，提升性能。

1 Siamese 网络简介

本文针对孪生神经网络 SiamFC 算法的特征挖掘深度不足、无法获得更多语义特征的问题，为该网络替换了特征提取能力较强的 VGG-16 网络，充分利用该网络的在网络深度上的优势，实现双层特征提取，结合模板的合理动态更新带来算法性能上的提升。

1.1 SiameseFC 跟踪算法

SiameseFC，简称 SiamFC，其网络结构如图 1 所示。该网络由模板区和搜索区两部分组成。模板区用于对要跟踪的目标提取信息，然后搜索区与其进行对比来判断目标的位置，SiamFC 使用相同的骨干网络对这两部分分别提取特征，然后对其进行卷积操作得到响应图。在一张图片中预设好要追踪的目标的区域，以目标为中心将这一区域进行裁剪得到 127×127 分辨率的图像(如果目标在原图片的边缘，则对空白部分填充像素平均值)，作为模板 z 输入到主干网络中，同时在下一张图片中选出分辨率为 255×255 搜索区 x，用互相关函数 $f(x,z)$ 得出两者的相似度，该函数的定义为：

$$f(x,z)=\phi(z)*\phi(z)+b \tag{1}$$

其中，$f(x,z)$ 为得出的响应图，$\phi(\cdot)$ 为提取特征使用的骨干网络，$*$ 为互相关操作，b 为偏置项。

式(1)得出的响应图为 17×17，经过双三次线性插值放大到与搜索区近似的大小，其中响应值最高的点即为目标中心。

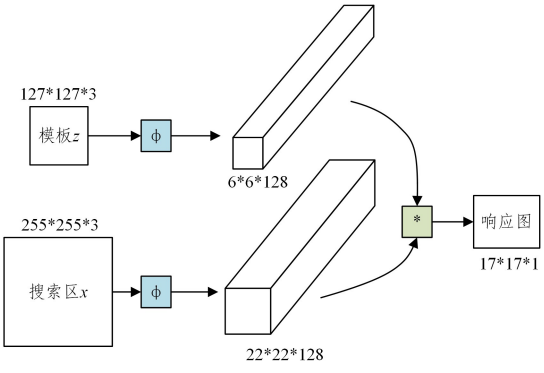


图 1 SiamFC 网络框架

Fig. 1 Framework of SiamFC network

在训练过程，由于该网络是一类典型的逻辑回归问题，使用 logsitic 损失函数，定义为：

$$l(y[u],v[u])=\log(1+e^{-y[u]v[u]}) \tag{2}$$

其中，$v[u]$ 为位置 u 的预测值，而 $y[u]$ 为位置 u 的真实值，且 $y\in\{-1,1\}$，即仅有正负两种情况。

而对于得到的整个响应图，损失函数可以是每一个点的损失的均值，定义为：

$$L(v,y)=\frac{1}{|D|}\sum_{u\in D}l(y[u],v[u]) \tag{3}$$

其中，D 为相应的点的数量，$\sum_{u\in D}l(y[u],v[u])$ 为每一个点的损失求和。

1.2 VGG16 网络

SiameseFC 网络使用了 AlexNet 作为特征提取的骨干网络，该网络诞生时间较早，特征提取能力已经逊于许多后续的网络结构，无法更高效地胜任目标追踪这类对神经网络深度有要求的工作。因此，本文将 SiamFC 中的 AlexNet 替换为 VGG-16 骨干网络。

VGG 网络实际上有 VGG-11，VGG-13，VGG-16，VGG-19 等细分种类，它们的区别是网络深度，即卷积层数不同。其中常用的为 VGG-16 和 VGG-19 网络。后者相比前者在两个池化层之间的卷积层由 3 层增加到 4 层。但 VGG-19 参数太大，训练时间过长，因此使用 VGG-16 更合适。VGG 网络采用 3×3 的小卷积核，以减少参数，从而让网络可以卷积到更大的深度。因此，VGG-16 网络堆叠了 13 个卷积层，以及 3 个池化层，组成了 16 层的网络，同时每层均使用 ReLU 作为激活函数。

为了使 VGG-16 网络适合作为 Siamese 网络的骨干网络，需要进行一定的修改。首先，由于每次卷积需要增加一个 padding 像素，会对后续的特征提取带来一定干扰，因此每次卷积后都会增加一个 corp 层，剪切掉这个 padding。然后，为减小计算压力以适应视频追踪的实时性要求，去掉了两个卷积核为 3×3 的卷积层，以及最后的一个池化层，因此总层数减少到 13 层，所减去的层数的卷积大小并无增加。修改后的网络结构如表 1 所列。

表 1 网络结构

Table 1 Network structure

网络层数	卷积大小	模板尺寸	搜索区尺寸
Conv1	3×3×64	127×127	255×255
Crop		125×125	253×253
Conv1-2	3×64×64	125×125	253×253
Crop		123×123	251×251
Pool1	2×2	61×61	125×125
Conv2	3×64×128	61×61	125×125
Crop		59×59	123×123
Conv2-2	3×128×128	59×59	123×123
Crop2		57×57	121×121
Pool	2×2	28×28	60×60
Conv3	3×128×256	28×28	60×60
Crop		26×26	58×58
Conv3-2	3×256×256	26×26	58×58
Crop		24×24	56×56
Conv3-3	3×256×256	24×24	56×56
Crop		22×22	54×54
Pool3	2×2	22×22	54×54
Conv4	3×256×512	11×11	27×27
Crop		11×11	27×27
Conv4-2	3×512×512	9×9	25×25
Crop		7×7	23×23
Conv4-3	3×512×512	7×7	23×23
Crop		5×5	21×21
Conv5	1×512×256	5×5	21×21

2 基于 SiameseFC 的双模板异步更新追踪方法介绍

由于 VGG-16 网络层数较多，可以把一个较浅层的特征也提取出来加以利用，形成深浅双模板追踪的方法，从而更充分地利用骨干网络所提取的信息。为了解决追踪的目标在一段时间后变化较大、模板不再有效的问题，基于 APCE 进行模板更新以保证长期追踪的稳定性。

2.1 基于 APCE 的模板更新

原版的 SiamFC 算法没有任何模板更新。一般而言，追踪的目标在视频中会随着时间产生或大或小的变化，但 SiamFC 跟踪器为了使时间复杂度不会过高，不影响跟踪的时效性，从第一帧得到的模板会在跟踪过程中一直使用下去，并不会更新。这导致在目标的外观形态发生明显变化后，跟踪的成功率会大幅下降，最终导致跟丢目标。视频跟踪领域遇到的另外两个问题：目标被遮挡和视频图像模糊同样会导致跟踪结果会因模板不再适用而产生漂移，无法跟住目标。因此，许多改进算法都提出了模板更新机制，例如 Hou 等[11]、Ren[14]等都提出了以 APCE 判断模板更新的算法，但这些文献仅把 APCE 作为判断是否更新的依据。然而 APCE 本身可以作为一个参数决定模板如何更新。因此，本文在前人工作的基础上提出了基于 APCE 的模板更新方法，根据 APCE 值的相对大小动态地改变模板更新。

由于每一帧追踪结束都更新模板会消耗巨大的计算资源，从而影响追踪时效性，同时视频中经常出现目标快速移动，或全部被遮挡，以及拍摄模糊等问题，很多视频中的帧完全无法作为模板使用。为了减少遇到无效模板的情况，同时保证及时更新，模板更新定为每 3 帧判断一次。如果判断失败则仍然不更新。更新的步骤如下：

(1)首先对于上一帧追踪时，经过互相关，在进行双三次线性插值，得到尺寸为 272×272 最终响应图，在其中取出最大的响应值$F_{\max}$，过程表示为：

$$F_{\max}=\max(response) \tag{4}$$

以$F_{\max}$和整个响应图的数据计算 $APCE$，如式(5)所示：

$$APCE=\frac{|F_{\max}-F_{\min}|^2}{mean(\sum_{w,h}(F_{w,h}-F_{\min})^2)} \tag{5}$$

其中，$F_{w,h}$代表 w 行 h 列的响应，$F_{\max}$，$F_{\min}$分别为最大和最小响应，$mean()$表示取平均。$APCE$ 为平均峰值相关能量，目的是反应$F_{\max}$这个值的变化的剧烈程度，从而可以反应一个追踪结果是否可信。在理想的追踪中，如果每一帧追踪都计算 $APCE$，这个值会基本维持不变或缓慢增大。而如果 $APCE$ 迅速减小，则极有可能代表目标被遮挡消失。

因此，可以基于 $APCE$ 判断是否更新模板，只有$F_{\max}$和 $APCE$ 都大于历史平均值一定比例时，即认为可以更新模板，公式如下：

$$\frac{|apce-mean(apce)|}{mean(apce)}>\lambda_1,\frac{|F_{\max}-mean(F_{\max})|}{mean(F_{\max})}>\lambda_2 \tag{6}$$

其中，λ_1和λ_2为大于历史平均值的比例。

如果 $APCE$ 持续减小到一定程度时，很有可能已经追踪失败，完全脱离目标。因此，如果 $APCE$ 小于一个阈值则直接将模板恢复为初始模板，公式为：

$$APCE>\lambda_3 \tag{7}$$

其中，λ_3为判断是否直接恢复初始模板的阈值。

(2)判断可以更新后，执行模板更新。更新模板方式为：同时根据原始的目标位置数据、上一帧追踪所确定的目标位置数据，在图片上裁切出模板区，两者同时使用 VGG-16 网络提取特征，在结合上一帧追踪中所使用的模板提取的特征，将三者融合。新目标跟踪模板同时结合了初始目标跟踪模板、旧模板和当前帧提取的模板，防止更新到错误模板，有较强的抗干扰能力。该过程的表达式为：

$$\begin{aligned}&T=w_1\times T_0+w_2\times T_{\text{last}}+w_3\times T_i\\&w_1+w_2+w_3=1\end{aligned} \tag{8}$$

其中w_1，w_2，w_3分别为初始模板、上一帧追踪使用模板和新提取模板的结合比例。

$APCE$ 对追踪结果的好坏有判断作用，$APCE$ 如果升高则表明追踪结果越好，而越好的结果也更适合将其作为新的模板用于后续的追踪判断。因此，本文算法将以 $APCE$ 与 $APCE$ 历史平均值的关系决定w_3的大小，公式为：

$$w_3=0.1*[1-(\frac{|apce-mean(apce)|}{mean(apce)}+1)^{-1}]+w \tag{9}$$

式(9)可以让w_3在 $APCE$ 快速变化时降低取值，变化较小时升高，幅度为正负 0.1。其中 w 和w_1一样为固定值，式(8)中的w_2则为 1 减去w_1和w_3后的剩余值。

2.2 基于 APCE 的双模板异步更新追踪算法步骤

在 VGG 网络中，层数越深，所取得的实验效果越好。conv4 层的提取能力强于 conv3 层，且和 conv5 层搭配使用时效果更佳。经过实验也确定 conv4 作为浅层要好于 conv3-3 层，因此，双模板的浅层模板为 conv4，深层模板为 conv5。具体的步骤如下：

(1)在跟踪的第一张图中，根据预设的目标坐标，以目标为中心裁剪出尺寸为 127×127 的模板图，并将尺寸缩放到 127×127，如果有超出图片边界的部分，用范围内其他部分的平均颜色值填充。将模板图输入 VGG-16 网络，在 conv5 层提取深层模板，在 conv4 层提取浅层模板。

(2)从跟踪的第 $x\in\{2,3,4,\cdots\}$张图开始，根据上一帧追踪得出的目标位置(或第一帧根据预设的目标位置)，以目标位置为中心裁剪并缩放到尺寸为 255×255 的搜索区。将搜索区输入 VGG-16 网络，同样在 conv5 层提取深层搜索区特征，在 conv4 层提取浅层搜索区特征，两者的尺寸分别为 21×21 和 5×5。

(3)对深层的模板和搜索区特征、浅层的模板和搜索区特征分别进行互相关操作，得出尺寸均为 17×17 两个响应图。再分别进行双三次线性插值，将尺寸还原为 272×272 的响应图，记为 $f(z_5,x)$和 $f(z_{3\text{-}3},x)$。将两个响应直接按比例结合，得到最终响应图。过程表示为：

$$f(z_1,z_2,x)=\omega_1\times f(z_5,x)+\omega_2\times f(z_{3\text{-}3},x) \tag{10}$$

其中，ω_1和ω_2为结合比例。

(4)以 2.1 节中的基于 APCE 的模板更新方法判断是否进行模板更新。其中，深浅两层模板独立进行更新判断和计算。由于浅层能代表的语义信息较少，多为模糊的物体轮廓，受干扰较小，无须频繁更新，因此将判断更新的频率定为每 5 帧一次，从而节约更多的计算资源。浅层模板与深层模板一起组成了异步的更新模式。具体过程如图 2 所示。

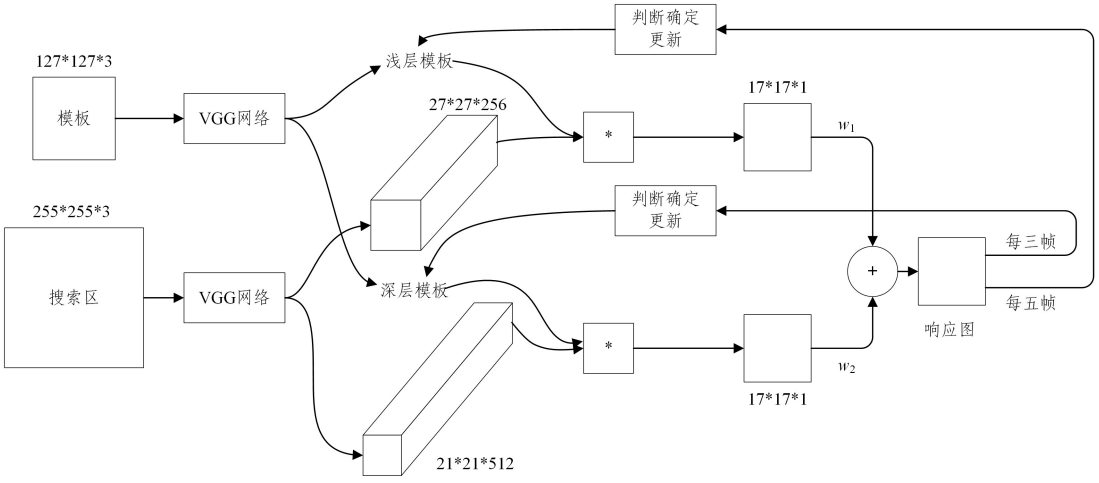


图 2 本文算法框架

Fig. 2 Framework of the proposed algorithm

3 实验与分析

为验证本文算法，进行了相关实验，实验环境为：Windows10 专业版操作系统，CPU 为 AMD Ryzen 7 3700X 8-Core Processor，GPU 为 NVIDIA GeForce GTX 1080Ti。在 python 编程环境下使用 pytorch 作为机器学习框架。本文使用 GOT-10K 数据集进行模型的预训练。本文使用目前流行且有代表性的 OTB2015(即 OTB100)和 VOT2018 数据集验证算法性能。OTB100 数据集包括 74 个彩色视频和 26 个黑白视频，涵盖了模糊、快速移动、遮挡等多种情况。VOT2018 数据集则为 60 个视频，全都为彩色。本文算法将与 SiamRPN[8]，SiamDW[10]，SRDCF[17]，CFNet[18]等主流的跟踪算法进行性能对比，同时也与同样使用双模板更新机制的文献[11]、文献[14]的算法进行对比。实验表明本文算法在多项指标上有一定的优势。

3.1 参数确定

本文所用的 SiamVGG 网络使用离线训练，采用随机梯度下降法优化网络参数。训练时，batch size 为 8，训练 epoch 数为 50，学习率随着 epoch 数的增加从 0.01 逐渐下降到 0.00001。结束后选取测试结果最好的一个 epoch 产生的模型，并继续完善算法。

本文描述的双模板异步更新的流程中，式(6)、式(8)、式(9)和式(10)中的数个参数未确定。为了确定这些参数，分别对每个公式在同一段视频上进行实验，在参数取值范围内进行足够多次数的跟踪，选取结果中成功率最高的参数取值。其中，式(10)中的ω_1和ω_2代表深层和浅层模板的融合比例。经过实验确定两者取值分别为 0.7 和 0.3，数据如表 2 所列。可以看到，ω_1取 0.7 时算法成功率最高，大于或小于 0.7 成功率均有下降。

表 2 不同参数取值下的成功率

Table 2 Success rate of different parameter values

成功率	ω_2	ω_1
0.450	0.1	0.9
0.587	0.2	0.8
0.714	**0.3**	**0.7**
0.639	0.4	0.6
0.685	0.5	0.5
0.538	0.6	0.4
0.612	0.7	0.3
0.389	0.8	0.2
0.493	0.9	0.1

式(6)中的λ_1和λ_2为判断是否更新模板的阈值。这两个值不宜太高也不宜太低，否则会难以更新或失去对模板更新的限制意义。经过大量实验，数据确定为 0.15 和 0.3，数据如表 3 所列。关于模板更新的加权系数，以 0.01 为精度改变取值再次进行实验，经过实验得出，w_1在 0.2～0.25 之间、w在 0.54～0.55 之间时，成功率都明显高于其他取值，再把缩小的取值范围对整个数据集的视频进行实验，选取平均成功率最高的取值，具体数据如表 4 所列。最终确定式(8)中的w_1为 0.23，式(9)中的 w 为 0.55。

表 3 确定λ_1和λ_2实验数据

Table 3 Experimental data to determine λ_1 *and* λ_2

$\lambda_1 \backslash \lambda_2$	0	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.6	0.65	0.70
0	0.558	0.653	0.367	0.592	0.586	0.639	0.532	0.572	0.469	0.436	0.505	0.519	0.562	0.558	0.549
0.05	0.432	0.569	0.601	0.598	0.636	0.64	0.532	0.504	0.469	0.437	0.505	0.519	0.562	0.558	0.549
0.10	0.393	0.408	0.393	0.658	0.63	0.58	0.629	0.521	0.439	0.437	0.519	0.519	0.562	0.558	0.549
0.15	0.483	0.627	0.618	0.701	0.523	0.447	0.714	0.521	0.591	0.481	0.519	0.519	0.562	0.558	0.549
0.20	0.583	0.629	0.665	0.592	0.579	0.485	0.306	0.306	0.558	0.481	0.519	0.519	0.562	0.558	0.549
0.25	0.644	0.641	0.641	0.641	0.535	0.535	0.483	0.498	0.543	0.513	0.519	0.519	0.562	0.558	0.549
0.30	0.515	0.515	0.515	0.515	0.515	0.515	0.515	0.59	0.511	0.511	0.51	0.509	0.561	0.561	0.551
0.35	0.505	0.505	0.505	0.505	0.505	0.505	0.505	0.505	0.505	0.515	0.515	0.516	0.516	0.52	0.548
0.40	0.515	0.515	0.515	0.515	0.515	0.515	0.515	0.515	0.515	0.505	0.505	0.56	0.548	0.548	0.548
0.45	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548
0.50	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548	0.548

表 4　在结果较好区域的w_1和 w 取值与成功率

Table 4　Values and sucess rates of w_1 and w in the region with better results

w \ w_1	0.53	0.54	0.55	0.56	0.57
0.20	0.669	0.713	0.687	0.624	0.435
0.21	0.435	0.709	0.649	0.687	0.412
0.22	0.419	0.536	0.659	0.676	0.682
0.23	0.672	0.705	0.71	0.551	0.597
0.24	0.644	0.703	0.67	0.635	0.682
0.25	0.644	0.702	0.65	0.641	0.611

3.2　定量分析

在定量分析中，评估算法性能的主要指标为成功率、精确度和 fps 三者。除此之外，在 VOT 数据集中还有 EAO 作为评价指标，EAO 可以同时反映跟踪的精确度和鲁棒性。

成功率反应算法跟踪得出的目标框与实际目标框的误差程度，误差小于规定值则判断为跟踪成功，成功跟踪的占比就是成功率。其中，第 i 帧的算法输出目标框B^i_{box}、与真实目标框B^i_{gt}的重叠度O_i表示为：

$$O_i=\frac{B^i_{\text{box}}\bigcap B^i_{\text{gt}}}{B^i_{\text{box}}\bigcup B^i_{\text{gt}}} \tag{11}$$

当第 i 帧的重叠度O_i大于阈值 $u\in(0,1)$判定为追踪成功，成功率 S 即为追踪成功的帧数占整个视频总整数的比例，可以表示为：

$$S=\frac{\sum_{i=1}^{n}S_i}{n}\times 100\% \tag{12}$$

其中，n 为视频总帧数，S_i表示第 i 帧结果是否为成功。成功时取值为 1，否则为 0。

精确度同理，反应算法跟踪得出的中心点与目标实际中心点差距小于规定值的比例。其中，算法得出的目标中心点(x_b,y_b)与目标真实中心点$(x_{\text{gt}},y_{\text{gt}})$的差距 l 可以表示为：

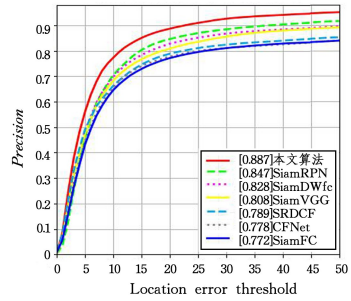
$$l=\sqrt{(x_b-x_{\text{gt}})^2+(y_b-y_{\text{gt}})^2} \tag{13}$$

当第 i 帧追踪结果算出的l_i大于阈值 v 时，P_i为 1，否则为 0。最终精确度 P 可以表示为：

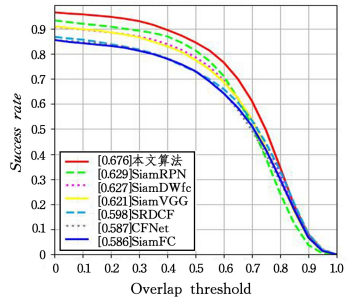
$$P=\frac{\sum_{i=1}^{n}P_i}{n}\times 100\% \tag{14}$$

fps 则为每秒平均能追踪的画面帧数。

最终实验结果如图 3 所示。在 OTB100 数据集上，本文跟踪器的成功率为 0.676，精确度为 0.887，fps 为 44。与没有模板更新的 SiamVGG 网络相比，本文算法的成功率与精确度分别提高了 5.5% 和 8%。而相比主流的 SiamRPN 和 SiamDW 算法，本文跟踪器的成功率分别提高了 4.0.7%和4.9%，精确度分别提高了 4% 和 5.9%。很明显，本文算法优于对比算法，并且可以证明本文提出的双模板更新机制对追踪效果的提升是有效的。同时，本文算法的 fps 相比不进行模板更新的追踪器有一定下降，但仍然可以满足视频追踪的实时性需求。



(a) Precision plots of OPE on OTB100



(b) Success plots of OPE on OTB100

图 3　各算法在 OTB100 上的成功率和精确度对比

Fig. 3　Comparison of sucess rate and precision rate of each algorithm on OTB100

与文献[11]、文献[14]的算法相比，本文使用了基于 APCE 的双模板更新 SiamFC 算法，模板动态更新，而对比算法均使用了固定的模板结合比例。在 OTB100 数据集中，文献[11]的算法的成功率为 0.614，精确度 0.834。文献[14]采用了 ResNet-22 网络，并且也有模板更新操作，而成功率和精确度分别为 0.660 和 0.886。可以看出，本文算法在并未采用 ResNet 的情况下，依然凭借改进的模板更新策略取得了领先。

在 VOT2018 数据集上，本算法的成功率为 0.596，EAO 为 0.381。作为对比的 SimaRPN 的成功率则为 0.584，EAO 为 0.383。由于 VOT 数据集的测试存在算法追踪的目标框脱离真实目标框，追踪器重启并重新开始追踪的机制。因此，本文算法的模板更新带来的暂时跟丢目标后可以重新跟上的能力并不能体现出来，在此数据集上的结果对比中 SiamRPN 的优势并不明显。

为了与精细化地分析每种算法的优势与不足，本文以 OTB 数据集中标注的 11 个属性为标准，包括：尺度变化(SV)、遮挡(OCC)、光照变化(IV)、形变(DEF)、运动模糊(MB)、快速运动(FM)、平面内旋转(IPR)、平面外旋转(OPR)、超出视野(OV)、背景杂波(BC) 和低分辨率(LR)，这些属性以精确度作为评价指标，将本文算法与对比算法在这些指标上的结果进行详细对比。最终结果如表 5 所列。

表 5　11 种属性下各算法的精确度

Table 5　Precision rate of each algorithm for 11 attributes

算法	SV	OPR	IPR	OCC	DEF	FM	IV	BC	MB	OV	LR
本文算法	0.887	0.880	0.899	0.824	0.846	0.837	0.852	0.900	0.874	0.826	0.996
SiamRPN	0.838	0.851	0.854	0.780	0.825	0.789	0.859	0.799	0.816	0.726	0.978
SiamDWFC	0.819	0.829	0.824	0.798	0.763	0.808	0.794	0.762	0.841	0.781	0.901
SiamFC	0.736	0.758	0.743	0.723	0.691	0.744	0.736	0.692	0.707	0.673	0.900
SRDCF	0.745	0.741	0.745	0.734	0.734	0.768	0.792	0.775	0.765	0.594	0.760
CFNet	0.731	0.759	0.786	0.699	0.714	0.705	0.707	0.756	0.680	0.601	0.888
Staple	0.726	0.737	0.768	0.726	0.752	0.708	0.783	0.749	0.698	0.664	0.690

可以看出，本文算法除了在光照变化(IV)属性的精确度上略逊于 SiamRPN 算法之外，在其余属性上的表现均为第一。

3.3 定性分析

本文算法提出的基于 APCE 的双模板异步更新是为了解决目标与初始模板相比变化较大，以及视频、目标剧烈变化，导致追踪错误失效的问题。为了验证本文算法的有效性，本文从 OTB100 数据集的全部跟踪结果中选取了部分，如图 4 所示，从以下面 4 个方面进行定性分析。

(1)尺度变化：以视频 dog1 和 clifbar 为例。视频 dog1 中从 900 帧开始出现了明显的尺度变化，而有模板更新机制的算法理论上能更好地适应这一变化。对比算法 SiamDWfc 和 SiamRPN 虽然都能跟上目标，但本文算法的精确度高达 0.953，而对比算法为 0.921 和 0.929，显然本文算法具有一定的优势。由于浅层特征提取可以获得目标的位置信息，因此可以更好地追踪大小变化的目标。

(2)遮挡：以视频 coupon 为例，视频在 126～131 帧的时间段将一叠纸分成了两叠，如图 4 所示，本文算法因为有基于 APCE 的模板更新机制，在物体被遮挡的过程中降低了上一帧所用模板的融合比例，提高了模板更新的质量，从而始终跟住了目标，精确度达到了 0.821。SiamDWfc 和 SiamRPN 则直接跟丢了目标，精确度仅为 0.300 和 0.365。

(3)平面内旋转：以视频 clifbar 为例，本文算法的精确度达到 0.85，而 SiamDWfc 和 SiamRPN 的精确度仅为 0.620 和 0.392。如图 4 所示，视频在 150 帧开始出现了旋转，对算法的特征提取能力有极高的要求。本文算法在目标旋转过程中的追踪结果始终最接近目标的真实位置，而 SiamRPN 和 SiamDWfc 偏差较大。因此，本文算法可以有效地适应旋转这一特性。

(4)模糊和快速运动：以视频 ironman 为例，视频中目标运动速度极快，产生了大量模糊。如图 4 所示，在场景的剧烈变化中所有算法都跟丢了目标，然而本文算法由于更新过模板，在后续重新跟住了目标，精确度达到了 0.726，远高于对比算法的 0.456 和 0.587。本文算法在模板更新时考虑了 3 个分支的模板，并基于 APCE 降低了上一帧模板的融合比例，相应地提高了当前帧的融合比例，并且由于存在初始模板，因而在这类场景下能够减少对模糊信息的误判，同时保持对快速变化的目标进行信息提取，从而表现出更好的鲁棒性。

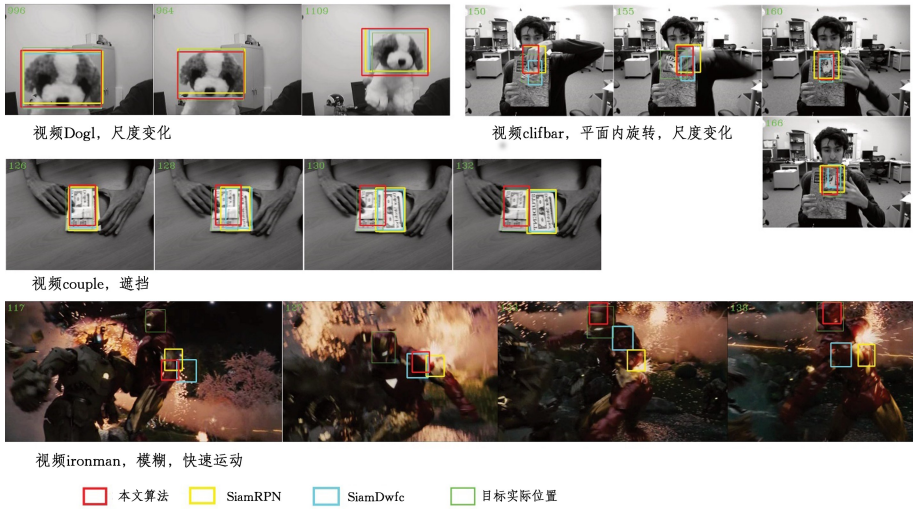


图 4 本文算法与对比算法的追踪结果

Fig. 4 Tracking results of the proposed algorithm and comparison algorithm

结束语 针对目标跟踪中普遍存在的遮挡模糊、特征提取能力不足的问题，本文提出了基于 SiameseFC 的双模板异步更新的追踪算法。算法的模板更新机制利用了初始、前一帧、当前 3 种分支，并结合平均峰值能量 APCE，以动态更新的方式提高模板更新的鲁棒性，降低在目标被遮挡、模糊情况下的跟丢风险，并且以异步的形式分别更新深浅层模板。在 OTB100 中的基准测试结果表明，本文算法的准确率和精确度较主流算法均有一定提升，并且能保持 44～46 fps 的速度以满足实时追踪需求。目前，最新的目标跟踪算法如 SiamRPN++用均匀分布采样解决了平移不变性问题，并采用了残差网络 ResNet，其在 OTB 的成功率达到了 0.696。后续研究将考虑把本文思路运用到此类算法中。

参 考 文 献

[1] HOU Z Q, HAN C Z. A Survey of Visual Tracking[J]. Acta Automatica Sinica, 2006(4): 603-617.

[2] GAO W, ZHU M, HE B G, et al. Overview of target tracking technology[J]. Chinese Optics, 2014, 7(3): 365-375.

[3] BOLME D S, BEVERIDGE J R, DRAPER B A, et al. Visual object tracking using adaptive correlation filters[C]// The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010). San Francisco, CA, USA: IEEE, 2010: 2544-2550.

[4] MA C, HUANG J B, YANG X, et al. Robust Visual Tracking via Hierarchical Convolutional Features[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(11): 2709-2723.

[5] QI Y, ZHANG S, QIN L, et al. Hedged Deep Tracking[C]// Computer Vision & Pattern Recognition. IEEE, 2016: 4303-4311.

[6] BERTINETTO L, VALMADRE J, HENRIQUES J F, et al. Fully-Convolutional Siamese Networks for Object Tracking

[C]// European Conference on Computer Vision. Cham: Springer, 2016.

[7] LI Y, ZHANG X. SiamVGG: Visual Tracking using Deeper Siamese Networks[J]. arXiv: 1902.02804, 2019.

[8] BO L, YAN J, WEI W, et al. High Performance Visual Tracking with Siamese Region Proposal Network[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018.

[9] LI B, WU W, WANG Q, et al. SiamRPN++: Evolution of Siamese Visual Tracking With Very Deep Networks[C]// 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR). IEEE, 2020, 32(10): 4475-4489.

[10] ZHANG Z, PENG H. Deeper and Wider Siamese Networks for Real-Time Visual Tracking[C]// 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020.

[11] HOU Z Q, CHEN L L, YU W S, et al. Robust Visual Tracking Algorithm Based on Siamese Network with Dual Templates[J]. Journal of Electronics & Informational Technology, 2019, 41(9): 2247-2255.

[12] SONG J H, ZHANG J, LIU Y J, et al. Conditional generative adversarial siamese networks for object tracking[J]. Control and Decision, 2021, 36(5): 1110-1118.

[13] CHEN F J, XIE W X. SiamVGG Network Target Tracking Algorithm with Anti-occlusion Mechanism[J]. Signal Processing, 2020, 36(4): 562-571.

[14] REN L C, YANG J Q, WEI Y X, et al. Tracking algorithm using siamese network based on feature fusion and dual-template nested update[J]. Computer Engineering, 2021, 47(7): 239-248.

[15] ZHAO F, ZHANG T, WU Y, et al. Antidecay LSTM for Siamese Tracking With Adversarial Learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 32(10): 4475-4489.

[16] WANG M, LIU Y, HUANG Z. Large Margin Object Tracking with Circulant Feature Maps[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2017: 4800-4808.

[17] DANELLJAN M, HAGER G, KHAN F S, et al. Learning Spatially Regularized Correlation Filters for Visual Tracking[C]// 2015 IEEE International Conference on Computer Vision (ICCV). IEEE, 2015.

[18] VALMADRE J, BERTINETTO L, HENRIQUES J F, et al. End-to-End Representation Learning for Correlation Filter Based Tracking[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition(CVPR). IEEE, 2017.

MA Han-da, born in 1966, master, professor. His main research interests include data mining, big data technology research and applications.