学校代码：10284
分 类 号：TP302.7
密　　级：公 开
U D C：004.93
学　　号：MG20370046

南京大学

硕士学位论文

论 文 题 目　面向真实场景的跟踪算法应用研究

作 者 姓 名　许翔

专 业 名 称　计算机科学与技术

研 究 方 向　智能系统与应用

导 师 姓 名　申富饶

2023 年 5 月 24 日

答辩委员会主席 戴新宇

评 阅 人 路通

徐明华

论文答辩日期 2023 年 5 月 22 日

研究生签名：

导师签名：

Tracking Algorithms in Real Scenarios

by

Xu Xiang

Supervised by

Professor Shen Fu-Rao

A dissertation submitted to
the graduate school of Nanjing University
in partial fulfilment of the requirements for the degree of
MASTER
in
Computer Science and Technology

School of Artificial Intelligence
Nanjing University

May24, 2023

南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目：面向真实场景的跟踪算法应用研究

计算机科学与技术 专业 2020 级硕士生姓名：许翔

指导教师(姓名、职称)：申富饶 教授

摘　要

机器学习已进入深度学习时代，而深度学习技术应用最为成功的领域就是计算机视觉。基于深度神经网络的算法模型已经成功应用于视频监控、人机交互、自动驾驶等任务。如今的计算机视觉研究呈现两类趋势。一是研究粒度逐渐精细。最常见的图片物体标注格式是包围框，而越来越多的现实任务需要模型输出像素级的结果。二是研究主体从图像发展为视频。应用越来越广泛的摄像头使得视频数据的采集变得容易，视频中物体运动连续性的存在可以帮助模型提高预测精度。在这样的背景下，本文选择研究视频单目标跟踪这一任务，特别是长时目标跟踪这一更加接近真实场景的任务，从网络模型和数据两个角度展开研究。本文的主要研究内容与贡献如下：

从网络模型角度，本文设计了长时跟踪与分割算法框架 SRF。现有长时跟踪框架多在短时跟踪器的基础上，通过增加组件的方式来提升跟踪精度。与之相反，本文受到集成学习中“部分可能比全部更好”这一现象的启发，对模型组件“做减法”，在让整个算法更加简洁的同时，保持了一定的跟踪精度。本文并没有设计新的神经网络。本文通过实验证明，在分析现有短时跟踪器优劣的基础上充分挖掘现有跟踪方法的潜力，也可以在长时目标跟踪任务上取得相同甚至超越现有方法的跟踪性能。

SRF 针对真实场景做了两点设计。考虑到不同现实任务中对模型运行速度的需求不同，SRF 引入了速度控制参数，使得用户可以调整长时跟踪算法的运行速度。据本文了解，SRF 是第一个速度可以连续调节的长时跟踪器。部分现实任务需要给出目标的准确位置与轮廓。SRF 的结果精炼模块可以输出目标的像素掩膜，从而同时实现长时目标跟踪与分割任务，并且在运行速度和准确性

之间得到平衡。SRF 是目前为数不多的可以同时完成视频目标分割与长时目标跟踪任务的方法。

从数据角度，本文设计了基于跟踪算法网络结构的智能标注算法 SiamAnno。本文观察到单目标跟踪中的结果精炼任务和机器辅助标注任务的相似点，将孪生网络结构应用在机器辅助标注任务中，实现了将标注人员输入的包围框自动转换成物体多边形轮廓的功能。本文的模型利用了孪生网络的单样本学习能力，使得模型可以更好的应对标注新类别物体或拍摄环境发生变化的情况。多个数据集上的实验结果也证明了模型出色的跨域标注能力。据本文所知，SiamAnno 是第一个在交互式标注任务中使用了孪生网络的模型。

为了更好地将本文的研究服务于真实场景，本文还开发了面向跟踪任务的图片标注工具系统，并将本文的算法嵌入其中。标注目标跟踪数据需要连续标注一个视频中的多张图像。基于 SRF 的“连续标注”功能可以基于用户在上一帧标注的结果推测目标在下一帧的位置；基于 SiamAnno 的“智能标注”功能可以帮助用户将包围框标注转换为多边形轮廓。两个算法从不同的角度降低了人工标注具有像素级分割掩膜的目标跟踪数据集工作量，本文系统具有广阔的应用前景。

关键词： 孪生网络；长时目标跟踪；视频目标分割；智能标注

南京大学研究生毕业论文英文摘要首页用纸

THESIS: Tracking Algorithms in Real Scenarios

SPECIALIZATION: Computer Science and Technology

POSTGRADUATE: Xu Xiang

MENTOR: Professor Shen Fu-Rao

ABSTRACT

Machine learning has entered the era of deep learning, whose most successful applications are around the area of computer vision. Algorithms and models based on deep neural networks have been successfully applied to tasks such as video surveillance, human-computer interaction, and autonomous driving. Nowadays, related research exists two trends. The first is the gradual refinement of the research granularity. Not long ago, bounding boxes are the most common format for image object annotation, but now more and more realistic tasks expect precise pixel-level results. Second, the data format has evolved from images to videos. Thanks to the wide usage of cameras, capturing video datasets becomes easier, and the moving continuity in videos may help the model improve prediction accuracy. In this paper, we study the video object tracking task. Especially, we focus on the long-term single object tracking task, which is closer to real scenarios. We conduct our research from the perspectives of both network model and data. The contents and main contributions of this paper are as follows:

We design a long-time tracking and segmentation framework SRF. Existing methods usually construct long-term trackers using short-time ones and improve the tracking accuracy by adding components. In contrast, we get inspiration from the phenomenon of "many could be better than all" in ensemble learning, do subtraction to the number of components, simplifying the entire method but still maintaining satisfactory tracking accuracy. What's more, we do not design new neural networks. We empirically find that by fully exploiting the potential of existing short-term trackers, it is possible to achieve the same or even better tracking performance on long-time tracking tasks.

We have two realistic-scenario-specific considerations when designing the SRF. Catering for different needs of running speed, SRF introduces a speed control parameter that allows users to adjust the inference speed of the long-time tracking algorithm. To the best of our knowledge, SRF is the first long-time tracker whose speed is continuously adjustable. Thanks to the refinement module, SRF is also one of the few methods that can perform both video object segmentation and long-time object tracking tasks. SRF reaches a balance between time and accuracy.

Precisely annotated data is also vital for training a tracking model. We design a smart annotation algorithm named SiamAnno, basing on the network structure of the short-term trackers. We hold the opinion that refining the results in a tracking framework is similar to annotating objects' masks given the bounding box annotations. As a result, we apply the Siamese network to our machine-assisted annotation model, which automatically converts the annotator's bounding-box inputs into polygon contours. By exploiting the one-shot learning ability of the Siamese architecture, our method shows great potential in handling cross-domain annotation tasks in which domain and environment shift frequently happen. To the best of our knowledge, SiamAnno is the first model that uses Siamese networks in an annotation model.

We also develop a tracking image annotation system by embedding the two algorithms we design in this paper, to better serve the real-world tasks. Annotating an object tracking dataset requires annotation of successive images in a video. The SRF-based "successive annotation" function infers the target's bounding box in the next frame using annotation results in the previous frame. The SiamAnno-based "smart annotation" function converts the bounding box annotation into polygon contours. These two functions reduce the workload of manually labeling segmentation masks in object tracking datasets from different perspectives. Our system has a broad application prospect.

KEYWORDS: Siamese Network, Long-Term Object Tracking, Video Object Segmentation, Smart Annotation

目　　录

插图清单

附表清单

第一章　绪论

1.1　研究背景与意义

信息技术的不断发展，推动人们将以机器学习为代表的新兴人工智能技术应用于现实任务中。例如，学术界和产业界人士认为全球经济正处于第四次工业革命的开端，提出“工业 4.0”这一概念，推动工业制造智能化来促进产业变革；我国科技部、工信部等六部门也于 2022 年 8 月印发《关于加快场景创新以人工智能高水平应用促进经济高质量发展的指导意见》，着力解决人工智能重大应用和产业化问题，全面提升我国的人工智能发展质量和水平。而在繁多的机器学习技术中，最热门的莫过于深度学习技术。

经典的机器学习算法依赖于人基于先验知识或经验构造的特征。相比之下，深度学习可以实现“端到端”（end-to-end）的学习与预测，直接将自然语言文本或图像等数据以其原始形态作为输入，模型就可以在其基础上自动学习特征，并进行预测。如今的“深度学习”在很大程度上是“深度神经网络学习”。以 AlexNet[1] 在 2012 年的 ImageNet[2] 图像分类竞赛中的一战成名作为代表性事件，以神经网络为核心的深度学习技术迅猛发展，并在多个领域不断刷新现有的机器学习性能指标。

在这些领域中，深度神经网络最成功的应用领域当属计算机视觉领域。从图像分类，到目标检测、实例分割、目标跟踪，深度学习方法在这些视觉任务中大放异彩，并成功应用于现实的工业任务中。如今的视觉研究呈现两种趋势。一是研究粒度逐渐精细。例如，图像分类模型将一张图像整体作为分类粒度，而目标检测算法则需要对图像中的每一个物体进行分类并判断大致位置，图像分割方法需要进一步给出每一个像素的类别归属。二是研究主体从图像发展为视频。早期研究多针对图像数据展开，然而世界是动态的，视频是更接近于人感知世界的方式，所有图像均可以看作是某个视频中的一帧。将视频作为研究主体，不仅能够更好的贴近现实环境，满足现实任务需求，还可以充分利用视频中物体的运动连续性。

在这样的背景下，本文选择研究视频目标跟踪（Video object tracking, VOT）这一任务。如图1-1[①]所示，视频目标跟踪模型在视频监控（video surveillance）、人机交互（human-computer interaction）、自动驾驶（automatic drive）等现实任务中有巨大的应用潜力（图片来自网络）。但是当前的目标跟踪研究多集中在短时跟踪任务上。在短时任务中，目标一直在视野里，跟踪模型被要求在每一帧都要报告目标的位置。相比之下，长时目标跟踪任务的跟踪器需要处理目标消失和再次出现的情况，更加符合真实场景。

图 1-1: 本文研究的应用价值

针对长时跟踪这一任务，本文除了从网络模型的角度出发，设计性能更加优良的长时跟踪算法外，还从数据的角度出发，研究如何能够以更低的代价生成更多的相关训练数据用于模型训练。在深度学习的时代，拥有详尽标注的数据集是非常重要的。特别是在计算机视觉领域，不限于目标跟踪的许多任务的结果精度已经从最初的包围框级别发展到精确的像素级别。例如，目标检测任务在近几年发展出了实例分割任务[3-5]，而每年举办的视频目标跟踪竞赛不仅有既存的包围框跟踪赛道，还衍生出了要求输出像素掩膜的赛道[6-8]。人们在将学术界的研究成果应用于现实任务中时，发现相比于模型结构的改动，增加训练数据所能带来的性能提升往往在很多时候更加明显。因此，设计可以降低标注人员工作量的智能标注算法、设计面向视频目标跟踪任务的图片标注工具系统都是很有意义的。

[①]图片来自网络

1.2 研究现状与挑战

当今的视频目标跟踪相关研究多集中于短时跟踪任务，其中的各类方法亦被称为“短时跟踪器”。本文将大多数短时跟踪器分为在线学习（online learning）方法和单样本学习（one-shot learning）方法两类。代表性的在线学习方法有ATOM[9]、DiMP[10]、PrDiMP[11]、FCOT[12]、KYS[13] 等。这些方法在跟踪过程中不断更新用于比对跟踪目标的模板，以适应跟踪目标的大小和外观变化。它们的表现非常鲁棒，在许多数据集上均保持着最优结果。相比之下，SiamRPN[14]、SiamRPN++[15]、SiamFC++[16]、DaSiamRPN[17]、SiamAttn[18]、SiamMask[19] 等方法仅使用人工在第一帧标注的目标模版，这些方法被称为单样本学习方法。它们在运行时不会对模型参数进行更新，这些方法亦被称为是完全离线的。整体而言，相较于在线学习方法，单样本学习方法的运行速度更快，且对物体的大小估计更加准确。

然而，短时跟踪器假设目标一直出现在视野中，这和现实情况有很大出入。因此，有研究者提出了长时目标跟踪任务[20]，将目标的消失与再出现纳入考虑，构建长时跟踪器。一种常见的构建长时跟踪器的方法是基于现有的短时跟踪器。现有的长时跟踪框架[21-22]，特别是历年 VOT 比赛的参赛模型[6,23-24]，多使用一个短时跟踪器作为局部跟踪器，再使用一个短时跟踪器作为验证器。然而，这些研究并没有表明为什么选择特定的短时跟踪器作为其长时跟踪框架中的相应组件，有些甚至只是简单地沿用了之前研究的做法。研究的挑战之一就是如何在在线学习方法和单样本学习方法两类短时跟踪器中精挑细选，使得长时跟踪算法可以充分发挥其中的短时跟踪器的优势。

为了弥补长时跟踪和短时跟踪任务之间的差距，既有工作在短时跟踪器的基础上引入了不同的模块来形成长时跟踪模块。LTMU 使用目标检测模型作为全局重检测器[21]，也有方法用一个目标候选关联网络[25] 跟踪多个混淆物来区分出真正的目标。图卷积网络也被证明可以提高轨迹预测的鲁棒性[26]。现有研究往往通过设计和添加新的神经网络来提高长时跟踪准确度。然而，本文通过实验发现，增加模块并不能够确定性地提高性能，现有跟踪器的能力也没有被充分挖掘。类似做法使得长时跟踪模型愈发臃肿，成为将模型应用于真实场景的又一挑战。

伴随着现今计算机视觉任务的研究粒度愈发精细的趋势，也有目标跟踪任务需要模型输出像素级的跟踪结果[6]。该任务可被视为是现有的半监督视频物体分割（video object segmentation, VOS）任务的近似，区别在于输出像素级结果的目标跟踪模型仍然使用目标跟踪领域的评价标准。然而，能同时完成长时目标跟踪和视频目标分割任务的框架相对少见[19,27]，相关方法有广阔的研究空间。

除了设计准确度更高的算法外，如何获取高质量的标注数据也是一大挑战。特别是在需要模型输出像素级结果的任务中，模型训练也依赖于像素级的标注数据集。然而，注释像素掩膜是一项极其耗费时间和人力的任务。有研究表明标注一个物体的像素掩膜需要标注者平均花费 20-30 秒的时间[28]。在智能标注的相关研究中，早期工作利用人工构造的颜色和纹理线索来区分背景和前景，其中一些方法已经在世界知名的图像处理软件中得到了应用[29-30]。近年来有工作尝试利用深度学习模型，配以不同的人机交互机制来辅助人工标注，减少人类在标注时的工作量[28,31-35]。标注工作必然面临标注之前从未见过的物体的需求。但是，本文发现现有方法并没有从网络结构的设计上考虑让模型具有少样本学习能力，使得方法在标注新的物体或在新的背景环境下标注的效果欠佳。另外，也缺少特别考虑了单目标跟踪数据集标注需求的标注工具。

总结来看，现有的长时单目标跟踪任务相关研究面临如下挑战：

- 现有长时跟踪算法在选择特定短时跟踪器作为其组件时缺乏对选择的解释。
- 通过增加组件来提高跟踪性能的做法令跟踪方法愈发臃肿。实验证明增加模块并不一定带来准确度的提高，现有短时跟踪器的能力也没有被充分发掘。
- 长时跟踪框架不仅需要输出包围框位置，还需要以像素掩膜的形式给出物体的像素级精确位置。
- 深度学习模型的训练依赖于高质量的标注数据。现有机器辅助标注算法在标注新的物体或在新的背景环境下标注的效果欠佳，也缺少特别考虑了单目标跟踪数据集标注需求的标注工具。

1.3 研究内容与贡献

首先，针对长时跟踪任务现有研究在网络模型方面的前两点挑战，本文提出了一个用于长时视频目标跟踪任务的“切换和精炼跟踪”算法框架（switch and

refine tracking framework, SRF）方法。本文并没有像既有方法那样对模型组件“做加法”，通过增加神经网络模块来提高性能，而是受到集成学习（ensemble learning）中“部分可能比全部更好”（many could be better than all）[36] 这一现象的启发，对整个长时跟踪方法“做减法”，在保持一定跟踪精度的同时，让整个算法更加简单。本文的方法从判别式在线跟踪器、单样本学习离线跟踪器和全局重检测器这三个互补的角度充分挖掘了现有方法的能力。本文的方法具有针对模版的“双重检查机制”，即在视频的每一帧都至少和第一帧模版与最新跟踪结果模版分别比对一次。本文的跟踪器间切换机制也很简单，没有引入额外的验证器，而是只使用到局部跟踪器和全局重检测器输出的置信度分数。本文还引入了一个速度控制参数（speed control parameter, SCP），来决定当短时跟踪器对自身跟踪结果不确定时交给全局重检测器的置信度分数阈值。就本文了解，SRF 是第一个速度可以连续调节的长时跟踪器。在多个具有不同评价标准的长时目标跟踪数据集[23-24,37-41] 上，本文的方法都取得了和现有最好方法相当甚至是超过最好方法的结果。

针对长时跟踪任务现有研究在网络模型方面的第三点挑战，本文方法中的结果精炼模块也为每一帧的每一个目标预测了分割掩膜。本文方法是目前为数不多的可以同时完成视频目标分割和长时目标跟踪任务的方法。与其他跟踪目标模版采用包围框而不是掩膜的工作[19,27] 相比，SRF 在速度和准确性之间达到了折中。

而针对长时跟踪任务现有研究在数据方面的挑战，本文尝试设计一个智能标注算法，来降低人工标注相关数据集时的负担。本文的目的是学习一个可以将一个物体的包围框转换成轮廓边界的分割网络。包围框通常更容易获得，这样可以减轻注释者收集高质量多边形注释的工作量。本文算法最大的创新点是本文将现有跟踪算法的网络结构应用于机器辅助标注模型中，因为本文发现单目标跟踪任务和智能标注任务的本质目的是相似的。一方面，在视频目标跟踪任务中，跟踪器被要求跟踪人工在第一帧中指定的目标，而这些目标可能不存在于训练集中。在数据标注任务中，标注之前从未见过的物体，或是在新的拍摄环境中进行标注也是必然需求。另一方面，基于孪生网络结构的跟踪模型常基于运动连续性，在目标在上一帧出现的位置附近寻找跟踪目标。在机器辅助标注任务中，标注工作者也需要以某种方式手动给出标注目标出现的大概位置。

这两个相似点启发本文可以将跟踪模型的网络结构应用于智能标注模型的设计。

然而据本文了解，现有的机器辅助标注方法并没有利用孪生网络结构的单样本学习能力。本文使用孪生网络作为本文机器辅助标注模型的骨干网络，设计了一个交互式分割网络 SiamAnno。与之前的工作相比，本文的 SiamAnno 不仅在域内标注任务中输出了清晰的物体边界，在标注新的数据集方面也显示出巨大的潜力。SiamAnno 在 ADE20k[42]、KITTI[43] 和 Rooftop[44] 上记录的最佳（state of the art, SOTA）结果表明，本文的基于孪生网络的模型在处理拍摄环境变化和标注新类别物体等“跨域”标注任务上具有显著潜力，在现实世界的跨领域注释应用中具有很大的实用性。据本文了解，SiamAnno 是第一个在交互式注释任务中使用孪生网络结构的模型。

在以上算法研究的基础上，本文实现了面向跟踪任务的图片标注工具系统。与常见的图像标注任务不同，标注目标跟踪数据集是对视频进行标注，因此可以利用本文的跟踪模型跟踪标注结果，根据标注者在上一帧对物体的标注，推断其在下一帧出现的位置。为了顺应计算机视觉任务预测粒度精细化的趋势，本文设计的智能标注算法可以基于用户输入的包围框预测物体的多边形轮廓，从而降低用户标注像素掩膜的工作负担，满足输出像素级目标位置的跟踪任务的需要。这样，本文的跟踪算法和标注算法分别为系统提供“连续标注”和“自动标注”功能，从而实现了更加智能、标注负担更低的面向跟踪任务的图片标注工具系统。

综上所述，本文的贡献主要有以下几点：

- 本文受到集成学习中“部分可能比全部更好”现象的启发，对长时跟踪方法中的组件数量做减法而不是做加法。对组件精挑细选，充分挖掘现有方法的潜力，也可取得和现有最好方法相当甚至是超过现有方法的跟踪精度。
- 本文在局部跟踪器和全局重检测器之间的切换机制引入了速度控制参数，使得整个方法的运行速度是可调的。据本文了解，SRF 是第一个速度可以连续调节的长时跟踪器。
- 本文工作是目前为数不多的结合视频目标分割任务和长时目标跟踪任务的研究之一。本文表明半监督的视频目标分割任务可以通过充分利用现有的目标跟踪模型来完成。
- 本文创新性地将跟踪模型的网络结构应用于智能标注任务中，以孪生网络作

为骨干网络设计了机器辅助标注模型 SiamAnno。该模型在处理真实标注拍摄环境变化和标注新类别物体等“跨域”标注任务上具有显著潜力。

- 本文实现了面向跟踪任务的图片标注工具系统，为用户提供基于上一帧标注推测目标在下一帧出现位置的“连续标注”功能和基于用户的包围框标注预测物体多边形轮廓的“自动标注”功能，降低标注者标注真实环境视频跟踪数据集的工作负担。

1.4 全文结构

本文围绕面向真实场景的跟踪算法，从网络模型和数据两个角度来展开研究。在网络模型方面，本文针对长时视频目标跟踪任务设计了长时跟踪算法 SRF；在数据方面，本文创新性地将跟踪算法的网络结构应用到智能标注任务中，设计了智能标注算法 SiamAnno。该算法可以反过来应用于跟踪任务中，为跟踪算法相关数据集的构建降低标注负担。本文还设计了面向跟踪任务的图片标注工具系统，并将以上两个算法作为系统的两个功能融入其中。本文一共分为六章，全文的内容结构脉络如图1-2所示。各个章节的具体安排如下：

第一章是绪论部分，围绕单目标跟踪算法，从面向真实场景的任务需求出

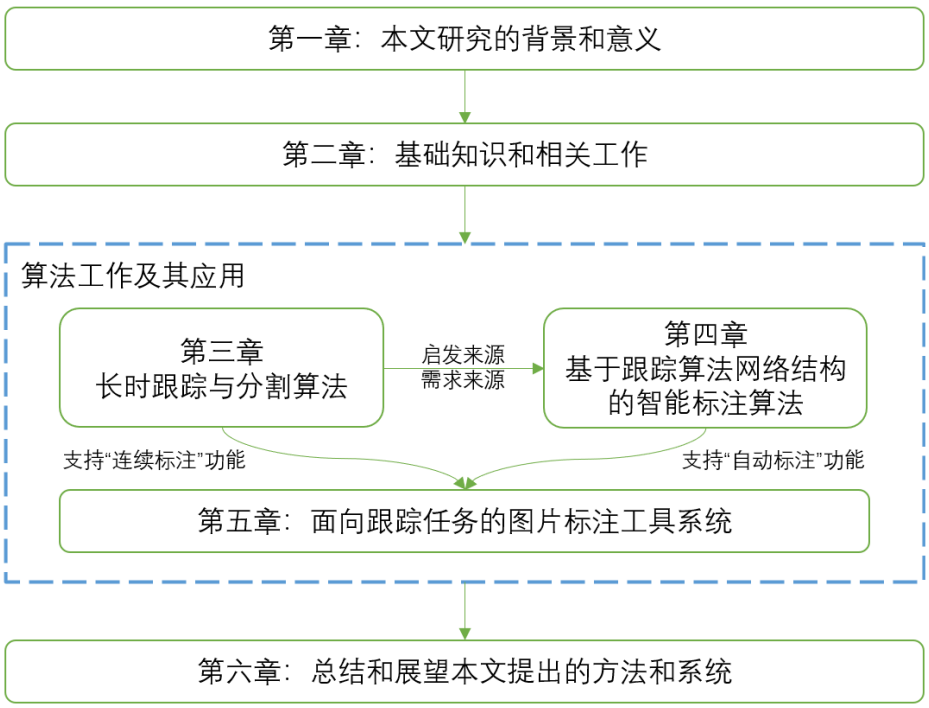

图 1-2: 本文内容结构脉络

发，分析相关的研究背景和方法应用在现实任务中的重要意义。第一章还简述了相关研究的现状和它们面临的挑战，并由此引出本文的研究内容及对相关研究领域做出的贡献。

第二章首先介绍了孪生网络的基础知识，在此基础上展开介绍了现有长时目标跟踪模型和机器辅助标注模型的研究思路和各自的优缺点。尽管本文设计的是一个长时目标跟踪模型，但考虑到现有长时跟踪模型常由短时跟踪模型构成，且本文将模型扩展到了视频目标分割任务中，因此本文在此章也介绍了短时目标跟踪模型和视频目标分割相关研究的情况。

针对面向真实场景的跟踪任务需求，第三章从网络模型的角度切入，提出了一个长时目标跟踪与分割算法 SRF。本文从算法框架的设计思想出发，详尽介绍了其中的三个组成部分：局部跟踪算法、全局重检测器和结果精炼模块。本文也描述了时间可控的模块间切换机制的设计细节，并展示了在多个数据集上的对比实验和消融实验的结果。

第四章则从数据的角度切入。本文受到第三章的结果精炼模块将预测的包围框转换成像素掩膜的优异能力的启发，将其改造，提出智能标注算法 SiamAnno，来降低标注真实场景跟踪数据集的工作量。该章首先介绍了模型的整体结构，并重点描述了其中轮廓预测头的实现方式。该章也提及了数据增强技巧、损失函数设计、训练方法等相关实现细节。在介绍了“域内”和“跨域”实验的概念后，本文通过在多个数据集上实验表明本文方法的有效性。

第五章围绕本文为真实跟踪任务设计的图片标注工具系统 SiamAnno Tool 展开。该章首先分析了用户在跟踪相关数据集的标注任务中的需求。本文以此出发设计系统架构。前两章设计的两个算法分别用于支持系统的“连续标注”和“自动标注”功能。该章一并介绍了相关的开发环境和实现细节。该系统进一步验证了本文提出算法的有效性和实用性。

第六章是全文的总结和展望。该章简单总结了全文工作，并展望了下一步的研究方向。

第二章　相关工作

2.1　孪生网络

孪生神经网络（Siamese neural network）最早由 Yann Lecun 等人于 2005 年提出，是度量学习的一种具体应用方式，常被用于计算相似性或匹配程度。其一般由两个分支组成，结构如图2-1所示。这两个分支有完全相同的结构，并共享网络参数。孪生网络的输入往往是“一对”数据。在图2-1中，本文用 X_1、X_2 表示这一对数据。在计算机视觉任务中，两幅图像构成这样的输入对。这两幅图像被分别送入两个分支 G_W，输出两张特征图 $G_W(X_1)$ 和 $G_W(X_2)$。两张特征图除了可以如图中所示的直接用某种度量方式来计算相似性 $E_W(X_1, X_2)$ 外，还可以经过某种融合或叠加的手段形成一张融合特征图，并被送入后续的网络中。相比于一般的神经网络，孪生网络充分利用了特征空间下的内在相似性（intrinsic similarity）。任何一种神经网络结构都可以用于孪生网络的具体实现。既有研究[45-46] 将卷积神经网络（convolutional neural network, CNN）、循环神经网络（recurrent neural network, RNN)、受限玻尔兹曼机（restricted Boltzmann machines, RBM）等结构应用于孪生网络中，在不同领域得到了广泛的应用。典型的应用领域有人脸验证[47]、数据降维[48]、相似性学习（similarity learning)[49]、单样本图像识别（one-shot image recognition)[50], 以及自然语言处理（natural language processing, NLP）的相关任务[51-53]。

近年来，孪生网络已经成为视频目标跟踪模型中的标准结构之一。这些基

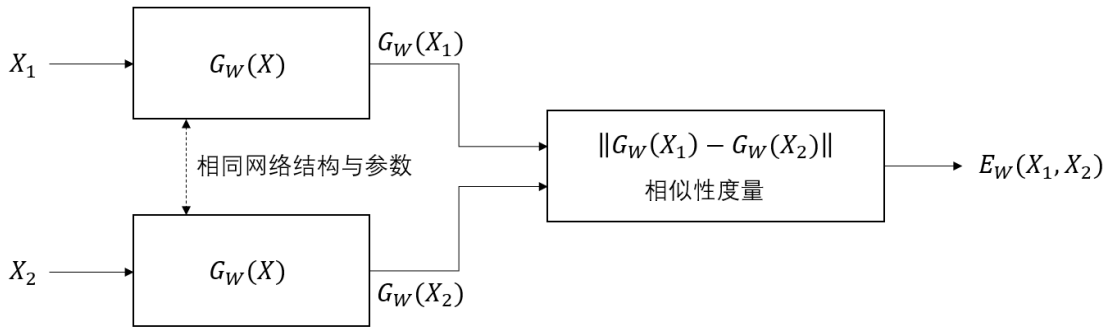

图 2-1: 孪生网络结构

于孪生网络的跟踪模型精度高，速度快，在相关数据集上多次创造了新的性能记录[14-16,54-56]。在下一小节，本文将对视频目标跟踪的相关工作做详细介绍，以此引出本文的长时目标跟踪方法。在单目标跟踪任务中，模型需要具备跟踪之前未曾见过的物体的能力，孪生网络在这其中发挥了重要作用。而在计算机视觉数据集的标注工作中，标注之前未曾见到的物体类别也是显然会遇到的情况。本文在章节2.4中以预测的输出形式将现有的交互式数据标注方法分为两类，但本文发现其中还未有利用了孪生网络的单样本学习能力来设计泛化能力更强的标注工具。为了弥补该空白，本文对现有跟踪模型进行改造，将孪生网络作为骨干网络，设计了一种新的机器辅助标注模型。该模型在跨领域标注任务上展现出了巨大的潜力。

2.2 视频目标跟踪

视频目标跟踪任务可以分为多目标跟踪和单目标跟踪两类任务。尽管从名称来看，二者的区别仅在于跟踪目标的数量多少，但实际上两类任务在评价指标等方面有着相当大的区别。本节将它们在目标数量、类别信息的先验、模版信息和主流方法上的区别罗列于表2-1中。多目标跟踪任务的实现往往更加关注帧间目标物体的匹配和轨迹的构建，常依赖于既有的目标检测方法来实现对目标物体本身的检测。在多目标跟踪任务中，测试时出现的目标也往往会出现在训练集中，也即多目标跟踪的模型不需要有跟踪任意类别目标的能力。然而在单目标跟踪任务中，跟踪器基于目标物体在第一帧的位置在视频中跟踪该物体，该物体有可能属于模型之前从来没有见过的物体类别。用一个四周与视频画面边界平行的包围框包围该物体，物体的“位置”就以该包围框的坐标位置给出。这一帧上的物体被称为“模版”，包含标注位置的这一帧被称为“模版帧”。当机器已在一部分帧上预测出物体的位置时，这些预测出来的物体位置和其所在的帧亦可以成为后续帧的“模版帧”。研究者常常根据视频的长度，将单目标跟踪

表 2-1: 单目标跟踪和多目标跟踪任务的对比

	目标数量	类别信息的先验	模版信息	主流方法
单目标跟踪	= 1	无	有	基于模版信息的匹配
多目标跟踪	≥ 1	有	无	基于目标检测框架

任务分为两种类型：短时单目标跟踪和长时单目标跟踪。在短时单目标跟踪任务中，目标客观上存在于视频的每一帧，任务也要求跟踪器报告目标在每一帧中的位置。但在长时单目标跟踪任务中，目标可能会从画面中消失，消失后还可能从任意位置重新出现。在长时跟踪任务中，跟踪器必须具备判断目标消失与否的能力，并在目标出现后重新检测出该目标。相比于短时单目标跟踪，长时单目标跟踪任务面对的是一个更现实但又更困难的情景。

2.2.1 短时目标跟踪

本文关注的是更加符合真实场景需求的长时单目标跟踪任务。但鉴于许多长时跟踪器[21-22,57-59]都以短时跟踪器作为基础构件，本小节先对短时跟踪器做简单介绍。很多短时跟踪器都采用前述的孪生网络的结构设计。在目标跟踪的语境下，孪生网络中的两个输入分支常被称为模板分支（the template branch）和搜索分支（the search branch）。模板分支将第一帧的真实标注或之前几帧的跟踪结果作为输入，生成深度特征图（feature map）或卷积滤波器，再与搜索分支以当前帧作为输入生成的特征图进行交叉相关性运算（cross correlation）或卷积运算。这样计算得到的热力图（heatmap）被分别送入分类分支和回归分支，产生最终的包围框估计，也即目标在这一帧的位置估计。

短时目标跟踪领域已经涌现出许多优秀的模型。SiamRPN[14]和 ATOM[9]分别被视为离线和在线方法的代表性跟踪器。许多离线方法都使用了孪生网络，因此研究者常常以 *Siam*-命名它们。既有研究表明，这两类方法中，离线方法具有更快的运行速度，并能更准确地估计物体大小，而在线方法可以充分利用以前的跟踪输出，产生更精确的位置预测和更稳健的前景背景区分结果。

在短时目标跟踪领域，研究者们尝试从各个角度来提高跟踪器的性能。除了广泛采用的修改骨干网络（backbone modification）的做法外，最近的工作还包括使用更好的元学习（meta-learning）方法（DiMP[10], FCOT[12]），设计更好的无锚点（anchor-free）方法（Ocean[60], SiamFC++[16]），尝试利用像素级信息（SiamMask[19], SiamAttn[18]）和时间信息（KYS[13], TTS[61], STEM[62]），从骨干网络（backbone network）中抽取多层特征使用（SiamRPN++[15], FCOT[12]），以不同方式进行相关型运算（AlphaRefine[55], PGNet[63]），或是在既有的目标检测网络上引入元学习的机制（MAML[64]）。近来也有方法尝试组合多个不同结构的骨

干网络[65]，或探索将 Transformer（TransformerTrack[66], TransT[67]）等新的网络结构、新的损失函数[68] 应用于跟踪网络的方法。

2.2.2 长时目标跟踪

相比于短时跟踪任务，长时目标跟踪任务更加符合真实场景的需要。这一任务最早由 Kalal 等人在 2012 年提出[20]。Kalal 等人同时设计了一个框架，提出一种“从错误中学习”（learning-from-error）的机制，让框架可以同时检测和跟踪目标。该框架被看作是第一个长时跟踪框架。从此之后，越来越多的研究者设计出不同的跟踪器，试图提高跟踪的准确度。在这些方法中，现在最广泛使用的一种设计思路就是交替使用短时跟踪器和长时跟踪器，这种设计也被称为短时长时交互机制（short-term and long-term interaction mechanism）。在一年一度的长时跟踪竞赛中，许多优秀的长时跟踪框架都是由短时跟踪器修改而来，或者将短时跟踪器作为自身框架的一部分。这类方法需要在使用短时跟踪器的基础上设计应对目标消失的机制。基于运动连续性的假设，RLT-DiMP[58] 对目标消失后的搜索范围添加了时空距离约束，过滤掉在远处突然出现的疑似目标。此外，该方法为了提高鲁棒性，从不同角度设计了数据增强的方法。例如在训练时变换图片背景，帮助模型学习到如何真正地辨别背景和目标；在测试时对同一张图片在不同位置随机擦除得到多个图像，在这些图像的每一张上都估计目标的位置，再将结果汇总，来增加输出的鲁棒性。此外，也有一些方法用集成学习的思路，针对同一帧同时运行短时和长时跟踪器，以得到更佳的跟踪性能，如 DeepMTA[69] 和 mlpLT[7] 等方法。但是同时运行多个跟踪器需要消耗很多的计算资源。

尽管在长时跟踪框架中使用短时跟踪器是一种非常常见的做法，但是短时跟踪器输出的置信度分数和位置估计可能不适合在长时跟踪的场景下直接使用。有些前述工作[21-22,57] 增加了一个“验证器”（verifier），从短时跟踪器输出的候选目标中择优选择最终预测。这些方法在实际实现时常常用另一个短时跟踪器作为验证器。在目标丢失时，SPLT[22] 使用了滑动窗口（sliding window）机制，首先运行一个预训练后的略读模块（skimming module）寻找目标可能存在的局部区域，再对这些局部区域应用基于 SiamRPN[14] 的局部跟踪器和验证器重新找到丢失的目标。LTMU[21] 也采用了相似的策略，区别在于它使用的是目标检测

网络 FasterRCNN[70] 来预测目标可能存在的局部区域。LTMU 还引入了一个元更新器（meta-updater），用历史帧的几何和外观线索指导短时跟踪器和验证器的参数更新，从而达到不断利用最新跟踪结果的目的。

另一类设计长时跟踪器的方法是完全基于两阶段的目标检测（two-stage object detection）框架[70]，在其中的区域候选网络（region proposal network, RPN）和/或区域卷积神经网络（region with CNN features, RCNN）中嵌入一个孪生网络结构。这样做的好处是方法不依赖于运动连续性，也即不需要对跟踪目标的位置和尺度的连续性做假设，可以起到规避累积预测误差的作用。部分方法直接取在每一帧图片上的最优检测目标作为目标在当前帧的预测位置[71]，也有方法将前后帧的关联纳入考量，在“轨迹”（tracklet）尺度上选择最优跟踪结果。例如 Siam R-CNN[27] 将连续几帧内可能的目标位置组合成一个轨迹，应用动态规划（dynamic programming）算法来选择其中的最佳轨迹。实验表明这种方法可以更有效地区分跟踪目标和干扰物。DeepMTA[69] 也采用了类似的多轨迹分析方法。但是这类方法在每一帧寻找跟踪目标时都使用整个图片作为输入，模型运行相对缓慢。

此外，基于关键点的跟踪器（keypoint-based trackers）也在跟踪任务上取得了一定成绩。ALIEN[72] 通过对同一个实例在不同的条件下重复采样的方式建立一个鲁棒的目标和背景的判别器。MUSTer[73] 将目标和背景的特征关键点保存在关键点数据库中，并通过在两个连续帧之间跟踪关键点的移动和在数据库中匹配关键点的方式找到最佳结果。也有方法将多任务学习（multi-task learning）和度量学习（metric learning）应用于跟踪任务中。它们在处理剧烈变化或辨别干扰物方面显示出一定的能力。然而，这些基于关键点的方法均采用各种人工构造的描述符（descriptor）来表示关键点特征，并应用经典的机器学习算法进行分类，这限制了它们的跟踪能力。

2.3 视频目标分割

视频目标分割任务可以被看作是视频目标跟踪任务的扩展。该任务要求跟踪模型报告目标在视频每一帧中的分割掩膜（segmentation mask）。这要求跟踪器不仅可以精确预测物体在每一帧的位置，还要具备估计物体轮廓的能力。视

频目标分割任务比视频目标跟踪任务更加困难。在一些早期工作中，粒子滤波器和颜色直方图（color histogram）相似性等经典方法被用于构建跟踪器[74]。这些方法仍然很大程度上依赖于人的先验知识和经验，而缺乏自动学习特征的能力。视频目标分割任务可以被分为两大类。一类任务只要求模型在测试时识别训练集中已有的物体类别[75]，被称为“有监督”的视频目标分割任务。本文关注的是另一类任务。在任务中，跟踪器需要根据在视频第一帧给出的分割掩膜注释来追踪目标，目标所属类别可能并没有出现在训练集中。此类任务被称为“半监督”（semi-supervised）的视频目标分割任务，跟踪未曾见过的物体这一需求也和前述的视频目标跟踪任务类似。对于这一类任务，现有的绝大多数方法[76-80]均使用第一帧的分割掩膜进行初始化，不同方法在运行速度和准确度等指标上各有优劣。而本文尝试只使用第一帧给定的包围框标注来预测后续帧的分割掩膜。这一问题假设在学术界受到的关注还相对较少，既有的相关研究包括 LWL[81]、SiamMask[19] 和 SiamR-CNN[27]。这其中，LWL 和 SiamMask 是针对长度较短的视频设计的，只有 SiamR-CNN 能同时进行长时跟踪和视频目标分割。但 SiamR-CNN 预测分割掩膜的流程是先输出包围框估计，再将包围框送入一个掩膜预测模块得到像素级的结果。

如今，无论是视频目标跟踪任务还是视频目标分割任务，相关研究的开展都在尝试更加贴近真实环境。例如，无人机的广泛使用带来了从空中视角进行检测、跟踪和统计密集人群的需求。在这一方向上，现在已有研究者开发了相关的数据集和算法模型[82]。此外，考虑到获取像素级标注的高昂成本，有些方法尝试只使用更低粒度的监督信息完成视频目标分割任务，例如 SSVOS[83] 可以只基于用户输入的潦草的笔迹来实现视频目标分割。在互联网的许多流媒体平台上，一段视频往往会配有一段相应的描述文本。SPFTN[84] 同时利用多任务学习和自步学习（self-paced learning）来基于文本信息在视频中定位和分割物体。这些研究虽然可以减轻标注的工作量，但从跟踪或分割的效果来看还有提升空间。

2.4 机器辅助标注

在真实场景中应用机器学习模型时，除了需要有一个性能优良的模型外，常常还需要有高质量的标注数据集。高质量的标注数据可以借助机器辅助标注模

型来获得。机器辅助标注任务本质上是语义分割或实例分割任务的一个子问题，在一般的分割任务的基础上增加了用户可交互等需求。按照分割模型的分类方法，本节将机器辅助标注的相关模型分为输出像素预测和输出轮廓预测的两类方法，分别回顾其中的代表性研究工作。

2.4.1 输出像素预测的方法

大多数实例分割模型将分割任务建模成逐像素的分类问题，也即预测每一个像素点是前景还是后景。GraphCut、GrabCut[29] 等早期工作基于颜色和纹理线索分割图像。近年来，深度学习方法成为机器辅助标注模型的主流，并在准确性等方面超越了经典方法。DEXTR[31] 将物体的最左边、最右边、最底部和最顶部四个点定义为“极端点”（extreme point），以用户输入的这四个极端点为基础预测物体的分割掩膜。IOG[32] 则要求用户在物体中心附近点击一次，再在物体角落点击两次。这样做不仅减少了用户的点击次数，其设计的网络所输出的掩膜质量更高。FCA-Net[33] 进一步降低了标注者的负担。该工作强调用户的首次点击在整个标注流程中的关键作用，因为基于首次点击输出的结果会影响用户后续的点击行为。研究表明，将整个目标分割的过程拆分为初步分割与细节优化两步[85-88]，或引入即插即用的边界精炼模块[89-90]，有助于模型输出更准确和更规范的物体轮廓预测。

但是，上述方法的输出是用 1 和 0 表示的分割掩膜，其中 1 表示待标注的物体，也即前景，0 表示背景。虽然这些方法设计了一定的交互机制，让用户通过点击或涂鸦等简单的操作来完善预测的掩膜，但交互后掩膜的变化有时会和设想有很大不同。例如，用户往往只希望微调自己所点击的区域附近的掩膜，但常发现在远离自己点击位置的区域，掩膜预测结果也会发生变化。如果想要完全避免这种意外变化，用户就需要逐个像素地直接修改掩膜，这非常耗时耗力。相比之下，如果使用输出轮廓预测的方法，用户可以直接拖动物体轮廓上的点来修改预测结果，这样对标注者更加友好。

2.4.2 输出轮廓预测的方法

经典的输出轮廓预测的方法通常需要精心设计能量函数（energy function）。如水平集分割方法（level set segmentation）[91] 将标注物体轮廓的过程建模为迭代收缩的曲线演化过程（curve evolution），通过对能量函数不断求导来预测物体的边界。DELSE[92] 则在此框架下引入了深度学习技术，使用卷积神经网络来预测其中的演化参数，使得整个框架可以进行端到端的训练。然而，在这些方法中，用户仍然不能直接拖动边界。

在知名的 Photoshop 图片编辑软件中，名为磁性套索（intelligent scissors）[30] 的工具允许用户简单地在物体的边缘附近移动鼠标来追踪边界。随着鼠标的移动，该工具会自动生成许多种子点，用户可以拖动或添加种子点来调整工具自动生成的边缘。而在学术界中，Polygon-RNN[93] 也采用了类似的实现思路。其采用深度神经网络按顺序依次预测各个顶点的位置。其中的循环神经网络可以接受用户的修正，重新预测物体轮廓。Polygon-RNN++[28] 在其基础上改进了网络结构和训练方案，提高了输出分辨率，但仍然存在运行时间慢和顶点数量可扩展性低等问题。不同于使用循环神经网络的前两者，Curve-GCN[34] 使用图卷积网络（graph convolutional network）同时预测所有顶点的位置。DACN[94] 则引入了多任务学习框架，结合使用边缘特征和分割特征，但不擅长预测不相连的物体。针对这些因其他物体遮挡而导致的物体被分割成多个区域的情况，Split-GCN[35] 引入了用来重构轮廓点拓扑结构的分离网络（separating network），让模型可以将一个封闭的轮廓拆分成多个部分。本文的方法和这些方法相同，也是从整张图片中裁剪出待标注物体及其附近背景，将其作为输入，以此预测物体边界上各个顶点的位置。在各自对应的交互软件中，用户可以直接拖动这些方法输出的物体边界。但不同于既有研究的是，本文并没有将研究重点放在如何优化预测头，而是关注如何设计一个更好的网络结构，让模型具有更强的泛化能力，从而能够更好地处理标注之前从未见过的物体的场景。本文受到孪生网络模型在目标跟踪领域应用的启发，将孪生网络引入机器辅助标注的模型中。实验显示本文的模型可以更好地标注新的数据集。

2.5 小结

作为度量学习的一种具体应用方式，孪生网络利用了特征空间下的内在相似性，常被用于计算两个输入的匹配程度。在视频目标跟踪任务中，基于卷积神经网络的孪生网络结构常被用于实现短时目标跟踪模型。这些跟踪模型具有速度快、精度高的特点，在多个数据集上都有着不错的跟踪性能。相比于短时跟踪任务，长时目标跟踪模型更加符合真实场景的需要。既有的长时目标跟踪框架也常将短时跟踪模型作为自身的一个重要组成部分。当确信目标在画面中时，短时跟踪模型可以充分利用运动连续性，更加关注目标出现位置附近的一块小区域，在保持预测准确度的同时提高模型推理速度。而当目标消失时，长时跟踪框架则在全局画面内搜索目标。许多用于全局搜索的模型也尝试将孪生网络结构嵌入自身。在这些跟踪任务中，具有单样本学习能力的孪生网络结构为跟踪之前从未见过的物体发挥了巨大的作用。标注新的数据也是在真实场景中应用机器学习模型常常面临的需求。对于机器辅助标注领域，本文分别回顾了输出像素预测的方法和输出轮廓预测的方法。相比于输出像素预测的方法，输出轮廓预测的方法更加用户友好。既有研究从多个角度尝试提高预测轮廓的准确度，并减少用户的交互次数。这些研究往往使用同分布的训练集和测试集，通过不断完善预测头的网络结构，来提高数据同分布情况下轮廓预测的准确度，而没有从网络结构设计的角度考虑如何提高标注之前从未见过的物体类别的能力。本文观察到现有视频目标跟踪模型在跟踪新物体类别上的优秀能力，遂尝试将相关模型进行改造，引入机器辅助标注任务中，让智能标注算法拥有更好的泛化能力。据本文了解，目前未有方法将孪生网络结构应用于样本辅助标注工具中。

第三章　长时跟踪与分割算法

3.1　长时跟踪算法概述

本文设计的长时跟踪方法 SRF 由如图3-1所示，其由一个局部跟踪器（local tracker），一个全局重检测器（global re-detector）和一个结果精炼模块（refine module）组成。在运行时，针对一个包含待跟踪目标的视频，该方法先指定局部跟踪器进行跟踪。局部跟踪器基于运动连续性假设，只在目标在上一帧出现位置的附近寻找该目标。当有足够的信心认为目标在画面中时，方法一直依赖局部跟踪器跟踪目标。只有当局部跟踪器无法发现目标时才调用全局重检测器。全局重检测器会在整个画面，也即整帧图片中定位该目标。当全局重检测器连续多帧检测到目标后，会再将任务交回给局部跟踪器。局部跟踪器和全局重检测器的输出都会被送入结果精炼模块，利用画面中更局部的信息来进一步优化预测结果，输出最终的包围框与像素级掩膜预测。局部跟踪器和全局重检测器之间需要一个切换机制。不同于一些既有研究中使用额外的验证器或时序信息处理模块的做法，本文的方法仅考虑两部分输出的置信度分数进行切换。本章

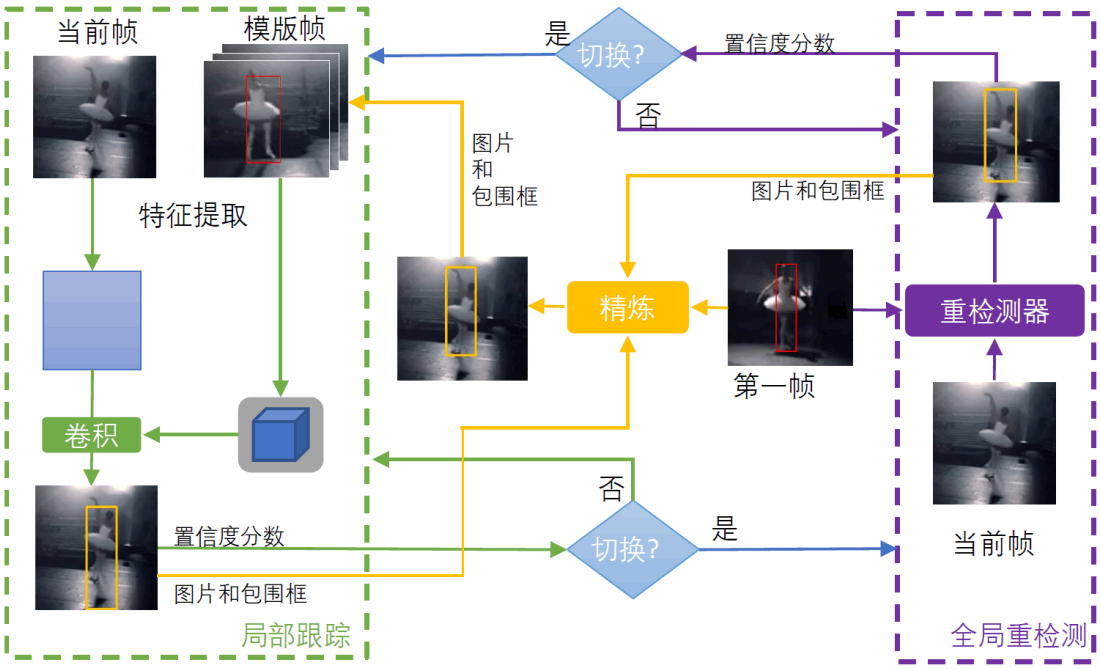

图 3-1: SRF 长时跟踪与分割算法结构图

将会深入介绍方法的各个部分。但在描述具体的实现细节之前，有必要介绍该方法的设计思想。

3.2 算法设计

3.2.1 算法的设计思想

由于更加符合真实场景的需要，长时目标跟踪任务近年来得到了越来越多的关注，各种模型和改进被不断提出。研究者们往往在前人工作的基础上，添加不同的组件，以弥补之前方法的缺点，获得更好的性能。这种做法导致最近的长时跟踪方法变得越来越臃肿，速度越来越慢。例如，LTMU 中使用了四个跟踪器（ATOM、SiamMask、SiamRPN、RT-MDNet）、一个目标检测模型（Faster R-CNN）和一个基于长短期记忆网络（long short-term memory, LSTM）的元更新器（包括 MetricNet[95]）。然而，本文实验发现组件不是越多越好。如表3-1所示，增加组件并不一定会单调地提高性能。在这一组实验中，SiamR-CNN 和 AlphaRefine 的引入提高了模型的 F-分数（F-score），但是随后再引入的两个组件并没有进一步提高模型性能，反而起到了负作用。本文分析认为，引入这两个组件后性能不佳的原因是 MetaUpdater 和 MetricNet 并没有有效地更新跟踪器使用的模板。它们可能选择了错误的帧替换掉了跟踪器原本使用的正确的模版，错误的模版导致跟踪器跟踪到错误的目标。这种错误可能会不断发生，甚至陷入恶性循环，令错误模版的影响在后续帧中累积。

表 3-1: “部分可能比全部更好”（"Many could be better than all"）

	组成	F-分数
A	SuperDiMP	0.664
B	A+SiamR-CNN	0.695
C	B+AlphaRefine	**0.705**
D	C+MetricNet	0.692
E	D+MetaUpdater	0.695

为了提高长时跟踪方法的性能，既存的研究工作通常对方法做加法，即通过添加更多的组件的方式试图提高跟踪精度。然而，更多的组件会令方法具有更高的计算复杂度，降低运行速度。单纯添加组件也并不一定能带来更好的性能。集成学习中存在一个现象，有时模型集成数量更多的方法的表现反而没有

模型数量较少的方法好，即" 部分可能比全部更好"[36]。鉴于现有的大部分长时跟踪方法也是通过集成的方法构造的，那么在长时目标跟踪任务中，是否也存在类似的现象？本文基于这样的疑问，尝试对方法的组件做减法，力图设计一个长时跟踪方法，在使用更少的组件同时仍然可以达到最佳的性能。此外，过去的研究往往通过设计一个神经网络来提高模型效果。但是本文的方法与它们不同，并没有设计一个全新的网络模型，而是试图挖掘现有跟踪器的潜力，通过利用现有方法来实现高精确度和高召回率。

本文还发现，尽管通过组合和修改短时跟踪器来构建长时跟踪器已经成为了一种默认的做法，但是现有的研究工作很少提及为何选择某个短时跟踪器作为自己的组件。正如第2.2.1节所述，近年的基于深度学习的跟踪器大体上可以被分为两类，分别是离线方法和在线方法。在线方法是判别式的（discriminative），通过在测试时不断更新目标模板来实现更高的检测结果鲁棒性。离线方法通常以 Siam-命名，更擅长大小估计，且运行速度更快。两类方法各有优缺点，因此本文试图将这两类跟踪器结合起来，令二者的优势互补。对于判别式的在线跟踪器，DiMP[10] 及其后续衍生方法在短时目标跟踪领域上曾多次刷新最优性能记录，本文基于性能和复杂度的考量从中选择最新的 TrDiMP[66] 作为局部跟踪器。离线跟踪器具有速度快的特点，它们适合作为结果精炼模块。因此，本文使用具有孪生网络结构的 AlphaRefine[55] 作为精炼模块。

全局性的重新检测机制对于长时跟踪任务是十分必要的。由于跟踪器只能接受整张画面的一个子区域作为输入，许多方法采用滑动窗口的方式在整张图片的每个子区域运行一遍短时跟踪器来搜索目标。这种做法较为粗暴。其他方法使用一个目标检测模型从整张图片中寻找跟踪目标。但是目标检测模型只能识别在训练集中预先定义的物体类别，例如 COCO[96] 中的 80 个类别，而不能处理新的物体类别。GlobalTrack[71] 和 SiamR-CNN[27] 是直接将孪生网络结构嵌入既有的目标检测框架的方法。一个非常自然的想法是将这样的方法作为长时跟踪方法中的全局重检测器。然而，GlobalTrack 的跟踪性能不尽人意，SiamR-CNN 中引入的基于动态规划的轨迹择优算法也难以和其他跟踪器或模块相结合。因此，本文简化了 SiamR-CNN，并将其作为全局重检测器。与原始的 SiamR-CNN 相比，简化后的模型与其他模块容易衔接，运行速度也更快，且简化模型对模型性能的负面影响很有限（见表3–8），跟踪精度仍然比 GlobalTrack 更高。

3.2.2 局部跟踪算法

在跟踪任务的一开始，目标有很大概率是存在于画面中的。在类似于这种方法有足够的信心认为目标在画面中的情况下，局部跟踪器利用运动的连续性来跟踪目标。局部跟踪器输出置信度最高的包围框对应的位置预测值 $\text{bbox}_{\text{st}}(\mathbf{x})$ 及其置信度得分 $\text{score}_{\text{st}}(\mathbf{x})$。本文用 $\mathbf{x}$ 来表示送入模型的搜索区域。该区域通常是整张画面中的一块子区域，其以目标在上一帧的位置为中心，大小是跟踪目标长宽的一定倍数。模型输入不使用整个画面而是专注于一个小的区域可以提高局部跟踪器的准确性并加快模型的运行速度。本文用 $\mathbf{x}$ 同时表示整帧画面或整帧画面的子区域，不做具体区分。这样既简化了表达，又不影响读者理解方法的原理。

理论上，任何一个现有的短时跟踪器都可以作为局部跟踪器使用。本文的方法使用 TrDiMP[66] 作为局部跟踪器。该方法不仅被证明是一种可靠的判别性跟踪方法，也利用了 Transformer 在捕捉时序上下文方面的优势。

本文用 Ψ 表示骨干网络，$\mathbf{z}$ 代表模板。在 Transformer 中，编码器（encoder）将一组模板 $\Psi(\mathbf{z})$ 的特征作为输入。编码器编码后的特征 $\text{Enc}(\cdot)$ 与来自搜索分支的特征 $\Psi(\mathbf{x})$ 被一同送入解码器（decoder）。编码器同时生成一个卷积核（convolution kernel）或一个跟踪模型 f。元学习器对该卷积核以岭回归的方式优化：

$$\min_{f} ||f * \text{Dec}(\text{Enc}(\Psi(\mathbf{z})), \Psi(\mathbf{x})) - \mathbf{y}||_2^2 + \lambda ||f||_2^2. \tag{3-1}$$

局部跟踪器在整个训练集上优化公式 (3-1) 中的目标函数。在实际任务中，很难只用一个坐标定义跟踪目标的准确位置。相比之下，高斯分布（Gaussian distribution）可以将一个点坐标转换成热力图形式的软标签，提供更丰富的信息，使学习过程更容易。在公式 (3-1) 中，$\mathbf{y}$ 是模板输入 $\mathbf{z}$ 经高斯分布转换后的真实标签，λ 是正则化项。这样的元学习方式可以捕捉到每个视频的独特特征。卷积核进一步与解码器输出的特征 $\text{Dec}(\cdot,\cdot)$ 进行卷积，生成最终的特征图 $\mathbf{r}$，

$$\mathbf{r} = f * \text{Dec}(\text{Enc}(\Psi(\mathbf{z})), \Psi(\mathbf{x})). \tag{3-2}$$

方法将在二维特征图 score_{st} 上的最高分对应的位置作为目标存在的位置，也即

$$\text{score}_{\text{st}}(\mathbf{x}) = \max_{i,j} \mathbf{r}_{i,j}. \tag{3-3}$$

本文按照 DiMP[10] 的实现方式，用 IoUNet[9] 预测包围框的位置和大小 bbox_{st}。

Transformer 是近来目标跟踪任务中的一个热门研究话题[66-67]。在自然语言处理领域，Transformer 常被用来建模序列关系，一些跟踪器[67] 也采用了类似做法。而在本文的局部跟踪器中，Transformer 用来替代在目标跟踪任务中被广泛使用的交叉相关性运算。关于编码器和解码器的更多细节可以参考[66]。

3.2.3 全局重检测器

当目标被认为在画面之外，或者方法对局部跟踪器输出的结果没有足够的信心时，就需要在整个画面中搜索该物体。目标检测模型可能可以用于这种情况，但目标检测任务和目标跟踪任务有很大的区别。目标检测任务中的目标属于预先定义的物体类别，而目标跟踪中的目标是由用户在第一帧的标注定义的。有方法[21] 直接将一个目标检测模型和一个跟踪器结合在一起使用，而本文则使用嵌入了孪生网络结构的目标检测模型在整张画面中寻找目标。这样做的好处是既保留了目标检测模型可以进行全局目标检测的能力，又可以处理属于之前没有见过的物体类别的目标。

SiamR-CNN 网络是现有两阶段目标检测网络的变体。原始的 SiamR-CNN 模型[27] 利用两个三级级联网络（cascades）[97] 进行重检测。这两个三级级联网络中一个将第一帧的特征作为模版，另一个则使用上一帧的特征。模型输出多个可能的目标框（proposal），每一个目标框都会被分配为某一个小轨迹的一部分，而最终的轨迹预测由所有小轨迹中分数最高的组成。这种基于小轨迹的目标跟踪方案在区分干扰物上有一定效果，但其运行非常耗时，且难以将其他短时跟踪器预测的目标框也融入该轨迹匹配框架中。为了解决原始的 SiamR-CNN 模型和短时跟踪器难以兼容的问题，本文方法使用了 SiamR-CNN 的简化版本，如图3-2所示。在全局重检测器中，搜索帧 $\mathbf{x}$ 和模板帧 $\mathbf{z}_0$ 分别被送入骨干网络 Φ，产生 $\Phi(\mathbf{x})$ 和 $\Phi(\mathbf{z}_0)$。RPN 网络基于 $\Phi(\mathbf{x})$ 输出候选框（region proposal），再

由 RoI-Align[98] 分别提取模版帧和这些候选框上的深度特征。

$$F(\mathbf{x}) = \text{RoIAlign}(\text{RPN}(\Phi(\mathbf{x}))), \tag{3-4}$$

$$F(\mathbf{z}_0) = \text{RoIAlign}(\Phi(\mathbf{z}_0)). \tag{3-5}$$

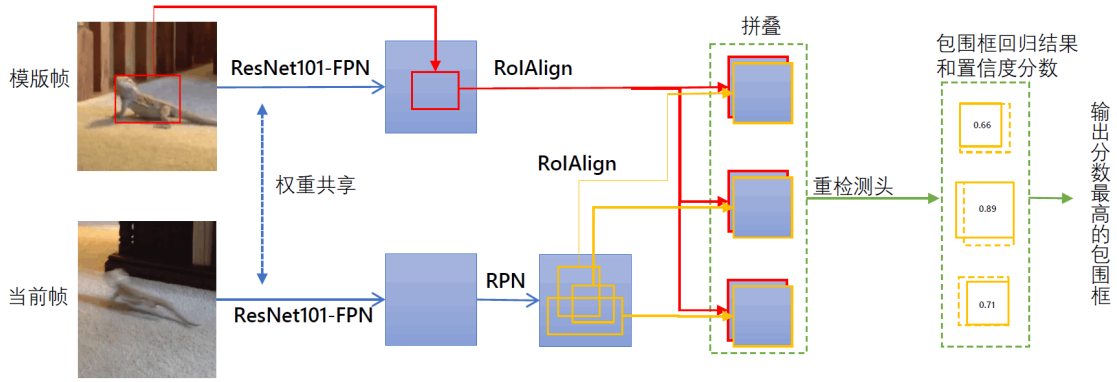

图 3-2: 全局重检测器的结构图

在对两个特征进行拼接（concatenate）后，重检测头（re-detection head）根据合成的特征 $[F(\mathbf{x}), F(\mathbf{z}_0)]$ 预测包围框的位置与大小，其中 $[\cdot,\cdot]$ 表示拼接操作。拼接后的特征拥有两个通道。对其使用 1×1 的卷积，得到只有一个通道的特征。此时特征的形状与 Faster R-CNN 中相同位置的特征形状相同，因此可以直接将现有目标检测模型的检测头在不修改其结构的情况下作为本文方法中的重检测头。在重检测头预测出每一个像素点的置信度分数后，方法在热力图 $\mathbf{x}$ 上挑选出置信度分数最高的 $\text{score}_{\text{lt}}(\mathbf{x})$ 的包围框，输出其位置和大小作为目标在该帧的定位结果，如下所示，

$$\mathbf{r}' = \text{RedetectionHead}([F(\mathbf{x}), F(\mathbf{z}_0)]), \tag{3-6}$$

$$\text{score}_{\text{lt}}(\mathbf{x}) = \max_{i,j} \mathbf{r}'_{i,j}. \tag{3-7}$$

重探测头由一个回归分支和一个分类分支构成。回归分支负责预测先验框从默认设定回归到当前帧物体的位置和形状所需要的偏移，分类分支判断每一个先验框中是否包含跟踪目标，输出相应的置信度分数。训练时，本文对回归分支应用 smooth L_1 损失 L_{reg} 来学习，对分类分支使用经典的交叉熵损失函数 L_{cls} 来学习。本文用一个平衡参数来平衡这两个损失函数对模型造成的影响，在

本文的实现中该平衡参数被默认设置为 10。

$$L_{lt} = 10 * L_{reg} + L_{cls}. \tag{3-8}$$

为了更好地拼接候选框和模版帧的特征，本文方法使用了 RoIAlign。只有当局部跟踪器未能找到物体时，方法才会唤醒全局重检测器。这种情况往往意味着目标的长宽比等尺寸发生较大幅度的变化。RoIAlign 可以在特征层面上一定程度抵消这种变化，使模板帧和搜索帧的匹配更加容易。本文认为在这种情况下，基于 RoIAlign 的拼接比交叉关联操作更加鲁棒。值得一提的是，本文在全局重检测中只将第一帧作为模板，而没有采用局部跟踪器使用的模板更新机制。本文在 **z** 中标记下标"0" 以显示这种区别。拼接操作和 RoIAlign 机制一同发挥作用，提高了重检测模块的检测能力。

3.2.4 结果精炼模块

包围框的标注只包含构成包围框的四个顶点的坐标。相比之下，像素级的标注对模型学习有更强的监督作用。既有研究表明在目标跟踪任务上应用像素级标注来训练的方法可以更好地预测目标的大小[19,21]。本文在图3-3中对比了本文方法在使用结果精炼模块前后的预测输出。图中绿色包围框是未经过结果精炼模块处理的结果，红色包围框是处理后的结果。从图中可以看出红色包围框，也即经过结果精炼模块处理后的结果更加精确。然而在常见的跟踪数据集中，标注往往仅以包围框的形式存在，而不提供更细致的像素级的分割掩膜。这要求模型能够在仅给出第一帧包围框的情况下精确区分后续所有帧中的每一个像素究竟是目标还是干扰物。

现有的长时跟踪器常使用 SiamMask[19] 来进一步优化预测结果。SiamMask 本身是一个可独立使用的短时跟踪模型。在长时跟踪器中已经存在一个局部跟踪器的前提下再引入一个完整的跟踪器会令整个模型更加笨重，所以本文选择使用一个轻量级的结果精炼模块 AlphaRefine[55]。该结果精炼模块也遵循孪生网络的设计，但它比方法中的其他两个部分更小巧。在一般的短时跟踪器中，搜索分支输入 **x** 的大小往往是目标大小的四倍，但在本文的结果精炼模块中，搜索分支的输入大小仅是局部跟踪器结果 $\text{bbox}_{\text{st}}(\mathbf{x})$ 或全局重检测器结果 $\text{bbox}_{\text{lt}}(\mathbf{x})$

的两倍，这样可以让模型更加关注跟踪目标及其周围小范围背景。搜索分支和目标分支的输出会被送入同一个骨干网络 Υ，分别得到搜索特征 $\Upsilon(\mathbf{x})$ 与模板特征 $\Upsilon(\mathbf{z}_0)$。两组特征再通过像素级的相关性运算（pixel-wise correlation）$\star$ 得到相关图（correlation map）$\mathbf{r}'$，

$$\mathbf{r}'' = \Upsilon(\mathbf{x}) \star \Upsilon(\mathbf{z}_0). \tag{3-9}$$

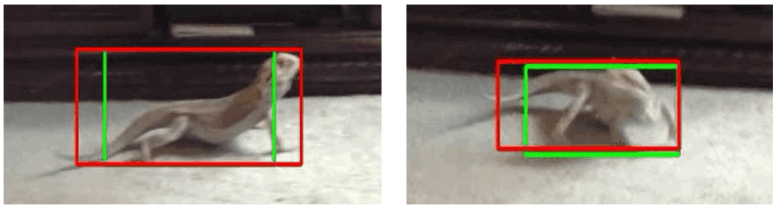

图 3-3: 结果精炼模块的效果

本文工作与既有研究的一个不同点是本文的结果精炼模块使用的是像素级相关性运算。像素级的相关性运算可以更好地保留空间信息。融合后的特征 $\mathbf{r}'$ 被分别送入一个关键点式（key-point style）的包围框预测头（bounding box prediction head）和一个掩膜预测头（mask head），得到整个长时跟踪器最终的目标包围框预测 $\text{bbox}_{\text{rf}}(\mathbf{x})$。不同于其他的短时跟踪器，AlphaRefine 本身并不输出置信度分数。AlphaRefine 也和现有的将包围框转换成分割掩膜的 Box2Seg[3] 方法不同。Box2Seg 基于预测的包围框输出预测掩膜，而本文的精炼模块同时优化包围框 $\text{bbox}_{\text{rf}}(\mathbf{x})$ 和分割掩膜 $\text{mask}_{\text{rf}}(\mathbf{x})$。不同于本文的局部跟踪器，精炼模块使用的模板始终是第一帧的人工标注 $\mathbf{z}_0$。在训练过程中，均方误差（mean squared error）L_{box} 和二元交叉熵损失（binary cross-entropy loss）L_{mask} 分别用于训练包围框输出和掩膜输出。总损失函数 L_{rf} 是这两个损失的加权和，如公式3-10所示。

$$L_{rf} = L_{box} + 1000 * L_{mask}. \tag{3-10}$$

3.2.5 时间可控的模块间切换机制

在之前的三个小节中，本文分别介绍了组成本文跟踪器的三个部分。这三个部分需要有机地结合在一起，这就需要本文设计一个决定何时使用某个组成部分的机制。这其中一个核心的问题是要如何在局部跟踪器和全局重检测器之

间进行切换。为了表达的方便，本节使用标志 st_flag 来标记跟踪器的使用状态。在开始时，st_flag 默认为真（true），此时方法使用局部跟踪器 ST()，仅在跟踪目标在上一帧出现的（预测）位置附近进行搜索。当局部跟踪器未能找到目标，或局部跟踪器对于其自身的预测感到不确定（$score_{st} < \eta$）时，方法将 st_flag 设为假（false），并唤醒全局重检测器 LT()，在当前画面的全局范围内重新搜索目标。当全局重检测器检测到目标，并且在连续 K 帧的范围内其预测的置信度分数均高于阈值 θ 时，方法就认为跟踪目标已经重新出现在画面中，并且不是被偶然检测到的。此时，SRF 将最新的预测结果更新到局部跟踪器的模板中，并将当前位置和大小等信息传给局部跟踪器。方法会重新将 st_flag 设置为真，并从下一帧开始让局部跟踪器在上一帧的全局重检测结果的基础上基于运动连续性进行局部搜索。在每一帧中，来自局部跟踪器或全局重检测器的包围框预测值会被送入结果精炼模块以得到最终输出。在这样的流程下，每一帧画面都要至少被和模版比对两次，一次在局部跟踪器或全局重检测器中，另一次在结果精炼模块中。本文把这样一个机制称为双重检查机制（double-check mechanism）。本文在算法3.1中展示了整个跟踪器对数据的处理流程。默认情况下，K 被设置为 2，θ 被设置为 0.7。

Algorithm 3.1 SRF 的数据处理流程

输入: 当前帧 **x**
输出: 精炼后的包围框预测值 $bbox_{rf}$

if st_flag 为 True **then**
  $bbox_{st}, score_{st}$ = ST(**x**)
  if 目标未找到，或目标置信度低于阈值 $score_{st} < \eta$ **then**
    设置 st_flag 为 False
  end if
end if
if st_flag 为 False **then**
  $bbox_{lt}, score_{lt}$ = LT(**x**)
  if $score_{lt} > \theta$ **then**
    $update\_counter$+ = 1
    if $update\_counter > K$ **then**
      用 LT 的结果更新 ST 的模版和状态
      重置 $update\_counter$ = 0
      设置 st_flag 为 True
    end if
  end if
end if
将 $bbox_{st}$ 或 $bbox_{lt}$ 输入结果精炼模块，得到最终预测 $bbox_{rf}$

速度是工业界将机器学习方法应用于实际任务时最为关注的一个问题。本文在决定局部跟踪器的置信度分数阈值 η 时引入了一个速度控制参数，如公式 (3-11) 所示。该参数可以控制局部跟踪器和全局重检测器之间的切换机制。利用局部跟踪器在一帧上的运行速度比全局重检测器快的这一特点，通过调节 SCP，本文可以调整被送入全局重检测器的画面数量，进而起到调节速度的作用。具体而言，在一个既存的跟踪数据集上，对于 SRF 中局部跟踪器未能成功跟踪到目标的所有帧，将局部跟踪器在这些帧上输出的置信度分数按照降序排序，而 SCP 是这些排序后的置信度分数的某个百分位数。当参数等于 0，也即默认值时，局部跟踪器中所有未成功跟踪的帧都被送入全局重检测器，此时 SRF 的结果是最优的。随着 SCP 向 1 移动，η 变得越来越小。此时即使局部跟踪器在更多的帧上未能准确地定位出跟踪目标，方法也会选择直接接受该结果而不唤醒全局重检测器重新搜索目标。参数越大，运行速度较慢的重检测器越可能被抑制，整体速度因而变得更快。

$$\begin{aligned} U &= \#\text{untracked frames in dataset using ST,} \\ \eta &= (SCP \times U)_{\text{th}} \text{ confidence score of untracked frames.} \end{aligned} \tag{3-11}$$

3.3 实验与分析

为了充分评估 SRF 的性能，本文在七个长时跟踪数据集和两个视频目标分割数据集上进行了实验。在所有实验中，除特别说明外，本文都使用在 VOT2019-LT 数据集上调整的超参数（θ, K），未针对每个数据集上进行超参数调整。

3.3.1 方法在长时跟踪任务上的表现

各类长时目标跟踪竞赛推动了这一研究领域的发展，而不同的比赛会使用不同的数据集。这些数据集各有各的特点，也对跟踪器提出了不同的能力要求。本文将在多个数据集上运行本文方法，并充分地对比本文结果和其他既有研究的结果。这些数据集中，VOT-LT 和 TLP 数据集的视频相对较长，很多时候利用运动的连续性可以较为轻松地在其中跟踪到目标物体，此时 SRF 的局部跟踪

器对跟踪的性能有很大的影响。在这类数据集上，SiamR-CNN 等基于全局目标检测的跟踪器往往效果欠佳，这也印证了在这些数据集上利用运动连续性的重要性。在 UAV 和 LaSOT 等数据集中，目标常发生非常快速的移动，此时结果精炼模块对提升跟踪准确度有很大帮助。在前述的四个数据集中，测试时跟踪目标所属的物体类别和训练数据中的物体类别是重复的。而对于 LaSOTExtSub 和 OxUvA 两个数据集的验证子集，其目标未曾出现在训练集中，也不属于预训练数据集所包含的物体类别。这要求模型具备跟踪之前未曾见到的物体的能力。得益于模板的双重检查机制，SRF 在这些视频上也表现出了较高的跟踪性能。本小节将依次介绍这些数据集，以及 SRF 在每个数据集上的表现。

VOT2018-LT 数据集共包含 35 个视频。该数据集诞生于 2018 年，被用于当年举办的 VOT 挑战赛长时跟踪赛道[23]。随后，该数据集被扩充至 50 个视频，并成为后续几年 VOT 挑战赛的指定数据集。人们习惯上将扩充后的数据集记作 VOT2019-LT。在许多目标检测研究中，人们往往划定一个固定的交并比（intersection over union, IoU）阈值，基于预测框和真实框的重复度来判断图像上的目标检测是否成功。VOT-LT 系列数据集没有简单地沿用这种评价方式。VOT 比赛官方为长时目标跟踪任务设计了一组新的评价指标：跟踪查准率（tracking precision, Pr）、跟踪查全率（tracking recall, Re）和跟踪 F-分数（tracking F-score）。计算这三个指标首先需要确定一个置信度阈值。划定不同的置信度阈值，可以得到这三个指标的不同的取值，其中 F-分数最高的一组指标值被作为最终汇报的结果。这样做一方面避免了手动设置阈值的需要，也令每一个跟踪模型都可以找到让自己性能最大化的置信度阈值，使得不同方法之间的比较都是在发挥自己最佳性能的情况下进行的。这样的评价方式令不同模型间的比较更加客观完整。对该组评价指标感兴趣的读者可参考[99] 了解详情。

表3-2展示了本文方法和其他方法在 VOT2018-LT 上的结果。本文按照跟踪 F-score 从大到小对跟踪器排名，分别用红色、蓝色和绿色标记每个指标的前三名。SRF 的 F-分数达到了 0.713，与目前最好的方法 KeepTrack[25] 持平。表3-3比较了本文方法和其他九个跟踪器在 VOT2019-LT 上的结果。和现存的其他跟踪器相比，SRF 在该数据集上也有较强的竞争力。

图3-4呈现了 SRF 和其他两个排名靠前的方法在 VOT2019-LT 数据集上的代表性结果。本文用红色、绿色和蓝色包围框分别表示 Siam R-CNN、SRF 和

表 3-2: 在 VOT2018-LT(LTB35) 数据集上的结果比较

跟踪器	F-分数	跟踪查准率	跟踪查全率
SRF (本文方法)	0.713	0.725	0.701
KeepTrack[25]	0.713	0.727	0.703
LTMU[21]	0.690	0.710	0.672
GlobalTrack-RCB[26]	0.681	0.647	0.718
Xuan et al.'s[57]	0.673	0.725	0.628
SuperDiMP[11]	0.671	0.678	0.663
SiamR-CNN[27]	0.668	0.667	0.675
TrDiMP[66]	0.653	0.673	0.635
PrDiMP[11]	0.634	0.646	0.623
SiamRPN++[15]	0.629	0.649	0.609
SPLT[22]	0.616	0.633	0.600
DeepMTA[69]	0.584	0.544	0.606
CALT[100]	0.410	-	-

表 3-3: 在 VOT2019-LT(LTB50) 数据集上的结果比较

跟踪器	F-分数	跟踪查准率	跟踪查全率
SRF (本文方法)	0.707	0.717	0.696
KeepTrack[25]	0.709	0.723	0.697
LTMU[21]	0.697	0.721	0.674
LT_DSE[24]	0.695	0.715	0.677
LTMU_B[6]	0.691	0.701	0.681
SiamR-CNN[27]	0.663	0.658	0.669
TrDiMP[66]	0.653	0.673	0.633
SuperDiMP[11]	0.647	0.654	0.641
PrDiMP[11]	0.632	0.641	0.623
SPLT[22]	0.559	0.591	0.530

DiMP 类跟踪器的跟踪结果，图中左上角的数字是该画面在原视频中的帧数。第一行图片来自 warmup 视频，该视频中的待跟踪目标是一个特定的球员。相比于 SRF，其他两个跟踪器没有处理好有干扰物的情况，混淆了跟踪目标和球场上穿着类似衣服的其他球员。值得一提的是，只有 SRF 在第 3409 帧跟踪到了正确的人。第二行图片来自 parachute 视频。该视频中的目标有很强的运动连续性，跟踪器如果能够利用好这一特点，可以很容易地定位到目标。不同跟踪器的跟踪结果印证了这一推论。例如，第 1629 帧、第 2021 帧和第 2335 帧上的结果表明每一帧都全局检测目标的 Siam R-CNN 在该视频上表现不佳，而 SRF 则凭借结果精炼模块定位到了正确的目标，并输出了更准确的结果，如第 1797 帧和第 2021 帧的结果所示。整体而言，相比于其他方法，SRF 丢失目标的次数更少，并且对物体大小的预测更加准确。

SRF 也存在一些不足。本文在图3-8中分别用红色和黄色包围框表示 SRF 的预测结果和实际真实情况，并进行对比。SRF 不擅长处理的情况主要分为两类。一类是视角发生变化（view change）时，另一类是当跟踪目标被其他物体遮挡时。当局部跟踪器或全局重检测器受到和目标外观相似的物体的干扰，或是目标被遮挡时，SRF 会输出错误的结果。该错误可能是因为 RoIAlign 不擅长处理目标被相似物体遮挡的情况。RoIAlign 本身并不具备区分物体的能力，而是通过调整物体的大小来方便特征的匹配。在这个过程中，不同尺寸的物体可能会被统一成相同的大小，例如图3-8中第二行的猫和狗。这二者在图中的大小本身相差较大，但经过 RoIAlign 后被统一成了同样的大小，这就增加了后续网络区分它们的难度。

LaSOT[37] 是目前数据量最大的有详细标注的跟踪数据集，其测试集共包括 280 个视频，每个视频的平均长度约为 2500 帧。LaSOT 使用的评价指标是精准率（precision rate）和成功率（success rate）[101]。精准率计算的是物体中心的预测位置和实际位置的距离相差小于一定阈值的帧数占到全部帧数的比例。若用物体的大小对该距离标准化，则可以按照同样的方式计算归一化精准率（normalized precision rate）。成功率是预测包围框与真实包围框的交并比在一定阈值以上的帧数与全部帧数的比例。本节通过调整该阈值，可以绘制一个成功率图（success plot），图中成功率曲线下方的面积被称为“曲线下面积”（area under curve, AUC）。一般通过比较跟踪器的 AUC 大小来对它们排名，排名结果如表3-4所示。SRF 的 AUC 达到了 67.1%，与当下最优的跟踪器 KeepTrack[25] 仅有 0.1%的差距。相比于其他方法，SRF 的精准率是最高的。

LaSOTExtSub[38] 是 LaSOT 的扩展数据集，其共包含 150 个视频序列。该数据集的特点是目标所属的物体类别与训练集不重复。它包含 15 个物体类别，这些类别不仅没有出现在 LaSOT 的训练集中，也不曾出现在常被用来做预训练的 ImageNet[2] 数据集中。这要求长时跟踪器具有跟踪以前未曾见过的物体的能力，也即所谓“一击即中”（one-shot）的能力。该数据集被认为比 LaSOT 本身的测试集更具挑战性。

本文在图3-4中最下面两行可视化地比较了本文方法与其他两种方法在该数据集上的结果。Siam R-CNN、SRF 和 DiMP 类跟踪器的跟踪结果被分别用红色、绿色和蓝色的包围框表示。针对跟踪新物体这一任务场景，即使存在干扰物

表 3-4: 在 LaSOT 数据集上的结果比较

跟踪器	精准率	归一化精准率	曲线下面积（AUC）
SRF (proposed)	70.8	75.7	67.1
KeepTrack[25]	70.4	77.4	67.2
TransT[67]	69.0	73.8	64.9
SiamR-CNN[27]	68.4	72.2	64.8
TrDiMP[66]	61.4	73.2	63.9
SuperDiMP[11]	65.3	72.2	63.1
PrDiMP[11]	60.8	68.8	59.8
GlobalTrack-RCB[26]	54.5	-	54.0
LTMU[21]	53.5	62.1	53.9
Ocean[60]	52.6	61.0	52.6
GlobalTrack[71]	52.7	59.7	52.1
DeepMTA[69]	47.4	-	52.0

或跟踪目标的移动非常混乱，SRF 仍然成功地跟踪到了目标。相比于其他两种方法，SRF 更擅长在跟踪失败后重新发现目标。例如，在图中第三行的 misc-9 视频中，跟踪器需要跟踪纸飞机。即使短暂地发生了如第 118 帧中展示的跟丢目标的情况，SRF 也在后续帧中跟踪到正确的目标。本文认为是模板的双重检查机制让 SRF 在跟踪之前未曾见过的物体方面相较于既有跟踪方法表现更优。图中第四行的 misc-10 视频中存在许多外观相似的干扰物。它们的运动非常混乱，即使是人类也很难准确地识别出目标。但 SRF 充分利用了运动连续性和模板的双重检查机制，在许多帧都跟踪到了正确的目标。

在 LaSOTExtSub 数据集上对模型的评价指标与在 LaSOT 数据集上的相同。图3-5中绘制了 12 个跟踪器在 LaSOTExtSub 数据集上的成功率图。虽然 KeepTrack[25] 的整体表现更好，但当预测包围框和真实包围框的交并比阈值高于 0.8 时，SRF 反超了 Keeptrack。这说明对于一些只需要非常准确的结果的任务，SRF 是一个更好的选择。

UAV20L 数据集[41] 共包含 20 条视频，视频的平均长度超过 2900 帧。该数据集是使用无人机收集的，视角和其他数据集有显著的不同，这也要求跟踪器必须具备处理低分辨率等空中任务特点的能力。该数据集采用和 LaSOT 类似的精确率和成功率评价指标。图3-6中比较了 SRF 和其他四个跟踪器的精准率和成功率图。与现有方法相比，SRF 在精准率上领先了 5%，在成功率上领先 4.5%。

相比于其他的数据集，TLP 数据集[39] 具有最长的视频平均时长，这让它成为研究跟踪器处理长时跟踪任务特有问题的能力的理想选择。TLP 由 50 个视频

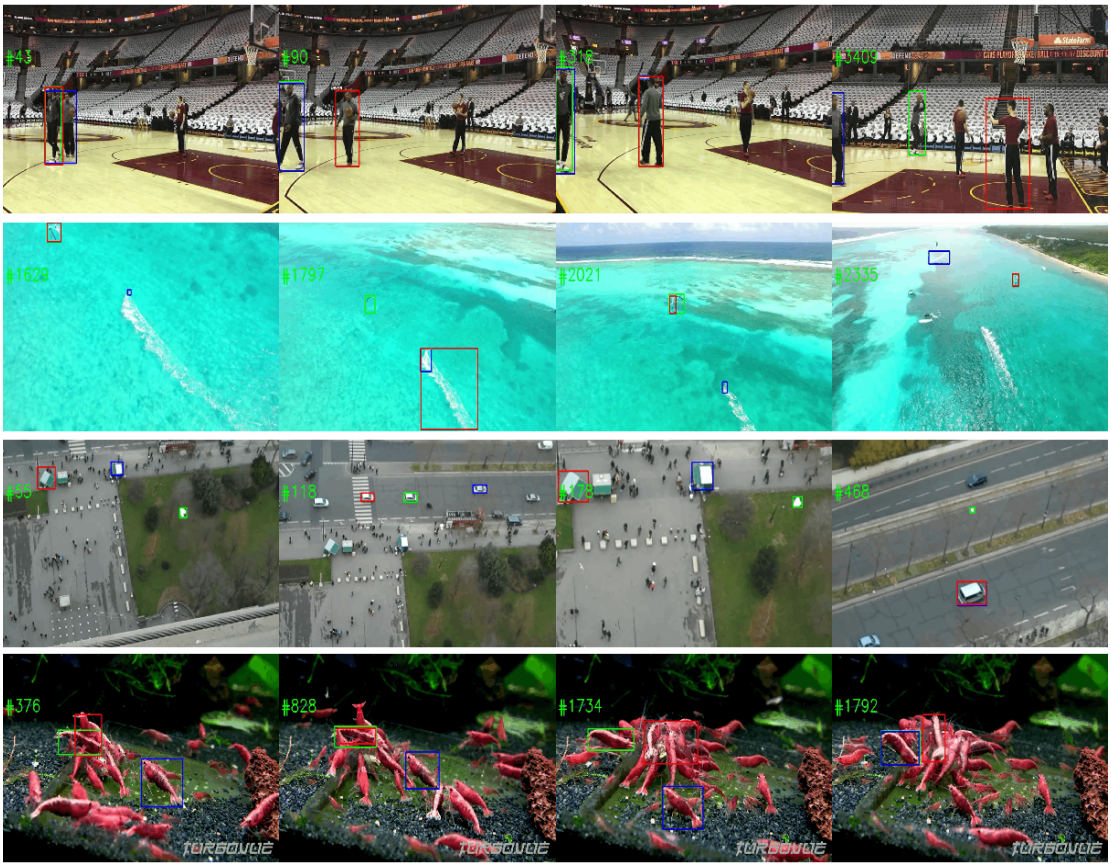

图 3-4: SRF 和现有 2 种方法在长时跟踪数据集上的代表性结果

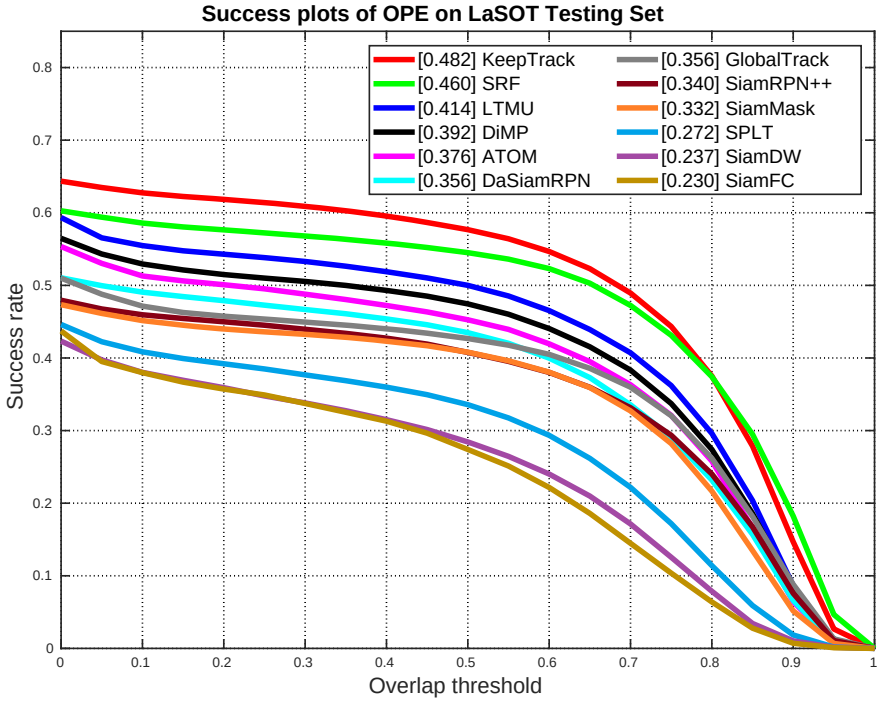

图 3-5: SRF 和现有 11 种方法在 LaSOTExtSub 数据集上的成功率图

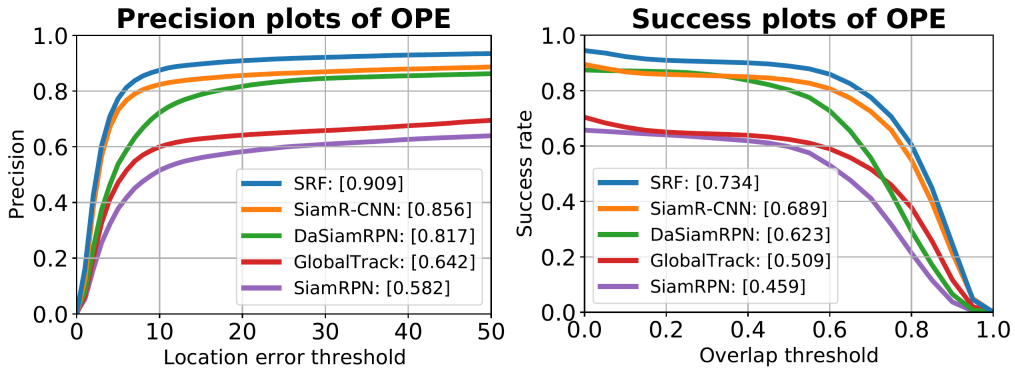

图 3-6: SRF 和现有 4 种方法在 UAV20L 数据集上的精确率图与成功率图

组成，共包含超过 67.6 万张画面。跟踪器需要在该数据集面对目标遮挡、快速运动、视角变化、比例变化等挑战。如图3-7所示，SRF 在成功率方面比之前最好的方法高出 4.5%，这表明本文方法在长时跟踪任务上具有显著优势。

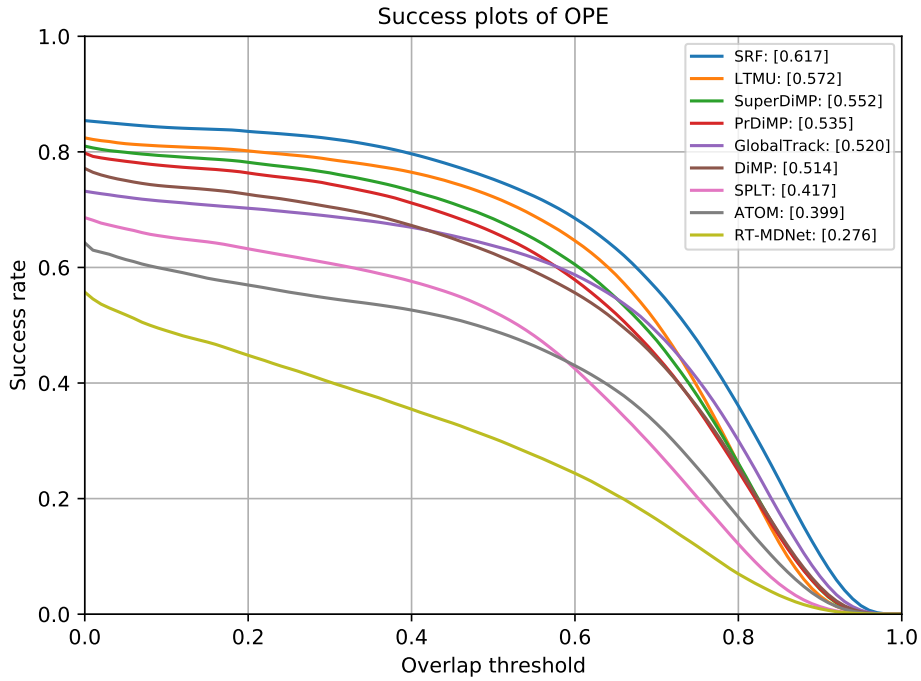

图 3-7: SRF 和现有 8 种方法在 TLP 数据集上的成功率图

OxUvA 数据集[40] 由 166 段视频组成，每段视频平均持续 2.4 分钟。视频中常伴随着物体的频繁消失。该数据集对跟踪器的评价方式与其他数据集不同。其未采用被广泛使用的 OPE（one-pass evaluation）基准，而是基于真阳性率（true positive rate, TPR）和真阴性率（true negative rate, TNR）计算二者的平均值 MaxGM。这一组评价指标要求跟踪器必须显式地输出目标是否存在于当前帧。SRF 通过设置明确的置信度阈值来判断是否发现目标。按照惯例，本文在数

表 3-5: 在 OxUvA 测试集上的结果比较

跟踪器	TPR	TNR	MaxGM
SRF (本文方法)	0.819	0.718	0.767
LTMU[21]	0.749	0.754	0.751
Xuan et al.'s[57]	0.625	0.879	0.741
SiamR-CNN[27]	0.701	0.745	0.723
SPLT[22]	0.498	0.776	0.622
GlobalTrack-RCB[26]	0.565	0.680	0.620
GlobalTrack[71]	0.574	0.633	0.603

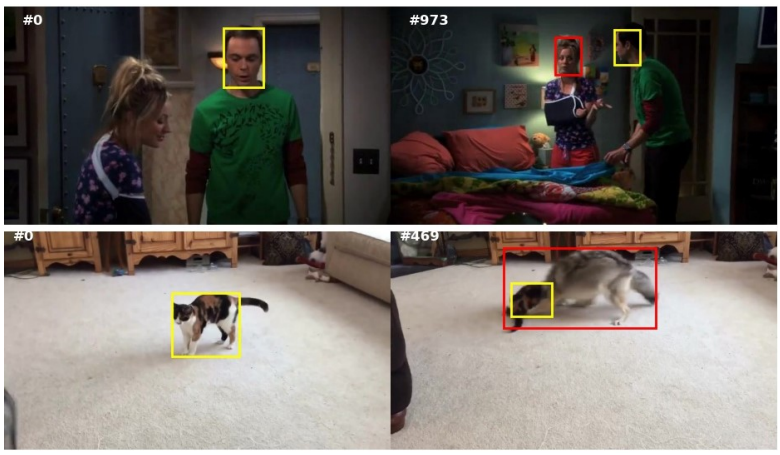

图 3-8: SRF 在 VOT2019-LT 数据集上的部分失败例

据集的开发集（dev）上对该阈值调优，将最优阈值应用在测试集（test）上，并将测试结果提交至官方评估服务器。表3-5对比了 SRF 和其他六个跟踪器在官方服务器上的评估结果。SRF 显著提高了现有方法在真阳性率这一指标上的纪录，同时刷新了 MaxGM 的最高纪录。SRF 的 MaxGM 为 0.767，高于现有最佳方法 1.6%，但本文方法在 TNR 这一指标上和 Xuan 等人的方法上有很大差距。

本文在长时目标跟踪任务的多个数据集上进行了实验。实验结果表明，不同于既有研究中通过增加模块来完善功能的思路，对跟踪器内的组件做减法，保持简单的结构同样能够在长时跟踪任务中取得优异的成绩。实验结果还说明新设计网络不是提高模型性能的唯一方法。充分挖掘现有方法的能力也可以取得与之相当的性能。

3.3.2 方法在视频目标分割任务上的表现

与前文介绍的长时跟踪数据集相比，视频目标分割数据集中的视频较短，但每一段视频都包含精确标注的像素级分割掩膜。DAVIS2017 数据集引入了两个

互补的评价标准来衡量模型在该数据集上的表现。杰卡德指数（the Jaccard index）$\mathcal{J}$ 是预测掩膜和实际掩膜的交并比大小，它衡量了模型输出与真实情况的匹配程度。F-度量（F-measure）$\mathcal{F}$ 更加关注轮廓的准确性，本文可以指定一定阈值令其容忍预测边界和物体实际轮廓之间的轻微不一致。$\mathcal{J}\&\mathcal{F}$ 是这两个评价指标的平均值。对这组评价指标感兴趣的读者可参考[75] 了解更多计算细节。

表3-6中展示了本文方法和其他现有方法在该数据集上的结果。虽然该表中包含了最新的将分割掩膜作为输入的视频目标分割模型，但出于公平性的考虑，本文主要关注与 SRF 有相同应用场景的只使用包围框作为输入的方法。在这些方法中，SiamR-CNN[27] 是和 SRF 最相似的，可以同时完成长时目标跟踪和视频目标跟踪任务。与其相比，SRF 尽管丢失了一些准确性，但有其三倍以上的运行速度。相比于现有方法，SRF 是一个在速度和准确性之间达到平衡的方法。

表 3-6: 在 DAVIS2017 验证集上的结果比较

测试时输入	训练时输入	跟踪器	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	time
包围框	包围框	SiamR-CNN[27]	0.706	0.661	0.750	0.32
		SRF (本文方法)	0.623	0.578	0.667	0.11
		SiamMask[19]	0.558	0.543	0.585	0.02
包围框	逐像素掩膜包围框	LWL[81]	0.706	0.679	0.733	0.17
逐像素掩膜	逐像素掩膜	LWL[81]	0.816	0.791	0.841	0.17
		FRTM-VOS[77]	0.767	-	-	0.05
		TVOS[79]	0.723	0.699	0.747	0.03

值得一提的是，尽管 LWL[81] 也可以只将包围框作为测试时的初始输入，但与其他三个同样使用包围框作为测试时输入的方法相比，LWL 的网络结构设计及其训练方法有很大不同。LWL 首先将分割掩膜作为输入，训练一个“完整版本”。得到该“完整版本”后，LWL 再在整个网络的头部添加一个“包围框编码器模块”（bounding box encoder module），在冻结其他网络参数的情况下单独对其训练。由此可见，LWL 即使只使用包围框作为测试时的初始输入，其训练时的初始输入仍然利用了像素级的分割掩膜标注。相比之下，其他三个方法（SiamMask[19]、SiamR-CNN[27] 和本文的 SRF）在训练和测试时都只使用第一帧的包围框标注作为输入。LWL 更高的指标分数得益于其使用了更多的监督信息。当面对只有包围框标注的现实跟踪任务时，SRF 仍有使用价值。

图3-9可视化地展示了 SRF 在该数据集上的结果。本文在图中第一帧绘制

的是实际标注，在后续帧绘制了模型的预测结果。不同于其他使用分割掩膜进行初始化的方法，即使是只使用第一帧的包围框作为输入，SRF 仍然输出了令人较为满意的掩膜预测结果。图3-10中则展示了两个 SRF 在面对物体快速移动或被遮挡时预测错误的情况。

Youtube-VOS[102] 是一个大规模的视频目标分割数据集，其验证集共包含 474 个来自 Youtube 网站的视频片段。验证集中包含 26 个未出现在其自身训练集中的物体类别，因此可以用来检验模型在新物体上的泛化能力。该数据集使用的评价指标和 DAVIS 相似，但对目标所属类别是否包含在训练集中做了区分。用于最后对不同方法进行排名的指标是 $\mathcal{J}$ 和 $\mathcal{F}$ 在包含和未包含在训练集中的物体类别上的四个评价指标 $\mathcal{J}_{seen}$、$\mathcal{J}_{unseen}$、$\mathcal{F}_{unseen}$ 和 $\mathcal{F}_{unseen}$ 的均值。本文在表3-7中对比了 SRF 和其他现有方法在该数据集的验证集上的表现。比较结果和在 DAVIS 数据集上的结果相似，即本文的 SRF 在跟踪的准确性和运行的时间开销之间取得了平衡。与 SiamR-CNN[27] 相比，SRF 为了追求更高的速度而牺牲了一些准确性，而与 SiamMask[19] 相比，SRF 在两类物体上的跟踪准确度都更高。在 DAVIS 和 Youtube 这两个数据集上的结果表明，半监督的视频目标分割任务也可以通过既有的视频目标跟踪模型来完成。

3.3.3 蒸馏实验

本文在 VOT2019-LT 数据集上展开对 SRF 的蒸馏实验。SRF 由三个部分组成，本节考察了三个部分的不同组合方式对结果的影响，结果如表3-8中所示。SRF 中使用的是 SiamR-CNN[27] 的简化版本，本节将其原始的完整版本也纳入考察的范围。在 VOT2019-LT 上，简化版 SiamR-CNN 产生的结果（0.656）与完整版（0.663）相当，但简化版的结构更简单，速度更快。SiamR-CNN 的跟踪查

图 3-9: SRF 在 DAVIS2017 数据集上的代表性运行结果

表 3-7: 在 Youtube-VOS2018 验证集上的结果比较

测试时输入	训练时输入	跟踪器	整体表现	$\mathcal{J}_{seen}$	$\mathcal{J}_{unseen}$
包围框	包围框	SiamR-CNN[27]	0.683	0.699	0.614
		SRF (本文方法)	0.623	0.662	0.537
		SiamMask[19]	0.528	0.602	0.451
包围框	逐像素掩膜包围框	LWL[81]	0.702	0.727	0.625
逐像素掩膜	逐像素掩膜	LWL[81]	0.815	0.804	0.764
		FRTM-VOS[77]	0.721	0.723	0.659
		TVOS[79]	0.678	0.671	0.630

全率高，TrDiMP[66] 的跟踪查准率高。本文结合这二者，让它们的优势互补。结合后，跟踪查准率和跟踪查全率进一步提高，F 分数也达到了 0.692。在此基础上再引入结果精炼模块，F-分数再提高 1.5%，达到 0.707。

表 3-8: SRF 中不同组件的有效性

SiamR-CNN	TrDiMP	结果精炼模块	F-分数	跟踪查准率	跟踪查全率
✓(简化)			0.656	0.652	0.659
✓(完整)			0.663	0.658	0.669
	✓		0.653	0.673	0.633
✓(简化)		✓	0.661	0.658	0.665
✓(完整)		✓	0.669	0.664	0.675
	✓	✓	0.674	0.692	0.658
✓(简化)	✓		0.692	0.699	0.685
✓(简化)	✓	✓	**0.707**	**0.717**	**0.696**

图 3-10: SRF 在 DAVIS2017 数据集上的部分失败例

3.3.4 超参数敏感性分析

在 SRF 的跟踪器切换机制中，当全局重检测器连续 K 帧都检测到目标后，方法会重新唤醒局部跟踪器，并从下一帧开始指定局部跟踪器基于运动连续性在局部范围内更快地搜索目标。这里，本文分析了全局重检测器连续检测帧数 K 对跟踪结果的影响。本文尝试了不同的 K 的取值，并将结果罗列在表3-9中。结果显示 $K = 2$ 时性能达到最优。在局部跟踪场景中，SuperDiMP[11]、TrDiMP[66] 等判别式跟踪器比 SiamR-CNN[27] 等全局方法表现更好。因此，如果全局重检测器重新发现目标，并且其输出的置信度分数很高时，更迅速地将检测任务交给局部跟踪器来完成有助于取得更准确的预测结果，这可能是较大的 K 导致 F-分数下降的原因。另一方面，相比于检测到一帧就交给局部跟踪器，连续两帧检测到目标可以更好地保证目标不是偶然发现的。

表 3-9: 全局重检测器连续检测帧数 K 对跟踪结果的影响

K	1	2	3	4	6
F-分数	0.7082	**0.7092**	0.7088	0.7082	0.7073

本文还分析了全局重检测器置信度阈值 θ 对跟踪结果的影响。在 SRF 的全局重检测器中，只有当检测结果的置信度分数超过一定阈值时，该结果才被认为是可靠的，并将这一帧计入到前述的 K 中。本文用 θ 表示该阈值。若阈值较小，则即使预测不准确，方法也会认为检测到了目标，这样做有可能把错误的检测结果也纳入到计数中；阈值过高又会延迟方法将跟踪任务从全局重检测器交回给局部跟踪器的时间，而不能充分利用局部跟踪器产生更优的预测结果。因此，需要在这两个影响条件中寻找一个平衡。本文尝试了四个不同的 θ 的取值，并将结果呈现在表3-10中。结果显示当 $\theta = 0.7$ 时 SRF 的性能最佳。

表 3-10: 全局重检测器置信度阈值 θ 对跟踪结果的影响

θ	0.5	0.6	0.7	0.8
F-分数	0.7074	0.7088	**0.7092**	0.7056

3.3.5 速度分析

在真实场景中部署机器学习模型时，速度是非常关键的考察因素。SCP 是本文跟踪器中的速度控制参数。本文以 0.25 为间距，尝试了五个 SCP 的取值，

每一个取值大小及相应的结果见表3-11。当参数等于 1 时，方法的运行速度最快，超过了 21 帧每秒（frame per second, FPS），而整个跟踪器仍保持了相当的准确率。

表 3-11: 在 VOT2019-LT 数据集上测试不同速度控制参数下 SRF 的表现

速度控制参数（SCP）	0	0.25	0.5	0.75	1
帧每秒（FPS）	14.6	15.4	16.8	18.3	21.3
F-分数	0.708	0.707	0.706	0.703	0.697

如公式 (3-11) 所示，对于局部跟踪器跟踪效果不佳的帧，SCP 越接近 1，其中被送入全局重检测器重新检测的帧越少。此时，更多的帧会直接将欠佳的局部跟踪器的输出而非全局重检测器的输出输入至结果精炼模块，这会降低方法的预测准确度。本文建议将 SCP 的初始值设置为 0，以充分发挥 SRF 各个组件的跟踪能力。该结果也说明了全局重检测器的重要性。

在表3-12中，本文从运行速度这一角度比较了 SRF 与其他代表性的长时跟踪器。SRF 记录的两个速度分别对应速度控制参数为 0 和 1 的情况。尽管本文并没有在最先进的 GPU 上运行本文的模型，但 SRF 仍然是所有方法中最快的。据本文所知，SRF 是第一个可以连续地调节速度的长时跟踪器。

表 3-12: SRF 与现有方法的运行速度比较

跟踪器	LTMU [21]	Global Track [71]	Siam R-CNN [27]	Keep-Track [25]	Xuan 等人的方法 [57]	SRF 本文方法
帧每秒（FPS）	13	6	4.7	12.7	3.8	**14.6 (21.3)**
设备	2080Ti	TitanX	V100	2080Ti	2080Ti	TitanXp

3.4 小结

这一章节介绍了一个新的面向真实场景的长时目标跟踪与分割算法 SRF，其由一个局部跟踪器、一个全局重检测器和一个结果精炼模块组成。简洁是本文方法相比于其他方法的一个特点。本文的实验结果表明，在长时跟踪器中增加新的模块并不一定意味着性能的提高。本文并没有设计新的神经网络作为跟踪器的一部分。相反，本文从互补的角度充分挖掘了现有各类跟踪器的潜力。在七个长时跟踪数据集上的实验结果证明，组合现有的研究工作，也可以达到和

基于新设计的神经网络的方法相当的性能。本文还将本文的跟踪器延伸到视频目标分割任务上，并证明半监督条件的视频目标分割任务可以基于现有的视频目标跟踪器来完成。得益于结果精炼模块，SRF 基于包围框的输入输出像素级的物体位置和物体轮廓。在两个有代表性的视频目标分割数据集上，SRF 都在推理速度和预测准确性之间取得了良好的平衡。SRF 也是第一个可以连续地调节速度的长时跟踪器。在最快情况下，跟踪速度可以超过 20 帧每秒，且仅伴随很少的精度损失。

第四章　基于跟踪算法网络结构的智能标注算法

在真实场景下的视频目标跟踪任务中，跟踪器需要跟踪之前从未见过的物体类别。在 LaSOTExtSub 等数据集上的实验结果也表明，现在的跟踪算法已经能够很好地处理这类问题。本文分析认为孪生网络结构在其中发挥了非常重要的作用，它可以帮助模型仅根据跟踪目标在第一帧的图像模版在后续帧中找到与之最相似的物体。而在标注任务中，标注工作者必然要标注之前从未见过的物体。那么跟踪算法的网络结构是不是也可以迁移到智能标注领域来使用呢? 本文带着这样的设想，试图设计一个基于孪生网络的机器辅助标注算法，让算法在面对标注任务中的物体类别变化或背景变化时表现出更强的泛化能力。本文将模型命名为 SiamAnno，以体现其是基于孪生网络（Siam）的辅助标注（Anno）模型。

虽然 SiamAnno 本质上是一个实例分割模型，但其输入并不是整张图像，而是从图像中裁剪出的待标注物体所在的小范围区域。SiamAnno 不仅在训练集和测试集同分布的情况下表现出有竞争力的标注精度，在跨域的标注任务中也显示出巨大的潜力。在简单描述本文方法的应用场景设想后，本文将从算法设计、模型结构和实现细节等角度详细介绍本文的模型。在多个数据集上的实验也证明了本文方法的有效性。

4.1　智能标注任务描述

本文设想的在图像中标注某个物体的流程如图4-1所示。其中，SiamAnno 是本文设计的标注模型，其有两个输入分支，本文将在下文做详细介绍。该标注模型本质上是一个实例分割网络，它可以将用户输入的物体包围框转化为由一系列顶点组成的物体轮廓。工具既允许用户通过点击和拖动鼠标的方式来绘制包围框，也可以直接接受现有图片数据集已有的包围框标注作为输入。模型的输入是以用户的包围框为中心，以一定比例扩展其大小，从原图中裁剪出的一块

子区域。该区域会被送入模型，以预测物体的轮廓。当工具将模型预测的物体轮廓绘制到界面后，轮廓上的顶点应是可以直接被用户拖动的，这样使用者可以轻松地修改完善该标注结果。标注模型是一个机器学习模型，因此需要使用一个数据集对其训练。如果使用该工具时的数据分布与训练集的分布相同，如使用了同一数据集的训练集和验证集，这种情况被定义为“域内”的标注任务（in-domain annotation）。如果待标注的数据集是一个全新的数据集，如包含与训练集不同的物体类别或图像拍摄环境，类似情况将被定义为“跨域”（cross-domain annotation）的标注任务。本文提出的 SiamAnno 模型能够在不重新训练或微调网络的情况下处理这种“零样本”（zero-shot）的情况。在章节4.3.3，本文将会使用多个数据集来模拟跨域标注的场景，验证本文方法的有效性。本文还将模型的输出用于目标跟踪任务的训练，以证明本文的方法对下游计算机视觉任务是有帮助的。

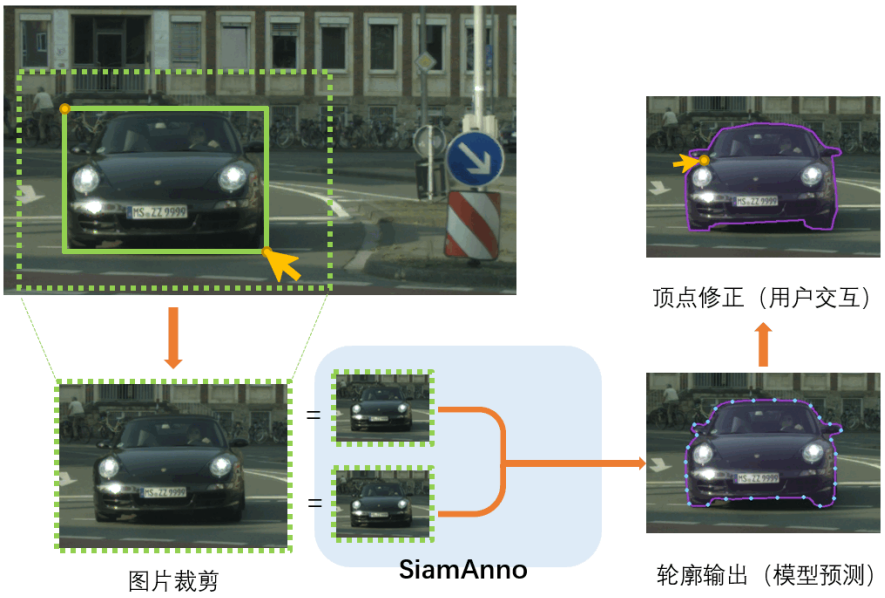

图 4-1: 智能标注模型 SiamAnno 的使用流程

4.2 算法设计

本文将预测任一物体的轮廓这一问题建模成一个相似性学习任务，使用孪生网络结构来完成。本章将从孪生网络、像素级相关性运算以及 U-net 风格的特征融合机制这三个角度对算法设计展开介绍。之后，本章会重点介绍其中的特征预测头部分。

本文已在章节2.1介绍了孪生网络的基本结构，其包括两个输入分支和一个特征融合模块。在机器辅助标注任务中，本文在其结构后面增加一个轮廓预测头，其在特征融合模块输出的相关图的基础上预测每一个顶点的位置。本文还引入了 U-net 风格的特征融合机制，充分利用骨干网络中的低层特征，以得到更准确的顶点位置回归结果。

4.2.1 基于孪生网络的特征提取

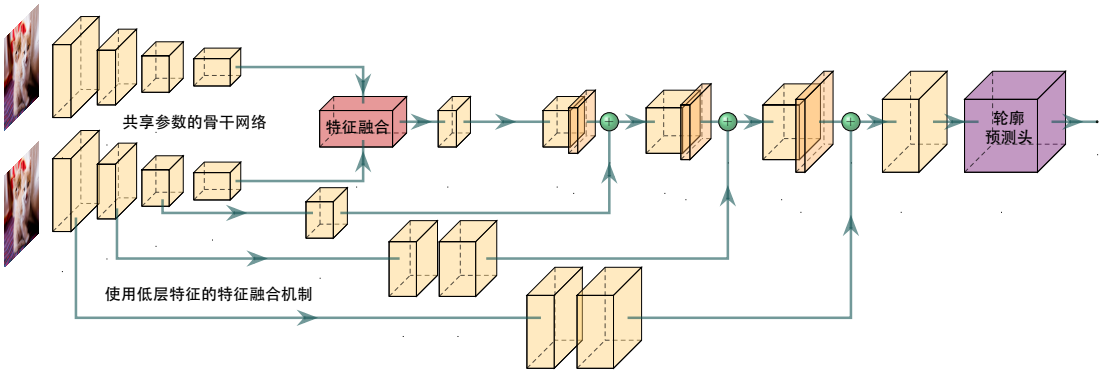

图 4-2: SiamAnno 模型结构图

SiamAnno 模型有两个输入分支，如图4-2所示。在标注任务中，两个分支被分别命名为目标分支和搜索分支。两分支均在整张图像中以待标注物体为中心裁剪出一个子区域作为输入。为了表述方便，本文定义决定该裁剪范围的“搜索范围”（search scale）参数 s，其表示的是裁剪区域相对于真实包围框大小的倍数关系。两个分支使用相同结构和参数的骨干网络来提取裁剪区域的特征，再由相关性运算模块负责融合特征。值得一提的是，尽管在裁剪图片时两分支使用的是相同的搜索范围参数，也即输入到两分支的图片范围是相同的，但两个分支的特征是以不同的大小被送入相关性运算模块的。对于目标分支，方法只使用特征图的中心部分。该部分的边长为原始特征图的 $1/s$。这样仅保留待标注物体本身的特征，而舍弃掉其周围的背景特征，让目标分支专注于目标本身。而对于搜索分支，方法不进行任何的二次裁剪，这样搜索分支可以包含更多的背景特征。本文使用 Resnet-34[103] 作为孪生网络两分支的骨干网络。两分支输出的特征将会被送入后续的相关性运算操作，以产生融合特征图。

4.2.2 像素级相关性运算

现有的孪生网络通常选择朴素的相关性运算（naive correlation）或深度相关性运算（depth-wise correlation）作为特征相关性计算模块。然而，目标跟踪领域的相关研究[55] 表明，这种将整个目标特征作为核函数来和搜索特征进行相关性运算的做法会模糊物体本身的形状信息，输出不清晰的相关性响应图（correlation responses）。在大多数视频目标跟踪方法中，搜索区域的大小是目标大小的四倍（$s = 4$）。而在本文的应用场景中，搜索范围参数的取值在 1 到 2 之间（$s \in (1, 2)$）。因此，本文模仿 AlphaRefine 方法[55]，使用像素级相关作为特征融合模块。这更适合孪生网络两分支的输入大小差异不大的情况，可以在特征融合时更好地维持物体的空间信息。

像素级相关性运算将每个像素作为一个核。本文用 $T \in \mathbb{R}^{C \times H_t \times W_t}$ 表示目标分支输出的特征，用 $S \in \mathbb{R}^{C \times H_s \times W_s}$ 表示搜索分支输出的特征。像素级相关性运算将 T 分解为 $H_t W_t$ 个核，每一个核表示为 $k_i \in \mathbb{R}^{C \times 1 \times 1}$。用每一个核按照朴素的相关性计算方法与 S 进行计算，得到整张图像的相关图 $C \in \mathbb{R}^{H_t W_t \times H_s W_s}$，如公式 (4–1) 所示。公式中 $\odot$ 表示朴素的相关性运算。

$$C = \{C_i | C_i = k_i \odot S\}_{i \in \{1, \ldots, H_t W_t\}}, \tag{4–1}$$

4.2.3 U-net 风格的特征融合机制

视频目标跟踪模型的研究工作[12,15,63] 常结合使用多尺度的特征图来提高跟踪的准确性和稳健性。鉴于本文模型也使用了孪生网络，我们也尝试在方法中使用多层特征图。本文引入了 U-net 风格的特征融合机制，在从骨干网络提取不同层次特征的同时，放大特征融合模块输出的相关图，帮助后续的轮廓预测头输出更准确的结果。为了表述的方便，本文将融合一次特征图定义为一步。本文使用从搜索分支的 Resnet-34 骨干网络中提取的 *conv*3、*conv*2 和 *conv*1 三层特征图。在第 j 步，来自上一步的融合特征图 M_{j-1} 首先被以内插（interpolate）的插值方式扩大一倍，再与来自骨干网络的特征图 C_j 相加。由此产生的特征图再通过卷积层和 ReLU 函数，送入下一步。

$$M_j = f_j(\text{Interpolate}(M_{j-1}) + C_j). \tag{4-2}$$

公式 (4-2) 表达了一个融合步，其中卷积运算被表示为 f_j（$j = 1, 2, 3$）。初始的 M_0 由前述的特征融合模块输出的相关图通过卷积层得到。需要说明的是，C_j 中的下标 j 不是 Resnet 中定义的 *conv* 的层数。j 与该数字的顺序恰恰相反。例如，C_1 是 *conv*3。融合多层特征有助于更精确地预测边界，因为层次靠后的特征具有高层次的语义信息，而层次靠前的特征包含颜色、形状等细节信息。将这些特征以一种从粗到细的方式进行融合，扩展出的特征图具有多层次的信息，可以帮助轮廓预测头以更高的分辨率估计轮廓上各个顶点的位置。

4.2.4 轮廓预测头

本文将从包围框预测物体轮廓的过程建模成顶点的收缩过程，这就需要轮廓预测头预测每一个顶点所需要的偏移量。近来的相关工作[104-105] 用神经网络表达经典的蛇形算法（snake algorithm）的思想，在实例分割任务上取得优异表现。这类方法的轮廓预测网络与本文的孪生网络结构有很好的兼容性，因此本文将其作为轮廓预测头。与原方法相比，本文方法在实现细节上有一定区别，本文将在下一小节阐述。

前述的 U-net 风格的特征融合机制产生的特征图会被送入该轮廓预测头。预测头首先将包围框作为初始边界，在其上采样 K 个顶点。从特征图上提取每个顶点的特征，在所有顶点上应用环形卷积（circular convolution），输出每个点收缩到物体真实边界所需要的偏移量。这样一个过程被称为运行一次“深度蛇形算法”，该过程可以循环进行。例如，刚刚得到的所有点又组成了一个新的“初始”边界，可以再次重复上述操作，进行迭代优化。在后续的迭代中，本文使用了基于注意力的变形机制（attentive deformation mechanism)[105]，让模型更加关注在之前的迭代中预测不准确的点。该机制自适应地调节原有结果和新预测结果的重要性关系来在新的预测量和原结果间做出取舍。如果深度蛇形算法的次数较少，模型预测的轮廓边界可能不准确；若运行的次数过多，又会增加过拟合的风险。因此，需要确定一个较优的深度蛇形算法迭代次数。本文将该次数记为 D，本文将在章节4.3.6探讨该取值的最优值。

在深度蛇形算法中，为了更加准确地预测每个顶点的位置，需要寻找有效利用环上相邻顶点之间关系的方法。一般而言，轮廓上的顶点沿着物体边缘自然地形成一个环。一些交互式标注方法[34-35]使用图卷积网络来预测每个顶点的偏移量。然而，图卷积网络中的池化操作会损失信息。相比之下，循环卷积[104]是一个更优的选择，其直接在每个顶点对其自身和相邻顶点的特征应用一维卷积。对任意顶点 p，公式 (4-3) 定义了对顶点 p 的循环卷积，其中 f 是特征图，k' 是待学习的卷积核，$\circ$ 是标准的一维卷积。

$$(f \circ k')_p = \sum_{r=-R}^{R} f_{p+r} k'_r \tag{4-3}$$

公式中，R 是卷积核的大小，也是一个顶点所考虑的邻居的数量。本文还可以引入空洞率（dilated rate），使之成为空洞循环卷积。串联起多个有不同空洞率的循环卷积可以利用多尺度的物体边缘信息，让方法不仅关注距离每一个顶点最近的邻居，同时也考虑距离较远的顶点的特征。

通过上述多层循环卷积后，特征将会被送入四个 1×1 的卷积层，再经过一个 tanh 激活函数，产生最终的顶点偏移估计。需要注意的是，公式 (4-3) 中的特征图不仅包含神经网络学习得到的特征，也包括顶点坐标本身。这样可以直接将坐标的位置信息包含到特征中。但是物体越大，顶点所需偏移也往往越大。如果直接让模型预测原始物体大小尺度下所需要的顶点偏移量，不同大小的物体所需要的偏移量不同，不利于稳定训练整个模型。因此，本文对物体坐标进行了归一化操作，将原始坐标转换为相对坐标，避免物体大小这一因素扰动模型的训练过程。对于一个物体，本文找到该物体在原始尺度下坐标的极端取值（即横纵坐标的最大最小值），再用类似 min-max 归一化的方法来规范每个坐标。相应地，模型输出的也是在 [0, 1] 范围内的相对偏移。该结果需要再乘以物体包围框的大小，来得到与原图尺度吻合的顶点位置。

4.2.5 实现细节

SiamAnno 的训练使用到两个损失函数。Smooth L_1 损失函数用于训练顶点的偏移输出，如公式 (4-4) 所示，公式中 N 是训练样本的数量，x_n 和 $\tilde{x}_n$ 分别是

真实和预测的顶点位置，W 是图片大小。

$$L_{snake} = \frac{1}{N}\sum_{n=1}^{N} smooth\_L_1\left(\frac{\tilde{x}_n}{W} - \frac{x_n}{W}\right). \tag{4-4}$$

计算损失函数需要将模型输出和学习目标一一对应。在标注任务中，需要对每一个预测的顶点都找到相应的真实轮廓上的顶点。本文使用了 DANCE[105] 方法中的分段匹配方案。具体来讲，本文用包围框和物体真实轮廓的交点将物体轮廓拆分成多个分段，在每一个分段内进行匹配。这样可以缓解其他方法中会产生的交错匹配现象（correspondence interlacing phenomenon），让模型的学习过程更加稳定。对该分段匹配方案感兴趣的读者可以阅读 DANCE 方法的原文来了解细节。此外，本文使用 Dice loss[106] 来训练基于注意力的变形机制。最后，本文按照公式 (4-5)，将 Dice 损失 L_{Dice} 和回归损失 L_{snake} 以一定比例 α 结合起来。本文默认设定 $K = 196$，$\alpha = 10$。

$$L = L_{Dice} + \alpha L_{snake} \tag{4-5}$$

对于一个训练数据集，本文对其中的每一个实例都在它所在的图像上以实例为中心按照前述的搜索范围参数将其裁剪出来，同时送入目标分支和搜索分支。两个分支的输入都会被调整为 256×256。SiamAnno 共需要被训练 100 轮（epoch），每一轮包含 4000 次迭代（iteration），每一轮迭代在每一张 GPU 上的批大小（batch size）为 8。本文同时使用 4 张 GPU 对其进行训练，实际批大小为 $4 \times 8 = 32$。所有网络参数均纳入训练，不冻结任何部分。本文使用 Adam 优化器来训练，设定初始学习率为 $5e^{-4}$，在第 40 轮和第 80 轮将学习率减半。

亮度调整（brightness adjustment）和灰度转换（gray scale transformation）被用于训练时的数据增强。孪生网络有两个输入分支，数据增强方法可以选择以相同的具体实现同时应用于这两个分支，也可选择分别应用。本文选择对每个输入图像单独应用亮度调整，而对属于同一对输入的两张图像同时应用灰度转换。

4.3 实验与分析

本文将在这一章节对 SiamAnno 展开全面的实验，以验证本文方法在域内和跨域标注任务中的有效性。既有的研究工作往往只关注方法在域内标注任务上的性能。一个现象是，相关论文常使用多个评价指标来展示方法在域内标注任务上的实验结果，但对于跨域标注情景下的实验，则只展示平均交并比一个指标的结果。本文认为，在标注任务中标注之前从未见到的物体类别是非常常见的，研究也应该更加充分评估模拟跨域标注的实验的结果。因此，针对跨域的标注任务，本文在平均交并比的基础上，同时报告了平均精度和边界 F 分数（boundary F-score）的结果。尽管没有其他研究在跨域标注任务实验中展示这两个指标结果，暂时无法用这两个指标来比较本文方法和既有研究，但本文希望结果可以为后续的研究工作提供一个可比的基线（baseline），来促进跨域标注相关研究的发展。

4.3.1 评价指标

交并比是机器辅助标注任务中最为常见的评价指标。为了保持不同方法间的可比性，本文沿用既有研究中计算平均交并比（mean IoU, mIoU）的方式：首先计算每一个实例的交并比，再计算每一个物体类别中所有实例交并比的平均值，再在此基础上计算所有类别的平均交并比的平均值。值得注意的是，本文汇报的是经过两次平均后的平均值，而不是直接对所有实例的交并比计算平均的结果。

基于交并比，可以计算平均精度（average precision, AP），其也是实例分割领域中被广泛使用的评价指标。但与平均交并比的计算不同，平均精度的计算并不考虑物体的所属类别，而是直接考察所有实例的预测准确度。每指定一个交并比阈值，就可以计算一个平均精度值。通过将该阈值从 0.5 以 0.05 为间隔增加到 0.95，可以得到一系列 AP 值，再计算它们的平均值并报告。该平均值在许多实例分割文献中通常被表示为 mAP@(0.5:0.95)。

平均交并比和平均精度两个指标都是通过比较预测结果和真实标注之间的面积差异来计算的，并未关注轮廓预测的准确性。边界 F 分数则基于形态学运算符（morphology operators）比较预测轮廓上的像素点和真实轮廓点，计算轮廓

点预测的命中率、失误率和假阳性率，在此基础上计算查准率和查全率。可以再指定一定的可容忍错误像素距离，来允许有较小的定位误差。关于该指标的更多介绍可以参考相关论文[107]。本文遵循既有研究的做法，报告允许 1 个或 2 个像素的误差的情况下的边界 F 分数，分别记为 F_{1px} 和 F_{2px}。

4.3.2 “域内”标注任务的实验

本文默认使用 Cityscapes[108] 的训练集对 SiamAnno 模型进行训练，因此首先使用该数据集来测试模型在域内标注任务中的性能。该数据集中的图片是 27 个欧洲城市的街道风景。其包含 2975 张训练图像、500 张验证图像和 1525 张测试图像。由于本文没有测试图像的真实标注注释，没有办法确定测试集中每一个实例的包围框，也就没有办法将他们作为模型的输入。因此，本文按照既有研究中的实现方式[28,34-35,93]，报告方法在验证集上测试的结果。

该数据集共包含 8 个物体类别，这八个类别的物体大小差异很大。既有研究往往首先报告每一个类别的平均交并比，再计算这些值的平均值。本文在表4-1中沿用了这种结果展示方法，便于与其他方法进行比较。表4-2中还比较了不同方法的平均精度和边缘 F 分数。SiamAnno 的平均精度为 39.6%，领先之前最好方法十个百分点。本文方法在平均交并比和边缘 F 分数这两个评价指标上的表现也很有竞争力。例如，本文方法非常擅长标注火车这类物体。

表 4-1: 在 Cityscapes 验证集上域内标注的逐类结果比较（mIoU）

方法	自行车	巴士	路人	火车	卡车	摩托车	乘用车	骑行者	**平均交并比**
Polygon-RNN	52.1	69.5	63.9	53.7	68.0	52.1	71.2	60.6	61.4
Polygon-RNN++	63.1	81.4	72.4	64.3	78.9	62.0	79.1	69.9	71.4
DACN	64.6	82.6	72.9	61.3	80.5	63.9	80.3	71.3	72.2
Polygon-GCN	64.6	85.0	72.9	61.0	79.8	63.9	81.1	71.0	72.7
PSP-DeepLab	67.2	83.8	72.6	68.8	80.5	65.9	80.5	70.0	73.7
Spline-GCN	67.4	85.4	73.7	64.4	80.2	64.9	81.9	71.7	73.7
DELSE	67.2	83.4	73.1	69.1	80.7	65.3	81.1	70.9	73.8
SiamAnno	63.9	80.6	72.1	70.3	80.1	64.0	79.4	68.2	72.3
SiamAnno†	*69.1*	*85.3*	*75.4*	*77.8*	*82.5*	*69.8*	*82.7*	*71.0*	*76.7*

表 4-2: 在 Cityscapes 验证集上域内标注的结果比较（mAP、F score）

方法	平均精度 mAP	F_{1px}	F_{2px}
DACN	-	45.27	59.89
Polygon-RNN++	25.5	46.57	62.26
PSP-Deeplab	-	47.10	62.82
Spline-GCN	-	47.72	63.64
DELSE	-	48.59	64.45
Split-GCN	29.6	**52.50**	**67.50**
SiamAnno	**39.6**	46.62	60.20
SiamAnno†	***48.5***	*52.43*	*66.68*

本文方法在域内注释任务上并没有取得非常领先的指标结果，本文对此进行了简单的分析。Cityscapes 数据集中有很多实例被其他物体遮挡而被分割成多个部分。本文方法的原理是从初始的包围框不断将边界变形收缩，直至完全包裹物体本身。这样显然不能很好地处理被分为多个部分的物体。本文发现 PolygonRNN++、CurveGCN 等方法也用类似的思路来优化物体轮廓，它们的平均交并比取值也与本文的接近。因此，本文认为这是 SiamAnno 在域内注释任务中没有超过现有最优方法的原因。

然而当用户使用本文的机器辅助标注工具时，他如果发现待标注的物体被分割成多个部分，可以对每个部分进行单独标注，而不是仅使用一个包围框选中整个感兴趣的实例。为了模拟这种情况，本文也考察了 SiamAnno 对每个部分分别进行标注时的准确度。本文将这种对每个部分分别进行标注的模式称为“组件模式”（component mode），而将原来的对每个物体整体直接标注的模式称为“实例模式”（instance mode）。组件模式的结果展示在表4-1和4-2的最后一行，用 † 将其和其他实例模式下的结果相区分。相比于实例模式，组件模式下所有指标的结果都有很大提高，特别是平均精度，从 39.6% 提升到 48.5%。

4.3.3 “跨域”标注任务的实验

相比于域内标注任务，跨域标注可能是真实场景中更加常见的需求。为了测试 SiamAnno 模型标注从未见过的物体类别的能力和在新的背景下标注物体的能力，本文使用 KITTI、ADE20k 和 Rooftop 三个数据集作为测试集来测试方法的性能。本文仍然使用在 Cityscapes 数据集上训练的模型，而测试使用的三个

数据集和 Cityscapes 的分布均不同，因此这属于跨域标注任务。本文首先介绍这三个用于测试的数据集。

和 Cityscapes 数据集类似，KITTI[43] 也是一个城市路景数据集，但数据量更小，且每张图片中包含的人和车的数量相对较少。该数据集和 Cityscapes 数据集包含的物体类别是重复的，但取景的城市不同，因此该数据集可用来测试模型能否适应环境的变化。为了保持不同方法间的可比性，本文沿袭现有研究[28,34,93]的实验方法，使用该数据集的衍生版本[109]，且只关注其中对汽车的标注。

ADE20k[42] 是一个包含了各类常见场景的视觉分割数据集。该数据集数据质量高，涵盖广泛的拍摄场景和物体类别，具有密集而详细的注释，适合用来测试方法在通用场景下的标注性能。为了保持不同方法间的可比性，本文沿袭现有研究[28,34,94] 的实验方法，只考虑模型在验证集中的电视接收器、巴士、小轿车、烤箱、人和自行车这六类物体的标注结果。

Rooftop[44] 数据集包含 65 幅城郊景色航拍图。和 Cityscapes 数据集相比，其不仅包含完全不同的物体类别，图片拍摄的角度也不同。大多数建筑的屋顶形状都呈现出复杂的多边形，本文用该数据集来测试了模型标注之前从未见过的物体的能力。本文将展示在该数据集的测试集上的实验结果。

表4-3中对比了本文方法和其他六个现有方法[28,34,93-94,110] 在上述三个数据集上的平均交并比结果。SiamAnno 在所有数据集上都超越了既有方法，这证明 SiamAnno 在处理跨域标注任务中的类别变化和环境变化有巨大的潜力。本文还在表4-4中报告了 SiamAnno 在这三个数据集上的平均精度和边缘 F 分数。现有文献不展示自己的方法在这两个评价指标上的表现，因此本文无法在这两个评价维度上做方法间的比较。但本文希望通过展示更全面的性能指标结果，来为未来的研究提供一个基线，推动跨域标注任务相关研究的发展。

表 4-3: 在 KITTI、ADE20k 和 Rooftop 数据集上跨域标注的结果比较（mIoU）

方法	KITTI	ADE20k	Rooftop
Polygon-RNN	74.22	-	-
Polygon-RNN++	83.14	71.82	65.67
PSP-Deeplab	83.35	72.70	57.91
Polygon-GCN	83.66	72.31	66.78
Spline-GCN	84.09	72.94	68.33
DACN	-	73.21	66.92
SiamAnno (本文方法)	**86.41**	**74.90**	**78.04**

表 4-4: 不同的训练集和测试集组合下 SiamAnno 的性能表现

测试数据集	平均交并比 mIoU	平均精度 mAP	F_{1px}	F_{2px}
COCO*	79.85	56.3	59.17	71.37
Cityscapes	72.33	39.6	46.62	60.20
Cityscapes†	76.69	48.9	52.88	67.90
KITTI	86.41	69.3	67.56	81.37
ADE20k	74.90	46.6	58.87	73.27
Rooftop	78.04	49.9	27.76	40.01

* 在 COCO 测试集上的实验是基于在 COCO 训练集上训练的模型完成的。后续的无 * 号标记的其他测试是在基于 Cityscapes 训练集上训练的模型完成的。

4.3.4 将智能标注算法应用于目标跟踪任务中

使用智能标注算法的用户最为关心的一个问题，是该算法能否提高既有模型在其他计算机视觉任务上的性能。现有的计算机视觉模型往往依赖于大量数据的训练。如果将训练中使用的带有包围框标注的训练集先用智能标注算法转换成像素级掩膜，再利用转换后的像素级掩膜训练模型，可以提高模型在诸如准确率、交并比等评价指标上的表现，就可以说明该智能标注算法对于下游的其他计算机视觉任务是有帮助的。本文基于视频目标跟踪任务验证 SiamAnno 的有效性。

本文在 Got10k 数据集[111] 上进行实验。沿袭既有研究的做法，本文使用两个评价指标来评价模型在该数据集上的表现。期望平均交叠（expected average overlap, EAO）可以被简单理解为是每一帧预测框和实际框的交并比的均值，成功率的定义则和其他数据集相同。两个指标都是越高越好。

鉴于这是一个短时跟踪任务，本节仅使用 SRF 中的局部跟踪器和结果精炼模块。本节的实验聚焦在调整结果精炼模块的训练数据集构成。需要说明的是，结果精炼模块中包含预测像素掩膜的分支，因此本身就需要已经包含了像素级标注的数据集对其进行训练。本节首先使用 Got10k、LaSOT 和 Youtube-VOS 这三个数据集的训练集对结果精炼模块进行训练，其中前两个数据集包含的是包围框标注，Youtube-VOS 数据集同时包含包围框和像素掩膜标注。本节再利用 SiamAnno 对 Got10k 的训练集生成像素掩膜预测，并不额外引入人工来修改预测结果。本节用前述的三个数据集重新训练结果精炼模块，但此时会利用上机器预测的 Got10k 的像素级标注信息。本节对比这两种训练策略下 SRF 在 Got10k

测试集运行的性能指标，如表4-5所示。本节发现，将训练集中原本只有包围框标注的部分数据转换成像素级标注可以提升模型性能。在实验中，EAO 和成功率这两个指标均有一定程度的提高。这说明本文的 SiamAnno 智能标注算法对现实计算机视觉任务是有帮助的。即使不增加新的训练数据，不依赖人工对不完美的机器标注结果进行修改，仅仅是在训练集中将机器生成的像素级标注结果替换掉原有的包围框结果，也有助于提升模型的表现。

表 4-5: 在 Got10k 测试集上 SRF 结合智能标注算法完成目标跟踪任务的结果比较

用包围框训练	用像素掩膜训练	EAO	成功率
Got10k LaSOT	Youtube-VOS	84.13	93.44
LaSOT	**Got10k** Youtube-VOS	**84.47**	**94.05**

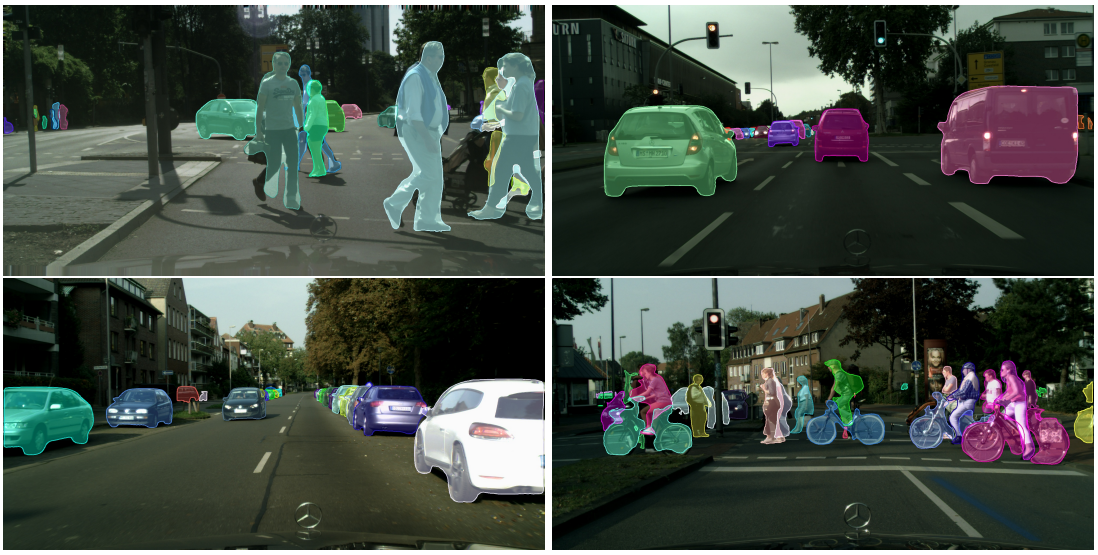

图 4-3: SiamAnno 在域内标注任务中的边缘预测结果

4.3.5 结果可视化

在这一小节，本文将会从多个角度展示 SiamAnno 的标注结果。图4-3以图像为单位展示了 Cityscapes 数据集下的标注结果。这是一个域内标注场景，图中绘制的是组件模式的结果。虽然不完美，但 SiamAnno 输出的结果可以达到令人满意的程度。图4-4更加详细地以每一个实例为单位展示了标注结果。SiamAnno 将会为每一个实例预测 196 个顶点。为了美观，本文在图中仅展示了一部分顶点

的预测结果。SiamAnno 更擅长标注汽车，而在标注人的能力上还欠佳，这可能是因为人常常有不规则的形状或运动，标注时容易受到相似物体或背景的干扰。在图4-5中，本文以一个被分割的对象为例，对比了实例模式和组件模式下模型的运行结果。SiamAnno 从初始包围框不断收缩至物体边缘的方法原理令其不擅长处理被分为多个部分的目标。图4-6展示了在跨域注解任务中 SiamAnno 的标注结果。图中顶部一组图像选自 KITTI 数据集，底部两组图像分别选自 ADE20k 和 Rooftop 数据集。本文并没有利用这些数据集重新训练或微调模型，而是仍然沿用在 Cityscapes 上训练的模型。可以看出，即使是面对拍摄环境或物体类别的变化，SiamAnno 在分布不同的三个数据集上都产生了高质量的轮廓预测结果。

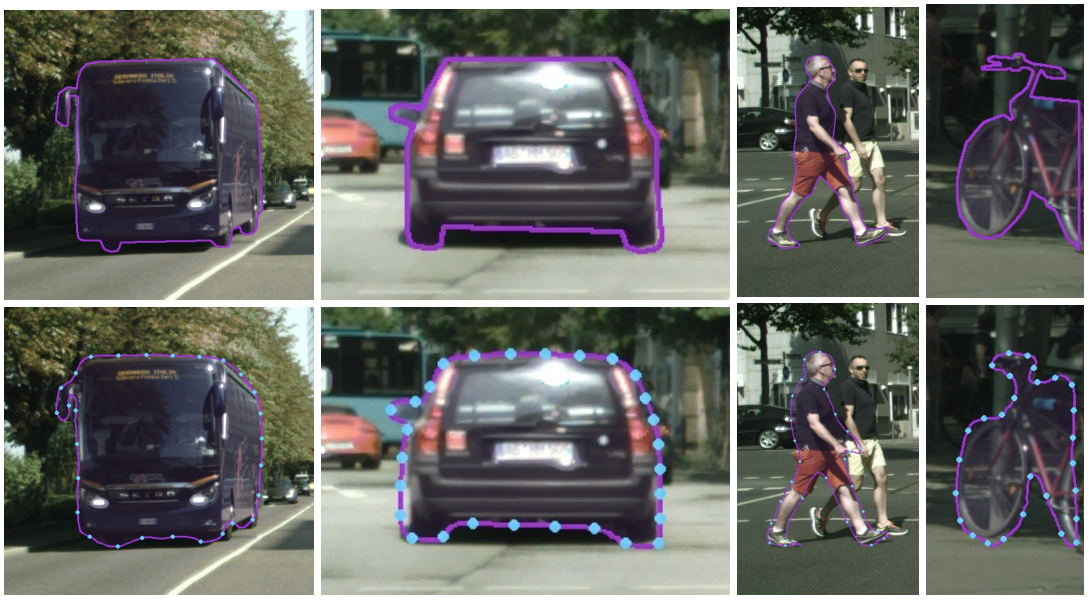

图 4-4: Cityscapes 数据集中实例的边缘预测结果与真实情况的对比

图 4-5: SiamAnno 组件模式（左）和实例模式（右）的预测效果比较

4.3.6 超参数敏感性分析

本节首先分析了搜索范围参数 s 对标注准确度的影响。在章节4.2.1的基于孪生网络的特征提取中，本文定义了搜索范围参数 s，用来决定对搜索分支特征图的裁剪大小。较小的 s 意味着特征只被保留较小一块区域，让模型更加关注

图 4-6: SiamAnno 在跨域标注任务中的边缘预测结果。

表 4-6: 搜索范围参数 s 对标注准确度的影响

s	平均交并比 mIoU	平均精度 mAP	F_{1px}	F_{2px}
1.0	75.29	44.8	52.03	65.89
1.2	76.03	47.0	51.97	66.55
1.4	**76.69**	**48.5**	**52.43**	**66.68**
1.6	76.29	48.1	52.22	66.29
1.8	76.43	48.0	51.38	65.46
2.0	75.89	47.1	51.18	65.21

目标本身的细节特征；较大的 s 则允许保留更多的背景特征，可能有利于从背景中区分目标。这两个影响因素相互牵制，需要寻找到一个可以平衡这两个因素的参数取值。为此，本文尝试了不同的搜索范围参数。表4-6中的结果表明当 $s = 1.4$ 时模型的性能最佳。在本文的其他实验中，默认设置 $s = 1.4$。

在章节4.2.4介绍的轮廓预测头中，本文定义了深度蛇形算法迭代次数 D。

每一次运行该算法，模型都会在上一轮预测的物体轮廓的基础上再次预测每一个顶点所需要的偏移量。本文分析了深度蛇形算法迭代次数 D 对标注准确度的影响。如果深度蛇形算法的次数较少，模型预测的轮廓边界可能不准确；若运行的次数过多，又会增加模型过拟合的风险。因此，需要确定一个较优的深度蛇形算法迭代次数。本节测试了当 D 取值范围是 1 至 4 的情况下方法的预测准确度，如表4-7所示。结果表明 $D = 3$ 或 4 是效果最优。考虑到计算复杂度的区别，在本文的其他实验中，本节默认设置 $D = 3$。

表 4-7: 深度蛇形算法迭代次数 D 对标注准确度的影响

D	平均交并比 mIoU	平均精度 mAP	F_{1px}	F_{2px}
1	75.67	46.2	51.73	65.91
2	76.11	47.1	52.33	66.32
3	**76.69**	**48.5**	52.43	66.68
4	76.55	**48.5**	**52.44**	**66.76**

4.4 小结

在这一章节，本文借鉴了孪生网络在视频单目标跟踪任务中的优异表现，将跟踪算法的网络结构迁移到机器辅助标注任务中，提出了一个基于跟踪算法网络结构的智能标注算法 SiamAnno。它结合了孪生网络出色的单样本学习能力和深度蛇形算法优异的轮廓预测能力。在学术研究中常见的域内标注任务上，SiamAnno 有着和现有方法相近的标注性能。在更符合真实场景需要的跨域标注任务上，SiamAnno 的准确度超越了既有的所有方法。在多个数据集上的实验表明 SiamAnno 可以很好的应对不同的拍摄环境和新出现的物体类别。本文通过实验证明，在不增加新的训练数据集的情况下，SiamAnno 还可以帮助提高现有单目标跟踪模型的跟踪精度。SiamAnno 是一个“取之于跟踪，用之于跟踪”的模型。

第五章　面向跟踪任务的图片标注工具系统

为了验证本文所提出的算法在实际图片标注场景中的有效性，本章搭建了面向单目标视频跟踪任务的图片标注工具系统，并将本文所提出的方法应用其中。单目标视频跟踪任务中有两类标注场景。一类是为跟踪模型提供更高质量的训练数据。这种情况需要在视频的每一帧都给出跟踪目标具体的包围框乃至像素掩膜的标注。另一类针对跟踪模型的测试，需要对每个视频的第一帧图像给出包围框标注。本章的系统可以满足这两类标注需求，也可满足目标检测、实例分割等其他常见的视觉任务数据集标注需求。本章节将对该系统的开发背景、用户需求、整体架构、实现细节、具体功能及识别效果等进行详细介绍。

5.1　相关背景

数据是一个机器学习任务中非常重要的组成部分。特别是对于深度学习模型，往往训练数据越多，训练后的模型性能越好。尽管在学术界，许多研究人员尝试从半监督学习、无监督学习、少样本学习等角度来降低训练机器学习模型时对数据和人工标注（监督信息）的依赖，但这些方法在真实场景中的落地应用较少。在现在的计算机视觉相关任务中，仍然是监督学习方法的实际应用效果最好。顾名思义，监督学习需要训练数据集中包含监督信息来指导模型学习，这些监督信息需要人工引入。学术研究往往为了保持不同方法间的可比性，直接使用既有的通用数据集来训练和测试模型，不会特别关注数据获取这一步。但在工业应用中，常需要对特定的场景重新采集并标注数据集。而且人们发现在这些现实任务中，相比于对模型结构进行修改，增加训练数据量对模型的提升效果往往更加明显。这也就意味着，如果想要在一个工业应用场景中使用一个深度计算机视觉模型，需要标注大量的视觉数据，且标的越多，用这些数据训练得到的模型效果越好。

在视觉任务中，一类非常常见的监督信息就是对物体位置和类别的标注，其

中物体位置常常以一个四周与坐标轴平行的矩形包围框给出。视频目标跟踪任务中的数据集也是以这样的形式给出监督信息。但随着如自动驾驶、虚拟现实等真实场景应用的发展，人们发现单纯用包围框来定位一个物体已经无法满足现实任务的需要。人们希望用多边形或像素掩膜来精确地表示画面中每一个物体的位置和轮廓，这就要求模型输出像素级的结果。在监督学习中，这也就意味着人需要对数据集中的每一张图像的每一个像素点指定其所属的物体类别，并且在模型训练时使用该像素级监督信息。相比于包围框，标注多边形或像素级结果的成本是非常高的。有没有什么方法既能够得到精确的像素级标注，又可以减轻标注人员的工作负担，提高标注效率呢?

在学术界，研究者们尝试设计不同的神经网络模型来降低标注工作量。这些模型在包围框、关键点点击、潦草笔迹等用户输入的基础上，预测待标注物体的轮廓或像素掩膜。有些模型还接受用户的修改，作为神经网络的二次输入，来微调预测结果。本文在章节2.4中介绍了部分此类方法，本文设计的 SiamAnno 也是这类方法之一。这些方法多停留在学术研究上，尽管开源了相关代码，但并没有提供配套的工具供用户直接使用。为了让学术上的研究成果更快更方便地应用在实际工业任务中，本文以 SiamAnno 为核心开发了一套开源标注工具。这套工具是在现有开源标注工具的基础上，保留了原有的包围框和物体类别标注功能，并添加了利用神经网络从包围框预测物体轮廓的功能，使得该工具可以降低标注不规则物体的具体轮廓所需要的工作量。特别地，本文还设计了连续标注的功能，利用机器学习模型基于上一帧的标注结果预测目标在下一帧的标注，试图进一步降低标注人员的工作量，令整个系统是一个面向跟踪任务的图片标注工具。相比于现有的标注工具，本文工具的优势主要有以下四点:

- 工具不仅允许用户直接在导入的图像上面通过拖拽鼠标的方式绘制包围框，也可以直接读取既存的包围框标注文件，以此为基础预测物体轮廓，实现标注格式转化的功能。
- 用户可以通过直接修改预测轮廓上的顶点来修改模型输出的结果，交互友好，不会出现其他方法中可能发生的在远离用户操作的区域发生意料之外结果变化的情况。
- 针对单目标视频跟踪任务定制了连续标注功能。在逐帧标注视频的场景中，系统可以调用现有跟踪模型预测标注目标在下一帧的位置，不需要用户在每

一帧都从零开始绘制包围框。

- 以网页的方式呈现标注工具，用户打开浏览器即可使用，不需要用户在本地配置运行环境。工具即开即用，对操作环境几乎无任何要求。

5.2 系统设计

本文结合在上一章介绍的 SiamAnno 机器辅助标注模型，开发了一套图片标注工具，并为单目标视频跟踪数据集的标注需求定制了连续标注功能。在具体开发之前，需要首先明确用户需求，再从用户需求出发确定系统的架构设计。

5.2.1 系统需求设计

本系统的主要目的是利用机器学习模型帮助标注人员标注新的单目标视频跟踪数据集，因此需要一整套包括视频帧上传、标注输入、轮廓预测、结果修改、连续标注和标注下载的完整流程。本节将这样一个流程细分为以下需求：

- 视频帧的上传：考虑到现有目标跟踪数据集多以逐帧图片的形式存在，因此系统需要支持用户从本地将待标注的逐帧图片上传至系统中。视频会被解码成一组图片，系统应支持用户以文件夹的形式上传一个视频的所有图片。
- 接受用户输入的标注：用户可以直接在系统中绘制物体的包围框或多边形轮廓，来对图片中的物体进行标注。用户绘制的包围框还可以作为后续机器学习模型的输入。
- 物体轮廓的预测：基于用户输入的包围框标注，系统调用已经训练好的机器辅助标注模型，预测该物体的轮廓，并将结果显示在系统界面上。
- 标注结果的修改：无论是用户直接输入的标注，还是机器预测的物体轮廓，都应允许用户进行修改。用户可以直接完善有错误的标注结果。
- 标注文件的下载：用户标注图片一般是为了训练机器学习模型，因此需要允许用户下载已经标注好的图片对应的标注信息。系统应支持至少一种常见的图片数据集标注格式。
- 物体类别的创建和指定：除了标注物体的位置和轮廓外，一般还需要标注物体所属的类别信息。系统需要支持用户自定义物体的类别信息，并将每个标注和类别信息对应起来。

- 已有标注文件的上传和读取：现存大量以包围框形式标注的数据。因此，除了直接接受用户以拖拽形式输入的标注外，系统还应支持直接读取既有的图片标注。这是系统实现自动将包围框标注转换为像素级标注的必要功能，应支持至少一种常见的图片数据集标注格式。
- 连续标注：对目标跟踪任务的训练数据集进行标注时，需要对来自同一个视频的多帧连续标注。考虑到目标的运动连续性，系统应支持从上一帧的标注结果推断出目标在下一帧的存在位置，以降低用户的标注负担。

除上述内容之外，一些更加长远的需求还包括切换使用多个机器辅助标注模型、增加模型训练功能等。这些需求暂未在本文版本的系统中实现。

5.2.2 系统架构设计

本节从上述的系统需求出发，将整个系统分为四个模块来实现，这四个模块分别是视频图像存储模块、用户标注交互模块、连续标注模块和标注结果保存模块。视频图像存储模块是用户进入该系统后的第一个界面，需要用户指定视频图像存储在服务器上的路径，或将本地的视频图像以文件夹的形式上传至服务器。标注结果保存模块负责整理用户标注的结果，执行必要的检查后将标注文件整理成一定的标注格式供用户下载。在标注视频解码得到的连续多帧图像的场景下，连续标注模块通过调用既有单目标跟踪模型，基于目标在上一帧的标注推测其在下一帧的位置。本文重点关注的是用户标注交互模块。该模块是整个标注工具系统的核心部分，也是在这四个模块中与用户交互次数最多的一个。该模块还负责调用在第4章中介绍的 SiamAnno 模型，用机器学习方法预测目标物体的轮廓多边形。系统的运行流程如图5-1所示。

现有的图片标注工具系统大多仅提供用户绘制包围框或多边形的标注功能，不需要较大算力，在本地安装即可使用。用户可能还需要自行配置软件的运行环境，甚至通过命令行的方式来启动工具。然而本文系统需要运行深度学习模型，对运行环境和计算能力有较高要求（需要 GPU），传统的用户本地部署方式可能并不是最佳选择。因此，本文选择采用客户端/服务器架构，用户在客户端与系统进行交互，简单的计算在用户本地进行，模型调用和存储由服务器负责，客户端和服务器之间通过 HTTP 网络接口进行通信。仅进行少量标注或想要尝鲜

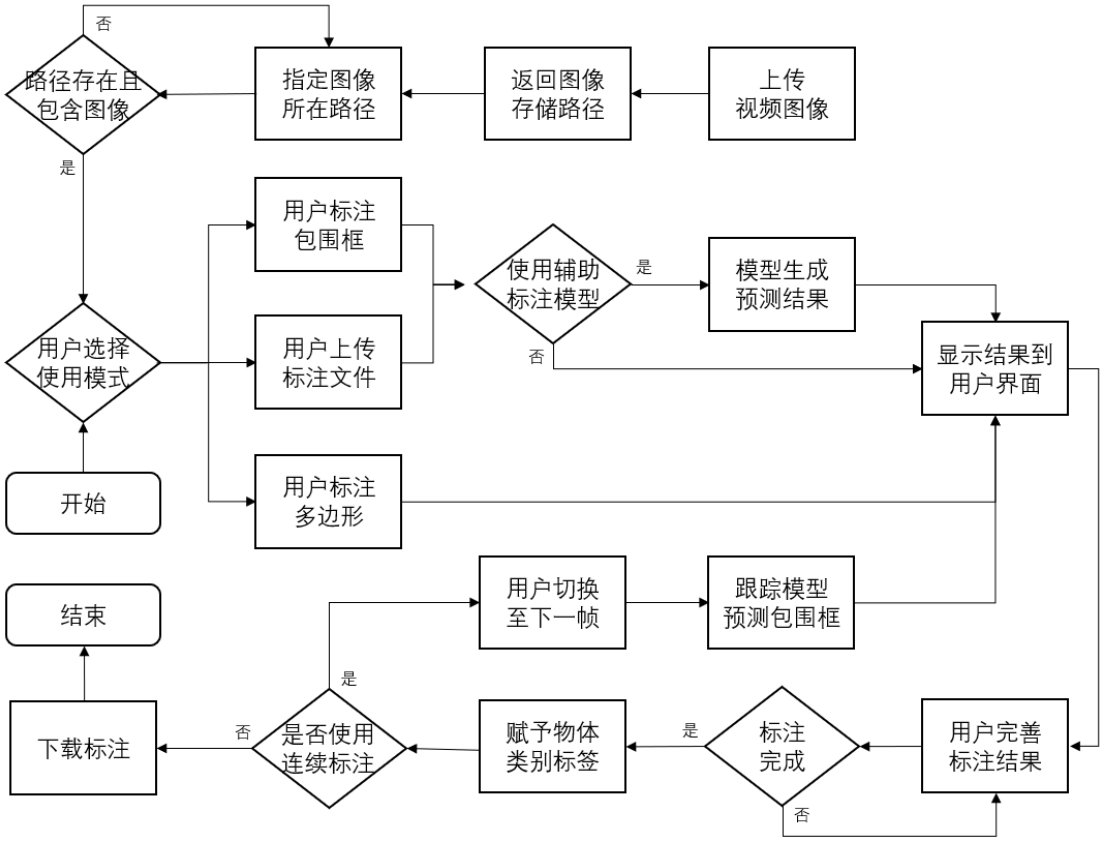

图 5-1: 图片标注工具系统流程图

体验的一般用户可以直接基于公开部署的服务器在自己电脑上使用该系统。专业用户则可以在自己的服务器上部署自己的专用系统。相较于现有的标注工具，本文系统的部署过程也并不复杂，并不会在系统安装这一步带来额外的工作量。

5.3 系统实现

基于5.2节的系统需求和架构设计，本文实现了基于网页应用的标注工具系统。本文将在这一小节分别介绍系统整体的开发环境及其中各模块的具体实现。

5.3.1 开发环境

本文采用客户端/服务器架构来设计整个系统，因此需要进行前后端分离。机器辅助标注功能和连续标注功能的实现依赖于深度神经网络模型的运行，需要大量计算资源，因此后端需要部署在具有 GPU 的服务器上。同时，为了减轻该服务器的计算负担，本文尽可能地将其他计算任务放在前端来完成。在实践中，本文将系统后端部署于 Linux 服务器上，CPU 型号为 Intel(R) Xeon(R) CPU

E5-2678 v3 @ 2.50GHz（12 核 24 线程），GPU 型号为 NVIDIA RTX 2080Ti。系统前端对软硬件的要求较低，只需要操作系统具备浏览器，并能够通过网络连接后端服务器。本文推荐使用 Chrome 浏览器。前端也不依赖于特定的操作系统，常见的 Windows、MacOS 和 Linux 均可。

在软件设计上，本文使用基于 Python 的 Flask 框架来实现后端服务器，接收来自前端用户的 HTTP 请求，执行相应任务后将渲染后的网页或结果发送回客户端。如果用户选择自己搭建后端服务器，则需要进行一定的运行环境配置。例如，运行机器辅助标注模型和用于连续标注的单目标跟踪模型需要深度学习框架 PyTorch。Python 和 PyTorch 的使用存在版本选择问题。本文使用 Python 3.7.11 和 PyTorch 1.8.2，借助 Anaconda 来管理后端的运行环境。Python 和 PyTorch 的高版本对低版本有较好的兼容性，本文的系统也可运行在更高版本的环境中。相比于后端环境，前端资源的加载会通过浏览器自动进行，一般不需要用户进行手动配置。本文使用 HTML、CSS 和 Javascript 来完成前端界面，结合开源前端框架 jQuery 和 Bootstrap 实现更便捷的界面开发和更友好的用户交互。需要注意的是，前端开源框架的跨版本兼容性较差，本文在系统中使用的是 jQuery 3.6.0 和 Bootstrap 4。

5.3.2 模块实现

本文将整个系统分为视频图像存储模块、用户标注交互模块、连续标注模块和标注结果保存模块四个模块。本章节将分别介绍这四个模块。

视频图像存储模块 本文将视频图像存储模块的系统流程图展示在图5-2。如图所示，若用户选择将自己本地的视频图像上传至系统中，系统会以文件夹为单位读取本地文件路径，取该文件夹的名称作为在服务器上的存储文件夹名称。若服务器中已经存在该路径，则使用上传时间点的时间戳作为文件夹名称。当文件夹中迭代地包含子文件夹时，系统会将其中的图像一并上传，但不会在服务器中保留子文件夹的结构。后端将实际存储在服务器上的路径回传给前端界面，并自动填充至待标注图像路径一栏中。

待用户修改和确认待标注图像的存储路径后，系统将在服务器上寻找该路径。如果该路径不存在，或该路径虽然存在，但其中不包含图像时，后端将会向前端返回错误信息，提示用户检查并重新输入图像路径。只有当图像路径真实

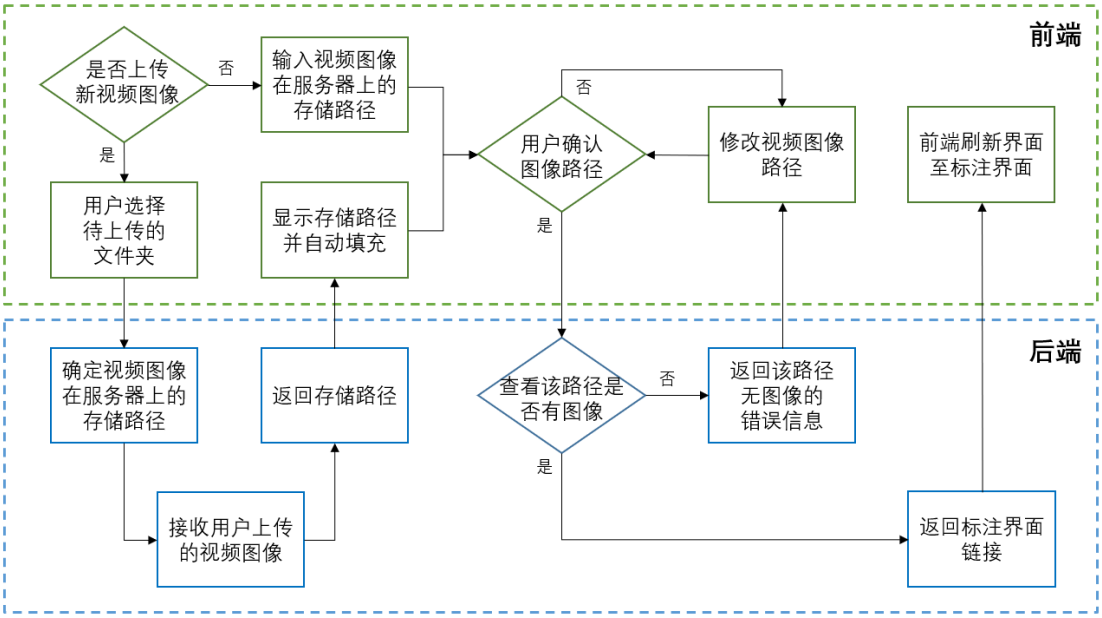

图 5-2: 视频图像存储模块流程图

存在且包含图像时，后端才会渲染图像标注页面，发送到前端，引导用户进入下一步的用户标注交互模块。

用户标注交互模块 用户标注交互模块是整个系统的核心，所有用户操作的标注动作都在该模块中进行。本文在图5-3中展示了该模块的流程图，图中省略了部分操作的实现，例如物体类别标签信息的添加和修改。在该模块中，用户可以选择直接手动标注物体的包围框或多边形轮廓，也可选择使用本文的机器辅助标注模型。机器辅助标注模型的计算资源开销大，运行依赖于 GPU 和特定的环境，因此将其放在后端实现。相应地，考虑到直接标注包围框或多边形时程序的计算量较少，也为了减轻通信成本和后端的计算任务量，本文将不依赖于机器学习模型的标注方式的实现全都放在前端来完成。无论是哪一种标注模式，系统都支持用户直接拖动包围框或多边形的顶点来修改标注结果。模块提供物体类别标签信息的添加、修改和删除功能。用户需要将每一个标注物体关联一个物体类别标签，否则该物体将被系统标记为“Undefined”（未定义的）物体。

该模块的前后端借助 JSON 数据格式来实现信息通信。当用户选择使用智能标注模型时，前端会将图像名称和包围框的坐标信息打包为 JSON 的字符串并以 HTTP 协议中的 POST 方法发送给后端。后端收到该 JSON 数据后，会将 SiamAnno 模型加载至 GPU，并预测物体轮廓。在章节4.2.5中，本文介绍 SiamAnno 模型会输出 196 个轮廓点的位置。这些轮廓点如果全部显示在用户界

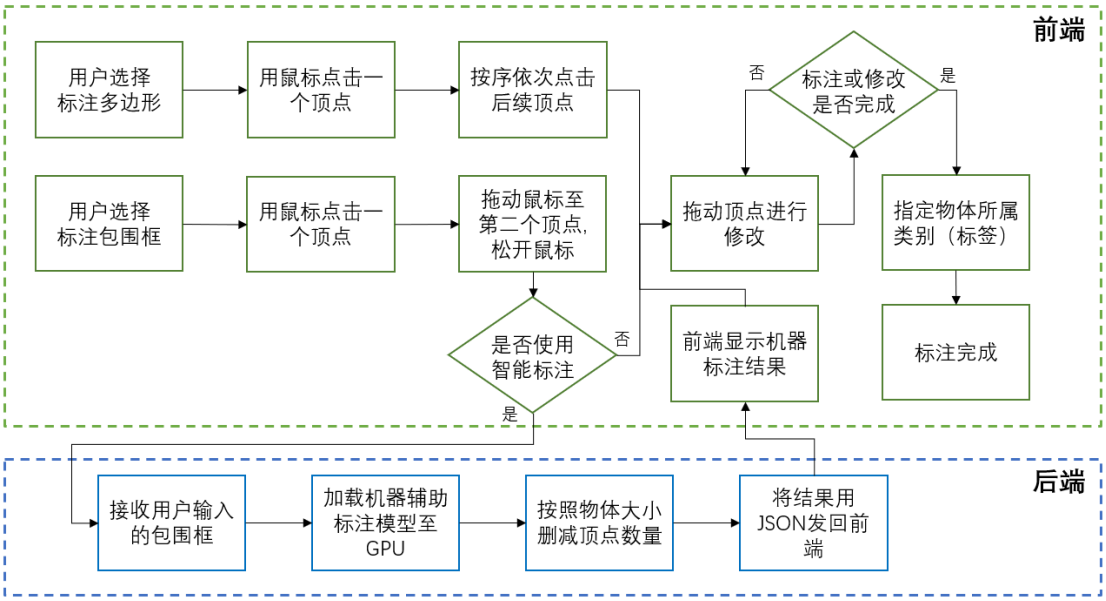

图 5-3: 用户标注交互模块流程图

面上，将非常影响界面观感，也不方便用户操作。因此，系统仅返回一部分轮廓点。系统首先计算标注框的面积，再根据物体大小设定采样间隔，只向用户返回按该间隔采样出来的轮廓点。

连续标注模块 针对单目标跟踪任务训练数据集的标注需求，本文设计了连续标注模块，利用 SRF 等现有的单目标跟踪模型跟踪用户在上一帧标注的包围框，预测标注物体在下一帧出现的位置，自动进行标注，以期减少用户的标注负担。和用户标注交互模块中的机器辅助标注模型类似，预测准确度较高的单目标跟踪模型往往需要运行在有 GPU 的特定环境，计算资源开销大，因此本文将其放在后端实现。前端主要负责与用户的交互，和上一帧标注框的位置、图像路径等必要信息的保存。前后端的信息通信通过 JSON 数据格式来实现，与用户标注交互模块相同。后端的跟踪模型预测出待标注物体在下一帧的包围框坐标，并返回给前端，显示在用户界面上，用户可以拖动包围框的顶点对结果进行修改。整个模块的流程如图5-4所示。

标注结果保存模块 用户的所有标注结果，以及物体的类别信息，都保存在用户本地的浏览器中，因此整个标注结果保存的实现不依赖于后端服务器。该模块的流程图如图5-5所示。系统默认将当前时间的时间戳作为标注文件的名称，文件的下载路径是用户浏览器默认的下载路径。系统会分别检查是否有标注记录，及是否有物体类别标签信息。只有这二者均存在时，才会实质上进入标注结果保存流程，否则将向用户返回错误提示，终止结果保存流程。系统应支持至少

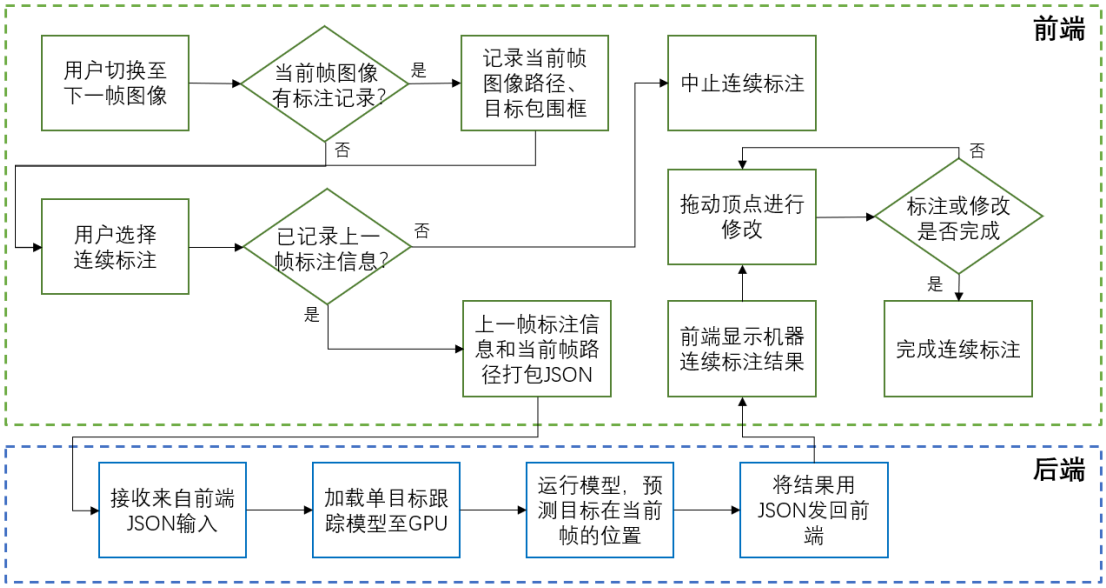

图 5-4: 连续标注模块流程图

一种常见的图像标注格式。考虑到 COCO 数据集在现今计算机视觉任务中的重要性，其伴生的 COCO 数据格式也成为了一种主流数据格式，因此本文的系统实现的是 COCO 数据格式的标注文件下载。COCO 数据格式本身是 JSON 字符串。系统分别组织物体类别标签信息、图像信息和标注信息，并用 JSON 将这些信息字符串化，利用 Javascript 中的“二进制大对象”（Blob）将 JSON 字符串形成文件，触发浏览器的下载动作。

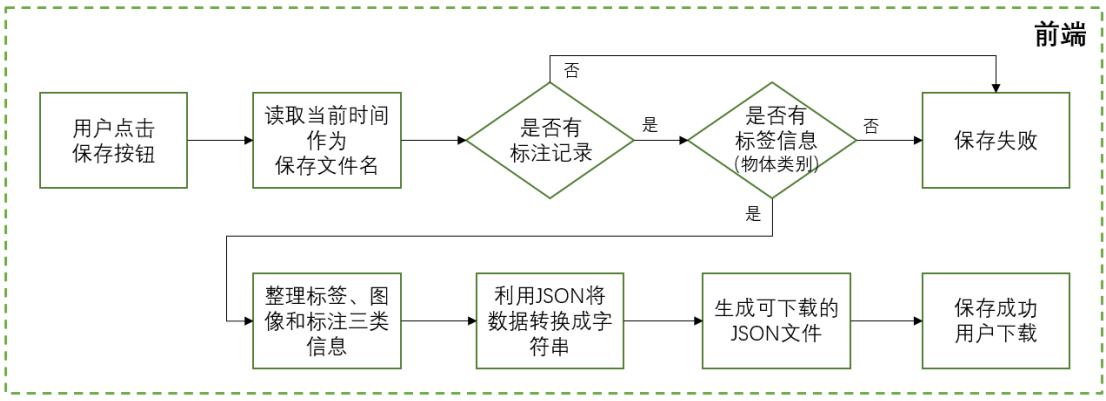

图 5-5: 标注结果保存模块流程图

5.4 系统使用流程与效果

为了更好地服务全球用户，系统的默认语言是英文。系统一共包含两个界面，分别是视频图像上传界面和用户标注交互界面。前者对应视频图像存储模

块，后者则包含用户标注交互模块、连续标注模块和标注结果保存模块。

用户进入该系统，将首先进入视频图像上传界面，如图5-6所示。用户可以点击“上传”按钮上传包含待标注视频图像的文件夹。上传过程中将显示上传进度条，待上传完成后会显示“成功”字样，并显示上传至服务器的路径名称。该路径名称将被自动填充至界面上的待标注图像路径输入栏中。当待标注图像已经储存于服务器端时，用户也可以跳过上传步骤，直接在该输入栏中输入相应路径。路径输入完成后，用户点击“选择”按钮，系统将检查该路径的合法性。若该路径存在且其中包含图像，前端界面将刷新显示下一步的用户标注交互界面。否则，系统将提示错误信息，引导用户修改输入的图像路径。

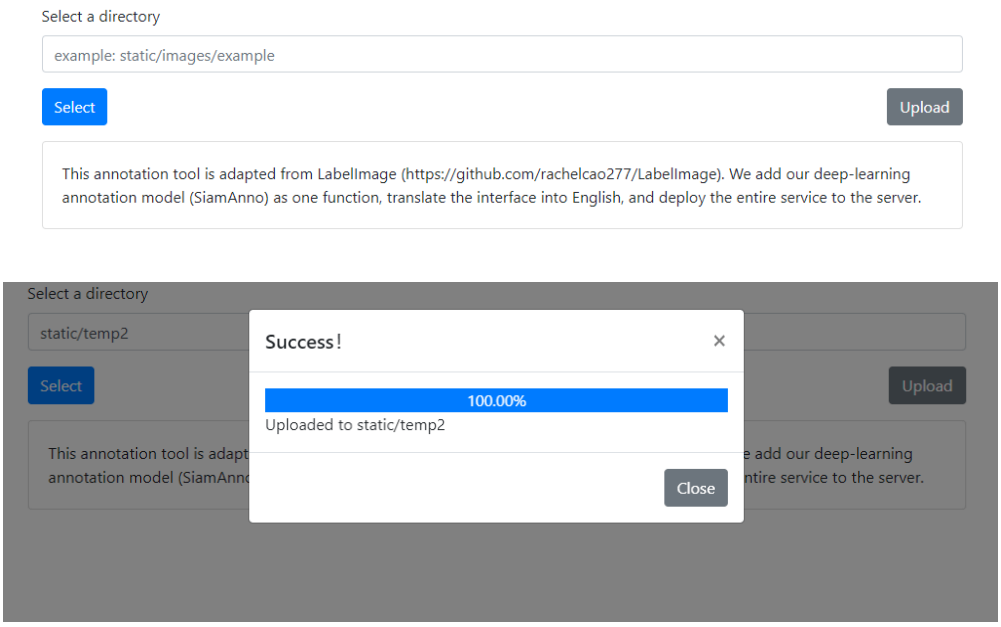

图 5-6: 视频图像上传界面

用户标注交互界面如图5-7所示。用户将在这个界面完成对图片的标注和标注结果的下载。工具栏位于界面的左侧，其中包含如标注模式选择、标注下载、上传既有标注文件、使用连续标注功能等按钮；界面上方显示了图像的名称，并包含如图像切换、标签显示等少量功能按钮；界面右侧是标注状态栏，包含已标注的物体和已执行的动作，本文在图5-8左侧中放大了这两部分状态栏。

从图5-7中可以看出，不同的标注目标被用不同颜色的蒙版遮罩，这便于用户区分不同物体。用户可通过点击界面右上方的标签显示按钮来决定是否要在蒙版上显示物体类别。图例是显示了物体类别的情况。用户可以通过滚轮来放大或缩小图像，界面左下角显示了实时的物体缩放比例，以及画面显示的图像

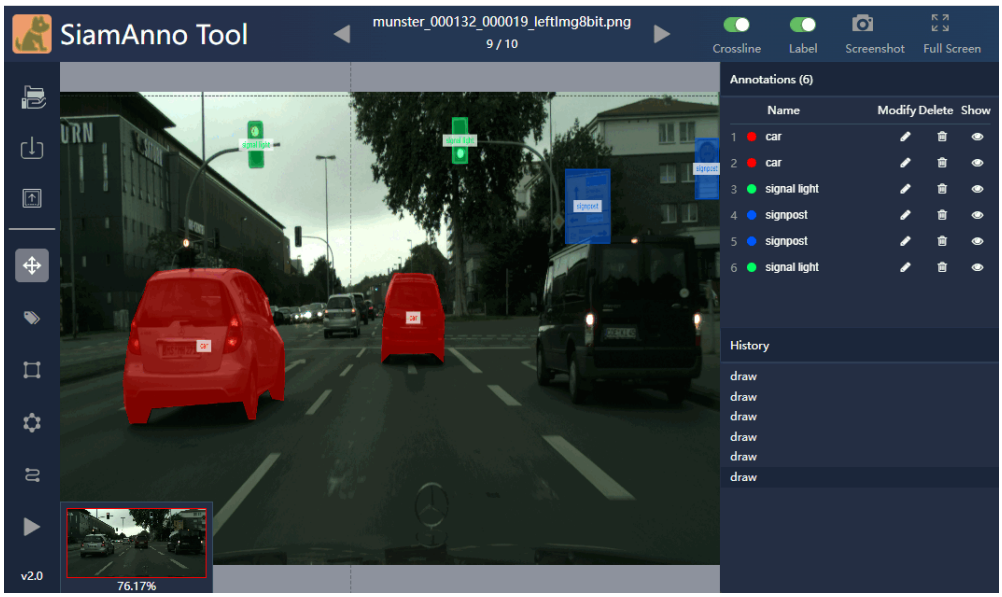

图 5-7: 用户标注交互界面

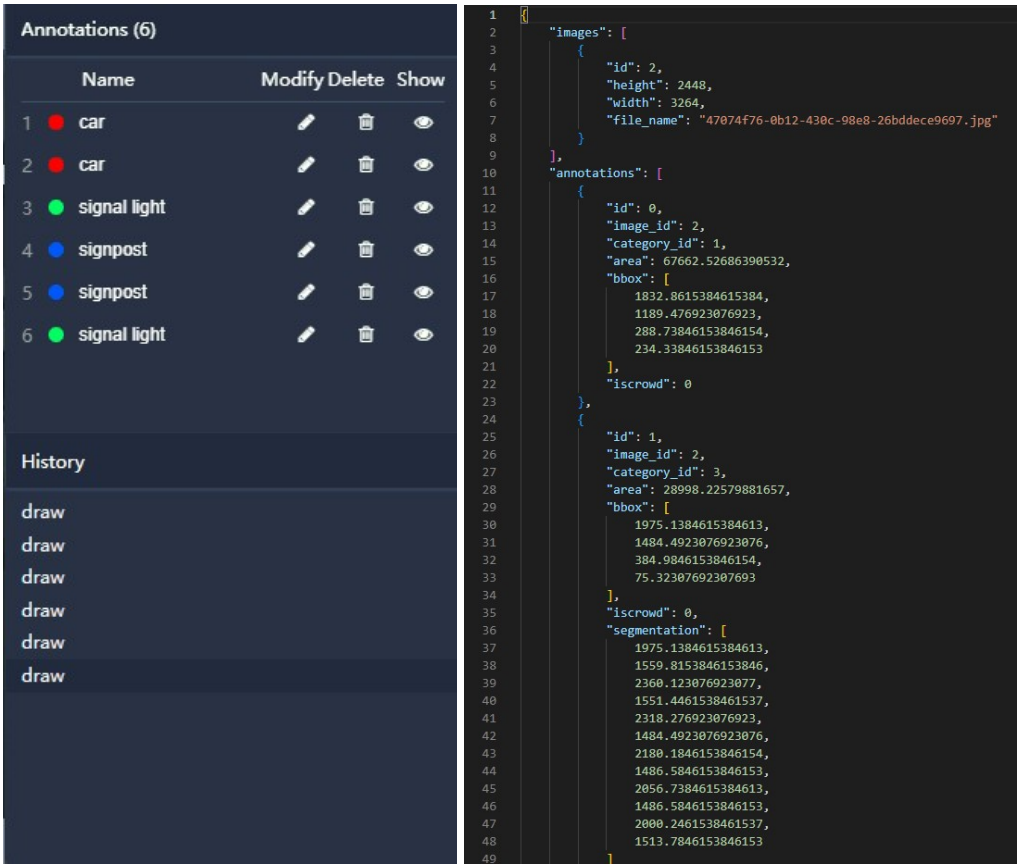

图 5-8: 标注界面细节与导出结果示例

范围。在图5-8中展示的已标注物体一栏，用户可以点击每一个标注的“修改”按钮来针对性地修改该物体的轮廓或类别标签，亦可点击“删除”或“显示”按钮来删除或不显示某些标注结果。下方的已执行动作栏则按照顺序显示了用户

的历史操作。用户点击任意操作，系统均将回退到该操作前的标注状态。

用户点击下载标注的按钮后，系统会生成 COCO 标注格式的标注文件，并自动触发浏览器来下载该文件。图5-8右侧展示了一个生成的标注文件的例子。值得一提的是，借助于系统的标注上传功能，用户可在下次“继续标注”之前没有标注完成的部分。例如在下一次标注的时候上传之前下载的标注文件，系统将读取该文件，并将相应的物体类别标签信息和标注信息刷新到用户界面上。用户可以在此基础上进行后续的标注工作。

5.5 小结

在本章中，本文基于第3章的单目标视频跟踪模型 SRF 和第4章的机器辅助标注模型 SiamAnno，搭建了一个面向跟踪任务的图片标注工具系统 SiamAnno Tool。用户可以在工具中上传待标注的视频图片，再选择多种标注方式对物体进行标注。基于机器辅助标注模型，系统可以将用户的包围框标注转换成多边形轮廓标注；基于单目标视频跟踪模型，系统可以基于上一帧的标注结果预测待标注物体在下一帧的位置与大小，自动完成标注。系统支持用户修改手动绘制或模型生成的标注结果，并可以生成 COCO 格式的标注文件供用户下载。系统使用客户端/服务器结构实现，用户只需要一个浏览器即可使用该系统，简单快捷。除了视频目标跟踪任务外，本文的系统也支持标注如目标检测、实例分割等其他下游计算机视觉任务的数据集。本文希望本章的工作可以为今后相关数据集的构建提供有效的标注工具。通过该系统，本文验证了本文提出的机器辅助标注方法的实用性和有效性，说明孪生网络结构在实际的标注任务中有着巨大的应用潜力。

第六章　总结与展望

本文围绕面向真实场景的单目标跟踪算法应用，从网络模型和数据两个角度，对长时跟踪算法及基于跟踪算法网络结构的智能标注算法展开研究，并设计了面向跟踪任务的图片标注工具系统，将两个算法嵌入其中，使得本文的研究不仅具有学术意义，更实现了将算法落地，为真实场景的现实需求服务。

相比于短时跟踪任务，在长时跟踪任务中，模型不仅要给出跟踪目标在画面中的位置，还需要处理目标从画面中消失和再出现的情况。本文在第3章针对这样一个更加接近真实场景的任务，设计了新的长时跟踪与分割算法框架 SRF。本文并没有按照既有研究中“打补丁”的思路通过增加新的组件来尝试提高算法的跟踪精度。相反，本文受到了集成学习中“部分可能比全部更好”的现象的启发，对模型组件做减法，在保持了一定的跟踪精度的同时，让整个算法框架更加简洁。本文的实验也证明了，提高跟踪精度并不一定需要设计新的神经网络，充分挖掘现有跟踪方法的潜力也可以在许多数据集上达到甚至超越现有方法。

真实的模型部署场景有不同的硬件条件，不同现实任务对跟踪速度也有着各自的要求。考虑到这种情况，本文的算法引入了一个速度控制参数，使得整个长时跟踪算法的运行速度是可以调整的。据本文了解，SRF 是第一个速度可以连续调节的长时跟踪器。此外，越来越多的现实任务需要机器预测更加准确的物体位置与轮廓。SRF 的结果精炼模块可以预测跟踪目标的像素掩膜，输出像素级的结果，进而同时实现长时目标跟踪与分割任务。SRF 也是目前为数不多的可以同时完成视频目标分割和长时目标跟踪任务的方法。

SRF 中的结果精炼模块具有基于包围框预测物体轮廓的能力。这启发本文可以将跟踪模型的网络结构应用于智能标注任务中，并反过来帮助目标跟踪任务收集更多高质量的数据集用于模型训练。本文在第4章设计了智能标注算法 SiamAnno，创新性地将孪生网络结构应用在机器辅助标注任务中，实现将标注人员输入的包围框自动转换成物体多边形轮廓的功能。将孪生网络结构作为模型的骨干结构，模型可以充分利用其单样本学习能力，在应对标注新类别物体或拍摄环境发生变化的情况下依然可以准确地预测物体的轮廓。这种“跨域”标

注场景在真实场景中是非常常见的，本文的方法具有很高的实用价值。实验表明本文的智能标注算法可以在不引入新数据集的情况下提高原有单目标跟踪任务的跟踪精度。

除了好的算法框架外，跟踪精度高的模型也离不开高质量的训练数据集。在第5章，本文将前两章的研究成果汇总在面向跟踪任务的图片标注工具系统中，从数据角度来服务面向真实场景的跟踪任务。目标跟踪数据集在逻辑上由视频构成，实际则常常将各帧以图片的形式提供。基于第3章的跟踪算法，系统提供了“连续标注”功能，基于物体在上一帧图像上的位置推测其在下一帧的位置；基于第4章的智能标注算法，系统可以根据用户输入的包围框预测物体的多边形轮廓，降低人工标注像素级分割掩膜的工作量，顺应当今计算机视觉任务预测粒度精细化的趋势，满足更多需要精准预测的现实任务的需要。本文也通过实验证明，对既有的包围框标注的跟踪数据集使用本文的智能标注算法，即使没有任何人工的修改完善，也可以提高现存跟踪算法的跟踪精度。

按照本文的现有进展，本文也总结分析了相关研究可以继续提升之处：

- 包括但不限于 SRF 的现有长时跟踪方法常由多个网络组成，其中每个部分使用了层数不完全相同的 ResNet[112] 作为骨干网络。如果能够重复使用骨干网络，将三个网络合并为一个，可以提高框架运行速度，减少内存使用。
- SRF 中使用了多个跟踪器。在其他计算机视觉任务中有研究使用一个代理共享网络（agent sharing network）[113] 来融合多个摄像机对同一个目标的跟踪结果，类似方法可以借鉴到长时跟踪框架中。
- 基于孪生网络的智能标注算法有很大的研究空间，如尝试不同的人机交互手段，或尝试使用基于视觉或运动显著性（visual or motion saliency）的方法[114-116] 更好地区分背景和干扰物。
- 今后的面向跟踪任务的图片标注系统可以嵌入更多的跟踪算法或智能标注算法，以满足不同场景任务的需要。此外，还可考虑从工程角度加速图片上传速度、提供断点续传、支持用户直接上传视频等功能。

致　　谢

光阴荏苒，日月如梭。我即将离开南大校园。回想这七年时光，有付出，有感慨，有收获，有怀念。此刻，我衷心地感谢在这过程中给予我关心、帮助和支持的师长、同学与家人们。

我现在拥有的专业知识和能力离不开南京大学计算机学科和全国重点实验室对我的培养和支持。学校和院系为学生提供了领先的培养体系，一流的教师授课和丰富的计算资源，不仅奠定了我的专业基础，也为我开展科研活动，探索领域前沿提供了充足的条件保障。

我尤其需要感谢的是我的导师申富饶教授。申老师为人谦逊，平易近人。即使有繁重的学院管理工作，申老师依然每周抽出时间和每一位同学开展个人讨论。当我的论文写作遇到困难时，申老师积极帮我打开思路，并耐心分享自己过往的研究经验。申老师总是设身处地为学生着想，不仅尊重每一个人的兴趣，在科研方向的选择上给予学生极大的自由度，在了解我希望毕业后从事跨专业的工作后，还尽可能地为我的求职提供条件保障。

我还要感谢赵健老师。赵老师一直认真参加组内的讨论班，不但分享自己的研究和论文写作经验，还帮助我修改论文，逐字逐句，不厌其烦地指出其中的语法错误和技术问题，并给出建设性的修改意见。

感谢一起做研究、做项目的 RINC 研究组的同门们。无论是在学习还是生活上，你们都给了我许多帮助。也感谢从本科陪伴至今的两位舍友，你们为我的研究生生活了增加了很多欢乐与温暖。

更要感谢你手你屋和你群。在这里，我如此幸运地收获了最为真诚无暇的友谊。每一次低落，必有你们向我伸手。“聚是一团火，散是满天星。”祝愿你手越来越好，也祝愿你手的每一个人都能成为闪闪发光的自己。

还要感谢我的家人，特别是我的父母。他们是我的避风港，是我最坚强的后盾。他们无条件地支持我的每一次选择，给予我最无私的爱。

毕业后，我将继续怀揣“嚼得菜根，做得大事”的南大人精神披荆斩棘，继续前行。你们的教诲和帮助将成为我走向社会的宝贵财富。

参考文献

[1] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60 (6): 84-90.

[2] DENG J, DONG W, SOCHER R, et al. ImageNet: A large-scale hierarchical image database[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2009: 248-255.

[3] LUITEN J, VOIGTLAENDER P, LEIBE B. PReMVOS: Proposal-generation, refinement and merging for video object segmentation[C]//Proceedings of the Asian Conference on Computer Vision. 2018: 565-580.

[4] LAN S, YU Z, CHOY C, et al. DISCOBOX: Weakly supervised instance segmentation and semantic correspondence from box supervision[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 3406-3416.

[5] TIAN Z, SHEN C, WANG X, et al. BoxInst: High-performance instance segmentation with box annotations[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 5443-5452.

[6] KRISTAN M, LEONARDIS A, MATAS J, et al. The eighth visual object tracking VOT2020 challenge results[C]//Proceedings of the European Conference on Computer Vision Workshops. 2020: 547-601.

[7] KRISTAN M, MATAS J, LEONARDIS A, et al. The ninth visual object tracking VOT2021 challenge results[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 2711-2738.

[8] KRISTAN M, LEONARDIS A, MATAS J, et al. The tenth visual object tracking VOT2022 challenge results[C]//Proceedings of the European Conference on Computer Vision Workshops. 2022.

[9] DANELLJAN M, BHAT G, KHAN F S, et al. ATOM: Accurate tracking by overlap maximization[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 4660-4669.

[10] BHAT G, DANELLJAN M, GOOL L V, et al. Learning discriminative model prediction for tracking[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 6182-6191.

[11] DANELLJAN M, GOOL L V, TIMOFTE R. Probabilistic regression for visual tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 7183-7192.

[12] CUI Y, JIANG C, WANG L, et al. Fully convolutional online tracking[J]. Computer Vision and Image Understanding, 2022, 224: 103547.

[13] BHAT G, DANELLJAN M, VAN GOOL L, et al. Know your surroundings: Exploiting scene information for object tracking[C]//Proceedings of the European Conference on Computer Vision. 2020: 205-221.

[14] LI B, YAN J, WU W, et al. High performance visual tracking with siamese region proposal network[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 8971-8980.

[15] LI B, WU W, WANG Q, et al. SiamRPN++: Evolution of siamese visual tracking with very deep networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 4282-4291.

[16] XU Y, WANG Z, LI Z, et al. SiamFC++: Towards robust and accurate visual tracking with target estimation guidelines[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(07): 12549-12556.

[17] ZHU Z, WANG Q, LI B, et al. Distractor-aware siamese networks for visual object tracking[C]//Proceedings of the European Conference on Computer Vision. 2018: 101-117.

[18] YU Y, XIONG Y, HUANG W, et al. Deformable siamese attention networks for visual object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6728-6737.

[19] WANG Q, ZHANG L, BERTINETTO L, et al. Fast online object tracking and segmentation: A unifying approach[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 1328-1338.

[20] KALAL Z, MIKOLAJCZYK K, MATAS J. Tracking-learning-detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 34(7): 1409-1422.

[21] DAI K, ZHANG Y, WANG D, et al. High-performance long-term tracking with meta-updater[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6298-6307.

[22] YAN B, ZHAO H, WANG D, et al. 'Skimming-Perusal' Tracking: A framework for real-time and robust long-term tracking[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 2385-2393.

[23] KRISTAN M, LEONARDIS A, MATAS J, et al. The sixth visual object tracking VOT2018 challenge results[C]//Proceedings of the European Conference on Computer Vision Workshops. 2018.

[24] KRISTAN M, MATAS J, LEONARDIS A, et al. The seventh visual object tracking VOT2019 challenge results[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. 2019.

[25] MAYER C, DANELLJAN M, PAUDEL D P, et al. Learning target candidate association to keep track of what not to track[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 13444-13454.

[26] ZHANG Y, MA B, WU J, et al. Capturing relevant context for visual tracking [J]. IEEE Transactions on Multimedia, 2020, 23: 4232-4244.

[27] VOIGTLAENDER P, LUITEN J, TORR P H, et al. Siam R-CNN: Visual tracking by re-detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6578-6588.

[28] ACUNA D, LING H, KAR A, et al. Efficient interactive annotation of segmentation datasets with Polygon-RNN++[C]//Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2018: 859-868.

[29] ROTHER C, KOLMOGOROV V, BLAKE A. "GrabCut" interactive foreground extraction using iterated graph cuts[J]. ACM Transactions on Graphics, 2004, 23(3): 309-314.

[30] MORTENSEN E N, BARRETT W A. Intelligent scissors for image composition [C]//Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques. 1995: 191-198.

[31] MANINIS K K, CAELLES S, PONT-TUSET J, et al. Deep Extreme Cut: From extreme points to object segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 616-625.

[32] ZHANG S, LIEW J H, WEI Y, et al. Interactive object segmentation with inside-outside guidance[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 12234-12244.

[33] LIN Z, ZHANG Z, CHEN L Z, et al. Interactive image segmentation with first click attention[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 13339-13348.

[34] LING H, GAO J, KAR A, et al. Fast interactive object annotation with Curve-GCN[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 5257-5266.

[35] KIM N, KANG B, CHO Y. Split GCN: Effective interactive annotation for segmentation of disconnected instance[J]. ArXiv preprint arXiv:2112.06454, 2021.

[36] ZHOU Z H, WU J, TANG W. Ensembling neural networks: Many could be better than all[J]. Artificial intelligence, 2002, 137(1-2): 239-263.

[37] FAN H, LIN L, YANG F, et al. LaSOT: A high-quality benchmark for large-scale single object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 5374-5383.

[38] FAN H, BAI H, LIN L, et al. LaSOT: A high-quality large-scale single object tracking benchmark[J]. International Journal of Computer Vision, 2021, 129(2): 439-461.

[39] MOUDGIL A, GANDHI V. Long-term visual object tracking benchmark[C]// Proceedings of the Asian Conference on Computer Vision. 2018: 629-645.

[40] VALMADRE J, BERTINETTO L, HENRIQUES J F, et al. Long-term tracking in the wild: A benchmark[C]//Proceedings of the European conference on computer vision. 2018: 670-685.

[41] MUELLER M, SMITH N, GHANEM B. A benchmark and simulator for UAV tracking[C]//Proceedings of the European Conference on Computer Vision. 2016: 445-461.

[42] ZHOU B, ZHAO H, PUIG X, et al. Scene parsing through ade20k dataset[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 633-641.

[43] GEIGER A, LENZ P, URTASUN R. Are we ready for autonomous driving? the kitti vision benchmark suite[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2012: 3354-3361.

[44] SUN X, CHRISTOUDIAS C M, FUA P. Free-shape polygonal object localization [C]//Proceedings of the European Conference on Computer Vision. 2014: 317-332.

[45] MUELLER J, THYAGARAJAN A. Siamese recurrent architectures for learning sentence similarity[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2016, 30(1): 2786-2792.

[46] NAIR V, HINTON G E. Rectified linear units improve restricted boltzmann machines[C]//Proceedings of International Conference on Machine Learning. 2010: 1-8.

[47] CHOPRA S, HADSELL R, LECUN Y. Learning a similarity metric discriminatively, with application to face verification[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2005, 1: 539-546.

[48] SALAKHUTDINOV R, HINTON G. Learning a nonlinear embedding by preserving class neighbourhood structure[C]//Artificial Intelligence and Statistics. 2007: 412-419.

[49] ZAGORUYKO S, KOMODAKIS N. Learning to compare image patches via convolutional neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 4353-4361.

[50] KOCH G, ZEMEL R, SALAKHUTDINOV R, et al. Siamese neural networks for one-shot image recognition[C]//Proceedings of International Conference on Machine Learning Deep Learning Workshop. 2015.

[51] CHEN K, SALMAN A. Extracting speaker-specific information with a regularized siamese deep network[J]. Advances in Neural Information Processing Systems, 2011, 24: 1-9.

[52] YIH W T, TOUTANOVA K, PLATT J C, et al. Learning discriminative projections for text similarity measures[C]//Proceedings of the 15th Conference on Computational Natural Language Learning. 2011: 247-256.

[53] REIMERS N, GUREVYCH I. Sentence-BERT: Sentence embeddings using siamese bert-networks[J]. ArXiv preprint arXiv:1908.10084, 2019.

[54] BERTINETTO L, VALMADRE J, HENRIQUES J F, et al. Fully-convolutional siamese networks for object tracking[C]//Proceedings of the European Conference on Computer Vision. 2016: 850-865.

[55] YAN B, ZHANG X, WANG D, et al. Alpha-Refine: Boosting tracking performance by precise bounding box estimation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 5289-5298.

[56] ZHANG D, FU Y, ZHENG Z. UAST: Uncertainty-aware siamese tracking[C]// Proceedings of International Conference on Machine Learning. 2022: 26161-26175.

[57] XUAN S, LI S, ZHAO Z, et al. Siamese networks with distractor-reduction method for long-term visual object tracking[J]. Pattern Recognition, 2021, 112: 107698.

[58] CHOI S, LEE J, LEE Y, et al. Robust long-term object tracking via improved discriminative model prediction[C]//Proceedings of the European Conference on Computer Vision. 2020: 602-617.

[59] WU H, YANG X, YANG Y, et al. Flow guided short-term trackers with cascade detection for long-term tracking[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. 2019.

[60] ZHANG Z, PENG H, FU J, et al. Ocean: Object-aware anchor-free tracking[C]// Proceedings of the European Conference on Computer Vision. 2020: 771-787.

[61] LI X, HUANG L, WEI Z. A twofold convolutional regression tracking network with temporal and spatial mechanism[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2021, 32(3): 1537-1551.

[62] ZHOU Z, LI X, ZHANG T, et al. Object tracking via spatial-temporal memory network[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2022, 32(5): 2976-2989.

[63] LIAO B, WANG C, WANG Y, et al. PG-Net: Pixel to global matching network for visual tracking[C]//Proceedings of the European Conference on Computer Vision. 2020: 429-444.

[64] WANG G, LUO C, SUN X, et al. Tracking by instance detection: A meta-learning approach[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6288-6297.

[65] JIANG M, ZHAO Y, KONG J. Mutual learning and feature fusion siamese networks for visual object tracking[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 31(8): 3154-3167.

[66] WANG N, ZHOU W, WANG J, et al. Transformer meets tracker: Exploiting temporal context for robust visual tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 1571-1580.

[67] CHEN X, YAN B, ZHU J, et al. Transformer tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 8126-8135.

[68] ZHANG H, CHENG L, ZHANG T, et al. Target-distractor aware deep tracking with discriminative enhancement learning loss[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2022, 32(9): 6267-6278.

[69] WANG X, CHEN Z, TANG J, et al. Dynamic attention guided multi-trajectory analysis for single object tracking[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2021, 31(12): 4895-4908.

[70] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[J]. Advances in Neural Information Processing Systems, 2015, 28: 1-9.

[71] HUANG L, ZHAO X, HUANG K. GlobalTrack: A simple and strong baseline for long-term tracking[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(07): 11037-11044.

[72] PERNICI F, DEL BIMBO A. Object tracking by oversampling local features[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 36(12): 2538-2551.

[73] HONG Z, CHEN Z, WANG C, et al. Multi-store tracker (MUSTer): A cognitive psychology inspired approach to object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2015: 749-758.

[74] CHIEN S Y, CHAN W K, TSENG Y H, et al. Video object segmentation and tracking framework with improved threshold decision and diffusion distance[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2013, 23(6): 921-934.

[75] PERAZZI F, PONT-TUSET J, MCWILLIAMS B, et al. A benchmark dataset and evaluation methodology for video object segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 724-732.

[76] LUKEZIC A, MATAS J, KRISTAN M. D3S-a discriminative single shot segmentation tracker[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 7133-7142.

[77] ROBINSON A, LAWIN F J, DANELLJAN M, et al. Learning fast and robust target models for video object segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 7406-7415.

[78] DUKE B, AHMED A, WOLF C, et al. SSTVOS: Sparse spatiotemporal transformers for video object segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 5912-5921.

[79] ZHANG Y, WU Z, PENG H, et al. A transductive approach for video object segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6949-6958.

[80] CHEN X, LI Z, YUAN Y, et al. State-aware tracker for real-time video object segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 9384-9393.

[81] BHAT G, LAWIN F J, DANELLJAN M, et al. Learning what to learn for video object segmentation[C]//Proceedings of the European Conference on Computer Vision. 2020: 777-794.

[82] WEN L, DU D, ZHU P, et al. Detection, tracking, and counting meets drones in crowds: A benchmark[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 7812-7821.

[83] HUANG P, HAN J, LIU N, et al. Scribble-supervised video object segmentation [J]. IEEE/CAA Journal of Automatica Sinica, 2021, 9(2): 339-353.

[84] ZHANG D, HAN J, YANG L, et al. SPFTN: A joint learning framework for localizing and segmenting objects in weakly labeled videos[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 42(2): 475-489.

[85] CHEN X, ZHAO Z, ZHANG Y, et al. FocalClick: Towards practical interactive image segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 1300-1309.

[86] LIN Z, ZHANG Z, HAN L H, et al. Multi-mode interactive image segmentation [C]//Proceedings of the 30th ACM International Conference on Multimedia. 2022: 905-914.

[87] HAO Y, LIU Y, WU Z, et al. EdgeFlow: Achieving practical interactive segmentation with edge-guided flow[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 1551-1560.

[88] DING Z, WANG T, SUN Q, et al. A dual-stream framework guided by adaptive gaussian maps for interactive image segmentation[J]. Knowledge-Based Systems, 2021, 223: 107033.

[89] DONG Z, LI J, FANG T, et al. Lightweight boundary refinement module based on point supervision for semantic segmentation[J]. Image and Vision Computing, 2021, 110: 104169.

[90] TANG C, CHEN H, LI X, et al. Look closer to segment better: Boundary patch refinement for instance segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 13926-13935.

[91] CASELLES V, KIMMEL R, SAPIRO G. Geodesic active contours[J]. International Journal of Computer Vision, 1997, 22(1): 61-79.

[92] WANG Z, ACUNA D, LING H, et al. Object instance annotation with deep extreme level set evolution[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 7500-7508.

[93] CASTREJON L, KUNDU K, URTASUN R, et al. Annotating object instances with a Polygon-RNN[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 5230-5238.

[94] DONG Z, ZHANG R, SHAO X. Automatic annotation and segmentation of object instances with deep active curve network[J]. IEEE Access, 2019, 7: 147501-147512.

[95] ZHAO J, DAI K, WANG D, et al. Online filtering training samples for robust visual tracking[C]//Proceedings of the 28th ACM International Conference on Multimedia. 2020: 1488-1496.

[96] LIN T Y, MAIRE M, BELONGIE S, et al. Microsoft COCO: Common objects in context[C]//Proceedings of the European Conference on Computer Vision. 2014: 740-755.

[97] CAI Z, VASCONCELOS N. Cascade R-CNN: Delving into high quality object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 6154-6162.

[98] HE K, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[C]//Proceedings of the IEEE/CVF Conference on Computer Vision. 2017: 2961-2969.

[99] LUKEŹIČ A, ZAJC L Č, VOJÍŘ T, et al. Performance evaluation methodology for long-term single-object tracking[J]. IEEE Transactions on Cybernetics, 2020, 51(12): 6305-6318.

[100] TANG F, LING Q. Contour-aware long-term tracking with reliable re-detection [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2019, 30 (12): 4739-4754.

[101] WU Y, LIM J, YANG M H. Online object tracking: A benchmark[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2013: 2411-2418.

[102] XU N, YANG L, FAN Y, et al. Youtube-VOS: A large-scale video object segmentation benchmark[J]. ArXiv preprint arXiv:1809.03327, 2018.

[103] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 770-778.

[104] PENG S, JIANG W, PI H, et al. Deep snake for real-time instance segmentation [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 8533-8542.

[105] LIU Z, LIEW J H, CHEN X, et al. DANCE: A deep attentive contour model for efficient instance segmentation[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021: 345-354.

[106] DENG R, SHEN C, LIU S, et al. Learning to predict crisp boundaries[C]// Proceedings of the European Conference on Computer Vision. 2018: 562-578.

[107] MARTIN D R, FOWLKES C C, MALIK J. Learning to detect natural image boundaries using local brightness, color, and texture cues[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(5): 530-549.

[108] CORDTS M, OMRAN M, RAMOS S, et al. The cityscapes dataset for semantic urban scene understanding[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 3213-3223.

[109] CHEN L C, FIDLER S, YUILLE A L, et al. Beat the mturkers: Automatic image labeling from weak 3d supervision[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 3198-3205.

[110] CHEN L C, PAPANDREOU G, KOKKINOS I, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(4): 834-848.

[111] HUANG L, ZHAO X, HUANG K. Got-10k: A large high-diversity benchmark for generic object tracking in the wild[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 43(5): 1562-1577.

[112] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 770-778.

[113] ZHU P, ZHENG J, DU D, et al. Multi-drone-based single object tracking with agent sharing network[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 31(10): 4058-4070.

[114] JIANG L, WANG Z, XU M, et al. Image saliency prediction in transformed domain: A deep complex neural network method[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33(01): 8521-8528.

[115] ZHOU F, YAO R, LIAO G, et al. Visual saliency via embedding hierarchical knowledge in a deep neural network[J]. IEEE Transactions on Image Processing, 2020, 29: 8490-8505.

[116] PRAMONO R R A, CHEN Y T, FANG W H. Spatial-temporal action localization with hierarchical self-attention[J]. IEEE Transactions on Multimedia, 2021, 24: 625-639.

[117] ZHANG Z, PENG H. Deeper and wider siamese networks for real-time visual tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 4591-4600.

[118] WANG N, ZHOU W, LI H. Reliable re-detection for long-term tracking[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2018, 29(3): 730-743.

[119] LUKEŽIČ A, ZAJC L Č, VOJÍŘ T, et al. FuCoLoT–A fully-correlational long-term tracker[C]//Proceedings of the Asian Conference on Computer Vision. 2018: 595-611.

[120] LI X, ZHAO L, JI W, et al. Multi-task structure-aware context modeling for robust keypoint-based object tracking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 41(4): 915-927.

[121] RAMESH B, ZHANG S, YANG H, et al. e-TLD: Event-based framework for dynamic object tracking[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 31(10): 3996-4006.

[122] HAN J, YANG L, ZHANG D, et al. Reinforcement cutting-agent learning for video object segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 9080-9089.

[123] CHEN C, WANG H, FANG Y, et al. A novel long-term iterative mining scheme for video salient object detection[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2022, 32(11): 7662-7676.

[124] TIAN Z, LI X, ZHENG Y, et al. Graph-convolutional-network-based interactive prostate segmentation in MR images[J]. Medical Physics, 2020, 47(9): 4164-4176.

[125] RAMADAN H, LACHQAR C, TAIRI H. A survey of recent interactive image segmentation methods[J]. Computational Visual Media, 2020, 6(4): 355-384.

[126] LIU W, MA C, YANG Y, et al. Transforming the interactive segmentation for medical imaging[C]//International Conference on Medical Image Computing and Computer-Assisted Intervention. 2022: 704-713.

[127] DU W, SHEN H, ZHANG G, et al. Interactive defect segmentation in X-Ray images based on deep learning[J]. Expert Systems with Applications, 2022, 198: 116692.

[128] WANG G, ZULUAGA M A, LI W, et al. DeepIGeoS: a deep interactive geodesic framework for medical image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 41(7): 1559-1572.

[129] BOYKOV Y Y, JOLLY M P. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images[C]//Proceedings of IEEE International Conference on Computer Vision. 2001, 1: 105-112.

[130] BROMLEY J, GUYON I, LECUN Y, et al. Signature verification using a" siamese" time delay neural network[J]. Advances in Neural Information Processing Systems, 1993, 6: 1-8.

[131] RONNEBERGER O, FISCHER P, BROX T. U-Net: Convolutional networks for biomedical image segmentation[C]//International Conference on Medical Image Computing and Computer-Assisted Intervention. 2015: 234-241.

简历与科研成果

基本信息

许翔，男，汉族，1998 年 5 月出生，江苏东台人。

教育背景

2020 年 9 月 — 2023 年 6 月 南京大学人工智能学院 硕士

2016 年 9 月 — 2020 年 6 月 南京大学计算机科学与技术系 本科

攻读硕士学位期间完成的学术成果

1. **Xiang Xu**, Zhao Jian, Jianmin Wu, Furao Shen, "Switch and Refine: A Long-Term Tracking and Segmentation Framework," in *IEEE Transactions on Circuits and Systems for Video Technology*, Sep. 2022.
2. 葛轶洲, **许翔**, 杨锁荣, 周青, 申富饶. ''序列数据的数据增强方法综述," 计算机科学与探索, Jul. 2021.

攻读硕士学位期间参与的科研课题

1. 国家自然科学基金面上项目 ''基于深度感知增量式联想记忆神经网络的信息融合系统研究" （项目编号 61876076，课题年限 2019 年 1 月 — 2022 年 12 月），负责序列数据的数据增强相关问题的研究。