硕 士 学 位 论 文

基于卷积神经网络的目标跟踪技术研究

Researches on Object Tracking
Based on Convolutional Neural Networks

作 者 姓 名：_____

学科、 专业：_____

学　　号：_____

指 导 教 师：_____

完 成 日 期：_____

大连理工大学

Dalian University of Technology

大连理工大学学位论文独创性声明

作者郑重声明：所呈交的学位论文，是本人在导师的指导下进行研究工作所取得的成果。尽我所知，除文中已经注明引用内容和致谢的地方外，本论文不包含其他个人或集体已经发表的研究成果，也不包含其他已申请学位或其他用途使用过的成果。与我一同工作的同志对本研究所做的贡献均已在论文中做了明确的说明并表示了谢意。

若有不实之处，本人愿意承担相关法律责任。

学位论文题目：_____

作 者 签 名 ：_____日期：_____年____月____日

摘　要

目标跟踪技术是图像处理和计算机视觉领域的一个重要研究方向，无论在军事领域还是人们的日常生活中都有着广泛的应用，例如在军事领域中的武器精确制导、侦查预警、无人机飞行器等，以及民用方面的智能交通、机器人视觉导航、医疗影像诊断等众多领域都有目标跟踪技术的身影。目标跟踪是在连续的视频序列中随着场景的不断变化对某一特定目标实现状态估计的过程。在实际的跟踪环境中，由于成像条件的复杂性和场景的多样性，使得实现稳定有效的目标跟踪仍然面临一系列的困难和挑战。本文针对目标跟踪过程中的技术难点，围绕目标外观建模对目标跟踪技术进行了深入的研究，提出了两种基于卷积神经网络的目标跟踪算法。

本文针对传统目标跟踪算法中的人工构造特征表达能力不足、难以提取涉及语义信息的问题，提出了一种基于卷积神经网络多尺度表达的目标跟踪算法。该方法利用卷积神经网络可以自动学习图像中的深层语义信息的特点，结合拉普拉斯金字塔构建多个尺度的卷积网络结构。使用视频跟踪数据集对网络模型以参数共享的方式进行由粗到细的训练，从而获取对尺度变化更具鲁棒性的目标多尺度外观表达。最后结合多示例学习算法的优势，构建基于多尺度表达的多示例分类器来实现目标的在线跟踪，并针对多示例算法易饱和的问题，对多示例算法进行改进。该方法使得目标的外观模型对目标变化以及尺度变化更加鲁棒，可以实现更加稳定的跟踪效果，提高了算法的准确率和成功率。

本文针对目标跟踪过程中的目标变化导致的漂移现象，提出了一种基于 Attention 机制的目标跟踪算法。该方法将视频序列的初始帧内容作为记忆单元，使得网络学习始终保持对初始帧目标特征的记忆，并根据视频帧之间的内容关联，利用初始帧设计 Attention 层次结构，构建基于 Attention 机制的卷积网络模型，通过学习使得网络自动关注目标中的关键位置。从网络的不同层中提取特征并进行金字塔空间池化处理，构建多专家分类器实现目标的在线跟踪。实验结果表明，该方法可以充分考虑目标的初始帧信息以及关键位置，有效缓解在跟踪过程中由于目标变化出现的漂移现象，实现在多种场景下进行稳定有效的跟踪。

关键词：目标跟踪；特征提取；卷积神经网络；Attention 机制

Researches on Object Tracking Based on Convolutional Neural Networks

Abstract

Object tracking is considered as an important issue in image processing and computer vision area. It has been widely used in both military fields such as precision guided weapons, detection of warning, unmanned aircraft and civilian fields such as intelligent traffic, robot navigation and medical imaging diagnosis. Object tracking is a process of estimating the state of a specific target in a continuous video sequence. Due to the complexity of the imaging conditions and the diversity of the scene, it is difficult to achieve the stable and effective tracking algorithm. According to the technical difficulties during the object tracking, this paper makes a deeply study on how to model the object appearance, based on which we propose two novel tracking algorithms based on convolutional neural networks.

In traditional object tracking methods, the hand-crafted selection of feature representations would limit the performance of visual tracking and the predictability power of semantic information of the object. To address this problem, this paper proposes a new algorithm for object tracking based on multi-scale representation of convolutional neural network. We make a multi-scale convolutional network architecture generated from the Laplace pyramid. Each scale is trained from coarse to fine using video datasets and shares weights across all layers. To solve the saturation problem, we improve the multiple instance learning method and train an improved MIL classifier with the multi-scale feature representations to realize the online tracking. This makes the appearance model more robust to changes of the object, which can achieve more stable tracking effect and improve the tracking accuracy and success rate.

To effectively alleviate the "drift" phenomenon caused by the appearance changes, an object tracking algorithm based on attention mechanism is proposed. The initial frame is regarded as the memory unit to make the network maintaining feature memory of the first frame. We design an attention layer with regard to the first frame according to the content correlation between video frames and construct a network based on attention to learn the importance of the image. From different layers, we extract features by spatial pyramid pooling and build multi-expert classifiers to achieve the target. Experimental results show the method can effectively alleviate the drift phenomenon, and achieve stable tracking results in a variety of scenes.

Key Words：Object tracking; Feature extraction; Convolutional neural network; Attention mechanism

目　　录

1 绪论

1.1 课题背景及意义

人类在对外界事物进行判断时，感知是最为重要的信息来源，其中包括触觉、味觉、视觉等，通过此类信息的支持，人类的思维可以与外界环境实现交互与连接。在这复杂的信息中，视觉信息是其中最为主要的信息来源，据统计，人类一半以上的信息是通过眼睛来获取的。然而，在很多领域，由于人的精力有限，单纯依靠人眼进行信息获取是很受限制且低效的，所以这种简单依靠人类视觉来完成各种信息获取任务的情况往往不够可靠。

随着数字计算机技术的飞速发展，研究人员一直致力于如何让计算机代替人眼完成相应的信息获取和处理工作，也就是说，如何使计算机、摄像机像人眼一样对目标进行感知、识别和追踪，并完成后续的图像信息处理工作。目标跟踪是其中最为主要的任务分支之一。先进的视频设备的出现，以及对自动化和智能设备需求的增长，使得越来越多的研究人员投入到目标跟踪技术的研究当中。

其中最主要的就是实现对视频图像序列的跟踪。该任务首先需要在视频序列中通过目标检测或者直接人工标记的方式，给定人们感兴趣的目标区域，然后在接下来的视频序列中标定出该区域的位置。在如今视频监控等设备的广泛应用环境下，视频序列已经成为巨大的数据源，其中出现的目标特征及其相应的运动场景蕴含着丰富的信息，因此对此类目标等进行识别、分类、跟踪等处理有着广泛的研究价值和应用潜力，主要包括以下几个方面：

（1）智能视频监控：在动态场景下对兴趣目标或者异常事件进行监测和分析。智能视频监控通常包括两种任务，一是实现场景解读和行为描述以获取目标事件的深层信息，二是实现广域范围内的实时自动视频监控任务以解放人力。随着围绕视频分析领域的大量革新数字产品与系统概念的提出，能够满足运营需求及协调方案的智能视频监控系统成为如今安全应用的关键因素，大面积的公共区域如银行、商场、停车场、车站等多种场所都会提供相关的基础设施作为安全管理或者行为分析的一种独特方法。例如：INMOVE[1]的目标之一是实现对足球场地中的球员位置进行实时跟踪以分析球员的体能或者战术；W4 视频管理系统[2]能够在人流密集的出入口位置实现对人的行为分析与追踪，对来源不明的可疑物品与人员进行判断识别。

（2）智能交通：为了提高交通系统安全性、生产力和环境友好性，智能交通系统被应用到运输与基础设施之间实现信息转换，包括道路交通安全运输、流动性管理、车辆

辅助系统等方面。例如，为了能够让道路更加安全，许多汽车设备制造商和供应商都在致力于发展高级车辆辅助系统（ADAS）[3], 可以自动完成对车辆、行人的检测，防止碰撞事故的发生。Tai 等[4]提出了一种能够对道路交叉口进行车辆监测和事故检测的实时图像跟踪系统，以合理规划道路使用和管理。

（3）视觉导航：在视觉控制领域中，以视觉导航为基础的移动机器人不再是新鲜事物，其中包含目标定位、自动地图构建、路径跟踪以及危险预警等多种涉及视觉跟踪的技术应用。其中一个重要的研究方向是机器人的自主行为导航[5]，通过摄像机、传感器等设备获取外部信号，使用多种图像处理技术、信号处理等方式对其进行处理，使其在一条合适安全的轨迹中运动。除了传统的地面机器人，近年来越来越多的无人机得到广泛的发展与使用。由于无人机自身荷载的限制，无法携带传感器等设备，使其更加依赖于基于视觉的导航策略。Van der Zwaan 等人[6]为无人机安装了类昆虫复眼的感光相机阵列，并且将他们连接到运动探测器中，使其能够计算出局部光流来确定位置进行导航。对于水下环境，由于其特殊的工作性质，视觉导航设备是一个优先考虑的选择。Antich 等人[7]利用电缆追踪算法即寻找电缆边缘来实现自主水下设备导航控制系统。

（4）军事精确制导：在科技化程度越来越高的现代战场上，军事精确制导对于军队的战斗力有着重要的作用。精确制导技术预先将目标图像载入弹上引导系统，在飞行过程中实时获取目标信息，并基于该信息对导弹上的引导系统进行操控，使导弹准确命中目标。在整个过程中，目标跟踪技术作为核心技术起到了决定性的作用。我国自主研发的近程反导武器系统以及航天航空设备上都有该目标跟踪技术的应用。

除此以外，目标跟踪还在人工智能、医学诊断[8]、视频检索等其他领域广泛使用，并发挥了良好的性能。由此可见，目标跟踪作为核心技术在多个领域都具有理论研究意义和应用价值。

1.2 国内外研究现状

目标跟踪技术在国内外研究人员的不懈努力下，出现了很多具有突破进展的算法。在文献[9]中，作者对近年来的跟踪算法作了非常全面的介绍，从各个角度对算法进行了的分类，并讨论了在应对特定场景环境时，应该采取的算法策略。国内的侯志强等人[10,11]也对视觉跟踪问题以及研究趋势进行了多个角度的论述。

通过分析跟踪任务的实质，多数算法在实现过程中关注跟踪任务的两个主要问题：一是构建目标的表示模型；二是设法保持目标模型的有效性。在跟踪过程中，当缺少足够的先验知识的情况下，静态模型往往很难应对可能出现的诸如几何形变、背景杂乱、

光线突变等问题，为了保持目标模型的鲁棒性，需要在后续的跟踪期间对参数进行更新，使得模型能够针对外观变化作出相应调整，而这也是一个跟踪算法能否成功的关键所在。传统的跟踪算法根据其原理大致由两类方式来实现，即生成方法和判别方法。基于生成方法的跟踪算法主要通过构建相关的模型来得到被跟踪目标的特征描述和外观模型，通过最小化重建误差来寻找最符合外观模型的目标区域。基于判别方法的跟踪算法根据前景和背景的差别构建目标的外观模型，除了需要跟踪目标的信息之外，通常还需要周围背景的信息来构建分类器来对前景目标和周围背景进行判别。下面分别对这两类跟踪算法的研究现状进行具体地阐述。

基于生成方法的跟踪算法通过学习一个外观模型来表征目标，通常可以使用某些生成过程来对目标进行描述。之后在接下来的每一帧中寻找与学习到的外观模型最相似的目标模型或者重建误差最小的目标区域。最常见的生成式目标外观建模算法包括稀疏表达[12,13]，模板匹配[14-16]，增强子空间学习[17]，密度估计[18,19]。例如，Liu 等人[13]提出的通过选出一系列稀疏的且区分能力强的特征，使其能够同时实现最小化目标重建误差和最大化判别能力来提高算法的有效性以及鲁棒性。但是由于区分特征的个数是确定的，对于周围场景不断变化的动平台跟踪，这种方法的有效性明显下降。为了适应目标变化，出现了多种对目标的外观模型进行动态更新的方法。Matthews 等人[16]提出了一种模板更新方法，通过与第一个模板进行校正来减小漂移。Kwon 等人[14]分别将观测模型和运动模型分解成多个基础观测模型和多个基础运动模型。每个观测模型表示目标的一种特定外观，每个移动模型表示目标的一种运动模式，以此来应对目标的多种姿态变化和外部环境的光线变化。

基于判别方法的跟踪算法是基于前景和背景的差别，设法构建一个能够将目标从背景中区分出来的外观模型，即将跟踪问题抽象为分类问题，我们将其称为一种 “检测跟踪”方法[20]。自适应检测跟踪方法主要包括以下过程：首先基于当前帧提取出目标样本的信息，利用在线更新的思想训练得到一个分类模型。接下来使用滑动窗口的方法在上一帧目标位置周围选取多个候选样本，利用分类器对每个候选框内容进行处理，获得判别分数最高即最有可能是目标的候选样本作为新一帧的目标位置。目前大多数成功的跟踪算法都是基于判别式的跟踪算法。例如，在文献[21]中提出了一种 P-N 学习方法，在训练分类器的过程中，使用了未标记的数据进行训练，并通过一对“专家”来评估未标记数据的分类结果，找出错误分类的正负样本并将其放入到训练数据中，依据错误样本增加了分类器的鲁棒性和判别性。Avidan[22]提出的集成跟踪利用集成学习方法训练多个弱分类器，每个分类器是一个基于最小二乘的线性分类器，并将其组成一个强分类器，之后分类器直接作用在每个像素点上得出一张置信图，置信图的峰值位置即为目标结果。

机器学习理论知识的应用也促使了一批判别式跟踪算法的出现。例如在线增强[23,24]，多示例学习[25]，朴素贝叶斯[26]，非线性支持向量机[27]，梯度提升决策树[28]，随机森林[29]等，这些算法都是通过预先标注好的训练样本来学习一个分类器，随后利用分类器实现对目标的跟踪。Babenko 等人[25]结合多示例学习提出的一种跟踪算法，在更新分类器的过程中，使用包含多个正样本的包作为样本集，在训练过程中，即使目标结果存在误差，样本包集中往往也包含了正确的样本，增加了样本选取的容错性，缓解漂移问题的影响。

近年来，随着硬件性能的不断优化与更新带动了计算机运算能力的大步提高。多核处理、并行计算等高性能计算的普及使得处理大规模复杂计算成为可能，同时也促使了深度学习在各个领域的发展，例如图像识别领域，语音识别领域，无人车技术等。2016 年，AlphaGo 所参与的一系列与顶尖棋手的对弈，背后起着重要作用的就是深度学习。其中，卷积神经网作为一种高效的识别方法在图像领域也逐渐被人们所重视。在 2012 年的大规模视觉识别赛中，Krizhevsky [30]等人利用了深度卷积网络模型实现分类任务，结果达到了 15.3%Top-5 错误率， 远远高于第二名所使用的传统方法的实验结果。接下来几年，卷积神经网络在目标检测领域快速发展，出现了 R-CNN[31]、SPP-NET[32]、Faster R-CNN[33]等。在图像跟踪领域，卷积神经网络也逐步进入研究人员的视野。文献[34]利用一个三层卷积神经网络在线学习目标特征表达，利用检测跟踪的方法实现对背景环境中的前景目标进行判别与区分，跟踪过程中没有对模型的离线预训练过程。Hong 等人[35]提出的在线跟踪算法，从一个已经训练好的 CNN 网络中，提取隐藏层的输出作为特征描述子，通过在线 SVM 学习目标的外观特征，然后对目标的显著图进行反向传播，获取最后的跟踪结果。Li 等人[36]利用多个卷积神经网络构成一个候选池，训练出多个候选外观模型以提供更多灵活性。但是文献中网络结构仅使用了两层卷积层，没有充分发挥卷积神经网络潜藏在更深层次的信息。Nam 等人[37]利用大规模标注完备的视频跟踪数据预训练一个卷积神经网络来获取一个通用的外观模型。所用的网络含有多个分支，每一个分支分别对应于一个视频域。在跟踪过程中，使用预训练的网络获取包含目标共性的域独立信息，然后重新构建某个视频序列的特定域层，并且在跟踪过程中在线更新网络结构。但是对于网络的更新往往很费时间，而且在跟踪失效的时候没有一个很好的机制去发现目标。除了使用传统卷积神经网络，近年来还提出利用全卷积网络实现跟踪的算法。Fan 等人[38]利用全卷积网络实现行人跟踪，将整幅图像帧输入到网络当中，利用一次前向传播进行预测，以节省冗余计算。Bertinetto 等人[39]提出了一个全卷积孪生网络结构，该结构包含两个输入，分别为模板图像和候选图像，利用相似性学习在候选图像区域中选出相似度最大的位置即为目标位置。

1.3 本文内容与章节安排

本文依托国家“十二五”规划中的重大科技专项项目“似神经网络系统采集及监控技术-基于内容的视频图像处理和人机交互技术”（2011ZX05039-003-3），以视频流的目标跟踪算法为主要研究对象，结合卷积神经网络的相关理论，深入研究最新的网络模型原理，分别提出了基于卷积神经网络的多尺度表达以及结合 Attention 机制的卷积神经网络两种方法，用于目标跟踪领域的研究。并通过对两种相关算法进行实验分析，验证了算法的有效性及应用价值。

本论文的具体内容安排如下：

第 1 章：绪论。本章首先从目标跟踪技术在多个领域的实际应用出发，介绍了本课题的研究背景及意义；接着对目标跟踪技术的国内外研究现状进行了介绍分析；最后对本文的具体工作内容以及章节安排进行了简要的说明。

第 2 章：目标跟踪技术概述与卷积神经网络。本章主要是对该论文中使用到的相关技术原理进行介绍。首先对目标跟踪技术的理论基础作了相关介绍；接下来介绍了神经元模型和传统的前馈神经网络；详细阐述了误差反向传播算法；最后重点分析介绍了卷积神经网络的各种层的类型以及结构特点。

第 3 章：基于卷积神经网络多尺度表达的多示例目标跟踪算法。本章在分析了以往目标跟踪算法外观模型使用人工构造特征的不足之后，提出了基于卷积神经网络多尺度表达的目标外观建模方式。首先利用拉普拉斯金字塔构建多个尺度的卷积网络结构，以参数共享的方式进行由粗到细的训练，获取对尺度变化更具鲁棒性的目标多尺度外观表达；并在此基础之上，改进了多示例跟踪算法，结合多尺度外观表达与改进的多示例分类器实现目标的在线跟踪；最后将算法与其他优秀算法进行比较，该方法使得目标的外观模型对目标变化以及尺度变化更加鲁棒，在算法的准确率和成功率方面都优于其他算法。

第 4 章：基于 Attention 机制的卷积神经网络的目标跟踪算法。本章针对目标跟踪过程中的目标变化导致的漂移现象，通过对神经网络最新技术的探索，提出了一种基于 Attention 机制的目标跟踪算法。该方法首先将视频序列的初始帧内容作为记忆单元，使得网络学习始终保持对初始帧目标特征的记忆，并根据视频帧之间的内容关联，利用初始帧构建基于 Attention 机制的卷积网络结构进行外观建模，使网络自动关注目标中的关键位置。并融合该模型的不同层次，构建多个专家分类器实现目标的在线跟踪，最后通过实验表明，利用初始帧信息和 Attention 机制可以充分考虑目标的原始外观以及关键位置，有效缓解由于目标变化导致的漂移问题，实现多种场景稳定有效的跟踪。

2 目标跟踪技术概述与卷积神经网络

2.1 目标跟踪技术概述

2.1.1 目标跟踪定义

目标跟踪过程可以定义为一个目标在特定场景移动时，在图像平面中的目标轨迹估计问题[9]。即对于一个视频序列，给定一个目标的初始状态，目标跟踪的任务是在接下来的每一帧视频序列中对目标状态进行评估，给出跟踪目标的一致性标签。在某些跟踪领域中，一个跟踪器可能还需要给出目标为中心的信息，例如目标的方向信息，位置信息以及目标形状信息等。通常，在通过摄像机或者录像机获取图像的视频序列的过程中，由于抖动、传输过程中的电子噪声等外部环境干扰因素，会造成图像模糊、出现噪声点分布的情况，所以，数字图像信息在进入模型之前，需要进行必要的预处理；对于当前帧图像，通常需要根据目标的运动模式给出目标的候选区域；然后针对每个候选区域进行特征的提取，实现候选区域的特征表达；由被跟踪目标的特征表达，来实现对目标特征的描述，进行外观模型的构建，并根据所提取特征来决策目标位置；由于被跟踪目标在整个数据序列中会不断的变化，其相应的特征也会产生更新，因此也要求模型进行相应的调整。图 2.1 所示就是目标跟踪的基本流程。

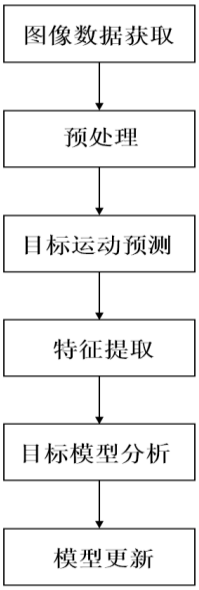


图 2.1 目标跟踪的基本流程

Fig. 2.1 Object tracking process

2.1.2 目标跟踪任务的难点

现有的大多数目标跟踪算法可以在可控的环境下有着较好的性能，但在面对复杂的环境背景、未知多变的目标运动，想要完成稳定、可靠的目标跟踪任务仍然存在很多挑战，主要的难点如下：

（1）遮挡问题

在目标的运动过程中，会遇见各种未知多变的环境，周围的环境物体会对所跟踪的目标形成遮挡。当发生部分遮挡时，大多数的算法无法预知是目标本身的变化还是周围物体的遮挡，影响现有模型的有效性；当发生全部遮挡时，无法提取出目标特征，造成结果误判或者丢失跟踪，对于长时间遮挡，如何恢复目标跟踪也是任务的难点。

（2）目标变化和视角变化问题

在目标跟踪的过程里，目标在姿态上会产生繁复的变换，出现目标外观前后不一致的情况。对于不同视角下的目标状态，所呈现出的外观形状、纹理结构、尺度大小往往会有很大差异，目标变化的未知性、多样性给目标跟踪任务增加了很大难度。

（3）复杂的目标运动问题

通常在跟踪任务中，都会对目标跟踪问题进行简化，对目标的运动模式做出限制，即假定目标在相邻帧之间只发生微小的位移变化。实际的跟踪目标往往不会呈现单一的运动模式，目标的快速移动、急走急停通常需要多种运动模型的叠加，这些都增加了目标跟踪任务的复杂性，造成对目标的运动模式难以建模，以及搜索策略的失效问题。

（4）环境光线变化及复杂背景问题

环境的光照变化是经常发生的现象，当周围环境发生光照变化时，会给目标外观和周围背景带来很大变化，影响现有外观模型的有效性，当周围背景随着时间推移发生变化，或者周围背景与目标相似易混淆的情况下，都会对跟踪造成困难。

（5）实时性处理需求问题

在实际的应用中，算法有着严苛的使用要求，其中最重要的就是实时性。例如在智能视频监控中，需要对意外突发情况及时发现实时跟进，确定目标位置信息，而在军事领域，这方面则更为严苛。为了保证算法有效性，既要达到精准性、鲁棒性，又要兼顾算法的复杂度与运算规模，往往使得跟踪算法陷入两难的境地，给设计准确高效的目标跟踪算法带来更多的困难。

本文分别提出了基于卷积神经网络的多尺度表达与结合 Attention 机制的卷积神经网络两种方法，用于目标跟踪领域的研究，并在效果上有显著提升。其中基于卷积神经网络的多尺度表达的方法针对目标的目标遮挡、尺度变化等问题，实现了被跟踪物体的多尺度、多层次的特征表达，从而保持了外观模型的稳定性与鲁棒性。而结合 Attention

机制的卷积神经网络模型通过对目标外观特征的记忆保持机制，使得模型在处理跟踪的过程中，能够对目标的关键位置实现准确定位，实现目标特征的记忆，有效地改善了由于目标跟踪的多种难点造成的错误跟踪导致的漂移现象。

2.1.3 目标跟踪外观表示

在目标跟踪问题中，如何有效表示目标外观是跟踪问题的基础，也是决定了目标跟踪算法能否成功的关键步骤之一。下面对目前已有的多种目标外观表示方法进行介绍。

（1）目标概率密度表示方法

目标的概率密度估计可以通过传统的参数化的方式来建立目标的外观模型，例如高斯函数和混合高斯函数。在多示例跟踪[25]中，就是利用高斯函数的参数特征来构建目标和背景的外观模型。也可以通过非参数估计概率密度的方式，通过从特定的形状区域中计算目标的外观概率密度，例如 Parzen 窗估计[40]和直方图。在文献[41]中采用在椭圆形目标块中，构建颜色直方图特征从而实现对目标外观的建模。

（2）模板表示方法

通过使用简单的几何形状或者轮廓剪影来描述目标的模板表示。模板表示方法的优点是在模板中既涵盖了目标的空间信息，也包含了目标的外观信息。但是在构建模板表示时，只是针对单个视图来对目标外观进行编码，只包含有限的建模能力，因此，它们只适合在目标对象姿态变化不大的情况下进行跟踪。

（3）主动外观模型表示法

主动外观模型采用同时对目标对象的形状和外观进行描述的方式来建立模型，它将目标形状定义为一组有限的标记集合，这些标记既可以在目标对象的边界上，也可以在目标对象的内部区域。对于每个标记都会存在一个以颜色、纹理或者梯度大小等形式构成的外观向量。主动外观模型采取了统计分析的思路，实现对训练数据的训练过程，使用主成分分析（PCA）来建立形状和其他相关的外观模型。

（4）多视角模型表示法

多视角模型表示法对目标对象的不同视图进行编码。一种方法是通过给定视图生成子空间，也就是子空间方法来表达目标的不同视图编码，生成子空间的方法可以是主成分分析和独立分量分析（ICA）。另一种是通过训练一组分类器，例如支持向量机（SVM）和贝叶斯网络来得到目标的不同视角的外观模型。但是多视角外观模型的一个局限是需要事先获取目标的所有视图外观。

2.2 神经网络概述

在人体的大脑中大约存在着几百亿个神经元，这些神经元编织出的繁杂而又逻辑功能鲜明的网就是我们的神经系统。而在人工智能领域，人们一直致力于使计算机能够像人类神经系统那样进行信息交互、自我学习。人工神经网络技术就是利用大量的网络节点单元来模拟人脑中的神经元，通过某种方式将这些神经元相互连接共同组成一个网络系统，网络的广泛互连使其能够具备自主学习的能力，模拟出几乎任何组合的非线性关系，同时其清晰的结构给计算机的编程实现等带来了易用性，使其能够应用到模式识别、信号处理等多种领域。

2.2.1 人工神经元模型

在人类大脑中，一个典型的神经元细胞通过一系列称为树突的结构来收集信号，然后将产生的电波信号通过轴突传递到成千上万的分支当中。在每个分支的末端，有一个称为突触的结构将来自于轴突的活动转换成电效应。当一个神经元接收到的输入刺激达到一定阈值时，就会使其处于激发状态，并继续向它的轴突释放电波信号；当其低于一定阈值时，就会处于抑制状态。学习过程就是通过改变突触的有效性，以此来改变一个神经元对其它神经元变化的影响。

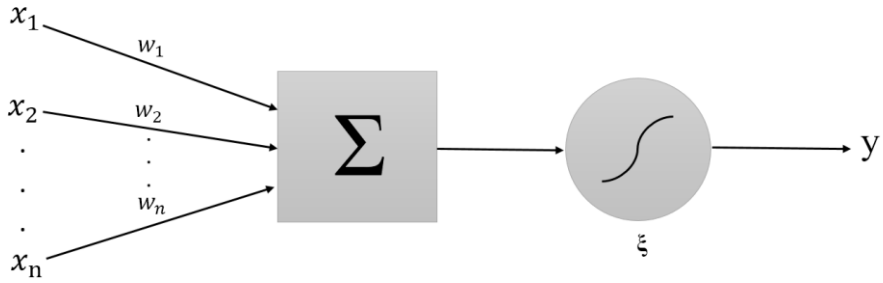


图 2.2 人工神经元模型

Fig. 2.2 Artificial neuron model

将人类神经元的工作过程抽象为一个数学计算过程，可以表示为如图 2.2 所示。通常在神经元的输入端，会有其他 n 个神经元与其连接进行信号输入，每个输入信号通过一个带有权重值的连接来调节每个输入信号的强弱，所有信号经过汇总之后与一个阈值进行比较，最后神经元的输出信号强度由一个激活函数处理产生。在图 2.2 中，x_i为神经元的第 i 个输入，ω_i为对应的连接权重，它们的加权和与一个给定阈值 ξ 进行比较，输出结果为 y，将它们表示为数学形式如下：

$$y = \varphi\left(\sum_{i}^{n}\omega_i x_i - \xi\right) \tag{2.1}$$

其中，φ函数为激活函数。激活函数可以是类 s 形状的非线性函数，也可以是分段线性函数或者阶跃函数。通常要求激活函数具有连续可微，单调递增有界的性质。在最初提出的人工神经元中，使用的是阶跃函数，其公式形式如下：

$$\varphi(x) = \begin{cases} 1, & x > 0 \\ 0, & x \le 0 \end{cases} \tag{2.2}$$

当输入信号的加权和大于阈值 0 时，则激励函数的输出为 1，意为对当前输入为正响应，否则为负响应。阶跃函数的特点是不连续，不可微，不利于数学分析，因此人们使用一个更为常用的与阶跃函数作用类似但是更平滑的 Sigmoid 函数来代替。Sigmoid 函数的函数曲线具有类 s 形状的特点，最典型的如图 2.3 所示：

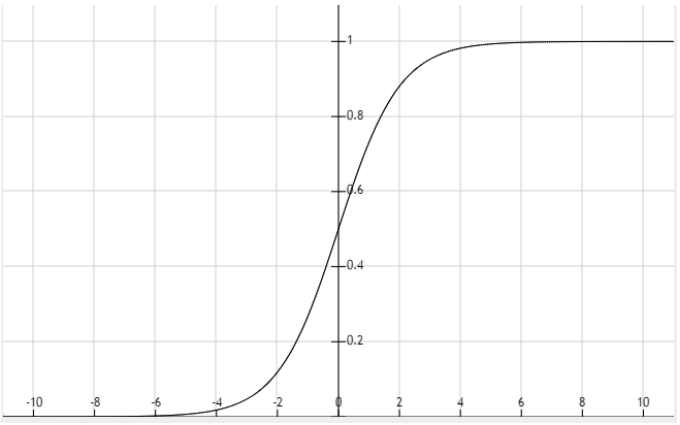


图 2.3　Sigmoid 函数

Fig. 2.3　Sigmoid function

从图中可以看出，Sigmoid 函数具有连续单调递增的性质，其导数形式也易于计算表达，这对神经网络的参数更新部分非常重要。Sigmoid 函数能够接收正负无穷区间的函数输入，但是只有有限的输出区间，即可以把实数范围内的输入值压缩到 0 到 1 之间，其公式如下：

$$S(x) = \frac{1}{1+e^{-x}} \tag{2.3}$$

与 Sigmoid 具有相似形状并能起到数据压缩作用的函数是双曲正切函数，它能够将实数范围内的函数输入限定在-1 到 1 之间。另外，由于对收敛速度和梯度保持的要求，

一些新的激活函数如 ReLU 等也被研究人员提出，在效果上有所提升，具体细节会在后面的章节进行描述。

2.2.2 前馈神经网络

将神经元依据不同的拓扑结构连接在一起能够得到不同的神经网络结构。如果将多个神经元按层组织连接在一起，就会得到前馈神经网络。典型的前馈神经网络如图 2.4 所示：

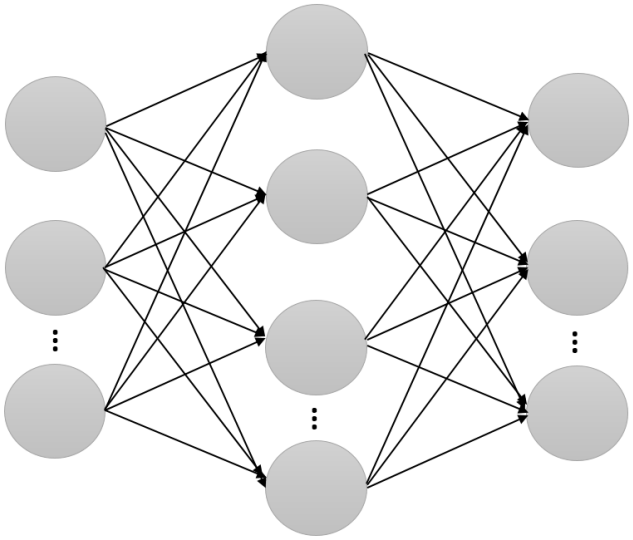


图 2.4 前馈神经网络

Fig. 2.4 Feedforward neural network

图中结构里的每个圆形节点是一个神经元，神经元之间的连线代表相应的权重。通常，在前馈神经网络中包含开始的输入层，中间的隐藏层和末端的输出层，每一层都是由若干个神经元节点组成，且同层的节点之间通常不能连接，相邻层之间单向连接，网络中不存在回路或者跨层次连接。信息通过输入层的每个输入馈送到中间的隐藏层，然后依次相前传递，隐藏层的每个输出作为后面一层的输入，直到获得整个网络的输出。

在神经网络中，重要的不是神经元，而是它们之间的连接权重。连接权重可以决定上一层每个输入节点对当前节点的重要性，确定当前节点的活跃程度，通过修改这些权重，节点单元可以选择处于何种状态。对于一组训练数据，神经网络就是通过不断调整连接权重来完成学习过程，也就是说网络中的连接权重包含了学习到的某种特定的数据模式。

2.3 误差反向传播算法

要实现多层网络的权值训练，需要大量的复杂计算。在 1986 年，Rumelhart 以及 Hinton 等科学家[42]对误差反向传播（Backpropagation，BP）算法进行了详细的分析，描述了如何利用训练数据的错误信息，从最后一层输出层到第一个隐藏层逐渐调整网络权重，实现网络学习的目的。误差反向传播算法也是目前为止大部分神经网络最为有效且常用的训练方法。它的主要过程分为以下几个步骤，一是前向传播过程，数据集通过中间隐藏层的逐层处理，最后通过输出层得到网络的输出；二是将网络的输出结果和真实结果作差值计算误差，并将误差逐层反向传播给之前的每一层，直至输入层；最后根据误差对网络中的连接权重进行修改调整，不断往复这个过程，达到收敛。

对于给定的一组训练数据集$D=\{(X_1,Y_1),(X_2,Y_2),\dots,(X_d,Y_d)\}$，$X_i\epsilon R^m$，$Y_i\epsilon R^n$，它们分别为 m 维输入向量，n 维的输出向量，即分别表示为$X_i=\left(x_1^i,x_2^i,\dots,x_m^i\right)$，$Y_i=\left(y_1^i,y_2^i,\dots,y_n^i\right)$。假设一个神经网络的输入层包含 m 个节点，中间的隐藏层包含 p 个节点，输出层包含 n 个节点。网络中第 i 个输入节点到第 u 个隐藏层节点之间的连接权重为v_{iu}，中间的第 u 个隐藏层节点与第 j 个输出层节点之间的连接权重为w_{uj}，b_u^h表示第 u 个隐藏层中的节点偏置项，b_j^y表示第 j 个输出层中的节点偏置项，非线性函数均采用 Sigmoid 函数。对于训练样本(X_k,Y_k)，经过前向传播过程，可以得到隐藏层的输出h_u^k和输出层的输出$\hat{y}_j^k$分别如下式所示：

$$h_u^k = sigmoid\left(\sum_{i=1}^{m} v_{iu}\cdot x_i^k + b_u^h\right) = sigmoid\left(net_u^h\right) \tag{2.4}$$

$$\hat{y}_j^k = sigmoid\left(\sum_{u=1}^{p} w_{uj}\cdot h_u^k + b_j^y\right) = sigmoid\left(net_j^y\right) \tag{2.5}$$

其中，net_u^h表示隐藏层 u 节点的所有输入，net_j^y表示输出层 j 节点的所有输入。

接下来计算网络的实际输出与目标值之间的误差，采用的损失函数为平方误差损失函数：

$$J = \frac{1}{2}\sum_{j=1}^{n}\left(\hat{y}_j^k - y_j^k\right)^2 \tag{2.6}$$

该损失函数实际上是一个关于输入样本(X_k,Y_k)和神经网络参数 $\theta=\left\{v_{iu},w_{uj},b_u^h,b_j^y\right\}$的多变量函数，其中需要进行调整的参数为 θ。以参数w_{uj}为例，损失函数对它的偏导数计算如下：

$$\frac{\partial J}{\partial w_{uj}} = \frac{\partial J}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial net_j^y} \cdot \frac{\partial net_j^y}{\partial w_{uj}} \tag{2.7}$$

由公式（2.6）可得：

$$\frac{\partial J}{\partial \hat{y}_j^k} = \hat{y}_j^k - y_j^k \tag{2.8}$$

对于 Sigmoid 函数，它的求导过程可以用它本身进行表示，即如公式 2.9 所示：

$$sigmoid'(x) = sigmoid(x)(1 - sigmoid(x)) \tag{2.9}$$

根据公式（2.5）和（2.9）可得：

$$\begin{aligned} \frac{\partial \hat{y}_j^k}{\partial net_j^y} &= sigmoid(net_j^y)\left(1 - sigmoid(net_j^y)\right) \\ &= \hat{y}_j^k(1 - \hat{y}_j^k) \end{aligned} \tag{2.10}$$

由公式（2.5）可知，net_j^y是关于w_{uj}的线性方程，则有：

$$\frac{\partial net_j^y}{\partial w_{uj}} = \frac{\partial\left(\sum_{u=1}^{p} w_{uj} \cdot h_u^k + b_j^y\right)}{\partial w_{uj}} = h_u^k \tag{2.11}$$

于是，将公式（2.8）、（2.10）、（2.11）代入公式（2.7）可得：

$$\frac{\partial J}{\partial w_{uj}} = \hat{y}_j^k(1 - \hat{y}_j^k)(\hat{y}_j^k - y_j^k)h_u^k = \delta_j^y \cdot h_u^k \tag{2.12}$$

类似的，可以推导出损失函数对参数v_{iu}，b_u^h，b_j^y的偏导数如下：

$$\begin{aligned} \frac{\partial J}{\partial v_{iu}} &= \frac{\partial J}{\partial h_u^k} \cdot \frac{\partial h_u^k}{\partial net_u^h} \cdot \frac{\partial net_u^h}{\partial v_{iu}} \\ &= \left(\sum_{j=1}^{n} \frac{\partial J}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial net_j^y} \cdot \frac{\partial net_j^y}{\partial h_u^k}\right) \cdot sigmoid(net_u^h)\left(1 - sigmoid(net_u^h)\right) \cdot x_i^k \\ &= \left(\sum_{j=1}^{n} \delta_j^y \cdot \frac{\partial net_j^y}{\partial h_u^k}\right) \cdot h_u^k(1 - h_u^k) \cdot x_i^k \\ &= \left(\sum_{j=1}^{n} \delta_j^y \cdot w_{uj}\right) \cdot h_u^k(1 - h_u^k) \cdot x_i^k \\ &= \delta_u^h \cdot x_i^k \end{aligned} \tag{2.13}$$

$$\frac{\partial J}{\partial b_j^y} = \frac{\partial J}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial net_j^y} \cdot \frac{\partial net_j^y}{\partial b_j^y} = \delta_j^y \tag{2.14}$$

$$\frac{\partial J}{\partial b_u^h} = \frac{\partial J}{\partial h_u^k} \cdot \frac{\partial h_u^k}{\partial net_u^h} \cdot \frac{\partial net_u^h}{\partial b_u^h} = \delta_u^h \tag{2.15}$$

为了调整网络中的权重参数，采用梯度下降法进行更新，即沿着优化目标的负梯度方向来寻找更优的参数解，需要选定一个学习率 α 来控制权重参数变化的幅度。学习率的大小会对网络收敛的快慢造成影响，当学习率取较大值时，网络中的参数会很快达到收敛，但可能会出现波动现象；当将学习率取一个很小的值时，会减慢网络中的参数收敛速度，需要更多的迭代次数。对于一个给定的学习率 α，参数w_{uj}的更新大小为：

$$\Delta w_{uj} = -\alpha \cdot \frac{\partial J}{\partial w_{uj}} \tag{2.16}$$

则根据公式（2.12）和（2.16）可得：

$$\Delta w_{uj} = -\alpha \cdot \delta_j^y \cdot h_u^k \tag{2.17}$$

类似的，对于参数v_{iu}, b_u^h, b_j^y，其更新大小分别为：

$$\Delta v_{iu} = -\alpha \cdot \delta_u^h \cdot x_i^k \tag{2.18}$$

$$\Delta b_j^y = -\alpha \cdot \delta_j^y \tag{2.19}$$

$$\Delta b_u^h = -\alpha \cdot \delta_u^h \tag{2.20}$$

2.4 卷积神经网络

2.4.1 卷积神经网络的发展历程

卷积神经网络是在人工神经网络的基础之上发展而来，在近年来的计算机视觉、自然语言处理、语音处理、自动控制等领域中取得了很多令人瞩目的研究成果，并实现了很多重要的商业应用。最初的卷积神经网络模型是在 1980 年，在文献[43]中提出的一个基于猫的视觉神经系统的仿生结构 Neocognitron。在结构中所采用的局部感受野思想以及权值共享、时空下采样方法，使得网络对目标变化具有一定鲁棒性，并且使得网络的参数规模显著减少。1989 年以来，LeCun 等人先后提出了一系列的卷积神经网络模型，被成为 LeNet。不同于 Neocognitron 使用无监督的学习方式，LeCun 等人采用有监督的学习方式对模型进行训练，使得模型可以对特定的任务进行优化，使其适用于实际的各种应用任务，在手写数字识别，物体识别，人脸检测和姿态估计以及自主机器人避障等领域都有很好的效果。但是限于当时内存和硬件条件，这些网络无法适用于更大尺度的图像和大规模的训练数据。随着 GPU 等设备在高性能计算领域的快速兴起，使得计算机的计算能力更加强大，并且出现了各种大规模数据集，例如 ImageNet[44]数据集和 MIT

Places[45]数据集，使得训练更大更复杂的网络模型成为现实。在 2012 年赢得了大规模图像分类大赛冠军的 AlexNet，在构建网络时采用了更多的层次结构，网络结构更加复杂，之后又相继出现了 ZF Net[46]，GoogLeNet[47]，VGGNet[48]和 ResNet[49]，这些网络被改进之后在不同领域发挥作用，使其在应用领域中大步发展。

2.4.2 卷积神经网络的结构

与传统的神经网络类似，卷积神经网络内部结构也是由若干个神经元组成，通过训练数据学习权重向量和偏置量。但是与传统神经网不同的是，卷积网络在它的结构中采用了局部感知、参数共享和网络池化等思想，使得结构更加复杂。在卷积神经网络中采用动物视觉神经的信息传递机理，即通过感受野来局部感知外部信息。在视觉系统的感知过程中，认知是一个从局部到全局的过程，对于某个关注点，仅仅在关注点的周围部分与之相关联，较远部分与关注点的相关性往往很弱。故在设计卷积网络的模型结构时，每一层的节点单元仅仅与上一层的部分节点单元进行连接，不需要对全部节点或者说全部图像进行感知，从而使得网络中需要学习的参数规模大大减小。另外通过权值共享的方式可以进一步减少网络中的节点数量，降低网络的计算复杂度。最常见的卷积网络的结构由卷积层-非线性变化层、池化层交替叠加组成，通常还会在卷积层的后面增加由激活函数构成的非线性变化层。如图 2.5 所示为其典型的网络结构。

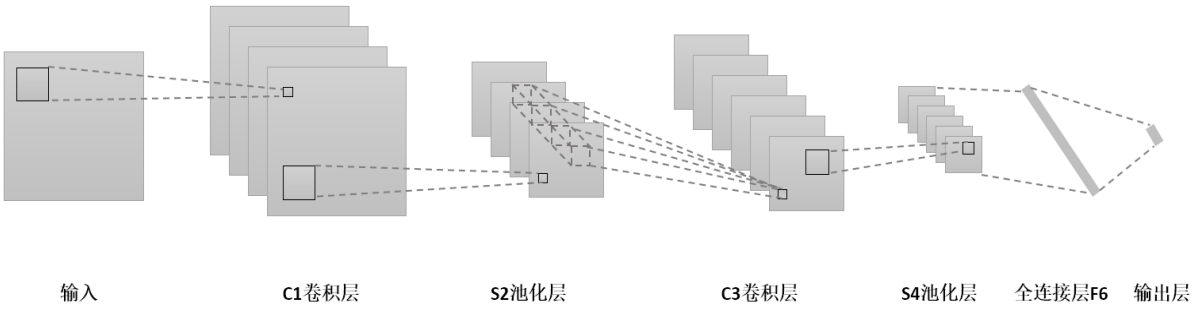


图 2.5 卷积神经网络

Fig. 2.5 Convolutional neural network

图中所示的卷积神经网络包含了输入层，卷积层，池化层，全连接层以及输出层。网络输入从第一层开始按顺序经过逐层处理，直到输入达到一定大小，然后在后面使用全连接层进行连接，从而使特征经过各层的提取后，最终生成类别标记的结果，最后一层全连接层通常会给出类别分数。具体的，网络中的节点首先通过一组滤波器也就是卷积核矩阵进行矩阵相乘得到下一层的网络输入，上一层每个位置的节点通常都会共享卷

积核中的权重值，每一个卷积核在所有位置滑动之后会在下一层形成一组节点单元，被称为特征图，多个卷积核就会得到多个特征图，通常随着层次的深入，特征图的大小会逐渐变小。接着网络中的池化层会对上一层的局部区域进行最大化或者平均化操作，在减少网络节点的同时能够最大程度的保留数据的最有价值信息，使得网络对于输入数据的微小位移变化具有不变性。最后根据不同的任务，通过全连接层将中间输出映射到不同的类别中。通过这样几部分的处理，从而搭建起卷积神经网络的整个结构。

2.4.3 卷积神经网络层的类型

介绍完卷积神经网络的整体结构之后，本节将对网络中的不同层次结构进行说明。

（1）卷积层

卷积层作为卷积神经网络的最核心部分，其卷积操作可以对输入数据进行特征提取。在网络结构中，第一层的卷积层通常能够提取底层次的特征如边线特征和角点特征；深层次的卷积层通常能够提取到更为抽象的高层次的特征。给定分辨率为 200×200 的二维图像，输入层就会产生 40000 个输入节点。如果中间的隐藏层有 20000 个节点，若采用全连接的方式，那么就会产生 $40000\times20000 = 8\times10^8$ 个权重参数，随着层数的增多，参数数量会急剧增涨，并且该种方式将图像看作一个一维向量，没有考虑二维图像的空间结构信息。卷积神经网络利用卷积操作实现了网络的局部连接，只对输入的局部区域进行连接，这样可以大大减少连接权重，并且可以对二维数据直接进行操作，很自然地包含空间信息。假设输入数据大小为 $N\times N$，有 H 个 $k\times k$ 大小的卷积核，滑动卷积核进行计算，每个位置的输出结果如下：

$$x_{ij}^{l} = \sum_{i}^{k}\sum_{j}^{k}\theta_{(k-i)(k-j)}x_{ij}^{l-1} \tag{2.21}$$

其中，$\Theta=\left[\theta_{ij}\right]_{k\times k}$ 为卷积模板，x_{ij}^{l-1} 为上一层的输出，H 个卷积模板将会产生相互独立的 H 个特征图，每个特征图的尺寸为$(N-k+1)\times(N-k+1)$。如果将公式 2.21 中的乘积累加操作用*表示，卷积层的输入有多个特征图 $X_1, X_2, \ldots, X_i$，卷积核的模板权重表示为 Θ_j，则对应卷积层的输出为：

$$O_j = act\left(\sum_i X_i * \Theta_j + b_j\right) \tag{2.22}$$

其中，act 为激活函数，b_j 为偏置量。在卷积核的滑动过程中，实际上是在所有位置共用同一个模板参数，也就是权值共享机制，它的理论基础是，在同一个数据环境下，如果

某个模板特征能够反映某个局部空间的属性，则它对于其他空间位置往往也是有效的。参数共享的思想同样能够大大减少需要学习的权重参数。

（2）池化层

池化层也叫做下采样层，是卷积神经网络的另一个重要组成部分。池化层会减少中间隐藏层的节点数量，减小网络的计算复杂度，并且能一定程度上抵制输入数据的噪声扰动和微小位移的影响。通常有两种操作来实现网络的池化层：最大池化操作和平均池化操作。在这两种情况下，输入数据被划分成不重叠的二维空间。如图 2.6 所示，输入大小为 4×4，池化窗口的大小 2×2，输入数据被分成 2×2 个不重叠的区域。对于平均池化来说，每个区域的平均值即为输出值；对最大池化来说，每个区域的最大值为输出值。池化层通常添加在卷积层的后面，二者交替出现。

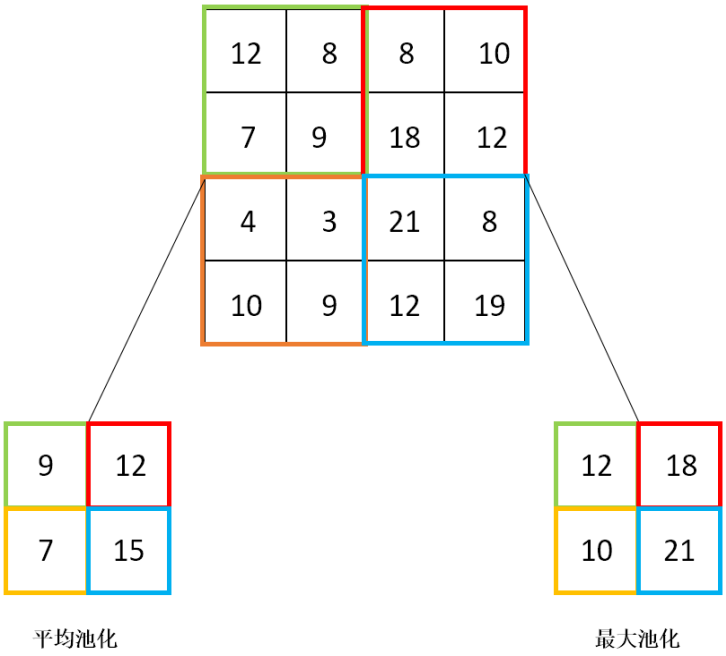


图 2.6　平均池化和最大池化

Fig. 2.6　Average pooling and max pooling

（3）非线性变化层

在一般的神经网络和卷积神经网络中都需要在网络中加入非线性操作，这是因为级联的线性系统仍然是一个线性系统，非线性操作能够确保网络具有更强的非线性能力。理论上，没有一个非线性函数要比其他非线性函数更具表达能力，只要它们是连续、有界且单调递增的函数。传统的前馈神经网络使用 Sigmoid 函数$\sigma(x) = \frac{1}{1+e^{-x}}$或者双曲正

弦函数$\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$。而在近几年的比较成功的网络中，更多的是采用 ReLU 修正线性单元，其数学形式如下：

$$ReLU(x) = max(0, x) \tag{2.23}$$

与采用其他非线性函数的神经网络相比，使用 ReLU 的网络具有更快的收敛速度。且由于 ReLU 的数学特性，梯度在传递过程中，能够很好的进行保持，从而有效地改善了梯度消失的现象[51]，且对模型的泛化能力不会带来明显的影响。最近，在文献[52]中介绍了一种称为 Leaky-ReLU 新的非线性函数，即：

$$Leaky\text{-}ReLU(x) = \max(0, x) + \alpha\min(0, x) \tag{2.24}$$

这里的 α 是一个超参数，这样避免了在进行梯度计算的时候，ReLU 可能出现的神经元梯度永远为 0 的情况。而文献[53]对参数 α 的设置作了进一步改进，通过训练方式来学习参数 α 的大小，使得模型能够更好的收敛，为模型的构建提供了更多的灵活性。

（4）全连接层

全连接层通常位于卷积神经网络的顶层。与传统的前馈神经网络类似，前一层的所有神经元经过权重矩阵的运算，与全连接层的神经元的全部进行连接。全连接层不再具有空间结构性，通常为一个一维向量，如图 2.6 中的全连接层 F6 所示，故在全连接层之后不会再出现卷积层结构。

（5）损失函数层

损失函数层是卷积神经网络的最后一层，它会给出输出类别的概率分布，其概率和为 1，之后计算网络输出与真实值的误差以进行“惩罚”，使整个网络的损失最小。对于不同的任务属性我们可以自行设计相应的损失函数。Softmax 损失函数通常应用于多分类任务中，而对于多个独立的二分类任务可以使用 Sigmoid 交叉熵代价函数。

2.5 本章小结

本章首先对目标跟踪技术的定义、难点以及常用的外观表示进行了简要介绍；接下来对传统神经网络的人工神经元模型和前馈神经网络作了相关介绍，并详细描述了误差反向传播算法；最后介绍了卷积神经网络的发展历程，对卷积神经网络的结构特点以及层级结构进行介绍分析，为接下来的内容提供理论基础。

3 基于卷积神经网络多尺度表达的多示例目标跟踪算法

3.1 问题描述

在解决跟踪问题的过程中，需要对目标区域构建外观模型，这是目标跟踪算法的一个重要环节。在实际的目标跟踪任务中，存在很多现实困难，例如物体遮挡、尺度变化、目标形变等情况，会改变目标原本的外观信息，因此，能否对目标区域构建稳定有效的外观模型决定了算法的性能好坏。

以往的大多数跟踪算法依赖人工构造特征构建目标的外观模型，对目标的本质特征表达能力有限，难以提取深层次的语义信息，特征的好坏往往取决于经验和大量的调试，尤其在复杂条件下，会对目标的外观模型的表达能力造成局限，造成目标外观模型的失效。另外，由于目标在跟踪过程中会随着移动过程，呈现多种视角的状态，其尺度的大小会出现前后不一致的情况。针对这些问题，本章在以往卷积神经网络在目标跟踪领域的研究之上，提出了基于卷积神经网络的多尺度表达的多示例目标跟踪算法。该算法针对目标的尺度变化，利用拉普拉斯金字塔构建多个尺度的卷积网络结构，通过参数共享的方式进行由粗到细地训练，利用卷积神经网络的自动学习深层特征的能力，获取涉及语义并且对尺度变化更具鲁棒性的多尺度外观表达，构建目标的判别式外观模型。与此同时，本章节还结合多示例学习算法的优势，将其作为算法的在线更新部分。但是由于其模型函数自身容易饱和，使得模型的区分能力下降，对跟踪性能造成限制。针对此问题，我们对模型增加了相应的惩罚项对其进行改进，并使用多尺度外观表达代替原始的以人工构造特征为基础的外观模型，构建在线跟踪器，实现目标的稳定跟踪。

3.2 算法描述

在分析某一个图像场景的情况下，算法通常无法事先了解感兴趣目标图像的尺度大小，因此为了能够考虑到目标图像的所有可能尺度，以便获取目标图像的最佳尺度描述，使算法自动适应或预测出目标图像的尺度大小，首先利用拉普拉斯变换将图像分解成多个尺度大小。之后针对每个尺度，构建不同尺度大小的卷积神经网络。接下来使用带标注的视频序列数据集对网络进行由粗到细的预训练。并利用训练好的网络提取目标的多尺度特征表达，利用多尺度特征表达构建改进的多示例分类器进行在线跟踪。算法的整体框架如图 3.1 所示，具体的步骤介绍如下：

（1）对图像做拉普拉斯变换，构建图像的金字塔空间，然后提取拉普拉斯金字塔的 3 种尺度下的图像作为网络模型的输入。

（2）针对每个尺度的图像搭建卷积神经网络模型结构，构成网络模型池。使用部分标准跟踪数据集对网络进行预训练，每种尺度图像对应一种网络，不同尺度间网络共享参数，尺度由粗到细进行训练。

（3）针对某一个视频序列集，将预训练的网络移除最后一层，添加一个随机初始化的 Softmax 层，利用视频序列的第一帧对网络进行微调。

（4）提取当前目标外观的多尺度特征表达，利用改进的多示例学习算法构建的多示例分类器得到下一帧的目标状态。

（5）利用新的目标状态更新多示例分类器，采用多步差模型更新方式更新网络，重复步骤 5，直到视频序列结束。

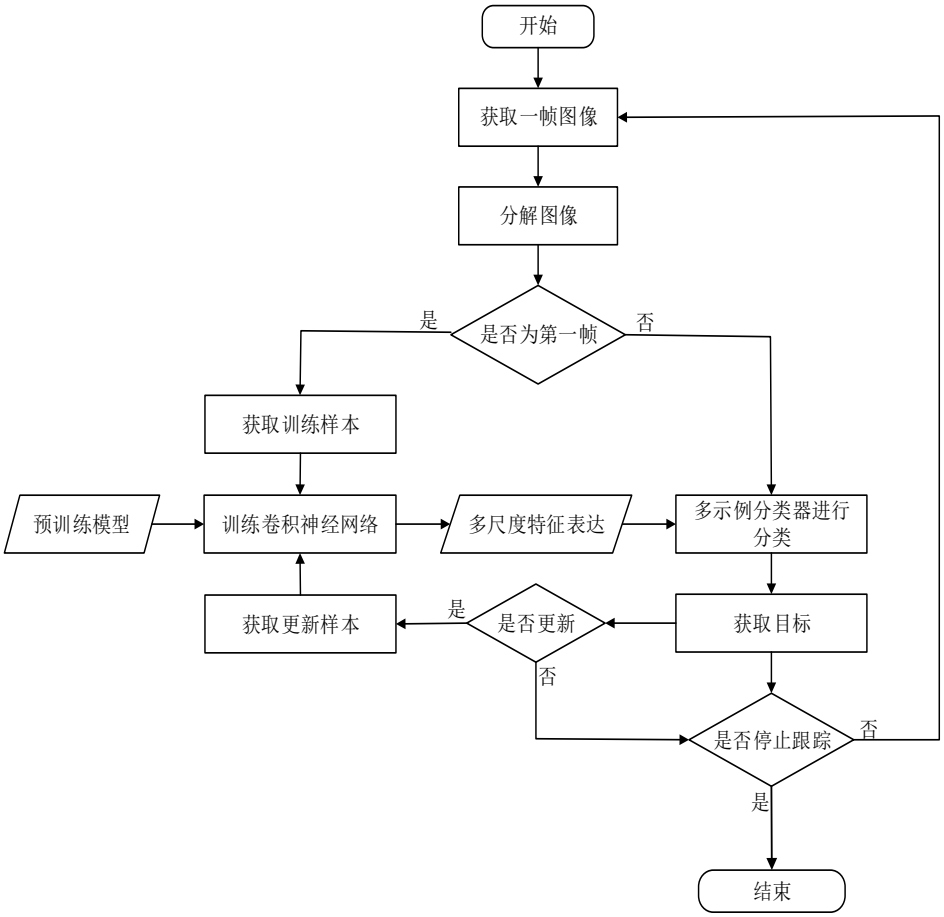


图 3.1　目标跟踪算法流程图

Fig. 3.1　The flow chart of the object tracking algorithm

3.3 卷积神经网络结构设计及目标建模

为解决被跟踪物体在运动过程中的尺度变化问题，首先利用目标的拉普拉斯金字塔对输入图像进行处理，得到图像的粗细不同的细节特征，然后利用处理后的图像作为不同尺度卷积网络结构的输入，通过跨尺度参数共享的方式进行训练，获取具有尺度不变性的特征表达。

3.3.1 目标的拉普拉斯金字塔构建

在一个真实的跟踪环境中，由于目标的运动影响，其尺度大小会出现前后不一致的情况。为了能够同时考虑不同尺度下的目标结构，消除尺度变化带来的影响，可以通过将图像与平滑核进行卷积构建尺度空间。文献[54]表明，高斯核和它的导数是唯一能够生成多尺度空间的平滑核。假设原始图像为$I(i,j)$，将它与可变核的高斯函数$G(i,j,\sigma)$进行卷积运算，得到图像的尺度空间$L(i,j,\sigma)$：

$$L(i,j,\sigma) = G(i,j,\sigma) * I(i,j) \tag{3.1}$$

其中，

$$G(i,j,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{i^2+j^2}{2\sigma^2}} \tag{3.2}$$

我们使用图像金字塔来更高效地构建尺度空间，主要包含两个步骤：滤波卷积和降采样操作。尺度空间会被分成 O 组，每组通过与原始图像不断进行卷积形成 K 个尺度图像，每组之后进行降采样使其分辨率为原来的一半，故对于某一层来说，尺度参数σ_s为：

$$\sigma_s = \sigma_0 2^{(O-o)+(K-k)/K} \tag{3.3}$$

其中，σ_0为初始尺度参数，$o \in \{1,\dots,O\}$，$k \in \{1,\dots,K\}$。重复这个过程即得到图像的高斯金字塔。在本文中，我们使用拉普拉斯金字塔作为我们的模型输入，通过对高斯金字塔的相邻层作差值即可得到：

$$\begin{aligned} D(i,j,\sigma_s) &= \left(G(i,j,\sigma_{s+1}) - G(i,j,\sigma_s)\right) * I(i,j) \\ &= L(i,j,\sigma_{s+1}) - L(i,j,\sigma_s) \end{aligned} \tag{3.4}$$

通过高斯核可以获取原始图像的低通滤波版本，在计算拉普拉斯金字塔之后，实际上是对图像作了不同尺度下的带通滤波，使得图像能够包含不同程度的图像细节。之后我们会使用这些不同尺度的图像来作为我们网络模型的输入。

3.3.2 卷积网络结构设计

在本章中，我们使用卷积神经网络来构成我们的网络模型，对数据进行有监督的训练学习。我们利用 Keras 深度学习框架搭建卷积神经网络模型结构，构成网络模型池。本章的网络结构采用了 VGG-S 网络的浅层网络部分，舍弃了深层的卷积层和全连接层，从而构成一个轻量级的卷积神经网络。具体的网络结构如图 3.2 所示：

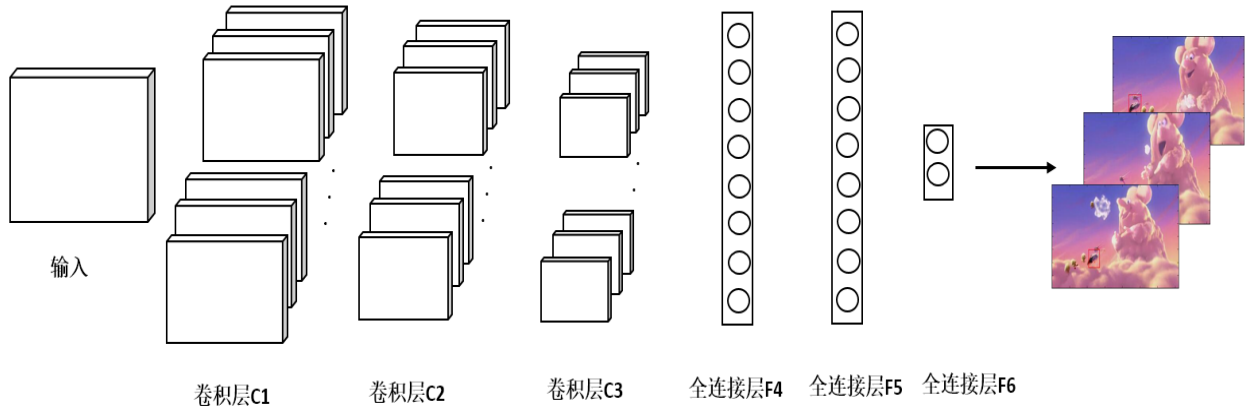


图 3.2 卷积神经网络结构

Fig. 3.2 The architecture of the proposed convolutional neural network

每一个网络模型包含三个卷积层，两个全连接层以及一个 Softmax 层。为了获得更好的训练效果，在本章中使用 VGG-net 的网络参数值初始化我们的网络权重。网络的输入是图像中用矩形框标注好的目标区域，将所有输入大小重新调整为 112×112，中间有 5 个隐藏层，分别为 3 个卷积层和 2 个全连接层。卷积层 C1 包含 96 个卷积核，每个卷积核的大小为 7×7，并在 C1 层的后面的增加了一个局部归一化层，用来对跨通道的像素点进行规范化，它的数学形式如下所示：

$$x_i = \frac{x_i}{\left(k + \left(\alpha \sum_j x_j^2\right)\right)^{\beta}} \tag{3.5}$$

其中，x_i是第 i 个通道的像素值，k 为设置的偏置量，α 和 β 为控制因子。在本章中，k 取值为 2，α 为 0.0001，β 为为 0.75。类似的，卷积二层 C2 包含 256 个卷积核，每个卷积核的大小为 5×5，卷积三层 C3 包含 512 个卷积核，每个卷积核的大小为 3×3。在各个卷积层之后，使用最大池化操作进行降采样，增加网络的鲁棒性同时逐渐降低维度。网络的顶层部分是两个全连接层，分别有 512 个神经元节点。最后一层为一个 Softmax

层，用来将中间的特征映射到一个分数向量，由于我们的目的是从背景中区分出前景目标，故使用一个二分类的分数向量分别表示前景和背景的分数。

3.3.3 预训练网络模型

由于视频跟踪序列的视频数据有限，我们利用已经训练好的网络参数对我们的网络进行初始化，我们使用的是在 ILSVRC-2012 数据集上训练的 VGG-S 网络参数。但是由于原始网络是用于视觉识别任务，考虑到跟踪任务与视觉识别具有本质上的差异，如果直接利用原始参数进行特征提取的话可能无法充分表达跟踪目标的外观性质，故我们会使用带标注的视频跟踪序列对网络进行再训练。此阶段的训练过程为离线的预训练过程。在预训练过程中，使用部分标准跟踪数据集不断优化参数。对于网络的输入，采用了三种尺度的输入图像，每种尺度图像对应一种网络，不同尺度间网络共享参数，尺度由粗到细进行训练。图像的粗尺度通常反映图像的整体结构，图像的细尺度包含较多的图像细节，这样可以同时关注细粒度的细节描述和粗粒度的整体描述。为了获取不同类别物体信息，针对不同类别视频集分别构建对应不同的网络，为捕获不同类别物体的共性特征，不同网络之间除最后一层外，其他层次共享网络参数迭代训练。整体的训练过程如图 3.3 所示。在网络的训练过程中，使用交叉熵作为损失函数，其数学形式如下：

$$L = -\frac{1}{n}\sum_{i}^{n}\left[t_i \log(p_i) + (1 - t_i)\log(1 - p_i)\right] \tag{3.6}$$

其中t_i为样本的真实标记，p_i为样本预测为目标的概率值。在训练过程中的参数更新阶段使用梯度下降法（SGD）不断优化参数。每次迭代时，首先向网络中输送某一类别的粗尺度的视频序列，在此基础之上，细尺度的继承权值继续训练，在此之后，使用另一类别的视频序列继续更新网络的共享层。重复这个过程，直到所有样本得到充分训练，最后我们保留三种尺度的网络模型参数。

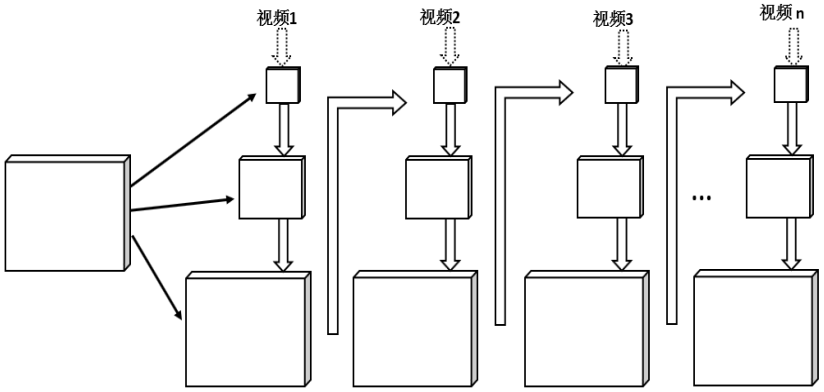


图 3.3 多尺度卷积神经网络训练过程

Fig. 3.3 The diagram of the multi-scale CNN network training process

3.4 改进的多示例学习分类器

多示例跟踪算法能够一定程度上解决在目标跟踪过程中出现的漂移现象，但是其分类器模型存在易饱和的问题，影响算法的性能。针对此问题，我们增加了相应的惩罚项进行改进，从而在保持函数收敛的情况下，有效缓解模型饱和问题。

3.4.1 多示例学习跟踪算法

在文献[25]中提出了基于多示例学习的跟踪算法。在更新分类器的过程中，如果使用单个正样本进行更新，由于跟踪的目标结果可能存在偏差，导致更新后的分类器也存在偏差，这种误差累计之后就会造成偏移。Babenko 提出的算法，使用包含多个正样本的包作为样本集，在训练过程中，即使目标结果存在误差，正确的目标也会存在于正样本中，增加了样本选取的容错性，在一定程度上解决了漂移问题。

取代传统学习方法的数据形式，多示例学习算法的训练数据形式为 $\{(X_1,y_1),(X_2,y_2),...,(X_n,y_n)\}$，其中，$X_i=\{x_{i1},x_{i2},...,x_{im}\}$ 表示一个包，x_{ij} 为每一个样本示例，y_i 为包的标签，具体定义为：

$$y_i = \max_j (y_{ij}) \tag{3.7}$$

其中 y_{ij} 是包中每个示例的标签，若在包中存在至少一个正示例，那么该包则会记为正标签，反之，则为负标签。在文献[25]中，采用了梯度提升算法来解决多示例学习问题。梯度提升算法通过最大化包的对数似然函数作为目标函数来训练一个增强分类器：

$$L=\sum_i \left(y_i \log p(y_i \mid X_i) + (1-y_i)\log\left(1-p(y_i \mid X_i)\right)\right) \tag{3.8}$$

其中，$p(y_i|\boldsymbol{X}_i)$为包的似然概率，通过 Noisy-OR(NOR)模型可以将其表示为每个样本示例的似然概率 $p(y_i \mid x_{ij})$ 的形式，即：

$$p(y_i \mid X_i) = 1-\prod_j \left(1-p\left(y_i \mid x_{ij}\right)\right) \tag{3.9}$$

对于样本示例的似然概率可以通过公式（3.10）进行计算：

$$p(y \mid x) = \sigma(H(x)) \tag{3.10}$$

其中，$\sigma(x)$为非线性变换 Sigmoid 函数，$H(x)$为多个弱分类器$h(x)$进行组合得到的强分类器。在训练分类器的过程中，算法会维持 M 个弱分类器 h。在接收新的样本时，会同时更新所有的弱分类器，之后，通过最大化公式（3.8），逐个选出 K 个最佳的弱分类器：

$$h_k = \arg\max_{h\in\{h_1,\ldots,h_M\}} L\left(H_{k-1} + H\right) \tag{3.11}$$

之后，通过将各个弱分类器加权求和得到一个强分类器 $H(x)$：

$$H(x) = \sum_{k=1}^{K} \alpha_k h_k(x) \tag{3.12}$$

在 MIL 算法中，弱分类器h_k由 Haar-like 特征f_k和四个参数$(\mu_1,\sigma_1,\mu_2,\sigma_2)$组成。在更新的过程中，会在线估计四个参数，分类器返回概率比值的对数，即：

$$h_k(x) = \log\left[\frac{p_t\left(y=1 \mid f_k(x)\right)}{p_t\left(y=0 \mid f_k(x)\right)}\right] \tag{3.13}$$

其中，$p_t(f_t(x)\mid y=1)\sim N(\mu_1,\sigma_1)$，类似的，$p_t(f_t(x)\mid y=0)\sim N(\mu_0,\sigma_0)$。令 $p(y=1)=p(y=0)$，根据贝叶斯公式可以推导出：

$$h_k(x) = \log\left[\frac{p_t\left(f_k(x)\mid y=1\right)}{p_t\left(f_k(x)\mid y=0\right)}\right] \tag{3.14}$$

当有新数据{(x_1,y_1),(x_2,y_2), … ,(x_n ,y_n)}到达分类器时，使用如下更新原则进行更新：

$$\mu_1 \leftarrow \gamma\mu_1 + (1-\gamma)\frac{1}{n}\sum_{i\mid y_i=1} f_k(x_i) \tag{3.15}$$

$$\sigma_1 \leftarrow \gamma\sigma_1 + (1-\gamma)\sqrt{\frac{1}{n}\sum_{i\mid y_i=1}\left(f_k(x_i)-\mu_1\right)^2} \tag{3.16}$$

其中，n 为新样本的个数，$0<\gamma<1$为学习率。对于参数μ_0和σ_0采用类似的更新方式。

3.4.2 改进的多示例学习算法

在多示例学习算法中，样本示例的似然概率通过 Sigmoid 函数进行计算，如公式（3.10）所示。根据 2.2.1 小节的介绍可知，Sigmoid 函数是一个非线性函数，能够将输入数据压缩至 0 到 1 之间。函数图像和其对应的导数图像如图 3.4 所示：

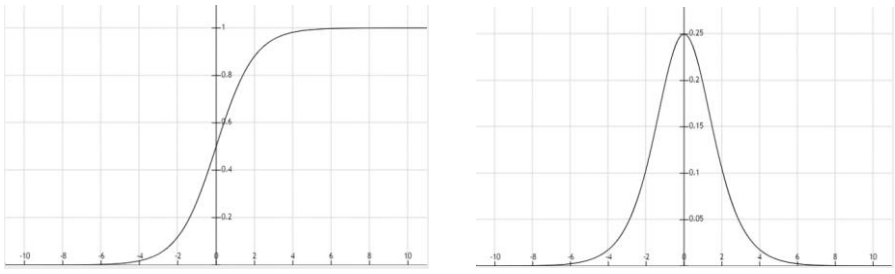


图 3.4 Sigmoid 函数和它的导数图像

Fig. 3.4 The plot of the Sigmoid function and its derivative

从图 3.4 的图左可以看出，图像的中部区域，也就是 0 值附近，图形很陡峭，变化很快，当 x 逐渐增大或逐渐减小时，函数很快趋近于 0 或者 1 达到饱和。从左侧的导数可以更加清晰的看出，当$|x|>6$的时候，函数的导数趋近于 0。分析公式（3.10）和公式（3.12）可以看出，在多示例算法的学习过程中，随着对弱分类器的选取以构成强分类器时，样本示例的似然概率很容易达到饱和，造成过拟合问题。为了解决这个问题，我们在 Sigmoid 函数中引入了一个惩罚因子来减缓函数饱和，改进后的 Sigmoid 函数如下式所示：

$$\sigma(x) = \frac{1}{1 + e^{\left(-x/\sqrt{k}\right)}} \tag{3.17}$$

其中，k 为组成强分类器的弱分类器个数。当弱分类器的个数逐渐增多时，惩罚因子可以快速抑制自变量的大小到一个合理的范围，减慢函数饱和的速度，同时能够确保函数收敛。

3.5 基于多尺度外观表达的多示例在线跟踪

该部分将以上处理方法进行融合，使用目标的多尺度外观表达代替多示例跟踪算法中以人工构造特征为基础的外观模型，构建在线分类器，并对模型的训练过程与更新策略进行了详细的介绍与分析。

3.5.1 多尺度外观表达

分析跟踪任务的几个重要步骤可知，目标跟踪算法的性能取决于对所跟踪目标的外观进行数学建模的方式，也就是如何建立一个目标的外观模型。在传统的跟踪算法中，大多是利用人工构造的特征来对目标的外观进行描述，但此类特征存在一个严重的缺陷，即：该特征可能只在特定场景下有着良好的跟踪性能，在面对实际跟踪任务中复杂的环境变化时，跟踪性能就会受到较大的影响。例如，对于颜色直方图特征来说，在色彩分明光线良好的情况下时，特征描述具有足够的区分能力，但是当目标处于光线阴暗或被阴影遮挡的时候，颜色直方图构成的特征描述可能就会失效。另外，自然图像中存在多尺度的结构信息，图像的粗尺度通常反映图像的整体结构，图像的细尺度包含较多的图像细节。另外，目标的尺度大小也会随着视角变化发生改变，这会给目标的尺度选择带来影响。

在本章中利用卷积神经网络的自动学习深层特征的能力，可以获取涉及语义信息的深层图像表达，同时利用拉普拉斯金字塔对图像进行多尺度分解，构建图像的多尺度表达，进而训练多尺度的卷积神经网络结构。该方法能够提取多尺度的卷积特征，构成表

达能力更强的外观模型。我们从三种尺度的卷积神经网络中提取卷积三层的特征为$\{T_i^k\}, i=1,2,3,\ k=3$，并将它们拼接在一起共同构成特征描述以对目标的外观进行编码。另外，随着网络层次的加深，模型会得到抽象的语义特征，但是由于该特征经过了网络中的池化操作，会造成空间结构信息的丢失。故对于精细尺度的卷积神经网络，我们同时还会提取卷积二层的特征$\{T_3^2\}$来表示目标的外观描述。故在本章中使用了不同尺度以及网络的不同层次的特征来对目标进行特征描述，即$[T_1^3;T_2^3;T_3^3;T_3^2]$，如此可以捕捉精细程度不同的细节描述，且底层的卷积特征包含了更多的图像中的局部结构特征，可以更好的帮助确定目标的位置。同时，在处理底层的卷积特征的时候，我们使用了一种新的池化方式，即行列最大池化，该池化方法分别保留行和列的最大值，从而尽可能的在保留特征信息的同时减少数据的维度。

3.5.2 跟踪算法流程及更新策略

在文献[25]介绍的多示例学习跟踪算法中，使用了 Haar-like 特征，该特征是由 2 到 4 个矩形区域的像素值进行运算得到的一种描述图像灰度变化的人工构造特征。在多示例学习跟踪算法中通过随机生成一组 Haar-like 特征来构成目标的特征描述。但是在多变的跟踪场景，如出现阴影的时候，Haar-like 特征不足以应对阴影对于图像灰度变化的干扰，易将外界阴影的变化误判为跟踪物体的灰度特征，从而对跟踪目标的识别产生严重干扰，丢失跟踪目标。在本章中，我们利用 3.3 小节介绍的多尺度卷积神经网络提取目标的外观描述来构成多示例学习跟踪算法中的特征空间。

在完成 3.3.3 小节描述的多尺度卷积神经网络的预训练过程之后，会得到三种尺度的网络模型参数。该模型没有直接用于在线跟踪任务中，我们移除网络的最后一层全连接层，仅保留网络前面的部分。每当开始跟踪一个视频序列的时候，我们在网络的最后一层重新添加一个随机初始化的全连接层。利用视频序列的第一帧给出的目标位置，对网络结构和新添加的全连接层采用 3.3.3 小节的训练方式进行参数微调。

在跟踪过程中，从 3 个尺度的网络中分别提取卷积三层的特征图作为卷积特征。同时提取精细尺度网络的卷积二层的特征共同组成外观模型的多尺度特征表达。之后，利用 3.4.2 小节介绍的改进的多示例学习跟踪算法进行在线跟踪。将得到的卷积特征作为特征池，利用多示例学习构建二分类器。当接收新的一帧图像时，我们在上一帧目标结果周围选取若干个候选目标$\{x_1, x_2, \ldots, x_n\}$，计算所有候选目标的似然概率$p(y|x) = \sigma(H(x))$，并按照公式（3.18）选取目标：

$$l_t^* = l\left(\arg\max_{x_i} p(y \mid x)\right) \tag{3.18}$$

在确定目标的位置之后，我们在当前目标周围按照 IoU 重叠率选取正负样本来更新分类器。另外，我们采用不同的更新策略对模型进行更新。在本章中我们采用多步差模型更新的方式来更新多尺度卷积神经网络结构。对于粗尺度的网络模型，采用快更新的方式来更新网络模型，以及时适应模型的外观变化；对于细尺度的网络模型，采用慢更新的方式来更新网络模型，能够避免模型改变可能引入的误差噪音和错误更新；对于中间尺度的网络模型，更新频率介于二者之间。通过这种方式，使得模型能够及时适应目标的外观变化，同时能够抵制错误跟踪对模型更新的影响。

3.6 实验结果与分析

本节是对基于卷积神经网络多尺度表达的多示例目标跟踪算法的实验验证与分析。本实验平台采用了 Ubuntu 操作系统，Intel Core i7-6700 3.40GHz 处理器，8GB 内存，配有英伟达 Tesla K40c GPU，使用 Python 和 Keras 深度学习框架进行仿真模拟。本章节采用的实验数据为目标跟踪标准数据库（OTB）[55]，在该数据集中包含了 100 个不同场景的视频跟踪序列，所有的视频跟踪序列都带有标注完好的目标位置。在视频数据中存在光照变化、尺度变化、物体遮挡、快速移动、运动模糊等在实际跟踪中可能遇到的问题。如图 3.5 所示显示的是部分视频序列的第一帧图像。图像中的红色方框即是待跟踪的目标。我们选取其中 60 个视频数据用来对模型进行预训练，剩余的视频数据则作为实验的验证数据集。

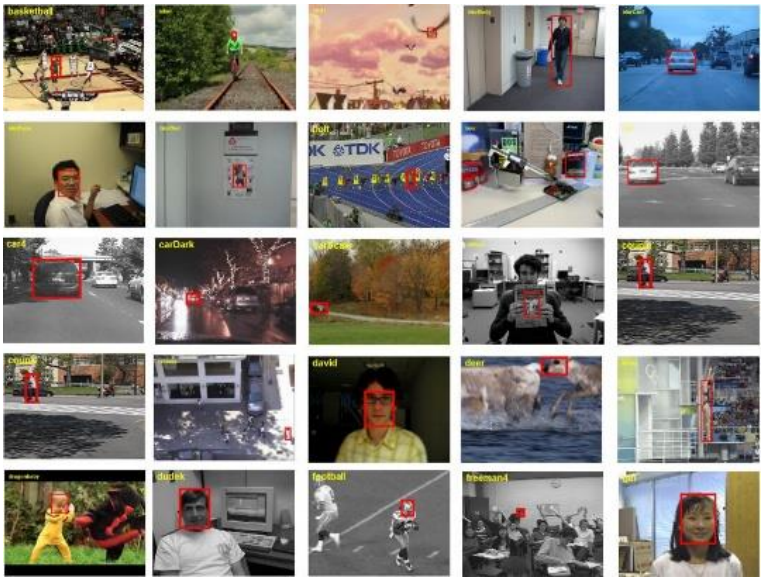


图 3.5 目标跟踪标准数据集的部分图像

Fig. 3.5 Sample images of the OTB dataset

为了验证本章提出来的目标跟踪算法的有效性，我们从两个方面对所提出算法进行分析验证。首先是跟踪算法的准确率，其次是算法的成功率。并选取多个经典的目标跟踪算法作为实验对照，它们分别为 MIL [25]，TLD [21]，Struck [27]，KCF [56]，SCT4 [57]，SCM [58]和 TGPR [59]。

（1）跟踪准确率

就算法的准确率方面，我们使用跟踪目标与真实位置的中心误差来评价算法的准确度，通过计算两者的中心点的欧氏距离来实现。具体方法是计算跟踪结果的中心位置与真实位置沿着 x 轴和 y 轴方向的坐标值的平方误差：

$$err = \sqrt{\left(x_t - x_g\right)^2 + \left(y_t - y_g\right)^2} \tag{3.19}$$

其中，x_t，y_t分别为跟踪目标中心位置在 x 轴和 y 轴方向的坐标值，x_g，y_g分别为真实目标中心位置在 x 轴和 y 轴方向的坐标值。该误差值越小，则说明跟踪算法的跟踪结果越准确，算法的性能越好。理想情况下是视频序列的每一帧的跟踪误差均为 0。为了更直观的描述算法在准确率方面的性能，我们设置不同的误差距离作为阈值，统计达到不同阈值要求的帧数占总帧数的百分比，并选取阈值为 20 个像素对应的百分比结果为最终分数，实验结果如图 3.6 所示。

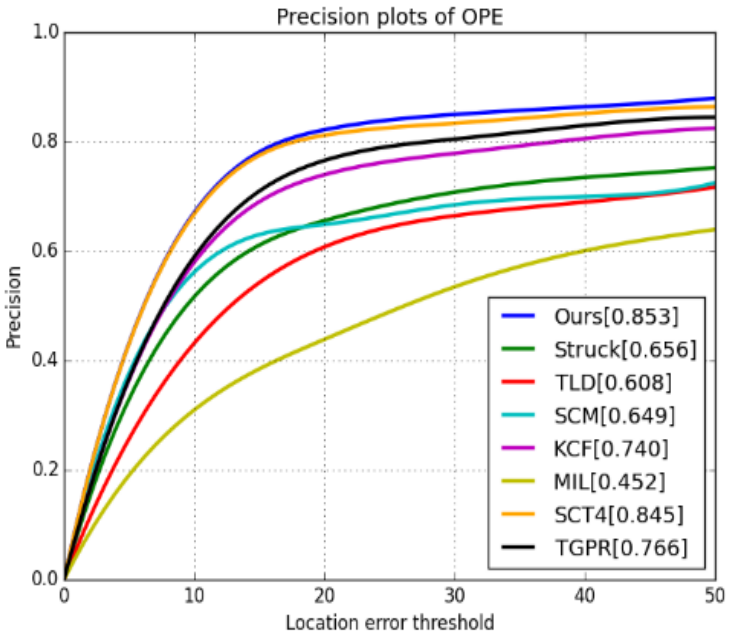


图 3.6　不同跟踪算法的准确率对比曲线图

Fig. 3.6　The precision plot of different tracking methods

（2）跟踪成功率

就跟踪算法的成功率方面，我们通过公式（3.20）所计算的跟踪目标和真实位置的重合率来评估跟踪算法的成功率：

$$S = \frac{|r_t \cap r_o|}{|r_t \cup r_o|} \tag{3.20}$$

其中，r_t为算法跟踪结果得到的目标面积，r_o为真实目标的面积，$\cap$代表交集操作，$\cup$代表并集操作。预先给定一个阈值，如果重合率大于这个阈值则认为跟踪成功，如果重合率小于这个阈值则认为跟踪失败，理想情况下是每帧的重合率均为 1。我们分别统计在不同阈值下的跟踪成功的百分比（即成功率），并以 AUC 面积大小作为最终分数。实验结果如图 3.7 的所示。

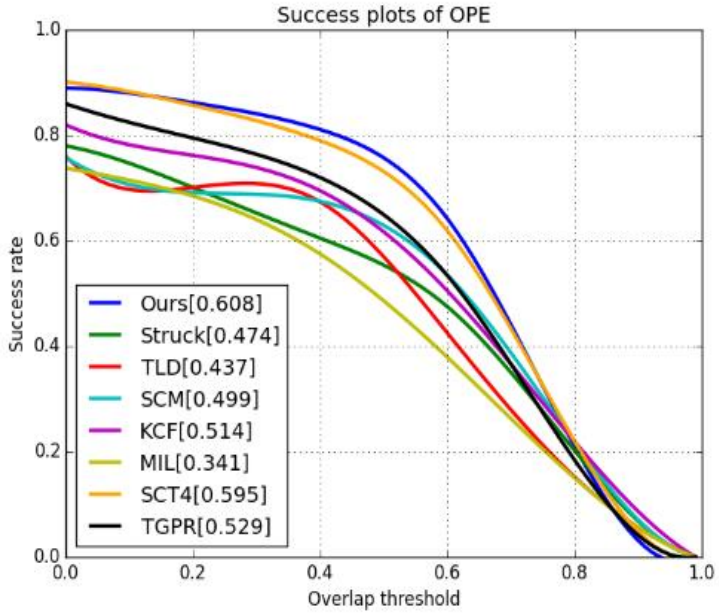


图 3.7　不同跟踪算法成功率的对比曲线图

Fig. 3.7　The success plot of different tracking methods

从以上实验结果中可以看出，本章提出的算法无论在准确率方面还是成功率方面都比其他算法具有更优的表现。在实验数据中存在多种不同的图像变化，例如，在视频序列 *Human5* 中，目标从远处逐渐移动到近处，由于拍摄距离的问题，导致目标的尺度大小发生明显变化。其他算法在跟踪过程中由于不能适应尺度变化而逐渐出现跟踪结果框严重不符甚至丢失跟踪的情况。本章提出的算法在学习目标的过程中考虑了多种不同的尺度变化，并使得多种尺度的模型权重共享，使跟踪器在跟踪过程中可以适应尺度变化，

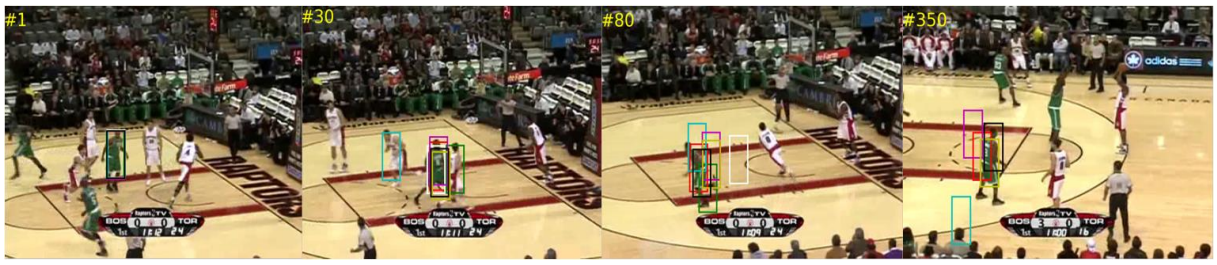
实现鲁棒跟踪。在视频序列 *Basketball* 中，大多数的跟踪器在初始阶段可以很好地对目标进行跟踪，但是当目标发生形变时一些跟踪器开始逐渐偏离目标甚至丢失跟踪。本章的算法能够在目标发生形变之后继续准确地跟踪，在整个跟踪过程中优于其他算法。

另外，我们还统计了部分视频序列的平均重合率，即计算跟踪器在某一视频序列的全部时间内的重合率的平均值。若平均重合率越大，说明结果越出色。结果如表 3.1 所示。从表中可以具体地看出本章提出的算法在大部分视频数据中都有更高的平均重合率，且在不同视频中表现稳定。在图 3.8 中展示了 *Basketball*，*Bolt2*，*Human5*，*Tiger1* 视频序列的部分视频帧的跟踪结果，从中可以更直观地看到，我们提出的基于卷积神经网络多尺度表达的多示例目标跟踪算法能够有效地对多种场景下的目标实现稳定跟踪。

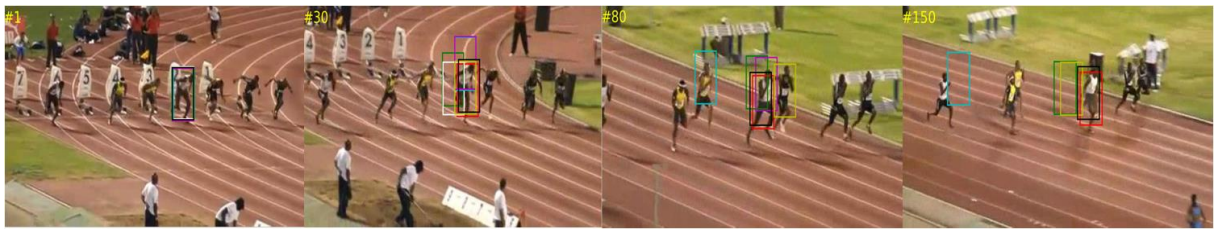
表 3.1　平均重合率对比数据

Tab. 3.1　Bounding box average overlap ratio

视频序列	MIL	TLD	Struck	KCF	SCT4	SCM	TGPR	本章算法
Basketball	18. 1	17. 2	18. 6	20. 3	38. 9	17. 5	22. 7	43. 2
Bolt2	9. 4	20. 5	33. 2	35. 7	42. 1	31. 8	39. 5	39. 7
Deer	35. 7	60. 8	83. 5	75. 1	79. 7	69. 0	79. 6	80. 9
Diving	37. 3	29. 7	33. 6	30. 9	42. 3	15. 1	11. 4	40. 0
Human5	30. 4	32. 6	38. 4	39. 1	54. 6	40. 3	52. 0	61. 5
MountainBike	14. 4	51. 9	66. 2	69. 5	69. 1	71. 4	62. 8	86. 6
Skiing	22. 1	19. 4	3. 2	9. 7	41. 0	7. 4	24. 3	50. 2
Tiger1	27. 8	3. 3	13. 9	8. 4	22. 7	5. 2	20. 2	25. 8
Woman	23. 6	38. 5	49. 7	42. 3	40. 1	38. 0	38. 6	62. 3



(a) *Basketball*



(b) *Bolt2*

(c) *Human5*

(d) *Tiger1*

图 3.8　不同算法在部分数据集的定性跟踪结果对比图

Fig. 3.8　Qualitative results of the proposed method on several test sequences

为了验证改进后的多示例跟踪算法的有效性，我们作了一个更加全面的分析。我们将添加的惩罚因子移除，并针对 *Basketball*，*David2*，*Tiger2*，*Deer* 和 *Woman* 视频序列与改进后的算法进行比较。类似的，利用阈值为 20 个像素时的跟踪准确率为评价标准，实验结果如表 3.2 所示。从表中可以看出增加了惩罚因子的算法在所有视频序列中都有更优的表现，例如在视频序列 *Deer* 中存在快速移动和轻微旋转的影响，当算法未改进时，由于函数内部易饱和的问题，不能对目标的外观进行充分学习，导致对目标的外观变化不敏感，跟踪的鲁棒性能下降。而在本章算法提出的改进之后，可以随着弱分类器个数的增加，自适应的减缓函数饱和的程度，从而可以使外观学习更充分，提高跟踪的性能。

表 3.2　阈值 20 的准确率对比

Tab. 3.2　The score of the precision at threshold 20

视频序列	Basketball	David2	Tiger2	Deer	Woman
本章方法	85. 32	84. 23	60. 76	86. 47	91. 62
本章方法（无惩罚因子）	68. 36	71. 54	59. 71	70. 48	80. 24

3.7 本章小结

本章提出了一种基于卷积神经网络多尺度表达的多示例目标跟踪算法。该算法利用卷积神经网络的自动学习深层特征的能力，可以获取涉及语义信息的深层图像表达，同时利用拉普拉斯金字塔构建图像的多尺度表达，设计了多尺度的卷积神经网络结构。在训练过程中提出了在不同尺度间参数共享并实现由粗到细的训练方式。利用标准视频数据集对模型进行预训练，提取多尺度的卷积特征，构成表达能力更强的外观模型。同时结合改进的多示例学习算法，解决模型易饱和造成的模型区分能力下降的问题。与现有的目标跟踪算法相比，该方法能实现更加稳定的跟踪，提高了跟踪的准确率和成功率。

4 基于 Attention 机制的卷积神经网络的目标跟踪算法

4.1 问题描述

在一些视觉认知的文献当中，人们已经注意到，当人类处于一个视觉场景的时候，通常不会将注意力立刻集中在整个场景当中。相反，他们会把注意力仅仅放在场景的某一个部分，在从不同部分场景提取相关的信息之后实现对整个场景的理解。随着神经网络的快速发展，以注意力机制为基础的模型结构在多种领域的任务中都取得了非常显著的成果，例如自然语言处理领域的标题生成，机器翻译，以及图像领域的图像识别等。对于深度神经网络来说，网络的内部表征往往是抽象的，不具体的，缺乏解释性的。注意力机制可以根据特定任务使模型预先捕获其应该关注的重点对象，使模型自动增加了解释维度。在文献[60]中提出了一种基于注意力机制的文本情感倾向性分析算法。其主要思想是，由于文本的上下文通常对语义的形成贡献有所不同，故为文本的上下文分配一个权重向量，该权重向量可以在模型训练过程中不断的学习，使得模型在语义分析时能够关注不同的部分，从而取得了更为精确的结果。文献[61]在视频动作识别任务中使用了基于注意力机制的神经网络模型，使得模型在处理整张图像时可以识别出关键的位置，从而增加动作识别的准确性。

在大多数传统的目标跟踪算法中通常是将整张图像内容同等看待，因此较少考虑图像内部不同部分的重要性。在跟踪过程中由于目标的外观变化或者错误跟踪，使得模型在更新过程中不可避免地引入误差，逐渐丢失真实的目标外观特征，导致外观模型逐渐失效出现漂移现象。而在视频序列中，由于被跟踪物体在不同帧之间始终存在，因此不同帧之间存在着内容相关性，该相关性信息可以作为目标跟踪的重要特征。

为缓解目标跟踪中出现的漂移现象，本章提出了一种基于 Attention（注意力）机制的卷积神经网络的目标跟踪算法。该方法将视频序列的初始帧内容作为记忆单元，使得网络学习始终保持对初始帧目标特征的记忆，并利用图像不同帧之间的前后内容关联，根据图像的第一帧的目标信息，在网络中构建 Attention 层次结构，构建基于 Attention 机制的卷积网络模型，学习特征图中每个位置的权重矩阵，使其自动关注目标中的关键位置，选取更加可靠的特征。通过对模型的训练学习，可以自动学习注意力权重参数。并从网络的不同层中提取特征进行金字塔池化处理，构建多专家分类器对目标的外观特征进行分类，实现在多种场景下进行稳定有效的跟踪。需要指出的是，可以在未来的工作中考虑使用可靠的中间帧结果作为记忆单元，进行 Attention 机制的尝试，本章只针对初始帧进行讨论。

4.2 Attention 机制思想

神经网络通过模仿人脑的结构，将大量神经元联结后组成复杂的网络，从而可以对输入信息中所隐藏的潜在复杂特征进行学习。近几年来随着深度学习的飞速发展，图像、文本等领域得到了突破性的进展，但深度学习自身的深层网络结构与大量的参数也带来了运算规模过大，资源消耗过多等问题。因此如何优化深度网络结构，使得深度学习的过程更为高效也是研究人员比较关注的问题。

研究人员在人眼视觉识别机制的研究中发现，由于人脑的资源有限，因此视觉信息进入大脑后，会对不同层次的信息进行初步筛选和精度筛选后再进行处理，而该筛选的机制才是人脑处理视觉信息的重要部分。基于该原理，Attention（注意力）机制在图像与文本处理中逐步得到广泛的应用。

在 Attention 机制提出之前，图像识别与跟踪任务都是把完整的图像作为模型的输入，进而通过参数的学习得到输出。但在目标跟踪任务中，被追踪物体在连续的图像序列中具有极强的目标性与关联性，模型不需要对整个输入图像进行参数的学习与调整。因此 Attention 机制为网络加入了关注区域的移动、缩放等，使得网络的学习更专注于被跟踪目标的关键部分。

Attention 可以分为 hard attention 和 soft attention 两种模型[61]。若将图像作为模型的输入，在进行一系列操作得到图像的特征信息后，Attention 机制可以对图像中选取更值得关注的部分，进而进行追踪物体的提取等操作。其中 hard attention 模型在进行特征权重的选择时，通过采样或阈值选取只针对最大的一个或几个权重进行特征选择计算，当特征集合中有大量相似权重的特征时，该方法会丢失大量信息。因此本章节借鉴了 soft attention 的思想，该方法是通过对特征的权重矩阵进行加权计算，并利用 Softmax 函数将它们归一化，这样所有的特征信息都会得到保留，同时对于突出的权重信息也保留的更多，该结构加入神经网络模型后，可以不断的通过训练改变参数矩阵的值，从而动态改变权值，使得预测被跟踪物体的特征描述更为贴合图像的变化与更新，其公式表示如下：

$$a_{t,i} = \frac{e^{W_i \cdot h_{t-1}}}{\sum_{j=1}^{M \times M} e^{W_j \cdot h_{t-1}}} \tag{4.1}$$

若 t 时刻模型的输入图像大小为$M \times M$，则$a_{t,i}$代表该时刻图像中位置 i 的重要性，其中W_i 代表了注意力权重矩阵，该权重矩阵的参数在模型的训练过程中不断的更新，使得模型能够将注意力保持在被跟踪物体的关键区域。

4.3 基于 Attention 机制的多专家目标跟踪算法设计

4.3.1 算法描述

该算法首先利用 Attention 机制和卷积神经网络构建目标的判别式外观模型，利用视频序列数据集对网络进行预训练；接着从不同的卷积层中提取卷积特征，利用空间金字塔池化归一化特征表达，针对每一层特征构建一个专家分类器；利用多个互补的专家分类器组成多专家模型对目标进行在线跟踪。整体的算法框架如图 4.1 所示，具体包括如下几个阶段：

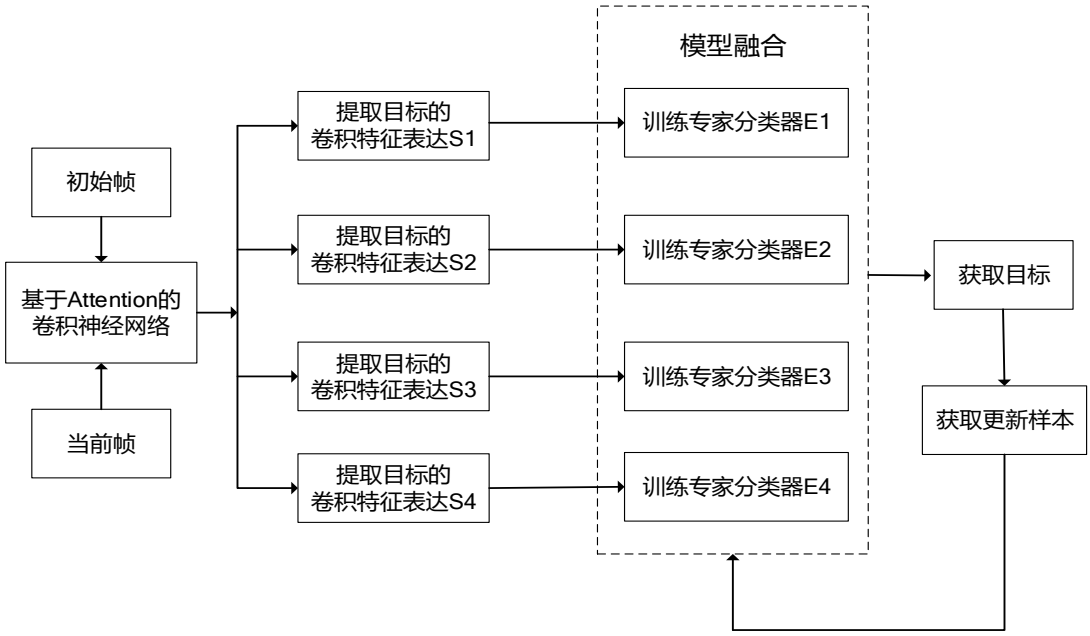


图 4.1 算法框架图

Fig. 4.1 The framework of the algorithm

（1）由于初始帧给定了真实的目标标注，因此该模型将视频序列的初始帧目标内容作为记忆单元，利用第一帧构建 Attention 层级结构，确定图像中的关注重点，自动为卷积特征图的各个位置分配权重，利用 Attention 层的输出权重重新计算卷积特征图。

（2）建立多层卷积网络结构建立目标的外观模型，利用目标跟踪标准数据集对网络进行多次迭代训练。

（3）从训练好的网络中提取不同层级的卷积特征，利用空间金字塔池化将这些来自不同层级的卷积特征进行尺度归一化处理。利用多示例分类器，为每一层特征构建相应的专家分类器。

（4）提取当前目标的关于 Attention 的外观特征表达，对建立好的多个专家分类器进行模型融合得到下一帧的目标状态。

（5）利用新的目标状态，采用多步差模型更新方式对多个专家分类器进行更新，重复步骤（4），直到视频序列结束。

4.3.2 基于初始帧内容的 Attention 模型

Attention 机制的基本思想是在计算更高层表达的时候为低层位置分配权重重要性。我们将一幅图像看作是一个全局上下文，不同位置的上下文内容对决定目标的属性的重要性有所不同。在本章中，我们以初始帧的真实标注样本内容为基准，将其看作网络的的记忆单元，建立 Attention 模型，自动为卷积特征图的各个位置分配权重，从而获取与首帧内容关联的外观特征表达。具体的算法思想如图 4.2 所示。

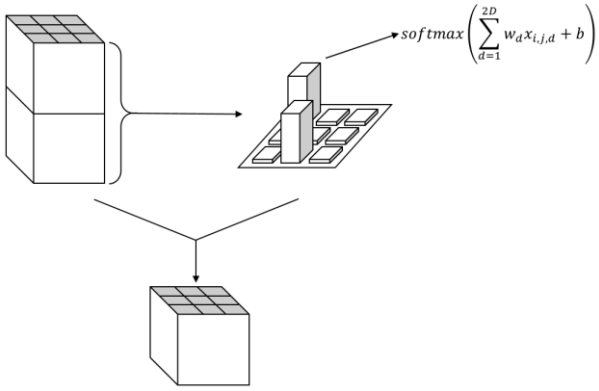


图 4.2 Attention 机制示意图

Fig. 4.2 The diagram of the proposed attention mechanism

我们将初始帧和当前帧同时输入到多层卷积网络中，并从卷积层中提取卷积特征图，其形状大小表示为 $D\times K\times K$，即存在 D 个卷积特征图，每个特征图的大小为 $K\times K$。我们将不同卷积特征图中同一位置的卷积值重新表示为一个 D 维向量$X_{i,j}$，则会得到 $K\times K$ 个这样的向量，将其表示为如下形式：

$$X_{i,j} = \left[X_{1,1}, X_{1,2}, ..., X_{K,K}\right] \quad X_{i,j} \in R^D \tag{4.2}$$

每一个位置的向量将卷积特征图分成了互不重叠的 $K\times K$ 个区域，利用 Attention 机制使得模型能够关注其中的某个区域。将初始帧与当前帧的卷积特征合并，并利用前向反馈网络来学习二者的内容关联，其计算方式如下：

$$g_{i,j} = \tanh\left(W_{attention}\left[X_{i,j}^{first}; X_{i,j}\right] + b_{attention}\right) \tag{4.3}$$

其中，$X_{i,j}^{first}$为第一帧的卷积特征，$W_{attention} \in R^{2D}$和$b_{attention} \in R^{1}$为网络需要学习的参数变量，在每个位置计算之后得到$\{g_{1,1}, g_{1,2}, \ldots, g_{K,K}\}$，接着利用 Softmax 公式来计算最后的权重矩阵$A = \{\alpha_{1,1}, \alpha_{1,2}, \ldots, \alpha_{K,K}\}$如下：

$$\alpha_{i,j} = \frac{\exp(g_{i,j})}{\sum_{m=1}^{K}\sum_{n=1}^{K}\exp(g_{m,n})} \tag{4.4}$$

最后将权重矩阵与卷积特征图进行内积运算：

$$X_{i,j}^{new} = \alpha_{i,j} * X_{i,j} \tag{4.5}$$

这样，完成了针对图像初始帧的 Attention 计算。由于模型是可微的，可以通过对视频数据进行训练来学习 Attention 模型中的参数，使其根据第一帧的图像信息，实现自动关注卷积特征图的不同位置。

4.3.3 Attention 机制卷积神经网络结构设计

在本章中，我们提出了一种基于 Attention 机制的卷积神经网络来对目标的外观进行建模。根据上一小节的原理，利用 Keras 深度学习框架，在网络中设计了 Attention 层次结构。具体的网络结构如图 4.3 所示：

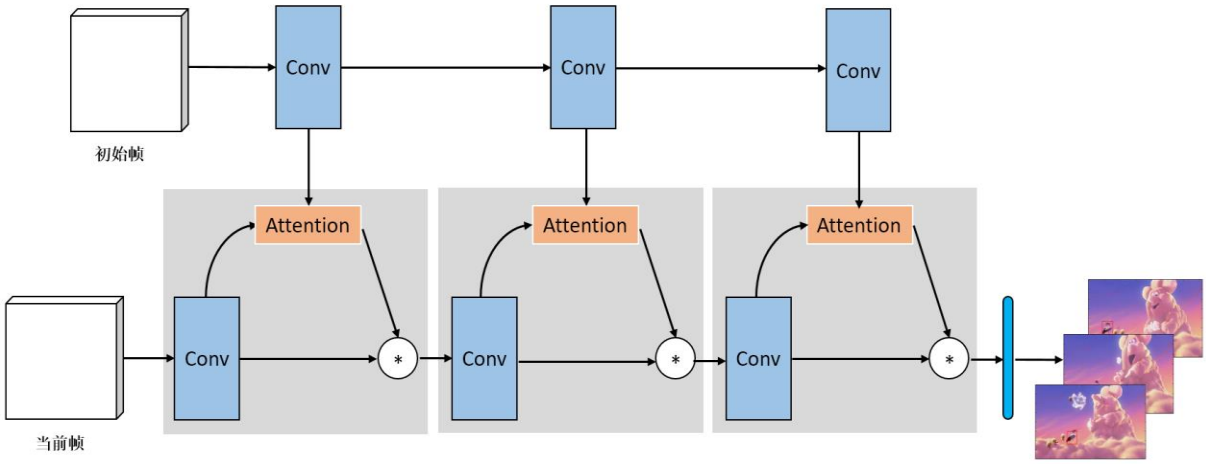


图 4.3 基于 Attention 机制的卷积神经网络结构

Fig. 4.3 The architecture of the convolutional neural network based on attention mechanism

在网络中包含两个输入，一个是初始帧，一个是当前帧。输入内容是图像中目标框的矩形区域，输入大小均重新调整为 112×112，在网络中，初始帧和当前帧会共享网络中的权重参数。我们将网络按照卷积层中输出卷积特征图的大小将网络分为多个阶段，每个阶段包括若干卷积层和 Attention 层。第一阶段 S1 的包含两个卷积层，分别包括 96 个卷积核；第二阶段 S2 有一个卷积层，包含 256 个卷积核；第三阶段 S3 有一个卷积层，包含 512 个卷积核； 第四阶段 S4 有一个卷积层，包含 512 个卷积核。网络中所有卷积层的卷积核大小均为 3×3。每一个卷积层的输出会送到 ReLU 非线性变化层中去，在每个阶段后面都会有最大池化操作和 dropout 层。当前帧在网络中前向计算的时候，在网络的最后三个卷积层会利用初始帧的对应输出来计算 Attention 层。网络的最后一层为 Softmax 层，使用二分类对输入进行目标和背景的区分。与第三章类似，我们采用有监督的训练方式对模型进行训练，利用部分视频数据集作为训练集，模型的损失函数为交叉熵损失函数，并采用梯度下降法对参数进行更新。

4.3.4 基于 Attention 目标外观表达的多专家在线跟踪

在本章中，我们利用训练好的 Attention 卷积网络模型，从网络中的不同层次中提取卷积特征并进行金字塔空间池化处理，建立相应的多示例分类器。我们将每一个分类器作为一个专家分类器，并融合多个互补的专家分类器实现对目标的跟踪。整体的算法框架如图 4.1 所示。

在开始对视频序列进行跟踪时，我们会在初始帧指定需要跟踪的目标。在跟踪过程中，由于初始帧的标注信息是最准确无误的，故我们始终保留第一帧的信息，将其看作记忆单元，并利用初始帧和 Attention 卷积网络对目标的外观进行建模，使网络具有记忆功能，能够在跟踪过程中始终保留对初始帧目标特征的记忆。通常，在网络的不同层次中所能提取的卷积特征具有不同的性质，网络的顶层特征捕捉的是更加抽象的语义特征，而在底层特征中包含了更多的类似于边缘、角点的结构特征。因此我们分别从网络的不同阶段的卷积层中提取卷积特征 S1，S2，S3，S4，构成互补的特征表达。由于不同阶段的卷积特征大小不一致，故我们采用了文献[32]介绍的空间金字塔池化对提取到的特征进行池化操作，使其具有相同的特征长度。之后，我们利用 3.4.2 小节介绍的改进的多示例学习算法分别对每一阶段的特征构建专家分类器 E1，E2，E3，E4，每个专家分类器是一个多示例分类器。 最后融合多个分类器的结果，将各个分类器的结果进行加权平均实现目标的跟踪。为了在跟踪过程中适应目标的变化，同时能够抑制目标遮挡等原因导致的外观变化的影响，本章采用不同的更新策略来对分类器进行更新。对于不同的专家分类器我们采用不同的更新频率，针对浅层特征构成的分类器，由于其对目标外观

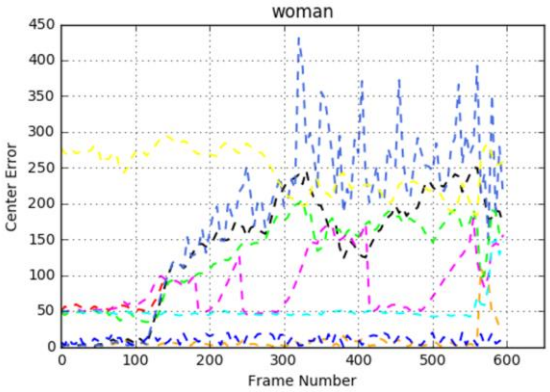
的结构特征敏感，采用快更新的方式以适应目标的变化；对于深层特征构成的分类器，由于其能抽取更多的深层次的抽象信息，采用慢更新的方式来避免周围的干扰因素，从而使模型更加鲁棒。

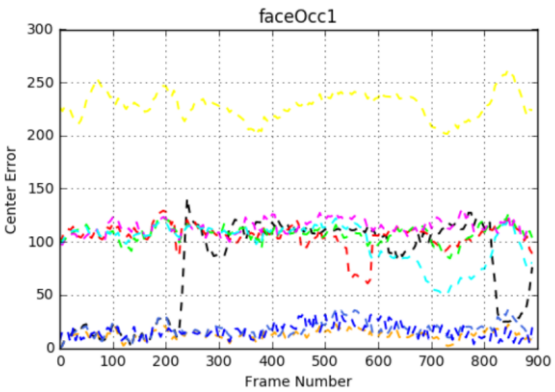
4.4 实验结果及分析

本节是对基于 Attention 机制的卷积神经网络的目标跟踪算法的实验验证与分析。本实验平台采用了 Ubuntu 操作系统，Intel Core i7-6700 3.40GHz 处理器，8GB 内存，配有英伟达 Tesla K40c GPU，使用 Python 和 Keras 深度学习框架进行仿真模拟。

为了验证本章提出的目标跟踪算法的性能，我们选取了公开的目标跟踪算标准评测[55]中使用的 8 种性能较优异的目标跟踪算法作为实验对照，并在目标跟踪标准数据集中的多个视频序列上进行实验。在这些视频序列中包含了多种不同的场景，所有的视频跟踪序列都带有标注完好的目标位置。在视频数据中存在光照变化、尺度变化、物体形变、复杂背景干扰等多种挑战。我们选取的跟踪算法分别为 SCM [58]，ASLA[62]，TLD [21]，IVT [17]，VTD [14]，CXT [63]，MTT[64]和 DFT[50]。我们利用其中 60 个视频序列作为辅助数据来对搭建好的模型进行预训练。

我们采用目标的中心位置误差来定量的对不同的目标跟踪算法进行比较。我们选取其中 8 个视频序列的对比结果进行展示，实验的结果如图 4.4 所示。在图中，显示了不同跟踪算法在视频序列的每一帧的跟踪结果的中心位置与真实中心位置的距离像素差。





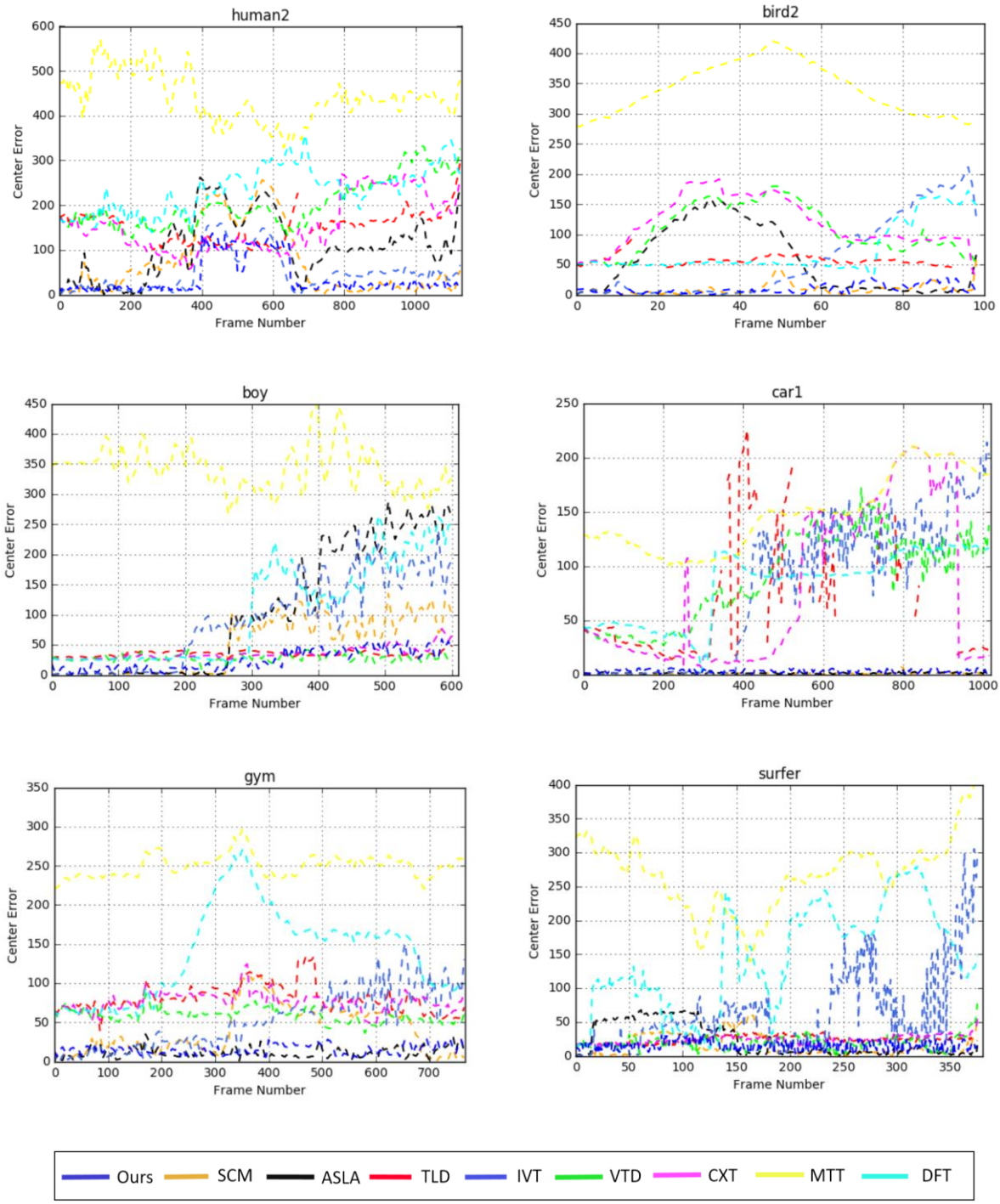


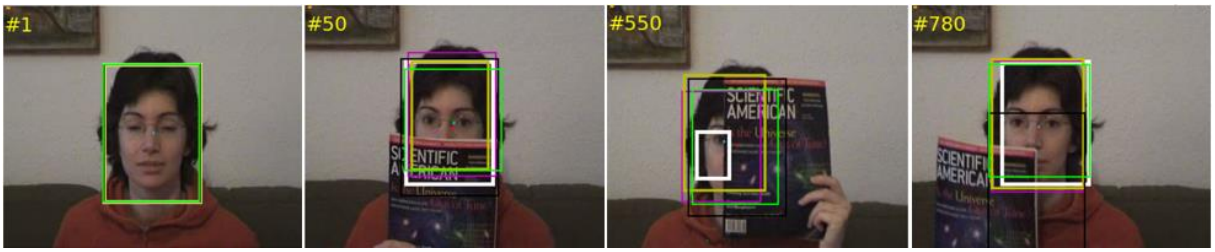
图 4.4　不同跟踪算法的中心误差对比曲线图

Fig. 4.4　Comparison of the central error of different tracking methods

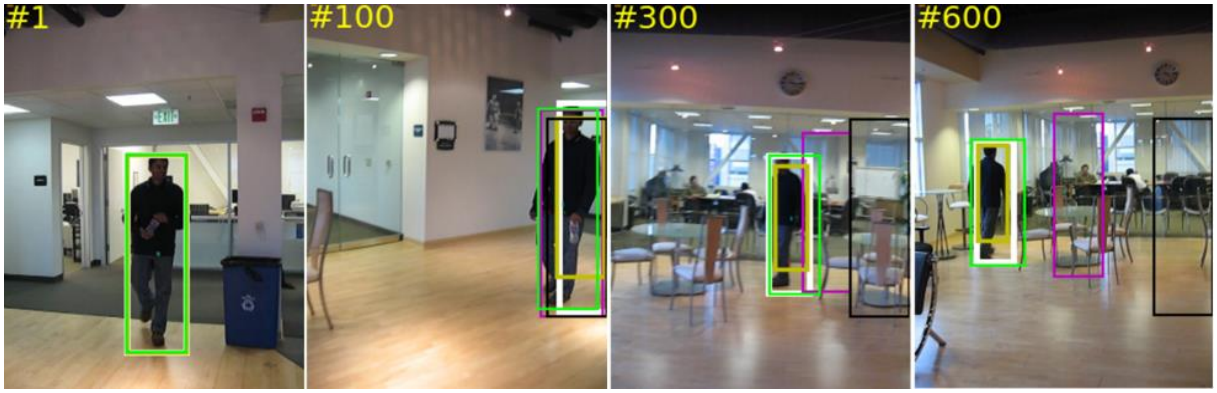
从以上实验结果可以看出，本章提出的目标跟踪算法距离目标真实位置的距离更近，具有更低的中心误差值，优于其他的跟踪算法。在所跟踪的视频序列中存在多种挑战，

例如在视频序列 *Bird2* 和 *Woman* 中，跟踪目标都存在被物体部分遮挡的情况，一些算法如 VID 算法和 DFT 算法在跟踪过程中由于错误跟踪的累积逐渐丢失目标。本章提出的算法通过结合初始帧的 Attention 计算，使其自动关注目标中的关键位置，在更新过程中能够选取更加可靠的特征，从而有效缓解出现的漂移现象，保持更加稳定的跟踪。在视频序列中 *Boy* 和 *Gym* 中存在一定程度的目标旋转和快速移动的问题，大部分的算法在视频序列的初始阶段能够较好的完成跟踪，但是当目标出现旋转或者剧烈的运动变化时就会偏离目标并且无法继续进行有效跟踪，但是本章算法在目标发生变化之后仍然能够继续准确地进行跟踪。在图 4.5 中展示了上述实验对应的视频序列的部分视频帧的跟踪结果，它们分别为 *Woman*，*FaceOcc1*，*Human2*，*Bird2*，*Boy*，*Car1*，*Gym*，*Surfer*，从图中可以更加直观地看出，本章提出的算法在多种具有挑战的场景中都表现出优于其他算法的性能。

(a) *Woman*



(b) *FaceOcc1*



(c) *Human2*

(d) *Bird2*

(e) *Boy*

(f) *Car1*

(g) *Gym*

(h) *Surfer*

图 4.5　不同算法在部分数据集的定性跟踪结果对比图

Fig. 4.5　Qualitative results of the proposed method on several test sequences

4.5 本章小结

本章提出了一种基于 Attention 机制的卷积神经网络的目标跟踪算法。该算法利用 Attention 机制能够通过网络训练自动学习关注重点的特点，提出了一种利用视频序列初始帧信息的 Attention 模型，消除复杂背景环境和目标变化对目标建模的影响，缓解目标跟踪的漂移现象。结合卷积神经网络的训练，自动学习特征图中每个位置的权重矩阵，构建基于注意力机制的判别式外观模型。利用该外观模型和改进的多示例分类算法构建多个互补的专家分类器，进行模型融合实现目标的在线跟踪。与现有的目标跟踪算法相比，该方法能够在多种场景下实现更加鲁棒的跟踪效果。

结　论

伴随着大量的目标跟踪算法的提出，目标跟踪技术已经在理论研究上得到了快速地发展，并且在多种相关领域中取得了优异的成果。然而，在实际的跟踪环境中存在诸多挑战，在不同的特定场景中会对目标跟踪算法的性能提出更高的需求。本文主要围绕利用卷积神经网络来提高目标跟踪方法性能进行研究，着重分析了基于判别式外观模型的目标建模方法，对以往的算法进行了分析总结，并针对它们的不足，提出了两种基于卷积神经网络的目标跟踪算法。本文的主要研究工作和成果如下：

（1）提出了基于卷积神经网络多尺度表达的多示例目标跟踪算法。该方法针对传统目标跟踪算法中的人工构造特征表达能力不足、难以提取涉及语义信息的问题以及目标运动过程中的尺度变化问题，利用卷积神经网络可以自动学习图像中的深层语义信息的特点，结合拉普拉斯金字塔构建多个尺度的卷积网络结构。使用视频跟踪数据集对网络模型以参数共享的方式进行由粗到细的训练，从而获取对尺度变化更具鲁棒性的目标多尺度外观表达，使其自适应目标的尺度变化。并结合多示例学习算法的优势，针对多示例算法易饱和的问题，增加了相应的惩罚项进行改进，使用多尺度外观表达代替原始的以人工构造特征为基础的外观模型，构建基于多尺度表达的多示例分类器来实现目标的在线跟踪。与现有的目标跟踪算法相比，该方法能实现更加稳定的跟踪，提高了算法的准确率和成功率。

（2）提出了基于 Attention 机制的卷积神经网络的目标跟踪算法。该方法针对目标跟踪过程中的目标变化导致的漂移现象，将视频序列的初始帧内容作为记忆单元，使得网络学习始终保持对初始帧目标特征的记忆，并根据视频帧之间的内容关联，利用 Attention 机制的理论思想，将视频序列的初始帧融入到网络模型的学习当中，构建基于 Attention 机制的卷积网络模型，通过模型训练可以自动确定卷积特征图中的权重分配，确定图像中的关注重点。利用金字塔池化处理来自不同层级的卷积特征，融合多个互补的专家分类器实现目标的在线跟踪。实验结果表明，该方法可以充分考虑目标的初始帧信息以及关键位置，有效缓解在跟踪过程中由于目标变化出现的漂移现象，在多种场景下展现了良好的跟踪效果，证实了该目标跟踪算法的有效性。

有待进一步研究：（1）本文主要关注的是对目标外观进行建模的阶段。在目标跟踪的过程中，需要通过在候选区域来选取目标的候选位置，这也是影响目标跟踪效果的关键因素，因此下一步可以尝试在算法中结合更加高效的搜索方法来进一步提高算法的性能。（2）对于记忆单元的选择，可以在未来的工作中考虑使用可靠的中间帧结果，进行 Attention 机制的尝试。

参 考 文 献

[1] Xu M, Orwell J, Lowey L, et al. Architecture and algorithms for tracking football players with multiple cameras[J]. IEE Proceedings-Vision, Image and Signal Processing, 2005, 152(2): 232-241.

[2] Kushwaha A K S, Srivastava R. Performance evaluation of various moving object segmentation techniques for intelligent video surveillance system[C]//Signal Processing and Integrated Networks (SPIN), 2014 International Conference on. IEEE, 2014: 196-201.

[3] Jones W D. Keeping cars from crashing[J]. IEEE spectrum, 2001, 38(9): 40-45.

[4] Tai J C, Tseng S T, Lin C P, et al. Real-time image tracking for automatic traffic monitoring and enforcement applications[J]. Image and Vision Computing, 2004, 22(6): 485-501.

[5] Bonin-Font F, Ortiz A, Oliver G. Visual navigation for mobile robots: A survey[J]. Journal of intelligent and robotic systems, 2008, 53(3): 263.

[6] van der Zwaan S, Santos-Victor J. An insect inspired visual sensor for the autonomous navigation of a mobile robot[J]. Proc. of the Seventh International Sysposium on Intelligent Robotic Systems (SIRS), 1999.

[7] Antich J, Ortiz A. Development of the control architecture of an underwater cable tracker[J]. International journal of intelligent systems, 2005, 20(5): 477-498.

[8] 张勇, 欧宗瑛, 侯建华. 基于主动轮廓模型的医学图像边界跟踪[J]. 仪器仪表学报, 2002(z1): 173-174.

[9] Yilmaz A, Javed O, Shah M. Object tracking: A survey[J]. Acm computing surveys (CSUR), 2006, 38(4): 13.

[10]侯志强, 韩崇昭. 视觉跟踪技术综述[J]. 自动化学报, 2006, 32(4): 603-617.

[11]黄凯奇, 陈晓棠, 康运锋, 等. 智能视频监控技术[J]. 计算机学报, 2014, 37(49): 1-10.

[12]Mei X, Ling H. Robust visual tracking using ℓ 1 minimization[C]//Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009: 1436-1443.

[13]Liu B, Yang L, Huang J, et al. Robust and fast collaborative tracking with two stage sparse optimization[J]. Computer vision–ECCV 2010, 2010: 624-637.

[14]Kwon J, Lee K M. Visual tracking decomposition[C]//Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010: 1269-1276.

[15]Xiu C, Wei S, Wan R, et al. CamShift Tracking Method Based on Target Decomposition[J]. Mathematical Problems in Engineering, 2015, 2015(1):1-20.

[16]Matthews L, Ishikawa T, Baker S. The template update problem[J]. IEEE transactions

on pattern analysis and machine intelligence, 2004, 26(6): 810-815.

[17]Ross D A, Lim J, Lin R S, et al. Incremental learning for robust visual tracking[J]. International journal of computer vision, 2008, 77(1): 125-141.

[18]Han B, Comaniciu D, Zhu Y, et al. Sequential kernel density approximation and its application to real-time visual tracking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(7): 1186-1197.

[19]Jepson A D, Fleet D J, El-Maraghi T F. Robust online appearance models for visual tracking[J]. IEEE transactions on pattern analysis and machine intelligence, 2003, 25(10): 1296-1311.

[20]Avidan S. Support vector tracking[J]. IEEE transactions on pattern analysis and machine intelligence, 2004, 26(8): 1064-1072.

[21]Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection[J]. IEEE transactions on pattern analysis and machine intelligence, 2012, 34(7): 1409-1422.

[22]Avidan S. Ensemble Tracking[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2007, 29(2):261-271.

[23]Grabner H, Leistner C, Bischof H. Semi-supervised on-line boosting for robust tracking[J]. Computer Vision–ECCV 2008, 2008: 234-247.

[24]Grabner H, Bischof H. On-line boosting and vision[C]//Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. IEEE, 2006, 1: 260-267.

[25]Babenko B, Yang M H, Belongie S. Robust object tracking with online multiple instance learning[J]. IEEE transactions on pattern analysis and machine intelligence, 2011, 33(8): 1619-1632.

[26]Zhang K, Zhang L, Yang M H. Real-time compressive tracking[C]//European Conference on Computer Vision. Springer Berlin Heidelberg, 2012: 864-877.

[27]Hare S, Golodetz S, Saffari A, et al. Struck: Structured output tracking with kernels[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 38(10): 2096-2109.

[28]Son J, Jung I, Park K, et al. Tracking-by-segmentation with online gradient boosting decision tree[C]//Proceedings of the IEEE International Conference on Computer Vision. 2015: 3056-3064.

[29]Saffari A, Leistner C, Santner J, et al. On-line random forests[C]//Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on. IEEE, 2009: 1393-1400.

[30]Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems. 2012: 1097-1105.

[31]Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate

object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[32]He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[C]//European Conference on Computer Vision. Springer International Publishing, 2014: 346-361.

[33]Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[34]Li H, Li Y, Porikli F. Robust online visual tracking with a single convolutional neural network[C]//Asian Conference on Computer Vision. Springer International Publishing, 2014: 194-209.

[35]Hong S, You T, Kwak S, et al. Online Tracking by Learning Discriminative Saliency Map with Convolutional Neural Network[C]//ICML. 2015: 597-606.

[36]Li H, Li Y, Porikli F. Deeptrack: Learning discriminative feature representations online for robust visual tracking[J]. IEEE Transactions on Image Processing, 2016, 25(4): 1834-1848.

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

[38]Fan J, Xu W, Wu Y, et al. Human tracking using convolutional neural networks[J]. IEEE Transactions on Neural Networks, 2010, 21(10): 1610-1623.

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

[40]Elgammal A, Duraiswami R, Harwood D, et al. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance[J]. Proceedings of the IEEE, 2002, 90(7): 1151-1163.

[41]Comaniciu D, Ramesh V, Meer P. Kernel-based object tracking[J]. IEEE Transactions on pattern analysis and machine intelligence, 2003, 25(5): 564-577.

[42]Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors[J]. 1986, 323(6088):533-536.

[43]Fukushima K. Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position- Neocognitron[J]. ELECTRON. & COMMUN. JAPAN, 1979, 62(10): 11-18.

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

[45]Zhou B, Lapedriza A, Xiao J, et al. Learning deep features for scene recognition using places database[C]//Advances in neural information processing systems. 2014: 487-495.

[46]Zeiler M D, Fergus R. Visualizing and understanding convolutional networks[C]//European conference on computer vision. Springer International Publishing, 2014: 818-833.

[47]Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1-9.

[48]Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[49]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.

[50]Sevilla-Lara L, Learned-Miller E. Distribution fields for tracking[C]//Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012: 1910-1917.

[51]Glorot X, Bordes A, Bengio Y. Deep Sparse Rectifier Neural Networks[C]//Aistats. 2011, 15(106): 275.

[52]Maas A L, Hannun A Y, Ng A Y. Rectifier nonlinearities improve neural network acoustic models[C]//Proc. ICML. 2013, 30(1).

[53]He K, Zhang X, Ren S, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1026-1034.

[54]Lindeberg T. Scale-space theory: A basic tool for analyzing structures at different scales[J]. Journal of applied statistics, 1994, 21(1-2): 225-270.

[55]Wu Y, Lim J, Yang M H. Online object tracking: A benchmark[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2013: 2411-2418.

[56]Henriques J F, Caseiro R, Martins P, et al. High-speed tracking with kernelized correlation filters[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(3): 583-596.

[57]Choi J, Jin Chang H, Jeong J, et al. Visual tracking using attention-modulated disintegration and integration[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 4321-4330.

[58]Zhong W, Lu H, Yang M H. Robust object tracking via sparse collaborative appearance model[J]. IEEE Transactions on Image Processing, 2014, 23(5): 2356-2368.

[59]Gao J, Ling H, Hu W, et al. Transfer learning based visual tracking with gaussian processes regression[C]//European Conference on Computer Vision. Springer

International Publishing, 2014: 188-203.
[60]Tang D, Qin B, Liu T. Aspect level sentiment classification with deep memory network[J]. arXiv preprint arXiv:1605.08900, 2016.
[61]Sharma S, Kiros R, Salakhutdinov R. Action recognition using visual attention[J]. arXiv preprint arXiv:1511.04119, 2015.
[62]Jia X, Lu H, Yang M H. Visual tracking via adaptive structural local sparse appearance model[C]//Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012: 1822-1829.
[63]Dinh T B, Vo N, Medioni G. Context tracker: Exploring supporters and distracters in unconstrained environments[C]//Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE, 2011: 1177-1184.
[64]Zhang T, Ghanem B, Liu S, et al. Robust visual tracking via multi-task sparse learning[C]//Computer vision and pattern recognition (CVPR), 2012 IEEE conference on. IEEE, 2012: 2042-2049.

攻读硕士学位期间发表学术论文情况

1　基于卷积神经网络的多尺度表达的目标跟踪方法. 第一作者（导师第二作者）. 发明专利（已受理），专利申请号：201611201895.0。（本硕士学位论文第三章）

2　Salience based object tracking in unconstrained environments. 第二作者（导师第四作者）. Neurocomputing. 主办单位:Elsevier. SCI 期刊检索.（Under Review, 本硕士学位论文第四章）

致　　谢

行文至此，三年硕士生涯已经走到那个圆满的终点。站在这里回首，我仿佛又能看到三年前初入大连理工大学汇聚之光实验室时的激动与期待，激动的是能够重新获取三年宝贵的提升自我的机会，期待的更是这三年硕士生涯能够带给我的改变与升华，但现在的我心中充盈最多的仍是感激。

首先感谢我的导师××教授在学术科研和生活上对我的细致认真的教导，×老师在图像领域耕耘多年，有着极高的学术造诣和科研成果，正是×老师的悉心指点，让我从研究方向的确立、论文的书写、实验的设计以及最终论文的完成都可以有条不紊的进行。×老师不仅仅在学术科研上对我们有着高要求，更是以身作则，以一个刻苦的研究者的身份影响着我们，无论早晚，我们都能看到×老师在办公室辛勤科研的身影，让我们可以随时向他咨询学术上的难题。×老师的这种对待科研和工作几十年如一日的精神，也是我硕士期间学到的最宝贵的东西之一。

其次，我还要感谢××老师的无微不至的指点。×老师和蔼亲切的教导让我很快的度过了初入实验室的迷茫与彷徨时期，对我学术研究的各个阶段都耐心的进行督促与指导，并以自己开阔的学术视野与经验给予我在论文的书写方面的宝贵意见，此外，×老师还在实验室的建设方面付出了极大的努力，为我们组织了各种温馨愉快的集体活动，关心我们的日常心理波动与生活上的困难，拉近与我们之间的距离，为我们提供了优异的科研与学习环境。

再次，我要感谢在整个硕士生涯中，在科研和生活上对我关照的师兄和师姐们，正是你们的存在，汇聚之光实验室才像一个大家庭一样，让人不舍离开。其中××、××师兄在学术方面对我不厌其烦的教导，并在实验环境的搭建方面提供了宝贵的帮助。××老师和××师姐也在平时的科研与生活上，给予我很大的关照。同时，也感谢××和××在科研上和我的热烈讨论，开阔了我的思路与视野。

我还要感谢我的家人在我求学道路上给我经济上的支持与源源不断的关爱。感谢我的室友××和××在我生活上的照顾，让我在硕士期间有着温馨和快乐的寝室环境。感谢我的男神在学习和生活上对我一如既往的支撑与陪伴。

最后在即将毕业的时刻，祝汇聚之光实验室能够发展壮大，祝所有帮助过我、关心过我的老师和同学们一切顺利，谢谢你们！

大连理工大学学位论文版权使用授权书

本人完全了解学校有关学位论文知识产权的规定，在校攻读学位期间论文工作的知识产权属于大连理工大学，允许论文被查阅和借阅。学校有权保留论文并向国家有关部门或机构送交论文的复印件和电子版，可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印、或扫描等复制手段保存和汇编本学位论文。

学位论文题目：_____

作 者 签 名 ：_____日期：_____年____月____日

导 师 签 名 ：_____日期：_____年____月____日