

# 基于深度学习的目标跟踪算法研究与实现

## 摘 要

目标跟踪是一种使用视频或图像序列的上下文信息对目标的外形和运动的信息进行建模,从而预测目标运动状态并标定目标位置的技术。它在智能交通、视频监控系统等领域有着广泛的应用。传统的目标跟踪算法使用相关滤波框架,以手工设计的底层的视觉特征对目标外形进行建模,尽管相关滤波使用了频域计算来提高跟踪速度,然而仅仅是手工特征或由浅层分类器提取出来的特征对于预测目标表观模型的语义信息能力是不够的,会降低跟踪精度的提高速度。然而,深度学习的出现使得目标跟踪又得到的进一步发展。

本文首先阐述了目标跟踪的基本框架;介绍了目前主要的目标跟踪的发展历程;分析了基于相关滤波的目标跟踪算法 KCF 和基于深度学习的目标跟踪算法 ATOM 在相同数据集上得到的实验结果;最后对深度学习在目标跟踪中存在的各种未解决的问题进行分析讨论,并对单目标跟踪领域未来的发展进行了延伸展望。

**关键字:** 单目标跟踪      相关滤波      深度学习

## Research and Implementation of Target Tracking Algorithm Based on Deep Learning

### ABSTRACT

Target tracking is a technology that uses the context information of a video or image sequence to model the shape and movement of the target, thereby predicting the target's motion state and calibrating the target position. It has a wide range of applications in intelligent transportation, video surveillance systems and other fields. Traditional target tracking algorithms use a correlation filtering framework to model the shape of the target with hand-designed underlying visual features. Although correlation filtering uses frequency domain calculations to

improve tracking speed, it is only manual features or extracted by shallow classifiers. The features that come out are not enough for predicting the semantic information ability of the target apparent model, and will reduce the speed of improving the tracking accuracy. However, with the advent of the era of big data and the emergence of deep learning, there are new opportunities for improvement in target tracking research.

This article first describes the basic research framework of target tracking; introduces the current development of main target tracking; analyzes the experimental results of the target tracking algorithm KCF based on correlation filtering and the target tracking algorithm ATOM based on deep learning on the same data set ; Finally, further analysis of various problems existing in the deep learning method in target tracking is carried out, and the future development direction of the single target tracking field is prospected.

**Key Words:** Visual tracking   Correlation filter   Kernel method   Deep learning

# 目 录

第一章 绪论	1
1.1 目标跟踪的研究目的和意义	1
1.2 国内外研究背景及现状	1
1.3 本文研究内容和结构安排	2
第二章 目标跟踪相关知识	3
2.1 相关滤波理论基础	3
2.2 相关滤波跟踪框架	4
2.3 目标跟踪相关深度学习简介	5
2.3.1 人工神经网络	5
2.3.2 多层感知机	5
2.3.3 激活函数	6
2.3.4 反向传播	7
2.3.5 几种典型的神经网络模型	8
第三章 核相关滤波跟踪算法的实现与分析	11
3.1 核相关滤波跟踪算法的基本框架	11
3.2 视频信息处理的特征描述	11
3.2.1 HOG 特征	11
3.2.2 LBP 纹理特征	12
3.2.3 深度特征	13
3.3 核相关滤波算法详述	13
3.3.1 运动模型	13
3.3.2 特征提取	14
3.3.3 检测模板	14
3.3.4 模型更新	16
3.4 核相关滤波器跟踪算法跟踪效果展示与分析	16
第四章 基于深度学习的目标跟踪算法的实现与分析	18
4.1 ATOM 目标跟踪算法	18
4.1.1 目标分类网络	18
4.1.2 目标评估网络	19
4.1.3 网络在线更新	20
4.2 ATOM 算法跟踪效果展示与分析	20
第五章 实验分析	22
5.1 实验设置以及数据集评价指标	22
5.2 对比实验与分析	22
5.2.1 主观结果与分析	22

5.2.2 客观结果与分析.....	23
<b>第六章 总结与展望.....</b>	<b>25</b>
6.1 本文工作总结.....	25
6.2 未来工作展望.....	25
<b>参考文献.....</b>	<b>26</b>
<b>致 谢.....</b>	<b>28</b>

## 第一章 绪论

### 1.1 目标跟踪的研究目的和意义

人类通过视觉来获取外界的信息。通过多年来电子信息、计算机和多媒体技术的发展，获取并且分析了海量的与视觉相关的信息，如何让计算机处理这些海量视频数据已然成为计算机视觉的研究的重点内容，目标跟踪技术是处理这些视频数据的重要手段。目标跟踪在计算机视觉领域中的位置举足轻重，它是行为理解等高级语义分析的根基。跟踪要解决的主要问题可以表示为：在视频序列中，通过给定第一帧中目标的位置和大小，在后续帧中能够估计给定目标的后续状态<sup>[1]</sup>。

目标跟踪在实际生活中也被广泛应用，例如智能监控（Intelligent Monitoring）、人机交互（Human-Computer Interaction）、无人驾驶（Autonomous Driving）等。目标跟踪在某些新兴领域具有重要价值，例如智能交通系统，智能医疗诊断和治疗，精确制导系统等，尤其是在军事和视频监控领域，在这些场景下，都需要对特定的目标对象进行建模和跟踪。

### 1.2 国内外研究背景及现状

随着目标跟踪技术在全球范围内的关注度提升，越来越多的高校和研究机构正在对目标跟踪相关的工作进行资金投入。较著名的是爱丁堡大学，加利福尼亚大学梅里德分校，牛津大学，萨里大学，麻省理工学院，大学雷丁，卡内基梅隆大学，韩国浦项工业大学，瑞典林雪平大学和其他大学，以及公司或研究机构，例如 IBM, Google, Facebook 和 Microsoft。大连理工大学智能图像分析与理解实验室，上海交通大学控制与信息处理实验室是中国比较在目标领域完成了大量杰出工作的实验室。

此外，视觉目标跟踪也被许多国际权威期刊和国际会议用作关注的焦点。其中 IVC（Image and Vision Computing）、IJCV（International Journal of Computer Vision）等是比较有影响力的期刊，比较权威的国际会议包括 ICCV（IEEE International Conference on Computer Vision）、CVPR（Computer Vision and Pattern Recognition）等。

生成式的跟踪和判别式的跟踪是经典的目标跟踪方法<sup>[2]</sup>。

对于生成式的目标跟踪算法来说，它的重点是从目标的外观模型下手。具体是给出一个代表目标的模型，然后不断从目标周围产生多个候选框，选取误差最小的作为目标新的位置，从而完成追踪。生成式的目标跟踪没有考虑背景因素。它对后续跟踪影响很大，难以进行精准的跟踪，甚至跟踪失败。

判别式的目标跟踪算法，其思想是将对目标的跟踪转化为训练分类器的问题。训练

样本中，把目标作为正样本，背景标作负样本，在跟踪结束后输出分类器中得出的分数最高的正样本。基于判别式的目标跟踪算法的创新点在于开始关注背景和目标的差异，使得跟踪得到了更好的发展。

有许多表现良好并且应用广泛的传统目标跟踪算法，比如 Meanshift 算法、卡尔曼滤波<sup>[4]</sup>算法和光流法<sup>[5]</sup>。Meanshift 算法的基本思想是基于概率密度分布的，在将目标定位到概率密度增加的方向的过程中，以便算法将继续迭代到最佳位置。Camshift 算法是基于 Meanshift 算法改进的。

基于传统算法改进一些算法被广泛使用。OAB<sup>[3]</sup>算法是基于 AdaBoost 改进的，训练一些弱分类器集成以获得强分类器。样本数量太少的时候，对正负样本的分类不准确，即不利于分类器的训练。MIL<sup>[6]</sup> (Multiple Instance Learning) 是被 B.Babenko 等人提出，原理是将多个样本作为一个样本集来看待，然后对样本集进行正负样本的标定。2010 年，Z.Kala 提出了 TLD<sup>[7]</sup> (Tracking Learning Detection) 跟踪领域，她结合了检测、跟踪和学习于一体，这个算法十分有效，即使目标被遮挡或者消失，该算法都能再次跟踪到。S.Hare 等人在 2011 年提出了具有比较杰出的跟踪效果的 Struck<sup>[8]</sup>跟踪算法。

MOSSE<sup>[9]</sup>在 2010 年将相关过滤器引入到了跟踪领域。原理是训练一个滤波模板，将它和目标特征作用后获得最大响应，其求解的过程在频域进行。2012 年，J.Henriques 等人提出了 CSK<sup>[10]</sup>方法。它用循环移位的方式采样，并且用了核函数训练分类器。2014 年，作者改进了 CSK，提出了 KCF<sup>[11]</sup>跟踪算法。它将算法拓展到多个通道，并且用 HOG 特征提取特征。

自 2015 年，深度学习被广泛用在计算机视觉领域。相对于手工特征，深度学习可以对图像进行更好的描述。基于深度学习的方法会采用深度特征，但这方法很耗时而且无法满足实时性的需求，因为此类方法通常需要大量的训练。

总体而言，目标跟踪技术在学术界和企业界受到了广泛的关注和快速发展。更多的人意识到目标跟踪技术的实用场景和价值，也充分地展示了该领域的广泛应用与前景。

### 1.3 本文研究内容和结构安排

本文将通过以下章节介绍课题的研究方向及成功：

第一章：首先阐述了目标跟踪的意义；然后对单目标跟踪算法的发展历程及现状进行介绍；最后介绍本文的结构安排以及主要工作。

第二章：详细介绍目标跟踪的基础理论知识。

第三章：介绍基于核相关滤波的目标跟踪框架；再详述算法；最后展示实验效果。

第四章：将介绍基于深度学习的目标跟踪算法；然后详述算法；最后分析实验结果。

第五章：分析以上两种算法的对比实验结果。

第六章：全文的总结与未来展望。

## 第二章 目标跟踪相关知识

### 2.1 相关滤波理论基础

相关滤波这个概念产生于信号处理。假设有两个输入信号 $S_1$ 和 $S_2$ ，并且对信号的相关操作可以解释为这两个信号在某一时刻的相似性。相关性和卷积具有很强的相似性。

将相关滤波的思想引入视觉目标的跟踪领域的原理是：相关这一操作可以评估信号之间的相似性，相关值越高就表示信号之间的相似性越高。在目标跟踪的问题的处理中，训练滤波器使其与感兴趣的目标区域进行相关操作以获取最大响应输出。

信号 $S_1$ 与信号 $S_2$ 的相关操作如下所示：

$$(s_1 \otimes s_2)(\tau) = \int_{-\infty}^{+\infty} s_1^*(t) s_2(t + \tau) dt \quad (2.1)$$

$$(s_1 \otimes s_2)(n) = \int_{-\infty}^{+\infty} s_1^*(m) s_2(m + n) \quad (2.2)$$

式(2.1)表示连续信号，式(2.2)表示离散信号，其中 $*$ 表示的是信号的共轭，相关描述信号的相似性。相关性在目标跟踪领域就是为了找寻与目标有最大相似度的信号。

在目标跟踪中，要先提取特征，提取出来的特征用于筛选和比较来找到目标状态，然后再进行对目标物体的建模。在相关滤波的目标跟踪中，将训练出的滤波器与包含跟踪对象的候选区域作用后得到响应图，将响应图中的最大值作为目标的当前的位置。

若 $c$ 表示滤波模板， $i$ 表示输入图像，则响应输出 $r$ 表示如式(2.3)。

$$r = i \otimes c \quad (2.3)$$

上式表明，滤波器 $c$ 的质量决定响应输出。 $\otimes$ 表示输入图像与滤波器进行卷积，而空间卷积计算量很大，基于相关滤波的目标跟踪把卷积操作转换进入频域中进行计算，这个过程减少计算量。用 $F$ 表示快速傅里叶变换，则式(2.3)可重新表示为下式(2.4)：

$$F(r) = F(i \otimes c) = F(i) \cdot F(C)^* \quad (2.4)$$

$$R = I \cdot C^* \quad (2.5)$$

其中，式(2.5)是(2.4)简化后的样子，由此式可知，相关滤波的主要问题是得到滤波器 $C^*$ ，如下式(2.6)：

$$C^* = \frac{R}{I} \quad (2.6)$$

由于运动的物体会碰到很多不同的情形让跟踪器难以跟踪到，所以要不断的更新滤波器用来适应新的场景以及目标的外观变化。所以需要在这过程中不断求解 $C^*$ ，如式(2.7)：

$$\min_{C^*} \sum_{x=1}^m |C^* I_x - R_x|^2 \quad (2.7)$$

相关滤波通过提取的特征和滤波器的相关操作获得响应图，并根据预测的目标状态在线更新滤波器，以使算法更加鲁棒。并且基于相关滤波的跟踪在计算速度上得到提高，保证了实时性。滤波器作用的是特征向量或者特征矩阵，所以特征提取的好坏是非常关键的。

## 2.2 相关滤波跟踪框架

相关滤波跟踪的流程图如图 2.1 所示。

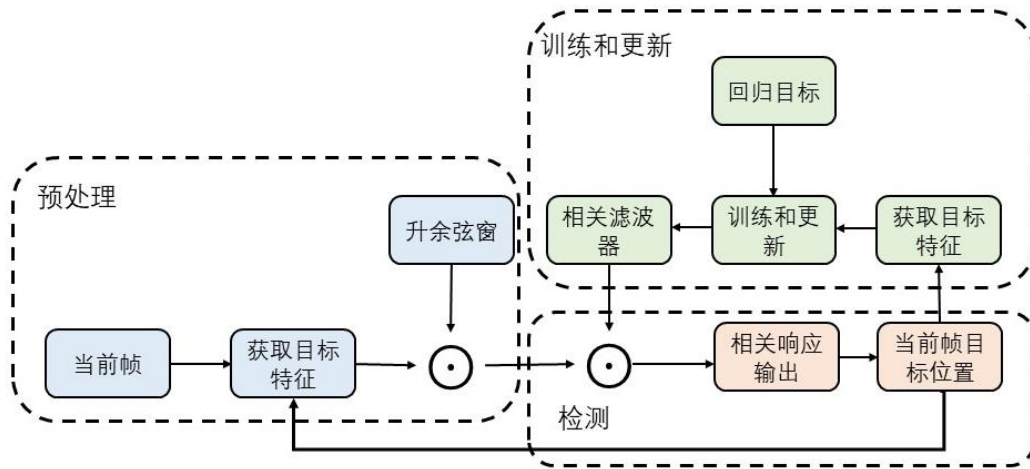


图 2.1 基于相关滤波的目标跟踪整体框架

相关滤波具体跟踪过程如下：（1）输入第一帧的位置作为目标的位置的真实值，在第一帧目标图像的周围进行采样，获得多个候选框用于训练和更新滤波器 $f$ ；（2）在后续帧中，根据前一帧的目标位置生成多个候选框，并通过余弦窗口对得到的特征进行平滑处理；（3）通过 DCF 变换将特征和滤波器 $f$ 转移到频域， $f$ 与每个候选框在频域进行卷积操作获得响应图，最大的响应值就作为跟踪对象在当前帧中的位置；（4）重复步骤（2）（3），持续更新训练滤波器，并且跟踪目标直至视频结束。

MOSSE，也就是输出滤波和最小均方误差，使得真实值与期望值的均方误差和最小化，得到相关滤波器 $f^*$ ，如式(2.8)：

$$\min_{f^*} \sum_i \|x_i \odot f^* - y_i\|^2 \quad (2.8)$$

$x_i$ 表示样本真实值， $y_i$ 表示样本期望值，通过下式(2.9)可求解 $f^*$ 。

$$f^* = \frac{\sum_i y_i \odot x_i^*}{\sum_i x_i \odot x_i^*} \quad (2.9)$$



对于 MOSSE 分类器来说，在非线性的特征空间进行跟踪效果不理想。所以，在改进中引入了核函数可以提高在高维特征空间的判别能力。

## 2.3 目标跟踪相关深度学习简介

具有多个隐藏层的人工神经网络是一种深度网络结构。深度网络结构分为人工神经网络和卷积神经网络这两种类型。

### 2.3.1 人工神经网络

神经网络<sup>[12]</sup>起源于对生物神经学的研究。人工神经网络通常是通过将多个神经节点相互连接而构建的。节点分为输入节点，隐藏节点和输出节点。

人工神经网络经历了漫长的发展阶段。首先是 1960 年代提出的“人工神经元”模型，称为“感知器”。其基本结构如图 2.2 所示：

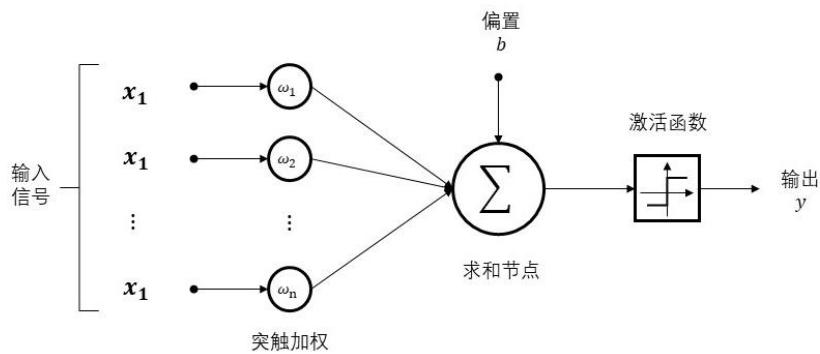


图 2.2 感知机

### 2.3.2 多层感知机

人工神经网络也被称为多层感知机（Multi Layer Perceptron, MLP）<sup>[13]</sup>。

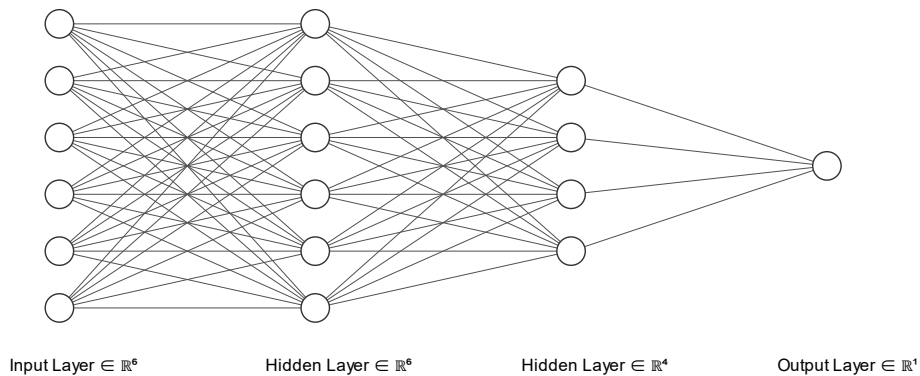


图 2.3 三层人工神经网络结构

输入层（Input Layer）接收输入的数据，隐藏层（Hidden Layer）处理输入数据，输出层（Output Layer）输出处理后的数据。深度学习模型主要包括自编码网络（Auto-encoder Networker）<sup>[14]</sup>、循环神经网络（Recurrent Neural Network, RNN）、生成式对抗网络（Generative Adversarial Network, GAN）和卷积神经网络（Convolutional Neural Network, CNN）<sup>[15]</sup>。

### 2.3.3 激活函数

一般情况下，我们常使用线性模型来解决问题。但是碰到较为复杂的情况，线性模型的效果不够理想，而非线性模型就可以了解目标的更深层信息来解决复杂的问题。模型的非线性映射通过添加激活函数<sup>[16]</sup>来实现。一般的激活函数包括 Sigmoid 函数，Tanh 函数和 Relu 函数。

Sigmoid 函数是 S 型函数，Sigmoid 曲线的斜率在接近 0 处相对较大，并且离 0 越远，该曲线就趋于平坦，这通常称为抑制区。它的缺点是:sigmoid 函数不是以零为中心的函数，这会导致网络训练速度一开始异常慢；sigmoid 函数包含指数计算，计算速度比较慢。

Sigmoid 函数的公式是：

$$\sigma(\theta) = \frac{1}{1 + e^{-\theta}} \quad (2.10)$$

Sigmoid 函数的示意图如图 2.4 所示：

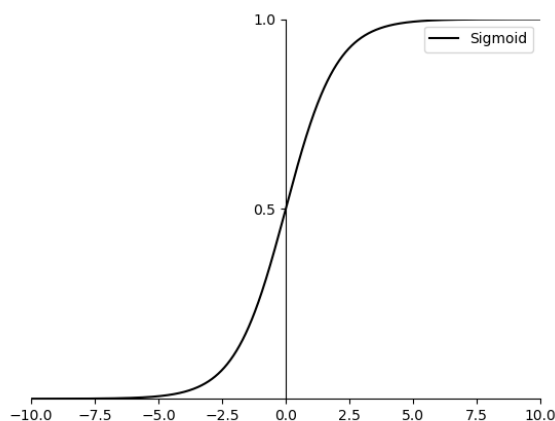


图 2.4 Sigmoid 函数图像

Tanh 函数是双曲正切函数。Tanh 的输出介于 (-1,1)，并且整个函数都以 0 为中心。一般隐藏层会使用 Tanh 函数。

Tanh 函数的公式为：

$$\text{Tanh}(\theta) = \frac{\sinh(\theta)}{\cosh(\theta)} = \frac{e^{\theta} - e^{-\theta}}{e^{\theta} + e^{-\theta}} \quad (2.11)$$

Tanh 函数的示意图如图 2.5 所示：

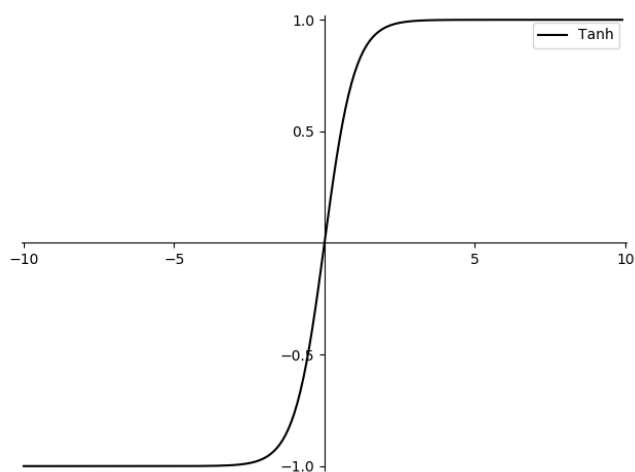


图 2.5 Tanh 函数图像

Relu 函数是分段线性函数，它的函数表达式为：

$$f(\theta) = \max(0, \theta) \quad (2.12)$$

Relu 函数的示意图如图 2.6 所示：

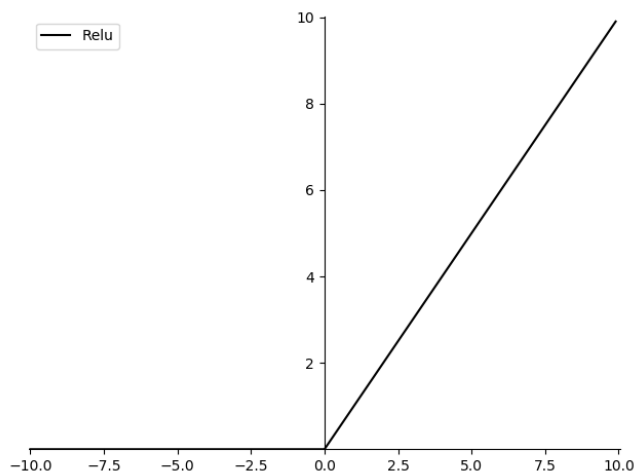


图 2.6 Relu 函数图像

当  $\theta$  为正时，Relu 函数的导数始终等于 1，而当  $\theta$  为负时，导数恒为 0。这就避免了出现梯度饱和。它也有一些缺点：当输入值为负数时，Relu 没有任何效果。

### 2.3.4 反向传播

在 1989 年，Yann 等人<sup>[17]</sup>在卷积神经网络中引入了反向传播算法。反向传播的功能

是优化神经网络参数。

前向传播的过程是输入的样本进入输入层进入网络,通过隐藏层后,从输出层输出。通过反向传播误差并使损失函数最小化来实现最佳算法,以减少真实的输出值和预测值的差异。

反向传播的目的是通过梯度下降的方法来更新每层的参数值。通过训练参数,通过迭代来调整每层的参数的权重,来使得每个参数都达到最小的误差。通过前向传播和反向传播不断的调整,最后达到算法给出的范围内。

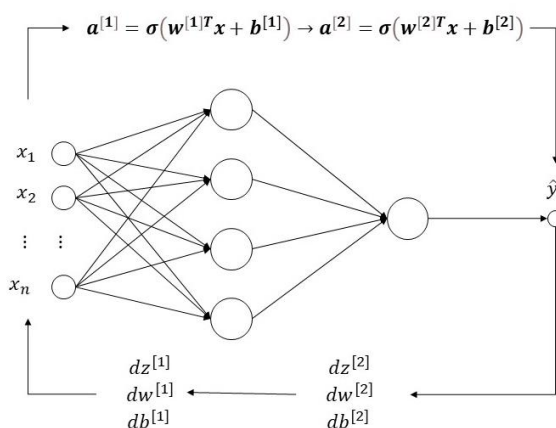


图 2.7 反向传播示意图

## 2.3.5 几种典型的神经网络模型

### 1. 卷积神经网络结构

卷积神经网络可以追溯到很久以前, Hubel 等人<sup>[18]</sup>提出的。深度学习中卷积用于处理类似于图像这样的数据。卷积网络使用与相对应的区域的局部连接,而不是像素单元的一对一连接。卷积网络的所有像素点对同一个卷积核的单通道共享权重。

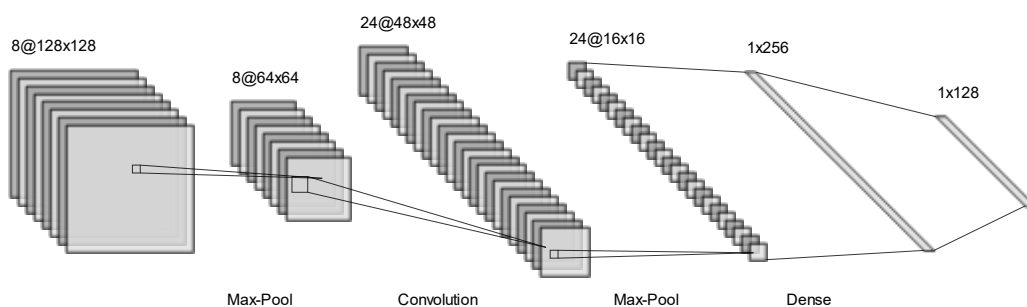


图 2.8 卷积网络结构图

卷积神经网络包含四层,分别是卷积层、激活函数、池化层和全连接层。其中,卷积层的功能是用来提取数据特征的。激活函数是为了可以使数据拥有非线性特点,这

可以提取更多的数据信息。池化层可以防止过拟合，它可以合理地减少参数的数量，也可以规范化图像的尺寸。实际应用中，比较常用的是最大池化。卷积神经网络的全连接层中的每个神经元将执行全连接操作，其中上一个像素的通道会扩展所有像素，并将所有局部特征组合为全局特征以计算每个类别的分数。

图 2.9 显示了最大池化和平均池化的比较。同一颜色部分中的 4 个元素是通过取最大值或取平均值运算得出相应位置的输出来计算的。

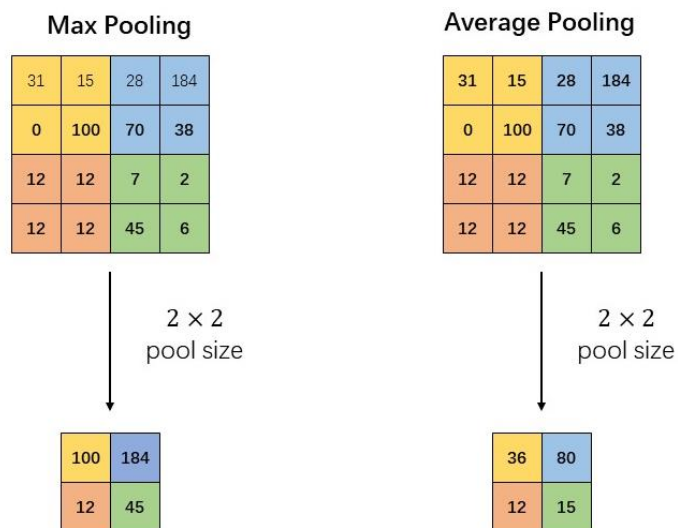


图 2.9 2×2 最大池化和平均池化示意图

## 2. 网络模型 Alexnet

AlexNet<sup>[19]</sup>是 Geoffrey Hinton 和他的学生 Alex Krizhevsky 在 2012 年提出的，成功获得了当年的 ImageNet 图像识别大赛的第一名。图 2.10 为 AlexNet 的结构图：

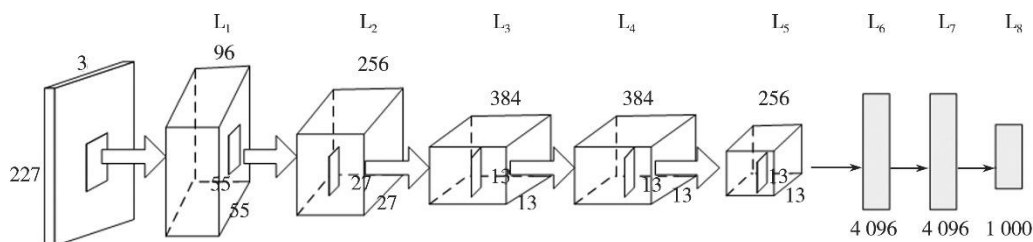


图 2.10 AlexNet

AlexNet 的主要特点主要包括：用双 GPU 进行训练，方便设计更大和更深的网络；使用 Relu 函数代替 Tanh，在一定程度上缓解了梯度消失问题并加快了网络收敛速度；提出局部响应归一化，可以降低错误率并提高网络性能；提出丢弃操作用来减缓过拟合。

## 3. 网络模型 VGG

牛津大学的学者 Simonyan 和 Zisserman 在 2014 年提出了 VGG 系列模型。VGGNet<sup>[20]</sup>在计算机视觉的领域被广大的学者应用，是出于它优良的泛化能力。

VGGNet 的结构如图 2.11 所示。VGGNet 用五组卷积，每部分都由卷积层加上激活

函数组成，并且每部分都有合并的操作。

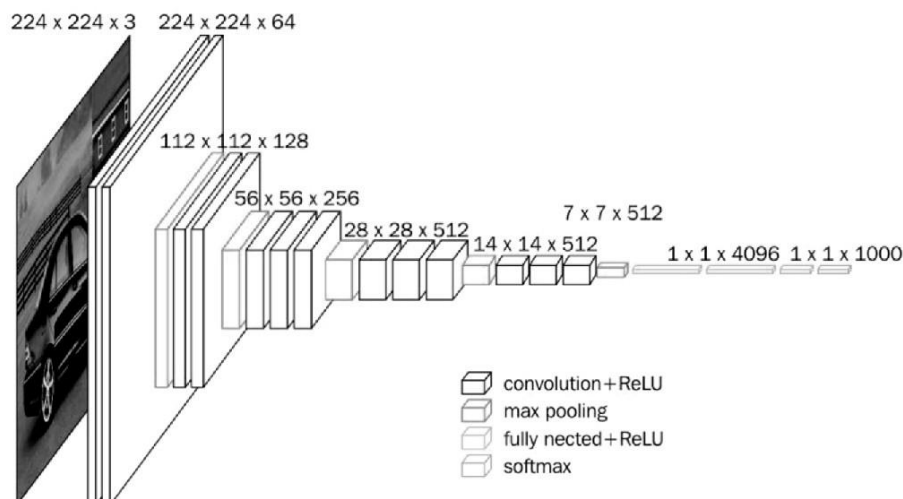


图 2.11 VGGNet 的网络结构

VGGNet 具有以下优点：VGGNet 的参数量减少，因为使用感受野较小的卷积内核（ $3 \times 3$  卷积内核）；提高网络的非线性表达能力。

#### 4. 网络模型 ResNet

残差网络（Residual Network, ResNet）由 Microsoft Research 的何凯明等人于 2015 年提出。第一次成功地训练了一个 152 层神经网络，并在计算机视觉竞赛中取得了较好的成绩。

何凯明等人为了解决网络层数越深而带来的梯度爆炸的问题，提出了残差单元。该残差单元的理论基础是：如果网络已达到较好的准确率，那么网络的性能不会被新添加恒等映射所影响。如下图所示：

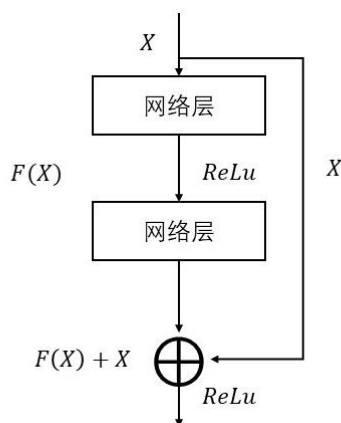


图 2.12 残差网络

残差网络具有以下优点：（1）残差学习缓解了卷积神经网络中性能下降的缺陷；（2）ResNet 具有高度的灵活性，对各项指标的要求从而叠加不同的层数，平衡了性能和速度。

## 第三章 核相关滤波跟踪算法的实现与分析

### 3.1 核相关滤波跟踪算法的基本框架

核相关滤波（High-Speed Tracking with Kernelized Correlation Filters, KCF）跟踪过程如图 3.1 所示。首先，目标的第一帧就作为目标的位置。接下来，就根据运动模型在目标周围获取多个候选框。然后不断更新检测模型和检测到的目标在新的一帧的位置。最后，让当前帧得到的目标跟踪的结果送到下一帧处理，跟踪到视频结束为止。

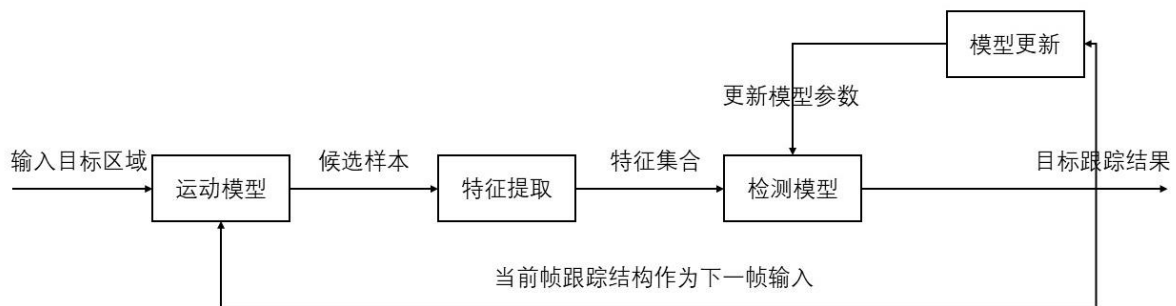


图 3.1 核相关滤波跟踪算法框架

### 3.2 视频信息处理的特征描述

#### 3.2.1 HOG 特征

2015 年，Dalal 等人提出了一种图像描述方法,该方法利用图像局部的梯度方向直方图的计算和统计来形成特征，因此被称为 HOG 特征。HOG 特征对几何变形和光照的影响具有良好的适应性，因此在行人目标跟踪过程中更适合选择 HOG 特征作为样本特征。HOG 特征图通过组合单元单元的梯度方向图来构成整个图像的特征描述符。

Felzenszwalb 等人通过对描述符执行 PCA 降维，形成 31 维特征图，从而生成面向快速直方图的梯度（FHOG）特征。HOG 特征的计算步骤如图 3.2 所示。

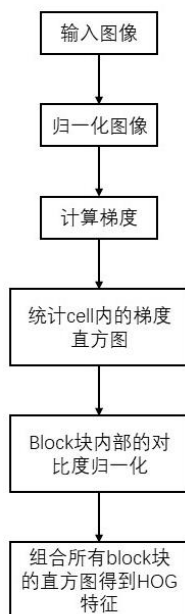


图 3.2 HOG 特征提取步骤

### 3.2.2 LBP 纹理特征

纹理特征是指由像素的灰度分布所表现出的周期性结构，该周期性结构是根据特定规则布置图像的特定结构单元的图案。

局部二值模式 (Local Binary Pattern, LBP) 用于描述图像像素点与相邻之间像素的关系。LBP 具有灰度不变性和旋转不变性的特征。图 3.3 显示了原始 LBP 特征提取原理。我们假设该图的左侧是  $3 \times 3$  图像，并且图中的数字表示像素的灰度值。以 9 个像素的中间像素为阈值，将周围的 8 个位置点的灰度与该中间点的灰度进行比较。如果其值大于中间点值，则将该点记录为数字 1，反之亦然，记录为数字 0。对所有像素进行操作后，将生成右侧所示的图。此时，将每个点的数字组合后产生了一串二进制数字，可以将其转换为十进制数字，从而可以获得中间点的 LBP 值。对于普通图像，将它们转换为灰度图像，并以上述方式处理整个图像，以获得整个图像的 LBP 特征值。对于边缘上的像素，可以适当地扩展图像，并且将扩展区域的像素值设为零。

通常，在分类和识别问题中，将图像分为块，在每个块中计算像素的 LBP 值，并计算该值的直方图统计量，最后将所有块的直方图连接起来，形成整体图像特征。

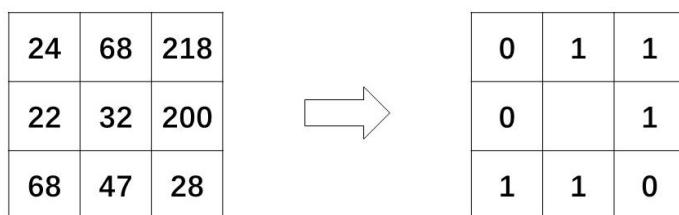


图 3.3 LBP 原理图



LBP 值计算公式如下：

$$LBP(x_G, y_G) = \sum_{p=1}^s 2^p \times s(I(p) - I(c)) \quad (3.1)$$

其中， $p$ 代表  $3 \times 3$  窗口中除中心像素点外的第  $p$  个像素点； $I(c)$  代表中心像素点的灰度值， $I(p)$  代表第  $p$  个像素点的灰度值； $s(x)$  如式(3.2)所示。

$$s(x) = \begin{cases} 0, & x \geq 0 \\ 1, & otherwise \end{cases} \quad (3.2)$$

### 3.2.3 深度特征

机器学习到的深度特征可以更好地对图像进行描述，从而使计算机可以对图像数据有更深刻的理解和更清晰的表达，并且省去了人工提取图像特征步骤。

在所有的深度网络中，卷积神经网络与图像处理关系最密切，而卷积网络在许多图像分类竞赛中都取得了不错的成绩。卷积神经网络将尝试输入图像中的每个可能位置，并使用卷积和合并来提取深层特征，即使用卷积核在图像上滑动，执行卷积操作并获得特征值，再这样对整个图像操作后，形成图像的特征矩阵。同时，从卷积结果上提取每一局部块的最大值，就是最大池化层的操作。卷积神经网络有提取多层特征的能力，其中浅层特征的几何细节信息表示能力强，分辨率高，适合目标定位；深层特征的接收范围较大，语义信息表示能力强，但特征图的分辨率低，更适合目标分类。

## 3.3 核相关滤波算法详述

### 3.3.1 运动模型

KCF 用循环矩阵的方式获得了相应的运动模型。运动模型将根据前一帧的结果生成与当前帧和目标感兴趣区域相对应的训练样本。详细地来看是，根据前一帧跟踪的结果，将该结果周围的区域作为感兴趣的区域的图像块，将其所对应的矩阵转为向量，即转换成  $n \times 1$  的向量  $x$ ， $x = (x_1, x_2, \dots, x_{n-1}, x_n)$ ， $n$  是  $x$  中的元素总个数。当前帧的训练样本就通过将向量  $x$  与(3.3) 所示的置换矩阵  $P$  进行循环移位。

$$P = \begin{bmatrix} 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} \quad (3.3)$$

将  $x$  作为基础样本，右循环移位可以产生新的训练样本，这样可以很快满足样本需求。样本集合表示为  $P$  与  $x$  的乘积  $Px = (x_n, x_1, \dots, x_{n-2}, x_{n-1})$ ，可以表达为通过对目标图

像块进行 $n$ 次循环移位 $\{P^\mu x | \mu = 0, \dots, n-1\}$ ，最终产生了的循环样本矩阵，其样本矩阵如公式(3.4)所示：

$$C(x) = \begin{bmatrix} (P^0 x)^T \\ (P^1 x)^T \\ (P^2 x)^T \\ \vdots \\ (P^{n-1} x)^T \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_n \\ x_n & x_1 & x_2 & \dots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \dots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \dots & x_1 \end{bmatrix} \quad (3.4)$$

式(3.4)中， $C(x)$ 的第一行是基本样本，下一行就是新样本循环矩阵，通过循环移位形成循环矩阵，获取矩阵的所有行，就形成了当前样本集的矩阵。循环矩阵有对角化的性质。

### 3.3.2 特征提取

在目标跟踪过程中，特征描述的好坏直接关系到最终的跟踪结果的优劣。不但要考虑到特征描述信息是否合适，还要考虑到算法的而复杂度，以及是否需要满足实时性的要求。

运动目标的特征包括学习特征和人工特征。学习特征是通过 CNN 训练获得的。统计特征是通过图像的信息而得到的特征，典型的统计特征包括直方图特征。

核相关的滤波算法使用 HOG 特征来提取特征，HOG 通常用于检测目标特征。将图像划分为多个单元，这样方便计算每个像素的梯度直方图，以获得最终的特征描述。它的特点是不受光学失真等的影响，可以在实际应用中提供更好的效果。

### 3.3.3 检测模板

#### 1. 岭回归

简单来说，岭回归，即对于函数 $f(z) = \omega^T z$ ，使训练样本 $x_i$ 与回归标签 $y_i$ 之间的平均误差最小，如公式(3.5) 所示：

$$\min \sum (f(x_i) - y_i)^2 + \lambda \|w\|^2 \quad (3.5)$$

$w$  代表滤波器参数， $\lambda$ 代表正则化的参数， $x_i$ 代表第 $i$ 个训练样本， $y_i$ 代表回归目标模型中的第 $i$ 个回归目标。通过对 $w$ 求导数，可以用下式求得该问题的闭式解：

$$w = (X^H X + \lambda I)^{-1} X^H y \quad (3.6)$$

$X^H$ 是 $X$ 的共轭转置矩阵， $I$ 表示单位矩阵。因为循环矩阵的可对角化性质，可简化以上公式为：

$$C(x) = F \text{diag}(\hat{x}) F^H \quad (3.7)$$

$F$ 是常量矩阵， $\hat{x}$ 是 $x$ 的傅里叶表换， $\text{diag}(\hat{x})$ 为对角阵。代入（3.6）可以得到闭式解

如下式：

$$\begin{aligned}\omega &= (F \text{diag}(\hat{x}^* \odot \hat{x}) F^H + \lambda I)^{-1} F \text{diag}(\hat{x}^*) F^H y \\ &= F \text{diag}\left(\frac{\hat{x}^*}{\hat{x}^* \odot \hat{x} + \lambda}\right) F^H y\end{aligned}\quad (3.8)$$

对 (3.8) 进行傅里叶操作，可以得到频域解：

$$\hat{\omega} = \frac{\hat{x}^* \odot \hat{y}}{\hat{x}^* \odot \hat{x} + \lambda} \quad (3.9)$$

其中， $\hat{\omega}$ ， $\hat{y}$  分别代表  $w$  和  $y$  的傅里叶变换后的结果， $\hat{x}^*$  是  $\hat{x}$  的共轭矩阵。

## 2. 核技巧的运用

当遇到复杂的非线性问题时，运用核技巧可以将低维的样本  $x$  映射到高维度的空间去，这样可以很快很方便的得出最优解。需要一个映射函数为  $\varphi(x)$ ，核函数  $k(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle$ 。其中， $\langle, \rangle$  表示样本  $x_i$  和样本  $x_j$  核函数的内积，具体如下式：

$$K_{ij} = k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j) \quad (3.10)$$

下式用于求解分类器的权重：

$$w = \sum \alpha_i \varphi(x_i) \quad (3.11)$$

$\alpha$  是  $n - 1$  的列向量，通过式(3.12)可以将求解  $\omega$  的问题转换成了求  $\alpha$  的问题，其回归函数如公式(3.12)所示：

$$f(x_j) = \langle \omega, \varphi(x_j) \rangle = \sum_{i=1}^n \alpha_i \langle \varphi(x_i), \varphi(x_j) \rangle \quad (3.12)$$

将组合核函数  $k(x_i, x_j)$  后得到核函数  $K$ ，组合后的核函数  $K$  也满足循环矩阵的特性。

将式(3.12)带入岭回归方程(3.5)中，可求得  $\alpha$  如下式：

$$\alpha = (K + \lambda I)^{-1} y \quad (3.13)$$

## 3. 快速目标检测

在这个阶段可以确定目标位置。根据样本和在线模型的响应值来确定的。

$X$  是当前帧训练样本集， $Z$  是下一帧的候选样本集。 $K^Z$  表示的是两个样本集之间的核矩阵。

$K^Z$  可由式(3.14)计算得到：

$$K^Z = C(k^{xz}) \quad (3.14)$$

将  $Z$  带入回归方程可得到  $Z$  相应的响应值：

$$\hat{y} = K^{xz} \otimes \hat{\alpha} \quad (3.15)$$

$\hat{y}$  和  $\hat{k}^{xz}$  分别表示  $y$  和  $k$  的傅里叶变换， $\hat{\alpha}$  是傅里叶变换后的滤波器模板。

### 3.3.4 模型更新

KCF 算法的模型更新分为目标特征模型参数更新和滤波器更新两个部分，分别以式 (3.16) 和式 (3.17) 表示。

$$x_t = (1 - \mu)x_{t-1} + \mu x_t \quad (3.16)$$

$$\alpha_t = (1 - \mu)\alpha_{t-1} + \alpha x_t \quad (3.17)$$

$x_{t-1}$  是第  $t-1$  帧的目标特征向量， $\mu$  是更新率， $\alpha$  是滤波器系数。

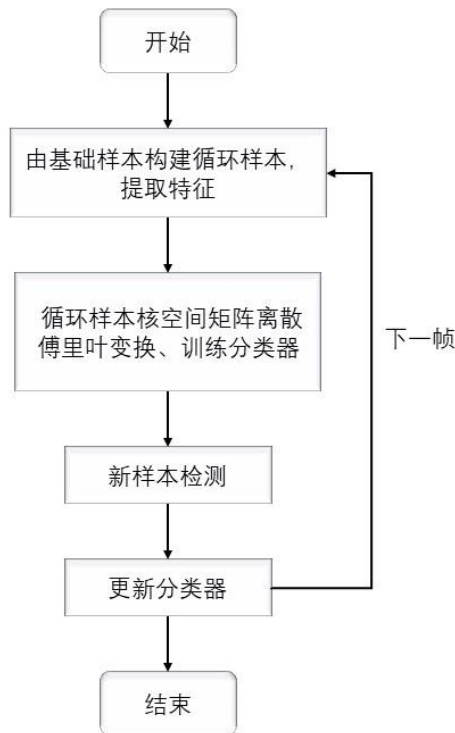


图 3.4 核相关滤波器模型更新的流程图

## 3.4 核相关滤波器跟踪算法跟踪效果展示与分析

### 1. 效果展示与分析



图 3.5 KCF 在 OTB 数据集 Basketball 测试集的跟踪效果展示



图 3.6 KCF 在 OTB 数据集 Bird1 测试集的跟踪效果展示

由图 3.5 可知，目标在篮球场上运动时，由于目标的快速移动以及发生遮挡，KCF 丢失了目标。由图 3.6 可知，当出现目标消失又重新出现的情况时，KCF 会丢失目标，没有办法重新跟踪到目标，从而跟踪失败。

## 2. 关于 KCF 算法的缺点

(1) 通过实验发现，目标跟踪框在整个跟踪过程中都没发生合适的变化，无论跟踪物体是否产生形变。在真实的场景中，目标框的尺寸不发生变化容易导致跟丢目标，发生同等比例的变化利于跟踪。

(2) KCF 算法通过跟踪的结果来训练滤波器和目标模型，这种完全没有考虑到目标丢失或者目标跟踪错误的情况，会导致跟踪器再也无法跟踪到正确的目标。

(3) KCF 算法使用了 HOG 特征这种人工特征来表示目标。这种特征对目标图像的描述力不强并且不够准确。因为在运动中的物体，它会产生不同程度的尺度上或者姿态上的形变，仅仅是人工特征没办法表示目标物体发生的复杂变化。

## 第四章 基于深度学习的目标跟踪算法的实现与分析

### 4.1 ATOM 目标跟踪算法

Martin Danelljan 在 CVPR 于 2019 年提出了基于最大重叠区域的跟踪算法(Accurate Tracking by Overlap Maximization, ATOM)。ATOM 算法的跟踪过程分为目标分类网络、目标评价网络以及网络在线更新。目标分类网络需要区分目标和背景，完成目标的粗略定位。目标评估网络是要精确的确定目标的中心位置，然后确定出目标框的尺寸。目标分类网络是需要根据跟踪到的视频帧来不断在线更新的，这可以让目标分类网络更好的判别出背景和目标。

#### 4.1.1 目标分类网络

目标评估网络和目标分类网络都使用 ResNet-18 进行特征提取。

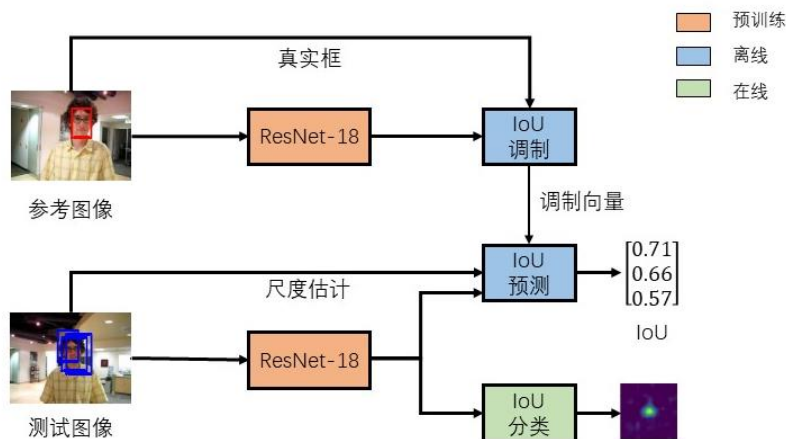


图 4.1 ATOM 结构图

如图 4.1 所示，ATOM 的跟踪流程大概为：通过第一帧的学习得到一个分类器，然后用于测试帧，对于测试帧进行一个置信度的分类，得到一个置信度的分数图，最亮的位置分数最高，然后根据这个位置进一步对目标尺寸进行估计，也就是随机生成 10 个与第一帧的跟踪框相似的候选框，并且把它们输入到 IoU predictor 里面计算它们与真实框的交并比，选取其中分数最高的三个取平均值得到一个框，再对其进行在线微调，得到最终的结果作为最终结果。

绿色的是目标分类网络是一个卷积网络，包括两个卷积层。它是在线训练的。它在跟踪的过程中不断被更新，通过将图像分为前景和背景，提供目标基表粗略的位置，比如图像的坐标。输入是 ResNet 的 block4 层的输出，输出是响应图。

蓝色的目标评估网络提供准确的目标尺度信息，是离线训练的，为了预测跟踪的物体与估计的跟踪框之间的重合度，不断优化跟踪框。

#### 4.1.2 目标评估网络

目标评估网络是用来估计目标尺度，分为两个分支：参考分支和预测分支。参考分支用于提取第一帧的特征信息，然后保存物体的外观信息。预测分支将产生的候选框与并前一帧的特征进行 IoU 值的计算。

用的是 LaSOT 数据集和 TrackingNet 数据来训练的目标评估网络。参考帧和测试帧作为图像对来输入到网络中进行训练。参考帧是通过对目标周围 5 倍大小的正方形区域进行提取得到的。测试帧是视频序列中的图片。对于每个图像对，会生成 16 个候选框，设置生成的候选框与真实的框的交并比的阈值为 0.1。

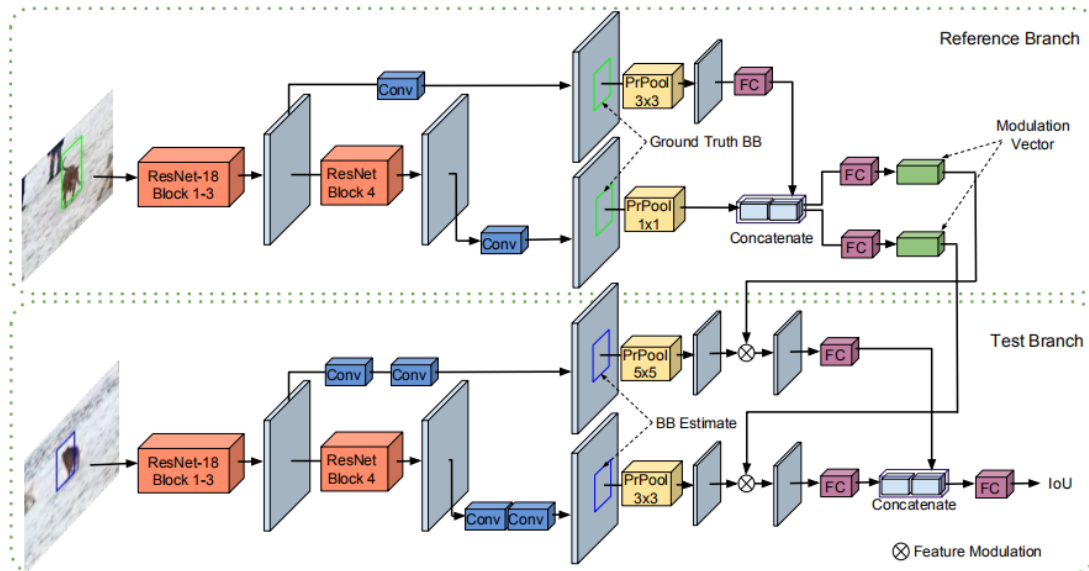


图 4.2 ATOM 目标评估网络

在参考分支中，网络提取了 block3 和 block4 的特征，然后输入到卷积层中获得对应的响应图。然后对特征进行池化操作并输出比例相同的两个局部特征图。两者进行卷积操作进行特征融合，然后各自进行全连接操作后各输出一个调制向量。

在预测分支中，网络也是提取了和参考分支中一样的 block3 和 block4 的特征，然后输入到卷积层获得特征图。对特征图进行池化操作得到两个比例相同的特征图，将参考分支中输出的两个调制向量分别作用不同的通道，给每个通道进行权重的分配。最后经过特征融合和全连接层后输出交并比的分数。

计算交并比的公式为：

$$\text{IoU}(B) = g(c(x_0, B_0) \cdot z(x, B)) \quad (4.1)$$

其中， $c(x_0, B_0)$  表示的是调制向量， $z(x, B)$  表示的是预测分支经过池化后的特征，



$g(\cdot)$ 表示的是全连接层。

### 4.1.3 网络在线更新

此阶段是为了解决当目标遇到干扰物而导致跟踪不准的问题。这种问题的出现一般由于参数过多和训练样本不足。在线更新是根据之前的跟踪结果来微调网络，让网络对当前的视频更加适应。

在 ATOM 算法中，网络每 10 帧更新一次，样本聚类方法采用的是一种基于高斯混合模型的样本聚类算法。

## 4.2 ATOM 算法跟踪效果展示与分析

目标分类网络第一层卷积层是为了给输入特征降维，用的卷积核的大小为  $1 \times 1$ 。第二个卷积层的输出为目标分类响应图。卷积层之间的激活函数使用 PReLU<sup>[17]</sup>。在跟踪视频的第一帧中，使用数据扩增，并对卷积参数的第一层进行初始化和优化。在之后的在新更新过程中，仅对第二层卷积参数进行优化。

目标评估网络的设计亮点是 IoUNet，该网络使用 PrRoI Pooling 池化操作，如图 3.1 所示。

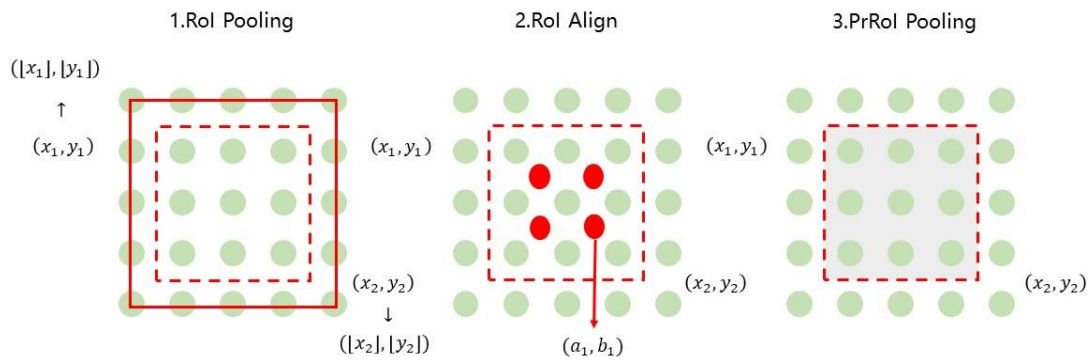


图 4.3 不同池化操作示意图

图 4.3 显示了不同的 RoI Pooling 操作。RoI Pooling 是由 Faster R-CNN 网络提出的，它可将不同尺度的特征图的输出得到统一尺度的输出。Mask R-CNN 网络中提出了 RoI Align，整个 RoI 被划分成 4 个区域，每一部分的中心点作为采样点通过双线性插值计算，以获得尺度统一的特征图。IoUNet 提出了 PrRoI Pooling 操作，通过双线性插值操作，feature map 上原本离散的特征值，可以再任意连续坐标  $(x, y)$  上得到连续性的结果。

在 ATOM 跟踪算法中，它对目标尺度的变化具有很好的适应性。在 Basketball 视频



序列中，人没有被遮挡的时候，或者说遮挡的面积比较小的时候，跟踪的效果很好。

在 Bird 视频序列中，可以看到当鸟的外形尺寸发生变化是，跟踪器也跟踪的很好。但是当出现目标消失又重现的情况时，出现了目标被跟丢的情况。随着跟踪过程的进行，这些已经跟丢目标的样本会决定目标分类网络最终的分类效果，导致分类器无法正确判别出背景和目标后产生错误的输出结果。但当出现很多个与目标相似的物体干扰时，跟踪器的表现一般，虽然丢失跟踪目标，但对于目标的尺度估计并不准确。



图 4.4 ATOM 算法在 OTB 数据集 Basketball 序列上的跟踪结果

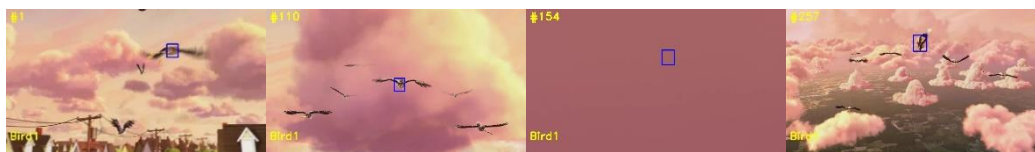


图 4.5 ATOM 算法在 OTB 数据集 Bird 序列上的跟踪结果

基于以上分析，对目标评估网络进行训练得到的模型对于常见的目标尺度变化能够较好地适应。但是当出现与目标相似的物体干扰、目标遮挡和目标消失执行该模型时效果并不好，原因是目标分类网络的判别能力不强，容易出现粗略的目标定位误差，无法为目标评估网络输入正确的粗略的目标定位结果，从而导致跟踪失败。

## 第五章 实验分析

### 5.1 实验设置以及数据集评价指标

实验的具体配置如表 5.1 所示:

表 5.1 实验环境设置表

项目	配置
处理器 (CPU)	Intel(R)Core(TM)i5-7200CPU@2.50GHz×4
显卡 (GPU)	GeForce 920MX
内存 (Memory)	3.7GB
操作系统 (OS)	Ubuntu 18.04
框架	PyTorch

OTB-100 数据集主要使用两种评价标准, 分别为准确率 (precision plots) 和成功率 (success plots)。成功率是用于衡量算法效果的另一个指标。它的计算方法是找到算法跟踪结果与手动标记结果之间的重叠率, 即 IoU。可以通过设置不同的重叠率来获得成功率曲线, 并使用成功率曲线下的面积大小 (Area under Curve, AUC) 作为评估指标, 可以解决主观描述过多的问题, 并可以有效地区分不同的跟踪算法表现。准确率是用于衡量算法跟踪结果中心位置准确性的指标, 它通过计算人类手动标记的目标的中心坐标与跟踪算法结果的中心坐标之间的欧式距离来评估算法, 并计算所有测试视频帧的中心坐标的平均误差以对该算法进行整体评估。

### 5.2 对比实验与分析

#### 5.2.1 主观结果与分析

在三个目标跟踪数据集上, KCF 算法与 ATOM 算法主观对比如下。

##### (1) Basketball



图 5.1 不同算法在 Basketball 测试集的主观效果: 蓝色, 绿色的矩形框分别代表 KCF 和 ATOM 的跟踪结果

##### (2) Blurcar1



图 5.2 不同算法在 Blurcar1 测试集的主观效果：蓝色，绿色的矩形框分别代表 KCF 和 ATOM 的跟踪结果

### (3) Biker



图 5.3 不同算法在 Biker 测试集的效果：蓝色，绿色的矩形框分别代表 KCF 和 ATOM 的跟踪结果

图 5.1，图 5.2，图 5.3 显示了针对三个数据集使用不同的跟踪方法进行目标跟踪的一些示例，蓝色，绿色的矩形框分别代表 KCF 和 ATOM 的跟踪结果。从图 5.7 可以看出，当运动员在空间中做运动时，场景较复杂。当目标被遮挡，KCF 会跟踪失败，并且后续一直跟丢目标，KCF 的尺度一直没有发生变化。ATOM 会在出现人物遮挡、快速移动的情况下跟踪的比较好。在图 5.2 中，出现车辆模糊的情况，KCF 和 ATOM 的跟踪效果都挺好。在图 5.3 中，当出现目标的快速移动、旋转的时候，KCF 出现了失去目标的情况，ATOM 虽然可以跟踪目标，但对于目标尺度的估计并不理想。

## 5.2.2 客观结果与分析

### (1) Basketball

表 5.2 在 Basketball 测试集上跟踪结果

方法	成功率	精确率
KCF	0.224	0.248
ATOM	0.527	0.727

### (2) Blurcar1

表 5.3 在 Blurcar1 测试集上跟踪结果

方法	成功率	精确率
KCF	0.767	0.865
ATOM	0.797	0.911

### (3) Couple

表 5.4 在 Biker 测试集上跟踪结果

方法	成功率	精确率
KCF	0.467	0.653
ATOM	0.771	0.934

本文分别对两个跟踪算法在三个数据序列上各进行了 5 次实验，表 5.2 到表 5.3 显示了两个算法在不同序列上面的成功率、精确率，本文是将传统的基于相关滤波的目标跟踪算法 KCF 和基于深度学习的跟踪算法 ATOM 进行了比较。在三个具有挑战的数据集上，ATOM 的成功率和精确度高于 KCF 算法的性能，ATOM 在精确度上比 KCF 高约 33 个百分点，在成功率上约高 26 个百分点。ATOM 在 Basketball 和 Blurcar1 数据集上达到了比较好的性能，KCF 则在 Basketball 和 Biker 的性能上展现出不好的跟踪性能。

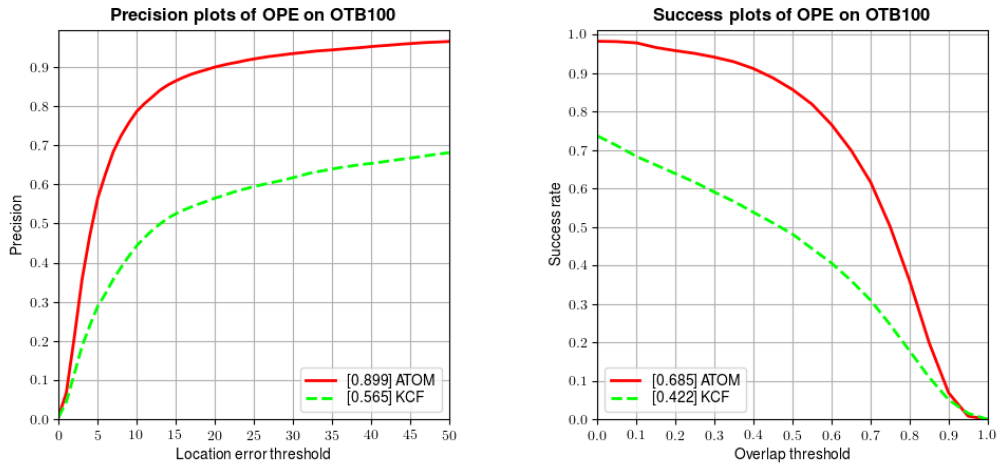


图 5.4 跟踪算法对比实验结果图

由图 5.4 分析可得，ATOM 算法在成功率和精确度上相较于 KCF 算法有很显著的优势。分析原因是，KCF 算法使用的人工特征并且尺度不变使其在面对相似干扰、目标遮挡、目标消失没有良好的适应性。而相对于 KCF 算法，ATOM 算法提取卷积网络深层次特征，并且采取了基于调制向量的网络策略利用第一帧的信息来优化目标的尺度信息，使其能够在成功率和精确度上优于 KCF 算法。

## 第六章 总结与展望

### 6.1 本文工作总结

视觉对人类的重要性不言而喻，人们通过视觉从自然界中捕获各类信息。利用计算机进行追踪在近几年来变得火爆起来。出现了一些跟踪效果比较杰出的相关滤波的跟踪算法，原理是根据人工提取的特征与相关滤波器在频域中进行操作得到响应图，响应值最大的点作为目标的位置。深度学习可以更加充分的描述图像的信息和特征。随着深度学习的发展，深度特征取代了手工特征，这大幅度的提高的跟踪的成功率和精确度。

本文主要研究了基于相关滤波器的目标跟踪算法 KCF 算法和 ATOM 算法。对于基于深度学习的目标跟踪问题，有如跟踪流程的控制等方向是具备研究价值的。

### 6.2 未来工作展望

目前，基于深度学习的目标跟踪的方法仍然主要停留在基于 ImageNet 预训练的特征的应用层面。近年来，ImageNet 的视频数据集已逐渐用于学习更合适的深度特征以进行目标跟踪，并且已经取得了一定的进展。伴随着 TrackingNet 等大规模数据集的出现，有可能基于海量跟踪视频来端到端学习深度特征，这有望进一步促进深度学习在视觉目标跟踪方面的突破。相对而言，标注长程的跟踪视频和构建大规模数据集更加困难，如何根据长程跟踪任务的特点及其与短程跟踪任务的联系，结合转移学习和深度学习构建合适的长期目标跟踪模型，是未来视觉目标跟踪研究的重要方向。

## 参考文献

- [1] 高文,朱明,贺柏根,吴笑天.目标跟踪技术综述[J].中国光学,2014,7(03):365-375.
- [2] 李均利,尹宽,储诚曦,汪鸿年.视频目标跟踪技术综述[J].燕山大学学报,2019,43(03):251-262.
- [3] Comaniciu D , Ramesh V , Meer P . Real-time tracking of non-rigid objects using mean shift[C]// Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. IEEE, 2003,2000,2:142-149.
- [4] Kalman R E . A New Approach To Linear Filtering and Prediction Problems[J]. Journal of Basic Engineering, 1960, 82D:35-45.
- [5] Yacoob Y , Davis L S . Recognizing human facial expressions from long image sequences using optical flow[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on, 1996, 18(6):636-642.
- [6] Babenko B, Yang M H, Belongie S. Visual tracking with online Multiple Instance Learning[J]. IEEE Trans Pattern Anal Mach Intell, 2009:983-990.
- [7] Kalal Z, Matas J, Mikolajczyk K. Online learning of robust object detectors during unstable tracking[C]. Computer Vision Workshops (ICCV Workshops),2009 IEEE 12th International Conference on. 2009:1417-1424.
- [8] Struck: Structured Output Tracking with Kernels[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015:2096-2109.
- [9] Bolme D S, Beveridge J R, Draper B A, et al. Visual object tracking using adaptive correlation filters[C]. IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2010:2544-2550.
- [10]Henriques J F, Caseiro R, Martins P, et al. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels[J]. Lecture Notes in Computer Science,2012,7575(1):702-715.
- [11]João F. Henriques, Caseiro R, Martins P, et al. High-Speed Tracking with Kernelized Correlation Filters[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015,37:583-596.
- [12]Marcel V G, Sander B. Editorial: Artificial Neural Networks as Models of Neural Information Processing[J]. Frontiers in Computational Neuroscience, 2017, 11:114.
- [13]Ryu I H, Won I, Kwon J. Detecting Ghost Targets Using Multilayer Perceptron in Multiple-Target Tracking[J]. Symmetry, 2018, 10(1):16.
- [14]Ronald J. Williams, David Zipser. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks[J]. Neural Computation,1989,1(2):270-280.
- [15]Lecun Y, Bottou L. Gradient-based learning applied to document recognition[J].

Proceedings of the IEEE, 1998, 86(11):2278-2324.

[16] A. Iliev, N. Kyurkchiev, S. Markov. On the approximation of the step function by some sigmoid functions[J]. North-Holland, 2017, 133:223-234.

[17] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition[J]. Neural Computation, 1989, 1(4):541-551.

[18] HUBEL D H, WIESEL T N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex.[J]. The Journal of physiology, 1962, 160-154.

[19] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet Classification with Deep Convolutional Neural Networks [J]. Advances in Neural Information Processing Systems, 2012, 2:1097-1105.

[20] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2015:1-14.

## 致 谢

毕业设计接近尾声，我在此感恩天津理工大学四年对我的栽培，让我不仅丰富了学术知识，还让我体验了大学的美好生活。特别感谢在本次毕业设计中给予我们热心帮助和细心指导的薛彦兵老师，老师对待学术兢兢业业刻苦钻研的精神使我记忆深刻。在学习上和生活上收到的老师悉心的指导与帮助，是我大学四年收获的珍宝一样的礼物。也特别感谢我的学长丁明远、学姐李思敏以及陈妍同学，他们在我毕设阶段给过我很多中肯的意见以及很大的鼓励，我将以他们为榜样继续我的学习生涯，不断对自己喜欢的领域投入无限热爱，争做一个明亮优秀的人！