硕士专业学位论文

电子价签检测算法研究与系统实现

Algorithm Research and System Implementation of Electronic Shelf Label Detection

作者：韩致远

导师：郭宇春

北京交通大学

2020 年 5 月

学位论文版权使用授权书

本学位论文作者完全了解北京交通大学有关保留、使用学位论文的规定。特授权北京交通大学可以将学位论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。学校可以为存在馆际合作关系的兄弟高校用户提供文献传递服务和交换服务。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：　　导师签名：

签字日期：　　年　　月　　日　　签字日期：　　年　　月　　日

学校代码：10004 密级：公开

北京交通大学

硕士专业学位论文

电子价签检测算法研究与系统实现

Algorithm Research and System Implementation of Electronic Shelf Label Detection

作者姓名：韩致远 学 号：18125015

导师姓名：郭宇春 职 称：教授

专业学位类别：电子与通信工程 学位级别：硕士

北京交通大学

2020 年 5 月

致谢

本论文的研究工作是在我的导师郭宇春教授的悉心指导下完成的，郭宇春教授开拓的学术思维、科学的工作方法和严谨的治学态度对我产生了极大的影响。当我在科学研究中遇到困难时，郭老师给予了我许多帮助，同时培养了我认真严谨的科研态度以及面对困难积极进取的精神。由衷感谢郭老师这两年来对我在科研上的指导和生活上的关心。

同样感谢陈一帅老师对本次科研工作的帮助和指导。读研期间，多次和陈老师一起讨论学术问题，陈老师总能提出一些新颖独到的见解，帮助我解决心中的疑惑。陈老师饱满的生活热情和积极的工作态度将会一直是我的学习榜样。

感谢实验室的所有老师，感谢赵永祥、李纯喜、郑宏云、张立军、孙强等老师在我研究生阶段对我的帮助，在此向各位老师表示衷心的感谢。同时感谢陈滨师兄、于兹灏师兄、魏中锐师兄、冯梦菲师姐、盛烨师姐、苏迪学姐对我学习过程中的无私帮助。同样也感谢曹中、王亚珊等同学对我学习和生活中的帮助，感谢你们陪我度过这段难忘的学习生涯。

感谢一直无微不至地关心、支持我的父母和其他亲人朋友，正是他们不断的鼓励和无私的付出，才使得我不断克服科研中的困难，并顺利地完成学业，成为社会的有用之才。

最后感谢北京交通大学这两年来对我的栽培，感谢电子信息工程学院为我提供的学习机会，感谢网络与安全实验室为我提供了良好的环境去不断试验和创新。

摘要

随着实体零售和传统电商迎来增长“瓶颈”，“新零售”的崛起为销售市场注入了新的活力。电子价签作为一种实时价格显示装置，成为了“新零售”最基础、最重要的元素之一。电子价签的自动视觉检测能够定位商品，实现货架的自动陈列，对支撑“新零售”体系线上和线下联动至关重要，同时也极具挑战。目前已有的目标检测方法并不能很好地检测出电子价签，基于区域生成的深度学习目标检测算法虽然检测精度较高，但不满足实时性的要求；基于回归方法的深度学习目标检测算法不能较好地提取出小尺度、高密度检测对象的特征。

针对上述问题，本文提出了一种新的特征融合方法，通过两次融合相邻的特征图，能够更好地融合低层特征图的位置信息和语义信息，提升对区域密集目标的检测精度。接着本文引入了注意力模块，进一步提升了对小尺度目标的检测效果。具体贡献如下：

(1) 大规模电子价签数据集的制作和检测系统的设计。本文完成了对电子价签数据集的采集、筛选以及标注工作，该数据集包含 14713 张图片。接着本文根据设计需求，设计了电子价签检测系统，并对各功能模块进行详细说明。

(2) 针对电子价签数据集检测对象尺度小、密度高等特点，提出了一种新的特征融合方法。该方法两次对相邻的特征图进行特征融合，使得网络更加关注低层特征图的位置信息和语义信息，提升对小尺度目标的检测精度。同时两次融合的低层特征图有部分重合，使得网络更加关注区域密集目标，提升对高密度目标的检测精度。实验结果显示：该方法相对 SSD 算法在性能上有显著提升，mAP 值提高了 8.8%。

(3) 在特征融合的基础上本文进一步引入了注意力机制，使特征融合更关注较低层特征图，进一步提升对小尺度目标的检测精度。实验结果显示：引入注意力模块可以进一步提示模型检测的性能，mAP 值提高 2.8%，共提升 11.6%。

(4) 本文设计和实现的电子价签检测系统有助于实现“新零售”体系线上和线下联动，完成对商品的自动定位、货架的自动陈列，具有重要的经济价值、社会价值。

图 29 幅，表 8 个，参考文献 53 篇。

关键词：深度学习；目标检测；特征融合；注意力

ABSTRACT

With the growth bottleneck arising in brick-and-mortar retailers and traditional e-commerce industry, the rise of “new retailing” has brought new energy into the sales market. As a real-time price display device, electronic shelf label becomes one of the most basic and important elements of “new retailing”. The automatic visual detection of electronic shelf label can locate goods and realize the automatic display of shelves, playing an important role in supporting the online and offline linkage of “new retailing” system. Existing object detection methods cannot detect the electronic shelf label very well. Although the deep learning object detection algorithm based on region proposal achieves high accuracy, it does not meet the real-time requirements. While, the algorithm based on regression method cannot extract the features of small and high-density objects well.

Aiming at solving the fore mentioned problems, this thesis proposes a new feature fusion method, which can better fuse the location information and semantic information of the low-level feature map by fusing the adjacent feature map twice, and therefore improve the detection accuracy of high-density objects. Furthermore, the attention module is introduced to improve the detection effect of small object. The contribution of the thesis can be summarized as follow:

(1) Production of massive electronic shelf label data set and design of detection system. This thesis competes the collection, filtering and annotation of the electronic shelf label data set, which contains 14713 pictures. In addition, according to the design requirements, this thesis designs an electronic shelf label detection system and describes all functional modules in detail.

(2) According to the characteristics of electronic shelf label, such as small scale and high density, this thesis proposes a new feature fusion method. This method fuses features of adjacent feature map twice, which makes the network pay more attention to the location information and semantic information of the low-level feature map, so that the detection accuracy of small objects is improved. At the same time, the low-level feature map fused once overlaps the one fused twice, which makes the network pay more attention to high-density objects and improves detection accuracy of high-density objects. Experiments prove that the performance of this method is significantly improved compared with SSD algorithm, and the mAP value is increased by 8.8%.

(3) On the basis of feature fusion, this thesis adds attention mechanism to fused feature maps, as a result, the fusion process pays more attention to low-level feature map. Attention module further improves the detection accuracy of small objects. Experiments prove that the method, with attention module introduced, can prompt the performance of detection, and the mAP value is increased by 2.8%, totally the proposed method exceeds the baseline model by 11.6%.

(4) The electronic shelf label detection system designed and implemented in this thesis is helpful to realize the online and offline linkage of the "new retailing" system. Besides, it completes the automatic location of goods and display of shelves. Therefore, electronic shelf label detection system has important economic and social value.

29 figures, 8 tables, 53 reference articles.

KEYWORDS：Deep learning; Object Detection; Feature Fusion; Attention

目录

1 引言

1.1 研究背景及意义

近年来，实体零售和传统电商的销售额增长速度逐渐下降，销售市场接近饱和。一方面，实体零售行业由于互联网的冲击以及本身经营模式的单一性，导致客源流失、竞争激化，改革转型已经迫在眉睫；另一方面，随着互联网的全面普及与发展，传统电商的用户增长速度正在逐渐放缓，互联网带来的流量红利也在逐渐萎缩。传统电商经过长时间的全速发展，不可避免地迎来了销售额增长的瓶颈期。根据国家商务部发布的《中国零售行业发展报告（2018/2019 年）》[1]，实体零售销售额增速已经跌至 8.9%，传统电商的销售额增速也从 2010 的 96.9%锐减至 23.9%，如图 1-1 和图 1-2 所示。

图 1-1 2010-2018 年商品零售额及增速
Figure 1-1 Retail Sales and Growth in 2010-2018

“新零售[2][3]”的出现为实体零售和传统电商提供了新的转型方向。“新零售”不同于传统电商和实体零售，是一种全新的零售行业表达方式。“新零售”将互联网的先进技术和新兴思想融入到传统的零售方式中，将传统的零售方式变得更加智能化、便捷化。“新零售”不仅是平台线上线下和物流的简单组合，还需要融入人工智能、大数据、云计算等新兴技术，从而提升用户的实际消费体验。“新零售”体系的发展需要智能的软硬件支持，电子价签作为一种实时价格显示装置，成为了最基础、最重要的元素之一。电子价签通过无线网络连接到数据库，能够在线或离线访问。不同于传统的纸质价签，电子价签可以实时准确地显示产品的各类信息。

电子价签由于其特点和优势，成为了智能化管理，线上线下一体化的基础和关键所在。与此同时，电子价签还可以完美解决线下门店商品价格变动频繁的问题，降低人工成本，提高门店运营效率。

图 1-2 2010-2018 年网上零售额及增速
Figure 1-2 Online Retail Sales and Growth in 2010-2018

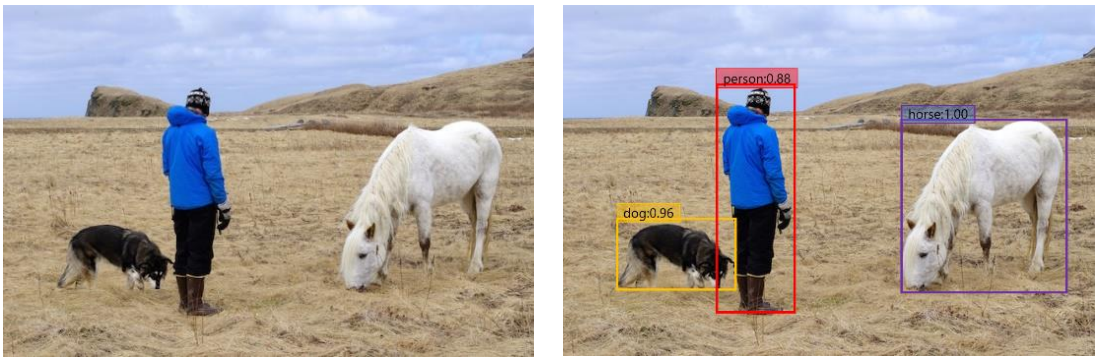
电子价签的自动视觉检测能够定位商品，实现货架的自动陈列，对支撑“新零售”体系线上和线下联动至关重要，同时也极具挑战。因为商品的库存量单位（Stock Keeping Unit,SKU）繁多，平均一个线下商场的商品种类大概在 4 万~5 万个，而目前对细粒度的物体识别还很困难，所以很难直接检测出所有商品的类别信息。因此可以通过检测电子价签的方式来识别对应的商品，根据检测的结果，可以进一步实现三字码检测、缺货检测等功能，从而实现对商品的实时陈列信息监控、智能化管理，大量减少人力成本，提升运营效率。

电子价签检测可以划分为目标检测领域的一个下游任务。目标检测领域发展至今，主要分为两种：第一种是传统目标检测方法，第二种是基于深度学习的目标检测方法。传统目标检测方法主要由三个步骤组成：区域选择，特征提取，分类器分类。传统目标检测方法现在已经逐步被基于深度学习的目标检测方式所取代。一方面，传统目标检测方法是基于滑动窗口的区域选择策略来对目标进行定位的，这种方法不能精度定位检测对象，且生成的窗口过度重复，导致模型的检测效率低下。另一方面，传统目标检测方法需要手动对图像的特征进行提取，例如检测目标的颜色特征、纹理特征等，当检测目标的环境发生改变时，检测效果会变得很差。而本文使用的电子价签数据集具有场景多变，背景复杂，检测目标像素值少、分布密集、数量极多等特点，因此传统的目标检测方法不能准确地检测出所有的电子价签目

标。虽然基于深度学习的目标检测方法近几年发展迅速，但目前市场上还没有有效针对这种小尺度、高密度数据集的目标检测方法。

针对上述问题，本文将设计一个有效的电子价签检测系统，实现对电子价签的自动视觉检测。本文工作的第一个部分是获取、筛选和标注真实的电子价签图片，并制作成合适的数据集用于后续的模型实验。第二个部分是根据实际需求设计电子价签检测系统，完善检测系统各个模块的功能，对数据和接口进行定义，保证系统能够正常运行。第三个部分是设计合理的基于深度学习的电子价签检测算法，与现有的目标检测算法进行对比，提升模型的检测精度和检测速度。第四个部分是对模型进行优化，进一步提升模型的整体性能。本文的研究工作有助于搭建实时的电子价签检测系统，实现对电子价签的自动视觉检测，进一步完成对货架商品的自动化陈列，具有重要的经济价值和社会价值。

1.2 国内外研究现状



a) 输入图像
a) Input Image

b) 检测结果
b) Result of Detection

图 1-3 目标检测
Figure 1-3 Object Detection

目标检测作为计算机视觉的重要分支之一，被广泛应用于多个工业领域，例如人脸识别、汽车自动驾驶、智能视频监控、机器人导航等。如图 1-3 a)和图 1-3 b)所示，目标检测方法不仅可以实现对图片中每个目标的识别和分类，还可以在目标四周标注出尺度、长宽比合适的矩形边界框，实现目标定位功能。不同类别的物体具有各种各样的外观、形状和姿态，这些都会对目标检测造成干扰，即使是同类别的物体，也会因为颜色、大小等特征的差异影响目标检测的结果。另外，目标检测还容易受到光照、遮挡等外界因素的干扰，因此目标检测一直是计算机视觉领域中最具挑战性的问题。通常准确性和实时性是衡量目标检测系统性能的重要指标。自从深度学习诞生以来，许多公司和团队就被吸引并加入到这个领域的研究。随着物

理基础设施的提升、深度学习应用的快速发展，新的目标检测方法也层出不穷，不断刷新检测的精度和速度。大数据时代，随着互联网的全面普及，市场上对于目标检测下游任务的需求也在不断增长。本节将会详细介绍目标检测研究的现状，包含传统的目标检测方法和基于深度学习的目标检测方法，并进一步分析这些现有目标检测研究的特点。

1.2.1 传统目标检测方法

传统目标检测方法主要由选择目标区域，手工特征提取，建立分类器对目标进行分类三个部分组成。

待检测的目标会出现在图片里的任意位置，因此需要对图片中的目标进行精确定位，也就是生成相关区域。最早的传统目标检测利用穷尽的思想，使用滑动窗口遍历整张图片，获得所有可能存在检测目标的区域。滑动窗口的形状可以通过设置不同的尺寸、长宽比来改变。使用滑动窗口的方法大概率可以获得目标所在区域，但是会产生大量的无关区域，时间复杂度过高，降低了后续特征提取、分类器分类的速度，严重影响了整体模型的性能。为了解决这一问题，传统的目标检测方法逐渐衍生出了通过颜色聚类、边缘聚类来生成区域的算法，例如 Selective Search[4]、Edge Box[5]、BING[6]、BING++[7]等。不同于滑动窗口遍历的方法，这类算法不需要重复遍历整张图片，能够快速筛除不相关的区域，从而起到提升检测速度的作用。

2013 年以前目标检测领域获取特征主要基于手工特征提取的方法。整体模型的检测效果在很大程度上取决于手工提取的特征能否准确表达检测目标的特征以及包含的信息，因此手工特征的设计十分重要。文献[8]提出了一种计算量小，能够充分表达边缘变化信息的 haar 特征（haar-like feature），文献[9]设计了一种具有尺度不变性的 SIFT 特征，文献[10]设计的 LBP 特征，提升了对物体纹理信息的表达。最具代表性的是文献[11]提出的 HOG(Histogram of Oriented Gradient, HOG)特征。HOG 特征能描述检测目标的形状边缘特征，并实现对目标的精确检测。HOG 特征主要是通过图像的梯度信息来获取检测目标的边缘信息，并利用图片局部梯度的信息获取目标的各类特征。HOG 特征利用直方图对物体的边缘信息进行编码，因此对特征有更强的表达能力。

传统目标检测方法最后需要通过分类器对筛选出的区域进行分类。常用的分类器有：SVM[12,13]，自适应增强[14]（Adaptive Boosting, Adaboost），随机森林[15]（random froest, RF）等。SVM 分类器能够使用核函数将低维度的空间映射到更高维度的空间，较好地解决非线性、小样本等问题。自适应增强算法使用不同的弱分类器进行训练，接着整合所有的弱分类器，获得一个性能更强的分类器，从而提升

整体的检测性能。随机森林是一种集成算法，利用多棵决策树[16,17]对数据进行训练并预测，最终取众树的结果作为输出。随机森林的输出不是由一颗决策树决定的，而是联合了所有决策树的判决结果。因此随机森林相较于单棵决策树，具有更高的分类准确率。

1.2.2 基于深度学习的目标检测方法

2013 年的前几年，传统目标检测算法的发展就已经停滞不前。但随着卷积神经网络模型 AlexNet[18]在 2012 年的 ImageNet 分类任务比赛中取得巨大成功，之后涌现出的 VGGNet[19]、Google Inception Net[20]、ResNet[21]也不断刷新计算机视觉领域分类任务的成绩，自此目标检测领域又有了新的发展方向。Girshick 研究团队在 2013 年将卷积神经网络（Convolutional Neural Networks, CNN）[18-22]引入到目标检测中，计算机视觉领域才突破了传统目标检测的束缚，进入了基于深度学习的目标检测算法新时代。受卷积神经网络的启发，Girshick 团队在 2013 年首次提出了区域卷积网络目标检测框架[23]（Region-based Convolutional Neural Network, R-CNN），自此基于深度学习的目标检测方法开始了迅速的发展。相较于传统目标检测，基于深度学习的目标检测方法可以自动从简单特征中提取出更加复杂的深层特征，能够适应更加复杂的场景，不仅学习成本更低，鲁棒性也更强。

基于深度学习的目标检测方法发展至今，形成了基于区域生成（Region Proposal）的深度学习目标检测算法，如 R-CNN，SPP-Net[24]（Spatial Pyramid Pooling in Deep Convolutional Network），Fast R-CNN[25]，Faster R-CNN[26]等；以及基于回归方法的深度学习目标检测算法，如 YOLO[27]（You Only Look Once）系列，SSD[28]（Single Shot Multibox Detector）系列等。

基于区域生成的深度学习目标检测算法是一种两阶段方法，需要先生成候选区域，再进行特征提取。R-CNN 算法主要由卷积神经网络和候选区域网络两部分组成。R-CNN 需要将候选框裁剪、放缩至相同尺寸，再使用卷积神经网络对候选框内容进行特征提取。这一操作严重影响了模型的检测速度。为了解决这一问题，Kaiming He 等人提出了 SPP-Net 模型，在特征图上引入了金字塔池化层(Spatial Pyramid Pooling, SPP）的概念，检测模型可以在同一张图片上提取特征，大幅提升了检测速度。Fast R-CNN 算法借鉴了 SPP-net 算法中金字塔池化的思想，提出了感兴趣区域池化层（Region of Interest Pooling, ROI Pooling），进一步简化了模型的复杂度。同时 Fast R-CNN 采用了多任务训练的模式，将定位损失也加入到总损失函数中，使分类和定位同时进行网络训练。之后 Girshick 团队又提出了新的检测框架 Faster R-CNN，Faster R-CNN 是第一个准实时（17 帧/秒）的深度学习目标检测

算法。Faster R-CNN 模型相当于候选区域网络（Region Proposal Network）和 Fast R-CNN 网络的进一步融合。Faster R-CNN 替换了传统的区域生成方法，使用候选区域网络来生成预选框，这一操作节省了区域生成的时间成本，提升了模型的检测速率。除了 R-CNN 系列，基于区域生成的目标检测算法还有许多其他优秀的工作。例如文献[29]提出的 segDeepM（Exploiting Segmentation and Context in Deep Neural Network）算法，将不同特征层的信息引入到模型检测中。文献[30-32]分别提出了 ION（inside outside network）算法，GBD-Net（Gated Bi-Directional Convolutional Neural Network）算法，AC-CNN（Attentive Contexts in Convolutional Neural Network）算法，这些算法成功向 Fast R-CNN 模型中添加了上下文信息，并且融合了多层感兴趣区域上获得的特征，提升了模型检测的准确性。HyperNet[33]算法集合了多层的特征图，得到包含多尺度信息的超特征（Hyper Feature），不同于 Faster R-CNN，HyperNet 算法更有利于检测小尺度物体。同年相关的工作还有文献[34-36]提出的 G-CNN 算法，LocNet 算法，MS-CNN 算法等。

如果说基于区域生成的目标检测算法将深度学习引入了目标检测领域，那么基于回归方法的目标检测算法则实现了目标检测的实时性，使得传统目标检测方法逐渐被深度学习方法所代替。Redmon 团队于 2016 年提出了 YOLO 算法，该算法将目标检测简化为回归问题。YOLO 首先将输入图像划分成 S*S 个尺寸相同的网格，每个网格负责预测目标中心点落在自己区域内的对象，接着生成该目标的边界框，完成对检测对象的分类和定位。YOLO 的出现极大地提升了检测速度，但同时也造成了检测精度的下降。SSD 算法借鉴了 YOLO 的回归思想，直接在特征图上生成默认框，对目标进行分类和定位。不同于 YOLO 算法，SSD 不仅在顶层特征图上进行预测，还使用不同分辨率的特征图进行预测，有效提升了检测精度。DSSD [37]（Deconvolutional Single Shot Detector）算法在 SSD 的基础上添加反卷积层来增强语义信息，虽然检测速度略有下降，但提升了模型整体的检测精度。文献[38]提出了 FSSD 模型，通过特征融合的方式，大幅度提升了模型的检测精度。文献[39]提出了 DSOD 模型，并且证明了通过网络微调可以略微提升直接训练检测的模型性能。文献[40]提出的 RRC 算法通过更改 SSD 网络构架，增加输入图片分辨率大小等方式在车辆检测领域取得了不错的效果。在此之后，Redmon 等人相继提出了 YOLO 的升级版本 YOLOv2[41]和 YOLOv3[42]，YOLOv2 相对于 YOLO 具有更高的检测效率，并且能够检测出更多的目标类别，YOLOv3 则提高了对小尺度目标的检测效果。近两年，受 YOLO 系列算法思想的启发，基于 anchor-free 的目标检测方法迅速发展。例如文献[43] 中提出的 CenterNet 模型通过中心点信息来回归出其他边界框的属性信息，文献[44]中提出的 CornerNet 模型则反其道而行，直接使用目标的两个角点（一般是左上角和右下角）来定义边界框。另外，文献[45]

提出的 ExtremeNet 模型，通过检测四条边的极值点以及中心点来确定检测目标。相关的工作还有文献[46-48]提出的 PLN 模型，RepPoints 模型，CSP 模型等。

目标检测发展至今，已经逐步从传统的方法过渡到了深度学习的方法，检测精度和检测速度也在不断提升，但是在某些问题上依旧存在很大提升的空间，例如小目标的检测。小目标由于其相对尺度较小，模型能够提取的特征较少，从而导致检测的效果不佳。针对这类问题，文献[49-50]通过平衡 IOU 阈值和正负样本数量，并使用不同的检测器进行组合的方法来提升对小目标的检测性能。文献[51]则提出了小目标检测的一般方法：提高输入图像的分辨率，多尺度特征表示等，并使用生成式对抗网络（Generative Adversarial Network, GAN）进一步提升小尺度物体的检测准确率。文献[52]提出了 DetNet 模型，通过对基础网络 ResNet 的优化提高了在小目标上的检测效果。

1.3 主要工作及研究内容与贡献

本文拟以电子价签检测算法研究与系统实现为研究内容，属于计算机视觉的目标检测任务。为了将问题具像化，本文利用现有的电子价签数据集作为研究对象进行实验，提升对检测对象的精确度及时效性。

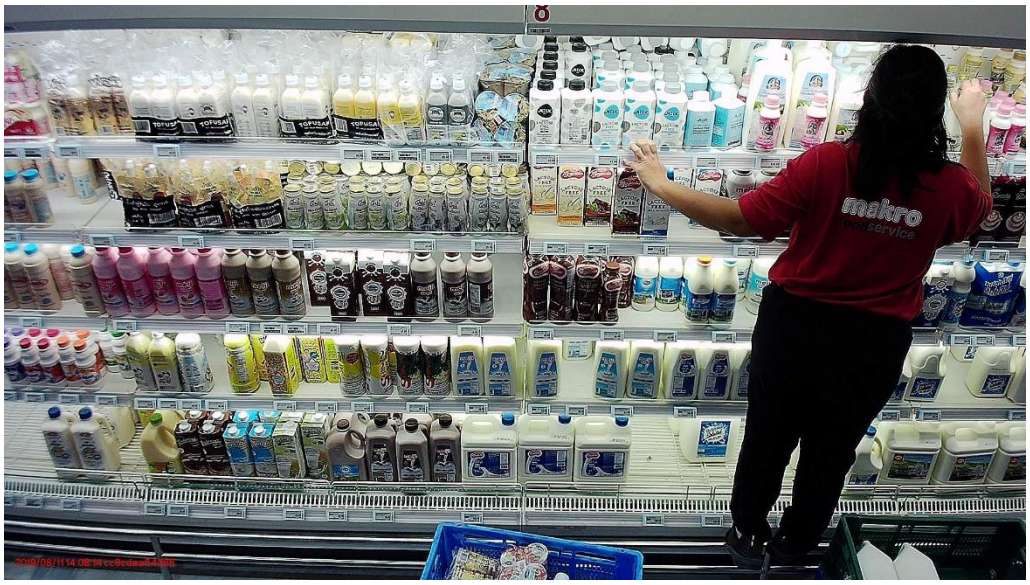


图 1-4 待检测图片
Figure 1-4 Picture to be Detected

本文的研究难点表现在三个方面：

(1) 数据集的制作。本文拟使用深度学习的目标检测方法进行电子价签的检测，需要包含大量电子价签图片的数据集进行模型训练。然而目前网络上并没有合适

的公开数据集，因此本文需要花费大量时间获取、筛选真实图片，并对图片中的检测目标进行详细的标注，以确保训练数据的准确性。

(2) 检测任务复杂。图 1-4 显示的是本文的检测图片，观察图片可以发现，检测对象中的电子价签尺度小、分布密集且数量众多。另外，图片的背景环境复杂、多变，检测模型容易受到光照、人流等因素的干扰。这种小尺度、高密度的目标检测任务是一个挑战。

(3) 检测方法不匹配。目前已有的深度学习目标检测方法并不能有效检测出该类数据集的小尺度、高密度目标。本文需要针对数据集的特点，设计新的检测模型来提升检测的准确性。

针对以上研究难点，本文的主要工作及研究内容如下：

(1) 创立实验环境，选择合适的深度学习平台进行模型框架的搭建。本文的实验环境搭建在 Linux 操作系统上，具有更强的稳定性，安装了 PyTorch 作为深度学习开发工具。本文的所有实验均基于 PyTorch 框架来实现，需要阅读 PyTorch 的文档和源码，了解、学习并使用 PyTorch 来进行开发和实验，编程实现基于 PyTorch 的电子价签检测算法。

(2) 获取、筛选和标注电子价签数据集，并对实验数据进行分析。本文的电子价签数据集属于私有数据集，通过 Navicat Premium、本地编译下载工具等软件获取真实数据集，并根据图片的时间分布、环境特点进行筛选，使得电子价签数据集更加多样化。筛选后的图片通过 labelImg-master 工具进行图片打标，制作成合适的电子价签数据集。最后深入分析现有的电子价签数据集，了解图像样本的特点，并且熟悉检测对象的特征，统计数据集的基础信息，整理成模型可用的电子价签数据集。

(3) 根据需求设计合适的电子价签检测系统，并对各模块的功能进行详细说明，完成对数据及接口的定义。

(4) 图片预处理。使用 OpenCV 等计算机视觉库对输入图片进行预处理，转换成合适的输入向量，并使用数据增强等手段，提升模型整体的鲁棒性。

(5) 融合多特征图进行预测。针对电子价签数据集的特点，本文提出了一种新的特征图融合方法，使得模型更加关注低层特征的信息，有助于提取出小尺度、高密度目标的特征信息，大幅度提升了模型的检测精度。

(6) 引入注意力机制进行预测。本文在特征图融合的基础上，通过残差网络实现了注意力模块，使得模型增加对低层特征图的注意力，更加注重对小尺度目标的特征提取，进一步增加了模型的检测精确率。

在实现电子价签检测系统的过程中，本文做出了以下三点贡献：

(1) 大规模电子价签数据集的采集、筛选以及标注。本文通过本地程序筛选和

获取云端图片，整理成电子价签数据集并进行图片标注。该数据集包含 14713 电子价签图片，检测目标达 12 万以上.

(2) 设计了一种针对小尺度、高密度目标检测的特征图融合方法。之前的 SSD 检测方法直接对各层特征图上进行预测，这种预测方式并不能很好的提取出电子价签的特征。本文根据电子价签数据集的特点提出了一种新的特征融合方式，使得模型整体更加关注底层特征图的语义信息和位置信息的融合，从而提升电子价签检测方法的准确率。在电子价签数据集上，该方法的 mAP 值提高了 8.8%。

(3) 在特征图融合中引入了注意力机制。注意力模块通过残差块堆叠而成，有效避免了梯度消失等问题。down-up 网络中使用降维和上采样等操作模拟残差块的结构，使得特征融合的过程中更加关注较低层的特征图，加强了融合特征中小尺度目标的语义信息，从而进一步提升了电子价签检测方法的准确率。本文最终的电子价签检测方法在特征融合的基础上，mAP 值又提高了 2.8%，总共提高了 11.6%。

1.4 论文组织结构

本文的组织结构如下：

第二章详细介绍了文本实验中需要使用的开发工具及其安装、使用方法。开发工具包括通用编程语言、图像标注工具、图像数据处理工具库，深度学习平台等。同时对图像特征表达方法等知识进行了梳理，并简要介绍了目标检测算法的相关内容，涉及到传统的目标检测方法和基于深度学习的目标检测方法。最后对本章节的内容进行了总结概括。

第三章对本文所设计的电子价签检测系统进行了介绍并对电子价签数据集进行分析和预处理。首先阐述了本文搭建的电子价签检测系统的整体构架，说明了系统设计中各个模块的作用及工作流程，并且定义了各模块的数据及接口。接下来介绍了电子价签数据集的标注内容和特点，并描述了数据集进行预处理和数据增强的方式。最后对本章节的内容进行了总结概括。

第四章对本文设计的电子价签检测算法进行了细致的介绍。本章主要包含特征图的融合，注意力模块的引入，训练方法的设计等工作内容。首先针对电子价签的样本特点介绍了本文的特征融合方法，大幅度提升了模型的检测精度。然后引入了注意力机制，在特征图融合的低层上添加注意力模块，进一步优化了模型的检测性能。接着阐述了本文设计的模型的训练方法，包括损失函数的设计，默认框的选取策略，数据增强方法，训练流程。最后对本章节的内容进行了总结概括。

第五章主要利用选取的评价指标对本文所设计的电子价签检测方法进行实验性能评估。首先对本文所设计的电子价签检测方法的实验参数进行了简单的说明，

选择最合适的参数进行对比实验，同时还展示了本文所设计的电子价签检测方法的训练过程、检测性能、检测结果样图。接着选择最佳的基线模型，并与本文所设计的电子价签检测方法的实验结果进行比较分析。然后分析各模块的作用并对模型性能的提升做出合理的解释。最后对本章节的内容进行了总结概括。

第六章主要对整篇论文进行了全面的概括总结并对下一步的研究方向进行了展望。主要展示了本文所解决的挑战与问题并列举了本文所做的贡献，接下来对本文的下一步研究方向进行分析展望。

2 论文相关知识介绍

本章介绍本文使用的开发工具及所涉及的相关知识，以便于读者对本文所做的工作进行更为细致、全面的了解。首先介绍了本文实验中所使用的所有开发工具，并且详细记录了它们的安装及使用方法，其中包括相应的数据处理工具库、计算机视觉库、深度学习框架、深度学习可视化工具。其次介绍了传统的图像特征和目标检测方法，包括 RGB 特征，HOG 特征等。然后介绍了基于深度学习的目标检测方法，例如基于区域生成的算法 R-CNN 系列等，基于回归方法的算法 YOLO 系列，SDD 系列等。接着介绍了目标检测任务中通用的评价指标。最后是对本章内容的小结。

2.1 开发工具介绍

2.1.1 Python 语言

Python 由 Guido van Rossum 发明，发布于 1991 年，是一种高级编程语言。与 Java、C++、PHP 等编程语言相比，Python 具有较高的解释性，编写的代码简洁干净、易于理解。Python 作为一种面向对象的动态编程语言，更适合初学编程者。另外 Python 有非常强大的第三方库，大大降低了开发成本，有利于实现大规模的编程程序。

如今 Python 在编程语言中已经排名第一，被科研人员和互联网公司大量使用。产生这种结果的原因是 Python 在开发、生产中具有全面的应用，例如图像用户接口，WEB 开发，金融，系统运维，科学运算，云计算等。在这些业界，Python 比其他编程语言具有更高的工业价值。另外 Python 的优点包括广泛的支持库、强大的集成功能、更好的提高程序员的工作效率、强大的生产力等，这些优点也促使 Python 成为人工智能的第一编程语言。

本文的开发内容，包括数据处理与深度学习算法两方面，都是使用 Python 完成的，只需要调取不同的工具包以实现不同的功能。

2.1.2 Anaconda 介绍

Anaconda 是一组数据科学软件包，专门针对 Python 语言所设计，可以通过 conda 包管理系统对软件包管理和环境管理。Anaconda 支持多版本 Python 版本的

并行使用，通常包含 Python2.7 和 Python3.5 两个版本。Anaconda 中包含了 NumPy，Pandas 等众多 Python 库，以及 Spyder、Jupyter Notebook 等集成开发环境，有利于开发人员快速学习 Python 的使用，完成相关工作的环境搭建。而且，如果主机上已经安装过 Python，Anaconda 的安装并不会破坏原有的环境。

Anaconda 可以通过“conda install”指令快速安装 Python 开发过程中需要使用的相关库，并通过“conda list”指令来查看已经安装成功的计算库。当 Anaconda 中的库过期导致程序报错时，可以在终端输入“conda upgrade --all”指令更新所有库，或者通过“conda upgrade --库名”来更新指定的库。

2.1.3 NumPy 数组运算库

NumPy 是使用 Python 进行科学计算的基础包。NumPy 里包含多维数组对象和各种派生对象。NumPy 可以实现数组的快速运算，例如基本线性代数，基本统计运算，随机模拟等。

在需要安装以及使用 NumPy 时，操作者只需要在终端执行：

“pip install numpy”

等待安装完成之后，在需要使用 NumPy 的 Python 文件中，操作者只需要在文件头部输入一行：

“import numpy as np”

即可以在代码中利用“np+.+功能”来使用 NumPy 中的功能。

2.1.4 Matplotlib 绘图库

Matplotlib 是 Python 的绘图库。Matplotlib 可以和 NumPy 组合使用，提供了一套和 MATLAB 相似的命令接口，具有多种绘图方案，十分适合交互式绘图。它也可以和图形工具包一起使用，如 PyQt 和 wxPython。

在需要安装以及使用 Matplotlib 时，操作者只需要在终端执行：

“python -m pip install matplotlib”

等待安装完成之后，在需要使用 Matplotlib 的 Python 文件中，操作者只需要在文件头部输入一行：

“from matplotlib import pyplot as plt”

即可以在代码中利用“plt+.+功能”来使用 Matplotlib 中的功能。

2.1.5 OpenCV 库

OpenCV 于 1999 年由 Gary Bradsky 创立，现如今广泛应用于计算机视觉和机器学习领域。OpenCV 支持各种主流的系统平台，例如 Windows、Linux、OS X、Android 和 iOS。另外 OpenCV 也支持 Python、C++、Java 等不同的编程语言。OpenCV 是一种轻量级的计算机视觉库，包含许多通用的计算机视觉方面的算法。在计算机视觉项目的开发中，OpenCV 拥有了丰富的常用图像处理函数库，提供了多种编程语言的外部接口，能够快速的实现一些图像处理和识别的任务。

在需要安装以及使用 OpenCV 时，操作者只需要在终端执行：

“pip install opencv-python ”

等待安装完成之后，在需要使用 OpenCV 的 Python 文件中，操作者只需要在文件头部输入一行：

“import cv2”

即可以在代码中利用“cv2+.+功能”来使用 OpenCV 中的功能。

2.1.6 PyTorch 深度学习框架

PyTorch 是由 Facebook 人工智能研究小组开发的神经网络框架，是基于 Torch 框架的 Python 包装器，被广泛应用于计算机视觉，自然语言处理，推荐等多个人工智能领域。PyTorch 提供简单、易于使用的接口，不仅代码简介，而且操作简单。PyTorch 主要面向 Python 编程语言，因此可以利用 Python 环境提供的所有功能和服务。不同于 TensorFlow 的静态计算图，PyTorch 提供了一个出色的动态计算图平台，可以根据设计需求实时地改变变量，大大提升了训练网络的效率。

文本中实验使用 NVIDIA GTX1080 显卡和 Ubuntu 16.04 操作系统，需要根据软硬件环境搭建相关的 GPU 计算平台。首先需要先安装 CUDA 和 cuDNN，CUDA 是一种 NVIDIA 推出的并行计算架构，能够使用 GPU 解决大量、复杂的计算问题，cuDNN 是用于深度神经网络的 GPU 加速库，具有高性能、易使用、低内存消耗等优点。在终端输入“sudo apt-get install nvidia-367”指令安装显卡驱动，接着根据软硬件环境安装 9.0 版本的 CUDA，在终端输入以下指令进行安装：

“sudo sh cuda_9.0.176_384.81_linux.run”

结束后需要设置 CUDA 的环境变量，在 home 目录中的.bashrc 文件里添加以下三行指令：

“export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/cuda-9.0/lib64 ”

“export PATH=$PATH:/usr/local/cuda-9.0/bin”

“export CUDA_HOME=$CUDA_HOME:/usr/local/cuda-9.0”

完成添加后在终端输入“source ~/.bashrc”指令即可完成 CUDA 环境的添加。

可以在终端输入“nvcc --version”来查看 CUDA 是否安装成功。安装 cuDNN 需要在 NVIDIA 官网上下载安装文件，解压后在当前目录中打开终端，并执行以下指令：

“sudo cp cuda/include/cudnn.h /usr/local/cuda/include/”

“sudo cp cuda/lib64/libcudnn* /usr/local/cuda/lib64/”

“sudo chmod a+r /usr/local/cuda/include/cudnn.h”

“sudo chmod a+r /usr/local/cuda/lib64/libcudnn*”

安装结束后在终端输入以下指令查看是否安装成功：

“cat /usr/local/cuda/include/cudnn.h | grep CUDNN_MAJOR -A 2”

到此相关的 GPU 架构和加速库已经安装成功。接着需要搭建用于实验的深度学习框架 Pytorch。根据 CUDA 和 cuDNN 版本，在终端命令行输入：

“conda install pytorch torchvision cudatoolkit=9.0 -c pytorch”

即可安装成功，在相关 Python 代码中输入：

“import torch”

即可使用 PyToch 深度学习框架来搭建自己的算法模型。

2.1.7 Torchvision 库

Torchvision 不包含于 PyTorch 深度学习框架，是一个独立的图像工具库，拥有丰富的数据集和深度学习模型，并且提供常用的图像操作函数。Torchvision 库主要包括以下几个包：

(1) datasets 包里拥有许多公开视觉数据集，用户可以直接下载到本地，并在程序中加载需要使用的数据集。另外用户可以通过 datasets 的子类来创建自己的数据集格式，效率极高。

(2) models 包里有主流的计算机视觉模型，例如 AlexNet、VGG、ResNet 和 Densenet 以及训练好的网络参数。

(3) transforms 包里有通用的图像操作函数，例如对图像的缩放、旋转、切割等。另外 transform 还支持图像数据、tensor 张量、numpy 数组之间的相互转换。

(4) utils 包的功能是将 tensor 张量存储到计算机硬盘里，并且输出一个批次的图像可以生成一个图像格网。

在需要安装以及使用 Torchvision 时，操作者只需要在终端执行：

“conda install torchvision -c pytorch”

等待安装完成之后，在需要使用 Torchvision 的 Python 文件中，操作者只需要在文件头部输入一行：

“import torchvision”

即可以在代码中使用 Torchvision 库中的功能。

2.1.8 TensorboardX 可视化工具

Google Tensorflow 的附加工具 Tensorboard 是一个优秀的视觉化工具。Tensorboard 可以记录数字、影像或者是音频信息，对于观察神经网络的训练过程十分有帮助。然而 Tensorboard 只能应用于 TensorFlow 平台，并不能跨平台使用。TensorboardX 的目的就是让 tensorboard 的功能可以被其他深度学习框架使用。Tensorboard 为科研人员提供了简洁的可视化网络结构，详细的网络参数变化过程，极大地提升了科研效率。

图 2-1 TensorboardX 可视化界面

Figure 2-1 Visual Interface of TensorboardX

在需要安装以及使用 TensorboardX 时，操作者只需要在终端执行：

“pip install tensorboardX”

等待安装完成之后，需要通过以下几个步骤来实现 TensorboardX 的可视化：

(1) 首先需要导入 tensorboardX。在 python 文件中输入：

“from tensorboardX import SummaryWriter”

“writer = SummaryWriter('log')”

(2) 对保存的模型进行可视化。在命令行输入：

“ tensorboard --logdir= 路径”

(3) 最后在第三方浏览器中打开命令行生成的地址，即可看到模型的结构，图 2-1 展示的就是 TensorboardX 的可视化界面。

2.1.9 visdom 可视化工具

visdom 是 FaceBook 开发的一款可视化工具，其实质是一款在网页端的 web 服务器，对深度学习框架 PyTorch 的支持较好，支持 python 语言。visdom 可以通过编码程序进行调用，在浏览器等第三方界面上做可视化展示，创建实时的网络训练过程图，用于调试、优化程序代码和对比实验结果。

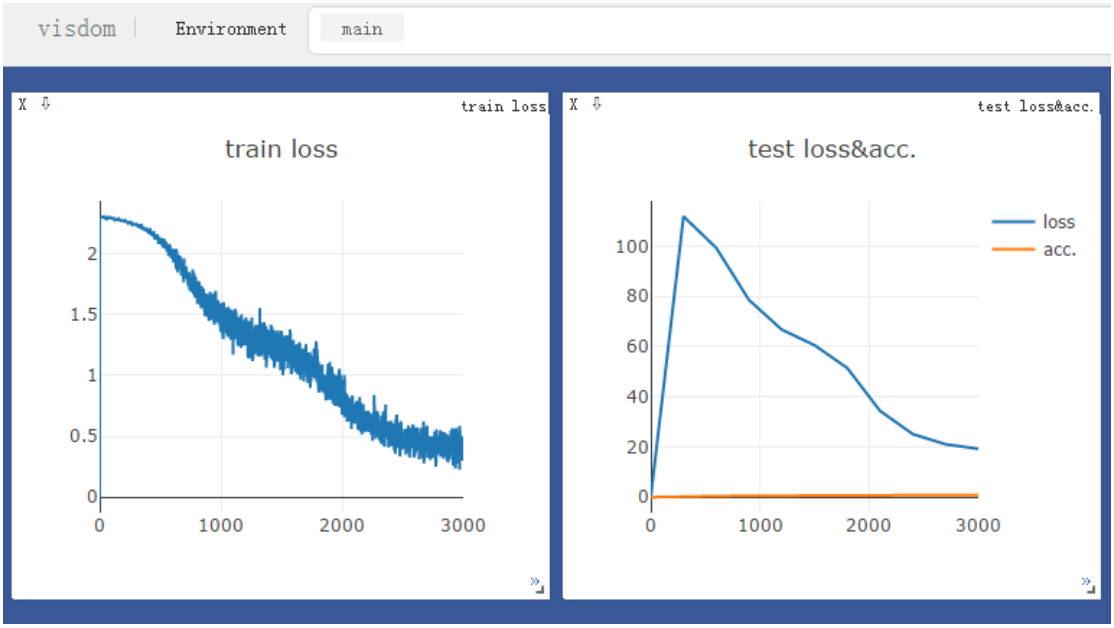


图 2-2 visdom 可视化界面

Figure 2-2 Visual Interface of visdom

在需要安装以及使用 visdom 时，操作者只需要在终端执行：

“pip install visdom”

等待安装完成之后，需要通过以下几个步骤来打开 visdom 的可视化界面：

(1) 首先需要激活 visdom。在终端输入以下指令：

“python -m visdom.server”

(2) 在 python 程序中创建 visdom 环境。在 python 文件中输入：

“from visdom import visdom”

“vis = visdom(env=”设置自己的环境名”)”

(3) 在浏览器界面查看训练过程。在浏览器中输入：

“http://localhost:8888”

如图 2-2 所示，就可以查看 visdom 绘制的训练过程图了。

2.2 传统图像特征和检测方法

在深度学习应用于目标检测之前，使用传统的图像特征来检测目标是计算机视觉领域的核心方法。传统的图像特征多以颜色或梯度为基础特征，并在此基础上发展出了边缘检测、角点检测等方法。本小节将会介绍一些传统的图像特征，包括颜色特征、梯度特征，还有传统目标检测方法的检测流程。

2.2.1 RGB 特征

目前工业界所使用的颜色标准大多为 RGB 色彩模式，也就是通过红绿蓝三种颜色通道的变化和叠加获得各种其他颜色，人类视觉分辨出的颜色大部分都能在 RGB 色彩模式中找到。在使用中，有时为了计算或存储，需要将彩色图转换为灰度图，在转换时大多采用心理学公式进行计算，公式如下所示：

$$\text{Gray} = 0.299 \times R + 0.587 \times G + 0.114 \times B \tag{2-1}$$

其中 Gray 表示灰度值，R 、G 、B 分别表示红绿蓝三种颜色的像素值。

颜色直方图是颜色特征最基本的表示方法，能够反映同一通道中像素值的统计分布，公式如下所示：

$$H(k) = \frac{n_k}{N} \tag{2-2}$$

其中 $H(k)$ 表示像素值 k 出现的概率，n_k 表示像素值 k 出现的频率，N 是图像中像素的总数。颜色直方图作为图像的全局特征，只是描述图像中各像素的分布情况，并不能描述图像中的物体特征。其优点是当图像发生平移和缩放等几何变换，或者图像质量改变时，颜色直方图不会产生很大变化。

2.2.2 方向梯度直方图

方向梯度直方图特征被广泛用于计算机视觉和图像处理，是一种目标检测的特征描述算子，HOG 特征通过局部区域的梯度方向直方图提取检测目标的特征。梯度大部分存在于图像中的边缘部分，利用梯度或边缘的方向密度分布能够较好

地描述局部目标的特点和形状。

计算图像的 HOG 特征，首先需要对图像进行预处理，将图像转换为灰度图，并进行颜色空间的归一化，这样能够起到调节图像对比度的作用，很大程度上减少了图像中光照变化所产生的影响，而且也降低了噪音干扰。接着便是计算每个像素点的梯度，像素 (x, y) 的梯度计算如下：

$$G_x(x, y) = H(x+1, y) - H(x-1, y) \tag{2-3}$$

$$G_y(x, y) = H(x, y+1) - H(x, y-1) \tag{2-4}$$

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \tag{2-5}$$

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_y(x, y)}{G_x(x, y)}\right) \tag{2-6}$$

其中 $G_x(x, y)$ 表示该像素点在 (x, y) 处的水平方向的梯度，$G_y(x, y)$ 表示该像素点在 (x, y) 处垂直方向的梯度，$H(x, y)$ 表示该点的灰度值。$G(x, y)$ 表示梯度的幅值，$\alpha(x, y)$ 表示梯度的方向。使用中最常用的方法是用 $[-1, 0, 1]$ 梯度算子卷积原图像，得到 x 方向的梯度分量，是用 $[1, 0, -1]$ 梯度算子卷积原图像，得到 y 方向的梯度分量，然后通过式(2-5)，式(2-6)得到梯度的大小和方向。计算 HOG 特征的第三步是为图像划分单元并构建梯度方向直方图，最后将每一个细胞内的梯度方向直方图统计特征合并，得到一张图像的完整 HOG 特征。

2.2.3 传统目标检测方法

用传统的方法对一张图像进行目标检测可以概括为三步：首先对待检测的图像进行分割，分割的作用是将整张图像变为许多候选区域，然后提取这些候选区域中可用于识别目标对象的特征，最后将获得的特征输入到分类器中，分类器会根据这些特征判断候选区域所属的类别。下面对这三个部分进行详细介绍：

(1) 区域选择。在检测开始前，检测系统并没有目标在图片中的位置信息，也不知道目标的尺寸和长宽比。因此检测系统需要通过枚举目标可能存在的区域来确定检测对象的位置信息。最直接的方法就是使用滑动窗口在整个图片上以不同尺寸、长宽比来进行遍历，这种方法虽然大概率可以获得和目标相关的区域，但会产生大量的冗余区域，计算成本过高，并造成特征提取和分类的效率过低。现实中滑动窗口的尺寸和长宽比一般设置为定值，因此如果图片中存在多个检测目标，且尺度变化较大，那么滑动窗口的检测准确性将特别低。

(2) 特征提取。提取出相关的区域后，需要设计一个特征来表达检测目标。设

计的特征要尽可能的包含检测对象的信息，适应不同的检测环境。特征质量的高低直接影响着分类结果的准确性，提取出的特征质量越高，则分类正确的可能性就越高。具有边缘信息的 HOG 特征和具有纹理信息的 LBP 特征在实验中使用较多。

(3) 分类器。根据特征选出的区域需要通过分类器来判断是否是检测目标，如果该区域被判定为是检测目标，则需要进一步通过分类器来判断它的所属类别。SVM 与集成学习在处理图像的分类问题时效率较高，所以这两种分离器在传统目标检测中经常被使用。

比较具有代表性的传统目标方法有 HOG 特征结合 SVM 分类器、可变组件模型算法（Deformable Parts Models，DPM）等。

HOG 特征结合 SVM 分类器的检测方法主要包括六个步骤:颜色空间归一化，计算梯度，组合相邻的细胞单元（cell）成块（block），对每个块内的梯度直方图进行归一化，收集所有块的 HOG 特征，SVM 分类器分类。首先需要归一化输入图像的颜色空间，这个操作是为了减少光照、背景等因素的影响，接着需要将检测窗口划分成尺度、长宽比相同的细胞单元，并获取所有细胞单元的梯度信息；将相邻的细胞单元拼接成块，显然块之间有重叠的部分，因此可以利用块之间重叠的信息，绘制所有块的梯度直方图；然后归一化每个块的梯度直方图，有利于减少各种噪声、背景等因素的影响；最后整理所有块的 HOG 特征信息，并通过特征向量来直观表达。在 HOG 特征的获取过程中，归一化因子的大小、细胞单元的大小、块的重叠大小、梯度方向的设置等因素都会影响网络的检测结果。最终使用线性支持向量机作为分类判别器，输入事先挑选的正样本、负样本以及误测样本三部分组成的样本集合，重新学习得到最终的分类判别器，学习得到感兴趣的目标特征。滑动窗口逐步获取检测图像的预选区域，通过 SVM 分类器对窗口的内容进行类别分类，并对分类结果进行修正，获得检测的目标区域，完成对检测对象的定位。

DPM 方法则在 HOG 特征的基础上进一步对模型进行了优化。DPM 算法的模型主要由一个 Root 滤波器和 Part 滤波器组成。DPM 算法需要先通过 HOG 获得特征，但是不同于原来的 HOG 算法，DPM 算法只采用了 HOG 中的 cell 结构，在某种程度上实现了对网络结构的降维。在此基础上提出了输入图像的 DPM 特征图，并对输入图像进行高斯金字塔上采样，再通过相同的操作获取其 DPM 特征图，这样就获得了两张分辨率不同的特征图。接着在输入图像获取的 DPM 特征图和已经训练完的 Root 滤波器上做卷积操作，获得 Root 滤波器的响应图。使用相同的操作，在上采样生成的 DPM 特征图和已经训练完的 Part 滤波器上做卷积操作，获得 Part 滤波器的响应图。为了生成最终的响应图，需要对 Part 滤波器的响应图进行下采样，使得 Part 滤波器响应图的分辨率和 Root 滤波器响应图的分辨率相同。加权平均获得的响应图中，响应值越大则该区域的亮度越大，成正比关系。

2.3 基于深度学习的目标检测方法

深度学习发展至今，在目标检测领域衍生出了两个主流方向：一是基于区域生成的深度学习目标检测算法；二是基于回归方法的深度学习目标检测算法。本小节将简要介绍一下目前主流的基于深度学习的目标检测方法。

2.3.1 R-CNN 目标检测方法

R-CNN 系列是基于区域生成的深度学习目标检测方法中最具代表性的检测方法，从 2013 年被发明出来就不断优化和改进，模型检测的精度和速度都在不断提升，现已成为工业界目标检测的标杆。

2013 年，Girshick 团队使用深度卷积神经网络提取检测目标的特征，取代了传统目标检测中手工设计特征的方法，并提出了新的检测框架 R-CNN。

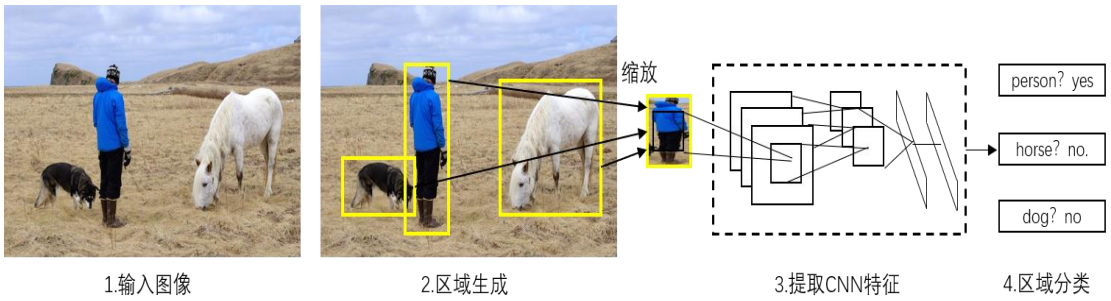


图 2-3 R-CNN 网络结构
Figure 2-3 Network of R-CNN

R-CNN 目标检测框架如图 2-3 所示，主要包含如下四个部分：

(1) 输入原始检测图像。

(2) 在输入图像上进行区域生成。通过 Selective Search 区域生成算法在图像上产生约两千个兴趣区域。

(3) 缩放兴趣区域的尺寸。为了确保卷积神经网络的全连接层能够输入相同维度的向量，需要将每个兴趣区域缩放到相同大小的尺度。

(4) 分类和边框回归。将相同尺度的兴趣区域输入到卷积神经网络中，获取该区域的特征，之后使用 SVM 分类器对获得的特征进行分类，同时对目标的兴趣区域进行边框回归，获得图像中所有目标的边界框。

相较于传统的目标检测方法，R-CNN 在公开数据集上具有爆炸性的表现。在著名挑战赛 PASCAL VOC 上，R-CNN 将 mAP 值提升了 31.7%，震惊了学术界，卷积神经网络提取图片特征的巨大优势开始现象。但 R-CNN 检测方法也存在许多

不足之处，最明显的就是 R-CNN 的检测速度太慢。R-CNN 在区域生成的阶段，需要通过 Selective Search 方法生成大量的兴趣区域，每个兴趣区域再通过卷积神经网络提取特征，最终进入 SVM 分类器进行分类。可以发现，R-CNN 需要对每一张输入图片进行两千次卷积神经网络提取特征和分类操作，卷积神经网络的训练过程本身就十分耗时，因此 R-CNN 的重复工作浪费了大量的计算资源。

2.3.2 SPPnets 目标检测方法

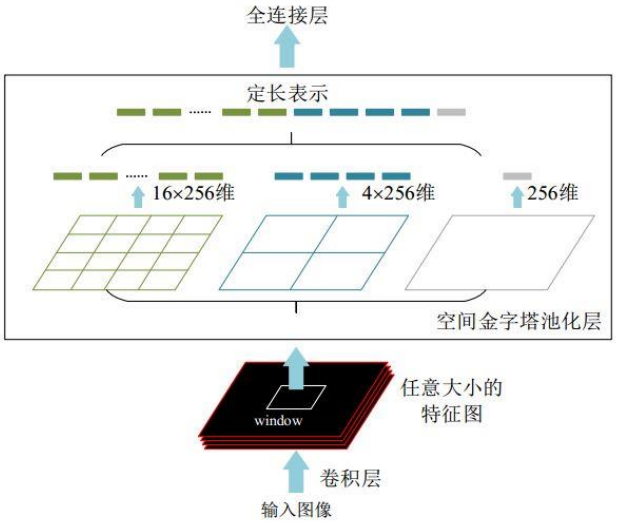


图 2-4 SPPnets 网络
Figure 2-4 Network of SPPnets

SPPnets 在 R-CNN 的基础上进行了优化。SPPnets 模型如图 2-4 所示，不同于 R-CNN，SPPnets 可以输入任意大小的检测图片，再通过卷积等操作获得特征。输入图像中的兴趣区域被一一映射到高维度特征图中，对应图中的 window 窗口。这一操作大大减少了计算成本，因为网络只用对输入图像进行卷积操作以获取特征图，不再需要对使用卷积对每个兴趣区域提取特征。空间金字塔池化层将兴趣区域分为 1*1，2*2，4*4 三个不同尺度的块，再对每个块使用池化操作，生成固定长度的特征向量，这使得 SPPnets 网络可以输入任意大小的图片。SPPnets 通过空间金字塔池化层获得全连接层的输入，并对最后固定长度的特征向量进行分类和边框回归，获得分类结果和目标定位。

虽然 SPPnets 提升了模型的检测性能，但依然存在许多问题，例如兴趣区域重叠问题导致计算资源的浪费，网络训练过程复杂导致模型性能下降等。

2.3.3 Fast R-CNN 目标检测方法

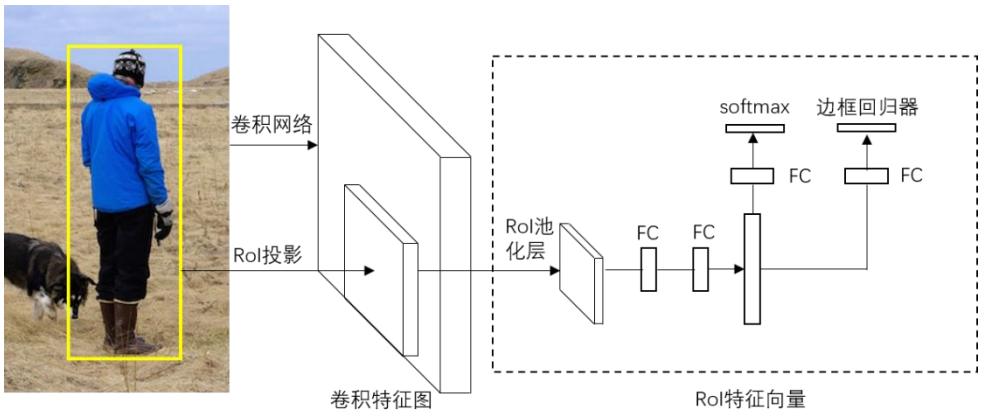


图 2-5 Fast R-CNN 网络
Figure 2-5 Network of Fast R-CNN

如图 2-5 所示，受 SPPnets 的启发，Girshick 团队又提出了一个 R-CNN 的简化升级版——Fast R-CNN。与 SPPnets 相似，Fast R-CNN 输入图像的每个兴趣区域被一一映射到特征图中。Fast R-CNN 去除了 R-CNN 后面一部分网络，并添加了 RoI 池化层，RoI 池化层是对空间金字塔池化层的优化。另外，Fast R-CNN 模型使用 softmax 分类器替换了多个 SVM 分类器，并且将定位损失函数加入到总损失函数中，这样网络训练就包含了目标边框回归的过程。Fast R-CNN 整合了网络训练的过程，大大减少了网络模型训练的时间成本。

2.3.4 Faster R-CNN 目标检测方法

Faster R-CNN 利用候选区域网络生成候选框，代替了传统的选择性搜索等方法，大大减少了区域生成的计算成本。

图 2-6 展示的是 Faster R-CNN 的原理图，主要包含以下三个部分：

(1) 特征提取。输入原始图片，并通过卷积神经网络获得图像特征

(2) 区域生成。在卷积神经网络最后一层卷积特征图的每个像素点上放置 k 个尺度、长宽比不同的默认框，生成候选区域，其中 k 一般取值为 9。

(3) 分类与回归。对每个默认框生成的区域进行初步的二分类，判断该区域是否包含检测目标（也可以理解为是否分类成背景），接着对候选框微调，并使用非极大值抑制方法筛选出合适数量的样本，控制正负样本的比例，最后对检测目标进行精确分类，确定非背景目标的所属类别。

Faster R-CNN 不再使用传统的区域生成方法，引入了简洁、高效的候选区域网络，使得模型网络共同使用卷积特征，降低了检测模型的时间复杂度。

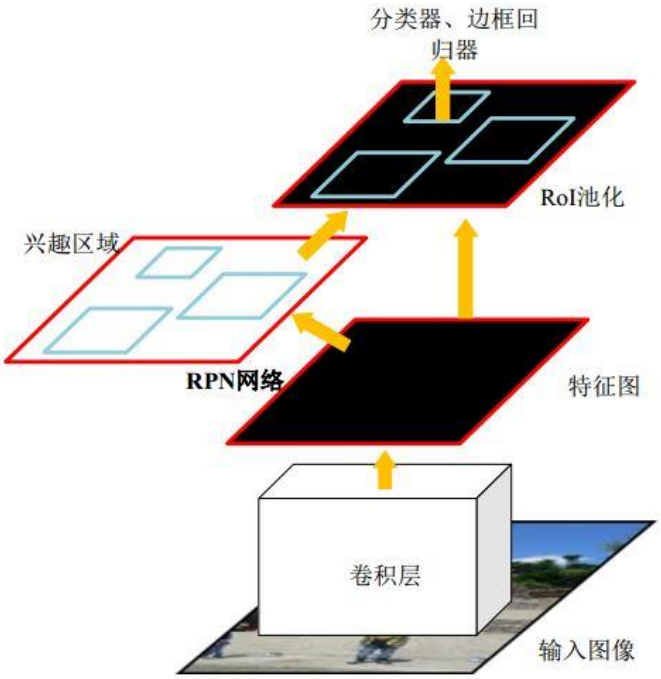


图 2-6 Faster R-CNN 网络
Figure 2-6 Network of Faster R-CNN

2.3.5 YOLO 目标检测方法

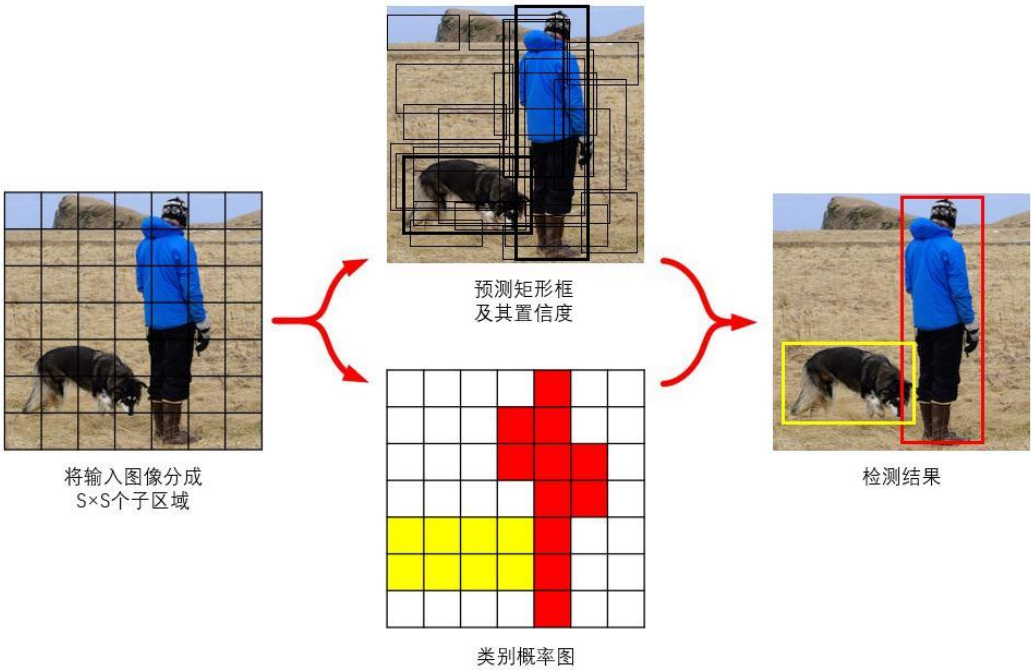


图 2-7 YOLO 检测过程
Figure 2-7 Detection Process of YOLO

基于区域生成的目标检测方法虽然有较高的检测精度，但检测速度较差，所以很难在许多对实时性要求较高的场合上使用。YOLO、SSD 等使用回归方法的目标检测方法，使用端到端的网络架构，检测速度获得了大幅度的提升。

图 2-7 可以简单描述 YOLO 的检测原理。YOLO 将固定分辨率的输入图像分割成 S*S 个网格（S 一般设置为 7），每个网格负责预测目标中心点落在自己区域内的对象，并预测 B 个边框的坐标（B 一般取值为 2），以及该对象分类的概率分布，最后通过非极大值抑制筛除不相关的边框，并获得预测结果。

YOLO 使用回归的方法极大地简化了目标检测的网络架构，使得深度学习的目标检测方法第一次满足实时性的需求。但 YOLO 简化网络的同时也带来了许多弊端，例如只在 S*S 网格上进行目标回归无法准确地定位目标，有时甚至会造成生成的边框无法包含检测目标的现象。

2.3.6 SSD 目标检测方法

SSD 不同于 YOLO，直接使用卷积神经网络获取边界框，并利用多个不同分辨率的特征图进行检测。另外 SSD 借鉴了 Faster R-CNN 的思想，采用不同尺度和长宽比的先验框（prior boxes）来获取目标区域，因此 SSD 的检测效果较好。

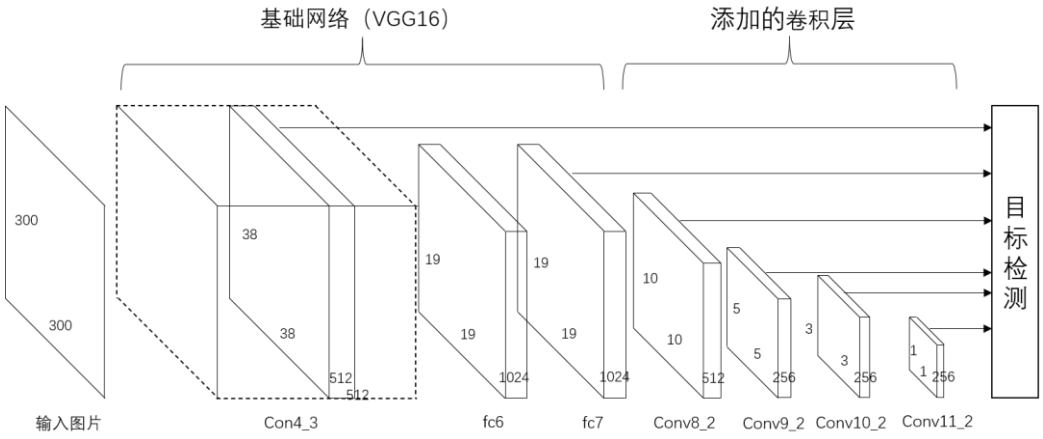


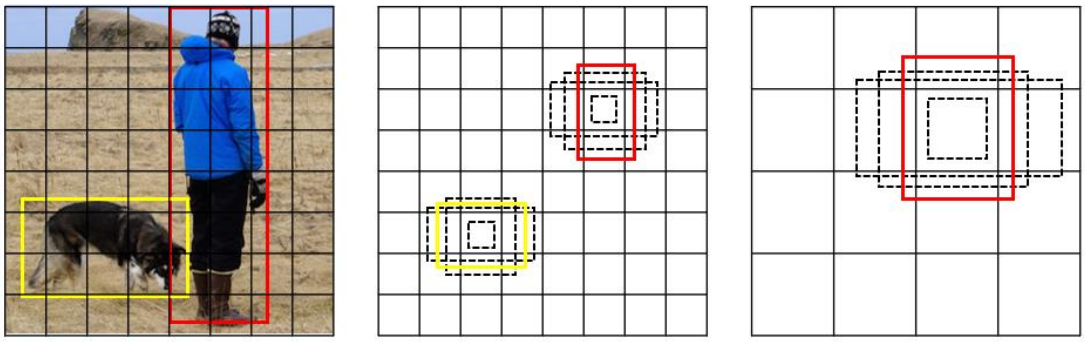
图 2-8 SSD 网络
Figure 2-8 Network of SSD

图 2-8 展示的是 SSD 的网络结构。SSD 前面是基础网络（base model）部分，可以选择 AlexNet、ResNet、Densenet、Shuffnet 等多种不同的网络结构。SSD 在基础网络后，又增加新的卷积特征图，这些特征图分辨率逐渐下降，因此具有不同的感受野。高分辨率的特征图可以用来检测小尺度目标，低分辨率的特征图可以用来检测大尺度目标，所以 SSD 能够检测多尺度的目标，提升了 YOLO 在不同尺度目

标上的检测精度。这种检测方法有两方面的优势：

(1) SSD 的多尺度检测是在基础网络上生成新的卷积特征图，不需要先生成目标区域，计算成本较低，因此相较于需要先生成区域的 Faster R-CNN 等方法，具有更快的检测速度。

(2) 针对不同的检测群体，SSD 可以调节预测特征图的组成方式。



a) 含真实框的图片 b) 高分辨率特征图 c) 低分辨率特征图

a) Image with Ground Truth Boxes b) High Resolution Feature Map c) Low Resolution Feature Map

图 2-9 SSD 原理图

Figure 2-9 Network of SSD

图 2-9 展示了 SSD 的多尺度检测原理，图 2-9 a)的输入图片中标注了狗的黄色真实框和人的红色真实框，图 2-9 b)和图 2-9 c)显示了不同分辨率的卷积特征图，显然图 2-9 b)中高分辨率的特征图更有助于检测狗等小尺度目标，图 2-9 c)中低分辨率的特征图更有助于检测人等大尺度目标。SSD 在所有预测特征图上都生成了多个默认框，默认框的尺度根据感受野的大小、尺度因子的大小等因素进行调节，主要用于确认检测目标的位置。在模型训练的过程中，真实框会和一个默认框匹配成功，获得最终的检测结果。

2.4 评价指标

不同的评价指标能展示模型不同方面的性能，目标检测方法通常需要比较准确性和实时性两方面的性能。衡量准确性的指标有精确率（Precision），召回率（Recall），P-R 曲线，平均精确率（Average Precision, AP），平均精确率均值（mean Average Precision, mAP）等。帧速率（Frames Per Second, FPS）通常是衡量实时性的指标。

2.4.1 精确率和召回率

表 2-1 混淆矩阵
Table 2-1 Confusion Matrix

	正样本	负样本
预测为正样本	True Positives	False Positives
预测为负样本	False Negatives	True Negatives

目标分类的混淆矩阵如表 2-1 所示。对混淆矩阵进行如下定义：

(1) 真阳性（True Positives）：正样本被正确识别为正样本。

(2) 真阴性（Ture Negatives）：负样本被正确识别为负样本。

(3) 假阳性（False Positives）：负样本被错误识别为正样本。

(4) 假阴性（False Negatives）：正样本被错误识别为负样本。

目标检测任务中精确率即正确检测出的目标所占所有被识别出的目标的比例。精确率的公式如下所示：

$$Prcision_i = \frac{tp_i^N}{N} = \frac{tp_i^N}{tp_i^N + fp_i^N} \tag{2-7}$$

召回率则表示所有识别为目标的对象中，正确检测出的目标所占的比例。召回率的公式如下所示：

$$Recall_i = \frac{tp_i^N}{tp_i^N + fn_i^N} \tag{2-8}$$

其中 i 表示检测的类别，N 表示置信度从大到小排序后的序号，利用前 N 个检测框计算模型的精确率和召回率。提前设定 IoU（Intersection over Union）取值，将前 N 个检测框中和真实框重叠度大于 IoU 的标记为真阳性，数量记为 tp_i^N。重合度低于 IoU 的则标记为假阳性，数量记为 fn_i^N。

改变阈值的大小，并保持 N 的取值不变，则会导致准确率和召回率的数值发生变化。将准确率设置成纵坐标，召回率设置成横坐标，即可得到 P-R（Precision-Recall）曲线。当召回率增长的同时，准确率也保持在一个较高的水平时，说明目标检测系统的性能较好。

2.4.2 F1 分数

F1 分数是一个加权平均数，融合了检测模型的精确率和召回率，可以作为衡量模型准确性的评价指标，计算公式如式 2-9 所示：

$$F_1 = 2 * \frac{Precision * Reacll}{Precision + Recall} \tag{2-9}$$

2.4.3 平均精确率和平均精确率均数

目标检测方法中除了使用准确率和召回率两个指标来表现模型的性能，通常还会使用平均精确率来衡量模型的性能。目前的平均精确率计算方式以 PASCAL VOC2007 定义的计算方式为准：在 P-R 曲线的横坐标（召回率）上取出 11 个点[0，0.1，0.2，•••，1]，然后计算在 11 个点中精确率的最大值是多少。最终可以得到 11 个精确率值，对这 11 个精确率值取平均数，就可以得到平均精确率了。
目标检测任务中，通常会存在多类别检测目标，因此需要通过平均精确率均值来衡量检测系统对所有类别物体的检测性能，计算的方式就是将所有类别物体的平均准确率再取平均值。

2.4.4 帧速率

帧速率是指目标检测模型在单位时间内检测出的图像数量，通常写作每秒检测帧数，是检测速度的一种表达形式，而检测速度又是判断检测系统是否满足实时性的重要指标。

2.5 本章小结

本章对本文使用的实验平台及相关知识进行了介绍，描述了实验技术平台以及相应的模块搭建。首先介绍了 Python 以及与数据分析，图片可视化相关的扩展包：Numpy、Matplotlib、OpenCV，接下来介绍了深度学习框架 PyTorch、Torchvision 的原理、安装及其应用，还介绍了记录训练过程的可视化工具 TensorboardX 和 visdom。其次介绍了传统的图像特征及目标检测方法。最后介绍了主流的基于深度学习的目标检测方法，主要包含 R-CNN 系列算法，YOLO 系列算法，SSD 系列算法，并对它们的评价指标进行了详细的说明。本章对全文所涉及的概念、原理及框架进行了系统的梳理，为对本文的理解提供了基础知识的支持。

3 系统设计与数据分析处理

本章对本文所提出的目标检测系统的整体框架进行具体论述，同时对数据集的制作和图片预处理进行说明。首先说明本文的电子价签检测系统的设计需求。接着对电子价签检测系统的工作模块进行说明，并根据需求及工作模块设计了相应的处理流程，并且补充说明了各模块数据的定义，相互串联的接口方式。接着对本文使用的电子价签数据集进行统计与分析，并对数据集图片进行了一些预处理和数据增强。最好对本章内容进行总结。接下来，本章将从系统设计，数据集标注与分析两个方面进行具体说明。

3.1 设计需求

本文拟设计的电子价签检测系统需要满足以下五方面的需求：

(1) 系统中有一个模块能够统一管理线下的相机群，可以远程控制相机，实现相机的开启/关闭，定时拍摄，修改分辨率，上传图像等功能。

(2) 系统中有一个模块能够接收到相机拍摄的图像，并进行存储和管理。考虑到采集的图片数据量较大，存储模块不能放置在本地，且需要定期对图片数据进行清洗和删除。

(3) 系统中有一个模块能够统一管理所有本地的算法，完成对拍摄图像中电子价签目标的检测，并在此基础上实现对商品的定位等功能。

(4) 系统中有一个模块能够起到中心控制的作用，对其余模块进行控制、调配，并对接外部的其他系统。

电子价签检测系统需要具有实时性，因此电子价签检测算法也需要具有实时性。

3.2 系统设计

根据上一节的设计需求，本文设计了如图 3-1 所示的电子价签检测系统，包括线下商场的相机群、相机路由、相机管理器，智能控制模块 AMP，云上图片文件存储系统，本地算法模块，以及从外部接入的 WISE 系统和 SHOPEWEB 系统。

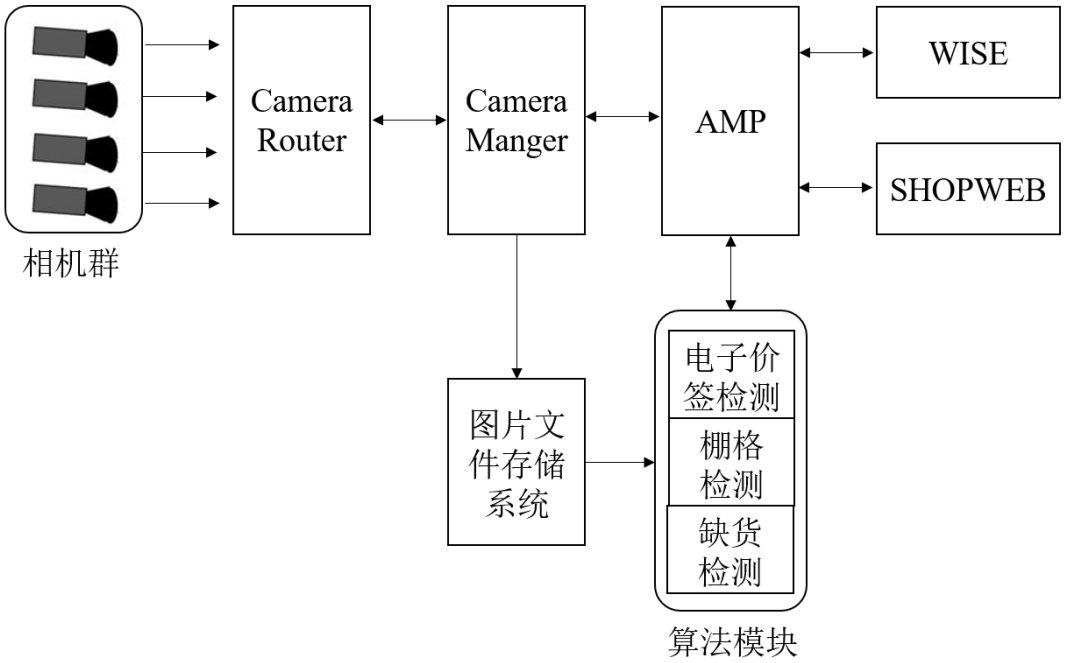


图 3-1 电子价签检测系统
Figure 3-1 Detection System of Electronic Shelf Label

3.2.1 工作模块

如图 3-1 所示，整个电子价签检测系统分为以下六个部分：

(1) 相机群、相机路由（Camera Router）、相机管理器（Camera Manger）。相机群通过相机路由接入系统，相机管理器负责遍历所有相机信息，上报相机信息到 AMP 模块。相机管理器支持对相机的常规操作，包括开启/关闭相机，相机参数控制，获取相机图像。

(2) AMP(AI Management Platform)模块。AMP 模块负责路由相机图像存储信息，向棚格检测模块和电子价签检测模块发送服务请求获取某一相机对应货架的棚格和价签的预测信息，向缺货检测模块发送服务请求获取某一相机对应货架的缺货检测信息。AMP 模块支持与外部 WISE 系统的陈列和缺货信息交换，也支持与外部的 SHOPEWEB 系统的价签信息交换。

(3) WISE 系统。WISE 系统主要是用来存放商场的所有商品信息的，例如某类商品的陈列信息、缺货信息等

(4) SHOPEWEB 系统。SHOPEWEB 系统主要是用来存放商品的价签信息的。

(5) 图片文件存储系统。图片文件存储系统负责相机群组生成的图片管理，能够支持基于时间，基于相机 ID 的图片查询。

(6) 算法模块。算法模块包含电子价签检测、棚格检测、缺货检测。电子价签检测模块负责接收对指定的货架图片的价签进行检测，利用 SHOPEWEB 的辅助判断，识别出指定货架上的价签 ID。棚格检测模块负责接收对指定的货架图片进行检测处理，基于价签和货架标识生成棚格信息。缺货检测模块负责接受对指定货架图片进行缺货检测，基于棚格信息对图片中每个棚格区域进行检测，生成缺货信息。

3.2.2 工作流程

电子价签检测系统的工作流程包括以下三个步骤：

(1) 初始化。首先将相机群的相机 IP 与货架 ID 关联，然后启动 AMP 模块，加载相机设备文件，相机管理器负责启动对相机的遍历，遍历过程中设置相机的参数，接着 AMP 模块发送初始设置命令到相机管理器，相机管理器负责获取相机群的图像，最后相机管理器负责将相机群的图像存储到图像文件存储器，并对获取的图像按照一定格式进行命名。

(2) 电子价签检测。AMP 模块向电子价签检测模块发送服务请求，传递参数：相机 ID，图片存储路径和图片名。电子价签检测模块接收到 AMP 模块的调用后，处理后返回价签信息给 AMP 模块。

(3) 与 WISE、SHOPEWEB 的对接。AMP 模块获取到缺货信息后，通过 HTTP 接口将信息传递给 WISE 系统，AMP 模块返回相关信息。

3.2.3 各模块数据及接口定义

本小节主要包含对部分模块的数据及接口定义，有助于设计算法表。

以棚格检测模块为例，对外输出的数据定义如下：

// ShelfGrid 数据结构：

{

“相机 ID”：相机编号

“检测时间”：图片拍摄时间戳

vector<GridInfo>：货架棚格图信息

}

// GridInfo 定义：

{

“货架 ID”：货架的编号，

“货架位置”：货架在图像中的位置，
“棚格图 ID”：棚格在图像中编号，
“棚格图位置”：棚格在图像中位置坐标，
“价签 ID”：价签在图像中编号，
“价签位置”：价签在图像中的位置，
“价签商品编码”：价签所对应的商品 SKU 编码
}

接口定义以 AMP 模块与 WISE 系统接口为例，AMP 模块与 WISE 系统对接的数据为缺货信息等：

//ShelfInfo 数据结构：
{
“相机 ID”：相机编号，
“检测时间”：图片拍摄时间戳，
vector< ProductInfo >：货架缺货信息
}
//ProductInfo 定义：
{
“货架 ID”：货架的编号，
“货架位置”：货架在图像中的位置，
“缺货商品名”：商品名称，
“缺货商品 SKU”： 商品 SKU 码，
“缺货程度”：濒临缺货/缺货，
“缺货棚格图 ID”：棚格图编号，
“缺货图像位置”：缺货棚格在图像中位置
}

3.3 数据集标注与分析

本文选用的数据来自某公司提供的真实商场电子价签图片数据集，并通过 labelImg 图片标注工具进行标注，标注信息存放在 xml 文件中。电子价签数据集总共包含 14713 张图片，检测对象超 12 万个。

首先通过 labelImg 图片标注工具对原始数据集图片进行标注，标注工具界面如图 3-2 所示：

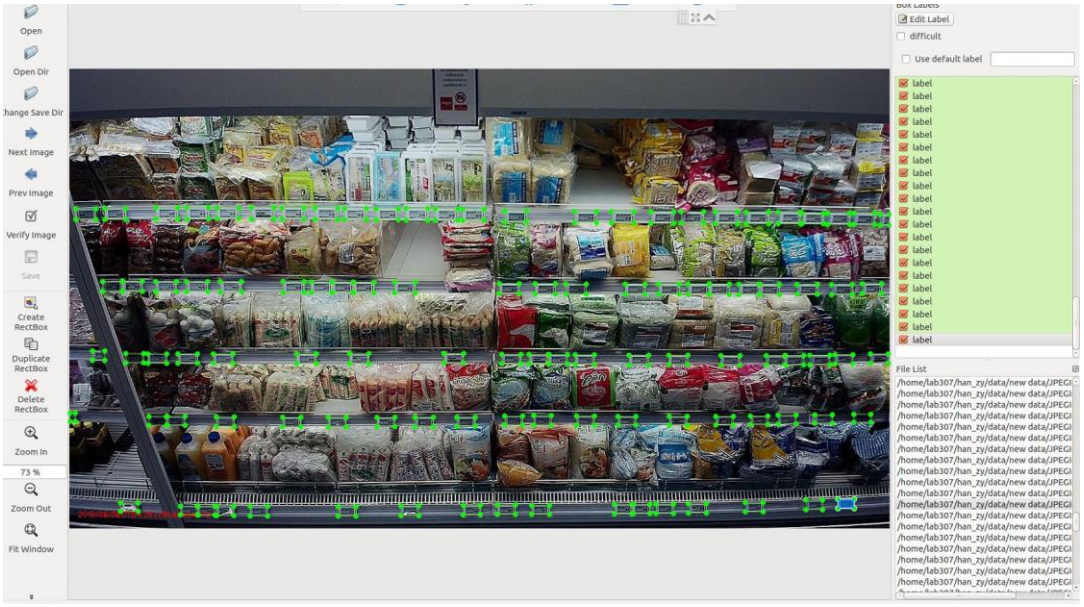


图 3-2 图片标注工具
Figure 3-2 Picture Annotation Tool

标注的信息将通过 xml 文件存放，xml 是一种具有结构性的源语言，能够对数据文件进行内容标记。xml 允许用户定义自己的标记内容，包括目标数据的记录，数据类型的定义等。xml 文件的存储结构如下所示：

```
<annotation>
    <folder> Electronic Shelf Label </folder>
    <filename>000001.jpg</filename>
    <path>/media/lab307/data/JPEGImages/000001.jpg </path>
    <source>
        <database> Electronic Shelf Label Database</database>
        <image>flickr</image>
    </source>
    <size>
        <width>486</width>
        <height>500</height>
        <depth>3</depth>
    </size>
    <segmented>0</segmented>
    <object>
        <name>person</name>
        <pose>Unspecified</pose>
```

```
        <truncated>0</truncated>
        <difficult>0</difficult>
        <bndbox>
            <xmin>174</xmin>
            <ymin>101</ymin>
            <xmax>349</xmax>
            <ymax>351</ymax>
        </bndbox>
        <part>
            <name>head</name>
            <bndbox>
                <xmin>169</xmin>
                <ymin>104</ymin>
                <xmax>209</xmax>
                <ymax>146</ymax>
            </bndbox>
        </part>
    </object>
</annotation>
```

xml 文件中标识了文件名称、位置、所属文件夹等信息。具体描述如表 3-1 所示：

表 3-1 xml 文件字段描述
Table 3-1 Detailed Description of xml

数据集字段	含义
<width>	输入图片的像素宽度
<height>	输入图片的像素高度
<depth>	输入图片的通道数
<name>	标注框所属类别
<difficult>	标注框是否属于难检测对象
<xmin>	标注框左下角横坐标
<ymin>	标注框左下角纵坐标
<xmax>	标注框右上角横坐标
<ymax>	标注框右上角纵坐标

本文的电子价签数据集主要由“Annotations”、“images”、“ImageSets”、“JPEGImages”、“labels”五个文件夹组成。其中“JPEGImages”文件夹存放的是未经处理的原始图片，总共包含 14713 张电子价签图片。“images”文件夹存放的是对原始图片重新编排后的所有图片，使用六位序列号从 000001 开始重新编排。“Annotations”文件夹里包含所有标注完的 xml 文件，序列号和“images”图片一一对应。“ImageSets”文件夹中式划分训练集、验证集和测试集的索引文件，本文通过 python 文件随机生成 6:2:2 的训练集、验证集和测试集。“labels”里存放的是所有图片在主机中的准确地址，用于快速将图片输入模型网络。

3.4 图片预处理

预处理的主要工作是将输入图像产生的张量转换成 PyTorch 可以使用的样式，并且使用一些数据增强手段，能够达到扩充数据样本，提升模型泛化能力的效果。

首先导入 Torchvision 库和子包 transform 和 datasets，在相关的 python 文件中输入：

“import torchvision”

“from torchvision import datasets,transform”

创建一个预处理格式列表：

```
“transform_train_list = [
        transforms.Resize((256,128), interpolation=3),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])]”
```

然后将预处理格式组合：

“data_transforms = transforms.Compose(transform_train_list)”

接着读取本地图片并进行预处理：

```
“image_datasets = datasets.ImageFolder(
        os.path.join(data_dir, train_all),data_transforms)”
```

最后创建加载器：

```
“dataloaders = {torch.utils.data.DataLoader(
        image_datasets, batch_size=opt.batchsize,
        shuffle=True, num_workers=8, pin_memory=True)”
```

常见的数据增强方法有：

(1) 随机比例缩放。“new_im = transforms.Resize((100, 200))(im)”

(2) 随机位置剪裁。“new_im = transforms.RandomCrop(100)(im)”

(3) 随机水平/垂直翻转“new_im = transforms.RandomHorizontalFlip(p=1)(im)”

(4) 随机角度旋转。“new_im = transforms.RandomRotation(45)(im)”

(5) 色度、亮度、饱和度、对比度等变化。

“new_im = transforms.ColorJitter(brightness=1)(im)”

“new_im = transforms.ColorJitter(contrast=1)(im)”

“new_im = transforms.ColorJitter(saturation=0.5)(im)”

“new_im = transforms.ColorJitter(hue=0.5)(im)”

3.5 本章小结

本章主要介绍了电子价签检测系统的整体框架搭建，并对各个模块内容，运行流程和数据接口进行了简单的描述。同时展示了数据集的标注工作，并针对现在拥有的电子价签数据集进行了简单的分析，详细了解了数据集的规模，存储文件包含的标注信息等，最后对电子价签数据集进行了基础的预处理操作，给出了详细的数据预处理方法，为第五章验证电子价签检测算法的优劣奠定了基础。

4 小尺度、高密度的目标检测

本章描述本文设计的基于深度学习的目标检测算法的实现过程。本章基于SSD 的模型结构，根据数据集小尺度、高密度的特点，提出了一种具有针对性的特征融合方式，通过 concatenate 操作将 SSD 的低层特征图和高层特征图整合，实现了低层位置信息和高层语义信息的信息互补。接着，本章引入了注意力（Attention）模块，模型在底层特征图上使用注意力机制，让深度神经网络更加关注小尺度目标，同时减少了背景中不必要的浅层特征信息，进一步提升了模型的检测精度。

4.1 问题定义与解决思路

目标检测任务中对小尺度目标的判定有两种方法，第一种是绝对尺寸的大小，如果检测目标的尺寸小于 32*32 像素，那么这个检测目标即被认为是小尺度目标；第二种是相对尺寸的大小，如果检测目标的长和宽不足输入图像的十分之一，那么它就属于小尺度目标。本文数据集中的电子价签满足上述两中定义方式，因此属于小尺度的目标检测。与通常的小尺度目标检测不同，本文待检测图片中的电子价签数量众多，在区域内分布密集，因此本文的电子价签检测是一个小尺度、高密度的目标检测问题。

之前的 SSD 等模型并不能很好地检测出电子价签，原因是 SSD 等模型仅对每一个特征图进行预测。高层的特征图经过反复卷积，覆盖的图像范围更广，包含的语义信息也更多，但却丢失了大量的位置信息，不利于电子价签的目标定位。低层的特征图则与之相反，因为之前经过的卷积次数较少，包含图像的特征语义信息较少，并不能有效检测出电子价签，这就造成了对电子价签的召回率不高的情况。因此对电子价签目标的检测，需要提供上下文信息的补充。而现有的 SSD 特征融合方法并不能有效提供上下文信息，因为之前的融合方法仅在 conv4_3，fc7，conv7_2 三层上做融合，并在此基础上继续做卷积生成新的特征图，但是本文的电子价签数据集检测对象分布密集，且尺度差别不大，仅做一次特征融合并不能有效提取出小范围区域内的特征信息，且在融合的特征图上继续卷积生成新的特征图则又会使特征图语义信息过深，降低对检测目标的召回率。因此本文需要设计一种既能多次融合特征，又能保持模型特征图语义信息适当的网络结构。

针对上述电子价签数据集的特点和设计需求，本文提出了一种具有针对性的特征图融合 SSD 结构。两次将 SSD 中三个相邻用于检测的特征图进行特征融合，充分利用了网络中相对低层特征图的位置信息和相对高层的语义信息。两次特征

融合均是对三个相邻的特征图进行融合，这种相邻特征图融合的方式联合了相近特征图的特征，并辅以上下文信息，取长补短，增强了底层特征图的语义信息和位置信息，可以有效提升对小尺度目标的召回率。另一方面两次特征融合的特征图相近且都尺度偏低，相当于对区域内的信息做了多次的特征提取，因此融合后的特征图保证了对分布密集的电子价签的特征提取，提高了对区域密集的电子价签的检测精度。

本文通过引入新的特征融合方式提升了模型的检测精度，但是融合的特征图虽然能够结合上下文信息，却没有侧重点。所有融合的特征图只是机械化地拼接在一起，待融合的特征图之间属于并行关系，因此本文希望在特征图融合之前引入一个注意力模块，通过注意力机制使得融合的特征图更加关注低层的特征，从而提升对电子价签的目标定位。

本文受文献[53]启发，通过模仿残差块的网络结构来实现注意力模块。本文设计的注意力模块包含两个分支：主干道和辅干道。主干道可以是一部分卷积神经网络，辅干道通过对特征图的缩放处理，生成和输入相同分辨率的特征图，起到注意力的作用。本文在辅干道中，首先对输入特征图进行降维操作，获取特征图的全局信息，再进行上采样操作，生成高分辨率的特征图并与主干道的特征图相结合，使得网络学习到新的特征信息，起到注意力的作用。在待融合低层特征图后添加注意力模块，有助于网络更加关注低层特征图的信息，进一步提升对小尺度目标的检测精度。

在 4.2 和 4.3 小节中，本文将对设计的特征融合方法和注意力模块进行详细的描述，并给出整体的网络架构。

4.2 特征融合

卷积神经网络具有层次性，其中卷积、池化操作可以采集到局部图像的各种特征，并对获取的特征进行深层抽象和融合。卷积神经网络一般具有多个卷积层和池化层，随着层数的堆叠，特征图中所包含的信息增多。卷积神经网络从一开始低层特征图获取到一般特征（例如边缘特征，纹理特征等），逐渐演变成高层特征图获取到高级语义特征（例如人体特征）。因此，将不同层的特征进行融合互补，既可以使用到低层的位置信息，又可以使用到高层的语义信息，对提升网络整体的检测性能具有一定的帮助。

不同层的特征图由于卷积等操作，一般分辨率是不同的。因此在对不同分辨率的特征图进行特征融合之前，需要将它们转化成相同的尺寸。在卷积神经网络中，池化操作可以对高分辨率特征图实现特征降维，从而使得模型可以抽取到更加广

泛的特征，从而也被称作下采样。相反的，将特征图从低分辨率转化成高分辨率的方法则被称作上采样。在卷积神经网络中，常用的上采样方法是反卷积（deconvolution）和双线性插值（bilinear）。

一般深度学习平台中实现反卷积功能的方式如图 4-1 所示，将一个分辨率大小为 4*4 的特征图上采样为 6*6 的特征图，首先对输入的 4*4 特征图的所有像素点周围扩充一圈 0，然后对 7*7 大小的特征图进行一次 padding 操作（即对整个特征图外围一圈扩充 0），再通过 3*3 的卷积核对特征图进行卷积，最后剪裁得到 6*6 大小的特征图。通过反卷积的操作，输出特征图的分辨率刚好是输入的两倍。

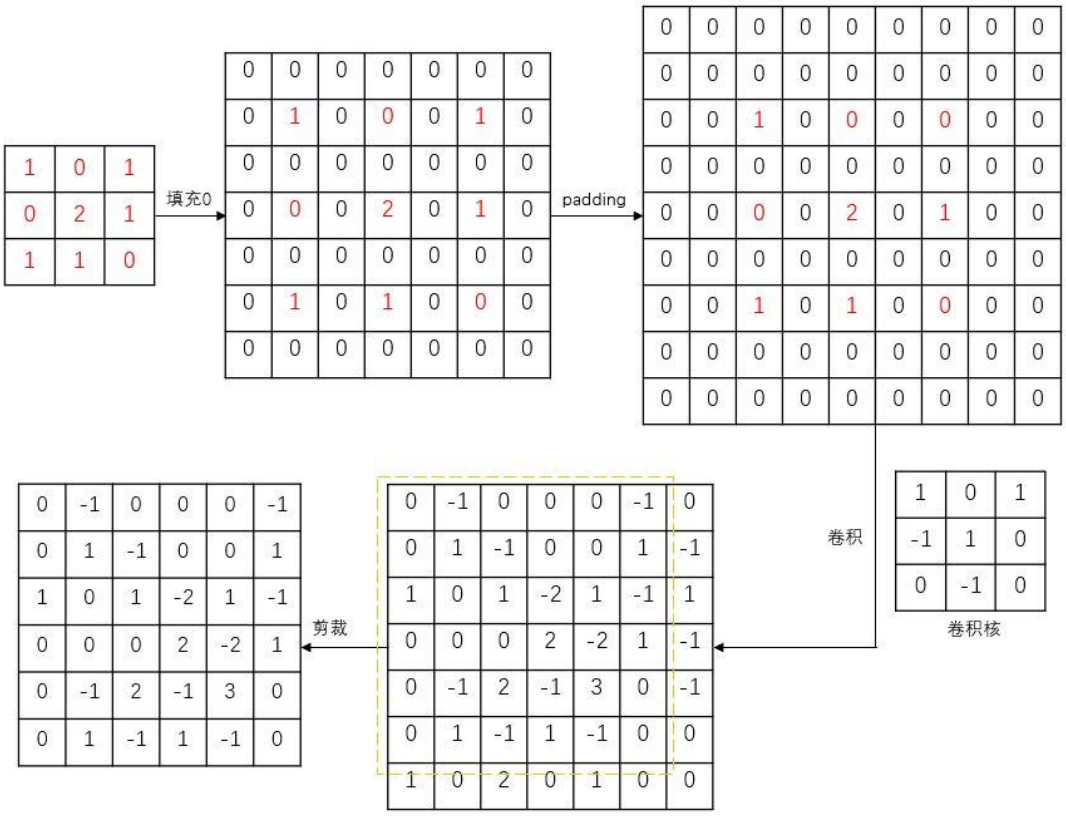
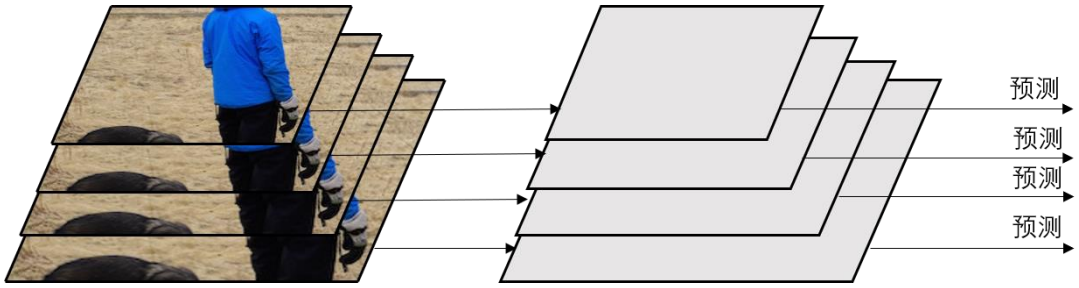


图 4-1 反卷积示意图
Figure 4-1 Deconvolution Diagram

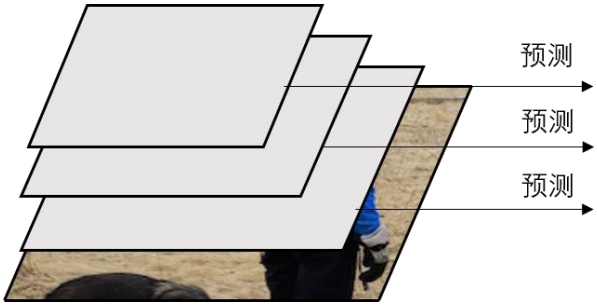
4.2.1 多尺度检测

传统的目标检测方法通常使用图像金字塔（Image Pyramid）来解决多尺度目标检测问题。图像金字塔使用不同分辨率的组合形式来表示图像的多尺度信息，这种方法处理方式简单直接但效率较高。图 4-2 a)展示了图像金字塔生成特征金字塔的过程，左侧是一系列以金字塔形状排列的图像，它们的分辨率逐步降低，且来源于同一张原始图的图像集合。特征金字塔直接从图像金字塔中获取，每一层不同分

辨率的图像经过卷积神经网络提取出各自的特征。虽然特征金字塔具有丰富的语义信息，但是因为每一层图像都需要通过卷积神经网络来获取特征，导致模型整体的时间复杂度较高。如图 4-2 b)，SSD 算法采用了特征金字塔的堆叠方式，不同之处在于，SSD 算法不需要将每一层图像都输入卷积神经网络来获取特征，而是将图像通过一个基础网络（例如 VGG）来获取顶层特征，在基础网络之后又增加了几层特征图，分别抽取不同层的特征进行目标检测，从而大大减少了提取特征的时间成本。



a) 特征金字塔
a) Feature Pyramid



b) SSD 检测方法
b) SSD Detection Method

图 4-2 多尺度检测方式
Figure 4-2 Multiscale detection method

本文设计的特征融合模型如图 4-3 所示，在 SSD 的基础上，两次将 SSD 中三个相邻用于检测的特征图进行特征融合，首先对 conv4_3，conv7，conv8_2 三个特征图进行特征融合，再对 conv7，conv8_2，conv9_2 三个特征图进行特征融合，充分利用了网络中相对低层特征图的位置信息和相对高层的语义信息。因为相邻两个特征图进行融合时，往往下层特征图大小是上层特征图的两倍，所以需要对上层特征图进行两倍的上采样操作。当输入图像的尺寸为 300*300 时，首先对 conv4_3，conv7，conv8_2 三个特征图进行特征融合时，需要先对 conv7，conv8_2 两个特征图进行上采样操作得到相同分辨率大小的特征图，融合得到大小为 38*38*512*2 大小的特征图。再对 conv7，conv8_2，conv9_2 三个特征图进行特征融合时，同理需

要先对 conv8_2，conv9_2 两个特征图进行上采样操作得到相同分辨率大小的特征图，融合得到大小为 19*19*1024*2 大小的特征图，最后在高层特征图以及融合过的特征图上获得分类和边框定位的结果。用来预测的高层特征图经过卷积具有不同的分辨率，其中 conv8_2 层的分辨率为 10*10，conv9_2 层的分辨率为 5*5，conv10_2 层的分辨率为 3*3，conv11_2 层的分辨率为 1*1。

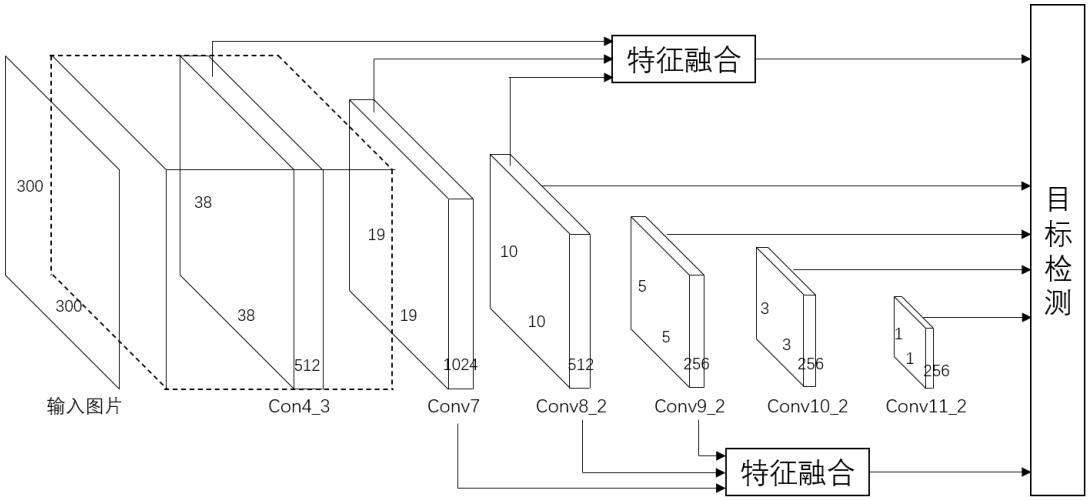


图 4-3 引入特征融合的模型
Figure 4-3 Model with Feature Fusion

4.2.2 融合方式

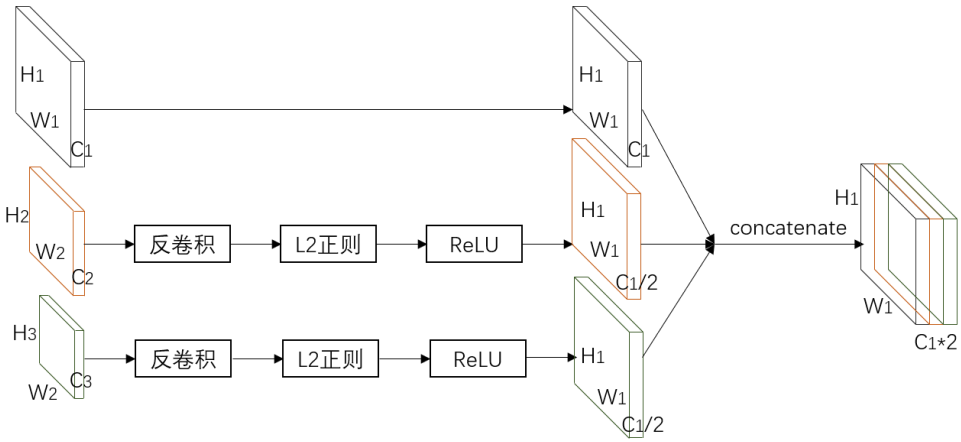


图 4-4 特征图的融合过程
Figure 4-4 Fusion Process of Feature Map

以 conv4_3，conv7，conv8_2 特征融合为例，融合过程如图 4-4 所示。分别对 conv7 和 conv8_2 进行反卷积操作，再经过 L2 正则化，ReLU 变换等操作，得到三个分辨率相同的特征图，再经过 concatenate 操作将三个特征图整合成为一个特征图，通道数变为最低层特征图通道数的两倍。如图 4-5 所示，concatenate 操作是将

两个或者多个分辨率相同，而通道数不同的特征图串联起来。得到的特征图不改变分辨率大小，只改变通道的数量，从而起到信息融合的作用。经过三个特征图 conv4_3，conv7，conv8_2 融合得到的特征图和 conv4_3 具有相同的分辨率，但是具有更深层的语义信息，因此在融合后的特征上进行分类和边框回归，能够得到更好的检测结果。

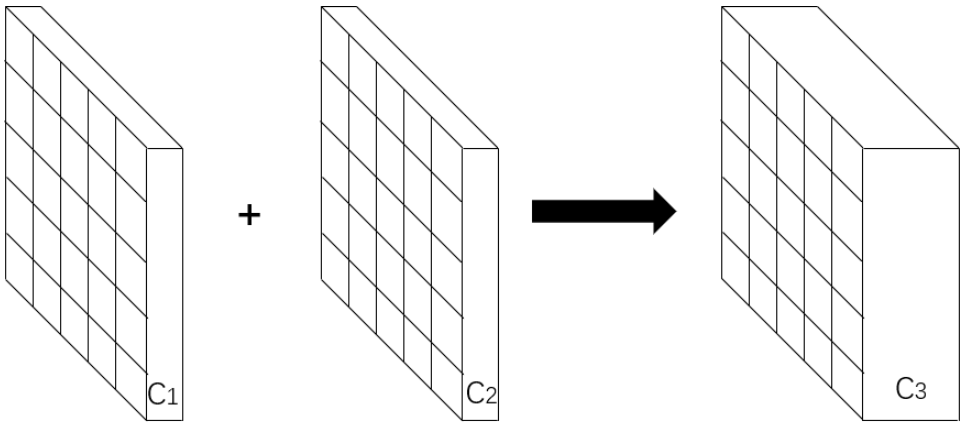


图 4-5 Concatenate 操作
Figure 4-5 Concatenate Operation

4.3 特征融合引入注意力机制

近几年，注意力机制在人工智能领域有着大量的应用，促使自然语言处理、计算机视觉、统计学习、智能推荐等领域飞速发展，现在已成为神经网络领域的一个重要概念。如果利用人类视觉机制进行直观的解释，注意力机制就是使我们的视觉系统更加倾向于关注图像中的部分目标信息，并忽略掉不相关的信息。在计算机视觉领域中，输入图像的某些部分可能会比其他部分对网络决策更有帮助。因此在某些特定的问题中，引入注意力机制有助于增强模型的判断能力，从而提升检测的精确率。

4.3.1 注意力模块

本文设计的注意力模块主要通过残差网络（residual network）进行实现。残差网络最初设计出来是为了解决深度神经网络中的退化问题，即深度神经网络随着网络深度的增加，并不能起到模型性能提升的作用，反而会因为网络层数太深，而出现梯度消失等现象，导致模型的整体训练效果变差。残差网络主要由残差块（residual block）堆叠而成。图 4-6 展示的是残差块的网络结构，当输入为 x 时，

学习到的特征记为$H(x)$，我们希望通过网络学习到残差$F(x)=H(x)+x$。当残差为 0 时，此时残差块相当于做了一次恒等映射，并不会使得网络性能下降。但实际网络模型中，残差不会为 0，这也使得残差块在输入特征的基础上又学习到了新的特征，从而提升模型整体的性能。

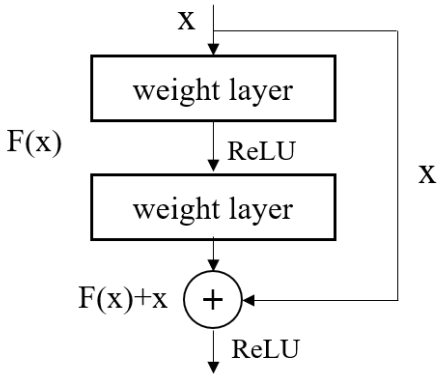


图 4-6 残差块
Figure 4-6 Residual Block

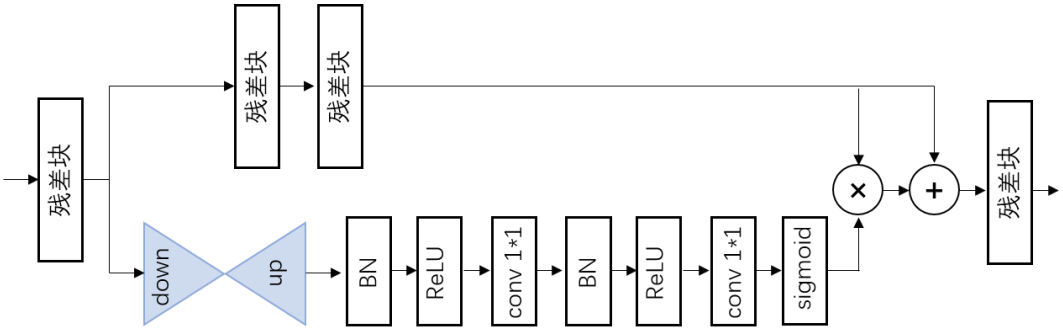


图 4-7 注意力模块
Figure 4-7 Attention Model

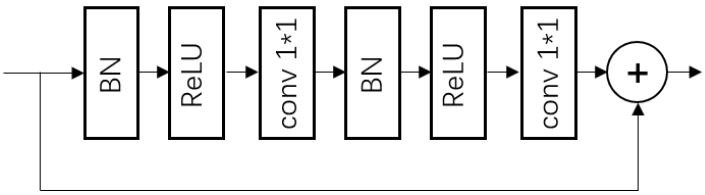


图 4-8 残差块网络结构
Figure 4-8 Network of Residual Block

本文设计的注意力模块如图 4-7 所示，输入特征经过一个残差块后分为两个干道，残差块的具体网络结构如图 4-8 所示，主干道中经过两个残差块，不改变主干道分辨率大小。辅干道经过一个 down-up 模块，然后经过两个 1*1 卷积层，最后通过 sigmoid 激活函数层将特征图归一化到[0,1]之间。整个注意力模块可以通过如

下公式表达：

$$H_{i,c}(x) = (1 + M_{i,c}(x)) * F_{i,c}(x) \tag{4-1}$$

其中 i 代表输入的特征图位置，c 表示特征图的通道数量，$F_{i,c}(x)$ 为主干道的特征，$M_{i,c}(x)$ 为经过 sigmoid 激活函数得到的每个像素点数值在[0,1]之间的特征图，$H_{i,c}(x)$ 是最终得到的特征图。

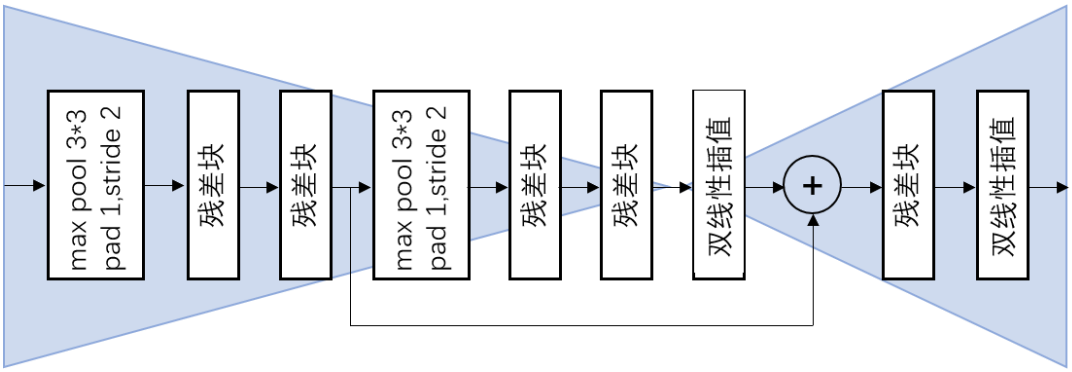


图 4-9 down-up 模块网络结构
Figure 4-9 Network of Down-up Model

注意力模块中辅干道经过的 down-up 模块是注意力模块的核心，也是注意力机制的表达，down-up 模块的网络结构在图 4-9 中进行了详细的说明。首先 down-up 模块使用 max pooling 操作对辅干道中输入的特征图进行下采样操作，将特征图缩放到分辨率为网络输出的 最小尺寸，接着再使用双线性插值等方法对全局特征图进行扩充，逐渐放大到与输入特征图相同分辨率大小的尺寸，并对中间相同尺寸的特征图进行 add 操作。通过这种方式网络具有更强的特征表达能力，因为 down-up 网络的里不仅包含了特征图的全局信息，还包含了特征图的局部特征。在 down-up 模块之后，对特征图使用了 2 个 1*1 的卷积层，主要是为了改变特征图的通道数，输出一个通道数仅为 1，分辨率大小与主干道特征图相同的权值特征图。最后使用 sigmoid 激活函数将权值特征图的所有数值归一化到[0,1]之间，并与主干道的特征图点乘，使得主干道输出特征图 $F_{i,c}(x)$ 中加强相关的特征，抑制不相关的特征。最后再将点乘的特征图与输入特征图相加，得到最终的输出特征图 $H_{i,c}(x)$。注意力模块理论上可以不断叠加，但是多层叠加注意力模块会使得模型整体的运算效率下降，影响模型整体性能。

4.3.2 模型构架

本文引入注意力模块后整体模型构架如图 4-10 所示，在 conv4_3，conv7，conv8_2 三个特征融合层的最低层 conv4_3 之后引入注意力模块，在 conv7，

conv8_2，conv9_2 三个特征融合层的最低层 conv7 之后引入注意力模块。

在较低层之后引入注意力模块使得特征融合的过程更加关注低层特征图，因为检测对象电子价签的目标较小，较低的特征图更能获取电子价签的特征信息，而在这些特征层之后引入注意力模块，使得整个模型更加注重这些特征图提取的特征，从而起到“注意力”的作用。另一方面，注意力模块中主要的组成单元是残差块，这种可以多次堆叠的网络在不会引起梯度消失等问题的基础上，加深了网络的层数，使得网络能够学习到更多的目标特征，有效减少光照、人流等因素的干扰。

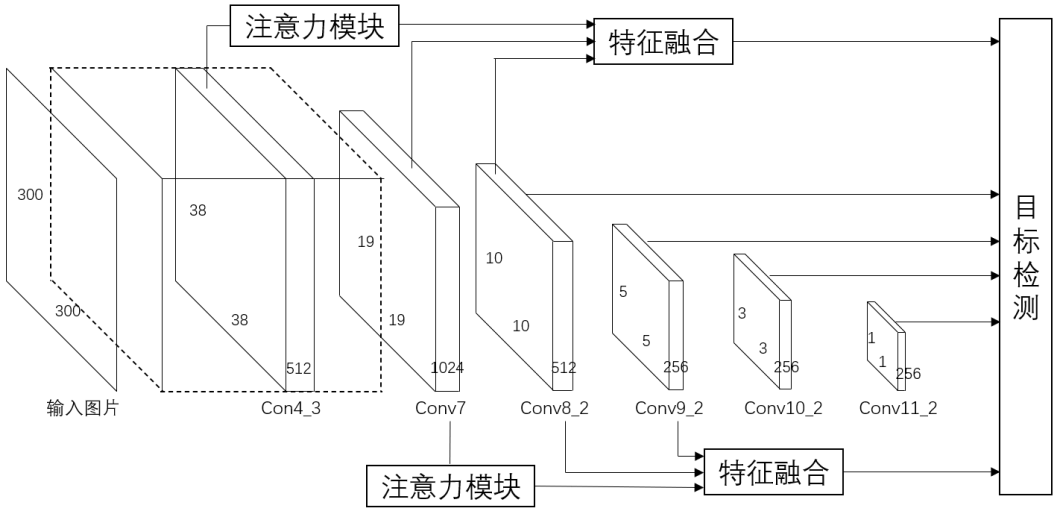


图 4-10 引入注意力机制的模型
Figure 4-10 Model with Attention

4.4 训练方法

本文模型的训练方法参考 SSD 系列算法，并根据电子价签数据的特点，对目标函数，默认框的选取等部分进行了改良，从而进一步优化了整体模型。

4.4.1 目标函数

本文模型训练过程中的总目标损失函数由两个部分组成，对应类别的置信度损失函数 $L_{conf}(x,c)$ 和搜索框的位置损失函数 $L_{loc}(x,l,g)$，公式如下：

$$L(x,c,l,g)=\frac{1}{N}(L_{conf}(x,c)+\alpha L_{loc}(x,l,g)) \tag{4-2}$$

其中 c 表示默认框的置信度，预测框通过 l 来表示，真实框通过 g 来表示，α 表示权重项，N 表示匹配到的 default box 的数量。本文中的 α 设置为 1。分类损失函数

$L_{conf}(x,c)$ 的概率通过 softmax 函数产生，公式如式(4-3)和式(4-4)所示，

$$L_{conf}(x,c) = -\sum_{i\in Pos}^{N} x_{ij}^{P} \log(\hat{c}_i^{p}) - \sum_{i\in Neg}^{N} \log(\hat{c}_i^{0}) \tag{4-3}$$

$$\hat{c}_i^{p} = \frac{\exp(c_i^{p})}{\sum_{p}\exp(\hat{c}_i^{p})} \tag{4-4}$$

其中 i 代表默认框的编号， j 代表真实框的编号， p 代表类别编号，当 p 为 0 时表示没有检测到目标。$x_{ij}^{p}=\{1,0\}$ 中取 1 表示此时第 i 个默认框和第 j 个真实框的重叠部分大于阈值，且真实框中的目标类别为 p 。因为电子价签数据集中只有电子价签和背景两种分类，所以式(4-3)可以简化为式(4-5)：

$$L_{conf}(x,c) = -\sum_{i\in Pos}^{N} x_{ij}^{1} \log(\hat{c}_i^{1}) - \sum_{i\in Neg}^{N} \log(\hat{c}_i^{0}) \tag{4-5}$$

定位损失函数 $L_{loc}(x,l,g)$ 和 Smooth L1 损失函数类似，如式（4-6）和式(4-7)所示，通过回归默认框的中心坐标 (cx,cy) 以及宽 w 和高 h 的偏移量来计算预测框 l 与真实框 g 之间的损失。

$$L_{loc}(x,l,g) = \sum_{i\in Pos}^{N}\sum_{m\in\{cx,cx,w,h\}} x_{ij}^{k} smooth_{L1}(l_i^{m} - \hat{g}_j^{m}) \tag{4-6}$$

$$\begin{cases} \hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^{w} \\ \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^{h} \\ \hat{g}_j^{w} = \log(g_j^{w}/d_i^{w}) \\ \hat{g}_j^{h} = \log(g_j^{h}/d_i^{h}) \end{cases} \tag{4-7}$$

4.4.2 默认框的选取

卷积神经网络中不同层有着不同尺寸的感受野（receptive fields），感受野指的是卷积神经网络中某个卷积特征图上的最小单位映射到原始图片上的分辨率大小。因此对不同感受野的特征图需要微调默认框的尺寸大小，本文中默认框的计算公式如下所示：

$$s_k = s_{\min} + \frac{s_{\max} - s_{\min}}{m-1}(k-1), k\in[1,m] \tag{4-8}$$

其中 m 表示总共需要做预测的特征图的个数， $s_{\min}$ 表示最低层特征图的尺度， $s_{\max}$ 表示最高层特征图的尺度，本文中 $s_{\min}$ 和 $s_{\max}$ 分别取 0.2 和 0.8。每一个默认框的宽

和高可以通过式(4-9)和(4-10)计算得到，默认框的中心位置可以通过式(4-11)获取：

$$w_k^a = s_k\sqrt{a_r} \tag{4-9}$$

$$h_k^a = s_k\sqrt{a_r} \tag{4-10}$$

$$(\frac{i+0.5}{|f_k|},\frac{j+0.5}{|f_k|}), i, j \in [0,|f_k|) \tag{4-11}$$

其中 a_r 表示默认框的长宽比，$|f_k|$ 是第 k 个特征图的分辨率，本文中 a_r 设置为 $a_r = \{1,2,3,1/2,1/3\}$ 。根据电子价签数据集的样本特点，本文并没有在长宽比为 1 时添加一个默认框，所以本文在特征图的每个像素点上都只预设了 5 个默认框。

4.4.3 数据增强

数据增强（data augmentation） 目的是使用人工手段从已有数据中产生一批新的数据的方法。数据增强有助于增加训练的图片数据量，提升模型的泛化能力；增加噪声数据，提高模型的鲁棒性。本文模型也使用了数据增强手段，随机对一张输入图片进行以下操作：

(1) 使用全部原始图片。

(2) 采样图片中的一部分，使其和目标的最小 Jaccard 重叠部分是 0.1，0.3，0.5，0.7 或 0.9。

(3) 随机获取一个采样部分。

采样得到的小块相当于对原始图片进行了裁剪、缩放操作。采样之后，我们以50%的概率对小块进行翻转操作，生成新的图像块。

为了增加真实框和预测框的匹配量，本文对每张图片中的部分目标进行拷贝、粘贴的操作，并且确保粘贴的目标不会与任何检测对象重叠，不会超出图像边界。粘贴的目标会随机进行尺度大小变换和旋转变换，进一步增强了模型的鲁棒性。

4.4.4 训练流程

本文的训练流程如图 4-11 所示，首先通过 PyTorch 深度学习框架搭建网络模型，接着对网络中参数进行设置，例如学习率，batch 的大小等等。初始化网络参数之后就可以向模型输入图片并开始训练网络权重。输入的图片经过数据增强后，根据本文提出的匹配策略，将默认框和真实框进行匹配，生成的正负样本获取本轮训练的预测值，分别计算出分类损失函数和定位损失函数。通过总的损失函数进行

反向传播，更新网络模型中的参数，进入下一轮训练，直到达到提前设置的最大训练轮数，网络的训练停止。

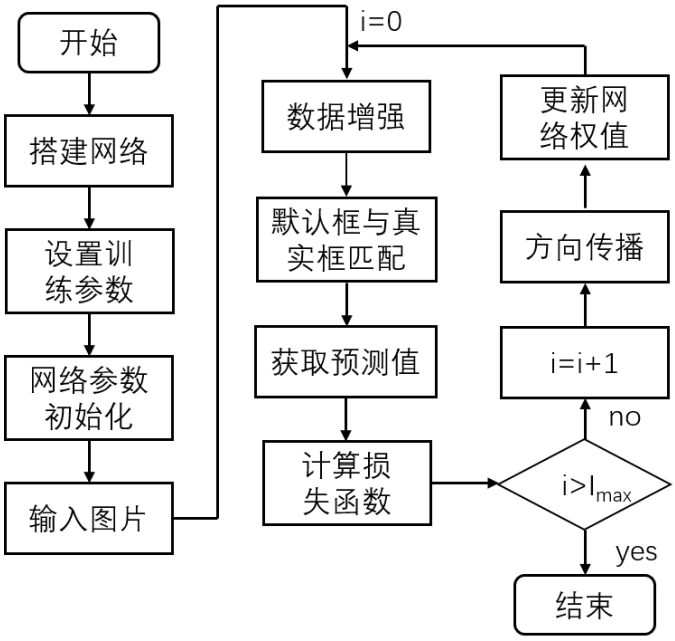


图 4-11 训练流程图
Figure 4-11 Training Flow Chart

4.3 本章小结

本章主要对本文提出的电子价签检测模型算法进行了详细的描述。从模型的特征融合方式，引入的注意力模块，训练方法三个方面进行了细致的描述。首先，针对电子价签数据集的特点，本文设计了一种新的特征融合方式，通过 concatenate 操作将低层特征图和高层特征图进行融合，有效提升了模型的检测精度。其次，本文引入注意力机制。在特征融合之前的低层特征图后增加注意力模块，使得模型能够更加关注电子价签的小目标特征，抑制不相关特征，提升了模型的精确率。最后给出了模型的损失函数，目标框的选取策略，图片数据增强方法以及网络的训练流程图，并对网络中的公式原理、参数设置进行了详细说明。

5 实验对比与结果分析

本章主要对本文所设计电子价签检测算法进行实验评估，利用选取的评价指标，对本文设计的模型算法的效果进行测评。首先展示本文设计的目标检测算法在电子价签数据集上的训练过程和检测效果，并且对特征图做可视化分析。接着将本文设计的模型逐一拆分对比，分析引入模块的作用及优化效果。

5.1 参数设置与结果展示

本小节将本文所选取的电子价签数据集与设计的目标检测算法相结合，利用电子价签检测算法对现有数据进行检测。在本节中主要对系统模型的参数进行说明和分析，调节参数获取最优的模型性能。之后将本文所设计的电子价签检测算法的训练过程、检测结果进行展示，并对结果进行简要分析。

本文实验的硬件配置为一个型号为 Intel(R) Core(TM) i7-7700K @ 4.20GHz 的 4 核 CPU，一个型号为 GTX 1080 的 GPU，16GB 内存等。软件配置为 Ubuntu16.04.1，cuda9.0.176，OpenCV，PyTorch 等。

5.1.1 参数设置

本文实验的模型训练参数如表 5-1 所示。经过大量的实验，本文根据软硬件条件和实验环境选取了最为合适的模型训练参数，保证了模型训练过程的稳定性。其中，学习率设置为 0.001，调整学习率的策略设置为 steps，之后网络中的调整策略还有：CONSTANT, EXP, POLY, SIG, RANDOM，steps 根据 batch_num 的数量在 100，25000，35000 调整学习率，变化比例分别为 10，0.1，0.1。batch（每次输入网络训练的图片个数）设置为 32，subdivisions（subdivisions 的作用是将每个 batch 再分割成数个子 batch，缓解计算机的内存压力，同时不降低训练的效果）设置为 4，当训练达到 max_batches 后停止学习。网络优化器选择 momentum，数值为 0.9。权重衰减正则项 decay 设置为 0.0005，可以有效防止过拟合。其他是一些生成更多训练样本的参数，例如 angle 可以通过旋转一定角度来生成更多的训练样本，saturation 和 exposure 可以通过调整图片的饱和度和曝光量来生成更多的训练样本，hue 则是通过调整色调来生成更多的训练样本。

表 5-1 模型训练参数
Table 5-1 Training Parameters of Model

参数名称	数值
learning_rate	0.001
police=steps	steps=100,25000,35000
scales	10，0.1，0.1
batch_size	32
max_batches	45000
subdivisions	4
momentum	0.9
decay	0.0005
angle	20
saturation	1.5
exposure	1.5
hue	0.1

5.1.2 实验结果展示

根据上一小节中的参数设置，本文设计的目标检测算法在训练过程中评价指标的变化曲线如图 5-1 所示，其中包含 Precision、Recall、mAP@0.5、F1 值的变化曲线。从表中可以发现，在网络的训练过程中，模型性能逐渐上升且在 100 轮后趋于稳定，最终四项指标均在 0.9 以上。召回率上升的同时，精确率也保持在一个较高的水平，因此本文设计的检测方法能够较好的检测出小尺度、高密度目标。

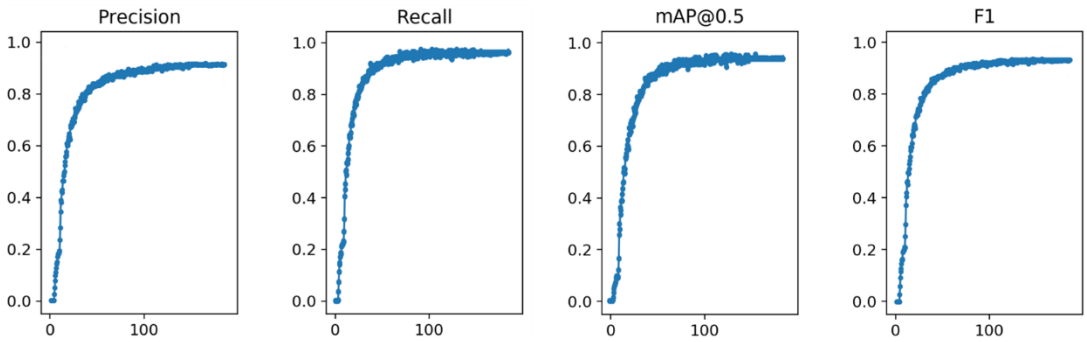


图 5-1 训练曲线
Figure 5-1 Training Curve

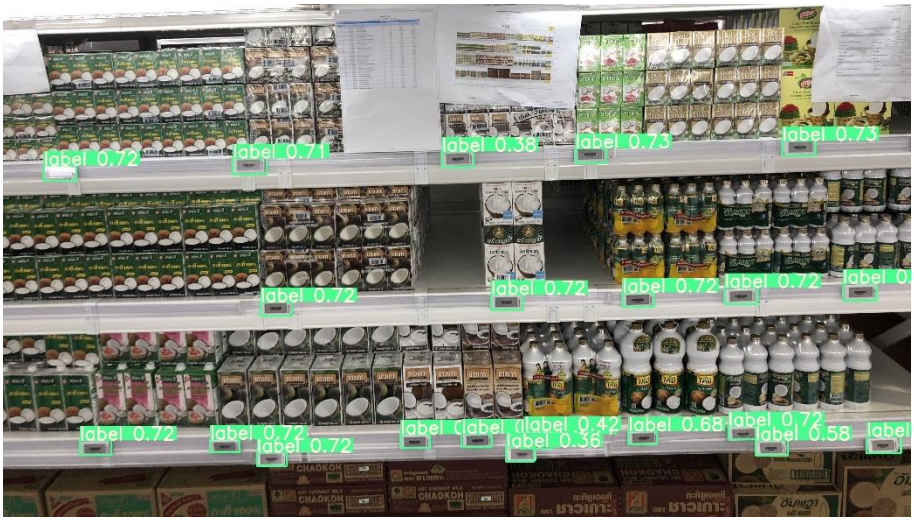
表 5-2 显示的是本文的目标检测算法在电子价签数据集的测试集上的检测结

果。从表 5-2 的检测结果可以看见，本文设计的目标检测算法精确率 0.929，被识别为电子价签的对象中大部分都是正确的；召回率高达 0.968，可以检测出大多数的电子价签。从另一方面看，本文设计的目标检测算法误检测和漏检测的概率极低，能够满足工业上线的需求。检测速度达到 29 帧/秒，满足了电子价签检测系统实时性的需求。

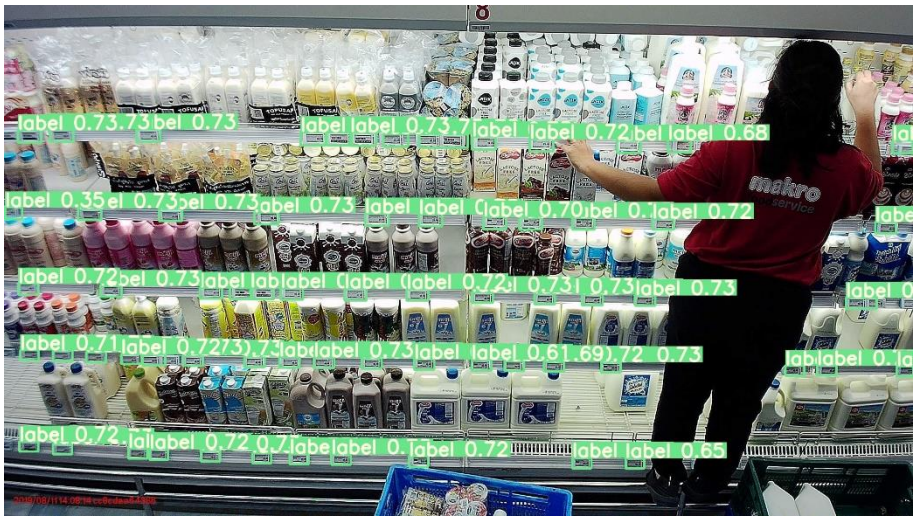
表 5-2 检测结果
Table 5-2 Detection Result

评价指标	数值
Precision	0.929
Recall	0.968
mAP@0.5	0.955
F1	0.948
FPS	29

本文设计的目标检测算法在电子价签数据集上单张图片的检测结果如图 5-2 所示。从图 5-2 a)和图 5-2 b)的检测结果可以发现，无论图片的关照是否充分、均匀，本文设计的检测算法都能准确检测出目标。从图 5-2 b)和图 5-2 c)的检测结果可以发现，本文设计的检测算法能够精确检测出小尺度、分布密集的目标，很少出现漏检和误检的现象，并且克服了人流等遮挡情况，能够检测出遮挡边缘的目标。综上所述，本文设计的电子价签检测算法克服了人流、光照等不利因素，在复杂的背景环境中精确地检测出了小尺度、高密度的目标，能够准确地识别出电子价签的位置和大小，且边框回归准确。



a) 第一张检测图像
a) First Detection Picture



b) 第二张检测图像
b) Second Detection Picture

c) 第三张检测图像
c) Third Detection Picture

图 5-2 单张图片检测结果
Figure 5-2 Detection Result of Single Picture

5.2 模型结果对比分析

本节主要对比不同目标检测算法在电子价签数据集上的检测效果，并且分析引入不同模块给整体模型带来的性能提升。本文选取 SSD 检测方法作为基线（baseline）进行试验，对其实验结果进行描述，并将实验结果与本文所提出的目标检测方法的结果进行对比，分析本文所提出的电子价签检测方法的优点。

5.2.1 SSD 基线模型

表 5-3 基线模型检测结果
Table 5-3 Detection Result of Baseline Models

检测方法	mAP@0.5	FPS
Faster R-CNN	0.887	7
YOLOv2	0.756	64
SSD	0.824	52

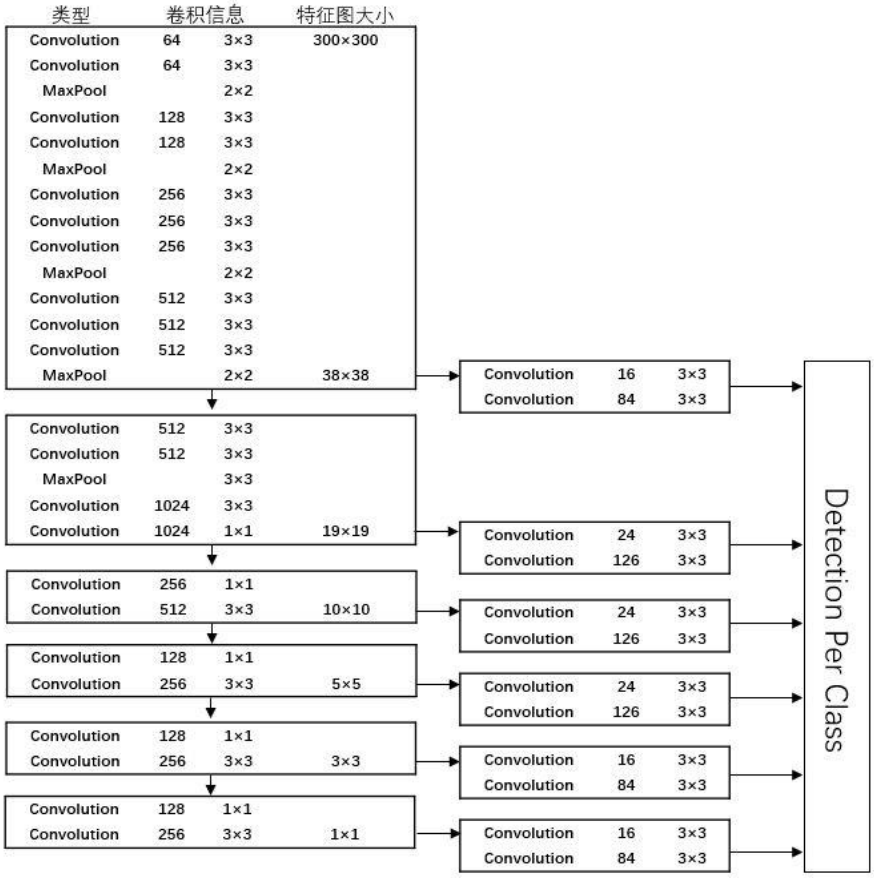


图 5-3 SSD 网络结构
Figure 5-3 Network Structure of SSD

本文在电子价签数据集上复现了 Faster R-CNN，YOLOv2，SSD 三种方法，实验结果如表 5-3 所示，其中 Faster R-CNN 的检测速度仅有 7 帧/秒，达不到实时性的要求，因此本文没有考虑将 Faster R-CNN 作为后续实验的基线模型。SSD 和 YOLOv2 的网络构架比较接近，因此使用这两个模型在电子价签数据集上进行了检测。通过检测结果可以看出，YOLOv2 的检测速率更高，能达到 64 帧/秒，但检

测精度与 SSD 相比，mAP 值低了 6.8%。另一方面，YOLOv2 的边框回归也不稳定，因此本文根据数据集的特点和电子价签检测系统的实时性需求，选择 SSD 作为基线模型进一步优化。

基线模型 SSD 的网络结构如图 5-3 所示。本文所有目标检测模型中使用的基础网络均为 VGG16，主要由 13 个 conv 层，5 个 pooling 层，3 个全连接层构成。本文使用的 SSD 模型也是在 VGG16 网络上进行更深层次的特征提取，以获取更好的目标检测效果。

本文所使用的电子价签数据集具有检测目标尺度小、分布密集等特点，较高的特征图的语义信息并不能很好的表示电子价签的特点，于是本文在下一小节的实验中对 SSD 模型进行了结构调整，并且删除了最后两个特征层的检测。最终的检测结果显示，根据数据集的特点调整过的 SSD 模型性能有小幅度的提升，mAP 值提升了 1.5%。

5.2.2 结果对比分析

通过以上对本文提出的电子价签检测方法与各基线模型方法的介绍与实验，最终将本文所涉及到的检测模型与实验结果进行整合，得到本文所设计的电子价签检测方法与基线方法的性能结果如表 5-4 所示。

表 5-4 各模型检测结果
Table 5-4 Detection Result of Various Models

检测方法	Precision	Recall	mAP@0.5	F1	FPS
SSD	0.848	0.868	0.839	0.858	59
特征融合 SSD	0.867	0.949	0.927	0.906	34
特征融合 SSD 引入注意力模块	0.929	0.968	0.955	0.948	29

根据表 5-4 中的数据结果，可以看出本文最终提出的电子价签检测方法相较于之前的检测方法来说检测性能大幅提升。特征融合 SSD 引入注意力模块的模型架构在 Precision、Recall、mAP@0.5、F1 四个评价指标上均取得最高值。SSD 模型引入本文设计的特征融合方法后，在电子价签数据集上精确率提升了 1.9%，召回率提升了 8.1%，mAP 值提升了 8.8%，F1 分数提升了 4.8%，模型能够找回更多的小尺度目标。添加注意力模块后，模型在电子价签数据集上相较于特征融合的 SSD

精确率提升了 6.2%，召回率提升了 1.9%，mAP 值提升了 2.8%，F1 分数提升了 4.2%，模型检测小尺度目标的精确度更高。检测模型的 mAp 值总共提高了 11.6%，检测速度达到 29 帧/秒，满足检测实时性的需求。对该结果进行深入研究，可以得到以下结论：

(1) 本文针对电子价签数据集的特点提出的新的特征融合方式，相较于之前的模型，mAP 值提高了 8.8%。之前的 SSD 模型将每个特征图送入检测器中进行结果预测，本文则将偏向低层的 conv4_3，conv7，conv8_2 和 conv7，conv8_2，conv9_2 通过 concatenate 操作进行特征融合，强化了检测模型对低层特征图的位置信息和语义信息的融合，使得模型能够更好的检测分布密集的电子价签目标群体，提升了对小尺度、高密度目标的检测性能。

(2) 本文在特征融合的基础上又引入的注意力模块，使 mAP 值在特征融合的基础上又提高了 2.8%。注意力模块通过残差块堆叠而成，能够使得局部网络结构更深，而不会出现梯度消失等现象。其中 down-up 模块通过降维和上采样的方式模拟残差块的结构，起到了注意力的作用。本文在特征融合层 conv4_3，conv7，conv8_2 中的低层特征图 conv4_3 后添加注意力模块，在特征融合层 conv7，conv8_2，conv9_2 中的低层特征图 conv7 后添加注意力模块。添加的注意力模块使得在特征融合的特征图中，更加关注层数较低的特征图，在一定程度上增强了融合特征中小目标的语义信息，从而进一步提升模型对小尺度目标的检测效果。

综上，对上述结果的解释是本文所提出的电子价签检测方法相对于以往的目标检测方法而言，在电子价签数据集上的检测更为准确。本文所提出的目标检测方法利用特征融合和注意力的共同作用，使得模型预测更加专注于小目标的低层特征图信息，能够更好的提取数据集中电子价签的目标特征，从而提升模型整体的检测性能，提高检测结果的准确率。

5.3 各模块结果与功能分析

上一小节对本文所提出设计的电子价签检测方法与以往的目标检测方法的效果进行对比，详细的说明了本文所提出设计的电子价签检测方法的优点。本节主要是对本文所提出设计的电子价签检测方法内部进行剖析，分析模型内本文所提出的特征融合方法与注意力模块的功能及作用并加以解释，以此来证明本文所提出设计的电子价签检测方法中各个模块都是不可或缺的。

5.3.1 特征融合的性能改善效果

表 5-5 特征融合方法性能分析
Table 5-5 Performance Analysis of Feature Fusion Method

检测方法	Precision	Recall	mAP@0.5	F1	FPS
SSD	0.848	0.868	0.839	0.858	59
特征融合 SSD	0.867	0.949	0.927	0.906	34

在本文所提出的电子价签检测方法中，新的特征融合方法能使得模型更加注重对小尺度目标的检测。两次底层特征图的融合使得模型能够更好的检测分布密集的电子价签目标群体。表 5-5 显示了引入了特征融合的检测方法与没有进行特征融合的检测方法的结果对比，没有进行特征融合的检测方法 mAP 值仅为 0.839，引入本文特征融合的模型召回率提升了 8.1%，能够检测出更多的正样本。从对比结果中可以分析出，本文设计的特征融合方法在提升电子价签检测结果方面有显著的效果。

5.3.2 注意力模块的性能增强效果

在本文所设计的电子价签检测方法中，又引入了注意力模块进一步增强了特征融合中最低层特征图的权重，进一步提升了对小尺度目标的检测性能。残差块的网络架构避免了梯度消失等问题的发生，同时辅干道的降维和上采样操作模拟残差块的结构，使得引入注意力模块的特征层能够学习到更多的特征信息。表 5-6 显示了引入注意力模块的检测方法与不包含注意力模块的检测方法的结果对比，精确率提升了 6.2%，模型能够更精确地检测出小尺度目标。显然引入了注意力模块的检测方法能够更加准确地检测出目标，具有更高的 mAP 值。从对比结果中可以分析出，本文设计的注意力模块在检测电子价签数据集上具有显著的效果。

表 5-6 注意力模块性能分析
Table 5-6 Performance Analysis of Attention Module

检测方法	Precision	Recall	mAP@0.5	F1	FPS
特征融合 SSD	0.867	0.949	0.927	0.906	34
特征融合 SSD 引入注意力模块	0.929	0.968	0.955	0.948	29

5.4 本章小结

本章主要对本文所涉及的实验进行归纳总结和对比，同时对产生结果的差异进行分析。首先对本文所设计的电子价签检测方法中涉及的参数进行简单说明，并选取最为合适的参数用于最终的实验，同时还展示了本文所设计的电子价签检测方法的训练过程、检测性能、检测结果样图。实验结果证明：本文提出的特征融合方法在电子价签数据集上可以将 mAP 值提高 8.8%，本文在特征融合中引入的注意模块将 mAP 值进一步提高 2.8%，共提升 11.6%最后对本文所设计的电子价签检测方法中特征融合模块与注意力模块进行解剖，分析各模块的作用以及对电子价签检测准确率提升所作的贡献，并对结果进行合理的解释。

6 总结及展望

本章对本文所做工作进行总结归纳并对本文相关方向的未来发展进行了展望。主要是对本文的贡献进行了归纳总结，详细的说明了本文所做的工作并列出了的本文所达成的关键性成果。同时也分析了本文实验中所存在的问题，主要包含尚待改进的地方和尚未解决的问题，并以此结合文本所研究内容的发展方向进行展望，同时对本文所提出的电子价签检测方法还可以改进的地方进行了阐述。

6.1 论文工作总结

本文设计并实现了一个电子价签检测系统，并将该系统用于识别“新零售”线下商场的电子价签。目前并没有合适的公开数据集，因此需要先采集图片，制作成合适的数据集进行实验。本文中的电子价签数据集具有检测目标数量多、尺度小、分布密集等特点，检测背景环境复杂且容易受到光照、人流等因素的影响，利用已有的目标检测方法并不能很好的检测出电子价签。因此本文针对电子价签数据集的特点提出了一种新的特征融合方式，即两次融合特征图，并引入注意力模块，使得整个网络的预测更加关注小目标和密集区域。下面将对本文的工作内容作详细的介绍：

(1) 获取、筛选和标注电子价签数据集，并进行数据分析和预处理。本文的电子价签数据集属于私有数据集，来自于线下商场的真实数据，根据图片的时间分布、环境特点进行筛选，使得电子价签数据集更加多样化。筛选后的数据集通过打标工具进行打标处理，制作成最终的电子价签数据集。电子价签数据集包含 14713 张电子价签图片，检测目标达 12 万以上。之后本文还统计、分析了数据集的基础信息，并对电子价签数据集图片进行了简单的预处理。

(2) 电子价签检测系统设计。本文根据真实的“新零售”商场功能需求，给出了电子价签检测系统的整体构架，并对系统中的各个模块的功能、进行了详细的说明。最后详细描述了电子价签检测系统的工作流程，并对各模块数据及接口进行了定义。

(3) 提出一种新的特征融合方法。针对电子价签数据集检测目标数量多、尺度小、分布密集等特点，本文设计了一种两次融合相邻特征图的融合方法。该方法能够更好地融合低层特征图的位置信息和语义信息，并且提升对区域密集目标的检测精度。经过实验证明，本文提出的特征融合方法在电子价签数据集上可以将 mAP 值提高 8.8%。

将注意力机制引入了电子价签检测算法，进一步提升了模型的检测精度。注意力模块通过残差块堆叠而成，更加关注融合之前的低层特征图，加强了对小尺度目标的检测。另外堆叠残差块能够加深网络的深度，使得低层特征图能够学习到更多的电子价签特征，避免光照等因素的影像。经过实验证明，本文引入的注意力模块在电子价签数据集上可以将 mAP 值提高 2.8%，共提升 11.6%

6.2 未来工作展望

本文设计了一个基于深度学习的电子价签检测方法，用于检测采集图片中的所有电子价签。电子价签的种类和尺度是多种多样的，内容也是千变万化，然而本文数据集中的电子价签种类比较单一。因此，未来的工作需要扩充数据集，丰富电子价签的种类，满足线下商场的各种检测需求。但数据集扩充之后，如何调整模型构架，精确检测出目标也将是一个挑战。本文的注意力模块可以重复堆叠，放置在不同的特征图之后可以改变特征融合的关注对象，因此需要通过大量实验证明，在电子价签数据集上堆叠多少个注意力模块是最佳的，是否能够通过不同特征图之后堆叠不同数量的注意力模块来增加网络的深度，学习到更多的电子价签特征，借此来提升检测精度。另一方面，尝试更好的目标检测方法来提升电子价签的检测准确率也是下一步的研究方向。

参考文献

[1] 商务部流通发展司. 中国零售行业发展报告(2018/2019 年)[M]. 北京:中国国际电子商务中心, 2019.

[2] 赵树梅, 徐晓红. “新零售”的含义、模式及发展路径[J]. 中国流通经济, 2017(05):14-22.

[3] 杜睿云, 蒋侃. 新零售:内涵、发展动因与关键问题[J]. 价格理论与实践, 2017, 000(002):139-141.

[4] Uijlings J R, Sande K E, Gevers T, et al. Selective Search for Object Recognition[J]. International Journal of Computer Vision, 2013, 104(2):154-171.

[5] Zitnick C L, Dollar P. Edge Boxes: Locating Object Proposals from Edges[C]// Proceedings of European Conference on Computer Vision. Springer International Publishing, 2014:391-405.

[6] Cheng M M, Zhang Z, Lin W Y, et al. BING: Binarized Normed Gradients for Objectness Estimation at 300fps[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014:3286-3293.

[7] Zhang Z, Liu Y, Bolukbasi T, et al. BING++: A Fast High Quality Object Proposal Generator at 100fps[J]. IEEE Trans Pattern Anal Mach Intell, 2016, PP(99):1-1.

[8] Viola P. A, MJ Jones. Rapid Object Detection Using a Boosted Cascade of Simple Features[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. IEEE, 2001.

[9] Lowe D G. Object Recognition from Local Scale-Invariant Feature[C]//Proceedings of the IEEE International Conference on Computer Vision. 1999:1150-1157.

[10] Ojala T, Pietikainen M, Maenpaa T. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. IEEE, 2002:971-987.

[11] Dalal N, Triggs B. Histograms of Oriented Gradients for Human Detection[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. IEEE, 2005.

[12] Burges C J C. A Tutorial on Support Vector Machines for Pattern Recognition[J]. Data Mining & Knowledge Discovery, 1998, 2(2):121-167.

[13] Cortes C, Vapnik V N. Support Vector Networks[J]. Machine Learning, 1995, 20(3):273-297.

[14] Yoav Freund, Robert E Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting[J]. Journal of Computer & System Sciences, 1999, 55:119-139.

[15] Leo Breiman. Random Forests[J]. Machine Learning, 2001, 45(1):5-32.

[16] Quinlan J R. Learning Efficient Classification Procedures and Their Application to Chess End Games[J]. Machine Learning, 1983:463-482.

[17] Utgoff P E. Incremental Induction of Decision Trees[J]. Machine Learning, 1989, 4(2):161-186.

[18] Krizhevsky A, Sutskever I, Hinton G. ImageNet Classification with Deep Convolutional Neural Networks[C]// Advances in Neural Information Processing Systems. Curran Associates Inc. 2012.

[19] Zeiler M D, Fergus R. Visualizing and Understanding Convolutional Networks[C]// European Conference on Computer Vision. Springer International Publishing, 2013.

[20] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Vison and Pattern Recognition, 2014.

[21] Szegedy C, Liu W, Jia Y, et al. Going Deeper with Convolutions[J]. IEEE International Conference on Computer on Computer Vision and Pattern Recognition. 2014.
[22] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[J]. IEEE International Conference on Computer Vision and Pattern Recognition. 2015.
[23] Girshick R, Donahue J, 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.
[24] He K, Zhang X, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. Transactions on Pattern Analysis & Machine Intelligence, 2015, 37(9):1904-1916.
[25] Girshick R. Fast R-CNN[C]// Proceedings of the IEEE International Conference on Computer Vision. 2015:1440-1448.
[26] Ren S, He K, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 39(6):1137-1149.
[27] Redmon J, Divvala S, et al. You only Look Once: Unified, Real-Time Object Detection[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:779-788.
[28] Liu W, Anguelov D, et al. SSD: Single Shot Multibox Detector[C]// Proceedings of European Conference on Computer Vision. Springer International Publishing, 2016: 21-37.
[29] Zhu Y, Urtasun R, Salakhutdinov R, et al. SegDeepM: Exploiting Segmentation and Context in Deep Neural Networks for Object Detection[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015:4703-4711.
[30] Bell S, Zitnick C L, Bala K, et al. Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:2874-2883.
[31] Zeng X, Ouyang W, Yan J, et al. Crafting GBD-Net for Object Detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, PP(99):1-1.
[32] Li J, Wei Y, et al. Attentive Contexts for Object Detection[J]. IEEE Transactions on Multimedia, 2017, 19(5):944-954.
[33] Kong T, Yao A, Chen Y, et al. HyperNet: Towards Accurate Region Proposal Generation and Joint Object Detection[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:845-853.
[34] Najibi M, Rastegari M, Davis L S. G-CNN: An Iterative Grid Based Object Detector[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:2369-2377.
[35] Gidaris S, Komodakis N. LocNet: Improving Localization Accuracy for Object Detection[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:789-798.
[36] Cai Z, Fan Q, Feris R S, et al. A Unified Multi-Scale Deep Convolutional Neural Network for Fast Object Detection[C]// Proceedings of European Conference on Computer Vision. Springer International Publishing, 2016:354-370.
[37] Fu C Y, Liu W, et al. DSSD: Deconvolutional single shot detector[J]. ArXiv Preprint ArXiv:1701.06659, 2017.
[38] Li Z, Zhou F. FSSD: Feature Fusion Single Shot Multibox Detector[J]. ArXiv Preprint ArXiv: 1712.00960, 2017.
[39] Huang G, LiuZ, Maaten L V D, et al. Densely Connected Convolutional Networks[C]//

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.
[40] Ren J, Chen X, Liu J, et al. Accurate Single Stage Detector Using Recurrent Rolling Convolution[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017:752-760.
[41] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017:6517-6525.
[42] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement[J]. ArXiv Preprint ArXiv:1804.02767, 2018.
[43] Duan K, Bai S, Xie L, et al. CenterNet: Keypoint Triplets for Object Detection[J]. ArXiv Preprint ArXiv:1904.08189, 2019.
[44] Law, Hei, Deng, Jia. CornerNet: Detecting Objects as Paired Keypoints[J]. International Journal of Computer Vision, 2018.
[45] Xingyi Zhou, Jiacheng Zhuo, Philipp Krähenbühl. Bottom-up Object Detection by Grouping Extreme and Center Points[J]. ArXiv Preprint ArXiv:1901.08043, 2019.
[46] Wang, Xinggang, Chen, Kaibing, Huang, Zilong, et al. Point Linking Network for Object Detection[J]. Computer Vision and Pattern Recognition, 2017.
[47] Ze Yang, Shaohui Liu, Han Hu, et al. RepPoints: Point Set Representation for Object Detection[J]. ArXiv Preprint ArXiv: 1904.11490, 2019.
[48] Wei Liu, Shengcai Liao, et al. High-level Semantic Feature Detection: A New Perspective for Pedestrian Detection[J]. ArXiv Preprint ArXiv: 1904.02948, 2019.
[49] Zhaowei Cai, Nuno Vasconcelos. Cascade R-CNN: Delving into High Quality Object Detection[J]. ArXiv Preprint ArXiv: 1712.00726, 2017.
[50] Wei Liu, Shengcai Liao, et al. Learning Efficient Single-stage Pedestrian Detectors by Asymptotic Localization Fitting[J]. The European Conference on Computer Vision (ECCV), 2018: 618-634.
[51] Jianan Li，Xiaodan Liang, et al. Perceptual Generative Adversarial Networks for Small Object Detection[J]. Conference on Computer Vision and Pattern Recoginition(CVPR), 2017.
[52] Zeming Li, Chao Peng, Gang Yu, et al. DetNet: A Backbone Network for Object Detection[J]. ArXiv Preprint ArXiv: 1804.06215, 2018.
[53] Wang F, Jiang M, Qian C, et al. Residual Attention Network for Image Classification[J]. Conference on Computer Vision and Pattern Recognition, 2017.

作者简历及攻读硕士学位期间取得的研究成果

一、作者简历

韩致远，男，1996 年 2 月生。2014 年 9 月至 2018 年 7 月就读于北京交通大学电子信息工程学院通信工程专业，取得工学学士学位。2018 年 9 月至 2020 年 6 月就读于北京交通大学电子信息工程学院电子与通信工程专业，研究方向是信息网络，取得工学专业硕士学位。在就读专业硕士学位期间，主要从事深度学习算法与目标检测方面的研究工作。

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京交通大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名： 签字日期： 年 月 日

学位论文数据集

表 1.1： 数据集页

关键词*	密级*	中图分类号	UDC	论文资助
深度学习；目标检测；特征融合；注意力	公开			
学位授予单位名称*		学位授予单位代码*	学位类别*	学位级别*
北京交通大学		10004	工学	硕士
论文题名*		并列题名		论文语种*
电子价签检测算法研究与系统实现				中文
作者姓名*	潘翰祺		学号*	18125015
培养单位名称*		培养单位代码*	培养单位地址	邮编
北京交通大学		10004	北京市海淀区西直门外上园村 3 号	100044
专业学位*		研究方向*	学制*	学位授予年*
电子与通信工程		信息网络	2	2020
论文提交日期*	2020.5.6			
导师姓名*	郭宇春		职称*	教授
评阅人	答辩委员会主席*		答辩委员会成员	
电子版论文提交格式 文本（ ） 图像（ ） 视频（ ） 音频（ ） 多媒体（ ） 其他（ ） 推荐格式：application/msword；application/pdf				
电子版论文出版（发布）者		电子版论文出版（发布）地		权限声明
论文总页数*				
共 33 项，其中带*为必填数据，为 21 项。				