分类号_____ 学号 M201571788

学校代码 10487 密级 _____

華中科技大學

硕士学位论文

用于行人检测数据增强的
生成对抗网络

学位申请人 ：欧阳熹

学 科 专 业 ：信息与通信工程

指 导 教 师 ：周潘 副教授

答 辩 日 期 ：2018 年 5 月 21 日

A Thesis Submitted for the Degree of Master of Engineering

The Generative Adversarial Network for Data Augmentation in Pedestrian Detection

Candidate : Xi Ouyang

Major : Information and Communication Engineering

Supervisor: Prof. Pan Zhou

Huazhong University of Science and Technology

Wuhan, P. R. China, 430074

May, 2018

独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期：2018年5月21日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密□，在_____年解密后适用本授权书。

本论文属于

不保密√。

（请在以上方框内打“√”）

学位论文作者签名：

日期：2018年5月21日

指导教师签名：

日期：2018年5月21日

摘要

目前最先进的一些行人检测的方法已经在很多的基准数据库中取得了令人印象深刻的成绩，但是这些方法在训练时需要大量的有标注的数据并且这些数据的标注过程往往需要耗费大量的人力和时间。在本论文中，提出了一种新颖的自动化方法去大量生成带有标注信息的行人数据，并且验证这些数据可以很好地直接给行人检测器进行训练。受到生成对抗网络（Generative Adversarial Network, GAN）近期成果的启发，本文提出的是一种可以在真实场景中生成逼真行人的生成对抗网络模型，并且生成的数据可以作为卷积神经网络（Convolutional Neural Network, CNN）结构的行人检测器的增强数据。相比于普通的对于图片生成非常有效果的生成对抗网络，本文设计的模型的实现目标是不一样的而且更具挑战性，原因在于：1、生成的行人需要去自然地与现实背景融合；2、为了给卷积神经网络结构的检测器进行训练，需要提供生成行人对应的位置信息。在本文中将提出的模型称为 Pedestrian-Synthesis-GAN，简称为 PS-GAN。

本文的方法是基于有多个判别器（Discriminator）的生成对抗网络。本文设计的这种新颖的生成对抗网络可以在图片上指定位置合成行人，并且可以很好地学习图片的背景信息。其中一个判别器可以使得 PS-GAN 去学习背景信息，例如道路结构、光照条件等等。这个判别器可以使得合成的行人自然地融合在背景中。另一个判别器可以使得 PS-GAN 学习到如何生成具有逼真外形和丰富细节的行人。与此同时，为了解决实际场景中行人的不同尺度问题，本方法还在生成对抗网络中加入了空间金字塔池化层（Spatial Pyramid Pooling, SPP）。

PS-GAN 是第一个把生成对抗网络用于行人或者是物体检测类任务的数据增强，本文在两个数据库中测试了提出的方法的效果，从视觉上看，本文提出的方法可以很好地在复杂背景的图片中合成逼真的行人。同时，本文实验中也把合成的数据用于训练行人检测器进行定量分析，对比在真实数据和加入合成数据的行人检测器的检测效

果，实验结果也可以很好地证明用本文提出的方法合成的数据可以提高行人检测器的效果。

关键词：行人检测，深度学习，生成对抗网络，空间金字塔池化

Abstract

State-of-the-art pedestrian detection models have achieved great success in many benchmarks. However, these models require a lot of annotation information and the labeling process usually takes lots of time and efforts. In this paper, we propose a method to generate labeled pedestrian data and adopt them to support the training of pedestrian detectors. Inspired by the recent success of Generative Adversarial Network (GAN), we propose to build a GAN-based model to generate realistic pedestrian images in real scene and utilize them as the augmented data to train the CNN-based (Convolutional Neural Network) pedestrian detector. Compared with adopting the regular GAN as a powerful tool for generating images, the goal of our model is different and more challenging due to: 1) generating pedestrians to fit the background scene well; 2) providing the corresponding locations of those synthetic pedestrians as the ground truths for the CNN-based detectors. We denominate it as Pedestrian-Synthesis-GAN (PS-GAN).

The proposed framework is built on the GAN model with multiple discriminators, trying to synthesize realistic pedestrian and learn the background context simultaneously. One discriminator aims to force PS-GAN to learn the background information like the road, light condition. It leads to smooth connection between the background and the synthetic pedestrian. The other makes PS-GAN to generate real pedestrians with more realistic shape and details. Moreover, to deal with the pedestrians of different sizes, we adopt the Spatial Pyramid Pooling (SPP) layer in the discriminator.

To the best of our knowledge, PS-GAN is the first work that utilizes GAN to generate data for pedestrian/object detection task. We execute experiments on two benchmarks. The results show that our framework can smoothly synthesize pedestrians on background images of variations and different levels of details. To quantitatively evaluate our approach,

we add the generated samples into training data of the baseline pedestrian detectors and show that the synthetic images are able to help improve the detector's performance.

Key words: Pedestrian Detection, Deep Learning, Generative Adversarial Network, Spatial Pyramid Pooling

目录

常见术语或缩略语解释：

英文简称	英文全称	中文全称
CNN	Convolutional Neural Networks	卷积神经网络
SIFT	Scale-Invariant Feature Transform	尺度不变特征转换
HOG	Histogram of Oriented Gradient	方向梯度直方图
SVM	Support Vector Machines	支持向量机
Faster R-CNN	Faster Regions with CNN features	快速卷积特征区域
ECCV	European Conference on Computer Vision	欧洲计算机视觉国际会议
CVPR	IEEE Conference on Computer Vision and Pattern Recognition	国际计算机视觉与模式识别会议
GAN	Generative Adversarial Network	生成对抗网络
RNN	Recurrent Neural Network	递归神经网络
RPN	Regional Proposal Networks	区域提出网络
DCGAN	Deep Convolutional Generative Adversarial Networks	深度卷积生成对抗网络
LSGAN	Least Squares Generative Adversarial Networks	最小二乘生成对抗网络
SPP	Spatial Pyramid Pooling	空间金字塔池化
VGG	Visual Geometry Group	视觉几何小组
GPU	Graphics Processing Unit	图形处理器
AP	Average Precision	平均精度

1. 绪论

1.1 研究背景和意义

在计算机视觉领域，行人检测是一个有着广泛应用的重要问题，应用领域包括并不限于：自动驾驶、视频监控和机器人设计[1]-[4]。行人检测目的是在开放场景的图片中识别并找到所有行人的位置，需要返回的是所有行人的边界框（Bounding Box）。因为在图像或者视频数据处理任务中，人的行动都是画面中的重点信息，所以设计算法来检测画面中的人一直以来都是很多计算机视觉领域研究者的重要研究方向。而深度学习（Deep Learning）的发展和各种大规模数据库的建立极大地推动了这个方面发展与进步。特别是得益于深度学习技术的发展，行人检测在实际的开放场景中已经取得了相当大的进展。近些年来与行人检测相关的应用在工业界已全面落地开花，应用领域相当广泛。因为很多工业界的应用首先就是需要检测人，比如智能视频监控，无论是对人体骨架分析、动作识别、身份再识别都需要首先进行行人检测来定位人所在区域。

然而，虽然行人检测已经被研究了很长时间，特别是近期的深度学习技术的发展，使得行人检测得到了极大的进步，但是行人检测仍然存在很多问题亟待解决。考虑到行人检测这种类型的物体检测任务由于行人是活动的物体，所以行人其实是同时包含刚体和非刚体的特征，这意味着行人会有一些自身的形体变化，然后同时行人的外形又容易受到衣着、大小、遮蔽、形态和位置的影响。这些问题都导致行人检测是物体检测领域一个有难度且急需解决的问题。同时行人检测的结果也很容易受到背景环境的影响，特别是在复杂的开放环境中，如何在各种环境中保证行人检测的精度也是一个很重要的问题。最近以来，基于深度学习中的卷积神经网络（Convolutional Neural Networks, CNNs）的行人检测方法，例如 Faster R-CNN[5]和 YOLO9000[6]，已经被广泛地在各种基准数据库中得到验证。当在大量的有标注训练样本的基础上训练时，这些模型相比与之前的传统方法都取得了显著的效果提升。

这些深度学习的方法本质上都是有监督的机器学习方法。有监督学习意味着模型在学习过程中需要有标签的数据，并且对于深度学习的算法而言，更是需要大量的数据进行对模型的训练。深度学习的算法一般只有经过大量数据训练之后才能达到很高的精度。但是，给行人检测标注数据是一个非常耗费人力和时间的任务。相比于给图片分类任务标注数据，给行人检测标注数据更为麻烦，需要给图片中每一个行人标注边界框。当有大量图片需要标注时，这是一个非常费时费力的工作，代价十分高昂。与此同时，以卷积神经网络为基础设计的行人检测模型又非常依赖于训练数据标注的质量和多样性。换而言之，当希望这些方法在测试时可以获得尽量好的表现，就希望在训练数据中能够尽可能地有和测试数据相似的场景和背景环境，例如相似的相机参数、相似的光照条件和背景。所以如何使这些行人检测模型运用在新的无标注的场景或者只有有限标注的场景时仍然保持非常高的准确率，这是一个对于研究人员来说很关注的问题，并且这个问题也有非常大的现实意义，例如当下人工智能应用最为关注的领域之一：自动驾驶。自动驾驶目前还没有哪个公司可以说他们实现了真正的 L4 级以上的完全无人驾驶，原因之一也是因为视觉感知模块中物体检测并不能保证实现和人一样在各种复杂条件和场景中都保持一个很高的水准，物体检测就包含行人检测，所以无法保证无人车在所有场景下的安全性。所以，设计一种方法能够能在仅仅依赖于有限的标注数据下训练，然后很方便地应用于无标注的新应用场景数据中并且能够达到一个较高的精度是一个非常重要的问题。

图 1.1　文献[7]提出的方法所得到的行人合成效果

解决这个问题的途径之一就是设计可以自动生成带有标注数据的方法。目前已经有研究人员提出了一些方法来实现在图片中生成行人并且给出这些合成的行人的位置框信息，例如在最近 Cheung 等人提出了一种新的方法可以在真实的环境中合成行人[7]，他们可以很好地在合成行人时考虑到真实环境中的背景信息。然而，因为他们是通过设计的三维渲染模型来生成行人，这些三维模型都是人工设计的，所以把这些行人合成到真实图像中时显得非常的假和不自然，如图 1.1 所示。

1.2 研究现状

因为研究的主题是给行人检测这一任务做数据增强，所以这部分本文将分成两个部分来讲国内外的研究现况及发展趋势：一是行人检测相关的进展，本文将会详细介绍近几年以来这个方向的代表性工作；二是给行人检测做数据增强的相关工作的介绍。本文将会分析这些工作中存在问题，以及本文的研究工作中如何针对性地解决这些问题。

1.2.1 行人检测领域的研究进展

行人检测从大类上来分可以分成两大类：一种是基于传统方法的行人检测；另一种是基于深度学习的行人检测，主要是基于卷积神经网络 CNN 的行人检测。

首先是基于传统方法的行人检测，传统方式来做也可以分成两种方式[8][9]:

（1）基于背景建模：

这种方式适用于视频中的行人检测，利用背景建模[10][11]的方法来分离图片中背景和前景信息，因为行人是一直运动的，这是背景中的前景信息。取出所有分离出的前景物体，然后在这些物体的目标区域内使用特征提取的方式比如 SIFT [12]、HOG [13] 特征。最后将这些提取的特征送入一个分类器中，比如支持向量机（SVM[14]）中，用这些分类器将前景物体分类，从而分类找到行人。这种方法的弊端就是需要有一个强大的背景建模的方法，对背景的建模可以把所有的运动的物体准确的分割和找出来，同时也要求分类器也要足够准确。目前的问题集中与：背景建模的方式必须能够对环境的变化有很好的鲁棒性（比如光照、抖动等等）；对图像中的复杂区域的适应性（比如很多物体相互交叠的区域）；及时对背景物体的改变做出反应，这是指一些物体可

能之前在运动然后突然停下，这时候这个物体要归于背景信息。同理，如果一个物体从静止到运动，就需要算法马上把它划为前景物体。所以这种方式只适用于运动的行人检测，而且这种方法实际中效果也不是很好。

（2）基于特征模板的统计机器学习方法：

这是传统方法来做行人检测最为常用的方法。这种方法是根据行人的特征信息来匹配行人[15][16]。首先使用滑动窗口的方法，在原图上用多尺度的窗口滑动然后用特征提取方式，最常用的特征是 HOG 特征，基于这些特征然后用分类器（SVM）进行分类找到行人。传统方法中大多都是应用 HOG 特征识别行人的整个身体，同时也有一些工作为了提高识别率，使用组合优化识别的检测算法来实现。这种算法是把行人分部分进行识别，首先通过腿部识别，再在腿部对应以上区域对肩膀至头部识别，从而提高识别精度。腿部由于走动会产生变化，可以用 HOG+SVM 的方式识别，然后肩膀和头部有一个比较固定的弧度形状，所以构造一个弧度的模板进行匹配。

但是这些传统的方法在开发环境中的行人识别效果都不是很理想，当引入深度学习之后，行人识别的准确率得到了很大的提升。深度学习在过去几年以来，在计算机视觉、语音识别和自然语音处理领域都取得了巨大的进展。特别是在计算机视觉领域，近几年来深度学习已经成为了这个领域最为广泛的方法，可以说是深度学习引领了这一次的人工智能浪潮。下面将重点介绍用深度学习的行人检测的方法。

深度学习在行人检测方面的应用主要是卷积神经网络（CNN）的使用。卷积神经网络一种由卷积层、池化层和全连接层堆叠而成一种多层神经网络结构，整个网络直接接受原始图像的输出然后直接得到分类结果[17]。整个网络通过反向传播进行端到端的学习。得益于 CNN 网络强大的表征能力，当有大量数据对 CNN 进行训练时，卷积神经网络为通用物体检测任务带来了全面的性能提升，著名的通用物体检测框架如 Faster R-CNN 和 YOLO9000。在近期的深度学习的方法中，行人检测领域相关的研究工作可以被总结成这三种类型：

第一类是将传统的检测方法与深度学习的方法相结合。有名的工作比如在 2016 年的欧洲计算机视觉大会（European Conference on Computer Vision, ECCV）上，中山大学林倞教授课题组使用区域预选网络（Region Proposal Network, RPN） 提取预选区

域（Proposal），在卷积神经网络的特征图中提取局部特征，然后使用提升树（Boosted Trees）在此基础上继续进行再次的分类，使得行人检测的性能得到了很大的提升[18]。

第二类是解决多尺度行人检测的问题，行人检测一个重要的问题就是如何保证对不同尺度的行人都保持较为鲁邦的检测效果。新加坡国立大学的颜水成老师团队[19]提出了一种同时考虑多种不同尺度行人的解决方案：其实就是同时训练两个网络，一个网络是为了去适应大尺度的行人，另一个则是为了小尺度的行人而设计，同时融合这两个网络的结果，这样可以使得网络同时对不同尺度的行人也有很好的鲁棒性，使得性能得到很大的提升。

第三类是加入图片的语义信息来辅助行人检测的效果。首先对整个图像进行语义信息的理解，其实就是使用语义分割的技术将图片的不同成分给分割出来，然后将分割得到的语义图作为一种先验信息的辅助输入到检测网络中，这样网络在进行行人检测的时候可以借助整体图片的语义信息，从而使得网络可以更好地理解整个图片的场景来提升行人检测的性能。例如在 2016 年的国际计算机视觉与模式识别大会（Conference on Computer Vision and Pattern Recognition, CVPR）的 C Daniel 等人[20]提出在做行人检测时引入语义分割图的信息，目的就是学习图片中的语义信息来帮助行人检测的效果。

1.2.2 行人检测数据增强的相关研究工作

目前，已有一些用于行人检测数据增强的方法被提出，下面介绍其中几种方法。

C Ernest 等人[21]提出一种可以生成任意数量有标注的人群视频的框架（LCrowdV），其生成的视频也可以用于训练人群场景理解的方法，其中就包括行人识别。在这个工作中，作者使用了大量的人工规则去控制行人的渲染。但是在这个方法也存在限制，他们的工作中使用的是虚拟的游戏引擎进行对行人的渲染，这导致这个工作不能准确地渲染出一些真实的户外环境。

H Hironori[22]等人提出一种用模拟环境数据训练行人检测器的思路，他们的考量是在一个全新的环境中，检测器只有非常少的这个场景中的数据可以训练，然后他们通过对这个场景进行几何建模渲染，然后加入渲染的不同外形的行人。通过这样的方式大量生成这种新场景下的数据，然后这样可以使得检测器在这一场景下的准确率超

过只用非常少的数据训练的检测器。但是这个工作明显的缺陷就是他们的方法只能提高检测器在单一场景下的准确率，因为他们使用了这一场景下大量的渲染数据来训练检测器，使得这样的检测器泛化能力并不好。

近期以来，最具代表性的工作之一也在本文的 1.1 小节也提到过，是 Cheung 等人提出的较为新颖的传统方法来给行人检测增强数据，作者使用一个三维模型渲染行人，然后通过投影模型将行人按照合适的比例放在图片上。但是这样的三维模型渲染的方式导致在图片上生成的行人看起来非常的不真实，所以用这样的数据去做行人检测的数据增强会导致模型学习到一些错误的数据分布。这篇文章的行人合成效果如图 1.1 所示，可以看到行人非常假，行人合成的效果图非常不自然，用这样的数据去训练检测器也会导致检测器学习到错误的行人检测规律。

从这些工作中可以认识到，如果仅仅是只用手工设计的规则去在复杂的真实场景中模拟行人，这是非常困难而且难以真正合成逼真的行人。于是，本文希望可以找到一种数据驱动的方法，即运用从真实的行人数据中学习如何去生成行人的方法来完成这个有挑战性的任务。

1.3 本文工作安排

受启发于近期以来生成对抗网络（Generative Adversarial Network, GAN）[23]在计算机视觉的多个领域中取得的惊人成果[24][25]，本文提出了一种基于生成对抗网络的方法，从而实现可以在真实场景图中合成逼真的行人，并且这些合成的数据可以作为增强的数据来训练以卷积神经网络为基础的行人检测模型。生成对抗网络设计的初衷就是去学习训练数据的真实分布，然后学习去生成类似的数据。相比于直接把生成对抗网络作为一种生成图片的有力工具，本文面临的问题是完全不同并且是更具挑战性的：1) 如何使得生成的行人能在真实图片中与背景自然地融合；2) 如何在生成图片的同时提供合成行人的边界框作为标签，如果没有标签的话将无法将生成的图片用于行人检测模型的训练。在本文中将设计的模型命名为 Pedestrian-Synthesis-GAN，在本论文接下来的内容中将把这个模型简称为 PS-GAN。本论文主要介绍提出的 PS-GAN 模型，围绕提出的模型进行研究和设计对比实验。具体的研究内容和组织安排如下：

第二章详细论述了生成对抗网络（GAN）的基本原理、应用领域以及与一种重要的 GAN 模型：Pix2Pix GAN[26]。本章首先介绍了 GAN 的由来和基本原理，包括 GAN 网络的训练策略，工作思想。然后详细介绍了 GAN 网络一些重要的改进工作，这些工作都是针对 GAN 的问题而设计提升的网络，并且本文最后提出的模型也使用了这些最新的技术。在本章的最后，介绍了将 GAN 运用于图像转换领域和一种重要的 GAN 网络的模型：Pix2Pix GAN，介绍了本文如何利用 Pix2Pix GAN 进行行人的生成，Pix2Pix GAN 将作为本论文提出的 PS-GAN 的对比模型，并以此引出本文如何针对行人生成任务改进得到 PS-GAN 模型；

第三章提出了一种新颖的生成对抗网络结构 PS-GAN 来完成行人数据的生成。首先该方法解决了如何用 GAN 网络来做给物体检测类任务生成数据的问题，因为物体检测类的任务不仅仅只是需要一张生成图像，同时也需要图像中生成物体的边界框标注信息。同时，该方法创新性地使用了多判别器的结构来同时解决行人合成的两个重要问题：1）与背景自然融合；2）生成逼真行人外形。多判别器的结构使得生成器相较于 Pix2Pix GAN 可以生成更为逼真和自然的图片；

第四章为了说明这个模型的有效性和鲁棒性，本文在两个大规模的数据库中评估了 PS-GAN 的性能：Cityscapes[27]和 Tsinghua-Daimler Cyclist Benchmark[28]。首先，运用提出的模型在 Cityscapes 数据库中都进行数据的生成，给出在这个数据中的视觉生成效果图，并且用真实和合成的数据来训练 Faster R-CNN 检测器，比较加入合成数据之后检测器效果的变化，以此证明这个模型用于数据增强的有效性。然后在 Tsinghua-Daimler Cyclist Benchmark 这个数据库中进行交叉数据集实验，即用在 Cityscapes 这个数据集中训练得到的模型直接在 Tsinghua-Daimler Cyclist Benchmark 这个数据集的图片中生成行人，同样地本文中也给出生成效果图，并且用真实和合成的数据来训练 Faster R-CNN 检测器，这个交叉数据集的实验是为了说明 PS-GAN 的泛化能力；

第五章是本论文的总结，对本文的研究方向和具体工作进行总结，对本文的贡献进行了详细总结，同时分析了本文提出的方法中还存在的问题和可能的未来研究方向。

2. 生成对抗网络 GAN 的原理与应用

本章主要对生成对抗网络 GAN 的基本原理和相关应用进行理论分析和说明。本章将会说明 GAN 网络的基本原理，与普通深度学习方法的区别，以及 GAN 网络近几年来的重要应用。此外，详细地介绍了 Pix2Pix GAN 模型，该模型的初衷是用于图片转换任务，本章将介绍如何将该网络用于行人检测任务，并且将这个模型作为本文提出的 PS-GAN 的重要对比模型。

2.1 生成对抗网络 GAN 的基本原理

生成对抗网络 GAN 是一种新兴的深度学习技术。深度学习在过去几年以来，在计算机视觉、语音识别和自然语音处理领域都取得了巨大的进展，甚至和强化学习结合之后也产生了惊人的效果[29]-[33]。特别是在计算机视觉领域，深度学习可以说目前最好的方法，并且在实际开放场景中深度学习仍然可以保证很好的效果，比如人脸识别[34][35]。这一次人工智能浪潮的兴起就是由深度学习而兴起的。而深度学习中最为广泛应用的两个网络就是卷积神经网络 CNN，和循环神经网络（Recurrent Neural Networks，RNN）[36]。CNN 在上文中已提到，是一种对图像输入非常有效的网络。而 RNN 是一种处理序列输入的网络。无论是 CNN 或者是 RNN，它们总体上的思想也是比较相似的：它们由具有可学习参数的神经元组成。每个神经元接收一些输入，执行一个点积，并跟随非线性单元。整个网络是端到端的学习：从一端的原始图像像素到另一端的分数。并且在最后一般都是全连接层，然后加上损失函数（例如 Softmax）。不同在于，CNN 网络充分利用了输入由图像组成的事实，并以更合理的方式约束了体系结构。而 RNN 网络则由于是序列数据输入，于是使用了时序模型来记忆历史状态。

虽然以 CNN 和 RNN 为核心的深度学习技术近年来取得了巨大的成果，推动了当前人工智能的大发展。但是这些方法从本质上来说，都是有监督的机器学习方法，它们相较于传统的机器学习方法而言之所以可以取得如此大的成功，原因在于它们强大的数据表征能力和端到端的学习方式。在有大规模的标注数据训练的时候，深度学习

网络强大的特征表征能力可以使得网络很好地学习到数据规律。同时端到端的学习方式使得整个网络的参数都是向着同一个方向进行优化，这也进一步提升了网络的学习效率。

有监督的机器学习方法一个最大的问题就在于需要大量有标注的数据来对网络进行训练。特别是对于深度学习的方法而言，CNN 和 RNN 的深度学习方法都是数据驱动的学习方式，并且一般深度学习网络参数众多更是需要大量数据来对网络进行训练。在当下的互联网社会中，大数据是易于收集的，但是只有数据是无法训练模型的，还需要人工标注数据。例如一个简单的分类任务-区分猫和狗，就需要收集大量猫和狗的图片，然后人工标注每一张图片是猫或者狗，这样的数据才能给网络训练。人工标注大量数据需要耗费大量的人力和时间。所以需要寻找无监督学习的方式来缓解这个问题。

从理论的角度来看，深度学习网络的目的就是各种人工智能场景的数据中（例如自然图像、语音和自然语言）找到一个复杂和深度的模型去表征这些数据分布[37]。在大部分的相关领域的工作中，取得重大进展的深度学习网络都使用的是有监督学习的判别模型（Discriminative Models），通常这些判别模型就是用来将高维的、丰富层次的数据映射成一个类别的标签[38][39]。这些工作的成功也主要得益于反向传播（Backpropagation），随机丢弃（Dropout）算法，和对梯度行为有很好约束的分段线性单元（Piecewise Linear Units）[40]-[42]。相比之下，由于难以模拟真实数据复杂的概率分布和无法很好地在生成模型中运用分段线性单元，无监督学习的深度生成模型（Generative Models）则没有太多令人印象深刻的成果。

但是近年来深度学习领域出现了一种新的无监督学习网络-生成对抗网络。生成对抗网络 GAN 是由 Goodfellow 在 2014 提出来的一种无监督学习网络。GAN 是一种很特别的网络结构，同时利用了生成模型和判别模型：生成器 G（Generator）和判别器 D（Discriminator）。GAN 的原理其实非常简单，简而言之：生成器 G 学习如何从一个输入噪声生成一张图像，而判别器 D 学习判断生成的图像是真实的图片还是由生成器 G 生成的“假的图像”。便于理解，可以想象这样一个场景：警察和印假钞的罪犯，警察会不断学习从而提高警察对假钞的识别能力，同时印假钞的罪犯也会不断提高印

假钞的工艺从而让警察无法辨认假钞。这个过程会反复进行，直到警察也无法区分真钞和假钞的区别。这个过程就可以对应到 GAN 里面生成器 G 和判别器 D 的博弈过程，这其实是一个对抗学习过程，生成器 G 就是印假钞的罪犯，判别器 D 就是警察。GAN 网络中运用了在深度学习中取得巨大成功的 Backpropagation 和 Dropout[43]去训练网络，在使用时仅仅用前向传播(Forward Propagation)算法就可以让生成模型 G 从噪声输入生成一个新的样本。整个过程并不需要任何近似算法或者马尔科夫链模型。

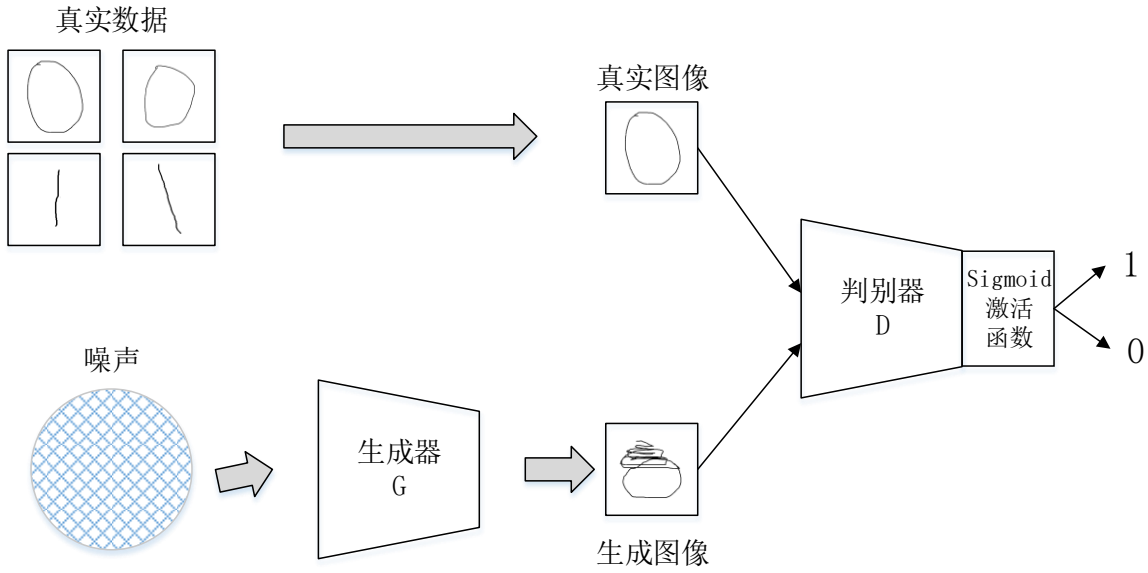

图 2.1　GAN 网络的原理图

生成器 G 学习如何从一个输入噪声生成一张图像，而判别器 D 学习判断生成的图像是真实的图片还是由生成器 G 生成的“假的图像”，如图 2.1 所示。这两个模型在训练过程中交替训练、对抗学习，从而互相促进增强。判别器 D 在训练中能力得到增强从而更清楚地分辨真实图片和假的图片，生成器 G 在训练中也得到加强得到非常类似真实图的生成图，可以让判别器 D 无法分辨真实或者生成图。下面将从数学原理上详细解释 GAN 网络的对抗学习过程。

为了学习生成器在真实数据 x 中的概率分布 p_g，本文定义给生成器的输入噪声分布为 $p_z(z)$，然后用 $G(z;\theta_g)$ 表示生成器从输入的噪声空间到真实数据空间的映射，其中 G 表示一个网络参数为 θ_g 的可微分函数（即生成器网络）。同时定义另一个网络模型 $D(x;\theta_d)$，这个网络输出的是一个标签值（即判别器网络）。$D(x)$ 代表从真实数

据 x 的概率判别结果。训练整个网络时，训练 D 去最大化给真实数据和从 G 生成的数据分类正确的标签的概率，同时训练 G 去最小化 $\log(1-D(G(z)))$。换而言之，D 和 G 是在进行两个人的博弈过程，价值函数 $V(D,G)$ 如下所示：

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p_z(z)}[\log(1-D(G(z)))]. \quad (2.1)$$

为了更形象地说明这个对抗学习过程，将训练过程的概率分布变化过程展示在图 2.2 中。GAN 的训练过程中同时更新判别和对抗模型的分布，判别模型分布 D（图 2.2 中的蓝色密点线）需要能够可以判别是从真实数据分布 p_x（图 2.2 中的黑色粗点线）中采样得到的样本还是从生成器生成的数据中 $p_g(G)$（图 2.2 中的绿色实线）中采样得到。在图 2.2 中的最下面的横线表示的是噪声 z 的一个采样的分布域，在真实情况中被认为是正态分布。中间那条横线是代表真实数据样本 x 的一部分分布空间。其中向上的箭头表示映射函数 $x=G(z)$ 如何去使得生成器原本非正态的概率分布 p_g 去拟合从噪声 z 转化到真实数据 x 的过程。为了拟合这个概率分布，G 在 p_g 高概率密度的区域收缩并且在低概率密度的地方扩张。从左向右依次描述图 2.2 中的图，第一张图：表示训练初始状态， p_g 的分布和 p_{data} 类似，D 也是一个相对较准的分类器；第二张：不更新 G，先对 D 迭代训练几次，D 就可以去分辨是不是真实数据，D 的网络参数收敛到 $D^*(x)=\frac{p_{data}(x)}{p_{data}(x)+p_g(x)}$；第三张图：固定 D，对 G 进行更新，从 D 传过的梯度将引导 $G(z)$ 去生成能被误分类成真实数据的采样；第四张图：在 D 和 G 交替训练一段时间之后，如果 G 和 D 的网络有足够的特征表达能力，它们将达到一个双方都无法再继续促进提高的平衡点 $p_g = p_{data}$。此时判别器 D 也无法分辨是真实数据还是生成数据，对真实数据判别的概率将成为 $D(x)=\frac{1}{2}$。

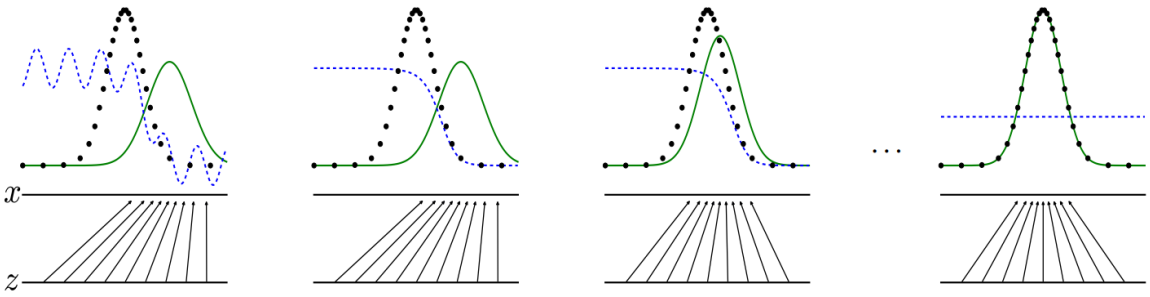

图 2.2　训练过程的概率分布变化过程

2.2 GAN 的改进和应用

GAN 的提出吸引了一大批研究者的注意，得益于它的巧妙的构思和不错的效果。GAN 之后出现了很多相关的优秀的研究工作，这些工作集中与两个方向，一是解决 GAN 的训练的稳定性和生成图片质量的问题，二是将 GAN 运用在其他很多领域而不仅仅只是用来生成图片。这一节将从这两个方面介绍 GAN。

2.2.1 针对 GAN 网络提出的改进

本节将重点讲 2 个重要的针对 GAN 训练稳定性而提升的改进：Deep Convolutional Generative Adversarial Networks (DCGAN)[44]和 Least Squares Generative Adversarial Networks（LSGAN）[45]。这两个 GAN 的变种模型正好从两个方面来提升 GAN 训练稳定性和生成图片质量：DCGAN 是提出更好的网络模型来训练 GAN 网络，而 LSGAN 则是针对 GAN 网络中损失函数（Loss Function）存在的先天缺陷来改良 GAN 的损失函数。并且这两种技术在本文提出的 PS-GAN 同样有所应用。

首先介绍 DCGAN：DCGAN 解决的核心问题是找到了一个对于 GAN 训练非常有效的卷积神经网络结构。原始的 GAN[22]网络使用的网络是多层感知机结构。多层感知机网络简而言之就是一种全连接的结构，即每一层的神经元都与下一层的神经元相连。这样的结构对于图片的输入并不友好，耗费的参数众多，并且并不高效，使得其只能处理一些简单的图片。现在将原始 GAN 的生成效果图展示在图 2.3 中，在图 2.3 的 a、b、c、d 中最右边的黄色框中的图像是生成图像，而左边 5 列都是与生成图非常类似的真实图像，其实这些图像都是相对简单的，而且生成效果并不是非常好，特别是在 c 和 d 的复杂图像中。

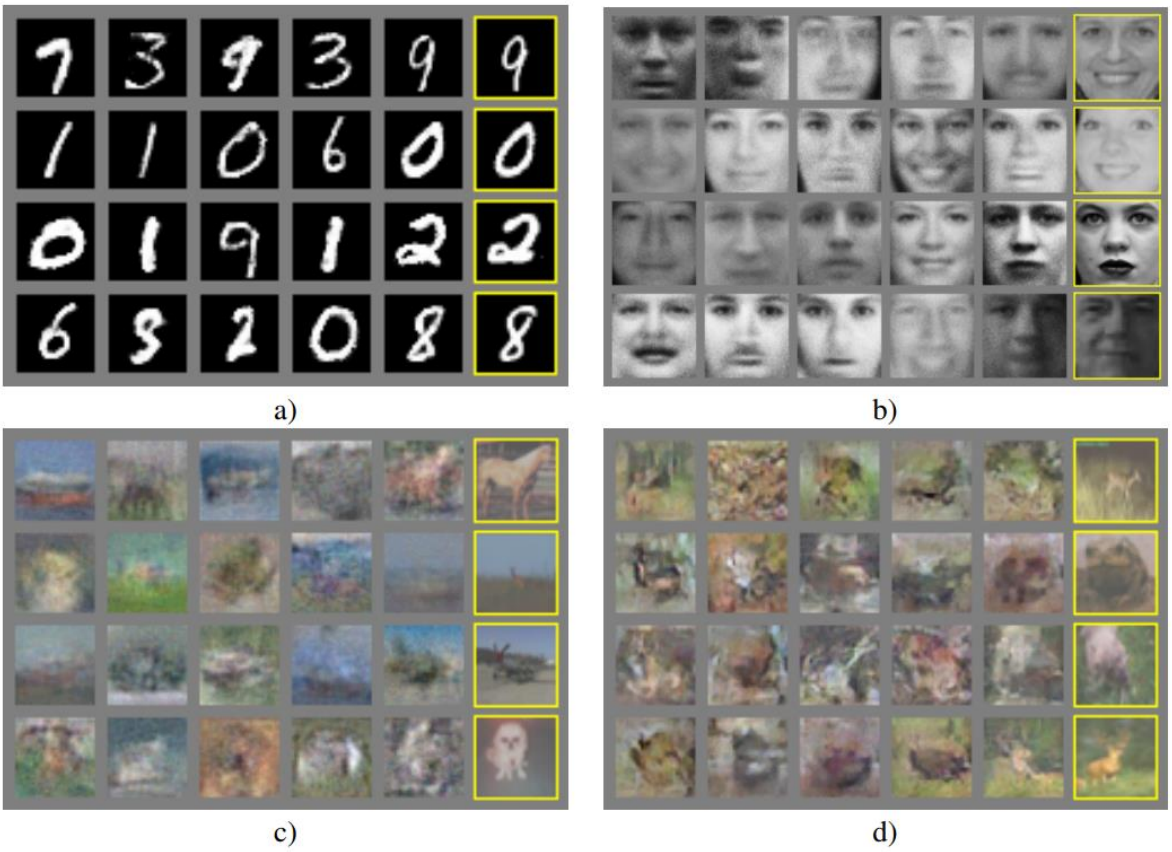

图 2.3　原始的 GAN[22]的生成图片效果图

而DCGAN就是针对生成图片这个任务，引入了卷积神经网络来替换多层感知机。卷积神经网络已经被许多研究者证明对于图片输入而言是非常适合的一种网络结构。但是对于 GAN 网络来说，由于是一个对抗学习过程，所以其实整个学习过程并不稳定，这是由于损失函数（公式2.1）的先天缺陷导致的，下面的章节中也会讲到在LSGAN中如何改良损失函数。在 DCGAN 中，作者通过大量的实验找到了一个由卷积神经网络搭建的较为稳定的生成器 G 结构，实验证明这样的结构可以生成质量较高的图片。DCGAN 的生成器网络结构图如图 2.4 所示，同时 DCGAN 的生成图片效果图如图 2.5 所示。这些图片是 DCGAN 在 LSUN[46]数据集的卧室场景图片训练之后的生成效果，可以看到即使在如此复杂的场景图片下，DCGAN 依然可以生成质量非常高的图片。

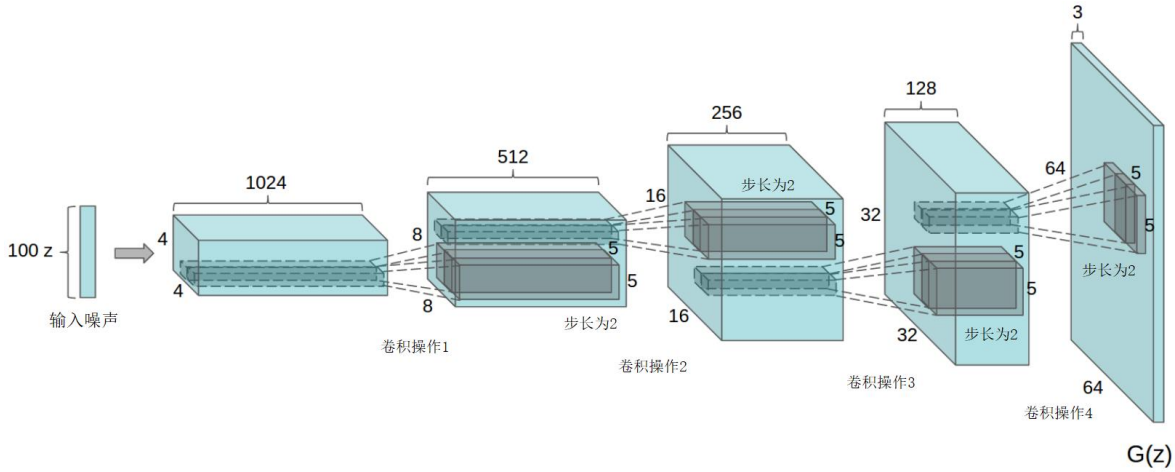

图 2.4　DCGAN 的生成器网络图

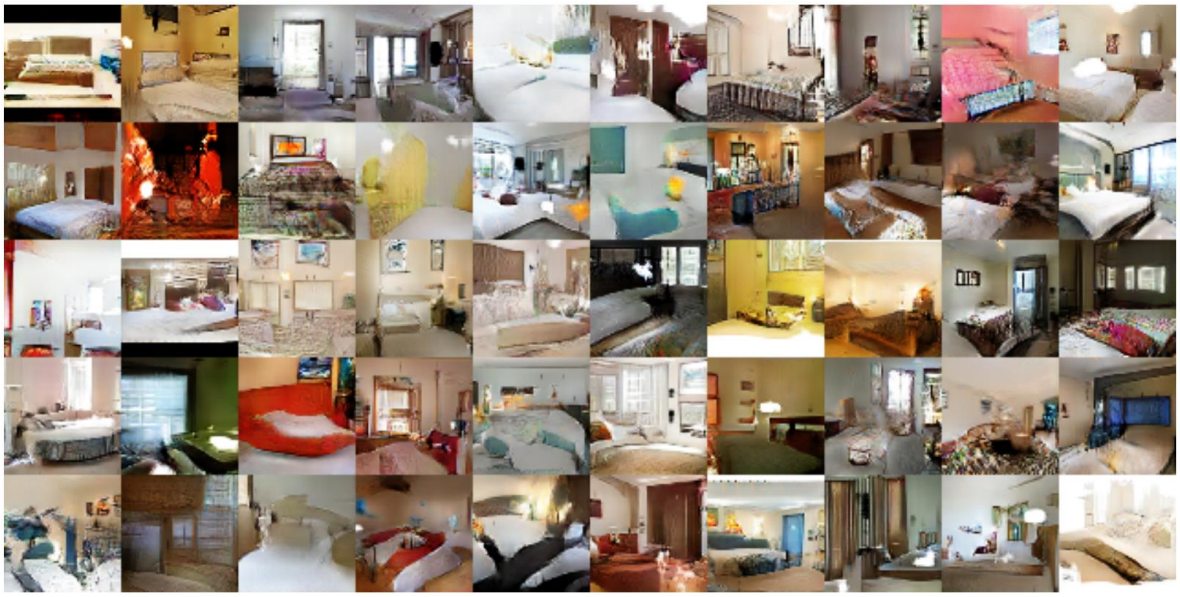

图 2.5　DCGAN 的生成图片效果图

另一种思路去提升 GAN 生成图片效果的方法是 LSGAN。相比于 DCGAN，LSGAN 并不专注于寻找一个高效的网络结构而是直接从数学分析的角度出发，去修改原始 GAN 的网络的损失函数。首先解释下为什么原始 GAN 网络的损失函数是存在问题的。原始 GAN 网络的损失函数已经展示在公式 2.1 中，对抗学习过程展示在图 2.1 中，可以看到原始 GAN 的损失函数其实就是一个 Sigmoid 交叉熵函数。这种交叉熵函数存在的问题在于，当生成器生产的样本在分类决策的边界上时，整个模型就会遭受梯度消失的问题，从而这些样本将不能被拉向真实的数据分布。因为边界上的数据已经让判别器近乎相信这是正确的数据，Sigmoid 交叉熵对这些样本的损失函数值非常小从而使得这些数据很难被正确训练，但是这些数据其实离真实数据还有一定的距离。基

于此，使用最小二乘损失函数的生成对抗网络（Least Squares Generative Adversarial Networks，LSGAN）[45]被提出来，最小二乘的损失函数如下所示：

$$\min_D V(D) = \min_D \frac{1}{2} E_{x\sim p_{data}(x)}[D(x)-a]^2 + \frac{1}{2} E_{z\sim p_z(z)}[D(G(z))-b]^2, \quad (2.2)$$

$$\min_G V(G) = \min_G \frac{1}{2} E_{z\sim p_z(z)}[D(G(z))-c)]^2. \quad (2.3)$$

其中符合的含义与原始 GAN 的损失函数公式 2-1 中是一致的。在这两个公式中，真实图片和生成图片分别用常数 a、b 来标志，同时常数 c 标示的是判别器判断生成图片是真实数据的难易程度，一般 a 和 c 的值是相等的。在 LSGAN 的文献[45]中，作者将这些值设为：$a=c=1, b=0$。最小二乘的损失函数，因为计算的是样本与边界距离的平方值，所以这个损失函数即使对在决策边界上的样本也有很大惩罚，使得这些样本也可以被正确分类。

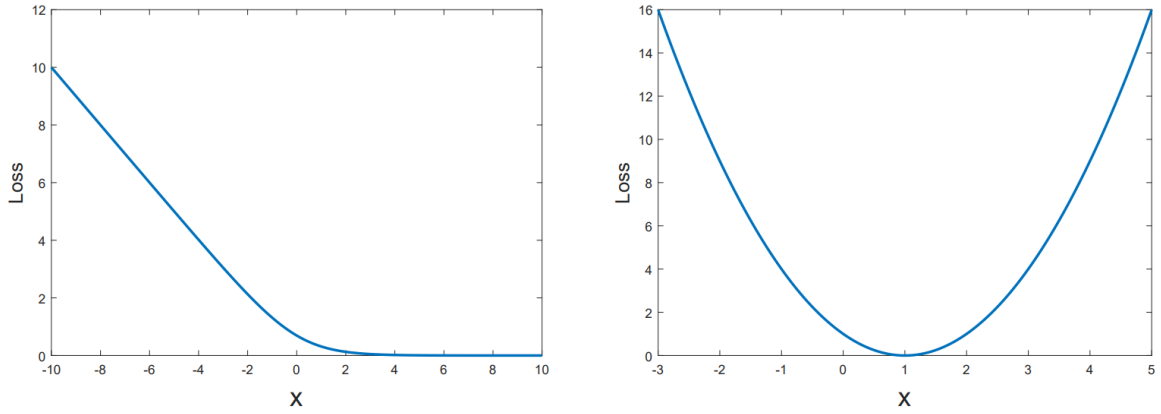

图 2.6　原始 GAN 的 Sigmoid 交叉熵损失函数和 LSGAN 的最小二乘损失函数

LSGAN 的最小二乘损失和原始 GAN 的 Sigmoid 交叉熵损失函数也可以从概率的角度来说明区别。将 Sigmoid 交叉熵和最小二乘的损失函数分别画图表示在图 2.6 中，图中左边的曲线图是 Sigmoid 交叉熵损失，右图是 LSGAN 的最小二乘损失。在图 2.6 中，可以清晰地看到，Sigmoid 函数有很大一段中曲线都是接近水平即梯度接近于 0，特别是当损失接近于 0 时，曲线的梯度越小。损失接近于 0 时就是模型接近于收敛到最好时，但是这时候梯度将变得非常小并逐渐消失为 0，这使得模型很难优化到最好，这也就是原始 GAN 训练不稳定的原因所在；而最小二乘损失因为是二次函数所以曲线就是一条抛物线，其曲线只在一个地方取值为 0，并且两边梯度都是朝向这个最优

点的。这就使得 LSGAN 训练更为稳定，可以比较好的训练到最优点。LSGAN 的生成图片效果图如图 2.7 中所示，同样的，这些图片也是 LSGAN 在 LSUN[46]数据集的卧室场景图片训练之后的生成效果，相比于 DCGAN 的结果，LSGAN 的结果多样性和细节都会更好。

图 2.7　LSGAN 的生成图片效果图

在本文为行人检测设计的 PS-GAN 中也使用了 DCGAN 和 LSGAN 中的技术，将会在第三章中详细说明。

2.2.2 GAN 网络的应用

由于 GAN 网络的无监督学习特性，有很多研究者致力于将 GAN 的思想应用到其他领域而不仅仅是图片生成，比如推特的研究者就将 GAN 应用于解决图片超分辨的问题，提出了一个新的 Super Resolution GAN（SRGAN）[25]，这篇文章使用 SRGAN 做出了相当惊人的效果。近期最为火热的 GAN 应该是加州伯克利大学研究团队提出来的 Pix2Pix GAN,Cycle GAN[47]和 Bicycle GAN[48]，这三个 GAN 是用来做图像转换（Image Translation）任务的，同样也是使用了 GAN 的思想之后效果非常好。Pix2Pix GAN 是最早的工作，做的是对应的两种图片之间的相互转换。 而 Cycle GAN 和 Bicycle GAN 是 Pix2Pix GAN 的后续工作，Cycle GAN 可以在训练时不需要配对的数据就可以训练得到两个图片域之间转换的效果，Bicycle GAN 则是拓展到了多个图片域的之间的相互转换。本节将重点介绍 Pix2Pix GAN 的原理，因为本文提出的 PS-GAN 就是基于 Pix2Pix GAN 来进行针对性的改进。

首先解释图片转换任务其实就是从一张图片翻译到另一张图片例如同一个地点从白天转换到黑夜的效果，从手绘图转换为实物的效果。所以这就使得普通的 GAN 模型无法完成这个任务，因为普通的 GAN 只能从一个噪声向量生成图片，而不能从

图片转换到图片。Pix2Pix GAN 训练时需要匹配好的对应两个图片域中的图片来训练，从而学习从其中一个图片域到另一个图片域。由于 Pix2Pix GAN 实现的是图片的转换，从而输入不再是一个噪声而是一张图片，因此生成器 G 和判别器 D 的设计都不同于原始的 GAN 网络，判别器 D 的设计也是这个工作中最具创造力的地方，其整体网络设计图展示在图 2.8 中。Pix2Pix GAN 中生成器 G 是一种编码器（Encoder）和解码器（Decoder）的结构实现了从图片到图片的转换过程。然后判别器 D 采用了配对验证的方式，不是验证生成的图是否是真实图，而是判断生成图和输入图是否是一个真实的图片对，如 2.8 所示，判别器 D 的输入是两张图片堆叠的图片对（Image Pair）。

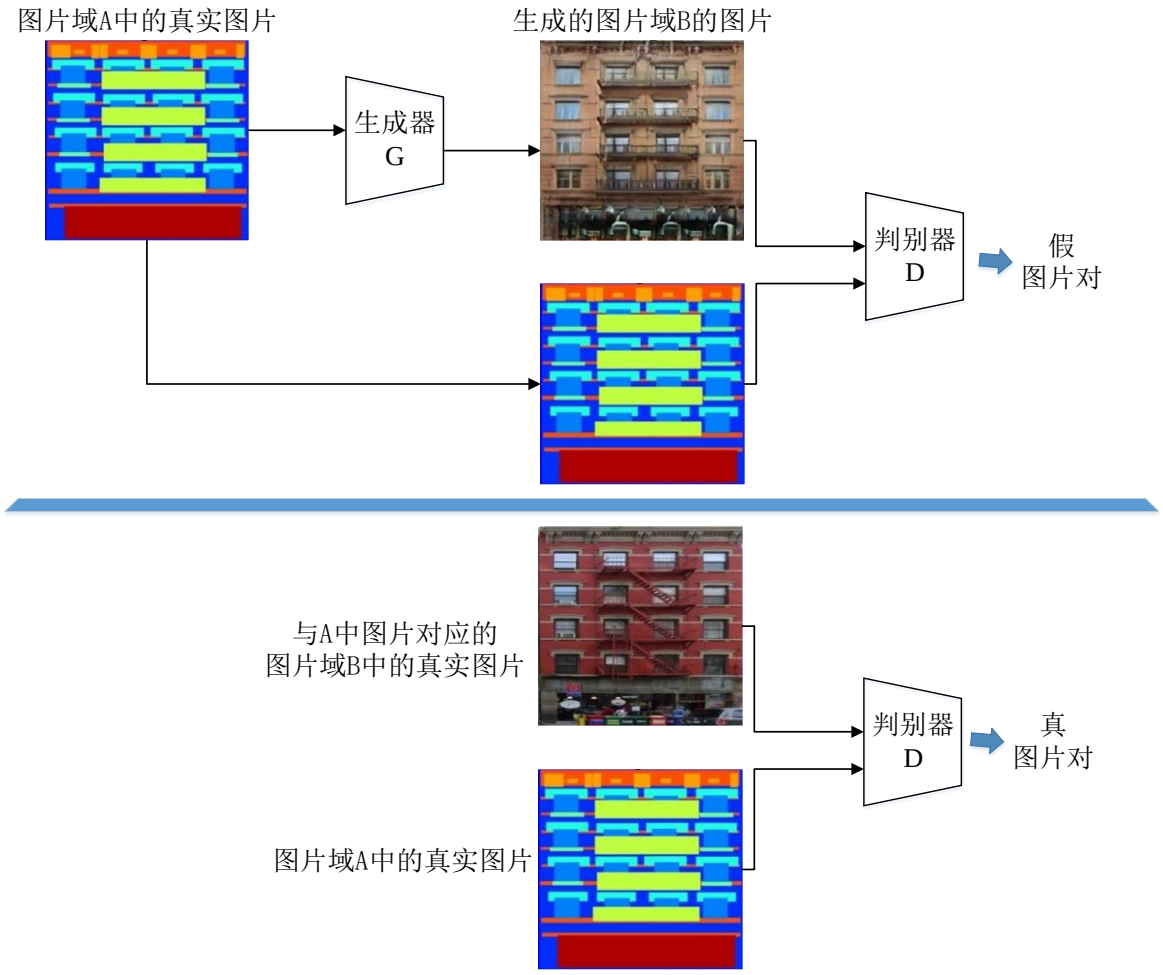

图 2.8　Pix2Pix GAN 的网络结构图

Pix2Pix GAN 的图片风格转换效果图展示在图 2.9 中，这里展示的是从手绘画生成实物图，其中展示了左边三列为一组，由左往右分别为输入，真实图和生成图，同样右边三列也是这样的。

图 2.9　Pix2Pix GAN 的图片转换效果图

2.3 Pix2Pix GAN 用于行人检测的数据增强

受启发于 GAN 网络优异的无监督学习生成图片的能力，于是把 GAN 引入来给行人检测增强数据是一个非常具有研究价值的方向。但是目前还没有任何工作把 GAN 成功引入了给物体检测来做数据增强。其中最具挑战的问题在于生成的数据不仅仅是需要生成一张图片同时也需要给出这张图片中行人位置的边界框来做训练标签。训练时的难点是如何让 GAN 学会在图片指定的边界框内合成行人，普通的 GAN 只能生成一整张图片，这里需要 GAN 能在限定地在指定区域生成一个行人同时得到这个行人的位置信息，并且在生成这个行人时还需要考虑到周围环境信息，使得这个行人的出现要符合背景环境信息。例如符合周围的光照、明暗、对比度等等。

在本论文中，创新性地用图像转换的方式解决了 GAN 在图片的指定位置中生成行人的任务。整体的思路如图 2.10 中所示，其中生成器 G 和判别器 D 都是 Pix2Pix GAN 中的结构。这个思路的创新之处在于，在训练时，将行人的边界框中用噪声框覆盖。这样一来，就可以得到一个图片对-原始有行人的图像和行人被覆盖了噪声框的图像。

如此一来，这个任务可以做成图片转换的任务，模型需要学会从噪声图转换到有真实行人的图像。这样一来，生成器生成图像可以作为训练数据的图像，然后噪声框的位置就可以作为生成行人的边界框位置信息，同时噪声框是可以控制随便加在图片中任意位置，这也意味着可以在图片任意位置合成行人。通过这样的方式，巧妙地解决了在生成图像的同时得到行人的边界框信息的问题。从检索的结果来看，这也是第一个用 GAN 的方式来给物体检测器做数据增强的方法。

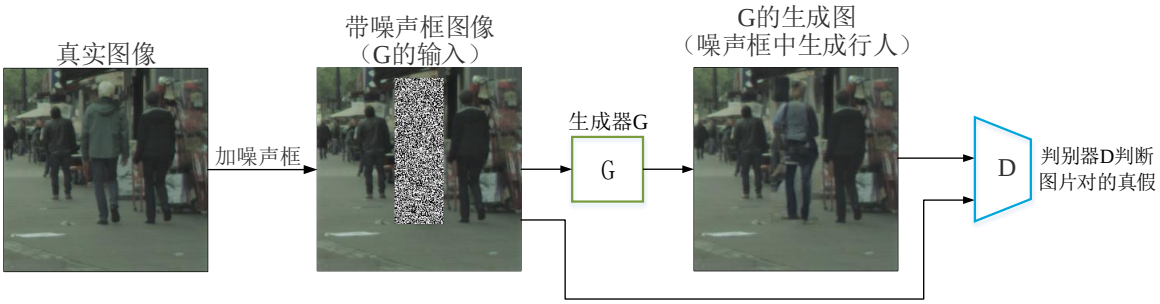

图 2.10　用 Pix2Pix GAN 做行人检测数据

虽然这样可以达到在图片中生成行人并且得到行人边界框的目的，但是这样只是借用了 Pix2Pix GAN 的方法而这个网络并不是设计用来完成这个任务。所以这样合成的行人效果并不是特别好，具体的行人合成效果将在本论文的第四章中详细展示。沿着这个思路，本文专门为了行人检测数据增强这一任务提出了 Pedestrian-Synthesis-GAN（PS-GAN）这一生成对抗网络模型框架，创新性地使用了多判别器的结构，使得 PS-GAN 的行人合成效果相比于 Pix2Pix GAN 来说有一个巨大的提升，而 Pix2Pix GAN 将作为提出的 PS-GAN 的对比模型，成为实验中的一个基础对比实验。

2.4 本章小结

本章首先介绍了 GAN 的基本原理、改进模型和一些应用方向。相比于其他的深度学习方法，GAN 是一种新颖的无监督学习网络，可以用来学习训练数据的分布。GAN 的训练过程，是一个对抗学习过程：生成器 G 和判别器 D 的对抗博弈。原始的 GAN 网络使用的是 Sigmoid 交叉熵损失来优化网络参数。由于原始的 GAN 网络存在生成效果不尽如人意和训练不稳定的情况，所以有研究者针对 GAN 网络的这些问题

提出一些改进模型：DCGAN 和 LSGAN。DCGAN 是针对 GAN 的网络进行重新设计，找到了一种高效的生成器和判别器的网络结构，这种结构可以极大地促进 GAN 的生成图片的效果。而 LSGAN 则是用最小二乘的损失替代了原始的 Sigmoid 交叉熵损失，使得 GAN 的训练更稳定。

同时介绍了将 GAN 应用于图片转换任务的 Pix2Pix GAN 模型，该模型在图片转换任务中取得了惊人的效果。本文也提出了一种新颖的将 Pix2Pix GAN 用于行人检测数据增强的方案，这也是第一次将 GAN 应用于物体检测类任务的数据增强。并且，由此在下一章中引出本文提出的 PS-GAN 模型。

3. 用于行人检测数据增强的 PS-GAN

在第一章的相关工作中介绍了使用传统方法来给行人检测做数据增强的方法，但是如图 1.1 中所示，其合成的行人非常不自然。由于仅仅靠复杂的人工规则来合成行人几乎是不可行的，因为现实环境存在太多的条件。本文在第二章中也介绍了引入 GAN 做数据增强：将 Pix2Pix GAN 用于行人检测的数据增强。但是这样合成的效果是无法满足实际需要的，于是本文提出了 Pedestrian-Synthesis-GAN（PS-GAN）。

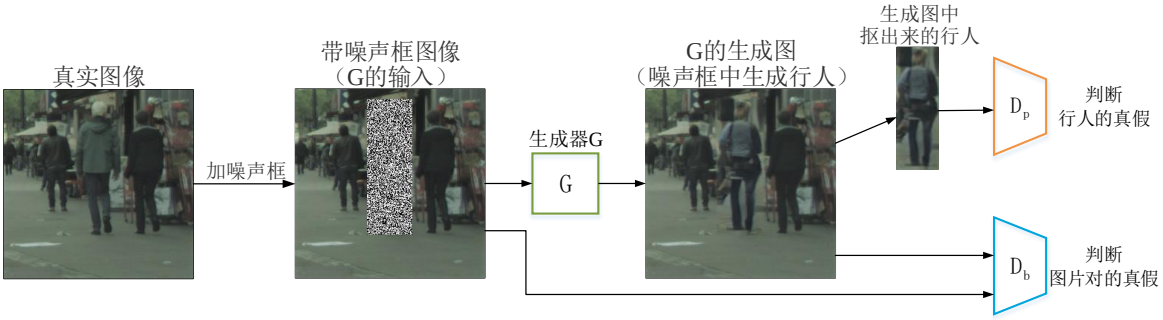

图 3.1　PS-GAN 模型结构的设计示意图

本文提出的模型 PS-GAN 应用了对抗学习的思想，并且创新地包含了多个判别器：判别器 D_b 用于学习背景的语义信息，同时判别器 D_p 用于鉴别合成的行人是否真实。整体的结构设计展示与图 3.1 中。首先，需要用带有行人边界框标注的数据来训练 PS-GAN，使其学会在图片中合成行人。实现方式是：用随机噪声框覆盖整个行人边界框，用这个带噪声框的图片当作训练数据（图 3.1 中的第二张图片），原始带有行人的图片（图 3.1 中的第一张图片）作为对应的标签来训练 PS-GAN，从而使得生成器 G 学会在噪声框中生成行人。判别器 D_b，接收一对图片作为输入，从而学会区分是真实图片和噪声图的图片对或者是生成图和噪声图的图片对（Image Pair）。同时，判别器 D_p，学习去分辨从边界框中抠出的行人图是真实图或者是生成的行人。这样，通过生成器 G、判别器 D_b、判别器 D_p 这三方的对抗学习，判别器 D_b 可以强迫生成器 G 去学习背景信息的知识，例如道路情况、光照条件等等，它使得合成的行人与背景“合二为一”。同时判别器 D_p 可以使得生成器 G 能够在噪声框中生成有着逼真外形和丰富细节的行人，达到“以假乱真”的效果。更进一步地，由于在图片中抠出的行人图片有着各式各样的尺度，本文提出的模型还在判别器 D_p 中加入了空间金字

塔池化层（Spatial Pyramid Pooling，SPP）[49]，由此尽量地减少不同尺度对网络带来的噪声。在这些训练之后，生成器 G 就可以学会在噪声框中生成非常逼真的行人，同时噪声框的位置就作为这些合成图片的边界款标注信息。如此，通过 PS-GAN，就可以得到有行人位置标注的合成数据来给行人检测模型的训练做数据增强。

本章详细介绍 PS-GAN 网络的结构，包括生成器和两个判别器的结构，同时介绍本文在 PS-GAN 中使用的损失函数。

3.1 生成器 G 的网络结构

生成器 G 的目的是去学习从输入到输出的映射函数：$G: x \to y$，其中 x 是输入的噪声图而 y 是真实行人图像。在本文提出的方法中，借鉴了在文献[50]中使用的一种增强的编码-解码器网络（Encoder-Decoder Network）。该网络被称为 U-Net，本文使用这种方法作为生成器的网络结构。该网络的结构图展示在图 3.2 中。

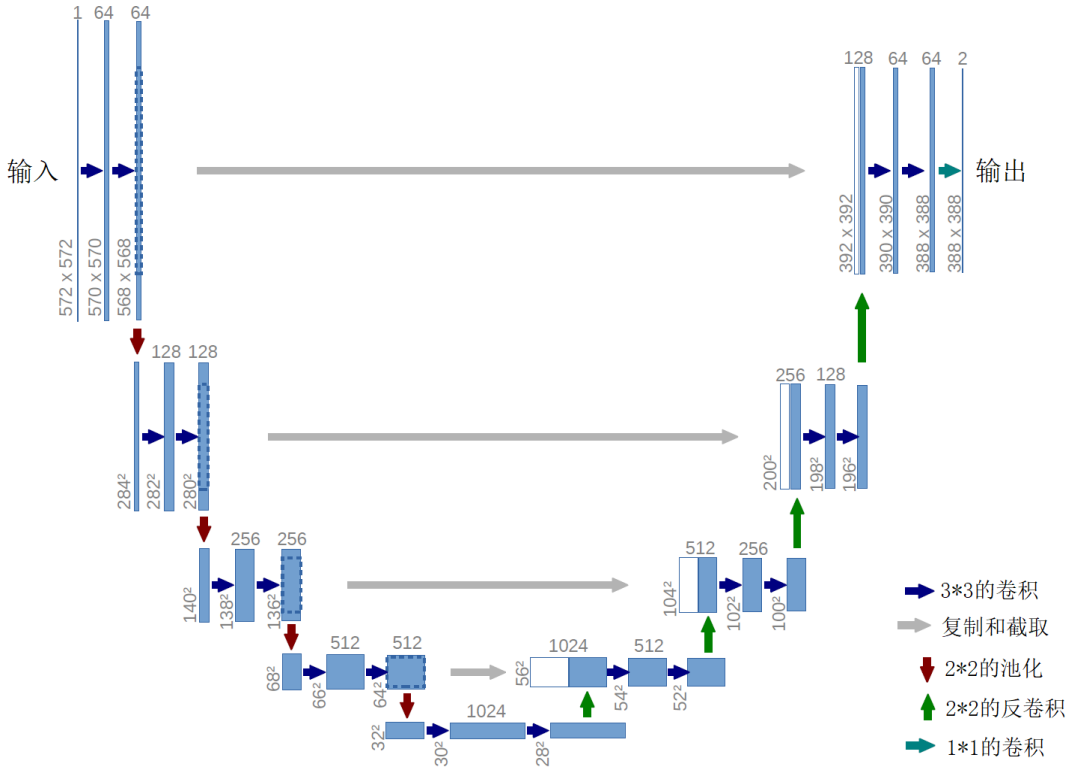

图 3.2　U-Net 结构的示意图

由图 3.2 所示，该网络保持了编码器-解码器网络的基本架构：即由一个编码器和一个解码器组成，输入和输出的图像保持一样的尺寸大小。正如图 3.2 所示，左半部

分就是一个编码器，主要由卷积、池化和激活层构成，主要功能就是将输入图像编码成一个 1024 维的特征向量；右半部分则是解码器网络，由反卷积、卷积和激活层构成，主要功能是从 1024 维的特征向量解码得到一个新的图像。两个网络通过中间的 1024 维特征向量连接，这样可以完成从一个图片域到另一个图像域的转换，所以编码-解码器网络被广泛地用于各种图像域转换的任务，例如语义分割、图像重建等等。而 U-Net 是一种增强型的编码-解码器网络，主要的不同在图 3.2 中也可以清晰地看到，编码和解码网络之间除了中间的特征向量连接以外，在编码和解码器对称的中间的网络隐层中也有很多侧连接（Side Connection）。在图像转换中，不同于图片分类的任务，图片分类更多地是去关心图片整体信息，但是图片转换中，图片局部信息也非常重要。一般的编码器-解码器网络只有中间的特征向量相连，而在大量卷积和池化操作之后，图片很多局部信息都已经丢失，而 U-Net 充分考虑了这些信息，从而可以实现非常好的图片转换效果。

3.2 判别器 D_p 的网络结构

对于判别器 D_p 的训练，从生成的图像中扣出来的行人作为负样本，同时从真实图 y 中的抠出来的真实行人 y_p 作为正样本。如此，D_p 训练去学习分辨在噪声框中的行人是否真实。这样在整个网络对抗学习中，D_p 可以强迫生成器 G 学会在噪声框 z 中生成一个逼真的行人。

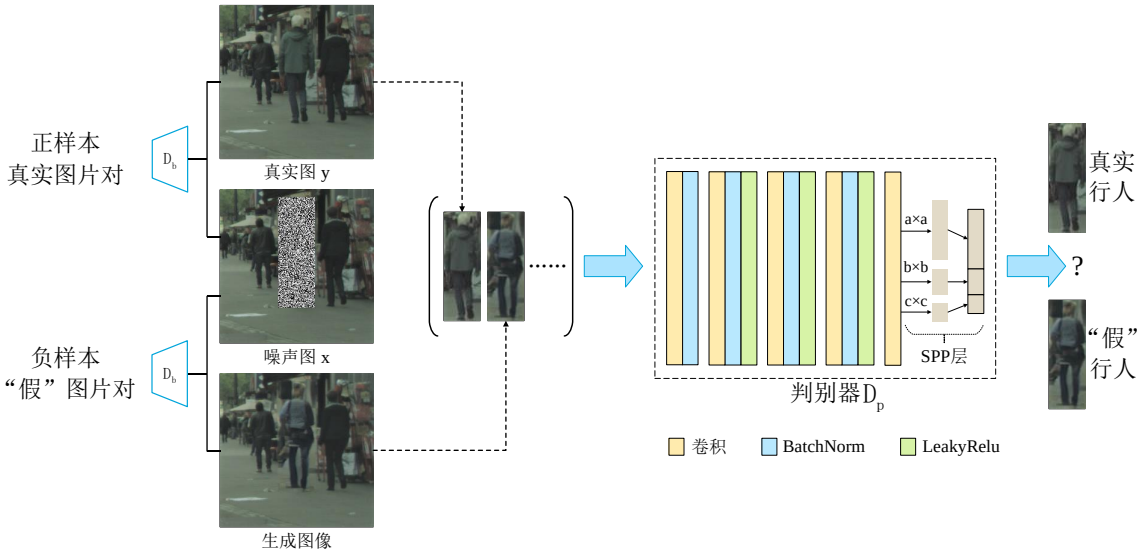

图 3.3　PS-GAN 的两个判别器 D_b 和 D_p 的结构示意图

判别器 Dp 的总体结构如图 3.3 右半部分所示，本文采用了一个 5 层卷积神经网络的结构，并且加入了激活函数（LeakyRelu）和批标准化（Batch Normalization）层。激活函数层是为整个模型提供非线性性，常见的激活函数有 Sigmoid 和 Tanh 等等，但是这些激活函数都存在一些缺陷，例如容易在模型优化时导数消失。LeakyRelu 是一种较新的激活函数设计，其定义如下，其中的参数 λ 是可以通过反向传播进行更新的：

$$f(x)=\begin{cases} x, x>0 \\ \lambda x, x\le 0 \end{cases}. \tag{3.1}$$

批标准化（Batch Normalization）层[51]是一种应对网络参数协方差变化的重要技术，这个层的加入可以使得网络参数保持标准的高斯分布，可以防止产生参数“爆炸”（即参数过大）。它可以加速模型的收敛速度，同时提高模型的泛化性。本文在判别器 Dp 的设计中加入了这两个最新的技术来提升网络的能力。

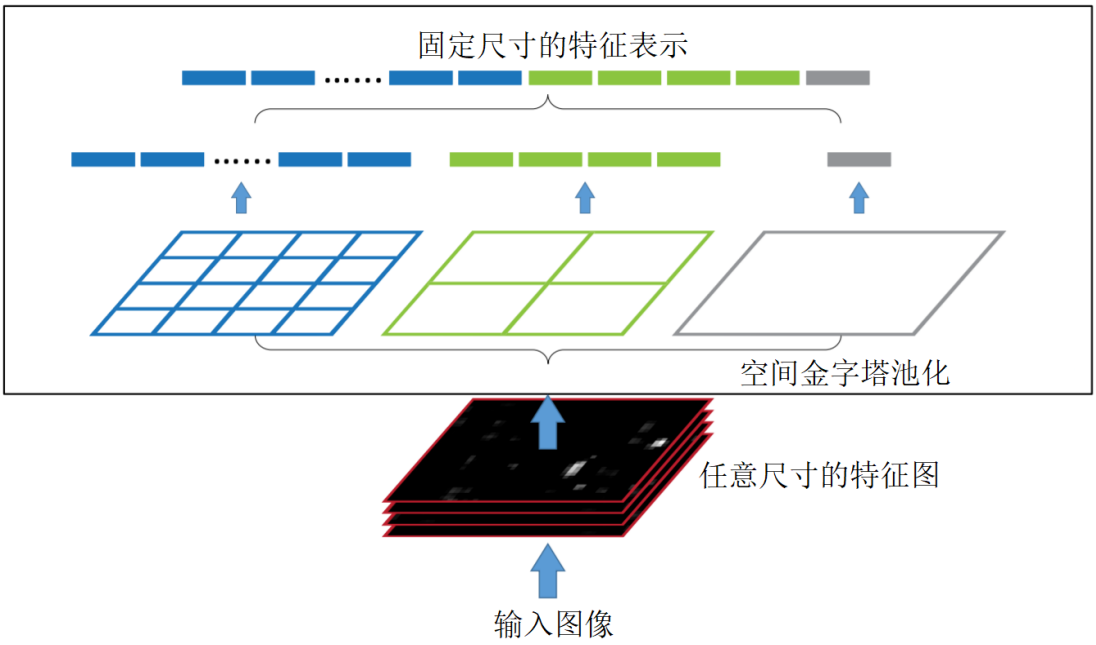

图 3.4　空间金字塔池化（SPP）的原理示意图

同时，本文创新性地在判别器网络 D_p 中加入了空间金字塔池化 SPP 层。通常而言，判别器网络一般都是接收固定尺寸的图片输入。但是对判别器 D_p 而言，由于需要输入从图片中抠出来的行人，而这些行人往往具有各种各样的尺寸和长宽比，如果把这些输入都简单地调整（Resize）到同一个尺度，这样将导致行人外形的严重形变

使得输入信息失真。为了解决这个问题，本文从物体检测领域引入了空间金字塔池化（SPP）层。该技术的作用是可以接收不同尺度图片的输入，从而克服了一般卷积神经网络只能处理固定尺寸图片的缺陷，从而避免了由图片调整（Resize）所带来的图片变形问题。空间金字塔池化，顾名思义，就是在网络中加入了一个特殊的池化层，该池化层把特征图按不同的比例切割成一个个小块（如图 3.4），然后在每个小块中取最大值输出拼接成一个特征向量。在本文提出的网络中，就如图 3.3 所示，特征图被 4*4, 2*2 和 1*1 这三种比例切割成小块，然后分别取其中最大值拼接成一个固定长度的特征表达。

由于这些技术的加入，使得判别器 D_p 可以很好地分辨行人是否真实。在训练时，对于判别器 D_p 的训练，使用了传统的生成对抗网络的损失函数来优化网络参数，详细的公式解释会在本章节 3.4 小节中进行说明。

3.3 判别器 D_b 的网络结构

整个模型设计的目的不仅仅只是去生成一些非常逼真的行人，另一个非常重要的目标是能够将行人非常自然地合成在真实的场景图像中。比如图 1.1 中，文献[7]已经可以用 3D 模型渲染出较为逼真的行人，但是这样人工地把行人放置在真实场景中会得到非常不自然的图像，这样的数据给行人检测模型训练也会导致这些模型学习到偏离真实数据的分布。这个问题的存在就需要模型能够去学习图片的语义信息，如光照条件、背景环境等等。

在本文的模型中，判别器 D_b 就是用来学习背景语义信息的。在判别器 D_b 的设计中，受启发于 Pix2Pix GAN 这个模型。Pix2Pix GAN 是用于图片转换的模型，可以从一个图片域学习到另一个图片域。在这个模型中，它的判别器设计与一般的生成对抗网络不同，它的判别器同时接收原始图和生成图或者原始图和真实转换图做成一个图片对（Image Pair）输入，判别器需要来判定这是否为真实图片对。在本文的模型中，同样，判别器 D_b 用来区分是否为真实图片对。真实图片对为噪声图 x 和真实图 y，而“假”图片对为噪声图 x 和模型生成的图。

判别器 D_b 的总体设计如图 3.3 的左半部分所示，整体的网络结构和 DCGAN 中的判别网络是基本一致的，但是有一些修改：1）因为要接收图片对输入，所以第一个

卷积层需要将通道数改为 6 个（两张彩色图片通道数叠加）；2）在判别器的最后使用了 PatchGAN 的设计。普通生成对抗网络在网络最后只给一个数值输出，然后根据这个数值输出做二分类来判断真假。PatchGAN 则是在网络最后给出一个特征块（Patch）的输出，然后根据这个特征块来判断真假，这样最后的特征块中包含了更多信息可以更为准确地对输入做真假判断。在本文的模型中，使用了一个 70*70 的特征块作为输出；3）使用 LSGAN 的损失函数而不是普通生成对抗网络的损失函数来优化这个模型的参数。相比于普通的生成对抗网络，LSGAN 把 Sigmoid 交叉熵损失函数换成了最小二乘的形式，这样模型参数在优化时可以更好地收敛和更稳定地训练。在下面的章节 3.4 中，将会用数学公式详细展示本文使用的损失函数。

3.4 模型参数训练优化方法

损失函数是深度学习模型训练过程的目标，也是整个模型参数的迭代方向，所以设计好损失函数对深度学习模型至关重要。正如在图 3.1 和图 3.3 中所示，本文设计的模型中包括两个对抗学习过程：$G \Leftrightarrow D_b$ 和 $G \Leftrightarrow D_p$。在整个损失函数的设计中同时使用了两种不同的损失函数，原始 GAN 的 Sigmoid 交叉熵损失和 LSGAN 中的最小二乘损失。这两种损失函数的混合使用可以给 PS-GAN 带来最好的行人生成效果，具体这两种损失函数对生成效果的影响将会在本论文的第四章中进行展示并详细说明，在这里先从数学角度介绍整个模型的损失函数。

首先说明的是生成器 G 和判别器 D_b 的对抗学习过程，公式表示如下：

$$\mathcal{L}_{LSGAN}(G,D_b)=E_{y\sim p_{gt\cdot image}(y)}[(D_b(y)-1)^2]+E_{x,z\sim p_{noise\cdot image}(x,z)}[(D_b(G(x,z)))^2], \quad (3.2)$$

其中 x 是带有噪声框的图像，y 是真实的带有行人的图像。在这里，使用 LSGAN 的最小二乘损失函数取代原始的生成对抗网络的损失函数。

为了在噪声框中生成尽可能逼真的行人，网络中也有生成器 G 和判别器 D_p 的对抗学习过程：

$$\mathcal{L}_{GAN}(G,D_p)=E_{y_p\sim p_{pedestrian}(y_p)}[\log D_p(y_p)]+E_{z\sim p_{noise}(z)}[\log(1-D_p(G(z)))], \quad (3.3)$$

其中 z 是噪声图 x 中的噪声框部分，y_p 是在真实图 y 中抠出来的行人图像。在这里，使用了普通的生成对抗网络的 Sigmoid 交叉熵损失（Sigmoid Cross-Entropy Loss）来更新网络参数。

同时，生成对抗网络的参数训练中也可以同时引入传统的损失函数来帮助模型训练[23]。在本文的模型中，使用了 L1 损失来控制生成图像和真实图之间像素级的差异：

$$\mathcal{L}_{\ell_1}(G) = E_{x,z\sim p_{noise\cdot image}(x,z),y\sim p_{gt\cdot image}(y)}[\|\mathrm{y} - G(x,z)\|_1], \tag{3.4}$$

通过 L1 损失的控制，可以辅助网络尽量学会去生成较为真实的图像。

最后，综合上述的所有损失函数来训练更新 PS-GAN 的所有参数，最终的损失函数定义为：

$$\mathcal{L}(G, D_b, D_p) = \mathcal{L}_{LSGAN}(G, D_b) + \mathcal{L}_{GAN}(G, D_p) + \lambda \mathcal{L}_{\ell_1}(G). \tag{3.5}$$

其中 λ 是用来控制 L1 损失在总体损失函数中的比重，在本文实验中发现，这个值设置成 100 是一个很好的选择。在整个模型优化过程中，通过这三种不同损失的加权和来控制整个模型参数的训练优化方向。

3.5 本章小结

由于传统方法合成的行人非常不真实，本文设计一种可以用 GAN 网络来做行人生成的方法。传统方法很难人工地去模拟所有的真实环境，而 GAN 则可以以数据驱动的方法从训练数据中自动学习到如何在真实的背景图中合成行人。这也是第一次 GAN 被引入做物体检测类任务的数据增强，提出的方法使得 GAN 在真实场景中合成行人的同时也可以给出图中行人的位置。

为了提升行人合成的效果，本章提出了多判别器结构的生成对抗网络：Pedestrian-Synthesis-GAN（PS-GAN）。PS-GAN 中使用了多个判别器的结构，分别对生成行人的效果和行人在背景中自然合成效果进行增强。其中一个判别器 D_b 可以使得 PS-GAN 去学习背景信息，例如道路结构、光照条件等等。这个判别器可以使得合成的行人自然地融合在背景中。另一个判别器 D_p 可以使得 PS-GAN 学习到如何生成具有逼真外形和丰富细节的行人。与此同时，为了解决不同尺度的行人问题，本文还在生成对抗网络中加入了空间金字塔池化层 （SPP）。SPP 的使用可以使得判别器

D_p 直接接收不同尺度的行人图像的输入，去除了对行人图像调整（Resize）到同一尺度导致物体形变的问题。另外，对于不同判别器，使用了不同的损失函数进行参数的优化使其达到最优的效果。

综上，PS-GAN 在能够生成逼真行人的同时，也可以使得行人自然地融合在真实场景中。为了说明其生成效果，将在下一章中使用 PS-GAN 在真实的数据集中生成行人，展示其生成的图片，同时使用生成的数据和真实数据去训练 Faster R-CNN 这一检测器，用检测器的检测精度的变化定量说明 PS-GAN 的生成效果。同时，本文也展示了 Pix2Pix GAN 的行人生成效果，Pix2Pix GAN 将在下面的实验中作为基准模型进行对比分析。

4. 真实场景数据中的实验和评估

为了说明提出的模型能够在真实场景中合成逼真的行人，首先在一个大规模的数据库 Cityscapes[26]测试本文提出的 PS-GAN，同时展示了在这个数据库中图片合成的质量。同时为了定量分析 PS-GAN 的效果，同时用真实的数据和加入了 PS-GAN 的生成图片的数据去训练 Faster R-CNN 模型。Faster R-CNN 是一种以卷积神经网络为基础的通用物体检测模型，该模型是目前效果最好的几个检测模型之一，已经在很多不同的基准数据库中证明了其优异的性能和鲁棒性。在本论文的实验中，使用的 Faster R-CNN 的模型都是以 VGG16[52]为基础的模型。本文中同时用真实数据和生成数据去训练这个检测器，通过比较加入生成数据之后检测器对行人检测准确率的变化就可以评估生成图片的质量。

为了进一步说明本文模型的鲁棒性，也为了证实模型可以在新的视频场景中仍然保持生成质量较好的图片，本实验中也在另一个数据库 Tsinghua-Daimler Cyclist Benchmark[27]中测试提出的模型。这些实验将会在本章节中详细介绍，需要说明的是：所有的实验都是使用 PyTorch[53]搭建的，并且在两块 Titan X GPU 上运行完成的。

4.1 在 Cityscapes 数据集上的实验结果

Cityscapes 数据库是一个用于城市场景语义信息理解的大规模数据库，里面包含了 50 个不同城市用双目摄像头收集而来的视频数据。相比于其他比较早期提出的行人检测的数据库 Caltech Pedestrian[54]和 KITTI[55]，Cityscapes 里面的图片分辨率更高（1024*2048 的图像），质量更好，并且包含丰富的各种形态的行人，这些都对训练提出的 PS-GAN 更为适合。

4.1.1 定性分析实验结果

这一部分将展示 PS-GAN 在 Cityscape 上生成图片的效果，并且和其 Pix2Pix GAN 进行对比，从视觉上定性分析 PS-GAN 合成图片的优异性能。

首先，在 Cityscapes 这个数据库中给的数据标签并不是行人的边界框信息，因为这个数据库采集原本的目的是用来做图片语义分割的作用，所以所有物体给的都是像

素级标签信息如图 4.1 所示。从而需要从这些像素级的标签中首先得到行人的边界框标签，又因为在这个数据库中几乎把所有的行人都标注出来了，其中也包括一些非常小和遮挡严重的行人。为了去除这些像素信息过少的行人，在生成行人边界框时选择把高度小于 70 个像素点和宽度小于 25 个像素点的行人去掉，这么小的行人即使能够检测出来也没有很大的实际意义。

图 4.1　Cityscapes 中像素级标签信息示例

在经过以上处理之后，在这个数据库中，一共可以得到 2326 张有标注信息的图像，其中总共包括了 9708 个有边界框标注的行人。在本文实验中，在其中随机选取了 500 张图片作为测试集，剩下的 1826 张图片作为训练集来训练 PS-GAN 模型。本实验中并不直接输入数据库中 1024*2048 的图片去训练 PS-GAN，这样的输入太大会导致模型需要学习的信息过大而无法很好地训练。替代的是，在原始图片中截取 256*256 的包含行人的图片块来对 PS-GAN 进行训练。在本文实验中，一共从 1826 张原图中拿取了 1200 个图片块来对 PS-GAN 模型进行训练，这些图片块中的人有着较为清晰的行人外形，这样对模型训练更有利。

为了全面地展示 PS-GAN 生成的行人的效果，本文设计了两组实验：1）在原本真实的行人框中用 PS-GAN 生成新的行人；2）在原本没有行人的位置生成新的行人。对于第一组实验，在 500 张测试图像中按照训练数据的处理方法：即在 1024*2048 的原图中抠取 256*256 的包含行人的图片块，然后把这些行人所在位置覆盖噪声框，用训练好的 PS-GAN 模型在这些噪声框里面生成新的行人。这样可以直观地对比同一位置处真实行人和生成行人的效果，实验结果展示在图 4.2 中。在图 4.2 中，将本文提出的 PS-GAN 和其他 4 种模型进行比较，其中前两列分别展示的是原始图和噪声图。基准模型是 Pix2Pix GAN，由这个基准模型生成的效果图在第三列。接下来的三列由对比模型 A、B、C 生成，其中，模型 A：基本架构和 PS-GAN 一致，但是在判别器

中移除了空间金字塔池化（SPP）层；模型 B：和最终的 PS-GAN 不同的是两个判别器中都使用了 LSGAN 的最小二乘损失函数；模型 C：和最终的 PS-GAN 不同的是两个判别器中都使用了普通的生成对抗网络的损失函数。最后一列是本文提出的 PS-GAN 的生成效果图。

对于第二组实验，在空白的场景中随机取 256*256 的图片块。同时考虑到行人不可能出现在不可能的位置如在墙上或者车上，去除这些不合理的图片块之后，在剩余的图片块中加入噪声框然后用训练好的 PS-GAN 模型在这些噪声块中生成行人，生成的效果图展示在图 4.3 中。

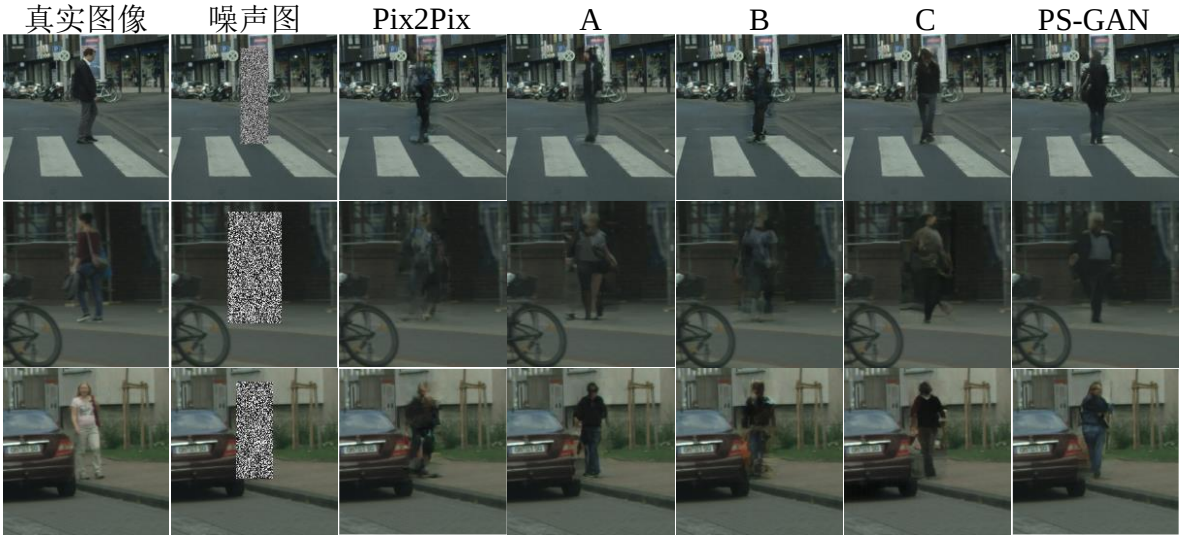

图 4.2　PS-GAN 和对比模型在 Cityscapes 中生成图片的效果图

图 4.3　PS-GAN 和对比模型在 Cityscapes 的空白背景图上生成图片的效果图

在图 4.2 和图 4.3 中，不仅仅只罗列了 PS-GAN 的合成效果图，同时也展示了其他对比模型的合成效果图，所有的模型都是在相同的训练集中训练了 200 次迭代的结果。可以看到无论是在图 4.2 或图 4.3 中，与基准模型 Pix2Pix GAN 相比，PS-GAN 模型生成图片的质量都有明显的提高。几乎所有的 Pix2Pix GAN 生成的结果都有着模糊的行人外形，但是 PS-GAN 模型可以得到非常清晰的行人。这样的结果充分证明了判别器 D_p 的加入，可以迫使生成器 G 学会很多的行人细节信息从而生成更为逼真的行人。

同时，为了分析空间金字塔池化（SPP）层的加入对模型的效果，本文实验设计了对比模型 A，该对比模型中在判别器 D_p 中去除了空间金字塔池化层。正如在图 4.2 和图 4.3 中所示，虽然模型 A 也可以生成较为清晰的行人图像，但是和 PS-GAN 模型的效果图对比，可以发现模型 A 的行人细节部分不如 PS-GAN 模型的丰富。例如在图 4.2 和图 4.3 的第一列中，PS-GAN 模型生成的行人的腿都可以被清晰地看到，但是模型 A 生成的行人都比较模糊。

在本文实验中，研究发现在判别器 D_b 中使用 LSGAN 的最小二乘损失函数可以很好的帮助模型学习背景语义信息，但是在判别器 D_p 中使用普通的生成对抗网络的损失函数反而效果更好。为了展示这一现象，本文设计了模型 B 和模型 C。对于模型 B 而言，这个模型在两个判别器中都使用了 LSGAN 的最小二乘损失函数，无论是在图 4.2 或图 4.3 中，它的效果都仅仅好于基准模型 Pix2Pix GAN 但是完全比不上 PS-GAN 模型的效果。对于模型 C，在两个判别器中都使用普通的生成对抗损失函数，实际上模型 C 也可以生成非常逼真的行人效果。在图 7 的最后一行中可以看的，在某些情况下，模型 C 甚至可以得到比 PS-GAN 模型还好一点的行人效果。然而模型 C 的问题在不能很好地学习到背景语义信息，可以看到在图 4.2 和图 4.3 中，模型 C 在噪声框中生成的图像有着和旁边明显的边界感，但是 PS-GAN 模型生成的图却没有这一现象。

经过分析认为这个现象是由 LSGAN 的最小二乘损失的特点决定的。最小二乘损失是计算生成图和真实图之间每个像素的平方差进行优化的，这样就导致最小二乘损失相比于普通的生成对抗网络损失来说对图像中每个点的变化更为敏感，因为每个小

的变化经过平方之后就会被放大。所以最小二乘损失会使得模型尽可能地去学习整个图片的总体信息，而普通的生成对抗网络损失则会对细节学习更好。这就正好符合本文设计的两个判别器的作用：D_b 去学习背景信息，D_p 专注于行人细节。所以在设计 PS-GAN 时，判别器 D_b 使用的是 LSGAN 的最小二乘损失，而判别器 D_p 与生成器 G 的对抗学习过程使用的是普通的生成网络损失。

在图 4.4 中展示了更多的对比结果，可以看到 PS-GAN 模型的合成效果都是显然好于其他模型的。为了更清晰地展示 PS-GAN 模型生成的行人效果，把生成的行人从合成图中扣出来，并且和原图对比并且展示在图 4.5 中（左边 5 个为 PS-GAN 生成的行人，右边 5 个为真实的行人）。可以看到 PS-GAN 模型生成的行人有着清晰的外形和丰富的细节。与文献[56]中工作相比，为了给行人重识别增强数据，他们使用了 12936 张图片来训练生成对抗网络，然而本文只用了 1200 张图片就可以得到非常清晰的行人图像。

图 4.4　PS-GAN 和对比模型在 Cityscapes 中生成图片更多示例效果图

图 4.5　PS-GAN 生成的行人和真实行人对比

4.1.2 定量分析实验结果

在上面的小节中，已经展示了 PS-GAN 模型合成行人的图片效果。在这一部分中，将结合 PS-GAN 模型合成的数据和真实数据来训练行人检测器 Faster R-CNN 从而定量地分析数据增强的效果。在这部分实验中，生成行人的实验设置和上面的章节是一致的，同样也是在 256*256 的图片块中生成行人。但是在这一部分中，将 256*256 的图片块贴回原图中从而给行人检测器得到新的训练数据，示例图片展示在图 4.6 中，左边展示的是原始的真实图片，右边是加入了合成的行人之后的效果图，可以看到行人都被非常自然地合成在真实场景中。同时 PS-GAN 模型也是仅仅用 1826 图片中的行人训练得到。

图 4.6　在 Cityscapes 数据集的真实场景图片中合成行人的效果图

为了比较合成的数据对检测器 Faster R-CNN 的提升效果，本实验中使用 3 中不同的数据来训练 Faster R-CNN。首先，是用 1826 个真实图像训练得到的 Faster R-CNN 作为基准模型，然后另外两个 Faster R-CNN 是分别加入来自 Pix2Pix 和 PS-GAN 合成数据得到的。需要说明的是，在加入合成数据时，所有的生成的行人都是加在原始图像里而没有引入新的图像。所有的检测器都是在同样的 500 张图片的测试集中进行测试，并且平均精度（Average Precision, AP）都是在这些模型收敛之后得到。同时，也测试了加入不同的合成行人数量对效果提升的影响，实验结果展示在表 4.1 中，展示

了在不同的设定下训练得到的 Faster R-CNN 的准确率，同时试验了分别加入 Pix2Pix GAN 和 PS-GAN 的合成数据的效果，表格中括号里的数值表示在对应真实图像中行人的总个数。

表 4.1 Cityscapes 数据集中在不同的设定下训练得到的 Faster R-CNN 的准确率

数据	Pix2Pix GAN	PS-GAN
1826 个真实图像（7729 个行人）	60.11%	
+3000 个合成行人	59.95%	61.02%
+5000 个合成行人	60.23%	**61.79%**
+8000 个合成行人	58.41%	61.59%
Pascal VOC 2007	34.13%	
Pascal VOC 2007 & 2012	36.85%	
300 个真实图像（1173 个行人）	47.08%	
+500 个合成行人	46.97%	47.36%
+1000 个合成行人	46.71%	**48.79%**
+2000 个合成行人	46.12%	48.11%
1000 个真实图像（4368 个行人）	52.72%	
+2000 个合成行人	52.07%	54.41%
+4000 个合成行人	51.68%	**56.19%**
+5000 个合成行人	51.24%	55.96%

可以看到，虽然用 1826 个真实图像已经可以把 Faster R-CNN 训练到一个较为饱和的状态，其精度可以达到 60.11%，但是加入合成数据仍然可以继续提升检测器的效果。当加入 5000 个由 PS-GAN 合成的行人时，可以看到 Faster R-CNN 的检测精度从 60.11%提升到了 61.79%（提高了 1.5%以上），明显提升了检测器的效果。相反地，当加入 8000 个由 Pix2Pix 合成的行人时，Faster R-CNN 的检测精度反而从 60.11%下降到了 58.41%。这种下降是由于 Pix2Pix 合成的行人效果太差，当加入过多的合成行人时使得检测器反而学习到了错误的数据分布。这个实验结果也符合上面章节中 Pix2Pix GAN 合成的行人的糟糕的视觉效果。

为了进一步探究合成数据的效果，本文中进行了更多的对比实验，并且也将结果展示在表 4.1 中，所有的模型都是在同样的 500 张测试图像中进行比较。首先测试了在 Pascal VOC[57]中训练得到的 Faster R-CNN 模型在这个数据集中的检测效果，正如表 4.1 所示，在 Pascal VOC 2007 和 Pascal VOC 2007&2012 数据集中分别训练的 Faster R-CNN 在 Cityscapes 分别可以达到 34.13%和 36.85%的精度。这个现象也说明了在不同场景中训练的 Faster R-CNN 检测效果有一个比较大的下降。同时也使用 300 个真实图像来训练 Faster R-CNN，并且和加入 Pix2Pix GAN 和 PS-GAN 合成数据的模型进行对比。在这个需要说明的是，为了避免 Pix2Pix GAN 和 PS-GAN 看到比 Faster R-CNN 更多的数据，本实验中基于 300 个用来训练 Faster R-CNN 模型的数据重新训练得到这两个生成对抗网络模型。同样的，这些合成的行人也都是加入到原始图像中而没有加入新的图像。正如表 4.1 中所示，用 300 张真实图像训练得到的 Faster R-CNN 可以实现 47.08%的精度。通过加入 PS-GAN 模型合成的数据，检测准确率可以进一步提升。当加入 1000 个合成行人时，取得最好的检测精度为 48.79%，比只用真实数据训练的模型提升了 1.71%而加入 2000 个合成行人可以提升 1.03%。在这些实验中，可以看到加入 Pix2Pix GAN 合成的行人反而会略微降低准确率。

本实验中也用 1000 个真实图像（其中共包含 4368 个行人）来训练 Faster R-CNN。同样的，Pix2Pix GAN 和 PS-GAN 也是基于这 1000 个图像重新训练。进行这个对比实验的目的是观察不同数据的训练数据中加入合成行人对提升 Faster R-CNN 的具体效果，分别加入了 2000、4000 和 5000 个合成行人来训练 Faster R-CNN。从表 4.1 中可以看到，即使 Faster R-CNN 相比于用 300 个来训练时更饱和的状态下，加入了 PS-GAN 的合成数据之后可以实现 56.19%的检测精度，超过只用真实数据的模型 3.47 个百分点。

4.1.3 不同行人检测器的检测效果图

为了更直观地展示用真实数据和加入合成数据训练得到的 Faster R-CNN 的检测效果的差异，将 Faster R-CNN 在测试集中的检测结果图放在图 4.7 和图 4.8 中，在这两张图中比较了不同真实数据量下加入合成行人之后的提升效果情况。在图 4.7 中，左边是在 300 张真实图像上训练得到的 Faster R-CNN 检测结果，右边是加入了 1000 个

PS-GAN 合成行人之后的结果；在图 4.8 中，左边是在 1000 张真实图像上训练得到的 Faster R-CNN 检测结果，右边是加入了 4000 个 PS-GAN 合成行人之后的结果。可以看到加入合成数据后，检测器明显地提升了检测精度，不仅仅能够检测到更多的行人同时也降低了误检率。

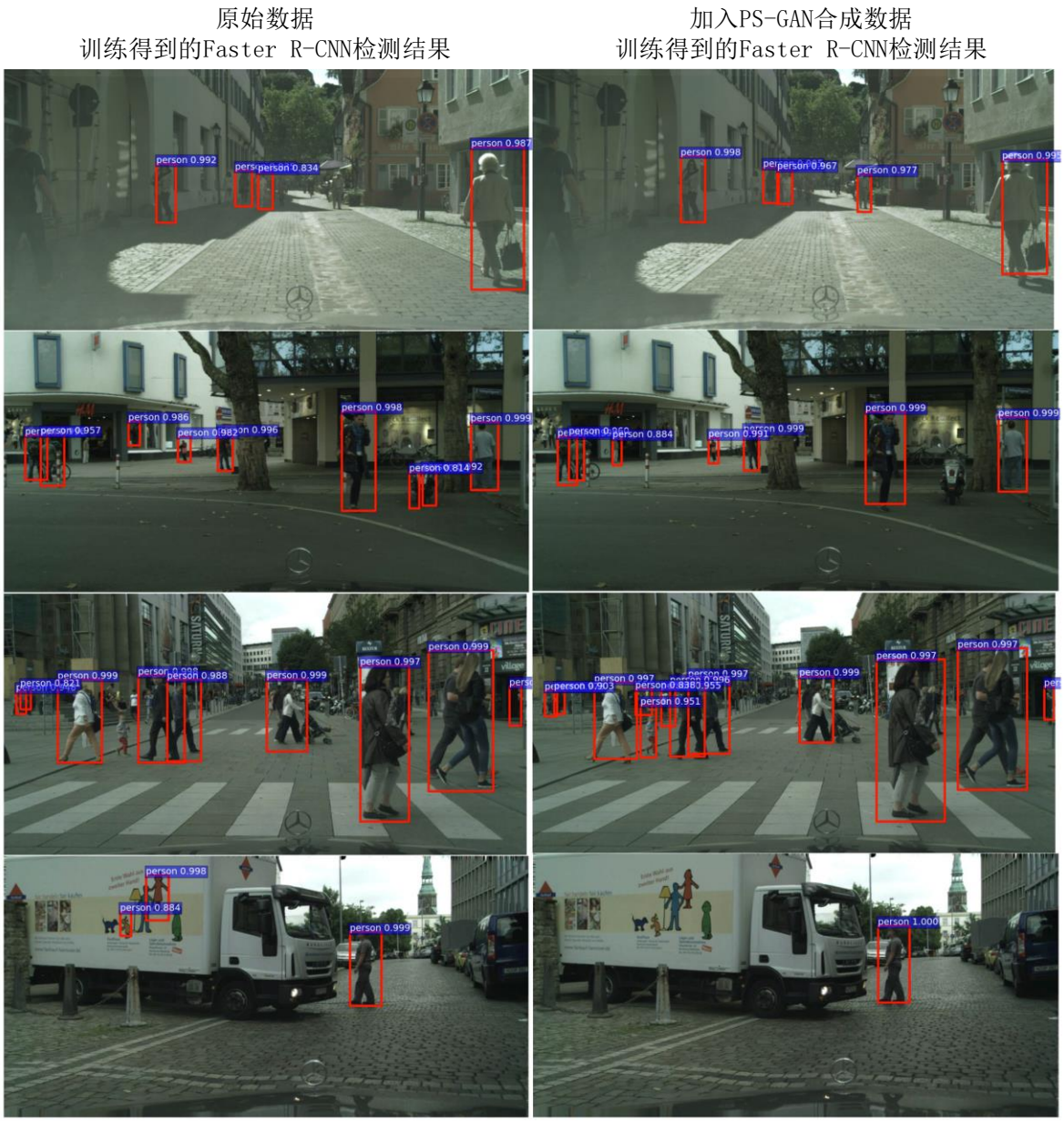

图 4.7　300 张真实图片和加入 1000 个合成行人训练的 Faster R-CNN 在 Cityscapes 上检测效果图

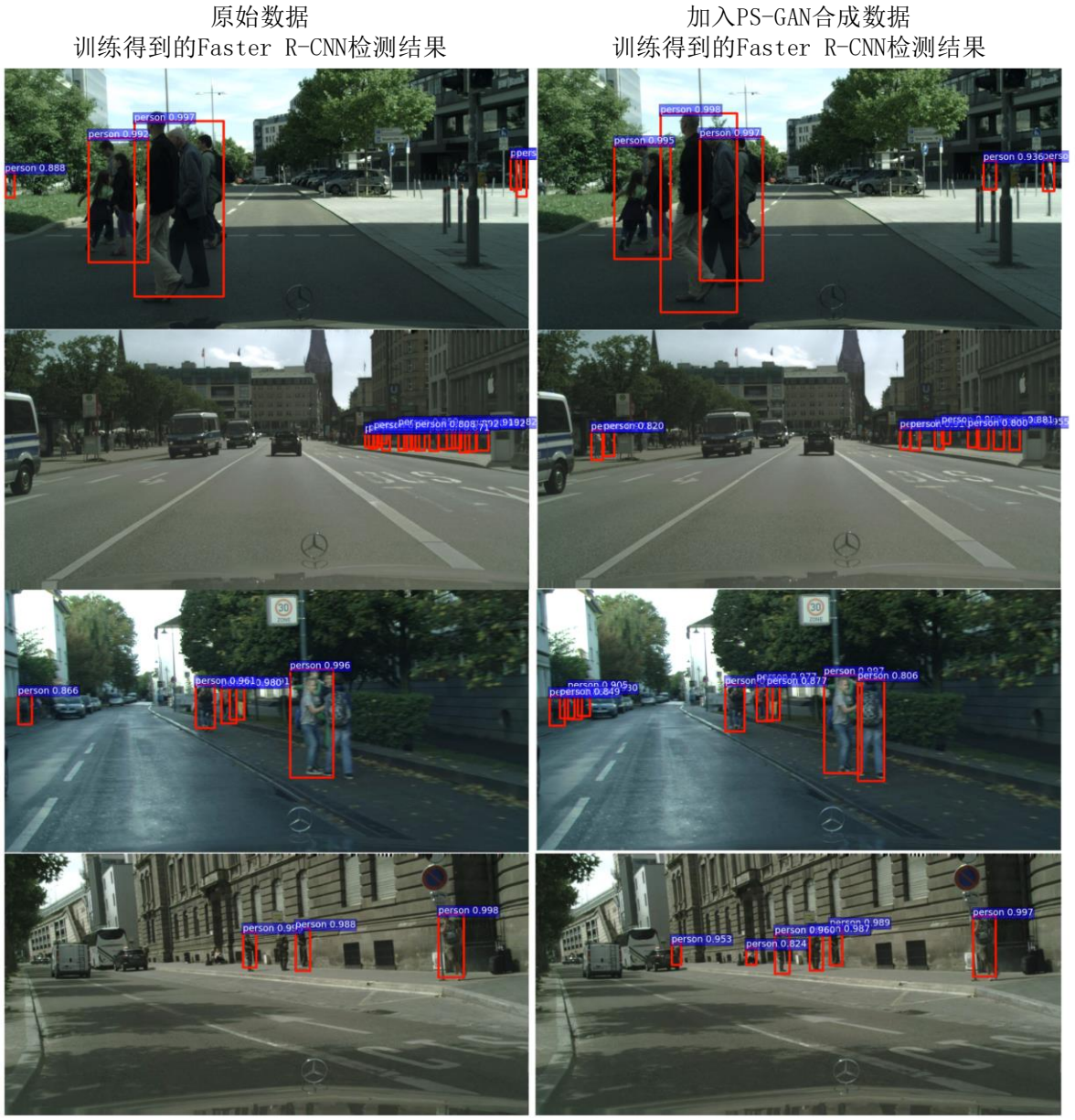

图 4.8　1000 张真实图片和加入 4000 合成行人训练的 Faster R-CNN 在 Cityscapes 上检测效果图

4.2 在 Tsinghua-Daimler 数据集上的实验结果

Tsinghua-Daimler Cyclist Benchmark 是一个收集用来做骑车人检测的数据集。这个数据集中包含 4 个部分：训练集、验证集、测试集和空场景集（“NonVRU” set），其中训练集中包括 9741 个图像，验证集中包含 1019 个图像，测试集中包含 2914 个图像。虽然这个数据库是采集作为骑车人检测的数据库，但是这个数据库中在测试集中同样给了行人的边界框，所以这个数据也可以用来验证行人检测模型的效果。并且

这个数据库中还给了一个空场景数据集（“NonVRU”Set），这个空场景数据集中包含了 1000 张没有任何标注的图片，都是空白的场景图。

为了展示本文提出的 PS-GAN 的泛化能力，还进行了交叉数据集的实验。交叉数据集实验的目的是模拟将提出的模型直接应用在新的未标注或者弱标注的场景的情况。在物体检测中，当检测器可以在训练时看到与测试集中类似的场景时可以显著地提高物体检测的精度。所以如果 PS-GAN 可以在新的场景中仍然保持不错的行人生成效果，那么当面对一个新的场景时，PS-GAN 可以在新的场景中生成数据给行人检测器训练，这个方法将可以有效地提升检测器在新的场景中的检测精度。

图 4.9　PS-GAN 和对比模型在 Tsinghua-Daimler 数据集的空白背景中合成行人的效果图

4.2.1 定性分析实验结果

首先，直接使用在 Cityscapes 数据（1826 个训练图片）中训练得到的 PS-GAN 模型在这个数据库的空白场景中生成行人。选用的空白场景图来自这个数据库的“NonVRU”集，但是其中有些图片不适合生成行人比如没有道路、光线过亮或者过

暗，去除这些不合适的图片之后，得到 650 张空白场景图。和在 Cityscapes 上面的做法类似，从这些图片中抠取出 256*256 的图片块并且在其中放入噪声框来做行人生成，效果图展示在图 4.9 中。可以看到，即使没有从 Tsinghua-Daimler Cyclist Benchmark 这个数据库中拿任何图片去继续训练 PS-GAN 模型，PS-GAN 仍然可以在这个数据库中生成高质量的行人和逼真的图像。

图 4.10　在 Tsinghua-Daimler 数据集的真实场景图片中合成行人的效果图

同样，本实验中也把带行人的 256*256 的图片块贴回原图中来得到给行人检测器数据增强的数据，效果图展示在图 4.10 中，左边展示的是原始的真实图片，右边是加入了合成的行人之后的效果图。需要说明的是，在 Cityscapes 和 Tsinghua-Daimler 这两个数据库之间有许多的不同之处，例如背景、光照条件和行人的样子，但是本文提出的 PS-GAN 模型仍然保持一个比较不错的生成效果。当然考虑到这两个数据库之间的差异，相比于在 Cityscapes 上的生成效果，PS-GAN 在这个 Tsinghua-Daimler 上面生成有所下降也是可以被接受的。例如，在 Tsinghua-Daimler 上生成的行人，在合成效果和行人细节方面在某些情况下确实有所下降。但是整体来说，在新的数据库上的生成效果仍然保持了很好的水准，看起来的效果也比较自然。

4.2.2 定量分析实验结果

在这个数据集中本文也进行了定量分析实验，但是和在 Cityscapes 上面有所不同的是，Faster R-CNN 和生成对抗网络的训练都没有使用 Tsinghua-Daimler 的数据而是都使用 Cityscapes 的数据。在这个数据库中，直接使用数据库中的测试集（包含 2914 个测试图像），并且使用数据库中标注的行人和骑车人标签都当成行人，因为骑车人和行人的外形也是非常类似的。实验结果展示在表 4.2 中，同时试验了加入 Pix2Pix GAN 和 PS-GAN 的合成数据的效果。括号中的数值表示在对应真实图像中行人的总个数或者合成图片中生成行人的总个数。

表 4.2 Tsinghua-Daimler 中在不同的设定下训练得到的 Faster R-CNN 的准确率

数据	Pix2Pix GAN	PS-GAN
1826 个 Cityscapes 中的真实图像（7729 个行人）	43.77%	
+650 张背景图（不含任何行人）	44.06%	
+650 张合成图像（4500 个合成行人）	44.11%	**46.41%**
Pascal VOC 2007	23.24%	
Pascal VOC 2007 & 2012	26.50%	
300 个 Cityscapes 中的真实图像（1173 个行人）	32.15%	
+300 张背景图（不含任何行人）	33.06	
+300 张合成图像（2000 个合成行人）	32.64%	**34.77%**
1000 个 Cityscapes 中的真实图像（4368 个行人）	42.42%	
+650 张背景图（不含任何行人）	43.02%	
+650 张合成图像（4500 个合成行人）	42.70%	**44.09%**

在表 4.2 中可以看到，当加入 650 个合成图像时，Faster R-CNN 效果有一个巨大的提升（提升了 2.64%）相比于用 1826 个 Cityscapes 真实图像训练的检测器。在这个数据库中不同于 Cityscapes 中的设置，在空白背景图中合成行人意味着在加入合成的行人同时也加入了新的图片信息。为了反映加入新的图片的影响，本实验中也对比了加入 650 张空的背景图训练得到的检测器的准确率。如表 4.2 中所示，加入空白背景图之后也可以对只用 Cityscapes 上图片训练得到的检测器提升精度，可以带来大概 0.29 个百分点的提升。这种提升的由来主要是可以在新的背景图中减少误检的行人。

同时也可以看到加入 Pix2Pix GAN 合成的行人也可以带来一定的效果提升，但是和 PS-GAN 带来的提升相距甚远，提升效果相差了 2.3 个百分点。

同时，本文也进行了使用不同数量的数据训练 Faster R-CNN 的实验。在表 4.2 中也展示了分别使用 300 和 1000 张图片训练 Faster R-CNN 的结果，并且也比较了加入空白背景图和合成图的不同效果。同样的，如同在上一节做的那样，在这两个对比设置中使用的生成对抗网络模型都是在 300 和 1000 张图片中重新训练过的。可以看到，在这两种情况中检测器的精度在加入合成数据后都可以得到提升。当加入空白背景时，检测器的精度也可以得到小幅度的提升，在这两个对比实验中分别提升了 0.91% 和 0.6%。而加入 PS-GAN 模型的合成数据可以非常有效地提升检测精度，分别提升了 2.62%和 2.52%。

特别是当加入 650 个 PS-GAN 合成图片到 1000 张真实图片时，检测器的检测精度从 42.44%提升到了 44.94%，这一准确率甚至于超过了用 1826 张真实图像去训练得到的检测器的效果（43.77%）。这一现象充分地说明了使用这一方法可以非常有效地帮助检测器提升在新场景数据中的检测精度。在这个数据库中，本文也对比了来自 Pix2Pix GAN 的合成效果，可以明显地看到 Pix2Pix GAN 的效果非常差，和只用真实数据训练得到的检测器效果类似。

4.2.3 不同行人检测器的检测效果图

同样的，为了更直观地展示用真实数据和加入 PS-GAN 合成数据训练得到的 Faster R-CNN 的检测效果的差异，将 Faster R-CNN 在 Tsinghua-Daimler 测试集中的检测结果图展示在图 4.11 和图 4.12 中。在图 4.11 中，左边是在 300 张真实图像上训练得到的 Faster R-CNN 检测结果，右边是加入了 300 张 PS-GAN 的合成图片之后的结果；在图 4.12 中，左边是在 1000 张真实图像上训练得到的 Faster R-CNN 检测结果，右边是加入了 650 张 PS-GAN 的合成图片之后的结果。可以看到，加入 PS-GAN 合成数据训练之后的检测器对于行人可以检测得更多，并且检测框标记的行人更准。

图 4.11　300 张真实图片和加入 300 张合成图片训练的 Faster R-CNN 在 Tsinghua-Daimler 上检测可视化效果图

图 4.12　1000 张真实图片和加入 650 张合成图片训练的 Faster R-CNN 在 Tsinghua-Daimler 上检测可视化效果图

4.3 用预训练的行人检测器评估生成行人效果

最后这个实验也是为了定量地说明 PS-GAN 生成的行人效果是远远好于 Pix2Pix GAN 的，用 PS-GAN 和 Pix2Pix GAN 同时分别在 Cityscapes 和 Tsinghua-Daimler 的空白背景图中生成 500 个合成图片，然后用在真实图片上预训练得到的检测器去检测生

成的行人，这样可以通过对比检测器对 PS-GAN 和 Pix2Pix GAN 生成行人的检测精度来说明这两个模型生成行人的效果。

在实验中，使用了两种预训练的检测器：用 Pascal VOC 2007 的真实数据和用 Cityscapes 中 300 个真实图像。所有的实验结果都列在表 4.3 中，可以看到无论是在 Cityscapes 或 Tsinghua-Daimler 中，Faster R-CNN 对 PS-GAN 生成的行人的检测精度都远好于 Pix2Pix GAN 生成的行人。在 Cityscapes 中生成的行人，PS-GAN 生成行人的检测精度都比 Pix2Pix GAN 高大约 30%，而在 Tsinghua-Daimler 中 PS-GAN 也比 Pix2Pix GAN 高大约 20 个百分点。从实验结果可以看到检测器对于 PS-GAN 生成的行人样本的检测精度都远远高于 Pix2Pix GAN 模型，这说明 PS-GAN 生成的行人是非常接近于真实行人的，从另一个侧面也展示了本文提出的 PS-GAN 的生成能力。

表 4.3 不同预训练模型对 PS-GAN 和 Pix2Pix 模型生成行人效果的检测精度对比

生成对抗网络模型	预训练的检测器	背景	
		Cityscapes	**Tsinghua-Daimler**
PS-GAN	Pascal VOC 2007	84.55%	88.45%
	Cityscapes	90.11%	90.46%
Pix2Pix GAN	Pascal VOC 2007	52.46%	69.42%
	Cityscapes	58.82%	71.68%

4.4 本章小结

本章详细介绍了本文提出的PS-GAN模型生成行人效果，首先进行了在Cityscapes数据集上的实验。不仅仅展示了 PS-GAN 生成行人的视觉效果图，同时也对比了 Pix2Pix GAN 这一对比模型的生成效果图，也展示了在模型中使用空间金字塔池化（SPP）层的作用，以及使用的不同损失函数对最终效果的影响。从视觉效果上看，PS-GAN 可以取得最好的生成效果，其生成的行人不仅非常逼真而且可以和真实背景融合地非常好。

同时也使用真实数据和合成数据分别训练 Faster R-CNN 模型，用 Faster R-CNN 在加入不同合成数据下的检测精度来定量地分析 PS-GAN 生成数据来做数据增强的有效性和可行性。展示了在不同真实数据量和加入不同合成数据量情况下 Faster

R-CNN 检测器的检测精度变化。同时，也把用真实数据和加入合成数据训练得到的 Faster R-CNN 的检测效果进行了展示。这些实验结果都说明了用本文提出的 PS-GAN 来做行人检测数据增强任务的可行性和有效性。

同时为了说明 PS-GAN 的泛化性，还在 Tsinghua-Daimler 数据集中进行了交叉数据集实验，即用只在 Cityscapes 的数据中训练的 PS-GAN 直接在 Tsinghua-Daimler 数据集的图片上生成行人，并且也用 Faster R-CNN 检测器进行定量实验。在 Tsinghua-Daimler 上的实验结果验证了将 PS-GAN 直接应用于新场景数据的可行性，并且这个方法可以很好地提升检测器在新场景数据中的检测效果。

5. 结论

本文提出的 PS-GAN 是第一个将生成对抗网络模型应用于为行人检测或者说物体检测类的任务来生成数据的方法。为了说明这个模型的有效性和鲁棒性，在两个大规模的数据库中评估了 PS-GAN 的性能：Cityscapes 和 Tsinghua-Daimler Cyclist Benchmark。在实验中运用提出的模型在这两个数据库中都进行数据的生成，并且用真实和合成的数据来训练 Faster R-CNN 检测器，以此证明这个模型用于数据增强的有效性。

5.1 论文的主要贡献

本文的具体贡献总结如下：

（1）提出的模型可以生成具体清晰外形和逼真效果的行人图像，并且可以非常自然地合成到真实场景中，由此生成“以假乱真”的训练数据。

（2）第一次将生成对抗模型成功运用于物体检测类任务，解决了如何在生成图像的同时也给出图像的合成行人的位置标注框的问题。

（3）提出了一种新颖的生成对抗网络结构。普通的生成对抗网络往往是由一个生成器 G 和一个判别器 D 进行对抗学习。但是在本文提出的 PS-GAN 中，设计了一种多判别器的网络结构，在训练中出现了一个生成器和两个判别器对抗学习的过程。

（4）用 PS-GAN 生成的数据可以直接用于训练以卷积神经网络为基础的行人检测器。用 PS-GAN 生成的数据做数据增强之后可以有效地提高行人检测器的检测准确率以及检测的鲁棒性。

（5）在交叉数据集的实验中，即使模型在一个数据集上训练然后在另一个数据集上测试，PS-GAN 仍然可以生成较为满意的合成图像，并且依然可以提升行人检测器的检测准确率。

5.2 进一步工作建议

本文提出了一种非常新颖的多判别器结构的生成对抗网络，并且第一次将生成对抗网络应用于物体检测类任务的数据增强。虽然 PS-GAN 可以生成非常逼真的行人图

像并且自然地合成在真实场景图片中，但是仍然存在着一些缺陷。因此，后续的工作可以从以下几个方面展开：

（1）进一步提升行人合成的效果，目前在一些极端条件下，PS-GAN 合成的效果还需要继续提升，例如在过亮、过暗的环境中，图片质量非常差的场景中；

（2）目前 PS-GAN 生成的行人的尺度变化还不够大，不能生成过小或者过大尺度的行人，这也会限制它去生成这些极端情况的数据。同时如何控制 PS-GAN 合成的行人在合理的位置中也是非常有意思的方向，例如不能使得生成的行人在水里或者树上；

（3）同时，将 PS-GAN 拓展到其他物体检测任务的数据增强也是未来重要的方向之一。

致谢

研究生三年生活匆匆结束，三年时光看似短暂其实很快，在研究生学习中我收获了很多。不仅仅是只是学习到了最前沿的知识，更多的是学习到了如何去做科研，如何快速地学习和进入一个新的领域。授人以鱼不如授人以渔，最大的收获就在于学会了如何学习的方法，我相信研究生三年时间扎实的训练会让我之后的工作和生活中受益良多。

在这三年时光中，我最想感谢的人就是我的导师：周潘教授。在我研究生的这段时光中，周潘老师的监督让我高效的学习和成长。我从周老师的身上不仅仅是学习到了专业技术，更学到了很多关于做人和做事的态度。周老师每天在实验室刻苦工作的状态和热爱工作的态度，正在潜移默化地影响着我去做一个踏实工作、热爱生活的人。周老师这几年的言传身教给我带来了很大的成长，从一个懵懵懂懂的本科生成为了可以在国际期刊和会议上发表文章的研究生，这些都要谢谢周老师的悉心指导。

同时，我也要感谢和我一起工作的同组实验室同学。在研究生期间时，遇到问题大家总是相互讨论学习，一起努力，一起前进。我的很多文章也是同组同学一起通力合作的成果，大家一起熬夜做实验和写论文，实验中得到的经验一起分享。在这个过程中，我也从同组的实验室同学身上学习到了很多，十分感谢实验室同学给我帮助和支持。

最后，我更应该感谢的就是我的家人，在我读研期间，我的爸妈一直在背后支持着我，当我遇到问题时，我总会从和他们的交流中获取到力量，他们对我关心才是我能够一直往前的最大原因。总之，在以后的人生道路上，我会用我更大的热情去回报在我成长路上给予我巨大帮助的学校、老师和同学。

最后衷心地感谢各位专家和教授们抽出宝贵的时间评阅本论文！

参考文献

[1] Enzweiler M, Gavrila D M. Monocular pedestrian detection: Survey and experiments[J]. IEEE transactions on pattern analysis and machine intelligence, 2009, 31(12): 2179-2195.

[2] Dollár P, Wojek C, Schiele B, et al. Pedestrian detection: A benchmark[C]//Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009: 304-311.

[3] Dollar P, Wojek C, Schiele B, et al. Pedestrian detection: An evaluation of the state of the art[J]. IEEE transactions on pattern analysis and machine intelligence, 2012, 34(4): 743-761.

[4] Zhang S, Benenson R, Omran M, et al. How far are we from solving pedestrian detection?[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1259-1267.

[5] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(6): 1137-1149.

[6] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[J]. arXiv preprint, 2017.

[7] Cheung, E. C., Wong, T. K., Bera, A., & Manocha, D. (2017). STD-PD: Generating Synthetic Training Data for Pedestrian Detection in Unannotated Videos. arXiv preprint arXiv:1707.09100.

[8] 苏松志，李绍滋，陈淑媛,等. 行人检测技术综述[J]. 电子学报，2012, 40(4):814-820.

[9] 黄咨, 刘琦, 陈致远,等. 一种用于行人检测的隐式训练卷积神经网络模型[J]. 计算机应用与软件, 2016, 33(5):148-153.

[10] Piccardi M. Background subtraction techniques: a review[C]//Systems, man and cybernetics, 2004 IEEE international conference on. IEEE, 2004, 4: 3099-3104.

[11] Sobral A, Vacavant A. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos[J]. Computer Vision and Image Understanding, 2014, 122: 4-21.

[12] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International journal of computer vision, 2004, 60(2): 91-110.

[13] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. IEEE, 2005, 1: 886-893.

[14] Chang C C, Lin C J. LIBSVM: a library for support vector machines[J]. ACM transactions on intelligent systems and technology (TIST), 2011, 2(3): 27.

[15] Oren M, Papageorgiou C, Sinha P, et al. Pedestrian detection using wavelet templates[C]//Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on. IEEE, 1997: 193-199.

[16] Dollár P, Wojek C, Schiele B, et al. Pedestrian detection: A benchmark[C]//Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009: 304-311.
[17] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
[18] Zhang L, Lin L, Liang X, et al. Is faster r-cnn doing well for pedestrian detection?[C]//European Conference on Computer Vision. Springer, Cham, 2016: 443-457.
[19] Li J, Liang X, Shen S M, et al. Scale-aware fast R-CNN for pedestrian detection[J]. IEEE Transactions on Multimedia, 2017.
[20] Costea A D, Nedevschi S. Semantic channels for fast pedestrian detection[C]//Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on. IEEE, 2016: 2360-2368.
[21] Cheung E, Wong T K, Bera A, et al. Lcrowdv: Generating labeled videos for simulation-based crowd behavior learning[C]//European Conference on Computer Vision. Springer, Cham, 2016: 709-727.
[22] Hattori H, Boddeti V N, Kitani K, et al. Learning scene-specific pedestrian detectors without real data[C]//Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on. IEEE, 2015: 3819-3827.
[23] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in neural information processing systems. 2014: 2672-2680.
[24] Pathak D, Krahenbuhl P, Donahue J, et al. Context encoders: Feature learning by inpainting[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2536-2544.
[25] Ledig C, Theis L, Huszár F, et al. Photo-realistic single image super-resolution using a generative adversarial network[J]. arXiv preprint, 2016.
[26] Isola P, Zhu J Y, Zhou T, et al. Image-to-image translation with conditional adversarial networks[J]. arXiv preprint, 2017.
[27] Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 3213-3223.
[28] Li X, Flohr F, Yang Y, et al. A new benchmark for vision-based cyclist detection[C]//Intelligent Vehicles Symposium (IV), 2016 IEEE. IEEE, 2016: 1028-1033.
[29] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436.
[30] Schmidhuber J. Deep learning in neural networks: An overview[J]. Neural networks, 2015, 61: 85-117.
[31] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J]. arXiv preprint arXiv:1312.5602, 2013.
[32] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems.

2012: 1097-1105.
[33] Hinton G, Deng L, Yu D, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups[J]. IEEE Signal Processing Magazine, 2012, 29(6): 82-97.
[34] Sun Y, Chen Y, Wang X, et al. Deep learning face representation by joint identification-verification[C]//Advances in neural information processing systems. 2014: 1988-1996.
[35] Liu Z, Luo P, Wang X, et al. Deep learning face attributes in the wild[C]//Proceedings of the IEEE International Conference on Computer Vision. 2015: 3730-3738.
[36] Lipton Z C, Berkowitz J, Elkan C. A critical review of recurrent neural networks for sequence learning[J]. arXiv preprint arXiv:1506.00019, 2015.
[37] Bengio Y. Learning deep architectures for AI[J]. Foundations and trends® in Machine Learning, 2009, 2(1): 1-127.
[38] Hinton G, Deng L, Yu D, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups[J]. IEEE Signal Processing Magazine, 2012, 29(6): 82-97.
[39] Bengio Y. Learning deep architectures for AI[J]. Foundations and trends® in Machine Learning, 2009, 2(1): 1-127.
[40] Jarrett K, Kavukcuoglu K, LeCun Y. What is the best multi-stage architecture for object recognition?[C]//Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009: 2146-2153.
[41] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks[C]//Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. 2011: 315-323.
[42] Goodfellow I J, Warde-Farley D, Mirza M, et al. Maxout networks[J]. arXiv preprint arXiv:1302.4389, 2013.
[43] Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors[J]. arXiv preprint arXiv:1207.0580, 2012.
[44] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. arXiv preprint arXiv:1511.06434, 2015.
[45] Mao X, Li Q, Xie H, et al. Least squares generative adversarial networks[C]//2017 IEEE International Conference on Computer Vision (ICCV). IEEE, 2017: 2813-2821.
[46] Song F Y Y Z S, Xiao A S J. Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop[J]. arXiv preprint arXiv:1506.03365, 2015.
[47] Zhu J Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks[J]. arXiv preprint arXiv:1703.10593, 2017.
[48] Zhu J Y, Zhang R, Pathak D, et al. Toward multimodal image-to-image translation[C]//Advances in Neural Information Processing Systems. 2017: 465-476.
[49] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks

for visual recognition[C]//european conference on computer vision. Springer, Cham, 2014: 346-361.
[50] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]//International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015: 234-241.
[51] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. arXiv preprint arXiv:1502.03167, 2015.
[52] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
[53] Paszke A, Gross S, Chintala S, et al. Automatic differentiation in PyTorch[J]. 2017.
[54] Dollar P, Wojek C, Schiele B, et al. Pedestrian detection: An evaluation of the state of the art[J]. IEEE transactions on pattern analysis and machine intelligence, 2012, 34(4): 743-761.
[55] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? the kitti vision benchmark suite[C]//Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012: 3354-3361.
[56] Zheng Z, Zheng L, Yang Y. Unlabeled samples generated by gan improve the person re-identification baseline in vitro[J]. arXiv preprint arXiv:1701.07717, 2017, 3.
[57] Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascalnetwork.org/challenges/VOC/voc2007/workshop/index.html

附录 1 攻读硕士学位期间的研究成果

[1] **Xi Ouyang**, Pan Zhou, Cheng-Hua Li, et al. Sentiment analysis using convolutional neural network, PICOM, 2015 IEEE International Conference on. IEEE, 2015: 2359-2364. （EI 检索）

[2] Lieyun Ding, Weili Fang, Hanbin Luo, Peter ED Love, Botao Zhong, **Xi Ouyang**. A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory, Automation in Construction, 2018, 86: 118-124.（SCI A 类）

[3] Chaoyun Zhang, **Xi Ouyang**, Paul Patras. ZipNet-GAN: Inferring Fine-grained Mobile Traffic Patterns via a Generative Adversarial Neural Network, CoNEXT. ACM, 2017: 363-375. （CCF B 类）

[4] **Xi Ouyang**, Shigenori Kawaai, et al. Audio-visual emotion recognition using deep transfer learning and multiple temporal models, ICMI, ACM, 2017: 577-582.（CCF C 类，获得 ICMI 的 Emotiw2017 的 Audio-video Based Emotion Classification Sub-challenge 第六名）

[5] Yan Xu[#], **Xi Ouyang[#]**, Yu Cheng[#], Shining Yu[#], Lin Xiong, et al. Dual-Mode Vehicle Motion Pattern Learning for High Performance Road Traffic Anomaly Detection, CVPR Workshop, 2018.（#表示同等贡献，获得 CVPR 的 NVIDIA AI City Challenge 中 Track2 第一名，CVPR 为 CCF A 类会议）

[6] **Xi Ouyang**, Kang Gu, Pan Zhou. Spatial Pyramid Pooling Mechanism in 3D Convolutional Network for Sentence-Level Classification, Under Reviewing, MINOR REVISIONS before acceptance (AQ). （SCI A 类）

[7] **Xi Ouyang**, Shuangjie Xu, Chaoyun Zhang, Pan Zhou, Yang Yang, Kun He and Xuelong Li. A 3D-CNN and LSTM based Multi-task Learning Architecture for Action Recognition, Under Review, REVISIED AND RESUBMITTED (RQ, Major Revision).（SCI A 类）

附录 2 攻读硕士学位期间参与的科研项目

[1] 国家自然科学基金，“基于学习机制的自组织认知无线网络安全协议研究”，项目编号：61401169。