

四川大學

本科生毕业论文（设计）



题 目 基于生成对抗网络的高速铁路巡检图

像缺陷生成技术及应用

学 院 计算机学院（软件学院）

专 业 计算机科学与技术

学生姓名 姚理议

学 号 2018141461133 年级 2018

指导教师 高绍兵

教务处制表

二〇二二年四月三十日

基于生成对抗网络的高速铁路巡检图像 缺陷生成技术及应用

计算机科学与技术专业

学生 姚理议 指导老师 高绍兵

【摘要】 高铁接触网是铁路系统中重要的供电设备，其零部件的稳定性对高铁安全有着重要意义。目前普遍使用的人工巡检方式效率低下，故障的判断滞后，有必要研发自动化缺陷检测方法。但由于缺陷样本极其稀少，数据不平衡严重，难以直接训练基于深度学习的目标检测方法，以完成缺陷的识别。旋转双耳销钉是高铁接触网中重要的紧固件，列车运行中的振动、冲击可能会造成开口销的缺失，成为安全隐患。针对这些问题，本文提出了一种基于条件生成对抗网络的缺陷样本生成方法，在旋转双耳销钉样本中进行实验，使用少量缺陷样本为生成器的训练提供参照，将正常样本重构为异常状态，生成高质量的缺陷样本，解决了数据不平衡问题。选取生成的高质量缺陷样本与正常样本组成新的平衡的数据集，标记为 PASCAL VOC 格式。本文使用新的旋转双耳销钉数据集训练 SSD、Faster R-CNN 和 YOLOv3 模型，并进行了对比试验和消融实验。实验结果发现 YOLOv3 检测效果最好，mAP 值可以达到 97.7%。与异常检测领域的分类算法对比，本文使用生成数据训练的检测模型表现出 SOTA 水平。实验证明，本文提出的缺陷生成方法可以有效提升高铁巡检图像缺陷检测模型的性能，可以应用于接触网系统安全巡检中。

【关键词】 CGAN；缺陷检测；高铁接触网；旋转双耳销钉；目标检测；异常检测；深度学习。

Generative Adversarial Networks-based Defective Inspection Image Generation Technology and Application in High-speed Railway System

Computer Science

Student: YAO Liyi

Adviser: GAO Shaobing

[Abstract] The overhead contact catenary is an important power supply equipment in the railway system, and the stability of its components is of great significance to the security of train operation. Currently, the common method of security inspection is manual recognition, which is inefficient, and the judgment of faults is lagging behind. It is necessary to develop an automatic defects detection method. However, due to the extremely rare occurrence of defective samples and serious data imbalance, it is difficult to directly train high-performance detection object models based on data-driven deep learning methods. The swivel clevis pin is an important kind of fastener in the high-speed rail catenary. Vibration and impact during train operation may cause the lack of split pins and become a safety hazard. This paper proposes a method for defective samples generation, based on conditional generative adversarial networks, which utilizes a small number of defective samples as reference or supervision for the training of the generator, reconstructs normal samples into anomalous states, and generates high-quality defective samples. And experiments are conducted on swivel clevis pin samples. By this method, the imbalance problem can be solved. The generated high-quality defective samples and normal samples are collected to form a new balanced dataset, labeled in PASCAL VOC format. SSD, Faster R-CNN, and YOLOv3 models are trained with the new dataset of swivel clevis pins, and comparative experiments and ablation experiments are also conducted. YOLOv3 has the best detection performance, and the mAP value reaches 97.7%. In contrast to classification algorithms of anomaly detection, detection models trained with generated data exhibits the same performance state-of-the-art ones. Experiments show that the defects generation method in this paper can effectively improve the performance of object detection models, and can be applied to the security inspection of the overhead contact system.

[Key Words] CGAN; defects detection; overhead contact systems; swivel clevis pins; object detection; anomaly detection; deep learning.

目 录

1 绪论	1
1.1 项目背景及研究意义	1
1.2 研究现状分析	3
1.2.1 目标检测技术国内外研究现状	3
1.2.2 异常检测技术国内外研究现状	4
1.2.3 高铁接触网安全巡检缺陷检测相关工作	5
1.3 本文研究内容及方法	6
2 相关基础知识介绍	8
2.1 高铁接触网图像采集	8
2.1.1 图像采集系统简介	8
2.1.2 旋转双耳销钉异常类型及检测难度分析	9
2.2 卷积神经网络简介	9
2.2.1 卷积神经网络原理	10
2.2.2 经典网络结构	11
2.3 生成对抗网络简介	13
3 基于条件对抗生成网络的缺陷样本生成	15
3.1 条件生成对抗网络	15
3.1.1 模型原理	15
3.1.2 网络结构	16
3.2 模型训练	17
3.3 实验结果与分析	19
4 旋转双耳销钉缺陷检测	23
4.1 数据集	23
4.2 实验结果与分析	23
4.2.1 评价指标	23
4.2.2 目标检测与消融实验	24
4.2.1 异常检测 SOTA 算法对比试验	26
4.3 改进思路讨论	27
总 结	29
致 谢	30
参考文献	31

1 绪论

1.1 项目背景及研究意义

高速铁路具有运送量大、速度快、成本低的优势，在我国交通运输网络中起到重要作用。随着技术的进一步成熟，高铁得到了大规模普及，我国高速铁路现已经突破四万公里。在规模如此庞大的铁路系统中，如何确保运营的安全成为管理部门新的挑战。

接触网是高速铁路系统的电力传输装置，保障了高铁列车的电能供应，如图 1-1 所示。接触网中悬挂线用于向列车输电，支撑装置维持系统的稳定性。高铁接触网是铁路系统中的重要一环，输电线或者支撑装置中任何零件出现异常都会成为安全隐患。一旦接触网出现故障，将严重影响高速列车的运行状态，所以需要快捷有效的方法对接触网设备进行安全巡检。对于接触网支撑装饰零部件状态缺陷的检测，目前我国主要采用传统的人工巡检方式，利用安装在巡检车上的接触网悬挂状态检测和监控装置（4C 装置）对接触网各个零部件实施高清照片的拍摄^[1]。虽然新型成像技术的发展可以得到大量高质量的图像，但仍需要在离线状态下对各部件的故障状态进行人工识别，检测周期长达数月，存在作量大、效率低、故障判断滞后等问题。而且，检查员也会因长时间、高密度的工作产生视觉疲劳，存在误判的风险。随着高铁系统规模扩大，这种效率低下的安全巡检方式难以适应高速、精确和自动化检测的发展趋势。因此有必要开发智能巡检技术，可以准确、快捷地识别接触网中关键零部件的缺陷状态并及时报告，为高铁运行提供安全保障。



图 1-1 高铁接触网

（图像由中国铁道科学院基础设施检测研究所提供）

高铁接触网支撑装置零部件众多，常见的有绝缘子、旋转双耳销钉、双耳套筒、绝缘子底座、定位支座等，如图 1-2 和图 1-3 所示。这些零部件对于接触网的稳定有着关键作用，是高铁列车运行安全的保障。在接触网缺陷检测任务中，一方面，需要在图像中对各

种零部件进行定位识别，另一方面则需要判断每一种零部件的状态是否异常。由于零部件众多，而且缺陷状态多样，该任务一直是一项极具挑战性的工作。

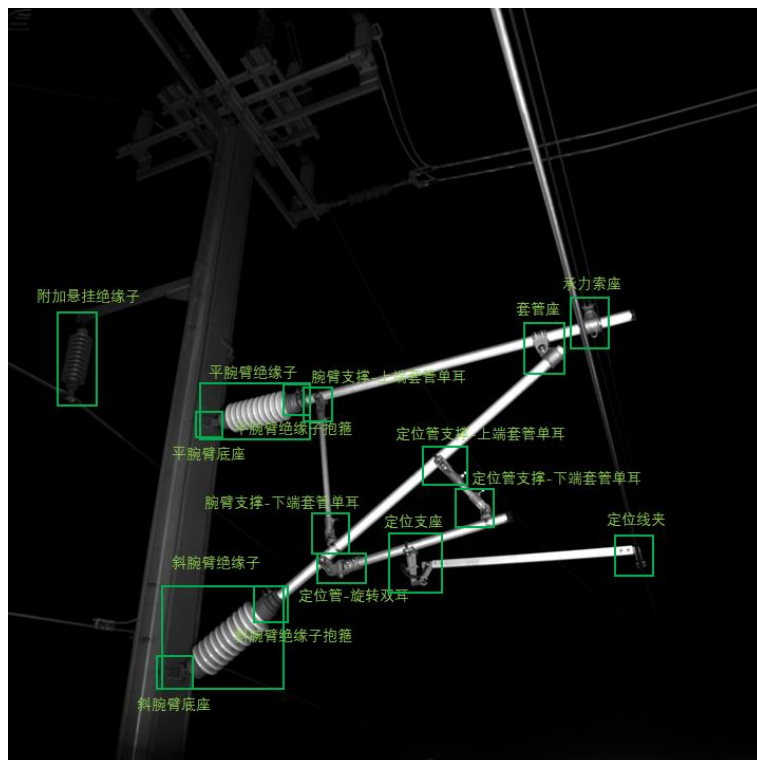


图 1-2 300-350km/h 线路支持装置*
(图像由中国铁道科学院基础设施检测研究所提供)

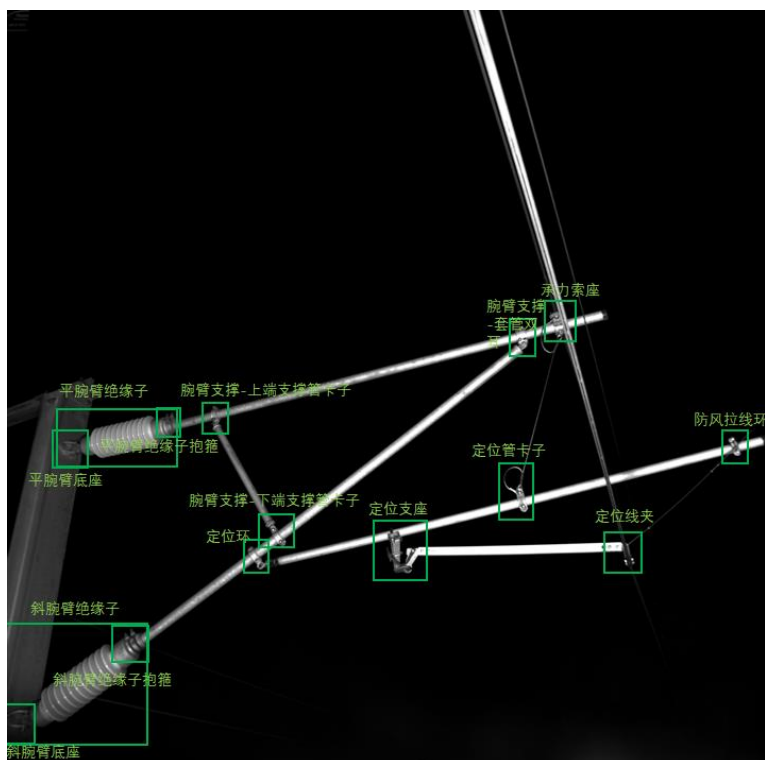


图 1-3 200-250km/h 线路支持装置
(图像由中国铁道科学院基础设施检测研究所提供)

在计算机视觉领域，目标检测技术可以实现不同尺寸物体的定位和分类。随着卷积神经网络（Convolutional Neural Networks, CNN）的发展，基于数据驱动的目标检测模型可以取得高准确率的性能。但是，这些主流的目标检测算法都需要有监督的训练，训练数据的质量对模型的性能有较大的影响。然而，在高铁接触网中零部件的缺陷样本及其稀少，安全巡检中采集到的数据集不平衡问题严重，很难直接用于训练以得到高质量的目标检测模型。针对数据不平衡问题，传统的图像处理方法是进行数据增强，如图像旋转、镜像翻转、添加噪声、锐化等。但这些方法只能从数量上丰富负样本，而无法增加缺陷的多样性，所以无法解决高铁接触网缺陷数据不平衡的问题。

诸如此类缺少负样本的分类任务也被称为单分类问题。在工业界，利用计算机视觉技术来识别生产、制造过程中产品异常的任务被称为异常检测，高铁接触网缺陷检测也是该领域的一个研究与应用方向。工业界的异常检测也属于单分类问题，很多学者在这一领域展开了研究，希望利用无监督或者半监督的方法，仅通过输入正样本进行训练，得到可以准确识别异常样本的模型。目前在异常检测领域，主要有基于预训练模型与生成模型进行训练两种思路。但是，目前异常检测领域 state-of-the-art (SOTA) 算法大部分都是输入小尺寸的样本，如经典的 AnoGAN 模型^[3]只能处理 64×64 的图像。基于上述的难点，本文综合了生成模型的优势，利用条件生成对抗网络（Conditional Generative Adversarial Networks, CGAN）^[4]重构数据的优势，搭建了一个端到端的生成模型，可以将输入的正常样本重构为异常样本。本文将该算法在旋转双耳销钉样本中进行了实验，搭建了一个平衡了数据集，并应用到目标检测模型上，取得了很好的检测效果。该方法兼具创新性与实用性，对实现高铁接触网系统中自动化安全巡检有一定的参考价值。

1.2 研究现状分析

1.2.1 目标检测技术国内外研究现状

目标检测是计算机视觉领域一项基础的 high-level 任务，通过提取目标的特征，在图像背景中实现对特定目标的定位和识别。

（1）传统的目标检测算法

传统的目标检测算法一般包括目标区域选取、特征提取、分类三个步骤。目标选取是通过暴力搜索、边缘检测、滑动窗口^[5]等方法，在图像中形成目标的候选区域。特征提取是计算边缘信息来分析目标与背景像素的特征差异，实现目标的准确定位。传统的特征提取算法有 HOG^[6]、LBP^[7]、SIFT^[8]等。这些算法大都是无监督的，通过人工手动构建特征，一方面是因为图像信息的缺乏，另一方面是受当时有限计算资源的制约^[5]。对于分类，是给每个目标分配标签类型，一般使用传统的机器学习分类器，如 SVM、k-means 等。

（2）基于深度学习的目标检测算法

深度学习中特征的提取是基于数据驱动的，首先在图像分类领域取得了很好的效果，

随后被研究人员应用到了目标检测领域。延续着先定位后分类的思路,最初的基于深度学习算法的目标检测模型是双阶段的。Girshick 等^[9]首先提出了基于 AlexNet^[10]区域卷积网络 (Regions with CNN features, R-CNN),在使用选择性搜索得到的候选区域上进行特征提取,之后用 SVM 完成分类。R-CNN 模型展现出远超传统目标检测模型的性能,有许多研究人员对其进行了改进。针对 R-CNN 计算开销大的问题,文献^[11]提出了 SPP-Net,优化了特征提取过程,减少了对候选区域的重复计算。文献^[12]将 R-CNN 和 SPP-Net 结合,提出了 Fast R-CNN 模型,虽然减少了计算开销,但模型整体的计算效率仍然偏低。为了加速计算,文献^[13]提出了 Faster R-CNN。Faster R-CNN 没有使用选择性搜索算法,而是提出了区域候选网络 (Region Proposal Networks, RPN),区域选择、特征提取、候选框回归等不同模块共享同一网络参数,大大减少了计算开销,并实现了端到端的训练。He^[14]等人提出了 Mask R-CNN,增加了一个 Mask 分支,使感受野提取的特征可以更准确地切合 ground truth。

单阶段的目标检测算法舍弃了区域选取部分,而是利用回归等方法将定位和分类合并到一个部分。最经典的单阶段模型使 YOLO (You only look once) 系列。2015 年,Redmon 等^[15]提出了 YOLO 算法,将图像划分为 $n \times n$ 个网格,计算目标中心在网格中的置信度,最后通过非极大值抑制来去除冗余目标框。这种模型结构简单,计算速度快,但是每个网格只能得到一个目标。文献^[16]改进了 YOLO 的训练过程,将训练数据集分类检测集和分类集,分类用于提高定位和分类的精度,可以识别 9000 多种物体,称为 YOLO9000。文献^[17]做出了进一步修改,提出了 YOLOv3,使用特征金字塔网络 (Feature Pyramid Networks, FPN)^[18]检测多种尺度的目标,并改进原有的网络结构,提出了 DarkNet53,提升了模型性能。文献^[19]提出了 YOLOv4,在 DarkNet53 基础上改进网络结构,引入了 CSP Darknet53,在损失函数、数据增强等方面优化了技术细节。之后还有许多研究人员继续研究,提出了 YOLOX^[2]、YOLOF^[20]等算法。除了 YOLO 系列,文献^[21]提出了 SSD (Single Shot MultiBox detector) 模型,通过生成不同尺度的回归框,检测不同尺寸的对象。文献^[22]提出了 RetinaNet,引入 focal loss 优化了检测器的损失函数,可以取得更好的检测结果。

1.2.2 异常检测技术国内外研究现状

目前大部分异常检测模型仅通过正常数据进行无监督或半监督的训练,大致可以分为基于预训练模型和生成模型两种。

(1) 基于生成模型的异常检测算法

生成模型包括自动编码器 (auto-encoder)^[23]、生成对抗网络 (Generative Adversarial Networks, GAN)^[24]等。用于异常检测的自动编码器假设只有在输入正常数据时,重构的输出输入与输入具有高度相似性;如果输入的是异常数据,重构的结果与输入则存在明显的差异。GAN 模型擅长提取数据的分布特征而进行重新表达,其中的生成器和判别器也可以分别视作自编码器模型中的解码器和编码器。Schlegl^[3]等提出了基于深度卷积生成对抗网络 (Deep Convolutional Generative Adversarial Networks, DCGAN) 的异常检测模

型 AnoGAN, 利用 Wasserstein 距离评估正常和异常数据之间的差异, 并根据隐空间中的数据的映射来计算异常分数。如果异常分数高于特定的阈值, 则样本被判定为异常。文献^[25]改进了 AnoGAN 并研发了 Fast AnoGAN (f-AnoGAN)。F-AnoGAN 修改了模型架构, 优化了训练模式, 拥有更快的速度和更好的性能。文献^[26]提出了一种半监督的编码器-解码器-编码器 (encoder-decoder-encoder) 模型, 称为 GANomaly。GANomaly 侧重于高维空间中原始数据与重构数据的差异, 具有更强的鲁棒性。

(2) 基于预训练模型的异常检测算法

直接使用正样本训练检测模型难以取得好的检测效果, 所以许多研究致力于从其他数据集中转移预训练网络的特征参数, 来检测新场景中的异常数据。文献^[27]研究了基于 CNN 的异常检测迁移学习, 并将单分类支持向量机 (One-Class Support Vector Machine, OCSVM)^[28]应用于该领域, 证明了预训练模型迁移的有效性。这类异常检测方法通常先在大规模的自然图像数据集 (如 ImageNet) 上训练网络, 而减少模型对任务数据集中正样本的依赖性, Patchcore 无需做进一步的调整。文献^[29]它提出了 SPADE (Semantic Pyramid Anomaly Detection) 模型, 利用包含各种特征层次结构的记忆库存进行细粒度、基于 K 最邻近 (KNN, K-Nearest Neighbor, KNN) 算法的异常检测。文献^[30]提出了 PatchCore 模型, 使用了记忆库中代表性最强的补丁特征。文献^[31]提出了 DifferNet 模型, 该模型利用改进的 CNN 来提取特征, 并采用归一化流来估计异常的可能性, 使模型可以有效地捕捉正常数据和异常数据之间的差异。文献^[32]通过使用跨尺度归一化流 (Cross-Scale Normalizing Flow, CS-Flow) 改进了 DifferNet。这种改进提高了隐空间的解释性和定位异常区域的能力。

1.2.3 高铁接触网安全巡检缺陷检测相关工作

在高铁接触网系统中, 研究人员致力于在现有检测方法的基础上, 研发出可以应用于安全巡检过程中的缺陷识别方法。文献^[33]将 Faster R-CNN 模型应用到高铁接触网中 5 个基础零件的检测中, 证明了该目标检测模型在铁路系统零部件检测中的有效性。文献^[34]改进了 Faster R-CNN 的检测过程, 第一步采用混合模型对零部件进行定位, 第二步采用基于深度 CNN 和线性 SVM 的集成分类器对开口销的缺陷状态进行分类, 这种改进可以有效降低背景对故障检测的干扰。文献^[35]参考 PVANET^[36]构建了基于 CNN 的自动编码器模型, 改进了销钉的特征提取方法, 其定位和分类精度更高。但是, 受数据不平衡的影响, 此类算法只能在特定场景下工作或检测某些特定组件, 迁移性较低。文献^[37]修改了 YOLOv4 的网络结构以检测轨道表面的缺陷状态。文献^[38]建立了一种具有双生成器结构的对抗性学习方法, 并改进了判别器结构。该模型具有很高的泛化能力, 并证明了 GAN 在高铁接触网系统安全巡检中的有效性。基于 AnoGAN, 文献^[39]提出了一种缺陷检测模型, 称为 DefGAN (Defect Detection GAN), 它模拟了 GAN 隐空间中缺陷样本的数据分布特征, 提升了检测效果。这种方法可以有效减少数据不平衡对训练的负面影响。文献^[40]利用深度卷积 DCGAN 生成有缺陷的样本, 在大绝缘体和等电位线两种零部件上进行了实验。虽然该模型可以完成分类任

务并取得较高的准确性，但无法在原始图像中对异常样本进行定位，仍未达到实际应用的要求，需要进一步的改进。

1.3 本文研究内容及方法

本文针对高铁接触网系统中零部件缺陷样本稀缺而导致的数据不平衡问题，提出了一种基于 CGAN 的缺陷样本生成方法，并搭建了一个端到端的缺陷生成模型。本文选择了一个极具代表性的零部件——旋转双耳销钉作为实验对象，首先利用少量正样本数据训练了一个 Faster R-CNN，用于定位零件，并剪裁为小图，作为 CGAN 训练数据集。本文改进了 CGAN 网络结构，针对缺陷检测任务优化了训练过程，开展实验并作出分析。最后，缺陷生成模型可以生成大量高质量的缺陷样本，与原始图像一起组成平衡的数据集并在不同的目标检测算法和异常检测算法上进行了实验，通过不同指标分析该方法的可行性和有效性。

本文的技术如图 1-4 所示，分为数据预处理、缺陷生成和缺陷检测三部分。

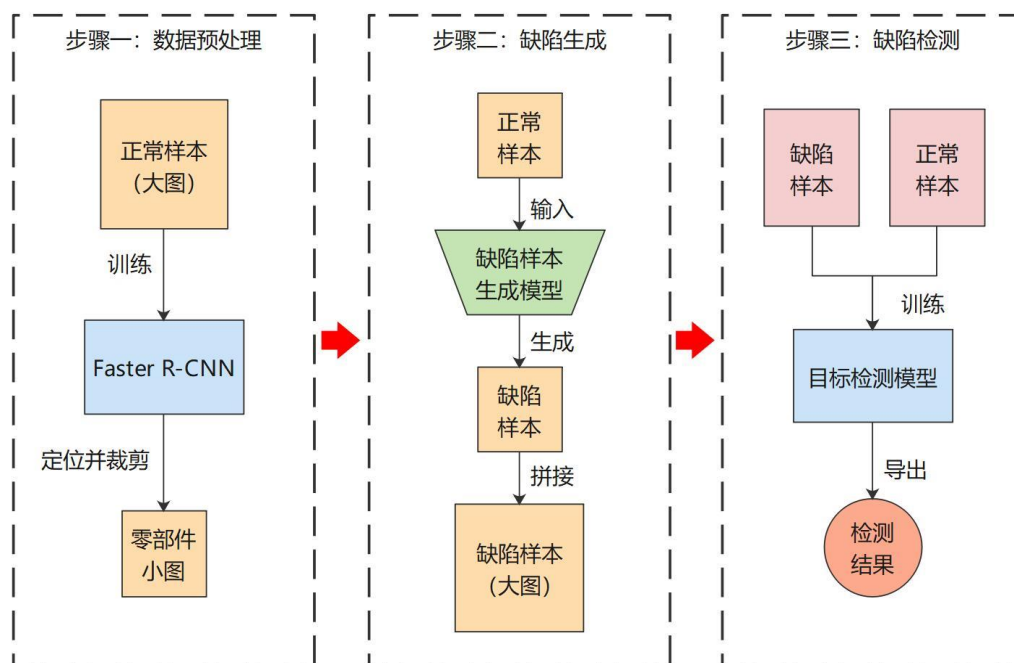


图 1-4 技术路线图

第一章为绪论，主要介绍了课题项目的背景和研究意义，分析了实习高铁接触网系统自动化巡检的意义，并阐述了目标检测、异常检测和高铁接触网系统缺陷识别三个方面国内外的相关研究工作内容。最后说明了本文的研究内容、技术路线和文章内容安排。

第二章介绍课题相关的基础知识，简单描述了高铁接触网中安全巡检的图像采集系统，以旋转双耳销钉为例分析了该领域缺陷检测任务的困难和挑战。之后分别介绍了卷积神经网络和生成对抗网络基础知识，包括原理、网络结构等。

第三章讲述了基于 CGAN 的缺陷生成模型，详细说明了模型的原理、架构、网络结构参数等技术内容，并进一步介绍了模型训练过程中的技术细节。最后是实验部分，展示并分析了缺陷生成的结果。

第四章的内容是缺陷检测。首先介绍了经过缺陷生成后搭建的旋转双耳销钉数据集，然后说明了检测实验的具体内容，包括实验组和对照组的检测结果、消融实验结果以及与异常检测 SOTA 算法的对比结果，最后从定性和定量两个维度展开分析，并对于模型的改进进行了讨论。

2 相关基础知识介绍

2.1 高铁接触网图像采集

2.1.1 图像采集系统简介

高铁接触网系统安全巡检中的图像采集工作需要特定的巡检车和采集装备，如图 2-1 所示。巡检车顶部安装高清摄像机，每一个相机安装角度不同，以捕获不同方向、不同区域的接触网设备图像，如图 2-2 所示。安全巡检分为静态评定和动态评定两种模式。静态评定是人工定点测量接触网设备的几何参数；动态评定是列车以相同速度在相同路段多次连续行驶一段距离，测定运行期间的各种数据，一般速度要低于 60km/h，以减少振动对接触网参数的影响。最后要综合静态和动态指标，对比人工测量和装置测量数据，以保证安全评定的准确性。同时，为了尽可能减少对正常列车行驶的影响，巡检工作一般安排在夜晚进行。



图 2-1 巡检车（左）和 4C 装置（右）
（图像由中国铁道科学院基础设施检测研究所提供）

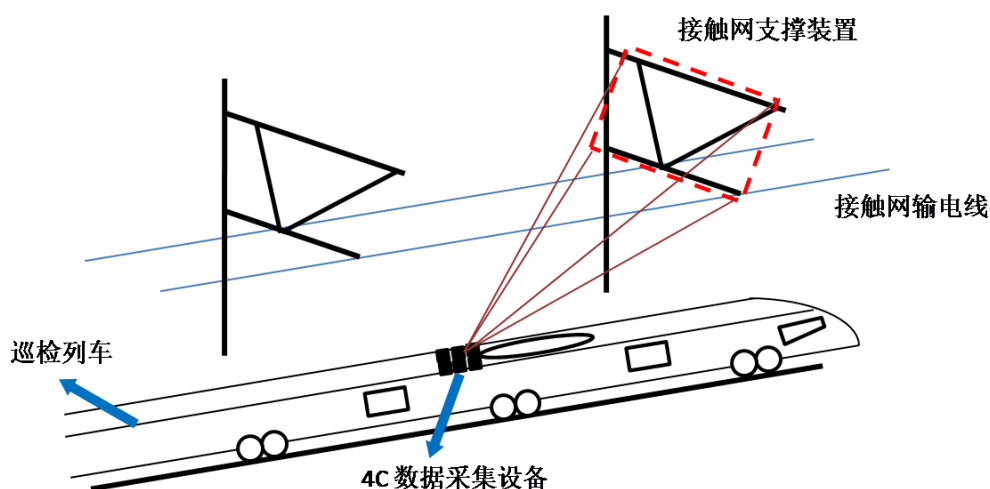


图 2-2 巡检车图像采集示意图

2.1.2 旋转双耳销钉异常类型及检测难度分析

旋转双耳销钉是接触网装置中重要的紧固件。安装时的操作不规范和列车行驶中产生的振动，都可能造成该零件的开口销部件松动脱落，这给列车的安全运行带来隐患。这对接触网支撑装置的结构稳定性提出了更高的要求，需要对销钉故障进行检测并采取措施排除。一般的检测算法是在局部元件中提取特征，以此实现目标零件的定位，并进一步进行缺陷分类。但旋转双耳销钉存在体积小，与其他零件相似度较高的特征，不易定位和分类。图 2-3 展示了横向和竖向旋转双耳销钉的正常和缺陷状态。为了方便实验，本文将竖向正常状态旋转双耳销钉、竖向异常状态（开口销缺失）旋转双耳销钉、横向正常状态旋转双耳销钉和横向异常状态旋转双耳销钉分别标记为 s_good、s_bad、h_good 和 h_bad。

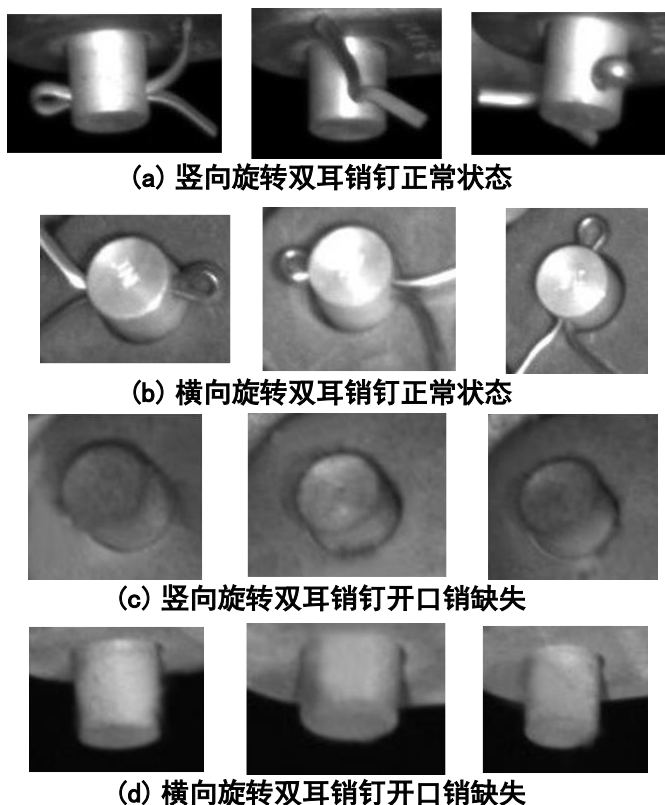


图 2-3 旋转双耳销钉状态

以旋转双耳销钉为例，可以分析高铁接触网零部件缺陷检测任务的难度：

(1) 缺陷样本缺失，数据集中正常样本为绝大多数，难以发挥基于有监督学习的检测器的优势。

(2) 零件种类众多，各零件几何状态、缺陷模式繁多。如旋转双耳销钉有横向和竖向两种，而开口销安装的角度、方向也有很大差别。传统的检测算法需要人工构建特征进行提取，难以适应接触网系统零部件缺陷种类繁多的情况。

(3) 巡检车捕获的原始图像中，各种零部件尺寸极小，如图 1-2 和图 1-3 所示。小物体检测本身就是目标检测方面的难点，目标定位难度较大使用小尺度的窗口进行定位时计算开销较大。

（4）不同零部件之间存在较高相似性，也可能存在重叠，增加了检测的难度。

2.2 卷积神经网络简介

卷积神经网络（Convolutional Neural Networks, CNN）是在传统神经网络模型中引入了卷积和池化操作而发展得到的模型。1998 年，LeCun 等^[41]使用 LeNet-5 在 MNIST 手写体数字数据集上达到了很好的效果。2012 年，Alex Krizhevsky 与 Geoffrey Hinton 设计的 AlexNet^[10]在 ImageNet 竞赛中以绝对优势夺得冠军。卷积神经网络在计算机视觉领域中取得了里程碑式的进展，相关的研究论文数量也呈现指数型增长，为机器学习揭开了新的篇章。

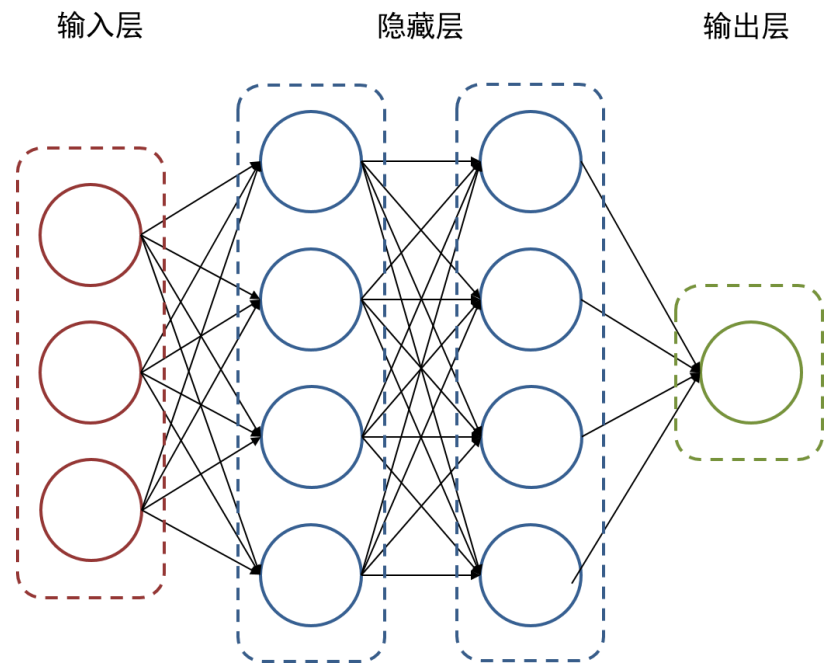


图 2-4 神经网络模型

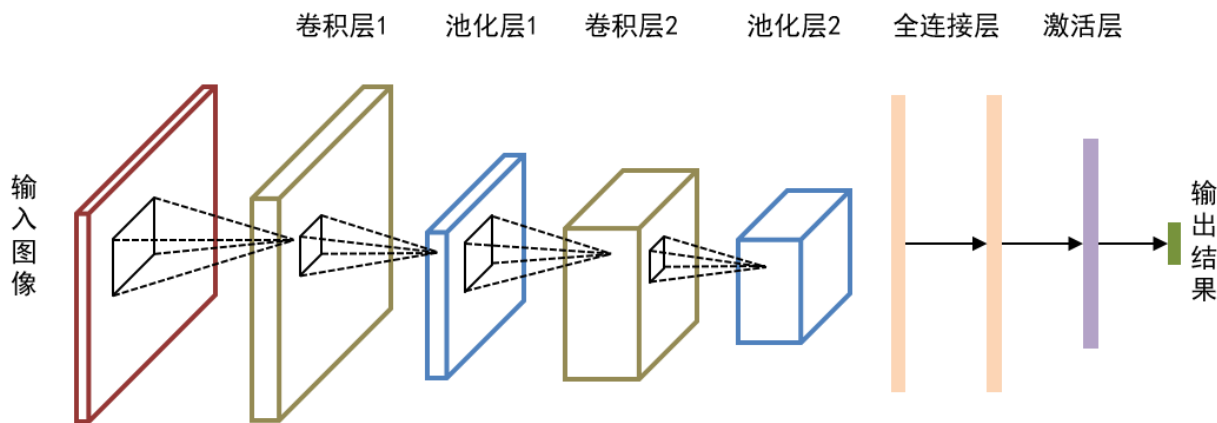


图 2-5 卷积神经网络模型

2.2.1 卷积神经网络原理

传统的神经网络模型包括输入层、隐藏层和输出层三个部分，每一层都采取全连接的

方式，如图 2-4 所示。卷积神经网络延续了层次模型结构，但每一层神经元只与前一层中一部分区域相连接，如图 2-5 所示。卷积神经网络内部主要由卷积层、池化层、全连接层和激活层构成。

(1) 卷积层

卷积层是网络的核心部分，由若干个卷积单元组成。每一个卷积单元相当于一个进行卷积操作的滤波器，对输入的图像的特定区域进行特征提取。同一层中所有卷积单元提取的特征组成特征图（feature map），作为卷积层的输出。卷积单元的参数通过前向传播进行优化，随着层数的加深，提取的特征更加复杂。卷积层是整个网络中计算开销最大的部分。

(2) 池化层

池化层中进行降采样操作，对输入的特征图进行降维。池化层中，特征图被分割为多个网格区域，在不改变特征的同时，每一个区域输出代表性最强的数值，减小特征图的尺寸，同时减少了参数数量和下一层的计算量。

(3) 全连接层

全连接层上由大量的神经元构成，上一层的数据转化为一维的向量，映射到每一个神经元上。全连接层将输入的特征图整合，输出为一个值，相当于网络中的“分类器”，实现了特征从隐藏空间到样本空间的映射。

(4) 激活层

线性模型对特征的表达能力不足，无法解决非线性问题。激活层中，激活函数在模型的表达中添加非线性因素，增强模型对特征的表达能力。常见的激活函数有 tanh、relu、sigmoid、softmax 等。

2.2.2 经典网络结构

除了上文提及的 LeNet-5 和 AlexNet，许多研究基于卷积神经网络，提出了网络层数更深、结构更复杂的模型。

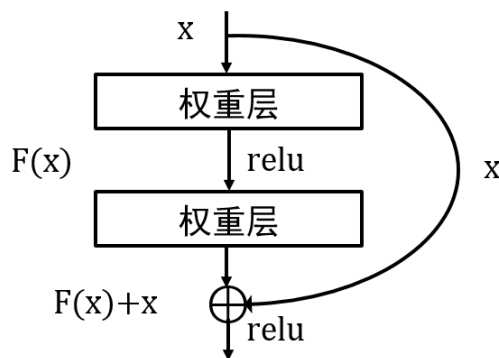


图 2-6 残差模块^[42]

(1) ResNet

随着网络层数的加深，模型可以提取更复杂的特征，理论上应该具有更好的表达能力。

但在实验中，研究人员发现，层数过深可能会遇到诸如梯度弥散和梯度爆炸等问题，导致深层网络的性能可能要逊于浅层网络的性能。针对此问题，文献^[42]提出了残差神经网络（Residual Networks, ResNet）。ResNet 核心是引入了残差模块，其结构如图 2-6 所示。残差模块将前一层的特征与本层的输出结果做对比，如果本层的特征表达能力低于上一层，则将上一层的权重参数传入下一层。残差模块的使用可以保证深层网络的性能不亚于浅层网络，可以充分发挥“深度”的优势。

（2） MobileNet

深度卷积神经网络虽然可以在各类视觉任务中取得出色的效果，但庞大的计算量不利于在工程实践中应用落地。2017 年，Google 研发团队提出了轻量级网络 MobileNet^[43]，可以植入移动端设备。MobileNet 提出了深度可分离卷积方法，先使用 3×3 的卷积核进行特征提取，再使用 1×1 的卷积核进行特征组合形成特征图，可以大幅度减少模型的计算量。在此基础上，文献^[44]引入了 ResNet 的残差模块与瓶颈设计，提出了 MobileNetv2。文献^[45]提出了 MobileNetv3，使用了 NAS（Neural Architecture Search）算法搜索网络中的参数，并添加了 Squeeze-and-Excitation 模块对特征进行校准。

（3） DarkNet

DarkNet 是目标检测算法 YOLO 系列的使用的网络架构，本质上是 ResNet 的一种变体，可以支持端到端的训练。YOLO9000^[16]提出 DarkNet19，YOLOv3^[17]使用 DarkNet53，其中数字代表网络的深度。

（4） 全卷积神经网络

全卷积神经网络（Fully Convolutional Networks, FCN）^[46]将卷积神经网络中的全连接层替换为卷积层，其中卷积核的尺寸为 1×1 ，可以得到不同尺寸的输出。由上文可知，卷积层的输出为一个特征图，每一个值对应输入图像中一个区域的表达。使用核尺寸 1×1 的卷积层代替全连接层后，输出的特征图中每一个值代表了输入图像中一个区域的分类结果。根据这个特性，FCN 可以分辨的图像的前景和背景，在语义分割领域表现优异。

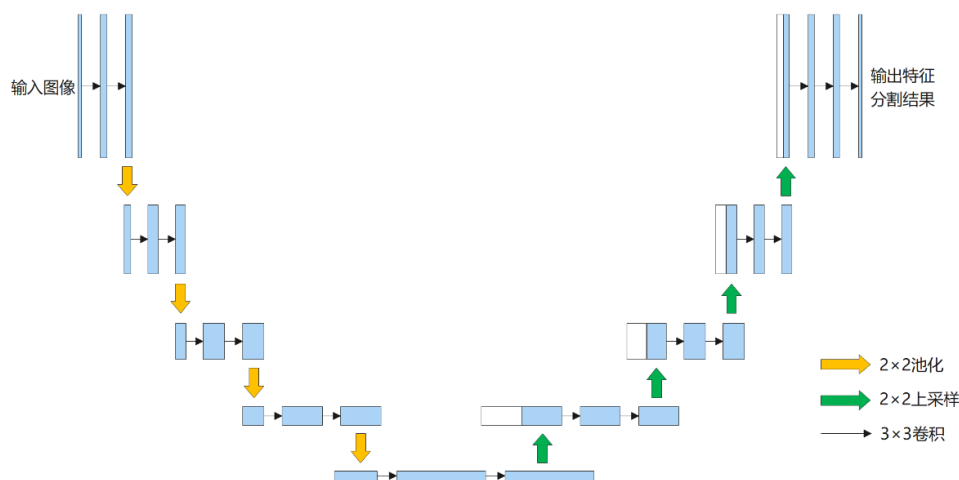


图 2-7 U-Net 示意图^[47]

(5) U-Net

U-Net^[47]由两个对称的 FCN 网络构成，如图 2-7 所示，因形似字母“U”而得名，其本质是一个 Encoder-Decoder 的编码器模型。U-Net 模型中，前半部分通过池化层（下采样）进行特征提取，不断扩大感受野，感知低频信息，可以提取到更复杂的特征；后半部分对特征图进行上采样操作，保持输出与输入尺寸的一致；左右两部分通过一个 Bottleneck 结构连接。同理与 FCN，U-Net 在语义分割领域取得了非常好的效果。

2.3 生成对抗网络简介

生成对抗网络（Generative Adversarial Networks, GAN）^[24]是 Ian Goodfellow 在博弈论的启发下提出的学习模型，通过对抗博弈的方式进行非监督学习。经典的 GAN 模型包括生成器（Generator, G）和判别器（Discriminator, D）两个主要的组成部分，如图 2-7 所示。生成器从输入的噪声中模拟数据的分布，并参照判别器给出的反馈信息进行调整，来生成更趋近于真实分布的数据，使得判别器也无法分辨。判别器的主要任务是进行二分类，区分生成器生成的模拟数据和真实数据。生成器输入特定噪声，利用极大似然估计通过先验分布 $P_y(y)$ 模拟真实数据的特征分布 P_g ，判别器的输入是真实数据和生成数据，输出 $D(x; \theta_d)$ ，表示样本数据为真实数据的概率。在训练过程中，生成器和判别器不断博弈，使生成器尽可能模拟到真实数据的分布，判别器尽可能区分模拟数据和真实数据，最终两个网络达到纳什平衡，使判别器无法判断出送入的样本是否为真实数据。GAN 的目标函数表示为：

$$\min_G \max_D V(G, D) = \mathbb{E}_{x \sim P_x(x)} [\log D(x)] + \mathbb{E}_{y \sim P_y(y)} [1 - \log D(G(y))],$$

其中， $P_x(x)$ 是真实的数据分布， $\log D(x)$ 是判别器 D 对真实数据的判断结果， $1 - \log D(G(y))$ 表示判别器对生成器生成数据的判断结果。

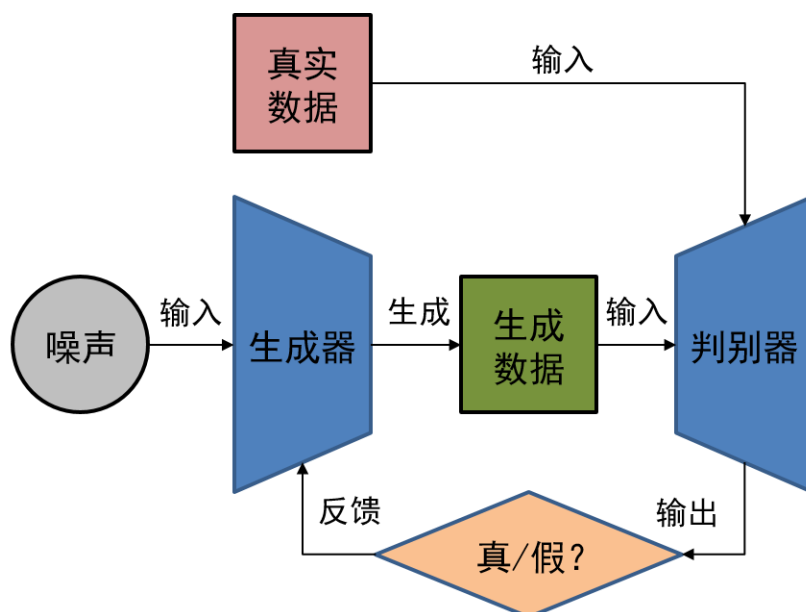


图 2-8 生成对抗网络基本架构

模型训练时, 判别器和生成器是交替训练的。训练判别器时, 将生成器固定, 这时的目标是使 $1 - \log D(G(y))$ 尽可能大, $\log D(x)$ 尽可能大。训练生成器时, 将判别器固定, 这时的目标是使 $1 - \log D(G(y))$ 尽可能小。最终达到纳什均衡时, 理想状态下, 判别器输出为 0.5, 即 $P_x(x) = P_y(y)$ 。

3 基于条件对抗生成网络的缺陷样本生成

3.1 条件生成对抗网络

3.1.1 模型原理

传统的 GAN 网络可以完成缺陷样本的生成，但由于输入的噪声随机，而生成目标特定（即只需要消除双耳销钉开口销），所以存在收敛速度较慢、生成效果不佳的问题。针对这种不足，本文使用条件生成对抗网络 CGAN 来完成样本生成任务。CGAN 在 GAN 基础上，针对随机噪声不确定性的限制做出了突破。在训练过程中，生成器的输入增加了带标签的样本作为“条件（condition）”，为生成器模拟数据分布提供了参照，可以加速模型的收敛并提升训练效果^[4]，本质上将生成器的训练从无监督学习转变为有监督学习。

在 CGAN 中的生成器，给定一个输入噪声 $p_y(y)$ 和额外信息 z ，之后将两者通过全连接层连接到一起，作为隐藏层输入。同样地，在判别器中输入图像 x 和额外信息 z ，也将连接到一起作为隐藏层输入，生成器和判别器的损失函数分别表示为：

$$\min_G l_G = \mathbb{E}_{y \sim p_y(y)} [1 - \log D(G(y)|z)], \quad (3.1)$$

$$\max_D l_D = \mathbb{E}_{x \sim p_x(x)} [\log D(x|z)] + \mathbb{E}_{y \sim p_y(y)} [1 - \log D(G(y)|z)]. \quad (3.2)$$

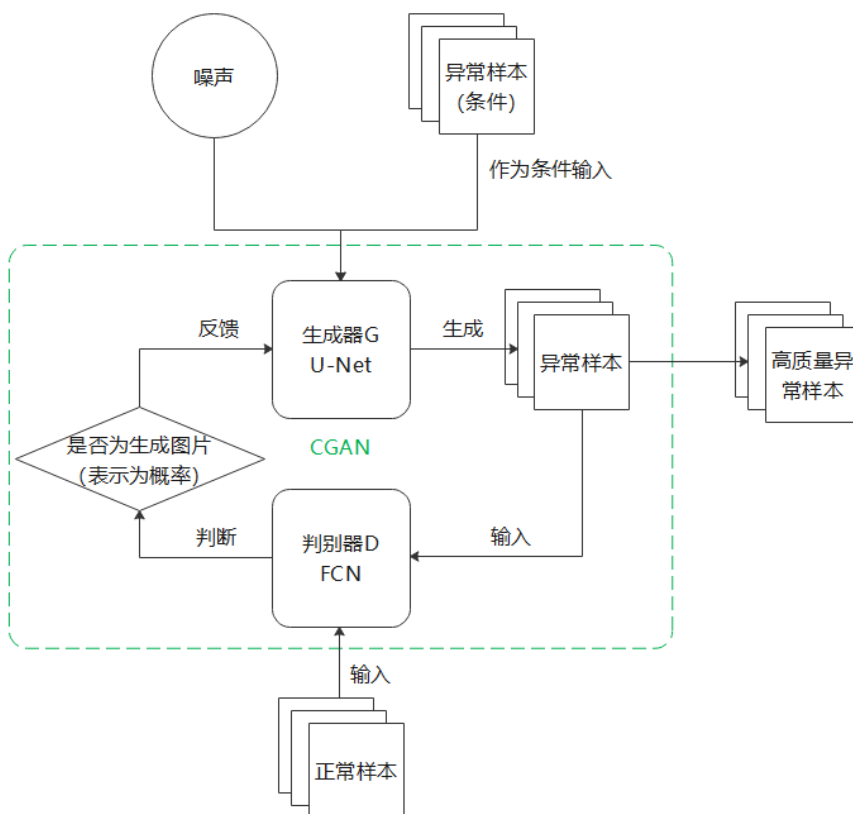


图 3-1 基于 CGAN 的缺陷生成模型结构

3.1.2 网络结构

(1) 生成器结构

对于生成器，本文使用 U-Net 网络搭建。U-Net 可以准确地捕捉异常样本和正常样本之间的差异，对特定目标的生成任务有较好的效果。以旋转双耳销钉为例，缺陷样本生成目的是抹除开口销部件的同时尽可能不向原始图像加入多余的特征，从而使生成效果更自然。生成任务的实质可以理解为将双耳销钉的开口销部分和其余部分分割，与语义分割任务类似。所以在语义分割方面表现卓越的 U-Net 对双耳销钉场景中缺陷样本生成有较大的优势，能更好地对开口销缺失这一特征进行捕捉，而不是对背景和其他零件部分进行处理。

本文生成器所使用的 U-Net 结构如图 3-2 所示，左侧部分各包含 4 个 block，每个 block 由 2 个卷积层和 1 个池化层构成，进行下采样操作，通道数依次为 64、128、256 和 512，没有设置全连接层。右侧部分则为左侧部分的镜像，将池化层替换为上采样操作，左右两部分通过一个通道数为 1024 的 Bottleneck 连接。

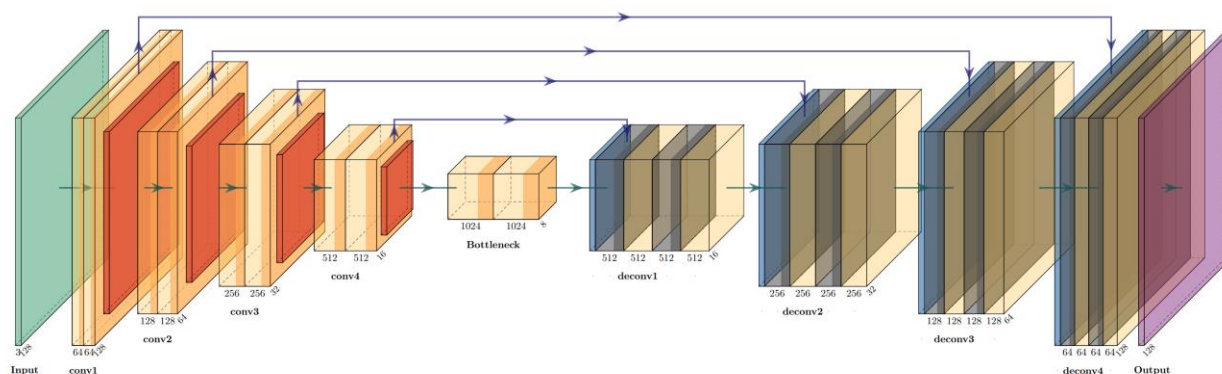


图 3-2 基于 U-Net 的生成器结构

(2) 判别器结构

对于判别器，本文使用 FCN 搭建，其结构如图 3-3 所示。FCN 在语义分割方面也有着出色的表现，正如上文讨论，对双耳销钉缺陷样本的生成有着较好的适应性；又因为判别器只负责二分类任务，所以 FCN 完全可以满足要求，又保证了计算的高效，无需使用像 U-Net 一样的复杂网络。

在判别器中，因为输入包括原始数据和生成器生成的数据，所以 FCN 的输入尺寸为 $256 \times 256 \times 6$ （将两张 $256 \times 256 \times 3$ 的图像做拼接）。判别器使用的 FCN 包括五个 block，block1 将输入图像分割为 $256 \times 256 \times 3$ 的两份数据分别做卷积处理。block2~block4 都包括 2 个卷积层和 1 个池化层，通道数依次递增为 64、128 和 256，block5 将两张图像对应的特征图拼

接为 $32 \times 32 \times 512$ ，最后输出判别结果，即是生成图像还是原始图像。

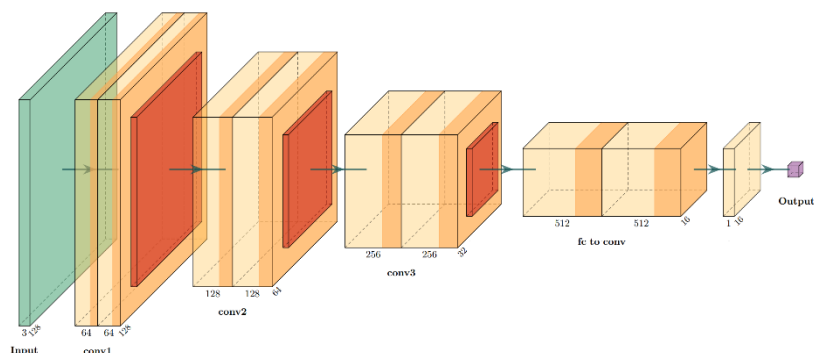


图 3-3 基于 FCN 的判别器结构

3.2 模型训练

在生成缺陷样本的过程中，首先用正常数据训练了一个 Faster R-CNN 模型，导出的模型用于从原始图像中定位正常的旋转双耳销钉，并裁剪为小图，标记为正常样本，用于训练判别器。由于缺陷样本的数量非常少，手动裁剪处于缺陷状态的销钉，然后对这些缺陷状态的裁剪图像进行增强处理，包括图像旋转 (60° 、 90° 和 180°)、水平镜像翻转、锐化 (laplacian 算子)，将处理后的数据添加到缺陷类中，之后输入到 CGAN 的生成器中，以在训练中提供一定的参考或监督。训练流程如下：

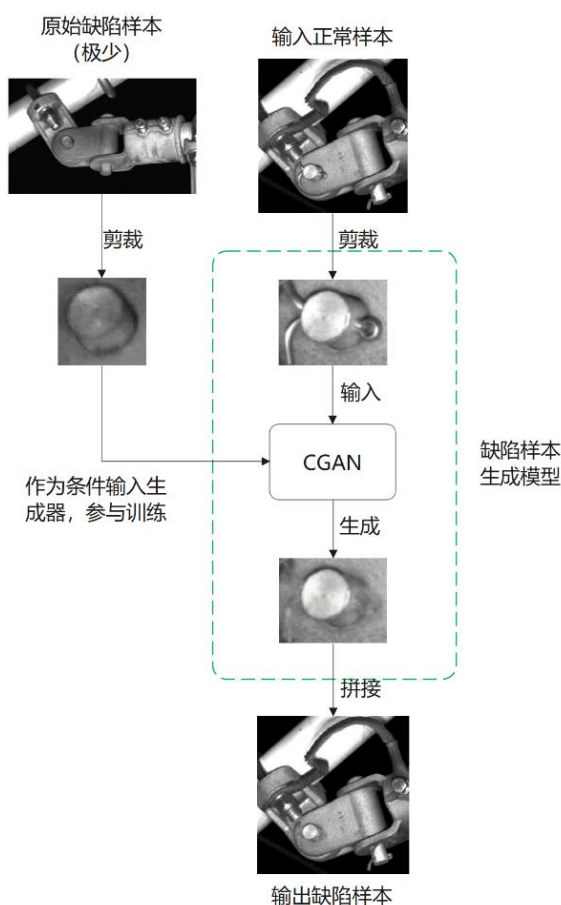


图 3-4 端到端的缺陷生成模型

(1) 初始化训练数据集。将 h_good 、 h_bad 、 s_good 和 s_bad 样本分别添加到集合中 S_1 、 S_2 、 S_3 和 S_4 中。

(2) 训练 CGAN 模型。固定判别器，将随机噪声和 S_2 数据输入到生成器中，更新 l_G ；固定生成器，将生成的样本和真实样本输入到判别器，更新 l_D 。判别器的输出结果作为反馈传递给生成器，再进入新一轮对抗学习。

(3) 每一轮训练后，从生成器生成的缺陷样本中挑选出高质量的生成图像，添加到 S_2 中。

(4) 导出生成模型 G_h 。

(5) 在 S_3 和 S_4 中上重复步骤 (2)、(3)，最终导出模型 G_s 。

缺陷生成算法伪代码如算法 3-1 所示。最终训练好的模型为端到端的缺陷生成模型，如图 3-4 所示。

伪代码 3-1 CGAN 训练过程

输入：

S_1 : h_good 样本数据; S_2 : h_bad 样本数据;

S_3 : s_good 样本数据; S_4 : s_bad 样本数据;

$epochs$: 训练轮数; n : 样本批次大小;

$\alpha = 0.0001$; $\beta_1 = 0$; $\beta_2 = 0.9$; ω_0 : 判别器初始参数; θ_0 : 生成器初始参数;

```

1.  $j = 1$ ;
2. while  $j \leq epochs$  do
3.   for  $i = 0, \dots, n$  do
4.     采样真实数据  $x \sim S_1$ , 生成器输出数据  $z \sim G_z$ ;
5.     更新判别器损失函数  $L_D$ ;
6.      $\omega_j \leftarrow Adam(\omega_{j-1}, \alpha, \beta_1, \beta_2)$ ;
7.   end for
8.   for  $i = 0, \dots, n$  do
9.     采样条件数据  $z \sim S_2$ , 随机噪声  $y \sim U[0, 1]$ ;
10.    更新生成器损失函数  $L_G$ ;
11.     $\theta_j \leftarrow Adam(\theta_{j-1}, \alpha, \beta_1, \beta_2)$ ;
12.  end for
13.   $j += 1$ ;
14. end while
15. 导出模型  $G_h$ ;
16. 在  $S_3$  和  $S_4$  上重复步骤 1-10, 导出模型  $G_s$ .
```

3.3 实验结果与分析

本文收集了 6204 张自然状态下的旋转双耳销钉数据样本，其中仅有 6 张为缺陷样本。本文首先对包含旋转双耳销钉的原始图像进行生成，输入的尺寸调整为 128×128 ，训练 50 epoch，生成结果如图 3-5 所示。生成的图像非常模糊，对背景做了非常大的处理，而对旋转双耳销钉本身的生成不足。经过此实验，本文决定采用先裁剪、后生成的方法。

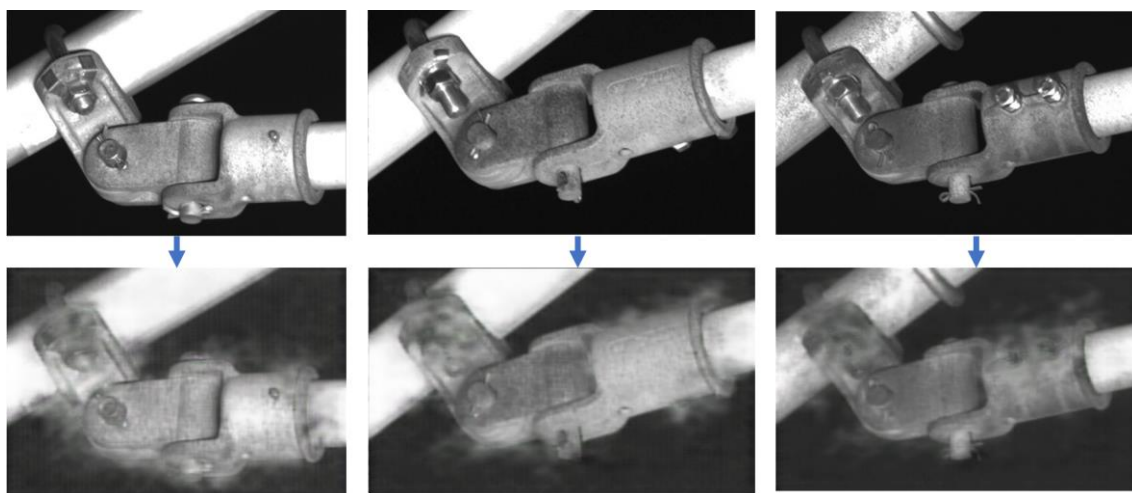


图 3-5 直接输入大图的生成结果

鉴于直接对大图处理效果不佳，本文按照 3.2 节内容先对图像进行了剪裁，然后输入训练 CGAN 训练。调整输入尺寸为 64×64 ，训练 25 epoch，loss 曲线如图 3-6 所示，训练过程中的生成结果如图 3-7 所示，最终的生成结果如图 3-8 所示。将生成的小图重新拼接回原始图像，其结果如图 3-9 所示。由下图可见，先裁剪后生成的方法是可行的。生成器在抹除开口销的同时并未改变零件其他区域和背景信息，可以生成肉眼判别为异常的缺陷样本。将生成的缺陷小图拼接回原始图像，没有明显的拼接痕迹，与自然状态下的缺陷样本具有极高的相似性。

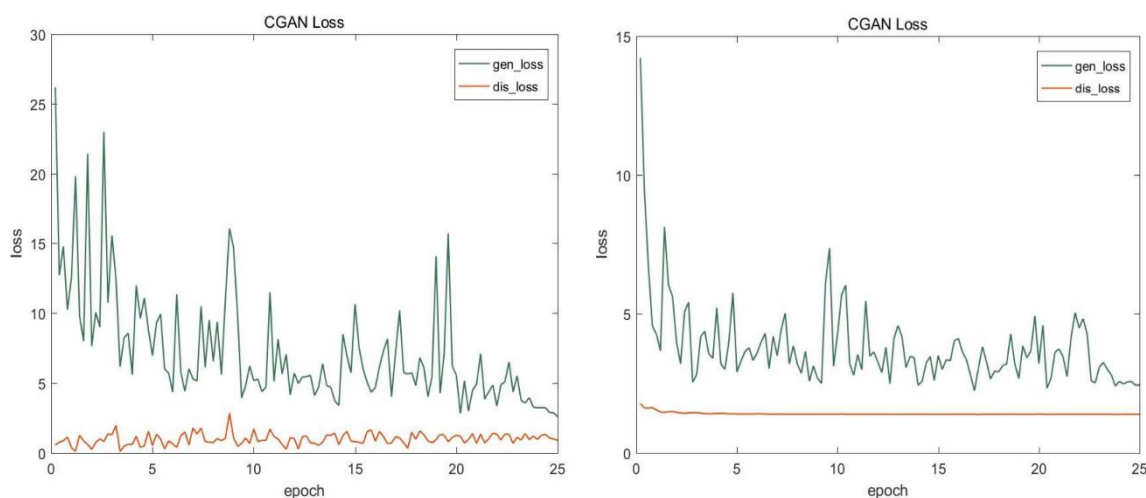
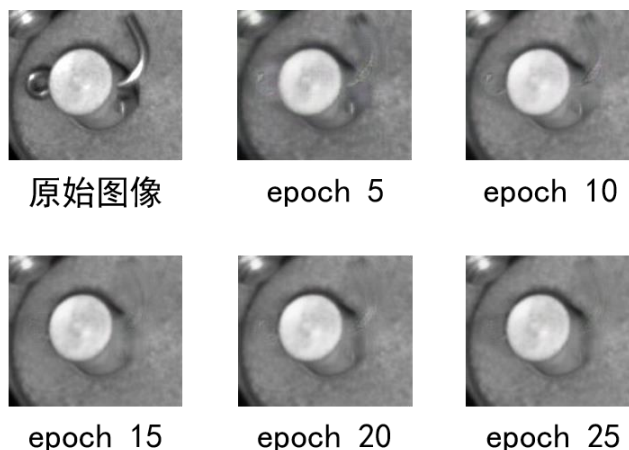
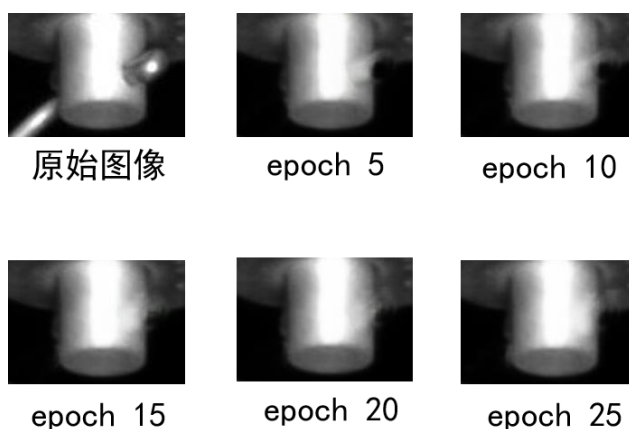


图 3-6 CGAN loss 曲线（左图为 G_h loss，右图为 G_s loss）



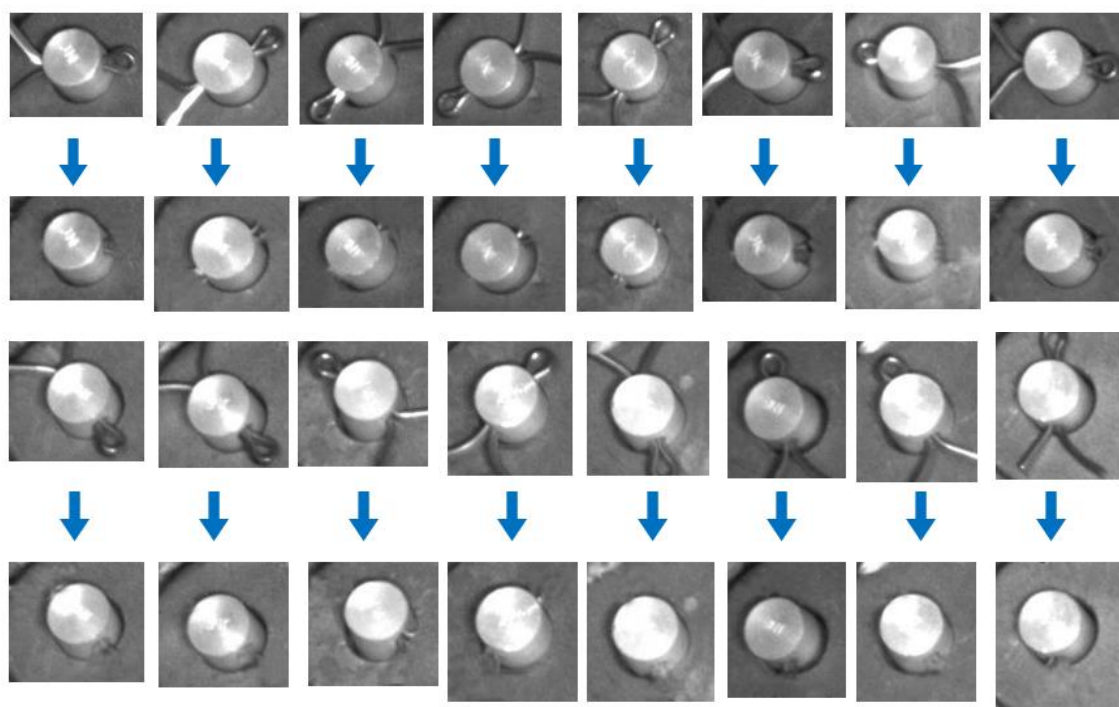
(a) 横向旋转双耳销钉训练过程



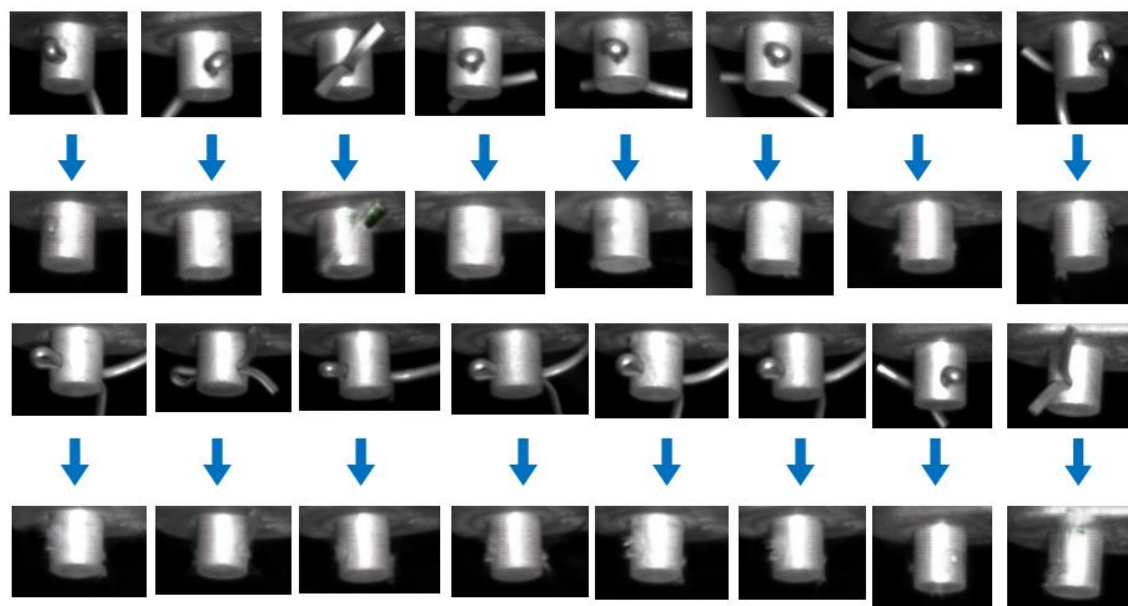
(b) 竖向旋转双耳销钉训练过程

图 3-7 CGAN 训练过程

将正常样本输入导出的缺陷生成模型，可以将数据重构为大量的异常样本，来弥补原始数据集中正负样本不平衡的问题。在此基础上，本文构建了一个平衡的数据集，并对其中的销钉区域进行标记，整理为 PASCAL VOC 的形式，可以用于目标检测、图像分类等视觉任务。详情可见第 4.1 节。



(a) 横向双耳销钉小图生成结果



(b) 竖向双耳销钉小图生成结果

图 3-8 旋转双耳销钉小图生成结果

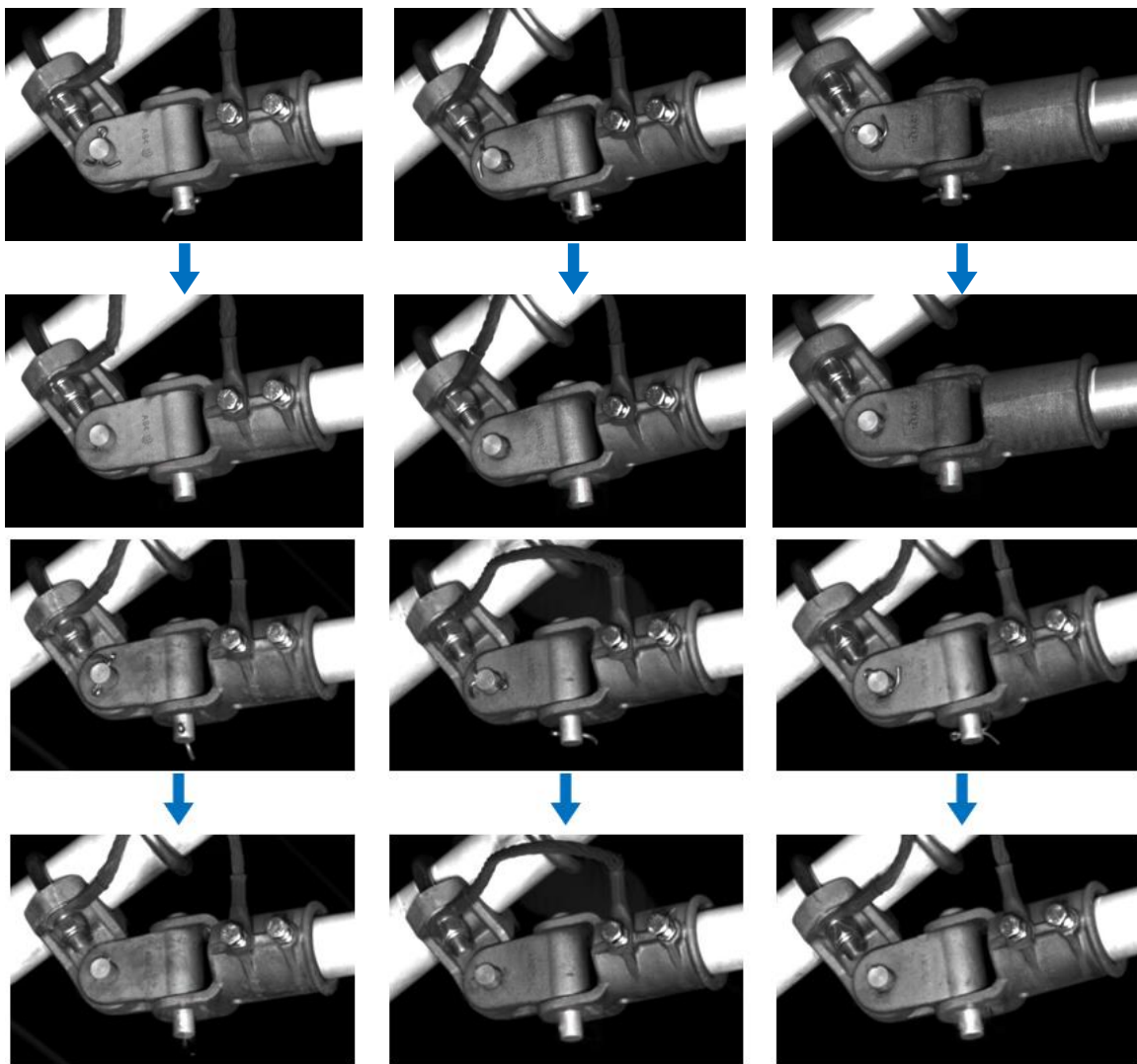


图 3-8 生成的缺陷小图拼接回原始图像

4 旋转双耳销钉缺陷检测

4.1 数据集

利用训练得到的缺陷生成模型，本文对正常样本进行重构，从中选取了符合人眼视觉要求的缺陷样本，与正常样本一起构成平衡的旋转双耳销钉数据集。为了方便说明，定义如下概念：

Dataset1: 包含正常样本、原始缺陷样本和生成的缺陷样本的平衡的旋转双耳销钉数据集；

Dataset2: 只包含原始图像的不平衡缺陷样本数据集；

Model1: 在 dataset1 上训练得到的目标检测模型；

Model2: 在 dataset2 上训练得到的目标检测模型。

Dataset1 与 dataset2 的数据规模如表 4-1 所示，二者都按照 6：2：2 的比例划分训练集、验证集与测试集。

表 4-1 检测数据集规模（单位：张）

	dataset1	dataset2
正常样本	573	1005
原始缺陷样本	6	6
生成缺陷样本	432	0
总计	1011	1011

4.2 实验结果与分析

4.2.1 评价指标

目标检测结果使用平均精度均值（mean average precision，mAP）进行定量评价，需要 TP (True Positive)、FP (False Positive)、FN (False Negative) 和 TN (True Negative) 四个指标。如表 4-2 混淆矩阵所示，TP 表示真实正样本被检测为正样本的结果，即正常样本中检测为正常样本的个数；FP 是指真实负样本被检测为正样本的个数，即缺陷样本被检测为正常样本的个数；FN 是真实正样本被检测为负样本的个数，即正常样本被检测为缺陷样本的个数；TN 是指真实负样本被预测为负样本的个数，即缺陷样本被检测为缺陷样本的个数。

表 4-2 混淆矩阵

	预测正样本	预测负样本
正样本	TP	FN
负样本	FP	TN

Precision-recall（P-R）曲线通常用于评估定位模型的性能。Precision(P)是精准度，recall(R)代表召回率，分别表示为公式 4 和公式 5。P-R 曲线下的面积被认为是每个类别的平均精度（Average Precision），如公式 6 所示。mAP 用于计算所有类别的 AP 平均

值，如公式 7 所示。mAP@0.5，mAP@0.75 和 mAP@[0.5, 0.95] 分别代表置信度阈值范围[0.5, 1]、[0.75, 1]和[0.5, 0.95]时 mAP 数值。

$$P = \frac{TP}{TP + FP} \times 100\%,$$

(4.1)

$$R = \frac{TP}{TP + FN} \times 100\%,$$

(4.2)

$$AP = \int P(R)dR,$$

(4.3)

$$mAP = \frac{1}{k} \sum_{i=1}^n AP_i.$$

(4.4)

为了与异常检测领域 SOTA 算法进行对比，本文使用 AUROC (Area Under the Receiver Operating Characteristic Curve) 评估分类器性能。AUROC 的几何意义是 ROC (Receiver Operating Characteristic Curve) 曲线下放的面积，也表示随机选择正样本的预测值大于负样本的预测值的概率。因此，可以用公式 8 中表示。

$$AUROC = \frac{\sum pred_{pos} > pred_{neg}}{Num_{pos} \times Num_{neg}} \times 100\%.$$

(4.5)

表 4-3 缺陷检测结果 (%)

算法模型	网络骨架	mAP@0.5		mAP@0.75		mAP@[0.5, 0.95]	
		model1	model2	model1	model2	model1	model2
SSD	MobileNetv2	65.6	42.1	59.6	40.1	63.6	40.6
Faster R-CNN	ResNet50	80.2	78.5	74.1	69.0	77.7	76.9
Faster R-CNN	ResNet101	86.1	78.4	80.3	72.1	85.3	78.2
YOLOv3	DarkNet53	97.7	78.2	70.4	67.2	96.7	76.3

4.2.2 目标检测与消融实验

本文分别使用 dataset1 和 dataset2 训练 SSD、Faster R-CNN 和 YOLOv3 模型，然后在 dataset1 上做了测试，以评估它们对缺陷样本的检测性能。为了进一步分析生成的缺陷样本在检测中的作用，对性能最好的目标检测模型上进行消融实验，model1 和 model2 分别在 dataset1 和 dataset2 上进行了测试。其中，YOLOv3 使用 DarkNet 深度学习框架接口，Faster R-CNN 和 SSD 使用 Tensorflow 目标检测接口。

如表 4-3 所示，使用 dataset1 训练的 YOLOv3 表现最好，在 mAP@0.5 和 map@[0.5, 0.95] 分上取得了最高分，分别为 97.7%和 96.7%，但 mAP@0.75 分数要低一些，只有 70.4%。相比之下，使用不平衡的 dataset2 训练的 YOLOv3 模型只有 78.20% mAP@0.5、67.2% mAP@0.75 和 76.3% mAP@[0.5, 0.95]的表现。至于 Faster R-CNN，基于 ResNet50 和

ResNet101 的 model1 的所有 mAP 分数都要高于 model2，但差距并不算非常显著。基于 ResNet101 的 model1 的 mAP@0.5、mAP@0.75 和 mAP@[0.5, 0.95] 的值分别比 model2 高 7.7%、8.2%和 7.1%。而基于 ResNet50 的 model1 和 model2 性能的差异导致的 mAP 分数提升较少。这可能是因为，在 Faster R-CNN 的 RPN 阶段，正负样本会按照一定的比例进行选取，最终的回归框也会经过 NMS 算法处理。因此，数据不平衡的影响将减少。此外，SSD 的 model1 和 model2 得分最低。SSD 的 model1 只有 65.6% mAP@0.5, 59.6% mAP@0.75 和 63.6% mAP@[0.5, 0.95]，而 SSD 的 model2 只得到 42.1% mAP@0.5, 40.1% mAP@0.75 和 40.6% mAP@[0.5, 0.95]。总体来看，使用 dataset1 训练的检测模型比使用 dataset2 训练的检测模型普遍具有更高的 mAP 分数，而 model2 的性能表明，SSD、Faster R-CNN 和 YOLOv3 等检测算法在缺陷检测任务中在直接使用正常数据训练很难取得良好的效果。因此，解决数据不平衡问题对使用目标检测算法处理缺陷检测任务是非常有效的，生成的缺陷样本有助于提高检测模型的性能。

表 4-4 YOLOv3 AP@0.5 (%) 消融实验结果

状态类型	model1		model2	
	dataset1	dataset2	dataset1	dataset2
h_bad	100	97.2	60.2	100
h_good	96.1	83.3	92.1	99.5
s_bad	99.0	100	76.9	100
s_good	95.8	97.2	83.5	97.1
平均值	97.7	94.6	78.2	99.2

为了进一步探究缺陷样本的存在对检测结果的影响，本文设计了消融实验，分别在 dataset1 和 dataset2 上测试了 YOLOv3 的 model1 和 model2，并评估了旋转双耳销钉的每个状态的检测结果，如 4-4 所示。Model1 在两个数据集中都获得了高分，分别为 97.7%和 94.6% mAP。而 model2 仅在 dataset2 中表现良好，mAP 为 99.2%。对于旋转双耳销钉的不同状态，model1 在大多数状态下都有较高的 AP 值，但在 dataset2 中的 h_good 样本的检测结果仅达到 83.3%。在 dataset1 中，model2 在对正常样本检测的 AP 值达到了 92.1%和 83.5%，但在对缺陷状态旋转双耳销钉的检测中仅有 60.2%和 76.9%的表现。然而，model2 在 dataset2 中表现出色，对于缺陷样本的检测准确率达到了 100%。考虑到 dataset2 中仅有 6 张缺陷样本，4 张在训练集，2 张在测试集，异常检测的高分并不能令人信服。与 dataset1 的结果相比，使用不平衡数据集 dataset2 训练的 model2 可以有效识别正常数据，但检测异常样本的能力偏弱。这一证据解释了 model2 在缺陷检测方面的低准确率主要是由于缺陷数据的稀有性造成的。如果能提供足够多的缺陷数据，像 YOLOv3 这样的检测模型就能得到很好的结果。因此，本文提出的缺陷生成方法可以有效地提高目标检测模型在识别旋转双耳销钉缺陷状态方面的性能。

4.2.1 异常检测 SOTA 算法对比试验

为了进一步验证本文提出的检测架构的有效性，设计了 YOLOv3 检测结果与异常检测领域 SOTA 算法的对比实验。为了方便实验，该实验只对小图进行分类测试，因为部分 SOTA 算法只能处理小尺寸的图像。对于目标检测的结果，将定位成功并识别正确的样本视为分类正确，而未能定位样本或识别错误的视为分类错误。

表 4-5 显示，使用 dataset1 训练的 YOLOv3 model1 可以达到与 SOTA 算法媲美的性能，例如 GANomaly、DifferNet 和 CS-Flow。AnoGAN 和 f-AnoGAN 通过评估两个图像的距离来检测异常，而鉴于旋转双耳销钉正常状态的多样性，两个正常样本之间的异常距离也可能较大；而相对的，开口销在样本中所占像素比例较小，正常样本与异常样本之间的距离可能很小。计算 AnoGAN 的旋转双耳销钉检测的异常分数 (anomaly score)，其结果如图 4-2 所示。可以看到，正常样本和异常样本之间分布重叠较多，Wasserstein 距离较小，证明了这一解释的合理性。GANomaly 具有更高的鲁棒性，在这个缺陷检测任务中表现更好。GANomaly 在训练阶段，会将输入的图像编码变换到隐空间，而 AnoGAN 只在推断阶段才有，所以检测效果逊于 GANomaly。DifferNet 和 CS-Flow 可以有效感知正常和异常数据之间的特征差异，AUROC 得分较高，分别为 96.2% 和 99.1%。YOLOv3 model1 和 model2 的性能如上文所述。

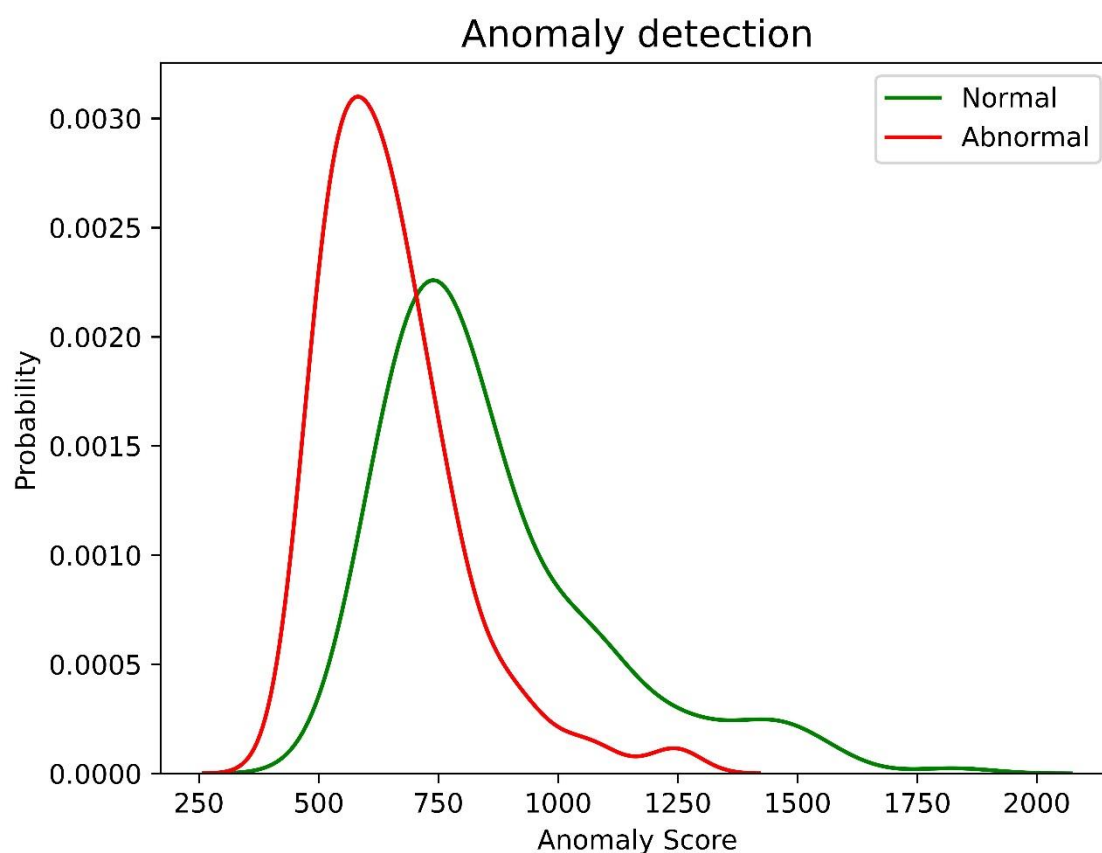


图 4-2 基于 AnoGAN 的旋转双耳销钉缺陷样本和正常样本的异常分数

表 4-5 YOLOv3 与 SOTA 算法分类 AUROC 对比结果 (%)

异常检测方法	AUROC
AnoGAN	70.3
f-AnoGAN	75.1
GANomaly	92.0
DifferNet	96.2
CS-Flow	99.1
YOLOv3 model2	63.3
YOLOv3 model1	97.6

4.3 改进思路讨论

(1) 定位与分类分步进行

本文先使用 CGAN 生成缺陷样本，解决数据不平衡问题，之后在平衡的数据集上训练目标检测模型，完成了缺陷识别任务。这是因为，虽然目前的异常检测算法已经可以在只输入正样本的情况下识别异常特征，但正如上文讨论的那样，大部分的 SOTA 算法通常处理小尺寸的目标，无法在原始图像中实现对目标的定位。实际情况中，巡检车上安装的设备所直接采集的是具有高像素值的图像，同时存在多种零部件，不能直接作为这些算法模型的输入。因此，为了实现快速、准确的自动化安全巡检，从原始图像中定位不同零部件是必要的。文献^[40]也讨论了这个问题，即便训练得到了较好的异常分类模型，但无法直接应用到安全巡检过程中。

但是，本文先生成缺陷样本，然后训练目标检测模型的方法也会带来较大的计算开销，最终导出的检测模型体量较大，难以植入移动设备。所以，可以优化过程，将最终的目标检测环节分为定位和分类双阶段进行。而 CGAN 中判别器的任务与分类器类似，可以参照文献^[48]的思路，使分类器参与 CGAN 的训练，或者将异常检测中 SOTA 算法应用到此处。这样一来，只需要再训练一个类似 RPN 的定位器，而无需训练一整个目标检测模型，如 YOLOv3。在此基础上，可以实现一个端到端缺陷检测模型，如图 4-1 所示。

(2) 生成模型优化为多分支（多任务）模型

在缺陷生成阶段，本文将旋转双耳销钉横向和竖向分为两种类型处理，导出 G_h 和 G_s 两个生成模型。文献^[49]提出了视觉领域的一个多任务架构，证明了多分支联合训练可以提高模型的效率，同时也可能提升准确率。由于横向和竖向销钉的生成是类似的任务，使用相同的网络结构，所以 G_h 和 G_s 可以合并为一个双分支模型，减少总体的计算开销和参数体量，其架构如图 4-2 所示。

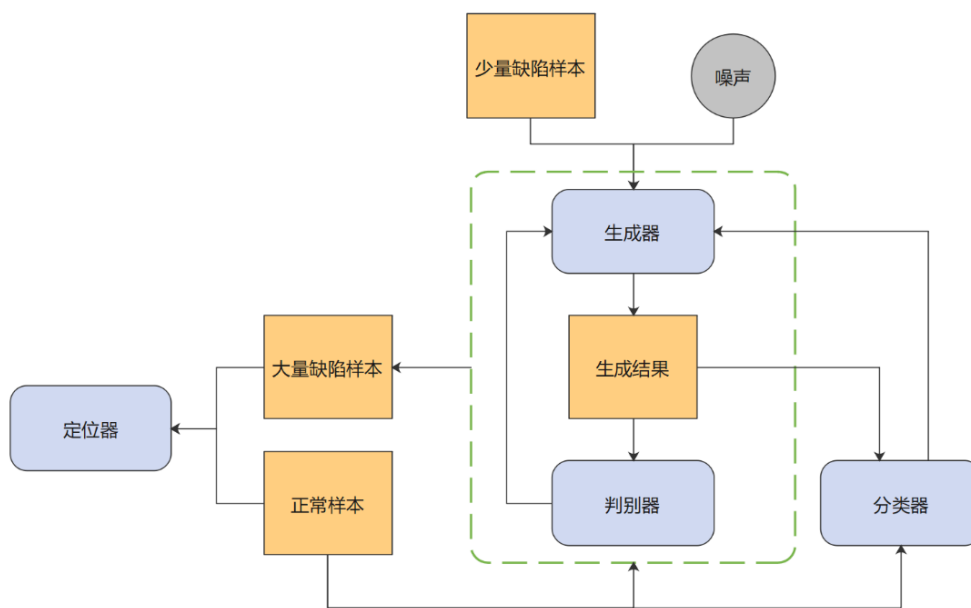


图 4-3 改进思路 1 (定位与分类分布进行)

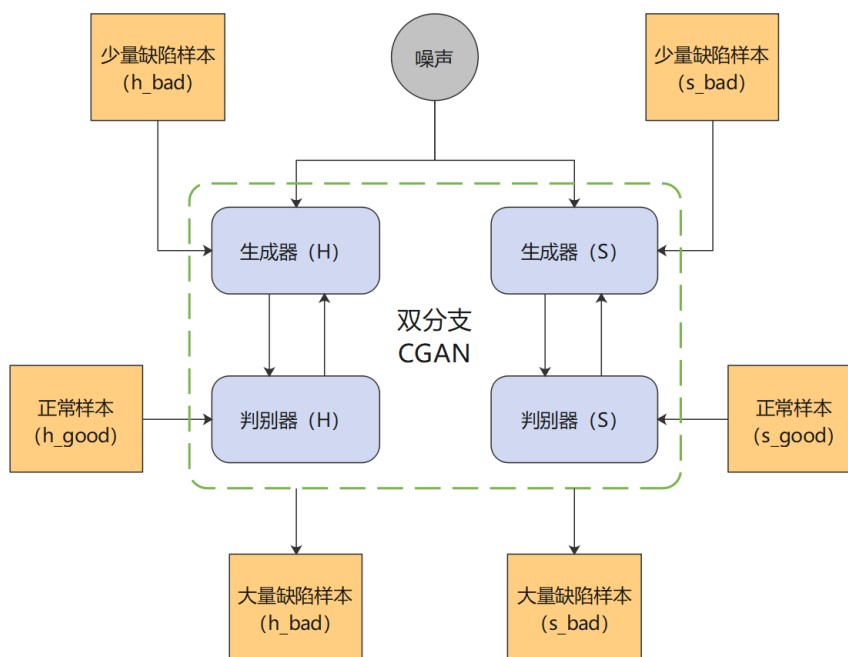


图 4-4 改进思路 2 (双分支模型)

总 结

本文针对高铁接触网系统中缺陷样本缺失、缺陷识别难度大的问题，提出了一种缺陷生成方案，搭建了一个基于条件生成对抗网络的缺陷样本生成模型，可以将正常状态的数据重构为异常状态，并在旋转双耳销钉零件上进行实验，取得了较好的生成与检测结果。本文的主要工作内容与贡献如下：

(1) 综述并分析了国内外在目标检测、异常检测领域的研究现状，着重研究了高铁系统中零部件缺陷检测的相关工作，进一步分析了高铁接触网系统中缺陷检测的难点。

(2) 介绍了高铁接触网系统安全巡检模式，包括图像采集设备、采集方式；介绍了卷积神经网络等深度学习基础内容，介绍了 GAN 和 CGAN 的原理、结构，为后续算法和实验的开展提供理论基础。

(3) 搭建了基于 CGAN 的端到端缺陷生成模型，先将目标零件（旋转双耳销钉）从背景图像中裁剪，输入生成器，将正常数据重构为异常数据，并拼接回原始图像。最后导出的模型可以生成符合人眼视觉要求的缺陷样本。本文利用该方法搭建了一个平衡的旋转双耳销钉数据集，并规范为 PASCAL VOC 的格式，可以用于目标检测等 high-level 等任务。

(4) 使用生成的数据集和原始数据集分别训练不同的目标检测模型，并对其检测结果进行了定量分析。实验数据显示 YOLOv3 的检测效果最好。设计消融实验，证明了生成的缺陷数据对检测结果的影响，可以有效提升检测模型的性能。本文进一步设计与异常检测领域 SOTA 算法的对比试验。实验结果证明了本文提出的生成方法的有效性和可行性。

虽然本文的模型可以完成接触网系统中旋转双耳销钉零部件的检测，但仍由很多需要改进和发展之处。一方面，如 4.3 节所探讨的，模型本身需要进一步优化。另一方面，高铁接触网领域零部件种类众多，缺陷类型多样，需要做更多的工作将本文提出的方法应用到其他零件上。而在技术的实际应用中，可能会遇到新的挑战，需要进一步的实验与探索。

致 谢

执笔至此，毕业设计也到了最后阶段，也标志着我在四川大学的学习生涯进入了尾声。回首这四年，感慨颇多，经历了紧张的学习、多彩的校园生活，也在疫情的阴霾中度过了最困难的时期，心中最想说的，便是对身边老师、同学和家人的感谢。

首先要感谢中国铁道科学院基础设施检测研究所杜馨瑜老师和北京铁科英迈技术有限公司顾子晨老师对本项目的支持，为课题提供了大量材料和数据资料。

在本科四年的学习生涯中，最感谢的是高绍兵老师。从 2019 年开始，我通过吴院导师制度联系到了高老师，并跟着实验室一起学习、研究。刚开始的时候，没有专业知识的支撑，对科研工作也时一筹莫展，不知道从何入手。感谢高老师的指教和监督，让我一点一点地学习计算机视觉、深度学习方面的知识，不断提升自己的专业水平和科研素养。感谢高老师对我的信任，在我自己都没有信心的时候让我负责课题项目，有机会去研究探索。同时，也要感谢实验室的师兄师姐们，尤其是邱健琿师兄，他给予我非常多的建议和帮助。希望实验室能越来越好，每一篇投稿都能中顶会顶刊。

我也要感谢本科四年中结识的每一位朋友，尤其感谢我曾经室友，马煜、张为、黎宇鑫，李浩斌，郭秀烨等，还有我现在的室友薛思言和唐金龙。我离开家乡来到千里之外的成都上学，他们给予了我最多的陪伴，让我并不觉得孤独。我犹记得第一个学期的期末复习阶段，每天晚上从自习室回来，穿过寒冷的夜色，回到寝室，和室友一起复习，相互答疑，心里总会觉得温暖。在此，还要特别感谢吴院篮球队。我没有参加太多的社团和部门活动，但却跟着球队训练了四年，把大多数的记忆都留在了球场和汗水中。虽然我们战绩胜少败多，球队活动也无法为保研推免加分，但在球队里面，我能感受到最纯粹的自己，保持着在这个竞争内卷时期最诚挚的热爱和最珍贵的友情。

最后，感谢我的父母，感谢他们对我求学之路中的经济和精神的支持。在我大学最迷惘的时期，父母的陪伴、开导给予我新的动力与希望。感谢他们二十多年来对我无私的爱、无私的付出，呵护我成长，教会我为人处世之道，一直在背后支持着我、理解着我。

当秋风染红故乡的枫叶时，我也会在港科的校园里，聆听海浪的声音。我在川大的旅程行将结束，也马上要开启一段新的求学生活。回顾四年的时光，有太多的感谢，有太多的不舍，都需要在这个夏天用告别画上句点。或许在将来的某天，重新回到成都，去川大走一走，在明远湖畔走一走，在荷花池和钟楼下走一走，依旧能回味起“海纳百川，有容乃大”的那份风骨和胸怀。

参考文献

- [1] 白玉新. 接触网常见故障及对策研究[J]. 中国铁路, 2009 (3) : 62-64.
- [2] Ge Z, Liu S, Wang F, et al. YOLOX: Exceeding YOLO Series in 2021[J]. arXiv e-prints arXiv: 2107.08430, 2021.
- [3] Schlegl T, Philipp S, Sebastian M. et al. Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery[C]// International Conference on Information Processing in Medical Imaging, Boone, USA, Springer, 2017:146-157.
- [4] Mehdi M and Simon O, Conditional Generative Adversarial Nets[J]. arXiv e-prints arXiv:1411.1784, 2014.
- [5] Dalal N, Triggs B. Histograms of Oriented Gradients for Human Detection[C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE, 2005.
- [6] Ojala T, Pietikainen M, Harwood D. A Comparative Study of Texture Measures with Classification Based on Feature Distributions[J]. Pattern Recognition, 1996, 29(1):51-59.
- [7] D.G. Low. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004.
- [8] Zou Z, Shi Z, Guo Y, et al. Object Detection in 20 Years: A Survey[J]. arXiv e-prints arXiv:1905.05055, 2019.
- [9] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation[C]// Computer Vision and Pattern Recognition, Columbus, OH, USA, IEEE, 2014: 580 - 587.
- [10] Krizhevsky A, Sutskever I and Hinton G. Imagenet classification with deep convolutional neural networks[C]// Conference on Neural Information Processing Systems, Lake Tahoe Nevada, United States, 2012.
- [11] He K, Zhang X, Ren S, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 37(9):1904-16.
- [12] Girshick R. Fast R-CNN[J]. arXiv e-prints arxiv: 1504.08083, 2015.
- [13] 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.
- [14] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick. Mask R-CNN[J]. arXiv e-prints arXiv: 1703.06870, 2017
- [15] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object

- Detection[C]// IEEE Conference on Computer Vision and Pattern Recognition, 2016: 779–788.
- [16] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]// IEEE Conference on Computer Vision and Pattern Recognition, 2017:6517–6525.
- [17] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement[J]. arXiv e-prints arXiv:1804.02767, 2018.
- [18] Lin T. Y, Dollar P, Girshick R, et al. Feature Pyramid Networks for Object Detection[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2017.
- [19] Bochkovskiy A, Wang C Y, Liao H. YOLOv4: Optimal Speed and Accuracy of Object Detection[J]. arXiv e-prints arxiv: 2004.10934, 2020.
- [20] Chen Q, Wang Y, Yang T, et al. You Only Look One-level Feature[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021:13034–13043.
- [21] Wei Liu, Cheng-Yang Fu, Alexander C. Berg, et al. SSD: Single Shot MultiBox Detector. European Conference on Computer Vision, Springer, Cham, 2016.
- [22] Lin T Y, Goyal P, Girshick R, et al. Focal Loss for Dense Object Detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017(99):2999–3007.
- [23] M. Rudolph, B. Wandt and B. Rosenhahn. Structuring Autoencoders[C]// IEEE/CVF International Conference on Computer Vision Workshop, 2019:615–623.
- [24] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, et al. Generative Adversarial Nets[J]. Advances in Neural Information Processing Systems, 2014:2672 – 2680.
- [25] Schlegl Thomas, Philipp Seeböck, Sebastian M, et al. f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks[J]. Medical Image Analysis, 2019(54):30 – 44.
- [26] Samet Akcay, Amir Atapour Abarghouei and Toby P. Breckon. GANomaly: Semi-supervised Anomaly Detection via Adversarial Training[C]// 14th Asian Conference on Computer Vision, Perth, Australia, 2018:622--637.
- [27] Jerone Andrews, Thomas Tanay, Edward Morton, and Lewis Griffin. Transfer representation-learning for anomaly detection[C]// International Conference on Machine Learning, Anomaly Detection Workshop. 2016.
- [28] Bernhard Schölkopf, Robert Williamson, Alex Smola, et al. Support Vector Method for Novelty Detection[C]// Advances in Neural Information Processing Systems, 1999(12):582–588.
- [29] Cohen N, Hoshen Y. Sub-Image Anomaly Detection with Deep Pyramid Correspondences[J]. arXiv e-prints arXiv:2005.02357, 2020.

- [30] Roth K, Pemula L, Zepeda J, et al. Towards Total Recall in Industrial Anomaly Detection[J]. arXiv e-prints arXiv:2106.08265, 2021.
- [31] Marco Rudolph, Bastian Wandt and Bodo Rosenhahn. Same Same But DifferNet: Semi-Supervised Defect Detection with Normalizing Flows[C]// IEEE Winter Conference on Applications of Computer Vision, 2021:1906–1915.
- [32] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn and Bastian Wandt. Fully Convolutional Cross-Scale-Flows for Image-based Defect Detection[C]// Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 2022:1829–1838.
- [33] 赵冰, 代明睿, 李平, 马小宁, 吴艳华. 基于深度学习的铁路关键部件缺陷检测研究[J]. 铁道学报, 2019 (41): 67–73.
- [34] 康高强, 高仕斌, 于龙, 陈健雄. 基于深度学习的高铁接触网旋转双耳开口销钉缺失故障检测[J]. 铁道学报, 2020, 42(10): 45–51.
- [35] Junping Zhong, Zhigang Liu, Guinan Zhang, Zhiwei Han, Wenxuan Zhang. A CNN-based defect inspection method for catenary split pins in high-speed railway[J]. IEEE Transactions on Instrumentation and Measurement, 2019(68): 2849 – 2860.
- [36] Hong S, Roh B, Kim K H, et al. PVANet: Lightweight Deep Neural Networks for Real-time Object Detection[J]. arXiv e-prints arXiv:1608.08021, 2016.
- [37] N. Anwar, Z. Shen, Q. Wei et al. YOLOv4 Based Deep Learning Algorithm for Defects Detection and Classification of Rail Surfaces[C]// International Intelligent Transportation Systems Conference, IEEE, 2021:1616–1620.
- [38] Muhammad Zaigham Zaheer, Jin-ha Lee, Marcella Astrid, Seung-Ik Lee. Old is Gold: Redefining the Adversarially Learned One-Class Classifier Training Paradigm[C]// Conference on Computer Vision and Pattern Recognition, IEEE, 2020:14171– 14181.
- [39] Dongkai Zhang, Shibin Gao, Long Yu, et al. DefGAN: Defect Detection GANs With Latent Space Pitting for High-Speed Railway Insulator[J]. Transactions on Instrumentation and Measurement, IEEE, 2021(70):1–10.
- [40] Y. Lyu, Z. Han, J. Zhong, et al. A Generic Anomaly Detection of Catenary Support Components Based on Generative Adversarial Networks[J]. Transactions on Instrumentation and Measurement, IEEE, 2020(69):pp. 2439–2448.
- [41] Yann, L. and Bottou, L. and Bengio, Y. and Haffner, Patrick. Gradientbased learning applied to document recognition[J]. Proceedings of the IEEE, 1986:2278–2324.
- [42] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. Deep Residual Learning for Image Recognition[C]// Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016:770–778.
- [43] Andrew G. Howard, Menglong Zhu, Bo Chen, et al. MobileNets: Efficient Convolutional

- Neural Networks for Mobile Vision Applications[J]. arXiv e-prints arXiv:1704.04861, 2017.
- [44] M. Sandler, A. Howard, M. Zhu, et al. MobileNetV2: Inverted Residuals and Linear Bottlenecks[C]// Conference on Computer Vision and Pattern Recognition, IEEE/CVF, 2018:4510-4520.
- [45] A. Howard, M. Sandler, B. Chen, et al. Searching for MobileNetV3[C]// International Conference on Computer Vision, IEEE/CVF, 2019:1314-1324.
- [46] J. Long, E. Shelhamer, T. Darrell. Fully convolutional networks for semantic segmentation[C]// Conference on Computer Vision and Pattern Recognition, IEEE, 2015:3431-3440.
- [47] O. Ronneberger, P. Fischer, T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation[J]. Medical Image Computing and Computer Assisted Intervention, 2015:234-241.
- [48] S. Vandenhende, B. De Brabandere, D. Neven and L. Van Gool. A Three-Player GAN: Generating Hard Samples to Improve Classification Networks[C]// International Conference on Machine Vision Application, 2019(1-6).
- [49] N. Shazeer, K. Fatahalian, W. R. Mark and R. T. Mullapudi. HydraNets: Specialized Dynamic Architectures for Efficient Inference[C]// Conference on Computer Vision and Pattern Recognition, IEEE, 2018:8080-8089.

附录：项目核心伪代码

CGAN 模型架构:

```
class CGAN(object):
    function build_generator(self):
        # 定义生成器
        c1 ← Conv2D(filters←64)(self.gen_input)
        b1 ← BatchNormalization()(c1)
        l1 ← LeakyReLU(alpha←0.2)(b1)
        c2 ← Conv2D(128)(l1)
        b2 ← BatchNormalization()(c2)
        l2 ← LeakyReLU(alpha←0.2)(b2)
        c3 ← Conv2D(256)(l2)
        b3 ← BatchNormalization()(c3)
        l3 ← LeakyReLU(alpha←0.2)(b3)
        c4 ← Conv2D(512)(l3)
        b4 ← BatchNormalization()(c4)
        l4 ← LeakyReLU(alpha←0.2)(b4)
        c5 ← Conv2D(512)(l4)
        b5 ← BatchNormalization()(c5)
        l5 ← LeakyReLU(alpha←0.2)(b5)
        c6 ← Conv2D(512)(l5)
        b6 ← BatchNormalization()(c6)
        l6 ← LeakyReLU(alpha←0.2)(b6)
        c7 ← Conv2D(512)(l6)
        b7 ← BatchNormalization()(c7)
        l7 ← LeakyReLU(alpha←0.2)(b7)
        c8 ← Conv2D(512)(l7)
        b8 ← BatchNormalization()(c8)

        t1 ← Conv2DTranspose(512)(b8)
        d1 ← Dropout(0.5)(t1)
        d1 ← BatchNormalization()(d1)
        d1 ← concat([d1, b7], 3)
        r1 ← ReLU()(d1)
        t2 ← Conv2DTranspose(512)(r1)
        d2 ← Dropout(0.5)(t2)
        d2 ← BatchNormalization()(d2)
        d2 ← concat([d2, b6], 3)
        r2 ← ReLU()(d2)
        t3 ← Conv2DTranspose(512)(r2)
        d3 ← Dropout(0.5)(t3)
        d3 ← BatchNormalization()(d3)
        d3 ← concat([d3, b5], 3)
        r3 ← ReLU(d3)
        t4 ← Conv2DTranspose(512)(r3)
        d4 ← Dropout(0.5)(t4)
        d4 ← BatchNormalization()(d4)
        d4 ← concat([d4, b4], 3)(d4)
        r4 ← ReLU(d4)
        t5 ← Conv2DTranspose(256)(r4)
        d5 ← Dropout(0.5)(t5)
        d5 ← BatchNormalization()(d5)
        d5 ← concat([d5, b3], 3)
        r5 ← ReLU(d5)
        t6 ← Conv2DTranspose(128)(r5)
```

```

d6 ← Dropout(0.5)(t6)
d6 ← BatchNormalization()(d6)
d6 ← concat([d6, b2], 3)
r6 ← ReLU(d6)
t7 ← Conv2DTranspose(64)(r6)
d7 ← Dropout(0.5)(t7)
d7 ← BatchNormalization()(d7)
d7 ← concat([d7, b1], 3)
r7 ← ReLU(d7)
t8 ← Conv2DTranspose(3)(r7)
output ← Activation('tanh')(t8)

gen_model ← Model(input←self.gen_input, outputs←output, name←'gen_model')
return gen_model

```

```

function build_discriminator(self):
    # 定义判别器
    c1 ← Conv2D(64)(self.dis_input)
    l1 ← LeakyReLU(alpha←0.2)(c1)
    c2 ← Conv2D(128)(l1)
    b2 ← BatchNormalization()(c2)
    l2 ← LeakyReLU(alpha←0.2)(b2)
    c3 ← Conv2D(256)(l2)
    b3 ← BatchNormalization()(c3)
    l3 ← LeakyReLU(alpha←0.2)(b3)
    c4 ← Conv2D(512)(l3)
    b4 ← BatchNormalization()(c4)
    l4 ← LeakyReLU(alpha←0.2)(b4)
    c5 ← Conv2D(1)(l4)
    output ← Activation('sigmoid')(c5)

    dis_model ← Model(inputs←self.dis_input, outputs←output, name←'dis_model')
    return dis_model

```

模型训练:

```

function evaluate(self, runstep, img_size):
    gen_model.load_weights(self.model_save_path)
    for step ← 0; step < len(self.test_picture_list); step++ do
        picture ← imread(self.test_picture_list[step])
        height ← picture.shape[0]
        width ← picture.shape[1]
        picture_resize ← resize(picture, (img_size, img_size))
        batch_picture ← expand_dims(array(picture_resize))
        gen_label ← gen_model(batch_picture)
        out_img ← resize(gen_label, (width, height)) # 恢复到原来尺寸
        imwrite(write_image_name, out_img)

function train(self, img_size):
    counter ← 0
    gen_model.load_weights(model_save_path) # 加载已保存的模型参数
    for epoch ← 0; epoch < self.epochs; epoch++ do
        random.shuffle(train_picture_list) # 打乱图像顺序, 随机选取训练样本
        for step ← 0; step < len(self.train_picture_list); step++ do
            counter += 1
            picture ← imread(self.train_picture_list[step])
            label ← imread(self.label_picture_list[step])

```

```

height ← picture.shape[0] # 得到图片的高
width ← picture.shape[1] # 得到图片的宽
picture_resize ← resize(picture, (img_size, img_size))
label_resize ← resize(label, (img_size, img_size))
batch_picture ← expand_dims(array(picture_resize))
batch_label ← expand_dims((label_resize))
gen_loss, dis_loss, gen_label ← update_loss(batch_picture, batch_label, counter)
gen_ls.append(gen_loss)
dis_ls.append(dis_loss)
if counter % self.test_iter_num == 0:
    evaluate(counter) # 测试
    gen_model.save_weights(model_save_path)
if counter % self.save_iter_num == 0:
    out_img ← resize(out_img, (width, height)) # 恢复到原来尺寸
    imwrite(out_img)
    gen_model.save_weights(model_save_path) # 保存模型
return gen_ls, dis_ls

```

端到端生产缺陷样本:

```

function generate(self, picture, img_size):
    gen_model.load_weights(self.gen_model_path)
    height ← picture.shape[0] # 得到图片的高
    width ← picture.shape[1] # 得到图片的宽
    picture_resize ← resize(picture, (img_size, img_size))
    batch_picture ← expand_dims(array(picture_resize))
    gen_label ← gen_model(batch_picture)
    out_img ← resize(gen_label, (width, height))

    return out_img

function gen_big_img(self, fs):
    for im_name in fs:
        path ← join(self.data_path, im_name)
        image ← imread(path)
        # 加载定位模型
        box_list, pin_list ← detectorF.detectObj(image, self.threshold, self.isSave)

        for idx ← 0; idx < len(pin_list); idx++ do
            box ← box_list[idx]
            class_name ← pin_list[idx]
            xc, yc, w, h ← box
            x ← xc - w//2
            y ← yc - h//2
            x1 ← max(0, x) # 判断是否在图像边缘
            y1 ← max(0, y)
            img_cut ← image[y1:y+h+expand, x1:x+w+expand]
            if class_name == 'h_good':
                img_gen ← generate(img_cut, gen_h_model_path)
            elif class_name == 's_good':
                img_gen ← generate(img_cut, gen_s_model_path)

            image[y:y+h, x:x+w] ← img_gen
        save_name ← save_path + im_name
        imwrite(save_name, image)

```