



南京大學

研究生畢業論文
(申請碩士專業學位)

論文題目 基于栽育端与性状表征的
水果病斑图像数据扩增技术

作者姓名 顾雪晴

专业学位类别(领域) 工程硕士

研究方向 软件工程

指导教师 陈振宇 教授

2022 年 05 月 20 日

学 号： MF20320050

论文答辩日期： 2022 年 05 月 20 日

指 导 教 师： (签字)



Data Augmentation Technology of Fruit Disease Image Based on Planting End and Feature Characterization

By

Xueqing Gu

Supervised by

Professor Zhenyu Chen

A Thesis

Submitted to the Software Institute

and the Graduate School

of Nanjing University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Engineering

Software Institute

May 2022

学位论文原创性声明

任何收存和保管本论文的单位和个人，未经作者本人授权，不得将本论文转借他人并复印、抄录、拍照或以任何方式传播，否则，引起有碍作者著作权权益的问题，将可能承担法律责任。

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不句含其他个人或集体已经发表或撰写的作品成果。本文所引用的重要文献，均已在文中以明确方式标明。本声明的法律结果由本人承担。

论文作者签名：_____

日期： 年 月 日

南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目：基于栽育端与性状表征的水果病斑图像数据扩增技术

工程硕士（软件工程领域） 专业 2020 级硕士生姓名：顾雪晴

指导教师（姓名、职称）：陈振宇 教授

摘 要

随着大数据时代到来，人工智能技术日趋完善，在自动驾驶、医疗诊断、工业制造等领域都有着广泛的应用。然而智慧农业受到基础设施建设不够完备、缺乏全面成熟的顶层设计等因素的限制，发展程度远不及其他领域。水果作为经济效益最高的农产品之一，适宜打造统筹管理下的智能化水果种植，来保证品质稳定、实现增产增收。其中包括苗情实时监控、病虫害识别以及定制化水肥供给等多种功能，都需要依靠图像数据作为基础。但栽育端环境复杂，图像传感设备价格高昂因此普及率低，采集到的图像又存在严重的样本失衡问题，这就导致水果图像的数量和质量都无法给智能终端提供正向反馈。加上农业领域本身的知识壁垒，需要根据性状区分不同的病害，也无形中加剧了数据获取的难度。

因此本研究着眼于水果病斑图像的数据扩增，将栽育端场景特点以及病斑的性状表征纳入扩增考虑，从而赋能水果种植端的病害监测。对于自然环境的模拟，从天气因素和生物影响两方面扩增正样本，旨在生成符合栽育端场景的对抗性样本，以提升相关智能设备的抗干扰能力、增强鲁棒性。而在水果病斑方面，进一步具体化为水果炭疽病的负样本扩增。从分割后融合、特征模拟和自动生成这三个方向，对水果病斑周期性的形态特征以及极具传染性的分布特征进行模拟和复现，以实现扩增图像的高真实度与高质量。

本研究提出的扩增方法能够在确保扩增结果质量的同时，具有更强的针对性，从而实现了对农业的深度赋能。其生成的样本不仅符合领域特征，而且正样本具有干扰性、负样本具有真实性。对比利用真实数据和基础扩增数据训练生成的网络，在模型鲁棒性和预测精度的提升方面，可以达到综合最优。此外，本研究将扩增技术应用到农业图像领域，填补了细分病害下种植端水果病斑图像数据集的空白，从而为水果种植端的监测告警、病害识别等提供了更多更可靠的数据支持。

关键词：数据扩增，智慧农业，水果炭疽病，图像处理，深度学习

南京大学研究生毕业论文英文摘要首页用纸

THESIS: Data Augmentation Technology of Fruit Disease Image Based on Planting End and Feature Characterization

SPECIALIZATION: Software Engineering

POSTGRADUATE: Xueqing Gu

MENTOR: Professor **Zhenyu Chen**

Abstract

With the advent of big data era, artificial intelligence technology is continually being improved, which has been widely used in automatic driving, medical diagnosis, industrial manufacturing and so on. However, smart agriculture is far less developed than other fields because it is limited by factors such as incomplete infrastructure construction and lack of comprehensive and mature top-level design. Fruit as one of the agricultural products with the highest economic benefits, is suitable for building intelligent planting under overall management to ensure stable quality and increase production and income. Intelligent fruit planting includes many functions, such as real-time seedling monitoring, pest identification and customized water and fertilizer supply, which all rely on image data as basis. However, the environment at planting end is complex, the popularity of image sensors is low which due to their high price, and there is a serious sample imbalance problem in the collected images. All above make the quantity and quality of fruit images unable to provide positive feedback to the intelligent terminal. Coupled with the knowledge barrier in the agricultural field itself, it is necessary to distinguish different diseases according to their characters, which virtually aggravates the difficulty of data acquisition.

Therefore, this thesis focuses on the data augmentation of fruit disease image, taking the scene characteristics at planting end and the feature characterization of disease spot into account, so as to enable the disease monitoring at fruit planting end. For the simulation of natural environment, positive samples are augmented based on weather factors and biological effects, in order to generate adversarial samples that meet the planting end scene, thus improving the anti-interference ability and robustness of relevant intelligent devices. In the aspect of fruit disease, we further concretize it into the

negative samples augmentation of fruit anthracnose. We propose technologies from three ways which are fusion after segmentation, feature simulation and automatic generation to reproduce the periodic morphological features and highly infectious distribution characteristics of fruit disease spots, so as to achieve high authenticity and high quality of the augmented image.

The augmentation methods proposed in this thesis can not only ensure the quality of augmented results, but also have stronger pertinence, thus realize the deep empowerment of agriculture. Samples generated by our methods not only accord with the characteristics of agriculture field, but also the positive samples have interference and the negative samples have authenticity. Compared with the network trained by real data and basic augmented data, it can achieve comprehensive optimization in the improvement of model robustness and prediction accuracy. In addition, this thesis applies the augmentation technology to the field of agricultural images, which fills the gap of the fruit image datasets at planting end under the subdivision of diseases, so as to provide more reliable and more data support for the monitoring alarm and disease identification at fruit planting end.

Keywords: Data Augmentation, Smart Agriculture, Fruit Anthracnose, Image Processing, Deep Learning

目录

表目录	vi
图目录	viii
第一章 引言	1
1.1 项目背景	1
1.2 研究现状	3
1.3 论文组织与主要贡献	6
1.4 本章小结	7
第二章 相关理论综述	8
2.1 通用图像扩增	8
2.1.1 坐标变换	8
2.1.2 颜色空间变换	9
2.1.3 像素点变换	12
2.2 病斑特征分析	14
2.2.1 水果炭疽病性状表征	14
2.2.2 颜色特征提取	16
2.2.3 形状特征提取	18
2.2.4 纹理特征提取	21
2.3 数字图像处理	24
2.3.1 滤波	24
2.3.2 边缘检测与边界提取	25
2.3.3 腐蚀和膨胀	25
2.3.4 图像金字塔	26
2.4 深度学习方法	27
2.4.1 生成式对抗网络	27
2.4.2 风格迁移	28
2.5 本章小结	28

第三章 基于栽育端和性状表征的图像数据扩增方法	30
3.1 图像采集场景及特征分析	30
3.2 正样本扩增技术	32
3.2.1 不可抗性天气因素	32
3.2.2 偶发性生物影响	35
3.3 负样本扩增技术	45
3.3.1 分割后融合技术	46
3.3.2 特征模拟技术	51
3.3.3 目标检测技术	60
3.3.4 自动生成技术	64
3.4 本章小结	69
第四章 实验设计与结果分析	73
4.1 数据准备与处理	73
4.1.1 目标检测数据集	73
4.1.2 对抗生成网络数据集	74
4.2 扩增效果检验指标	75
4.2.1 图片质量	75
4.2.2 图片真实性	76
4.2.3 扩增有效性	78
4.3 实验设计与结果分析	81
4.4 本章小结	88
第五章 总结与展望	89
5.1 总结	89
5.2 未来工作展望	90
参考文献	92
简历与科研成果	98
致谢	99

表 目 录

3.1 病斑颜色及形状特征量均值·····	52
3.2 不同区域灰度共生矩的特征量均值·····	55
4.1 扩增图像质量评估·····	81
4.2 扩增图像真实性评估·····	83
4.3 扩增图像有效性评估 1——鲁棒性·····	85
4.4 扩增图像有效性评估 2——精确性·····	86

图 目 录

1.1	智慧农业概览	1
1.2	通用扩增不安全示例	3
2.1	仿射变换	9
2.2	弹性变换	9
2.3	归一化的 RGB 空间图	10
2.4	HSV 空间图	11
2.5	L*a*b* 立体空间图	11
2.6	人为因素	13
2.7	模拟镜头失真	13
2.8	叶片炭疽病性状表现	15
2.9	果实炭疽病性状表现	16
2.10	链码及边界表示	19
2.11	霍夫变换	20
2.12	圆形 LBP 算子	22
2.13	旋转不变的 LBP 算子	22
2.14	滤波原理	24
3.1	种植端图像采集的干扰因素	30
3.2	技术路线总览	31
3.3	雾场景扩增	32
3.4	雨场景扩增	35
3.5	雪场景扩增	35
3.6	形态学分割法获取昆虫前景	37
3.7	自动抠图法提取复杂背景下的昆虫前景	40
3.8	昆虫自动抠图网络架构图	41
3.9	泊松融合得到昆虫影响扩增结果	44
3.10	泊松融合原理	44

3.11 负样本扩增思路·····	46
3.12 决策树和 kmeans 聚类分割病斑结果比较·····	50
3.13 病灶图像集·····	50
3.14 病灶插入的融合方法比较·····	51
3.15 分割后融合扩增结果·····	52
3.16 病斑颜色直方图比较·····	53
3.17 使用霍夫变换对水果病斑图像进行圆检测·····	54
3.18 病斑图像不同区域 LBP 统计直方图比较·····	54
3.19 螺旋遍历晕染病斑·····	56
3.20 炭疽病早期病斑模拟扩增·····	56
3.21 叶边缘轮廓获取示意图·····	58
3.22 炭疽病晚期病斑模拟·····	61
3.23 YOLO-v3 目标检测结果·····	62
3.24 DCGAN 生成水果病叶·····	65
3.25 CycleGAN 原理示意图·····	67
3.26 CycleGAN 生成病叶病果·····	69
4.1 YOLO-v3 训练集 label 数据形式·····	74
4.2 验证有效性使用的 CNN 网络架构·····	80
4.3 水果检测模型对扩增正样本的识别效果·····	87
4.4 病果检测模型对扩增负样本的识别效果·····	87

第一章 引言

1.1 项目背景

大数据与云计算、物联网等多种信息技术的联合应用在多个领域已得到长足的发展，包括智慧医疗 [1][2]、智慧教育 [3][4] 等。然而我国作为农业大国，受到地理条件复杂、经济发展差异等因素的影响，在智能化农业的基础设施与体系建设方面与经济发达国家依旧存在较大差异。因此我国迫切需要在完善硬件条件的基础上，综合利用农业传感和遥感数据 [5]、物联网与云计算 [6] 等技术，为智慧农业体系赋能。以实现农业生产过程自动化、监管可视化、决策智能化的目标，从而为农业可持续发展增添新的活力。



图 1.1: 智慧农业概览

数据作为人工智能的基石，在智慧农业的建设过程中发挥着至关重要的作用。尤其是种植端，例如温湿度传感器、光照传感器和风速传感器等常被用于监测作物生长和发育的状况，以便改变培育要素。而在诸多的农业传感设备中，图像传感器凭借具象且丰富的信息获取能力，在数字农业产业园中被广泛应用 [7]，以帮助完成病虫害信息告警 [8][9]、自动植保作业、结合专家知识定制培育策略 [10] 和远程病害诊断 [11] 等。因此图像数据成为解锁智慧农业的关键。

但简单对农业领域图像的特征进行分析后可以发现，图像数据获取主要存在以下几个方面的挑战：

1、就内容种类来说，图像在农业中的应用不同，其呈现内容自然有所差异。大方向上，农业图像主要用于农作物生长过程的监测、农业资源调查、农产品外观分类和质量检测等。而具体的细分领域又包括粮食作物、经济作物、水果花卉等。因此农业图像具有较强的区分性，需要针对特定情境获取，数据集普适度低。

2、从图像采集频率出发，越是高清的图像越能对农业智能系统产生正向反馈，而受种植成本影响，高精度的传感设备及相关仪器难以广泛应用，使得农业图像的采样频率低。此外，在实际的数据采集工作中，往往忽视行为数据，只注重统计数据，导致数据积累不够。

3、从图像质量的角度看，农业图像会受到仪器和采集环境等因素的影响和制约，如设备元件的损坏和自然条件下的各种天气变化，都可能导致异常情况下的低质量图像。

4、对顶层架构进行分析，现实场景下传感器进行图像采集的标准不够完善，数据采集通路间不具有统一的规范，这样良莠不齐的图像难以构成有效的数据集，从而桎梏了后续的图像处理和研究。

此外，由于农业领域知识存在专业性壁垒，图像标注时还需要结合专家经验实现不同农业应用场景下的鉴别与评价，例如种植端的栽育告警、分选端的评级检测等。这也进一步加剧了农业图像数据获取的难度，无形中掣肘了智慧农业的发展。因此对于农业领域数据扩增的研究迫在眉睫。

正如前面所提到的，对于农业图像扩增，首先需要界定其内容与应用场景。

在众多的农作物产品中，水果的商品率和经济效益高、对培育要素和技术条件要求严格。而在传统的水果生产流程里，种植端常常出现管理粗放、生产盲目且缺乏体系标准的问题。这既容易导致水果品质不受控，也让消费者在购买时无源可溯，对糖度、产地、安全度等质量指标一无所知。与此同时，相比于进口水果，传统果园出品的产品包装简陋、品牌意识缺乏，没有明显的溢价效应，常常优质但不优价。因此国家大力推动我国的水果产业向高质量转型发展，将大数据和物联网等技术应用于水果生产种植，帮助果农增产增收。

本研究顺应此态势，进一步将农业图像扩增细化为水果图像的数据扩增。而在智慧水果的功能领域，病害的监测识别 [12][13] 又是至关重要的一项。因为水果病害发展往往十分迅猛，如果不能对其进行及时捕捉，将可能造成大规模的损失。这就要求水果种植端的智能设备不仅要有精准告警病害的能力，还要能适应复杂多变的自然环境，因此对训练数据提出了多方面的要求。而在匮乏的种植端水果图像数据中，样本分布还呈现极度不均衡的情况。所以本研究的不仅是扩增出数量更多且分辨率较高的水果病斑图像，而且要对栽育端复杂

的自然环境以及病斑的性状表征进行模拟和复现，以获取更加真实有效的扩增图片，更好地赋能智慧农业。

1.2 研究现状

数据对于机器学习至关重要 [14]，但在 AI 发展中却经常被忽视或处理不当。当前智慧农业出现的瓶颈，除了基础设施不够完善外，很大一个原因就是农业数据的缺失。尤其是匮乏且质量低的图像数据，严重限制了农业智能设备的发展。

除此之外，诸如智慧医疗等领域，也严重依赖大数据来进行图像分析，如果得不到足够多的训练数据，就会使网络陷入过拟合的境地。因此数据扩增技术进入人们的视野，它企图从数据空间的角度解决有限数据的问题。

数据扩增包含一系列增强训练数据集大小和质量的技术，以便使用它们构建更好的深度学习模型 [15][16][17]。一般来说可以将图像数据扩增算法划分为基于基本图像处理的数据扩增和基于深度学习的数据扩增。其中通过基本的图像处理方法来扩增数据主要包括几何变换、颜色空间变换、核滤波器、混合图像和随机擦除 [18] 等，部分处理方式在本研究中作为正样本扩增技术进行应用，第二章将会对其进行详细描述。

研究认为，虽然基于基本图像处理的扩增方法相对易于实现，但在应用“安全性”的角度却还是需要进一步考量 [19]。所谓的安全性是指其在转换后保留标签的可能性，例如在 ImageNet 网络中，对猫和狗的旋转和翻转通常是安全的，但在数字识别任务中则不安全（如图 1.2）。每一种图像处理功能都可能会在某种失真程度上导致标签更改变换，因此特定于领域的扩增设计以及开发可推广的扩增策略 [20] 充满挑战。

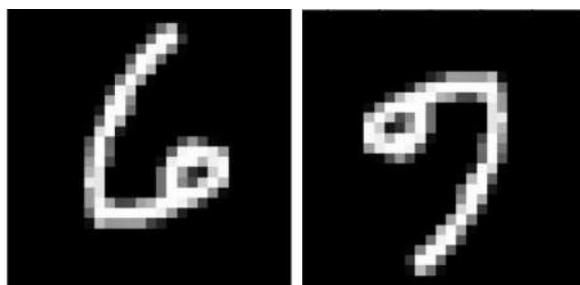


图 1.2: 通用扩增不安全示例

单纯使用通用的扩增方法 [21]，将诸如裁剪、翻转、色偏和随机擦除等方法结合起来的确可以产生大规模膨胀的数据集。然而这对于模型训练来说，并不

一定是有利的。在数据非常有限的领域，这反而可能会导致进一步的过拟合。因此更多的研究将目光投向使用深度学习进行扩增。

基于深度学习的扩增方法主要包括对抗性训练、基于 GAN 的数据扩增和风格迁移。

对抗性训练使用两个或多个网络的框架，这些网络的损失函数中编码有不同的目标。通过限制对抗网络可用的扩增集和扭曲集，它可以学习产生导致错误分类的扩增，从而形成有效的搜索算法，这些扩增对于加强分类模型中的弱点是有价值的。因此对抗性训练的扩增结果虽然可能不代表测试集中可能出现的样本，但可以改善学习的决策边界中的弱点。

[22] 发现，如果对测试图像进行简单的变换，如旋转和平移等，就很容易导致深度 CNN 模型分类错误。效果最明显的实验中，将图像随机变换使得 MNIST 数据集准确率降低了 26%、CIFAR10 准确率降低了 72%，ImageNet (Top 1) 的准确率降低了 28%。基于此发现，[23] 针对深度卷积神经网络，研究出一种名为“快速梯度符号法”的对抗样本生成技术，其对于 MNIST 数据集生成的对抗样本，使 maxout 网络以 97.6% 的平均置信度对 89.4% 的对抗样本进行了错误分类。将对抗样本加入训练数据中后，提高了 MNIST 分类任务的性能，对抗样本的错误率从 89.4% 下降到 17.9%。

[24] 试验了一种新的对抗训练方法，并比较了原始测试数据和对抗样本的性能。结果显示实验中的对抗性训练并没有提高测试的准确性，但确实显著提高了对抗样本的测试准确性。[25] 在快速梯度符号法的基础上进行改进，提出 DeepFool 使用神经网络来寻找导致误分类的可能的最小噪声扰动。

不同于其他大多数向输入或隐藏层添加噪声的扩增方法，[26] 提出了一种在每次迭代中随机替换标签的正则化技术 DisturbLabel。在基于 MNIST 数据集训练的 CNN 网络 Lenet 上，DisturbLabel 产生了 0.32% 的错误率，而基线错误率为 0.39%。与 0.39% 的基线相比，干扰标签结合 dropout 正则化产生了 0.28% 的错误率。

对抗训练的原则又衍生出非常流行的生成建模框架——生成对抗网络 GANs [27]，虽然 GANs 并不是现有唯一的生成式建模技术，但在计算速度和结果质量方面显著领先。生成性建模可以通过学习数据集从而创建人工实例，得到的结果可以保留与原始训练集相似的特征。普通的 GAN 架构在生成和鉴别网络中使用多层感知机，能够在简单的图像数据集（例如 MNIST 手写数字集）上产生可接受的结果，然而它无法为更高分辨率和更复杂的数据集产生高质量的结果。因此，许多研究通过不同的网络架构、损失函数和进化方法来修改 GAN 框架，极大地提高了 GANs 创建样本的质量。在扩展 GANs 的概念以产生更高

分辨率的输出图像的新体系结构中，DCGANs、Progressively Growing GANs 和 Conditional GANs 在数据扩增中具有最大的应用潜力。

DCGAN[28] 提出了一种在 GAN 框架中使用卷积层以产生更高分辨率图像的策略，该体系结构的提出是为了扩展生成器和鉴别器网络的内部复杂性。[29] 借助 DCGANs 生成肝脏病变医学图像，并测试了其有效性。研究发现，使用经典的数据增强器可以获得 78.6% 的敏感性和 88.4% 的特异性，而在添加了 DCGANs 生成的样本后，敏感性和特异性分别增加到 85.7% 和 92.4%。

Progressively Growing GANs[30] 训练一系列分辨率渐进增长的网络，从低分辨率图像开始，通过向网络中逐步添加层的方式来渐进地提高分辨率。这种增量性质允许模型在训练过程中，可以首先在整体上学习图像分布的大规模结构，然后将重点转移到越来越精细的尺度细节上，而不必同时关注所有尺度。渐进式的训练存在几个好处：1) 早期较小图像的生成更加稳定，因为类信息和模式更少，网络能够使用 WGAN-GP 损失甚至 LSGAN 损失可靠地合成百万像素级图像。2) 减少了训练时间，因为大多数迭代都是在较低的分辨率下完成的，因此 PG-GANs 获得与其他 GANs 网络类似的结果质量，通常可以快 2-6 倍。应用研究也证明了，渐进增长的 GANs 在人脸识别领域能够取得非常惊人的效果。

除了提高 GAN 的分辨率大小外，另一个提高输出质量的架构是 Conditional GANs[31]。条件 GANs 向生成器和鉴别器添加条件向量，以缓解模式崩溃问题。除了向生成器输入一个随机向量 z ，条件 GANs 还输入一个 y 向量，这类似于一个独热编码的类标签，指代的是生成器和鉴别器的特定类。

尽管 GANs 的潜力受到了多方瞩目，但要想从当前的尖端体系结构中获得高分辨率的输出还是比较困难的，因为盲目增加生成器生成图像的输出尺寸可能会导致训练不稳定或不收敛。此外，GANs 还需要大量的数据进行训练，由于初始数据集大小受到限制，因此在数据扩增方面，GANs 可能并不是一个实用的解决方案。

风格迁移 [32] 除了图像的艺术应用之外，也是数据扩增的有效工具之一。对数据扩增来说，风格迁移在某种程度上类似于颜色空间光照变换，它扩展了光照变化，并允许对不同的纹理和艺术风格进行编码。在通过风格迁移的扩增方式生成新图像时，可以人为决定从哪些样式进行采样 [33]。但选择要采样的样式风格可能是一项具有挑战性的任务。对于自动驾驶汽车等应用，从夜到日、冬到夏、雨到晴的角度考虑对训练数据进行风格变换是非常直观的。但在其他应用领域中，要转换的风格集并不明显。从易于实现的角度出发，可以选择由 k 个风格组成的集合，并将其应用于训练集中的所有图像，再通过风格迁移进行数据扩增。

许多约束条件（如低保真度摄影机）可能会导致深度学习模型在物理模拟训练和现实世界部署时泛化能力差，使用风格迁移扩增训练数据，可以完成从模拟环境到现实世界的过渡。[34]发现，风格的多样性比在尽可能真实的环境中模拟更有效。

虽然风格迁移可以提高模拟数据集的泛化能力，但缺点在于需要选择将图像传输到的风格。如果风格集太小，可能会在数据集中引入更多偏差。

通过调研现有的图像数据扩增技术可以发现，以简单图像处理作为基础扩增的方式，虽然实现方便，但有可能改变图像的数据标签，导致扩增结果失效，反而无法解决模型过拟合等问题。而利用深度学习的方式进行数据扩增，在实现过程中会受限于初始数据集规模、模型架构和优化方向等多种因素，反而可能引入新的偏差。除此之外，上述的扩增方法都无法解决农业领域的知识壁垒，不能进行针对性的专业扩增。因此对于农业领域的图像数据，还需要进一步的研究和实验，来探索真正快速有效的扩增手段。

1.3 论文组织与主要贡献

本文主要研究水果病斑图像的数据扩增，从栽育端复杂的自然环境和农业领域病斑的性状表征两个角度出发，旨在扩增出更多质量和分辨率较高，且更加真实有效的种植端水果病斑图像。本文的组织结构如下：

第一章是文章的引言，首先介绍了项目提出的背景和研究意义，其次对当前数据扩增研究的现状进行了简要介绍和利弊分析，最后总体叙述了论文的组织 and 主要贡献。

第二章是相关的理论综述。首先介绍了通用的图像扩增技术，将其细分为坐标变换、颜色空间变换和像素值变换。其次是图像特征分析，主要为基于特征模拟的扩增方法进行铺垫。详细描述了水果炭疽病的性状表征和图像特征提取的方法。然后介绍了扩增方法中使用到的数字图像处理技术。最后对对抗生成网络及风格迁移思想进行了简单阐述，这也是本文利用深度学习进行扩增的理论基础。

第三章详细介绍了基于栽育端和性状表征的图像数据扩增方法，首先对水果种植端的图像采集场景及特征进行了分析，然后将扩增方法具体分为正样本扩增技术和负样本扩增技术。对于以种植端环境模拟为基础的正样本扩增，本文从不可抗的天气因素和偶发的生物影响两个角度出发，模拟现实场景下雨雪雾等自然现象并形成生物干扰的图像。而对于以水果炭疽病为复现目标的负样本扩增，则主要拓展了三种扩增思路。第一种是分割后融合技术，通过泊松融合将分割处理后得到的典型病斑插入正常水果图像中。第二种思路是特征模拟技

术，对炭疽病病斑的颜色、形状及纹理进行特征分析后，借助图像处理方法对病斑进行模拟。第三种思路是自动生成技术，主要使用将卷积神经网络应用于对抗生成的 DCGAN 和基于风格迁移思想构建的 CycleGAN 来自动生成水果病斑图片。

第四章介绍了本研究的实验设计和结果分析，由于涉及的实验较多，因此将实验数据简要分为目标检测数据集和对抗生成网络数据集，阐述了数据准备和处理的相关细节。其次对于扩增效果的检验，也分为了图片质量、真实性和扩增有效性三个维度，分别选取不同指标进行评估。最终详述实验设计，根据确定的评价标准对扩增结果进行了检验，利用图表的形式总结分析，发现本研究的扩增方法在扩增效果和领域专业性上具有综合优势。

第五章是对本研究的总结与展望，强调创新点的同时，对研究结果进一步剖析审视，指明扩增方法的局限性，以及实验过程中仍存在的不足之处，为未来的拓展工作规划了方向。

本文的**主要贡献**在于：农业领域的图像数据匮乏且普遍质量较低，而生物多样性又构成了领域内的知识壁垒，为智慧农业的发展带来了挑战。水果产业作为最适宜发展智能化的农业细分领域，更加需要完备的数据集提供支持。本研究以水果种植端病害监测为应用场景，提出的扩增方法一方面能够产生更多高质量和高真实度的种植端水果病斑图像，解决样本不均衡的问题，弥补种植端病害水果数据集的空白。另一方面在扩增效果和领域针对性上面也表现良好，对于提升智能终端的鲁棒性和精确性，达到了综合最优。从而进一步赋能智慧果园，为农业领域的智能化建设奠定了数据基石。

1.4 本章小结

本章主要是引言部分，简单分析了项目提出的背景，自然而然引出对水果病斑图像扩增的必要性及其社会价值，阐述了本研究课题的意义。同时对图像数据扩增的研究现状进行了分析，提出各自存在的利弊，专业性的领域扩增依旧是当前的技术难点。最后对论文的组织架构分章节概括，简要说明了本研究的主要贡献。

第二章 相关理论综述

2.1 通用图像扩增

种植端采集的水果图像由于是现实场景下拍摄的图片，因此区别于工业领域和医学领域的图像数据，色彩组织更加复杂，且不拘泥于场景和区域的限制，因时因地而异。故对于水果图像的正样本扩增，我们也考虑了安全的通用扩增技术，将关注点放在增强图像的多样性和随机性上，以期提升数据服务的有效性。

通用扩增可以简单分为三类：首先是坐标变换，只影响图像的尺度和像素点之间的相对位置。其次是颜色空间的变换，将原图应用于不同的颜色空间中，各个颜色空间之间可以互相转换。最后是像素值变换，包括模糊、增加噪声和修改 RGB 通道值等，这一类变换对原图信息的修改最大，扩增得到的图片效果与原图相去甚远。

2.1.1 坐标变换

坐标变换一般可以分为仿射变换与其他。仿射变换会保持二维图像的“平直性”和“平行性”，但是角度会有所改变。其包括平移变换、旋转变换、缩放变换（或尺度变换）、倾斜变换（或错切变换、剪切变换、偏移变换）、翻转变换，是一次空间线性变换或多次不同空间线性变换的组合。此外，坐标变换还包括等距、相似、裁剪、弹性变换等。

图片经过平移或旋转后，可能会引入黑色色块，并对原图信息造成不同程度的损失，如图2.1(b)和2.1(c)。缩放变换包括普通缩放、垂直缩放以及等比缩放，可以设置水平因子和垂直因子，通过变换矩阵实现，但同样会使原图片失真，影响清晰度。翻转包括垂直翻转和水平翻转，相较来说对于原图的细节保留会更加完备。与之对应的是裁剪变换，该操作只会保留原图中的部分内容，因此裁剪变换的百分比会控制在一定的阈值范围内。

弹性变换是对像素点各个维度产生 $(-1,1)$ 区间的随机标准偏差，并用标准差为 σ 的高斯滤波器对各维度的偏差矩阵进行滤波，最后用放大系数 α 控制偏差范围。一般来说， α 越小， σ 越大，产生的偏差越小，和原图越接近，如图2.2。

弹性变换的具体操作如下：首先创建随机位移场来使图像变形，即 $\Delta x(x,y) = \text{rand}(-1,+1)$ 、 $\Delta y(x,y) = \text{rand}(-1,+1)$ ，其中 $\text{rand}(-1,+1)$ 生成一个在 $(-1,1)$ 之间均匀分布的随机数。然后用标准差为 σ 的高斯函数对 Δx 和 Δy 进行卷积。如果 σ

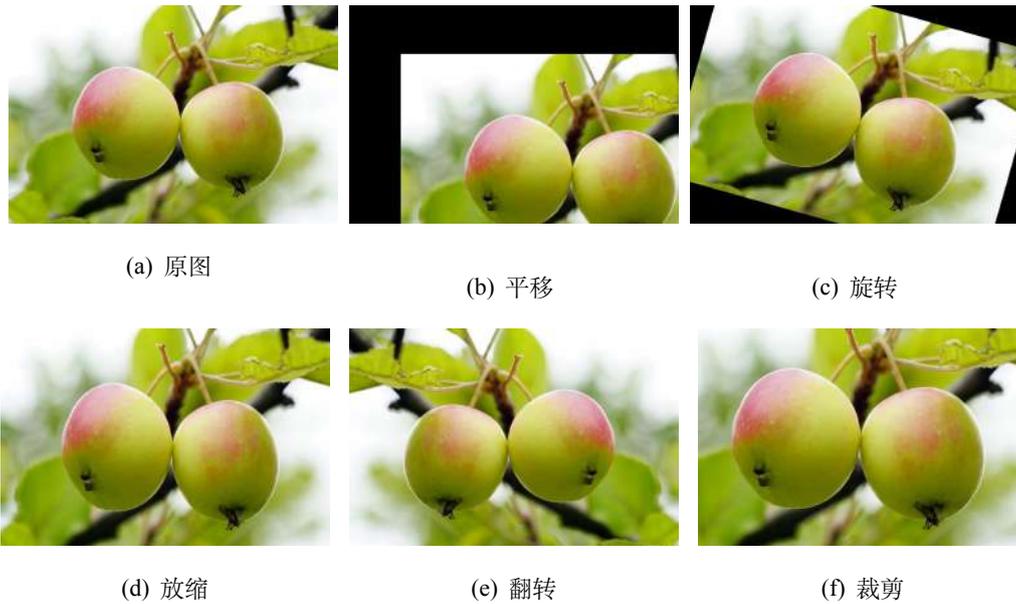


图 2.1: 仿射变换

很大，则卷积结果值就很小，因为随机值平均为 0，此时若进一步将位移场标准化（达到 1 的范数），则该字段接近常数，具有随机方向。如果 σ 很小，那么归一化后位移场看起来像一个完全随机的字段。对于中间的 σ 值，位移场效果最接近弹性变形，其中 σ 是弹性系数。最后将位移场乘以控制变形强度的比例因子 α 。在 MNIST 实验（输入图像尺寸为 29×29 ）中，产生最佳结果的值是 $\sigma=4$ 和 $\alpha=34$ 。



图 2.2: 弹性变换

2.1.2 颜色空间变换

1、RGB 空间

数字图像处理存在六大颜色空间，根据人眼识别定义出的空间是 RGB 空间，即将每张图像中的每个像素点看做由 R、G、B 三个分量组成的。任意一种颜色都可以通过红、绿、蓝三原色按不同比例混合得到，而每种颜色分量的灰度

级范围被确定在 0-255 之间，因此彩色图片中的每个像素点都可以使用一个具有三个数字的向量进行表示。

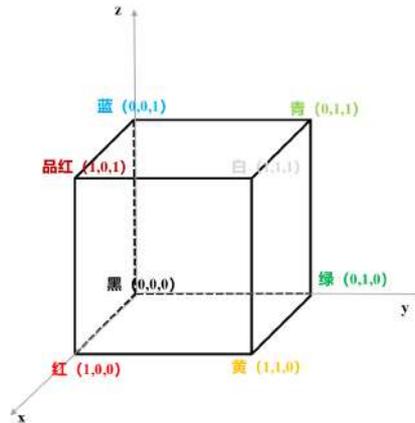


图 2.3: 归一化的 RGB 空间图

RGB 模型是一个立方体，原点对应黑色，离原点最远的顶点对应白色，如图 2.3。将其对应到笛卡尔坐标系中，RGB 颜色分量分别对应三个坐标轴。此种颜色空间一般用于显示器等设备，但由于存在直观性差的问题，很难从 RGB 值中得知所代表颜色的认知属性，因此该颜色空间并不符合人对颜色的感知心理。另外，RGB 颜色空间是最不均匀的颜色空间之一，两种颜色之间的直觉差异不能采用两个颜色点之间的距离表示。

2、HSI/HSV 空间

HSL 是面向视觉感知的彩色模型，用 H (Hue)、S (Saturation)、L (Lightness) 这三类参数来描述颜色特性。其中 H 表示颜色的频率，称为色相；S 表示颜色的深浅程度，称为饱和度；L 表示亮度。人们在观察一个彩色物体时，直觉上会使用色调、饱和度和亮度来描述其颜色。色调用于刻画颜色属性，例如红色、蓝色以及青绿色等；饱和度实际度量了某一种颜色被白光所稀释的程度；亮度则是一类比较主观的描述，很难用数学的方式测量，其作为感官对于色彩描述的关键，体现了无色的强度概念。相对应地，强度（灰度）是灰度图最为有效的描述子，它既可以被测量也很容易解释。因此提出了 HSI（色调、饱和度、强度）彩色模型，该模型能够从彩色图像所携带的彩色信息（色调与饱和度）里消去强度分量的影响，非常适合彩色特性的检测与分析。因为 H 和 S 分量与人感受颜色的方式密切相关，其对于色彩的描述直觉上自然贴切，所以 HSI 模型也被广泛应用于基于色彩描述的图像处理方法之中。

彩色图像可以采用 RGB 或 HSI 等颜色模型来描述，二者之间存在着严格的数学关系，可以相互转换。实际应用时，应该根据需要进行适当的颜色模型。

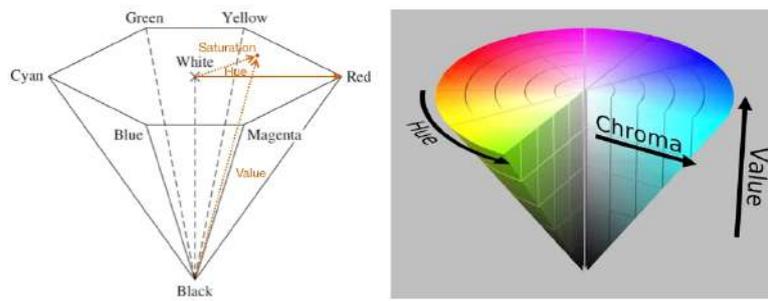
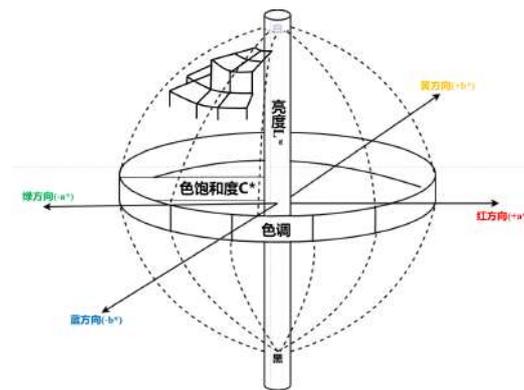


图 2.4: HSV 空间图

HSV 模型与 HSI 模型相比，和人类对颜色的感知更接近。H 代表色相，S 代表饱和度，V (Value) 代表色明度。HSV 模型的坐标系统可以是圆柱坐标系统，但一般用六棱锥来表示，与 HSI 模型比较相似，如图 2.4。六边形边界表示色调，六个角分别代表 RGB 三原色以及他们的补色，水平轴表示饱和度，纵轴表示亮度。可以通过比较 HSI、HSV 与 RGB 空间的转换公式，来比较 HSI 与 HSV 的区别。

3、 $L^*a^*b^*$ 空间

Lab 色彩空间是由国际照明委员会 (CIE) 于 1976 年制定的，图 2.5 展示了 Lab 色彩空间的三维立体空间模型图。它通过 L^* 、 a^* 、 b^* 三个互相垂直的坐标轴，来表示一个色彩空间： L^* 轴表示亮度，其中黑色在底端，白色在顶端； a^* 轴是红—绿色轴， $+a^*$ 表示品红色， $-a^*$ 表示绿色； b^* 轴是黄—蓝色轴， $+b^*$ 表示黄色， $-b^*$ 表示蓝色。

图 2.5: $L^*a^*b^*$ 立体空间图

利用 L^* 、 a^* 、 b^* 三个数值可以描述自然界中的任何颜色，而这些颜色的色调及特征都可以通过 a^* 、 b^* 的数值来表示。与其他色彩空间不同， L^* 通道值的范围在 0-100 之间，其中 0 表示黑色，100 表示白色。而 a^* 和 b^* 的取值范围是

$[-100, 100]$ 。在该颜色空间中，可以方便地对颜色进行各种操作。如把 a^* 或者 b^* 的数值取反，使其变成 $-a^*$ 和 $-b^*$ ，那么反映在图像上就是进行了相反色的操作。如果仅输出 L^* 通道，则可以用彩色原图复制单色。

研究一个 Lab 模式的三通道图像可以发现， a^* 、 b^* 两个通道仅仅定义了图像的色彩部分而忽略了其形态细节。因此在 Lab 模式的图像中，颜色和层次是分离的。但这也使得图像的灰色曲线和彩色曲线可以分开调整：使用不包含颜色信息的 L^* 通道确定图像中的最亮值和最暗值，调整灰度曲线，可以保证彩色部分不受影响。这样既能够避免在色彩校正过程中出现一些跳跃性的、阶调不连续的颜色，也能够只在 L 通道上锐化图像来强调图像的整体细微层次。

由于 $L^*a^*b^*$ 颜色空间包含了 RGB 和 CMYK 颜色空间，因此当色彩数据从大的颜色空间转换到比它小得多的空间时，不会因为数据量的缺失而引起色彩的偏差，从而造成色彩损失。

4、其他颜色空间

其余颜色空间，例如 YUV、YCbCr (YCC)、YPbPr 等通常会被用于影像连续处理或数字摄影系统中，即应用于电视系统、数位摄影等地方。其中的 Y 分量都是表示明亮度 (Luminance、Luma)。

在 YUV 颜色模型中，U、V 表示的是色度 (Chrominance/Chroma)。YUV 是欧洲电视系统所采用的颜色模型 (属于 PAL 制式)，颜色被分为一个亮度信号和两个色差信号进行传输。YCbCr (简称 YCC) 中，Cb 和 Cr 分别为蓝色和红色的浓度偏移量成分。YCbCr 是 YUV 压缩和偏移的版本。

YPbPr 类似于 YCbCr，但 YPbPr 选用的 CIE 色度坐标略有不同。一般 SDTV 传输的色差信号被称作 Cb、Cr，而 HDTV 传输的色差信号被称作 Pb、Pr。

2.1.3 像素点变换

像素点变换，其又分为人为因素和模拟镜头失真。人为因素包括增加噪声 (高斯噪声、泊松噪声、拉普勒斯噪声)、随机擦除和 RGB 通道值更改。一般来说，噪声的概率密度会服从某种分布，因此根据概率分布可以分为高斯噪声、泊松噪声、拉普勒斯噪声。为原图加入某种概率关系下的噪声数据，即可达到人为增加噪声的目的。随机擦除是按照一定比例修改图片中的像素点，随机将某一通道或几个通道的数值置为 0。当 RGB 三通道全部被置为 0 时，该像素点呈黑色。一般来说，随机擦除的范围和概率可以手动设定。RGB 通道值更改是将所有像素的某一通道值全部增加或减少某一数值，如图 2.6(c) 就是将图像中每个像素点的 R 通道 (红色) 数值随机增加 10-100 之后的结果。

模拟镜头失真包括模糊 (均值模糊、高斯模糊、中值模糊)、亮度更改、锐



图 2.6: 人为因素

度更改、对比度更改和直方图均衡化等。



图 2.7: 模拟镜头失真

模糊可以在图像预处理时减少噪声，主要基于卷积运算。对于均值、高斯、中值模糊来说，卷积算子计算都是线性操作，因此又叫线性滤波。在图像上对目标像素给一个模板，该模板包括了其周围的邻近像素，用模板中全体像素的平均值代替中心像素值的方法为均值模糊；用数据从小到大排序后的中间值代替中心像素值的方法叫做中值模糊；用高斯函数确定的权重与模板邻域内像素进行加权平均得到的灰度值，去替代模板中心像素值的方法被称作高斯模糊。

噪声所在的频段主要在高频段，而图像边缘信息也主要集中在高频部分。这就会导致原始图像在经过平滑处理之后，物体的边缘和轮廓变得模糊。为了避免上述不利因素的影响，我们可以利用图像锐化技术，来清晰化物体的边缘。图像锐化的实现分为高通滤波和空域微分。前者采用高通滤波的方式让图像中的

高频分量顺利通过，并适当抑制中低频分量，使图像的细节更加清楚。后者使用一阶微分运算，突出图像中物体的边界，以此实现图像的锐化。这里的一阶微分主要指梯度模运算，而图像的梯度模值实际上包含了边界及细节信息。

当图像过曝或者光线不足时，图像中的内容也会显得模糊而难以分辨，这时就需要调节图像的亮度和对比度，来提升图像整体的质量。亮度调整实际是图像的明暗变化，对应到像素点上就是 RGB 值的大小。当 RGB 三通道值全为 255 时，像素点最亮，呈白色；当 RGB 三通道值全为 0 时，像素点最暗，显示为黑色。此外也可以通过转换色彩空间后，更改对应的亮度通道值实现。而对比度则是不同像素点之间的差值，差值越大则对比越明显。图像亮度和对比度的改变可以根据公式 2.1 实现。其中 (i, j) 表示像素点位置， f 和 g 是调整前后的像素值大小。亮度调整依靠增益变量 β ，提高亮度需要扩大增益变量，使像素值向白色靠近，反之亦然。而提高对比度就是增加像素点之间的差值，因此需要调整对比度系数 α ，该值大于 1 时图像整体变亮，小于 1 时图像整体变暗。

$$g(i, j) = \alpha f(i, j) + \beta \quad (2.1)$$

当一幅图像的灰度直方图几乎覆盖了整个灰度范围，并且除了个别灰度值的个数较为突出外，整体灰度值近似于均匀分布时，这幅图像就具有较大的灰度动态范围和较高的对比度，同时图像细节更为丰富。因此，直方图均衡化的目的就是将原图的直方图分布变为近似于均匀分布，从而增强图像的对比度。由于是对灰度级进行处理，因此直方图均衡化一般用于灰度图。而对于彩色图像，较为直接的处理方式是抽取 R、G、B 分量，分别进行直方图均衡化，如图 2.7(e)。这样处理虽然效率很高，但是会产生背景杂讯，导致色调改变及失真。因此还可以将 RGB 图像映射到 HSI 空间，仅对 S 和 I 分量做直方图均衡。

2.2 病斑特征分析

由于水果病斑存在多样性和差异性，因此在研究过程中需要针对特定的病害类型进行扩增。其中水果炭疽病最为常见，且危害的水果种类十分广泛，而其性状表征在不同品种间又不存在显著差异，因此考虑将炭疽病作为病斑扩增的具体表现。

2.2.1 水果炭疽病性状表征

市面上常见的果树作物基本都可能会遭受炭疽病的侵袭，如苹果、葡萄、柑橘等。除果树外，大部分的果蔬作物以及园林植物也深受其害。炭疽病对发育到中后期的果实，尤其是即将成熟的果实危害最大，同时由于其分生孢子盘的传

染性，也会对叶片、茎蔓等造成危害。因此本节中，我们对水果炭疽病在叶片、果实、茎蔓和枝干上，不同周期下的性状表征进行了详细描述。

水果炭疽病在叶片上的特征表现为：刚刚开始侵染时，会在叶片上形成一个针尖大的小斑点，在斑点旁边会有一些黄色的晕圈，并且一般是从叶片边缘或者叶脉中心处开始侵染的，如图2.8(a)。随着叶片受侵害时间的增长，病害范围会逐渐扩大，形成一个圆形或椭圆形的病斑。病斑具有轮纹斑纹，呈深褐色至灰白色，病部中央散生或轮生黑褐色小点，如图2.8(b)。等到染病后期，会导致病叶黄化脱落。



(a) 发病早期



(b) 发病晚期

图 2.8: 叶片炭疽病性状表现

此外根据染病时天气情况的不同，叶片感染的炭疽病也可分为急性炭疽病（叶枯型）和慢性炭疽病（叶斑型）。急性炭疽病一般发生在高温、持续阴雨的天气，病情多从幼嫩叶的叶尖、叶缘开始发病，病部会从淡青略带褐色的水烫状小斑，逐步扩大为轮纹状的水渍大斑，潮湿天气时还会出现粉红色胶状物。急性炭疽病发病十分迅速，会使病叶快速腐烂脱落，如果防治不及时会造成整株大面积的落叶现象。而慢性炭疽病一般发生在短期阴雨后转晴的天气，发病初期病叶叶尖、叶缘会产生黄褐色或灰白色的病斑。随着病情发展，病斑会逐渐扩大，受害叶片病部颜色变为深褐色，病斑表现为隆起、轮纹状。

果实方面，染病初期果皮上出现褐色至黑褐色病斑，呈圆形或近圆形，如图2.9(a)。然后病部斑块会逐渐扩大、色泽加深、软腐凹陷，同时在病斑中央会出现无数稍由褐变黑的小粒点，整体病斑呈同心轮纹状排列，如图2.9(b)。随着病情的发展，病斑腐烂并在果实上蔓延扩大。严重时，病部会一直腐烂到果实中心，使果肉变褐、味道变苦。同时也会使大部分甚至整个果实腐烂，造成落果，或者在枝条上皱缩形成僵果。此外，如果在温暖潮湿的环境下，病菌会突破表皮，涌出一层粉红、绯红色的黏稠液物质。一个病果常有多个病斑，病斑连片后会导致全果变黑、腐烂，或者出现干疤和泪痕。在储存期的果实也会被炭疽病侵

染，从果蒂或附近开始发病，最开始是淡褐色水浸状，然后颜色变深并扩大。



(a) 发病早期

(b) 发病晚期

图 2.9: 果实炭疽病性状表现

对于感染炭疽病的茎蔓，病斑会呈黑色的圆形或纺锤形，形态凹陷。且病斑上有许多黑色小点，即病原的分生孢子盘。例如在草莓茎蔓上发病时，茎部感染会出现纺锤形或椭圆形的深褐色斑点，呈溃疡状，稍凹陷。当病斑围绕匍匐茎和叶柄一圈时，病斑以上部位会干缩枯死。在高湿条件下，病斑上长出粉红色黏质孢子堆，后期变成污黑色。如果侵染到根茎部位，就会引起草莓的全株死亡。

部分果树枝梢也可感染，例如柑橘等。其中慢性炭疽病主要是受冻的秋梢易得，它从枝梢中部开始，然后向下蔓延。最开始病斑是椭圆形的淡褐色，然后呈现出梭形、稍微有些凹陷。当病斑环绕枝梢以后，上部的枝梢会迅速干枯死亡，也可在顶梢向下枯死。而高温高湿环境下，嫩梢嫩枝会表现为从上到下、水烫状的急性萎缩凋零状态；受害枝梢叶腋周边，会出现下陷的椭圆近似菱形的褐色病斑。随着病情的发展，病斑会绕着枝梢茎秆蔓延扩大，直到枝梢干枯死亡。病部表现为灰褐色到灰白色，并有非常细小的黑色斑点。

综合上述的调研结果可以发现，水果炭疽病会根据感染周期的不同，呈现出变化的性状表征；并且由于分生孢子盘存在传染性，因此在病害监测时可以通过果实、叶片等不同部位的表现，共同确定植株感染炭疽病的可能性。种植端的图像传感器由于拍摄精度和位置的影响，通常难以对植株的茎蔓和枝干进行捕捉，而果实和叶片由于染病风险最大且病害表现易于察觉，将作为本研究的扩增重点。

2.2.2 颜色特征提取

颜色特征是基于像素点的特征，由于所有的像素点对图像都有贡献，因此颜色特征是全局特征，描述了图像或图像区域所对应景物的表面性质，且对图

像的方向、大小等变化不敏感。

对于水果病斑图片而言，病斑区域与正常区域的颜色区分度最大，视觉对比最明显。相关研究 [35] 也发现，可以通过叶片或果实的色调范围，判断其是否染病。因此提取颜色特征，可以在后续使用特征模拟进行病斑图像扩增时，起到指导作用。

1、颜色直方图

颜色直方图是最常用的表达颜色特征的方式，被广泛应用于许多图像检索系统中。它主要描述不同色彩在整幅图像中的占比，而并不关心其所处的空间位置。其优点在于直方图结果不受图像旋转或平移等变化的影响，如果借助归一化还可以进一步避免受图像尺度变换的影响。不过缺点也在于，其不能有效表达颜色空间分布的信息。

可以基于不同的颜色空间和坐标系构建颜色直方图，计算时需要将颜色空间划分成若干个小颜色区间，这些区间对应的就是直方图横轴上的最小间隔，整个过程被称为颜色量化 (color quantization)。然后，通过计算颜色落在每个小区间内的像素点数量就能得到颜色直方图。具体来说，就是颜色直方图中的 x 轴表示像素值，y 轴表示某一像素值范围内像素点的个数。通常颜色直方图是基于 RGB 色彩空间的，因此可以得到三个通道上不同的像素分布情况。

除了将颜色空间中各个维度的分量均匀划分的方法外，颜色量化还包括许多其他方式，例如聚类、向量量化以及神经网络方法等。相比之下，使用聚类的方式可以考虑到图像颜色特征在整个空间中的分布情况，从而避免某些区间的像素数量异常稀疏的情况出现，使量化更为有效。

但上述的颜色量化方法也存在一定的问题，例如两幅图像的颜色直方图几乎相同，但互相错开了一个 bin，此时如果使用欧拉距离或者 L_1 距离来计算二者的相似度，则会发现相似度值很小，与感官认知产生差异。因此为了避免这个问题，可以考虑采用二次式距离或对颜色直方图进行事先平滑过滤的方式，来衡量相似但不相同颜色之间的相似度。

2、颜色矩

颜色矩是一种简单有效的颜色特征表示方法，它以数学方法为基础，通过计算矩来描述颜色分布。由于颜色信息主要分布在低阶矩阵中，因此用一阶矩、二阶矩和三阶矩足以表达图像的颜色分布。

一阶颜色矩利用一阶原点矩，计算每个颜色分量的均值，如公式2.2，反映了图像整体的明暗程度。二阶颜色矩采用二阶中心矩的平方根，计算每个颜色分量的方差，如公式2.3，可以反映图像颜色的分布范围或不均匀性。三阶颜色

矩通过三阶中心矩的立方根，计算每个颜色分量的偏斜度，如公式2.4，来反映图像颜色分布的对称性。

$$\mu_i = \frac{1}{N} \sum_{j=1}^N P_{ij} \quad (2.2)$$

$$\sigma_i = \left[\frac{1}{N} \sum_{j=1}^N (P_{ij} - \mu_i)^2 \right]^{\frac{1}{2}} \quad (2.3)$$

$$\zeta_i = \left[\frac{1}{N} \sum_{j=1}^N (P_{ij} - \mu_i)^3 \right]^{\frac{1}{3}} \quad (2.4)$$

其他提取颜色特征的算法，大多需要对图像进行量化处理，而量化处理容易导致误检验，且产生的图像特征维数较高，不利于检索。使用颜色矩的方法无需进行颜色空间量化，特征向量维数低，便于计算。

2.2.3 形状特征提取

在物体区分时，形状特征与尺度测量可以结合起来，作为区分不同物体的依据。与之对应地，感染不同病害而形成的水果病斑，在视觉上也可能以不同的形状呈现。因此形状特征作为区别于其他病症的条件之一，也可以视作特定病害的固有性状进行分析。

简单的形状特征根据基础特征值计算而来，例如圆形度、矩形度和复杂度等，在理解时也比较简单。但如果想要更加全面地提取形状特征，则需要根据边界、区域等进行描述。在对水果炭疽病的性状表征进行分析后可以发现，果实和叶片上的炭疽病斑块，常呈圆形或椭圆形。因此本研究基于病斑边界对斑块的形状特征进行提取，以基础特征值——圆形度和复杂度，以及 Hough 变换作为指标，计算其相关值。

1、圆形度

图像的像素不同于坐标轴上的点，其坐标分离并不连续（坐标只能为整数）。因此在描述各边界点像素的坐标时，需要引入链码的定义。所谓链码，其实是图像处理中用于表示图像中线条、平面以及轮廓的编码方式。描述曲线的链码用曲线起始点的坐标和边界点斜率来表示，而区域边界轮廓可理解为相邻边界像素之间的单元连线逐段相连而成，且边界链码具有方向性。

按照中心像素点邻接方向个数的不同，常用的链码还可以分为 4 连通链码和 8 连通链码，后者能够准确地描述中心像素点与其邻接点的信息，因此使用更加频繁。

通过链码可以表示出病斑的边界信息，从而计算病斑的周长与面积。而圆形度正是通过这二者，来衡量病斑区域与标准圆形的相似程度。圆形度又可以

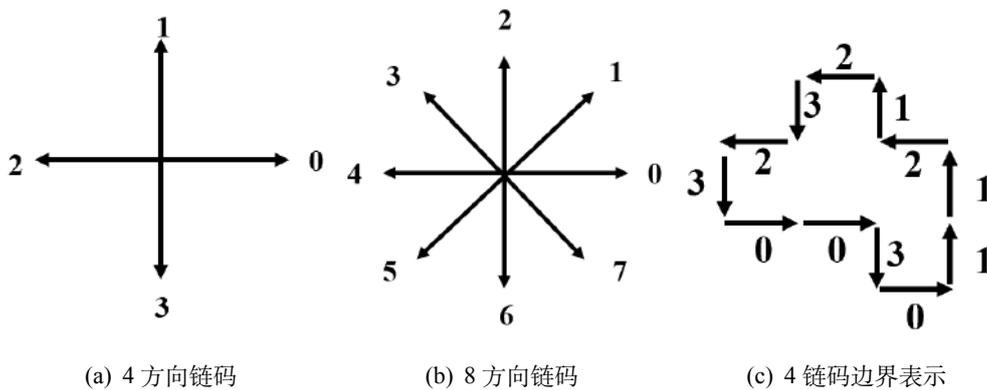


图 2.10: 链码及边界表示

称为区域的紧凑性，假设病斑的周长为 L ，面积为 S ，则圆形度的计算被定义为公式2.5。

$$Roundness = \frac{4 * \pi * S}{L^2} \quad (2.5)$$

根据上述的计算公式可以发现，当病斑区域的形状越接近圆形时，其圆形度的计算值就越接近 1。而在面积相同的情况下，具有光滑边界的形状周长更短，圆形度更大，表明该形状更密集。

2、复杂度

复杂度同样根据病斑的面积和周长值进行计算，描述了病斑区域形状的复杂程度，具体公式见2.6。

$$Complexity = \frac{L^2}{S} \quad (2.6)$$

3、Hough 变换

Hough 变换作为形状特征提取的方法，被广泛应用在图像分析、电脑视觉以及数位影像处理中，它通过一种投票算法检测具有特定形状的物体。经典霍夫变换是用于检测图像中的直线，后来霍夫变换扩展到任意形状物体的识别，例如圆和椭圆。

理论上，直线检测的霍夫变换就是对于原图中的每一条直线都映射到新的参数空间中去，找出参数空间里变换线密集的地方，最后对应回原始的空间坐标系。

具体实现时，可以先对图片进行边缘检测，然后将图像上每一个非零像素点，在参数坐标系中变换为另一条线。运用两个坐标空间之间的变换，将在原空间中具有相同形状的曲线或直线映射到另一个坐标空间的一个点上形成峰值，从而把检测任意形状的问题转化为统计峰值问题。

例如一条直线在直角坐标系下可以用 $y = kx + b$ 表示，经过霍夫变换后将该方程的参数和变量交换。即直线 $y = kx + b$ 上某一点在直角坐标系下的位置坐标为 (x_1, y_1) ，将其映射到参数空间后则表示为 (k, b) 。由于同一条直线上的点在参数空间的坐标表示相同，因此在直角坐标系下属于同一条直线的点会在参数空间形成多条直线并内交于一点，故可用该原理进行直线检测。

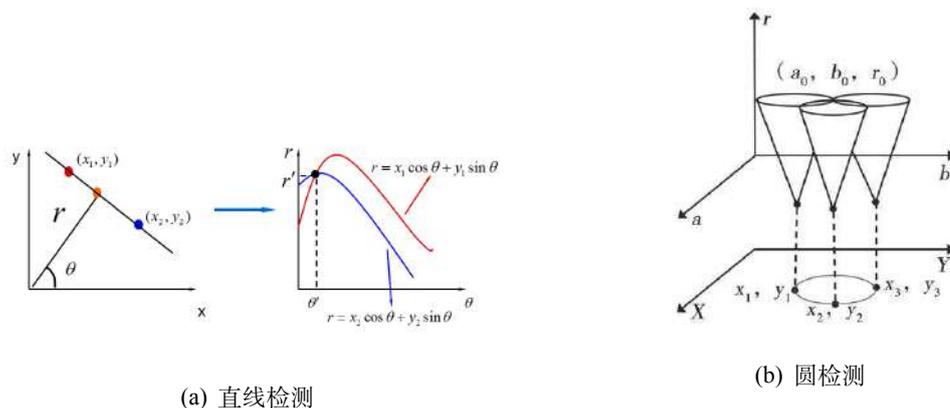


图 2.11: 霍夫变换

Hough 变换除了适用于直线检测外，还适用于任何形式通过 $f(x, a) = 0$ 表示的图形的检测，其中 x 表示坐标向量， a 表示系数向量。对于一个半径为 r ，圆心为 (a, b) 的圆，可以将其表示为：

$$(x - a)^2 + (y - b)^2 = r^2 \quad (2.7)$$

因此对于圆形上的一点 (x, y) ，可以用三维参数 (a, b, r) 表示，对应到参数空间就是一个圆锥。而原空间的一个圆就对应着圆锥相交的那个点，该特定点在参数空间的三维参数一定，因此对应着固定半径和固定圆心坐标的图像空间的圆。

但三维空间的变换操作比较费时，为了加快 Hough 变换检测圆的速度，可以利用图像梯度信息。首先对圆标准方程中的 x 求导，得到：

$$2(x - a) + 2(y - b) \frac{dy}{dx} = 0 \quad (2.8)$$

这样压缩三维参数空间到二维，可以减少圆检测的计算量。但由于该方法使用了边界斜率，斜率的量化误差会导致圆检测的抗干扰性能减弱，从而降低检测精度。

2.2.4 纹理特征提取

纹理是物体表面的固有特征之一，可以认为是颜色或灰度在空间内以一定形式变化而产生的图案或模式。炭疽病病斑表面根据发病周期的不同，也会有形态各异的纹理表现。例如急性炭疽病的叶片表面会从水烫状小斑，逐步扩大为云纹状的水渍大斑。而慢性炭疽病叶片表面在后期会逐渐扩大，呈现出隆起的轮纹状。因此纹理特征提取，也是区分病斑阶段的重要因素。

1、局部二值模式

局部二值模式 (local binary pattern, LBP) 是一种用于描述图像局部纹理特征的算子，作用是进行特征提取。

原始的 LBP 算子定义在 3×3 的窗口内，以窗口中心像素点的值为阈值，将相邻 8 个像素的灰度值与其进行比较。若其值小于中心像素值，则将该像素点位置标记为 0，否则为 1。由此对 3×3 邻域内的 8 个点全部进行标记后，可以得到 8bit 的无符号数，即为该中心像素点的 LBP 值，反映了窗口内的纹理信息。

而对于图像的 LBP 特征向量提取需要使用统计直方图表示，具体来说就是首先将待处理图像按照 16×16 的区域 (cell) 划分，以 cell 中的每个像素点为中心都可以按照上述方式得到一个 LBP 值。统计这 256 个 8 位二进制数中每个数字出现的频率，可以得到当前 cell 的统计直方图，然后进一步归一化处理。最后将每个 cell 的直方图相连形成的特征向量，就是整幅图像的 LBP 纹理特征向量。

LBP 具有亮度变化鲁棒性和灰度尺度不变性，即窗口内像素的灰度值出现了亮度的线性变换或非线性变换也不会影响其 LBP 值，因为中心像素的灰度值与 8 邻域内像素的灰度值的大小关系不变。但基本的 LBP 算子只能覆盖固定半径范围的区域，无法满足不同尺寸和频率纹理的需要，因此 LBP 还存在许多改进的版本：

(1) 圆形 LBP 算子

为了适应不同尺度的纹理特征，改进后的圆形 LBP 算子允许在半径为 R 的圆形领域内有任意多个像素点，从而得到诸如半径为 R 的圆形区域内含有 p 个采样点的 LBP 算子。几种常用的 LBP 形式如图 2.12 所示，其中 LBP_p^r 表示在以 r 为半径的圆形领域内，含有 p 个采样点的 LBP 局部编码。

(2) LBP 旋转不变模式

原始的 LBP 算子是灰度不变的，但却不是旋转不变的，因为图像经过旋转后就会得到不同的 LBP 值。为了实现旋转不变性，可以不断转动圆形领域得到一系列初始定义的 LBP 值，取其最小值作为该领域的 LBP 值，如图 2.13。

(3) LBP 等价模式

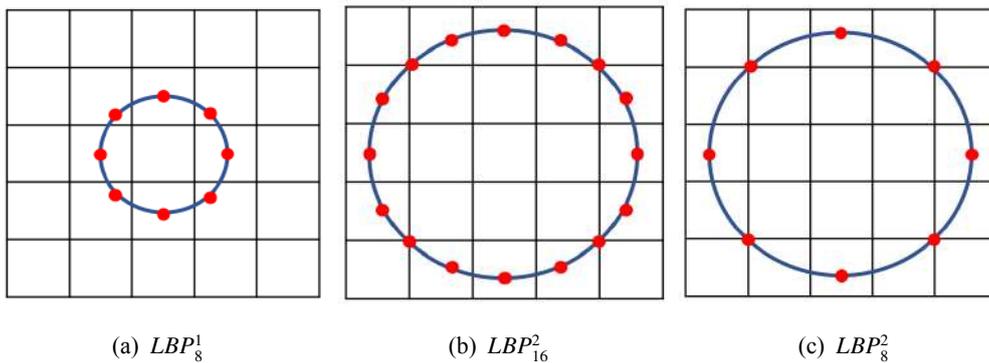


图 2.12: 圆形 LBP 算子

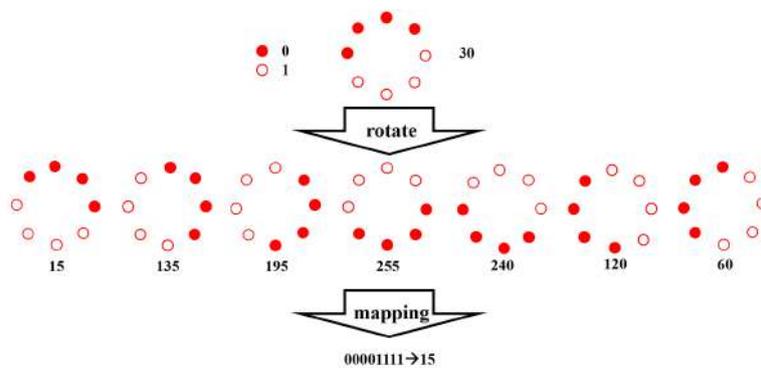


图 2.13: 旋转不变的 LBP 算子

对于半径为 R 的圆形区域内含有 p 个采样点的 LBP 算子，将会产生 2^p 种二进制表示模式。随着区域内采样点数的增加，模式的种类数量随之急剧增多，这不利于纹理的表达。如果将 LBP 算子用于纹理分类或人脸识别时，会采用 LBP 模式的统计直方图来表达图像信息，而如此庞大的模式种类将使得数据量过大，导致直方图过于稀疏。因此，LBP 等价模式对原始的 LBP 进行降维，使得在数据减少的情况下依旧能完整地表达图像信息。

2、灰度共生矩

灰度共生矩阵 (GLDM) 是在假定图像中各像素间的空间分布关系包含了图像纹理信息的前提下，提出的具有广泛性的纹理分析方法。它是像素距离和角度的矩阵函数，通过计算图像中一定距离和一定方向的两个像素点灰度之间的相关性，来反映图像在方向、间隔、变化幅度以及变化快慢上的综合信息。

灰度共生矩阵实质上是一幅图像中两个像素灰度级的联合直方图，是一种二阶统计量。普通的灰度直方图是对图像上单个像素具有的灰度值进行统计的结果，而灰度共生矩阵是对图像上保持一定距离的两个像素，所分别具有的灰

度状况进行统计得到的：

(1) 取 $N \times N$ 图像中的任意一点 (x, y) ，以及偏离它的另一点 $(x + a, y + b)$ ，设该点对的灰度值为 (g_1, g_2) ；

(2) 令点 (x, y) 在整个图像上移动，则会得到各种 (g_1, g_2) 值，设该灰度图像的灰度值级数为 k ，则 (g_1, g_2) 的组合共有 k^2 种；

(3) 对于整个图像矩阵，统计出每一种 (g_1, g_2) 值出现的次数，然后排列成一个方阵，再用 (g_1, g_2) 出现的总次数将它们归一化为出现的概率 $P(g_1, g_2)$ ，即可得到灰度共生矩阵。

对于纹理变化缓慢的图像，其灰度共生矩阵对角线上的数值较大。而对于纹理变化较快的图像，其灰度共生矩阵对角线上的数值较小，对角线两侧的值较大。

尽管一幅图像的灰度共生矩阵能够反映出图像灰度关于方向、相邻间隔、变化幅度等方面的综合信息，但由于灰度共生矩阵的数据量通常比较大，因此一般不会直接对矩阵进行操作，而是基于它构建一些统计量来描绘原图像的特征信息，作为区分纹理的指标。如能量、熵、对比度、均匀性、相关性、方差、差平等，下面就本研究中提及的特征量进行详细阐述：

对比度 可以用于反映图像的清晰度和纹理的沟纹深浅，它实际上度量的是图像灰度矩阵的值是如何分布的，以及图像中局部变化的程度。纹理的沟纹越深，反差越大，图像局部越清晰。

$$Contrast = \sum_i \sum_j (i - j)^2 P(i, j) \quad (2.9)$$

角二阶矩 又称为能量，能量变化反映了图像灰度分布的均匀程度和纹理的粗细度。若灰度共生矩阵的元素值相近，则能量较小，表示图像纹理较为细致。若其中一些值大而其他值小，则能量值会比较大。能量值大表示一种较为均匀，且变化比较规则的纹理模式。

$$ASM = \sum_i \sum_j P(i, j)^2 \quad (2.10)$$

熵 图像包含信息量的随机性度量。当共生矩阵中所有值均相等或者原图像矩阵的像素值表现出最大随机性时，熵最大。因此熵值表明了图像灰度分布的复杂程度，熵值越大则图像越复杂。

$$Entropy = - \sum_i \sum_j P(i, j) \log P(i, j) \quad (2.11)$$

反差分矩阵 反映了图像纹理局部变化的大小。若图像纹理的不同区域之间较为均匀，变化缓慢，那么逆方差的值会比较小。

$$IDM = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \frac{P(i, j|d, \theta)}{1 + (i - j)^2} \quad (2.12)$$

相关性 反映了图像纹理的一致性。如果图像中有水平方向的纹理，则 0° 灰度共生矩阵的相关性值会大于其他几个角度方向的灰度共生矩阵的相关性值。

2.3 数字图像处理

由于水果数据图像采集场景的复杂，在图像扩增的操作前以及过程中都会涉及相当一部分的图像处理工作。而在计算机视觉领域，一般会将图像看作二维的数字信号，并将数字信号的处理方式移植到图片处理中，实现阈值化、卷积滤波等操作，以达到预期效果。下面简要介绍了本研究中使用到的一些数字图像处理方法。

2.3.1 滤波

所谓滤波，可以将其看作为一种过滤算法，应用到图像领域，就是滤除掉图片中不需要的像素点。其原理就是对图像中的每个像素点，使用滤波矩阵作为卷积核，进行逐个元素相乘再求和的操作。因此卷积核（滤波矩阵）决定了最终得到的图像。

根据功能的不同，可以将图像滤波简单分为平滑滤波和锐化滤波；而根据特点，又可以分为线性与非线性。一般来说，平滑滤波可以减弱或消除图像中的高频分量，但不影响低频分量，由此达到消除噪声的效果。其典型实现有均值滤波（如图2.14(a)）、中值滤波等。而锐化滤波可以减弱或消除图像中的低频分量，但不影响高频分量，因此能增强被模糊的细节，使目标边缘变得明显。典型的锐化滤波包括基于拉普拉斯算子的二阶微分锐化（如图2.14(b)）、非锐化掩模等。

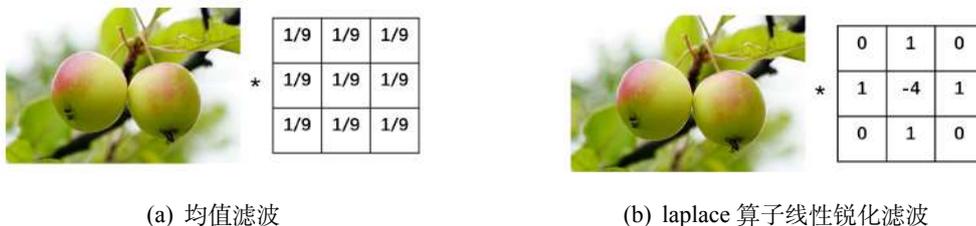


图 2.14: 滤波原理

2.3.2 边缘检测与边界提取

使用边缘线来表示图像信息，可以简化图像、方便后续处理，同时还能消除图像数字化时所混入的噪声，因此边缘检测和边界提取，是数字图像处理中非常重要的部分。

对于边缘检测而言，其目的是找到图像中亮度变化剧烈的像素点构成的集合，表现出来往往是轮廓。在对现实世界的图像采集中，有以下几种情况在图像中会形成一个边缘：(1) 物体深度或表面方向不连续，例如物体处在不同的平面上、同一物体的不同侧面；(2) 物体的材料不同，这会导致光在其表面的反射系数存在差异，如雨天路面的积水；(3) 场景中受到的光照强度不同，例如太阳在地面上投有树荫。

因此在灰度图中，物体在轮廓处会具有不同的灰度值，从而产生灰度跳变。图像边缘检测的原理就是检测出图像中所有灰度值变化较大的点，并将这些点连接起来构成若干线条，作为图像的边缘。上一小节提到滤波能够使边界更加清晰，因此可以通过滤波来提取图像特征。而边缘检测本质上就是一种滤波算法，区别在于滤波器的选择，但滤波规则完全一致。

基本的边缘检测滤波器包括 Sobel、Roberts、Prewitt、Canny 和 Marr-Hildreth 算子。其中 Roberts 算子、Sobel 算子、Prewitt 算子运算效率高，对噪声有一定抑制作用，缺点在于检测出的边缘质量较低，如边缘粗糙、定位误差、间断点多等。Canny 算子不容易受噪声干扰，得到的边缘精细且准确，适合要求高精度的应用场景，但运算代价较高，处理实时图像比较困难。Marr-Hildreth 算子虽然边缘检测效果相对比较好，但是对于噪声比较敏感。

边缘检测的结果会返回仅将物体边缘像素置为白色的黑色背景 mask，在此基础上我们还需要进一步的边界提取。由于 mask 是二值图像，因此可以将所有物体内部的点删除（置为背景色）。具体来说就是首先逐行扫描 mask 图像，如果发现一个黑色像素的 8 个邻域都是黑色像素，则该像素点为内部点，在目标图像中将其删除。实际上这相当于采用一个 3×3 的结构元素对原图进行腐蚀，使得只有那些 8 个邻域都是黑色的内部点被保留。再用原图像减去腐蚀后的图像，即可删除内部点、留下边界像素。

2.3.3 腐蚀和膨胀

腐蚀和膨胀操作是形态学处理的基础，主要针对的是二值图像（0 或 1）。图像腐蚀类似于白色区域被“蚕食”，它会缩减细化图像中的高亮区域或白色部分，使得到的结果比原图的高亮区域更小。腐蚀操作的输入对象主要包括二值图像和卷积核，卷积核遍历到的像素位置，如果其周围全部是白色则保留白色，否则

变为黑色。如果目标物体与背景相邻，那么图像腐蚀操作会使其外围区域被归并到背景中去，从而使得物体边缘向内部收缩，因此可以去除图中小区域的无意义物体。

图像膨胀是腐蚀操作的逆操作，类似于“扩充”白色区域。它将图像中的高亮区域或白色部分进行扩张，得到比原图更大的高亮区域，以及更粗的线条。进行膨胀操作后，图像中与目标物体相邻的背景会被归并到物体中，使得物体边缘向外延展。因此通过膨胀可以去除物体中小面积的孔洞区域，或者补全物体的边缘凹陷部分。与此同时，膨胀操作还可以用于去噪。直接用腐蚀的方式去除噪声可能会压缩图像，但如果对腐蚀过的图像进行膨胀处理，则不仅可以去除噪声，还能使物体保持原有形状。

膨胀操作和腐蚀操作一样，都接收二值图像以及卷积核作为输入，其中卷积核决定了邻域被蚕食或扩充的程度。二者的不同点在于，当腐蚀卷积核覆盖的区域并非全为 1 时，当前卷积核被扫描到的中心点像素值就被置为 0，而膨胀卷积核只需要区域内包含 1 即可置中心像素值为 1。

2.3.4 图像金字塔

图像金字塔是图像多尺度表达的一种，是以多分辨率来解释图像的有效但概念简单的结构。最初被用于机器视觉和图像压缩，现在多用于图像分割。金字塔底部是待处理图像的高分辨率表示，顶部是低分辨率的近似。一幅图像的金字塔是一系列以金字塔形状排列的，分辨率逐步降低，且来源于同一张原始图的图像集合。图像金字塔的层级由下向上递增，层级越高图像就越小，对应的分辨率就越低、视觉效果越模糊

一般来说，有两种类型的金字塔：高斯金字塔和拉普拉斯金字塔。前者用于向下采样，使图像变小；后者用于从底层图像重建上层未采样图像，即上采样，使图像变大。这里的上采样和下采样与金字塔的方向相反，是图像尺寸的操作，上采样即图像尺寸加倍，下采样即尺寸减半。

高斯金字塔包含了一系列低通滤波器，其截至频率在层级之间以因子 2 递增。具体来说，使用高斯金字塔进行下采样时，首先需要进行高斯滤波，用高斯核对源图像进行卷积，然后去除所有的偶数行和偶数列，最终得到的结果图像尺寸为原图的 1/4。相反，进行上采样时就将图像每个方向扩大到原来的两倍，新增的行和列用 0 填充，然后使用核对放大后的图像进行卷积，得到新增像素的近似值。但这样得到的图像由于信息丢失的原因，视觉上会比较模糊。而拉普拉斯金字塔会记录使用高斯金字塔下采样后再进行上采样得到的图片，与下采样前原图的差异，因此拉普拉斯金字塔也可以认为是残差金字塔。

2.4 深度学习方法

相较于使用基础变换与图像处理的方式来扩增图像，利用深度学习的方式进行图像数据扩增，不仅可以人为干预扩增方向，使结果可控；还能够实现图像特征的自动学习和精准复现，令生成图片与真实图像更加接近。除此之外，在抠图和融合等方面利用深度神经网络，可以得到比粗糙的数据图像处理更加精细的结果，进一步保证了扩增图片的真实性。

对应到本研究中，我们使用深度学习完成对飞虫前景的精细化提取，还将真实拍摄的水果病斑图像作为训练集，通过生成对抗网络，借助图像卷积与风格迁移的思想构建自动生成模型。对于抠图网络的详细实现，将在第三章中详细说明，这里我们对生成对抗网络的原理以及风格迁移思想进行简要概述。

2.4.1 生成式对抗网络

有监督的机器学习算法通常需要大量数据进行训练，而数量有限的训练数据可能会抑制机器学习算法的性能，导致过拟合。因此现在的许多研究，都致力于从现有数据中提取尽可能多的信息。生成式对抗网络（GANs）就是解决数据量受限的方法之一，它通过生成与真实图像相似的合成样本，扩充了原本的数据集。

GANs 的原理来自零和博弈，其构建了一组生成器-鉴别器的对抗网络，来学习生成具有与给定训练集分布相同的合成样本。生成器不断使生成的图像在视觉上与真实图像趋于相似，以至于鉴别器无法将其区分出来；而鉴别器利用生成图像进行对抗训练，不断提高其识别能力。两个对抗网络在训练中保持平衡和同步，最终达到纳什均衡 [36]。

GANs 的优势在于其无需预先建模，直接对真实样本进行采样训练，就可以达到理论上完全逼近真实样本的效果。因此在水果病斑的图像扩增中，我们可以简单地通过改变训练数据集的方式，干预模型的生成结果，得到不同病理时期、不同分布情况下的水果病斑图片。虽然极大极小策略和纳什均衡表明 GANs 可以达到最优，但需要使用梯度下降法训练网络时，连续高维参数构成凸函数才能保证，所以模型收敛并不稳定 [37]。因此如果训练图片的尺度过大、像素过高，则简单的 GANs 模型就会过分自由不可控。因此本研究在第三章中，引入深度卷积网络构成生成器与鉴别器，增加了网络的复杂性，以便生成分辨率更高的图片。

2.4.2 风格迁移

卷积神经网络通过分离和重组图像内容和风格可以创造新的艺术图像，因此使用 CNNs 以不同风格渲染内容图像的过程被称为神经风格迁移 (NST)。风格迁移的原理，是从相应的风格图像中建模和提取风格信息，并转移到内容图像中。用数学表达的方式可以阐述为，接受两个图像 x_S 和 x_C 作为输入，并将 x_S 的样式应用于 x_C 的内容，产生一个新的图像 x' 。从广义的层面来看，风格迁移可以认为是目标提取和纹理迁移，即从源图像提取和转移纹理到目标图像，迁移过程需要维护风格和内容的 consistency。

当前的 NST 方法主要分为两类 [38]，基于图像优化的方法和基于模型优化的方法。第一类方法首先从相应的源图像中提取风格和 content 信息，将它们重组为目标表示，然后迭代地重建与目标表示匹配的风格化结果。一般来说，不同的基于图像优化的 NST 算法共享相同的图像重建技术，但它们对视觉风格建模的方式不同。基于图像优化的 NST 算法的常见限制在于，由于需要迭代图像优化过程，因此计算成本很高。而第二类基于模型优化的 NST 通过利用基于模型优化的图像重建技术来重建风格化的结果，解决了速度和计算成本的问题。即针对一个或多个风格图像，前馈网络在源图像集上进行了优化，通过单次向前传递生成风格化结果。

在基于模型优化解决风格迁移任务的具体实现上，一种比较常见的做法是以有监督的方式训练前馈卷积神经网络，利用损失函数衡量输出图像与真实图像之间的差异，以期不断迭代而达到一致。

而在损失函数的设计方面，常用的有 per-pixel 损失来计算像素之间的差异 [39]；基于从预处理卷积神经网络中提取的高级图像特征表示之间的差异计算感知损失 [40]；基于图像特征之间的相似性来度量图像之间的相似性，而忽略特征空间位置的上下文损失 [41] 等。本研究在此基础上，选用循环一致性损失 [42] 作为风格迁移思想的应用，并将其与生成对抗网络结合。在不成对图像的处理中，循环一致损失存在优越性。根据实验结果，在初始数据集不足的情况下该方法也能得到比较真实的图像。具体方法的阐述和实验结论详见论文的第三、四章。

2.5 本章小结

本章主要是对研究中涉及的理论和方法进行综述。首先介绍了通用的扩增方法，分为坐标变换、颜色空间转换和像素值变换分点阐述。这是各领域进行图像数据扩增最常用的手段，虽然可能会有改变源图像标签的风险，但其在增强受训网络的精确性和鲁棒性方面，依旧有着不可或缺的地位。因此同样作为正

样本扩增手段，存在于本研究的扩增方法里。其次介绍病斑的特征与分析方法，这里对应于负样本扩增中病斑特征模拟的扩增手段。总结了水果炭疽病在不同时期和不同分布下的性状表征，再对常见的颜色、形状和纹理提取方法进行了详细说明。然后介绍研究中使用到的数字图像处理方法，诸如滤波、边缘检测和边界提取、腐蚀膨胀等，这些处理手段将被应用于后续的病斑分割、融合和模拟。最后介绍深度学习方法，着重描述了对抗生成网络和风格迁移思想，对应于负样本扩增中的自动生成技术，是将深度学习应用于农业领域水果图像扩增的理论基石。

第三章 基于栽育端和性状表征的图像数据扩增方法

3.1 图像采集场景及特征分析

水果病虫害监测的智能化建设，可以增强终端设备对病虫害的预警能力，提高控决策的科学性和有效性，及时控制病虫害对于水果的危害，降低果农损失、增加产收。因此本研究的应用场景，就在于为病虫害监测系统提供更多更有效的数据支持。

建设完善的病虫害实时监控系统，需要依托现代农业的基础设施，完成对气象信息、虫情信息、病菌孢子的图像与数据采集。然后通过图像信息库及技术分析功能，精准把握当前监测水果的健康状态、分析病虫害感染情况，制定针对性的培育方案和防治措施。

而栽育端通过图像传感设备采集图片，难免会受到自然环境下各种不可抗性和偶发因素的影响，例如特殊的天气情况导致镜头模糊，如图3.1(a)；或者昆虫干扰影响判断，如图3.1(b)。前者降低了传感器所摄图像的清晰程度，后者可能会使得终端设备错将昆虫判别为病斑，从而影响后续的推理决策。



(a) 天气影响



(b) 昆虫影响

图 3.1: 种植端图像采集的干扰因素

考虑到图像采集场景的复杂性和多样性，本研究通过模拟栽育端的特征来扩增水果图像，以加噪声的正样本扩增为切入点，从天气因素和昆虫干扰两方面入手，旨在提升病害监测告警系统的鲁棒性和抗干扰能力。此外，鉴于基础扩增的简易性和有效性，我们同时还使用了不对标签进行改变的通用扩增技术来获取更多的对抗性正样本。

农业领域大规模图像数据集缺失、积累不足且质量低的问题，一直是掣肘智慧农业发展的重要制约因素，水果图像集更是存在样本严重不均的情况。病

害水果的图像占比少，使得智能监测设备的精确性降低。考虑栽育端的各种外部影响，主要是用来协助扩增具有迷惑性的正样本，帮助提高受训练网络的鲁棒性。而对于水果病斑图像的负样本扩增，则能提升监测网络的精度，缓解样本失衡的问题。因此负样本扩增技术，将作为本研究工作的重点内容。

常见的水果病斑通常具有周期性和传染性，在性状表征方面具体表现为形态特征和分布特征。本研究以水果最容易感染的疾病之一——炭疽病为例，研究了其在感染初期和感染后期，并发侵染以及轮斑软腐的情况。

通过对不同患病周期的水果图像进行扩增，帮助监测系统鉴别当前水果染病的严重程度，从而制定相应的对策。此外，水果炭疽病并不仅在果实上存在表现。如叶片上若出现炭疽病病斑，则昭示了该植株很大程度上已经被病菌侵袭。因此本研究提出的负样本扩增技术，还会根据水果炭疽病不同部位的症状，在叶片和果实图像上都进行相应的扩增，帮助系统尽早判断传染风险并及时反馈，以完成农业告警。

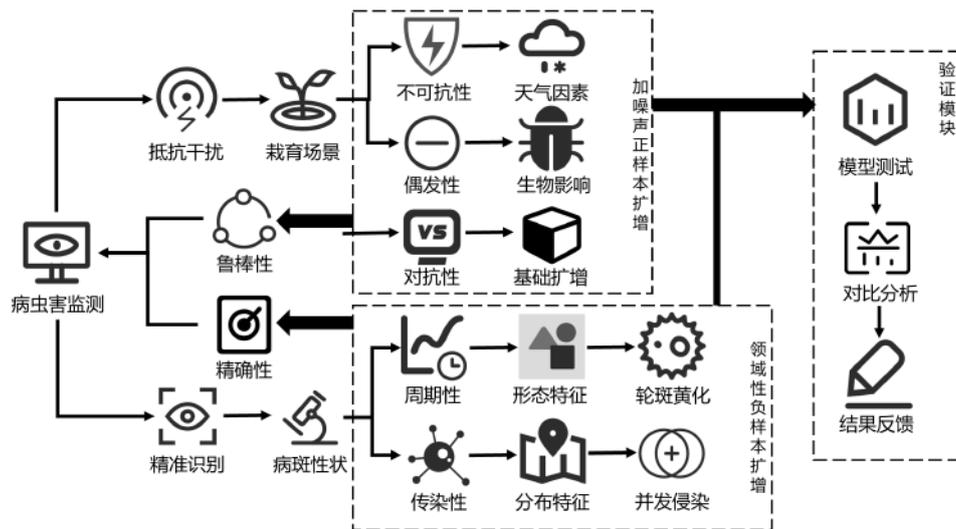


图 3.2: 技术路线总览

综上所述，本研究对于水果图像的数据扩增方法，主要分为基于栽育端特点的正样本扩增以及基于炭疽病性状表征的负样本扩增，技术路线如图3.2所示。研究目的是在符合种植端场景和水果病斑生物特性的同时，生成更多真实且高质量的图像数据，借此提升种植端水果病虫害监测系统的能力。本章中，将根据总览的技术路线详细阐述实现细节。

3.2 正样本扩增技术

3.2.1 不可抗性天气因素

对于水果图像扩增来说，生长过程中遇到的病害主要发生在栽育端。由于暴露在自然条件下，对于图像传感设备的挑战就更加严峻。因此本研究考虑对原图增加雨、雪、雾等影响因素的干扰，以模拟自然环境下可能遇到的各种天气情况。

1、雾场景扩增

模拟现实生活中雾景下拍摄的效果，需要调整原图的模糊程度，以制造白色团雾的景象。雾化场景可以通过调整图像中的每个像素点的值实现，具体操作步骤如下：

- (1) 首先将图像归一化，根据图像尺寸中较长边的开方值确定雾化范围，将图像中心定为雾化中心。
- (2) 计算每个像素点距离雾化中心的距离，乘以设定参数 0.04 后加上雾化尺寸以确定每个点的模糊程度 d 。
- (3) 利用指数公式 $t_d = e^{-\beta d}$ 将每个像素点的模糊值归一化到 0-1 的区间内，其中 β 值表示雾的浓度。
- (4) 最后调整每个像素点处的值： $img_{ij} = img_{ij} * t_d + A * (1 - t_d)$ ，自定义参数 A 决定最后图像的亮度。

设定不同的雾浓度 (β) 和图像亮度 (A) 参数值，得到的雾场景扩增图像如3.3所示，算法伪代码见 Algorithm1。



(a) $A=0.8, \beta=0.03$



(b) $A=0.7, \beta=0.05$

图 3.3: 雾场景扩增

2、雨场景扩增

Algorithm 1: 雾场景扩增算法

```

1 Input: Image to be processed: img; Lightness of result: A; Thickness of fog:
   beta;
2 Output: Image with fog;
3 rows, cols, chs  $\leftarrow$  Shape of img;
4 size  $\leftarrow$  Sqrt(Max(rows, cols))*1.15;
5 center  $\leftarrow$  (rows/2, cols/2);
6 foreach i in rows do
7   foreach j in cols do
8     d  $\leftarrow$  -0.04*distance(center,(i,j))+size;
9     td  $\leftarrow$  exp(-beta*d);
10    result(i, j)  $\leftarrow$  img(i, j)*td+A*(1-td);
11 return result;

```

而雨天场景下，镜头常常会捕捉到一些细密的雨丝，利用图像加权的方式，将生成的雨滴噪声和原始图像叠加即可得到模拟的下雨场景。此外修改雨滴噪声图中，模糊线条的长度和密度可以更改雨势的大小。由于背景是黑色，所以调整后的图像亮度会下降，更适合模拟雨天时因为乌云导致的光线昏暗。

- (1) 首先利用 0-255 之间的均匀随机数生成不同密度的噪声图像，来模拟不同大小的雨量。
- (2) 通过阈值来控制噪声水平，只保留最大的部分作为噪声。将噪声图像里小于 $256 - value * \alpha$ 的像素值全部置为 0。因为生成的噪声是浮点数，所以在 *value* 上乘了尺度缩小因子 α （本研究中设为 0.01）。
- (3) 将噪声加上运动模糊，以模仿雨滴。使用对角阵自带的 45 度倾斜，初始化雨滴噪声的倾斜角，生成模糊核。高斯模糊上述旋转后的对角核，使得雨有宽度。
- (4) 利用旋转后的核进行滤波，得到二维 *rain* 噪声。最后将其扩张为与原图相同的三通道图像，加权合成新图。

假设原图为 *img*，雨滴噪声图为 *noise*，则得到的雨天图像结果可以表示为： $rain_img = \alpha * img + (1 - \alpha * noise)$ ， α 的值决定了原图的比重，一般设为 0.9。算法伪代码见 Algorithm2，图3.4展示了雨天场景下的扩增图片。

Algorithm 2: 雨场景扩增算法

```

1 Input: Image to be processed: img; Level of noise: value; Custom
   convolutional kernel: kernel; Length, inclination angle and size of
   raindrops: length, angle, w; Scale factor of original image: alpha;
2 Output: Image with rain;
3 noise ← Random(range=(0,256), shape=img.shape);
4 v ← value*0.01;
5 noise ← Set pixel value as 0 where that is less than (256 - v);
6 noise ← Convolute the image of noise with kernel;
7 trans ← getRotateMatrix2D((length/2, length/2), angle-45, 1-length/100.0);
8 diag ← Generate diagonal matrix with length;
9 k ← warpAffine(dig, trans, (length, length));
10 k ← GaussianBlur(k, (w, w));
11 blurred ← Use k to filter noise;
12 rain ← Expand dimation of blurred to three channels as img;
13 result ← addWeighted(img, alpha, rain);
14 return result;

```

3、雪场景扩增

模拟下雪时的拍摄效果，实际是用白色像素点替代原图中部分像素，以复现雪花遮挡镜头的情境。具体实现与雨场景的雨滴噪声生成过程相似，区别在于雪花噪声是利用白色像素替代原图中的像素，而非叠加，因此这里也不再赘述下雪场景的扩增伪代码。

实际应用时，我们可以通过调整参数，修改雪花层的密度，即低分辨率空间中每个像素成为雪花的概率。此外还可以通过设置雪花运动模糊的角度以及下落速度，在图像中“拉伸”每个雪花，以达到不同雪势大小下拍摄图片的效果。

因为积雪场景下拍摄的图片会受到阳光反射，形成图像过曝、区域亮度激增的特点。所以积雪堆积的图片可以通过调整部分区域的亮度得到。具体实现时，首先利用公式3.1根据 RGB 值计算亮度，

$$L = 0.299 \times R + 0.587 \times G + 0.114 \times B \quad (3.1)$$

将其转换到 HSL 色彩空间后，设定亮度阈值 *lightness*，亮度低于此值的所有像素的亮度将乘以 *lightness_multiplier*（作为参数设定）。其算法伪代码见 Algorithm3，下雪场景和积雪场景的扩增图片如图3.5。



图 3.4: 雨场景扩增



(a) 下雪场景



(b) 积雪场景

图 3.5: 雪场景扩增

3.2.2 偶发性生物影响

由于传感器在采集图像时可能会意外拍摄到落在水果上的昆虫，反馈回监测设备后存在将其错误识别为水果病斑的可能性。因此为了应对这种偶发性的生物影响，在一定程度上提升智能终端的鲁棒性，本研究考虑在正样本扩增方面为图像增加昆虫影响，得到带干扰因素的正样本。

只有当昆虫前景遮挡原本健康的果实或者叶片时，所得到的图像才可能使病害监测系统产生误判。但这种偶然情况下拍摄到的图像数量比较少，因此考虑通过扩增手段来还原。在实现过程中，我们首先使用形态学分割或自动化抠图的方式来提取昆虫前景，以应对复杂程度不同的源图像，同时确保满足高效性或精细化方面的需求。随后借助泊松融合，将提取得到的昆虫前景融合至正常水果图像的特定区域。

1、形态学分割

利用形态学分割获取昆虫前景的方式，转化更加快速，图像无需经过模型处理即可完成分割。但缺点在于只适用于背景简单的图片，如果图像中有其他轮廓清晰的物体，可能会被一并分割。实现的具体处理步骤如下，算法伪代码见 Algorithm4:

Algorithm 3: 积雪场景扩增算法

```

1 Input: Image to be processed: img; Threshold of lightness: lightness;
   Multiplier of lightness: inc;
2 Output: Image with SnowyLandscape;
3 snow_noise  $\leftarrow$  img.copy();
4 L  $\leftarrow$  Use value of RGB to calculate lightness;
5 foreach i in len(L) do
6   foreach j in len(L(i)) do
7     if L(i,j)  $\leq$  lightness then
8       snow_noise(i, j)  $\leftarrow$  snow_noise(i, j) + L(i, j) * (inc - 1);
9 snow_noise  $\leftarrow$  Set pixel value as 255 if that is more than 255;
10 result  $\leftarrow$  Round snow_noise and change its type to uint8;
11 return result;

```

- (1) 首先将图片转化为灰度图，利用 Canny 算子进行边缘检测，得到边界为白色、背景为黑色的最佳轮廓图。
- (2) 定义一个 3×3 的卷积核，对轮廓图进行膨胀。即遍历这张图片中的每一个像素，当遇到像素点为白色（255）时，将其周围卷积核大小的像素点全部置为白色。
- (3) 继续利用 3×3 的卷积核膨胀上述操作后得到的图像。因为昆虫区域一般不会占比太大，所以此时得到的 *mask* 已经基本能将需要分割的部分置为白色。
- (4) 最后通过与原图进行掩膜位运算，提取 *mask* 白色区域的内容，即可得到昆虫前景图。

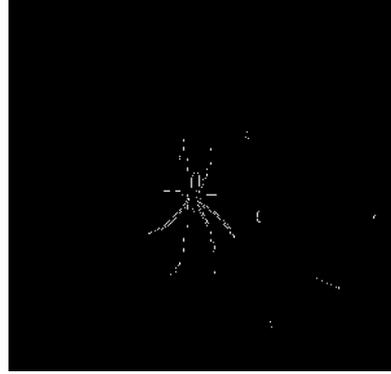
这里使用边缘检测算法中最为常用的 Canny 算子来完成简单背景图像下昆虫前景的边界提取，这也是现阶段效果最为优秀的边缘检测算子之一。

本文在第二章中提到，图像的梯度包含了边缘和细节信息，而物体边缘处常具有梯度的极大值。因此一阶微分边缘算子（梯度边缘算子）就是利用梯度阶跃的原理，来进行边缘检测。

梯度是一个矢量，它具有方向 θ 和模 ΔI 。梯度的方向提供了边缘的趋势信息，因为梯度方向始终垂直于边缘方向，而梯度的模值大小提供了边缘的强度



(a) 原图



(b) Canny 边缘检测图



(c) 膨胀轮廓边缘



(d) 昆虫前景提取

图 3.6: 形态学分割法获取昆虫前景

信息，二者的计算公式如下。

$$\theta = \arctan\left(\frac{I_y}{I_x}\right) \quad (3.2)$$

$$\Delta I = \left(\frac{\partial I}{\partial x} \frac{\partial I}{\partial y}\right) \quad (3.3)$$

$$|\Delta I| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} = \sqrt{I_x^2 + I_y^2} \quad (3.4)$$

在实际使用中，通常利用有限差分进行梯度近似，因此对于上面的公式，有如下的近似：

$$\frac{\partial I}{\partial x} = \lim_{h \rightarrow 0} \frac{I(x + \Delta x, y) - I(x, y)}{\Delta x} \approx I(x + 1, y) - I(x, y), (\Delta x = 1) \quad (3.5)$$

$$\frac{\partial I}{\partial y} = \lim_{h \rightarrow 0} \frac{I(x, y + \Delta y) - I(x, y)}{\Delta y} \approx I(x, y + 1) - I(x, y), (\Delta y = 1) \quad (3.6)$$

而 Canny 边缘检测在一阶微分算子的基础上，额外增加了非极大值抑制和双阈值检测这两项改进，使得检测效果和精度有了明显地提升。前者有效地抑制了多响应边缘的问题，提高了定位精度，后者利用双阈值来显著减少边缘的漏检率。Canny 算法的具体实现分为四个步骤完成：

第一步通过高斯滤波，根据待滤波的像素点及其邻域点的灰度值按照高斯公式生成的参数规则进行加权平均，这样可以有效平滑图像并过滤原图中叠加的高频噪声。实际上我们可以把高斯滤波器看做金字塔结构，滤波器值视为权重，故距离当前像素点越远权重越小，对灰度值的贡献也越小。即使当前像素点为噪声点，高斯滤波器也会通过周围点的灰度值来制约噪声的影响。

第二步计算图像中每个像素点的梯度大小和方向，然后通过公式3.7计算角度图像，来为非极大值抑制提供方向上的指导。因为上一步中得到的梯度图像存在边缘粗宽、弱边缘干扰等众多问题，因此需要通过非极大值抑制的方法来寻找像素点局部最大值。将非极大值所对应的灰度值置 0，这样就能剔除一大部分非边缘的像素点，消除边缘检测带来的杂散响应。

$$\phi(x, y) = \arctan \left| \frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right| \quad (3.7)$$

第三步是应用双阈值检测来确定真实的和潜在的边缘。由于噪声的影响，经常会在本应该连续的边缘出现断裂的问题。为了消除伪边缘，故采用滞后阈值化的方法，设定两个阈值：一个为高阈值 T_h ，一个为低阈值 T_l 。将小于低阈值的点置 0，认为其是假边缘，将大于高阈值的点置 1，并视其为强边缘，介于中间的像素点则进行进一步地检查。

第四步通过抑制孤立的弱边缘完成最终的边缘检测。将高阈值图像中得到的边缘连接成轮廓，当到达轮廓的断点时，该算法会在断点的 8 邻域中寻找满足低阈值的点。再根据此点收集新的边缘，直到整个图像闭合。如图3.6(b)就是利用 Canny 算子对原图进行边缘检测的结果。

在得到 Canny 算子检测出的昆虫边缘图像后，本研究进一步使用卷积核扩张边界区域，填充昆虫内部的像素点。但此时得到的 mask 还不能覆盖全部的昆虫前景，中心依旧存在“空洞”。因此继续使用图像膨胀，令 mask 边缘向内收缩，获得完整的昆虫前景遮罩。图3.6(c)就是对边缘检测的结果使用两次 3×3 的卷积核膨胀后的图像。最终将 mask 与原图相与，即可得到快速分割的昆虫前景。

这种基于形态学图像处理进行前景提取的方式，实现起来简单快速，对于背景并不复杂的图像可以得到良好的分割效果。但大多数情况下，昆虫图像中可能存在其他轮廓清晰的物体。而且由于自然选择，生物存在拟态和保护色，因此自然场景下拍摄的昆虫图像常与背景有相似的颜色或纹理，使得图像边界难

以区分。简单的形态学分割无法应对这种边缘不确定的情况，故本研究考虑图像抠图算法来实现复杂背景下昆虫前景的提取。

Algorithm 4: 形态学分割法获取昆虫前景

```

1 Input: Path of image to be processed;
2 Output: Insect foreground;
3  $image \leftarrow cv2.read(\text{path of image});$ 
4  $gray\_img \leftarrow \text{Convert RGB } image \text{ to Gray domain};$ 
5  $canny\_img \leftarrow \text{Use canny algorithm to get best edge of } gray\_img;$ 
6  $length, width \leftarrow \text{shape of } canny\_img;$ 
7  $kernel\_size \leftarrow 3;$ 
8 foreach  $row$  in  $length - 1$  do
9   foreach  $col$  in  $width - 1$  do
10     if  $canny\_img(row, col)$  is 255 then
11       Traverse  $canny\_img$  with convolution kernel of
12          $kernel\_size \times kernel\_size;$ 
13        $bp \leftarrow \text{Positions that need to be set as white around the pixel point}$ 
14          $[row, col];$ 
15 foreach  $pos$  in  $bp$  do
16    $canny\_mask \leftarrow \text{Set } pos \text{ in } canny\_img \text{ to be white};$ 
17  $dilate\_mask \leftarrow \text{Dilate } canny\_mask \text{ with convolution kernel of } 6 \times 6;$ 
18  $mask \leftarrow \text{Use bitwise\_and Operator to extract the white area};$ 
19 return  $mask;$ 

```

2、自动化抠图

另一种获取昆虫前景的方式是基于全卷积网络，构建对昆虫进行自动抠图的深度学习模型。这种方法虽然有前期训练成本，但扣图边缘细节保留完善，能够较为精准地将昆虫从复杂背景中分割出来。

图像抠图的目标是估计前景对象的不透明度，通过计算每一个像素点的透明度值来分离前景和背景，从而实现前景提取的目标。传统的抠图技术因为仅使用低级特征且缺乏高级上下文，所以在处理具有相似的前景和背景颜色，或复杂纹理的图像时，算法性能很差。因此本研究在这里，使用基于深度学习的 Image Matting 方法，来准确提取图像中的昆虫前景，以便完成后续的图像融合。

在具体实现时，图像抠图的目标可以表述为，通过估计像素的不透明度值来创建 α 遮罩。假设原图像 I 是通过公式3.8，从三个底层图像——前景图像 F 、背景图像 B 和 α 遮罩 (mask) 生成的：

$$I_p = \alpha_p F_p + (1 - \alpha_p B_p) \quad (3.8)$$

其中 p 表示图像中的某一像素，数值 $\alpha_p \in [0, 1]$ 。而自然图像一般是 RGB 图，因此存在三个通道，在每个通道中该公式都成立。由于像素 p 处的 RGB 颜色 I_p 是已知的，而前景颜色、背景颜色和遮罩估计 α 都是未知的，那么抠图问题中每个像素将被 7 个未知值约束，但只有 3 个已知值。故估计图像的透明度值是一个严重约束不足的问题，这种约束不足的问题可以通过添加更多信息的方式来解决，如 trimap (包含确定性的前景和背景以及不确定区域) 或背景图像等，即标记一些确定为前景或背景的像素。

传统方法通过采样相邻像素来估计 α 值或定义 α 传播的相似性度量，而基于深度学习的抠图方法通过从大量标记数据中学习有区别的表示，直接预测 α 遮罩来解决该问题。因为产生 trimap 等辅助输入，需要具有一定专业知识的人为干预，不符合自动化扩增的要求。因此本研究进一步选用自动图像抠图的方式，不依赖 trimap 作为输入，以减少人工标注的代价、消除专业知识的壁垒。最终得到的模型对具有复杂背景的昆虫图像可以产生精细化的抠图结果，具体如图3.7所示。



(a) 原图



(b) 抠图结果

图 3.7: 自动抠图法提取复杂背景下的昆虫前景

在网络架构方面，本研究借鉴了端到端的动物图像抠图算法 [43] 和深度自然图像 AIM (Automatic Image Matting) [44] 算法，构建了昆虫图像 AIM 网络，以获取边界精细的 α 遮罩。因为没有 trimap 作为辅助输入，所以图像的过渡区域未知，无法通过可用的前后景信息来区分细节。因此本研究的抠图模型就需要理解图像前景和背景的整体语义划分，同时尽可能避免结果产生伪影。

如图3.8所示，该网络主要由一个编码器和两个解码器组成。编码器 Feature Encoder 从输入图像中提取语义特征，两个解码器共享相同的编码，分别对图像进行粗粒度的语义分割和细粒度的抠图指导，下文中表示为粗粒度解码器 (Coarse-grained Decoder, CD) 和细粒度解码器 (Fine-grained Decoder, FD)。CD 通过学习全局语义特征负责识别粗略的前景和背景，FD 通过学习局部结构特征负责抠图未知区域的细节。

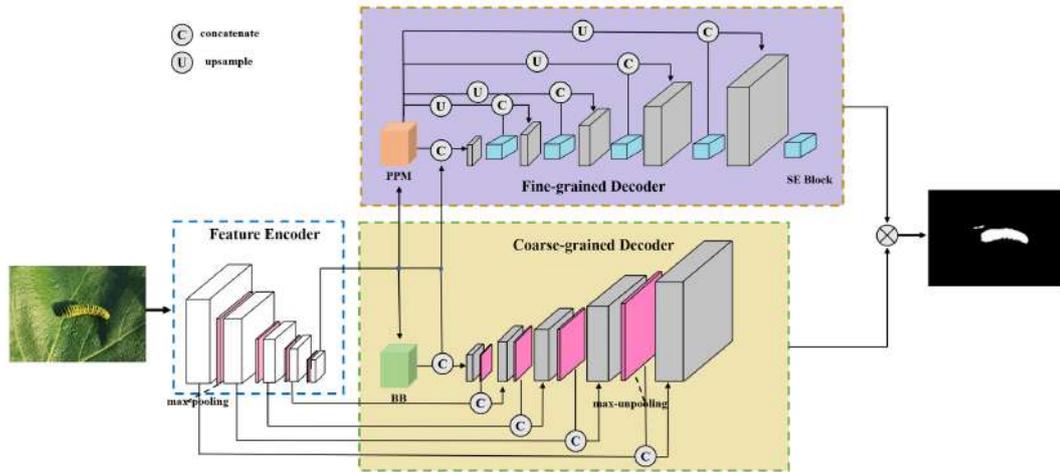


图 3.8: 昆虫自动抠图网络架构图

本研究使用在 ImageNet 上预训练的 ResNet-34[45] 网络作为编码器，接收单幅图像作为输入，通过 $E_0 - E_4$ 五个模块进行处理。为了保留局部细节的特征，在每一个处理模块中，将原模型 conv1 的步长从 2 修改到 1，并添加步长为 2 的最大池化层，使每个模块仅将分辨率减半（原始网络为了显著减少计算量并增加感受野，其输出是输入的 $1/4$ ）。要注意这里将保留最大池化层的索引，应用于 Fine-grained Decoder 中的 max-unpooling 层。

粗粒度解码器使用一个较大的感受域来识别图像中简单的语义部分，对应于编码器使用了对称堆叠的 5 个解码器块 $D_4^C - D_0^C$ 。每个解码器块都有 3 个连续的 3×3 卷积层和一个上采样层，同时每个解码器块后添加了 Squeeze-and-Excitation (SE) [46] 注意模块，从而选择信息最丰富的特征，并过滤掉不太有用的特征，来进一步增加解码器学习区别语义特征的能力。此外，为了实现较大的感受野，选择新增金字塔汇集模块 (PPM) [47] 来接收编码器的输出，提取全局上下文。输出得到上采样的 PPM 特征，与各个解码器模块里的 SE 模块输出相连，用作下一个解码器块的输入。使用交叉熵损失 L_{CE} 来监督此解码器的训

练，损失函数定义见公式3.9。

$$L_{CE} = - \sum_{i=1}^3 C_g^i \log(C_p^i) \quad (3.9)$$

其中， $C_p^i \in [0, 1]$ 是 i th 的预测概率， C_g^i 是真实标签。为了使粗粒度解码器可以学习到有用的语义特征，并将图像划分为前景、背景和过渡区域，还需要提供明确的语义监控信号。由于昆虫图像属于具有突出前景和不透明内部的图像，因此利用传统的 `trimap` 作为语义表示（这里的 `trimap`，是通过对真实的 α 遮罩进行腐蚀和膨胀而得到的）。最终 CD 的输出 C_p 是一个前景、背景和不确定区域的三通道类概率图。

细粒度解码器的基本结构与 CD 相同，但不同点在于其目的是提取过渡区域中的细节，即底层结构特征。因此使用 `Bridge Block` 代替 CD 中的 `PPM` 模块，来获取不同感受野中的局部上下文信息。同时遵循 `Unet`[48] 结构，将每个编码器模块 E_i 的输出与相应的解码器模块 D_i^f 相连，并通过具有反向索引的 `max-unpooling` 层，来恢复精细的结构细节。FD 的训练损失由 α 预测损失 L_α^T 和拉普拉斯损失 L_{lap}^T 组成：其中， α 损失是未知的过渡区域中，真实的 α 遮罩与预测的 α 遮罩之间的绝对差值；而拉普拉斯损失 L_{lap}^T 被定义为真实的拉普拉斯金字塔和预测的拉普拉斯金字塔之间的 L_1 距离。具体公式见3.10-3.12：

$$L_{FD} = L_\alpha^T + L_{lap}^T \quad (3.10)$$

$$L_\alpha^T = \frac{\sum_i \sqrt{((\alpha_i - \alpha_i^F) \times W_i^T)^2 + \epsilon^2}}{\sum_i W_i^T} \quad (3.11)$$

$$L_{lap}^T = \sum_i W_i^T \sum_{k=1}^5 \|Lap^k(\alpha_i) - Lap^k(\alpha_i^F)\|_1 \quad (3.12)$$

其中 i 表示像素索引， $W_i^T \in \{0, 1\}$ 表示像素 i 是否属于过渡区域。为了计算的稳定性，增加 $\epsilon = 10^{-6}$ 。此外，使用 5 个层次的拉普拉斯金字塔， Lap^k 表示金字塔的第 k 层。

最后用 FD 的预测结果代替 CD 预测的过渡区域，来生成最终预测的 α 遮罩。两个解码器被同时训练以实现交互，同时损失也可以及时传递。整合预测结果的训练损失 L_{final} ，由 α 预测损失 L_α 、拉普拉斯损失 L_{lap} 和分割损失 L_{comp} 组成：

$$L_{final} = L_\alpha + L_{lap} + L_{comp} \quad (3.13)$$

这里的 α 损失和拉普拉斯损失是对于整个 α 遮罩，根据公式3.11和3.12计算得到的。而分割损失通过计算基于真实 α 遮罩的分割图像与基于预测 α 遮罩的分割图像之间的绝对差得到，公式如3.14。

$$L_{comp} = \frac{\sum_i \sqrt{(C(\alpha_i) - C(\alpha_i^{final}))^2 + \varepsilon^2}}{N} \quad (3.14)$$

其中 $C(\cdot)$ 表示利用 mask 从原图像中提取前景得到分割图像， α^{final} 是最终预测的 α 遮罩， N 表示 α 遮罩中的像素数量。最终训练整个网络的损失函数为：

$$L = L_{sum} + L_{CE} + L_{FD} \quad (3.15)$$

3、泊松融合

在得到边界清晰的昆虫前景图像后，还需要将其融入正常的水果图片中。对于融合区域的确定，本研究使用目标检测技术，相关内容将在后文中详细阐述，这里着重介绍选用的融合方法。

为了保证扩增得到的昆虫影响图像没有违和感，需要扩增结果在前后景相接的边界，不会有生硬的颜色或纹理过渡。粗暴的图像拼贴显然无法满足要求，因此需要图像融合技术来完成。

图像融合方法大致分为两类，基于空间域的融合方法和基于变换域的融合方法。空间域技术通过获取没有变化的图像的像素或块等来产生融合图像，将要融合的图像像素以线性或非线性方式组合。变换域技术合并互补变换系数，并应用逆变换来形成融合图像。通过诸如小波 [49] 或金字塔 [50] 之类的适当变换，将要组合的输入图像从空间域变换到某个其他域。利用合适的融合规则对变换后的图像进行融合，再使用逆变换重构原始图像。

在当前已经得到较为精细的昆虫前景的基础上，不必使用耗时且复杂的融合方法，只需前景边界可以自然过渡到水果图像背景中，即可获得我们的扩增结果。考虑到扩增的效率和质量，此处采用基于 Poisson 方程的融合方法，求解像素的最优值。其选择融合区域的过程简单方便，且最终得到的结果接近无缝融合，生成的昆虫影响下的扩增图像详见图3.9。

纹理特征存在明显差异的前后景图片（昆虫图像为前景，水果图像为后景），直接拼贴后会有清晰的边界。为了使融合图像更加自然，其过渡边缘要尽量平滑。同时需要提取前景图像中的部分细节，允许改变其颜色（绝对信息），但在融合后应尽可能保留前景图像的梯度（相对信息）。正如文章前面所提到的，图像梯度描述的是像素之间的相对变化，本质上是像素与其相邻像素的差值，可以用于捕获图像上的亮度变化。因此泊松融合利用泊松方程来保持图像融合后



图 3.9: 泊松融合得到昆虫影响扩增结果

的前景区域梯度变化尽可能小，从而达到内容平滑的效果；同时使得边界上的像素值相同来保持前后景的边界一致。

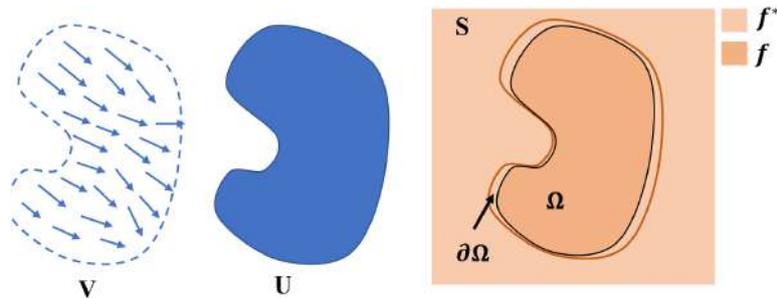


图 3.10: 泊松融合原理

如图3.10所示，假设 U 是需要被融合的前景图片； Ω 是融合后的目标图像中被前景覆盖的区域，即我们主要希望提取的前景部分 roi ； S 是背景图片。若 U 的梯度场为 v ， $\partial\Omega$ 是 Ω 的边界，将融合图像在 Ω 内的像素表示为函数 f ，在 Ω 外的像素表示为函数 f^* ，那么融合目的则可以简单阐述为：

$$\min_f \iint_{\Omega} |\nabla f - v|^2 \quad (3.16)$$

$$f|_{\partial\Omega} = f^*|_{\partial\Omega} \quad (3.17)$$

上述公式表示，尽量使融合后图像 Ω 区域内的像素 f 与前景图像 U 的梯度信息相似，可以最大程度保持原始纹理，融合效果也随之越好。令边界像素等于背景像素，确保了边界一致，也为泊松方程设置了边界条件。

泊松融合的具体实现步骤如下：

- (1) 计算前景图像感兴趣区域 (roi) Ω 的梯度场以及背景图片的梯度场；

- (2) 将背景图像上对应于要被 Ω 区域覆盖的梯度场，替换为 roi 的梯度场；
- (3) 对上述替换后的梯度场求偏导，得到融合图像的散度；
- (4) 根据泊松重建方程构建系数矩阵，求解边界区域的像素值。

已知散度和边界点上的像素值，就可以构建泊松方程组 $Ax = b$ 进行泊松重建了。这里的 b 值为融合图像的散度值；系数 A 实际为一个系数矩阵 $A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ ，矩阵的每一行有五个非零元素，对应于拉普拉斯算法的卷积核；而 x 就是期望得到的融合结果的像素颜色值。

假设一张 4×4 的图像 $A = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ x_5 & x_6 & x_7 & x_8 \\ x_9 & x_{10} & x_{11} & x_{12} \\ x_{13} & x_{14} & x_{15} & x_{16} \end{bmatrix}$ ， x_i 表示各个位置上不同的

像素值，如果应用拉普拉斯卷积核，则只能得到 4 个方程式：

$$\begin{cases} x_2 + x_5 + x_7 + x_{10} - 4x_6 = \text{div}x_6 \\ x_3 + x_6 + x_8 + x_{11} - 4x_7 = \text{div}x_7 \\ x_6 + x_9 + x_{11} + x_{14} - 4x_{10} = \text{div}x_{10} \\ x_7 + x_{10} + x_{12} + x_{15} - 4x_{11} = \text{div}x_{11} \end{cases} \quad (3.18)$$

而泊松融合的要求确定了边界处的实际值，提供了更多等式，满足狄利克雷边界条件。最终矩阵化方程组为 $Ax = b$ 的形式，就能求得融合图像的像素点值。

3.3 负样本扩增技术

基于炭疽病的水果病斑图像扩增，是本研究工作的重点内容。负样本扩增的思路主要从三个方向着手，如图3.11所示：(1) 根据第二章中对水果炭疽病领域性的专业分析，可以发现，炭疽病病斑在不同的生长周期和染病位置，呈现不同的性状表征。但在各类水果上又有着相似的表现，不存在物种间的差异性。因此可以将炭疽病病斑按照感染早期与晚期、病原生发于叶片和果实进行分类，分割得到各个周期内、不同分布下的典型斑块。滤除噪点后组成需要保留的前景病灶，最后使用泊松融合将其插入至正常图片中。(2) 借助第二章里对图像特征分析的方法，从颜色、形状和纹理三个方面总结病斑特征，同时结合领域知识对正常图像进行特征调整，以模拟真实病斑。注意上述两个方法在实现时，需要通过目标检测模型，确定病斑插入或合成的区域。(3) 使用感染炭疽病的

水果图像集训练生成式对抗网络，通过构建的生成器得到病斑图像，这里本研究主要使用 DCGAN 和 CycleGAN 方法来完成。

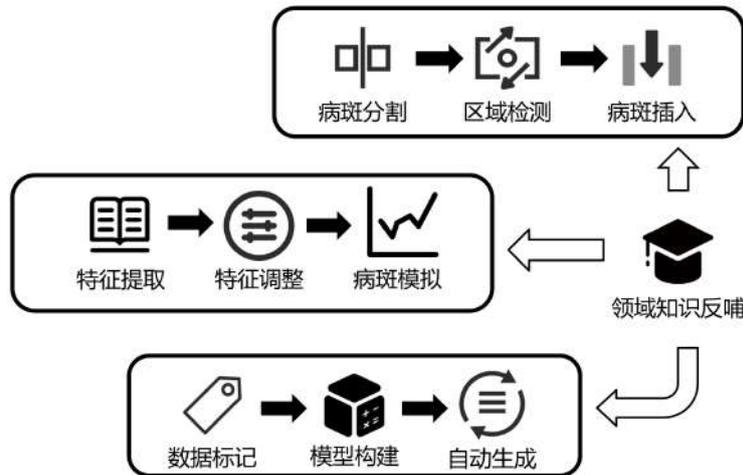


图 3.11: 负样本扩增思路

3.3.1 分割后融合技术

不同于正样本扩增时使用的图像抠图技术，这里进行的病斑分割属于硬分割问题，即对每一个像素点进行分类。

之所以在正样本扩增的昆虫提取时使用复杂的 Image Matting 使边界更加清晰，是因为昆虫图像可能有复杂的背景，或由于生物进化的影响，昆虫本身的颜色和纹理会与图像背景非常相似，导致简单的图像分割难以将其从原图像中剥离出来。而水果病斑常常有明显的边界特征，与正常的果实或叶片存在显著的颜色、纹理差异，因此可以对病斑图像进行粗糙的像素级分割。

1、病斑分割

考虑到这里的分割任务是像素级，可以采用简单的分类模型对图像中的每个像素进行分类，因此本研究主要使用机器学习算法，对比了有监督的决策树和无监督的 kmeans 算法。实验结果显示前者需要人工干预但分割精度更好，后者无需预处理但可能会保留更多的噪点像素。

(1) 决策树

决策树是一种树形结构，其每个内部结点表示一次属性判断，根据每个分支处不同的判断结果走向不同的叶节点，每个叶节点都对应着一种分类结果。

决策树的生成算法主要有 ID3、C4.5 和 Cart 树 (Classification and Regression Tree)，其中 Cart 树的分类效果一般优于其他决策树。

ID3 以信息熵和信息增益为衡量标准，从而实现对数据的归纳分类。这里引入信息论的观点，即信息熵就是系统不确定性（不纯度）的度量。那么决策树上每个节点在分类前后，其系统不确定性应该有所降低。不确定性程度降低得越大，也就意味着该分支对于结果的区分情况越好，而这个不确定性的差值就被定义为信息增益。

由于 ID3 算法是基于奥卡姆剃刀原理，根据信息增益不断选择最优特征进行分割，因此可能会偏向于选择类别较多的属性，从而达到局部最优，而非整体最优。此外 ID3 算法也不支持处理连续值、缺失值和剪枝操作，可能会出现过拟合的情况。C4.5 使用信息增益率缓解 ID3 了对取值较多特征的偏好问题，在处理连续数据和弥补缺失值上进行了改进，使用剪枝预防模型过拟合。

Cart 树使用 Gini 系数作为构建二叉树的指标，使决策树能够处理分类问题，本研究就是利用基于 Cart 树构建的决策树算法实现像素点分类。同样 Cart 树也可以应对回归问题，用的则是节点内目标变量的方差来定义。Gini 系数与信息熵的概念类似，都是用于衡量不纯度和系统不确定性的指标。区别在于信息熵的计算是对数运算，而 Gini 系数是乘法运算，因此在计算效率上更高。

以 Cart 树为例，构建二叉决策树的过程是从根节点开始，根据训练数据集，计算现有特征的 Gini 系数。在所有可能的特征以及其所有可能的切分点中，选择 Gini 系数最小的特征和对应的切分点，作为最优特征和最佳切分点，据此将本节点上的训练数据依特征分配到两个子节点中去。不断递归上述步骤，直至满足条件停止。一般来说算法停止的条件是 Gini 系数低于某个阈值，或样本数小于设定的阈值等，但为了降低决策树的复杂度、避免过拟合的情况，会采取一定策略的预剪枝操作来限制树的生长。

由于决策树是有监督的机器学习方法，因此可以从已知分类结果的样本中学习特征，来对新的数据给出正确分类。在本文的研究中，使用决策树进行病斑分割主要分为以下几个步骤：

- (a) 人为对病斑图像采集 *train_ground* 和 *train_target* 作为训练集，前者是待分割图像中的背景切片，后者是需要提取的病斑部分的切片。
- (b) 分别对 *train_ground*、*train_target* 和原图进行特征提取，得到特征值矩阵。根据第二章中的阐述可知，图像在不同颜色空间中有相异的特征表示，因此在特征提取的过程中，首先将读入的 RGB 图像映射到 HSV、YCrCb 和 HSL 三种颜色空间中。再拼接不同空间下同一图像的 Green 分量、色相、饱和度、强度、亮度等九个特征向量，组成对应图像的特征值矩阵。

- (c) 设定 *train_ground* 对应的分类标签为 0, *train_target* 对应的分类标签为 1, 使用决策树对其进行拟合。
- (d) 最后利用训练完成的决策树, 对整张图片中的每个像素进行分类, 分类为 1 的部分保留, 分类为 0 的像素点置为白色, 得到病灶区域。

(2) kmeans 聚类

聚类是数据挖掘中的概念, 即按照某个特定标准 (如距离) 把一个数据集分割成不同的类或簇。使得同一个簇内的数据对象尽可能相似, 同时不同簇中数据对象的差异性也尽可能地大, 以达到类内凝聚、类间离散的效果。

聚类分析的原理是基于数据之间的相似度, 将相似的数据合并在一起。数学上衡量两个点的相似性可以用距离来表示, 常用的距离包括欧式距离、曼哈顿距离、切比雪夫距离等。

kmeans 算法作为聚类算法的代表, 其中 *k* 表示初始的中心点数, 即预计划分的类目数量, 属于超参数, 需要人为设定。means 表示计算方式, 通过计算每个簇的所有样本的平均值来不断更新迭代, 簇中所有数据的均值 μ_j 通常被称为这个簇的质心 (centroids)。kmeans 的核心任务就是根据设定的聚类数量, 找出 *k* 个最优质心, 将离这些质心最近的数据分别分配到质心代表的簇中。其算法步骤可以总结为:

- (i) 随机初始化 *k* 个样本作为初始的聚类中心;
- (ii) 计算每个样本到聚类中心的距离, 并将其分配到距离最小的聚类中心所对应的类中;
- (iii) 针对每个类别, 计算该簇的均值点作为新的聚类中心;
- (iv) 重复上述的第二、第三步, 直到质心的位置不再发生变化或达到某个终止条件 (迭代次数限制、最小误差变化等)。

kmeans 算法的优点在于算法复杂度低, 执行速度快, 在一般情况下聚类效果不错, 可以达到局部最优。但是缺点在于对 *k* 值的选取异常敏感, 初始簇中心的选定会对结果产生一定影响。

通过对病斑分割的相关研究 [51][52] 进行调查后发现, 主流的方法是使用 kmeans 聚类来对病斑图像进行识别和分割。且反复实验后发现, 当 *k* 值取 3 或 4 时, 能够得到较好的分割结果。最终计算距离时, 通过最小化数据对象和相应聚类之间距离的平方和来执行分类。具体实现图像分割的步骤如下:

- (a) 读入病斑图像后，将其从 RGB 空间转换至 $L^*a^*b^*$ 颜色空间；
- (b) 在 a^*b^* 空间中使用 kmeans 聚类对颜色进行分类，k 值设定为 3 或 4；
- (c) 从 kmeans 的分类结果中标记图像中的每个像素，生成按颜色分割后的图像；
- (d) 选择包含病斑的细分区域作为病灶图像。

通过图3.12比较二者的分割结果发现，决策树算法需要前期人为选择用于监督训练的目标区域和背景区域，无法实现自动化，但分割精度更好，分割效果人为可控。而 kmeans 聚类算法无需预处理，就能对病斑图像进行自动分割，但可能会保留更多的噪点像素，且分割效果会随着输入图片的不同而产生较大差异。

此外，由于决策树分割会造成病灶区域的像素点丢失，而 kmeans 分割会引入许多无关像素。因此在制作病灶集时，对比使用这两种方式得到的分割结果，择其优作为预生成病灶。同时借助滤波、裁剪等图像处理方式，进行进一步的补像素和去噪声处理，来得到较为干净的病斑斑块。

2、病灶插入

通过上述的图像分割技术，我们得到了水果果实和叶片在感染炭疽病不同时期内的病斑图像，进一步整合分类后组成了病斑图集，以供后续的病灶插入。图3.13展示了分割得到的炭疽病典型病灶集，根据叶片和果实的区域特性、发病早期和晚期的周期特性进行分类。

对于给定的正常水果图像，利用目标检测技术获取需要插入斑块的区域坐标，并插入病灶，即可得到扩增的水果病斑图像。这里使用的目标检测技术将会在本文的第 3.3.3 节详细阐述。本研究在这里使用的图像融合技术，与前面在昆虫影响图像的扩增方法中使用的融合手段一致，通过求解泊松方程实现边界的平滑过渡。

病灶插入最直接的思路是图像的拼贴，即前景直接覆盖对应背景区域的像素，如图3.14(a)。但由于病灶区域与健康区域之间存在显著差异，因此过渡边缘会非常不自然。故进一步考虑 α 混合，如图3.14(b)，让转变尽可能平滑，以便在合成图像中隐藏伪影。具体实现时，利用将病斑前景二值化后的灰度图生成权重 α ，根据 $res = (1 - \alpha) * fg + \alpha * bg$ 得到最终的融合图像。这种方法虽然简单快速，但当源图像和目标图像之间存在一些配准错误时，它会模糊细节。

而基于图像金字塔的融合方法，可以将待融合的两张图片分解到不同频段上，针对不同分解层上的特征与细节，采用不同的融合算子以达到突出特定频段上特征与细节的目的。因此本研究还尝试了基于拉普拉斯金字塔的融合方式，如图3.14(c)：(1) 首先得到前后景图像的拉普拉斯金字塔，以及最顶层的高斯金



图 3.12: 决策树和 kmeans 聚类分割病斑结果比较



图 3.13: 病灶图像集

字塔；(2) 生成 mask 图像，确定融合位置，方便后续的叠加融合；(3) 以 mask 作为权值，将前后景每层的拉普拉斯金字塔图像进行相加，得到一个新的图像金字塔；(4) 最后使用单幅图像重构的方式，对这个新的拉普拉斯金字塔以及最高一级的高斯金字塔图像重构，得到最终的融合图像。

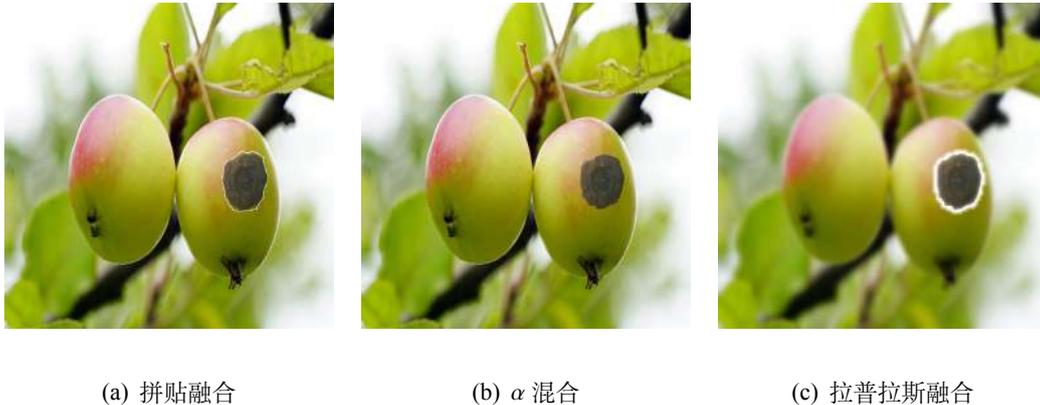


图 3.14: 病灶插入的融合方法比较

拉普拉斯金字塔融合考虑将源图像不同频段上的特征和细节进行融合，以期达到完美的过渡效果。但在实际应用时，拉普拉斯融合需要前后景图像的尺寸相同，且分辨率最好为 2 的次方，否则在下采样后再上采样时，图像无法相加。且由于前景外边缘是白色像素，因此融合后反而会引入新的噪点，使边缘更加明显。

综合比较下，泊松融合对图像的梯度域进行处理，提高了源图像和目标图像之间的空间和颜色一致性，融合效果最好。在实现时，强制融合结果与前景之间具有相似的颜色。进一步利用散度算子实现梯度约束，使得融合结果和背景图像具有相同的高频信号，来捕捉原始背景图像中丰富的细节，如纹理和边缘。从而令最终得到的图像真实自然。

如图3.15，展示了利用分割后融合技术得到的水果病斑扩增图。使用该扩增方法，可以任意分割典型病斑组成新的病灶集。分割手段还可以移植到其余的病症中。除了插入至目标检测返回的坐标位置外，还可以人为确定病斑融合的区域，为扩增集的多样性和随机性提供了更多选择。

3.3.2 特征模拟技术

根据本文第二章中，对于患有炭疽病的水果果实和叶片的农业领域调研，可以总结出：在发病前期，果实以及叶片上都会出现褐色至黑褐色的圆形病斑，周围包裹黄色晕圈。从针尖到豆粒大小不等，星罗分布在果实表面或叶片上部。发



图 3.15: 分割后融合扩增结果

展到后期，病斑会逐渐扩大，形成同心轮纹，且在中央散生褐色至黑色小点。

而从特征分析的角度出发，有研究 [35] 表明：色调在 (45~47)、(48~50) 区间下区分病变叶片和正常叶片的效果比较明显。此区间也是叶片病态部位的色调值，正常叶片在此区间内的色调分布很少。果实方面，一般产生病变的水果会出现褐黑色至灰白色的色调，而正常水果则较少会出现。由此可以看出，病斑区域和正常区域之间，存在可以明显区分的图像特征。

因此，借助第二章中阐述的图像特征提取方法，可以进一步数据化分析病斑的颜色、形状和纹理特征。结合总结得到的领域知识，通过特征模拟的方式，实现水果病斑图像的数据扩增。

表 3.1: 病斑颜色及形状特征量均值

	果实-早期	果实-晚期	叶片-早期	叶片晚期
RGB 均值	[128.89, 94.64, 89.68]	[130.98, 102.28, 75.30]	[131.35, 96.78, 39.45]	[169.14, 138.83, 104.86]
HSV 均值	[71.63, 87.98, 129.07]	[14.99, 107.85, 130.98]	[18.18, 184.94, 131.35]	[15.64, 101.71, 169.16]
二阶颜色矩	[31.75, 27.42, 27.30]	[11.92, 11.96, 12.70]	[20.89, 22.06, 18.10]	[25.82, 28.77, 27.95]
三阶颜色矩	[36.86, 31.89, 31.45]	[15.05, 15.08, 14.93]	[23.97, 25.32, 21.38]	[29.28, 32.12, 31.48]
圆形度	1.1204	1.1866	1.2142	0.96485
复杂度	88.0218	10.59	61.7713	15.62485

表3.1展示了对果实和叶片上不同周期的炭疽病病斑，颜色及形状的特征量分析结果。其中颜色矩的计算是针对 RGB 颜色分量，因此一阶颜色矩即为 RGB

均值。表中数据是通过多组真实病斑计算其特征量后取均值得到的。注意由于患病早期会在多处产生病斑，因此统计其圆形度时，仅对其中一小块完整病斑分割处理。图3.16展示了对分类后的典型病斑图像，切割其纯病灶区域形成的颜色直方图。为了形成对比，我们还对正常的果实和叶片区域构造颜色直方图。

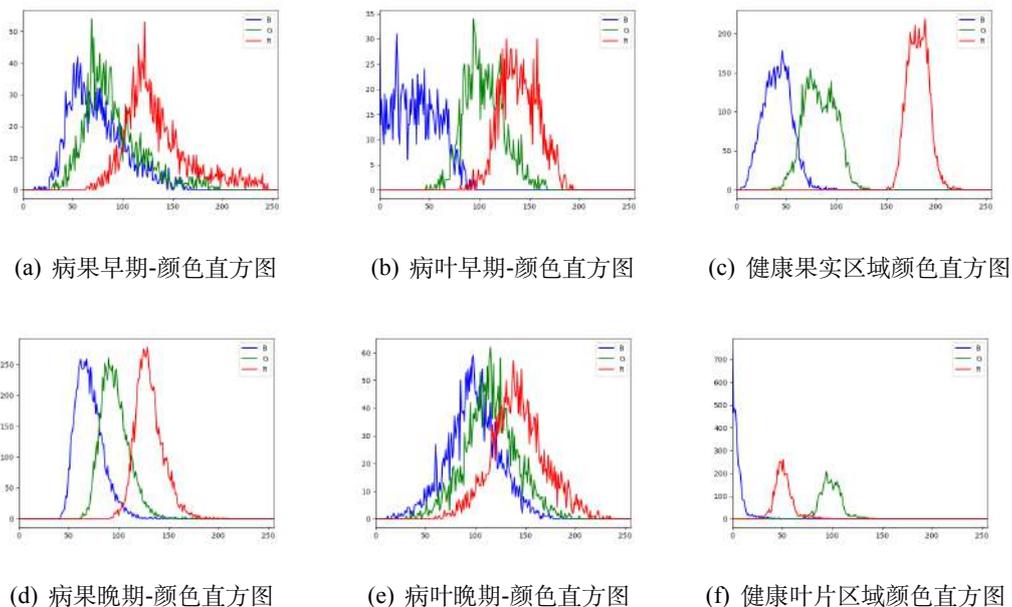


图 3.16: 病斑颜色直方图比较

分析表3.1中的数据可知，病斑区域 RGB 三通道的分布范围和均匀程度较为相似，颜色分布具有对称性，即颜色分量的偏斜度相近。而从颜色直方图的展示结果看，早期病斑区域的 RGB 分量凸起集中在左侧，说明亮度较暗，而晚期则集中于中部，说明病灶细节丰富。且病灶区域 RGB 三分量分布的重叠程度，明显大于正常区域，因此病斑在色彩呈现上以灰褐色调为主。

此外，我们对病斑的形状特征进行分析。上表中的结果显示，早期病斑由于分布杂乱，因此复杂度较高，但四种病斑的圆形度都接近于 1。进一步通过霍夫变换进行圆检测，可以发现病灶区域近似为圆形，如图3.17。

纹理上，我们分别对整体的病斑图像、分割出的方形病斑（带有部分非病斑像素的裁剪图像）以及纯病斑区域获得 LBP 统计直方图。这里使用等价模式的 LBP 算子，具有旋转不变性和灰度不变性。

分析图3.18可以发现，纯病斑局域具有相似分布的纹理，能够明显与病斑边缘以及背景图像区分开来。除了根据 LBP 算子提取纹理特征外，本研究还使用灰度共生矩对病斑的对比度、熵、相关性等进行了统计。我们对 10 张病斑图像

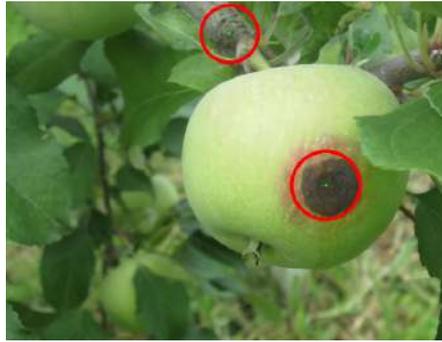


图 3.17: 使用霍夫变换对水果病斑图像进行圆检测

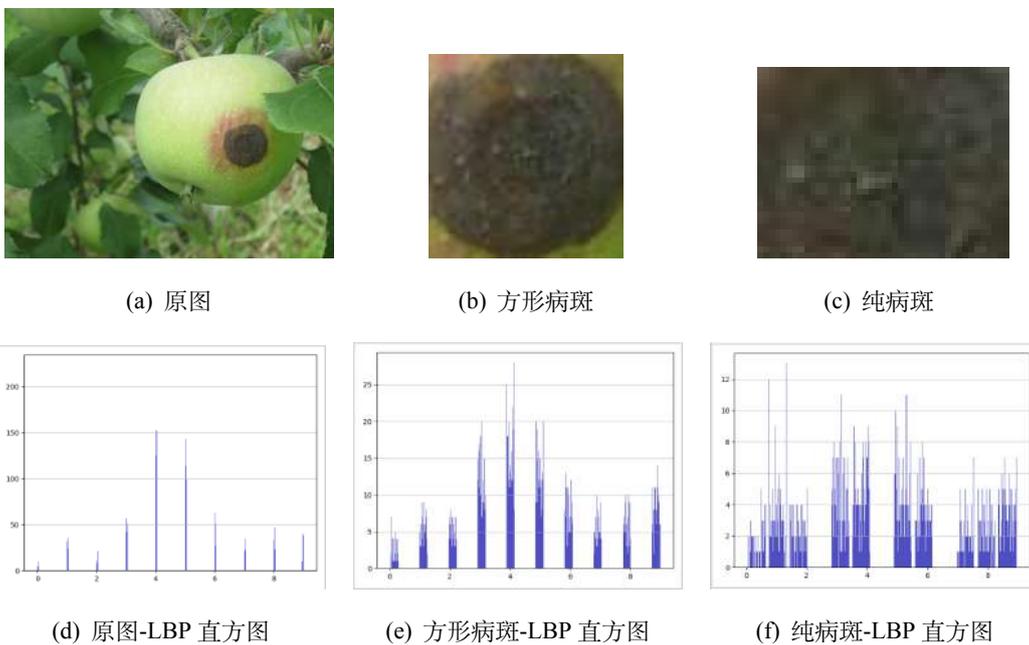


图 3.18: 病斑图像不同区域 LBP 统计直方图比较

进行裁剪，得到对应的方形斑块和纯病斑区域，统计他们的灰度共生矩特征量均值，最终得到的统计结果见表3.2。

结合第二章中的特征量分析表中数据。从对比度和角二阶矩可以看出，纯病斑区域相较于其他区域，局部变化的幅度较大，且沟壑较深、纹理细致。较高的熵值和较低的相关性说明斑块的灰度分布复杂，纹理不具有 consistency。而反差分矩阵值又说明病灶区域虽然变化幅度大，但是变化缓慢。因此进行病斑模拟时，应该考虑增加细致且复杂的纹理，同时适度增强区域对比度以贴近现实病灶清晰而反差大的纹理效果。

将数据统计结果与领域性的专业分析相结合，可得到病斑的整体特征。故本研究在模拟时将病斑近似为圆形，以统计分析的 RGB 均值或 HSV 均值作为

表 3.2: 不同区域灰度共生矩的特征量均值

	原图	方形斑块	纯病斑
对比度 CON	[0.19, 0.29, 0.19, 0.28]	[0.26, 0.38, 0.26, 0.35]	[0.32, 0.47, 0.31, 0.39]
差异性 DISL	[0.17, 0.23, 0.17, 0.23]	[0.24, 0.33, 0.25, 0.32]	[0.30, 0.40, 0.28, 0.35]
反差分矩阵 HOMO	[0.92, 0.89, 0.92, 0.89]	[0.88, 0.84, 0.87, 0.84]	[0.85, 0.81, 0.86, 0.83]
熵 ENT	[0.28, 0.26, 0.28, 0.26]	[0.41, 0.39, 0.41, 0.39]	[0.55, 0.52, 0.56, 0.54]
相关性 COR	[0.99, 0.98, 0.99, 0.98]	[0.94, 0.91, 0.93, 0.91]	[0.53, 0.31, 0.55, 0.42]
角二阶矩 ASM	[0.08, 0.07, 0.08, 0.07]	[0.17, 0.15, 0.17, 0.15]	[0.31, 0.27, 0.31, 0.29]

病斑底色，通过模拟同心轮纹、增强病灶对比度并晕染外边缘，来还原病斑纹理。特征模拟的具体实现又细化为早期病斑模拟与晚期病斑模拟，其主要思想为在正常的水果图片上进行圆形扣除与像素点特征调整，具体步骤描述如下。

1、早期病斑模拟

炭疽病感染初期，在叶片和果实上都是呈现为针尖大小的黑色圆形斑点，外围有黄色晕圈。因此对于叶片和果实上的炭疽病斑点，模拟操作较为相同：

- (a) 通过目标检测技术（具体方法的阐述见本文第 3.3.3 节）得到 roi 区域的坐标，后续病斑形成将在这个范围内进行。
- (b) 找到目标区域的中心，在以其为中点、面积为 roi 面积 30% 的正方形区域内，随机获取 N 个坐标点作为小病斑的圆心。其中 N 是生成小病斑的数量，默认为 6。
- (c) 若方域边长为 R，再随机生成 N 个 0.02-0.06 之间的小数，与边长相乘作为病斑半径，以确保早期病斑相对于果实或叶片的面积占比较小。
- (d) 在以步骤 2 得到的坐标点为圆心、步骤 3 得到的数据为半径的区域内，修改原图像的 RGB 值或 HSV 值，形成黑褐色的斑点，同时根据真实病斑的 RGB 波动形成纹理。
- (e) 从斑点外边缘向内侧区域进行螺旋状遍历，用像素点四周的 RGB (HSV) 均值代替原像素点值，以达到增加外部晕染的效果。这里考虑螺旋状遍历，符合晕染由外及内，向内部平稳过渡的特质。
- (f) 最后对图像进行均值滤波，以造成模糊外边缘的效果。

早期病斑模拟是通过改变指定圆域内像素点的特征得到，但实际上炭疽病病斑的形状并非是完美的圆形，因此最后两步的晕染和模糊外边缘，实际是对

图像进行羽化。让修改后的像素点与其周边区域过渡更加自然，病斑状态更加贴合实际。整个模拟步骤详细的算法伪代码见 Algorithm5。

其中，螺旋状遍历的具体操作如下图3.19，这里用以当前像素点为中心的 3×3 方域内，9个像素点的特征（RGB/HSV）均值替代原像素点处特征值，来达到平滑过渡的效果。但传统的遍历方式容易导致病斑周围的晕圈在区域的下半部分渐趋消失，因此采用由外向内辐射的方式遍历。这样外部像素受晕染操作的影响最大，更接近健康区域的特征，能够更好地还原病斑从中心蔓延的效果。

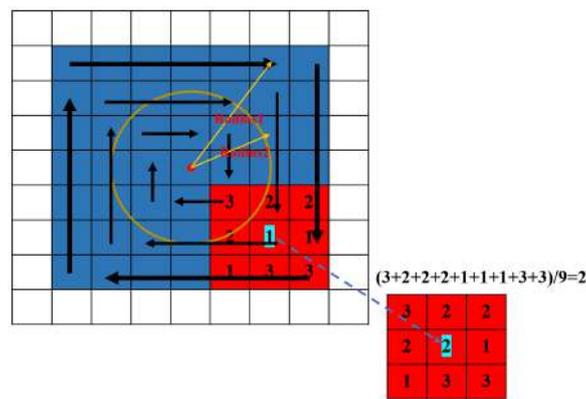


图 3.19: 螺旋遍历晕染病斑

需要注意的是，这里遍历结束的位置是根据病斑半径 $radius$ 的大小设定。为了保证病部中央侵染时间最长，因此颜色最深的性状特点，本研究为 $radius$ 的最小值设定了限制条件，确保遍历不会停止在病斑中心。最终得到的水果果实及叶片，感染炭疽病早期的扩增结果如图3.20。



(a) 果实病斑模拟



(b) 叶片病斑模拟

图 3.20: 炭疽病早期病斑模拟扩增

2、晚期病斑模拟

随着炭疽病侵染范围的扩大,病斑中心会逐渐出现较为清晰的同心轮纹,轮纹颜色较于斑块底色稍显更深,同时病部中央色泽加深、软腐凹陷,在病斑表皮下会出现无数稍由褐变黑的小粒点。

在对晚期的病斑进行模拟时,需要将果实和叶片区分开来实现,因为扩大的病斑可能会侵染至叶片边缘,因此需要获取叶片轮廓。此外,叶片炭疽病根据受到侵染时外部环境的不同,具体分为急性炭疽病和慢性炭疽病,二者病状有所差异。因此扩增时除了在叶片上模拟形成大块轮纹状病斑外,还可以对叶缘进行黄褐色的颜色特征改变。具体操作步骤如下,注意前提步骤与早期病斑模拟的第一步一致,通过目标检测得到病斑生成区域的坐标。

2-1 果实炭疽病晚期病斑模拟

- (a) 设定 roi 的中心位置为病斑圆心,形成面积为 roi 区域面积 10% 的圆形,将其内部像素点的 RGB/HSV 值设置为对病斑色调进行特征分析时得到的平均值。
- (b) 中心以随机生成的黑色像素填充,以模拟病部中央散生的黑色粒点。
- (c) 从里向外画同心圆,轮纹色调由深至浅,得到同心轮纹,轮纹圈数与半径根据果实或叶片 (roi) 的面积占比大小设置。轮纹默认 3 圈,最大轮纹半径默认为病斑半径的 0.6。
- (d) 适度增强病斑区域的对比度,同时在病斑边缘添加晕染,操作类似于早期病斑的晕染,螺旋访问像素点并用其四周像素点的色调均值替换其原本色调。
- (e) 为生成的图片增加透明度通道,羽化边界、平滑图像。

2-2 叶片炭疽病晚期病斑模拟

- (a) 首先将读取的 RGB 图像转化为二值图像,方便提取轮廓。
- (b) 对图像中的目标区域使用 findContours 进行轮廓检测,仅保存轮廓的拐点信息,把所有的轮廓拐点保存到 contours 向量内,拐点与拐点之间直线段上的信息点不予保留,如图3.21(a)。
- (c) 利用拐点数量确定最大拐点集,作为目标叶片边缘的像素信息,如图3.21(b)。根据 roi 在原图中的坐标,将轮廓点的像素坐标映射至原图中。

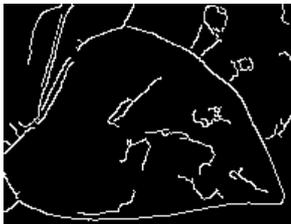
根据第二章对炭疽病性状表征的总结:对于患有慢性炭疽病的水果叶片,其叶缘会逐渐向内散生黄褐色或灰白色的病斑;而感染急性炭疽病的水果叶片,多

Algorithm 5: 早期病斑模拟算法

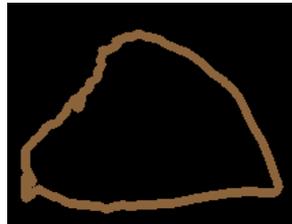
```

1 Input: Image to be processed: img;
2 Coordinates of the upper left corner of the target detection box:  $x_1, y_1$ ;
3 Coordinates of the lower right corner of the target detection box:  $x_2, y_2$ ;
4 Numbers of the early spots to be simulate: num;
5 Output: Fruit image with early anthrax spot;
6  $center\_x, center\_y \leftarrow (x_1 + x_2)/2, (y_1 + y_2)/2$ ;
7  $square \leftarrow (x_2 - x_1) * (y_2 - y_1) * 0.3$ ;
8  $edge \leftarrow \text{Round}(\text{Sqrt}(square))$ ;
9  $lu, ld, ru, rd \leftarrow$  Compute diagonal coordinates of the central square of the
   detection frame;
10 foreach i in num do
11    $percent \leftarrow \text{Random}(\text{range}=(0.02,0.06))$ ;
12    $radius_i \leftarrow \text{Round}(edge * percent)$ ;
13 foreach i in (lu, ld) do
14   foreach j in (ru, rd) do
15     foreach index in num do
16        $dis \leftarrow \text{Distance}(img(i, j), (center\_x, center\_y))$ ;
17       Set the RGB value or HSV value of img according to dis and
          $radius\_index$ ;
18       Generate texture according to RGB value fluctuation of real spots;
19 Traverse pixels spirally to dyeing the outer of spot;
20  $result \leftarrow \text{MeanBlur}(img)$ ;
21 return result;

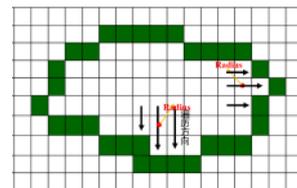
```



(a) roi 轮廓检测



(b) 目标叶片轮廓点集



(c) 叶边缘病斑模拟的遍历方式

图 3.21: 叶边缘轮廓获取示意图

从远离主脉的叶缘内侧开始发病，病部会从淡青略带褐色的水烫状小斑，逐步扩大为轮纹状的水渍大斑。因此对于这两种情况的病斑特征模拟，处理方式有所差异。承接上步，对于慢性炭疽病模拟，在得到叶边缘轮廓点集后，

- (i) 在与原图 shape 相同的全零矩阵（黑色背景）中，使用 drawContours 绘制外轮廓，选择较粗的轮廓线同时将绘制颜色设置为黄褐色，即可得到由叶边缘向内扩散的侵染效果。

而对于急性炭疽病模拟，则效仿果实晚期病斑的模拟，区别在于遍历像素点的方式不同。

- (i) 使用 drawContours 函数同时设定较粗轮廓线，得到外轮廓的所有点集（采集外轮廓时仅保留了拐点）。
- (ii) 根据 roi 长度和宽度的大小，判断框选区域内叶片近似于横形长势或是竖形长势。
- (iii) 如果是横形长势，则首先从外轮廓点集中找到 y 轴最大的点，即叶子外边缘中最底端像素；如果是竖形长势，则在外轮廓点集中找到 x 轴最大的点，即叶子外边缘最右端像素，将其作为最边缘点。
- (iv) 根据特征分析的结果，设定病斑的最大半径是叶片 roi 短边的 15%。从最边缘点向内缩减 roi 短边 10% 的像素点，将其确定为病斑中心点。横形长势向上找，y 值减小；竖形长势向左找，x 值减小。
- (v) 在框选区域内通过遍历条件构建轮纹病斑，模拟方式与果实晚期病斑模拟的步骤相同。注意如果到达叶片的外边缘，则停止遍历，具体遍历方式见图3.21(c)。

叶片急性炭疽病模拟的过程中，病斑中心像素点的位置是根据叶片的长势确定的，这与农业领域的病状表现吻合。

之所以选择横形叶片底端向上的位置，以及竖形叶片右端向左的位置，是方便在遍历时避开叶边缘，即当遍历到存在于外轮廓点集中的像素时，内层循环可以及时退出，不再对外边缘以外的像素点进行处理。由于果实晚期病斑的模拟过程与早期病斑的模拟类似，因此这里只详述了急性炭疽病叶片晚期病斑模拟过程的算法伪代码，具体见 Algorithm6。

最后对于病斑的羽化处理，早期病斑模拟时采用的是均值滤波，因为病斑本身较小，外围像素值较少，可以对图片进行整体模糊。但晚期病斑由于面积较大，在图像中比较明显，因此考虑增加透明度通道的方式晕染病斑外缘。此外还

可以利用腐蚀的方式进行羽化，但该方法只能针对灰度图，对彩色图像的羽化效果并不好。

增加透明度通道以羽化病斑的操作步骤如下：

- (i) 首先根据原图的 shape，建立初始值全为 0 的单通道透明度图层。
- (ii) 同时构建距离矩阵，计算 roi 域内所有像素点与病斑中心的距离。
- (iii) 设定透明度在距离病斑中心 0.8 个半径，和 1.2 个半径的像素点之间变化，外侧透明度值为 0.5（半透明），内侧透明度值为 1（不透明），中间像素依据公式 3.19 按照距离等比变化

$$\alpha[i, j] = 0.5 + \frac{dis[i, j] - 0.8 * radius}{1.2 * radius - 0.8 * radius} * (1 - 0.5) \quad (3.19)$$

- (iv) 将 0-1 之间的透明度矩阵映射到 0-255 之间，转换成图像表示的数据形式。
- (v) 病斑羽化后得到 RGBA 图像，转化到 RGB 空间后存储。

羽化操作的具体实现见算法伪代码 Algorithm7，最终得到水果感染炭疽病晚期的病斑模拟结果如图 3.22。由于早期病斑的生成位置具有随机性，因此对于每张输入图像可以得到不同的扩增结果。而晚期病斑可以通过修改病灶生成的区域，以及病斑特征，如色调值、半径以及轮纹圈数等获得定制化、多样性的病斑模拟图像。

3.3.3 目标检测技术

无论是借助真实病斑图像的分割后融合技术，还是对病斑特征分析后对正常图像进行调整的特征模拟技术，都需要获得病斑插入区域或形成位置的坐标。因此我们在扩增前，应利用目标检测技术得到 roi（region of interest）的坐标。

对于需要进行扩增的水果果实图像来说，目标检测的目的是识别正常图像中不同种类的水果，并返回检测框坐标；而对进行样本扩增的水果叶片图像来说，目标检测的结果需要返回图像中叶片的区域坐标。

考虑到目标检测场景的复杂，即一张图像中可能存在多种类别的水果或多个叶片，还要兼顾扩增方法的效率，因此本研究选取 YOLO-v3 作为扩增前返回 roi 坐标的目标检测技术。主要训练水果识别模型和叶片识别模型，以应对不同的样本扩增需求，同时保证检测结果能及时返回且准确有效。

目标检测主要分为 R-CNN[53] 和 YOLO[54] 两个系列，前者基于区域检测，而后者是基于区域提取的算法。本研究选用 YOLO 算法，将生成候选区域和目



(a) 果实病斑模拟



(b) 叶片急性炭疽病病斑模拟



(c) 叶片慢性炭疽病病斑模拟

图 3.22: 炭疽病晚期病斑模拟

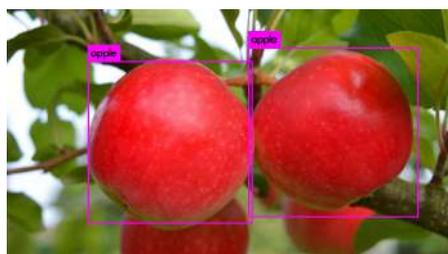
标检测的两个阶段合二为一，虽然舍弃了部分精度但基本上达到了实时检测的效果，检测结果如图3.23。

1、网络结构

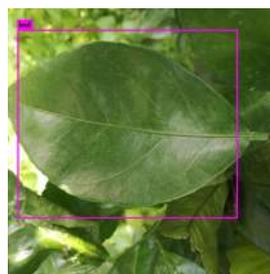
YOLO-v3 是全卷积网络，除卷积层和上采样层外，还借鉴残差网络采用了跳连接。其基本思想是通过输入图像进行特征提取，结合默认边界框 anchor 将特征图谱中蕴含的关键信息提取出来。由于网络中不使用任何形式的池化，因此可以防止低级特征的丢失。

YOLO-v3 首先将图片划分成网格，代替滑动窗口操作，以避免精细划分下产生的巨大计算量。每个单元格对应 A 个检测框， A 的数量根据目标检测需要鉴别的类目数量确定，以确保当目标中心位于单元格内感受野中时，其中一个检测框可以用来检测目标。

直接预测边界框的宽度和高度会导致训练期间的梯度变得不稳定，因此现在大多数目标检测器预测对数空间变换。同时简单地预测与预定义的默认边界框 anchor 之间的偏移，然后对输出进行公式3.20-3.23的变换以获得预测框坐标。YOLO-v3 有三个 anchor，可以为每个单元预测三个边界框。anchor box 的参数



(a) 果实检测框



(b) 叶片检测框

```
终端 1  X  BoxValue.txt  X
1  左部边框的位置: 650 顶部边框的位置: 103 右部边框的位置: 1109 底部边框的位置: 565
2  左部边框的位置: 218 顶部边框的位置: 140 右部边框的位置: 662 底部边框的位置: 582
```

(c) 果实检测框坐标

```
终端 1  X  BoxValue.txt  X  obj.data
1  左部边框的位置: 66 顶部边框的位置: 0 右部边框的位置: 413 底部边框的位置: 669
2
```

(d) 叶片检测框坐标

图 3.23: YOLO-v3 目标检测结果

是通过 kmeans，统计在训练集中最常出现的边界框形状和尺寸而得到的。

$$b_x = \sigma(t_x) + c_x \quad (3.20)$$

$$b_y = \sigma(t_y) + c_y \quad (3.21)$$

$$b_w = p_w e^{t_w} \quad (3.22)$$

$$b_h = p_h e^{t_h} \quad (3.23)$$

其中 b_x 、 b_y 、 b_w 、 b_h 是预测边界框的中心坐标 x 、 y 、宽度和高度； t_x 、 t_y 、 t_w 、 t_h 是网络的输出； c_x 和 c_y 是当前单元格的左上角坐标； p_w 和 p_h 是候选框的维度。前两个公式通过 sigmoid 函数来预测边界框的中心坐标，使输出值保持在 0-1 之间。YOLO 不预测边界框中心的绝对坐标，而是通过预测偏移确定，因此还要根据网格单元的左上角坐标进一步计算。后两个公式通过对输出进行对数空间变换，然后与 anchor 相乘来得到预测的边界框尺寸。

2、类别置信度

YOLO-v3 输出的每一个预测框都包含 $5+A$ 个参数，前 4 个值是上一节计算得到的检测框中心坐标以及宽和高；第 5 个值表示预测框置信度，区间在 $[0, 1]$ 之间； A 对应待检测的类目数量，这 A 个值即为不同类别的置信度，表示检测目标属于特定类别的概率。

在 YOLO-v3 之前的版本，YOLO 对类别置信度的计算使用 softmax 函数。但 YOLO-v3 进行了改进，选用 sigmoid 函数，原因是对类别分数进行 softmax 意味着类别之间是互相排斥的，但实际分类时可能会有不互斥的情况，类别之间存在属于或包含的关系，所以选用 sigmoid 函数更为严谨。因此在计算时，如果一张图像经过特征提取后的某一类输出经过 sigmoid 函数约束后大于 0.5，就表示该边界框负责的目标属于该类。

预测框置信度分数表示当前预测框是否包含物体，以及对包含对象的自信程度，但并不确认物体类别。根据该置信度分数，可以过滤数量庞大的边界框，低于设定阈值的会被删除，以减少计算量。再利用非最大值抑制（NMS）解决同一图像多重检测的问题，这是一个迭代-遍历-消除的过程。简单来说，首先将所有框的得分排序，选中最高分及其对应的框。遍历其余的框，如果和当前最高分框的预测类别相同且重叠面积大于一定阈值，则将该框删除。从未处理的框中继续选一个得分最高的，重复上述过程。

通过目标分数阈值和非最大值抑制来获得检测结果的具体步骤可以归结如下：（1）首先利用目标置信框阈值，对目标分数低于阈值的边界框，将其所有属性值设置为零。（2）利用已有的边界框属性——中心坐标以及边界框的高度和宽度计算剩余框的一对角点坐标，计算 IOU（重叠面积）值。（3）按类别进行 NMS，如果有两个相同类别的边界框的 IOU 大于阈值，则具有较低类别置信度的边界框被去除。

3、多尺度预测

YOLO-v3 的检测层在卷积步长分别为 3 种不同尺寸的特征图上进行检测，每个单元网格使用 3 个 anchor 来预测边界框，最终 anchor 的总数为 9。除卷积网络外，还利用类似 FPN（feature pyramid networks）的上采样和融合方法。

输入图像在经过多次卷积之后，特征图谱所包含的语义信息将更加高级。因为网络层次加深后，感受野会更大，相对应网络的分类效果就会更好。但在一个尺度很小的特征图谱中预测目标框的位置，其偏差将会变大，因此为了平衡目标检测和分类两种目的下，对于尺度不同的需求，提出了层级金字塔（FPN）的概念，将通道先降后升，再加入不同尺度的信息，输出各种新的特征图谱。

在多个尺度的特征图谱上做检测，帮助网络学习有助于小物体检测的细粒度特征。这样网格单元划分得越细，检测出的物体就可以越精细。

通过 YOLO-v3 训练得到的果实检测模型可以较为精准地返回 roi 坐标，但缺点在于如果检测目标为新的水果品类，那么模型则需要对应的数据集重新训练。而叶片检测模型仅对单独拍摄的叶片图像有较好的检测结果，至于自然场景下叶片密集的图像，其得到的结果不尽人意。如果要对栽育端拍摄的果叶图

像训练叶片检测模型，在数据集标注和模型训练时耗费的成本会比较高，因此考虑人为选定 roi 坐标。

3.3.4 自动生成技术

分割后融合技术和特征模拟技术都是主要使用图像处理算法来进行负样本扩增的，虽然充分考虑了领域知识且对原始的图像数据集没有数量要求，但在真实性方面依旧无法达到以假乱真的程度。此外由于这两种方法在扩增前还需要目标检测来确定病灶插入和模拟的位置，因此也降低了样本生成的效率。

为了得到更贴近现实场景下拍摄图片的扩增结果，且在确保图像质量的前提下进一步提高扩增手段的效率，本研究选用生成式对抗网络，来自动生成高分辨率且高置信度的扩增负样本。

需要注意的是，由于水果炭疽病具有传染性和周期性，因此上述的负样本扩增都会从叶片感染和果实感染、病害早期和病害晚期这几个方面考虑。但受到现有水果图像数据集的限制，没有办法找到炭疽病不同阶段下大量的水果病斑图片作为训练数据，因此这里不再考虑病害周期的影响。

1、深度卷积对抗网络

深度卷积对抗网络 (DCGAN) 是医疗和工业领域被应用来实现图像数据扩增的最广泛的 GAN 变体。该技术将深度卷积神经网络 CNN 和生成对抗网络 GAN 相结合，将生成式模型引入无监督训练，同时利用卷积网络强大的特征提取能力来提高生成网络的学习效果。

不同于常规的 CNN 网络通过卷积下采样进行特征提取，通过池化下采样来降低特征的维度，深度卷积对抗网络中的 G 网络利用反卷积来实现上采样，将一张噪声图片变为更高分辨率的目标图像。这里的反卷积又称为转置卷积，可以看作卷积操作的逆操作，能将提取的特征图还原到和原图同样尺寸的大小。

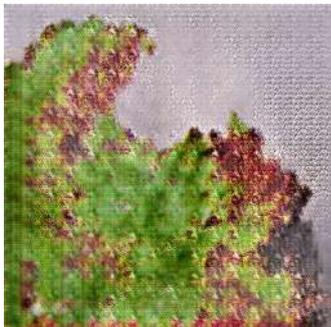
DCGAN 的原理与对抗生成网络的原理一致，区别在于使用 CNN 来实现对应的生成网络和鉴别网络，此外还进行了如下改进：

- (1) 去除池化层。生成网络使用反卷积进行上采样，鉴别网络利用固定 stride 的卷积层代替池化层，使下采样不再是抛弃某些固定位置的像素值，而让网络自己去学习下采样方式。
- (2) 在卷积层后增加批标准化层 (batch normalization)，用于对卷积层输出进行归一化，帮助网络更快更好收敛，防止参数迭代更新过程中的震荡和初始化不良导致的训练问题。

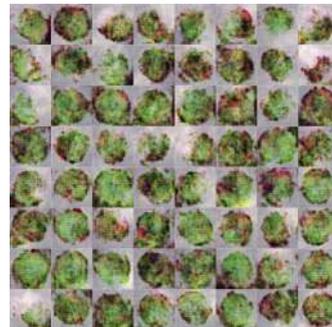
- (3) 移除全连接层，使网络变成全卷积网络。因为全局均值池化虽然有助于模型的稳定性但也降低了模型的收敛速度，因此在实现时将生成器噪声 reshape 成 4 维张量，来替代全连接操作。
- (4) 更改激活函数。生成器除了输出层采用 Tanh 外，全部使用 ReLU 作为激活函数，判别器所有层都使用 LeakyReLU 作为激活函数。

使用 DCGAN 实现水果病斑图像的负样本扩增，主要是借助生成网络，从随机生成的噪声图片开始，利用反卷积进行上采样，在对抗学习中不断优化图片细节、提升分辨率，直至得到与输入图像尺寸一致、无法被鉴别器识别的高质量图片。

以叶片病斑扩增为例，首先制作训练数据集，将 1109 张病叶图片调整为统一尺寸和格式，指定输出图片的大小并训练 200 个 epoch。为了保证图片的精度，设置图片输出尺寸为 1024×1024 ，但是训练时间也因此大幅增长。最终生成的病叶效果如图 3.24。观察扩增结果可以发现，DCGAN 对于训练数据集的依赖程度比较深，高质量生成模型的训练成本相对较高。但优点在于扩增只需要足够数量的训练数据集，就能自动生成所需的目标图片。且由于结果与真实图像不相关，因此得到的扩增样本数量不受限制，一定程度上能够提升受训模型的对抗性。



(a) 单张病叶图片



(b) 病叶图片集

图 3.24: DCGAN 生成水果病叶

2、循环一致对抗网络

除了通过对抗生成网络自动生成病斑图片外，还可以借助风格迁移的思想，操纵 CNN 修改正样本中的图像表示，基于源图像创建新的图像结构，得到扩增的负样本。

2-1 风格迁移思想

风格迁移的工作原理是通过改变 CNN 上的顺序表示，使某一幅图像的样式可以转移到另一幅图像上，同时保留其原始内容。风格迁移扩展了光照变化，并允许对不同的纹理和艺术风格进行编码。对图像数据扩增来说，这在某种程度上类似于颜色空间和光照变换。

使用风格迁移进行数据扩增，可以根据实际需要决定从哪些样式进行采样，同时帮助模型提升从模拟环境到现实世界过渡时的泛化能力，避免了物理条件的损耗。

在本研究中，由于是借助风格迁移的思想来扩增水果病斑图像，因此与传统的风格化图片有所不同。常见的风格迁移应用是将真实图片或简单的涂鸦转变成具有给定艺术品外观的风格化图像，在迁移纹理的同时使源图像具有艺术品的绘画风格、素描风格或抽象风格。但在水果病斑图像的数据扩增应用中，内容图像被确定为无病斑的正常水果图像，用于进行迁移的风格图像集由感染炭疽病病斑的水果图像组成。固定受侵染的水果图像为风格图像集，当需要对负样本进行扩增时，可以根据设定的风格得到不同分布（果实、叶片）下水果的炭疽病发病图。但是由于这里的风格图像不存在所谓的绘画风格，因此就需要对原始风格迁移的常见网络进行针对性的调整。

在水果图像的负样本扩增中，风格迁移要实现的是提取病斑图像的风格特征，将其应用于未患病的正常图片上，这是一个图像到图像的转化过程。可以将目标简单看作一个映射 $G: X \rightarrow Y$ ， X 代表未患病的正样本图像集， Y 表示患有炭疽病的负样本图像集。该映射的学习通常需要成对的训练数据，例如相同场景和拍摄角度下，同一颗苹果未患病图片与患病图作为训练样本对。显而易见的是，这种数据的获取是困难且成本高昂的。因此考虑在不成对的训练样本下，学习这种映射关系。

2-2 损失函数改进

第二章中提到，风格迁移中使用的损失函数通常分为两类：（1）像素到像素的损失函数，用于比较相同空间坐标下的像素，例如 L_2 损失以及感知损失（通常以粗略水平计算）；（2）全局损失函数，通过比较在整个图像上收集的统计数据捕获风格和纹理，例如 Gram 损失。但在不成对的训练样本下，映射就变成了高度缺乏约束的学习任务，无法使用常见的损失函数来进行限制。

虽然缺乏成对样本的监督，但可以在集合层次上利用监督。在 X 域中获取一组图像用于训练生成器 G ，使得 $\hat{y} = G(x)$ 。当对应的鉴别器 D 无法区别 Y 域中的真实图像 y 和生成图像 \hat{y} 时，生成器 G 达到最优性能。理论上，此时 G 生成图像的数据分布与实际图像的真实分布 $p_{data}(y)$ 趋于一致，因此该对抗损失可

以表示为：

$$L_{GAN}(G, D_Y, X, Y) = E_{y \sim p_{data}(y)}[\log D_Y(y)] + E_{x \sim p_{data}(x)}[\log(1 - D_Y(G(x)))] \quad (3.24)$$

其中， $G(x)$ 旨在生成与域 Y 看起来相似的图像， D_Y 旨在在尽可能区分生成图像与实际样本 y 。因此，对抗损失的目标最终可以表示为：

$$\min_G \max_{D_Y} L_{GAN}(G, D_Y, X, Y) \quad (3.25)$$

然而，这种转换并不能保证单个输入 x 和输出 y 以有意义的方式配对——有无限多的映射 G 可以产生 Y 上的相同分布。与此同时在实践中，也很难孤立地优化对抗目标，例如所有输入图像映射到相同的输出图像，出现模式崩溃问题，使得优化无法取得进展。因此需要对目标加上进一步的限制条件。

考虑到在图像转换中，映射的逆映射应不对原图产生改变。例如将 RGB 图像映射到 HSV 空间后，再映射回 RGB 空间，其像素细节不应该有所损失。因此假设映射 G 的逆映射为 $F: Y \rightarrow X$ ，那么对于域 X 中的任意一张图像 x ，经过转换循环后都应该返回原图像： $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$ 。这可以被称之为正向循环一致，那么类似地，对于来自域 Y 的每个图像 y ， G 和 F 也应该满足反向循环一致性： $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$ 。因此在原本对抗损失的基础上，将映射 G 与逆映射 F 耦合，引入循环一致性损失来保证单个输入可以映射到期望的输出，如图 3.25。

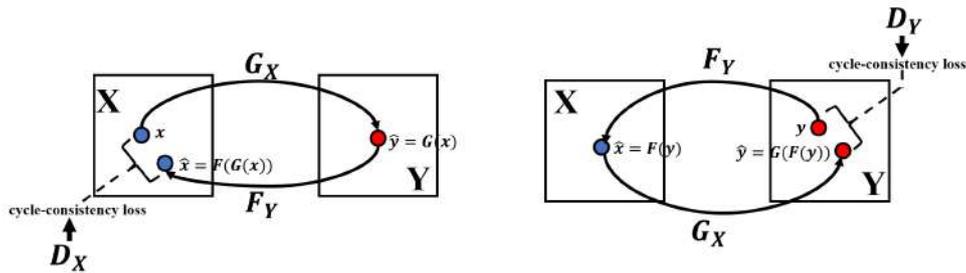


图 3.25: CycleGAN 原理示意图

同时训练映射 G 和 F ，通过循环一致性损失来鼓励 $F(G(x)) \approx x$ 和 $G(F(y)) \approx y$ ，具体损失公式如下所示：

$$L_{cyc}(G, F) = E_{x \sim p_{data}(x)}[\|F(G(x)) - x\|_1] + E_{y \sim p_{data}(y)}[\|G(F(y)) - y\|_1] \quad (3.26)$$

由于模型包括了两个映射，因此应该训练两对生成器与鉴别器，其中 D_X 用于区分图像 x 与转化的图像 $F(y)$ ， D_Y 用于区分图像 y 与转化的图像 $G(x)$ 。对于逆映射对应的生成器 F 与鉴别器 D_X ，其对抗损失的形式与正向映射一致，目标可以定义为：

$$\min_F \max_{D_X} L_{GAN}(F, D_X, Y, X) \quad (3.27)$$

综上，最终的整体损失函数就由两部分组成，分别是对抗性损失和循环一致性损失，公式如下：

$$\begin{aligned} L(G, F, D_X, D_Y) = & L_{GAN}(G, D_Y, X, Y) \\ & + L_{GAN}(F, D_X, Y, X) \\ & + \lambda L_{cyc}(G, F) \end{aligned} \quad (3.28)$$

其中 λ 用于控制两个目标的相对重要性，最终目的是为了解决：

$$G^*, F^* = \arg \min_{G, F} \max_{D_X, D_Y} L(G, F, D_X, D_Y) \quad (3.29)$$

2-3 网络实现与扩增结果

实现时采用实时风格迁移的生成网络架构，整体是一个先降采样后上采样的过程。其中包含 3 个卷积层来提取输入图像的特征，后接 9 个残差块来实现达到转换的同时保留原始图像特征的目标。这里每个 Reset 模块是一个由两个卷积层构成的神经网络层，残差块的数量根据输入图片的尺寸决定（输入图片被 resize 成 256×256 的分辨率）。最后利用反卷积层从特征向量中还原出低级特征，得到生成图像。

鉴别器使用 70×70 的 PatchGANs 网络，其本质依旧是卷积网络，通过提取特征来确定图片的具体类别。这种补丁级鉴别器架构比完整的图像鉴别器具有更少的参数，并且能够以全卷积的方式应用于任意大小的图像上。

准备 trainA 和 trainB 两份数据集，其中 trainA 代表域 X，包含了未患病的水果图像，trainB 代表域 Y，包含的是罹患炭疽病的水果图像。根据病症感染表现位置的不同，分别训练叶片和果实的两种生成模型。

由于这里主要是利用循环一致对抗网络实现负样本扩增，因此只关注 G 映射： $X \rightarrow Y$ ，即从 trainA 数据集中抽出一部分图片作为 testA，生成对应的 fake 图片，扩增效果如图 3.26。

虽然循环一致对抗网络可以使用不成对的图片进行训练，但在数据准备时还是需要 X 域和 Y 域尽可能相似。如果是两个完全不相关的图像领域，那么网络就很有可能不奏效。此外和其他生成式对抗网络一样，训练数据集的质量和



图 3.26: CycleGAN 生成病叶病果

数量越好，模型的生成效果就越好，这也可以从上面的扩增结果中看出。病叶与健康叶片在数据集域上相似度较高，且数量总和达到了 1500 张，因此模型生成的病叶就具有较强的真实性，肉眼基本无法分辨真实图片和伪造图片。而苹果果实的生成模型，由于训练数据集数量的限制，且所摄图片大多是处于自然环境中，干扰因素较多，无法保证域的相似性，因此其生成结果并不是很好。

对比 DCGAN 和 CycleGAN 这两种自动生成的扩增方式，前者虽然对训练数据集要求较高，但生成结果不依赖真实图像，因此理论上可以获得任意数量的扩增图片。CycleGAN 得到的结果真实度高、效果好，对训练集的限制也比较少，但是生成时需要通过对真实图片进行修改，扩增数量有限。因此本研究将二者综合应用，以应对不同的扩增需求和实现目标。

3.4 本章小结

第三章是本研究的重点，主要阐述了水果病斑图像的扩增方法实现。在本章中，我们首先对水果图像的采集场景及特征进行了分析。考虑到扩增图像主要是用于病虫害智能化监测系统的训练，因此图像采集常发生在种植端。而在

自然环境下，容易受到不可抗的天气情况以及偶发性的生物因素的影响，对图像传感器产生干扰。基于上述观察，本文将正样本扩增的重点放在图像的雨雪雾场景扩增，以及昆虫影响的扩增上来。

对原图像增加雨雪雾等自然条件因素，可以通过修改原图的亮度、增加噪声和模糊处理等手段完成；而昆虫干扰则需要利用形态学分割或自动抠图的方式，提取其他图像中的昆虫前景，再使用泊松融合将其过渡平滑地融入到作为背景的正常水果图像中。

在农业领域中，不同的病害会呈现出不同的性状特点。本研究以水果易感且不存在种间差异的炭疽病为目标，来实现水果图像的负样本扩增。主要分为三个技术路线进行。首先是根据发病的周期与分布，将早期果实、晚期果实、早期叶片和晚期叶片上的炭疽病病斑利用决策树和 kmeans 从真实图像中分割出来。对病灶区域进行进一步的处理，如过滤噪声和缺失像素点填充后，得到代表性斑块。然后使用泊松融合，将典型病灶根据目标检测返回的区域坐标，插入不同的正常水果图像中。第二种扩增方法是对病斑本身的颜色、形状和纹理进行分析，利用总结归纳的特征并结合领域知识，从而进行病斑模拟。由于这两种方法都是基于正常水果图像进行处理，因此考虑 YOLO-v3 作为目标检测网络，训练对应的果实识别模型和叶片识别模型，获取后续插入或生成病斑的区域坐标。第三种方法是利用深度学习技术，自动生成水果病斑图像。鉴于 CNN 在图像处理中的卓越表现，故首先考虑应用深度卷积对抗网络 DCGAN，扩增结果显示其对于有较高质量的训练数据集更加友好。借助风格迁移的思想，水果图像负样本的扩增也可以阐述为将病斑图像的纹理特征和风格特点移植到正样本之中，因此考虑训练图像到图像的生成网络 CycleGAN。利用循环一致性重构损失函数，解决了不成对图像的训练问题。扩增结果相比 DCGAN 更加真实，且对于训练数据集的质量和数量更加宽容。

Algorithm 6: 叶片晚期病斑模拟算法

```

1 Input: Image to be processed: img;
2 Coordinates of the upper left corner of the target detection box:  $x_1, y_1$ ;
3 Coordinates of the lower right corner of the target detection box:  $x_2, y_2$ ;
4 Output: Leaf image with late anthrax spot;
5 leaf_img  $\leftarrow$  img[ $x_1 : x_2, y_1 : y_2$ ];
6 leaf_gray  $\leftarrow$  Convert RGB leaf_img to Gray domain;
7 blur_img  $\leftarrow$  GaussianBlur(leaf_gray);
8 thresh  $\leftarrow$  Binary blur_img with setting pixel value as 255 where that is more
   than 127;
9 contours  $\leftarrow$  findContours(thresh);
10 contours_num  $\leftarrow$  size(contours);
11 foreach i in contours_num do
12   Find contoursi which has the largest number of contour point sets;
13 radius  $\leftarrow$  Min( $(x_2 - x_1), (y_2 - y_1)$ )*0.15;
14 if  $(x_2 - x_1) > (y_2 - y_1)$  then
15   Find point p with the largest y value in contoursi;
16   center  $\leftarrow$  ( $p_x, p_y - 0.1 * \text{Min}((x_2 - x_1), (y_2 - y_1))$ );
17   result  $\leftarrow$  Ergodic pixel longitudinally to simulate late anthrax spot;
18 if  $(x_2 - x_1) \leq (y_2 - y_1)$  then
19   Find point q with the largest x value in contoursi;
20   center  $\leftarrow$  ( $q_x + 0.1 * \text{Min}((x_2 - x_1), (y_2 - y_1), q_y)$ );
21   result  $\leftarrow$  Ergodic pixel laterally to simulate late anthrax spot;
22 Enhance the contrast of the disease area moderately;
23 return result;

```

Algorithm 7: 图像羽化算法

```
1 Input: Image to be processed: img;  
2 Output: Disease image with alpha matte;  
3 row, col, channel  $\leftarrow$  img.shape;  
4 alpha  $\leftarrow$  zeros(shape=(row, col));  
5 radius  $\leftarrow$  Shape as img but with single channel and each value in (x,y)  
   represents distance to (0,0);  
6 r_min, r_max  $\leftarrow$  1/3 * Max(radius), 0.8 * Max(radius);  
7 alpha  $\leftarrow$  Set value as 0 if it is less than r_min, as 1 if it is more than r_max;  
8 alpha  $\leftarrow$  Set value between r_min and r_max as transition value depending  
   on distance;  
9 img  $\leftarrow$  Add dimension of img to 4-channel with alpha;  
10 result  $\leftarrow$  Convert RGBA img to RGB domain;  
11 return result
```

第四章 实验设计与结果分析

4.1 数据准备与处理

本研究主要着眼于水果图像的数据扩增，且针对的数据应用场景主要是栽育端的智能监测设备，以协助完成农业告警、对策制定等功能。因此扩增得到的水果图像数据，要求要符合种植端图像传感设备的采集特点，且能够提升监测终端的鲁棒性和精确性。相对应地，用于处理的图像数据也应该是栽育端采集得到的水果图像。

与此同时，由于负样本扩增是以炭疽病为出发点，同时考虑到周期性和传染性的特征，因此将训练数据集粗略划分为水果果实数据集与水果叶片数据集。两类数据集中各自包含未染病的正样本和感染炭疽病的负样本，其中负样本还根据病斑感染周期的不同划分了轻重症（早期和晚期）。

但在公开数据集平台 CVonline 公布的 1571 组数据集中，农业相关的数据集仅有 8 组，且大部分针对的都是叶片分割与鉴别。而与病虫害相关的农业图像数据集，又主要是诸如玉米、小麦一类的农作物图片，水果病害的数据集非常稀少。仅有的水果病斑图像数据集，也都是不同疾病下的果叶照片，且并未根据病害的感染情况进行分类。这些都为本研究的数据获取带来了极大的挑战。

最终经过对现有公开数据集的筛选和网页搜索，获取到栽育端拍摄的、分辨率较高的水果果实正样本图片 50 张，患有炭疽病的水果果实图片 109 张。叶片方面主要使用的是云南农业大学采集到的草莓叶片数据集，其中患病叶片 1109 张，健康叶片 456 张，以及栽育端拍摄的 45 张炭疽病叶片图像和 34 张健康叶片图像。

4.1.1 目标检测数据集

本研究使用基于 TensorFlow 的 YOLO-v3 网络对水果和叶片进行检测，并返回目标框的坐标。实验借助 TITAN XP 的 GPU 辅助计算，部署环境为 Ubuntu18.04 + Conda3 + Python3.7 + CUDA10.0。水果检测模型需要区分出不同类型的水果，因此收集了 275 张水果图像作为训练数据，其中包含苹果图像 94 张、香蕉图像 91 张和橘子图像 90 张。叶片检测模型的训练数据集，是 73 张单独拍摄的橘叶图像与自然环境下拍摄的 45 张橘叶图像混合得到。

除了图像数据外，YOLO-v3 还需要标签进行有监督的训练，每幅图像对应的 label 与图中的检测框有关。在果实检测模型中，由于训练图像可能存在多个水果果实或者多种种类的水果，因此训练图像的 label 数量可能不止一个。其具

体表现形式如图4.1。

```
0 0.7255 0.809322033898305 0.283 0.3813559322033898  
0 0.616 0.3778248587570621 0.248 0.3573446327683616  
0 0.333 0.4696327683615819 0.314 0.3771186440677966  
0 0.493 0.18149717514124294 0.252 0.3149717514124294  
0 0.7745 0.21468926553672316 0.265 0.3305084745762712
```

图 4.1: YOLO-v3 训练集 label 数据形式

其中第一列整数代表类目，因为水果检测模型的训练数据中包含三种品类，所以其类目表示分别为 0、1、2，而叶片检测模型的类目表示只有 0。后面的四个数据代表目标框位置，前两个数据是目标框中心点的 x 、 y 坐标，后两个数据是目标矩形框的宽度 w 和高度 h 。数据被归一化到 0 到 1 之间，将图像的左上角坐标视为 $(0,0)$ ，右下角坐标视为 $(1,1)$ ，即可根据相对位置确定目标框。

数据准备时，首先对收集到的图像使用标注工具 `labelimg` 进行标注，框选出对应物体并输入类目名称即可。将标注好的图片保存到 Annotation 中，即可得到每张图像对应的 xml 标注文件。利用 python 程序，从 xml 文件中获取图像的宽和高，得到标注框对应类名的 index，并根据其左上角和右下角的横纵坐标，计算相对位置，得到 label 数据。将所有的 xml 文件转化为写有 label 的 txt 文件后，就完成了对 YOLO-v3 模型训练集的数据准备。

4.1.2 对抗生成网络数据集

对抗网络的训练一般需要数量较多、分辨率较高的图像数据集。前面提到由于患有炭疽病的水果果实图像数量不足，对于 DCGAN 网络的训练效果不尽人意，因此这里只考虑利用 1565 张草莓叶片图像作为数据集训练深度卷积对抗网络，部署环境如下：Windows10+Anaconda3+Python3.7，不使用 GPU 加速。

数据准备时只需要利用双线性插值的方式，将训练集图像全部 `resize` 为同一分辨率，放入 `train_datas` 文件夹即可。为了保证生成图像的质量，这里不对原图作尺度压缩，而将原来 256×256 的图片扩张到 1024×1024 输入网络，同时设置输出图像的宽和高与输入保持一致。

训练循环一致对抗网络需要两个领域内的图像，对应到本研究中就是 456 张健康叶片与 1109 张患病叶片、50 张正常果实与 109 张病害果实组成的成对数据集，以训练从领域 X 到领域 Y 的图像转换网络。处理时同样需要训练图片的尺度统一为 $N \times N$ ，且 N 应该为偶数。基于 Pytorch 的 CycleGAN 网络在训练时同样使用 TITAN XP 的 GPU 加速计算，主要的实验环境为：Ubuntu18.04 + Conda3 + CUDA10.0 + python3.6 + pytorch1.4.0。

4.2 扩增效果检验指标

对于扩增图片的效果，本研究主要从图片质量、真实性和扩增有效性三个角度检验。图片质量确保扩增结果的可用性，高质量图片作为训练数据可以在一定程度上提高受训网络的精度。高真实度的扩增结果不仅能够迷惑鉴别网络，还能使人在感知上无法对图片的真伪进行区分，保证扩增结果尽可能贴合实际采集场景。而数据扩增技术最终还是要为数据接收对象而服务，因此这里我们将扩增有效性定义为对受训网络精度和鲁棒性的提高程度。

4.2.1 图片质量

图像质量的评价标准包括均方误差 (MSE)、峰值信噪比 (PSNR) 和信息熵 (Entropy)。

1、均方误差

均方误差首先计算原始图像和扩增图像像素值的均方差，然后通过均方差值的大小确定扩增图像的失真程度。MSE 的值越小，则扩增图像质量越好。这个指标在图像质量评价中相当于一个中间的评价指标，原理和最小二乘法相似。公式如下：

$$MSE = \frac{1}{M * N} \sum_{0 \leq i \leq N} \sum_{0 \leq j \leq M} (f_{ij} - \hat{f}_{ij})^2 \quad (4.1)$$

2、峰值信噪比

峰值信噪比是最大信号量和噪声强度的比值，用于衡量图像有效信息与噪声之间的比率，能够反映图像是否失真。PSNR 的值越大，表示生成的图像质量越好。由于数字图像都是以离散的数字表示图像的像素，以大小为 $L * L$ 的图像为例，一般取 $L = 255$ 。公式为：

$$PSNR = 10 * \lg \frac{L * L}{MSE} \quad (4.2)$$

3、信息熵

图像信息熵是图像质量评价的常用指标，它从信息论的角度反映图像信息丰富程度。熵用来描述随机性，可以衡量事物的混乱程度。香农提出信息熵的概念，并指出信息熵是消除不确定性所需信息量的度量，也即未知事件可能含有的信息量。如果某一个变量是高度可预测的，那么他就有较低的熵，相反如果是高度不确定的，则其熵就较高。因此通常情况下，图像信息熵越大，其蕴含的信息量就越丰富，质量越好。

假设某图像有 n 个灰度级，那么对于某个像素点 I 就有 n 种可能的取值。这里灰度级表示了像素点的亮暗差别，如果灰度级越多，那么图像层次越清楚逼真。则该像素点 I 的信息量就可以表示为：

$$I(i) = p_i \log \frac{1}{p_i} = -p_i \log p_i \quad (4.3)$$

其中 i 表示灰度级为第 i 级， P_i 表示第 i 级灰度出现的概率。

而图像的熵实际是所有像素点信息量的总和，计算公式见式4.4。当图像的熵最大时，图像点在 0-255 的灰度级之间均匀分布，图像具有高对比度且灰度变化也较为明显，整体图像质量较好。

$$Entropy = \sum_{i=0}^{n-1} I(i) = - \sum_{i=0}^{n-1} p_i \log p_i \quad (4.4)$$

4.2.2 图片真实性

鉴别图片的真实性，主要是将扩增图片与真实图片进行比较，如果相似度较高，则可以认为扩增结果与真实拍摄的图片比较接近，因此选用结构相似度 (SSIM) 作为扩增结果真实度的衡量指标之一。

此外，由于本研究还利用了生成对抗网络作为扩增手段之一，因此可以借鉴 GAN 网络的质量评估指标。现有的研究提出了一系列评价指标来量化 GAN 的生成效果，这些指标主要着眼于生成图片的真实性和多样性，要求指标结果应与人类感知趋于一致，且不应当有较大的计算复杂度。因此对于扩增图片真实度方面的衡量，我们经由对比之后选用 Frechet Inception 距离得分 (FID) 和最大平均差异 (MMD) 作为扩增真实性的评价指标。

1、结构相似度

随着对人类视觉系统认识的深入，人们开始尝试利用 HSV 的特性来评价图像。结构相似度 SSIM 来源于结构相似理论，被用于衡量两幅图像的结构相似性大小，其最大值为 1，反应了图像的轮廓、细节等相似度，具体表示为：

$$SSIM(x, y) = [l(x, y)]^\alpha * [c(x, y)]^\beta * [s(x, y)]^\gamma \quad (4.5)$$

其中， $l(x, y)$ 表示亮度变化， $c(x, y)$ 表示差异变化， $s(x, y)$ 表示结构变化。

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (4.6)$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad (4.7)$$

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x + \sigma_y + C_3} \quad (4.8)$$

式中 μ_x 和 μ_y 表示图像 x 、 y 的均值，反映了亮度信息（类似于相关系数，避免图像对不齐的问题）。 σ_x 和 σ_y 表示图像的方差，反映了对比度信息。 σ_{xy} 表示图像 x 、 y 的协方差，反映了结构相似性信息。 C_1 、 C_2 、 C_3 为接近于零的常数，防止分母为零时导致结果异常。一般取 $\alpha\beta\gamma = 1$ 。

此外，SSIM 满足以下特性：

(1) 对称性。 $SSIM(x, y) = SSIM(y, x)$

(2) 有界性。 $SSIM \leq 1$

(3) 最大值唯一性。只有当两幅图像完全相同时， $SSIM = 1$ ，否则 SSIM 的值在 0-1 之间。

利用公式4.9对所有图像块的 SSIM 求均值，可以得到整幅图像的结构相似度。

$$MSSIM(X, Y) = \frac{1}{M} \sum_{j=1}^M SSIM(x_j, y_j) \quad (4.9)$$

2、FID (Frechet Inception Distance)

FID 计算真实样本、扩增样本在特征空间之间的距离，得到真实图片与扩增图片提取特征向量后二者之间的均值、协方差的距离评价。当扩增图片和真实图片特征越相近时，均值之差的平方越小，协方差也越小，那么其和（FID）也就越小。因此 FID 值越小，扩增图片与真实图片就越相似，扩增样本的真实性也就越高。

FID 首先使用 Inception-v3 作为特征提取器，去掉最后的 pooling 层，可以在图片中提取到 N 维的高层特征。然后利用高斯模型对特征空间进行建模，再去求解两个特征之间的距离。例如对于已经拥有的真实图像，假设其特征向量满足高斯分布，该分布就能用一个均值和方差确定。那么当扩增的图片与真实图片分布相似时，我们即可认为扩增图片在真实性上趋于实际图片。因此 FID 使用均值和协方差矩阵来计算两个分布之间的距离，较低的 FID 值意味着较高的图片质量和多样性。

FID 指标只是使用 Inception-v3 作为特征提取器，而不依赖图片的具体类别，也不必担心每个类别内部只产生一模一样的图片，导致诸如 IS (Inception Score) 中多样性判别失效，无法预测 mode collapse 的问题。实践发现，FID 对噪声有

较强的鲁棒性，对生成图片的评价质量较高，接近于人类视觉判断。不足的地方在于理论中高斯分布的简化假设在实际中可能并不成立。

与 FID 类似的还有 Kernel Inception Distance (KID) 指标。KID 通过计算 Inception 表征之间最大均值差异的平方来度量两组样本之间的差异。此外，与依赖经验偏差的 FID 不同，KID 有一个三次核的无偏估计值，因而能更好地匹配人类的感官认知。

3、MMD(Maximum Mean Discrepancy)

MMD 在迁移学习中具有非常广泛的应用，它是在希尔伯特空间中对两个分布的差异的一种度量。故可以考虑使用 MMD 度量真实数据集 $real_datas$ 和扩增数据集 $augment_datas$ 的分布之间的距离。若 MMD 距离越小，则表示 $real_datas$ 和 $augment_datas$ 越接近，扩增图像真实度越高、扩增效果越好。

计算 MMD 时，首先选择一个核函数 $K(x, y)$ 来将两个样本映射为实数，例如多项式核函数：

$$K(x, y) = (\gamma x^T y + C)^d \quad (4.10)$$

或高斯核函数：

$$K(x, y) = \exp(-\|x - y\|^2) \quad (4.11)$$

则 MMD 距离可表示为：

$$E_{x, x' \sim p_{real}}[K(x, x')] - 2E_{x \sim p_{real}, y \sim p_{aug}}[K(x, y)] + E_{y, y' \sim p_{aug}}[K(y, y')] \quad (4.12)$$

实际计算时，由于不可能遍历所有样本求期望，因此需要使用批量样本估计 MMD 值：对于来自于真实样本集的 n 个样本 x_1, x_2, \dots, x_n 和来自扩增得到的 n 个样本 y_1, y_2, \dots, y_n ，MMD 的估算值计算见公式4.13。注意这里 MMD 是使用样本估计的，因此即使两个数据集完全相同，得到的 MMD 值也未必为 0。

$$\frac{1}{C_n^2} \sum_{i \neq i'} K(x_i, x_{i'}) - \frac{2}{C_n^2} \sum_{i \neq j} K(x_i, y_j) + \frac{1}{C_n^2} \sum_{j \neq j'} K(y_j, y_{j'}) \quad (4.13)$$

4.2.3 扩增有效性

扩增有效性是数据扩增最重要的衡量标准。由于本研究主要针对的是水果病斑的数据图像扩增，旨在帮助训练种植端的病虫害监测模型，支持信息及时告警以及个性化培育策略制定。因此我们在这里选用分类模型作为受训网络，来考察扩增图像是否能够提高对健康水果和患病水果的识别效果。具体来说，就是将数据受众设定为简单的 CNN 二分类网络，通过扩增前数据和扩增后数据对模型训练结果的影响，确定扩增图像是否有效

这里选择使用简单的卷积神经网络作为受训模型的主要原因有三：

(1) 卷积神经网络 (CNNs) 对于大型的图像处理，具有出色表现。它可以通过改变网络的深度和广度来控制学习图像中实体的能力，同时还可以对图形的本质（即统计学的平稳性和像素相关性的位置）做出有力且基本正确的假设。相比于有着同样大小层数的标准前馈神经网络，CNNs 具有更少的连接和参数，更容易被训练。

(2) 自从 2012 年 AlexNet 在 ILSVRC 中将卷积神经网络的巨大优势带入人们视野后，CNNs 就迎来了迅速的发展，其变体被广泛应用于图像领域。自然而然地，农业图像数据的反馈终端也必然是以 CNNs 作为基石。因此我们如若证明扩增结果对基本的 CNN 分类网络有效，那么相信移植到监测水果病虫害的农业智能设备中时，也同样可以发挥作用。

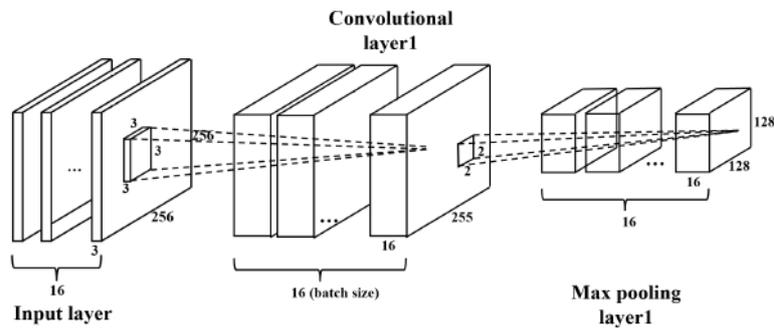
(3) 正如本文在第一章中提到的，训练数据和模型架构是决定最终模型效果的关键因素。随着 CNNs 的发展，现有的研究在网络深度、优化方法、卷积核设置和学习率设计等方面都进行了改进，以达到更加优越的性能。但为了更明显地体现训练数据本身对模型的影响，这里我们只选用简单的 CNN 网络，以最小化模型架构对结果的提升，使得数据表现更加突出。

本研究中用于测试有效性而构建的简单 CNN 网络是三层卷积模型，最后一层使用全连接，如图 4.2 所示。

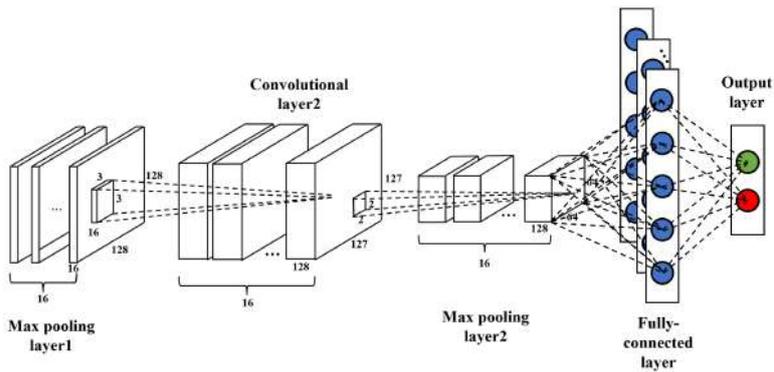
卷积层具有局部感知和参数共享的特点，通过对隐含单元和输入单元间的连接加以限制，以达到减少参数和降低计算量的目的。而权重共享既使得重复单元能够对特征进行识别，而不考虑它在可视域中的位置；也让特征抽取变得更加有效，因为它极大减少了需要学习的自由变量的个数。通过控制模型的规模，提升了卷积网络对视觉问题的泛化能力。

激活函数层提供了网络的非线性建模能力，使神经网络可以更好地解决较为复杂的问题。常见的激活函数包括 sigmoid 函数和 tanh 函数，但是他们都存在饱和性，容易在训练过程中导致梯度消失。ReLU 函数缓解了梯度消失的问题，增强了神经网络的稀疏表达能力，但可能会出现神经元无法被激活的情况。对 ReLU 进行改进后得到了 LReLU、PReLU 与 RReLU 等激活函数，一定程度上缓解了神经元死亡的问题，但在函数复杂度上不如 ReLU 简单。综合比较后，本研究选用 ReLU 函数作为 CNN 网络架构里的激活函数。

池化层的工作实际是下采样，主要是为了减小特征图，降低参数矩阵的规模以及全连接层中的参数数量，根本目的是为了防止过拟合。池化操作对每个深度切片独立，因此不会改变深度的大小。规模一般为 2×2 ，如果输入单元的大小不是 2 的整数倍，则会采取边缘补零 (zero-padding) 的方式将其补成 2 的



(a) Conv1+Pooling1



(b) Conv2+Pooling2+Fully-connected

图 4.2: 验证有效性使用的 CNN 网络架构

倍数, 再进行池化。常见的池化运算有: 最大池化 (取 4 个点中的最大值代替原本的 4 个点)、平均池化 (取 4 个点的均值进行代替) 和高斯池化 (借鉴高斯模糊的方法来完成池化)。

全连接层将学到的特征表示映射到样本标记空间, 根据特征的组合进行分类, 可以减少特征位置对分类带来的影响。在实现过程中, 可以把全连接层看作为卷积核是 1×1 的卷积。

这里 CNN 的每层是 $y = wx + b$ 的决策模型, 卷积层产生特征向量, 根据这些特征向量代入 x 进行计算。因此在每层中定义了卷积层的初始化参数, 包括权重和偏置。

第一层卷积层的卷积核形状为 $[3, 3, 3]$, 加上偏置后经过 ReLu 函数激活。对每张输入图像, 中间层得到的 feature map 数量为 16。后接池化层选择最大池化, 池化窗口的高和宽都设置为 2, 滑动步长也设为 2, 以全 0 的方式填充。使用局部响应归一化 (LRN) 函数进一步防止过拟合, LRN 计算公式如下:

$$b_{x,y}^i = \frac{a_{x,y}^i}{(k + \alpha \sum_{j=\max(0,i-n/2)}^{\min(N-1,i+n/2)} (a_{x,y}^j)^2)^\beta} \quad (4.14)$$

其中，偏移值 k 设为 1，缩放因子 α 为 0.001/9.0，指数 β 设置为 0.75，邻域长度 $n/2$ 为 4。

CNN 的第二层与第一层一样，不同之处在于卷积核形状为 [3,3,16]，输出的 feature map 数量为 128。最后使用全连接，需要对 max_pooling2 层输出的特征向量进行 reshape，将其映射到 4096 维。由于训练时对数据按照 16 的 batch size 进行了分批，因此最终输出的大小为 [batch_size, class_num]。模型在训练过程中利用 softmax 交叉熵函数计算损失，并采用 AdaGrad 算法进行优化。

为了控制变量，我们保持训练网络的数据集数量一致（正样本数量和负样本数量都相同），网络架构以及训练参数都不变，根据最终得到模型的分分类效果确定本研究的扩增方法是否有效。

4.3 实验设计与结果分析

1、扩增图像质量分析

首先对扩增图片的质量进行衡量,选用均方误差(MSE)、峰值信噪比(PSNR)、熵(Entropy)和对均方误差开根号的均方根误差(RMSE)以及将 RMSE 的值归一化到 (0,1) 之间的归一化均方根误差(NRMSE)作为评价标准,与扩增前的原图进行比较,得到评估值。具体计算方式已经在上一节中进行了详细阐述,质量评估结果见表4.1。

表 4.1: 扩增图像质量评估

	MSE	RMSE	NRMSE	PSNR	Entropy
雨场景扩增	206.14	14.36	0.08	24.99	7.49
雾场景扩增	1375.67	37.09	0.21	16.75	6.71
雪场景扩增	448.9	21.19	0.16	21.61	7.74
昆虫影响扩增	425.13	20.62	0.12	21.85	7.57
病斑分割后融合扩增	36.13	6.01	0.05	32.55	7.21
病斑特征模拟扩增	111.68	8.93	0.05	31.36	7.46
DCGAN 扩增	3291.58	57.37	0.36	12.96	7.36
CycleGAN 扩增	180.17	13.42	0.09	25.57	7.11

注意这里由于基于 DCGAN 的扩增方法，并不借助任何原有图像，可以自动生成，因此对于 DCGAN 扩增的质量评估指标值的计算（除 Entropy 以外），在训练集中选用肉眼上最为接近的图像作比。

而对于其余扩增方法的质量评估，本研究在每种扩增方法下，挑选了 5 组典型的扩增前后图像作为对照组，计算扩增生成的图片对应于原图的

失真程度，然后取均值。信息熵计算则直接通过所有扩增图像的信息熵均值得到。

通过对扩增图像的质量评估结果进行分析，可以发现，病斑分割后融合扩增和病斑特征模拟扩增所得到的图像，NRMSE 的值最小、PNSR 值最大，说明其与原图像之间的差异最小，扩增结果质量较高。DCGAN 扩增得到的图像与原图的失真程度最大，这主要是因为对照原图是人为选取的肉眼上较为相似的真实图片，因此像素表现上才会有较大差异。但 DCGAN 扩增结果的信息熵表现比较良好，说明图像中还是包含了较为丰富的信息量，图像质量有所保证。在三种天气情况的模拟扩增中，雾场景扩增的失真程度最大，这主要是由于雾场景扩增在实现时是以图像中心为雾浓度最高点，依据距离向外扩散，而改变图像中的像素值的，因此对图像中的所有像素点都进行了干扰。而雨场景扩增和雪场景扩增，对像素值的影响仅体现在雨丝或雪花覆盖的区域，因此与原图的差异不及雾场景扩增大。也正是由于雾场景扩增在实现时模糊了全图，因此该方法得到的图片信息熵也比较低。

2、扩增图像真实性分析

接着，我们对扩增图像的真实性进行衡量，选用上一节中提到的结构相似度 (SSIM)、Frechet Inception 距离得分 (FID) 和最大平均差异 (MMD) 值作为评估指标。

真实性主要考察扩增出的图片与真实场景下拍摄图像的差异，因此首先确定不同扩增手段所对应的真实样本。对于正样本扩增来说，需要收集种植端在雨雪雾三种天气条件下拍摄到的图像，以及昆虫影响下的果实照片。对于负样本扩增，只需要将真实的水果炭疽病病斑图作为对比即可。这里真实图片的选取，考虑了感官上与扩增图像的相似性：例如雨场景扩增的原图是橘子树果实照片，那么其对应的真实图片就是雨天的橘子图像。若昆虫扩增选取的前景是毛毛虫，扩增区域限制在苹果果实上，那么其对应的真实图片就是爬着毛毛虫的苹果。而负样本扩增的真实图片根据果实的种类、病斑的周期与分布选取。虽然人为选定对照组在一定程度上降低了扩增图像与真实场景图像的结构差异，但依旧会对评估数据造成影响。

对于结构相似度值的获取，沿用图像质量评估时的思路，每种扩增手段使用 5 组真实图片与扩增图片形成的对照组，计算他们的均值得到各自扩增方法的 SSIM 值 (DCGAN 扩增的对照图依旧选取的是肉眼上相似的图片)。

FID 提取真实图片和扩增图片的特征向量，计算二者均值和协方差的距离，MMD 度量的是真实数据集与扩增数据集分布上的距离，它们都是用于评估 GAN 网络生成结果的指标。因此对基于 DCGAN 的扩增方法，只需要对训练

数据集（病斑图像）与扩增结果计算度量值即可。而基于 CycleGAN 的扩增方法，其训练集包含 trainA 和 trainB，其中 trainB 是本研究需要自动生成的真实图像样本，所以将 trainB 中的数据与扩增结果作比得到评估值。

其余的扩增方法由于不是通过模拟真实数据分布得到的，因此需要进一步的实验设计。根据 MMD 和 FID 的要求可知，进行度量的两组数据集，其样本量可以不同，但特征数量必须相同。因此在量化除自动生成外的 6 种扩增方法的真实性时，对每种扩增应用如下步骤：

- 首先选取一张源图像 x ，如苹果果实图片，同时获得对应当前扩增方法的真实图片 x_{real} 。例如对应天气条件扩增的雨雪雾场景下拍摄的苹果图片、对应负样本扩增的感染炭疽病早期的苹果果实等。
- 对原图像应用扩增，注意需要与真实图片对应，例如负样本扩增时要根据真实图片染病的周期和分布确定具体扩增规则。
- 调整扩增的参数（如雨势大小、病斑插入的位置等）得到多张扩增图片 x_{aug} ，对真实图像 x_{real} 和扩增结果 x_{aug} 计算 FID 值和 MMD 值，注意需要将真实图片 resize 成与原图像相同的尺寸。
- 重复上述的步骤 3-5 次，每次选取不同的源图像，取均值得到该扩增方法的 FID 值和 MMD 值。

最终得到的有关扩增结果真实性的度量值如下表 4.2：

表 4.2: 扩增图像真实性评估

	SSIM	FID	MMD
雨场景扩增	0.138	587.81	4.1050
雾场景扩增	0.362	498.11	4.6864
雪场景扩增	0.062	876.37	5.0043
昆虫影响扩增	0.448	223.22	6.7073
病斑分割后融合扩增	0.518	170.17	4.1959
病斑特征模拟扩增	0.289	203.08	3.3927
DCGAN 扩增	0.084	332.93	0.9316
CycleGAN 扩增	0.882	120.07	0.3030

分析上表的数值可以看出，在正样本扩增的四种方法中，昆虫影响扩增图像的 MMD 值最高，即与源数据的分布最不相同。这主要是因为昆虫干扰的水果图像，昆虫所处的位置以及昆虫本身形态可能差异较大。但该扩增方法的 FID

值较小，说明真实样本和扩增样本在特征空间的距离相近。而对于三种天气因素的扩增，得到的样本与真实图像在结构上面相似度很低，这主要是由于选取的真实图像与扩增结果的背景差异较大，且本身雨雪雾场景就会对图像整体产生影响，雨丝和积雪分布的不同造就了整体结构差异的显著。因此他们在特征空间相距也很远，对应的 FID 值就很大。DCGAN 是自动生成的图片，因此虽然选择了视觉上较为相似的真实图像，但在结构表现上依旧差异很大，SSIM 值最低。但整体样本集的分布相近，从较小的 MMD 值就可以看出。基于 CycleGAN 的扩增方法得到的图片，与真实的病斑图像相比，无论是结构相似度，还是特征向量间的距离、数据集的分布，效果都是最好的。这主要有两个原因，一是 CycleGAN 网络在训练时就旨在将原图像映射到真实数据域中，所以该网络从架构设计方面就自然地会去逼近真实图片的数据分布；二是不同于其他扩增方法，用于计算真实度量值所使用的图片是人为设定的，评估 CycleGAN 扩增时，用来实验的数据都是从与训练集同一批分割出来的验证集图像，因此确保了数据结构的相似。

3、扩增图像有效性分析

因为最终的扩增图像还是服务于农业领域的智能终端，因此对于扩增方法有效性的衡量，是基于一个简单的 CNN 网络，将其作为智能设备的网络架构替代。扩增结果对于该网络训练精确性和鲁棒性的提升，是本研究最为关注的问题，故实验主要分为两部分。

第一部分主要关注扩增数据对于网络抗干扰能力的提升：

- 首先使用真实图像数据训练 CNN 网络。在具体实现过程中，病果分类模型 1-1.1 由 57 张健康果实图片与 206 张病变果实图片组成的数据集训练得到；病叶分类模型 1-2.1 由 1109 张草莓病叶和 456 张健康叶片训练得到。
- 保持负样本数据集不变，应用本研究提出的 4 种正样本扩增技术以及筛选过的通用扩增方法分别对果实数据集中的正样本和叶片数据集中的正样本进行扩增。
- 具体地，对于健康果实图片，使用 12 张经过雨雪雾扩增和昆虫干扰扩增得到的图片，结合利用通用扩增技术得到的 13 张图片，替换原来真实正样本数据集中的 25 张图像，组成新的水果果实正样本数据集（与模型 1-1.1 的正样本数量保持相同），训练 CNN 网络得到新的病果分类模型 1-1.2。
- 对于健康叶片图像，利用本研究新提出的 4 种扩增得到 40 张图片，使用通用扩增技术得到 110 张图片，再结合真实的 306 张健康叶片图像组成病

叶分类正样本数据集，训练得到新模型 1-2.2。注意上述选用的通用扩增技术保持安全性，即并不改变图像的 label。

- 为了与单独使用基础扩增的效果进行对照，本研究通过翻转、增强对比度等扩增方法，以及设定图像旋转角度、平移位置等参数来安全扩增正样本。得到基础扩增后的果实图片 25 张、叶片图片 150 张，与真实数据组成新的正样本数据集（图片数量保持一致），分别训练病果分类模型 1-1.3 和病叶分类模型 1-2.3。
- 将模型 1-1.1、1-1.2 和 1-1.3，以及模型 1-2.1、1-2.2 和 1-2.3 组成对照组，同时对测试数据集进行分类。注意测试集由真实图片组成，其中包含许多感官上具有迷惑性的图片，如飞虫落在果实上、积雪的叶片等。

设置训练的 epoch 为 1000，最终的模型精度都可以到达 1，但是在数据集有限的情况下，后期可能会出现过拟合的现象。因此我们取 6 次实验下最早达到精度为 1 的 epoch——300 为界，比较 6 个模型之间精度和损失的变化，具体数据见表 4.3。表中 100_acc 表示 epoch 为 100 时模型的精度，100_loss 表示 epoch 为 100 时模型的损失，test_acc 是通过 20 张易混淆的真实图片进行测试（果/叶），得到的模型分类准确率。

表 4.3: 扩增图像有效性评估 1——鲁棒性

	modell-1.1	modell-1.2	modell-1.3	modell-2.1	modell-2.2	modell-2.3
100_acc	0.75	0.8125	0.125	0.5	0.6875	0.5
100_loss	0.6813	0.6700	0.7182	0.6938	0.6617	0.6962
200_acc	0.875	0.75	0.75	0.625	1.0	0.8125
200_loss	0.5508	0.6243	0.6542	0.5829	0.0141	0.3652
300_acc	0.875	1.0	0.875	1.0	1.0	0.875
300_loss	0.2570	0.0594	0.3442	0.0373	0.1013	0.6093
test_acc	0.8	0.9	0.85	0.85	0.95	0.85

实验结果表明，使用本研究提出的正样本扩增技术训练的 CNN 模型能够最快最好地达到最优。且对于存在干扰的样本也能良好分类，在测试集中获得最高的准确率。由此得出，本研究提出的正样本扩增方法，能够提升受训模型的鲁棒性。同时在训练条件有限的情况下，较以真实数据和基础扩增训练得到的模型，本研究的扩增技术可以一定程度提高模型的精度。

进一步地，我们着重在第二部分研究扩增技术对于网络精度的提升。为了凸显本研究扩增方法的优越性，我们依旧以真实图像和通用扩增作比，将模型目标划分为病果分类与病叶分类，为网络准备了六份训练数据。分别是：

- 由现实场景下拍摄图片组成的真实数据集 1，与鲁棒性的实验数据相同：果实数据集 57/206 (positive / negative)、果叶数据集 456/1109；
- 采取通用扩增方法得到的扩增数据集 2，果实数据集 25+32/206 (base_aug + true_pos / true_neg)、果叶数据集 150+306/1109，同样沿用鲁棒性实验中的扩增数据；
- 采用本研究的扩增方法得到的扩增数据集 3。对于果实数据集，使用本研究的正样本扩增技术得到 12 张图片，结合通用扩增得到的 13 张图片替代真实数据集中的 25 张原始图片。再使用本研究的负样本扩增技术，得到 40 张负样本图像，与真实的 166 张病果图片合并，重新组成负样本数据集。同样地，叶片数据集的组织方式可以拆分为：正样本 =110 张基础扩增图 +40 张正样本扩增图 + 306 张真实图像，负样本 =1009 张真实图像 + 100 张负样本扩增图。

由于对比的训练数据沿用了鲁棒性实验中的数据，因此只需训练扩增数据集 3 得到病果分类模型 2-1 和病叶分类模型 2-2。模型测试使用了新的数据集——叶片测试数据集由 100 个正样本和 100 个负样本组成，其中正样本是对健康叶片图像经过基础扩增得到的，数据具有对抗性，而负样本全部为真实病叶图像；果实测试数据集由 30 个正样本和 30 个负样本组成，来源与叶片测试集相同，正样本是真实果实图像经过基础扩增的结果，负样本是真实病果图像。实验得到的测试准确率见表 4.4。

表 4.4: 扩增图像有效性评估 2——精确性

	healthy_acc	disease_acc	overall_acc
modell-1.1	0.09	0.98	0.535
modell-1.3	0.28	0.97	0.625
model2-1	0.36	0.98	0.67
modell-2.1	0	1	0.5
modell-2.3	0.4	0.6	0.5
model2-2	0.17	1	0.583

分析上表发现，对于叶片分类模型 modell-1.1、modell-1.3 和 model2-1 来说，全部基于真实图像数据训练的网络对于病叶的检测效果很好。但当数据（正样本）具有对抗性时，抗干扰能力不强。而基于基础扩增得到的数据训练生成的网络，虽然具有较强的鲁棒性，但存在漏检病叶的风险。而以本研究提出的扩增方法训练得到的网络，无论是鲁棒性还是精确性方面，都能达到综合最优。对

于果实分类模型 model1-2.1、model1-2.3 和 model2-2，上述的分析依旧成立，但 model2-2 受限于训练数据，对抗性方面的表现不如模型 model1-2.3 好。根据整体的测试准确率 overall_acc 可见，本文提出的扩增方法，与真实数据集和基础扩增相比，可以较为明显地提高模型的预测精度。

最后研究扩增图片对于目标检测网络的影响。这里以第三章中使用的 YOLO-v3 作为测试网络，训练能识别出健康水果果实的检测模型 3-1，和能识别罹患炭疽病与否的果实检测模型 3-2。应用正样本扩增技术对原图进行干扰，将得到的图像送入模型 3-1，发现模型出现漏判或错判，如图 4.3。说明正样本扩增对图像本身产生了一定程度的影响，能够干扰成熟模型的判断。同样，利用负样本扩增技术得到炭疽病果实图像，使其经过模型 3-2 进行病果检测，最终成功识别出生成的病果果实，如图 4.4。这样的实验结果也表示，本研究生成的负样本可以欺骗训练良好的模型，侧面反应了扩增结果的真实性和有效性。

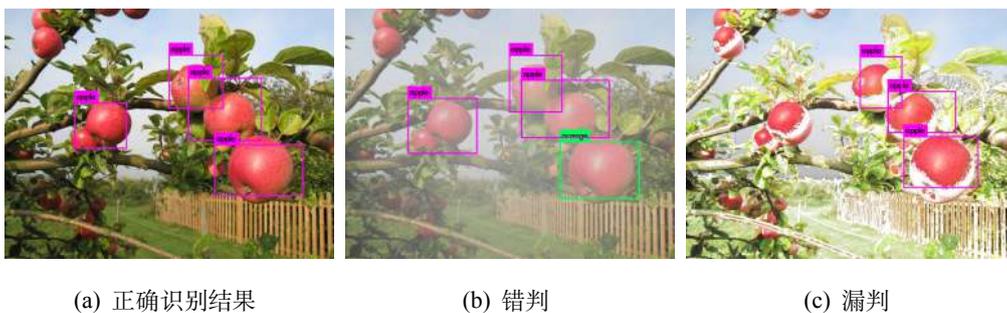


图 4.3: 水果检测模型对扩增正样本的识别效果

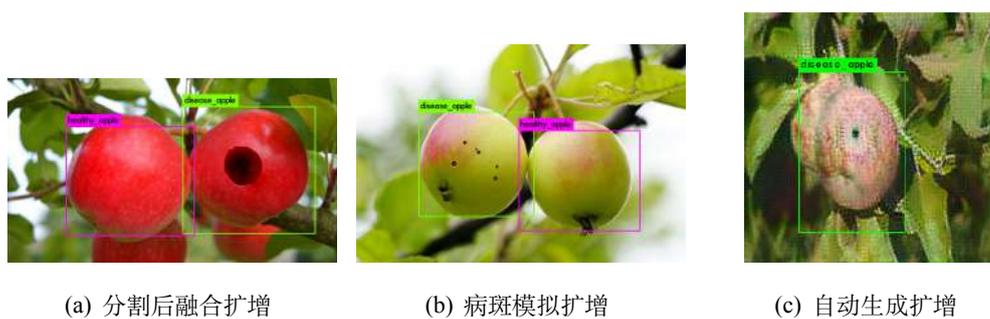


图 4.4: 病果检测模型对扩增负样本的识别效果

上述实验证明，本文提出的扩增方法能够生成质量较高、真实性较强的水果炭疽病图像，从而利用有限的真实数据训练种植端的病虫害监测设备，帮助提高其鲁棒性和精确性。这既在一定程度上缓解了由于图像数据缺乏而导致的模型过拟合问题，同时也降低了数据采集的成本，为智慧农业的发展奠定了基

石。此外，本研究提出的扩增思路也可以移植到其他类型的病害扩增上。用户根据扩增需求和应用场景的不同，可以简单调整方法中的参数或进行组合调配，实现理论上数量不受限制的图像扩增。

4.4 本章小结

本章首先介绍了研究过程的数据准备与处理，由于负样本扩增使用了目标检测技术和自动生成技术，因此需要对各自的网络训练数据集进行相应处理。而对于扩增效果的检验，本文从图像质量、真实性和扩增有效性三个维度，挑选了不同的指标作为评估标准。实验重点在有效性的衡量上，因此设计了一个简单的 CNN 分类网络作为栽育端病虫害监测设备的架构替代，来确保最终的扩增图片在提升模型鲁棒性和精确性方面都有良好的表现。最终的实验结果表明，本研究中提出的 8 种扩增方法所生成的图像，质量都比较高，其中自动生成的扩增结果与真实图像分布最为接近。此外，使用本研究的扩增方法还能提升网络的抗干扰能力，使模型在精度方面优于基础扩增和真实数据集训练得到的网络。与此同时，实验还进一步验证了正样本扩增会对成熟模型的判断产生影响，且负样本扩增得到的水果病斑图像具有较强的迷惑性。

第五章 总结与展望

5.1 总结

受农业领域智能建设因数据规模和质量而掣肘的启发，本研究以水果图像作为切入点，分析其典型应用场景和现有的数据分布情况。确立了以水果病虫害的监测设备作为数据应用目标、以图像采集环境和病斑的性状表征作为数据扩增重点的研究课题。

从栽育端的角度出发，水果图像易受天气情况和生物因素的影响，但整体而言依旧是正样本。但由于上述干扰，病害监测设备可能会对传感器采集来的图片产生误判。因此除了确保安全性的基础扩增技术外，本研究还引入了雨雪雾等自然场景与昆虫干扰，来进行正样本扩增，以此提升模型的鲁棒性。而对于水果病斑图像，不同的病害存在相异的性状表征。根据水果病害的易感程度与种间表现一致性，本研究确立水果炭疽病为主要研究对象，从分割后融合、特征模拟和自动生成三条技术路线出发，实现水果病斑图像的负样本扩增。

通过分析水果炭疽病的性状表征，我们发现，其存在显著的传染性和周期性。即植株若被侵害，则在果实、叶片或茎蔓上都可能出现预兆。而不同分布下患病早期和晚期的病斑，也有着相异的图像特征。因此在扩增实现时，我们进一步对上述的研究方案进行细化区分。

对于分割后融合的扩增思路，首先利用图像分割技术，以病斑区域和感染程度为划分，处理生成典型病灶集。然后利用泊松融合，将真实斑块过渡平滑地插入到正常水果图像中，插入区域的坐标通过目标检测技术 YOLO-v3 确定。特征模拟技术主要是分析了炭疽病病灶的颜色、形状和纹理特征，结合农业领域专业知识，总结设计了病斑模拟扩增的实现方法，生成区域同样根据目标检测返回的 roi 坐标确定。而自动生成的扩增方式则是依赖生成对抗网络，基于 DCGAN 和 CycleGAN 模拟真实图像的数据分布，从而得到水果病斑图像。

本研究的创新之处在于：

- 1、从水果数据图像的应用场景分析，对栽育端传感器获取到的图片异常情况进行了总结，发现不良的拍摄情境会对监测终端产生负反馈。因此在通用扩增的基础上，提出了新的正样本扩增方法。既考虑到水果种植阶段各种自然条件的影响，还创新性地构建了昆虫抠图网络来实现昆虫干扰，解决了仅使用基础扩增安全性不足的问题，提高了受训网络对于对抗样本的鉴别能力。

2、进一步研究现有的数据集，得出水果图像数据缺失、分布严重失衡的现状。因此结合农业领域知识和人工智能技术，从炭疽病的性状表征切入，提出了针对领域特性的负样本扩增技术，对水果病斑周期性的形态特征以及极具传染性的分布特征做了模拟和复现。打破生物专业性的壁垒，以及通用扩增无法针对特定领域数据有效扩增的限制。

3、应用本研究的扩增方法得到的图像，不仅质量和分辨率较高，而且接近现实场景拍摄的图片。对比真实数据和基础扩增，在提高模型鲁棒性和精确度方面的综合表现优越。正样本扩增技术可以对成熟模型产生干扰，而负样本扩增的结果可以骗过训练良好的检测模型，真实度高。因此，本研究提出的扩增方法在扩增有效性和领域专业性上具有双重优势。

4、填补了公开数据集领域，种植端水果图像的空白，同时构建了针对炭疽病细化分类的数据图像。用户能够根据扩增需求和应用场景的不同，自由组合或移植调配提出的扩增方法。既缓解了训练数据有限的问题，也降低了智慧农业建设的成本，为病虫害监测系统完成及时告警和对策制定奠定了数据基石。

5.2 未来工作展望

虽然扩增结果显示本研究的扩增方法在各方面达到了综合最优，但依旧存在问题和改进的方向。

1、由于数据集本身数量和质量的限制，本研究用于扩增的图片并不是分类完善且高质量的图像数据，这就使得自动生成技术的效果不尽人意。例如对于水果果实的炭疽病图片，其在早期和晚期的图像数据集并不是非常多，从而难以使用基于 DCGAN 的自动扩增方法得到水果果实的病斑图片。因此对于此方向下后续的扩增研究，建议与果园或农业研究院达成深度合作，集中获取分类完善（果实和果叶、病害早期和晚期）的高分辨率真实图像，用于进一步的研究探索。

2、因为叶片边缘形状并不规则，且晚期炭疽病叶片上的病斑常常出现区域性的轮斑软腐，所以在实现针对晚期病叶的扩增时，病斑分割后融合技术和病斑模拟技术常常无法得到非常真实的效果。一是区分侵染的叶片边缘并不容易，二是大面积的黄化斑驳难以分割和复现，因此实现时只针对了完整的轮纹病斑。此外，扩增时对病斑形态的总结，也忽略了特殊状态下的情况。例如潮湿天气时，病斑中央会伴有红色胶质物；病害发展到后期，果皮会出现皱缩，形成干疤和泪痕等。因此如果期望扩增图像能更加符合真实的性状表征，可能还需要更为复杂的模拟技术或生成手段。与此同时，第三章也提到，种植端果叶图像的叶片检测模型训练耗时耗力，本研究使用的模型只能在背景简单的图像里得到

较好的检测结果。而对于含有诸多果叶的图像，则选择人为干预的方式，确定病斑区域坐标，这就降低了扩增的效率。

3、在昆虫扩增的实现上，本研究可以利用自动抠图技术将昆虫前景提取出来，再进行融合。但是训练抠图网络需要复杂的前期准备，执行过程也比较耗时。且自然界昆虫众多，对训练样本提出了较高的要求，模型无法保证抠图结果完全符合预期。因此可以考虑自定义背景为白色的昆虫图集，使用形态学分割后快速融合。

4、出于对扩增效率的考虑，本研究使用的融合方法较为简单，一定程度上会模糊前景的细节。对其进行改进时，可以使用更为复杂的融合技术。在牺牲一部分扩增速度的前提下，得到更完美的融合结果。

5、在对扩增结果进行真实性评估时，由于没有成对数据，因此除自动生成的扩增方法外，其余扩增方法得到的图像都是与人为选取的真实图片进行比较的。这就使得二者之间误差较大，计算得到的评估值意义不高。如果有条件的话，应该选取成对的图像，例如同一角度下对同一水果，拍摄雨雪雾场景和普通场景下的图像。然后对正常情况的原图像实现不同天气条件的扩增，再进行相应的真实度评估，这样得出的结果才更加有效可靠。

参考文献

- [1] ABDEL-BASSET M, MANOGARAN G, GAMAL A, et al. A novel intelligent medical decision support model based on soft computing and IoT[J]. IEEE Internet of Things Journal, 2019, 7(5): 4160–4170.
- [2] LIAN W, XUE T, LU Y, et al. Research on hierarchical data fusion of intelligent medical monitoring[J]. IEEE Access, 2019, 8: 38355–38367.
- [3] QIN F, ZENG W, LI L, et al. Construction of big data monitoring platform for teaching quality under intelligent education[C] // 2020 International Wireless Communications and Mobile Computing (IWCMC). 2020: 1594–1597.
- [4] CHEN L, CHEN P, LIN Z. Artificial intelligence in education: A review[J]. Ieee Access, 2020, 8: 75264–75278.
- [5] FAROOQ M S, RIAZ S, ABID A, et al. A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming[J]. IEEE Access, 2019, 7: 156237–156271.
- [6] MEKALA M S, VISWANATHAN P. A Survey: Smart agriculture IoT with cloud computing[C] // 2017 international conference on microelectronic devices, circuits and systems (ICMDCS). 2017: 1–7.
- [7] AMIN A B, DUBOIS G O, THUREL S, et al. Wireless Sensor Network and Irrigation System to Monitor Wheat Growth Under Drought Stress[C] // 2021 IEEE International Symposium on Circuits and Systems (ISCAS). 2021: 1–4.
- [8] RUSTIA D J A, LIN C E, CHUNG J-Y, et al. Application of an image and environmental sensor network for automated greenhouse insect pest monitoring[J]. Journal of Asia-Pacific Entomology, 2020, 23(1): 17–28.
- [9] 李海珍. 智慧农业在农作物病虫害预测预报中的应用 [J]. 农业工程技术, 2021, 41(21): 3.
- [10] VILLARRUBIA G, PAZ J F D, IGLESIA D H, et al. Combining multi-agent systems and wireless sensor networks for monitoring crop irrigation[J]. Sensors, 2017, 17(8): 1775.

- [11] MAHLEIN A-K. Plant disease detection by imaging sensors—parallels and specific demands for precision agriculture and plant phenotyping[J]. *Plant disease*, 2016, 100(2): 241–251.
- [12] JHURIA M, KUMAR A, BORSE R. Image processing for smart farming: Detection of disease and fruit grading[C] // 2013 IEEE second international conference on image information processing (ICIIP-2013). 2013 : 521–526.
- [13] AWATE A, DESHMANKAR D, AMRUTKAR G, et al. Fruit disease detection using color, texture analysis and ANN[C] // 2015 International Conference on Green Computing and Internet of Things (ICGCIoT). 2015 : 970–975.
- [14] JAIN A, PATEL H, NAGALAPATTI L, et al. Overview and importance of data quality for machine learning tasks[C] // Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020 : 3561–3562.
- [15] PEREZ L, WANG J. The effectiveness of data augmentation in image classification using deep learning[J]. *arXiv preprint arXiv:1712.04621*, 2017.
- [16] SHIJIE J, PING W, PEIYI J, et al. Research on data augmentation for image classification based on convolution neural networks[C] // 2017 Chinese automation congress (CAC). 2017 : 4165–4170.
- [17] MIKOŁAJCZYK A, GROCHOWSKI M. Data augmentation for improving deep learning in image classification problem[C] // 2018 international interdisciplinary PhD workshop (IIPhDW). 2018 : 117–122.
- [18] ZHONG Z, ZHENG L, KANG G, et al. Random erasing data augmentation[C] // Proceedings of the AAAI conference on artificial intelligence : Vol 34. 2020 : 13001–13008.
- [19] SHORTEN C, KHOSHGOFTAAR T M. A survey on image data augmentation for deep learning[J]. *Journal of big data*, 2019, 6(1): 1–48.
- [20] CUBUK E D, ZOPH B, MANE D, et al. Autoaugment: Learning augmentation policies from data[J]. *arXiv preprint arXiv:1805.09501*, 2018.

-
- [21] TAYLOR L, NITSCHKE G. Improving deep learning with generic data augmentation[C] // 2018 IEEE Symposium Series on Computational Intelligence (SSCI). 2018 : 1542 – 1547.
- [22] ENGSTROM L, TRAN B, TSIPRAS D, et al. A rotation and a translation suffice: Fooling cnns with simple transformations[J], 2018.
- [23] GOODFELLOW I J, SHLENS J, SZEGEDY C. Explaining and harnessing adversarial examples[J]. arXiv preprint arXiv:1412.6572, 2014.
- [24] LI S, CHEN Y, PENG Y, et al. Learning More Robust Features with Adversarial Training[J], 2018.
- [25] MOOSAVI-DEZFOOLI S-M, FAWZI A, FROSSARD P. Deepfool: a simple and accurate method to fool deep neural networks[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016 : 2574 – 2582.
- [26] XIE L, WANG J, WEI Z, et al. Disturblabel: Regularizing cnn on the loss layer[C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016 : 4753 – 4762.
- [27] GOODFELLOW I, POUGET-ABADIE J, MIRZA M, et al. Generative adversarial nets[J]. Advances in neural information processing systems, 2014, 27.
- [28] RADFORD A, METZ L, CHINTALA S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. arXiv preprint arXiv:1511.06434, 2015.
- [29] FRID-ADAR M, DIAMANT I, KLANG E, et al. GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification[J]. Neurocomputing, 2018, 321 : 321 – 331.
- [30] KARRAS T, AILA T, LAINE S, et al. Progressive growing of gans for improved quality, stability, and variation[J]. arXiv preprint arXiv:1710.10196, 2017.
- [31] MIRZA M, OSINDERO S. Conditional generative adversarial nets[J]. arXiv preprint arXiv:1411.1784, 2014.

- [32] GATYS L A, ECKER A S, BETHGE M. Image style transfer using convolutional neural networks[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016 : 2414–2423.
- [33] JACKSON P T, ABARGHOUEI A A, BONNER S, et al. Style augmentation: data augmentation via style randomization.[C] // CVPR Workshops : Vol 6. 2019 : 10–11.
- [34] TOBIN J, FONG R, RAY A, et al. Domain randomization for transferring deep neural networks from simulation to the real world[C] // 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). 2017 : 23–30.
- [35] 崔艳丽, 程鹏飞, 董晓志, et al. 温室植物病害的图像处理及特征值提取方法的研究——基于色度的特征值提取研究 [J]. 农业工程学报, 2010.
- [36] FEDUS W, ROSCA M, LAKSHMINARAYANAN B, et al. Many paths to equilibrium: GANs do not need to decrease a divergence at every step[J]. arXiv preprint arXiv:1710.08446, 2017.
- [37] SALIMANS T, GOODFELLOW I, ZAREMBA W, et al. Improved techniques for training gans[J]. Advances in neural information processing systems, 2016, 29.
- [38] KYPRIANIDIS J E, COLLOMOSSE J, WANG T, et al. State of the” art” : A taxonomy of artistic stylization techniques for images and video[J]. IEEE transactions on visualization and computer graphics, 2012, 19(5) : 866–885.
- [39] DONG C, LOY C C, HE K, et al. Image super-resolution using deep convolutional networks[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 38(2) : 295–307.
- [40] JOHNSON J, ALAHI A, FEI-FEI L. Perceptual losses for real-time style transfer and super-resolution[C] // European conference on computer vision. 2016 : 694–711.
- [41] MECHREZ R, TALMI I, ZELNIK-MANOR L. The contextual loss for image transformation with non-aligned data[C] // Proceedings of the European conference on computer vision (ECCV). 2018 : 768–783.

- [42] ZHU J-Y, PARK T, ISOLA P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks[C] // Proceedings of the IEEE international conference on computer vision. 2017 : 2223 – 2232.
- [43] LI J, ZHANG J, MAYBANK S J, et al. Bridging composite and real: towards end-to-end deep image matting[J]. International Journal of Computer Vision, 2022 : 1 – 21.
- [44] LI J, ZHANG J, TAO D. Deep Automatic Natural Image Matting[J], 2021.
- [45] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016 : 770 – 778.
- [46] HU J, SHEN L, SUN G. Squeeze-and-excitation networks[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2018 : 7132 – 7141.
- [47] ZHAO H, SHI J, QI X, et al. Pyramid scene parsing network[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2017 : 2881 – 2890.
- [48] 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. 2015 : 234 – 241.
- [49] ZHONG Z, BLUM R S. Region-Based Image Fusion Scheme For Concealed Weapon Detection[J], 2003.
- [50] TOET A. A morphological pyramidal image decomposition[J]. Pattern Recognition Letters, 1989, 9(4) : 255 – 261.
- [51] DUBEY S R, JALAL A S. Adapted approach for fruit disease identification using images[G] // Image processing: Concepts, methodologies, tools, and applications. [S.l.] : IGI Global, 2013 : 1395 – 1409.
- [52] MODI H, PATEL M, PATEL M, et al. Implementation of Algorithm to Detect the Diseases in Fruit Using Image Processing Technique[J]. International Journal of Applied Engineering Research, 2019, 14(9) : 2093 – 2106.

-
- [53] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580–587.
- [54] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779–788.

简历与科研成果

基本情况 顾雪晴，女，汉族，1998年3月出生，江苏省扬中市人。

教育背景

2020.9 ~ 2022.6 南京大学 软件学院 硕士

2016.9 ~ 2020.6 南京邮电大学 计算机、软件、网络空间安全学院 本科

读研期间的成果（包括发表的论文及参与的专利）

致 谢

时光荏苒，转眼两年的研究生生活就要结束了。值此毕业之际，在这里对两年里给予过我帮助的老师同学、家人朋友致以由衷的谢意。

首先在这里，要感谢我的导师陈振宇老师，在两年里一直用严谨的学术态度和开阔的知识眼界帮助我不断提高、敦促我努力进步。从论文开题到完稿，不厌其烦地帮助我修改、打磨。虽然他经常需要出差开会，还有繁重的教学任务，但他依旧会利用有限的时间为研究生同学们组织研讨会、讲解论文撰写的细节，组织专家学者为我们进行答辩。他看到和我们各自研究领域相关的文章会及时推送给我们学习，小到请假签字和系统审核这样琐碎的事宜，也会热情地帮助我们完成。感谢陈老师，作为我的领路人指引我不断前行。

除此之外，我还要感谢实验室的学长学姐们，尤其是刘佳玮学姐，在学习和生活等各个方面都给予了我莫大的帮助和支持。实验室的其他老师同学，也都是我学习的榜样，帮助我不断成长。还要感谢我研究生阶段的三位舍友，让我感受到了家一般的友善和温暖，排解了我的忧愁和焦虑，为我带来了许多欢笑和难忘的记忆。

与此同时，我想特别感谢我的父母，作为我的精神后盾和心灵慰藉默默地支持着我，用暖心的话语和日常的关心纾解了我的烦恼。同样我的朋友们也在我迷茫和困惑时，不停送来问候，再一次向我最亲爱的人们表示由衷的感谢。

最后，我还要感谢论文评审和答辩的各位专家和老师们，对于我研究的批评指正，使我受益匪浅。感谢特殊时期工作的防疫人员，为我们这些普通人的安全和生活默默付出、不计回报。