



东北林业大学

# 毕 业 论 文

基于深度学习的小麦病害分类  
模型特征分析

学生姓名:	马群
专业班级:	信息与计算科学 2015 级 1 班
指导教师:	史春妹 讲师
学 院:	理学院

2019 年 6 月

# 基于深度学习的小麦病害分类模型特征分析

马群

## 摘要

为了实时监控小麦病害情况并及时采取防治措施,找出一种适合分类处理小麦常见病害的神经网络模型。本文首先以小麦病害图片资料为基础,经过挑选、裁剪后对图像进行序列化操作形成数据集,然后通过构建的多个深度神经网络模型进行学习,在学习过程中使用随机梯度下降法对学习过程进行控制。然后在此基础上改进表现较差的神经网络模型,在两个维度上形成对比,以此寻找最适合处理小麦病害分类问题的深度神经网络模型。实验结果表明,在参与实验的多个神经网络结构中,以卷积神经网络(convolutional neural networks, CNN)表现最为出众,整体识别准确率达99%, (交叉)验证准确率在(validation accuracy) 75%左右。这表明使用卷积神经网络进行小麦常见病害的识别是有效且可行的,理论上为小麦病害的实时分析提供了有效且强力的分析手段。

**关键字** 小麦病害; 卷积神经网络; 特征分析; 深度学习

# Feature analysis of wheat disease classification model based on deep learning

## Abstract

This is the abstract in English. This is the abstract in English. This is the abstract in English. This is the abstract in English. This is the abstract in English.

**Keywords** Wheat disease; Convolutional Neural Network; Convolutional deep belief networks

# 目录

<b>1</b>	<b>前言</b>	<b>6</b>
1.1	研究背景及意义	6
1.2	农作物病害识别的国内外研究现状	6
1.2.1	国内研究现状	6
1.2.2	国外研究现状	7
1.3	深度学习与神经网络的发展	8
1.4	论文结构和研究内容	9
1.4.1	主要研究内容	9
1.4.2	论文结构	9
<b>2</b>	<b>卷积网络</b>	<b>10</b>
2.1	卷积	10
2.2	卷积网络简介	11
2.3	卷积网络的特点	11
2.4	卷积网络的结构	12
<b>3</b>	<b>基于卷积网络的小麦病害图像识别</b>	<b>13</b>
3.1	本文卷积网络结构	13
3.2	LeNet-5 和 AlexNet 结构简介	14
3.3	实验步骤及操作	15
3.3.1	数据的获取及处理	15
3.3.2	卷积网络模型的建立	16
<b>4</b>	<b>实验结果分析</b>	<b>18</b>

# 1 前言

## 1.1 研究背景及意义

小麦是我国三大粮食作物之一，其种植区域主要分布在长江以北的大部分地区，种植面积位居第二，仅次于水稻。病害是影响小麦等农作物产量和质量的首要问题。全世界范围内小麦病害大约有 200 多种，每年造成的产量损失约为 15%~20%。在我国危害较严重的有三十多种，其中以白粉病（Wheat powdery mildew）、锈病（Puccinia striiformis West，包括条锈、秆锈、叶锈）、叶枯病（Wheat leaf blotch）、赤霉病等在我国主要小麦产地分布较广，为害较为严重 [1]。

传统形式的小麦病虫害是依靠经验识别、人工喷施农药进行防治的。在大面积的小麦种植模式下，人工防治不仅需要大量的人力物力，而且在喷施农药的过程中极为不安全。更进一步地讲，即使是在技术人员的帮助下进行病害防治，也不能做到实时监控小麦病害情况、及时实施防治工作。因此如何做到实时监控并报告小麦病害情况成为了现代农业生产中的重要目标。

自 1980 年机器学习被称为一个独立的方向开始，经过一代又一代人的努力，诞生了大量经典的分类算法。其中以朴素贝叶斯（Naive Bayes, NB）、Logistic 回归（Logistic Regression, LR）、决策树（Decision Tree, DT）、支持向量机（Support Vector Machine, SVM）等浅层机器学习模型最常用，它们的出现为小麦病害的自动识别提供了有力的理论支持。但是这些经典分类算法的图像特征提取策略是基于先验知识制定的，效率不高且不适合应用在大规模特征提取方面 [2]。近年来，由 AlphaGo 带来的人工智能热潮使得深度学习一词出现在公共视野里。深度学习是机器学习领域中一种以人工神经网络为架构，通过数据进行表征学习的算法。

如此一来，将现代的深度学习技术与传统图像处理相结合的方式成为了农作物病害识别的新手段。深度学习在图像处理领域的优势不仅仅在于能够准确地提取特征，还在于它通过处理大量图像数据时能不断地自我学习并取得更高的准确率。

在以深度学习为主要手段的图像处理过程中，卷积神经网络模型对图像的处理由很高的优势，故本文研究的内容会以卷积神经网络为基础，通过建立不同结构的神经网络模型并加以比较，找出最适合小麦病害识别的深度网络模型。

## 1.2 农作物病害识别的国内外研究现状

### 1.2.1 国内研究现状

国内在农作物病害识别方面的研究起步较晚，多数是在国外相关研究启发下开始的。近年来机器学习领域的研究呈现井喷式发展，随之带动了农作物病害识别的研究，并取得了许多卓有成效的识别技术。2009 年，王守志等实现了基于核 K-均值聚类方法的玉米叶部病害识别，实验涉及的 4 种玉米病害识别准确率达 82.5% [3]；2011 年，陈丽等提出了一种基于图像处理技术和概率神经网络技术的玉米叶部病害识别方法，利用遗传算法

优化选择出 4 个分类能力强的分类特征，由概率网络（PNN）分类器识别病害，平均准确率为 90.4%，高于 BP 神经网络 [4]；2012 年，张建华等提出了一种基于粗糙集和 BP 神经网络的棉花病害识别方法，该方法能准确识别 4 种棉花病害，平均识别准确率达到 92.72% [5]；王树文等利用基本图像处理方法对黄瓜叶部病害图像进行处理，综合运用二次分割、形态学滤波得到病斑区域。提取三种特征并采用 BP 算法训练多层前向神经网络对黄瓜病害进行分类，该系统的平均识别准确率为 95.31% [6]；2013 年，张飞云等利用量子神经网络进行玉米病害分类识别，对玉米灰斑病、玉米普通锈病和玉米小斑病的识别准确率达 92.5%、97.5% 和 92.5%，高于误差反向传播神经网络法的识别率（分别为 90.0%、90.0% 和 92.5%），可用于玉米叶部病害识别 [7]；2014 年，余秀丽等设计并实现了一种基于 SVM（Support Vector Machine，支持向量机）的小麦叶部常见病害识别方法。随机试验结果表明，利用所提取的特征可以有效地实现小麦叶部常见病害的识别，基于形状特征综合识别率可达 99.33%，利用支持向量机算法进行小麦病害叶片识别是有效的、可行的 [8]；2018 年，张航等提出了一种基于卷积神经网络的小麦病害识别方法，利用随机梯度下降法对一个具有五层结构的网络模型行学习过程控制，其综合识别率可达 99% [9]。

### 1.2.2 国外研究现状

国外关于农作物病害识别的研究起步较早，在上世纪九十年代已有多种识别方法被提出。在早期的处理方法大多是对病害图像进行前期处理，包括图像分割、滤波、简单分类等技术，或是辅助以人工分析，然后再结合农作物病理学知识对处理后的数据进行分类识别。1997 年，Tucker 与 Chakraborty 提供了一种专用软件，可以检测向日葵和燕麦叶片上的病变，提供病变数量和类型以及病害严重程度的数据，但是该软件的分类准确率达不到预想程度 [10]；1999 年，Sasaki 等使用光谱反射特性和滤光片图像构建了一个植物病害自动诊断系统，发现 500、600 和 650nm 的滤光镜图像比其他滤光镜图像更适合于识别，在创建鉴别参数对健康叶片和患病叶片进行分类后，达到了 5% 或更小的误差率 [11]。

由于图像处理技术的限制，早期的农作物病害识别技术并不能满足人们的要求。近年来，随着图像处理和人工智能的发展，机器学习应用到农作物病害识别领域的例子越来越多。2008 年，Phadikar 和 Sil 等介绍了一种基于水稻植株感染图像的水稻病害检测系统，该系统使用图像生长、分割技术处理感染部分，然后使用 SOM 神经网络将四种叶子的感染部分进行分类处理，实验结果令人满意 [12]；2014 年，Mathura 与 Uttar Pradesh 介绍了一种基于邻近像素点像素强度的改进和差直方图，与梯度滤波器相配合使用可以对苹果病害的检测达到 99% 的准确率 [13]；2016 年，Sladojevic 等实现了最新一代的卷积神经网络（CNN），该模型能够识别健康叶片中的 13 种不同类型的植物病害，实验结果的平均精确度达到了 96.3% [14]；2017 年，Fuentes 和 Yoon 等提出了一种深度学习方法来检测番茄病害，该方法含有三种神经元架构：基于区域的快速卷积网络（Faster R-CNN）、基于区域的全卷积网络（R-FCN）和单发多核检测器（SSD），实验结果表明，该系统能有效识别九种不同类型的病虫害 [15]；

### 1.3 深度学习与神经网络的发展

深度学习是机器学习领域中一种以人工神经网络为架构，根据数据进行表征学习的算法。它的前身是人工神经网络，基本特点是模仿人脑神经元处理和传递信息的方式，本质是解决贡献度的分配问题。

对深度学习的研究最早可以追溯到 1943 年，神经科学家麦卡洛克 (W.S.McCulloch) 和数学家皮兹 (W.Pitts) 建立了一个基于神经网络和数学的模型，称为 MCP 模型。MCP 模型是按照大脑神经元的结构和工作原理构造出来的一个简化了的、抽象的模型，也就是所谓的“模拟大脑”，深度学习和人工神经网络的大门由此开启 [16]。1958 年，计算机科学家罗森布拉特 (Rosenblatt) 提出了一个由两层神经元组成的神经网络，称之为“感知器” (Perceptrons)。这也是第一次将 MCP 模型用于机器学习分类。“感知器”算法使用 MCP 模型对输入的多维数据分为两类，且能够使用梯度下降法在训练过程中自动学习并更新权值。1962 年，Novikoff 定理证明该方法具有收敛性，理论与实践效果引起第一次神经网络热潮。

神经网络之父 Geoffery Hinton 于 1986 年发明了适用于多层感知器 (MLP) 的 BP (Backpropagation) 算法，该方法采用 Sigmoid 进行非线性映射，有效解决了非线性分类和学习的问题。Sigmoid 函数的输出在 (0,1) 内，单调连续且易于求导，非常适合用作输出层。但是它也有软饱和性的缺点，一旦落入饱和区， $f'(x)$  就会变得接近于 0，很容易产生梯度消失。90 年代中期，支持向量机 (Support Vector Machine, SVM) 算法诞生，随之各种浅层机器学习模型被提出。SVM 是一种有监督的学习模型，通常应用于模式识别，分类以及回归分析等。支持向量机以统计学为基础，和神经网络有明显的差异，支持向量机等算法的提出再次使深度学习的发展受到阻碍。2006 年 Geoffrey Hinton 和 Ruslan Salakhutdinov 提出了深层网络训练中梯度消失的解决方案：无监督预训练下对权值进行初始化，有监督训练权值微调 [17]。2011 年，ReLU 激活函数被提出，该激活函数能够有效的抑制梯度消失问题。

2011 年，微软首次将深度学习应用在语音识别领域，取得了重大突破。微软研究院和 Google 的语音识别研究人员先后采用卷积神经网络使语音识别错误率降低 20%~30%，是语音识别领域十多年来的突破性进展。2012 年，Hinton 的课题组为了证明深度学习的能力，首次参加 ImageNet 图像识别比赛，其构建的 CNN 网络 AlexNet 在 ImageNet 评测上将错误率从 26% 降低到 15%，在第二名 (SVM 方法) 面前以压倒性的优势夺得冠军。

2016 年 3 月，由谷歌 (Google) 公司开发的 AlphaGo (基于深度学习算法) 与围棋世界冠军、职业九段棋手李世石进行围棋大战，以 4 比 1 的总比分获胜；2016 年末至 2017 年初，该程序在中国棋类网站上注册名为“大师” (Master) 的帐号并与中日韩数十位围棋高手进行快棋对决，连赢 60 局，无一败绩；2017 年 5 月，在乌镇围棋峰会上，AlphaGo 与世界排名第一的围棋冠军柯洁对战，以 3 比 0 的最终比分获胜。围棋界公认阿尔法围棋的能力已经超过人类职业围棋顶尖水平。

深度学习虽然已经成为众多科研领域的热门研究内容，但是目前还处于发展阶段，不管是理论方面还是实践方面都还有许多问题待解决，不过由于我们处在了一个“大数据”



时代,以及计算机处理能力的大大提升,新理论的验证周期会大大缩短,人工智能的发展必然会很大程度地改变这个世界。

## 1.4 论文结构和研究内容

### 1.4.1 主要研究内容

本文以提高卷积网络在小麦病害图像识别的准确率为目标,提出了具有 12 层结构的卷积网络。针对前期实验的不足之处对网络结构加以更改,在得到了训练过程中表现良好的网络模型的模型之后,将其与经典卷积网络模型 LeNet-5、AlexNet 的实验结果进行比较,分析本文模型的长处与不足,验证该模型在小麦病害图像识别中的有效性。

### 1.4.2 论文结构

第一章是引言,首先介绍了小麦病害图像识别的研究背景和意义,然后综述了深度学习与神经网络的发展历史及研究现状,最后介绍了本文的主要研究内容和章节安排。

第二章主要介绍了卷积的基础理论知识,然后简单介绍了卷积神经网络的发展及其特点和主要结构层次。

第三章搭建了一个结构上较新颖的卷积神经网络模型,并针对小麦病害识别加以修改,使其尽量符合预期要求。接下来主要介绍了实验内容,首先是数据源的获取和处理,然后是不同模型在相同数据源下的训练过程。

第四章是针对该实验结果的分析以及对不同模型的结果加以对比,说明本文新模型的优势。然后对数据集的质量进行了解释,以及数据集对训练结果的影响。最后是对本文的总结分析,以及对老师指导的感谢。

## 2 卷积网络

### 2.1 卷积

卷积（Convolution）是分析数学中的一种积分变换的方法，是其中一个函数反转并平移后与另一个函数乘积的积分。设  $f$  与  $g$  是  $\mathbf{R}_1$  上的两个可积函数，做积分后的新函数就成为函数  $f$  与  $g$  的卷积：

$$f * g = \int_{\tau \in A} f(\tau)g(t - \tau)d\tau$$

卷积也经常应用在图像处理中，因为图像是一个二维结构，所以适合用二维卷积对图像做特征提取等操作。给定一个图像  $\mathbf{I} \in \mathbb{R}^{M \times N}$ ，和滤波器  $\mathbf{K} \in \mathbb{R}^{M \times N}$ ，一般  $m \ll M$ ， $n \ll N$ ，其卷积为：

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n) \quad (2-1)$$

下式为二维卷积示例：

$$\begin{pmatrix} 2 & 0 & 1 & 1 & 1 \\ -1 & 0 & -3 & 0 & 1 \\ 2 & 1 & 1 & -1 & 0 \\ 0 & -1 & 1 & 2 & 1 \\ 1 & 2 & 1 & 1 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} = \begin{pmatrix} -1 & 1 & -1 \\ 2 & 2 & 4 \\ -1 & 0 & 0 \end{pmatrix} \quad (2-2)$$

卷积是可交换的，我们可以等价地把 (2-1) 式写作：

$$(2-1) \Leftrightarrow S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n) \quad (2-3)$$

(2-3) 式也被称为  $I$  和  $K$  的互相关（Cross-Correlation），它是一个衡量两个序列相关性的函数。通过 (2-1)、(2-3) 两式对比可知，卷积和互相关的区别仅仅在于卷积核是否进行了翻转（Flip）。许多机器学习库中实现的“卷积运算”其实是互相关函数，之所以称之为卷积运算，这是因为卷积核的特征提取能力与其是否翻转无关。在训练过程中，学习算法会在核合适的位置自动更新为恰当的值，所以一个基于互相关学习算法所学习到的核，是使用卷积运算所学到的核的翻转。

数字图像是一个二维的离散信号，对数字图像做卷积操作其实是在图像上滑动一个卷积核（即滤波器），将图像点上的像素值与卷积核对应位置的值相乘，然后将相乘后所有的值相加，作为特征图像上对应位置的像素值。图 2-1 展示了 (2-2) 中反转前后的卷积核在小麦锈病图像中特征提取效果的对比图。

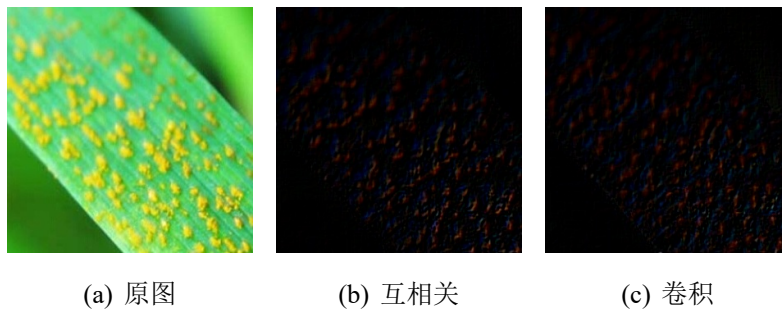


图 2-1 卷积核的翻转对特征提取的影响

## 2.2 卷积网络简介

卷积网络 (Convolutional Network), 也叫卷积神经网络 (Convolutional Neural Network, CNN), 是一种具有局部连接、权重共享等特性的深层前馈神经网络, 专门用来处理具有类似网格结构的数据的神经网络。

1962 年, Hubel 和 Wiesel 通过对猫脑视觉皮层的研究, 首次提出了一种新的概念——“感受野” (Receptive Field), 对后来人工智能网络的发展起了很大的推动作用 [18]。感受野是受生物学上感受野的机制而提出, 在生物学上描述的是神经系统的一些神经元的特性。而在人工神经网络中, 感受野指的是指的是卷积神经网络层输出的特征向量上的像素点对应的输入图像上的区域, 通俗地讲, 就是特征向量上的一个点对应的输入图像上的区域。1980 年, Fukushima [19] 基于生物神经学的感受野理论提出了神经认知机和权重共享的卷积神经层, 这被视为卷积神经网络的雏形。1989 年, LeCun[20] 将反向传播算法与权值共享的卷积神经层相结合, 发明了卷积神经网络, 并首次将卷积神经网络成功地应用到美国邮局的手写字符识别程序中。1998 年, LeCun[21] 提出了卷积神经网络的经典网络模型 LeNet-5, 它是第一个成功应用于数字识别问题的卷积神经网络。

## 2.3 卷积网络的特点

卷积运算通过三个重要的思想帮助改进深度学习系统: 稀疏交互 (Sparse Interactions)、参数共享 (Parameter Shareing) 和等变表示 (Equivariant Representations)。

稀疏交互的本质是对全连接的规避, 这是通过使核的大小远小于输入的大小来达到的。当处理一张图像时, 输入的图像可能包含几十万个像素点, 但是我们可以使用只有数十个像素点的卷积核来达到提取特征的目的。这意味着我们不仅可以存储更少的参数, 而且还大大减少了计算量, 提高了统计效率。稀疏交互的图形化表示如图 2-2 所示。

参数共享是指在一个模型的多个函数中使用相同的参数。在传统的神经网络中, 当计算一层的输出时, 权重矩阵的每一个元素只使用一次, 当它乘以输入的一个元素之后就再也用不到了。在卷积神经网络中, 核的每一个元素都作用在输入的每一个位置上。参数共享保证了我们只需要学习一个参数集合, 而不是每一个位置都需要学习一个新的集合。因此, 卷积在存储需求和统计效率方面极大地优于稠密矩阵的乘法运算。

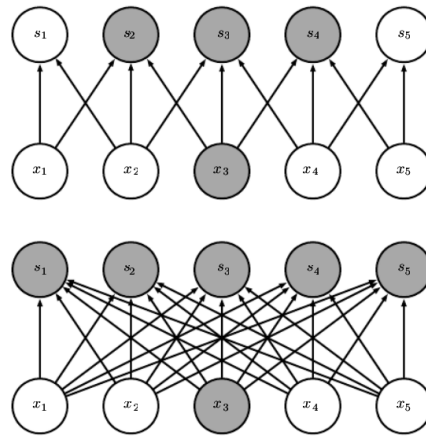


图 2-2 稀疏交互

在处理图像数据时，卷积产生了一个二维映射来表明某些特征在输入中出现的位置。如果我们移动输入中的图像，它的表示也会在输出中移动同样的量，这种性质就叫做等变表示。对于卷积来说，参数共享的特殊形式使得神经网络层具有对平移等变的性质。如果一个函数满足输入改变，输出也以同样的方式改变这一性质，我们就说它是等变的。特别的，如果函数  $f(x)$  与  $g(x)$  满足  $f(g(x)) = g(f(x))$ ，我们就说  $f(x)$  对于变换  $g$  具有等变性。对于卷积来说，如果令  $g$  是输入的任意平移函数，那么卷积函数对于  $g$  具有等变性。

## 2.4 卷积网络的结构

卷积网络一般由数个卷积层（Convolutional layer）、池化层（Pooling layer）、全连接层及输出层交叉堆叠而成，卷积层和池化层一般会取多个，采用卷积层和池化层交替堆叠的模式。卷积层并行地计算多个卷积产生一组线性激活响应，每一个线性激活响应将会通过一个非线性的激活函数，例如整流线性激活函数。然后使用池化层来进一步调整这一层的输出。最后添加 0 2 个全连接层。目前，整个网络结构趋向于使用更小的卷积核（比如 11 和 33）以及更深的网络结构。此外，由于卷积的操作性越来越灵活，池化层的作用也变得越来越小，因此目前比较流行的卷积网络中池化层的比例也越来越低，趋向于全卷积网络。

### 3 基于卷积网络的小麦病害图像识别

#### 3.1 本文卷积网络结构

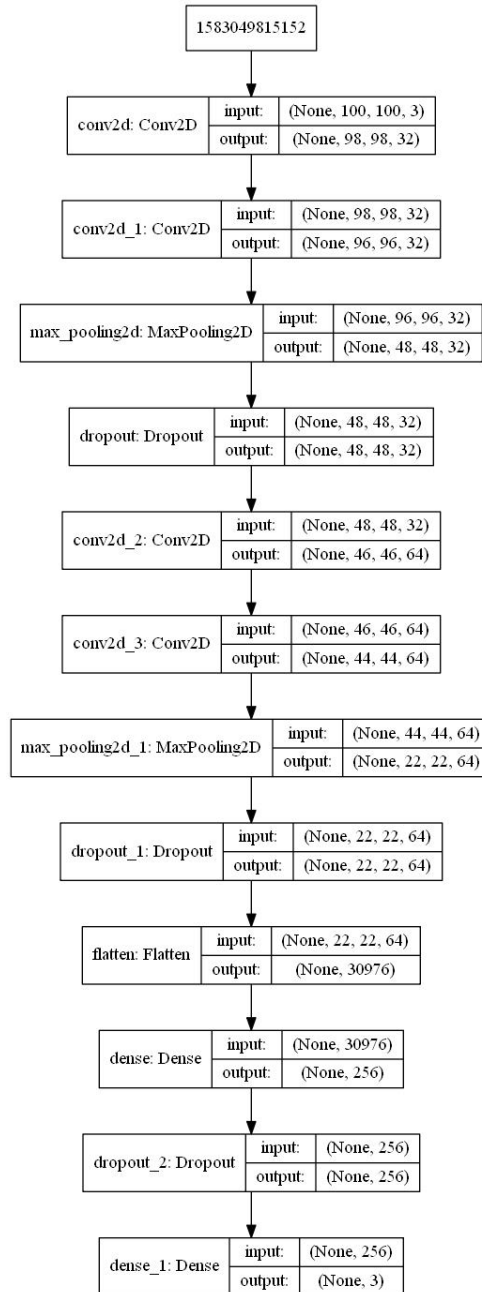


图 3-1 本文网络结构

第一层和第二层为卷积层，它们的过滤器大小均为  $3 \times 3$ ，深度为 32，不使用全零填充，步长为 1。第一层接受的输入层大小为  $100 \times 100 \times 3$ ，所以输出的尺寸为  $100 + 3 - 1 = 98$ ，深度为 6。这一层共有  $3 \times 3 \times 1 \times 32 + 32 = 320$  个参数，其中 32 个为偏置。下一层节点矩阵有  $98 \times 98 \times 32 = 307328$  个节点，每个节点和  $3 \times 3 = 9$  个当前层节点相连，所以本卷积层共有  $307328 \times (9 + 1) = 3073280$  个连接。同理，第二层中共有  $96 \times 96 \times 32 \times (3 \times 3 + 1) = 2949120$

个连接。

第三层为最大池化层，是一个  $96 \times 96 \times 32$  的节点矩阵。在本层中使用的过滤器大小为  $2 \times 2$ ，步长为 2。所以本层的输出大小为  $48 \times 48 \times 16$ 。

第四层为 Dropout 层，Dropout 层是 Hinton[22] 为了防止过拟合、提高神经网络的性能而提出的。该层会随机丢弃一些神经元，dropout rate 为 0.25。

第五至八层结构与前四层相似，不过为了提取更深的特征，将深度增加为 64 层。

第九层 Flatten 操作将输入  $22 \times 22 \times 64$  的张量展平为 1 维  $1 \times 30976$  的张量，为下一层的全连接操作做准备。

第十层的全连接层（Dense）使用 ReLU（Rectified Linear Unit, ReLU）作为激活函数。ReLU 即线性整流函数，指代数学中的斜坡函数： $f(x) = \max(0, x)$ 。该层输入大小为 30976，输出节点个数为 256 个，总共参数为  $30976 \times 256 + 256 = 7930112$  个。

第十一层仍为 Dropout 层，随机丢弃率为 0.5。

第十二层为全连接层，该层使用 Softmax 函数作为激活函数，该函数能将一个含任意实数的 K 维向量  $\mathbf{Z}$  “压缩”到另一个 K 维实向量  $\sigma(\mathbf{Z})$  中，并使得每一个元素的值域均在  $(0, 1)$  之间，并且所有原色的和为 1。该函数的形式通常由下式给出：

$$\sigma(\mathbf{Z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad \text{for } j = 1, \dots, K$$

该层输入大小为  $256 \times 1$  的张量，输出大小为 3，即小麦病害类别个数。

### 3.2 LeNet-5 和 AlexNet 结构简介

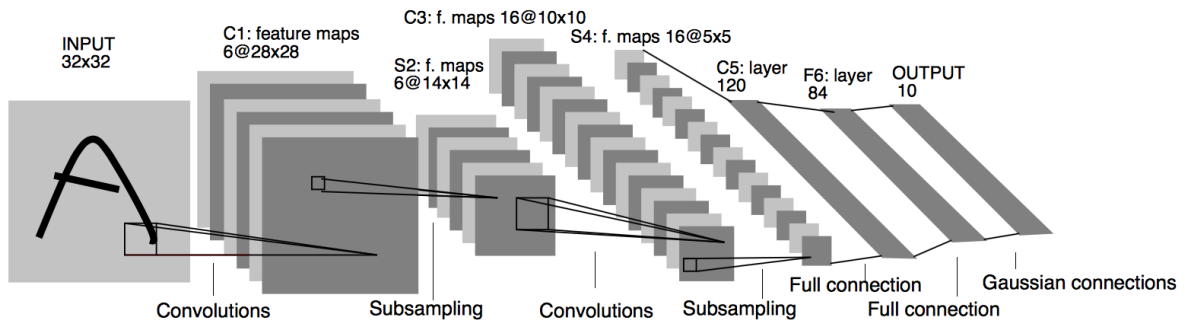


图 3-2 LeNet-5 网络结构

第一层输入的是原始像素，LeNet-5 模型接受的输入层大小为  $32 \times 32 \times 1$ 。第一个卷积层过滤器尺寸为  $5 \times 5$ ，深度为 6，不使用全零填充，步长为 1，输出尺寸为  $32 - 5 + 1 = 28$ ，深度为 6。这层共有  $5 \times 5 \times 1 \times 6 + 6 = 156$  个参数，其中 6 个为偏置。下一层的节点矩阵有  $28 \times 28 \times 6 = 4704$  个节点，每个节点和  $5 \times 5 = 25$  个节点相连，所以本层共有  $4704 \times (25 + 1) = 122304$  个连接。

第二层池化层的输入是  $28 \times 28 \times 6$  的节点矩阵。该层的过滤器大小为  $2 \times 2$ ，步长也为 2。所以本层输出大小为  $14 \times 14 \times 6$ 。第三层是输入大小为  $14 \times 14 \times 6$  的卷积层，卷

积核大小为  $2 \times 2$ ，步长为 2。第四层是输入大小为  $10 \times 10 \times 6$  的卷积层，过滤器大小为  $2 \times 2$ ，步长为 2。第五至七层均为全连接层，最后一层的输出为 10 个预测值。

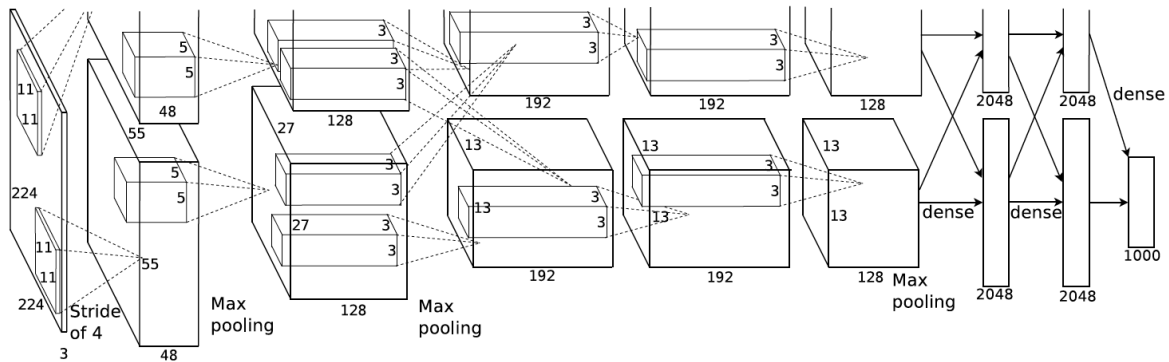


图 3-3 AlexNet 网络结构

AlexNet 的输入图片大小为  $224 \times 224 \times 3$ ，第一个卷积层使用 96 个较大的卷积核 ( $11 \times 11$ ) 步长为 4；接着是 LRN 层；在之后是一个过滤器为  $3 \times 3$  的最大池化层，步长为 2。接下来的卷积层中使用的卷积核都是  $3 \times 3$  或  $5 \times 5$  的大小，并且步长均为 1。最大池化层依然是  $3 \times 3$ ，步长为 2。最后三个相连的全连接层，最后输出 1000 个预测结果。

### 3.3 实验步骤及操作

#### 3.3.1 数据的获取及处理

实验所用的图片均来自网络。首先利用 Python 爬虫分别在百度图片和谷歌图片上爬取小麦的白粉病、叶枯病、锈病图片，保存至本地文件夹并做好备份。手工去除不清晰、含水印等劣质图片后，使用 Python 脚本将原始图片切割为多份  $300px \times 300px$  的图片。然后将这些图片按 3:1:1 的比例随机分为 train、test、evaluate 三个数据集，使用 Python 将全部图片的尺寸缩小为  $100px \times 100px$  后转为 numpy 数组，最后将这些数组序列化为 \*.npy 文件，这样的做法为重复实验省去了磁盘 IO 和图片处理操作的时间。数据集的样本和比例分布如图 3-4 所示：

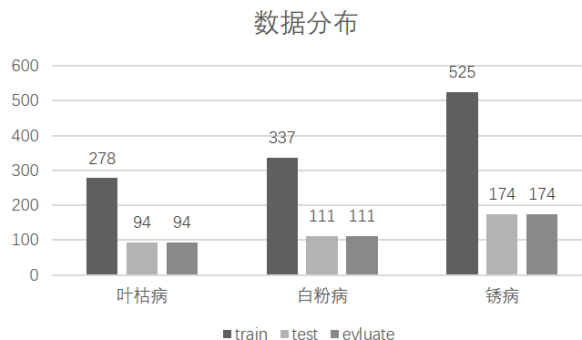


图 3-4 实验数据分布



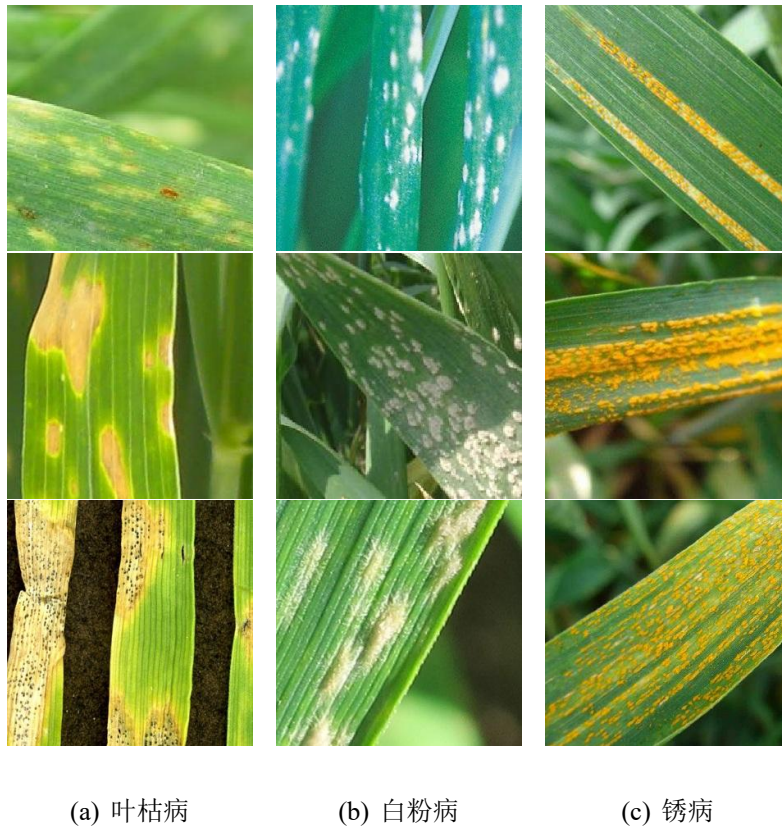


图 3-5 数据样本

### 3.3.2 卷积网络模型的建立

在机器学习异常火热的今天，越来越多的机器学习库可供大家学习。本文使用最受欢迎的深度学习框架 *TensorFlow* 搭建卷积网络模型，具体环境为：

- 操作系统：Windows 10 x64 Version 1809
- 处理器：Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz
- 显卡：Geforce GTX 950M
- 软件环境：
  - Python 3.6.8 64-bit
  - tensorflow-gpu 1.12.0
  - keras 2.1.6-tf
  - tensorboard 1.12.1

在 TensorFlow 中已经集成了 Keras，它是使用 Python 编写的高级神经网络 API，能够以 TensorFlow 作为后端运行。实验中为快速搭建卷积网络，使用 Keras 可节省大量时间。以下即为核心代码：



```
model = Sequential()
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(100, 100, 3)))
model.add(Conv2D(32, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(256, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(3, activation='softmax'))
sgd = SGD(lr=0.0001, decay=1e-6, momentum=0.01, nesterov=True)
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
```

为了便于实验后的数据分析,需要在代码中加入回调函数 *keras.callbacks.TensorBoard*, 设置了必要的参数后, 程序在执行时便会在指定的文件夹写入运行过程的日志文件以供分析。

## 4 实验结果分析

执行  $tensorboard -logdir = [logdir]$  命令后,

## 参考文献

- [1] 中国作物种质信息网. 小麦病害. Website. <http://www.cgris.net/kp/小麦病害.htm>.
- [2] 林中琦. 基于卷积神经网络的小麦叶部病害图像识别研究. 山东农业大学, 2018.
- [3] 王守志 [1, 何东健, 李文, and 王艳春. 基于核 k-均值聚类算法的植物叶部病害识别. 农业机械学报, 40(3):152–155, 2009.
- [4] 陈丽 and 王兰英. 概率神经网络在玉米叶部病害识别中的应用. PhD thesis, 2011.
- [5] 张建华, 祁力钧, 冀荣华, 王虎, 黄士凯, and 王沛. 基于粗糙集和 bp 神经网络的棉花病害识别. 农业工程学报, 28(7):161–167, 2012.
- [6] 王树文 and 张长利. 基于图像处理技术的黄瓜叶片病害识别诊断系统研究. PhD thesis, 2012.
- [7] 张飞云. 基于量子神经网络和组合特征参数的玉米叶部病害识别. 南方农业学报, 44(8):1286–1290, 2013.
- [8] 余秀丽, 徐超, 王丹丹, 张卫园, 屈卫锋, and 宋怀波. 基于 SVM 的小麦叶部病害识别方法研究. PhD thesis, 2014.
- [9] 张航, 程清, 武英洁, 王亚新, 张承明, and 殷复伟. 一种基于卷积神经网络的小麦病害识别方法. 山东农业科学, 50(3):137–141, 2018.
- [10] CC Tucker and S Chakraborty. Quantitative assessment of lesion characteristics and disease severity using digital image processing. *Journal of Phytopathology*, 145(7):273–278, 1997.
- [11] Yutaka Sasaki, Tsuguo Okamoto, Kenji IMOU, and Toru TORII. Automatic diagnosis of plant disease. *Journal of the Japanese Society of Agricultural Machinery*, 61(2):119–126, 1999.
- [12] Santanu Phadikar and Jaya Sil. Rice disease identification using pattern recognition techniques. In *2008 11th International Conference on Computer and Information Technology*, pages 420–423. IEEE, 2008.
- [13] Shiv Ram Dubey and Anand Singh Jalal. Fruit disease recognition using improved sum and difference histogram from images. *International Journal of Applied Pattern Recognition*, 1(2):199–220, 2014.
- [14] Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, Dubravko Culibrk, and Darko Stefanovic. Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, 2016, 2016.

- [15] Alvaro Fuentes, Sook Yoon, Sang Kim, and Dong Park. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, 17(9):2022, 2017.
- [16] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of mathematical biology*, 52(1-2):99–115, 1990.
- [17] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.
- [18] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.
- [19] Kunihiro Fukushima and Sei Miyake. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer, 1982.
- [20] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.
- [21] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [22] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.