学号 2014302590083

密级_____

武汉大学本科毕业论文

基于深度学习的高分辨率遥感影像建筑物提取

院（系）名 称 ： 遥感信息工程学院

专 业 名 称 ： 遥感科学与技术

学 生 姓 名 ： 王毅

指 导 教 师 ： 崔卫红 副教授

二〇一八年六月

郑 重 声 明

本人呈交的学位论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本学位论文的知识产权归属于培养单位。

本人签名：_____ 日期：_____

摘　要

本文主要讨论基于深度学习的高分辨率遥感影像建筑物提取方法，通过搭建浅层卷积神经网络与深层 ResNet 残差网络来训练提取建筑物目标。本文探讨了各种不同的深度学习网络结构，调试实验了激活函数、损失函数、优化器及网络权值偏置等多种网络参量以评价各自对本研究课题的适用性。两种不同结构的网络模型被用来训练及测试实验数据，并对其识别精度做了评价。最后，通过与基于传统统计学习的分类方法进行比较，表明了深层卷积神经网络在图像识别建筑物提取领域的卓越性能。

关键词：深度学习；卷积神经网络；深度残差网络；建筑物提取

ABSTRACT

In this thesis, we study building extraction from high-resolution remote sensing images based on deep learning methods. Convolutional neural network and deep residual network are trained to classify and extract buildings. A variety of deep learning structures, and evaluate different network parameters, such as activation functions, loss functions and optimizers, are studied. We use trained network models to classify buildings from our datasets, and then make evaluations about the results. Finally, by comparing with traditional statistical classification methods, we show the excellent performance of the deep convolutional neural networks in the field of image recognition.

Key words: deep learning； convolutional neural network; building extraction

目　录

第 1 章 绪论

1.1 研究背景

随着城市建设的快速发展，时效完备的城市地理空间信息数据集已成为数字及智慧城市建设的迫切需要。建筑物作为最重要的城市地物目标，其大范围、短时效的信息整合与更新也是当下的热门研究课题。高分辨率遥感影像能够清晰地获取大范围地物信息，捕捉地物目标的图谱特征，因此成为获取建筑信息的主要数据源。传统卫星影像信息的提取多采用目视、人工、半自动化等方法，但是效率较低，而伴随着日益增加的数据获取、更新频率，研究高分辨率遥感影像中建筑物的高精度、自动化提取具有重大的现实意义。

深度学习是机器学习领域的一个研究方向，其目的在于建立模型模拟人类脑神经的连接结构，在图像、文本和声音等信息的识别中已经取得很大进展，对于遥感影像地物目标的提取也具有重要的借鉴意义。日前已有研究证实基于深度学习卷积神经网络的影像目标自动提取将大大提高高分辨率遥感影像的识别精度，且大幅减少人为工作量。但是由于深度学习的发展历史并不久远，且新型网络框架层出不穷，其在遥感影像建筑物提取方面的具体应用还并不完善，需要更多更为深入的实验研究。

1.2 研究现状

建筑物提取具有较长的研究历史，根据数据源的不同，建筑物提取方法可分为基于雷达影像数据的高度信息辅助提取、基于航空立体影像数据的高差信息辅助提取以及基于高分辨率遥感影像（简称高分影像）的空间、光谱信息提取[1]。一般情况下，基于雷达影像和航空立体影像的方法对数据要求非常严格，并且很难实现大范围、多时相的目标识别，不适用于大尺度地表，而基于高分影像的方法因其便捷性和实用性得到了更广泛的研究拓展和应用。

传统基于高分遥感影像的建筑物提取方法有人工目视解译、半自动解译和全自动解译三个层次。人工目视解译方法直接依靠人类经验及实地考察，效果较好但成本太高；半自动解译方法是业内目前使用最广泛的方法，综合利用人工经验

和计算机软件进行有监督的目标识别；全自动解译方法还没有得到普及，但随着计算机和人工智能领域的飞速发展，越来越多的全自动解译方法正在得到研究和检验。国际上现有的具有代表性的全自动解译方法有以下几类[2]：

（1）传统模式识别方法，利用影像的光谱特征信息进行统计识别，代表性方法有最大似然、最小距离、ISODATA、K-MEANS 等方法；

（2）边缘检测方法，基于建筑物影像的边缘特征进行边缘线段检测，提取规则建筑物的轮廓，主要方法为直线检测、Hough 变换、图搜索等，对建筑物进行建模；

（3）隐形特征检测方法，利用建筑物的阴影、纹理等隐形特征来识别建筑物，主要方法为灰度共生矩阵的计算和拓展；

（4）形态学方法，利用各种特定的形态学建筑指数整合建筑特征，进而提取建筑信息；

（5）面向对象方法，将识别单元从单个像素扩展为对象图斑，并综合上述各建筑特征，得到基于多尺度分割和多特征融合的面向对象建筑提取方法；

（6）机器学习和深度学习方法，构建训练框架，利用大量样本自动训练特征，主要为基于卷积神经网络的各种改进方法。

研究表明，相对于传统的统计方法，基于机器学习算法（如 SVM 支撑向量机和 NN 人工神经网络）的遥感影像特征分类能够获得更好的分类效果[3]。但是受各种因素影响，用于目标检测的特征一般为人工设计，难以完整表达地物目标的真实特征。由此能够自动学习提取目标特征的深度神经网络学习方法得到广泛关注，Volodymyr Mnih 等首先设计了一个由一个输入层、三个卷积层、一个池化层和一个全连接层组成的卷积神经网络对 Massachusetts 的公开建筑物影像数据集进行了训练和建筑物提取，取得了非常理想的效果[4]；Emmanuel Maggiori 及 Shunta Saito 等人在此基础上又对网络做了改进，修改了卷积核、输出函数等训练参数并调整了网络层次，并再次取得了突破[5,6]。国内方面，刘大伟等使用深度信念网络对美国亚特兰大北部某区域的 Resurs DK1 影像做了提取，对比浅层机器学习算法分类效果有了明显进步[7]；陈文康等使用 caffenet 网络四川丹棱县的无人机遥感影像进行训练，并获得了良好的分类效果[8]。研究表明，速度更快、效率更高、更自动化的深度学习方法将成为未来数年内遥感影像建筑物

及其他各种地物目标提取识别研究的热门。

1.3 论文结构

本文使用深度学习中的卷积神经网络方法提取高分辨率遥感影像中的建筑物目标，基于 Tensorflow 深度学习框架搭建卷积神经网络，通过分析遥感影像建筑物特征设计并调整网络结构，训练样本数据得到较为良好的网络模型，最后通过测试集分析模型的检测效果以及进行精度评定。文章共分为 5 个部分：

第一章为绪论，主要介绍本次研究背景、意义以及遥感影像建筑物提取技术发展和研究现状，并提出本研究的主要内容和研究方法。

第二章介绍深度学习的基本原理、发展概述以及主要的深度学习网络，重点介绍应用于本实验的浅层卷积神经网络和 ResNet 深层卷积神经网络，以及网络训练优化的参数选择。

第三章介绍应用于遥感影像建筑物提取的卷积神经网络原理，详述影像数据的预处理要求和实验使用的两种网络的具体结构。

第四章介绍本文实验过程，包括训练、测试、结果分析和精度评定。

第五章为总结和展望，总结实验成果，分析实验不足，为后续研究提供参考。

第 2 章 深度学习原理：神经网络与卷积神经网络

2.1 概述

深度学习（Deep learning）又称深度神经网络（Deep neural network），归属于机器学习范畴，其最重要的作用在于“表征学习”，亦即自动处理原始数据，不需要人工提取特征[9]。传统机器学习如支撑向量机、最大熵等方法属于“浅层”学习方法，依赖人类经验提取样本特征，并且只能获取单层学习经验，而深度学习通过对原始信号的多次特征变换，将原空间的样本特征变换转化到新的空间，进而自动地学习得到层次化的特征表示，分类效果更好且泛化能力更强，因而在图像识别领域有着明显优势。

深度学习起源于最早的单层感知机算法，拥有输入层、输出层和一个隐藏层，其基本结构如图 2-1 所示，但它只能处理最简单的线性函数，并无太大的实际意义。这个缺点被后来的多层感知机克服，增加了隐含层的个数，使用 Sigmoid 或 Tanh 等连续非线性函数模拟神经元对激励的响应，并使用梯度下降算法通过使代价函数（输入样本标签与输出层输出的误差）最小来训练各隐含层的权值，形成了最初的神经网络模型。

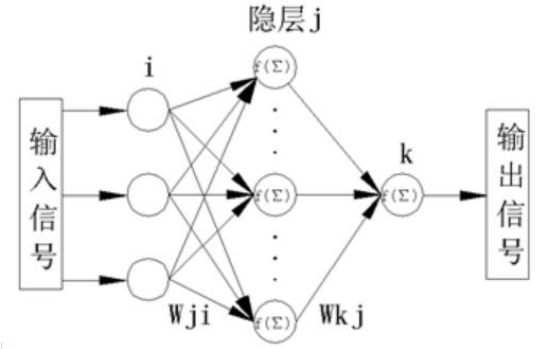


图 2-1：单层感知机

按图 2-1 所展示的基本神经网络的前向流通步骤，神经网络通过自动训练隐层参数来构建分类器，数据的传导包括前向与反向两个传播过程。第一步，输入信号传入网络后，经过多个隐层前向传播到输出层，这其中经历了各隐层参数的计算处理；第二步，比较输出层的数据与理想目标（真实值）计算得到一个损失函数或者代价函数值，用以表示网络输出与真实目标的差距；第三步，计算损失函数与各隐层参数的梯度并获取其下降的方向；最后，通过优化器更新网络参数

来减少损失函数；以后重复上述步骤不断更新网络，直到损失函数达到最小，亦即网络收敛。收敛完毕的神经网络即为一个可用的分类模型，能够对相似的输入信号进行自动分类识别。

神经网络的层次深度直接决定了它对现实生活的描述能力，但是随着网络层数的增加，网络复杂性指数级提升，这使得网络的优化容易陷入“局部最优”而无法到达“全局最优”，因此早期的神经网络实践效果并不理想。2006 年 Hinton 等提出了通过无监督预训练优化网络权值再进行微调的方法缓解了局部最优的问题，将神经网络的层数提高到了 7 层，自此神经网络真正有了深度，拉开了深度学习的序幕[10]。

深度神经网络由多个单层网络组成，常见的单层网络按编码情况分为只包含编码器、只包含解码器、既有编码器也有解码器三类。其中编码器提供从输入层到隐含层的前向映射，解码器以输出尽可能接近输入为目标将隐层特征反向映射到输入层。发展至今的深度神经网络大致可分为 3 类，如图 2-2 所示：

（1）前馈深度网络（FFDN），如多层感知机（MLP）、卷积神经网络（CNN），由多个编码器层组成；

（2）反馈深度网络（FBDN），如反卷积网络（DN）、稀疏自编码网络（HSC），由多个解码器层组成；

（3）双向深度网络（BDDN），如深层波尔兹曼机（DBM）、深层信念网络（DBN），通过多个编码器层和解码器层的叠加组成。

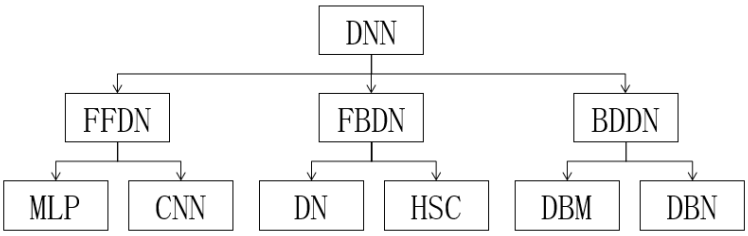


图 2-2：深度神经网络分类[11]

2.2 卷积神经网络（CNN）

神经网络层数的增加提高了网络模拟复杂现实的能力，但同时也使网络参数迅速增多，严重阻碍了网络训练的收敛效率。图 2-3 展示了一个典型神经网络的层级结构，内部各层全部连接，每一个连接都有其特定的权值参数，因此整个网络参数的个数随输入输出单元及层数的增加呈指数级增长，在现实实践输入数据

量巨大的情况下极易造成参数爆炸，并严重影响网络的训练效果。针对这个问题，结合图像识别中固有的局部模式，卷积神经网络应运而生。

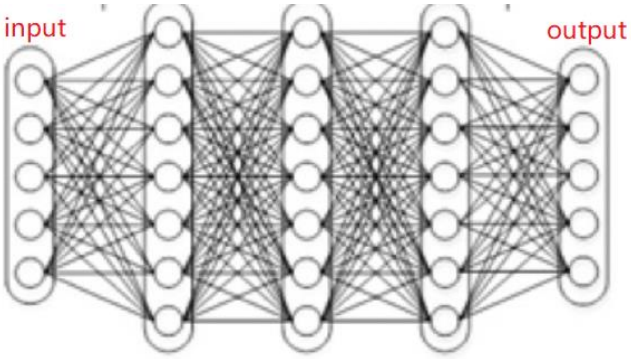


图 2-3：典型神经网络结构

卷积神经网络的层级结构与典型网络类似，但在层间互联上做了改进，尤其适用于图像信息的识别。对于卷积神经网络而言，邻层的神经元并不直接相连，而以“卷积核”为中介，一个神经元只与部分邻层神经元形成局部连接，如图 2-4 右侧所示，由此大大减少了网络参数的数量，并因为同一个卷积核在同一特征平面中权值共享，图像特有的局部上下文信息也能得以受用。

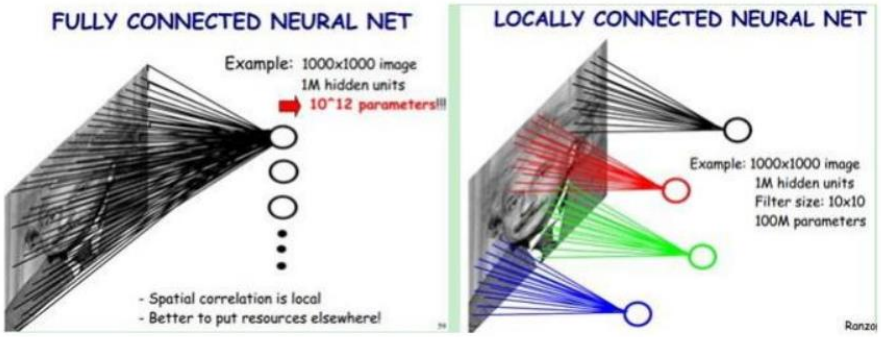


图 2-4：全连接与局部连接[12]

典型的卷积神经网络的基本结构如图 2-5 所示，由输入层、卷积层、池化层(下采样层)、全连接层和输出层组成[13]：

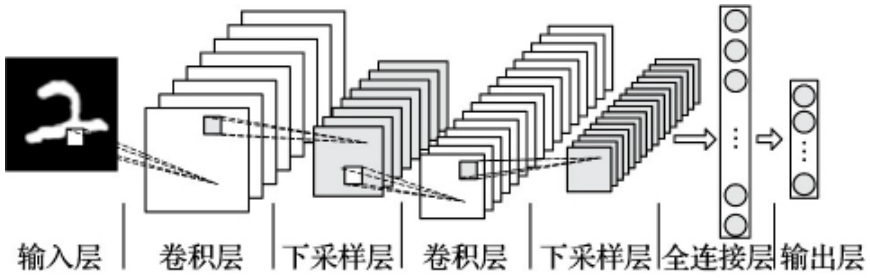


图 2-5：典型卷积神经网络

（1）输入层

卷积神经网络的输入一般为原始图像矩阵，为实际需要，可对原始数据进行预处理后输入；

（2）卷积层

卷积层（convolutional layer）由输入图像经过卷积操作后得到，卷积层由多个特征平面组成，每个特征平面的每个神经元通过卷积核与上层平面局部区域相关联。卷积核为对局部加权求和的权值矩阵，亦即对输入图像进行滤波处理得到更集中的特征。图 2-6 展示了对一个 5*5 的输入图像使用 3*3 的卷积核与 1 的步长进行卷积操作的过程，同一特征层内使用同一个卷积核（即权值共享），充分利用图像上下文信息的同时大大减少了网络参数的数量。多个卷积核对输入层图像进行多次卷积，并经过激活函数输出，进而得到多个特征平面。

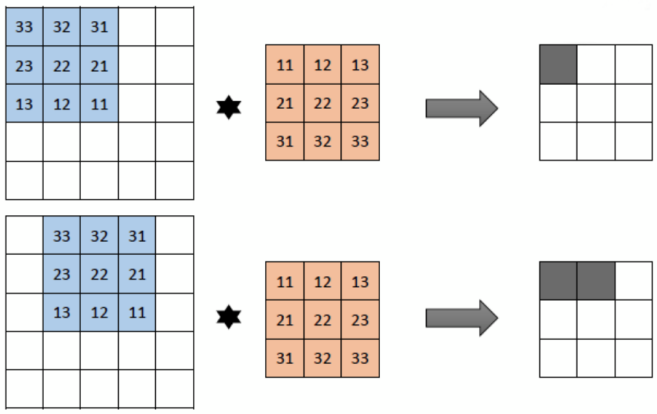


图 2-6：卷积与卷积核

此外如图 2-7 所示，上层输入图像可能有多个通道，因此每个卷积核卷积后得到的结果是由各个通道分别卷积得到的值的和。假设输入为 n 个通道的 h*w 大小的图像，使用 k 个 x*y 大小的卷积核进行卷积操作，卷积步长分别为 s1 和 s2，最终得到 k 个特征平面，大小为：

$$\left[\frac{(h-x+2*pad1)}{s1}+1\right]*\left[\frac{(w-y+2*pad2)}{s2}+1\right]$$

其中 pad 是一个张量，代表每一维填充多少行/列，表示卷积后的图像是否保留最外圈的值。

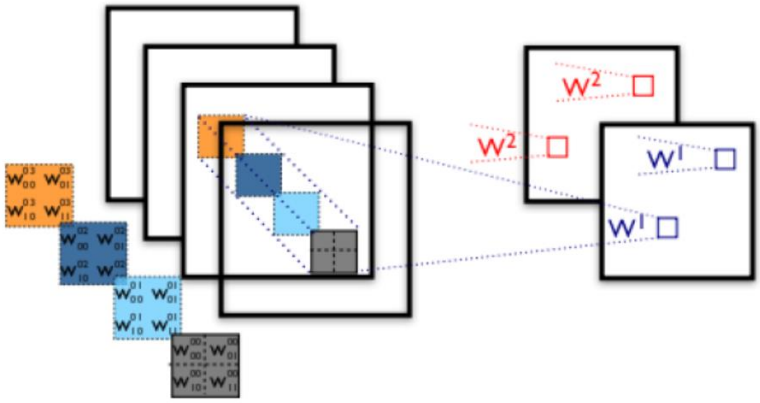


图 2-7：2 个卷积核对 4 通道图像卷积得到 2 个特征平面

卷积神经网络通过卷积层操作提取输入数据的特征，一般情况下，训练完毕的网络低层卷积提取边缘、线条、形状等低级特征，而更高层的卷积能够提取更复杂的组合特征[14]。

（3）池化层

池化层（pooling layer）又称为下采样层，通常与卷积层配合出现，由多个特征平面组成，特征平面的个数与上层卷积层的特征平面数相同。池化层的输入为卷积层的特征平面，池化层输出平面与卷积层平面一一对应，并且池化层神经元也与输入层区域对应。池化层的作用为进一步提取特征，并减少特征层神经元的个数。池化操作实际上与卷积操作相同，即使用卷积核（池化核，一般为 2*2 大小）对上层输入做卷积处理，得到更深层次的特征平面。

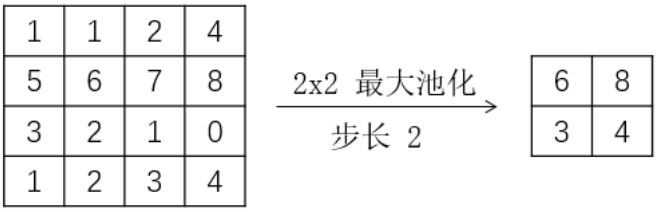


图 2-8：2*2 的最大值池化

图 2-8 展示了一个 2*2 最大值池化的过程，在实际操作中如果池化层的输入单元大小不是 2 的倍数，一般采取边界补零的方式补成偶数，然后再池化。假设输入为 k 个 h*w 大小的特征平面，经过步长为 s 的 x*y 池化后，得到 k 个池化后的特征平面，大小为：

$$\left[\frac{h-x}{s}+1\right]*\left[\frac{w-y}{s}+1\right]$$

常用的池化方法有最大值池化、平均池化和随机池化，其中最大值池化为取

局部最大值，平均池化为对局部所有值取平均，随机池化为根据概率取局部中任一值[15]。研究表明，对于具有稀疏特征的图像适用于最大值池化，并且使用线性分类器如 SVM 支撑向量机时最大值池化能够获得一个更好的性能[16]。随机池化方法按照区域内值的大小赋以概率并随机选择，能够有效利用一些较小激励的神经元，且避免过拟合的问题[17]。此外，通常池化操作的步长设置为与池化核的大小相同，即使各池化区域不重合；但在某些情况下，使用重叠采样方法能够有效增强泛化能力[18]。

（4）全连接层

经过多个卷积和池化层后，卷积神经网络连接着 1 个或多个全连接层，其中每个神经元与前一层的所有神经元相连构成全连接，亦即普通深度神经网络的层间连接，能够整合前面特征平面中具有类别区分性的信息[19]。假设最后一个卷积池化层输出 k 个 h*w 大小的特征平面，则全连接层有 k*h*w 个神经元，从而做到所有神经元全部连接。

另外，为了避免大型网络训练较小数据集出现的过拟合问题，如图 2-9 所示，常在训练时给全连接层加入 dropout 结构，通过概率控制使部分节点无效。由于这种随机性，每次输入到网络中的训练样本对应的网络结构不同，但是权值依然共享，这种方法降低了神经元间相互依赖的复杂性，可以有效避免过拟合[18]。

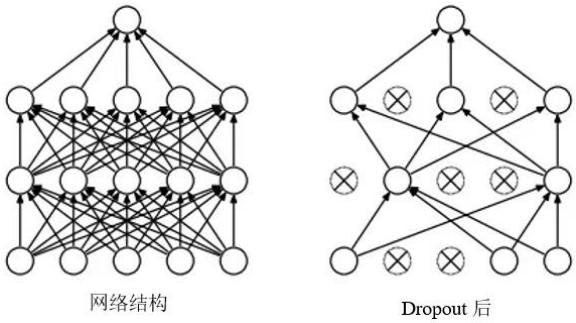


图 2-9：dropout 结构

（5）输出层

卷积神经网络的输出层实际为最后一个全连接层，根据需要输出合适数量的神经元值。对于分类问题，一般的网络训练会将图像标签进行 one_hot 处理，即将类别标签转换为 0、1 的二值向量，输出层神经元数目为图像标签 one_hot 后的类别数。其作用在于可以在之后使用 Softmax 函数处理输出每个类别的概率值，进而判断其归属的类别。

卷积神经网络的输出与原有标签构成网络训练组合，与普通神经网络类似，网络训练使得输出与标签（理想输出）的差距最小。与传统深度神经网络相比，卷积神经网络结构更为简单，其卷积池化的过程就是特征提取的过程，而由于卷积核的权值共享和局部信息整合，网络中可训练的参数变少，泛化能力得到大大增强。另外，卷积神经网络的可拓展性更强，层数可以达到很深，进而具有更强的复杂现实表达能力，而且更易于训练。

2.3 深度残差网络（ResNet）

卷积神经网络因其在图像识别方面的卓越效果在计算机视觉领域中被广泛使用，并很快被引入到遥感影像的分类提取研究中。然而简单浅层的网络无法有效满足复杂图像尤其是大尺度多光谱遥感影像的需求，因此深层卷积神经网络得到热点关注。2012 年多伦多大学 Hinton 团队提出的 8 层 AlexNet 网络在计算机视觉顶级赛事 ILSVRC 比赛中成果显著，同时也拉开了深度卷积神经网络在计算机视觉领域应用的序幕；2014 年 Google 公司的 GoogleNet 和牛津大学的 VGGNet 在当年的 ILSVRC 中再次大放异彩，两个网络分别具有 19 层及 22 层结构，在分类错误率上优于 AlexNet 数个百分点，再一次将深度卷积神经网络推上了新的巅峰。

网络层数的增加使整个模型增强了特征表示能力，近年来基于 VGG 和 AlexNet 等深层卷积网络的模型算法也在遥感影像分类领域得到了实验和探索。但是随着后续研究的深入，深层网络出现了新的问题。如图 2-10 所示，何恺明等通过实验发现，在网络深度较深时，继续增加层数并不能提高性能，甚至出现了显著的退化[20,21]。这种称为 Degradation 的现象并不是参数过多导致的过拟合问题，因为训练误差也在同步增大；同时训练中的网络模型确实收敛，因此与梯度消失和爆炸的关系也并不太大。这个问题的具体原因还待深究，但如果将增加的层都变为全等映射，那么肯定不会产生性能损失。如果增加的层近似单位映射，也就是说能够尽可能多的学习原始输入，那么训练效果可能得到提升。基于这种思路，深度残差学习（Deep Residual Learning）应运而生。2015 年提出的深度残差网络为图像识别领域带来了又一次突破，其远超其他大型网络的深度和并未受损的训练精度使其成为最新最火爆的研究热点。而鉴于深度残差网络出现时间很短，业内对其应用至遥感信息的提取研究依然缺乏，由此本实验将其作为重

点研究对象，讨论这个新兴网络结构在建筑物提取中的应用效果。

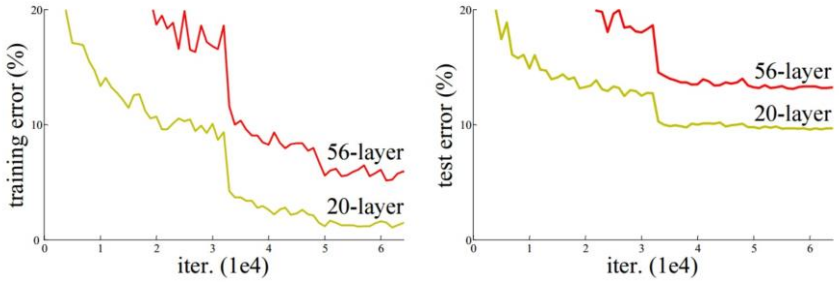


图 2-10：训练层数增加，训练及测试误差均上升[20]

深度残差网络的整体结构与普通深度学习网络几乎一致，区别在于前者训练学习的是每层输出与输入的残差。假设神经网络的输入为 x，正常期望的输出为 H(x)，现在直接把输入 x 传到输出作为初始结果，即图 2-11 所示的“快捷映射”（shortcut connection），那么此时所需学习的目标即为 F(x)=H(x)-x，它是最优解 H(x)和全等映射 x 的残差。通过设置这样的残差单元，网络每个模块只学习残差 F(x)，并且原始输入能够作用于每一个中间隐藏层，网络更加稳定有效，一定程度上解决了之前所说的 Degradation 退化问题。

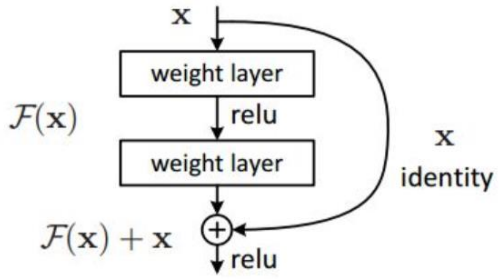


图 2-11：残差学习单元（快捷映射）[21]

实际应用中，考虑计算成本问题，残差单元（block）也可进行优化，如图 2-12 所示将两个 3*3 的卷积层替换为两个 1*1 的卷积和 1 个 3*3 的卷积。新结构中，中间的 3*3 卷积层首先在一个 1*1 卷积层下降维，然后在另一个 1*1 层下还原，减少计算量的同时保持了精度，这种结构被称为“瓶颈”（bottleneck）。

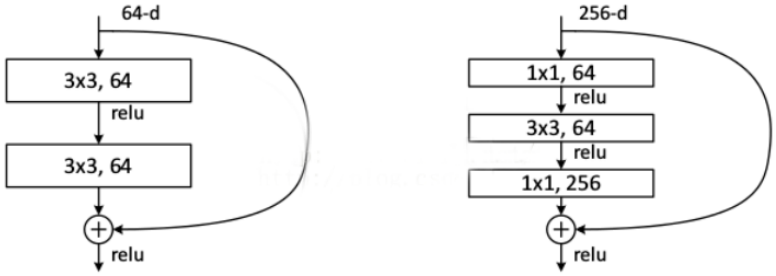


图 2-12：原始残差单元与改进后的“瓶颈“单元[21]

表 2-1 给出了几种典型残差网络的结构层次，浅层结构使用简单残差单元，深层结构使用改进后的“瓶颈“单元。测试表明从 18 层到 152 层，网络深度不断增加，而网络效果并没有受到前述退化问题的影响。

表 2-1：典型残差网络结构

层名	输出大小	18 层	34 层	50 层	101 层	152 层
Conv1	112x112	7x7, 64, stride 2				
Conv2	56x56	3x3 max pool, stride 2				
		$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{bmatrix}\times3$
Conv3	28x28	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{bmatrix}\times8$
Conv4	14x14	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times6$	$\begin{bmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{bmatrix}\times23$	$\begin{bmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{bmatrix}\times36$
Conv5	7x7	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{bmatrix}\times3$
	1x1	Average pool, 1000-d fc, Softmax				

图 2-13 展示了 19 层 VGG 网络、34 层普通深度神经网络与 34 层 ResNet 网络的结构对比图，图中可见 ResNet 网络与普通网络结构的最大区别在于层间跳跃的全能映射（即各层“快捷映射”）。另外，残差网络控制了参数数量，网络结构偏瘦；网络存在比较明显的层次递进关系，特征表达突出；网络池化层较少，大量使用卷积层进行下采样，可以提高传播效率；最后，网络没有使用 dropout 结构，而使用全局池化和批归一化进行正则化处理，进一步提高训练效率。

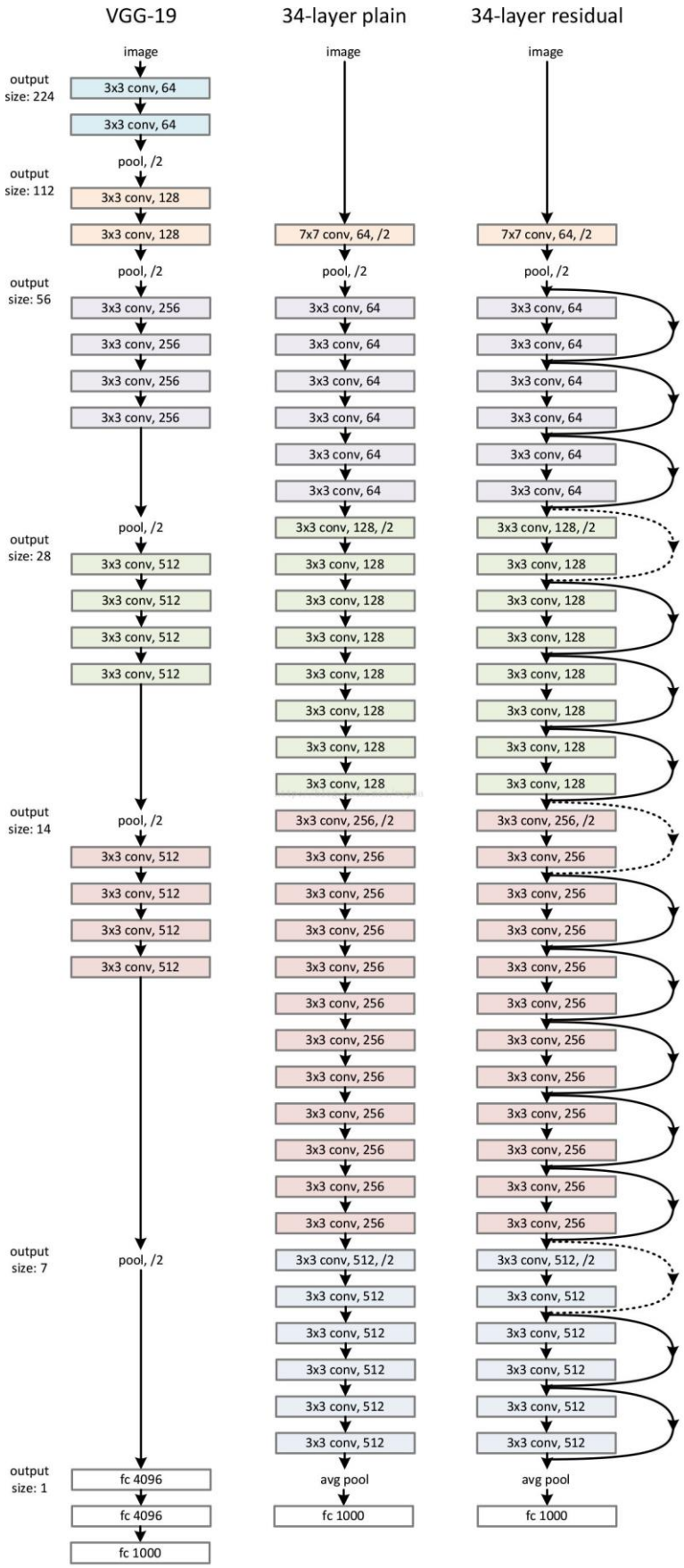


图 2-13：VGGNet、普通 DNN 与 ResNet[21]

深度残差网络对比普通“平整”网络有着显著优势，但也有些地方需要注意。首先残差网络在浅层时并未表现出更多的优势，因此其必须要配合较大的深度；其次即使采用了残差单元，网络也并不是越深越好，在 1200 层 Resent 网络的实验中，分类效果甚至不如 34 层网络，除开过拟合问题外，很有可能如此深的网络使得其再次出现之前的退化问题；最后 ResNet 网络与其他网络的区别主要在于残差的学习，其他网络使用的结构优化方法在此也适用，比如 ResNet 第二个版本的批归一化等优化操作，都可以使网络效果更上一层楼。

2.4 网络训练及优化

输入数据经过初始化后的各层网络传播到输出层，又与样本真实标签对比得到误差，由此误差反向传播其梯度到各个隐层单元，并依此更新各隐层参数数值，重复以上步骤使得误差收敛减少，最终使网络模型能够输出理想的真实标签。为使网络能够更快更好地训练完毕，网络当中的各种函数参数就显得尤为重要，通常情况下网络的优化方案除自身层次结构的调整外，还有层间激活函数、输出代价函数以及误差的优化函数三种：

2.4.1 激活函数

如图 2-14 所示，激活函数（activation function）作用于神经网络每一层的输出，其作用为向网络加入非线性因素，使其可以更好地解决较为复杂的现实问题。如图 2-15 所示，激活函数使网络可以解决非线性问题。

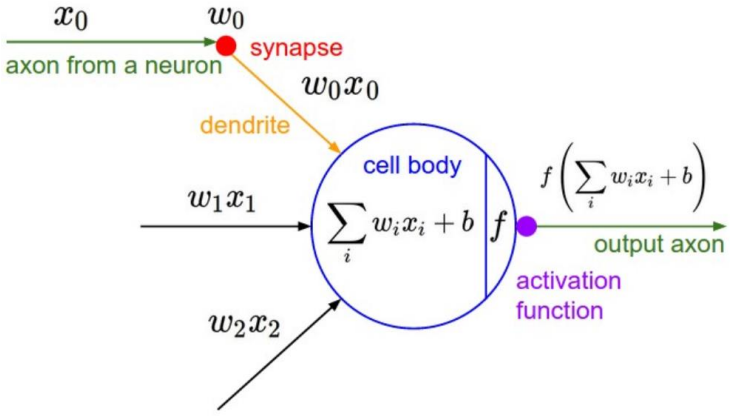


图 2-14：激活函数

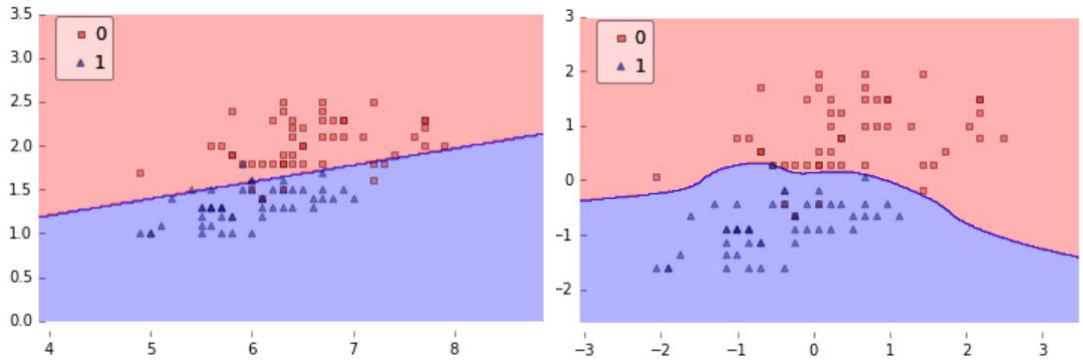


图 2-15：当简单线性边界无法分类时，使用激活函数可以获得非线性边界

常用的激活函数有 Sigmoid、Tanh 双曲正切、ReLU 与 Softmax 四种：

（1）Sigmoid 函数

如图 2-16 所示，Sigmoid 函数形状类似指数函数，其应用范围最为广泛，也是最接近人脑神经元的函数。Sigmoid 函数定义如公式 2-1 所示：

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2-1}$$

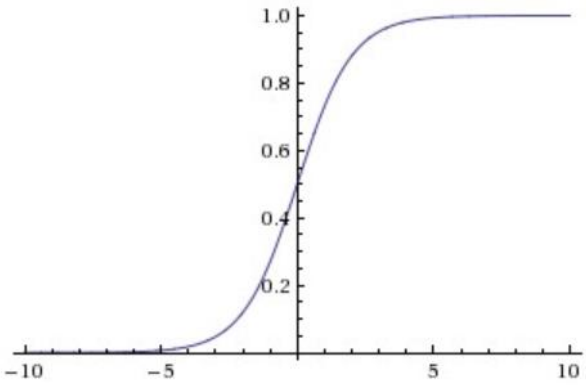


图 2-16：Sigmoid 函数

Sigmoid 函数的优点在于其输出在(0,1)之间，输出范围有限，单调连续，优化稳定，可以用作输出层；缺点在于其饱和性，即当输入远离原点时，函数梯度剧烈减小，容易出现梯度消失[1]问题，这样不利于权重的优化。

（2）Tanh 函数

Tanh 是双曲正切函数，其定义如公式 2-2 所示：

[1] 梯度爆炸与梯度消失：深度学习中误差采用“链式法则”反向传播，每层的梯度需要进行连乘操作，当各层梯度均很小（小于 1）时，连乘后的梯度趋于 0，即梯度消失；当各层梯度均很大（大于 1）时，多层连乘后的梯度趋于无穷，即梯度爆炸。

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2-2}$$

如图 2-17 所示，Tanh 函数与 Sigmoid 函数的曲线相近，区别在于其输出区间在（-1，1）之间，且以 0 为中心，但同样也有饱和性与梯度消失问题。一般二分类问题中，Tanh 函数适用于隐藏层。

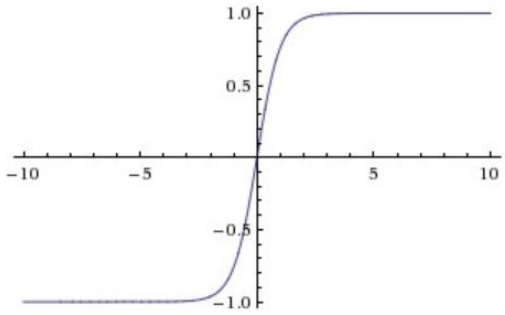


图 2-17：Tanh 函数

（3）ReLU 函数

如图 2-18 所示，ReLU 函数是一个分段函数，它是近年来最受欢迎的激活函数，其定义如公式 2-3 所示：

$$f(x) = \begin{cases} x, x \geq 0 \\ 0, x < 0 \end{cases} \tag{2-3}$$

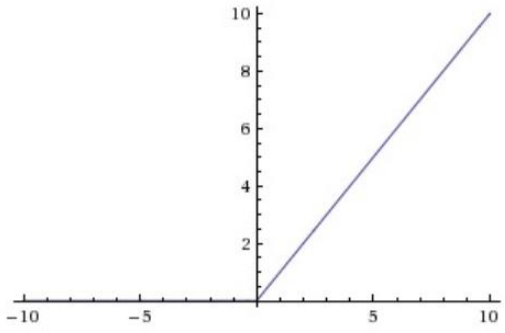


图 2-18：ReLU 函数

相较于 Sigmoid 和 Tanh 函数，ReLU 函数在输入为正数时不会出现饱和，可以保持梯度不变，进而缓解梯度消失。但是 ReLU 函数也有缺陷，当输入为负数时函数响应为 0，即完全不被激活，这在前向传播中问题不大，而在反向传播更新权值时会导致“神经元死亡”，即权重无法更新，梯度也会完全降到 0。

（4）Softmax 函数

在有关识别分类的问题中，神经网络的输出层一般使用 Softmax 激活函数，

其作用在于将网络输出转为类别概率值，便于判断训练结果。Softmax 函数的定义如公式 2-4 所示：

$$f(x) = \frac{e^x}{\sum e^x} \tag{2-4}$$

其中，x 表示数组中的元素，f(x)表示输出，即元素的 Softmax 值是该元素的指数与所有元素指数和的比值。通过公式可知一组元素的 Softmax 函数值分布在 0 和 1 之间且和为 1，这使得网络的输出可以用概率表示，输出越接近于 1，则属于该类别的概率越大。

各种不同的激活函数对网络有着不同的作用，并有着各自的优缺点，针对具体应用需要尝试使用各种函数进而选择最合适者。此外，业内也已出现了 maxout 等改进的激活函数，但是面对特定训练场景还需通过实验来进行最终选择合适的函数。

2.4.2 代价函数

输入信号经过网络前向传播到输出层后得到输出信号，网络需要将其与标签对比进行训练，即使用代价函数或者损失函数来表示训练误差。代价函数用来估量模型预测与真实值的不一致程度，神经网络的训练正是通过使代价函数逐渐减小来使网络参数接近理想目标。

常见的代价函数有二次代价函数、对数代价函数、指数代价函数、Hinge 代价函数、0-1 代价函数、绝对值代价函数等：

（1）二次代价函数

二次代价函数即最小二乘代价函数，它是线性回归的一种方法，原则为使各点到回归线的距离之和最小（平方和最小）。二次代价函数通常使用欧几里得距离进行误差的距离度量，其定义如公式 2-5 所示：

$$L(y, f(x)) = \sum_{i=1}^{n} (y - f(x))^2 \tag{2-5}$$

其中$y - f(x)$表示残差，整个公式表示残差的平方和，使其最小化为网络训练的目标，亦即最小化二次代价函数。在实践操作中，有时也会使用均方差作为改进

的二次代价函数。

（2）对数代价函数

对数代价函数对应逻辑回归，它求得满足伯努利分布的样本的似然函数，并用对数求极值。逻辑回归并没有求对数似然函数的最大值，而是把极大化作为思想，推导它的风险函数为最小化的似然函数的负值。

$$L(y, P(y|x)) = -\log P(y|x) \tag{2-6}$$

如公式 2-6 所示，在极大似然估计中，先取对数再求导数，然后再寻找极值点，代价函数 L 就是使$-\log P(y|x)$最小，亦即使概率$P(y|x)$达到最大。

（3）指数代价函数

指数代价函数对应机器学习中的 Adaboost 算法，设样本总数为 n，Adaboost 的指数代价函数如公式 2-7 所示：

$$L(y, f(x)) = \frac{1}{n}\sum_{i=1}^{n} \exp(-y_i f(x_i)) \tag{2-7}$$

（4）Hinge 代价函数

Hinge 代价函数对应机器学习 SVM 算法，其标准形式如公式 2-8 所示：

$$L(y) = \max(0, 1 - ty) \tag{2-8}$$

其中 y 是预测值，在-1 到 1 之间，t 为目标值，其含义为鼓励 y 的值在-1 和 1 之间，而并不鼓励分类器过于自信（y 的绝对值大于 1），这样使分类器更专注于整体的分类误差而不拘泥于个别样本。

（5）0-1 与绝对值代价函数

常见的代价函数除了上述几种，还有 0-1 代价函数与绝对值代价函数等（如公式 2-9 与 2-10 所示）：

0-1 代价函数：
$$L(y, f(x)) = \begin{cases} 1, y \neq f(x) \\ 0, y = f(x) \end{cases} \tag{2-9}$$

绝对值代价函数：
$$L(y, f(x)) = |y - f(x)| \tag{2-10}$$

上述各种代价函数适用于各自的算法场景，如果网络使用线性激活函数，则二次代价函数具有较好效果，但在卷积网络使用 Sigmoid 等 s 型激活函数时，二次代价函数常常因远离中心处梯度几乎消失而带来收敛过慢的问题。此时使用

交叉熵代价函数，其定义如公式 2-11 所示：

$$C = -\frac{1}{n}\sum_{x}[ylna + (1-y)\ln(1-a)] \tag{2-11}$$

其中 C 表示代价函数，n 为样本总数，x 表示样本，y 表示实际值，a 表示预测输出。通过求导可以发现，当使用交叉熵代价函数时，误差越大则梯度越大，参数调整就越快，训练速度也越快。此外，在输出层使用 Softmax 作激活函数时，通常会使用对应的对数释然代价函数，其实质上是多类别推广后的交叉熵代价函数。

2.4.3 优化函数

在神经网络的训练中，输出层输出神经元经过前述代价函数的计算后得到当前的损失值，网络训练的目标是使其减到最小。优化函数（优化器）作用于代价函数之后，根据损失值反馈到之前的网络层中，进而向目标方向更新网络参数。网络优化实际上就是对梯度的优化，因此最基本的优化函数为梯度下降法，其他常用的优化函数基本都是在此基础上的改进。常用的梯度下降优化方法有批量梯度下降法、随机梯度下降法、小批量梯度下降法、Momentum 优化法、NAG 优化法、AdaGrad 优化法、RMSProp 优化法、Adadelta 优化法与 Adam 优化法，分别介绍如下：

（1）BGD 批量梯度下降法

BGD 梯度下降方法是无约束条件优化最简单也是最常用的方法，其核心思想为负梯度方向函数曲线下降最快，每次迭代时根据负梯度的方向更新权值，进而求得代价函数最小值。在训练中，每一步迭代都使用训练集的所有内容，根据每一个输入生成估计跟实际输出比较，统计所有误差进行平均，以此作为更新参数的依据。迭代更新过程为：

1）提取训练集所有输入$\{x_i\}$及其相关输出$\{y_i\}$

2）计算梯度、误差并更新参数（如公式 2-12 所示）：

$$g \leftarrow +\frac{1}{n}\nabla_\theta \sum_i L(f(x_i;\theta), y_i) \tag{2-12}$$

$$\theta \leftarrow \theta - \epsilon g$$

其中θ为初始参数，ϵ为学习率。

BGD 梯度下降法利用了训练集中的所有数据，因此能够充分利用训练样本的所有信息，当代价函数达到最小后梯度为 0，即能够收敛至全局最优点；但这种方法并不适用于非常大的数据集，也会使运行速度变慢。

（2）SGD 随机梯度下降法

随机梯度下降法和批量梯度下降法的区别在于计算梯度的样本没有使用全部数据，而是随机选取，且每次迭代都针对单个样本数据。这种方法的训练速度很快，因为并不需要遍历全局样本，但也因此容易陷入局部最优解而无法收敛至全局最优，这就需要对学习率进行多次衰减以控制优化进程。

（3）MBGD 小批量梯度下降法

小批量梯度下降法是前述两种方法的综合，在每次迭代时随机选择小部分样本计算内部所有梯度的和，以此来更新参数。迭代更新过程为：

1）从训练集随机抽取一批容量为 m 的样本x_i及相关输出y_i

2）计算梯度、误差并更新参数（如公式 2-13 所示）：

$$g \leftarrow +\frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta),y_i)$$

$$\theta \leftarrow \theta - \epsilon g \qquad (2\text{-}13)$$

$$\epsilon \leftarrow h(\epsilon)$$

其中θ为初始参数，ϵ为学习率，$h(\epsilon)$为学习率衰减函数。

小批量梯度下降法训练速度也很快，并可以大大减少局部收敛的可能性，但是由于样本仍是抽取的，得到的梯度还有误差，因此也需要使学习率逐渐减小。常用的学习率衰减方法多种多样，一般为线性衰减或指数衰减，另外初始学习率的设置也需要实验确定。

（4）Momentum 优化法

Momentum 方法主要是为了缓解前述随机梯度下降法迭代梯度噪声的问题，它借用了物理中动量的概念，即使当前权值的改变受到上一次或几次权值改变的影响，类似于增加惯性变量。这里引入了新的速度变量来表示动量，迭代更新过程为：

1）从训练集中随机抽取一批容量为 m 的样本x_i及相关输出y_i

2）计算梯度、误差并更新速度和参数（如公式 2-14 所示）：

$$\begin{aligned} g &\leftarrow +\frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta),y_i) \\ v &\leftarrow \alpha v - \epsilon g \\ \theta &\leftarrow \theta + v \end{aligned} \tag{2-14}$$

其中θ为初始参数，ϵ为学习率，参数α表示每回合速率 v 的衰减程度。

Momentum 方法利用“惯性”作用于学习速度的改变，训练中前后梯度方向一致时，学习速度更快；前后梯度方向不一致时，也能及时控制波动。

（5）NAG（Nesterov Momentum）优化法

NAG 优化方法是对 Momentum 优化的改进，在 Momentum 方法中梯度下降会盲目跟随惯性，因此为其加上控制器，使其梯度下降的幅度随最优的接近而降低。迭代更新过程为：

1）从训练集中随机抽取一批容量为 m 的样本x_i及相关输出y_i

2）计算梯度、误差并更新速度和参数（如公式 2-15 所示）：

$$\begin{aligned} g &\leftarrow +\frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta+\alpha v),y_i) \\ v &\leftarrow \alpha v - \epsilon g \\ \theta &\leftarrow \theta + v \end{aligned} \tag{2-15}$$

其中θ为初始参数，ϵ为学习率，参数α表示每回合速率 v 的衰减程度。

（6）AdaGrad 优化法

AdaGrad 优化方法是基于 SGD 的一种算法，它的核心思想是对比较常见的数据给予较小的学习率，对比较罕见的数据给予较大的学习率。这种方法的学习率根据以往参数自动调整，迭代更新过程为：

1）从训练集中随机抽取一批容量为 m 的样本x_i及相关输出y_i

2）计算梯度、误差，更新梯度累计量 r 并更新速度和参数（如公式 2-16 所示）：

$$\begin{aligned} g &\leftarrow +\frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta),y_i) \\ r &\leftarrow r + g \odot g \end{aligned}$$

$$\Delta\theta = -\frac{\epsilon}{\delta + \sqrt{r}} \odot g$$
$$\theta \leftarrow \theta + \Delta\theta \qquad (2\text{-}16)$$

其中θ为初始参数，ϵ为学习率，r 为梯度累计量（初始化为 0），是一个很小的常量，以防出现分母为 0 的情况。

AdaGrad 方法能够自动更改学习率，适合于数据稀疏的数据集，但是在深度网络层数过多时学习率会越来越低，最终造成训练提前结束。

（7）RMSProp 优化法

RMSProp 方法借鉴了 AdaGrad 方法，但引入了一个新的衰减系数，使梯度累计量 r 每回合衰减一定比例。迭代更新过程为：

1）从训练集中随机抽取一批容量为 m 的样本x_i及相关输出y_i

2）计算梯度、误差，更新梯度累计量 r 并更新速度和参数（如公式 2-17 所示）：

$$g \leftarrow +\frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta), y_i)$$
$$r \leftarrow \rho r + (1-\rho) g \odot g$$
$$\Delta\theta = -\frac{\epsilon}{\delta + \sqrt{r}} \odot g$$
$$\theta \leftarrow \theta + \Delta\theta \qquad (2\text{-}17)$$

其中θ为初始参数，ϵ为学习率，r 为梯度累计量（初始化为 0），ρ为衰减动力系数，δ为防止分母为 0 的小量。

相较于 AdaGrad 方法，RMSProp 方法很好地解决了学习率降低训练过早结束的问题，但是又引入了新的超参数，网络调整的复杂度再次增加，同时也还是依赖于初始学习率。

（8）Adadelta 优化法：

Adadelta 方法也是 AdaGrad 方法的延伸，其特点在于控制之前偏导的累加到一定的窗口中。迭代更新过程为：

1）从训练集中随机抽取一批容量为 m 的样本x_i及相关输出y_i

2）计算梯度、误差，更新梯度累计量 r 并更新速度和参数（如公式 2-18 所示）：

$$\begin{gathered} \mathrm{g} \leftarrow +\frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta),y_i) \\ \mathrm{r} \leftarrow \mathrm{r} + \gamma \mathrm{g} \odot \mathrm{g} + (1-\gamma)\mathrm{g} \odot \mathrm{g} \\ \Delta\theta = -\frac{\epsilon}{\delta + \sqrt{r}} \odot \mathrm{g} \\ \theta \leftarrow \theta + \Delta\theta \end{gathered} \tag{2-18}$$

其中θ为初始参数，ϵ为学习率，r 为梯度累计量（初始化为 0），δ为防止分母为 0 的小量，γ为动态改变权重的控制系数，类似于 Momentum 方法中的速度 v。

（9）Adam 优化法

Adam 优化方法本质上是带有动量项的 RMSProp 方法，利用梯度的一、二阶矩阵估计来动态调整每个参数的学习率，并且学习率的范围确定，各个参数的变化较为平稳。迭代更新过程为：

1）从训练集中随机抽取一批容量为 m 的样本x_i及相关输出y_i

2）计算梯度、误差，更新动量 r 和 s，并更新速度和参数（如公式 2-18 所示）：

$$\begin{gathered} \mathrm{g} \leftarrow +\frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta),y_i) \\ \mathrm{s} \leftarrow \rho_1 \mathrm{s} + (1-\rho_1)\mathrm{g} \\ \mathrm{r} \leftarrow \rho_2 \mathrm{r} + (1-\rho_2)\mathrm{g} \odot \mathrm{g} \\ \mathrm{s} \leftarrow \frac{s}{1-\rho_1} \\ \mathrm{r} \leftarrow \frac{r}{1-\rho_2} \\ \Delta\theta = -\frac{\epsilon s}{\delta + \sqrt{r}} \\ \theta \leftarrow \theta + \Delta\theta \end{gathered} \tag{2-19}$$

其中θ为初始参数，ϵ为学习率，r、s 为动量参数，δ为防止分母为 0 的小量。

第 3 章 基于深度学习的遥感影像建筑物提取原理

3.1 数据预处理

3.1.1 裁剪样本

遥感影像过大的尺寸不利于神经网络的隐层运算，因此需要将其裁剪成若干个小尺寸图像。如图 3-1 所示，为了避免裁剪过后丢失整幅影像的空间关联信息，使用可重叠的裁剪方法，以 16 的步长将一张 1500*1500 的影像裁剪为 8100 张 64*64 大小的小图像。在对标签的裁剪中，取前述 64*64 小图的中心部分，即以 16 的步长将 1500*1500 的标签裁剪为 8100 张 16*16 的小标签。这样网络输入为 64*64 的图像，而输出与标签只对应中心 16*16 个像素，能够更有效地利用图像上下文信息。裁剪完成后，训练集得到 1109700 张影像，验证集得到 32400 张影像，测试集得到 121500 张影像，另外为了便于在大量数据中搜索，将每张影像的文件名按行存储在各个文件夹下的 txt 文件中。

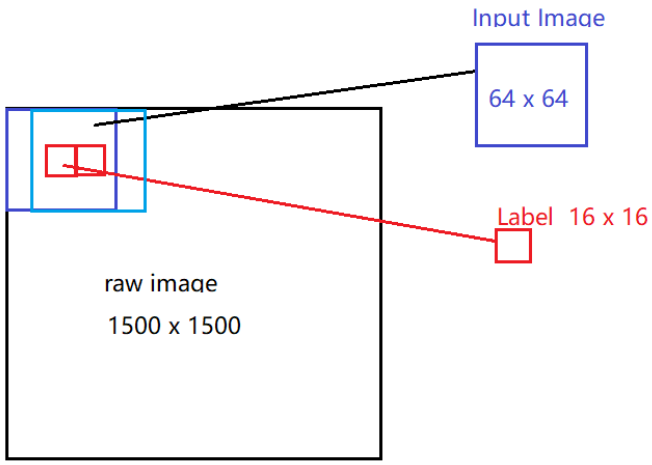


图 3-1：数据裁剪

3.1.2 转化数据格式

裁剪完之后的数据数量非常大，直接读取效率很低，因此使用 tensorflow 提供的标准数据格式 tfrecord 来存储我们的训练和测试数据。tfrecord 数据文件是一种将图像数据和标签统一存储的二进制文件，可以更好的利用内存空间，

提高程序运行效率。实验中按照前述 txt 文件内文件名的顺序将样本数据的图像、标签、图像名、序列号导入生成 tfrecord 文件，其中训练集、验证集和测试集各生成一个文件。

3.2 网络结构

3.2.1 浅层卷积神经网络

图 3-2 展示了一个浅层卷积神经网络结构，输入层为裁剪后的遥感影像，其后卷积层、池化层和全连接层依次堆叠，最后一层全连接层经过激活函数后得到最终的输出层。

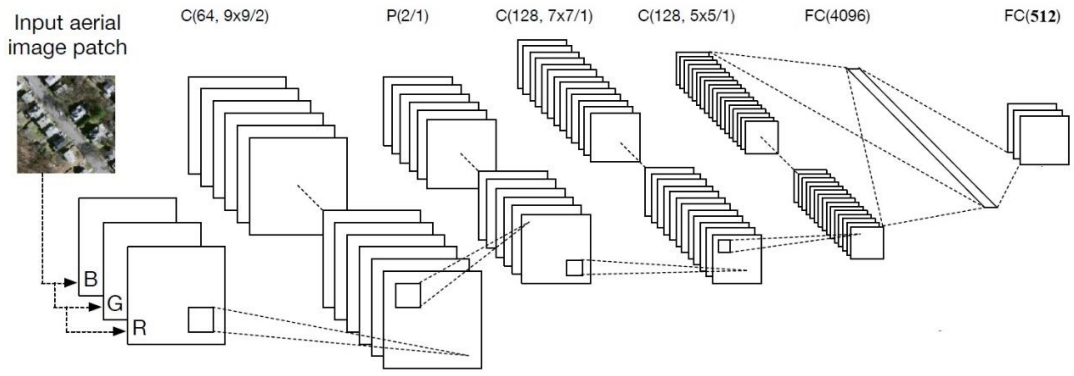


图 3-2：浅层卷积神经网络框架[6]

如图 3-3 所示，网络输入为 64*64 大小的三通道图像；第一个卷积层有 64 个 9*9 的卷积核，卷积步长为 2，这一层输出 64 个 32*32 大小的特征平面；紧接着为一个池化层，采用 2*2 池化核的最大值池化，池化步长为 2，这一层输出 64 个 16*16 大小的特征平面；第二、第三个卷积层都有 128 个卷积核，卷积核大小分别为 7*7 和 5*5，都输出 128 个 16*16 大小的特征平面；第三个卷积层后连接着第一个全连接层，该层有 4096 个神经元；第二个全连接层有 512 个神经元，对应输入图像中心的 256 个像素及各自的 2 个类别；输出层为全连接层的维度转换，将 512 个神经元转换成 256*2 的二维输出，每个像素有两个通道，值取[0,1]或[1,0]，分别表示建筑物或非建筑物，对应 one_hot 处理后的影像标签（图 3-3）。

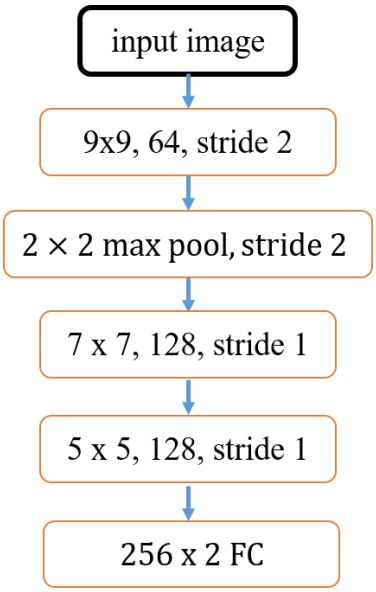


图 3-3：建筑物提取浅层网络结构

除最后一层外，各网络层的激活函数均为 ReLU 函数，最后一层每个像素的两个通道使用 Softmax 函数输出，得到其属于两个类别的概率值。

3.2.2 ResNet 深度残差网络

图 3-4 展示了应用于建筑物提取的 ResNet 网络结构，输入输出与前述浅层卷积神经网络相同，其中输入为裁剪后的三通道遥感影像，经过 ResNet 各层后通过新设的全连接层输出。

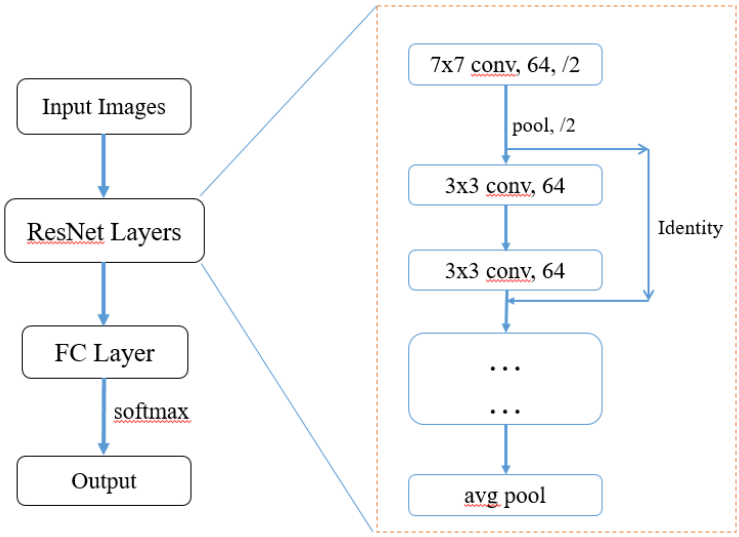


图 3-4：ResNet 网络框架

如图 3-5 所示，网络输入为 64*64 大小的三通道图像；第一个卷积层有 64

个 7*7 的卷积核，卷积步长为 2，这一层输出 64 个 32*32 大小的特征平面；紧接着为一个池化层，采用 2*2 池化核的最大值池化，池化步长为 2，这一层输出 64 个 16*16 大小的特征平面。接下来为 ResNet 的堆叠残差单元，每 3 层卷积计算一次残差，以 50 层网络为例，分别为 3 个卷积步长为 2、4 个卷积步长为 2、6 个卷积步长为 2 及 3 个卷积步长为 1 的残差块。最后一个残差单元输出 2048 个 2*2 大小的特征平面，另外增加一个含有 512 个神经元的全连接层，并将维度转换成 256*2 经过 Softmax 函数输出。其他各层网络的激活函数均为 ReLU 函数。

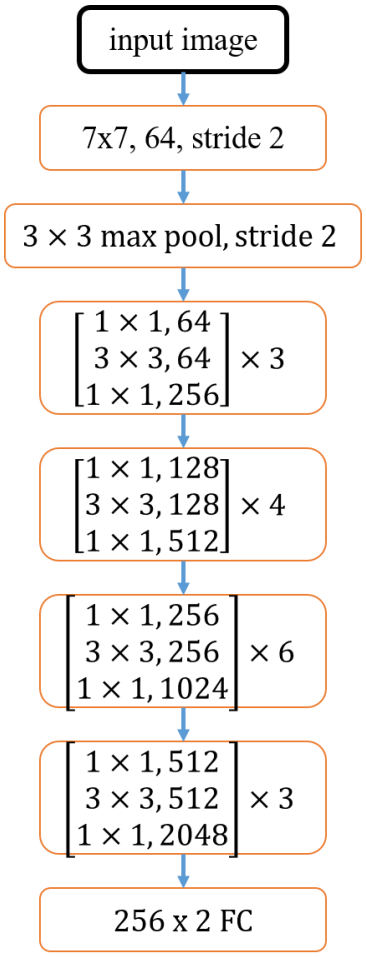


图 3-5：ResNet_50 建筑提取网络结构

第 4 章 基于深度学习的遥感影像建筑物提取实验

4.1 实验数据

本实验使用两组数据源，第一组为 Massachusetts 地区的 1 米分辨率的公开建筑物影像数据集，其中训练集为 137 张大小为 1500*1500 的三通道影像，交叉验证集为 4 张 1500*1500 的三通道影像，测试集为 10 张 1500*1500 的三通道影像，每一张影像都有对应相同大小的标签图像（二值图，0 为非建筑，1 为建筑）；第二组为 Vaihingen 地区近红外、红、绿三波段 9cm 分辨率建筑物影像数据集，共有 15 张大小不一的三通道影像，每张影像都有对应大小的标签。

另外，如图 4-1 所示，对于第一组数据中的 137 张训练样本，其中有 30 张影像损坏，因此需要首先剔除异常样本及其标签，最终得到 107 张良好的训练影像。

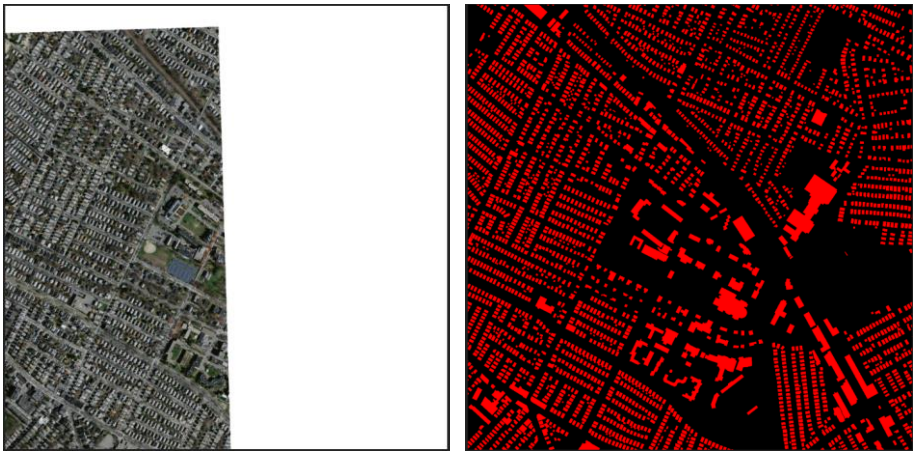


图 4-1：损坏样本（左图为影像，右图为标签）

为描述方便，本文除结果分析部分外默认基于第一组数据。

4.2 实验流程

本文使用 python 语言基于 Tensorflow 深度学习框架，分别使用浅层卷积神经网络与 ResNet 深度残差网络进行实验。实验技术路线如图 4-2 所示，主要有以下几个步骤：

（1） 收集数据，获取训练样本

（2） 数据预处理（包括裁剪、格式转换等）

（3） 搭建学习框架（1、浅层卷积神经网络 2、深度残差网络），分割样本区、确定卷积核大小、滑动步长，构建损失函数和优化函数，设置超参数（学习率、学习率减少频率、学习率衰减率、动量参数等）

（4） 训练网络，根据训练效果调整网络结构、参数等，获取收敛较好的模型

（5） 对测试数据集进行预测，并做精度评定

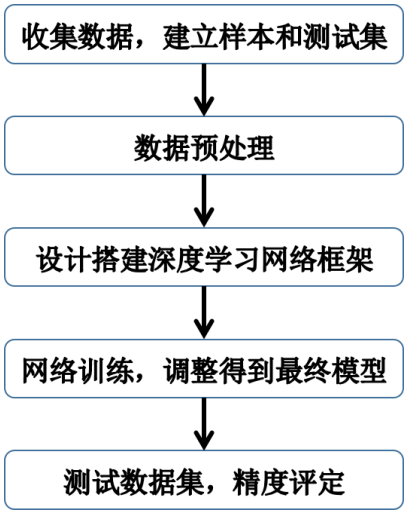


图 4-2：技术路线

4.3 网络参数设置

4.3.1 网络超参数

卷积神经网络训练的超参数主要有样本批次大小、初始学习率、迭代训练次数、学习率衰减频率等，选择合适的超参数对网络训练有很大作用。批次大小表征一次迭代样本的数目，影响网络的收敛性能；初始学习率与隐层参数的更新幅度有关，学习率过大网络无法收敛、过小网络训练效率低下；迭代次数表示训练每一批样本的批次数，需要设置足够大以使样本得到充分训练；学习率衰减频率与网络收敛性能相关，当训练足够多的样本时，需要降低学习率以使网络进一步逼近全局最优点。本实验中样本随机分批训练，批次大小为 100，共 8667 批，初始学习率设为 0.00005，迭代次数为 86670 次，即每张小图训练 10 遍，每迭代 8667 次学习率衰减一次，即乘以系数 0.1。

4.3.2 权与偏置

神经网络训练的是网络模型参数，这里的参数主要是各隐含层的权值和偏置值。在训练这些参数达到理想目标之前，需要将其初始化。不同的初始化值对网络有着非常大的影响，尤其在网络层数很深、学习率较低的情况下。常用的初始化方法有定值初始化和随机分布初始化，前者适用于偏置的初始化，本实验将所有的偏置值初始化为 0.1 这个固定值；后者适用于权值的初始化，本实验中将各权值初始化为标准差为 0.01 的高斯分布值。

4.3.3 代价函数和优化函数

本实验代价函数的计算基于网络输出和标签，前文已介绍过网络的输出层为 100*16*16*2 的双通道预测概率，而其对应标签为 100*16*16*1 的分类二值图，这里使用 tensorflow 自带的函数将两者维度统一，即将每个像素的标签转为 one_hot 形式，得到 100*16*16*2 的每个像素取值为[0,1]或[1,0]的新标签，这样即可进行后续代价函数的计算和优化。

在第 3 章里详细介绍过各种代价函数和优化函数的特点，同时指出具体实验适用于何种代价、优化函数并没有明确的规律。本实验中我们尝试了部分适用于深度神经网络的代价和优化函数的组合，同一组合在两种网络模型中的训练结果相似，但不同组合效果不一，如表 4-1 所示：

表 4-1：各种优化及代价函数效果

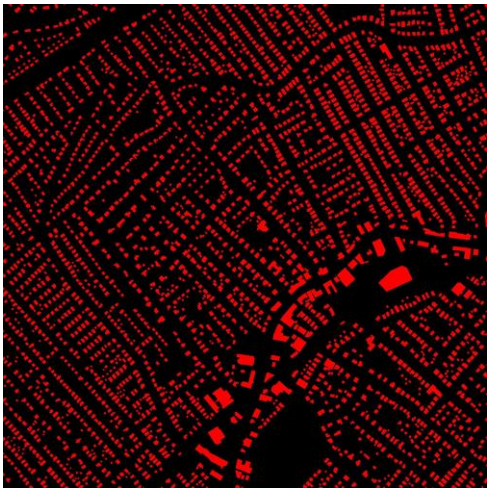
代价函数（loss）	优化函数（optimizer）	收敛速度	收敛效果
二次代价函数	SGD 优化	慢	差，无法收敛
	MBGD 优化	慢	差，无法收敛
交叉熵代价函数	MBGD 优化	较慢	较差
	Momentum 优化	快	一般
	Adadelta 优化	较快	好
	Adam 优化	快	好

4.4 实验结果与精度评定

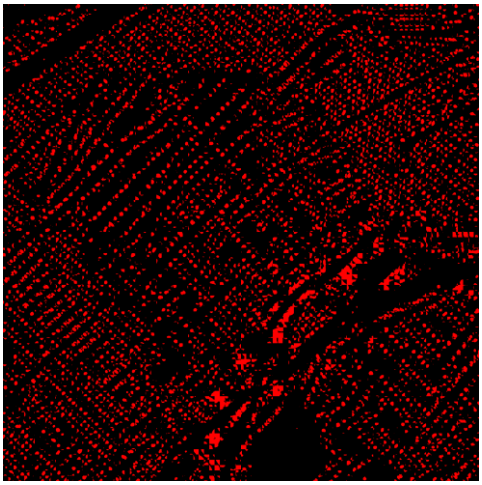
4.4.1 实验结果

图 4-3、4-4 分别展示了两组数据的预测结果与原图标签的对比，其中图 4-3(a)和(b)分别为第一组数据的一张样本原图及对应标签，图 4-3(c)为使用浅层卷积神经网络预测结果，图 4-3(d)为使用 ResNet_50 深度残差网络预测结果；图 4-4(a)和(b)分别为第二组数据的一张样本原图及对应标签，图 4-4(c)为使用 ResNet_50 深度残差网络预测结果。

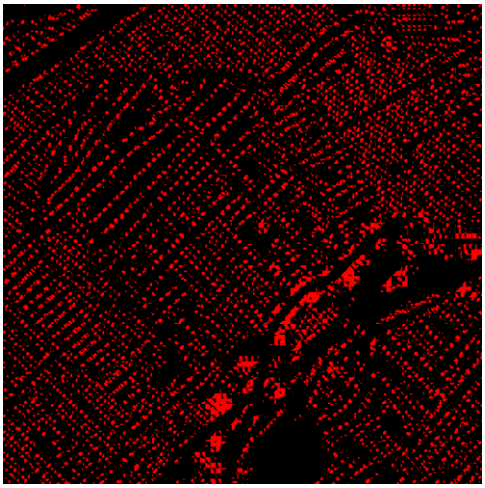
（a）测试原图



（b）测试标签

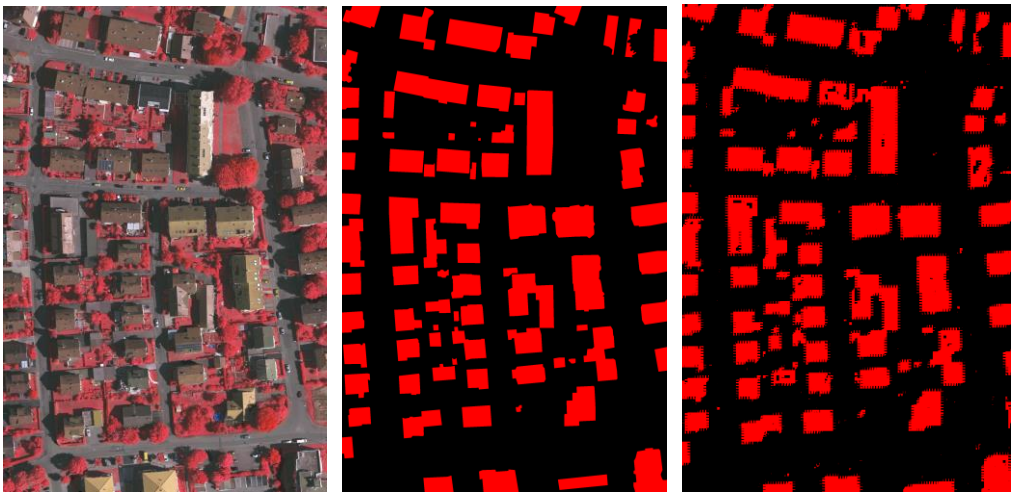


（c）浅层 CNN 预测



（d）ResNet_50 预测

图 4-3：第一组数据样本及预测结果



（a）测试原图　　（b）测试标签　　（c）ResNet_50 预测

图 4-4：第二组数据样本及预测结果

由图可见，训练完毕的两种网络提取精度都较高，能够识别出大部分建筑目标。第一组数据的实验中，使用两种卷积网络提取出的图斑块较为琐碎，这是样本训练不足导致的问题，当增大训练次数时得到缓解。放大后的第一组数据断裂的部分大都出现在 16*16 的小区域边缘，同样的问题在第二组数据中也有体现，这与预处理时图像裁剪的大小及后续网络对边缘信息的补充有关，另外增加训练批次的大小以及增加训练次数后也可以有效缓解此现象。

影像中大多较为明显的建筑目标能够被网络有效提取，而一些模糊区域较易出现误分，尤其是建筑阴影区域，网络提取效果并不理想。这个问题在第二组分辨率较高的数据实验中尤为明显，所有的建筑边缘都存在锯齿状条纹。在网络的设计中对输入影像进行重叠裁剪，这使网络有条件学习到包括阴影、纹理在内的多种上下文信息，但由于网络提取特征的自动性和不可控性，这些特征并没有被突出。可以考虑的改进为对影像预处理时提前做阴影分析，将建筑物阴影信息整合到建筑物目标整体中去，然后再输入网络训练。

对于两种网络各自的提取效果而言，在训练次数较少时两者分类精度相差不大，表明浅层网络已能学习到样本的主要特征；而当训练次数增加时，ResNet 网络的优势开始显现，它更深的网络结构在多次迭代时可以学习到更复杂的样本特征，因此在细节上 ResNet 网络提取效果相对更好。

4.4.2 可视化分析

为了检测网络训练的效果，将代价函数 loss 值和当前训练精度记录下来进

行跟踪。其中 loss 值为比较网络预测输出与标签计算得到的交叉熵代价函数值；训练精度通过比较预测与标签得出，将预测概率分配至具体类别，即概率较大的一方赋 1，另一方为 0，这样得到 one_hot 形式的预测值，再将两通道转至一维，得到类似标签形式的二值图像，再比较其与真实标签，计算相同像素的重叠度。图 4-5 为使用 tensorboard 可视化输出的代价函数和精度变化，明显可见一个可收敛的网络模型的 loss 值的减少和精度的增加：

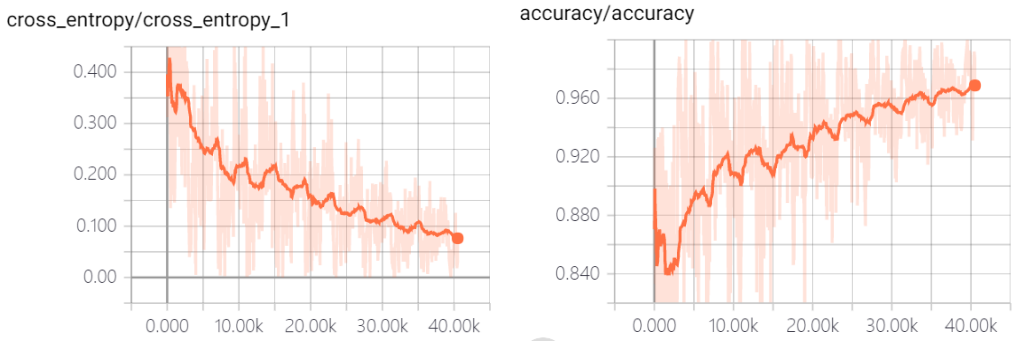


图 4-5（a）：浅层卷积神经网络参数跟踪

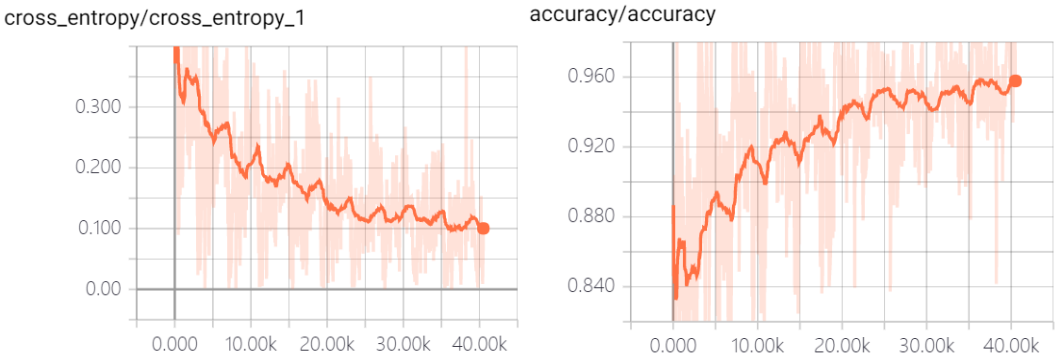
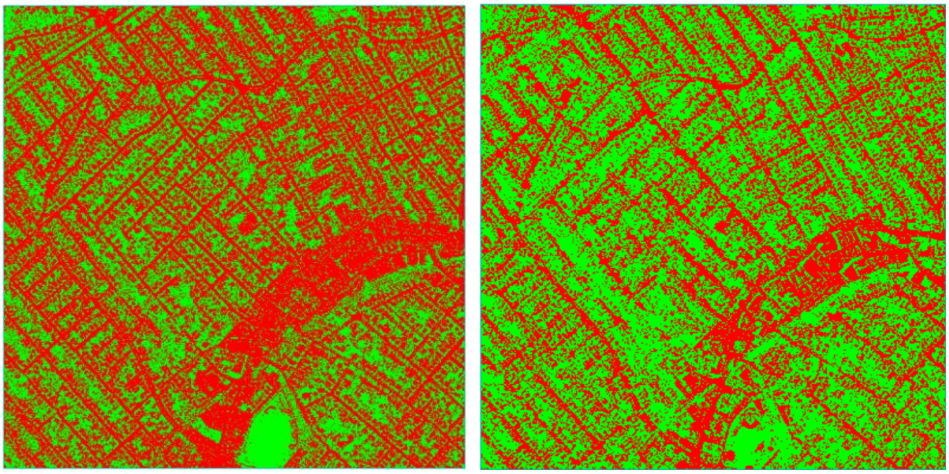


图 4-5（b）：ResNet 深度残差网络参数跟踪

4.4.3 与传统方法对比

为了测试深度学习方法的检测效果，另外使用基于 ISODATA 的非监督分类器以及基于支持向量机（SVM）的监督分类器对测试样本进行分类，并将之作为对照。图 4-6 展示了 ISODATA 和 SVM 方法对第一组数据集的提取结果。基于 ISODATA 的方法通过获取影像像元光谱统计信息，并以此计算特征空间的最小距离进行分类；基于支持向量机的方法通过建立最优超平面，并使得类别样本距离最大化进行分类。由图 4-3 与图 4-6 三种分类方法的目视比较可以看出，同为计算机自动分类，基于 ISODATA 的统计分析方法特征单一且不能完全表征目标信息；基于支持向量机的方法需要人工选取样本，存在波动较大的人为误差并且特征依然单

调。两种方法都明显存在较多错误，而深度学习方法可以自动提取多层次的特征，因此确有较大优势。



（a）SVM　　（b）ISODATA

图 4-6：SVM 与 ISODATA 分类结果

4.4.4 精度评定

表 4-2 给出了三种分类器的具体精度数值，其中总体精度为正确像元数与像元总数的比值，错分误差为错分为建筑的像元数与分类后建筑像元总数的比值，漏分误差为未分为建筑而实为建筑的像元数与实际建筑像元总数的比值。由表可见在忽略 ISODATA 参数调整及 SVM 监督样本的选择等人为因素影响的情况下，基于深度学习的建筑提取方法精度远高于 ISODATA、SVM 等基于统计学习的建筑物提取方法，此外，ResNet 网络的精度要高于自搭建 CNN 网络，并且随 ResNet 层数的增加精度也在增加，证明在足够大的训练样本支持下，深层网络确实能够更好地表征目标特点。

表 4-2：各分类器预测精度

分类器	总体精度	错分误差	漏分误差
ISODATA	65.13%	42.59%	47.44%
SVM	74.20%	32.44%	36.91%
CNN	83.23%	9.55%	19.22%
ResNet_50	87.37%	6.68%	15.98%
ResNet_101	90.70%	6.22%	14.69%
ResNet_152	91.26%	5.93%	13.32%

总结与展望

本文主要介绍了基于深度学习的高分辨率遥感影像建筑物提取方法，分别使用自搭建的浅层卷积神经网络和 ResNet 深度残差网络训练及预测提取建筑目标，具体完成了以下工作：

（1）收集并整理了国内外遥感影像建筑物提取的方法，介绍了国内外基于深度学习方法的最新研究成果；

（2）介绍了深度学习的发展历史和基本原理，重点介绍了基本卷积神经网络和深度残差网络的基本结构及其特点；

（3）介绍了使用深度学习（卷积神经网络）提取遥感影像建筑物目标的方法，详细阐述了网络训练过程中的激活函数、损失函数及优化函数；

（4）设计并完成了基于浅层卷积神经网络和深度残差网络的遥感影像建筑物提取工作，包括数据预处理、网络搭建、训练参量的选择、网络训练以及模型的完成导出；

（5）使用训练好的模型对测试数据集进行测试并与传统 ISODATA 和 SVM 提取方法进行对比并计算精度，得出深层 ResNet>浅层 CNN>SVM 方法>ISODATA 方法的基本结论；

（6）使用基于 python 语言的 tensorflow 深度学习框架完成全部实验，为后续更深入的研究积累经验。

同时，由于时间和研究水平的限制，本文也还存在一些不足有待改进：

（1）目前业内已经提出的深度学习模型多种多样，但本文只讨论了适用于图像检测的简单卷积神经网络和比较前沿的深度残差网络，其他卷积网络模型如 GoogleNet、AlexNet 以及其他深度网络模型如循环神经网络、深度信念网络其实也可用于影像识别，但具体效果如何需要更多更全面的实验；

（2）本文讨论了卷积网络的层次结构及损失函数、优化器等网络参量对网络模型的影响，但由于时间限制并没有仔细验证所有可能的参量组合，因此可能错过对本实验更有效的方案；

（3）本文仅采用了 ISODATA 方法与 SVM 支持向量机方法作为对比实验，参照实验数量较少；

（4）本文使用的原始数据集并没有进行过多预处理工作，而对比原始影像、

标签与模型预测结果，可以发现预测中漏分的地方绝大多数在影像中为建筑物的阴影，因此如果在数据预处理中加入建筑物的阴影特征，网络学习效果可能会有较大提升。

参考文献

[1] 吴炜，骆剑承，沈占锋,等．光谱和形状特征相结合的高分辨率遥感图像的建筑物提取方法[J]．武汉大学学报(信息科学版)，2012，37(7):800-805.

[2] 吕凤华，舒宁，龚龑,等．利用多特征进行航空影像建筑物提取[J]．武汉大学学报(信息科学版)，2017，42(5):656-660.

[3] Hofmann P. Detecting informal settlements from IKONOS image data using methods of object oriented image analysis-an example from Cape Town (South Africa) [J]. Jürgens, C.(Ed.)：Remote Sensing of Urban Areas/Fernerkundung in urbanen Räumen，2001：41-42.

[4] Mnih V. Machine Learning for Aerial Image Labeling[J]. Doctoral, 2013.

[5] Maggiori E, Tarabalka Y, Charpiat G, et al. Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification[J]. IEEE Transactions on Geoscience & Remote Sensing, 2016, PP(99):1-13.

[6] Saito S, Aoki Y. Building and road detection from large aerial imagery[C]// Image Processing: Machine Vision Applications VIII. 2015:1814–1821.

[7] 刘大伟，韩玲，韩晓勇．基于深度学习的高分辨率遥感影像分类研究[J]．光学学报，2016(4):298-306.

[8] 陈文康．基于深度学习的农村建筑物遥感影像检测[J]．测绘，2016，39(5):227-230.

[9] Marcu A, Leordeanu M. Dual Local-Global Contextual Pathways for Recognition in Aerial Imagery[J]. 2016.

[10] 张庆辉，万晨霞．卷积神经网络综述[J]．中原工学院学报，2017，28(3):82-86.

[11] 尹宝才，王文通，王立春．深度学习研究综述[J]．北京工业大学学报，2015(1):48-59.

[12] Sun Y, Wang X, Tang X. Deep Learning Face Representation from Predicting 10,000 Classes[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2014:1891-1898.

[13] 李彦冬，郝宗波，雷航．卷积神经网络研究综述[J]．计算机应用，2016，36(9):2508-2515.

[14] McCulloch, Warren S, Pitts, et al. A logical calculus of the ideas immanent in nervous activity[J]. Bulletin of Mathematical Biology, 1990, 52(1–2):99-115.

[15] Boureau Y, Roux N L, Bach F, et al. Ask the locals: Multi-way local pooling for image recognition[J]. 2011, 58(11):2651-2658.

[16] Y-Lan Boureau, Jean Ponce, Yann LeCun. A theoretical analysis of feature pooling in visual recognition. International Conference on Machine Learning, 2010, 32(4):111-118

[17] Y-Lan Boureau, Francis Bach, Yann LeCun, et al. Learning mid-level features for recognition//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, USA, 2010: 2559-2566

[18] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks[C]// International Conference on Neural Information Processing Systems. Curran Associates Inc. 2012:1097-1105.

[19] Sainath T N, Mohamed A R, Kingsbury B, et al. Deep convolutional neural networks for LVCSR[C]// IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2013:8614-8618.

[20] He, Kaiming, and Jian Sun. "Convolutional neural networks at constrained time cost." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

[21] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[J]. 2015:770-778.

致谢

本次研究是在崔卫红老师的耐心指导和严格监督下完成的，非常感谢崔老师在毕业设计各阶段所给予的耐心指引和督促。同时，由于实验对设备要求较高，崔老师尽可能地给我们提供理想的研究设施，让我们能够方便地实现各种想法和灵感。在老师的指导下，我们才能够顺利地完成此次毕业设计，在此需向崔老师致以最衷心的感谢。

此外，非常感谢两位师姐对我们的付出。熊师姐在实验前期给予了我非常大的帮助，让我能够快速理解并整理好自己的实验思路；杨师姐为我们的实验设备贡献了很多，使我们研究方案的实施得到保障。

感谢一起参与崔老师深度学习课题的几位伙伴，灵感的碰撞使我们的毕设工作充满乐趣。

感谢学校和家人的支持，让我有信心和毅力完成学业。