

Broadview[®]
www.broadview.com.cn

深度学习私房菜

跟着案例学TensorFlow

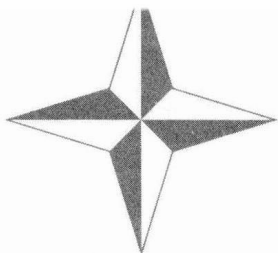
程世东◎编著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



深度学习私房菜

跟着案例学TensorFlow



程世东◎编著

电子工业出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书通过案例讲解如何使用 TensorFlow 解决深度学习的实际任务，每章都包含 TensorFlow 1.x 和 2.0 的代码实现。全书共分 10 章，主要讲解卷积神经网络、LSTM、Seq2Seq、Transformer、BERT、文本卷积、GBDT、FM、FFM、Dlib、MTCNN、VGG、AlphaGo / AlphaZero、BiLSTM、DQN、Gym、GAN 等技术，包含的项目有 CIFAR-100 图像分类、彩票预测、古诗生成、推荐系统、广告点击率预测、人脸识别、中国象棋、汉字 OCR、FlappyBird 和超级马里奥、人脸生成。

本书假设读者具有适当的 Python 编程基础和深度学习基础（比如梯度下降、反向传播等知识）。全书以案例的方式讲解涉及的知识，包括理论、算法和解决思路，适合相关专业的大学生或研究生，以及对深度学习感兴趣的读者参考阅读。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目（CIP）数据

深度学习私房菜：跟着案例学 TensorFlow / 程世东编著. —北京：电子工业出版社，2019.8

ISBN 978-7-121-36499-0

I. ①深…II. ①程…III. ①人工智能—算法—研究 IV. ①TP18

中国版本图书馆 CIP 数据核字 (2019) 第 089264 号

策划编辑：郑柳洁

责任编辑：葛 娜

印 刷：三河市良远印务有限公司

装 订：三河市良远印务有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×980 1/16 印张：30.25 字数：691 千字

版 次：2019 年 8 月第 1 版

印 次：2019 年 8 月第 1 次印刷

定 价：128.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：(010) 51260888-819, faq@phei.com.cn。

前言

在过去的几年中，人工智能和深度学习是一个不断被提及的话题，最令大众熟知的恐怕就是 AlphaGo 与柯洁、李世石的围棋大战了。而最近，人工智能 OpenAI Five 在 DOTA 2 的比赛中击败了世界冠军 OG，人工智能的发展总会给人带来惊喜。除了上述“大事件”，人工智能和深度学习早就深入我们的生活当中，比如无人驾驶汽车、人脸识别，或者订外卖时 App 给我们做的推荐。在经历了互联网、移动互联网的浪潮之后，可以说现在我们已经步入了人工智能的时代。

我第一次接触机器学习和深度学习时就被深深地吸引了，感觉自己就像一块海绵被投入到大海里，看书、看视频、看源码，学习新的知识使我感到非常兴奋。相信此刻的你也跟当年学习的我的状态是一样的吧，本书将我对深度学习的理解、从开源社区学到的知识分享给大家，希望能对你的学习提供一些帮助。

本书以案例的形式，讲解各种深度学习理论和相应场景的实践，包含 TensorFlow 1.x 和 TensorFlow 2.0 的代码实现。全书共分 10 章。第 1 章讲解了卷积神经网络理论知识，第 2 章讲解了如何进行 CIFAR-100 图像分类实践。第 3 章介绍了循环神经网络（RNN 和 LSTM），以及在彩票预测和古诗生成上的实践。在第 3 章的最后，介绍了 Seq2Seq、Transformer 和 BERT 模型。第 4 章以电影推荐系统为例，分享了推荐系统的实现。第 5 章介绍了广告点击率预测。第 6 章讲解了人脸识别的实践，包含使用 OpenCV、dlib 和 MTCNN 进行人脸检测，使用 dlib、FaceNet 和 VGG16 等方式提取人脸特征，然后讨论了比较人脸特征的几种方式。最后，使用上述技术实现了一个在视频中找人的应用。第 7 章分析了 AlphaGo 和 AlphaZero 的论文原文，讲解了蒙特卡罗树搜索（MCTS）和神经网络的结构，并且通过实现中国象棋复现了 AlphaZero。第 8 章介绍了 OCR 在汉字识别上的应用，并且讲解了 BiLSTM 的多种类型，最后实现了一个端到端的汉字识别网络。第 9 章介绍了 DQN 算法，用于玩 *Flappy Bird*。然后介绍了 OpenAI Baselines 和 Gym 的用法，并用于玩“超级马里奥”。最后介绍了 OpenAI 提出的具有好奇心的强化学习算法。第 10 章讲解了生成对抗网络（GAN）及衍生的变种 DCGAN、WGAN 和 WGAN-GP 算法，用于人脸生成的实践。最后介绍了 PG-GAN 和 TL-GAN 的理论。

限于篇幅，以及作者能力有限，书中难免有错漏之处，本书仅仅是将作者掌握的知识做了总结与分享。当然，这些知识不属于我个人，首先要感谢那些工作在人工智能第一线的科学家们，是他们将研究成果公布出来，让大家可以阅读论文和博客。更要感谢开源社区的贡献者们，使我

们可以阅读源码参考学习。还要感谢同样喜欢技术分享的人们，我所能做的跟他们一样，就是将分享的“火炬”继续传递下去。书中在引用时都会给出引用的出处，在这里一并表示感谢。

写作本书的过程，是对自己所学知识的一次梳理，回过头来重新审视自己对某些知识的理解，又是一次成长。同时，写作的过程又是孤独的、寂寞的，有时觉得自己就像在山洞里练剑一样。感谢我的妻子对我的理解和支持，在写作期间，她在国内带着孩子，听从了我“在交稿前不要来打扰”的安排。最要感谢的是我的妈妈，是她培养了我学习与钻研的习惯。最后还要感谢电子工业出版社的郑柳洁和葛娜老师，她们对本书的出版和编辑提供了很多专业性的指导和帮助，没有她们的付出，本书无法与大家见面。

希望本书的内容能够为你提供帮助，权当抛砖引玉，为你的深度学习知识打下基础。

程世东

2019年4月于日本

读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），扫码直达本书页面。

- **提交勘误：**您对书中内容的修改意见可在【提交勘误】处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- **与读者交流：**在页面下方【读者评论】处留下您的疑问或观点，与其他读者一同学习交流。

页面入口：<http://www.broadview.com.cn/36499>

目录

1	卷积神经网络与环境搭建	1
1.1	概述	1
1.2	卷积神经网络	2
1.2.1	卷积层	3
1.2.2	修正线性单元	6
1.2.3	池化层	8
1.2.4	全连接层	8
1.2.5	softmax 层	9
1.2.6	LeNet-5 网络	9
1.3	准备开发环境	10
1.3.1	Anaconda 环境搭建	10
1.3.2	安装 TensorFlow 1.x	11
1.3.3	FloydHub 使用介绍	13
1.3.4	AWS 使用介绍	18
1.4	本章小结	26
2	卷积神经网络实践：图像分类	27
2.1	概述	27
2.2	卷积神经网络项目实践：基于 TensorFlow 1.x	27
2.2.1	数据预处理	28
2.2.2	网络模型	33
2.2.3	训练网络	39
2.3	卷积神经网络项目实践：基于 TensorFlow 2.0	41
2.3.1	TensorFlow 2.0 介绍	41
2.3.2	CIFAR-100 分类网络的 TensorFlow 2.0 实现	44
2.4	本章小结	60

3	彩票预测和生成古诗	61
3.1	概述	61
3.2	RNN	61
3.3	LSTM	63
3.4	嵌入矩阵	66
3.5	实现彩票预测	69
3.5.1	数据预处理	70
3.5.2	构建神经网络	71
3.5.3	训练神经网络	75
3.5.4	分析网络训练情况	83
3.5.5	生成预测号码	88
3.6	文本生成	93
3.7	生成古诗：基于 TensorFlow 2.0	96
3.7.1	数据预处理	96
3.7.2	构建网络	99
3.7.3	开始训练	102
3.7.4	生成古诗	102
3.8	自然语言处理	106
3.8.1	序列到序列	106
3.8.2	Transformer	108
3.8.3	BERT	112
3.9	本章小结	118
4	个性化推荐系统	119
4.1	概述	119
4.2	MovieLens 1M 数据集分析	120
4.2.1	下载数据集	120
4.2.2	用户数据	120
4.2.3	电影数据	122
4.2.4	评分数据	123
4.3	数据预处理	123
4.3.1	代码实现	124

4.3.2	加载数据并保存到本地	127
4.3.3	从本地读取数据	128
4.4	神经网络模型设计	128
4.5	文本卷积神经网络	130
4.6	实现电影推荐：基于 TensorFlow 1.x	131
4.6.1	构建计算图	131
4.6.2	训练网络	139
4.6.3	实现个性化推荐	144
4.7	实现电影推荐：基于 TensorFlow 2.0	154
4.7.1	构建模型	154
4.7.2	训练网络	166
4.7.3	实现个性化推荐	166
4.8	本章小结	169
5	广告点击率预估：Kaggle 实战	170
5.1	概述	170
5.2	下载数据集	170
5.3	数据字段的含义	171
5.4	点击率预估的实现思路	172
5.4.1	梯度提升决策树	172
5.4.2	因子分解机	172
5.4.3	场感知分解机	174
5.4.4	网络模型	175
5.5	数据预处理	176
5.5.1	GBDT 的输入数据处理	177
5.5.2	FFM 的输入数据处理	177
5.5.3	DNN 的输入数据处理	179
5.5.4	数据预处理的实现	180
5.6	训练 FFM	188
5.7	训练 GBDT	197
5.8	用 LightGBM 的输出生成 FM 数据	203
5.9	训练 FM	207

5.10	实现点击率预估：基于 TensorFlow 1.x	218
5.10.1	构建神经网络	219
5.10.2	训练网络	225
5.10.3	点击率预估	231
5.11	实现点击率预估：基于 TensorFlow 2.0	237
5.12	本章小结	245
6	人脸识别	246
6.1	概述	246
6.2	人脸检测	247
6.2.1	OpenCV 人脸检测	247
6.2.2	dlib 人脸检测	251
6.2.3	MTCNN 人脸检测	254
6.3	提取人脸特征	264
6.3.1	使用 FaceNet 提取人脸特征	264
6.3.2	使用 VGG 网络提取人脸特征	265
6.3.3	使用 dlib 提取人脸特征	272
6.4	人脸特征的比较	276
6.5	从视频中找人的实现	282
6.6	视频找人的案例实践	284
6.7	人脸识别：基于 TensorFlow 2.0	302
6.8	本章小结	303
7	AlphaZero / AlphaGo 实践：中国象棋	304
7.1	概述	304
7.2	论文解析	305
7.2.1	蒙特卡罗树搜索算法	307
7.2.2	神经网络	312
7.2.3	AlphaZero 论文解析	314
7.3	实现中国象棋：基于 TensorFlow 1.x	317
7.3.1	中国象棋着法表示和 FEN 格式	317
7.3.2	输入特征的设计	321
7.3.3	实现神经网络	323

7.3.4	神经网络训练和预测	327
7.3.5	通过自我对弈训练神经网络	330
7.3.6	自我对弈	334
7.3.7	实现蒙特卡罗树搜索：异步方式	340
7.3.8	训练和运行	353
7.4	实现中国象棋：基于 TensorFlow 2.0，多 GPU 版	354
7.5	本章小结	364
8	汉字 OCR	365
8.1	概述	365
8.2	分类网络实现汉字 OCR	365
8.2.1	图片矫正	366
8.2.2	文本切割	368
8.2.3	汉字分类网络	369
8.3	端到端的汉字 OCR：基于 TensorFlow 1.x	371
8.3.1	CNN 设计	372
8.3.2	双向 LSTM 设计	374
8.3.3	CTC 损失	385
8.3.4	端到端汉字 OCR 的网络训练	388
8.4	汉字 OCR：基于 TensorFlow 2.0	395
8.4.1	CNN 的实现	395
8.4.2	双向 LSTM 的实现	396
8.4.3	OCR 网络的训练	403
8.5	本章小结	406
9	强化学习：玩转 Flappy Bird 和超级马里奥	407
9.1	概述	407
9.2	DQN 算法	407
9.3	实现 DQN 玩 Flappy Bird：基于 TensorFlow 1.x	412
9.4	实现 DQN 玩 Flappy Bird：基于 TensorFlow 2.0	417
9.5	使用 OpenAI Baselines 玩超级马里奥	424
9.5.1	Gym	424
9.5.2	自定义 Gym 环境	426

- 9.5.3 使用 Baselines 训练 431
- 9.5.4 使用训练好的智能体玩游戏 437
- 9.5.5 开始训练马里奥游戏智能体 438
- 9.6 具有好奇心的强化学习算法 443
- 9.7 本章小结 444
- 10 生成对抗网络实践：人脸生成 445**
- 10.1 概述 445
- 10.2 GAN 446
- 10.3 DCGAN 447
 - 10.3.1 生成器 448
 - 10.3.2 判别器 449
- 10.4 WGAN 449
- 10.5 WGAN-GP 451
 - 10.5.1 WGAN-GP 算法 451
 - 10.5.2 训练 WGAN-GP 生成人脸：基于 TensorFlow 1.x 452
 - 10.5.3 训练 WGAN-GP 生成人脸：基于 TensorFlow 2.0 462
- 10.6 PG-GAN 和 TL-GAN 469
- 10.7 本章小结 473

1

卷积神经网络与环境搭建

1.1 概述

在计算机视觉和人工智能领域，人们希望智能体能够像人一样，不仅能看到图像，更要看懂图像，比如目标检测和物体识别。这就需要训练智能体对物体进行分类，并能够识别出这些分类物体，甚至能够进行人脸识别（区分出是否是同一人）。当能够识别出图像中的物体之后，智能体还要能够理解图像所包含的含义，比如图像中一辆车行驶在马路上、一只猫在捉老鼠等。更进一步，人们还要求智能体能够预测图像中物体下一步要做什么事情，这就涉及人体骨骼关键点检测等技术。2019年2月，由李飞飞等研究者发表的论文 *Peeking into the Future: Predicting Future Person Activities and Locations in Videos* 中提出了一种端到端的模型，可以用来预测行人的未来活动和行动轨迹，论文地址是 <https://arxiv.org/abs/1902.03748>。

总的来说，计算机视觉应用的最终目标是使智能体能够像人一样看待、看懂这个世界。视觉是人类从世界获取信息的主要途径，也是智能体了解世界的途径。我们希望自动驾驶汽车能够识别出人、道路等其他物体，这一切都可以使用卷积神经网络实现。

20世纪90年代末，Yann LeCun研究的卷积神经网络开始流行，LeNet-5网络能够用来识别手写数字。自2010年以来，每年的ImageNet大规模视觉识别挑战赛（ILSVRC），很多研究团队都在ImageNet的超大图像数据集（含有1000个标签类别、超过100万张的训练图像）上评估验证他们的最新算法。2012年，基于卷积神经网络的AlexNet在ImageNet上实现了15%的错误率，这一成绩轻松击败了第二名26%的错误率。此后在历年的比赛中，更多的基于卷积神经网络的优秀网络接踵而来，比如VGG（2014年）、GoogLeNet（2014年）、ResNet（2015年）等。

本章将讲解这一重要的网络——卷积神经网络的原理。

1.2 卷积神经网络

卷积神经网络（Convolutional Neural Network, CNN）跟传统的神经网络有些类似，它们都由可学习的神经元节点组成，包含权重和偏置。传统神经网络的各种计算在卷积神经网络中都一样，比如前向传播、反向传播等。

传统神经网络的结构如图 1.1 所示。可以看出，网络的每一层都是以全连接的形式连接起来的。而卷积神经网络与传统神经网络相比，在结构上做了改进，卷积神经网络通常包含以下几层：

- 卷积层（Convolutional Layer）
- 修正线性单元（Rectified Linear Unit, ReLU）
- 池化层（Pooling Layer）
- 全连接层（Fully-Connected Layer）
- softmax 层

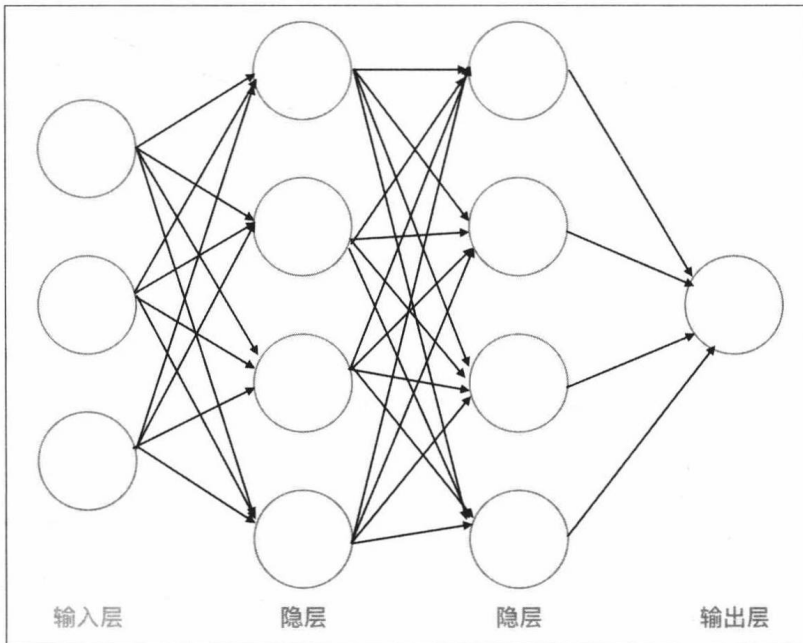


图 1.1 传统神经网络的结构

1.2.1 卷积层

卷积层是整个卷积神经网络的核心，这里就不介绍卷积的计算了，我们来直观感受卷积是如何工作的，如图 1.2 所示。

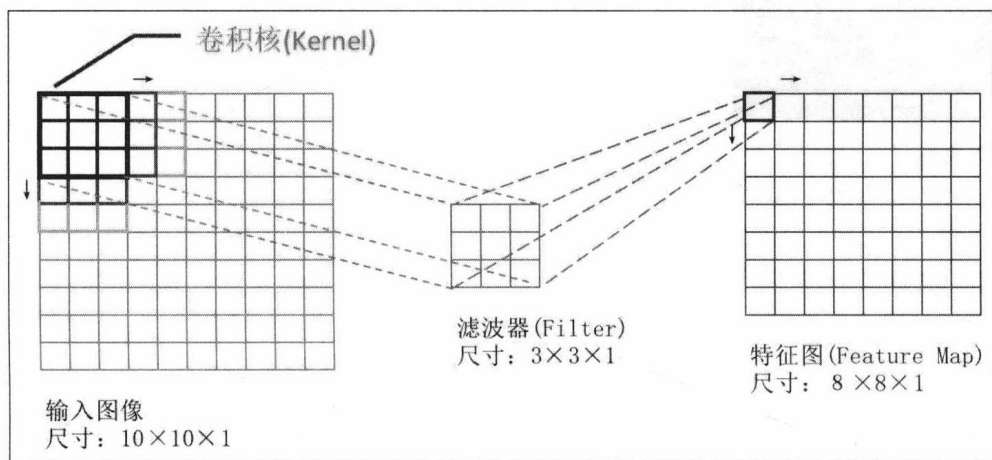


图 1.2 卷积过程

假设输入是一张灰度图图像，那么输入的深度就是 1，如图 1.2 所示输入图像尺寸是 $10 \times 10 \times 1$ 。现在选择一个尺寸（通常选择 3×3 或 5×5 ，如图 1.2 左侧所示），从输入图像左上角开始水平移动（水平方向到头的话，垂直移动一步后再次水平移动），直到滑过整个输入图像，最终到达图像右下角。这个在输入图像上选择的尺寸（Kernel Size），叫作卷积核（Kernel）。在这个过程中，每次移动都将该尺寸的输入图像与相同尺寸（ 3×3 或 5×5 ）的卷积参数做运算，这个卷积参数叫作滤波器（Filter，如图 1.2 中间所示），并得到一个输出（如图 1.2 右侧所示）。每移动一步，都会得到一个输出，当滤波器与整个输入图像运算完之后，最终得到的输出就是对输入图像提取出的特征，叫作特征图（Feature Map）。这就是卷积层的功能，在整个移动（卷积）过程中，卷积层的权重（滤波器，Filter Size）是不变的，即权值共享。

卷积核在输入图像上移动时，每移动一步的长度叫作步幅（Stride，如图 1.3 左侧所示），这个长度表示移动的像素数，最后输出的特征图尺寸跟移动的步幅相关。如果步幅是 1 的话，则输出图像的尺寸和输入图像基本相同。为什么是基本相同呢？这取决于卷积核在图像边界上的处理方式。有两种方式：一种是卷积核在图像边界范围内移动，称作有效填充（Valid Padding），如图 1.4 所示；另一种是在图像边界外用 0 填充，卷积核在填充 0 的位置开始移动，这样输出图像的尺寸和输入图像是一致的，称作相同填充（Same Padding），如图 1.5 所示。当步幅为 2 时，输出图像的尺寸是输入图像的一半，相当于降采样。

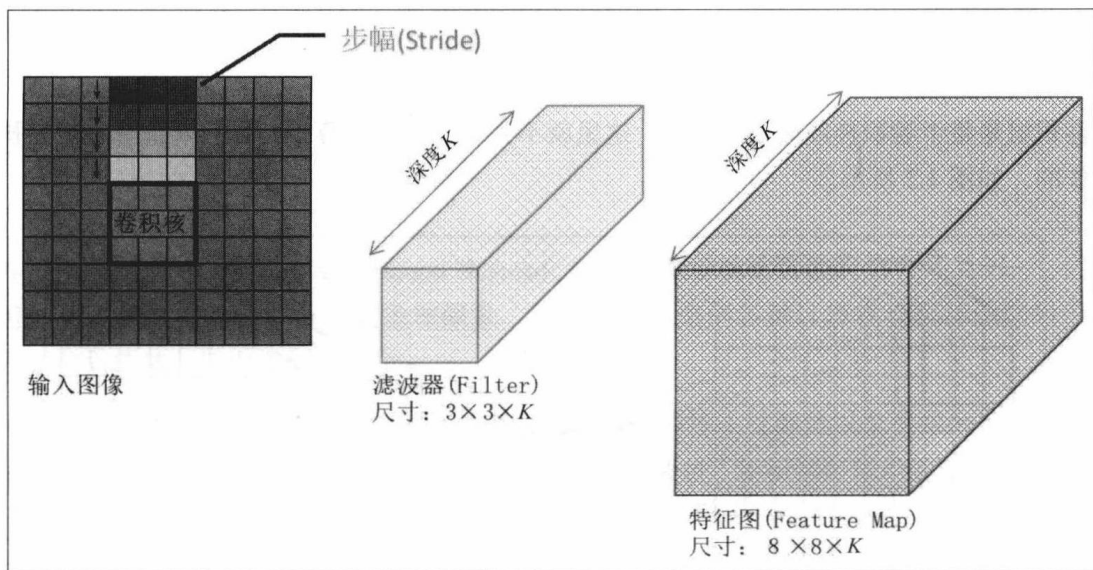


图 1.3 滤波器的深度

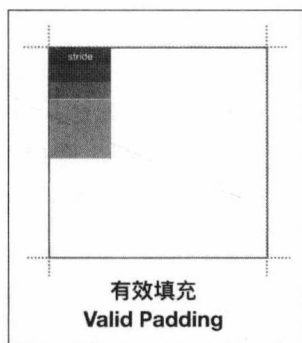


图 1.4 有效填充

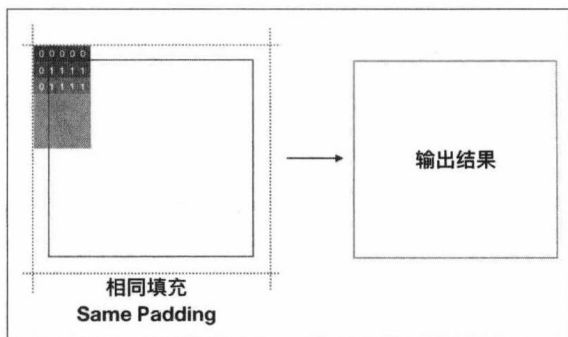


图 1.5 相同填充

现在来看看更常见的情况，通常我们会对输入图像提取多个特征，这代表我们要使用多个滤波器。图 1.3 中间所示的滤波器尺寸是 $3 \times 3 \times K$ ，表示卷积层使用 K 个滤波器，输出深度为 K (K 维) 的特征图（也可以理解成是一个具有 K 个颜色通道的图像，如图 1.3 右侧所示），每一维特征图都是由输入和训练的卷积参数（滤波器）做点积，然后加上偏置得到的，训练的每一维滤波器都可以用来提取图像的不同特征。

卷积神经网络就是使用权值共享的卷积层来代替全连接层的。图 1.2 所示的演示只是经过了一层卷积，考虑传统的神经网络通常会使用多个隐层，在使用多个卷积层之后，整个网络看起来如图 1.6 所示。

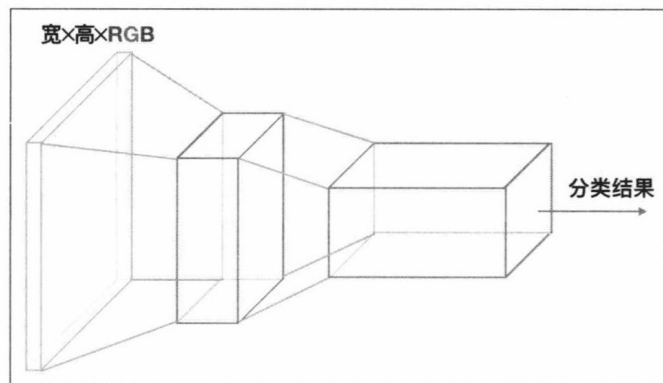


图 1.6 卷积神经网络金字塔结构

图 1.6 左侧是输入图像，经过卷积之后，会逐步改变图像的维度，宽度和高度逐渐变小，并且深度不断变大，这样深度的信息通常可以表示复杂的抽象特征。如果在网络的最后使用分类器，那么根据卷积层学到的图像特征最终会得到该图像的分类结果。

对于卷积层，我们只需要设计好卷积核的尺寸、滤波器的个数（输出的深度）、移动的步幅以及填充方式，然后将输入图像交给卷积层，输出的特征图即是提取到的特征。第一层卷积层提取到输入图像的特征作为输入，再由第二层卷积层继续提取特征，接着后面的卷积层继续提取，直到网络的最后一层。这样越往后的卷积层提取到的特征越复杂。最开始的卷积层学习到的是最基本的特征，例如直线、斜线、弧线。可以将这个过程理解成放大镜，最开始看到的是最高倍放大的结果，即图中的某个边缘。第二层卷积层在基本特征的基础上，就好比稍微缩小了放大镜的倍率，看到的图像稍微清晰一些，是由直线、斜线和弧线组成的某种图像，例如矩形、圆形、多边形。再继续，下一层卷积层提取到的特征更高级一些，好比再次缩小了放大镜的倍率，好像看到了某种物体的一部分，例如眼睛、鼻子、耳朵……直到最后一层卷积层，提取到更抽象的特征，能够通过识别到的眼睛、鼻子、耳朵确认输入图像是否是某种人脸或者动物或者任意我们要识别的东西，从而得到分类结果。

卷积层的好处首先就是权值共享。由全连接的方式变成了小范围的局部连接，并且用局部连接的卷积核滑过整个输入图像的权值是共享的。这大大减少了要学习的参数数量，也减小了网络的复杂度，降低了过拟合风险。

另一个好处就是带来了平移不变性。比如我们要在图像中识别一只狗，不管这只狗是大是小、是黄色的还是黑色的、是站着的还是躺着的，也不管在图像的哪个位置，只要卷积神经网络学习到狗的特征之后，无论在什么条件下都能够识别出图像中的狗，而不用训练网络单独学习狗在左边的情况、狗在右边躺下的情况……否则，工作量就大了。一旦出现没有想到的情况，就识别失败了。实际上，这就是人们手工制定特征和神经网络学习特征的差别，泛化能力不同。所以，卷积神经网络的平移不变性带来了很好的泛化能力，提高了网络的鲁棒性。

1.2.2 修正线性单元

在卷积层之后，通常要加入非线性激活函数，目的是通过引入非线性，使得神经网络能够逼近任意函数，解决复杂的现实问题。

关于激活函数，有很多种选择，比如 Sigmoid 函数。

$$\text{公式: } f(z) = \frac{1}{1 + e^{-z}}$$

$$\text{导数: } f'(z) = f(z)(1 - f(z))$$

Sigmoid 函数的曲线如图 1.7 所示（实线）。

Sigmoid 函数也叫作 Logistic 函数，特点是能够将输入映射到 (0,1)。输出值在 0~1 之间，符合概率输出的意义，适合用于解决二分类问题。但是 Sigmoid 函数有缺陷，我们来看看反向传播时 Sigmoid 函数的导数曲线，如图 1.7 所示（虚线）。

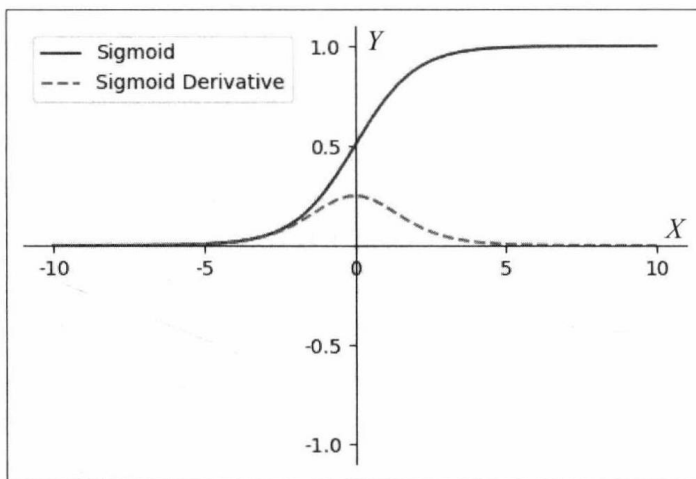


图 1.7 Sigmoid 函数的曲线和 Sigmoid 函数的导数曲线

Y 轴是梯度值，从 Sigmoid 函数的导数曲线可以看出，梯度在饱和区域非常平缓，趋近于 0。这意味着，当网络层数较多时，Sigmoid 函数在不断的反向传播中梯度会逐渐变小，这导致更新网络参数异常缓慢，网络难以收敛，这就是梯度消失问题。这可以理解成梯度趋近于 0 的神经元都“死”了，不再起任何作用，一旦梯度消失蔓延下去，整个网络也不再有效。梯度趋近于 0，会造成大量信息丢失，是训练深层网络必须要解决的问题。另外，Sigmoid 函数和反向传播计算量大，包含指数和除法等运算。

现在介绍另一个激活函数，即线性整流函数，又称修正线性单元（ReLU）。

$$\text{公式: } f(x) = \max(0, x)$$