

# 卷积神经网络

维基百科，自由的百科全书

卷积神经网络（Convolutional Neural Network, **CNN**）是一种前馈神经网络，它的人工神经元可以响应一部分覆盖范围内的周围单元，<sup>[1]</sup>对于大型图像处理有出色表现。

卷积神经网络由一个或多个卷积层和顶端的全连通层（对应经典的神经网络）组成，同时也包括关联权重和池化层（pooling layer）。这一结构使得卷积神经网络能够利用输入数据的二维结构。与其他深度学习结构相比，卷积神经网络在图像和语音识别方面能够给出更好的结果。这一模型也可以使用反向传播算法进行训练。相比较其他深度、前馈神经网络，卷积神经网络需要考量的参数更少，使之成为一种颇具吸引力的深度学习结构<sup>[2]</sup>。

## 目录

定义	
概览	
发展	
结构	
<span> </span> <span> </span> 卷积层	
<span> </span> <span> </span> 线性整流层	
<span> </span> <span> </span> 池化层	
<span> </span> <span> </span> 损失函数层	
应用	
<span> </span> <span> </span> 影像识别	
<span> </span> <span> </span> 视频分析	
<span> </span> <span> </span> 自然语言处理	
<span> </span> <span> </span> 药物发现	
<span> </span> <span> </span> 围棋	
微调（fine-tuning）	
经典模型	
可用包	
参考	

## 定义

“卷积神经网络”表示在网络采用称为卷积的数学运算。卷积是一种特殊的线性操作。卷积网络是一种特殊的神经网络，它们在至少一个层中使用卷积代替一般矩阵乘法

## 概览

## 发展

# 结构

## 卷积层

卷积层是一组平行的特征图（feature map），它通过在输入图像上滑动不同的卷积核并运行一定的运算而组成。此外，在每一个滑动的位置上，卷积核与输入图像之间会运行一个元素对应乘积并求和的运算以将感受野内的信息投影到特征图中的一个元素。这一滑动的过程可称为步幅  $Z_s$ ，步幅  $Z_s$  是控制输出特征图尺寸的一个因素。卷积核的尺寸要比输入图像小得多，且重叠或平行地作用于输入图像中，一张特征图中的所有元素都是通过一个卷积核计算得出的，也即一张特征图共享了相同的权重和偏置项。

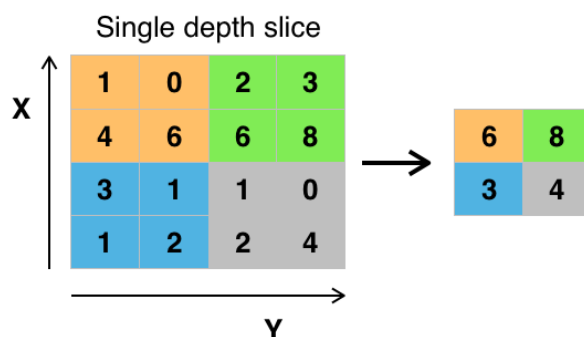
## 线性整流层

线性整流层（Rectified Linear Units layer, ReLU layer）使用线性整流（Rectified Linear Units, ReLU） $f(x) = \max(0, x)$  作为这一层神经的激励函数（Activation function）。它可以增强判定函数和整个神经网络的非线性特性，而本身并不会改变卷积层。

事实上，其他的一些函数也可以用于增强网络的非线性特性，如双曲正切函数  $f(x) = \tanh(x)$ ,  $f(x) = |\tanh(x)|$ ，或者 Sigmoid 函数  $f(x) = (1 + e^{-x})^{-1}$ 。相比其它函数来说，ReLU 函数更受青睐，这是因为它可以将神经网络的训练速度提升数倍<sup>[3]</sup>，而并不会对模型的泛化准确度造成显著影响。

## 池化层

池化（Pooling）是卷积神经网络中另一个重要的概念，它实际上是一种形式的降采样。有多种不同形式的非线性池化函数，而其中“最大池化（Max pooling）”是最为常见的。它是将输入的图像划分为若干个矩形区域，对每个子区域输出最大值。直觉上，这种机制能够有效地原因在于，在发现一个特征之后，它的精确位置远不及它和其他特征的相对位置的关系重要。池化层会不断地减小数据的空间大小，因此参数的数量和计算量也会下降，这在一定程度上也控制了过拟合。通常来说，CNN 的卷积层之间都会周期性地插入池化层。



池化层通常会分别作用于每个输入的特征并减小其大小。当前最常用形式的池化层是每隔2个元素从图像划分出  $2 \times 2$  的区块，然后对每个区块中的4个数取最大值。这将会减少75%的数据量。

除了最大池化之外，池化层也可以使用其他池化函数，例如“平均池化”甚至“L2-范数池化”等。过去，平均池化的使用曾经较为广泛，但是最近由于最大池化在实践中的表现更好，平均池化已经不太常用。

由于池化层过快地减少了数据的大小，当前文献中的趋势是使用较小的池化滤镜，<sup>[4]</sup>甚至不再使用池化层。<sup>[5]</sup>

## 损失函数层

损失函数层（loss layer）用于决定训练过程如何来“惩罚”网络的预测结果和真实结果之间的差异，它通常是网络的最后一层。各种不同的损失函数适用于不同类型的任务。例如，Softmax 交叉熵损失函数常常被用于在 K 个类别中选出一个，而 Sigmoid 交叉熵损失函数常常用于多个独立的二分类问题。欧几里德损失函数常常用于结果取值范围为任意实数的问题。

# 应用

## 影像识别

卷积神经网络通常在影像识别系统中使用。

## 视频分析

相比影像识别问题，视频分析要难许多。CNN也常被用于这类问题。

## 自然语言处理

卷积神经网络也常被用于自然语言处理。CNN的模型被证明可以有效的处理各种自然语言处理的问题，如语义分析<sup>[6]</sup>、搜索结果提取<sup>[7]</sup>、句子建模<sup>[8]</sup>、分类<sup>[9]</sup>、预测<sup>[10]</sup>、和其他传统的NLP任务<sup>[11]</sup>等。

## 药物发现

卷积神经网络已在药物发现中使用。卷积神经网络被用来预测的分子与蛋白质之间的相互作用，以此来寻找靶向位点，寻找出更安全有效的潜在治疗方法。

## 围棋

卷积神经网络在计算机围棋领域也被使用。2016年3月，AlphaGo对战李世乭的比赛，展示了深度学习在围棋领域的重大突破。

## 微调 (fine-tuning)

---

卷积神经网络（例如Alexnet、VGG网络）在网络的最后通常为softmax分类器。微调一般用来调整softmax分类器的分类数。例如原网络可以分类出2种图像，需要增加1个新的分类从而使网络可以分类出3种图像。微调（fine-tuning）可以留用之前训练的大多数参数，从而达到快速训练收敛的效果。例如保留各个卷积层，只重构卷积层后的全连接层与softmax层即可。

## 经典模型

---

- LeNet
- AlexNet
- VGG
- GoogLeNet
- ResNet

## 可用包

---

- roNNie (<https://www.ronnie.ai>): 是一个简易入门级框架,使用Tensorflow 计算层.可于python下载 pip3 ronnie
- Caffe: Caffe包含了CNN使用最广泛的库。它由伯克利视觉和学习中心（BVLC）研发，拥有比一般实现更好的结构和更快的速度。同时支持CPU和GPU计算，底层由C++实现，并封装了Python和MATLAB的接口。
- Torch7 ([www.torch.ch](http://www.torch.ch))
- OverFeat
- Cuda-convnet
- MatConvnet

- Theano：用Python实现的神经网络包<sup>[12]</sup>
- TensorFlow
- Paddlepaddle([www.paddlepaddle.org](http://www.paddlepaddle.org) (<http://www.paddlepaddle.org/>))
- Keras

## 参考

---

1. Convolutional Neural Networks (LeNet) – DeepLearning 0.1 documentation. DeepLearning 0.1. LISA Lab. [31 August 2013].
2. Convolutional Neural Network. [2014–09–16].
3. Krizhevsky, A.; Sutskever, I.; Hinton, G. E. Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems. 2012, 1: 1097–1105. (原始内容存档于2015–02–16) .
4. Graham, Benjamin. Fractional Max–Pooling. 2014–12–18. arXiv:1412.6071 [cs.CV].
5. Springenberg, Jost Tobias; Dosovitskiy, Alexey; Brox, Thomas; Riedmiller, Martin. Striving for Simplicity: The All Convolutional Net. 2014–12–21. arXiv:1412.6806 [cs.LG].
6. Grefenstette, Edward; Blunsom, Phil; de Freitas, Nando; Hermann, Karl Moritz. A Deep Architecture for Semantic Parsing. 2014–04–29. arXiv:1404.7296 [cs.CL].
7. Learning Semantic Representations Using Convolutional Neural Networks for Web Search – Microsoft Research. [research.microsoft.com](http://research.microsoft.com). [2015–12–17].
8. Kalchbrenner, Nal; Grefenstette, Edward; Blunsom, Phil. A Convolutional Neural Network for Modelling Sentences. 2014–04–08. arXiv:1404.2188 [cs.CL].
9. Kim, Yoon. Convolutional Neural Networks for Sentence Classification. 2014–08–25. arXiv:1408.5882 [cs.CL].
10. Collobert, Ronan, and Jason Weston. "A unified architecture for natural language processing: Deep neural networks with multitask learning." Proceedings of the 25th international conference on Machine learning. ACM, 2008.
11. Collobert, Ronan; Weston, Jason; Bottou, Leon; Karlen, Michael; Kavukcuoglu, Koray; Kuksa, Pavel. Natural Language Processing (almost) from Scratch. 2011–03–02. arXiv:1103.0398 [cs.LG].
12. 深度网络：Theano项目主页。(<http://deeplearning.net/software/theano/>)

---

取自“<https://zh.wikipedia.org/w/index.php?title=卷积神经网络&oldid=55632314>”

---

本页面最后修订于2019年8月13日 (星期二) 07:10。

本站的全部文字在知识共享 署名–相同方式共享 3.0协议之条款下提供，附加条款亦可能应用。（请参阅使用条款）  
Wikipedia®和维基百科标志是维基媒体基金会的注册商标；维基™是维基媒体基金会的商标。  
维基媒体基金会是按美国国内税收法501(c)(3)登记的非营利慈善机构。