



图像处理与计算机视觉

第十三讲：深度学习

任课教师：寇旗旗

计算机科学与技术学院



目录

- 深度学习概述
- 深度学习常见场景
- 深度学习常用算法介绍
- 深度学习常用框架介绍
- TensorFlow入门
- 深度学习未来和展望

1.1 深度学习概述

- 深度学习(Deep Learning, DL)由Hinton等人于2006年提出，是机器学习(Machine Learning, ML)的一个新领域。
- 深度学习起源于人工神经网络，它的定义：通过组合低层特征形成更加抽象的高层特征或类别，从而从大量的输入数据中学习有效特征表示，并把这些特征用于分类、回归和信息检索的一种技术。
- 深度学习被引入机器学习使其更接近于最初的目标----人工智能（AI，Artificial Intelligence）。深度学习是学习样本数据的内在规律和表示层次，这些学习过程中获得的信息对诸如文字、图像和声音等数据的解释有很大的帮助。它的最终目标是让机器能够像人一样具有分析学习能力，能够识别文字、图像和声音等数据。

1.2 与机器学习关系

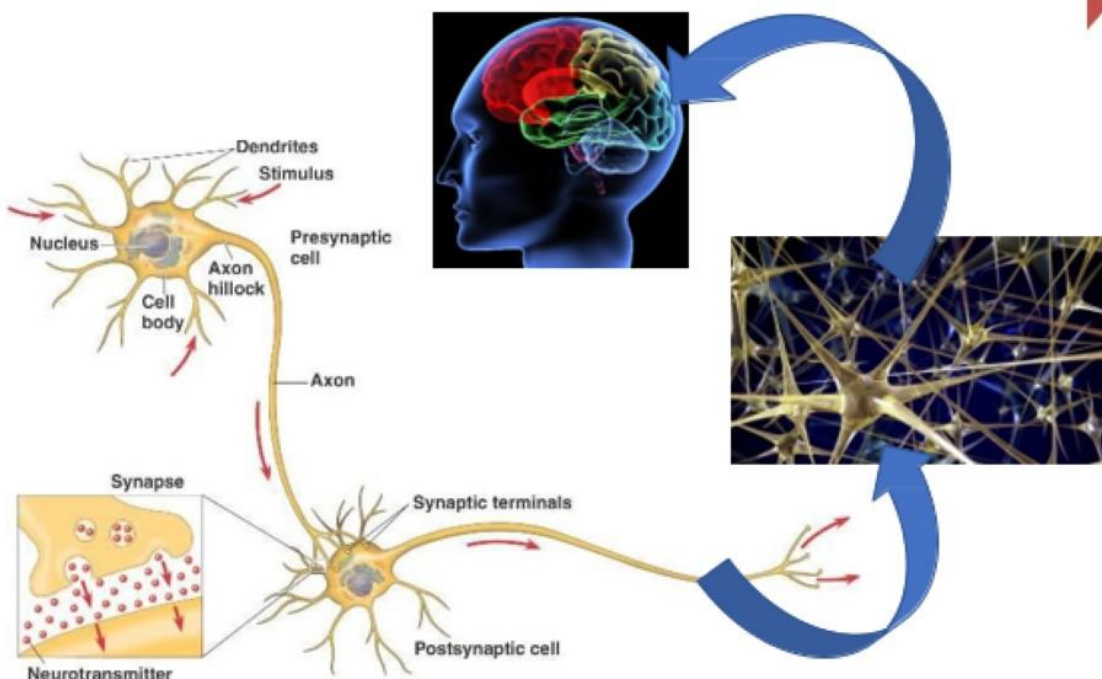


如上图，人工智能是最早出现的，也是最大、最外侧的同心圆；其次是机器学习，稍晚一点；最内侧，是深度学习，当今人工智能大爆炸的核心驱动。

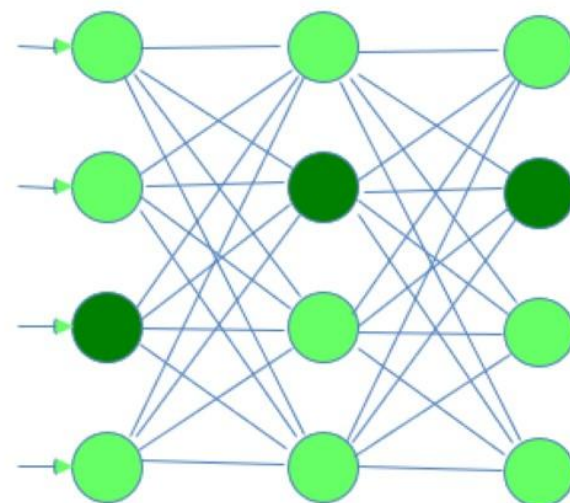
第三次人工智能热潮以深度学习为核心

深度学习属于连接主义学派，是机器学习非常重要的分支

浅层神经网络对复杂函数的表示能力有限 → 更多层，深层



人类大脑



人工神经网络

1.3 基本概念

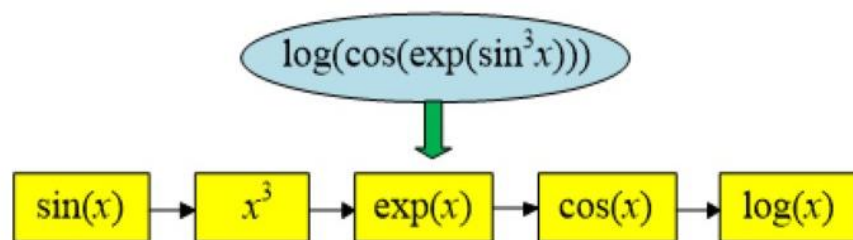
- 深度学习(Deep Learning, DL)：通过组合低层特征形成更加抽象的高层特征或类别，从而从大量的输入数据中学习有效特征表示，并把这些特征用于分类、回归和信息检索的一种技术。
- 人工神经网络：是一种模仿动物神经网络行为特征，进行分布式并行信息处理的算法数学模型。这种网络依靠系统的复杂程度，通过调整内部大量节点之间相互连接的关系，从而达到处理信息的目的。
- 深度神经网络：包含多个隐含层的神经网络。
- 模型：可以理解成算法和数据的合集。
- 模型训练：利用模型的算法，使用深度神经网络进行权值的训练，最终得出一个最优解。
- 模型预测：使用训练完成的模型进行预测，得出分类识别结果。

深度学习原理

- 人脑采用**分层机制**处理输入信息，**逐层提取特征并不断抽象**，特征的层次越高便越能体现输入的本质信息
- 例如，在理解句子时，**单词和句子为多对一**，**句子和语义为多对一**，**语义和意图仍为多对一**，但可能情况在逐渐减少，因而可以逐步确定句子的意图
- 深度学习是**模拟人脑的分层机制**，通过由下向上、由低级到高级逐层抽象的特征学习，来解决复杂的模式识别问题

深度学习原理

- 深度学习通过**深层非线性网络实现复杂函数的逼近**，具有强大的学习数据本质特征的能力



- 与单层结构相比，**多层结构可以用较少的参数表示复杂的函数**

深度学习原理

- 深度学习的实质是建立多隐层的机器学习模型，利用海量训练数据自动学习数据的本质特征，从而最终提升分类或回归的准确性



- 深度学习的显著特点
 - 结构的深度：5个乃至10个以上的隐层
 - 特征学习：逐层特征变换，使分类或回归更容易
- 与人工构造特征相比，自动学习特征更能够刻画数据丰富的内在信息和本质属性



阅读材料

- 神经网络

- <http://cs231n.github.io/neural-networks-1/>

- <http://cs231n.github.io/neural-networks-2/>

- <http://cs231n.github.io/neural-networks-3/>

- <http://cs231n.github.io/neural-networks-case-study/>

- 卷积神经网络

- <http://cs231n.github.io/convolutional-networks/>

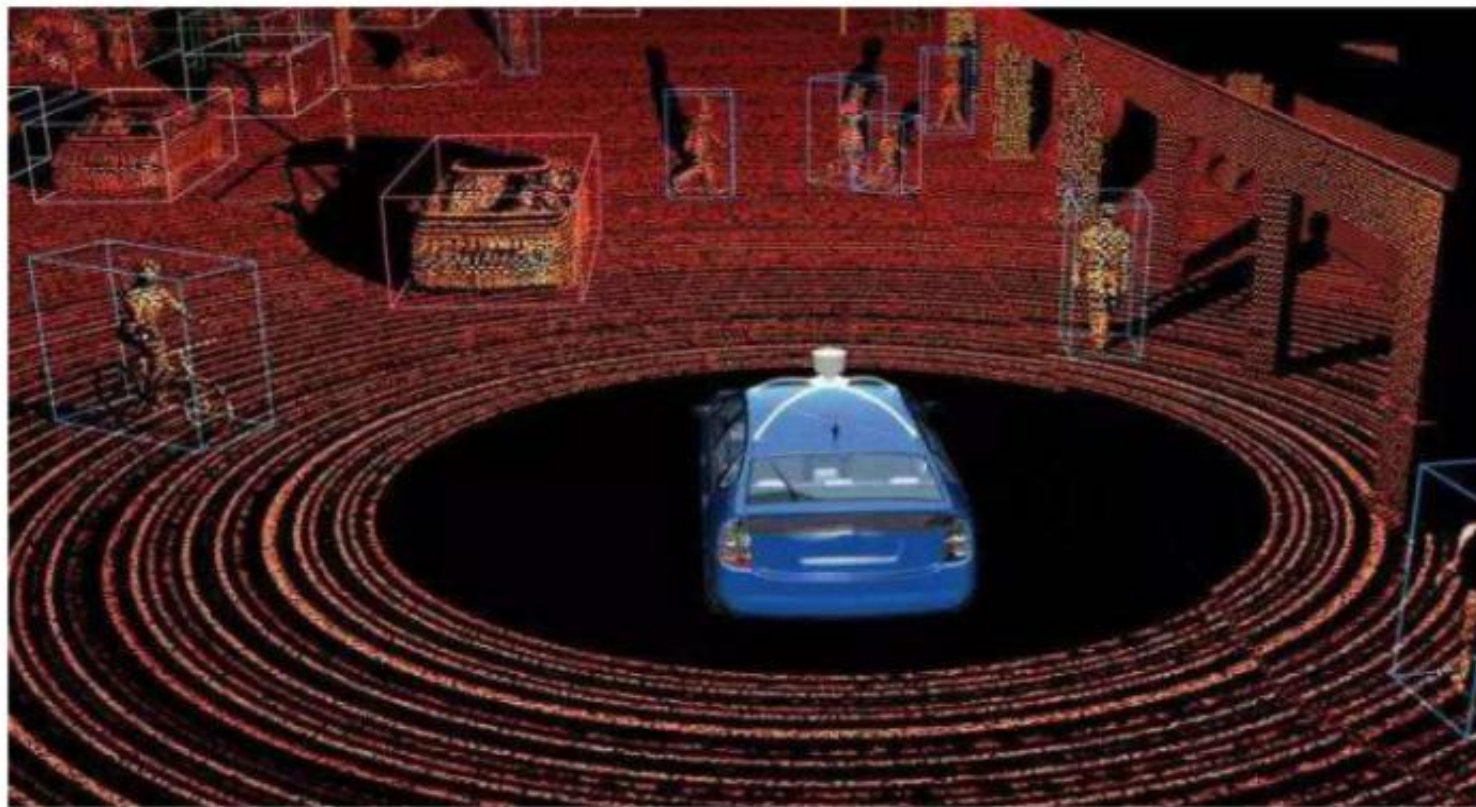
2 深度学习场景

当前深度学习使用的场景主要在无人驾驶，人脸识别，拍照购，智能客服，文字识别，语音识别，目标检测，图片分类等方面。



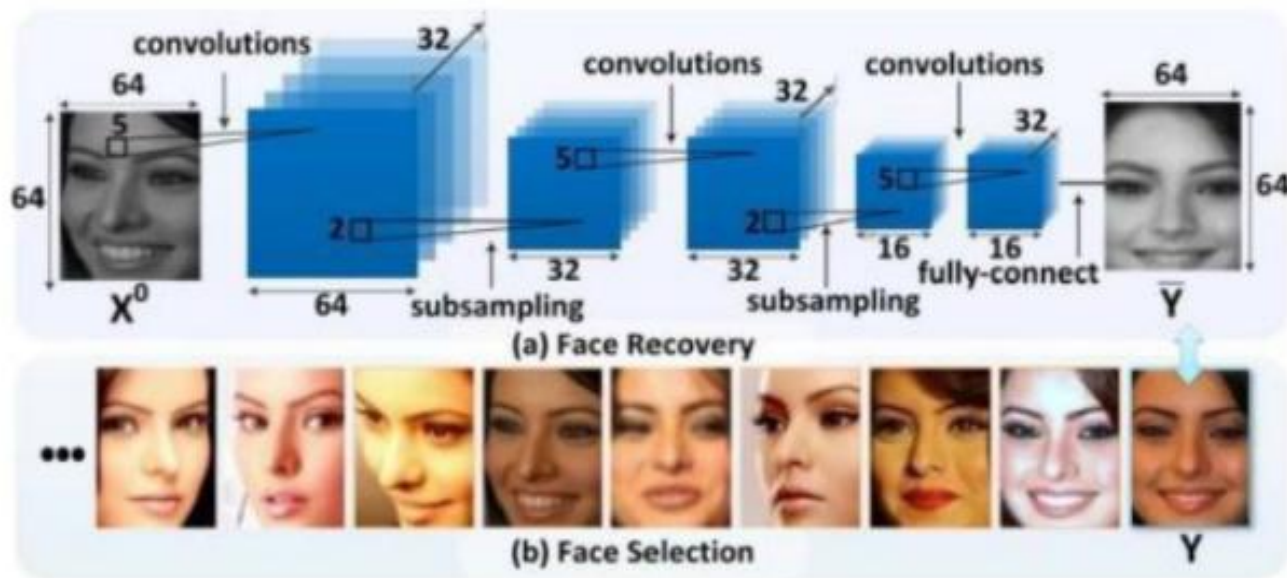
2.1 深度学习场景 - 无人驾驶

无人驾驶：深度学习利用其深层的神经网络，通过一定的算法能训练出一个识别率非常高的分类器，从而能够使环境感知部分高精度的完成，为驾驶决策模块提供正确的环境信息，保证无人驾驶正常的完成。



2.2 深度学习场景 - 人脸识别

人脸识别：人脸信息的识别，是对人脸的信息加以提取然后进行识别的办法，一个最重要的目标就是分辨不同人的信息，辨别身份。面部识别的主要方式有：几何结构、子空间局部特征以及深度学习。



安防领域

开户审核

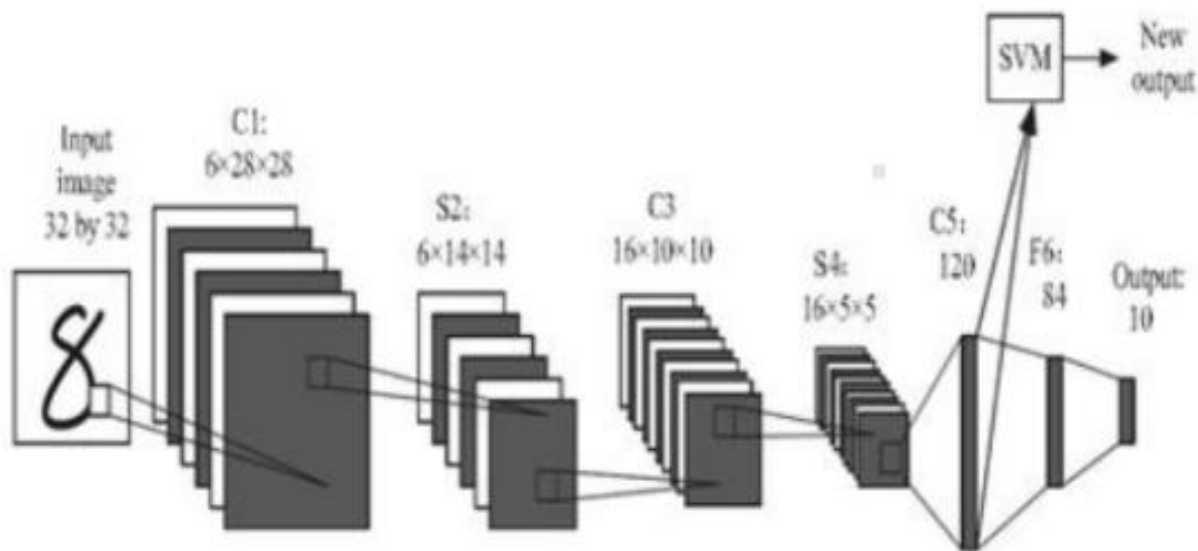
考勤门禁

.....

Figure 3: Pipeline of canonical view face selection (b) and face recovery (a).

2.3 深度学习场景 - 文字识别

文字识别：基于深度学习的文字识别系统的实现方法，属于图像处理技术领域，将包含文字的图片进行预处理、切分、识别，重组成一段文本，从而实现图片到文本的转换。涉及到图像预处理、图片切分、图片识别和文字重组。



审核

车牌识别

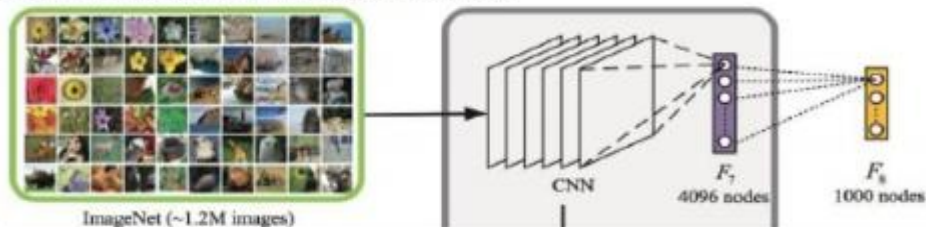
文档识别

.....

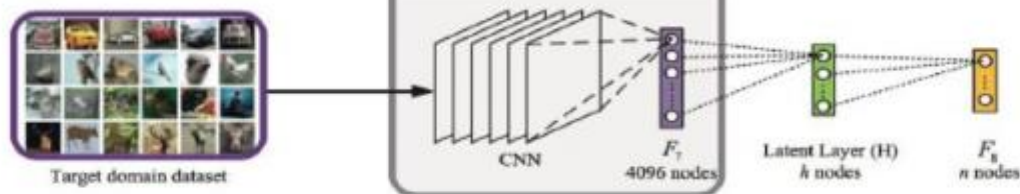
2.4 深度学习场景 - 图像识别

图片识别：图像识别与人脸及文字识别类似，主要流程包括图像预处理，图像分割，图像特征提取和图像分类。

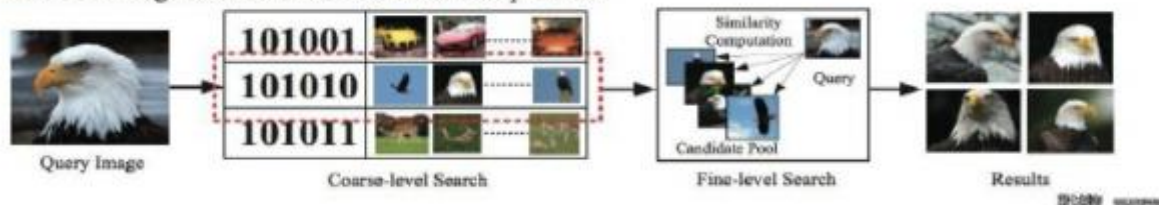
Module1: Supervised Pre-Training on ImageNet



Module2: Fine-tuning on Target Domain



Module3: Image Retrieval via Hierarchical Deep Search



图像审核

拍照购

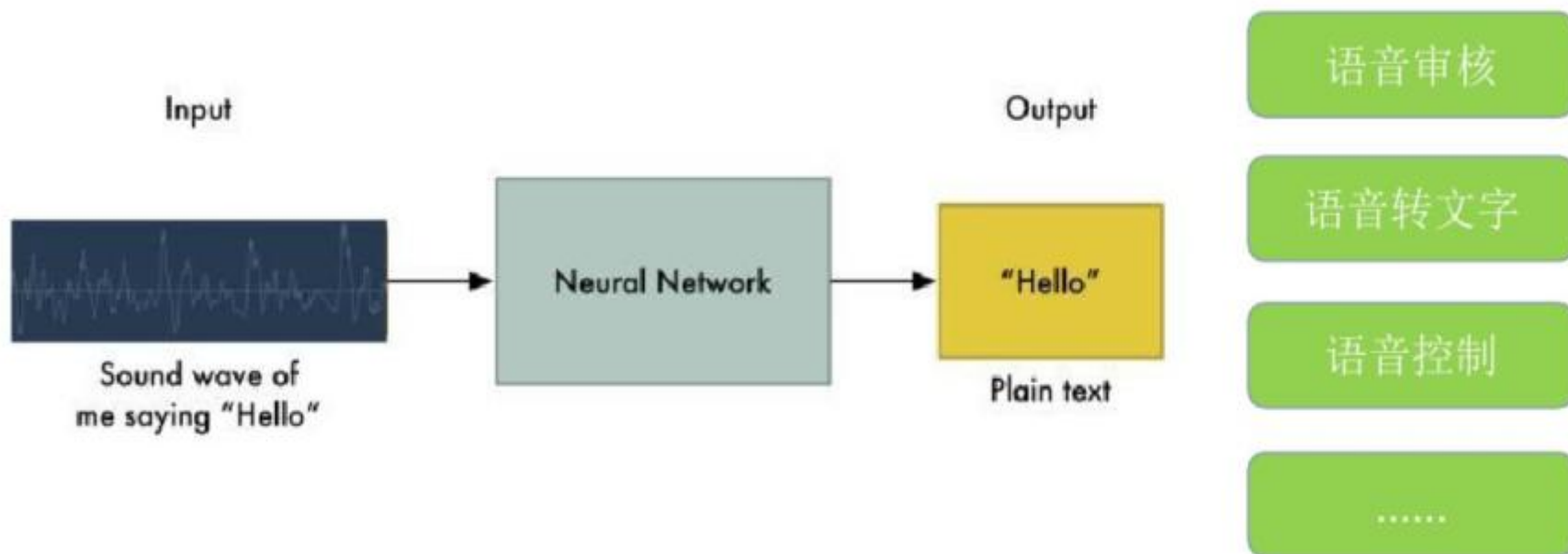
工业医疗检测

目标识别

.....

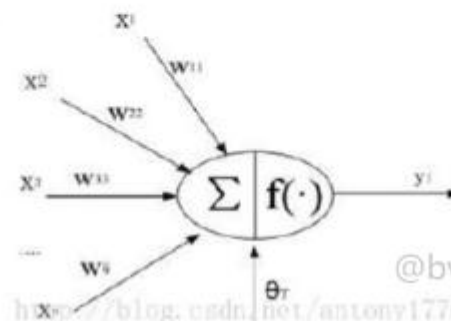
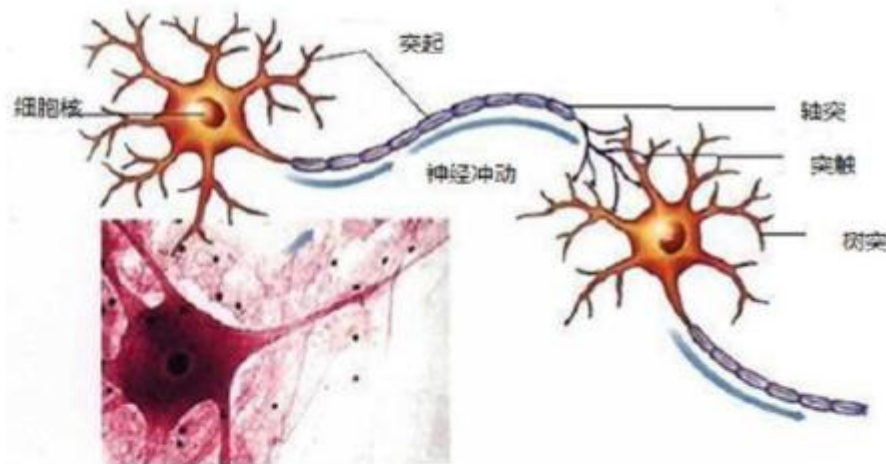
2.5 深度学习场景 - 语音识别

语音识别：一个完整的语音识别系统可大致分为3部分：语音特征提取、声学模型与模式匹配和语言模型与语言处理。其中声学模型是识别系统的底层模型，并且是语音识别系统中最关键的一部分。



3.1 人工神经网络 (ANN)

- 人工神经网络 (Artificial Neural Networks) 是一种模仿生物神经网络行为特征，进行分布式并行信息处理的算法数学模型。这种网络依靠系统的复杂程度，通过调整内部大量节点 (神经元) 之间相互连接的权重，从而达到处理信息的目的。



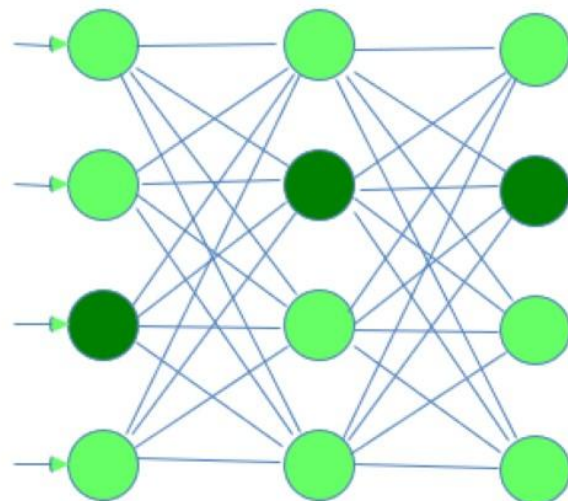
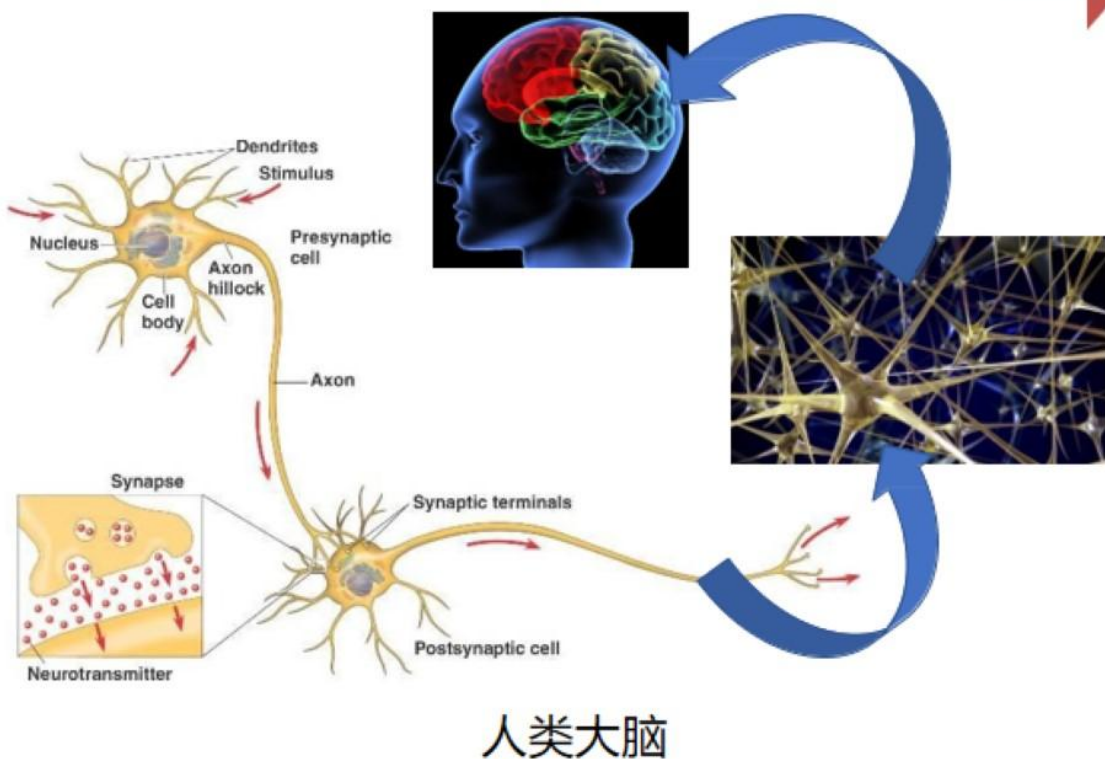
@bv...2@163.cc

<http://blog.csdn.net/antony1776>

□ 第三次人工智能热潮以深度学习为核心

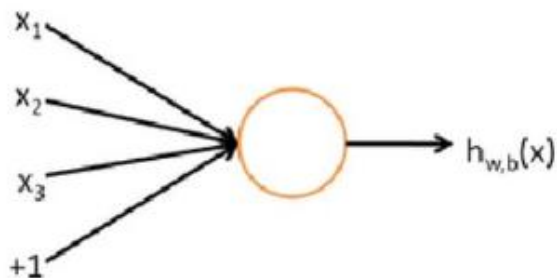
□ 深度学习属于连接主义学派，是机器学习非常重要的分支

□ 浅层神经网络对复杂函数的表示能力有限 → 更多层，深层

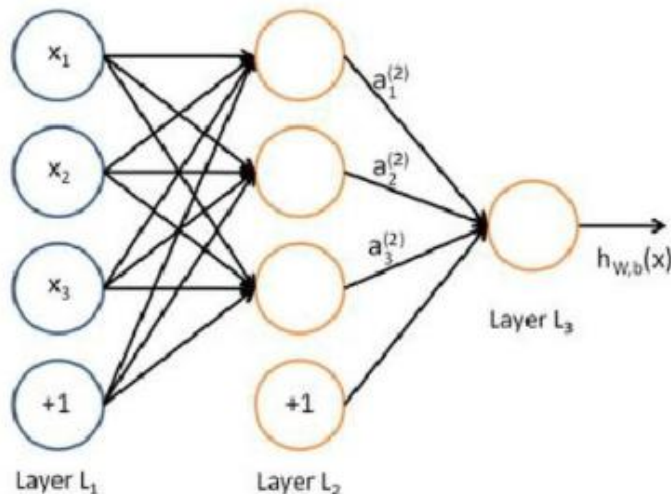


3.1 人工神经网络 (ANN)

● 神经网络



$$h_{W,b}(x) = f(W^T x) = f(\sum_{i=1}^3 W_i x_i + b)$$



$$a_1^{(2)} = f(W_{11}^{(1)} x_1 + W_{12}^{(1)} x_2 + W_{13}^{(1)} x_3 + b_1^{(1)})$$

$$a_2^{(2)} = f(W_{21}^{(1)} x_1 + W_{22}^{(1)} x_2 + W_{23}^{(1)} x_3 + b_2^{(1)})$$

$$a_3^{(2)} = f(W_{31}^{(1)} x_1 + W_{32}^{(1)} x_2 + W_{33}^{(1)} x_3 + b_3^{(1)})$$

$$h_{W,b}(x) = a_1^{(3)} = f(W_{11}^{(2)} a_1^{(2)} + W_{12}^{(2)} a_2^{(2)} + W_{13}^{(2)} a_3^{(2)} + b_1^{(2)})$$

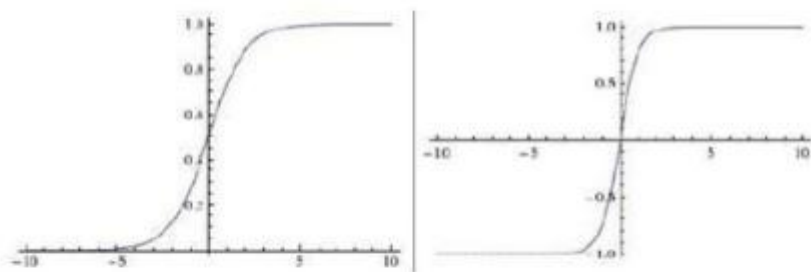
上述为一个简单地神经元结构， x_1, x_2, x_3 为输入， w_1, w_2, w_3 为权重， b 为偏置项，可以理解为使得我们的输出更加的偏向于真实值的调整项， z 为线性组合之后的输出，其中 $z = w_1 x_1 + w_2 x_2 + w_3 x_3 + b$ ， $g(z)$ 为激活函数将上面的线性变换通过激活函数进行非线性变换， a 是非线性变换之后的输出，则 $a = g(z) = g(w_1 x_1 + w_2 x_2 + w_3 x_3 + b)$

3.1 人工神经网络 (ANN)

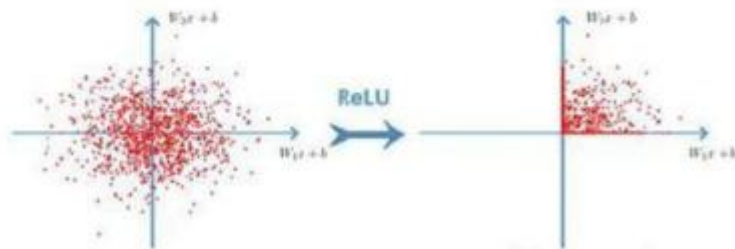
● 人工神经网络的重要概念：

1 权值矩阵：相当于神经网络的记忆！在训练的过程中，动态调整和适应。

2 激励函数：

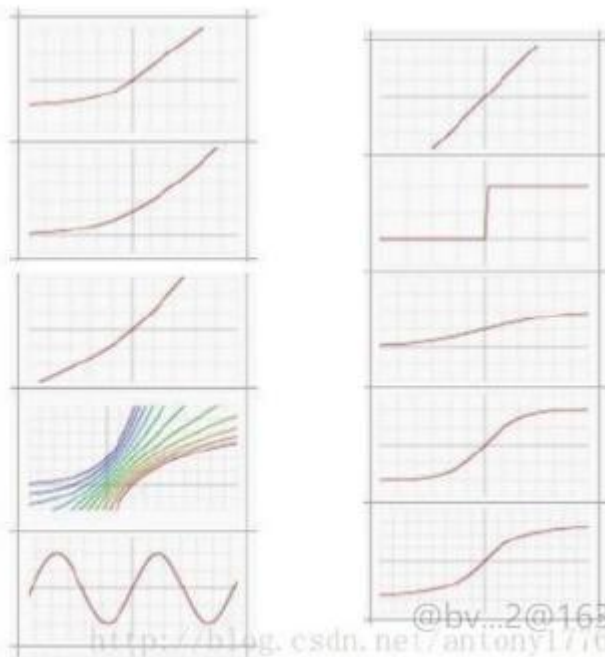


Sigmoid



Relu

Others



3.1 人工神经网络 (ANN)

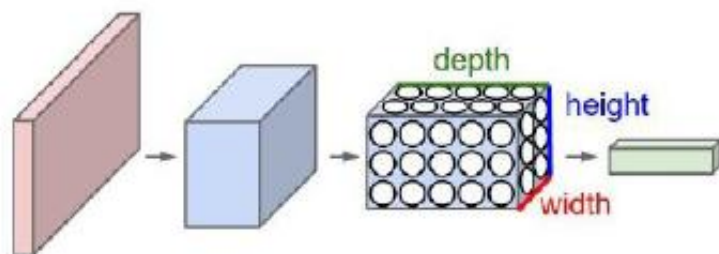
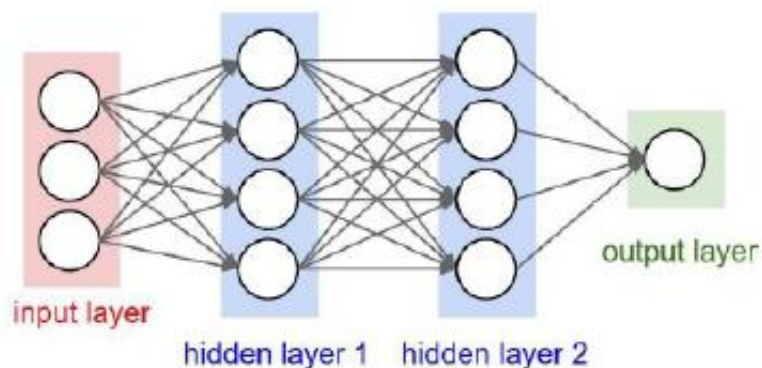
- 人工神经网络的重要概念：

激励函数很重要，无论是对建立神经网络的模型，还是理解神经网络。首先要了解，它有以下几个影响：

- 1 如何能更好的求解目标函数的极值！——高等数学中求解函数极值的知识！
可微，单调！
- 2 如何提升训练效率，让梯度的优化方法更稳定；
- 3 权值的初始值，不影响训练结果！

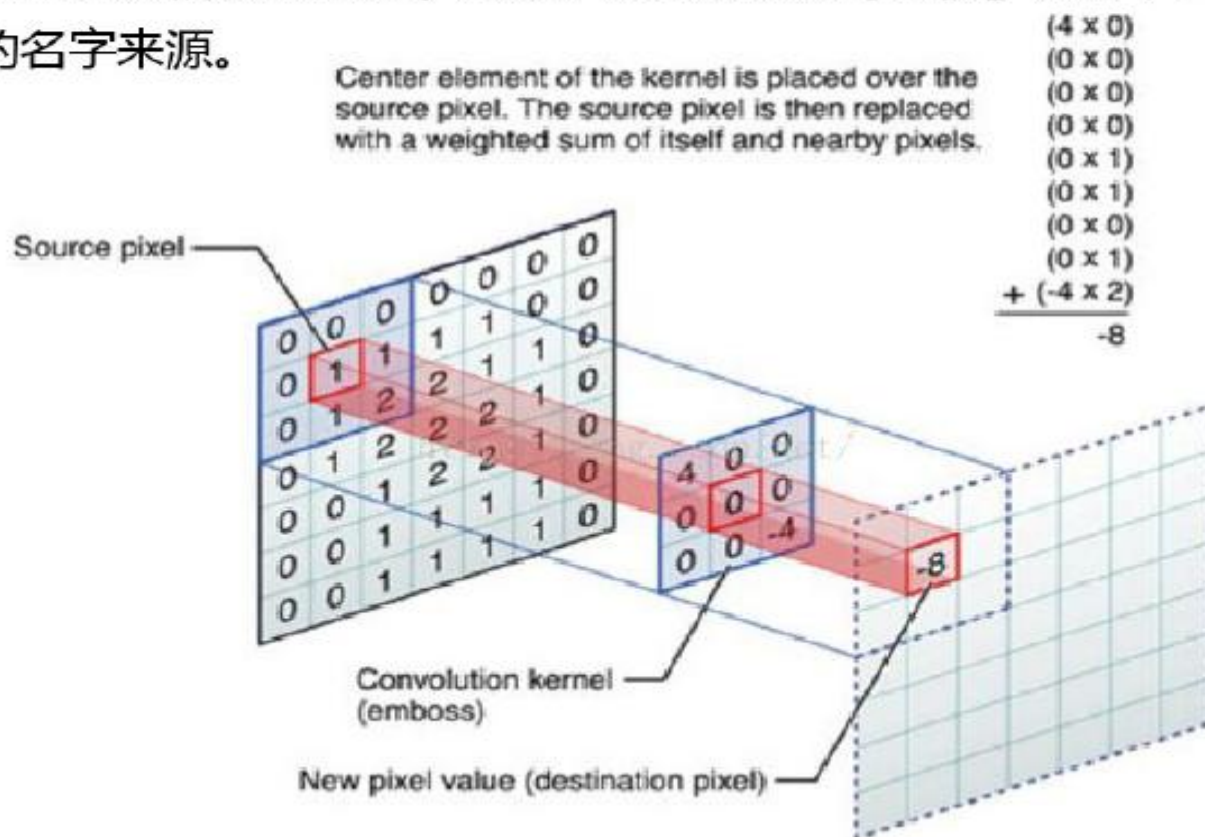
3.1 卷积神经网络 (CNN)

- 卷积神经网络 (Convolutional Neural Networks / CNNs / ConvNets) 与普通神经网络非常相似，它们都由具有可学习的权重和偏置常量(biases)的神经元组成。每个神经元都接收一些输入，并做一些点积计算，输出是每个分类的分数，普通神经网络里的一些计算技巧到这里依旧适用。
- 与普通神经网络不同之处：卷积神经网络默认输入是图像，可以让我们把特定的性质编码入网络结构，使是我们的前馈函数更加有效率，并减少了大量参数。



3.1 卷积神经网络 (CNN)

- 卷积操作：对图像（不同的数据窗口数据）和滤波矩阵（一组固定的权重：因为每个神经元的权重固定，所以又可以看做一个恒定的滤波器filter）做内积（逐个元素相乘再求和）的操作就是所谓的『卷积』操作，也是卷积神经网络的名字来源。



3.1 卷积神经网络 (CNN)



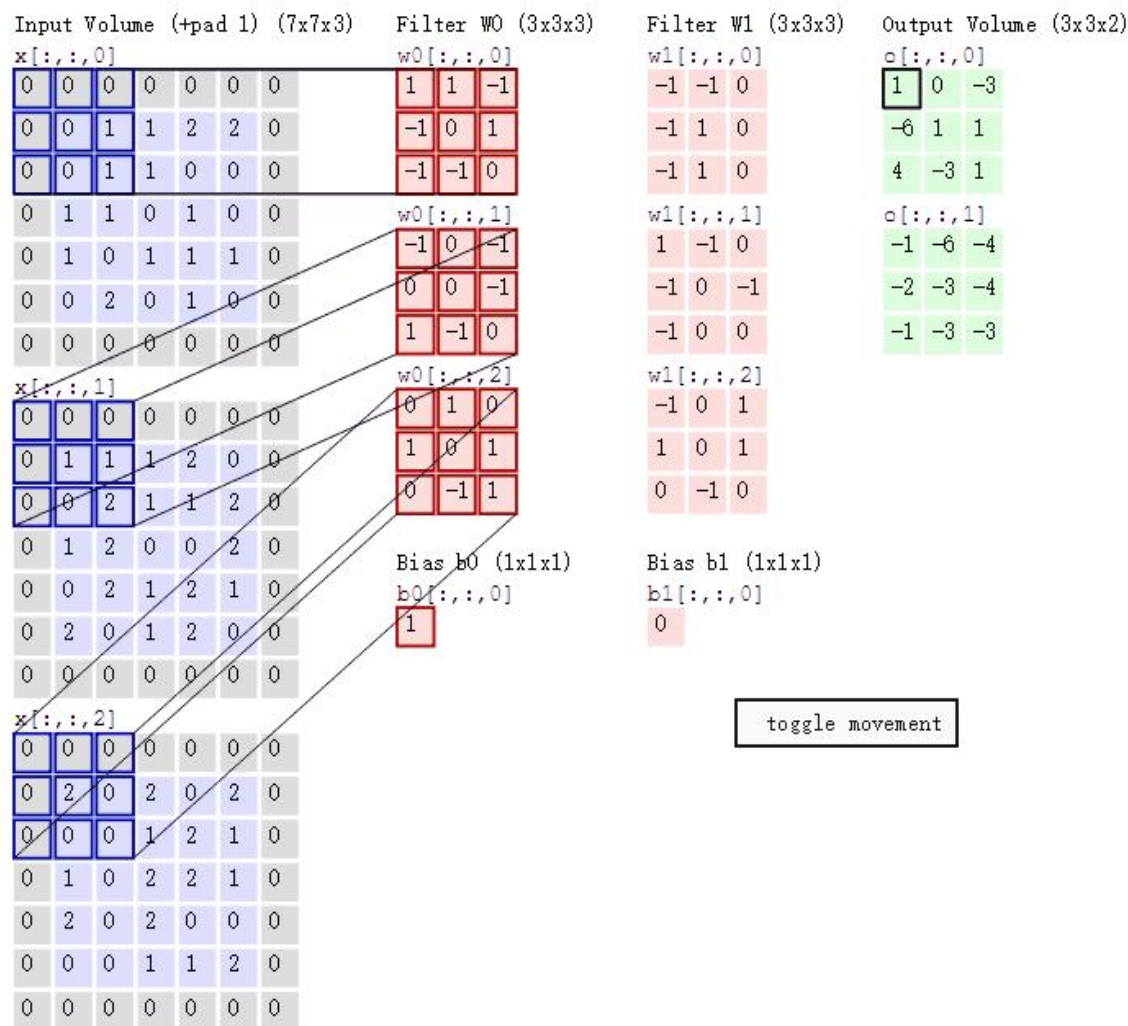
$$\begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 8 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} =$$



$$\begin{array}{|c|c|c|} \hline -1 & -1 & 0 \\ \hline -1 & 0 & 1 \\ \hline 0 & 1 & 1 \\ \hline \end{array} =$$



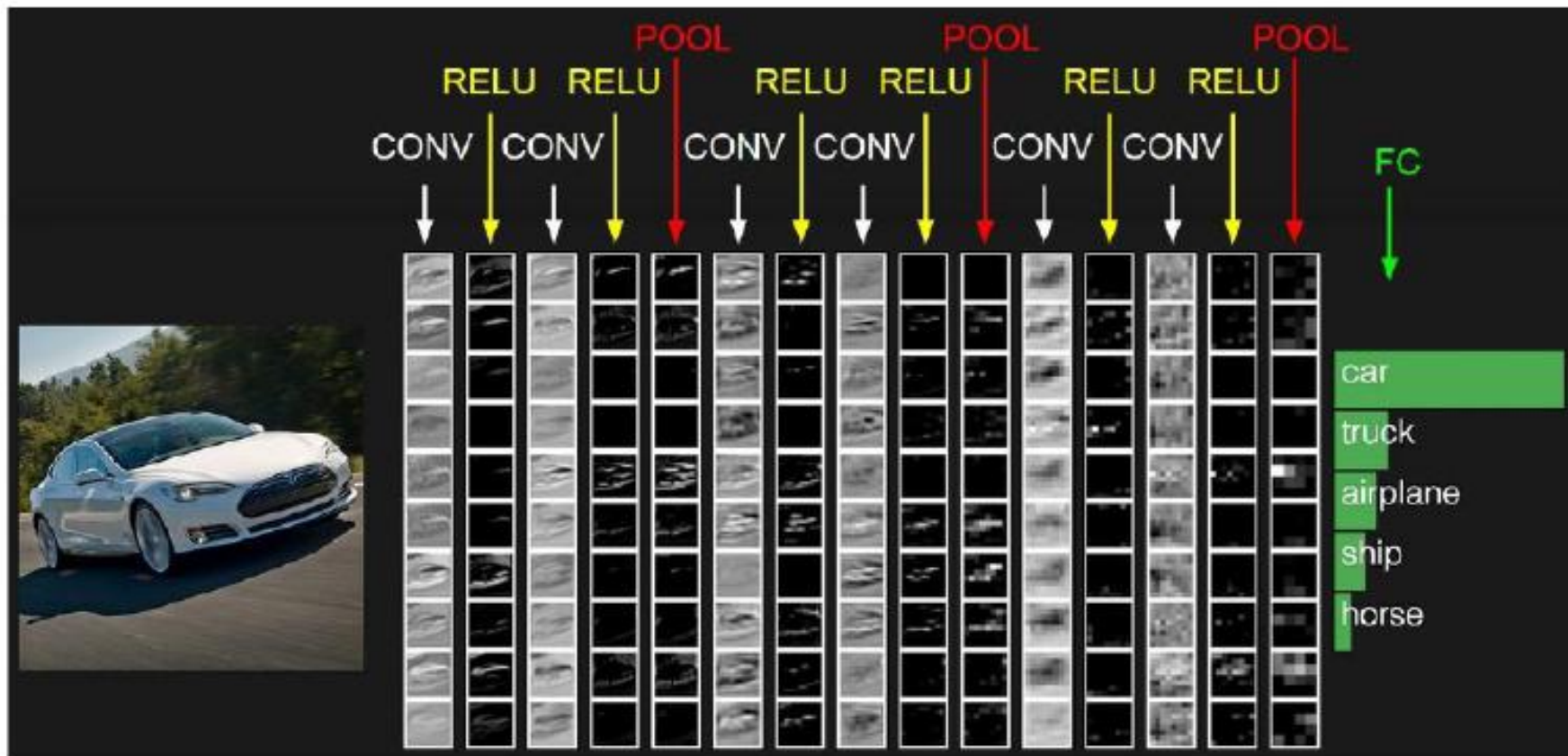
3.1卷积神经网络(CNN)



3.1 卷积神经网络 (CNN)

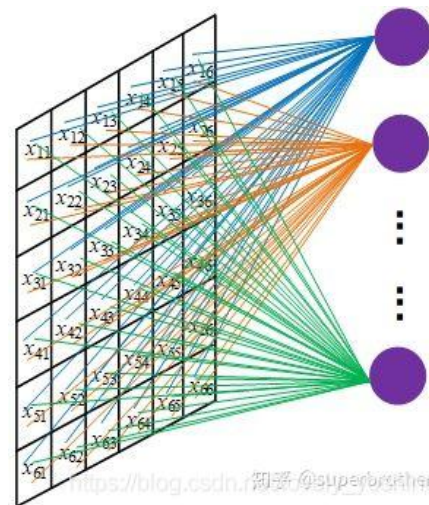
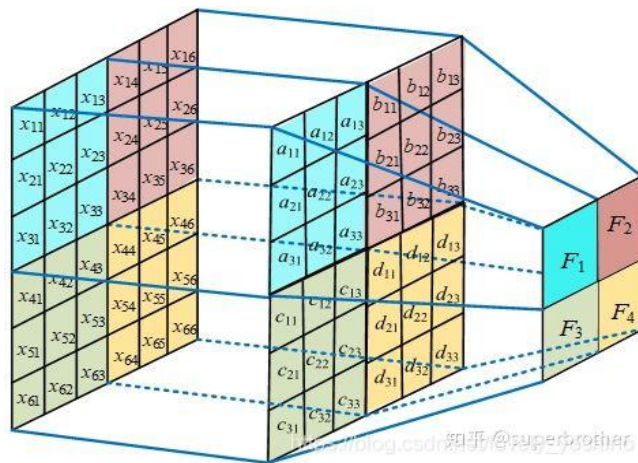
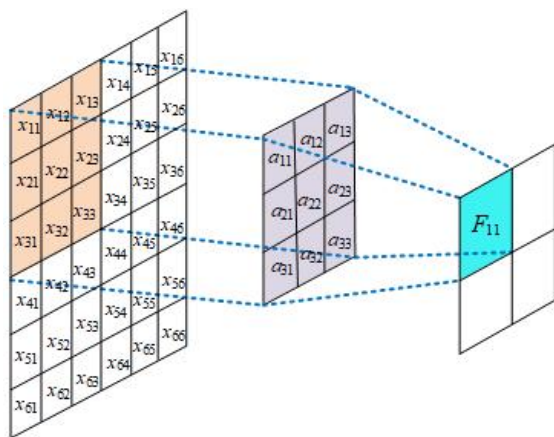
- 卷积层 (Convolutional layer) , 卷积神经网络中每层卷积层由若干卷积单元组成, 每个卷积单元的参数都是通过反向传播算法优化得到的。卷积运算的目的是提取输入的不同特征, 第一层卷积层可能只能提取一些低级的特征如边缘、线条和角等层级, 更多层的网络能从低级特征中迭代提取更复杂的特征。
- 线性整流层 (Rectified Linear Units layer, ReLU layer) , 这一层神经的激励函数 (Activation function) 使用线性整流 (Rectified Linear Units, ReLU) $f(x)=\max(0,x)$ 。
- 池化层 (Pooling layer) , 通常在卷积层之后会得到维度很大的特征, 将特征切成几个区域, 取其最大值或平均值, 得到新的、维度较小的特征。
- 全连接层 (Fully-Connected layer) , 把所有局部特征结合变成全局特征, 用来计算最后每一类的得分。

3.1 卷积神经网络 (CNN)



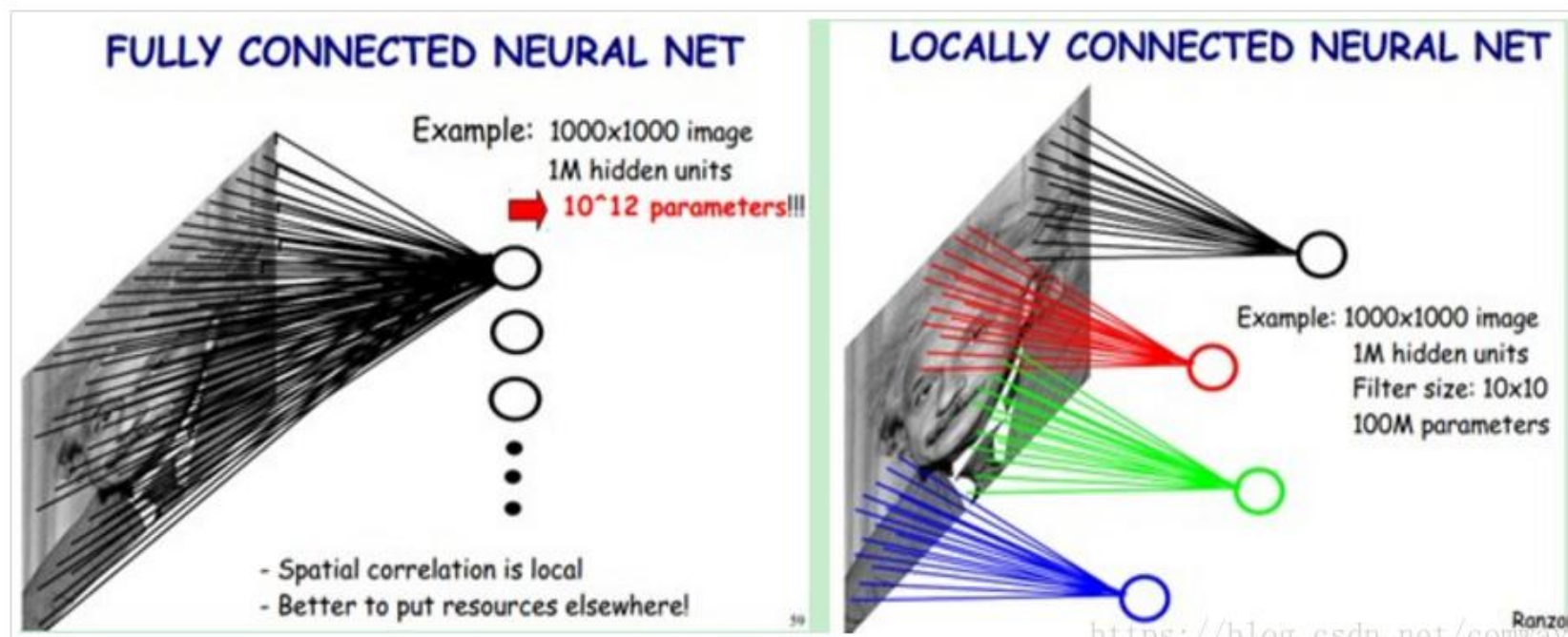
3.1 卷积神经网络(CNN)

●在图像处理中，往往把图像表示为像素的向量，比如一个1000x1000的图像，可以表示为一个1000000的向量。在上节中提到的神经网络中，如果隐含层数目与输入层一样，即也是1000000时，那么输入层到隐含层的参数数据为1000000x1000000= 10^{12} ，这样就太多了，基本没法训练。所以图像处理要想练成神经网络大法，必先减少参数加快速度。



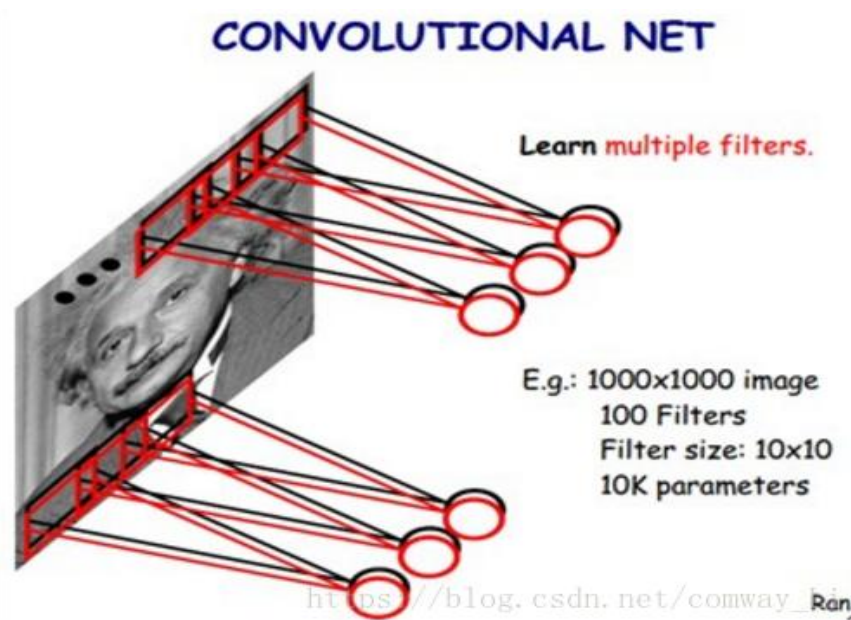
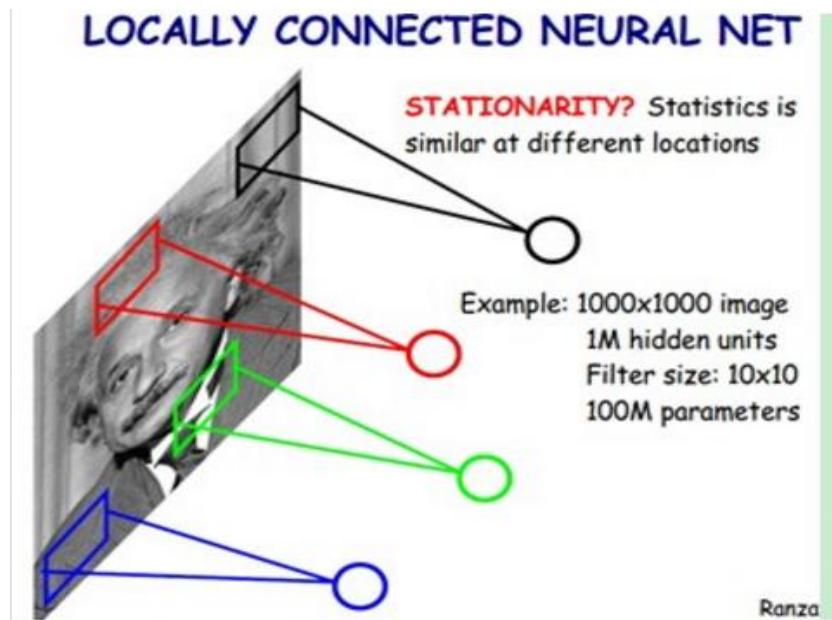
3.1 卷积神经网络 (CNN)

- 卷积神经网络有两种神器可以降低参数数目，第一种神器叫做局部感知。
- 在下方右图中，假如每个神经元只和 10×10 个像素值相连，那么权值数据为 1000000×100 个参数，减少为原来的万分之一。而那 10×10 个像素值对应的 10×10 个参数，其实就相当于卷积操作。



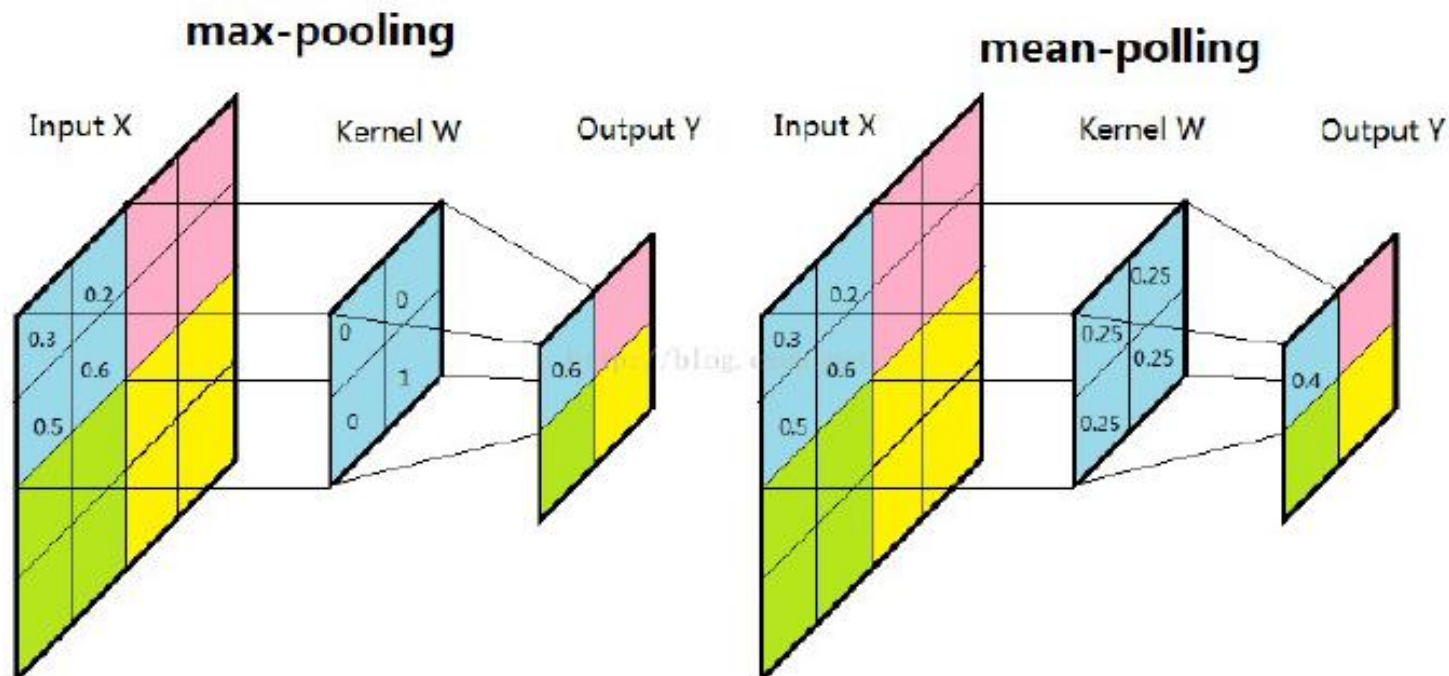
3.1卷积神经网络(CNN)

原理：所谓权值共享就是说给定一张输入图片，用一个卷积核来卷积这张图，卷积核里的值叫做权重，这张图的每个位置是被同一个卷积核扫的，即卷积的时候所用的权重是一样的。其实权值共享这个词说全了就是整张图片在使用同一个卷积核内的参数，比如一个 $3*3*1$ 的卷积核，这个卷积核内9个的参数被整张图共享，而不会因为图像内位置的不同而改变卷积核内的权系数。说的再直白一些，就是用一个卷积核不改变其内权系数的情况下卷积处理整张图片（当然CNN中每一层不会只有一个卷积核的）。



3.1 卷积神经网络 (CNN)

- 池化，也称作下采样，可以实现降维。常用有最大值池化和均值池化。



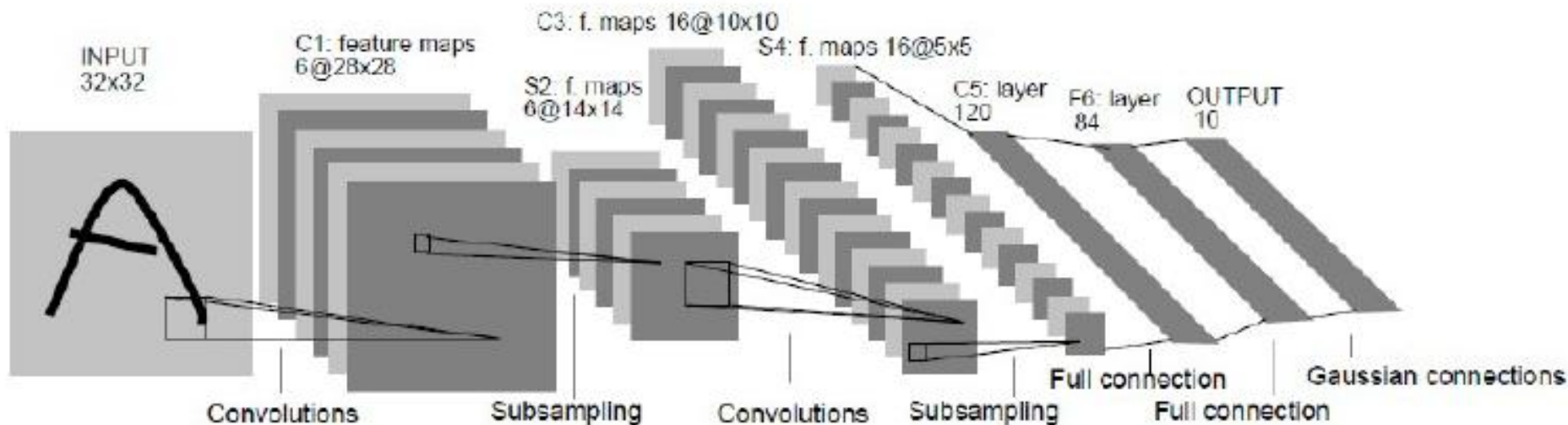


3.1 卷积神经网络 (CNN)

- 全连接层：连接所有的特征，将输出值送给分类器（如softmax分类器），最终得出识别结果。

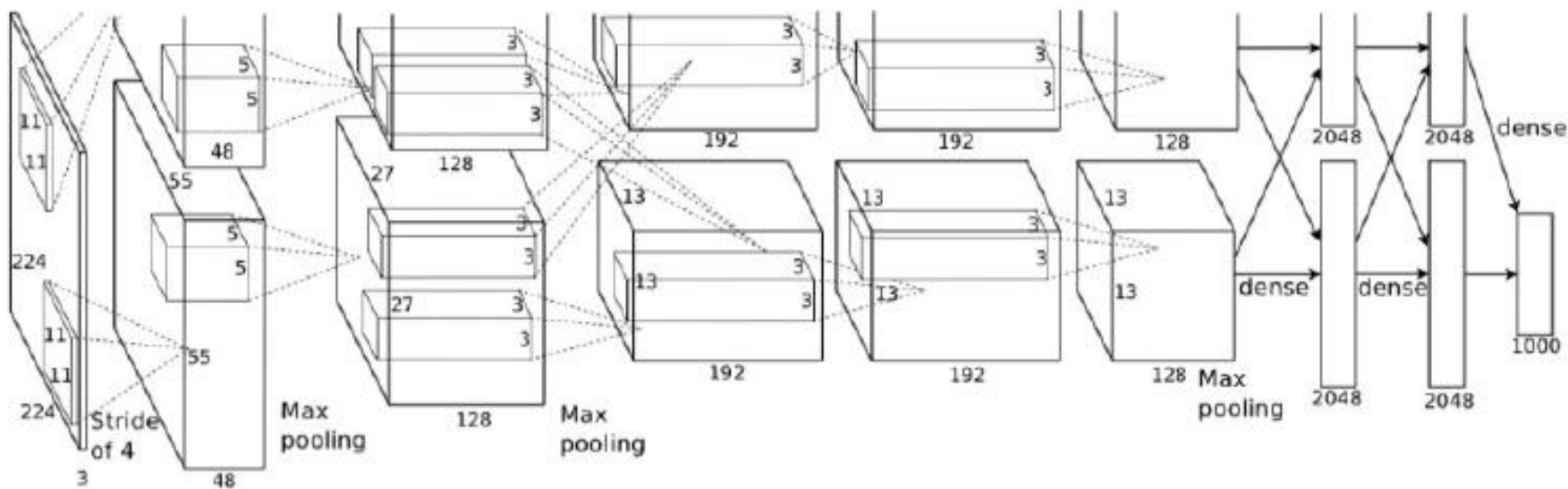
3.2 常见网络模型

- LeNet



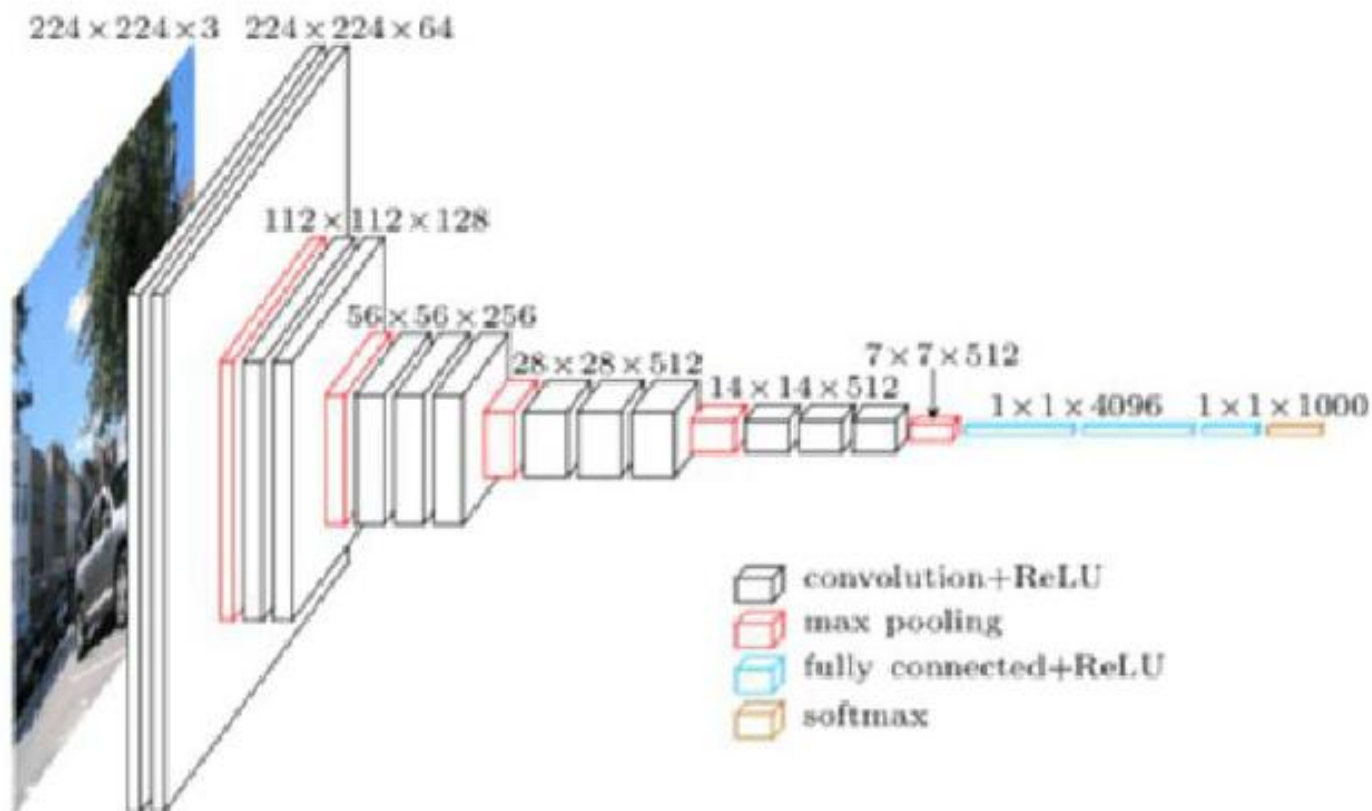
3.2 常见网络模型

- AlexNet



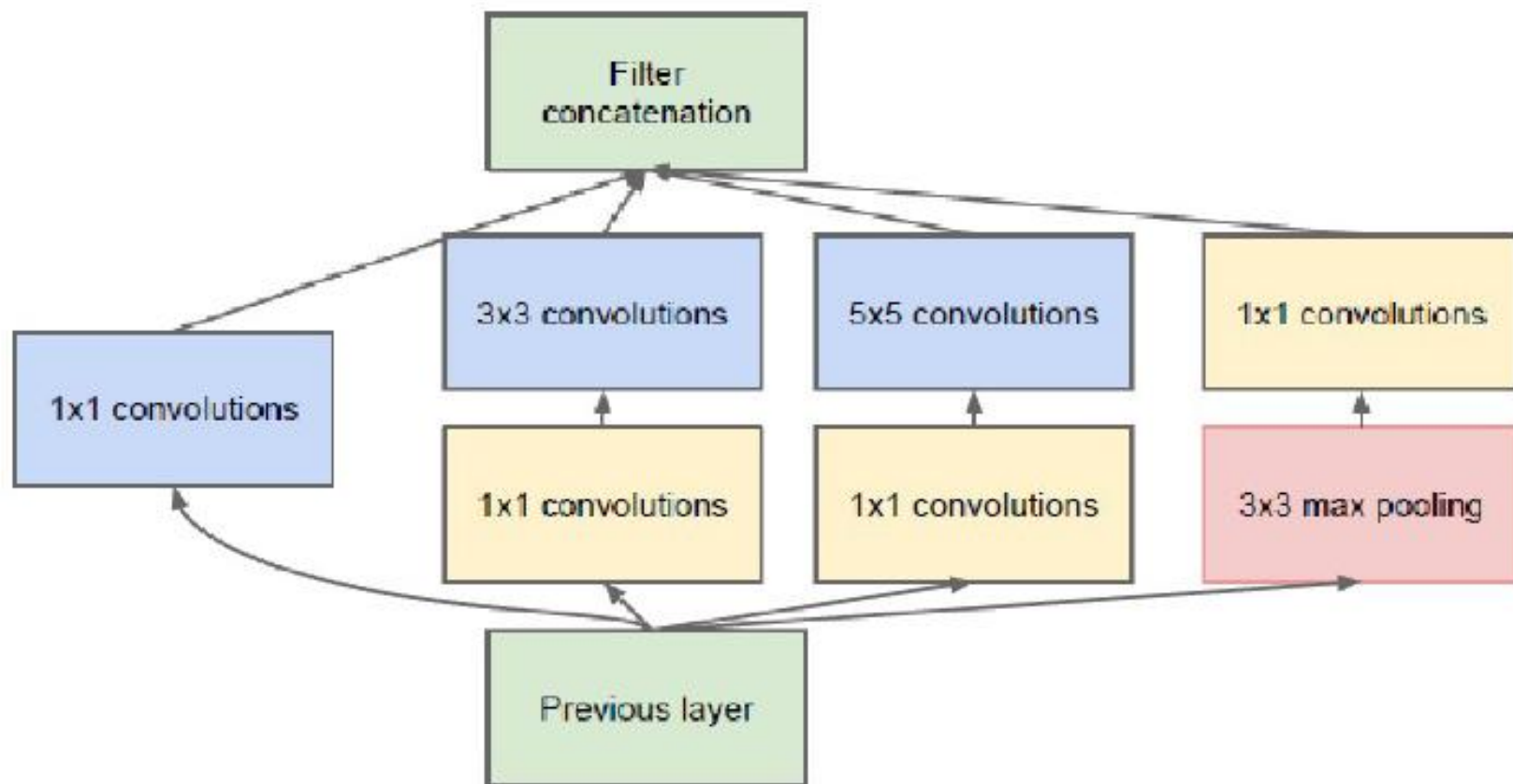
3.2 常见网络模型

● VGG16



3.2 常见网络模型

- GoogleNet (InceptionV4)





CNN模型比较.doc

3.2 常见网络模型

● 比较

模型名	AlexNet	VGG	GoogLeNet	ResNet
初入江湖	2012	2014	2014	2015
层数	8	19	22	152
Top-5错误	16.4%	7.3%	6.7%	3.57%
Data Augmentation	+	+	+	+
Inception(NIN)	-	-	+	-
卷积层数	5	16	21	151
卷积核大小	11,5,3	3	7,1,3,5	7,1,3,5
全连接层数	3	3	1	1
全连接层大小	4096,4096,1000	4096,4096,1000	1000	1000
Dropout	+	+	+	+
Local Response Normalization	+	-	+	-
Batch Normalization	-	-	-	+



3.3 其他深度学习算法

- 自动编码器 (AutoEncoder)
- 稀疏编码 (Sparse Coding)
- 限制玻尔兹曼机 (RBM)

4.1 开源框架概述

- 深度学习研究的热潮持续高涨，各种开源深度学习框架也层出不穷，其中包括TensorFlow、Caffe、Keras、CNTK、Torch7、MXNet、Leaf、Theano、DeepLearning4J、Lasagne、Neon等等。下图是各个开源框架在GitHub上的数据统计（2017年初）。

框 架	机 构	支持语言	Stars	Forks	Contributors
TensorFlow	Google	Python/C++/Go/...	41628	19339	568
Caffe	BVLC	C++/Python	14956	9282	221
Keras	felixhallet	Python	10727	3575	322
CNTK	Microsoft	C++	9063	2144	100
MXNet	DMLC	Python/C++/R/...	7393	2745	241
Torch7	Facebook	Lua	6111	1784	113
Theano	U. Montreal	Python	5352	1868	271
Deeplearning4J	DeepLearning4J	Java/Scala	5053	1927	101
Leaf	AutumnAI	Rust	4562	216	14
Lasagne	Lasagne	Python	2749	761	55
Neon	NervanaSystems	Python	2633	573	52

4.1 开源框架概述

- Google、Microsoft、Facebook等巨头都参与了这场深度学习框架大战，此外，还有毕业于伯克利大学的贾扬清主导开发的Caffe，蒙特利尔大学Lisa Lab团队开发的Theano，以及其他个人或商业组织贡献的框架。下表是主流深度学习框架在各个维度的评分。

	模型设计	接 口	部 署	性 能	架构设计	总体评分
TensorFlow	80	80	90	90	100	88
Caffe	60	60	90	80	70	72

续表

	模型设计	接 口	部 署	性 能	架构设计	总体评分
CNTK	50	50	70	100	60	66
Theano	80	70	40	50	50	58
Torch	90	70	60	70	90	76
MXNet	70	100	80	80	90	84
DeepLearning4J	60	70	80	80	70	72

4.2 TensorFlow

- TensorFlow最初是由研究人员和Google Brain团队针对机器学习和深度神经网络进行研究所开发的，目前开源之后可以在几乎各种领域适用。
- TensorFlow灵活的架构可以部署在一个或多个CPU、GPU的台式以及服务器中，或者使用单一的API应用在移动设备中。



4.3 Caffe

- Caffe由加州大学伯克利的PHD贾扬清开发，全称Convolutional Architecture for Fast Feature Embedding，是一个清晰而高效的开源深度学习框架，目前由伯克利视觉学中心（ Berkeley Vision and Learning Center，BVLC ）进行维护。（ 贾扬清曾就职于MSRA、NEC、Google Brain，他也是TensorFlow的作者之一，目前任职于Facebook FAIR实验室。 ）
- Caffe2脸书 (Facebook) 出品，为生产环境设计，提供在各种平台（ 包括移动设备 ）的运行。



4.4 Torch

- Torch是一个有大量机器学习算法支持的科学计算框架，其诞生已经有十年之久，但是真正起势得益于Facebook开源了大量Torch的深度学习模块和扩展。Torch另外一个特殊之处是采用了编程语言Lua(该语言曾被用来开发视频游戏)。
- PyTorch是基于Torch的衍生，支持Python语言，实现了机器学习框架 Torch 在 Python 语言环境的执行。



4.5 Theano

- 2008年诞生于蒙特利尔理工学院,Theano派生出了大量深度学习Python软件包,最著名的包括Blocks和Keras。Theano的核心是一个数学表达式的编译器,它知道如何获取你的结构。并使之成为一个使用numpy、高效本地库的高效代码,如BLAS和本地代码(C++)在CPU或GPU上尽可能快地运行。它是为深度学习中处理大型神经网络算法所需的计算而专门设计的,是这类库的首创之一(发展始于2007年),被认为是深度学习研究和开发的行业标准。



theano



目录

- 深度学习概述
- 深度学习常见场景
- 深度学习常用算法介绍
- 深度学习常用框架介绍
- TensorFlow入门
- 深度学习未来和展望



5.1 引言

- TensorFlow的API主要分两个层次，核心层和基于核心层的高级API。核心层面向机器学习的研究人员，以及对模型控制精细的相关人员。高级API使用和学习相对容易，简化重复性任务，使不同的用户之间保持一致性。
- 高级API，如`tf.contrib.learn`可以帮助管理数据集data set，估计量estimators，训练training，推理inference
- 注意，一些高级API的方法名中包含contrib，意味着这些API依然处于开发过程中，这些方法在后续的TensorFlow版本中可能改变或者不再使用
- 本章从核心层开始，后边会提到如何使用`tf.contrib.learn`实现模型。了解核心层，在使用高级API的时候知道程序是如何工作的。

5.2 安装

- 推荐使用Anaconda环境安装，集成了Python，numpy等。
- Windows要求Python版本为3.5。
- 在线环境直接使用pip install tensorflow (tensorflow_gpu)
- 离线环境可下载whl包进行安装 (<https://pypi.python.org/pypi>)，然后同样执行pip install xx.whl进行安装，安装过程中若提示缺少依赖包可下载安装后再继续安装。
- 推荐Python开发工具 PyCharm



- 安装成功后验证安装是否成功

```
$python
```

```
>>import tensorflow as tf
```

```
>>tf.__version__ #查看TF版本
```

```
>>tf.__path__ #查看TF安装路径
```



6.1 未来与展望

- 无监督半监督学习所占比重会越来越大。
- 用于学习的硬件设备会越来越强大，效率越来越高。
- 有向移动端转移的趋势。
- 可视化开发，开发框架会越来越普遍，入门更加简单。
- 小数据样本的训练所占比重会增大。