教育部-华为智能基座课程

《人工智能基础与实践》

第5章：卷积神经网络I

授课教师：丛润民

山东大学

控制科学与工程学院

章节目录 CONTENTS

前言

在用全连接前馈网络来处理图像时， 存在以下两个问题：

- ✓ 参数太多：如果输入图像大小为100×100×3，在全连接前馈网络中，第一个隐藏层的每个神经元到输入层都有30,000个互相独立的连接，每个连接都对应一个权重参数。随着隐藏层神经元数量的增多，参数的规模也会急剧增加。这会导致整个神经网络的训练效率非常低，也很容易出现过拟合。
- ✓ 局部不变性特征：自然图像中的物体都具有局部不变性特征，比如尺度缩放、平移、旋转等操作不影响其语义信息。而全连接前馈网络很难提取这些局部不变性特征，一般需要进行数据增强来提高性能。

前言

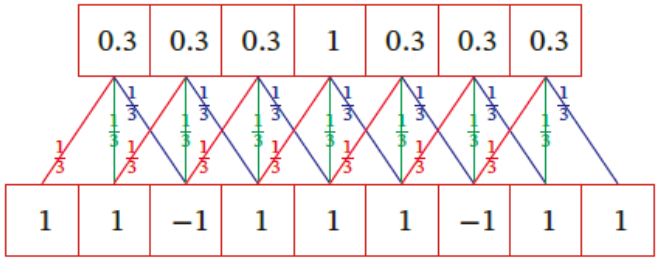
感受野（Receptive Field）机制主要是指听觉、视觉等神经系统中一些神经元的特性，即神经元只接受其所支配的刺激区域内的信号。

卷积神经网络（Convolutional Neural Network，CNN）是受生物学上感受野机制的启发而提出的。卷积神经网络一般是由卷积层、池化层和全连接层交叉堆叠而成的前馈神经网络。全连接层一般在卷积网络的最顶层。卷积神经网络有三个结构上的特性：局部连接、权重共享以及池化。这些特性使得卷积神经网络具有一定程度上的平移、缩放和旋转不变性。和前馈神经网络相比，卷积神经网络的参数少。

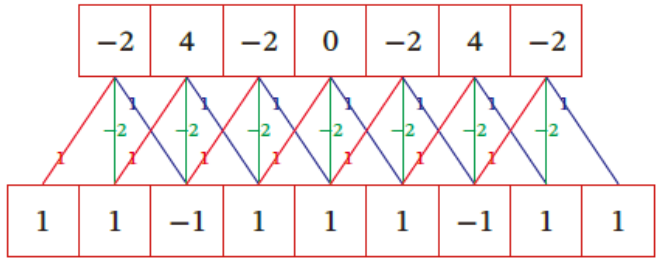
章节目录 CONTENTS

卷积运算 卷积的定义

卷积（Convolution）是分析数学中一种重要的运算。在信号处理或图像处理中，经常使用一维或二维（离散）卷积。



(a) 滤波器 [1/3, 1/3, 1/3]



(b) 滤波器 [1, −2, 1]

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -1 & 0 & -3 & 0 & 1 \\ 2 & 1 & 1 & -1 & 0 \\ 0 & -1 & 1 & 2 & 1 \\ 1 & 2 & 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 0 & -2 & -1 \\ 2 & 2 & 4 \\ -1 & 0 & 0 \end{bmatrix}$$

卷积运算 一维卷积

- 一维卷积经常用在信号处理中，用于计算信号的延迟累积。
- 假设一个信号发生器每个时刻t产生一个信号x_t，其信息的衰减率为w_k，即在$k-1$个时间步长后，信息为原来的w_k倍，假设$w_1 = 1, w_2 = 1/2, w_3 = 1/4$，时刻$t$收到的信号$y_t$为当前时刻产生的信息和以前时刻延迟信息的叠加。

$$
\begin{aligned}
y_t &= 1 \times x_t + 1/2 \times x_{t-1} + 1/4 \times x_{t-2} \\
&= w_1 \times x_t + w_2 \times x_{t-1} + w_3 \times x_{t-2} \\
&= \sum_{k=1}^{3} w_k x_{t-k+1}.
\end{aligned}
$$

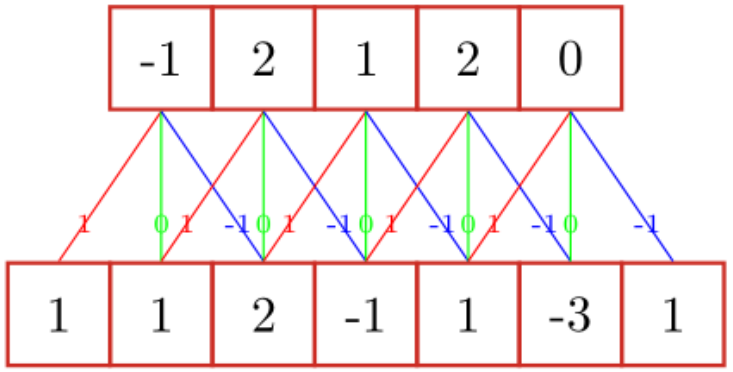
滤波器（filter）或卷积核（convolution kernel）

卷积运算 一维卷积

- 给定一个输入信号序列x和滤波器w，卷积的输出为：

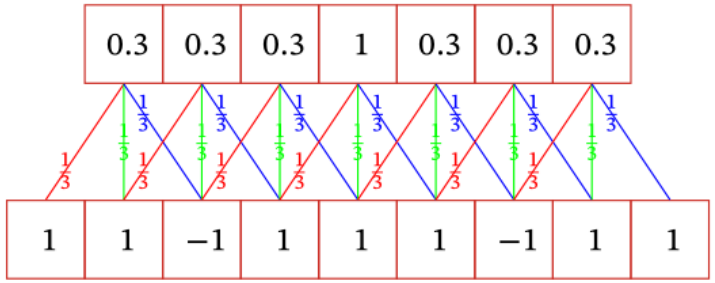
$$y_t = \sum_{k=1}^{F} w_k x_{t-k+1} \qquad \boldsymbol{y} = \boldsymbol{w} * \boldsymbol{x}$$

Filter: [-1,0,1]



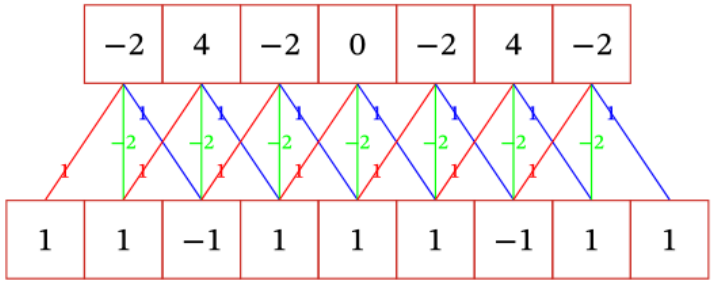
卷积运算 一维卷积

- **不同的滤波器**来提取信号序列中的**不同特征**



(a) 滤波器 [1/3, 1/3, 1/3]

低频信息



(b) 滤波器 [1, −2, 1]

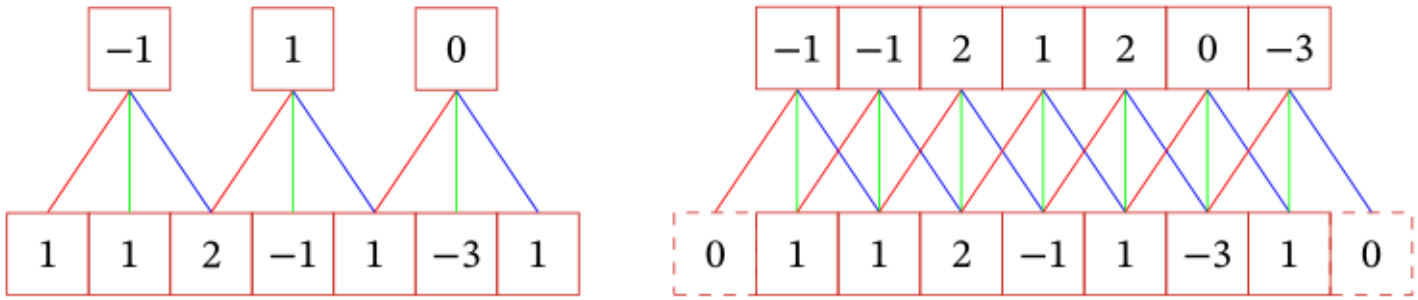
高频信息

$$y''(u) = y(u+1) + y(u-1) - 2y(u)$$

二阶微分

卷积运算 一维卷积

- 引入滤波器的滑动步长S 和 零填充P
  - **步长**（Stride）是指卷积核在滑动时的时间间隔。
  - **零填充**（Zero Padding）是在输入向量两端进行补零。



(a) 步长 $S = 2$　　(b) 零填充 $P = 1$

卷积运算 一维卷积

- 卷积的结果按输出长度不同可以分为三类：
  - **窄卷积**：步长 $T = 1$，两端不补零 $P = 0$，卷积后输出长度为 $M - F + 1$
  - **宽卷积**：步长 $T = 1$，两端补零 $P = F - 1$，卷积后输出长度 $M + F - 1$
  - **等宽卷积**：步长 $T = 1$，两端补零 $P = (F - 1)/2$，卷积后输出长度 M

- F 滤波器大小
- 在早期的文献中，卷积一般默认为窄卷积。
- 而目前的文献中，卷积一般默认为等宽卷积。

卷积运算 二维卷积

卷积也经常用在图像处理中。因为图像为一个二维结构，所以需要将一维卷积进行扩展，进行二维卷积。

给定一个图像$\boldsymbol{X} \in \boldsymbol{R}^{\boldsymbol{M}\times\boldsymbol{N}}$和一个滤波器$\boldsymbol{W} \in \boldsymbol{R}^{\boldsymbol{U}\times\boldsymbol{V}}$，一般$U << M, V << N$，其卷积为：

$$y_{ij} = \sum_{u=1}^{U}\sum_{v=1}^{V} w_{uv}x_{i-u+1,j-v+1}.$$

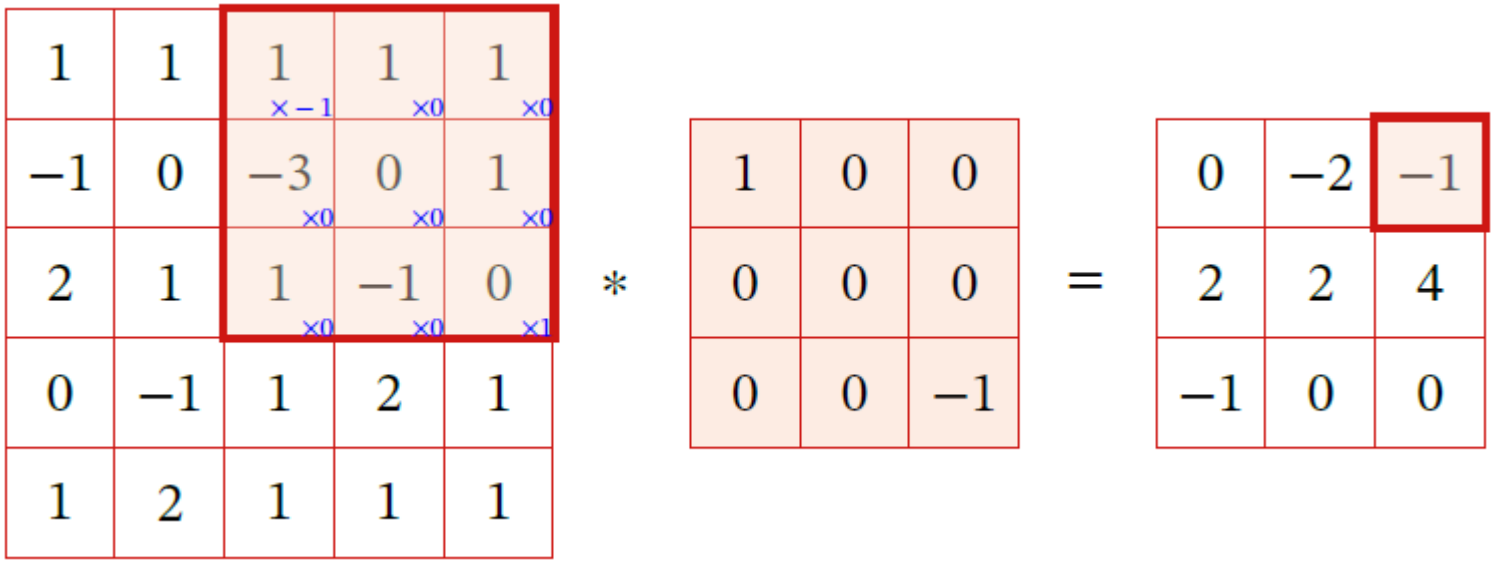
为了简单起见，这里假设卷积的输出 y_{ij} 的下标(i, j) 从(U, V) 开始。

卷积运算 二维卷积

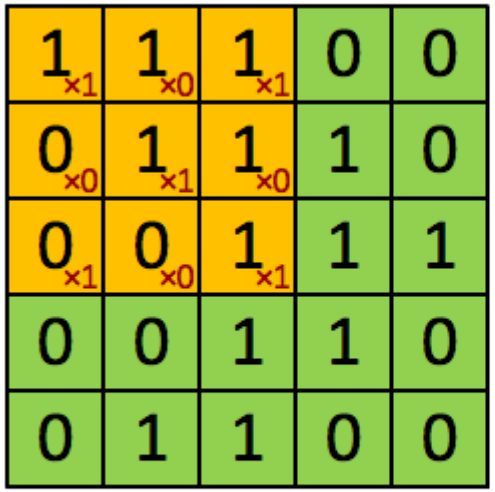
给定一个输入信息 $\boldsymbol{X}$ 和滤波器 $\boldsymbol{W}$ 的二维卷积定义为：

$$\boldsymbol{Y} = \boldsymbol{W} * \boldsymbol{X}$$

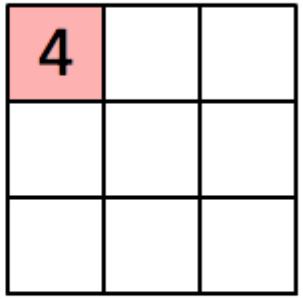
其中$*$ 表示二维卷积运算。



卷积运算 二维卷积



Image



Convolved Feature

INPUT IMAGE

18	54	51	239	244
55	121	75	78	95
35	24	204	113	109
3	154	104	235	25
15	253	225	159	78

WEIGHT

1	0	1
0	1	0
1	0	1

429

0	0	0	0	0	0	0	0
0	18	54	51	239	244	188	0
0	55	121	75	78	95	88	0
0	35	24	204	113	109	221	0
0	3	154	104	235	25	130	0
0	15	253	225	159	78	233	0
0	68	85	180	214	245	0	0
0	0	0	0	0	0	0	0

WEIGHT

1	0	1
0	1	0
1	0	1

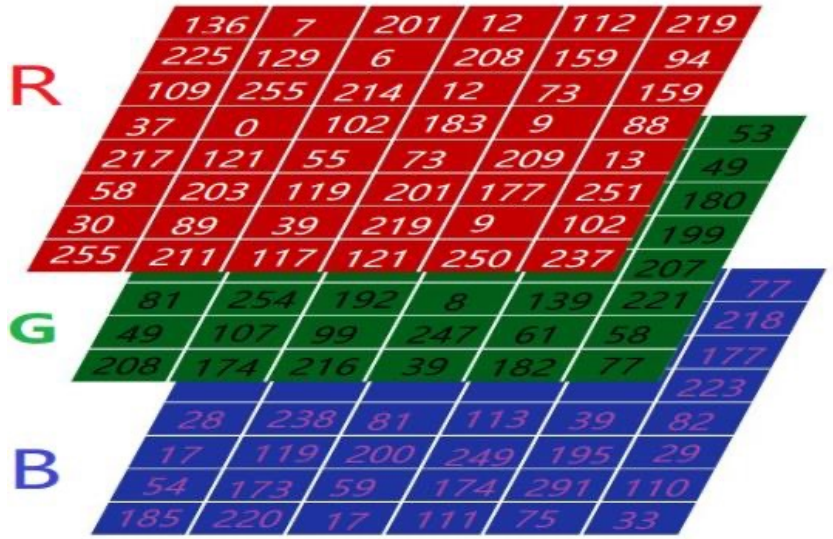
139

卷积运算 二维卷积

- 在图像处理中，卷积经常作为特征提取的有效方法。
- 在图像处理中常用的均值滤波（Mean Filter）就是一种二维卷积，将当前位置的像素值设为滤波器窗口中所有像素的平均值，即$w_{uv} = \frac{1}{UV}$。

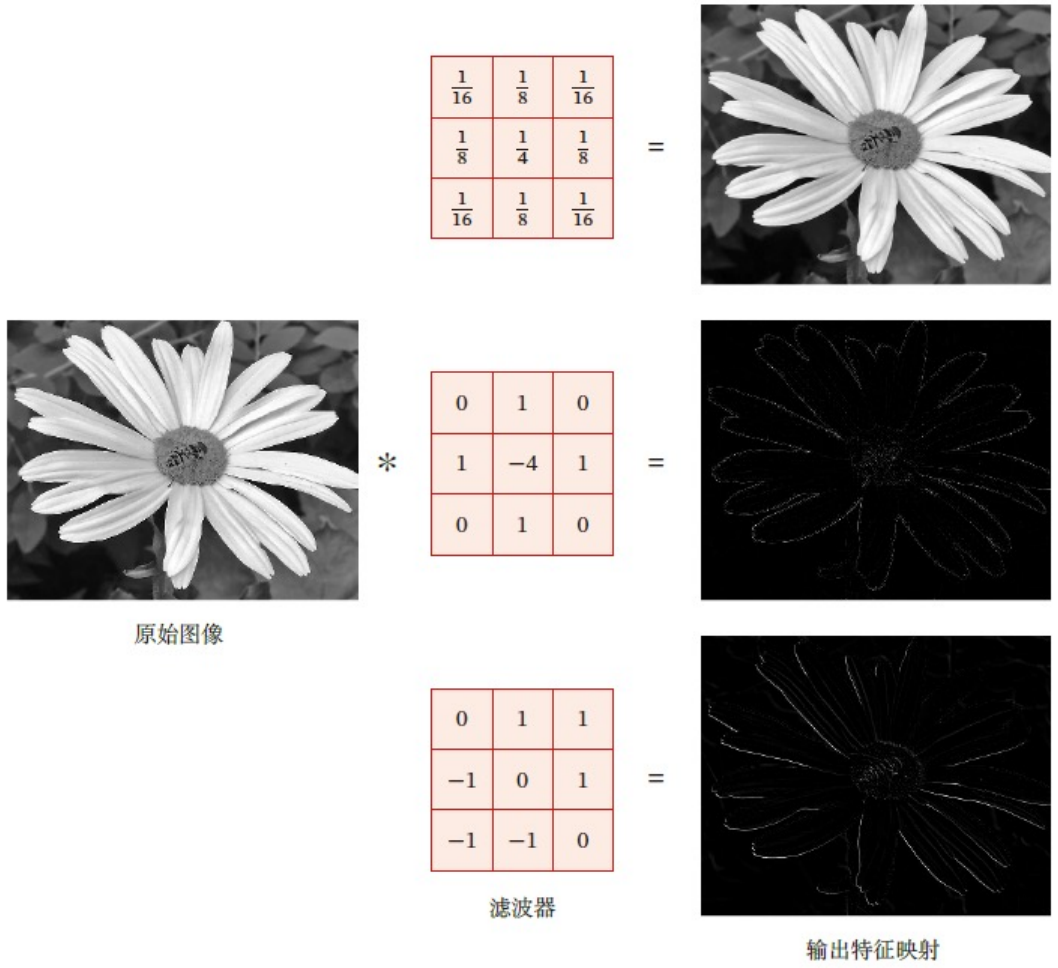
What We See

What Computers See



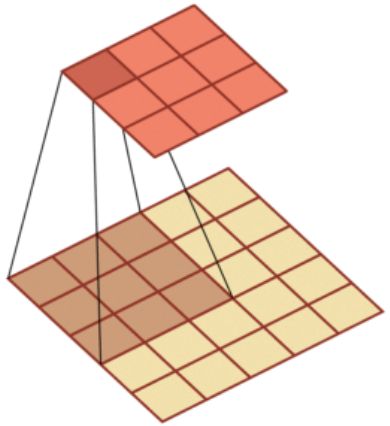
卷积运算 二维卷积

- 一幅图像在经过卷积操作后得到结果称为**特征映射**（Feature Map）。
- 右图给出在图像处理中几种常用的滤波器，以及其对应的特征映射。图中最上面的滤波器是常用的高斯滤波器，可以用来对图像进行平滑去噪；中间和最下面的滤波器可以用来提取边缘特征。

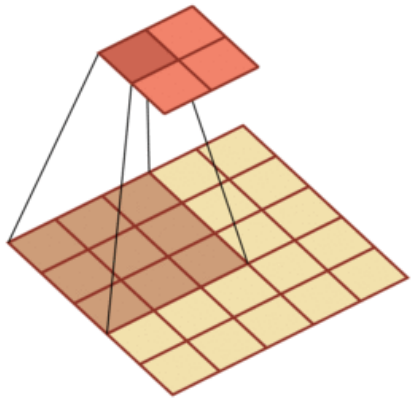


Input

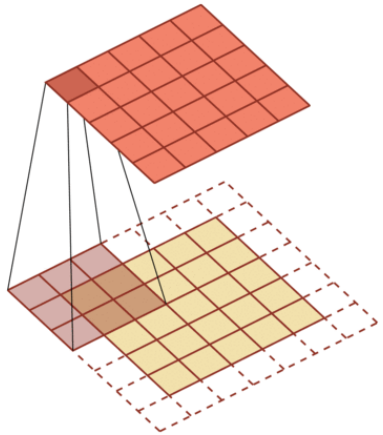
卷积运算 二维卷积



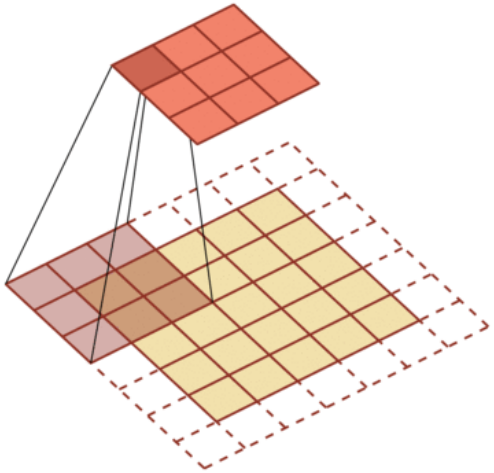
步长1，零填充0



步长2，零填充0



步长1，零填充1



步长2，零填充1

$$W_2 = (W_1 - F + 2P)/S + 1$$
$$H_2 = (H_1 - F + 2P)/S + 1$$

在上面的两个公式中，W_2是卷积后的feature_map的宽度， W_1是卷积前图像的宽度；F是滤波器的宽度，P是零填充的数量（零填充：指在原始图像周围补几圈0，如果P的值是1，那么就补1圈0）；S是步长；H_2是卷积后的feature_map的高度；H_1是卷积前图像的高度。

卷积运算 卷积运算与互相关运算

- 在机器学习和图像处理领域，卷积的主要功能是在一个图像（或某种特征）上滑动一个卷积核（即滤波器），通过卷积操作得到一组新的特征。在计算卷积的过程中，需要进行卷积核翻转。在具体实现上，一般会以互相关操作来代替卷积，从而会减少一些不必要的操作或开销。
- 互相关（Cross-Correlation）是一个衡量两个序列相关性的函数，通常是用滑动窗口的点积计算来实现。
- 给定一个图像$\boldsymbol{X} \in \boldsymbol{R}^{M \times N}$和一个卷积核$\boldsymbol{W} \in \boldsymbol{R}^{U \times V}$，它们的互相关为

$$y_{ij} = \sum_{u=1}^{U} \sum_{v=1}^{V} w_{uv} x_{i+u-1, j+v-1}.$$

卷积运算 卷积运算与互相关运算

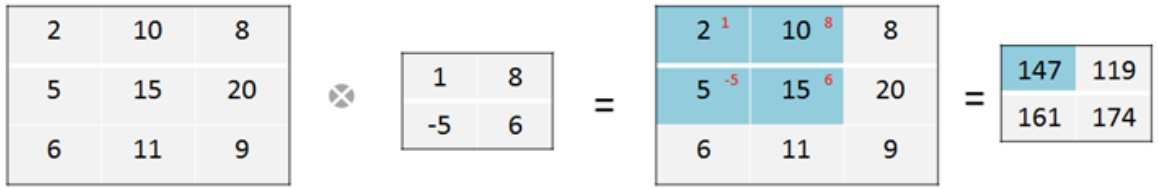
互相关：

$$y_{ij} = \sum_{u=1}^{U} \sum_{v=1}^{V} w_{uv} x_{i+u-1,j+v-1}$$

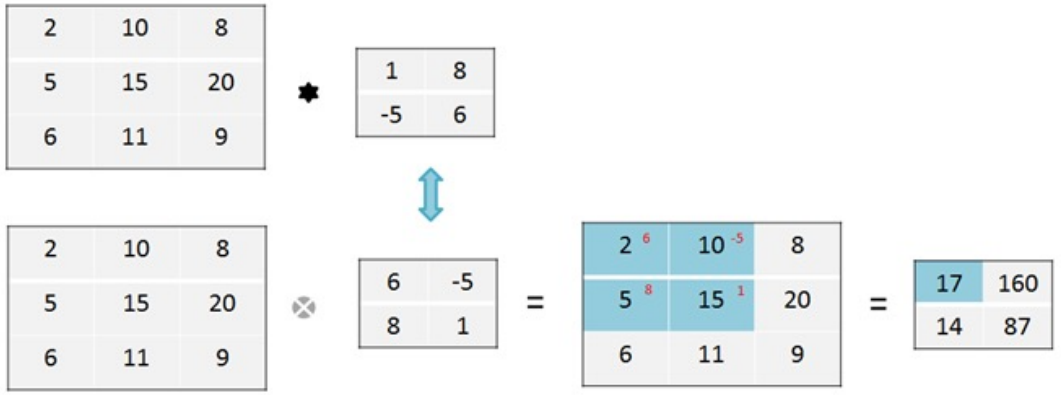
卷 积：

$$y_{ij} = \sum_{u=1}^{U} \sum_{v=1}^{V} w_{uv} x_{i-u+1,j-v+1}$$

协相关：是把核与输入数据对应相乘再求和



卷积：是把核先反转180，再作协相关



两个公式相比较可知，互相关和卷积的区别仅仅在于卷积核是否进行翻转。
因此互相关也可以称为不翻转卷积。

卷积运算 卷积运算与互相关运算

给定一个输入信息$\boldsymbol{X}$和滤波器$\boldsymbol{W}$ 的二维互相关定义为：

$$\boldsymbol{Y} = \boldsymbol{W} \otimes \boldsymbol{X} = \text{rot180}(\boldsymbol{W}) * \boldsymbol{X},$$

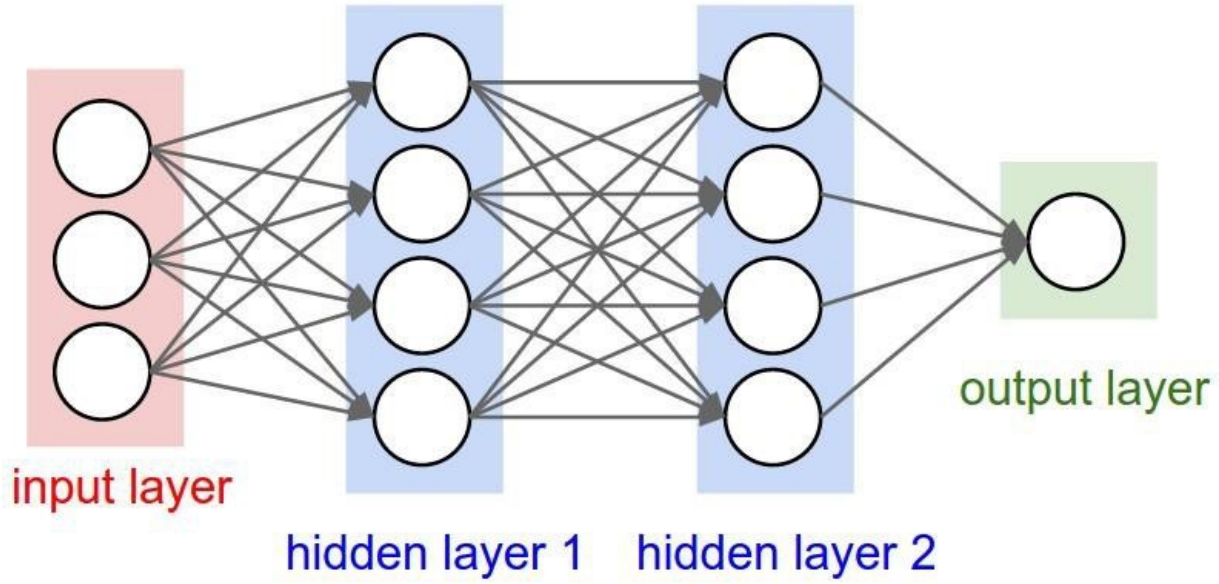
其中⊗表示互相关运算，rot180(·) 表示旋转180 度，$\boldsymbol{Y} \in \mathrm{R}^{M-U+1,N-V+1}$为输出矩阵。

在神经网络中使用卷积是为了进行特征抽取，卷积核是否进行翻转和其特征抽取的能力无关。特别是当卷积核是可学习的参数时，卷积和互相关在能力上是等价的。因此，为了实现上（或描述上）的方便起见，我们用互相关来代替卷积。事实上，很多深度学习工具中卷积操作其实都是互相关操作。

章节目录 CONTENTS

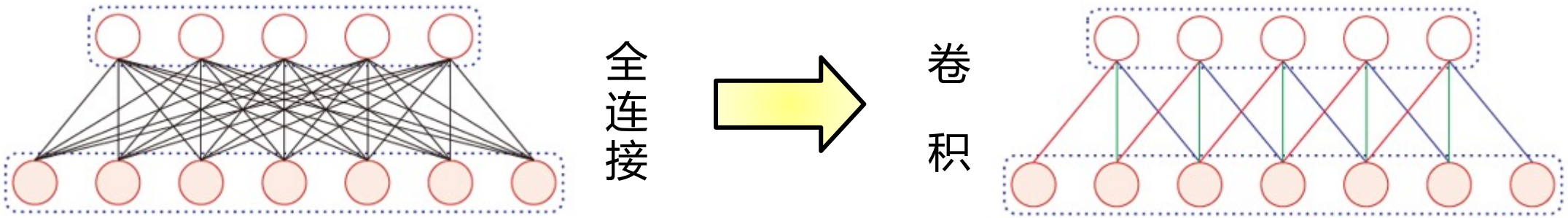
卷积的动机

在全连接前馈神经网络中，如果第l层有M_l个神经元，第$l-1$层有M_{l-1}个神经元，则连接边有$\mathrm{M}_l \times \mathrm{M}_{l-1}$个，也就是权重矩阵有$\mathrm{M}_l \times \mathrm{M}_{l-1}$参数。当$\mathrm{M}_l$和$\mathrm{M}_{l-1}$都很大时，权重矩阵的参数非常多，训练的效率会非常低。



卷积的动机

在卷积层（假设是第l层）中的每一个神经元都只和下一层（第$l-1$层）中某个局部窗口内的神经元相连，构成一个局部连接网络。这样，卷积层和下一层之间的连接数大大减少，由原来的$\mathrm{M}_l \times \mathrm{M}_{l-1}$个连接变为$\mathrm{M}_l \times K$个连接，$K$ 为卷积核大小。



稀疏交互（局部连接、局部感受野）

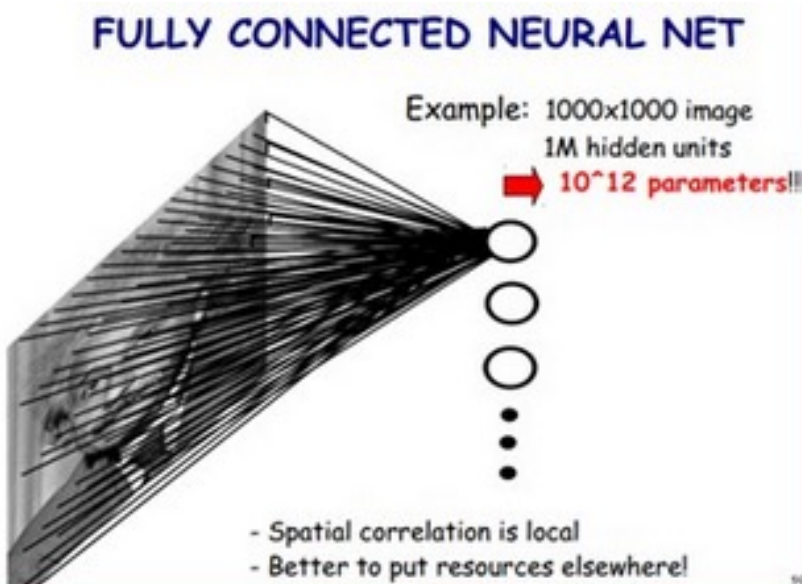
卷积的动机

给定1000x1000像素的图像，假设神经网络具有1百万个隐层神经元，则全连接需要10^12个权值参数：

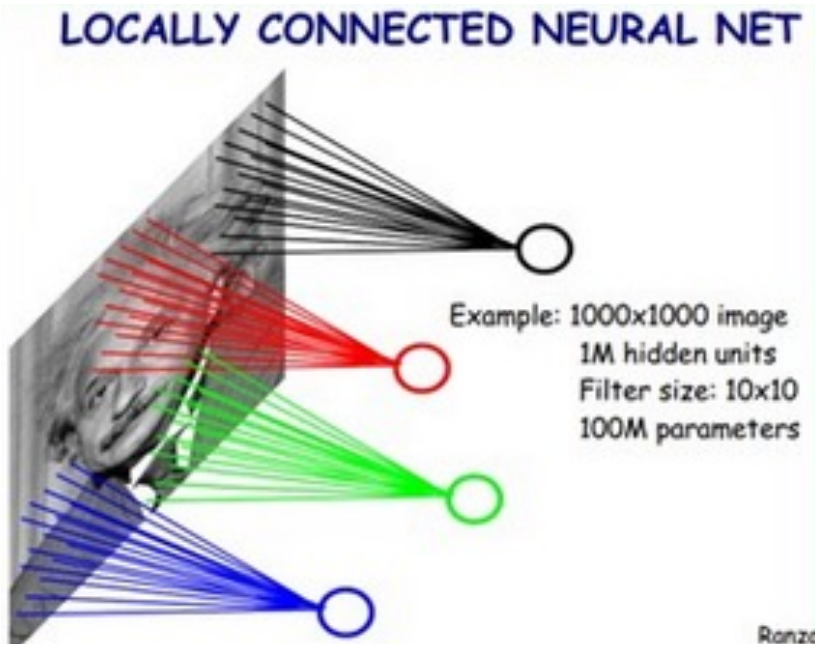
1000x1000x1000000=10^12

局部感受野是10x10，隐层每个感受野只需要和这10x10的局部图像相连接，所以1百万个隐层神经元就只有一亿个连接，即10^8个参数：

10X10X1000000=10^8



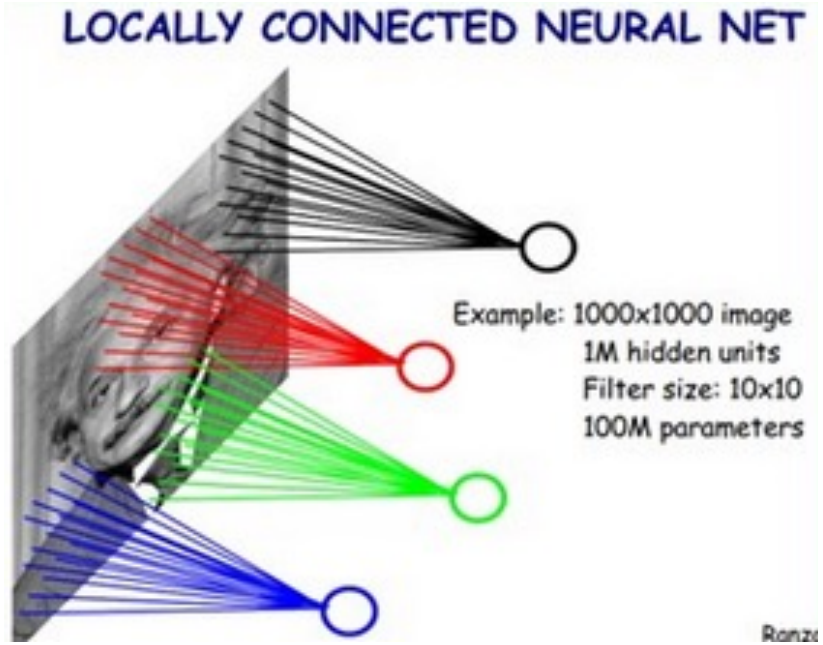
少了4个0（数量级）



卷积的动机

局部感受野是10x10，隐层每个感受野只需要和这10x10的局部图像相连接，所以1百万个隐层神经元就只有一亿个连接，即10^8个参数：

10X10X1000000=10^8



权值共享

每一个神经元存在10x10=100个连接权值参数。那如果我们每个神经元这100个参数是相同的呢？也就是说每个神经元用的是同一个卷积核去卷积图像。

只需：100个参数

问：这样的后果是什么呢?

答：这样只提取了一种特征

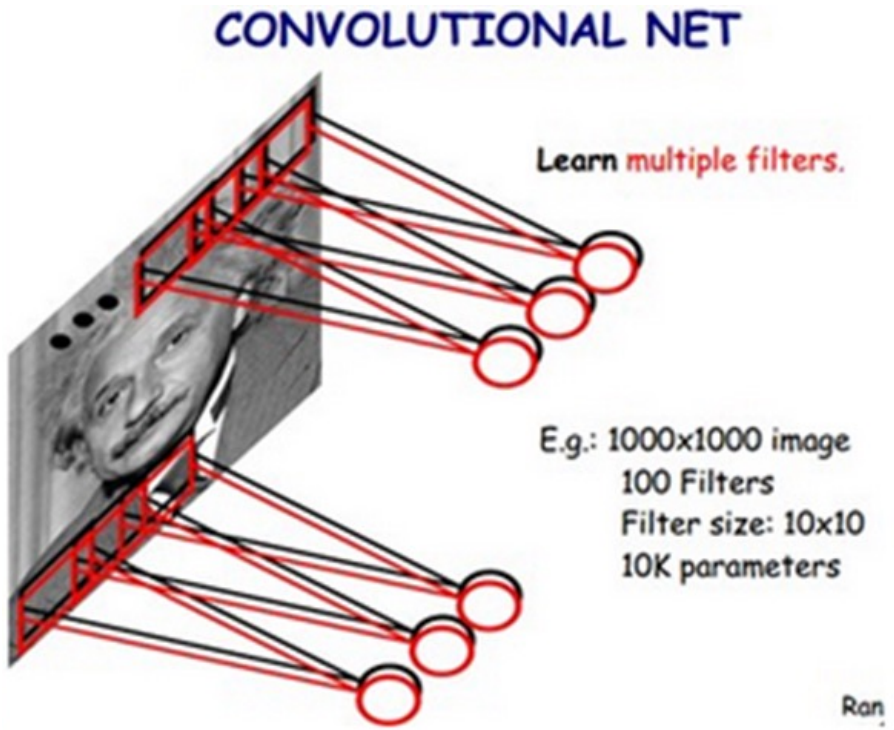
权重共享：作为参数的卷积核$w(l)$对于第l层的所有的神经元都是相同的。如下图中，所有的同颜色连接上的权重是相同的。权重共享可以理解为一个卷积核只捕捉输入数据中的一种特定的局部特征。因此，如果要提取多种特征就需要使用多个不同的卷积核。

假如一种滤波器，也就是一种卷积核就是提出图像的一种特征，那么我们如需要提取不同的特征，怎么办?

答：加多几种滤波器。

假设有100种滤波器，每种滤波器的参数不一样，表示它提取输入图像的不同特征。所以100种卷积核就有100个Feature Map（我们定义由同一种滤波器卷积得到的向量组合，为Feature Map）。

问：这时我们这一层有多少个参数了?



答：100种卷积核（不同特征）x 每种卷积核共享的100个参数
=100x100=10K，也就是1万个参数。

卷积的动机

除此之外，卷积网络还具有**平移不变性**，它是由卷积+池化共同实现的。在欧几里得几何中，平移是一种几何变换，表示把一幅图像或一个空间中的每一个点在相同方向移动相同距离。比如对图像分类任务来说，图像中的目标不管被移动到图片的哪个位置，得到的结果（标签）应该是相同的，这就是卷积神经网络中的平移不变性。

平移不变性意味着系统产生完全相同的响应（输出），不管它的输入是如何平移的。

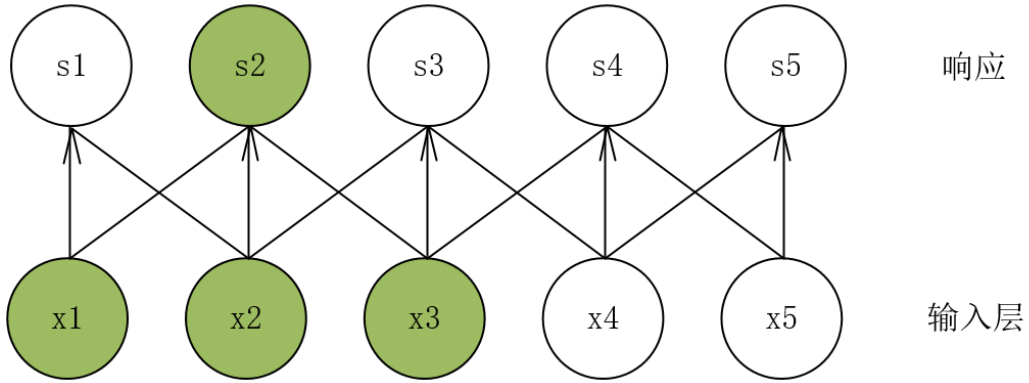
卷积：简单地说，图像经过平移，相应的特征图上的表达也是平移的，即平移不变性。

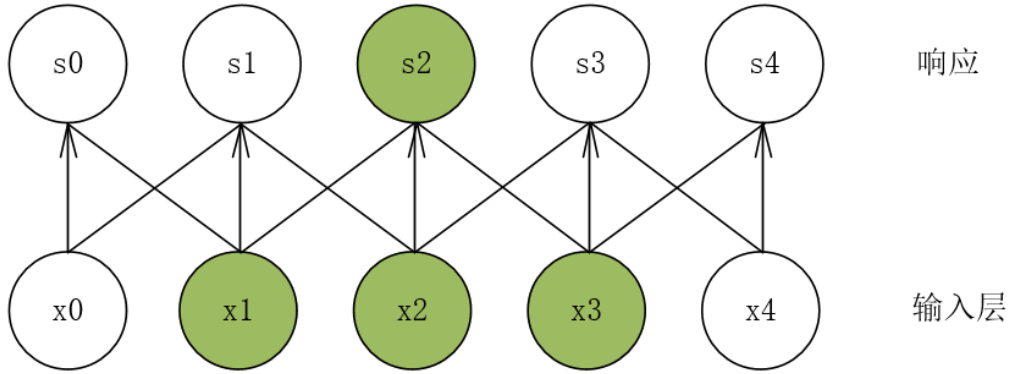
池化：比如最大池化，它返回感受野中的最大值，如果最大值被移动了，但是仍然在这个感受野中，那么池化层也仍然会输出相同的最大值。

这两种操作共同提供了一些平移不变性，即使图像被平移，卷积保证仍然能检测到它的特征，池化则尽可能地保持一致的表达。

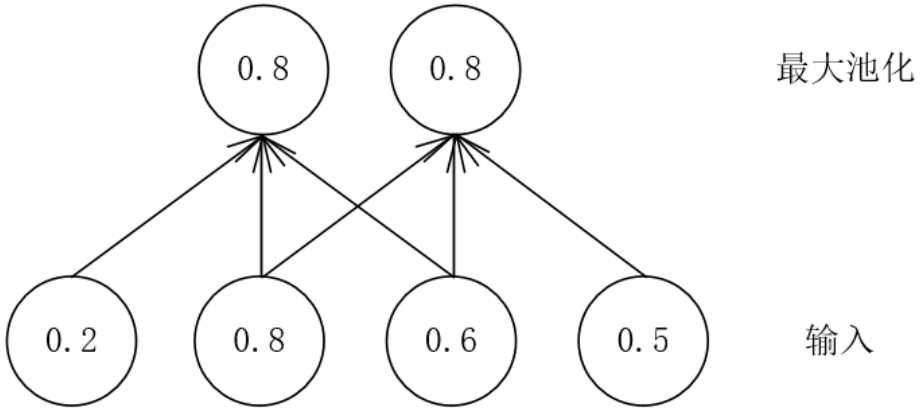
卷积的动机

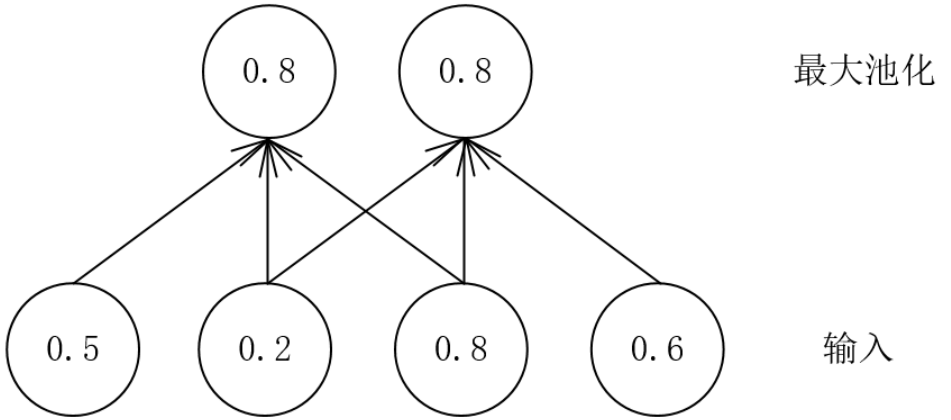
卷积的平移不变性





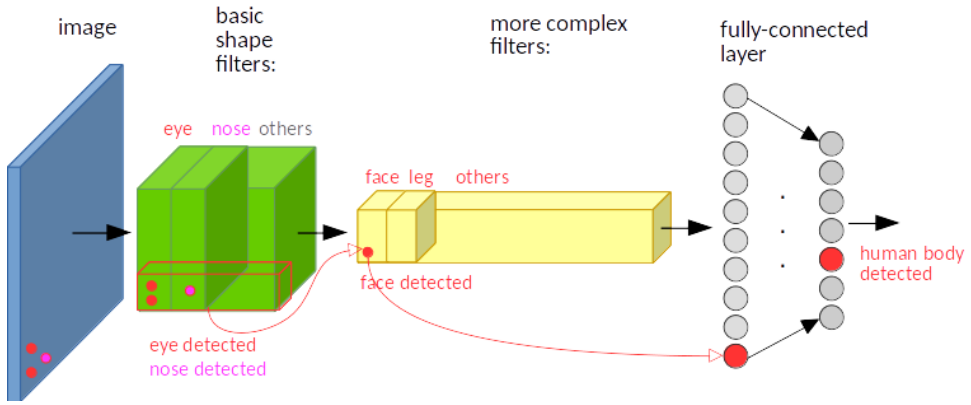
池化的平移不变性



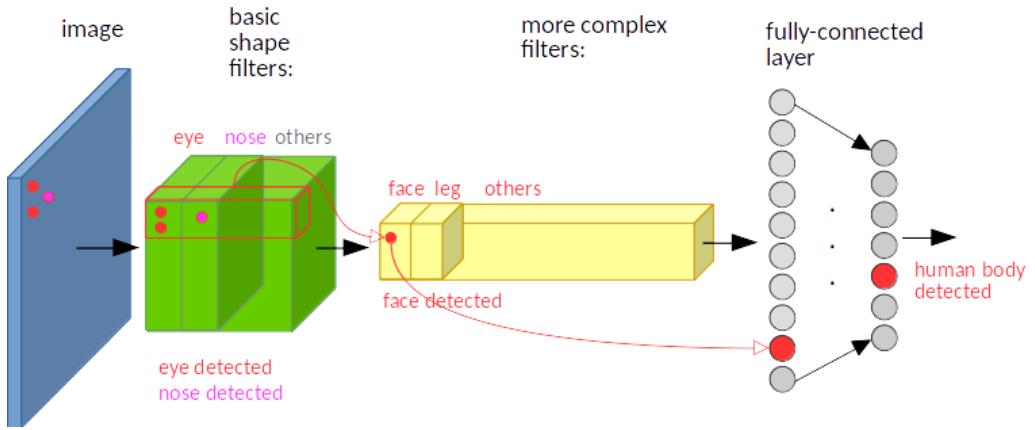


卷积的动机

输入图像的左下角有一个人脸，经过卷积，人脸的特征（眼睛，鼻子）也位于特征图的左下角。



假如人脸特征在图像的左上角，那么卷积后对应的特征也在特征图的左上角。



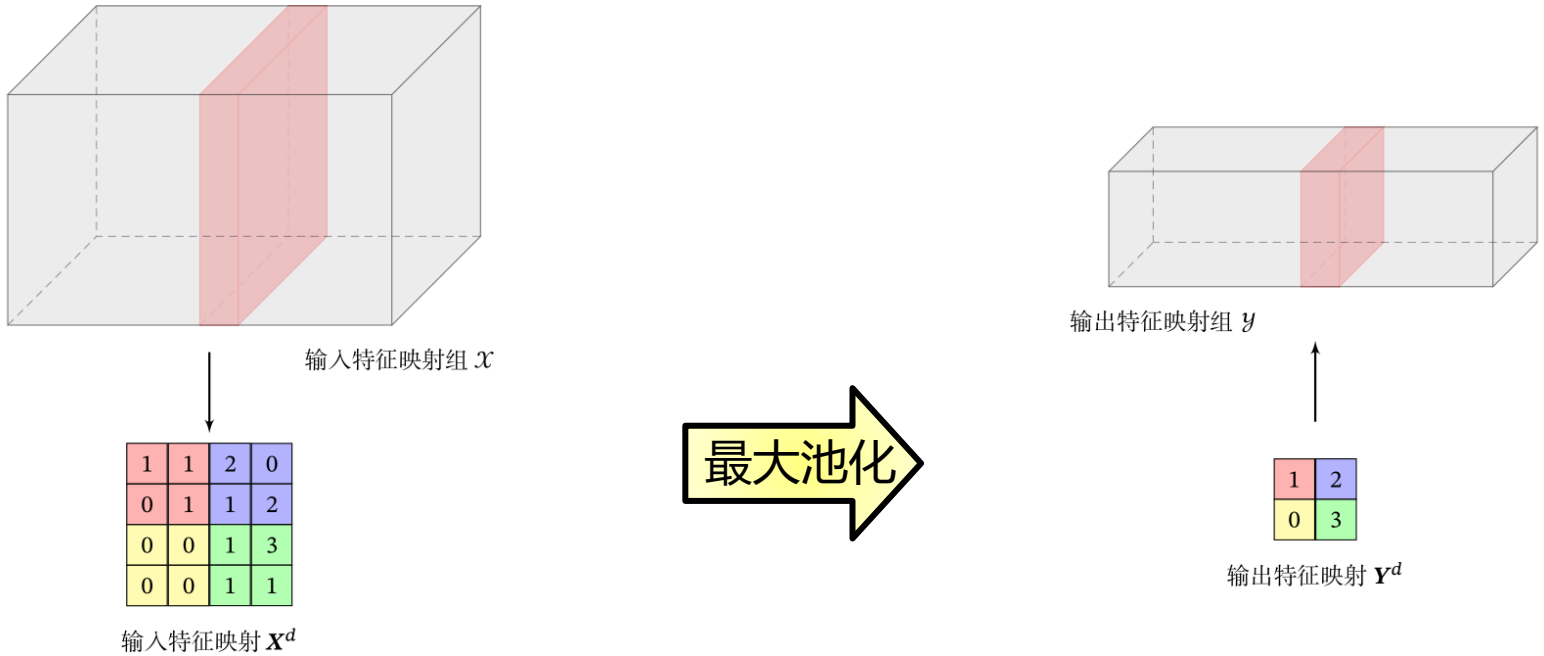
章节目录 CONTENTS

池化

- 卷积层虽然可以显著减少网络中连接的数量，但特征映射组中的神经元个数并没有显著减少。如果后面接一个分类器，分类器的输入维数依然很高，很容易出现过拟合。为了解决这个问题，可以在卷积层之后加上一个池化层，从而降低特征维数，避免过拟合。
- 池化层（Pooling Layer）也叫子采样层（Subsampling Layer），其作用是进行特征选择，降低特征数量，从而减少参数数量。

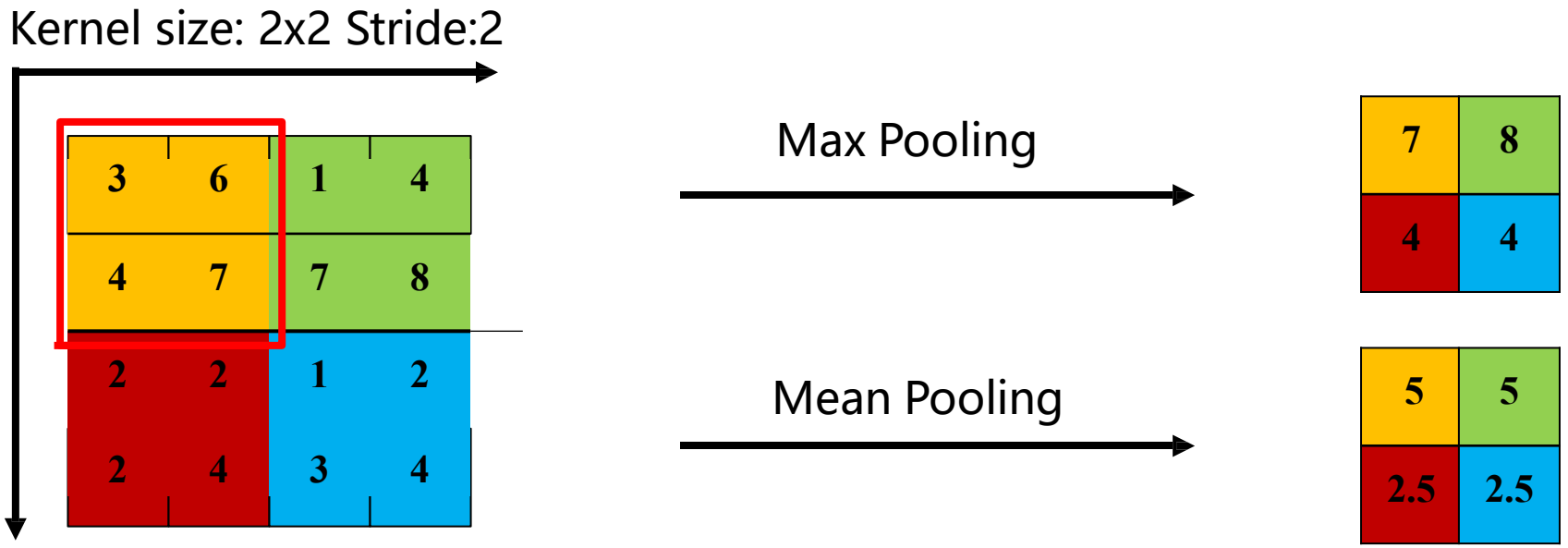
池化

假设池化层的输入特征映射组为$\mathcal{X} \in R^{M\times N\times D}$，对于其中每一个特征映射$X^d \in R^{M\times N}, 1 \le d \le D$，将其划分为很多区域$R^d_{m,n}$，$1 \le m \le M', 1 \le n \le N'$，这些区域可以重叠，也可以不重叠。池化（Pooling）是指对每个区域进行下采样（Down Sampling）得到一个值，作为这个区域的概括。



池化

最大池化(Maximum Pooling 或 Max Pooling)：对于一个区域 $R_{m,n}^d$，选择这个区域内所有神经元的最大活性值作为这个区域的表示：$y_{m,n}^d = \max_{i \in R_{m,n}^d} x_i$.



平均池化(Mean Pooling)：一般是取区域内所有神经元活性值的平均值：$y_{m,n}^d = \frac{1}{|R_{m,n}^d|} \sum_{i \in R_{m,n}^d} x_i$

其中 x_i 为区域 R_k^d 内每个神经元的活性值。

池化

平均池化：主要用来抑制邻域值之间差别过大，造成的方差过大。如，输入（2,10），通过均值池化后是（6），对于输入的整体信息保存的很好

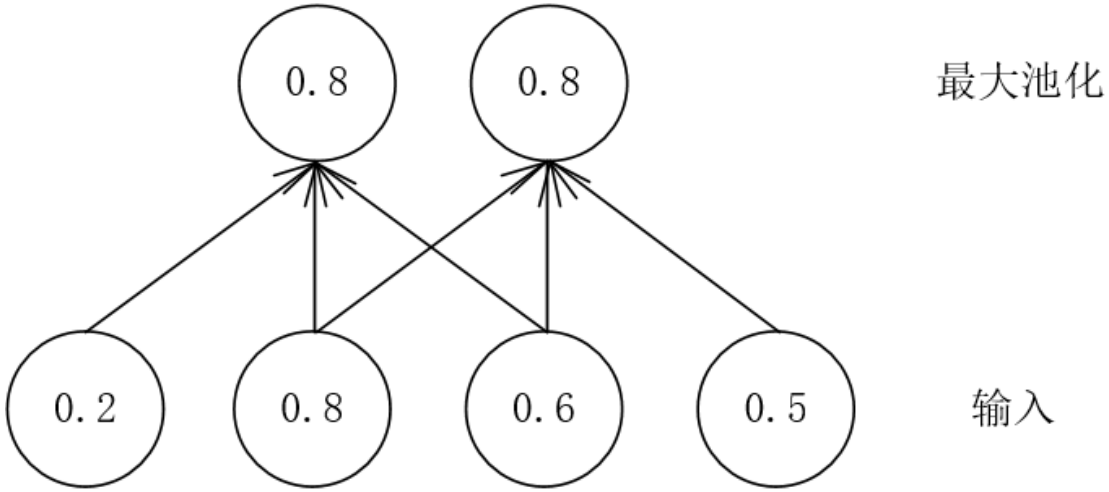
- 在计算机视觉中：对背景的保留效果好!

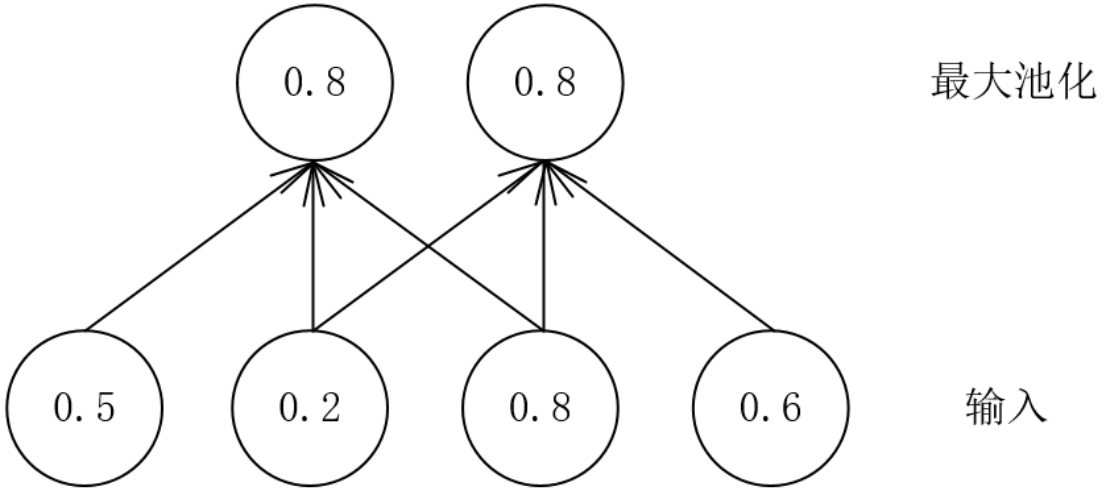
最大池化：能够抑制网络参数误差造成的估计均值偏移的现象。如，输入（1,5,3），最大池化后是（5），假如输入中的参数1，有误差，变为了1.5，这时输入是（1.5,5,3），最大池化后结果还是（5）

- 在计算机视觉中：对纹理的提取较好!

池化

- 更好的获取平移不变性

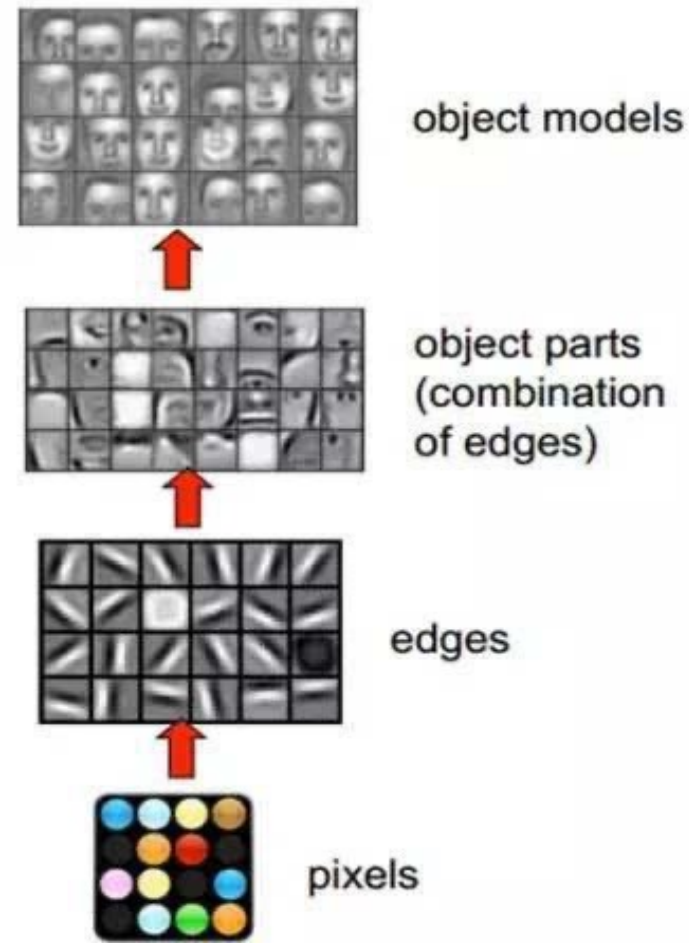




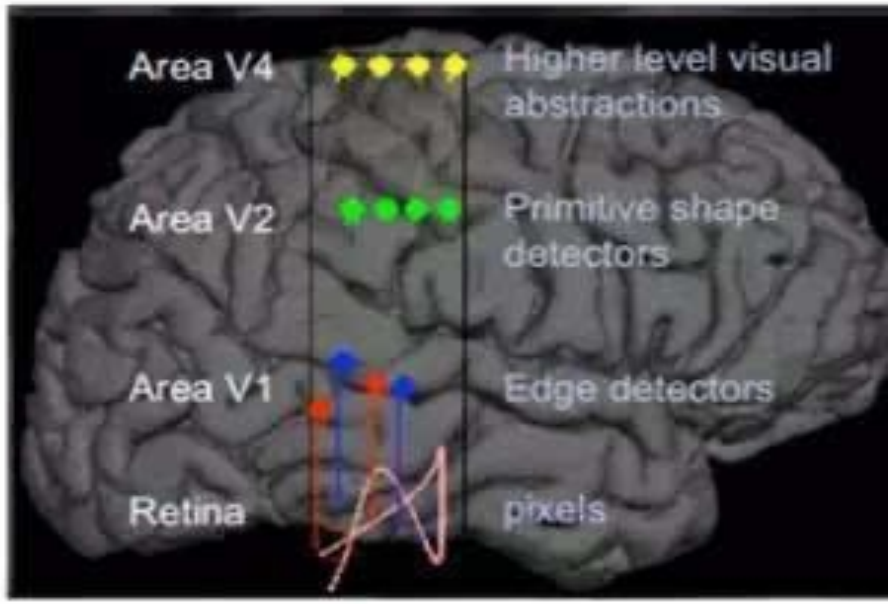
- 更高的计算效率（减少了神经元数）

章节目录 CONTENTS

卷积神经网络



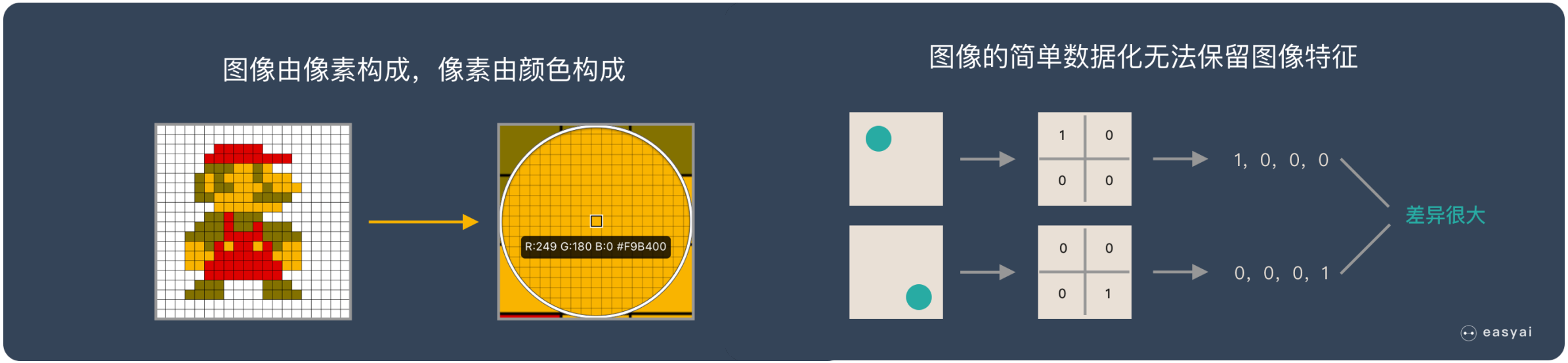
动物和人的大脑如何识别图像？



卷积神经网络　出现原因

在 CNN 出现之前，图像对于人工智能来说是一个难题，有2个原因：

- 图像在数字化的过程中很难保留原有的特征，导致图像处理的准确率不高
- 图像需要处理的数据量太大，导致成本很高，效率很低



卷积神经网络　出现原因

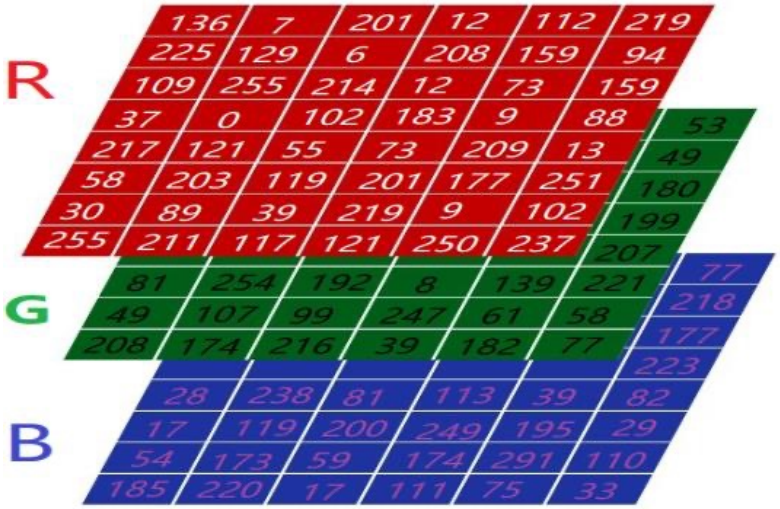
人眼看到的图片

What We See

计算机看到的图片

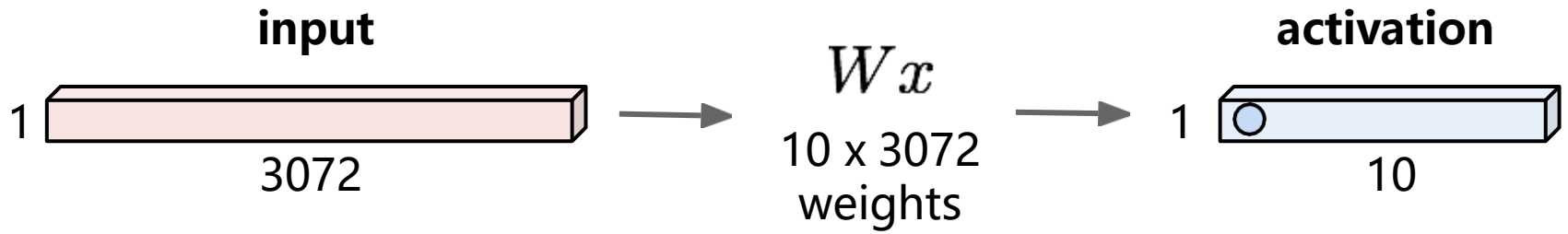
What Computers See

RGB三通道
二维数组

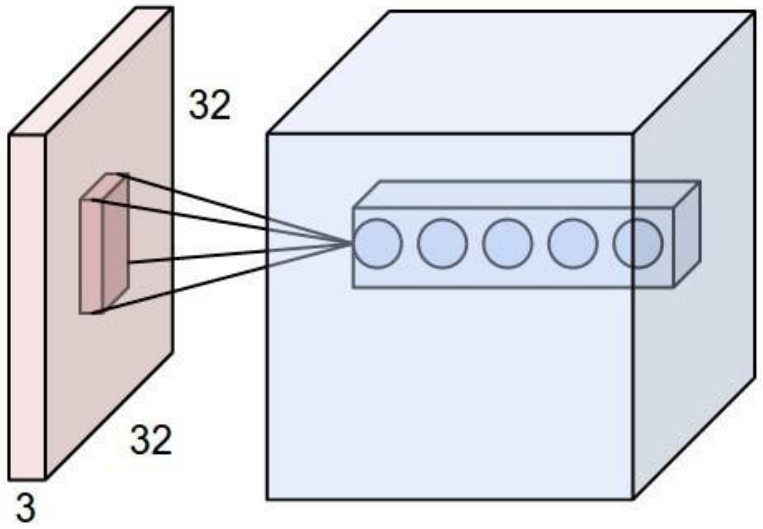


卷积神经网络　出现原因

全连接层　32x32x3 image -> stretch to 3072 x 1



CNN 解决了这个问题，用类似视觉的方式保留了图像的特征，当图像做翻转、旋转或者变换位置时，它也能有效的识别出来是类似的图像。



卷积神经网络　一般结构框架

卷积层

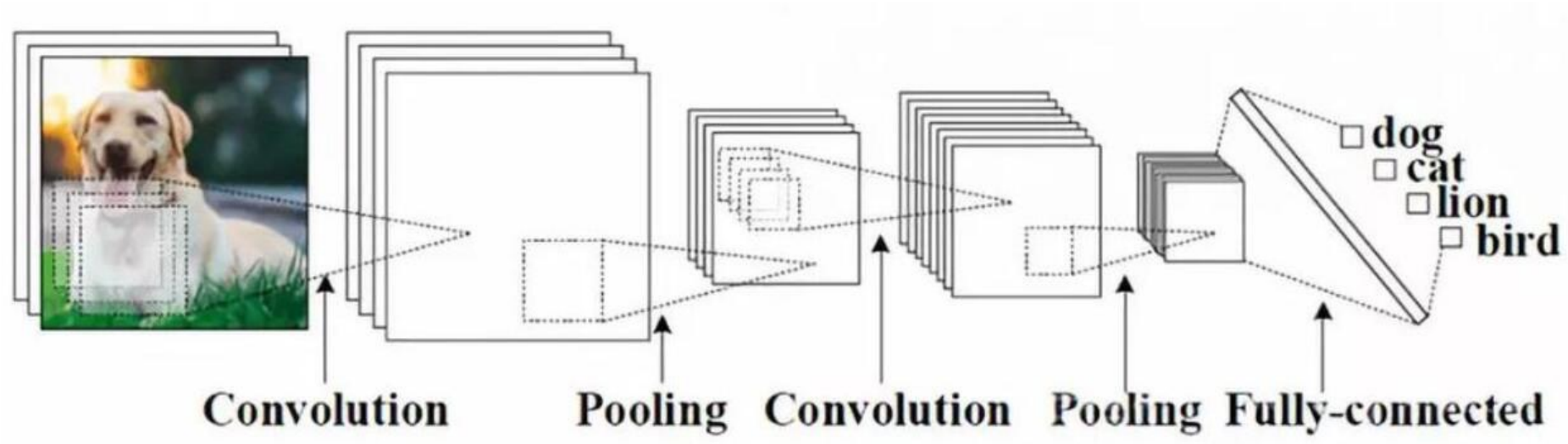
- 提取特征

池化层

- 降维、防止过拟合

全连接层

- 输出结果



卷积神经网络　一般结构框架：卷积层

卷积层

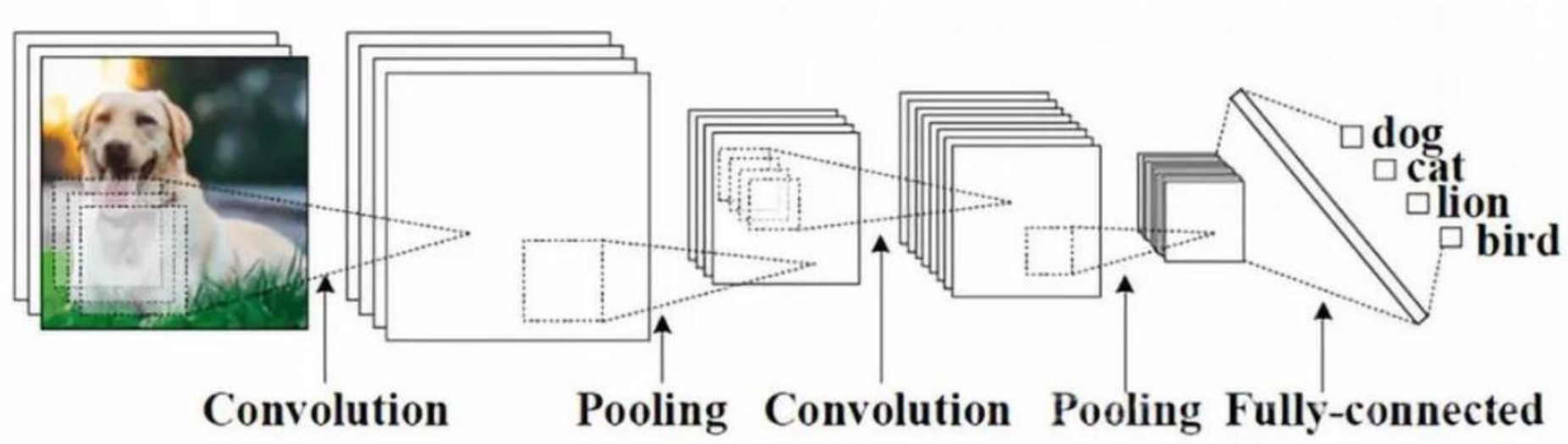
- 提取特征

池化层

- 降维、防止过拟合

全连接层

- 输出结果



卷积神经网络　一般结构框架：卷积层

卷积层——提取特征

=

Blue

Green

Red

123	94	83	2
123	94	83	4
123	94	83	2
34	44	187	92
34	76	232	124
67	83	194	202

30 124 142

按0填充

pad

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	123	94	83	2	0	0
0	0	34	44	187	92	0	0
0	0	34	76	232	124	0	0
0	0	67	83	194	202	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

pad

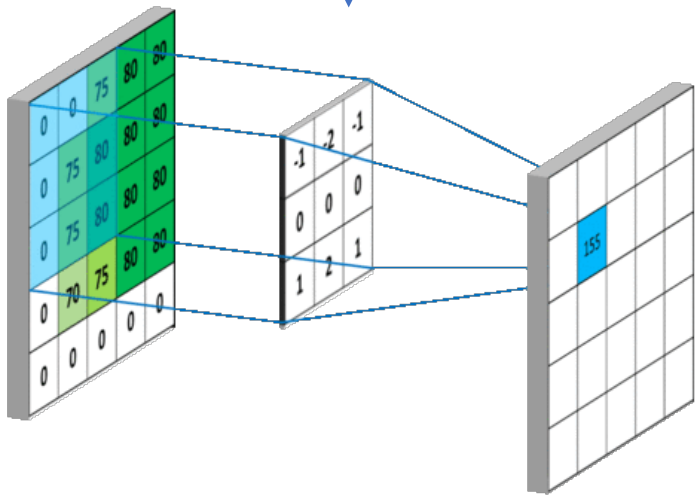
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	123	94	2	4	0	0
0	0	11	3	22	192	0	0
0	0	12	4	23	34	0	0
0	0	194	83	12	94	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

pad

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	123	94	83	2	0	0
0	0	34	44	37	30	0	0
0	0	34	114	234	124	0	0
0	0	49	18	204	142	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

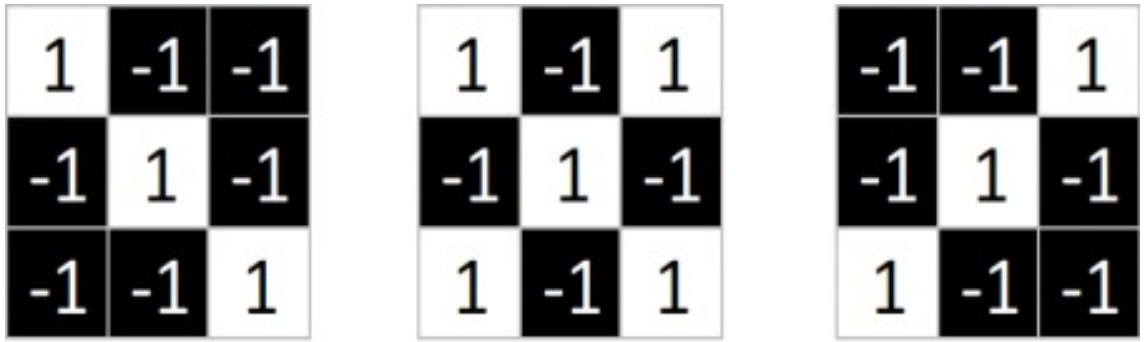
pad

卷积核



卷积神经网络　一般结构框架：卷积层

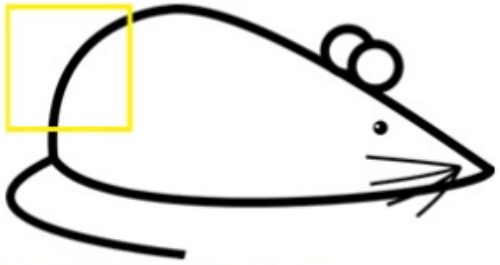
卷积核是什么?



卷积核Kernel也叫滤波器filter，代表图像的某种特征；也称为神经元。比如垂直边缘、水平边缘、颜色、纹理等等，这些所有神经元加起来就好比就是整张图像的特征提取器集合。卷积核越深越能检测图像更高级别、更高层次、更复杂、更抽象、更泛化的特征。

卷积神经网络　一般结构框架：卷积层

Original image



Visualization of the filter on the image

Visualization of the receptive field

0	0	0	0	0	0	30
0	0	0	0	50	50	50
0	0	0	20	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0

Pixel representation of the receptive field

*

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = (50*30)+(50*30)+(50*30)+(20*30)+(50*30) = 6600 (A large number!)

Visualization of the filter on the image

0	0	0	0	0	0	0
0	40	0	0	0	0	0
40	0	40	0	0	0	0
40	20	0	0	0	0	0
0	50	0	0	0	0	0
0	0	50	0	0	0	0
25	25	0	50	0	0	0

Pixel representation of receptive field

*

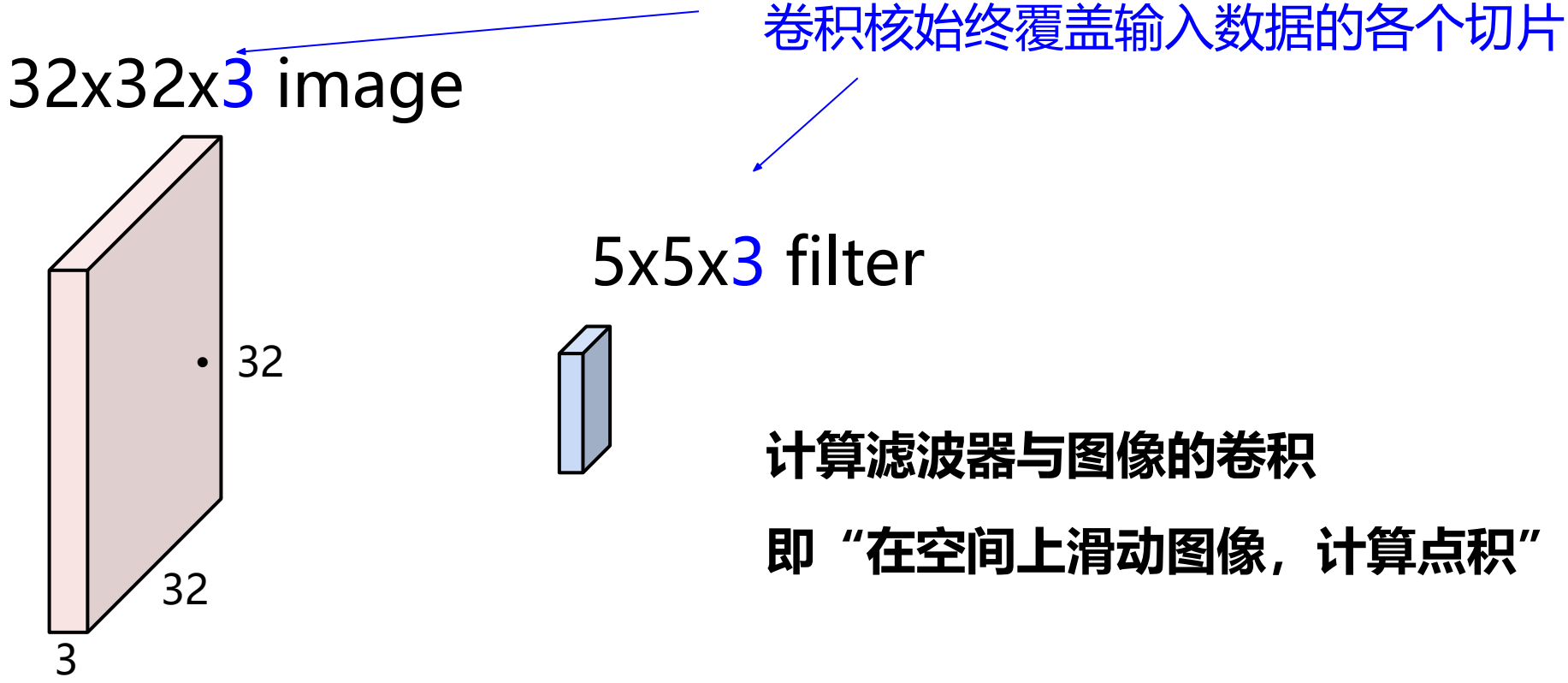
0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = 0

结论：相似则输出一个明显变大的值，否则输出极小值。

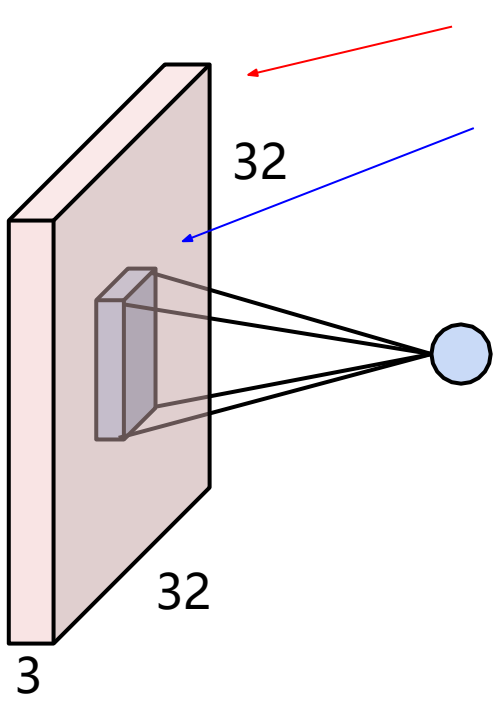
卷积核深度应该与输入一致



计算滤波器与图像的卷积

即“在空间上滑动图像，计算点积”

利用卷积核进行卷积计算



32x32x3 image

5x5x3 filter w

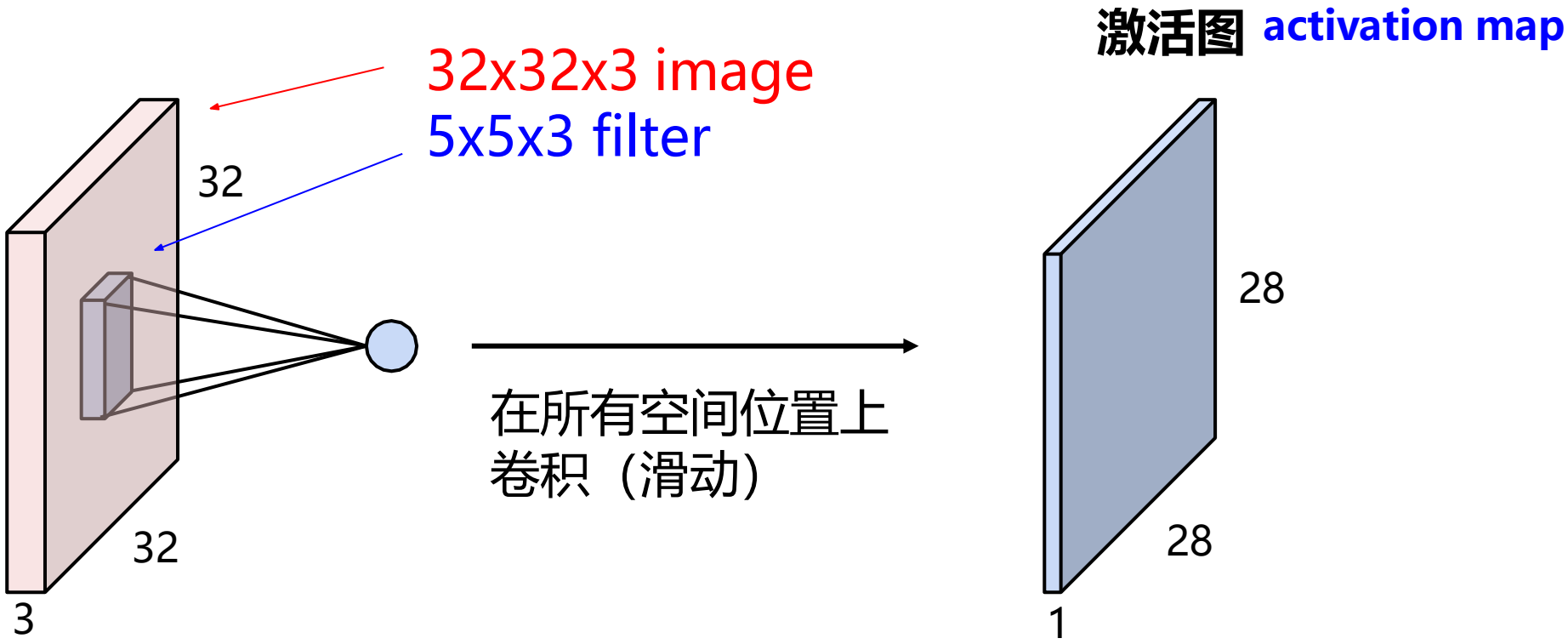
卷积结果为一个标量

在滤波器和图像的5x5x3小块之间取点积的结果

（即5 * 5 * 3 = 75维点积+偏差）

$$w^T x + b$$

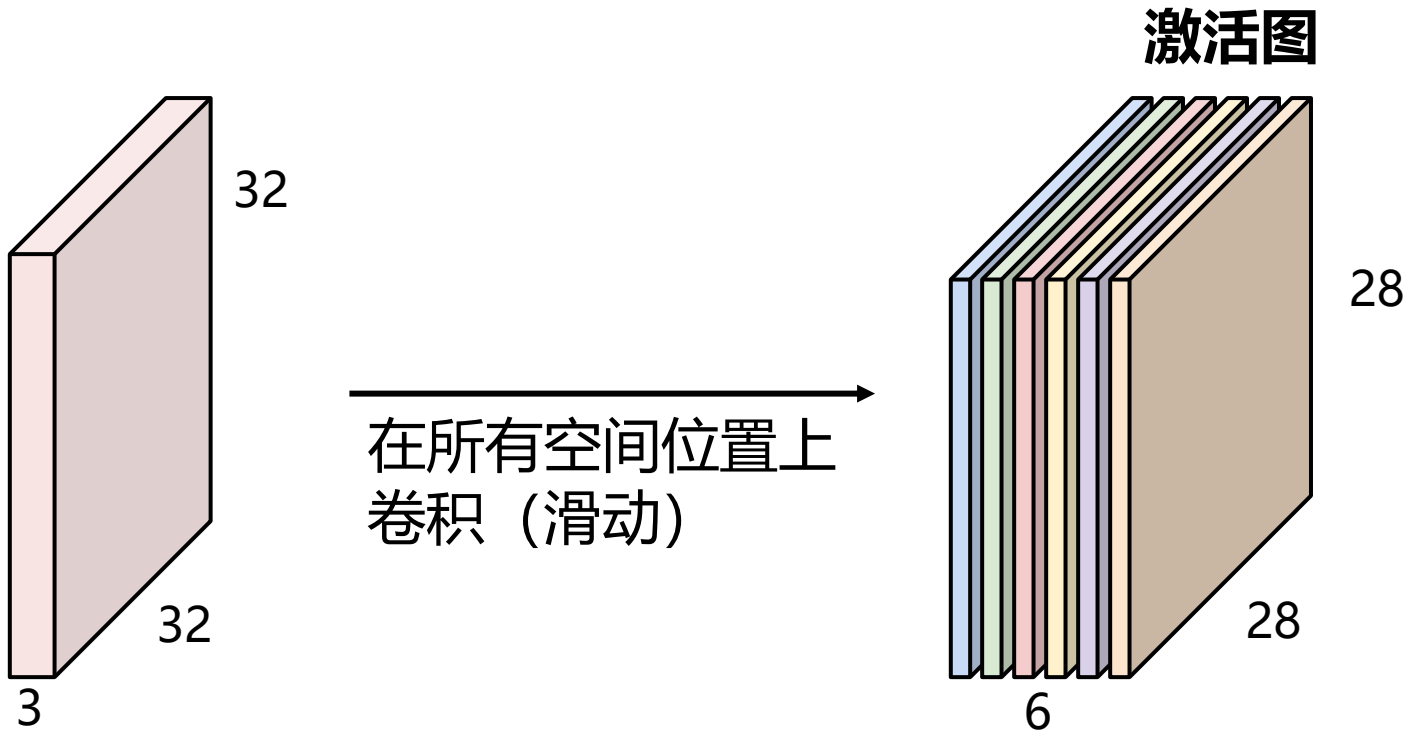
卷积结果



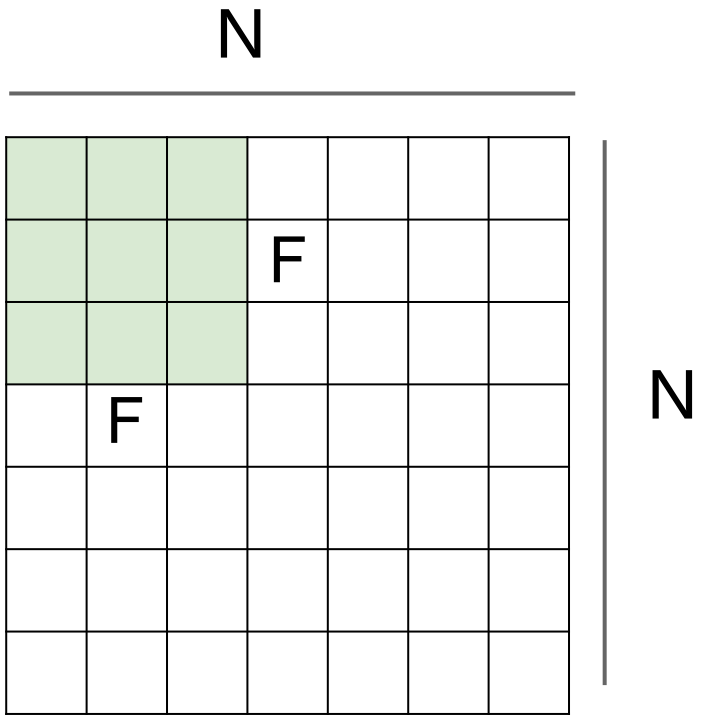
28=(32-5)/1+1

卷积结果：考虑多个滤波器

因此，如果我们有6个5×5×3滤波器，我们将获得6个单独的激活图：



我们将它们堆叠起来，以获得尺寸为28x28x6的“新图片”！

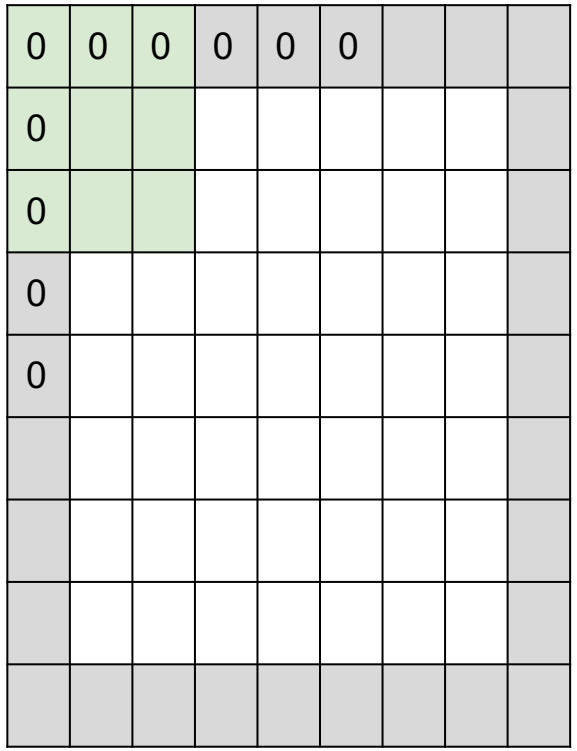


输出大小：
(N - F) / stride + 1

例如. N = 7, F = 3:
步长 1 => (7 - 3)/1 + 1 = 5
步长 2 => (7 - 3)/2 + 1 = 3
步长 3 => (7 - 3)/3 + 1 = 2.33 :\

卷积神经网络　一般结构框架：卷积层

在实际中：常用零填充边框



例如. 输入图片大小为 7x7

3x3 滤波器, **步长**设置为**1**

具有1个像素边框的填充 => 输出大小为?

7x7!

通常，常见的情况是，卷积层步长设置为1，滤波器大小为FxF，则一般使用（F-1）/ 2个像素进行零填充（将在空间上保留大小）

例如 F = 3 => zero pad with 1

F = 5 => zero pad with 2

F = 7 => zero pad with 3

(N – F+2P) / stride + 1

样例：

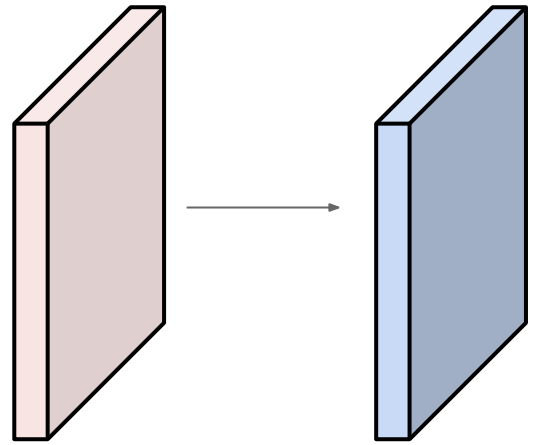
输入图片大小: **32x32x3**

10个卷积核，大小为 5x5x3，步长为1, pad = 2

输出图片大小:

(32-5+2*2)/1+1 = 32

因此输出大小为 **32x32x10**



样例：

输入图片大小: **32x32x3**

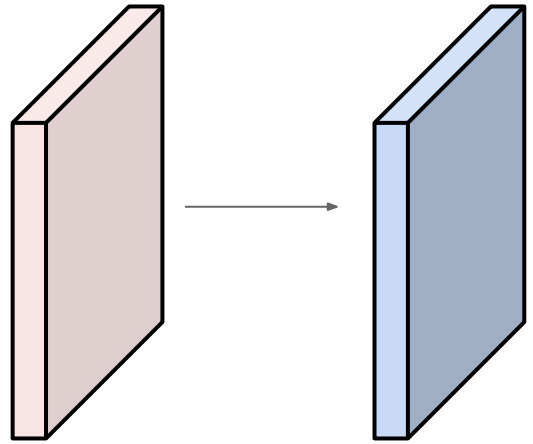
10个卷积核，大小为5x5x3，步长为1, pad = 2

这一层中的参数数量?

每个滤波器具有

5 * 5 * 3 +1 = 76个参数(+1 for bias)

=> 76 * 10 = **760**



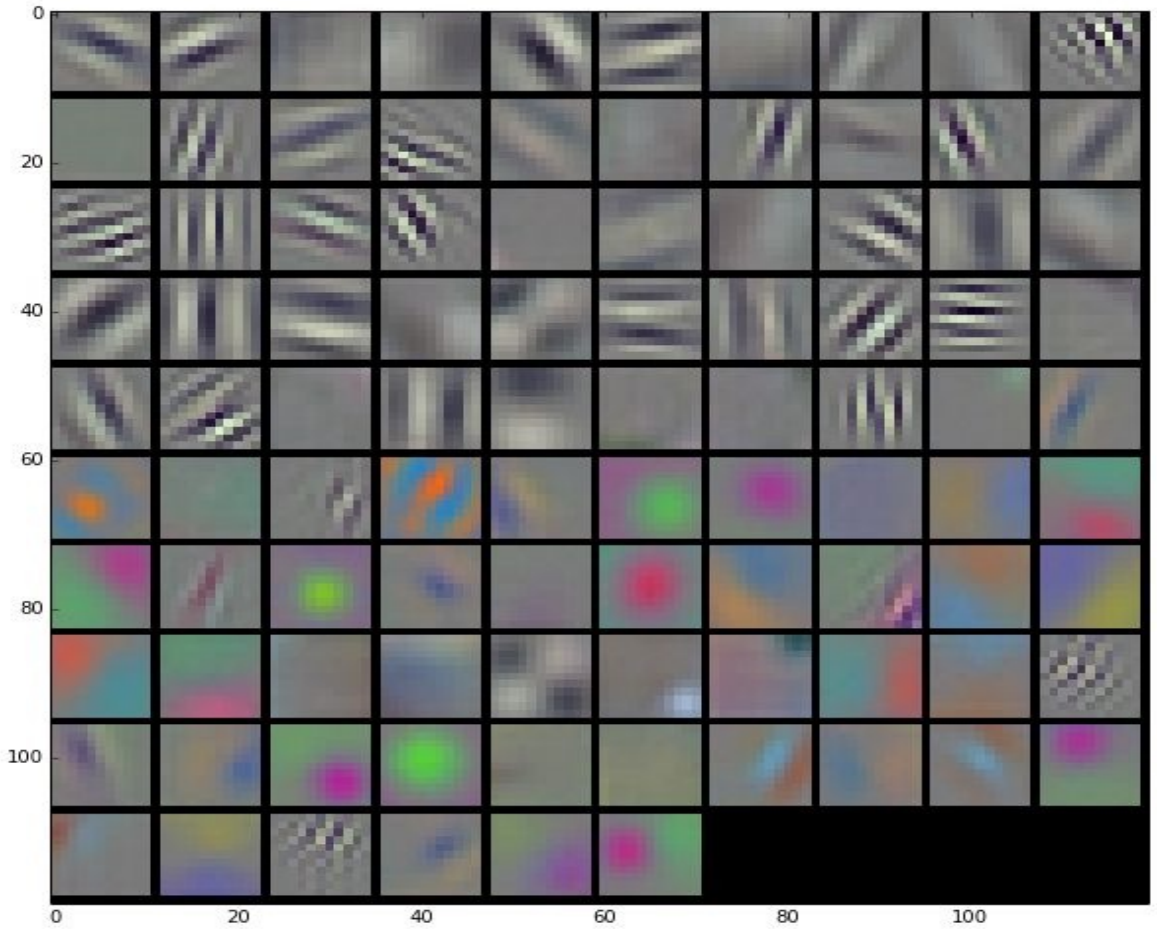
总结： 给定一个卷积层

- 需要四个参数
  - ✓ 滤波器数目K
  - ✓ 滤波器大小F
  - ✓ 步长S
  - ✓ 零填充个数P
- 输入图片大小为$W_1 \times H_1 \times D_1$
- 经过卷积后输出大小为$W_2 \times H_2 \times D_2$，其中
  - ✓ $W_2 = (W_1 - F + 2P)/S + 1$
  - ✓ $H_2 = (H_1 - F + 2P)/S + 1$
  - ✓ $D_2 = K$
- 通过参数共享，每个滤波器引入$F \cdot F \cdot D_1$个参数，因此一共有$(F \cdot F \cdot D_1) \cdot K + K$个参数
- 输出结果中，第d个切片（大小为$W_2 \times H_2$）是对输入通过第d个滤波器以步长S做卷积然后加上$bias$的结果

卷积神经网络　一般结构框架：卷积层

不同的卷积核多角度检测图像特征

每一个卷积核代表提取不同的特征，多个卷积核提取的特征然后进行组合。



可视化卷积核

卷积神经网络

特征图可视化

从上图中可知，浅层学到的特征为简单的边缘、角点、纹理、几何形状、表面等，到深层学到的特征则更为复杂抽象，为狗、人脸、键盘等等，有几点需要注意：

1. 卷积神经网络每层的卷积核权重是由数据驱动学习得来，不是人工设计的，人工只能胜任简单卷积核的设计，像边缘，但描述复杂模式的卷积核则十分困难。

Zeiler M D , Fergus R . Visualizing and Understanding Convolutional Networks. ECCV 2014.

卷积神经网络

特征图可视化

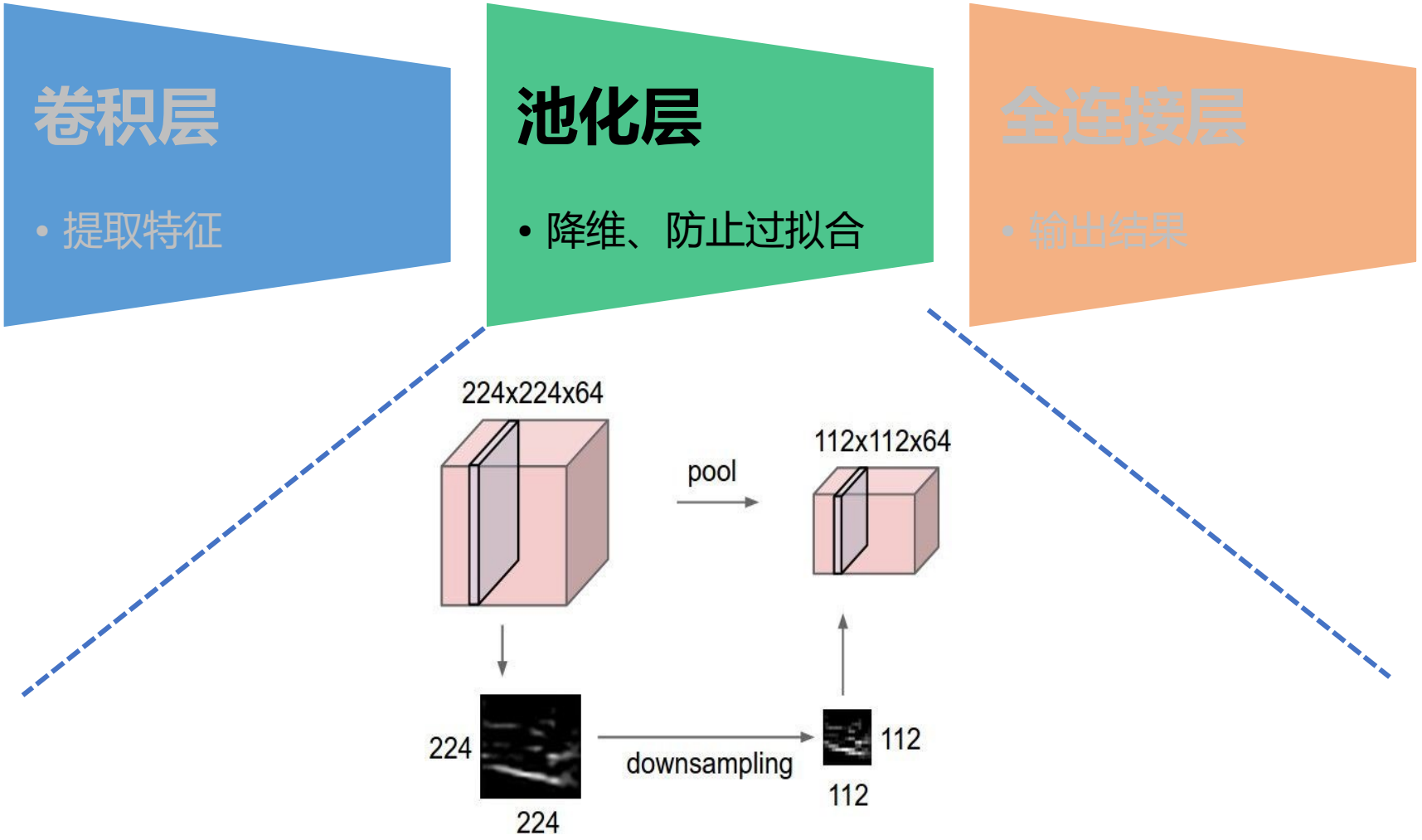
2. 数据驱动卷积神经网络逐层学到由简单到复杂的特征（模式），复杂模式是由简单模式组合而成，比如Layer4的狗脸是由Layer3的几何图形组合而成，Layer3的几何图形是由Layer2的纹理组合而成，Layer2的纹理是由Layer1的边缘组合而成，从特征图上看的话，Layer4特征图上一个点代表Layer3某种几何图形或表面的组合，Layer3特征图上一个点代表Layer2某种纹理的组合，Layer2特征图上一个点代表Layer1某种边缘的组合。

卷积神经网络

特征图可视化

3. 多层卷积层的多卷积核的这种组合是一种相对灵活的方式在进行，不同的边缘->不同纹理->不同几何图形和表->不同的狗脸->不同的物体...，浅层模式的组合可以多种多样，使深层可以描述的模式也可以多种多样，所以具有很强的表达能力，不是“死板”的模板，而是“灵活”的模板，泛化能力更强。

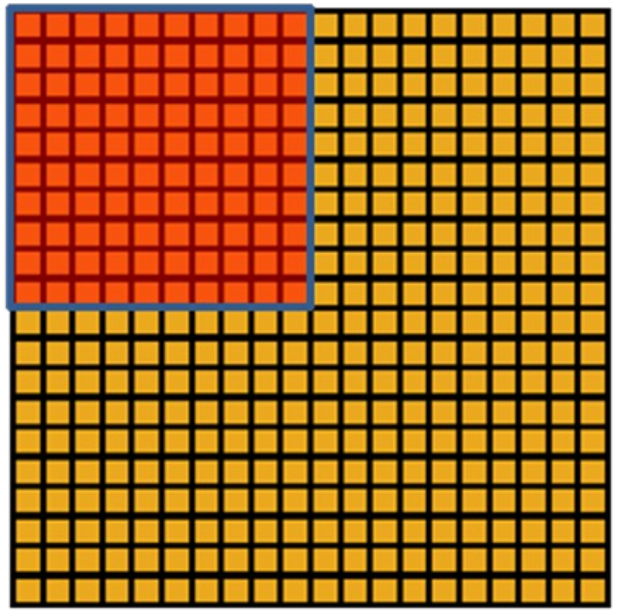
卷积神经网络　一般结构框架：池化层



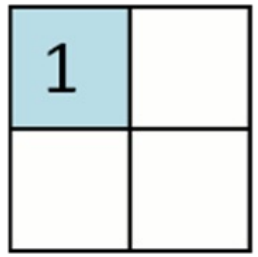
- 使表示更小，更易于管理；- 对每个激活图进行独立操作

卷积神经网络　一般结构框架：池化层

池化层（下采样）——数据降维，避免过拟合



卷积特征



池化特征

卷积神经网络　一般结构框架：全连接层

卷积层

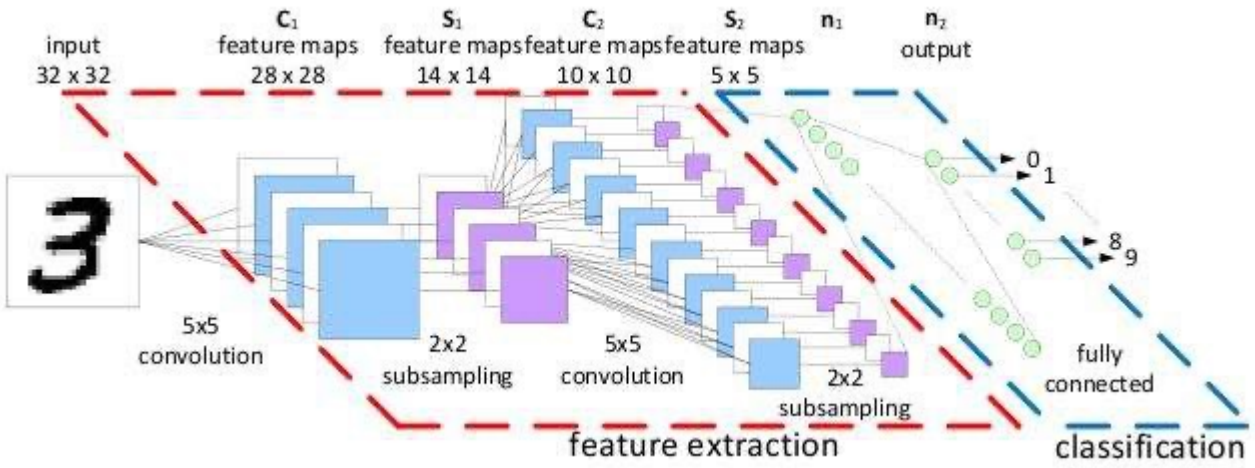
- 提取特征

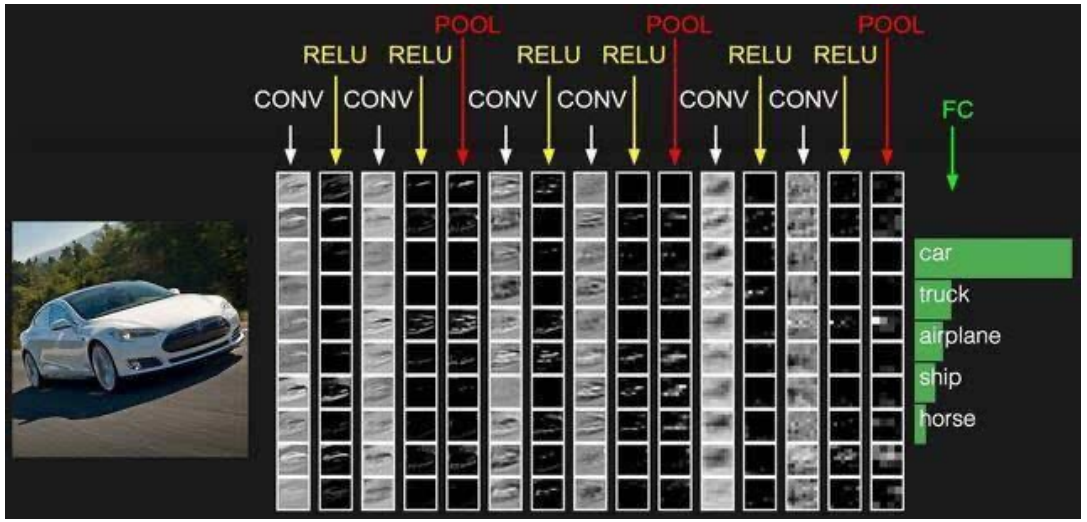
池化层

- 降维、防止过拟合

全连接层

- 输出结果



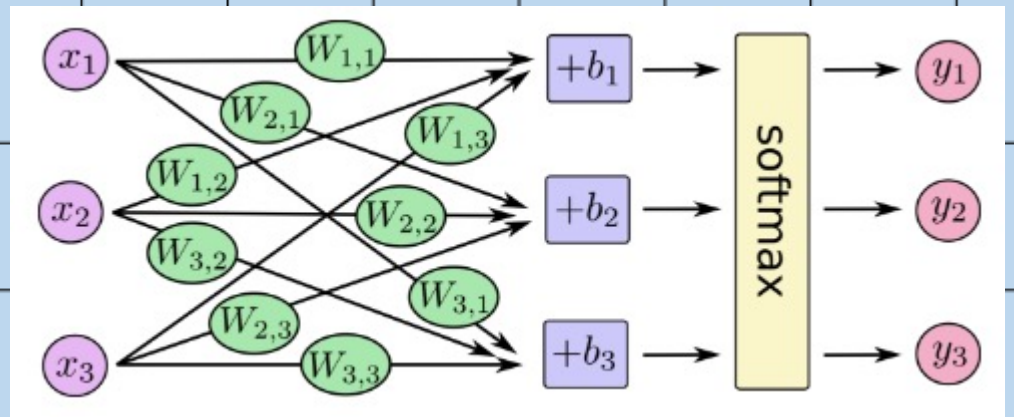


图像特征图的“分布式特征表示”映射到样本标记空间。在整个卷积神经网络中起到“分类器”的作用。

卷积神经网络　一般结构框架：全连接层

隐含层输出的特征图

	猫	狗	花	草	汽车	轮船	…	目标n
神经元1	权重值							
神经元2								
神经元3								
…								
神经元n								权重值

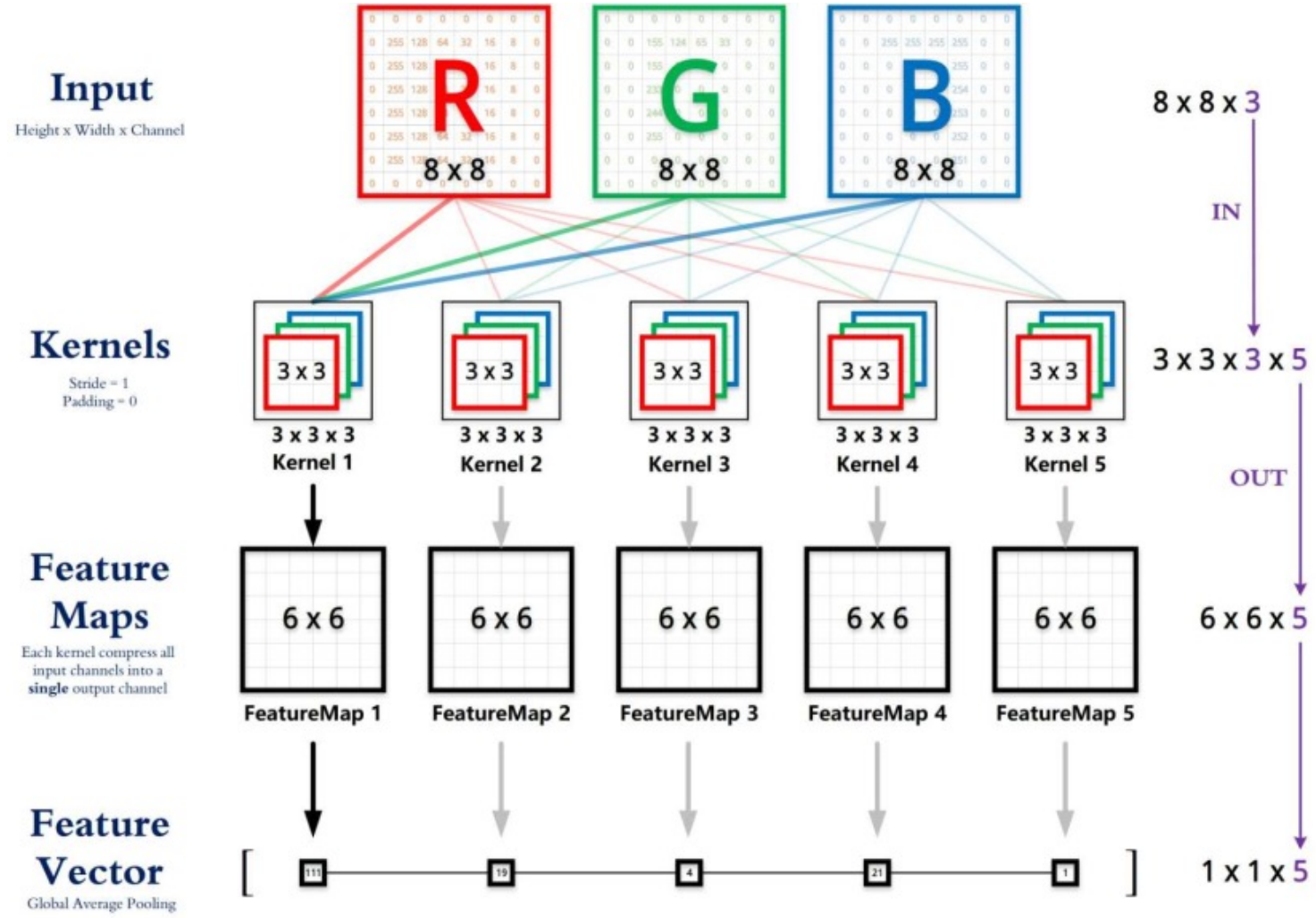


SoftMax

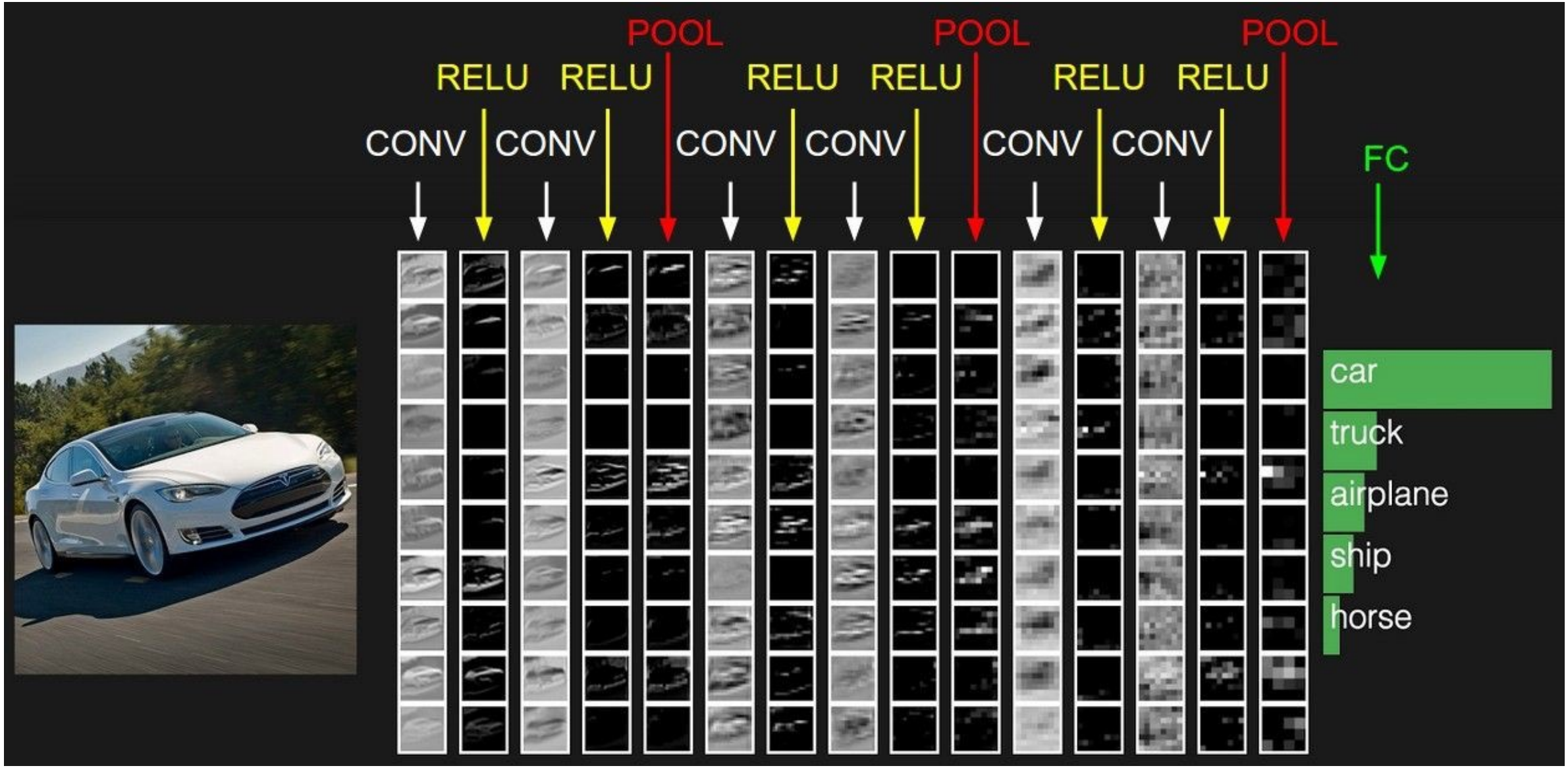
通过Softmax函数将多个标量映射为一个概率分布，输出分类结果的置信度。

卷积神经网络

CNN卷积神经网络识别图像的过程



卷积神经网络


POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

- **以（多维）数组形式出现的信号**
- **局部相关性强的信号**
- **特征可以出现在任何位置的信号**
- **物体平移和变形不变的信号**
- **一维卷积网络：循序信号，文本**
  - 文本、音乐、音频、演讲、时间序列
- **二维卷积网络：图像，时频表示（语音和音频）**
  - 目标检测、定位、识别
- **三维卷积网络：视频，体积图像，断层扫描图像**
  - 视频识别/理解
  - 生物医学图像分析
  - 高光谱图像分析

CNN擅长处理什么数据?