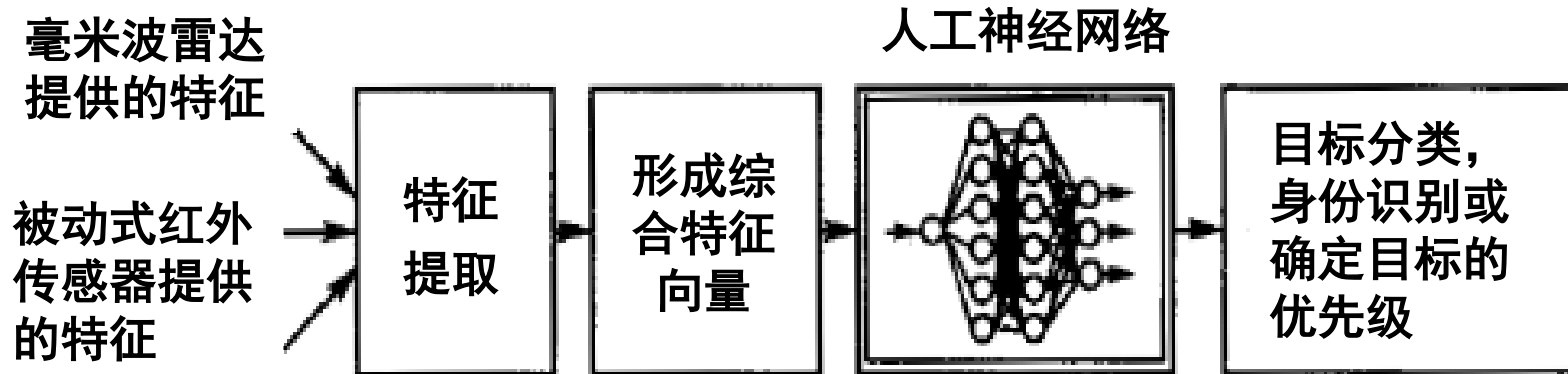


-自动化学院学科核心课-

# 检测技术与自动化

## 第6章 现代检测技术-4-多传感器信息融合





- 根据网络类型，使用监督或无监督方法离线训练网络
- 输出节点把观测到的物体归入某一类

在人工神经网络分类器中的特征级数据融合



## 任务需求：

脱机识别优、良、中、差四个手写汉字

**预处理：**样本图像的归一化、平滑去噪、字符切分、二值化、汉字细化处理

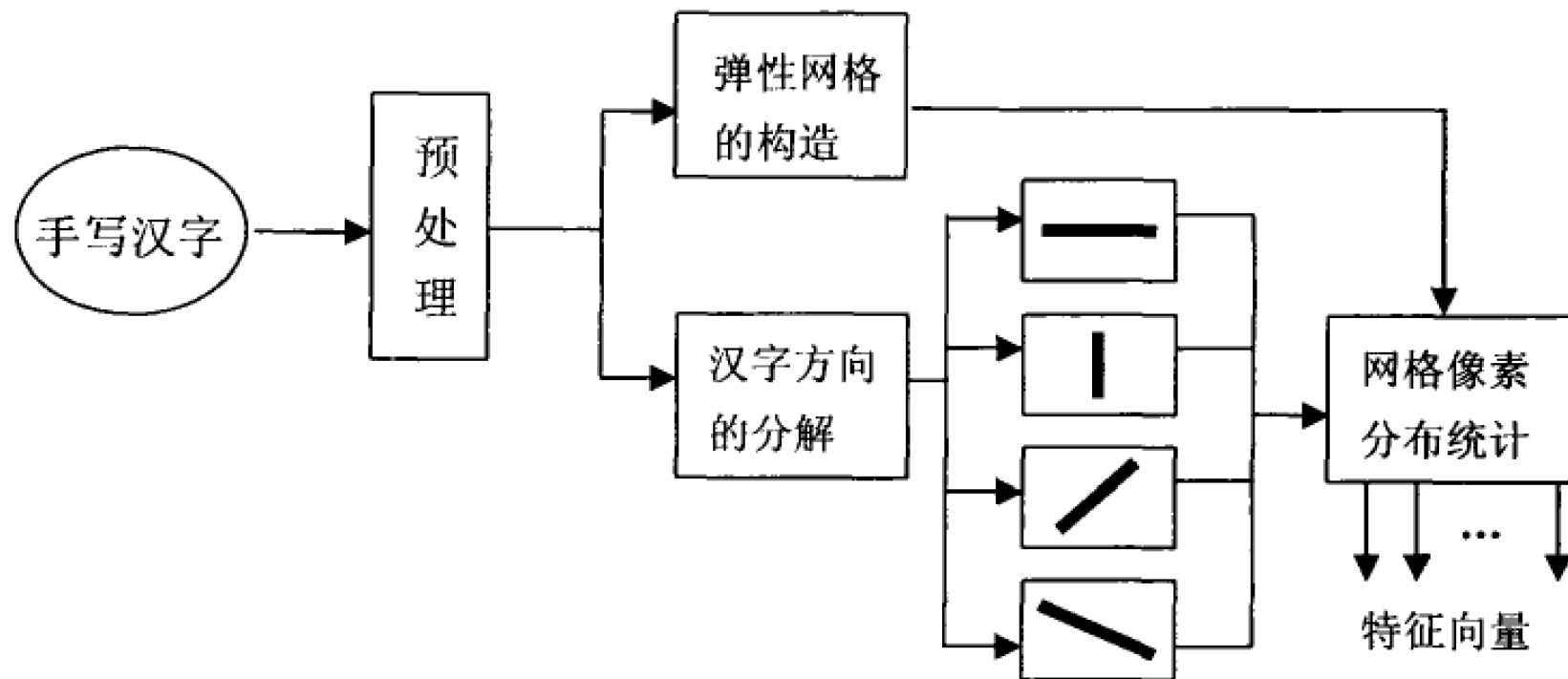
**特征提取：**非均匀网格划分+汉字笔划分解

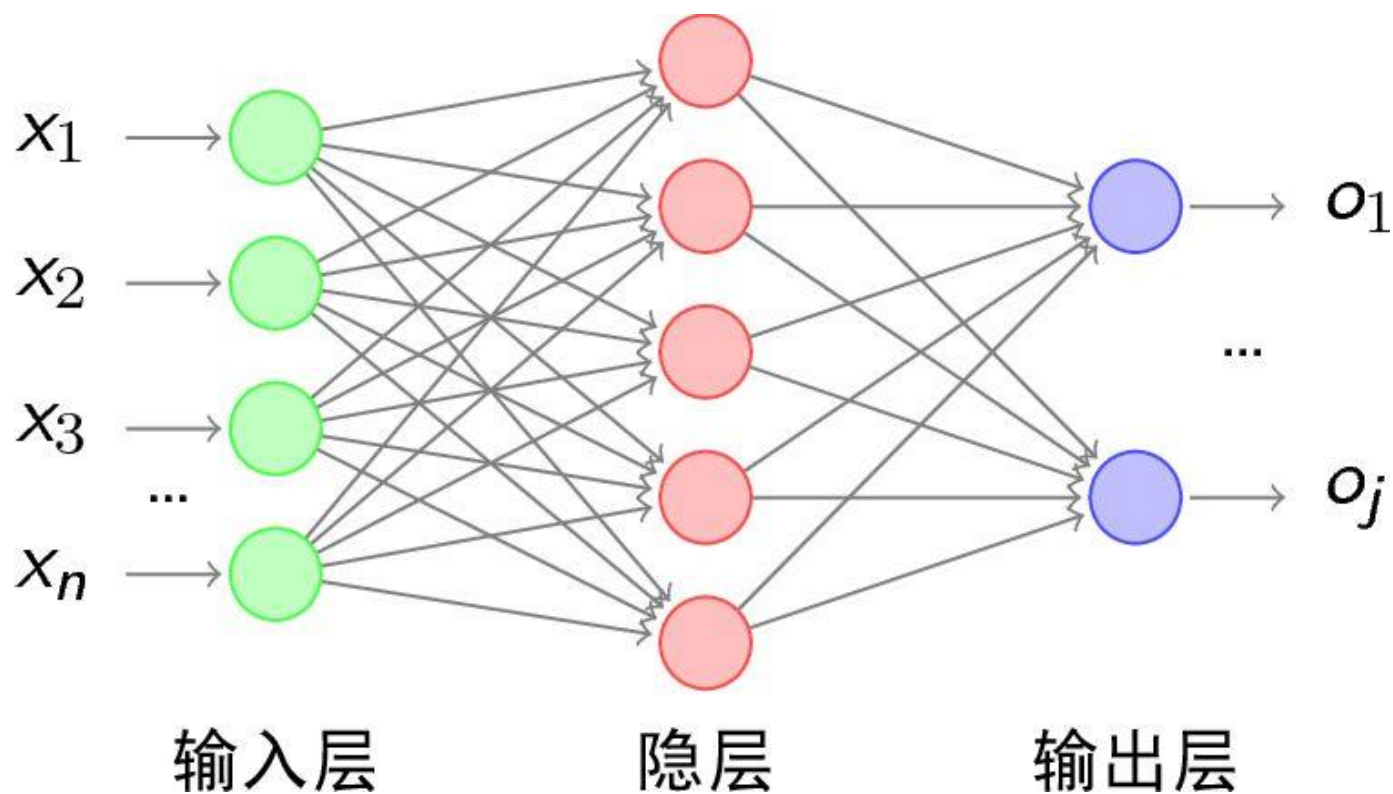
优 优 优 优 优 优 优 优 良 良 良 良 良 良 良 良  
优 优 优 优 优 优 优 优 良 良 良 良 良 良 良 良  
中 中 中 中 中 中 中 中 差 差 差 差 差 差 差 差  
中 中 中 中 中 中 中 中 差 差 差 差 差 差 差 差



4×4 弹性网格划分

特征向量:  $4 \times 4 \times 4$   BP神经网络输入





## 3层BP神经网络



## 阅读思考方向：

1. 论文汉字识别原理和步骤？
2. BP神经网络作用？
3. 如何确定BP神经网络结构：输入、输出、隐层？
4. 如何进一步简化神经网络输入（特征提取）？
5. 神经网络初始参数设置？
6. 神经网络的训练数据如何获取？
7. 神经网络改进思路？

.....



## 思考：

特征提取因人而异，依赖于人的经验。

明确特征，之后送到神经网络里去训练，最后保存模型，再来预测分类的准确性。

特征数目过少→欠拟合

特征数目过多→过拟合





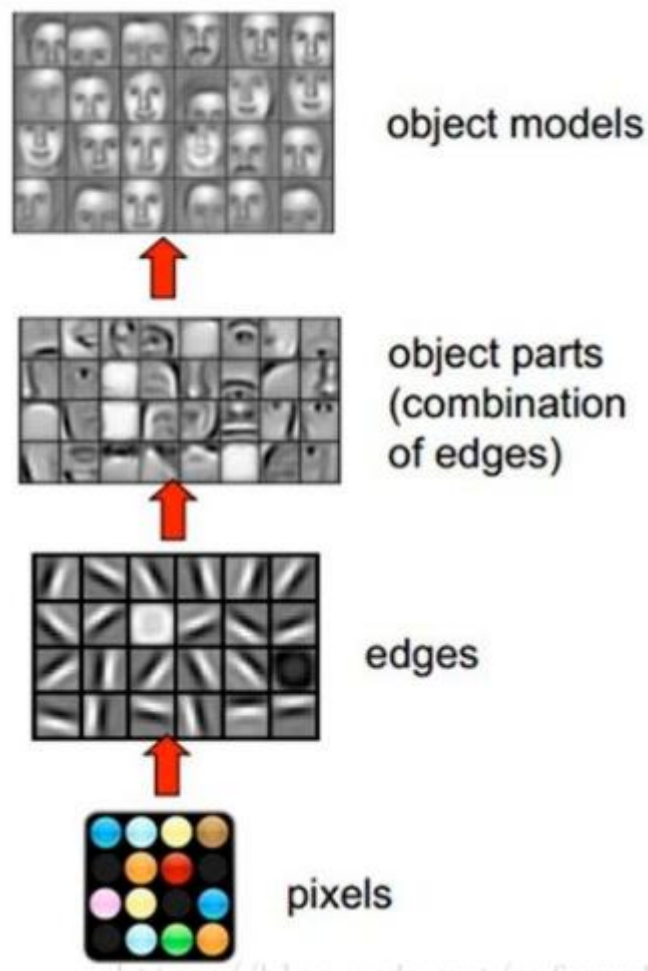
## 知识拓展：

如何实现特征自动提取？ ➔ 卷积神经网络

CNN由纽约大学Yann Lecun于1998年提出，本质是一个多层感知器。

包含卷积计算且具有深度结构的前馈神经网络（Feedforward Neural Networks），是深度学习（deep learning）的代表算法之一。





卷积神经网络（Convolutional Neural Network）模仿人类识别图像的多层过程：

- ①瞳孔摄入像素；
- ②大脑皮层某些细胞初步处理，发现形状边缘、方向；
- ③抽象判定形状（如圆形、方形）；
- ④进一步抽象判定（如判断物体是气球）。



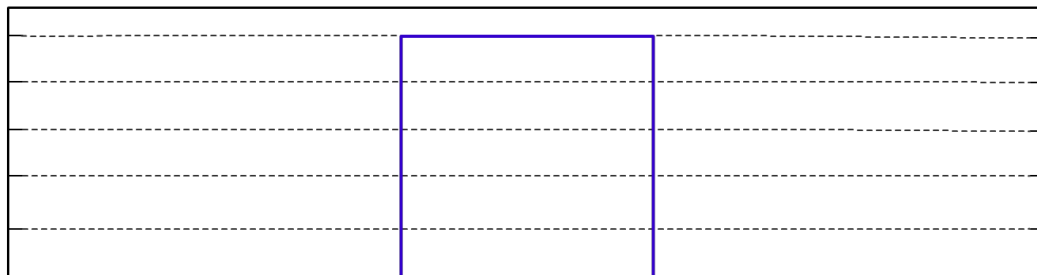
# 卷积积分

特征映射

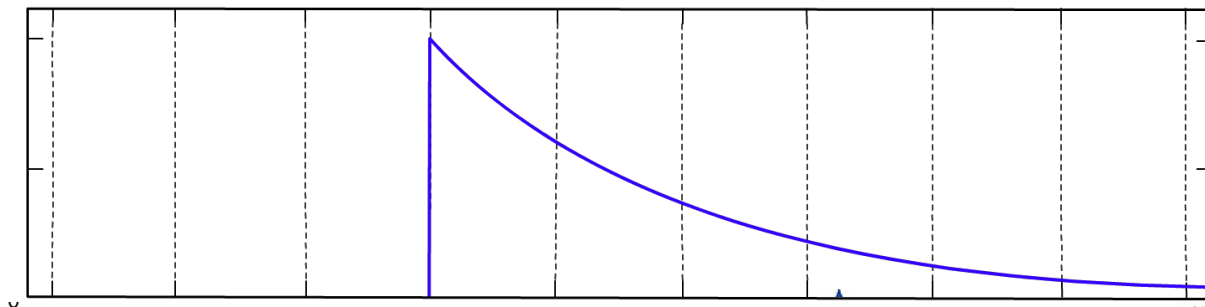
输入

核

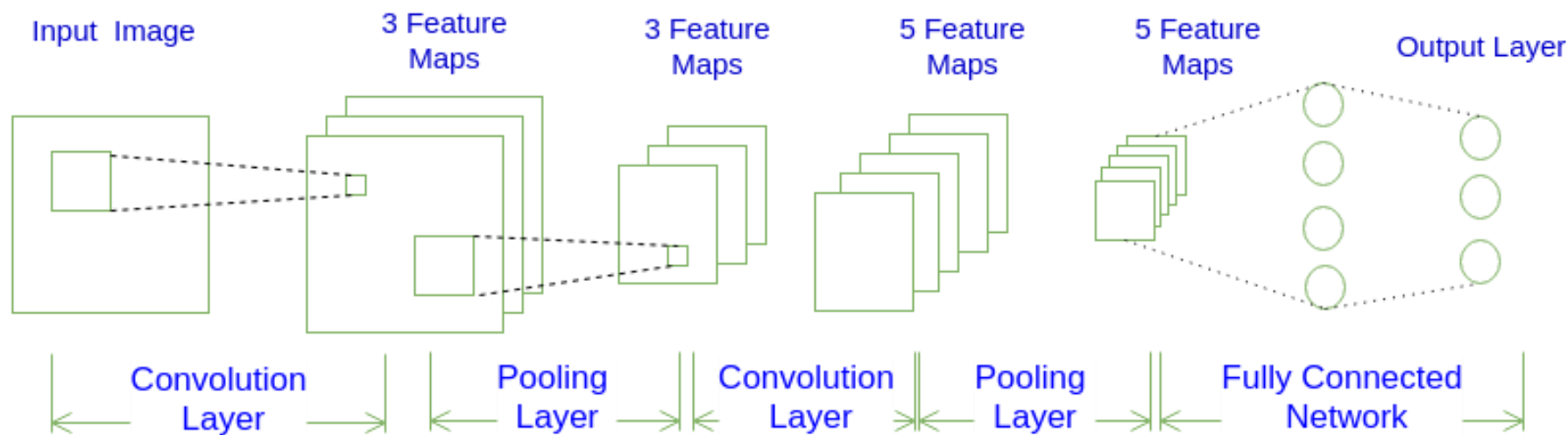
$$f_1(t) * f_2(t) = \int_{-\infty}^{\infty} f_1(\xi) f_2(t - \xi) d\xi$$



—  $f_1(\xi)$   
—  $f_2(t - \xi)$   
—  $f_1(t) * f_2(t)$



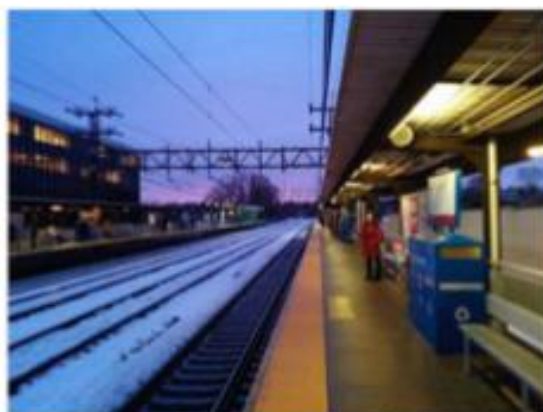
# 卷积神经网络 (Convolutional Neural Networks, ConvNets or CNNs) 是一种在图像识别与分类领域被证明特别有效的神经网络。



# 卷积网络已经成功地识别人脸、物体、交通标志，应用在机器人和无人车等载具。



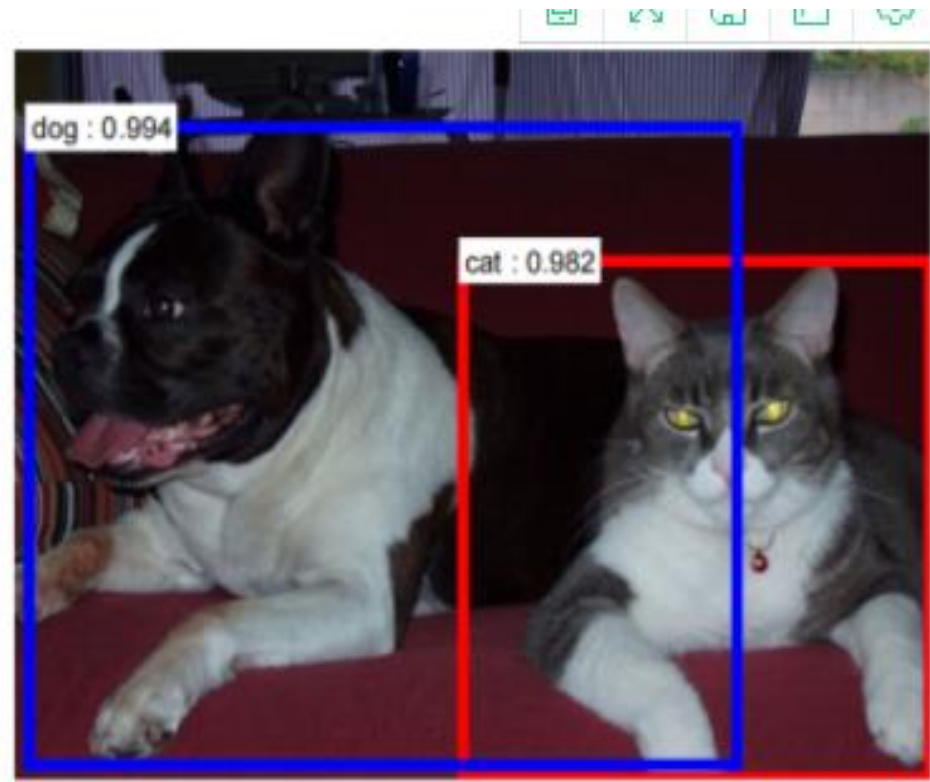
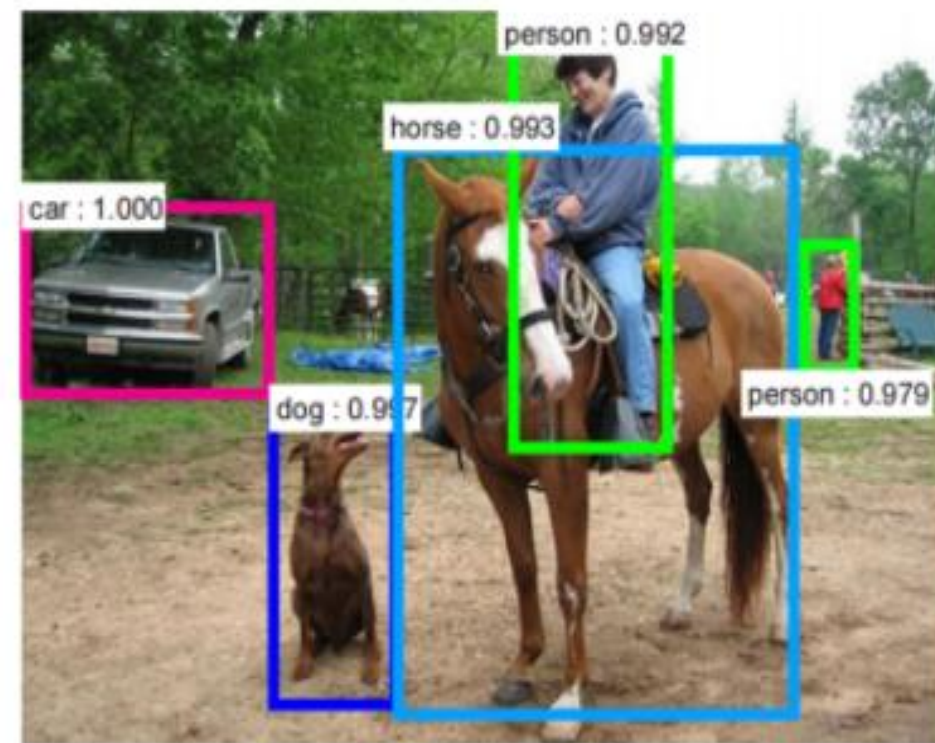
night bridge city suspension bridge river



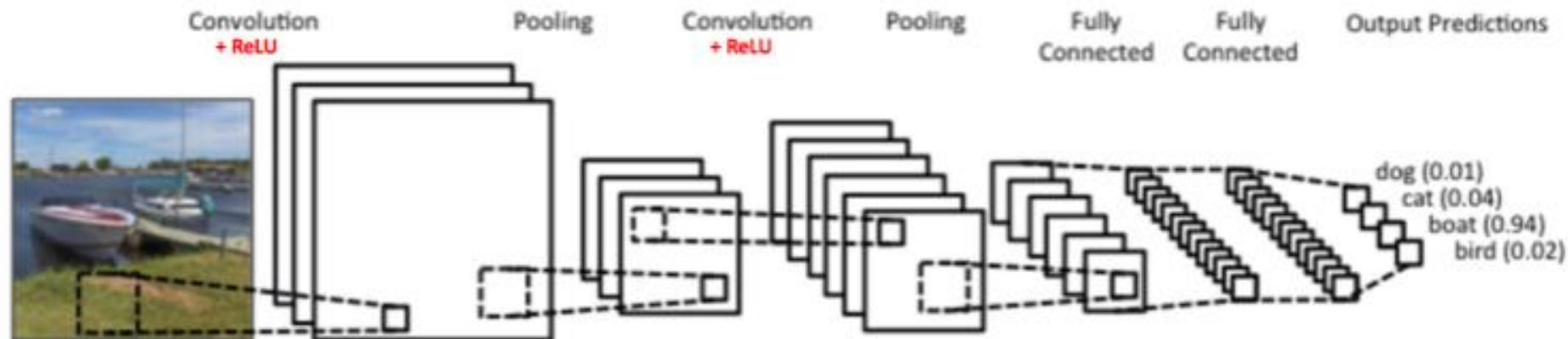
train subway railroad railway station transportation



competition tennis athlete stadium ball many spectators

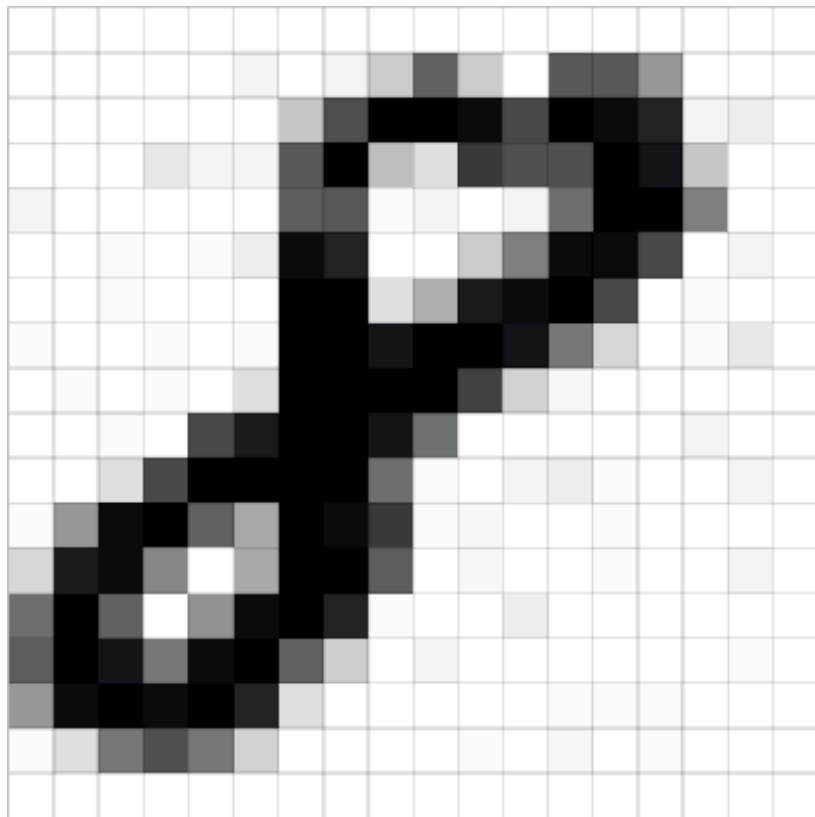






## 卷积神经网络结构

1. 卷积 (Convolution)
2. 非线性激活(ReLU)
3. 池化或下采样(Pooling)
4. 分类 (全连接层, Fully Connected)



每张图片都可以表示为像素值组成的矩阵





**通道：**常见相机拍出来的照片有三个通道——红、绿、蓝-可以想象为是三个2d矩阵（每种颜色对应一个）叠在一起，每个矩阵的值都在0-255之间。

**灰度**图像只有单通道。为简单起见先只考虑灰度图像，即一个2d矩阵。矩阵中的每个像素值还是0到255。



# 1. 卷积（Convolution）

卷积神经网络是因为“**卷积**”**操作**而得名的。卷积的根本目的是从输入图片中提取特征。

卷积用一个小方阵的数据学习图像特征，可以保留像素之间的空间关系。

一个 $5 \times 5$ 的图像，其像素值为0和1，下面的绿色矩阵是灰度图的特例（常规灰度图的像素值取值0-255），同时考虑如下的 $3 \times 3$ 矩阵：

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

1	0	1
0	1	0
1	0	1

卷积核/  
滤波器

**5×5图像和3×3矩阵之间的卷积计算，  
可由下图的动画所表示：**

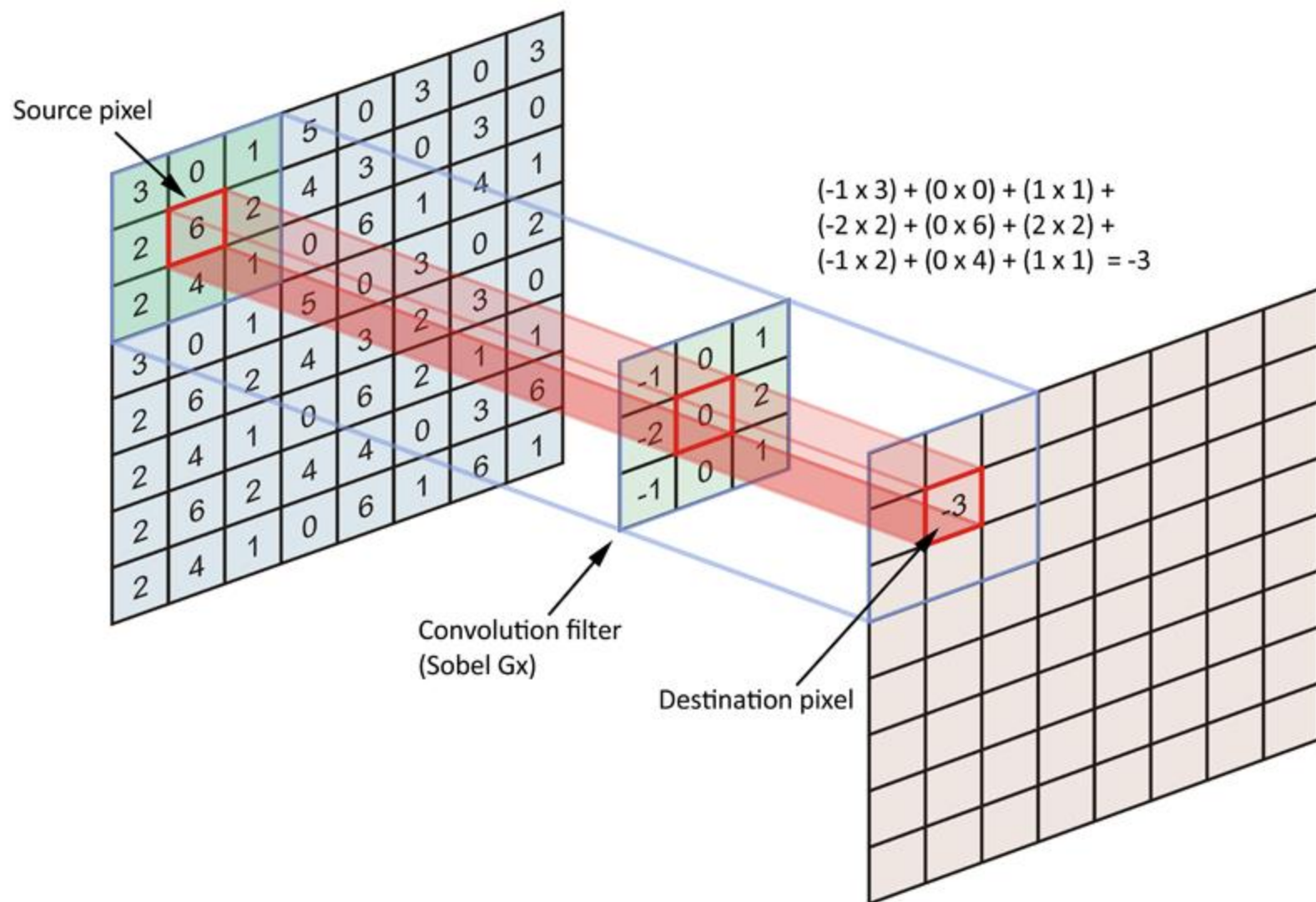
1 <sub>x1</sub>	1 <sub>x0</sub>	1 <sub>x1</sub>	0	0
0 <sub>x0</sub>	1 <sub>x1</sub>	1 <sub>x0</sub>	1	0
0 <sub>x1</sub>	0 <sub>x0</sub>	1 <sub>x1</sub>	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		



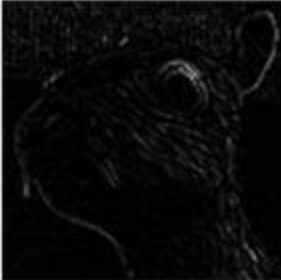
Convolved  
Feature

**得到卷积  
特征/特  
征映射**






## 卷积核/滤波器

## 卷积图像

Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	

## 卷积核/滤波器

## 卷积图像

	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
<b>Sharpen</b>	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
<b>Box blur</b> (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

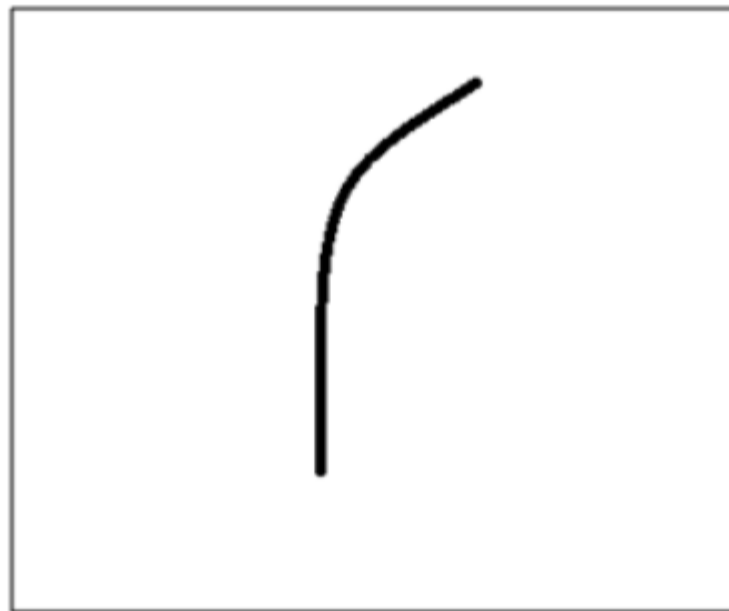


我们要识别图像中的某种特定曲线，即这个滤波器要对这种曲线有很高的输出，对其他形状则输出很低。

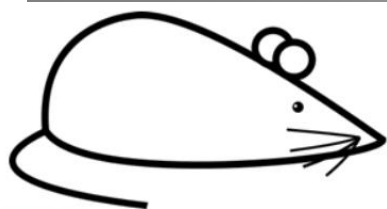
如设计的滤波器和要识别的曲线如下：

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

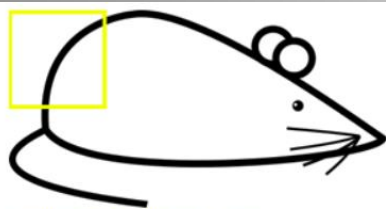
Pixel representation of filter



Visualization of a curve detector filter



Original image



Visualization of the filter on the image

当它移动到上面的位置时，按照矩阵操作，这个区域的图像像素值与滤波器相乘，可以得到一个很大的值：



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!)

而当这个滤波器移动到其他区域时，会得到一个相对很小的值：



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



对整个原图进行一次卷积，得到一张激活图。  
对应的高值区就是我们要检测曲线的位置。

训练卷积神经网络的某一个卷积层时，实际在训练一系列的滤波器。

相当程度上，构建卷积神经网络的任务就在于构建这些滤波器。也就是，将这些滤波器变成（改变滤波器矩阵的值，也就是Weight）--能识别特定的特征。这个过程为训练。



Input



Input Volume (+pad 1) (7x7x3)

$x[:, :, 0]$						
0	0	0	0	0	0	0
0	0	1	1	0	2	0
0	2	2	2	2	1	0
0	1	0	0	2	0	0
0	0	1	1	0	0	0
0	1	2	0	0	2	0
0	0	0	0	0	0	0
$x[:, :, 1]$						
0	0	0	0	0	0	0
0	1	0	2	2	0	0
0	0	0	0	2	0	0
0	1	2	1	2	1	0
0	1	0	0	0	0	0
0	1	2	1	1	1	0
0	0	0	0	0	0	0
$x[:, :, 2]$						
0	0	0	0	0	0	0
0	2	1	2	0	0	0
0	1	0	0	1	0	0
0	0	2	1	0	1	0
0	0	1	2	2	2	0
0	2	1	0	0	1	0
0	0	0	0	0	0	0

Filter W0 (3x3x3)

$w0[:, :, 0]$		
-1	1	0
0	1	0
0	1	1
$w0[:, :, 1]$		
-1	-1	0
0	0	0
0	-1	0
$w0[:, :, 2]$		
0	0	-1
0	1	0
1	-1	-1

Bias b0 (1x1x1)

$b0[:, :, 0]$		
1		

Filter W1 (3x3x3)

$w1[:, :, 0]$		
1	1	-1
-1	-1	1
0	-1	1
$w1[:, :, 1]$		
0	1	0
-1	0	-1
-1	1	0
$w1[:, :, 2]$		
-1	0	0
-1	0	1
-1	0	0

Bias b1 (1x1x1)

$b1[:, :, 0]$		
0		

Output Volume (3x3x2)

$o[:, :, 0]$		
6	7	5
3	-1	-1
2	-1	4
$o[:, :, 1]$		
2	-5	-8
1	-4	-4
0	-5	-5

toggle movement

## 三通道图像 特征提取

## 三通道图像（特征提取）

0	0	0	0	0	0	...
0	156	155	156	158	158	...
0	153	154	157	159	159	...
0	149	151	155	158	159	...
0	146	146	149	153	158	...
0	145	143	143	148	158	...
...	...	...	...	...	...	...

Input Channel #1 (Red)

0	0	0	0	0	0	...
0	167	166	167	169	169	...
0	164	165	168	170	170	...
0	160	162	166	169	170	...
0	156	156	159	163	168	...
0	155	153	153	158	168	...
...	...	...	...	...	...	...

Input Channel #2 (Green)

0	0	0	0	0	0	...
0	163	162	163	165	165	...
0	160	161	164	166	166	...
0	156	158	162	165	166	...
0	155	155	158	162	167	...
0	154	152	152	157	167	...
...	...	...	...	...	...	...

Input Channel #3 (Blue)

-1	-1	1
0	1	-1
0	1	1

Kernel Channel #1



308

1	0	0
1	-1	-1
1	0	-1

Kernel Channel #2



-498

0	1	1
0	1	0
1	-1	1

Kernel Channel #3



164

+

+

+ 1 = -25

↑  
Bias = 1

Output

-25				...
				...
				...
				...
...	...	...	...	...

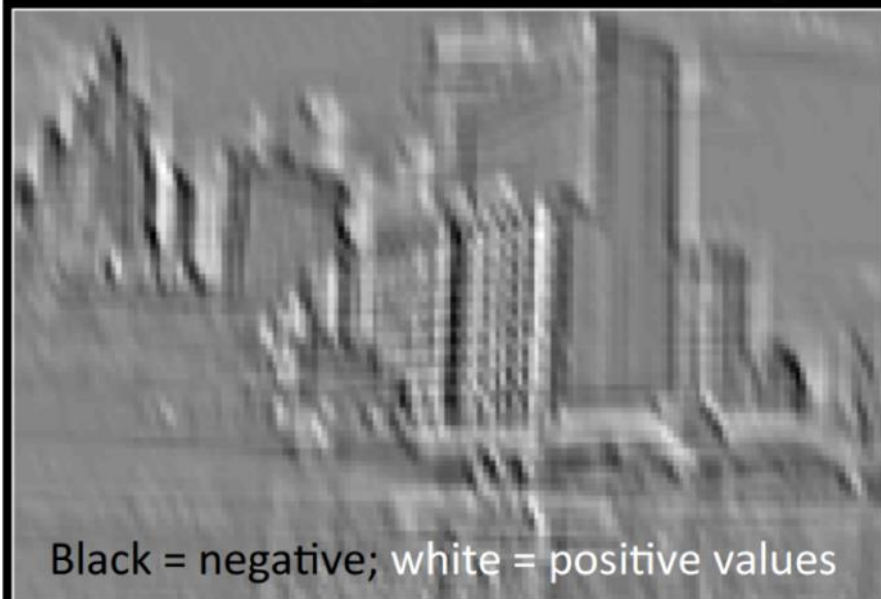
最终得到的卷积结果是原始图像各通道上的综合信息结果。



## 2. 非线性激活(ReLU)

ReLU是以像素为单位生效的，其将所有负值像素替换为0。ReLU的目的是向卷积网络中**引入非线性**，因为真实世界里大多数需要学习的问题都是非线性的（单纯的卷积操作是线性的——矩阵相乘、相加，因此需要额外的计算引入非线性）。

Input Feature Map



ReLU

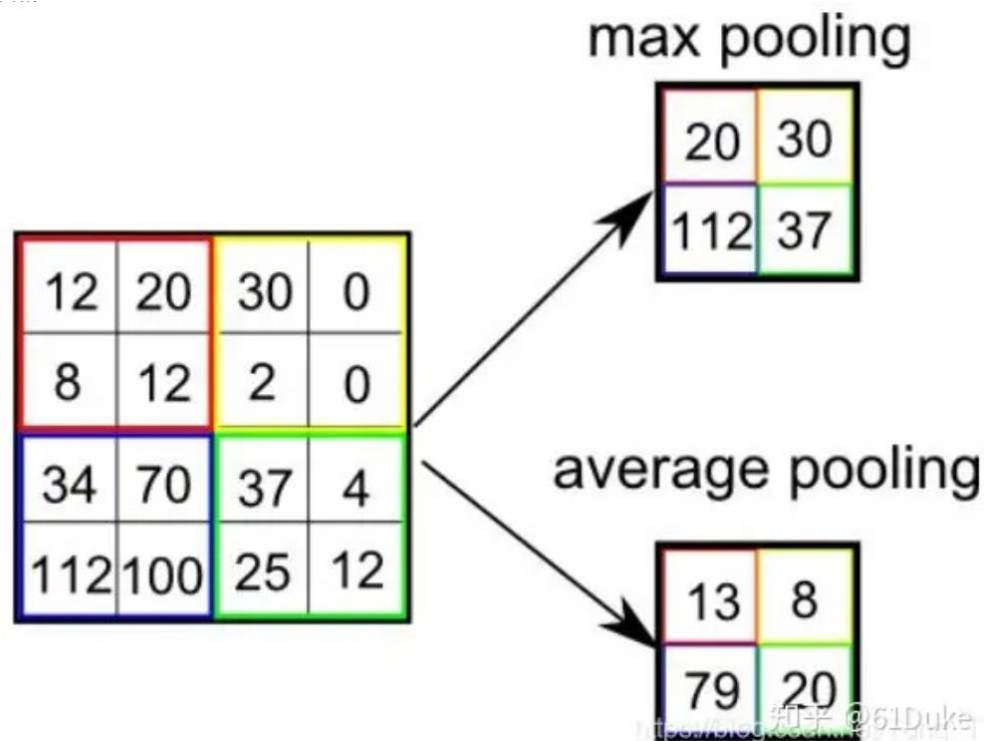


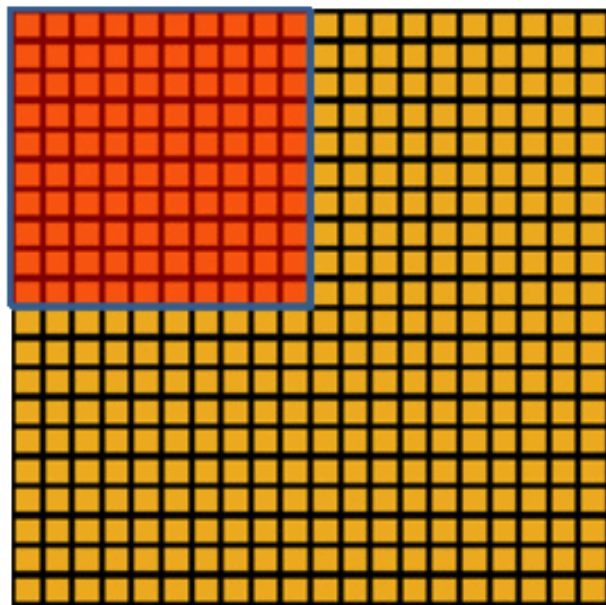
Rectified Feature Map



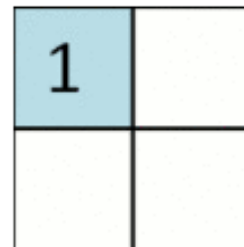
### 3. 池化或下采样(Pooling)

空间池化（也叫亚采样或下采样）降低了每个特征映射的维度，但是保留了最重要的信息。空间池化可以有很多种形式：最大(Max)，平均(Average)，求和(Sum)等等。





卷积特征



池化特征

卷积后接池化层（Pooling），能够很好的聚合特征、降维来减少运算量。



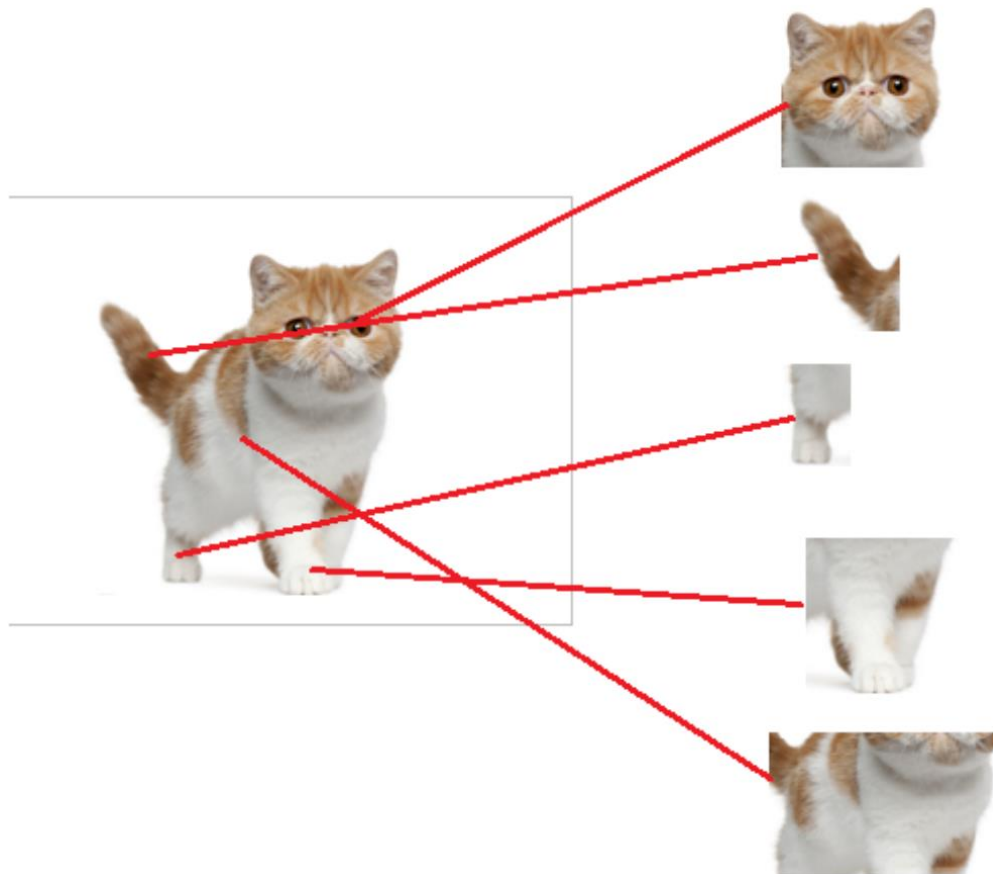
## 4. 分类（全连接层, Fully Connected）

**卷积：**提取的特征是局部特征。“不识庐山真面目，只缘身在此山中”。

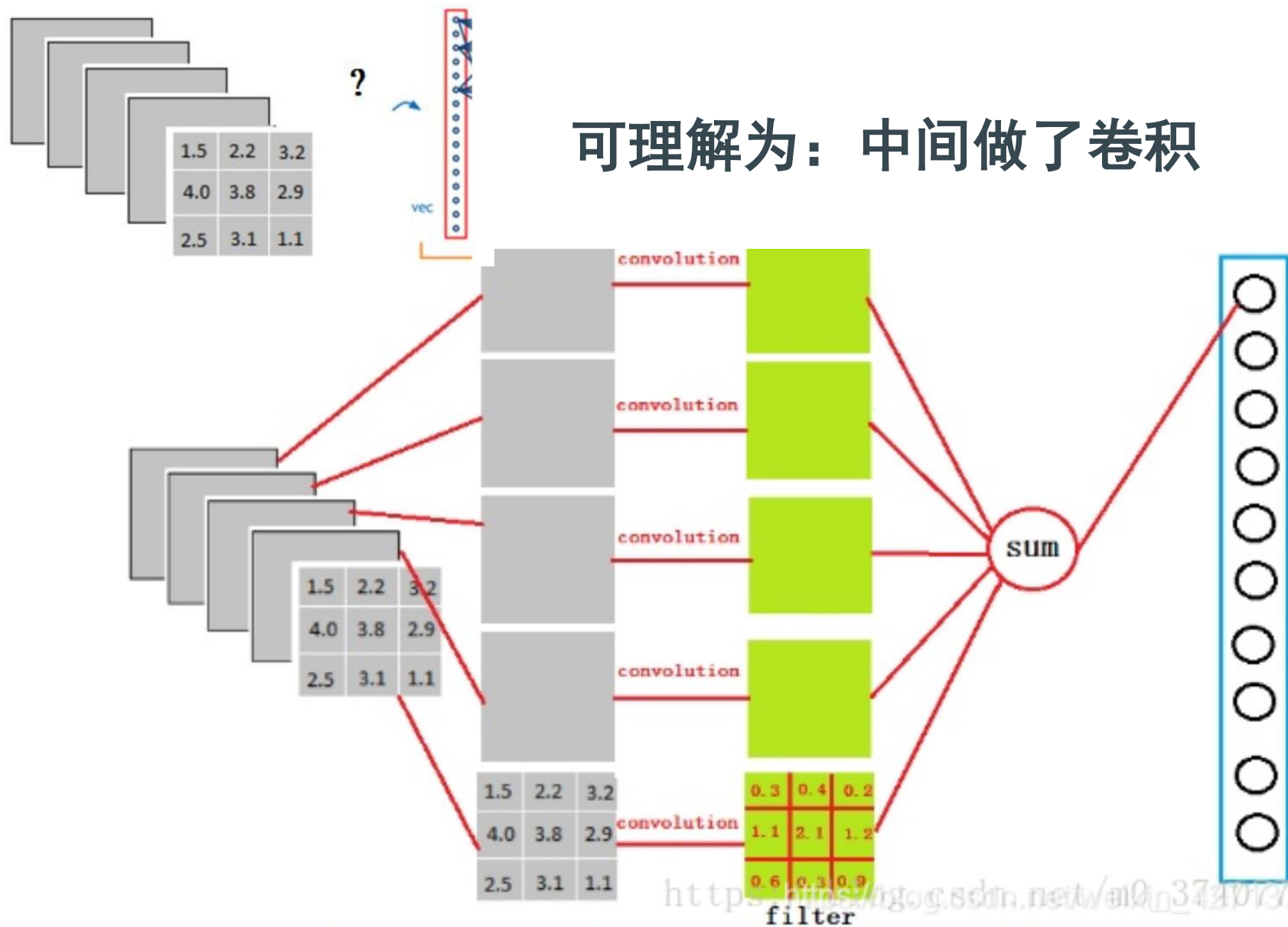
**全连接层：**看到的是全局特征。全连接是“不畏浮云遮望眼，自缘身在最高层”。

卷积+池化→特征提取器

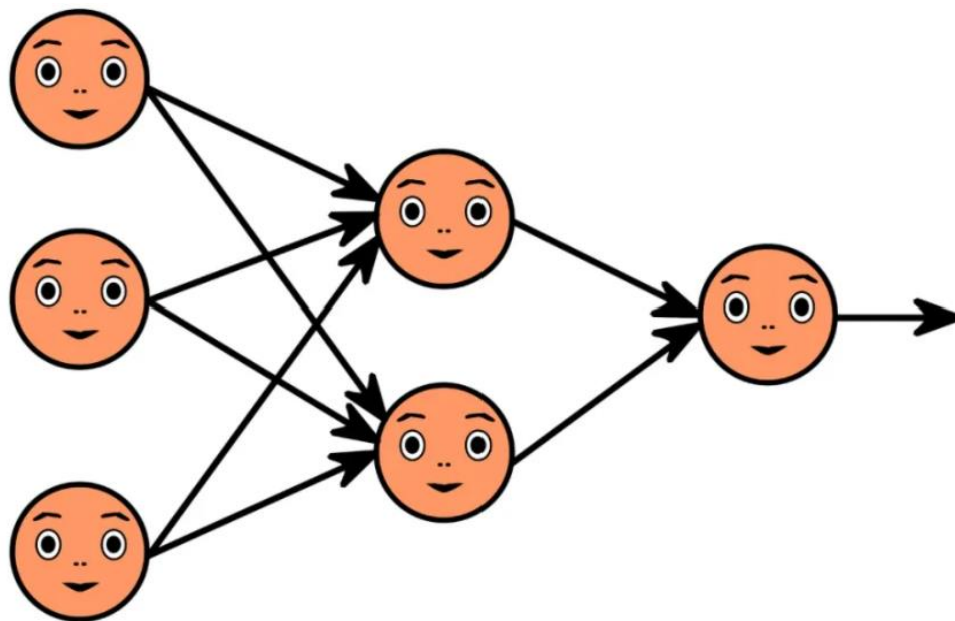
全连接层→分类器



若得到以上特征，就可以判断是猫了



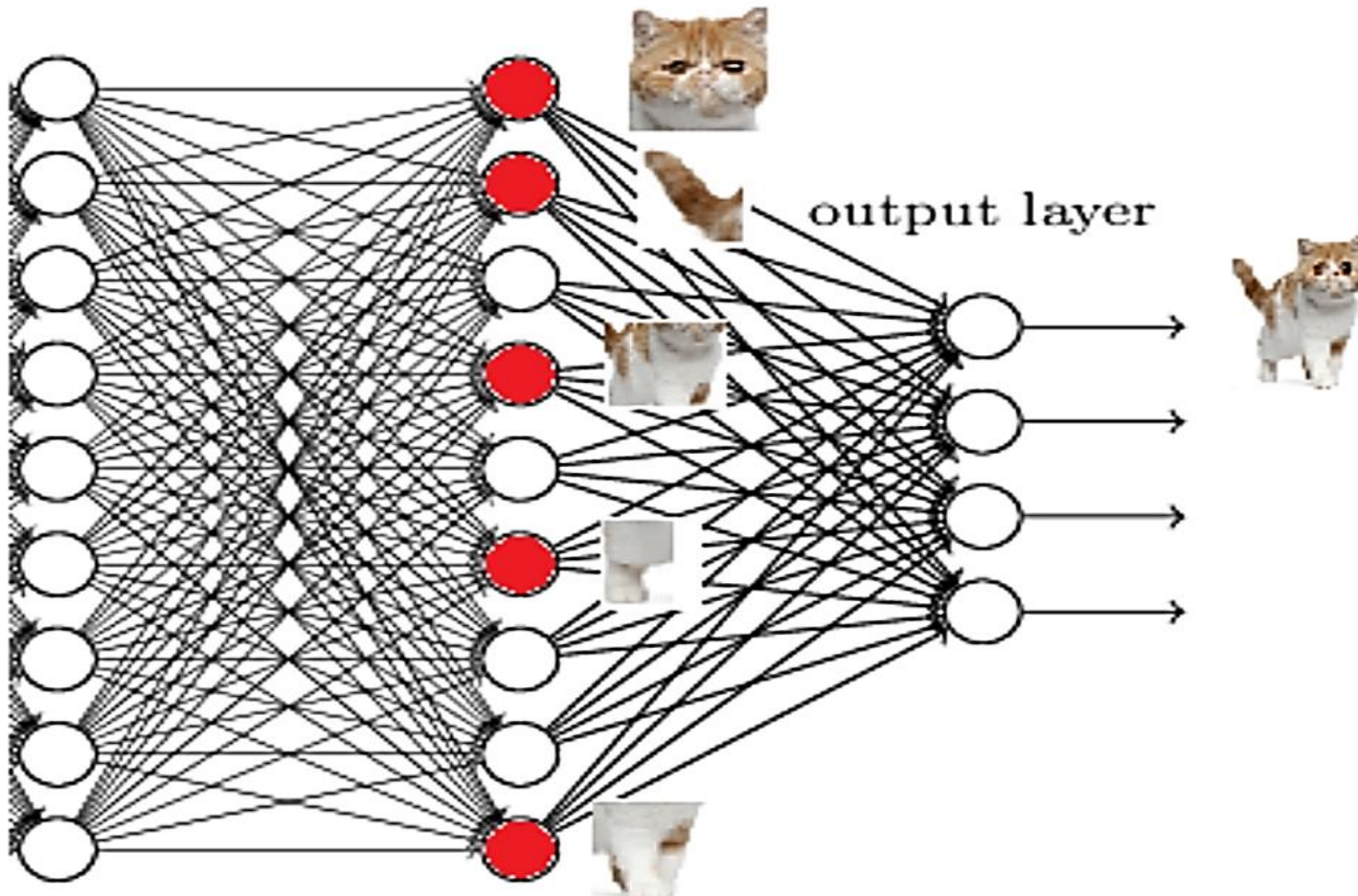




**全连接层 (Fully Connected Layer), 有时也称为密集层 (Dense Layer), 因为这一层的每个神经元都与前一层的每个神经元连接在一起形成一个全连接的网络结构。**

**既然是所有神经元的全部连接, 说明它可以捕获到前面的所有特征, 通过运算可以完备融合所有特征。**

红色的神经元表示这个特征被找到（激活），同一层的其他神经元，要么猫的特征不明显，要么没找到。把找到的特征组合在一起，发现最符合要求的是猫。

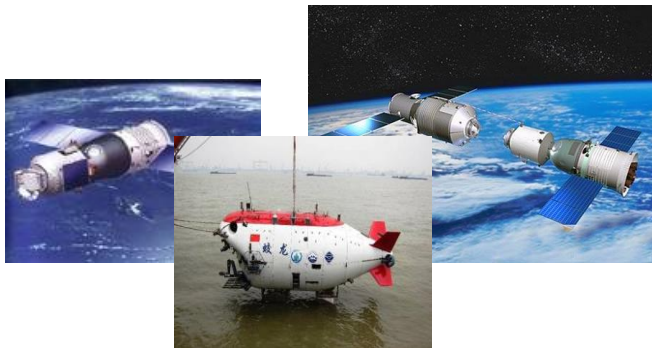


全连接层



## 思考：

- 如何利用卷积神经网络实现汉字识别？
- 如何利用卷积神经网络实现多传感器信息融合？



# Thanks

