



温州大學  
WENZHOU UNIVERSITY

# 深度学习-第七章-卷积神经网络

黄海广 副教授

2021年04月

# 本章目录

2

- 01** 计算机视觉概述
- 02** 卷积神经网络简介
- 03** 卷积神经网络计算
- 04** 卷积神经网络案例

## 01 计算机视觉概述

## 02 卷积神经网络概述

## 03 卷积神经网络计算

## 04 卷积神经网络案例

# 计算机视觉

4

- 图像分类
- 目标检测
- 图像分割
- 目标跟踪
- OCR文字识别
- 图像滤波与降噪
- 图像增强
- 风格迁移
- 三维重建
- 图像检索
- GAN



## 图像获取

提取二维图像、三维图组、图像序列或相关的物理数据，如声波、电磁波或核磁共振的深度、吸收度或反射度



## 预处理

对图像做一种或一些预处理，使图像满足后继处理的要求，如：二次取样保证图像坐标的正确，平滑、去噪等



## 特征提取

从图像中提取各种复杂度的特征，如：线，边缘提取和脊侦测，边角检测、斑点检测等局部化的特征点检测



## 检测/分割

对图像进行分割，提取有价值的内容，用于后继处理，如：筛选特征点，分割含有特定目标的部分



## 高级处理

验证得到的数据是否匹配前提要求，估测特定系数，对目标进行分类

# 图像分类

5



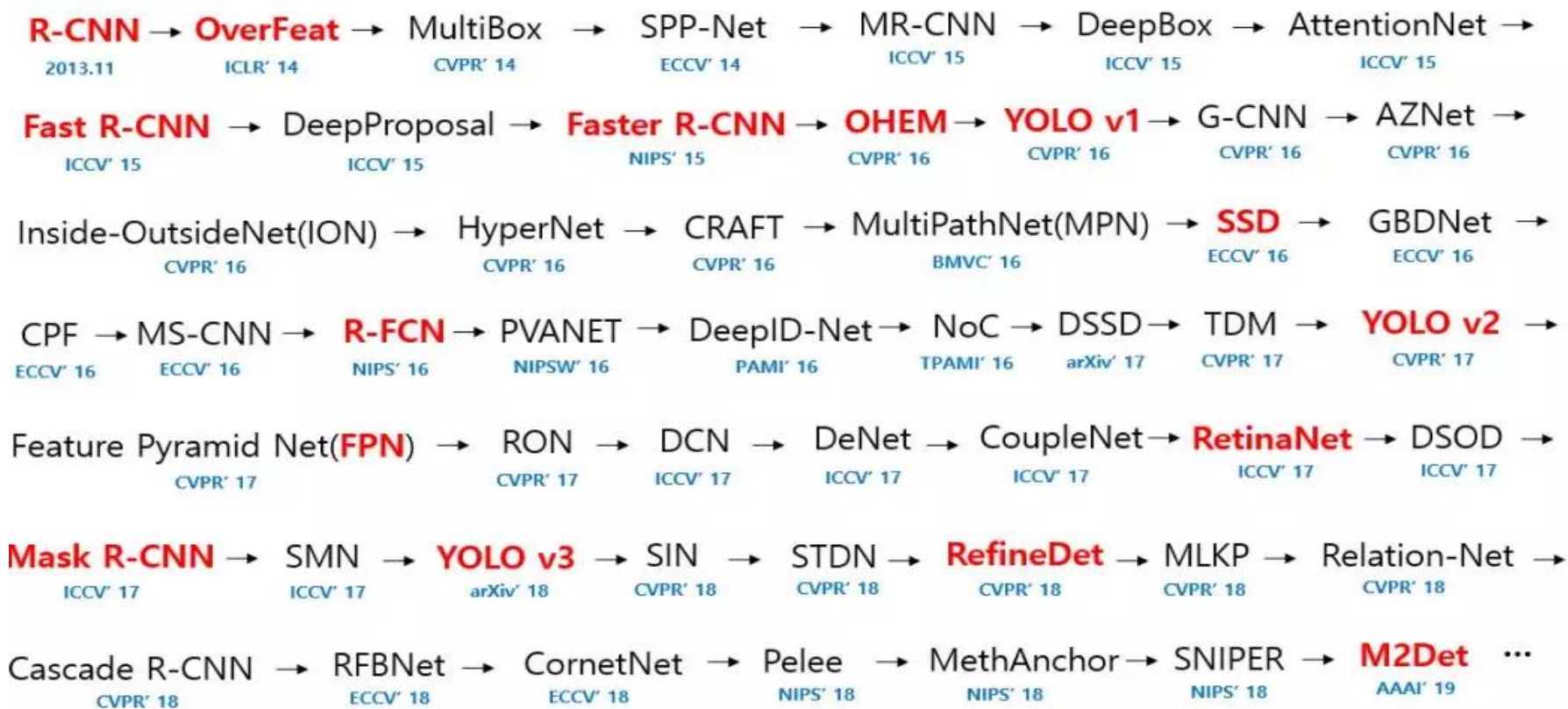


# 目标检测

6

**one-stage**(YOLO,YOLO9000,YOLOV3,YOLOV4, YOLOV5,SSD等)

**two-stage**(OverFeat, R-CNN, Fast R-CNN, Faster R-CNN 等)



目标检测结合了目标分类和定位两个任务。

# 目标检测

7

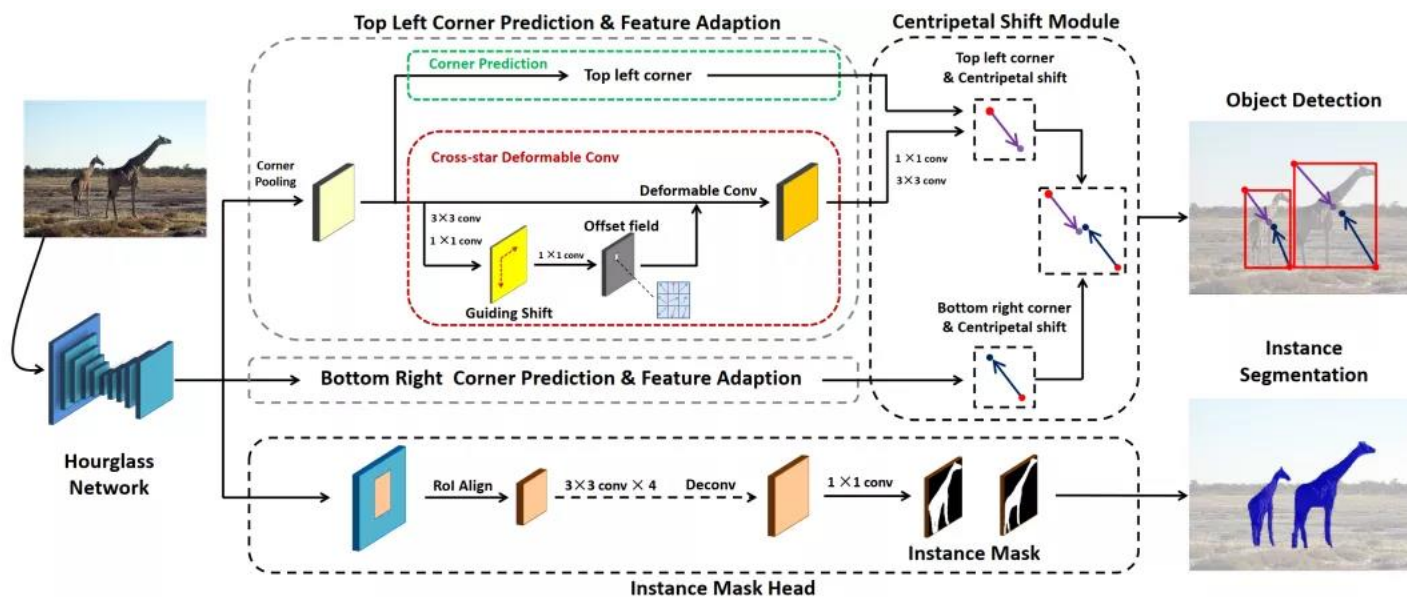


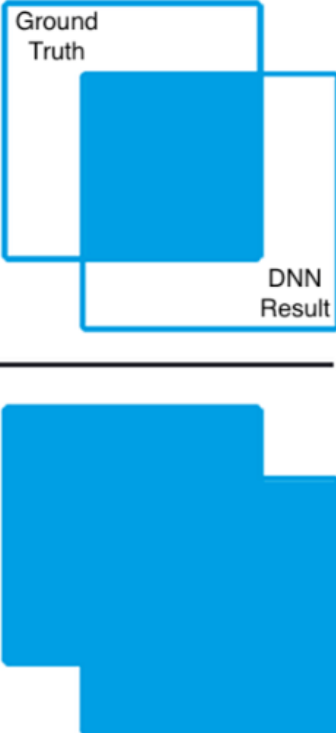
Figure 2. An overview of CentripetalNet. As the corner prediction and feature adaption of top-left corner and bottom-right corner are similar, we only draw top-left corner module for simplicity. Centripetal shift module gets predicted corners and adapted features, then it predicts the centripetal shift of each corner and performs corner matching based on the predicted corners and centripetal shifts. During matching, if the positions of the shifted corners are close enough, they form a bounding box with a high score.

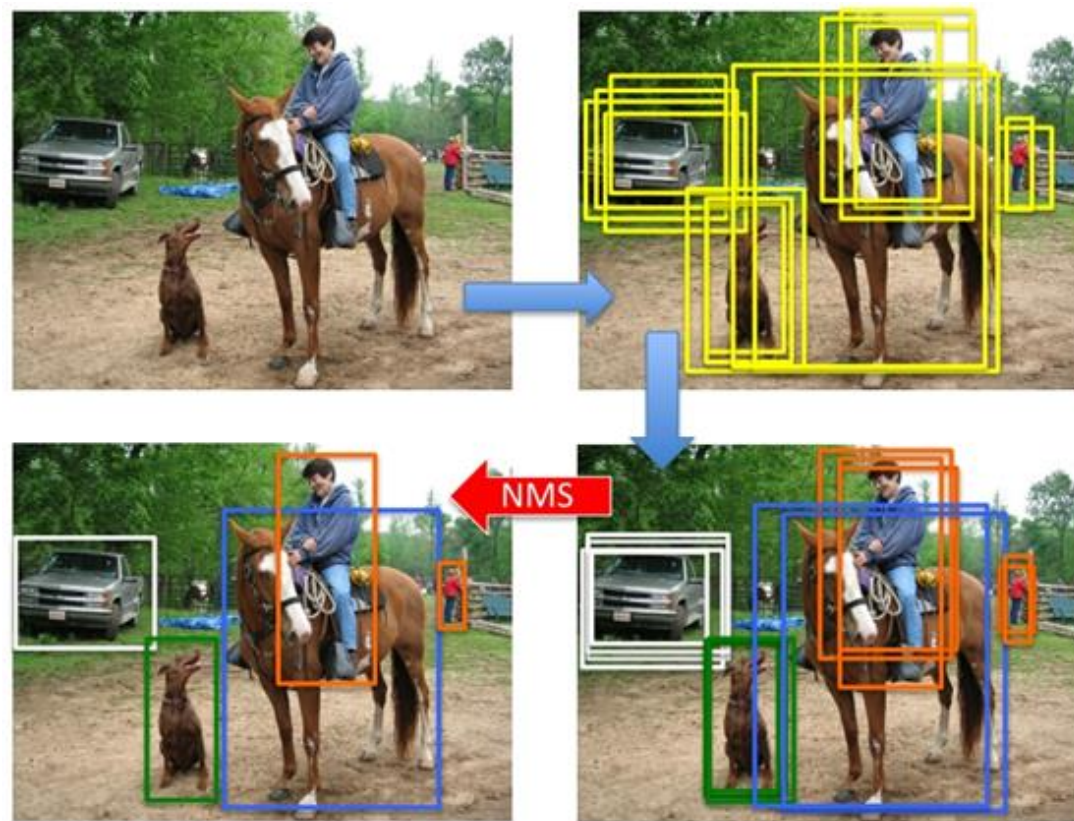


Figure 5: Illustration of the detection results on the target domain. First and second rows: Cityscapes  $\rightarrow$  Foggy-Cityscapes. Third and fourth rows: Sim10K  $\rightarrow$  Cityscapes. Best view in color.

# 目标检测

8

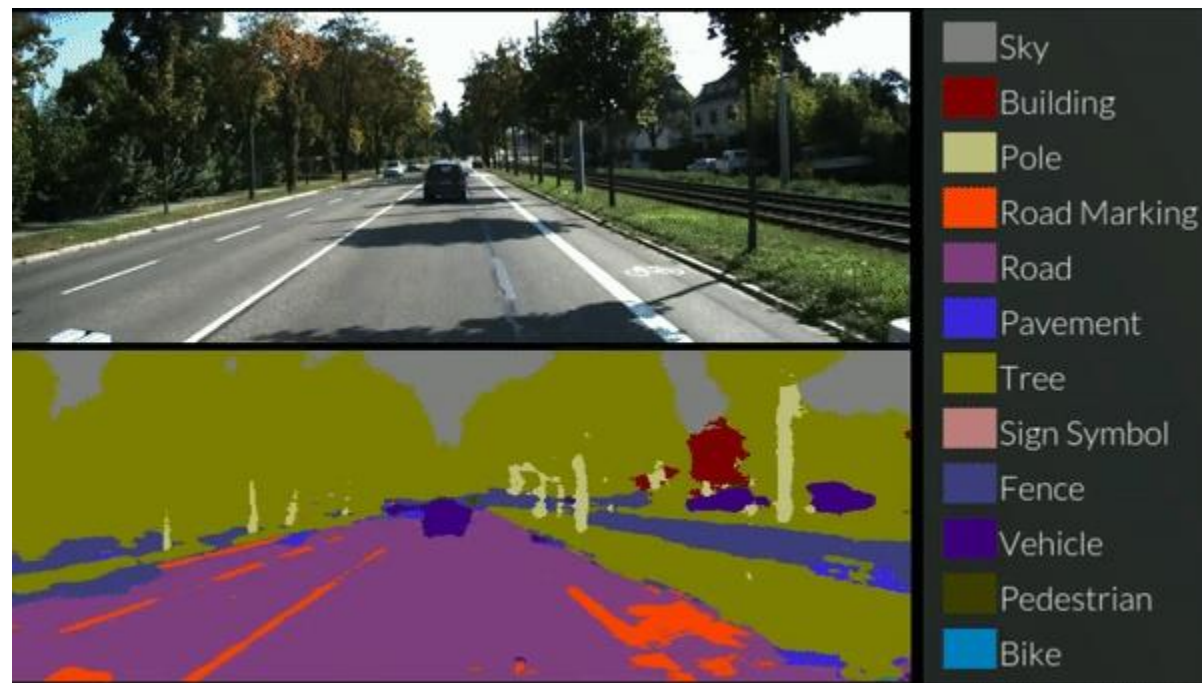
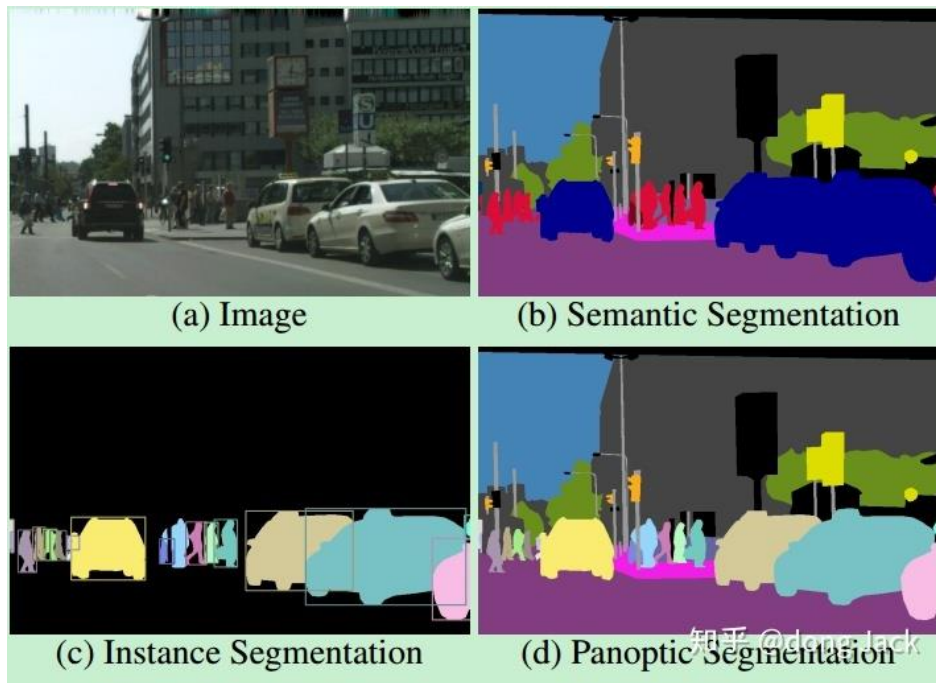
$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$






# 图像分割

9



# 目标跟踪

10

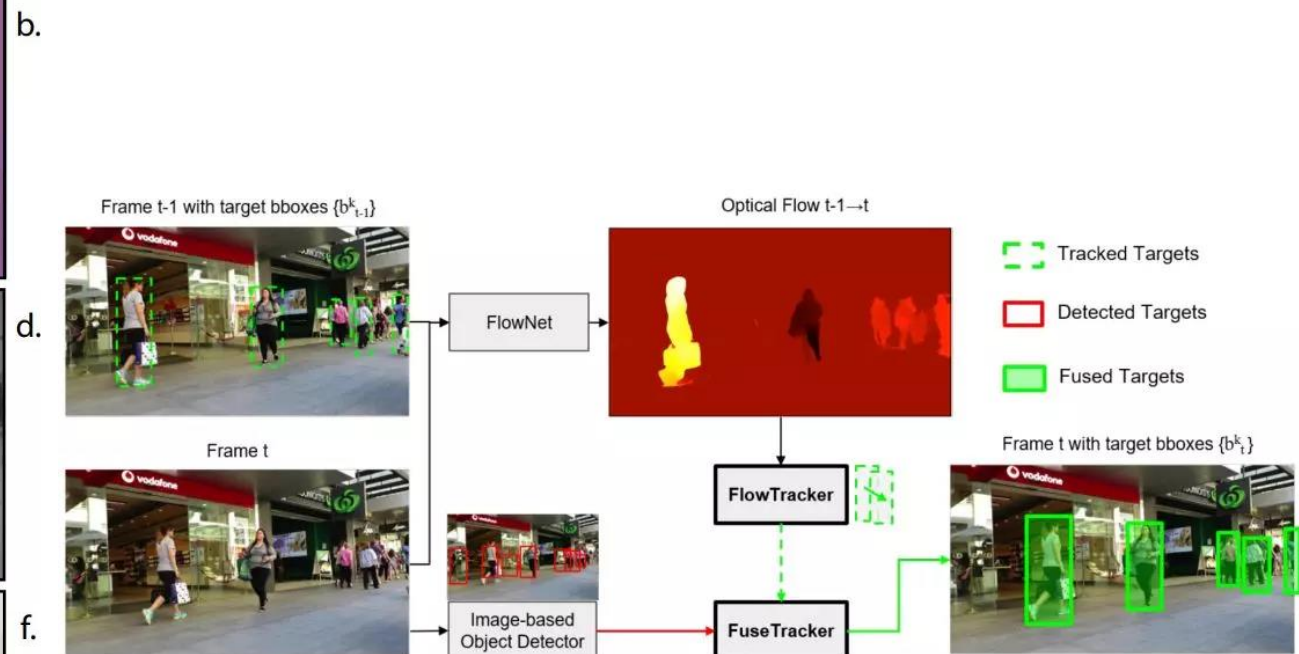
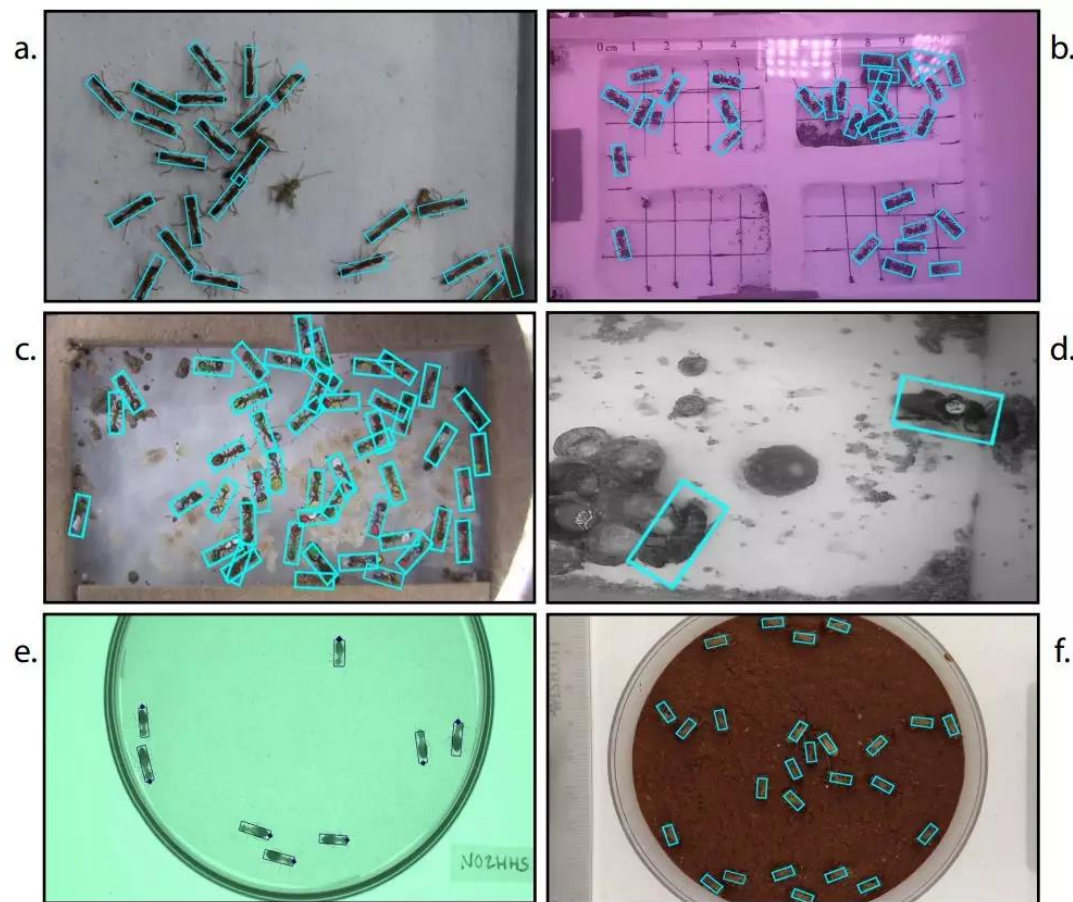
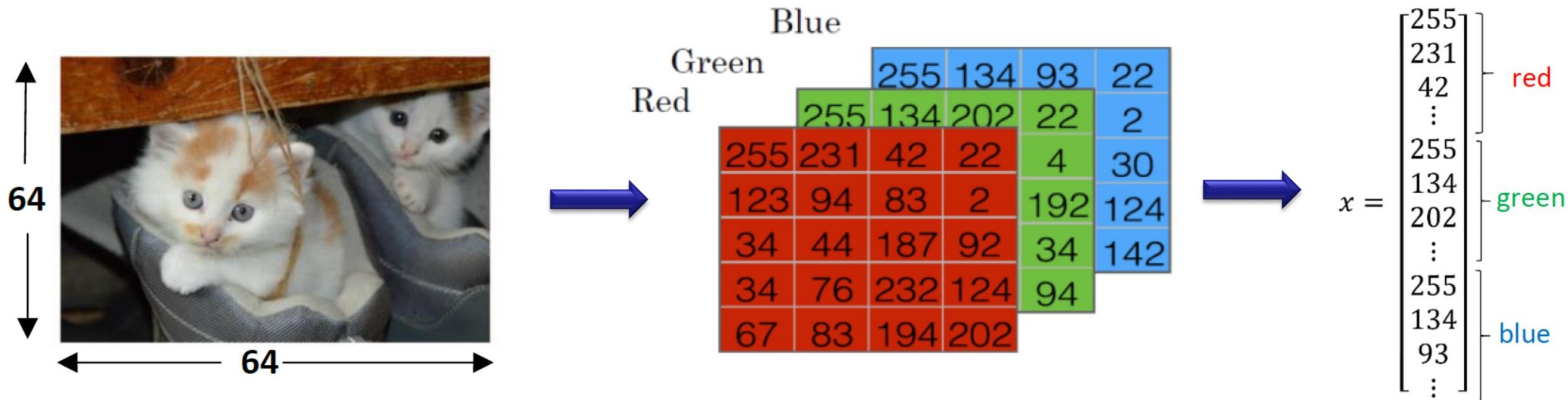


Figure 2. The overall view of Flow-Fuse-Tracker (FFT) for multiple object tracking. FlowTracker and FuseTracker (in bold grey boxes) are the two DNN modules of our FFT network. In the FlowTracker, the optical flow is generated from two sequential frames, and the target bboxes  $\{b_{t-1}^k\}$  (green dashed bboxes) at frame  $t-1$  are regressed to the bboxes  $\{b_t^k\}$  at frame  $t$  through the optical flow. In the FuseTracker, the bboxes from both  $\{b_t^k\}$  and public detections  $D_t$  (red bboxes) at frame  $t$  are refined and fused. The FuseTracker outputs the final tracking results (green bboxes with shadow). (Best viewed in color)

# 计算机视觉

11

## 图像的数字表示



一张图片数据量是 $64 \times 64 \times 3$ ，因为每张图片都有3个颜色通道。  
如果计算一下的话，可得知数据量为12288

**01** 计算机视觉概述

**02** 卷积神经网络概述

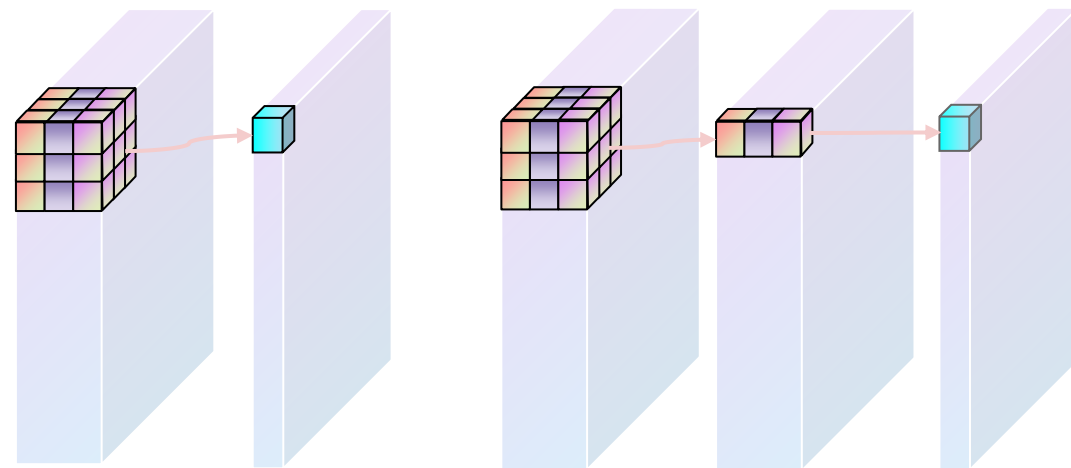
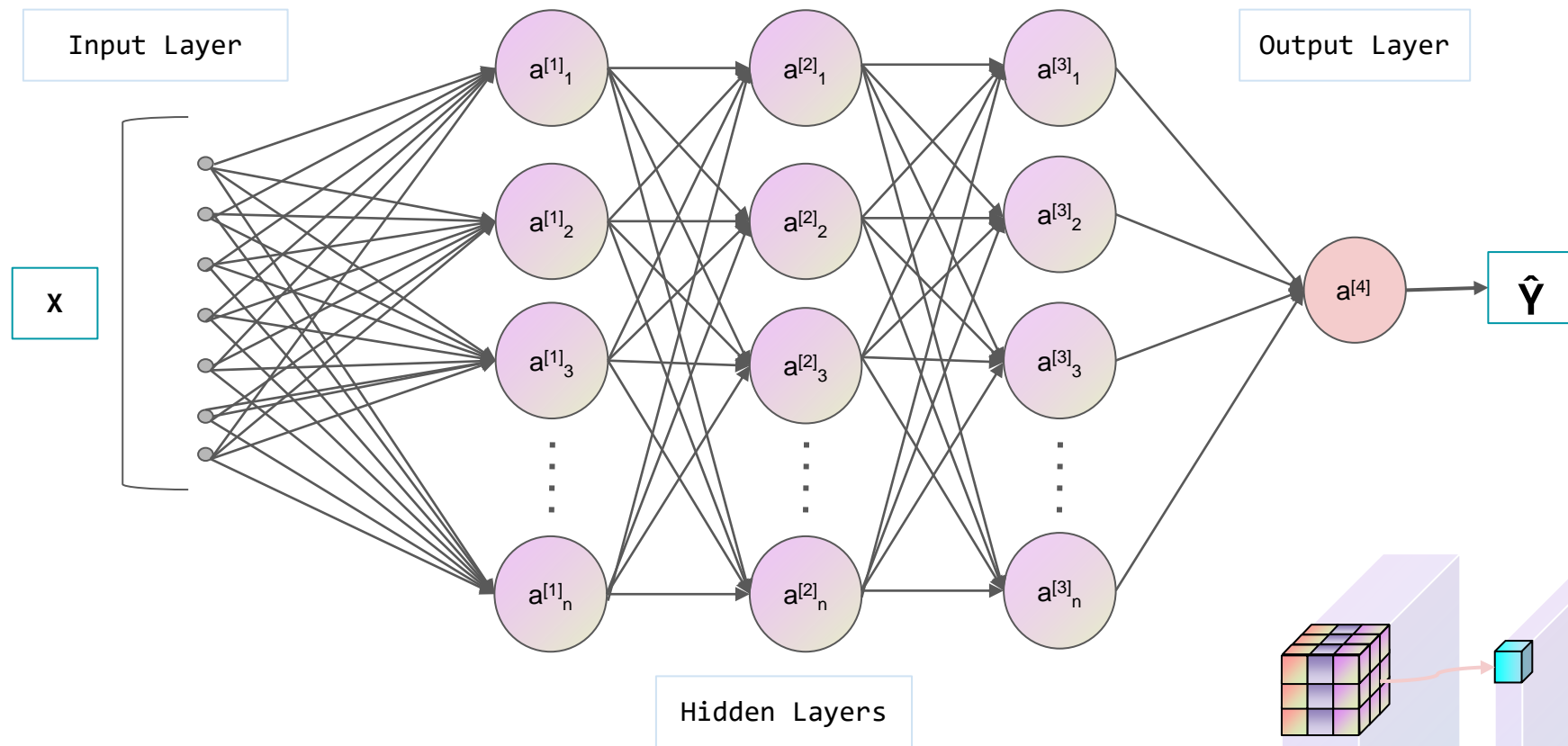
**03** 卷积神经网络计算

**04** 卷积神经网络案例



# 深层神经网络和卷积神经网络

13

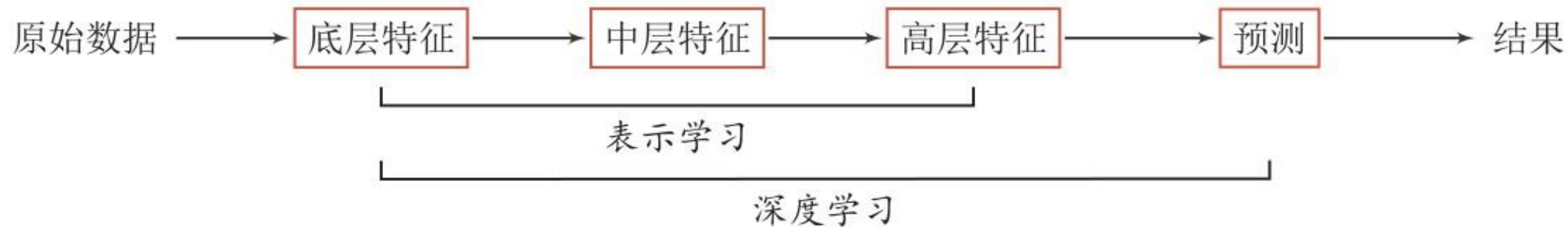
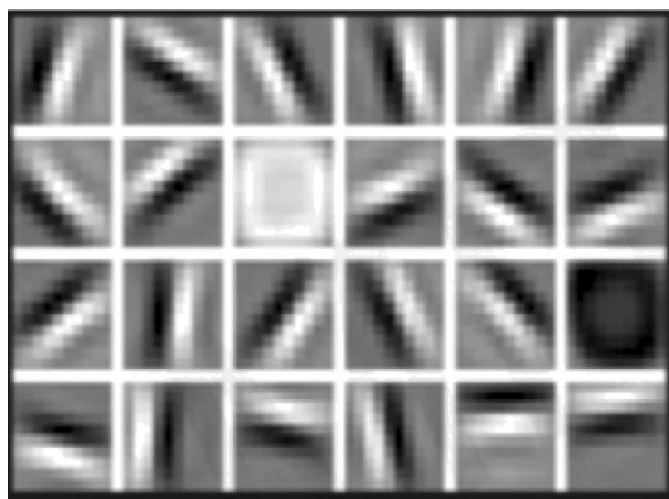


进行卷积计算，处理大量特征

# 卷积神经网络

14

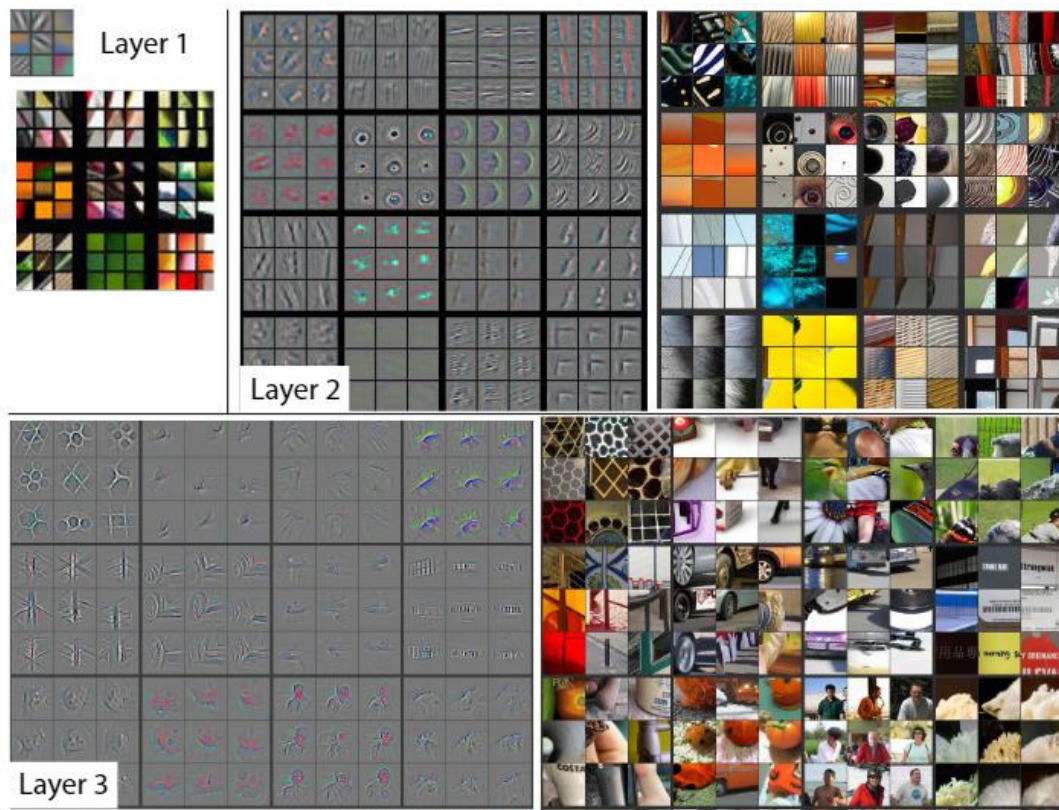
**深度学习 = 表示学习 + 浅层学习**



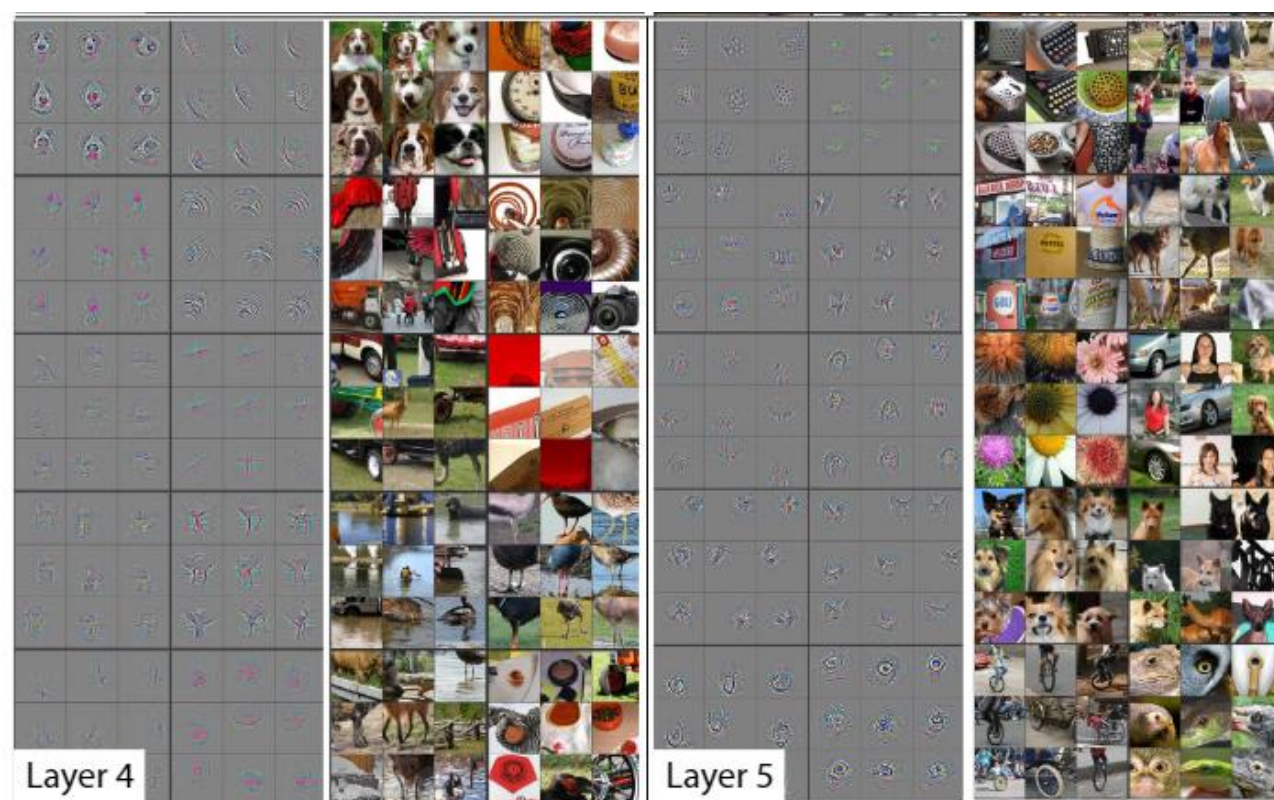
# 多层卷积能抽取复杂特征

15

**浅层**学到的特征为简单的边缘、角点、纹理、几何形状、表面等



**深层**学到的特征则更为复杂抽象，为狗、人脸、键盘等等





# 边缘检测

16

## 边缘检测

神经网络的前几层是通常检测边缘的，然后，后面的层有可能检测到物体的部分区域，更靠后的一些层可能检测到完整的物体

Vertical edge detection

3 <sup>1</sup>	0 <sup>0</sup>	1 <sup>-1</sup>	2	7	4
1 <sup>1</sup>	5 <sup>0</sup>	8 <sup>-1</sup>	9	3	1
2 <sup>1</sup>	7 <sup>0</sup>	2 <sup>-1</sup>	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

"convolution"  
\*

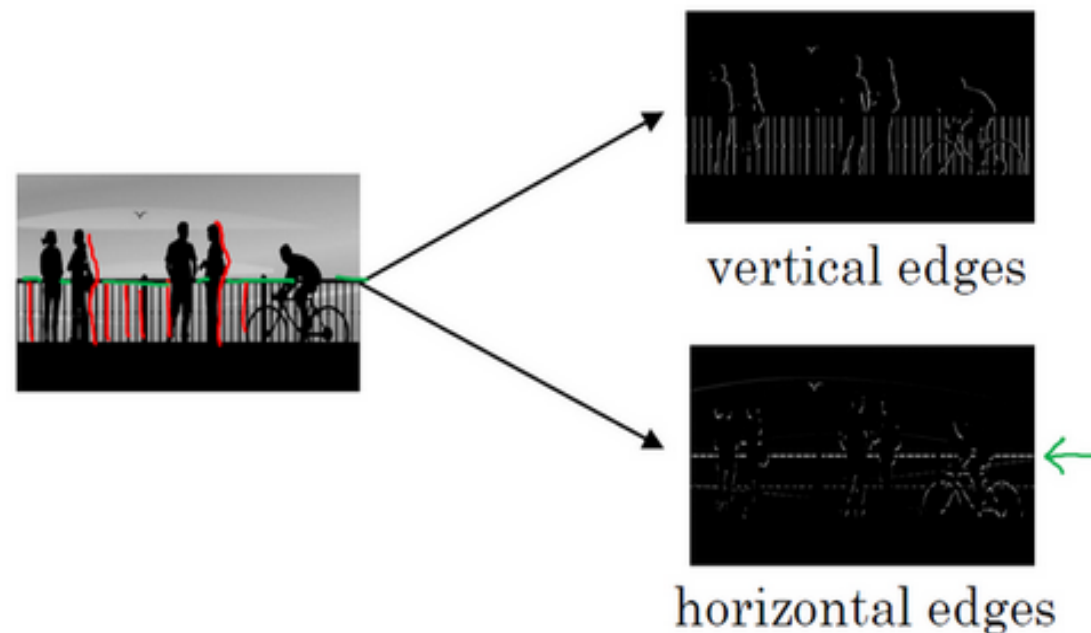
1	0	-1
1	0	-1
1	0	-1

3x3  
filter

=

-5				

4x4



$$\begin{bmatrix} 3 \times 1 & 0 \times 0 & 1 \times (-1) \\ 1 \times 1 & 5 \times 0 & 8 \times (-1) \\ 2 \times 1 & 7 \times 0 & 2 \times (-1) \end{bmatrix} = \begin{bmatrix} 3 & 0 & -1 \\ 1 & 0 & -8 \\ 2 & 0 & -2 \end{bmatrix}$$

$$3 + 1 + 2 + 0 + 0 + 0 + (-1) + (-8) + (-2) = -5$$



**01** 计算机视觉概述

**02** 卷积神经网络概述

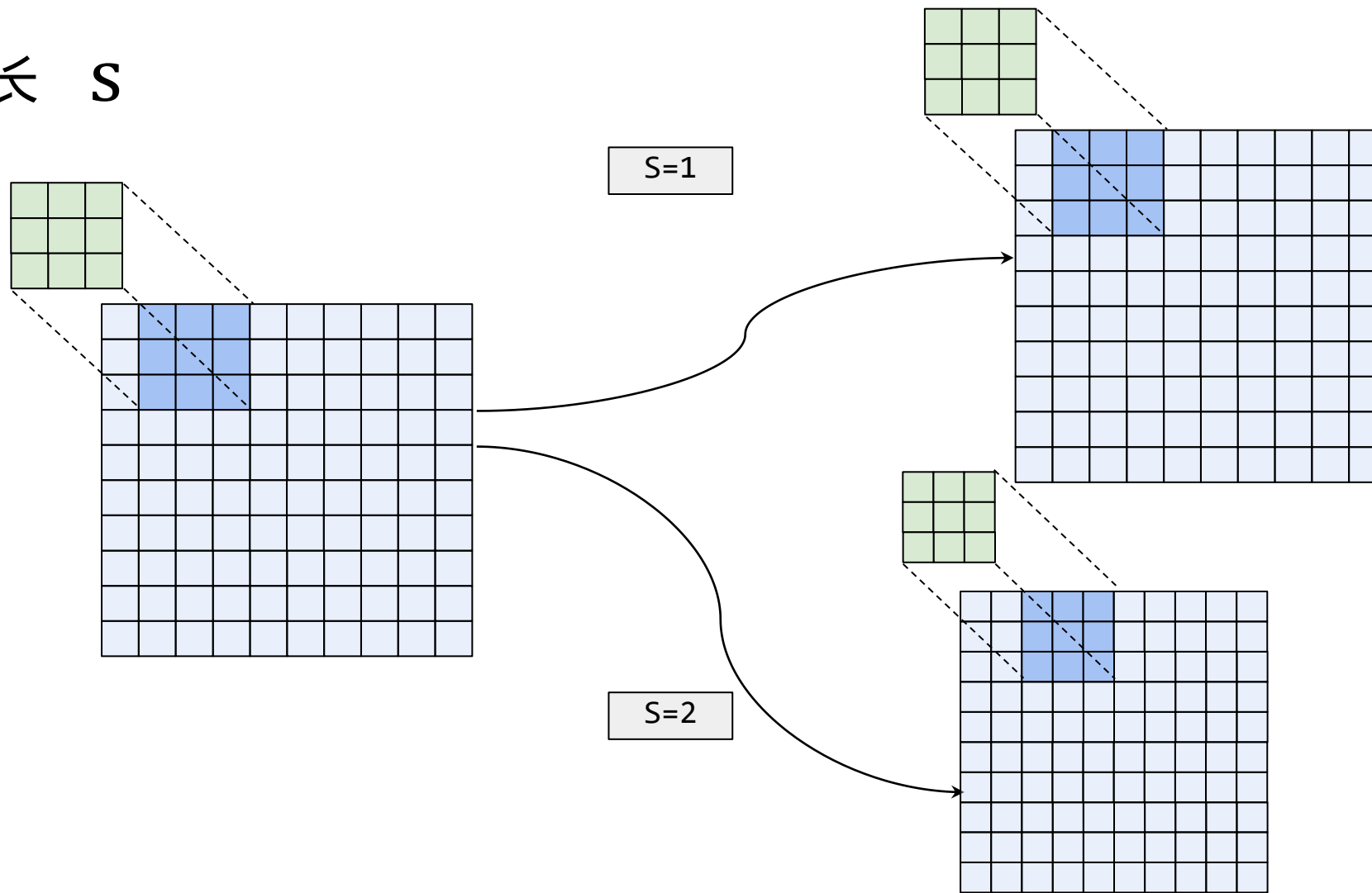
**03** 卷积神经网络计算

**04** 卷积神经网络案例

# 卷积步长

18

卷积步长  $S$



# 卷积计算

Padding  $p$  { **Other** same卷积  
**0** Valid卷积

卷积操作之前填充这幅图像

卷积步长  $s$

图像尺寸  $n$

Strided convolution

Input (7x7):

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3 <sup>3</sup>	4 <sup>4</sup>	6 <sup>4</sup>
3	2	4	1	9 <sup>1</sup>	8 <sup>0</sup>	3 <sup>2</sup>
0	1	3	9	2 <sup>-1</sup>	1 <sup>0</sup>	4 <sup>3</sup>

Kernel (3x3):

3	4	4
1	0	2
-1	0	3

Output (3x3):

91	100	83
69	91	127
44	72	74

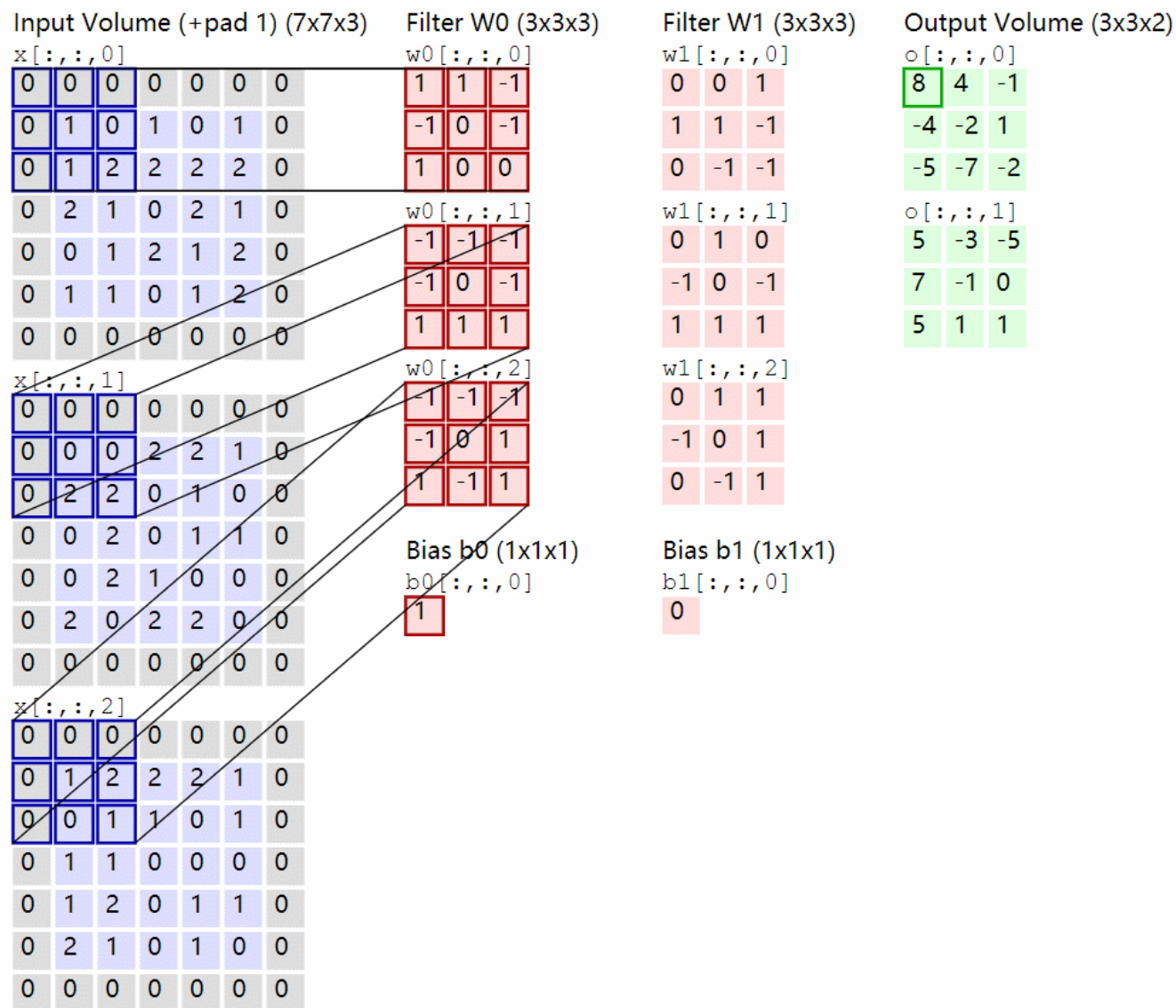
Handwritten notes:

- $n \times n * f \times f$
- padding  $p$
- stride  $s$
- $s = 2$
- $\lfloor \frac{n+p-f}{s} + 1 \rfloor \times \lfloor \frac{n+p-f}{s} + 1 \rfloor$
- $\frac{7+0-3}{2} + 1 = \frac{4}{2} + 1 = 3$
- $\lfloor 7 \rfloor = \text{floor}(7)$

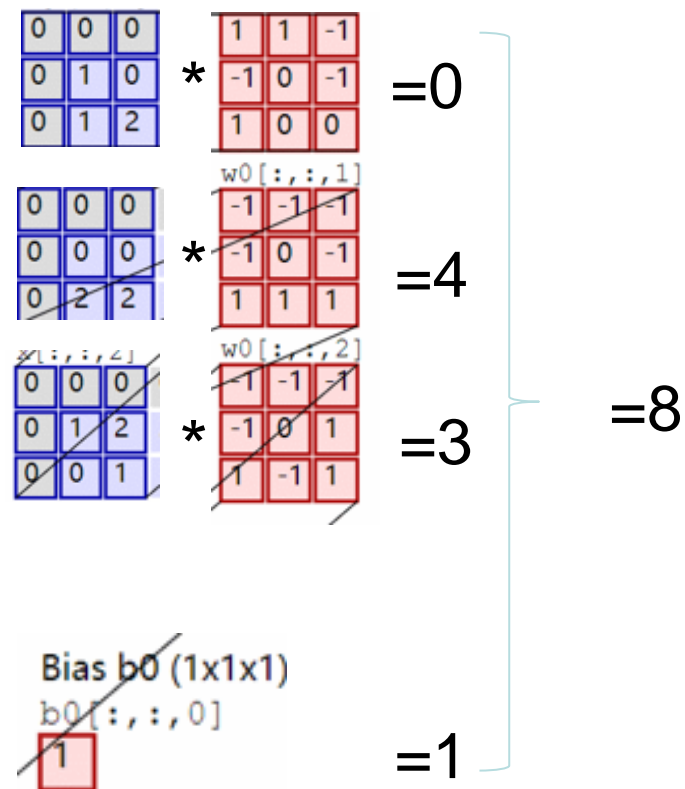
$$n = 7, p = 0, f = 3, s = 2, \frac{7+0-3}{2} + 1 = 3$$

# 三维卷积计算

20



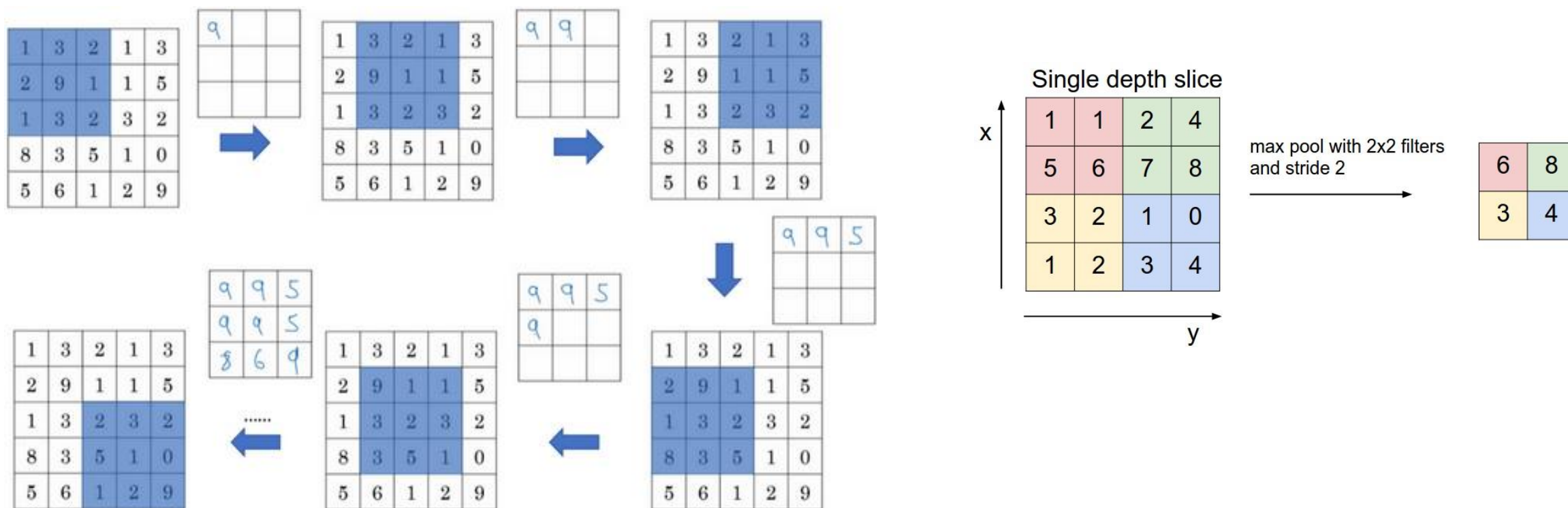
$$8=0+4+3+1$$





# 池化

21



最大池化的输入就是  $n_H \times n_W \times n_c$ , 假设没有padding, 则输出  $\lfloor \frac{n_H-f}{s} + 1 \rfloor \times \lfloor \frac{n_W-f}{s} + 1 \rfloor \times n_c$

平均池化不常用

# 4.卷积神经网络案例

22

**01** 计算机视觉概述

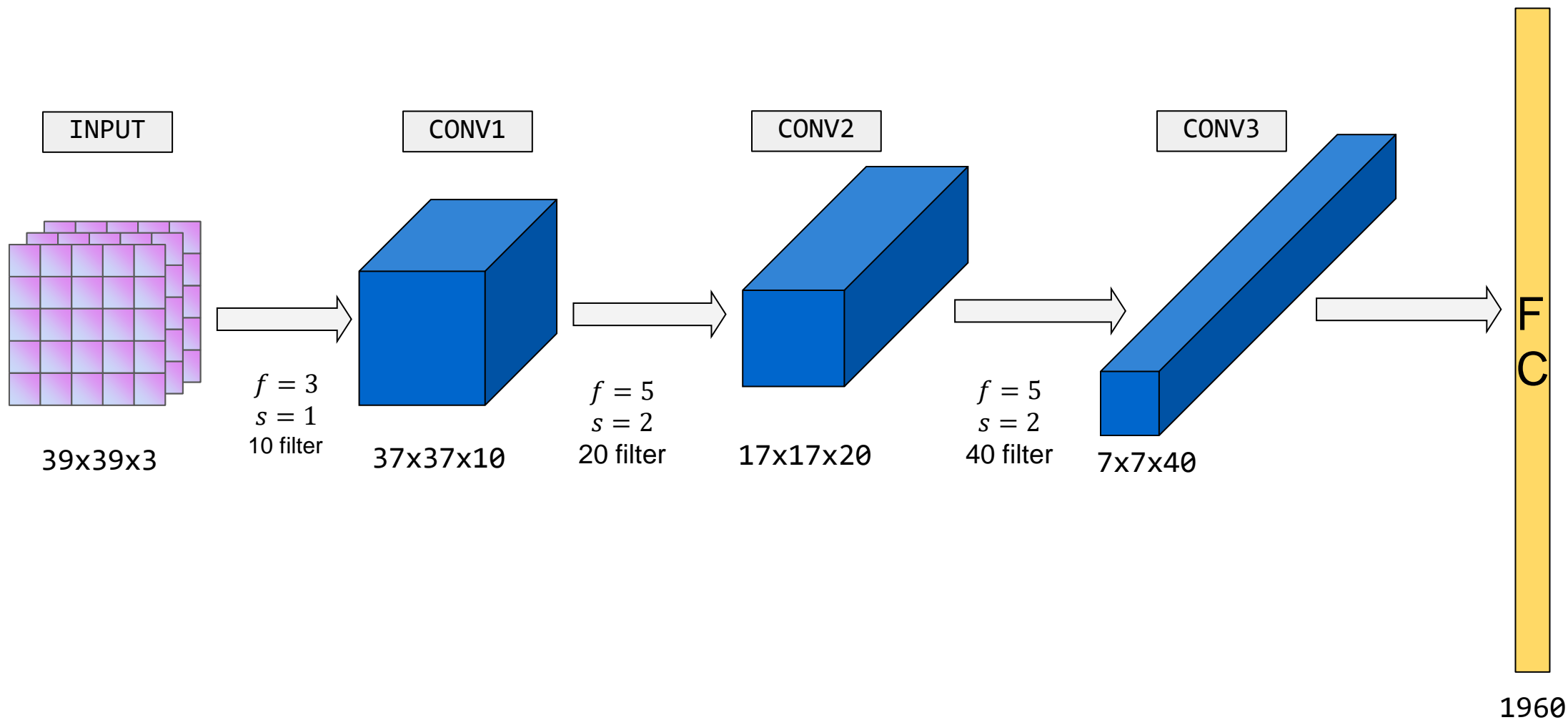
**02** 卷积神经网络概述

**03** 卷积神经网络计算

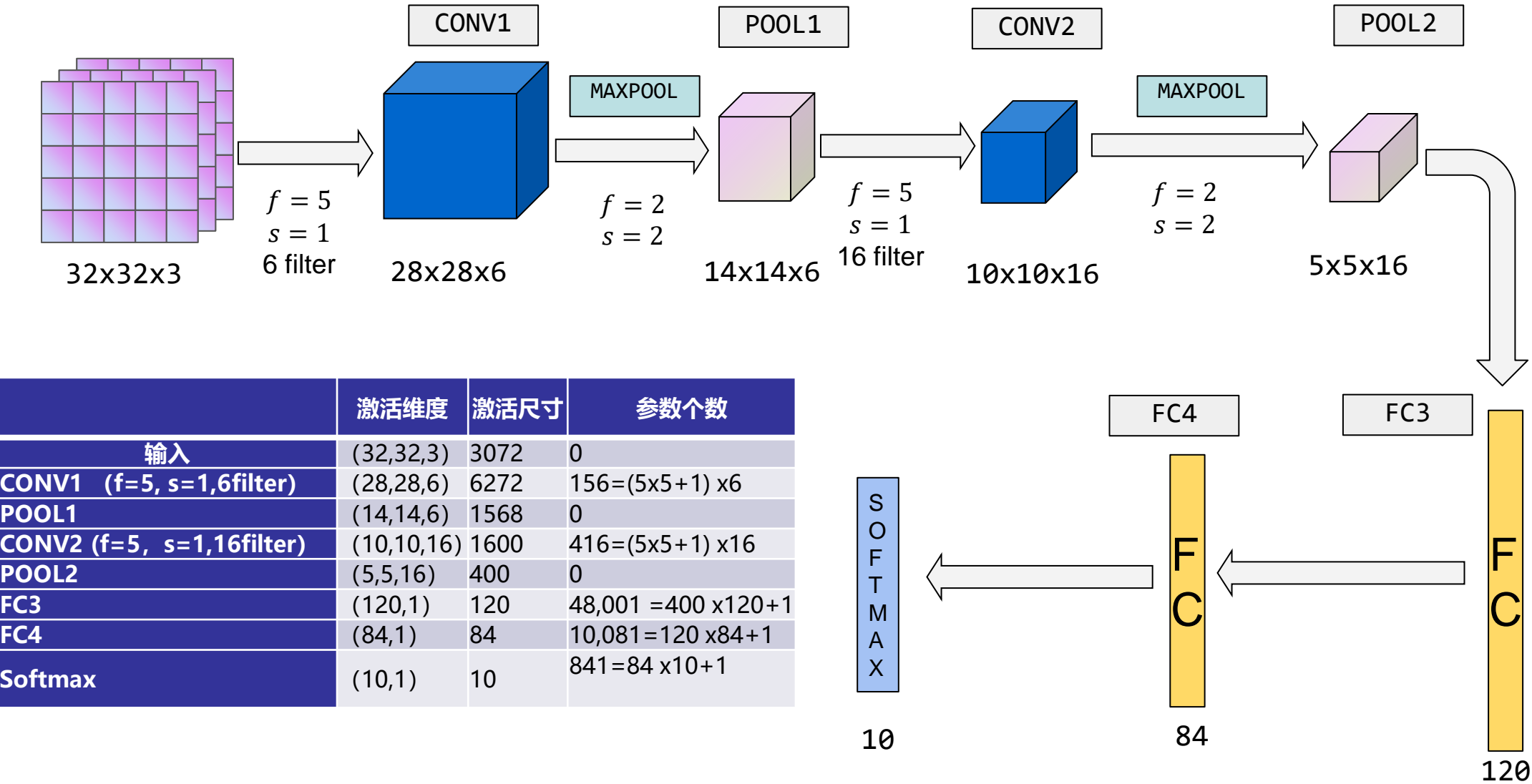
**04** 卷积神经网络案例

# 4.卷积神经网络案例1

23



# 4.卷积神经网络案例2





# 卷积神经网络作用

25

## 参数共享

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

 $*$ 

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

 $=$ 

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

如果你用一个 $3 \times 3$ 的过滤器检测垂直边缘，那么图片的左上角区域，以及旁边的各个区域（左边矩阵中蓝色方框标记的部分）都可以使用这个 $3 \times 3$ 的过滤器。即使减少参数个数，这9个参数同样能计算出16个输出。

# 卷积神经网络作用

26

## 稀疏连接

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

 $*$ 

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

 $=$ 

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

右图这个绿色格子的0是通过 $3 \times 3$ 的卷积计算得到的，它只依赖于这个 $3 \times 3$ 的输入的单元格，右边这个输出单元（元素0）仅与36个输入特征中9个相连接。而且其它像素值都不会对输出产生任影响，这就是稀疏连接的概念。

1. IAN GOODFELLOW等, 《深度学习》, 人民邮电出版社, 2017
2. Andrew Ng, <http://www.deeplearning.ai>

谢谢!

