

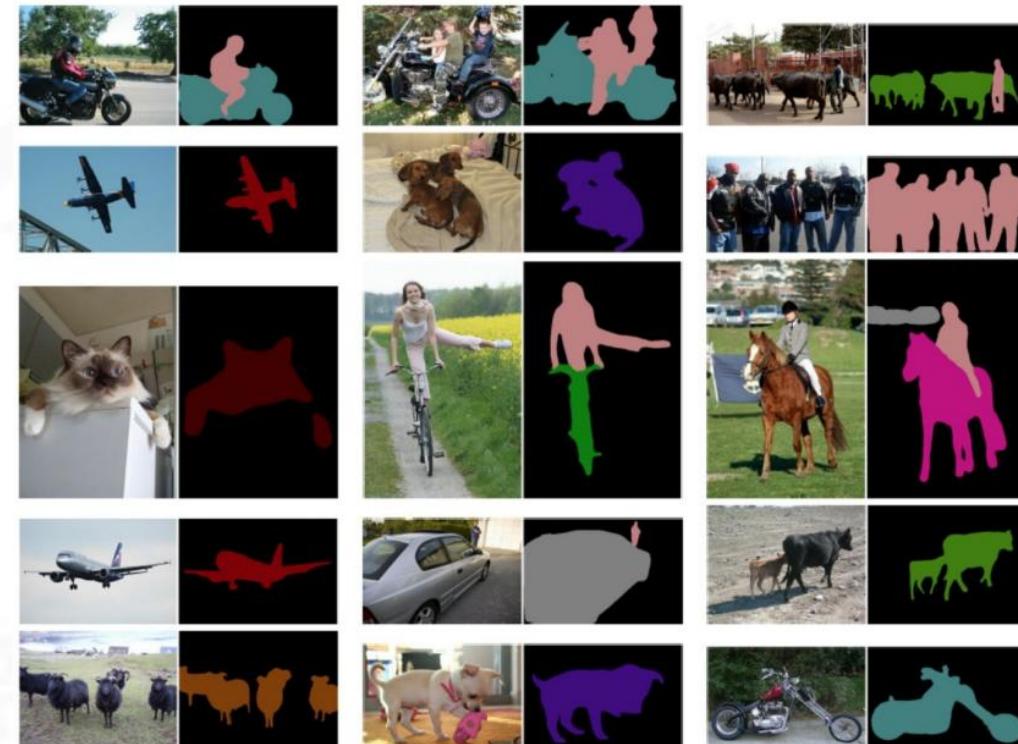
# 1. 背景概述



北京交通大学

## 图像分割

- ▲ **基本定义：**将一张图像分割成多个具有**语义意义**的区域或像素集合的过程。  
    旨在将图像中的不同区域或目标**准确地分离开来**，以实现对图像内容的**理解和分析**。
- ▲ **简单分类：**基于传统图像处理的方法和基于深度学习的方法。
- ▲ **研究意义：**是许多**视觉理解系统**中的一个重要组成部分，在广泛的应用中发挥核心作用，  
    包括医学图像分析、自动驾驶、视频监控和增强现实等众多领域。



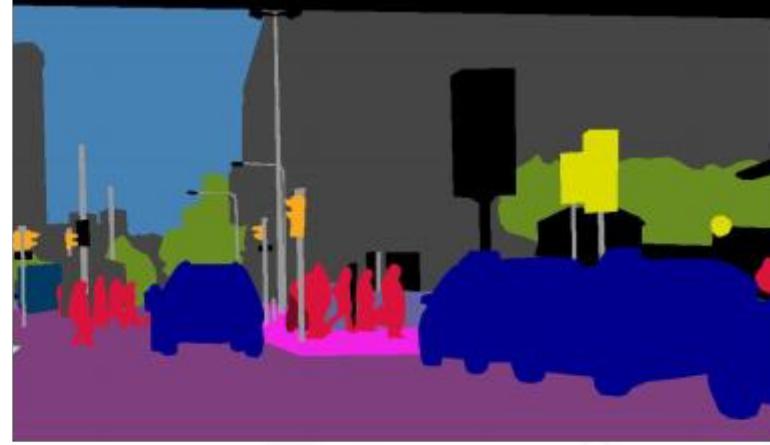
# 1. 背景概述



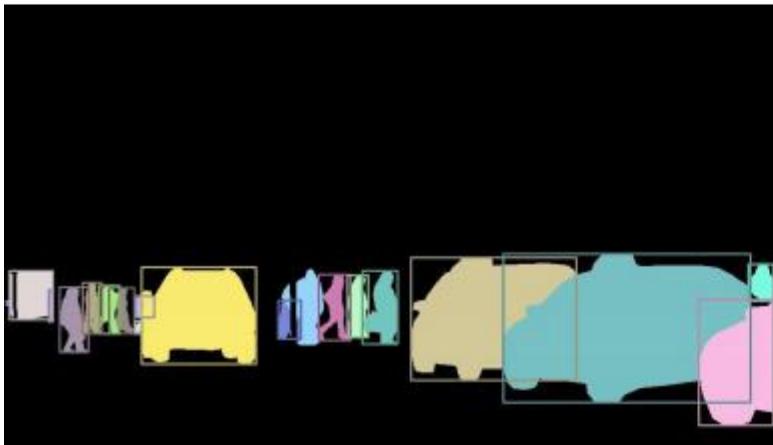
北京交通大学



待分割图像



语义分割



实例分割



全景分割

## ▲ 语义分割:

- (1) 目标类别
- (2) 背景类别

## ▲ 实例分割:

- (1) 目标类别
- (2) 目标实例

## ▲ 全景分割:

语义分割 & 实例分割

## 2.1 传统图像分割方法



北京交通大学

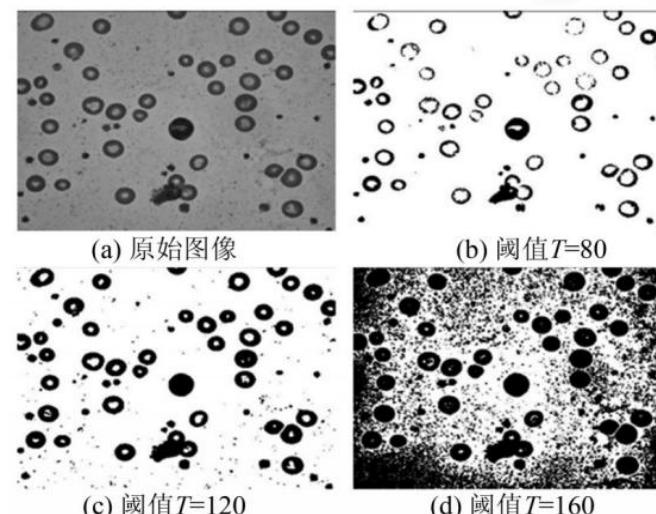
### ▲ 基于阈值的分割方法

- 实质是通过设定不同的灰度阈值，对图像灰度直方图进行分类，灰度值**在同一灰度范围内**的像素**认为属于同一类**且具有一定相似性。
- 用  $f(i,j)$  表示原始图像像素  $(i,j)$  的灰度值，通过设定阈值  $T$ ，将图像中的像素分为目标和背景两类，实现输入图像  $f(i,j)$  到输出图像  $g(i,j)$  的变换：

$$g(i,j) = \begin{cases} 1, & f(i,j) \geq T \\ 0, & f(i,j) < T \end{cases}$$

其中， $g(i,j) = 1$  表示属于目标类别的图像， $g(i,j) = 0$  表示属于背景类别的图像。

- 基于阈值的图像分割方法的关键是**选取合适的灰度阈值**，以准确地分割图像中的像素。



阈值  $T$  越大，则分为目标类别的像素点越多，图像逐渐由浅变深

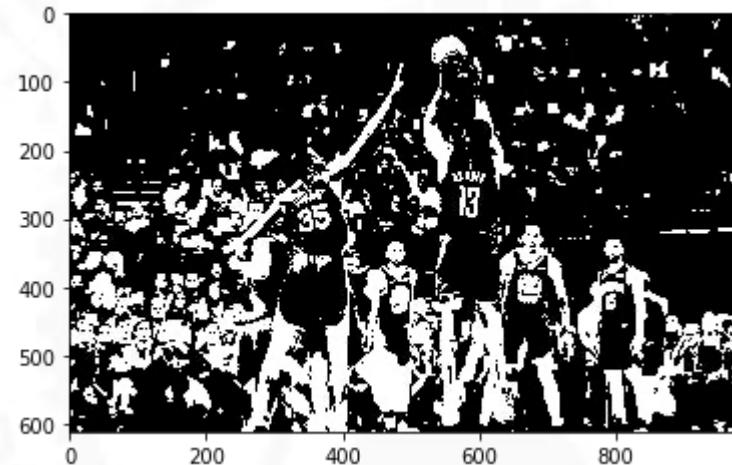
## 2.1 传统图像分割方法



北京交通大学

### ▲ 基于阈值的分割方法

- 基于阈值的图像分割方法根据不同的准则有不同的分类，常见的分类为：  
    基于**点**的全局阈值分割方法、基于**区域**的全局阈值分割方法、局部阈值分割方法等。
- 适用于目标灰度分布均匀、变化小，目标和背景灰度差异较明显的图像，  
    简单易实现且效率高。
- 通常只考虑像素自身的灰度值，**未考虑图像的语义、空间等特征信息**，且易受噪声影响，  
    对于复杂的图像，阈值分割的效果并不理想。
- 在实际分割应用中，基于阈值的分割方法通常作为预处理方法或与其他分割方法结合使用。



## 2.1 传统图像分割方法



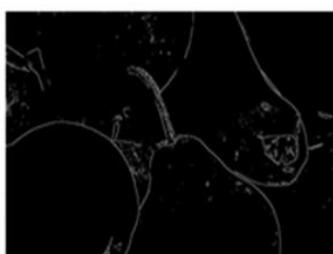
北京交通大学

### ▲ 基于边缘的分割方法

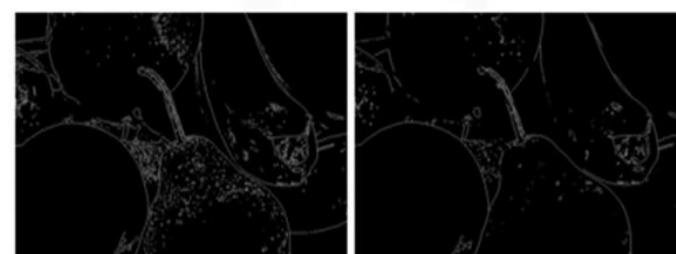
- 在图像中若**某个像素点与相邻像素点的灰度值差异较大，则认为该像素点可能处于边界处。**若能检测出这些边界处像素点，连接它们形成边缘轮廓，从而将图像划分成不同的区域。
- 根据处理策略的不同，基于边缘的分割方法，可分为串行边缘检测法和并行边缘检测法。
  - **串行边缘检测法：**需先检测出**边缘起始点**，从起始点出发，通过相似性准则搜索并连接相邻边缘点，完成图像边缘的检测。
  - **并行边缘检测法：**借助**空域微分算子**，用其模板与图像进行卷积，完成图像边缘的检测。
- 在实际应用中，并行边缘检测法直接借助微分算子进行卷积实现分割，过程简单快捷，性能相对优良，是最常用的边缘检测法。
- 常用的边缘检测微分算子有：**Roberts**、**Sobel**、**Prewitt**、**LoG**、**Canny**等。



(a) 原始图像



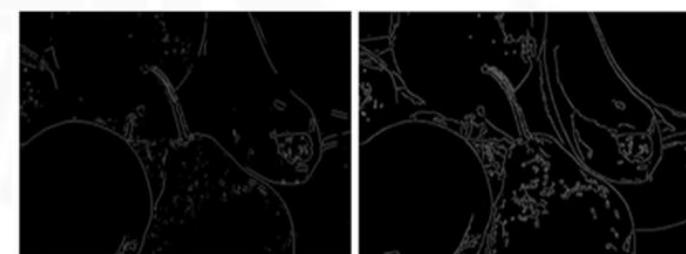
(b) Roberts算子



(c) Sobel算子



(d) Prewitt算子



(e) LOG算子

(f) Canny算子

## 2.1 传统图像分割方法



北京交通大学

### ▲ 基于区域的分割方法

- 根据图像的空间信息进行分割，通过像素的相似性特征对像素点进行分类并构成区域。
- 较为常用的有区域生长法和分裂合并法。

➤ **区域生长法**: 通过将具有相似性质的像素点集合起来，构成独立的区域，以实现分割。

具体步骤：先选择一组种子点（单个像素或小区域）作为生长起点，然后根据生长准则，**将种子点附近与其具有相似特征的像素点归并到种子点所在的像素区域内**，再将新像素作为种子点，反复迭代至所有区域停止生长。

➤ **分裂合并法**: 是通过不断地分裂合并，得到图像各子区域。

具体步骤为：先将图像划分为规则的区域，然后根据相似性准则，**分裂那些特性不同的区域，合并特性相同的邻近区域**，直至没有分裂合并发生。



(a) 原始图像



(b) 图像灰度化



(c) 区域生长法



(d) 分裂合并法

- ◆ 区域生长法计算简单，但对噪声敏感，易导致区域空缺，图中头盔受背景颜色的干扰出现了残缺的现象；
- ◆ 分裂合并法分割法对复杂图像有较好的效果，但其计算复杂，且分裂时边界可能被破坏，途中车轮的轮廓信息在合并过程中被破坏，导致车轮边缘出现了模糊现象。

## 2.1 传统图像分割方法



北京交通大学

### ▲ 基于聚类的分割方法

- 将**具有特征相似性的像素点**聚集到同一区域，反复迭代聚类结果至收敛，最终将所有像素点聚集到几个不同的类别中，完成图像区域的划分，从而实现分割。
- 较为常用的有简单线性迭代聚类法(SLIC)。
- SLIC：基于聚类思想，将图像中的像素划分为超像素块，也称为**超像素分割**。
- SLIC的具体步骤：
  - 将RGB彩色图像通过映射转化到Lab颜色空间，Lab颜色空间由  $(L, a, b)$  三元素组成，其中， $L$  代表亮度， $a$  代表从洋红色至绿色的范围， $b$  表示从黄色至蓝色的范围。相比于 RGB 空间，Lab空间能够保留更宽的色彩区域，提供更加丰富的色彩特征。
  - 将每个像素点颜色特征  $(L, a, b)$  及坐标  $(x, y)$  组合成向量  $(L, a, b, x, y)$  进行距离度量，包括像素点  $i$  和  $j$  之间的颜色距离  $d_c$  和空间距离  $d_s$ ：

$$d_c = \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2}$$

$$d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$$

$l_n(n = i, j)$ 表示在颜色空间中亮度的特征距离值；

$a_n$ 、 $b_n$ 分别表示在颜色空间中色阶品红、正黄系的特征距离值；

$x_n$ 、 $y_n$ 分别表示像素点的横、纵坐标值。

## 2.1 传统图像分割方法



北京交通大学

### ▲ 基于聚类的分割方法

(接上页) 再通过  $D'$  对最终距离进行度量:

$$D' = \sqrt{\left(\frac{d_c}{N_c}\right)^2 + \left(\frac{d_s}{N_s}\right)^2}$$

其中， $N_c$ 表示最大颜色距离，通常取常数m； $N_s$ 是类内最大空间距离， $N_s = S = \sqrt{N/K}$   
 $\sqrt{N/K}$ ， $N$ 是图中像素点总数， $K$ 为预分割超像素块的总和，超像素块的大小为 $N/K$ ，相邻种子点距离为S。

- 综上，两个像素点之间的距离度量公式可表示为：

$$D' = \sqrt{\left(\frac{d_c}{m}\right)^2 + \left(\frac{d_s}{S}\right)^2}$$

- 超像素SLIC算法中，像素间的相似性由对应  $(L, a, b, x, y)$  向量间的距离度量，**两个向量的距离越小则对应像素点的性质越相似**，反之，则对应像素点的性质相似性越低。
- 基于聚类的图像分割方法利用图像灰度、纹理等特征信息作为聚类准则，将图像分割转化成像素点聚类的问题，性能稳定且鲁棒性好。

超像素 SLIC 算法图像根据纹理特征，将图像划分为多个局部小区域，前景目标荷花和荷叶有明显的边缘轮廓信息



(a) 原始图像



(b) 超像素SLIC算法

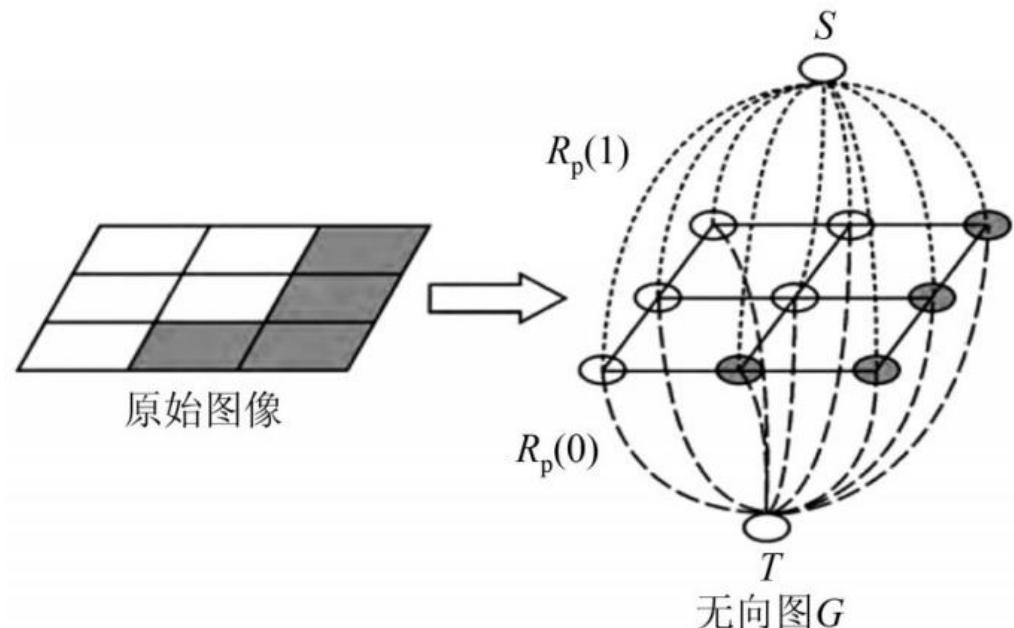
## 2.1 传统图像分割方法



北京交通大学

### ▲ 基于图论的分割方法

- 将分割问题转换成图的划分，通过对目标函数的最优化求解，完成分割过程，包括：**Graph Cut**、**Grab Cut**和**One Cut**等常用算法。
- Graph Cut 算法基于图论的思想，将最小割（min cut）问题应用到图像分割问题中，可以将图像分割为前景和背景，是经典的基于图论的图像分割方法。



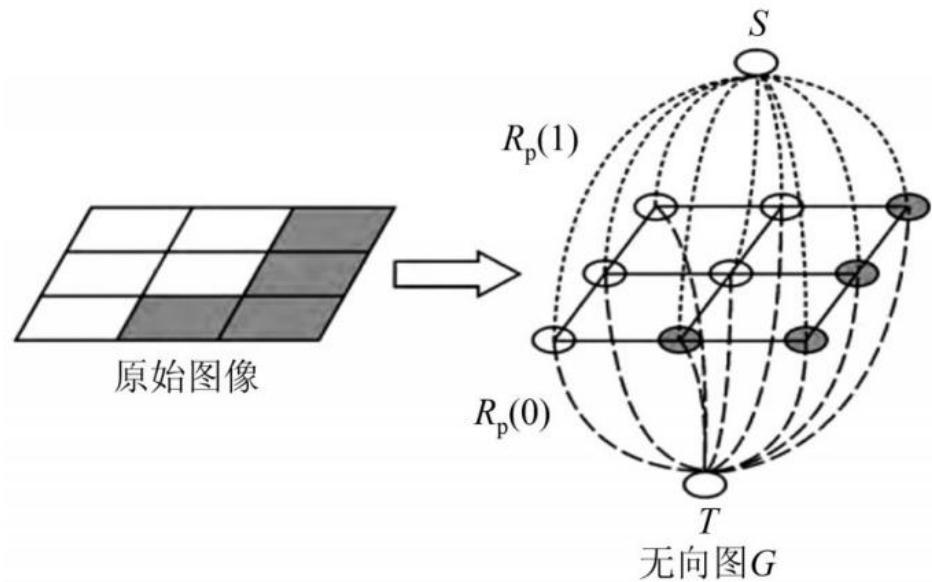
- ◆ 先将图像映射为带有权重的无向图  $G=(V, E)$ ，其中，无向图中的节点对应原图中的像素点，对每个相邻点进行连接形成边（实线），边的权重代表像素点之间的相似性。
- ◆ 除此之外，每个节点还要和终端顶点S和T进行连接形成边（虚线），与S相连的边  $R_p(1)$  的权重由该节点（像素点）前景目标概率表示，与T相连的边  $R_p(0)$  的权重由该节点的背景概率表示。
- ◆ 这样处理后，在无向图中就会形成两种顶点和边：一种是代表像素点的普通节点以及普通节点彼此相连形成的边；另一种是终端顶点S和T以及连接它和节点的边。

## 2.1 传统图像分割方法



北京交通大学

### ▲ 基于图论的分割方法



(a) 原始图像



(b) Graph Cut算法

- ◆ 如果边集合 $E$ 中的所有边都断开，将会导致 $S-T$ 图的分开，称之为cut。若一种cut的过程中其对应边的所有权值之和最小，则称之为min cut，对应的能量损失函数最小。
- ◆ 至此，**将复杂的图像分割问题转化成了求解能量损失函数最小值的问题。**
- ◆ 通过寻找 min cut 过程的不断迭代，求得能量损失函数最小值，就可以实现前景目标与背景的分离，从而实现图像分割。

- 使用Graph Cut算法对图片进行分割，可以获取前景目标大致的轮廓，实现目标与背景的分离。
- 基于图论的Graph Cut算法在利用图像灰度信息的同时使用了区域边界信息，通过最优化求解，得到最好的分割效果。
- 然而，该算法计算量大，且更倾向于对具有相同类内相似度的图像进行分割。

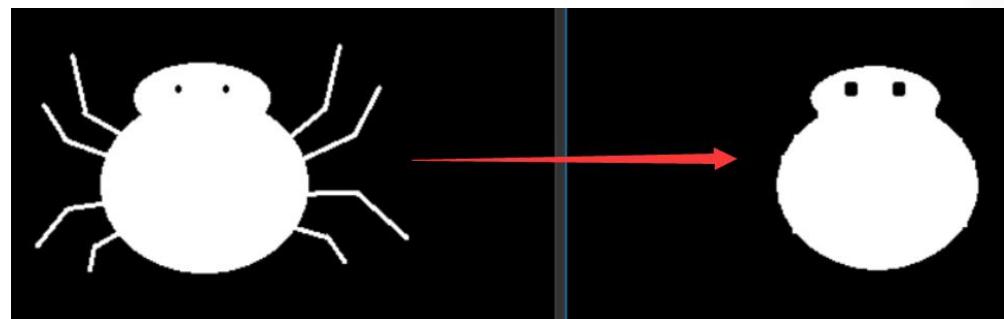
## 2.1 传统图像分割方法



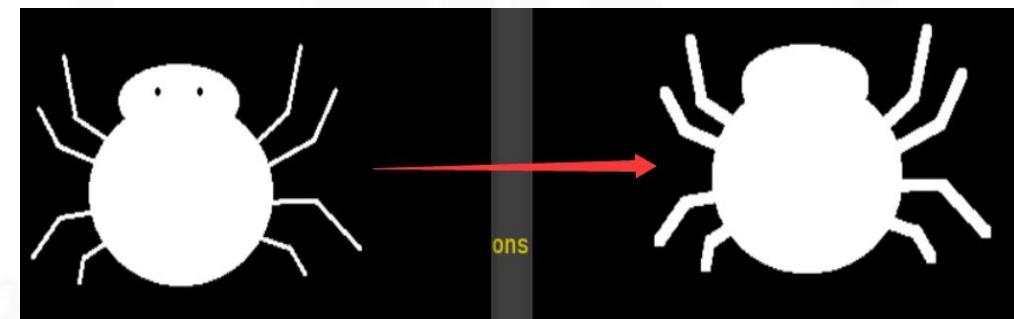
北京交通大学

### ▲ 基于特定理论的分割方法

- ◆ 随着分割任务要求及复杂度的提高，图像分割方法也在不断地改进，特别是在新理论和新方法的发展中，针对图像分割任务，出现了很多特定理论和方法。
- 数学形态学法：例如**腐蚀、膨胀、开运算、闭运算**。腐蚀与膨胀能够消除噪声；分割出独立的图像元素，连接相邻的元素；寻找图像中极大值区域或极小值区域；计算图像梯度等。  
**腐蚀和膨胀是对于图像高亮区域而言**。膨胀会使高亮区域更大，腐蚀则会让高亮区域更小。膨胀是求局部最大值的操作，而腐蚀则是求局部最小值的操作。
- 遗传算法：模拟自然的优胜劣汰获得最优解，实现最优化的分割。
- 其他常用方法：**小波变换法**、活动轮廓模型、模糊理论、粗糙集理论等。



腐 蚀



膨 胀

## 2.1 传统图像分割方法



北京交通大学

PASCAL VOC



Microsoft COCO



(a) 原始灰度图



(b) OTSU阈值法



(c) Canny边缘检测法



(d) 区域生长法



(e) Mean Shift聚类法



(f) Graph Cut图论法

- **OTSU阈值法**对于灰度区别较大的图像，能够明显地分割出前景目标（人和车）；
- Canny边缘检测法分割结果中的边缘轮廓信息比较明显，但也存在很多杂乱的噪声点；
- 区域生长法能够分割出前景目标，但是目标区域的细节分割不够精细；
- 基于聚类的Mean Shift算法分割可以对目标和背景区域进行各自的聚类；
- 基于图论的Graph Cut算法能够明显地区分出前景目标和背景，但也引入了噪声点。

## 2.1 传统图像分割方法



北京交通大学

分割方法	使用图像	难易度	耗时	内存
OTSU阈值法	目标与背景的 灰度差值大	易	短	小
Canny边缘检测法	像素灰度值 突变明显	一般	短	小
区域生长法	大部分图像	难	长	大
Mean Shift算法	目标类间性质 区别明显	较难	较长	大
Graph Cut算法	大部分图像	较难	较长	较大

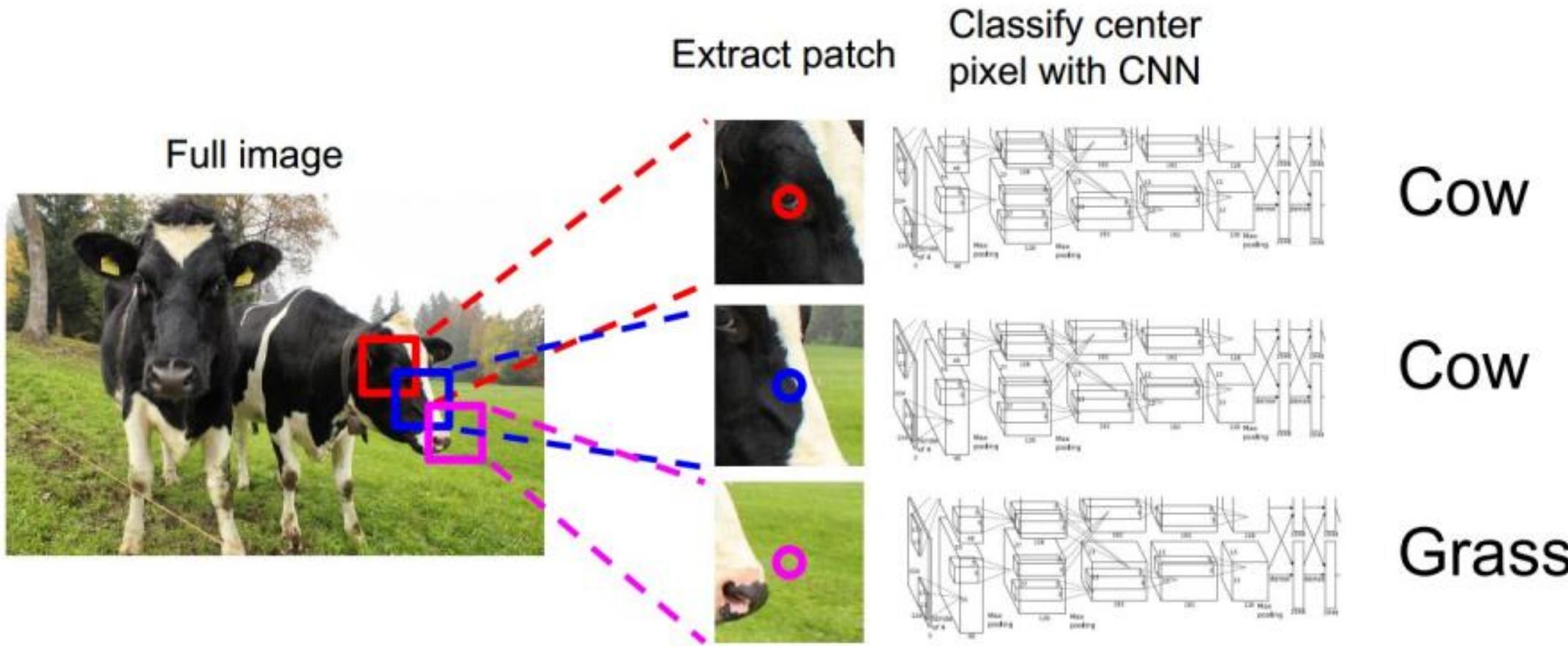
传统分割方法在分割时**会引入很多无关或者没有意义的阴影和区域**，有的甚至会出现噪声点，对分割结果造成干扰，无法精准地区分出前景目标和背景。

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### 原始方法1：滑动窗口



▲ 存在问题：

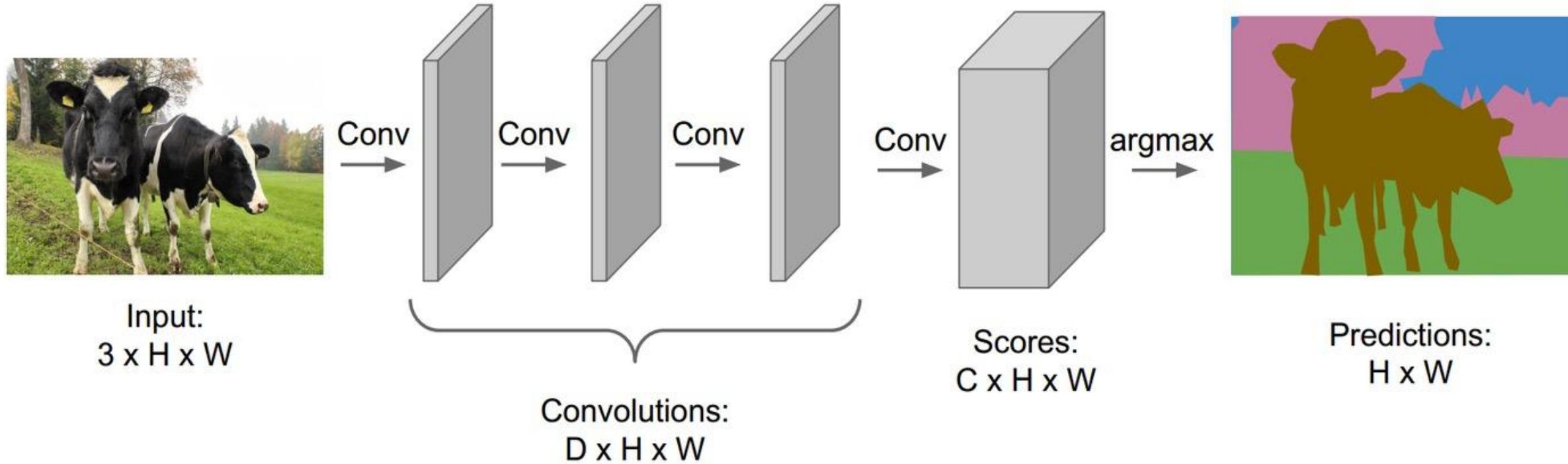
- (1) 计算代价高昂； (2) 全局上下文信息丢失

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### 原始方法2：不带降采样的CNN



#### ▲ 存在问题：

- (1) 感受野较小；
- (2) 对于大尺寸图像计算代价昂贵；
- (3) 不能使用现存的预训练网络

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

两个重要的发展线索：



- ▲ 使用大感受野（例如使用下采样），以捕捉全局背景的大空间区域
  - 但具有大感受野的特征图的分辨率较低
  - 找到从低分辨率到图像级分辨率的方法
- ▲ 使用低分辨率层的信息以捕捉更精细的细节

## 2.2.0 基于深度学习的分割方法—知识预备

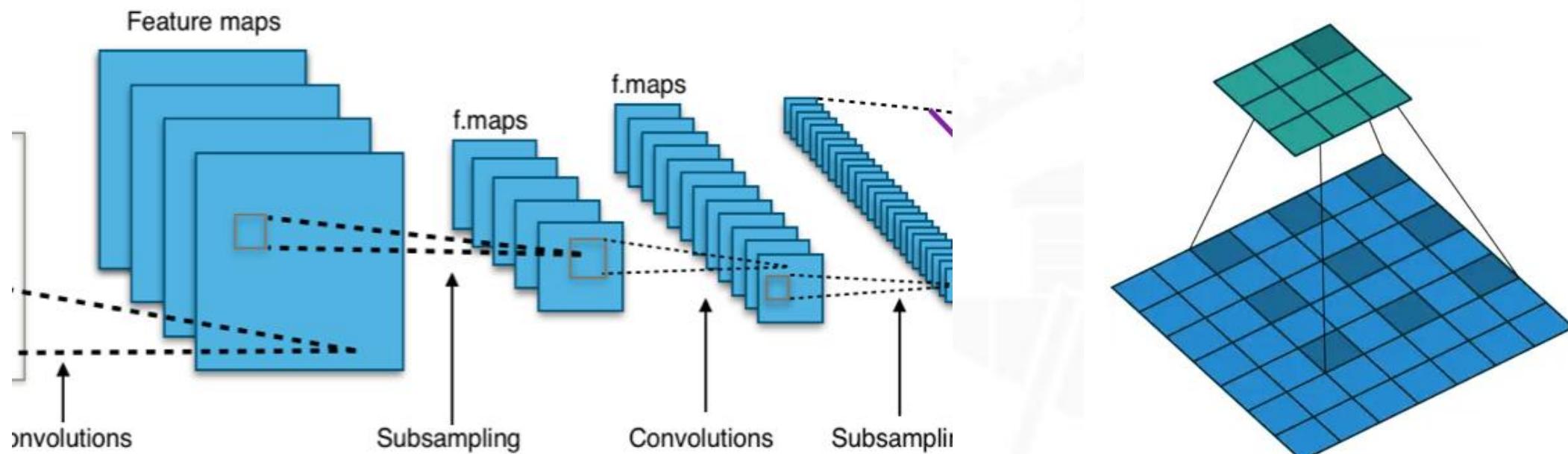


北京交通大学

问题1：如何捕获全局上下文信息？

➤ 两种思路：

- (1) 使用最大/平均池或步长>1的卷积对特征图进行降采样
- (2) 使用“膨胀”卷积 (dilated convolutions)，也称“空洞”卷积 (atrous convolutions)



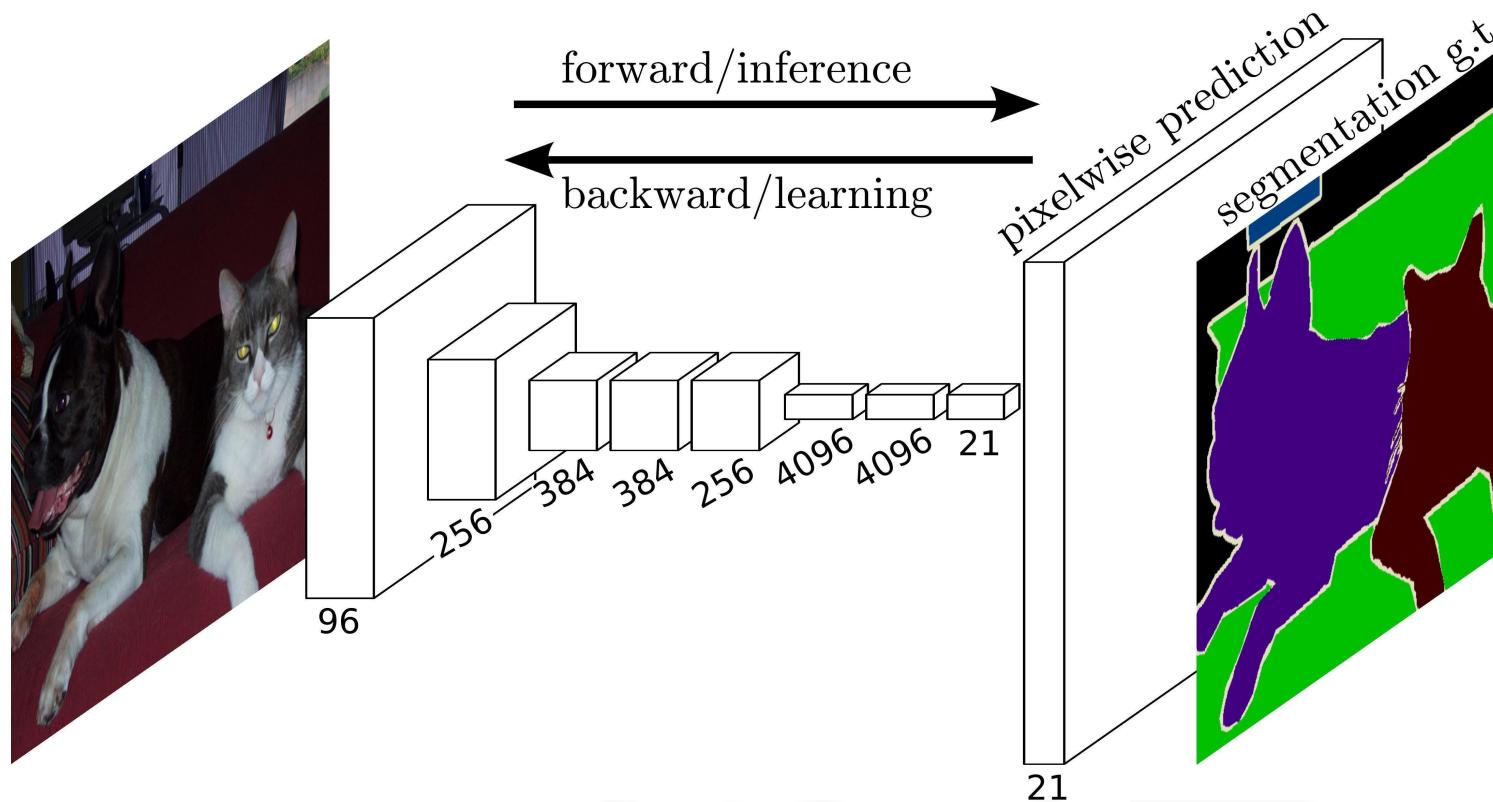
## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### 解决方案1：简单的Encoder架构

- 仅使用卷积将输入图像编码为特征向量
- 将已编码的特征向量 “上采样” 到图像大小的分辨率



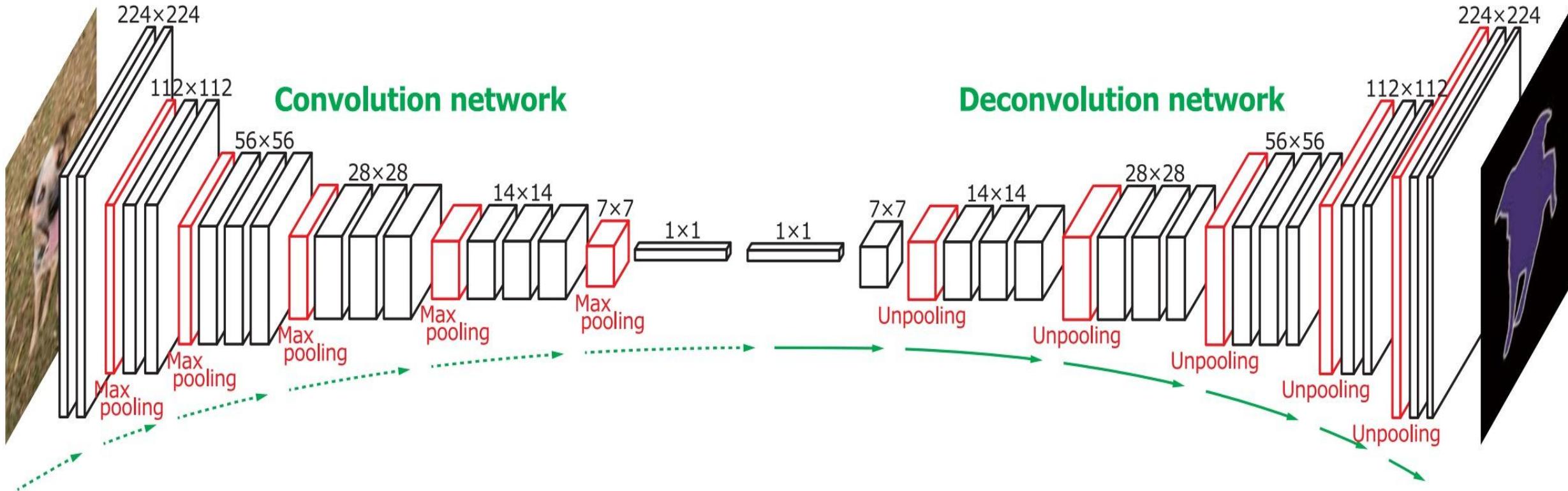
从低分辨率直接一步解码到了高分辨率！！！

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### 解决方案2：Encoder-Decoder架构



后续内容会解释*Unpooling*的原理~~

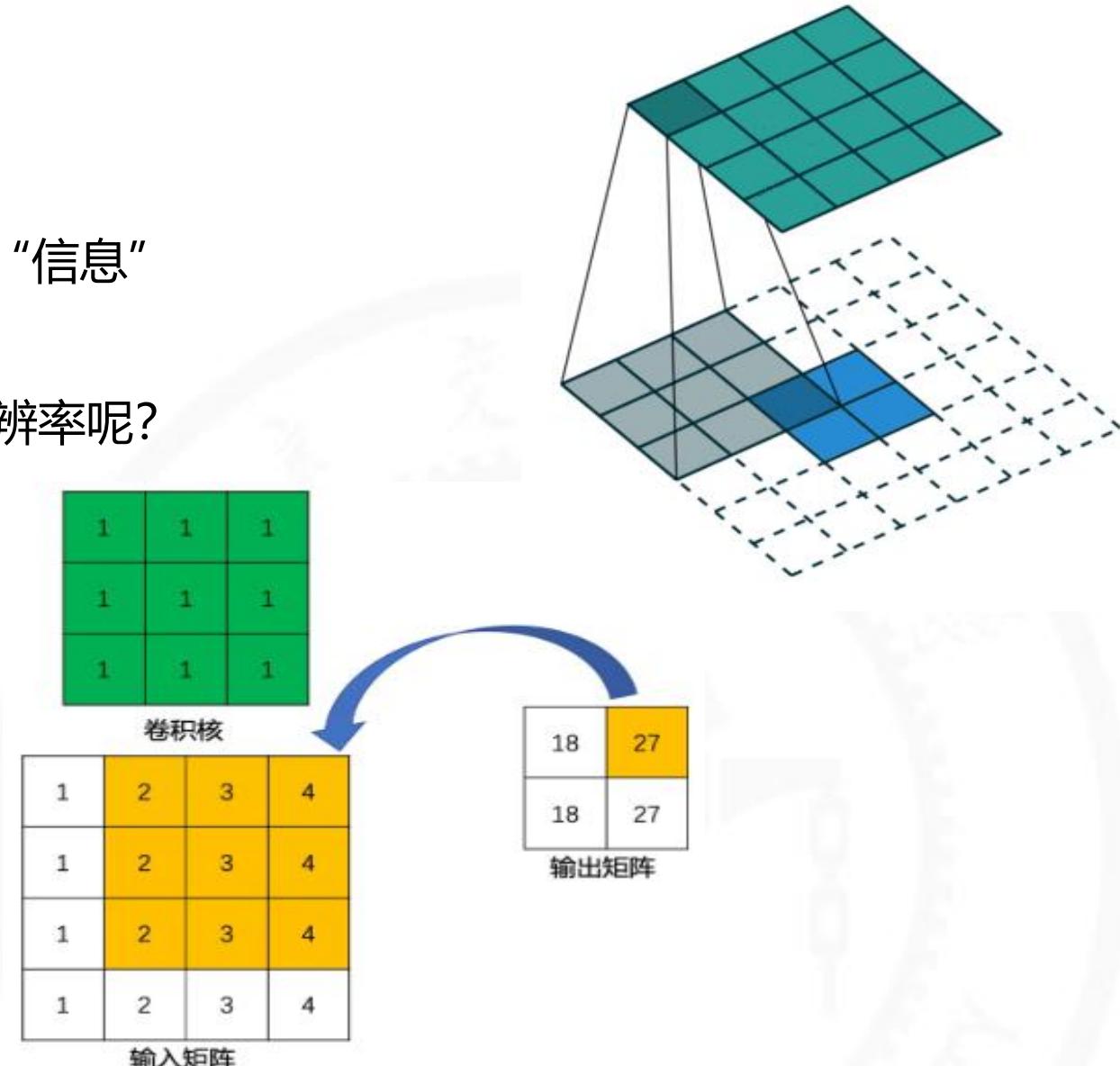
## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### 问题2：如何上采样特征？

- 下采样是所有CNN架构的一部分
  - 有助于覆盖图像的大空间区域。
  - 更深的低分辨率图告诉我们图像中的“信息”
- 但是，图像分割 = 像素级标签
- 那么最终该如何恢复到图像级大小的分辨率呢？
- 有许多解决方案：
  - 最近邻、双线性、双三次上采样技术
  - Unpooling
  - 转置卷积或反卷积



## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### ▲ 最近邻上采样

1	2
3	4



1	1	1	2	2	2
1	1	1	2	2	2
1	1	1	2	2	2
3	3	3	4	4	4
3	3	3	4	4	4
3	3	3	4	4	4

**Input:** C x 2 x 2

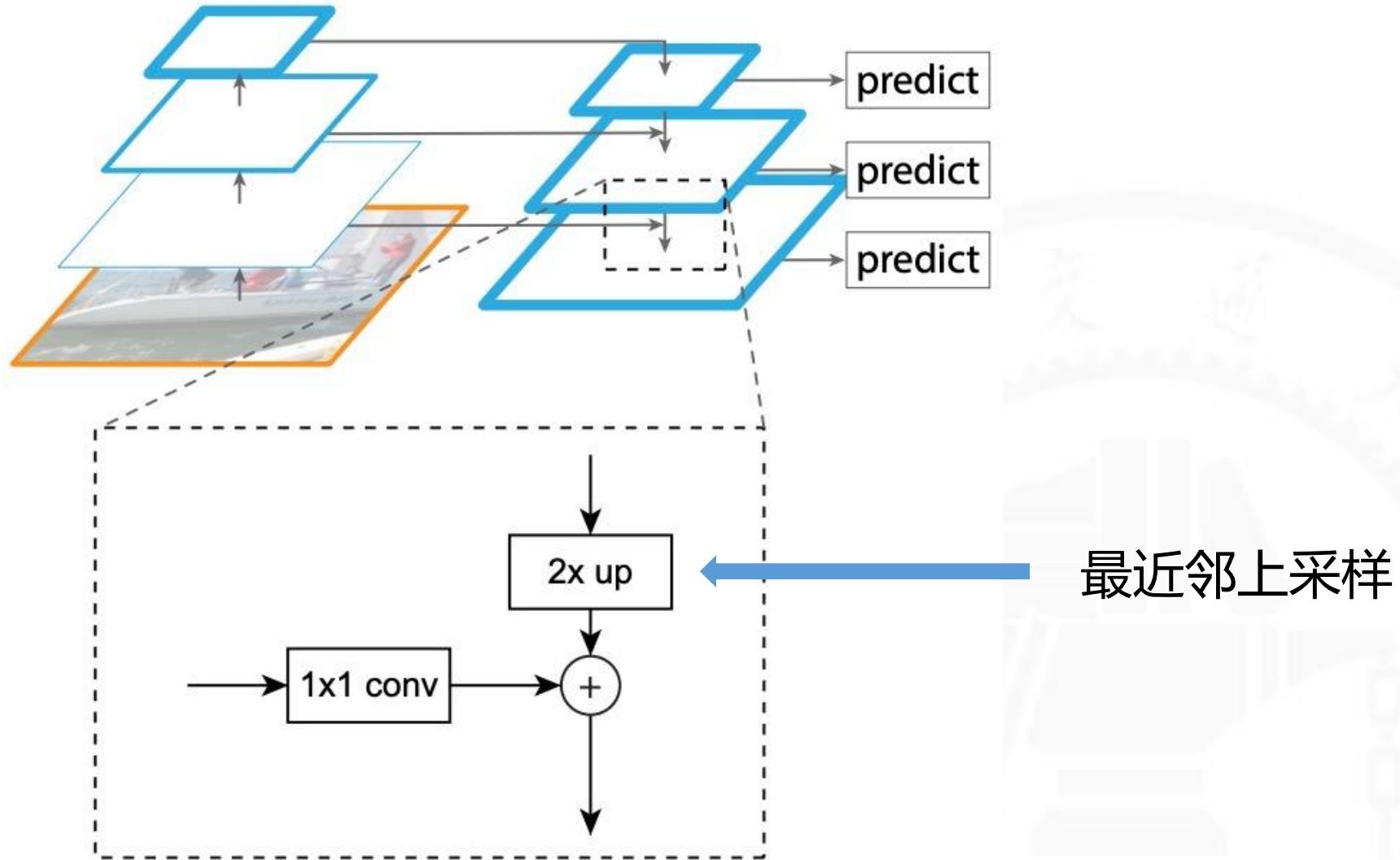
**Output:** C x 6 x 6

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

eg: 在特征金字塔FPN中的最近邻上采样结构

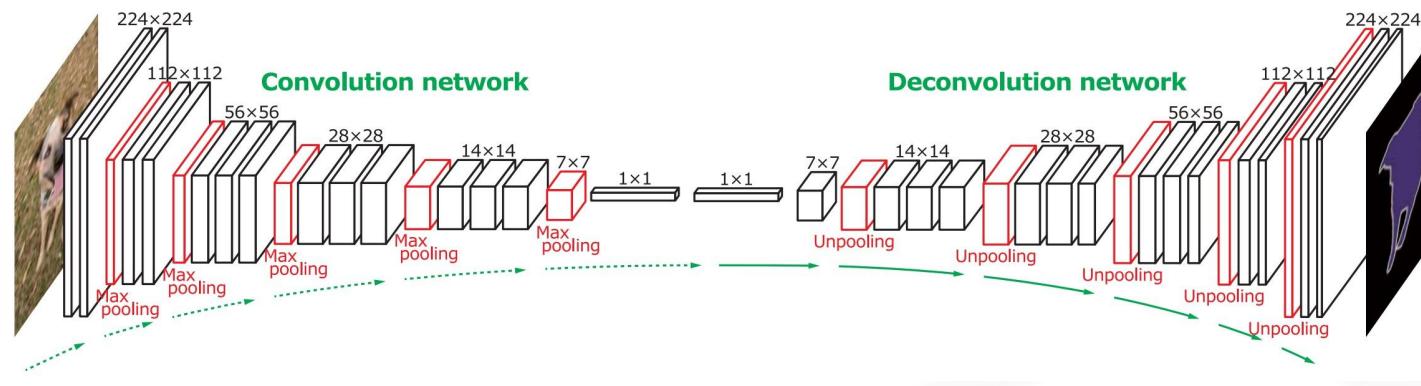


## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### ▲Unpooling



网络的其余部分

在最大池化时存储索引

5	2	4	6
3	7	2	5
8	9	4	2
7	2	3	5

最大池化结果

7	6
9	5

最大池化索引

(1,1)	(0, 3)
(2, 1)	(3, 3)



特征图unpool

2	1
3	7

Unpooled结果

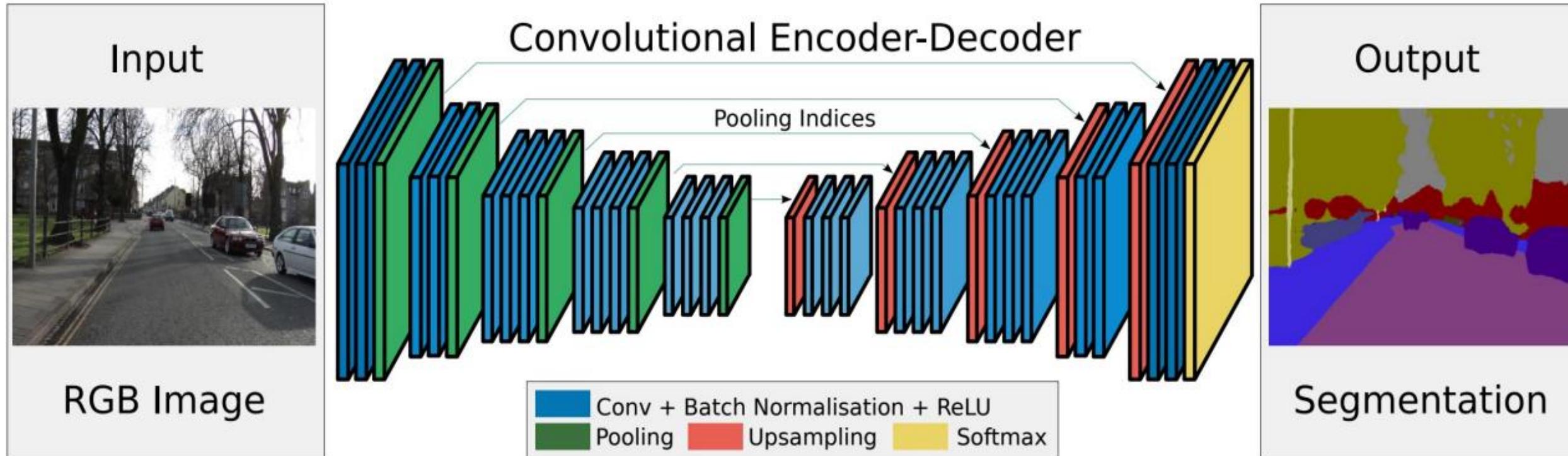
0	0	0	1
0	2	0	0
0	3	0	0
0	0	0	7

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

eg: SegNet中的Unpooling



## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

### ▲ 转置卷积 (Transposed Convolution)

- 学习参数 (卷积核K的参数) 以扩大输入。
- 跨步 (s) 和填充 (p) 的定义：
  - (在输出上使用带有s和p的K) 的结果与输入具有相同的形状
  - 即如果获取输出并使用具有大小为K的滤波器和给定的s和p的标准卷积，则将得到与输入具有相同维度的
- 也称为 “反卷积” (deconvolution) *但这种称呼不太准确，容易引起误解~*
- 计算公式：假设输入特征图的尺寸为  $H_{in} \times W_{in}$ , 通过转置卷积生成的输出特征图的尺寸  $H_{out} \times W_{out}$  可由以下公式计算：

$$H_{out} = (H_{in} - 1) \times S + K - 2P$$

$$W_{out} = (W_{in} - 1) \times S + K - 2P$$

其中：

- $H_{in}, W_{in}$ : 输入特征图的高度和宽度。
- $H_{out}, W_{out}$ : 输出特征图的高度和宽度。
- $S$ : 步长 (stride) 。
- $K$ : 卷积核的大小 (kernel size) 。
- $P$ : 填充 (padding) 。

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

▲ 转置卷积，步长为2

Input

0	1
2	3

Kernel

0	1
2	3

=

0	0		
0	0		

Kernel x 0

+

		0	1
		2	3

Kernel x 1

+

		0	2
		4	6

Kernel x 2

+

		0	3
		6	9

Kernel x 3

=

0	0	0	1
0	0	2	3
0	2	0	3
4	6	6	9

Output

假设输入特征图的尺寸为  $H_{in} \times W_{in}$ ，通过转置卷积生成的输出特征图的尺寸  $H_{out} \times W_{out}$  可由以下公式计算：

$$H_{out} = (H_{in} - 1) \times S + K - 2P$$

$$W_{out} = (W_{in} - 1) \times S + K - 2P$$

其中：

- $H_{in}, W_{in}$ : 输入特征图的高度和宽度。
- $H_{out}, W_{out}$ : 输出特征图的高度和宽度。
- $S$ : 步长 (stride)。
- $K$ : 卷积核的大小 (kernel size)。
- $P$ : 填充 (padding)。

## 2.2.0 基于深度学习的分割方法—知识预备

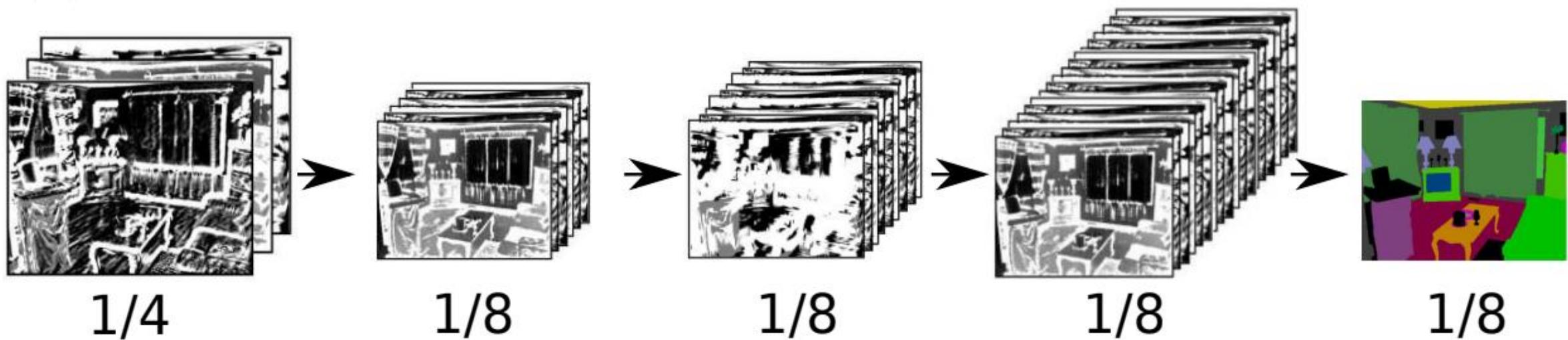


北京交通大学

问题3：如何获取精确的边界位置？

➤ 相关思路：

- 高分辨率层包含更多的位置信息，而较深的低分辨率层有较多的类别信息
- 解码器可以利用高分辨率层中的信息，以帮助改进更精细的边界细节

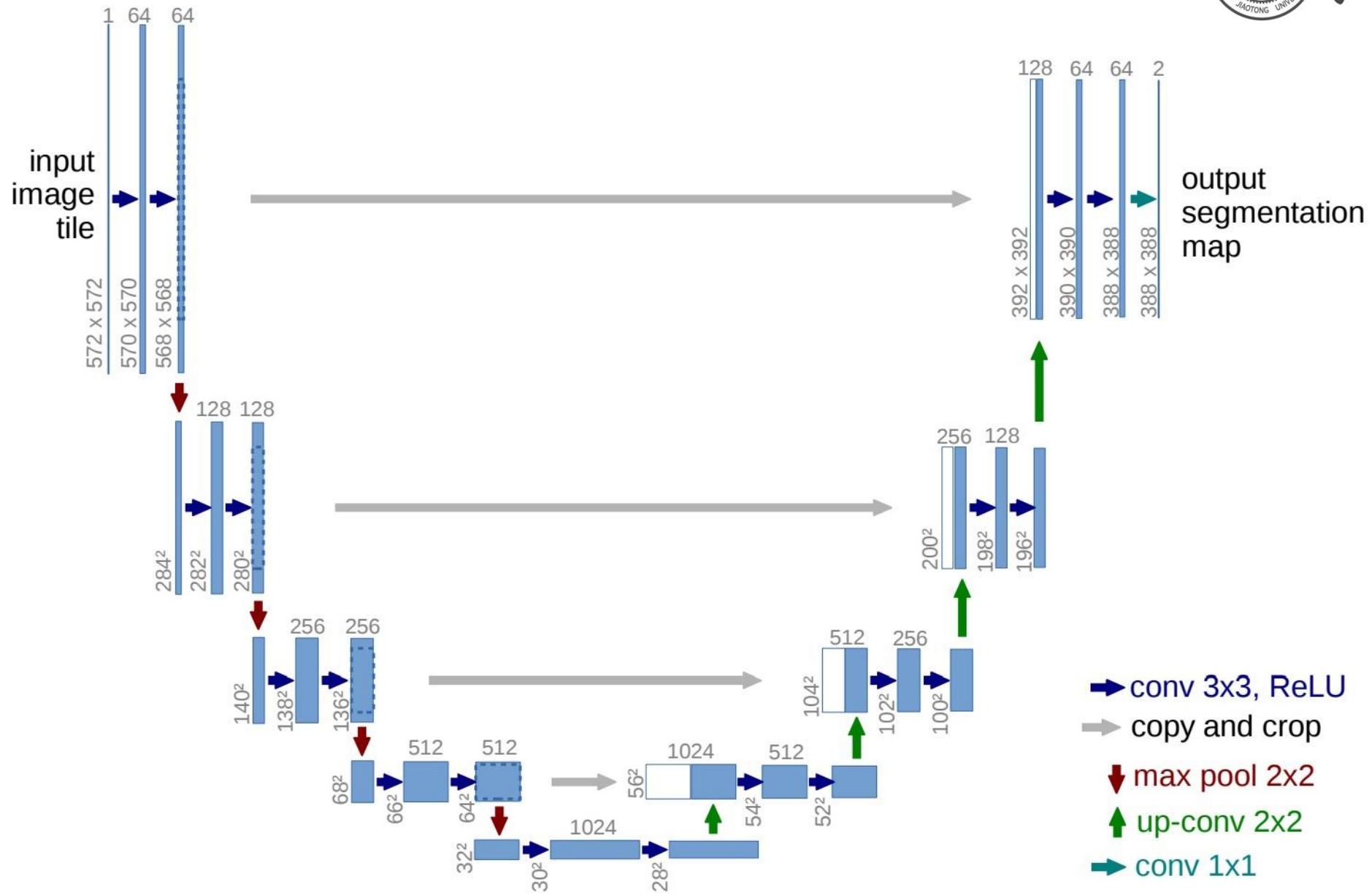


## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

U-Net



## 2.2.0 基于深度学习的分割方法—知识预备

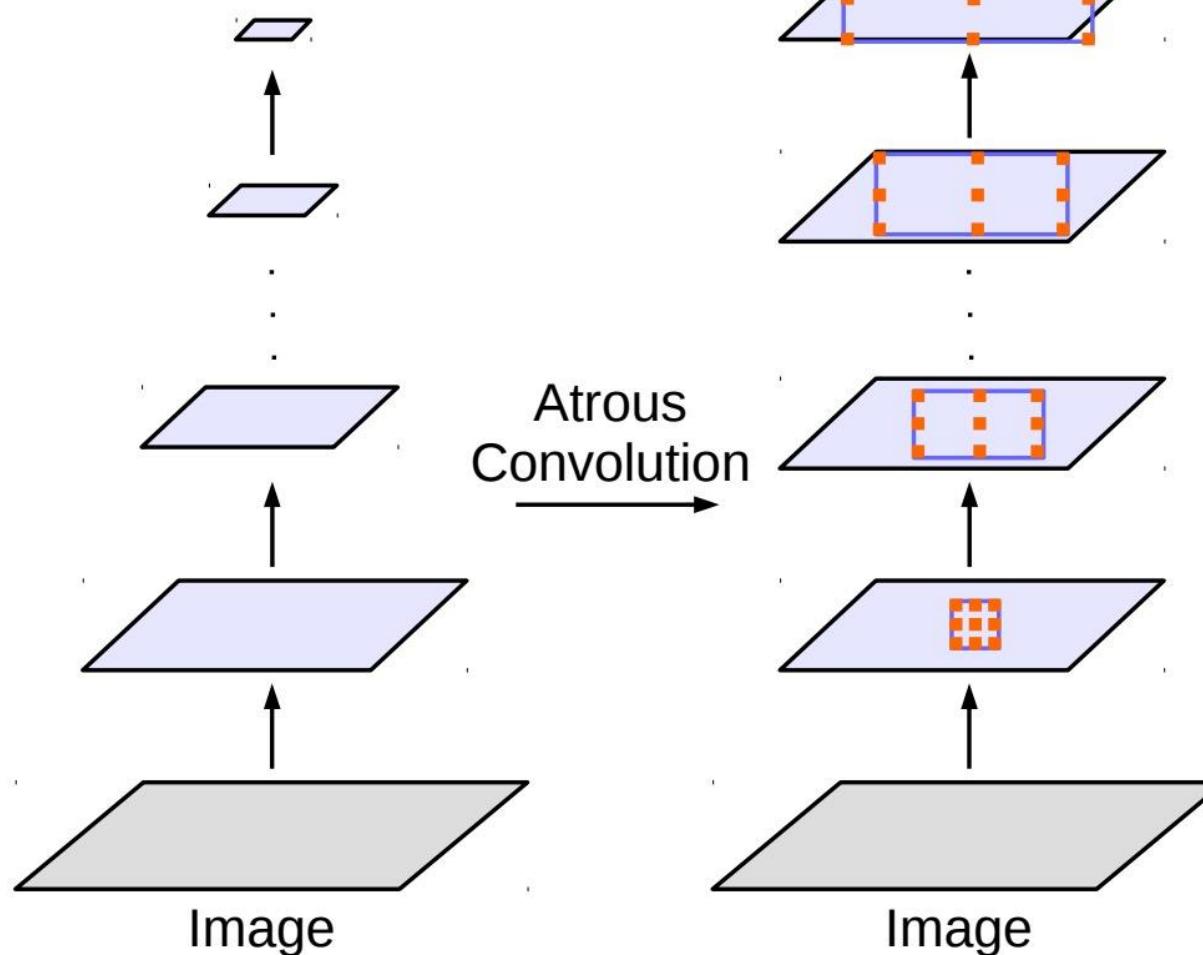


北京交通大学

### ▲ 膨胀卷积 (Dilation Convolution)

- 又称作 空洞卷积 (Atrous Convolution)，通过在卷积核中引入膨胀率 (dilation rate) 来扩展感受野。

Small Resolution



对于输入特征图  $x$ , 卷积核  $w$ , 膨胀卷积的输出  $y[i]$  可表示为:

$$y[i] = \sum_k w[k] \cdot x[i + r \cdot k]$$

其中:

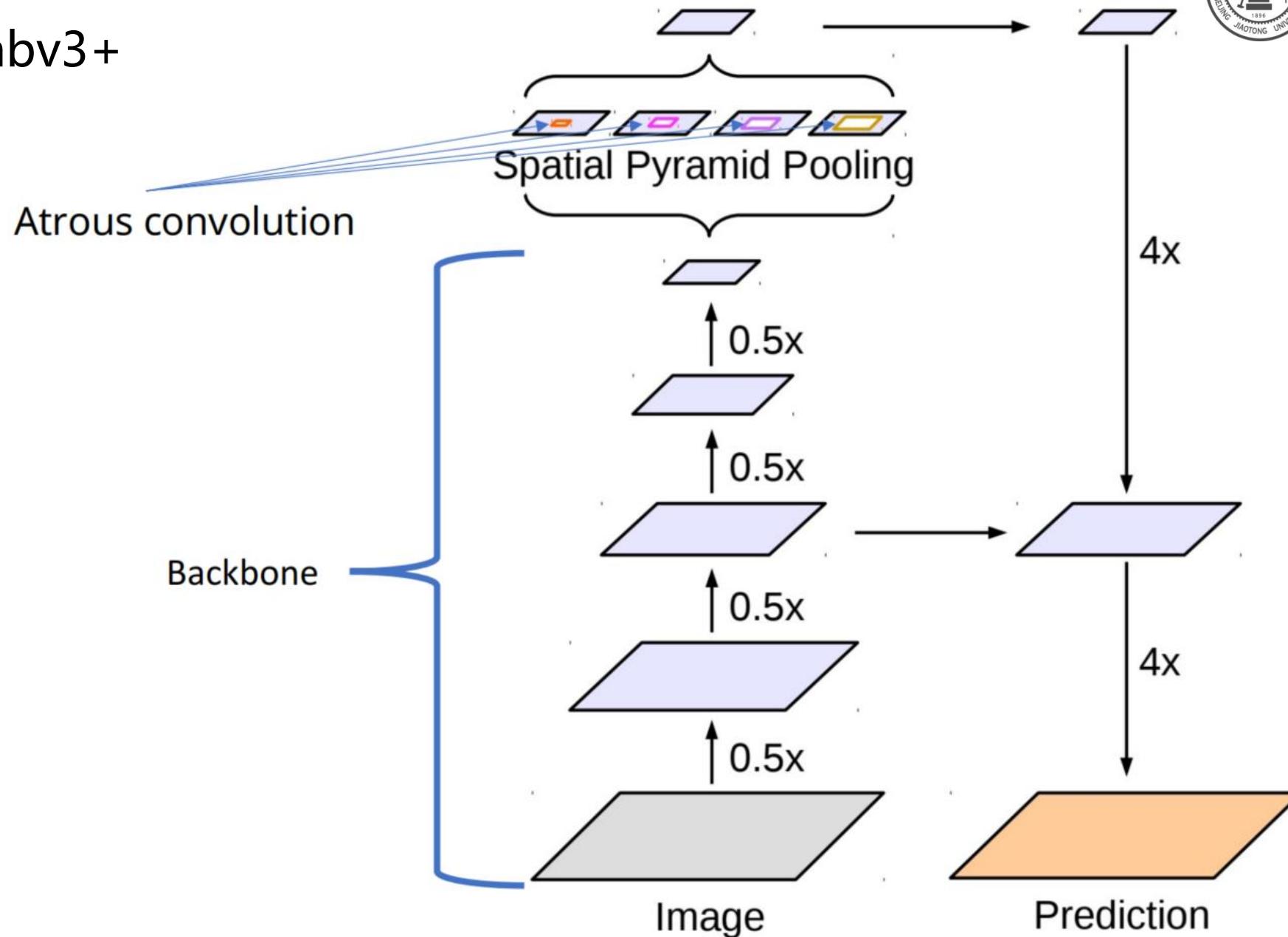
- $r$ : 膨胀率 (Dilation Rate), 表示相邻卷积核元素之间的间隔。
  - $k$ : 卷积核的索引。
  - $x$ : 输入特征图。
- 
- 经典方法DeepLabv3+就是在最后一个特征图上并行应用不同膨胀率的atrous卷积，在上采样和像素级预测之前将其连接起来。

## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

DeepLabv3+

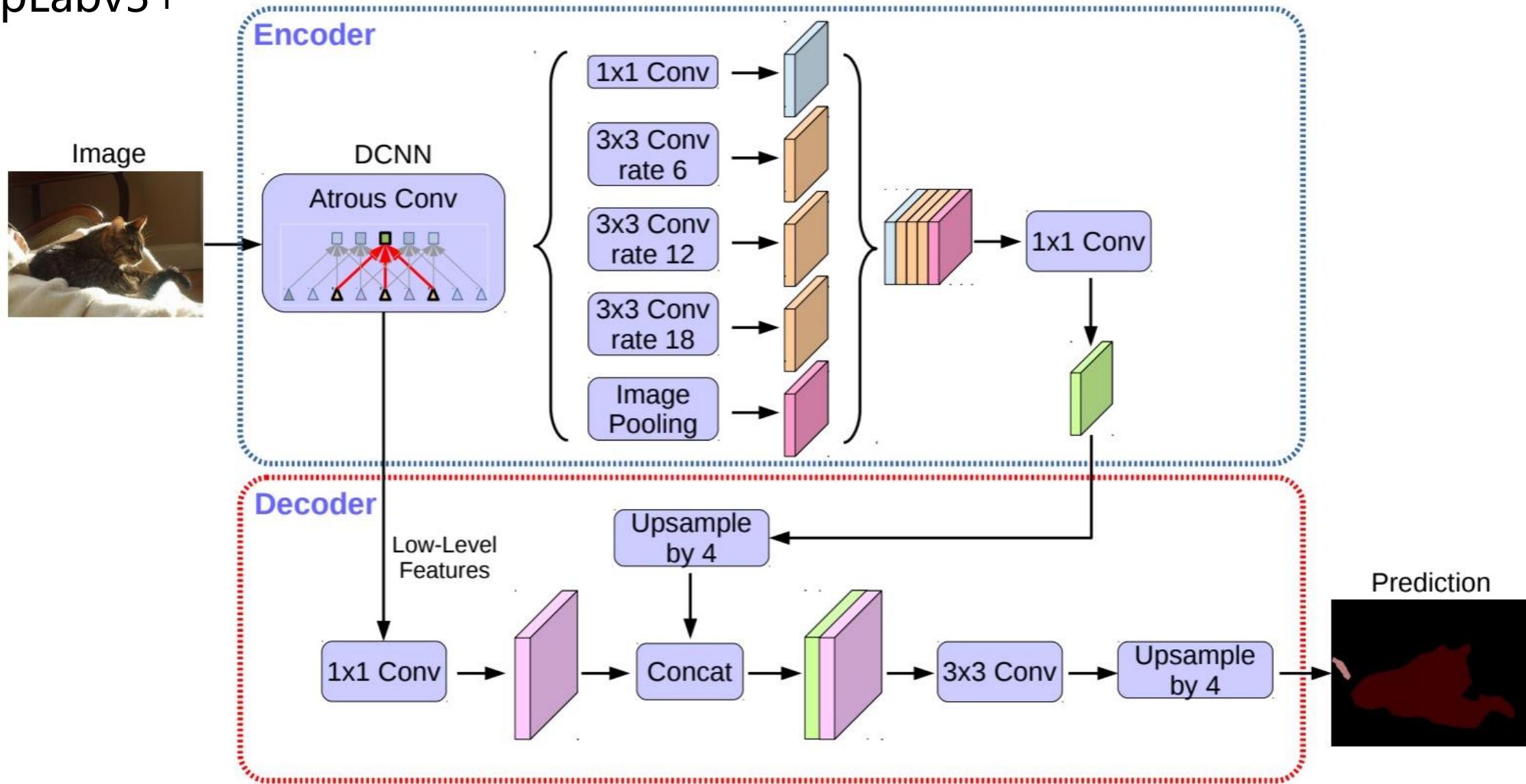


## 2.2.0 基于深度学习的分割方法—知识预备



北京交通大学

DeepLabv3+



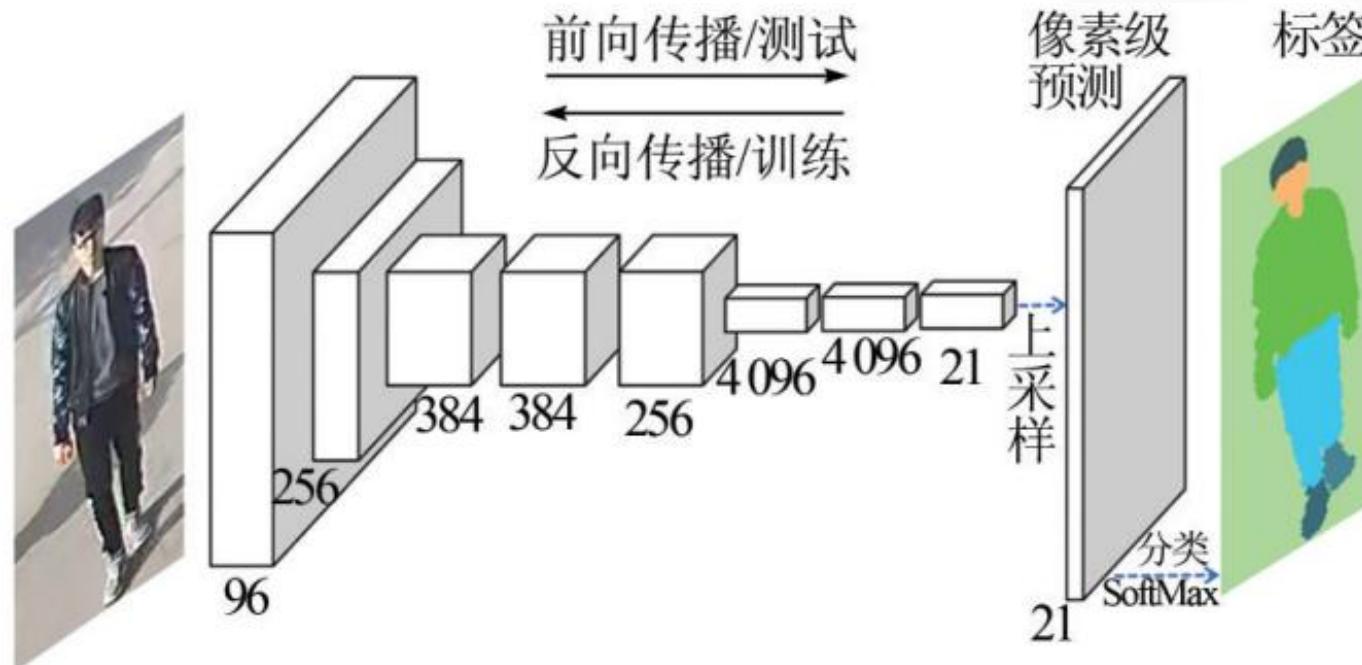
## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ 全卷积网络FCN

- 是深度学习用于语义分割的开山之作，确立了图像语义分割（即对目标进行像素级别分类）的通用网络模型框架。
- 与CNN不同，FCN采用“全卷积”方式，在经过8层卷积处理后，**对特征图进行上采样实现反卷积操作**，然后通过Softmax层进行分类，最后输出分割结果。



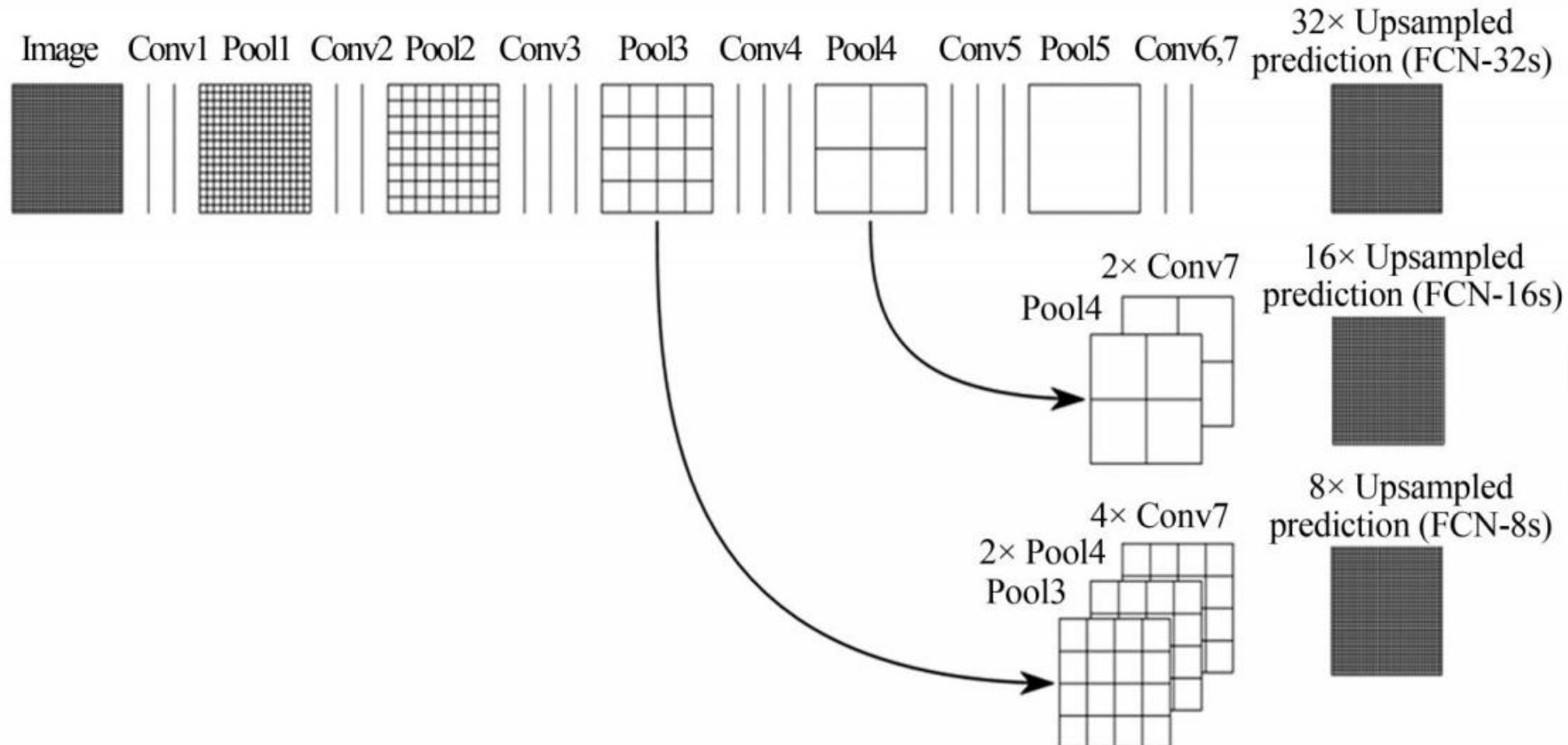
在 FCN 模型中，由于经过多次卷积操作，特征图的尺寸远小于输入图，且丢失了很多底层的图像信息，如果直接进行分类，则会影响分割精度！

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ 全卷积网络FCN



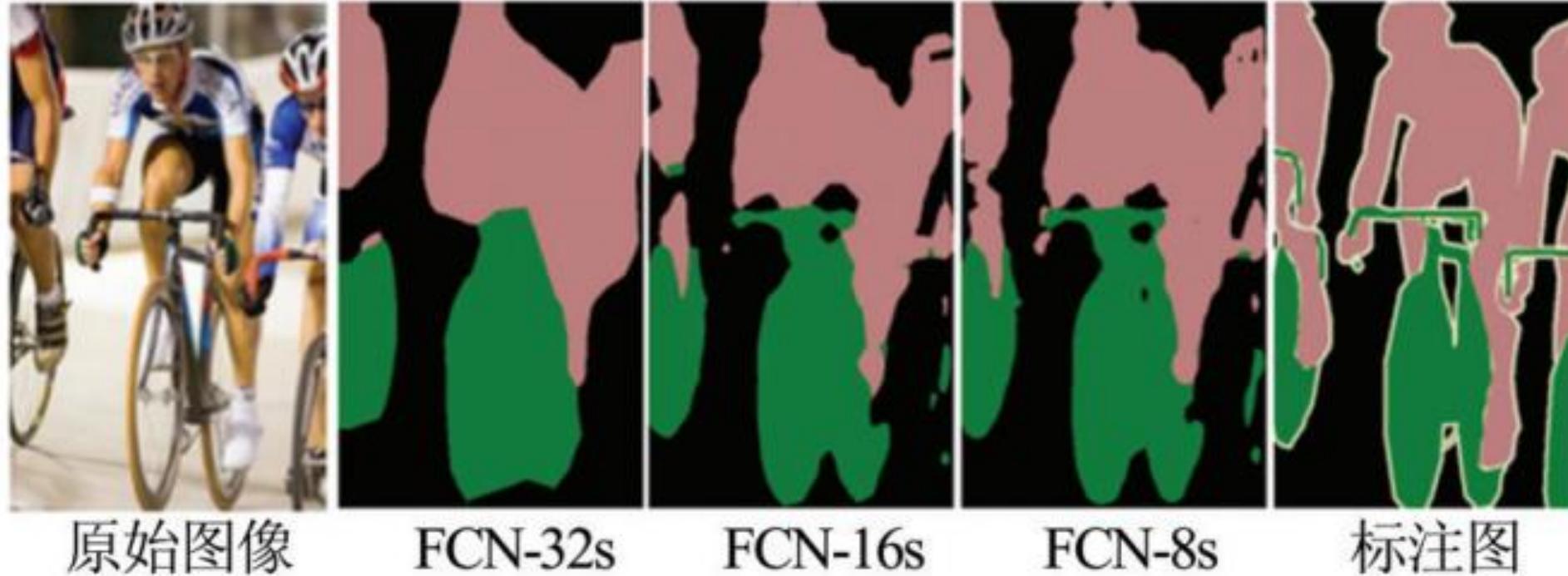
FCN结构图

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ 全卷积网络FCN



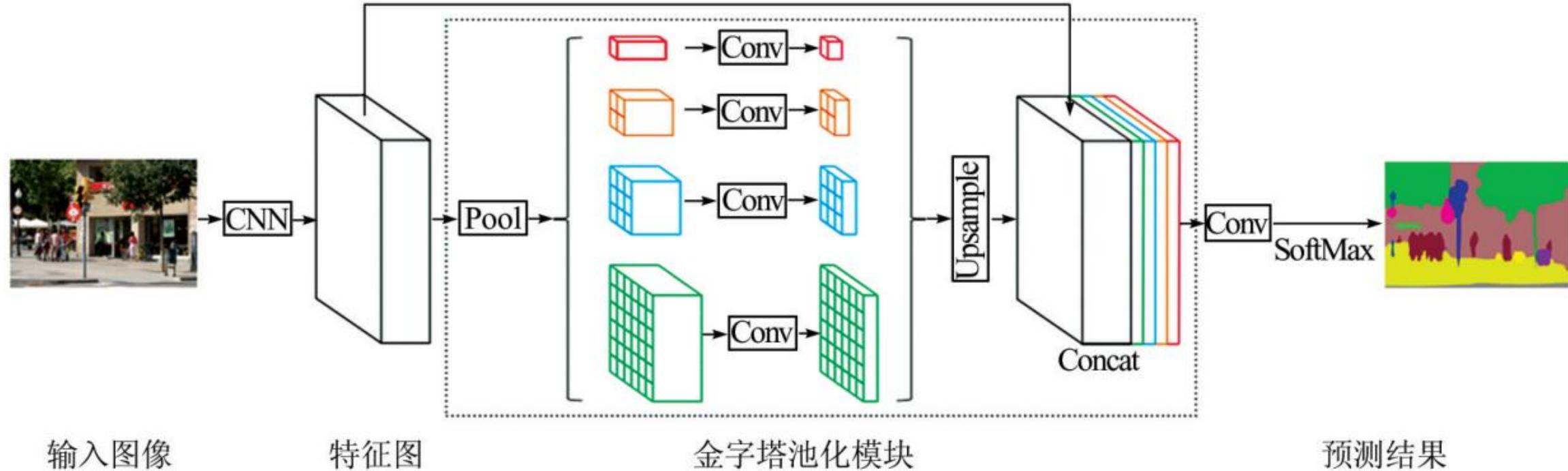
FCN-8s 模型由于整合了更多层的特征信息，相比于FCN-32s和FCN-16s可以分割得到更加清晰的轮廓信息，分割效果相对较好。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ 金字塔场景解析网络 (PSPNet)



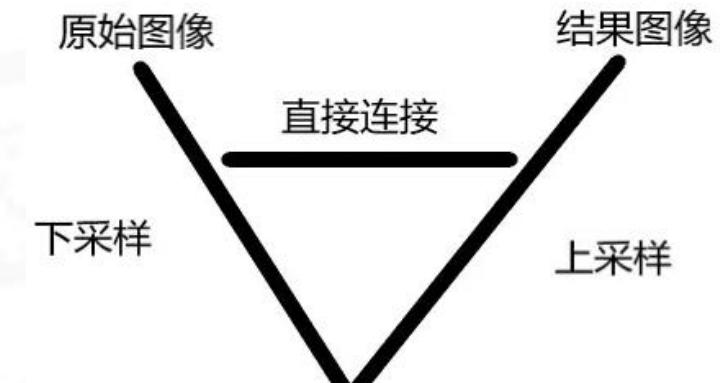
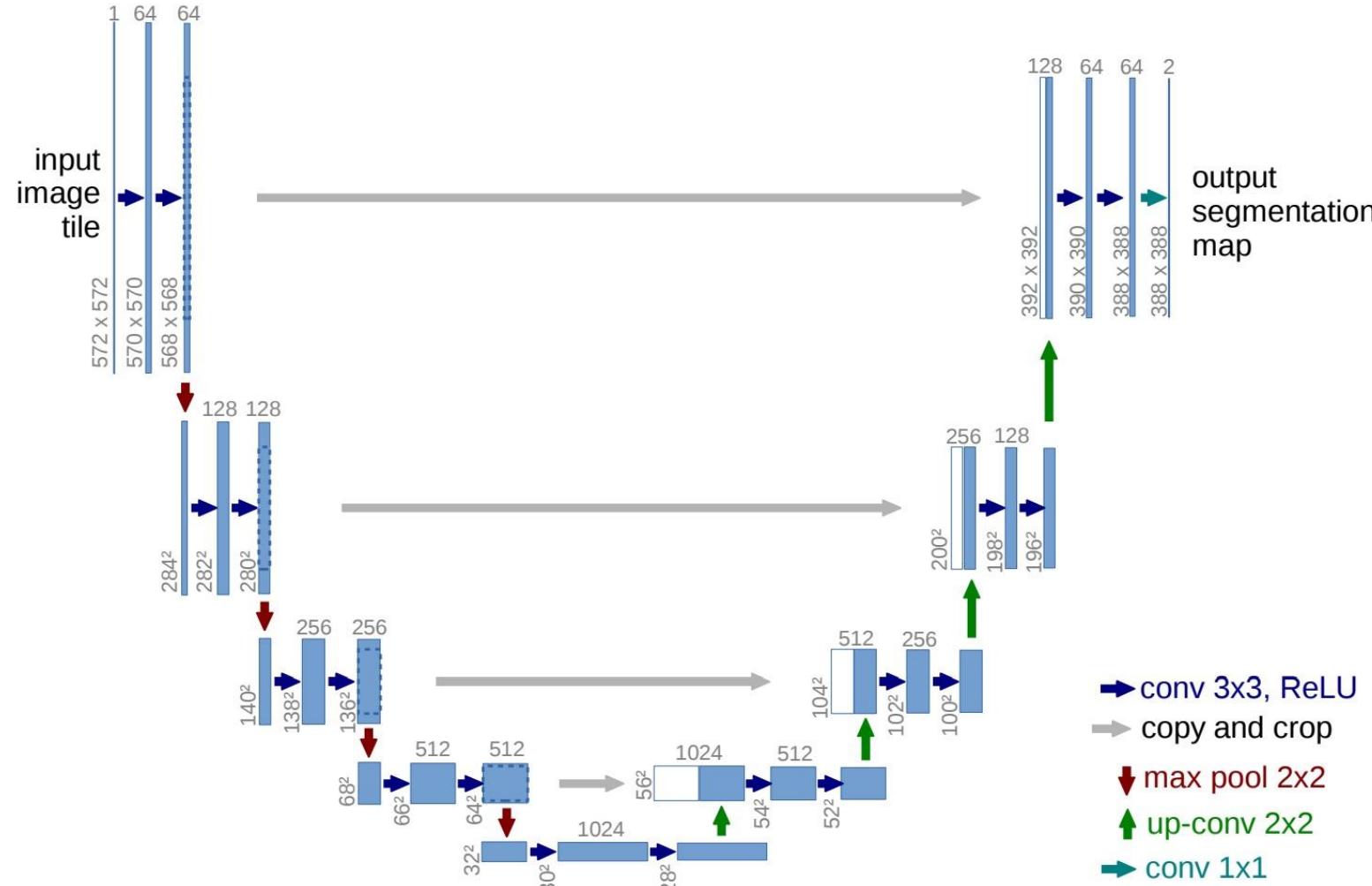
- 给定输入图像，首先使用CNN得到最后一个卷积层的特征图，再用金字塔池化模块收集不同的子区域特征，并进行上采样，然后**拼接融合各子区域特征以形成包含局部和全局上下文信息的特征表征**，最后将得到的特征表征进行卷积和Softmax分类，获得最终的对每个像素的预测结果。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ U-Net



- 下采样
- 上采样
- skip-connection

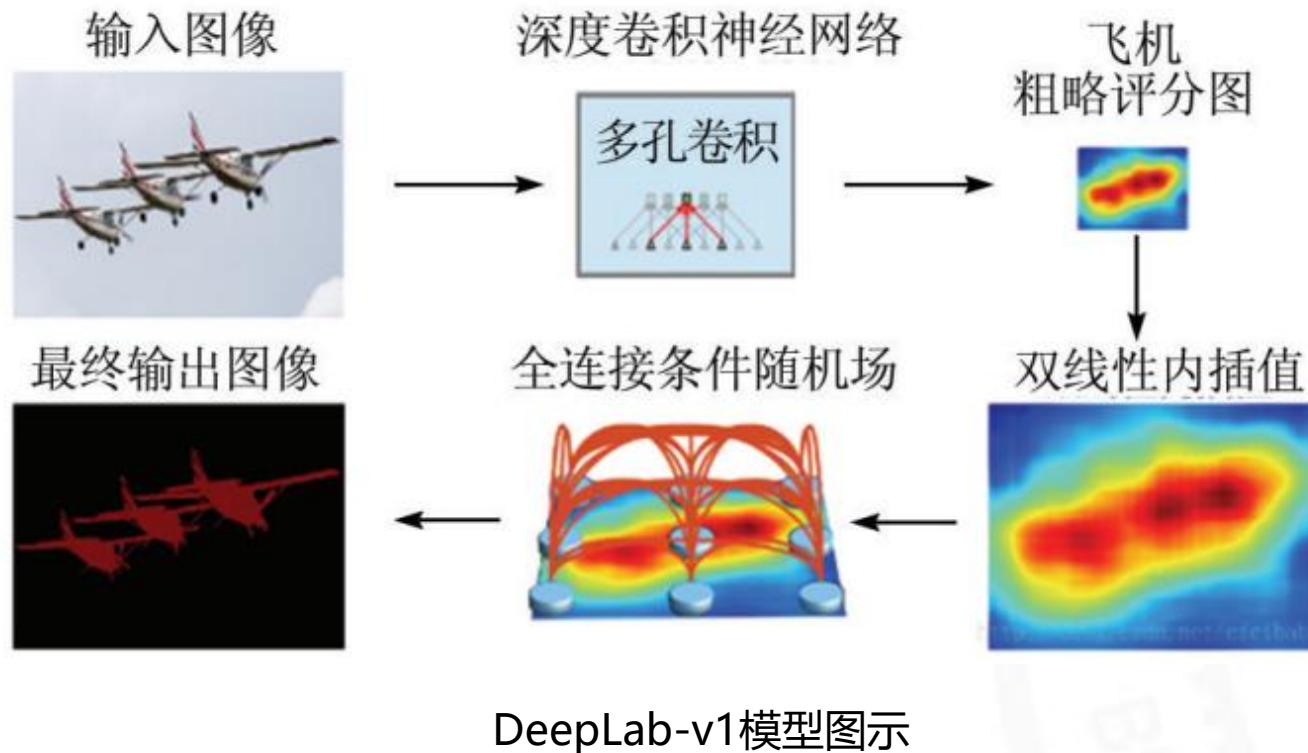
## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ DeepLab系列

- DeepLab系列模型的核心是使用**atrous卷积**，即采用在卷积核里插孔的方式，不仅能在计算特征响应时明确地控制响应的分辨率，还能扩大卷积核的感受野，在不增加参数量和计算量的同时，整合更多的特征信息。



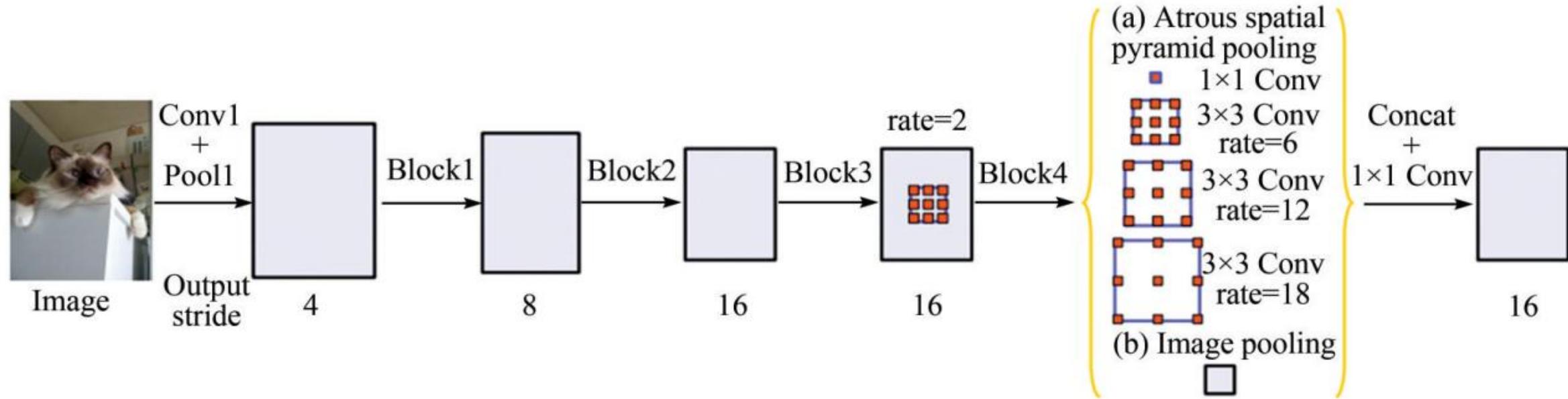
- 输入图像经过带有atrous卷积层的DCNN处理后，得到粗略的评分图，双线性内插值上采样后引入**全连接条件随机场 (CRF)**作为后处理，充分考虑全局信息，对目标边缘像素点进行更准确地分类，排除噪声干扰，从而提升分割精度。
- DeepLabv2在DeepLab模型基础上将 atrous 卷积层扩展为**多孔空间金字塔池化 (ASPP)** 模块，级联多尺度atrous卷积层并进行特征图融合，保留全连接CRF作为后处理。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ DeepLab系列



- DeepLabv3模型：输入图像经过卷积池化后，图像尺寸缩小了 4 倍，再依次经过 3 个 Block 模块 (Block1~ Block3) 进行卷积、ReLU、池化处理，图像依次缩小 8、16、16 倍，然后经过 Block4 处理后进入ASPP模块，ASPP 通过融合不同atrous卷积 ( rate=6、12、18) 处理后，与  $1 \times 1$  卷积层、全局池化层进行整合，得到缩小16倍的特征图，再进行分类预测得到分割图。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ DeepLab系列



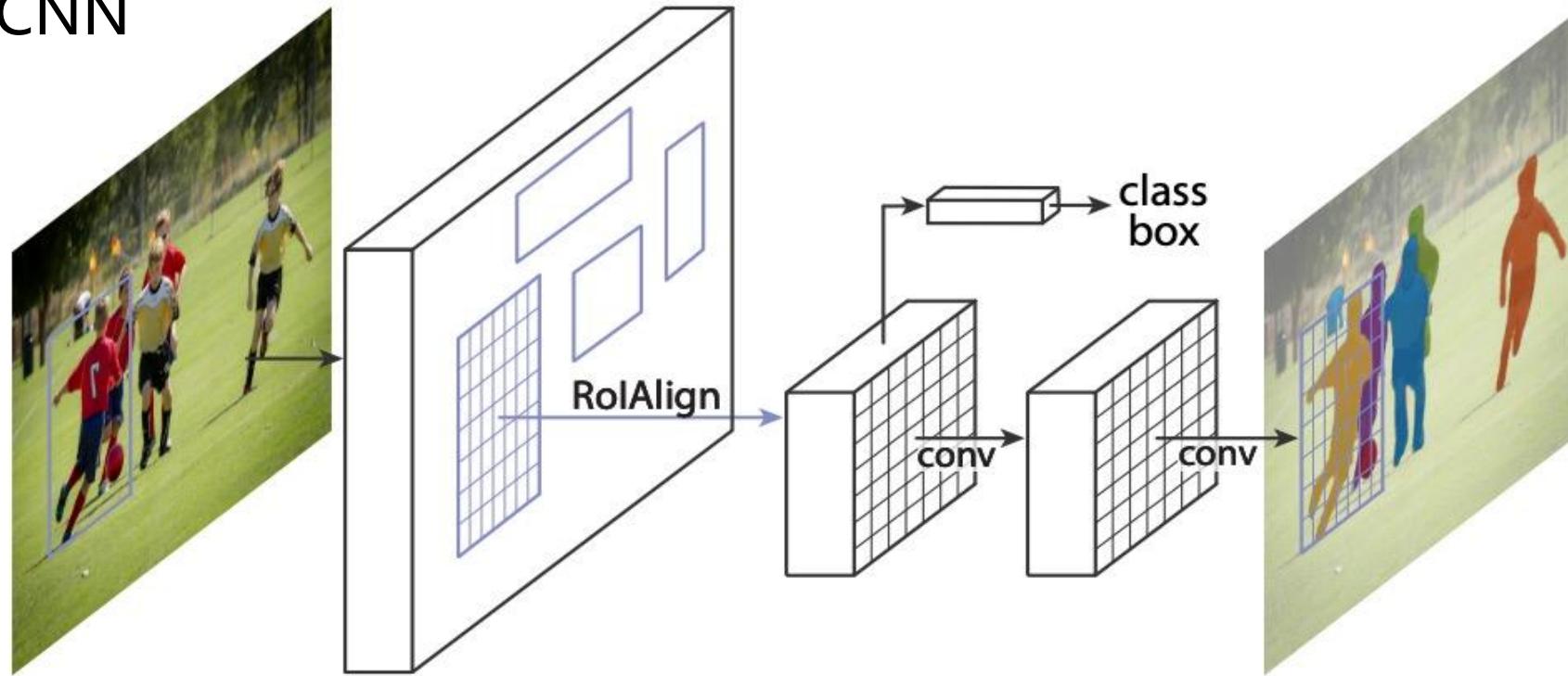
DeepLabv3+分割后的图像中能够明显区分出前景目标和背景，  
目标边缘轮廓清晰，说明其能够实现细粒度的分割。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Mask R-CNN



- 第一阶段：首先用RPN提取出候选目标的边界框，然后对边界框里的RoI进行**RoIAlign**处理，将RoI划分为 $m \times m$ 的子区域；
- 第二阶段：与预测类和边界框回归任务并行，增加了**为每个RoI输出二分类掩码的分支**，相当于用FCN对每个RoI进行分割，以像素到像素的方式预测分割掩码。
- 区别于前面所提到FCN等一系列语义分割模型，Mask R-CNN 在语义分割的基础上实现了实例分割。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Mask R-CNN

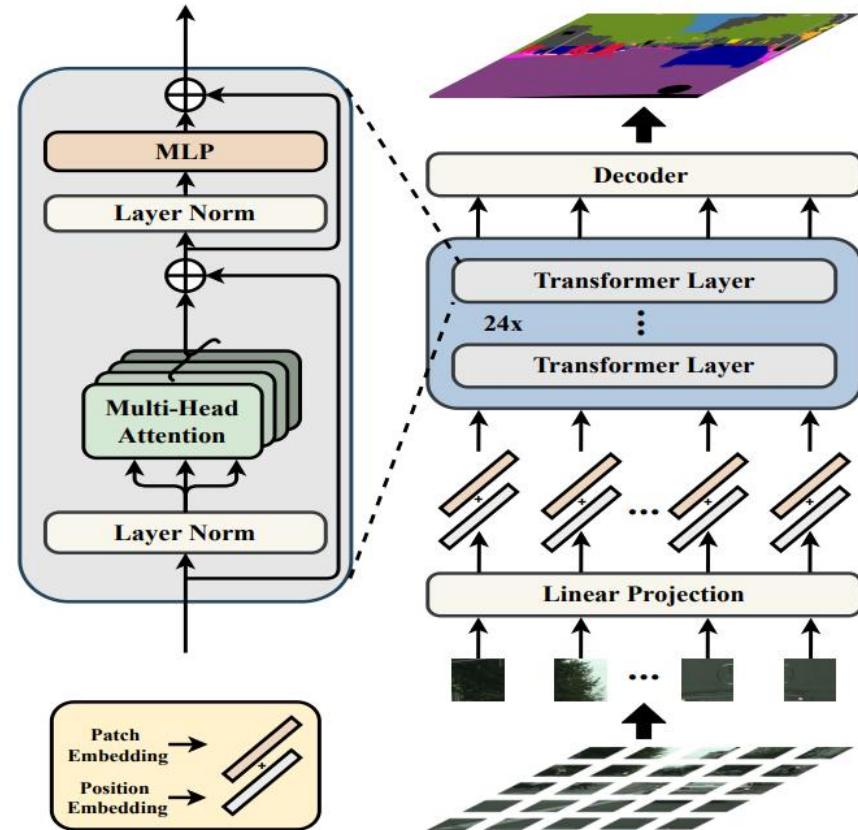


Mask R-CNN在实现对图像的前景目标精准检测定位的同时，实现了像素级实例分割，对同类目标不同个体进行了区分。

## 2.2 基于深度学习的分割方法

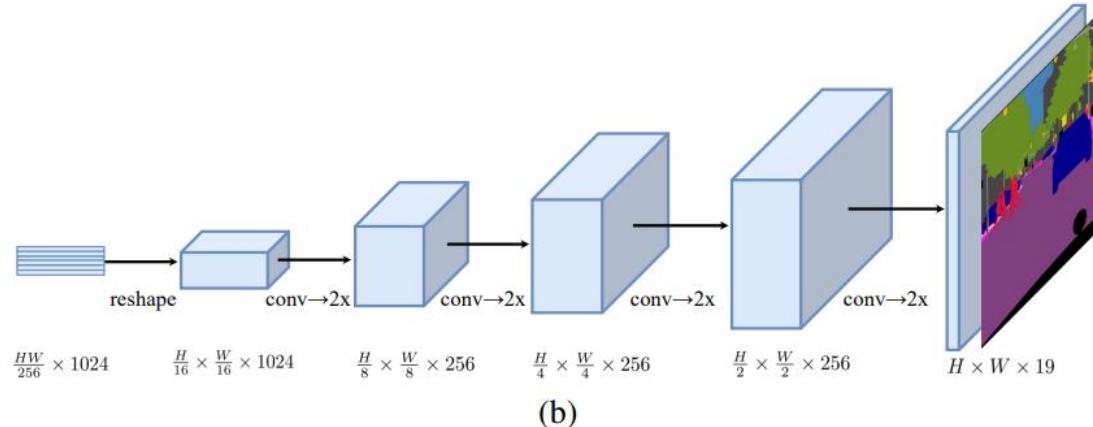


### ▲ SEmgentation TRansformer (SETR)

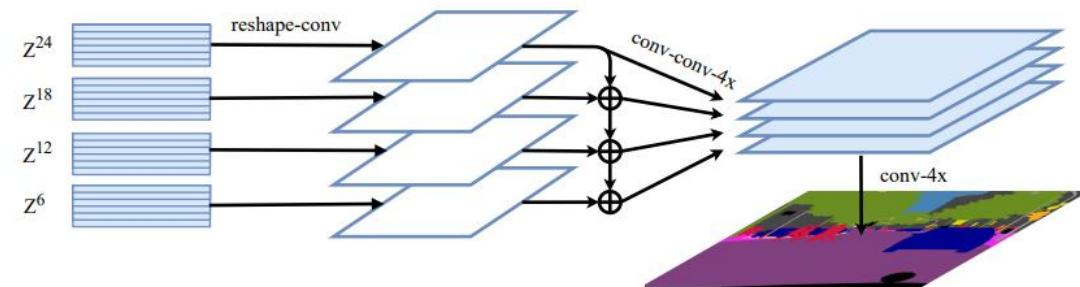


(a)

SETR是尝试将Transformer应用到图像分割中的第一个研究~



(b)



(c)

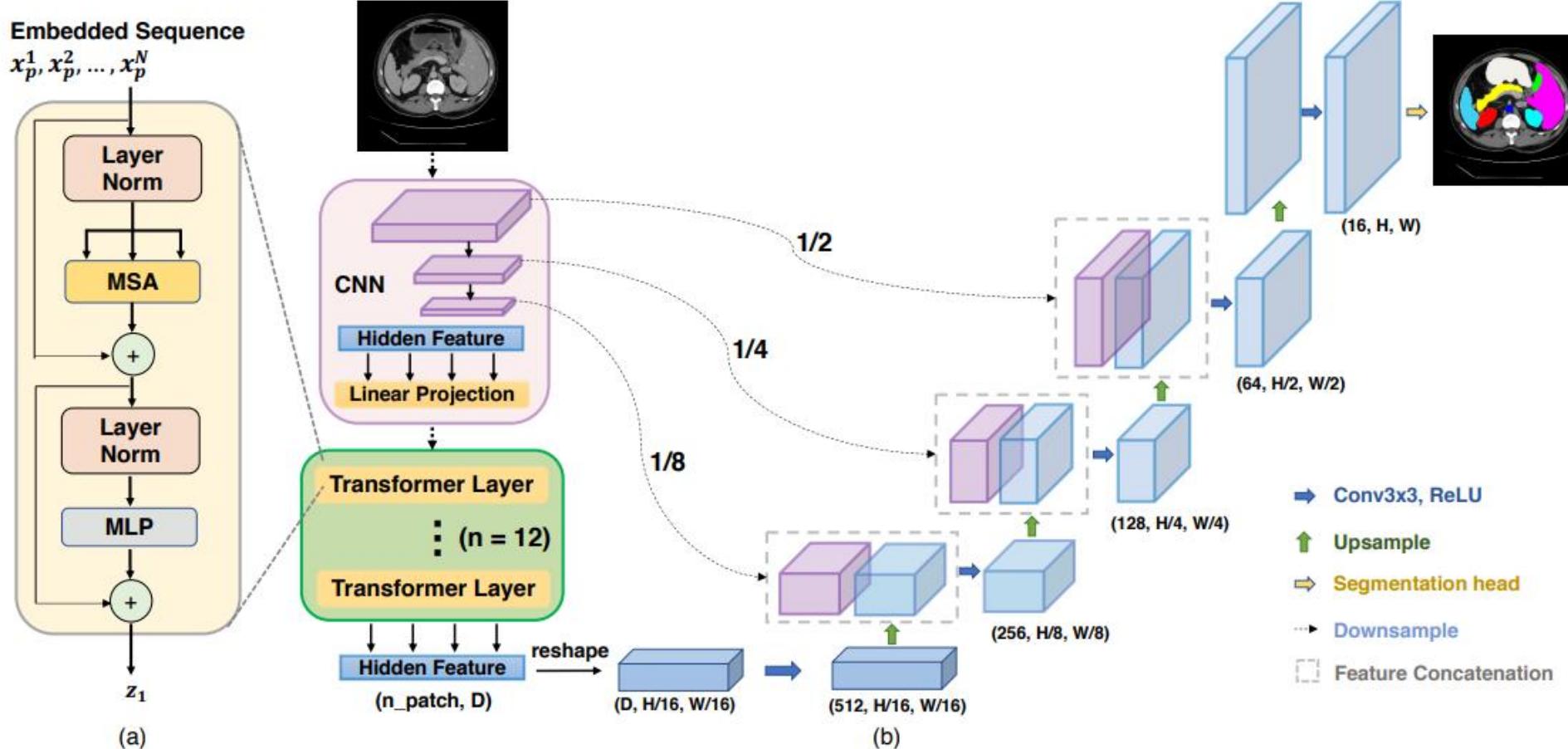
- 核心架构仍是Encoder-Decoder的结构
- 相比于传统基于CNN的编码器结构，SETR采用Transformer进行编码。
- 解码器部分提供了三种设计方式：**原始上采样(NU)**、**渐进式上采样(PUP)**和**多层次特征聚合(MLA)**。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ TransUNet



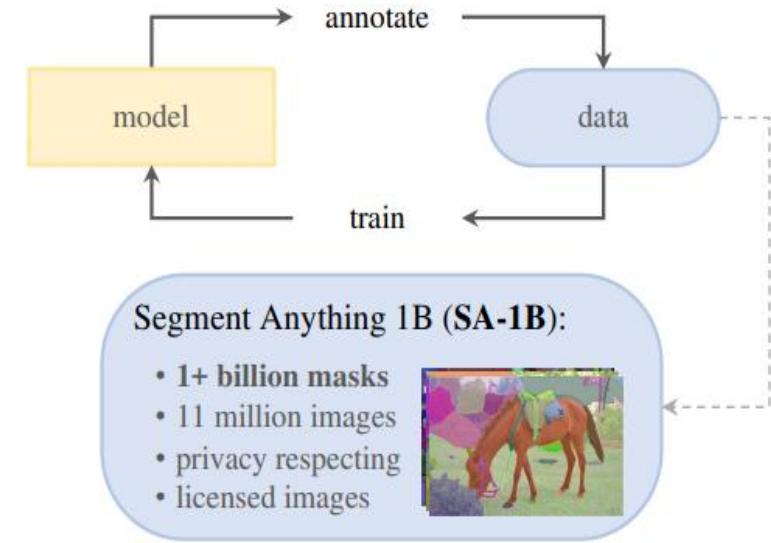
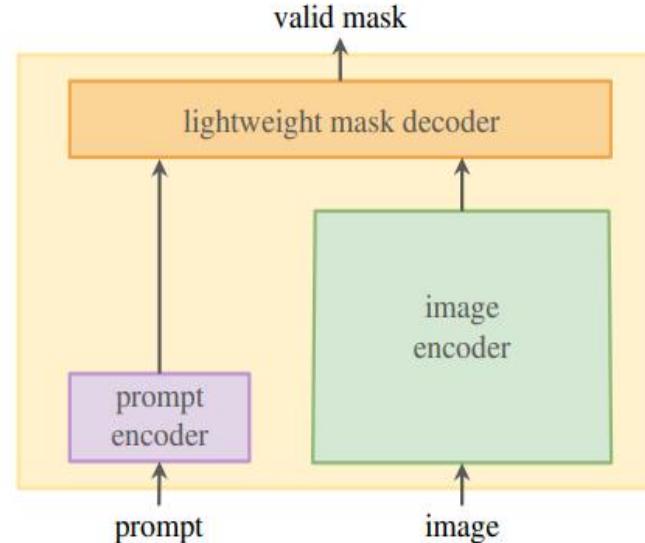
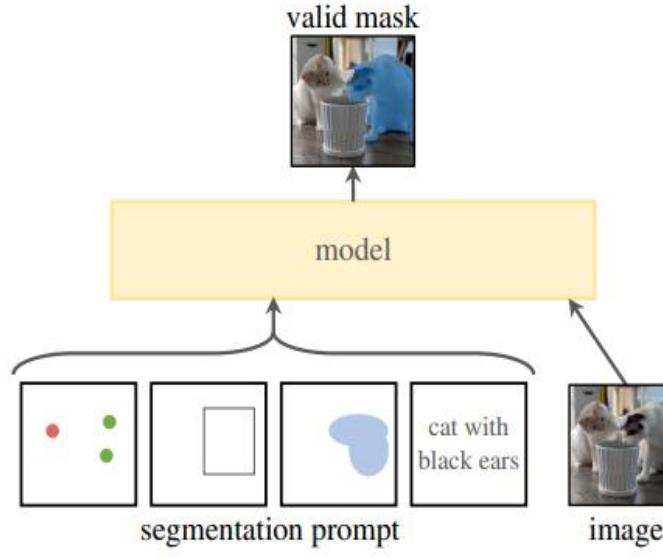
- 编码过程采用ViT中的Hybrid结构，即先用CNN提取特征，再将**特征图切分成指定个数的patch**，然后通过reshape和线性映射后输入到transformer中，进一步学习特征表示。
- **解码过程与U-Net的解码器一致**，即把中间特征图逐步拼接到解码过程产生的特征图上。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Segment Anything(SAM)



- 不同于一般的图像分割算法，SAM旨在构建一个图像分割的基础模型，即使是在训练阶段没有见过的物体类别也能够分割，即**zero-shot**。
- 其核心是**减少图像分割对特定任务建模的专业知识、训练计算量和自定义数据的标注的需求**，实现方法就是采用可提示的方法，根据不同的数据进行训练，并且可以适应特定的任务，类似于自然语言处理模型中如何使用提示。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Segment Anything(SAM)

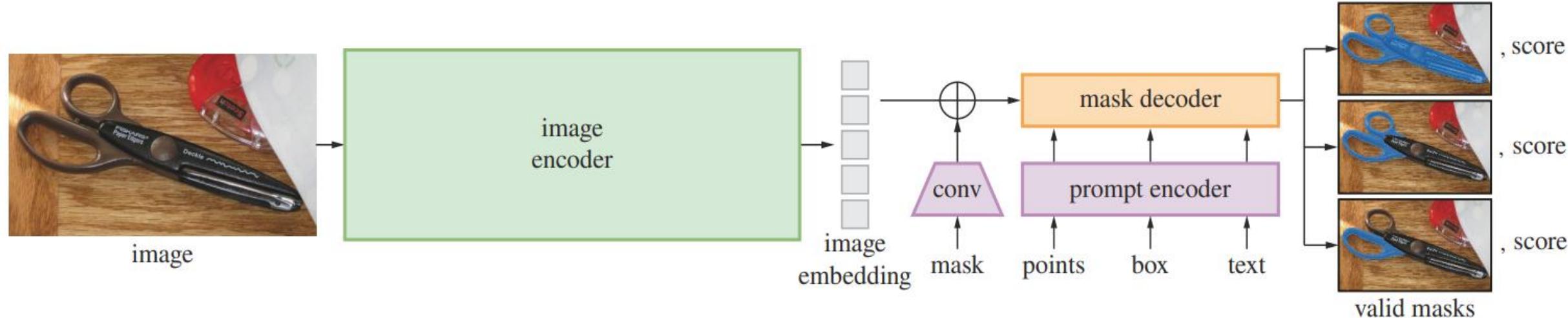
- 实现这一目标取决于三个部分：任务、模型和数据，即三个方面的问题：
  - 什么样的任务可以实现**零样本泛化**？
  - 相应的模型架构是什么？
  - 什么样的数据能够为这个任务和模型提供动力？
- 因此，对应的三个组成部分的设计：
  - **可提示分割任务**，其目标是在给定任何分割提示的情况下返回有效的分割掩码。
  - 模型必须支持灵活的提示，需要**分摊实时计算掩码以允许交互使用**，并且必须具有模糊性意识。  
强大的图像编码器计算图像特征，提示编码器编码提示信息（点、框、掩码和文本等），  
然后将两个信息源组合在一个轻量级掩码解码器中，该解码器预测分割掩码；  
**SAM 具有三个组件：图像编码器、灵活的提示编码器和快速掩码解码器。**
  - 数据引擎分为三个阶段：辅助手动（人工最多，此时模型效果还不行）、半自动（半人工半自动，模型效果持续提升中）和全自动（人工参与最少甚至不需要，都由模型生成掩码）。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Segment Anything(SAM)



- SAM 利用提示分割任务作为预训练目标，**利用提示工程来处理一般的下游分割任务。**
- 为了增强模型适应提示的灵活性并提高其抗干扰的鲁棒性，  
**SAM分为三个部分：图像编码器、提示编码器和掩码解码器。**  
这种划分有效地分配了计算成本，从而产生了具有足够适应性和通用性的分割模型。
- SAM 的优势在于其能够有效地泛化不同的细分任务，这要归功于提示词工程方法。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Segment Anything(SAM)

- 关于SAM提出的数据引擎：由于缺乏足够的公共数据进行训练，研究人员利用训练-标注迭代过程形成数据引擎，同时实现模型训练和数据集构建。
- 具体过程可分为三个阶段。
  - (1) **辅助手动阶段**。专业标注者使用浏览器上的交互式标注工具，结合SAM进行手动标注。SAM首先使用公共数据集进行训练。随着数据逐渐增加，SAM的图像编码器的尺寸也随之增加。在此阶段结束时，收集了430万个掩码和12万张图像。
  - (2) **半自动阶段**。为了增加掩模的多样性并提高模型的性能，研究人员首先通过模型预先填充了掩码，使模型可以做出高置信度的预测。然后他们要求标注者交互式地标注未填充的部分。在此阶段结束时，一张图像平均可以提供72个掩码。
  - (3) **全自动阶段**。由于收集了足够的掩模并引入了模糊感知模型，可以进行SAM的最终训练和SA-1B数据集的获取。即使提示不明确，模糊感知模型也使SAM能够预测有效的掩码。

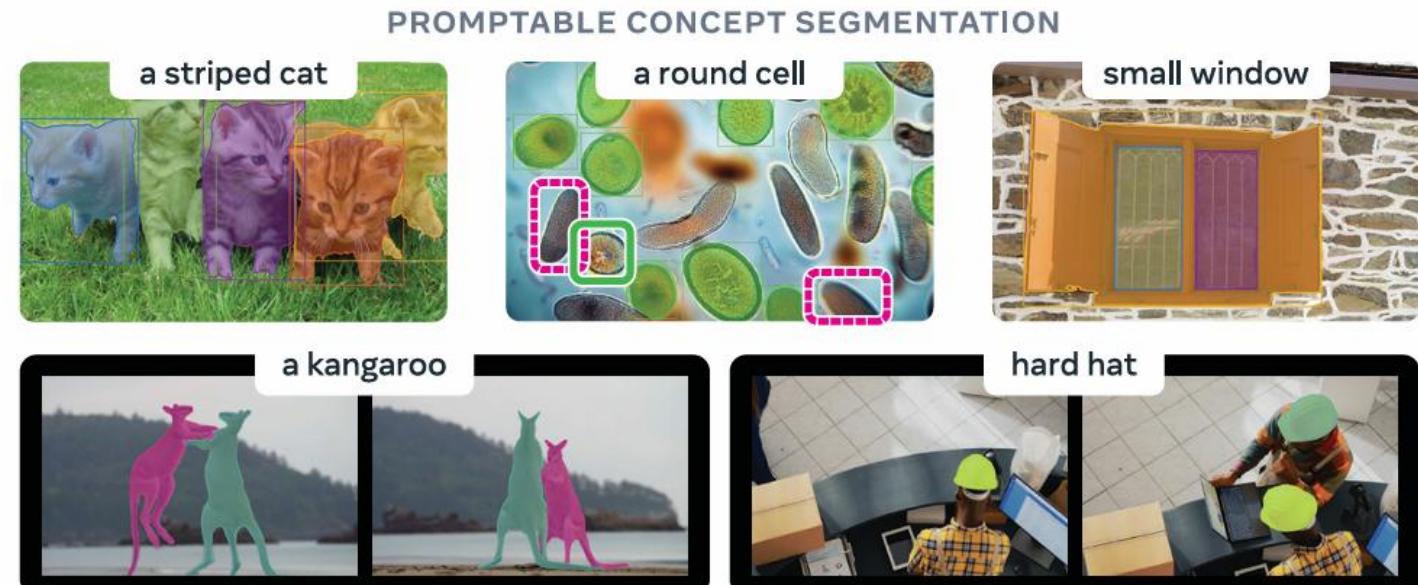
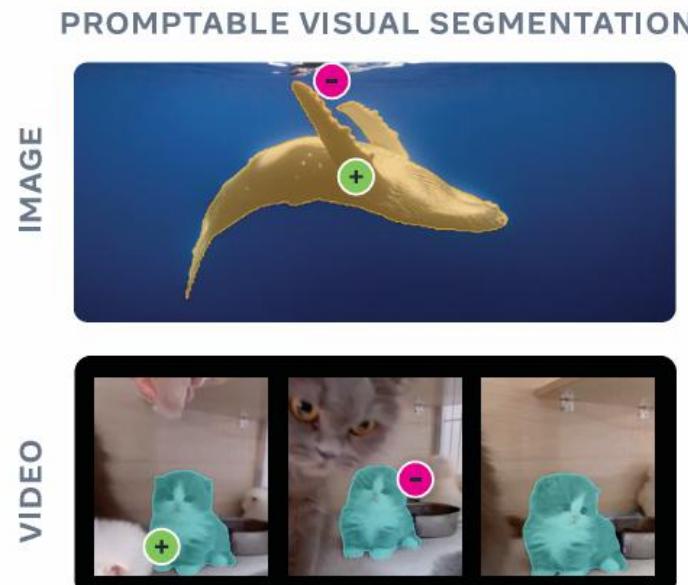
eg: 一些例子：输入一张图，可使用点、框或者点+框进行提示需要分割的物体（默认分割全部物体）

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Segment Anything with Concepts (SAM3)



Prompts: positive or negative points

Prompts: noun phrase and/or positive or negative image exemplar

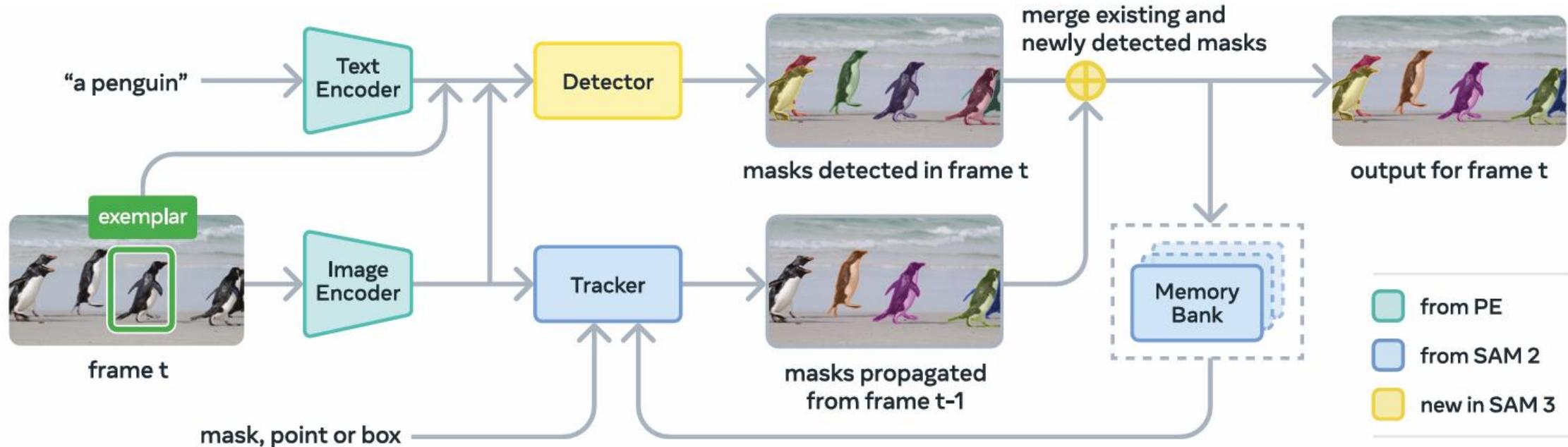
- ICLR2026 《SAM3: Segment Anything with Concepts》, 2025.11已开源。
- <https://github.com/facebookresearch/sam3>
- 在线体验地址: <https://segment-anything.com/>
- 与SAM1和SAM2不同, SAM3使用了: **可提示概念分割** (Promptable Concept Segmentation, PCS) 。
- 给SAM3一张图或者一段不超过30秒的短视频, 然后用一句话、一张示例图, 或者两者结合的方式告诉它你要找什么“概念”, 它能把这个概念下的所有实例都检测、分割、并跟踪出来。

## 2.2 基于深度学习的分割方法



北京交通大学

### ▲ Segment Anything with Concepts (SAM3)



- 多模态编码输入：提示类型：支持文本描述、示例图像、点、框、掩码等多种形式。  
文本编码器和图像编码器将不同的提示信号映射到一个共享的特征表征空间，实现跨模态的概念对齐。
- 检测器：在当前帧中，根据编码后的概念提示，检测并分割出所有与之匹配的实例。  
具有开放词汇能力，无需预定义类别，可直接理解文本概念。
- 跟踪器：将上一帧的实例掩码传播到当前帧，利用时序信息保持视频分割的时间一致性，防止结果闪烁或跳跃。
- 记忆库与合并：记忆库存储已检测实例的历史信息，支持长期的实例管理和关联。  
合并策略将当前帧新检测到的掩码和从上一帧传播来的掩码进行融合与去重，形成完整分割结果。

### 3. 研究数据

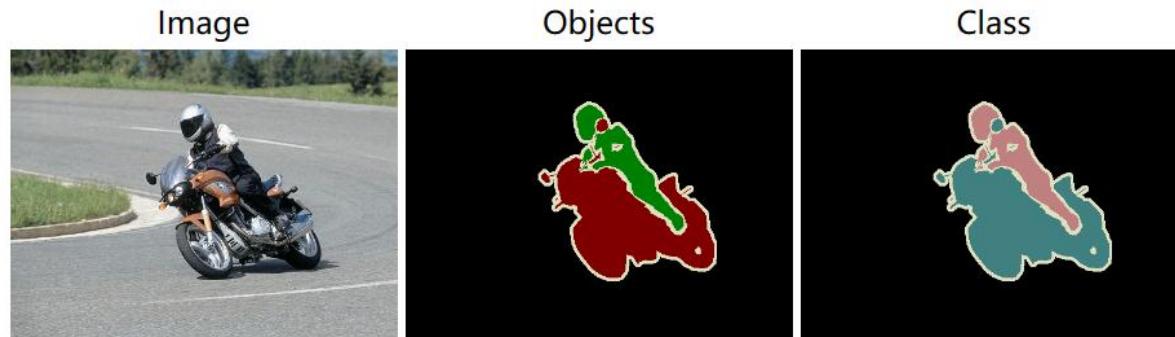


北京交通大学

#### ▲ 常用数据集

##### (1) PASCAL VOC: (<http://host.robots.ox.ac.uk/pascal/VOC/>) ,

共有两个常用版本：VOC2007和VOC2012。2007版由5k张训练图像和12k个被标注的目标组成，2012版由11k张训练图像和27k个带注释的目标组成。从2007年开始，PASCAL VOC每年的数据集都是包含四个大类，总共20个小类。



##### (2) MS COCO: (<http://cocodataset.org/#home>) ,

是一个大规模的图像识别、分割、标注数据集。共包含80种类别。

COCO数据集由微软赞助，其对于图像的标注信息不仅有类别、位置信息，还有对图像的语义文本描述。

近年来逐渐成为了图像语义理解算法性能评价的“标准”数据集。



### 3. 研究数据



北京交通大学

#### ▲ 常用数据集

##### (3) Cityscapes: (<https://www.cityscapes-dataset.com/>) ,

Cityscapes, 即城市景观数据集, 包含一组不同的立体视频序列, 记录在50个不同城市的街道场景。

拥有5000张在城市环境中驾驶场景的图像 (2975train, 500 val, 1525test) 。

它具有19个类别的密集像素标注 (97% coverage) , 其中8个具有实例级分割。



##### (4) ADE20K数据集: (<https://groups.csail.mit.edu/vision/datasets/ADE20K/index.html>) ,

由 27000张图像组成, 这些图像来自于SUN(2010年普林斯顿大学公开的数据集)和Places(2014年MIT公开的数据集) , ADE20K中由超过3000个物体类别, 其中很多图像组成物体的零部件的类别, 以及组成零部件的零部件的类别, 如汽车的零部件, 门, 窗户。ADE20K中还标注了实例的 id, 可用于实例分割。数据中的图像都进行了匿名化处理, 做了人脸和车牌号的模糊, 去除了隐私信息。



### 3. 研究数据

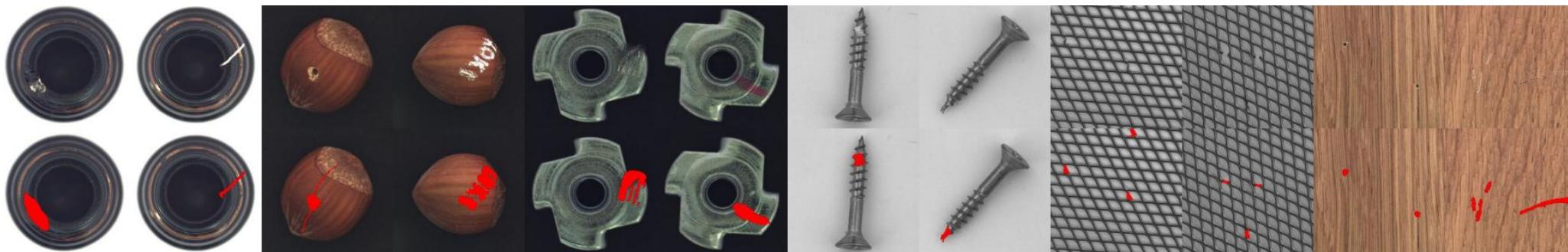


北京交通大学

#### ▲ 常用数据集

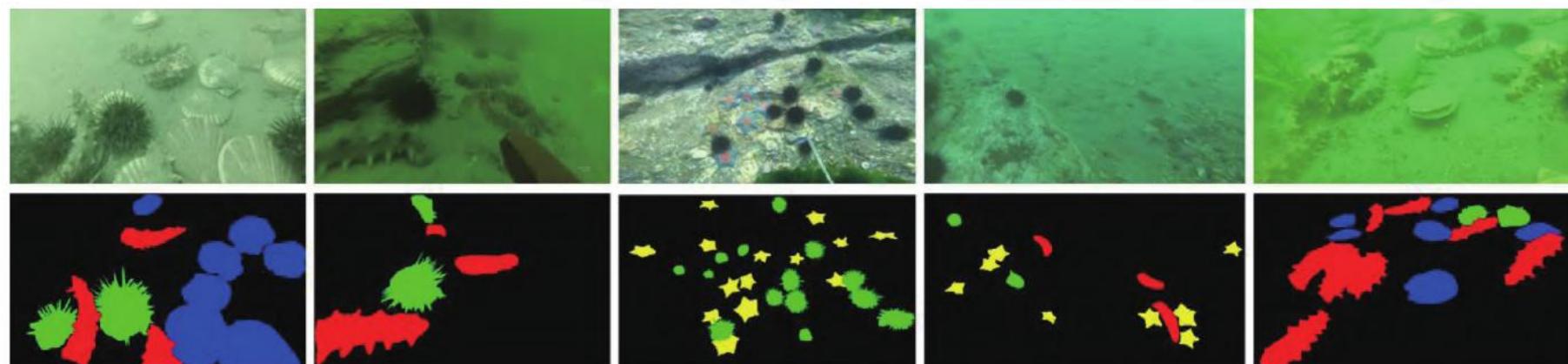
(5) MVTec AD: (<https://www.mvtec.com/company/research/datasets/mvtec-ad>) ,

该数据集包含5354张不同目标和纹理类型的高分辨彩色图像。它包含用于训练的正常（即不包含缺陷）的图像，以及用于测试的异常图像。异常有70种不同类型的缺陷，例如划痕、凹痕、污染和不同结构变化。



(6) DUT-USEG数据集: (<https://github.com/baxiyi/DUT-USEG>) ,

由本文提出了真实场景水下语义分割数据集 DUT-USEG，该数据集包括6617张水下图像，包含了海参、海胆、扇贝和海星4个类别，其中1487张图像具有本文手工添加的语义分割标注和实例分割标注，剩余的5130张图像具有目标检测框标注。



## 4. 评价指标



北京交通大学

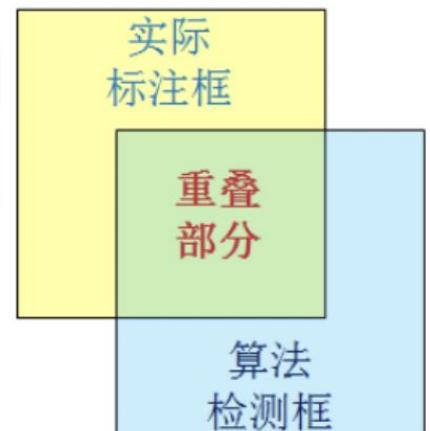
### ▲ 常用的评价指标：(包括但不限于以下这些~)

- PA(Pixel Accuracy) 像素准确率
- CPA(Class Pixel Accuracy) 类别像素准确率
- IoU(Intersection over Union) 交并比
- Dice相似度系数
- mIoU(mean Intersection over Union) 平均交并比
- mPA(mean Pixel Accuracy) 类别平均像素准确率

$$PA = \frac{\sum_{i=0}^k p_{ii}}{\sum_{i=0}^k \sum_{j=0}^k p_{ij}} \times 100\%$$

- **PA** : 表示分类正确的像素占总像素的比例。
- **IoU** : 衡量模型预测的边界框与真实边界框之间的重叠程度。  
通过计算预测框与**真实框的交集与并集之间的比例**以衡量重叠程度。
- **mIoU** :

$$mIoU = \frac{1}{k+1} \sum_{i=0}^k \frac{p_{ii}}{\sum_{j=0}^k p_{ij} + \sum_{j=0}^k p_{ji} - p_{ii}} \times 100\%$$



## 6. 未来展望



北京交通大学

- 改善分割准确性：图像分割方面的研究将继续聚焦于实现更高的分割精度。  
可以通过引入更先进的网络结构、特征提取方法和优化算法，对视觉数据实现更准确、更快速的像素级感知与分割。
- 小/零样本学习：图像分割将更好地适应小样本、零样本和少标注及无标注的场景。  
通过引入元学习、迁移学习和增量学习等技术，能够在具有小样本情况下进行准确分割，实现对新颖类别和新颖场景的快速适应。
- 多模态数据融合：图像分割将更关注于多模态数据融合。  
通过结合图像、语音、文本等多种数据信息，可以更全面理解和识别复杂环境中的目标，提高检测的准确性和鲁棒性。
- 跨域自适应学习：图像分割将更聚焦于跨域场景下的自适应学习。  
通过在不同领域、不同视角和不同感知模态之间进行知识迁移和特征共享，从而实现在新颖场景中的快速适应和精准分割。
- 提高检测实时性：实时性是图像分割中的重要问题，在许多应用中具有重要的价值和需求，如自动驾驶、异常检测、机器人巡航等，对实时性有极高的要求。  
图像分割将更关注模型轻量化、分布式计算、数据预处理与优化等方面。