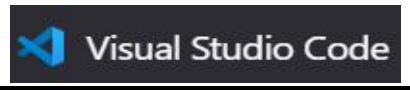
# 计算机视觉（Computer Vision）

# 第12讲 图像分割

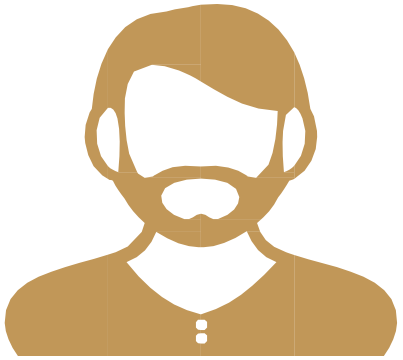
传媒与信息工程学院
欧 新 宇

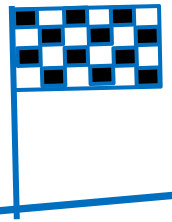


# 基于深度学习的图像分割

# 第13章 计算机视觉

- 图像分割概述
- 图像分割的关键技术和评价指标
- FCN全卷积网络
- U-Net/PSPNet
- DeepLab系列分割模型
- 实例分割和全景分割





Part
01

# 图像分割概述

/ 图像分割概述

/ 图像分割的分类

/ 图像分割的应用

/ 常见的图像分割数据集

# 1.1 图像分割概述

# 1.1 图像分割概述

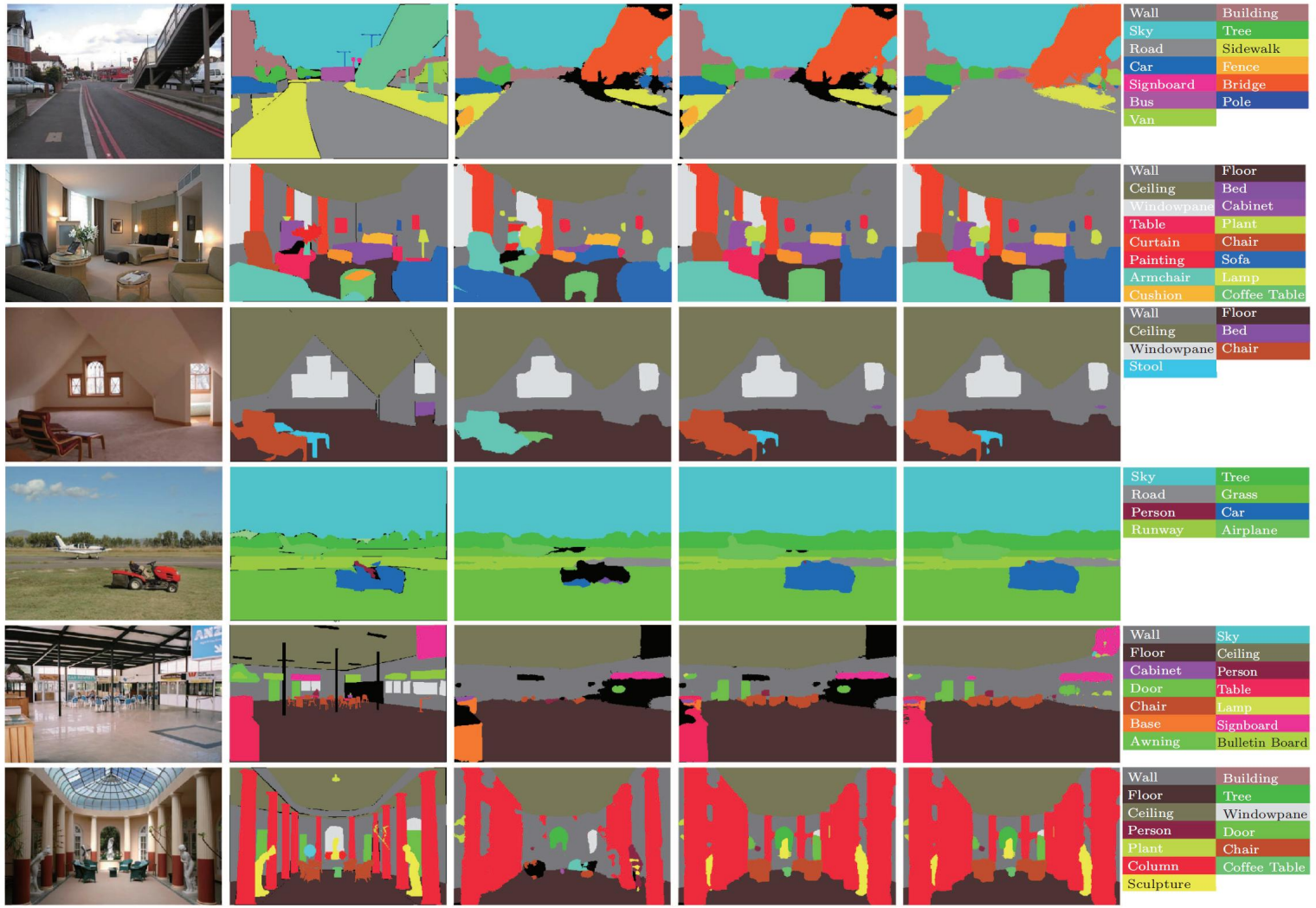
## 什么是图像分割？

# 1.1 图像分割概述

## 什么是图像分割?

# 1.1 图像分割概述

## 什么是图像分割?



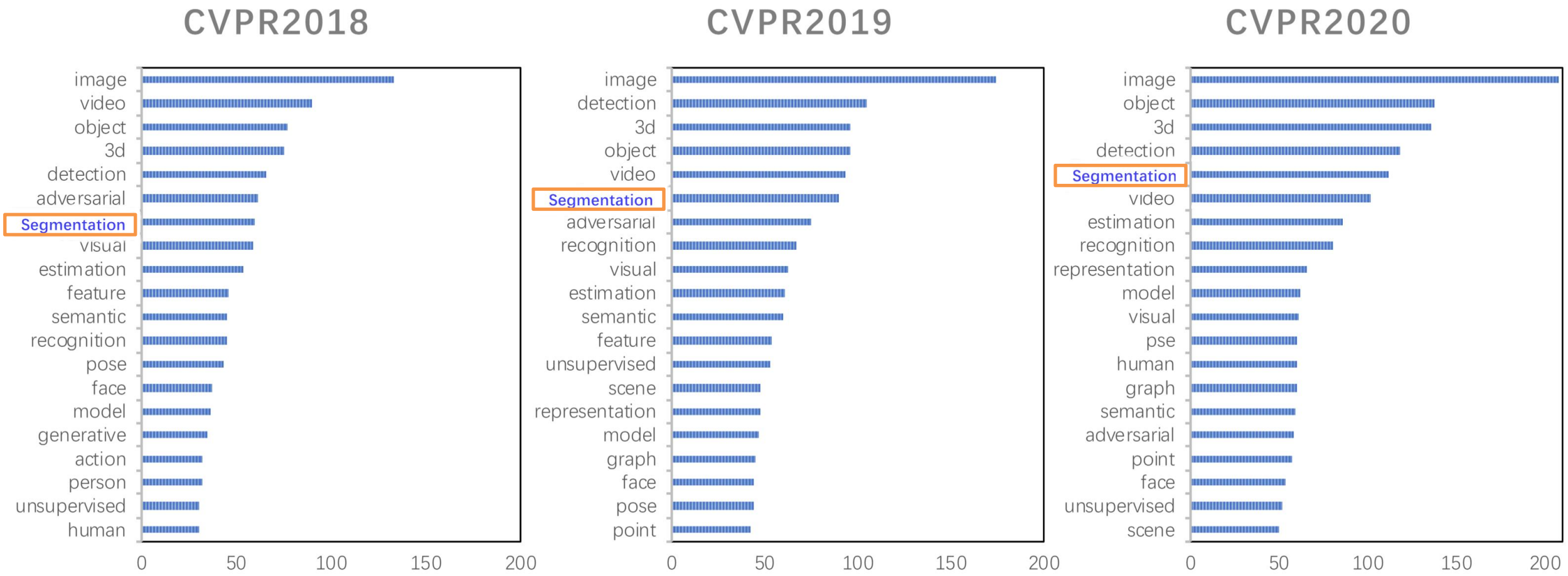

Examples of scene parsing on the *SceneParse150* dataset.

(a) Image
(b) Ground-Truth
(c) Baseline
(d) Our OENet-A
(e) Our OENet-B

Region Enhancement Networks for Scene Parsing. JCST2017

# 1.1 图像分割概述

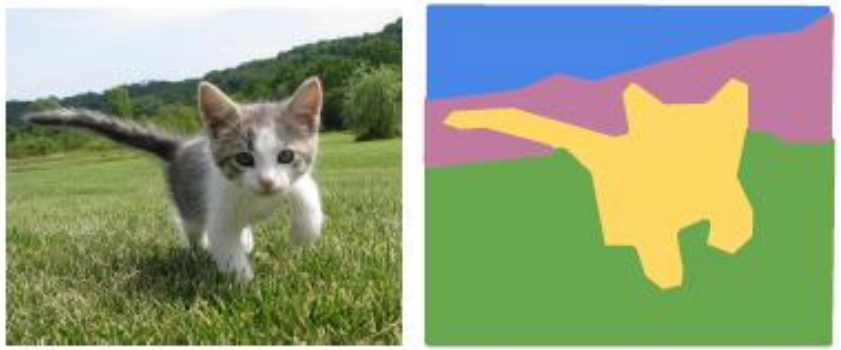
## 图像分割在计算机视觉任务中的位置



# 1.1 图像分割概述

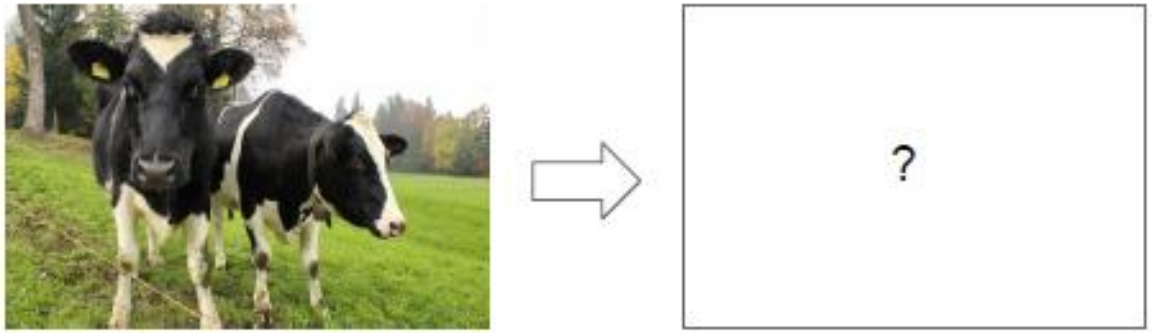
## 问题定义

### 训练阶段




训练数据(x, y)：对于每一个训练图像x，每个像素都使用语义类别进行标注

### 推理阶段

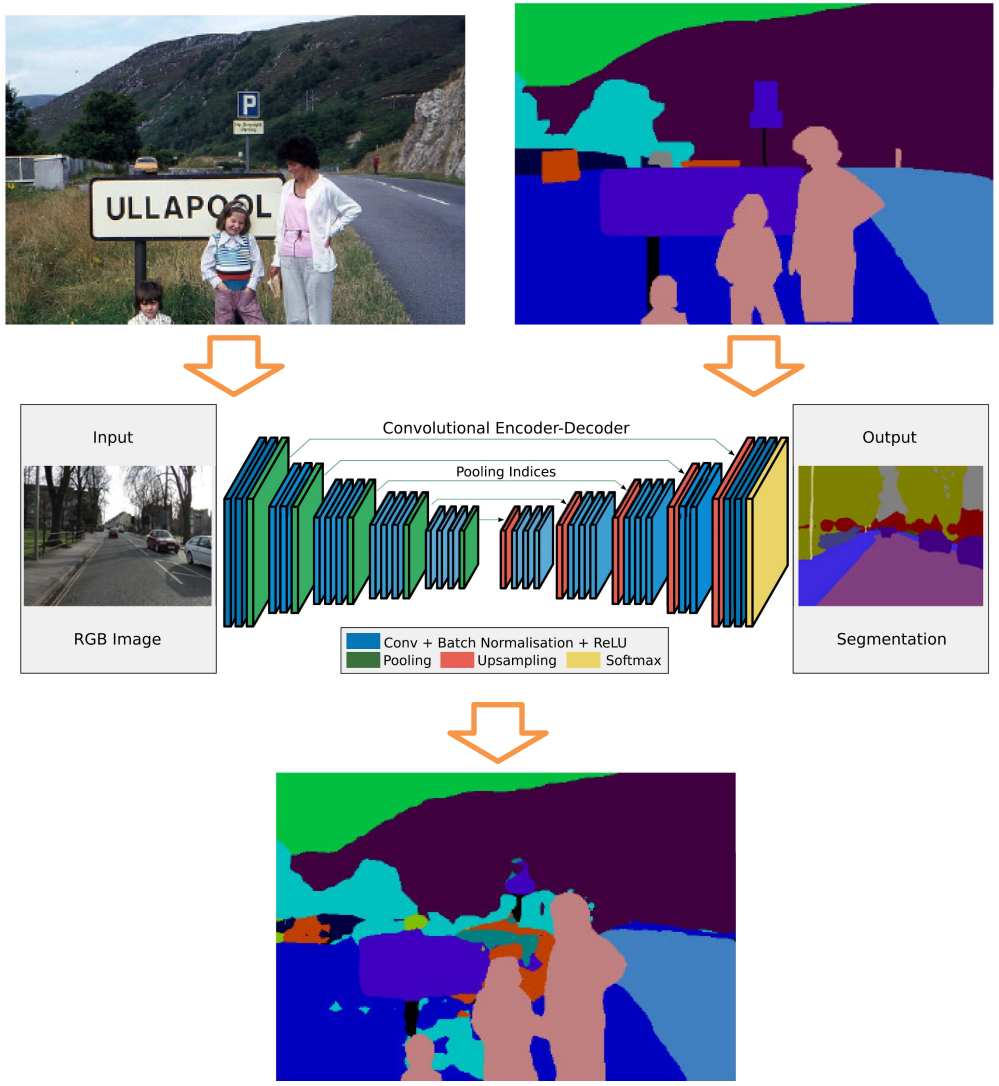



在推理阶段，对于每个新图像都逐像素进行分类。

# 1.1 图像分割概述

## 图像分割的基本流程

- **输入**：图像（RGB）
- **算法**：深度学习模型
- **输出**：与输入大小一致的单通道图
- **训练过程**：
  - 输入：image + Label
  - 前向：out = model(image)
  - 计算损失：loss = loss_func(out, label)
  - 反向：loss.backward()
  - 更新权重：optimizer.minimize(loss)

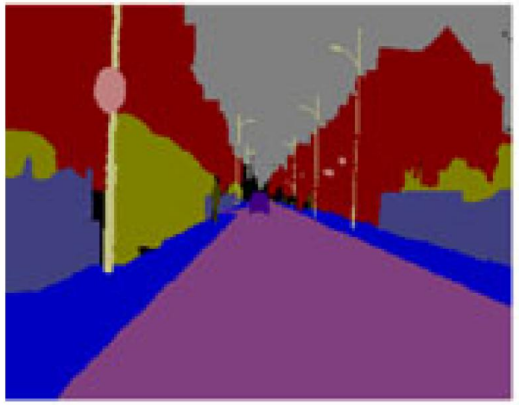


# 1.1 图像分割概述

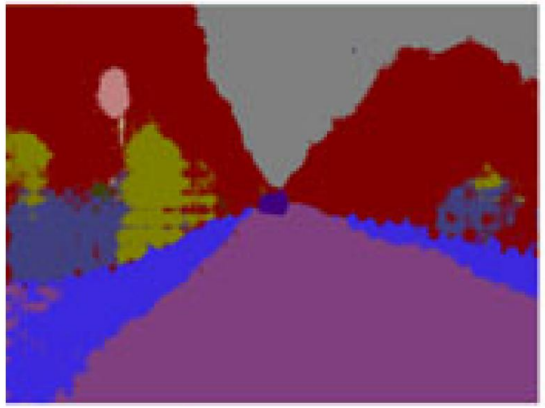
## 图像分割的根本目的

### 图像分割的根本目的：像素级分类

输入图像



Ground Truth



预测图

# 1.2 图像分割的类型

## 图像分割的类型

图像是由像素点构成的矩阵，**图像分割**等同于像素级分类，并将相同类的像素组合在一起。图像分割包含三种典型任务**语义分割**、**实例分割**和**全景分割**。

- **语义分割（Semantic Segmentation）**是典型的图像分类问题，它将整个图像分成若干像素组，并对其进行标记和分类，每个类别使用一种颜色（ID）进行标识，语义分割是对图像中现有目标进行精确的边界分割。

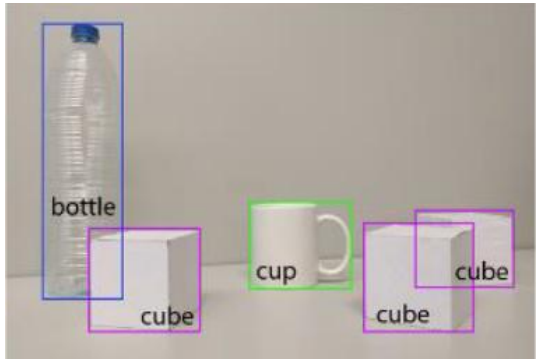
- **实例分割（Instance Segmentation）**是更严格的分割方法，可以理解为语义分割和目标检测的结合，除了实现语义分割的功能外，它还需要将属于同一个类别的不同个体区分开。

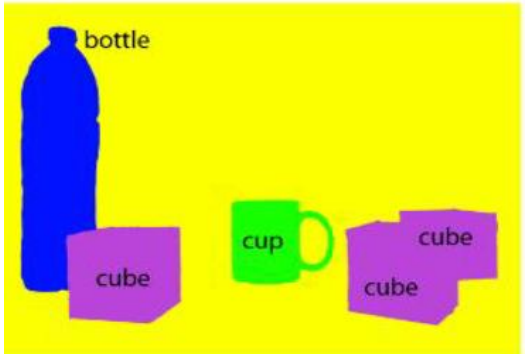
- **全景分割（Panoptic Segmentation）**在实例分割的基础上将非感兴趣区域进行像素级分类，同时对前景和背景进行分割。

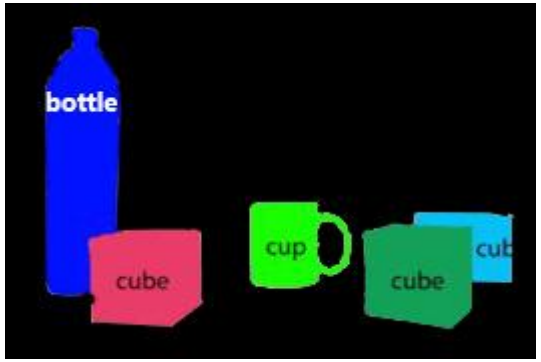
# 1.2 图像分割的类型

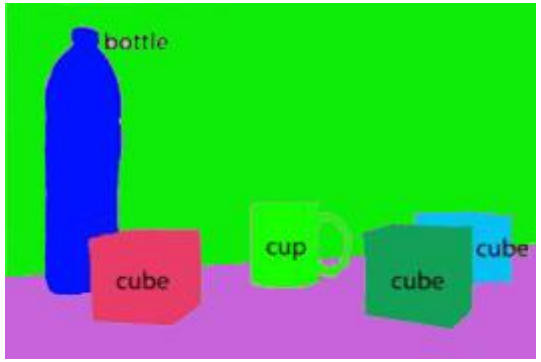
## 语义分割、实例分割和全景分割

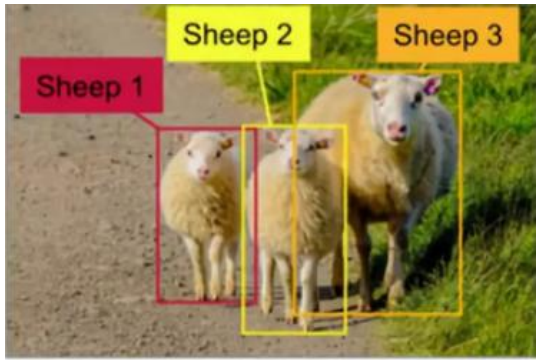
目标检测 + 语义分割 = 实例分割 全景分割

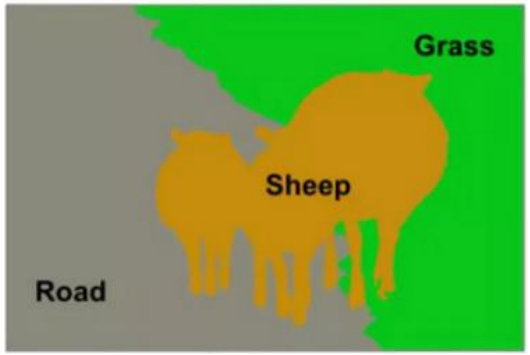


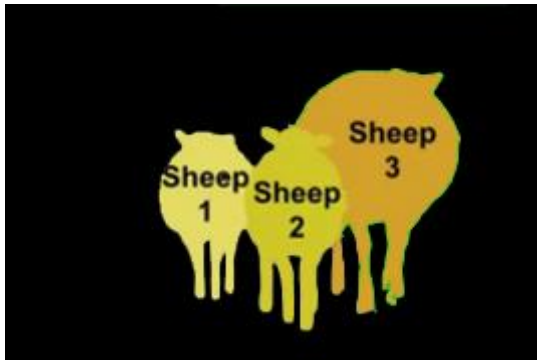



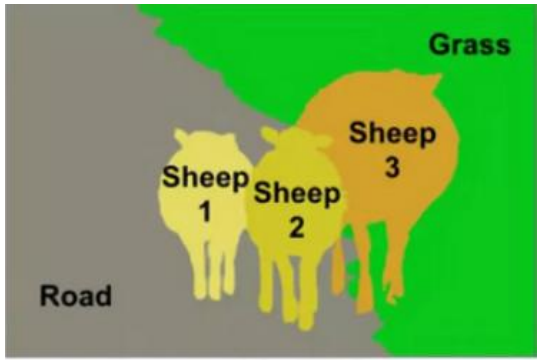




每个像素分类




感兴趣区域
个体分类




背景像素分类
感兴趣区域个体分类

Images form mscoco

# 1.2 图像分割的类型

## 视频目标分割


Video from: http://youtu.be/8f9y17-OAwl

***视频目标分割（VOS）***

给定第一帧指定目标的Mask，求特定目标后续Mask。

# 1.2 图像分割的类型

## 视频实例分割

***视频实例分割（VIS）***

- 根据目标检测的框，求目标的mask；
- 区分不同的对象

Video from: http://youtu.be/8f9y17-OAwI

# 1.3 图像分割的应用

## 图像分割的应用

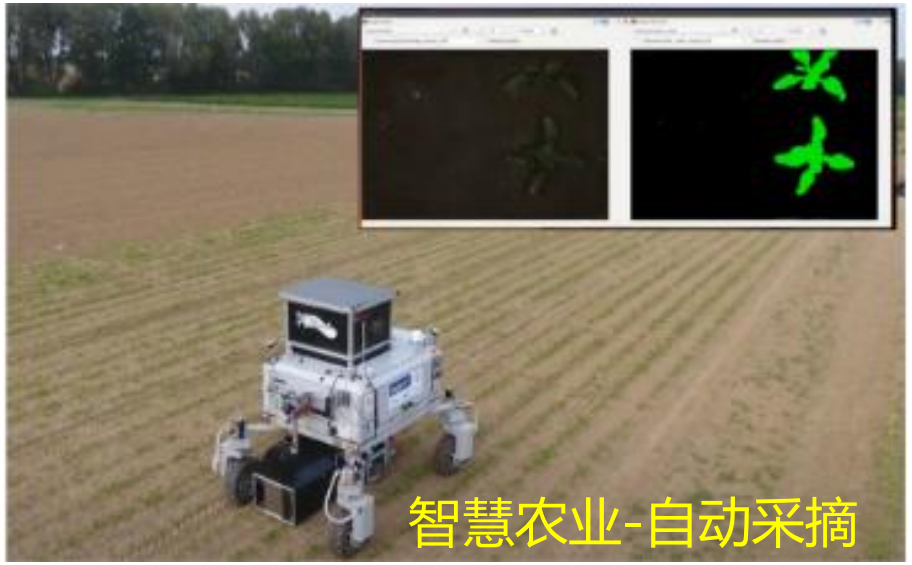
**图像分割**是场景理解（Scene Understanding）的关键技术，在多种应用场景中具有重要意义，是计算机视觉领域核心问题之一。

- 人像分割：头发分割、人脸分割、背景分割
- 自动驾驶：行人检测、车辆分割、车道线检测
- 医学图像：病例图、CT、MRI
- 工业质检：工业机器人、分拣机器人

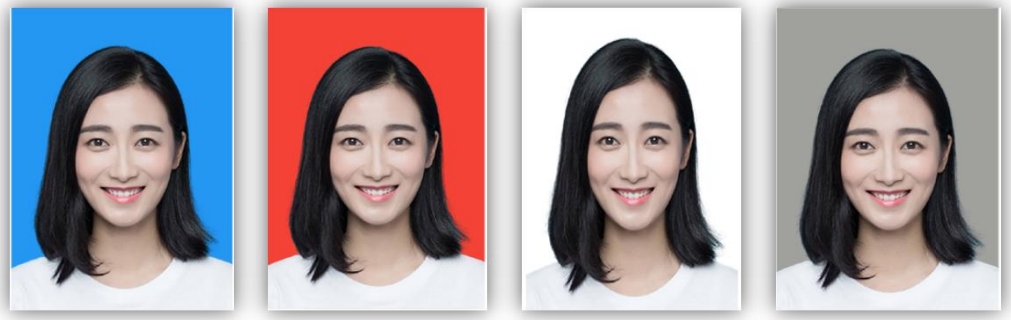
# 1.3 图像分割的应用

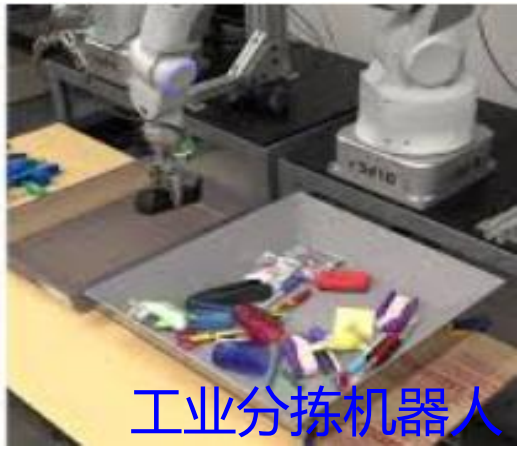
## 图像分割的应用






人脸分割





# 1.4 常见的图像分割数据集

## 图像分割常用数据集

图像分割数据集非常的多，特别是针对不同应用领域的数据集也是玲琅满目的，下面列举出几个常见的、通用的Benchmark数据集。

- PASCAL Visual Object Classes(VOC)数据集
- PASCAL Context数据集
- COCO(Microsoft Common Objects in Context)数据集
- Cityscapes城市道路数据集
- CamVid自动驾驶数据集
- KITTI自动驾驶数据集
- ADE20K室内场景数据集

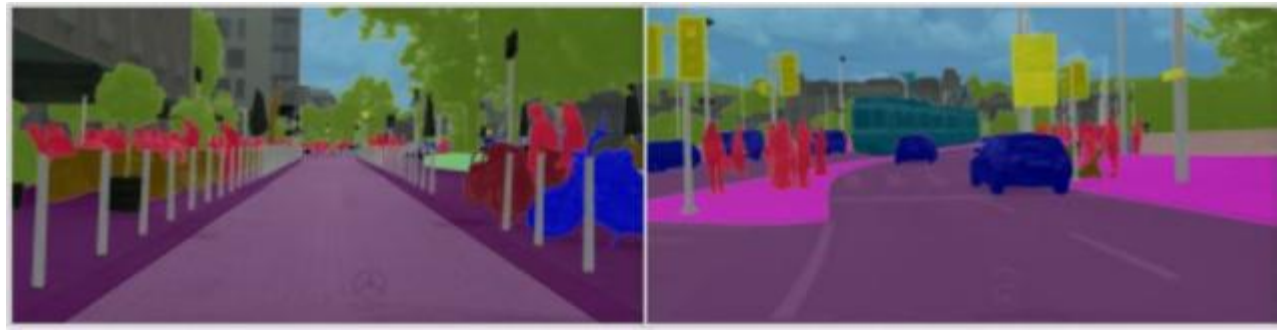
# 1.4 常见的图像分割数据集

## 图像分割数据集

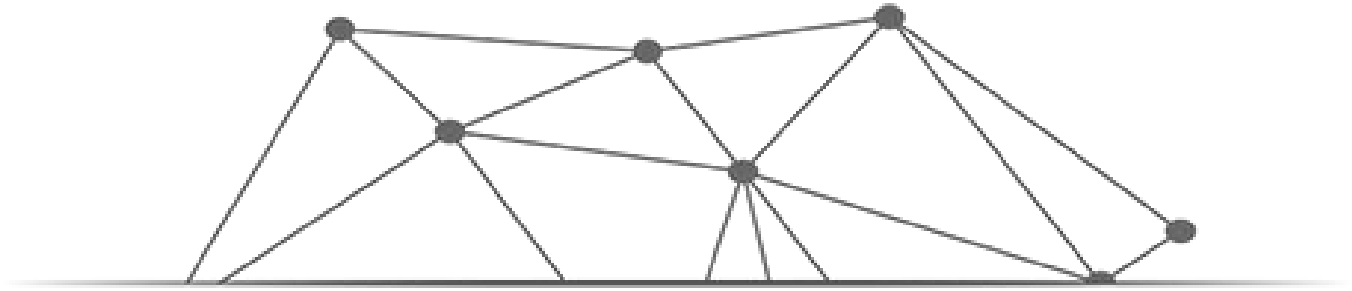
ADE20K


MS COCO

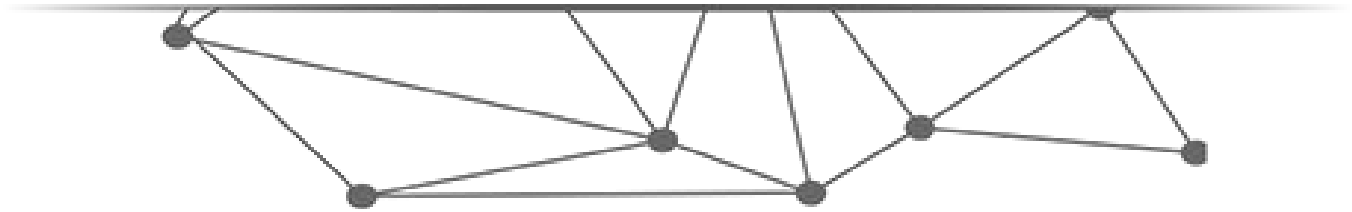
Pascal Context



Cityscapes



# 课堂互动 13.3.1



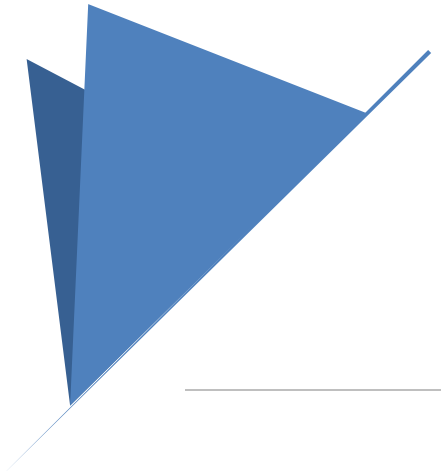
Part
02

# 评价指标和关键技术

/ 图像分割的评价指标

/ 图像分割的关键技术

/ 传统图像分割技术



# 图像分割的评价指标

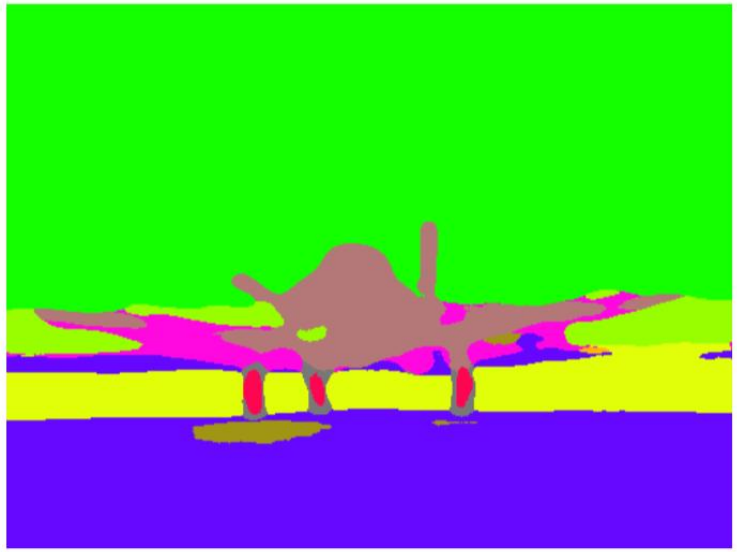
# 2.1 图像分割的评价指标

## 分割质量的评价

输入图像

预测图1

预测图2

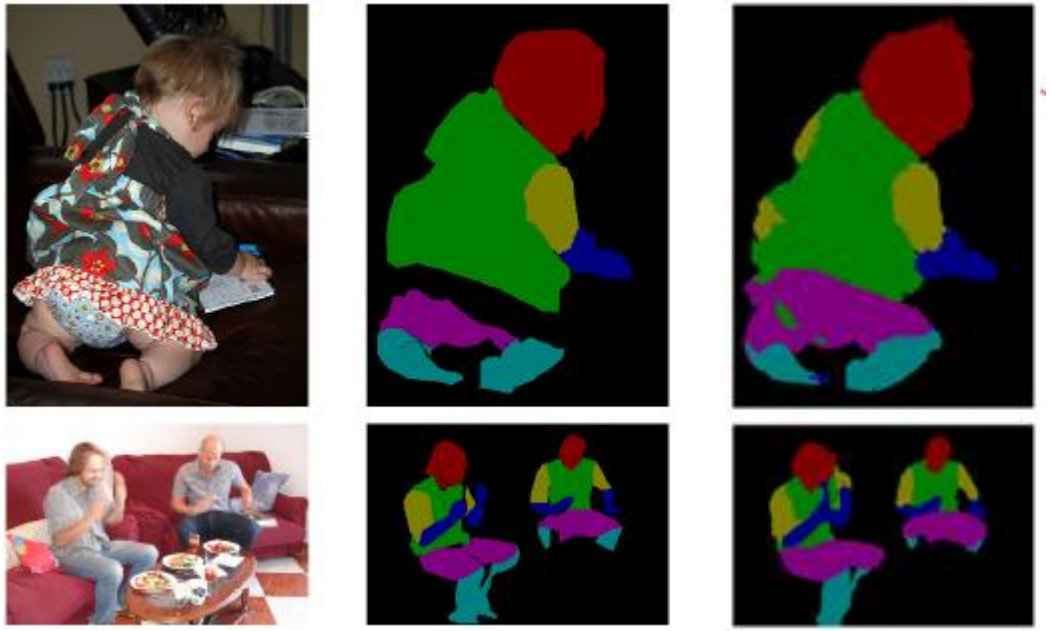


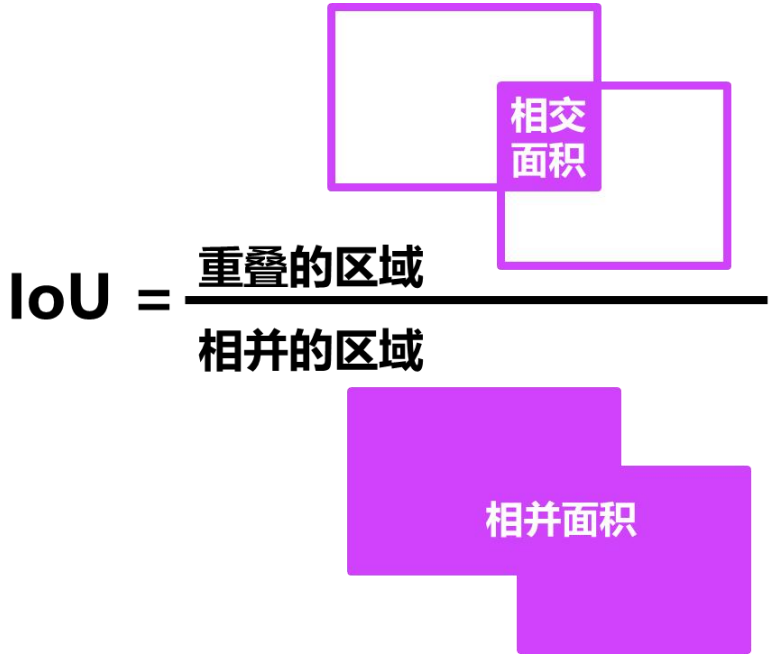
**哪一个输出结果更好?**

# 2.1 图像分割的评价指标

## mIOU和mAcc

- mIoU：mean Intersection-Over-Union
- mACC: mean Accuracy






$$IoU = \frac{\text{重叠的区域}}{\text{相并的区域}}$$

**IoU**：每一类的IoU

**mIoU**：所有类IoU的平均值

**ACC**：每一类正确像素点与所有像素点的比例

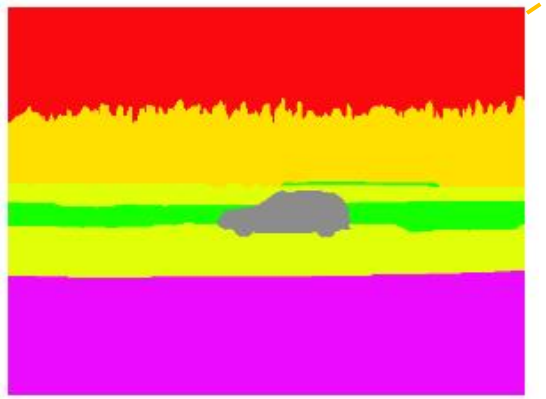
**mACC**：所有类ACC的平均值

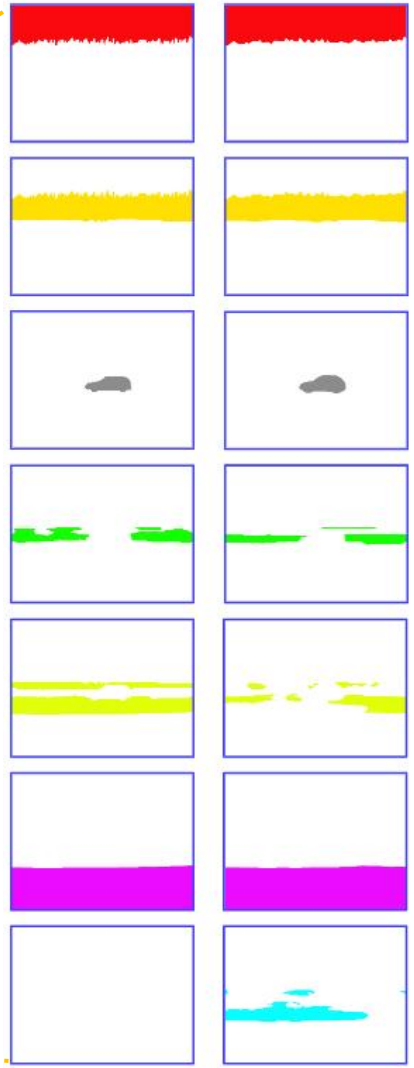
# 2.1 图像分割的评价指标

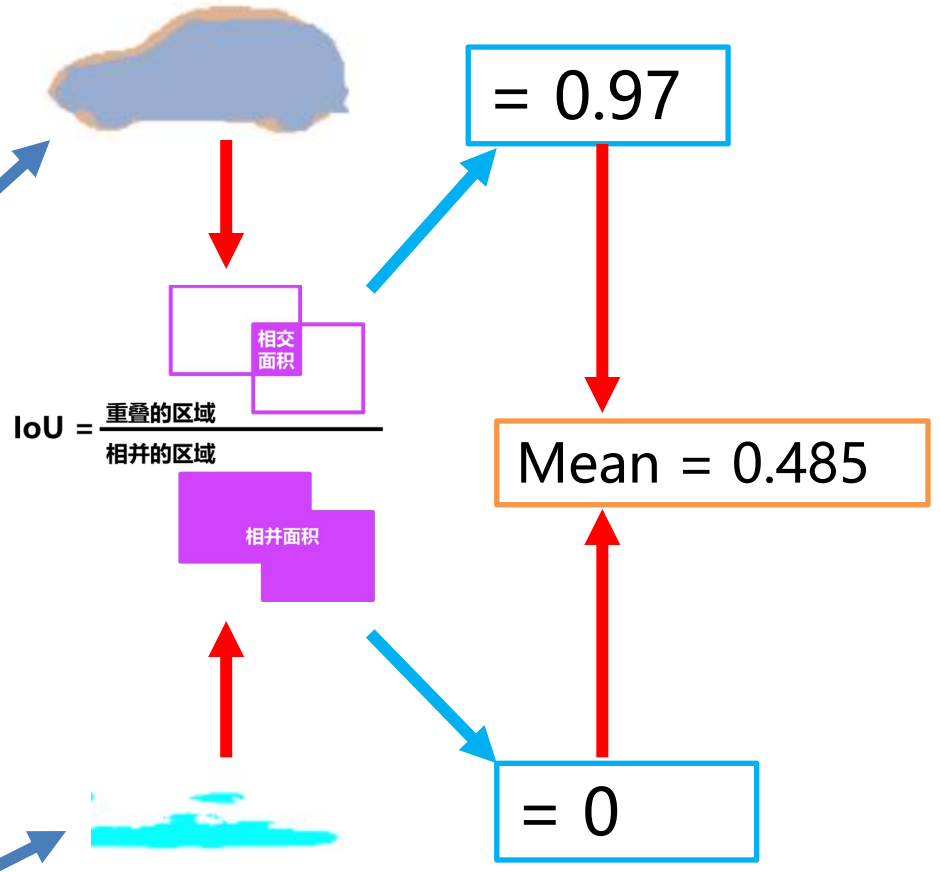
## 平均交并比（mIoU）

**IoU**：每一类的IoU

**mIoU**：所有类IoU的平均值

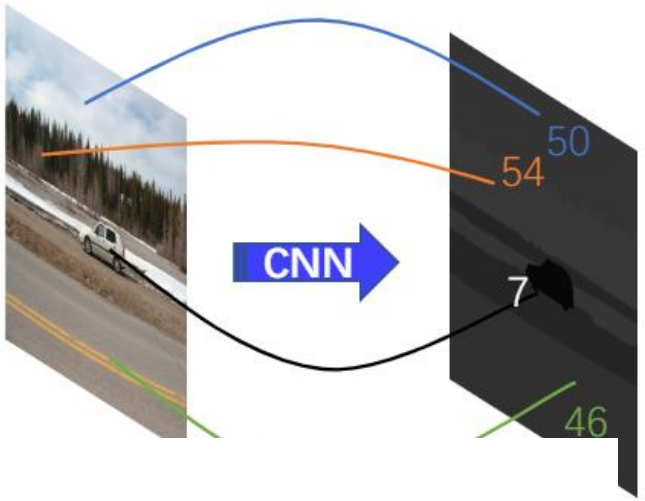






# 2.1 图像分割的评价指标

## 平均精确度（mAcc）




**ACC**：每一类正确像素点与所有像素点的比例

**mACC**：**方法一**：所有类ACC的平均值；

**方法二**：Pred和GT对应位置的“分类”准确率

预测

| | | | | |
|---|---|---|---|---|
| 50 | 50 | 50 | 50 | 50 |
| 54 | 54 | 54 | 54 | 54 |
| 37 | 37 | 7 | 37 | 51 |
| 46 | 46 | 46 | 46 | 46 |
| 46 | 46 | 46 | 46 | 46 |

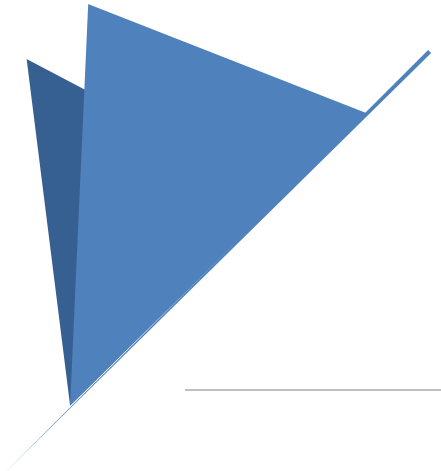
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 50 | 50 | 50 | 50 | 50 | . | . | . | 46 | 46 |

Ground Truth

| | | | | |
|---|---|---|---|---|
| 50 | 50 | 50 | 50 | 50 |
| 54 | 54 | 54 | 54 | 54 |
| 37 | 37 | 7 | 37 | 51 |
| 46 | 46 | 46 | 46 | 46 |
| 46 | 46 | 46 | 46 | 46 |

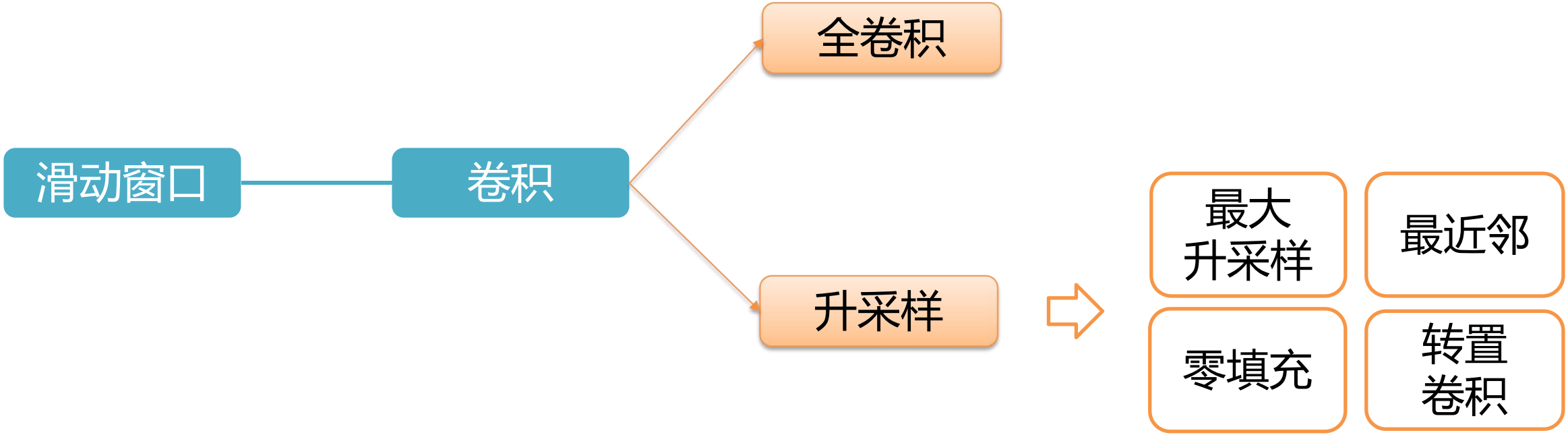
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 50 | 50 | 50 | 50 | 50 | . | . | . | 46 | 46 |

精确度



# 图像分割的关键技术

# 2.2 图像分割关键技术



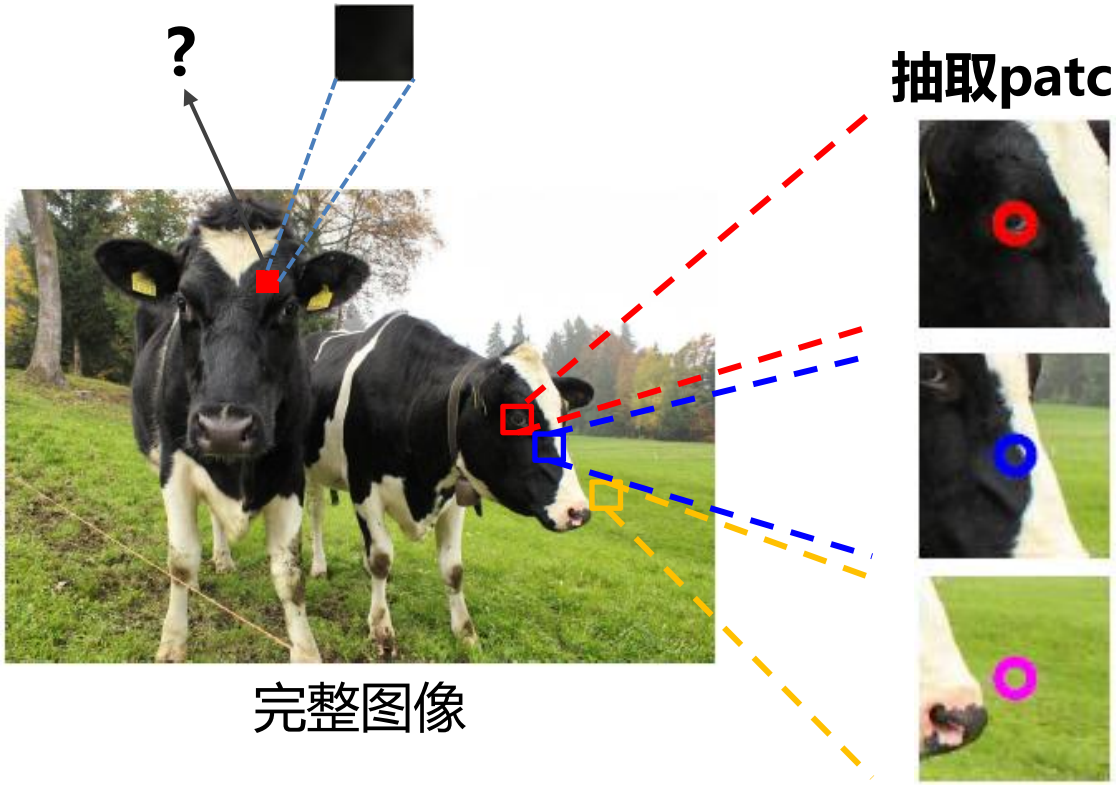
# 2.2 图像分割关键技术

## 滑动窗口 Sliding Window

像素没有上下文信息无法进行分类

Q：如何对没有上下文的像素进行分类?

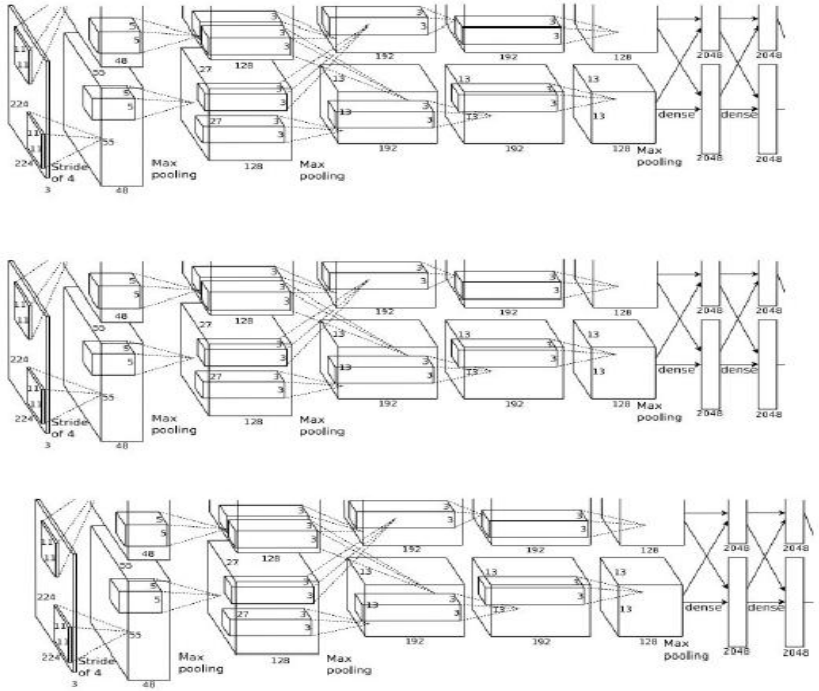
**速度慢，且效率极低！重叠的patch无法重用共享特征。**




**抽取patch**



**使用CNN分类中心像素**



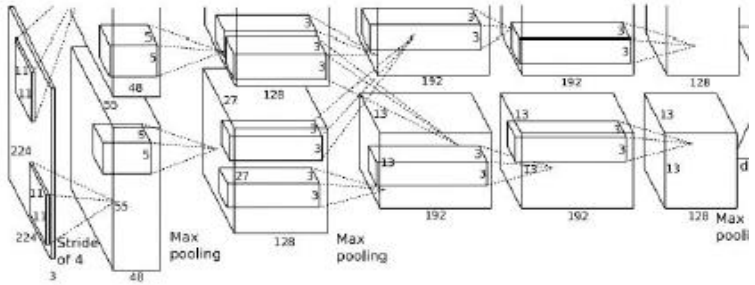

Farabet et al, “Learning Hierarchical Features for Scene Labeling,” TPAMI 2013

Pinheiro and Collobert, “Recurrent Convolutional Neural Networks for Scene Labeling”, ICML 2014

# 2.2 图像分割关键技术

## 卷积 Convolution

完整图像




输出预测

**直觉：**使用卷积网络对整个图像进行编码，并在顶部进行语义分割。

**问题：**基于卷积的分类架构中的**卷积特征图**的尺寸会随着网络的加深逐渐变小。然而，图像分割要求输出的**预测图**和**输入图像**具有相同的尺寸。

# 2.2 图像分割关键技术

## 全卷积 Fully Convolutional

只包含卷积层，不包含降采样操作，可以一次性对所有的像素都进行预测。

Input:
3 x H x W

Conv → Conv → Conv → Conv → argmax

卷积层尺度始终不变
D x H x W

Scores:
C x H x W

Predictions:
H x W

激活值最大的通道的索引作为最终的类别的索引

**问题：**始终使用原始图像分辨率进行卷积操作，计算代价非常高。

每个类别都会生成一张分数图，总共C类

# 2.2 图像分割关键技术

## 全卷积网络 Fully Convolutional

设计一个包含一系列卷积层的网络（不含全连接层），并在中间同时集成**升采样**和**降采样**结构！

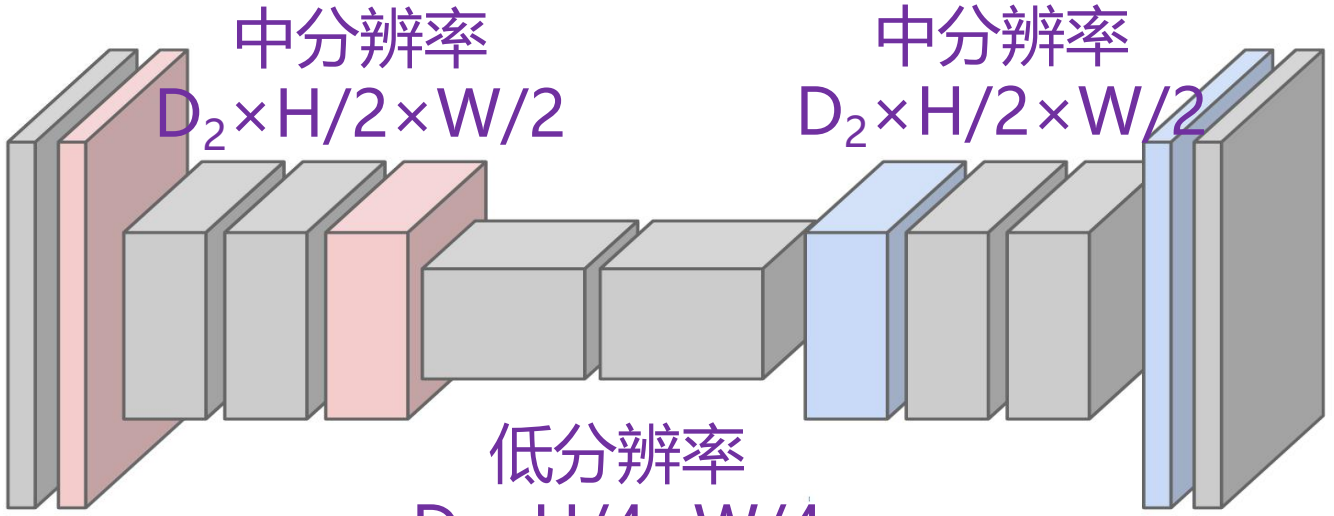
**降采样：**

**Pooling, strided convolution**

**升采样：**

**？？？**

**输入图像**
**3×H×W**



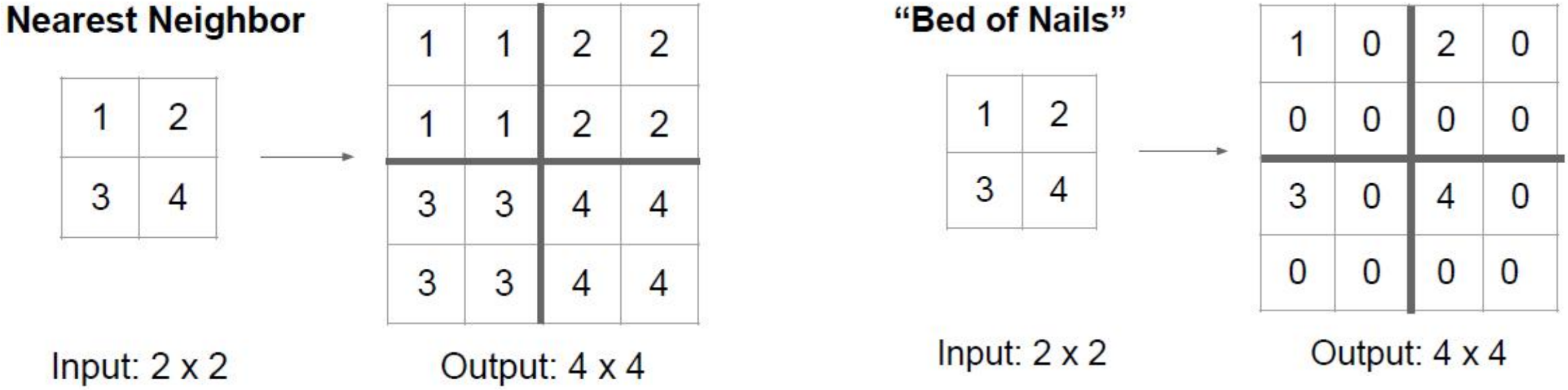

**预测图像**
**H×W**

Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015
Noh et al, “Learning Deconvolution Network for Semantic Segmentation”, ICCV 2015

# 2.2 图像分割关键技术

## 升采样

**升采样(Unpooling)**又称为上采样，指任何可以让图像(卷积特征图)变成更高分辨率的技术，通常使用**重采样**或**插值算法**实现。常见的升采样操作包括：**最近邻**，**零填充**，**最大升采样**和**转置卷积**。



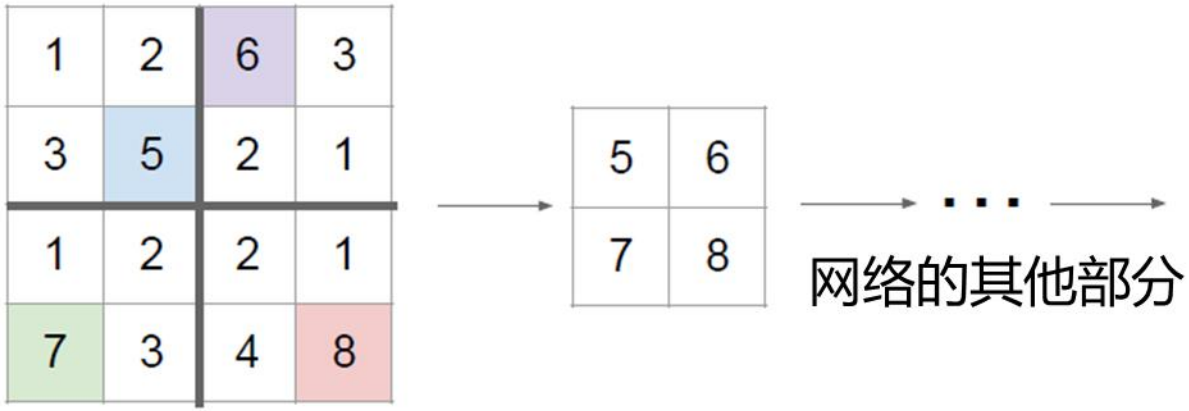
# 2.2 图像分割关键技术

## 升采样 - 最大升采样

升采样在CNN中常被看作Max-Pooling的逆操作。然而，Max-Pooling是不可逆的。

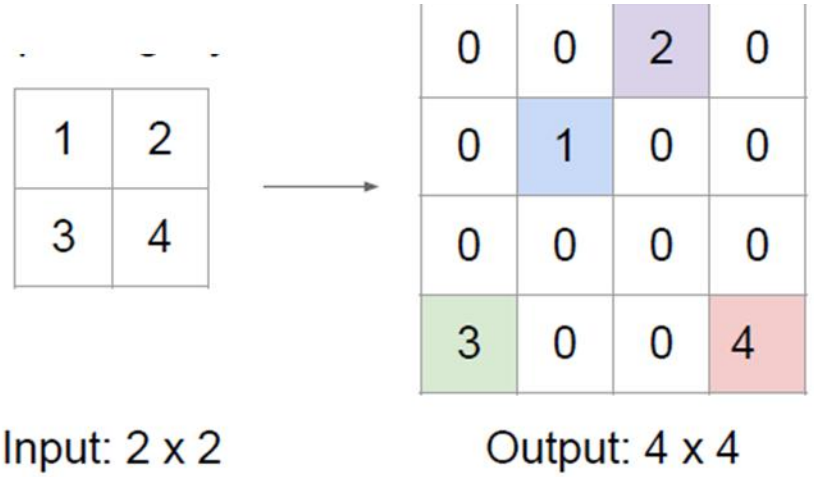
**Max Pooling**

需要**记住**哪个位置是最大值

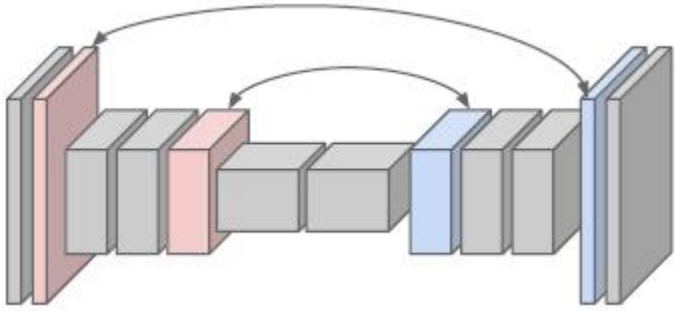



**Max UnPooling**

**调用**Pooling层的位置信息进行升采样



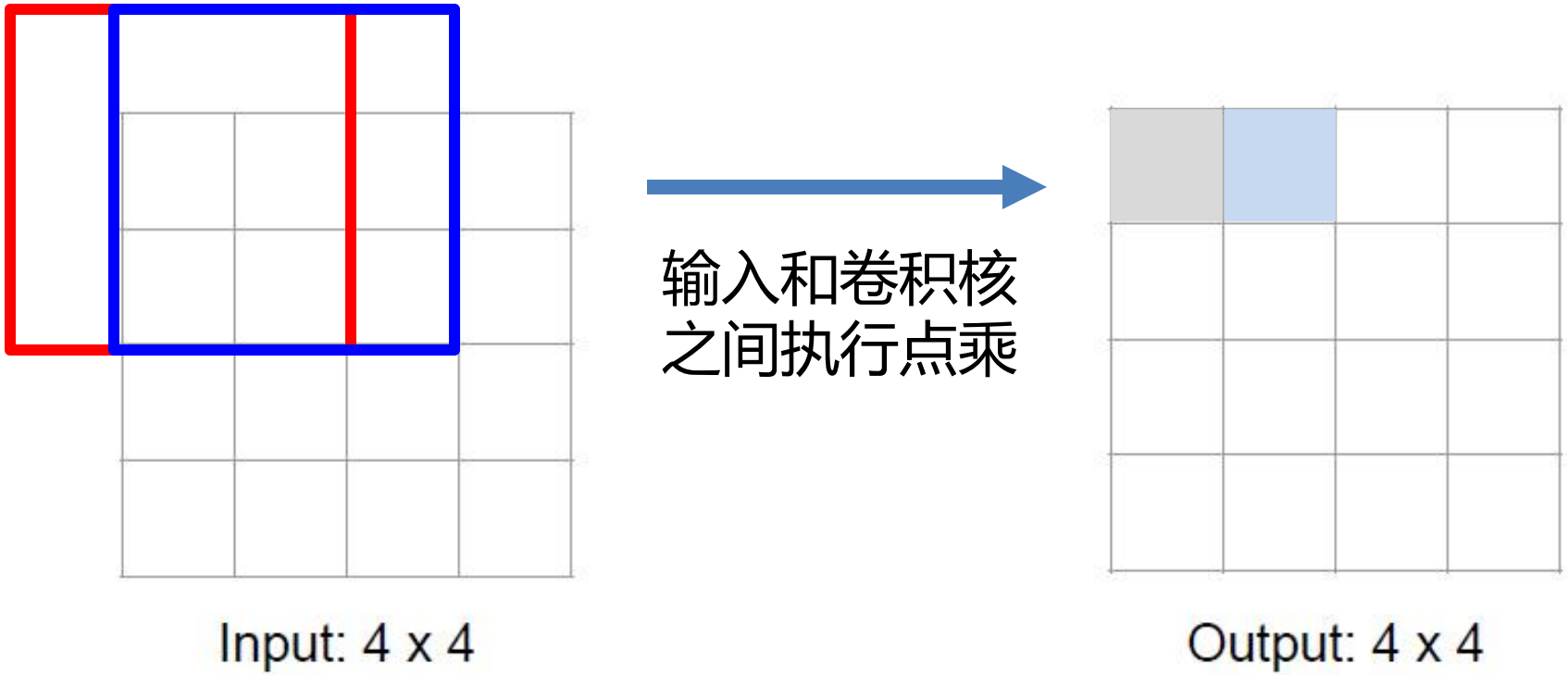

**升采样层和降采样层互为关联对**



# 2.2 图像分割关键技术

## 可学习的升采样 – 转置卷积

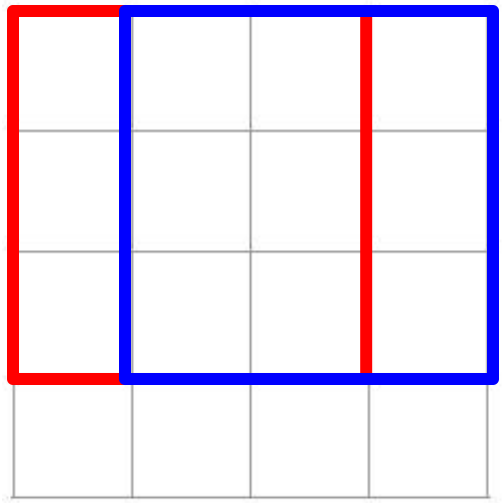
**回顾：**普通3×3卷积，步长：1，填充：1



# 2.2 图像分割关键技术

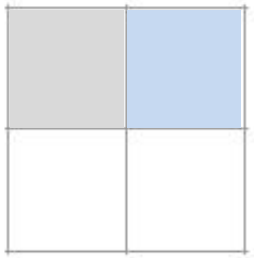
## 可学习的升采样 – 转置卷积

**回顾：**普通3×3卷积，**步长：2**，填充：1



Input: 4 x 4

输入和卷积核之间执行点乘



Output: 2 x 2

卷积核在输入中移动2个像素，在输出中移动1个像素。

步长给出了输入到输出之间的运动比例。

我们可以把这种跨步长的卷积称为“**可学习降采样**”。

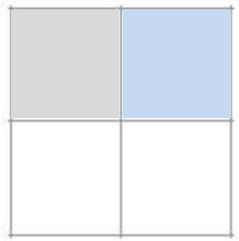
# 2.2 图像分割关键技术

## 可学习的升采样 – 转置卷积

**其他名称**

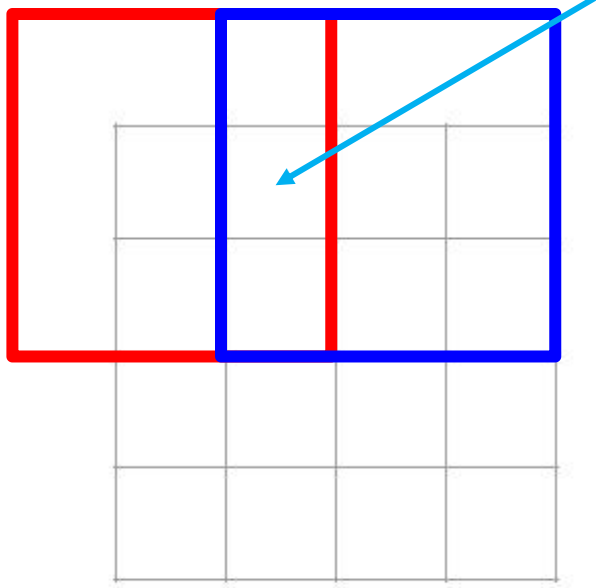
- 反卷积 (Deconvolution)(bad)
- 升卷积 (Upconvolution)
- 分数步长卷积 (Fractionally strided convolution)
- 后馈步长卷积 (Backward strided convolution)

3×3 **转置**卷积，**步长：2**，填充：1



Input: 2 x 2

输入为卷积核提供权重



Output: 4 x 4

输出重叠部分执行加和操作

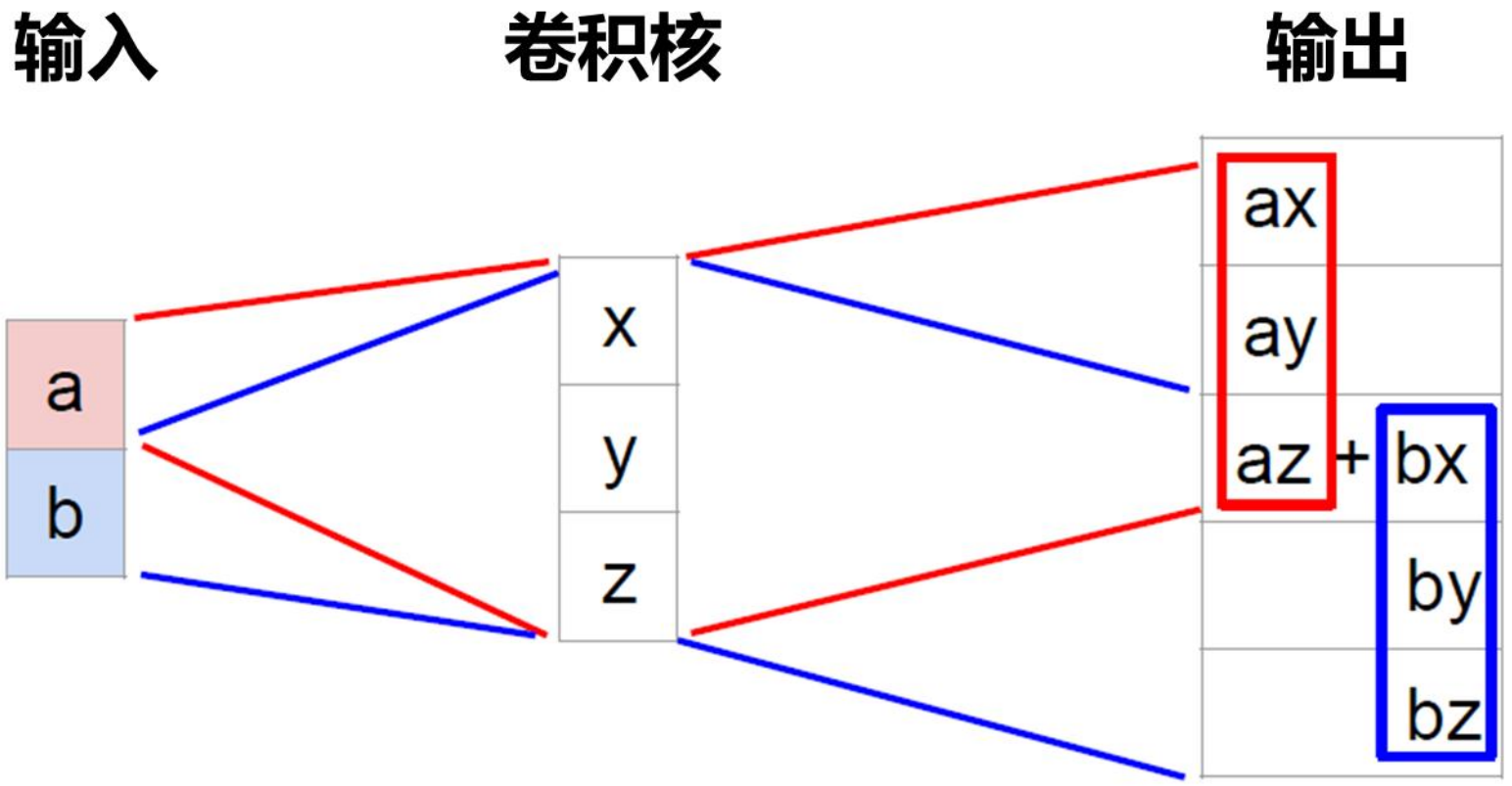
输入每移动1个像素，卷积核在输出中移动2个像素

步长给出了输入到输出之间的运动比例。

**Q: 为什么称为转置卷积?**

# 图像分割关键技术

## 可学习的升采样 – 1维运算案例




输出是使用卷积核对输入的加权，重叠部分执行加和。

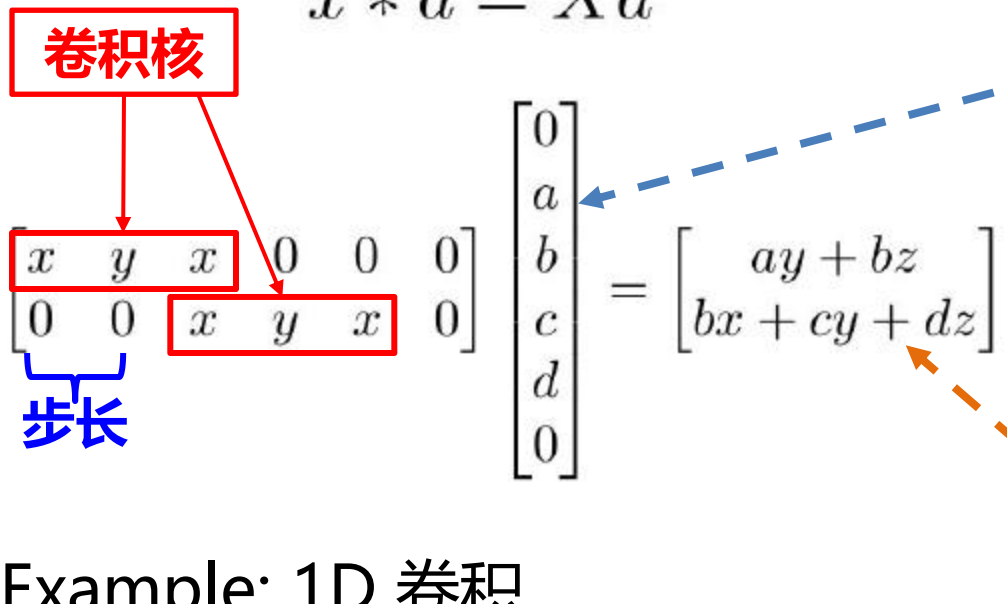
卷积可以理解为向量的内积操作。

# 2.2 图像分割关键技术

## 可学习的升采样 – 1维运算案例

**卷积运算可以表示为矩阵乘法**

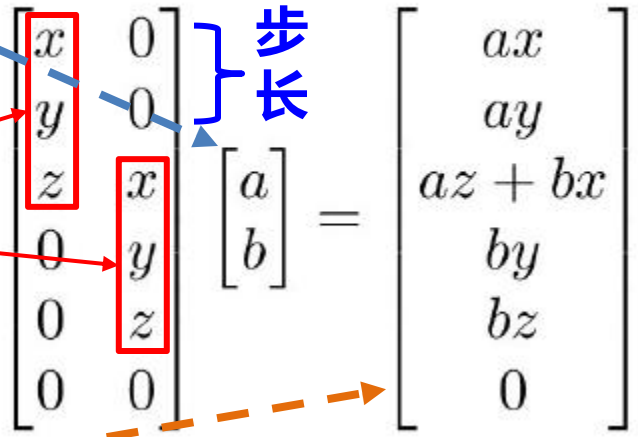
$$\vec{x} * \vec{a} = X\vec{a}$$




Example: 1D 卷积
kernel_size:3, stride=2, padding=1

**卷积转置可以表示为将卷积核转置后的矩阵乘法**

$$\vec{x} *^T \vec{a} = X^T\vec{a}$$




Example: 1D 转置卷积
kernel_size:3, stride=2, padding=1

# 2.2 图像分割关键技术

## 全卷积 Fully Convolutional

设计一个包含一系列卷积层的网络，并在中间同时集成**升采样**和**降采样**结构！

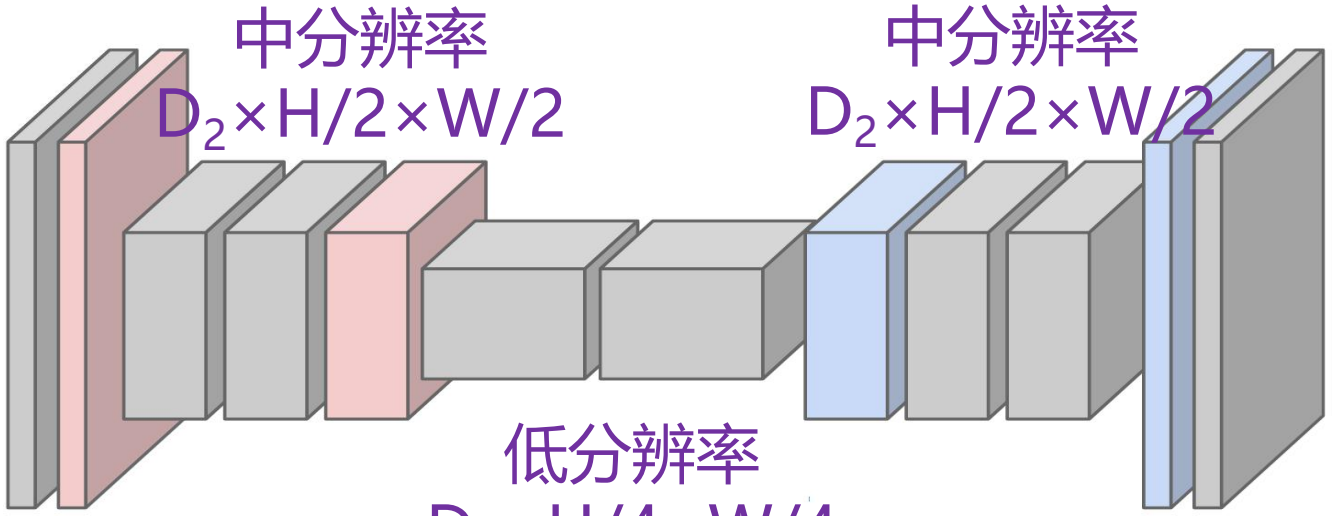
**降采样：**

**Pooling, strided convolution**

**升采样：**

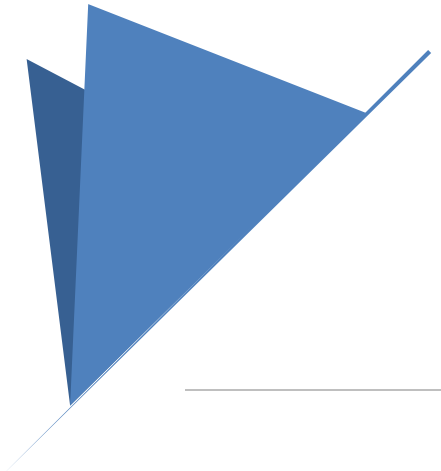
**最大升采样或转置卷积**

**输入图像**
**3×H×W**




**预测图像**
**H×W**

Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015
Noh et al, “Learning Deconvolution Network for Semantic Segmentation”, ICCV 2015



# 传统图像分割技术

# 2.3 传统图像分割技术

**传统图像分割**根据颜色、灰度级、纹理、形状等**低级视觉特征(low-level)** 将图像划分为若干个互不相交的区域。同一区域内表现出某种特征的一致性；不同区域间表现出明显的特征差异。

- **基于阈值的分割方法**

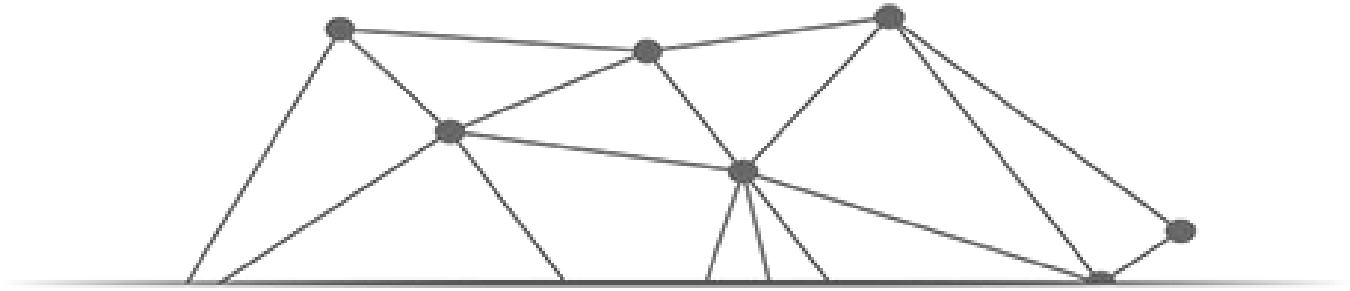
利用阈值对图像进行分割，方法简单粗暴，但不够精确。阈值的设定分为全局阈值和局部阈值两种。

- **基于像素聚类的分割方法**

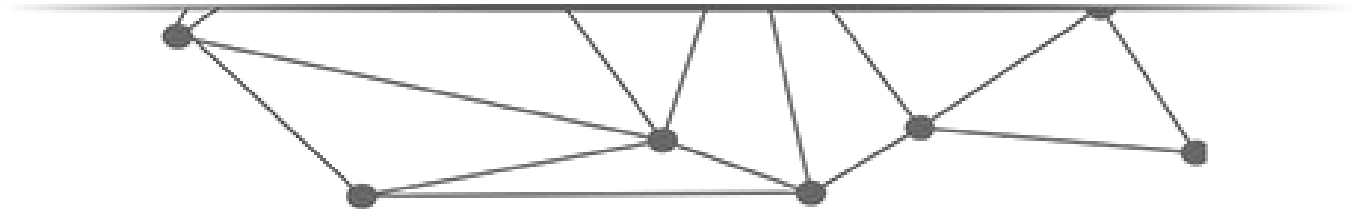
聚类方法先选定一部分像素作为聚类中心，然后根据相邻像素之间的关系进行元素聚类；调整聚类中心反复聚类，直到分割完成且损失函数小于阈值

- **基于图划分的分割方法**

将图像中的像素点作为图模型的顶点，像素间的关系作为图模型的边，利用图模型的最大流来求最小割，典型算法包括N-cut和Grab cut。



# 课堂互动 13.3.2



Part 03

# 经典图像分割模型

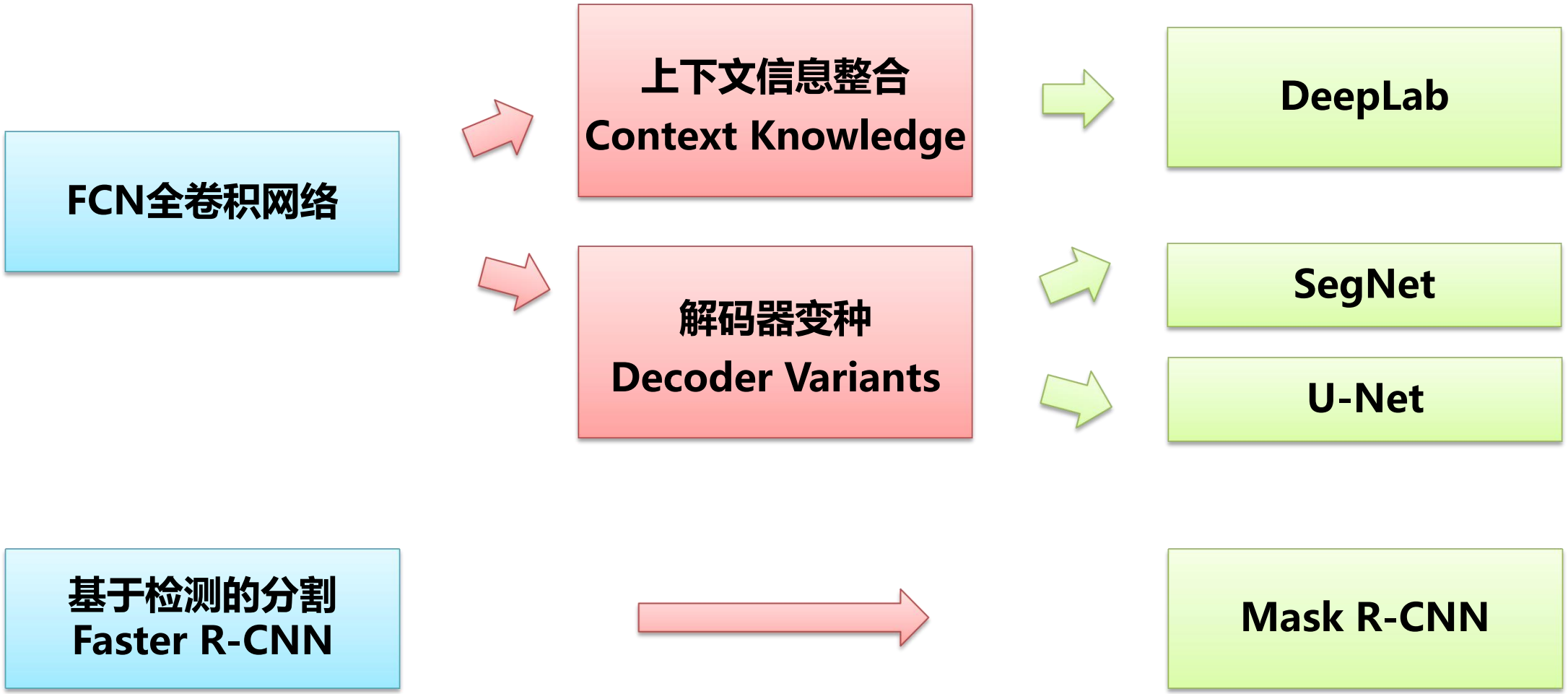
/ FCN全卷积网络

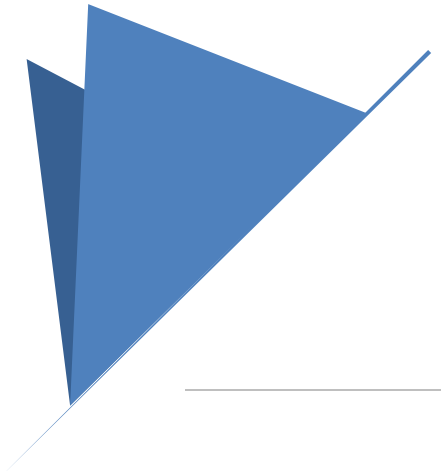
/ U-Net/PSPNet/SegNet 模型

/ DeepLab 系列模型

# 3. 基于卷积神经网络的图像分割

## 主流图像分割体系结构





# FCN 全卷积网络

# 3.1 FCN全卷积网络

## 本节内容

01 **Why does FCN work?**
FCN分割网络的基本概念；FCN的原理；Feature map的上采用

02 **What is FCN?**
FCN网络的结构；FCN每层的具体操作；FCN的层融合

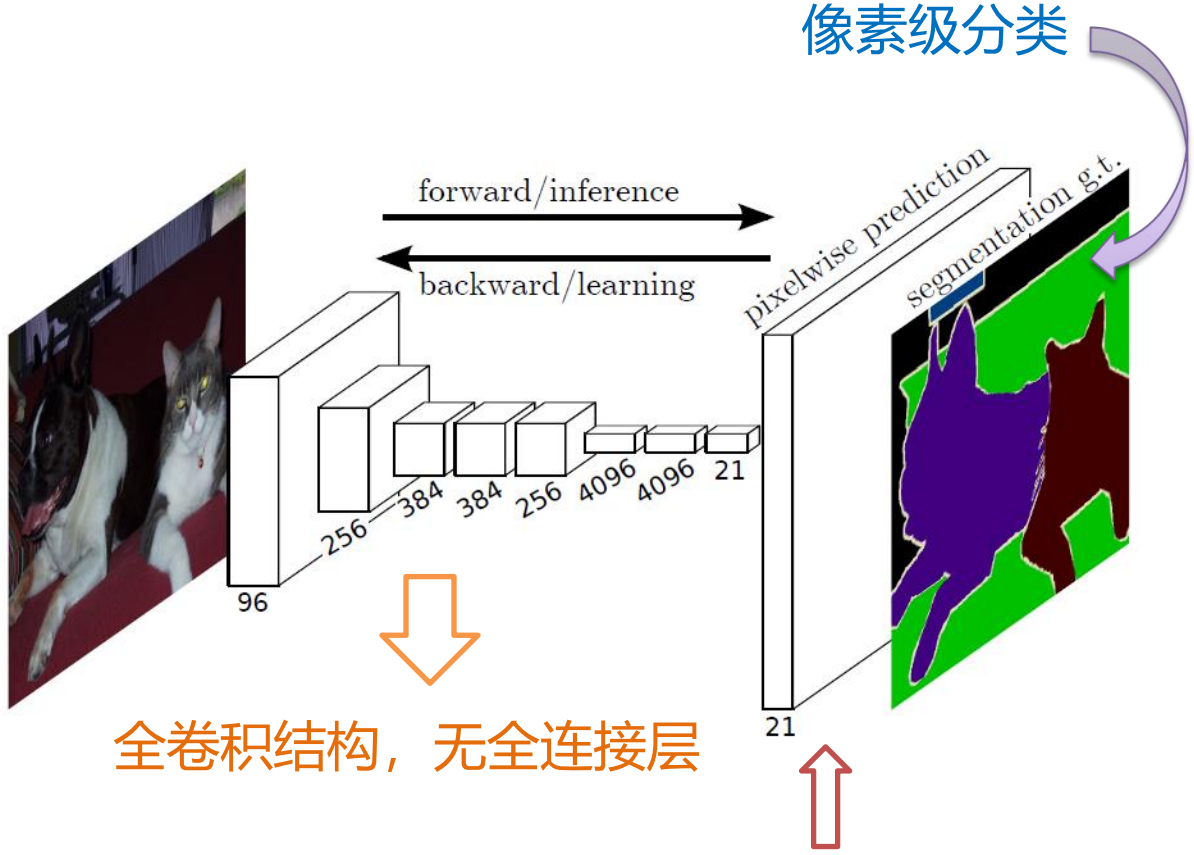
03 **How to build FCN?**
如何实现FCN网络；FCN的骨干网络

# 3.1.1 全卷积网络是怎么工作的?

## FCN全卷积网络简介

**FCN** = Fully Convolutional Network (全卷积网络)

- **什么是全卷积?**
  - ✓ 全卷积，没有全连接层
- **语义分割是什么?**
  - ✓ 语义分割 ≈像素级分类（逐像素分类）
- **图像分割和图像分类的关系**
  - ✓ 将FC层替换为Conv层

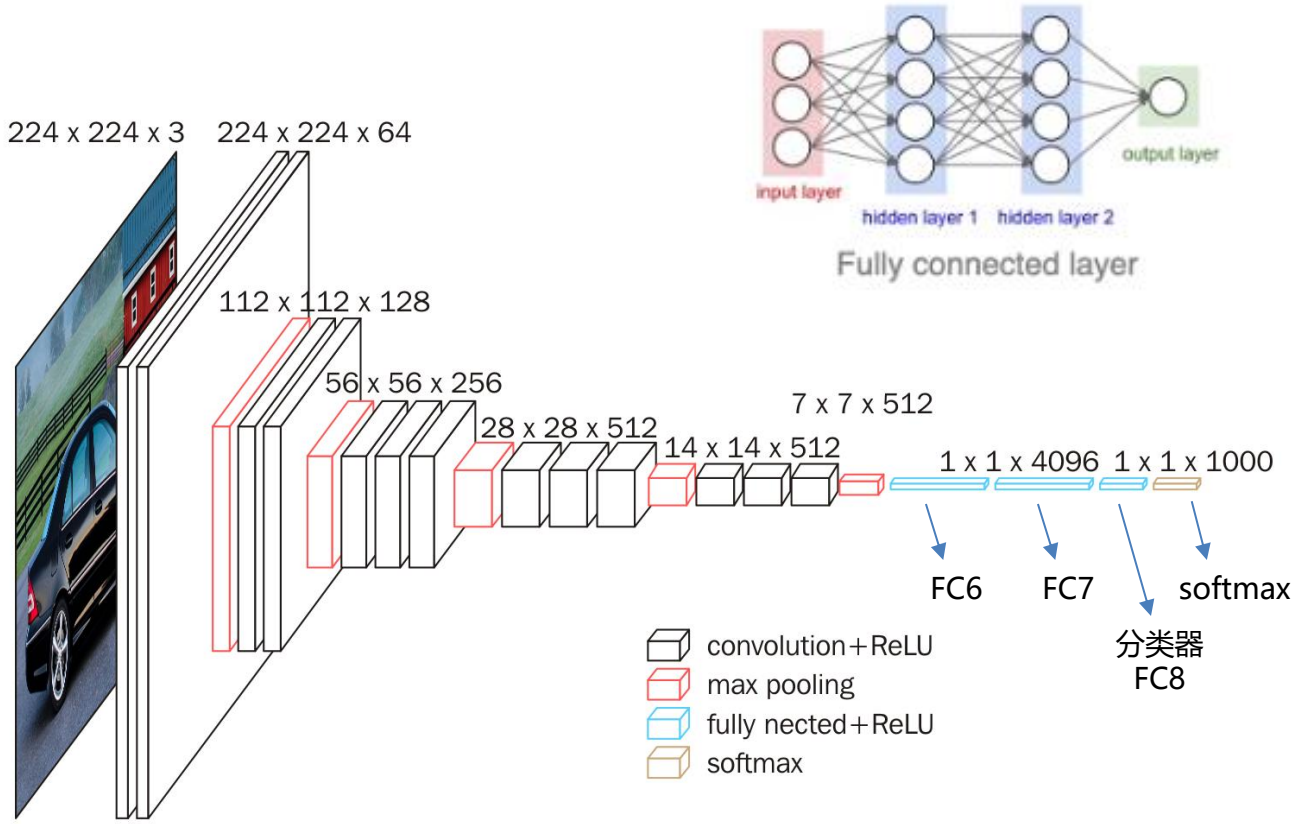



Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR2015

# 3.1.1 全卷积网络是怎么工作的?

## 从图像分类网络到全卷积网络

### 回顾：分类CNN是什么样子的?




VGG-16 拓扑结构图

***图像分类网络的基本结构***

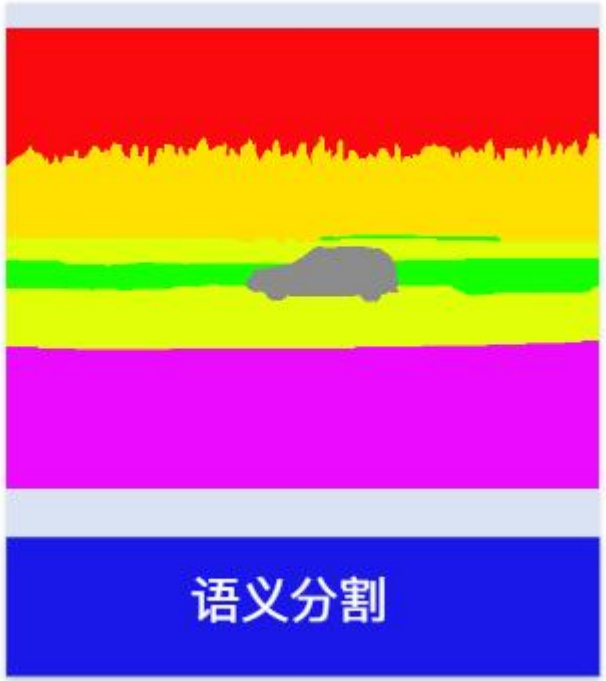
- **输入**：224×224×3
- **特征**：堆叠conv和pooling
- **输出**：1×1×1000（类别）

✓ 分类：1张图 → 1个类别

✓ 语义分割：1张图 → 1张热图

# 3.1.1 全卷积网络是怎么工作的?

## 从图像分类网络到全卷积网络



语义分割

对样本的每个像素逐一进行分类，再组合成一张完整的响应图。

输入图像


图像分类

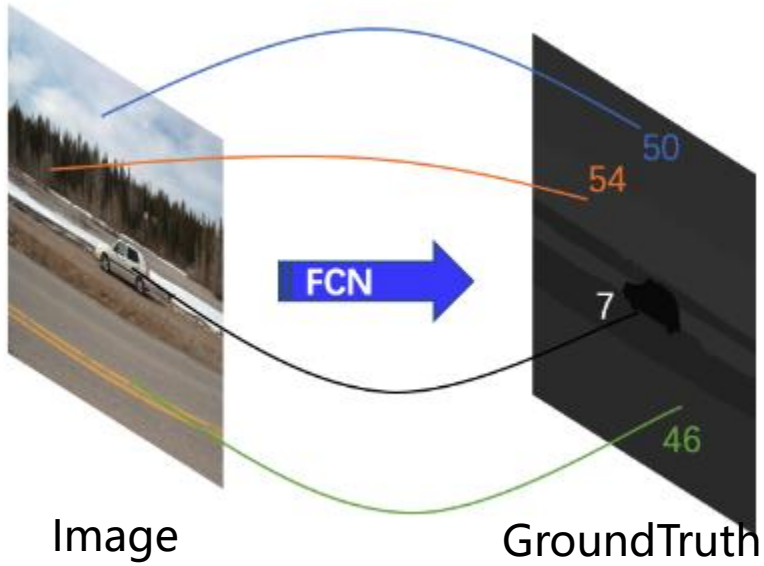
对整幅图像进行分类，将其划分为一个类别，或输出一系列可能类别的概率。

# 3.1.1 全卷积网络是怎么工作的?

## 如何实现分割?

### 全卷积 → 像素级分类

- **语义分割 ≈ 像素级分类**
  - ✓ 输入size == 输出size
- **输入**：RGB图像
  - ✓ 每个位置：(0~255, 0~255, 0~255)
- **输出**：原图中每个像素 对应 的 类别索引
  - ✓ 每个位置：0~n_classes-1



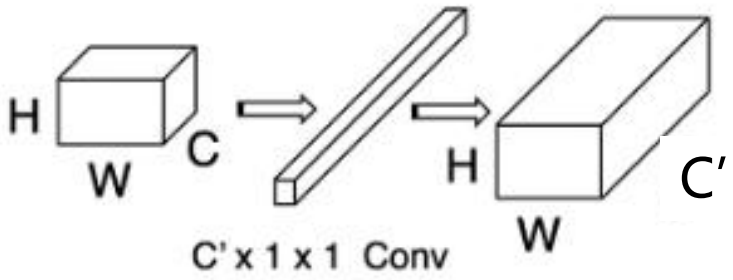

最终目标

# 3.1.1 全卷积网络是怎么工作的?

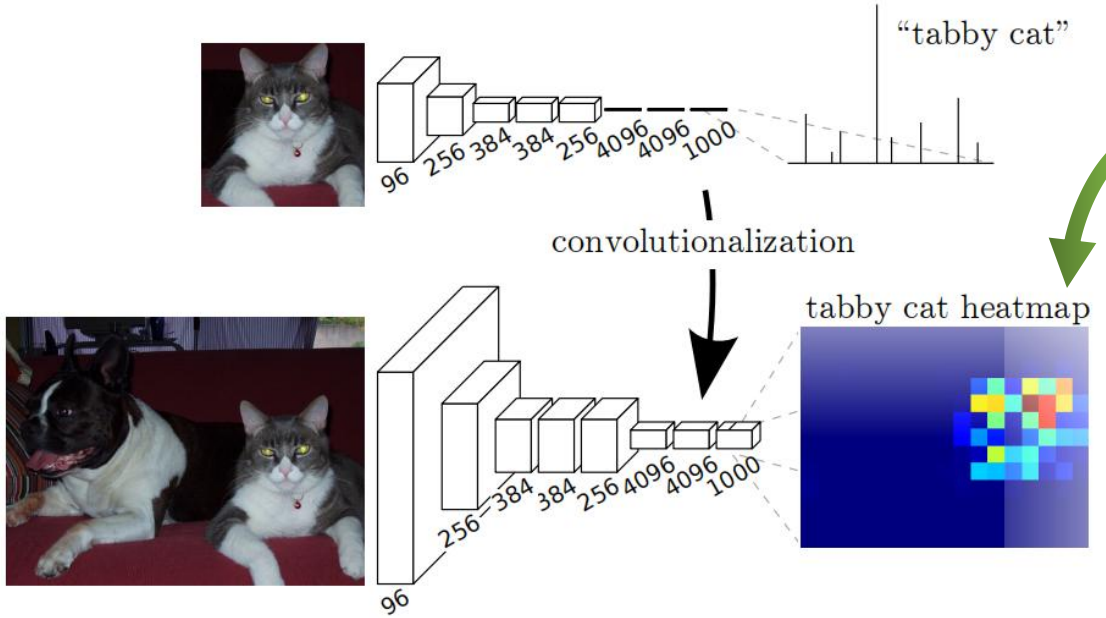
## 如何将全连接层替换为卷积层

FC层 → 1×1卷积

- 1×1卷积Conv：不改变长和宽
  只改变通道的大小




最后的特征网络可以看作是所有类别的热力图。

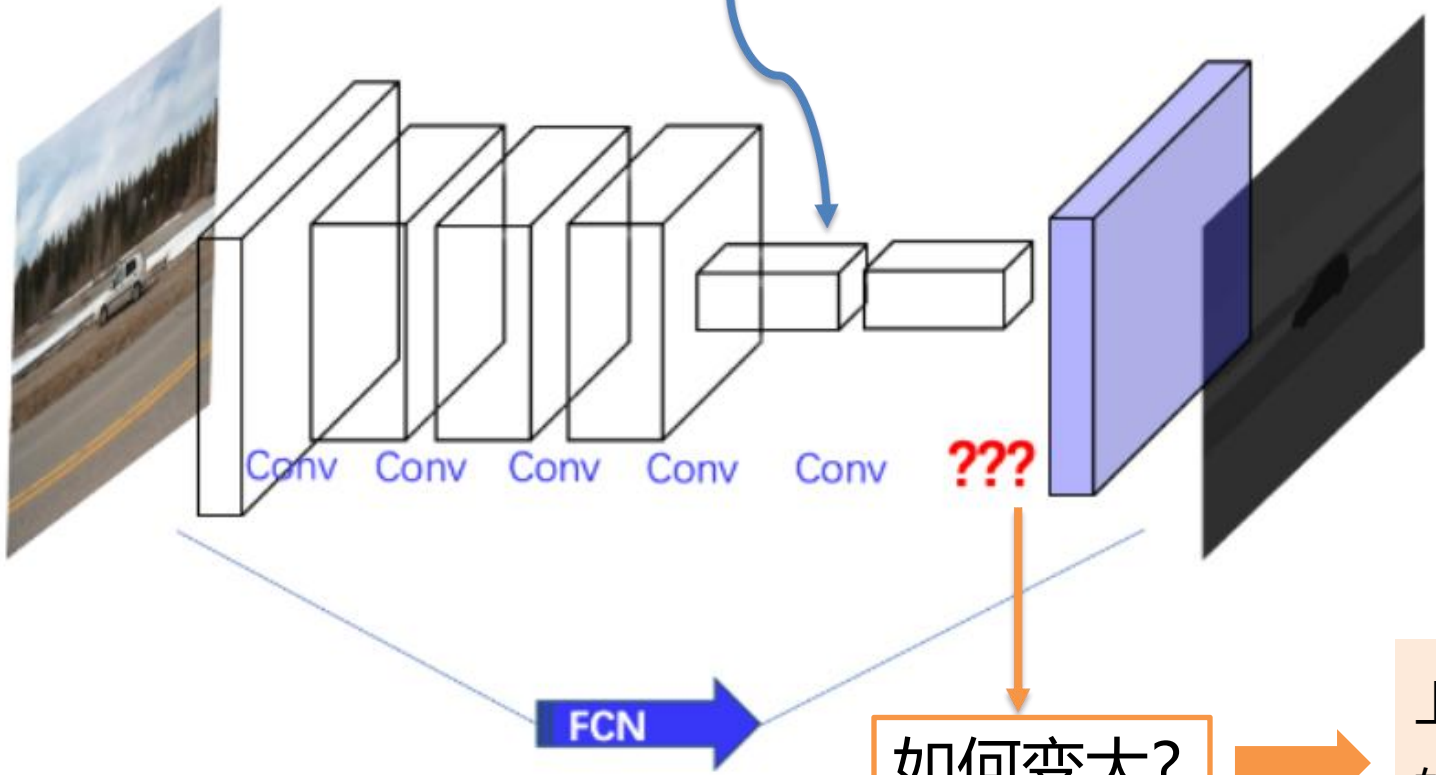



将全连接层转换为卷积层，可以让分类网络能够输出一个热图力(heatmap)。
在VOC中，每个像素包含21个类别。

# 3.1.1 全卷积网络是怎么工作的?

## 如何使特征图变大?

卷积：Featuremap越来越小




如何变大?

上采样Up-samping

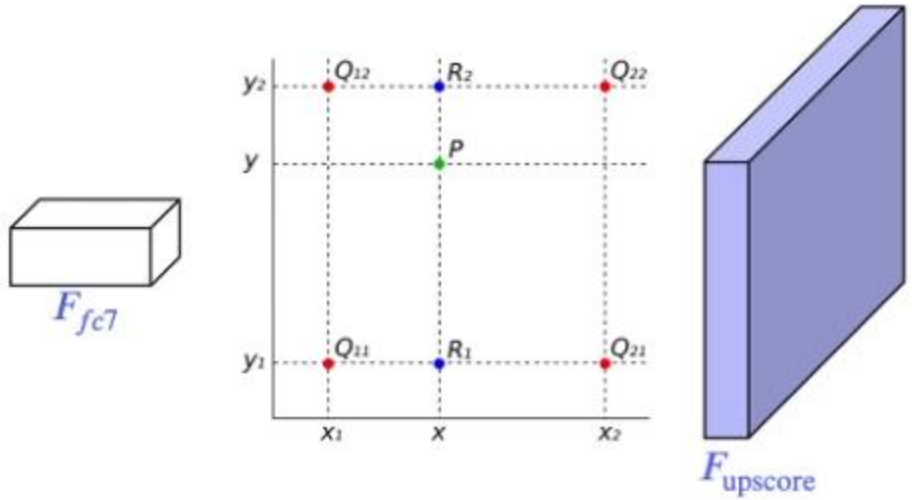
转置卷积Transpose Conv

上池化UnPooling

# 3.1.1 全卷积网络是怎么工作的?

## 如何使特征图变大?

### 1. 上采样 Up sampling




***给定*: Feature map**

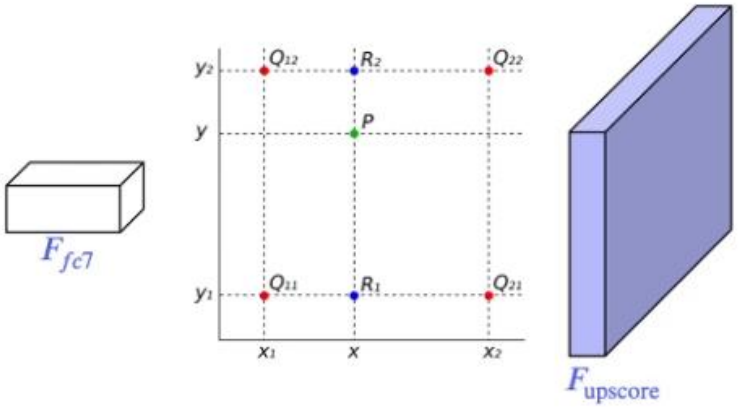
- 源：$F_{fc7}\ shape = [N \times C \times H \times W]$
- 目标：$F_{upscore}\ shape = [N \times C' \times H' \times W']$
- 升采样的过程：可以看作是的“图像” $F_{fc7}$到“图像” $F_{upscore}$的resize
- Resize的实现方法：双线性插值 bilinear Interpolation

# 3.1.1 全卷积网络是怎么工作的?

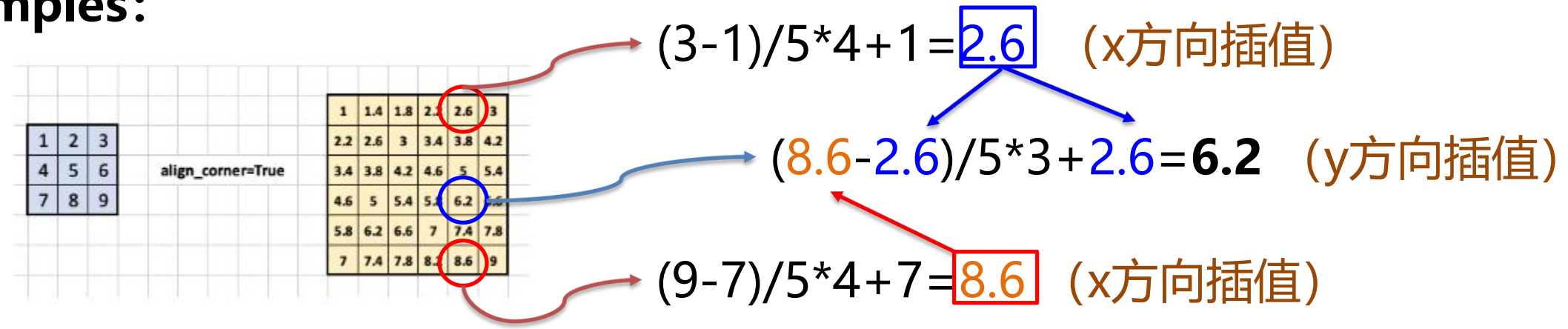
## 如何使特征图变大?

### 1. 上采样 Up sampling

- 任务：获得P点的像素值
- 步骤1：按x方向插值，获得R1和R2的值
- 步骤2：按y方向插值，获得P的值




- **Examples:**



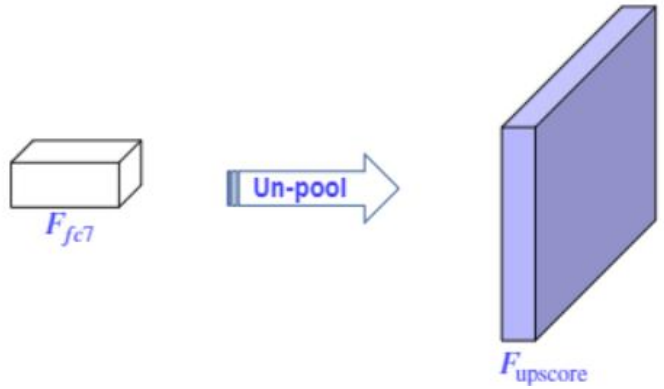

代码实现：Demo_Interpolation

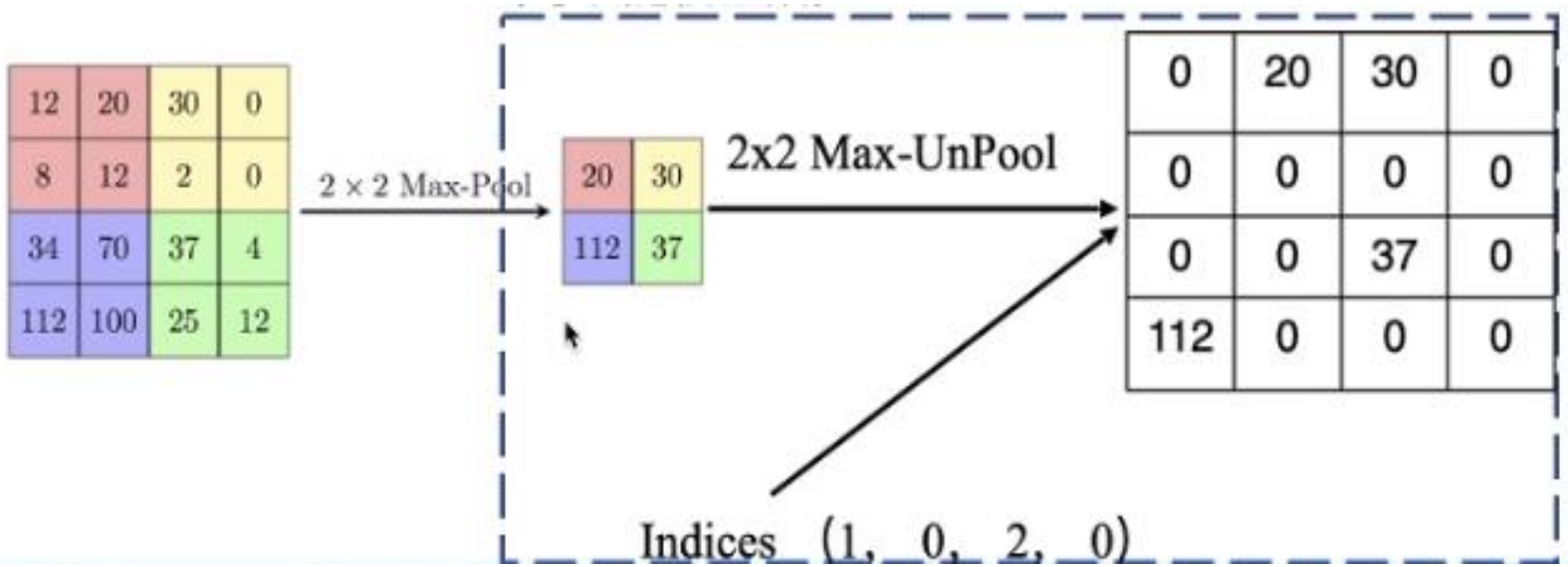
# 3.1.1 全卷积网络是怎么工作的?

## 如何使特征图变大?

### 2. 向上池化 Up-pooling

- Pooling的反向操作!
- 需要在pooling过程中记录pool的索引，并在对应位置补0



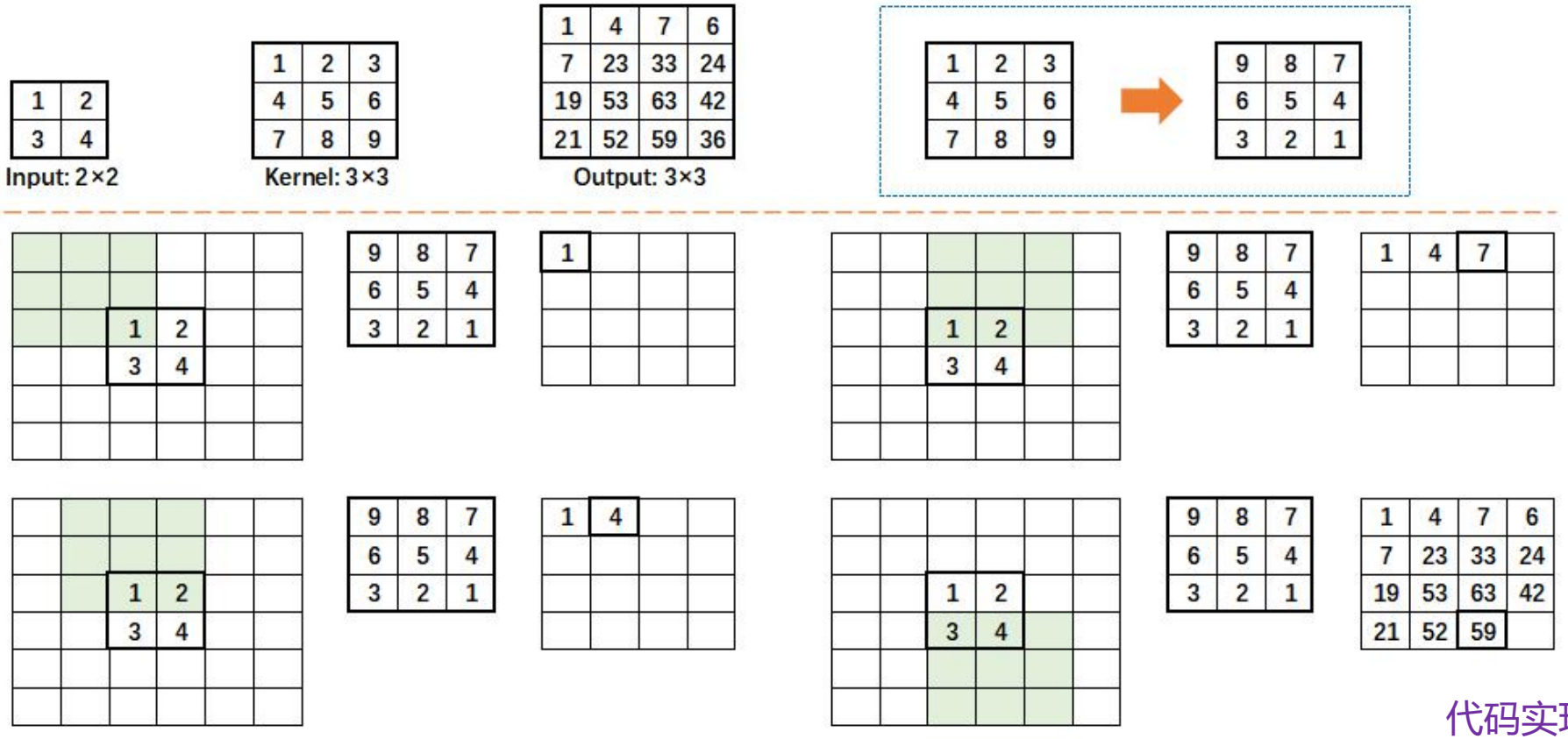



# 3.1.1 全卷积网络是怎么工作的?

## 如何使特征图变大?

### 3. 转置卷积 Transpose Conv

**基本算法：Kernel顺时针旋转180度 + Padding + Conv运算**




Kernel
顺时针旋转180度

（左右翻转+上下翻转）

代码实现：Demo_TransposeConv

# 3.1.1 全卷积网络是怎么工作的?

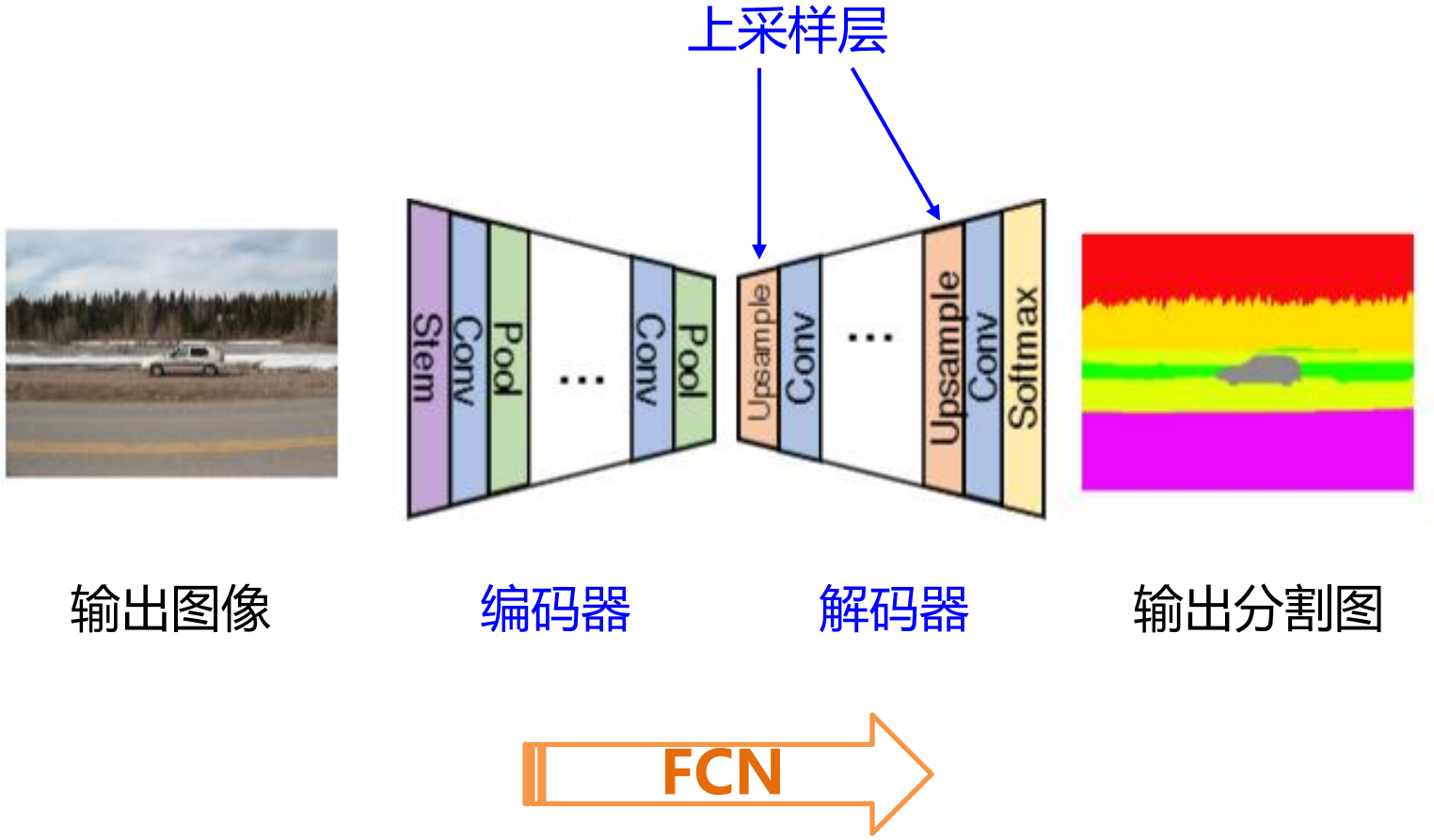
## 全卷积网络的工作原理

***FCN全卷积网络的体系结构***

- 编码器 —— 解码器

  Encoder —— Decoder

- FCN的基本模块
  - ✓ 卷积层 Convolution
  - ✓ 降采样层 Down-sampling
  - ✓ 上采样层 Up-sampling

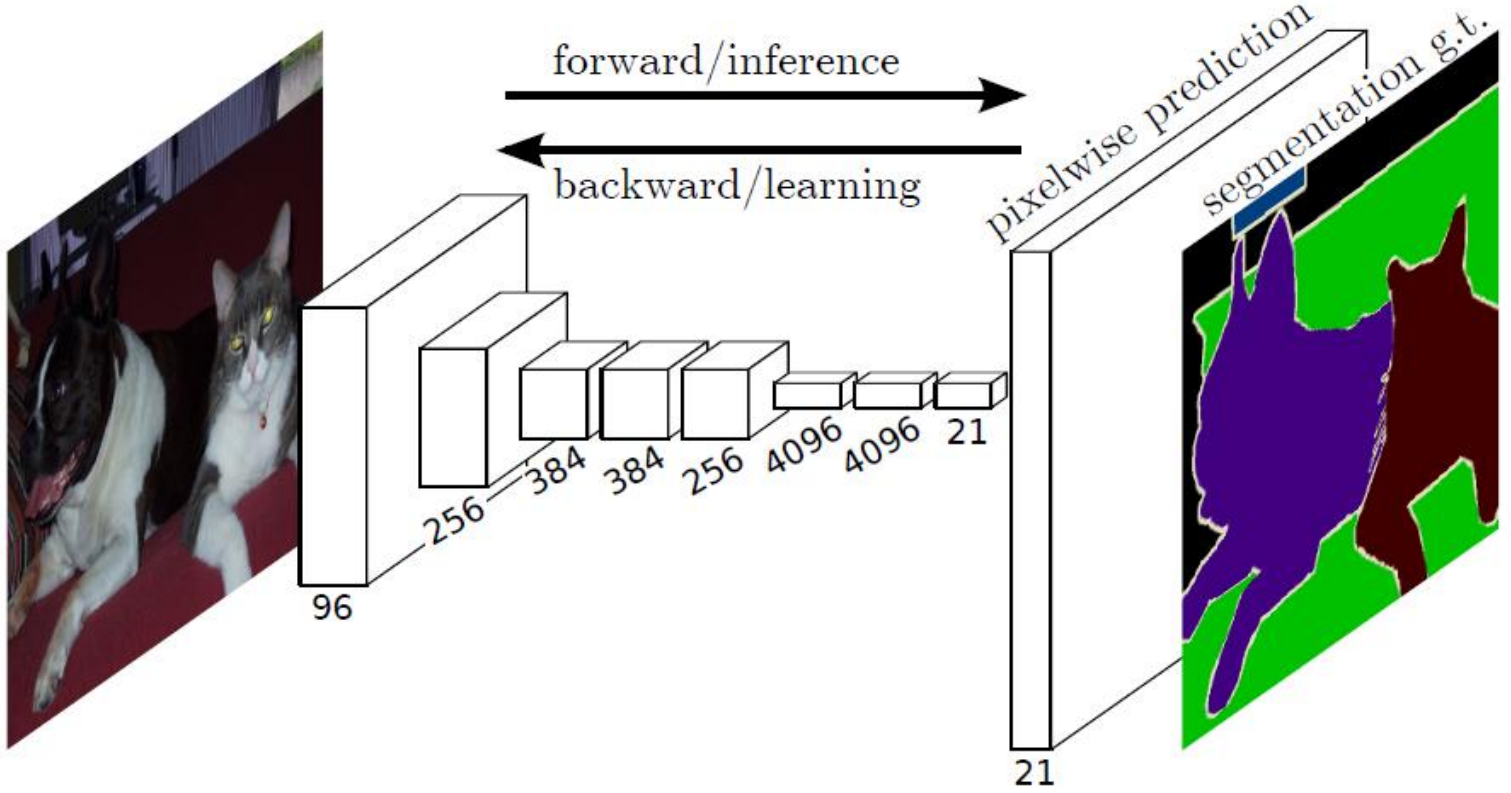


# 3.1.2 第一个全卷积网络——FCN全卷积网络

## 图像分割领域的里程碑——FCN全卷积网络

**全卷积网络（Fully Convolution network，FCN）**由UC Berkeley开发，它开启了基于卷积神经网络解决语义分割问题的先河。FCN完成了像素级端到端的语义分割框架，并且支持任意尺度的输入。

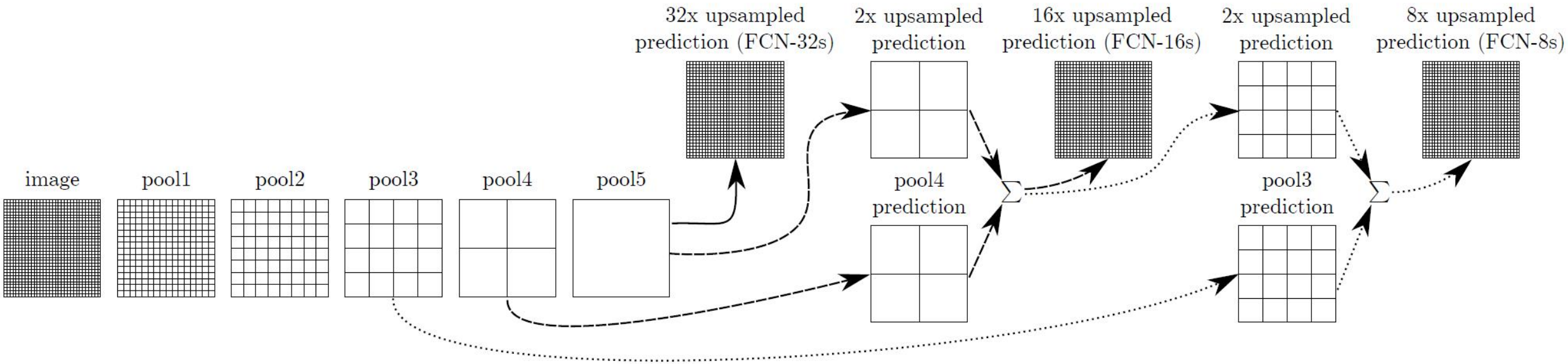
FCN在CNN(VGG-16)的基础上将全连接层**替换**为卷积层，输出空间映射而不是分类分数。这些映射由小步幅卷积上采样（反卷积、转置卷积）获得，用于产生密集的像素级预测。



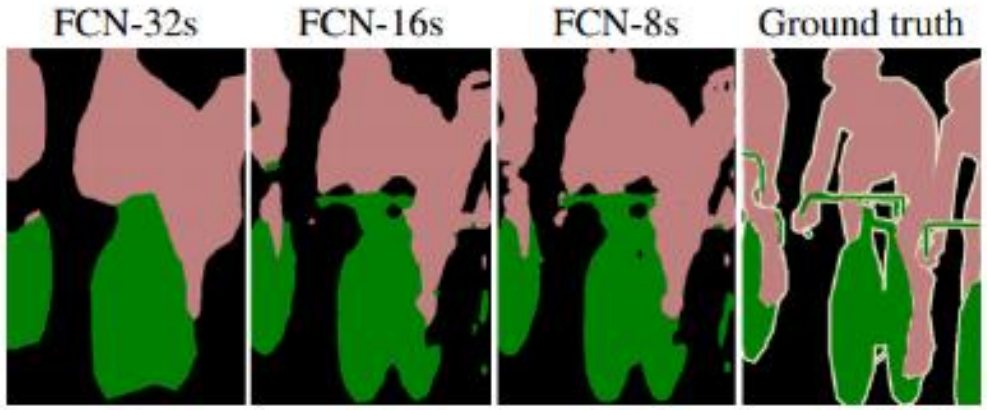

Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR2015

# 3.1.2 第一个全卷积网络——FCN全卷积网络

## FCN全卷积网络的基本结构



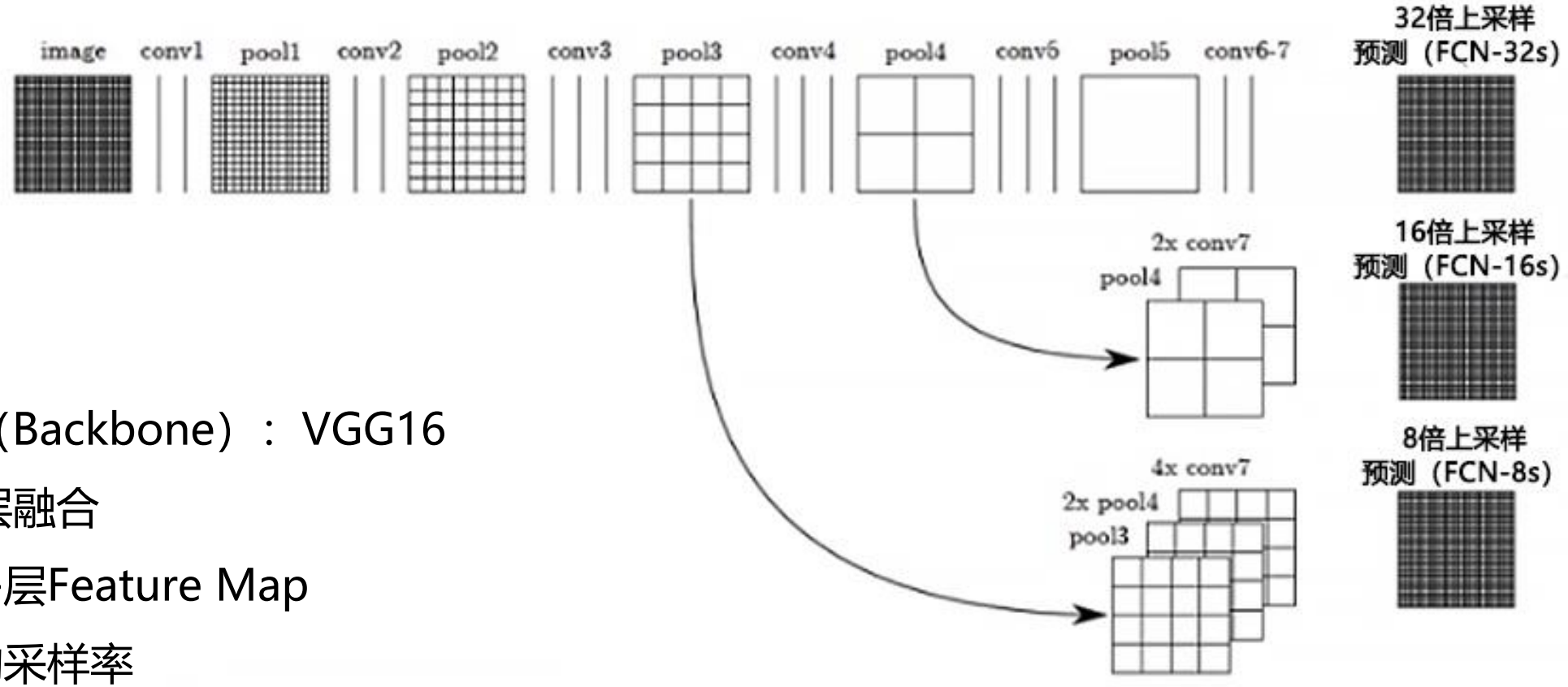

池化层和带跨步的卷积会导致特征图的分辨率不断下降，FCN利用**双线性插值**将响应张量的长宽**上采样**到原图大小。多尺度让输出的预测更加精细，低层更关注细节，高层更关注整体




Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR2015

# 3.1.2 第一个全卷积网络——FCN全卷积网络

## FCN全卷积网络的基本结构

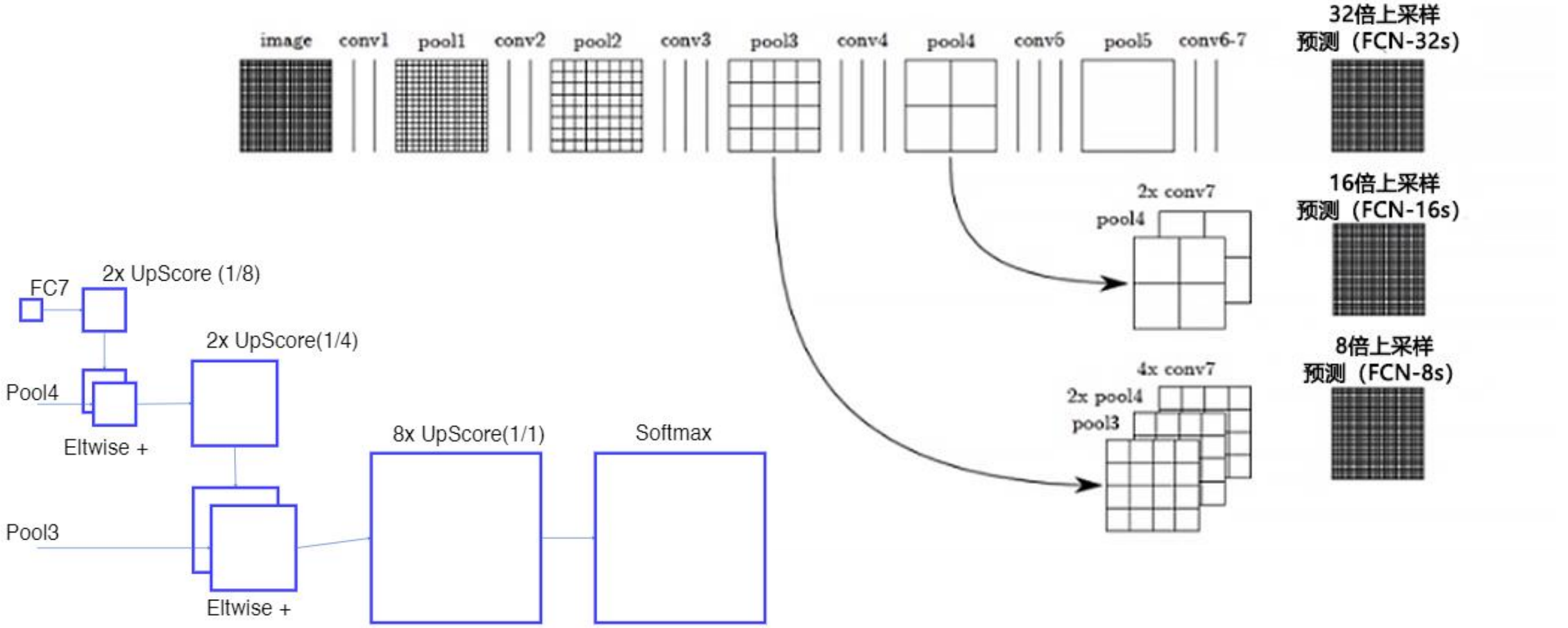



- 骨干网络（Backbone）：VGG16
- FCN的多层融合
  - ✓ 集成多层Feature Map
  - ✓ 不同的采样率
  - ✓ 组合方式：Element-wise Add

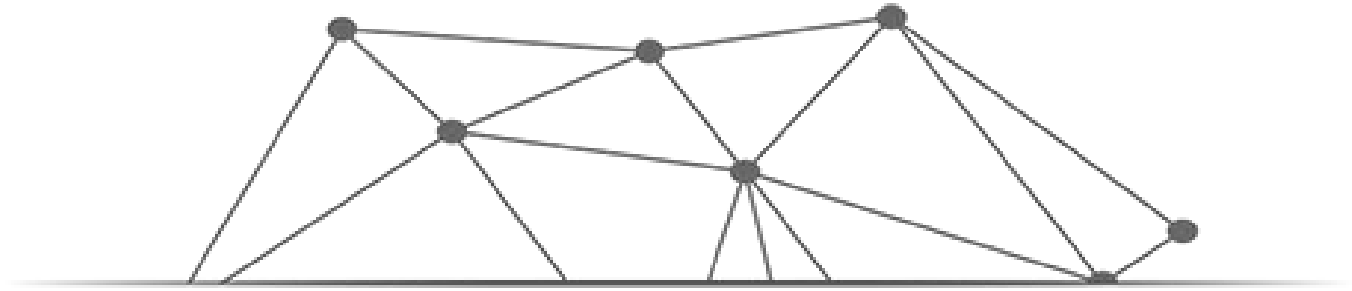
Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR2015

# 3.1.2 第一个全卷积网络——FCN全卷积网络

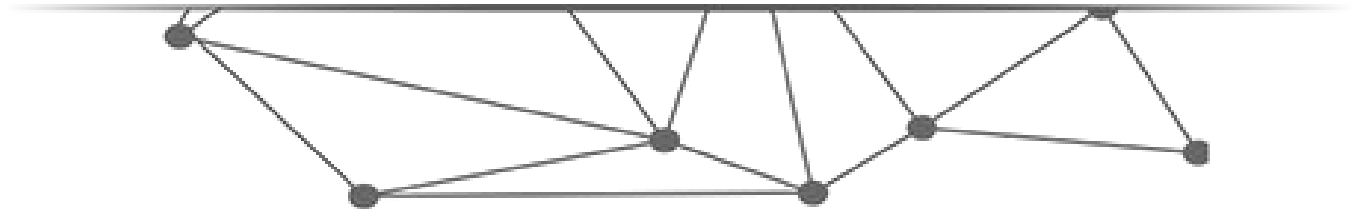
## FCN全卷积网络的基本结构

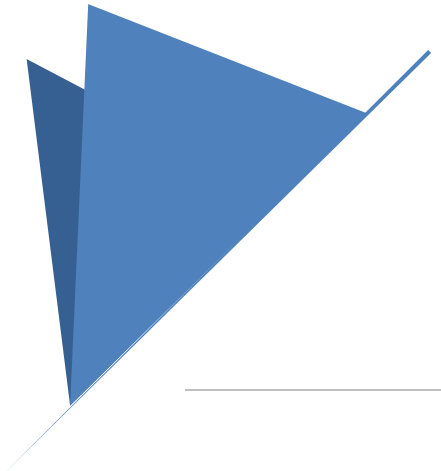



Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR2015



# 课堂互动 13.3.3





# U-Net/PSPNet/SegNet

# 3.2 经典图像分割模型——U-Net/PSPNet模型

## 本节内容

01 **Recap of FCN**
回顾FCN分割网络的基本结构；FCN网络的优缺点

02 **What is U-Net? And how to build a U-Net?**
U-Net网络的基本结构和原理; 如何实现一个U-Net网络，如何对U-Net进行扩展

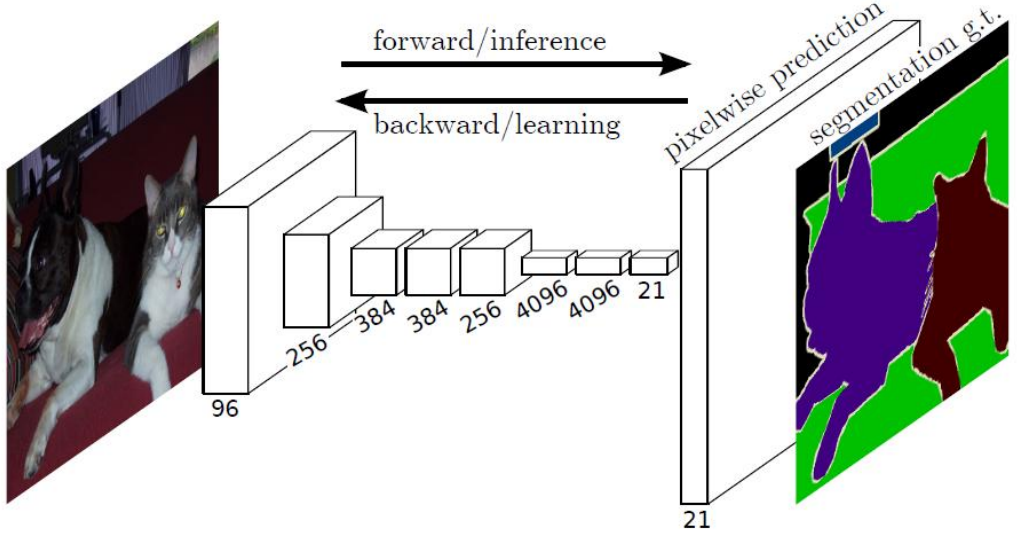
03 **What is PSPNet? And how to build PSPNet?**
PSP网络的基本结构和特点；PSP模块的具体形式；Dilated Conv的基本原理

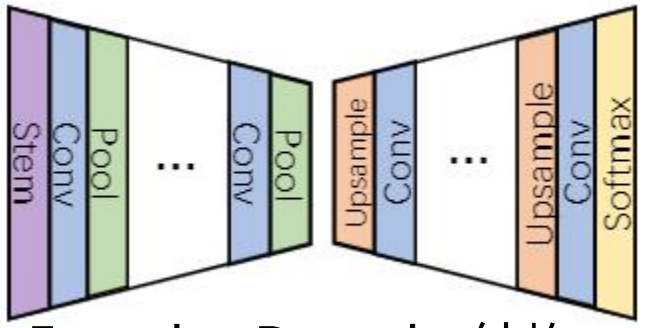
04 **What is SegNet? And what is its characteristics?**
SegNet网络的结构和特点是什么?

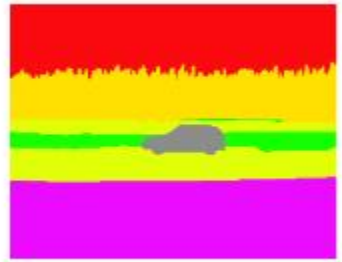
# 3.2.1 Recap：全卷积网络FCN

## Recap: FCN的基本结构

- **全卷积网络**
  - ✓ 没有全连接层，只有卷积层
  - ✓ 语义分割≈像素级分类
- **编码器-解码器结构**
- **基本组成模块**
  - ✓ 卷积层
  - ✓ 下采样层
  - ✓ 上采样层



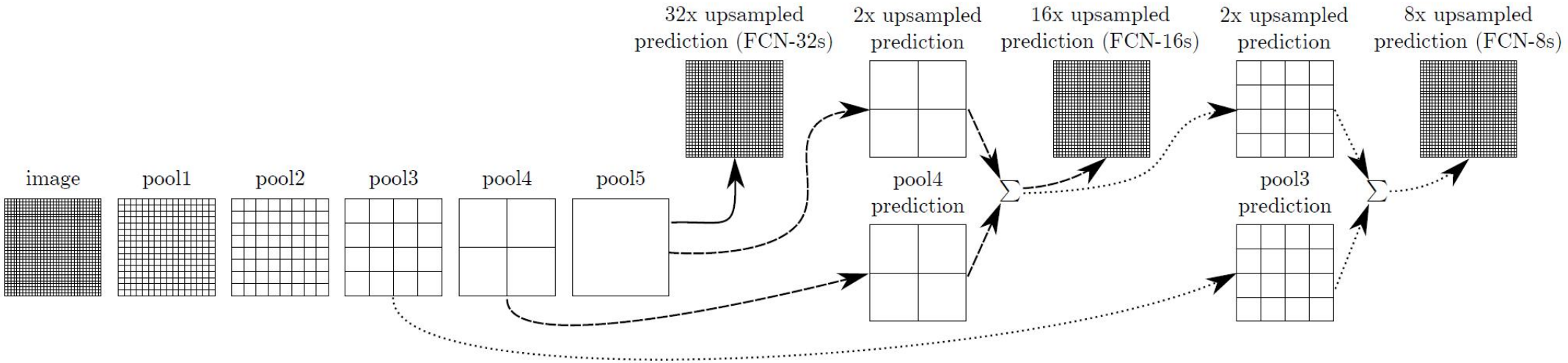



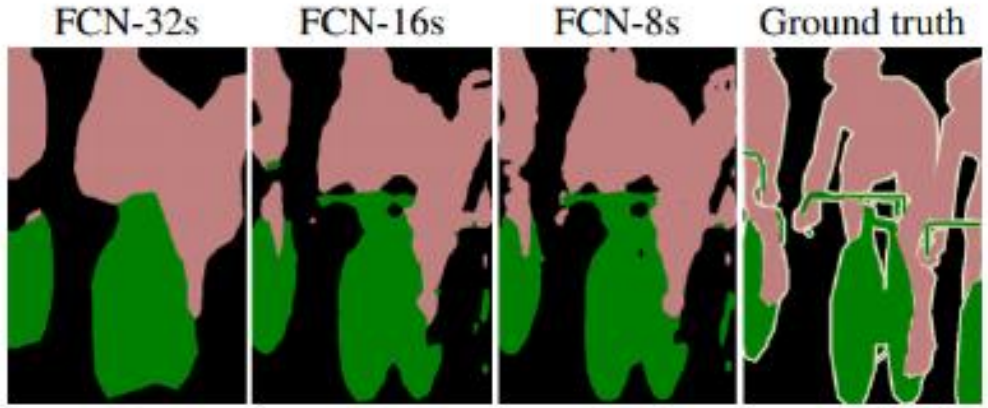
Encoder-Decoder结构

# 3.2.1 Recap：全卷积网络FCN

## Recap: FCN的优缺点

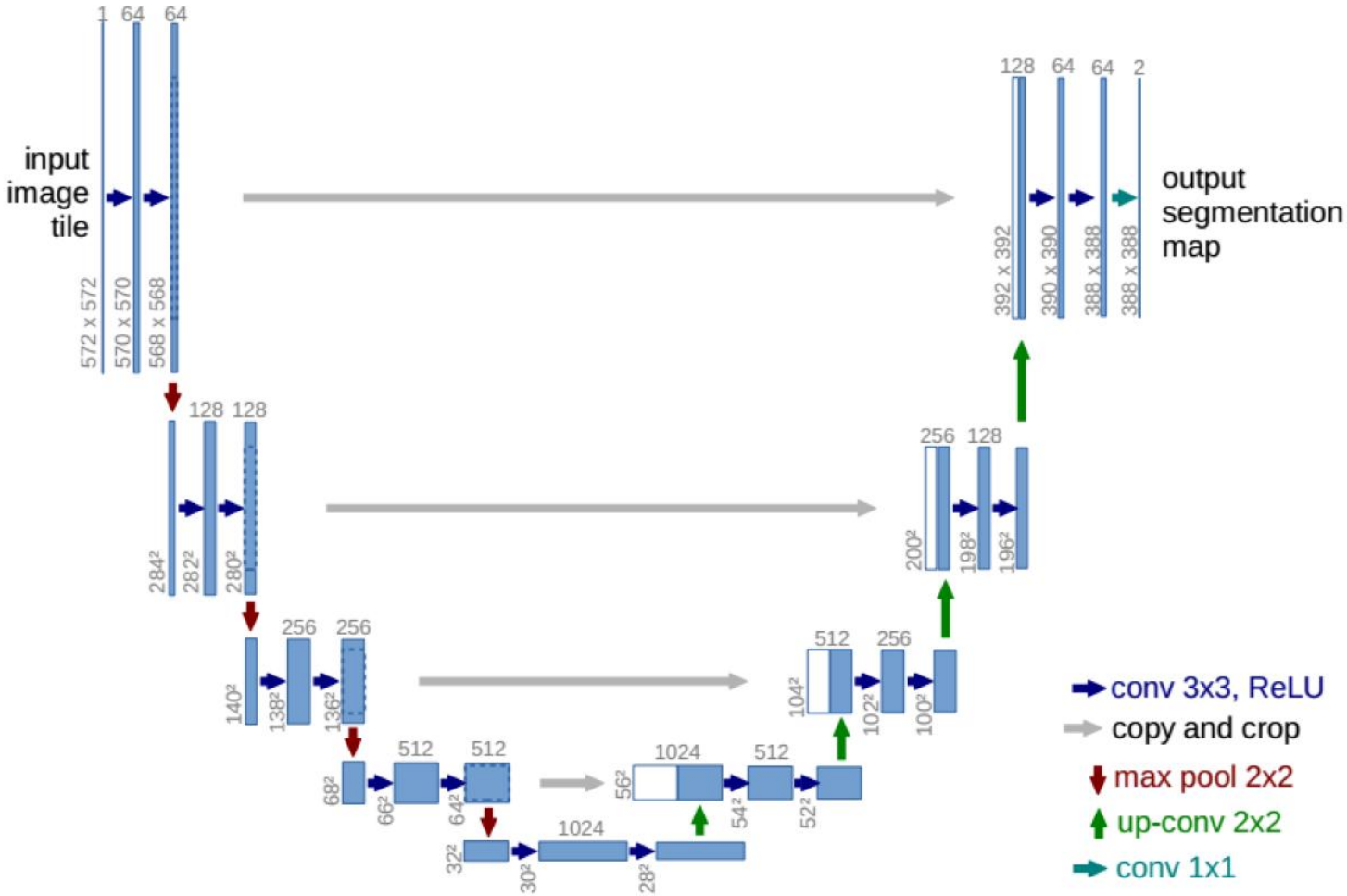



- **优点**
  - ✓ 支持任意尺寸输入（全卷积）
  - ✓ 效率高（相较过去）
  - ✓ 组合了深层和浅层信息
- **缺点**
  - ✓ 分割结果不够精细
  - ✓ 没有考虑上下文信息 (≈ 左顾右盼 ≈ 看看旁边有什么)



# 3.2.2 医学影像分割的重要网络U-Net

## U-Net的框架结构




**U-Net也是解码器变种的一个经典网络**

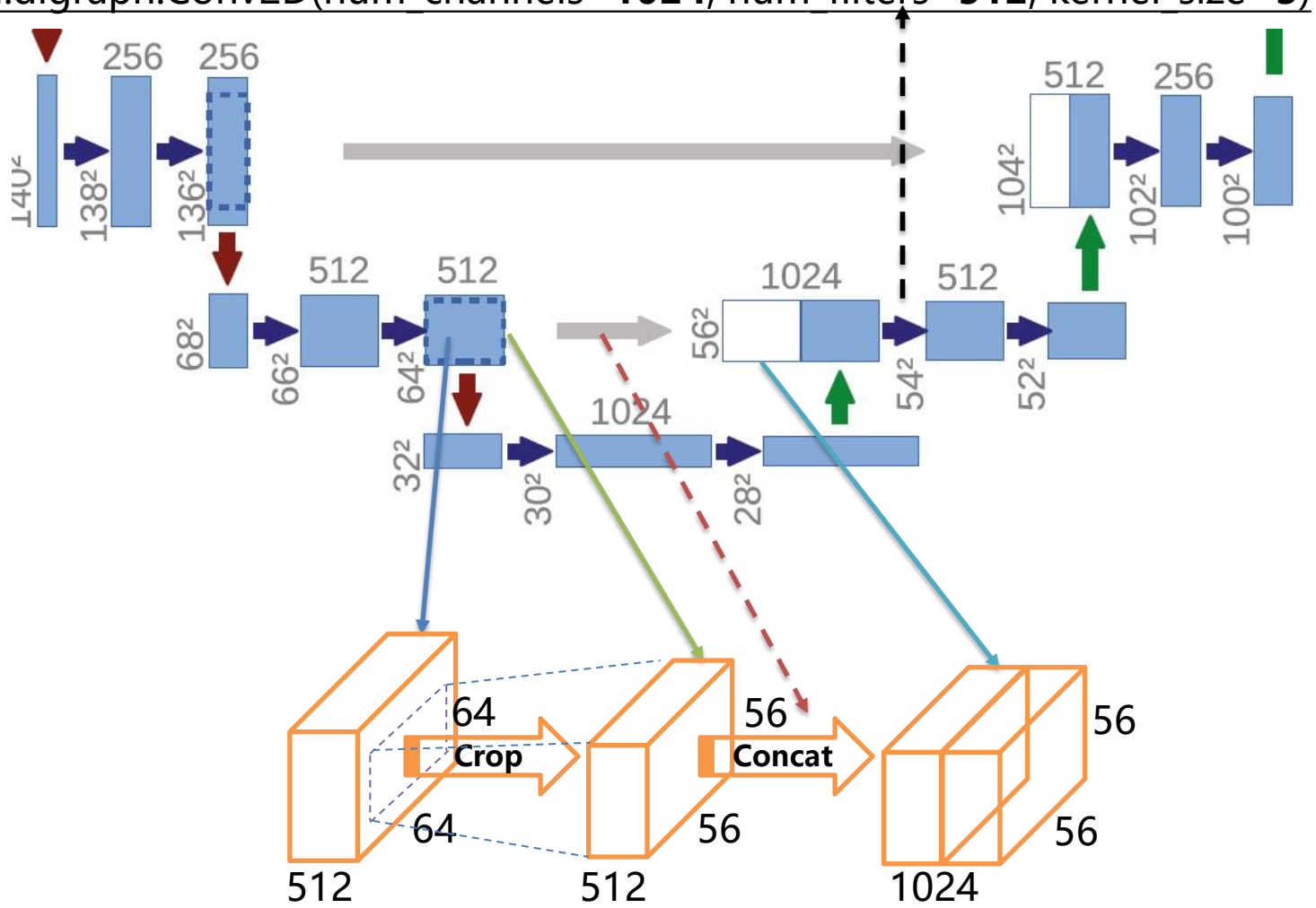
- 解码器采用了反卷积滤波器进行上采样，并增加对应编码器的特征。
- 在解码器部分，并没有进行单纯的上采样操作，每次上采样后伴随着两次卷积。
- 与SegNet不同，医学图像对于实时性的要求不高，但对图像的**精度要求苛刻**。

Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015

# 3.2.2 医学影像分割的重要网络U-Net

## U-Net的框架结构

Fluid.digraph.Conv2D(num_channels=**1024**, num_filters=**512**, kernel_size=**3**)




***U-Net的Copy and Crop***

- Skip结合方式：Concatenation
- 如果尺寸 发生变化：Crop
- Concat之后：Conv

是否能将 64 Crop to 56，
变为56 padding to 64?

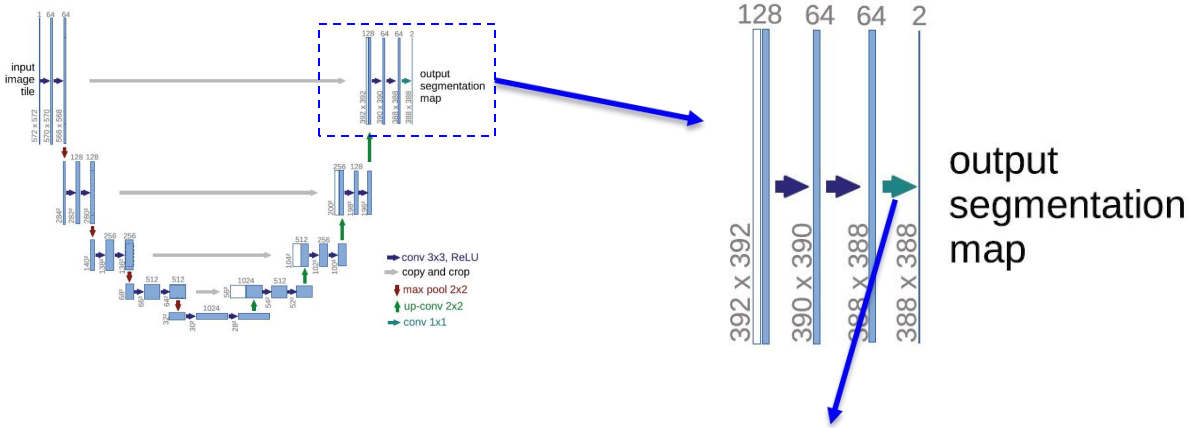
Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015

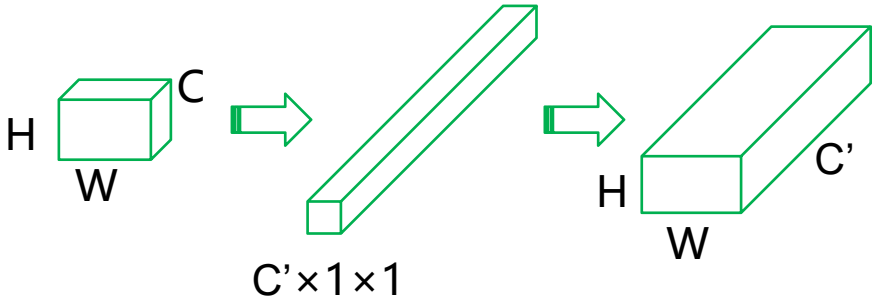
# 3.2.2 医学影像分割的重要网络U-Net

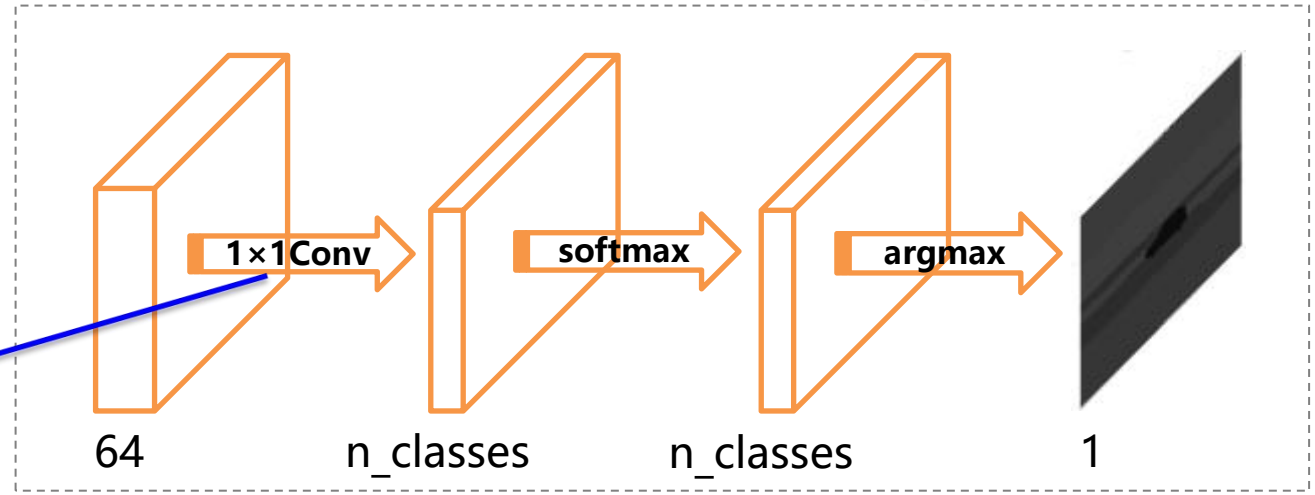
## U-Net的框架结构

### *U-Net的输出层*

1. 使用**1×1卷积**将最后一个卷积层的维度转换为类别数;
2. 使用softmax进行分类
3. 使用argmax函数获得概率最大类别的索引



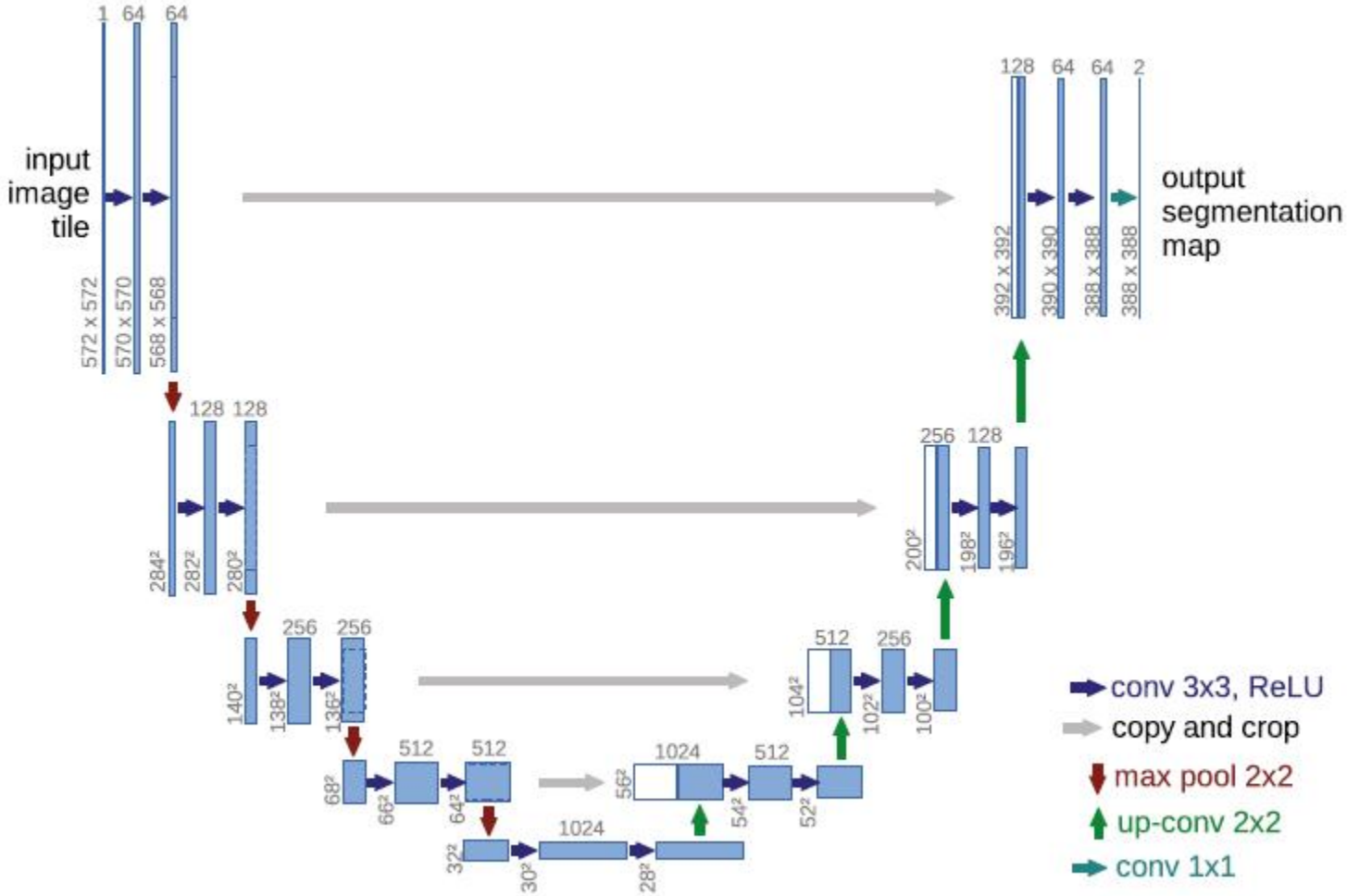

Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015

# 3.2.2 医学影像分割的重要网络U-Net

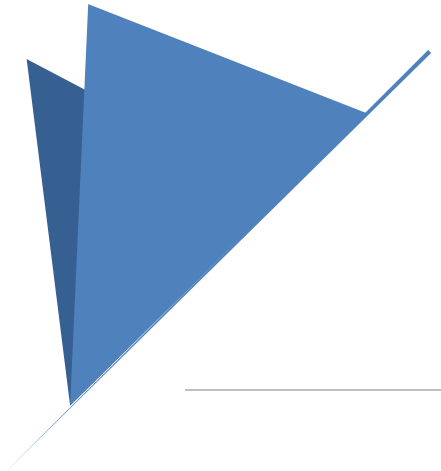
## U-Net的框架结构

***U-Net的主要操作：***

1. Conv 3×3, (with bn, relu)
2. Pool 2D
3. Transpose Conv 2×2 （增大2x）
4. Crop, Concat
5. Conv 1×1 （分割类别）
6. SoftMax, argmax, squeeze




Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015



# PSPNet

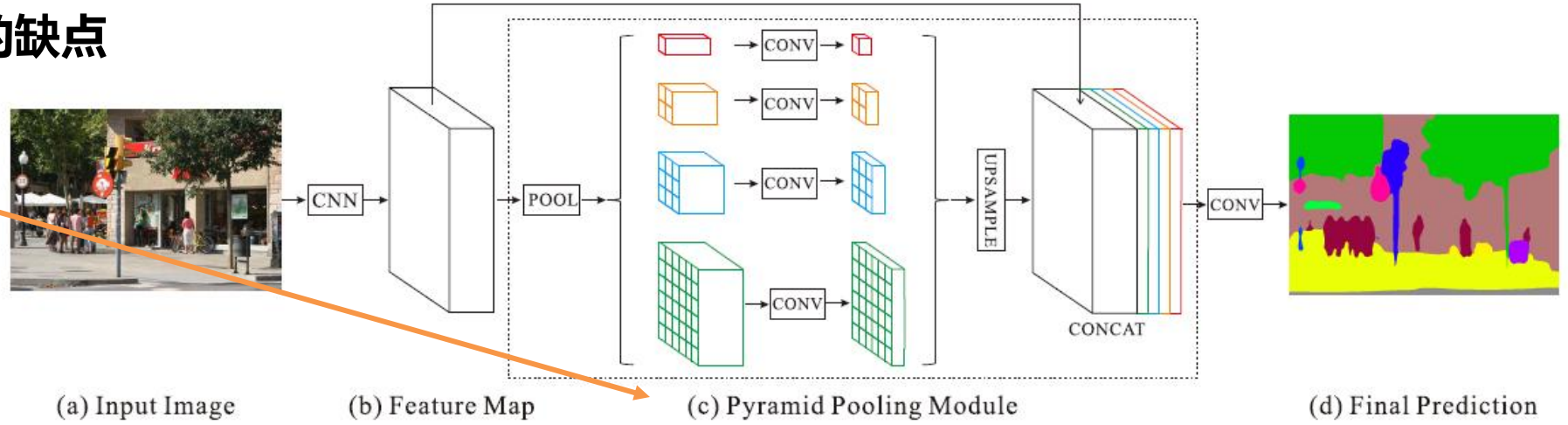
# 3.3 PSP分割网络

## 什么是PSP分割网络?

***What does PSPNet look like?***

- **PSP网络：Pyramid Scene Parsing Network**
  - ✓ *Pyramid: 空间金字塔结构*
  - ✓ *Scene Parsing：场景解析，语义分割(Semantic Segmentation)的一种*

- **PSPNet的提出->FCN的缺点**
  - ✓ *分割结果不够精细*
  - ✓ *没有考虑上下文信息*



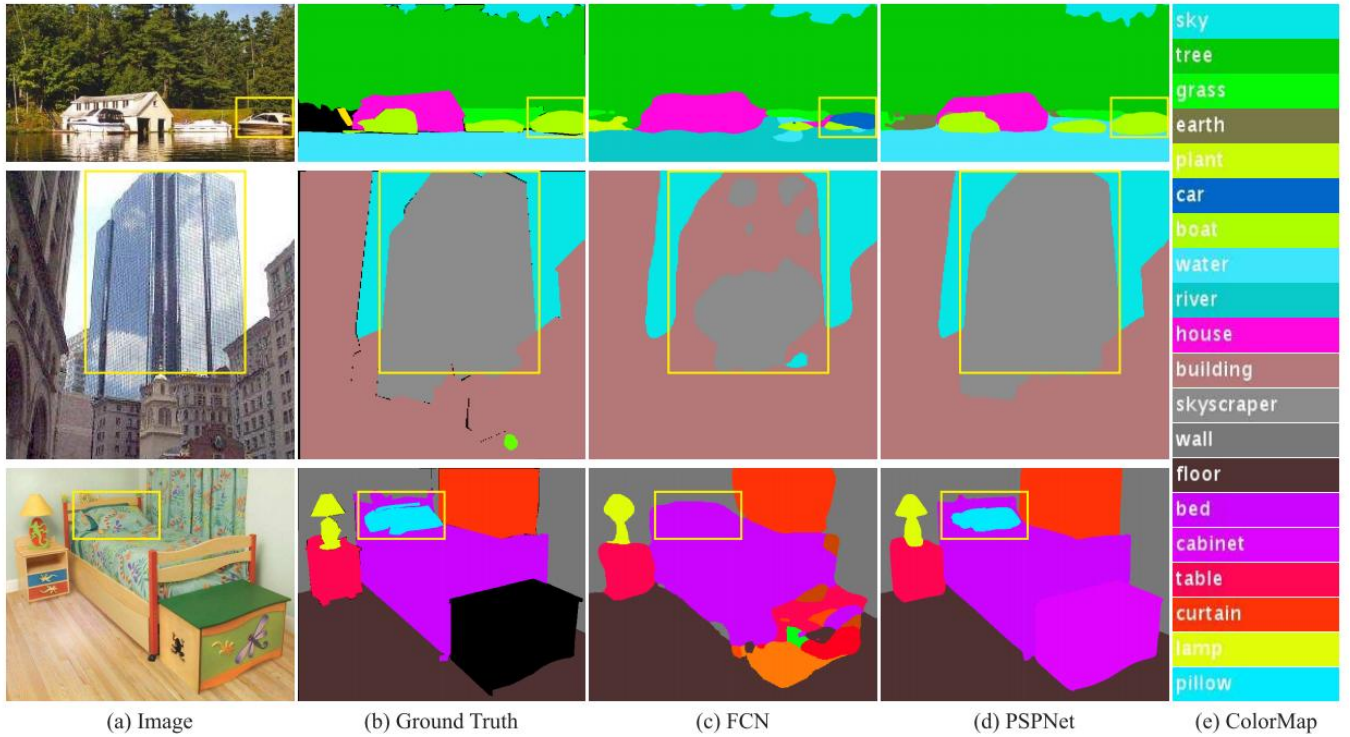

Olaf Ronneberger, Philipp Fischer, Thomas Brox. Pyramid Scene Parsing Network. CVPR 2017

# 3.3 PSP分割网络

## 什么是上下文信息?

***What is context information?***

*图中类别boat区域和类别car区域的外观相似，很容易被误认，原因是模型只有local局部信息，所以boat容易被识别为car；building 和 skyscraper 也存在类似的情况。这种现象被称为缺少上下文信息。*




(a) Image (b) Ground Truth (c) FCN (d) PSPNet (e) ColorMap

- ***解决办法:***

充分利用全局信息（Global Information），即增大感受野是比较好的方法。

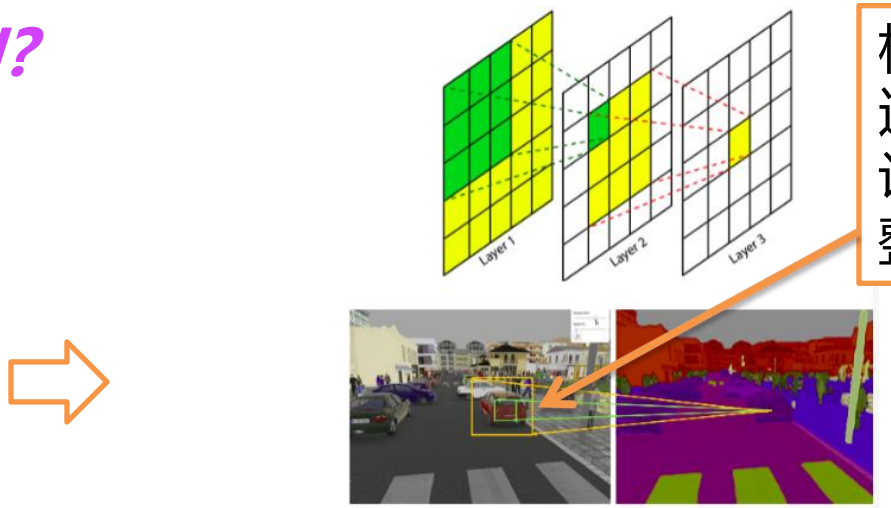
Olaf Ronneberger, Philipp Fischer, Thomas Brox. Pyramid Scene Parsing Network. CVPR 2017

# 3.3 PSP分割网络

## 如何获取上下文信息

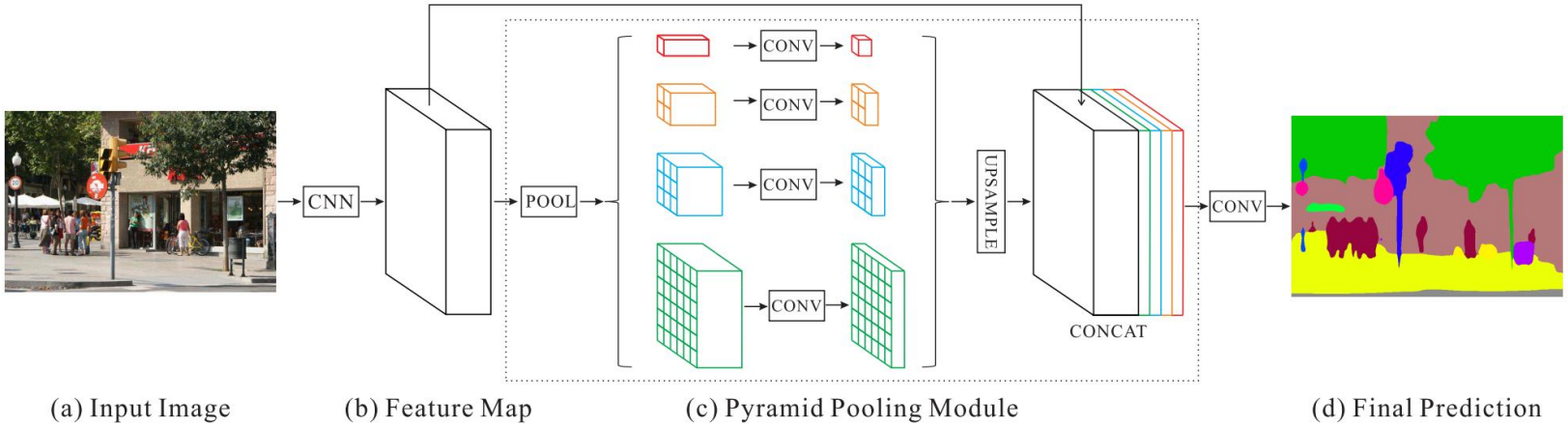
***What does Receptive Field mean in CNN?***

- ***感受野（Receptive Field）***
  - ✓ *用于产生特征的输入图像中的区域大小*
  - ✓ *感受野只针对局部操作，如Conv，Pooling*




标准感受野过小，无法识别目标的整体信息。

***PSPNet通过增大感受野来实现上下文信息的获取，更进一步，特征金字塔(Feature Pyramid)是实现感受野扩大的有效方法***



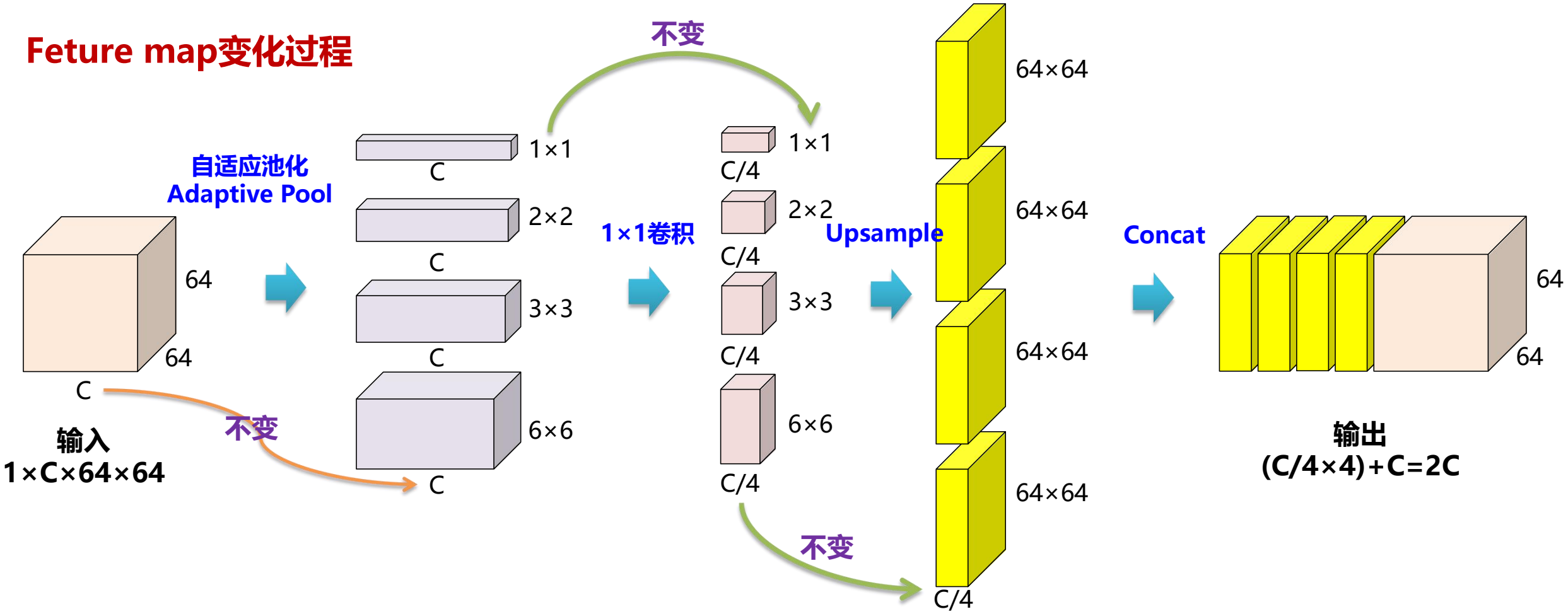

(a) Input Image (b) Feature Map (c) Pyramid Pooling Module (d) Final Prediction

Olaf Ronneberger, Philipp Fischer, Thomas Brox. Pyramid Scene Parsing Network. CVPR 2017

# 3.3 PSP分割网络

## Pyramid Pooling模块

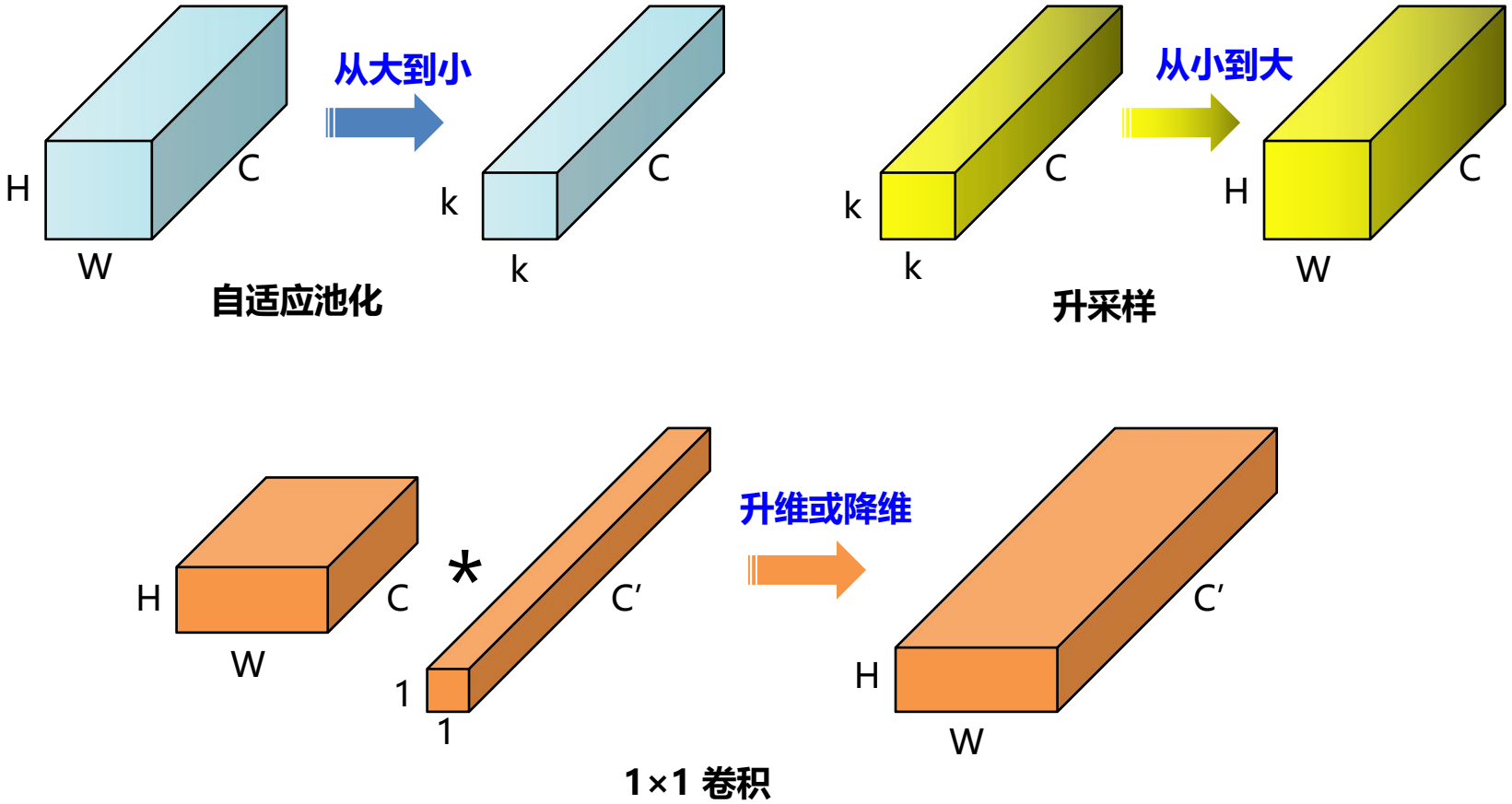
*What is inside Pyramid Pooling Module?* ***基本结构***



# 3.3 PSP分割网络

## Pyramid Pooling模块

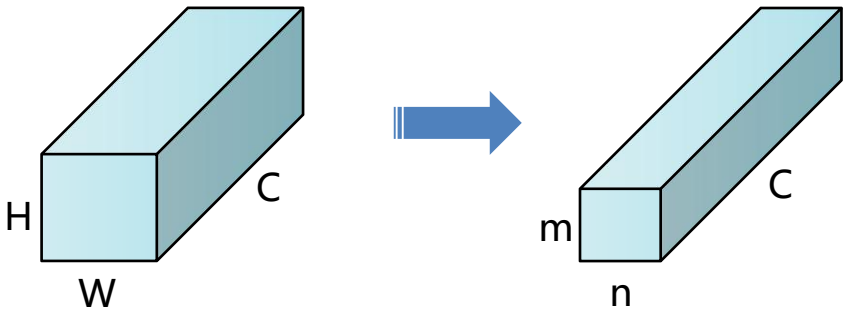
***What is inside Pyramid Pooling Module?*** ***重要操作***

# 3.3 PSP分割网络

## Pyramid Pooling模块

### *自适应池化（Adaptive Pool）*




**h**start = floor(i * H / m) =10/5=2
**h**end = ceil((i + 1) * H / m) =2*10/5=4
**w**start = floor(j * W / n) = 2*10/3=6
**w**end = ceil((j + 1) * W / n) = 3*10/3=10
output[:, :, i, j] = max(input[:, :, **h**start: **h**end, **w**start: **w**end]) =max(2:4,6:10)=0.63

W=10

H=10

```
[[0.38 0.05 0.78 0.98 0.73 0.92 0.92 0.28 0.95 0.26]
 [0.77 0.67 0.01 0.94 0.3  0.84 0.77 0.89 0.19 0.72]
 [0.94 0.31 0.2  0.57 0.35 0.45 0.33 0.62 0.25 0.27]
 [0.59 0.16 0.58 0.7  0.62 0.19 0.6  0.28 0.44 0.63]
 [0.07 0.57 0.29 0.82 0.63 0.06 0.91 0.77 0.92 0.93]
 [0.16 0.86 0.72 0.88 0.79 0.99 0.43 0.9  0.9  0.6 ]
 [0.09 0.45 0.61 0.12 0.59 0.77 0.53 0.04 0.03 0.55]
 [0.76 0.22 0.78 0.27 0.84 0.94 0.02 0.06 0.98 0.83]
 [0.88 0.97 0.1  0.96 0.86 0.53 0.94 0.86 0.84 0.19]
 [0.55 0.64 0.19 0.78 0.99 0.13 0.84 0.53 0.78 0.26]]
```

```
Tensor(shape=[1, 1, 5, 3], dtype=float64, place=CPU
       [[[[0.98000000, 0.98000000, 0.95000000],
          [0.94000000, 0.70000000, 0.63000000],
          [0.88000000, 0.99000000, 0.93000000],
          [0.78000000, 0.94000000, 0.98000000],
          [0.97000000, 0.99000000, 0.94000000]]]])
```

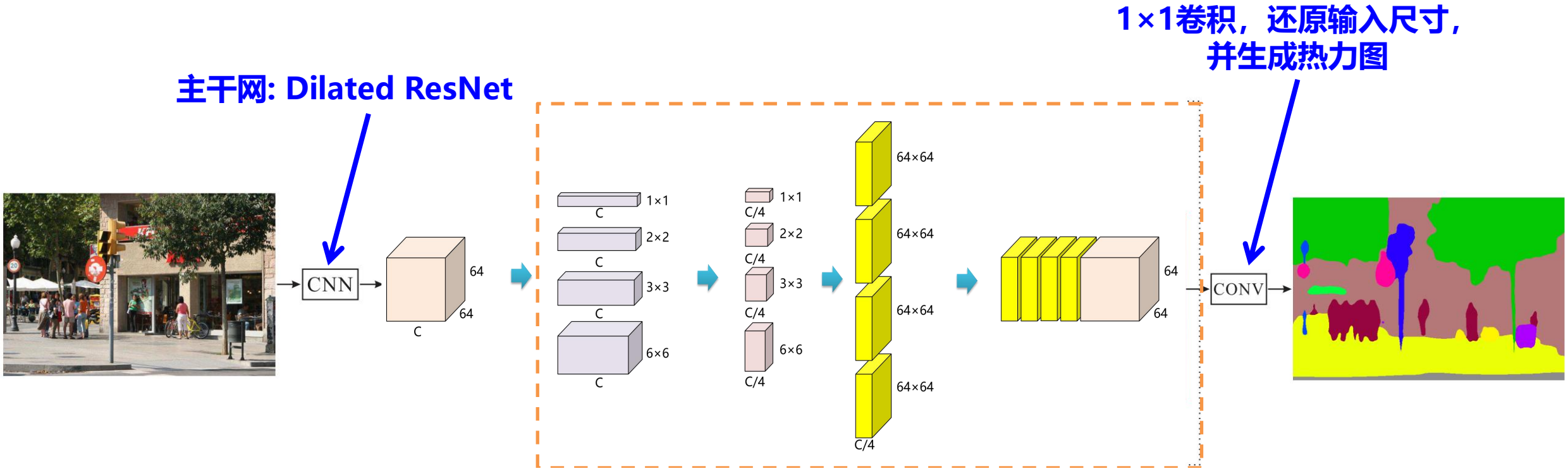
i=1, j=2

m=5

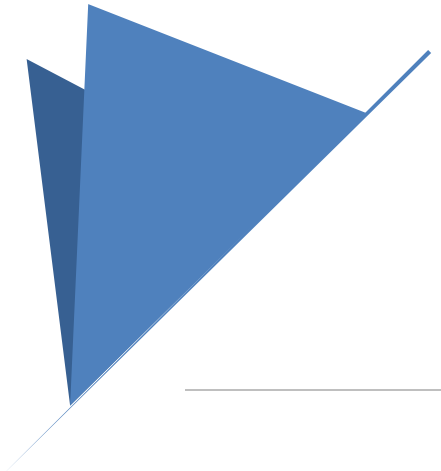
n=3

# 3.3 PSP分割网络

## PSP网络体系结构图



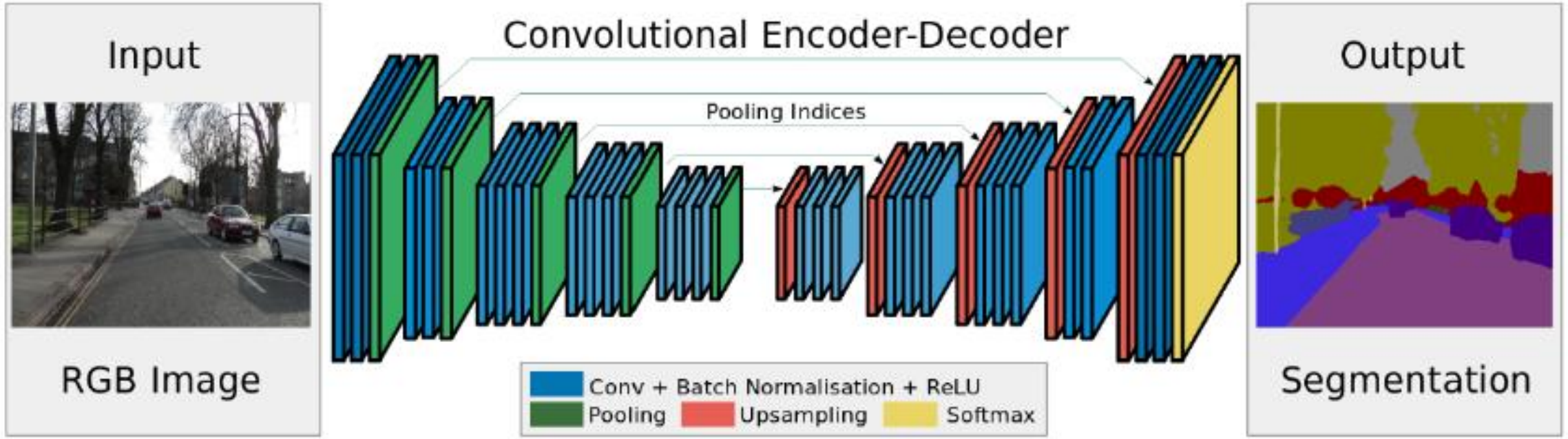

(a) 输入图像　(b) 特征图　(c) 金字塔池化模块　(d) 最终预测



# SegNet

# 3.4 实时分割网络SegNet

## SegNet的拓扑结构图




SegNet使用了经典的**编码器-解码器**(encoder-decoder)结构。SegNet专注于道路场景下的语义分割，由于自动驾驶对于分割速度的要求非常高，SegNet在解码器部分进行了优化，使得其能够满足实时语义分割的需求。

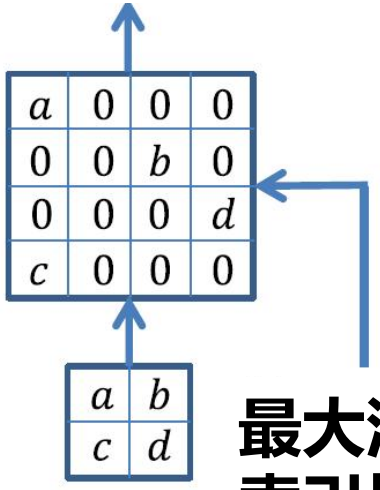
Vijay Badrinarayanan , Alex Kendall, Roberto Cipolla. SegNet: A Deep Convolutinal Encoder-Decoder Architecture for Image Segmentation. IEEE Transactions on PAMI, 2017

# 3.4 实时分割网络SegNet

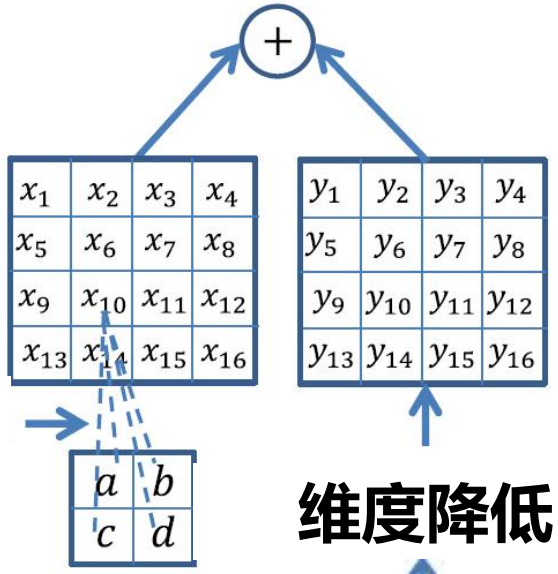
## SegNet的核心算法

使用可训练的解码器执行转置卷积

SegNet




最大池化
索引地址




FCN

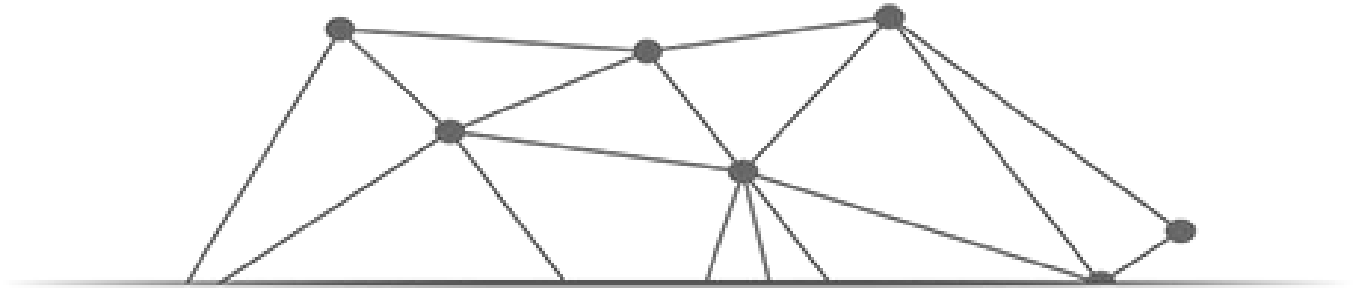
转置卷积
升采样

维度降低

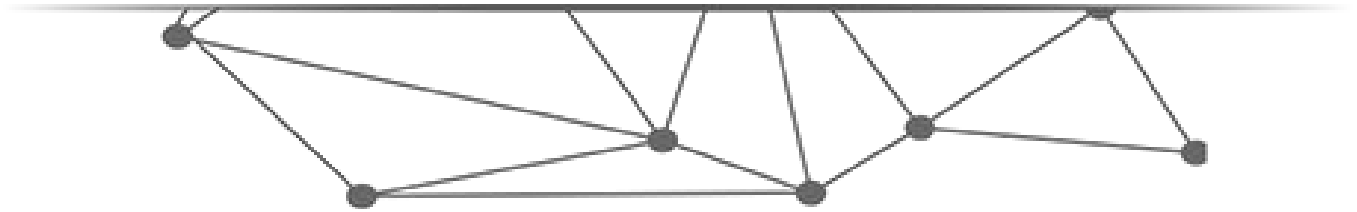
编码器特征图

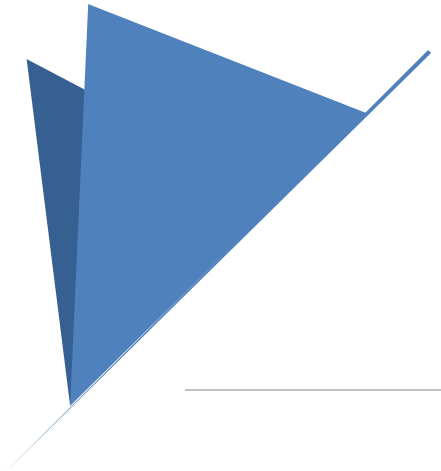
- 池化运算时，SegNet使用对应编码器记录的位置信息进行上采样，其他位置置为零。这样的方法加快了网络的运行速度，满足了实时分割的需求。
- FCN使用反卷积滤波器进行上采样，然后与对应编码器的特征映射相加。

Vijay Badrinarayanan, Alex Kendall, Roberto Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. IEEE Transactions on PAMI, 2017



# 课堂互动 13.3.4





# DeepLab 系列模型

# 3.5 经典图像分割模型——DeepLab系列分割模型

## 本节内容

01 **DeepLab的核心技术**

简要介绍 DeepLab涉及的一些核心技术，包括：

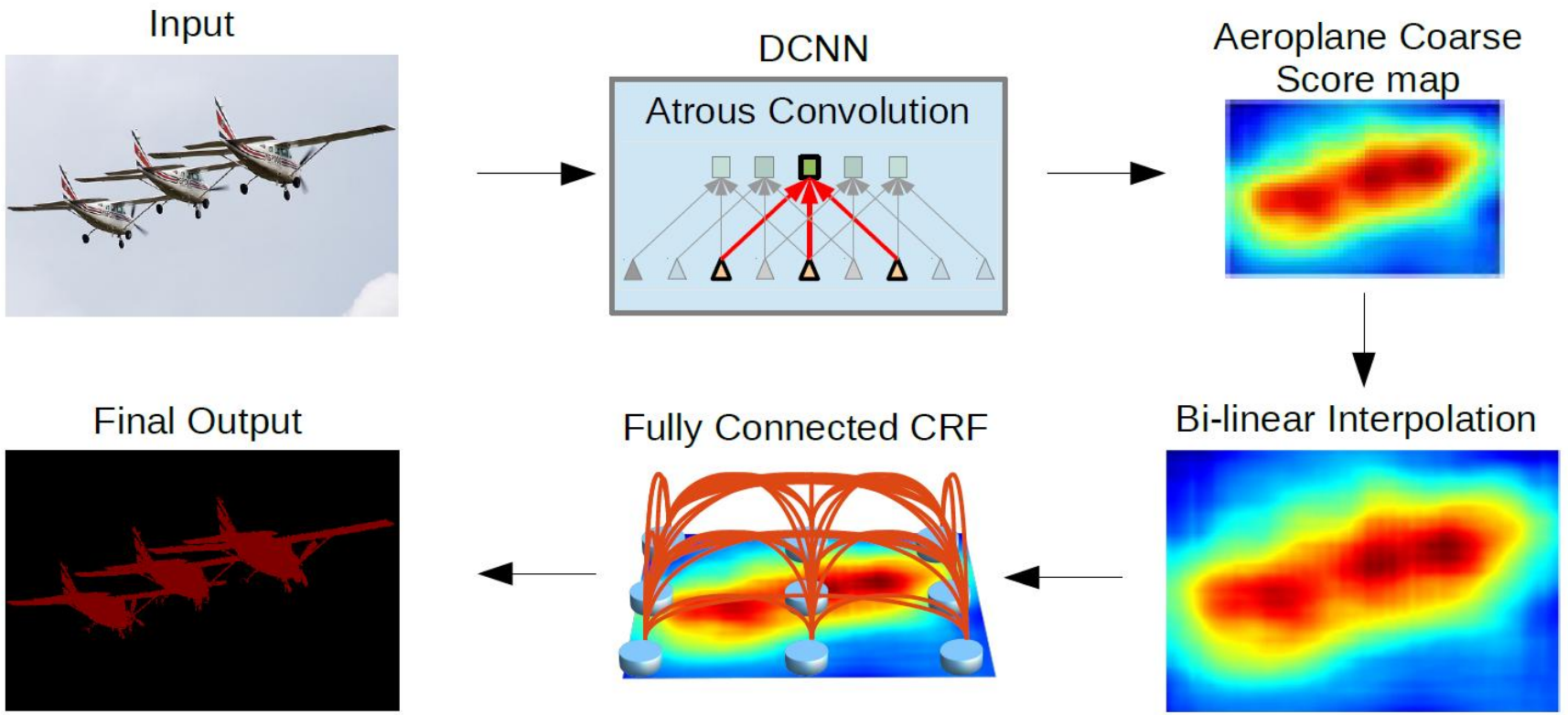
- Dilated Conv：空洞卷积
- ASPP：空洞空间金字塔池化
- Full-connected CRF：全连接条件随机场
- MultiScale-Train：多尺度训练

02 **What is DeepLab?**

DeepLab系列 (v1, v2, v3, v3+) 分割网络的基本结构、关键技术和原理

# 3.5.1 DeepLab的核心技术

## 条件随机场（Conditional Random Field, CRF）



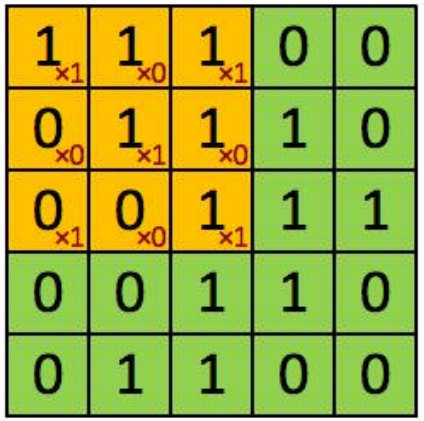

**CRF**是一种后处理技术，可以实现对分割网络的精细调整。

但CRF的计算量较大，花费时间较长，在后续的DeepLab v2及之后的版本被Atrous空间金字塔所取代。

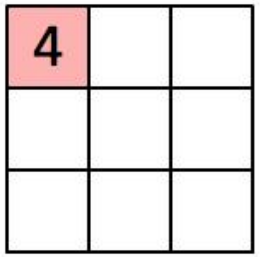
**注意：在DeepLab中，使用的是网络实现CRF，即Fully-Connected CRF**

# 3.5.1 DeepLab的核心技术

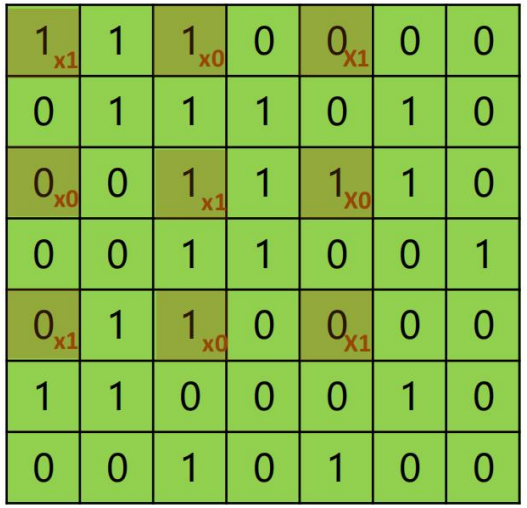
## 空洞卷积（Dilated Convolutions）



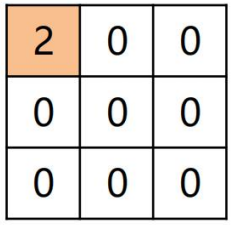

图像




卷积特征



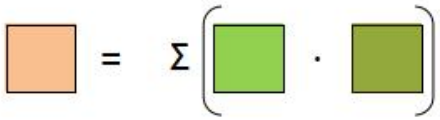

图像




(空洞)
卷积特征

**空洞卷积的关键点**

- ✓ 将Kernel扩大并填0
- ✓ 在FeatureMap间进行卷积运算




**空洞卷积**又称为**扩张卷积**（A'trous Conv, Dilated Conv），根据扩张填充尺寸，为空元素位置补零。这种卷积核可以指数级地扩大感受野而不丢失分辨率，这意味着空洞卷积可以在任意分辨率图片上高效地提取密集特征。

# 3.5.1 DeepLab的核心技术

## 空洞卷积（Dilated Convolutions）

- *输出特征图的尺寸*

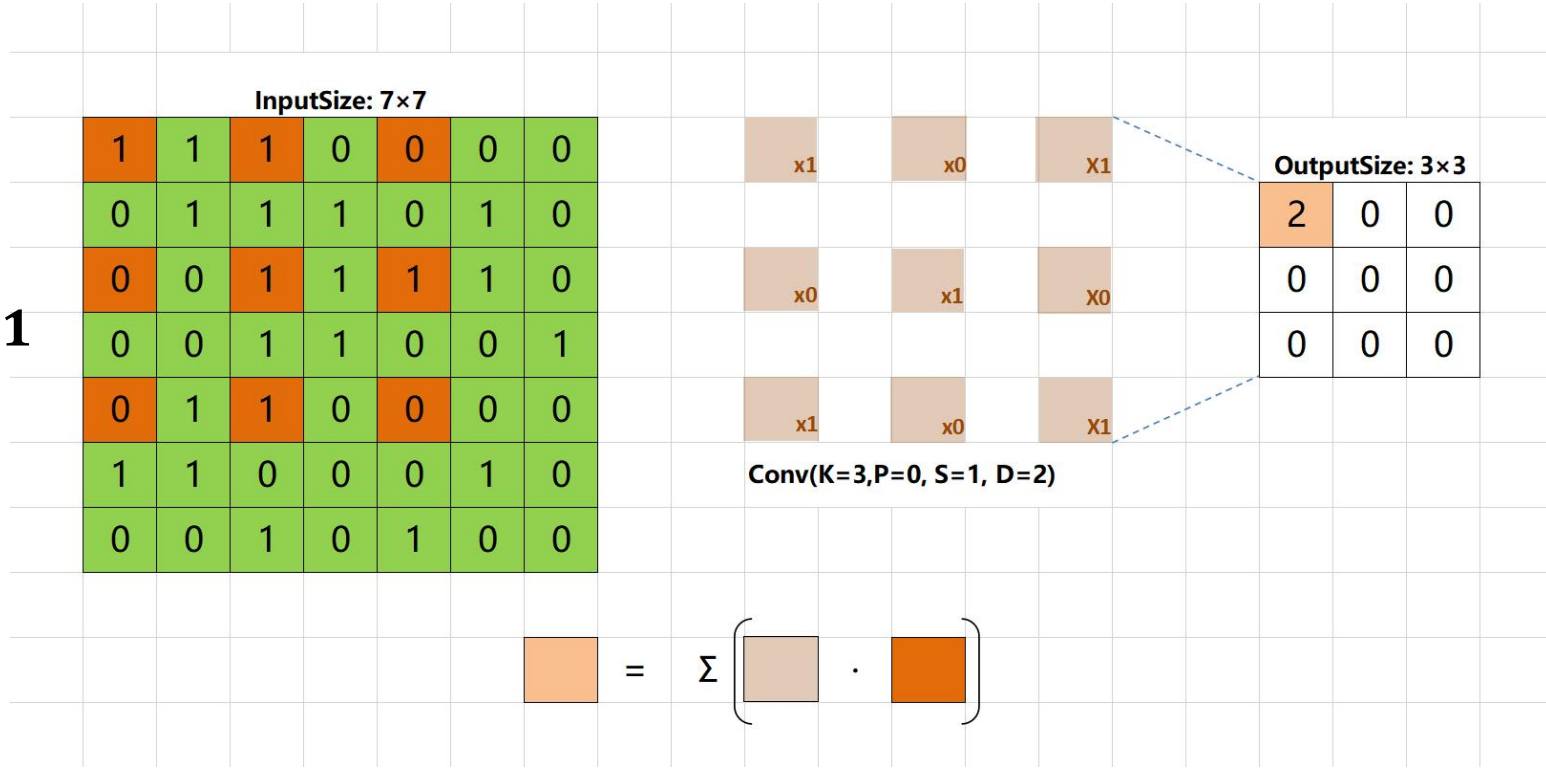
$$H_{out} = \frac{H_{in} + 2P - (D * (K-1) + 1)}{S} + 1$$

Dilation

Padding

KernelSize

stride




- *空洞卷积核的尺寸*

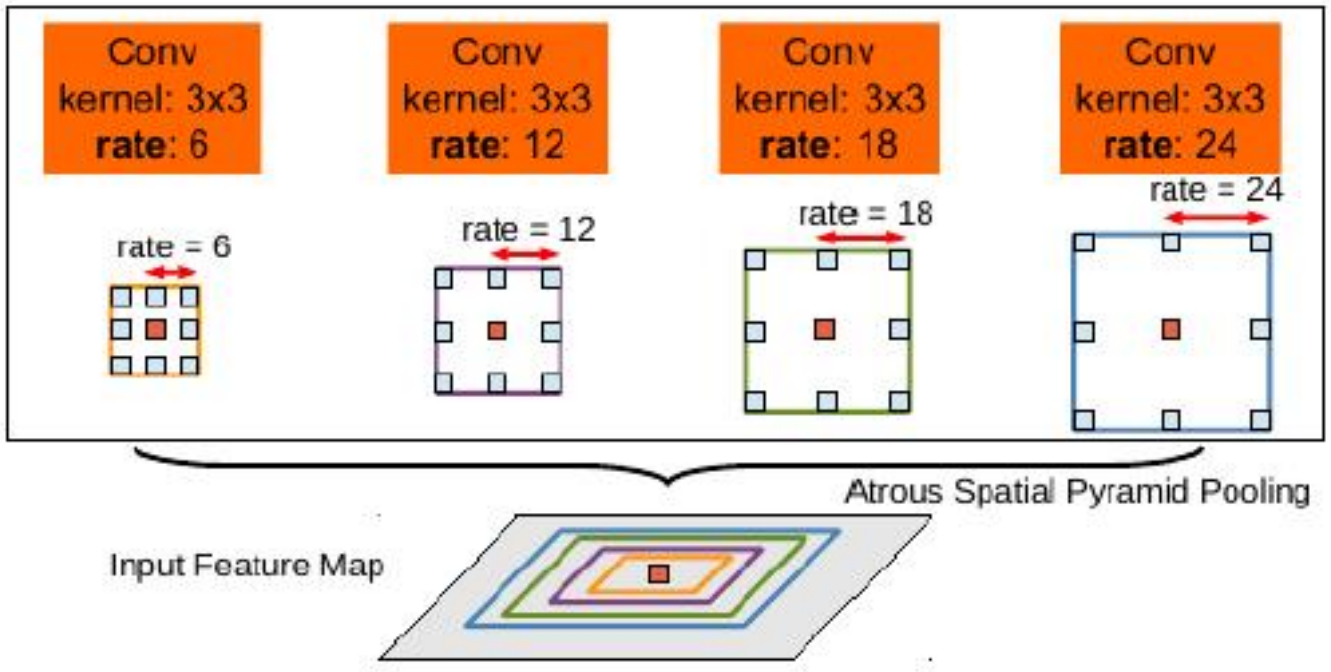
$$K_D = K + (K-1)(D-1)$$

空洞卷积的KernelSize

# 3.5.1 DeepLab的核心技术

## 空洞空间金字塔池化

空洞空间金字塔池化(**Atrous Spatial Pyramid Pooling, ASPP)，是DeepLab的核心结构，用于**从多个不同视野中进行特征提取，并最后进行融合以达到上下文信息融合的作用。

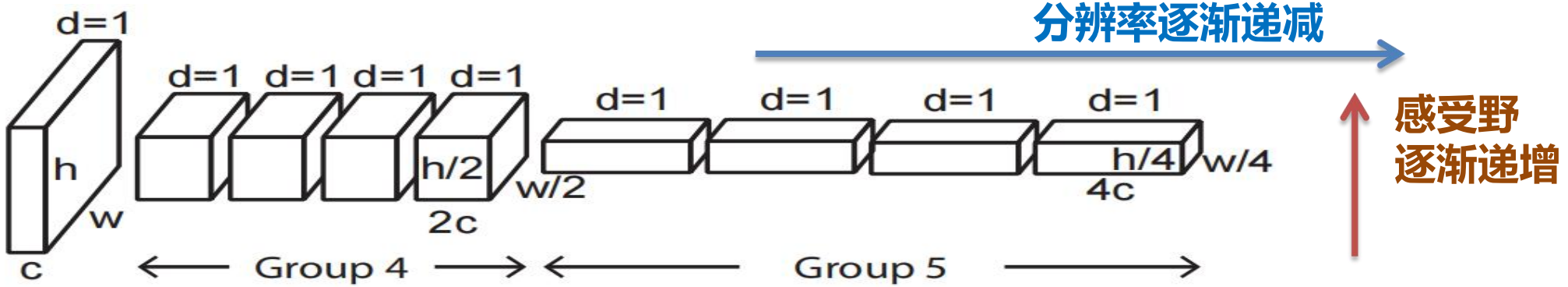



rate可以理解为空洞的尺度，空洞卷积尺度 = K+(K-1)*(rate-1)

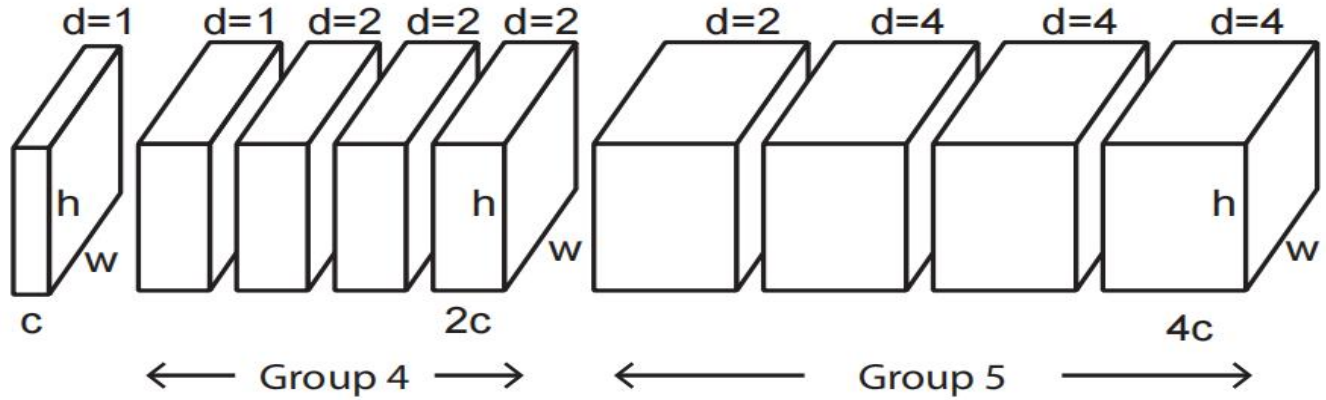
Liang-Chieh Chen, Yunkun Zhu, George Papandreou, Fioran Schroff, Hartwig Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. ECCV 2018

# 3.5.1 DeepLab的核心技术

## Dilated ResNet




标准ResNet




PSP网络的Backbone

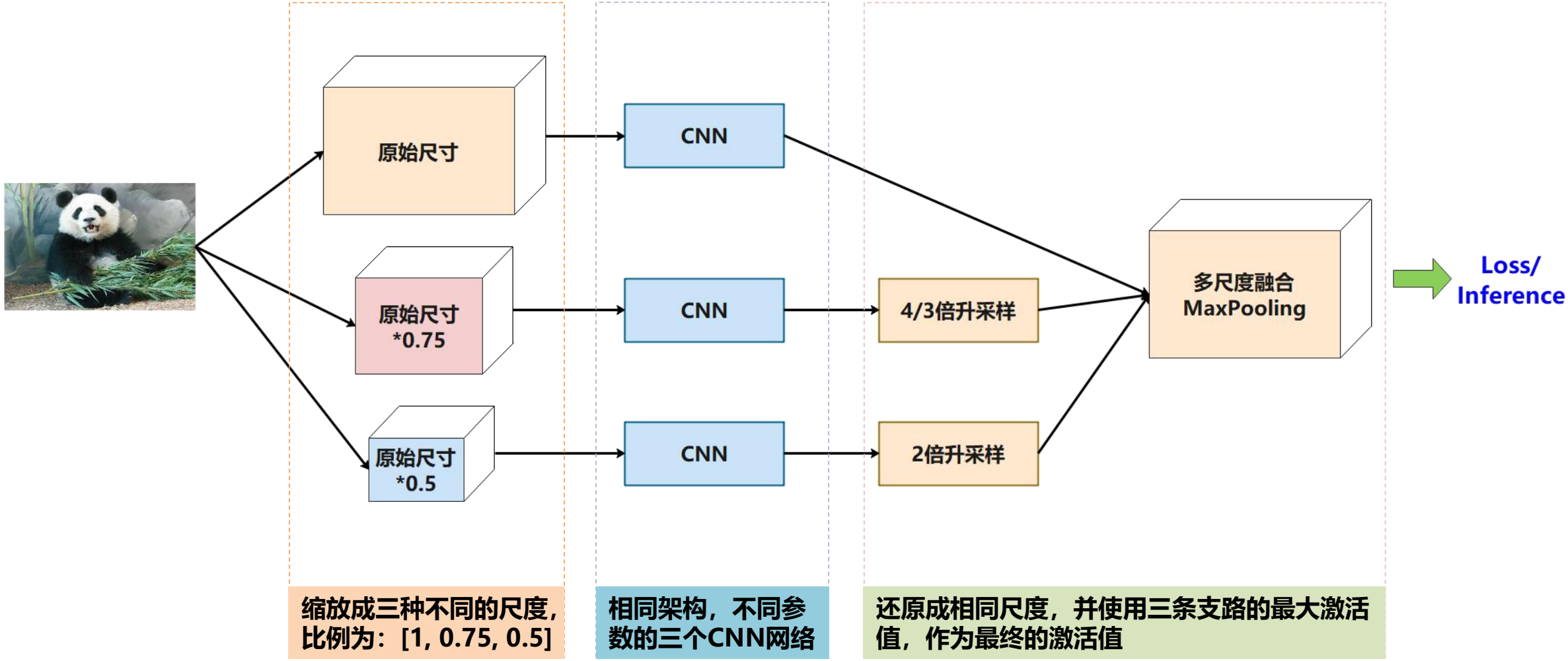
空洞卷积ResNet （Dilated ResNet）

特征图的分辨率和感受野的尺度由于使用了空洞卷积（配合padding），而保持不变。

Thomas Funkhouser. Dilated Residual Networks. CVPR2017

# 3.5.1 DeepLab的核心技术

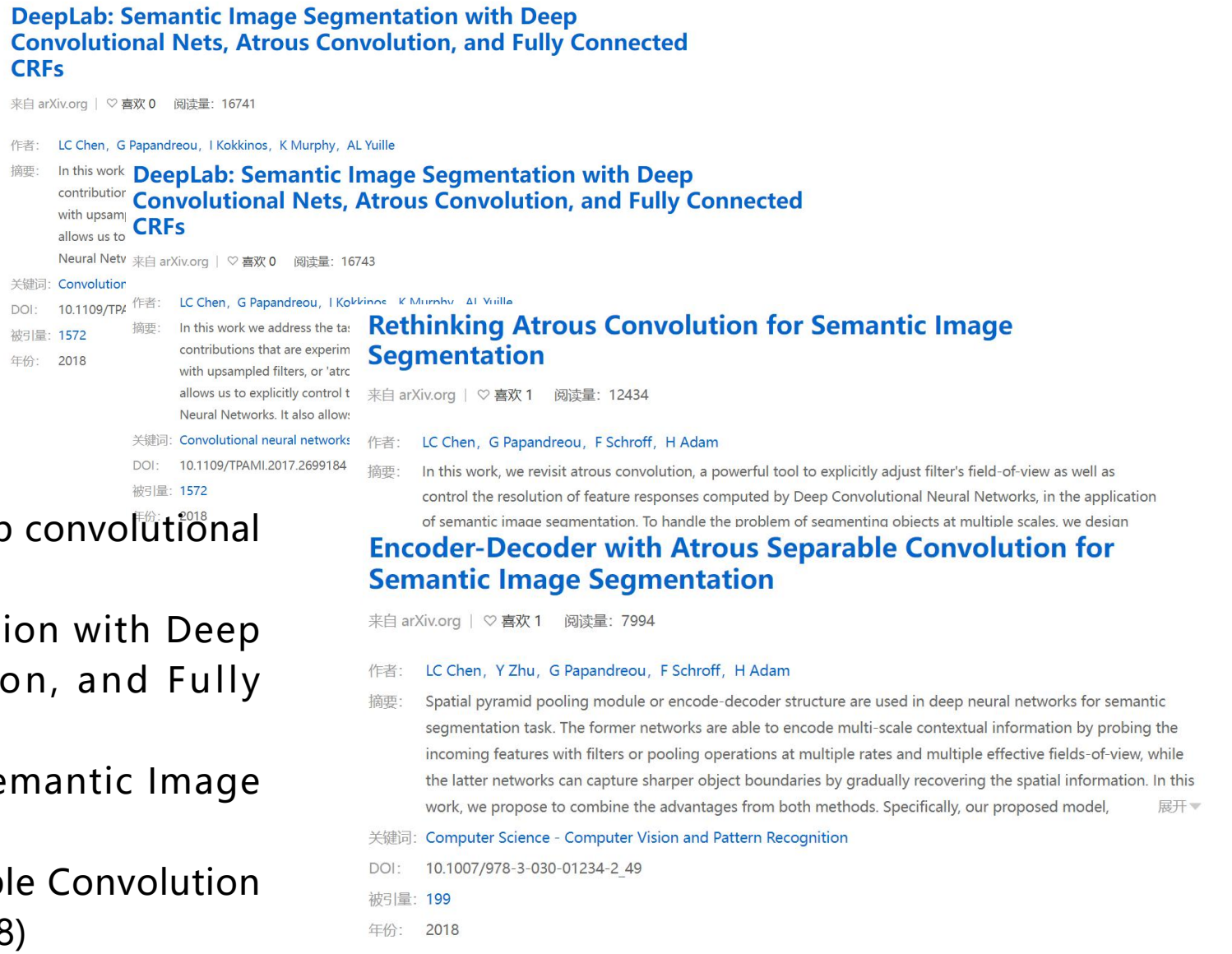
## MultiScale-Train



# 3.5.2 DeepLab系列网络

## DeepLab系列网络

- **V1:** Semantic image segmentation with deep convolutional nets and fully connected CRFs (ICLR 2015)
- **V2:** DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs (TPAMI 2018)
- **V3:** Rethinking Atrous Convolution for Semantic Image Segmentation (arXiv)
- **V3+:** Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation (ECCV 2018)



# 3.5.2 DeepLab系列网络

## DeepLab系列网络的体系结构

- ***DeepLab系列网络关键结构对比***

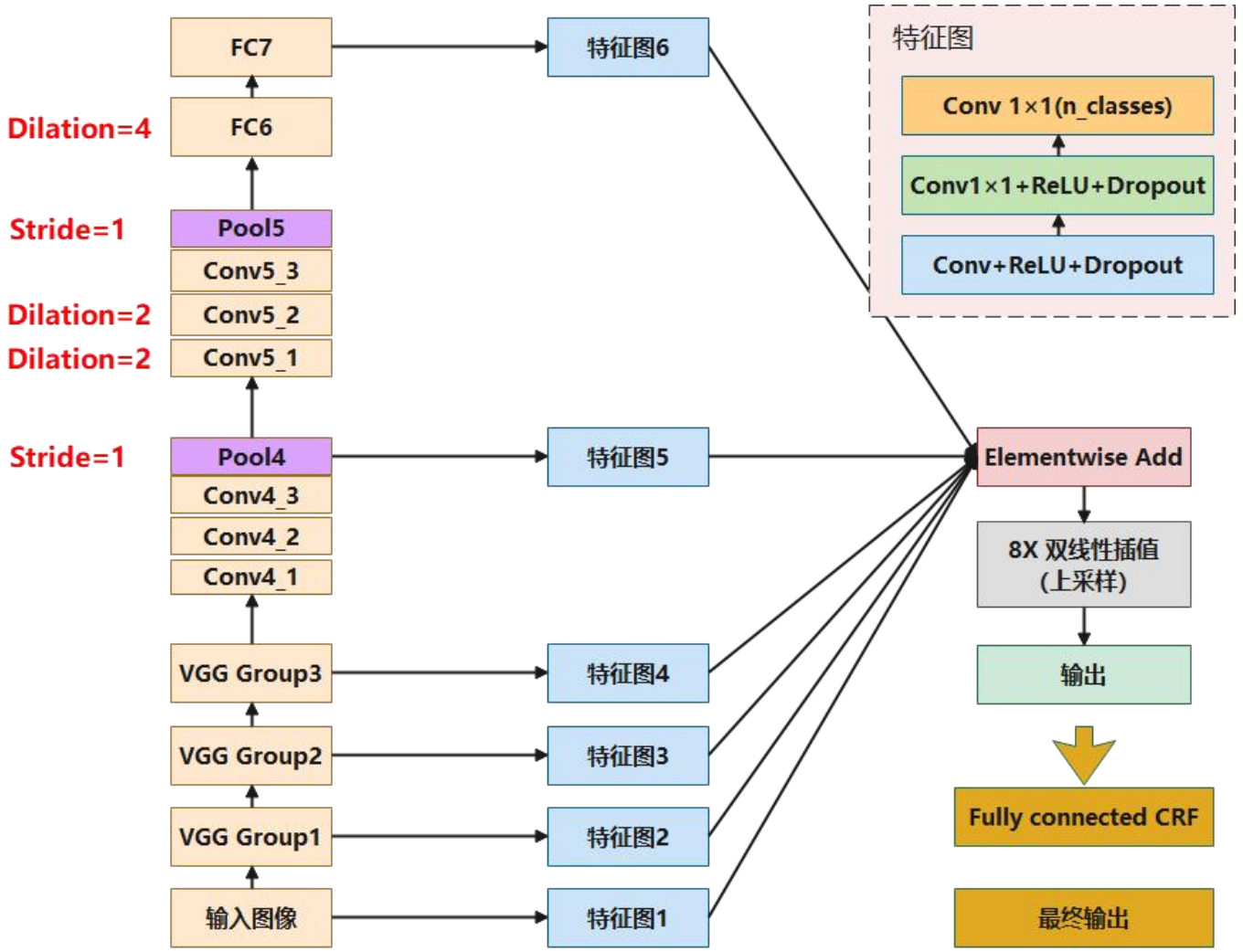
| Architecture | Backbone | Atrous Conv | MultiScale | Full-connect CRF |
|---|---|---|---|---|
| DeepLab v1 | VGG-16 | Atrous Block | Traning | Yes |
| DeepLab v2 | ResNet | ASPP | Traning | Yes |
| DeepLab v3 | MG ResNet | 升级版ASPP | Inference | No |
| DeepLab v3+ | Xception | ASPP+decoder | Inference | No |

# 3.5.2 DeepLab系列网络

## DeepLab v1

*What is DeelLab v1 ?*

- **Pool4, Pool5**：Stride = 1 (原始VGG=2) **扩大视野**
- **Conv5_1, Conv5_2:** Dilated Conv (Dilation=2)
- **FC6**：Dilated Conv (Dilation=4) **保持视野不变**
- **FC7**：Conv1×1
- **特征图**：
  - ✓ Conv(Stride)+ReLU+Dropout **尺寸统一**
  - ✓ Conv1×1+ReLU+Dropout
  - ✓ Conv1×1 (n_classes)
- **Elementwise Add**：按像素加 **多层特征融合**
- **8X双线性插值**：原来VGG包含5个pool(Stride=2)，池化后的原始图缩小32倍；DeepLab的Pool4, 5缩小Stride后，池化后的原始图缩小8倍，因此在输入分割器前需要进行8倍上采样
- **Fully-connnected CRF**：后处理，不参与训练。**优化输出**




Architecture of DeepLab v1

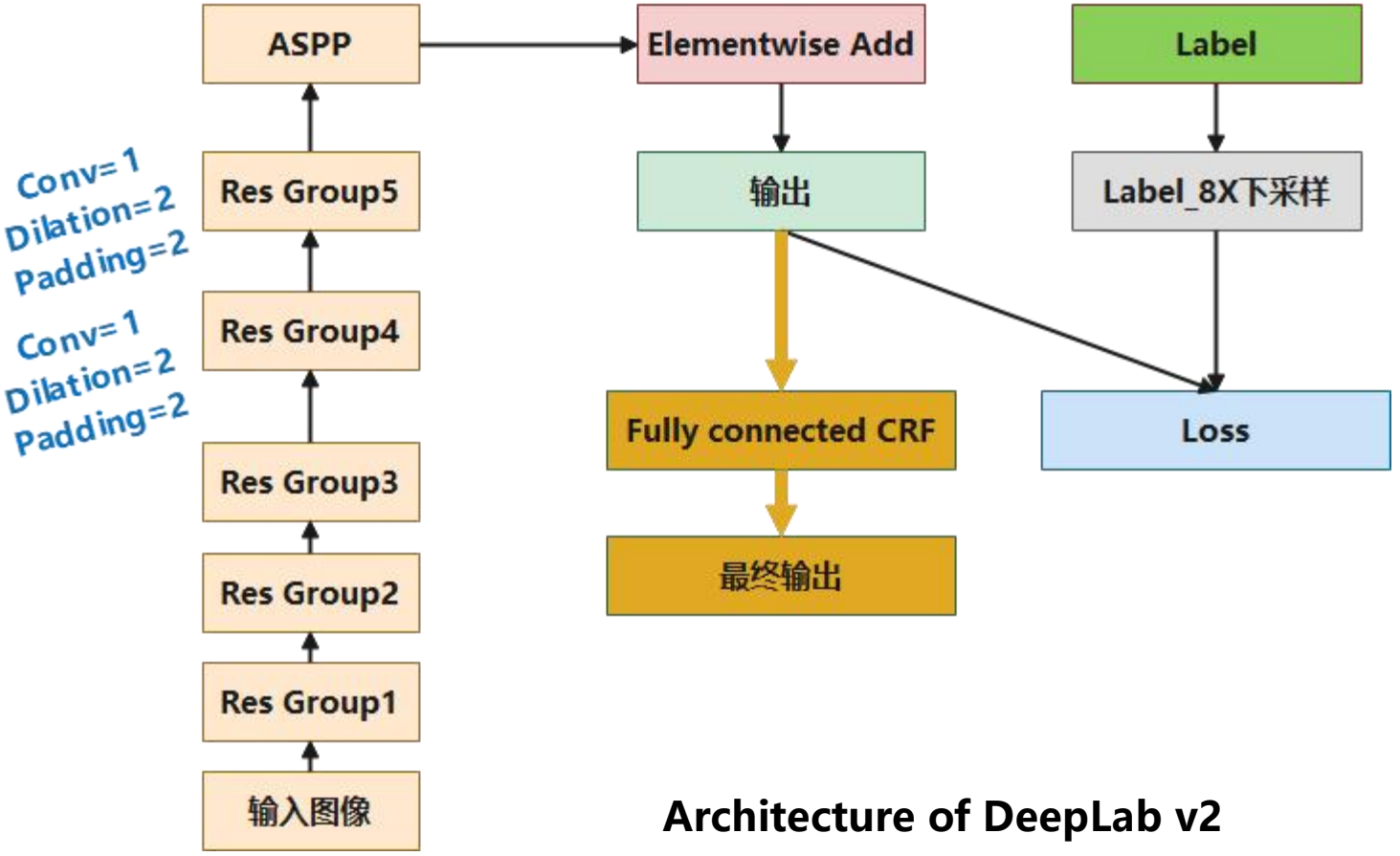
Liang-Chieh Chen, George Papandreou, et.al. Semantic image segmentation with deep convolutional nets and fully connected CRFs. ICLR2015

# 3.5.2 DeepLab系列网络

## DeepLab v2

*What is DeepLab v2 ?*

- **Backbone**: Dailated ResNet
- **Header**：ASPP Module
  - ✓ **A**trous **S**patial **P**yramid **P**ooling
- **Label**：8X下采样
- **预测**：**CNN**+Fully-Connected CRF



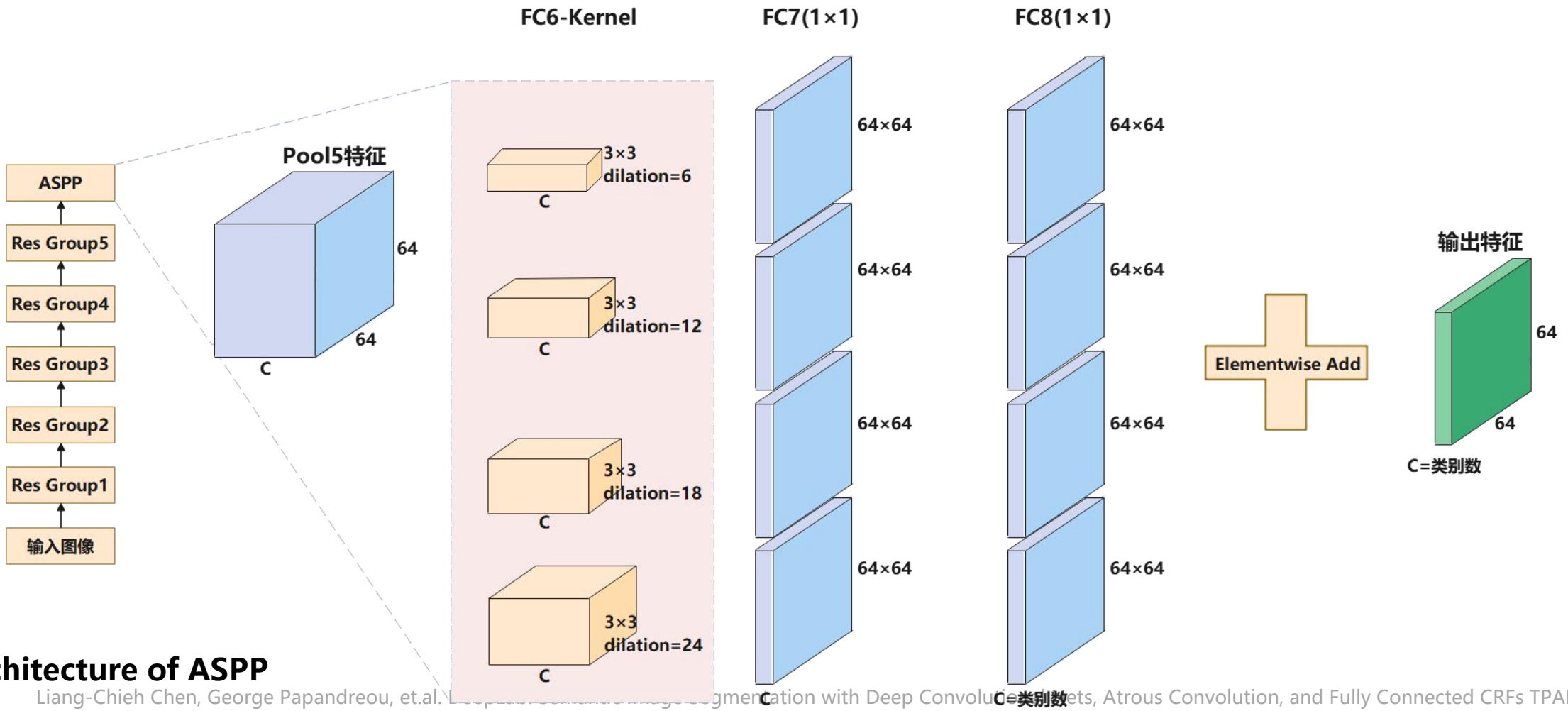

**Architecture of DeepLab v2**

Liang-Chieh Chen, George Papandreou, et.al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs TPAMI2018

# 3.5.2 DeepLab系列网络

## DeepLab v2

*What is ASPP modual ?*




**Architecture of ASPP**

Liang-Chieh Chen, George Papandreou, et.al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs TPAMI2018

# 3.5.2 DeepLab系列网络

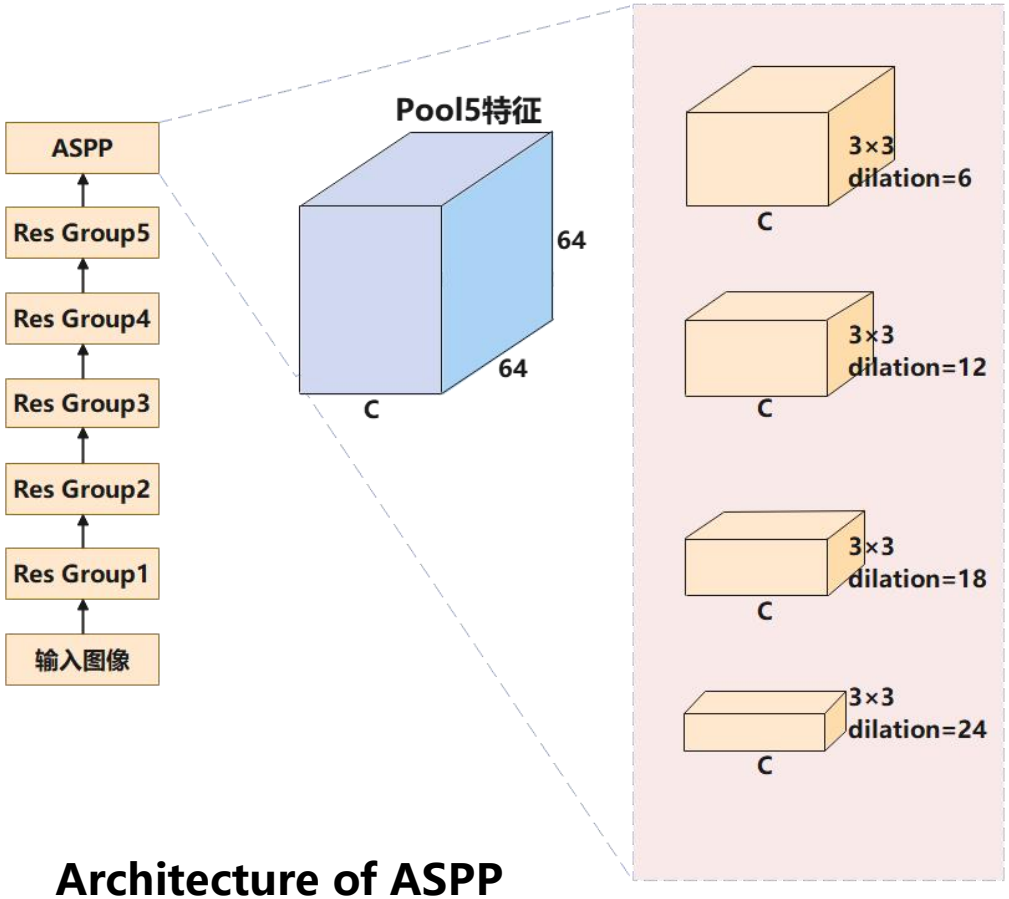
$$H_{out} = \frac{H_{in} + 2P - (D * (K-1) + 1)}{S} + 1$$

= (10+2*3-(3*2+1))/1 + 1

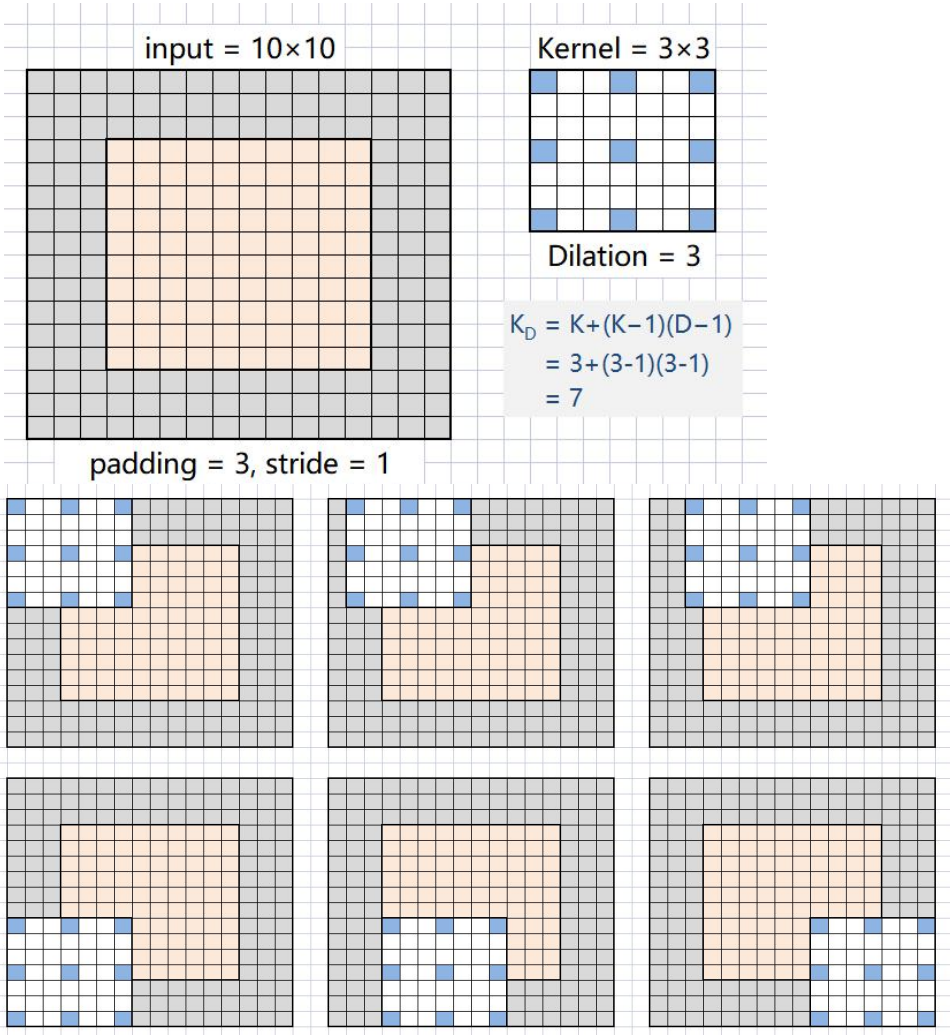
= 10

## DeepLab v2

*What is ASPP modual ?*




**Architecture of ASPP**



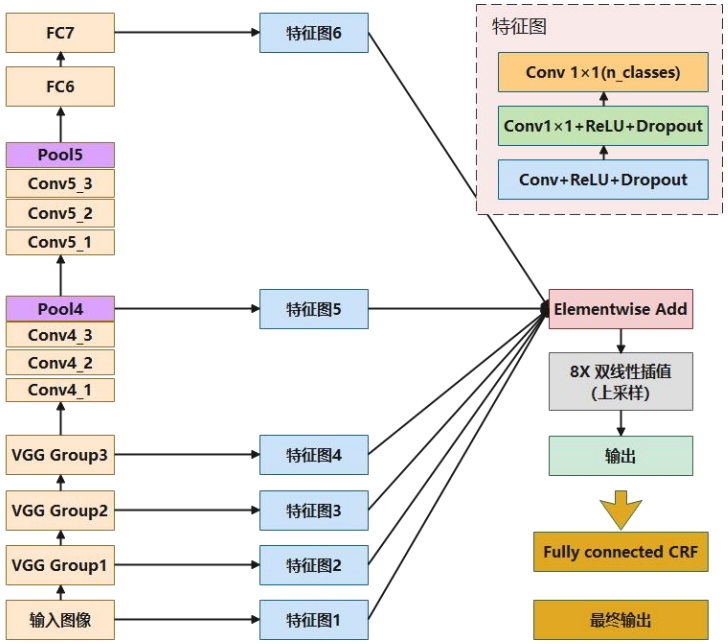

Liang-Chieh Chen, George Papandreou, et.al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs TPAMI2018

# 3.5.2 DeepLab系列网络

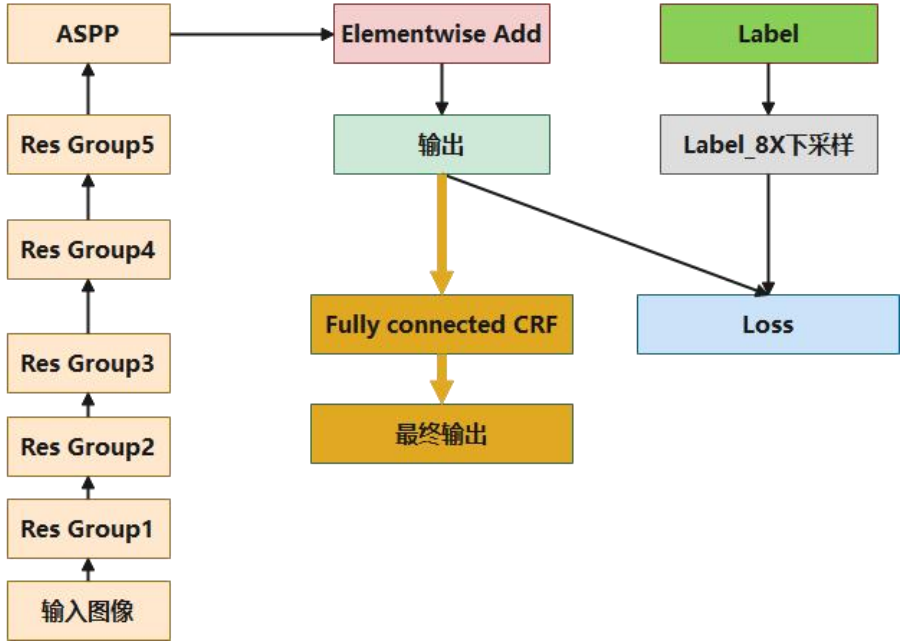
## DeepLab v3

*What is DeepLab v3 ?*

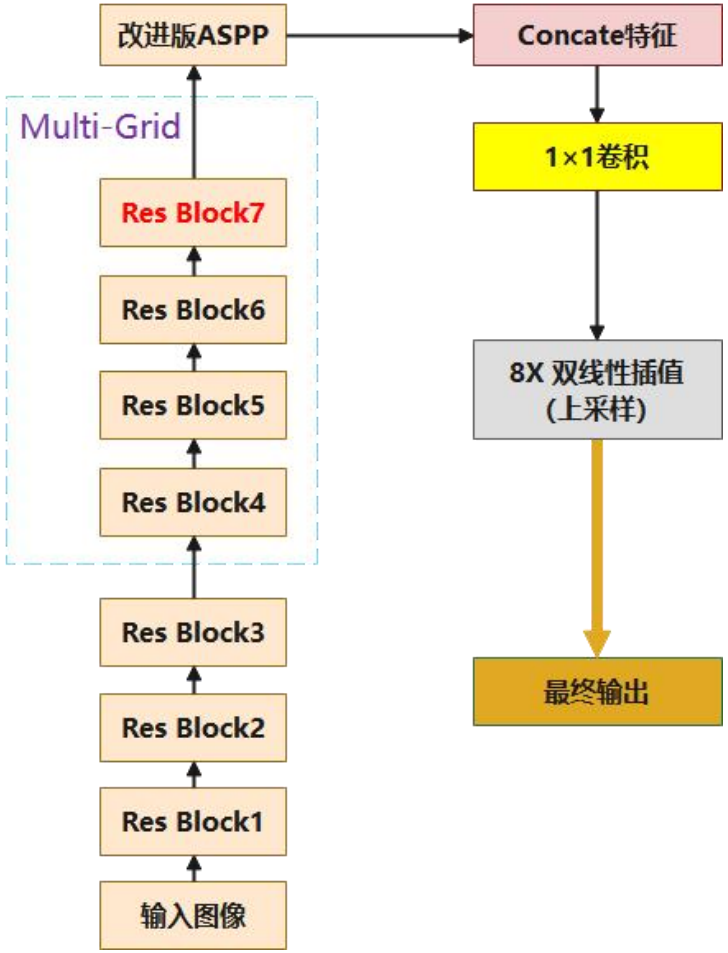
*From **DeepLab v1** to **DeepLab v3***




DeepLab v1




DeepLab v2

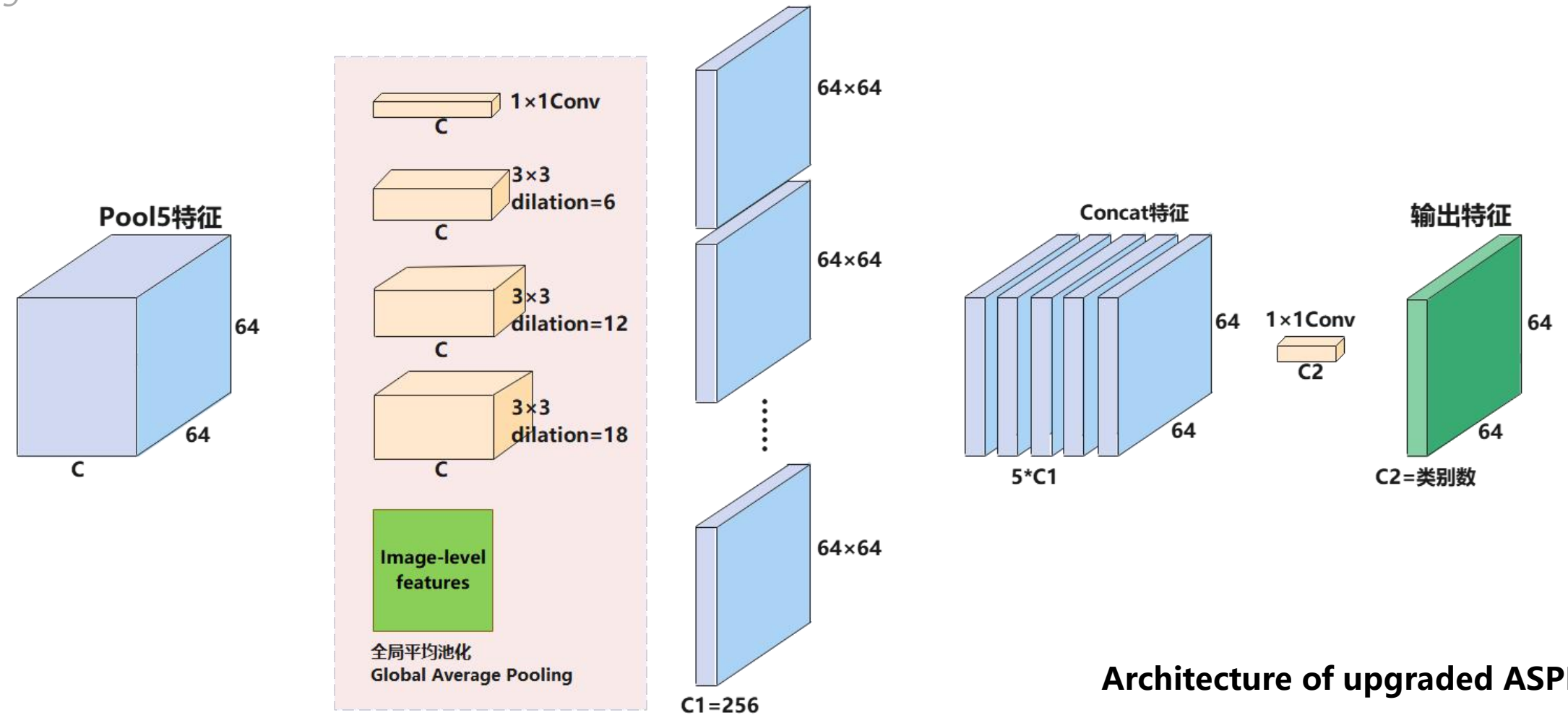



DeepLab v3

# 3.5.2 DeepLab系列网络

## DeepLab v3 - 升级版ASPP模块

*What is upgraded ASPP ?*




**Architecture of upgraded ASPP**

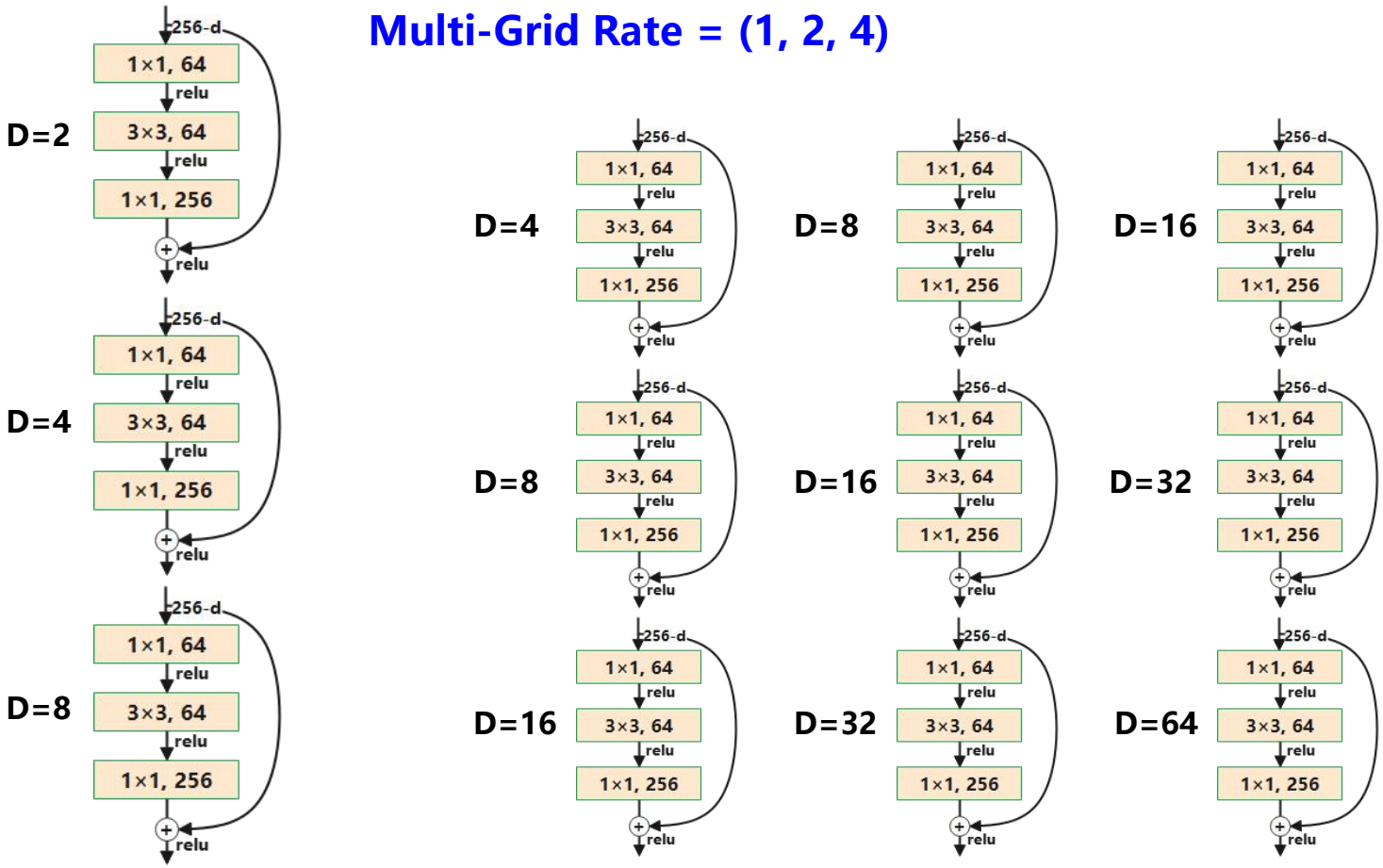
Liang-Chieh Chen, George Papandreou, Florian Schroff, Hartwig Adam. Rethinking Atrous Convolution for Semantic Image. arXiv1706

# 3.5.2 DeepLab系列网络

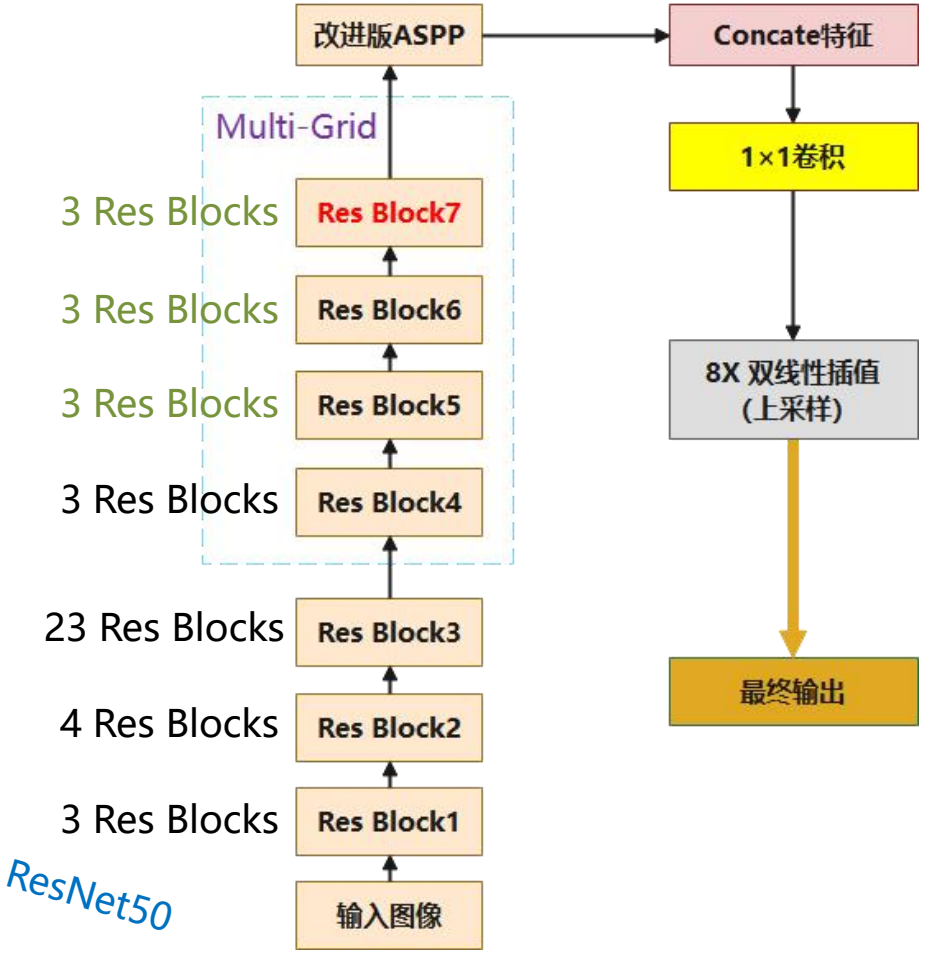
## DeepLab v3 - MultiGrid

*What is MultiGrid?*

**Multi-Grid Rate = (1, 2, 4)**




Block4, D=2　Block5, D=4　Block6, D=8　Block7, D=16



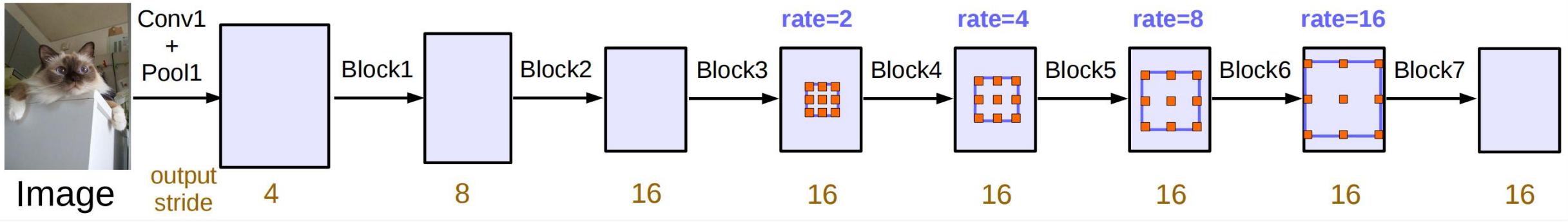

DeepLab v3

Liang-Chieh Chen, George Papandreou, Florian Schroff, Hartwig Adam. Rethinking Atrous Convolution for Semantic Image. arXiv1706

# 3.5.2 DeepLab系列网络

## DeepLab v3 - Go deeper & ASPP with ImagePooling

*What is DeepLab v3 shown in the vanilla paper?*




Going deeper with atrous convolution.

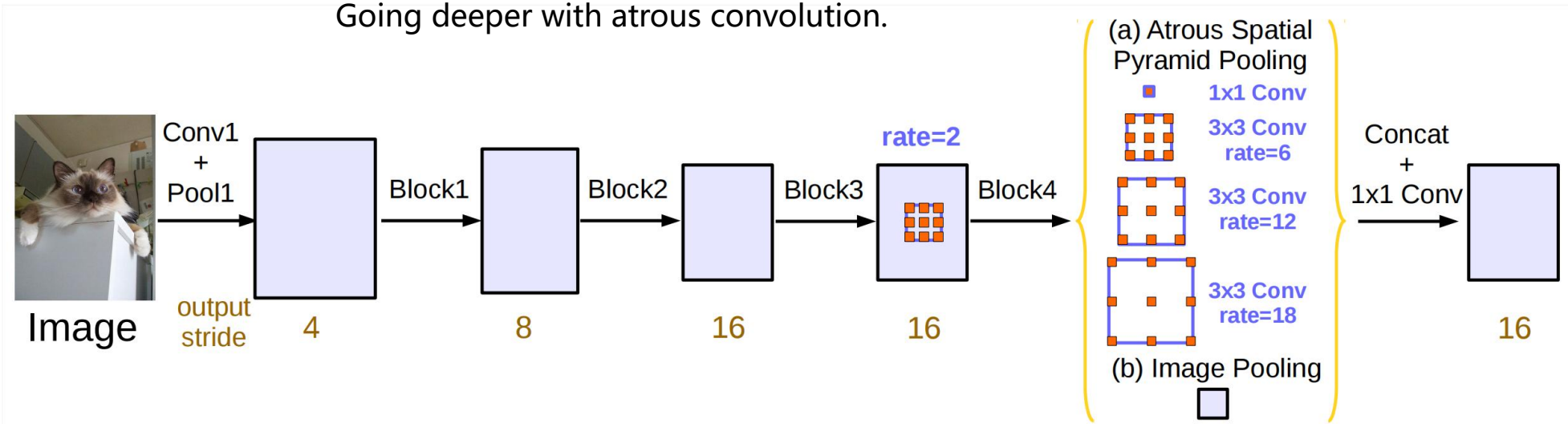


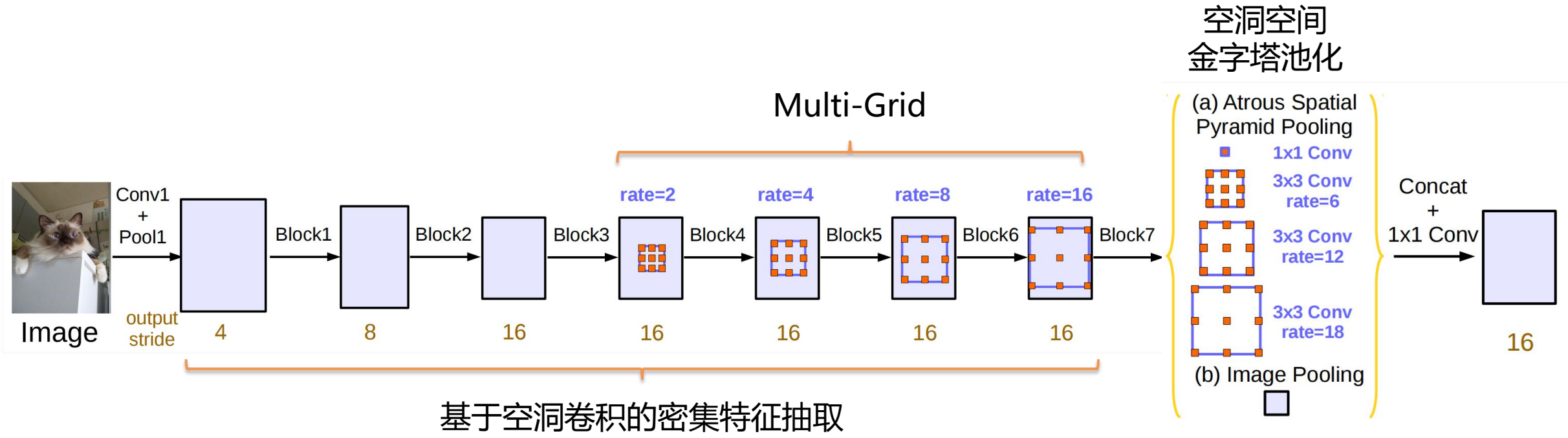

Figure 5. Parallel modules with atrous convolution (ASPP), augmented with image-level features.

mage. arXiv1706

# 3.5.2 DeepLab系列网络

## DeepLab v3 - 完整结构

*What is DeepLab v3 shown in the vanilla paper?*




Liang-Chieh Chen, George Papandreou, Florian Schroff, Hartwig Adam. Rethinking Atrous Convolution for Semantic Image. arXiv1706

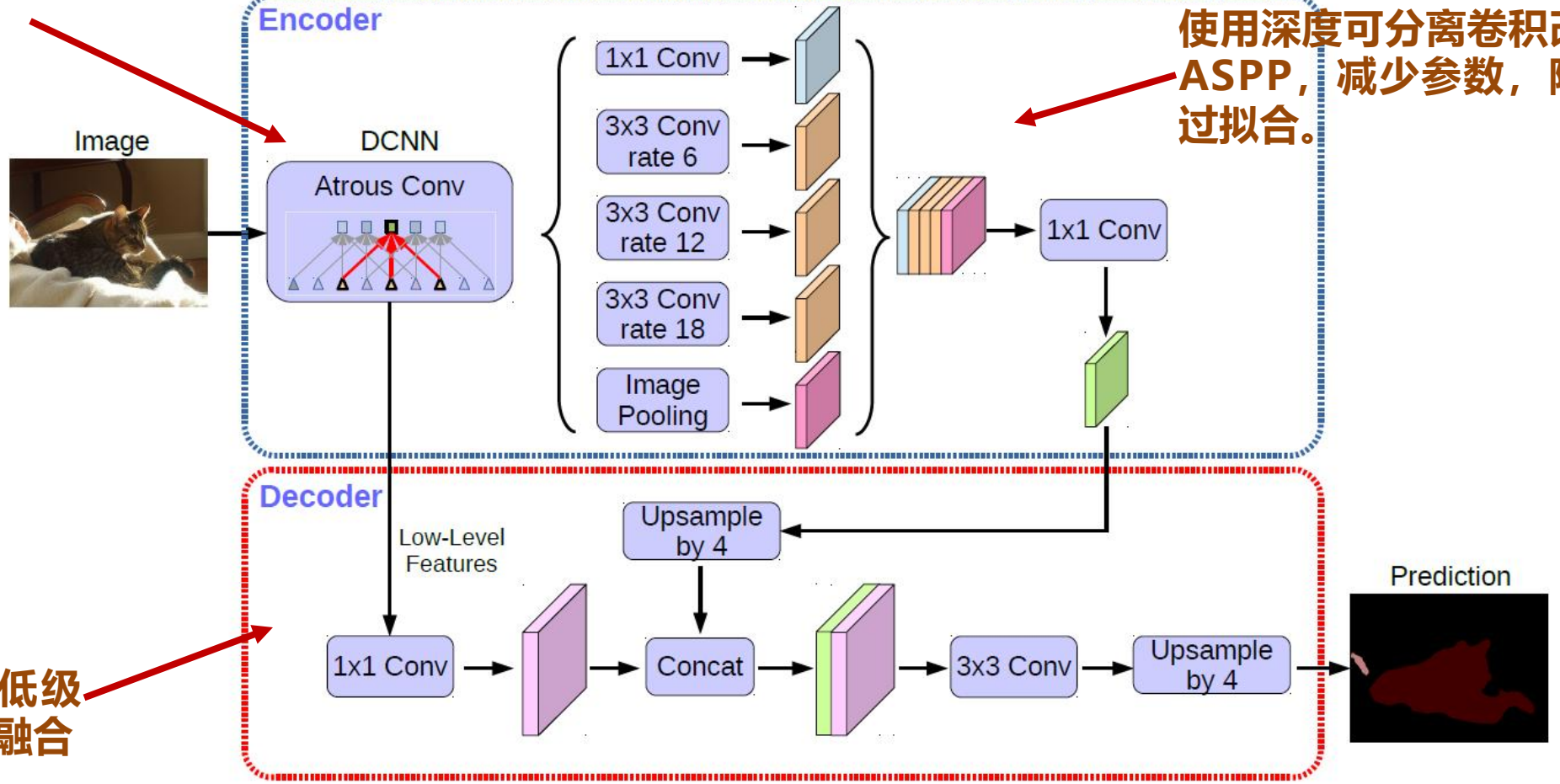
# 3.5.2 DeepLab系列网络

使用更优的Backbone——微软Xception，并使用空洞卷积对该Backbone进行改造，在深度可分离卷积后增加BN和ReLU。

## DeepLab v3+ 体系结构

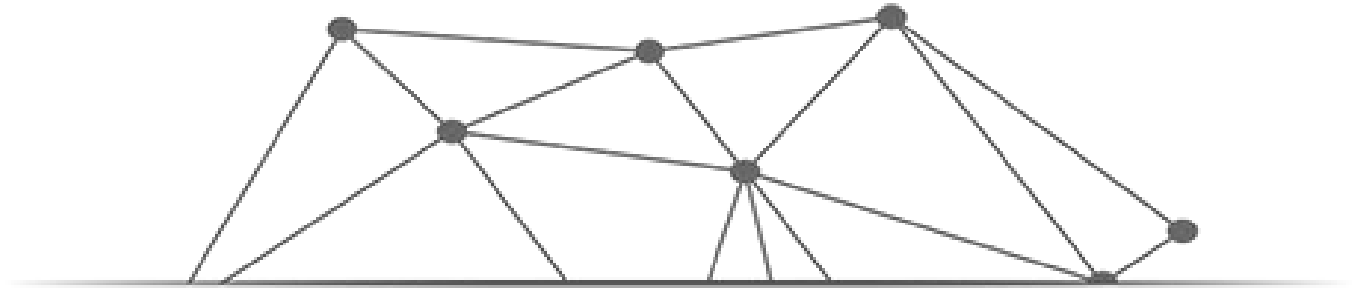
*目前最优DeepLab版本*

使用深度可分离卷积改造ASPP，减少参数，降低过拟合。

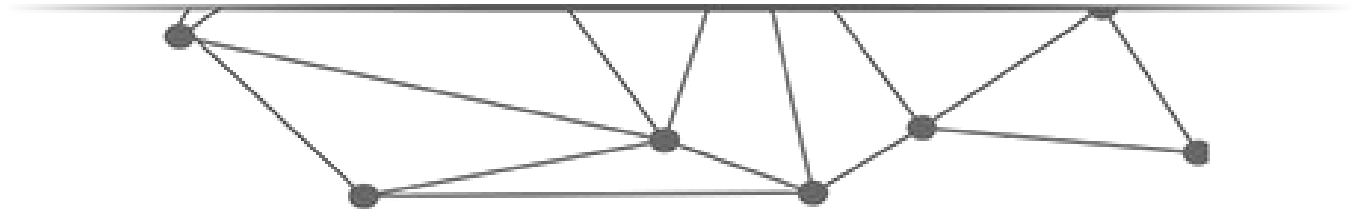



增加Decoder，对低级特征和高级特征进行融合

Liang-Chieh Chen, Yunkun Zhu, George Papandreou, Fioran Schroff, Hartwig Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. ECCV 2018



# 课堂互动 13.3.5



Part
04

# 实例分割和全景分割

# 4. 实例分割和全景分割

## 本节内容

01 **What is Instense Segmentation and what is Panoptic Segmentation?**
实例分割的基本概念；全景分割的原理

02 **经典实例分割模型**
Proposal-based **Mask RCNN** [ICCV2017]；
Proposal-free **SOLO**, **SOLOv2** [ECCV2020, NIPS2020]

03 **经典全景分割模型**
Proposal-based **UPSNet** [CVPR2019]；
Proposal-free **Panoptic-DeepLab** [CVPR2020]

# 4.1 基本概念

## 语义分割和全景分割

- ***语义分割（Semantice Segmentation）***

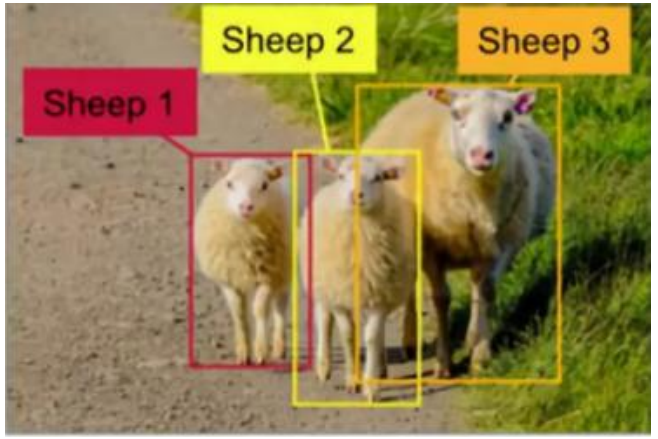
对每个像素（包括背景）进行分类，并将相同像素使用相同的索引（颜色）进行编号，**不区分个体。**

- ***实例分割（Instance Segmentation）***

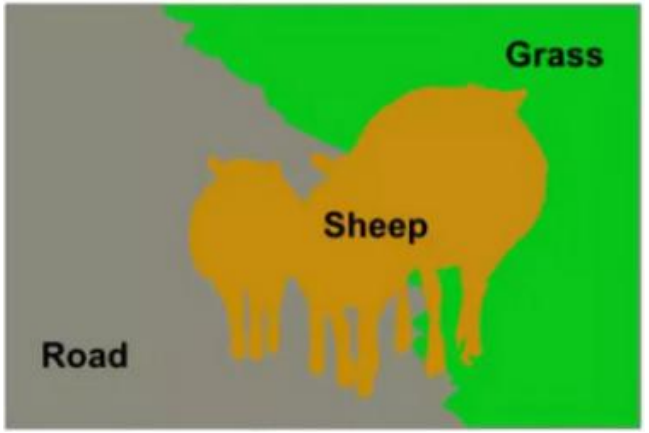
对每个像素（不包括背景）进行分类，同时**区分不同的个体**，并将同一类别且属于同一个体的区域，使用相同的索引（颜色）进行编号。

- ***全景分割（Panoptic Segmentation）***

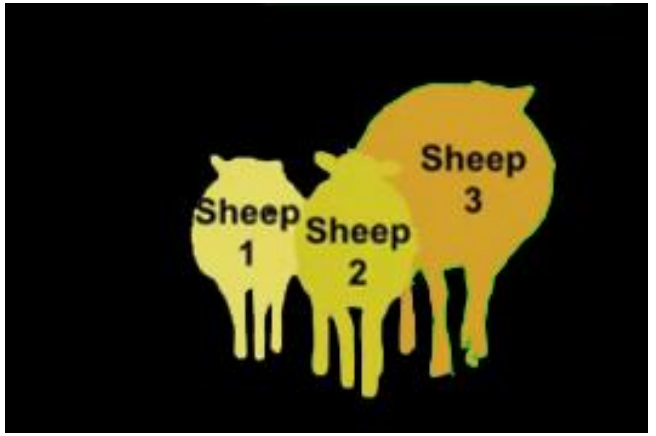
对每个像素（包括背景）都进行分类，同时**区分不同的个体**，并将同一类别且属于同一个体的区域，使用相同的索引（颜色）进行编号。



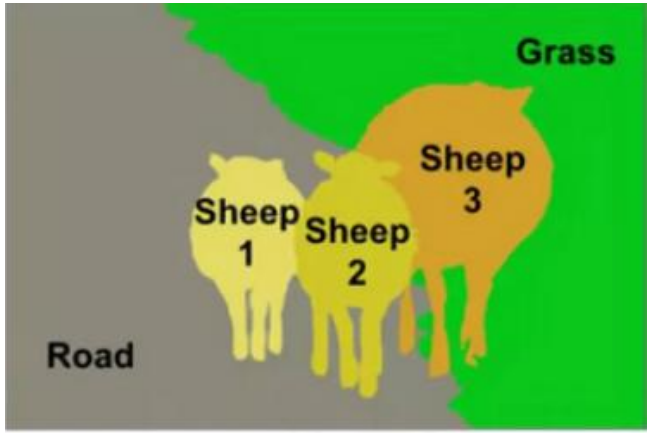

原图像




语义分割

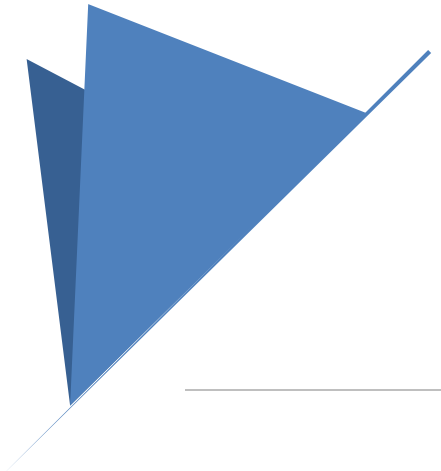



实例分割




全景分割

Alexander, et.al. Panoptic Segmentation. *CVPR 2019.*

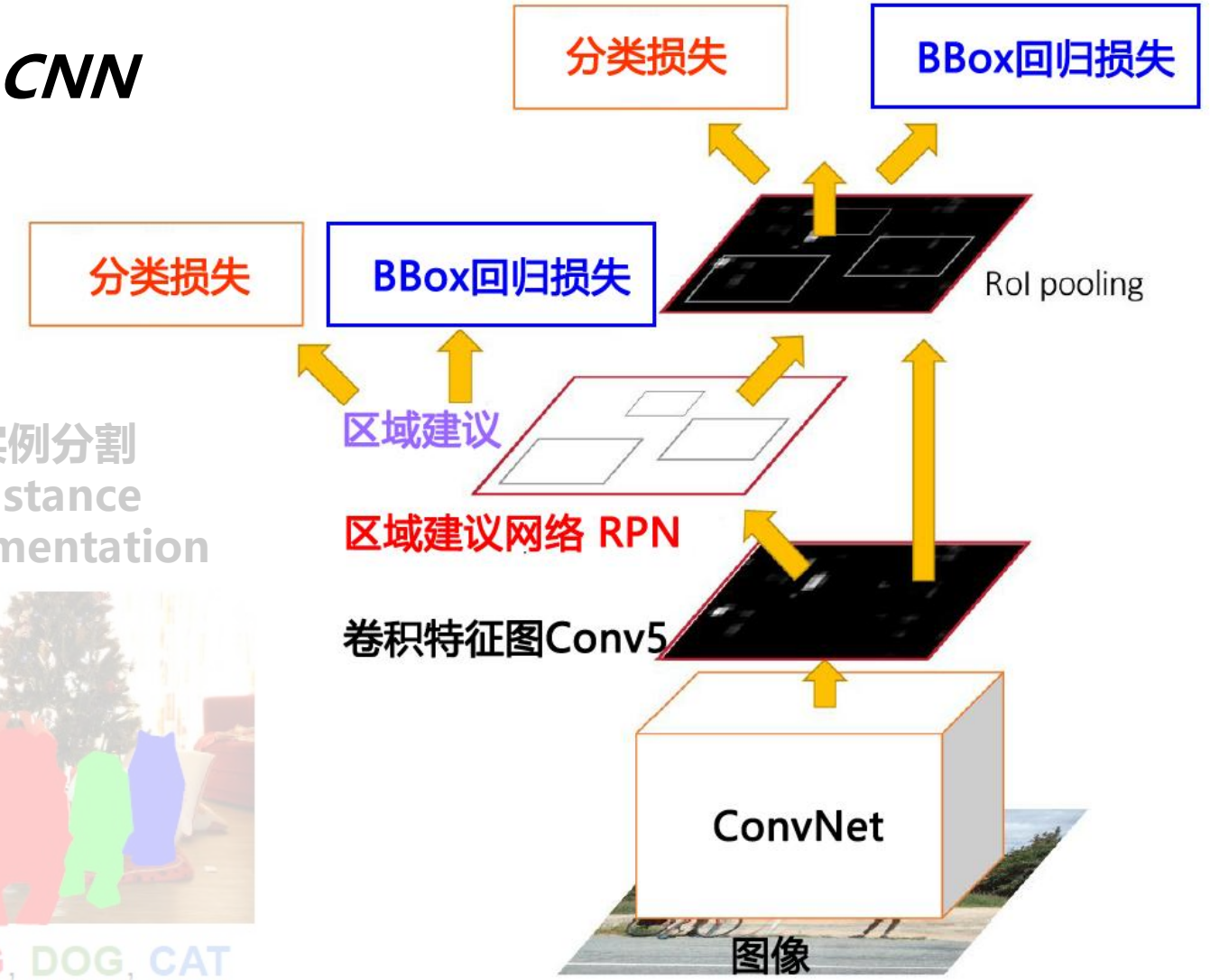


# Mask RCNN

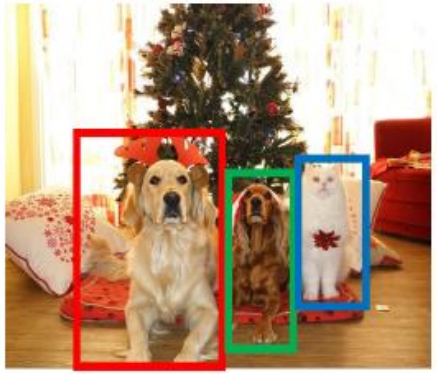
# 4.2 Mask RCNN

## 基于检测的实例分割Mask R-CNN

***目标检测：Faster R-CNN***




**目标检测**
**Object Detection**



DOG, DOG, CAT

实例分割
Instance
Segmentation



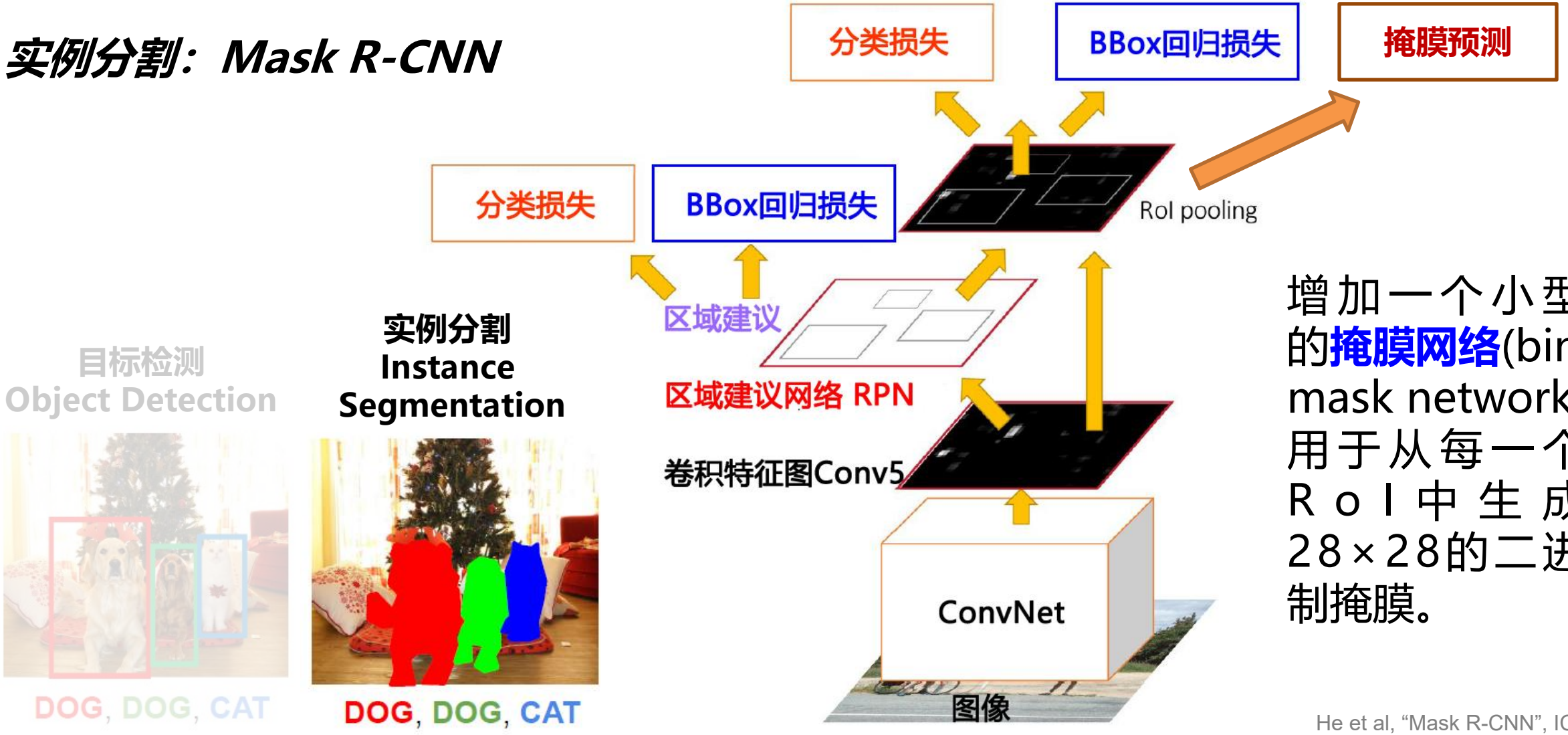
DOG, DOG, CAT

He et al, “Mask R-CNN”, ICCV 2017

# 4.2 Mask RCNN

## 基于检测的实例分割Mask R-CNN

***实例分割：Mask R-CNN***




增加一个小型的**掩膜网络**(bin mask network)用于从每一个RoI中生成28×28的二进制掩膜。

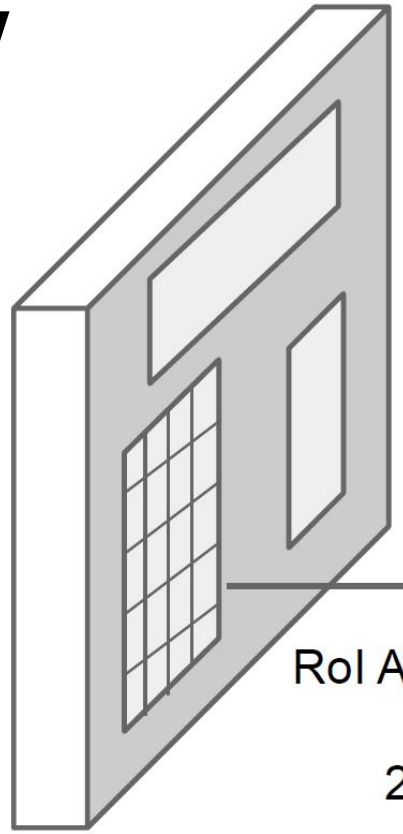
He et al, “Mask R-CNN”, ICCV 2017

# 4.2 Mask RCNN

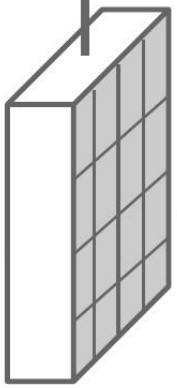
## 基于检测的实例分割Mask R-CNN

***实例分割：Mask R-CNN***

CNN
+RPN

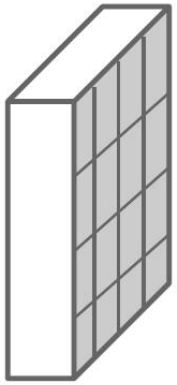


RoI Align



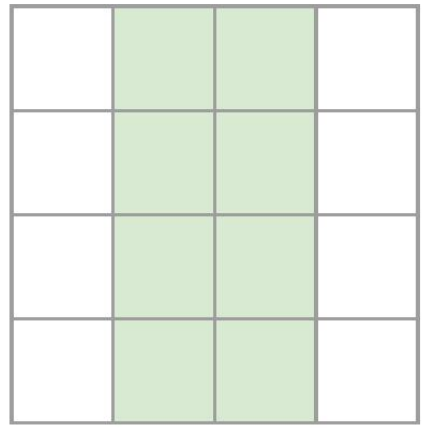
256 x 14 x 14

Conv



256 x 14 x 14

Conv



分类分数：C个

边界框坐标(每类)：4×C个

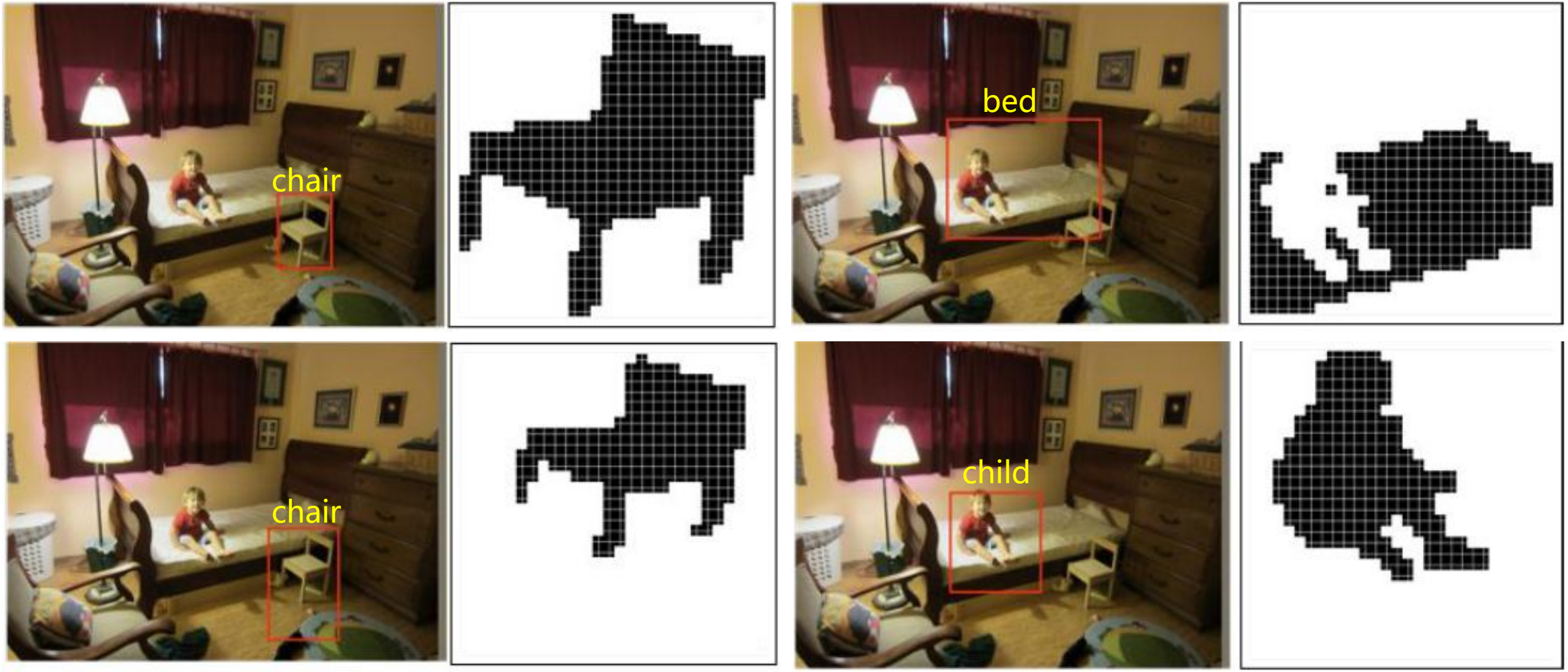
对每个类都预测
一个掩膜（C类）

C x 28 x 28

He et al, “Mask R-CNN”, ICCV 2017

# 4.2 Mask RCNN

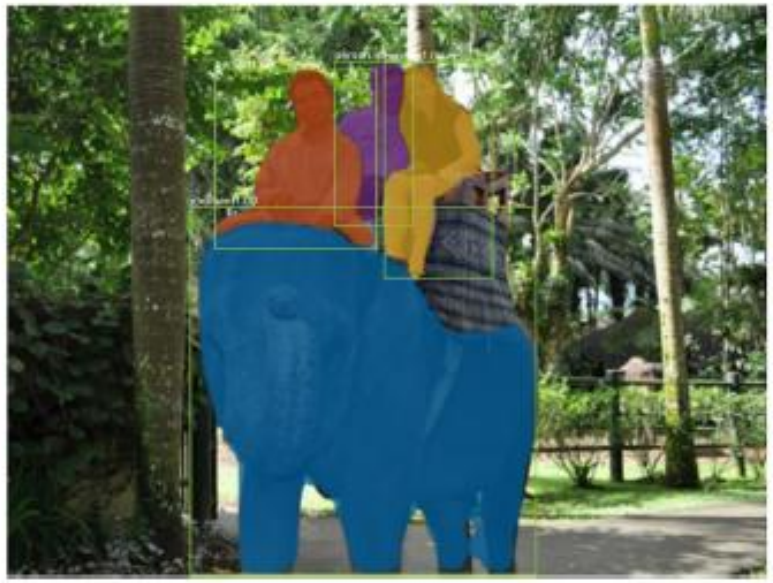
## 基于检测的实例分割Mask R-CNN

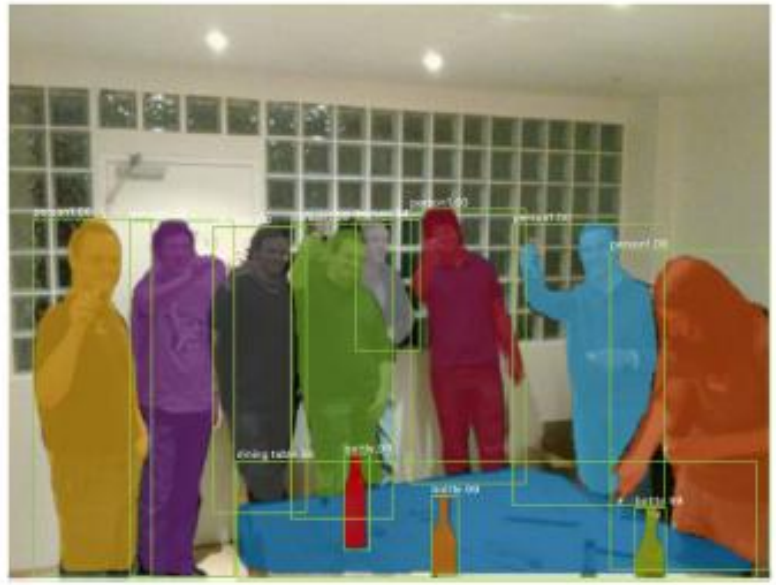


# 4.2 Mask RCNN

## 基于检测的实例分割Mask R-CNN

***Very Good Results!***





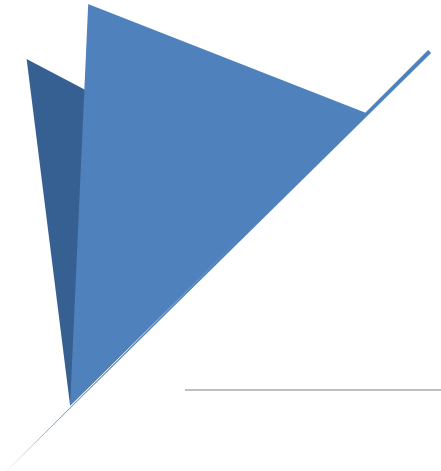
He et al, “Mask R-CNN”, ICCV 2017

# 4.2 Mask RCNN

## 基于检测的实例分割Mask R-CNN

***Mask R-CNN 也可以做姿态识别 (Pose)***


He et al, “Mask R-CNN”, ICCV 2017



# Solo

## Segmenting Object by Location

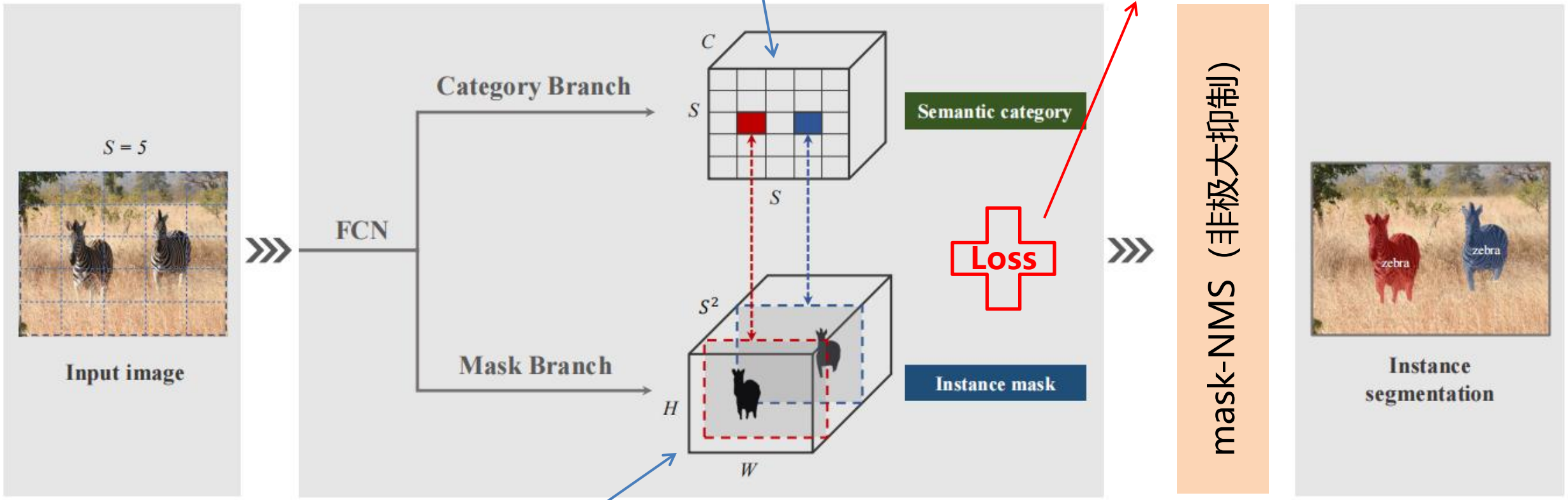
# 4.3 Solo: Segmenting Object by Location

特征图被划分为$S \times S$的网格，每个网格都进行语义类别识别（基于概率），即每个网格只能属于一个类别（类比Yolo）。S的典型设置为[12, 24, 36, Pyramid]

**Focal Loss**
缓解难易样本和正负样本不平衡

**Dice Loss**
比BCE优5-7%

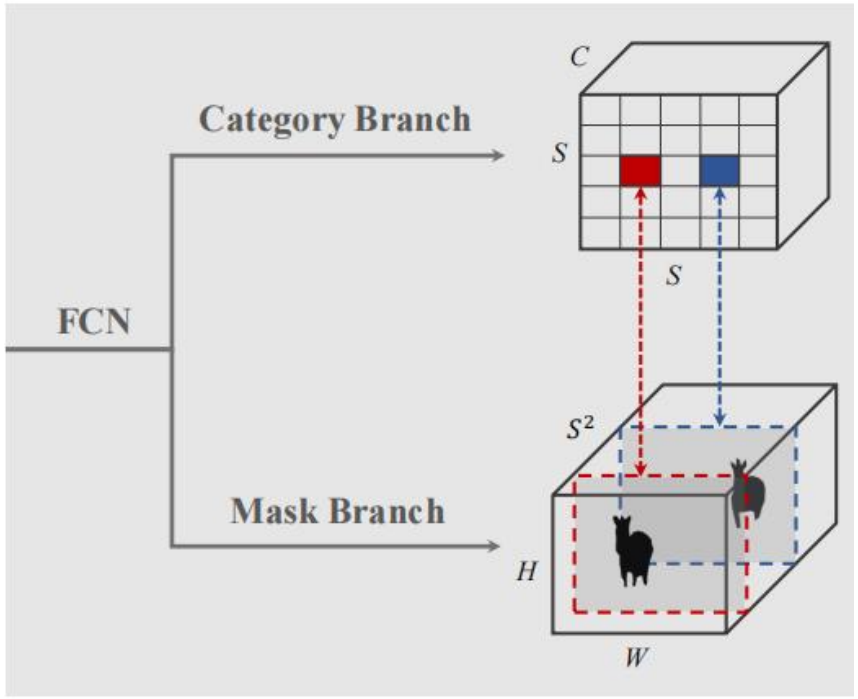
$$L = L_{category} + \lambda L_{mask}$$




对于每个正样本网格，同时生成实例mask。mask层的深度等于$S \times S$，即每个网格都有机会参与实例mask的生成。

Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, Lei Li. SOLO: Segmenting Object by Locations. ECCV 2020.

# 4.3 Solo: Segmenting Object by Location




***缺点***

1. 事先使用超参数$S \times S$进行划分定义，导致模型**对数据集有较强的依赖**，但是，当Grid的感受野与大多数目标具有相同尺度时具有较好的性能；若数据集目标对象尺度差异较大，小目标将会难以被识别。
2. mask分支的性能不是特别理想，表示和学习不充分
3. 由于每个Grid**只对应于一个目标类**，对于**密集目标**的分割Solov1效果不是很好（参考YoloV1）
4. 推理阶段，需要对**每个类**都执行mask-NMS，**速度较慢**

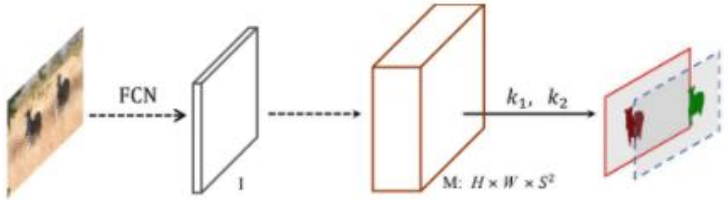
Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, Lei Li. SOLO: Segmenting Object by Locations. ECCV 2020.

# 4.3 Solo: Segmenting Object by Location

## SOLOv2: Dynamic, Faster and Stronger

### *SOLOv2的改进（创新点）*

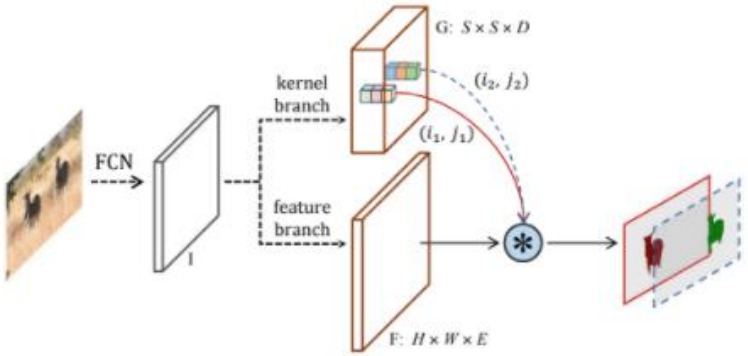
1. 使用**动态卷积**的思想，主动学习mask核，使得网络能够根据位置信息动态地分割实例，有效提高mask的性能。
2. 提出**Matrix NMS**，大大提高了预测阶段的后处理速度。




(a) SOLO

### *SOLOv2的优点*

1. 设计简单、优雅
2. Proposal-Free，速度快
3. 大、中目标分割效果好，更加精细；小目标有具有一定的竞争力

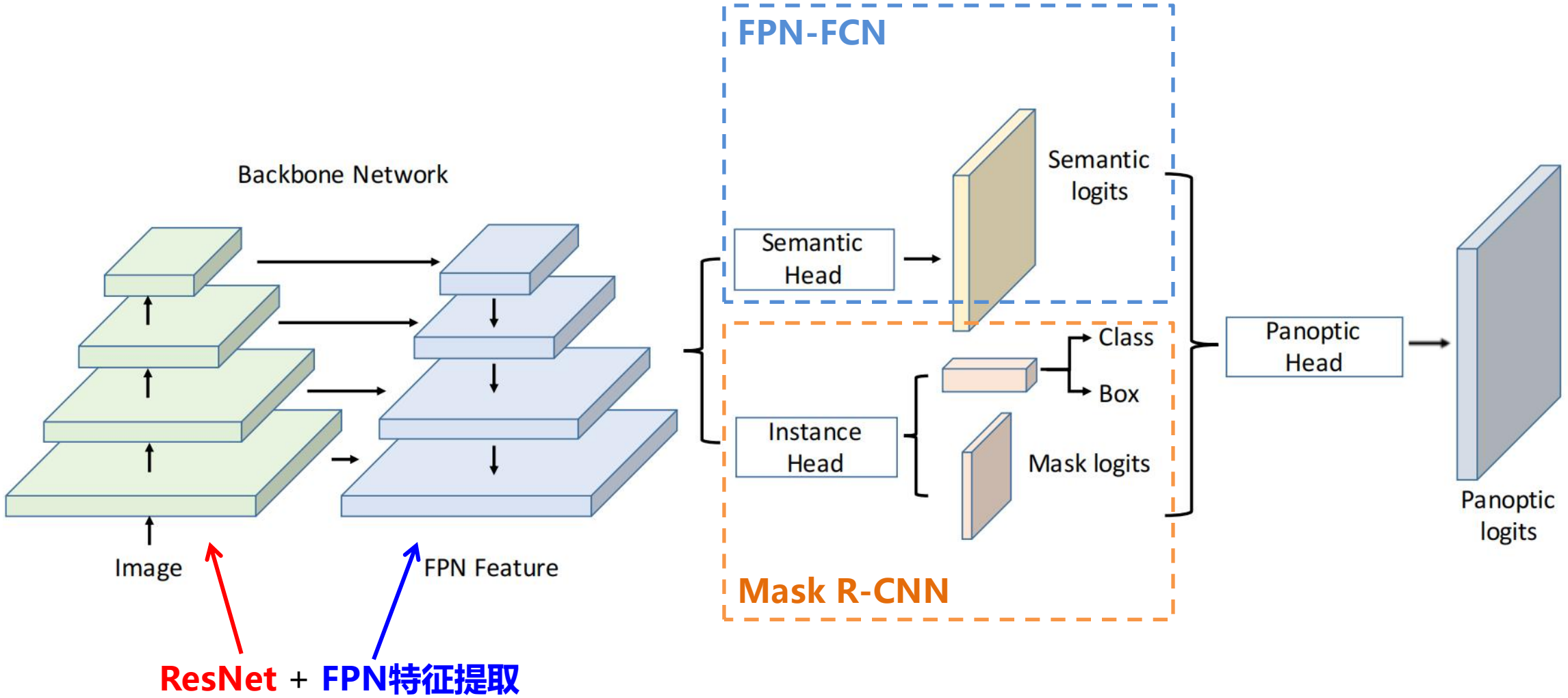



(b) SOLOv2

Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, Chunhua Shen. SOLOv2: Dynamic, Faster and Stronger. NIPS2020.

# 4.4 UPSNet

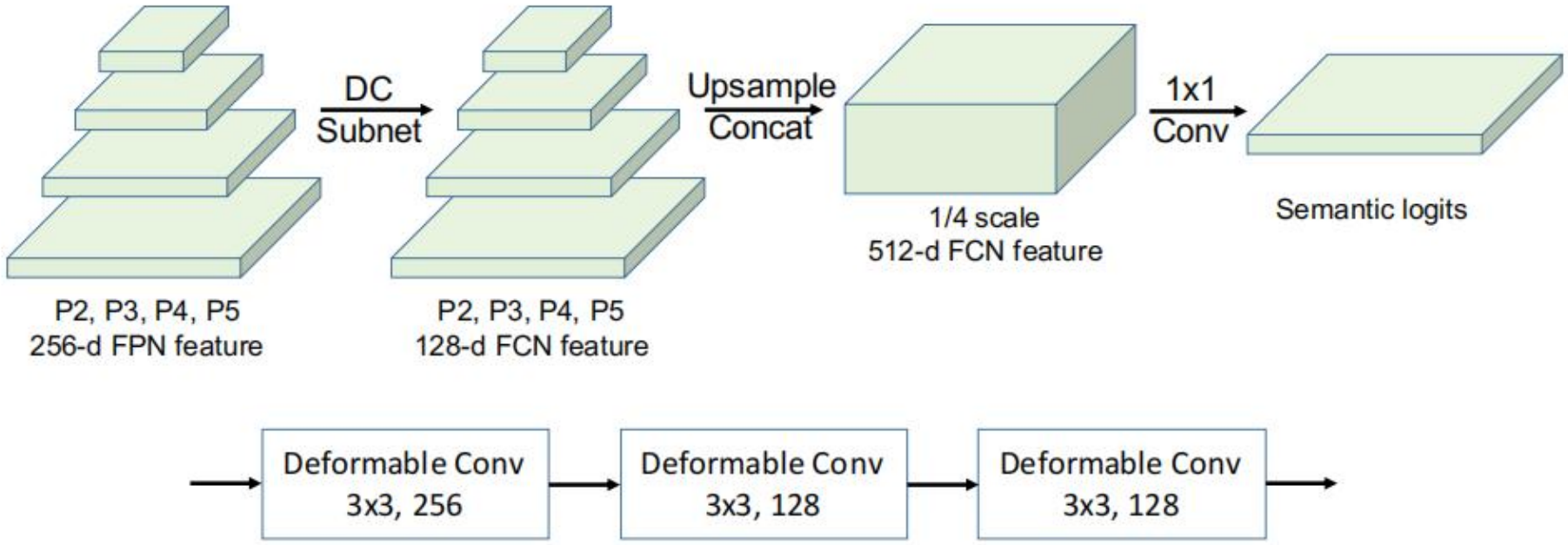
## USPNet总体体系结构




Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersion Tumer, Raquel Urtasun. UPSNet-A Unified Panoptic Segmentation Network. CVPR 2019.

# 4.4 UPSNet

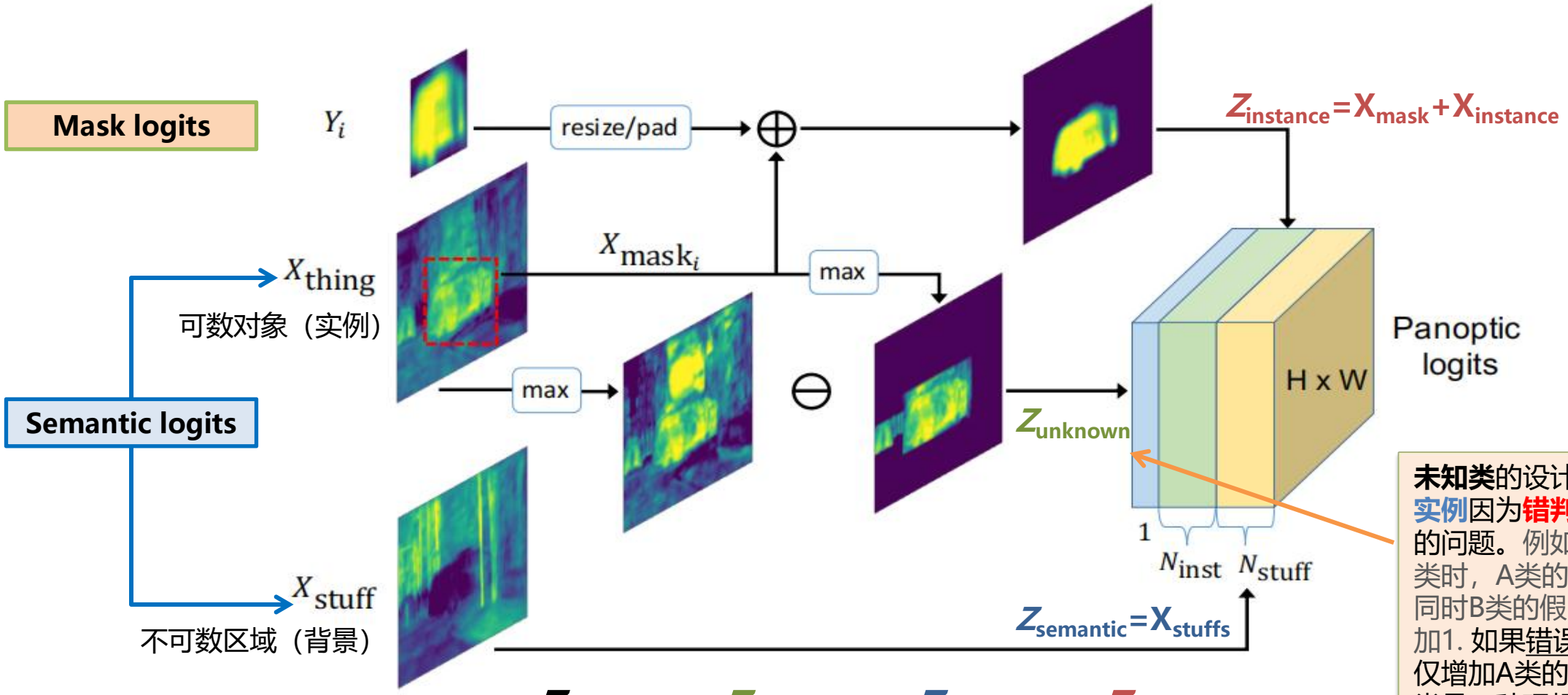
## Semantic Segmentation head: FPN-FCN




Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersion Tumer, Raquel Urtasun. UPSNet-A Unified Panoptic Segmentation Network. CVPR 2019.

# 4.4 UPSNet

## Panoptic Segmentation Head




$$Z_{panoptic} = Z_{unknown} + Z_{semantic} + Z_{instance}$$

**未知类**的设计是为了避免**一个实例**因为**错判**而导致**双重惩罚**的问题。例如：$A_G$被判定为B类时，A类的假负FN将增加1，同时B类的假阳性（FP）也增加1. 如果错误不可避免，那么仅增加A类的FN，而不影响B类是一种理想的状态。

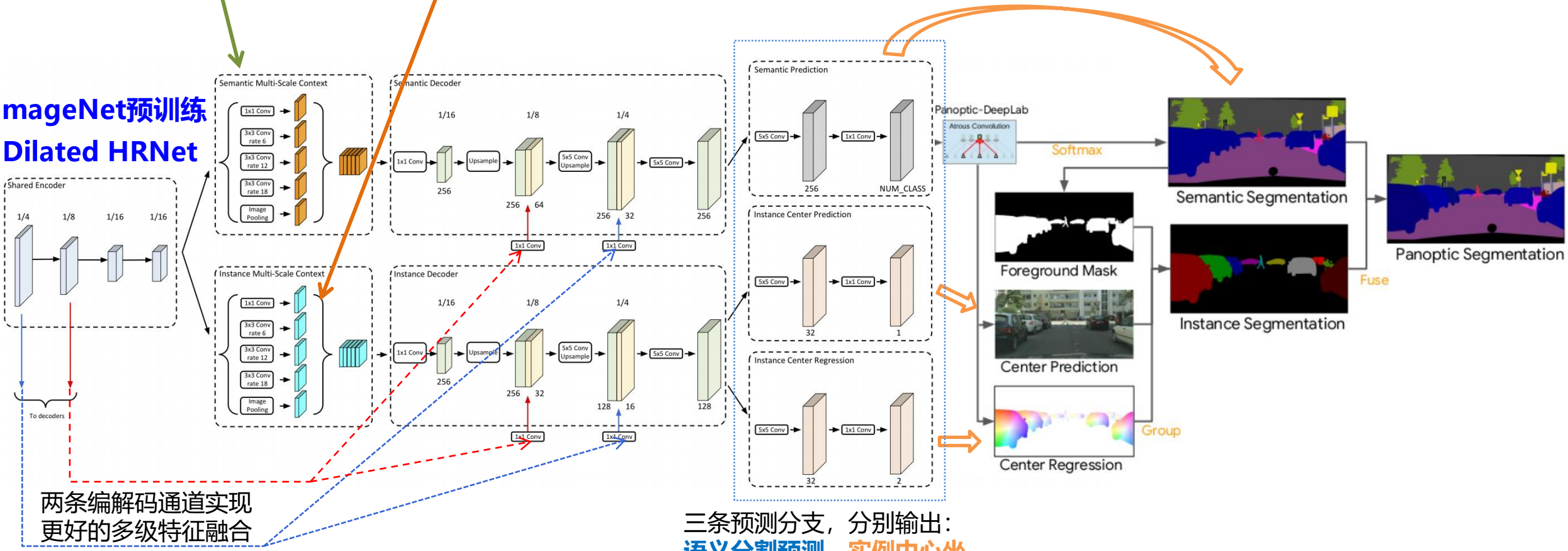
Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersion Tumer, Raquel Urtasun. UPSNet-A Unified Panoptic Segmentation Network. CVPR 2019.

# 4.5 Panoptic-DeepLab

两个独立的ASPP分别用于提取实例多尺度上下文特征和语义多尺度上下文特征

ImageNet预训练

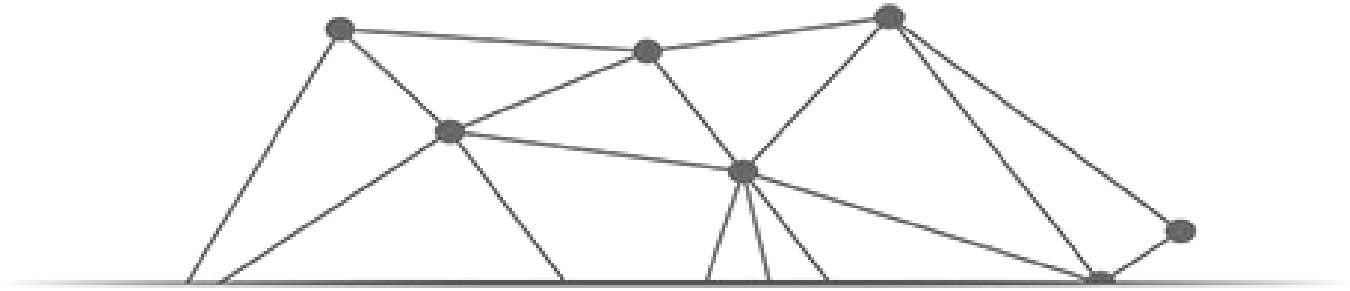
Dilated HRNet



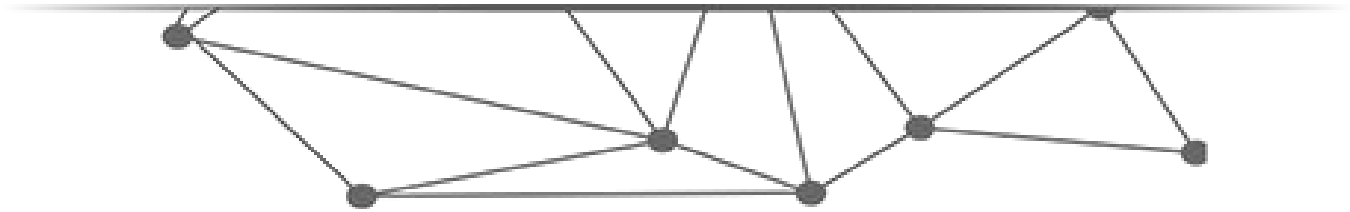

两条编解码通道实现更好的多级特征融合

三条预测分支，分别输出：语义分割预测、实例中心坐标及实例区域回归

Bowen Cheng, Maxwell D.Colins, Yukun Zhu, Ting Liu, Thomas S. Huang, Hartwig Adam, Liang-Chieh Chen. Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation. CVPR 2020.



# 课堂互动 13.3.6



# 欧老师的联系方式

读万卷书 行万里路 只为最好的修炼


QQ：14777591 (宇宙骑士)

Email：ouxinyu@alumni.hust.edu.cn

Website: http://ouxinyu.cn

Tel：18687840023

地址：安宁校区 诚远楼201

南院 智能应用研究院A306-2