

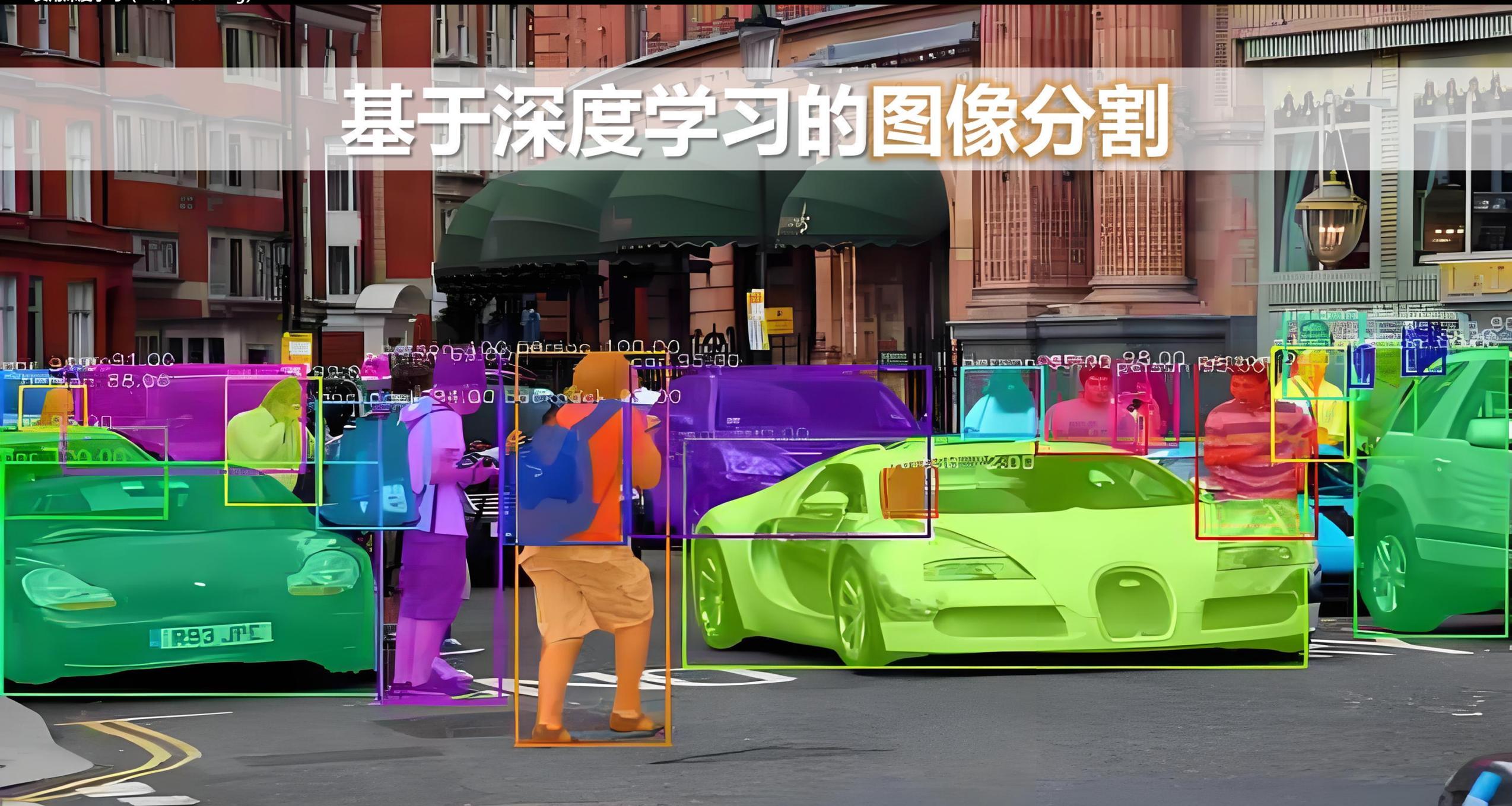
第13章 计算机视觉

传媒与信息工程学院

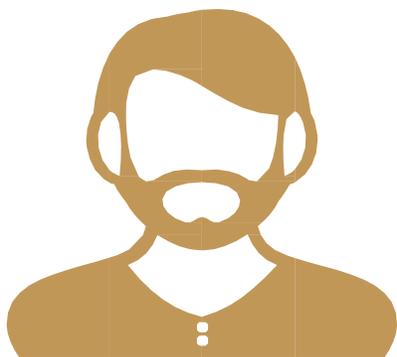
欧新宇



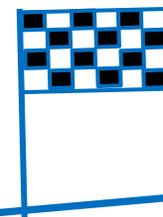
基于深度学习的图像分割



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



播放



00:00 / 00:47

倍速



1.1 图像分割概述

什么是图像分割?



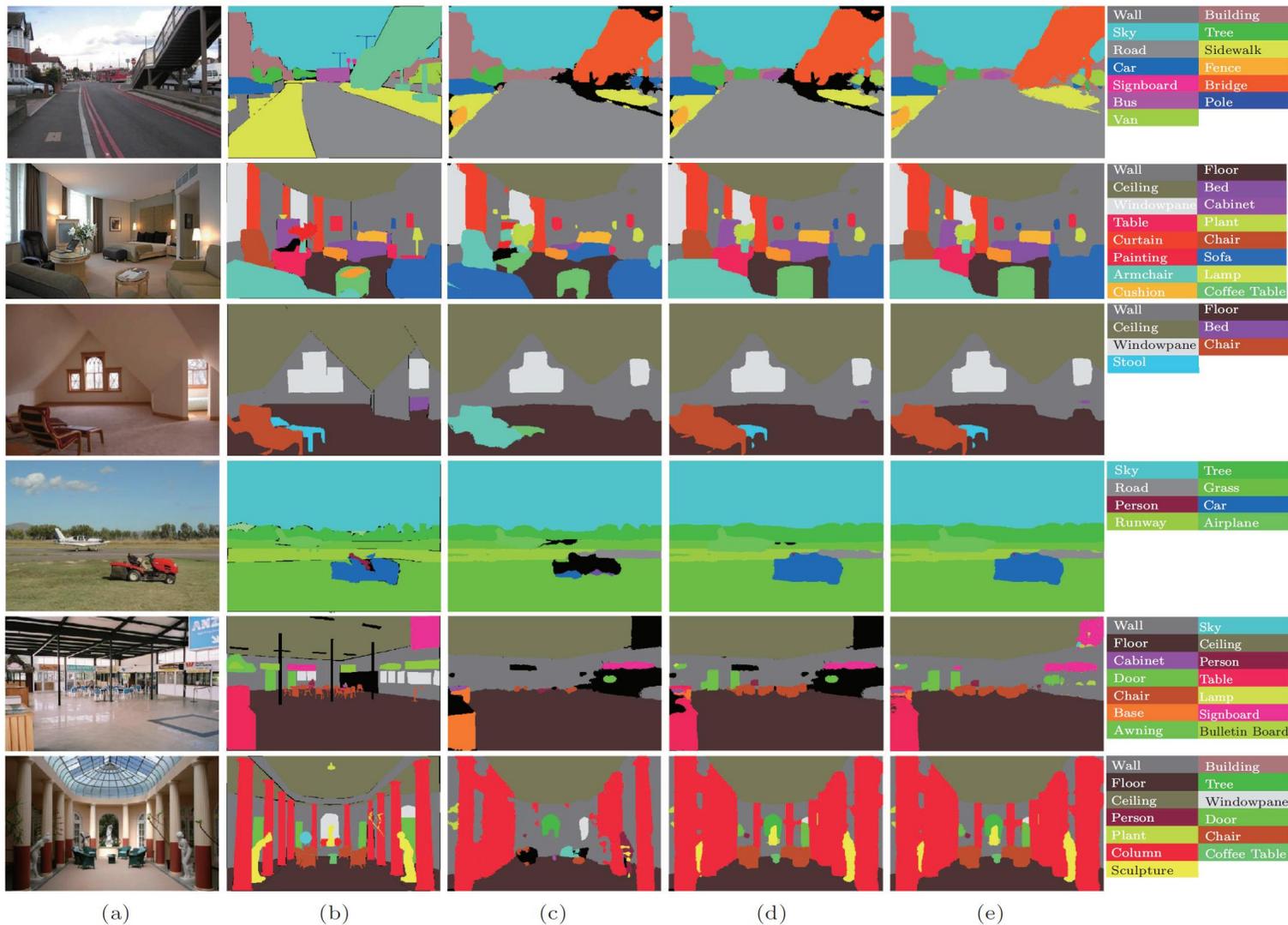
1.1 图像分割概述

什么是图像分割?



1.1 图像分割概述

什么是图像分割?



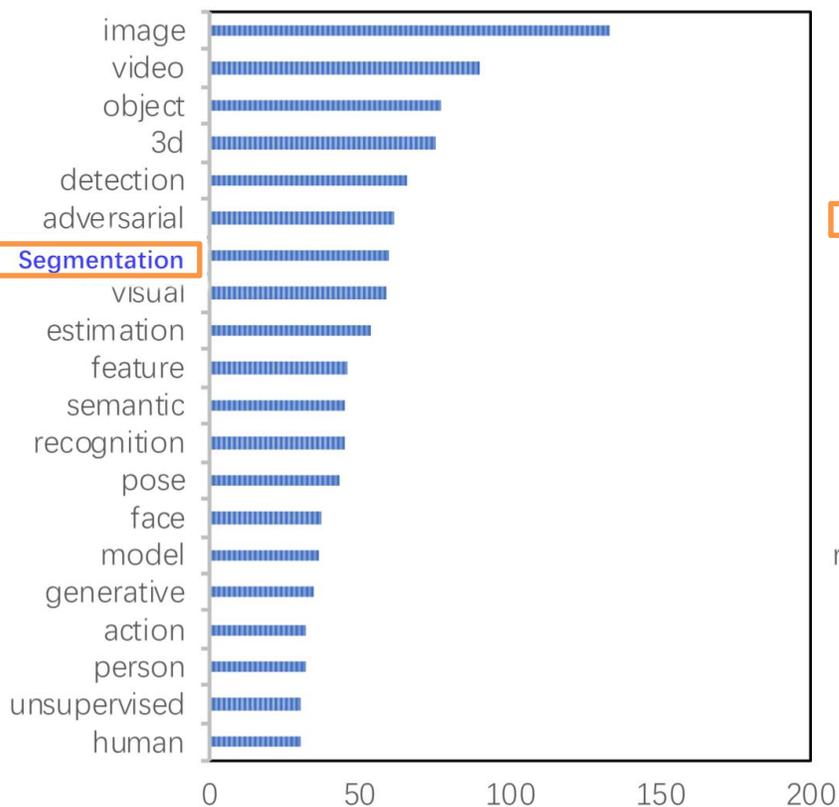
Examples of scene parsing on the *SceneParse150* dataset.

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

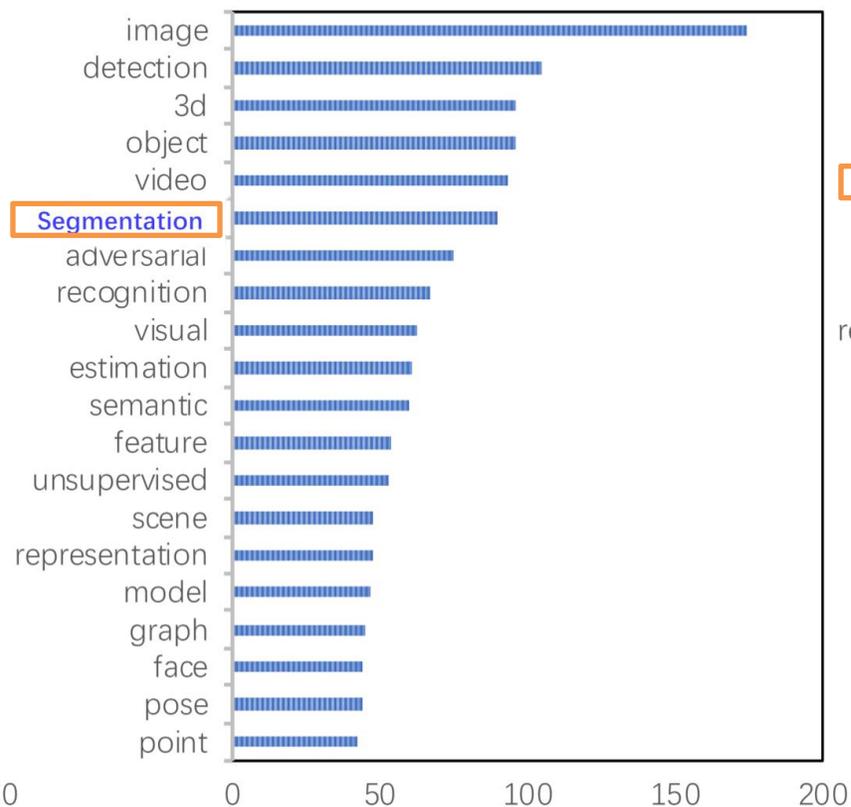
1.1 图像分割概述

图像分割在计算机视觉任务中的位置

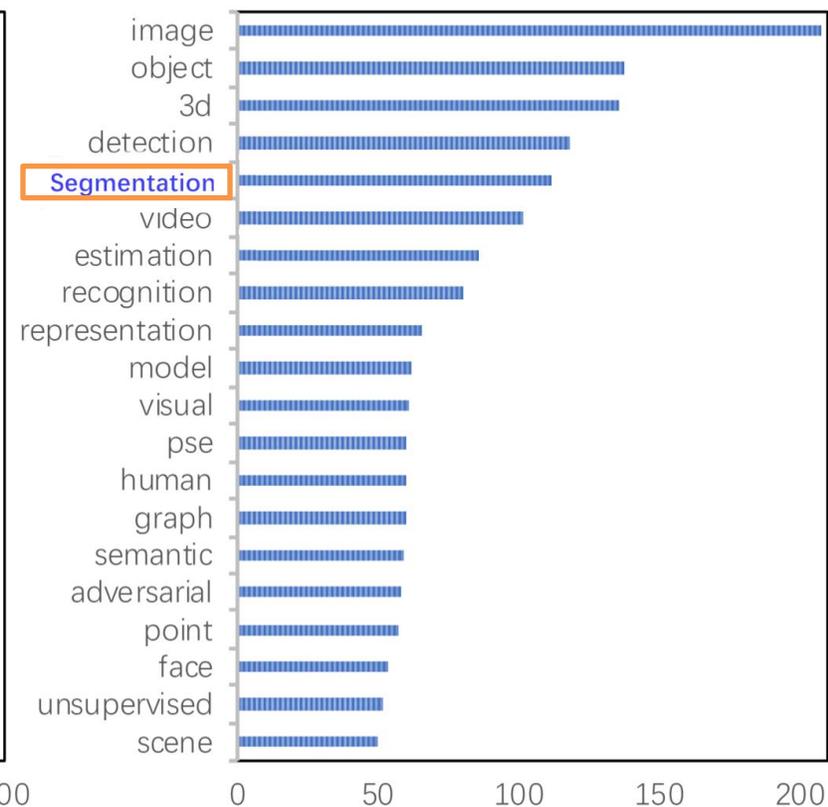
CVPR2018



CVPR2019



CVPR2020



1.1 图像分割概述

问题定义

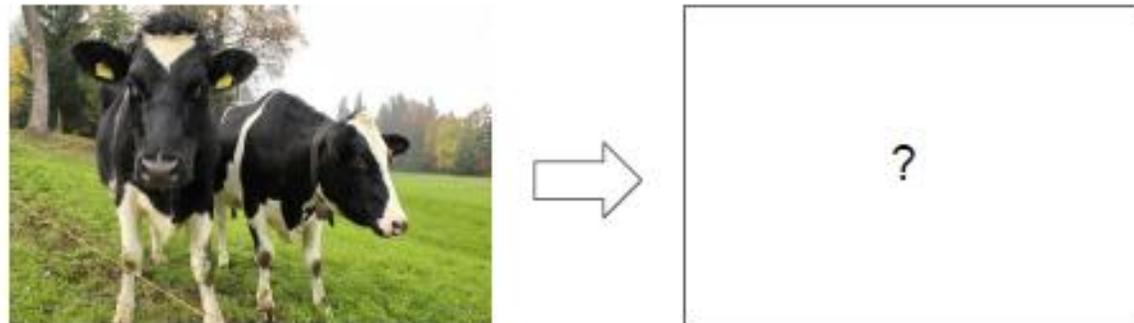
训练阶段



GRASS, CAT,
TREE, SKY, ...

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

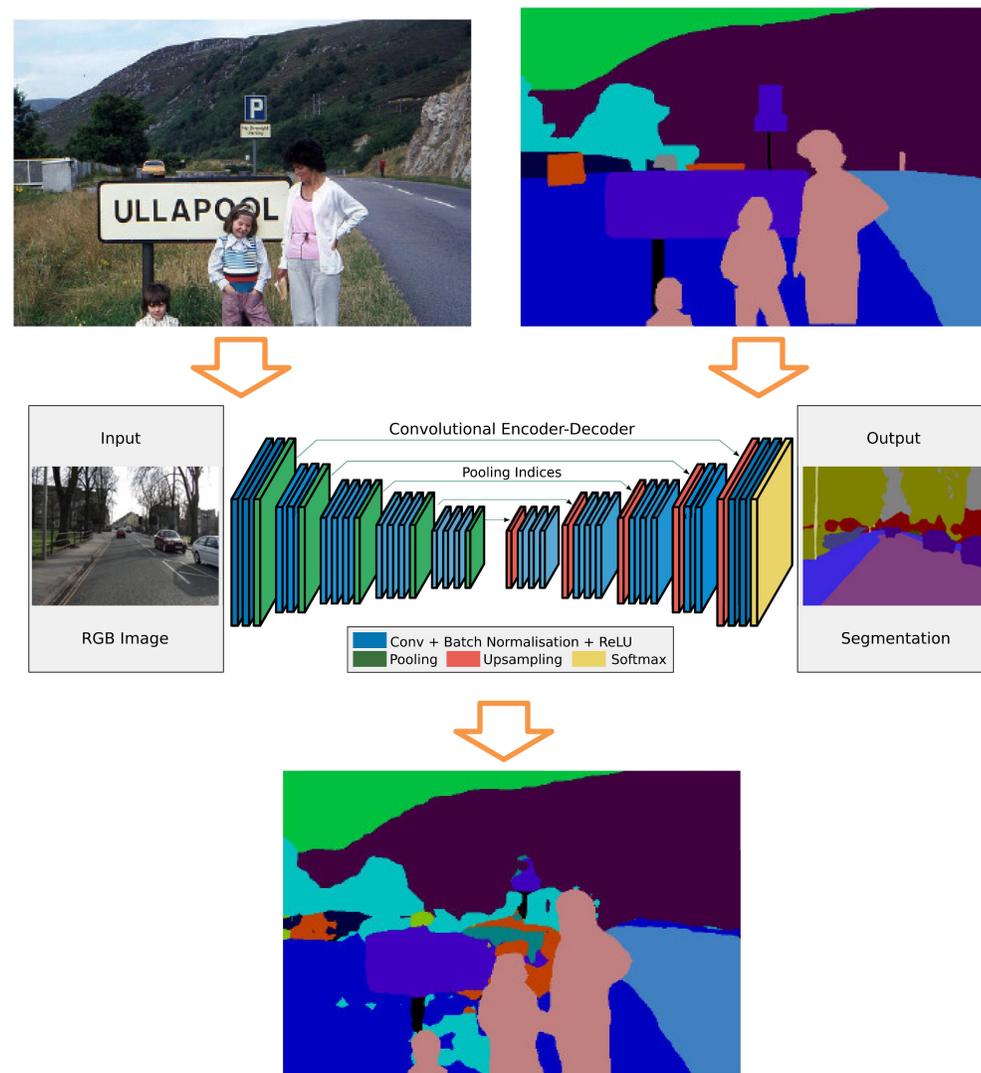


在推理阶段, 对于每个新图像都逐像素进行分类。

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)`



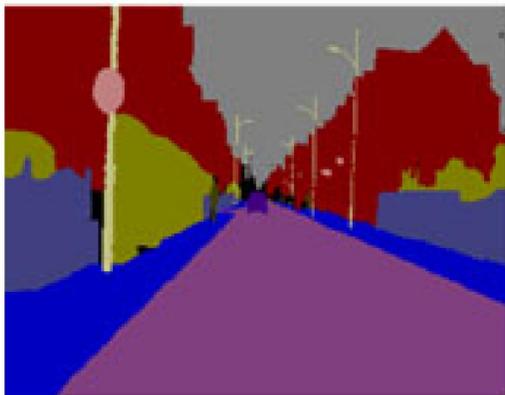
1.1 图像分割概述

图像分割的根本目的

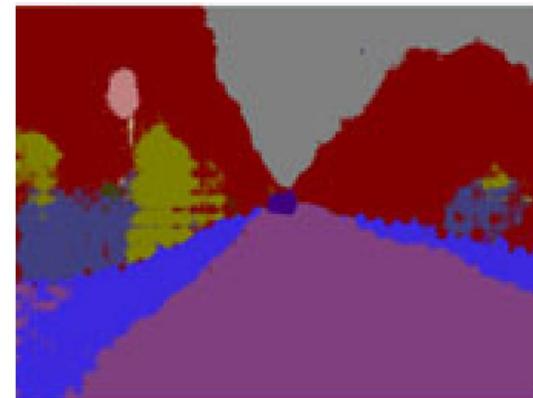
图像分割的根本目的：像素级分类



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

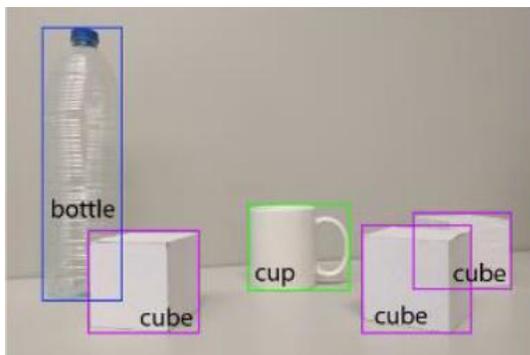
图像是由**像素点**构成的矩阵，**图像分割**等同于**像素级分类**，并将**相同类的像素**组合在一起。图像分割包含三种典型任务**语义分割**、**实例分割**和**全景分割**。

- **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

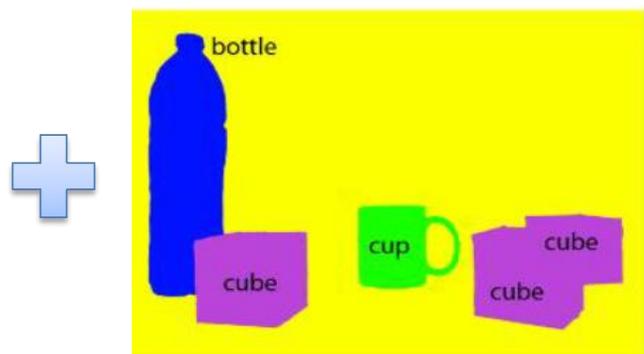
语义分割、实例分割和全景分割

目标检测



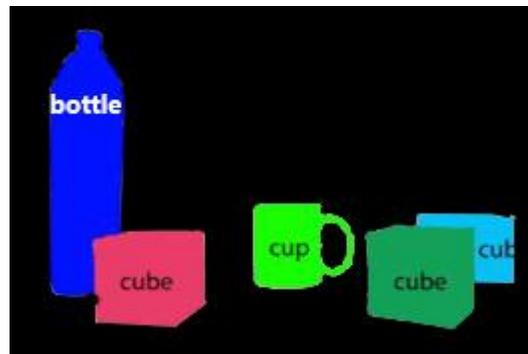
+

语义分割

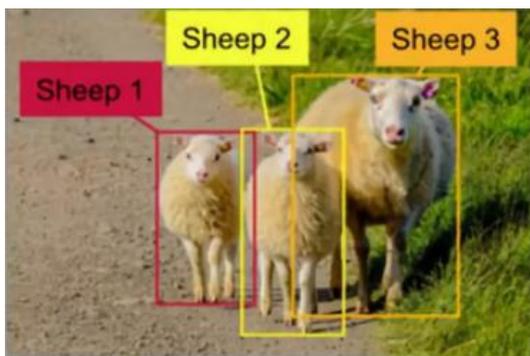
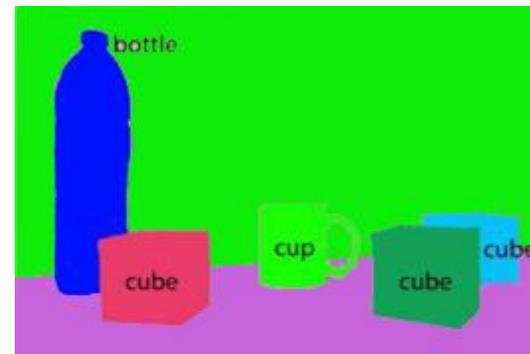


=

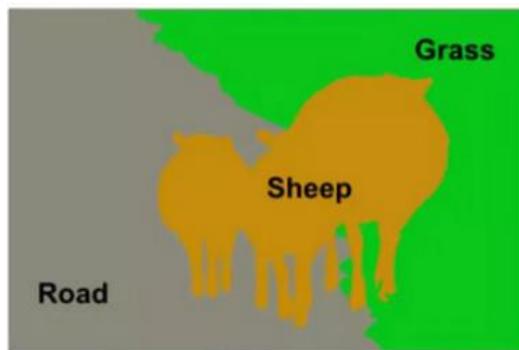
实例分割



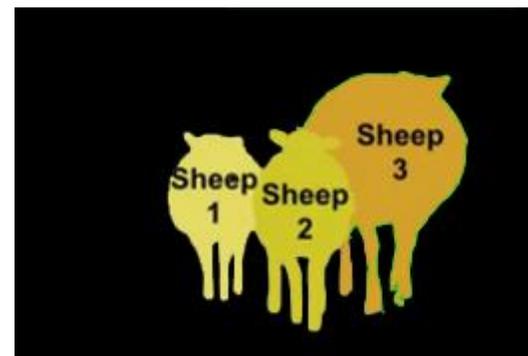
全景分割



+



=



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

1.2 图像分割的类型

视频目标分割



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。

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

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



1.3 图像分割的应用

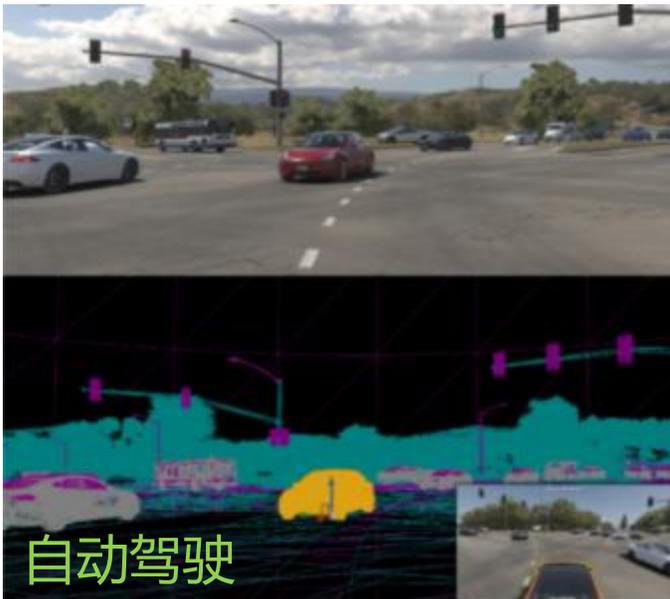
图像分割的应用

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。

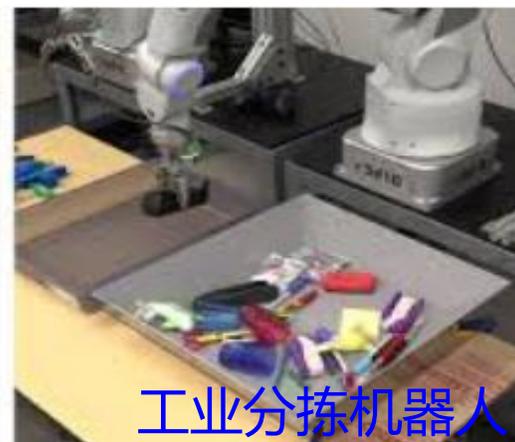
- 人像分割：头发分割、人脸分割、背景分割
- 自动驾驶：行人检测、车辆分割、车道线检测
- 医学图像：病例图、CT、MRI
- 工业质检：工业机器人、分拣机器人

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琳琅满目的，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



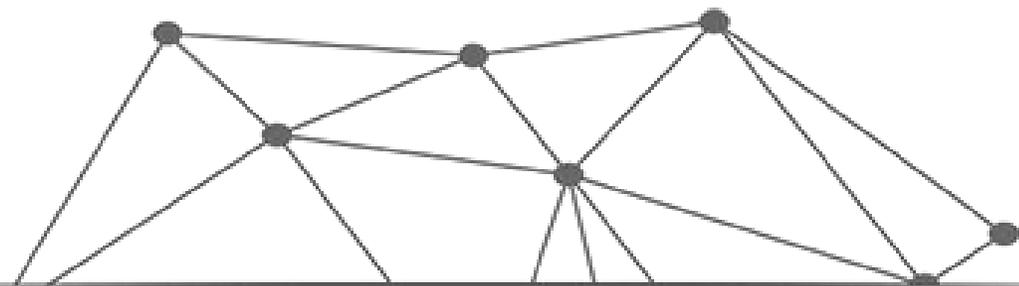
MS COCO



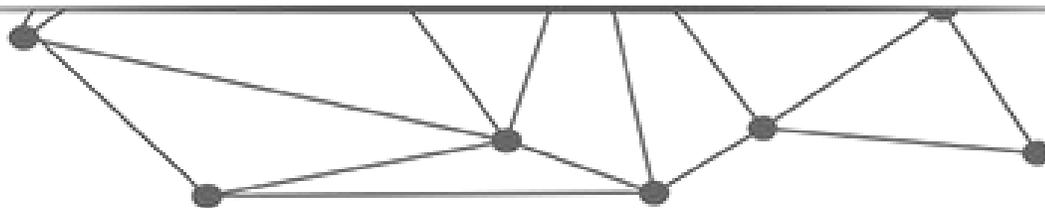
Pascal Context



Cityscapes



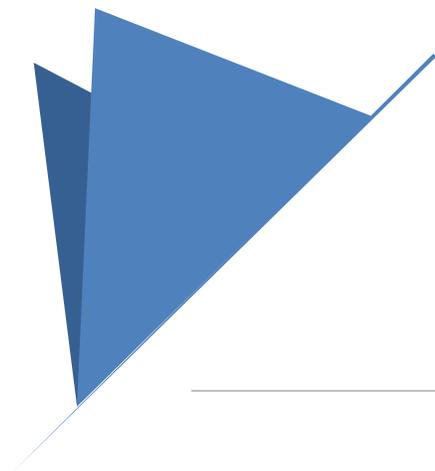
课堂互动 13.3.1



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

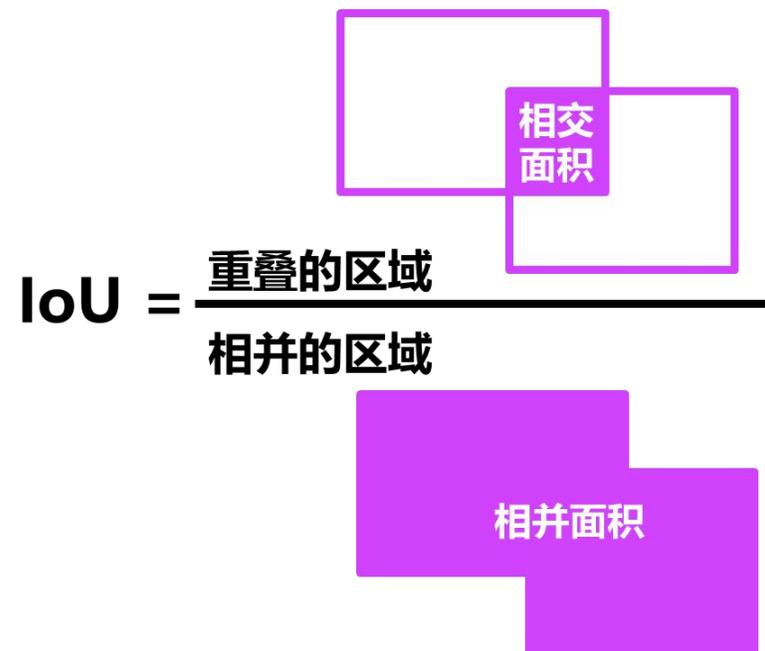
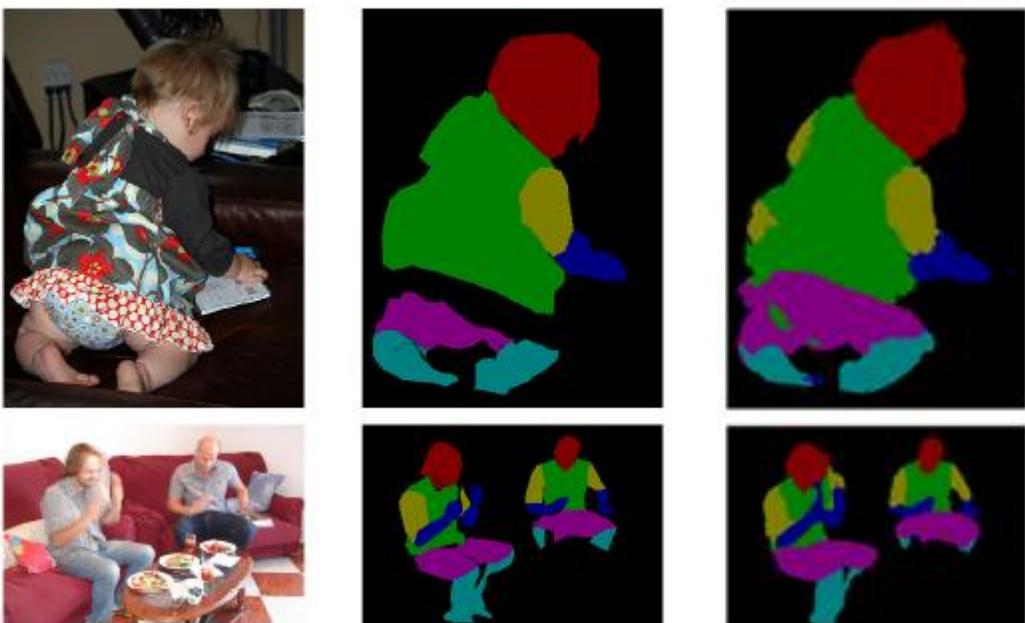


哪一个输出结果更好?

2.1 图像分割的评价指标

mIoU和mAcc

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



IoU: 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

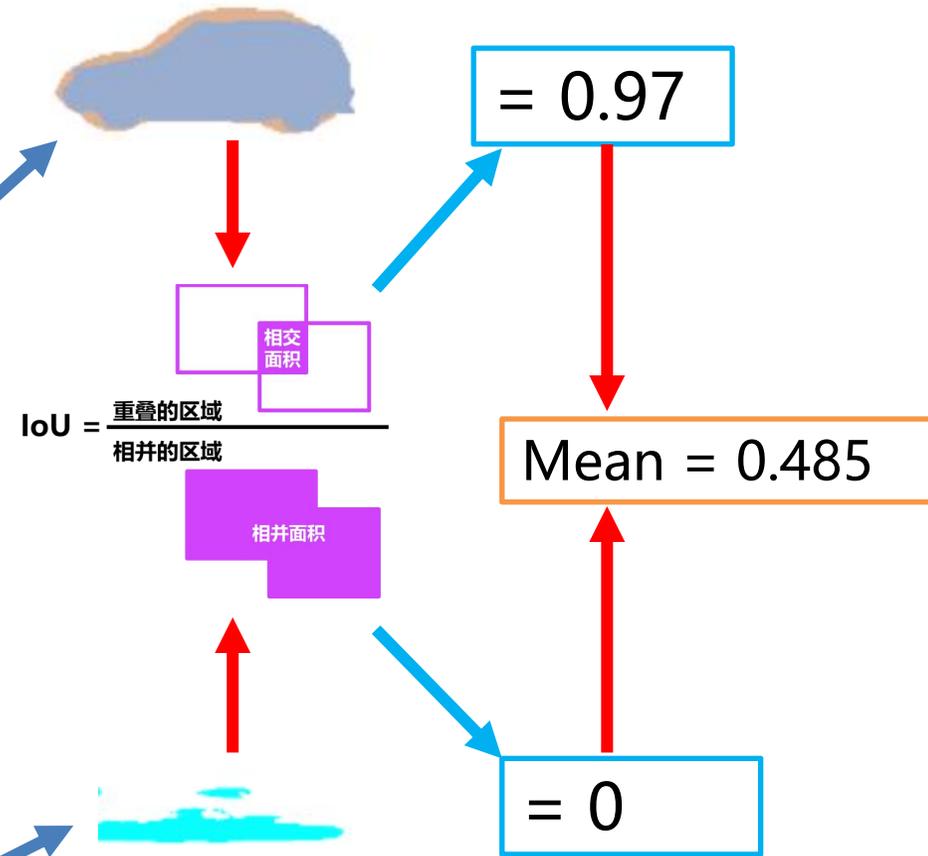
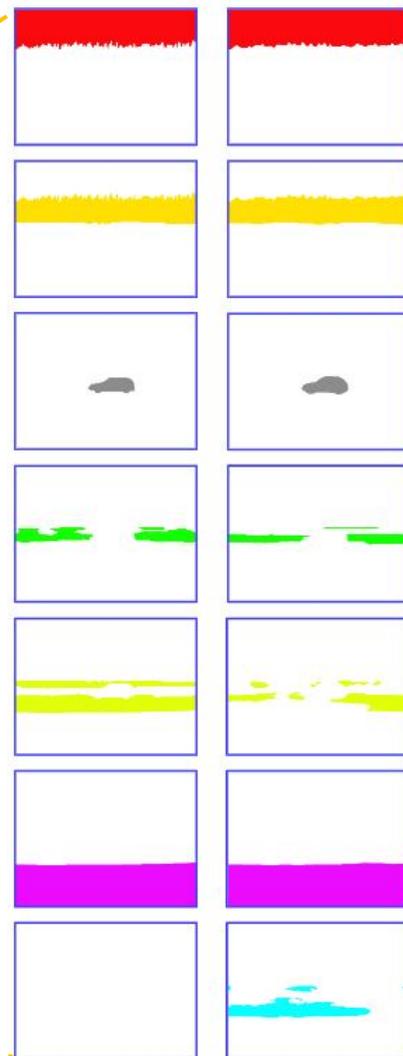
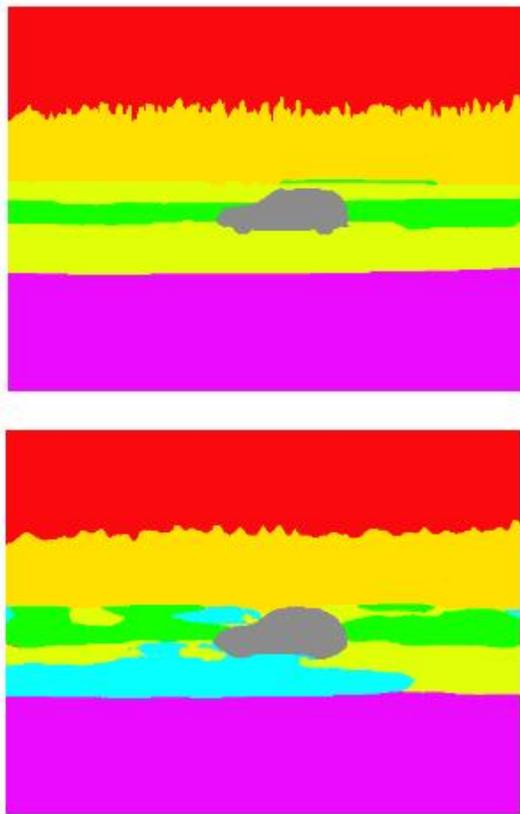
mACC: 所有类ACC的平均值

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



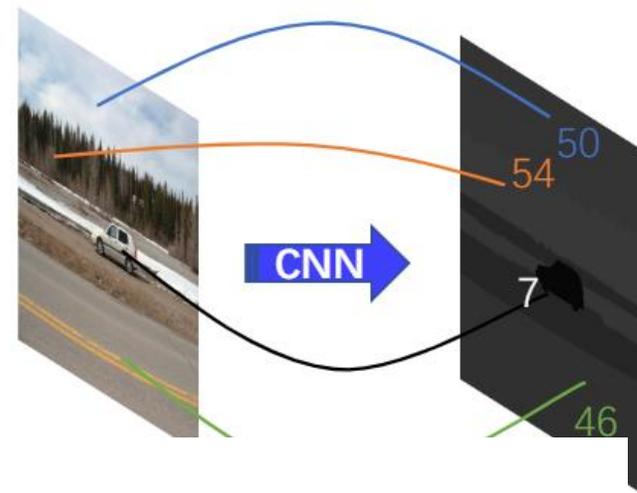
2.1 图像分割的评价指标

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

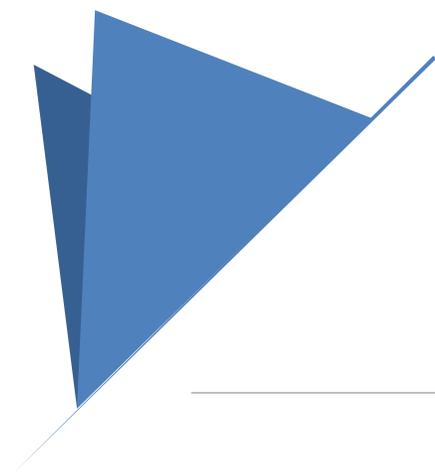


50	50	50	50	50
54	54	54	54	54
37	37	7	37	51
46	46	46	46	46
46	46	46	46	46



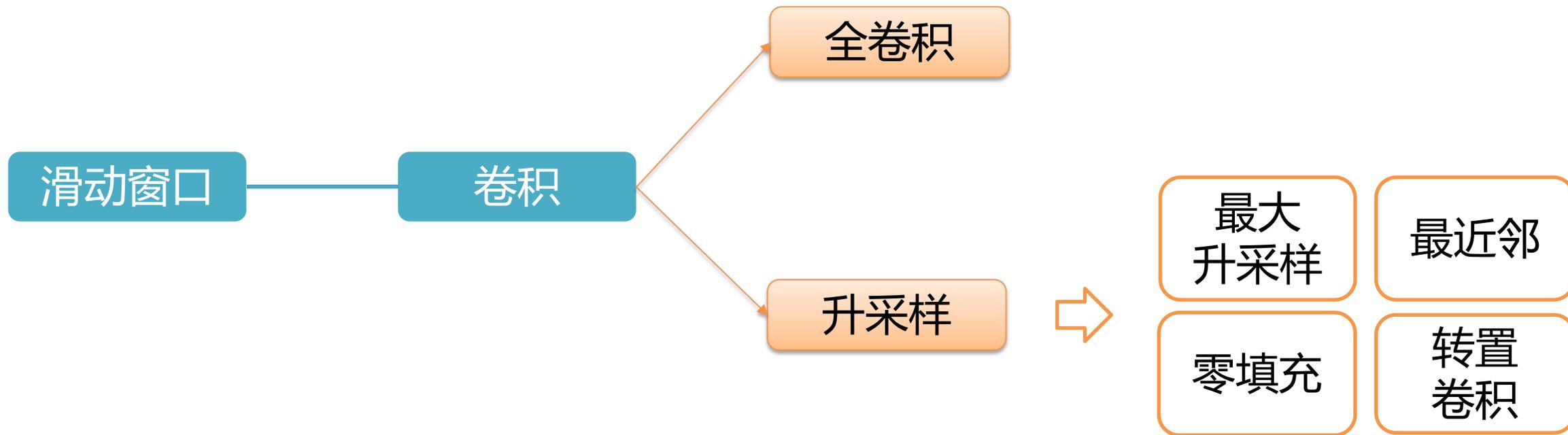
50	50	50	50	50	...	46	46
----	----	----	----	----	-----	----	----

精确度



图像分割的关键技术

2.2 图像分割关键技术



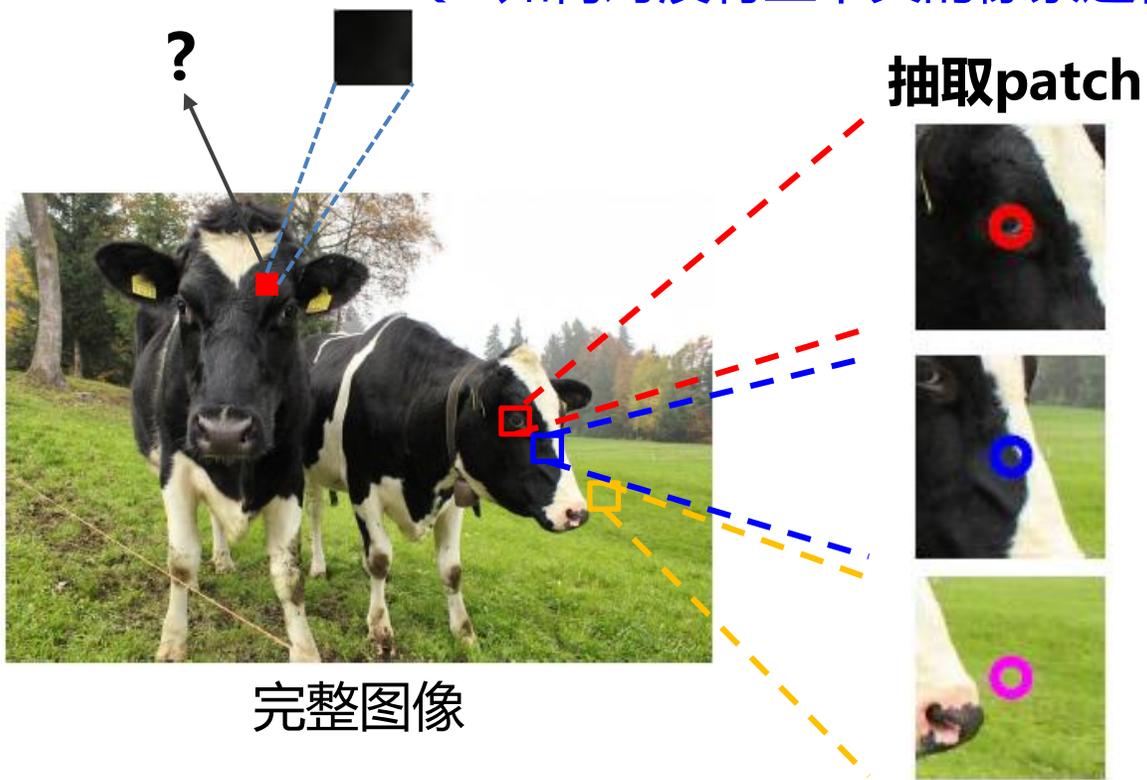
2.2 图像分割关键技术

滑动窗口 Sliding Window

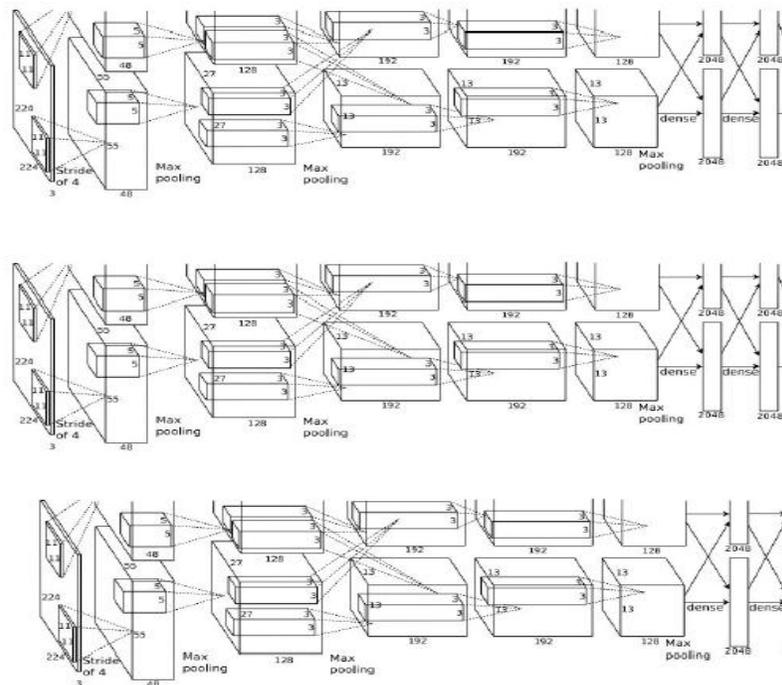
像素没有上下文信息无法进行分类

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。



使用CNN分类中心像素



奶牛

奶牛

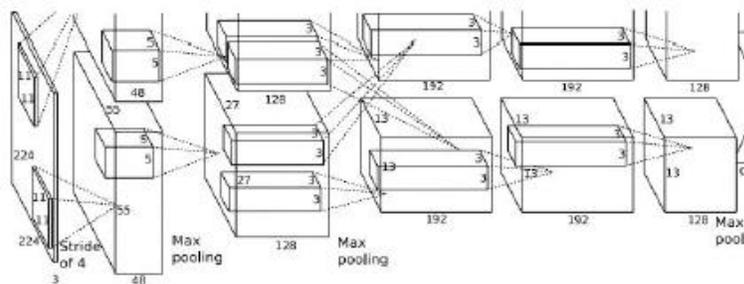
草

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**。

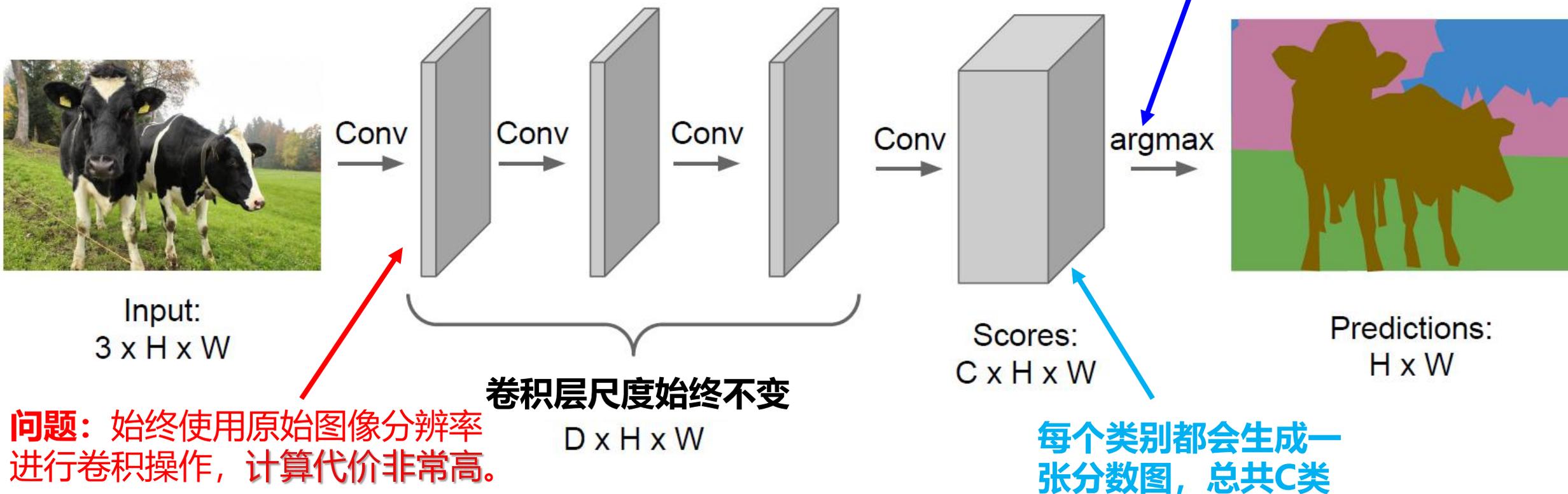
然而，图像分割要求输出的**预测图**和**输入图像**具有相同的尺寸。

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。

激活值最大的通道的索引作为最终的类别的索引



2.2 图像分割关键技术

全卷积网络 Fully Convolutional

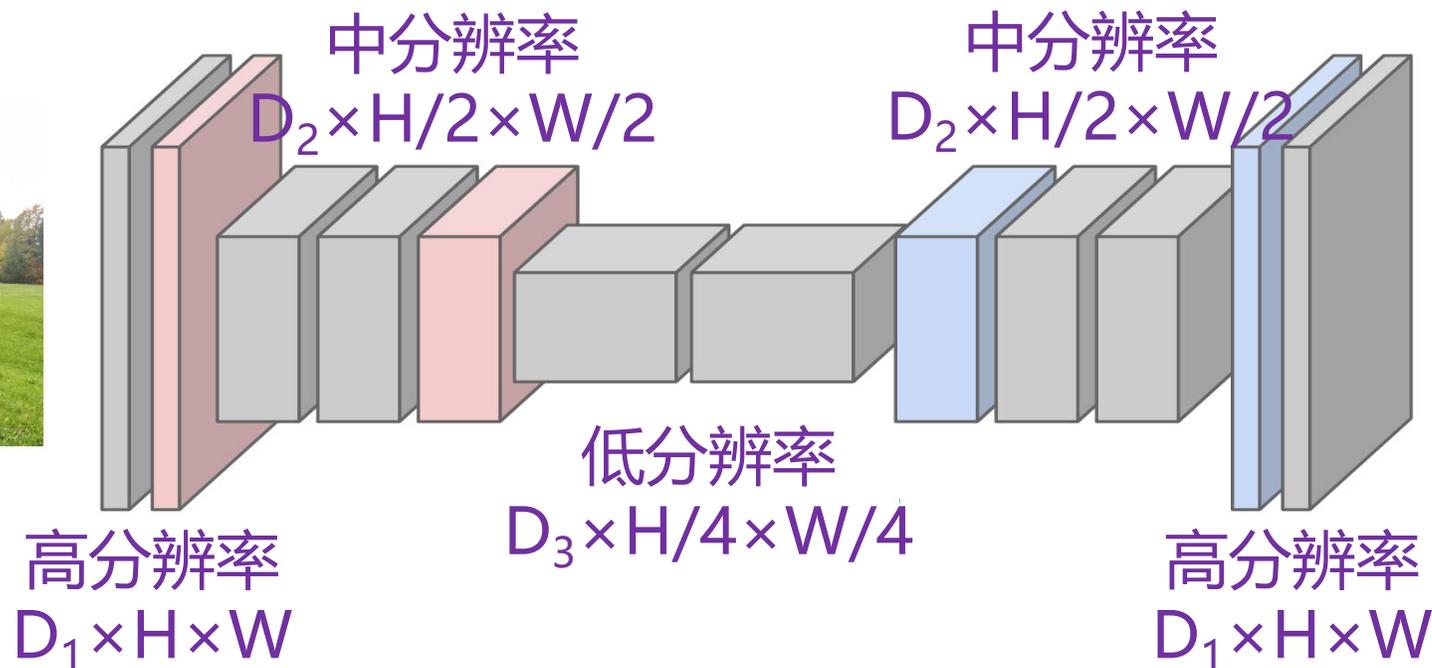
设计一个包含一系列卷积层的网络 (不含全连接层), 并在中间同时集成升采样和降采样结构!

降采样:

Pooling, strided convolution



输入图像
 $3 \times H \times W$



升采样:

???



预测图像
 $H \times W$

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**。

Nearest Neighbor

1	2
3	4



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

Input: 2 x 2

Output: 4 x 4

“Bed of Nails”

1	2
3	4



1	0	2	0
0	0	0	0
3	0	4	0
0	0	0	0

Input: 2 x 2

Output: 4 x 4

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

1	2	6	3
3	5	2	1
1	2	2	1
7	3	4	8

Input: 4 x 4

5	6
7	8

Output: 2 x 2

网络的其他部分

Max UnPooling

调用Pooling层的位置信息进行升采样

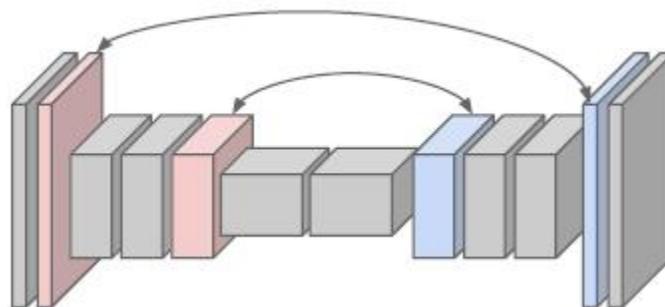
1	2
3	4

Input: 2 x 2

0	0	2	0
0	1	0	0
0	0	0	0
3	0	0	4

Output: 4 x 4

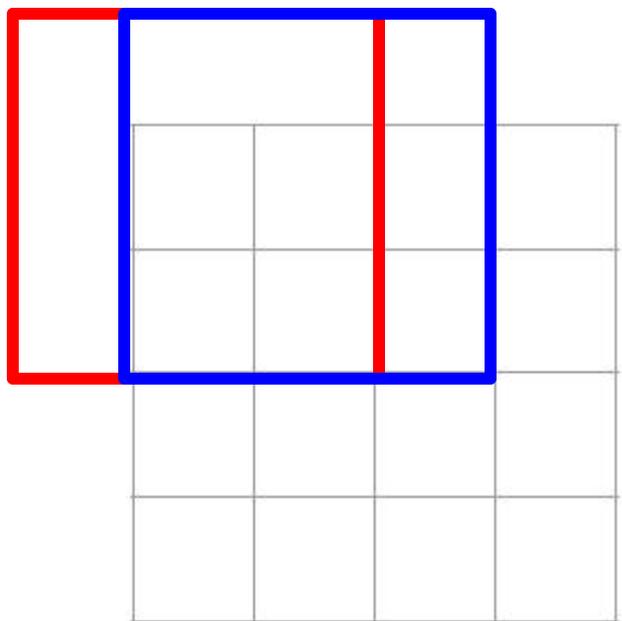
升采样层和降采样层互为关联对



2.2 图像分割关键技术

可学习的升采样 – 转置卷积

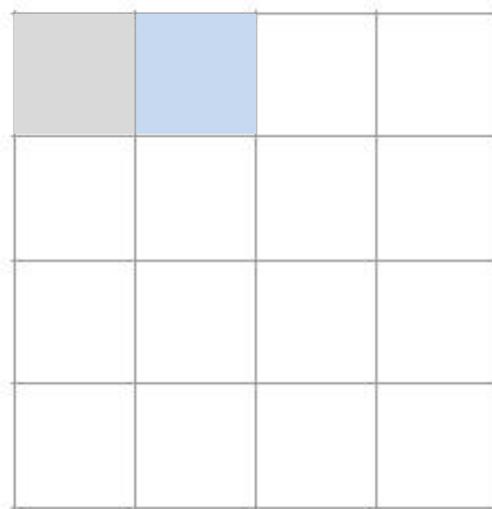
回顾：普通 3×3 卷积，步长：1，填充：1



Input: 4 x 4



输入和卷积核
之间执行点乘

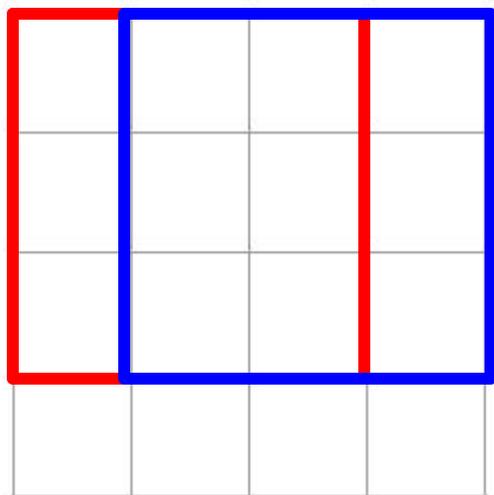


Output: 4 x 4

2.2 图像分割关键技术

可学习的升采样 – 转置卷积

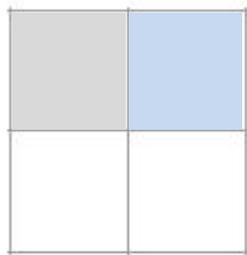
回顾：普通 3×3 卷积，步长：2，填充：1



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。

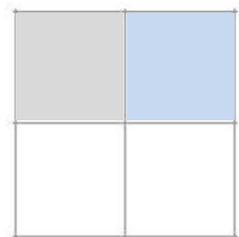
我们可以把这种跨步长的卷积称为“可学习降采样”。

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

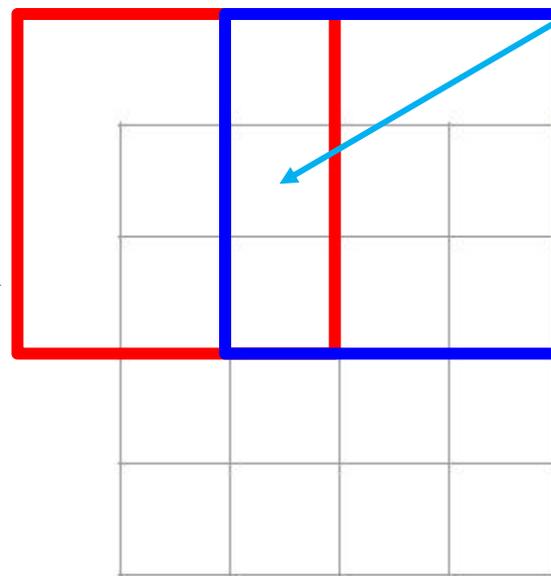
- 反卷积 (Deconvolution)(bad)
- 升卷积 (Upconvolution)
- 分数步长卷积 (Fractionally strided convolution)
- 后馈步长卷积 (Backward strided convolution)



Input: 2 x 2

3×3 转置卷积, 步长: 2, 填充: 1

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

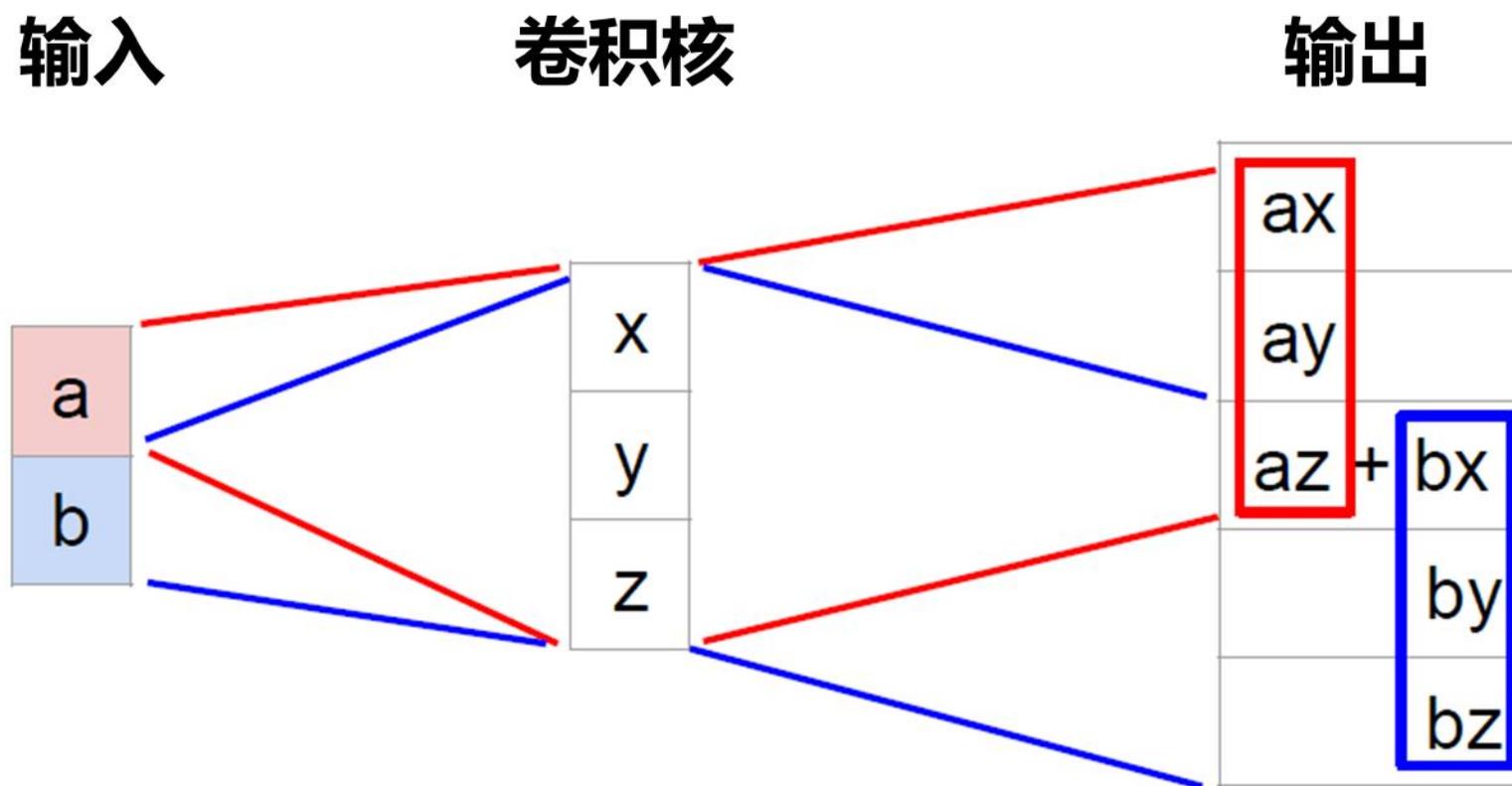
输入每移动1个像素, 卷积核在输出中移动2个像素

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。

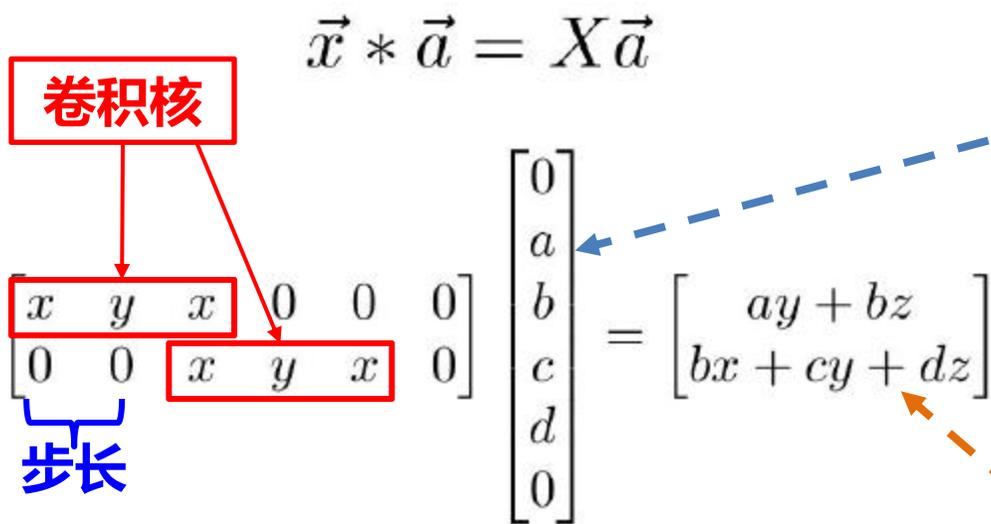
卷积可以理解为向
量的内积操作。

2.2 图像分割关键技术

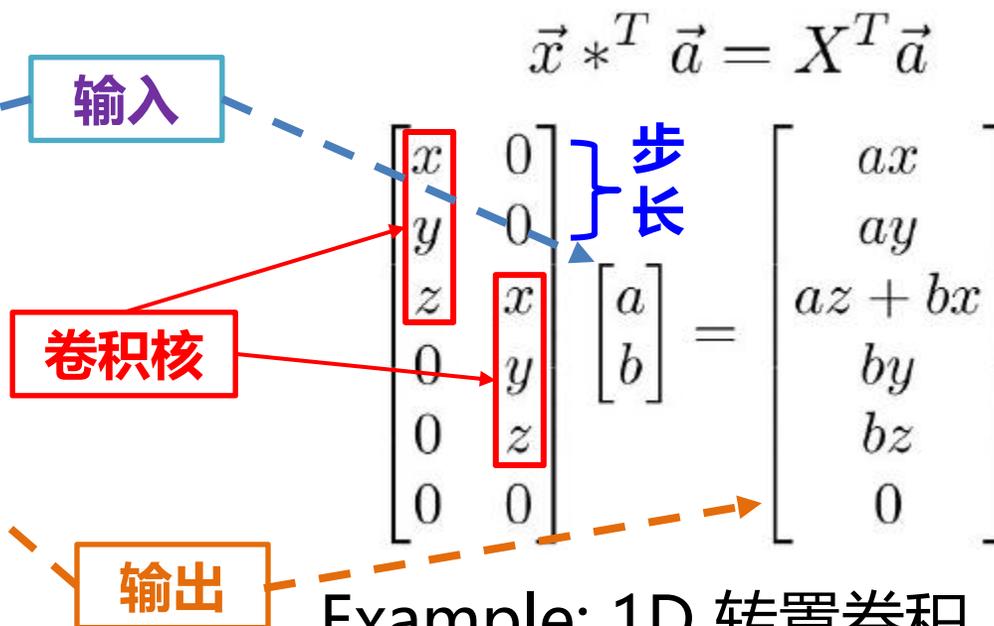
可学习的升采样 - 1维运算案例

卷积运算可以表示为矩阵乘法

卷积转置可以表示为将卷积核转置后的矩阵乘法



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



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

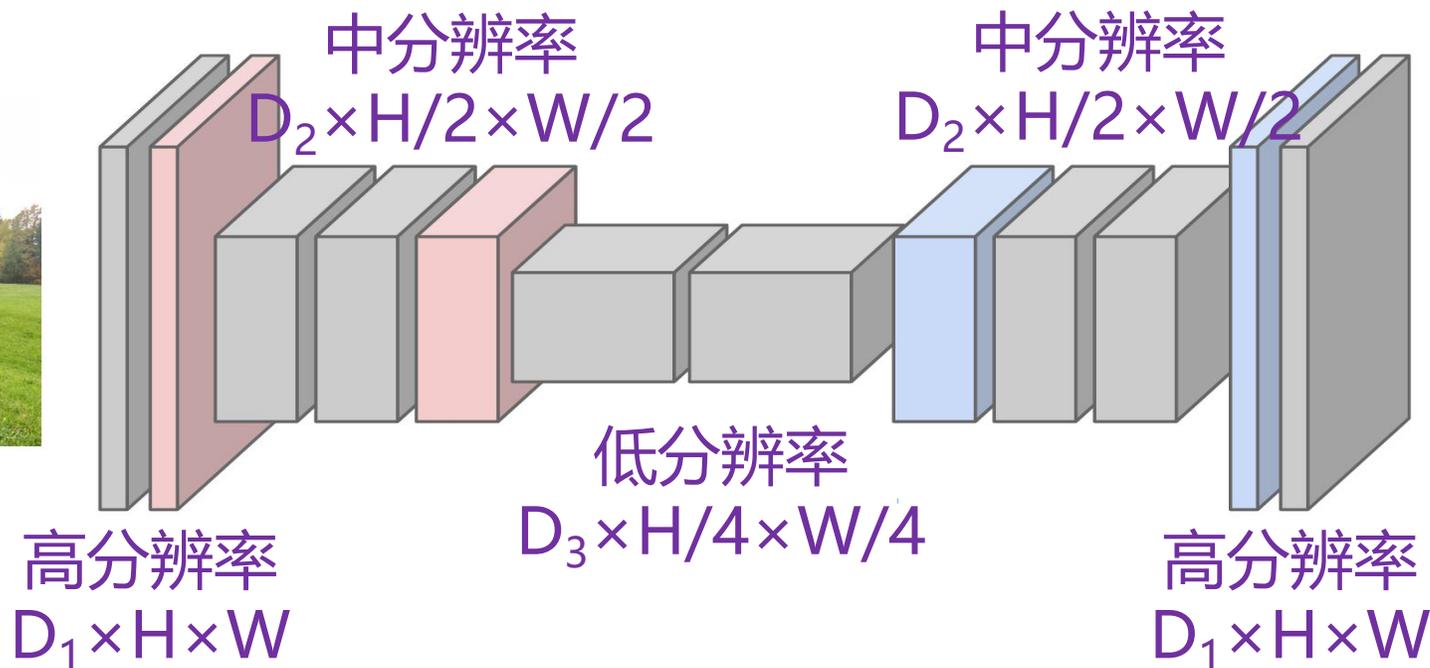
设计一个包含一系列卷积层的网络，并在中间同时集成升采样和降采样结构！

降采样:

Pooling, strided convolution



输入图像
 $3 \times H \times W$



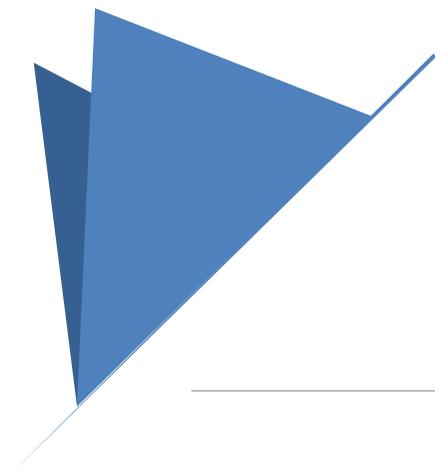
升采样:

最大升采样或转置卷积



预测图像
 $H \times W$

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

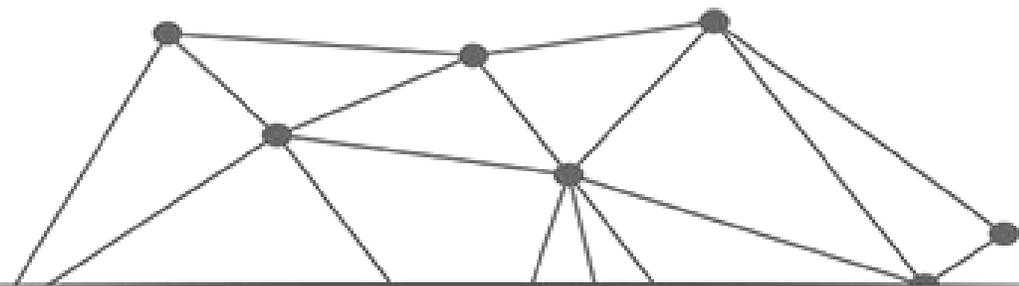
利用阈值对图像进行分割，方法简单粗暴，但不够精确。阈值的设定分为全局阈值和局部阈值两种。

● 基于像素聚类的分割方法

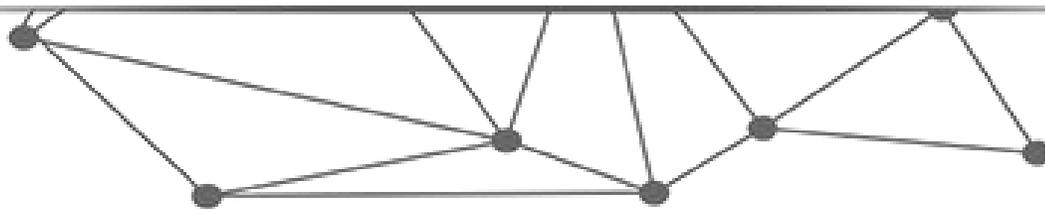
聚类方法先选定一部分像素作为聚类中心，然后根据相邻像素之间的关系进行元素聚类；调整聚类中心反复聚类，直到分割完成且损失函数小于阈值

● 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

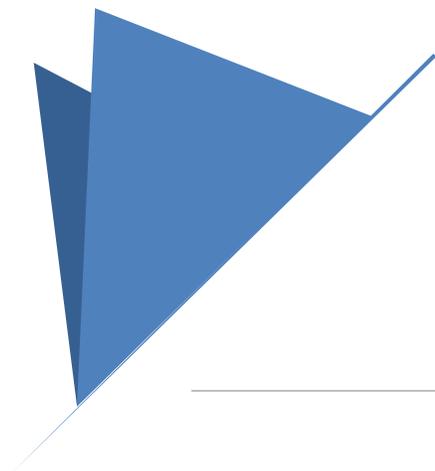
/ U-Net/PSPNet/SegNet 模型

/ 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

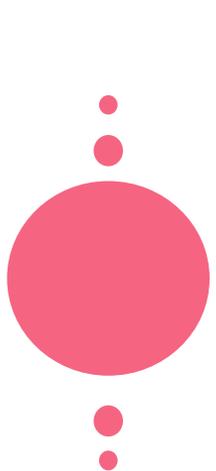




FCN 全卷积网络

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的上采样

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?

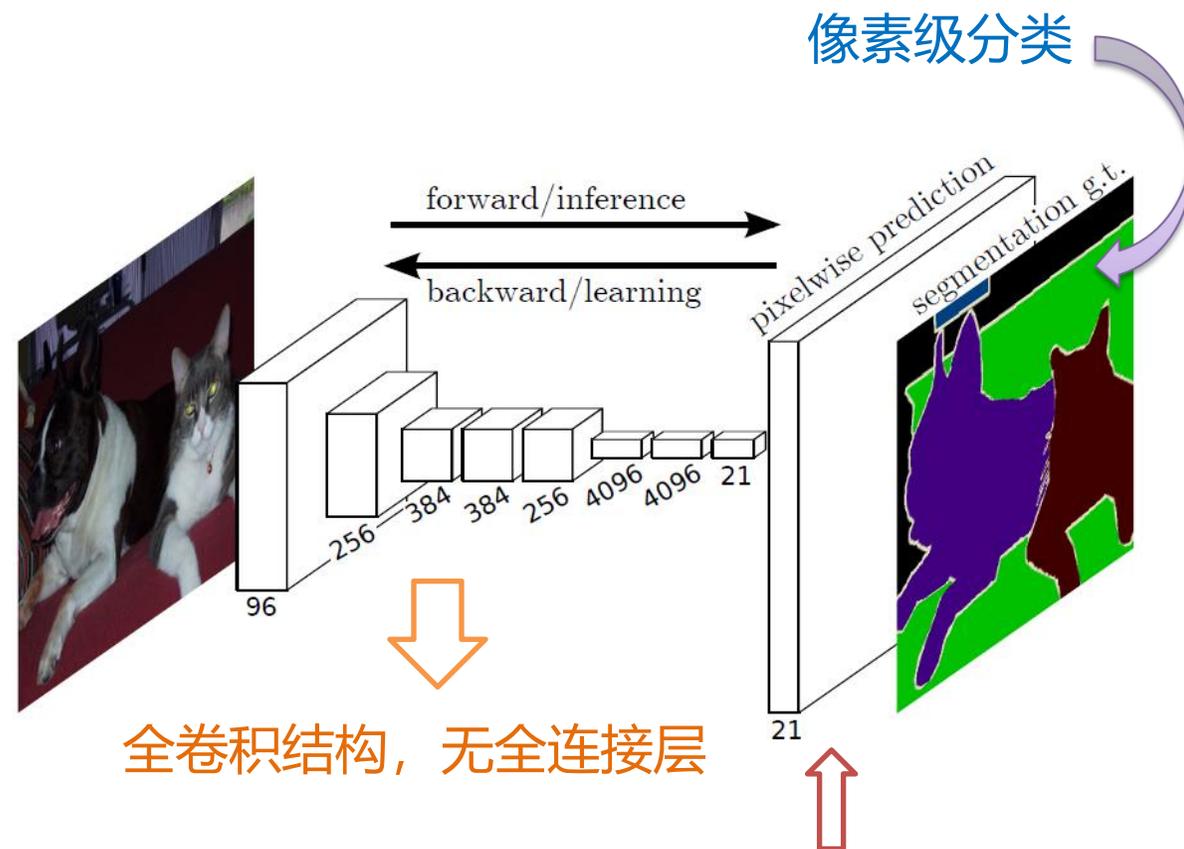
✓ 全卷积, 没有全连接层

● 语义分割是什么?

✓ 语义分割 \approx 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



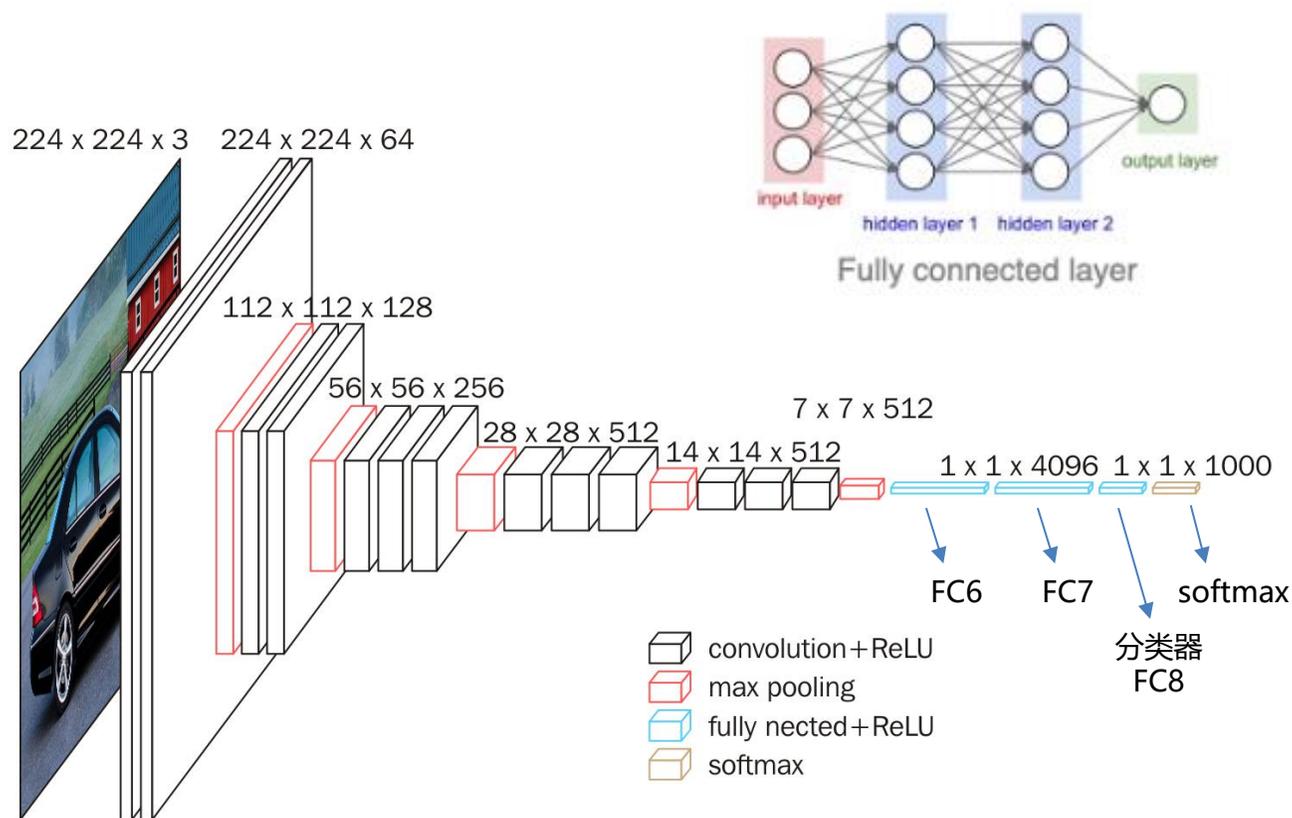
FC层替换为Conv层

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

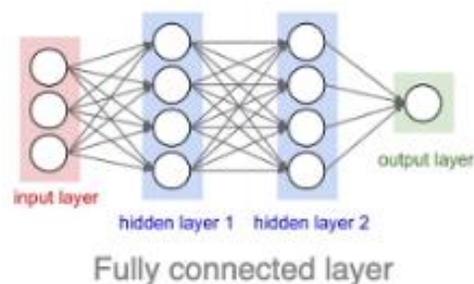
3.1.1 全卷积网络是怎么工作的?

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

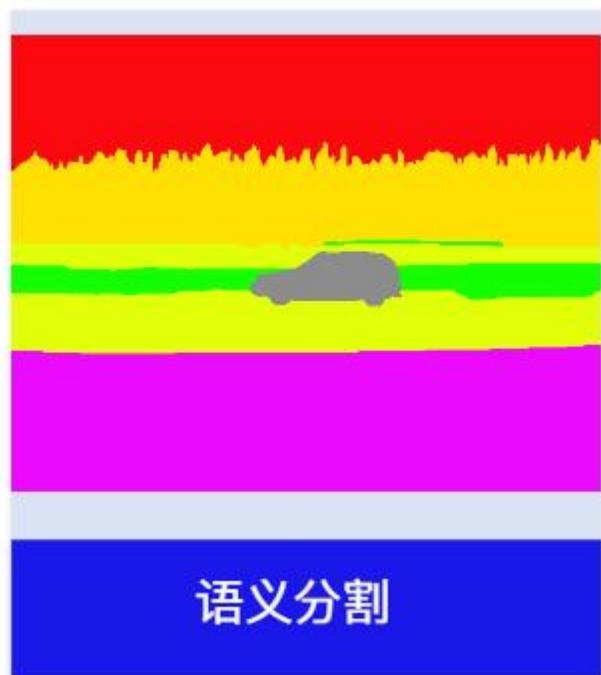
- 输入： $224 \times 224 \times 3$
- 特征：堆叠conv和pooling
- 输出： $1 \times 1 \times 1000$ (类别)

✓ 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



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。



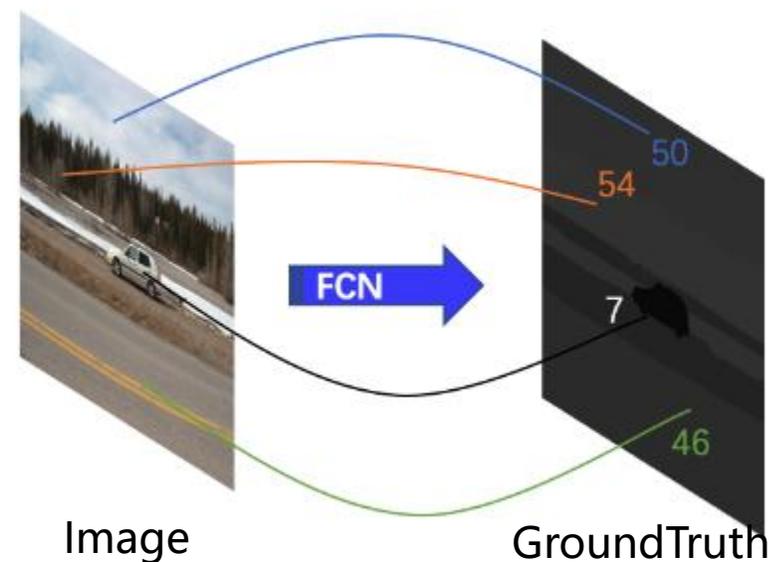
对整幅图像进行分类，将其划分为一个类别，或输出一系列可能类别的概率。

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



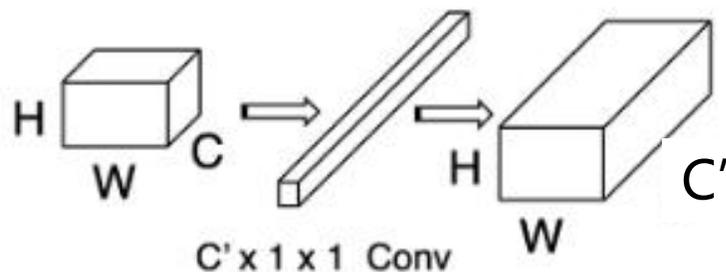
最终目标

3.1.1 全卷积网络是怎么工作的?

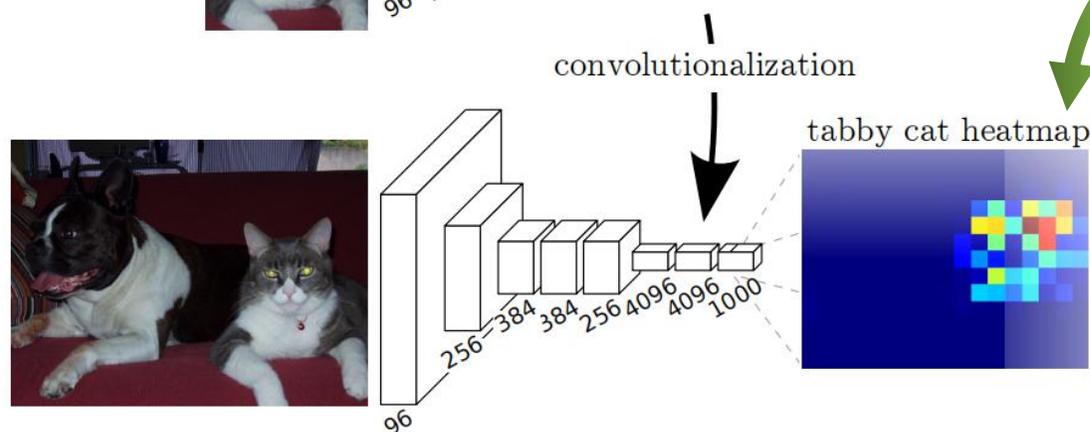
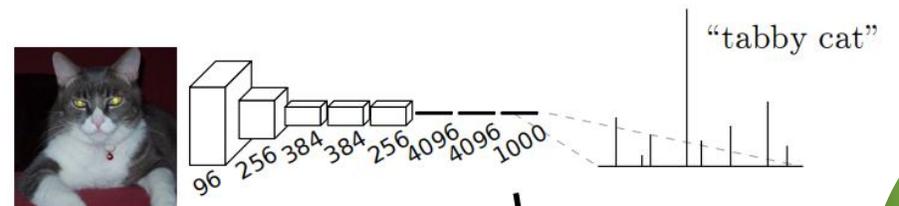
如何将全连接层替换为卷积层

FC层 \rightarrow 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**。

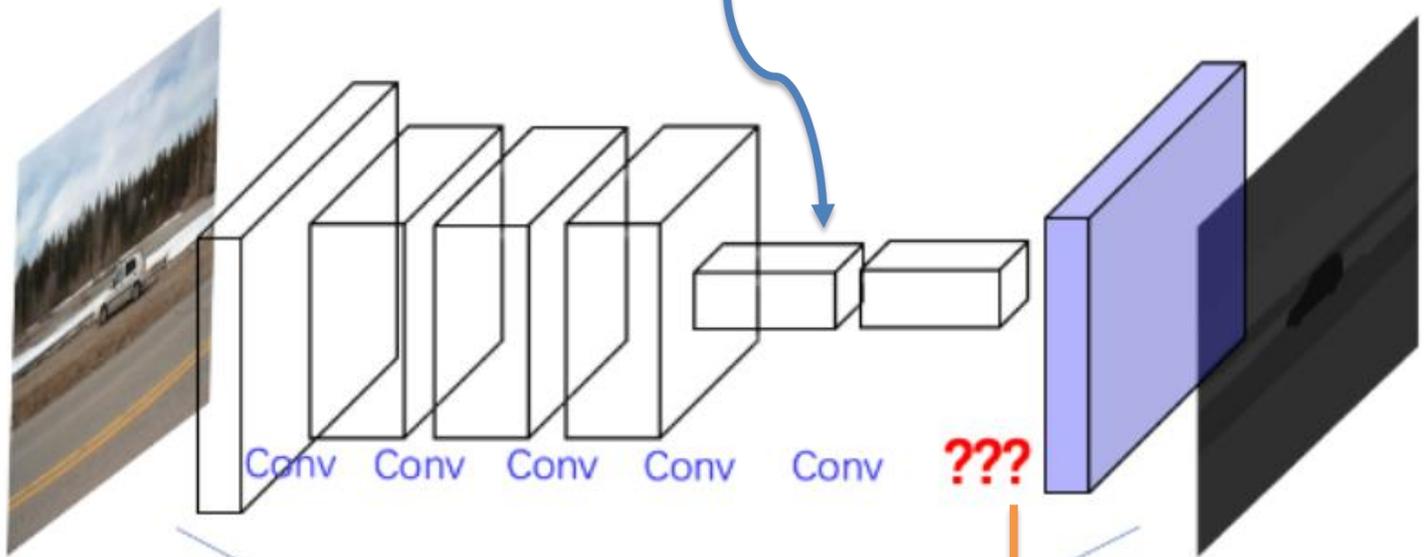


将全连接层转换为卷积层，可以让分类网络能够输出一个热图力(heatmap)。在VOC中，每个像素包含21个类别。

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-sampling
- 转置卷积Transpose Conv
- 上池化UnPooling

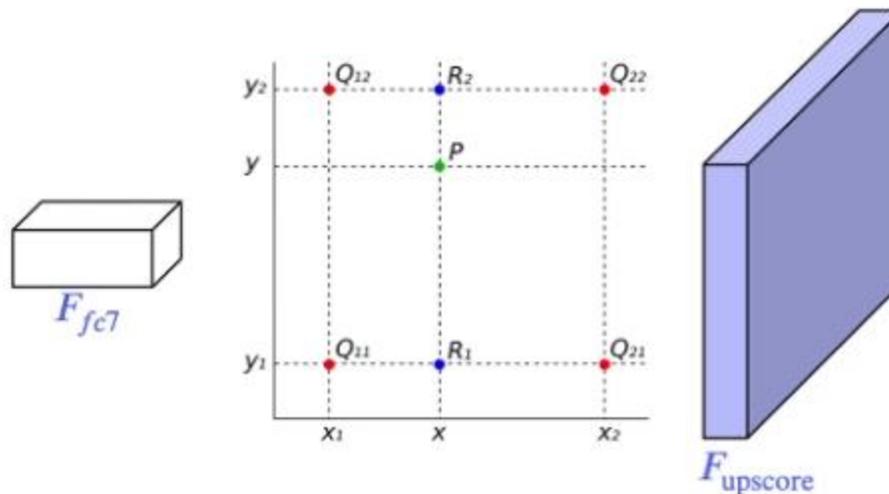
3.1.1 全卷积网络是怎么工作的?

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“图像” F_{fc7} 到 “图像” $F_{upscore}$ 的resize
- Resize的实现方法: 双线性插值 bilinear Interpolation

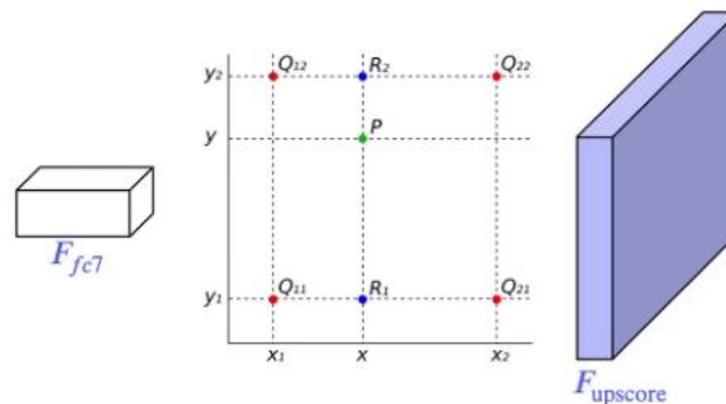


3.1.1 全卷积网络是怎么工作的?

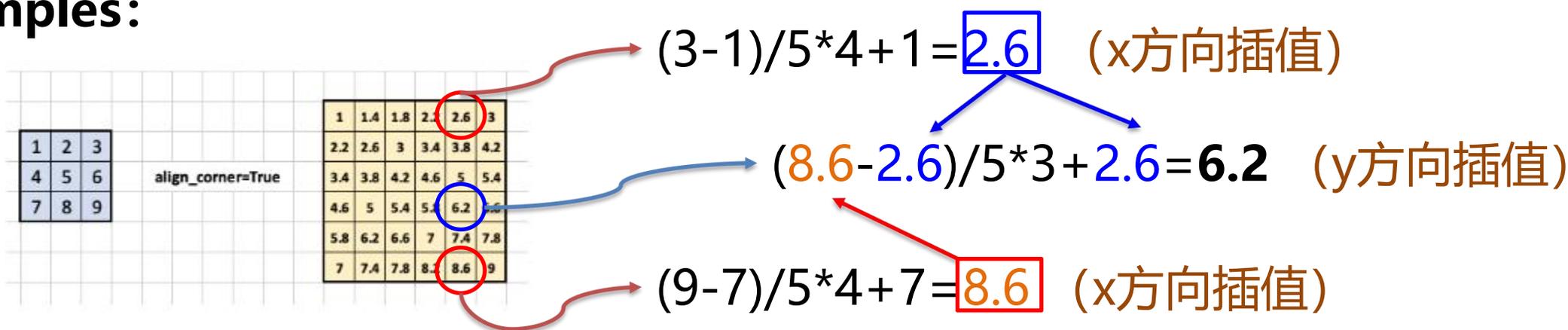
如何使特征图变大?

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:



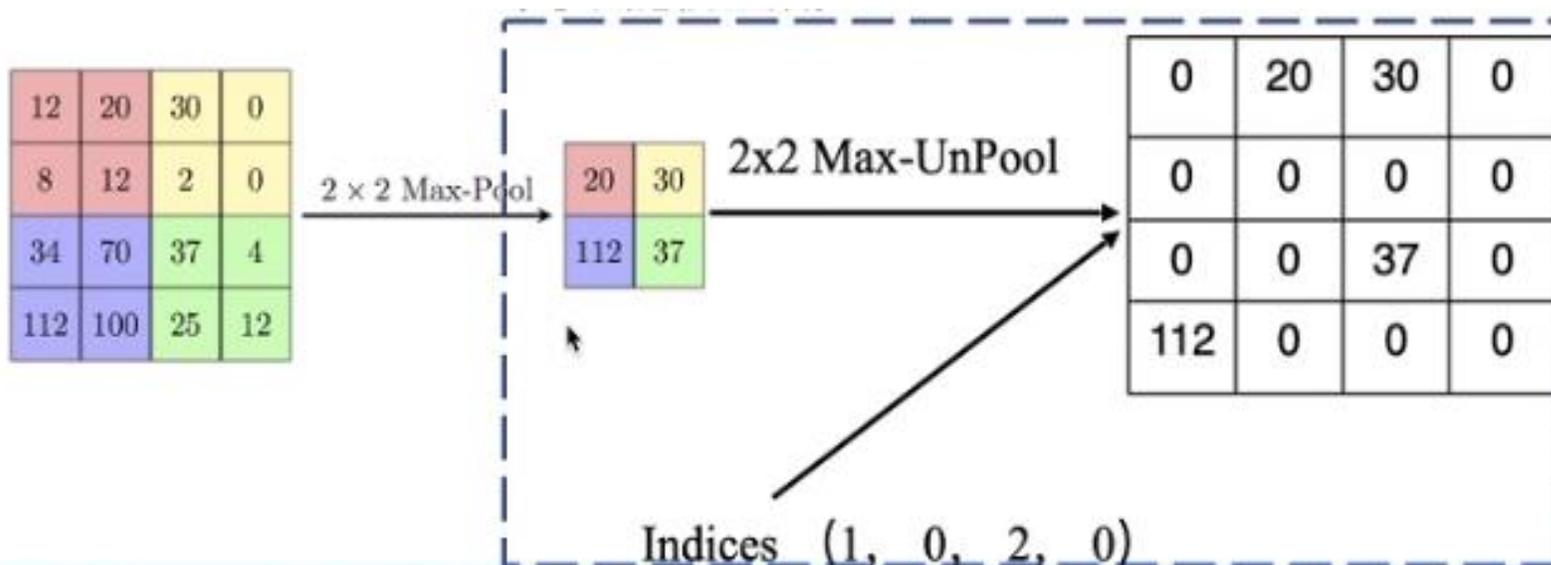
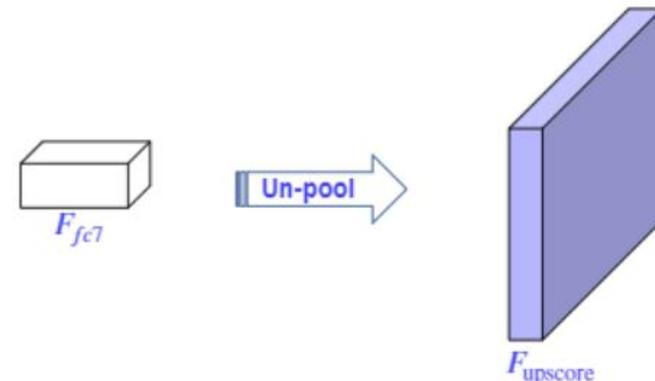
代码实现: Demo_Interpolation

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

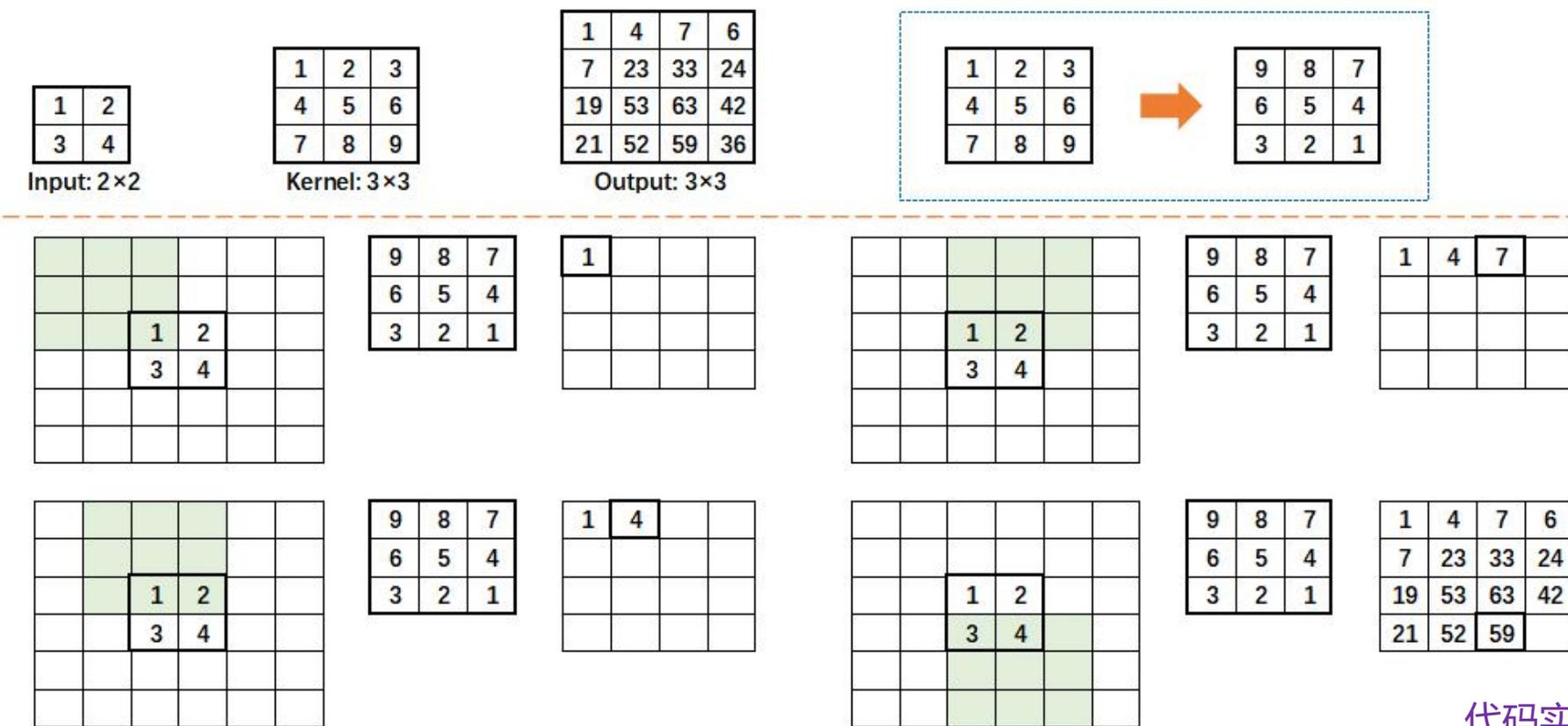


3.1.1 全卷积网络是怎么工作的?

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



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

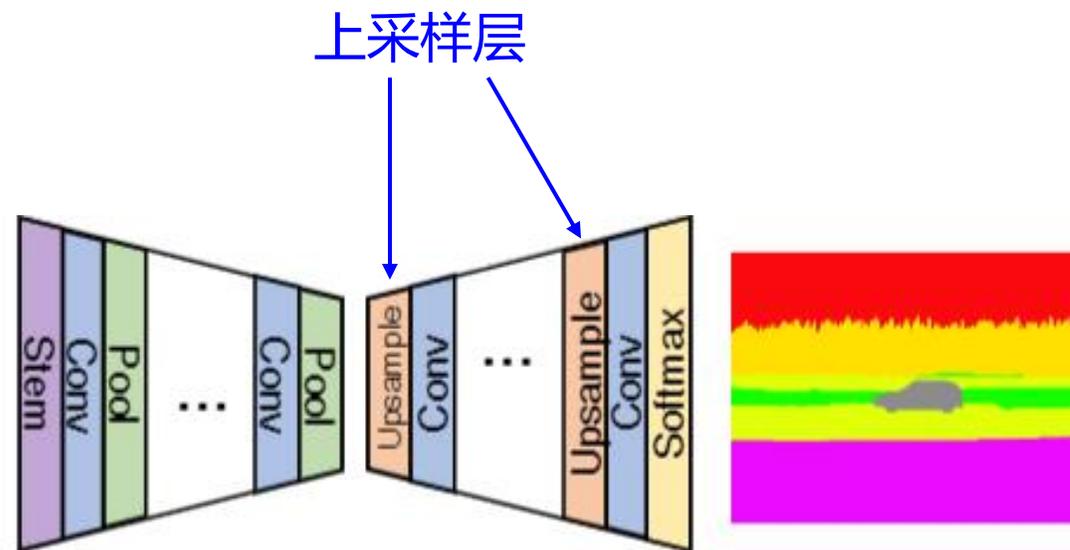
Encoder —— Decoder

- FCN的基本模块

- ✓ 卷积层 Convolution
- ✓ 降采样层 Down-sampling
- ✓ 上采样层 Up-sampling



输出图像



编码器

解码器

输出分割图

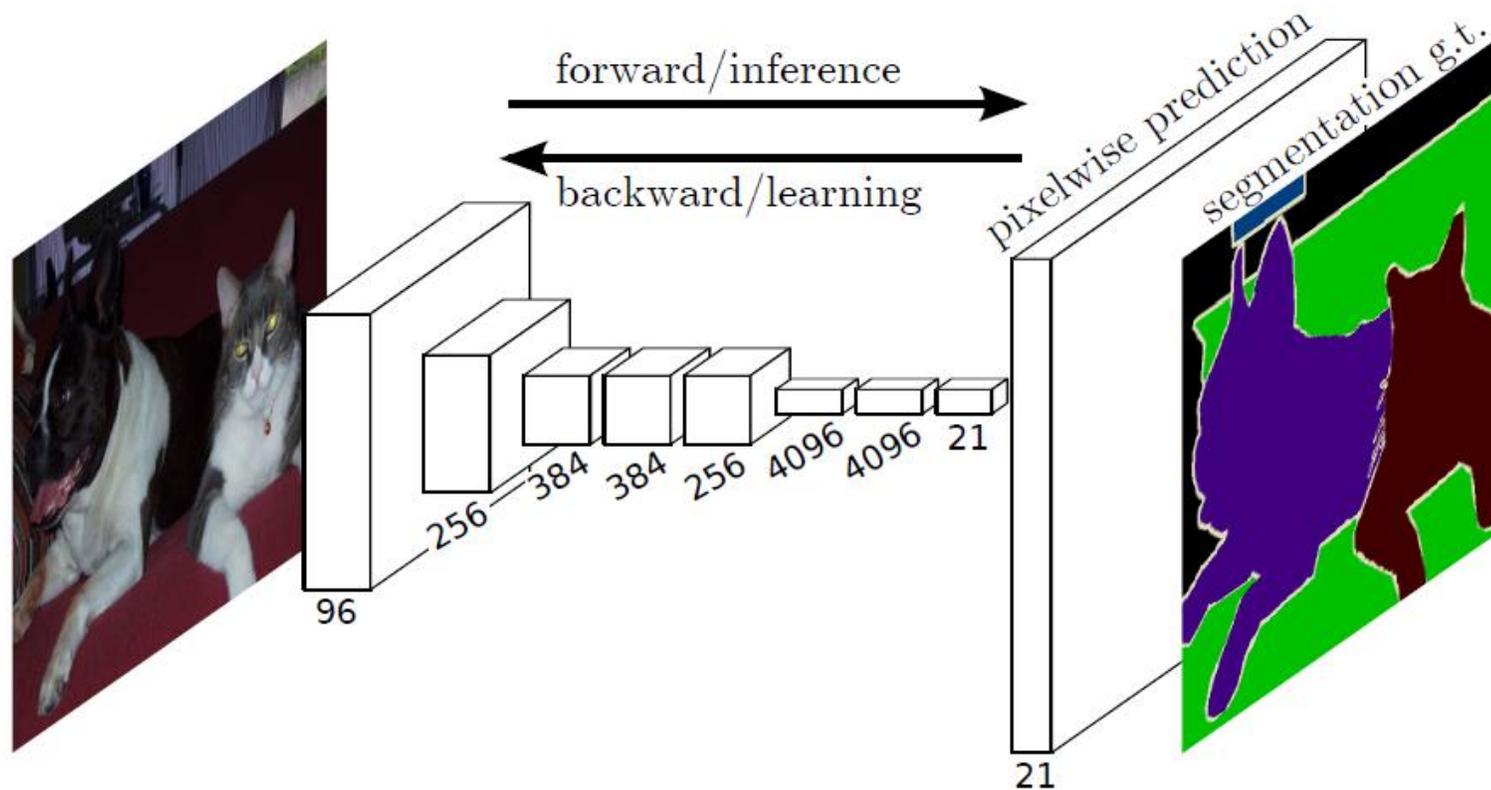


3.1.2 第一个全卷积网络——FCN全卷积网络

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。

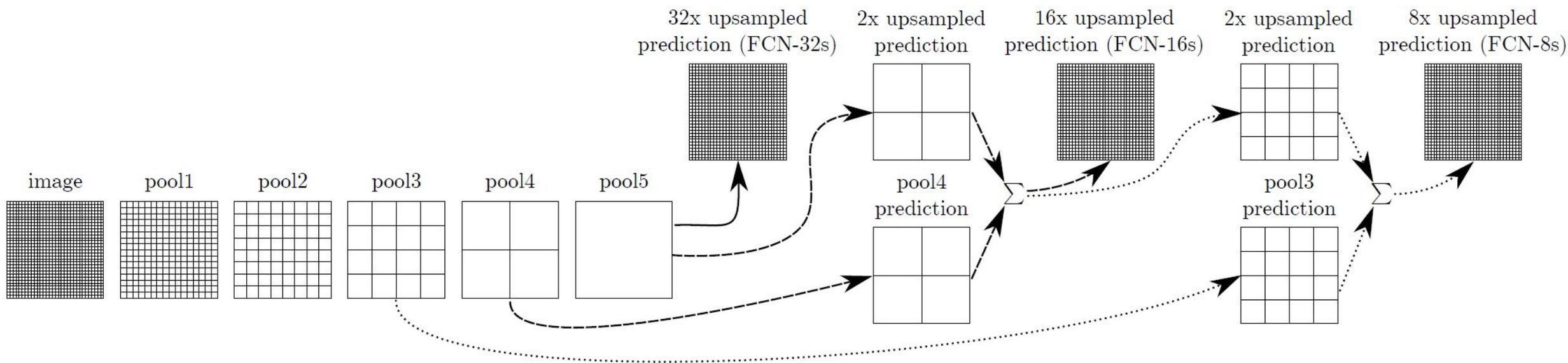
FCN在CNN(VGG-16)的基础上将全连接层替换为卷积层, 输出空间映射而不是分类分数。这些映射由小步幅卷积上采样 (反卷积、转置卷积) 获得, 用于产生密集的像素级预测。



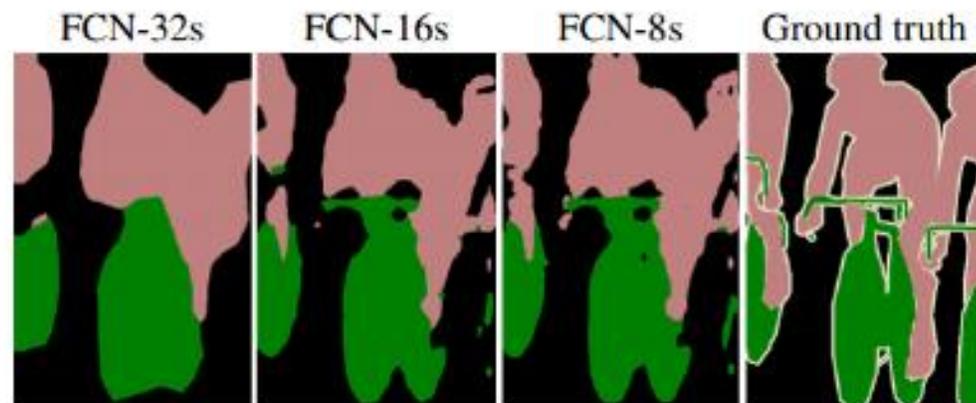
Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR2015

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



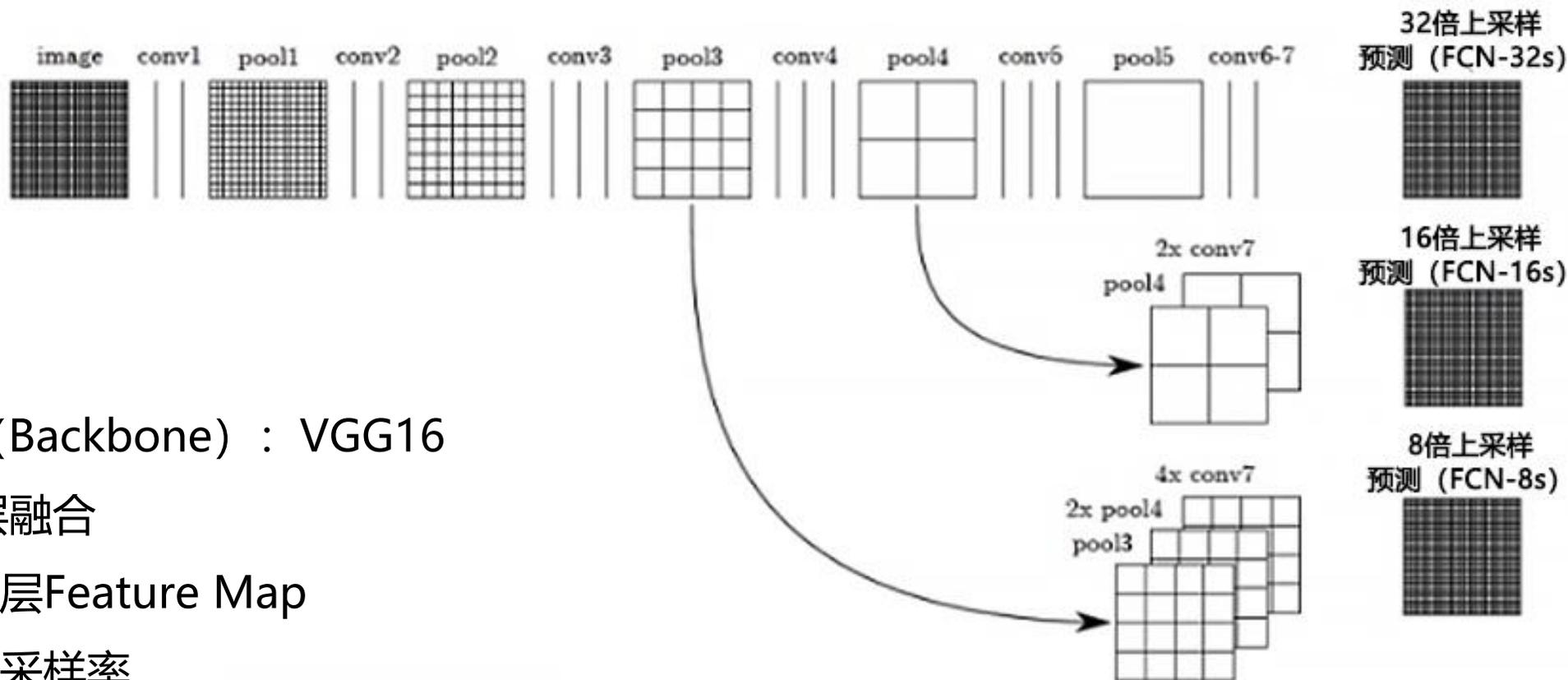
池化层和带跨步的卷积会导致特征图的分辨率不断下降，FCN利用双线性插值将响应张量的长宽上采样到原图大小。多尺度让输出的预测更加精细，低层更关注细节，高层更关注整体



Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully convolutional networks for semantic

3.1.2 第一个全卷积网络——FCN全卷积网络

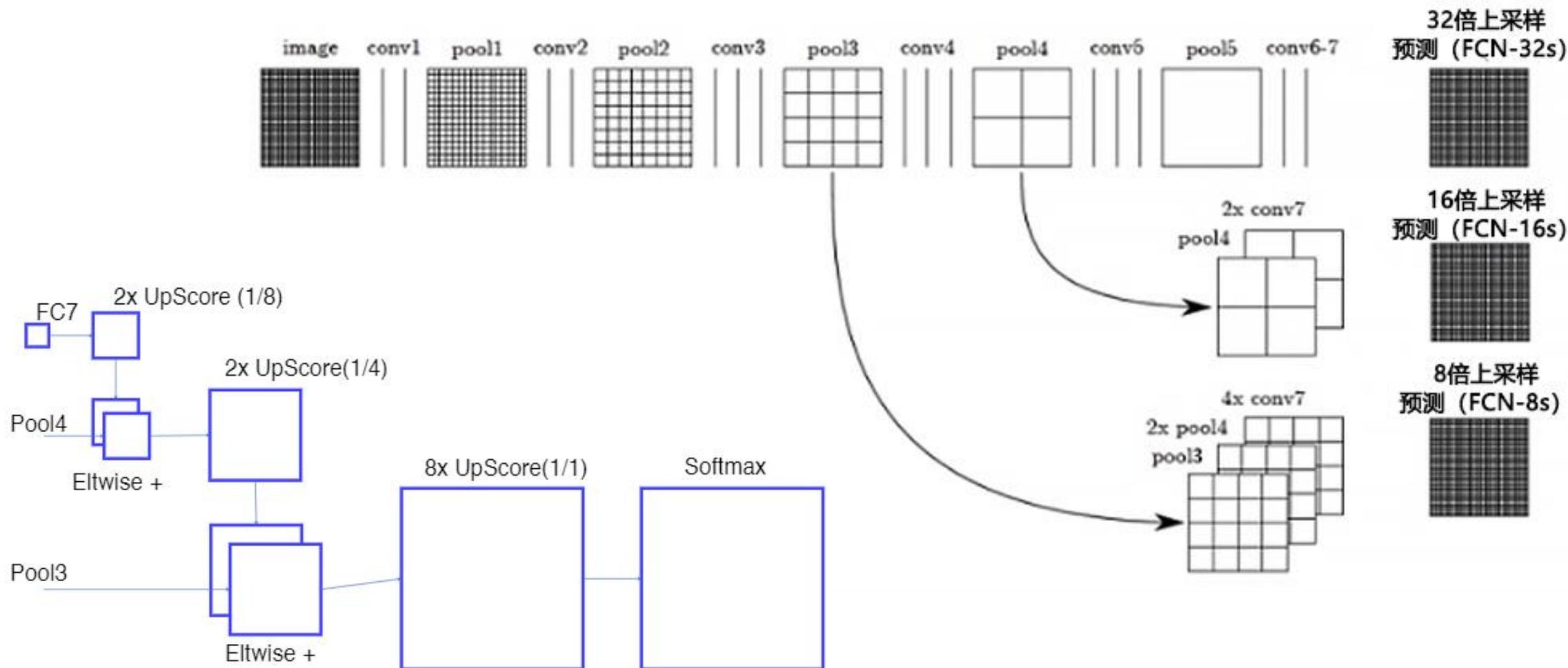
FCN全卷积网络的基本结构



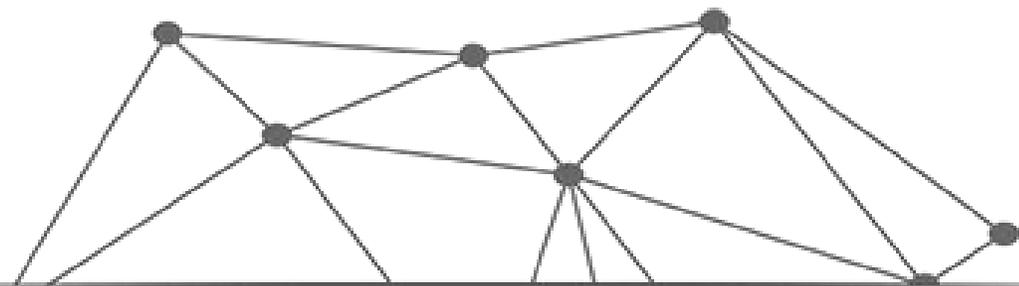
- 骨干网络 (Backbone) : VGG16
- FCN的多层融合
 - ✓ 集成多层Feature Map
 - ✓ 不同的采样率
 - ✓ 组合方式: Element-wise Add

3.1.2 第一个全卷积网络——FCN全卷积网络

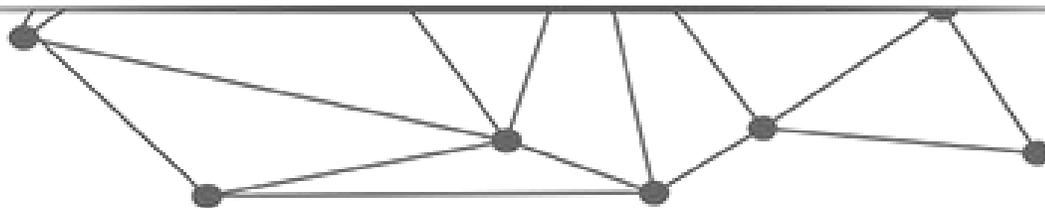
FCN全卷积网络的基本结构

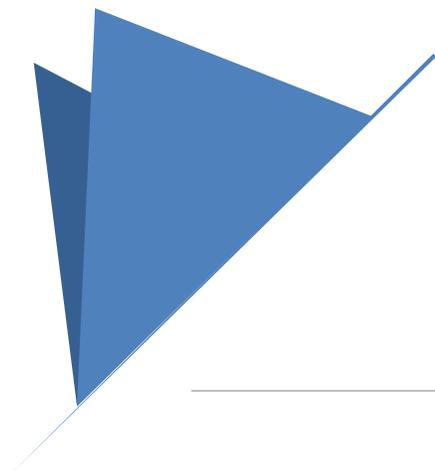


Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR2015



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

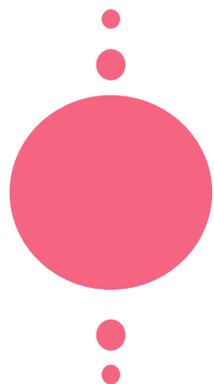
What is PSPNet? And how to build PSPNet?

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

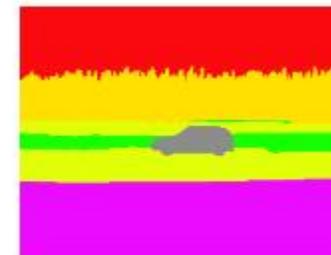
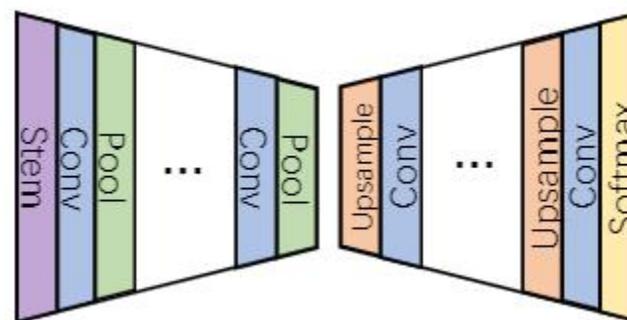
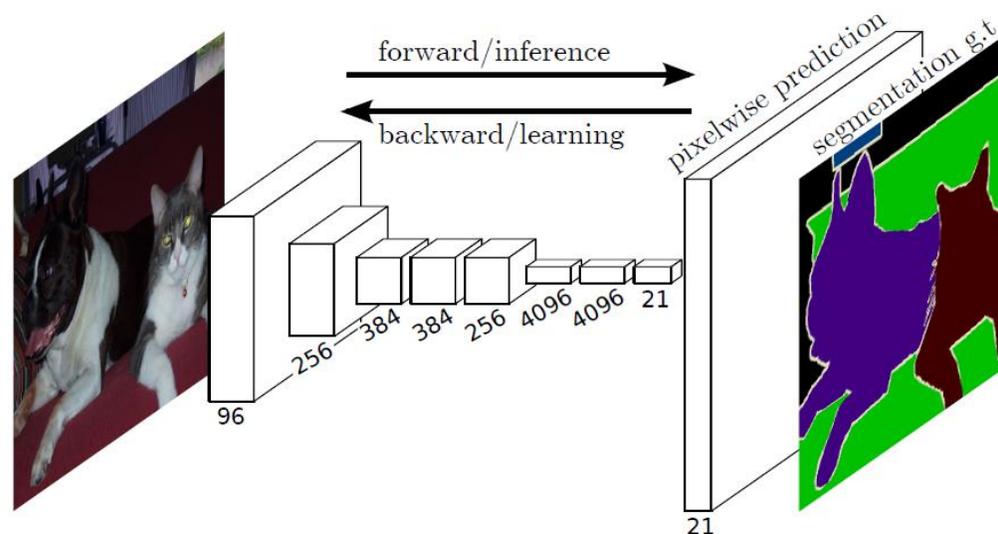
● 全卷积网络

- ✓ 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

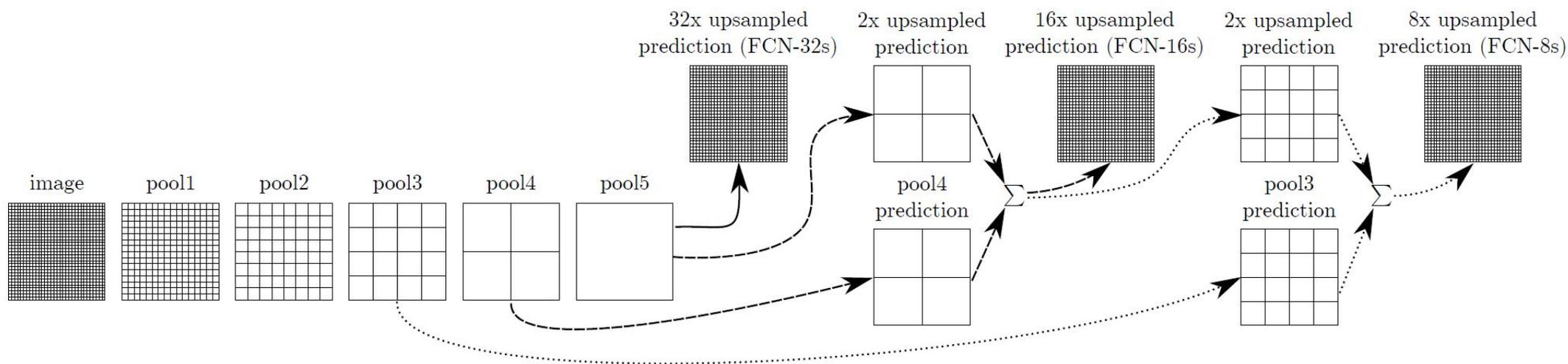
- ✓ 卷积层
- ✓ 下采样层
- ✓ 上采样层



Encoder-Decoder结构

3.2.1 Recap: 全卷积网络FCN

Recap: FCN的优缺点

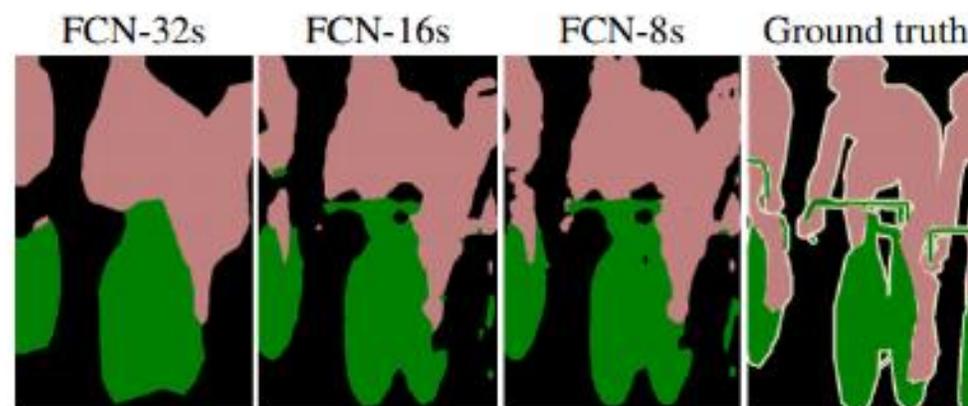


● 优点

- ✓ 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

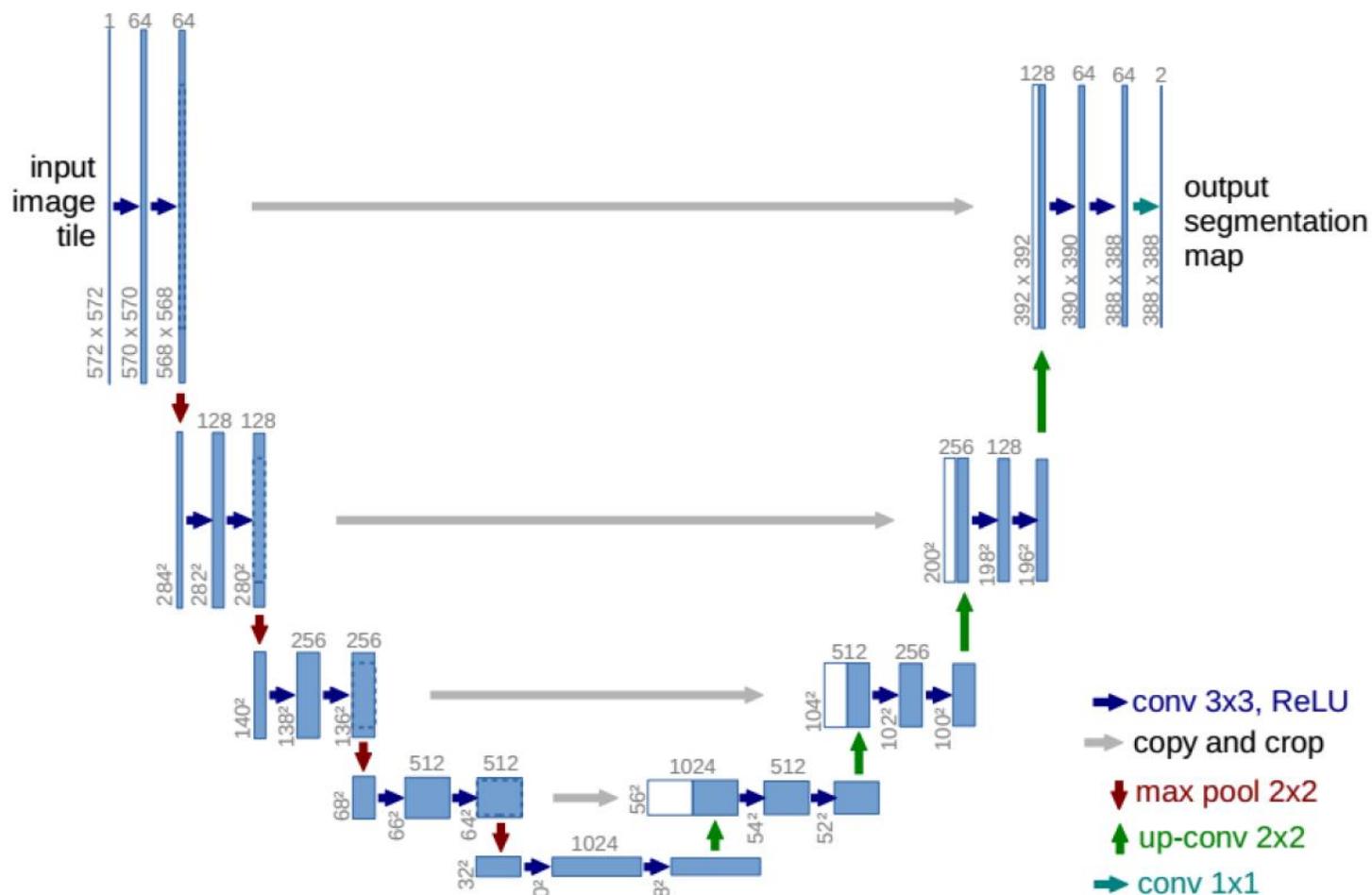
● 缺点

- ✓ 分割结果不够精细
- ✓ 没有考虑上下文信息 (\approx 左顾右盼 \approx 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



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

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。

3.2.2 医学影像分割的重要网络U-Net

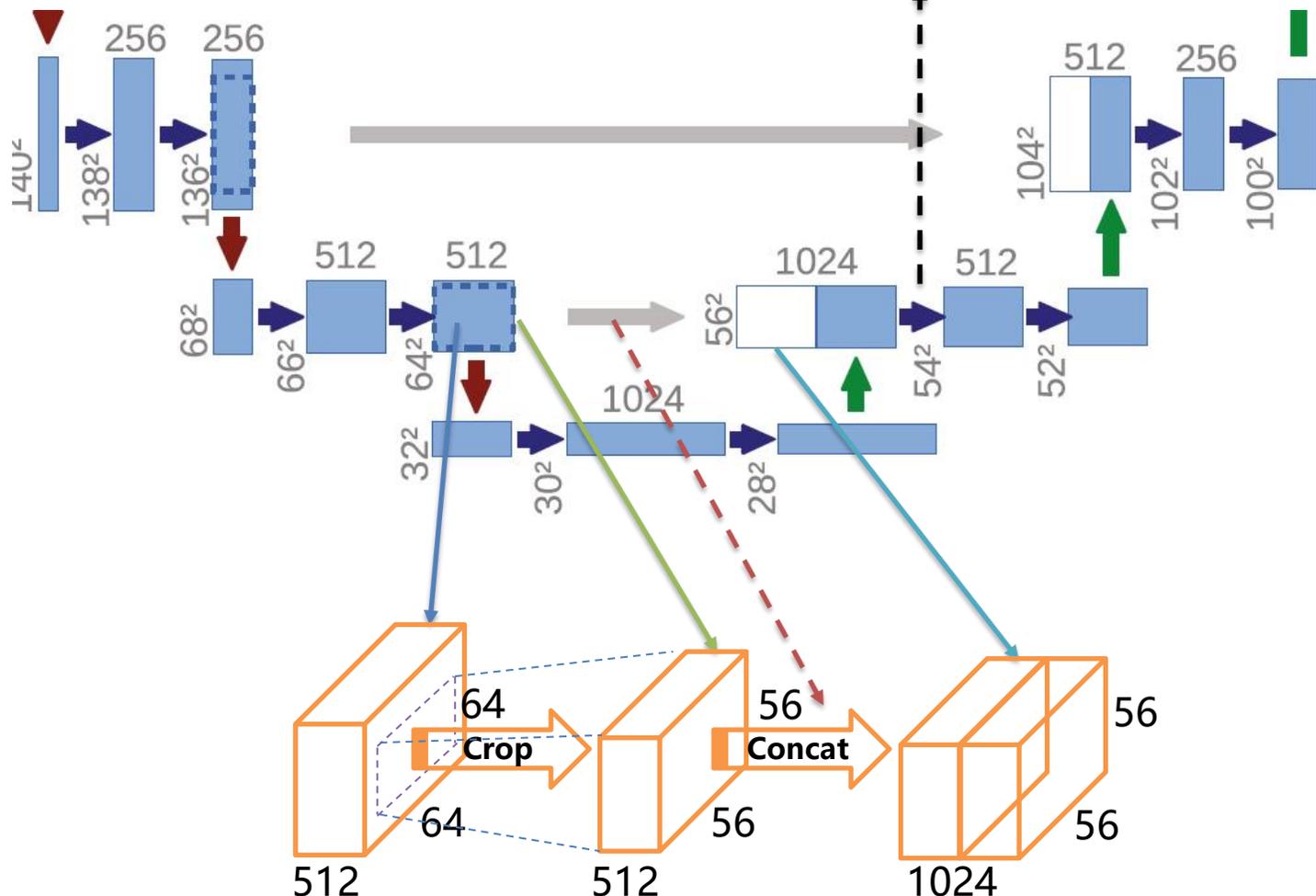
U-Net的框架结构

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?



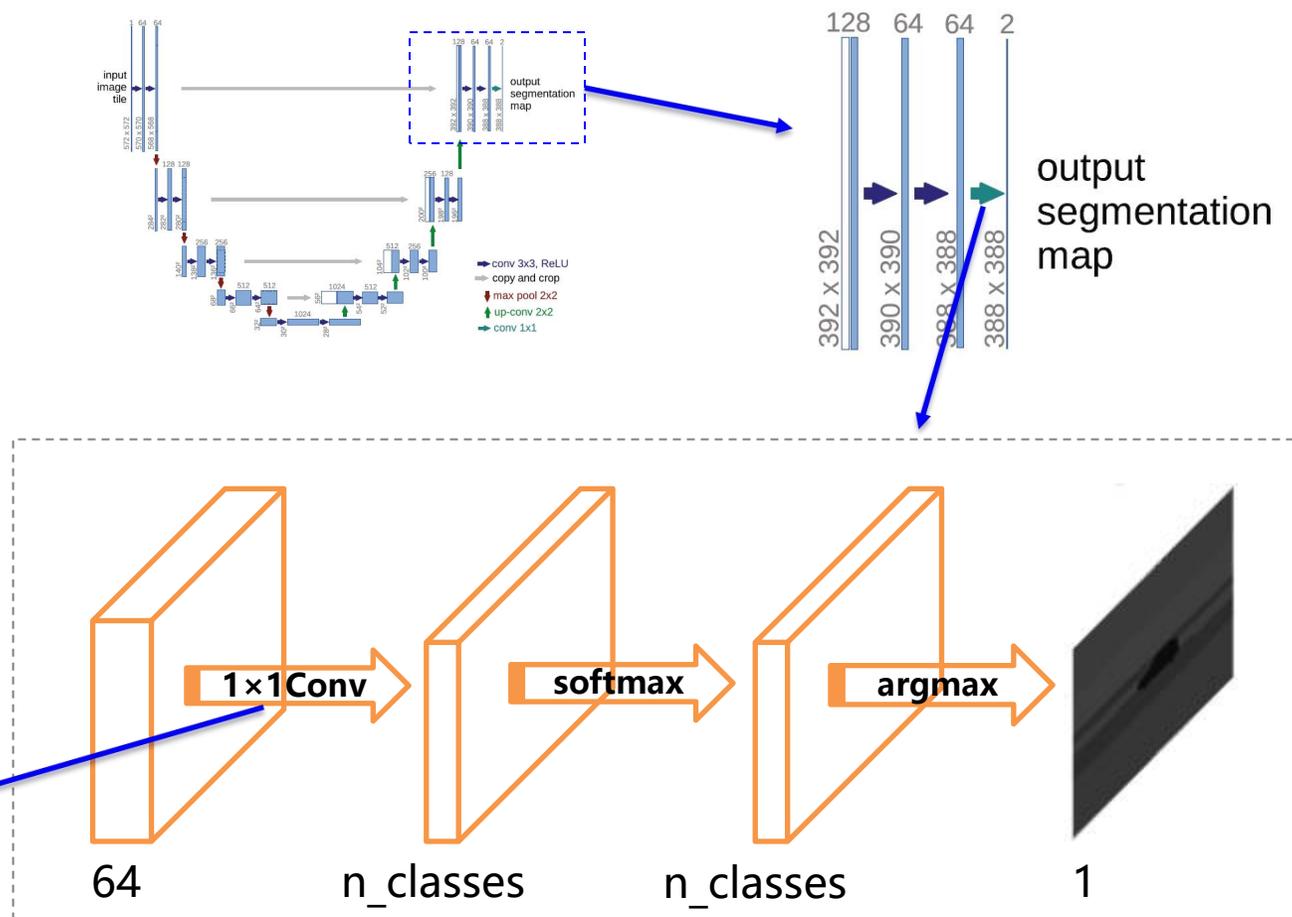
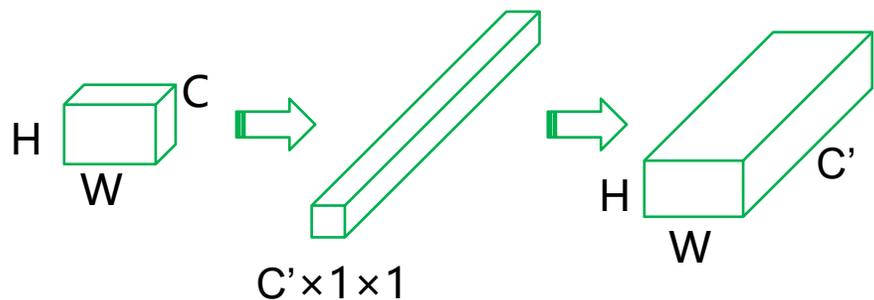
Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015

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



3.2.2 医学影像分割的重要网络U-Net

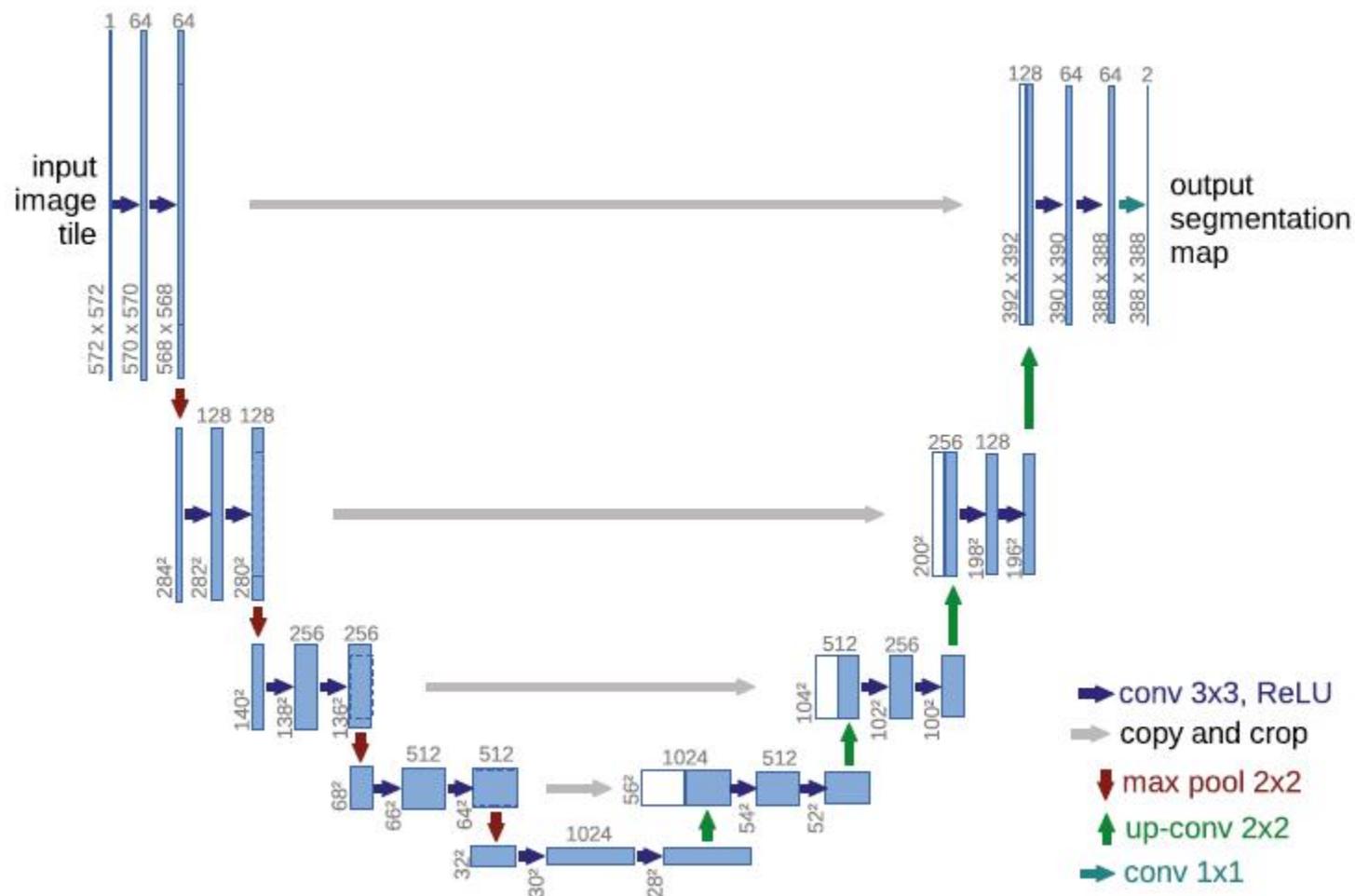
U-Net的框架结构

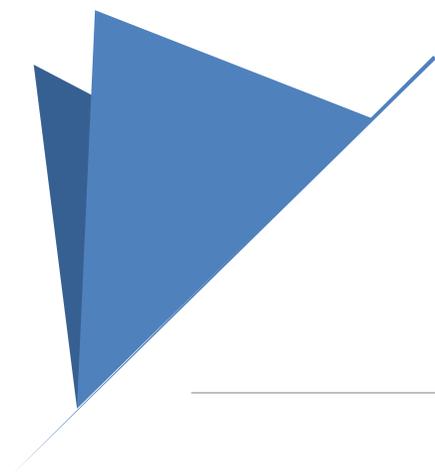
U-Net的主要操作:

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

增大2x

分割类别





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**

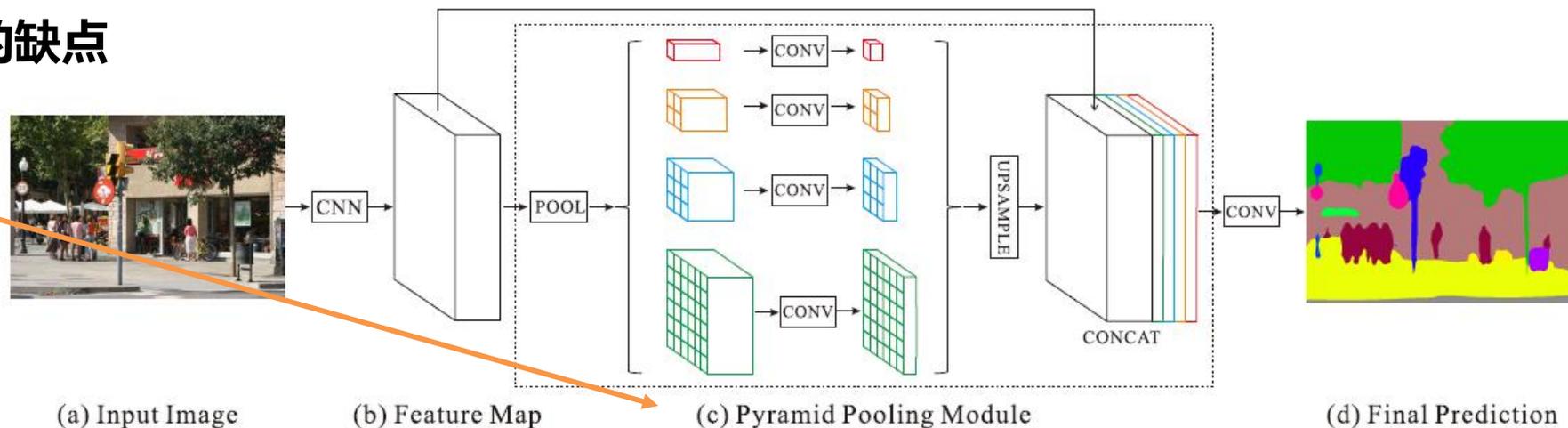
- ✓ *Pyramid*: 空间金字塔结构

- ✓ *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



3.3 PSP分割网络

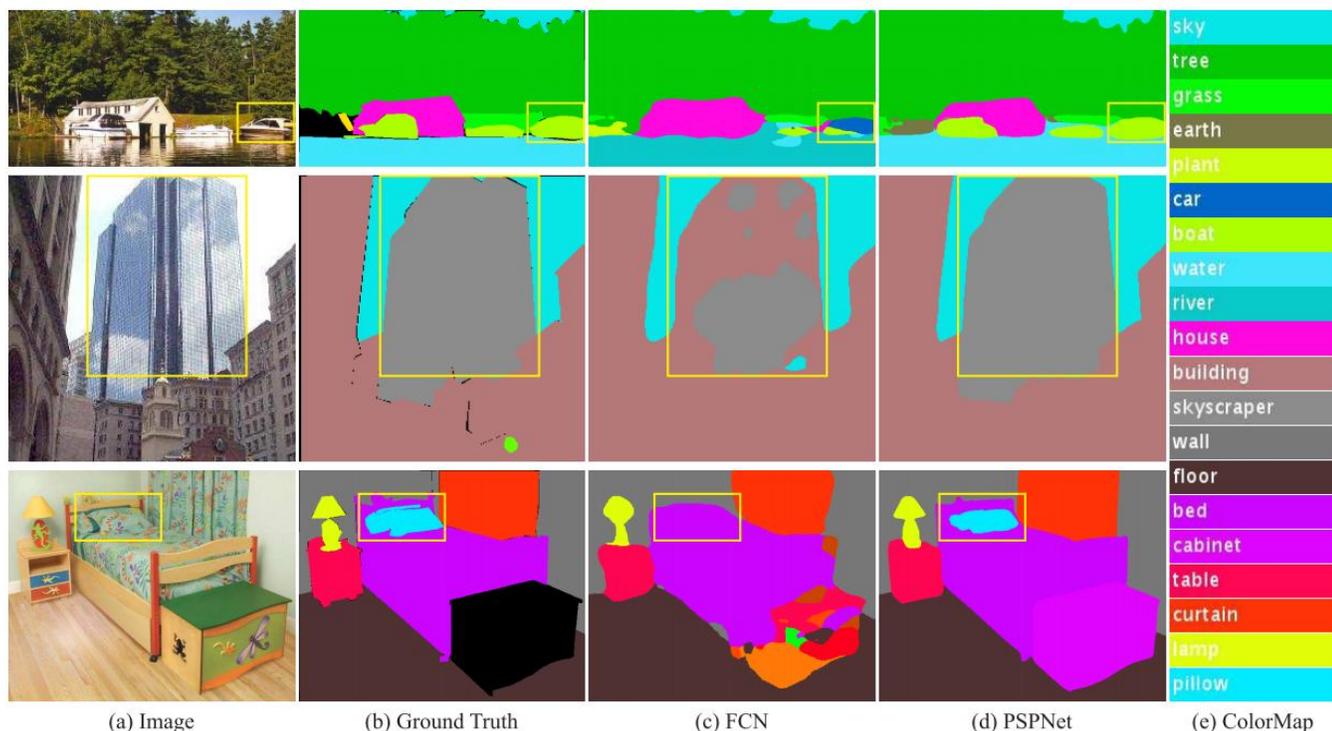
什么是上下文信息?

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**。

- **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。



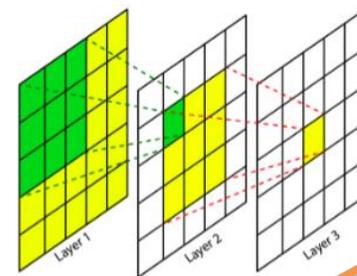
3.3 PSP分割网络

如何获取上下文信息

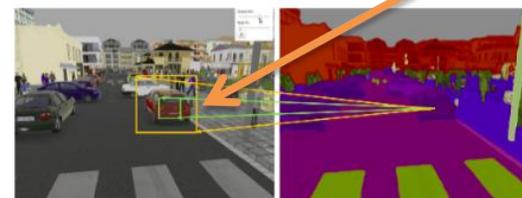
What does Receptive Field mean in CNN?

● 感受野 (Receptive Field)

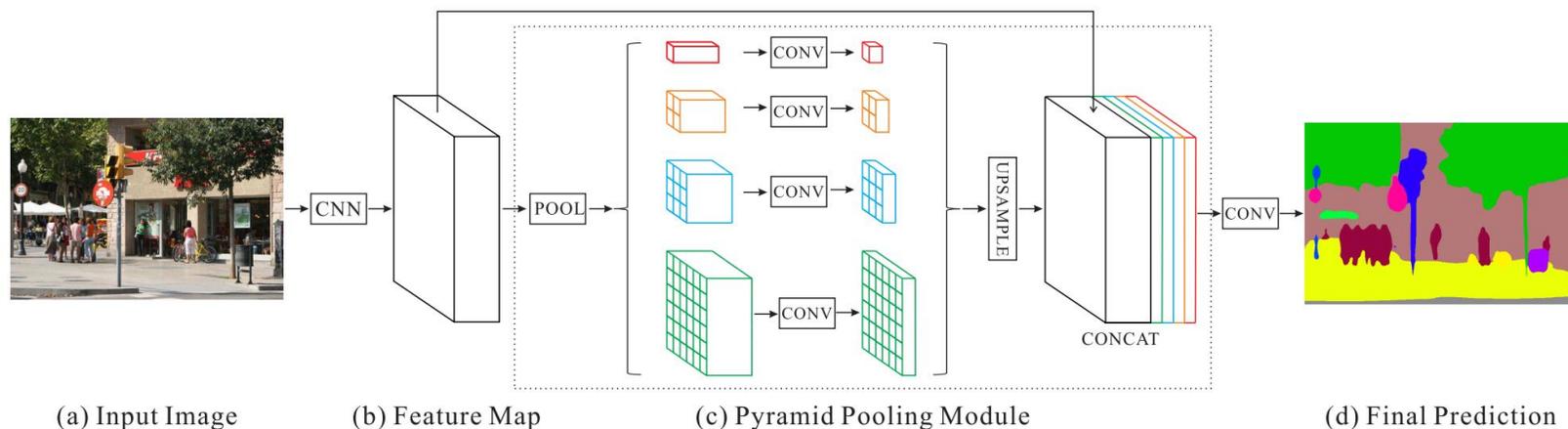
- ✓ 用于产生特征的输入图像中的区域大小
- ✓ 感受野只针对局部操作, 如Conv, Pooling



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

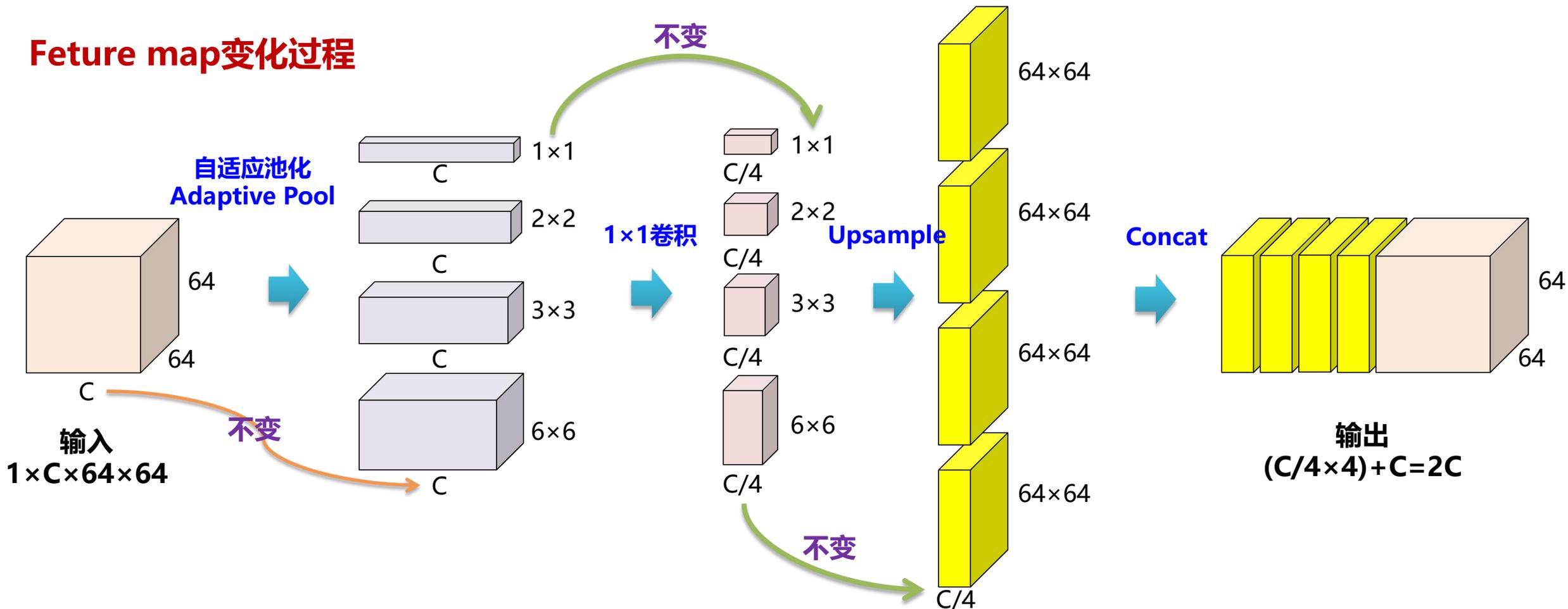


3.3 PSP分割网络

Pyramid Pooling模块

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

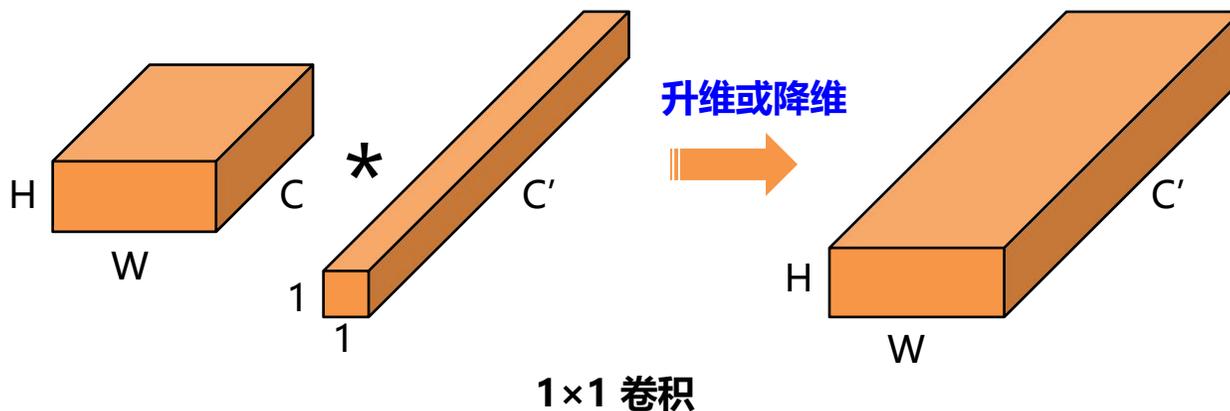
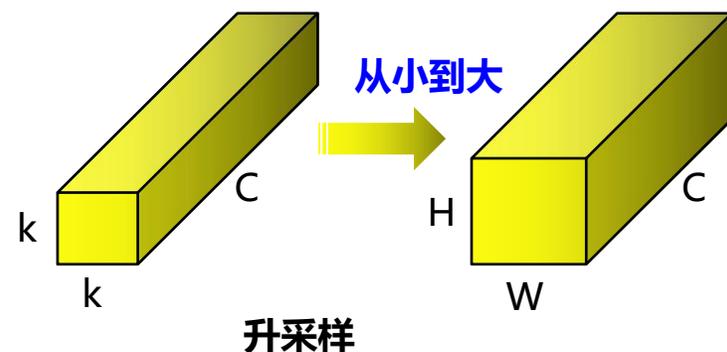
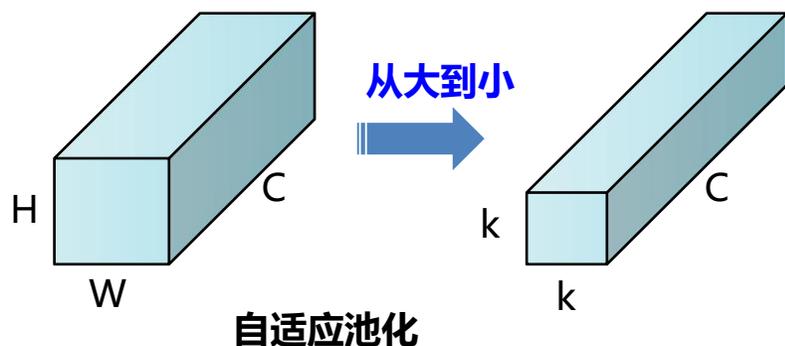
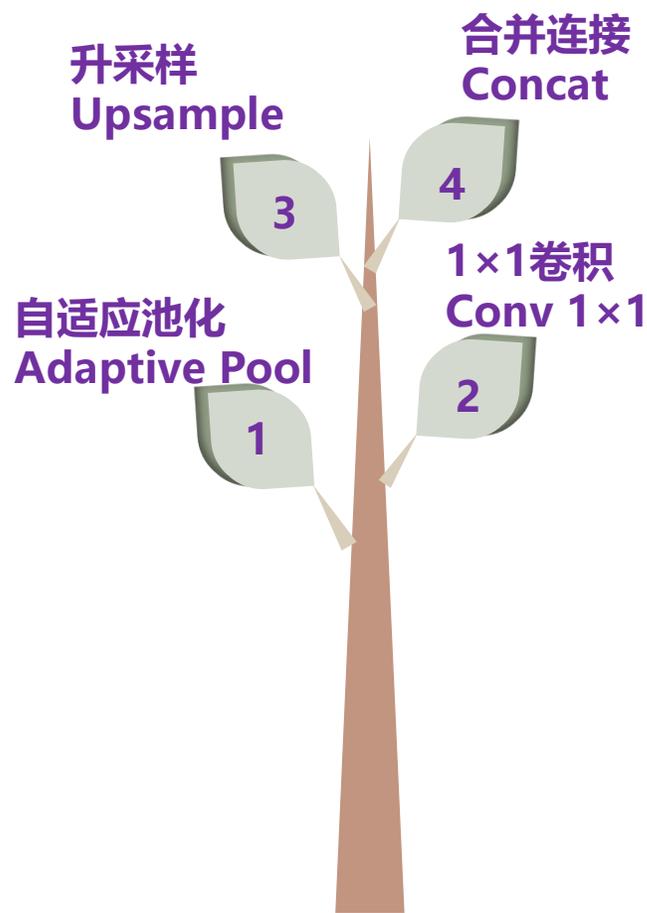
Feature map变化过程



3.3 PSP分割网络

Pyramid Pooling模块

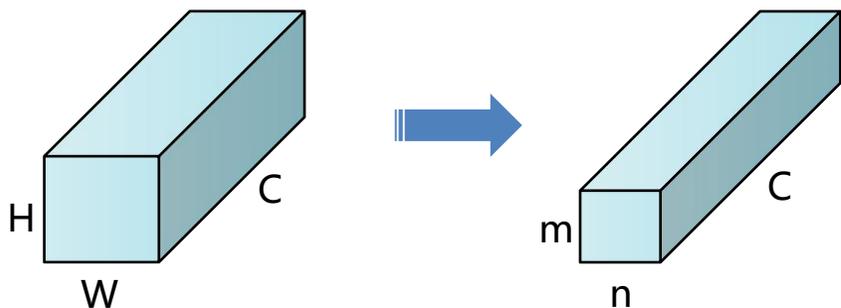
What is inside Pyramid Pooling Module? **重要操作**



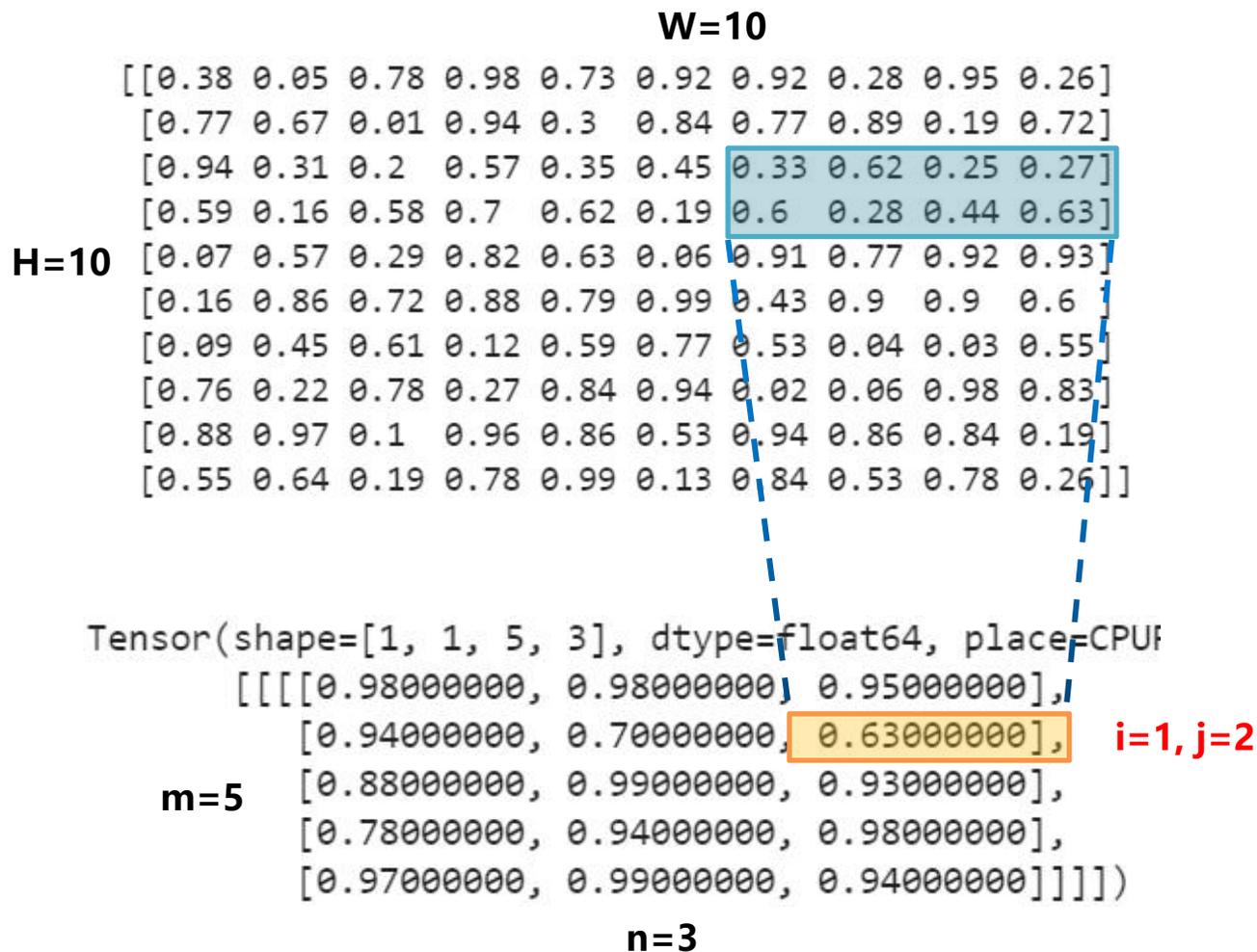
3.3 PSP分割网络

Pyramid Pooling模块

自适应池化 (Adaptive Pool)



$hstart = \text{floor}(i * H / m) = 10/5=2$
 $hend = \text{ceil}((i + 1) * H / m) = 2*10/5=4$
 $wstart = \text{floor}(j * W / n) = 2*10/3=6$
 $wend = \text{ceil}((j + 1) * W / n) = 3*10/3=10$
 $output[:, :, i, j] = \max(input[:, :, hstart: hend, wstart: wend]) = \max(2:4, 6:10) = 0.63$



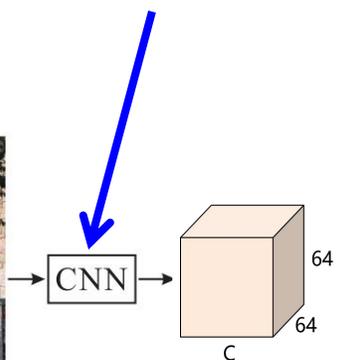
3.3 PSP分割网络

PSP网络体系结构图

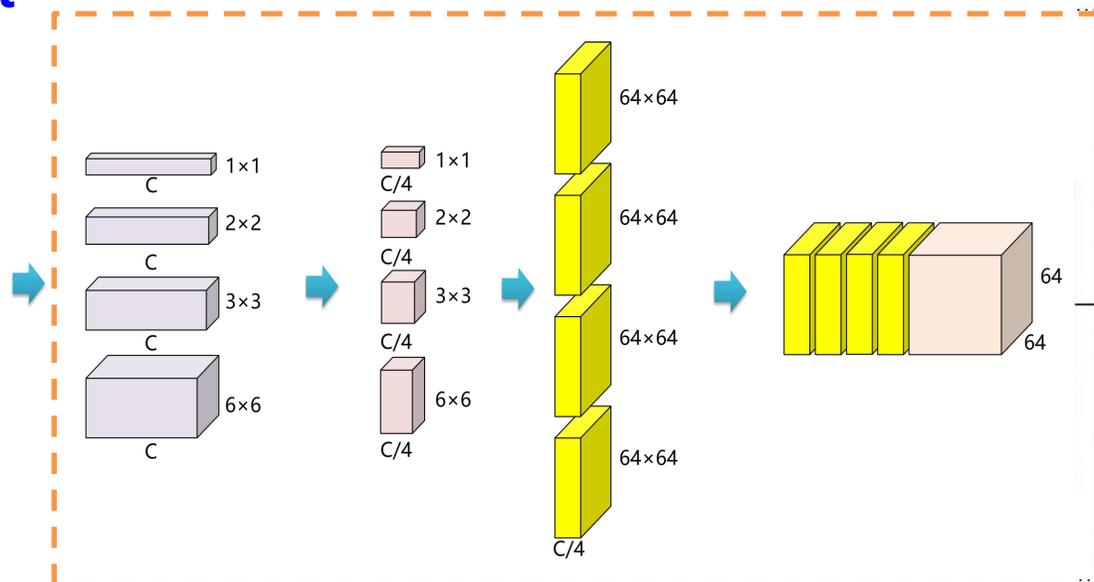
主干网: Dilated ResNet



(a) 输入图像



(b) 特征图



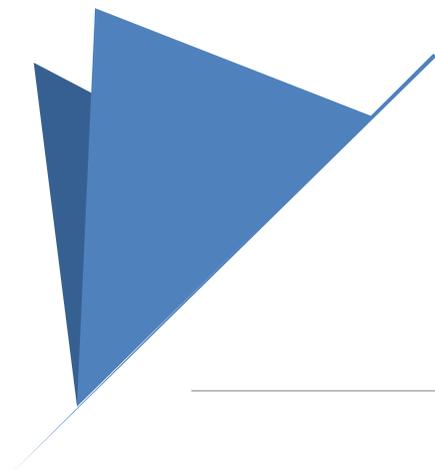
(c) 金字塔池化模块

1×1卷积, 还原输入尺寸,
并生成热力图

CONV



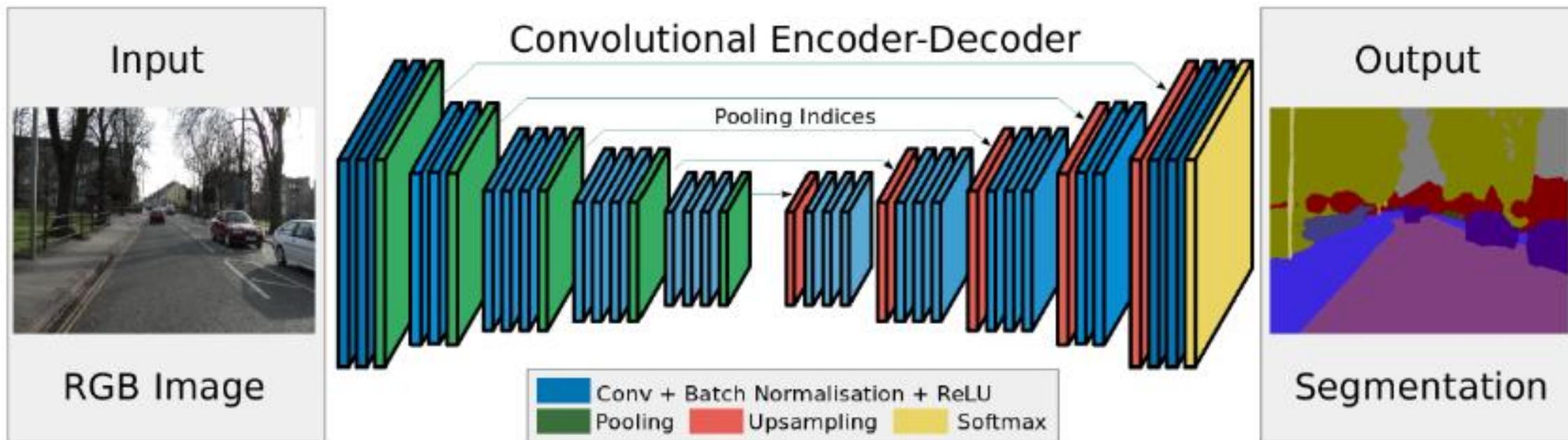
(d) 最终预测



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。

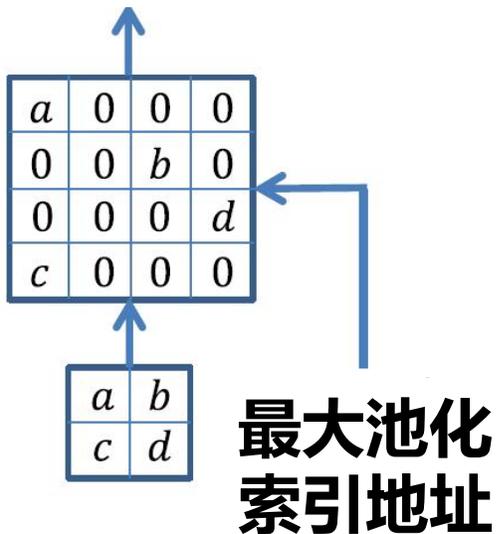
Vijay Badrinarayanan, Alex Kendall, Roberto Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. IEEE Transactions on PAMI, 2017

3.4 实时分割网络SegNet

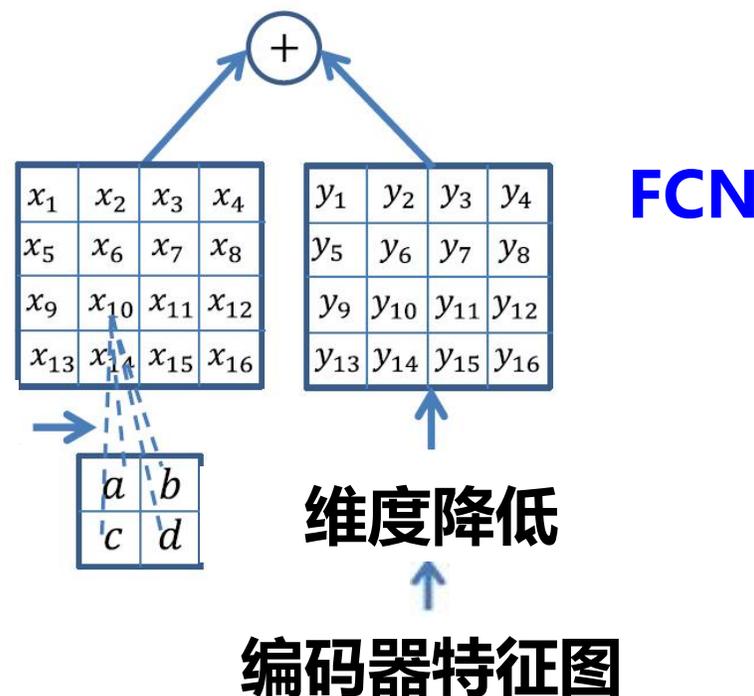
SegNet的核心算法

使用可训练的解码器执行转置卷积

SegNet

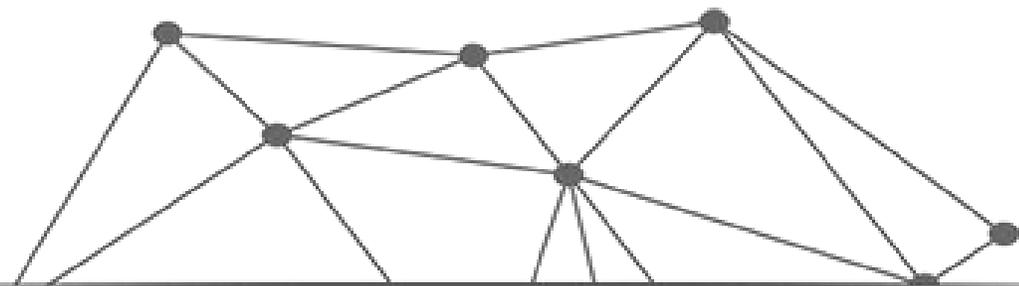


转置卷积
升采样

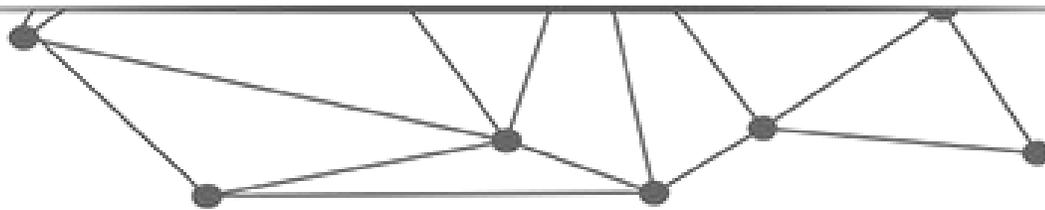


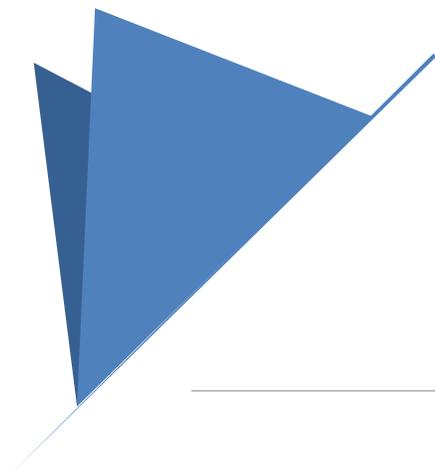
- 池化运算时，SegNet使用对应编码器记录的**位置信息**进行**上采样**，其他位置置为**零**。这样的方法**加快了网络的运行速度**，满足了**实时分割**的需求。
- FCN使用反卷积滤波器进行上采样，然后与对应编码器的特征映射相加。

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

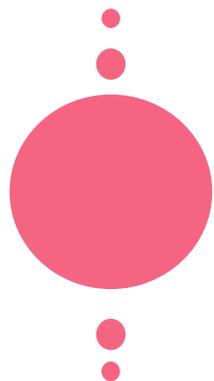
简要介绍 DeepLab涉及的一些核心技术，包括：

- 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

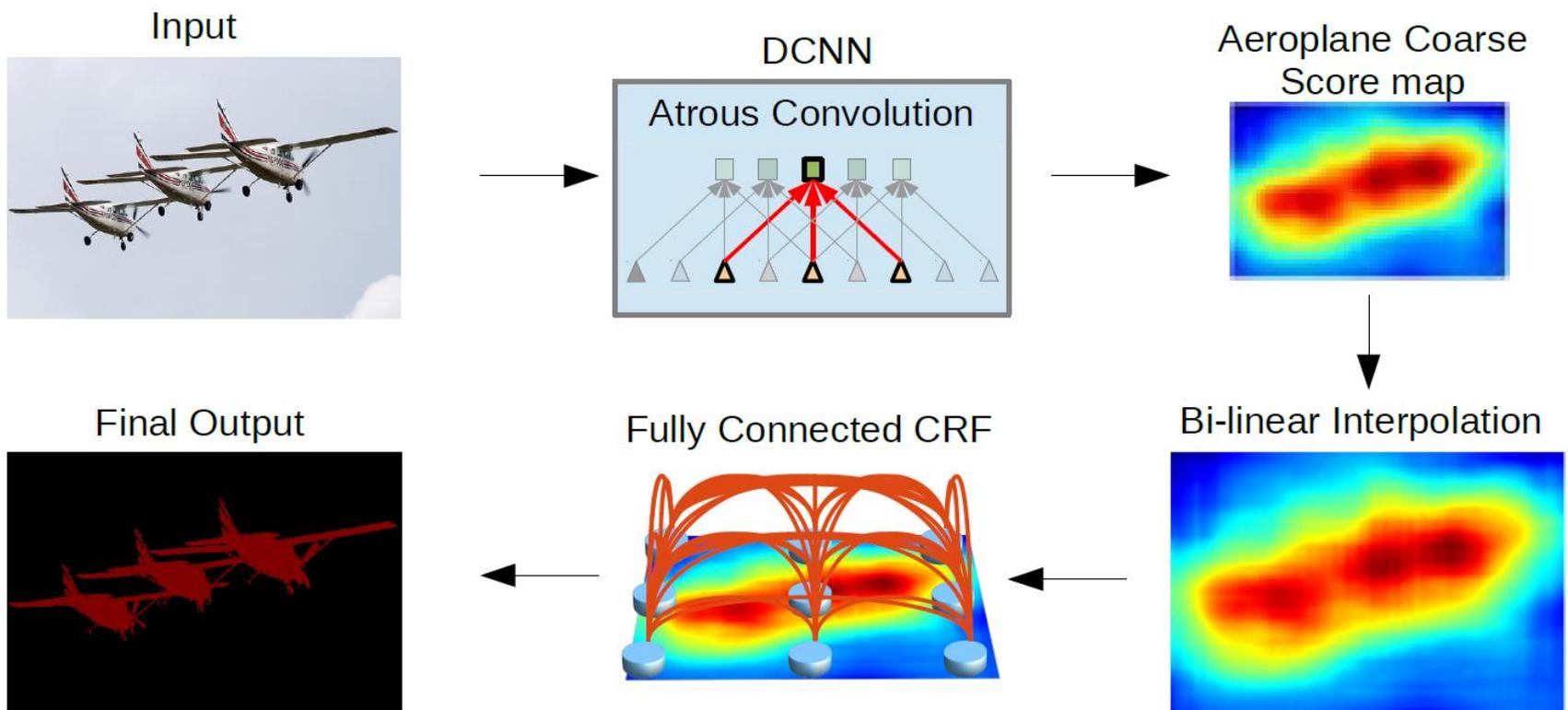
What is DeepLab?

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)

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

图像

4		

卷积特征

1 _{x1}	1	1 _{x0}	0	0 _{x1}	0	0
0	1	1	1	0	1	0
0 _{x0}	0	1 _{x1}	1	1 _{x0}	1	0
0	0	1	1	0	0	1
0 _{x1}	1	1 _{x0}	0	0 _{x1}	0	0
1	1	0	0	0	1	0
0	0	1	0	1	0	0

图像

2	0	0
0	0	0
0	0	0

(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

$$\text{Orange Box} = \sum \left(\text{Green Box} \cdot \text{Dark Green Box} \right)$$

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

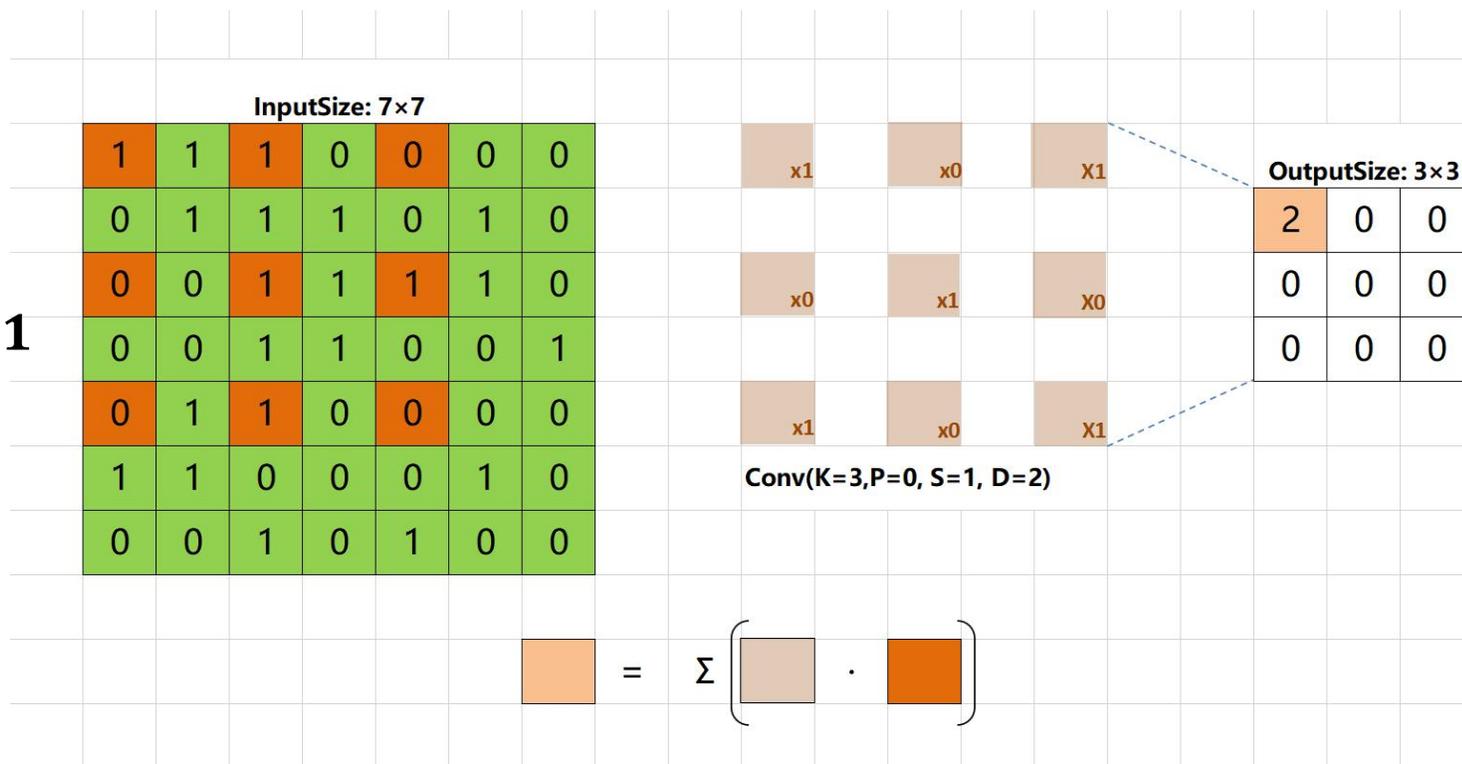
$$H_{out} = \frac{H_{in} + 2P - (D * (K - 1) + 1)}{S} + 1$$

Padding: $2P$
 Dilation: D
 KernelSize: K
 stride: S

● 空洞卷积核的尺寸

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

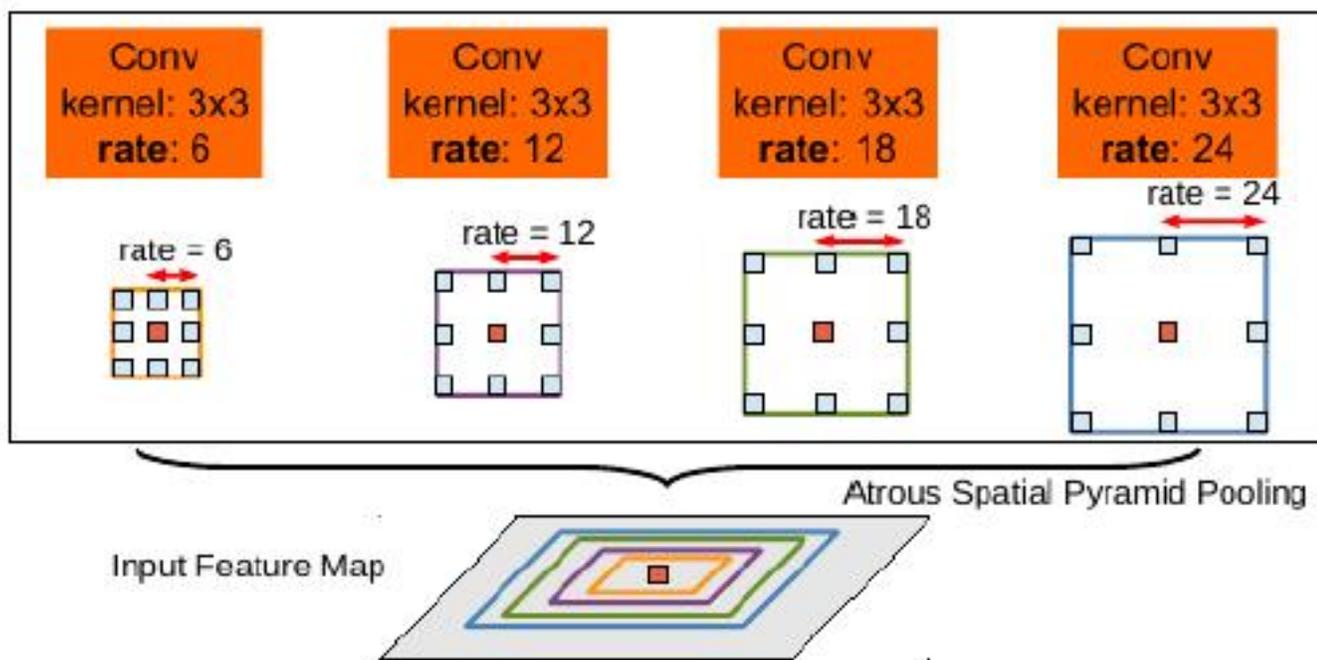
空洞卷积的KernelSize



3.5.1 DeepLab的核心技术

空洞空间金字塔池化

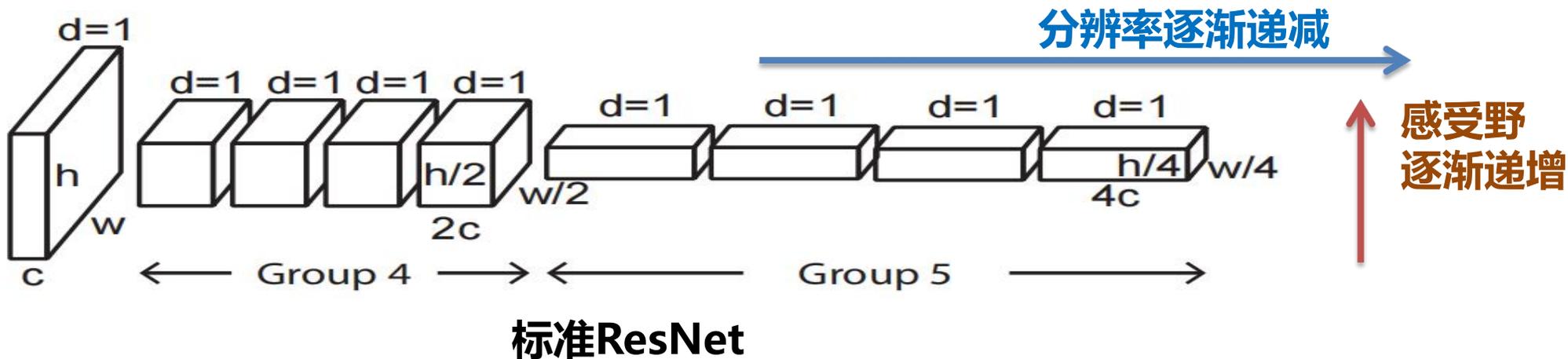
空洞空间金字塔池化(Atrous Spatial Pyramid Pooling, ASPP), 是DeepLab的**核心结构**, 用于从**多个不同视野中进行特征提取**, 并最后进行融合以达到**上下文信息融合**的作用。



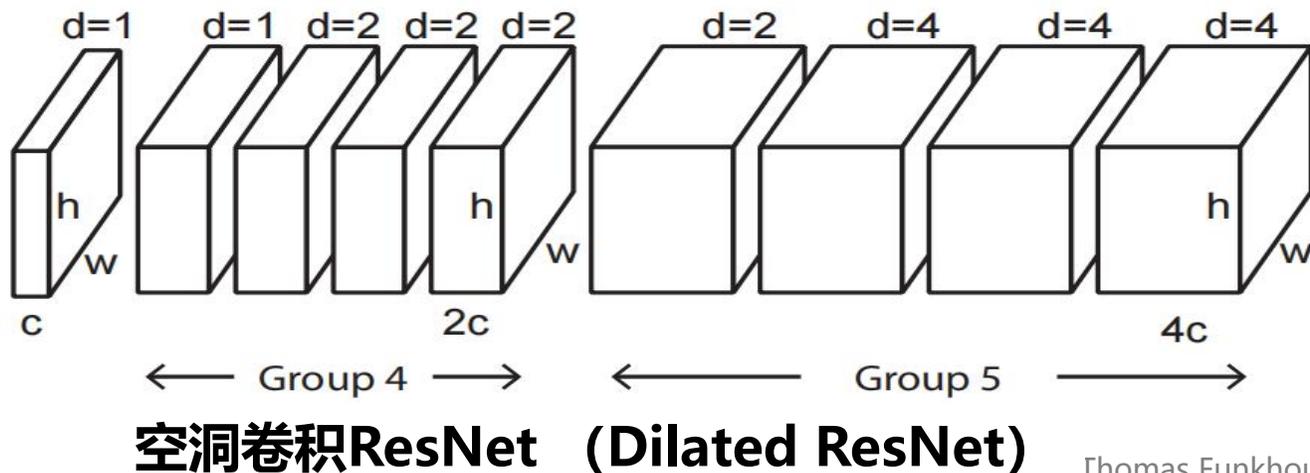
rate可以理解为空洞的尺度, 空洞卷积尺度 = $K + (K-1) * (rate-1)$

3.5.1 DeepLab的核心技术

Dilated ResNet



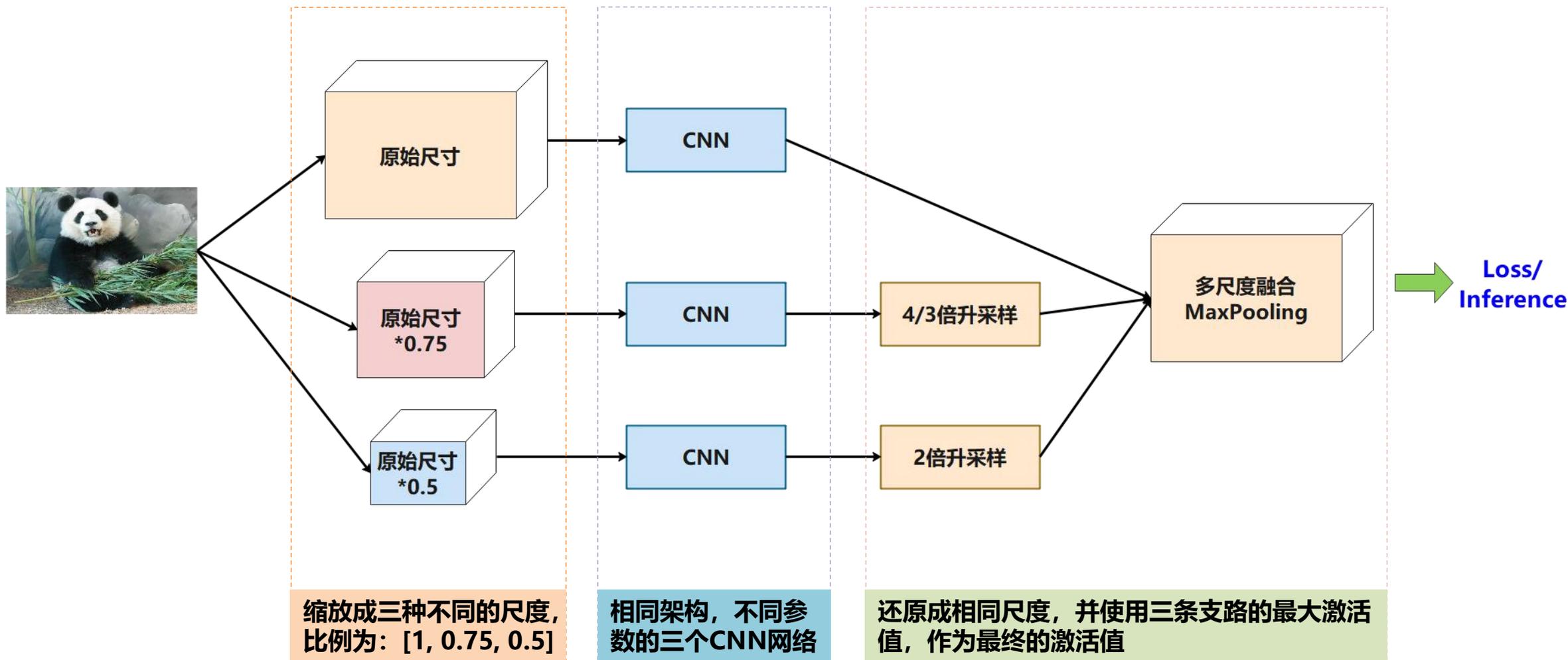
PSP网络的 Backbone



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。

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)

DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs

来自 arXiv.org | ♡ 喜欢 0 阅读量: 16741

作者: LC Chen, G Papandreou, I Kokkinos, K Murphy, AL Yuille

摘要: In this work we address the task of semantic image segmentation with deep convolutional neural networks. It also allows us to explicitly control the resolution of feature responses computed by Deep Convolutional Neural Networks. It also allows us to explicitly control the resolution of feature responses computed by Deep Convolutional Neural Networks. It also allows us to explicitly control the resolution of feature responses computed by Deep Convolutional Neural Networks.

来自 arXiv.org | ♡ 喜欢 0 阅读量: 16743

关键词: Convolutional

DOI: 10.1109/TPAMI

被引量: 1572

年份: 2018

作者: LC Chen, G Papandreou, I Kokkinos, K Murphy, AL Yuille

摘要: In this work we address the task of semantic image segmentation with deep convolutional neural networks. It also allows us to explicitly control the resolution of feature responses computed by Deep Convolutional Neural Networks. It also allows us to explicitly control the resolution of feature responses computed by Deep Convolutional Neural Networks.

关键词: Convolutional neural networks

DOI: 10.1109/TPAMI.2017.2699184

被引量: 1572

年份: 2018

Rethinking Atrous Convolution for Semantic Image Segmentation

来自 arXiv.org | ♡ 喜欢 1 阅读量: 12434

作者: LC Chen, G Papandreou, F Schroff, H Adam

摘要: In this work, we revisit atrous convolution, a powerful tool to explicitly adjust filter's field-of-view as well as control the resolution of feature responses computed by Deep Convolutional Neural Networks, in the application of semantic image segmentation. To handle the problem of segmenting objects at multiple scales, we design

Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation

来自 arXiv.org | ♡ 喜欢 1 阅读量: 7994

作者: LC Chen, Y Zhu, G Papandreou, F Schroff, H Adam

摘要: Spatial pyramid pooling module or encode-decoder structure are used in deep neural networks for semantic segmentation task. The former networks are able to encode multi-scale contextual information by probing the incoming features with filters or pooling operations at multiple rates and multiple effective fields-of-view, while the latter networks can capture sharper object boundaries by gradually recovering the spatial information. In this work, we propose to combine the advantages from both methods. Specifically, our proposed model, [展开](#)

关键词: Computer Science - Computer Vision and Pattern Recognition

DOI: 10.1007/978-3-030-01234-2_49

被引量: 199

年份: 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

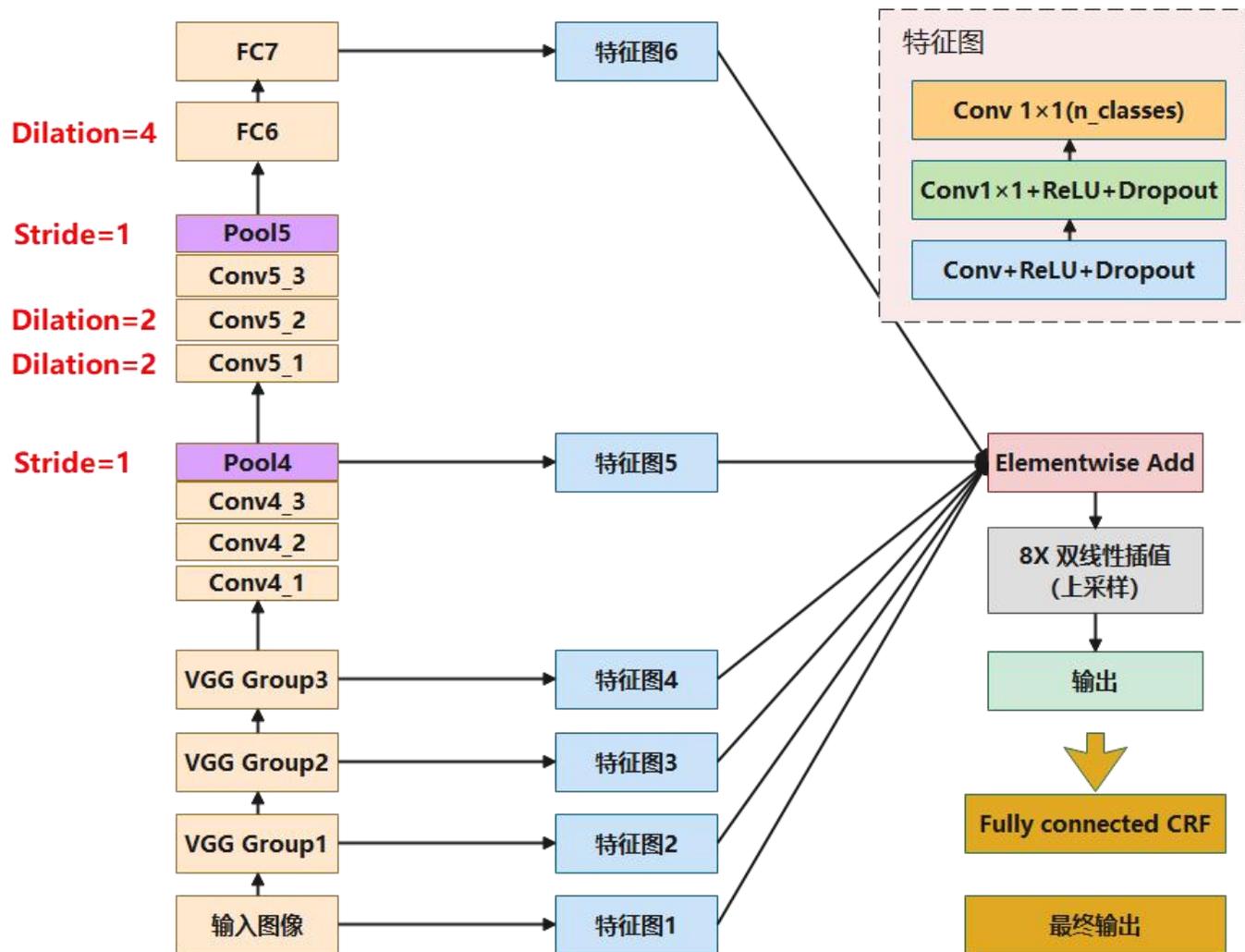
Architecture	Backbone	Atrous Conv	MultiScale	Full-connect CRF
DeepLab v1	VGG-16	Atrous Block	Traning	Yes
DeepLab v2	ResNet	ASPP	Traning	Yes
DeepLab v3	MG ResNet	升级版ASPP	Inference	No
DeepLab v3+	Xception	ASPP+decoder	Inference	No

3.5.2 DeepLab系列网络

DeepLab v1

What is DeepLab 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-connected CRF:** 后处理, 不参与训练。 **优化输出**



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

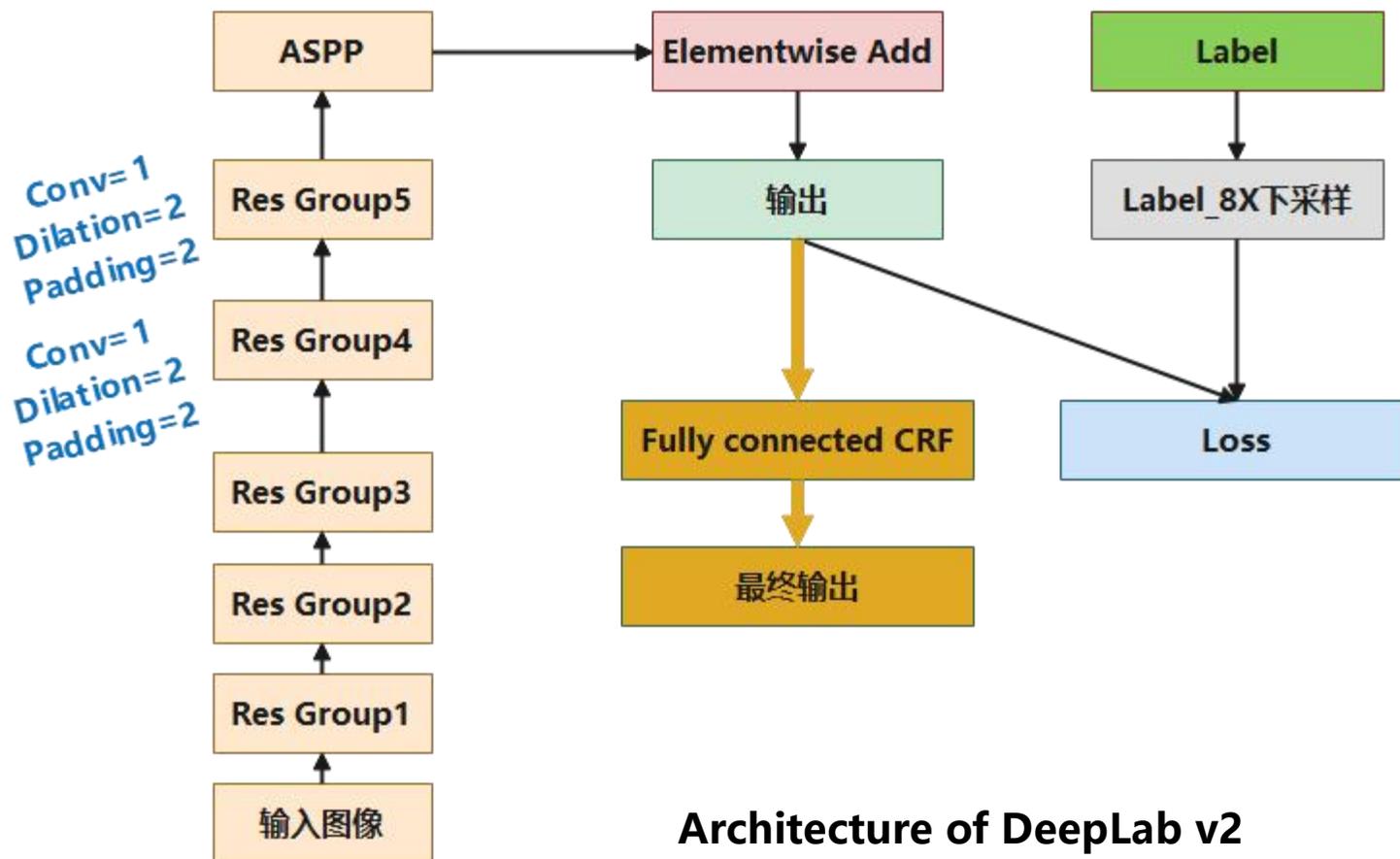
Architecture of DeepLab v1

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

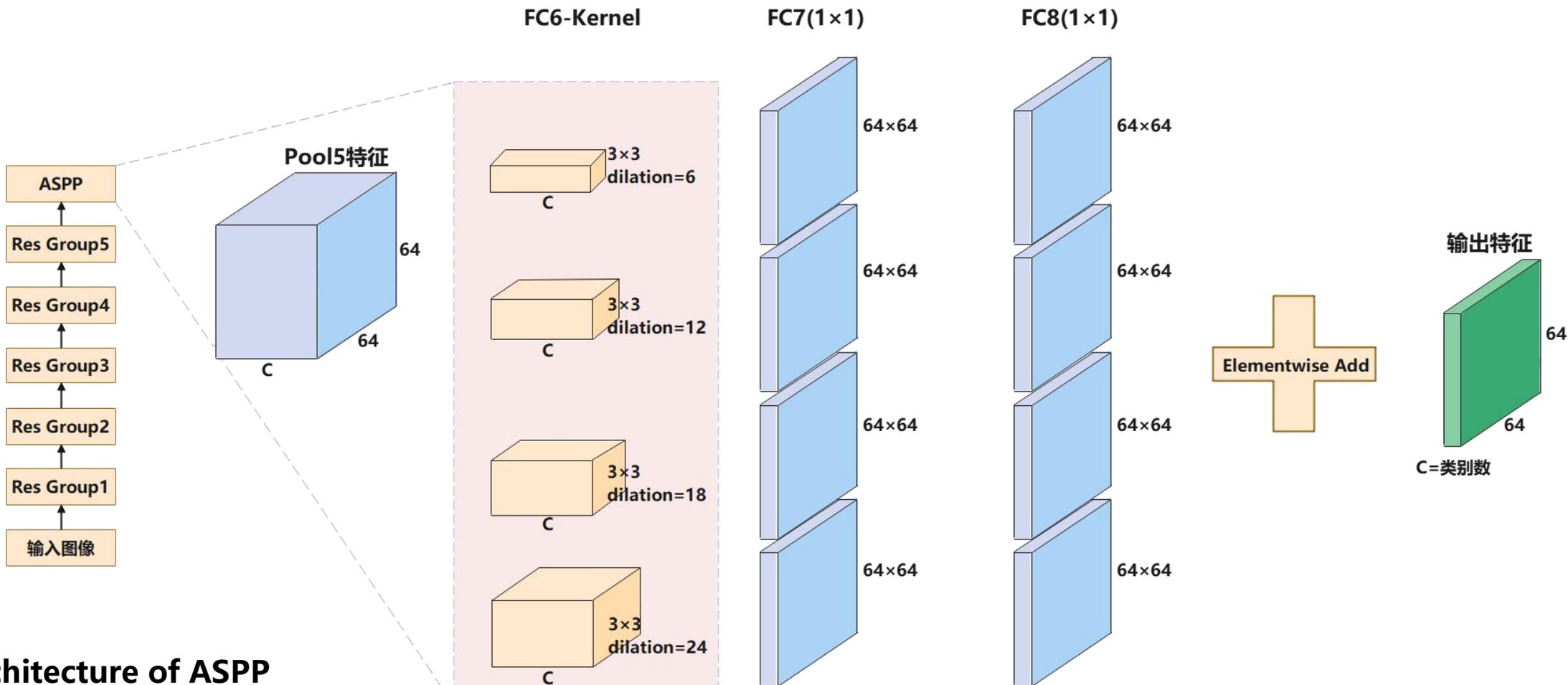


Architecture of DeepLab v2

3.5.2 DeepLab系列网络

DeepLab v2

What is ASPP module ?



Architecture of ASPP

Liang-Chieh Chen, George Papandreou, et al. Segmentation with Deep Convolutional Neural Networks, Atrous Convolution, and Fully Connected CRFs TPAMI2018

3.5.2 DeepLab系列网络

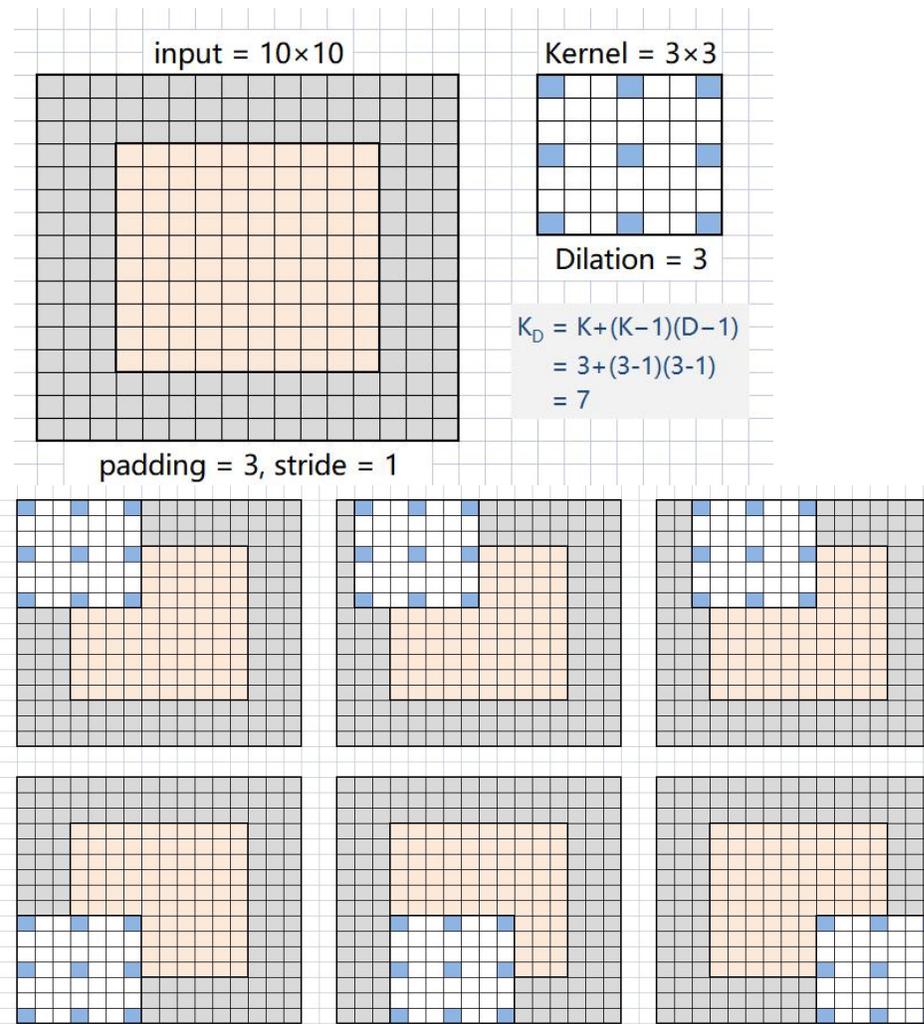
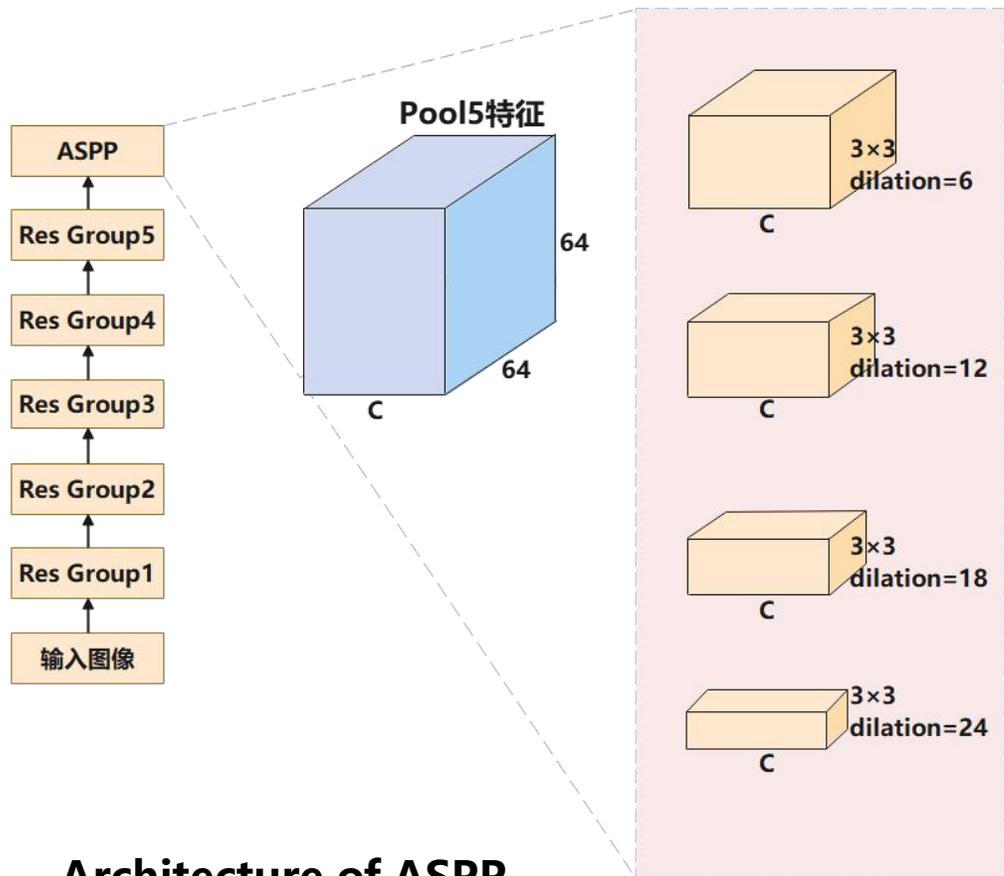
$$H_{out} = \frac{H_{in} + 2P - (D * (K - 1) + 1)}{S} + 1$$

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

$$= 10$$

DeepLab v2

What is ASPP module?



Architecture of ASPP

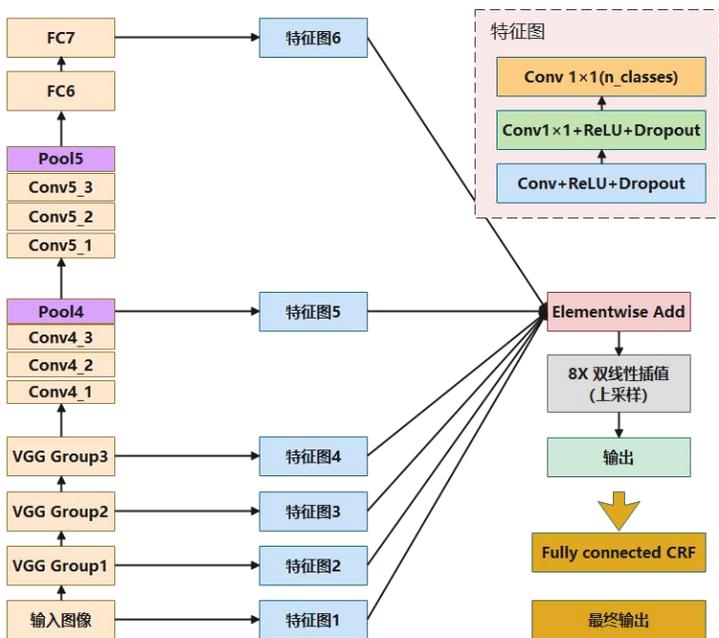
Liang-Chieh Chen, George Papandreou, et.al: DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs TPAMI2018

3.5.2 DeepLab系列网络

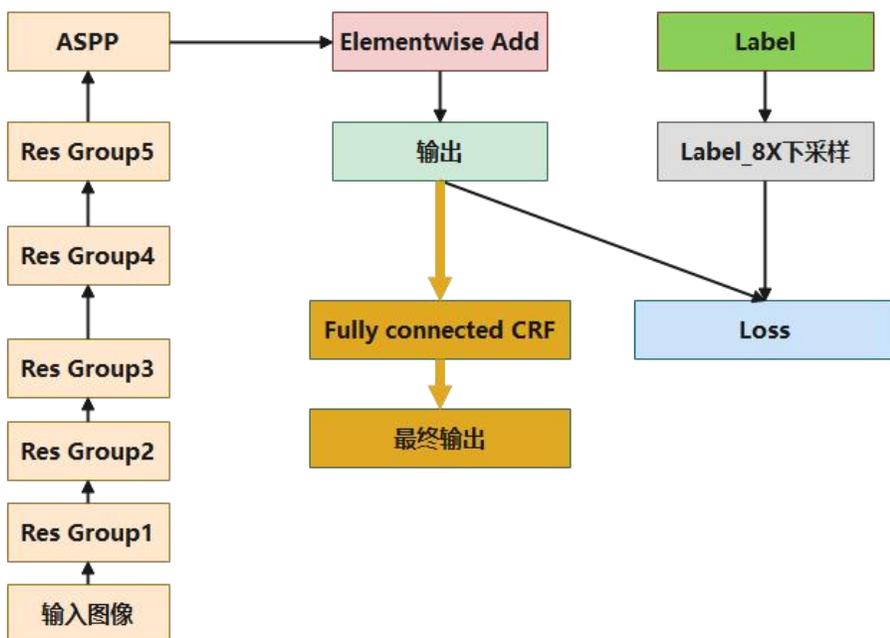
DeepLab v3

What is DeepLab v3 ?

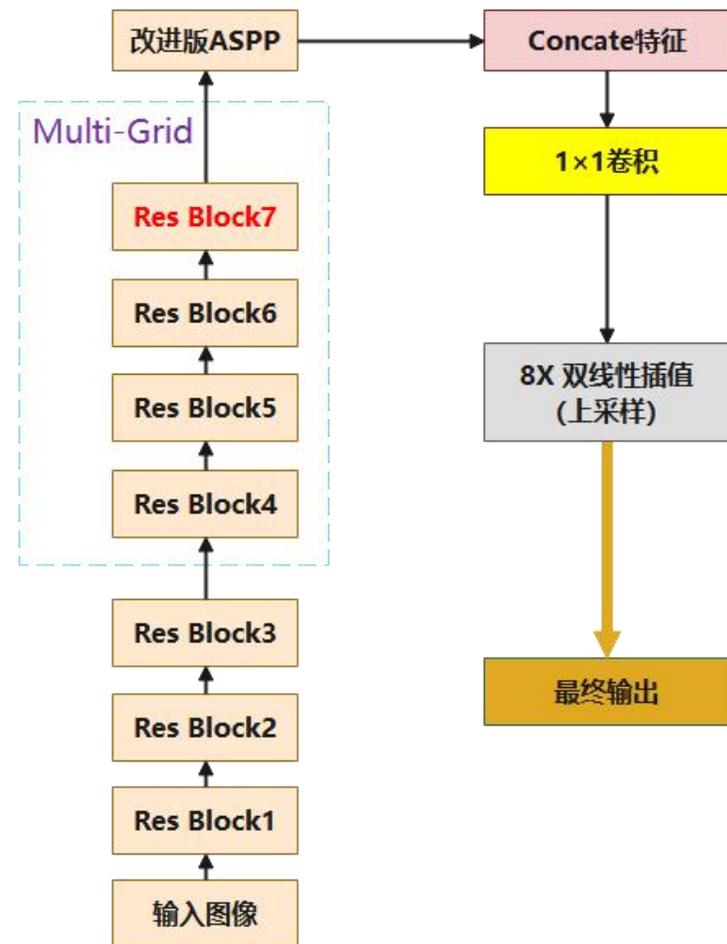
From DeepLab v1 to DeepLab v3



DeepLab v1



DeepLab v2

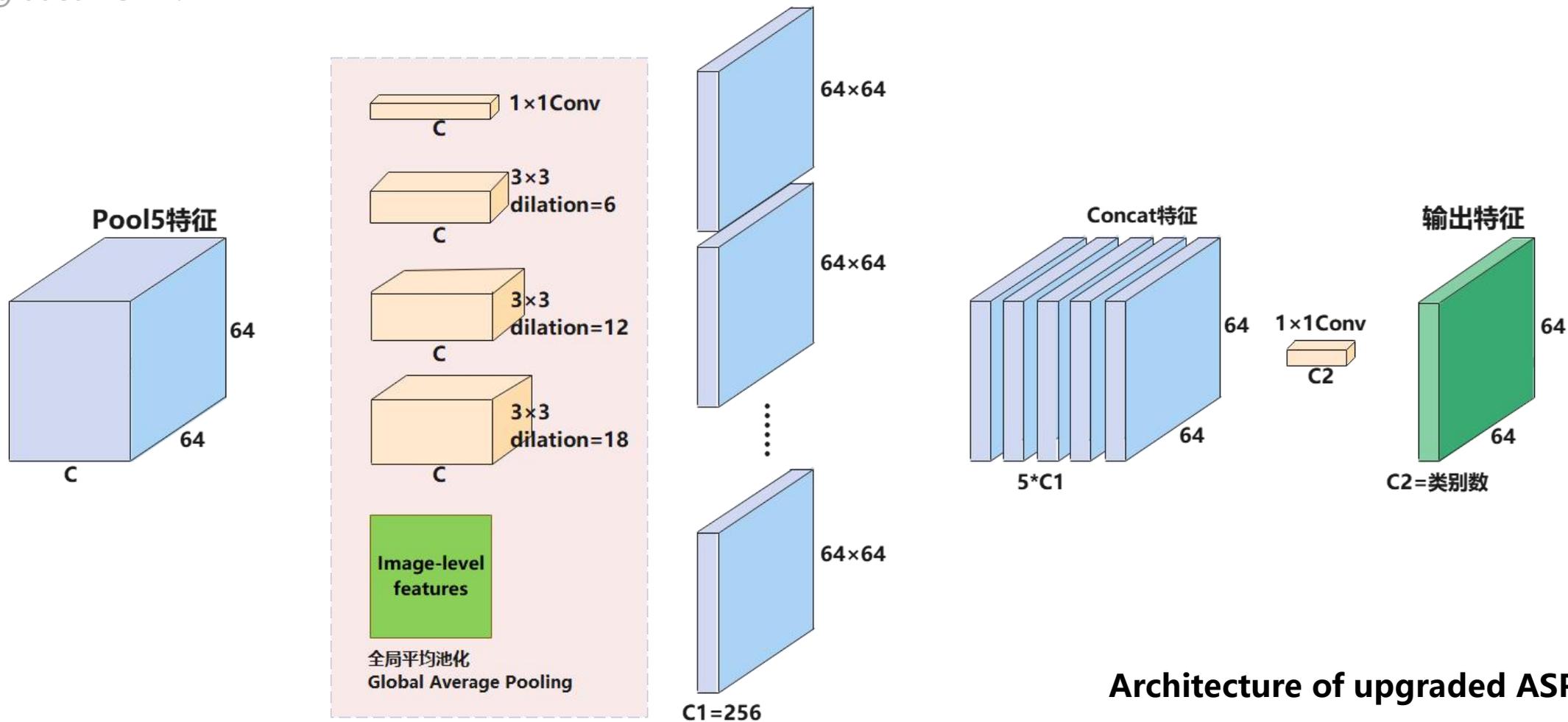


DeepLab v3

3.5.2 DeepLab系列网络

DeepLab v3 - 升级版ASPP模块

What is upgraded ASPP ?



Architecture of upgraded ASPP

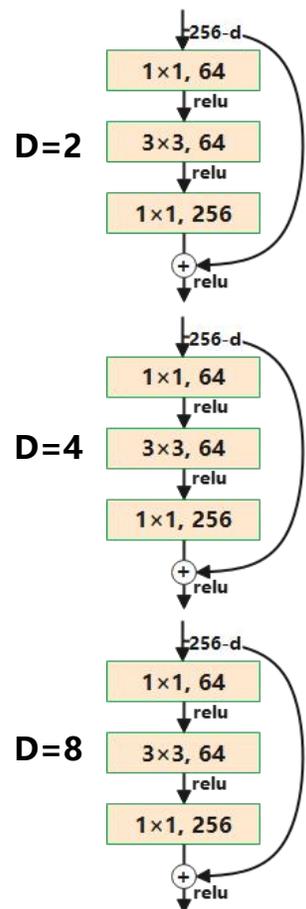
Liang-Chieh Chen, George Papandreou, Florian Schroff, Hartwig Adam. Rethinking Atrous Convolution for Semantic Image. arXiv1706

3.5.2 DeepLab系列网络

DeepLab v3 - MultiGrid

What is MultiGrid?

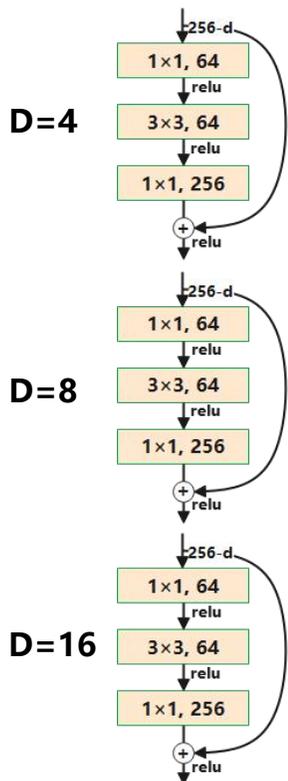
Multi-Grid Rate = (1, 2, 4)



D=4

D=8

Block4, D=2

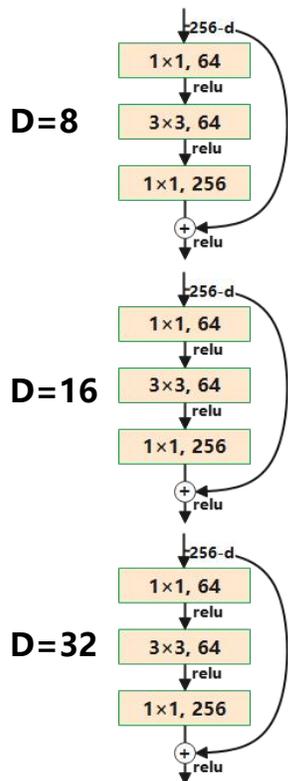


D=4

D=8

D=16

Block5, D=4

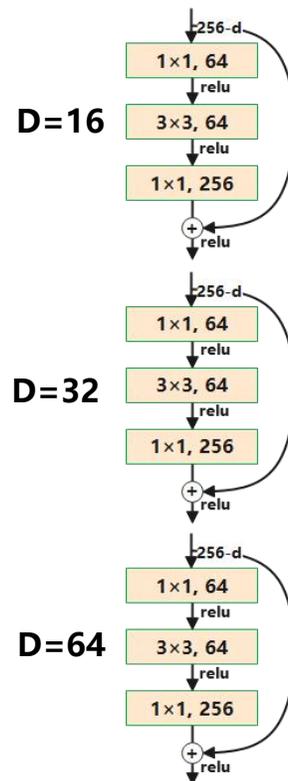


D=8

D=16

D=32

Block6, D=8

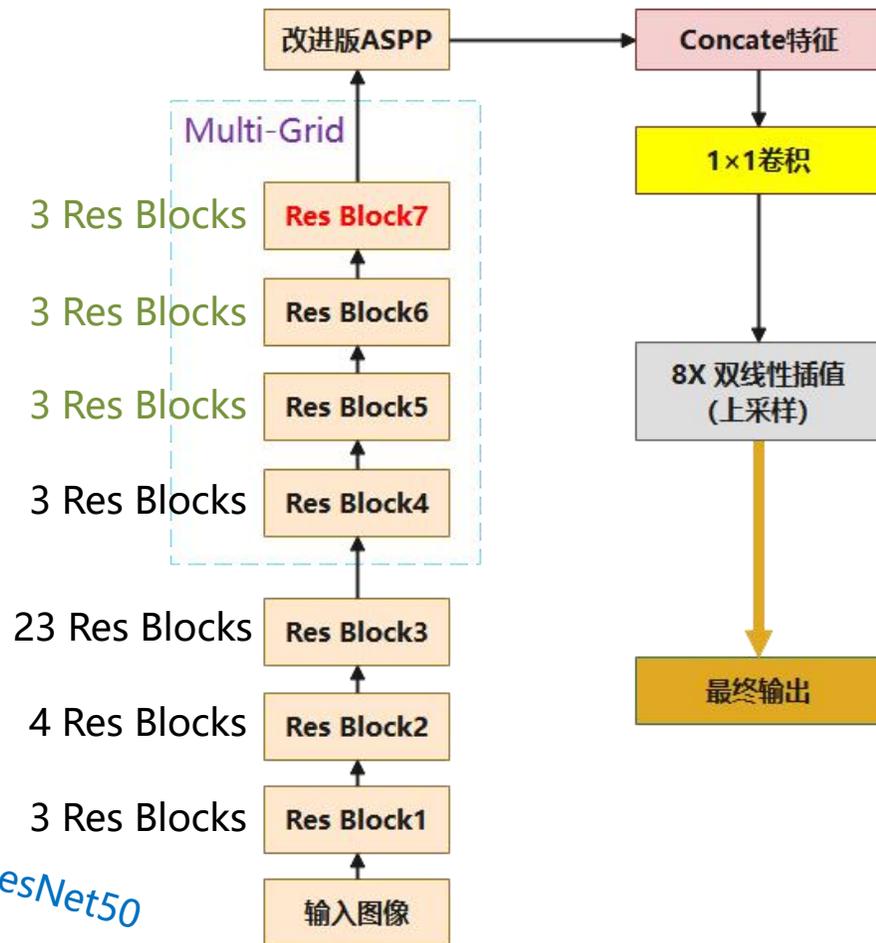


D=16

D=32

D=64

Block7, D=16

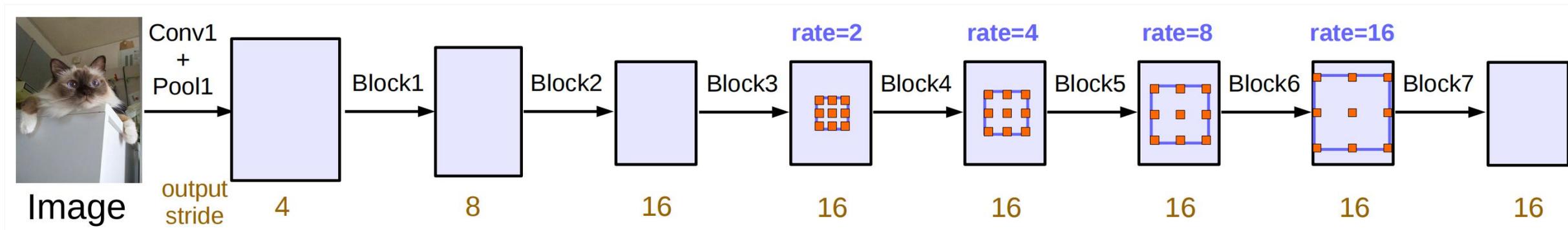


DeepLab v3

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.

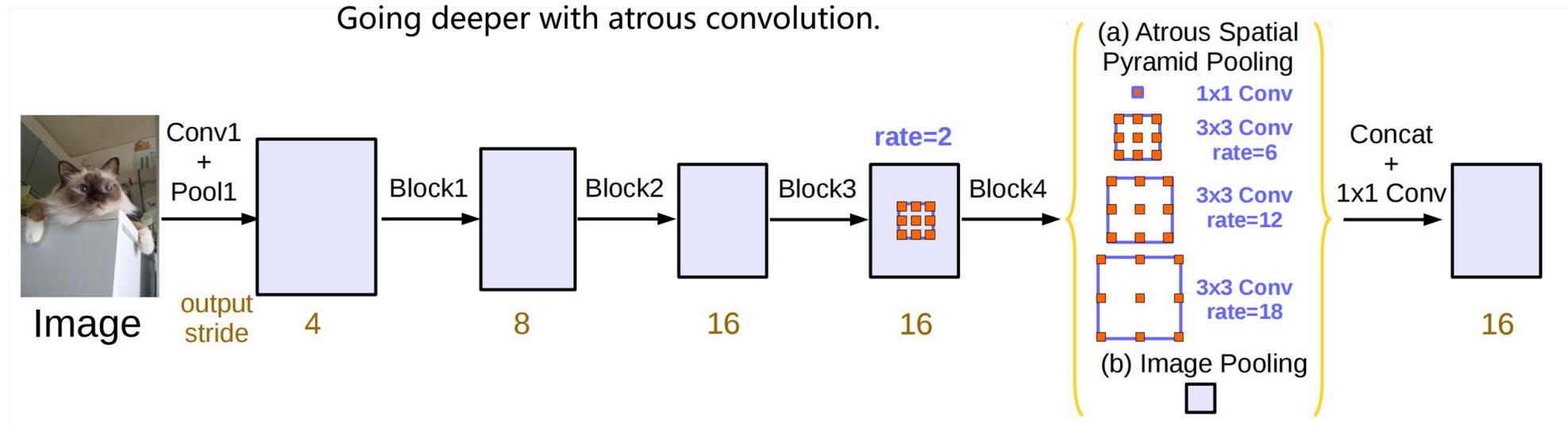


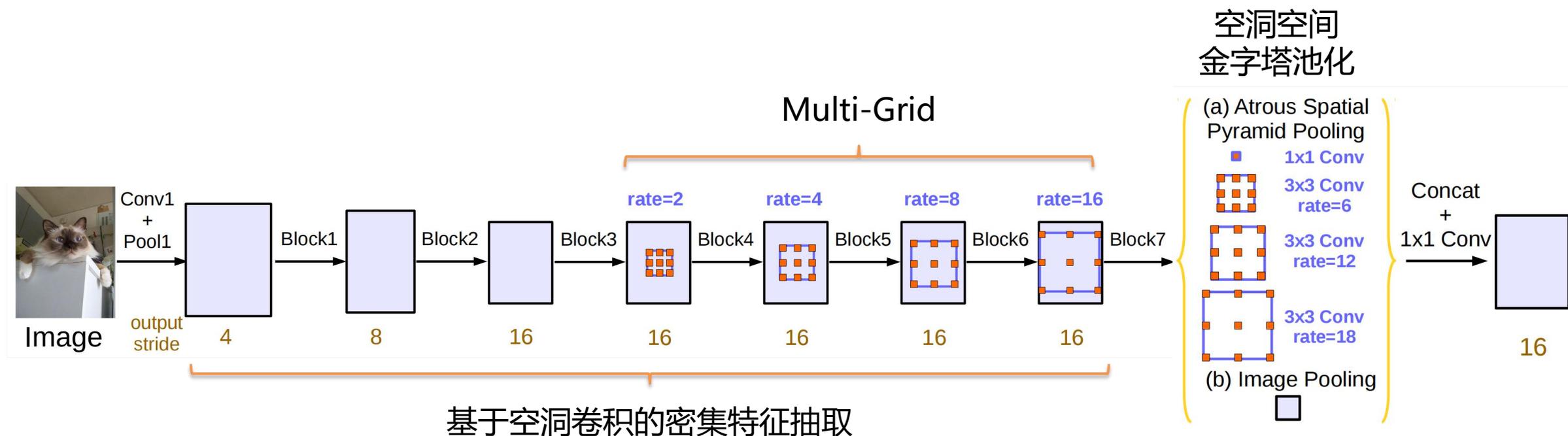
Figure 5. Parallel modules with atrous convolution (ASPP), augmented with image-level features.

mage. arXiv1706

3.5.2 DeepLab系列网络

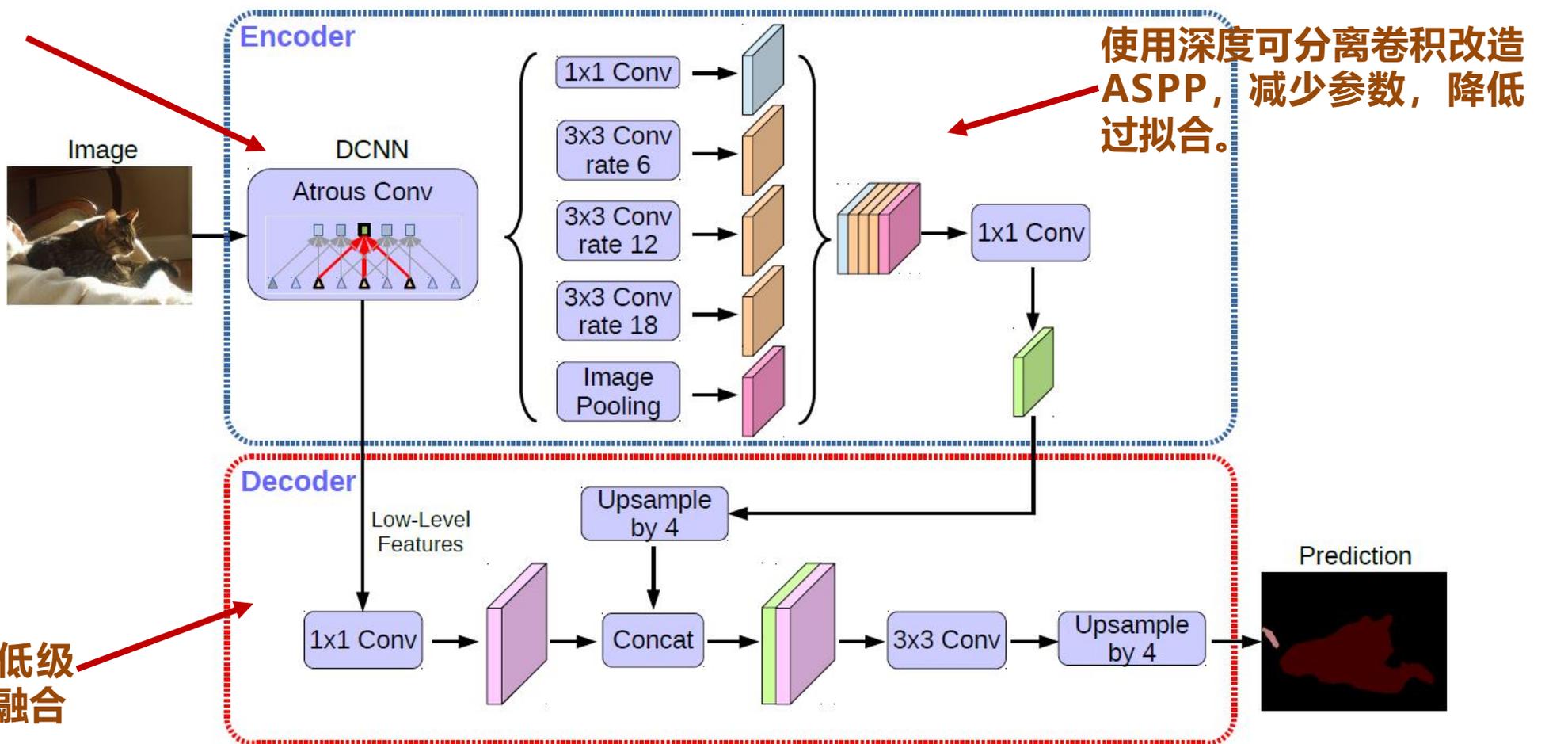
DeepLab v3 - 完整结构

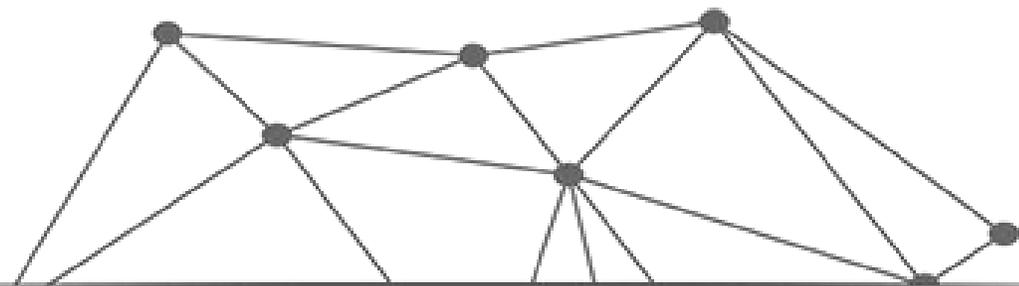
What is DeepLab v3 shown in the vanilla paper?



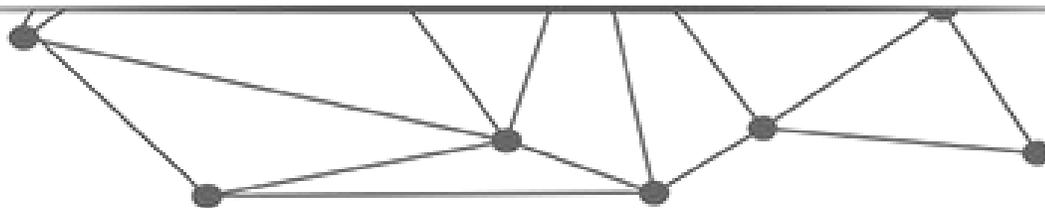
3.5.2 DeepLab系列网络

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。





课堂互动 13.3.5

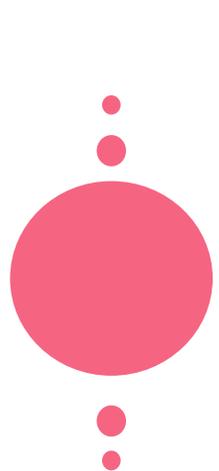


Part
04

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 Instance 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

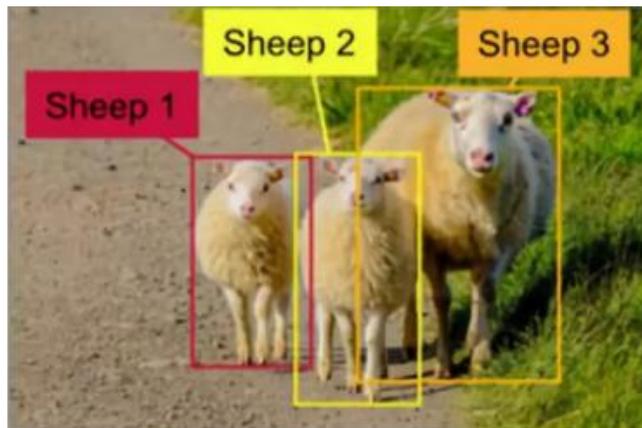
Proposal-based **UPNet** [CVPR2019];

Proposal-free **Panoptic-DeepLab** [CVPR2020]

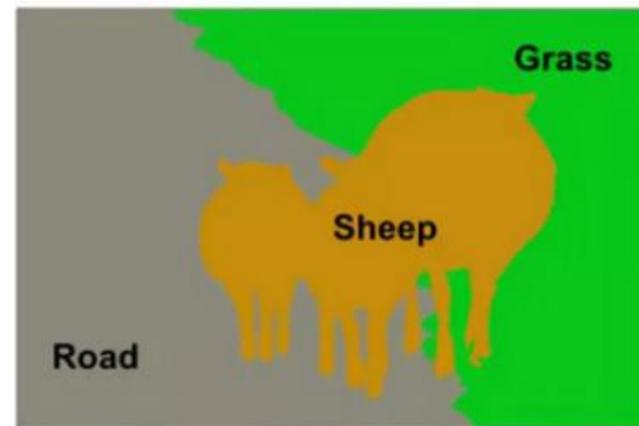
4.1 基本概念

语义分割和全景分割

- 语义分割 (Semantics 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。



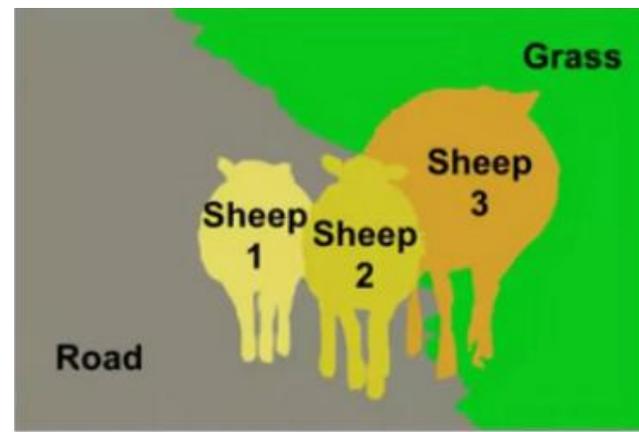
原图像



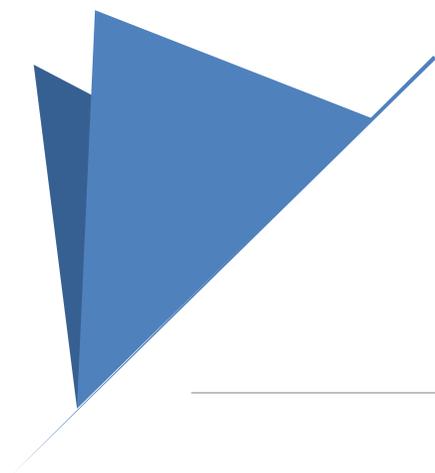
语义分割



实例分割



全景分割



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

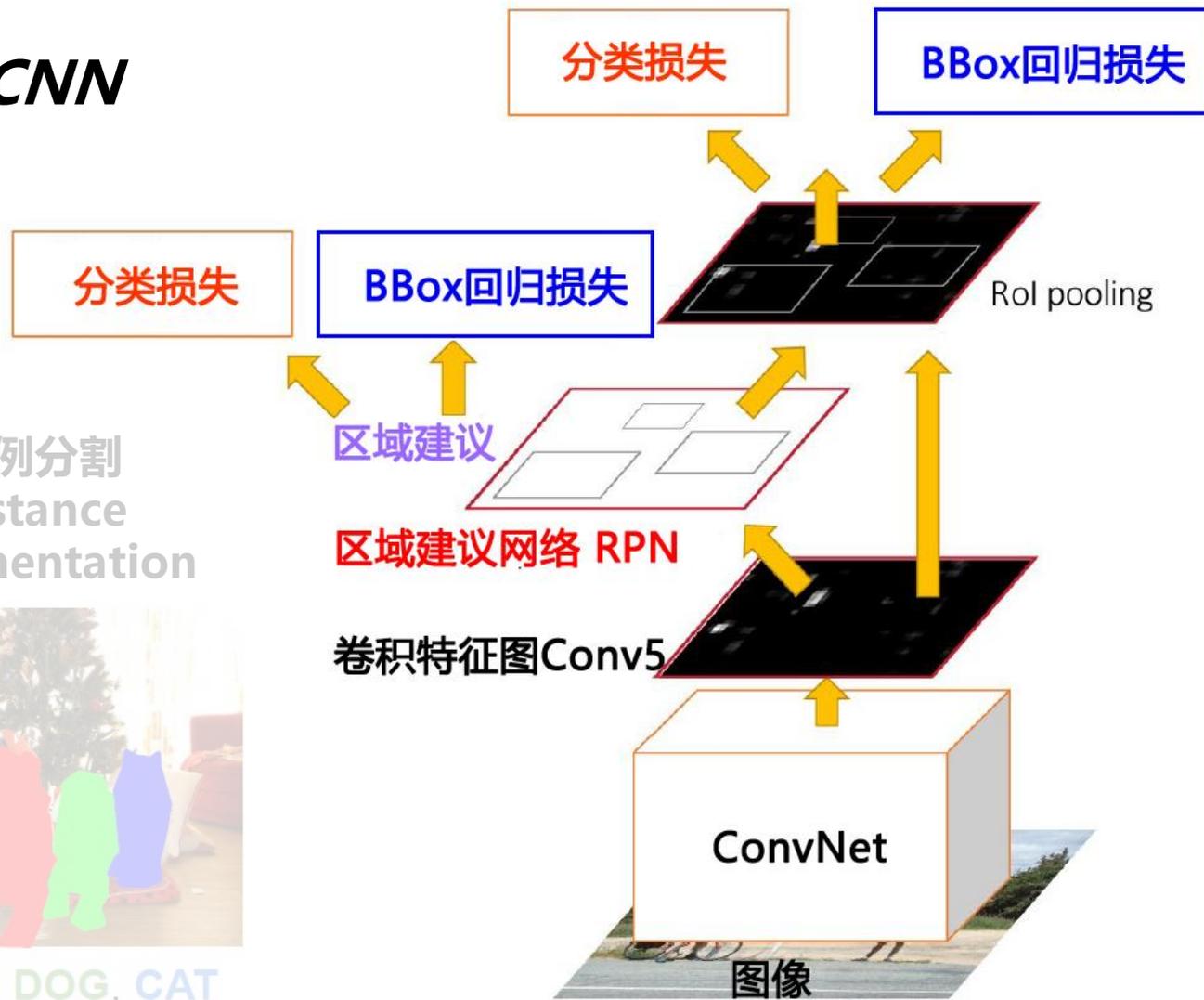


DOG, DOG, CAT

实例分割
Instance Segmentation



DOG, DOG, CAT



4.2 Mask RCNN

基于检测的实例分割Mask R-CNN

实例分割: Mask R-CNN

目标检测
Object Detection

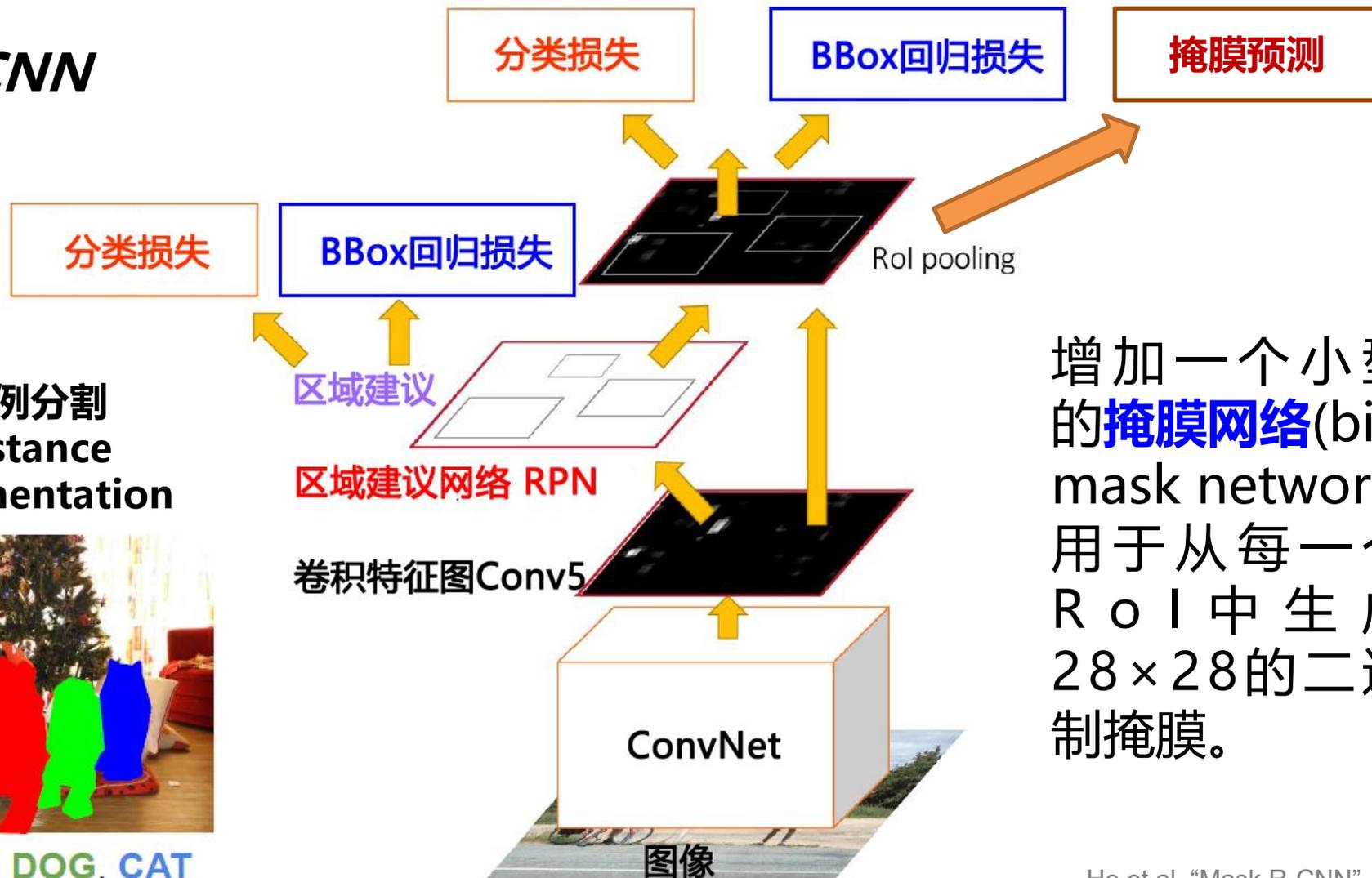


DOG, DOG, CAT

实例分割
Instance Segmentation



DOG, DOG, CAT

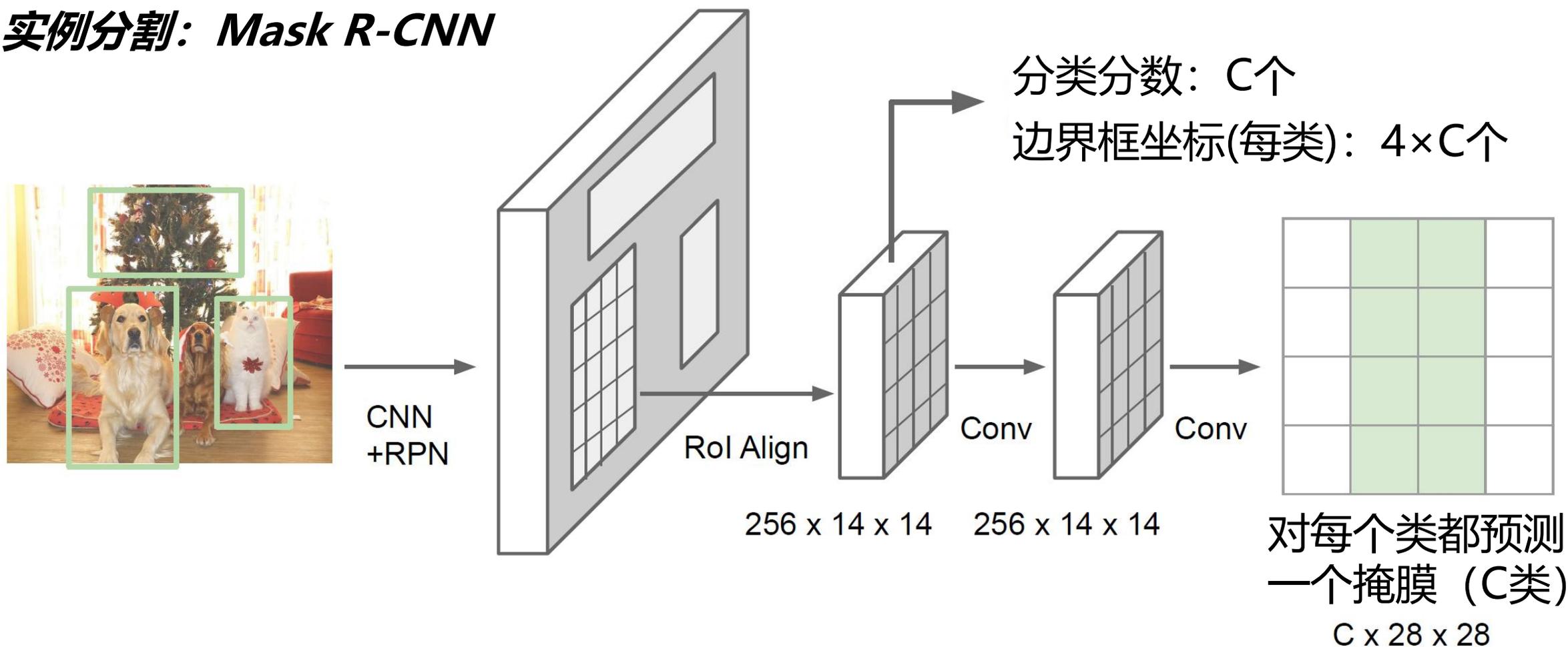


增加一个小型的**掩膜网络**(bin mask network) 用于从每一个 RoI 中生成 28×28 的二进制掩膜。

4.2 Mask RCNN

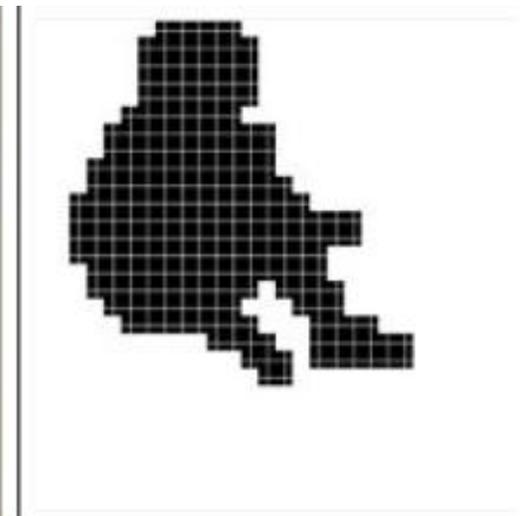
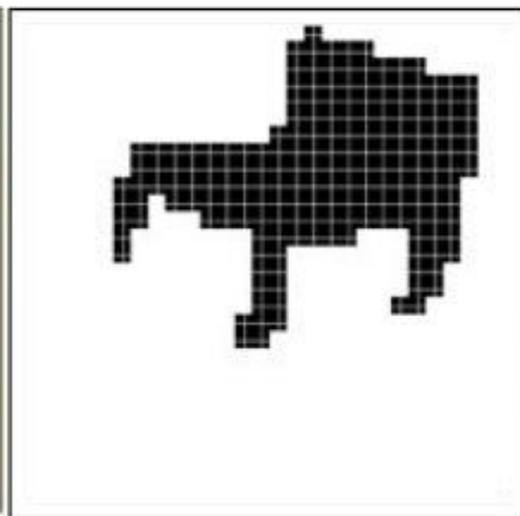
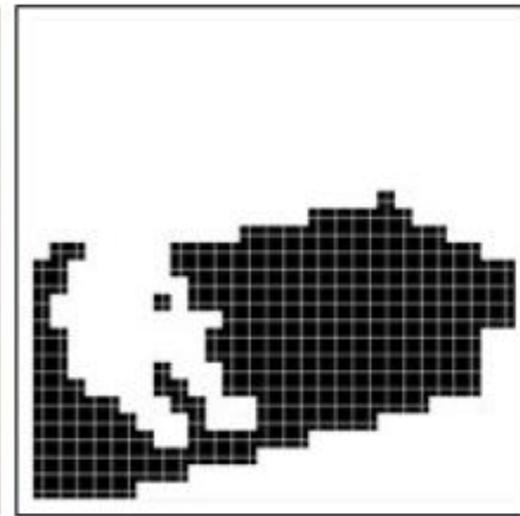
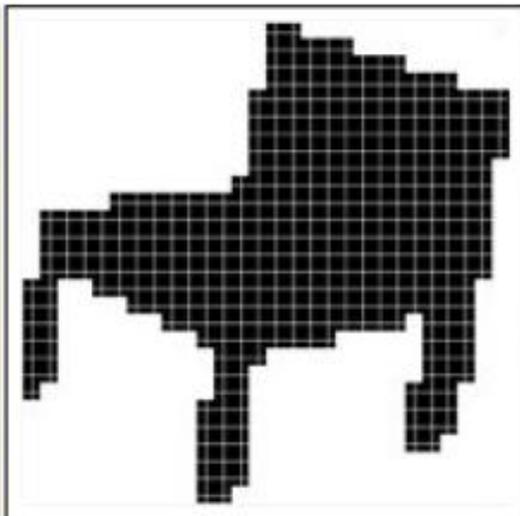
基于检测的实例分割Mask R-CNN

实例分割: Mask R-CNN



4.2 Mask RCNN

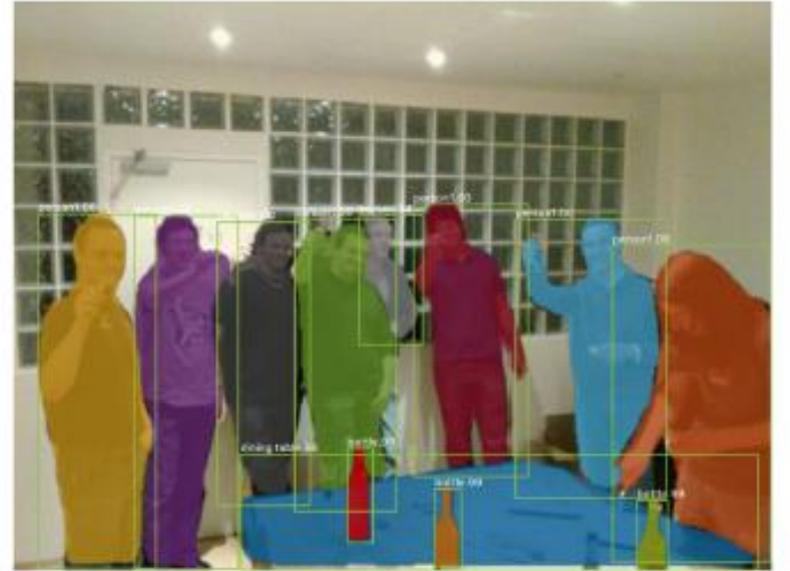
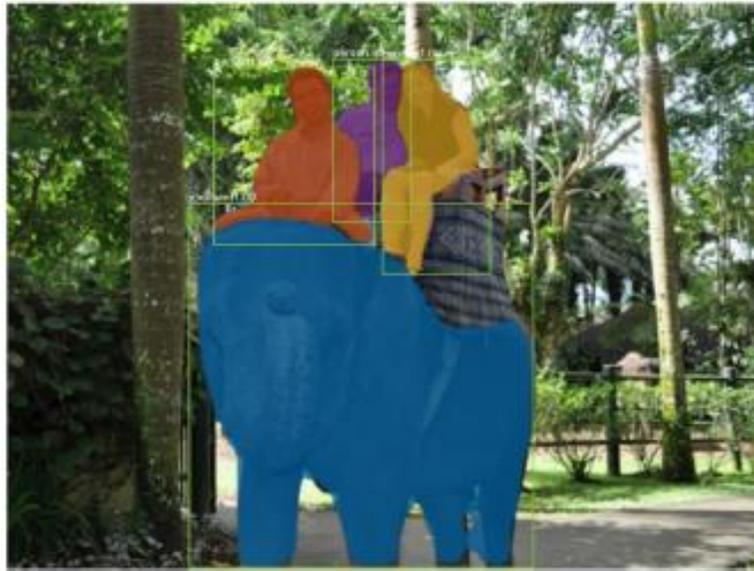
基于检测的实例分割Mask R-CNN



4.2 Mask RCNN

基于检测的实例分割Mask R-CNN

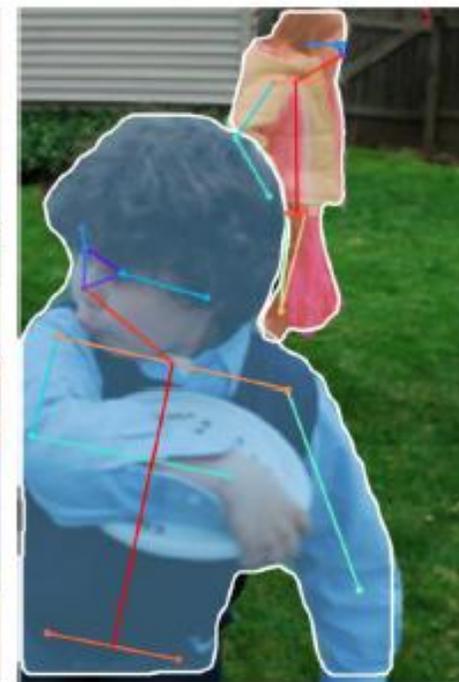
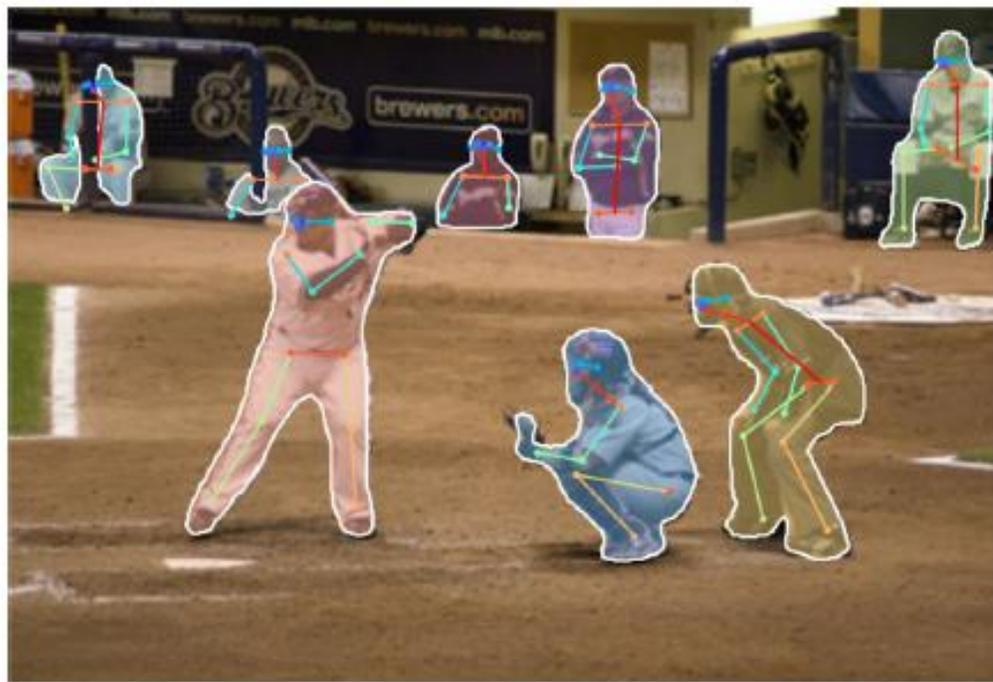
Very Good Results!

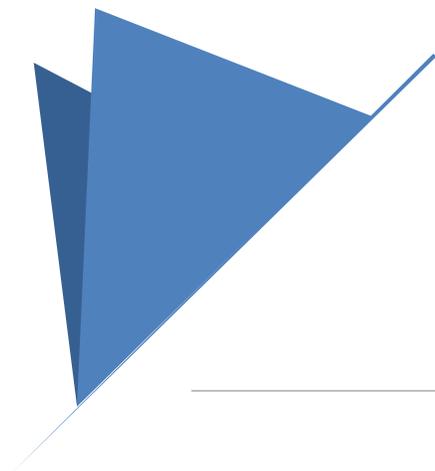


4.2 Mask RCNN

基于检测的实例分割Mask R-CNN

Mask R-CNN 也可以做姿态识别 (Pose)



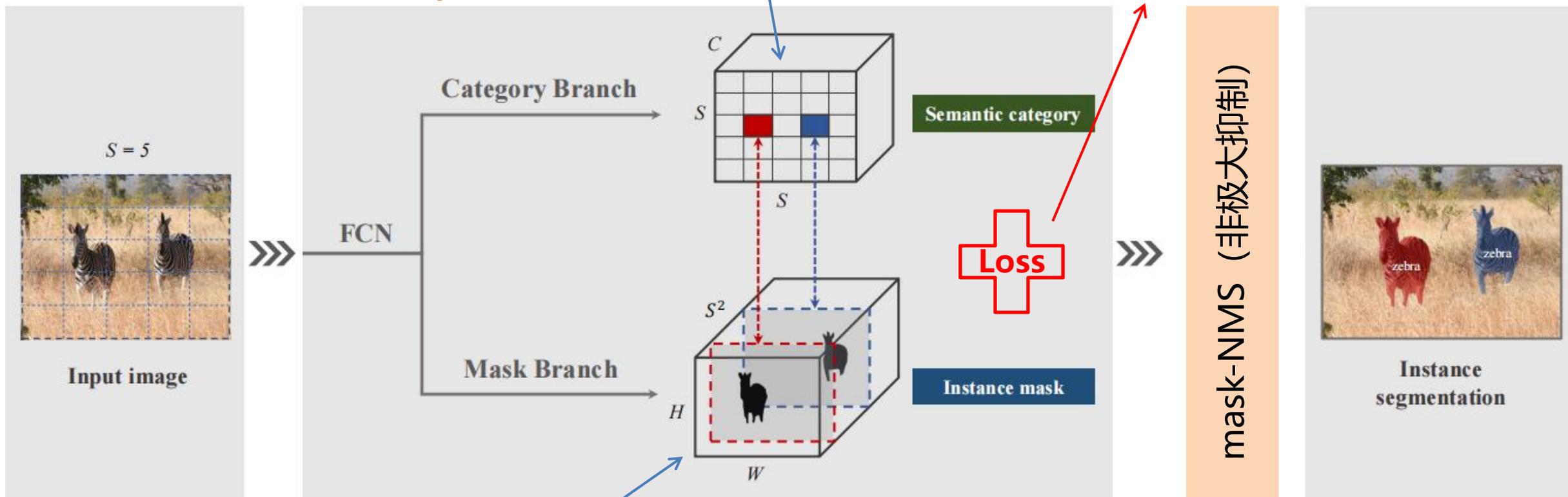


Solo

Segmenting Object by Location

4.3 Solo: Segmenting Object by Location

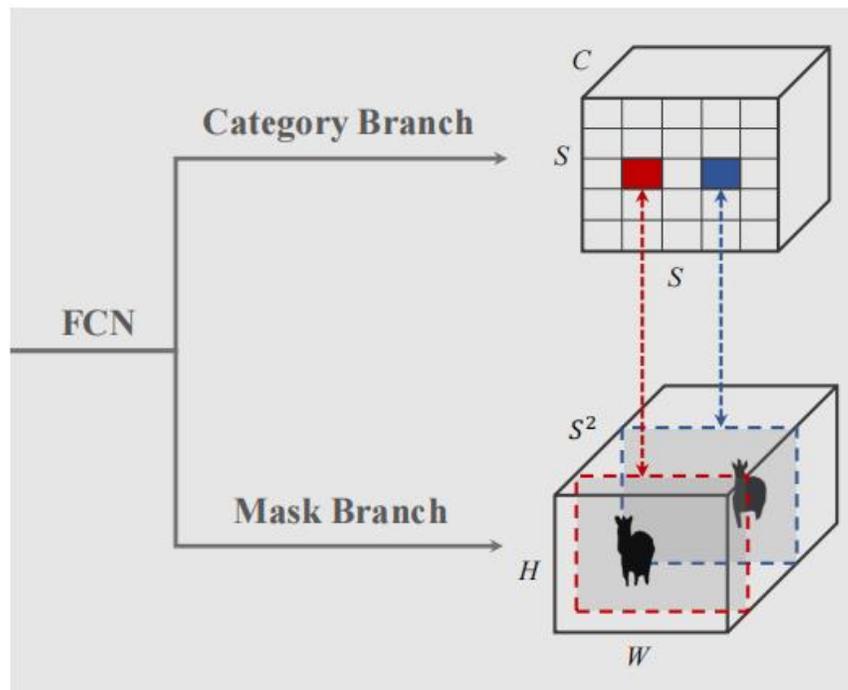
特征图被划分为 $S \times S$ 的网格，每个网格都进行语义类别识别（基于概率），即每个网格只能属于一个类别（类比Yolo）。 S 的典型设置为[12, 24, 36, Pyramid]



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**

4.3 Solo: Segmenting Object by Location

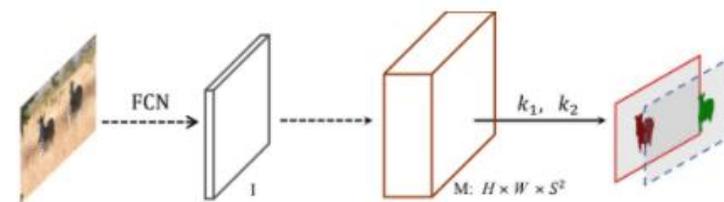
SOLOv2: Dynamic, Faster and Stronger

SOLOv2的改进 (创新点)

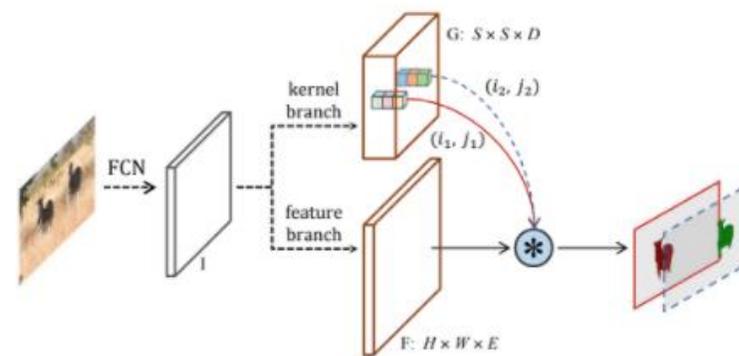
1. 使用**动态卷积**的思想，主动学习mask核，使得网络能够根据位置信息动态地分割实例，有效提高mask的性能。
2. 提出**Matrix NMS**，大大提高了预测阶段的后处理速度。

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



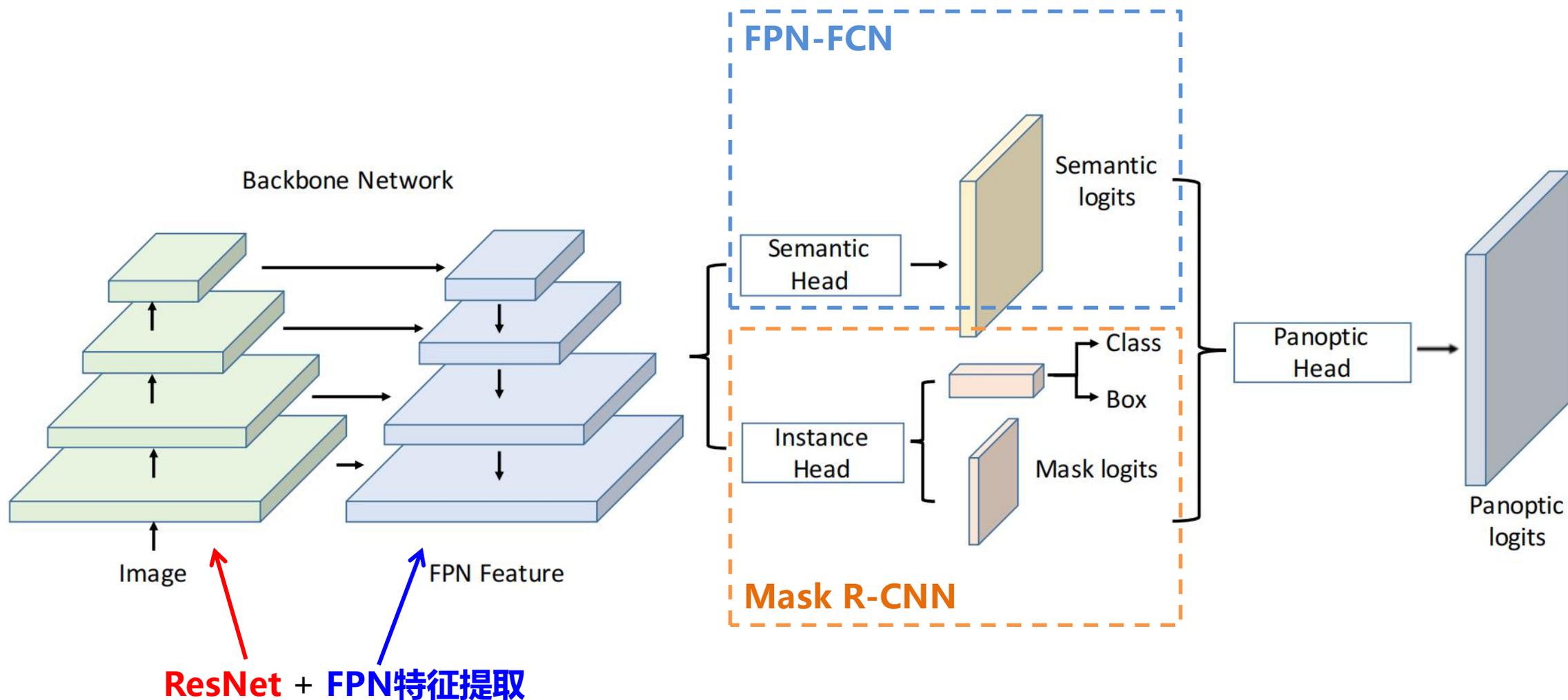
(a) SOLO



(b) SOLOv2

4.4 UPSNet

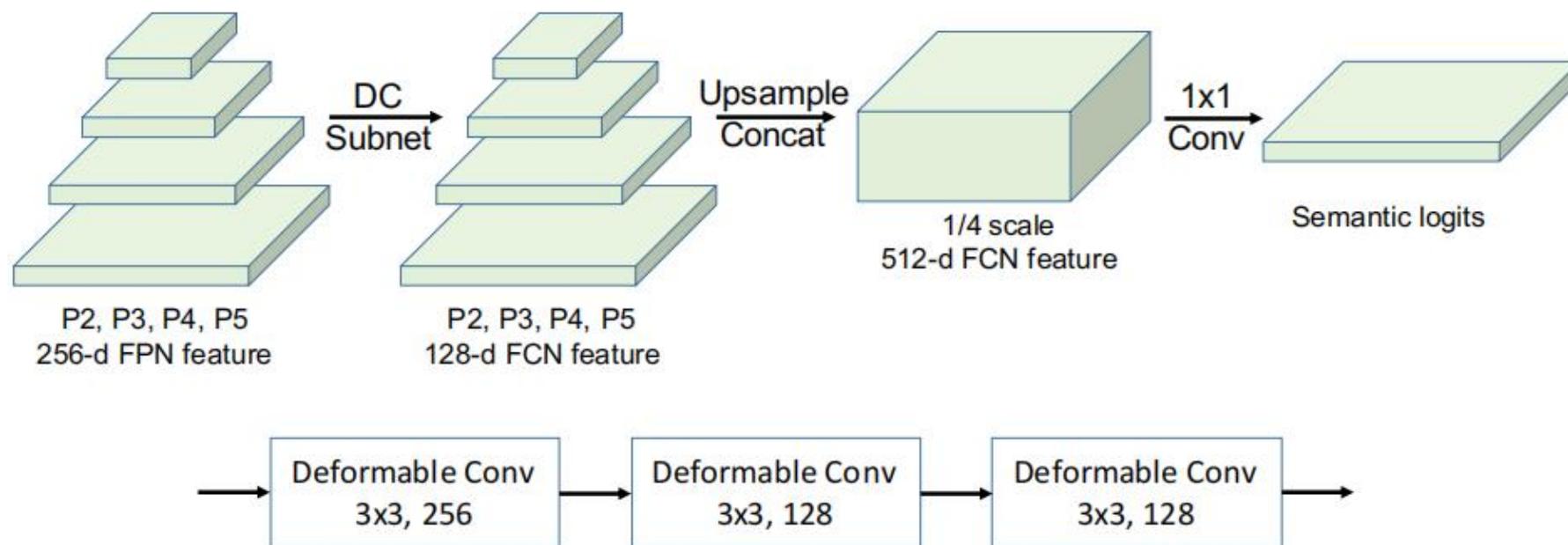
UPSNet总体体系结构



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

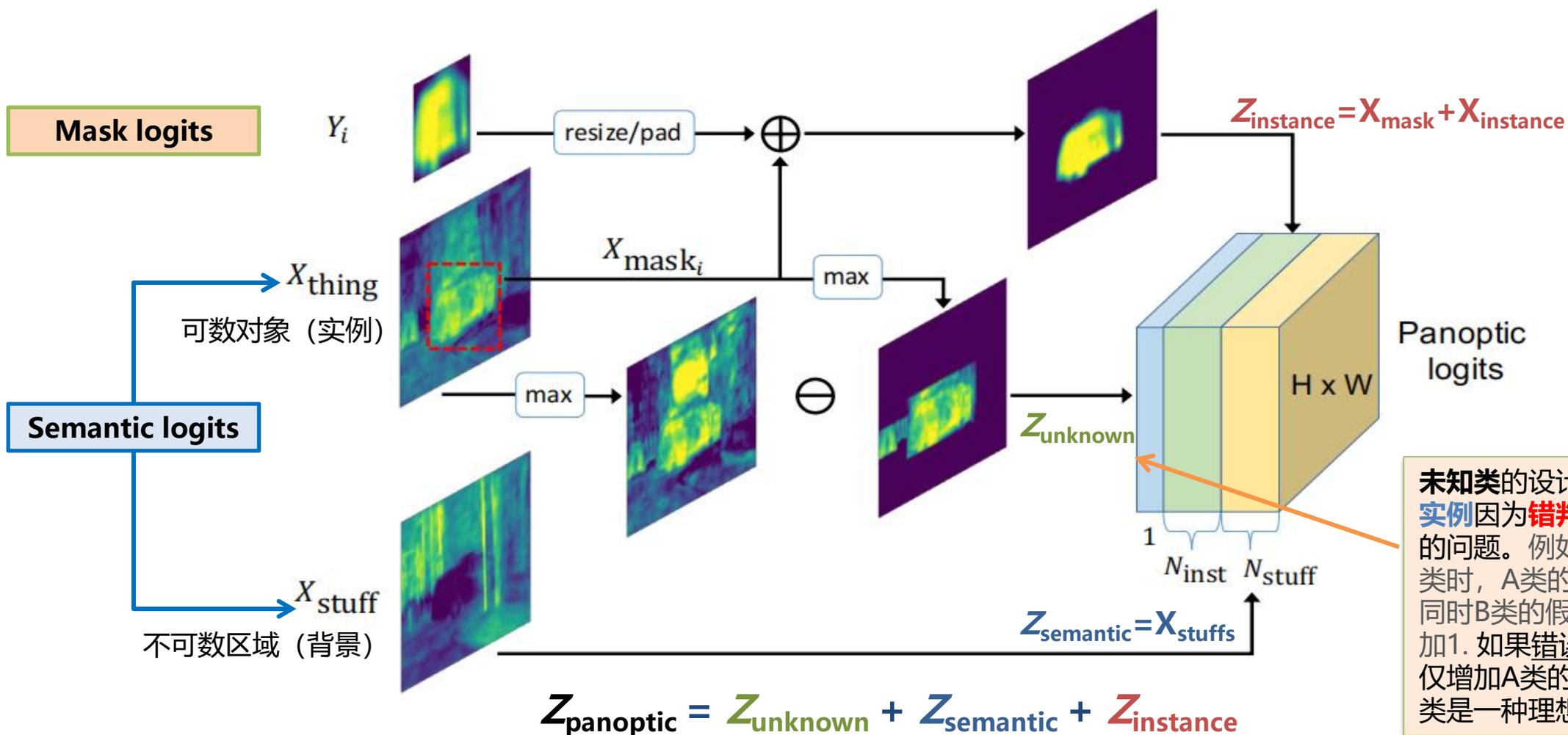
4.4 UPSNet

Semantic Segmentation head: FPN-FCN



4.4 UPSNet

Panoptic Segmentation Head



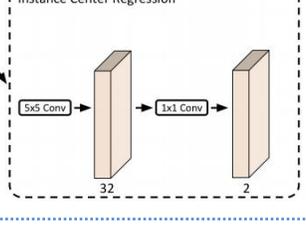
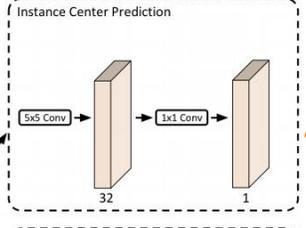
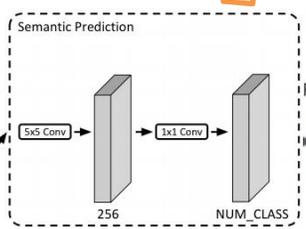
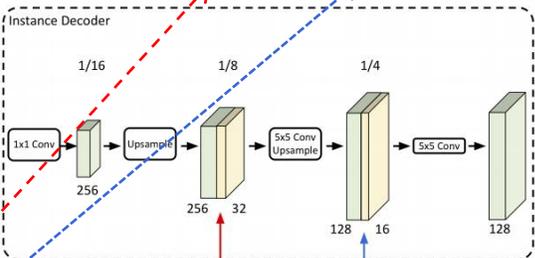
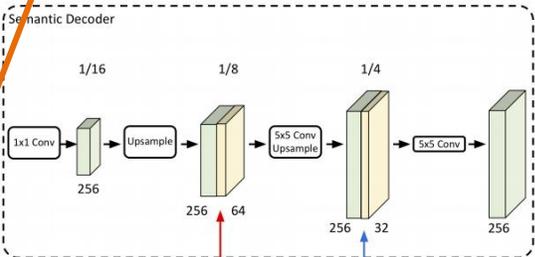
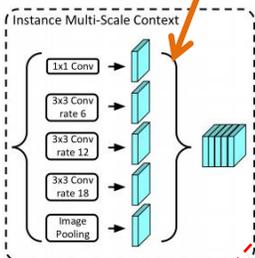
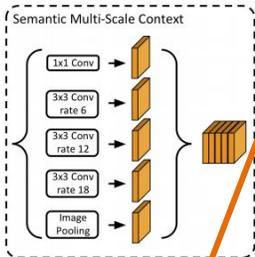
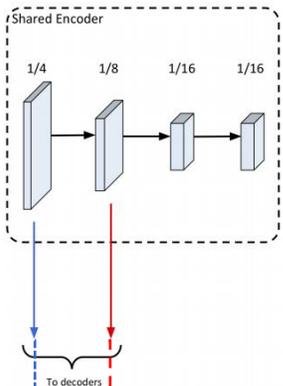
未知类的设计是为了避免一个实例因为错判而导致双重惩罚的问题。例如： A_G 被判定为B类时，A类的假负FN将增加1，同时B类的假阳性 (FP) 也增加1。如果错误不可避免，那么仅增加A类的FN，而不影响B类是一种理想的状态。

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

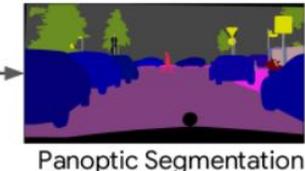
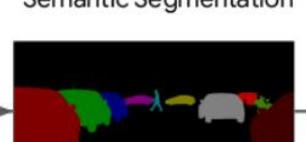
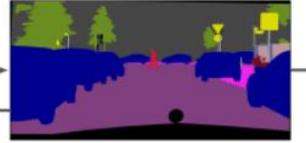
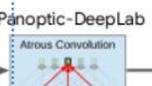
两个独立的ASPP分别用于提取实例多尺度上下文特征和语义多尺度上下文特征

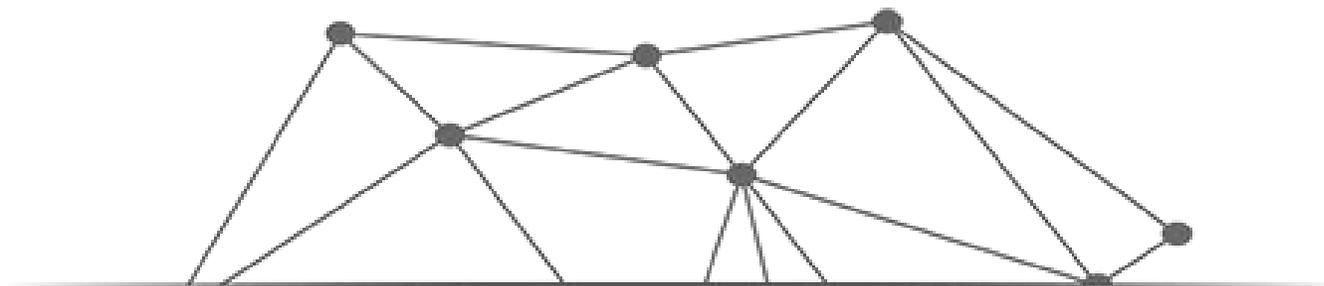
ImageNet预训练
Dilated HRNet



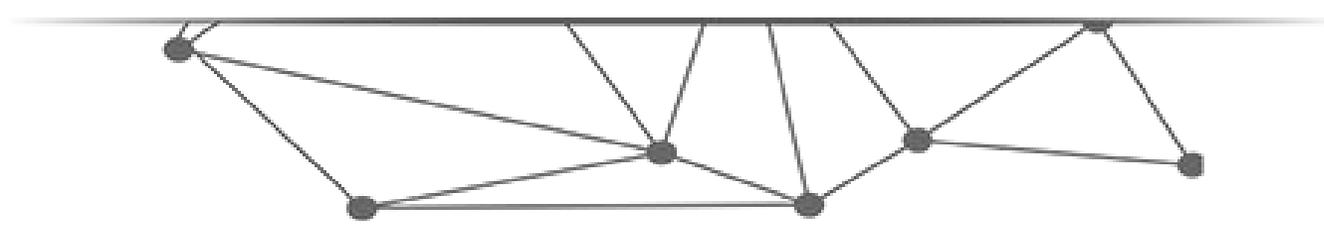
两条编解码通道实现更好的多级特征融合

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





课堂互动 13.3.6



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