



模式识别文献综述

基于深度学习的图像语义分割算法

姓名：罗雁天

院系：清华大学电子系

学号：2018310742

日期：March 14, 2019



目录

| | |
|---|-----------|
| 1 引言 | 3 |
| 1.1 评价指标 | 3 |
| 1.2 数据集 | 4 |
| 1.3 综述结构 | 5 |
| 2 基于深度学习的经典图像语义分割算法 | 7 |
| 2.1 全卷积网络 (FCN) | 7 |
| 2.1.1 卷积化 | 7 |
| 2.1.2 反卷积 | 8 |
| 2.1.3 跳跃结构 | 9 |
| 2.1.4 实验结果 | 9 |
| 2.2 DeepLab | 10 |
| 2.2.1 Hole 算法 | 11 |
| 2.2.2 条件随机场 | 12 |
| 2.2.3 实验结果 | 13 |
| 2.3 DilatedConv | 13 |
| 2.3.1 空洞卷积 | 13 |
| 2.3.2 网络结构 | 15 |
| 2.3.3 实验结果 | 16 |
| 2.4 DeepLab v2 | 17 |
| 2.4.1 ASPP 模块 | 18 |
| 2.4.2 实验结果 | 18 |
| 2.5 PSPNet | 19 |
| 2.5.1 算法结构 | 19 |
| 2.5.2 实验结果 | 20 |
| 2.6 DeepLab v3 | 20 |
| 2.6.1 级联模块 | 21 |
| 2.6.2 新的 ASPP 模块 | 22 |
| 2.6.3 实验结果 | 22 |
| 3 最新图像语义分割算法 | 24 |
| 3.1 Depth-aware CNN | 24 |
| 3.1.1 Depth-aware Convolution | 24 |
| 3.1.2 Depth-aware Average Pooling | 25 |
| 3.1.3 网络结构 | 25 |
| 3.1.4 实验结果 | 26 |

摘要

在计算机视觉领域，图像分割指的是将数字图像细分为多个图像子区域(像素的集合)(也被称作超像素)的过程。图像分割的目的是简化或改变图像的表示形式，使得图像更容易理解和分析。图像分割通常用于定位图像中的物体和边界(线，曲线等)。更精确的，图像分割是对图像中的每个像素加标签的一个过程，这一过程使得具有相同标签的像素具有某种共同视觉特性。

简单来说，图像分割可以看做是像素级别的分类，其在医疗领域、自动驾驶等方面有着重要的应用，在目前的算法研究中，图像分类可以分为语义分割和实例分割。

本文主要从图像语义分割的方向进行调研，介绍了近几年来基于深度学习的图像语义分割算法，并且对这些算法进行对比与总结，最后提出了对图像分割未来方向的展望。

关键字：图像语义分割；深度学习；全卷积网络；条件随机场

Abstract

In the field of computer vision, image segmentation refers to the process of subdividing a digital image into multiple image sub-regions (a collection of pixels) (also referred to as superpixels). The purpose of image segmentation is to simplify or change the representation of the image, making the image easier to understand and analyze. Image segmentation is often used to locate objects and boundaries (lines, curves, etc.) in an image. More precisely, image segmentation is a process of tagging each pixel in an image, which results in pixels with the same tag having some common visual characteristics.

In short, image segmentation can be regarded as a pixel-level classification, which has important applications in the medical field and automatic driving. In the current algorithm research, image classification can be divided into semantic segmentation and instance segmentation.

This paper mainly investigates the direction of image semantic segmentation, introduces the image semantic segmentation algorithm based on deep learning in recent years, and compares and summarizes these algorithms. Finally, it puts forward the prospect of the future direction of image segmentation.

Key Words: Image Semantic Segmentation; Deep Learning; Fully Convolutional Networks; Conditional Random Field

第 1 章 引言

图像分割 (Segmentation) 指的是将数字图像细分为多个图像子区域 (像素的集合)(也被称作超像素) 的过程。图像分割的目的是简化或改变图像的表示形式,使得图像更容易理解和分析。图像分割通常用于定位图像中的物体和边界 (线,曲线等)。更精确的, 图像分割是对图像中的每个像素加标签的一个过程, 这一过程使得具有相同标签的像素具有某种共同视觉特性。

图像分割又可以分为图像语义分割 (Semantic Segmentation) 与实例分割 (Instance Segmentation)。语义分割是在像素级别上的分类, 属于同一类的像素都要被归为一类, 因此语义分割是从像素级别来理解图像的。而实例分割不但要进行像素级别的分类, 还需在具体的类别基础上区别开不同的实例。比如说图像有几个人甲、乙、丙, 那边他们的语义分割结果都是人, 而实例分割结果却是不同的对象。

图像分割在实际应用中非常广泛, 在医学影像中来用来进行肿瘤和其他病理的定位、组织体积的测量、计算机引导的手术等, 在卫星图像中定位物体, 用于人脸识别、指纹识别, 近几年在自动驾驶领域图像分割也起着至关重要的作用。

1.1 评价指标

在图像分割领域主要有如下 4 个评价指标:

- pixel accuracy (Acc): 像素准确率;
- mean pixel accuracy of different categories (mAcc): 类平均像素准确率;
- mean Intersection-over-Union of different categories (mIoU): 类平均识别准确度;
- frequency-weighted IoU (fwIoU): 频率加权的识别准确度。

其中 IoU(Intersection-over-Union) 表示预测位置与真实位置之间的重叠程度, IoU 越高, 预测的位置越准确。图1.1表示了 IoU 的几何意义。



图 1.1: IoU 表示含义示意图

四个指标的计算方式如式1.1所示：

定义 1.1: 计算方式

$$\begin{aligned}
 Acc &= \sum_i \frac{n_{ii}}{s} \\
 mAcc &= \frac{1}{n_C} \sum_i \frac{n_{ii}}{s_i} \\
 mIoU &= \frac{1}{n_C} \sum_i \frac{n_{ii}}{s_i + \sum_j n_{ji} - n_{ii}} \\
 fwIoU &= \frac{1}{s} \sum_i s_i \frac{n_{ii}}{s_i + \sum_j n_{ji} - n_{ii}}
 \end{aligned} \tag{1.1}$$



1.2 数据集

图像分割常用的数据集包括 PASCAL VOC^[1]、MS COCO^[2] 等，包含深度信息图像的数据集有 NYUv2^[3] 等。

- PASCAL VOC(The PASCAL Visual Object Classification) 是目标检测、分类、图像分割领域一个有名的数据集。从 2005 年到 2012 年，共举办了 8 个不同的挑战赛。用于图像分割的 VOC2012 数据集提供原图以及图像语义分割和图像实例分割两种 png 图 (如图1.2所示)，共分为 20 类，包括背景为 21 类，分别如下：
 - Person: person;
 - Animal: bird, cat, cow, dog, horse, sheep;
 - Vehicle: aeroplane, bicycle, boat, bus, car, motorbike, train;
 - Indoor: bottle, chair, dining table, potted plant, sofa, tv/monitor.

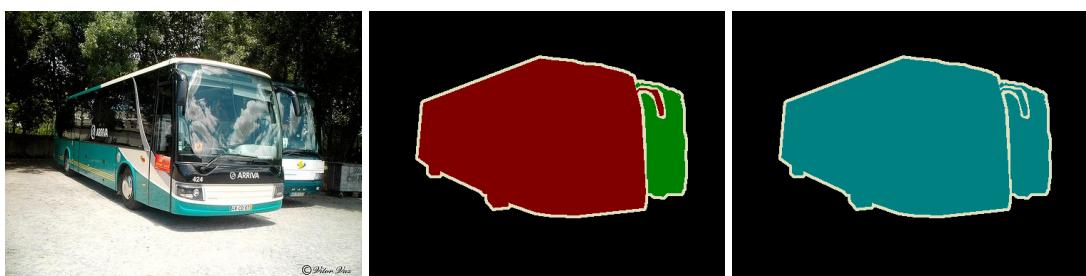


图 1.2: PASCAL VOC 图像分割数据集示例 (左图：原始图像；中图：实例分割的标签图；右图：语义分割的标签图)

- MS COCO(Common Objects in COntext) 是微软建立的数据集。这个数据集也用于多种竞赛：图像标题生成、目标检测、关键点检测和图像分割。图像包括 91 类目标，328000 影像和 2500000 个 label。图1.3展示了 MS COCO 举办的各个竞赛中数据集的示例。

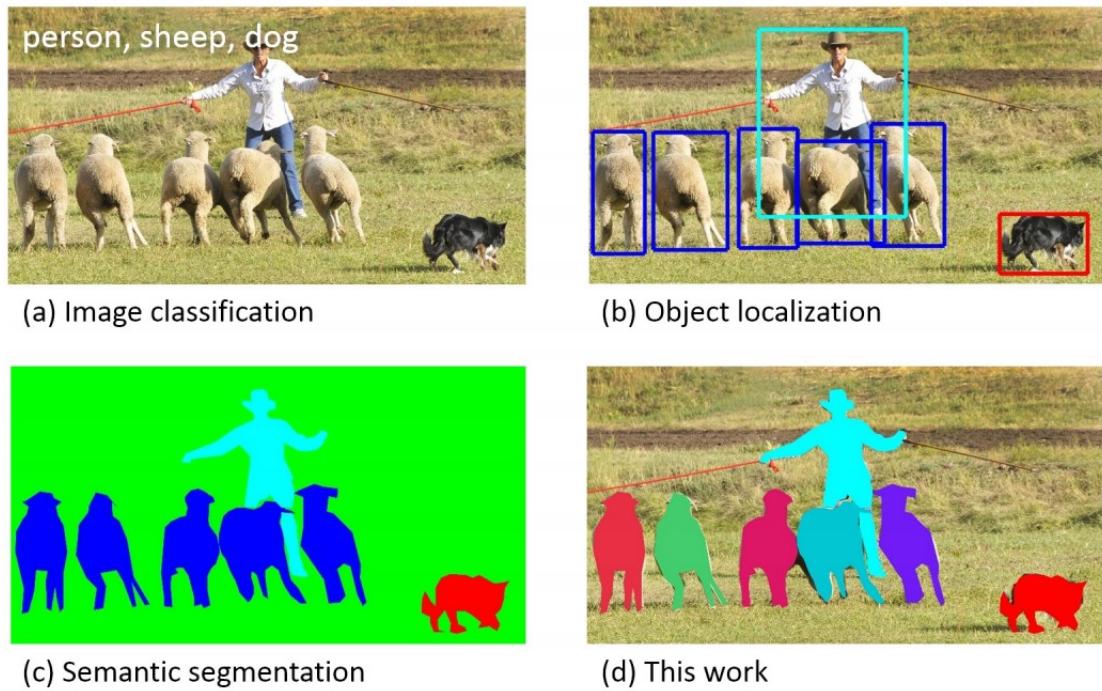


图 1.3: MS COCO 数据集多种图像任务示例

- NYUv2 数据集是使用 Kinect 采集的一系列包含深度信息的图像，包含如下几个部分：
 - 有标签的：视频数据的一个子集，伴随着密集多标签。此数据已被预处理以填补缺少的深度标签；
 - 原始数据集：利用 Kinect 测得的原始的 RGB、Depth、加速度数据；
 - 工具箱：用于操作数据和标签的有用的工具；
 - 用于评估的训练和测试部分。

有标签的数据如图1.4所示。

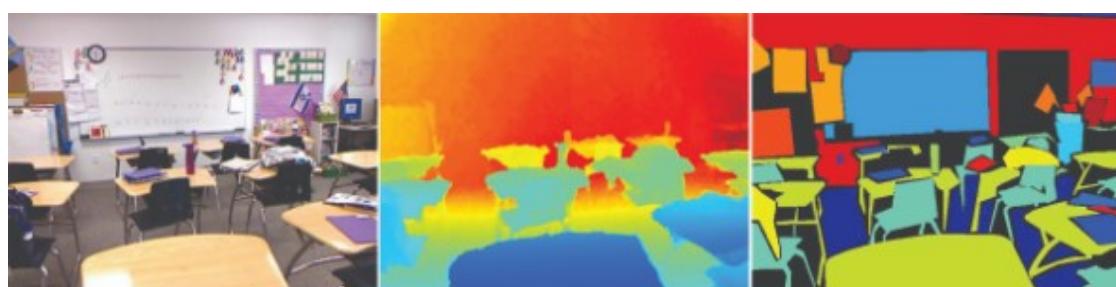


图 1.4: NYUv2 有标签的数据集示例。(左图: Kinect 相机输出的图像; 中图: 预处理深度信息; 右图: 添加一系列标签图)

1.3 综述结构

本综述主要包含如下内容：

- 第一章为引言部分，主要介绍了图像分割的评价指标以及图像分割方面的数据集；
- 第二章介绍了图像语义分割方向近几年来最经典的几种算法，并且在每一小节都与之前算法的实验结果进行了对比，从中我们可以直观的感受到图像分割实验效果的一步步提升；
- 第三章主要从 2018 年图像分割方向的论文出发，学习一下目前图像分割算法的研究方向；
- 第四章为对图像分割算法的总结以及对未来研究方向的展望。

第 2 章 基于深度学习的经典图像语义分割算法

在深度学习方法流行之前，TextonForest 和基于随机森林分类器等语义分割方法是用得比较多的方法。不过在深度卷积网络流行之后，深度学习方法比传统方法提升了很多，所以这里就不详细讲传统方法了。最初的学习方法应用于图像分割就是 Patch classification。Patch classification 方法，顾名思义，图像是切成块喂给深度模型的，然后对像素进行分类。使用图像块的主要原因是全连接层需要固定大小的图像。由于在全卷积网络出现之后对语义分割的效果提升了很多，因此在此也不再详述 Patch Classification 的方法。

2.1 全卷积网络 (FCN)

全卷积网络 (Fully convolutional networks, FCN^[4]) 于 2015 年被首次提出，并且获得了当年 CVPR 的 best paper。相比于之前使用带全连接层的卷积神经网络进行图像分割，FCN 主要涉及以下三个技术：

- 卷积化 (Convolutionalization);
- 上采样 (Upsampling)，也叫反卷积 (Deconvolution);
- 跳跃结构 (Skip Architecture).

2.1.1 卷积化

FCN 将传统 CNN 中的全连接层转化为一个个的卷积层。如图2.1所示，上图为传统的 CNN 网络结构 (AlexNet[5])，前 5 层是卷积层，第 6 层和第 7 层分别是一个长度为 4096 的全连接层，第 8 层是一个长度为 1000 的全连接层。在 FCN 中，将最后的三层全连接层全都替换为卷积层，卷积核大小分别为 (4096,1,1), (4096,1,1), (1000,1,1)。

网络结构如图2.2所示，虚线上半部分为全卷积网络 (蓝：卷积层，绿： max pooling)，输入可为任意尺寸的图像，下半部分为反卷积 (上采样) 结构，最后输出与原图像大小相同，通道数为 21(PASCAL VOC 数据集 20 类物体类别 +1 类背景)。

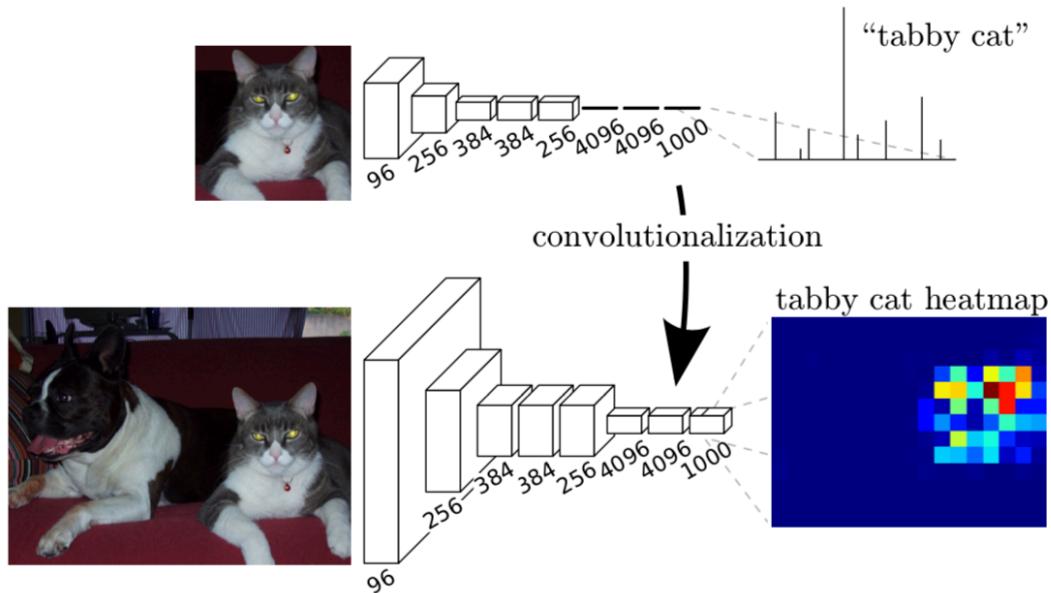


图 2.1: FCN 与传统 CNN 对比图

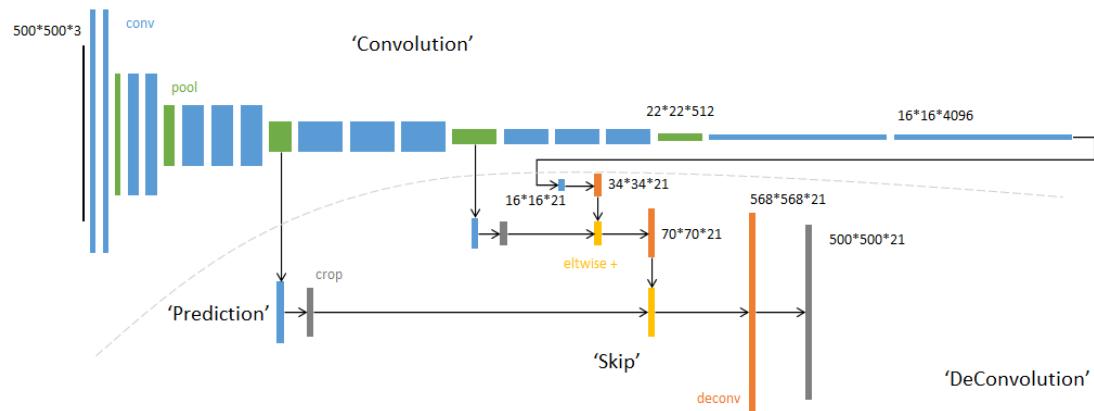


图 2.2: FCN 网络结构示意图

2.1.2 反卷积

经过全卷积网络之后得到的 feature map 相比于原图像要小，为了得到和原图像一样的 feature map 以便进行像素级的分类，FCN 采用反卷积的方式将最后一层的 feature map 进行放大(图2.2中虚线以下橙色的部分)。

图2.3展示了正常卷积与反卷积的对比图，左图展示的正常的 no padding no strides 情况下的卷积操作，可以看出，卷积之后 feature map 会变小，右图展示的是 no padding no strides 情况下的反卷积操作，可以看出反卷积之后 feature map 会增大。简单来看，反卷积其实可以看做先对 feature map 进行上采样增加像素，然后再进行卷积的过程，卷积的参数值通过训练得到。

https://github.com/vdumoulin/conv_arithmetic给出了各种类型卷积的动态图(正常卷积、反卷积以及之后要涉及的空洞卷积)，可以通过动态图进一步认识各

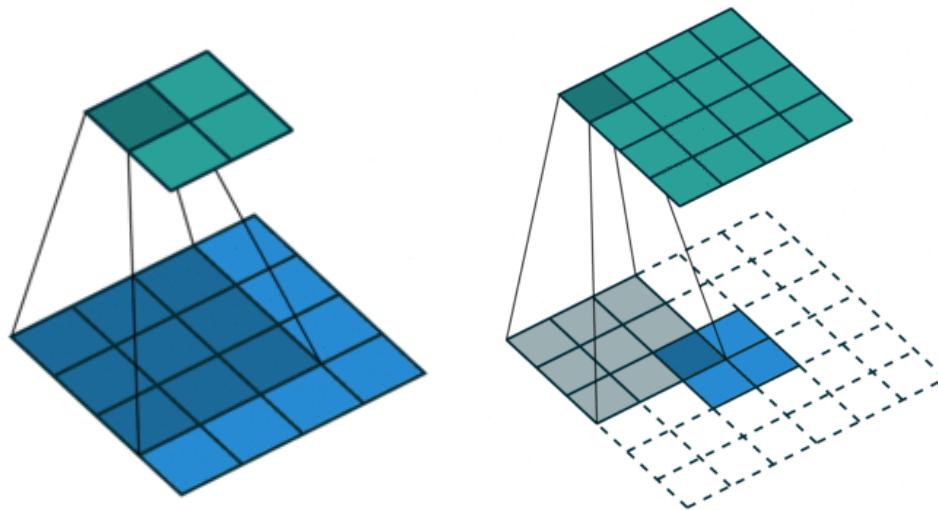


图 2.3：正常卷积与反卷积对比图

种卷积操作。论文 [6] 给出了详细的数学公式以及卷积前后 feature map 大小的变化公式，可以更进一步的认识卷积。

2.1.3 跳跃结构

如图2.4所示展示了 FCN 中跳跃结构示意图。从图中可以看出，对原图进行卷积 conv1、pool1 后图像缩小为 1/2；对图像进行第二次卷积 conv2、pool2 后图像缩小为 1/4；对图像进行第三次卷积 conv3、pool3 后图像缩小为 1/8，此时保留 pool3 的 featuremap；对图像进行第四次卷积 conv4、pool4 后图像缩小为 1/16，此时保留 pool4 的 featuremap；对图像进行第五次卷积 conv5、pool5 后图像缩小为 1/32，然后把原来 CNN 操作过程中的全连接编程卷积操作的 conv6、conv7，图像的 featuremap 的大小依然为原图的 1/32，此时图像不再叫 featuremap 而是叫 heatmap。其实直接使用前两种结构就已经可以得到结果了，这个上采样是通过反卷积 (deconvolution) 实现的，对第五层的输出 (32 倍放大) 反卷积到原图大小。但是得到的结果还上还不够精确，一些细节无法恢复。于是将第四层的输出和第三层的输出也依次反卷积，分别需要 16 倍和 8 倍上采样，结果过也更精细一些了。这种做法的好处是兼顾了 local 和 global 信息。

2.1.4 实验结果

在此篇论文中，作者分别使用了一次反卷积、两次反卷积以及三次反卷积操作进行实验，并且使用修改过的 VGG 网络结构进行训练，最后在 PASCAL VOC 数据集上的性能指标如图2.5所示。由结果图中可以直观的看出，FCN 的图像分割结果和 ground truth 相比已经有了较好的效果，但是边缘部分差距还是较大，在之后的几种算法中会有进一步的提升。

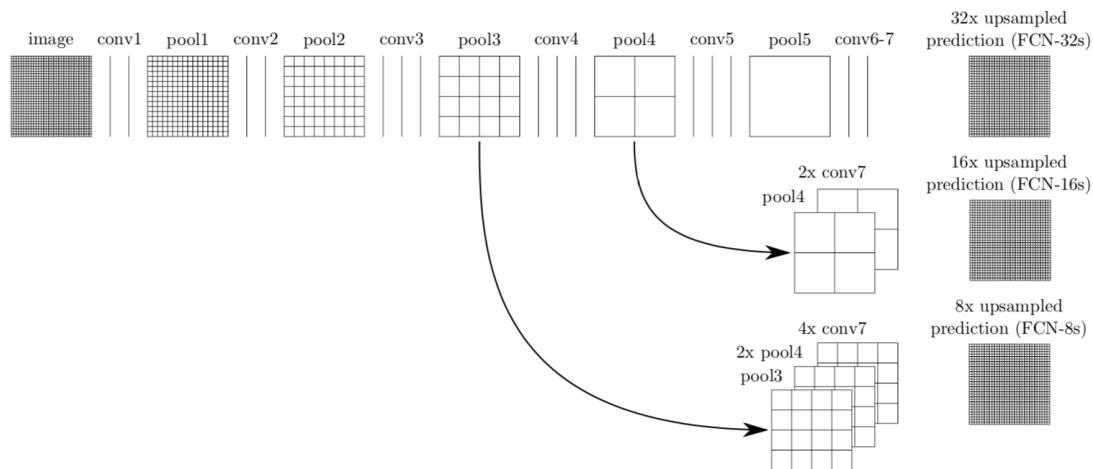


图 2.4: 跳跃结构示意图

| | pixel acc. | mean acc. | mean IU | f.w. IU |
|---------------|---------------|--------------|-------------|-------------|
| FCN-32s-fixed | 83.0 | 59.7 | 45.4 | 72.0 |
| FCN-32s | 89.1 | 73.3 | 59.4 | 81.4 |
| FCN-16s | 90.0 | 75.7 | 62.4 | 83.0 |
| FCN-8s | 90.3 | 75.9 | 62.7 | 83.2 |

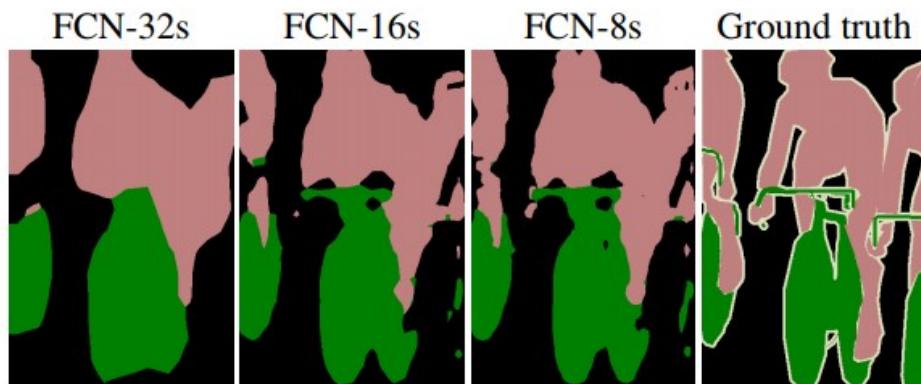


图 2.5: 实验结果性能指标与 PASCAL VOC 数据集上的实验效果图

2.2 DeepLab

从 FCN 的实验结果图中可以看出来，虽然分割的整体区域比较正确了，但是分割效果还是比较粗糙，细节不明显。DeepLab v1(文章 [7], 发表于 ICLR2015) 使用 Hole 算法 (Atrous Algorithm) 和条件随机场 (CRF) 来进一步的提升分割效果，其算法结构图如图2.6所示。DeepLab v1 收录于 ICLR 2015，是 DeepLab 系列的第一篇文章，之后我们还会介绍该系列的后续文章。

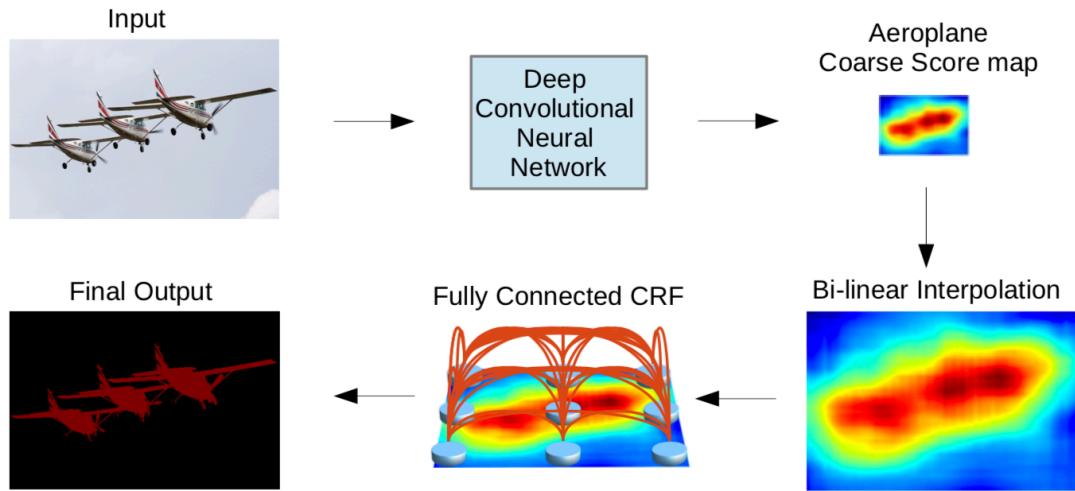


图 2.6: DeepLab 算法结构示意图

2.2.1 Hole 算法

由于普通的卷积感受野较小，需要增加池化层来增加感受野，但是池化层又会损失信息，所以使用空洞卷积在不损失信息的情况下增加感受野的范围。

Hole 算法又可以看做带孔 (Hole) 卷积，传统的卷积或者 pooling 中，一个 filter 中相邻的权重作用在 feature map 上的位置都是物理上连续的。而在 Hole 算法中，一个 filter 中相邻的权重不一定作用在 feature map 上的位置都是物理上连续的，而是跟 hole size 相关的。如图2.7所示表示的是卷积核大小 kernel_size=3，输入步长 input_stride(也就是 hole size)=2，输出步长 output_stride=1 的一维带孔卷积示意图。可以看出卷积核作用在输入 feature map 上的位置不是连续的。后续章节会有关于 Hole 算法的更详细的介绍，在此不再赘述。

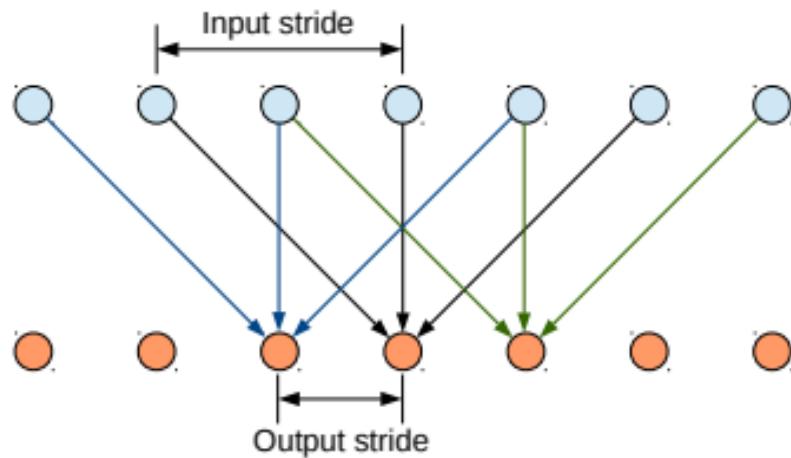


图 2.7: Hole 算法示意图

2.2.2 条件随机场

只使用全卷积网络能够预测到目标的大概位置但是位置比较模糊，论文 [8] 中提出的全连接条件随机场尝试找到图像像素之间的关系：相近且相似的像素大概率为同一标签，考虑像素的概率分配标签，通过迭代来细化分割的结果。

条件随机场服从吉布斯分布，如式 (2.1) 所示，其中 $E(X)$ 是 x 取某个值的能量， $Z(I)$ 是归一化的函数。

$$P(X|I) = \frac{1}{Z(I)} \exp(-E(X|I)) \quad (2.1)$$

为了做图像分割，只需要后验概率最大，因此只需能量函数最小即可，因此条件随机场优化的目标函数便是能量函数 $E(X)$ (式 (2.2)).

$$E(x) = \sum_i \psi_i(x_i) + \sum_{i,j} \psi_{i,j}(x_i, x_j) \quad (2.2)$$

能量方程的第一项 $\psi_i(x_i)$ (式 (2.3)) 称为一元势函数，用于衡量当像素点 i 的颜色值为 y_i 时，该像素点属于类别标签 x_i 的概率。在 DeepLab 中，此概率是通过 CNN 的输出得到的。

$$\psi_i(x_i) = -\log(P(x_i)) \quad (2.3)$$

能量方程的第二项 $\psi_{i,j}(x_i, x_j)$ 称之为成对势函数 (pairwise)，用于衡量两事件同时发生的概率 $p(x_i, x_j)$ ，我们希望两个相邻的像素点，如果颜色值 y_i, y_j 非常接近，那么这两个像素点 x_i, x_j 属于同一个类别的概率应该比较大才对；反之如果颜色差异比较大，那么我们分割的结果从这两个像素点裂开的概率应该比较大才对。这一能量项正是为了让我们的分割结果尽量从图像边缘的地方裂开，也就是为了弥补之前 FCN 边缘的地方分割的不足，我们可以采用式 (2.4) 来计算。

$$\psi_{i,j}(x_i, x_j) = u(x_i, x_j) \sum_{m=1}^M w^m K_G^m(f_i, f_j) \quad (2.4)$$

其中 K_G 是一个高斯核，用于度量像素点 i 和 j 的特征向量相似度的一个高斯权重项。特征向量 f_i 我们可以用 (x, y, R, G, B) 表示，也就是以像素点的像素值和坐标位置作为特征向量。然后 $u(x_i, x_j)$ 表示两个标签之间的一个兼容性度量。通过最小化式 (2.2) 的能量函数，我们就可以实现 CRF 隐变量 X 的推理。

图2.8显示了 DeepLab 中使用 CRF 迭代来细化分割结果的示意图，从图中可以看出，使用 CRF 迭代可以使得分割的边缘效果逐渐增强。

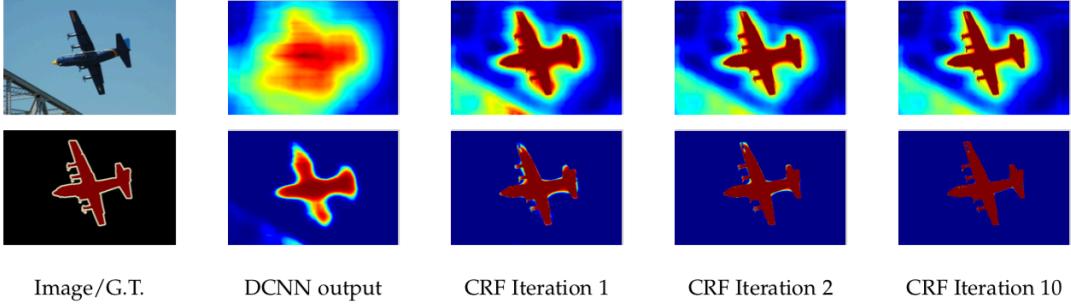


图 2.8: 使用 CRF 细化分割效果。可以看出，随着迭代次数的增加，图像分割的效果逐渐增强

2.2.3 实验结果

DeepLab 综合了之前出现的全卷积网络 FCN，提出了带洞卷积，在此我们列出与 FCN 算法在 PASCAL VOC 数据集上实验效果对比如图2.9所示。从图中可以看出，DeepLab 在边缘部分的分割效果相比于 FCN 有了进一步的提升。

2.3 DilatedConv

论文 Multi-Scale Context Aggregation by Dilated Convolutions[9] 是收录于 ICLR 2016 的一篇文章，提出了一种卷积网络模块，可以聚合多尺度的上下文信息，而不会丢失分辨率或重复分析重缩放的图像。该模块基于空洞卷积，其支持指数级扩展感受野而不损失分辨率或覆盖范围。

2.3.1 空洞卷积

与上一节 DeepLab 中提出的带孔卷积类似，主要目的是增加感受野。传统卷积操作增加感受野的速度是很慢的，所以一般会在卷积层之后增加池化层进一步增加感受野，但是增加池化层之后会使得图像进行了下采样，使得图像丢失了很多细节信息。

设 $F : \mathbb{Z}^2 \rightarrow \mathbb{R}$ 是一个离散函数，令 $\Omega_r = [-r, r]^2 \cap \mathbb{Z}^2$ ，再设 $k : \Omega_r \rightarrow \mathbb{R}$ 是一个大小为 $(2r + 1)^2$ 的离散滤波器。因此传统的离散卷积操作可以 * 描述为：

$$(F * k)(\mathbf{p}) = \sum_{\mathbf{s}+\mathbf{t}=\mathbf{p}} F(\mathbf{s})k(\mathbf{t}) \quad (2.5)$$

对其进行拓展，我们可以定义膨胀系数为 l 的空洞卷积运算 $*_l$ 如下：

$$(F *_l k)(\mathbf{p}) = \sum_{\mathbf{s}+l\mathbf{t}=\mathbf{p}} F(\mathbf{s})k(\mathbf{t}) \quad (2.6)$$

| Method | mean IOU (%) |
|--------------------------|--------------|
| MSRA-CFM | 61.8 |
| FCN-8s | 62.2 |
| TTI-Zoomout-16 | 64.4 |
| DeepLab-CRF | 66.4 |
| DeepLab-MSc-CRF | 67.1 |
| DeepLab-CRF-7x7 | 70.3 |
| DeepLab-CRF-LargeFOV | 70.3 |
| DeepLab-MSc-CRF-LargeFOV | 71.6 |

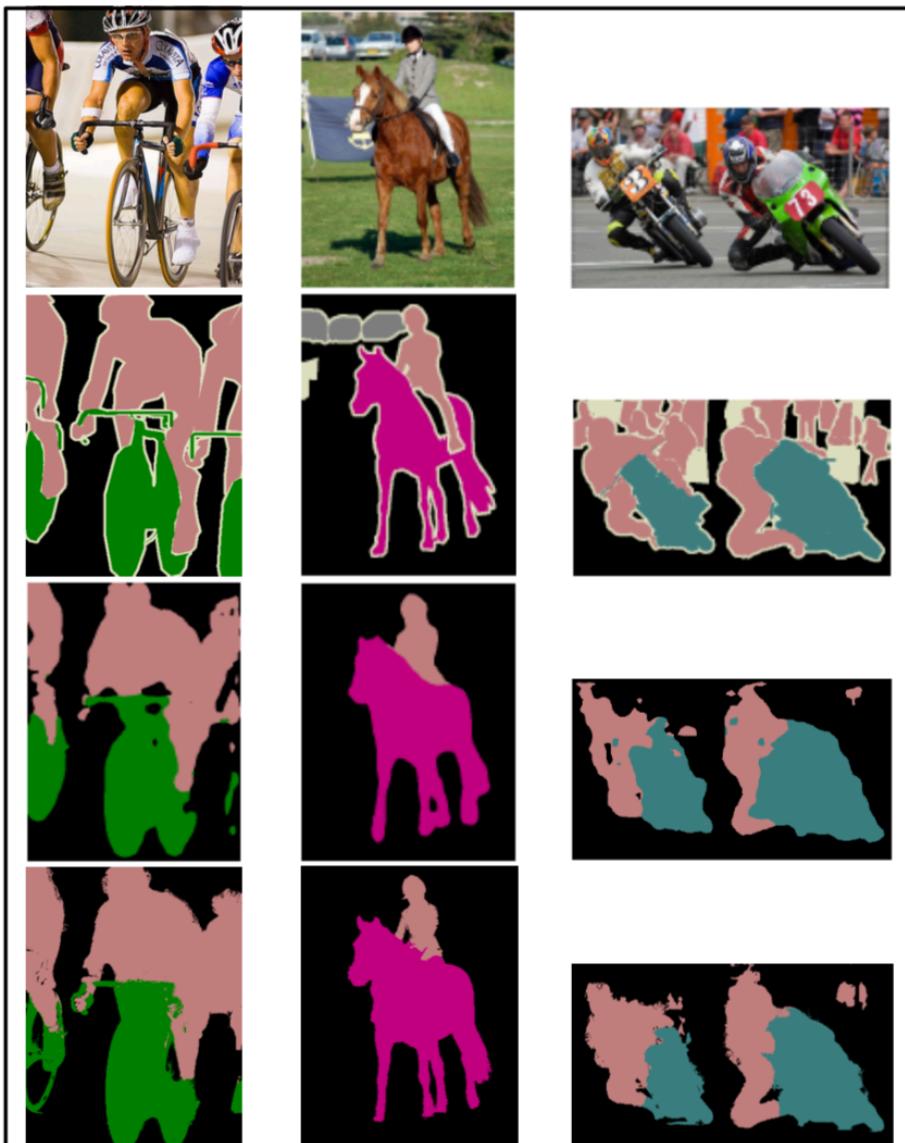


图 2.9: DeepLab 与 FCN 在 PASCAL VOC 数据集上的实验结果对比图。上图为测试指标的对比示意图。下图为实验效果对比图：第一行为原始图像，第二行为 ground truth，第三行为 FCN 分割效果图，第四行为 DeepLab 分割效果图

如图2.10所示展示了膨胀系数为 2 的空洞卷积操作的示意图，与传统卷积相比，可以看出空洞卷积的卷积核作用于 feature map 上的物理位置是不连续的。

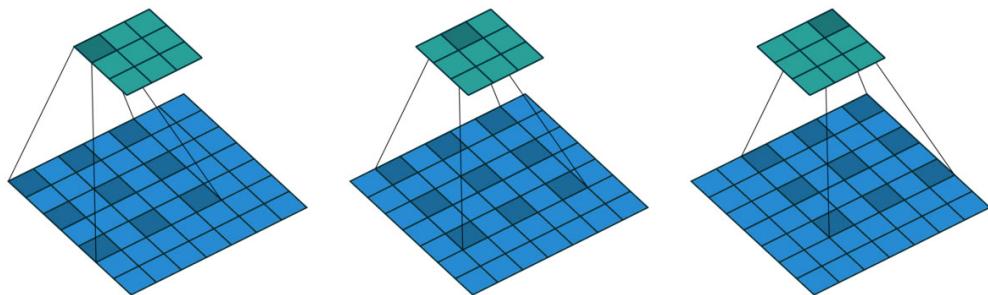


图 2.10: 空洞卷积操作示意图

图2.11展示了空洞卷积带来感受野变化的示意图。(a) 表示普通的卷积，卷积核为 3×3 ，感受野也为 3×3 ，较小;(b) 表示扩张系数为 2 的空洞卷积，卷积核为 3×3 ，但是感受野有 7×7 ，比之前大了一点;(c) 表示扩张系数为 4 的空洞卷积，卷积核为 3×3 ，但是感受野有 15×15 ，更大了。

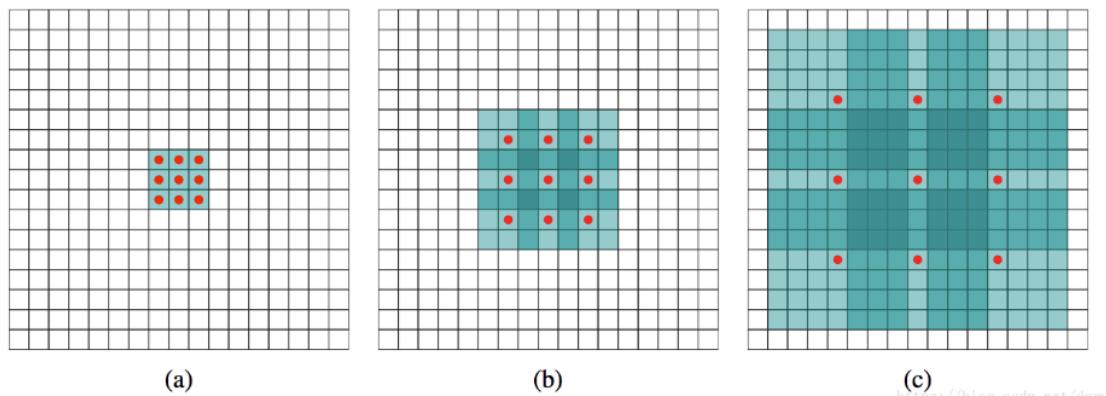


图 2.11: 空洞卷积感受野变化示意图

<https://blog.csdn.net/dcr>

2.3.2 网络结构

DilatedConv 的网络结构分为两个模块：front-end 模块和 Context 模块。front-end 模块是在 VGG^[10] 的基础上进行修改得到的，该模块的作用是为了产生 feature map 为 context 模块作用。图2.12展示了 DilatedConv 网络结构和原始 VGG 网络结构的对比。主要进行了如下几点修改：

- 与全卷积网络 FCN 一样，DilatedConv 也将 VGG 中的全连接层修改为卷积层；
- 去掉了最后两个池化层；
- 将后续的卷积修改为带孔卷积；
- 使用 ImageNet 上预训练的参数进行 Finetune。

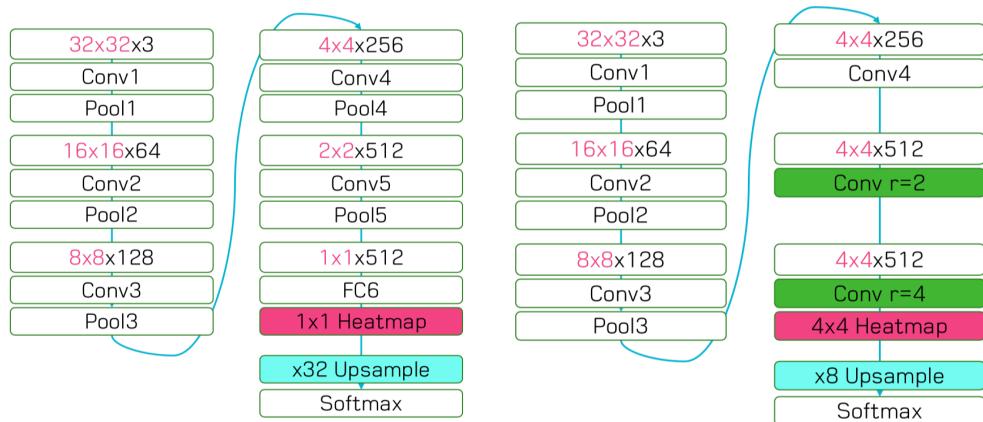


图 2.12: VGG 网络结构示意图和修改之后 DilatedConv 网络结构实体图的对比图。左图为 VGG 网络的原始结构示意图，右图为修改之后的 DilatedConv 网络结构示意图

Context 模块采用 front-end 模块输出的 feature map 作为输入，使用空洞卷积组成 8 层网络结构，front-end 模块和 Context 模块连接示意图如图2.13所示，左图表示 front-end 模块，可以看出最后输出 $64 \times 64 \times C$ 的 feature map，右图表示 Context 模块，输入为 $64 \times 64 \times C$ 的 feature map。

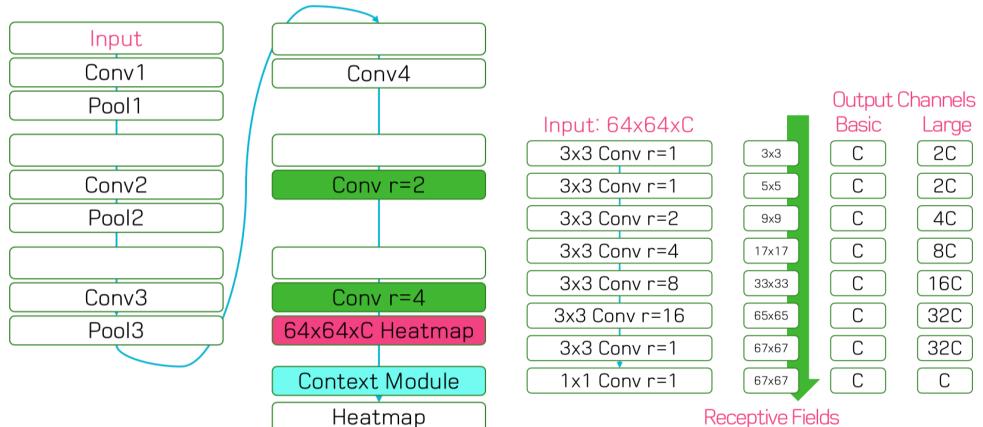


图 2.13: DilatedConv 网络 front-end 模块和 Context 模块示意图。

2.3.3 实验结果

在此，我们将 DilatedConv 算法的结果与之前提到的 FCN 和 DeepLab 进行对比，如图2.14所示，上图展示了三种算法在 PascalVOC 数据集上 meanIoU 的对比情况，可以看出，Dilatedconv 算法相比于 FCN 和 DeepLab 算法有了进一步的提升，下图表示分割效果的直观对比图，从中我们也能较为清楚的看出，Dilatedconv 算法的分割效果有了更进一步的提升。

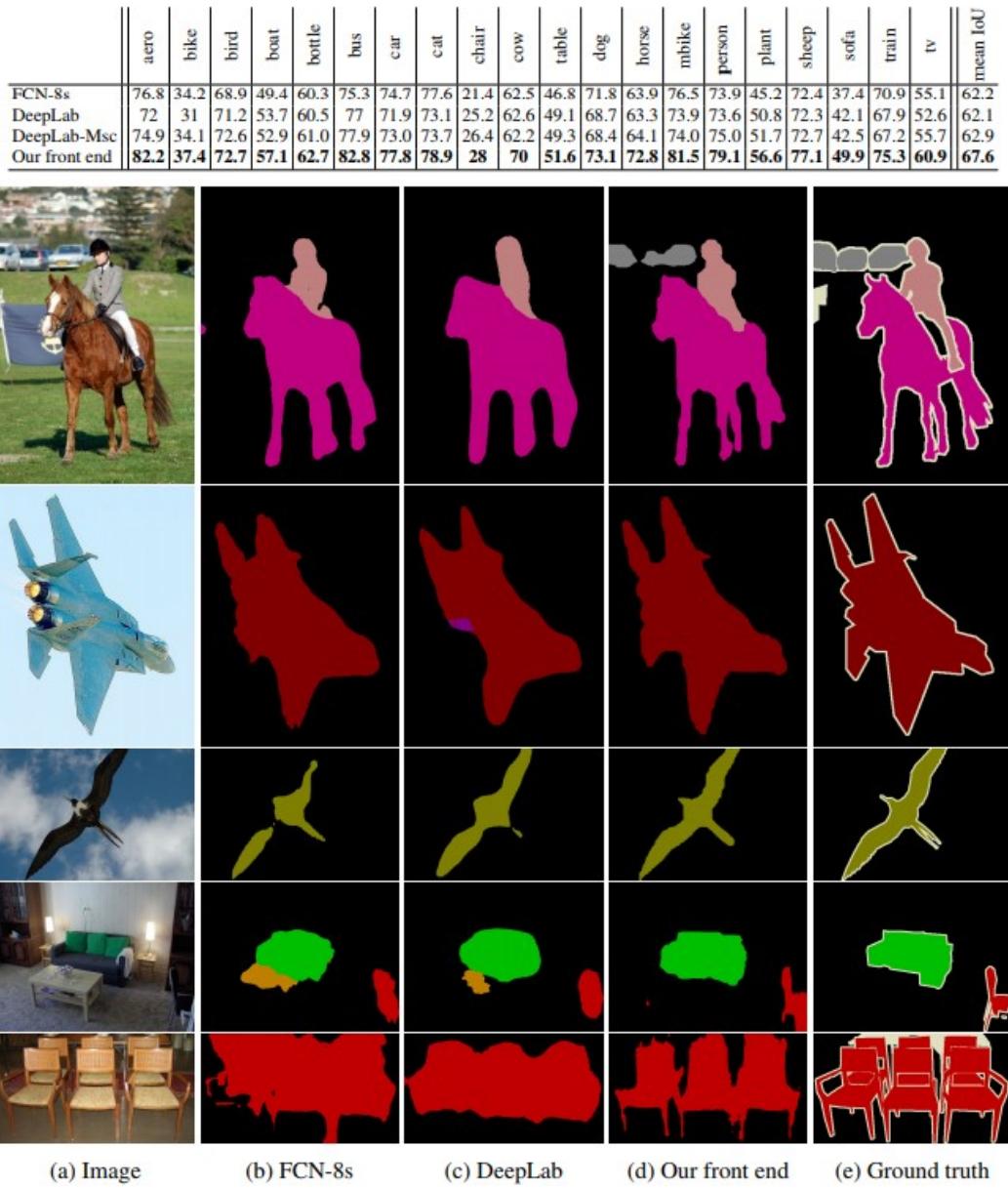


图 2.14: DilatedConv 算法与 FCN、DeepLab 算法结果对比图

2.4 DeepLab v2

DeepLab v2(DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs)^[11] 是 DeepLab 系列的第二篇文章，最早版本于 2016 年 6 月在 arXiv 上发表，被 TPAMI 2017 收录。他依然使用了之前提到的全卷积网络、空洞卷积、条件随机场等算法，相比于 DeepLab v1，他提出了空洞空间卷积池化金字塔 (atrous spatial pyramid pooling (ASPP))，以多尺度的信息得到更强健的分割结果。ASPP 并行的采用多个采样率的空洞卷积层来探测，以多个比例捕捉对象以及图像上下文。除此之外，DeepLab v2 基础网络结构由 VGG16 转为 ResNet，使用了不同的学习策略，并在使用在 MS-COCO 数

据集上预训练的参数进行 finetune。图2.15展示了 DeepLab v2 算法的结构。

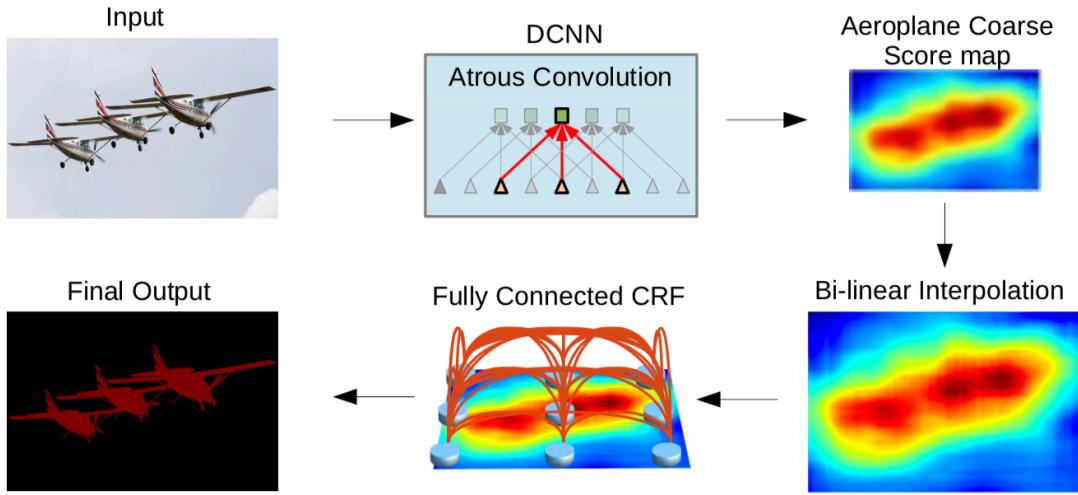


图 2.15: DeepLab v2 算法结构示意图

2.4.1 ASPP 模块

受 SPPNet 中 SPP 模块的启发，它指出在任意尺度的区域，可以用从单个尺度图像中进行重采样提取的卷积特征进行准确有效地分类。我们用不同采样率的多个并行的空洞卷积实现了他们的方案的一个变体。并行的采用多个采样率的空洞卷积提取特征，再将特征融合，类似于空间金字塔结构。图2.16和图2.17展示了 ASPP 模块和 ASPP 模块在网络中结构的示意图。

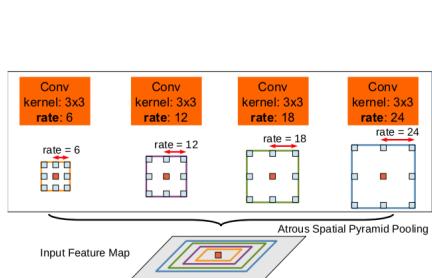


图 2.16: ASPP 模块示意图

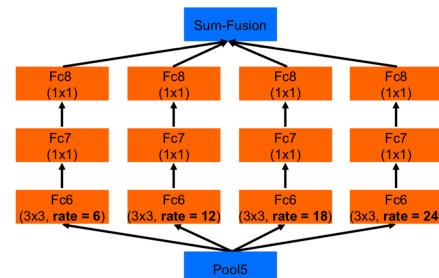


图 2.17: ASPP 结构示意图

2.4.2 实验结果

DeepLab v2 融合了之前算法所提到的模块并且加入了新的 ASPP 模块，使得分割效果有了进一步的提升。图2.18展示了 DeepLab v2 算法与之前出现的图像分割算法在 PASCAL VOC2012 测试集上 mIoU 的对照表格，从中我们可以看出来，DeepLab v2 算法的 mIoU 值相比于之前的算法都高，因此，其分割效果也得到了进一步的提升。

| Method | mIOU |
|---------------------------------|------|
| DeepLab-CRF-LargeFOV-COCO [58] | 72.7 |
| MERL_DEEP_GCRF [88] | 73.2 |
| CRF-RNN [59] | 74.7 |
| POSTECH_DeconvNet_CRF_VOC [61] | 74.8 |
| BoxSup [60] | 75.2 |
| Context + CRF-RNN [76] | 75.3 |
| QO_4^{mres} [66] | 75.5 |
| DeepLab-CRF-Attention [17] | 75.7 |
| CentraleSuperBoundaries++ [18] | 76.0 |
| DeepLab-CRF-Attention-DT [63] | 76.3 |
| H-ReNet + DenseCRF [89] | 76.8 |
| LRR_4x_COCO [90] | 76.8 |
| DPN [62] | 77.5 |
| Adelaide_Context [40] | 77.8 |
| Oxford_TVGV_HO_CRF [91] | 77.9 |
| Context CRF + Guidance CRF [92] | 78.1 |
| Adelaide_VeryDeep_FCN_VOC [93] | 79.1 |
| DeepLab-CRF (ResNet-101) | 79.7 |

图 2.18: DeepLab v2 算法与之前出现的图像分割算法在 PASCAL VOC2012 测试集上 mIoU 的对照表

2.5 PSPNet

论文 Pyramid Scene Parsing Network^[12] 是 CVPR2017 收录的关于场景解析的文章，拿下了 2016 年的 ImageNet 比赛中 scene parsing 任务的冠军，也可以用于做图像语义分割。这篇文章出发点是在语义分割算法中引入更多的上下文信息 (context information)，这样能够避免许多误分割，PSPNet 在 FCN 算法的基础上引入更多上下文信息是通过全局均值池化操作 (global average pooling) 和特征融合实现的，因此特征呈金字塔结构，这也是论文名叫 pyramid 的原因。

2.5.1 算法结构

图2.19展示的 PSPNet 算法结构示意图。首先输入图像经过一个特征提取网络提取特征，这部分作者采用的是添加了空洞卷积的 ResNet 网络，提取到的特征（具体而言 stride=8）作为后面 pyramid pooling 模块的输入。在 pyramid pooling 模块中构建了深度为 4 的特征金字塔，不同深度的特征是基于输入特征通过不同尺度的池化操作得到的，池化的尺度是可以调整的，这篇文章中给出的池化后的特征尺寸分别是 1×1 、 2×2 、 3×3 和 6×6 。然后通过一个 1×1 卷积层将特征维度缩减为原来的 $1/4$ ，最后将这些金字塔特征直接上采样到与输入特征相同尺

寸，然后和输入特征做合并，也就是 concat 操作得到最终输出的特征图。

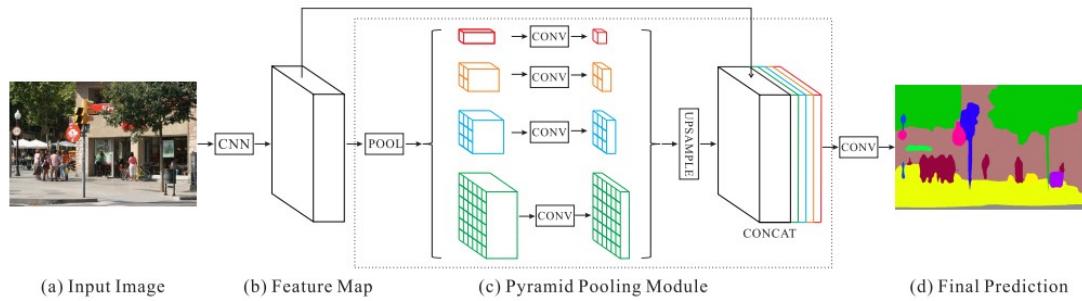


图 2.19: PSPNet 算法结构示意图

2.5.2 实验结果

图2.20展示 PSPNet 和 FCN 在数据集 ADE20k 上的实验效果对比，从中可以看出，FCN 的分割效果存在着很多的误分割，而 PSPNet 减少了很多误分割。第一行中 FCN 算法误将船分割成车，显然一辆车在水上的概率是很小的，这种是属于明显不匹配的误分割。第二行中 FCN 算法误将摩天大厦分割成建筑物，摩天大厦和建筑物这两个类别本身是比较接近的，这种是属于类别相近的误分割，这部分个人认为是和数据集相关的。第三行中 FCN 算法误将枕头分割成床，枕头本身区域较小，而且纹理和床较为接近，这种是属于难以觉察的误分割。作者认为这些误分割都可以通过引入更多的上下文信息进行解决，当分割层有更多全局信息时，出现上述几种误分割的概率就会相对低一些，这种思想目前在许多图像领域都有所应用，而引入更多上下文信息的方式也很多，比如：1、增大分隔层的感受野，这种方式是最直观的，视野越广，看到的东西也越多，而增大感受野也有许多方式，比如空洞卷积（dilated convolution），这是在 deeplab 算法上成功应用的实现方式，另外 PSPNet 的全局均值池化操作也是增加感受野的一种方式。2、深层特征和浅层特征的融合，增加浅层特征的语义信息，这样在浅层进行分割时就有足够的上下文信息，同时也有目标的细节信息，这种做法早在 FCN 中就有了，但是包括融合策略和分割层的选择都有一定的优化空间。

图2.21展示了 PSPNet 和图像分割其他算法在 PASCAL VOC 2012 测试集上结果比较，表格下半部分表示使用了 MS-COCO 数据集上预训练的参数进行 finetune 的结果，可以看出，PSPNet 相比于之前出现的算法分割效果的指标有了进一步的提升。

2.6 DeepLab v3

DeepLab v3(Rethinking Atrous Convolution for Semantic Image Segmentation)^[13] 是 DeepLab 系列的第三篇文章，最早在 arXiv 上发布于 2017 年 6 月。相比于之

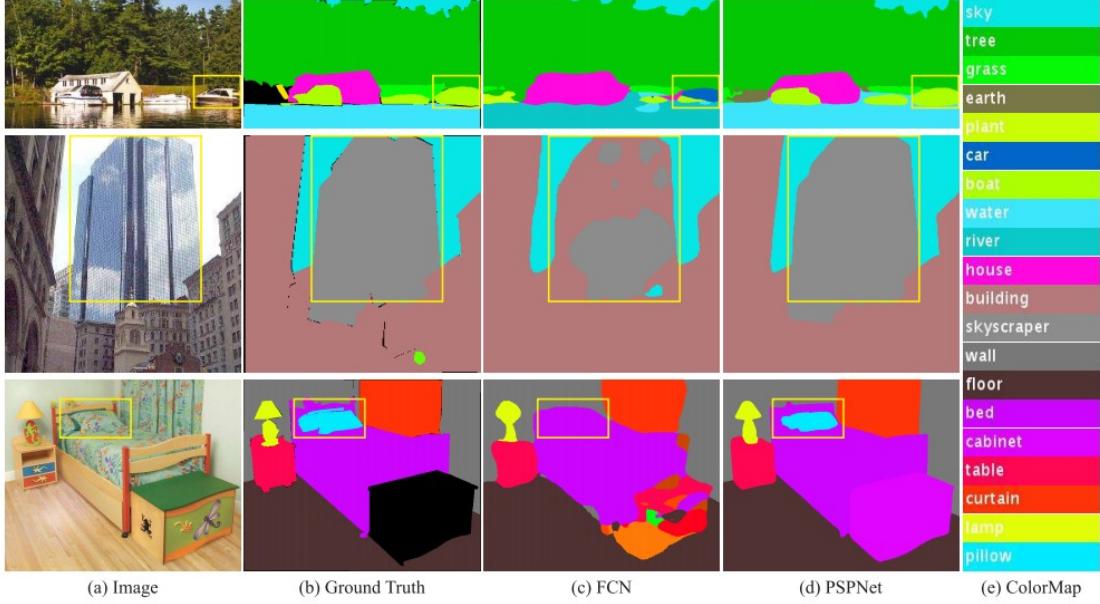


图 2.20: PSPNet 和 FCN 在数据集 ADE20k 上的实验效果对比

| Method | aero | bike | bird | boat | bottle | bus | car | cat | chair | cow | table | dog | horse | mbike | person | plant | sheep | sofa | train | tv | mIoU |
|-----------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| FCN [26] | 76.8 | 34.2 | 68.9 | 49.4 | 60.3 | 75.3 | 74.7 | 77.6 | 21.4 | 62.5 | 46.8 | 71.8 | 63.9 | 76.5 | 73.9 | 45.2 | 72.4 | 37.4 | 70.9 | 55.1 | 62.2 |
| Zoom-out [28] | 85.6 | 37.3 | 83.2 | 62.5 | 66.0 | 85.1 | 80.7 | 84.9 | 27.2 | 73.2 | 57.5 | 78.1 | 79.2 | 81.1 | 77.1 | 53.6 | 74.0 | 49.2 | 71.7 | 63.3 | 69.6 |
| DeepLab [3] | 84.4 | 54.5 | 81.5 | 63.6 | 65.9 | 85.1 | 79.1 | 83.4 | 30.7 | 74.1 | 59.8 | 79.0 | 76.1 | 83.2 | 80.8 | 59.7 | 82.2 | 50.4 | 73.1 | 63.7 | 71.6 |
| CRF-RNN [41] | 87.5 | 39.0 | 79.7 | 64.2 | 68.3 | 87.6 | 80.8 | 84.4 | 30.4 | 78.2 | 60.4 | 80.5 | 77.8 | 83.1 | 80.6 | 59.5 | 82.8 | 47.8 | 78.3 | 67.1 | 72.0 |
| DeconvNet [30] | 89.9 | 39.3 | 79.7 | 63.9 | 68.2 | 87.4 | 81.2 | 86.1 | 28.5 | 77.0 | 62.0 | 79.0 | 80.3 | 83.6 | 80.2 | 58.8 | 83.4 | 54.3 | 80.7 | 65.0 | 72.5 |
| GCRF [36] | 85.2 | 43.9 | 83.3 | 65.2 | 68.3 | 89.0 | 82.7 | 85.3 | 31.1 | 79.5 | 63.3 | 80.5 | 79.3 | 85.5 | 81.0 | 60.5 | 85.5 | 52.0 | 77.3 | 65.1 | 73.2 |
| DPN [25] | 87.7 | 59.4 | 78.4 | 64.9 | 70.3 | 89.3 | 83.5 | 86.1 | 31.7 | 79.9 | 62.6 | 81.9 | 80.0 | 83.5 | 82.3 | 60.5 | 83.2 | 53.4 | 77.9 | 65.0 | 74.1 |
| Piecewise [20] | 90.6 | 37.6 | 80.0 | 67.8 | 74.4 | 92.0 | 85.2 | 86.2 | 39.1 | 81.2 | 58.9 | 83.8 | 83.9 | 84.3 | 84.8 | 62.1 | 83.2 | 58.2 | 80.8 | 72.3 | 75.3 |
| PSPNet | 91.8 | 71.9 | 94.7 | 71.2 | 75.8 | 95.2 | 89.9 | 95.9 | 39.3 | 90.7 | 71.7 | 90.5 | 94.5 | 88.8 | 89.6 | 72.8 | 89.6 | 64.0 | 85.1 | 76.3 | 82.6 |
| CRF-RNN [†] [41] | 90.4 | 55.3 | 88.7 | 68.4 | 69.8 | 88.3 | 82.4 | 85.1 | 32.6 | 78.5 | 64.4 | 79.6 | 81.9 | 86.4 | 81.8 | 58.6 | 82.4 | 53.5 | 77.4 | 70.1 | 74.7 |
| BoxSup [†] [7] | 89.8 | 38.0 | 89.2 | 68.9 | 68.0 | 89.6 | 83.0 | 87.7 | 34.4 | 83.6 | 67.1 | 81.5 | 83.7 | 85.2 | 83.5 | 58.6 | 84.9 | 55.8 | 81.2 | 70.7 | 75.2 |
| Dilation8 [†] [40] | 91.7 | 39.6 | 87.8 | 63.1 | 71.8 | 89.7 | 82.9 | 89.8 | 37.2 | 84.0 | 63.0 | 83.3 | 89.0 | 83.8 | 85.1 | 56.8 | 87.6 | 56.0 | 80.2 | 64.7 | 75.3 |
| DPN [†] [25] | 89.0 | 61.6 | 87.7 | 66.8 | 74.7 | 91.2 | 84.3 | 87.6 | 36.5 | 86.3 | 66.1 | 84.4 | 87.8 | 85.6 | 85.4 | 63.6 | 87.3 | 61.3 | 79.4 | 66.4 | 77.5 |
| Piecewise [†] [20] | 94.1 | 40.7 | 84.1 | 67.8 | 75.9 | 93.4 | 84.3 | 88.4 | 42.5 | 86.4 | 64.7 | 85.4 | 89.0 | 85.8 | 86.0 | 67.5 | 90.2 | 63.8 | 80.9 | 73.0 | 78.0 |
| FCRNs [†] [38] | 91.9 | 48.1 | 93.4 | 69.3 | 75.5 | 94.2 | 87.5 | 92.8 | 36.7 | 86.9 | 65.2 | 89.1 | 90.2 | 86.5 | 87.2 | 64.6 | 90.1 | 59.7 | 85.5 | 72.7 | 79.1 |
| LRR [†] [9] | 92.4 | 45.1 | 94.6 | 65.2 | 75.8 | 95.1 | 89.1 | 92.3 | 39.0 | 85.7 | 70.4 | 88.6 | 89.4 | 88.6 | 86.6 | 65.8 | 86.2 | 57.4 | 85.7 | 77.3 | 79.3 |
| DeepLab [†] [4] | 92.6 | 60.4 | 91.6 | 63.4 | 76.3 | 95.0 | 88.4 | 92.6 | 32.7 | 88.5 | 67.6 | 89.6 | 92.1 | 87.0 | 87.4 | 63.3 | 88.3 | 60.0 | 86.8 | 74.5 | 79.7 |
| PSPNet [†] | 95.8 | 72.7 | 95.0 | 78.9 | 84.4 | 94.7 | 92.0 | 95.7 | 43.1 | 91.0 | 80.3 | 91.3 | 96.3 | 92.3 | 90.1 | 71.5 | 94.4 | 66.9 | 88.8 | 82.0 | 85.4 |

图 2.21: PSPNet 和图像分割其他算法在 PASCAL VOC 2012 测试集上结果比较

前的算法，主要有以下几点改进：

- 提出了更通用的框架，适用于任何网络；
- 复制了 ResNet 最后的 block 并且级联起来；
- 在 ASPP 模块中使用 BN 层，并且有了新的 ASPP 结构；
- 取消了后续的 DenseCRF，减少了训练时间

2.6.1 级联模块

如图2.22所示，复制 ResNet 最后一个 block 的多个副本，并且级联起来，图2.22中的 block5-7 是 block4 的副本。每个 block 中包含三个卷积 (MultiGrid)，每个 block 中最后一个卷积的步长为 2(最后一个 block 除外)，为了维持原图尺寸，使用不同采样率的空洞卷积来代替原来的卷积。

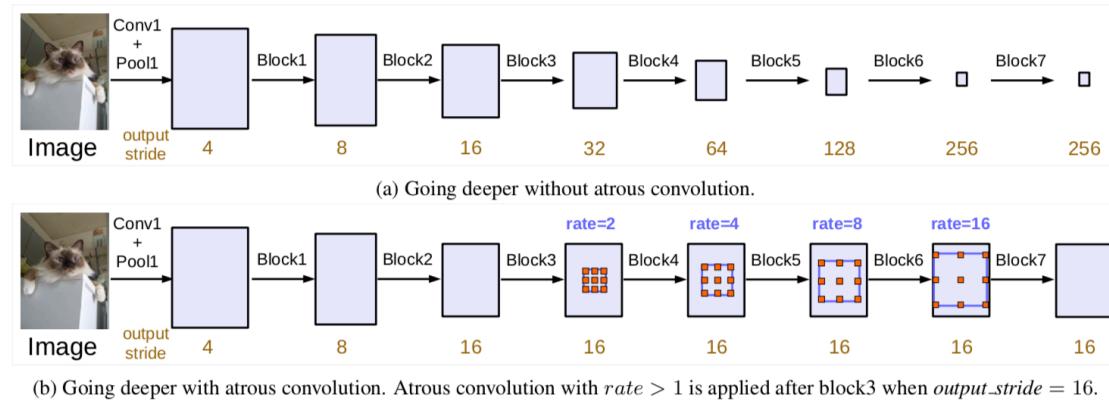


图 2.22: 级联模块使用空洞卷积和不使用空洞卷积的示意图

2.6.2 新的 ASPP 模块

相比于原 ASPP 模块有如下改变：

- ASPP 中应用了 BN 层；
- 随着采样率的增加，卷积核中有效的权重减小了；
- 使用模型最后的 feature map 做全局平均池化；
- 包括一个 1×1 的卷积和 3 个 3×3 采样率分别为 (6,12,18) 的空洞卷积，并且每个卷积都有 BN 层和一个全局平均池化层；
- 所有的分支通过 1×1 的卷积级联起来。

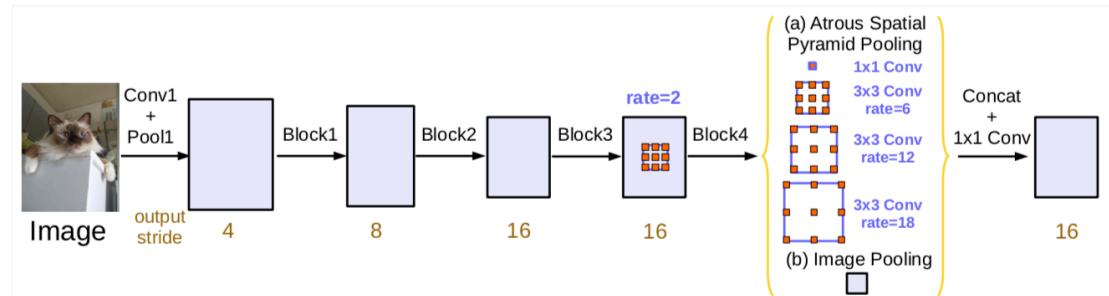


图 2.23: 新的 ASPP 模块

2.6.3 实验结果

图2.24展示了DeepLab v3 和之前算法在PASCAL VOC 2012 测试集上图像分割性能指标的对比表格，可以看出，DeepLab v3 相比于之前的算法，分割的效果有了更进一步的提升，同时，由于其并没有使用CRF，所以训练时间也比DeepLab v1 和 v2 少了很多，因此，DeepLab v3 效果的提升是比较大的。图2.25展示了DeepLab v3 在 PASCAL VOC 数据集上分割效果示意图，可以看出分割的效果相比于最初的FCN已经有了很大的提升，虽然没有使用CRF，但是细节部分分割效果也非常好。

| | PASCAL VOC 2012 | Cityscapes (IoU / iIoU) | Contribution |
|-----------------------|-----------------|-----------------------------------|---|
| FCN-8s-CVPR15 | 62.2% | 65.3% / 41.7% | FCN |
| FCN-8s-PAMI17 | 67.2% | | |
| DeepLab v1 | 71.6% | 63.1% / 34.5% | |
| CRF-RNN | 72.0% | 62.5% / 34.4% | CRF (End-to-End) |
| Dilated Conv FrontEnd | 71.3% | 10-Layer Context 67.1% / 42.0% | Cascade Dilated |
| Dilated Conv Context | 73.5% | | |
| Dilated Conv+ CRFRNN | 75.3% | | |
| DeepLab v2 | 79.7% | 70.4% / 42.6% | Dilated+ASPP+CRFs+ResNet |
| PSPNet | 85.4% | 81.2% / 59.6% | Pyramid Pooling + Aux. Loss |
| DeepLab v3 | 85.7% | 81.3% / 62.1% | Modified Layer & ASPP + BatchNorm + Traning Strategies |
| DeepLab v3-JFT | 86.9% | | |

图 2.24: PASCAL VOC 2012 实验结果对比

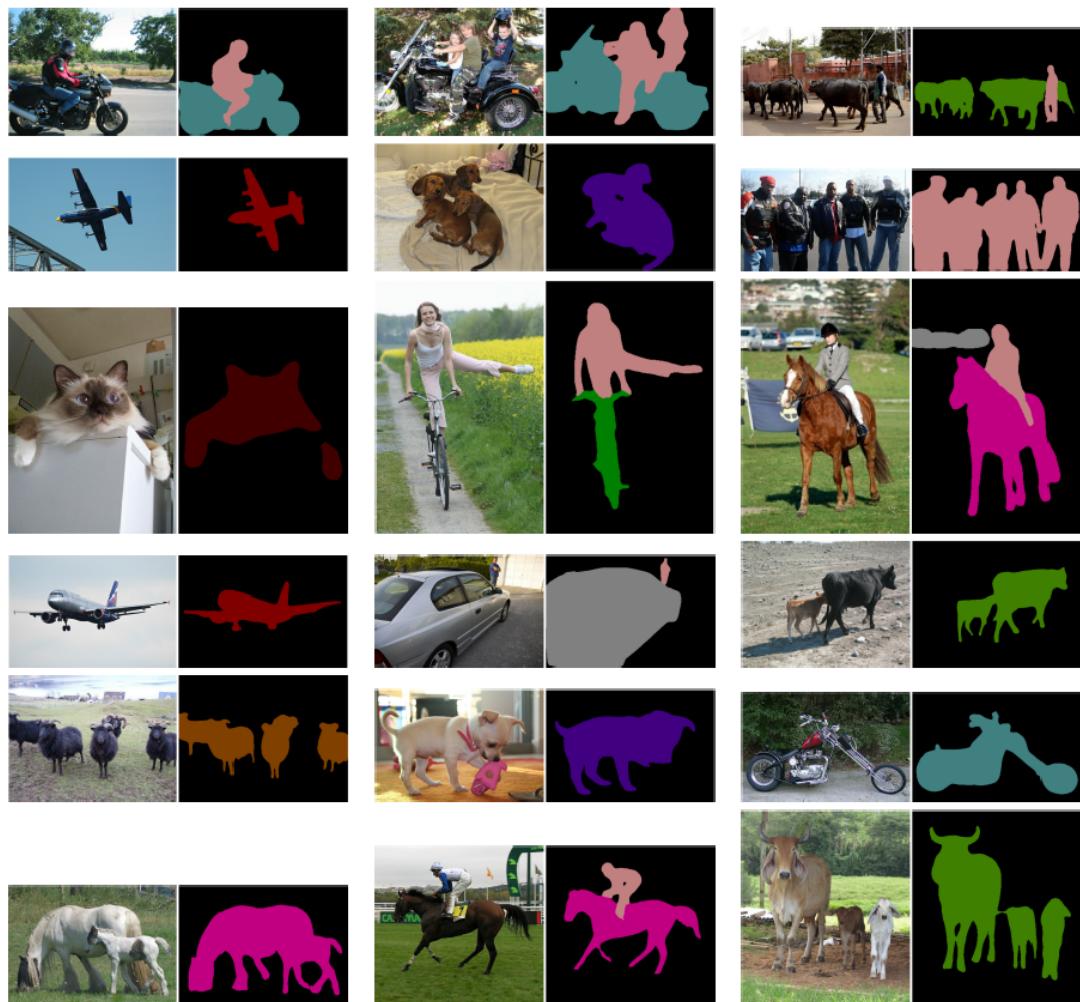


图 2.25: DeepLab v3 在 PASCAL VOC 数据集上分割效果示意图

第3章 最新图像语义分割算法

上一章主要研究了近几年基于深度学习的图像语义分割算法，并对实验结果进行了对比。在本章主要研究2018年图像分割领域的CVPR、ECCV的文章，调研一下目前图像分割领域的前沿方向。

3.1 Depth-aware CNN

Depth-aware CNN for RGB-D Segmentation^[14]是ECCV 2018上关于图像语义分割的一篇论文，针对RGB-D的图像提出了一种新的算法。

此前关于RGB-D图像的分割方法主要有：

- 使用全卷积网络FCN将RGB信息和Depth信息使用两个独立的CNN来进行处理。这样处理会使得参数量和训练时间变为单个CNN的两倍，并且像素之间的关联也因此变弱。
- 使用3D networks来处理深度信息，但是这样操作会使得计算复杂度提升很多。

这篇论文提出了一种使用2D CNN来处理RGB-D图像语义分割的算法。为了解决像素之间深度信息关联性以及计算复杂度，参数量过多的问题，算法主要采用以下3个方式来解决：

- 2D CNN。使用传统的2D卷积神经网络结构不引入新的变量可以解决参数量过多、计算量过大的问题；
- depth-aware convolution。定义一种新的卷积方式来处理像素间深度信息关联问题；
- depth-aware average pooling。类似于depth-aware convolution，定义一种新的均值池化方式来处理像素间关联的问题。

3.1.1 Depth-aware Convolution

图3.1所示表示的是正常的卷积操作，按照式(3.1)的公式进行计算。其中 \mathcal{R} 是 p_0 的邻域， w 是卷积核。

$$y(p_0) = \sum_{p_n \in \mathcal{R}} w(p_n) \cdot x(p_0 + p_n) \quad (3.1)$$

图3.2所示表示的是 Depth-aware 的卷积操作，按照式 (3.2) 的公式进行计算。

$$y(p_0) = \sum_{p_n \in \mathcal{R}} w(p_n) \cdot F_D(p_0, p_0 + p_n) \cdot x(p_0 + p_n) \quad (3.2)$$

其中 F_D 表示像素之间深度信息的关联，如式 (3.3) 所示。

$$F_D(p_i, p_j) = \exp(-\alpha |D(p_i) - D(p_j)|) \quad (3.3)$$

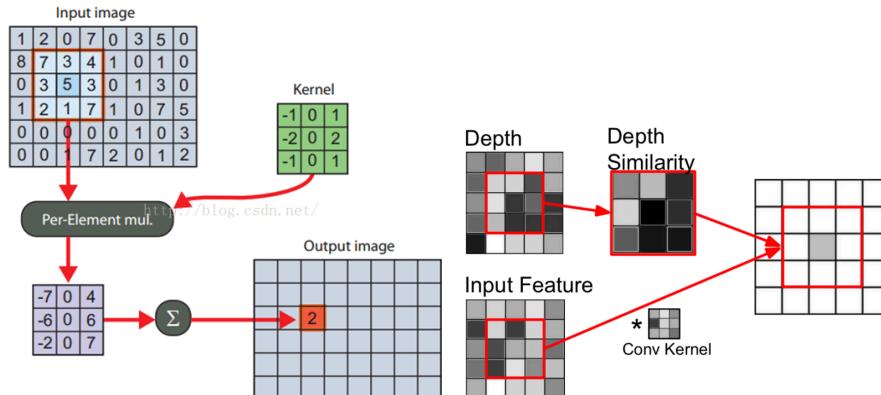


图 3.1: 传统的卷积操作

图 3.2: depth-aware 卷积操作

3.1.2 Depth-aware Average Pooling

与 Depth-aware 卷积类似，Depth-aware 平均值池化也引入了深度信息。图3.3所示表示的是正常的平均值池化操作，按照式 (3.4) 的公式进行计算。其中 \mathcal{R} 是 p_0 的邻域。

$$y(p_0) = \frac{1}{|\mathcal{R}|} \sum_{p_n \in \mathcal{R}} x(p_0 + p_n) \quad (3.4)$$

图3.4所示表示的是 Depth-aware 的平均值池化操作，按照式 (3.5) 的公式进行计算，其中 F_D 也表示式 (3.3) 所示的像素之间深度信息的关联。

$$y(p_0) = \frac{1}{|\mathcal{R}|} \sum_{p_n \in \mathcal{R}} F_D(p_0, p_0 + p_n) \cdot x(p_0 + p_n) \quad (3.5)$$

3.1.3 网络结构

本算法使用 DeepLab 作为 baseline，使用了 VGG16 和 ResNet-15 的网络结构，将其中的卷积操作改为式3.2中的卷积操作，将其中的均值池化操作改为式 (3.5) 中的均值池化操作。网络结构如图3.5所示。

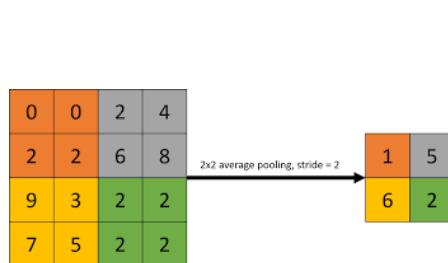


图 3.3: 传统的均值池化操作

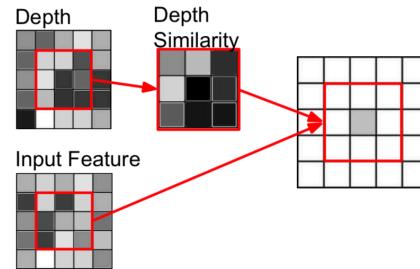


图 3.4: depth-aware 均值池化操作

| layer name | conv1_x | conv2_x | conv3_x | conv4_x | conv5_x | conv6 & conv7 |
|---------------------|----------|-----------|-----------|-----------|-----------|-------------------|
| Baseline DeepLab | C3-64-1 | C3-128-1 | C3-256-1 | C3-512-1 | C3-512-2 | C3-1024-12 |
| | C3-64-1 | C3-128-1 | C3-256-1 | C3-512-1 | C3-512-2 | C1-1024-0 |
| | maxpool | maxpool | C3-256-1 | C3-512-1 | C3-512-2 | globalpool+concat |
| | | | maxpool | maxpool | avgpool | |
| D-CNN | DC3-64-1 | DC3-128-1 | DC3-256-1 | DC3-512-1 | DC3-512-2 | DC3-1024-12 |
| | C3-64-1 | C3-128-1 | C3-256-1 | C3-512-1 | C3-512-2 | C1-1024-0 |
| | maxpool | maxpool | C3-256-1 | C3-512-1 | C3-512-2 | globalpool+concat |
| | | | maxpool | maxpool | Davgpool | |

图 3.5: DeepLab 和 DCNN 使用 VGG16 的结构

3.1.4 实验结果

本篇论文将提出的算法与 DeepLab 的 baseline 在 NYUv2^[3] 数据集上进行测试，NYUv2 数据集包括 1449 像素集标注的 RGB-D 图像，实验按照 40 个 class，将数据划分为训练集：795 张图像和测试集：654 张图像进行实验得到如图3.6和图3.7所示的结果。从图3.6中可以看出，修改了卷积操作和均值池化操作之后，在具有深度信息的图片上分割指标相比于上一章的 DeepLab 网络有了很大的提升。

| | Baseline | HHA | D-CNN | D-CNN+HHA |
|-----------|----------|------|-------------|-------------|
| Acc (%) | 50.1 | 59.1 | 60.3 | 61.4 |
| mAcc (%) | 23.9 | 30.8 | 39.3 | 35.6 |
| mIoU (%) | 15.9 | 21.9 | 27.8 | 26.2 |
| fwIoU (%) | 34.2 | 43.0 | 44.9 | 45.7 |

图 3.6: DeepLab 和 DCNN 在评价指标上的对比结果

从图3.7中可以直观的看出，加入了深度信息的 Depth-aware 卷积和 Depth-aware 均值池化使得图像分割效果有了很大的提升。从中我们也能看出，当我们能够获得的先验信息更多的时候，选择能够利用较多信息的算法可以为我们的识别算法带来很大的提升。



图 3.7: DeepLab 和 DCNN 在测试集上的效果图

第4章 总结

参考文献

- [1] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.
- [2] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. *arXiv e-prints*, page arXiv: 1405.0312, May 2014.
- [3] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012.
- [4] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [6] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *ArXiv e-prints*, mar 2016.
- [7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.
- [8] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.
- [9] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.
- [10] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

- [11] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018.
- [12] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. *CoRR*, abs/1612.01105, 2016.
- [13] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.
- [14] Weiyue Wang and Ulrich Neumann. Depth-aware cnn for rgb-d segmentation. *arXiv preprint arXiv:1803.06791*, 2018.