# 图像分割综述

## 简介

起源：2007年CV界已经开始讨论，但直到14年long等人的FCN才逐渐兴起。

图像语义分割就是机器自动从图像中分割出对象区域，并识别其中的内容。

物体检测需要用一个矩形框将对应的物体框出来，并且分辨出所框出来的物体的类别。物体分割不但需要分辨出物体的类别还需要对物体进行精确切割，而不仅仅是画出一个框。

因此，这里面涉及到两个任务：

（1）用物体识别技术识别物体，并用边界框表示出物体边界；

（2）根据语义不同给像素分类。

## 子领域简介

图像分割（image segmentation）任务的定义是：根据某些规则将图片分成若干个特定的、具有独特性质的区域，并提出感兴趣目标的技术和过程。

目前图像分割任务发展出了以下几个子领域：语义分割（semantic segmentation）、实例分割（instance segmentation）以及2018年刚兴起的新领域全景分割（panoptic segmentation）。

对于只有一个标签的（只区分类别）的任务，我们称之为“语义分割”（semantic segmentation）；对于区分相同类别的不同个体的，则称之为实例分割（instance segmentation）。由于实例分割往往只能分辨可数目标，因此，为了同时实现实例分割与不可数类别的语义分割，2018年Alexander Kirillov等人提出了全景分割（panoptic segmentation）的概念。

想要理清三个子领域的区别就不得不提到关于图像分割中 things 和 stuff 的区别：图像中的内容可以按照是否有固定形状分为 things 类别和 stuff 类别，其中，人，车等有固定形状的物体属于 things 类别（可数名词通常属于 things）；天空，草地等没有固定形状的物体属于 stuff 类别（不可数名词属于 stuff）。

### 2.1 语义分割（semantic segmentation）

语义分割（semantic segmentation）目的是在一张图里分割聚类出不同物体的pixel。

目前的主流框架都是基于FCN，FCN区别于分类网络（例如AlexNet）的最主要的差别是pixel-wise prediction，即FCN每个像素点都有个probability, 而AlexNet是一张图一个prediction。另外，AlexNet或者VGG通过一个小的trick(caffe/net\_surgery.ipynb at master · BVLC/caffe · GitHub)就可以转变成FCN，其他的分类网络也一样。

分类网络通常会在最后连接几层全连接层，它会将原来二维的矩阵（图片）压扁成一维的，从而丢失了空间信息，最后训练输出一个标量，即分类标签。而图像语义分割的输出需要是个分割图，且不论尺寸大小，但是至少是二维的，所以一般的做法是丢弃全连接层，换上全卷积层。这样比较下来，语义分割任务实际上是像素级的稠密估计任务。

### 2.2 实例分割（instance segmentation）

实例分割（instance segmentation，也称为Simultaneous Detection and Segmentation）。它要解决的问题是检测（Object Detection）和语义分割（Semantic Segmentation）综合的一个问题。比起检测，需要得到物体更精确的边界信息；比起语义分割，需要区分不同的物体个体。在实践中，典型的实例分割系统只能关注小部分视觉信息，一般约为 100 个目标类别。

1. instance segmentation与semantic segmentation

大致做法是在dense feature map上面整合个instance region proposal/score map/RoI, 然后再分割。这里instance segmentation本身又是跟object detection是紧密相关的。之前，之所以大家猛搞semantic segmentation而忽略instance segmentation的一个原因是没有好的数据集。

2）Instance segmentation与detection：

Semantic segmentation用一个FCN能解决得不错了，而Instance segmentation其实就和detection差不多了。

instance segmentation 的基本思路就是先做一遍 detection ，然后再在 detection 结果里面估计 segmentation 的结果。这个领域对应的脉络实际上是十分清晰的。

3）instance-level segmentation和多任务学习

有时候多个标注/任务是并列关系，可以通过Multi-Task Learning的框架来学习。另外一些情况，多个任务是递进关系，前一个任务的结果可以帮助后一个任务，例如将每一个人都独立的检测出来之后再分割每个人身体的Mask。合理利用这种递进关系，可以得到比并列关系更好的结果，这其实就是Instance segmentation的核心思想。

### 2.3 全景分割（panoptic segmentation）

全景分割可以说是语义分割和实例分割的结合。下图是全景分割结果，不同things类别被分割开，并且things类别的不同个体也被彼此分割开。不同stuff类别也被分割开。

全景分割的通用思路是将架构分为三个独立的部分：object instance segmentation 部分，stuff segmentation 部分，两子分支结果融合部分。其中，object instance segmentation 网络和 stuff segmentation 网络一般都是相互独立。

目前用于全景分割的常见公开数据集包括：MSCOCO、Vistas、ADE20K 和 Cityscapes。

全景分割的指标：Panoptic Quality(PQ)

### 2.4 场景解析（scene parsing）

### 2.5 协同分割（Co-segmentation）

定义：simultaneous segmentation of similar regions from two (or more) images，从两个(或更多)的图像中同时分割出相似区域。

问题产生背景：提高图像分割效果的措施有：(1)合并不同度量措施，加入全局优化函数（倾向于相干的地区）；(2)交互式信息；(3)对象类信息；(4)加入相似图像同时分割（即协同分割）。

应用：l）Object driven image retrieval 对象驱动的图像检索；2）video tracking and segmentation 视频跟踪和分割；3）interactive image editing 交互式图像编辑；4）image similarity measures 图像相似性度量。

目前，co-segmentation已经从简单的两幅图像协同分割扩展到多幅图像以及多对象。另外，受于协同概念的影响，一些类似的研究方向也一呼百出，如co-saliency、co-labeling、co-location和video co-segmentation等等，其思想基本相同，不同的仍旧是不同思想和技术解决不同的问题，算法是相同的，应用场景变成了多幅相似图像分割而已，这种方式也不乏是一种创新。

协同分割问题已经向着复杂图像集的处理方面发展，目前主要研究的方向已转向处理前景变化较大、背景和前景相似、图像集中背景相似、包含多对象等的图像集。但总体而言，这些算法还有一定的限制，如必须知道对象的数目、或者一幅图像中所有的对象都要出现等。如何能够准确、自动地从大量图像中分割出相似对象仍旧有研究的空间。后续会补上相关文献。

国内研究协同分割、显著性检测和视频分割的大牛已不少，例如成都电子科技大学的李宏亮、华中科技大学的陶文兵、天津大学的付华柱、工信所的操晓春等博士。

从论文效果上可以看出，协同分割普遍比单张图片的分割效果要好。

### 2.6人体解析（human parsing）

人体解析(Human Parsing)是细粒度的语义分割任务，是指将在图像中捕获的人分割成多个语义上一致的区域，旨在识别像素级别的人类图像的组成部分，例如，身体部位和衣物。作为一种细粒度的语义分割任务，它比仅是寻找人体轮廓的人物分割更具挑战性。

精度高的算法一般都会结合人体姿态（pose）进行人体解析。

谷歌开源了一个轻量级的算法，即BodyPix 2.0。

### 2.7 Matting

Matting是细粒度的语义分割任务，就是软分割，就是“抠图”。它所针对的是不同实例的交界处，一般应用就是抠出各种带毛的或者头发的边缘。

图像分割就是将图片中的像素分成多个类别，如果是前背景分割，那么就是分成两个类别，一个类别代表前景，一个类别代表背景，这类问题我们也称之为硬分割（Hard Segmentation）。Matting也是一类前背景分割问题，但是matting不是硬分割，而是软分割（Soft Segmentation），像玻璃、头发这类前景，对应像素点的颜色不只是由前景本身的颜色决定，而是前背景颜色融合的结果，matting问题的目标就是，找出前背景颜色，以及它们之间的融合程度。

matting传统方法主要分为两类：

1. 一类是color sampling的方法，以Bayesian matting为代表，通过对前景和背景的颜色采样构建高斯混合模型，但是这种方法需要高质量的trimap，不易获取。
2. 另一类是Propagation的方法，根据像素亲和度将用户绘制的信息传播到不确定像素，以Poisson Matting和KNN matting为代表，但是也不是自动抠图。

生成trimap一般就是简单的腐蚀膨胀操作。不同于一般的语义分割图，trimap中的每个像素都是三分类的，是soft的，对于交界处单独分为一类。

## 4.评价指标

1）区域相似度（Region Similarity）：区域相似度是掩膜 M 和真值 G 之间的

2）轮廓精确度（Contour Accuracy）：将掩膜看成一系列闭合轮廓的集合，并计算基于轮廓的 F 度量，即准确率和召回率的函数。即轮廓精确度是对基于轮廓的准确率和召回率的 F 度量。

直观上，区域相似度度量标注错误像素的数量，而轮廓精确度度量分割边界的准确率。

mIoU：这个指标是应用最多的，也是目前排名分割算法的依据。IoU就是每一个类别的交集与并集之比，而mIoU则是所有类别的平均IoU。论文均使用这一指标比较。由于分割也是分类问题，分类问题的指标一般使用混淆矩阵来求解。

Speed：由于有些分割算法是针对实时语义分割设计的，所以速度也是一个很重要的评价指标，当然评价速度需要公平比较，包括使用的图像大小、电脑配置一致。

当然还有其他指标，如pixel accuracy、mean accuraccy等。

语义分割边缘轮廓描述度量（BF）

Deeperlab提出了全景分割的评价指标解析覆盖（parsing covering，PC，该指标考虑了实例分辨率大小。区别于PQ，PQ会将相同“东西“类别的所有区域当作一个实例，不会考虑实例的尺寸大小。

## 5.实际应用

1）对人。B站防止弹幕挡住人，行人检测，交通监控。

2）医学影像分割。医学图形处理，定位各种异常，如肿瘤或特定的器官。被用于在医学扫描影像中识别显著性元素。该方法对识别图像中的异常（如肿瘤）十分有效。提升算法的准确率并解决低召回率的问题对于这种应用十分重要。

3）卫星图像，遥感影像的语义分割。从卫星图像中识别土地类型。典型的用例包括对水体进行分割以提供准确的地图信息。其他高级用例包括绘制道路图、确定作物类型、确定免费停车位等等。CVPR DeepGlobe 2018 Road Extraction Challenge，就是检测遥感图像中的道路。

4）自动驾驶。语义分割用于识别车道、车辆、人和其他感兴趣的物体，关注实时性。

5）场景理解。语义分割通常是更复杂任务的基础，如场景理解和可视化问答（VQA）。场景理解算法的输出通常是一个场景图或一段字幕。

6）时尚产业。语义分割在时尚产业中被用来从图像中提取出服装对象，为零售商店提供类似的建议。

7）指纹识别等取证技术

## 数据集

截至到2019年年初，VOC2012/Cityscape/CamVid/MSCOCO，所有数据集均没有达到饱和，仍有很大发展空间，无论在准确性还是实时性。

### 忽视标签(ignore label)

在介绍数据集之前，先来了解一下忽视标签的概念与作用。忽视标签从字面上理解，就是该标签不用于计算损失，也不用于计算精度。

以一个三个类别的例子为例。该例子，包含3个目标类别，1个背景类别，1个忽视类别。则标签可以设置为：0,1,2,3,4。其中，0代表背景，1-3代表目标类别，4代表忽视目标。那么，目标类别数到底该设为多少呢？4还是5。这里我看了很多地方，最简单的方式就是设置为4，这样训练过程中，4这个类别将不计算损失，同时我们也不需要进行预测argmax。

### VOC2012数据集

原始VOC2012数据集：1464训练集，1449验证集，1456测试集。经过加强的数据集，包含10582训练集，1449验证集，1456测试集。目前论文结果均使用加强的数据集进行训练。该数据集特点包括：

1）数据集包含21类别目标，20类目标+1类背景，分辨率大小不完全相同，为此训练过程为了能处理batch图像，因此需要将图像固定到某一大小，deeplab里面使用321大小。

2）该数据集中，label处理时发现并没有忽视标签，均在0-21之间，因此不需要设置忽视标签。在deeplab版本里作者将ignore\_index设为255，标签里面本来就没有255，设不设置没关系。

3）每幅图像包含的目标类别很少，但每个目标的轮廓还是比较复杂，因此，该数据集的分割精度没有想象中那么高。

VOC2012数据集：http://host.robots.ox.ac.uk/pascal/VOC/voc2012/

### CamVid数据集

2008年提出的一个数据集在SegNet里面将其做了处理https://github.com/alexgkendall/SegNet-Tutorial。

训练集367，验证集101，测试集233。三个均有标注文件。有一下几个特点：

1）图像数量太少，因此很少有人跑这个数据集，只有少数轻量级分割网络会跑，如ENet/Segnet/LinkNet/Bisenet等，现在能看到的最好精度也就60多mIOU。

2）原始数据集分辨率为960×720，而Segnet将其处理为480\*360，后者也是使用得多，可能是因为Segnet将其处理后的文件夹上传的原因吧。

3）该数据集也存在忽视标签，Segnet作者将其统一设置为11，类别数为11，标注为0-10，因此计算softmax损失时，需要设置ignore\_index=11，或者在处理标签图片时，将mask[index==11] = -1，这样处理后不计算忽视标签的损失。

4）由于图像分辨率为480\*360，且有11类目标，造成细小目标偏多，如行人，交通灯等。

### Cityscapes数据集

2016提出的用于自动驾驶的数据集。是一个包含 50 个城市街景的数据集，也是提供无人驾驶环境下的图像分割用的数据集。

训练集2975，验证集500，测试集1525。训练集与验证集包括Fine annotations以及额外的19998个coarse annotations，可以于训练集一起训练。测试集标注没有公开。数据集主要特点包括：

'name'/'id'/'trainId'/'category'/'catId'/'hasInstances'/'ignoreInEval'/'color '等，'id'是原始的标注id，而'trainId'是用于训练的Id，'ignoreInEval'为忽视对象。原始的标注包括35类，其中有16类为在评测中忽视的类别，因此训练时候只有19类，因此标签可以设置为0-18，验证的时候也可以使用0-18。如果要测试，则需要将0-18反变化回原始标签。

2）原始标注文件包括：color.png/instanceIds.png/labelIds.png/polygons.json四个文件，其中，labelIds.png文件为原始标注，因此需要将该文件转换为trainIds，可以参考labels.py和preparation，转换后忽视标签为255，而其他标签为0-18，这样训练过程使用19类目标，忽视255。在处理过程中，可以将mask[index==255] = -1，这样损失函数不会计算。或者计算softmax损失的时候设置ignore\_index=255。

3）图像分辨率较大，1024\*2048，如果使用原图训练，计算量太大，一般会crop一定的大小，但也不能太小，太小效果很差，猜想：这也是这个数据集精度很难提升的一个原因。

4）从提交的评测结果可以看出，基本都是深度学习方法(没有全部过一遍)。传统算法在这种环境复杂的交通场景做分割难度还是太大了。

5）每幅图像涵盖的目标类别较多，同时，存在很多细小目标，使得分割难度进一步增大。

### MSCOCO数据集

微软团队公布的，可以用来进行图像 recognition、segmentation 和 captioning 的数据集。主要从复杂的日常场景中截取，主要有 91 个类别，虽然类别比 ImageNet 少很多，但每一类的图像很多。

MSCOCO人像数据集：http://mscoco.org/explore/

### Vistas数据集

是全球最大的和最多样化的街景图像数据库，以帮助全球范围内的无人驾驶和自主运输技术。

### ADE20K数据集

是一个可用于场景感知、分割和多物体识别等多种任务的数据集。相比于大规模数据集 ImageNet 和 COCO，它的场景更多样化，相比于 SUN，它的图像数量更多，对数据的注释也更详细。

### Pascal Context数据集

### 人体部件数据集/人像解析数据集

目前三大主流人体部件数据集：ATR，Pascal-Person-Part和LIP

## 7.传统算法

1. 阈值技术

该技术的主要目的在于确定图像的最佳阈值。强度值超过阈值的像素其强度将变为1，其余像素的强度值将变为零，最后输出的是一个二值图。

用于选择阈值的方法有：Otsu，k均值聚类，和最大熵法。

直方图阈值化方法；混合化特征空间聚类

1. 运动与交互分割

该技术基于图像中的运动来进行分割。其思想很直观，在假设目标是运动的情况下找出两幅图中的差异，那么不同之处一定就是目标位置。

最简单的就是两图像的灰度图直接相减。

1. 边界检测

基于边界检测的图像分割算法。包含多种数学方法，其目的在于标出数字图像中处于图像亮度变化剧烈，或者更正式的讲，具有不连贯性的区域中的点。由于区域边界和边具有很高关联性，因此边界检测通常是这一种分割技术的前提步骤。

1. 区域增长方法

主要建立在同一区域中相邻像素具有相近像素值的假设之上。常见步骤为将像素与其近邻像素作比较，如果满足相似性标准，则该像素就可以被划分到以一个或更多其近邻点组成的聚类中去。相似性标准的选择很关键，并且在所有实例中其结果易受到噪声影响。

例如，种子区域生长法、区域分裂合并法、分水岭法。在实习的时候，我曾经使用它们来去除印章，事实证明对于扫描图像而言效果还是不错的。

5）其他

SVM等，研究人员一般使用纹理基元森林(TextonForest)或是随机森林(Random Forest)方法来构建用于语义分割的分类器。

# 基于深度学习的图像分割

## 常用算法

语义分割的行进路线是：FCN-unet-segnet-refinenet-large kernel-pspnet-deeplabv3+-各种花里胡哨attention的网络-各种图卷积网络-NAS

实例分割的行进路线是：two-stage的rcnn-mask的各种变体；one-stage的实例分割模型（不进行检测，直接分割）。

全景分割。

2015年，FCN被提出开始，它是基于深度学习的语义分割网络的开山之作，此后基于深度学习的语义分割网络被陆续提出。

2015、2016年，MNC、FCIS是这两年中最优秀的语义分割成果；

2017年，Mask R-CNN是提出的，性能上大幅度领先于之前的网络；

2019年，SOLO系列是轻量化。

截至到2019主流的算法，精度较高的算法包括：

1）谷歌Deeplab v3+，对于人像分割效果很好

2）U-Net++也可以

3）pspnet

4）HRNet-v2，全能型。HRNetV2-W40在比UNet++，DeepLabv3参数量更少时，实现更高的mIOU；HRNetV2-W48在和PSPNet参数量相同时，实现更高的mIOU。

5）mask-rcnn

截至到2019主流的算法，实时语义分割：

1. SegNet(池化索引结构)
2. ENet(减少下采样次数)
3. ESPNet(新型空间金字塔池化)
4. ICNet(多尺度图像作为输入)
5. BiSeNet(双路网络架构)

## 趋势

研究进展趋缓，甚至已进入瓶颈期，提升0.5 mIoU都非常困难。算法的改进方面，大多是网络结构上的改进，新模块中各种残差，各种增大感受野。

精度方面

1）分割就是分割，就是mask。

2）分割从表示上的局限性和其他任务之间的相关性，其他CV领域的思路对分割都可能有启发。例如Anchor-free的一系列工作，例如CornerNet, CenterNet, FCOS等，其实就是用FCN+Regression去做检测，实际上就是改变了groundtruth的表示形式。图像分割领域的论文PolarMask就是将实例分割问题转化为研究中心分类和极坐标距离回归的轮廓预测问题。

3）分割是learning，学到了真实世界的发布。之前都在supervised并且没有noise的框架下讨论问题，现在是考虑在半监督或者加噪声的提前下研究。

2019年，主流的方向是将语义分割网络进行轻量化处理，精度上都差不多到顶了。实例分割这个问题近几年的发展在很大程度上是由 COCO 数据集和比赛推动的。从 MNC，FCIS 到 PANet，都是在 COCO instance segmentation track 上拿第一名的方法。

## 难点

图像领域的一般性问题：尺度问题。因为相机和拍摄物体的距离远近引起的同一个物体在图像中可能占不同大小的画幅->多尺度问题。角度问题。拍摄角度物体的不同->物体多姿态（或者多视角）问题。光照问题。

1）分割边缘不准的问题。语义分割中，模型最容易误判的 pixel基本上都在物体边缘。

2）在同一副图像中不同类别或实例的像素不均衡的问题。

不同物体分割的难度也并不一样。在大多数情况下，我们是直接把训练模型后面直接接一个softmax，然后交叉熵训一波。但是这样训出来的模型，是建立在每个物体类别的像素数，以及每个物体的分割难度是差不多的假设下的。从图3可以看出往往样本少的，结构复杂导致不好分割的类别效果会比较差（例如sofa，bike，chair），而且整体的结果的方差还是相对比较大的。

3）标注费事费力，且标注中是可能存在噪声的。

4）对遮挡区域进行建模。如何定义图像的上下文问题。利用合理的上下文的建模机制，帮助网络猜测遮挡部分的语义信息。

5）分割领域的网络耗显存的问题。

6）分割领域的网络的速度。

数据集受限，少有的几个数据集已经刷到接近天花板。

算力消耗大，学界和一些小型业界团队往往被算力掣肘。

分割领域已经很难发论文了，研究同质化严重。

## 实际使用

可用的框架有如下几种：

FastAI 库：给出一张图像，该库能为图像中的物体创建蒙版；

Sefexa 图像分割工具：Sefexa 是一个用于图像分割、图像分析、创造基本事实的免费的半自动工具；

Deepmask：Facebook 研究中心的 Deepmask 是 DeepMask 和 SharpMask 的 Torch 实现；

MultiPath：它是一个来自「用于对象检测的多路径网络」的目标检测网络的 Torch 实现；

OpenCV：一个有超过 2500 种的优化算法的开源计算机视觉库；

MIScnn：一个医学图像分割的开源库。它仅需数行代码就能用 SOTA 卷积神经网络和深度学习模型建立路径；

Fritz：Fritz 提供了包括移动设备中的图像分割工具在内的几种计算机视觉工具。

## 常见框架

encoder-decoder方法：与经典的FCN中的skip-connection思想类似，encoder为分类网络，用于提取特征，而decoder则是将encoder的先前丢失的空间信息逐渐恢复，decoder的典型结构有U-Net/segnet/refineNet，该类方法虽然有一定的效果，能恢复部分信息，但毕竟信息已经丢失了，不可能完全恢复。

dialed FCN方法：deeplabv1提出的方法，将vgg的最后的两个pool层步长置为1，这样网络的输出分辨率从1/32变为1/8。可以保留更多的细节信息，同时也丢掉了复杂的decoder结构，但这种方法计算量大。

当然效果最后的就是将这两者结合，就是deeplabv3+的做法，但是可想计算量有多大。

### 全卷积

从FCN把深度学习引入图像分割到现在，一个通用的框架已经大概确定了。即前端使用FCN全卷积网络输出粗糙的label map，后端使用CRF条件随机场/MRF马尔科夫随机场等优化前端的输出，最后得到一个精细的分割图。

分类网络通常会在最后连接几层全连接层，它会将原来二维的矩阵（图片）压扁成一维的，从而丢失了空间信息，最后训练输出一个标量，即分类标签。而图像语义分割的输出需要是个分割图，且不论尺寸大小，但是至少是二维的，所以一般的做法是丢弃全连接层，换上全卷积层。这样比较下来，语义分割任务实际上是像素级的稠密估计任务。

### 编码-解码模型（Encoder-Decoder）

这是语义分割的主流框架，一般都是one-stage的。分为三个部分，即编码器（encoder）、解码器（decoder）和后处理（post-processing）。

#### 编码器（encoder）

即downsample的步骤。

一般来说，编码器直接使用预训练好了的分类网络，例如VGG/ResNet这类的分类网络（最最常用的是resnet512）。使用的分类网络性能越强，最后的效果也就越好。

这里需要注意的问题就是，传统的分类网络的下采样倍数太大了，不利于分割。通用的解决方法是去除pooling，在需要downsample的地方，使用dilated convlution之类的卷积。

在图像分割领域，由于是pixel级别的分类，所以一般是下采样会到原图的1/8或者1/16倍左右。

缩小特征图分辨率的过程就是zoom-out，具体是通过下采样downsample。这个过程中，卷积核所能感知的范围是随着网络深度加深而逐步增大的。另外，zoom-in就是图形放大；zoom-out就是图形缩小。

#### 解码器（decoder）

即upsample的步骤。一般会直接upsample到原图分辨率，或者先upsample到原图分辨率的一半，然后再做双线性插值。

upsample的方法三种：1）卷积+biliear（虽然计算量小，但精度一般都不高）；2）卷积+deconvlution；3）卷积+unpooling。

另外，实际上，decoder并不需要和encoder一样多一样深，《enet》中对此做过说明，大概意思是，encoder已经学习到了需要的东西，而decoder只是对feature map做一个upsample的refine，Encoder-Decoder的过程并不是要去做一个autoencoder。

解码器的任务是把低分辨率的特征（编码器得到的）投影到高分辨率的图像上，比较好的decoder利用了中间特征图用来映射。一般是通过skip connections或 pyramid pooling。

#### 后处理（post-processing）

后处理一般是用来刷榜，实际应用中一般不会用。常见的后处理方法是CRF。

它有两个问题：1）不符合end-to-end。2）太慢了，正常使用的dense crf会比神经网络慢很多，最后的实用场景基本不可能使用。

更深层地来讲，深度学习+概率图模型（PGM，Probabilistic Graphical Models）是一种趋势。深度学习可以更好的提取特征，而PGM能够从数学理论很好的解释事物本质间的联系。最常用的图模型就是CRF，概率图模型的网络化也是一种趋势，毕竟目标是end-to-end的学习系统。

### Mask-RCNN系列

说白了就是多任务（multi-task）学习。

### RNN系列

## 常用的技术和小trick

利用更丰富的空间信息，即扩大感受野。

结构位置信息丰富的浅层语义信息

### 卷积

#### 反卷积（deconvolution）

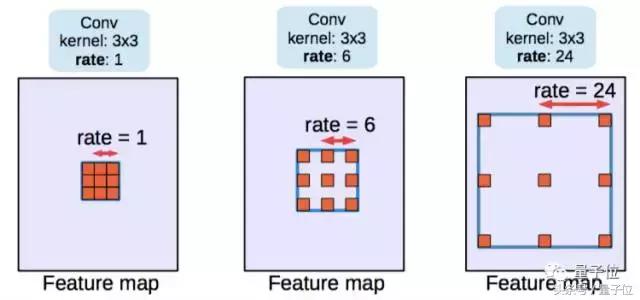
反卷积，转置卷积，deconvolution。

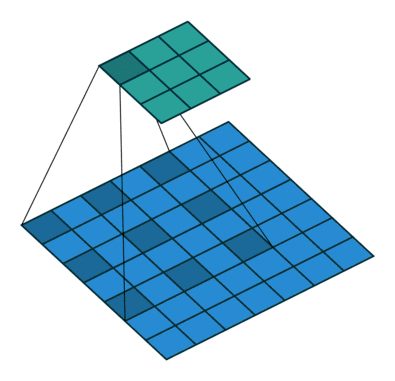
主要是用在编码-解码模型（Encoder-Decoder）中，不过从2019年以来，转置卷积就越来越少地用于分割领域。

#### 空洞卷积（dilated convolution）

空洞卷积，dilated convolution，atrous convolution。

在特征图的分辨率不变的前提下，扩大了感受野。原先许多结构采用了池化或striding操作来扩大感受野，而空洞卷积完全可以代替池化或striding操作。





#### 可变形卷积（deformable convolution）

会导致实际运算速度大幅度下降。

### 尺度

#### downsample和zoom-out features

见《编码器（encoder）》的小节。

#### Upsampling

见《解码器（decoder）》的小节。

#### 多尺度

ASPP是空间空洞卷积，本质就是在空间池化（SPP）的基础上使用了空洞卷积（Atrous convolution）。

多尺度方法，一般都是整合来自多个尺度特征图的方法。例如，在PSPNet中，先使用四种不同尺寸的卷积核和步长来执行池化操作，然后使用双线性插值进行上采样使得特征图统一到同一个分辨率，最后将代表不同尺度信息的特征图concat到一起并预测。

### 后处理和精细化

在CNN后得到的结果后，通常还会添上一种传统算法以提升精度。特别是对于matting任务会有很大的改进，即针对边缘的细节也被保留了，例如达到发丝级别。

1. K-nearest-neighbors methods
2. 条件随机场（conditional random fields，CRFs）

CRF方法是一种基于底层图像像素强度进行“平滑”分割的图模型，在运行时会将像素强度相似的点标记为同一类别。加入条件随机场方法可以提高1~2%的最终评分值。

可以通过考虑像素之间的先验关系来避免这些问题，如果目标是连续的，那么相邻的邻像素往往具有相同的标签。使用条件随机场（CRF）对这样的关系进行建模。

主要缺点是不能将其用作端到端体系结构的一部分。

在《CRF-as-RNN》中用RNN代替了CRF。

3）precision matrices。precision matrix 又被称为 concentration matrix精度矩阵/浓度矩阵，如果两个随机变量的协方差矩阵(covariance matrix)的逆矩阵存在。

### 训练方面的优化

#### 更多的数据

加入COCO人像数据集后，精度有了质的飞跃。

#### 数据增强技术

无论对于什么样的数据集，规模多大的数据集，在语义分割任务中，数据随机缩放给模型带来的收益永远是最大的。

对于随机翻转，镜像翻转，色彩偏移等数据增强技术手段为模型所带来的收益加一起也不一定有尺度的随机缩放所带来的效果好。

对原始图像进行标准化与归一化。

#### Loss函数的改进

1）二分类

focal loss。为了解决正负样本比例严重失衡的问题。是由log loss改进而来的。

2）多分类

多分类常见的是交叉熵，其他的损失函数也比较多，也可以对交叉熵加权。

3）Dice-loss

Dice-loss用于解决类极度不平衡的语义分割问题。它在《V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation 》一文中被提出，被用于计算预测出的类和真实类之间的重叠。特别地，对于二类分割，使用Dice-loss效果较好。

语义分割任务中CE loss和Dice loss的混合已经成为标准操作，而新的一些loss比如Focal Loss 和Exponential Logarithmic loss（指数对数loss）旨在通过设计loss形式达到难例挖掘的作用。

#### 训练策略

标签平滑，简单理解就是对预测结果进行惩罚。

def label\_smoothing(inputs, epsilon=0.1):

K = inputs.get\_shape().as\_list()[-1] # number of channels

return ((1-epsilon) \* inputs) + (epsilon / K)

### 模型的集成

已知神经网络的集合比单个网络更健壮和准确。然而，为模型平均训练多个深度网络在计算上是昂贵的。前面有讲，通过循环调整网络的学习率可使网络依次收敛到不同的局部最优解也可达到模型集成的作用，此处所讲的主要是多模型集成。

（1）直接平均法

直接平均法是最简单有效的多模型集成方法，通过直接平均不同模型产生的类别置信度得到最后额预测结果

（2）加权平均法

加权平均法是直接平均的基础上加入权重来调节不同模型输出间的重要程度。

1. 投票法（voting）

### Refinement方法

采用合适的CNN模型处理，得到输入图像低分辨率图像块的预测分类结果，然后结合降采样的原始图像，再次对上一步预测分类结果再进行迭代处理。这项举措有助于增大深度网络的感受野范围，同时能够对于原始图像信息的修正处理。注意，如果采用的网络参数相同，而且每次输入的图像相同，这种思想类似于RNN网络。

# 参考资料

https://www.zhihu.com/question/322077321

https://zhuanlan.zhihu.com/p/72743589

https://zhuanlan.zhihu.com/p/48670341

https://www.zhihu.com/question/272988870/answer/575302856

https://www.sohu.com/a/102118078\_114877