CVPR 19 : An End-to-End Network for Panoptic Segmentation

作者来自

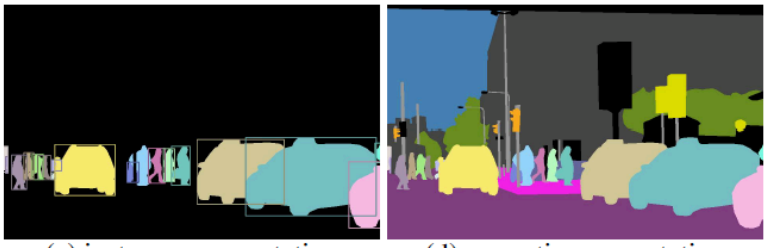
- 1 Zhejiang University
- 2 Megvii Inc. (Face++)
- 3 Huazhong University of Science and Technology
- 4 Peking University
- 5 The University of Tokyo

本文是在 Panoptic Segmentation 一文发表后的重要贡献。

01. Introduction

Panoptic Segmentation 是一个新的分割问题。它在 18 年年初由 Kaiming He 提出，发表在 arXiv 上，目前引用量仅仅为 31。目前在 CVPR 19 的文章中，已知的四篇文章里，有两篇是做全景分割的，一篇做弱监督学习，一篇做小样本下的医疗影像分割。

(a) image (b) semantic segmentation



(c) instance segmentation (d) panoptic segmentation

它可以认为是语义分割和实例分割的融合。

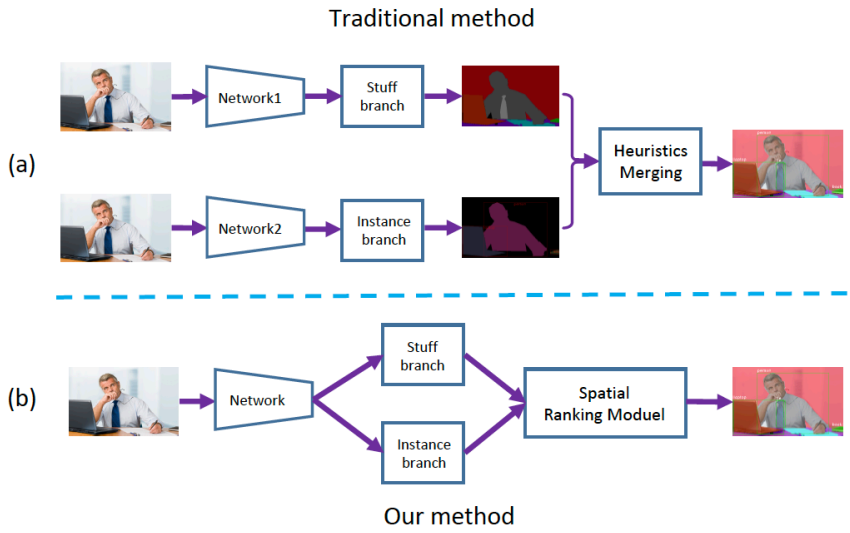
语义分割问题，要求对图片中所有的像素预测出类别标签。**实例分割**，则仅针对“可数的”类别要求进行预测类别，同时对于同类别内的物体，要求区分其 ID。即：

- things – countable objects such as people, animals, tools
- stuff – amorphous regions of similar texture or material such as grass, sky, road

而全景分割，则是对语义分割和实例分割的结合，要求在整张图片上对所有像素进行分类。

对于不可数的类别，如天空、路面和草地，仅要求给出类别标签，不要求给出类内 ID。对于可数的类别如车和人，则要求给出类别标签和类内 ID。

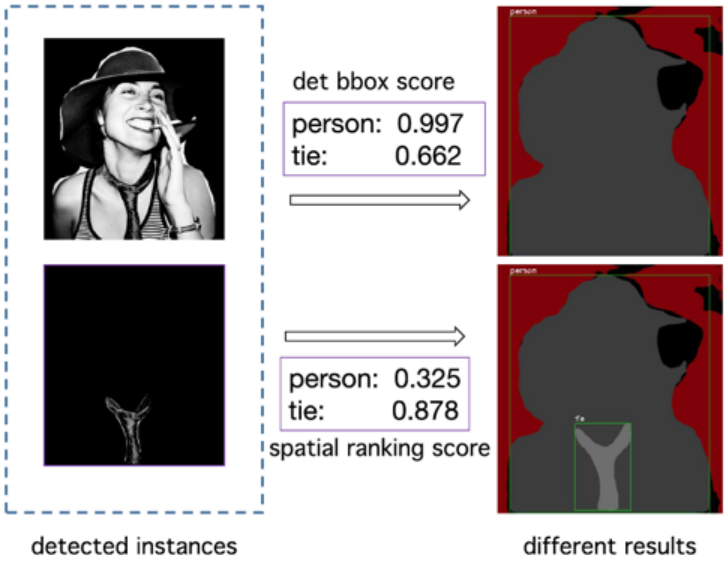
02. Motivation



已有的全景分割方法，是较为直接的对语义分割网络和实例分割网络的输出结果进行融合。融合方式是，对于每个像素来说，两**个网络的预测结果中谁的概率大（置信度高）就认为应该属于这个类别**。从另一个角度来看，即置信度高的类别的 mask 位于上层，置信度低的位于下层（被挡住），本质上是一个 overlapping 的问题。

但是这样的计算过程是**低效的**，同时由于骨干网络部分是分开训练的，在网络结构方面两个分支可能不同，所以图片的细粒度也可能不同。最后即使通过上采样获得了相同的图片尺寸，其仍然会受到中间计算过程的**细粒度的影响**。

另一方面，原本的融合方式也过于直接，以下图为例。

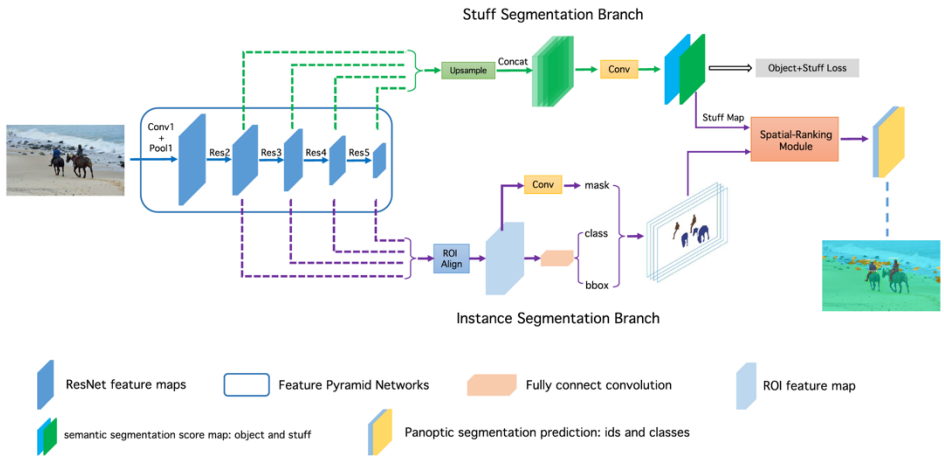


先对左上角的图片做 detection 分割，得分最高的前两个类别分别是人和领带。然后进行分割，得到人和领带的 mask。此时根据现有的融合办法，在预测时，人的置信度更高，所以此处应该是人的 mask 覆盖在领带的 mask 上，即会得到右上角的融合结果。

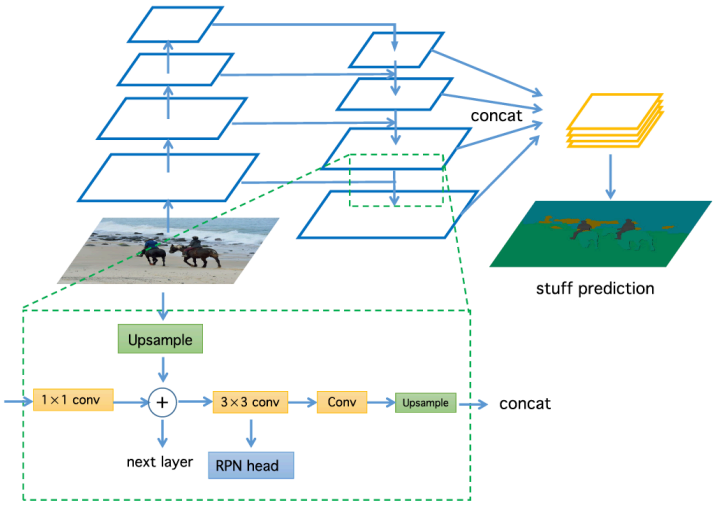
但是此处应该是领带位于图片的上层而人位于图片的下层。由此也就带来了很大的误差。

因此本文提出了 Spatial Ranking Module 方法来解决这个问题。

03. Network Structure



网络分为两个分支，在特征提取部分统一为 ResNet。这样能够使两个分支共享相同的细粒度，并平衡语义分割和实例分割的 loss（为什么？）。



上图为语义分割的分支。

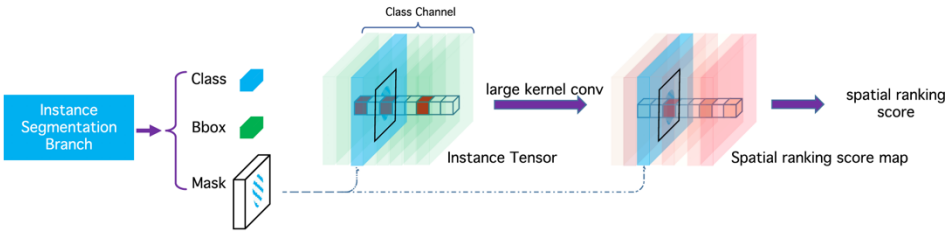


Figure 4. An illustration of spatial ranking score map prediction. The pixel vector in instance feature map represents instance prediction result in this pixel. The red color means that the corresponding category object includes this pixel and the multiple red channels indicate the occlusion problem between instances. We use the panoptic segmentation category label to supervise spatial ranking score map.

上图为实例分割的分支。

Mask 可以认为是一个预测概率图。对于 Mask 上的每个像素，都有 N（N 为类别数量）个通道，每个通道都是对这个像素所属类别的预测概率。

先将 Mask 映射到 Instance Tensor 上。Instance Tensor 初始化全为 0，然后对于有值的地方设为 1。

再经过一层卷积（卷积核尺寸很大，使用 kx1+1xk 的方式拆开）。得到一个 ranking score Map。然后用 pixel-wise cross entropy loss 来优化它。

$$L_{\text{srm}} = CE(S_{map}, S_{label})$$

然后，使用以下公式得到每个实例的排序分数。

$$P_{objs} = \frac{\sum_{(i,j)\in objs} S_{i,j,cls} \cdot m_{i,j}}{\sum_{(i,j)\in objs} m_{i,j}} \tag{3}$$

$$m_{i,j} = \begin{cases} 0 & \text{(i,j)} \in \text{instance} \\ 1 & \text{(i,j)} \notin \text{instance} \end{cases} \tag{4}$$

即计算实例的平均预测得分，将排序的判断依据从像素扩展到整个预测的实例范围上，更加可信。

为什么要使用卷积。直接在 MASK 上做不可以吗？不可以，因为 MASK 上没有形成掩膜，仍保留着所有的概率值，即使这些概率值不属于这个 instance。

可以认为将 Mask 映射到 Instance Tensor 的过程，是一个选择的过程：即对于 Mask 上的某个 Channel 来说，同一个 Channel 内有对整个图每个像素属于这个 Channel 所代表的类别的概率的预测值。对于大于某个阈值的概率，可以判断这个像素很大可能性属于这个类别。然后将大于阈值的点映射为 1，小于的点映射为 0。