PSPNet

论文地址：Pyramid Scene Parsing Network

发表日期：Submitted on 4 Dec 2016 (v1), last revised 27 Apr 2017 (this version, v2)

创新点

1. 提出金字塔场景解析网络（pyramid scene parsing network ，PSPNet），通过不同区域的上下文信息整合来得到全局上下文信息（global context information）。
2. 使用一种有效的优化策略： deeply supervised loss（实际上就是中继监督策略）。

思想

PSPNet的提出是用于复杂场景解析的（scene parsing），场景解析实际上就是语义分割，只不过场景解析场景比较复杂，物体的类别也比较多，因此场景解析的任务也比较难。

作者在文中提出，FCN类方法在复杂场景和unrestricted vocabulary的情况下，仍然存在一些问题。比如：

> 物体具有相似的表面，这个导致很难判断。比如下图的第一行，船被错误地预测为车。但是，如果这种场景下能有“河流”的上下文信息，可能就不至于判断错误。因此，上下文信息在复杂场景中对物体的识别是非常重要的。

因此，为了更准确的场景理解，就需要更多的场景上下文信息。而作者发现，FCN类方法很多都缺乏利用全局场景类别信息的策略（lack of suitable strategy to utilize global scene category clues）。 Spatial pyramid pooling 和 Spatial pyramid pooling network 在这方面有些尝试。不同于这些方法，本文提出了金字塔场景解析网络（PSPNet），可以更好地整合全局特征。PSPNet使用了global clues和local clues，使得结果更为准确。

除此之外，本文还提出了一种称为deeply supervised loss的优化方法。

观察

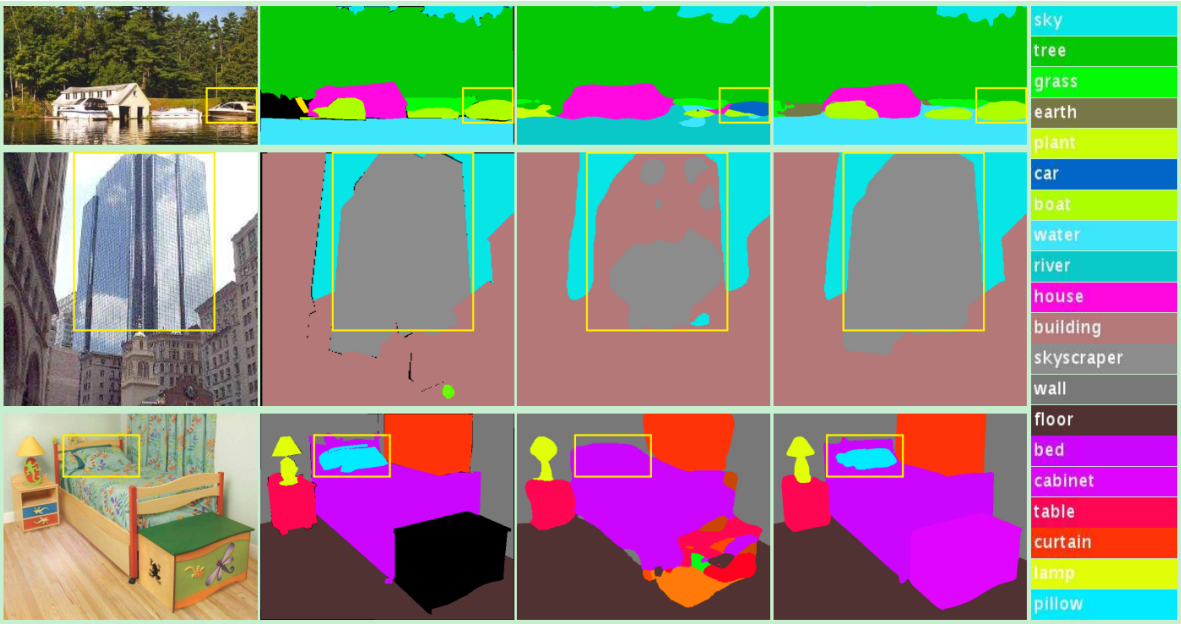

Figure 2. Scene parsing issues we observe on ADE20K [43] dataset. The first row shows the issue of mismatched relationship – cars are seldom over water than boats. The second row shows confusion categories where class "building" is easily confused as "skyscraper". The third row illustrates inconspicuous classes. In this example, the pillow is very similar to the bed sheet in terms of color and texture. These inconspicuous objects are easily misclassified by FCN.

作者观察了FCN类方法在场景解析中失败的一些例子，正是受这些观察的启发，作者提出了PSPNet。

主要有三点：

1. Mismatched relationship

上下文关系（context relationship）对于复杂场景解析来说是非常重要的。举个例子，飞机要么在天上飞，要么在飞机场上跑，但不可能在一个公路上。对于第一行中的例子，FCN将船预测成car，但如果能知道car几乎不可能在水上这个信息的话，就不会预测错误。

2. Confusion categories

在ADE20K数据集中存在很多类别很模糊的类，比如说field和earth，mountain和hill，wall，housebuilding和skyscraper。比如下图2中，摩天大楼就被FCN一部分预测为building，一部分预测为skyscraper。这个问题可以通过利用类别之间的相关关系可以得到解决。

3. Inconspicuous Classes

场景中包含任意大小的物体，一些小尺寸、不显著(Inconspicuous)的物体很难被发现，比如路灯和信号牌等。而，大尺寸的物体可能会超出FCN的接受野，导致预测结果不连续(discontinuous)。如上图中的第三行，pillow(枕头) 和 sheet(床单) 外形比较相似，Overlooking the global scene category may fail to parse the pillow（不知怎么翻译）。因此，为了能较好的考虑不同尺寸大小的物体，需要特别注意包含不显著物体的不同子区域。

模型

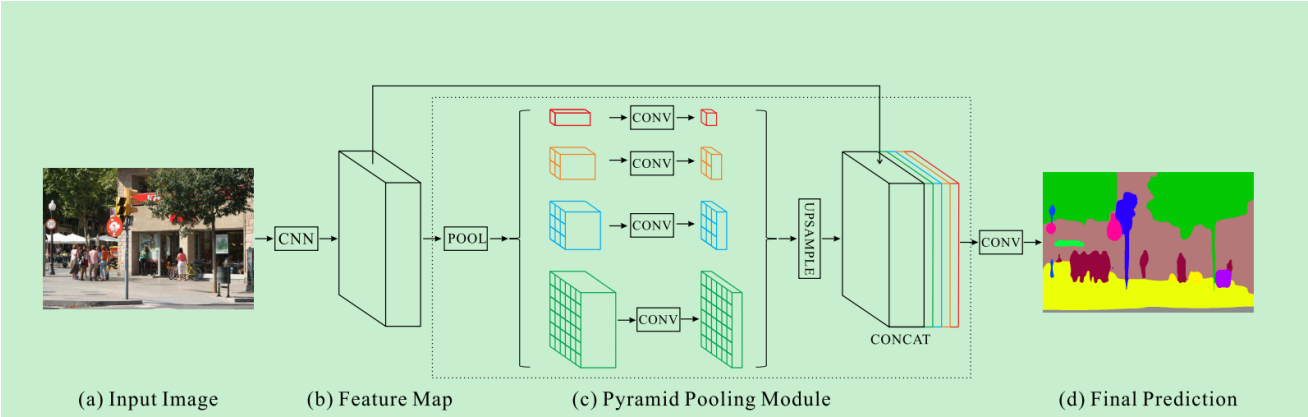

Figure 3. Overview of our proposed PSPNet. Given an input image (a), we first use CNN to get the feature map of the last convolutional layer (b), then a pyramid parsing module is applied to harvest different sub-region representations, followed by upsampling and concatenation layers to form the final feature representation, which carries both local and global context information in (c). Finally, the representation is fed into a convolution layer to get the final per-pixel prediction (d).

网络模型细节：

1. 用CNN网络提取图片特征，得到最后一层的卷积层的feature map（b）。骨架网络的选择是多样的，不过这篇文章中作者选择的是ResNet架构。
2. 然后通过空间金字塔进行pool。pool之后的feature map通道比较大，因此通过1x1的卷积进行降维处理。此时得到的feature map尺寸是不一样的，因此通过上采样方法将它们的尺寸变成一样，最后进行concat就得到了包含局部上下文信息和全局上下文信息的特征表达。

   （c）中红色部分代表global pooling，捕获的是 global contextual prior。2x2、3x3、6x6捕获的是不同尺度 sub-regions contextual prior
3. 将2得到的结果输入到最后的卷积层中进行预测。最后得到的结果是原图的1/8。因此在测试的时候还要经过上采样得到最后的分割图。

特征金字塔融合了四种尺度的特征。

注意一点：作者说金字塔的层级数和和各个层级的大小是可以修改的。

训练

为了更好地训练网络，作者在ResNet的stage4增加了一个附属loss。附属loss可以更好地优化学习。为了平衡两个losss，作者在它们前面添加了权重。

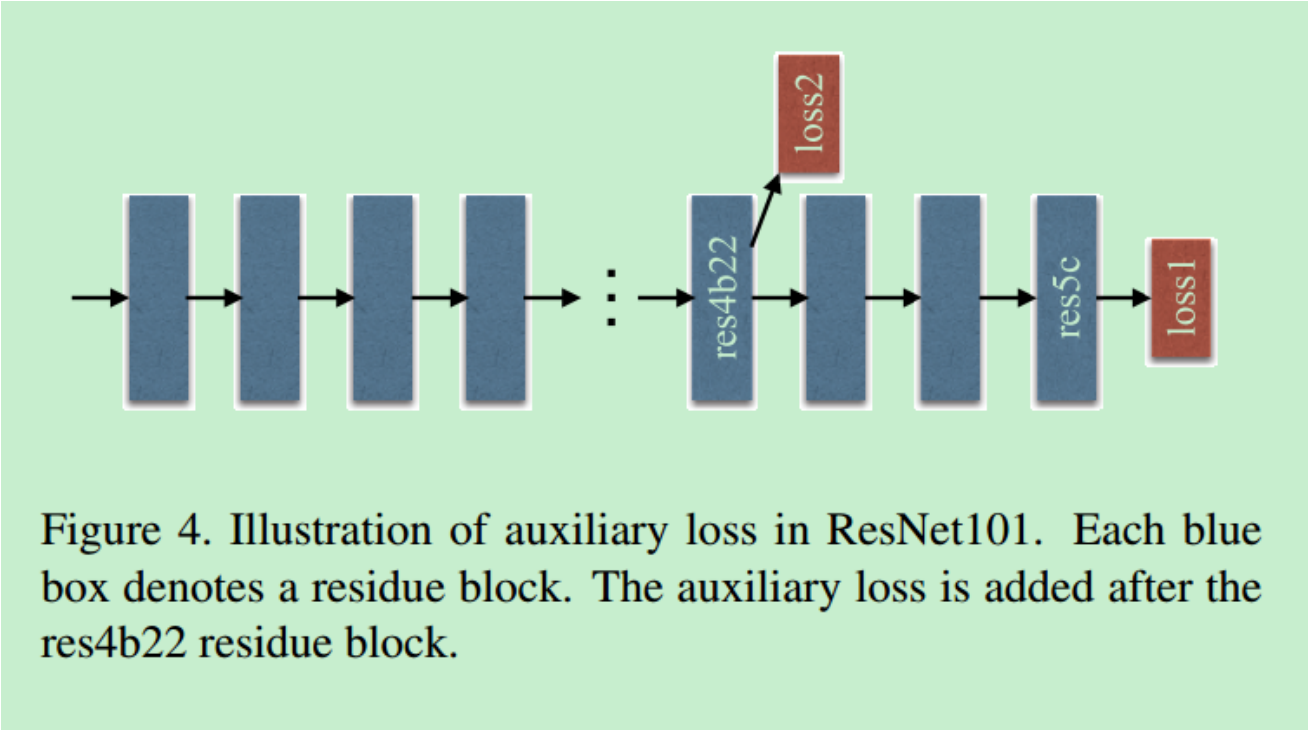

Figure 4. Illustration of auxiliary loss in ResNet101. Each blue box denotes a residue block. The auxiliary loss is added after the res4b22 residue block.

http s://www.cnblogs.com/everyday-haoguo/p/Note-PSPNet.html

实验结果

VOC 2012的结果

一些可视化的结果：

Figure 7. Visual improvements on PASCAL VOC 2012 data. PSP-Net produces more accurate and detailed results.

和其他方法的比较

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**

Table 6. Per-class results on PASCAL VOC 2012 testing set. Methods pre-trained on MS-COCO are marked with '†'.