Feature Pyramid Networks for Object Detection 译文

Tsung-Yi Lin, Piotr Dollar, PossGirshick, Kaiming He, Bharath Hariharan, and Serge Belongie

Facebook AI Research(FAIR)

## **摘要**

特征金字塔是识别系统中用于检测不同尺度目标的基本组件。但最近的深度学习目标检测器已经避免了金字塔表示，部分原因是它们是计算和内存密集型的。在本文中，我们利用深度卷积网络内在的多尺度、金字塔分级来构造具有很少额外成本的特征金字塔。开发了一种具有横向连接的自顶向下架构，用于在所有尺度上构建高级语义特征映射。这种称为特征金字塔网络（FPN）的架构在几个应用程序中作为通用特征提取器表现出了显著的改进。在一个基本的Faster R-CNN系统中使用FPN，没有任何不必要的东西，我们的方法可以在COCO检测基准数据集上取得最先进的单模型结果，结果超过了所有现有的单模型输入，包括COCO 2016挑战赛的获奖者。此外，我们的方法可以在GPU上以6FPS运行，因此是多尺度目标检测的实用和准确的解决方案。代码将公开发布。

## **引言**

识别不同尺度的目标是计算机视觉中的一个基本挑战。建立在图像金字塔之上的特征金字塔（我们简称为特征化图像金字塔）构成了标准解决方案的基础[1]（图1（a））。这些金字塔是尺度不变的，因为目标的尺度变化是通过在金字塔中移动它的层级来抵消的。直观地说，该属性使模型能够通过在位置和金字塔等级上扫描模型来检测大范围尺度内的目标。

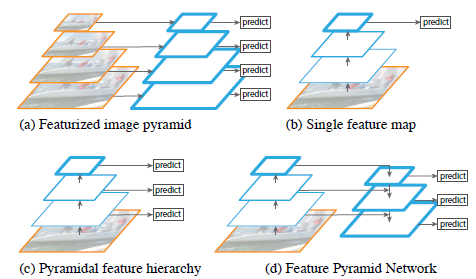


图1。（a）使用图像金字塔构建特征金字塔。每个图像尺度上的特征都是独立计算的，速度很慢。（b）最近的检测系统选择只使用单一尺度特征进行更快的检测。（c）另一种方法是重用ConvNet计算的金字塔特征层次结构，就好像它是一个特征化的图像金字塔。（d）我们提出的特征金字塔网络（FPN）与（b）和（c）类似，但更准确。在该图中，特征映射用蓝色轮廓表示，较粗的轮廓表示语义上较强的特征。

**解释图1：**

1. 图像金字塔，将图像做成不同的sacle,然后不同scale的图像生成对应的不同scale的特征，缺点是增加了时间成本，有些算法会在测试的时候使用图像金字塔。
2. 像SPPNet,Fast R-CNN, Faster R-CNN是 采用这种方式，即采用网络的最后一层特征。
3. 像SSD, 采用这种多尺度特征融合的办法，没有上采样的过程，从网络的不同的层，抽取不同尺度的特征做预测，这种方式不会增加额外的计算量，作者认为SSD没用用到足够的底层的特征，而作者认为底层的特征对检测小物体很有帮助。
4. 本文采用的方式，顶层特征通过上采样和底层特征做融合，并且每层都是独立预测的。

特征化图像金字塔在手工设计的时代被大量使用[5，25]。它们非常关键，以至于像DPM[7]这样的目标检测器需要密集的尺度采样才能获得好的结果（例如每组10个尺度，octave参考SIFT特征）。**对于识别任务，工程特征大部分已经被深度卷积网络（ConvNets）[19，20]计算的特征所取代**。除了能够表示更高级别的语义，ConvNets对于尺度变化也更加鲁棒，从而有助于从单一输入尺度上计算的特征进行识别[15，11，29]（图1（b））。但即使有这种鲁棒性，金字塔仍然需要得到最准确的结果。在ImageNet[33]和COCO[21]检测挑战中，最近的所有排名靠前的输入都使用了针对特征化图像金字塔的多尺度测试（例如[16，35]）。**对图像金字塔的每个层次进行特征化的主要优势在于它产生了多尺度的特征表示，其中所有层次上在语义上都很强，包括高分辨率层。**

尽管如此，**特征化图像金字塔的每个层次都具有明显的局限性。推断时间显著增加（例如，四倍[11]）**，**使得这种方法在实际应用中不切实际**。此外，在图像金字塔上端对端地训练深度网络在内存方面是不可行的，所以如果被采用，图像金字塔仅在测试时被使用[15，11，16，35]，这造成了训练/测试时推断的不一致性。出于这些原因，Fast和Faster R-CNN[11，29]选择在默认设置下不使用特征化图像金字塔。

但是，图像金字塔并不是计算多尺度特征表示的唯一方法。深层ConvNet逐层计算特征层级，而对于下采样层，特征层级具有内在的多尺度金字塔形状。这种网内特征层级产生不同空间分辨率的特征映射，但引入了由不同深度引起的较大的语义差异。高分辨率映射具有损害其目标识别表示能力的低级特征。

One-stage目标检测器（SSD）[22]是首先尝试使用ConvNet的金字塔特征层级中的一个，好像它是一个特征化的图像金字塔（图1（c））。理想情况下，SSD风格的金字塔将重用正向传递中从不同层中计算的多尺度特征映射，因此是零成本的。但为了避免使用低级特征，SSD放弃重用已经计算好的图层，而从网络中的最高层开始构建金字塔（例如，VGG网络的conv4\_3[36]），然后添加几个新层。因此**它错过了重用特征层级的更高分辨率映射的机会**。**我们证明这些对于检测小目标很重要。**

**本文的目标是自然地利用ConvNet特征层级的金字塔形状，同时创建一个在所有尺度上都具有强大语义的特征金字塔。**为了实现这个目标，我们所依赖的架构将低分辨率、强语义的特征与高分辨率、弱语义的特征通过自顶向下的路径和横向连接相结合。（图1（d））。其结果是一个特征金字塔，在所有级别都具有丰富的语义，并且可以从单个输入图像尺度上进行快速构建。换句话说，我们展示了如何创建网络中的特征金字塔，可以用来代替特征化的图像金字塔，而不牺牲表示能力，速度或内存。

最近的研究[28，17，8，26]中流行采用自顶向下和跳跃连接的类似架构。他们的目标是生成具有高分辨率的单个高级特征映射，并在其上进行预测（图2顶部）。相反，我们的方法利用这个架构作为特征金字塔，其中预测（例如目标检测）在每个级别上独立进行（图2底部）。我们的模型反映了一个特征化的图像金字塔，这在这些研究中还没有探索过。

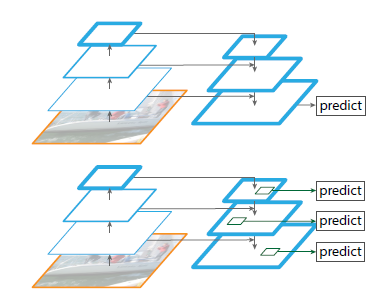


图2。顶部：带有跳跃连接的自顶向下的架构，在最好的级别上进行预测（例如，[28]）。底部：我们的模型具有类似的结构，但将其用作特征金字塔，并在各个层级上独立进行预测。

我们评估了我们称为特征金字塔网络（FPN）的方法，其在各种系统中用于检测和分割[11，29，27]。没有任何不必要的东西，我们在具有挑战性的COCO检测基准数据集上报告了最新的单模型结果，仅仅基于FPN和基本的Faster R-CNN检测器[29]，就超过了竞赛获奖者所有现存的严重工程化的单模型竞赛输入。在消融实验中，我们发现对于边界框提议，FPN将平均召回率（AR）显著增加了8个百分点；对于目标检测，它将COCO型的平均精度（AP）提高了2.3个百分点，PASCAL型AP提高了3.8个百分点，超过了ResNet[16]上Faster R-CNN强大的单尺度基准线。我们的方法也很容易扩展掩模提议，改进实例分隔AR，加速严重依赖图像金字塔的最先进方法。（目标检测，语义/实例分割）

另外，我们的金字塔结构可以通过所有尺度进行端对端培训，并且在训练/测试时一致地使用，这在使用图像金字塔时是内存不可行的。因此，FPN能够比所有现有的最先进方法获得更高的准确度。此外，这种改进是在不增加单尺度基准测试时间的情况下实现的。我们相信这些进展将有助于未来的研究和应用。我们的代码将公开发布。

## **相关工作**

****手工设计特征和早期神经网络****。SIFT特征[25]最初是从尺度空间极值中提取的，用于特征点匹配。HOG特征[5]，以及后来的SIFT特征，都是在整个图像金字塔上密集计算的。这些HOG和SIFT金字塔已在许多工作中得到了应用，用于图像分类，目标检测，人体姿势估计等。这对快速计算特征化图像金字塔也很有意义。Dollar等人[6]通过先计算一个稀疏采样（尺度）金字塔，然后插入缺失的层级，从而演示了快速金字塔计算。在HOG和SIFT之前，使用ConvNet[38，32]的早期人脸检测工作计算了图像金字塔上的浅网络，以检测跨尺度的人脸。

****Deep ConvNet目标检测器****。随着现代深度卷积网络[19]的发展，像OverFeat[34]和R-CNN[12]这样的目标检测器在精度上显示出了显著的提高。OverFeat采用了一种类似于早期神经网络人脸检测器的策略，通过在图像金字塔上应用ConvNet作为滑动窗口检测器。R-CNN采用了基于区域提议的策略[37]，其中每个提议在用ConvNet进行分类之前都进行了尺度归一化。SPPnet[15]表明，这种基于区域的检测器可以更有效地应用于在单个图像尺度上提取的特征映射。最近更准确的检测方法，如Fast R-CNN[11]和Faster R-CNN[29]提倡使用从单一尺度计算出的特征，因为它提供了精确度和速度之间的良好折衷。然而，多尺度检测性能仍然更好，特别是对于小型目标。

****使用多层的方法****。一些最近的方法通过使用ConvNet中的不同层来改进检测和分割。FCN[24]将多个尺度上的每个类别的部分分数相加以计算语义分割。Hypercolumns[13]使用类似的方法进行目标实例分割。在计算预测之前，其他几种方法（HyperNet[18]，ParseNet[23]和ION[2]）将多个层的特征连接起来，这相当于累加转换后的特征。SSD[22]和MS-CNN[3]可预测特征层级中多个层的目标，而不需要组合特征或分数。

最近有一些方法利用横向/跳跃连接将跨分辨率和语义层次的低级特征映射关联起来，包括用于分割的U-Net[31]和SharpMask[28]，Recombinator网络[17]用于人脸检测以及Stacked Hourglass网络[26]用于关键点估计。Ghiasi等人[8]为FCN提出拉普拉斯金字塔表示，以逐步细化分割。尽管这些方法采用的是金字塔形状的架构，但它们不同于特征化的图像金字塔[5，7，34]，其中所有层次上的预测都是独立进行的，参见图2。事实上，对于图2（顶部）中的金字塔结构，图像金字塔仍然需要跨多个尺度上识别目标[28]。

## **FPN**

我们的目标是利用ConvNet的金字塔特征层级，该层次结构具有从低到高的语义，并在整个过程中构建具有高级语义的特征金字塔。由此产生的特征金字塔网络是通用的，在本文中，我们侧重于滑动窗口提议（Region Proposal Network，简称RPN）[29]和基于区域的检测器（Fast R-CNN）[11]。在第6节中我们还将FPN泛化到实例分割中。

我们的方法以任意大小的单尺度图像作为输入，并以全卷积的方式输出多层适当大小的特征映射。这个过程独立于主卷积体系结构（例如[19，36，16]），在本文中，我们呈现了使用ResNets[16]的结果。如下所述，**我们的金字塔结构包括自下而上的路径，自上而下的路径和横向连接。**

****自下而上的路径****。自下向上的路径是主ConvNet的前馈计算，其计算由尺度步长为2的多尺度特征映射组成的特征层级。通常有许多层产生相同大小的输出映射，并且我们认为这些层位于相同的网络阶段。对于我们的特征金字塔，我们为每个阶段定义一个金字塔层。我们选择每个阶段的最后一层的输出作为我们的特征映射参考集，我们将丰富它来创建我们的金字塔。这种选择是自然的，因为每个阶段的最深层应具有最强大的特征。

具体而言，对于ResNets[16]，我们使用每个阶段的最后一个残差块输出的特征激活。对于conv2，conv3，conv4和conv5输出，我们将这些最后残差块的输出表示为{C2,C3,C4,C5}，并注意相对于输入图像它们的步长为{4，8，16，32}个像素。由于其庞大的内存占用，我们不会将conv1纳入金字塔

****自顶向下的路径和横向连接****。**自顶向下的路径通过上采样空间上更粗糙但在语义上更强的来自较高金字塔等级的特征映射来幻化更高分辨率的特征**。**这些特征随后通过来自自下而上路径上的特征经由横向连接进行增强。每个横向连接合并来自自下而上路径和自顶向下路径的具有相同空间大小的特征映射。自下而上的特征映射具有较低级别的语义，但其激活可以更精确地定位，因为它被下采样的次数更少。**

**图3**显示了建造我们的自顶向下特征映射的构建块。使用较粗糙分辨率的特征映射，我们将空间分辨率上采样为2倍（为了简单起见，使用最近邻上采样）。然后通过按元素相加，将上采样映射与相应的自下而上映射（其经过1×1卷积层来减少通道维度）合并。迭代这个过程，直到生成最佳分辨率映射。为了开始迭代，我们只需在C5上添加一个1×1卷积层来生成最粗糙分辨率映射。最后，我们在每个合并的映射上添加一个3×3卷积来生成最终的特征映射，这是为了减少上采样的混叠效应。这个最终的特征映射集称为{P2,P3,P4,P5}，对应于{C2,C3,C4,C5}，分别具有相同的空间大小。

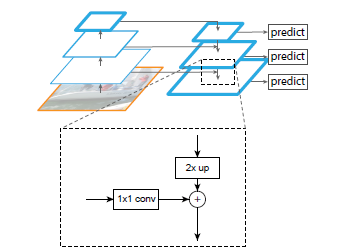
****

图3 构建模块说明了横向连接和自顶向下路径，通过加法合并

由于金字塔的所有层都像传统的特征图像金字塔一样使用共享分类器/回归器，因此我们在所有特征映射中固定特征维度（通道数记为d）。我们在本文中设置d=256，因此所有额外的卷积层都有256个通道的输出。在这些额外的层中没有非线性，我们在实验中发现这些影响很小。

简洁性是我们设计的核心，我们发现我们的模型对许多设计选择都很鲁棒。我们已经尝试了更复杂的块（例如，使用多层残差块[16]作为连接）并观察到稍微更好的结果。设计更好的连接模块并不是本文的重点，所以我们选择上述的简单设计。

## **应用**

我们的方法是在深度ConvNets内部构建特征金字塔的通用解决方案。在下面，我们采用我们的方法在RPN[29]中进行边界框提议生成，并在Fast R-CNN[11]中进行目标检测。为了证明我们方法的简洁性和有效性，我们对[29，11]的原始系统进行最小修改，使其适应我们的特征金字塔。

### **4.1. RPN的特征金字塔网络（说白了就是一个3X3的滑动卷积）**

RPN[29]是一个滑动窗口类不可知的目标检测器。在原始的RPN设计中，一个小型子网络在密集的3×3滑动窗口，单尺度卷积特征映射上进行评估，执行目标/非目标的二分类和边界框回归。这是**通过一个3×3的卷积层实现**的，后面跟着两个用于分类和回归的1×1兄弟卷积，我们称之为网络头部。目标/非目标标准和边界框回归目标的定义是关于一组称为锚点的参考框的[29]。**这些锚点具有多个预定义的尺度和长宽比，以覆盖不同形状的目标。**

**我们通过用我们的FPN替换单尺度特征映射来适应RPN**。**我们在我们的特征金字塔的每个层级上附加一个相同设计的头部（3x3 conv和两个1x1兄弟convs）**。**由于头部在所有金字塔等级上的所有位置密集滑动，所以不需要在特定层级上具有多尺度锚点(尺度已经在金字塔层级上体现出来了）**。**相反，我们为每个层级分配单尺度的锚点**。在形式上，我们定义锚点{P2,P3,P4,P5,P6}分别具有${32^2,64^2,128^2,256^2,512^2}$个像素的面积。正如在[29]中，我们在每个层级上也使用了多个长宽比${1:2,1:1,2:1}$的锚点。所以在金字塔上总共有十五个锚点。

如[29]，我们根据锚点和实际边界框的交并比（IoU）比例将训练标签分配给锚点。形式上**，如果一个锚点对于一个给定的实际边界框具有最高的IoU或者与任何实际边界框的IoU超过0.7，则给其分配一个正标签，如果其与所有实际边界框的IoU都低于0.3，则为其分配一个负标签。（和Faster R-CNN是一样的）。**请注意，实际边界框的尺度并未明确用于将它们分配到金字塔的层级；相反，实际边界框与已经分配给金字塔等级的锚点相关联。因此，除了[29]中的内容外，我们不引入额外的规则。

**我们注意到头部的参数在所有特征金字塔层级上共享**；我们也评估了替代方案，没有共享参数并且观察到相似的准确性。共享参数的良好性能表明我们的金字塔的所有层级共享相似的语义级别。这个优点类似于使用特征图像金字塔的优点，其中可以将常见头部分类器应用于在任何图像尺度下计算的特征。

通过上述改编，RPN可以自然地通过我们的FPN进行训练和测试，与[29]中的方式相同。我们在实验中详细说明实施细节。

### **4.2. Fast R-CNN的特征金字塔网络**

Fast R-CNN[11]是一个基于区域的目标检测器，利用感兴趣区域（RoI）池化来提取特征。Fast R-CNN通常在单尺度特征映射上执行。**要将其与我们的FPN一起使用，我们需要为金字塔等级分配不同尺度的RoI。**

我们将我们的特征金字塔看作是从图像金字塔生成的。因此，当它们在图像金字塔上运行时，我们可以适应基于区域的检测器的分配策略[15，11]。在形式上，我们通过以下公式将宽度为w和高度为h（在网络上的输入图像上）的RoI分配到特征金字塔的级别$P\_k$上：

$$k = ⌊k\_0+\log\_2(\sqrt{wh}/224)⌋$$

这里224是规范的ImageNet预训练大小，而$k\_0$是大小为$w×h=224^2$的RoI应该映射到的目标级别。类似于基于ResNet的Faster R-CNN系统[16]使用C4作为单尺度特征映射，我们将$k\_0$设置为4。直觉上，方程（1）意味着如果RoI的尺寸变小了（比如224的1/2），它应该被映射到一个更精细的分辨率级别（比如$k=3$）

我们在所有级别的所有RoI中附加预测器头部（在Fast R-CNN中，预测器头部是特定类别的分类器和边界框回归器）。再次，预测器头部都共享参数，不管他们在什么层级。在[16]中，ResNet的conv5层（9层深的子网络）被用作conv4特征之上的头部，但我们的方法已经利用了conv5来构建特征金字塔。因此，与[16]不同，我们只是采用RoI池化提取7×7特征，并在最终的分类层和边界框回归层之前附加两个隐藏单元为1024维的全连接（fc）层（每层后都接ReLU层）。这些层是随机初始化的，因为ResNets中没有预先训练好的fc层。请注意，与标准的conv5头部相比，我们的2-fc MLP头部更轻更快。

基于这些改编，我们可以在特征金字塔之上训练和测试Fast R-CNN。实现细节在实验部分给出。

## **5. 目标检测实验**

我们在80类的COCO检测数据集[21]上进行实验。我们训练使用80k张训练图像和35k大小的验证图像子集（trainval35k[2]）的联合，并报告了在5k大小的验证图像子集（minival）上的消融实验。我们还报告了在没有公开标签的标准测试集（test-std）[21]上的最终结果。

正如通常的做法[12]，所有的网络骨干都是在ImageNet1k分类集[33]上预先训练好的，然后在检测数据集上进行微调。我们使用公开可用的预训练的ResNet-50和ResNet-101模型。我**们的代码是使用Caffe2重新实现py-faster-rcnn**。

### **5.1. 区域提议与RPN**

根据[21]中的定义，我们评估了COCO类型的平均召回率（AR:average Recall）和在小型，中型和大型目标($AR\_s, AR\_m​, and AR\_l$)上的AR。我们报告了每张图像使用100个提议和1000个提议的结果($AR^{100} and AR^{1k}$)。

****实施细节****。表1中的所有架构都是端对端训练。输入图像的大小调整为其较短边有800像素。我们采用8个GPU进行同步SGD训练。小批量数据包括每个GPU上2张图像和每张图像上256个锚点。我们使用0.0001的权重衰减和0.9的动量。前30k次小批量数据的学习率为0.02，而下一个10k次的学习率为0.002。对于所有的RPN实验（包括基准数据集），我们都包含了图像外部的锚盒来进行训练，这不同于[29]中的忽略这些锚盒。其它实现细节如[29]中所述。使用具有FPN的RPN在8个GPU上训练COCO数据集需要约8小时。

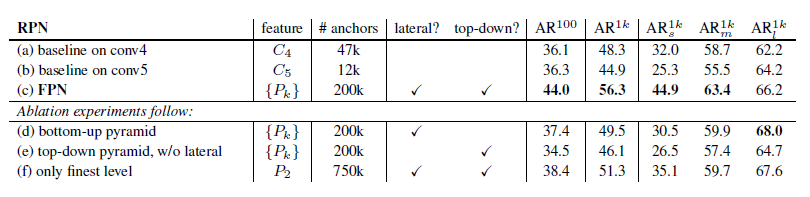


表1。使用RPN[29]的边界框提议结果，在COCO的minival数据集上进行评估。所有模型都是通过trainval35k训练的。列“lateral”和“top-down”分别表示横向连接和自上而下连接的存在。列“feature”表示附着头部的特征映射。所有结果都是基于ResNet-50的并共享相同的超参数。

#### **5.1.1 消融实验**

****基线进行比较****。为了与原始RPNs[29]进行公平比较，我们使用$C\_4$(与[16]相同)或$*C*\_5$的单尺度映射运行了两个基线（表1（a，b）），都使用与我们相同的超参数，包括使用5种尺度锚点的单尺度映射运行了两个基线（表1（a，b）），都使用与我们相同的超参数，包括使用5种尺度锚点的单尺度映射运行了两个基线（表1（*a*，*b*）），都使用与我们相同的超参数，包括使用5种尺度锚点${32^2, 64^2, 128^2, 256^2, 512^2}$。表1（b）显示没有优于（a），这表明单个更高级别的特征映射是不够的，因为存在在较粗分辨率和较强语义之间的权衡。

将FPN放在RPN中可将AR^{1k}提高到56.3（表1（c）），这比单尺度RPN基线（表1（a））增加了****8.0****个点。此外，在小型目标（AR^{1k}\_s）上的性能也大幅上涨了12.9个点。我们的金字塔表示大大提高了RPN对目标尺度变化的鲁棒性。

\*\*自上而下改进的重要性如何？\*\*表1（d）显示了没有自上而下路径的特征金字塔的结果。通过这种修改，将1×1横向连接和后面的3×3卷积添加到自下而上的金字塔中。该架构模拟了重用金字塔特征层次结构的效果（图1（b））。

\*\*横向连接有多重要？\*\*表1（e）显示了没有1×1横向连接的自顶向下特征金字塔的消融结果。这个自顶向下的金字塔具有强大的语义特征和良好的分辨率。但是我们认为这些特征的位置并不精确，因为这些映射已经进行了多次下采样和上采样。更精确的特征位置可以通过横向连接直接从自下而上映射的更精细层级传递到自上而下的映射。因此，FPN的AR^1k的得分比表1（e）高10个点。

\*\*金字塔表示有多重要？\*\*可以将头部附加到P\_2的最高分辨率的强语义特征映射上（即我们金字塔中的最好层级），而不采用金字塔表示。与单尺度基线类似，我们将所有锚点分配给P\_2特征映射。这个变体（表1（f））比基线要好，但不如我们的方法。RPN是一个具有固定窗口大小的滑动窗口检测器，因此在金字塔层级上扫描可以增加其对尺度变化的鲁棒性。

另外，我们注意到由于P\_2较大的空间分辨率，单独使用P\_2会导致更多的锚点（750k，表1（f））。这个结果表明，大量的锚点本身并不足以提高准确率。

### **5.2. 使用Fast/Faster R-CNN的目标检测**

接下来我们研究基于区域（非滑动窗口）检测器的FPN。我们通过COCO类型的平均精度（AP）和PASCAL类型的AP（单个IoU阈值为0.5）来评估目标检测。我们还按照[21]中的定义报告了在小尺寸，中尺寸和大尺寸（即AP\_s,AP\_m和AP\_l）目标上的COCO AP。

****实现细节****。调整大小输入图像，使其较短边为800像素。同步SGD用于在8个GPU上训练模型。每个小批量数据包括每个GPU2张图像和每张图像上512个RoI。我们使用0.0001的权重衰减和0.9的动量。前60k次小批量数据的学习率为0.02，而接下来的20k次迭代学习率为0.002。我们每张图像使用2000个RoIs进行训练，1000个RoI进行测试。使用FPN在COCO数据集上训练Fast R-CNN需要约10小时。

## **扩展：分割提议**

FPN同样可以扩展到实例分割的应用场景！

## **7. 结论**

我们提出了一个干净而简单的框架，用于在ConvNets内部构建特征金字塔。我们的方法比几个强大的基线和竞赛获胜者显示出了显著的改进。因此，它为特征金字塔的研究和应用提供了一个实用的解决方案，而不需要计算图像金字塔。最后，我们的研究表明，尽管深层ConvNets具有强大的表示能力以及它们对尺度变化的隐式鲁棒性，但使用金字塔表示对于明确地解决多尺度问题仍然至关重要。

## **References**

[1] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. Pyramid methods in image processing. RCA engineer, 1984.

[2] S. Bell, C. L. Zitnick, K. Bala, and R. Girshick. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In CVPR, 2016.

[3] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In ECCV, 2016.

[4] J. Dai, K. He, Y. Li, S. Ren, and J. Sun. Instance-sensitive fully convolutional networks. In ECCV, 2016.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In CVPR, 2005.

[6] P. Dollar, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. TPAMI, 2014.

[7] P.F.Felzenszwalb,R.B.Girshick,D.McAllester,andD.Ramanan. Object detection with discriminatively trained part-based models. TPAMI, 2010.

[8] G.GhiasiandC.C.Fowlkes.Laplacianpyramidreconstruction and refinement for semantic segmentation. In ECCV, 2016.

[9] S. Gidaris and N. Komodakis. Object detection via a multi-region & semantic segmentation-aware CNN model. In ICCV, 2015.

[10] S. Gidaris and N. Komodakis. Attend refine repeat: Active box proposal generation via in-out localization. In BMVC, 2016.

[11] R. Girshick. Fast R-CNN. In ICCV, 2015.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.

[13] B.Hariharan,P.Arbelaez,R.Girshick,andJ.Malik.Hypercolumns for object segmentation and fine-grained localization. In CVPR, 2015.

[14] K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask r-cnn. arXiv:1703.06870, 2017.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In ECCV. 2014.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR, 2016.

[17] S. Honari, J. Yosinski, P. Vincent, and C. Pal. Recombinator networks: Learning coarse-to-fine feature aggregation. In CVPR, 2016.

[18] T. Kong, A. Yao, Y. Chen, and F. Sun. Hypernet: Towards accurate region proposal generation and joint object detection. In CVPR, 2016.

[19] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In NIPS, 2012.

[20] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. Neural computation, 1989.

[21] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dolla虂r, and C. L. Zitnick. Microsoft COCO: Common objects in context. In ECCV, 2014.

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed. SSD: Single shot multibox detector. In ECCV, 2016.

[23] W. Liu, A. Rabinovich, and A. C. Berg. ParseNet: Looking wider to see better. In ICLR workshop, 2016.

[24] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.

[25] D. G. Lowe. Distinctive image features from scale-invariant keypoints. IJCV, 2004.

[26] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In ECCV, 2016.

[27] P. O. Pinheiro, R. Collobert, and P. Dollar. Learning to segment object candidates. In NIPS, 2015.

[28] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dolla虂r. Learning to refine object segments. In ECCV, 2016.

[29] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015.

[30] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun. Object detection networks on convolutional feature maps. PAMI, 2016.

[31] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In MIC- CAI, 2015.

[32] H. Rowley, S. Baluja, and T. Kanade. Human face detection in visual scenes. Technical Report CMU-CS-95-158R, Carnegie Mellon University, 1995.

[33] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015.

[34] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In ICLR, 2014.

[35] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In CVPR, 2016.

[36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.

[37] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. IJCV, 2013.

[38] R. Vaillant, C. Monrocq, and Y. LeCun. Original approach for the localisation of objects in images. IEE Proc. on Vision, Image, and Signal Processing, 1994.

[39] S. Zagoruyko and N. Komodakis. Wide residual networks. In BMVC, 2016.

[40] S. Zagoruyko, A. Lerer, T.-Y. Lin, P. O. Pinheiro, S. Gross, S. Chintala, and P. Dolla虂r. A multipath network for object detection. In BMVC, 2016. 10