# 薰风读论文：Feature Pyramid Network 详解特征金字塔网络FPN的来龙去脉

**薰风初入弦**
上海交通大学 计算机科学与技术博士在读

已关注

89 人赞同了该文章

> 这是薰风读论文的第4篇投稿.....

## 薰风说 Thinkings

FPN属于思想简洁但十分有效的模型，不需要过多赘述，这里主要是跟着论文的思路回顾目标检测的**骨干网络（Backbone）**。

注意，FPN并不是独立的目标检测算法，而只是一个**Backbone。**

FPN的主要贡献是：

借鉴了ResNet的跳接，结合了浅层特征和深层特征。（深层高等语义，低分辨率；潜层低等语义，高分辨率）
借鉴了SSD的检测策略，在不同分辨率的特征图上分别做预测。

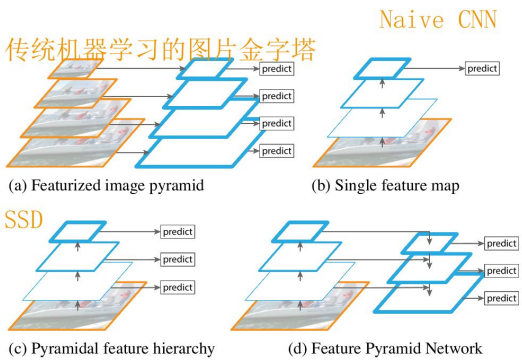


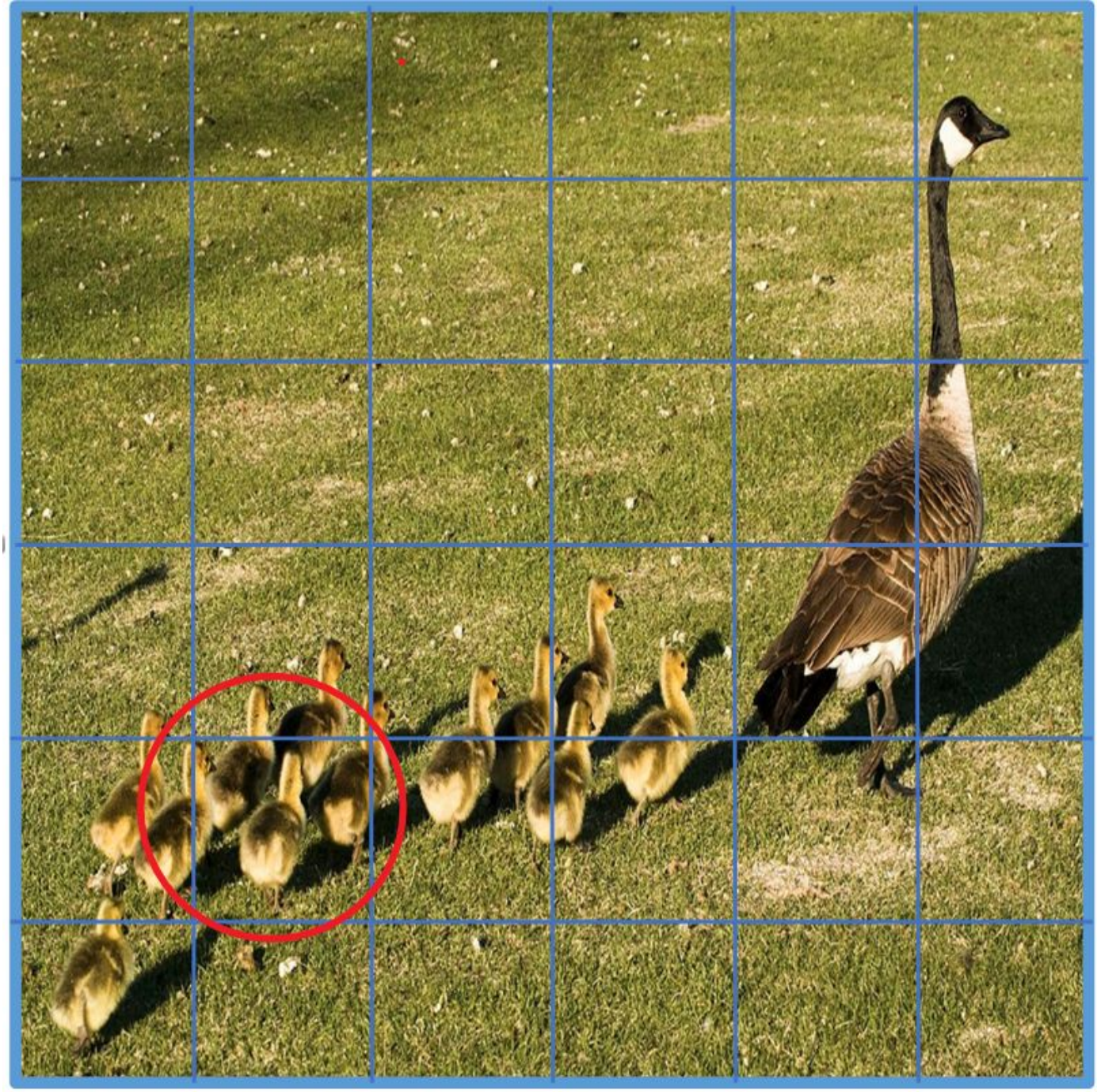

Figure 1. (a) Using an image pyramid to build a feature pyramid. Features are computed on each of the image scales independently, which is slow. (b) Recent detection systems have opted to use only single scale features for faster detection. (c) An alternative is to reuse the pyramidal feature hierarchy computed by a ConvNet as if it were a featurized image pyramid. (d) Our proposed Feature Pyramid Network (FPN) is fast like (b) and (c), but more accurate. In this figure, feature maps are indicate by blue outlines and thicker outlines denote semantically stronger features.

彩蛋：很多人可能没注意，这上图代表特征图的蓝色边框粗细不一，越粗表示特征语义更强

## 一、引言 Introduction

目标检测任务中，一个反复被提起，至今仍很棘手的问题，就是图片中的物体有大有小。而往往当目标物体过小时，识别会极其困难。

这里拿Yolo举个例子，Yolo（v1）把输入图片分成不同的网格，每个网格负责检测“中心落在”网格内部的物体，但每个网格只能预测一个物体，因此遇到如下图所示的情况时.....

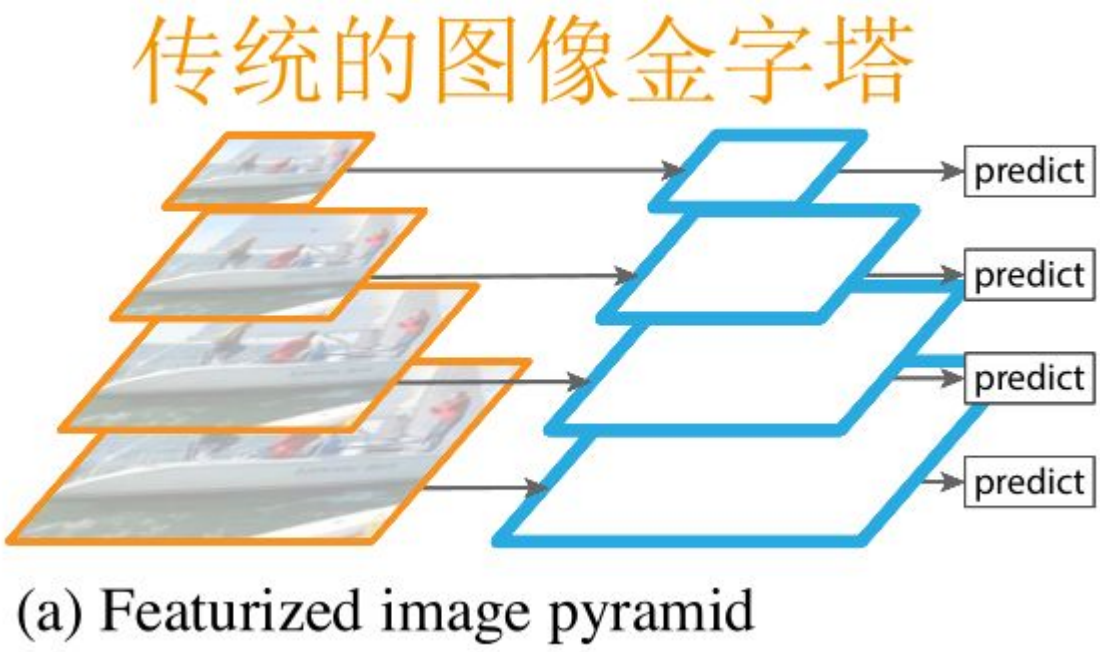


Yolo：咋回4鸭？（咋在一个网格返回4个鸭？）

识别小物体是目标检测算法的老大难问题，而特征金字塔网络（Featuer Pyramid Network,FPN）就是用来解决不同尺寸目标预测的。

也正因为老大难，之前很多论文都给出了自己的做法，在介绍FPN之前先梳理一下。

## 1. 在图像金字塔提特征 Featurized Image Pyramid




十分想当然的思路，既然你图片太小我预测不出来，那就把图片缩放到不同的大小，分别预测。

这是最直观的办法，自然也是最蠢的办法。

想象一下它的过程：

> 把一张图片提取特征预测→放大→提取特征预测→放大→提取特征预测……

这个过程中，每个特征提取/预测都是独立进行的！即使同一张图片的不同分辨率，模型之间也很难共享它们中间提取的特征。这让模型预测的过程费时费力，比如本来一个图片预测需要五秒，一个五级的图像金字塔耗时绝对不会少于25秒。

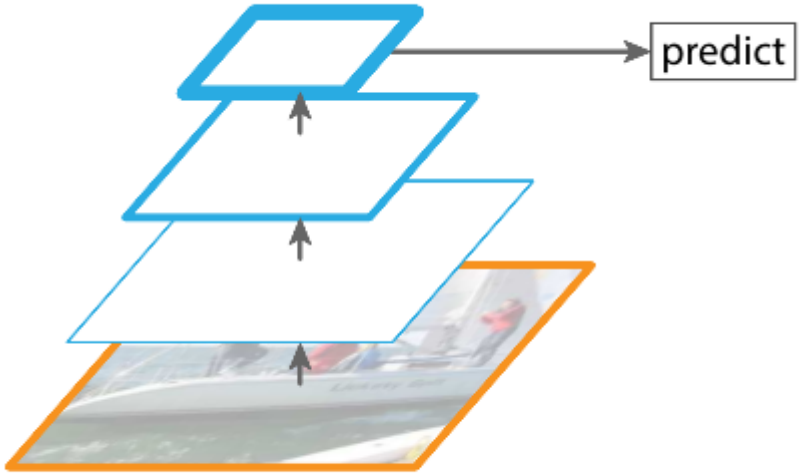
这还只是预测，训练模型更加耗时，因此出于这种考虑，**图像金字塔通常只能在预测时使用。**

这又导致了新的问题，即此时模型**本质上只是为了某种分辨率训练的**，只是强行被拿去检测别的分辨率，这样的效果自然不可能好。

## 2. 原始CNN Naïve CNN

现在我们都知道，若以表征学习（representation learning）的角度思考CNN，那么CNN实际上是有层次地学习图片的特征，即深度越深，则特征的语义越高级（从线到面，再到物体特征）

由于后续存在池化和降采样，浅层网络的特征图可以保留更多的分辨率，但是特征语义较为低级。




(b) Single feature map

因此，CNN只能寄希望于小物体的特征够在最后一层“存活下来”，但由于分辨率的损失，这显然是不切实际的。

但CNN这种层级计算特征如此高效，fast（er） R-CNN也是靠这种层级结构达到加速效果的。因为此时至少特征计算是“共享的”，不用像图像金字塔那样傻乎乎一个个算。

那么问题来了，层级计算的时候，有辣么多分辨率的特征，那我们何不全用上？

### 3. 金字塔型特征层级 Pyramidal feature hierarchy

实际上SSD就是“把不同分辨率特征”全用上的先驱之一。SSD会在不同分辨率的特征上直接预测，那么不就大物体小物体都能预测到了吗？

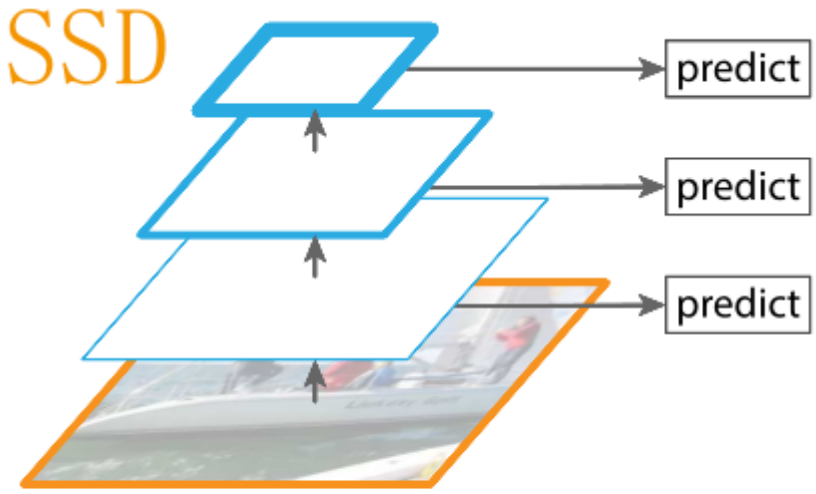
这想法很不错，但还没有那么好。SSD这么做有两个问题：

底层特征语义不够
最高分辨率并不高

正如之前分析的，实际上，底层特征的语义较低，携带信息的“有效性”略少，导致预测结果不佳。

SSD缓解上述问题的方法是，太底层的语义干脆不用不就好了？

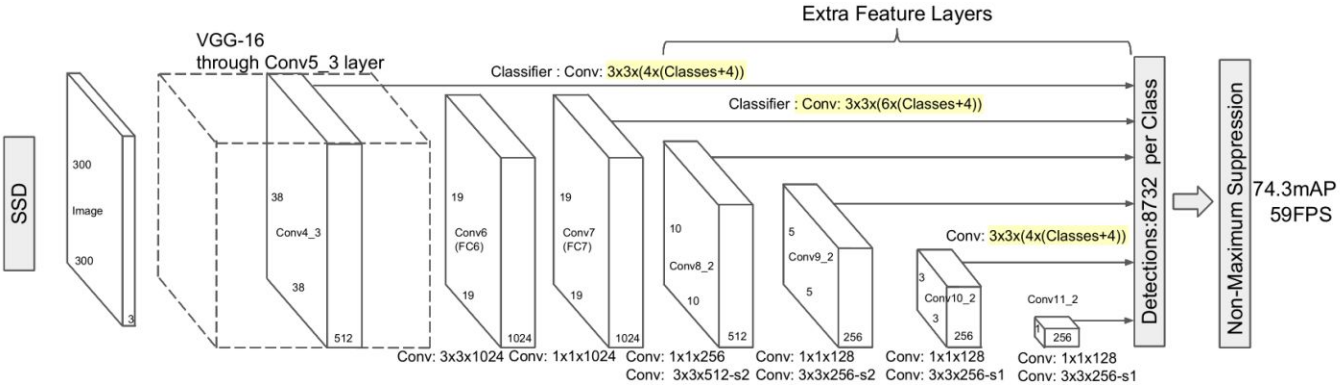
所以，你以为SSD是这样的：




(c) Pyramidal feature hierarchy

卖家秀

实际上是这样的：




买家秀

实际上，太浅层的特征SSD也根本没用上。这也不能怪SSD，因为技术的发展永远是循序渐进的，它能意识到可以在不同分辨率特征上检测已是不易。

分析到这，一个很自然的想法就出现了“我们是否可以结合深浅特征，兼顾分辨率与特征语义？”原来大家都会说不可能，直到一个叫ResNet的模型告诉我们什么叫跳接。

现在，正文终于可以开始了233

## 二、特征金字塔网络

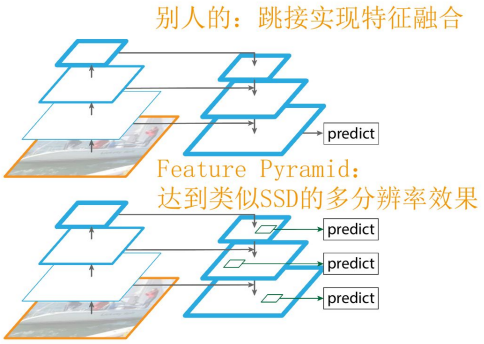



Figure 2. Top: a top-down architecture with skip connections, where predictions are made on the finest level (*e.g.*, [28]). Bottom: our model that has a similar structure but leverages it as a *feature pyramid*, with predictions made independently at all levels.

FPN是一种自顶向下路径和横向连接将低分辨率、语义强的特性与高分辨率、语义弱的特性结合起来的体系结构。

**FPN通过讲浅层的特征跳接到深层的特征，兼顾了分辨率与特征语义。**但实际上，这也不是FPN的首创，在对分辨率要求更高的语义分割问题中早有涉猎（如医学图像常用的U-Net）。

但FPN的独到之处在于**将深/潜层特征融合**与**多分辨率预测结合**了起来。

有了初步认识，接下来就是模型的细节了。

## 三、模型细节 Details

从图上可以看出，FPN可以分为三部分：

自底向上的部分
自顶向下的部分
连接两部分的跳接

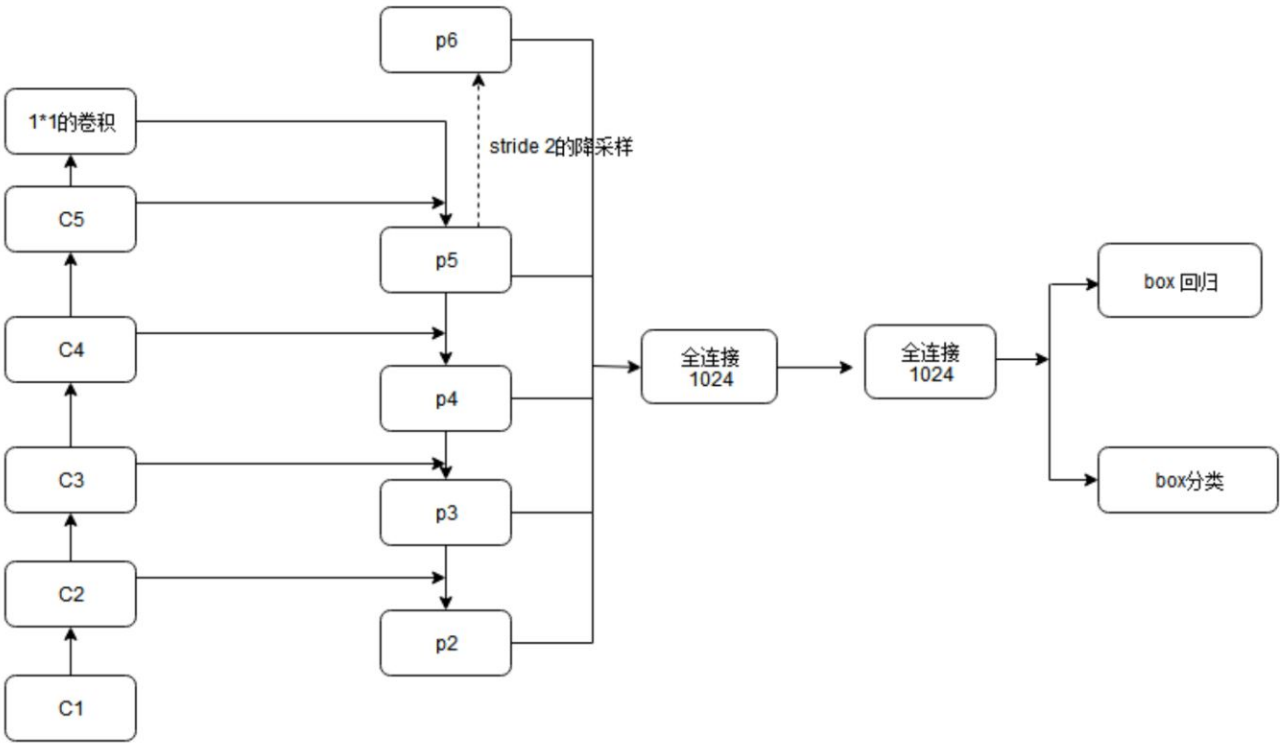
注意，实际上这个网络掰直了更像一个漏斗，可以理解为这里的**底=高分辨率，顶=低分辨率。**

**自底向上**

前馈Backbone的一部分，每一级往上用step=2的**降采样。**

输出size相同的网络部分叫一级(Stage)，举个例子:下图是fasterRCNN的网络结构，左列ResNet用每级最后一个Residual Block的输出，记为{C1,C2,C3,C4,C5}。

FPN用2~5级参与预测（因为第一级的语义还是太低了），而{C1,C2,C3,C4,C5}这几级等效于在图片上以{4,8,16,32}为步长提取特征。

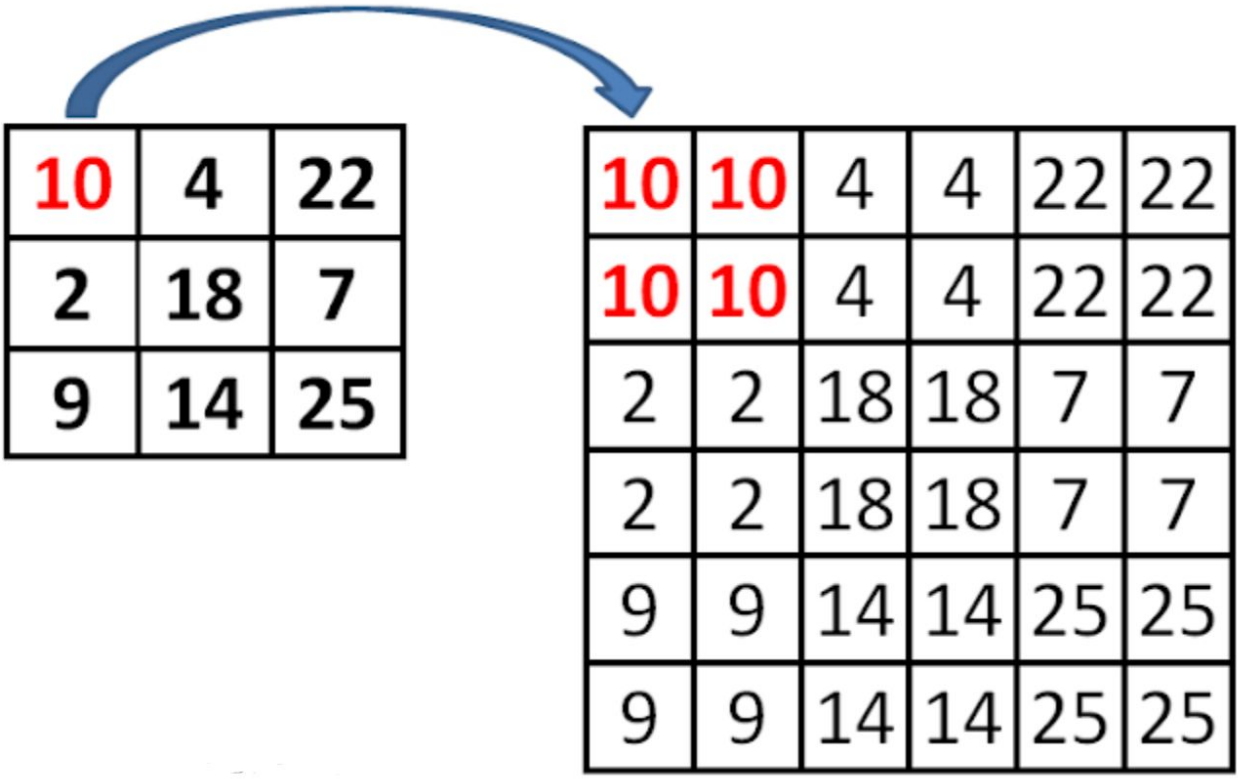



摘自博客：https://blog.csdn.net/xiamentingtao/article/details/78598027

**自顶向下&跳接**

对底向上的特征图一级级**上采样**，再加上底向上跳接来的高分辨低语义特征用于预测。

上采样的方法是简单的**最近邻插值**




一个最近邻插值的例子

其中跳接时前后维度不一致问题的解决方法采用了1*1conv，对跳接不了解的请看我的ResNet的跳接笔记。

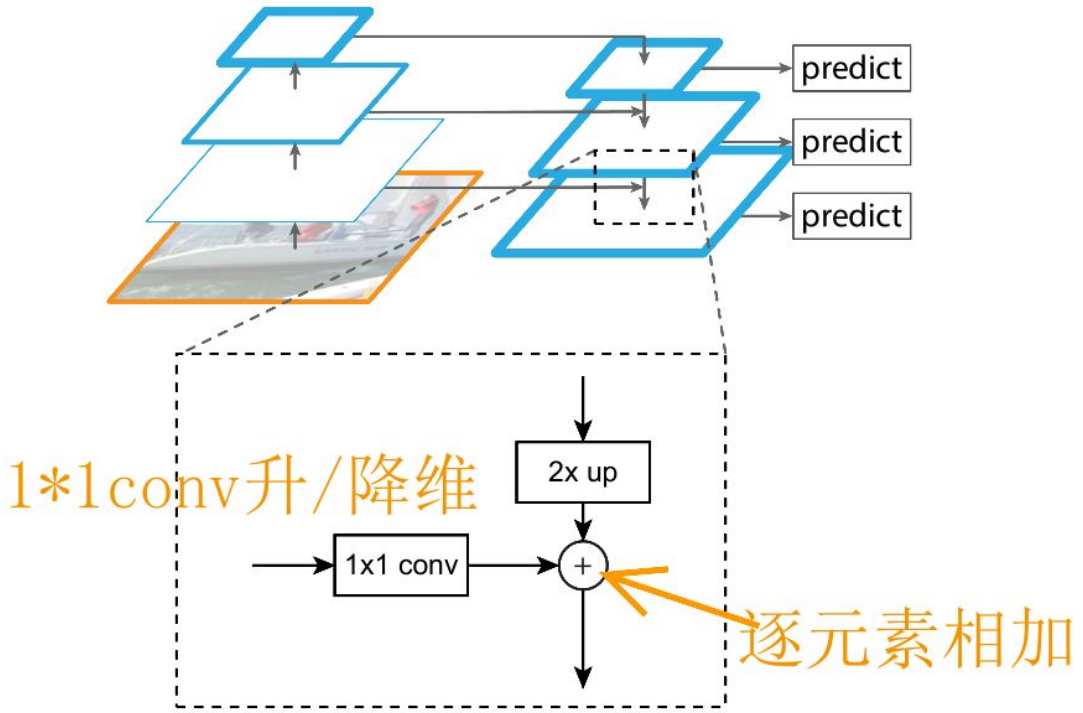



Figure 3. A building block illustrating the lateral connection and the top-down pathway, merged by addition.

通过跳接融合特征之后，还要进行一个3*3conv，以减轻最近邻插值带来的影响（因为周围的数字都比较相近，会出现重叠现象）

还是用那张RPN的结构，这部分的几级被称为{$P_2$,$P_3$,$P_4$,$P_5$,$P_6$}。其中2~5分别对应C的2~5，第六级是$P_5$直接降采样得到的。这用于检测的五个特征图的分辨率分别为：$\{32^2, 64^2, 128^2, 256^2, 512^2\}$

最后再看一眼网络图，一个完整的FPN架构就完成啦~