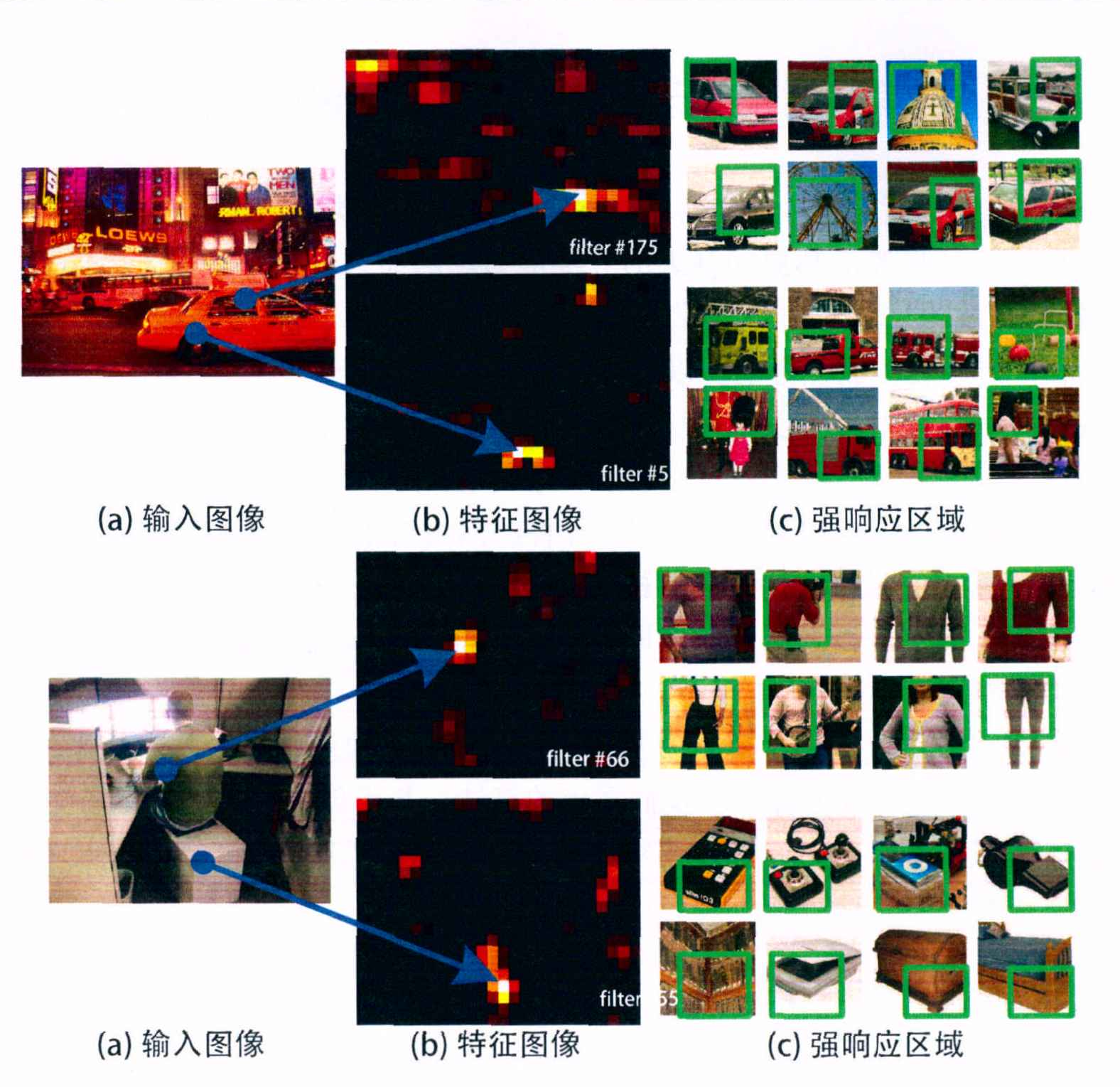
**Ⅲ.Method**

深度卷积网络在图像识别任务上显示出了强大的能力，并且具有很好的语义抽象能力，与这种能力一起出现的是超越以往方法的计算量。在一般的检测任务上，以R-CNN为代表的神经网络检测算法先依靠区域生成算法（region proposal）进行子区域筛选，大大减少了需要分类的子区域个数。但是使用深度神经网络对数千个区域进行逐一分类，所需要的计算代价和时间代价都限制了神经网络检测算法在学术研究和工业领域上的进一步发展。同时，在算法框架上来讲，这种对每个区域独立分类的做法也不是最优的。

本文针对深度卷积神经网络的检测算法，从本质上的计算复杂度出发，提出了基于空间金字塔池化层的算法，使得子区域分类任务之间可以共享特征，大大降低了算法的计算复杂度和实用中物体检测所需的时间。

**3.1卷积层与特征图像**

我们先考虑两个个拥有七层的分类网络（Zeiler and Fergus），ZP网络的前五个参数层是卷积层，池化层穿插其中。这些池化层也可以被认为是卷积层，因为都是以滑窗的形式实现。卷积层之后是全连接层，以及作为输出的N路softmax层，其中N是分类类别数。

上述的深度神经网络需要固定大小的输入图像，原因是全连接层依赖于长度固定的特征向量。而卷积层采用滑动窗口的方式处理图像，输入图片大小可以不一样，但是输出图像与输出图像的长宽比保持大致相同。这里输出的图片称为特征图像，这些图像不仅包括了响应的强度也包括了空间位置信息。

**图3.1** 可视化特征图像。（a）PASCAL VOL 2007数据集中的两张图像。（b）conv5层的某些特征图像。箭头指向区域为图像中最强相应和他们的位置。（c）ImageNet数据集中对这些滤波器产生强相应的图像。绿色框标记了产生最大响应的感受区域。

在图3.1中，我们将一些特征图像可视化了，它们是由conv5层的滤波器产生的。图3.1（c）展示了在ImageNet图片集中对这些滤波器响应最大的图片。我们可以看到一个滤波器可以被特定的语义激活，比如第55个滤波器可以被圆形物体激活，第66个滤波器可以被“^” 形物体激活，第118个滤波器可以被“V”物体激活。这些输入图像中的形状会在特征图像上相应位置被激活。

值得一提的是我们在生成图3.1的特征图像时没有固定输入图像的大小，深度卷积层生成这些特征图像的过程就像传统方法产生特征图像一样，在传统方法中，尺度不变特征变换（SIFT）特征或者图像块被密集提取（densely extract）并编码。编码的方式包括矢量量化（vector quantization），稀疏编码（sparse coding）。这些编码的特征包括图像特征，然后由词袋模型（Bag-of-Words，BoW）或者空间金字塔池化。所以，深度卷积特征也可以通过相似的方法池化。

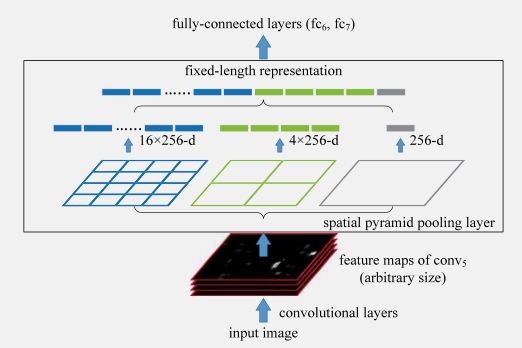
**3.2空间金字塔池化层**

卷积层接收任意大小的输入，并可以产生变大小的输出，而全连接层需要固定长度的特征向量。分类器（SVM/softmax）或全连接层需要固定大小的输入向量。这些向量可以通过词袋模型将特征池化。空间金字塔池化方法提升了BoW因为可以保留更多的空间信息，因此具有更好的性能。空间金字塔池化的每个区域的大小与图像大小成固定的比例，因此池化区域的数量是固定的，与图像大小无关。这种新方法与以往的滑窗型池化方式不同——滑窗的数量取决于输入尺寸。

为了使深度卷积神经网络可以接收任意大小的输入，我们将最后一个池化层（最后一个卷积层之后，例如conv5）替换为空间金字塔池化层，图3.2介绍了我们的方法。在每个空间区域之后，我们将每个滤波器得到的结果进行最大池化。空间金字塔池化的输出结果是k\*M维的向量，其中M是空间区域的总数，最后一层卷积层滤波器个数为k。这个固定维数的向量可以被用作全连接层的输入。

当使用空间金字塔池化时，输入的图像可以是任意尺寸的，不仅仅是允许任意长宽比也允许任意的尺度的图片输入。我们可以将输入图像伸缩到任意尺度，其中min（长，宽=180，224…）,并使用同一个深度网络。当输入的图像是不同尺度时，深度网络会使用相同大小的滤波器在提取到不同尺度的特征。

有趣的是，最大颗粒的金字塔级别使用一个区域覆盖整个图像，这本质上是一种“全局池化”操作，经研究发现，全局均值池化可以减小模型的大小及过拟合现象。在测试阶段所有的全连接层之后使用全局平均池化来提高精度，在弱监督物体识别任务中使用全局最大池化操作。这种全局池化的操作与传统方法中的词袋方法由类似的效果。



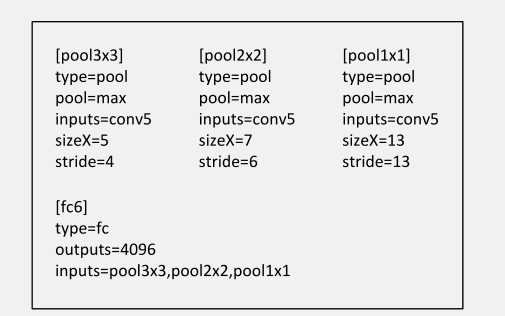
**图3.2** 包含空间金字塔池化层的网络结构。其中conv5是最后一层卷积层，256是conv5层的滤波器数量

**3.3训练网络**

理论上，上述的深度卷积网络可以通过标准的反向传播训练，而与输入图像尺寸关系不大。但在实际情况中用GPU实现的时候更倾向于在固定尺寸的输入图像上运行。接下来我们描述我们的训练方案，可以利用GPU处理单一尺寸的优势，同时保留空间金字塔池化的优点。

**3.3.1 单一尺度训练**

在我们之前的工作中，我们会首先从原始图片上截取大量固定尺寸的输入图像（224\*224），这些截取得到的图像可以用于图像增强。对于一个给定了尺寸的输入图像，我们可以预先计算用于空间金字塔池化的空间区域大小。例如，在conv5层的特征图像尺寸是a\*a，某一层金字塔池化的空间区域大小是n\*n，我们通过滑动窗口的形式来实现池化过程，窗口大小和步长都为 [a/n],方括号表示向下取整。通过最大池化得到了池化层后，我们将元素连接起来得到全连接层，图3.3展示了一个三级的金字塔池化方式。



**图3.3** 一个三级金字塔池化的例子，其中sizeX表示池化滑窗的大小，这个池化的是尺度大小为13\*13的特征图像，所以，三个池化滤波器大小为3\*3、2\*2、1\*1

我们使用单一尺寸来训练的主要目的是提高我们多尺寸训练的准确度，是我们进行多尺度训练的基础。

**3.3.2 多尺度训练**

我们的多尺度金字塔池化网络是为了能够用在多种尺度的图片上，为了解决训练过程中图像尺度大小变化的问题，我们首先要提前固定图像的尺寸大小，这次我们使用180\*180的尺寸而非224\*224。我们先截取到224\*224尺寸的图片，然后将图片的大小压缩到180\*180，前后两个图片的差别在于分辨率也就是尺寸而非内容。为了让网络能够接收180\*180的输入尺寸，我们实现了另一个固定大小输入的网络（180\*180），这个网络在conv5后的特征图像大小为10\*10，我们使用类似于上一节的办法，池化区域的大小和步长依然是[a/n]，这样一来这个网络的全连接层大小和224\*224图片大小的全连接层长度相等，这样，不同尺度的图片就可以有相同长度的特征向量，就实现了多尺度训练。

为了减少运算损耗，我们在一个网络上训练完成对同一尺度图片集的一次遍历，然后我们切换到另一个尺度的图片集完成一次遍历，这个过程中训练参数保持不变，然后进行如此反复的学习过程。在训练过程中，我们发现，这种训练方法的收敛速度和传统方式的差不多。

多尺度训练的目的是模拟变化的输入尺寸，利用现有的经过优化的固定尺寸的网络来训练另一个尺度的图片。包括上述的两个尺度的图像训练，我们同时使用了尺度大小随机处于[180，224]图片来进行训练，每个尺度各完成一次遍历迭代。我们在下文的实验部分会具体介绍。

上述的单尺度和多尺度的训练方法仅用于训练网络。所以在测试阶段，我们可以将SPP-net用于各种尺度的图片了。