用于视觉识别的深度卷积网络中的空间金字塔池化

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun

摘要

现有的深度卷积神经网络（CNN）需要一个固定尺寸（如 224×224）的输入图像。这一要求是" 人为的"，可能会降低任意尺寸/尺度的图像或子图像的识别精度。在这项工作中，我们为网络配备了另一种池化策略，即" 空间金字塔池化"，以消除上述要求。新的网络结构，称为 SPP-net，可以生成一个固定长度的表示，而不考虑图像的大小/尺度。金字塔池化对物体变形也很稳健。有了这些优势，SPP-net 应该在总体上改善所有基于 CNN 的图像分类方法。在 ImageNet 2012 数据集上，我们证明了尽管各种 CNN 架构设计各不相同，但 SPP-net 能够提高它们的准确度。在 Pascal VOC 2007 和 Caltech101 数据集上，SPP-net 使用单一的全图像表示法而不进行微调就能达到最先进的分类结果。

SPP-net 的力量在物体检测方面也很显著。使用 SPP-net，我们只计算一次整张图像的特征图，然后在任意区域（子图像）汇集特征，生成固定长度的表示，用于训练检测器。这种方法避免了重复计算卷积特征。在处理测试图像时，我们的方法比 R-CNN 方法快 24-102 倍，同时在 Pascal VOC 2007 上取得了更好或相当的准确度。

在 2014 年 ImageNet 大规模视觉识别挑战赛（ILSVRC）中，我们的方法在所有 38 个团队中物体检测排名第二，图像分类排名第三。这份手稿还介绍了为这次比赛所做的改进。

1 简介

我们正在见证视觉界的快速、革命性的变化，主要是由深度卷积神经网络（CNN）[1] 和大规模训练数据的可用性 [2] 引起的。基于深度网络的方法最近在图像分类 [3]、[4]、[5]、[6]、物体检测 [7]、[8]、[5]、许多其他识别任务 [9]、[10]、[11]、[12]，甚至非识别任务方面的技术水平上有了很大提高。

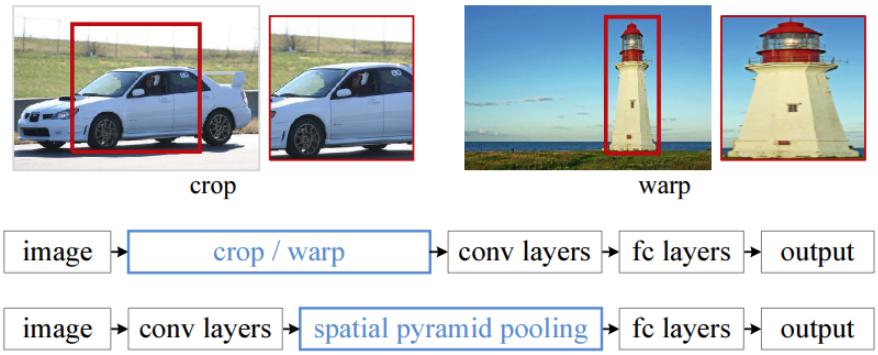

图 1: 顶部：裁剪或扭曲以适应固定尺寸。中间：一个传统的 CNN。底部：我们的空间金字塔集合网络结构。

然而，在 CNN 的训练和测试中存在一个技术问题：普遍的 CNN 需要一个固定的输入图像尺寸（如 224×224），这就限制了输入图像的长宽比和比例。当应用于任意尺寸的图像时，目前的方法大多通过裁剪 [3]、[4] 或扭曲 [13]、[7] 将输入图像适配到固定尺寸，如图1（顶部）所示。但是裁剪的区域可能不包含整个物体，而扭曲的内容可能导致额外的的几何变形。由于内容的损失或失真，识别的准确性可能会受到影响。此外，当物体的尺度变化时，预先定义的尺度可能不适合。固定输入尺寸忽略了涉及尺度的问题。

那么，为什么 CNN 需要一个固定的输入大小呢？一个 CNN 主要由两部分组成：卷积层和后面的全连接层。卷积层以滑动窗口的方式运作，并输出代表激活的空间排列的特征图（图2）。事实上，卷积层不需要固定的图像尺寸，可以生成任何尺寸的特征图。另一方面，全连接层根据其定义需要有固定大小/长度的输入。因此，固定尺寸的约束只来自全连接层，它存在于网络的更深阶段。

在本文中，我们介绍了空间金字塔池化（SPP）[14], [15] 层来消除网络的固定尺寸约束。具体来说，我们在最后一个卷积层的顶部添加一个 SPP 层。SPP 层汇集特征并产生固定长度的输出，然后将其送入全连接的层（或其他分类器）。换句话说，我们在网络层次结构的较深阶段（卷积层和全连接层之间）进行一些信息” 聚合”，以避免一开始就进行裁剪或扭曲。图1（底部）显示了引入 SPP 层后网络结构的变化。我们称这种新的网络结构为 SPP-net。

空间金字塔集合 [14], [15]（俗称空间金字塔匹配或 SPM[15]），作为语

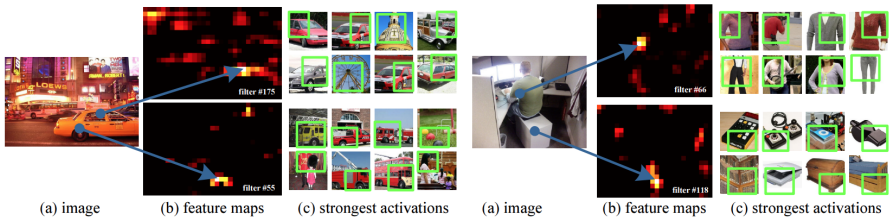

图 2: 特征图的可视化。(a) Pascal VOC 2007 中的两幅图像。(b) 一些 conv_5 过滤器的特征图。箭头表示最强的反应和它们在图像中的相应位置。(c) 相应过滤器反应最强的 ImageNet 图像。绿色的矩形标志着最强反应的感受区。

料袋（BoW）模型 [16] 的延伸，是计算机视觉中最成功的方法之一。它将图像划分为从精细到粗糙的层次，并将局部特征聚集在其中。在最近 CNN 盛行之前，SPP 长期以来一直是分类（如 [17]、[18]、[19]）和检测（如 [20]）的领先和竞赛获奖系统的关键组成部分。然而，SPP 还没有在 CNN 的背景下被考虑。我们注意到，SPP 对于深度 CNN 来说有几个显著的特性。1）SPP 能够无视输入尺寸而产生一个固定长度的输出，而之前的深度网络 [3] 中使用的滑动窗口池化不能；2）SPP 使用多级空间分块，而滑动窗口池只使用单一的窗口大小。多级池化已被证明对物体变形具有鲁棒性 [15]；3）由于输入尺度的灵活性，SPP 可以汇集在不同尺度上提取的特征。通过实验我们表明，所有这些因素都提升了深度网络的识别精度。

SPP-net 不仅可以从任意大小的图像/窗口中生成表征用于测试，而且还允许我们在训练过程中输入不同大小或比例的图像。用不同尺寸的图像进行训练可以增加尺度不变性并减少过拟合。我们开发了一种简单的多尺寸训练方法。对于一个接受可变输入尺寸的单一网络，我们用共享所有参数的多个网络来近似它，而这些网络中的每一个都用固定的输入尺寸进行训练。在每个历时中，我们用一个给定的输入尺寸训练网络，并在下一个历时中切换到另一个输入尺寸。实验表明，这种多尺寸训练和传统的单尺寸训练一样收敛，并能带来更好的测试精度。

SPP 的优势与具体的 CNN 设计是正交的。在 ImageNet 2012 数据集的一系列控制变量实验中，我们阐明了对于现有的四个典型模型 [3],[4],[5]（或它们的变种），SPP 相较于对应的无 SPP 版本，对所有模型都有所提升。这些架构有不同的卷积核数量/大小、步长、深度或其他设计。因此，我们

有理由猜测，SPP 应该能改善更复杂（更深更大）的卷积结构。SPP-net 在 Caltech101[21] 和 Pascal VOC 2007[22] 上也显示了最先进的分类结果，并只使用了单一完整的图像表示，没有进行微调。

SPP-net 在物体检测方面也显示出巨大的优势。在领先的物体检测方法 R-CNN[5] 中，候选窗口的特征是通过深度卷积网络提取的。这种方法在 VOC 和 ImageNet 数据集上都表现出了显著的检测精度。但是 R-CNN 中的特征计算是很耗时的，因为它对每张图像的数千个扭曲区域的原始像素反复应用深度卷积网络。在本文中，我们表明我们可以在整个图像上只运行一次卷积层（不管窗口的数量如何），然后通过 SPP-net 在特征图上提取特征。这种方法产生的速度比 R-CNN 快一百多倍。请注意，在特征图（而不是图像区域）上训练/运行检测器实际上是一个更流行的想法 [23], [24], [20], [5]。但是 SPP-net 继承了深度 CNN 特征图的力量，同时也继承了 SPP 在任意窗口大小上的灵活性，这就导致了出色的准确性和效率。在我们的实验中，基于 SPP-net 的系统（建立在 R-CNN 管道上）计算特征的速度比 R-CNN 快 24-102 倍，同时具有更好或相当的准确性。利用 EdgeBoxes[25] 最近的快速提议方法，我们的系统处理一幅图像（包括所有步骤）只需 0.5 秒。这使得我们的方法对现实世界的应用很实用。

本稿件的初步版本已经发表在 ECCV 2014 上。基于这项工作，我们参加了 ILSVRC 2014 的比赛 [26]，在所有 38 个团队中，物体检测排名第 2，图像分类排名第 3（均为只提供数据的赛道）。我们为 ILSVRC 2014 做了一些修改。我们表明，SPP 网络可以提升各种网络的深度和规模（第 3.1.2-3.1.4 节），超过无 SPP 的对应网络。此外，在我们的检测框架的驱动下，我们发现对具有灵活位置/大小的窗口的特征图进行多视图测试（第 3.1.5 节）可以提高分类精度。本稿件也提供了这些修改的细节。

2 带有空间金字塔池化的深度网络

2.1 卷积层和特征图

考虑流行的七层架构 [4, 6]。前五层是卷积层，其中一些接着池化层。这些池化层也可以被认为是” 卷积”，因为它们使用的是滑动窗口。最后两层是全连接的，以 N-way softmax 作为输出，其中 N 是类别的数量。

上述的深度网络需要一个固定的图像大小。然而，我们注意到，对固定尺寸的要求只是由于全连接层需要固定长度的向量作为输入。另一方面，卷

积层接受任意大小的输入。卷积层使用滑动卷积核，其输出与输入的长宽比大致相同。这些输出被称为特征图 [1]–它们不仅涉及反应的强度，而且还涉及它们的空间位置。

在图2中，我们可视化了一些特征图。它们是由 conv_5 层的一些卷积核生成的。图2(c) 显示了 ImageNet 数据集中这些卷积核最强的激活图像。我们看到一个卷积核可以被一些语义内容所激活。例如，第 55 个卷积核（图2，左下）被圆形激活最多；第 66 个卷积核（图2，右上）被 $\wedge$ 形激活最多；第 118 个卷积核（图2，右下）被 $\vee$ 形激活最多。输入图像中的这些形状（图2(a)）激活了相应位置的特征图（图2中的箭头）。

值得注意的是，我们生成图2中的特征图时没有固定输入尺寸。这些由深度卷积层生成的特征图类似于传统方法中的特征图 [27], [28]。在这些方法中，SIFT 向量 [29] 或图像斑块 [28] 被密集提取，然后进行编码，例如，通过向量量化 [16]、[15]、[30]、稀疏编码 [17]、[18] 或 Fisher 内核 [19]。这些编码后的特征由特征图组成，然后通过词袋（BoW）[1] 或空间金字塔 [2, 3] 进行汇集。类似地，深度卷积特征也可以用类似的方式进行汇集。

2.2 空间金字塔池化层

卷积层接受任意大小的输入，但它们产生的输出是大小可变的。分类器（SVM/softmax）或全连接层需要固定长度的向量。这种向量可以通过将特征汇集在一起的词袋（BoW）方法 [1] 产生。空间金字塔集合 [2, 3] 改进了 BoW，因为它可以通过在局部空间仓中集合来保持空间信息。这些空间仓的大小与图像大小成正比，因此无论图像大小如何，仓的数量是固定的。这与之前的深度网络 [4] 中的滑动窗池化形成对比，后者的滑动窗数量取决于输入大小。

为了对任意大小的图像采用深度网络，我们用空间金字塔池化层取代了最后一个池化层（例如在最后一个卷积层之后的 pool_5）。图3说明了我们的方法。在每个空间仓中，我们汇集每个卷积核的响应（在本文中我们使用最大池化）。空间金字塔池化的输出是一个 kM 维向量，仓的数量表示为 M（k 是最后一个卷积层中的卷积核数量）。固定维度的向量是全连接层的输入。

通过空间金字塔集合，输入图像可以是任何尺寸。这不仅允许任意的长宽比，还允许任意的比例。我们可以将输入图像的大小调整为任何比例（例如，$\min(w, h) = 180, 224, \ldots$），并应用相同的深度网络。当输入图像处于

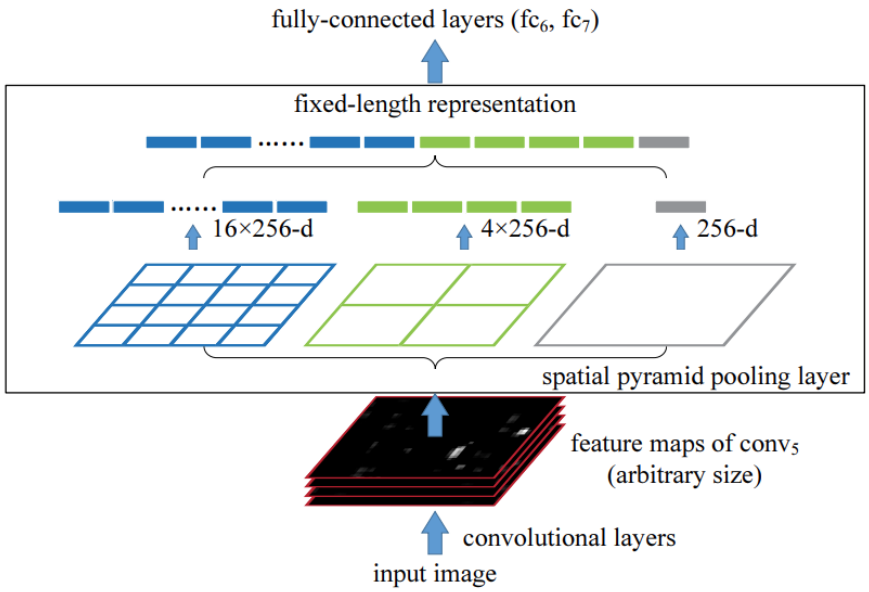

图 3: 一个具有**空间金字塔池化层**的网络结构。这里 256 是最后一个卷积层 $conv5_5$ 的卷积核数量。

不同尺度时，网络（具有相同的卷积核大小）将提取不同尺度的特征。尺度在传统方法中起着重要的作用，例如，SIFT 向量通常是在多个尺度上提取的 [29]，[27]（由斑块和高斯卷积核的大小决定）。我们将表明，尺度对深度网络的准确性也很重要。

有趣的是，最粗糙的金字塔层仅有一个覆盖整个图像的单仓。这实际上是一种“全局池化”操作，在几个同时进行的工作中也有研究。在 [31]、[32] 中，全局平均池化被用来减少模型的大小，也减少了过拟合；在 [33] 中，全局平均池化被用在所有 fc 层之后的测试阶段，以提高准确性；在 [34] 中，全局最大池化被用于弱监督的物体识别。全局池化操作对应于传统的 Bag-of-Words 方法。

2.3 网络训练

理论上，上述网络结构可以用标准的反向传播算法 [1] 来训练，而不考虑输入图像的大小。但在实践中，GPU 的实现（如 cuda-convnet[3] 和 Caffe[35]）最好是在固定的输入图像上运行。接下来我们将介绍我们的训练

方案，该方案利用了这些 GPU 实现，同时仍然保留了空间金字塔池的行为。

大一大小训练

和以前的工作一样，我们首先考虑一个网络从图像中获取固定尺寸的输入（224×224）。裁剪的目的是为了增加数据。对于一个给定尺寸的图像，我们可以预先计算出空间金字塔池化所需的仓尺寸。考虑 conv_5 之后的特征图，其大小为 $a \times a$（例如 13×13）。对于一个有 $n \times n$ 个仓金字塔层级，我们将这个池化层级实现为滑动窗口池化，其中窗口大小 $win = \lceil a/n \rceil$，$stride = \lfloor a/n \rfloor$，$\lceil \cdot \rceil$ 和 $\lfloor \cdot \rfloor$ 表示向上取整和向下取整的操作。通过一个 l 级的金字塔，我们实现了 l 个这样的层。下一个全连接的层（fc_6）将把 l 个输出连接起来。图4显示了 *cuda-convnet* 风格 [4] 的 3 级金字塔池化 $(3 \times 3, 2 \times 2, 1 \times 1)$ 的配置实例。

```
[pool3x3]          [pool2x2]          [pool1x1]
type=pool          type=pool          type=pool
pool=max           pool=max           pool=max
inputs=conv5       inputs=conv5       inputs=conv5
sizeX=5            sizeX=7            sizeX=13
stride=4           stride=6           stride=13

[fc6]
type=fc
outputs=4096
inputs=pool3x3,pool2x2,pool1x1
```

图 4: 一个 *cuda-convnet* 风格 [4] 的 3 级金字塔池的例子。这里 sizeX 是池化窗口的大小。这个配置是针对一个 conv_5 的特征图大小为 13×13 的网络，所以 $\text{pool}_{3\times3}, \text{pool}_{2\times2}$ 和 $\text{pool}_{1\times1}$ 层将分别有 $3 \times 3, 2 \times 2$ 和 1×1 个仓。

我们进行单一尺度训练的主要目的是为了实现多级池化行为。实验表明，这是提高准确率的一个原因。

多大小训练

我们的拥有 SPP 的网络预计将应用于任何尺寸的图像。为了解决训练中不同图像尺寸的问题，我们考虑一组预定义的尺寸。我们考虑两种尺寸。180×180，以及 224×224。我们没有裁剪一个较小的 180×180 区域，而是将上述 224×224 区域的大小调整为 180×180。因此，两种比例的区域只在分辨率上有区别，而在内容/布局上没有区别。为了让网络接受 180×180 的输入，我们实现了另一个固定大小的输入（180×180）网络。在这种情况下，$conv5$ 之后的特征图大小为 $a \times a = 10 \times 10$。然后我们仍然使用 $win = \lceil a/n \rceil$ 和 $stride = \lfloor a/n \rfloor$ 来实现每个金字塔池化的层级。这个 180 网络的空间金字塔池化层的输出与 224 网络的固定长度相同。因此，这个 180 网络的每一层的参数与 224 网络的参数完全相同。换句话说，在训练过程中，我们通过两个共享参数的固定尺寸网络来实现变化输入尺寸的 SPP 网络。

为了减少从一个网络（如 224）切换到另一个网络（如 180）的开销，我们在一个网络上训练每个完整的纪元，然后在下一个完整的纪元切换到另一个网络（保持所有权重）。这样反复进行。在实验中，我们发现这种多规模训练的收敛率与上述单规模训练相似。

我们的多规模训练的主要目的是模拟不同的输入尺寸，同时仍然利用现有的经过优化的固定尺寸实现。除了上述两个尺寸的实现，我们还测试了一个使用 $s \times s$ 作为输入的变体，其中 s 是在每个历时中通过从 $[180, 224]$ 中随机均匀取样得到。我们在实验部分报告了这些变体的结果。

请注意，上述单/多尺寸的解决方案仅用于训练。在测试阶段，在任何尺寸的图像上应用 SPP-net 都是直接的。

3 使用 SPP-NET 进行物体检测

深度网络已经被用于物体检测。我们简要回顾一下最近最先进的 R-CNN 方法 [7]。R-CNN 首先通过选择性搜索 [20] 从每个图像中提取大约 2000 个候选窗口。然后，每个窗口中的图像区域被扭曲成一个固定的大小（227×227）。一个预先训练好的深度网络被用来提取每个窗口的特征。然后在这些特征上训练一个二分类 SVM 分类器进行检测。R-CNN 产生的结果具有令人信服的质量，大大超过了以前的方法。然而，由于 R-CNN 将深度卷积网络重复应用于每张图像的约 2000 个窗口，因此很耗时。特征提取是测试中的主要时间瓶颈。

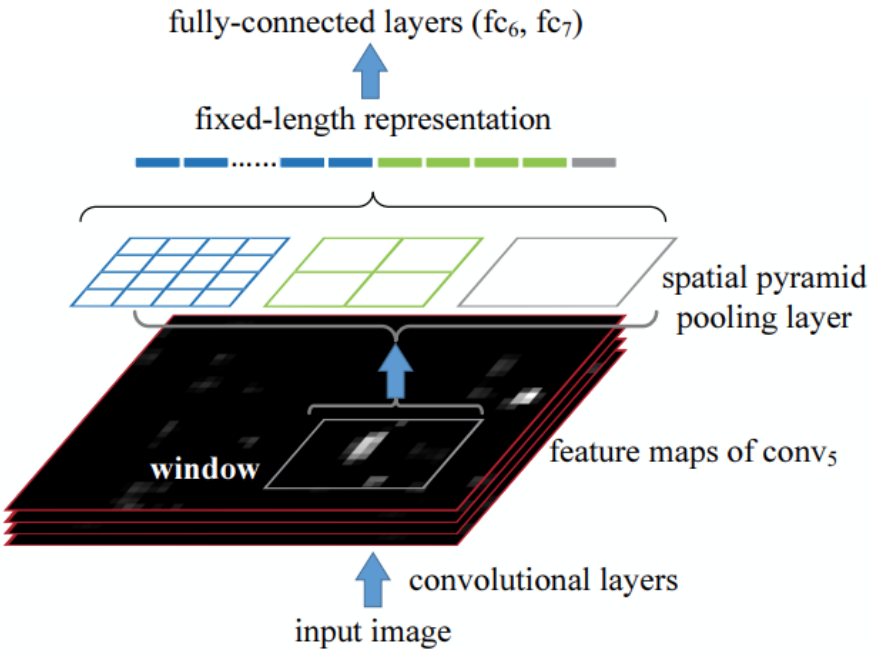

图 5: 将特征图上的任意窗口的特征汇集。特征图是由整个图像计算得到的。池化是在候选窗口中进行的。

我们的 SPP-网络也可用于物体检测。我们只从整个图像中提取一次特征图（可能是多尺度的）。然后，我们在特征图的每个候选窗口上应用空间金字塔池化，以池化这个窗口的固定长度表示（见图5）。因为耗时的卷积只应用一次，所以我们的方法运行速度可以快上几个数量级。

我们的方法是从特征图的区域中提取窗口的特征，而 R-CNN 则是直接从图像区域中提取。在以前的工作中，可变形部分模型（DPM）[23] 从 HOG[24] 特征图的窗口提取特征，选择性搜索（SS）方法 [20] 从编码的 SIFT 特征图的窗口提取。Overfeat 检测方法 [5] 也是从深度卷积特征图的窗口中提取，但需要预先定义窗口大小。相反，我们的方法可以从深度卷积特征图的任意窗口中提取特征。

3.1 检测算法

我们使用选择性搜索的" 快速" 模式 [20]，为每幅图像生成约 2,000 个候选窗口。然后我们调整图像的大小，使 $\min(w,h)=s$，并从整个图像中提取特征图。我们暂时使用 ZF-5 的 SPP-net 模型（单尺寸训练）。在每个候选窗口中，我们使用一个 4 级空间金字塔（1×1，2×2，3×3，6×6，共 50 个仓）来汇集特征。这就为每个窗口生成了 12,800 维（256×50）的表示。这些表征被提供给网络的全连接层。然后我们在这些特征上为每个类别训练一个二元线性 SVM 分类器。

我们对 SVM 训练的实现遵循 [20]，[7]。我们使用 ground-truth 窗口来生成正样本。负样本是那些与正样本窗口最多重叠 30% 的样本（用交集比（IoU）衡量）。任何负面样本如果与另一个负面样本重叠超过 70%，则被删除。我们应用标准的难负例挖掘 [23] 来训练 SVM。这个步骤迭代了一次。为所有 20 个类别训练 SVM 需要不到 1 小时。在测试中，分类器被用来对候选窗口进行评分。然后我们对打分后的窗口使用非最大抑制 [23]（阈值为 30%）。

我们的方法可以通过多尺度特征提取来改进。我们调整图像大小，使 $\min(w,h)=s\in S=\{480,576,688,864,1200\}$，并计算每个尺度的 conv_5 的特征图。结合这些尺度的特征的一个策略是逐个通道汇集它们。但我们根据经验发现，另一种策略可以提供更好的结果。对于每个候选窗口，我们选择一个单一的尺度 $s\in S$，使得该尺度的候选窗口的像素数最接近 224×224。然后我们只使用从这个尺度中提取的特征图来计算这个窗口的特征。如果预设的尺度足够密集，并且窗口近似于正方形，我们的方法大致相当于将窗口的大小调整为 224×224，然后从中提取特征。尽管如此，我们的方法只需要从整个图像中计算一次特征图（在每个尺度上），而不管候选窗口的数量如何。

我们还按照 [7] 对我们的预训练网络进行了微调。由于我们的特征是由任何大小的窗口的 conv_5 特征图汇集而成的，为了简单起见，我们只对完全连接的层进行微调。在这种情况下，数据层接受 conv_5 之后的固定长度的集合特征，然后是 fc6,7 层和一个新的 21 路（一个额外的负类别）fc8 层。fc8 的权重是用 $=0.01$ 的高斯分布初始化的。我们将所有的学习率固定为 1e-4，然后调整为所有三个层的 1e-5。在微调过程中，正样本是那些与真实窗口 $[0.5,1]$ 重叠的样本，而负样本是 $[0.1,0.5]$。在每个迷你批中，25% 的样本是阳性的。我们用 1e-4 的学习率训练 250k 个迷你批，然后用 1e-5 训

练 50k 个迷你批。因为我们只对 fc 层进行微调，所以训练速度非常快，在 GPU 上大约需要 2 个小时（不包括预先缓存特征图，这需要 1 个小时）。同样按照 [7]，我们使用边界盒回归来对预测窗口进行后处理。用于回归的特征是来自 conv_5 的集合特征（与 [7] 中使用的 pool_5 特征相对应）。用于回归训练的窗口是那些与 ground-truth 窗口重叠至少 50% 的窗口。

Appendices

在附录中，我们描述了一些实施细节：

减均值。 224×224 的训练/测试图像通常通过减去每个像素的平均值进行预处理 [3]。当输入图像是任意尺寸时，固定尺寸的平均图像就不能直接适用。在 ImageNet 数据集中，我们将 224×224 的平均图像扭曲成所需的大小，然后减去它。在 Pascal VOC 2007 和 Caltech101 中，我们在所有的实验中使用恒定的平均数（128）。

池化仓的实现 在应用网络时，我们使用以下实现方式来处理所有的仓。将 conv_5 特征图的宽度和高度（可以是全图或一个窗口）表示为 w 和 h。对于一个有 $n \times n$ 个仓的金字塔层级，第 (i, j) 个仓在 $\left[\lceil \frac{i-1}{n} w \rceil, \lfloor \frac{i}{n} w \rfloor\right] \times \left[\lceil \frac{j-1}{n} h \rceil, \lfloor \frac{j}{n} h \rfloor\right]$。直觉上，如果需要四舍五入，我们在左边/顶部边界采取向下取整操作，在右边/底部边界采取向上取整操作。

将一个窗口映射到特征图 在检测算法（以及对特征图的多视图测试）中，在图像域中给出了一个窗口，我们用它来裁剪已经被多次下采样的卷积特征图（例如 conv_5）。所以我们需要在特征图上为窗口对齐。

在我们的实现中，我们将一个窗口的角点投射到特征图中的一个像素上，使得这个角点在图像域中最接近该特征图像素的感受野的中心。由于所有卷积层和池化层的填充，这种映射很复杂。为了简化实施，在部署过程中，我们为卷积核大小为 p 的层填充了 $\lceil p/2 \rceil$ 个像素。这样，对于一个中心位于 (x', y') 的响应，它在图像域中的有效接受域中心位于 (Sx', Sy')，其中 S 是之前所有步长的乘积。在我们的模型中，ZF-5 在 conv_5 的 S 为 16，Overfeat-5/7 在 $\text{conv}_{5/7}$ 的 S 为 12。对于给定图像域中的窗口，我们通过

$x' = \lceil x/S \rceil + 1$ 以及 $x' = \lfloor x/S \rfloor - 1$ 来分别对左上角和右上角的边界进行投影。如果填充不是 $\lceil p/2 \rceil$，我们需要诶 x 加上适当的偏移。

References

[1] Josef Sivic and Andrew Zisserman. "Video Google: A text retrieval approach to object matching in videos". In: *Computer Vision, IEEE International Conference on*. Vol. 3. IEEE Computer Society. 2003, pp. 1470–1470.

[2] Kristen Grauman and Trevor Darrell. "The pyramid match kernel: Discriminative classification with sets of image features". In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 2. IEEE. 2005, pp. 1458–1465.

[3] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories". In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. IEEE. 2006, pp. 2169–2178.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25 (2012), pp. 1097–1105.

[5] Ross Girshick et al. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.

[6] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *European conference on computer vision*. Springer. 2014, pp. 818–833.