深度学习在物体检测与识别中的应用

熊郁文

3130000829

orpine@zju.edu.cn

摘要

在大规模图像数据中进行精确的物体检测与识别是计算机视觉中一个非常重要的问题。从 2012 年以来，深度学习在图像分类问题方面取得了令人瞩目的成绩。而在物体检测方面，自 2014 年开始，Ross Girshick 在 CVPR 2014 上发表的论文中提出了 R-CNN，成功的将深度学习应用到物体检测中，并取得了远好于传统方法的结果。本文将对 2014 年以来，使用卷积神经网络在物体检测问题上取得良好结果的算法进行简要介绍，包括 R-CNN [1]，SPP-net [2]，Fast R-CNN [3] 以及 Faster R-CNN [4]。

1. 引言

物体的检测一直是计算机视觉中比较重要和困难的问题。在很多领域有广泛的应用，比如安保方面的的行人检测与跟踪，交通领域的车辆检测等等。因此这个问题在计算机视觉，模式识别与机器学习等领域都是非常活跃的研究方向。传统的方法一直是基于 SIFT 或者 HOG 等手工设计的特征，在此基础上进行一些工作。然而这些方法已经被证实效果并不理想，SegDPM [5] 在 Pascal VOC 2010 数据集上的表现也仅有 40.4% 的 mAP，远远称不上令人满意。卷积神经网络（CNN）是由 20 世纪 80 年代由 Yann LeCun 提出的一种神经网络，最初用于手写数字识别 [6]，其特殊的结构使其对二维的图像数据输入能得到非常好的结果。但由于当时的机器性能限制，CNN 未能得到更多的关注。直到 2012 年，Geoffrey Hinton 的博士生 Alex Krizhevsky 利用一个 8 层的 CNN 在 ILSVRC 2012 的图像分类任务上一举取得了远超以往算法的成绩 [7]。点燃了学术界对于深度卷积神经网络在计算机视觉方面的研究的热情，但在当时，深度卷积神经网络只在一定程度上解决了对整张图像的分类问题，而更加困难的对图片中多个物体进行定位与识别的问题还尚未解决。本文将对深度卷积神经网络在物体检测与识别方面的研究进展做简要介绍。第二节将介绍将卷积神经网络拓展到物体检测问题上并在 PASCAL VOC 竞赛中取得突破性进展的的第一种方法——R-CNN，R-CNN 证明了 CNN 提取出来的『Rich feature』效果远远好于传统的方法，第三节将介绍在 R-CNN 基础上进行改进，去除了 R-CNN 的冗余计算使得速度与效果都得到提升的 SPP-net 与 Fast R-CNN，第四节将介绍 R-CNN 的最新进展——Faster R-CNN，这种算法进一步提升了 Fast R-CNN 的速度，基本能够做到实时的物体检测。文章的最后将对这几种算法做一个总结，并对未来进行展望。

2. R-CNN 介绍

R-CNN 由 Ross Girshick 等人于 CVPR 2014 上提出。在这之前，大家已经意识到 CNN 在图像分类方面的威力，而如何才能将 CNN 应用在物体检测方面这是一个问题。与图像分类问题不同之处在于，物体的检测问题需要定位出物体的位置。R-CNN 将 Pipeline 分为三部分来解决用 CNN 进行物体检测的问题。第一个部分试图生成与待检测的物体类别无关（category-independent）的 region proposal，有别于传统的 sliding

window 做法，这些 region proposal 就是检测目标的“候选”，后续的检测识别将只会在这些 region proposal 上进行。第二部分是用于提取图像特征的深度卷积神经网络。第三部分则是一组用来对 CNN 提取出的特征进行类别分类的线性 SVM。这样就成功的将物体检测问题变成了在 bounding box 中的图像分类问题。

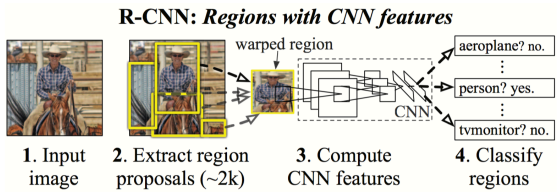

图 1. R-CNN 框架图

2.1. R-CNN 设计

第一个问题是，如何快速的得到 region proposal，得益于前人的工作，我们有许多现成的方法可以使用，包括 objectness，selective search，category-independent object proposals，constrained parametric min-cuts (CPMC)，multi-scale combinatorial grouping 等等。作者在 R-CNN 中使用的是 selective search。

作者使用了 Alex Krizhevsky 在 ILSVRC 2012 中所使用的 CNN 网络 AlexNet [7]，对于每个输入，去掉最后一层 softmax 分类层而提取出了一个 4096 维的特征向量。需要注意的是，对于 AlexNet 这样有全连接层的网络，输入图片的大小是有严格限制的（作者使用的网络版本是 227x227），因此对于每个 region proposal，作者使用了 wrapping 的方式即将每个 region proposal 缩放成规定大小的正方形来作为输入。

最后，基于时间和内存的考量，作者选择了线性 SVM 作为最后的分类器。

2.2. R-CNN 的训练与测试

作者提出，可以将利用 ILSVRC 2012 年数据训练的 AlexNet 作为一个提取图片的特征的黑盒，但是为了让处理分类问题的网络在处理物体识别问题上也工作的很好，我们需要对其进行 fine-tuning。在 Fine-tuning 阶段，作者在 AlexNet 权值的基础上，将最后一层 1000 路 Softmax 改为了 PASCAL VOC 所需的 21 类，对于 selective search 获取的 region proposal，与 ground-truth bounding box 的 IoU $\geqslant 0.5$ 的 region proposal 被认为是 positive sample，否则是背景。

之后，利用 Fine-tuning 过后的 CNN 来提取特征（提取出 Softmax 层之前的 4096 位向量），利用提取出来的特征来训练 SVM，作者对每一类检测物体训练一个 SVM。在训练中，作者将与任意 ground-truth bounding box 的 IoU（Intersection of Union）$\geqslant 0.5$ 的 region proposal 认为是正样本，与所有 ground-truth bounding box 的 $\leqslant 0.3$ 的 region proposal 认为是负样本，其余丢弃。作者指出在这里通过验证集选择一个合适的阈值是十分重要的，对 mAP 结果会有较大的影响。同时，由于内存问题，作者对于 SVM 采用了 hard negative mining 的训练方法，不过考虑到这不是本文介绍重点，而且作者在后续文章中放弃了 SVM 作为分类器，在此就不再赘述了。

在测试时,对于每一张输入图片,作者会使用 selective search 来提取两千个左右的 region proposal，wrap 后将其送入 CNN 中提取出 feature，再利用 SVM 来进行分类确定每个 region proposal 的类别（是否为某一类物体或者是背景）。最后再利用 non-maximum suppression 去除掉重叠的 bounding box。

作者另外还实现了 bounding box regression，这在本文以及后续文章的实验中被证实是一种十分有效的减少 mislocation 的做法，通过训练数据（ground truth 中标记的位置与 feature vector）训练出一个线性回归模型，试图对 selective search 所找到的 region proposal 的位置进行修正（输出 bounding box 中心点以及长、宽的修正量）。这种做法对 mAP 能有 3-4% 的提升，具体的转换可在作者主页上的 supplement 中找到。

2.3. R-CNN 的总结

R-CNN 被认为是 CNN 在物体检测领域的开山之作。而且它让人们意识到特征在计算机视觉领域的重要性，这也从论文标题看得出来，标题并没有强调『R-CNN』这一方法，而是强调『Rich feature hierarchies』。在测试结果上，R-CNN 相较于以往的方法确实有相当大的提高（在 PASCAL VOC 2010 数据集上 mAP 从

40.4% 提升至 53.7%），但其对于数据处理过于粗糙，仍有提升的空间（后面将介绍的几种方法也证实了这一点）。对于每张输入图片，selective search 都会提取出 1~2 千个不等的 region proposal 要通过 CNN 进行 feature 提取，这些 region proposal 有许多是互相重叠的，因此有大量冗余的计算，这也导致了 R-CNN 的速度非常之慢，提取出来的 feature 也要暂存到硬盘上，中间文件可能会有上百 GB 之多，这都是 R-CNN 的缺陷。

3. SPP-net 与 Fast R-CNN 介绍

SPP-net 由 Kaiming He 等人于 ECCV 2014 上提出。Fast R-CNN 则由 Ross Girshick 自己在 ICCV 2015 上提出，鉴于他们俩较为相似，这里将它们放在一起介绍。

SPP-net 是基于 R-CNN 所做的一些改进，Fast R-CNN 是在 SPP-net 上进一步改进之后的结果。相比 SPP-net 有更快的速度与更好的结果。接下来将主要介绍他们与 R-CNN 所不同的部分。

3.1. SPP-net 的改进

Kaiming He 等人首先注意到了 R-CNN 的不足之处——第一，由于 CNN 的全连接层对 feature vector 的长度有要求，对于不满足指定大小的 region proposal，无论是 warpping 或是 cropping 的做法，都会在一定程度上损失原有数据的信息，不利于正确识别图像；第二，对于同一张图片，region proposal 有上千个，其中很多都是重叠的，会有大量的重复计算。

值得一提的是，CNN 中的卷积层对于输入实际上是没有任何约束的，也就是说卷积层可以处理任意尺度的输入，是之后的全连接层限制了这一点。因此，为了解决这个问题，Kaiming He 等人在 R-CNN 中引入了空间金字塔池化（Spatial Pyramid Pooling (SPP) layer）。其本质就是，对于一个 feature map 做多个尺度的池化。由图2中可以看到，对于从 AlexNet 的第五层卷积层输出的 feature map，对它们做 1x1，2x2，4x4 的池化，由于 conv5 有 256 个 filter，所以最后的结果会是一个 256 维的向量、4 个 256 维的向量和 16 个 256 维的向量。最后将他们连接起来成为全连接层的输入。此时我们可以发现，这种处理方式对于图片的大小是没有要求的，因为中间这一层池化，对于任意尺度的图片都可以处理成定长的 feature vector（图中所给的例子是 (1+4+16)X256=5376 维）然后在输入给全连接层。

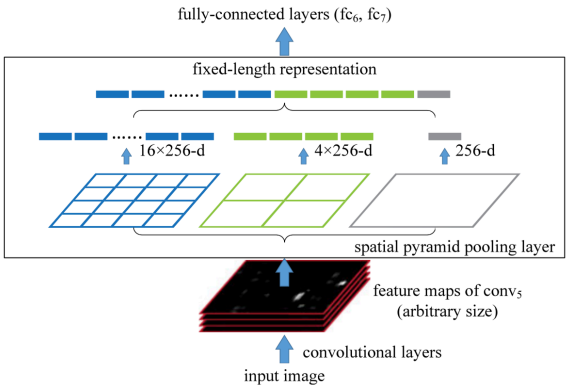

图 2. SPP layer 结构

SPP layer 的引入，成功解决了第一个问题。在 test 阶段，对于 selective search 得到的 region proposal，如何不进行 feature map 的重复计算？我们现在已经有能力直接对整张图片计算 feature map，那么实际上就可以通过计算直接将 region proposal 在原图片中的位置转换为在 conv5 输出的 feature map 中的位置（这一步转换公式较为复杂）。这样一来原本较为耗时的 CNN 的计算就转变成了四个坐标值的计算，大大减少了计算时间。在文章中作者将原始图片放大成若干大小，选取 region proposal 在缩放后的图像中大小最接近 224x224 的图片。最后的时间消耗上相对于 R-CNN 有数十倍的提升 (在 PASCAL VOC 2007 的测试集上单张图片 0.382s vs 14.46s)。

3.2. Fast R-CNN 的改进

Fast R-CNN 一文指出 R-CNN 有三个明显的缺陷：第一，训练时 pipeline 被分为了三个阶段，对于使用 selective search 提取好的 region proposal，需要先利用 CNN 提取特征，然后再训练分类器（R-CNN 与 SPP-net 都使用了 SVM），最后训练 bounding box regression，十分麻烦和复杂。也因此导致了空间开销

十分之大，这里是可以考虑进行简化的；第二，训练时太耗时间和内存及硬盘空间；第三，测试阶段速度太慢。而 SPP-net 在加速了 R-CNN 的同时，并没有解决第一个缺陷，同时还引入了别的问题：无法更新 SPP layer 之前的网络层的权值，对于更深的网络来说，这样将会限制网络的精度。

而 Fast R-CNN 则将训练的三个阶段统一起来，提高速度的同时也去除了额外磁盘空间的限制。大致框架如图3。

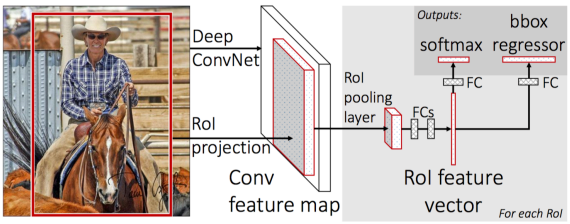

图 3. Fast R-CNN 结构

作者在文章中提出了 RoI pooling layer，这实际上是 SPP layer layer 的一种简化版本，模仿 SPP layer，我们可以将任意大小的图片 max pooling 成 $H \times W$ 的大小 (文章中为 7×7)，而 RoI pooling layer 的意义在于，对于提取到一半的 feature map，给定一个 region proposal，我们可以算出它在 feature map 中的位置，并通过 RoI pooling 得到一个固定大小的 feature map 如 7×7 将其送入后续的 layer，同时 RoI pooling layer 也可以进行反向传导，更新 RoI pooling layer 前面的网络层权值。这样一来，CNN 只需处理每张图片一次，再利用 RoI pooling layer 处理每个 region proposal 即可。同时作者指出 R-CNN 和 SPP-net 在 fine-tuning 时效率是十分低下的，原因是训练样本都是来自许多不同的图片，这样会导致训练的输入太大。而 Fast R-CNN 每次只使用两张图片，每张图片提取 64 个 region proposal 来构成一个大小为 128 的 mini-batch。由于一张图片只需要过一次 CNN，这种方式会比 R-CNN 中对于每个 region proposal 都需要 warpping 并且过一遍 CNN 的做法速度大大提升。

Fast R-CNN 还将分类器替换为 softmax，并把它与 bounding box regressor 统一起来同时进行 joint traning，提出了 smooth-L1 作为 bounding box regression 的 loss function，认为这种函数形式将会提高鲁棒性，在全连接层还利用了 SVD 分解来减少需要训练的参数进行加速。而在测试时，之前所描述的 SPP-net 会对图像进行多个尺度的缩放然后从缩放的图片中选取大小合适的 region proposal，而 Fast R-CNN 则采取了将所有图片缩放到一个固定尺度的做法。前者比后者可以比后者得到稍好一点的结果 (mAP 增加 1% 左右)，但后者的速度会快很多，而且对于规模更大的网络 (如 VGG16)，现有 GPU 的 12G 显存限制无法做到多尺度的缩放。

Fast R-CNN 在网络上做得多处改进使得其比 SPP-net 又快上数倍 (在 PASCAL VOC 2007 的测试集上单张图片 0.32s vs 2.3s)，在 PASCAL VOC 2007 上的 mAP 也达到了 68.8%。

作者同时还在文章中给出一些其他的 insight，如更多的 data 将会带来更好的结果，region proposal 的数量需要正好合适而不是越多越好等等，这对我们更加深入的理解物体检测领域的一些细节有所帮助，但不是本文介绍重点。

4. Faster R-CNN 介绍

Faster R-CNN 是 Kaiming He 等人与 Ross Girshick 合作在微软研究院所做的工作。在 NIPS 2015 上发表。

Fast R-CNN 成功的将 CNN，classifier 与 bounding box regressor 统一，这三部分都利用了 GPU 加速运行，此时利用 selective search 提取 region proposal 的部分就变成了瓶颈，因为这部分是跑在 CPU 上的。能否将提取 region proposal 的工作也转移到 GPU 上运行，将整个框架统一起来做到真正的 end-to-end 呢？答案是肯定的，这就是 Faster R-CNN 主要所做的工作，通过定义一种新的网络结构，Faster R-CNN 做到了又快又好的提取 region proposal。

为了让 region proposal 的提取也能在 GPU 上运行，作者提出了一种叫做 Region Proposal Networks (RPN) 的网络结构。这是一个全部由卷积层构成的网络，作者在实验中使用了 ZF model [8] 的前 5 层和 VGG [9] 的前 13 层。作者指出，对于卷积网络输出的 feature map，也可以用于提取 region proposal。作者在

feature map 上使用了一个小型的网络进行滑动窗口的扫描。提取出一个低维向量 (对于 ZF 和 VGG 两种不同规模的网络，长度分别为 256 和 512)。最后将这个向量送入两个全连接层。这一部分如图4所示 对于每个

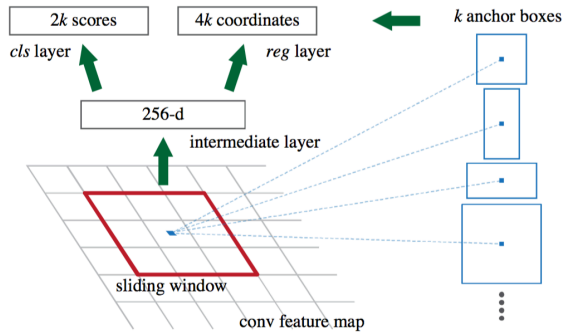

图 4. RPN 结构

窗口，作者进行了多个尺度与高宽比的采样，生成 k 个 anchor (对于从窗口中采样得到的 region proposal，作者在论文中称为 anchor)，文中使用了三种尺度与三种高宽比，最后对于每个窗口将会产生 9 个 anchors，因此 k=9。在训练时，作者将与某个 ground-truth box 的 IoU>0.7 的 anchor 标记为 positive，与所有的 groun-truth box 的 IoU 都 <0.3 的 anchor 标记为 negative，其余的 anchor 以及超过了图片边界的 anchor 将被丢弃。RPN 最后的两个全连接层 cls layer 与 reg layer 分别会产生 2k 个 score (为了实现上的方便，作者采用了 2 类的 softmax 层来给出第 k 个 anchor 是物体/背景的判断) 以及 4k 个 coordinates (即第 k 个 anchor 的 bounding box regression 结果)。值得注意的是，Fast R-CNN 对于任意大小的 region proposal 都是用相同的 bounding box regressor，而此处有 k 个 regressor 对应不同大小的 anchor。

为了做到图片在网络中的『 1-pass 』，作者还将 RPN 与后面处理 anchor 的 Fast R-CNN 的卷积层进行了权值共享，同时进一步降低参数个数，加快速度，提高鲁棒性。这里有很多种做法，作者在论文中提出了一种『 4 步做法 』，先独立训练一个 RPN，将得到的 anchor 送入 Fast R-CNN 中训练 Fast R-CNN，再利用 Fast R-CNN 的网络参数初始化 RPN，固定卷积层参数微调 RPN 的全连接层。最后再微调 Fast R-CNN 的全连接层。这样一来，两部分网络共享了卷积层的参数，而又有各自不同的全连接层参数。之后，作者更在 GitHub 的开源代码中给出了 end-to-end 的训练方式，这种方式简化了训练步骤，同时效果并未减弱。

因为 RPN 可以利用 GPU 进行加速，Faster R-CNN 的速度又上升了一个台阶。将 selective search 替换为 RPN 后，耗费时间可以达到原来的 1/10，在作者的实验中，使用 VGG + Fast R-CNN 时每秒可处理五张图片。换成规模小一点的 ZF 网络则每秒可以处理 17 张图片，已经达到距离实时处理仅有一步之遥。

5. 总结

本文简要介绍了近两年来物体检测领域的主流算法，鉴于篇幅原因，有许多细节未能详细介绍，如果感兴趣的话可以去查阅相关论文。从 14 年 Ross Girshick 提出 R-CNN 到如今，也才不过短短两年多的时间，计算机视觉在物体检测领域的进展可用突飞猛进来形容。从 R-CNN 一路走来，SPP-net，Fast R-CNN，Faster R-CNN。速度越来越快，检测精度越来越高，让我们又一次感受到了深度学习的能力，同时 Kaiming He 等人也从别的角度入手，在 2015 年提出 Deep Residual Network，将 CNN 网络提升至前所未有的 152 层 [10]，今年更是加深到 1000 层以上 [11]。利用更好的 feature 配合 Faster R-CNN 取得了 ILSVRC 2015 年的图像检测任务冠军。同时 Ross Girshick 在 Fast R-CNN 工作中所做的实验也发现，对于深度学习来说，增加数据对 mAP 的提升仍有帮助，让我们不禁细想深度学习的极限在何处。在 PASCAL VOC 数据集上，mAP 从 40% 到 50% 到 60% 再到 70%，速度也从非实时做到了准实时，接下来的提升可能会越来越困难，从 R-CNN 一脉相承的这一套方法仍然存在一个缺陷，那就是没有用到图片中的 context 信息来获取先验知识，如果能够整合这一点，想必又能做到一个提升，我们期望在不久的将来能看到 mAP 提升到 95% 甚至更高的那天。

References

[1] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate

object detection and semantic segmentation. In *Computer Vision and Pattern Recognition*, 2014.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Computer Vision–ECCV 2014*, pages 346–361. Springer, 2014.

[3] Ross Girshick. Fast r-cnn. In *International Conference on Computer Vision (ICCV)*, 2015.

[4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

[5] Sanja Fidler, Roozbeh Mottaghi, Raquel Urtasun, et al. Bottom-up segmentation for top-down detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3294–3301. IEEE, 2013.

[6] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[8] Matthew D Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. In *Computer Vision – ECCV 2014*, pages 818–833. Springer International Publishing, Cham, September 2014.

[9] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. *arXiv preprint arXiv:1603.05027*, 2016.