

基于拟物框重排序和共享卷积特征的物体检测

刘云

南开大学

摘要 目前检测效果最好的物体检测算法一般先产生大量拟物框，以试图不区分类别地找出图片中所有的物体；然后，在对这些拟物框进行分类和位置修正，从而确认图片中物体的准确位置。因此，物体检测算法的准确率往往受限于拟物框的数量和质量。本文提出了一种深度学习算法，对 EdgeBoxes [13] 算法产生的大量拟物框进行重新排序，并选择得分较高的少数拟物框进行物体检测。由于拟物框重排序和物体检测都需要对图片进行卷积，本文提出的算法能够使得这两个过程共享图片的卷积特征，减少对图片的一次卷积过程，从而节省了运行时间。实验结果表明，本文的算法取得了较好的检测准确率和检测速率。

1 引言

物体检测是计算机视觉领域一个由来已久的话题，鉴于它的广泛用途，历来受到研究者们的重视。在早期的物体检测算法中，通常先产生百万级数量的滑动窗口，以遍历图片中的每个位置，然后对这些滑动窗口提取特征、分类，代表作是 DPM 模型 [5]。之后，研究者提出了拟物性采样的概念，即从图片找出一些方框，以试图包含图片中所有完整的物体但不区分它们的类别；然后对这些拟物框提取特征，并进行分类和回归，即从产生的拟物框中找出确实包含物体的并判断物体类别。与基于滑动窗口的方法相比，由于拟物框的数量（通常几千至几百）远远小于滑动窗口，且其准确性远高于滑动窗口，所以基于拟物框的物体检测方法在速度和准确率上都远超过基于滑动窗口的检测算法。目前，拟物性采样加后续分类的物体检测模式已经被大多数研究者所采用。

观察 EdgeBoxes [13] 产生的拟物框，我们发现随着数量的增加，拟物框能够更好地覆盖更多的物体。但是，过多的拟物框，对物体检测又提出了挑战。为了解决这个矛盾，我们计划对 EdgeBoxes 产生的大量拟物框进行重排序和位置微调，然后将得分高的少量拟物框输入下一步检测系统中，进行分类。

近几年来，随着深度学习的蓬勃发展，一系列计算机视觉的问题用深度学习取得了很大的进展，比如图片分类、语义分割、边缘检测等，物体检测也是其中之一。2014 年，Girshick 等提出了著名的 RCNN [7] 框架，随后又发展出 Fast RCNN [6] 和 Faster RCNN [9] 算法。RCNN 和 Fast RCNN 用非深度学习方法

Selective search [11] 生成拟物框，其中，RCNN 对每个拟物框进行卷积，提取卷积特征；Fast RCNN 则先对整张图片进行卷积，然后用其提出的 ROI 池化方法从每个拟物框中提取卷积特征。由于 Fast RCNN 只对整张图片做了一次卷积，因此缩短了执行时间。Faster RCNN 不再使用 Selective search 来产生拟物框，而是在神经网络中先生成拟物框，再做检测，并且生成拟物框和检测部分共享图片的卷积特征。

受到 Faster RCNN 的启发，本文产生拟物框部分和检测部分也共享了图片的卷积特征。首先，拟物框重排序部分，用卷积特征为每个 EdgeBoxes [13] 产生的拟物框提取特征，通过卷积和全连接层为每个拟物框计算一个分数，取得分最高的数百个拟物框输入检测网络中。仍然用之前为图片提取的卷积特征来为这些挑选出来的拟物框计算特征，最后根据这些特征进行分类。实验结果表明，我们的新算法取得了比 Faster RCNN 更好的检测效果。

2 相关工作

本部分简要回顾了与本文密切相关的一些拟物性采样和物体检测算法，其中，对于物体检测算法，我们重点简述了基于深度学习的一些方法。

虽然拟物性采样被提出仅有几年的时间，但是在物体检测领域却取得了辉煌的成就，甚至被应用到一些与图片内容相关的领域上，比如基于内容的图片压缩、语义分割等。Alexe 等人 [1] 提出了一种融合多种特征的拟物性采样方法，以更快更好检测出图片中所有的完整物体。张自鸣等人 [12] 先提取简单的梯度特征，再用级联 SVM 进行评分。程明明等人 [2] 等人将 [12] 中提取方向梯度的过程进行了简化，并将卷积运算转化成 8×8 的二进制运算，从而大大提高了运算速度。但是 [2] 和 [12] 类似，精度上都不是很理想。Edgeboxes [13] 是一种利用边缘来搜索拟物框的算法，由于图片中的物体往往具有完整的边缘，所以这种方法取得了较好的结果，尤其当产生拟物框的数量足够多时。Selective search [11] 是先将图片进行超分割得到很多超像素，然后提取了多种特征，根据这些特征将认为相似的区域进行递归的合并。如上文所示，根据 Edgeboxes [13] 的性质，本文选择了它作为我们的基础方法先来产生大量的初始拟物框。

在基于深度学习的物体检测方面，Girshick 等于 2014 年提出了著名的 RCNN [7] 框架，将 Selective search [11] 生成的方框从图片中截取出来并缩放到一定大小，输入到神经网络，通过前向传导为每个拟物框生成维数相等的卷积特征；最后训练 K 个分类器 (K 为物体类别) 来将这些特征分为 $K + 1$ 类（加 1 是因为

多了背景)。RCNN 的缺点是每个拟物框都需要在深度神经网络中进行一次前向传递，而使用的方框近 2000 个，这就使得 RCNN 的速度很慢；并且它的模块分散，不便于拓展。于是，Girshick 又提出了 Fast RCNN [6] 方法，使每张图片只在神经网络中进行一次前向传导，并提出 ROI 池化的方法为每个方框得到一个固定维度的特征；其次，Fast RCNN 将特征提取、物体分类和位置回归都统一到一个框架中，使系统的训练、调试和开发都更加方便。尽管如此，由于 RCNN 和 Fast RCNN 所用的产生拟物框的 Selective search 方法处理一张图片需要 2 秒钟的时间，并且精度仍然满足不了要求，使得拟物性采样成为了物体检测的瓶颈。在此背景下，Shaoqing Ren [9] 提出了 Faster RCNN，利用图像经过卷积得到的特征计算拟物性，并且计算拟物性和检测部分共享相同的卷积特征，这样相当于将计算拟物框的时间从 2 秒缩短到 10 毫秒。Faster RCNN 生成的拟物框的质量要比 Selective search 高很多，用几百个拟物框就能达到比 Selective search 的 2000 个更好的效果，因此 Faster RCNN 的物体检测效果比 Fast RCNN 好。但是，Faster RCNN 进行拟物性采样时采用了类似穷搜的方式产生很多方框，然后进行筛选。所以仍然有改进的空间。

我们的方法先用 Edgeboxes [13] 产生大量拟物框，然后设计网络进行重新排序，最后将得分高的拟物框输入检测网络进行分类。和 Faster RCNN 类似，我们的方法也能使拟物性采样和物体检测共享卷积特征。以下部分将详细介绍我们的方法。

3 网络架构

3.1 拟物性采样

由于深度学习中的很多工作都采用了 VGG16 [10] 深度神经网络，所以我们的检测算法也是在 VGG16 的基础上进行修改。VGG16 有 13 个卷积层和 3 个全连接层，卷积层分为 5 个阶段，每个阶段后都有一个池化操作；由于最后一个全连接层是 ImageNet [3] 上图片分类的输出（1000 个输出），所以我们舍弃了最后一层。

我们先用 EdgeBoxes [13] 生成大约 6000 个拟物框，然后对它们进行评分，取评分较高的输入检测网络中进行检测。拟物性采样部分的网络如图1上半部分所示。首先，在 VGG16 [10] 后面加上一个 3×3 、通道数为 128 的卷积层，并接一个 ReLU 层进行非线性化；然后，在其后连接一个 ROI 池化层 [6]，将卷积特征采样成固定的 $128 \times 7 \times 7$ 大小，具体做法是将特征矩阵等分成 7×7 个窗格，

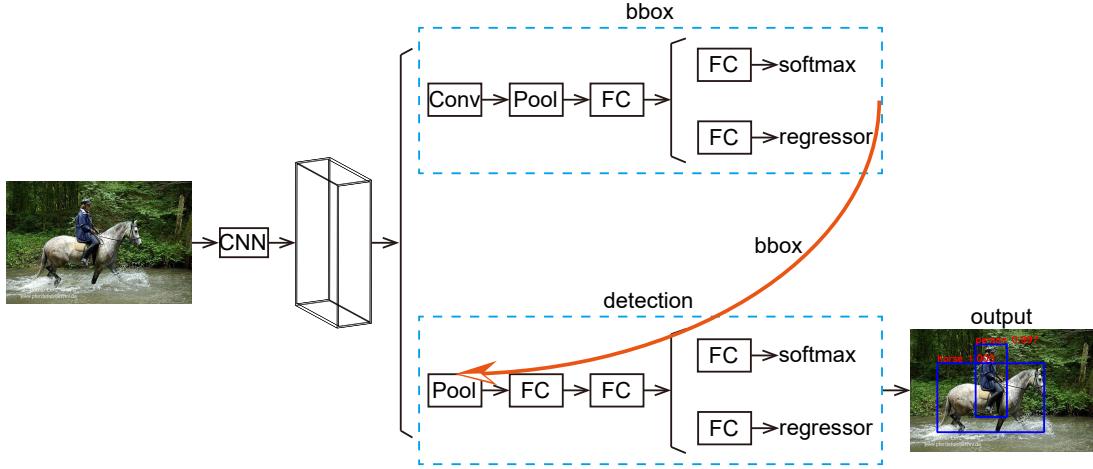


图 1：本文网络架构示意图，bbox 是拟物性采样部分网络，detection 是检测部分网络。图中 CNN 表示卷积网络，本文即为 VGG16 [10]；Pool 为 Fast RCNN [6] 中的 ROI 池化方法。

并在每个窗格内做最大值池化操作；在 ROI 池化之后接上一个输出为 512 维的全连接层，仍然使用 ReLU 的非线性化；最后，分为两个子分支，一个分支接输出为 2 维的全连接层，再接上一个 Sigmoid 层用来输出不是和是一个拟物框的概率；另一个分支接上输出为 8 维的全连接层，用来表示拟物框的位置修正。

训练该部分网络的损失函数定义如下：

$$L(p_i, p_i^*, t_i, t_i^*) = L_{cls}(p_i, p_i^*) + \lambda p_i^* L_{reg}(t_i, t_i^*). \quad (1)$$

该式中， i 表示一个输入的拟物框； p_i 表示预测出的该拟物框是一个物体的概率； p_i^* 是 1 表示该拟物框是一个物体，0 表示该拟物框不是一个物体； t_i 表示预测的该拟物框相对于真实物体边框的位移； t_i^* 表示实际上该拟物框相对于真实物体边框的位移。 t_i 和 t_i^* 的计算和 [6] 中相同。

3.2 共享卷积特征

物体检测部分的网络如图1下方所示，和 Fast RCNN 相同，不过所用的拟物框是从拟物性采样部分取排序的前一部分，在训练时取前 2000 个，测试时取前 300 个。由于拟物性采样和物体检测所用的卷积特征很多相似之处，所以我们考虑使得它们能够共享卷积层特征，框架如图1所示，这样即可以只对图片做一次前向传导即可。设计的训练算法如算法1所示。从该算法中可以看出，在第四步之前，网络仍然是没有共享卷积参数的，因为拟物性采样和物体检测还一直是分开训练。但是，第四步和第五步通过固定卷积层参数（CNN 部分），分别只微调

算法 1 联合训练过程

第一步：在 ImageNet 分类数据集上预训练 VGG16 网络，使得网络有一个合适初始值进行如下训练。

第二步：训练拟物性采样部分的网络用来重新排序 EdgeBoxes 产生的拟物框，用第一步训练的参数初始化 CNN 部分。

第三步：用第二步的模型对 EdgeBoxes 的拟物框进行排序，并取前 2000 个训练物体检测部分，此时仍然用第一步训练的参数初始化 CNN 部分。

第四步：用第三步训练的模型初始化网络，重新训练拟物性采样部分的网络，但是固定 CNN 部分的参数不变。

第五步：用第四步的模型对 EdgeBoxes 的拟物框进行排序，并取前 2000 个训练物体检测部分，此时用第三步训练的模型初始化网络，但是固定 CNN 部分的参数不变。

第六步：将第四步和第五步训练得到的模型参数组合起来，得到一个完整的模型。

拟物性采样和检测部分独有的网络，从而达到了共享卷积层参数的目的。

3.3 实现细节

本文的网络使用著名的深度学习框架 Caffe [8]，因为计算机视觉领域的很多深度学习工作都是用它实现的。新增加的、VGG16 网络中没有的层都用均值为 0、方差为 0.001 的高斯分布来初始化权重，用常数 0 来初始化偏移参数。拟物性采样和物体检测部分网络的 momentum 都设置为 0.9, weight_decay 都设置为 0.0005。使用随机梯度下降来进行训练，拟物性采样迭代 80000 次，其中前 60000 次使用 0.001 的学习率，后 20000 次使用 0.0001 的学习率；在物体检测上迭代 40000 次，其中前 30000 次使用 0.001 的学习率，后 10000 次使用 0.0001 的学习率。训练拟物性采样时，每次迭代使用 2 张图片，为每张图片取 128 个拟物框，共计 256 个拟物框进行训练；训练物体检测时，每次迭代仍使用 2 张图片，但是为每张图片取 64 个拟物框，共计 128 个拟物框。所有的训练和测试都是在一个 NVIDIA TITAN X GPU 上进行的。

4 实验结果

本文使用 VOC 2007 数据集 [4] 来评测我们算法的性能，VOC 2007 是一个广泛使用的用来评测物体检测性能的数据集。它包含 5011 张训练图片和 4952 张测试图片，有 20 类物体被标记出来。实验中，我们用训练集训练网络，用测试集测试网络；训练时，图片进行了水平反转以增加训练数据。我们的实验分为两部分，第一部分先来比较排序前后拟物框质量的变化，再比较新的物体检测系

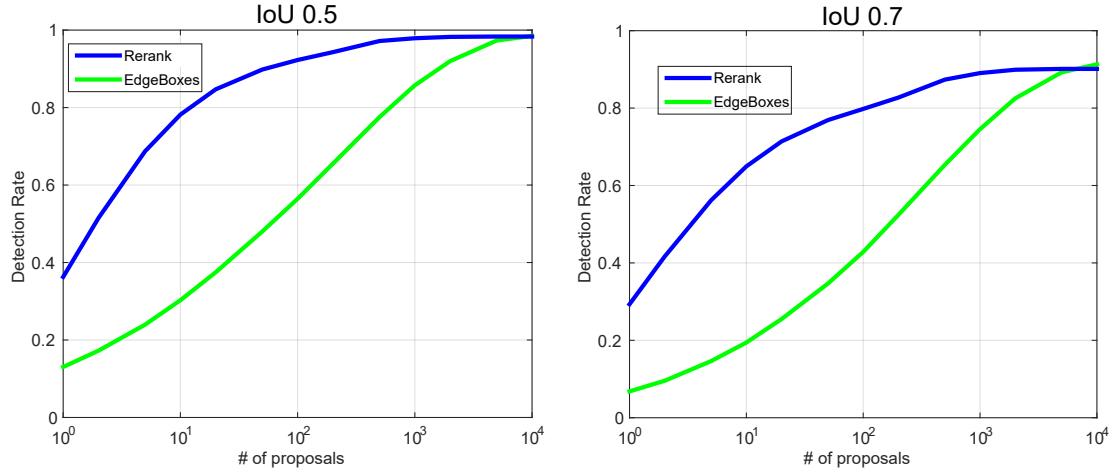


图 2: 在 VOC2007 数据集上评测的 EdgeBoxes 产生的拟物框和应用本文算法进行排序后的拟物框，从图中容易看出，排序前后的差距非常明显。

统的检测质量。

4.1 拟物性采样

拟物性采样一般使用找回率来进行评测，当判断一个拟物框是否正确时，是看它与所有物体边框的最大重叠 *overlap* 是否大于一个阈值 IoU。若 *overlap* 大于等于 IoU 时，则此拟物框是正确的；否则，则认为该拟物框是错误的。*overlap* 的计算公式如下：

$$\text{overlap}(R, G) = \max_i \frac{R \cap G_i}{R \cup G_i}, \quad (2)$$

其中 R 代表一个生成的拟物框，G 代表真实物体的边框。

使用提出的网络进行排序前后拟物性采样的质量评测结果如图2所示，左图表示 IoU 取 0.5 时召回率随拟物框数量的关系曲线，而右图是 IoU 取 0.7 时的情况。图中蓝色曲线代表使用设计的网络排序后的曲线，绿色曲线表示 EdgeBoxes 直接生成的拟物框。从图中可以清晰地看到，两条曲线之间的差距非常大，差距最大的地方超过 45% 的提升，这说明了对 EdgeBoxes 产生的拟物框进行重新排序是十分必要的。在最后面两条曲线逐渐相交于一点，这是因为算法主要是对 EdgeBoxes 生成的拟物框拟物框进行重新排序，只将质量高的拟物框提前了，而没有改变拟物框数量最大时的质量。实际上，当取 100 个拟物框，IoU 是 0.5 时召回率从 56.7% 提高到了 92.5%；IoU 是 0.7 时召回率从 43.1% 提高到了 80.4%。当取 300 个拟物框，IoU 是 0.5 时召回率从 71.3% 提高到了 95.9%；IoU 是 0.7 时召回率从 58.3% 提高到了 85.8%。在物体检测中，我们只需要选取前 300 个拟物框输入检测网络即可；由于检测网络运算量更大，所以这样不仅可以提高准

确率还可以提高检测速度。

4.2 物体检测

我们同样在 VOC 2007 数据集 [4] 数据集上来评测物体检测的性能，并使用该数据集定义的 mAP 指标，mAP 越大，效果越好，mAP 的最大可能值是 1。我们将我们的方法与相关的 RCNN [7]、Fast RCNN [6]、Faster RCNN [9] 等进行对比，评测结果如表1所示。

表 1: 各种方法在 VOC 2007 数据集上物体检测结果的对比

method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
RCNN	73.4	77.0	63.4	45.4	44.6	75.1	78.1	79.8	40.5	73.7	62.2	79.4	78.1	73.1	64.2	35.6	66.8	67.2	70.4	71.1	66.0
Fast RCNN	74.5	78.3	69.2	53.2	36.6	77.3	78.2	82.0	40.7	72.7	67.9	79.6	79.2	73.0	69.0	30.1	65.4	70.2	75.8	65.8	66.9
Faster RCNN	67.5	78.5	67.3	51.9	51.5	76.2	79.8	84.4	50.2	74.3	66.9	83.2	80.0	73.9	76.5	37.1	69.4	65.7	76.5	74.2	69.2
Ours	69.0	79.0	66.9	55.7	54.9	79.8	79.7	85.3	52.8	75.4	69.8	75.7	79.0	76.2	77.4	40.3	73.7	66.1	76.5	70.7	70.2

由表1易知，我们的方法取得了最好的检测结果，比 Fast RCNN 高了 3.3%，比 Faster RCNN 高了 1.0%。同时，我们的算法处理一张图片只需要 0.130 秒的时间。为了直观的展示本文算法进行物体检测的结果，我们挑选了一些例子，如图3所示。

5 结论

本文提出了一个新的物体检测算法，该算法先将 EdgeBoxes 生成的大量拟物框进行重新排序，然后取排名靠前的少量拟物框用于物体检测。并且，拟物框重排序和物体检测可以共用卷积特征，这使得每张图片只需要只需要在卷积神经网络进行一次前向传导即可，节省了算法的运行时间。在著名数据集 VOC2007 [4] 上的评测结果表明，我们的算法取得了较好的结果。但是，我们发现该方法在小物体上的检测效果明显不如大物体，推测原因是经过多次池化后卷积特征尺度变得太小了，该问题或许可以通过提取和结合中间层卷积特征来解决，这将是下一步的工作目标。

参考文献

- [1] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. PAMI, 34(11):2189–2202, 2012.



图 3: 一些物体检测结果的例子

- [2] Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. In CVPR, pages 3286–3293, 2014.
- [3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In CVPR, pages 248–255. IEEE, 2009.
- [4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>.
- [5] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models.

- PAMI, 32(9):1627–1645, 2010.
- [6] Ross Girshick. Fast r-cnn. In ICCV, pages 1440–1448, 2015.
- [7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, pages 580–587, 2014.
- [8] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In ACM MM, pages 675–678. ACM, 2014.
- [9] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In NIPS, pages 91–99, 2015.
- [10] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.
- [11] Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. IJCV, 104(2):154–171, 2013.
- [12] Ziming Zhang, Jonathan Warrell, and Philip HS Torr. Proposal generation for object detection using cascaded ranking svms. In CVPR, pages 1497–1504. IEEE, 2011.
- [13] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In ECCV, pages 391–405. Springer, 2014.