RefinedBox: 通过精炼生成少量且高质量的物体推荐

刘云, 李仕杰, 程明明*

CCS, Nankai University, Tianjin, P.R.China, 300350

摘要

最近，拟物性采样通过猜测物体位置显示出用于各种视觉任务的价值，例如物体检测、语义实例分割、多标签图像分类和弱监督学习等。我们受到以下事实的启发：许多传统的拟物性采样方法会生成密集的推荐来涵盖尽可能多的物体，但是 i) 他们通常无法对这些推荐进行正确地排序，并且 ii) 物体推荐的数量非常大。例如，著名的拟物性采样方法，Edge Boxes 和 Selective Search，可以通过为每个图像生成数千个推荐，来实现高检测召回率。但是，由于误报的大量存在以及繁重的计算量，所生成的大量推荐导致后续的分析变得困难。为了显著减少物体推荐的数量，我们设计了一个轻量级的神经网络来优化最初的物体推荐。所提出的优化包括两个并行过程，即重新排序和推荐框回归。所提出的网络可以通过与其他高级任务联合训练来共享卷积特征，因此优化物体推荐的速度非常快。我们在本文中展示了一个与物体检测联合训练的示例。大量的实验表明，与一些著名的拟物性采样方法相比，我们的方法只生成很少的推荐即可达到最佳性能。

关键词: 物体推荐; 更少的物体推荐; 物体推荐挖掘

1. 引言

生成少量物体推荐的同时覆盖图像中尽可能多的物体，可以通过减少搜索空间和误报，而对于后续高级应用（如，物体检测 [2, 3]、语义实例分割 [4, 5]、多标签分类 [6]、视频总结 [7] 和深度多实例学习 [8] 等）的效率和准确性至关重要。在过去的十年中，已经提出了许多自下而上的拟物性采样方法，旨在生成密集的推荐来覆盖尽可能多的物体，例如 Selective Search [9]、Edge Boxes [10] 和 MCG [11]。由于使用传统的手工提取的特征很难表示高层语义信息，因此这些自下而上的方法通常 (i) 无法对生成的物体推荐进行正确排序，并且 (ii) 必须使用大量的物体推荐才能确保检测召回率。虽然这些现有的自下而上的算法可以通过在每张图像上生成数千个推荐来实现较高的检测召回率，但是由于存在大量的误报且计算负载繁重，这些生成的大量的物体推荐使后续的分析变得非常困难 [6, 8, 12, 13]。最近，一些基于深度学习的拟物性采样方法在该领域引起了很多关注，包括 RPN [14]、DeepMask [15] 和 SharpMask [16]。由于卷积神经网络（Convolutional Neural Network, CNN）具有强大的表征能力，因此与传统的自下而上的算法相比，这些方法可通过较少的候选框来实现较高的检测召回率。然而，RPN [14] 通过从下采样的卷积特征图（1/16 尺度）中采样锚点来生成物体推荐，而 DeepMask [15] 和 SharpMask [16] 通过扫描图像小块来发现物体。因此，这种次优的物体推荐采样策略使他们难以充分利用 CNN 的强大能力。所导致的结果就是，图像中真实物体的数量（例如通常少于 10 个）仍然比通过这些基于深度学习的方法生成的推荐的数量（例如通常数百个）少得多。

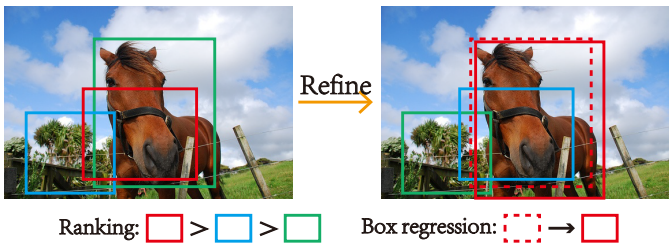


图 1: 物体推荐优化的概述。左图显示了初始的物体推荐，右图显示了优化后的结果。我们首先通过计算新的拟物性分数对物体推荐重新排序，然后对每个推荐框进行回归来实现准确定位。

我们能否在保持高召回率的同时大幅减少物体推荐的数量呢？这对于更广泛的应用至关重要，例如当从大量未标记/弱标记的数据中挖掘知识 [6, 8] 时，大量误报的存在不仅对计算效率甚至是系统稳定，都构成了

*Corresponding author: MM Cheng (cmm@nankai.edu.cn).

本文为 Neurocomputing 论文 [1] 的中译版。

重大挑战。已经有一些针对特定视觉任务来减少推荐数量的研究。例如，Wei 等人 [6] 对 BING 算法 [17] 生成的边界框采用归一化切割（Normalized Cut）[18] 进行聚类，并在每个聚类中挑选出拟物性分数最高的那个推荐。他们将所挑选出的物体推荐应用于多标签图像分类，并达到了最佳性能。Qi 等人 [12] 通过计算每个像素的所有拟物性分数的总和（相应推荐框覆盖了该像素）来引入每个像素的融合分数，所得到的融合分数图被用于估计物体位置。Li 等人 [13] 采用了遮挡策略，可以为每个物体类别收集质量更高的推荐。对于一个类别，如果被一个推荐框所遮挡的图像使得此类的分类分数有明显下降，则针对此类别就收集这个推荐。

在本文中，我们专注于减少推荐数量的同时获得较高的检测召回率。我们观察到，当候选框的数量足够多时，某些传统的拟物性采样方法可以实现较高的检测召回率，这是因为与深度学习中简单的推荐采样策略 [14, 15, 16] 不同，传统方法通常设计巧妙的策略来搜索物体的所有可能位置。当然，大量的候选框会在后续的应用中引起许多误报，从而影响最终的性能。但是，如果我们可以从大量的候选框中选择优秀的候选框，那么这将有利于一系列视觉任务。最近，已经有几种算法被提出来改善物体推荐，包括 DeepBox [19] 和 MTSE [20]。DeepBox 建立了一个神经网络来重新计算初始框的拟物性分数，然后对其进行重新排序。MTSE 试图使用超像素优化每个框，具体是通过使每个框紧密覆盖一些内部的超像素。然而，DeepBox 的物体推荐质量比 RPN [14] 还差，因此无法减少推荐数量。此外，MTSE 的性能取决于超像素的质量，并且 MTSE 中的图像分割会导致计算负荷的显著增加。

为了结合传统拟物性采样方法的优势和 CNN 强大的表征能力 [14, 15, 16, 21, 22]，我们提出了一种新的方法来在神经网络的单次前向传播中，对现有推荐框进行重新排序和推荐框回归。我们方法的概述如图1所示。我们对候选框的优化包括两个步骤：重新排序和推荐框回归。重新排序这一步尝试根据推荐框覆盖完整物体的紧密程度对推荐框进行重新排序。推荐框回归这一步则是尝试微调框的形状和位置，使其更紧密地覆盖真实物体。为了实现这个目标，我们的优化网络旨在学习新的拟物性分数并同时进行框回归。所提出的网络在计算上是轻量级的，因此它的应用只消耗很少的时间。优化的训练过程是以端到端的方式进行的。为了简洁起见，在本文的其余部分中，我们将所提出的方法称为 RefinedBox。由于 RefinedBox 是轻量级且易于优化的，因此可以通过与高级应用联合训练来共享卷积特征。为了展示一个联合训练的示例，我们通过在基础网络（例如 VGG16 [23]）的最后一个卷积层之后连接我们的优化层，来将 RefinedBox 和著名检测框架 Fast R-CNN[3] 整合为一个统一的框架，然后引入一种交替微调策略进行训练。最终，我们的优化网络可以与后续的物体检测网络共享基本卷积层，从而使物体推荐的优化过程非常高效。

用各种传统方法生成的推荐框作为输入，我们在 PASCAL VOC2007 [24] 和 MS COCO [25] 数据集上评测了所提出的方法。对于在 VOC2007 数据集上的拟物性采样，我们的方法在重叠率（Intersection-over-Union, IoU）为 0.5 和 0.7 下的检测召回率分别为 80.4 % 和 67.9%，且每张图像仅使用 10 个优化后的推荐框。当仅使用 10 个推荐框进行物体检测时，我们的方法的平均精度（mean Average Precision, mAP）为 65.4%，而 RPN [14] 的 mAP 是 54.1%。实验表明，所提出的 RefinedBox 方法可以在物体推荐数量有限的情况下生成高质量的物体推荐。

2. 相关工作

既然本文针对的是物体推荐的优化，我们首先简要描述拟物性采样的最新发展。然后，我们继续讨论边界框的优化技术。我们将相关研究工作大致分为四个部分：基于分割的拟物性采样方法、基于边缘的方法、基于 CNN 的方法以及物体推荐的后处理方法。

基于分割的拟物性采样方法使用图像分割作为输入，并尝试找到这些图像块的正确组合来覆盖所有完整的物体。这些方法通常结合一些底层特征（例如显著性、颜色、SIFT [26] 等）来对边界框进行评分，然后选择分数较高的推荐框。Selective Search [9] 是最著名的拟物性采样方法之一，它利用穷举搜索和分割的优势，通过对超像素进行分层合并来获得高质量的推荐。MCG [11] 引入了一种可有效利用多尺度信息的高性能图像分割算法。通过探索组合空间，将所产生的具有多尺度层次结构的区域组合为物体推荐。Manen 等人 [27] 建立了图像超像素的连通图，并使用普里姆算法（Prim's algorithm）的随机版本生成了具有较大边缘权重的期望总和的生成树。这些生成树的边界框就是最终的物体

推荐。Rantalankila 等人 [28] 对超像素执行局部搜索以形成分割层次，并使用全局搜索来获得中间层次结构的图割分割。很多其他的拟物性采样方法 [29, 30, 31] 都属于此类。

基于边缘的方法利用了自然图像中的完整物体通常具有明确的封闭边界的观察 [32]。近年来，已经提出了多种使用边缘特征的高效算法。Zhang 等人 [33] 设计了一种级联排序 SVM（CSVM）方法，使用梯度特征获得物体推荐。Cheng 等人 [17] 提出了一种非常有效的算法 BING，该算法通过将 CSVM [33] 量化为一些二进制运算使其以 300fps 的速度运行。Lu 等人 [34] 提出了一种新的基于封闭路径积分的封闭轮廓度量。Edge Boxes [10] 根据每个边界框中完全包含的轮廓数来计算拟物性分数。

基于 CNN 的方法直接从 CNN 生成物体推荐，例如 RPN [14]、DeepMask [15] 和 SharpMask [16]，这是受到 CNN 具有强大的特征表征学习能力的启发 [21, 22]。RPN [14] 同时预测整个图像的卷积特征每个位置的物体边界和拟物性分数。DeepMask [15] 的训练目标有两个：给定一个图像小块，系统首先输出一个与类别无关的分割蒙版，然后输出该小块以整个物体为中心的可能性。SharpMask [16] 提出使用一种新颖的自上而下的优化方法来增强前馈网络，以进行物体分割。由此产生的自下而上/自上而下的体系结构能够高效地生成高保真的物体蒙版。但是，对于自然图像，这些基于 CNN 的方法生成的物体推荐的数量仍然太多（通常为几百个）。

物体推荐的后处理致力于改进物体推荐，以便在图像中准确定位物体。Kuo 等人 [19] 提出了一个名为 DeepBox 的小型神经网络，用于重新计算已存在的物体推荐框的拟物性分数，然后根据新的拟物性分数对这些推荐框重新排序。Chen 等人 [20] 尝试将物体推荐框与超像素对齐。Zhang 等人 [35] 进一步讨论了拟物性采样的优化。他们首先使用边缘，然后使用超像素来优化物体推荐框。他们基于分割的优化加速了 MTSE [20] 中超像素的生成，因此最终的系统可以以非常快的速度运行。He 等人 [36] 提出了具有不同方向的定向物体推荐，而不仅仅是常规方法中使用的垂直框。在本文中，我们建立了一个优化网络来优化现有的边界框。通过我们的方法生成的优化框在拟物性采样的评测和物体检测的评测中均达到了最佳性能。

3. RefinedBox

3.1. 网络架构

我们的方法以其他拟物性采样方法生成的物体推荐作为输入，然后尝试对其进行优化。优化包括两步：重新排序和推荐框回归。为了对现有的物体推荐框重新排序，我们使用深度神经网络中的语义信息重新计算每个推荐框的拟物性分数。为了进行推荐框回归，我们设计网络来学习每个物体推荐框的中心坐标、宽度和高度的回归。

VGG16 [23] 是深度学习领域广泛使用的骨干网络体系结构。它由 13 个卷积层和 3 个全连接层组成。受到之前研究 [3, 14] 的启发，本文基于 VGG16 构建我们的网络来阐述我们的优化方法。图2中显示了我们的网络体系结构。所提出的网络采用自然图像和相应的初始物体推荐框作为输入，初始框是由其他拟物性采样方法生成的。在本文中，我们以一些著名的拟物性采样方法为例，如 Edge Boxes [10]、MCG [11]、Selective Search [9] 和 RPN [14]。输入图像首先经过一些卷积层的前馈，例如 VGG16 中的 13 个卷积层。为了减少推荐框优化的时间消耗，我们设计了一个计算轻量级的神经网络。具体而言，我们首先在第 13 个卷积层之后连接一个卷积核大小为 3×3 的卷积层来将通道数从 512 减少到 128。然后，连接一个 ROI 池化层（ROI Pooling [3]），将每个初始框区域下采样为固定的特征图大小，即 7×7。ROI 池化层将输入特征图划分为宽度和高度相同的网格，并在每个网格中进行最大池化操作。然后，将其连接到一个只有 512 个输出神经元的全连接层。在所添加的卷积层和全连接层之后分别连接 ReLU 层。最后，使用排序和推荐框回归两个分支来分别重新计算拟物性分数并获得每个初始框的位置偏移。排序分支是一个具有两个输出神经元的全连接层，两个输出神经元分别表示该推荐框是否是一个物体的概率。推荐框回归分支预测推荐框的回归值，这将在下面进行描述。

在 RefinedBox 的训练中，为每个初始的物体推荐框分配一个是否为物体的二值的类标签。损失函数可以写成

$$L_{obj}(p,u) = -[1_{\{u=1\}} log\ p_1 + 1_{\{u\neq 1\}} log\ p_0], \qquad (1)$$

其中，p 是在全连接层的两个输出上求 softmax 计算得到的，u 是此框的标签（1 或 0）。推荐框回归层是一个

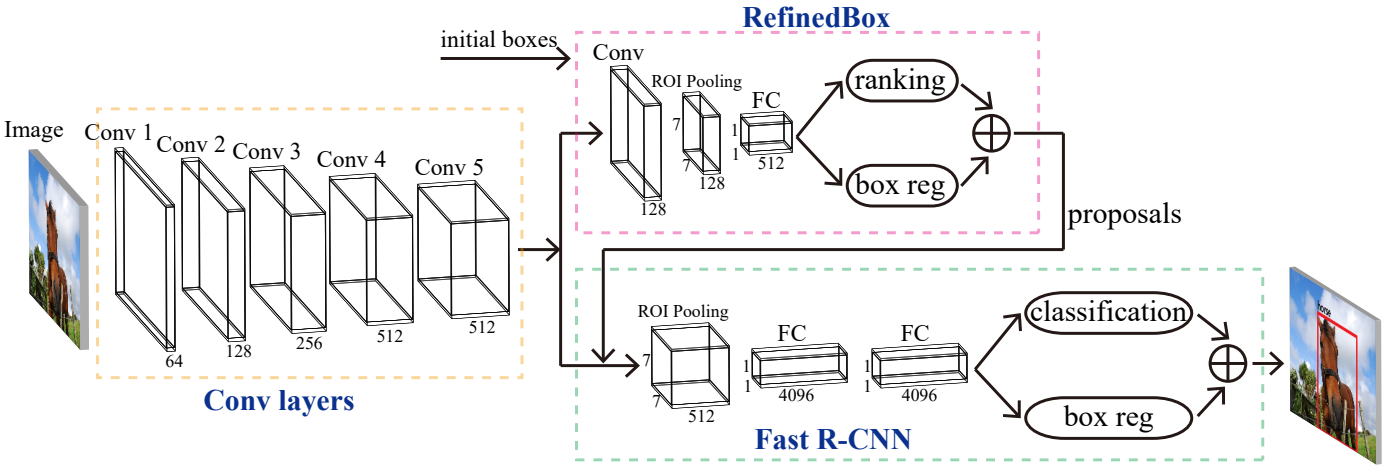


图 2: 所提出的网络架构概述。我们以与目标检测进行联合训练为例。所提出的网络将自然图像和由其他拟物性采样方法（例如 Edge Boxes）生成的相应初始框作为输入。设计 RefinedBox 分支来优化初始框，然后将优化的推荐框输入到 Fast R-CNN 分支中进行分类。值得注意的是，推荐框的优化和随后的物体检测可以共享卷积特征。

全连接层，旨在学习坐标偏移。我们按以下方式对四个坐标进行参数化：

$$\begin{aligned} t_x &= (x - x_{in})/w_{in}, & t_y &= (y - y_{in})/h_{in}, \\ t_w &= log(w/w_{in}), & t_h &= log(h/h_{in}), \\ v_x &= (x^* - x_{in})/w_{in}, & v_y &= (y^* - y_{in})/h_{in}, \\ v_w &= log(w^*/w_{in}), & v_h &= log(h^*/h_{in}), \end{aligned} \tag{2}$$

其中，x、y、w 和 h 分别代表推荐框的中心的坐标、宽度和高度。变量 x、x_{in} 和 x^* 分别是对于预测框、输入框和真值框来说的；y、w 和 h 也使用类似的定义。变量 v 是回归目标，t 是预测的元组。推荐框回归的损失函数定义为：

$$\begin{aligned} L_{reg} &= \sum_{i \in \{x,y,w,h\}} \text{smooth}_{L_1}(t_i - v_i), \\ \text{smooth}_{L_1}(x) &= \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise,} \end{cases} \end{aligned} \tag{3}$$

其中，$\text{smooth}_{L_1}(x)$ 是一个著名的回归损失函数 [3]。因此，联合损失函数可以写成

$$L(p, u, t, v) = L_{obj}(p, u) + \lambda \cdot 1_{\{u=1\}} L_{reg}(t, v), \tag{4}$$

其中，参数 λ 是一个平衡参数，在本文中将其设置为 1。

3.2. 与物体检测联合训练

到目前为止，我们已经描述了如何训练物体推荐的优化网络。由于所提出的网络是轻量级的，因此它有与高级应用共享卷积特征的潜力。在这里，我们以物体检测为例，阐述 RefinedBox 及其后续应用的联合训练过程。为了测试 RefineBox 生成少量且高质量物体推荐的能力，我们仅使用每张图像中 RefinedBox 生成的前 10 个物体推荐进行物体检测。

算法 1 RefinedBox 的交替训练过程。

Input: 所提出的神经网络的骨干网络 W_{VGG}、RefinedBox 模块 W_{RB}、物体检测模块 W_{Det}；初始的物体推荐 B_{in}；在 ImageNet 上预训练的骨干网络 W^{pre}_{VGG}

Output: 训练好的 W_{VGG}、W_{RB} 和 W_{Det}

Step 1: $W_{VGG} \leftarrow W^{pre}_{VGG}$; $W_{RB} \leftarrow random()$

Step 2: $W_{VGG}, W_{RB} \leftarrow finetune(W_{VGG}, W_{RB}; B_{in})$

Step 3: $B' \leftarrow rerank(B_{in}; W_{VGG}, W_{RB})$

Step 4: $W_{VGG} \leftarrow W^{pre}_{VGG}$; $W_{Det} \leftarrow random()$

Step 5: $W_{VGG}, W_{Det} \leftarrow finetune(W_{VGG}, W_{Det}; B')$

Step 6: $W_{RB} \leftarrow random()$

Step 7: $W_{RB} \leftarrow finetune(W_{RB}; W_{VGG}, B_{in})$

Step 8: $B' \leftarrow rerank(B_{in}; W_{VGG}, W_{RB})$

Step 9: $W_{Det} \leftarrow random()$

Step 10: $W_{Det} \leftarrow finetune(W_{Det}; W_{VGG}, B')$

如图2所示，我们在卷积层之后连接一个著名的检测框架 Fast R-CNN [3]，将其作为与 RefinedBox 并行的一个分支。将由 RefinedBox 分支生成的优化后的物体推荐输入到 Fast R-CNN 中。为了使 RefinedBox 和 Fast R-CNN 共享相同的卷积特征，我们使用了一个交替进行的微调过程，如算法1所示。物体检测的训练取决于先前步骤所生成的重新排序的物体推荐。在步骤 6

之前，物体推荐和物体检测的网络是分别训练的。然后，固定骨干网络，并对用于 RefinedBox 和物体检测的特定层进行微调。经过交替训练，两个网络形成一个统一的网络。

对于其他高级应用，联合训练也以类似的方式进行。换句话说，通过将 W_{Det} 替换为其他任务的模块，算法1也适用于其他任务。算法1的关键是，通过在高级任务和 RefinedBox 模块之间进行交替训练，使高层任务和 RefinedBox 共享相同的骨干网络，因此输入图像只需要通过骨干网络一次。

浮点运算（Floating-Point Operations, FLOPs）的数量通常用于衡量网络的计算消耗，其中浮点运算表示乘加运算（Multiply-Add Operations）。对于每个物体推荐框，Fast R-CNN 分支的全连接层有 120.0M（million）个 FLOPs，而 RefinedBox 分支的全连接层只有 3.2M 个 FLOPs。因此，RefinedBox 分支只会带来很少的额外的计算开销。

3.3. 实现细节

对于 RefinedBox 的训练，每个随机梯度下降（Stochastic Gradient Descent, SGD）的小批量都是从一张图像中选择 256 个推荐框作为训练样本构造而成的。在每一批中，所采样的推荐框一半为正样本，一半为负样本。重叠率（Intersection-over-Union, IoU）是指两个框的相交面积与并集面积的比率。正采样框与真值框的 IoU 重叠率至少为 0.7，而负采样框与真值的最大 IoU 重叠率在 $[0.1, 0.5)$ 之间。初始学习率设置为 1e-3，并在 12 个纪元后除以 10。SGD 总共运行 16 个纪元。

为了训练检测模块，每个小批量都有 256 个来自同一图像的物体推荐。与 Fast R-CNN [3] 中一样，这些物体推荐中有 25% 的推荐与真值的 IoU 重叠率至少为 0.5，它们被视为正样本。其余的负样本与真值的最大 IoU 重叠率在区间 $[0.1, 0.5)$ 内。使用 RefinedBox 生成的前 1000 个推荐进行训练。对于前 12 个纪元，学习率为 1e-3，而对于另外 4 个纪元，学习率除以 10。对于测试，每张图像仅使用 RefinedBox 的前 10 个推荐。相比之下，传统的物体推荐方法（例如 Edge Boxes 和 Selective Search）通常需要数千个推荐。我们基于公开的代码[1] 实现了所提出的方法。训练和测试是在一块 GTX TITAN X GPU 上进行的。

[1] https://github.com/rbgirshick/py-faster-rcnn

表 1: 在 PASCAL VOC2007 测试集上关于 DR 的评测结果（%）。RefinedBox[1]、RefinedBox[2]、RefinedBox[3]、和 RefinedBox[4] 分别表示基于 Edge Boxes、MCG、Selective Search 和 RPN 的 RefinedBox。

#WIN	DR (IoU=0.5)				DR (IoU=0.7)				Time
	10	30	50	100	10	30	50	100	(s)
BING	37.5	51.0	60.4	70.1	16.9	20.2	22.5	24.4	**0.003**
CSVM	40.8	56.1	64.2	74.3	16.2	20.9	23.1	25.5	0.33
EdgeBoxes	45.9	60.0	66.7	75.4	31.0	43.8	51.1	60.8	0.25
Endres	54.8	68.9	75.6	83.3	35.1	47.1	52.2	59.0	19.94
GOP	13.7	29.5	40.7	60.0	0.7	15.6	22.3	35.6	0.29
LPO	38.2	59.4	66.4	75.3	17.5	34.8	41.3	48.8	0.46
MCG	51.7	69.3	75.8	82.1	30.2	45.4	51.7	60.1	17.46
Objectness	38.2	50.2	56.4	65.4	17.4	22.6	25.0	29.3	0.91
Rahtu	34.3	46.9	53.3	62.3	21.9	32.1	38.1	45.8	0.67
RandomPrim	34.4	50.7	59.2	70.7	16.4	28.1	34.4	44.5	0.12
Rantalankila	0.6	3.1	6.5	14.9	0.2	1.2	2.6	7.4	3.57
SelectiveSearch	37.1	54.3	61.8	71.8	19.9	32.7	39.6	49.4	1.60
RPN	60.1	73.8	80.7	89.0	32.9	47.6	54.5	64.4	0.10
DeepBox	58.1	71.8	77.2	84.5	40.7	55.4	62.7	70.9	0.45
DeepMaskZoom	61.8	78.5	84.7	91.0	44.2	58.1	63.8	71.1	1.20
SharpMaskZoom	62.6	79.5	85.4	91.9	47.0	60.9	66.5	74.0	0.57
RefinedBox[1]	80.4	88.3	90.6	**92.7**	67.9	**76.4**	**79.2**	**82.4**	0.31
RefinedBox[2]	**80.5**	87.6	88.8	89.6	68.2	75.2	76.4	77.1	17.52
RefinedBox[3]	79.2	86.4	88.2	89.7	**68.6**	76.1	78.0	79.6	1.66
RefinedBox[4]	79.5	**88.6**	**90.8**	92.4	65.3	75.2	77.6	79.5	0.16

4. 实验

4.1. 实验设置

数据集：我们在两个广泛使用的物体检测数据集上评测了所提出的方法，即 PASCAL VOC2007 [24] 和 MS COCO [25]。PASCAL VOC2007 数据集 [24] 由 2501 张训练、2510 张验证和 4952 张测试图像组成，所有图像都带有相应的 20 个物体类别的标注。我们在 VOC2007 的 *trainval* 集（训练和验证集）上训练模型，并在 VOC2007 的测试集上进行测试。MS COCO 数据集 [25] 由 82783 张训练图像和 40504 张验证图像组成。我们将其训练集用于训练，将其验证集用于物体推荐的评测。

对比的方法：为了证明所提出的物体推荐的优化方法的有效性，我们将所提出的方法与现有的主流的物体推荐方法进行了比较，包括基于非深度学习的方法，如 BING [17]、CSVM [33]、Edge Boxes [10]、Endres [30]、GoP [37]、LPO [31]、MCG [11]、Objectness [32]、Rahtu [29]、RandomPrim [27]、Rantalankila [28] 和 Selective Search [9]，以及最近的基于深度学习的方法，如 RPN [14]、DeepBox [19]、DeepMaskZoom [15] 和 SharpMaskZoom [16]。所使用的 DeepMaskZoom 和 SharpMaskZoom 分别是 DeepMask [15] 和 SharpMask [16] 的最好版本。我们首先与这些方法进行拟物性采样的比较。然后，对

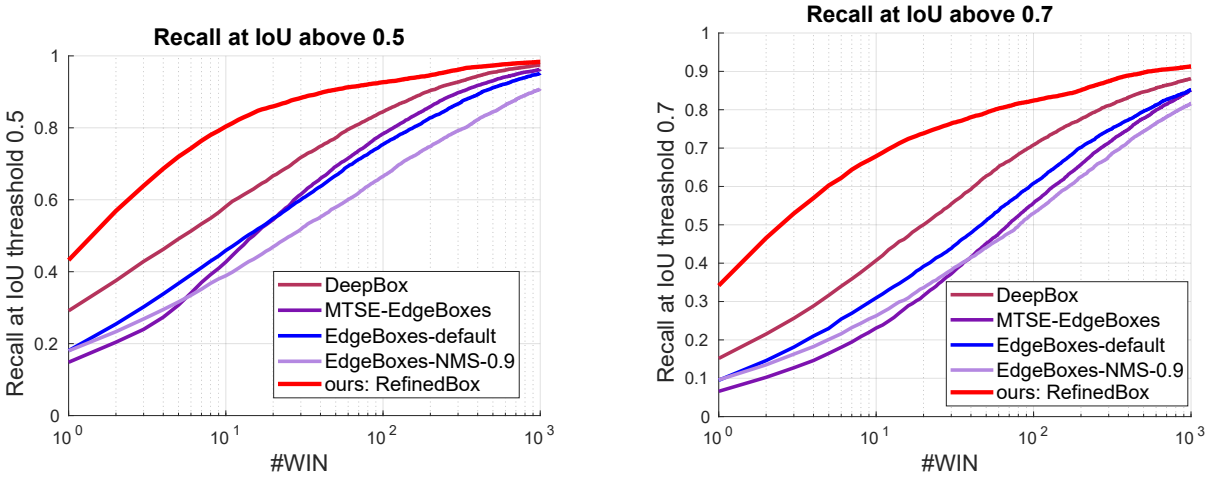


图 3: PASCAL VOC2007 数据集上不同优化算法的评测。这两个子图分别显示在 IoU 阈值分别为 0.5（左）和 0.7（右）下的物体检测召回率 *vs.* 物体推荐数（#WIN）。EdgeBoxes-default 使用 Edge Boxes [10] 方法的默认参数，EdgeBoxes-NMS-0.9 将非极大值抑制（Non-Maximum Suppression, NMS）的参数改为 0.9。

于 PASCAL VOC2007 数据集 [24]，我们将这些方法生成的物体推荐输入到基于区域的物体检测框架 Fast R-CNN [3] 中，以便于在物体检测中评测推荐的质量。我们的实验表明，我们的方法可以为物体检测生成高质量的物体推荐，并且效率很高。

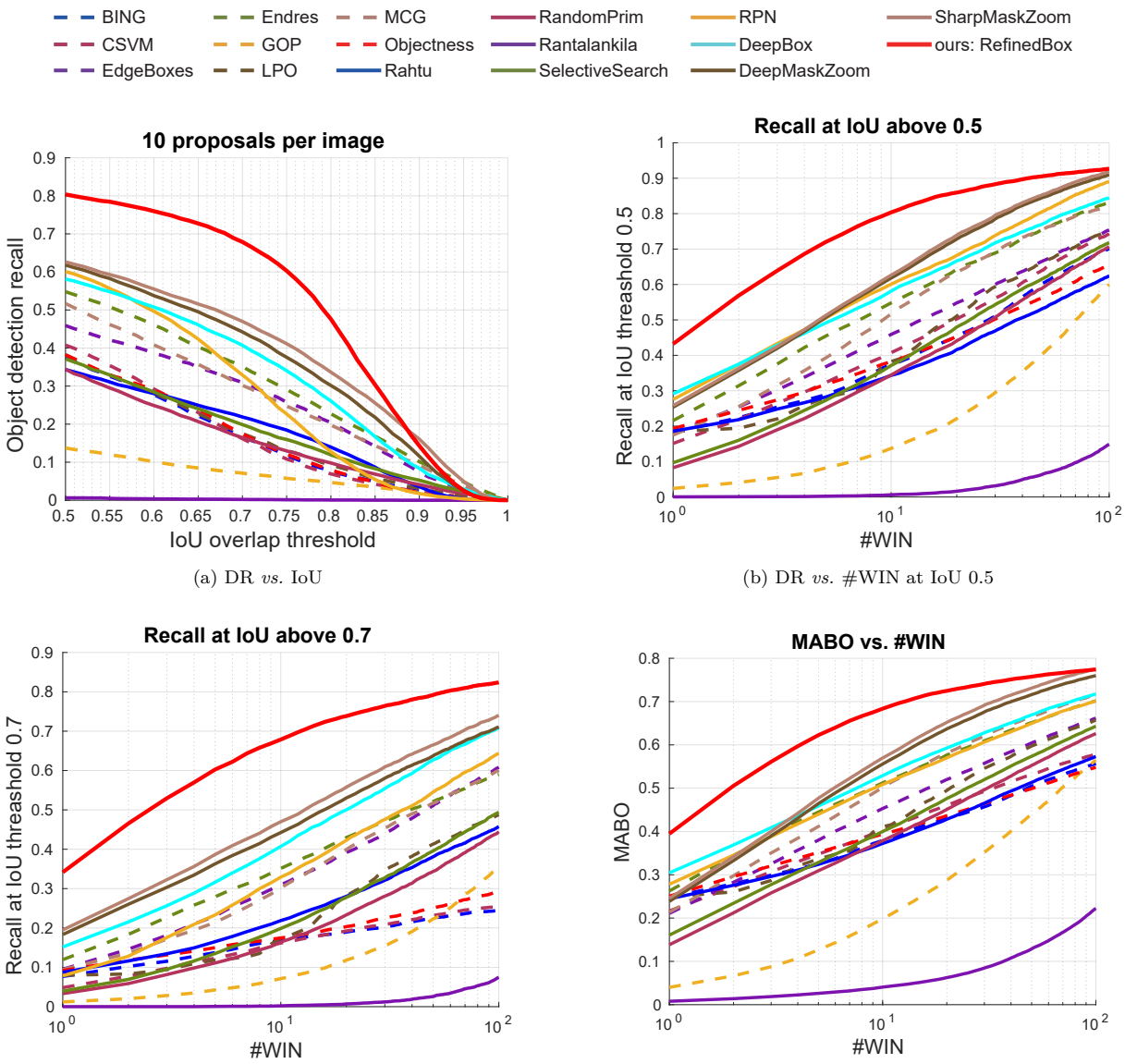
指标：为了评测物体推荐，我们使用的指标有物体检测召回率（Detection Recall, DR）、平均最佳重叠率（Mean Average Best Overlap, MABO）和平均召回率（Average Recall, AR）。检测召回率（DR）认为当真值物体与一个物体推荐的 IoU 重叠率大于阈值时，那么就认为这个真值物体被找到了。为了计算特定类别的平均最佳重叠率（ABO），我们计算（属于此类的）每一真值标注与为相应图像生成的物体推荐之间的最佳 IoU 重叠率，并对该类别中的所有真值物体进行平均。MABO 被定义为所有类别的平均 ABO [9]。Hosang 等人 [38] 引入了 AR，以计算一定数量的物体推荐在 IoU 阈值为 [0.5 : 0.05 : 0.95] 下的平均召回率。

表 2: 在 PASCAL VOC2007 测试集上关于 AR、MABO 和 mAP（每张图像使用 10 个物体推荐的物体检测性能）的评测结果（%）。RefinedBox[1]、RefinedBox[2]、RefinedBox[3] 和 RefinedBox[4] 分别表示基于 Edge Boxes、MCG、Selective Search 和 RPN 的 RefinedBox。

#WIN	AR				MABO				mAP
	10	30	50	100	10	30	50	100	
BING	16.5	21.3	24.6	27.9	37.9	45.7	50.5	55.7	34.4
CSVM	17.0	22.7	25.5	29.1	40.3	49.2	53.1	57.9	35.7
EdgeBoxes	26.3	36.3	41.3	48.0	45.3	55.7	60.4	66.2	39.1
Endres	31.1	40.5	44.8	50.6	51.2	60.9	65.1	70.2	42.8
GOP	6.8	14.6	20.7	31.9	19.8	35.2	44.0	56.4	13.3
LPO	17.2	31.1	36.7	43.2	41.1	54.6	59.7	65.7	34.5
MCG	27.6	40.5	45.9	52.9	50.1	62.1	66.5	71.6	41.2
Objectness	16.8	22.0	24.6	28.8	39.4	46.5	49.9	54.8	34.9
Rahtu	18.5	26.5	30.8	36.8	37.2	46.5	51.3	57.3	32.4
RandomPrim	16.1	25.8	31.3	39.6	37.9	49.6	55.3	62.6	31.9
Rantalankila	0.2	1.2	2.7	7.0	4.1	8.5	12.9	22.3	2.4
SelectiveSearch	18.6	29.8	35.5	43.6	40.0	52.0	57.4	64.3	34.1
RPN	28.4	38.1	42.7	48.9	50.8	60.6	65.0	70.1	54.1
DeepBox	33.9	44.5	49.2	54.9	52.9	62.8	66.9	71.8	50.9
DeepMaskZoom	37.1	48.5	53.2	59.1	55.6	67.6	71.6	76.0	52.7
SharpMaskZoom	39.7	51.5	56.1	62.0	57.0	69.2	73.1	77.3	53.5
RefinedBox[1]	53.0	**58.7**	**60.6**	**62.4**	68.4	**74.1**	**75.8**	**77.4**	65.4
RefinedBox[2]	**53.7**	58.4	59.3	59.8	**68.9**	73.8	74.7	75.3	65.2
RefinedBox[3]	53.5	**58.7**	60.0	61.1	67.9	73.2	74.6	75.8	**65.5**
RefinedBox[4]	49.8	56.1	57.7	59.0	66.6	72.9	74.3	75.4	65.0

4.2. 在 VOC2007 数据集上的拟物性采样的评测

这里，我们首先将所提出的 RefinedBox 与其他物体推荐优化方法进行比较，包括 DeepBox [19] 和 MTSE [20]。图3展示了不同推荐优化方法之间的比较结果。我们选择 Edge Boxes [10] 来生成输入到这些优化算法的初始推荐，但是我们将其非极大值抑制的默认参数从 0.75 改为 0.9 来获得更多推荐框。我们发现，当 IoU 阈值为 0.5 和 0.7 时，所提出的 RefinedBox 均比其他方法实现了更高的物体检测召回率，且与其他方法之间的差距非常大。当每张图像仅使用一个推荐，RefinedBox 在 IoU 0.5 和 IoU 0.7 时的检测召回率分别为 43.2% 和 34.2%，而原始 Edge Box 的召回率分别为 29.1% 和 15.2%。此外，RefinedBox 可以与后续的物体检测共享卷积层，并且 RefinedBox 的其他层在计算上是轻量级的，因此 RefinedBox 是一个高效的检测框架。实际上，RefinedBox 和后续物体检测的总时间消耗类似于 Faster R-CNN [14]，每张图像大约需要 0.13 秒。



(a) DR *vs.* IoU

(b) DR *vs.* #WIN at IoU 0.5

(c) DR *vs.* #WIN at IoU 0.7

(d) MABO *vs.* #WIN

图 4: 在 PASCAL VOC2007 测试集上的评测结果（%）。(a) 展示了物体检测召回率 *vs.* IoU 重叠率阈值，每张图像使用 10 个物体推荐。(b) 和 (c) 分别展示了在 IoU 阈值 0.5 和 0.7 下，物体检测召回率 *vs.* 物体推荐数（#WIN）。(d) 显示了 MABO *vs.* 物体推荐数，每张图片最多使用 100 个推荐。

DeepBox 建立了一个独立的网络来对推荐框进行重新排序。而 MTSE 首先对图像进行分割，然后使用超像素来优化推荐框，然而，图像分割步骤是一个耗时的操作。因此，RefinedBox 更适合在许多应用中使用。

接下来，如图4所示，我们与最近的拟物性采样方法进行比较。RefinedBox 依然使用 Edge Boxes 作为输入，我们使用默认参数来对 Edge Boxes 进行评测。我们的方法在所有情况下都实现了最佳性能。当 IoU 为 0.7 时，对于物体检测召回率 *vs.* 物体推荐数量，RefinedBox 相对于其他方法的性能提升非常大。较高的检测召回率和较少的推荐将有利于后续的高层应用。最近，RPN 在物体检测中非常流行，但是我们提出的 RefinedBox 比它准确得多。每张图像仅包含 10 个物体推荐的 RefinedBox 的物体检测召回率和每张图像使用 100 个推荐的 RPN 差不多。从 RPN 到 RefinedBox 的提升证明了我们方法的有效性。只需很少的物体推荐，RefinedBox 就可以取得比其他方法更好的性能，包括最近著名的基于深度学习的 DeepMask [15] 和 SharpMask [16]。仅使用 30 个物体推荐，当 IoU 重叠率分别为 0.5 和 0.7 时，RefinedBox 可以实现 88.3% 和 76.4% 的物体检测召回率。这将满足很多应用对于少量但高质量的物体推荐框的要求。

表 3: 在 MS COCO 验证集上关于 DR 的评测结果（%）。RefinedBox[1]、RefinedBox[2]、RefinedBox[3]、和 RefinedBox[4] 分别表示基于 Edge Boxes、MCG、Selective Search 和 RPN 的 RefinedBox。

	DR (IoU=0.5)				DR (IoU=0.7)			
#WIN	10	30	50	100	10	30	50	100
BING	11.8	17.3	22.4	28.8	2.1	2.8	3.5	4.2
EdgeBoxes	17.7	26.2	30.7	37.7	11.4	18.1	21.8	27.5
GOP	11.3	22.7	30.0	41.1	7.3	13.8	18.1	25.1
LPO	15.1	26.6	32.2	42.1	7.0	14.4	18.4	24.7
MCG	24.5	36.7	42.5	50.6	14.7	23.5	28.1	34.6
Objectness	13.9	20.9	25.0	31.6	5.8	8.3	9.7	11.8
Rahtu	12.2	19.7	24.1	30.1	7.4	12.6	15.9	20.6
RandomPrim	12.9	22.4	28.2	37.2	6.2	11.7	15.3	21.4
SelectiveSearch	12.2	20.1	24.6	31.6	4.5	8.7	11.5	16.0
RPN	30.6	46.2	55.1	65.0	19.8	31.6	38.4	46.6
DeepBox	21.9	32.3	38.4	47.5	14.8	23.0	27.8	34.7
DeepMaskZoom	37.4	52.6	59.1	66.4	28.4	40.3	45.6	52.2
SharpMaskZoom	37.6	52.9	59.4	66.6	29.3	41.5	46.7	53.2
RefinedBox[1]	44.7	57.1	61.8	67.3	37.9	48.0	51.8	56.2
RefinedBox[2]	**45.4**	56.9	61.2	65.9	38.3	47.3	50.5	53.6
RefinedBox[3]	44.4	56.5	61.3	66.8	**38.5**	**48.9**	**53.1**	**57.6**
RefinedBox[4]	44.6	**57.3**	**62.4**	**68.1**	38.3	48.6	52.6	56.7

表 4: 在 MS COCO 验证集上关于 AR 和 MABO 的评测结果（%）。RefinedBox[1]、RefinedBox[2]、RefinedBox[3]、和 RefinedBox[4] 分别表示基于 Edge Boxes、MCG、Selective Search 和 RPN 的 RefinedBox。

	AR				MABO			
#WIN	10	30	50	100	10	30	50	100
BING	3.5	5.0	6.4	8.0	16.3	21.7	25.5	31.0
EdgeBoxes	9.9	15.1	17.9	22.3	21.0	28.2	32.0	37.6
GOP	6.6	12.5	16.4	22.6	15.2	27.4	33.9	42.3
LPO	6.9	13.4	16.9	22.6	20.8	30.4	35.3	43.1
MCG	13.6	21.3	25.3	30.9	27.5	37.9	42.9	49.5
Objectness	5.8	8.5	10.1	12.7	18.5	24.5	27.6	32.2
Rahtu	6.5	10.7	13.3	17.0	16.3	23.1	26.8	31.9
RandomPrim	6.1	11.1	14.4	19.7	18.3	27.2	32.1	39.4
SelectiveSearch	5.0	8.9	11.4	15.4	17.5	24.5	28.2	33.6
RPN	16.1	25.0	30.2	36.1	29.3	41.2	47.7	55.0
DeepBox	12.5	18.9	22.5	27.8	23.9	33.2	38.2	45.4
DeepMaskZoom	23.6	33.5	38.0	43.4	35.6	48.6	53.9	59.6
SharpMaskZoom	24.6	34.8	39.3	44.7	36.2	49.3	54.6	**60.3**
RefinedBox[1]	30.3	37.9	40.7	43.9	41.0	51.0	54.8	59.1
RefinedBox[2]	**31.3**	38.4	40.9	43.4	**42.1**	**51.8**	55.3	59.2
RefinedBox[3]	30.9	**38.8**	**41.8**	**45.2**	41.0	51.1	55.1	59.6
RefinedBox[4]	30.4	38.2	41.1	44.3	40.9	51.3	**55.4**	59.9

为了量化这些图，我们在表1中列出了相应的数字。与各种初始输入方法相比，RefinedBox 实现了更好的性能。在使用 Edge Boxs 且 IoU 阈值为 0.5 下，当每张图像使用 10、30、50 和 100 个物体推荐时，RefinedBox 的检测召回率比次优方法（SharpMaskZoom [16]）分别高 17.8%、8.8%、5.2% 和 0.8%。在 IoU 阈值为 0.7 下，当每张图像分别使用 10、30、50 和 100 个物体推荐时，基于 EdgeBoxes 的 RefinedBox 的检测召回率比 SharpMaskZoom 分别高 20.9%、15.5%、12.7% 和 8.4%。我们的目标是大幅减少物体推荐的数量，而评测结果正表明我们已经实现了这一目标。我们还注意到，RPN [14] 比传统的基于非深度学习的方法要好得多，而这也是为什么 Faster R-CNN 可以实现比 Fast R-CNN 更好的检测性能的关键原因。由于 RefinedBox 旨在从先前方法生成的所有推荐中选择和优化好的推荐，因此，影响最大的因素是输入的物体推荐的上限，即当具有足够数量的物体推荐时，之前方法能够取得的最大的检测召回率，而不是在一定数量的物体推荐下的性能。在 VOC2007 数据集上，Edge Boxes 可以通过足够数量的推荐实现较高的检测召回率，这就是基于 Edge Boxes 的 RefinedBox 性能最佳的原因。每张图像的 RefinedBox 的运行时间约为 0.06 秒，与传统的拟物性采样方法相比，这是非常快的。我们在表2中报告了各种对比方法的 AR 和 MABO。不出所料，RefinedBox 再次达到了最佳性能。

4.3. 在 VOC2007 数据集上的物体检测

因为物体检测是物体推荐的一个重要应用，所以我们根据在物体检测中的性能来测试不同拟物性采样算法的质量。我们将上述方法产生的物体推荐输入到著名的基于区域的物体检测框架 Fast R-CNN [3] 中，并使用上述联合训练算法来优化 RefinedBox。我们使用 [35] 中的设置。每张图像的前 1000 个物体推荐用于重新训练 Fast R-CNN 网络。所有这些方法都在 VOC2007 的 *trainval* 集上进行训练，并在测试集上进行测试。需要注意的是，每张图像仅使用前 10 个物体推荐来评测不同方法生成少量物体推荐的能力。

结果如表2所示。就 mAP 而言，RefinedBox（优化后的结果）分别比原始的 Edge Boxes、MCG、Selective Search 和 RPN 高 23.63%、24.03%、31.43% 和 10.93%。与其他拟物性采样方法相比，RefinedBox 也可以实现更高的检测性能。这些评测结果表明，RefinedBox 可以生成少量且高质量的物体推荐。有趣的是，RPN [14] 在物体检测方面的性能比 DeepBox [19]、DeepMask [15] 和 SharpMask [16] 稍好，而 RPN 在拟物性采样方面表现较差。这可能是因为 RPN 是针对于 Faster R-CNN 框架 [14] 中的物体检测精心设计的。我们在图5中展示了 RefinedBox 和其他方法关于物体检测的定性比较。我们可以看到，RefinedBox 显著提高了其他方法的检测性能。

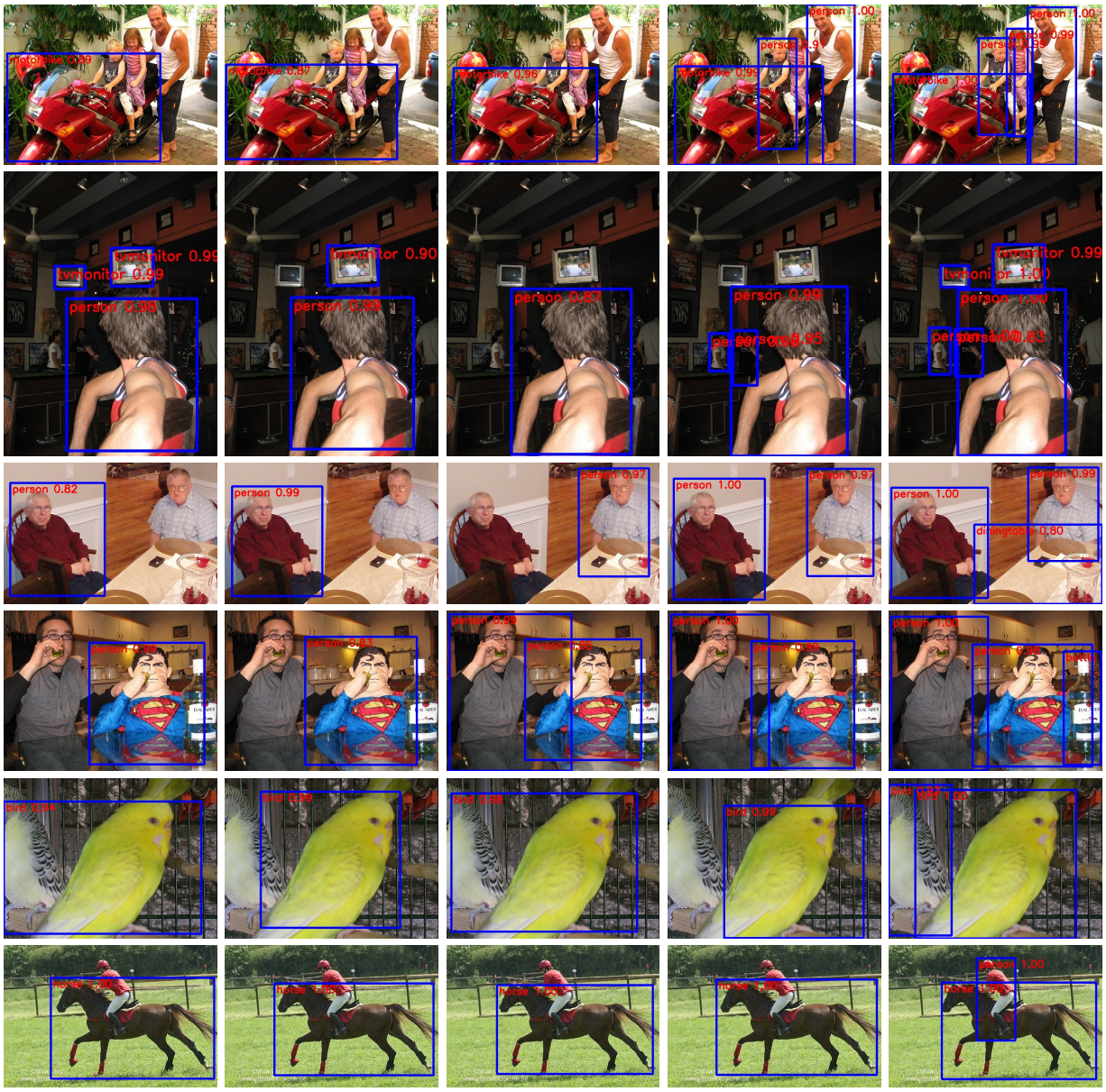


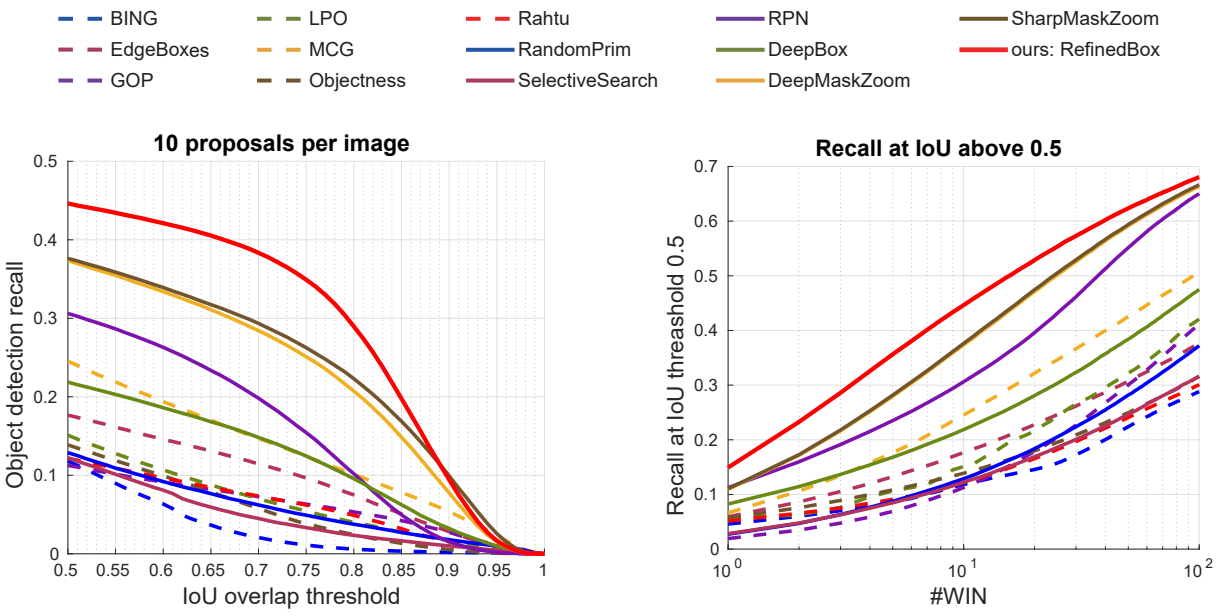
图 5: 仅使用前 10 个物体推荐进行物体检测的定性比较。这里，RefinedBox 使用 Edge Boxes [10] 作为输入。所有图像均来自 VOC2007 测试集。

4.4. 在 COCO 数据集上的拟物性采样的评测

在这一部分中，我们在 COCO 数据集上评测所提出的方法和其他方法。图6展示了 DR 和 MABO 的结果。在每个图中，RefinedBox 与其他方法之间都存在很大的差距，这表明了 RefinedBox 在生成少量的物体推荐中的有效性。表3中总结了关于 DR 的数值比较，表4中展示了各种方法的 AR 和 MABO。就所有指标而言，RefinedBox 的性能明显优于其他方法。SharpMask [16] 是次优方法，且仅比 DeepMask [15] 要好一点。需要注意的是，SharpMask 和 DeepMask 使用蒙版标注进行训练，而 RefinedBox 仅使用框标注进行训练。这进一步证明了进行适当的框优化以生成高质量物体推荐的重要性。

5. 总结

在本文中，我们提出一种使用重新排序和框回归的物体推荐优化方法。因为添加的层被设计为计算轻量级的，因此该方法是非常高效的。大量实验表明，RefinedBox 可以显著地减少以前算法生成的物体推荐的数量。由于所提出的优化网络可以很简单的被优化，



(a) DR *vs.* IoU

(b) DR *vs.* #WIN at IoU 0.5

(c) DR *vs.* #WIN at IoU 0.7

(d) MABO *vs.* #WIN

图 6: 在 MS COCO 验证集上的评测结果。RefinedBox 使用 RPN [14] 作为输入。(a) 物体检测召回率 *vs.* IoU 重叠率阈值，每张图像使用 10 个推荐。(b) 和 (c) 分别展示了当 IoU 阈值为 0.5 和 0.7 时，物体检测召回率 *vs.* 物体推荐数（#WIN）。(d)MABO *vs.* 物体推荐数，每张图片最多使用 100 个推荐。

因此我们发现可以将其与后续应用一起进行联合训练。在物体检测上的评测证明了 RefinedBox 的有效性。

局限性. 由于 RefinedBox 模块的效率与初始推荐的数量成正比，因此对于包含太多初始推荐的复杂图像，RefinedBox 的效率可能较低。由于骨干网络中的下采样操作，使得 RefinedBox 是在较小的特征图上进行的，因此，包含许多小物体的图像会影响其性能，就像物体检测方法一样 [2, 14, 39]。

未来工作. 少量且高质量的物体推荐可以满足许多高层应用的要求，如多标签图像分类 [6]、行人检测 [40]、深度多实例学习 [8] 等。使用更少但更准确的物体推荐，这些任务有望实现更好的性能。将来，我们计划将所提出的优化方法应用于其他高层应用中，例如从大量未标记数据中挖掘知识。

致谢. 本研究得到了项目编号为 2018AAA0100400 的“新一代人工智能重大项目”、“国家自然科学基金”（61620106008）和“天津自然科学基金”（17JCJQJC43700）的资助。

参考文献

References

[1] Y. Liu, S. Li, M.-M. Cheng, RefinedBox: Refining for fewer and high-quality object proposals, Neurocomputing 406 (2020) 106

– 116. doi:10.1016/j.neucom.2020.04.017.

[2] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: IEEE Conf. Comput. Vis. Pattern Recog., 2014, pp. 580–587.

[3] R. Girshick, Fast R-CNN, in: Int. Conf. Comput. Vis., 2015, pp. 1440–1448.

[4] K. He, G. Gkioxari, P. Dollar, R. Girshick, Mask R-CNN, in: Int. Conf. Comput. Vis., 2017, pp. 2961–2969.

[5] A. Arnab, P. H. Torr, Pixelwise instance segmentation with a dynamically instantiated network, in: IEEE Conf. Comput. Vis. Pattern Recog., 2017, pp. 441–450.

[6] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, HCP: A flexible CNN framework for multi-label image classification, IEEE Trans. Pattern Anal. Mach. Intell. 38 (9) (2016) 1901–1907.

[7] Y. J. Lee, K. Grauman, Predicting important objects for egocentric video summarization, Int. J. Comput. Vis. 114 (1) (2015) 38–55.

[8] J. Wu, Y. Yu, C. Huang, K. Yu, Deep multiple instance learning for image classification and auto-annotation, in: IEEE Conf. Comput. Vis. Pattern Recog., 2015, pp. 3460–3469.

[9] J. R. Uijlings, K. E. Van De Sande, T. Gevers, A. W. Smeulders, Selective search for object recognition, Int. J. Comput. Vis. 104 (2) (2013) 154–171.

[10] C. L. Zitnick, P. Dollár, Edge Boxes: Locating object proposals from edges, in: Eur. Conf. Comput. Vis., 2014, pp. 391–405.

[11] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, J. Malik, Multiscale combinatorial grouping, in: IEEE Conf. Comput. Vis. Pattern Recog., 2014, pp. 328–335.

[12] X. Qi, Z. Liu, J. Shi, H. Zhao, J. Jia, Augmented feedback in semantic segmentation under image level supervision, in: Eur. Conf. Comput. Vis., 2016, pp. 90–105.

[13] D. Li, J.-B. Huang, Y. Li, S. Wang, M.-H. Yang, Weakly supervised object localization with progressive domain adaptation, in: IEEE Conf. Comput. Vis. Pattern Recog., 2016, pp. 3512–3520.

[14] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, in: Adv. Neural Inform. Process. Syst., 2015, pp. 91–99.

[15] P. O. Pinheiro, R. Collobert, P. Dollár, Learning to segment object candidates, in: Adv. Neural Inform. Process. Syst., 2015, pp. 1990–1998.

[16] P. O. Pinheiro, T.-Y. Lin, R. Collobert, P. Dollár, Learning to refine object segments, in: Eur. Conf. Comput. Vis., 2016, pp. 75–91.

[17] M.-M. Cheng, Y. Liu, W.-Y. Lin, Z. Zhang, P. L. Rosin, P. H. Torr, BING: Binarized normed gradients for objectness estimation at 300fps, Computational Visual Media 5 (1) (2019) 3–20.

[18] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.

[19] W. Kuo, B. Hariharan, J. Malik, DeepBox: Learning objectness with convolutional networks, in: Int. Conf. Comput. Vis., 2015, pp. 2479–2487.

[20] X. Chen, H. Ma, X. Wang, Z. Zhao, Improving object proposals with multi-thresholding straddling expansion, in: IEEE Conf. Comput. Vis. Pattern Recog., 2015, pp. 2587–2595.

[21] Tao, Dapeng and Guo, Yanan and Yu, Baosheng and Pang, Jianxin and Yu, Zhengtao, Deep multi-view feature learning for person re-identification, IEEE Trans. Circ. Syst. Video Technol. 28 (10) (2017) 2657–2666.

[22] Han, Junwei and Zhang, Dingwen and Cheng, Gong and Liu, Nian and Xu, Dong, Advanced deep-learning techniques for salient and category-specific object detection: A survey, IEEE Signal Process. Mag. 35 (1) (2018) 84–100.

[23] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Int. Conf. Learn. Represent., 2015.

[24] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, A. Zisserman, The PASCAL visual object classes challenge 2007 (VOC2007) results, http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html (2007).

[25] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common objects in context, in: Eur. Conf. Comput. Vis., 2014, pp. 740–755.

[26] D. G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.

[27] S. Manen, M. Guillaumin, L. Van Gool, Prime object proposals with randomized prim's algorithm, in: Int. Conf. Comput. Vis., 2013, pp. 2536–2543.

[28] P. Rantalankila, J. Kannala, E. Rahtu, Generating object segmentation proposals using global and local search, in: IEEE Conf. Comput. Vis. Pattern Recog., 2014, pp. 2417–2424.

[29] E. Rahtu, J. Kannala, M. Blaschko, Learning a category independent object detection cascade, in: Int. Conf. Comput. Vis., 2011, pp. 1052–1059.

[30] I. Endres, D. Hoiem, Category-independent object proposals with diverse ranking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 222–234.

[31] P. Krahenbuhl, V. Koltun, Learning to propose objects, in: IEEE Conf. Comput. Vis. Pattern Recog., 2015, pp. 1574–1582.

[32] B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2189–2202.

[33] Z. Zhang, P. H. Torr, Object proposal generation using two-stage cascade SVMs, IEEE Trans. Pattern Anal. Mach. Intell. 38 (1) (2016) 102–115.

[34] C. Lu, S. Liu, J. Jia, C.-K. Tang, Contour box: Rejecting object proposals without explicit closed contours, in: IEEE Conf. Comput. Vis. Pattern Recog., 2015, pp. 2021–2029.

[35] Z. Zhang, Y. Liu, X. Chen, Y. Zhu, M.-M. Cheng, V. Saligrama, P. H. Torr, Sequential optimization for efficient high-quality object proposal generation, IEEE Trans. Pattern Anal. Mach. Intell. 40 (5) (2017) 1209–1223.

[36] S. He, R. W. Lau, Oriented object proposals, in: Int. Conf. Comput. Vis., 2015, pp. 280–288.

[37] P. Krähenbühl, V. Koltun, Geodesic object proposals, in: Eur. Conf. Comput. Vis., 2014, pp. 725–739.

[38] J. Hosang, R. Benenson, P. Dollár, B. Schiele, What makes for

effective detection proposals?, IEEE Trans. Pattern Anal. Mach. Intell. 38 (4) (2015) 814–830.

[39] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: IEEE Conf. Comput. Vis. Pattern Recog., 2017, pp. 2117–2125.

[40] S. Paisitkriangkrai, C. Shen, A. van den Hengel, Pedestrian detection with spatially pooled features and structured ensemble learning, IEEE Trans. Pattern Anal. Mach. Intell. 38 (6) (2016) 1243–1257.