CV论文阅读：密集物体检测的焦点损失

引言

现在最高精度的目标识别方法是从R-CNN推广的two-stages的方法。它是在candidate object locations的稀疏集合上面用了分类器。总之，one-stage检测器被运用于possible object locations的规则的密集样本，它的训练速度更快且更简单，但相比two-stages的精确率差了许多。本文我们调查为什么是这种情况。我们发现在密集探测器训练期间遇到的极端前景 - 背景类别失衡是主要原因。我们建议通过重塑标准的交叉熵损失来解决这个类别的不平衡问题，从而降低指定给良好分类示例的损失。

我们的新的Focal Loss专注训练在hard example的稀疏集合上，能够防止大量的easy negatives在训练中压倒训练器。为了评估loss的效率，我们设计了一个简单的密集检测器（dense detector），名叫RetinaNet。结果显示当RetinaNet在focal loss上训练之后，它能够匹配之前的one-stage detector的速度，并且优于已有的最佳水平的two-stage detector。

一、简介

现在的最佳水平的object detector都是基于two-stages，提案驱动的机理的。像R-CNN框架一样，第一步，生成候选物体定位（candidate object location）的稀疏集合；第二步，利用卷积神经网络构造分类器将第一步的集合分为前景和背景。通过一系列的进展，这两阶段的框架始终达到最高精度的COCO的benchmark挑战。

尽管two-stages探测器的成功，一个自然的问题是：一个简单的one-stage探测器能达到相似的精度吗？one-stage的检测器应用于对象位置、尺度和纵横比的规则的密集样本集。one-stage探测器，如YOLO和SSD的最新工作，显示出有希望的结果，产生更快的探测器，与目前最先进的two-stages探测器相比，高出其精度在10%-40%以内。

本文进一步把性能极限提高了：我们提出了一个one-stage object检测器，它第一次可以匹敌最佳水平的更复杂的two-stages检测器的state-of-art COCO AP，如Feature Pyramid Network FPN或Mask R-CNN。为了达到这个结果，我们在训练时把类别不均衡作为主要的障碍，它阻碍了one stage方法达到state-of-the-art的精度，并且我们提出了新的loss function消除这些障碍。

通过两阶段级联和采样启发式技术解决类R-CNN类探测器中的类不平衡问题。提议阶段（例如Selective Search，EdgeBoxes，DeepMask，RPN）将候选对象位置的数量迅速缩小到一个小数目（例如1-2k），滤除大部分背景样本。在第二个分类阶段，为了保持前景和背景之间的可管理平衡，执行抽样启发式，例如固定的前景背景比（1：3）或在线硬示例挖掘（OHEM）。

相比之下，one-stage检测器必须处理更大的一组candidate object locations，它在图像上规则采样。在训练中，这常常意味着枚举100K个位置，密集地覆盖空间位置、尺度和纵横比。虽然也可以应用类似的采样启发式算法，但由于训练过程仍然被easily classified background examples所支配，所以它们是低效的。这种低效率是在目标识别的典型问题，通常通过如bootstrapping、hard example mining的技术解决。

在本文中，我们提出了一种新的损失函数，作为处理类不平衡的先前方法的更有效的替代方案。损失函数是一个动态缩放的交叉熵损失，其中缩放因子随着对正确类别的置信度增加而衰减到零，见图1。直观地，这个缩放因子可以自动减小训练过程中easy example的贡献的比例并快速聚焦hard examples的模型。实验表明，我们提出的焦点损失使我们能够训练一个高精度，one-stage检测器，它能显著地胜过用sample heuristic或hard example mining训练one-stage的方法（之前的state-of-the-art方法）。最后，我们注意到，焦点损失的确切形式并不重要，我们表明其他样例可以达到类似的结果。

为了证明所提出的焦点损失的有效性，我们设计了一个简单的one-stage检测器RetinaNet，以其在输入图像中对象位置的密集采样而命名。

其设计包含高效的in-network feature pyramid和使用anchor boxes。它借鉴了来自[22, 6, 28，20]的各种最近的想法。ValnANET是高效和准确的，我们基于RESNET-101-FPN主干的最佳模型，在5 fps的运行下，实现了COCO测试DEAP AP 39.1，超过了先前从一个和两个阶段检测器得到的最佳单模型结果，见图2。

二、相关工作

经典对象检测器：滑动窗口范例，其中分类器应用于密集的图像网格，具有悠久而丰富的历史。 最早的成功之一是LeCun等人的经典著作《who applied convolutional neural networks》手写数字识别。 Viola和Jones 使用增强物体检测器进行人脸检测，导致这些模型被广泛采用。引入HOG [4]和积分通道特征[5]为行人检测提供了有效的方法。 DPMs 帮助将密集探测器扩展到更一般的对象类别，并在PASCAL上取得了最佳结果多年。 虽然滑动窗口方法是传统计算机视觉领域的领先检测范式，但随着深度学习的复兴，接下来描述的两阶段检测器迅速成为物体检测的主导。

Two-stages探测器：现代物体探测的主要范例是基于两阶段的方法。作为选择性搜索工作的开创者，第一阶段生成一组稀疏的候选投标，其中应包含所有对象，同时滤除大部分负面位置，第二阶段将投标分类为前景类/背景。R-CNN将第二阶段分类器升级为卷积网络，其准确性有了很大提高，并迎来了现代物体探测时代。 多年来，R-CNN在速度方面得到了改善，并且通过使用了学习对象提案。 区域提议网络（RPN）将提案生成与第二阶段分类器集成为一个单一的卷积网络，形成更快的RCNN框架。 已经提出了许多这个框架的扩展。

One-stage探测器：OverFeat 是第一个基于深度网络的现代一阶段目标探测器。最近SSD和YOLO重新关注一阶段方法。 这些探测器已经进行了速度调整，但其准确度落后于两种方法。 固态硬盘的AP降低了10-20％，而YOLO专注于更加极端的速度/精度平衡。参见图2.最近的工作表明，通过降低输入图像分辨率和数量 的提案，但是即使计算预算较大，单阶段方法的准确率也会降低[17]。 相反，这项工作的目的是了解一级探测器是否能够以相似或更快的速度运行时匹配或超过两级探测器的准确度。

我们的RetinaNet探测器的设计与以前的密集探测器有许多相似之处，特别是RPN引入的“锚点”概念以及SSD和FPN中使用的金字塔特征。 我们强调，我们的简单检测器能够达到不是基于网络设计创新的顶级结果，而是由于我们的新型损失。

Class Imbalance：像增强型检测器和DPMs这样的经典单级目标检测方法，以及像SSD这样的更新方法，在训练过程中都会面临很大的类别不平衡。这些检测器对每个图像评估104-105个候选位置，但只有少数位置包含对象。这种不平衡会导致两个问题：（1）由于大多数地点都是容易产生不利于学习信号的负面因素，所以培训效率低下；（2）整体而言，简单的负面因素会压倒训练并导致退化的模型。一个常见的解决方案是执行某种形式的hard negative mining，在训练过程中samples hard examples或更复杂的采样/重新测量方案。 相反，我们表明，我们提出的focal loss自然处理one-stage探测器所面临的类不平衡，并且使我们能够在没有采样的情况下有效地训练所有示例，并且不会让easy negative主导损失和梯度。

Robust Estimation：通过降低具有大错误的示例的损失（hard examples），减少异常值的贡献来设计鲁棒损失函数（例如Huber损失），这一点引起了很大兴趣。 相比之下，我们的焦点损失不是针对异常值，而是为了解决阶层不平衡问题，通过减轻inliers（easy examples）来解决阶级失衡问题，即使他们的数量很多，他们对总损失的贡献也很小。 换句话说，焦点损失执行robust loss的相反作用：它将训练集中在一组稀疏的hard examples上。

三、 Focal Loss

Focal Loss是被设计来针对one-stage object detection方案的，其中在训练中有在前景和背景类别之间的完全不平衡存在（1:1000）。先从对于binary classification的交叉熵（CE,cross entropy）损失来介绍Focal Loss。移除y∈{-1,1}是ground truth class，p∈[0,1]是模型对于标签y=1的估计概率。

为了方便标记，记Pt:重写

CE损失可以看作是图1中的蓝色（顶部）曲线。这种损失的一个显着特性，可以在其图中很容易看出，即使是很容易分类的例子（pt：5）也会产生损失 不平凡的幅度。当大量的easy examples叠加，这些小的损失值可以主导那些稀少的类。

3.1. Balanced Cross Entropy

针对class imbalance的常用方法是用一个权重参数α∈[0,1]对于类1，1-α对于类-1。实际应用上，α一般被设定为类频率的逆或者作为超参数，通过交叉验证设定。为了标记方便，定义αt，相似的定义Pt。α-balanced CE loss:

3.2. Focal Loss Definition

训练时遇到很大的类别不平衡会主导交叉熵损失。易分负样本在梯度和损失中占据主导地位。而α平衡了正负样本的重要性，它不会区别易分样本和难分样本。与之不同，作者将损失函数变形降低易分样本的权重，专注于训练难分负样本。

更加形式化地来说，作者加了(1-Pt)^γ到交叉熵上。γ是可以可以调节的专注参数γ>0。这样，Focal loss定义为：

说一下Focal loss的属性：（1）当一个样例被误分类，那么Pt很小，那么调制因子（1-Pt）接近1，损失不被影响；当Pt→1，因子（1-Pt）接近0，那么分的比较好的（well-classified）样本的权值就被调低了。（2）专注参数γ平滑地调节了易分样本调低权值的比例。γ增大能增强调制因子的影响，实验发现γ取2最好。

直觉上来说，调制因子减少了易分样本的损失贡献，拓宽了样例接收到低损失的范围。举例来说，当γ=2时，一个样本被分类的Pt=0.9的损失比CE小1000多倍。这样就增加了那些误分类的重要性（它们损失被缩了4倍多，当Pt<0.5且γ=2）。我们又用了α-balanced的Focal Loss的变体。作者发现它能提升一点点精度。我们注意到损失层的实现将用于计算p的sigmoid操作与损失计算相结合，导致更大的数值稳定性。

虽然在我们的主要实验结果中我们使用上面的焦点损失定义，但其确切形式并不重要。 在附录中，我们考虑焦点损失的其他实例，并证明这些可以同样有效。

3.3.Class Imbalance and Model Initialization

Binary分类模型是默认初始化为对于y=-1和y=1有相同的概率的。在这样的初始化之下，由于类不平衡，出现频率高的类会主导总的损失，在训练早期导致不稳定。为了对抗这个，作者提出“优先”的概念，在训练初期对于模型对于低频率的类（背景）估计的p给予“优先”。作者把这个“优先”（prior）记做 π ,设定它，以至于模型对于低频率类别（rare class）的样本的估计p很低，比如说0.001。这是模型初始化的改变，而不是损失函数的改变。我们发现这点能改进训练的稳定性（对于在类极不平衡的情况下的交叉熵和focal loss都有效）。

3.4.Class Imbalance and Two-stage Detectors

Two-stage detectors常用交叉熵损失，而不用 α-balancing 或者我们的方法。它们用两种途径解决这个问题：a.two-stage cascade（双阶段级联）b.biased minibatch sampling（有偏批量采样）。第一个级联阶段是一个对象建议机制[35,24,28]，它将几乎无限的可能对象位置集合减少到一两千个。 重要的是，选定的提案不是随机的，但可能与真实的对象位置相对应，这可以消除绝大多数简单的否定。 当训练第二阶段时，通常使用偏倚抽样来构造包含例如1：3比例的正面到负面示例的小型贴片。 这个比例就像一个隐含的“平衡因素”，通过抽样来实现。 我们提出的焦点损失旨在直接通过损失函数在一个阶段的检测系统中解决这些机制。

四、RetinaNet Detector

RetinaNet是由骨干网和两个特定任务子网组成的单一统一网络。 主干负责计算整个输入图像上的卷积特征映射，并且是一种现存（已有）的卷积网络。 第一个子网在骨干的输出上执行卷积对象分类; 第二个子网执行卷积bounding box regression。 这两个子网具有一个简单的设计，我们专门为一阶段密集检测而提出，请参见图3.虽然这些组件的细节有许多可能的选择，但大多数设计参数对精确值并不特别敏感，如实验。 我们接下来描述RetinaNet的每个组件。

Feature Pyramid Network Backbone:我们采用[20]中的特征金字塔网络（FPN）作为RetinaNet的主干网络。简而言之，FPN用自上而下的路径和横向连接增强了标准卷积网络，因此网络从单个分辨率输入图像有效地构建了一个丰富的多尺度特征金字塔，参见图3（a）-（b）。金字塔的每个级别都可以用于检测不同比例的对象。FPN改进了完全卷积网络（FCN）的多尺度预测，如RPN [28]和DeepMask式提案的增益所示，以及两阶段检测器如Fast R-CNN或Mask R-CNN。

继[20]之后，我们在ResNet架构之上构建了FPN [16]。我们构造一个金字塔，层次为P3到P7，其中l表示金字塔等级（第l层分辨率是第一层的 1/2l）。正如[20]中的所有金字塔级别都有C = 256个通道。金字塔的细节通常遵循[20]，并有一些适度的差异.虽然许多设计选择并不重要，但我们强调使用FPN骨干网时:使用仅来自最终ResNet层的特征的初步实验产生低AP。

Anchors：我们使用类似于[20]中RPN变体的translation-invariant anchor boxes。在金字塔等级P3到P7上，锚点的面积分别为322到5122。 如[20]中所述，在每个金字塔等级，我们使用三个长宽比为｛1:2; 1:1;2:1｝。 对于比[20]更密集的覆盖范围，我们在每个级别添加尺寸为｛20,21/3,22/3｝的原始3个纵横比锚点集合的锚点。 这在我们的设置中改善了AP。 总共有每个级别和级别的A = 9个锚点，它们覆盖了与网络输入图像相关的32-813像素的范围。

每个锚点被分配一个分类目标的长度为K的one-hot向量，其中K是对象类别的数量，以及一个4向量的box regression目标。我们使用RPN中的分配规则，但修改了多类别检测和调整后的阈值。具体而言，使用0.5的交叉口联合（IoU）阈值将锚点分配给ground-truth对象框; 如果他们的IoU位于[0,0.4），则为背景。由于每个锚点至多被分配一个对象框，因此我们将其长度K标签矢量中的相应条目设置为1，并将所有其他条目设置为0。如果未分配锚点，这可能会在[0.4,0.5)中发生重叠，它在训练中被忽略。Box regression targets被计算为每个锚点与其分配的对象框之间的偏移量，或者如果没有分配则被忽略。

Classification Subnet：分类子网预测每个锚点（A个）和对象类别（K个）在每个空间位置处的对象存在概率。该子网是每个FPN级别的小型FCN; 该子网的参数在所有金字塔级别共享。它的设计很简单。从给定金字塔等级的C通道输入输入特征映射，子网应用4个3x3个conv层，每个带有C个滤波器，每个都以ReLU作为激活函数，接着是带有KA滤波器的3x3 conv层。最后用sigmoid激活来输出每个空间位置的KA二进制预测，参见图3（c）。在大多数实验中我们使用C = 256和A = 9。

> 与RPN相比，我们的对象分类子网更深入，仅使用3x3卷积，并且不与box regression子网共享参数（下面介绍）。 我们发现这些更高层次的设计决策比超参数的特定值更重要。

Box Regression Subnet：与object classification子网络平行，作者在金字塔每个层都接到一个小的FCN上，意图回归每个anchor box对邻近ground truth object的偏移量。回归子网络的设计和分类相同，不同的是它为每个空间位置输出4A个线性输出。对于每个空间位置的A个anchor，4个输出预测anchor和ground-truth box的相对偏移。与现在大多数工作不同的是，作者用了一个class-agnostic bounding box regressor，这样能用更少的参数更高效。Object classification和bounding box regression两个网络共享一个网络结构，但是分别用不同的参数。

4.1. Inference and Training

Inference：RetinaNet的inference涉及把图片简单地在网络中前向传播。为了提升速度，作者只在每个FPN，从1k个top-scoring预测中提取box预测（在置信度阈值0.05处理之后）。多个层来的Top prediction聚在一起然后用NMS（非极大值抑制）以0.5为阈值。

Focal loss：作者在分类子网络输出的地方用了focal loss并发现在γ为2的时候效果比较好。同时RetinaNet在γ属于[0.5,5]有相对的鲁棒性。作者重点指出训练Retina时候，在每个采样图片里面，focal loss被加到所有的100K个anchor上面的。这与通常的heuristic sampling(RPN)或者 hard example mining（OHEM,SSD）选择anchor的一小部分集合（对于每个minibatch大概256）不同。作者用了特定的anchor（不是全部的anchor，因为大部分的anchor是easy negative在focal loss中有微小的作用）来归一化。最后α是用在设定在出现频率低的类别，有一个稳定的范围，它也和γ一起。这样能把两者融合，调两个参数。一般来说，当γ增大，α应该稍微减小（α=0.25和 γ=2效果最好）。

Initialization:作者在ResNet-50-FPN和ResNet-101-FPN的backbone上面做实验。基础模型是在ImageNet1K上面预训练的。除了最后一层，RetinaNet的子网络都是初始化为bias b=0和权值weight用高斯初始化 σ=0.01。classification子网络的最后一层的conv层，作者的bias初始化为b=-log((1-π)/π)其中π表示每个anchor在开始训练的时候应该被标记为背景的置信度 π。作者用 π= .01在所有的实验中。这样初始化能够防止大的数量的背景anchor在第一次迭代的时候产生大的不稳定的损失值。

Optimization:RetinaNet是用SGD训练的。作者用了同步的SGD在8个GPU上面，每个minibatch16张图，每个GPU训练2张图。所有的模型都是训练90K迭代的，初始学习率是0.01（会在60k被除以10，以及在80K除以10）。作者只用图像的横向翻转作为唯一的数据增广方式。权值衰减0.0001以及动量0.9。训练的损失是focal loss和标准的smooth L1 loss作为box回归。

五、Experiments

我们在具有挑战性的COCO基准[21]的边界框检测轨道上提出实验结果。对于训练，我们遵循常规练习[1,20]并使用COCO trainval35k split（来自火车的80k图像和来自40k图像val分裂的随机35k图像子集的联合）。我们报告病变和敏感性研究，通过评估minival split（val的其余5k图像）。对于我们的主要结果，我们在test-dev split上报告COCO AP，它没有公共标签，并且需要使用评估服务器。

5.1. Training Dense Detection

我们运行大量实验来分析密集检测的损失函数的行为以及各种优化策略。 对于所有实验，我们使用深度为50或101的ResNets [16]，并在顶部构建特征金字塔网络（FPN）[20]。 对于所有消融研究，我们使用600像素的图像比例进行训练和测试。

网络初始化：我们第一次尝试训练RetinaNet时使用了标准的交叉熵（CE）损失，而没有对初始化或学习策略进行任何修改。这很快就失败了，在训练期间网络发散。但是，简单地初始化我们模型的最后一层，使得检测对象的先验概率是π=：01（见x4.1）可以进行有效的学习。 用ResNet-50训练RetinaNet，这个初始化已经在COCO上产生了一个可观的AP 30.2。 结果对于π的确切值是不敏感的。所以我们使用π=：01进行所有实验。

5.2. Model Architecture Design

Anchor Density: one-stage检测系统中最重要的设计因素之一是其密度要覆盖可能图像框的空间。 Two-stages探测器可以使用区域合并操作将盒子分类到任意位置，比例和纵横比[10]。 相比之下，由于一阶段检测器使用固定的采样网格，所以在这些方法中实现高覆盖率盒子的流行方法是在每个空间位置使用多个“锚点”[28]以覆盖各种尺度和纵横比的盒子。

我们扫过FPN中每个空间位置和每个金字塔等级使用的尺度和纵横比锚点的数量。我们考虑从每个位置处的单个方形锚点到12个锚点的情况，每个位置跨越4个亚倍频程尺度（2k = 4， 对于k≥3）和3个纵横比[0.5,1,2]。 使用ResNet-50的结果显示在表1c中。 仅使用一个方形锚就能达到惊人的AP（30.3）。 但是，如果每个位置使用3个刻度和3个纵横比，则AP可以提高近4个点（至34.0）。 我们在这项工作中使用了此设置进行所有其他实验。

最后，我们注意到增加超过6-9个锚点并没有显示进一步的收益。 因此，尽管two-stages系统可以对图像中的任意方框进行分类，但性能的饱和度会下降。密度意味着two-stages系统的较高潜在密度可能不具有优势。

Speed versus Accuracy（速度与准确度）：较大的骨干网络可以提供更高的准确度，但也会降低推理速度。同样适用于输入图像比例（由较短的图像侧定义）。我们在表1e中显示这两个因素的影响。在图2中，我们绘制了RetinaNet的速度/精度折衷曲线，并将其与使用COCO test-dev上的公开数字的最近方法进行比较。该图显示RetinaNet由我们的焦点损失启动，形成了对所有现有方法的上限，从而打破了低精度的制度。具有ResNet-101-FPN和600像素图像比例（为简单起见，我们用RetinaNet-101-600表示）的RetinaNet与最近发布的ResNet-101-FPN更快的R-CNN [20]的精度相匹配，同时运行在122每个图像的ms数量为172 ms（均在Nvidia M40 GPU上测量）。使用更大的比例可以让RetinaNet超越所有两阶段方法的准确性，同时速度更快。为了实现更快的运行时间，只有一个工作点（500像素输入），使用ResNet-50-FPN比ResNet-101-FPN有所改进。解决高帧率制度可能需要特殊的网络设计，如[27]，并超出了这项工作的范围。我们注意到，在公布后，现在可以通过[12]中更快的R-CNN变体获得更快和更准确的结果。

5.3. Comparison to State of the Art

我们在具有挑战性的COCO数据集上评估RetinaNet，并将测试开发结果与近期最先进的方法（包括一阶段模型和两阶段模型）进行比较。 表2列出了我们使用刻度抖动训练的RetinaNet-101-800模型的结果，并且比表1e中的模型长1.5倍（带来1.3 AP增益）。 与现有的一阶段方法相比，我们的方法与最接近的竞争对手DSSD [9]相比，实现了5.9点AP健康差距（39.1 vs. 33.2），同时也更快，见图2.与最近的两阶段方法相比， 基于Inception-ResNet-v2-TDM [32]，RetinaNet比性能最高的Raster CN-R更快达到2.3点。 插入RetNeNet-32x8d-101-FPN [38]作为RetinaNet骨干，可进一步提高1.7个AP的性能，在COCO上超过40个AP。

六、Conclusion

作者将类别不平衡作为阻碍one-stage方法超过top-performing的two-stage方法的主要原因。为了解决这个问题，作者提出了focal loss，在交叉熵里面用一个调整项，为了将学习专注于hard examples上面，并且降低大量的easy negatives的权值。作者的方法简单高效。并且设计了一个全卷积的one-stage的方法来验证它的高效性。在具有挑战性的COCO数据集上面也达到了state-of-the-art的精度和运行时间。