厦门大学

实验报告

题目：基于 RetinaNet 的交通标志检测

学院：信息科学与技术学院

专业：计算机科学系

年级：2018 级

课程：计算机视觉

姓名：陈燊

学号：23020181154198

备注：

1. RetinaNet

RetinaNet 是 Facebook AI 团队在 2018 年提出的目标检测框架。RetinaNet 结合了 FPN 网络与 FCN 网络，在目标网络检测框架上并无特别亮眼点，其最大创新在于 Focal loss 的提出以及在 one-stage 目标检测网络的成功应用。Focal loss 是一种改进的交叉熵损失（cross-entropy, CE），它通过在原有的交叉熵损失上乘上一个衰减因子，使得 Focal loss 成功地解决了目标检测中的类别不平衡问题。RetinaNet 的作者通过后续实验成功表明 Focal loss 可以成功应用在 one-stage 目标检测网络中，并最终能以更快的速率实现与 two-stage 目标检测网络近似或更优的效果。

1.1 类别不平衡问题

常规的 one-stage 目标检测网络一般在训练时会先生成许多目标候选区域，然后再分别对这些候选区域进行分类与位置回归。而在这些生成的数万个候选区域中，绝大多数都是不包含待检测目标的图片背景，这样就造成了机器学习中经典的训练样本正负不平衡的问题。它往往使得网络损失被占大多数但包含信息量却很少的负样本所支配，从而无法得出一个能对模型训练提供正确指导的损失。

通常解决此问题的方法是负样本挖掘，或采用更复杂的算法来过滤负样本从而使正负样本数，从而维持一定比率的正负样本采样。而在 RetinaNet 中提出的 Focal loss 在 one-stage 目标检测任务上表现突出，同时有效地解决了目标检测中潜在的类别不平衡问题。

1.2 Focal loss

常规的交叉熵损失公式如下：

$$\mathrm{CE}(p,y)=\begin{cases}-\log(p) & \text{if } y=1\\ -\log(1-p) & \text{otherwise.}\end{cases}$$

当为正样本（y=1），则置信分数 p 越大 loss 越小，当为负样本（y=-1），则置信分数 p 越大则 loss 越小。这种损失有一个显著的特征，就是即使是很容易分类的例子（pt》0.5）也会造成损失，理论上这种 easy 的样本本不应对检测器产生影响，但是当有大量简单的样本存在时，即使他们各自产生的 loss 很小，他们 loss 的总和可能会对检测器产生巨大影响。

为此，作者提出了 Focal loss：

$$\mathrm{FL}(p_t)=-\alpha_t(1-p_t)^{\gamma}\log(p_t).$$

其中α_t为权重因子，用以控制正负样本产生的损失在总体损失中所占的比重。而$(1-\rho_t)^{\gamma}$为 Focal loss 的核心项，确保了容易的样本产生较小的损失，而困难的样本产生较大的损失，从而保证网络训练过程中主要的关注点都放在前景物体的检测中。RetinaNet 通过 Focal loss 有效得解决的 one-stage 网络在目标检测中的类别不平衡问题。

1.3 网络结构

RetinaNet 本质上是 Resnet + FPN +两个 FCN 子网络，其目标检测框架如图 1 所示。RetinaNet 的主干网络可选用任何有效的特征提取网络，如 VGG16 或 Resnet 系列。论文中作者分别尝试了 Resnet-50 与 Resnet-101。而 FPN 则是对 Resnet-50 里面自动形成的多尺度特征进行了强化利用，从而得到了表达力更强、包含多尺度目标区域信息的特征图集合。最后在 FPN 所得到的特征图集合上，分别使用两个 FCN 子网络来完成目标框类别分类与位置回归任务，其中这两个 FCN 子网络具有相同的网络结构，但是各自独立不共享参数。

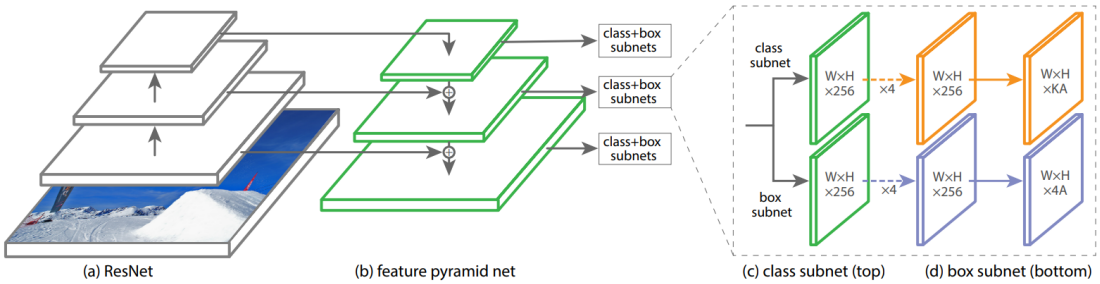

图 1 RetinaNet 网络框架图

2. 实验

本次实验利用 pytorch 实现了基于 RetinaNet 的交通标志检测算法。算法可以对警告（j）、禁令（z）、指示（s）、道路（l）、红绿灯（d）等五种交通标志进行检测，并在测试集上取得了非常好的表现。项目源码可见：https://github.com/chenshen03/Traffic-Sign-Detection-with-RetinaNet。

2.1 数据集

为了确保有足够多的数据来训练 RetinaNet 以得到更好的检测效果，实验中将 *Computer Vision Traffic Sign（CVTS）*和 *Tsinghua_Tencent_100K* 两个交通标注数据集结合起来用以训练。

2.1.1 CVTS 数据集

Computer Vision Traffic Sign，CVTS 是计算机视觉课程上所有人手工标注得到的交通标志数据集。一共有警告、禁令、指示、道路、交通灯五大类别，每个大类别又包含若干个小类别，共 77 个小类别。其部分类别示例如图 2 所示：

图 2 CVTS 类别示例

数据清理：CVTS 数据集共包含 2620 张图片，但由于标注质量参差不齐，存在漏标、误标、检测框未对齐等多种情况，原始数据无法直接用以网络训练。因此，我们对 CVTS 数据集进行了清理，去除掉图片质量和标注质量较差的数据。经过清理之后，得到的数据集共有 1250 张，分别包含了 754 个红绿灯标志、623 个禁令标志、345 个指路标志以及 244 个警告标志。显然，清理后的数据集存在着不同类别上的数据量不平衡的问题。

数据增强：为了确保网络在每个类别上的表现均衡，我们对清理后的数据集进行了数据增强，确保每个类别的图片量都在 1000 张以上。经过数据增强后，得到的数据集包含 6333 张图片，数据量大致确保了在小型网络上能够得到相对较好的结果。

2.1.2 Tsinghua_Tencent_100K 数据集

Tsinghua_Tencent_100K 是清华大学在 2016 年发布的交通标志数据集。该数据集是从 1,000,000 张腾讯街景图片中选取的 300,000 张交通标志图片构成。在去除掉出现次数少于 100 次的交通标志数据后，得到的数据集共有 37212 张图片，共包含指示标志、禁令标志、警告标志 3 个大类，其中又细分为 42 个小类。其交通标志示例如图 3 所示。

图 3 Tsinghua_Tencent_100K 类别示例

类别转换：为了将 Tsinghua_Tencent_100K 数据集应用到我们的五类交通标志检测任务中，我们对该数据集的类别进行了转换，将以 **i** 开头的类别转换为 **s**（指示标志），以 **p** 开头的类别转换为 **z**（禁令标志），以 **w** 开头的类别转换为 **j**（警告标志）。

数据合并：Tsinghua_Tencent_100K 的交通标志数据具有很好的标注质量，能够有效得提高算法的检测准确度，因此我们从该数据集的三大类中随机采样 1000 张图片，和 CVTS 数据集合并起来，从而构成了我们最后实验中所使用的数据集：*CVTS-TT100K*。

CVTS-TT100K 数据集共包含 9333 张图片，包含了警告标志、禁令标志、指示标志、道路标志和交通灯五个类别。我们从 CVTS-TT100K 中随机采样 1000 张图片作为测试集，剩下的 8333 张图片作为训练集。

2.2 网络训练

实验中使用 Resnet101 作为 RetinaNet 的图像特征提取框架，由于 Resnet101 要求的输入图片尺寸为 512*512，因此我们将 CVTS-TT100K 中的所有图片以及对应的 ground truth 都 resize 到 512*512。此外，由于交通标志的检测框普遍偏小，为了确保 RetinaNet 能够有效得检测到交通标志，我们将 anchor 的大小改变为[100, 500]的范围。

在训练中，我们设置参数 momentum 为 0.9，weight decay 为 2e-4，batch size 为 8，learning rate 为 1e-4，每 10 个周期下降为原来的十分之一。我们将 RetinaNet 在 CVTS-TT100K 数据集上训练了 80 个周期，其中训练损失曲线如图 4 所示。从图中可以看出，在第 50 个周期时，网络基本已经达到收敛，分类损失和检测损失基本都下降到了非常小的值。

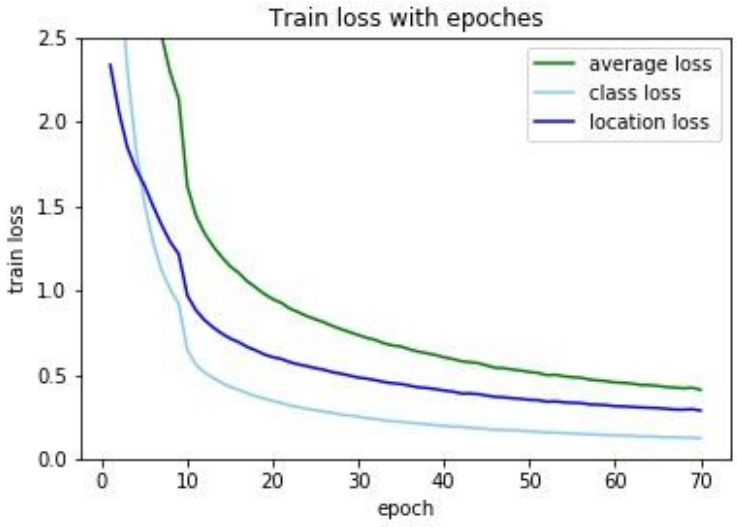

图 4 训练损失收敛曲线

2.3 实验分析

我们测试了模型在测试集上的 accuracy 和 recall，首先将 IOU 小于 0.5 的检测框全部去除掉，然后分析模型在不同边框分布下的表现，结果如表 1 所示。从表中可以看出，在检测框的范围为[0, 400]时，基本测定了所有生成检测框的综合表现，但是该情况下模型的 accuracy 和 recall 相并不是最好的，这说明模型在测试集上依旧有生成一些不符合要求的检测框，导致了整体的表现下降；而在检测

框的范围为[32, 96]时，模型的 accuracy 和 recall 同时达到最高，这说明测试集中交通标志的大小基本分布在这个区域，过大或过小都会导致检测框与实际情况偏移过多。

另外，表 1 中还给出了当检测框范围为[0, 512]时，不同类别下的 accuracy 和 recall。可以看到模型在警告标志（j）和指示标志（s）上有着较高的表现，这可能是由于数据集中融合了 Tsinghua_Tencent_100K 中标注质量很好的数据集，使得算法在这两个类别上的表现比较好；而在指路标志（d）上的表现就差强人意了，recall 更是低至 0.5061，通过可视化检测结果可以发现这是由于指路标志和指示标注过于相似，导致部分指路标志被误分类为指示标志。

表 1 不同类别和不同边框大小下的 accuracy 和 recall

iou	size	class	accuracy	recall
0.5	[0, 400]	z, j, s, l, d	0.8753	0.7877
0.5	[0, 32]	z, j, s, l, d	0.7597	0.6892
0.5	[32, 96]	z, j, s, l, d	0.9187	0.8391
0.5	[96, 400]	z, j, s, l, d	0.8824	0.7895
0.5	[0, 512]	z	0.8475	0.8412
0.5	[0, 512]	j	0.9290	0.8932
0.5	[0, 512]	s	0.9063	0.8463
0.5	[0, 512]	l	0.8710	0.8710
0.5	[0, 512]	d	0.7887	0.5061

为了使得上述表格的结果更为直观，我们给出了不同检测框范围下的 **acc-recall** 曲线，如图 5 所示。显然，当检测框范围为[32, 96]时，对应的 acc-recall 曲线有着最大面积，因此在该情况下模型的表现最好。此外，图 6 给出了 ground truth 和预测得到检测框的大小分布直方图。可以看到预测的检测框分布基本和 ground truth 一致，只是整体得到的检测框数量相对较少一点，因此直方图分布也较为平扁。而当检测框范围处于[20, 50]时，此时的检测框分布最多，这是由于交通标志普遍较小。

最后，我们测试了基于 RetinaNet 的交通标志检测算法在测试集上的 mAP 表现，测试结果为 **0.8586**。虽然 mAP 与官方结果有一定的差距，这主要是因为我们训练中所使用了数据集量相对较少（不足 10000 张），无法充分发挥深度神经网络的效果，所以这个结果也是在预料之中。总而言之，RetinaNet 在五类交通标志检测任务中也是取得了比较不错的表现。

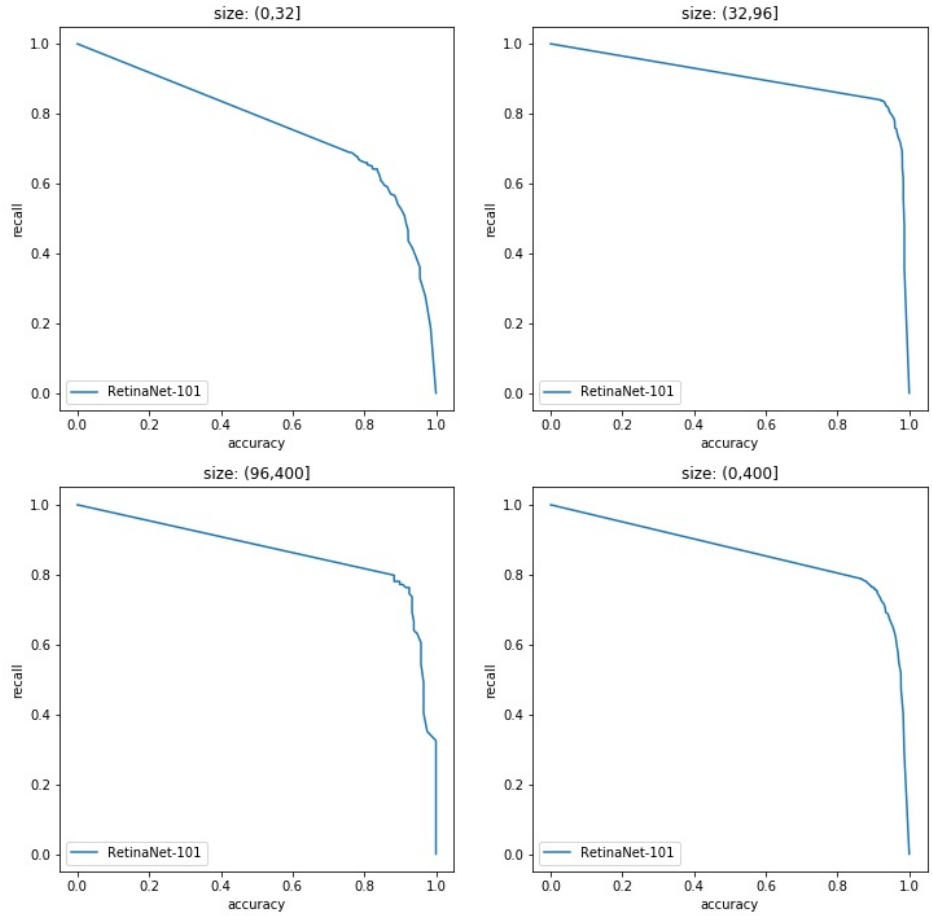

图 5 不同检测框范围下的 acc-recall 曲线

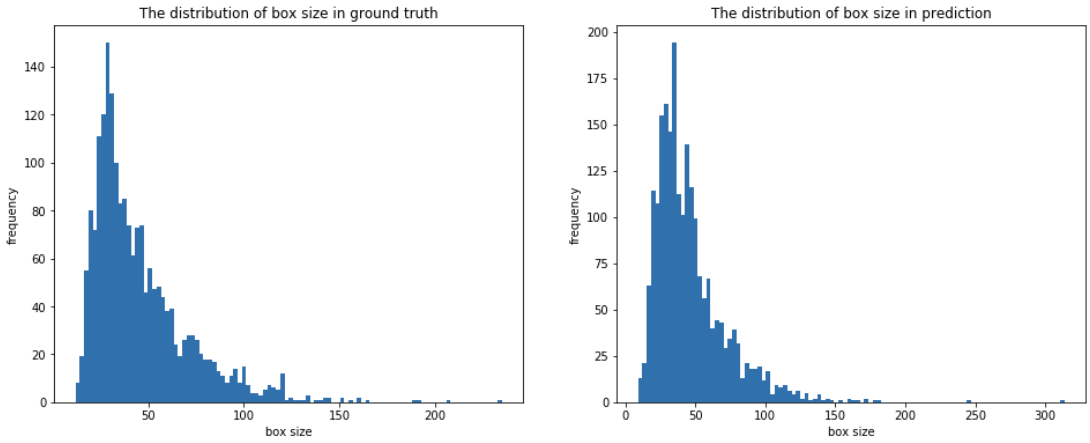

图 6 ground truth 和预测的检测框大小分布直方图

2.4 检测结果

图 7 给出了基于 RetinaNet 的交通标志检测算法在 CVTS-TT100K 上的部分检测结果示意图。可以看到算法能够很好得检测出在不同光照、不同大小的每个交通标志。

图 7 CVTS-TT100K 的部分检测结果示例

3. 结论

本次实验实现了基于 RetinaNet 的交通标志检测算法。为了保证数据集的质量，我们首先对 CVTS 数据集进行了清理和数据增强，然后将 Tsinghua_Tencent_100K 的标签转化为我们的目标类别，最后将这两个数据集进行采样合并，从而得到了我们实验中所采用的 CVTS-TT100K 数据集。该数据集大概有 10,000 张图片，确保网络能够充分得到训练。

在实验中，我们分析了模型的 acc-recall 指标以及 mAP 指标，实验结果表明了我们所采用的算法能够有效得解决五类交通标志检测任务，在每个类别上都达到了非常高的准确率和召回率。最后，我们给出了 CVTS-TT100K 数据集上的部分检测结果示例，进一步说明了算法的有效性和先进性。当然，我们训练得到的模型还不是非常优异，如果我们增加训练集的的质量和数量，RetinaNet 在五类交通标志检测任务中可以达到更好的表现。

参考文献

[1] Zhu, Zhe, et al. "Traffic-sign detection and classification in the wild." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[2] Li, Yuming, et al. "TAD16K: An enhanced benchmark for autonomous driving." Image Processing (ICIP), 2017 IEEE International Conference on. IEEE, 2017.

[3] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." *IEEE transactions on pattern analysis and machine intelligence* (2018).

[4] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[5] He, Kaiming, et al. "Identity mappings in deep residual networks." European conference on computer vision. Springer, Cham, 2016.

[6] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[7] Redmon, Joseph, and Ali Farhadi. "YOLO9000: better, faster, stronger." arXiv preprint (2017).

[8] Girshick, Ross. "Fast R-CNN." Proceedings of the IEEE international conference on computer vision. 2015.

[9] Ren, Shaoqing, et al. "Faster R-CNN: Towards real-time object detection with region proposal networks." Advances in neural information processing systems. 2015.

[10] Dai, Jifeng, et al. "R-FCN: Object detection via region-based fully convolutional networks." Advances in neural information processing systems. 2016.

[11] Lin, Tsung-Yi, et al. "Feature Pyramid Networks for Object Detection." CVPR. Vol. 1. No. 2. 2017.

[12] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[13] Zitnick, C. Lawrence, and Piotr Dollár. "Edge boxes: Locating object proposals from edges." European conference on computer vision. Springer, Cham, 2014.

[14] Uijlings, Jasper RR, et al. "Selective search for object recognition." International journal of computer vision 104.2 (2013): 154-171.