Towards Adversarially Robust Object Detection

承子杰
dept. AMSS（数学与系统科学研究院）
*of CAS (*中国科学院*)*
chengzijie22@mails.ucas.ac.cn
202228000243001

I. 主要内容

目标检测是计算机视觉中较为基础的研究方向之一，在无人驾驶等领域具有广泛的应用。最近的一些研究指出一些蓄意设计的对抗性样本对于目标检测器的性能会产生严重影响，但还未有研究涉及如何提高检测器的鲁棒性。为此，本文作者分析了对目标检测器不同的攻击类别，希望从中找出它们的共同点，从而提出一种对抗训练方法来提高检测器的鲁棒性。最后作者在 PASCAL-VOC 和 MS-COCO 等数据集上进行大量实验，证明了该方法的有效性。

II. 目标检测问题与攻击分析

目标检测任务将一张图像 $x \in [0,255]^n$ 作为输入，输出 K 个目标预测 $\{p_k, b_k\}^K$，其中 $p_k \in \mathbb{R}^C$ 表示在 C 个类别上的预测概率，$b_k = [x_k, y_k, w_k, h_k]$ 表示预测目标的包围盒。最后使用 NMS 方法作为后处理移除冗余检测来获得最后结果，其基本流程可见图 1。

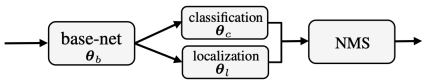

图 1. 目标检测器架构

若我们设 θ 为检测器参数，L 为损失函数，$f(x)$ 为目标检测器，则模型训练目标可表示为

$$\min_{\theta} L(f_\theta(x), \{y_k, b_k\})$$

在实际计算中，我们通常将上述损失函数分开表示为分类损失与位置损失，即可表示为

$$\min_{\theta}(L_{cls}(f_\theta(x), \{y_k, b_k\}) + L_{loc}(f_\theta(x), \{y_k, b_k\}))$$

从上述分析可以看出，目标检测可以分解为分类与位置检测两个子任务，虽然使用损失函数不同，但两者共享 base-net 进行特征提取，本质可以看成一个多任务学习问题。

Attacks for Object Detection	Components			
	loss_{cls}		loss_{loc}	
	T	N	T	N
ShapeShifter [6]	✓			
DFool [32], **PhyAttack** [11]		✓		
DAG [57], **Transfer** [55]	✓	✓		
DPatch [31]	✓		✓	
RAP [23]		✓	✓	
BPatch [22]		✓		✓

图 2. 不同攻击分类

根据上述检测器的设计框架，作者认为虽然攻击的方法有很多种，但设计原则都是利用单个任务损失函数的变体或其组合来实现对检测器进行攻击。为此，作者将一些常见的攻击进行了分类 (对检测类别进行攻击和对检测位置进行攻击)，见图 2。同时根据图 1，作者认为对单个损失函数攻击能对整个检测器产生影响主要基于以下两个原因：

- 由于 base-net 是分类与位置检测的共享网络，对 base-net 的弱点攻击会导致在两个任务上性能均下降。
- 由于最后分类与位置检测会被 NMS 耦合到一起，单个任务会影响最后结果。

III. 面向攻击的鲁棒检测

受前一小节的启发，作者从任务丢失和任务梯度未对齐两个方面分析单任务损失对检测器鲁棒性的影响。

在任务丢失角度，作者认为由于 base-net 的存在，两个任务是相互影响的。对一个任务的攻击必然会影响 base-net 的特征提取，进而导致另一个任务性能的下降。为证明这一个观点，作者采用控制变量法。在考虑分类时，将定位因子边缘化，从而将问题转化为多类别分类任务；在考虑定位时，可以将类别信息边缘化，从而转化为类别未知下的目标检测问题。具体实验中，我们采用 NMS 之前的结果进行性能衡量，并采用 PDG 方法进行攻击。同时我们认为当锚框与标注数据的 IoU>0.5 时该锚框是正样本。在分类任务中，主要关心锚框的分类精度；在定位任务中，主要关心的是包围盒与标注样本的平均 IoU。实验结果如

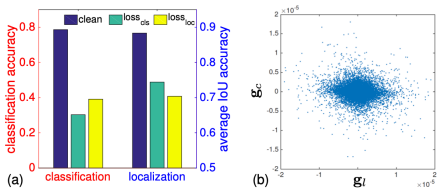

图 3. 任务损失对检测器鲁棒性实验

图 3 所示。图 a 结果可以说明两个任务确实是相互影响，证实了作者的观点。

对于任务梯度未对齐，作者认为两个任务的梯度存在一定的共同方向，但并没有完全对齐，从而导致任务梯度的错位，这可能会混淆后续的对抗性训练。为此作者画出了两个任务损失函数梯度的逐点梯度图（图 3(b)）进行进一步分析。通过分析可以发现两个明显问题：

- 两个任务的梯度大小不相等（未呈球形），从而存在任务不平衡的问题。
- 两个任务的梯度方向也不一致（未呈对角线），从而存在任务梯度冲突的可能性。

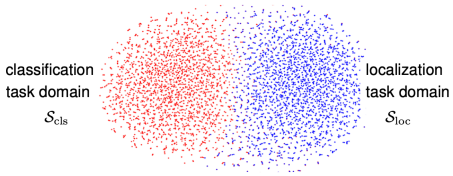

图 4. 两个任务的梯度域

最后，给定单一干净的图像 x，作者画出了在不同对抗样本下，两个任务的梯度域（图 4）。可以发现两个任务的梯度域并未完全分离，重合部分说明的是两个任务间的相互影响，分离部分说明的是两个任务的梯度未对齐。

基于上述分析，文章作者提出了改进的训练目标，即

$$\min_{\theta}[\max_{\bar{x}\in S_{cls}\cup S_{loc}} L(f_\theta(\bar{x}),\{y_k,b_k\})]$$

其中，S_{cls} 和 S_{loc} 分别表示每个任务的可行域，即

$$S_{cls} \triangleq \{\bar{x}|arg\max_{\bar{x}\in S_x} L_{cls}(f(\bar{x}),\{y_k\})\}$$
$$S_{loc} \triangleq \{\bar{x}|arg\max_{\bar{x}\in S_x} L_{loc}(f(\bar{x}),\{b_k\})\}$$
$$S_x = \{z|z\in B(x,\epsilon)\cap[0,255]^n\}$$
$$B(x,\epsilon) = \{z|\|z-x\|_\infty \le \epsilon\}$$

需要注意的是，与传统对抗性训练相比在分类上存在几个重要区别：

- **存在多任务源**，即有多个不同的监督源可用于对抗生成与训练。
- **面向任务的域约束**，即引入 $\bar{x}\in S_{cls}\cup S_{loc}$，将可行集约束至一个图像集，并对这个图像集的分类损失与定位损失进行最大化处理。该方法的优点是可以生成由每个任务指导的对抗样本，而不会受到任务之间的相互干扰。

最后，作者指出，如果我们将面向任务的域放宽到 S_x，并设置与整个图像相对应的包围盒坐标，为每个图像分配一个单一的类别标签，则训练转变为分类设置下的常规对抗训练。从而，文章提出的鲁棒性检测对抗训练，就可以被认为是分类设置下常规对抗训练的一种自然的泛化。整个算法伪代码流程可见图 5。

Algorithm 1 Adversarial Training for Robust Detection

Input: dataset $\mathcal{D}$, training epochs T, batch size S, learning rate γ, attack budget ϵ

for $t = 1$ **to** T **do**
  for random batch $\{\mathbf{x}^i, \{y_k^i, \mathbf{b}_k^i\}\}_{i=1}^S \sim\mathcal{D}$ **do**
    $\cdot\ \tilde{\mathbf{x}}^i \sim B(\mathbf{x}^i, \epsilon)$
    compute attacks in the classification task domain
    $\cdot\ \bar{\mathbf{x}}_{\text{cls}}^i = \mathcal{P}_{\mathcal{S}_\mathbf{x}}\big(\tilde{\mathbf{x}}^i + \epsilon\cdot\text{sign}\big(\nabla_\mathbf{x}\text{loss}_{\text{cls}}(\tilde{\mathbf{x}}^i,\{y_k^i\})\big)\big)$
    compute attacks in the localization task domain
    $\cdot\ \bar{\mathbf{x}}_{\text{loc}}^i = \mathcal{P}_{\mathcal{S}_\mathbf{x}}\big(\tilde{\mathbf{x}}^i + \epsilon\cdot\text{sign}\big(\nabla_\mathbf{x}\text{loss}_{\text{loc}}(\tilde{\mathbf{x}}^i,\{\mathbf{b}_k^i\})\big)\big)$
    compute the final attack examples
    $\cdot\ \mathbf{m} = \mathcal{L}(\bar{\mathbf{x}}_{\text{cls}}^i,\{y_k^i,\mathbf{b}_k^i\}) > \mathcal{L}(\bar{\mathbf{x}}_{\text{loc}}^i,\{y_k^i,\mathbf{b}_k^i\})$
    $\cdot\ \bar{\mathbf{x}}^i = \mathbf{m}\odot\bar{\mathbf{x}}_{\text{cls}}^i + (1-\mathbf{m})\odot\bar{\mathbf{x}}_{\text{loc}}^i$
    perform adversarial training step
    $\cdot\ \boldsymbol{\theta} = \boldsymbol{\theta} - \gamma\cdot\nabla_{\boldsymbol{\theta}}\frac{1}{S}\sum_{i=1}^S\mathcal{L}(\bar{\mathbf{x}}^i,\{y_k^i,\mathbf{b}_k^i\};\boldsymbol{\theta})$
  end for
end for
Output: learned model parameter $\boldsymbol{\theta}$ for object detection.

图 5. 算法伪代码

IV. 实验结果

作者在 PASCAL-VOC 和 MS-COCO 数据集上进行训练，并使用了 DAG 和 RAP 对算法性能进行测试，测试结果见图 6 和图 7。

architecture		DAG [57] STD	DAG [57] ours	RAP [23] STD	RAP [23] ours
SSD	+VGG16	0.3	28.5	6.6	44.9
RFB	+ResNet50	0.4	27.4	8.7	48.7
FSSD	+DarkNet53	0.3	29.4	7.6	46.8
YOLO	+DarkNet53	0.1	27.6	8.1	44.3

图 6. PASCAL-VOC 数据集上实验结果

model	architec.	backbone	clean	attack
standard	SSD	VGG16	39.8	2.8
ours	SSD	VGG16	27.8	16.5
	SSD	DarkNet53	20.9	18.8
	SSD	ResNet50	18.0	16.4
	RFB	ResNet50	24.7	21.6
	FSSD	DarkNet53	23.5	20.9
	YOLO	DarkNet53	24.0	21.5

图 7. MS-COCO 数据集上实验结果

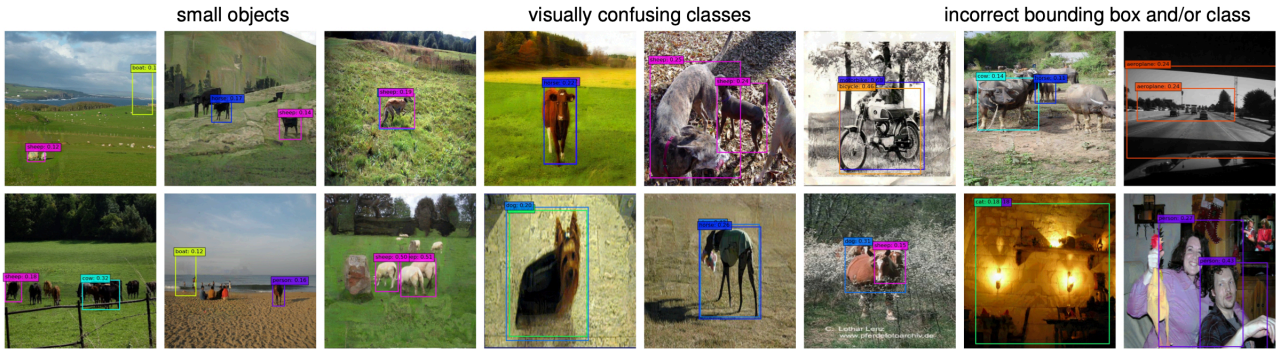

图 8. 检测失败例子

通过在 PASCAL-VOC 和 MS-COCO 数据集上的实验结果表明，该算法确实提高了目标检测器的鲁棒性，这对监控场景等领域的视觉任务具有十分重要的意义。

此外作者给出了鲁棒目标检测器检测失败的例子 (图 8)，并指出具有挑战性的小图像依旧是鲁棒目标检测器难以处理的对象。作者认为对于该挑战，可能需要研究性能更优的目标探测器来解决该问题，并进一步加深对鲁棒性和目标检测架构的研究。

References

参考文献

[1] Zhang H , Wang J . Towards Adversarially Robust Object Detection[J]. IEEE, 2019.