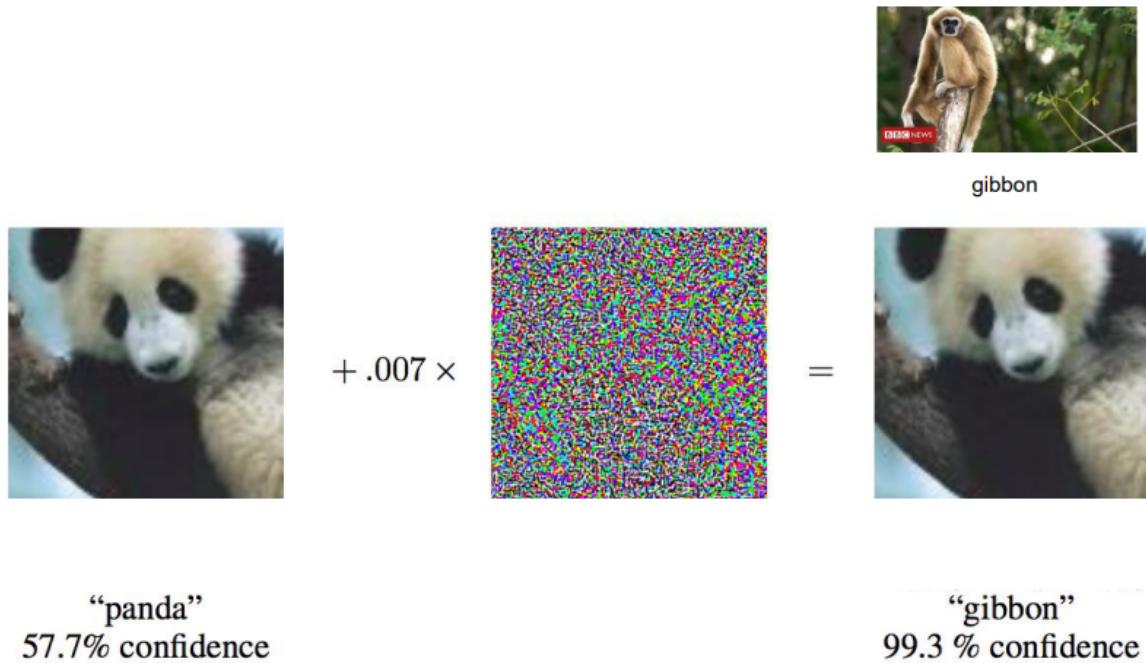


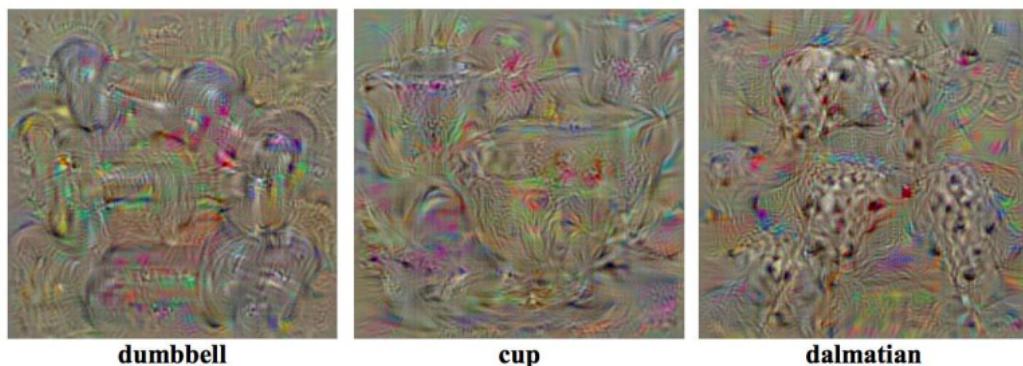
Lecture10 Adversarial Examples



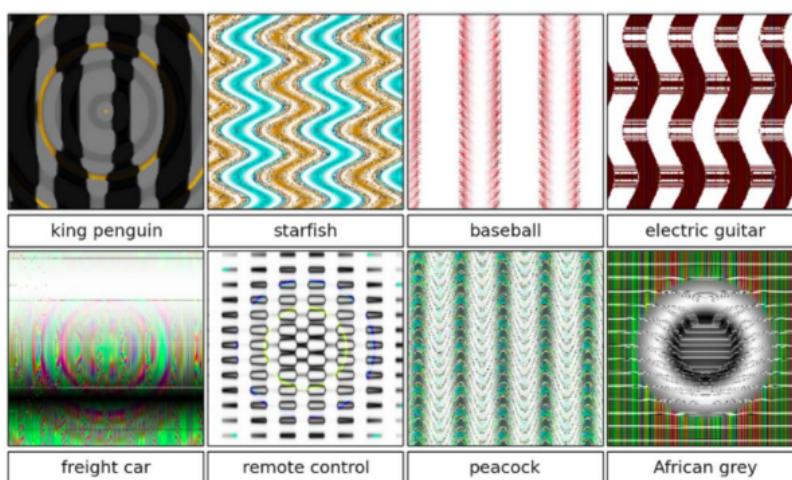
1. 对抗样本介绍

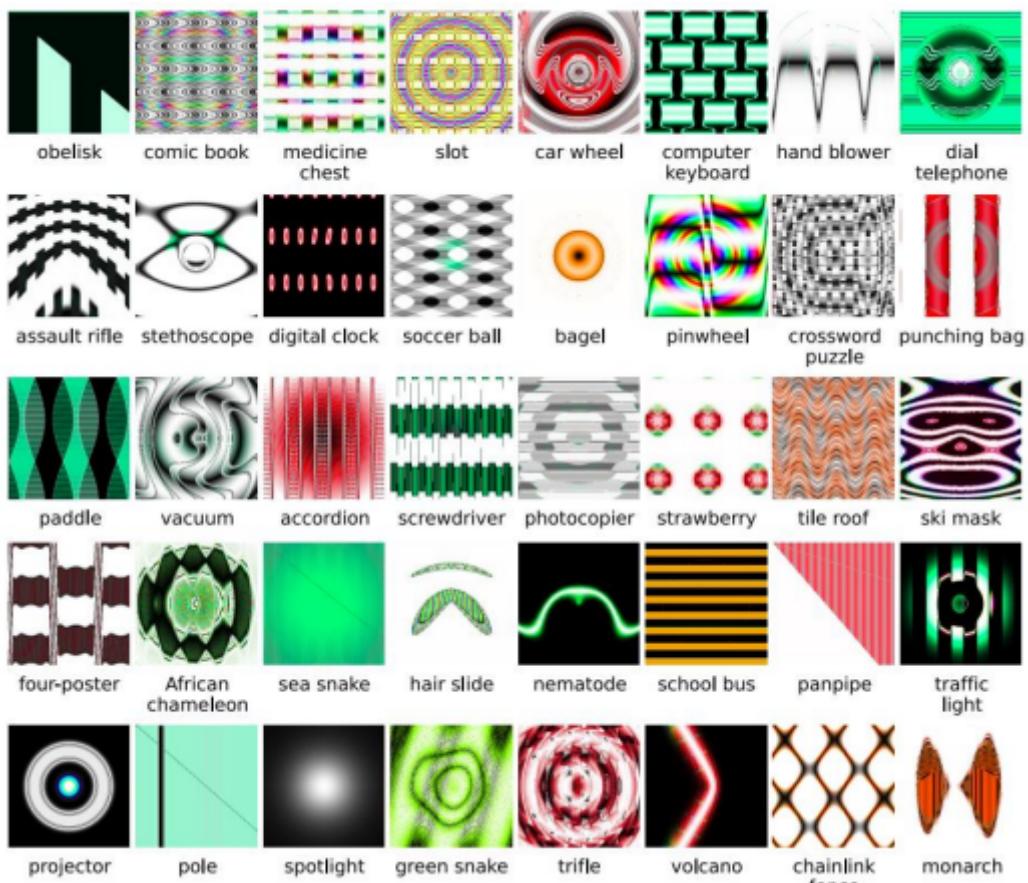
生成更容易扰乱的输入

- 我们可以使用梯度上升生成奇怪的图像，以最大限度地激活给定的单位

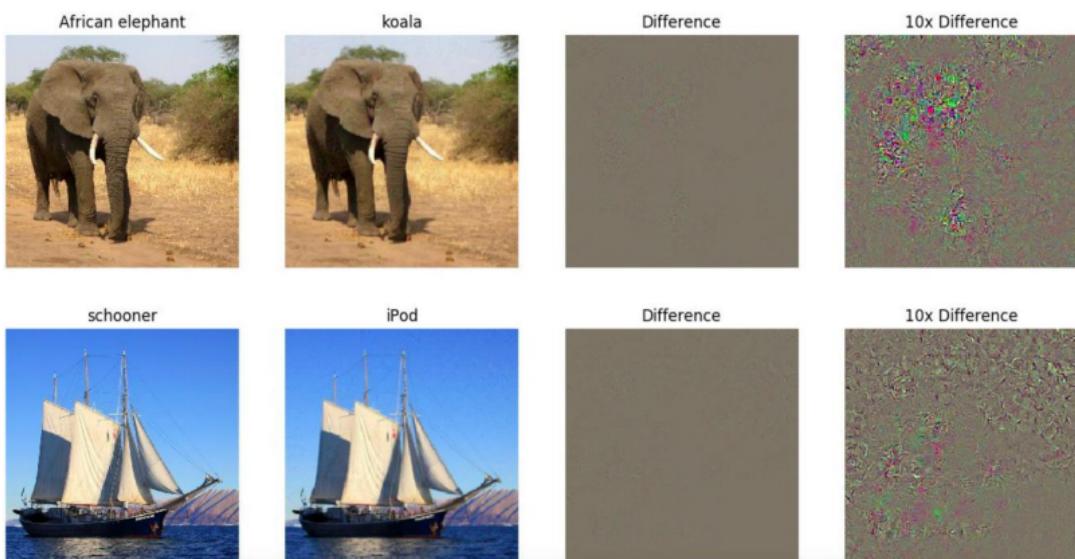


- 它很容易产生人类感知上毫无意义上的图像，但网络以很高的信心把它划分为给定的类别





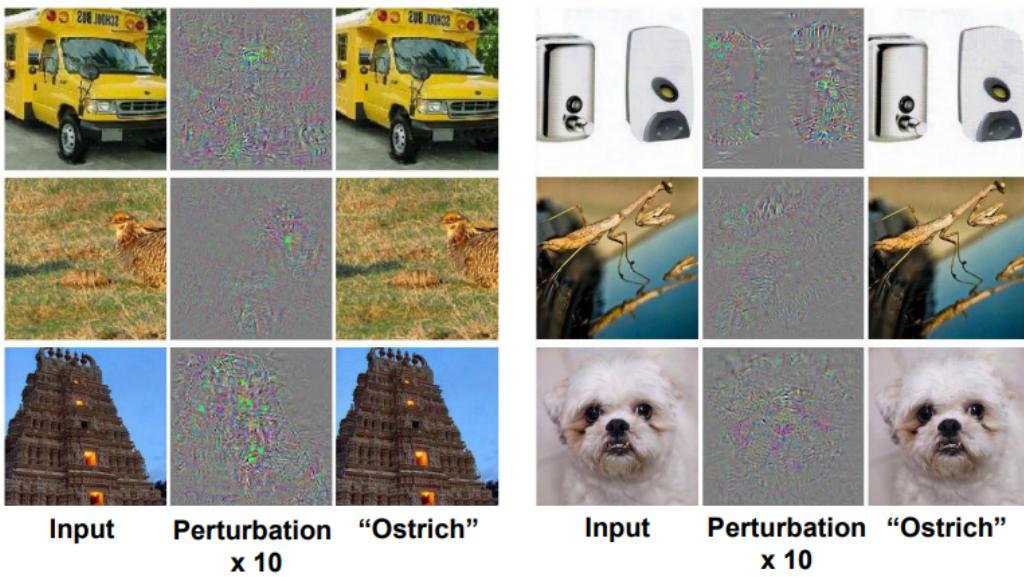
- 我们可以通过不知不觉地干扰输入图像来“欺骗”神经网络，从而使其分类错误



- 这些东西就是对抗样本

2. 如何生成对抗样本

找到最小的对抗性扰动



- 从正确分类的图像 x 开始
- 找到扰动 r 最小情况 $\|r\|_2$ 使得
 - $x + r$ 会被错误的分类 (分类为特定的目标类)
 - 所有的 $x + r$ 的值都在有效的范围内
- 约束非凸优化，可以用 L-BFGS 实现

梯度上升 Gradient Ascent

- 与其寻找最小可能的扰动，不如在预期的方向上采取小的梯度步长

减少分类正确的梯度/增加分类正确的 Loss

- 对于分类正确的类 y^*

$$x \leftarrow x - \eta \frac{\partial f(x, y^*)}{\partial x} \quad x \leftarrow x + \eta \frac{\partial L(x, y^*)}{\partial x}$$

增加分类错误的梯度/减少分类正确的 Loss

- 对于分类错误的类 \hat{y}

$$x \leftarrow x + \eta \frac{\partial f(x, \hat{y})}{\partial x} \quad x \leftarrow x - \eta \frac{\partial L(x, \hat{y})}{\partial x}$$

欺骗一个线性分类器

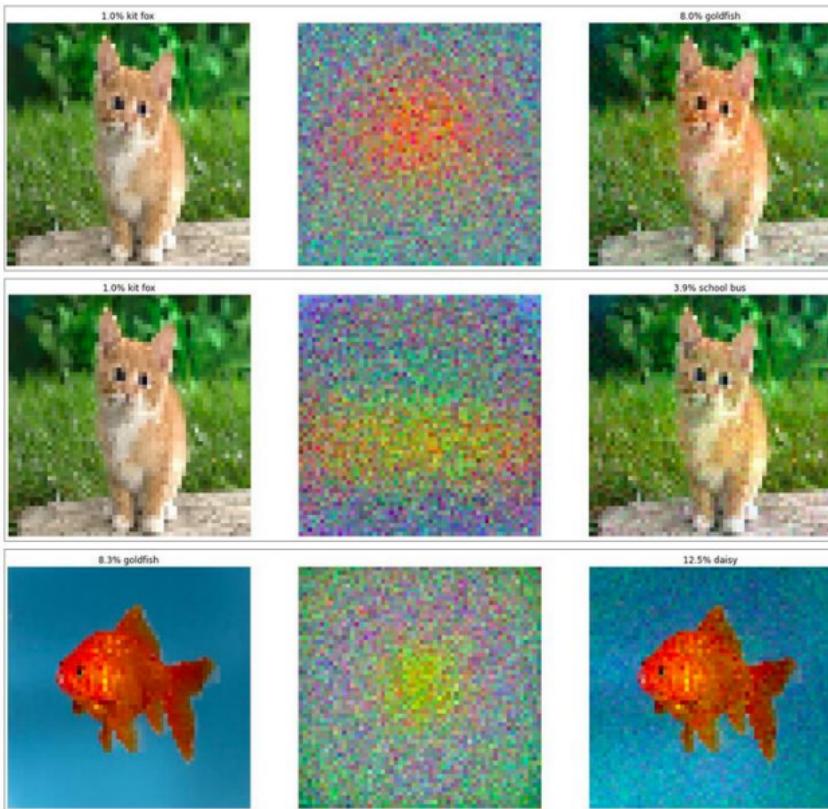
如果我们使用增加分类错误的梯度，即

$$x \leftarrow x + \eta \frac{\partial f(x, \hat{y})}{\partial x}$$

那么对于一个线性分类器来说 $f(x, \hat{y}) = w^T x$, 求导后有 $\eta \frac{\partial f(x, \hat{y})}{\partial x} = \eta w$

$$x \leftarrow x + \eta w$$

- 为了欺骗线性分类器，在测试示例中添加目标类权重的小倍数



线性情况的分析

权重为 w 的分类器对对抗样本 $x + r$ 的响应

$$w^T(x + r) = w^T x + w^T r$$

- 假设像素值具有精度 ϵ , 即分类器通常在 $\|r\|_\infty \leq \epsilon$ 的情况下期望将 x 和 $x + r$ 预测为一类
- 如何选择一个 r 去最大化增长激活 $w^T r$, 在给定约束 $\|r\|_\infty \leq \epsilon$ 的情况下?

$$r = \epsilon \operatorname{sgn}(w)$$

- $\operatorname{sgn}(w)$: 符号函数, 取 w 的符号

那么, 公式为

$$w^T(x + r) = w^T x + \epsilon w^T \operatorname{sgn}(w)$$

- 如果 w 是一个 d 维度的权重矩阵，且每个元素的扰动 $\epsilon \operatorname{sgn}(w)$ 的平均值是 m ，那么激活增加了
 - ϵdm
 - 维度越高，就越容易对输入进行许多小的更改，从而导致输出发生较大的变化

Toy 示例

x	2	-1	3	-2	2	2	1	-4	5	1
w	-1	-1	1	-1	1	-1	1	1	-1	1

$$w^T x = -2 + 1 + 3 + 2 + 2 - 2 + 1 - 4 - 5 + 1 = -3$$

$$\sigma(w^T x) = \frac{1}{1 + e^{(-3)}} = 0.047$$

x	2	-1	3	-2	2	2	1	-4	5	1
w	-1	-1	1	-1	1	-1	1	1	-1	1
$x + r$	1.5	-1.5	3.5	-2.5	2.5	1.5	1.5	-3.5	4.5	1.5

这里 $d = 10, m = 0.5$

$$w^T(x + r) = -3 + 10 * 0.5 = 2$$

$$\sigma(w^T(x + r)) = \frac{1}{1 + e^{-2}} = 0.88$$

生成对抗样本

最速梯度符号法 Fast gradient sign method

找到分类正确的类 y^* 的 loss 梯度，取元素符号，按结果方向更新

$$x \leftarrow x + \epsilon \operatorname{sgn} \left(\frac{\partial L(x, y^*)}{\partial x} \right)$$

迭代梯度符号法 Iterative gradient sign method

采取多个小步骤，直到分类错误，每个小步骤将原来的图像偏移邻近的 ϵ 一次

最小可能类方法 Least likely class method

尝试以最小初始分数将图像错误分类为 \hat{y} 类

$$x \leftarrow x - \epsilon \operatorname{sgn} \left(\frac{\partial L(x, \hat{y})}{\partial x} \right)$$

三种方法示例

$\epsilon = 32$ 的情况



Clean image



"Fast"; L_∞ distance to clean image = 32

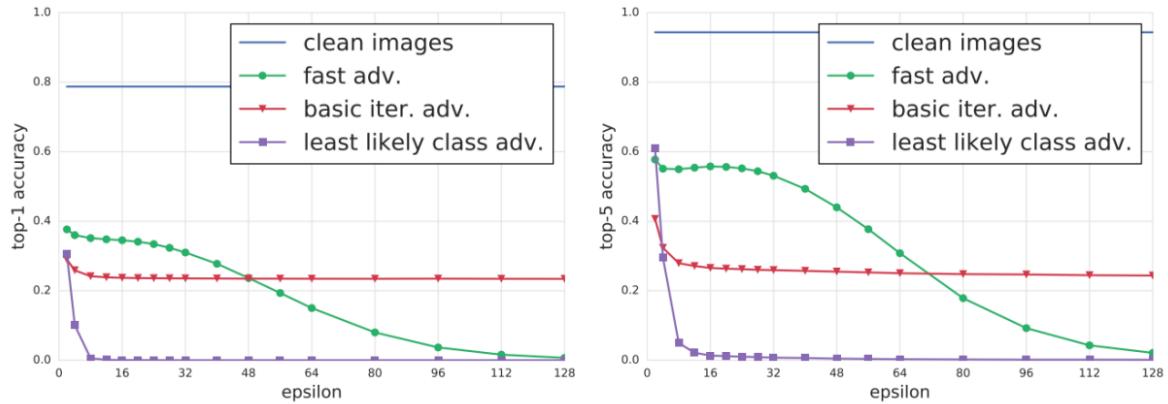


"Basic iter."; L_∞ distance to clean image = 32



"L.l. class"; L_∞ distance to clean image = 28

分类精度与 ϵ 变化情况



照片和数字图片的对抗样本攻击效果

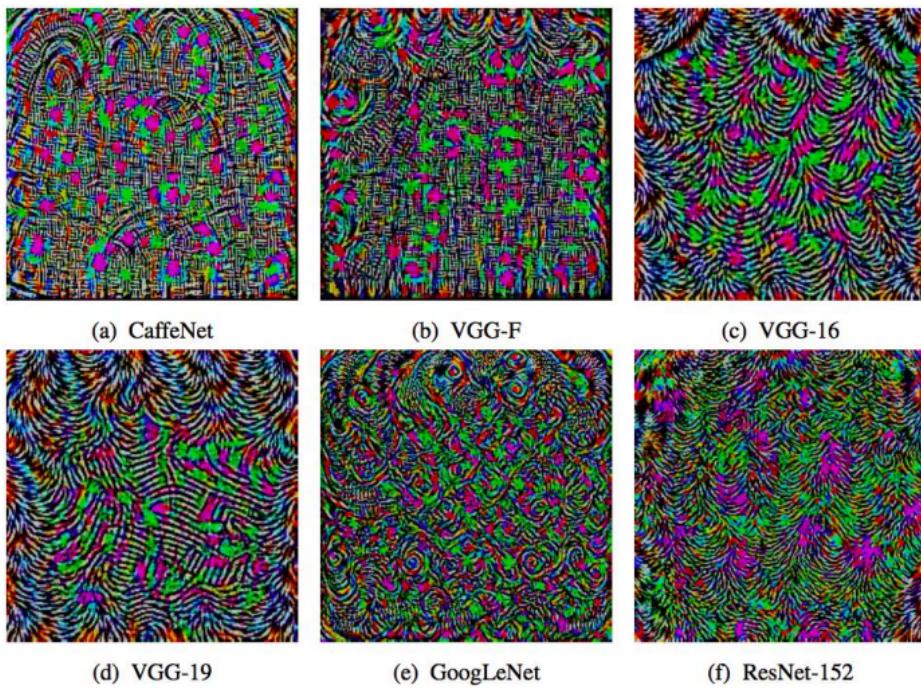
Adversarial method	Photos				Source images			
	Clean images		Adv. images		Clean images		Adv. images	
	top-1	top-5	top-1	top-5	top-1	top-5	top-1	top-5
fast $\epsilon = 16$	81.8%	97.0%	5.1%	39.4%	100.0%	100.0%	0.0%	0.0%
fast $\epsilon = 8$	77.1%	95.8%	14.6%	70.8%	100.0%	100.0%	0.0%	0.0%
fast $\epsilon = 4$	81.4%	100.0%	32.4%	91.2%	100.0%	100.0%	0.0%	0.0%
fast $\epsilon = 2$	88.9%	99.0%	49.5%	91.9%	100.0%	100.0%	0.0%	0.0%
iter. basic $\epsilon = 16$	93.3%	97.8%	60.0%	87.8%	100.0%	100.0%	0.0%	0.0%
iter. basic $\epsilon = 8$	89.2%	98.0%	64.7%	91.2%	100.0%	100.0%	0.0%	0.0%
iter. basic $\epsilon = 4$	92.2%	97.1%	77.5%	94.1%	100.0%	100.0%	0.0%	0.0%
iter. basic $\epsilon = 2$	93.9%	97.0%	80.8%	97.0%	100.0%	100.0%	0.0%	1.0%
l.l. class $\epsilon = 16$	95.8%	100.0%	87.5%	97.9%	100.0%	100.0%	0.0%	0.0%
l.l. class $\epsilon = 8$	96.0%	100.0%	88.9%	97.0%	100.0%	100.0%	0.0%	0.0%
l.l. class $\epsilon = 4$	93.9%	100.0%	91.9%	98.0%	100.0%	100.0%	0.0%	0.0%
l.l. class $\epsilon = 2$	92.2%	99.0%	93.1%	98.0%	100.0%	100.0%	0.0%	0.0%

普遍的对抗性的扰动 Universal adversarial perturbations

目标：对于给定的网络，找到一个与图像无关的扰动向量，使所有图像都有高概率被误分类

- 从 $r = 0$ 开始
- 循环使用训练样本 x_i (在多个类中)
 - 如果 $x_i + r$ 被错误的分类的话，跳到 x_{i+1}
 - 找到最小扰动 Δr 让 $x_i + r + \Delta r$ 被分类到另外一个类
 - 更新 $r \leftarrow r + \Delta r$, 确保 $\|r\| \leq \epsilon$
- 当训练样本的被欺骗率达到目标值时终止

从不同网络结构计算的扰动向量



普遍的对抗性的扰动可以很好地推广到各个模型

下表为在计算一个模型（行）的扰动并在其他模型（列）上测试时，被欺骗率

	VGG-F	CaffeNet	GoogLeNet	VGG-16	VGG-19	ResNet-152
VGG-F	93.7%	71.8%	48.4%	42.1%	42.1%	47.4 %
CaffeNet	74.0%	93.3%	47.7%	39.9%	39.9%	48.0%
GoogLeNet	46.2%	43.8%	78.9%	39.2%	39.8%	45.5%
VGG-16	63.4%	55.8%	56.5%	78.3%	73.1%	63.4%
VGG-19	64.0%	57.2%	53.6%	73.5%	77.8%	58.0%
ResNet-152	46.3%	46.3%	50.5%	47.0%	45.5%	84.0%

黑盒对抗样本

- 假设对抗者只能**用选定的输入查询目标网络并观察输出**，不能自己灌入输入
- 关键思想：利用合成输入数据学习替代目标网络，利用**替代**网络制作对抗实例
- 成功攻击第三方 api MetaMind，亚马逊和谷歌，但只能在低分辨率的数字和街道标志图像上使用

3. 对抗样本的特点

- 对于任何输入图像，通常很容易产生一个非常相似的图像，被同一网络错误分类
- 为了获得一个对抗样本，我们**不需要进行精确的梯度上升**
- 对抗图像（有时）可以在打印和拍照等转换过程中幸存下来
- 用相同的扰动**可以攻击许多图像
- 能够欺骗一个网络的对抗样本很有可能欺骗具有不同参数甚至结构的网络

4. 为什么神经网络很容易被欺骗

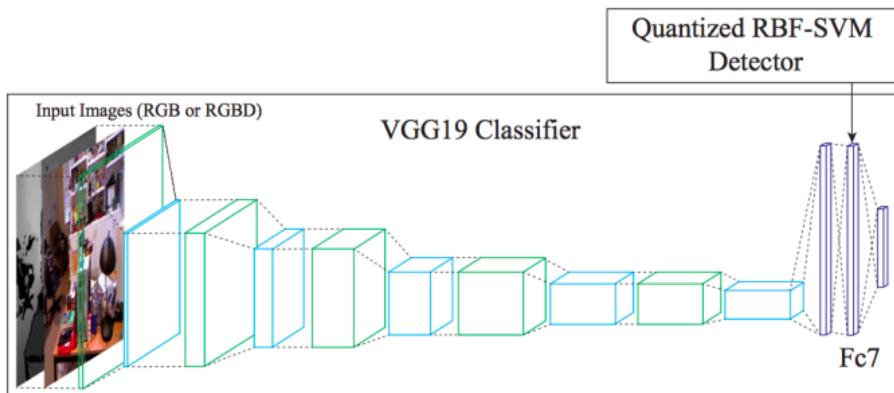
- 网络“过于线性化”：在给定输入的情况下，很容易以一种可预测的方式操纵输出
- 输入**维度很高**，因此可以通过少量改变单个输入来获得输出的巨大变化
- 神经网络可以适应任何东西，但没有什么能阻止它们在训练样本之间不规则地运行

- 与直觉相反的是，网络既可以很好地概括自然图像，也容易受到敌对例子的影响
- 对抗性的例子可以很好地概括，因为当训练不同的模型执行相同的任务时，它们可以学习相似的功能

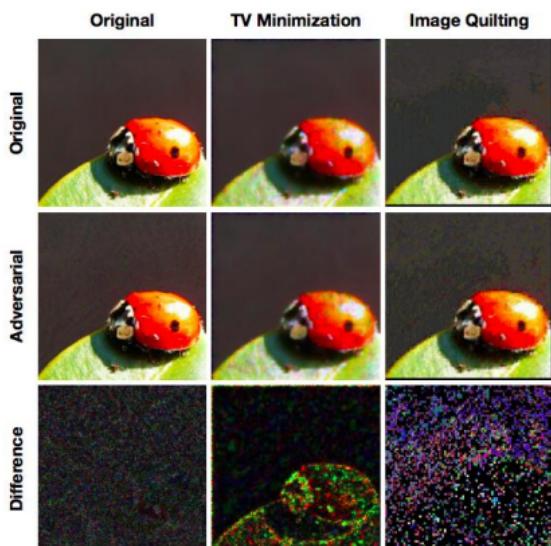
5. 防御对抗样本

对抗性训练：可以通过使用对抗性例子来增强或规范训练，使网络具有一定的抵抗力

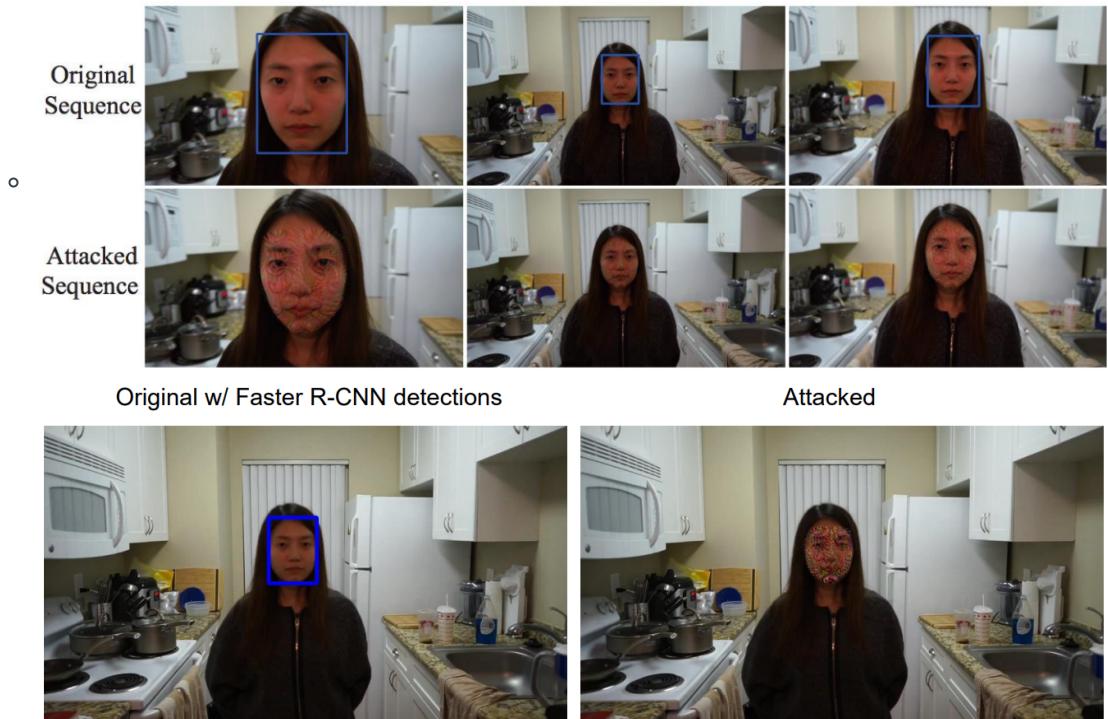
- 训练一个单独的模型来**拒绝对抗样本**：SafetyNet



- 设计高度非线性结构，对对抗扰动具有鲁棒性
- 对输入图像进行**预处理**，以破坏对抗扰动



- 要骗过 Faster R-CNN 或 YOLO 这样的探测器要比骗过分类器难得多
 - 目前需要大的扰动



- 用实物欺骗探测器就更难了



- 设计用于跨多个架构传输的对抗性示例也可能在快速演示设置中混淆人类的视觉系统

