计算机科学与技术学院神经网络与深度学习课程实验报告

实验题目：Network Visualization		学号：201900130143
日期：11/18	班级： 智能班	姓名：吴家麒
Email：wjq_777@126.com		

实验目的：

In this assignment, you will also explore methods for visualizing the features of a pretrained model on ImageNet.

• Explore various applications of image gradients, including saliency maps, fooling images, class visualizations

实验软件和硬件环境：

Anaconda3 + Jupyter notebook

实验原理和方法：

本次实验将对三种不同的深度学习可视化方法进行实现。

通过可视化深度学习的过程，我们就可以对深度学习的结果是如何产生的有一个清晰的认识，并且我们可以通过可视化的结果将模型向更优化的方向改进。

Saliency Map 是一种快速的方法来判断图像的哪一部分影响了网络的分类决策。

Saliency Maps 告诉我们图像中的每个像素对该图像分类评分的影响程度。为了计算它，我们计算对应于正确类的非归一化分数（标量）对于图像中每个像素的梯度。如果图像形状为(3,H,W) (3,H,W)(3,H,W)，那么这个梯度的尺寸也是(3,H,W)(3,H,W)(3,H,W)。这个梯度告诉我们：图像中的一个像素的微小变化将使分类评分发生多大的变化。为了计算显著性图，我们取梯度的绝对值，然后取 3 个输入通道上的最大值;最终的 Saliency Maps 因此具有形状(H , W) (H,W)(H,W)，并且是非负的。

我们也可以使用图像梯度来产生“fooling image”。

利用一张原有的图片，给定一个目标分类，通过对该图片在目标分类上的得分进行梯度上升，或者对该图片在目标分类上的 loss 进行梯度下降，来修改原有图片，使其在目标分类上的得分最高（loss 最低）。原有图片和修改后的图片人眼一般看不出差别。

Class visualization 这种方法与和 Fooling Images 有点类似，不过把开始的图片换成了一个随机的噪声图片。给定一个目标分类，通过对该图片在目标分类上的得分进行梯度上升，或者对该图片在目标分类上的 loss 进行梯度下降，来修改原有图片，使其在目标分类上的得分最高。

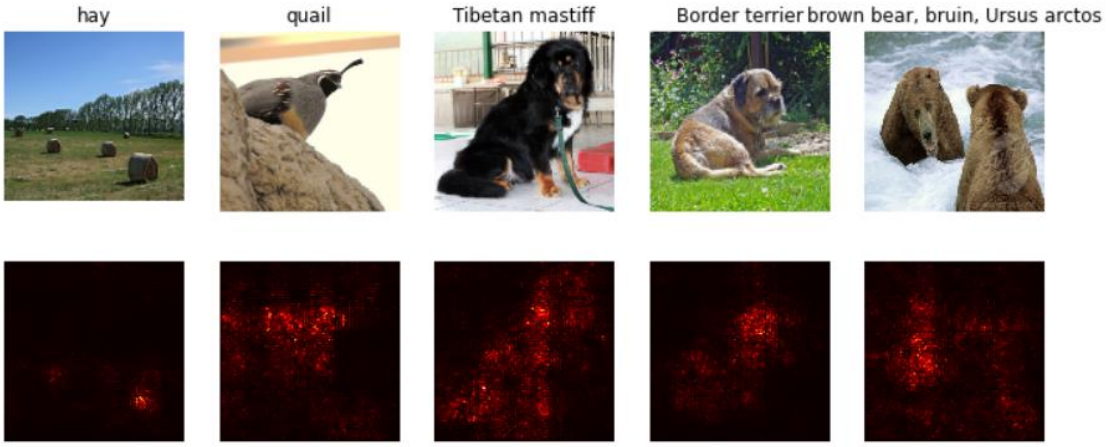
实验步骤：（不要求罗列完整源代码）

1、Saliency map

使用 gather 方法取出每个正确分类的得分，然后计算正确类的非归一化分（标量）对于图像中每个像素的梯度。然后取梯度的绝对值，用三个通道上的最大值表示 Saliency map：

```
scores=model(X)
scores=scores.gather(1,y.view(-1,1)).squeeze() #取出每个正确分类的得分
scores.backward(torch.FloatTensor([1.0,1.0,1.0,1.0,1.0])) #正确分类得分对于图像中像素的梯度
saliency=X.grad.data
saliency=saliency.abs() #取梯度的绝对值
saliency,i=torch.max(saliency,dim=1) #三通道最大值
saliency=saliency.squeeze()
```

计算结果可视化：



2、Fooling image

对该图片在目标分类上的得分进行梯度上升，当网络将图像分类为目标类时停止。在计算更新步骤时，首先标准化梯度：dX = learning_rate * g / ||g||_2

```
for i in range(100):
    scores = model(X_fooling)
    _, index = scores.max(dim=1)
    if index == target_y:
        break
    target_score = scores[0, target_y]
    target_score.backward()

    im_grad = X_fooling.grad
    X_fooling.data += learning_rate * (im_grad / im_grad.norm())
    X_fooling.grad.zero_()
```

生成结果：

可以看到，我们将原有图片向目标类修改后，看上去没什么差异很小，但将 difference 放大十倍，可以看到是有区别的。这在一定程度上反映了模型将图片判定为 stingray 这个类时关注的一些点。

3. Class visualization

这种方法与和 Fooling Images 有点类似，不过把开始的图片换成了一个随机的噪声图片。给定一个目标分类，通过对该图片在目标分类上的得分进行梯度上升，或者对该图片在目标分类上的 loss 进行梯度下降，来修改原有图片，使其在目标分类上的得分最高（loss 最低）

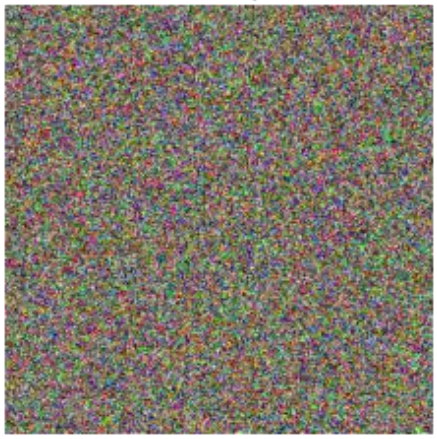
$$I^* = \arg\max_I (s_y(I) - R(I))$$

其中，I 是图像，y 是目标类别，$s_y(I)$是神经网络判断图像 I 在目标类别 y 中的得分，$R(I)$是正则项。我们的目标是让图像在 y 类中的得分最大。

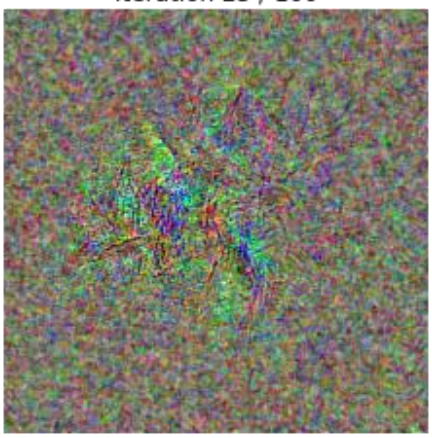
```
model.eval()
score = model(img)
sy = score[:,target_y]
model.zero_grad()
sy.backward()
dimg = img.grad + 2 * l2_reg * img
with torch.no_grad():
    img += learning_rate * dimg / torch.norm(dimg)
```

梯度上升迭代结果：

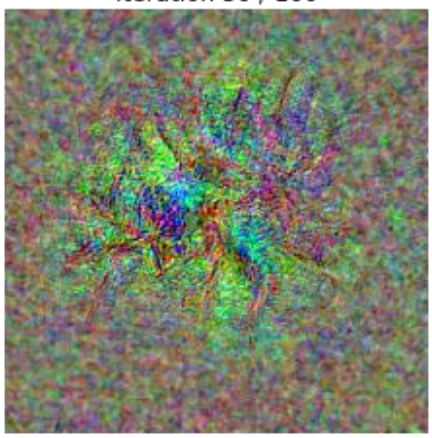
tarantula
Iteration 1 / 100



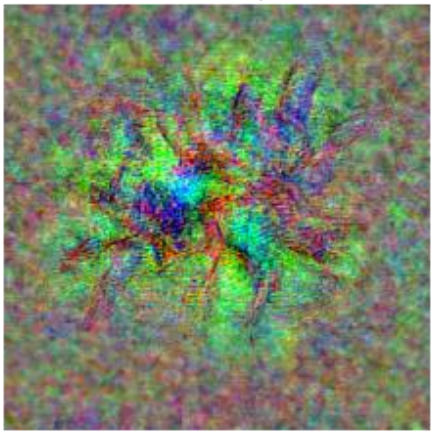
tarantula
Iteration 25 / 100



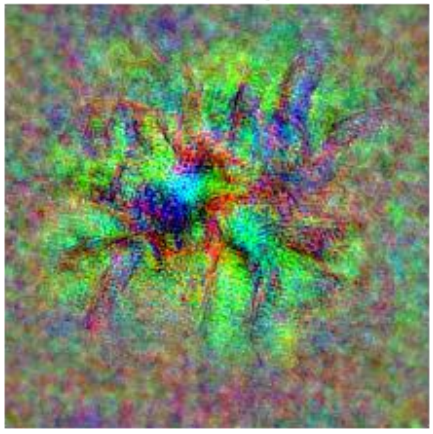
tarantula
Iteration 50 / 100



tarantula
Iteration 75 / 100



tarantula
Iteration 100 / 100



结论分析与体会：

1、从本次实验的结果中可以发现，Saliency Map 可以用来快速的检测图像中的哪一部分对神经网络的决策产生了较大的影响；Fooling image 和 class visualizasion 都是通过将原始图片向目标类别转换来分析出某一特定类别图片中对神经网络决策有较大影响的部分。

2、通过本次实验对神经网络的可视化方法进行了学习和运用，对神经网络的实际训练过程有了更深入的理解。

3、对于神经网络在实际应用中的过程有了更为深入的理解和学习。