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 (PyTorch)		学号：201900130151
日期：2021.11.22	班级： 人工智能	姓名： 莫甫龙
Email：m1533979510@163.com		

实验目的：

补全代码，实现 Saliency Maps、Fooling Images、Class Visualization

实验软件和硬件环境：

Vs code

Win11

实验原理和方法：

Saliency Maps：

Saliency Maps 主要是用来表示每个像素对图像分类得分的影响程度。这里是通过反向传播，来计算每个图片像素的梯度的绝对值，然后在三个通道中选最大值。

Fooling Images：

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，使得 loss 最低，从而使得分最高。

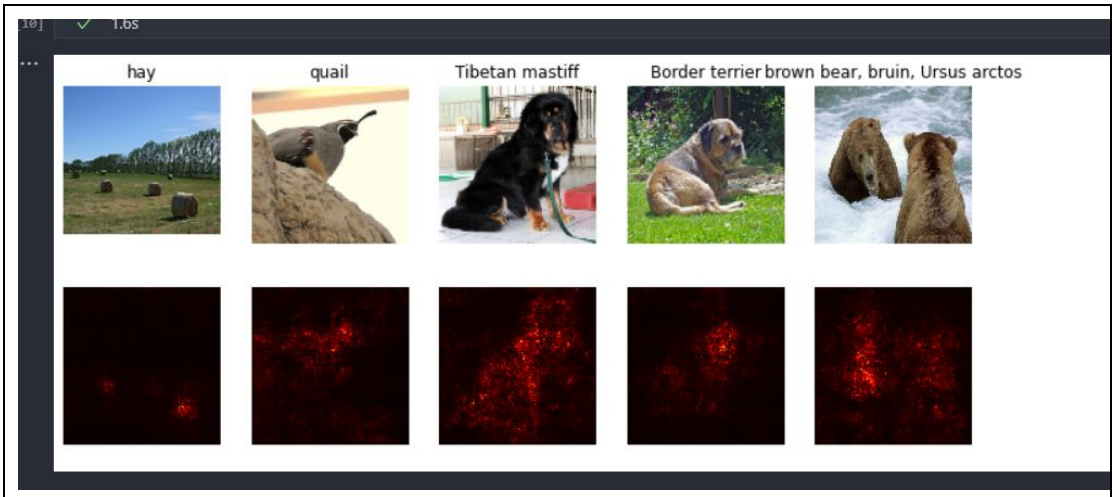
Class Visualization：

和 Fooling Images 类似。

实验步骤：（不要求罗列完整源代码）

Saliency Maps：

```
##############################################################################
# *****START OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
model.zero_grad()
loss_fun = torch.nn.CrossEntropyLoss()
y_pred = model(X)
loss = loss_fun(y_pred, y)
loss.backward()
saliency,_ = X.grad.abs().max(axis=1)
pass

# *****END OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
##############################################################################
#                               END OF YOUR CODE                             #
```



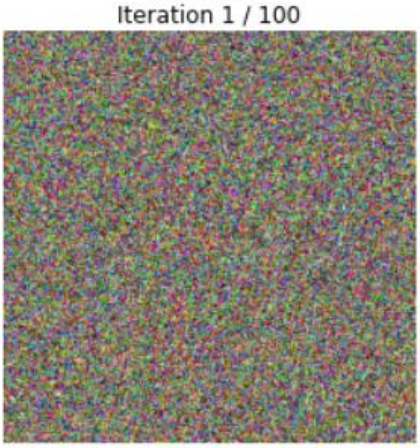
Fooling Images:

```
# *****START OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
model.eval()
iters =100
y = torch.LongTensor([target_y])
loss_fun = torch.nn.CrossEntropyLoss()
for i in range(iters):
  score = model(X_fooling)
  if score.argmax(axis=1) == y:
    break
  loss = loss_fun(score, y)
  model.zero_grad()
  loss.backward()
  with torch.no_grad():
    g = X_fooling.grad
    dX = learning_rate * g / torch.norm(g)
    X_fooling -= dX
pass

# *****END OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
```

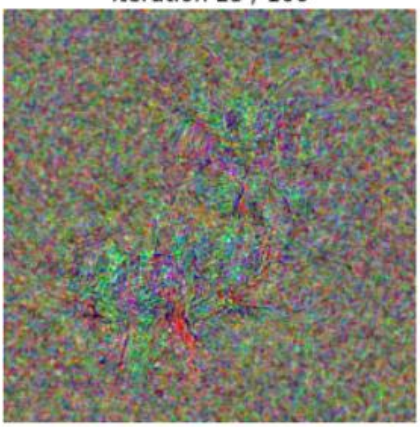
Class Visualization:

```python
for t in range(num_iterations):
    # Randomly jitter the image a bit; this gives slightly nicer results
    ox, oy = random.randint(0, max_jitter), random.randint(0, max_jitter)
    img.data.copy_(jitter(img.data, ox, oy))

    ########################################################################
    # TODO: Use the model to compute the gradient of the score for the     #
    # class target_y with respect to the pixels of the image, and make a   #
    # gradient step on the image using the learning rate. Don't forget the #
    # L2 regularization term!                                              #
    # Be very careful about the signs of elements in your code.            #
    ########################################################################
    # *****START OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
    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)

    pass

    # *****END OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
```

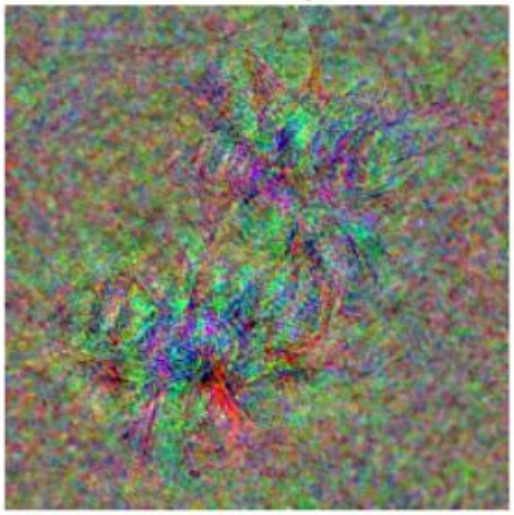
tarantula
Iteration 1 / 100



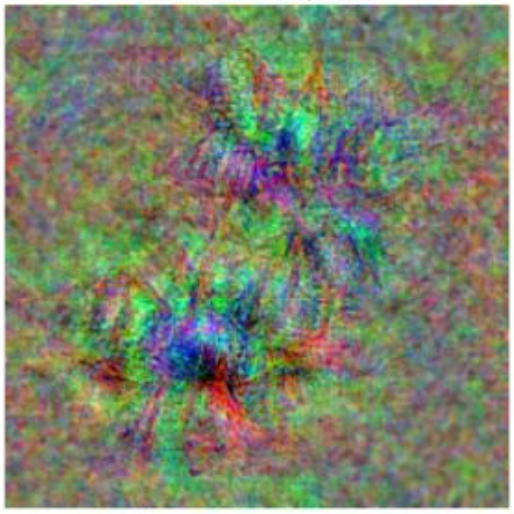
tarantula
Iteration 25 / 100

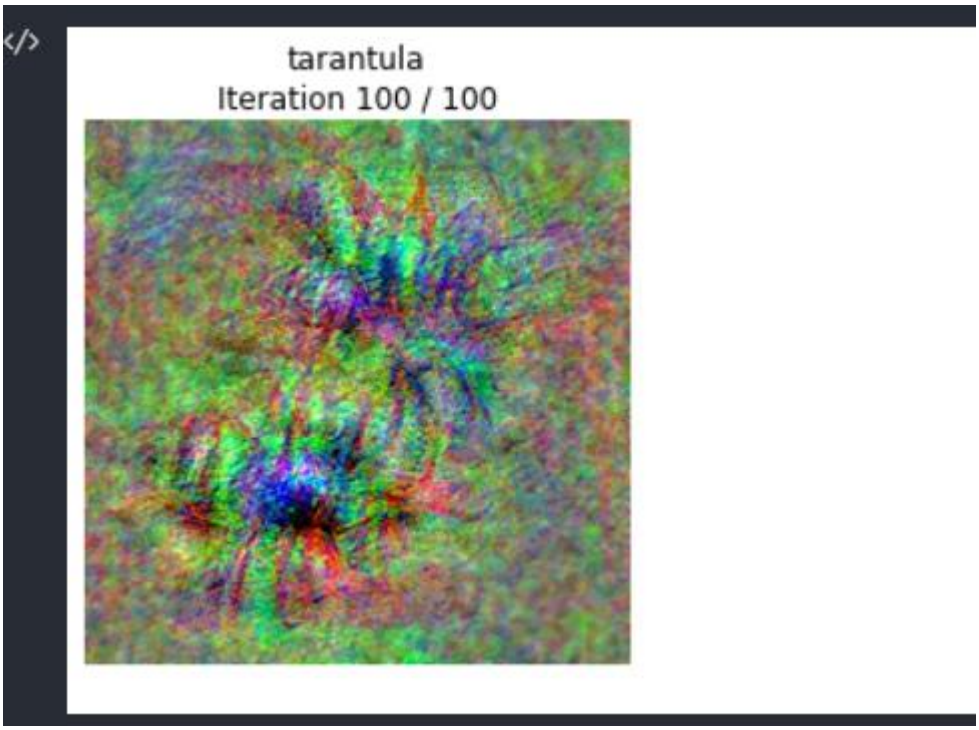


tarantula
Iteration 50 / 100



tarantula
Iteration 75 / 100





结论分析与体会：

基本了解了 Saliency Maps、Fooling Images 和 Class Visualization 三种图像生成技术。

就实验过程中遇到和出现的问题，你是如何解决和处理的，自拟 1－3 道问答题：

这个实验遇到的问题主要是配环境的问题，配了很多次都会报错，说核死亡或者命令没有响应，后面通过安装同学发的环境导出的 txt 文档才解决的问题。