Machine Learning HW5 Report

學號：R07943107　系級：電子所碩一　姓名：徐晨皓

1. **(1%) 試說明 hw5_best.sh 攻擊的方法，包括使用的 proxy model、方法、參數等。此方法和 FGSM 的差異為何？如何影響你的結果？請完整討論。（依內容完整度給分）**

A. 攻擊方法

(1) Proxy model：ResNet-50

(2) 方法：

首先，hw5_best.sh 使用 PyTorch pretrained model ResNet-50。在 preprocessing 部分，使用線上 torchvision.models 說明文件中的方法。即先將 data normalize 至[0,1]，再用$mean = [0.485, 0.456, 0.406]$與$std = [0.229, 0.224, 0.225]$ normalize 一次。

hw5_best.sh 的攻擊方法是基於 FGSM 的方法再做改良。我們可以發現在 FSGM 中 epsilon 的大小與 success rate 有很大的關係，在助教提供的 sample code 裡只有對 image 加一次 noise，這樣並不能保證產生出來的圖片能讓 model 錯誤辨識。因此我採用 iterative 的方法來加 noise，若加一次 noise 無法成功就會進入下一個 iteration，直到 model 錯誤辨識或達到最大 iteration 數才停止。為避免ε的大小讓圖片振盪無法收斂，ε的值會隨著 iteration 數量增加而減小，本次作業設定為$\varepsilon_i = \frac{\varepsilon_0}{i}$，其中$\varepsilon_i$為第 i 次 iteration 的ε，$\varepsilon_0$為初始ε值。

(3) 參數：

(a) ε_0: 0.06

(b) Iterations: 50

B. 與 FGSM 的差異

主要的差異為，當當前的 image 無法讓 model 錯誤辨識時，hw5_best.sh 會對 image 加數次 noise 直到 model 錯誤辨識。這樣比起 simple FGSM 能有更多機會產生出成功錯誤辨識的圖片。

C. 如何影響結果

從實驗結果中，使用 simple FGSM 僅有 92.5%的 success rate，L-inf. norm 為 6.0。但使用 iterative FGSM 能有 100%的 success rate，L-inf. norm 為 4.17。因此我認為 iterative 的方法能更有效的產生出有效攻擊 model 的圖片。

2. **(1%) 請列出 hw5_fgsm.sh 和 hw5_best.sh 的結果（使用的 proxy model、success rate、L-inf. norm）。**

A. hw5_fgsm.sh

(1) Proxy model: ResNet-50

(2) Success rate: 0.925

(3) L-inf. norm: 6.0000

B. hw5_best.sh

(1) Proxy model: ResNet-50

(2) Success rate: 1.000

(3) L-inf. norm: 4.1700

3. **(1%) 請嘗試不同的 proxy model，依照你的實作的結果來看，背後的 black box 最有可能為哪一個模型？請說明你的觀察和理由。**

A. 最有可能的 black box：**ResNet-50**

B. 觀察與理由：

(1) 從助教所提示的六種 pretrained models 辨識 200 張 images 的成功率，我們就可以猜到 black box 為何。下表為六種 pretrained models 對於 200 images 的辨識成功率 (使用 torchvision.models 說明網頁中的 preprocess 方法)。ResNet-50 有 100%的正確率，因此在實作 FGSM 前基本上就已確認 black box 為 ResNet-50。

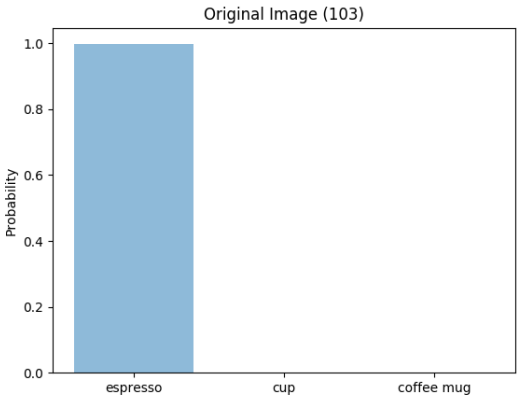
Model	Classification Success Rate
VGG-16	173/200 (86.5%)
VGG-19	174/200 (87.0%)
ResNet-50	200/200 (100.0%)
ResNet-101	186/200 (93.0%)
Densenet-121	185/200 (92.5%)
Densenet-169	183/200 (91.5%)

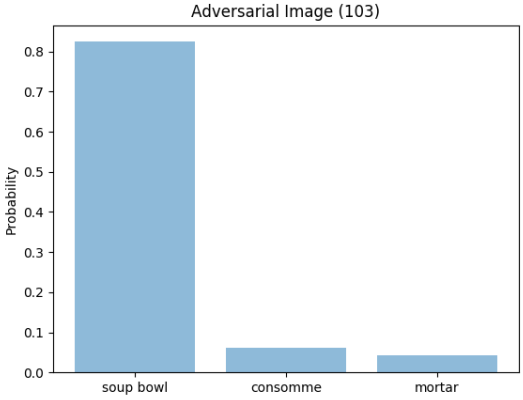
(2) 從實作的 FGSM 中，當我們使用不同的 proxy model 時，攻擊成功率有顯著的差別。下表為六種 pretrained model 對於 FGSM（使用 hw5_fgsm.sh）產生出的攻擊圖片的辨識失敗率。從表中，ResNet-50 有最高的辨識失敗率 92.5%，其餘五種 model 的辨識失敗率皆低於 50%，因此 ResNet-50 最有可能是 black box。

Model	Classification Failure Rate
VGG-16	59/200 (29.5%)
VGG-19	55/200 (27.5%)
ResNet-50	185/200 (92.5%)
ResNet-101	97/200 (48.5%)
Densenet-121	78/200 (39.0%)
Densenet-169	77/200 (38.5%)

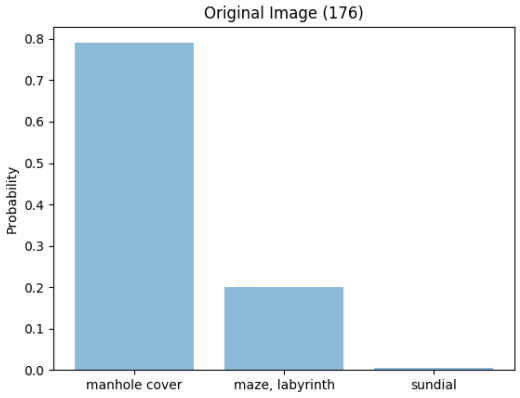
4. **(1%) 請以 hw5_best.sh 的方法，visualize 任意三張圖片攻擊前後的機率圖（分別取前三高的機率）。**

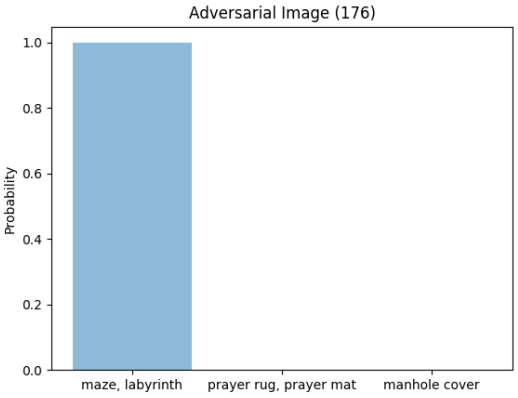
103.png



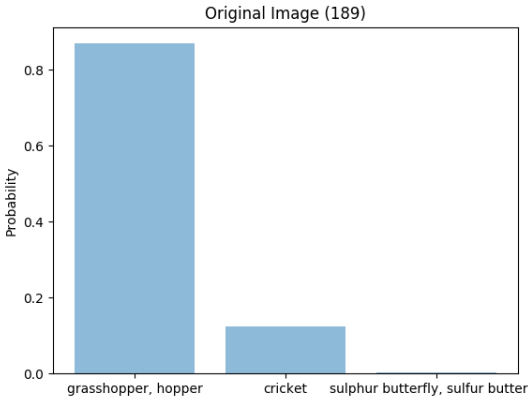


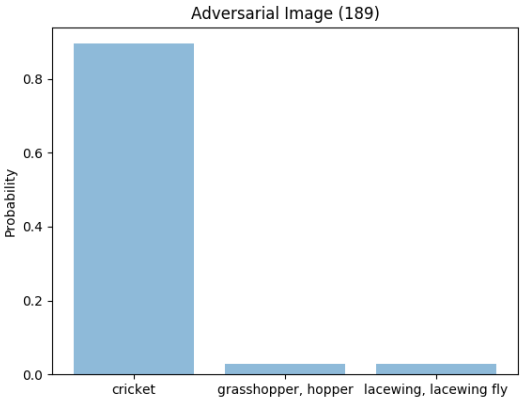
176.png





189.png





5. **(1%) 請將你產生出來的 adversarial image，以任一種 smoothing 的方式實作被動防禦 (passive defense)，觀察是否有效降低模型的誤判的比例。請說明你的方法，附上你防禦前後的 success rate，並簡要說明你的觀察。另外也請討論此防禦對原始圖片會有什麼影響。**

本題選用 Gaussian filter 作為 smoothing 的方式。使用 scipy.ndimage 中的 gaussin_filter 實作，sigma 設為 1。

A. 在對由 hw5_best.sh 產生出的 adversarial images 使用 Gaussian smoothing 後，實驗結果如下:

	Before smoothing	After smoothing
Success rate	100%	64%
L-inf. norm	4.17	116.05

原先成功率為 100%的 adversarial images，在使用 Gaussian smoothing 後，成功率降為 64%。因此我認為以 smoothing 的方法並無法有效的降低 model 辨識成功率。

B. 選用 Gaussian filter 作為 smoothing 的方式。在對原始圖片使用 Gaussian smoothing 後，實驗結果如下:

Success rate	27.5%
L-inf. norm	116.13

原始圖片在使用 Gaussian smoothing 後，success rate 僅為 27.5%。因此我認為將圖片模糊化或平滑化並無法很有效的成功攻擊一個 model。要攻擊一個 model 較好的方法還是了解其結構，對其結構的弱點攻擊(如 FGSM)，這樣才能有效的攻擊 model。