國立臺灣大學電機資訊學院電信工程學研究所

碩士論文

Graduate Institute of Communication Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

探求與分析對抗性擾動中隱藏的人類可識別資訊

Discovering and Analyzing Human-Recognizable Information Hidden in Adversarial Perturbations

門玉仁

Dennis Y. Menn

指導教授: 李琳山 教授

Advisor: Lin-shan Lee, Ph.D.

中華民國 112 年 6 月

June, 2023

doi:10.6342/NTU202301498

致謝

功讀碩士學位兩年的時間，對於我而言是場困難的挑戰。讀書與研究有著明顯的不同，研究是一段非結構性的學習，這些知識往往四散於各篇論文中，其中重點在於沉澱自己的想法，並設計實驗驗證假設，這與大學的課程很不一樣。此外，讀碩士的期間受疫情影響，導致許多活動都被迫中止。在這個艱困的時間，很謝謝我所遇到的貴人給予我的幫助，讓我得以順利的渡過這一場考驗。

跟隨琳山教授四年半的日子中，教授的研究與處事態度給我很大的啟發。研究上教授非常注重思考並帶給我一個重要的觀念，亦即直觀的了解道理才算真正意義上的理解。當我們可以直觀的看待問題時，才能從別人遺漏的線索中，找到自己的方向。同時，教授很注重我們的表達，老師逐字更改我們的論文，也逐頁的修改口試投影片，在修改的過程中，我才體會到好的報告是讓別人可以簡單的聽懂報告內容。

教授在處事的態度上給與我很大的啟發。老師告訴我們要有闖盪的膽量，只有當我真正面臨一些選擇的時候，我才開始理解這些話的意義。老師的話在我心中種下一顆種子，未來我碰到困難時，想起這些話，種子就會開始發芽。

對我而言，琳山教授像一個指北針，無論地貌如何複雜，老師總是可以告訴我們何去何從。謝謝老師四年半來對我的指導與包容。

我很感謝宏毅教授，宏毅教授是一把火炬，在寒冷而漆黑的研究路上，給與

doi:10.6342/NTU202301498

我們研究方法而照亮著我們的路途，同時，老師非常的溫暖，即便遭遇冰冷的挫折，老師永遠會給與我們溫暖與希望。無論什麼時候，宏毅教授總是盡可能的幫助我們，非常謝謝教授。

研究的過程永遠要有同袍與我並肩作戰，非常感謝子軒與恆成哥，他們總是樂意教導我如何寫程式、協助我處理棘手的問題，並與我討論後指出我思考的錯誤。我也很感謝永遠充滿笑容的宣叡與文靜優雅的孟妍，與他們討論問題和聊天總是可以得到很多收穫。很感謝彤恩姊對我們的照顧，彤恩姊總是熱情的幫助我們解決問題。也非常謝謝其他實驗室的夥伴們給與我許多幫助。

在困境中，我的父母與姐姐永遠支持我、幫助我，無論面臨什麼困難，他們永遠站在我旁邊，我很幸運能有這樣的家人。

讀碩士的期間，我與阿姨、外公、外婆同住。外公、外婆年事已高，非常謝謝阿姨不辭辛勞的照顧外公、外婆與幫助我，即便阿姨工作非常忙碌，也永遠會騰出時間陪伴外公、外婆。也謝謝外公將他畢生累積的經驗，注入到對於我的提醒中，對我而言，外公如太陽一般耀眼。謝謝最疼我的外婆，外婆永遠最掛念我!

我知道，是眾多人給與我的幫助，讓我能夠順利的渡過這些難關，未來有能力時，我也同樣會回饋這個社會。

doi:10.6342/NTU202301498

摘要

在眾多機器學習領域中，類神經網路已展現出其頂尖的性能。然而，研究者也發現在輸入資料中添加微小的擾動，亦即對抗性擾動，就有機會混淆類神經網路的判斷。此一現象意味著，在應用類神經網路至如自動駕駛或語者識別等真實世界的任務時，其安全性和可靠性仍然會因為對抗性擾動的存在而面臨威脅。然而，時至今日，仍沒有實用的演算法能夠有效的防範對抗性擾動，其中一個原因是人們並不理解對抗性擾動的作用機制。

本論文提出對抗性擾動含有人類可以識別的資訊，並進一步以實驗證明這是導致類神經網路預測錯誤的一個重要因素。這與過去廣為人知的觀點，認為類神經網路的判斷失誤源於人類無法解讀的資訊，相當不同。

本論文的研究還發現了兩種存在於擾動中的效應，即遮掩效應和生成效應，這些效應可以被人類解讀，也會導致模型辨識錯誤。而且，這兩種效應在不同攻擊演算法和資料集中都可以被觀察到。

這些發現有機會幫助研究者能夠更深入的解析對抗性擾動的特性，包括其作用機制、可轉移性，以及對抗性訓練如何增強模型的可解釋性，使得吾人得以更深入的了解類神經網路的運作原理，並促進防禦演算法的研發。

關鍵字：機器學習、類神經網路、資訊安全、對抗性擾動、圖形識別

doi:10.6342/NTU202301498

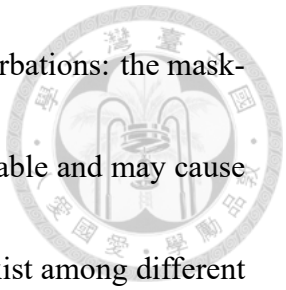
doi:10.6342/NTU202301498

Abstract

Neural networks have achieved state-of-the-art performance in many machine learning tasks. However, researchers have found that adding small perturbations to input data, known as adversarial perturbations, can cause neural networks to make incorrect predictions. This phenomenon signifies that when applying neural networks to real-world applications such as autonomous driving or speaker verification, their safety and reliability still face threats due to adversarial perturbations. However, nowadays, there is still a lack of practical algorithms that can effectively defend against adversarial attacks. One of the reasons for this is that researchers are still unclear about the underlying mechanisms of adversarial perturbations.

This paper proposes that adversarial perturbations contain human-recognizable information. Our experiments show that this information is an essential factor leading to prediction errors in neural networks. This finding opposes the widely held belief that adversarial perturbations are unrecognizable to humans.

doi:10.6342/NTU202301498



This paper also discovers two effects present in adversarial perturbations: the masking effect and the generation effect. Both effects are human-recognizable and may cause neural networks to make mistakes. More importantly, these effects exist among different attack algorithms and datasets.

Our findings may help researchers gain a deeper understanding of the nature of adversarial perturbations, including their working mechanism, transferability, and how adversarial training enhances the interpretability of models, etc., leading to a deeper understanding of neural networks and the development of more effective defensive algorithms.

Keywords: Machine Learning, Neural Network, Information Security, Adversarial Perturbations, Pattern Recognition

doi:10.6342/NTU202301498

目錄

頁碼

致謝 **i**

摘要 **iii**

Abstract **v**

目錄 **vii**

圖目錄 **xi**

表目錄 **xiii**

第一章 導論 **1**

1.1 研究動機 1

1.2 研究方向 2

1.3 主要貢獻 3

1.4 章節安排 3

第二章 背景知識 **5**

2.1 對抗性擾動 5

2.2 攻擊演算法 6

2.2.1 白盒攻擊 7

2.2.2 黑盒攻擊 9

2.3 防禦演算法 10

doi:10.6342/NTU202301498

2.4 擾動的特性及現象 14
2.5 本章總結 17
第三章 相關研究 19
3.1 類神經網路的局部線性 19
3.2 對抗性實例偏離資料分佈 21
3.3 擾動改動非強健性特徵 22
3.4 數學證明擾動的存在 23
3.5 本章總結 23
第四章 探求擾動的本質 25
4.1 本論文所提出之假設 25
4.2 本論文提出假設之相關研究 26
4.3 本論文所提之假設面對的挑戰 27
4.3.1 擾動的可識別性 27
4.3.2 通用對抗性擾動的存在 28
4.3.3 問題的複雜性 29
4.3.4 對於類神經網路的不理解 29
4.4 解析可識別資訊 29
4.5 實驗方法 30
4.6 實驗設定 31
4.6.1 輸入資料 31
4.6.2 產生擾動 32
4.6.2.1 單模型設定 32
4.6.2.2 雜訊多模型設定 32

doi:10.6342/NTU202301498

4.6.3 模型架構 32
4.6.3.1 MNIST 的模型架構 33
4.6.3.2 CIFAR10 的模型架構 33
4.6.3.3 ImageNet 的模型架構 33
4.6.4 攻擊參數 34
4.6.4.1 基礎迭代攻擊法 34
4.6.4.2 CW 氏攻擊 35
4.6.4.3 深層懸弄攻擊 35
4.6.5 高斯雜訊 35
4.6.6 呈現擾動 36
4.7 改良演算法 37
4.7.1 裁剪效應 37
4.7.2 校正輸出值 37
4.7.3 加速演算法 38
4.8 實驗結果 39
4.8.1 無特定目標攻擊 39
4.8.2 擾動的可識別性 41
4.8.2.1 機器評估 41
4.8.2.2 人類評估 42
4.8.3 評估攻擊強度 43
4.8.4 遮掩效應的影響 44
4.8.5 有特定目標攻擊 53
4.9 本章總結 54
第五章 實驗觀察 **57**
5.1 搜尋式攻擊 57

doi:10.6342/NTU202301498

5.2 雜訊的影響 59
5.2.1 移除雜訊 60
5.2.2 探討標準差的影響 61
5.2.3 格狀條紋的顯現 61
5.2.4 細節資訊的消失 62
5.3 擾動的特性 62
5.3.1 互補性 62
5.3.2 比較生成的擾動 63
5.4 消失的貓 65
5.4.1 本章總結 68
第六章 綜合討論 **69**
6.1 類神經網路的脆弱性 69
6.2 擾動的可轉移性 69
6.3 模型的可解釋性 70
6.4 非穩健性特徵的作用 71
6.5 本章總結 71
第七章 結論與展望 **73**
7.1 研究總結 73
7.2 未來展望 74
參考文獻 **75**

doi:10.6342/NTU202301498

圖目錄

2.1 展示對抗性擾動對於類神經網路的影響，圖中左、中、右分別是熊貓的圖片、對抗性擾動與對抗性實例。 6

2.2 防禦性蒸餾的訓練步驟。 11

2.3 防禦性生成對抗網路的演算法示意圖。 13

2.4 對抗性訓練增進模型的可解釋性。 16

3.1 類神經網路的局部線性。 20

3.2 輸入資料與決策邊界的分佈。 22

4.1 無法辨別擾動中的可識別資訊。左、中、右分別是原始圖片、方形攻擊與深層懸弄攻擊所產生的擾動。 27

4.2 通用對抗性擾動與卷積層的權重，左圖是通用對抗性擾動，右邊兩張圖是 VGG-16 模型的權重。 28

4.3 裁剪效應，左、中、右側圖片分別是檸檬的圖片、裁剪後及未裁剪的擾動。 38

4.4 無特定目標攻擊演算法生成的擾動。 40

4.5 難以判斷類別的圖片。 43

4.6 輸入圖片與對抗性實例。 46

4.7 原始擾動與提取輪廓後的擾動。 46

4.8 擾動面積與模型辨識正確率之關係圖。 48

4.9 擾動面積微分值與模型辨識正確率之關係圖。 50

4.10 擾動 L_{inf} 範數與模型辨識正確率之關係圖。 52

4.11 擾動中的可識別資訊會導致模型判斷錯誤。 53

4.12 有特定目標攻擊所產生的對抗性實例。 55

doi:10.6342/NTU202301498

5.1 單模型設定下，方形攻擊產生的擾動與對應的圖片。 58
5.2 方形攻擊產生的擾動具備遮掩效應。 58
5.3 方形攻擊產生的擾動不具備明顯遮掩效應的例子。 59
5.4 多模型設定 (不添加雜訊) 下產生的擾動。 60
5.5 高斯雜訊的標準差對於生成擾動的影響。 62
5.6 對抗性擾動的互補性。 63
5.7 比較不同攻擊演算法產生擾動的餘弦相似度。 64
5.8 比較梯度與 CW 氏攻擊法所產生擾動的可識別性。 65
5.9 針對輸入圖片為貓的類別所產生的對抗性擾動。 65

doi:10.6342/NTU202301498

表目錄

4.1 實驗中使用的攻擊演算法及對應的參數。 35
4.2 實驗中使用的資料集與對應的雜訊參數。 36
4.3 評估模型的正確率: 表格中第 (i,j) 元素意味著，在第 i 種設定下，評估模型判斷 j 方法產生擾動的正確率。 42
4.4 無特定目標攻擊下產生的擾動攻擊強度。 45

doi:10.6342/NTU202301498

doi:10.6342/NTU202301498

第一章　導論

1.1　研究動機

類神經網路 (Neural Network) 在廣泛的機器學習任務中均有著十分出色的表現 [1]，電腦視覺 [2]、語音辨識 [3] 以及自然語言處理 [4] 領域為其中數例。由於優秀的性能以及廣泛的應用能力，類神經網路已經漸漸地融入了我們日常的生活中，常見的應用包括自動駕駛 [5] 、聲紋識別 [6]、人臉識別 [7]、醫學影像分析 [8] 等。可以想見在不久的未來，類神經網路還會更廣泛的出現在我們的生活當中。

然而，賽氏 (Szegedy) 等人發現，稍微改動輸入資料便有機會可以操縱類神經網絡的判斷結果 [9]，這種改動稱為對抗性擾動 (adversarial perturbation)。添加對抗性擾動之量值往往十分微小，甚至可以在人類無法覺察異常的情況下，使得類神經網路判斷錯誤，從而影響其可靠性。這會使得類神經網路的應用面臨許多潛藏危機。舉例而言，將對抗性擾動添加至停車標誌，可以使得自動駕駛汽車將停車標誌誤判為 45 英里的限速標誌，從而導致事故的發生 [10]；此外人們佩戴經特殊設計的眼鏡，也可以使得人臉辨識系統錯誤識別佩戴者的身份，導致系統錯誤辨識人的身分別 [11]。

由於對抗性擾動的存在使得應用類神經網路產生安全疑慮，因此，從 2013 年

doi:10.6342/NTU202301498

發現對抗性擾動的存在後 [9]，此議題就備受學界關注。然而迄今為止，學者仍未能找到一種有效的演算法來抵禦對抗性擾動的攻擊，究其根本原因是我們對於擾動作用機制與類神經網路運作原理的不了解。因此，本研究著重於探討對抗性擾動的特性與作用機制，並從中窺探類神經網路的運作原理。

1.2 研究方向

本論文首先回顧對抗性擾動的相關特性。接著，藉由這些特性，我們推斷擾動中應該隱藏一些資訊。不同於先前的研究，我們假設這些資訊應該符合人類的某種預期，亦即若是擾動使得類神經網路將雞的圖片判斷錯誤，則擾動中應含有一隻代負值的雞俾能遮掩原圖片中的雞，我們將此資訊稱為人類可識別資訊。因此，當擾動加到圖片上，其作用即為遮掩圖片的公雞，導致類神經網路判斷錯誤。

接著，我們做實驗來驗證假設。要在對抗性擾動中，觀測到人類可識別資訊具有挑戰性，因為擾動中含有大量的雜訊，導致可識別資訊被這些雜訊所淹沒。此外，擾動中的可識別資訊往往是殘缺的，亦即雖然擾動的分佈符合人類的預期，卻不容易從個別擾動中觀察到這些資訊。此外，對抗性攻擊 (Adversarial attack) 具有多樣化的演算法以及不同的攻擊設定 [9, 12–16]，使得識別擾動中包含的資訊更為困難。

此外，我們必須在不改變擾動本質的情況下，最小化雜訊的影響並重構殘缺的資訊。因此，本研究需要針對同一張圖片，使用大量的類神經網路來產生不同的擾動，並且藉由其他技巧進一步提升擾動的數目，最終平均所有產生的擾動。這是因為不同類神經網絡所生成擾動的雜訊與殘缺部分往往相互獨立，因此平均這些擾動可以有效地降低雜訊的影響，同時重構可識別資訊。

doi:10.6342/NTU202301498

我們發現使用上述的方法處理完擾動後，可識別資訊確實出現於擾動中。因此，我們進一步定量的評估擾動的可視性與攻擊能力，並且實驗在不同的資料集、攻擊設定與攻擊演算法來驗證我們的論點。

然後我們探討影響可識別資訊清晰程度的因子，以確保可識別資訊源於對抗性擾動中。同時，我們也將實驗中觀測到許多有趣現象的現象呈現在論文中。

最後，我們發現對抗性擾動具備可識別資訊，此觀點可以解釋諸多擾動具備的特性，包括可轉移性 (Transferability)、類神經網路的脆弱性 (Vulnerability)、對抗性訓練 (Adversarial Training) 增進模型的可解釋性 (Explainability)、非穩健性特徵 (Non-robust Feature) 的作用等。

1.3 主要貢獻

本論文的主要貢獻包含:

- 發現對抗性擾動本身即具備人類可識別資訊，並普遍存在於對抗性擾動中
- 檢驗可識別資訊為導致模型誤判的重要因子
- 發現兩類不同的可識別資訊
- 觀察擾動所衍生的相關現象
- 解釋擾動所具備的性質

1.4 章節安排

本論文的章節安排如下:

doi:10.6342/NTU202301498

- 第二章 背景知識
- 第三章 相關研究
- 第四章 探求擾動的本質
- 第五章 實驗觀察
- 第六章 綜合討論
- 第七章 結論與展望

doi:10.6342/NTU202301498

第二章　背景知識

2.1　對抗性擾動

對抗性擾動是一種添加於輸入資料的擾動，其目的是使類神經網路判斷錯誤或是操縱類神經網路的判斷結果。為了增加擾動所帶來的危害，其量值應盡可能小，這樣添加擾動至輸入資料時，人們才不會察覺到相異之處。這些添加擾動的輸入資料被稱為對抗性實例 (Adversarial Example)。

我們可以用數學嚴謹的定義對抗性擾動，給定類神經網路的參數為 θ，且在對抗性擾動之範數值 (Norm) 小於一量值 ϵ 的限制下，針對輸入資料 x，尋找對抗性擾動 δ，以最大化模型的輸出向量 $f(x+\delta)$ 與輸入資料的標簽 y 的損失值 $J(x+\delta, y, \theta)$。請見以下公式：

$$\underset{\delta}{\text{argmax}}\ J(x+\delta, y, \theta),\ subject\ to\ |\delta|_p < \epsilon \tag{2.1}$$

一般常用的範數度量為 L_0, L_2, L_{inf} 範數，範數值的大小限制會隨資料集的不同而改變，一般生成對抗性擾動的演算法會先根據資料集（Dataset）決定範數度量方法以及量值限制 ϵ，再經由反向傳播演算法 (Backpropagation) [17] 來迭代優化公式2.1中的擾動 δ。實際生成的對抗性實例請見圖2.1，圖取自 [12]。圖中左、中、右分別是熊貓的圖片、對抗性擾動與對抗性實例，括弧內的文字與數值則是

doi:10.6342/NTU202301498

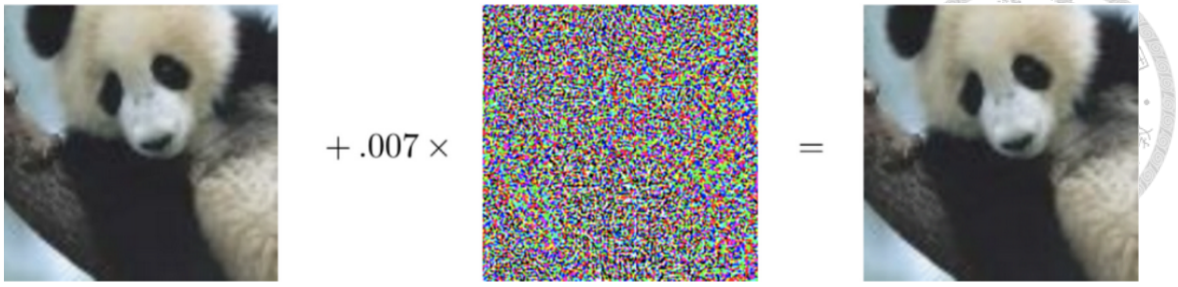


圖 2.1: 展示對抗性擾動對於類神經網路的影響，圖中左、中、右分別是熊貓的圖片、對抗性擾動與對抗性實例。

類神經網路對於圖片的判斷類別以及信心分數。正常情況下，類神經網路可以正確的辨識圖片類別，然而在圖片上加入微小的對抗性擾動後，類神經網路卻會將圖片誤判為長臂猿。值得注意的是，雖然模型的預測結果對於兩張圖片有著迥異的輸出，人眼卻很難判讀其中差異。

2.2 攻擊演算法

攻擊演算法可以依照攻擊的目的分為兩大類，分別為無特定目標攻擊（Untargeted Attack）和有特定目標攻擊（Targeted Attack）:

無特定目標攻擊演算法旨在使類神經網路判斷錯誤，例如，數字辨識中，只要生成的對抗性擾動會導致類神經網路將數字 0 辨識為 0 以外的數字，即為攻擊成功。

然而，有特定目標攻擊演算法不僅要使模型辨識錯誤，還需要將對抗性實例判斷成特定類別。例如，在數字辨識中，若攻擊的目標為 1，則生成的對抗性擾動要讓模型將數字 0 辨識成數字 1 才算攻擊成功。

雖然無特定目標和有特定目標攻擊演算法的目的不同，但是往往稍微修改公式2.1的損失函數 (Loss Function)，便可以將有特定目標攻擊轉變為無特定目標攻

doi:10.6342/NTU202301498

擊演算法，反之亦然。此外，稍微修改攻擊演算法也可以使其適應至不同的範數限制，因此，範數限制和是否為有特定目標攻擊無法成為分類攻擊演算法的依據，以下依照生成對抗性擾動時的條件將演算法區分為白盒（White Box）與黑盒 (Black Box) 攻擊演算法。

白盒演算法在尋找擾動時可以獲取欲攻擊的目標模型 (Target Model) 所有的參數值，而黑盒攻擊僅能得知目標模型針對輸入資料所計算的輸出向量，其餘的資訊均無法得知。

2.2.1 白盒攻擊

生成對抗性擾動攻擊目標模型時，此類演算法需先計算對抗性擾動經目標模型後得到的損失值 (Loss Value)，再由反向傳播演算法迭代更新對抗性擾動的數值，以最大化損失值並達到使模型誤判的目的。

然而，計算反向傳播演算法時，需要獲取模型所有參數的數值。但是在實際情況下，攻擊者較難取得目標模型的參數量值，因此，這些攻擊不易直接作用於現實情況，不過，這類演算法所產生的擾動可以被用來評估模型所受到的最嚴重攻擊情況。以下介紹常見的白盒攻擊演算法。

1. 快速梯度正負問攻擊法 (Fast Gradient Sign Method, FGSM)[12]: 為了使模型預測錯誤，此演算法旨在最大化模型的訓練損失值。詳細的作法為將輸入資料 x 經由模型 f 計算訓練損失值，再由反向傳播算法計算損失值對於模型輸入資料的梯度，並取梯度值的正負號乘以 ϵ，即為擾動，而對抗性實例 x_{adv} 則是擾動與輸入資料 x 的和，該演算法的公式如下:

$$x_{adv} = x + \epsilon \cdot sign[\vec{\nabla}_x J(\theta, x, y)] \tag{2.2}$$

doi:10.6342/NTU202301498

此演算法的目標是透過計算一次反向傳播，來最大化訓練損失值，使得模型在判斷輸入資料時出現錯誤。

2. 基礎迭代攻擊法 (Basic Iterative Method, BIM) [11]: 基礎迭代攻擊法通過多次迭代快速梯度正負問攻擊法來優化對抗性擾動，使生成擾動對於目標模型具備更強的攻擊能力，其中 $Clip_{x,\epsilon}$ 為一函數，作用為將對抗性擾動 δ 超過範數限制 ϵ 的數值設為 ϵ，該演算法的數學公式如下：

$$x_{adv}^{(t+1)} = \text{Clip}_{x,\epsilon}\{x_{adv}^{(t)} + \alpha \cdot \text{sign}(\nabla_x J(\theta, x_{adv}^{(t)}, y_{true}))\} \tag{2.3}$$

此演算法的核心思想是將快速梯度正負問攻擊法產生的擾動添加至原始資訊中，再對新的對抗性實例進行快速梯度正負問攻擊法，以生成更強的對抗性擾動。通過多次迭代，可以進一步提高擾動的攻擊能力。相較於快速梯度正負問攻擊法，此演算法需要更長的運算時間，但是在白盒攻擊的設定下，也能對模型造成更大的傷害。

3. 深層懸弄 (DeepFool) [13]: 此演算法先推導在線性二元分類器上，如何最小化擾動的 L_2 範數同時能使分類器判斷錯誤，再將結果延伸至一般的類神經網路上。

線性二元分類器對於輸入資料 x 的輸出值為 $f(x) = w^T x + b$，其中 w 是模型的權重，b 是模型的偏差。我們可以求得最小的擾動使得分類器辨識錯誤，此時擾動 δ 為 x 到決策邊界 (Decision Boundary) $H = \{x : w^T x + b = 0\}$ 的向量，亦即:

$$\delta = f(x) \cdot \frac{w}{||w||_2^2} \tag{2.4}$$

由於類神經網路具備局部線性 (Piece-wise Linear) 的特質 [12]。因此，我們可以假設在小範圍內類神經網路具備線性的特性。所以，我們可以由

doi:10.6342/NTU202301498

式2.4疊代計算對抗性擾動 ϵ，直至攻擊成功。其中類神經網路之權重 w 需透過偏微分求得梯度值。此外，公式2.4可以推廣至多元分類器上，因此，深層懸弄攻擊也可以推廣至一般的類神經網路上。

實驗結果證明，由深層懸弄演算法生成的擾動有著比快速梯度正負問攻擊法更小的範數值，這是由於在尋找決策邊界時，我們已經使用了輸出向量中競爭類別 (Competeing Class) 的資訊，亦即信心分數第二大的類別，然而，快速梯度正負問攻擊法攻擊並未直接使用此條件。

4. CW 氏攻擊 (Carlini and Wagner Attack)[14]: 在尋找對抗性擾動時，前述的演算法均構築於類神經網路類具備局部線性的特質上。但是，類神經網路與線性模型有本質上的不同。為了解決此問題，CW 氏演算法將尋找對抗性擾動視為一個最佳化問題。因此，我們只需要設計損失函數 $J(x_{adv}, y)$，同時考量攻擊能力與範數的大小，再由反向傳播演算法計算擾動以最小化損失值。論文中使用的損失函數請見公式2.5：

$$J(x_{adv}, y) = ||\delta||_p + c \cdot \max(\max(f_i(x_{adv}) : i \neq y) - f_y(x_{adv}), \kappa) \qquad (2.5)$$

損失函數由兩項組成，第一項是擾動的範數大小，第二項是擾動的攻擊能力。攻擊能力的衡量方法為競爭類別 i 的輸出值 $f_i(x_{adv})$ 減去標籤類別 y 的輸出值 $f_y(x_{adv})$，當此數值超越 κ 時，我們便轉而最小化擾動的範數值。

2.2.2 黑盒攻擊

此類攻擊演算法僅需要模型對輸入資料的輸出向量，即可生成對抗性擾動。因此，此類攻擊更貼近於現實世界中，駭客攻擊類神經網路的做法。以下介紹兩種具代表性的黑盒演算法。

doi:10.6342/NTU202301498

1. 密文填塞攻擊 (Oracle Attack) [18]：此演算法旨在訓練一個與目標模型相近的源模型（Source Model），接著，由白盒攻擊演算法在源模型上生成對抗性擾動，再將該擾動可轉移 (Transfer) 至目標模型上，以提升生成對抗性擾動攻擊目標模型的能力。為了訓練與目標模型相近的源模型，我們首先要將源模型訓練在與目標模型相同的任務上，但是兩個模型的訓練資料集不需要相同。為了增強對抗性擾動的攻擊能力，我們需要提供輸入資料給目標模型，然後獲取對應的輸出向量。接下來，我們可以使用這些輸出向量作為標籤，對源模型的參數進行微調（Fine-tune），從而讓源模型更貼近於目標模型。這樣一來，由源模型生成的對抗性擾動對目標模型的攻擊能力就會得到提升。接著，我們用白盒攻擊演算法攻擊源模型，再將得到的擾動用來欺騙目標模型。

2. 方形攻擊 (Square Attack) [15]: 此演算法的步驟為，先隨機初始化對抗性擾動，再經由不斷地從擾動中隨機增加新的正方形圖案，如果新增的方形會使模型辨識結果變差，則將方形保留於擾動中；反之，則捨棄該方形。此演算法重複以上步驟，直到模型辨識錯誤為止。

   方形攻擊的演算法本質上為隨機搜尋 (Random Search)，因此，好的初始值對於搜尋的結果影響至關重大。論文中，作者將條狀紋路作為擾動的初始值，因為卷積神經網路特別容易受此類紋路的干擾 [19]。由方形攻擊所產生的對抗性擾動對於類神經網路有很強的干擾能力。

2.3 防禦演算法

為了抵禦不同的攻擊演算法，防禦演算法應運而生。其目的旨在避免模型受到對抗性擾動的影響，從而增進模型的可靠性。防禦的策略大致可以分為檢驗輸

doi:10.6342/NTU202301498

入資料是否受到擾動污染及降低擾動對模型的影響兩大類，本文僅介紹後者。

1. 防禦性蒸餾 (Defensive Distillation) [20]: 防禦性蒸餾是一種將知識蒸餾 (Knowledge Distillation) [21] 的技術用於訓練模型，使其具有防禦擾動的能力，以下介紹防禦性蒸餾演算法。

   執行防禦性蒸餾時，我們需要兩個模型 - 初始模型 (Initial Model) 和蒸餾模型 (Distilled Model)。首先，我們要在資料集上用輸入資料 x 與標籤 y 訓練初始模型，然後再訓練蒸餾模型。蒸餾模型的訓練方式為給定輸入資料 x，預測初始模型對於輸入資料的輸出向量 $f(x)$，請見圖2.2，圖取自 [20]。

   使用初始模型的輸出向量 $f(x)$ 做為訓練資料的標簽，可以提供更多的資訊給蒸餾模型。更重要的是，模型的輸出向量比獨一餘零 (One-hot Encoding) 之標籤更有彈性，因此可以減少訓練模型時過度貼合 (Overfitting) 的可能性，使得模型更能抵禦對抗性擾動。然而，此方法被證實不能抵禦強力的攻擊演算法，如 CW 氏攻擊演算法 [14]。

2. 對抗性訓練 (Adversarial Training) [22]：對抗性訓練是一種常見且直觀的防禦演算法，透過將模型用含有對抗性擾動的資料集來訓練，我們可以增強模型對於擾動的抵抗能力。

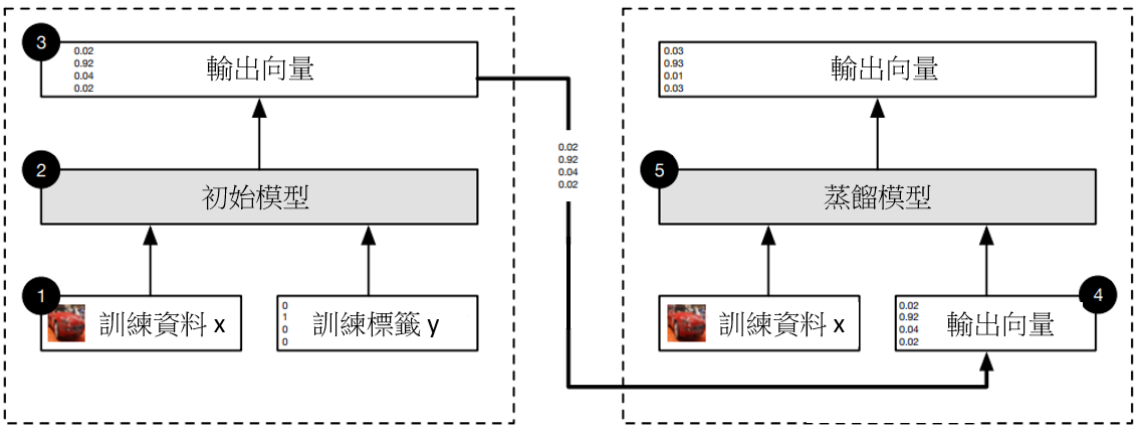


圖 2.2: 防禦性蒸餾的訓練步驟。

doi:10.6342/NTU202301498

數學上，對抗性訓練為極小化極大算法 (Min-max Optimization Problem) [22]，首先在訓練時我們會先計算對抗性實例 x_{adv}，以最大化損失函數 $J(\theta, x_{adv}, y)$。接著，更新模型的參數 θ 以最小化損失函數 J，以此降低對抗性擾動對於類神經網路的影響，請見以下公式：

$$\min_{\theta} \max_{x_{adv}} J(\theta, x_{adv}, y) \tag{2.6}$$

對抗性訓練是目前最有效的防禦演算法 [23]。然而，此訓練需要動態生成對抗性擾動，這會大幅增加模型的訓練成本，導致演算法無法延伸至需大量運算資源的模型上。

3. 隨機平滑 (Randomized Smoothing) [24]: 隨機平滑是一種將任意模型 f 轉換為平滑分類器 (Smoothed Classifier) g 的演算法；相比於一般模型，平滑模型對於對抗性擾動有著更出色的抵抗力。

將模型轉化為平滑模型的過程十分簡單，在預測輸入資料 x 的類別 y 時，我們只要將不同的高斯雜訊 z 加入相同的輸入資料中，從而生成新的輸入資料，然後統計模型的輸出類別，最終平滑模型的預測結果 c 為出現次數最多的類別，這個過程可以用以下公式表示：

$$g(x) = \arg\max_{c} P(f(x+z) = c) \; where \; z \sim N(0, \sigma^2 I) \tag{2.7}$$

高斯雜訊的變異量 σ^2 會影響平滑模型的正確性和防禦力，當變異量較大時，平滑模型在對抗擾動的干擾下正確率會更高，但對於乾淨的資料，其辨識率會降低。

此演算法能增進模型的防禦力是因為對抗性擾動的數值往往很小，因此相較於輸入資料，擾動受高斯雜訊的影響會大於輸入資料。總體而言，這種方法

doi:10.6342/NTU202301498

是一種有效且可擴展至大規模模型的防禦演算法。

4. 基於生成對抗網路的防禦 (GAN-based Defense) [25]: 生成對抗網路 (Generative Adversarial Network, GAN) [26] 是一種學習資料分佈的生成模型，許多研究旨在利用生成模型來去除對抗性擾動的影響，防禦性生成對抗網路 (Defense-GAN) 是此類研究中具代表性的演算法 [25]，其運作原理如下:

訓練生成器的方法是給定輸入資料 x 並隨機產生雜訊 z，生成器的訓練目標為產生與輸入資料相近的圖片 $G(z)$。訓練完生成器後，給定受擾動汙染的資料 x'，我們利用反向傳播演算法尋找 z^*，使得 $G(z^*)$ 相似於 x'，由於生成器訓練於乾淨資料上，因此我們期待 $G(z^*)$ 不包含擾動的資訊，如此，我們得以還原出乾淨的資料，再將其送入分類器判讀類別，請見圖2.3，圖取自 [25]。

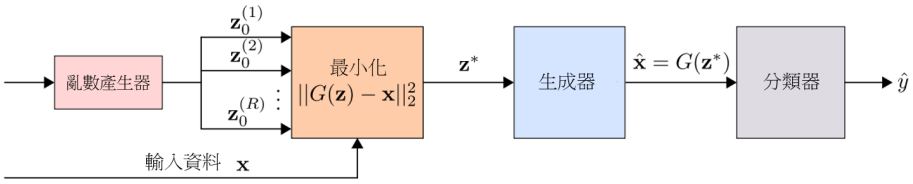


圖 2.3: 防禦性生成對抗網路的演算法示意圖。

此演算法成功的關鍵在於判讀資料前，需經過生成器去除擾動的影響，這樣會增加攻擊的難度，因為攻擊時就需要同時攻擊兩個不同功能的模型。即便如此，對抗性擾動仍然可以穿透此演算法的防線。攻擊者可以攻擊生成器，使其生成出誤導分類器的對抗性實例，進而使分類器判斷錯誤。此外，生成器在移除雜訊的過程會喪失原資料的部分資訊，導致模型的辨識率下降，此問題在複雜的資料上尤為明顯。

對抗性機器學習最關鍵的問題，是如何訓練一個對擾動有防禦能力的模型。然而，現存的防禦演算法所訓練出來的模型，往往無法在乾淨的資料上保持高辨

doi:10.6342/NTU202301498

識率，同時對擾動有著良好的抵禦能力；此外，部分成效佳的防禦演算法所需的訓練成本過於高昂，導致無法延伸至較大規模的模型上。

至今為止，無法發展有效的防禦演算法的因素之一是對於對抗性擾動作用機制的不了解。若能了解對抗性擾動的作用機制，我們就有機會從源頭遏止對抗性擾動的影響，使類神經網路在應用上更為可靠。

2.4 擾動的特性及現象

學者在研究對抗性擾動時，發現了許多擾動的特性與相關衍伸現象，這些特性/現象背後的機制大多尚未明朗。然而，了解對抗性擾動背後的機制不僅可以使我們更進一步地瞭解其本質，同時也能夠更深入地理解類神經網路的運作機制。以下我們將介紹對抗性擾動所具備的重要特性。

1. 普遍性 (Prevalence): 類神經網路在不同的應用上，例如：語音辨識、影像辨識、語意判讀；與不同的網路架構，例如：全連接神經網路 (Fully Connected Neural Network)、卷積神經網路、自注意力變換網路 (Transformer)；或是不同的訓練任務，例如: 預訓練 (Pre-training)、分類、回歸等任務上，均存在對抗性擾動，同時這些擾動會嚴重的影響類神經網路的預測結果 [27]。這意味著對抗性擾動的存在與類神經網路最基本的組成有著密切的關連，同時也凸顯出研究此議題的重要性。

2. 可轉移性 (Transferability): 從一個類神經網路生成的對抗性擾動，可以使不同的類神經網路判斷錯誤，即便兩模個型間有著不同的架構，或是訓練在不同的資料集上，這種現象稱為擾動的可轉移性 [9, 28]。

   因此，在沒有目標模型參數的情況下，攻擊者也能生成具威脅性的擾動。此

doi:10.6342/NTU202301498

外，學者也發現生成於類神經網路的擾動，除了可以可轉移至不同架構的類神經網路，還可以使傳統的機器學習算法，例如支持向量機（Support Vector Machine）、邏輯回歸 (Logistic Regression) 和決策樹（Decision Tree) 判斷錯誤 [28]。

3. 類神經網路的脆弱性 (Vulnerability): 類神經網路對於對抗性擾動十分敏感，即便擾動程度微小，甚至肉眼不易察覺擾動的添加，它都能對類神經網路的性能造成嚴重影響 [9]，請見圖2.1。這種高度敏感的特性，使得類神經網路在實際應用中面臨巨大的挑戰，因為攻擊者可以在人類無法察覺的情況下，利用微小的擾動來誤導模型，導致模型做出錯誤的判斷。

4. 對抗性訓練增強模型的解釋能力 (Interpretability): 類神經網絡中損失值對於輸入的梯度值與生成的擾動往往充滿著雜訊，因此對於人類來說很難理解梯度值蘊藏著什麼樣的資訊 [29, 30]，研究發現對抗性訓練能夠增強模型的解釋能力，使得人類可以理解梯度值所蘊含的資訊 [12, 31]，同時也能夠使生成的擾動更符合人類的感知 [32, 33]。請參見圖2.4。該圖顯示了在第一行的輸入資料下，一般模型和經過對抗性訓練的模型生成的梯度值，其中第二行圖片顯示了一般模型生成的梯度值，而第三行則是經對抗性訓練的模型所生成的梯度值。從圖中可以觀察到，一般模型產生的梯度非常不清晰。然而，對抗性訓練可以使我們從梯度值中觀察到輸入資料的輪廓。值得注意的是，在鳥的圖片中，生成的梯度值與原圖的色系有互補的趨勢。

doi:10.6342/NTU202301498

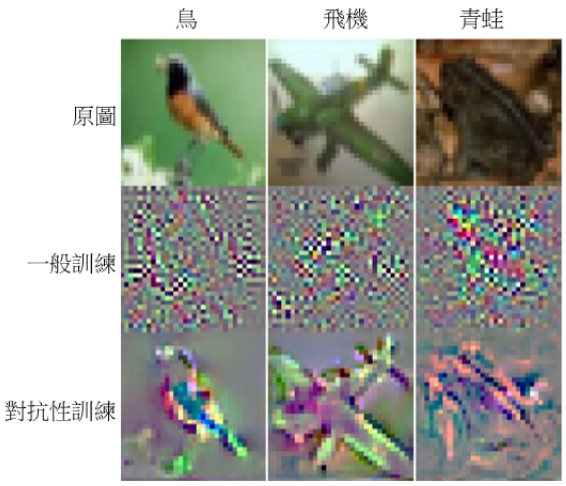


圖 2.4: 對抗性訓練增進模型的可解釋性。

這項發現有其重要性，因為在科學研究中，模型的可解釋性提供科學家額外的資訊，並有助於找出關鍵因子 [34]。此外，增進模型的可解釋性也有助於學者了解類神經網路的運作機制。

5. 模型訓練於受污染的資料集仍具高辨識率: 有學者採用了一種特殊的方法來訓練類神經網絡 [35]，該方法使用對抗性實例來訓練類神經網絡，此外，這些輸入資料的標籤為模型誤判的類別。舉例而言，在訓練分辨貓與狗的模型時，我們首先用有特定目標攻擊將資料集的貓被模型誤判為狗，再將狗攻擊成貓，然後將貓的類別標示為狗，狗標示為貓。

   接著使用這個資料集來訓練模型，照理來說訓練出來的模型於測試資料集的正確率應該會很低，畢竟訓練資料的標籤都是錯誤的。然而，該模型在 CIFAR10 的測試資料集上進行驗證時，然仍保有 63.3% 的準確率 [35]，在此資料集上隨機猜測的正確率僅為 10%。

   這項研究提出了一項新的觀點，即對抗性擾動為存在於資料集的一種特徵 (Features)，該特徵雖然對於人類而言沒有實質意義，但是對於模型的判斷卻十分具有影響力。

doi:10.6342/NTU202301498

2.5 本章總結

本章首先介紹了經典的對抗性攻擊演算法，其中部分演算法會在後續的實驗中被用到。接著，我們說明了常見的防禦演算法以及目前對抗性機器學習所面臨的困境。最後，我們講解了擾動所具備的性質，並指出這些性質背後的機制尚未完全明朗。在第六章，我們將更深入地探討這些機制。

doi:10.6342/NTU202301498

doi:10.6342/NTU202301498

第三章　相關研究

對抗性擾動對類神經網路的應用造成許多傷害，因此防止對抗性攻擊至關重要，而了解對抗性擾動造成的攻擊則是防禦的第一步。儘管學界積極的研究此議題，許多不同的學說也被提出用來解釋擾動的作用機制，但是這些學說往往無法完整的解釋所觀察到的現象，此外，不同的學說也常相互矛盾。因此，普遍的認知是擾動的作用機制仍然是未解之謎 [23, 27]。本章將介紹過去學者對於此問題所提出的觀點。

3.1　類神經網路的局部線性

此派學說認為類神經網路具備局部線性的特性 [12, 36]，而對抗性擾動來源於此特性。因為，線性模型無可避免的會受到對抗性擾動的影響，只要對抗性擾動 δ 的方向與模型之權重 w 相同，計算出來的結果會正比於權重之 L_2 範數 $||w||_2$，而 $||w||_2$ 又正比於維度，因此當 w 之維度很高時，$w^T\delta$ 值也會很大。因此，即使每一維度僅有小的改變 k，加總起來數值也會變得非常大，請見以下公式。

$$w^T x_{adv} = w^T x + w^T \delta,\ where\ \delta\ = k \cdot sign(w) \tag{3.1}$$

同時，學者也發現類神經網路具備局部線性的特性 [12]。在實驗中，學者首先記錄模型對於輸入資料的輸出值，接著加入對抗性擾動 δ，並在維持擾動方向不變

doi:10.6342/NTU202301498

的條件下，逐步調整擾動之量值，並觀察模型輸出的變化，請見圖3.1，圖取自[12]。左圖顯示模型輸出量值隨擾動大小的變動，不同線的顏色代表不同類別的輸出值，其中紅線代表輸入資料數字 4 的輸出值；右圖則呈現了不同量值的擾動加到圖片後的形狀，其中黃色圈起來的部分代表模型仍能正確判斷圖片類別。我們可以觀察到模型輸出數值與擾動量值的變化呈現線性關係，因此，此實驗說明類神經網路具備局部線性的特性，而線性模型會遭受對抗性擾動的影響，因此類神經網路也會受到擾動的影響。

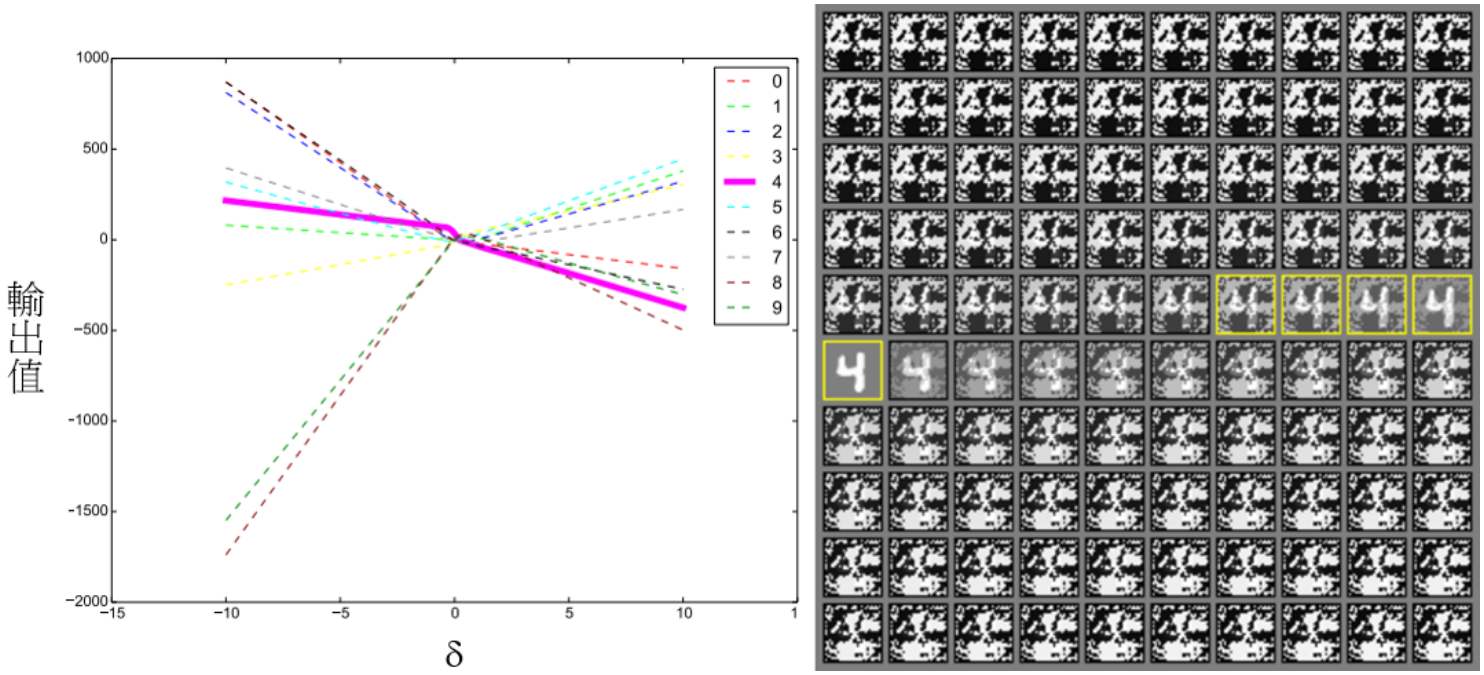


圖 3.1: 類神經網路的局部線性。

此外，學者根據模型局部線性的特性分析對抗性擾動的可轉移性。由於模型為局部線性，因此通過分析擾動與模型權重 (梯度值) 的餘弦相似度 (Cosine Similarity)，我們可以計算對抗性空間 (Adversarial Space) 的維度，而對抗性空間代表此空間內的點都可以使類神經網路判斷錯誤。學者發現抗性空間十分廣闊，此外，兩個不同模型中，它們的對抗性空間有相當大的部分重疊，因此對抗性擾動可以從一模型轉移至另一模型 [36]，對抗性擾動空間大程度的重疊意味著不同模型學習的權重十分相似。

doi:10.6342/NTU202301498

此學說雖然解釋了對抗性擾動的普遍性、脆弱性、可轉移性，但是仍存在著一些瑕疵。首先，模型的輸入、輸出值的變動近似於線性關係並不意味著模型為局部線性，其中關鍵的差異在於權重 w 是否維持定值。此外，現實中存在不具備局部線性的類神經網路，該模型對於對抗性擾動並沒有更強的抵禦性 [37]。

此外，當模型訓練在複雜的資料集上，如 ImageNet[38]，各個模型所學習到的權重差異大，權重間的餘弦相似度近乎垂直 [39]，但是對抗性擾動仍然具有可轉移性。

3.2 對抗性實例偏離資料分佈

此派學說認為，對抗性擾動會導致輸入資料偏離一般正常訓練資料的分佈，導致類神經網路判斷錯誤 [40, 41]。

一般情況下，我們將模型訓練在正常資料上，因此模型僅能在正常的資料分佈上進行辨識。但是，對抗性擾動會改變資料的分佈，當模型之決策邊界貼合於資料分部的流型 (Manifold) 時，對抗性擾動會使資料穿透模型的決策邊界，從而使模型產生錯誤的判斷，圖3.2展示了此現象。圖中資料分為圈與叉兩類，這些資料均分佈於黑色的流型上，紅色流行代表模型的決策邊界，模型的判斷依據為紅色流行之上為圈、之下為叉。由於紅色流行貼合於黑色流行，導致輸入資料稍微偏離原資料分佈，就會使類神經網路判斷錯誤。

此學說認為降低決策邊界與資料流型的貼合程度，可以有效地增進模型對於擾動的抵禦能力。而模型的過度貼合源自於訓練時的過度擬和，因此，訓練時正則化 (Regularize) 模型參數可以加強模型的抵禦能力。

此論點更趨向於描述現象，而非解釋對抗性擾動存在的原因。畢竟，對抗性

doi:10.6342/NTU202301498

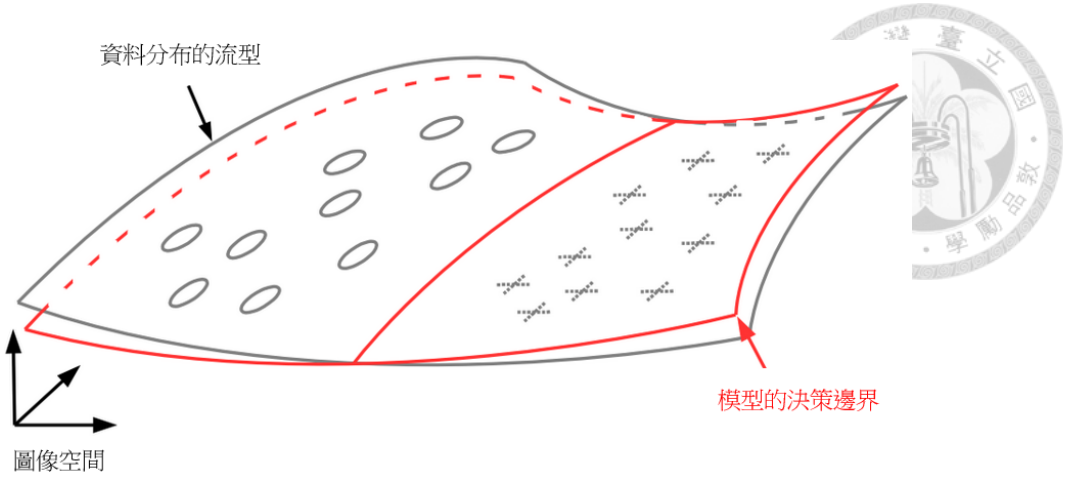


圖 3.2: 輸入資料與決策邊界的分佈。

擾動會使輸入資料的分佈異於一般資料，否則模型不會判斷錯誤。而決策邊界也需要貼近於資料所分佈之流型，否則微小的擾動無法使模型判斷錯誤，問題的核心應該是模型的決策邊界緊密貼合資料分佈流型的詳細成因。

3.3 擾動改動非強健性特徵

此派學說源於觀測到的實驗現象，亦即模型訓練於受污染的資料集仍具高辨識率，請見章節 2.4。

此學說的觀點為，用來判讀輸入資料的資訊稱為特徵 (Features)，而特徵分成兩類，強健性特徵 (Robust Feature) 與非強健性特徵 (Non-robust Feature)。強健性特徵為人眼用來分辨輸入資料類別的特徵，而非強健性特徵則是人類無法辨別，但是對於模型卻有著高指向性的資訊 [35, 42]。

非強健性特徵的存在導源於訓練模型時，訓練目標為最大化模型的準確性，因此，模型不會理會人類依據哪些特徵來判讀資料，它會學習任何對提升準確性有幫助的特徵。因此，模型最終可能學習到一些對於人類來說並非辨識圖片類別的特徵，但對於機器的辨識卻具有重要的價值。

doi:10.6342/NTU202301498

對抗性擾動的功能就是改變這些非強健性特徵，雖然人類無法判讀其中意義，但是這些特徵卻能對模型的判讀產生巨大的影響。此學說也可以解釋對抗性擾動的可轉移性，由於任何兩個模型都可能學習類似的特徵（包含強健性與非強健性特徵），因此更改一模型之非強健性特徵也可能影響另一模型。

然而，此派學說無法說明為什麼非強健性特徵對模型的影響遠勝於強健性特徵；而非強健型特徵是否存在仍有待討論，畢竟，資料集的標籤是依照人類的認知而分類，很難想像存在一種不為人所知，又能區分資料類別特徵。

3.4 數學證明擾動的存在

有些學派傾向用數學來證明對抗性擾動的存在，此類學說往往先假設類神經網路 f 符合一些性質，然後證明在 $||\delta||_p < r$ 的情況下，是否存在擾動 δ，使得 $f(x) \neq f(x+\delta)$。

一些常見的假設包含限制類神經網路的利普希茨 (Lipschitz) 常數與活化函數的種類，再證明對抗性擾動在該條件下是否必然存在 [43–45]。

這些推導出的結論往往僅適用於利普希茨常數小的情況，但是，利普希茨常數對於模型的本質有著關鍵性的影響，類神經網路與線性模型的差異即可視為利普希茨常數不同所導致，因此，這些結論是否能延伸至現實世界中仍有待討論。

3.5 本章總結

解釋對抗性擾動存在原因的學說眾多，本章回顧四種常見的學派。近年來，研究擾動本質的論文相對較少，相反地，更多的研究投入在不同應用領域上研發攻擊與防禦演算。我們將在下一章說明我們對於此議題的觀點。

doi:10.6342/NTU202301498

doi:10.6342/NTU202301498

第四章　探求擾動的本質

本章首先介紹我們對於擾動作用機制的假設，然後，我們闡述提出此假設所面臨的難題，再介紹實驗方法、架構，最後執行實驗以驗證假設的正確性。

4.1　本論文所提出之假設

我們認為類神經網路可以正確分類輸入資料的前提是，它必須對人類用以區分圖片類別的資訊 (以下簡稱可識別資訊) 給予高梯度值，例如汽車的輪胎、飛機的機翼或物體的輪廓，這個想法來源於前人對類神經網路梯度值的研究 [29, 30]。

因此，我們認為若要使模型辨識錯誤，則對抗性擾動必須要改變圖中的可識別資訊，所以，擾動也會含有這些可識別資訊的若干成分。

同時，我們認為擾動存在著兩種不同的作用機制，分別為遮掩效應 (Masking Effect) 與生成效應 (Generation Effect)[46]。

遮掩效應旨在遮蓋/減小原圖中可識別資訊的像素值，以增加模型辨識的難度，舉例來說若要使模型將母雞的圖片判斷錯誤，產生的對抗性擾動會含有具負號的母雞資訊，這個效應多見於無特定目標攻擊中。

生成效應則在原有的資料上新增可識別資訊，使得類神經網路將對抗性實例判斷成特定的類別。舉例來說，若要用無特定目標攻擊演算法使模型將輸入的母

doi:10.6342/NTU202301498

雞圖片判斷成公雞，則擾動中可能會含有雞冠的資訊。

我們提出擾動具備兩種特性，亦即遮掩與生成效應，而這兩種資訊符合人類的感知，並且為導致模型辨識錯誤的原因之一，同時，我們也發現這種現象存在於不同的攻擊演算法中。

4.2 本論文提出假設之相關研究

過往研究發現在有目標性攻擊設定、特定條件下產生的擾動中可以觀察到人類可識別資訊。埃氏的論文 [47] 提出使用多個模型同時優化擾動，並且在模型輸入層加上視網膜層 (Retinal Layer)，會導致人類在極短的時間內（60-70 毫秒）於對抗性範例的辨識正確率降低約一成。值得注意的是，一旦延長了人類判讀對抗性範例的時間，則人類的辨識正確率基本上不會受到對抗性擾動的任何負面影響。該論文作長推測，這是因為在極短的時間內，大腦的神經元處理模式變得更類似於前饋型 (Feed Forward) 類神經網路的處理模式。然而，值得一提的是，我們在作者所提供的一部分例子中，確實觀察到了具有人類可識別資訊的對抗性擾動。其他研究也觀察到，在有目標性攻擊中，增強擾動的穩健性會導致部分生成的擾動更符合人類的認知 [48, 49]。

不同於前人的研究方法，我們加總不同模型產生的擾動，就可以直接觀察到隱藏於其中的人類可識別資訊（在圖片中添加額外雜訊會進一步提升其可識性），這顯示可識別資訊本身就存在於擾動中。此外，本研究主要探討無目標性攻擊，並發現人類可識別資訊普遍存在於各類圖片和攻擊演算法中，而不僅僅侷限於有目標性攻擊設定下的特例，我們也進一步驗證了可識別資訊會導致模型辨識錯誤。這項發現顯示穩健性特徵應為導致模型辨識錯誤的主要原因，而不是非穩健性特徵，這與廣為人知的學說相左 [35]，也進一步讓我們了解擾動的性質。同時，

doi:10.6342/NTU202301498

我們也由可識別資訊導致模型辨識錯誤的假設出發，發現許多擾動的相關現象與成因可以得到合理的解釋。

4.3 本論文所提之假設面對的挑戰

提出對抗性擾動含有人類可識別資訊具有挑戰性，因為可識別資訊並不容易直接從對抗性擾動中辨認出來，此外，通用對抗性擾動 (Universal Adversarial Perturbation) 的存在 [16]、多樣的攻擊演算法與設定、以及對於類神經網路的不了解，都增加了提出此學說的難度。

4.3.1 擾動的可識別性

一般情況下，辨認對抗性擾動中的可識別資訊很困難，因為這些資訊通常不完整，並且受到雜訊嚴重的干擾。圖4.1中展示了一輛救護車與不同演算法所產生的擾動，我們可以發現，人眼很難從擾動中直接觀察到救護車的輪廓，也因此我們很難想像其中具備可識別資訊。

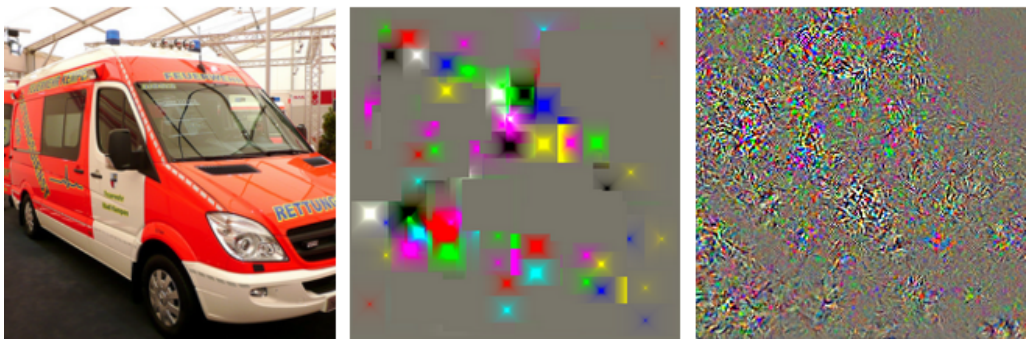


圖 4.1: 無法辨別擾動中的可識別資訊。左、中、右分別是原始圖片、方形攻擊與深層懸弄攻擊所產生的擾動。

doi:10.6342/NTU202301498

4.3.2 通用對抗性擾動的存在

通用對抗性擾動神奇的地方是，同一擾動可以使多種輸入資料均被模型判讀錯誤，此外，通用對抗性擾動也具有可轉移性 [16]。圖4.2中左圖展示了 VGG-16 生成的通用對抗性擾動。

顯而易見的，沒有相同的可識別資訊會重複出現於各式各樣的圖片中，因此，我們的上述假設無法解釋通用對抗性擾動的存在。然而，此擾動的作用機制仍然是學界尚未解開的謎團 [50]。

我們對於通用對抗性擾動的作用機制有著以下的猜想: 卷積神經網路在構成複雜圖案時，是由前端的卷積核 (Convolutional Kernel) 和下一層的卷積核逐層進行卷積運算所求得。前端的卷積核具有重複的幾何形狀，例如條紋、曲線、斜線等 [51]，這些形狀是構成複雜圖案的基石，因此在辨別不同圖片時，這些卷積核都會發揮作用。因此，即便判斷不同的圖片，只要前端的卷積核與圖片進行卷積運算後產生偏差，模型的判斷就會受影響。透過對前端卷積核的線性組合產生擾動，可以最大化卷積後的偏差，讓模型針對不同種類的圖片都會判斷錯誤，從而形成通用對抗性擾動。

從其他學者的研究中，我們發現通用對抗性擾動和卷積核權重之間存在一些共通的基本紋路，請見圖4.2，左圖取自 [52]、右邊兩張圖取自 [53]。

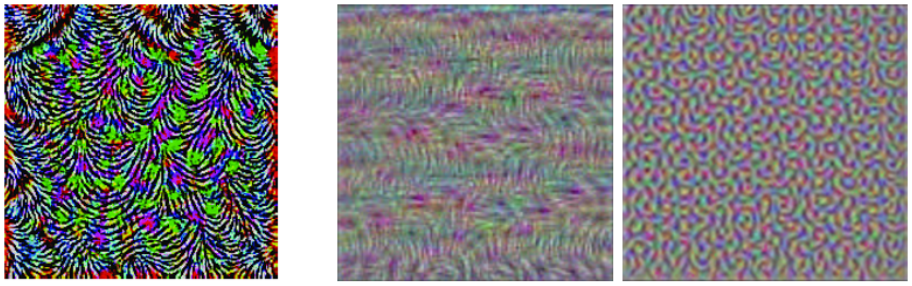


圖 4.2: 通用對抗性擾動與卷積層的權重，左圖是通用對抗性擾動，右邊兩張圖是 VGG-16 模型的權重。

doi:10.6342/NTU202301498

4.3.3 問題的複雜性

研究對抗性擾動導致模型辨識錯誤的原因具有挑戰性，因為許多因素都可能造成模型辨識錯誤。在白盒攻擊的情況下，即使擾動沒有可識別的資訊，只要擾動與模型梯度的雜訊相符，也可能會造成模型的誤判。添加高斯雜訊、通用對抗性擾動、高頻紋路 [19] 和可識別資訊，都會影響模型的辨識。此外，多樣的攻擊演算法和攻擊設定也增加了問題的複雜度。在眾多的因素的影響下，探討此問題就變得十分困難。

4.3.4 對於類神經網路的不理解

類神經網路擁有龐大的參數量，在使其具有強大功能的同時，也使其運作機制變得極其複雜。由於學術界尚未完全瞭解類神經網路判斷輸入資料類別的依據，因此無法知道類神經網路和人類的判斷依據是否相同 [54, 55]。這使得分析擾動導致模型辨識錯誤的原因更加困難。

4.4 解析可識別資訊

雖然擾動看起來並不符合人類的感知，但是我們仍然認為它包含著可識別資訊。我們認為這是由於擾動中存在大量的雜訊和資訊缺失所致。在接下來的內容中，我們將解釋這兩個現象與其成因。

類神經網路的梯度值通常非常嘈雜 [30]，我們認為這是因為梯度值的雜訊和輸入資料的內積接近於零。因此，模型無法感知梯度中雜訊的存在，自然也無法在正常訓練的框架下消除雜訊。擾動通常來自於類神經網路的梯度值充滿雜訊 (Very Noisy)。

doi:10.6342/NTU202301498

擾動中的可識別資訊常常不完整，因為當類神經網路對輸入資料進行分類時，它只需要部分的信息就足以對輸入資料進行正確的分類。因此，類神經網路不一定會學習到完整的資訊，這也意味著擾動不需要修改圖片中所有的信息就可以欺騙類神經網路。

4.5 實驗方法

為了驗證擾動含有可識別資訊，我們需要在不更改擾動本質的情況下，重現擾動的可識別資訊。因此，我們要減少擾動的雜訊，並重構其缺失的資訊，方能呈現擾動的可識別資訊。

本研究使用的方法是針對同一張圖片，在不同的類神經網路中產生擾動，再加總這些擾動。由於不同類神經網路產生的擾動帶有相異的雜訊，同時缺失的資訊通常互相獨立，因此，加總擾動可以有效地降低雜訊的影響，同時拼湊缺失的資訊。最重要的是，此方法可以大程度地保留對抗性擾動的本質。

然而，受限於類神經網路的數量，我們無法獲取足夠的擾動。受梯度平滑演算法（SmoothGrad）的啟發 [30]，我們將不同的高斯雜訊加入相同的圖片中，這樣可以大幅增加擾動的數目。因為只要輸入資料略有差異，產生出來的擾動就會不同。最後，我們加總針對同一張圖片所產生的擾動，以獲得最終的對抗性擾動。

接下來，我們需要驗證擾動是否包含可識別的資訊，同時，我們也需要驗證這些資訊是否為導致模型判斷錯誤的原因。因此，我們會定量評估擾動的可識別性，以及測試並驗證可識別資訊的攻擊能力。

doi:10.6342/NTU202301498

4.6 實驗設定

本節將詳細介紹實驗中我們如何產生與呈現對抗性擾動，我們總共採用了三種資料集，分別為 MNIST[56]、CIFAR10[57]、與 ImageNet[38] 資料集，針對不同的資料集，在每個子章節中，我們會分別介紹各個資料集所使用的參數。

在這項實驗中，我們使用類神經網路來生成和評估對抗性擾動。我們將所有用於實驗的模型分為兩類，分別為源模型和測試模型。源模型用於生成對抗性擾動，而測試模型用於評估擾動的攻擊強度和可識別性。

4.6.1 輸入資料

由於實驗中我們需要產生大量的擾動，這會需要龐大的運算資源，因此，我們僅能選擇資料集中一部分的圖片來進行實驗。由於 MNIST 和 CIFAR10 資料集均包含 10 種類別，因此，我們選擇測試集 (Testing Set) 每個類別中前 10 張圖片進行實驗。對於這兩個資料集，我們各會得到 100 張圖片。

然而，ImageNet 資料集有 1000 種類別，因此，我們從中挑選了 20 種類別[1]，並使用驗證集 (Validation Set) 中每類的前 10 張圖片進行實驗，最終我們獲得了 200 張圖片。我們使用驗證集的原因是因為 ImageNet 的測試集並未公開。所有圖片的像素值都是介於 [0,1] 之間。

[1]大白鯊、公雞、樹蛙、綠曼巴、大熊貓、救護車、穀倉、棒球、掃帚、子彈列車、出租車、大砲、茶壺、泰迪熊、電車、錢包、檸檬、披薩、杯子、雛菊。

doi:10.6342/NTU202301498

4.6.2 產生擾動

實驗中，我們主要通過兩種不同的設定來產生對抗性擾動：（1）單模型設定（2）雜訊多模型設定，以下將介紹此二種設定。

4.6.2.1 單模型設定

在這個設定中，我們使用了一般的方法來產生對抗性擾動，即將圖片送到單一的源模型中生成擾動。

4.6.2.2 雜訊多模型設定

這個設定的目的在於驗證擾動是否包含人類可識別的資訊。因此，我們需要針對每一張輸入圖片生成大量的對抗性擾動。數學上這意味著，對於一筆輸入資料 x，我們首先產生 n 個平均值為 0，標準差為 σ 的高斯雜訊 $N(0,\sigma^2)$。對於第 i 個源模型，我們計算出輸入資料所對應的擾動 $\delta_i(\cdot)$，最後我們將所有產生的擾動取平均值，請見公式4.1：

$$\delta(x) = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\delta_i(x + z_{ij}(0,\sigma^2)),\ z_{ij} \sim N(0,\sigma^2) \tag{4.1}$$

除了以上兩種設定，我們也測試了雜訊單模型與多模型設定，此部分將在下一個章節介紹。

4.6.3 模型架構

實驗中，我們需要源模型來產生擾動，測試模型來檢驗擾動。本節介紹實驗中不同資料集所使用的模型架構。

doi:10.6342/NTU202301498

4.6.3.1 MNIST 的模型架構

所有用於 MNIST 實驗中的模型均為自行訓練的模型，此外，每一個模型都採用了相同的架構，其架構類似於 VGG-16[58]。然而，在訓練模型時，每個模型的初始化參數並不相同，因此訓練出來的模型也不一樣。在實驗中，我們共訓練了 101 個模型，在雜訊多模型設定下，總共有 100 個源模型產生對抗性擾動；在單模型設定下，僅有一個源模型產生擾動。在這兩種設定下，我們使用相同的測試模型，同時，測試模型不會與源模型重複。

4.6.3.2 CIFAR10 的模型架構

在 CIFAR10 實驗中，我們使用了 70 個模型，這些模型來自於 PyTorchCV[59]。在雜訊多模型設定下，我們選取了 66 個模型作為源模型。在單模型設定中，源模型為 ResNet-56[60]，這與測試模型中的 ResNet-56 模型是同一模型，因此，在單模型設定下，我們不僅可以了解擾動的黑盒攻擊能力，同時也可以評估模型的白盒攻擊效力。我們使用四個模型做為測試模型，包括 DenseNet-40[61]、DiaResNet-164[62]、PyramidNet-110[63] 和 ResNet-56，選擇這些模型是因為這些模型架構較具代表性，且彼此的差異性較大。值得注意的是在雜訊多模型的設定下，源模型和測試模型的架構完全不同。

4.6.3.3 ImageNet 的模型架構

在 ImageNet 實驗中，我們下載了 274 個模型，這些模型也來自於 PyTorchCV，其中有 270 個模型用作源模型，剩餘的 4 個模型做為測試模型。由於生成 ImageNet 的擾動需要較長的運算時間，因此我們選擇模型的浮點運算數 (FLOPs) 均少於每秒 32 億次。在單模型設定中，源模型 ResNet-50 與測試模

doi:10.6342/NTU202301498

型中的 ResNet-50 相同。在雜訊多模型設定下，我們使用 270 個源模型，這些模型與測試模型都不相同。測試模型包括 VGG-16、ResNet-50、DenseNet-121 和 BN-Inception[64]。這些測試模型是常用於測試擾動攻擊能力的模型，且彼此間有較大的差異性。

4.6.4 攻擊參數

在實驗中，我們對基礎迭代攻擊法和 CW 氏攻擊進行了有特定目標和無特定目標攻擊的評估。由於深層懸弄攻擊沒有有特定目標攻擊的演算法，因此，我們只測試了其無特定目標攻擊演算法。我們將不同的攻擊演算法應用在三個資料集上，由於這三個資料集資料的維度以及特性不同，因此攻擊演算法所需的參數也不一樣，以下先介紹各種演算法選取參數的方法，再詳列個資料集的攻擊參數。

4.6.4.1 基礎迭代攻擊法

此演算法共有三個參數，分別是擾動的 L_{inf} 範數量值 ϵ、每次進行反向傳播演算法時改變擾動的大小限制 α 以及計算擾動的迭代次數，詳情請見章節2.2.1。對於不同的資料集，學術界對於 ϵ 的數值有特定的限制。在本研究中，我們基於生成擾動的品質和可負擔的運算資源，選擇了迭代 50 次來進行實驗。而 α 是根據 2 倍的 ϵ 除以迭代次數所求得的，這麼做的目的是希望在優化擾動的過程中，生成擾動的 L_{inf} 範數能夠達到 ϵ 的數值。無論是有特定目標攻擊還是無特定目標攻擊，我們都使用相同的參數設定。

doi:10.6342/NTU202301498

4.6.4.2 CW 氏攻擊

此演算法共有三個參數，包括 c、κ 與迭代次數，其數學意義請參見公式2.5。在攻擊中，我們將迭代次數設定為 1000，並且我們將 c 與 κ 的值皆設定為 5，我們不按照原論文 [14] 將 c 與 κ 的值分別設定為 1 與 0 是因為加大 c 與 κ 可以使產生的可識別資訊更為明顯。

4.6.4.3 深層懸弄攻擊

此攻擊包含兩個參數，分別為 ξ 與迭代次數。由於深層懸弄攻擊所計算的擾動剛好位於決策邊界，為了使模型產生誤判，我們必須將擾動乘以 $(1+\xi)$，以越過決策邊界。在實驗中，我們將 ξ 設為 0.02，迭代次數為 50。

詳細的攻擊參數請見表格4.1[46, 65]。

資料集	攻擊模式	基礎迭代攻擊法	CW 氏攻擊	深層懸弄攻擊
MNIST	無特定目標	ϵ: 0.2, α: 0.008, 迭代次數:50	c: 1, κ: 0, 迭代次數: 1000	ξ:0.02, 迭代次數: 50
	有特定目標	同上	c: 1, κ:20, 迭代次數: 1000	無特定目標攻擊
CIFAR10	無特定目標	ϵ: 0.03, α: 0.0012, 迭代次數:50	c: 1, κ: 0, 迭代次數: 1000	ξ:0.02, 迭代次數: 50
ImageNet	無特定目標	ϵ: 0.02, α: 0.0008, 迭代次數:50	c: 5, κ: 5, 迭代次數: 1000	ξ:0.02, 迭代次數: 50
	有特定目標	同上	同上	無特定目標攻擊

表 4.1: 實驗中使用的攻擊演算法及對應的參數。

4.6.5 高斯雜訊

在本實驗中，我們複製輸入圖片，並添加不同的高斯雜訊至圖片中。由於資料集、模型數量及攻擊演算法均會影響生成擾動的可識別性。因此，在不同的設定下，我們需要複製不同數目的圖片，並添加不同程度的高斯雜訊。

我們發現複製越多的圖片，生成的擾動會越清楚，但是運算時間也會變得更長。此外，添加高斯雜訊的標準差越大，生成的擾動也越為清晰，但這也意味著

doi:10.6342/NTU202301498

更大程度的更動擾動的本質。因此，在選擇參數時，我們執行過不同的實驗，以確保在運算資源可負擔的情況下，添加標準差最小的雜訊至擾動中，並與擾動的可識別性取得平衡。詳細的實驗數據請參見表4.2[46, 65]。

在表格中，第 (i,j) 元素代表在雜訊多模型的設定下，對於第 i 個資料集和第 j 個攻擊演算法，我們複製了圖片 n 次並加入平均值為 0、標準差為 σ 的高斯雜訊。擾動總數則是針對一張圖片，由公式4.1加總最終的擾動時，我們總共需要由源模型生成多少個擾動。

資料集	基礎迭代攻擊法	CW 氏攻擊	深層懸弄攻擊	擾動總數
MNIST	n: 20, σ: 0.2	n: 20, σ: 0.2	n: 20, σ: 0.2	2000
CIFAR10	n: 100, σ: 0.05	n: 100, σ: 0.05	n: 20, σ: 0.1	6600[2]
ImageNet	n: 10, σ: 0.02	n: 10 , σ: 0.05	n: 10 , σ: 0.05	2700

表 4.2: 實驗中使用的資料集與對應的雜訊參數。

4.6.6 呈現擾動

產生對抗性擾動後，我們要將擾動以圖片的形式呈現，因此需要調整擾動的倍率。需要注意的是，由於有特定目標和無特定目標攻擊的作用機制不同，因此呈現擾動的方式也不一樣。

我們對於無特定目標攻擊的呈現方法是先將擾動乘以-1，然後線性調整其平均值和標準差，使其與資料集的平均值和標準差相符。將擾動乘以-1 是因為無特定目標攻擊常常會產生遮掩效應；而平均擾動後，像素的量值會產生偏差，造成顏色失真。因此，調整擾動的均值和標準差可以使其顏色更貼近輸入資料的色系，有助於我們觀察現象。

生成效應通常出現在有特定目標攻擊中，因此我們只需要讓擾動變得更明顯

[2]由於運算量過於龐大，深層懸弄攻擊的擾動總數僅為 1320

doi:10.6342/NTU202301498

即可。我們採用的方法是等比例放大擾動，放大的比例會根據不同的資料集而有所變更。例如，在 MNIST 資料集中，我們將最大值放大為 1；而在 ImageNet 資料集中，我們將最大值放大為 0.5。

4.7 改良演算法

在加總大量對抗性擾動的過程中會產生一些問題，因此我們需要修改既有的演算法，以符合我們的實驗需求，以下介紹我們修改演算法的原因與方法。

4.7.1 裁剪效應

在實驗中，我們處理對抗性實例的方法不同於常規做法。標準的處理方法是將像素值裁剪至 0 和 1 之間，但我們不這麼做。這是因為裁剪會影響擾動的可識別性，例如，如果原圖的像素值為 1，則裁剪後僅會保留負的擾動值，呈現擾動時需要乘以-1，因此，裁剪會增大擾動值，進而使呈現的擾動更貼近於原圖。然而，在實驗中，我們很重視擾動的可識別性要源於攻擊演算法，因此我們允許像素值超出 0 和 1 之間的範圍。

從圖4.3可以觀察到上述的現象，圖中左、中、右三張圖，分別是檸檬的圖片、裁剪後、以及未裁剪的擾動。我們可以發現相較於未裁剪的擾動，裁剪後的擾動有著更鮮豔的顏色，同時其輪廓更貼近於原圖。

4.7.2 校正輸出值

在輸入圖片中添加高斯雜訊會改變模型的輸出結果，從而影響生成的對抗性擾動。例如在深層懸弄攻擊中，加雜訊可能會導致模型錯誤辨別圖片，導致演算

doi:10.6342/NTU202301498

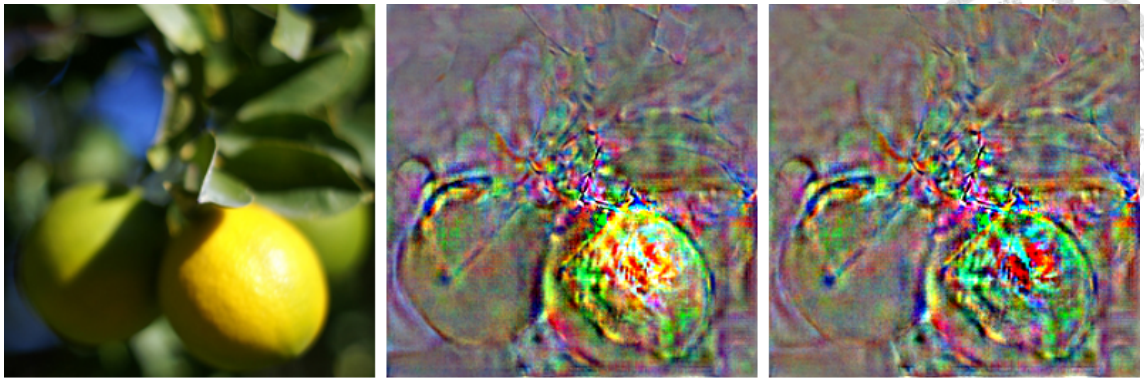


圖 4.3: 裁剪效應，左、中、右側圖片分別是檸檬的圖片、裁剪後及未裁剪的擾動。

法無法啟動。另外，有些演算法會選擇分數次高的類別作為攻擊的目標，例如 CW 氏攻擊。此時，加雜訊會影響模型輸出值，進而更改分數次高的類別。為了最小化雜訊對擾動的影響，我們引入了校正輸出值的方法。

我們採用的方法是先求得校正向量 $calib.$，即原圖的輸出向量 $f(x)$ 減去加雜訊後的輸出向量 $f(x + N(0, \sigma^2))$。之後，在計算擾動 $\delta(x)$ 的輸出向量時，我們會加上校正向量，以消弭雜訊的影響，請見以下的數學式。

$$\begin{aligned} f'(x + N(0,\sigma^2) + \delta(x)) &= f(x + N(0,\sigma^2) + \delta(x)) + calib. \\ calib. &= f(x) - f(x + N(0,\sigma^2)) \end{aligned} \tag{4.2}$$

由於模型在原圖附近的輸出值呈現局部線性的特性，因此加入少量的雜訊後，我們可以透過校正的方式來近似沒加雜訊的輸出值。

4.7.3 加速演算法

使用深層懸弄演算法產生擾動時，我們需要計算每個類別對輸出值的梯度值，以確定擾動的方向。然而，在 ImageNet 實驗中，我們需要處理 1000 個類別，這使得用深層懸弄演算法計算擾動變得非常昂貴。在 ImageNet 資料集與雜訊多模型的設定下，用單顆 Telsa V100 圖形處理器 (Graphics Processing Unit, GPU) 計算

doi:10.6342/NTU202301498

實驗所需的資料要花約三年的時間，因此，我們提出了一種改進方法，我們僅計算輸出值前 10 大類別的梯度值，並由這 10 個類別決定擾動的方向，這種改進能夠提高計算速度 100 倍。我們只在 ImageNet 實驗中加速深層懸弄演算法。

4.8 實驗結果

4.8.1 無特定目標攻擊

實驗中，我們從不同的資料集中獲取圖片，並在單模型和雜訊多模型的設定下，由基礎迭代攻擊法、CW 氏和深層懸弄無特定目標攻擊演算法產生擾動，並將計算出來的擾動乘以-1 縮放後呈現在圖4.4中 [46, 65]。

在三個資料集中，我們均觀察到在單模型設定下，產生的擾動充滿雜訊，並且很難觀測到可識別資訊。然而，在雜訊多模型設定下，產生的擾動可以減少背景雜訊，同時重建缺失的資訊。因此，我們可以觀察到與原圖相似的擾動，並驗證了先前的猜想，即對抗性擾動含有可識別資訊。

值得注意的是，在圖片中的第六行，生成的擾動只有狗的臉部與原圖相似，這意味著可識別資訊並不需要完全貼近輸入圖片，有時只要具備物體特徵的一部分，擾動就足以成功混淆模型。

此外，我們觀察到無特定目標攻擊生成的擾動除了有遮掩效應外，還具備了生成效應。例如，在圖片中的第二行，當數字 1 遭受基礎迭代攻擊法和 CW 氏攻擊時，生成的擾動分別顯示出模糊的 4 和清晰的 7 的特徵。當使用有特定目標攻擊模式時，我們可以觀察到更明顯的生成效應，關於生成效應的介紹請見章節4.1。

doi:10.6342/NTU202301498

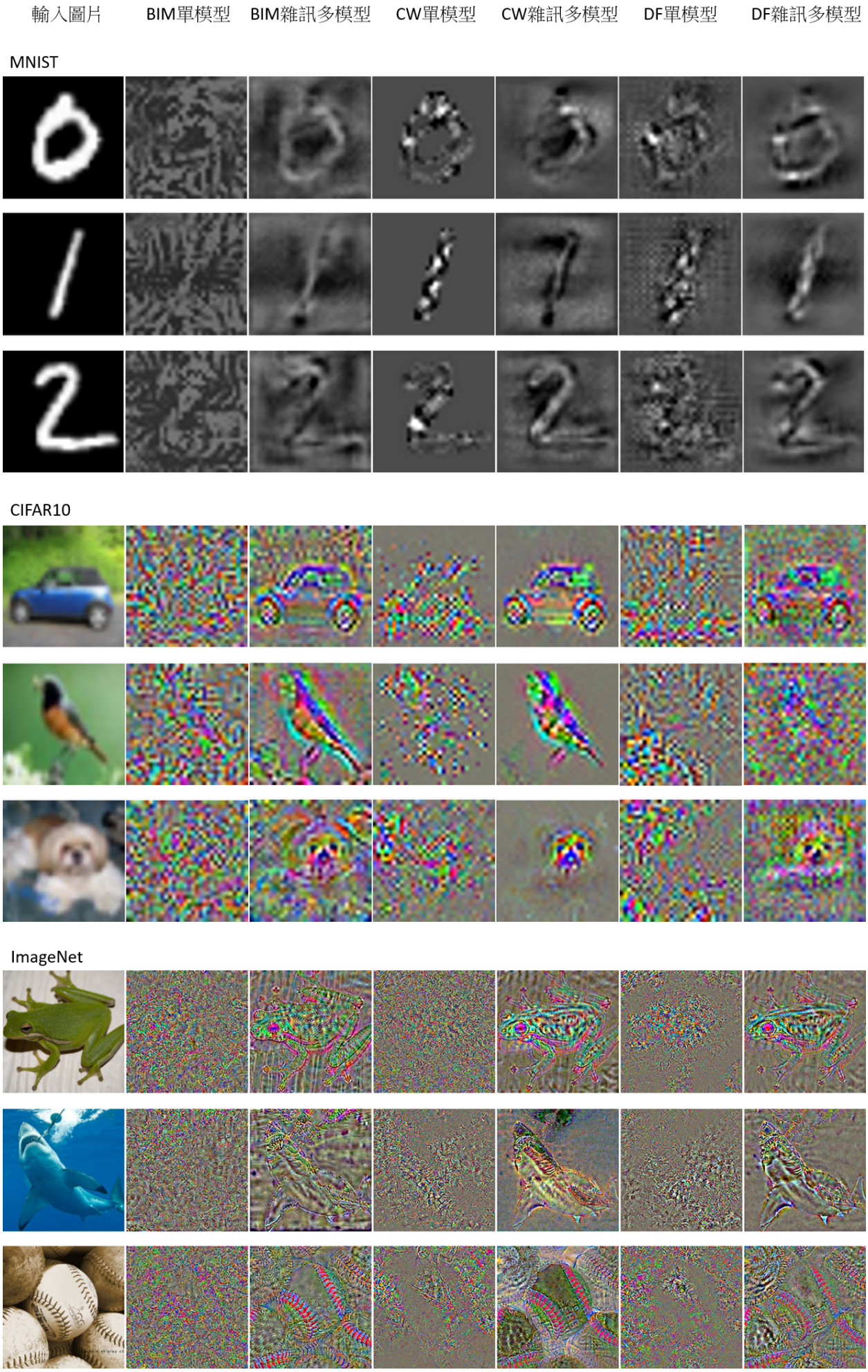


圖 4.4: 無特定目標攻擊演算法生成的擾動。

doi:10.6342/NTU202301498

4.8.2 擾動的可識別性

本文強調對抗性擾動具備人類可識別資訊，因此，我們要做實驗來驗證擾動的可識別性，然而，學界對於評估可識別性的方法仍未有定論 [66]，因此，我們進行了人類的評估實驗，同時也使用類神經網路進行評估。這兩種評估方法大致相同，只是測驗對象與前處理方式不同。在進行評估前，我們會先縮放對抗性擾動並乘上-1。接著，將每個擾動的標籤設置為輸入圖片的標籤，因為遮掩效應在無特定目標攻擊演算法中更為顯著，因此乘上負號後，擾動應該與原圖相似。

4.8.2.1 機器評估

以下將介紹對於三個資料集所產生的擾動，我們進行機器評估實驗的設定。

MNIST 資料集產生擾動的前處理方式為，我們先將擾動乘以-1，然後，將大於 0 的數值設為 1，小於 0 的值設為 0。這是因為 MNIST 資料集中數值主要分佈於 0 或 1，這樣的處理有助於提升評估的正確率。實驗中使用的評估模型與章節4.6.3中的測試模型一樣。

在 CIFAR10 中，我們的前處理方式是將擾動乘以-1，並線性縮放擾動使其平均值與標準差符合於 ImageNet 資料集的平均值與標準差，我們再乘以 0.05。我們通過實驗發現這種前處理可以大幅的提升模型評估的正確率。接著，我們使用章節4.6.3中測試模型裡的 PyramidNet 做為評估模型。

在 ImageNet 中，我們採用與 CIFAR10 相同的前處理方式，但在最後階段，將擾動乘以 0.5。接著，我們使用測試模型中的 VGG-16 來評估擾動的種類。由於 ImageNet 包含 1000 個不同的類別，從中選擇正確的類別十分困難。因此，模型評估擾動時，我們僅觀察實驗中使用 20 個類別的輸出值，並從中選擇最大值的類

doi:10.6342/NTU202301498

		基礎迭代攻擊法		CW 氏攻擊		深層懸弄攻擊	
資料集	評估模型	單模型	雜訊多模型	單模型	雜訊多模型	單模型	雜訊多模型
MNIST	自訓練模型	18%	57%	46%	44%	17%	68%
CIFAR10	PyramidNet	9%	68%	7%	55%	11%	19%
ImageNet	VGG-16	5.5%	56.0%	4.5%	38.0%	5.0%	38.0%

表 4.3: 評估模型的正確率: 表格中第 (i,j) 元素意味著，在第 i 種設定下，評估模型判斷 j 方法產生擾動的正確率。

別作為判定結果。

表4.3列出模型判斷擾動的正確率。在 ImageNet 中、雜訊多模型的設定下，我們可以觀察到評估模型能夠正確的判斷 56% 擾動的類別，而在單模型設定下，評估模型只有 5% 的正確率，相當於隨機猜測。在 MNIST 與 CIFAR10 中，我們也觀察到類似的結果。模型的正確判讀證實多雜訊模型產生的擾動確實含有明顯的可識別資訊。

4.8.2.2 人類評估

我們進一步進行人類評估實驗，由於實驗費用較高，因此我們僅評估 ImageNet 資料集、雜訊多模型設定下，基礎迭代攻擊法所產生的擾動。

我們將 200 個擾動隨機均分成 4 組，每組請 12 位受試者來進行測試。每位受試者負責決定 50 張擾動的類別，每張擾動有 20 個類別可以選擇，這些類別為實驗中所使用的圖片類別。

在排除每組最低和最高的分類正確率後，我們求得平均正確率為 80.7%。由於隨機猜測僅能產生 5%的正確率，因此，此實驗結果再次驗證雜訊多模型設定下，擾動中確實含有明顯的人類可識別資訊。

進一步分析受試者答錯題目的原因時，我們發現原圖中標籤類別所對應的物件有時不夠明顯，導致難以單從原圖判斷圖片類別，因此更難從生成的擾動中判

doi:10.6342/NTU202301498

斷原圖類別。請見圖4.5，圖中是一張計程車的照片，然而單從圖片我們很難知道圖片應該被歸類到哪一個類別中。在測驗的 200 張圖片中，約有 16 張圖片有著類似於圖4.5的問題，其圖片類別不容易被辨識。

圖 4.5: 難以判斷類別的圖片。

4.8.3 評估攻擊強度

為了驗證對抗性擾動中可識別資訊是混淆類神經網路的關鍵因子，我們要進一步評估生成擾動的攻擊能力。

在評估擾動的攻擊能力前，我們仍要先對擾動進行前處理，這是因為擾動的攻擊能力會受到範數值影響。因此，為了公平比較擾動的攻擊能力，我們需要統一範數大小。我們的處理的方式是，取擾動 δ 的正負號，再將正負號乘以一個量值 ϵ，成為新的擾動 δ'，請見以下公式。

$$\delta' = \epsilon \cdot sign(\delta) \tag{4.3}$$

如此，每個擾動都具備有相同的 L_{inf} 和 L_2 範數。這種處理擾動的方法與快速梯度正負問攻擊法類似 [12]。我們將對同一個資料集設定相同的 ϵ 值：對於 MNIST，我們設定 ϵ 值為 0.2；對於 CIFAR10，ϵ 值為 0.03；而 ImageNet 的 ϵ 值為 0.02。接下來，我們將使用公式4.3處理的擾動加到輸入圖片中，然後使用測試模

doi:10.6342/NTU202301498

型來計算對抗性實例的正確率。

表4.4記錄了測試模型對不同資料集和攻擊演算法產生對抗性實例的正確率。表格中的元素 (i,j) 表示測試模型 i 在攻擊算法 j 產生擾動的干擾下之正確率。在表4.4中 (b) 和 (c) 的子表格中，單模型設定下，源模型和測試模型使用相同的 ResNet，即所謂的白盒攻擊。為確保公平性，在計算單模型的平均正確率時，從 ResNet 獲得的數值將被排除在計算外。

我們可以發現在 ImageNet 資料集中，測試模型平均能正確分類 81.8% 的輸入圖片。為了凸顯擾動的攻擊能力，我們將雜訊添加至輸入圖片中做為比對。具體而言，雜訊為隨機將像素值設為 $+\epsilon$ 或 $-\epsilon$。我們發現，在 ImageNet 中，加入雜訊並不會影響測試模型的判斷。然而，在單模型設定下，基礎迭代攻擊法產生的擾動會使得測試模型的平均正確率下降至 63.3%。此外，從雜訊多模型生成的擾動會進一步使正確率下降至 13.2%，類似的結果也可以在 CIFAR10 和 MNIST 資料集觀察到。

圖4.6展示了由公式4.3生成的對抗性實例。圖中左列是輸入圖片、右列是對抗性實例，在雜訊多模型設定下，我們添加了基礎迭代攻擊法所產生的擾動至圖片中。在 MNIST 資料集中，測試模型會將對抗性實例錯誤地判定為 8；在 CIFAR10 資料集中，模型會將對抗性實例錯誤地判定為船；而在 ImageNet 中，模型會將計程車誤判為救護車。相比之下，對於 MNIST 資料集，我們需要添加更明顯的擾動才能使模型判斷錯誤，這可能是因為輸入資料的維度較低的原因。

4.8.4 遮掩效應的影響

在此研究中，我們進一步探討遮掩效應對於模型識別的影響。根據先前的定義，擾動的作用之一在於遮掩輸入資料中的可識別資訊，由於可識別資訊是人類

doi:10.6342/NTU202301498

MNIST 資料集	參照資料			單模型		雜訊多模型		
測試模型	輸入圖片	雜訊	基礎迭代	CW 氏	深層懸弄	基礎迭代	CW 氏	深層懸弄
測試模型	100%	88%	3%	56%	15%	0%	3%	0%

(a) MNIST 資料集

CIFAR10 資料集	參照資料			單模型		雜訊多模型		
測試模型	輸入圖片	雜訊	基礎迭代	CW 氏	深層懸弄	基礎迭代	CW 氏	深層懸弄
DenseNet-40	92%	77%	33%	52%	44%	13%	9%	34%
Dia-ResNet-164	96%	90%	51%	72%	68%	31%	22%	56%
PyramidNet-110	94%	79%	33%	53%	57%	18%	9%	39%
ResNet-56	92%	80%	0%	14%	21%	21%	9%	40%
平均正確率	94%	82%	39%	59%	56%	21%	12%	42%

(b) CIFAR10 資料集

ImageNet 資料集	參照資料			單模型		雜訊多模型		
測試模型	輸入圖片	雜訊	基礎迭代	CW 氏	深層懸弄	基礎迭代	CW 氏	深層懸弄
BN-Inception	81.5%	81.5%	64.0%	77.0%	68.0%	16.5%	22.0%	15.0%
DenseNet-121	83.5%	83.5%	58.5%	77.5%	66.5%	10.5%	16.5%	13.0%
VGG-16	79.0%	79.0%	67.5%	76.5%	70.5%	12.5%	20.5%	17.5%
ResNet-50	83.0%	83.0%	0.0%	7.0%	4.5%	13.0%	18.5%	14.0%
平均正確率	81.8%	81.8%	63.3%	77.0%	68.3%	13.2%	19.7%	15.2%

(c) ImageNet 資料集

表 4.4: 無特定目標攻擊下產生的擾動攻擊強度。

doi:10.6342/NTU202301498

圖 4.6: 輸入圖片與對抗性實例。

用來分類輸入資料的依據，因此，我們無法精準的定義可識別資訊，而一個較為廣泛的定義是，將圖片中物件輪廓內的區域視為可識別資訊 (簡稱輪廓)，而其他部分則視為非可識別資訊 (簡稱背景)。因為人類判斷物件類別的依據必定位於輪廓內。如果我們關於擾動降低可識別資訊導致模型辨識錯誤的假設為真，則輪廓內的擾動相較於背景處的擾動會具備更強的攻擊能力。以下，我們進行實驗來驗證此假設。

我們採用了具有物件位置的 ImageNet-S 資料集 [67] 進行了實驗。然而，該資料集沒有包含實驗所需的所有圖片，且少部分圖片的標註有一些誤差。因此，我們下載了標註軟體 [68]，並重新手動標注 16 張圖片的位置。經過處理後的擾動如圖4.7所示，其中左邊的圖片為原始的擾動，而右邊的圖片為提取輪廓的擾動。

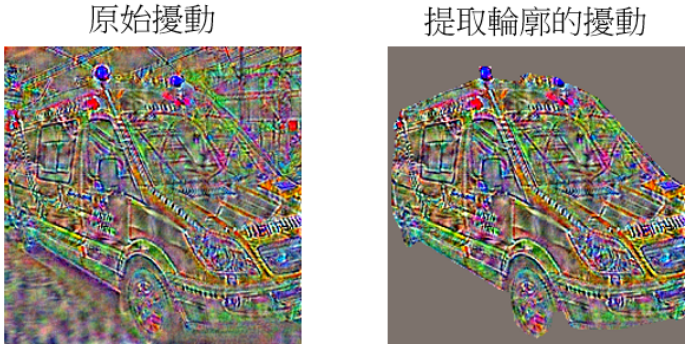


圖 4.7: 原始擾動與提取輪廓後的擾動。

為了探究人類可識別資訊是否成為導致模型辨識錯誤的主要原因，我們根據公式4.3對擾動進行處理。接著，我們分別提取其輪廓和背景，將它們添加至原圖

doi:10.6342/NTU202301498

上，並分別送入至測試模型評估擾動對攻擊效力的影響。實驗結果表明，在雜訊多模型設定下，基礎迭代攻擊法、CW 氏攻擊、深層懸弄等攻擊算法生成的擾動輪廓使得四個測試模型的平均正確率分別降低至 32.3%、37.0%、33.5%，相比之下，僅憑背景部分則使模型的平均正確率分別下降至 62.1%、69.0%、60.6%。這一結果驗證了遮掩效應在模型辨識錯誤中的主導地位。值得注意的是，此時輪廓與背景在圖片中所占的面積比為 0.83，意味著即使擾動的輪廓部分面積相對較小，其攻擊能力仍顯著強大。

我們進一步的探討可識別資訊對於擾動攻擊能力的影響。利用 OpenCV 套件 [69]，我們選擇 3x3 的全一矩陣作為卷積核，對擾動的輪廓與背景進行膨脹 (Dilation) 或侵蝕 (Erosion) 處理來更改擾動所占面積，然後將其送入四個測試模型以計算平均辨識正確率。我們繪製了擾動所占圖片面積比例與模型正確率的關係圖，如圖4.8所示。左圖展示了多模型設定下的結果，右圖則為雜訊多模型設定下的結果。兩者均呈現相似趨勢：在相同的擾動面積下，擾動之輪廓的攻擊能力明顯優於背景，這表明遮掩效應而非背景雜訊是導致模型辨識錯誤的主要原因。

doi:10.6342/NTU202301498

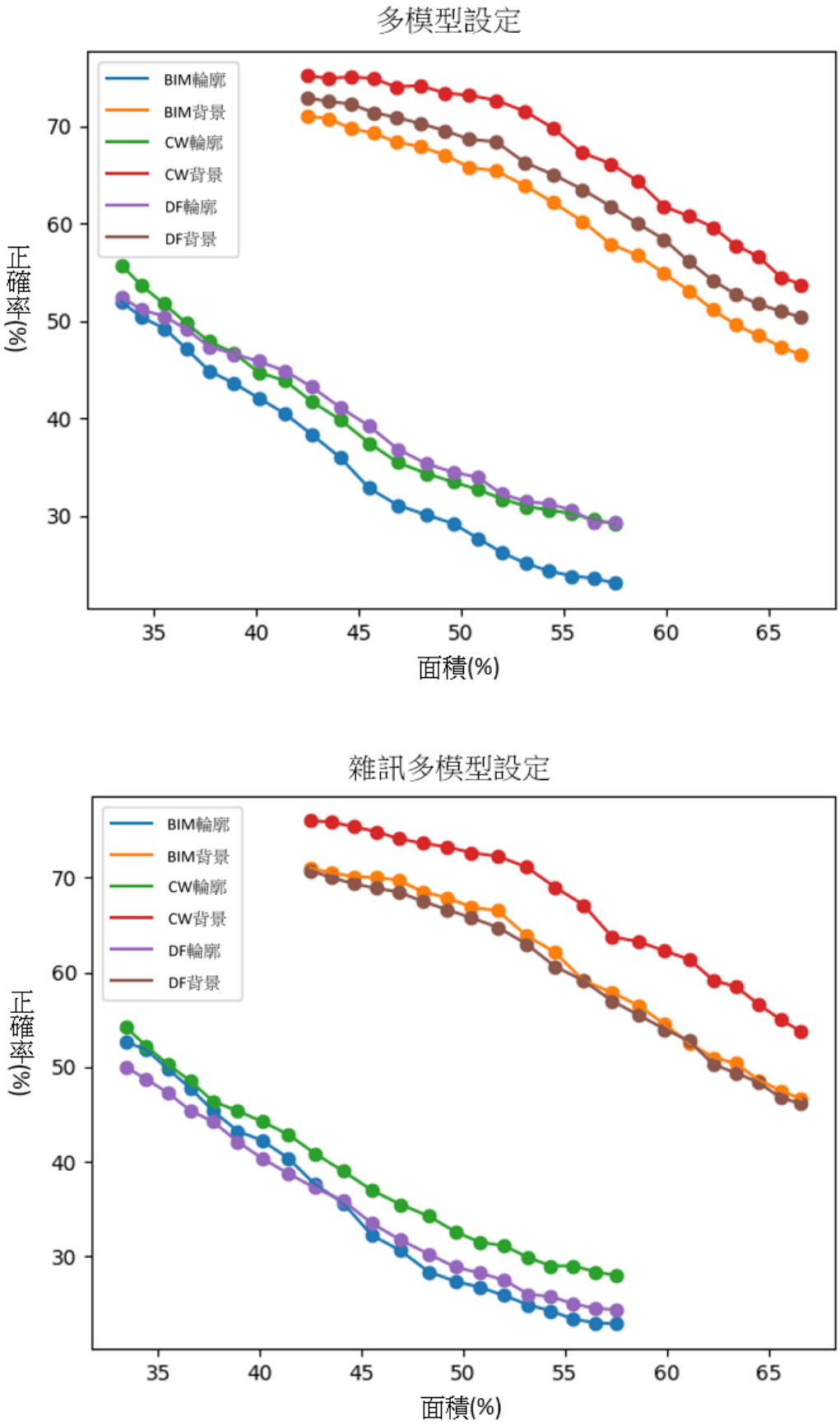


圖 4.8: 擾動面積與模型辨識正確率之關係圖。

doi:10.6342/NTU202301498

我們計算了正確率相對於擾動面積的微分值，並將結果呈現於圖4.9。在該圖中，灰色虛線所表示的面積反映了擾動輪廓在整張圖片中所佔的比例。此外，圖中的各點代表當擾動面積增加 1% 時，模型的平均識別率將下降多少個百分點。

進一步觀察圖4.9，我們可以將其區分為三個部分：輪廓邊界內部（灰虛線左側）、輪廓附近（與灰虛線相鄰）以及背景區域（灰虛線右側）。從圖中可以看出，在輪廓邊界內部，微分值呈現一個高峰，這表明遮掩效應對模型識別具有顯著影響。在輪廓附近區域，微分值出現了第二個高峰。我們推測這是因為該區域靠近圖片邊緣，而圖片邊緣對卷積神經網路的判斷具有關鍵性影響，因此，在這部分添加擾動同樣會對模型的識別造成較大的影響。

相對而言，在背景區域，微分值較小，這意味著該部分的攻擊效力趨於飽和。綜上所述，我們認為本實驗結果再次證實了遮掩效應對模型判斷錯誤具有重要影響。

doi:10.6342/NTU202301498

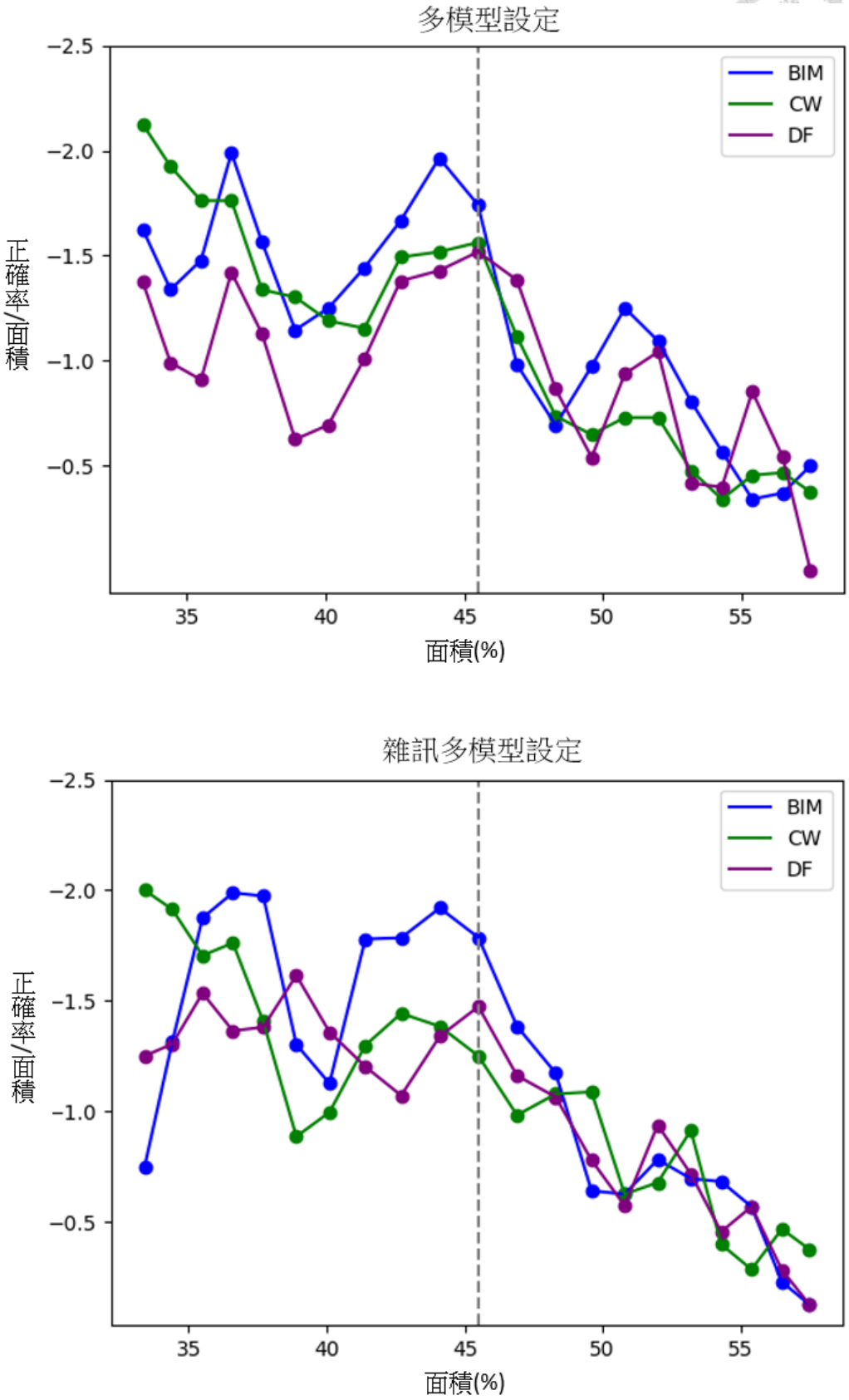
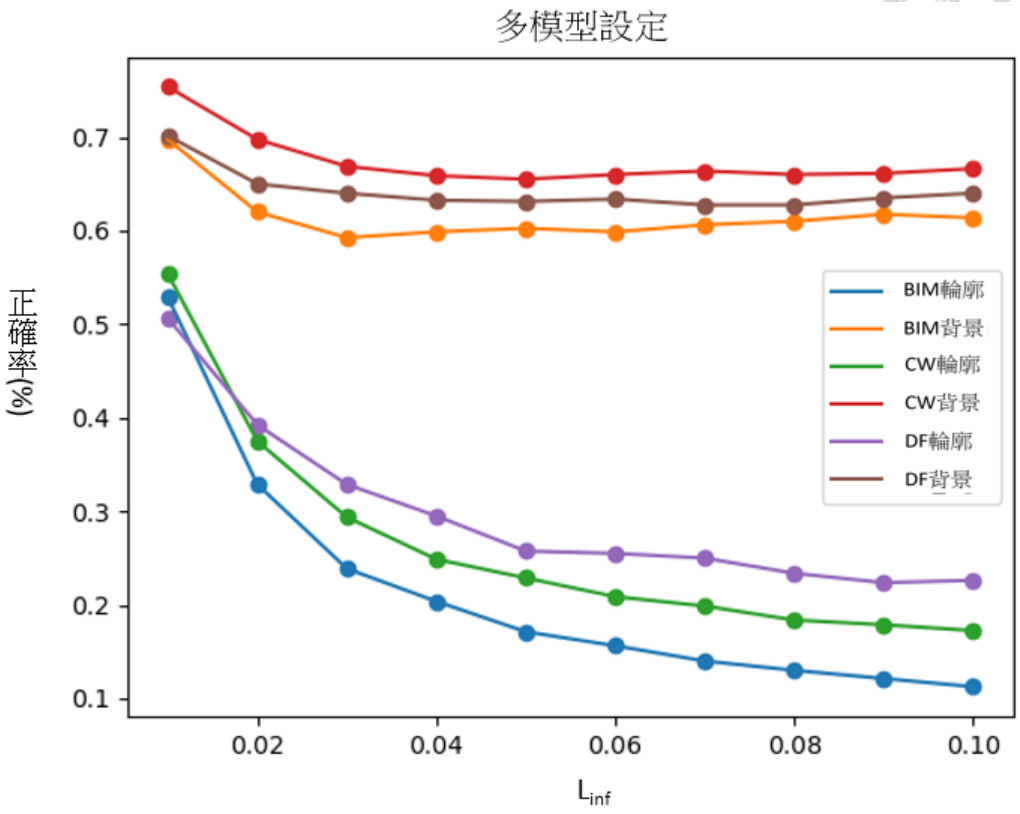


圖 4.9: 擾動面積微分值與模型辨識正確率之關係圖。

doi:10.6342/NTU202301498

我們接著就擾動的輪廓與背景調整 Linf 範數從 0.01 至 0.1，每 0.01 為一間隔，並觀測測試模型平均正確率的變化。我們發現隨著範數的增加，背景的擾動所造成模型正確率下降，但很快就飽和了，並且測試模型的正確率仍高達於 0.6-0.7。反觀輪廓的部分，並不容易飽和，並且可以進一步使模型的正確率降至 0.1-0.2，請見圖4.10，這再次顯示了背景與輪廓本質上的不同。

doi:10.6342/NTU202301498



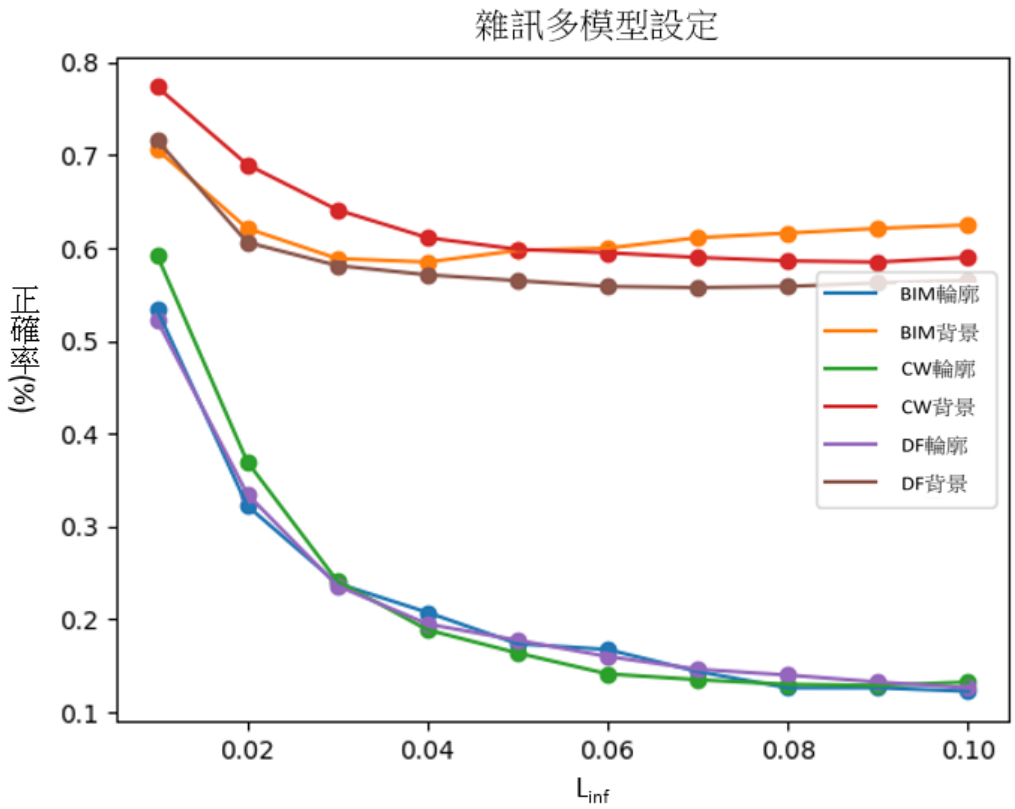


圖 4.10: 擾動 L_{inf} 範數與模型辨識正確率之關係圖。

doi:10.6342/NTU202301498

我們驗證了擾動中輪廓的部分較背景有更強的攻擊效力，這與我們所提出的遮掩效應為致使模型辨識錯誤的推論相合。

最終，我們將此節的研究成果彙整成圖4.11。我們發現，加總多個針對同一張圖片產生的對抗性擾動後，再去除背景雜訊，擾動中的人類可識別資訊就會變得清晰可見。稍微減去原圖片中的樹蛙資訊後，測試模型便會錯誤地辨別圖片。這些證據皆指向擾動的本質是人類可識別資訊。

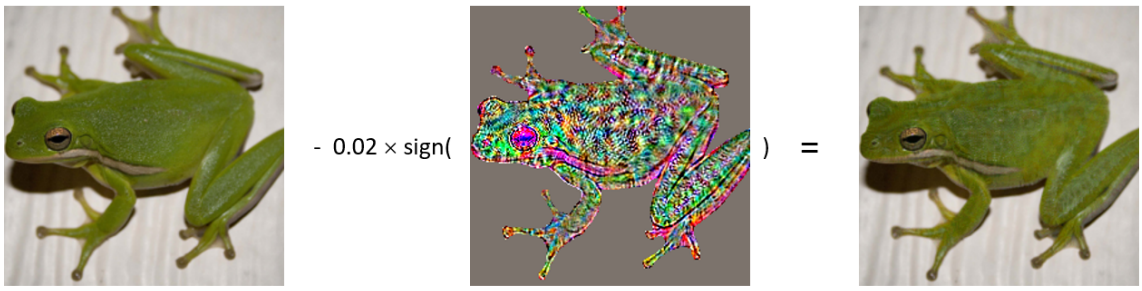


圖 4.11: 擾動中的可識別資訊會導致模型判斷錯誤。

4.8.5 有特定目標攻擊

在有特定目標攻擊的模式下，生成效應會更加明顯。有特定目標攻擊的實驗設定與無特定目標攻擊的設定相同，但在 ImageNet 資料集上，我們使用了不同類別的圖片。這是因為生成資訊的明顯程度與攻擊類別有很大的關係。如果我們將毫不相干的類別進行有特定目標攻擊，如汽車變成蛇，不同模型會產生相異的可識別資訊，因為將汽車轉變成蛇的方法並不唯一。這會導致當我們平均不同模型產生的擾動時，這些資訊互相抵消，最終無法得出明確的結果。然而，如果我們將母雞變成公雞，或將蛞蝓變成蝸牛，需要添加的資訊則趨於固定，因為母雞與公雞的差異主要為雞冠；蛞蝓有了殼就變成了蝸牛。因此，不同模型產生的答案也趨於一致。最終，平均擾動後的結果也較容易觀察到。

圖4.12[65] 展示了有特定目標攻擊演算法生成的對抗性實例。為了進一步突

doi:10.6342/NTU202301498

顯其效果，我們對擾動乘以一常數以進行縮放，使得擾動中的最大值為 0.5。在第一行圖片中，輸入圖片是暹羅貓，而攻擊目標為老虎。在多雜訊模型的設定下，我們觀察到貓毛的顏色轉變為橙色，黑色條紋更加明顯，眼睛的顏色也從藍色變為橙色。這些變化與老虎的特徵相對應。第二行圖片中，擾動加強了母雞身上雞冠的顏色，使雞冠變大、變紅，我們也可以看到雞頭下方出現一塊綠色的色塊，同時雞身上的羽毛也變得更加斑斕，這些是公雞的特徵。第三行圖片我們可以觀察到對抗性擾動在蜷起的蛞蝓上添加了螺旋狀紋路，使其更加像蝸牛殼。這些例子都顯示出擾動具備人類可識別資訊。

生成效應所產生的可識別資訊並不像遮掩效應那麼明顯，而且，不是每張圖片都可以觀察到這個現象。正如前述所敘，我們推測這是因為生成並沒有標準的答案，而遮蓋原圖中具鑑別度的特徵則有標準答案，因此產生的可識別資訊會更明顯。生成效應所產生的可識別資訊並不明顯，也與有特定目標攻擊的目標類別不容易轉移至不同模型相互呼應 [39]。

4.9 本章總結

本節發現由公式4.1所產生的擾動具備人類可識別資訊。進一步的研究顯示，這種資訊可以分為兩類，一類為常見於無特定目標攻擊的遮掩效應，另一類為常見於有特定目標攻擊的生成效應。這項發現有助於解釋對抗性擾動為什麼會導致模型判斷錯誤。同時，這項發現暗示著類神經網路的判斷與人類相似。

doi:10.6342/NTU202301498

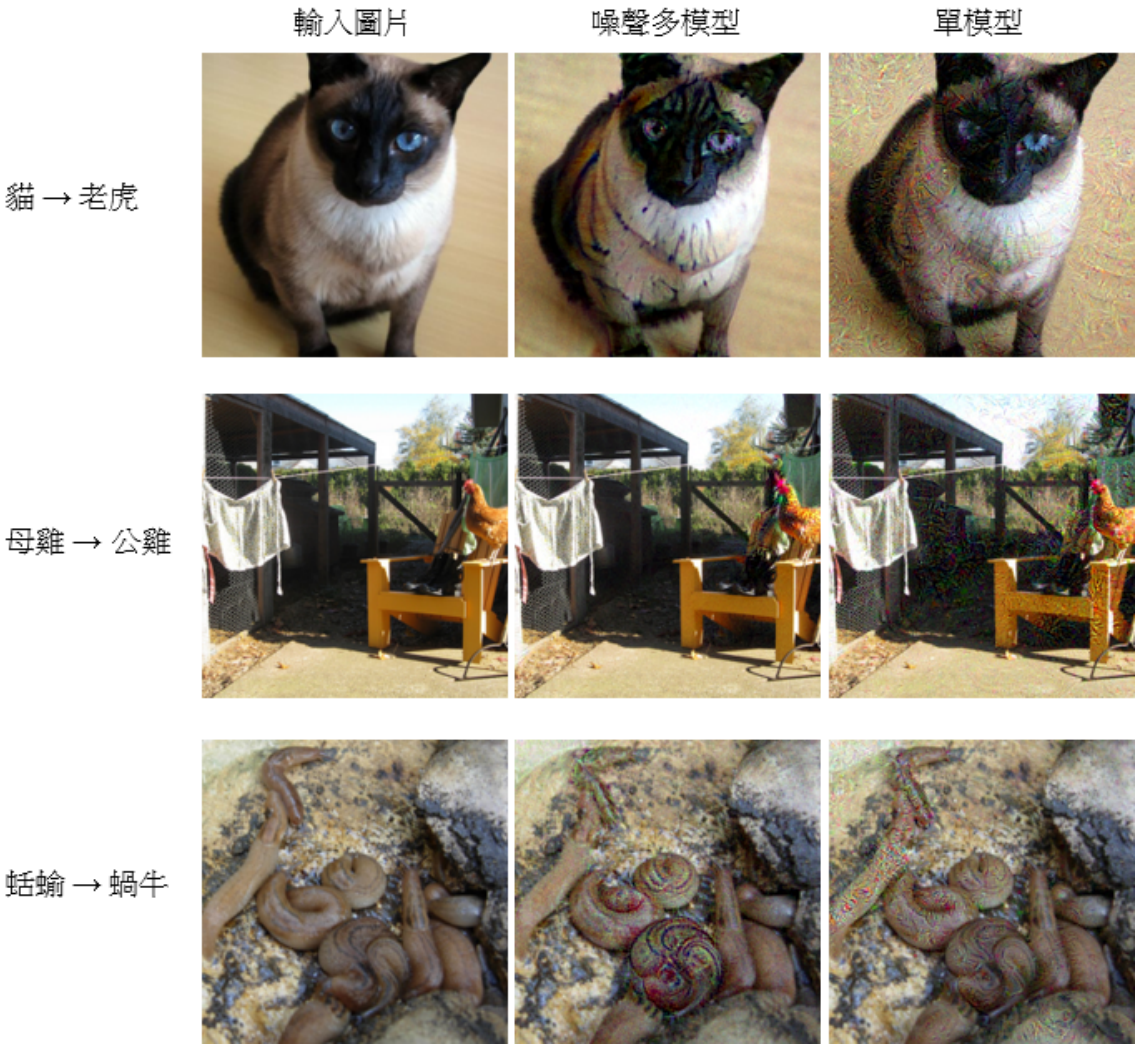


圖 4.12: 有特定目標攻擊所產生的對抗性實例。

doi:10.6342/NTU202301498

doi:10.6342/NTU202301498

第五章　實驗觀察

在前述實驗中，公式4.1將成千上萬的擾動累加在一起。這就像使用高倍率的顯微鏡深入探究微觀世界，使我們能夠更加深入地瞭解類神經網路的運作原理。本章節的討論重點將集中在實驗過程中觀察到的一些引人入勝的現象。

5.1　搜尋式攻擊

在本節中，我們將探究隨機搜尋式攻擊演算法所生成擾動的遮掩效應。為此，我們選擇使用方形攻擊進行實驗。與前述實驗採用的反向傳播演算法不同，在這個攻擊演算法中，我們通過迭代隨機搜尋來進行優化對抗性擾動。在實驗中，我們選擇了以 L_{inf} 範數為基礎的方形攻擊方法。為了獲得更純粹的實驗結果，我們將擾動的初始值從原先的條狀紋路改為 0。這一改變主要目的是希望遮掩效應導致模型辨識錯誤，而非源於條狀紋路的影響。在圖5.1中，我們展示了在單模型設定下所產生的擾動與其對應圖片。

值得注意的是，隨機搜尋在優化擾動方面的效果遠不如利用模型梯度。因此，為了生成單張圖片的對抗性擾動，我們需要將大約 2.7 萬張擾動疊加在一起。這意味著使用一顆 RTX3090 顯示卡生成一張完整的擾動需要花費大約 5 個小時。受到預算和資源限制，我們僅製作了 40 張圖片所對應的對抗性擾動。

doi:10.6342/NTU202301498

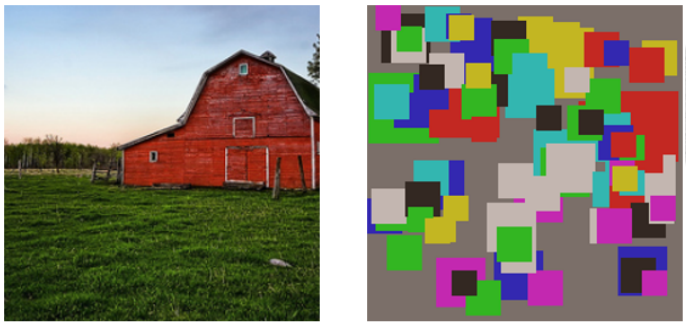


圖 5.1: 單模型設定下，方形攻擊產生的擾動與對應的圖片。

實驗結果顯示，只要疊加足夠多的隨機搜尋生成的擾動，這些擾動仍然會呈現出人類可以識別的信息。如圖5.2所示，生成的擾動與原始圖片具有一定的相似性。這一現象進一步支持了擾動中存在人類可識別信息的觀點，因為即使是隨機搜尋所產生的結果，我們仍然可以觀察到遮掩效應。然而，我們必須指出，並非

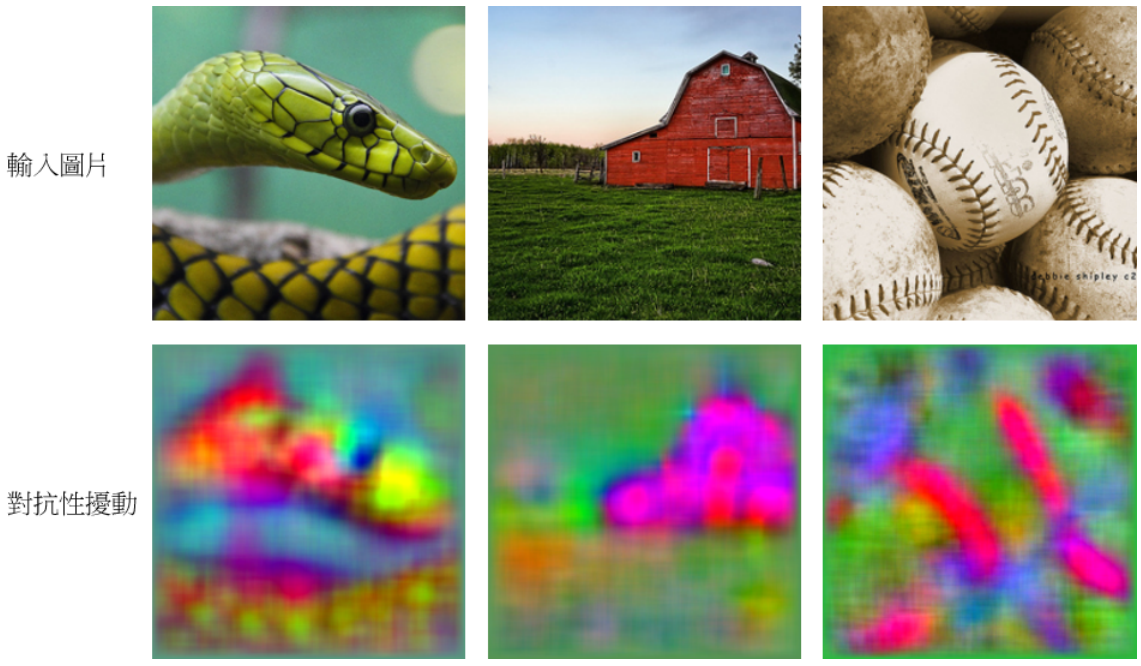


圖 5.2: 方形攻擊產生的擾動具備遮掩效應。

所有方形攻擊所生成的擾動都能顯示出顯著的遮掩效應。在某些情況下，例如當輸入圖片中物件形狀過於複雜，或者模型主要依賴物體邊緣進行分類時，這時生成的擾動可能不會具有明顯的遮掩效應。畢竟，要通過大量疊加的方形區塊來強調圖像邊緣，實際上需要相當多的擾動才有可能達到這一目的。圖5.3便展示了兩

doi:10.6342/NTU202301498

例具有較不明顯遮掩效應的對抗示例，我們可以看到樹蛙所對應的擾動輪廓並不清楚，而鯊魚對應的擾動僅有頭部與魚鰭較為明顯，其餘部分接模糊不清。

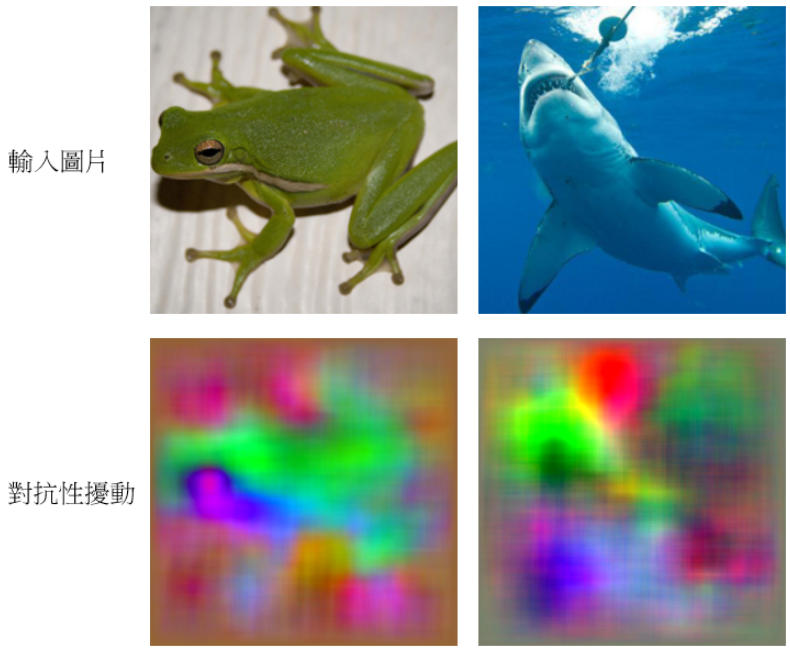


圖 5.3: 方形攻擊產生的擾動不具備明顯遮掩效應的例子。

5.2 雜訊的影響

本節將探討在公式4.1中添加高斯雜訊對生成擾動的影響。由於生成擾動的過程中我們向原始圖像中添加了高斯雜訊，我們進一步研究在不添加雜訊的情況下，人類可識別資訊是否仍然存在。接下來，在單模型的設定下，我們調整高斯雜訊的標準差，並觀察到生成的擾動會出現棋盤效應（Checkerboard Artifacts）[70]，同時生成擾動的細節資訊也會受到影響。基於這些觀察結果，我們可以總結出高斯雜訊對生成擾動的影響。

doi:10.6342/NTU202301498

5.2.1 移除雜訊

為了證實人類可識別資訊的確存在於對抗性擾動中，而非在產生擾動的過程中添加的高斯雜訊所致，我們在不添加雜訊的情況下，直接加總多個模型生成的擾動，並觀察可識別資訊是否依然存在。受限於源模型的數量，此實驗僅在 ImageNet 資料集上進行。

通過觀察圖像，我們可以看到不同演算法產生的擾動都具有若干遮掩效應，如圖5.4所示。儘管在多模型設定下，生成的擾動不如雜訊多模型設定下的清晰，但是加總擾動的數量也只有雜訊多模型設定下的十分之一。此外，圖4.8也顯示，多模型設定下，提取輪廓的擾動為導致模型辨識錯誤的關鍵因素。這說明人類可識別資訊源於擾動本身，而非添加的雜訊所導致。

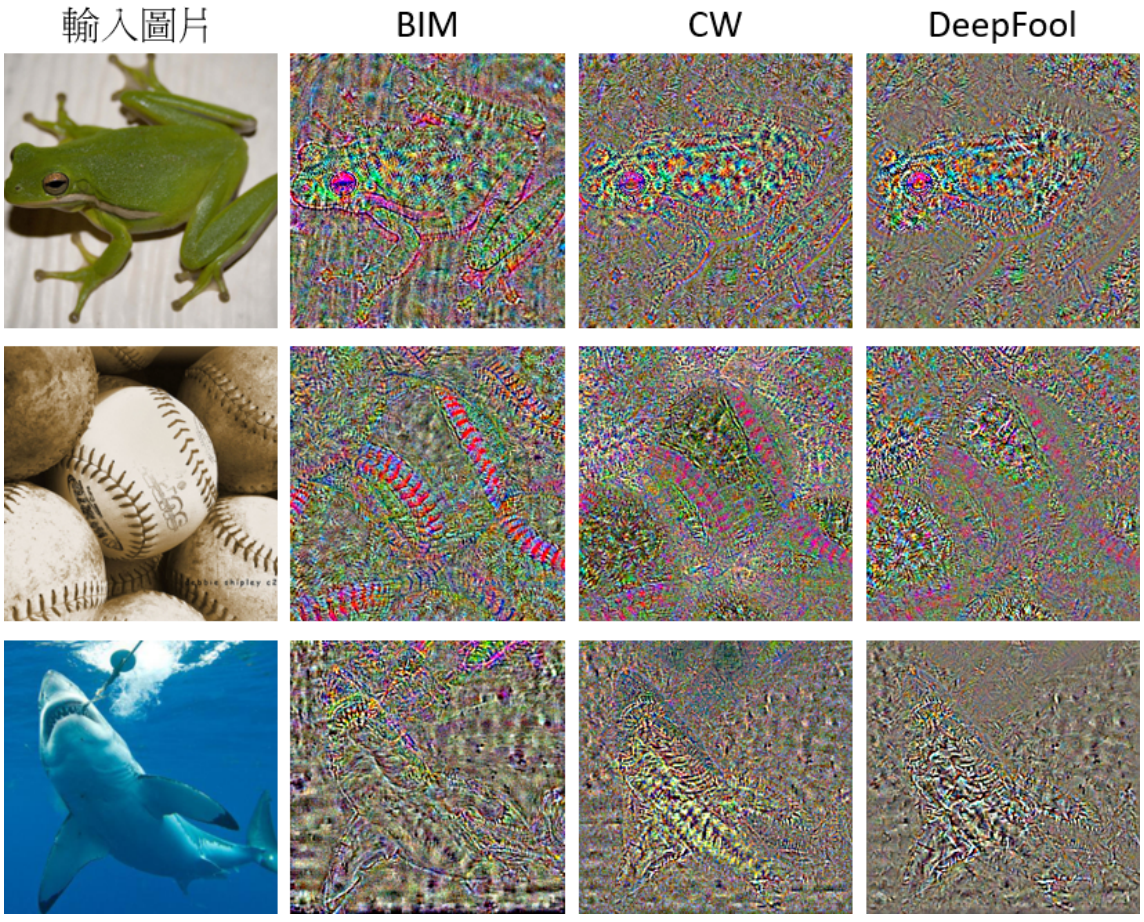


圖 5.4: 多模型設定 (不添加雜訊) 下產生的擾動。

doi:10.6342/NTU202301498

5.2.2 探討標準差的影響

此節我們探討公式4.1添加雜訊對生成擾動的影響。為了簡化問題，我們觀察在單模型設定下，雜訊對生成擾動的影響。實驗過程中，我們對同一張圖片添加1000 種不同的高斯雜訊，然後平均這 1000 個生成的擾動，這種實驗設定被稱為雜訊單模型設定。在實驗中，添加的雜訊平均值為 0。當雜訊的標準差接近 0 時，雜訊對生成擾動的影響可忽略不計。然而，當我們逐步將高斯雜訊的標準差增至 0.5 時，生成的擾動將受到雜訊的嚴重干擾。透過觀察變化趨勢，我們能更深入地瞭解雜訊對擾動的影響。

5.2.3 格狀條紋的顯現

隨著雜訊的標準差逐漸增加，我們觀察到生成的擾動中出現了明顯的格狀條紋。經過查閱相關文獻，我們發現這一現象已經被其他學者研究過 [70]。這是因為在卷積神經網路中，卷積核與圖片進行卷積運算時，會重複計算圖片中特定的像素，導致相應部分的梯度值被放大，從而產生明顯的格狀條紋。

然而，我們發現隨著雜訊標準差的增加，格狀條紋的周期也會相應變大。如圖5.5所示，圖片下方的數字表示添加雜訊的標準差。當標準差為 0.1 時，擾動中出現的條紋周期為 1 像素。然而，當我們將雜訊標準差調整至 0.5 時，可以觀察到排列整齊的格狀條紋，其周期約為 30 像素（一行重複 7 次）。

格狀條紋的周期與重複運算的卷積核所處層數密切相關。這意味著在雜訊標準差大的情況下，格狀條紋主要受到靠近輸出層的卷積核影響；而在雜訊標準差較小的情況下，格狀條紋則主要受到靠近輸入層的卷積核所影響。然而，這一現象背後的機制仍有待進一步的探討和研究。

doi:10.6342/NTU202301498

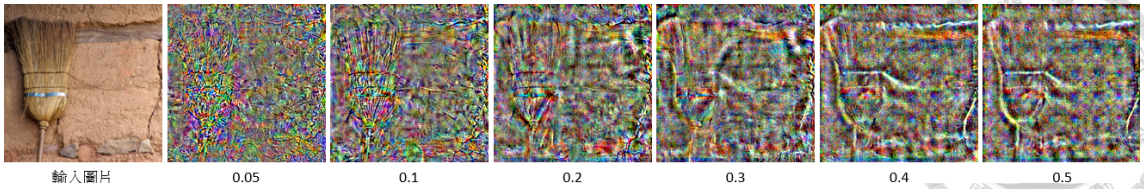


圖 5.5: 高斯雜訊的標準差對於生成擾動的影響。

5.2.4 細節資訊的消失

添加雜訊會破壞生成擾動的細節資訊。圖5.5中，我們可以看到當標準差為0.05 時，生成的擾動具有清晰且銳利的線條，呈現類似掃把鬃毛般的紋理，這些紋理的走勢並不完全與原圖一致。當我們將標準差提升至 0.5 時，我們只能觀察到掃帚的大致輪廓，圖片中的細節資訊已經消失。我們推測這是因為大標準差的雜訊會掩蓋圖片中的微小細節，導致生成的擾動失去這些資訊，最終僅保留最為顯著的輪廓。

綜合實驗結果，我們發現雜訊對擾動的影響主要體現在兩個方面：首先，它會增加擾動中格狀條紋的週期；其次，它會導致擾動的細節資訊消失，僅剩下大致輪廓。這些發現並不影響擾動中具備人類可識別資訊的結論。

5.3 擾動的特性

由於公式4.1能夠有效地還原擾動中的可識別資訊，因此我們能夠更進一步觀測擾動的本質。以下是我們觀測到一些擾動有趣的性質。

5.3.1 互補性

我們發現在無特定目標攻擊模式下，擾動的遮掩效應可能以一種互補的形式出現，即圖片的輪廓並未被擾動遮掩，但是圖片的背景部分卻受到擾動的覆蓋。

doi:10.6342/NTU202301498

我們將這一現象稱為「擾動的互補性」，如圖5.6所示。圖中由左至右分別為輸入圖片（雛菊）、由 CW 氏攻擊演算法在多模型以及雜訊多模型設定下產生的擾動。為了突顯擾動的互補性，我們將擾動值小於其平均 L_1 範數的像素值設為 0 (圖中灰色部分)，從圖中我們可以觀察到，花瓣部分的擾動值較小，而背景部分的數值較大。

我們推測此現象背後的原因為降低圖片中可識別資訊與提升背景像素值具有相似的效果，即降低模型輸出的標籤分數和提升其他類別的分數都可能導致模型辨識出錯誤。因此，我們可以在擾動中觀察到這種互補的特性。

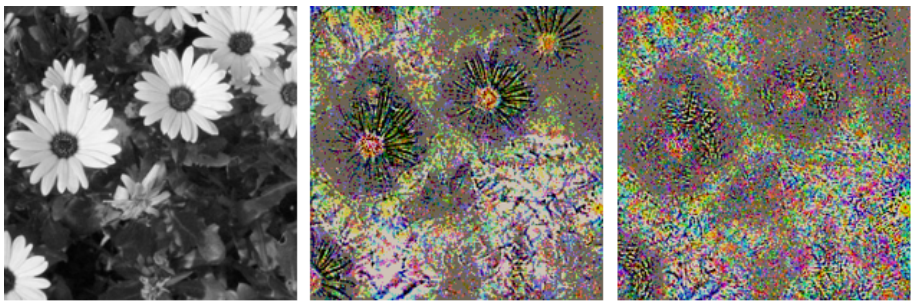


圖 5.6: 對抗性擾動的互補性。

5.3.2 比較生成的擾動

我們比較了不同攻擊演算法所產生的對抗性擾動，並發現生成的擾動間具有高相似性，詳見圖2.1。在提取擾動的輪廓後，我們進一步計算了擾動間的餘弦相似度。出乎意料地，在雜訊多模型設定下，基礎迭代攻擊法、CW 氏與深層懸弄攻擊演算法產生擾動的餘弦相似度約為 0.5，詳見表5.7。

此發現有著更為深遠的啟示：假設不同攻擊演算法產生的擾動經過多次累加後，最終均收斂至相當接近的擾動，暗示著存在某種轉換可以直接將輸入圖片對應至擾動。

doi:10.6342/NTU202301498

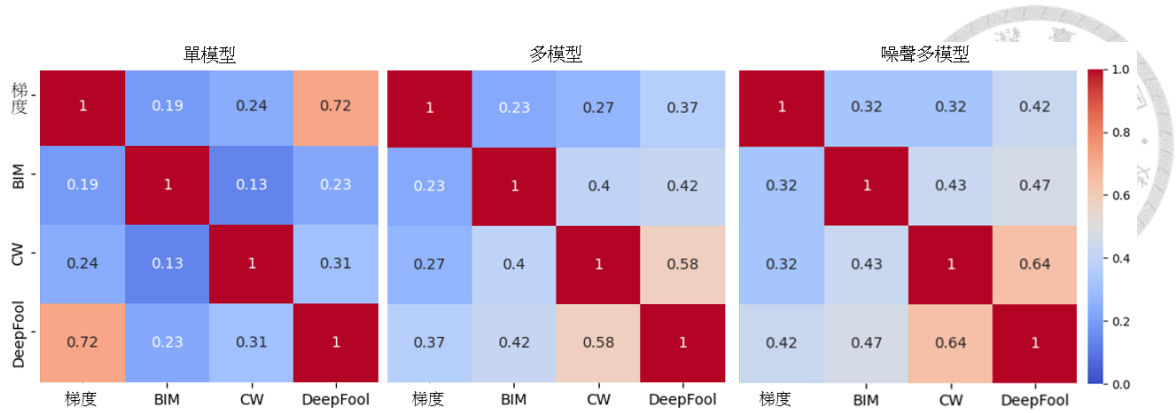


圖 5.7: 比較不同攻擊演算法產生擾動的餘弦相似度。

儘管這些擾動具有一定程度的相似性，它們彼此間仍存在一些差異。我們推測這可能是因為在生成擾動的過程中，有些演算法的目標函數會限制擾動的範數，導致最終生成擾動彼此間的差異，其他導致差異的原因，仍需進一步研究。

我們觀察到，在單模型設定下，所獲得的餘弦相似度顯著低於多雜訊模型設定，與我們早先的假設相吻合。這主要是由於在同一模型下，不同演算法產生的擾動因含有大量隨機性高的雜訊而相似度低。但在多雜訊模型設定下，這些擾動的雜訊被有效地消除，從而提升了相似度。

在單模型設定下，模型梯度與由深層懸弄攻擊演算法產生的擾動展現出高度的餘弦相似度。這可能源於深層懸弄演算法在多數情況下只需少量迭代便能使模型出錯，從而導致所產生的擾動與模型梯度高度相似。

由於模型梯度與另外兩種攻擊演算法之間的餘弦相似度較低，進一步比較梯度與其他攻擊算法產生的擾動後，我們發現梯度的可識別性較弱，請見圖5.8。原始圖片、模型梯度、及 CW 氏攻擊結果依序展示在左、中、右圖。相較於梯度，CW 氏攻擊產生的擾動更專注於熊貓輪廓，並且具有較低的雜訊強度與更清晰的輪廓，這種現象在實驗資料集中廣泛存在。

我們推斷，相較於單次迭代的梯度，多次迭代使得攻擊演算法在優化可識別資訊上更為出色。此外，CW 氏攻擊在優化擾動時，亦會降低擾動範數，進一步

doi:10.6342/NTU202301498

排除與目標物件無關的訊息。這個研究結果暗示，多次迭代的對抗性擾動更能揭示模型的辨識依據。

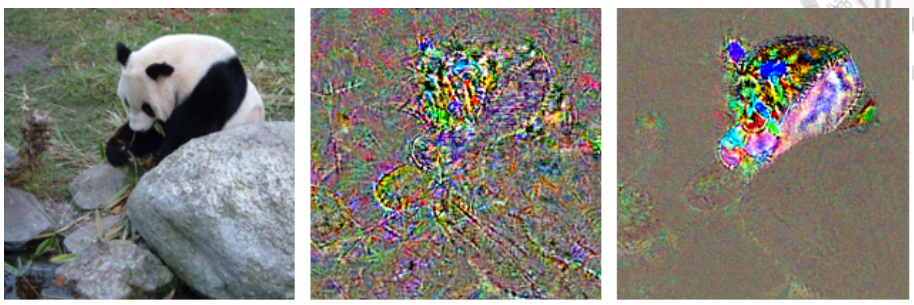


圖 5.8: 比較梯度與 CW 氏攻擊法所產生擾動的可識別性。

5.4 消失的貓

我們發現類神經網路會將雜訊分類為特定的類別，這個現象存在於不同的資料集和類神經網路架構中。以下我們使用 CIFAR10 和 ImageNet 資料集來呈現此現象。

CIFAR10：我們觀察到 CIFAR10 資料集中，貓的類別所產生的擾動並不具備清析的輪廓，請見圖5.9，但是測試模型仍然有 92.5% 的機率正確的將這些擾動分類為貓，高於其餘類別的平均正確率 63.0%。

圖 5.9: 針對輸入圖片為貓的類別所產生的對抗性擾動。

doi:10.6342/NTU202301498

為了深入探討這一現象，我們進行了一項實驗，首先隨機產生了 100 種不同的等方向性高斯雜訊 (Isotropic Gaussian Noise)，其維度與輸入圖片相同，同時雜訊每一維度之平均值和標準差與資料集內所有圖片像素值的平均值及標準差相同。接下來，我們將這些雜訊分別送入四個不同的測試模型中進行分類，以下是我們的實驗結果：

DenseNet40 貓：100%

DiaResNet164 貓：77%、鳥：23%

PyramidNet110 貓：95%、鳥：5%

ResNet56 鳥：100%

實驗數值代表有多少百分比之高斯雜訊被分類為貓或鳥，我們十分驚訝的發現，對於任意模型而言，100 種高斯雜訊均被分類成貓與鳥。我們推測這一現象是由於 CIFAR10 資料集中貓的型態多變而鳥類的品種多元，導致相對於其他類別，辨識貓和鳥的圖片具有更高的挑戰性。為了提高辨識正確率，類神經網路採取了一種策略，亦即專注於學習如何分類易識別的類別。將剩餘難以區分的類別統一劃分到“其他類別”。在 CIFAR10 資料集中，貓或鳥的類別即為這種“其他類別”。

ImageNet：我們進一步觀察訓練在 1000 個類別的 ImageNet 資料集上的模型是否仍能觀察到此現象。我們將 1000 個有著與資料集相同平均值與標準差的高斯雜訊送到測試模型進行分類。結果顯示 BN-Inception、DenseNet121、VGG16、ResNet50 的判斷結果分別高度集中於鏈甲（Chain Mail）、祈禱毯 (Prayer Rug)、門墊 (Door Mat)、紗窗（Window Screen）等類別，分類結果如下所示：

BN-Inception 鏈甲：99.3%、地毯：0.3%、門墊：0.2%、鏈子：0.2%

DenseNet121 地毯：93.6%、鏈甲：6.0%、線蟲：0.4%

VGG16 門墊：98.9%、沙洲：1.0%、洗碗布：0.1%

doi:10.6342/NTU202301498

ResNet50 紗窗: 78.8%、牛仔褲: 13.7%、網球: 3.8%、犰狳: 2.8%、高爾夫球: 0.8%、鵝: 0.1%

我們發現這些“其他類別”如祈禱毯、鎖子甲、門墊和窗簾都具備一些基本且明顯的幾何形狀，例如毯子一般為方形，而窗簾則具備格狀條紋。雖然在 ImageNet 與 CIFAR10 資料集上，雜訊皆被模型判定為特定類別，但是所對應類別的性質十分不同。

我們對於此現象的解釋是，類神經網路在判斷輸入資料時會先從一個“基本類別”開始，當模型從輸入資料獲得更多的資訊時，模型便會給予輸入資料更精確的判斷。這就好比決策樹 (Decision Tree) 的判斷流程，當輸入資料沒有提供額外資訊時，預測結果會停留在最初始的節點 (Node)，獲取更多輸入資料的訊息時，模型便能一步一步的細分輸入資料的類別。

所謂的“基本類別”就是我們觀察到的“其他類別”，而決定“基本類別”的方式取決於如何最大化模型的正確率，當資料集存在少數與眾不同且難以辨別的類別時，類神經網路會傾向將該類別定為基本類別，如此，可以降低誤判此類別的可能性。存在多種類別時，類神經網路會選擇一種常見的基本幾何圖形做為“基本類別”。

這一現象對我們的啟示是，在訓練類神經網路時，我們可以在原有的類別上新增一個“其他類別”，並在訓練過程中加入與資料集類別無關的資料，將其標記為“其他”。通過額外增加“其他類別”，可以促使模型在原先被默認為“其他類別”的類別上更加確實地學習相應資訊。在進行輸入資料類別判斷時，忽略“其他類別”的輸出值，這個方法可能有助於提高模型的正確率。

doi:10.6342/NTU202301498

5.4.1 本章總結

在本章節中，我們首先探究了搜尋式攻擊中的可識別資訊。接著，我們分析了添加雜訊對於生成擾動中可識別資訊的影響，結果顯示可識別資訊主要源於擾動本身，而非添加雜訊所引起。然後，我們比較了不同演算法產生擾動的餘弦相似度，發現在雜訊多模型架構下，擾動的餘弦相似度可高達 0.5。這一發現暗示我們可能存在某種函式可以直接將圖片轉換為擾動中的可識別資訊。此外，我們觀察到對抗性攻擊演算法相較於梯度更能揭示模型的判斷依據。最後，我們發現了一個名為「消失的貓」的奇特現象，通過這一現象，我們能夠更深入地了解模型的辨識機制。

doi:10.6342/NTU202301498

第六章　綜合討論

我們的研究揭示，對抗性擾動不僅含有人類可識別的信息，且可識別信息是引發類神經網路誤判的因素之一，這暗示了類神經網路與人類的判斷在某程度上的一致性。此發現為擾動相關現象提供了合理的解釋。

6.1　類神經網路的脆弱性

為了正確地辨識輸入圖像的類別，類神經網路在計算輸出值時，產生的梯度值在輪廓內的可識別資訊處會特別大 [30]。我們的研究顯示，擾動通常針對輪廓內的可識別資訊之像素進行微調，因此，類神經網路的梯度會放大擾動的影響，導致微小的擾動即足以讓類神經網路產生辨識錯誤。

6.2　擾動的可轉移性

學者發現，不同類神經網路可能受到相同的擾動干擾，從而對輸入資料產生誤判，這一現象被稱為擾動的可轉移性。我們的研究結果可以解釋此現象。

類神經網路在進行辨識時，會依據輸入資料中能夠用來區分類別的資訊，例如物體的輪廓進行辨識 [30]。然而，我們的研究結果顯示，擾動具有遮掩效應，也就是說，擾動的作用在於降低圖片中輪廓內可識別資訊的像素值，進而會降低

doi:10.6342/NTU202301498

類神經網路梯度與圖像內積的數值。然而，為了正確辨識輸入資料，不同類神經網路產生的梯度值在輪廓內的可識別資訊處會特別大，因此，只要對抗性擾動的作用是降低圖片中輪廓中可識別資訊的像素值，擾動便具有可轉移性。

6.3 模型的可解釋性

對抗式訓練會增進類神經網路產生擾動與梯度的可識別性，請見圖2.4。

我們的發現可以解釋此現象。因為，在圖像加入擾動時，輸出值對於輸入像素的梯度值大致維持不變 [12]。因此，訓練時將擾動加到輸入資料中最小化損失函數，其作用即為最小化輸出值對於輸入像素梯度值之 L_2 範數 [71]。

根據我們的假設，人眼無法輕易的從類神經網路的擾動與梯度值觀測到物體輪廓與特徵，是因為雜訊掩埋了這些訊息，同時這些訊息往往是殘缺的。

然而，最小化梯度值的 L_2 範數可以降低擾動中的雜訊，因為雜訊對於模型的判斷並沒有實質的助益，並且會增大梯度值的 L_2 範數，因此，在最小化 L_2 範數的過程，雜訊即會被去除。

此外，對抗性訓練也有助於重現可識別資訊殘缺的部分。我們用線性模型來說明這個概念。

假設兩個像素在一項任務中具備相同的影響力，那麼只要兩像素所對應的權重和為定值，則權重與像素內積後加總的數值即相同。然而，如果我們將權重的 L_2 範數一併考慮，則兩像素上有著相等的權重時，L_2 範數將達到最小值。

線性模型的權重對應著類神經網路的梯度值。因此，對抗性訓練會使類神經網路傾向於均勻分配梯度值，從而重構梯度值殘缺的部分，進而提供更完整的訊

doi:10.6342/NTU202301498

息。

從以上說明可以得知，對抗性訓練能夠去除梯度值的雜訊和重構梯度殘缺的資訊，因此，此二結論均會提升類神經網路產生擾動與梯度的可識別性。

6.4 非穩健性特徵的作用

前人曾發現: 即使模型是在受對抗性擾動汙染的資料集上進行訓練，並且重新標示資料集的類別為模型誤判的類別，該模型仍可以在一般乾淨的測試資料集上達到高準確率 [35]。

我們的發現可以解釋此現象，這是因為對抗性擾動中包含了可識別資訊。因此，類神經網路即便在受汙染的資料集上進行訓練，仍具備辨識可識別特徵的能力，換句話說，模型仍然能從擾動中學習到正確辨識輸入資料的相關資訊。因此，這些模型在未受擾動影響的乾淨測試資料集上，具有正確分類測試資料的能力。

6.5 本章總結

實驗中，我們不僅發現了對抗性擾動具備人類可識別資訊，許多相關現象都可以被此發現解釋。

doi:10.6342/NTU202301498

doi:10.6342/NTU202301498

第七章　結論與展望

7.1　研究總結

在本研究中，我們主要探討了對抗性擾動導致類神經網路判斷錯誤的原因。有別於先前學者傾向認為對抗性擾動是人類與模型判斷之間存在本質差異的證據，我們的發現顯示，在疊加數千個擾動之後，可以有效地剔除擾動中的雜訊並重建缺失的資訊，從而恢復擾動中的可識別資訊。此外，我們將擾動的輪廓與背景分離，確定導致模型辨識錯誤的主要原因源自擾動中的可識別資訊。

我們的研究結果揭示了擾動具有兩種效應，即遮掩效應和生成效應，這兩種效應都會導致模型辨識出錯，但其作用機制並不相同。遮掩效應通過降低可識別資訊的像素值起作用，而生成效應則是通過產生新的可識別資訊來導致模型辨識失誤。

此外，在探討實驗過程中，我們觀察到許多有趣的現象，包括：不同的攻擊演算法產生的擾動具有高餘弦相似度；相對於模型梯度，攻擊演算法為更具可解釋性的工具；以及「消失的貓」這一神秘現象。最後，我們用此學說來解釋擾動相關現象背後的成因。

doi:10.6342/NTU202301498

7.2 未來展望

原始輸入圖片與擾動之可識別資訊間，可能存在某種轉換方式。這一想法來源於擾動的可識別資訊是從眾多模型產生擾動累加而成的結果，累加後顯示的可識別資訊應該有其存在的道理。此外，不同攻擊演算法產生的擾動具有高餘弦相似度，這也間接指向了此轉換的存在。

展望未來，我們希望能夠解開輸入圖片與擾動之間的轉換關係，如此一來，我們不僅能更深入地理解類神經網路的運作原理，同時也能真正了解對抗性擾動的本質。進而，在未來應用類神經網路時，我們將對其可靠性有更充分的信心。

doi:10.6342/NTU202301498

參考文獻

[1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. Nature, 521(7553):436–444, 2015.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. Imagenet classification with deep convolutional neural networks. Communications of the Association for Computing Machinery, 60:84–90, 2017.

[3] Geoffrey Hinton, Li Deng, Dong Yu, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine, 29:82–97, 2012.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. Attention is all you need. In Advances in Neural Information Processing Systems, 2017.

[5] Claudine Badue, Rânik Guidolini, Vivacqua Carneiro, et al. Self-driving cars: A survey. arXiv, 2019.

[6] Mohsin Kabir, F. Mridha, Jungpil Shin, et al. A survey of speaker recognition: Fundamental theories, recognition methods and opportunities. IEEE Access, 9:79236–79263, 2021.

doi:10.6342/NTU202301498

[7] Iacopo Masi, Yue Wu, Tal Hassner, et al. Deep face recognition: A survey. In Graphics, Patterns and Images, 2018.

[8] Litjens Geert, Kooi Thijs, Ehteshami Babak, et al. A survey of deep learning in medical image analysis. Medical Image Analysis, 42:60–88, 2017.

[9] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, et al. Intriguing properties of neural networks. In International Conference on Learning Representations, 2014.

[10] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, et al. Robust physical-world attacks on deep learning visual classification. In IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[11] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. arXiv, 2016.

[12] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In International Conference on Learning Representations, 2015.

[13] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[14] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In IEEE Symposium on Security and Privacy, 2017.

[15] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, et al. Square attack: A query-efficient black-box adversarial attack via random search for the l2 norm. In International Conference on Machine Learning, 2020.

doi:10.6342/NTU202301498

[16] SM Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Universal adversarial perturbations. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[17] David Rumelhart, Geoffrey Hinton, and Ronald Williams. Learning representations by back-propagating errors. Nature, 323(6088):533–536, 1986.

[18] Nicolas Papernot, Patrick McDaniel, Somesh Jha, et al. Practical black-box attacks against deep learning systems using adversarial examples. In Asia Conference on Computer and Communications Security, 2017.

[19] Andrew Ilyas, Abdelrahman Jalal, Carsten Etmann, et al. A fourier perspective on model robustness in computer vision. In IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[20] Nicolas Papernot, Patrick McDaniel, Xi Wu, et al. Distillation as a defense to adversarial perturbations against deep neural networks. In IEEE Symposium on Security and Privacy, 2016.

[21] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv, 2015.

[22] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, et al. Towards deep learning models resistant to adversarial attacks. In International Conference on Learning Representations, 2018.

[23] Kui Ren, Tianhang Zheng, Zhan Qin, et al. Adversarial attacks and defenses in deep learning. Engineering, 6:346–360, 2020.

doi:10.6342/NTU202301498

[24] Xuanqing Liu, Minhao Cheng, Huan Zhang, et al. Towards robust neural networks via random self-ensemble. In European Conference on Computer Vision, 2017.

[25] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In International Conference on Learning Representations, 2018.

[26] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, et al. Generative adversarial nets. In Advances in Neural Information Processing Systems, 2014.

[27] Naveed Akhtar, Ajmal Mian, Navid Kardan, et al. Advances in adversarial attacks and defenses in computer vision: A survey. IEEE Access, 9:155161–155196, 2021.

[28] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: From phenomena to black-box attacks using adversarial samples. arXiv, 2016.

[29] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: visualising image classification models and saliency maps. In International Conference on Learning Representations Workshop, 2014.

[30] Daniel Smilkov, Nikhil Thorat, Been Kim, et al. Smoothgrad: removing noise by adding noise. In Workshop on Visualization for Deep Learning, 2017.

[31] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, et al. Robustness may be at odds with accuracy. In International Conference on Learning Representations, 2019.

[32] Andrew Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In Association for the Advancement of Artificial Intelligence Conference, 2018.

doi:10.6342/NTU202301498

[33] Shibani Santurkar, Andrew Ilyas, Dimitris Tsipras, et al. Image synthesis with a single (robust) classifier. In Advances in Neural Information Processing Systems, 2019.

[34] A. Davies, P. Veličković, L. Buesing, et al. Advancing mathematics by guiding human intuition with AI. Nature, 600:70–74, 2021.

[35] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, et al. Adversarial examples are not bugs, they are features. In Advances in Neural Information Processing Systems, 2019.

[36] Florian Tramèr, Nicolas Papernot, Ian Goodfellow, et al. The space of transferable adversarial examples. arXiv, 2017.

[37] Athalye Anish, Carlini Nicholas, and Wagner David. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In International Conference on Machine Learning, 2018.

[38] Jia Deng, Wei Dong, Richard Socher, et al. ImageNet: A large-scale hierarchical image database. In IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[39] Yanpei Liu, Xinyun Chen, Chang Liu, et al. Delving into transferable adversarial examples and black-box attacks. In International Conference on Learning Representations, 2017.

[40] Thomas Tanay and Lewis Griffin. A boundary tilting perspective on the phenomenon of adversarial examples. arXiv, 2016.

doi:10.6342/NTU202301498

[41] Yang Song, Taesup Kim, Sebastian Nowozin, et al. PixelDefend: Leveraging generative models to understand and defend against adversarial examples. In International Conference on Learning Representations, 2018.

[42] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, et al. Adversarially robust generalization requires more data. In Advances in Neural Information Processing Systems, 2018.

[43] Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. In Advances in Neural Information Processing Systems, 2017.

[44] Henry Gouk, Eibe Frank, Bernhard Pfahringer, et al. Regularisation of neural networks by enforcing Lipschitz continuity. arXiv, 2018.

[45] Cem Anil, James Lucas, and Roger Grosse. Sorting out lipschitz function approximation. In International Conference on Machine Learning, 2019.

[46] Dennis Menn and Hung-yi Lee. Searching for the essence of adversarial perturbations. arXiv, 2022.

[47] Gamaleldin Elsayed, Shreya Shankar, Brian Cheung, et al. Adversarial examples that fool both computer vision and time-limited humans. In Advances in Neural Information Processing Systems, 2018.

[48] Anish Athalye, Logan Engstrom, Andrew Ilyas, et al. Synthesizing robust adversarial examples. CoRR, 2017.

[49] Tom Brown, Dandelion Mané, Aurko Roy, et al. Adversarial patch. arXiv, 2017.

doi:10.6342/NTU202301498

[50] Chaoning Zhang, Philipp Benz, Chenguo Lin, et al. A survey on universal adversarial attack. In International Joint Conference on Artificial Intelligence, 2021.

[51] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In European Conference on Computer Vision, 2014.

[52] Guillermo Ortiz-Jimenez, Apostolos Modas, Seyed-Mohsen Moosavi-Dezfooli, et al. Optimism in the face of adversity: Understanding and improving deep learning through adversarial robustness. arXiv, 2021.

[53] Deep learning fundamentals. https://deeplizard.com.

[54] Robert Geirhos, Carlos Temme, Jonas Rauber, et al. Generalisation in humans and deep neural networks. In Advances in Neural Information Processing Systems, 2018.

[55] Chiyuan Zhang, Samy Bengio, Moritz Hardt, et al. Understanding deep learning requires rethinking generalization. In International Conference on Learning Representations, 2017.

[56] Li Deng. The MNIST database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine, 29(6):141–142, 2012.

[57] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf, 2009.

[58] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv, 2014.

[59] Oleg Sémery. Computer vision models on PyTorch. https://github.com/osmr/imgclsmob, 2018.

doi:10.6342/NTU202301498

[60] Kaiming He, Xiangyu Zhang, Shaoqing Ren, et al. Deep residual learning for image recognition. In IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[61] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, et al. Densely connected convolutional networks. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[62] Zhongzhan Huang, Senwei Liang, Mingfu Liang, et al. DIANet: Dense-and-implicit attention network. arXiv, 2019.

[63] Dongyoon Han, Jiwhan Kim, and Junmo Kim. Deep pyramidal residual networks. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[64] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, et al. Rethinking the inception architecture for computer vision. In IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[65] Dennis Menn and Hung-yi Lee. Searching for the essence of adversarial perturbations. arXiv, 2023.

[66] Robert Hoffman, Shane Mueller, Gary Klein, and Jordan Litman. Metrics for explainable ai: Challenges and prospects. arXiv, 2019.

[67] Shanghua Gao, Zhong-Yu Li, Ming-Hsuan Yang, et al. Large-scale unsupervised semantic segmentation, 2022.

[68] Amaury Bréhéret. Pixel Annotation Tool. https://github.com/abreheret/PixelAnnotationTool, 2017.

[69] G. Bradski et al. The opencv library. https://opencv.org, 2000.

doi:10.6342/NTU202301498

[70] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. Distill, 2016.

[71] Carl-Johann Simon-Gabriel, Yann Ollivier, Leon Bottou, et al. First-order adversarial vulnerability of neural networks and input dimension. In International Conference on Machine Learning, 2019.

doi:10.6342/NTU202301498