

國立
台北科技大學

圖形識別
Final Project Report
Garbage Classification

姓名:楊子駿

學號:112318076

1. 動機與目的

在當代社會中，垃圾分類已成為一項全球性的環保挑戰。從最早的簡單垃圾處理到今天的循環利用和減廢策略，我們在垃圾管理上已取得顯著進展。然而，隨著人類活動和消費模式的多樣化，產生的垃圾類型和數量也日益增加，從日常生活垃圾到工業廢料，種類繁多，使得有效的垃圾分類和處理變得更加複雜和困難。

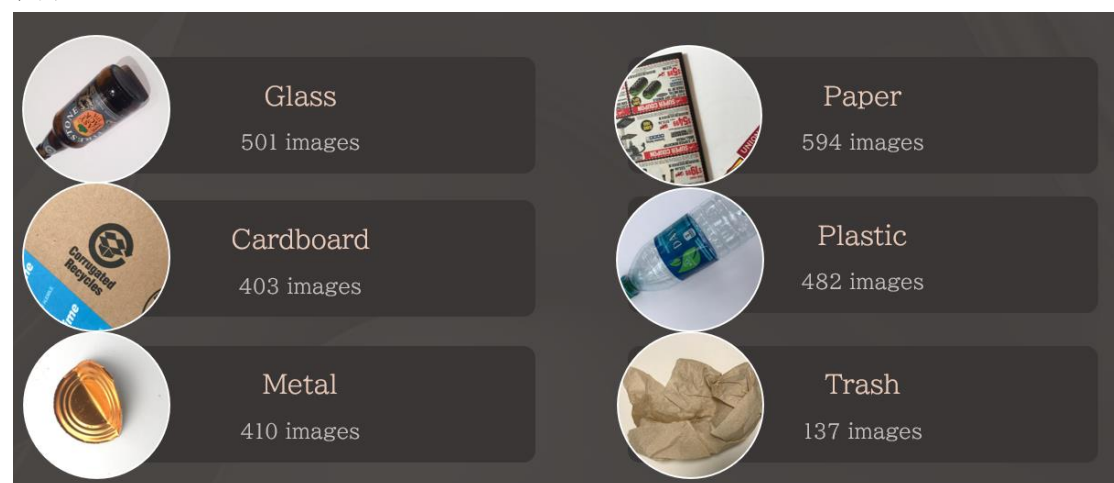
目前，垃圾分類主要依靠人工分揀和簡單的機械處理，這不僅效率低下，還容易導致分類不準確，進而影響垃圾的再利用和資源回收。此外，不當的垃圾處理方式對環境造成嚴重污染，如土壤和水源的污染，以及溫室氣體的排放，威脅到人類的健康和地球的可持續發展。

針對這一問題，本專題的目標是通過技術創新，為垃圾分類提供一種新的解決方案。該技術旨在通過分析和識別垃圾照片，自動進行精確的垃圾分類，從而提高分類效率和準確性。我計劃使用 Kaggle 上的 TrashNet 數據集進行模型訓練，並採用先進的卷積神經網絡進行深度學習，以期達到高效的垃圾識別和分類，以應對日益增長的垃圾處理問題。

2. 實驗方法

在當今科技快速發展的時代，深度學習技術已成為解決諸多挑戰的關鍵工具之一。我的專題「Garbage Classification」旨在運用深度學習技術，開發一套能夠有效分類照片中垃圾的模型，以提高垃圾分類的準確性和效率。為此，我採用了 Kaggle 上的 TrashNet 數據集，這是一個包含不同類型垃圾照片的資料庫。

實驗的第一步是數據預處理和分割。我們將數據集中 70% 的數據作為訓練集，用於模型的學習和訓練，而剩餘的 30% 則作為驗證集，用於評估模型的性能和準確度。考慮到數據集的總量不大，我採用了數據增強技術，如旋轉、翻轉、縮放等方法來增加數據多樣性，這有助於提高模型的泛化能力並防止過度擬合。



圖一：TrashNet 數據集分類示意圖

在模型架構的選擇上，我決定使用 DenseNet-121 和 DenseNet-169。DenseNet 是一種創新的卷積神經網絡結構，與傳統的卷積網絡相比，它的特點在於其「密集連接」的架構。在 DenseNet 中，每一層都直接與之前所有層相連接，這意味著網絡的每一層都可以直接訪問其之前層的特徵圖，從而促進了特徵的再利用，減少了參數的數量，並提高了效率。

這種密集連接方式有幾個明顯的優點：首先，它強化了特徵的傳播，使得每一層都可以接收到之前所有層的貢獻；其次，由於更有效的特徵使用，它有助於減少網絡的深度和複雜度，從而降低過擬合的風險；最後，這種結構還提高了網絡的訓練效率。

Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112×112	7×7 conv, stride 2			
Pooling	56×56	3×3 max pool, stride 2			
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56×56	1×1 conv			
	28×28	2×2 average pool, stride 2			
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28×28	1×1 conv			
	14×14	2×2 average pool, stride 2			
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	14×14	1×1 conv			
	7×7	2×2 average pool, stride 2			
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	1×1	7×7 global average pool			
		1000D fully-connected, softmax			

圖二：DenseNet121/169 主要架構

我的主要訓練環境包括一台配備了 NVIDIA GTX 1080 顯卡和 Intel Core i7-3770 處理器的電腦。這樣的硬件配置足以支持我的深度學習模型訓練，並確保在處理大量數據時的計算效率。

本專題的主要目的是比較和評估不同模型結構、批量大小和其他參數設定對垃圾分類準確性的影響。通過這些實驗，我希望找出最適合垃圾分類任務的模型配置，並進一步優化模型的性能。

3. 模型訓練及實驗結果

Model	Batch size	Epoch	Accuracy	Loss	Validation accuracy	Validation loss
DenseNet121	8	100	0.971	0.283	0.951	0.34
	16	100	0.934	0.445	0.812	0.918
DenseNet169	8	100	0.979	0.231	0.972	0.32
	16	100	0.98	0.21	0.983	0.54

圖三：DenseNet121/169 訓練結果

從上述的分類結果可以注意到，DenseNet121 和 DenseNet169 模型在不同批量大小下的表現有顯著差異。對於 DenseNet121 模型，當批量大小為 8 時，我觀察到訓練精確度達到了 0.971，而驗證精確度為 0.951，顯示出良好的學習能力和泛化性能。然而，當批量大小增加到 16 時，訓練精確度下降到 0.934，

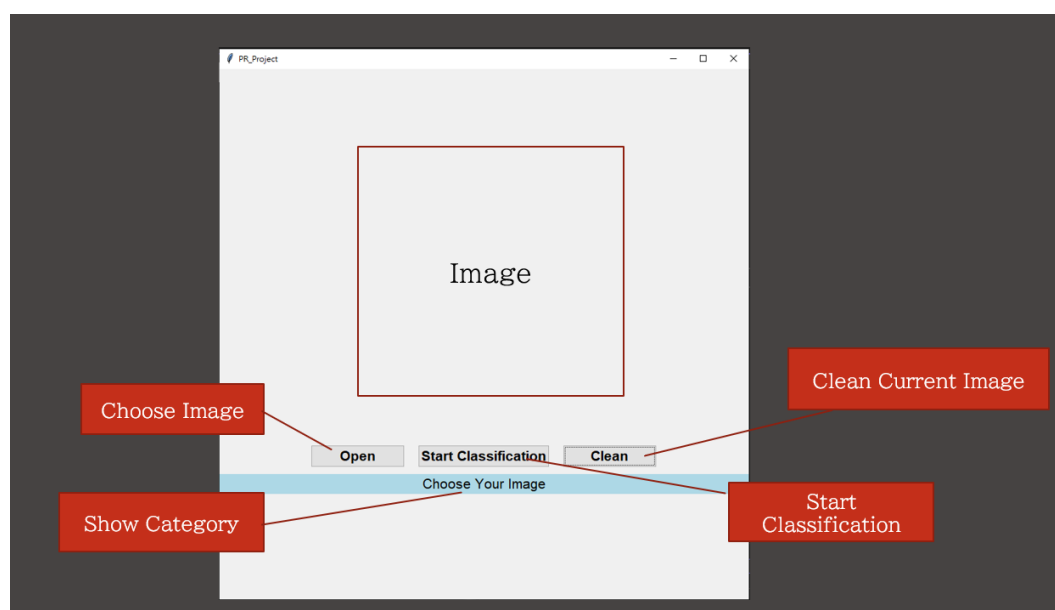
驗證精確度也相應下降至 0.812，這可能表明更大的批量大小對於該模型來說並非最優選擇。

相比之下，DenseNet169 模型在批量大小為 8 時表現優異，訓練精確度達到 0.979，驗證精確度更是高達 0.972，而當批量大小提升到 16 時，其表現更是驚人，訓練精確度和驗證精確度分別達到了 0.98 和 0.983。這一結果指出，對於 DenseNet169 來說，較大的批量大小有助於提高模型的性能。

在損失方面，我注意到 DenseNet169 在批量大小為 16 時的驗證損失（Validation loss）為 0.54，相比之下，批量大小為 8 時的 0.32 有所增加。這可能意味著雖然提高批量大小有助於提高精確度，但也可能增加模型在驗證集上的損失，這需要進一步的分析來確定最佳的批量大小。

在經過一系列的深度學習實驗後，DenseNet169 模型在批量大小為 16 的條件下展現出最優的性能。這一模型在 100 個訓練週期後達到了 0.98 的訓練精確度和 0.983 的驗證精確度，儘管驗證損失稍高於批量大小為 8 時的結果，但整體表現仍然非常卓越。

而最後我將訓練好的模型整合進用戶界面(UI)。可以通過點擊“Choose Image”按鈕來選擇他們想要分類的垃圾圖片。選擇圖片後，用戶可以點擊“Start Classification”按鈕來運行垃圾分類模型。在分類過程中，如果用戶想要重新開始，可以使用“Clean Current Image”按鈕來清除當前圖片，準備進行新一輪的分類，而分類結果將通過點擊“Show Category”按鈕展示。



圖三：UI 整合圖

4. 結論

整體來說，我認為我的模型訓練結果有達到預期，可以將大部分的垃圾圖片成功分類，也有將其編寫成一個可以使用的 GUI 介面，未來期許可以將其整合成一個應用程式，讓使用方式更多元和現代，使使用者快速且有效的分類各種垃圾，減輕清潔人員的工作，還給大家乾淨的地球環境。