Deep Learning - Assignment II

RE6121029 張立勳

關鍵字—DynamicConv2D, Residual-in-Residual Dense Block, RRBD, Self-Attention Layer

I. 介紹

本次作業為一深度學習模型實作，資料是基於 ImageNet-mini 資料集。其內容可分為二大部分，第一部分需設計一個特殊的卷積模組，能夠處理任意數量的輸入通道；第二部分需設計一個 4 至 6 層的 CNN、Transformer 或 RNN 網絡架構，使其在 ImageNet-mini 資料集上能夠達到 ResNet-34 90%的表現。上述二部分都使用 ImageNet-mini 數據集進行訓練並與傳統模型(ResNet)做比較。

資料來源: https://cchsu.info/files/images.zip

II. 使用方法

A. *DynamicConv2D*

DynamicConv2D 是一個可以執行動態卷積操作的自訂 PyTorch 模組，此模組可基於輸入通道數量調控卷積核權重，使輸入可以允許不同的通道數量。首先，在初始化階段定義了動態卷積核的權重和偏差，在前向傳播過程中，卷積會根據輸入的通道數動態調整卷積核的權重和偏差。如果輸入的通道數少於初始化階段時隨機指定的，則只選擇對應通道的權重，使此模組能夠彈性處理具有不同通道數的輸入，特別適合多傳感器數據處理或處理圖像的不同特徵圖。

B. *Residual-in-Residual Dense Block (RRBD)*

DenseBlock 是一個基本的卷積模塊，包含卷積、批次正規化和 ReLU 激活函數。在前向傳播過程中，輸入首先通過卷積層，然後進行批次正規化，最後通過 ReLU 激活函數。而 RRDB 是一個包含多層 DenseBlock 的模塊，並在輸出時添加了殘差連接層。初始化階段中定義了 3 層 DenseBlock，每層的輸入通道數會隨著增長率逐漸增加，在此模型通道數固定為 32。除此之外，在 DenseBlock 模塊後，還有一個額外的卷積層來調整通道數，以及 ReLU 激活函數。在前向傳播過程中，輸入依次通過每個 DenseBlock，並且每個 DenseBlock 的輸出都與前一層的輸入進行連接，接著通過一層卷積層和 ReLU 激活函數，最後將輸出與初始輸入進行相加，形成殘差連接。此模組運用了密集連接和殘差學習的優點，使得模型能夠有效捕捉多層特徵，同時保持較低的參數量和計算成本。

C. *Self-Attention Layer*

Self-Attention Layer 增強了模型的表現，使其能夠專注於輸入的不同區塊。此模組使用三個卷積層來生成查詢（query）、鍵（key）和值（value）特徵圖，分別為 query_conv、key_conv 和 value_conv 3 層，query_conv 和 key_conv 這兩個卷積層將通道數減少為原來的八分之一，而 value_conv 則保持與原來相同的通道數。在前向傳播過程中，首先輸入通過這些卷積層生成查詢、鍵和值特徵圖，然後查詢特徵圖被重塑並轉置，以匹配批量矩陣相乘所需的維度。接著，通過對查詢和鍵進行批量矩陣相乘，並根據輸入通道數進行縮放，來計算注意力分數，這些注意力分數通過 Softmax 函數轉換成注意力權重。將值特徵圖與注意力權重相乘，並將結果重塑為與原始輸入相同的維度。最後，將輸入添加到結果中形成殘差連接，這使得網絡能夠動態地專注於輸入的不同區塊，提高模型捕捉長距依賴關係和複雜模式的能力，有效提升模型的表現，使其在處理圖像、語音等多種任務時具備更強的適應性和表現力。

III. 實作結果

本次作業第一部分需設計一個特殊的卷積模組，能夠處理任意數量的輸入通道。將 ResNet-18 的第一個卷積層替換成 DynamicConv2D，以處理不同通道數的輸入。除此之外，在本次實作中將有 DynamicConv2D 的新模型與原始模型做比較，實作中所有圖片皆使用固定尺寸(84x84)，應用多種圖像增強技術，包含水平和垂直翻轉、旋轉和扭曲，並評估模型在不同輸入下的準確性和耗時，包括 RGB、RG、RB、GB、R、G 和 B。

A. *DynamicConv2D*

表 1 是以測試資料集檢驗有 DynamicConv2D 的 ResNet18 模型對不同輸入的預測結果(%)。

	RGB	RG	RB	GB	R	G	B
Accuracy	54.22	53.56	52.00	54.22	50.00	52.67	52.00
Precision	56.11	53.97	52.44	56.55	51.50	54.41	54.26
Recall	54.22	53.56	52.00	54.22	50.00	52.67	52.00
F1-Score	53.69	52.56	50.86	54.08	48.99	51.93	51.44
FLOPS	926579712						

Table 1. 不同輸入下模型預測結果

由上表可觀察到三通道的表現優於二通道和單通道，但彼此差距不大。除此之外，可以由二通道和單通道的預測結果得出輸入有 R 的資料在此模型的預測能力較差，反映在輸入為 RG、RB、R 的表現上。FLOPS（每秒浮點運算次數）是一個衡量模型複雜度的常用指標，在此模型中所有輸入的 FLOPS 是相同的，代表在此模型中不同輸入不會影響其效能。因此，只須考慮準確率相關指標的話，建議使用三通道輸入。

表 2 是以測試資料集檢驗有 DynamicConv2D 的 ResNet-18 模型與原始 ResNet-18 模型的預測結果(%)。

	ResNet18(DynamicConv2D)	ResNet18
Accuracy	54.22	38.44
Precision	56.11	38.91
Recall	54.22	38.44
F1-Score	53.69	37.66
FLOPS	926579712	296456064

Table 2. ResNet18 有無 DynamicConv2D 的模型預測結果

從表 2 可以看出，在相同的訓練條件下，ResNet-18（DynamicConv2D）在測試集上的三通道表現顯著優於原始 ResNet-18。然而，其 FLOPS 高出三倍以上。

本次作業第二部分需設計一個 4 至 6 層的 CNN、Transformer 或 RNN 網絡架構，使其在 ImageNet-mini 資料集上能夠達到 ResNet-34 90%的表現。實作模型架構為

Identify applicable funding agency here. If none, delete this text box.

5 層，第一層是一個 Dense Block，使用的 kernel size 是 5x5，輸出 512 個通道；第二層是三層的 RRBD，每層使用的 kernel size 是 3x3，輸出都是 32 個通道，再接一卷積層使輸出擴成 512 個通道，並以 MaxPooling 萃取特徵精華；第三層是 Self-Attention 層，輸出 512 個通道，並以 AdaptiveAveragePooling 擷取特徵；第四層是 Dropout 層，避免模型過擬和；最終第五層是全連接層，輸出 50 個通道。在本次實作中將實作模型與比較模型(ResNet-34)做比較，實作中所有圖片皆使用固定尺寸(84x84)，應用多種圖像增強技術，包含水平和垂直翻轉、旋轉和扭曲，並評估模型的準確性和耗時。

B. Simple CNN

表 3 是以測試資料集檢驗實作模型(Simple CNN)與比較模型(ResNet-34)的模型預測結果(%)。

	SimpleCNN	ResNet34
Accuracy	46.89	55.11
Precision	51.64	55.89
Recall	46.89	55.11
F1-Score	46.72	54.22
FLOPS	7697151972	612876800

Table 3. Simple CNN 和 ResNet-34 的模型預測結果

由表 3 可看出 Simple CNN 不是很有效率的模型，在 FLOPS 指標上，它與 ResNet-34 差 10 倍以上，整體模型表現近似於 ResNet-34 的 9 成，綜合來說 ResNet-34 是更好的模型。

IV. 結論

在第一部分，本次作業設計了一個能夠處理不同輸入通道數量的卷積模塊 DynamicConv2D。實作結果顯示改進後的 ResNet-18 在 Accuracy、Precision、Recall 和 F1-Score 方面優於原始 ResNet-18，儘管在 FLOPS 方面有較高的計算成本。而且本次實作也驗證，使用三個輸入通道可獲得最佳性能，儘管此模塊可以適應更少的通道並保持相當的效能。

在第二部分，Simple CNN 雖然顯示出不錯的 Accuracy、Precision、Recall 和 F1-Score，但 FLOPS 表現上遠低於 ResNet-34，這也許代表模型過於複雜，運算成本高，或許可以減少層數，降低模型複雜度，並有可能實現超越 ResNet-34 模型。

總體而言，本次實作設計的卷積模塊和網絡架構為處理多樣化輸入的卷積神經網絡提供了許多的想法和未來可能的改進方向，不僅如此，也在本次實作中感受到了 Self-Attention 機制和動態卷積在深度學習中的重要性。