



**國立臺北科技大學**

**電子工程系碩士班**

**碩士學位論文**

**使用 EfficientNet 做台灣美食分類**

**Taiwanese Cuisines Classification using EfficientNet**

**研究生：林芷羽**

**指導教授：賴冠廷 博士**

**中華民國一百一十一年七月**

**國立臺北科技大學**  
**研究所碩士學位論文口試委員會審定書**

本校 電子工程 研究所 林芷羽 君

所提論文，經本委員會審定通過，合於碩士資格，特此證明。

學位考試委員會

委

員：

黃育賢

賴冠廷

陳俊傑

指導教授：

賴冠廷

所

長：

蘇學凱

中 華 民 國   一 百 一 十 一   年   七   月   二 十 二   日

# 摘要

論文名稱：使用 EfficientNet 做台灣美食分類

頁數：三十三頁

校所別：國立臺北科技大學 電子工程系 碩士班

畢業時間：一百一十學年度 第二學期

學位：碩士

研究生：林芷羽

指導教授：賴冠廷 博士

關鍵詞：EfficientNet、深度學習、食物辨識、卷積神經網路

現今科技發達，深度學習被廣泛應用在各種方面，包含人臉辨識、車牌辨識、聲音辨識等等，這使我們在生活中更加方便，不需要多餘的時間等待人工審核即可做許多的事情，深度學習的應用對我們的生活有著很大的幫助。

而本論文主要為了讓外國人來到台灣能多加了解台灣的飲食文化，所以致力於提高台灣美食的辨識率，本實驗使用了 Taiwanese Food 101 數據集來做深度學習的訓練，數據集中包含了 101 種台灣道地美食，原始 dataset 每種食物圖片各 200 張，而本研究為了提高辨識率又將每種食物的圖片資料庫各增加了 50 至 100 張左右，每一種類數量大約是 250 至 300 張，而訓練的模型我們這裡使用了 VGG16、InceptionV3、MobileNet、Xception、DenseNet、EfficientNet 等等，並從中找出了最適用的模型 EfficientNet。

最後我們使用 EfficientNet 來訓練，不斷的在測試集上評估模型並進行調整，最高的準確率達到 85.1%，此研究之後將可以使用在各種地方，帮助大家去了解台灣美食，並附上資料庫可供使用：<https://www.kaggle.com/kuantinglai/taiwanese-food-101>。

# ABSTRACT

Title: Taiwanese Cuisines Classification using EfficientNet

Pages: 33

School: National Taipei University of Technology

Department: Electronic Engineering

Time: July, 2022

Degree: Master

Researcher: LIN ZHI YU

Advisor: KUAN-TING LAI Ph.D.

Nowadays, with the development of technology, deep learning is widely used in various aspects, including face recognition, license plate recognition, voice recognition, etc. This makes our life more convenient because we can do many things without waiting for extra of human processing time. This paper is mainly for foreigners to understand more about Taiwan's food culture, so we are committed to improving the recognition rate of Taiwanese foods. In order to improve the recognition rate, we expand the original dataset by adding extra 50 to 100 pictures to each class, and the number of each category becomes about 250 to 300 pictures. The training models we used for classification include VGG16, InceptionV3, MobileNet, Xception, DenseNet, EfficientNet. The best model among them was EfficientNet. We used EfficientNet to train and continuously tune the model on the test set. Finally, the highest accuracy rate reached 85.1%. We believe this study can be used in various applications to help people understand Taiwanese cuisines. The dataset can be downloaded from Kaggle (<https://www.kaggle.com/kuantinglai/taiwanese-food-101>).

## 誌謝

研究所兩年的時間，我要感謝的人實在太多了，首先感謝我的指導教授賴冠廷老師，從我什麼都不太會來到實驗室，一步一步帶我了解實驗室所有的研究內容，並帶我參與 Coding 365 的計畫慢慢學習、慢慢探索感興趣的部分，從完全不會到有一點了解，最後終於有一個研究方向，同時老師也找了許多學長、同學們協助我，最後才能完成此篇論文，真的非常感謝老師不厭其煩的指導。

再來感謝很熱心教我、協助我的學長們，從我進來什麼都不會，璨綸學長推薦我一些可以參考的書，我慢慢研究的過程中，所有程式不會的部分只要詢問學長們，他們都會快速有耐心地教我，也帶我了解許多寫論文應該注意的事項，還有課程跟畢業的相關內容，真的非常感謝璨綸學長、俊嘉學長、啟宏學長、朝育學長、勇勤學長的照顧。

另外還要感謝學弟在程式方面的幫忙，也協助我跑了許多程式，讓我論文可以順利完成實驗，還有 GERRY 在論文程式方面的大力協助。

最後，非常感謝實驗室的同學岳聰、昌諭、鼎鈞，帶我熟悉環境，一起學習，時常關心我的狀況，協助我在北科大的各種大小問題，所有活動也帶我一起參與，因為有大家的協助，才讓我在北科大能夠順利完成論文，非常謝謝大家。

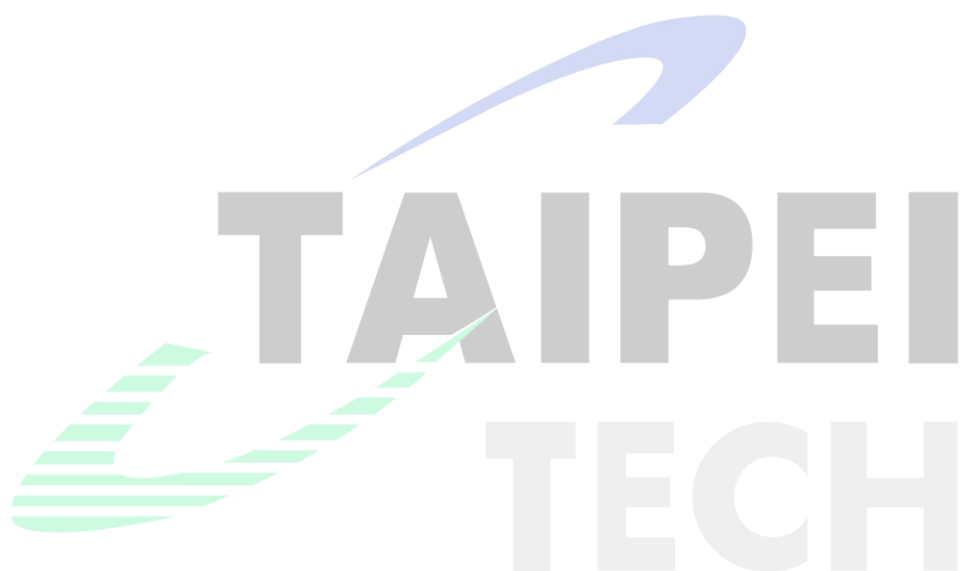
# 目錄

摘要 .....	i
ABSTRACT .....	ii
誌謝 .....	iii
目錄 .....	iv
表目錄 .....	vi
圖目錄 .....	vii
Chapter 1. 介紹 .....	1
1.1 研究背景 .....	1
1.2 研究動機 .....	2
1.3 研究目的 .....	2
1.4 論文架構 .....	2
Chapter 2. 文獻探討 .....	4
2.1 資料庫種類 .....	4
2.2 辨識方法 .....	5
2.3 模型介紹 .....	6
2.3.1 VGG .....	7
2.3.2 Inception .....	8
2.3.3 MobileNet .....	9
2.3.4 Xception .....	11
2.3.5 DenseNet .....	12
2.3.6 EfficientNet .....	13
2.4 食物辨識的應用 .....	14
2.4.1 營養評估 .....	14

2.4.2 餐點辨識.....	15
2.4.3 計價系統.....	15
2.4.3 飲食健康管理.....	17
Chapter 3. 研究方法.....	18
3.1 Taiwanese Food 101 數據集.....	18
3.2 模型訓練.....	20
Chapter 4. 研究結果與分析.....	26
4.1 研究結果.....	26
4.2 模型訓練結果比較.....	28
Chapter 5. 結論與建議.....	30
5.1 結論.....	30
5.2 貢獻.....	30
5.3 研究限制.....	31
參考資料.....	32

## 表目錄

表 2-1 CNN 模型比較[9] .....	7
表 4-1 在各模型上的測試準確率 .....	29





## 圖目錄

圖 2-1 中東美食 [1] .....	4
圖 2-2 泰國快餐 [2] .....	5
圖 2-3 使用 AlexNet、GoogLeNet 和 ResNet-50 訓練測試 Food-5K、Food-11 和 Food-22 結果[5].....	6
圖 2-4 Inception 與 InceptionV2 的架構[10] .....	8
圖 2-5 depthwise-separable convolutions.....	9
圖 2-6 $X \times X \times 1$ 的 kernel.....	10
圖 2-7 Depthwise Convolution[11] .....	10
圖 2-8 $1 \times 1 \times X$ 的 Kernel.....	10
圖 2-9 Pointwise Convolution[11].....	11
圖 2-10 Residual Learning 概念圖[12].....	12
圖 2-11 利用 CIFAR-10 針對各層做 Batch Normalization[12] .....	12
圖 2-12 DenseNet 的前饋方式(feed-forward)[13] .....	13
圖 2-13 EfficientNet-B1 到 B7 和其它 CNN 模型的效能比較[14].....	14
圖 2-14 將照片中每種菜色的座標框出來 [19] .....	16
圖 2-15 自助餐菜色自動辨識結帳系統 [19] .....	17
圖 3-1 研究方法.....	18
圖 3-2 Taiwanese Food 101 資料庫 .....	19
圖 3-3 Taiwanese Food 101 資料庫中的食物圖片 .....	19
圖 3-4 使用的模型 .....	20
圖 3-5 統一圖片張數.....	20
圖 3-6 模型訓練 Taiwanese Food 101 過程 .....	21
圖 3-7 測試集上評估模型並保存 .....	21

圖 3-8 對測試集進行預測並生成混淆矩陣和分類報告 .....	21
圖 3-9 Taiwanese Food 101 混淆矩陣和分類報告 .....	25
圖 4-1 訓練 5 次得到的數據值 .....	26
圖 4-2 在測試集上評估並保存模型 .....	27
圖 4-3 在測試集上預測的準確度 .....	27
圖 4-4 在測試集上的錯誤分類數量 .....	28
圖 4-5 模型年代 .....	29
圖 5-1 一種食物呈現多種樣貌 .....	31



# Chapter 1. 介紹

## 1.1 研究背景

美食無處不在，當你打開社交 APP，隨時都能看到不一樣的食物被拍的美美的，有來自日本、美國、法國、義大利、澳洲等等，來自世界各地多樣化的食物映入眼簾，各種不同的美食，有可能你很想去吃卻不知道他是什麼或者來自哪裡，為了了解圖片中的美食也許你要問許多的親朋好友才能得知它究竟是什麼食物，或者找遍所有資料也仍就不知道是什麼。

為了解決這個問題，許多人開始使用各種方法來辨識世界各地的美食，這樣的研究使我們不需要透過詢問他人才能知道是什麼美食，只要透過一些 APP 點一下便能得知它是什麼食物。

而這些美食能使用機器辨識後又延伸出了許多生活上的應用，像是許多無人商店出現，這是一種新零售的概念，商店內沒有任何收銀服務人員，只需要拿起商品到結帳區付錢即可，這樣的發明讓大家的的生活更便利，也可以降低人力成本，此研究需要使用許多先進的技術來達成，其中最重要的一項便是 AI 人工智慧的機器視覺技術，必須先讓機器知道你拿了什麼商品，才能接著去做後續的計算價錢並結帳，食物辨識的研究在此就占了非常重要的腳色。

在科技發達的現在，除了生活方便之外的應用之外，人們更加注重自己的健康飲食，會使用許多方法來管理自己的飲食與健康，有些人可能每天手寫記錄自己吃了哪些東西，再去查詢並計算食物的熱量等等，為了方便記錄，營養評估和飲食健康管理系統便陸陸續續的出現在我們生活中，只要將食物進行拍攝便能幫您記錄起來，並顯示食物的熱量

與食物的成分等等，這些方便的應用使我們不需要再自己花許多時間來記錄，簡單的操作一些系統就能完成這些工作。這些應用都需要食物辨識為基礎來做後續的延伸，如果一開始就將食物辨識錯誤，那後續的應用將無法實現。

## 1.2 研究動機

食物的辨識有很多人做了，但辨識台灣美食的研究卻幾乎沒有，即使有一些人做了，但辨識率也不是特別高，這使得即便外國人來到台灣使用辨識食物的 APP 也不能知道是什麼食物，也有可能辨識錯誤，導致台灣的當地美食無法快速的被大家了解，實在是太可惜了，為了使台灣美食能更加快速地被了解，提高辨識率是不可或缺的重點，因此本研究致力於提高台灣美食的辨識率。

另外也希望可以將高辨識率的台灣美食應用在生活中，像是使用在無人商店、營養評估、飲食健康管理等等，可以增加生活的方便性，造就一個更健康的社會。

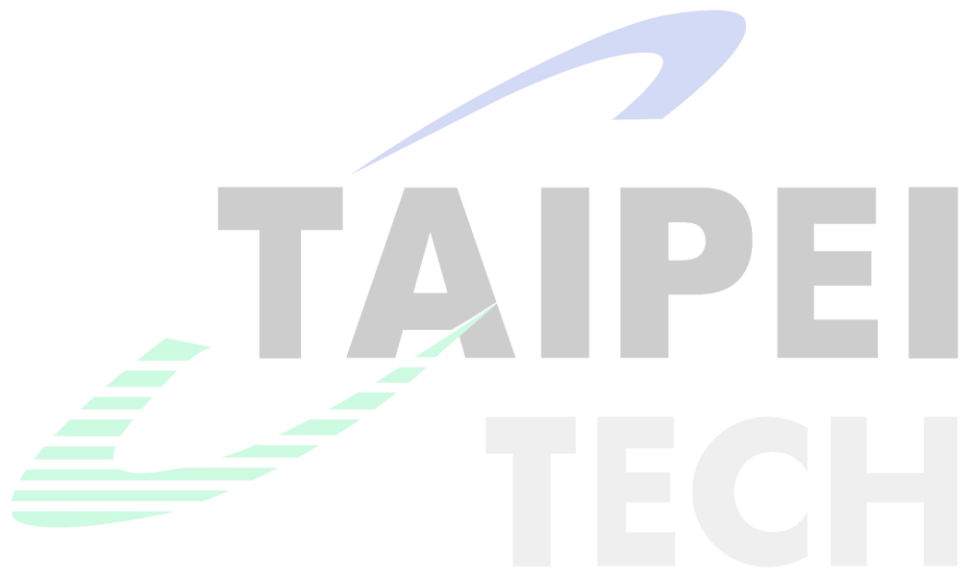
## 1.3 研究目的

本研究為了提高 Taiwanese Food 101 辨識率，使用了許多不同的方法，像是使用不同的模型去訓練，其中包含 VGG16、InceptionV3、MobileNet、Xception、DenseNet、EfficientNet 等等，另外也有增加圖片數量去嘗試增加辨識率、將圖片做不同的處理、嘗試了多種方法，最終得到 EfficientNet 為所有方法中辨識率最高的模型。

## 1.4 論文架構

第一章介紹研究的背景、動機與目的。第二章主要針對食物的辨識進行研究，文獻中不同的辨識方法、不同的資料庫、以及不同的模型。第三章為研究方法，Taiwanese Food

101 數據集如何經過訓練，最後得到結果。第四章對研究結果進行介紹、分析與比較。第五章得出 EfficientNetB3 為本研究最佳的預測結果，也是本論文的主題，另外提出了對於 Taiwanese Food 101 的貢獻以及研究上所遇到的種種限制，最後提出建議。



## Chapter 2. 文獻探討

本章介紹資料庫的種類，不同文獻中提到的辨識方法，以及深度學習的各種模型。

### 2.1 資料庫種類

有很多食物辨識的研究，因為食物種類十分多樣，不同地區有不同的美食，而要做這些食物辨識時都需要整理食物的資料庫，網路上的資料庫十分多樣化，像是種中東美食 [1]（圖 2-1）、泰國快餐 [2]（圖 2-2）、Food-5K [3] 和 Food-11 [3] 等等資料庫，各式各樣的美食資料庫，在網路以及各種文獻中都能找到。

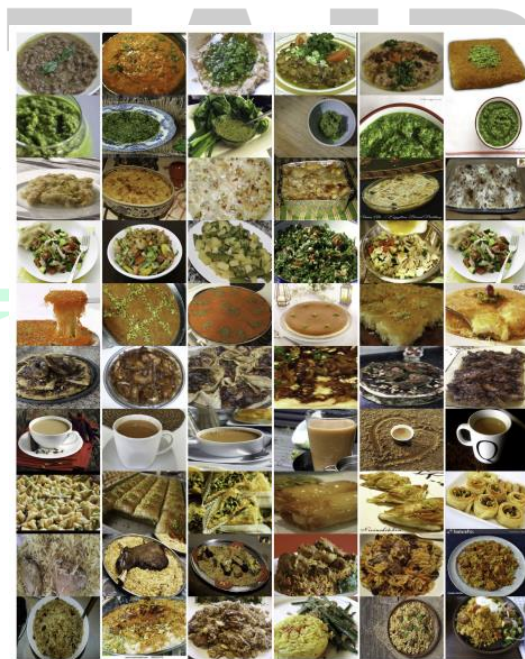


圖 2-1 中東美食 [1]

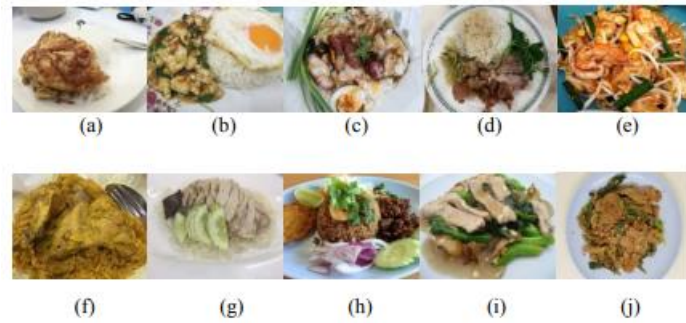


圖 2-2 泰國快餐 [2]

## 2.2 辨識方法

可以用來做食物辨識的方法有很多種，如文[4]中提到使用顏色分割和霍夫變換以及 Cascaded Object Detector (COD) 的方式來篩檢西紅柿是否成熟，顏色分割和霍夫變換是較為傳統的方法，將一張圖片的顏色作分割後再去做 RGB 和一些運算處理，最後判定是否為成熟的西紅柿，而 Cascaded Object Detector (COD) 為機器學習的方法，COD 會生成 histogram of oriented gradient (HOG) 特徵去檢驗西紅柿的狀況，COD 的辨識結果高達 95%，比較來說機器學習更優於傳統的顏色分割。

有些人研究食物辨識是希望能識別不認識的美食，進而對美食增加了解，有些人則是希望有更多延伸的利用，像是文[5]，主要是為了做飲食的研究，希望可以用在發展健康飲食，研究中使用了 AlexNet [6]、GoogLeNet[7] 和 ResNet-50[8]模型來訓練 Food-5K 和 Food-11 食物資料庫，Food-5K 資料庫中包含食物與非食物，而 Food-11 資料庫中則有 11 種不同食物的類別，他們將資料庫隨機分割成 70% 用來訓練、30% 用來測試，另外他們在研究中創建了 Food-22[5]資料庫，包含蔬菜、水果、穀物、肉類和奶製品總共 5 種類別的食物，也使用一樣的方法來進行訓練與測試，並做比較如（圖 2-3），可以看出類別多了一倍，準確度大約也下降了快一半，對於本研究的 101 種食物來說準確率肯定會更低。



Database	Number of Classes	Classifier	DCNN/Feature Extractor	Training Time (s)	Accuracy (%)	Average Time (s)	Average Accuracy (%)
Food-5K	2	Re-trained DCNN	AlexNet	12386	98.00	12509	98.37
			GoogLeNet	12487	99.07		
			ResNet-50	12654	98.05		
		SVM trained on extracted features from pre-trained DCNN	AlexNet	3	97.53	3	98.06
			GoogLeNet	4	98.02		
			ResNet-50	1	98.80		
		ANN trained on extracted features from pre-trained DCNN	AlexNet	1	99.40	1	99.03
			GoogLeNet	1	98.5		
			ResNet-50	1	99.20		
Food-11	11	Re-trained DCNN	AlexNet	119073	78.89	125134	83.52
			GoogLeNet	135983	83.59		
			ResNet-50	120347	88.09		
		SVM trained on extracted features from pre-trained DCNN	AlexNet	583	76.84	404	83.39
			GoogLeNet	621	87.66		
			ResNet-50	4	85.66		
		ANN trained on extracted features from pre-trained DCNN	AlexNet	27	81.40	21	83.33
			GoogLeNet	35	81.50		
			ResNet-50	2	87.10		
Food-22	22	Re-trained DCNN	AlexNet	540380	59.67	423316	58.72
			GoogLeNet	550458	61.30		
			ResNet-50	179110	55.18		
		SVM trained on extracted features from pre-trained DCNN	AlexNet	1767	58.37	1324	56.05
			GoogLeNet	2197	57.52		
			ResNet-50	6.93	52.25		
		ANN trained on extracted features from pre-trained DCNN	AlexNet	80	55.23	53	53.65
			GoogLeNet	76	54.74		
			ResNet-50	2	50.98		

圖 2-3 使用 AlexNet、GoogLeNet 和 ResNet-50 訓練測試 Food-5K、Food-11 和 Food-22

結果[5]

食物辨識用在健康飲食方面，除了大方向的對所有人之外，也會也針對某些族群的研究，文[1]中就是針對糖尿病患者所做的飲食研究，他們使用特徵提取、單獨分類和結合了三個不同的技術研究，研究顯示結合不同的技術後，辨識率有所提升。

## 2.3 模型介紹

CNN 是卷積神經網路的簡稱，是深度學習的一個基礎模型，主要用在影像識別上，CNN 是參考人的大腦視覺組織所建立的深度學習模型，當學會基礎的 CNN 後，對於其他的深度學習模型會較容易上手。

CNN 可以保留圖像的資訊，不像一些需要展開做處理的模型，會導致失去很多的資訊，效果就會不佳，CNN 是透過卷積核滑動對圖像做提取訊息的動作，然後在經過步長跟填充去控制圖像的長寬，這樣一來就可以保留圖像的資訊，得到較佳的效果了。

以下介紹幾個由 CNN 延伸出去的模型，分別是 VGG、InceptionV3、MobileNet、Xception、DenseNet、EfficientNet。



### 2.3.1 VGG

VGG 是來自牛津大學計算機視覺組（Visual Geometry Group）所研究出的模型，是由 Simonyan 和 Zisserman 在《Very Deep Convolutional Networks for Large Scale Image Recognition》文獻中提出的卷積神經網路模型，主要證明了增加網絡的深度可以影響網絡的性能。

VGG 的結構簡潔且一致皆為 3\*3 的卷積核和 2\*2 的 max pooling，使用了 3\*3 的卷積核來替代 7\*7 與 5\*5 的卷積核，保證在相同條件下提升網絡的深度，並且提升神經網絡的效果，但是消耗的資源較多，也使用了較多的參數，造成內存使用較高。

它的結構，第一層通道數為 64，每層都會翻倍，通道數目增加使更多信息能被提出，而卷積核負責擴大通道數、池化負責縮小寬高，使得模型更深更寬可控制計算量並且增加規模。

由（表 2-1）可以看出相對層數差不多的模型來說，VGG 的準確度比 Inception 和 AlexNet 高，但所消耗的時間多了不少。

表 2-1 CNN 模型比較[9]

Network	Layers	Top-1 error	Top-5 error	Speed (ms)
AlexNet	8	42.90	19.80	14.56
Inception-V1	22	-	10.07	39.14
VGG-16	16	27.00	8.80	128.62
VGG-19	19	27.30	9.00	147.32
ResNet-18	18	30.43	10.76	31.54
ResNet-34	34	26.73	8.74	51.59
ResNet-50	50	24.01	7.02	103.58
ResNet-101	101	22.44	6.21	156.44
ResNet-152	152	22.16	6.16	217.91
ResNet-200	200	21.66	5.79	296.51

VGG16：

VGG16 最大的特點就是簡單，包含 16 個隱藏層，其中包含 13 個卷積層和 3 個全連接層。

VGG19：

VGG19 包含 19 個隱藏層，其中包含 16 個卷積層和 3 個全連接層，與 VGG16 只有網路深度不同的差別，造成了 VGG19 佔用了較高的記憶體，準確率上升，但計算時間加長。

### 2.3.2 Inception

InceptionV2 為 5\*5 卷積層（圖 2-4 左圖），InceptionV2 則是將 Inception 的 5\*5 卷積層改成兩個 3\*3 卷積層（圖 2-4 右圖）。

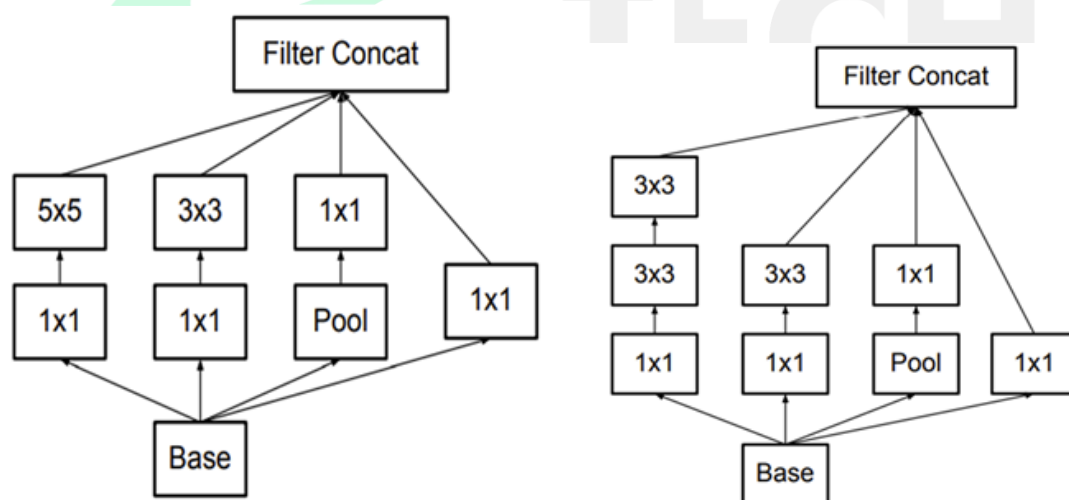


圖 2-4 Inception 與 InceptionV2 的架構[10]

InceptionV3 和 InceptionV2 都出自同一篇論文，論文中提到了幾點，輸入到輸出維

度需要慢慢下降，才不會過度壓縮，導致特徵消失難以辨識，在網路中使用非線性可以讓訓練速度加快，增加網路寬度跟深度可以有更好的效能，InceptionV3 對比 InceptionV2 做了許多的改進，像是將卷積改成了不對稱的卷積，縮小特徵圖等等，以此得到更佳的结果。

### 2.3.3 MobileNet

神經網路可以高精度的去識別圖像，但是需要模型參數和大量的計算，不適合用在嵌入式設備上，所以才會提出了 MobileNet，他可以減少參數的數量和計算的成本，主要是使用深度可分離卷積（depthwise-separable convolution），把一個核分成兩個獨立的核，接著去做 Depthwise Convolution 和 Pointwise Convolution（圖 2-5）。

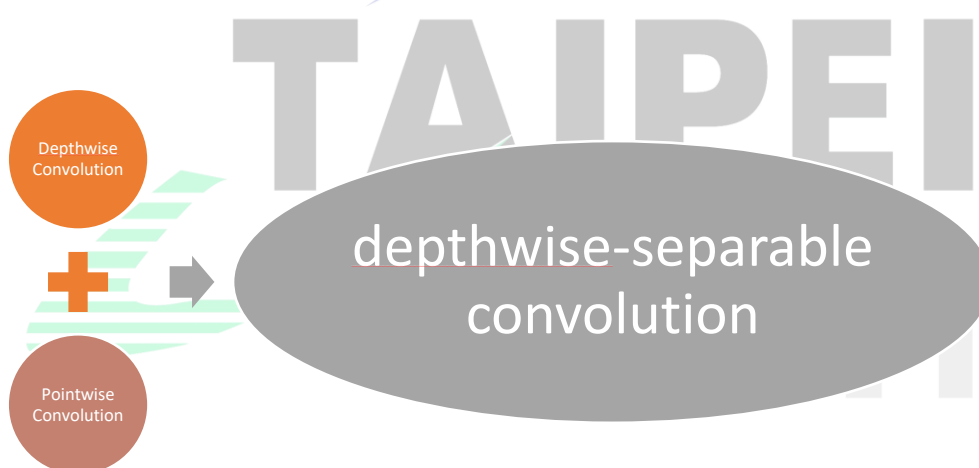


圖 2-5 depthwise-separable convolutions

Depthwise Convolution：

一般 CNN 的深度會根據 channel 大小深度去做變化，假設寬高為 5 的 filter，輸入層 channel 為 3，那 kernel 的維度大小就是  $5 \times 5 \times 3$ ，而 Depthwise Convolution 層則是相反的，先對輸入層 channel 建立一個  $X \times X \times 1$  的 kernel（圖 2-6），接著再對 channel 資料做 convolution 運算，如果有 3 個 channel，就會建立 3 個 kernel，分別對 3 個 channel 資

料做運算處理，得到跟輸出層相同寬高的中間層（圖 2-7）。

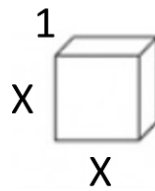


圖 2-6  $X \times X \times 1$  的 kernel

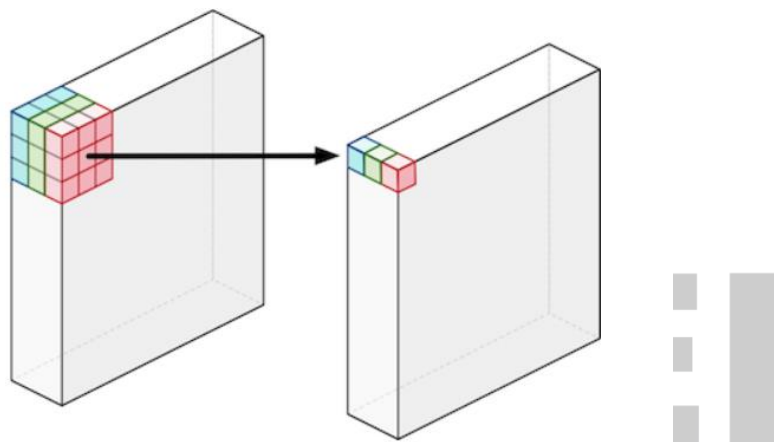


圖 2-7 Depthwise Convolution[11]

Pointwise Convolution :

首先對每個輸出的 channel 建立一個  $1 \times 1 \times X$  的 kernel，接著對輸入層的所有點做 convolution 運算，如果輸出層有  $Y$  個 channel，就會建立  $Y$  個  $1 \times 1 \times X$  的 kernel（圖 2-8），接著將每個 kernel 對輸入層做運算，最後可以得到  $Z \times Z \times Y$  的輸出層（ $Z$  為輸出層的長寬）（圖 2-9）。

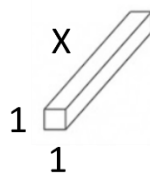


圖 2-8  $1 \times 1 \times X$  的 Kernel

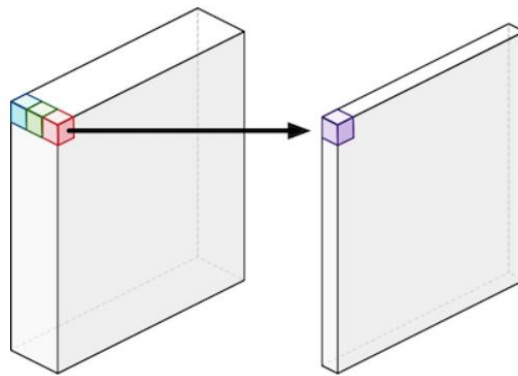


圖 2-9 Pointwise Convolution[11]

### 2.3.4 Xception

Xception 是極端的 Inception，以 InceptionV3 為基礎去改進，使用深度可分離卷積 depthwise-separable convolutions 去代替 Inception module，加入了 ResNet 的殘差學習 (Residual Learning) 方法。

Residual Learning：

$x$  為我們的輸入，層數我們視為  $H(x)$  函數，假設也可以使用  $F(x) = H(x) - x$  來代替，那原來的函數則為  $H(x) = F(x) + x$ ，可看出兩個式子是相等的 (圖 2-10)，但在機器學習上大不相同。

如果多餘的層都為恆等映射，深層模型的表現應該較好，但退化問題指出許多的線性層其實很難去逼近恆等映射，利用殘差學習反而容易讓殘差值接近 0，使整個函數接近恆等映射 (圖 2-11)。

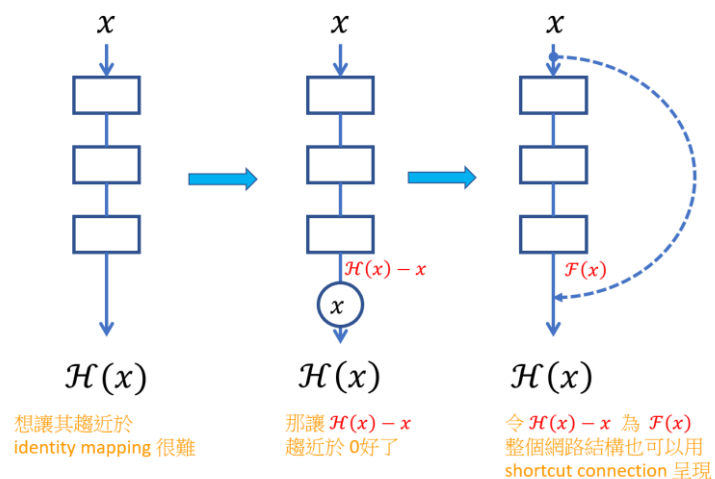


圖 2-10 Residual Learning 概念圖[12]

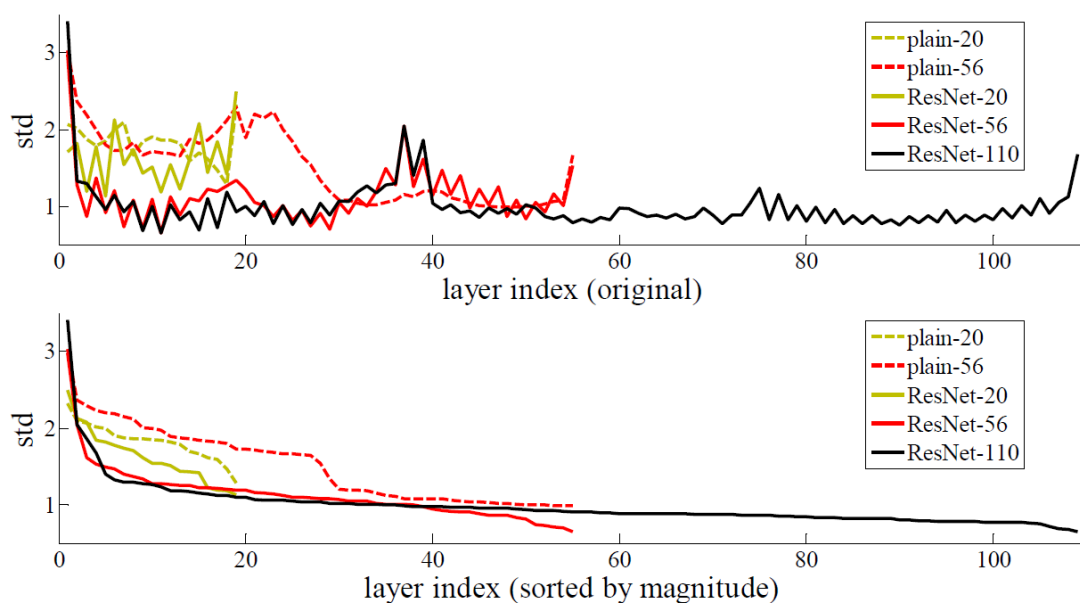


圖 2-11 利用 CIFAR-10 針對各層做 Batch Normalization[12]

### 2.3.5 DenseNet

DenseNet 設計的想法是源自於卷積網路接近輸入的層與輸出的層之間的連接較短，卷積網路可以更深入，也可以更精確有效的訓練。它是使用前饋方式（feed-forward）將

每一層與其他層連接， $L$  層的卷積網路有  $L$  的連接，每一層與其他層之間還有  $L(L+1)/2$  個直接連接，前面每一層都會成為後一層的輸入（圖 2-12）。

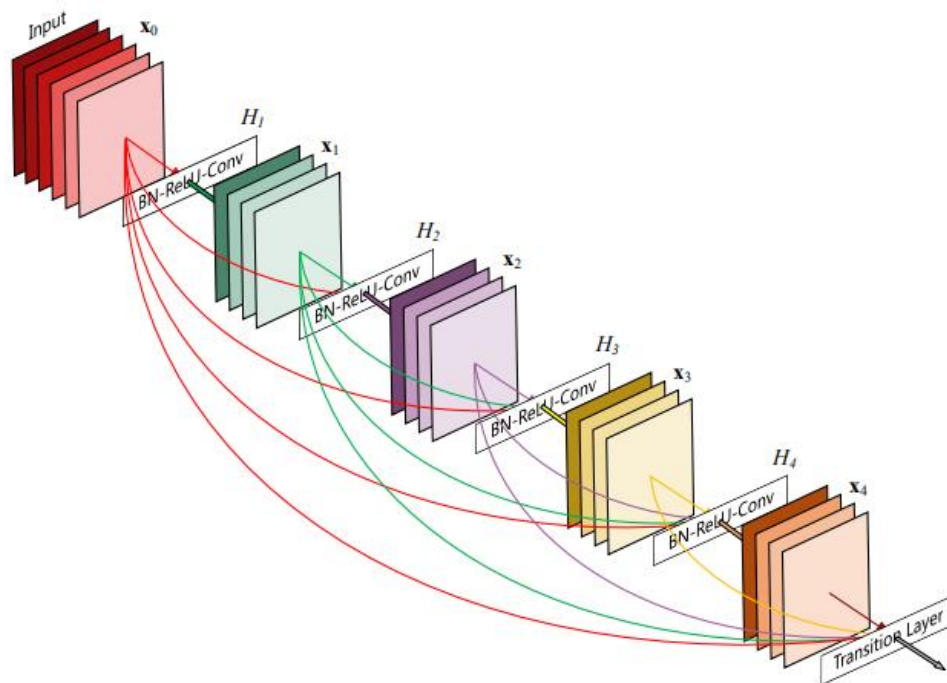


圖 2-12 DenseNet 的前饋方式(feed-forward)[13]

DenseNet 可以緩解梯度消失的問題(vanishing-gradient problem)，提升資訊的傳輸效率，每一層都能直接得到輸入訊號，得以訓練更深的網路，並且加強特徵使用，可以同時減少參數的數量又能提升網路效能。

### 2.3.6 EfficientNet

EfficientNet 是一種更高效的方法，一般來說模型加寬或加深雖然是可行的，但很快就會飽和，當模型引數太多時反而會造成效率不佳，然而 EfficientNet 則是有原則的擴充套件，逐漸地增加，引數不多可以使效率提高，EfficientNet 依靠 AutoML 和復合縮放來實現性能且不影響資源效率。

EfficientNet-B1 到 B7 是由 EfficientNet-B0 縮放方法改進得來的。EfficientNet 和其他模型比較起來相對準確率較高，由圖可以看出，EfficientNet 可以達到高精確度，且模型較小、效能提升（圖 2-13）。

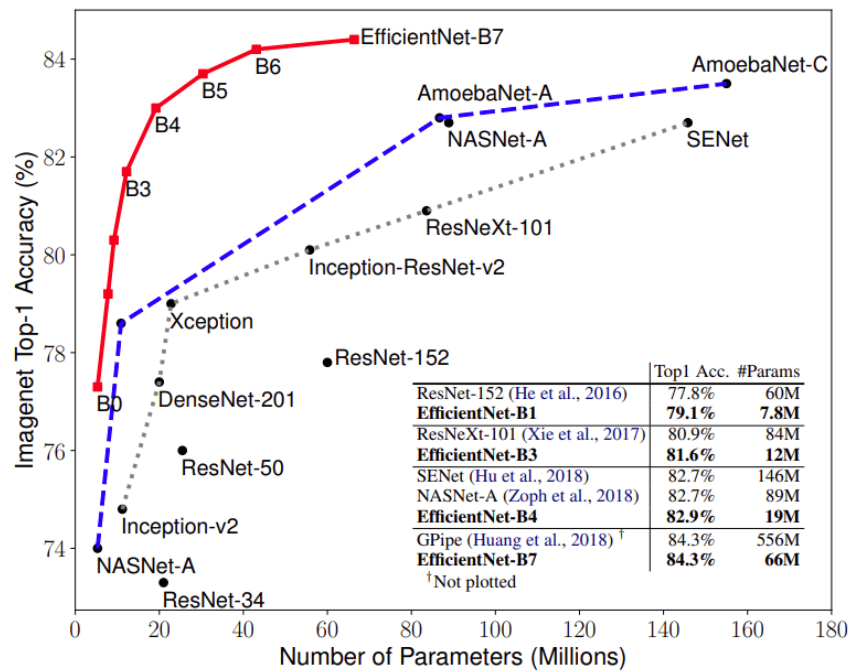


圖 2-13 EfficientNet-B1 到 B7 和其它 CNN 模型的效能比較[14]

## 2.4 食物辨識的應用

食物辨識做完成後，可以有很多的後續應用，像是營養評估、餐點辨識、計價系統和飲食健康管理等等，以下列舉幾個已有的例子。

### 2.4.1 營養評估

文[15]使用深度學習做食物辨識，應用在食物類別識別系統和營養估算系統，先建立了食物影像數據集，接著做標記和影像處理，並建構了殘差卷積神經網絡，並且移除



最大池化層減少信息的損失，最後得到的準確率 Micro-F1 為 83.63% 和 Macro-F1 為 76.90%，此研究改善殘差卷積神經網路比原本的準確率更高，最後研究衛生福利部收集食物營養素訊息，計算每種食物在含有多少營養素。

這部分適合使用在各式各樣的人身上，可以根據營養素訊息去做最佳的調整，一些病人可能缺乏某些部分的營養，如果人們不知道什麼食物有哪些營養素，則可以根據辨識營養素系統，辨識完後再給病人攝取這些食物，又或者某些病人不適合攝取糖類、蛋白質等等，也可以經過識別後將這些食物移除再給病人食用，以避免攝取不該攝取的食食物導致過量，另外健康的人也可以使用營養素辨識去做飲食調整，使大家都能更加健康。

### 2.4.2 餐點辨識

到世界各地旅行，最重要的一件事就是「吃」了，如果到了國外看不懂菜單上的字，那可能會造成點餐困難，要吃什麼也沒辦法點，或者只能隨便比手畫腳的點餐，結果上餐時不是你想吃的，文[16]就是針對這些問題所做的研究，文設計並開發出一套可以透過截取特徵點辨識食物以及辨識中文菜單上文字的系統，食物辨識的部分使用了四個特徵法：Gabor 濾波器、局部二元模式 (LBP)、尺度不變特徵轉換 (SIFT) 和稀疏編碼 (Sparse Coding)，再對每種特徵去做整合，把弱分類整合成強分類，另外還建立了 67 種食物的資料庫，每種食物圖片各 100 張，並且加上 Google 的 Tesseract 光學字元辨識 (OCR) 去整合一套菜單辨識系統。整套系統也可以使用在本研究，協助外國人來台灣時也可以輕鬆點餐，同時也可以了解台灣的各種小吃。

### 2.4.3 計價系統

文[17]主要針對自助餐的計價方式所做的，因為自助餐的份量跟種類差異，使用人眼很難去評估準確的價錢，容易造成紛爭，使用點餐系統計價的話就不會有此問題了，

可以得到快速且準確的計價，在影像處理上分為兩個部分，第一部分使用傳統的物件分割，分出前景和背景，接著將前景分為多個物件去編號做處理，第二部分是使用現代影像分格，直接將影像分為大小均勻的格子，接著將格子進行處理，此研究使用 310 張自助餐影像，研究結果正確分類率為 93%、分格區塊準確分類率為 92%，且確實能準確的辨識餐盤中的食物種類和份量。這套系統若使用在本研究可以提升點餐的效率，也可以降低人力成本，使用於一些小吃的部分的話可以提前做好，客人到了就直接拿取在做計價的動作，在計價時也可以知道是什麼食物多少價錢，也可以避免排隊花費很多時間。

文[19]也是對自助餐進行辨識並計價的系統，是採用 YOLOv3 模型來製作，YOLOv3 只需要一個步驟即可預測框及種類，原理是先將圖片切成 13x13、26x26、52x52 三種尺寸，然後分別在切出來的這些格子中進行辨識，此研究有 575 張圖片，訓練前要將照片中每種菜色的座標框出來[18]（圖 2-14），接著再進行訓練及測試，最後再加上計價系統來完成整個自助餐菜色自動辨識結帳系統（圖 2-15）。



圖 2-14 將照片中每種菜色的座標框出來 [19]

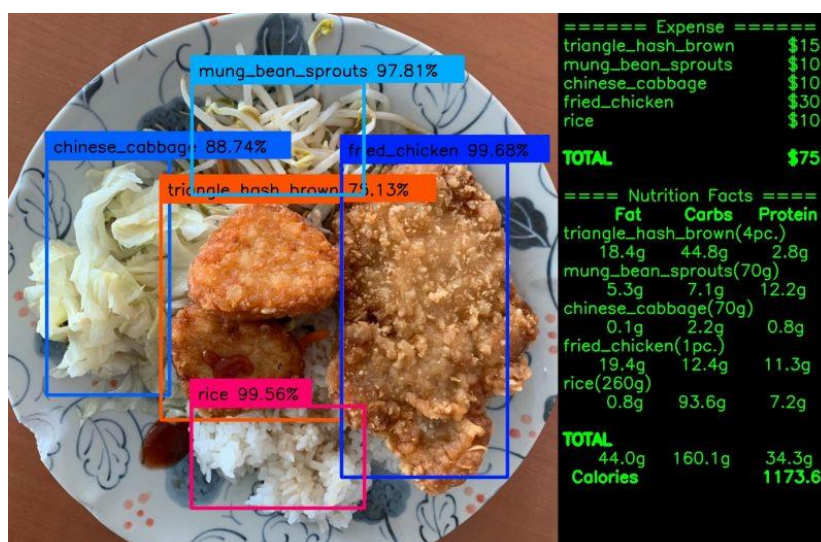


圖 2-15 自助餐菜色自動辨識結帳系統 [19]

### 2.4.3 飲食健康管理

由於現在人們有了健康生活的意識，許多人開始使用 APP 紀錄生活飲食，但許多 APP 只能手動輸入食物與熱量、或者可以辨識食物缺乏管理功能、又或者能管理但無法辨識，此文[19]就是針對這個問題所做的研究，此研究使用 RESTful web service 和 No-SQL database 來讓系統更加有彈性且方便，建立友善的使用者介面，可以使用電腦視覺讀取食物相關訊息，也可以手動輸入訊息，並加上資料庫系統，讓大家都可以去擴充食物資料庫，這樣一來大家都可以透過這套系統紀錄並監控飲食了。這系統可以使用在 Taiwanese Food 101 上，一方面可以記錄飲食，另一方面也可以做成美食地圖，讓來台灣的遊客可以去解鎖這些美食，也可以增加一些樂趣。

## Chapter 3. 研究方法

本實驗使用各種模型來訓練 Taiwanese Food 101 數據集，比較各種訓練模型，最後得到訓練的最佳結果（圖 3-1）。

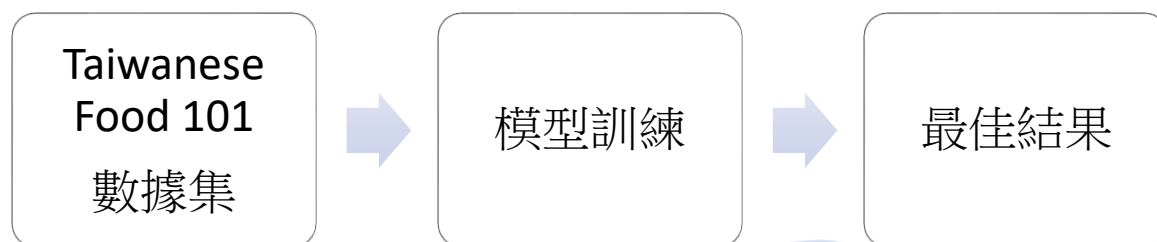


圖 3-1 研究方法.

### 3.1 Taiwanese Food 101 數據集

本實驗使用學長之前所整理的 Taiwanese Food 101 資料庫（圖 3-2），資料庫中的食物皆為台灣的小吃美食，各種飯類、麵類、還有各式小菜、湯、飲料、小吃、甜點等等（圖 3-3），總共有 101 種食物，每一種食物各 200 張，為了提高辨識率，在本實驗又增加了許多圖片，每種食物圖片增加 50 到 100 張不等，去進行深度學習的訓練。



圖 3-2 Taiwanese Food 101 資料庫



圖 3-3 Taiwanese Food 101 資料庫中的食物圖片



## 3.2 模型訓練

本研究使用了以上介紹過的模型 VGG、Inception、MobileNet、Xception、DenseNet 以及 EfficientNet 來做訓練（圖 3-4），首先定義基本函數，調整圖片達到適合模型訓練使用，接著將模型下載，由於每一類的圖片數量不一致，這裡做了一個統一的處理，將資料庫中每類隨機取出同樣張數的圖片來使用（圖 3-5），接著將取出的圖片分為訓練、測試和驗證三部分。

<input type="checkbox"/>	 DenseNet121-taiwanese food-75.99.h5
<input type="checkbox"/>	 EfficientNetB3-taiwanese food-86.11.h5
<input type="checkbox"/>	 InceptionV3-taiwanese food-17.47.h5
<input type="checkbox"/>	 MobileNet-taiwanese food-71.03.h5
<input type="checkbox"/>	 VGG16-taiwanese food-4.27.h5
<input type="checkbox"/>	 VGG19-taiwanese food-2.48.h5
<input type="checkbox"/>	 Xception-taiwanese food-81.05.h5

圖 3-4 使用的模型

```
working_dir = r'./'  
img_size=(200,300)  
max_samples=140  
min_samples=0  
column='labels'  
train_df=trim(train_df, max_samples, min_samples, column)
```

圖 3-5 統一圖片張數

下一步進行訓練，選擇其中一種模型先進行訓練，每一種模型訓練 40 次，但由於訓練時間消耗較長，進行五次設定一次暫停（圖 3-6），看是否要繼續執行，接著在測試集上評估模型並將模型保存（圖 3-7），最後對測試集進行預測並生成混淆矩陣以及測試分類報告（圖 3-8）。

```

1 / 40   8.119   42.518   5.01284   67.428   0.00100   0.00100   accuracy   0.00   5802.66
2 / 40   3.634   76.450   2.71008   77.954   0.00100   0.00100   accuracy   79.81   5825.24
3 / 40   1.981   86.796   1.91917   79.245   0.00100   0.00100   accuracy   13.53   5826.66
4 / 40   1.291   92.793   1.61135   80.933   0.00100   0.00100   val_loss   16.04   5804.06
5 / 40   0.973   96.047   1.39742   84.111   0.00100   0.00100   val_loss   13.28   5778.53
enter H to halt training or an integer for number of epochs to run then ask again
h
training has been halted at epoch 5 due to user input
training is completed - model is set with weights from epoch 5
[training elapsed time was 10.0 hours, 27.0 minutes, 46.02 seconds]

```

圖 3-6 模型訓練 Taiwanese Food 101 過程

```

accuracy on the test set is 81.05 %
model was saved as ./Xception-taiwanese food-81.05.h5
class csv file was saved as ./class_dict.csv

```

圖 3-7 測試集上評估模型並保存

	precision	recall	f1-score	support
bawan	0.94	0.89	0.91	18
beef_noodles	1.00	1.00	1.00	11
beef_soup	0.91	0.91	0.91	11
bitter_melon_with_salteds_eggs	1.00	1.00	1.00	10
braised_napa_cabbage	0.62	0.50	0.56	10
braised_pork_over_rice	0.78	0.88	0.82	8
brown_sugar_cake	0.90	0.90	0.90	10
bubble_tea	1.00	0.82	0.90	11
caozaiguo	0.88	0.70	0.78	10
chicken_mushroom_soup	1.00	0.88	0.93	8
chinese_pickled_cucumber	1.00	1.00	1.00	10
coffin_toast	0.85	1.00	0.92	11
cold_noodles	0.56	0.90	0.69	10
crab_migao	0.91	1.00	0.95	10

圖 3-8 對測試集進行預測並生成混淆矩陣和分類報告

				precision	recall	f1-score	support
			bawan	0.94	0.89	0.91	18
			beef_noodles	1.00	1.00	1.00	11
			beef_soup	0.91	0.91	0.91	11
			bitter_melon_with_salted_eggs	1.00	1.00	1.00	10
			braised_napa_cabbage	0.62	0.50	0.56	10
			braised_pork_over_rice	0.78	0.88	0.82	8
			brown_sugar_cake	0.90	0.90	0.90	10
			bubble_tea	1.00	0.82	0.90	11
			caozaiquo	0.88	0.70	0.78	10
			chicken_mushroom_soup	1.00	0.88	0.93	8
			chinese_pickled_cucumber	1.00	1.00	1.00	10
			coffin_toast	0.85	1.00	0.92	11
			cold_noodles	0.56	0.90	0.69	10
			crab_migao	0.91	1.00	0.95	10
			deep-fried_chicken_cutlets	0.78	0.70	0.74	10
			deep_fried_pork_rib_and_radish_soup	0.75	0.90	0.82	10
			dried_shredded_squid	1.00	0.56	0.71	9
			egg_pancake_roll	0.89	0.80	0.84	10
			eight_treasure_shaved_ice	0.83	1.00	0.91	10
			fish_head_casserole	0.50	0.45	0.48	11
			fried-spanish_mackerel_thick_soup	0.69	0.82	0.75	11
			fried_eel_noodles	0.90	1.00	0.95	9
			fried_instant_noodles	0.73	0.80	0.76	10
			fried_rice_noodles	1.00	0.92	0.96	12
			ginger_duck_stew	0.80	0.62	0.70	13
			grilled_corn	0.83	1.00	0.91	10
			grilled_taiwanese_sausage	0.80	0.80	0.80	10



hakka_stir-fried	0.70	0.64	0.67	11
hot_sour_soup	0.69	0.90	0.78	10
hung_rui_chen_sandwich	0.80	0.80	0.80	10
intestine_and_oyster_vermicelli	0.85	1.00	0.92	11
iron_egg	1.00	1.00	1.00	10
jelly_of_gravey_and_chicken_feet_skin	0.83	0.83	0.83	12
jerky	1.00	1.00	1.00	10
kung-pao_chicken	0.64	0.88	0.74	8
luwei	1.00	0.75	0.86	8
mango_shaved_ice	0.85	1.00	0.92	11
meat_dumpling_in_chili_oil	0.67	1.00	0.80	8
milkfish_belly_congee	0.78	0.70	0.74	10
mochi	0.75	0.90	0.82	10
mung_bean_smoothie_milk	1.00	0.90	0.95	10
mutton_fried_noodles	0.78	0.70	0.74	10
mutton_hot_pot	0.54	0.70	0.61	10
nabeyaki_egg_noodles	1.00	0.70	0.82	10
night_market_steak	1.00	1.00	1.00	11
nougat	0.94	1.00	0.97	15
oyster_fritter	1.00	1.00	1.00	8
oyster_omelet	0.73	1.00	0.85	11
papaya_milk	0.75	1.00	0.86	9
peanut_brittle	0.88	0.88	0.88	8
pepper_pork_bun	0.62	0.80	0.70	10
pig_s_blood_soup	0.80	1.00	0.89	8
pineapple_cake	0.80	0.89	0.84	9
pork_intestines_fire_pot	0.73	0.80	0.76	10
potsticker	0.70	0.78	0.74	9
preserved_egg_tofu	1.00	0.62	0.77	8
rice_dumpling	1.00	0.64	0.78	11

rice_noodles_with_squid	1.00	0.73	0.84	11
rice_with_soy-stewed_pork	0.90	0.90	0.90	10
roasted_sweet_potato	1.00	0.60	0.75	10
sailfish_stick	0.67	0.75	0.71	8
salty_fried_chicken_nuggets	0.83	1.00	0.91	10
sanxia_golden_croissants	0.90	0.90	0.90	10
saute_spring_onion_with_beef	0.86	0.67	0.75	9
scallion_pancake	0.71	0.56	0.63	9
scrambled_eggs_with_shrimp	0.90	0.90	0.90	10
scrambled_eggs_with_tomatoes	1.00	0.78	0.88	9
seafood_congee	0.75	0.60	0.67	10
sesame_oil_chicken_soup	0.80	0.40	0.53	10
shrimp_rice	1.00	1.00	1.00	9
sishen_soup	0.88	0.88	0.88	8
sliced_pork_bun	1.00	0.73	0.84	11
spicy_duck_blood	0.80	0.50	0.62	8
steam-fried_bun	0.55	0.75	0.63	8
steamed_cod_fish_with_crispy_bean	0.91	1.00	0.95	10
steamed_taro_cake	0.80	0.80	0.80	10
stewed_pig_s_knuckles	1.00	0.83	0.91	12
stinky_tofu	0.75	0.60	0.67	10
stir-fried_calamari_broth	0.62	1.00	0.76	8
stir-fried_duck_meat_broth	0.90	0.75	0.82	12
stir-fried_loofah_with_clam	1.00	1.00	1.00	10
stir-fried_pork_intestine_with_ginger	0.88	0.78	0.82	9
stir_fried_clams_with_basil	1.00	0.75	0.86	8
sugar_coated_sweet_potato	1.00	1.00	1.00	7
sun_cake	0.69	1.00	0.82	9
sweet_and_sour_pork_ribs	0.82	0.90	0.86	10
sweet_potato_ball	0.82	1.00	0.90	9

	taiwanese_burrito	0.75	0.90	0.82	10
	taiwanese_pork_ball_soup	0.90	1.00	0.95	9
	taiwanese_sausage_in_rice_bun	0.89	0.89	0.89	9
	tanghulu	1.00	1.00	1.00	8
	tangyuan	0.91	1.00	0.95	10
	taro_ball	1.00	0.88	0.93	8
	three-cup_chicken	0.67	0.75	0.71	8
	tube-shaped_migao	1.00	0.70	0.82	10
	turkey_rice	1.00	0.92	0.96	12
	turnip_cake	0.67	0.60	0.63	10
	twist_roll	0.90	0.90	0.90	10
	wheel_pie	0.80	0.86	0.83	14
	xiaolongbao	1.00	0.88	0.94	17
	yolk_pastry	1.00	0.80	0.89	10
	accuracy			0.84	1008
	macro avg	0.85	0.84	0.83	1008
	weighted avg	0.85	0.84	0.84	1008

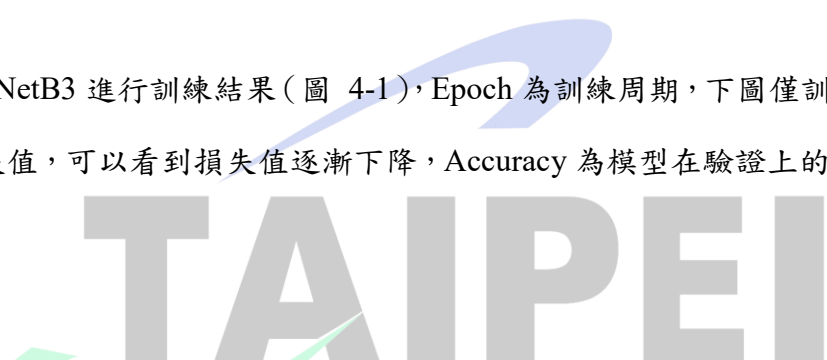
圖 3-9 Taiwanese Food 101 混淆矩陣和分類報告

## Chapter 4. 研究結果與分析

本章主要對研究結果進行介紹、分析與比較，介紹我們的訓練過程，對於我們訓練後適合的模型進行測試，最後選取辨識率高於 70% 的模型進行分析與比較，準確率分別為：MobileNet-71.03%、DenseNet121-75.99%、Xception-81.05%、EfficientNetB3-85.1%，EfficientNetB3 為最佳的實驗結果。

### 4.1 研究結果

使用 EfficientNetB3 進行訓練結果 (圖 4-1)，Epoch 為訓練周期，下圖僅訓練 5 次，Loss 為訓練的損失值，可以看到損失值逐漸下降，Accuracy 為模型在驗證上的準確率，準確率逐漸上升。



Epoch	Loss	Accuracy	V_loss	V_acc	LR	Next LR	Monitor	% Improv	Duration
1 /40	8.291	34.286	4.99523	68.620	0.00100	0.00100	accuracy	0.00	5965.09
2 /40	3.916	70.290	2.70371	78.947	0.00100	0.00100	accuracy	105.01	6211.92
3 /40	2.259	82.030	1.91000	81.132	0.00100	0.00100	accuracy	16.70	5584.77
4 /40	1.577	88.239	1.48109	85.104	0.00100	0.00100	accuracy	7.57	5580.12
5 /40	1.211	92.383	1.32979	85.104	0.00100	0.00100	val_loss	10.22	5587.15


圖 4-1 訓練 5 次得到的數據值

在訓練後，得到了確定能使用的基礎模型，在有了基礎模型的情況下進行進一步的模型評估，(圖 4-2) 為在測試集上評估模型，測試準確率為 85.1%，在模型評估後將模型進行保存，保存好的模型可用在後續的應用。

```
34/34 [=====] - 66s 2s/step - loss: 1.3298 - accuracy: 0.8510
accuracy on the test set is 85.10 %
model was saved as ./EfficientNetB3-taiwanese food-85.10.h5
class csv file was saved as ./class_dict.csv
```

圖 4-2 在測試集上評估並保存模型.

接著在測試集上進行預測（圖 4-3），precision 為我們預測的精度，可以看出 bawan 為 0.94 稍有錯誤，beef\_noodles 預測結果為 1 無錯誤，下圖僅列出其中 14 種食物，數值好看食物在測試集上預測的準確度的高低。



	precision	recall	f1-score	support
bawan	0.94	0.89	0.91	18
beef_noodles	1.00	1.00	1.00	11
beef_soup	0.91	0.91	0.91	11
bitter_melon_with_salteds_eggs	1.00	1.00	1.00	10
braised_napa_cabbage	0.62	0.50	0.56	10
braised_pork_over_rice	0.78	0.88	0.82	8
brown_sugar_cake	0.90	0.90	0.90	10
bubble_tea	1.00	0.82	0.90	11
caozhaiguo	0.88	0.70	0.78	10
chicken_mushroom_soup	1.00	0.88	0.93	8
chinese_pickled_cucumber	1.00	1.00	1.00	10
coffin_toast	0.85	1.00	0.92	11
cold_noodles	0.56	0.90	0.69	10
crab_migao	0.91	1.00	0.95	10

圖 4-3 在測試集上預測的準確度

下圖將數值轉為直條圖（圖 4-4）將所有食物種類的錯誤圖片數量列出來，方便看錯誤的圖片種類及數量，得知 101 種食物中有 30 種食物 0 錯誤，另外 71 種錯誤圖片數量分別為下圖。

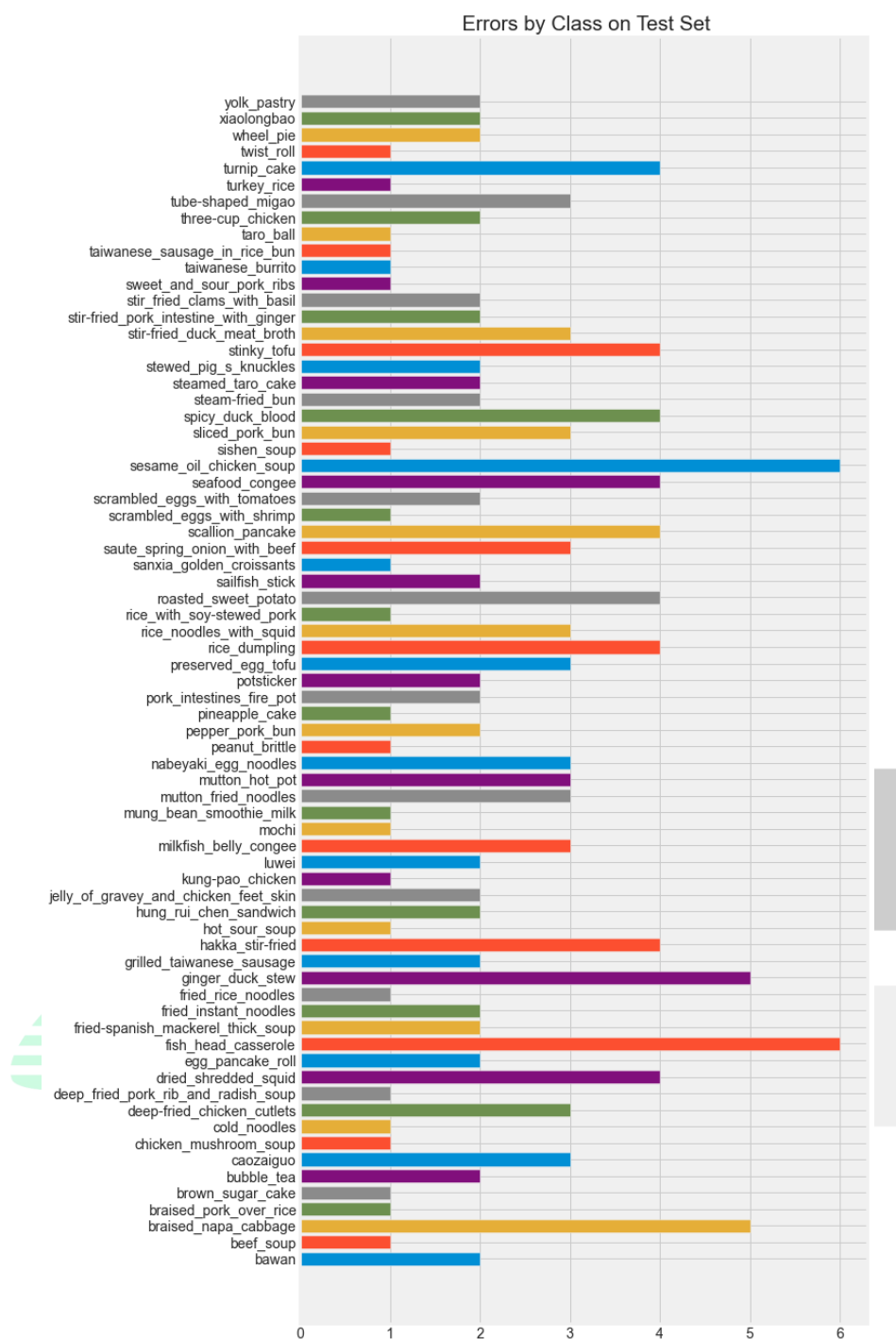


圖 4-4 在測試集上的錯誤分類數量

## 4.2 模型訓練結果比較

將所有模型進行訓練後，做出比較表格，挑出較新較好的 4 種模型進行比較，模型

準確率分別為：MobileNet-71.03%、Xception-81.05%、DenseNet121-75.99%、EfficientNetB3-85.1%，可以得知 EfficientNetB3 是所有模型中效果最好的，Xception 也是較高的，其它的準確率則未達 80%（表 4-1）。

表 4-1 在各模型上的測試準確率

Model	accuracy
MobileNet	71.03%
Xception	81.05%
DenseNet121	75.99%
EfficientNetB3	85.1%

然而 MobileNet、Xception 和 DenseNet121 的結果又比 VGG16、VGG19 好，應該是因為 VGG 是屬於比較早期的模型，MobileNet、Xception、DenseNet121 和 EfficientNetB3 是較新的模型（圖 4-5），不管是在解析度方面或是技術方面都有更多的進步。

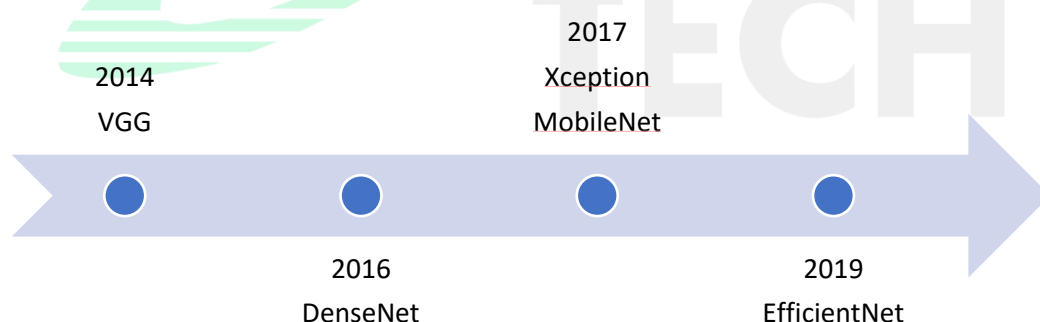


圖 4-5 模型年代

## Chapter 5. 結論與建議

本章結論得出 EfficientNetB3 為本研究最佳的預測結果，也是本論文的主題，另外提出了對於 Taiwanese Food 101 的貢獻以及研究上所遇到的種種限制，最後提出建議，希望對 Taiwanese Food 101 研究有興趣的其他朋友有所幫助。

### 5.1 結論

由研究結果得知 EfficientNetB3 的準確率高達 85.1%，與其它模型相比較高，MobileNet、Xception 和 DenseNet121 的結果又比 VGG16、VGG19 好，新的模型使用在 Taiwanese Food 101 上有較佳的實驗結果，然而在一些限制之下目前還無法突破 90% 的準確率，但跟前期剛開始相較來說已經有了大幅度的提升。

### 5.2 貢獻

本研究主要增加了 Taiwanese Food 101 資料庫的豐富度，並將資料庫公開於 Kaggle 提供給大家做使用，以及對 Taiwanese Food 101 使用各種模型訓練進行比對，得到的最終的最佳測試結果。

研究前期，使用了許多不同的方法去進行訓練以及辨識，訓練的辨識率結果大概都介於 60% 到 70%，後來將圖片數量做統一隨機抽取相同數量進行訓練，不斷修改程式去訓練並保存模型，過程中也不斷的去嘗試更換新的模型，有些不適用的模型在前期篩選掉的，VGG 在這也是屬於較不適用的模型，由於現在越來越多新模型的出現，舊的雖然不至於太差，但大家還是比較會選擇使用新的模型，使用上來較容易，要調整的數值也



相對較少，不過一些參數會因為圖片或者辨識內容要去做不同的調整或者修改。

此辨識率高達 85.1%，研究結果可以使用於無人商店、自助餐計價、以及一些健康飲食熱量記錄等等，這些對於我們來說不僅可以使生活更加便利，也可以減少人力成本，是現代不可或缺的技术。

### 5.3 研究限制

由於台灣的小吃圖片並不是特別多，訓練的圖片數量有限，在訓練的過程中只能更換訓練方法，或者使用不同的模型來提升準確度，如果圖片數量可以再增加，應該有機會可以再提高準確率，另外有些圖片的解析度可能不太夠或者拍的不是很清楚，增加了辨識的難度，如果將 Taiwanese Food 101 資料庫增加圖片並加以挑選，準確度應該也可以有所提升，我覺得最大的一個問題還是在食物本身上，有些食物是我們使用肉眼也難以分辨的食物，更何況是機器，一些很相似的食物、或者不同拍攝角度、又或者同一種食物多種做法，顏色不一樣或者料理方式不同，食物本身呈現出來的樣子也不同（圖 5-1）。



圖 5-1 一種食物呈現多種樣貌

## 參考資料

- [1] Marwa Qaraqe , Muhammad Usman , Kashif Ahmad , Amir Sohail ,and Ali Boyaci,  
“Automatic Food Recognition System for Middle-Eastern Cuisines, ” in Proc. Institution  
of Engineering and Technology (IET), vol. 14, pp. 2469-2479, Oct. 2020.
- [2] Narit Hnoohom and Sumeth Yuenyong,Thai, “Fast Food Image Classification Using  
Deep Learning, ” in Proc. IEEE, pp. 116-119, Feb. 2018.
- [3] Ecole Polytechnique Fédérale de Lausanne (EPFL), “Food Image Dataset, Food-5K,  
Food-11 ,” Please refer to web site: (<https://www.epfl.ch/labs/mmspg/downloads/food-image-datasets/>).
- [4] Alam Siddiquee, Kazy Noor E.Islam, Md Shabiul Dowla, Mohammad Yasin Ud Rezaul,  
Karim Mohammed, and Grout, Vic, “Detection, Quantification and Classification of  
Ripened Tomatoes: a Comparative Analysis of Image Processing and Machine Learning,”  
in Proc. IET Image Processing, Vol. 14, pp. 2442-2456, Sep. 2020.
- [5] Kh Tohidul Islam, Masud Pervez, Sudanthi Wijewickrema, and Stephen O'Leary, “An  
Exploration of Deep Transfer Learning for Food Image Classification, ” in Proc. IEEE, pp.  
1-5, Dec. 2018.
- [6] Alex Krizhevsky, Ilya Sutskever, Geoffrey Everest Hinton, AlexNet, please refer to web  
site: (<https://zh.wikipedia.org/zh-tw/AlexNet>).
- [7] ILSVRC (ImageNet Large Scale Visual Recognition Competition),GoogLeNet , please  
refer to web site : (<https://medium.com/image-processing-and-ml-note/inception-v1-googlenet-winner-of-ilsvrc-2014-image-classification-15b1ea62cc11>)
- [8] ResNet and ResNetV2 please refer to web site: (<https://keras.io/api/applications/resnet/>)
- [9] Jcjohnson, cnn-benchmarks please refer to web site: (<https://github.com/jcjohnson/cnn-benchmarks>)

- [10] 李馨伊, Inception, please refer to web site: (<https://medium.com/ching-i/inception-%E7%B3%BB%E5%88%97-inceptionv2-inceptionv3-93cd42054d23>)
- [11] Cheng-Shiang Li, Depthwise Separable Convolution, please refer to web site: (<https://blog.yeshuanova.com/categories/deep-learning/>)
- [12] Allen Tzeng, Deep Residual Learning for Image Recognition, Convolutional Neural Network
- [13] Gao Huang, Zhuang Liu, Laurens van der Maaten & Kilian Q. Weinberger, Densely Connected Convolutional Networks, Densely Connected Convolutional Networks (CVPR 2017 Best Paper Award)
- [14] Vardan Agarwal, EfficientNet, please refer to web site: (<https://iter01.com/533367.html>)
- [15] Jia-HongChen, Food Image Recognition and Nutrition Estimation via Deep Learning
- [16] Yi-Ting (Evelyn) Tsai, “Cloud-Based Mobile Platform for Chinese Food Identification and Menu Character Recognition,” in Master Thesis, pp.1-60, Jul. 2014
- [17] Jiang, Yu-Xian, “Automated Pricing System for Parity Cafeteria based on Image Recognition,” in Master Thesis, pp.1-88, Feb. 2020.
- [18] AndyWu, 採用 YOLOv3 模型的自助餐菜色自動辨識結帳系統:  
<https://reurl.cc/RrNDdD>
- [19] Lo, Wei-Lin, “IoT based Food Consumption Monitoring and Analysis in E-health Application,” in Master Thesis, pp.1-60, Feb. 2019.