

應用影像結合卷積神經網路於柳丁分級之研究

1 蕭偉泓, 2 陳志明

1 嶺東科技大學

E-mail: drum831@gmail.com

2 嶺東科技大學

E-mail: jhiehming@gmail.com

摘要

利用水果的表面特徵來預測其內在品質，例如甜度等，一直是水果自動分級面臨的關鍵課題。本研究提出一種結合影像分析和卷積神經網路的 CNN-Orange 模型，用於預測柳丁的內在品質。研究結果顯示，將柳丁的品質分為三個等級的情形下，CNN-Orange 模型的測試準確率可達 93%，證明本方法的有效與實用性。

關鍵字: CNN 卷積神經網路; 影像辨識; 深度學習。

1. 前言

行政院農委會在新農業創新推動 2.0 裡提到，為了落實科研發展機制，導入或研發自動化為推動智慧農業的重要方向[1]。為了落實農委會的上述政策，本研究針對柳丁作為研究的對象，利用影像結合深度學習人工智慧的技術，進行柳丁甜度自動分級的預測研究，解決長久以來面臨無法利用外觀表面特徵預測內在品質的關鍵課題。

柳丁是台灣本地盛產的水果，深受大家的喜愛，也是台灣水果外銷的重要項目之一。由於水果的銷售通常會依照水果的大小、重量、顏色、甜度、酸鹼度等來進行分級，不同等級銷售價格也不會相同。尤其，大部分的人食用喜好並不相同。有些人喜歡吃甜度較高的柳丁，但也有些人不喜歡；又有一些人是不能吃太甜的柳丁，例如說糖尿病的病人。Villa 等人[2]提出糖尿病的特徵是由於胰島素分泌、作用或兩者缺陷導致的高血糖症。這一類糖尿病患者不能攝取高甜度食品，會導致血糖劇烈起伏，且影響身體健康，所以飲食上必須特別注意甜度的攝取。因此，如何進行水果的內在品質分級一直是非常重要的研究課題。

但是柳丁的內在品質，通常不容易經過外在的表面特徵直接來進行預測，傳統方法只能透過實際試吃或破壞性甜度檢測才能知曉。但受到新冠疫情的影響，已經無法現場試吃；且檢測儀器價格也很昂貴

不親民。所以如何找到內在品質的關鍵特徵與量化指標，才能夠作為分級或購買時的參考。

傳統的水果分級研究大多使用影像辨識進行外觀特性的辨識[3]，例如顏色和大小。在影像辨識之前，需要將收集到的水果影像資料進行分割處理，利用大津(OTSU)演算法將背景或是其他雜物影像去除，使水果外觀特徵能夠被電腦精確的識別[4]。也有一些研究是使用機器學習去進行水果的甜度預測[5]，藉由特徵的分析與選取來優化模型的參數。但是，使用深度學習並搭配影像去預測水果的內在品質的研究目前尚不多見。

另外，Bejo 等人[6]提出一種非破壞性甜度檢測的方法，利用傳統影像的 RGB 及 HSB 色彩空間來分析 Chokanan 芒果色調與甜度之間的關係。Supekar 等人[7]提出一種基於芒果的成熟度、大小、形狀和缺陷等，應用圖像處理技術提取顏色、幾何和形狀等相關特徵，並使用隨機森林分類器預測芒果成熟度、大小和形狀類別，K-Means 聚類應用於預測芒果缺陷類別。Das 等人[8]提出一種使用卷積神經網路(CNN)的自動番茄成熟度分類系統，將番茄成熟度分為三類(綠色、紅色和黃色)，來達到預測番茄成熟度的目的。Sangsongf 等人[9]提出使用非破壞性檢測的方式，通過 CNN 卷積神經網路從圖像中預測鳳梨的甜度。Azman 等人[10]提出一種使用 CNN 卷積神經網路的方法，將鳳梨依照成熟度分為三個等級(成熟、部分成熟和未成熟)。上述的研究成果已經針對包括芒果、鳳梨和番茄等標的，利用影像或深度學習的技術進行探討並證明其有效性。但是，這些研究成果中還沒有人針對柳丁的內在品質進行研究探討。

綜合上面的文獻探討，我們可以得到一個簡單的結論：利用影像結合人工智慧(Artificial Intelligent, AI)對水果進行甜度或是成熟度的品質識別，是最近比較新穎且具可行性的研究方法。因此，本研究提出一種利用影像分析結合卷積神經網路的方法，CNN-Orange。首先透過實驗收集柳丁的影像資料及糖度數據，然後建構 CNN 卷積神經網路模型，提取影像特徵並訓練模型，驗證後的模型預測結果可以作為柳丁自動分級之依據。

本文後續章節安排如下：第 2 節說明本研究的對象、分級準則以及數據收集方法。第 3 節討論影像的前處理以及 CNN 卷積神經網路模型等方法及其架構。第 4 節討論研究的結果及分析。第 5 節是結論和後續的工作。

2. 實驗材料

2.1. 分級準則

分級準則參考現行規範與專業農民之意見後得出，將柳丁依據糖度分為 A、B 和 C 三個等級。其中糖度 12 Brix 以上為 Level A、糖度 9~12 Brix 為 Level B、糖度 9 Brix 以下為 Level C，如表 1 所示。

表 1: 柳丁的糖度分級表

分級	糖度
Level A	12 Brix 以上
Level B	9~12 Brix
Level C	9 Brix 以下

2.2. 柳丁糖度量測

柳丁的甜度資料通常利用糖度計來量測，本研究使用的是市售 PAL-alpha 糖度計，相關規格如圖 1 所示。量測過程首先將柳丁切一半，並使用柳丁榨汁機將柳丁打成汁，然後將柳丁液體的部分滴在糖度計上測量糖度值。



圖 1: PAL-alpha 糖度計

型號	PAL-alpha
測量範圍	Brix 0.0...85.0%
解析度	Brix 0.1%
精密度	Brix $\pm 0.2\%$
溫度補償	10...100°C 自動溫度補償
樣本需求	0.3ml
反應時間	3 sec
防護等級	IP65

2.3. 外觀影像數據資料收集

本研究在 2021 年 10 月至 2021 年 11 月期間從雲林古坑果園中，共採收了 286 顆柳丁。並將每顆柳丁依其外在的三個特徵，蒂頭、果身和果底，分別拍攝 1 張、2 張和 1 張共四張的影像照片，作為影像特徵數據集，如圖 2 至圖 4 所示。本實驗收集的柳丁影像數據集共 1144 張，分別依其糖度值以人工標註為

A、B 和 C 三個分類。其中 183 顆柳丁共 732 張影像用於模型訓練，46 顆柳丁 184 張影像用於模型驗證以及 57 顆柳丁 228 張影像用於測試，詳細如表 2 所示。



圖 2: 柳丁蒂頭照片



圖 3: 柳丁果身照片



圖 4: 柳丁果底照片

表 2: 影像數據集分為訓練、驗證和測試集

分級	訓練	驗證	測試
Level A	200 張 (50 顆)	48 張 (24 顆)	76 張 (19 顆)
Level B	264 張 (66 顆)	68 張 (17 顆)	76 張 (19 顆)
Level C	268 張 (67 顆)	68 張 (17 顆)	76 張 (19 顆)
合計	732 張 (183 顆)	184 張 (46 顆)	228 張 (57 顆)

3. 研究方法

3.1. 軟硬體及流程架構

本研究的作業系統使用 Ubuntu 20.10 進行開發，CPU 使用 Intel(R) Core(TM) i7-10750H CPU @ 2.60GHZ 2.59 GHZ，GPU 使用 NVIDIA GeForce RTX-2060，開發的程式語言為 Python。

本研究整體流程圖，主要分為三階段，影像數據的前處理(影像縮放和影像背景去除)、CNN 卷積神經網路模型訓練和分類結果，如圖 5 所示。

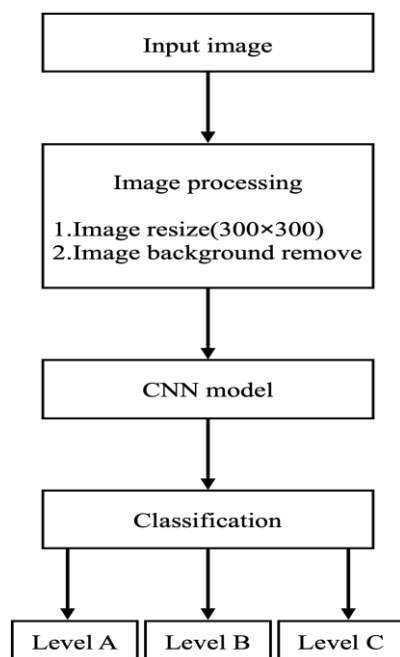


圖 5: 流程架構圖

3.2. 影像前處理

通常，影像前處理是影像分析中非常重要的步驟。這個步驟是在數據集的原始影像上進行的，用來準備將影像輸入至下一階段的模型訓練當中。因此，本研究應用到的影像前處理有：影像裁切、影像大小調整、影像背景去除和歸一化。

影像裁切[11]

主要目的是將原始影像等比例裁剪，以利於下一步調整影像大小用。

影像大小調整[12]

影像大小調整的主要目的是要將數據集內所有圖像的圖像尺寸和深度資訊一致化，以適應後面卷積神經網路 CNN 的輸入大小。因此，這裡將所有圖像轉換為具有 300 x 300 x 3 大小的 3D 圖像尺寸矩陣 (RGB 圖像)。

影像背景去除

主要的目的是要將原始影像中的柳丁提取出來，由於背景存在著許多雜訊，因此透過影像背景去除可以將影像處理的更乾淨，對於下一步的模型訓練也有許多幫助，如圖 6 所示。

歸一化[13]

主要目的是可以加快梯度下降的求解速度，進而加快網絡的收斂。所以，將影像送至模型訓練前，必須做歸一化的動作。

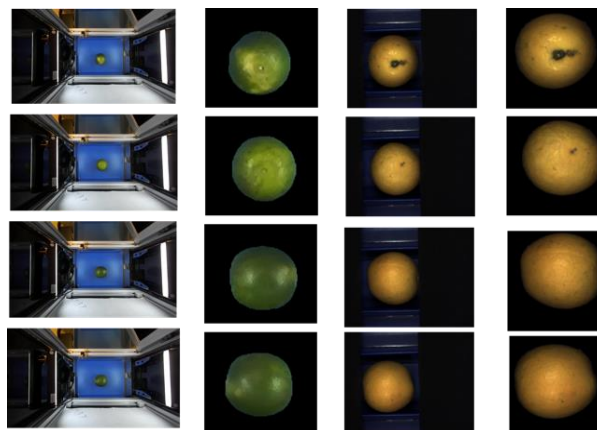


圖 6: 影像前處理

3.3. CNN 卷積神經網路[14]

近年來，由於電腦計算能力的不斷進步，大幅提升神經網路在模型訓練上的方便性與有效性，使得深度學習技術，得到了高度的關注。例如，電腦視覺[15]、語音辨識[16]等領域的應用。其中在 2012 年 ImageNet 電腦視覺挑戰賽上得獎的 AlexNet 模型，更是讓 CNN 卷積神經網路取得大眾目光，進而蓬勃發展的重要里程碑。

卷積神經網路主要是由卷積層 (Convolutional Layer)[17]、池化層 (Pooling Layer)[18] 和全連接層 (Full Connected Layer) 所組成的，基本架構如圖 7 所示。

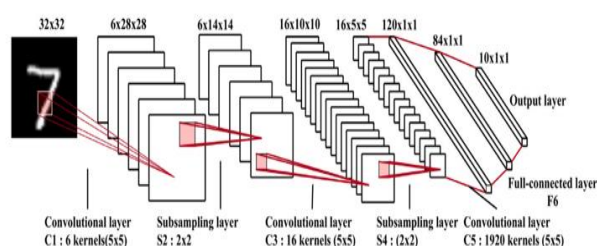


圖 7：卷積神經網路架構圖[14]

卷積層

卷積層 (Convolution Layer) 主要是負責提取圖像中的特徵，透過許多的卷積核 (Filter, Kernel) 在圖像上進行滑動，來達到擷取特徵的目的。

池化層

池化層(Pooling Layer)主要分為兩種方式，最大池化(MaxPooling)和平均池化(AveragePooling)；計算方式和卷積層類似，都是透過滑動設定的參數進行運算。主要是利用降維度來減少整體參數量，避免過度擬合(Overfitting)的問題。

全連接層

全連接層(Full Connected Layer)就是一個分類器，把經過數個卷積和池化後的結果，進行分類；全連接層的神經網路大都由三個部分所組成，分別是輸入層(Input Layer)、隱藏層(Hidden Layer)和輸出層(Output Layer)。

失活層(Dropout Layer)

主要是為了防止模型在訓練的過程中有過度擬合(Overfitting)的現象產生。

3.4. 本文提出之 CNN-Orange 模型

本研究結合影像及卷積神經網路應用於柳丁之品質分級預測，深度學習之架構如圖 7 所示。基於卷積神經網路提出一種名為 CNN-Orange 的模型，此模型架構共 19 層，依序包括了高斯噪聲層[19]、卷積層、最大池化層、全局平均池化層[20]、全連接層和 Dropout Layer[21]，CNN-Orange 深度學習架構圖如圖 8 所示。

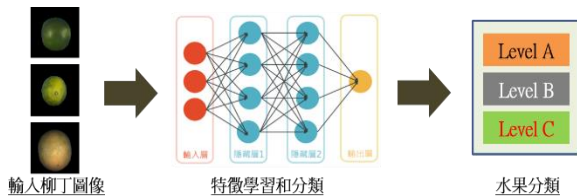


圖 8：CNN-Orange 深度學習架構圖

卷積層

在本模型中共使用了 6 層的卷積層，每一層的 Filters 依序分別為 8、16、32、64、128 和 256；Kernel Size 依序分別為 13×13、11×11、9×9、7×7、5×5 和 3×3；而 Activation 則都是採用 RELU 函數[22]。

池化層

在本模型中共使用了 2 層的最大池化層，每一層的 Pool Size 都是採用 3×3，而 Strides 都是採用 2×2。還使用了一層的全局平均池化層。

全連接層

在本模型中共使用了 6 層的全連接層，每一層的輸出空間維度則依序採用 8、16、32、64、128 和 256。輸出層由於本研究最終是分為三個等級，所以設定為 3，而 Activation 設定使用 Softmax 函數[23]，將輸出為 [0, 1] 範圍內的概率，而所有分類概率加總必須為 1。

本模型的整體架構採用基礎的捲積層搭配池化層，在第一個捲積層及最後一個捲積層後面各接一層最大池化層，目的為了利用池化層降低維度並整合特徵；而沒有在每一層的捲積層後都接池化層的原因是假如加入過多的池化層，會導致過多的特徵資訊流失；最後一層池化層後接一層全域均值池化層的主要目的是為減少參數，好接上全連接層。而捲積層以及全連接層使用了各 6 層的原因是為了防止模型欠擬合。詳細模型內容，如圖 9 和表 3 所示。

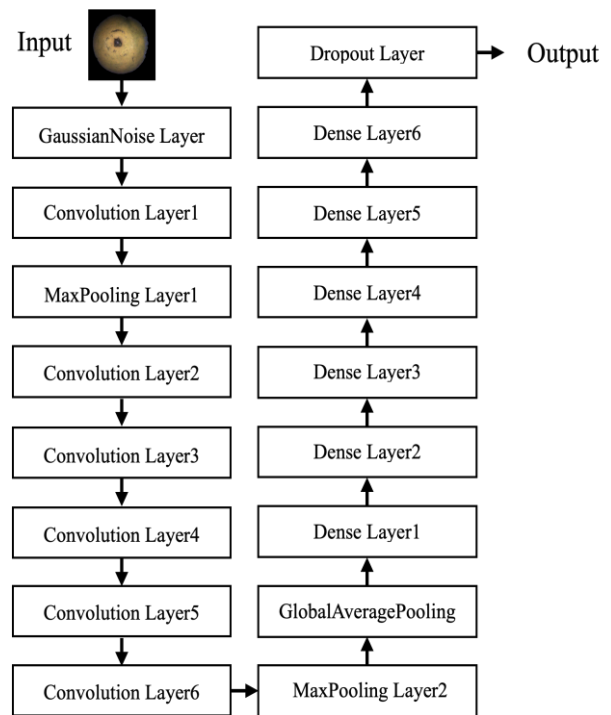


圖 9：CNN-Orange 模型分層配置圖

表 3: CNN-Orange 模型參數表

Layer(Type)	Output Shape	Param #
Input	300×300×3	0
GussianNoise	300×300×3	0
Convolution	288×288×8	4064
MaxPooling	143×143×8	0
Convolution	133×133×16	15504
Convolution	125×125×32	41504
Convolution	119×119×64	10046

Convolution	115×115×128	204928
Convolution	113×113×256	295168
MaxPooling	56×56×256	0
GlobalAveragePooling	1×256	0
Dense	1×8	2056
Dense	1×16	144
Dense	1×32	544
Dense	1×64	2112
Dense	1×128	8320
Dense	1×256	33024
Dropout	1×256	0
Dense	1×3	771
Total Params: 708,555		
Trainable Params: 708,555		
None Trainable params: 0		

3.5. 模型評估指標

混淆矩陣是一種評估分類模型好壞常用的方法，其中的評估指標有準確率、精確率、召回率和 F1-Score，分別說明如下。

準確率 (Accuracy)

準確率主要是評估預測模型中正確數量所佔的比例。以正負例二元分類為例，其準確率公式如下：

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FP} \quad (1)$$

其中

- TP (True Positive)：演算法預測為正例 (P)，實際上也是正例 (P) 的個數。
- TN (True Negative)：演算法預測為負例 (N)，實際上也是負例 (N) 的個數。
- FP (False Positive)：演算法預測為正例 (P)，實際上是負例 (N) 的個數。
- FN (False Negative)：演算法預測為負例 (N)，實際上是正例 (P) 的個數。

精確率 (Precision)

準確率有一個缺點，就是資料的樣本不均衡，這個指標是不能評價模型的效能優劣的。在準確率沒有參考價值時，就會使用精確率這個指標。評估在模型預測為正樣本的結果中，真正是正樣本所佔的百分比。精確率公式如下：

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

一個模型必須要同時符合準確與精密這兩個指標，才可算是精準。

召回率 (Recall)

這個指標就是在實際為正樣本中，被預測為正樣本所佔的百分比。即所有正例中，有多少被正確地判定為正。召回率公式如下：

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1-Score

當需要同時兼顧精確率和召回率時，就會考慮使用 F1-Score 這個指標，優點是能將精確率和召回率的數值調和平均。其中最大值是 1，最小值是 0。F1-Score 公式如下：

$$F1 - Score = 2 \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

4. 研究結果與分析

4.1. 模型訓練結果與分析

在配置模型訓練階段時，權重是隨機初始化的，優化器是使用 Adam[24]進行訓練的；Learning Rate 是設定 0.0001；在模型訓練的階段，我們設定 Batch Size(每次梯度更新的數量)為 32；Epochs 設定為 100，主要是用來設定訓練模型的時期數；在訓練過程中為了防止模型過擬合，使用 Early Stopping[25]來達到更好的收斂。

訓練完成後，為了評估模型性能，我們使用了準確率、損失率來評估模型，訓練結果曲線如圖 10 和圖 11 所示。

圖 10 在說明訓練集和驗證集每個時期的準確率學習曲線。圖中藍色曲線是代表訓練集準確率的學習曲線，而橘色曲線則是代表驗證集中準確率的學習曲線。可以看出，大約在 5 次訓練之後其準確率就趨於穩定收斂。但在驗證集的部分雖然也很快的達到訓練集的準確度，但整體來看似乎不是非常穩定有較大的變異，其中原因係本研究在訓練過程中設有早停機制(Early Stopping)，避免過度擬合問題。



圖 10：訓練和驗證集準確率學習曲線

圖 11 在說明訓練集和驗證集每個時期的損失率學習曲線，圖中可以得知，藍色曲線是訓練集損失率的學習曲線，而橘色曲線是驗證集損失率的學習曲線。這裡 Loss 設定 CategoricalCrossentropy，主要是用來計算預測值及真實值的多類別交叉熵。由此可知，不需要長時間的訓練時期就可以得到低損失率，而驗證及損失率跟訓練集是差不多的，所以代表沒有過擬合的問題。損失率公式如下：

$$Loss(E) = -\sum_{m=1}^k \sum_{n=1}^l x_{m,n} \log(\rho_{m,n}) \quad (5)$$



圖 11：訓練和驗證集損失學習曲線

4.2. 模型測試結果與分析

本研究測試結果之混淆矩陣如圖 12 所示。依據節 3.5. 所述之評估指標，其計算結果如表 4 所示。

	Level A	Level B	Level C
Level A	69	7	0
Level B	0	73	3
Level C	0	7	69

圖 12：模型混淆矩陣

表 4: 分類模型評估指標

	Precision	Recall	F1-Score
Level A	1.00	0.91	0.95
Level B	0.84	0.96	0.90
Level C	0.96	0.91	0.93
Accuracy	0.93		

根據圖 11 可以得知，將 Level A 預測為 Level A 的有高達 69 個，將 Level A 預測為 Level B 的有 7 個，將 Level A 預測為 Level C 的有 0 個；將 Level B 預測為 Level A 的有 0 個，將 Level B 預測為 Level B 的有高達 73 個，將 Level B 預測為 Level C 的有 3 個；將 Level C 預測為 Level A 的有 0 個，將 Level C 預測為 Level B 的有 7 個，將 Level C 預測為 Level C 的有高達 69 個。由以上可知，該模型預測的表現良好。進一步分析混淆矩陣中辨識錯誤的影像類別，依結果顯示可以知道其中包括蒂頭 3 張、果身 13 張和果底 1 張。計算測試集中各影像特徵的辨識準確率，蒂頭部分可達 95.24%；果身部分為 87.38%；果底部分高達 98.39%。詳如表 4 所示。由以上結果可知，柳丁的甜度分級中影像特徵相關性最高為果底影像，其次為蒂頭影像，果身影像的相關性相較就比較低，因此容易造成辨識錯誤。

表 5: 影像特徵分類比較

	蒂頭	果身	果底
測試數量	63	103	62
辨識錯誤數	3	13	1
辨識錯誤率	0.0476	0.1262	0.0161
測試準確率	0.9524	0.8738	0.9839

從表 4 可以得知，在準確率表現上，達到了 93%；再進一步分析 Level A、Level B 和 Level C 在精確率上的表現，分別達到 1.00、0.84 和 0.96；其中預測是 Level B 中只有 84% 是真正的 Level B，低於 Level A 和 Level C 的 96% 以上。在召回率上分別達到了 0.91、0.96 和 0.91；F1-Score 上分別達到了 0.95、0.90 和 0.93。所以，就評估指標結果來說，驗證本研究提出之 CNN-Orange 模型可以精準地來進行柳丁甜度的分類預測。

5. 結論與後續工作

5.1 結論

1. 本研究中，我們實現了一種使用結合影像分析與卷積神經網路的方法，並驗證 CNN-Orange 模型用以解決柳丁內在品質的分類問題的有效性。
2. 應用實際採收的柳丁拍攝影像數據集用以訓練 CNN-Orange 的模型參數，測試準確率可以達到 93%，足證本研究方法的有效與實用性。
3. 依研究結果可知，果底部分的影像特徵可以得到 98.39% 的辨識準確率；蒂頭部分的影像特徵也可以達到 95.24%；相較之下，果身只有 87.38%。所以，柳丁的甜度分級實務上可採用果底和蒂頭兩者的影像作為外觀特徵，使辨識的準確率提升到 96.8%。
4. 假設同一顆柳丁，但不同部分辨識出不同的結果時，則以較高機率的部分作為辨識結果。
5. 本研究成果配合精密機械研究發展中心的機器人技術，整合成自動化分級系統在 2021 台灣機器人與智慧自動化展上展出並獲得許多廠商的關注，如圖 13 所示。

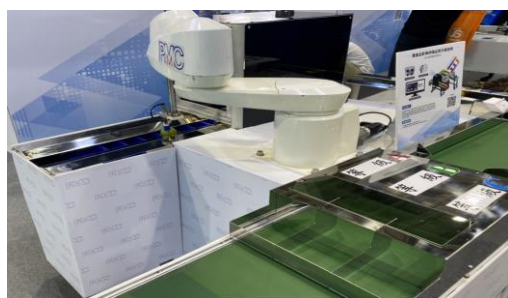


圖 13：2021 台灣機器人與智慧自動化展

5.2 後續工作

1. 訓練數據的資料集數量上可以再增加，以提升準確率與精密度。
2. CNN 模型的選擇與訓練，可以有其他的配置，有興趣的研究者可以參考本文做進一步的發展。

參考文獻

- [1] 行政院農委會. 新農業創新推動方案 2.0. 台北台灣. 110 年(<https://www.coa.gov.tw/ws.php?id=3>)
- [2] Villa, M., Qiu, J., & Pearce, E. L. (2018). A sweet deal for diabetes. *Trends in Endocrinology & Metabolism*, 29(1), 1-2.
- [3] Janardhana, K., Jesi, V. E., Vijayaragavan, M., & Kumar, R. B. D. A. N. (2021). Non-destructive classification of fruits based on color by using machine learning techniques. *International Journal of Modern Agriculture*, 10(1), 1057-1069.
- [4] Kanimozhi, B., & Malliga, R. (2017). Classification of ripe or unripe orange fruits using the color coding technique. *Asian Journal of Applied Science and Technology (AJAST) Volume*, 1, 43-47.
- [5] Nazulan, W. N. S. W., Asnawi, A. L., Ramli, H. A. M., Jusoh, A. Z., Ibrahim, S. N., & Azmin, N. F. M. (2020, November). Detection of Sweetness Level for Fruits (Watermelon) With Machine Learning. In 2020 IEEE Conference on Big Data and Analytics (ICBDA) (pp. 79-83). IEEE.
- [6] Bejo, S. K., & Kamaruddin, S. (2014). Determination of Chokanan mango sweetness ('Mangifera indica') using non-destructive image processing technique. *Australian Journal of Crop Science*, 8(4), 475-480.
- [7] Supekar, A. D., & Wakode, M. (2020). Multi-parameter based mango grading using image processing and machine learning techniques. *INFOCOMP Journal of Computer Science*, 19(2), 175-187.
- [8] Das, P., & Yadav, J. P. S. (2020, September). Automated Tomato Maturity Grading System using CNN. In 2020 International Conference on Smart Electronics and Communication (ICOSEC) (pp. 136-142). IEEE.
- [9] Sangsongfa, A., Am-Dee, N., & Meesad, P. (2020). Prediction of Pineapple Sweetness from Images Using Convolutional Neural Network. *EAI Endorsed Transactions on Context-aware Systems and Applications*, 7(21).
- [10] Azman, A. A., & Ismail, F. S. (2017). Convolutional neural network for optimal pineapple harvesting. *ELEKTRIKA-Journal of Electrical Engineering*, 16(2), 1-4.
- [11] Qingqing, H., Tianbao, Y., & Bin, L. (2021). A Survey on Crop Image Segmentation Methods. *The Proceedings of The 8th International Conference on Intelligent Systems and Image Processing 2021*.
- [12] Wang, Q., & Yuan, Y. (2014). Learning to resize image. *Neurocomputing*, 131, 357-367.
- [13] Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448-456). PMLR.
- [14] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354-377.

- [15] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- [16] Graves, A., Mohamed, A. R., & Hinton, G. (2013, May). Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing* (pp. 6645-6649). Ieee.
- [17] Komatsu, T., & Li, R. (2016). Convolution identities for Tribonacci numbers with symmetric formulae. *arXiv preprint arXiv:1610.02559*.
- [18] Masci, J., Giusti, A., Ciresan, D., Fricout, G., & Schmidhuber, J. (2013, September). A fast learning algorithm for image segmentation with max-pooling convolutional networks. In *2013 IEEE International Conference on Image Processing* (pp. 2713-2717). IEEE.
- [19] Zhou, D. P. (2013). An Variable Coefficient Images Denoising Method. In *Applied Mechanics and Materials* (Vol. 333, pp. 832-835). Trans Tech Publications Ltd.
- [20] Lin, M., Chen, Q., & Yan, S. (2013). Network in network. *arXiv preprint arXiv:1312.4400*.
- [21] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1), 1929-1958.
- [22] Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.
- [23] Nwankpa, C., Ijomah, W., Gachagan, A., & Marshall, S. (2018). Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*.
- [24] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- [25] Caruana, R., Lawrence, S., & Giles, C. (2000). Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. *Advances in neural information processing systems*, 13.