



結合 CGAN 與 YOLOv4 進行芒果等級分類

Combining Conditional Generative Adversarial Networks  
and YOLOv4 for Mango Classification

指導教授：李麗華 博士

研 究 生：江伶嫻

中華民國 111 年 2 月

朝陽科技大學資訊管理系  
Department of Information Management  
Chaoyang University of Technology

碩士論文  
Thesis for the Degree of Master

結合 CGAN 與 YOLOv4 進行芒果等級分類  
Combining Conditional Generative Adversarial Networks and  
YOLOv4 for Mango Classification

指導教授：李麗華 博士(Li-Hua Li)

研 究 生：江伶琪 (Ling-Qi Jiang)

中華民國 111 年 2 月

February, 2022

## 中文摘要

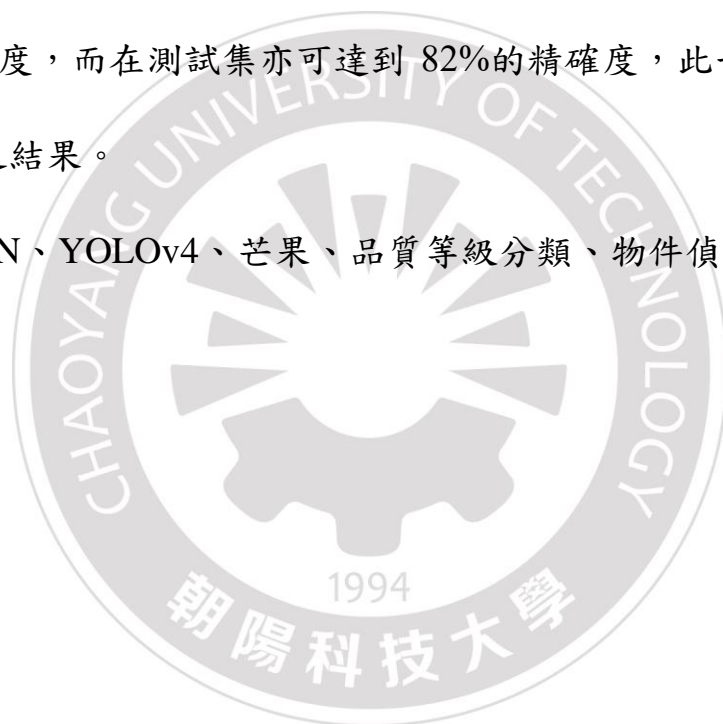
芒果是台灣重要出口的農產品之一，在 2020 年的總出口值為 26,099 千美元，為台灣帶來了許多經濟效益。為提升芒果的分類品質，芒果採收後的檢查與分類作業是十分重要的，因為不正確的品質分類將會影響購買者的意願。過去有學者們在芒果品質分類上進行各種研究，例如：透過影像處理技術找出缺陷處或分析芒果的顏色、大小等各項評估指標來進行分級。也有學者透過卷積神經網路(Convolutional Neural Networks; CNN)進行芒果等級辨識。然而水果有新鮮度的時限，如何快速且準確地進行檢測分類，讓採收後的芒果在經由更有效的等級分類後可以快速運輸販售或加工，以利發揮最大的經濟效益是現今非常重要議題。

本研究運用台灣 AI CUP 2020:愛文芒果影像辨識雙項競賽所提供的芒果資料集，並依據此資料集將芒果分為 A、B 和 C 三類，分別為出口用、內銷用和加工用。

本研究結合 Conditional Generative Adversarial Networks (CGAN)和 YOLOv4 建構一個愛文芒果等級辨識系統，用來改善採收後分類的準確度。CGAN 可以根據標籤產生逼真的圖片使得資料更多樣化，藉此來增強資料，同時也可以幫助資料量不足時的資料擴充。YOLOv4 則用於分類等級和檢測芒果位置。本研究也將 YOLOv3 SPP 和 YOLOv4 模型進行比較，實驗結果顯示在結合 CGAN 和 YOLOv4 的模型上有最好的表現，其中訓練集可達

到 85%的精確度，而在測試集亦可達到 82%的精確度，此一結果優於台灣 AI CUP 比賽之結果。

**關鍵字：**CGAN、YOLOv4、芒果、品質等級分類、物件偵測、人工智慧



## **Abstract**

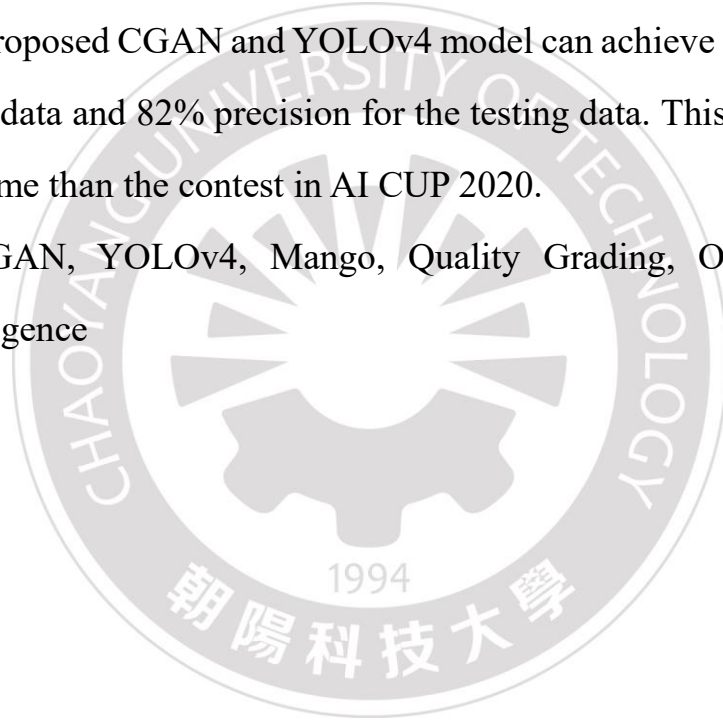
Mango is one of Taiwan's important agricultural exports. In 2020, the total export value was US\$26,099 dollars. The export of mango has provided Taiwan with numerous economic benefits. Therefore, the inspection and classification process of harvested mangoes is very important, and the results will affect the willingness of repurchase. In the past, many scholars have proposed research topics on mango quality classification such as identifying defects through image processing technology or analyzing mango color, size and other evaluation indicators for classification. Some scholars used neural networks to identify the mature degree of mango. Since the freshness of fruit will decay over time, the speed and accuracy of mango classification are important for mango harvesting process so that the harvested mango can be rapidly transported, distributed and sold to obtain maximum economic benefits after classification.

In this research, we use the data which provided by AI CUP 2020: Mango Grades Classification Competition. The AI CUP 2020 dataset contains mangoes images and they are labelled as grade A, grade B, and grade C. Based on these labels, mango fruit will be sent to export (Grade A), to sale in the domestic market (Grade B), or to the juice factory (Grade C).

In this research, we propose the Irwin mango grades classification system by combining Conditional Generative Adversarial Networks (CGAN) and YOLOv4 to improve the classification of harvested mangoes. CGAN can help data enhancement by generating labeled images to increase the data diversity. CGAN can also help data expansion when the amount of data is insufficient. YOLOv4 is used to classify and detect the position of mangoes in an image. This research also compares both YOLOv3 SPP and YOLOv4 models. The experimental results

show that our proposed CGAN and YOLOv4 model can achieve 85% of precision for the training data and 82% precision for the testing data. This result has better precision outcome than the contest in AI CUP 2020.

**Keywords:** CGAN, YOLOv4, Mango, Quality Grading, Object Detection, Artificial Intelligence



## 致謝

在研究所期間學習到了許多研究上的能力，像是報告的要點、研究所須的知識以及做事的方法以及態度。首先我要感謝的是唐元亮老師，老師提供了我工作機會，讓我可以不為生活費所困擾，也從中學習到怎麼把知識用簡單明瞭的方式教導給別人，在研究方面如果我們有想學習研究的相關課程，老師會提供經費讓我們去學習，非常謝謝老師。

另外想感謝就是指導老師--李麗華老師，老師在研究方面提供了我許多寶貴的建議，讓我在研究以及寫作上都有明顯的進步，也讓我學習到了做事情應有的處理態度及方式，會讓我在未來工作時更有幫助。也要特別感謝陳宏益老師，感謝老師讓我當了他兩年的助教，也提供了我額外打工的機會，讓我從中學習到了許多事情。

除此之外，我也要感謝在研究過程中幫助過我的 Christine Dewi，當我在困惑時可以跟她討論，她總是提供給我許多的建議，也會教導我在研究時的許多資訊。最後也要感謝許多在研究中幫助以及很關心我的老師、同學、朋友以及家人，讓我可以順利的完成學業。

## 目錄

第壹章 緒論 .....	1
1.1 研究背景與動機 .....	1
1.2 研究目的 .....	8
1.3 研究範圍 .....	9
1.4 研究規劃步驟 .....	10
1.5 論文架構 .....	10
第貳章 文獻探討 .....	11
2.1 資料擴增(Data Augmentation) .....	11
2.1.1 RGB 轉換到 HSV .....	13
2.1.2 GAN .....	14
2.1.3 Mosaic .....	16
2.2 資料增強 .....	17
2.3 芒果辨識 .....	17
2.4 AlexNet .....	18
2.5 YOLO .....	19
2.6 AI CUP .....	24
第參章 研究方法 .....	28
3.1 芒果資料集 .....	29



3.2 參數設定步驟.....	32
3.3 YOLOv4 架構.....	33
3.4 驗證方式.....	35
<b>第肆章 實驗與結果</b> .....	<b>38</b>
4.1 AlexNet 實驗與結果.....	38
4.2 YOLOv3 SPP、YOLOv4 以及 CGAN YOLOv4 實驗與結果.....	40
4.3 CGAN YOLOv4 實驗與結果.....	52
4.3.1 實驗方法.....	53
4.3.2 實驗結果.....	55
<b>第伍章 結論</b> .....	<b>58</b>
<b>參考文獻</b> .....	<b>60</b>

## 表目錄

表 1、109 年度各國輸出量和價值[本研究整理] .....	3
表 2、ALEXNET 架構表 .....	19
表 3、分級標準的規則 .....	31
表 4、訓練和測試影像資料張數分布 (均值 MEAN=1,866) .....	31
表 5、各模型參數設定 .....	43
表 6、實驗設備環境 .....	46
表 7、驗證 YOLOv3 SPP 在訓練集和測試集的結果 .....	48
表 8、驗證 YOLOv4 在訓練集和測試集的結果 .....	49
表 9、驗證 CGAN YOLOv4 在訓練集和測試集的結果 .....	50
表 10、競賽 BASELINE MODEL、最佳模型和實驗模型的 WAR 比較 .....	51
表 11、不同 CGAN YOLOv4 模型在訓練集的結果 .....	55
表 12、不同 CGAN YOLOv4 模型在測試集的結果 .....	56

## 圖目錄

圖 1、106 年到 109 年總出口量 [本研究整理].....	2
圖 2、106 年到 109 年總出口價值[本研究整理].....	2
圖 3、GAN 架構圖.....	15
圖 4、YOLOv4 架構圖.....	22
圖 5、芒果各等級的問題範例圖.....	26
圖 6、本研究之 CGAN 流程圖.....	28
圖 7、本研究提出三種芒果辨識模型之建構流程圖.....	28
圖 8、訓練資料集中的(A)等級 A、(B)等級 B 和(C)等級 C 範例圖.....	30
圖 9、YOLOv4 的簡易架構運行流程圖.....	34
圖 10、測試流程圖.....	36
圖 11、先用 YOLOv3 進行芒果定位，此為找到芒果之情形.....	39
圖 12、先用 YOLOv3 但沒有定位到芒果的處理方式.....	39
圖 13、(A)ALEXNET 模型準確率及(B)ALEXNET 損失過程折線圖.....	40
圖 14、使用 LABELIMG 標註資料集之介面.....	41
圖 15、CGAN 資料前處理流程示意圖.....	45
圖 16、CGAN YOLOv4 的實驗流程.....	46
圖 17、YOLOv3 SPP 訓練圖.....	47
圖 18、YOLOv4 訓練圖.....	49

圖 19、CGAN YOLOv4 訓練圖.....	50
圖 20、YOLOv3 SPP、YOLOv4 和 CGAN YOLOv4 預測結果圖 .....	52
圖 21、CGAN YOLOv4 改進版流程圖.....	53
圖 22、圖片調整亮度 .....	54
圖 23、圖片經三種不同方式進行調整的結果圖 .....	55

## 第壹章 緒論

### 1.1 研究背景與動機

芒果是台灣重要出口的農產品之一，從台灣行政院農委會的統計[1]得知 2017 年到 2020 年的芒果總出口量(如圖 1)和總出口值(如圖 2) 都是逐年提高，其中 2017 年的總出口量為 4,762 公噸、2018 年的總出口量為 5,201 公噸、2019 的總出口量為 8,108 公噸、而 2020 年的總出口量為 9,579 公噸。由此可知，自 2017 年到 2020 年台灣芒果的出口量成長了幾乎兩倍(參見圖 1)。從台灣行政院農委會的統計數據[1]得知台灣芒果 2017 年的總出口值為 14,826 千美元、2018 年的總出口值為 17,361 千美元、2019 年的總出口值為 22,124 千美元、2020 年的總出口值為 26,099 千美元，因此外銷芒果的成長量也在四年內提高了 1.76 倍。由上述資料可以得知，近年來我國芒果的外銷銷售量及金額都是持續在增長，代表著台灣芒果外銷的需求量佔有一定重要地位。

台灣芒果外銷的國家包含日本、中國、美國以及香港等地(如表 1)。從台灣行政院農委會的統計數值[1]以 2020 年的外銷數據來看，出口到日本的量為 692 公噸、中國 3,981 公噸、韓國 825 公噸、香港 3,727 公噸以及其他 170 公噸，出口值大約為 0.22 億美元，由圖 2 可以得知 2017 年到 2020 年的出口值成長了約 1.7 倍，故芒果外銷對台灣來說有相當的經濟價值。我們將 2020 年各國的出口量和出口值做成(如表 1)，由表 1 我們可以得知，台

灣芒果主要外銷國家都是同為亞洲地區的中國、香港、日本以及韓國。台灣外銷的需求量那麼多代表台灣出產的芒果有很好的品質。

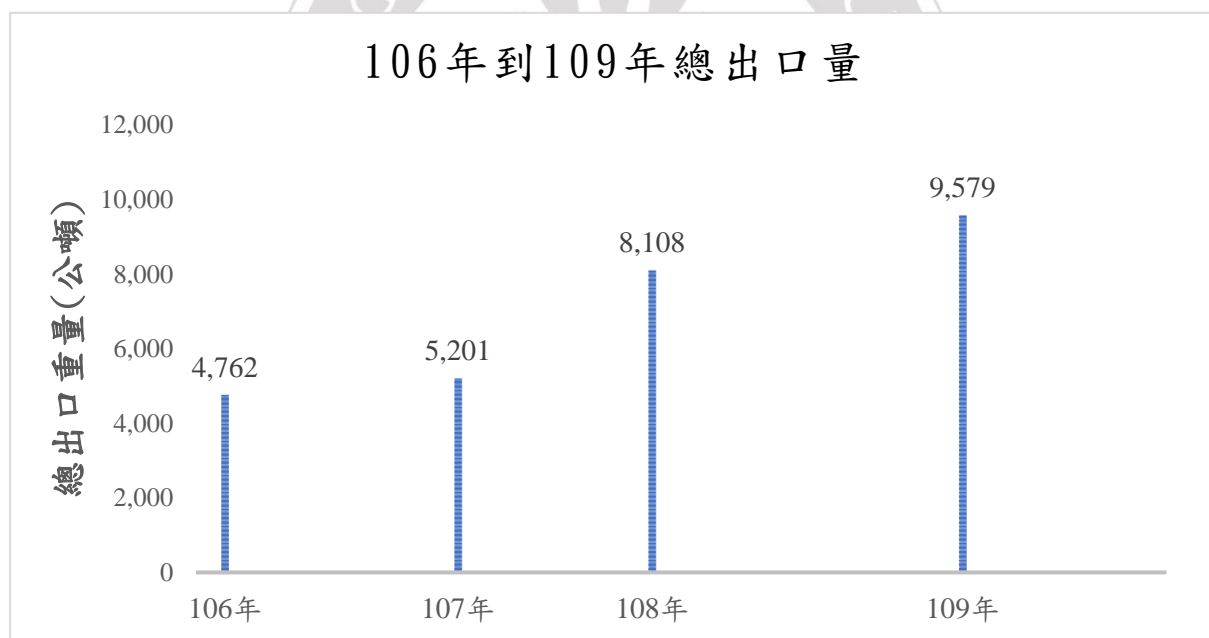


圖 1、106 年到 109 年總出口量 [本研究整理]

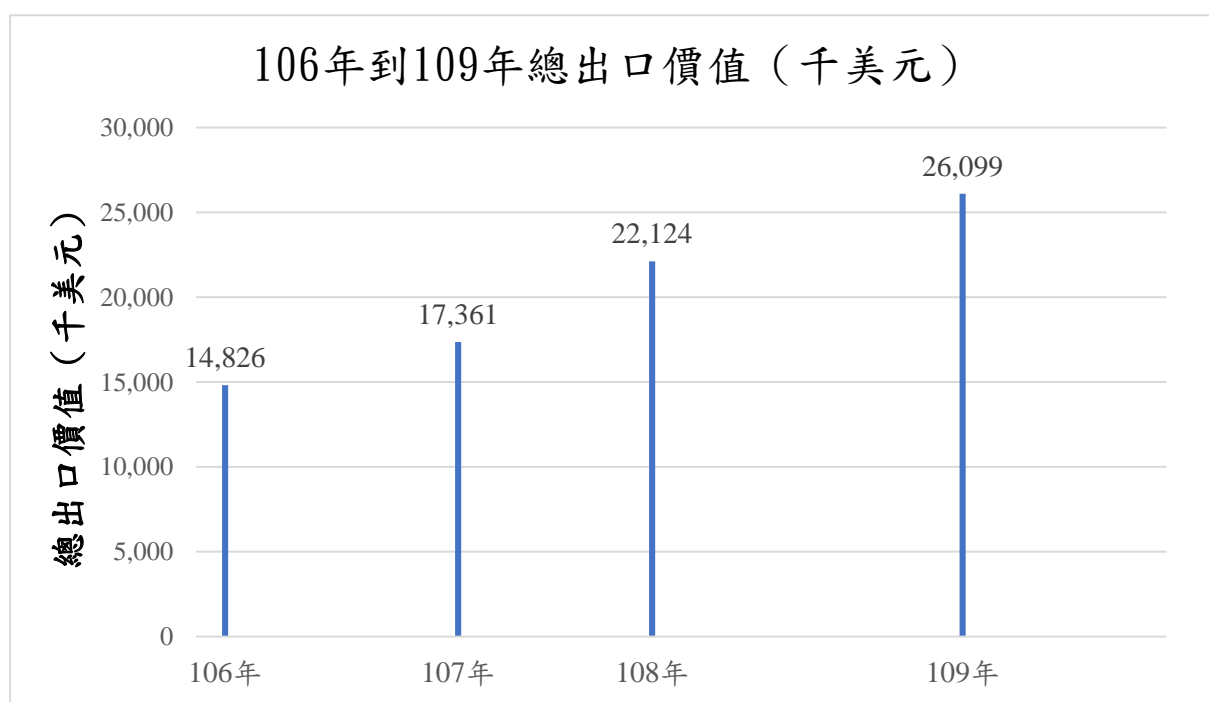


圖 2、106 年到 109 年總出口價值[本研究整理]

表 1、109 年度各國輸出量和價值[本研究整理]

國家	重量(公噸)	價值(千美元)
澳大利亞	18	130
汶萊	7	33
加拿大	35	86
中國	3,981	7,846
香港	3,727	7,440
日本	692	5,421
韓國	825	4,602
澳門	11	23
馬來西亞	32	69
新加坡	250	449

水果品質檢測對於出口是很重要的，外銷的品質好壞會影響購買者的意願，也因為水果有新鮮度的限制，所以從採收到出口需要快速及準確的檢測，根據不同的水果有不同的檢測方式及評定規則，大部分都會根據水果的顏色、外觀、形狀、重量、甜度、大小或缺陷來區分。水果品檢測方式主要可以分成(1)利用人力肉眼觀察用以往的經驗進行篩選，這種方式非常耗時、需要大量的人力也因不一致的評斷方式和主觀的意見所以容易誤認；(2)利用機器篩選：用機器篩選可以有更一致的標準和加速流程；(3)利用電腦視覺去分類：電腦視覺可以利用相機和攝影機獲取圖像，將圖像經過圖片前處



理後，進而分析圖像並從中找出特徵後再進行分類，這種方式跟人眼相比，分類的標準較一致，通常分類的結果也會更好。

常見的圖片前處理方法有轉換色彩空間(Color space)[2][3][4]、圖片二值化(Binarization) [2][3][4]、移除背景[4]、降噪 (Noise reduction) [2]、銳利化(Sharpen)[5]和邊緣檢測(Edge detection)[6]等。現今也有不少學者利用人工智慧(Artificial Intelligence, AI)，如機器學習(Machine Learning, ML)和深度學習(Deep Learning, DL)進行分類。

有關機器學習的分類[7-10]，其主要做法是分析、計算資料集裡的特徵，並由特徵的相似分析進行分類，機器學習常見的方法如 K-Nearest Neighbor[7][8]、K-means Clustering [8]、Support Vector Machine [9]和 Random Forest [10]。深度學習[11-16]可以依據所輸入的資料，自動化的學習資料特徵並自行調校分類，而深度學習常見的方法如: Convolutional Neural Network (CNN)[11]、Faster Region Convolutional Neural Network (Faster-RCNN)[12]、YOLO[13]。在本論文撰寫期間 YOLO (You Only Look Once)模型已有 4 種不同的版本發表。雖 YOLO 在網路上有出現第 5 個版本的程式，但截至本論文撰寫期間，該版本仍未正式有論文發表，因此本研究先以 YOLO 已發表的學術版本做為探討。

近來也有不少學者使用 GAN(Generative Adversarial Network)[14]、DCGAN(Deep Convolutional GAN)[15]、WGAN(Wasserstein GAN)[16]和



CGAN(Conditional GAN)[17]等各種對抗生成網路模型來產生圖片、擴增資料集再結合模型進行分類。上述各種分類方法都被嘗試使用在各種水果的分類作業上，而近來的趨勢則有更多應用是以電腦視覺結合人工智慧來進行水果分類。

水果品質檢測是一個熱門的研究領域，如以 Scopus 電子資料庫(<https://www.scopus.com/home.uri>)所收錄的論文為例，若使用關鍵字“fruit quality”進行查詢，則可以搜尋到 16,277 篇的相關論文。又以 Web of Science (WoS)電子資料庫為例，使用關鍵字“fruit quality”查詢，可以找到 11,041 篇相關的論文。由此可知，水果品質檢測這方面的研究一直是熱門的議題，其中包括了在芒果品質檢測[2][3][9][18]、蘋果品質檢測[11][12]、草莓品質檢測[19]、番茄品質檢測[20]和櫻桃品質檢測[21]等。

根據上述兩個論文資料庫所搜尋到的論文數量結果和舉例的水果種類，可以得知，不論水果的種類為何，水果品質的檢測都是非常重要的。雖然不同水果之間的品質檢測方式和品質評斷規則差異很大，但運用科學方法與技術來解決水果檢測的研究一直是受學者們重視的議題。目前水果檢測大約可概為兩種類型，第一種是利用影像處理的方式找出缺陷處，缺陷處可用於模型訓練分類或制定評斷規則去分類，例如:缺陷面積比例或缺陷類型。第二種則是建立模型將水果圖片進行學習，讓模型能依據水果所對應的標籤來自動化認知圖片中的水果是屬於哪一類。

在電腦視覺方面 Momin 等學者[2]將芒果依據面積分成大中小 3 類，並將 RGB 圖片轉換到 HIS(Hue, Saturation, Intensity)色彩空間，再經由二值化後使圖片平滑化(Image smoothing)、膨脹(Dilate)將芒果經過面積計算分成大中小。Vyas 等學者[3]將芒果分成等級 1、2、3 和不合格的芒果，將 RGB 圖片轉為灰階後利用中值濾波器(Median filter)去噪(Denoising)，再利用 Ostu 方法將圖片二值化得到背景為黑芒果為白的二值圖，利用二值圖可以計算缺陷面積比，即芒果缺陷面積除以芒果的總面積，接著再利用二值圖找出芒果面積的中心位置，根據芒果區域圖找出芒果的長軸和短軸區分芒果種類，將 RGB 圖片轉為 Lab 色彩空間，這裡不是轉整張圖片，而是定義將長軸上 80%的面積和短軸上 30%的面積這個條型區域轉為 Lab 色彩空間，這麼做的原因是節省轉換和處理時間。利用轉換後的 Lab 圖片分析 a 通道，找出主色的範圍，利用主色去分類芒果的成熟度，將所有的判斷依據定義的規則分成等級 1、2、3 和不合格的芒果。

運用機器學習(Machine Learning)進行水果檢測的研究方面則有 Deulkar and Barve[22]學者，他們將番茄分為三個等級，獲取番茄圖片的周長、面積、中位數和顏色當作特徵，利用 Ostu 方法將圖片二值化，再利用 Support Vector Machine(SVM)當作分類器。

運用人工智慧進行水果檢測的研究方面則有 Fan 等學者[11]將蘋果分為正常和有缺陷兩類，結合每秒 5 個蘋果的 4 線道的分選機器，經過照明

後，利用相機擷取圖片，再將圖片放入卷積神經網絡 (Convolutional Neural Network) 簡稱 CNN 去訓練。Suganya 等學者[12]使用 Faster Region Convolutional Neural Network(Faster-RCNN)[23]將蘋果分為缺陷和沒有缺陷兩類。Wu 等學者[18]使用 AICUP 2020[24]的芒果資料集，與本研究所選用的資料集相同，先將資料大小縮成 224x224，做特徵縮放(Feature scaling)、正規化(Normalization)資料後，利用 Mask-RCNN[25]將背景移除，Wu 等學者修改 Shvets 等學者[26]提出的 TernaNet11、TernaNet16 和 AlbuNet34 架構，在三個模型 encoder 後都加入分類器去分類芒果等級。Shvets 等學者所提出的模型原先是運用於血管發育異常檢測和定位，三個模型原始架構是一個編碼及解碼架構，分別利用不同的先前訓練編碼器 VGG11 encoder、VGG16 encoder 和 ResNet34 encoder。

縱觀上述各類方法，不難發現運用機器學習是現今水果檢測的趨勢，但大部分的模型都沒辦法自動化標註水果在圖片的位置，因此需要大量的前處理，由於學者們也整合非常多的模型和影像前處理，因此水果檢測所需要的時間較長。而在實際的芒果檢測中，大部分都是將芒果放在輸送帶上進行運輸並檢測芒果，因此需要採用可以即時預測的模型。而為了改善上述這些研究的問題以及更符合實際動態即時檢測流程，本研究提出運用 YOLO 模型將水果定位並進行水果等級辨識之實作模型。因此本研究運用 YOLO 來當作的實作模型，在本論文撰寫期間 YOLO 發表 YOLO[27]、YOLOv2[28]、

YOLOv3[29]和 YOLOv4[13]四個版本，為了探討哪個版本更適用於芒果圖片，本研究會對 YOLO 進行實作、探究和比較。

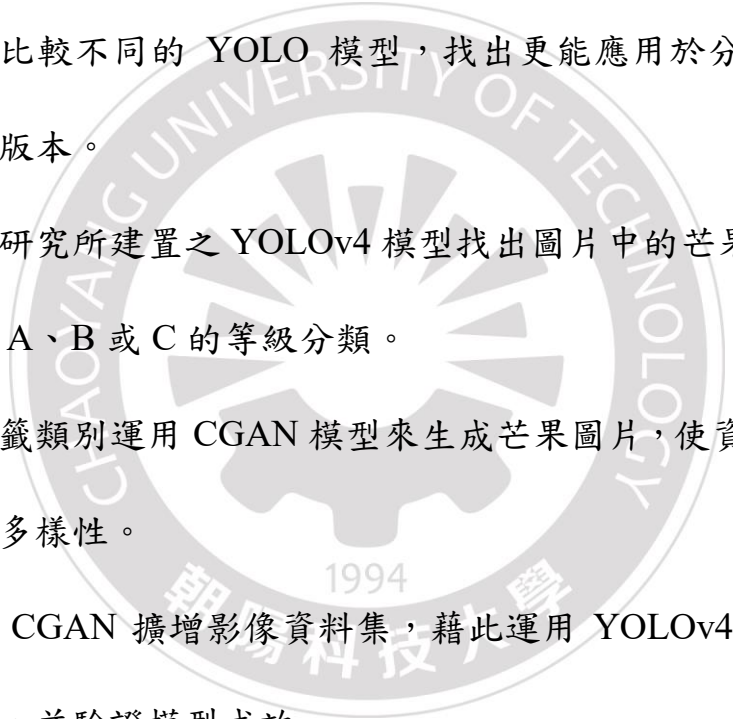
芒果外銷對於台灣是重要的，所以採收後的品質檢測的技術仍需不斷的改進，如何發展更精確快速地檢測技術，使得更加快速的把水果送出販賣，獲取更大的經濟利益，是本研究所注重的。為了研發更好的芒果檢測技術，台灣經濟部專案委託 AIda 網路平台辦理 AI CUP [24]，AIda 是一個由經濟部技術處科技專案支持、委由工研院巨量資訊科技中心與工研院產業科技國際策略發展所共同開發執行之人工智慧共創平台，該平台特別針對台灣的愛文芒果在 2020 年辦理芒果等級自動化分類辨識競賽。這個競賽是為了改善芒果採收後處理技術，並將採收後的愛文芒果依品質篩選為 A、B、C 三等級，依序為出口用、內銷用、加工用。

## 1.2 研究目的

為了解決芒果等級分類問題，也為了能更符合實際應用上，本研究選用物件偵測的模型來解決問題，而物件偵測模型中，YOLOv4 在過去學者的研究中[30][31][32]顯示其速度和準度的表現都相較於其他模型優異，故本研究欲提出以 YOLOv4 來提升台灣芒果影像辨識用於分級問題。本研究利用台灣 AI CUP 競賽[24]所提供的芒果資料集建立一套影像辨識自動篩選芒果系統，分類 A、B、C 三個等級。

本研究目的如以下四點：



- 
- (1)實驗及比較不同的 YOLO 模型，找出更能應用於分類芒果等級的 YOLO 版本。
  - (2)利用本研究所建置之 YOLOv4 模型找出圖片中的芒果位置，並將芒果進行 A、B 或 C 的等級分類。
  - (3)根據標籤類別運用 CGAN 模型來生成芒果圖片，使資料集中的圖片更具有多樣性。
  - (4)提出以 CGAN 擴增影像資料集，藉此運用 YOLOv4 來增強芒果檢測分類，並驗證模型成效。

### 1.3 研究範圍

本研究利用台灣 AI CUP 競賽[24]所提供的愛文芒果資料集建立一套影像辨識自動篩選芒果系統，根據競賽提供的分類規則去分類芒果的 A、B 和 C 三個等級。而台灣 AI CUP 競賽所提供的芒果圖片資料，主要的品種為台灣愛文芒果，因此本研究之模型主要係以愛文芒果之影像為主。不過本研究旨在探究何種機器學習模型較佳，因此未來如欲應用本研究所提之模型於其它品種之芒果，則將模型重新訓練即可。

本研究主要以訓練能找出標地物(即芒果)圖框的 YOLO model 為主要之建置模型，而本論文主要係探討 YOLOv3 SPP、YOLOv4 和 CGAN 結合 YOLOv4 等 3 種模型進行訓練和驗證成果探討，並與 AI CUP 競賽的 baseline

模型和最佳模型進行比較。因此，農場之芒果影像拍攝技術或品種分辨之生產線之物聯網技術，並不在本研究含蓋之範圍裡。

#### 1.4 研究規劃步驟

本研究所規劃之步驟及流程簡要說明如下：

- (1) 確認研究題目；
- (2) 尋找相關文獻，包括芒果辨識、物件偵測、機器學習方法和影像處理等；
- (3) 規劃研究方法、步驟及實驗內容；
- (4) 蒐集資料、撰寫程式、進行模型訓練、及模型之測試與實驗；
- (5) 分析實驗結果、評估不同之模型成效，進行比較；
- (6) 探討研究之結果，撰寫論文。

#### 1.5 論文架構

本論文架構分為五個章節：第壹章緒論說明研究背景、研究動機和研究目的；第貳章文獻探討分為六個部分：(1)探討研究中用到的資料擴增，(2)探討研究中用到的資料增強，(3)芒果辨識相關研究，(4)介紹 AlexNet 模型與架構，(5)YOLO 的模型演進，和(6)介紹 AICUP，第參章說明本研究模型建置之方法、介紹資料集和說明整體系統建置與測試流程，第肆章為實驗設定、實驗結果、模型比較與討論，最後第伍章為結論，總結本研究 and 未來工作。

## 第貳章 文獻探討

物件偵測(Object Detection)可以在影片或是圖片上標出物件的範圍、辨識物件的類別和物件的機率，且它可被應用於很多產業中，例如:醫學上的腫瘤辨識[33]、交通上的號誌辨識[34]、產品的瑕疵辨識[35]、水果的種類辨識[36]或是成熟度分類[37]等各方面都可以加以應用。

物件偵測有一階段(One stage)式的偵測或和二階段(Two stage)式的偵測。一階段(One stage)式的偵測是指一個模型能同時偵測物件位置和辨識物件，例如:You Only Look Once (YOLO)[13] 或 Single Shot MultiBox Detector(SSD)[38]。二階段(Two stage)式的偵測則是指偵測物件位置和辨識物件是分開二個階段去進行的，例如:Faster R-CNN[23]。

為了可以更有效的找出芒果本身的品質及瑕疵，本研究採用可以框出水果物件的YOLO架構，做為我們芒果的分類模型。由於YOLO模型已由最早的YOLO演進到YOLOv2、YOLOv3，和近期受到學者廣泛討論的YOLOv4。基於上述之說明，本章將先著重在物件偵測之相關研究回顧，接著針對YOLO文獻進行說明，特別在YOLOv3和YOLOv4的研究介紹和說明上。

### 2.1 資料擴增(Data Augmentation)

資料擴增(Data Augmentation) [13][17]是一種可以增加資料集的技術，資料集會影響模型的訓練成效，利用資料擴增可以克服訓練資料不足或避免模型過度擬合的問題。為解決過度擬合和資料不足的問題，研究者通常會

運用一些影像處理的技術來增加資料集。資料集增加的方式，係將影像資料進行各種處理，常見的方式有：(1)水平或垂直翻轉、(2)旋轉、(3)裁減、(4)放大縮小、(5)色彩空間轉換、(6)色調調整、(7)飽和度調整、(8)明亮度調整、(9)增加噪點和(10)混合圖片等，此外也可以利用一些混合和生成的技術來產生新圖片，如：CutMix[39]、Mixup[40]和 Mosaic[13]；如：利用 Generative Adversarial Nets (GAN)生成圖片；如：使用 Conditional Generative Adversarial Network (CGAN)產生健康檸檬和不健康檸檬的圖片[41]，如：利用 Deep Convolutional Generative Adversarial Network (DCGAN)產生肝臟病變的斷層掃描(CT)圖像[42]等。

在本研究中所使用的資料擴增方法[13][17][43]條列如下：

- (1) 將 RGB 圖片轉換到 HSV 色彩空間，利用轉換完的 HSV 圖片，進行色調調整、飽和度調整和明亮度調整。調整色調增強範圍 -0.1 到 0.1，飽和度增強範圍 1 到 1.5 和明亮度增強範圍 1 到 1.5，隨機範圍數字生成增強圖片。
- (2) 隨機調整角度。
- (3) 上下左右翻轉。
- (4) 縮放圖片大小。
- (5) 使用 Mosaic 採隨機縮放的方式將四張圖片拼成同一張。
- (6) 使用 CGAN 擴增圖片。



### 2.1.1 RGB 轉換到 HSV

RGB 是我們日常生活中常見的色彩空間，是由 Red、Blue 和 Green 三原色所組成，將這三種顏色依不同比例可以組成不同的顏色。由於 RGB 色彩空間很容易亮度的影響，進行影像處理時很容易改變其顏色，故在影像處理方面通常會將圖片轉換到不同的色彩空間獲取不同的訊息。最常見的例子就是轉換到 HSV 色彩空間，HSV 色彩空間將顏色的色調(Hue)、飽和度(Saturation)和明亮度(Value)分別拆出來，若我們要進行亮度的調整會更加的容易。

舉 RGB(Red、Blue & Green)圖片轉換到 HSV 色彩空間為例，利用 HSV 方法可以進行色調調整、飽和度調整和明亮度調整。這種方式係將一張圖片中的 RGB 值，即分別為紅色(Red,  $R$ )、綠色(Green,  $G$ )和藍色(Blue,  $B$ )的像素 Pixel 值，透過公式(1)將( $R, G, B$ )顏色範圍從 $[0, 255]$ 調成介於 $[0, 1]$ 的值，命名為( $R', G', B'$ )。而 max\_color 則是選出( $R', G', B'$ )最大值者，相對的 min\_color 選出( $R', G', B'$ )最小值者。本研究運用 Saravanan 等學者[43]轉換公式，利用公式(3)到公式(5)去求得 HSV 轉換後的色彩空間值。根據公式(2)得知 max\_color 對應利用公式(3)得到色相值(Hue,  $H$ )、而公式(4)可求出飽和度值(Saturation,  $S$ )和公式(5)可得到明度值(Value,  $V$ )。

$$\begin{aligned} R' &= R/255 \\ G' &= G/255 \\ B' &= B/255 \end{aligned} \tag{1}$$

$$\begin{aligned} \max\_color &= \text{Max}(R', G', B') \\ \min\_color &= \text{Min}(R', G', B') \end{aligned} \quad (2)$$

$$H = \begin{cases} 60^\circ \times \left( \frac{G' - B'}{\max\_color - \min\_color} \bmod 6 \right), & \text{if } \max\_color = R' \\ 60^\circ \times \left( \frac{B' - R'}{\max\_color - \min\_color} + 2 \right), & \text{if } \max\_color = G' \\ 60^\circ \times \left( \frac{R' - G'}{\max\_color - \min\_color} + 4 \right), & \text{if } \max\_color = B' \end{cases} \quad (3)$$

$$S = \begin{cases} 0, & \text{if } \max\_color = 0 \\ \frac{\max\_color - \min\_color}{\max\_color} = 1 - \frac{\min\_color}{\max\_color}, & \text{otherwise} \end{cases} \quad (4)$$

$$V = \max\_color \quad (5)$$

### 2.1.2 GAN

GAN(Generative Adversarial Network)是由 Goodfellow 等學者[14]在 2004 年提出的對抗生成模型，GAN 的網路架構是由兩個神經網路模型所組成，一個是生成器(Generator)  $G$ ，另一個則是判斷器(Discriminator)  $D$ ，GAN 是同時對這兩個模型  $G$  和  $D$  進行訓練，主要的目的是為了生成出逼真影像，Goodfellow 等學者[14]在論文中實驗了 MNIST、the Toronto Face Database(TFD)、和 CIFAR-10 三個資料集。GAN 運行流程如下(參見架構圖如圖 3)，生成器  $G$  根據隨機噪聲  $z$  生成圖片去欺騙判斷器  $D$ ，讓它誤以為生成的圖片是真的，判斷器  $D$  則是一個二元分類器，其目的為要正確分辨出該圖片是真的還是假的，兩個模型的任務是相反的，所以產生了對抗，再利用判斷器分類的損失反向傳播去更新生成器和判斷器。

GAN 的優化公式係來自[14]，其公式請參見下面的公式(6)。在公式(6)可看出等式後第一項是  $E_{x \sim P_{data}(x)}[\log D(x)]$ ，而  $E_{x \sim P_{data}(x)}$  是指真實資料，這

部分是在判別真實資料;等式後第二項  $E_{z \sim P_Z(z)} [\log(1 - D(G(z)))]$  ,  $E_{z \sim P_Z(z)}$  是指  $G$  產生的假資料，這部分是在判別假資料，越大的  $V$  代表能區分真實和假圖片。整體的 GAN 模型優化值  $\min \max V(G, D)$  如公式(6)。

$$\min \max V(G, D) = E_{x \sim P_{data}(x)} [\log D(x)] + E_{z \sim P_Z(z)} [\log (1 - D(G(z)))] \quad (6)$$

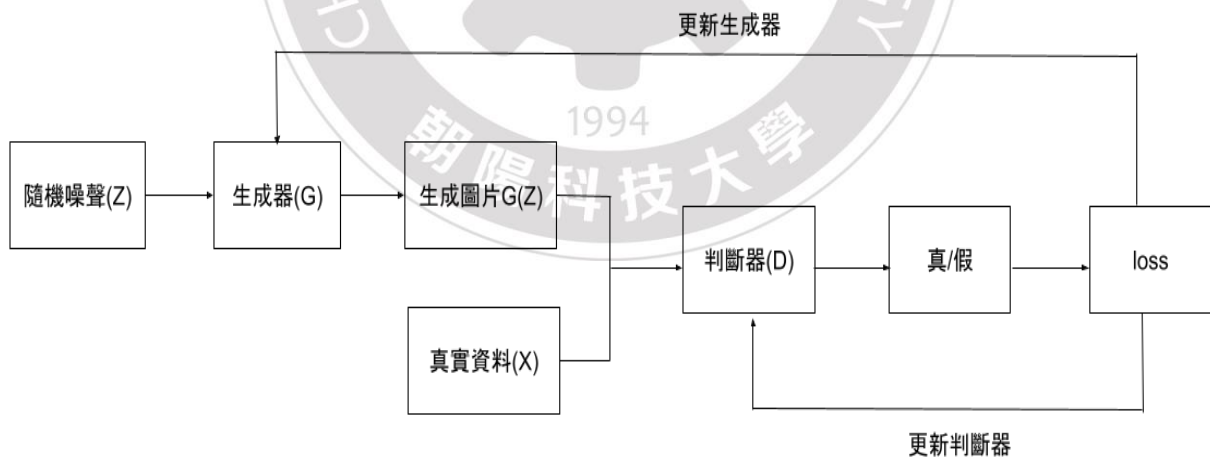


圖 3、GAN 架構圖

由於傳統的 GAN 模型其生成的圖片是隨機的，在生成圖片時，將難以控制生成的可能圖像，為求生成的圖像可以更有效率、更容易接近目標圖像，因此在 GAN 的基礎上，再加入判斷條件來強化模型的圖像生成，這樣的技術便衍生出 CGAN 模型。CGAN 是由 Mirza 和 Osinderoy [17] 在 2014 年提出，Mirza 和 Osinderoy [17] 在 GAN 的基礎上針對生成器和判斷器都加入條件來約束  $y$  ,  $y$  可以是任何類型的輔助信息，例如:類別標籤，用於控制生成的圖片。而 CGAN 模型的收斂則是利用公式(7)的優化公式來達成，此公

式和 GAN 的優化公式雷同但  $x$  由條件機率  $x/y$  來取代，而  $z$  則由條件機率  $z/y$  來取代，詳見公式(7)。

$$\minmax V(G, D) = E_{X \sim P_{data}(x)} [\log D(x|y)] + E_{Z \sim P_Z(z)} [\log (1 - D(G(z|y)))] \quad (7)$$

本研究欲採用 CGAN(Conditional Generative Adversarial Network) [17]來生成芒果圖片，藉此增強原本資料集的影像，之後再結合 YOLOv4 進行分類，以此來強化分類的效果。

### 2.1.3 Mosaic

在本研究在 YOLOv4 中使用 Mosaic[13]資料擴增方法，Mosaic 是在 YOLOv4 論文中提出的資料擴增方法，Mosaic 是參考 CutMix[39]的延伸方法。CutMix 方法是為了提升訓練效率及提升模型準確度，其方法為對兩張圖片進行操作產生新的訓練圖片，將圖片部份區域裁切掉並將裁切的部分利用另一張圖片填充，這種方式可以同時定位兩個目標。Mosaic 方法也是為了提升訓練效率及提升模型準確度，因此 Mosaic 對四張圖片進行隨機縮放、裁減和完成其他資料增強後，拼接成一張圖片和修正目標框到對應的位置，Mosaic 利用這種方式產生新的訓練圖片，可以提高訓練效率、擴增資料集、適應不同尺度的定位能力和豐富檢測物體的背景。

## 2.2 資料增強

資料增強是一種透過影像前處理方法去增加圖片的細節、增加對比度或者是使圖片的邊緣資訊更加明顯的技術。而常見的方式有圖片銳化、直方圖均衡 (Histogram Equalization) 和限制對比度自適應直方圖均衡 (CLAHE)[44]等方法。圖片銳化是一種增強圖片邊緣對比度的方法，可以透過設定濾波矩陣使得圖像邊緣更為清晰。

直方圖均衡和 CLAHE 方法都是對圖片的像素分布進行調整，直方圖均衡是透過延伸圖片中像素分布較為集中的地方，將其均勻分布到直方圖其他區間，藉此改善對比度。但由於直方圖均衡是整體圖片的調整，會將雜訊的對比度也跟著放大，所以有 CLAHE 方法的出現。CLAHE 是一種局部的均衡，將圖片分為小區塊進行處理，並將每個區塊去做直方圖均衡，並根據所設定的閾值去進行分布調整，如果大於設定的閾值會將超出的部分平均分配到其他直方圖區間，以確保直方圖總面積不變。CLAHE 可以透過限制放大去改善雜訊對比度被放大的問題以及改善圖片的對比度。

## 2.3 芒果辨識

有關芒果辨識的研究，Razak 等學者[45]使用模糊圖像分析(Fuzzy Image Analysis)來評斷芒果等級，利用芒果的大小、芒果 RGB 圖片的平均顏色、利用邊緣檢測來評估芒果表皮。該研究將 RGB 的圖片先轉換到灰階，設定閾值將圖片二值化。二值化之後的圖片再利用邊緣檢測方法來找出邊緣，接



著將不屬於芒果的邊緣刪除。將刪除後的圖加以填滿去計算面積，獲得三種顏色成分的平均值和皮膚圖像的邊緣數量。最後則是運用 Fuzzy Inference Rule 將得到的 3 個資訊進行分類，再根據芒果等級對芒果品質進行排名。

Pise and Upadhye [9]將芒果依成熟度分為黃芒果、綠芒果和紅芒果。他們則是運用 RGB 將圖片轉為灰階，設定閾值將圖片二值化，利用二值化的圖片分離背景和芒果，計算芒果大小。芒果的形狀則是使用邊緣檢測方法去計算求得的，該研究利用 RGB 資料、芒果大小和芒果形狀並使用 Naive Byes 進行分類。

有別於一般利用影像處理後獲取的特徵進行分類 Sriram 等學者[46]則使用深度學習(Deep learning)方法中的 Convolutional Neural Networks (CNN) 來分類芒果的種類，其芒果的類型分別為 Totapuri、Alphanso、Malgoba、Raspuri 和 Badami。該研究提出由 CNN 的模型從圖片中自行學習特徵，利用先前 GoogLeNet 在 ImageNet dataset 上訓練的權重並更改最後一層分類輸出使用遷移學習(Transfer Learning)技術，重新訓練最後一層始符合芒果種類分類。

## 2.4 AlexNet

AlexNet 是 Krizhevsky 等學者[47]在 2012 年所提出的一個卷積神經網路，其網路架構是由卷積層、最大池化層以及全連接層所組成，詳細架構如下表 2。Krizhevsky 等學者[47]在前兩個全連接層後都加入了 Dropout 以及

使用資料擴增的技術減少模型的過度擬合，Dropout 會根據設定的機率丟棄神經元並使用 ReLU 當作激活函數，使得模型收斂速度較快。

表 2、AlexNet 架構表

Type	Kernel Size	Stride
Convolution	$11 \times 11$	$4 \times 4$
Maxpooling	$3 \times 3$	$2 \times 2$
Convolution	$5 \times 5$	$1 \times 1$
Maxpooling	$3 \times 3$	$2 \times 2$
Convolution	$3 \times 3$	$1 \times 1$
Convolution	$3 \times 3$	$1 \times 1$
Convolution	$3 \times 3$	$1 \times 1$
Maxpooling	$3 \times 3$	$2 \times 2$
Fully connected	4096	
Dropout(0.5)		
Fully connected	4096	
Dropout(0.5)		
Fully connected	1000	

本研究欲將圖片經前處理定位芒果位置，並移除大部分背景。接著將完成前處理的圖片使用資料擴增技術增加 1 倍，送入 AlexNet [47] 模型進行芒果圖片分類。

## 2.5 YOLO

YOLO(You Only Look Once)[27]是一種深度學習神經網路，目前共有 4 個版本，YOLO 是由 Redmon 等學者[27]在 2016 年提出的一個新的物件檢測方式，它運用了神經網路並可以直接從完整的圖像中預測邊界框和類別的機率。

YOLO 的卷積網路架構是參考 GoogleNet 模型，YOLO 的網路架構是由 24 卷積層(Convolutional Layer)和 2 層全連接層(Fully Connected Layer)所組成的，和 GoogleNet 不同的地方在於 Redmon 等學者在  $3 \times 3$  的卷積層前面用  $1 \times 1$  的卷積層來減少 filter 數量。YOLO 的激活函數係採用 Leaky ReLU，而最後一層則是採用 Linear activation。YOLO 直接預測 bounding box 的座標值和類別，並使用 Non-max suppression 來消除重複檢測的預測框。

YOLOv2 是由 Redmon 和 Farhadiy 學者在 2017 年[28]提出的改良版，YOLOv2 提出 Darknet-19 當作新的特徵提取器，網路架構與 VGG16 相似，包括 19 個卷積層和 5 個最大池化層(Maxpooling)，主要採用  $3 \times 3$  卷積， $2 \times 2$  最大池化層，在每層卷積層的後面都加入批量標準化(Batch Normalization)改善模型的收斂，最終採用全局平均池化(Global Average Pooling)做預測。YOLOv1 採用  $224 \times 224$  的圖片分類器，但因為解析度較低，不利於檢測模型，因此 YOLOv2 提高解析度至  $448 \times 448$ 。YOLOv2 移除 YOLOv1 網路的全連接層(Fully Connected Layer)，改用 anchor box 來預測邊界框。跟 YOLOv1 不一樣的是 anchor box 不是直接預測 bounding box 的座標值，而是預測 anchor box 的 offset (座標的偏移值) 與 confidence scores (置信度)。為了選取合適的 anchor box，使得模型更容易學習並預測出準確的 bounding box，因此 YOLOv2 採用 K-means 聚類方法對訓練集中物體的 bounding box 做了聚類分析。



YOLOv3 是由 Redmon 和 Farhadiy 學者[29]在 2018 年再次提出的新版本模型。YOLOv3 提出 Darknet-53 當作新的特徵提取器，從第 0 層到 74 層，一共有 53 層卷積層(Convolutional Layer)，其餘為 Resnet 層。跟 Darknet-19 相比，Darknet-53 去移除所有的最大池化層、增加了更多的 1x1 和 3x3 的卷積層，但因為加深網路層數容易導致梯度消失或爆炸，所以 Redmon 和 Farhadiy 學者在 Darknet-53 加入了 Resnet 網路的殘差架構(Residual Network)來解決梯度的問題。YOLOv3 使用 Feature Pyramid Network (FPN)的方法[48]，採用多尺度的 feature map 對不同大小的物體進行檢測，提升小物體的檢測能力。YOLOv3 使用 Logistic regression 來預測每個 bounding box 的 confidence，以 bounding box 與 ground truth 的 IoU(Intersection over union)做為判定標準。

YOLOv4 是由 Liao 等學者[13]在 2020 年所提出的進階模型，Liao 等學者將一個物體檢測器分成四個部分：(一)輸入(Input):輸入的圖片(二)骨幹(Backbone):獲取圖片特徵(三)頸(Neck):提取不同層級的特徵以及融合特徵和(四)頭(Head):檢測目標框和類別的檢測器。

YOLOv4 是架構由 CSPDarknet53、Spatial Pyramid Pooling(SPP)、Path Aggregation Network(PAN)和 YOLOv3 head 所組成，其架構說明可參考[49]其架構圖請參見圖 4。YOLOv4 採用 CSPDarknet53 當作骨幹來獲取特徵，將原先 YOLOv3 的 Darknet53 加入了 Cross-Stage Partial

Network(CSPNet)[50]，CSPNet 將特徵圖分成兩個部份，一部份經過 dense layer 和過度層後再與另一部份連接，過度層包含卷積層和池化層，這種方式可以優化網路中重複梯度的問題，並減少計算量。YOLOv4 使用 Mish[51] 當作激活函數，Mish 的公式為  $f(x) = x \tanh(\ln(1 + e^x))$ ，Mish 是連續可微的非單調激活函數，跟 Relu 相比，Mish 的梯度更平滑，並且在負值時允許有較小的負梯度，可以穩定網路梯度，具有更好的泛化能力。在 YOLOv4 模型中，採用了 SPP [52]和 PAN[53]當作頸(neck)來提取不同層級的特徵，SPP 將特徵圖透過不同大小的最大池化後的特徵圖，最後將特徵圖串接成固定尺寸的特徵圖，可以更好的進行特徵融合。PAN 則可以將不同骨幹和不同檢測器的參數統整。而 YOLOv4 在 head 的部份延用 YOLOv3 head[29]當作預測類別及 box 的檢測器。

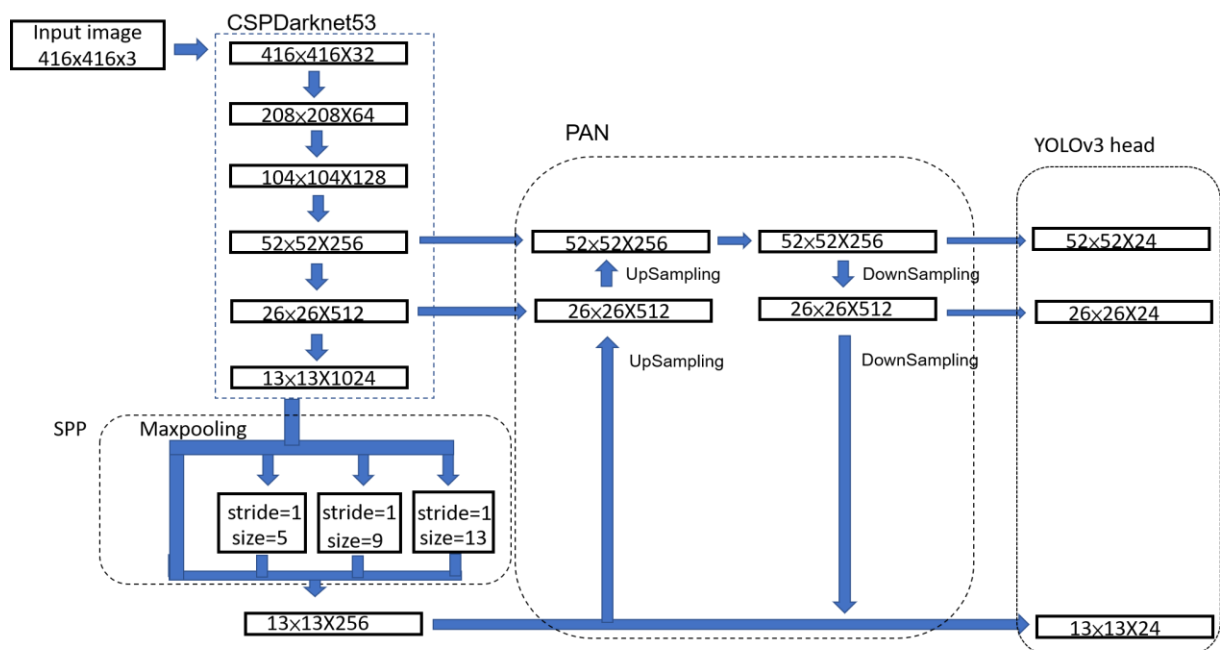


圖 4、YOLOv4 架構圖

為了讓訓練可以更適合在單個 GPU 上訓練，YOLOv4 提出了新的資料增強方式，分別是 Mosaic 和 Self-Adversarial Training (SAT)。Mosaic 採隨機縮放裁減的方式混合 4 張訓練圖片，因此不需要大量的 mini-batch size。SAT 則是分成兩階段，第一階段神經網路改變圖片本身，而不修改權重，透過這種方式對自身對抗式攻擊，製造出圖片沒有檢測對象的錯覺，這種方式可以增加樣本的訓練難度。第二階段就是利用修改的圖片進行訓練。先前使用 IoU (Intersection over union) loss 來 Bounding Box Regression Loss，IoU 係找出由模型所預測的 bounding box 和實際的物件之 bounding box 的交集除以聯集，如公式(8):

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} = \frac{|A \cap B|}{|A \cup B|} \quad (8)$$

但上述這種計算方式遇到預測框和目標框不相交時，IoU 為 0，如此將無法反映兩個框之間的遠近。為了解決這樣的問題，Rezatofighi 等學者[54]在 2019 年提出了 Generalized Intersection over Union (GIoU)[54]找出同時包含預測框和真實框的最小包圍框  $C$ ，來解決這個問題，如公式(9):

$$GIoU = IoU - \frac{|C - A \cup B|}{|C|} \quad (9)$$

GIoU 需要先找出最小包圍框  $C$ ，這種方式會消耗大量時間，影響收斂速度。Zheng 等學者[55]在 2020 年提出 Distance Intersection over Union (DIoU) 和 Complete IOU (CIoU)，DIoU 利用預測框的中心點  $b$  和目標框的中心點

$b^{gt}$ ，去算他們的歐式距離  $p$ ， $c$  表示時包含預測框和真實框的最小包圍框的對角線，如公式(10)，Zheng 等學者[55]認為一個好的目標框應該同時考慮重疊面積、中心點和縱橫比，所以也在 DIoU 的基礎上加上  $av$ ，提出即利用  $a$ (權重函數)， $v$ (衡量長寬比的相似性)提出了 CIoU，計算公式如(11):

$$DIoU = IoU - \frac{p^2(b, b^{gt})}{c^2} \quad (10)$$

$$CIoU = IoU - \frac{p^2(b, b^{gt})}{c^2} + av \quad (11)$$

根據上述 IoU 的演進 YOLOv4 改採用 CIoU loss 來當作 Bounding Box Regression Loss，而原本 YOLOv3 使用 Non-Maximum Suppression(NMS)來消除重疊的預測框，到了 YOLOv4 模型，則是改使用 DIoU 去做 NMS。

## 2.6 AI CUP

AI CUP 為台灣教育部所發起的人工智慧技術及應用人才培育計畫中的其中一項子計畫，主旨是希望能透過競賽，培養學生解決 AI 問題的經驗與能力，並導入人工智慧技術與觀念至教育體系，期望將來在投入實習場域進行實戰時，能學以致用，培育更多 AI 人才。

AI CUP 目前係由工研院巨量資訊科技中心(巨資中心)與工研院產業科技國際策略發展所(產科所，ISTI)共同開發人工智慧共創平台(<https://aidea-web.tw/>)，在該平台裡已執行 AI CUP 從 2018 年到 2021 共舉辦了 10 個不同領域的競賽，競賽類型包括:影像辨識方面的愛文芒果影像辨識和水稻無

人機全彩影像植株位置自動標註與應用；自然語言方面的生醫論文自動分析和音訊方面的和弦辨識競賽和歌聲轉譜競賽等。其中愛文芒果影像辨識是在 2020 年所舉辦的，主旨是希望於愛文芒果產業導入自動化、精準化概念的 AI 影像辨識技術。

AI CUP 的愛文芒果影像辨識係由 BIIC Lab 與台灣瓦克國際股份有限公司合作，耗時數年收集愛文芒果影像資料，希望在未來利用資料庫建立一套自動芒果篩果系統，改善採收後處理技術。愛文芒果採收後依篩選品質訂有 A、B、C 三個等級，這樣的分類結果，將芒果依序做為出口用、內銷用、加工用，所提供的芒果資料集包含訓練集與測試集，在訓練集有 5,600 張圖片，測試集有 800 張圖片。

由愛文芒果的資料集圖片，可以觀察出，芒果各等級的標記很難找到可以定義明確的規則，只能約略看出可能的分類方向，我們可以由圖 5 的範例來觀察這樣的分類困難。在圖 5 中我們可以觀察到，A 級的芒果應該偏向是毫無瑕疵且顏色偏紅且均勻、B 級有些微瑕疵但顏色小部份偏橘，C 級應有瑕疵斑點，這三種等級的芒果圖片可以參見圖 5。不過，當仔細查看資料庫的芒果圖片後，發現並不全然符合上述分級規則，有部份芒果圖片在標記上是混淆不易理解的。



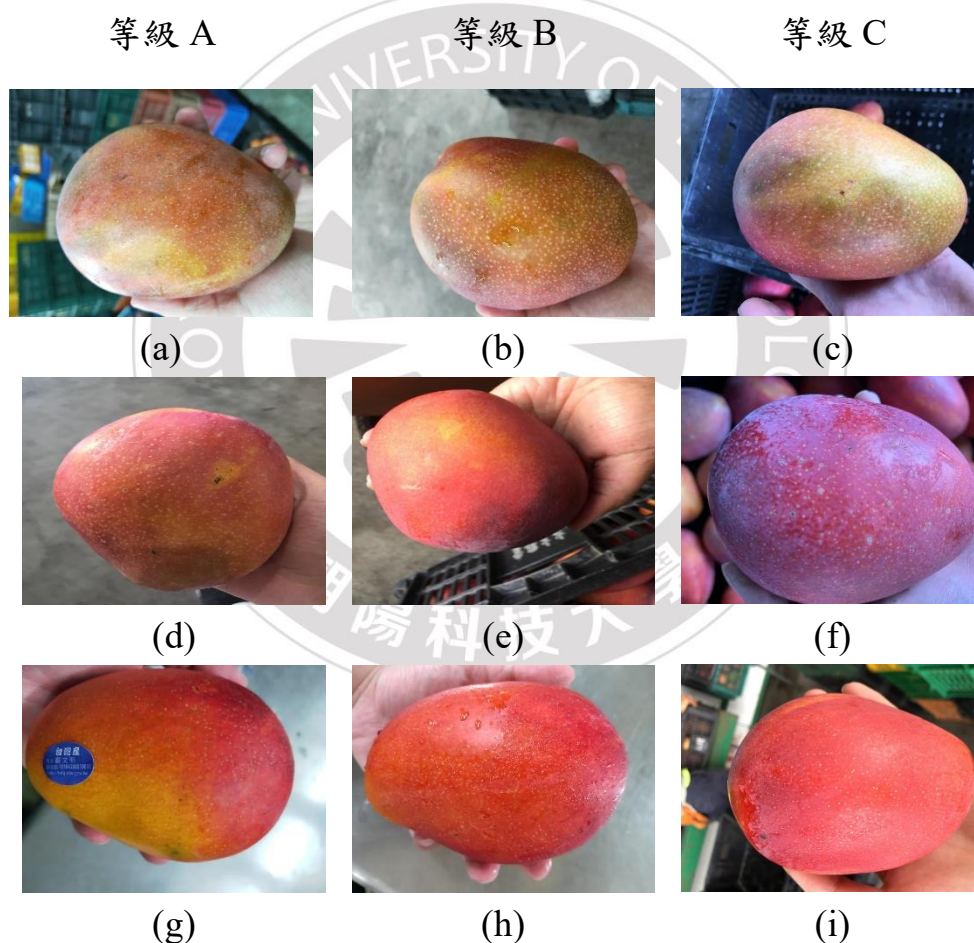


圖 5、芒果各等級的問題範例圖

由圖 5 可以看出，分辨的困難點是芒果顏色的分布均勻度以及缺陷的類型，另外就是會有一些芒果圖片，它們被歸屬在不同等級但芒果的圖片卻很相似，因此即便是機器學習模型，也很容易誤認為成錯誤的等級。例如：圖 5 的(a)(b)(c)三張圖看起來十分相似，肉眼也難以區分是不同等級，其中(a)(b)兩圖均沒有缺陷，差異點在於顏色分布，而(c)有缺陷但圖片不明顯。又例如：圖 5 的(d)是 A 類但卻有看到瑕疵，而圖 5 的(f)和(i)看起來似乎沒有缺陷、顏色分布也十分均勻，但卻標註為 C 級。又例如：圖 5 的(g)標記為 A 級，但和圖 5 的(h)(標記為 B 級)相較起來，似乎(h)更像 A 級，但卻被標記

為 B 級。由圖 5 的示例可以得知，AICUP 2020 的芒果資料庫，其等級分類似乎無法完全明確可辨識，因此在模型學習和辨識上也會產生誤差，這也就是為什麼 AI CUP 2020 年的競賽最高準確率排名僅有 83%。

基於上述的說明，本研究會針對圖片進行亮度調整、預先篩選訓練圖片等圖片前處理、使用一部分的資料集訓練 CGAN 模型，利用 CGAN 模型生成圖片進行圖片擴增和使用不同的圖片增強方法進行實驗以及比較。

## 第參章 研究方法

本研究主要以擷取台灣 AI CUP 競賽[24]中所提供的芒果資料集，利用 YOLO 模型來建立一套影像辨識自動篩選芒果系統。本研究之資料處理及模型建立訓練步驟如下面圖 6 和圖 7。

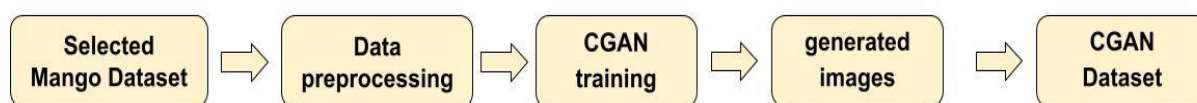


圖 6、本研究之 CGAN 流程圖

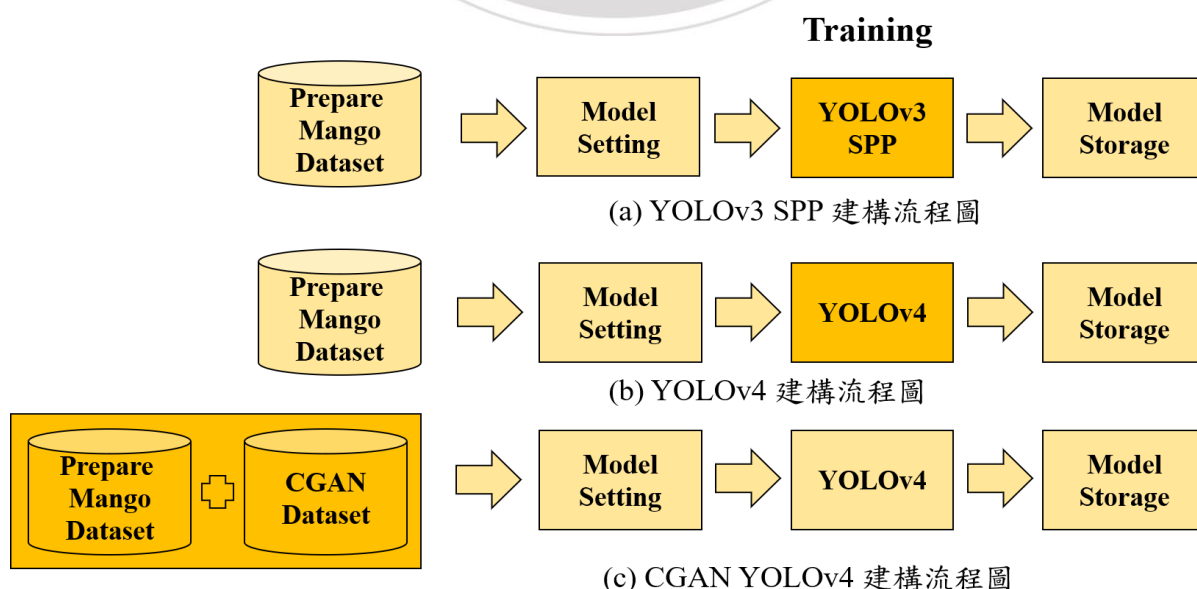


圖 7、本研究提出三種芒果辨識模型之建構流程圖

本研究的資料係由 AI CUP 競賽[24]下載的芒果影像，芒果資料集透過標註工具 labelImg(<https://github.com/tzutalin/labelImg>)標註目標框後，利用 K-Means 分群法重新計算 anchor box，設定 YOLOv4 和 YOLOv3 SPP 模型要用的資料增強方法和輸入圖片大小、類別個數和根據類別的層數輸出。



接著本研究將芒果資料集等級 A、等級 B 和等級 C 的圖片輸入到參數設定好的 YOLOv4 和 YOLOv3 SPP 訓練。為了找到更佳的辨識模型，本研究共建置了 3 個不同的 YOLO 模型，分別為：

- (1) 單純只使用 YOLOv4 模型進行芒果辨識，
- (2) 運用了 YOLOv3 SPP 模型進行芒果辨識，以及
- (3) 結合 CGAN 和 YOLOv4 (在本實驗中簡稱 CGAN YOLOv4)來進行影像資料增強之芒果辨識。

上述三種 YOLO 模型經過編譯及修改模型設定檔分別建置後，亦將影像資料匯入並分別訓練這三個模型，訓練完成後再將權重個別加以儲存，接著利用已訓練好之個別模型的權重，進行模型之測試資料集預測，藉由所產生的預測框和分類，對這三個模型進行模型成效驗證。並將三個模型的最終成果與台灣 AICUP 比賽的 baseline 還有比賽最佳成績進行比較。以下針對本研究的方法再進行詳細的說明。

### 3.1 芒果資料集

本研究的資料係取自 AI CUP 2020:愛文芒果影像辨識雙項競賽[24]的資料集，資料下載後，根據資料集所提供的標籤，本研究使用 labelImg[56]標註目標框和芒果類別，將芒果分為 3 類，分別為 A、B 和 C 3 個等級(如

圖 8)。由於資料庫裡的芒果影像圖片大多為手持芒果拍攝或是放在芒果籃裡拍攝，因此這些影像資料必須再加以處理。

圖 8 所示為等級 A、等級 B 和等級 C 各兩張範例，在圖 8 可以明顯地看到等級 A 的顏色分布均勻偏紅色，且沒有缺陷；等級 B 和等級 A 相比其顏色較不紅，可能有小缺陷，B 類的顏色除了紅色還有黃色、橘色和黃綠色等；等級 C 芒果則可以很清楚看到明顯的缺陷，至於顏色則偏紅、偏黃、偏橘、偏綠等都有。



圖 8、訓練資料集中的(a)等級 A、(b)等級 B 和(c)等級 C 範例圖

本研究先將資料分成訓練和測試兩個資料集，分別有不同大小的圖片，訓練集有 5,600 張，測試集有 800 張。其分級標準的規則如表 3，表 3 為比賽方定義的規則，不過經過據資料集的檢視和觀察後，發現資料集裡面有些

圖片所標示的等級，與下面的規則描述並不相符，因此這個規則並非十分準確僅能做為參考。

表 3、分級標準的規則

A	完全沒有瑕疵，色澤均勻漂亮。
B	色澤不均，瑕疵範圍若可以一個拇指蓋住，還是可以賣的狀況，都是 B。
C	不能賣，只要有黑斑、炭疽病、爛，都算是 C。

本研究將資料集裡的影像資料進行類別分布計算(如表 4)，將訓練集各類數量加總後平均數( $\mu$ )約為 1,866 張，A 類比平均值少 74 張，B 類比平均值多 202 張，C 類較平均值少 126 張，由上述的數據可以得知資料集各類張數和平衡張數差異不大；同樣的測試集加總後平均數( $\mu$ )約為 266 張，A 類較平均值少 23 張，B 類較平均值多 27 張，C 類較平均值少 2 張，因此測試資料集各類張數和平均張數亦差異不大。整體而言本研究所要學習及預測的芒果資料集，還算是一個平衡的資料集(Balanced Dataset)。

表 4、訓練和測試影像資料張數分布 (均值 Mean=1,866)

Grade	Training ( $\mu \approx 1,866$ )	Testing( $\mu \approx 266$ )
A	1,792	243
B	2,068	293
C	1,740	264
Total	5,600	800

### 3.2 參數設定步驟

在資料擷取、標記及分配訓練及測試集後，接下來要對本研究所運用之模型進行參數設定。茲說明如下。

Step 1. Prepare the Mango Image:  $M = \{m_1, m_2, \dots, m_l, \dots\}, l \in N$ 。

Step 2. 使用 K-Means 來重新計算 anchor box

(1) 設定要分類的群數  $K$ ，產生  $K$  個中心框，每個中心框有  $w_i$  跟  $h_i$  分別代表寬和長，即  $(w_i, h_i), i \in \{1, 2, \dots, K\}, K \in N$ 。

(2) 計算每個框和中心框的  $IoU$  值，利用  $IoU$  值來計算距離  $d$ ， $d=1-IoU$ ，根據  $d$  的大小來分配每個框屬於哪個中心框。

(3) 根據分配完的結果，重新計算中心框，每個中心框利用屬於該中心框的框，計算寬和長的平均值當作新的中心框， $n$  為屬於該中心框的數量， $w_i = \frac{1}{n} \sum w_i, h_i = \frac{1}{n} \sum h_i$

(4) 持續步驟 2 和 3，直到每個框不再變更群組時。

Step 3. 設定模型參數

(1) 設定圖片大小， $s \times s, s \in N$ 。(本研究設  $s = 416$ 。)

(2) 設定類別數  $numClass$ 。

(3) 根據  $numClass$  設定層數， $filter = (3 \times (5 + numClass))$

若以輸入為  $416 \times 416 \times 3$  的 RGB 圖片為例

Set the input  $R_j, j \in \{1, 2, \dots, 173056\}$

Set the input  $G_j, j \in \{1, 2, \dots, 173056\}$

Set the input  $B_j, j \in \{1, 2, \dots, 173056\}$

Set  $numClass=3$ 。

Set the three filter before YOLOv3 prediction based on  $numClass$

$$(3 \times (5 + 3)) = 24$$

Step 4. 將資料集經過下面介紹的資料處理步驟後，準備輸入到各個模型中。

以下再針對三種不同的模型加以說明其建置之內容與流程。

### 3.3 YOLOv4 架構

本研究的 YOLOv4 架構係參考[49][57]所繪製的模型圖，下面圖 9 為本研究重新繪製 YOLOv4 的簡易架構運行流程圖，由下面圖 9 可以看到 YOLOv4 是由 CSPDarknet53、SPP、PAN 和 YOLOv3 head 所組成的模型。YOLOv4 模型先由圖片輸入，接著經由 CSPDarknet53 獲取特徵，藉由 SPP 和 PAN 提取不同層級的特徵後輸出特徵圖，最後利用 YOLOv3 Head[29]當作預測類別及預測框的檢測器，輸出的預測框會透過 Non-Maximum Suppression (NMS)後再輸出最後的結果。



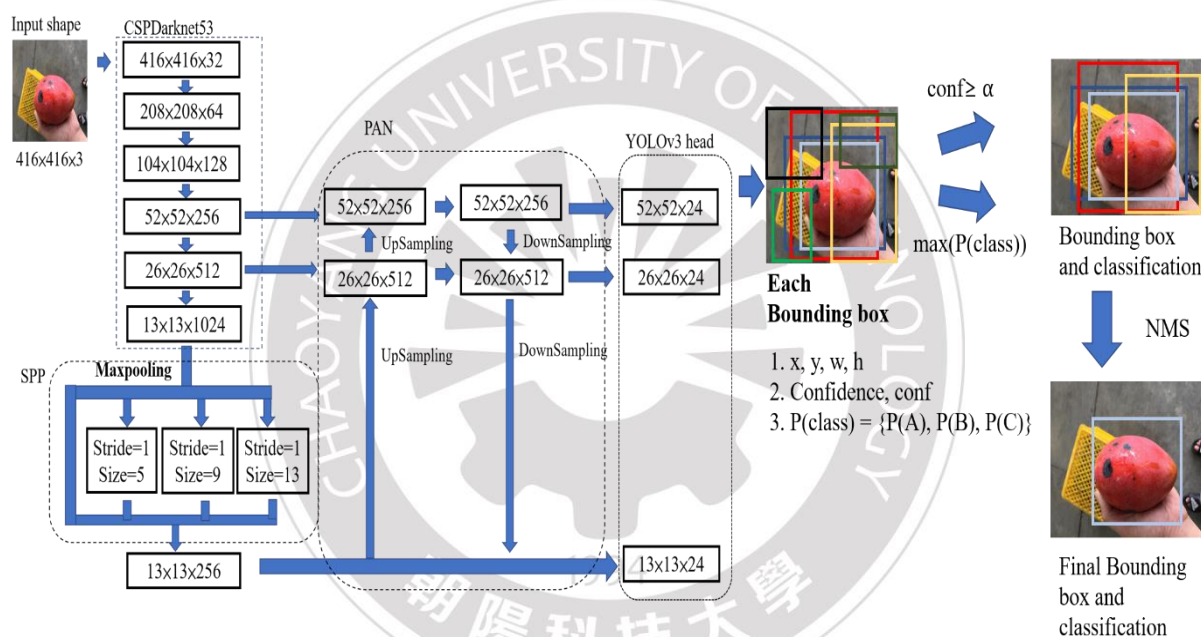


圖 9、YOLOv4 的簡易架構運行流程圖

YOLOv4 的模型操作流程如下：

- (1) 輸入圖片大小為  $s \times s \times 3$  ( $s=416$ )，將輸入切割為  $13 \times 13$ 、 $26 \times 26$  和  $52 \times 52$  的網格，如果偵測的物體的中心落在網格內，這個網格就要負責去偵測該物體。每個網格要負責預測 3 個 bounding boxes 和屬於各類別的機率。
- (2) 每個 bounding box 會預測出  $x, y, w, h$  和信心度  $conf$  五個參數和每個類別的機率  $P(class) = \{P(A), P(B), P(C)\}$ 。
- (3) 如果  $conf \geq \alpha$ ， $\alpha$  為設定閾值(threshold)，代表有物體。相反代表沒有物體。
- (4) 選擇最大可能的類別當作類別標籤。



(5) 通過 Non-Maximum Suppression (NMS) 消除重疊的預測框並輸出最後結果，NMS 透過設定閾值(threshold)，將所有預測框依據信心度  $conf$  排序，選擇信心度  $conf$  最高的框與其他框計算  $IoU$ 。如果  $IoU > \beta$ ,  $\beta$  是一個設定的閾值(threshold)，代表兩個框是框住同一個物體，將信心度較小的設為 0，刪除框，如果  $IoU < \beta$  代表框住的是不同物體，留下框。從留下來的框重複上述步驟直到沒有框為止。

(6) 訓練完成儲存權重。

### 3.4 驗證方式

圖 10 為測試流程圖。YOLOv3 SPP、YOLOv4 和 CGAN YOLOv4 模型訓練完成後，利用所產生的權重對訓練集和測試集進行驗證。分別利用 Precision 精確度(如公式 12)、Recall 召回率(如公式 13)、F1-score(如公式 14)、Average Precision 平均精度(如公式 15)、Mean Average Precision (MAP) (如公式 16)和  $IoU$ (如公式 8)來驗證模型並利用 Weighted Average Recall (WAR) (如公式 17)與 AI CUP 比賽的 baseline 模型和最佳模型進行比較。

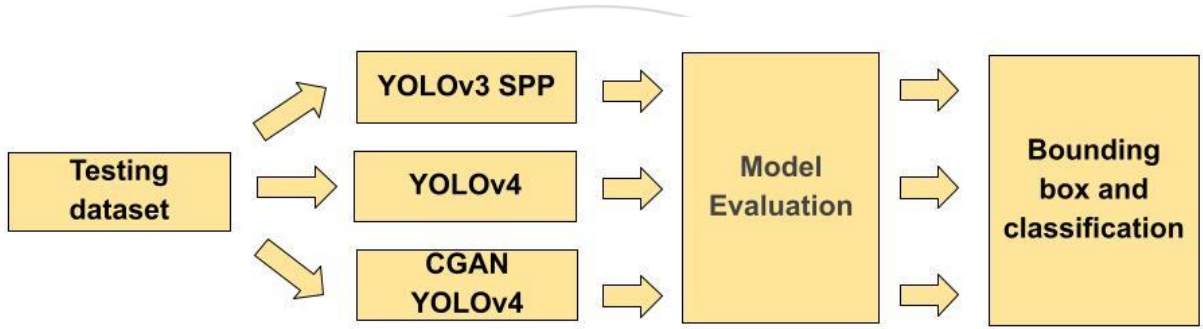


圖 10、測試流程圖

Precision 計算主要在判斷為真的情況下，有多少是真的如公式(12)。  
 Recall 主要在判斷為真的情況下，有多少被正確判斷出來如公式(13)。其中  
 $TP$  代表是真正的類別模型也被模型判斷是該類別， $FP$  代表原本不是該類別  
 模型，卻被判斷為該類別， $FN$  代表是原為該類別但模型沒有判斷出來。  
 而 F1 Score 則是  $Precision$  和  $Recall$  的加權平均值如公式(14)。

$$Precision = \frac{TP}{TP + FP} \quad (12)$$

$$Recall = \frac{TP}{TP + FN} \quad (13)$$

$$F1-score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (14)$$

Average Precision(AP)代表平均精度，一般定義是找到精度-召回曲線下的面積如公式(15)。 $P$  代表 Precision， $r$  代表排序機率。MAP 就是所有類別的 AP 加總平均如公式(16)，其中  $numClass$  代表類別，本研究將芒果分為三類，所以  $numClass=3$ 。

$$AP = \int_0^1 P(r) dr \quad (15)$$

$$MAP = \frac{\sum_{i=1}^{numClass} AP}{class} \quad (16)$$

加權平均召回率(Weighted Average Recall ;WAR)，可以依據資料集類別的分布情形分配權重， $w(i)$ 為每個類別真實數量/所有類別總數， $Recall(i)$ 為每個類別的 Recall 值，其中  $numClass$  代表類別，本研究將芒果分為三類，所以  $numClass=3$ ，計算如公式(17)。

$$WAR = \sum_{i=1}^{numClass} w(i) \times Recall(i) \quad (17)$$

## 第肆章 實驗與結果

本章節會分別介紹本研究所提的四種模型並討論結果。實驗順序為 AlexNet、YOLOv3 SPP、YOLOv4 以及 CGAN YOLOv4。本研究的資料係由 AICUP 2020:愛文芒果影像辨識雙項競賽[24]網站裡所提供，資料下載後，訓練集共有 5,600 張，測試集共有 800 張，輸出的標籤結果為 A、B 和 C 三個等級。本實驗會根據圖 7 的訓練流程圖分別實驗 YOLOv3 SPP、YOLOv4 以及 CGAN YOLOv4 三個模型，並利用 3.4 節所提到的驗證方式來驗證找出最佳模型。在 4.3 節會將最佳模型進行更進一步的實驗，將圖片經過調整亮度、圖片篩選結合 CGAN 資料集，並實驗直方圖均衡、圖片銳化和限制對比度自適應直方圖均衡 (CLAHE) 方法來增強圖片改善模型成效進行訓練。

### 4.1 AlexNet 實驗與結果

首先本研究先以 AlexNet 模型進行實驗，資料前處理流程請參見圖 11 和圖 12。本實驗首先為了要先定位出芒果的位置，本研究會先利用 YOLOv3 模型進行芒果位置定位，並移除大部分的背景。實驗中發現在芒果較大的圖片都無法定位，因此本研究再使用 Sobel 找出邊緣輪廓，篩選出最大輪廓，再裁出矩形區域，並調整圖片大小為  $224 \times 224 \times 3$ 。利用資料擴增方法將圖片經隨機水平翻轉或垂直翻轉產生新資料，經由上述做法產生新訓練集  $5,600(\text{裁減後}) + 5,600(\text{裁減後資料擴增}) = 11,200 (\text{張})$ 。

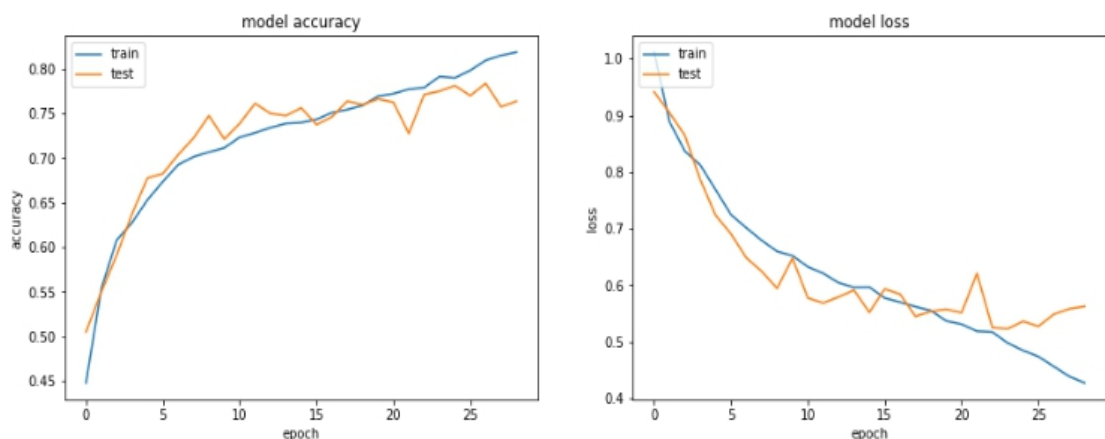


圖 11、先用 YOLOv3 進行芒果定位，此為找到芒果之情形



圖 12、先用 YOLOv3 但沒有定位到芒果的處理方式

資料定位好並經前述處理後，再將資料輸入到 AlexNet 模型進行訓練，並取得如圖 13 的訓練結果，圖 13(a)可以觀察到藍色線(train)在訓練集達到 84%的準確度，在橘色線(test)測試集的準確度達到 76%。圖 13 (b)可以觀察到藍色線(train)在訓練時的損失是穩定下降，而在橘色線(test)測試集的損失到約 22 epoch 時逐漸往上升呈現模型過度擬合的情形，所以此實驗只保留 22 epoch 之前的參數當作最終模型。



(a) AlexNet 模型準確率折線圖

(b) AlexNet 損失過程折線圖

圖 13、(a)AlexNet 模型準確率及(b)AlexNet 損失過程折線圖

由於 AlexNet 模型在芒果資料集上的成效不是很好，故本研究接著實驗下一小節所提到的三個 YOLO 模型。

## 4.2 YOLOv3 SPP、YOLOv4 以及 CGAN YOLOv4 實驗與結果

labelImg[56]是一個圖形介面的標註工具，可以存成 PASCAL VOC 格式、YOLO 格式和 CreateML 格式。利用資料集提供給的等級標籤，我們使用 labelImg[56]人工標註目標框和類別，如圖 14 所示產生 YOLO 格式的標籤，一個標註包含類別、中心 x、中心 y、寬 w 和長 h，資料以空白間隔，例如：[2 0.543155 0.519841 0.913690 0.835317]，製作訓練集和測試集。YOLOv4 和 YOLOv3 SPP 使用作者 Alexey Bochkovskiy [58]所提供的 Darknet，到 Darknet[58]網站下載後，經過編譯產生 Darknet 模型。據此模型，進行三個實驗，分別是 YOLOv3 SPP、YOLOv4 和 CGAN 結合 YOLOv4，參數設定和資料處理說明如下。



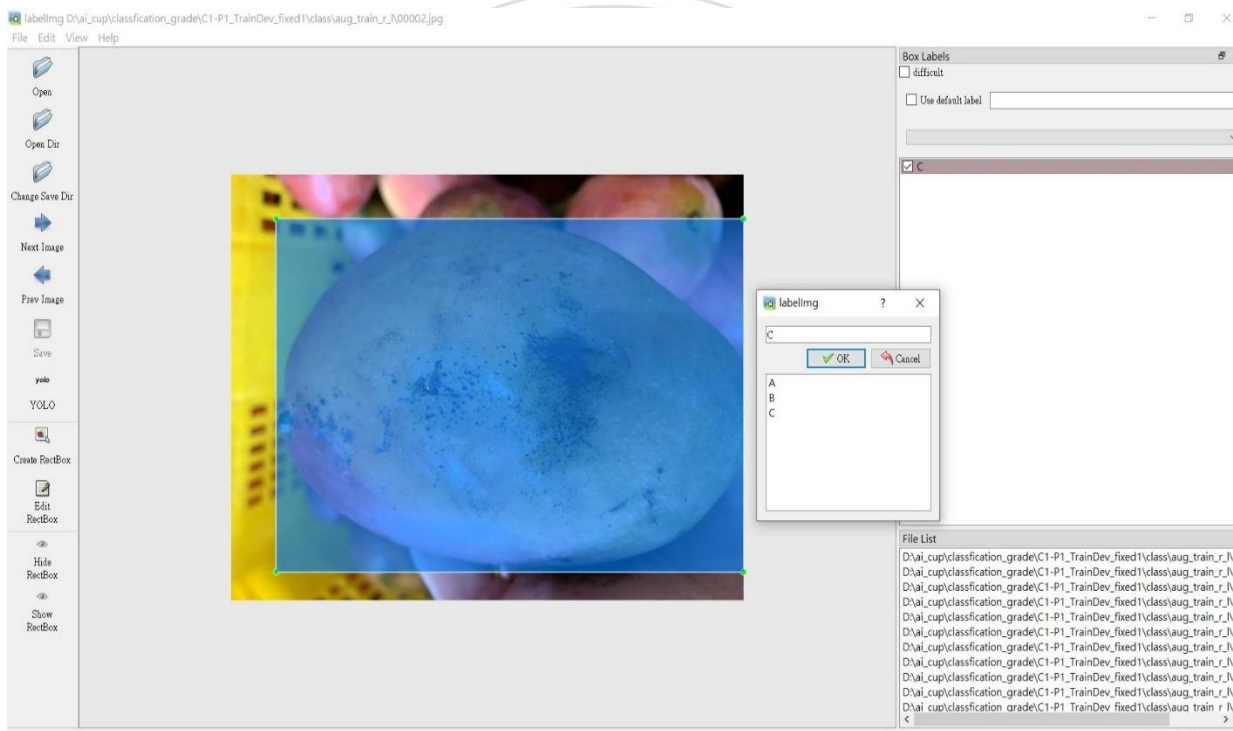


圖 14、使用 labelImg 標註資料集之介面

### [實驗 1]

實驗 1 為 YOLOv3 SPP 結合原始的芒果資料集進行訓練，其參數設定為圖片大小  $416 \times 416 \times 3$ ，saturation 設為 1.5，exposure 設為 1.5，hue 設為 0.1，momentum 設為 0.949，decay 設為 0.0005，learning\_rate 設為 0.001，訓練圈數設為 6000，將 numClass 設為 3 類，filter 設為  $(3 + 5) \times 3 = 24$ 。設定在訓練時所用到的資料增強技術，飽和度 saturation 設為 1.5，調整為 1 倍到 1.5 倍、exposure 曝光度設為 1.5，調整為 1 倍到 1.5 倍和色調 hue 設為 0.1，調整 0.1 到 0.1，上述隨機範圍增強產生圖像和  $\text{angle}=0$ 。根據我們的資料集利用 K-Means 基於預設值設定群數為 9 群，重新計算產生新的 anchor (241,241,

225,321, 274,267, 298,295, 261,362, 314,324, 342,333, 356,358, 379,386)更換預設 anchor。

#### [實驗 2]

實驗 2 為 YOLOv4 結合原始芒果資料集進行訓練，其參數設定為圖片大小 416x416x3，saturation 設為 1.5，exposure 設為 1.5，hue 設為 0.1，momentum 設為 0.949，decay 設為 0.0005，learning\_rate 設為 0.001，訓練圈數設為 6000，將 numClass 設為 3 類，filter 設為  $(3 + 5) \times 3 = 24$ 。設定在訓練時所用到的資料增強技術，飽和度 saturation 設為 1.5，調整為 1 倍到 1.5 倍、exposure 曝光度設為 1.5，調整為 1 倍到 1.5 倍和色調 hue 設為 0.1，調整 0.1 到 0.1，上述隨機範圍增強產生圖像、mosaic=1 和 angle=0。根據我們的資料集利用 K-Means 基於預設值設定群數為 9 群，重新計算產生新的 anchor (241,241, 225,321, 274,267, 298,295, 261,362, 314,324, 342,333, 356,358, 379,386)更換預設 anchor。

#### [實驗 3]

實驗 3 為 CGAN YOLOv4 結合 CGAN 資料集和原始芒果資料集進行訓練，其參數設定為圖片大小 416x416x3，saturation 設為 1.5，exposure 設為 1.5，hue 設為 0.1，momentum 設為 0.949，decay 設為 0.0005，learning\_rate 設為 0.001，訓練圈數設為 6000，將 numClass 設為 3 類，filter 設為  $(3 + 5) \times 3 = 24$ 。設定在訓練時所用到的資料增強技術，飽和度 saturation 設為 1.5，調

整為 1 倍到 1.5 倍、exposure 曝光度設為 1.5，調整為 1 倍到 1.5 倍和色調 hue 設為 0.1，調整-0.1 到 0.1，上述隨機範圍增強產生圖像、mosaic=1 和 angle=0。根據我們的資料集利用 K-Means 基於預設值設定群數為 9 群，重新計算產生新的 anchor (207,287, 257,250, 286,281, 251,351, 309,309, 330,336, 411,303, 357,351, 377,384)更換預設 anchor。上述三個實驗的模型參數設定統整如表 5。

表 5、各模型參數設定

	YOLOv3 SPP	YOLOv4	CGAN YOLOv4
圖片大小	416×416×3		
saturation	1.5		
exposure	1.5		
hue	0.1		
angle	0		
mosaic	0	1	1
momentum	0.949		
decay	0.0005		
learning_rate	0.001		
訓練圈數	6000		
numClass	3		
filter	24		
anchor	(241,241, 225,321, 274,267, 298,295, 261,362, 314,324, 342,333, 356,358, 379,386)		(207,287, 257,250, 286,281, 251,351, 309,309, 330,336, 411,303, 357,351, 377,384)

由於實務上芒果各個角度都可能有缺陷處，但所使用的訓練資料集無法將各種角度都呈現出來，因此為了讓資料集可以更多樣化，本研究選用了 CGAN 做為資料擴增。本研究在訓練資料集中挑選部分標籤等級較不模糊的圖片，且因為要人工進行圖片處理所以在張數挑選上只挑選了少數的圖片進行處理，分別為 A 類 73 張、B 類 70 張和 C 類 74 張，再以資料擴增去擴大圖片張數。

CGAN 的資料前處理流程圖如圖 15，圖片先將圖片裡的手人工移除並填滿背景，將芒果圖片轉為同一方向，接著將圖片大小等比例縮放到  $256 \times 256 \times 3$ ，不足的部分則補上白色，每張圖片再產生四張圖，藉此擴大資料集產生 CGAN 的訓練資料集。最終 CGAN 的訓練資料集包含 A 類為 292 張、B 類為 280 張和 C 類為 296 張，總計 868 張。

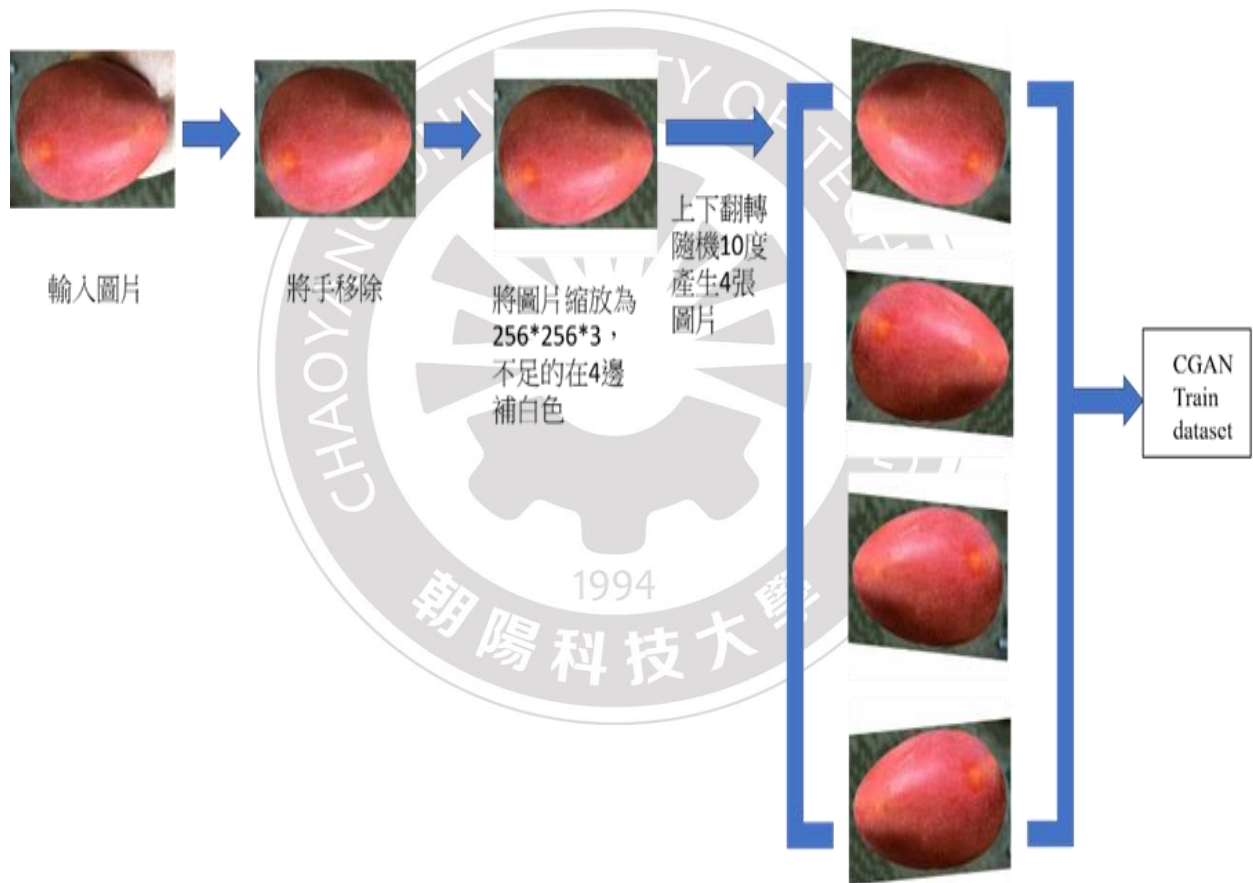


圖 15、CGAN 資料前處理流程示意圖

CGAN YOLOv4 的實驗流程如圖 16，將 CGAN Train Data 放入 CGAN 中訓練，利用訓練完的 CGAN 再生成 A 類 87 張、B 類 194 張和 C 類 150 張，結合原始資料集，可產生新的資料集整合後之 A 類有 1,879 張、B 類有 2,262 張和 C 類有 1,890 張的圖片放入 YOLOv4 訓練，總計擴增為 6,031 張。

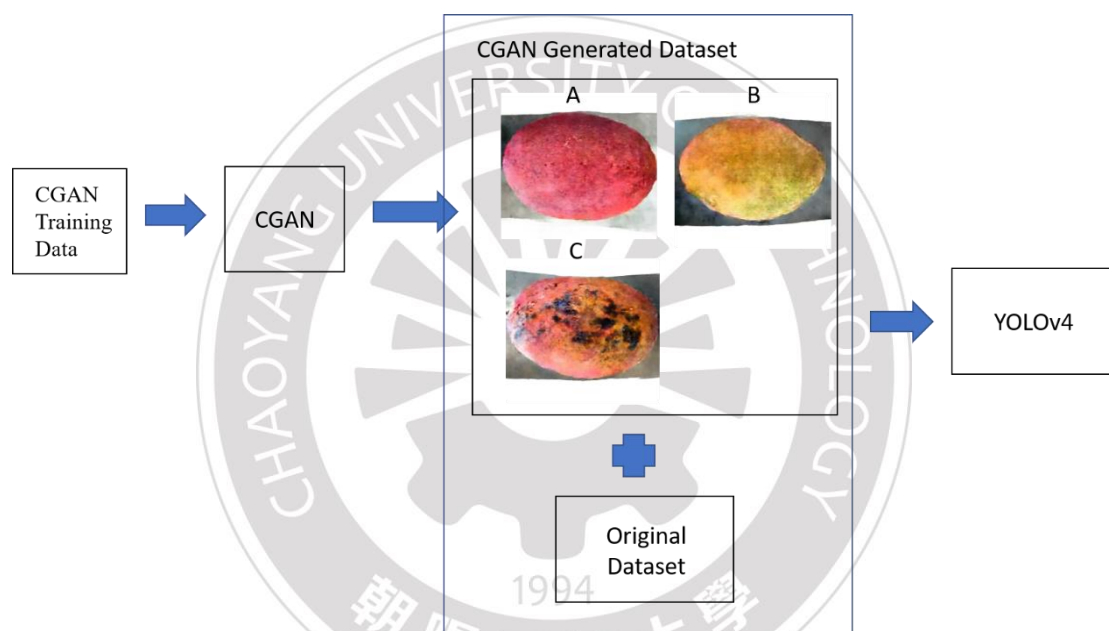


圖 16、CGAN YOLOv4 的實驗流程

本研究訓練時的設備環境如表 6，作業系統為 Windows，GPU 為 NVIDIA GeForce RTX2080 6GB，CPU 為 i7-9750H 和 32GB 的 DDR4 記憶體。

表 6、實驗設備環境

實驗設備環境	
OS	Windows
CPU	i7-9750H
GPU	NVIDIA GeForce RTX2080 6GB
Ram	DDR4 32GB

YOLOv3 SPP 和 YOLOv4 的模型訓練採用如表 4 說明的訓練和測試集張數。而 CGAN 則為 6,031 張。

訓練完成後，利用最後訓練權重，使用 Precision 準確率、Recall 召回率、F1-score、Mean Average Precision (mAP)、IoU，分別在針對 YOLOv3 SPP、YOLOv4 和 CGAN YOLOv4 模型進行驗證，測試的資料有 800 張的



芒果圖片，各個模型的訓練圖如圖 17、圖 18 和圖 19。模型建置並訓練後再進行測試資料集之驗證，測試結果如表 7、表 8 和表 9。

由下面圖 17 YOLOv3 SPP 的訓練圖可以觀察到，模型的訓練在 mAP 部份較不穩定。表 7 為 YOLOv3 SPP 驗證資料集的成果，其中測試資料集的 Precision 僅達到 56%，Recall 有較好的表現可達到 70%，而 F1-score 達到 62%、Avg IoU 效果並不佳只有 47.42%，mAP 尚可約為 74.56%。

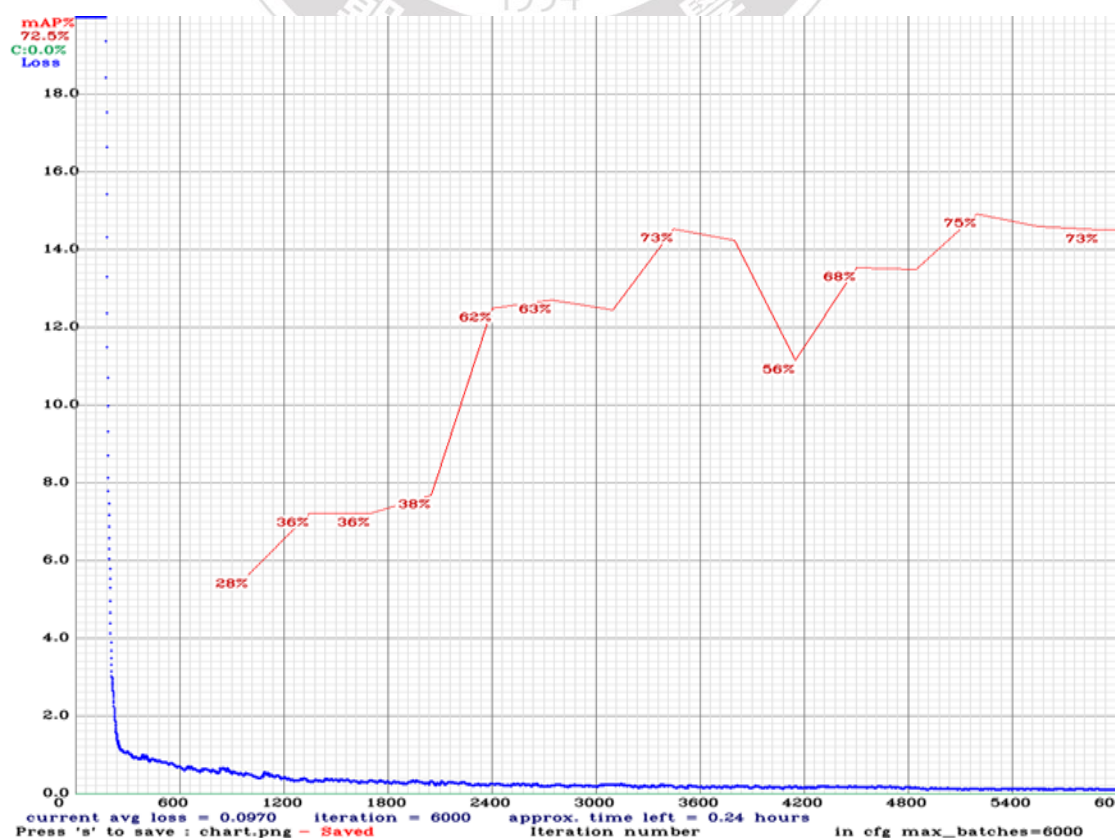


圖 17、YOLOv3 SPP 訓練圖

表 7、驗證 YOLOv3 SPP 在訓練集和測試集的結果

	Train	Test
TP	3,980	561
FP	2,969	441
FN	1620	239
Precision	57%	56%
Recall	71%	70%
F1-score	63%	62%
Avg IoU	49.05%	47.42%
mAP	<b>74.66%</b>	<b>74.56%</b>

下面圖 18 則為 YOLOv4 訓練圖，由於 YOLOv4 是 YOLOv3 SPP 的改進版，除了原有的 SPP 外加入了 PAN 可以更好的進行特徵融合，YOLOv4 也在模型中加入了 Mosaic 資料增強，所以與圖 17 相比 mAP 較為穩定也獲得比較高的 mAP 值達 89%。在表 8 所整理出 YOLOv4 在測試集的各項指標值為：Precision 達到 81%、Recall 達到 84%、F1-score 達到 82%、Avg IoU 達到 73.67% 和 mAP 達到 89.85%。

由上面兩個不同的 YOLO 模型所呈現的結果(參見表 7 和表 8)，可以觀察出 YOLOv4 在芒果圖片資料學習不論是在訓練或測試集上，都比 YOLOv3 SPP 的分級結果好。

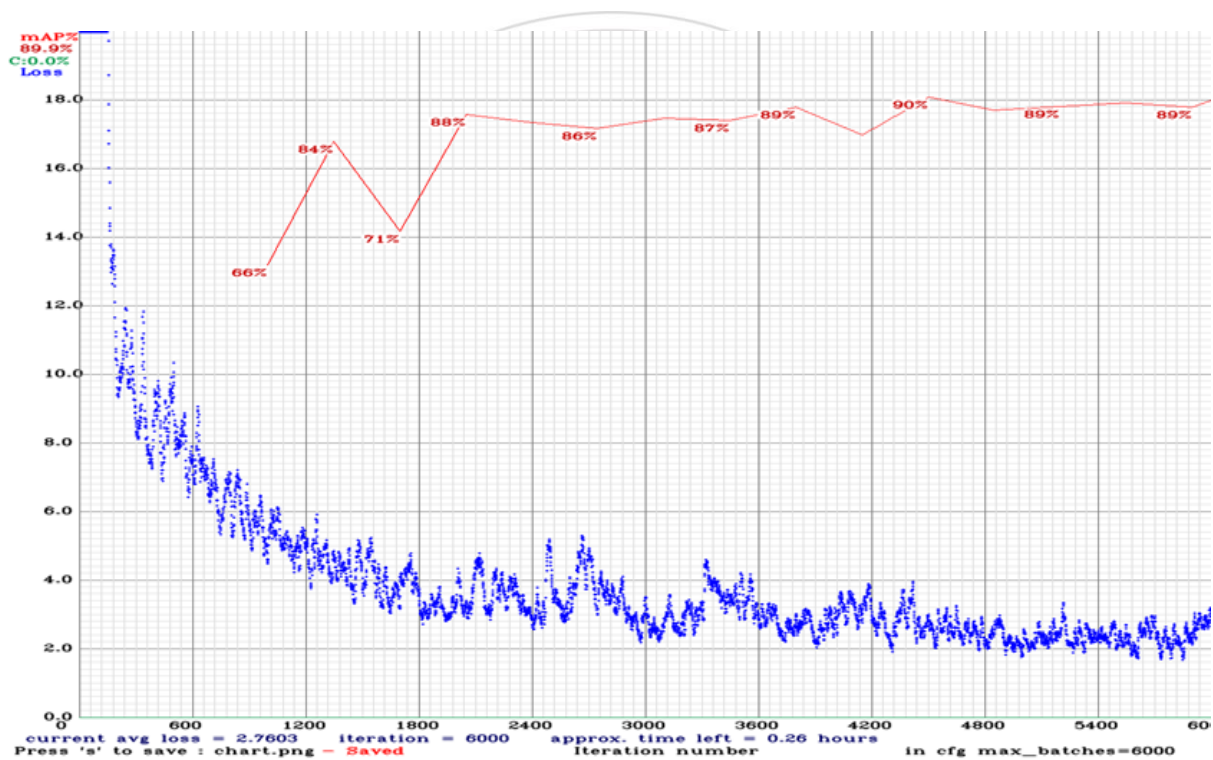


圖 18、YOLOv4 訓練圖

表 8、驗證 YOLOv4 在訓練集和測試集的結果

	Train	Test
TP	4,887	670
FP	926	161
FN	713	130
Precision	84%	81%
Recall	87%	84%
F1-score	86%	82%
Avg IoU	77.5%	73.67%
mAP	<b>93.15%</b>	<b>89.85%</b>

圖 19 為 CGAN 結合 YOLOv4 方法的模型訓練圖，跟圖 18 相比雖然一開始的 mAP 不高，但卻能穩定增加，而在訓練進入 2000 圈時 mAP 比 YOLOv4 高出了 1%，最後 mAP 甚至可以到達 90%。由下面表 9 可以看出，CGAN 結合 YOLOv4 之測試集，其 Precision 可以達到 82%、Recall 達

到 85%、F1-score 達到 83%、Avg IoU 達到 73.59% 和 mAP 達到 90.07%，由此亦可看出，CGAN 結合 YOLOv4 仍然可以再比 YOLOv4 些微的提升。根據上面數據顯示，加入了 CGAN 資料集之後，可以讓訓練資料更多樣化，也能學習到更多不同形樣的芒果，所以整體的成效可以比 YOLOv4 略好。

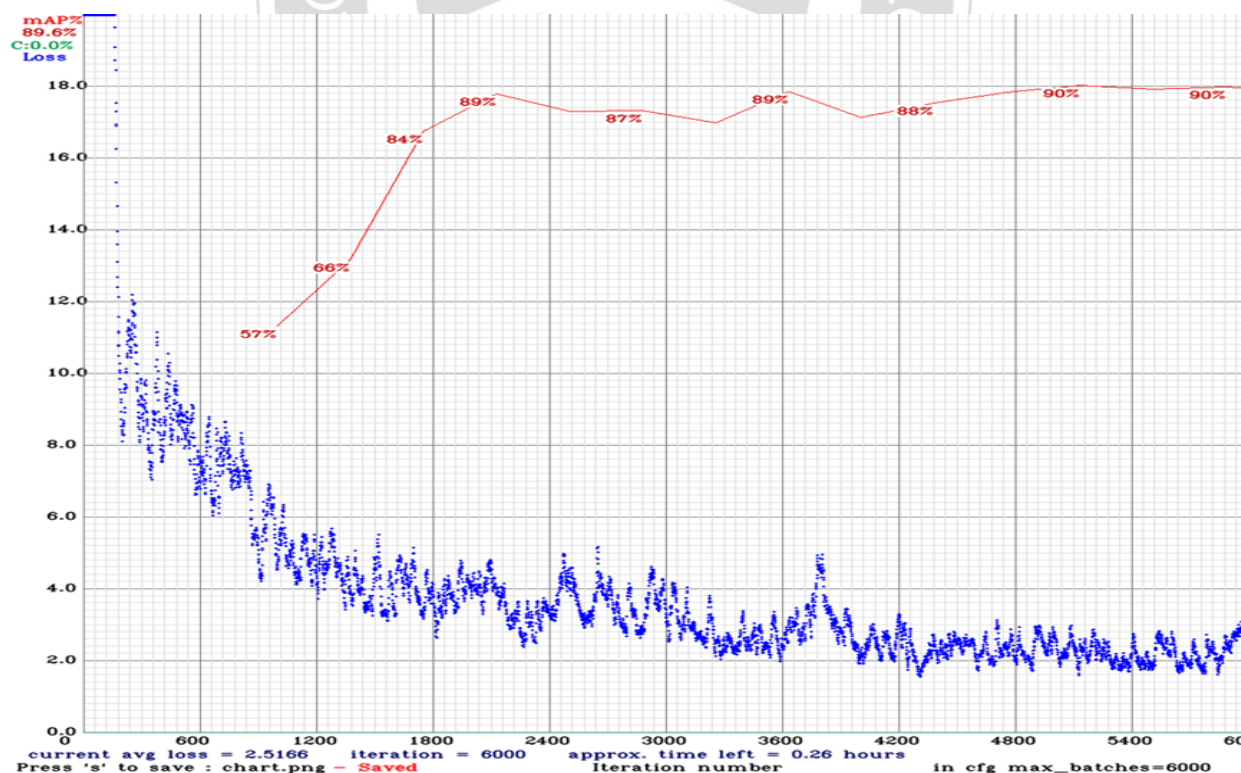


圖 19、CGAN YOLOv4 訓練圖

表 9、驗證 CGAN YOLOv4 在訓練集和測試集的結果

	Train	Test
TP	4,896	679
FP	844	149
FN	704	121
Precision	85%	82%
Recall	87%	85%
F1-score	86%	83%
Avg IoU	77.16%	73.59%
mAP	<b>93.58%</b>	<b>90.07%</b>

我們也將本研究的模型與 AICUP 比賽 baseline model，包括 ResNeXt、AlexNet、VGG16、DenseNet、ShuffleNet、Fusion model 和最佳模型做 Weighted Average Recall(WAR)的比較，如表 10。由表 10 可以看到本實驗的 YOLOv4 和 CGAN YOLOv4 模型比 baseline model 都還要高。與競賽最高模型相比在 CGAN YOLOv4 模型上提升了 1.2%。

表 10、競賽 baseline model、最佳模型和實驗模型的 WAR 比較

	model		WAR
AI CUP	Baseline	ResNeXt	56.75%
		AlexNet	69.63%
		VGG16	64.50%
		DenseNet	52.38%
		ShuffleNet	57.38%
		Fusion model	71.63%
	Rank	Best	<b>83.60%</b>
本研究	Ex1	YOLOv3 SPP	70.13%
	Ex2	YOLOv4	83.75%
	Ex3	<b>CGAN YOLOv4</b>	<b>84.88%</b>

圖 20 為使用 YOLOv3 SPP、YOLOv4 和 CGAN YOLOv4 預測結果圖，各等級標籤都選一張做為代表，分別為左圖 A 類、中間 B 類和右邊 C 類，藉由圖 20 可以比較模型在同一張圖片的真實圖片表現。在圖 20 中 YOLOv3 SPP 將 A 類誤認為 B 和 C 類沒有預測出來。YOLOv4 和 CGAN YOLOv4 在三張圖上預測都對，而 CGAN YOLOv4 的各類別信心度都較 YOLOv4 高一些，顯現由此範例可以看出 CGAN YOLOv4 的確在芒果辨識上，可以優於其他模型。



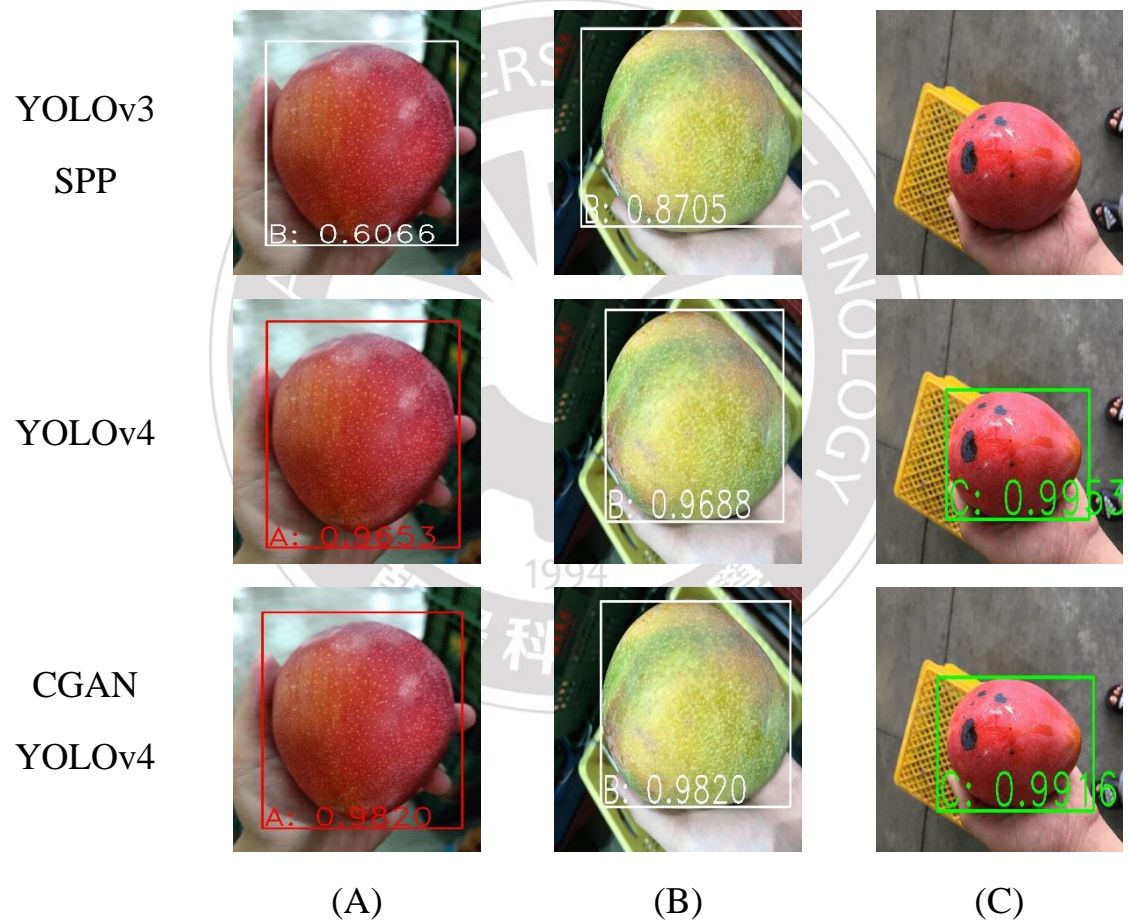


圖 20、YOLOv3 SPP、YOLOv4 和 CGAN YOLOv4 預測結果圖

由上述實驗可知 CGAN YOLOv4 的確有較好之分類成效，因此本研究在針對 CGAN YOLOv4 進行更細節之實驗研究。

### 4.3 CGAN YOLOv4 實驗與結果

本研究針對 CGAN YOLOv4 規劃更細節之改善方法實驗流程如圖 21。本研究提出三個模型，分別是 Model\_H、Model\_S 和 Model\_CE。上述三個模型都會將圖片使用相同的前處理方式，將圖片調整亮度，另用顏色制定一套規則預先篩選訓練的圖片。而 CGAN 的部分三個模型亦也相同，將原先 CGAN 生成的模型增加訓練圖片重新訓練。



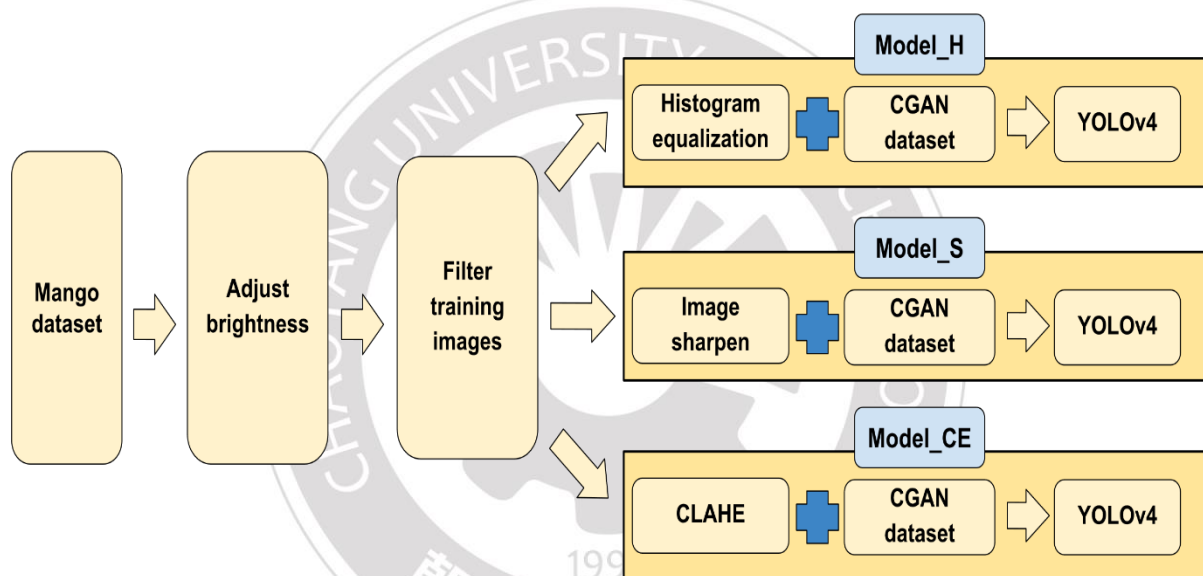


圖 21、CGAN YOLOv4 改進版流程圖

接著各個模型分別利用不同的方法來進行圖片增強進行，Model\_H 利用直方圖均衡、Model\_S 利用圖片銳化和 Model\_CE 利用 CLAHE 方法來進行圖片增強。最後上述三個模型分別結合 YOLOv4 進行訓練，最後與 CGAN YOLOv4 進行比較。

### 4.3.1 實驗方法

#### 4.3.1.1 CGAN 圖片前處理流程

CGAN 前處理流程架構圖如圖 15，將圖片使用上下左右顛倒隨機旋轉 10 度範圍，每 1 張圖片再產生經旋轉的 8 張圖片，擴增 CGAN 訓練資料集，與原先實驗相比再增加了一倍的張數，即 1,736 張。

#### 4.3.1.2 調整亮度

將圖片經由整張圖片調整亮度後，產生的結果如圖 22。由圖 22 我們可以觀察到芒果右上方較暗部分有明顯的改善，但缺點是原本較亮部分也會變得更亮。



(a)原圖

(b)亮度調整後

圖 22、圖片調整亮度

#### 4.3.1.3 篩選圖片

資料集中 A 類以及 B 類圖片先轉換到 HSV 色彩空間後，根據色調以及飽和度篩選。C 類則是根據缺陷大小以及缺陷數量進行篩選。

#### 4.3.1.4 直方圖均衡、銳化和 CLAHE 方法

將 4.3.1.2 說明中調整亮度後的圖片，使用直方圖進行均衡、銳化和 CLAHE 等方法來加強影像品質，其結果圖如圖 23，可以看到相較於(a)原圖，(c)銳化和(d)CLAHE 兩種方法有顯著的改善。

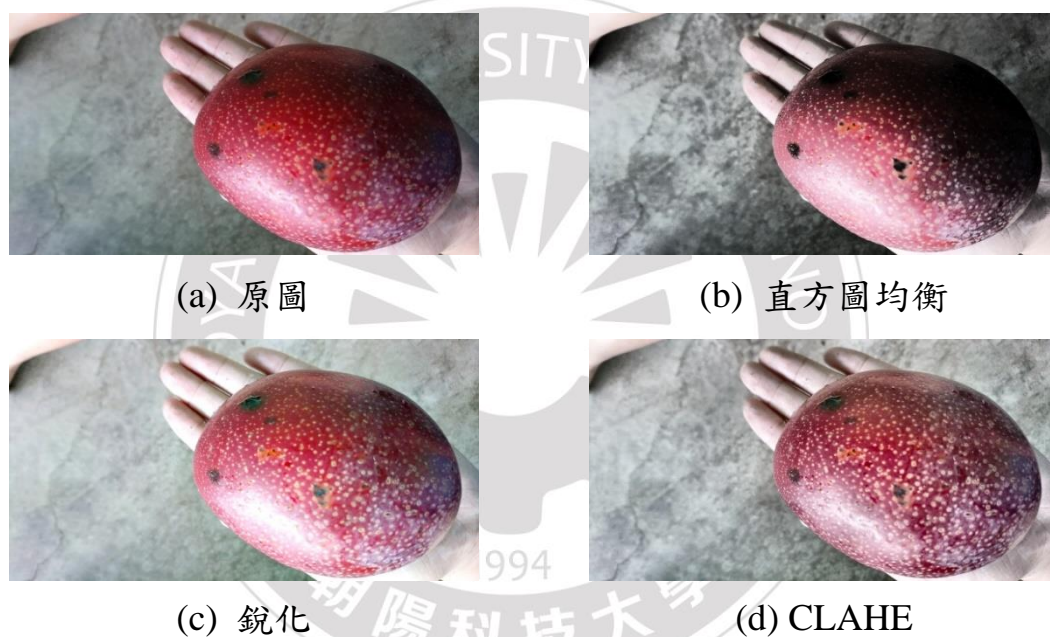


圖 23、圖片經三種不同方式進行調整的結果圖

#### 4.3.2 實驗結果

利用 3.4 小節所提到的驗證方式，即 TP、FP、FN、Precision、Recall、F1-Score、Avg IoU 和 mAP 等，個別驗證所實驗的三個模型：即 Model\_H、Model\_S 以及 Model\_CE。模型建置並訓練後再進行訓練以及測試資料集之驗證並與 CGAN YOLOv4 模型進行比較，測試結果如表 11 和表 12。

表 11、不同 CGAN YOLOv4 模型在訓練集的結果

	Train			
	CGAN YOLOv4	Model_H	<b>Model_S</b>	Model_CE
TP	4,896	5,352	5,390	5,403
FP	844	870	822	867
FN	704	688	650	637
Precision	85%	86%	87%	86%
Recall	87%	89%	89%	89%
F1-score	86%	87%	<b>88%</b>	88%
Avg IoU	77.16%	78.55%	<b>80.01%</b>	77.11%
mAP	<b>93.58%</b>	<b>95.09%</b>	<b>95.53%</b>	<b>95.41%</b>

表 11 是模型的訓練結果，我們可以觀察到表 11 中 Model\_S 模型在訓練集的所有評估指標結果是最好的。與 CGAN YOLOv4 相比，在 Precision、Recall、F1-score、Avg IoU 以及 mAP 都提升 2%。

表 12、不同 CGAN YOLOv4 模型在測試集的結果

	Test			
	CGAN YOLOv4	Model_H	Model_S	Model_CE
TP	679	658	636	649
FP	149	183	185	197
FN	121	142	164	151
Precision	82%	78%	77%	77%
Recall	85%	82%	80%	81%
F1-score	83%	80%	78%	79%
Avg IoU	73.59%	70.96%	70.83%	68.63%
mAP	<b>90.07%</b>	87.87%	87.72%	87.99%

不過，接下來本研究針對三個改善模型進行測試集的實驗，如表 12，但發現所實驗的三個模型在測試集的所有評估指標與 CGAN YOLOv4 相比都比較差，故 CGAN YOLOv4 是最後的最佳模型。

根據上述表 11 以及表 12 的模型表現，運用影像改善以及影像增強在訓練集的表現上是有所提升的，但卻在測試集成效較差，我們認為主要的原因有兩點，分別是圖片增強方法和資料集。

首先是運用的圖片增強方法，如圖 23 所示，圖 23 將細節的部分顯示得更強但卻因此缺失了部分色彩，導致當沒有缺陷時需依顏色這個情形會辨識錯誤，這點可以在未來研究中找尋更適合的影像增強方法進行改善，並在改善的同時須讓影像細節和色彩取得平衡。

第二點是資料集，資料集中有很多資料標籤過於模糊，在不同等級的芒果都出現相似的圖片，並且有等級標示錯誤的圖片的問題，所以導致改良後的模型成效仍較差。

故經過本研究之基於考量影像處理的時間、速度以及準確率等綜合考量集與模型比較後，建議仍以 CGAN YOLOv4 為最佳模型。



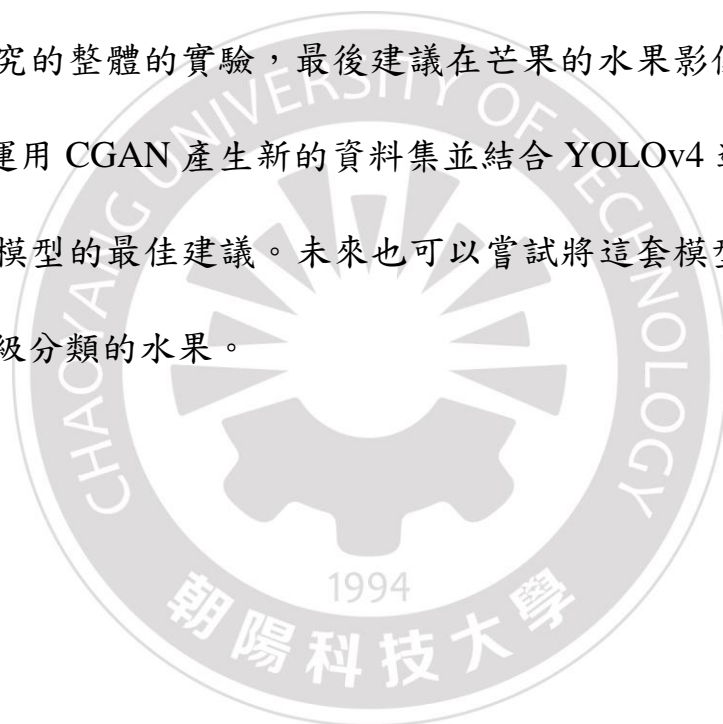
## 第伍章 結論

本研究藉由台灣 AI CUP 競賽[24]所提供的愛文芒果資料集建立一套影像辨識自動篩選芒果系統，使用訓練集 5,600 張芒果圖片，測試集 800 張芒果圖片，分別實驗 AlexNet、YOLOv3 SPP、YOLOv4 和 CGAN YOLOv4 模型，其中 CGAN YOLOv4 模型會再額外加上生成的資料集。根據上表 7、表 8 和表 9 的各項驗證方式可以發現 CGAN YOLOv4 的表現平均最佳，在訓練集可獲得 85% 的 Precision、85% 的 Recall、83% 的 F1-score、73.59% 的 Avg IoU 和 90.07% 的 mAP。而在與 AI CUP 比賽的 baseline 模型和最佳模型的 WAR 比較也提高了 1.2%。

在本研究針對 CGAN YOLOv4 更進一步的實驗設計：Model\_H、Model\_S 和 Model\_CE 三個模型進行圖像改善下，再與 CGAN YOLOv4 進行比較，所產生的訓練集表現如表 11，測試集的表現如表 12，與其他 YOLOv3 SPP 和 YOLOv4 模型相比下是比較好的，但若與 CGAN YOLOv4 比較，則在訓練集上 Model\_S 為最佳，但在測試集來看，則 CGAN YOLOv4 仍為最佳。

故經過本研究之綜合考量及與模型比較後，建議仍以 CGAN YOLOv4 為最佳模型，基於考量影像處理的時間、速度以及準確率之上。不過本研究的未來研究建議仍可以朝影像增強、影像改善還有資料集的準備上面去強化，本研究相信在資料集的改善也會同時提昇模型本身的效能。

經過本研究的整體的實驗，最後建議在芒果的水果影像辨識以及等級分類上，能夠運用 CGAN 產生新的資料集並結合 YOLOv4 進行影像辨識這會是一個目前模型的最佳建議。未來也可以嘗試將這套模型運用在不同種類且需進行等級分類的水果。



## 參考文獻

- [1] 台灣行政院農委會, “農業統計資料網,” 2021.  
<https://agrstat.coa.gov.tw/sdweb/public/trade/TradeCoa.aspx>.
- [2] M. A. Momin, M. T. Rahman, M. S. Sultana, C. Igathinathane, A. T. M Ziauddin, and T. E. Grift(2017), “Geometry-based mass grading of mango fruits using image processing,” *Information Processing in Agriculture*, vol. 4, no. 2, pp. 150–160, doi: 10.1016/j.inpa.2017.03.003.
- [3] A. M. Vyas, B. Talati, and S. Naik(2014), “Quality inspection and classification of mangoes using color and size features,” *International Journal of Computer Applications*, vol. 98, no. 1, pp. 1–5.
- [4] D. Sahu and R. M. Potdar(2017), “Defect identification and maturity detection of mango fruits using image analysis,” *American Journal of Artificial Intelligence*, vol. 1, no. 1, pp. 5–14.
- [5] Y. Li, X. Zheng, and X. Wang(2008), "Fruit discrimination on region feature," *Proceedings of the 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, Jinan, China, pp. 590-594, doi: 10.1109/FSKD.2008.5.
- [6] R. Pydipati, T. F. Burks, and W. S. Lee(2006), “Identification of citrus disease using color texture features and discriminant analysis,” *Computers and Electronics in Agriculture*, vol. 52, no. 1–2, pp. 49–59, doi: 10.1016/j.compag.2006.01.004.
- [7] A. Kumar and G. S. Gill(2015), “Computer vision based model for fruit sorting using K-nearest neighbour classifier,” *International Journal of Electrical and Electronics Engineers*, vol. 2, no. 1, pp. 49–52.

- [8] W. D. N. Pacheco and F. R. J. López(2019), "Tomato classification according to organoleptic maturity (coloration) using machine learning algorithms K-NN, MLP, and K-Means Clustering," *Proceedings of the 2019 XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA)*, Bucaramanga, Colombia, pp. 1-5, doi: 10.1109/STSIVA.2019.8730232.
- [9] D. Pise and G. D. Upadhye(2018), "Grading of harvested mangoes quality and maturity based on Machine Learning Techniques," *Proceedings of the 2018 International Conference on Smart City and Emerging Technology (ICSCET)*, Mumbai, India, pp. 1-6, doi: 10.1109/ICSCET.2018.8537342.
- [10] B. J. Samajpati and S. D. Degadwala(2016), "Hybrid approach for apple fruit diseases detection and classification using random forest classifier," *Proceedings of the 2016 International Conference on Communication and Signal Processing (ICCSP)*, Melmaruvathur, India, pp. 1015-1019, doi: 10.1109/ICCSP.2016.7754302.
- [11] S. Fan, J. Li, Y. Zhang, X. Tian, Q. Wang, X. He, C. Zhang, and W. Huang(2020), "On line detection of defective apples using computer vision system combined with deep learning methods," *Journal of Food Engineering*, vol. 286, doi: 10.1016/j.jfoodeng.2020.110102.
- [12] Suganya, Vinodha, Thilagavathi, and Pavithra(2019), "A fruit quality inspection sytem using Faster Region Convolutional," *International Research Journal of Engineering and Technology (IRJET)*, vol. 6, no. 3, pp. 6717–6720.

- [13] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao(2020), “YOLOv4: optimal speed and accuracy of object detection,” pp. 1-17, [Online]. Available: <http://arxiv.org/abs/2004.10934>.
- [14] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio(2014), “Generative adversarial nets,” *Proceedings of the 27th International Conference on Neural Information Processing Systems(NIPS)* , Montreal, Canada, vol. 2, pp. 2672–2680.
- [15] A. Radford, L. Metz, and S. Chintala(2016), “Unsupervised representation learning with deep convolutional generative adversarial networks,” *Proceedings of the 4th International Conference on learning representations (ICLR)*, Caribe Hilton, San Juan, Puerto Rico, 2016, pp. 1–16.
- [16] M. Arjovsky, S. Chintala, and L. Bottou(2017), “Wasserstein Generative Adversarial Networks,” *Proceedings of the 34th International Conference on Machine Learning (ICML)*, Sydney, Australia. pp. 214-223.
- [17] M. Mirza and S. Osindero(2014), “Conditional Generative Adversarial Nets,” pp. 1–7, [Online]. Available: <http://arxiv.org/abs/1411.1784>.
- [18] S. L. Wu, H. Y. Tung, and Y. L. Hsu(2020), “Deep learning for automatic quality grading of mangoes: methods and insights,” *Proceedings of the 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Miami, FL, USA, pp. 446–453, doi: 10.1109/ICMLA51294.2020.00076.
- [19] X. Liming, and Z. Yanchao(2010), “Automated strawberry grading system based on image processing,” *Computers and Electronics in Agriculture*, vol. 71, pp. 32–39, doi: 10.1016/j.compag.2009.09.013.



- [20] M. P. Arakeri and Lakshmana(2016), “Computer vision based fruit grading system for quality evaluation of tomato in agriculture industry,” *Procedia Computer Science*, vol. 79, pp. 426–433, doi: 10.1016/j.procs.2016.03.055.
- [21] M. Momeny, A. Jahanbakhshi, K. Jafarnezhad, and Y. D. Zhang(2020), “Accurate classification of cherry fruit using deep CNN based on hybrid pooling approach,” *Postharvest Biology and Technology*, vol. 166, doi: 10.1016/j.postharvbio.2020.111204.
- [22] S. S. Deulkar and S. S. Barve(2018), “Feature based fruit quality grading system using Support Vector Machine,” *Proceedings of the 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, Bangalore, India, pp. 2367–2372.
- [23] S. Ren, K. He, R. Girshick, and J. Sun(2017), “Faster R-CNN: towards real-time object detection with Region Proposal Networks,” *The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.
- [24] AI CUP, “Mango grades classification competition,” 2020. <https://aidea-web.tw/topic/72f6ea6a-9300-445a-bedc-9e9f27d91b1c>.
- [25] K. He, G. Gkioxari, P. Dollar, and R. Girshick(2017), “Mask R-CNN,” *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.
- [26] A. A. Shvets, V. I. Iglovikov, A. Rakhlin, and A. A. Kalinin(2018), “Angiodysplasia detection and localization using deep Convolutional Neural Networks,” *Proceedings of the 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Orlando, FL, USA, pp. 612-617, doi: 10.1109/ICMLA.2018.00098.

- [27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi(2016), “You only look once: unified, real-time object detection,” *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 779–788, doi: 10.1109/CVPR.2016.91.
- [28] J. Redmon and A. Farhadi(2017), “YOLO9000: better, faster, stronger,” *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 6517–6525, doi: 10.1109/CVPR.2017.690.
- [29] J. Redmon and A. Farhadi(2018), “YOLOv3: an incremental improvement,” pp 1-6, [Online]. Available: <http://arxiv.org/abs/1804.02767>.
- [30] R. Siddiqi(2019), “Automated apple defect detection using state-of-the-art object detection techniques,” *SN Applied Sciences*, vol. 1, no. 11, doi: 10.1007/s42452-019-1393-4.
- [31] J. Zhao and J. Qu(2019), "A detection method for tomato fruit common physiological diseases based on YOLOv2," *Proceedings of the 2019 10th International Conference on Information Technology in Medicine and Education (ITME)*, Qingdao, China, pp. 559-563, doi: 10.1109/ITME.2019.00132.
- [32] W. Yijing, Y. Yi, W. Xue-fen, C. Jian, and L. Xinyun(2021), "Fig fruit recognition method based on YOLO v4 deep learning," *Proceedings of the 2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Chiang Mai, Thailand. pp. 303-306, doi: 10.1109/ECTI-CON51831.2021.9454904.

- [33] Z. Li, M. Dong, S. Wen, X. Hu, P. Zhou, and Z. Zeng(2019), “CLU-CNNs: object detection for medical images,” *Neurocomputing*, vol. 350, pp. 53–59, doi: <https://doi.org/10.1016/j.neucom.2019.04.028>.
- [34] C. Dewi, R. C. Chen, Y. T. Liu, Y. S. Liu, and L. Q. Jiang(2020), “Taiwan stop sign recognition with customize anchor,” *Proceedings of the 12th International Conference on Computer Modeling and Simulation(ICCMS)*, pp. 51–55, Brisbane, QLD, Australia, doi: <https://doi.org/10.1145/3408066.3408078>.
- [35] P. M. Bhatt, R. K. Malhan, P. Rajendran, B. C. Shah, S. Thakar, Y. J. Yoon, and S. K. Gupta(2021), “Image-based surface defect detection using deep learning: a review,” *Journal of Computing and Information Science in Engineering*, vol. 21, no. 4, doi: <https://doi.org/10.1115/1.4049535>.
- [36] H. Kuang, C. Liu, L. L. H. Chan, and H. Yan(2018), “Multi-class fruit detection based on image region selection and improved object proposals,” *Neurocomputing*, vol. 283, pp. 241–255, doi: <https://doi.org/10.1016/j.neucom.2017.12.057>.
- [37] X. Zhou, W. S. Lee, Y. Ampatzidis, Y. Chen, N. Peres, and C. Fraisse(2021), “Strawberry maturity classification from UAV and near-ground imaging using deep learning,” *Smart Agricultural Technology*, vol. 1, doi: <https://doi.org/10.1016/j.atech.2021.100001>.
- [38] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg(2016), “SSD: Single shot multibox detector,” *Proceedings of the 2016 European Conference on Computer Vision(ECCV)*, Amsterdam, The Netherlands, vol. 9905, pp. 21–37, doi: [10.1007/978-3-319-46448-0\\_2](https://doi.org/10.1007/978-3-319-46448-0_2).

- [39] S. Yun, D. Han, S. Chun, S. J. Oh, J. Choe, and Y. Yoo(2019), "CutMix: Regularization strategy to train strong classifiers with localizable features," *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), pp. 6022-6031, doi: 10.1109/ICCV.2019.00612.
- [40] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz(2018), "MixUp: Beyond empirical risk minimization," *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada, pp. 1-13.
- [41] J. J. Bird, C. M. Barnes, L. J. Manso, A. Ekárt, and D. R. Faria(2021), "Fruit quality and defect image classification with Conditional GAN data augmentation," [Online]. Available: <http://arxiv.org/abs/2104.05647>.
- [42] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger and H. Greenspan(2018), "Synthetic data augmentation using GAN for improved liver lesion classification," *Proceedings of the 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI)*, Washington, DC, USA, pp. 289-293, doi: 10.1109/ISBI.2018.8363576.
- [43] G. Saravanan, G. Yamuna, and S. Nandhini(2016), "Real time implementation of RGB to HSV/HSI/HSL and its reverse color space models," *Proceedings of the 2016 International Conference on Communication and Signal Processing(ICCSP)*, Melmaruvathur, India, pp. 462-466, doi: 10.1109/ICCSP.2016.7754179.
- [44] K. Zuiderveld(1994), "Contrast limited adaptive histogram equalization," *Graphics Gems IV*, pp. 474-485.

- [45] T. R. Razak, O. Mahmood, A. M. N. A Bakar, K. A. b. Ahmad, and R. Mansor(2012), "Mango grading by using fuzzy image analysis," *Proceedings of the International Conference on Agricultural, Environment and Biological Sciences (ICAEBs'2012)*, Phuket, Thailand, pp. 18–22.
- [46] R. Sriram, A. T. M, and J. Girija(2018), "Mango classification using Convolutional Neural Networks," *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 11, pp. 1729–1731.
- [47] A. Krizhevsky, I. Sutskever, and G. E. Hinton(2017), "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, doi: <https://doi.org/10.1145/3065386>.
- [48] T. -Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie(2017), "Feature Pyramid Networks for object detection," *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 936-944, doi: 10.1109/CVPR.2017.106.
- [49] J. Jiang, X. Fu, R. Qin, X. Wang, and Z. Ma(2021), "High-speed lightweight ship detection algorithm based on YOLO-V4 for three-channels RGB SAR image," *Remote Sensing*, vol. 13, no. 10, doi: 10.3390/rs13101909.
- [50] C. Y. Wang, H. Y. M. Liao, Y. H. Wu, P. Y. Chen, J. W. Hsieh, and I. H. Yeh(2020), "CSPNet: A new backbone that can enhance learning capability of CNN," *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Seattle, WA, USA, pp. 1571–1580, doi: 10.1109/CVPRW50498.2020.00203.



- [51] D. Misra and L. K. Bhubaneswar(2020), “Mish: a self regularized non-monotonic activation function,” *Proceedings of the 31st British Machine Vision Virtual Conference(BMVC)*.
- [52] K. He, X. Zhang, S. Ren, and J. Sun(2014), "Spatial pyramid pooling in deep convolutional networks for visual recognition", *Computer VisionCECCV 2014*, pp. 346-361.
- [53] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia(2018), “Path Aggregation Network for instance segmentation,” *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 8759–8768, doi: 10.1109/CVPR.2018.00913.
- [54] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese(2019), “Generalized intersection over union: a metric and a loss for bounding box regression,” *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 658–666, doi: 10.1109/CVPR.2019.00075.
- [55] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren(2020), “Distance-IoU Loss: faster and better learning for bounding box regression,” *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)*, vol. 34, no. 7, pp. 12993–13000, doi: 10.1609/aaai.v34i07.6999.
- [56] Tzutalin, “LabelImg,” 2015. <https://github.com/tzutalin/labelImg>.
- [57] C. Dewi, R. C. Chen, Y. T. Liu, X. Jiang, and K. D. Hartomo(2021), "Yolo V4 for advanced traffic sign recognition with synthetic training data generated by Various GAN," *IEEE Access*, vol. 9, pp. 97228-97242, doi: 10.1109/ACCESS.2021.3094201.
- [58] AlexeyAB, “darknet.” <https://github.com/AlexeyAB/darknet>.