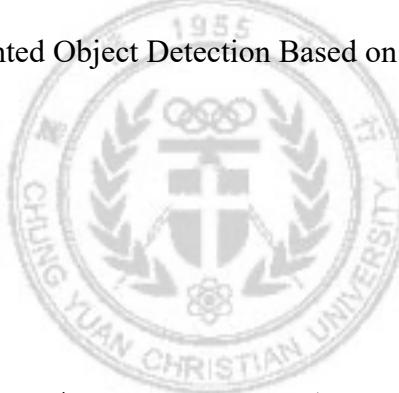


中原大學  
資訊工程學系  
碩士學位論文

基於 YOLOv7 之旋轉物件辨識方法

An Oriented Object Detection Based on YOLOv7



指導教授：蘇志文  
研究生：林冠良

林冠良

中華民國  
112年6月

中華民國 112 年 6 月

## 摘要

近年來，物件偵測技術在空拍影像應用中扮演了越來越重要的角色。然而，現有的物件偵測模型通常只能偵測物件的位置和類別，而無法辨識物件的角度。為了解決這個問題，我們對 YOLOv7 模型進行了修改，以實現對物件角度的準確偵測。這些修改包括增加了一個旋轉檢測層，用於檢測物件的旋轉角度，並在損失函數中加入了旋轉角度的損失項，此外也在交聯比損失項加入更能幫助角度損失項收斂的權重係數，使得模型能更準確地學習物體角度資訊，以確保偵測出的物件邊界框能夠更好地貼合物件。

為了驗證修改後的 YOLOv7 模型的效果，我們使用了大型遙測空拍數據集 DOTA1.0 資料集進行了實驗。實驗結果表明，修改後的 YOLOv7 模型在資料集上取得了不錯的成果，讓基於迴歸方式的旋轉物件偵測方法多了另一種可行性，同時也驗證了我們的修改可以有效地提高旋轉物件偵測模型的準確性，並有望未來能在實際應用中發揮更大的作用。

關鍵字：單階段物體偵測方法、角度偵測、遙測空拍影像處理

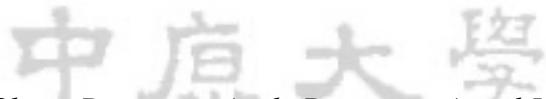


中原大學

## Abstract

In recent years, object detection technology has played an increasingly important role in aerial image applications. However, most existing object detection models can only detect the position and category of objects, but not the orientation of objects. To address this issue, we modified the YOLOv7 model to achieve accurate detection of object orientation. These modifications included adding a rotation detection layer to detect the rotation angle of objects, and adding a loss term for the rotation angle in the loss function to further improve the model's accuracy, furthermore, we also add an angle factor to the IoU Loss to help the model learn the object angle information more precisely.

To verify the effectiveness of the modified YOLOv7 model, we conducted experiments using DOTA1.0, the well-known aerial image dataset. The experimental results showed that our modified YOLOv7 model performed really well on oriented object detection, and also give more possibilities to regression-based oriented object detection model. This indicates that our modifications can effectively improve the accuracy of oriented object detection models and have the potential to have a greater impact in future practical applications.



**Keywords:** *Single-Stage Object Detection, Angle Detection, Aerial Image Processing*

## 致謝

碩班的這兩年，一路上受到了許多人的幫助與鼓勵，也因為有這些助力，我才能順利地畢業。首先感謝我最重要的指導教授蘇志文教授，這兩年來在實驗上提供了我非常完善的實驗環境與設備，在研究上也持續提點跟教導我，在論文撰寫上也用心地幫我校正，讓我在碩班的路上一步步向前，扎扎实實的學習，最後得以完成我的研究。在此致上最誠摯的感謝，謝謝老師。

另外也感謝 Lab602 的各位，謝謝泰弘學長，指引了剛開始修改程式碼一頭霧水的我正確的方向，也傳授了我非常多寶貴的經驗，給了我莫大的幫助。謝謝峻瑋在碩班兩年中與我互相扶持，一同在研究上給予新的方向與想法。謝謝心平與松軒，在我對研究缺乏動力的時候，給我鼓勵與打氣。謝謝承宥、彥綸、婉菁給予我在研究期間的各種幫助，你們是最棒的學弟妹。

最後謝謝身邊的家人與朋友，謝謝家人在經濟跟精神上給我的鼓勵與支持，讓我能更專注於學習研究。謝謝女朋友在我對自己的研究沒有自信的時候，給了我動力跟勇氣繼續堅持下去。另外也感謝在碩班生涯中，幫助過鼓勵過我的所有人。

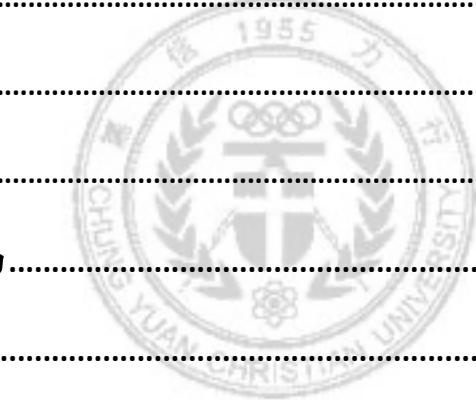
沒有以上的各位就沒有成功畢業的我，謝謝你們，我銘記在心。

中原大學

# 目次

摘要 .....	I
Abstract.....	II
致謝 .....	III
目次 .....	IV
圖目次 .....	VI
表目次 .....	VI
第一章 序論 .....	1
1.1 研究動機.....	1
1.2 論文架構.....	2
第二章 相關文獻 .....	3
2.1 物件偵測.....	3
2.1.1 二階段 (Two-stage) 物件偵測器 .....	3
2.1.2 單階段 (One-stage) 物件偵測器 .....	5
2.2 旋轉物件偵測 .....	6
2.2.1 旋轉物件角度迴歸 .....	6
2.2.2 角度損失計算 .....	7
第三章 研究方法 .....	9
3.1 物體偵測.....	9
3.1.1 新的重參數化方法 (Re-parameterized model) .....	9
3.1.2 新的動態標籤分配策略 (Dynamic label assignment strategy) .....	10
3.1.3 模型縮放與擴展 (Compound model scaling) .....	11
3.1.4 更快的速度與更高的準確率 .....	11

<b>3.2 角度偵測.....</b>	<b>13</b>
3.2.1 旋轉邊界框之角度定義.....	13
3.2.2 角度偵測頭.....	14
<b>3.3 數據增強.....</b>	<b>16</b>
3.3.1 馬賽克（Mosaic Data Augmentation） .....	16
3.3.2 翻轉（Flip up-down, Flip left-right Data Augmentation） .....	17
<b>3.4 基於 YOLOv7 之空拍單階段旋轉邊界框物體偵測器 .....</b>	<b>18</b>
<b>第四章 實驗結果與分析.....</b>	<b>21</b>
4.1 實驗環境.....	21
4.2 實驗資料.....	21
4.3 實驗結果.....	23
<b>第五章 結論與未來方向.....</b>	<b>28</b>
<b>參考文獻 .....</b>	<b>29</b>



中原大學

## 圖目次

圖 2- 1 R-CNN[6]模型流程圖 .....	4
圖 2- 2 YOLO[9]流程圖 .....	5
圖 2- 3 RoI Transformer[5]提出之 RRoI Learner .....	6
圖 2- 4 GWD[13]中的高斯分佈轉換 .....	7
圖 2- 5 GWD[13]中提及之旋轉物體角度預測的三大問題 .....	8
圖 3- 1 本方法訓練及測試階段流程示意圖 .....	9
圖 3- 2 YOLOv7[1]說明 Identity 連結破壞殘差與串連架構 .....	10
圖 3- 3 YOLOv7[1]主導頭與輔助頭之動態標籤分配示意圖 .....	11
圖 3- 4 YOLOv7[1]所提出之模型模組縮放解決方法 .....	12
圖 3- 5 Computational block 縮放後導致 Transition layer 輸入通道寬度縮放示意圖 .....	12
圖 3- 6 YOLOv7[1]與其他 YOLO 系列之速度與精度比較 .....	12
圖 3- 7 不同物件角度之定義與本論文對角度之定義 .....	13
圖 3- 8 YOLOv7 提出之 ELAN[18]架構 .....	14
圖 3- 9 YOLOv7 偵測頭以及我們所加入之角度偵測頭 .....	15
圖 3- 10 邊界框預測資訊 .....	15
圖 3- 11 馬賽克數據增強示意圖 .....	16
圖 3- 12 翻轉資料增強示意圖，(a)為水平翻轉，(b)為垂直翻轉 .....	17
圖 3- 13 本論文所使用之 YOLOv7 網路架構圖 .....	18
圖 3- 14 純絕對角度差與最小角度差示意圖 .....	19
圖 4- 1 DOTA-1.0[2]標註檔可視化影像 .....	21
圖 4- 2 DOTA-v1.0 所有類別之範例圖 .....	22
圖 4- 3 使用基礎參數訓練的預測結果圖 .....	23
圖 4- 4 初始參數（左）與加入正切倒數加權係數（右）模型表現比較（1） .....	25
圖 4- 5 初始參數（左）與加入正切倒數加權係數（右）模型表現比較（2） .....	26
圖 4- 6 偵測錯誤之結果 .....	27

## 表目次

表 3-1 交聯比與角度損失權重組合表 .....	20
表 4-1 DOTA 資料集分割前後類別數量比較表 .....	22
表 4-2 初始參數與角度加權係數模型表現比較 .....	24
表 4-3 不同交聯比計算方法模型表現比較 .....	27
表 4-4 各類別之平均精度與平均精度均值 .....	27



中原大學

# 第一章 序論

## 1.1 研究動機

隨著無人機的普及，遙測航拍的影像越來越易於取得。在拍攝成本日趨低下的發展下，人們也開始將物件辨識的技術大量應用於遙測航拍影像的辨識上，例如郵輪航線分析，十字路口車流量分析等等。但由於航拍影像與一般影像的拍攝環境、視野、規格有著很大差異，因此如何保留航拍影像的優勢與特點，成為一個新的重要議題。在過往的經驗中，當使用以物體上、下、左、右為邊界的水平邊界框去訓練旋轉物件偵測模型時，對於密集排列與長寬比變化大的物體辨識上，容易因與其他物體或大面積背景的特徵混淆而難以準確定位。甚至因為損失函數規劃上的不周全，導致預測角度與真實角度相差 90 度或者物體長短邊互換的問題。由於前述的這些短處，使得大部分的空拍影像分析研究，轉而著重於旋轉擬合框為基礎的旋轉物件偵測器模型。雖然說在角度上比起水平邊界框訓練模型準確不少，但同樣也有其模型自身的漏檢的缺點。在本篇論文中，因受到 YOLOv7[1]強大的性能啟發，希望能以 YOLOv7 為基礎，在原本成熟的模型上，以旋轉擬合框做訓練資料，並加入物件角度偵測，去研究如何在模型中加入額外的計算因子，進而優化空拍影像物件的角度偵測任務。

在本研究中，我們對 YOLOv7 模型進行了修改，額外增加了一個旋轉偵測層，用於偵測物件的旋轉角度，以實現對物件角度的預測。並嘗試將角度資訊與邊界框資訊做整合，為交聯比損失項添加角度損失資訊，使得模型能更加容易在初期階段進行收斂，使模型迴歸出正確的預測角度。

為了驗證我們所提方法的有效性，我們採納了最具有代表性的遙測空拍影像資料集 DOTA1.0[2]進行實驗與分析。從我們的實驗結果可以看出，我們修改後的 YOLOv7 模型可以準確地偵測空拍影像中的物件角度，並在各種複雜場景中取得了不錯的表現，也改善了上述所提到之水平邊界框角度偵測問題。因此，我們相信我們的研究可以將最新的 YOLOv7 技術有效應用在遙測空拍影像上，為空拍影像上的物件偵測問題提供一個新的解決方案，有望在未來的實際應用中發揮更大的作用。

## 1.2 論文架構

在本論文中，共分為五個章節。

- 第一個章節為緒論，其中包括研究動機以及論文架構。
- 第二章節將會介紹本論文相關之文獻，包括單雙階段的物件偵測模型以及旋轉物件的角度偵測以及損失計算。
- 第三章節闡述本論文之研究方法，說明本論文使用之基礎架構的優點、如何加入旋轉角度之偵測頭、與使用的數據增強方式、在損失函數中新增角度資訊，最後統整本論文提出之模型架構。
- 第四章節的部分展示本論文提出之方法所得到的實驗結果，並與不同的方法的實驗結果進行比較與分析。
- 第五章節作為總結並依據實驗結果提出相對應之結論，進一步提出未來的發展方向。



中原大學

## 第二章 相關文獻

近十年間，隨著深度學習的高速發展，在影像分類與物件辨識等方面，以 YOLO 系列為首的深度學習技術，已經被廣泛地應用在大量影像分析上。例如近幾年由 Wang 等人提出的二階段物體檢測模型 YOLOv7，又再次將此系列推向新的巔峰，其囊括高效率、輕量化以及高準確率等特性，可以說是在物件辨識的領域內立下了新的里程碑。隨著 YOLO 等深度學習技術的快速發展與實務上的成功應用，研究議題也從一般的影像內容逐漸延伸到更特定的主題與方向。例如針對小型物體的偵測、可應用於嵌入式系統的超輕量化模型等。在此同時，隨著空拍機的普及，越來越大量的空拍影像已被用於分析與觀察，其能提供的資訊也日益劇增，為了更準確地取得空拍影像中的物體範圍，在水平物體辨識的基礎上，又接續發展出加入角度的旋轉物體辨識，如 R<sup>3</sup>Det[3]、SCRDet[4]、RoI Transformer[5]等。有些從物件本身處理過程下手，提升模型對於旋轉物體學習的效率，有些則是從修改損失函數下手，針對物件的角度去求導出近似於旋轉物體交聯比的損失函數，而結果也都達到不錯的表現。讓物件辨識的技術從一開始的物件定位分類，到現在預測角度，而在這章節，我們也會一一作介紹，從物件偵測到角度偵測各種深度學習的方法與發展演變。

### 2.1 物件偵測

由於深度學習的快速發展，機器視覺的技術被大量應用於空拍影像的分析上，而其中最大宗的莫過於物件辨識的技術，而就架構層面上又可再細分為一階段物件辨識與二階段物件辨識，其分別有著彼此的優缺點，以下分別對兩種類別進行介紹。

#### 2.1.1 二階段（Two-stage）物件偵測器

二階段顧名思義便是將物件辨識的任務分為兩個階段，第一階段在原始輸入的圖片提取圖片中可能包含物件的區域，第二階段對挑選出來的候選區域各別進行物件辨識的任務。最早期提取區域的方式為滑窗法（Sliding Window），透過滑動預先設定好的窗口對整張圖片進行掃描。掃描的過程中，圖片中的物件必須剛好被窗口掃描到才會被視為偵測到該物體，同時也需要因應各式各樣的物體大小去設定窗口的大小以及滑動步長，

導致最後可能會得到大量的待處理窗口，進而使得模型的複雜度太高效率下降，不適合應用於實時任務上。爾後為了改善以上缺點，又發展出選擇性的搜尋方法，透過找出圖片中物體可能存在的潛在區域：區域候選（Region Proposal），將相似或連續區域以合併成子區域的方法去提取候選邊界框，去大幅減少候選區域，讓先前過多窗口的問題得到改善，進而達到高效率與高召回率（Recall）的結果。

提到二階段物體偵測，勢必得介紹 2013 年由 Ross Girshick 等人提出最早且最為人所知的 R-CNN[6]，如圖 2-1。也是首次將卷積神經網路（Convolutional Neural Network, CNN）應用在物體偵測上，其整體模型任務步驟為：一、對輸入圖片提取區域候選（Region Proposal），二、利用選擇搜尋（Selective Search）產生大約 2000 個區域候選，三、對區域候選大小進行尺寸標準化，四、輸入進卷積神經網路進行卷積與池化處理，五、進到模型底端的全連接層（Fully Connected Layer）進行分類以及物件位置邊界框的迴歸（Bounding-Box Regression）。

在 R-CNN 如里程碑般開創先河後，隨後便演變成 Fast R-CNN[7]。將特徵提取、區域候選標準化與物件分類全部整合進卷積神經網路（CNN）計算處理中，提升了不少整體效率。在 Fast R-CNN 之後，又進一步改進為 Faster R-CNN[8]，提出兩大新概念：區域候選網路（Region Proposal Network）與錨框（Anchor Box），前者把提取區域候選的任務一併納入網路進行學習，後者產生固定大小與數量的候選錨框去適應不同大小的物件，取代了先前選擇搜尋這種繁雜且耗時的方法，提升效率且實現端到端的實作方式。

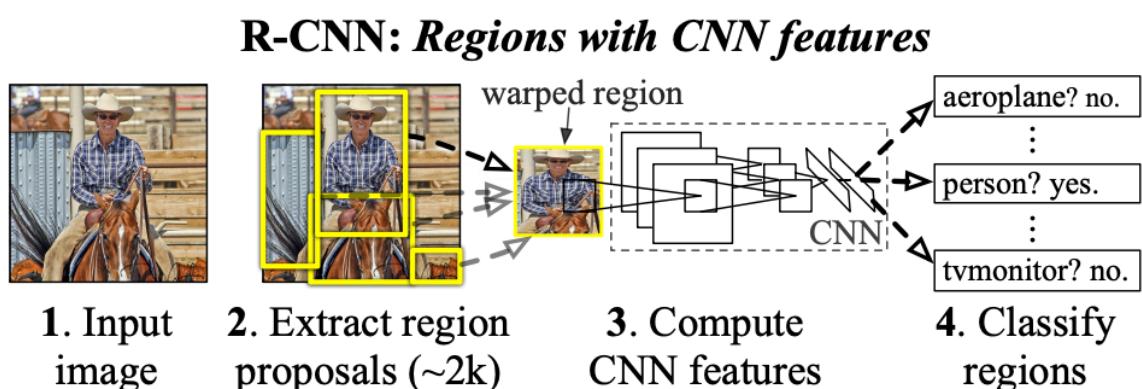


圖 2-1 R-CNN[6]模型流程圖

### 2.1.2 單階段 (One-stage) 物件偵測器

單階段物件辨識則是省略上一部份二階段物件辨識之生成候選區域的步驟，也因此常被稱為 Region-free，將物體定位與分類兩種任務合而為一，一次做到位。在效率上比起二階段物件辨識上升了一個檔次，而最著名的模型是 2015 年由 Joseph Redmon 等人提出的 YOLO ( You Only Look Once ) [9]，如圖 2-2，而其後更是衍伸出不同新的版本，對物件辨識領域的影響甚鉅。YOLO 的做法是先將圖片劃分為  $S \times S$  個網格 (Grid)，每一個網格負責預測 B 個物件中心落在此網格的物件邊界框，每個邊界框預測出五個數值，分別為邊界框中點座標 (x, y)、寬高 (w, h) 與置信度 (confidence) 與卷積神經網路得到每個類別的對應機率，並利用類別機率跟置信度進行非極大值抑制 (Non-Maximum Suppression, NMS) 去刪掉多餘的邊界框，再得到最後的預測結果。

在作者提出第一代 YOLO 之後，隨後又陸續發佈了兩個新版本，分別為結合分類數據集的 YOLOv2[10]以及採用先驗錨框(Anchor Box)與特徵金字塔網路(Feature Pyramid Network, FPN) [11]的 YOLOv3[12]。相對於二階段的偵測方法，更具彈性的框架與顯著增快的偵測效率更符合大量實際應用的需求，使得 YOLO 系列越來越具有影響力。這樣看下來，與上一段的二階段物件辨識相比，以均勻遍布的先驗錨框來取代精挑細選的區域候選，有效得到了更快的推理速度，更能應用於實時 (Real-time) 的物件辨識任務。

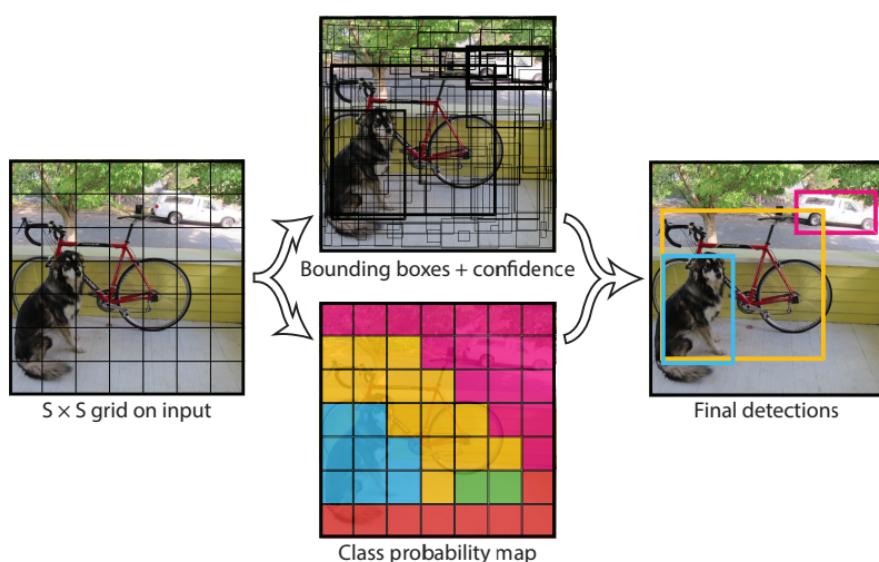


圖 2-2 YOLO[9]流程圖

## 2.2 旋轉物件偵測

隨著航拍影像分析需求的上升，對預測結果的要求也不再侷限於物件中心點、寬高跟分類而已。港口船流與十字路口的車流中的每個物體的移動方向與角度性，逐漸成為分析航拍影像的一大要點。促使各式各樣的旋轉物體偵測器的誕生，有些從整體模型運算下手，有些從完善旋轉物體交聯比損失函數下手，以下將一一介紹與本論文相關的文獻。

### 2.2.1 旋轉物件角度迴歸

從模型本身運算上入手的，一定得提及最經典的旋轉物體偵測方法 RoI Transformer，此方法由 Jian Ding 等人於 2018 年提出。主要提出了兩大貢獻，第一點提出了監督式旋轉感興趣區域學習方法（Supervised Rotated RoI Learner），如圖 2-3，將水平感興趣區域（RoI）轉換為旋轉感興趣區域（RRoI）的可學習模組。這樣的設計不僅可以有效地減輕 RoI 與目標間的誤差，也可以避免大量使用定向物體檢測的錨框；第二點設計了一個旋轉感興趣區域校準（Rotated Position Sensitive RoI Align）方法，用於空間不變特徵提取，有效地增強目標分類和邊界迴歸。處理方法本質上與水平感興趣區域校準（RoI Align）是一樣的，差別在於插值採樣時採用的是座標的角度偏移量而不是垂直水平偏移量，該方法能有效的校準旋轉物體的區域候選（Region Proposal）在映射回特徵圖（Feature map）時的座標。

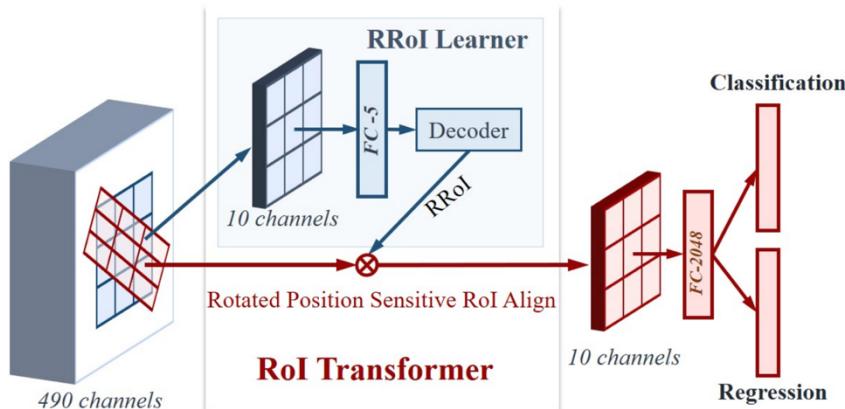


圖 2-3 RoI Tranformer[5]提出之 RRoI Learner

## 2.2.2 角度損失計算

在評估一般預測的水平邊界框時，我們常會使用預測框與真實框的交聯比（Intersection over Union）的值來評估該預測結果的好壞。有別於水平框的交聯比計算，旋轉物體交聯比的計算上本身不可導，因而沒有辦法精準計算旋轉物體的預測好壞。2019 年由 Yang Xue 等人提出的 SCRDet，作者為傳統的平滑平均絕對誤差（Smooth L1）損失函數加入了 IoU 的常數因子，在邊界情況下，新提出的 IoU Smooth L1 Loss 損失函數近似於 0，消除了角度預測上在邊界時損失的劇增。

在 2021 年，同樣也是由 Yang Xue 等人提出的 GWD[13] 損失函數則是以另一個方式來解決旋轉物件的損失計算，如圖 2-4。作者將任意旋轉矩形近似成一個二維的高斯分佈，通過計算分佈之間的 Wasserstein 距離解決旋轉物體交聯比損失計算上不可導的問題。同時也因為將單純的角度值轉換為高斯分佈的緣故，巧妙地解決了在角度迴歸上的三大問題：一、旋轉邊界的角週期性（Periodicity of Angular, PoA），二、邊的交換性（Exchangability of Edge, EoE），三、正方形角度問題（Square-Like Problem, SLP），如圖 2-5。

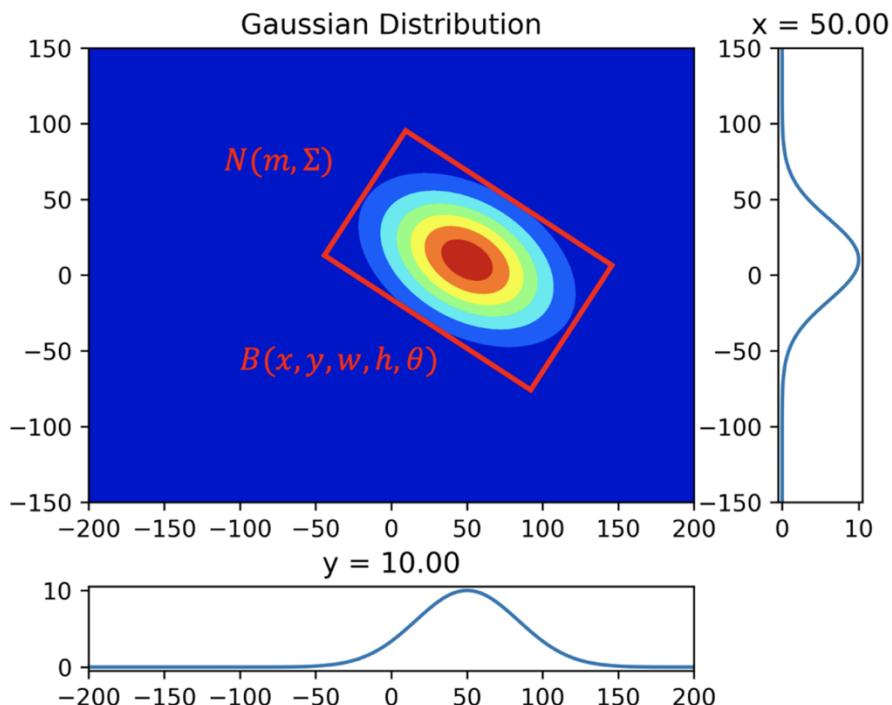


圖 2-4 GWD[13] 中的高斯分佈轉換

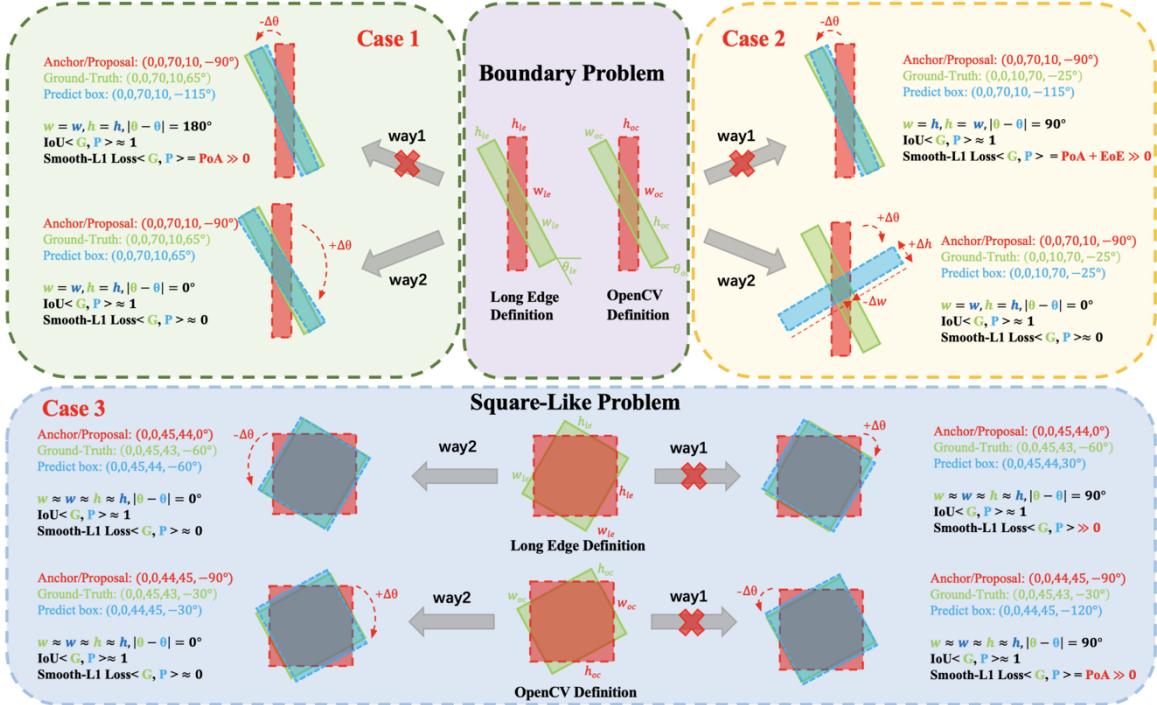


圖 2-5 GWD[13]中提及之旋轉物體角度預測的三大問題



中原大學

## 第三章 研究方法

於此章節，我們將詳細說明本論文所使用之基礎模型以及其優點，並接著介紹我們提出之修改方法。為了應付多類別以及多物體的偵測任務，我們選用本身在物件偵測上已有優異表現的 YOLOv7。並額外在模型末端之預測層中加入旋轉角度的偵測頭，使模型額外具備預測物體角度的能力；而在角度偵測的方面，為了不使模型在預測層上過於厚重，我們則是採用直接迴歸的方式去預測物體的角度，訓練與測試之流程如圖 3-1。

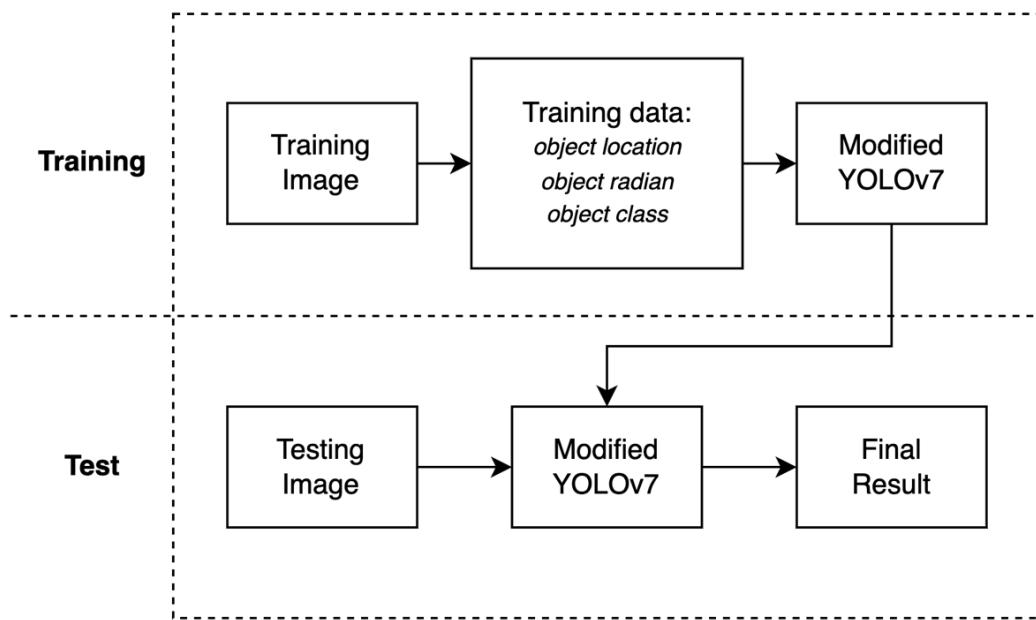


圖 3-1 本方法訓練及測試階段流程示意圖

### 3.1 物體偵測

2022 年由 Chien-Yao Wang 等人提出 YOLOv7，為 YOLO 系列提出了多種修改的方法，提高其準確性、速度以及穩健性，主要修改的部分包括三類：

#### 3.1.1 新的重參數化方法（Re-parameterized model）

作者等人發現 RepConv 中的 Identity 連結會破壞 ResNet[14]中的殘差（Residual）與 DenseNet[15]中的串連（Concatenation），提出不使用 Identity 連結的改良版 RepConv（RepConvN）來設計網路的架構，如圖 3-2。

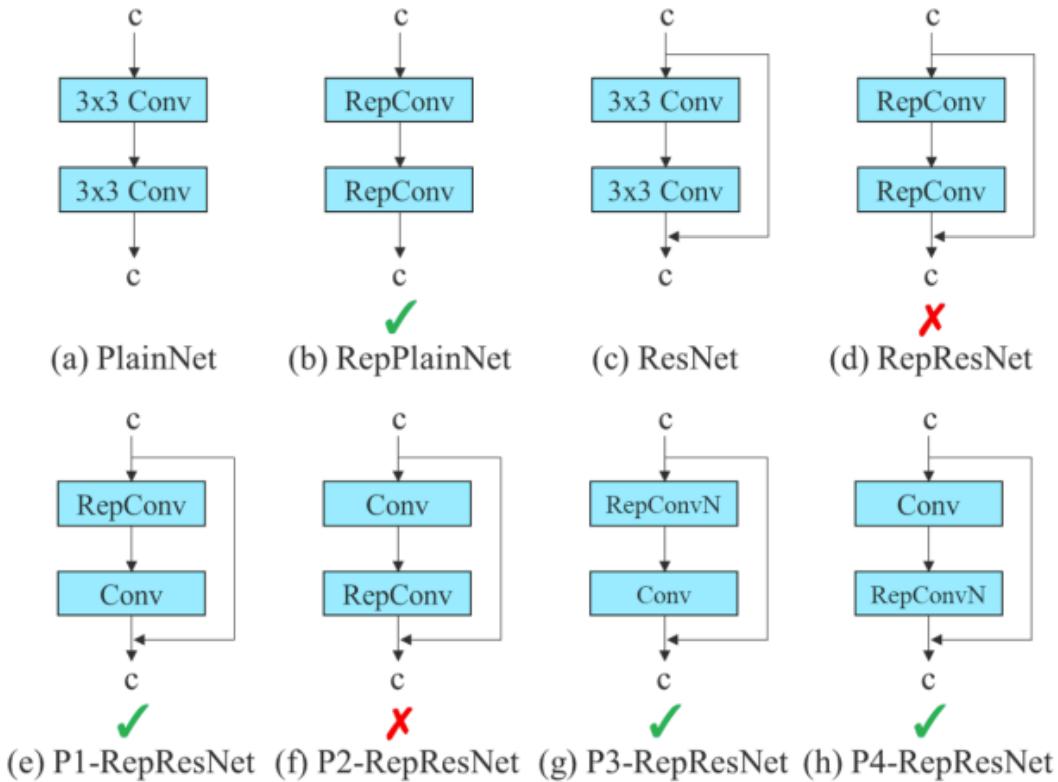


圖 3-2 YOLOv7[1]說明 Identity 連結破壞殘差與串連架構

### 3.1.2 新的動態標籤分配策略 (Dynamic label assignment strategy)

作者討論了在分類上提出兩種新的輔助頭跟主導頭分配軟標籤的做法，如圖 3-3。

(一) Lead head guided label assigner：由於主導頭有較強的學習能力，因此讓主導頭預測結果與物體真實標籤最佳化運算得出軟標籤，再將此軟標籤作為輔助頭與主導頭的真實標籤進行學習，讓較淺的輔助頭直接學習主導頭所學到的資訊，而主導頭則是更關注於未學到的殘差資訊；(二) Coarse-to-fine lead head guided label assigner：此方法是將上述方法一最佳化運算後得出的軟標籤再分為兩類 Coarse label 以及 Fine label。在 Coarse label 上放寬對正樣本的限制，讓 YOLOv7 中更多網格視為正樣本，並將此標籤用於學習能力較差的輔助頭上。而後者 Fine label 則用於主導頭，此外，在優化的任務上優先優化輔助頭的召回率，而主導頭的輸出會從輔助頭中高召回率的結果中篩選出高準確率的作為模型最後輸出。

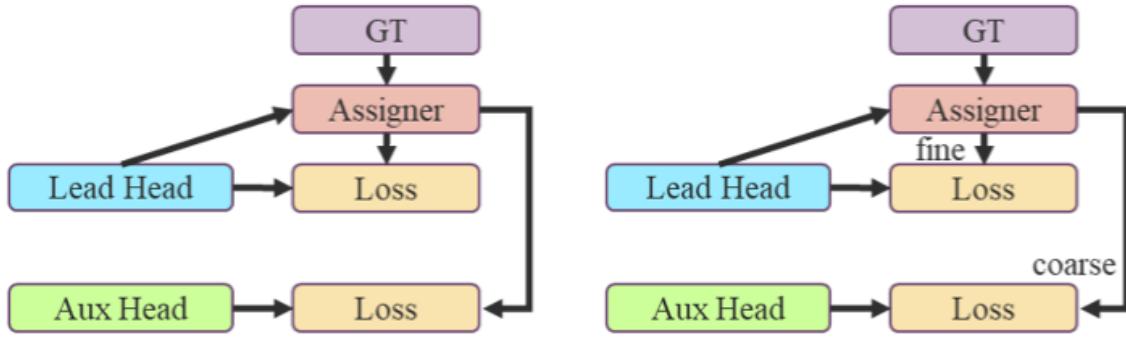


圖 3-3 YOLOv7[1]主導頭與輔助頭之動態標籤分配示意圖

### 3.1.3 模型縮放與擴展（Compound model scaling）

在之前有提及模型縮放的論文中，如 EfficientNet[16]與 Scaled-YOLOv4[17]各自基於 PlainNet 與 ResNet[14]類型的架構上進行縮放時，每層之間的輸入輸出量並不會改變，因此能夠將圖像大小、模型層數與通道數量這些常見的縮放因子與參數量、計算量之間的影響獨立分析。作者進一步分析出在以計算模組（Computational block）串連的基礎的模型下並不能直接套用上述兩種模型的縮放方法，由於對於深度進行縮放將影響到後續計算模組（Computational block）的過渡層（Transition layer）的輸入，如圖 3-4 所示。此外也提出了相對應的解決方法，在深度進行縮放完後，計算計算模組（Computational block）輸出通道的變化量，並以該變化量對過渡層（Transition layer）的寬度進行相對應變化量之縮放，如圖 3-4。

### 3.1.4 更快的速度與更高的準確率

因為以上的修改以及優化使得 YOLOv7 與之前的最頂尖（State of the Art, SOTA）的即時物件偵測模型相比降低了 40% 參數量、50% 每秒浮點運算次數（FLOPs），並有更快的推理運算速度及準確率。此外，所有模型皆非轉移學習，如圖 3-6，YOLOv7 對比其他 YOLO 系列在精度以及速度上都得到了不小的提升，可說是擁有非常穩健的表現。

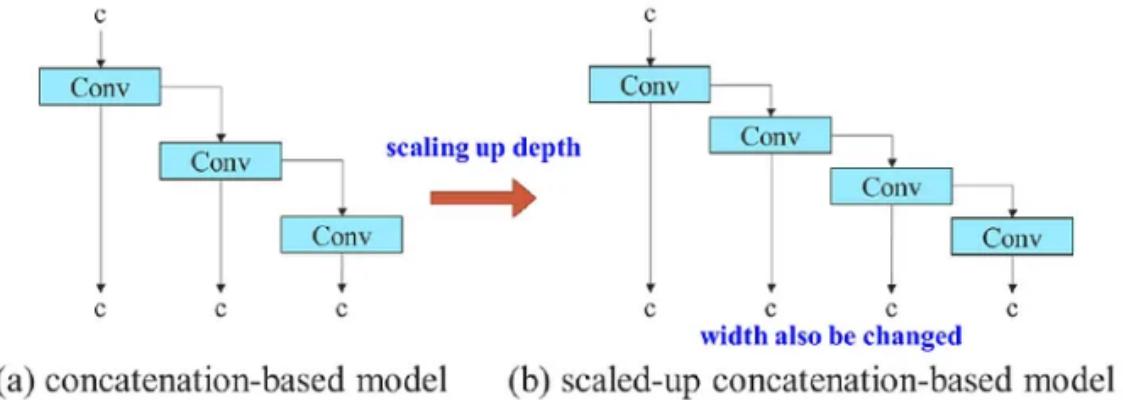


圖 3-5 Computational block 縮放後導致 Transition layer 輸入通道寬度縮放示意圖

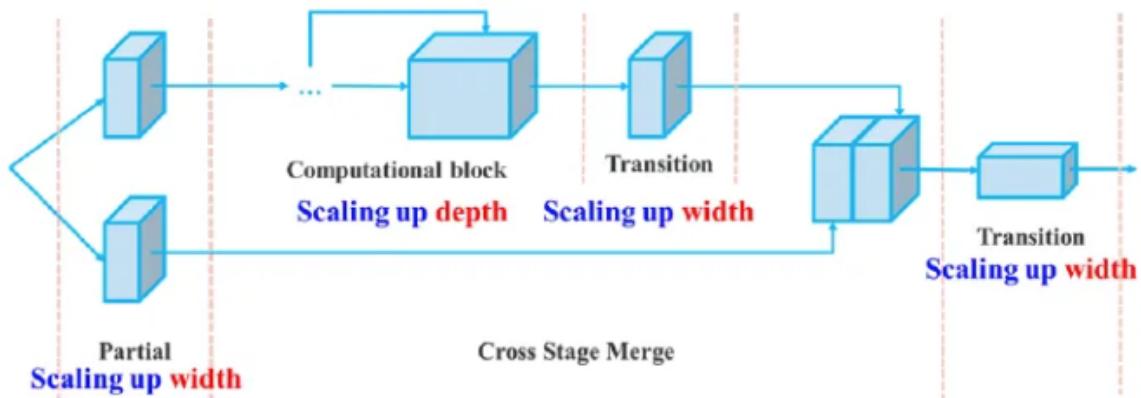


圖 3-4 YOLOv7[1]所提出之模型模組縮放解決方法

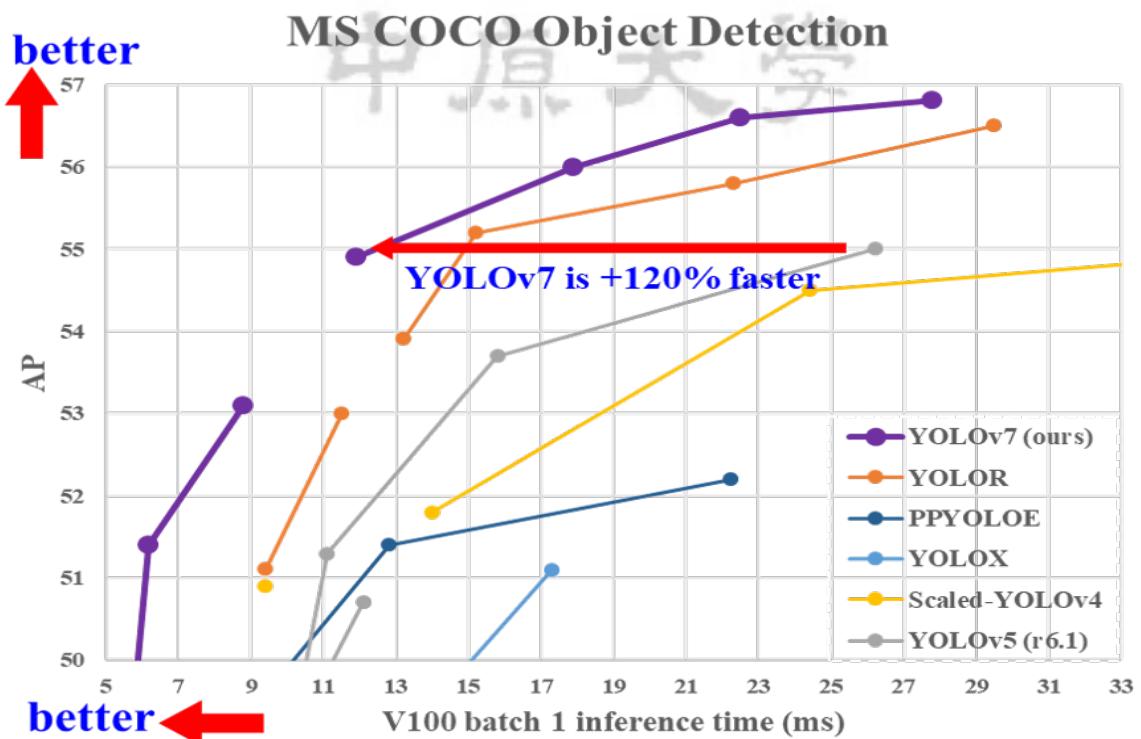


圖 3-6 YOLOv7[1]與其他 YOLO 系列之速度與精度比較

## 3.2 角度偵測

### 3.2.1 旋轉邊界框之角度定義

DOTA 資料集有自身原本的標籤格式，為了資料集能順利輸入進我們所修改的模型訓練。我們將資料集原先物體的四個角點的座標格式轉換為中心點座標( $x, y$ )、寬度(w)、高度(h)與角度( $\theta$ )等五個參數的 YOLO 格式，在後面三者的值域定義上又因認知上的不同，分為 OpenCV 表示法與長邊表示法。上述兩種表示法雖然可以明確表達物體的角度，但為了能表示像十字路口車流等動態物體方向性，我們在角度( $\theta$ )的值域上提出如圖 3-7(c)，以 X 軸正向為 0，逆時針為正，最大值 360 度為 1 的標準化角度表示法，以達到在預測時及時看出物體的方向性，三種角度表示法差異如圖 3-7。

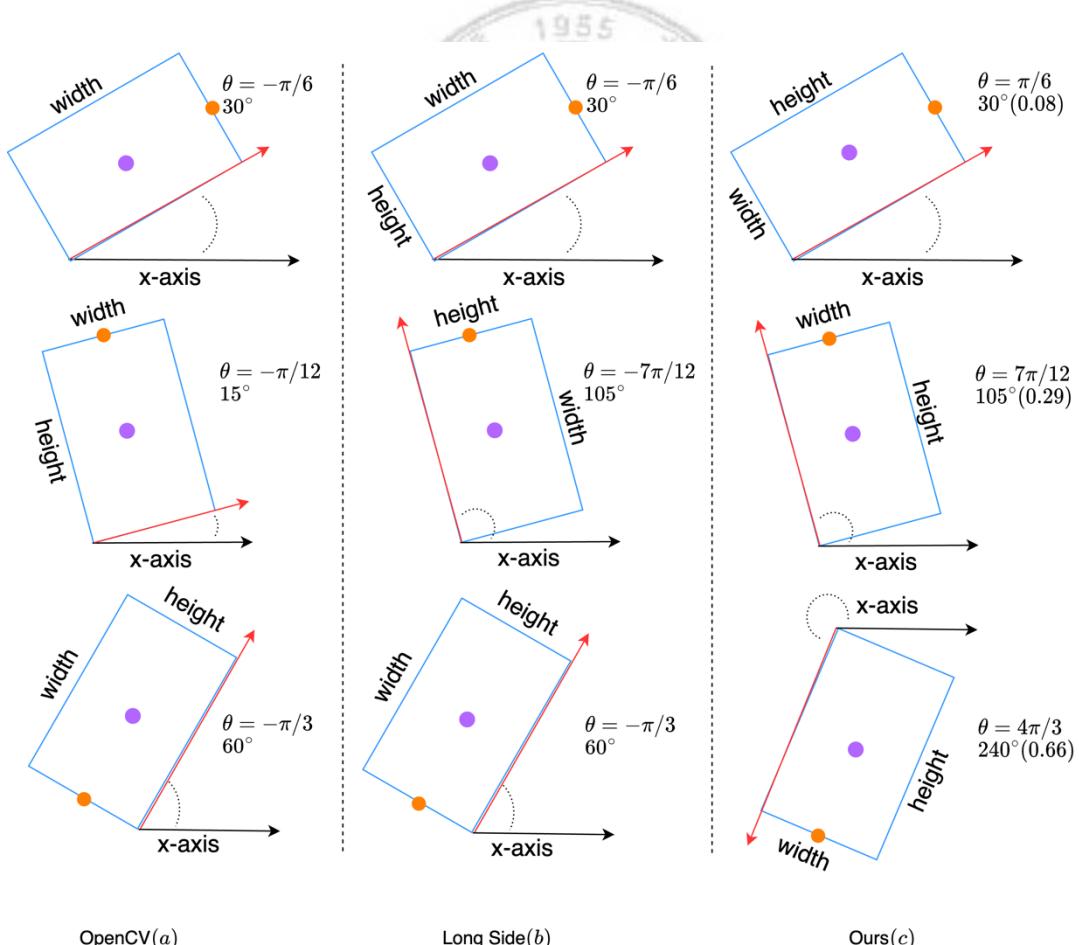


圖 3-7 不同物件角度之定義與本論文對角度之定義

### 3.2.2 角度偵測頭

YOLOv7 本身在最後的偵測頭部分，原先只有迴歸物件中心點( $x, y$ )、物件長寬( $w, h$ )四個值，而 YOLOv7 有別於以往的 YOLO 系列模型，在對模型的深度進行縮放時，加入了新的網路架構 ELAN[18]，如圖 3-8。同時為了維持深度縮放完的輸入輸出的大小，計算深度縮放的變化量並對寬度進行同樣的處理，進而控制最短與最長梯度路徑，使更深的網絡可以有效地學習和收斂。

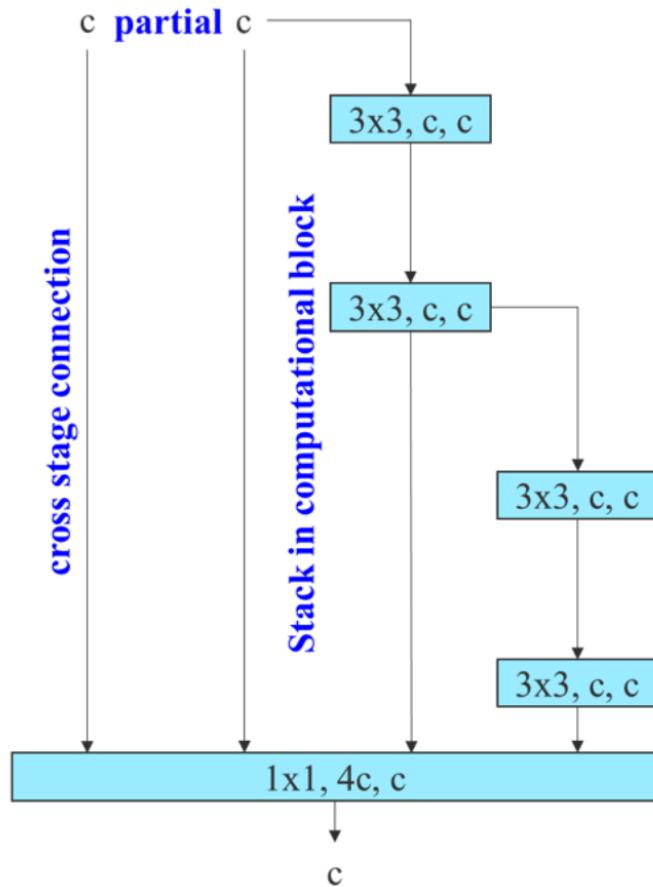


圖 3-8 YOLOv7 提出之 ELAN[18]架構

而本論文所提出之額外的角度數值預測上，我們在模型最後的預測層中另外加入了  
一個角度分支偵測頭，去另外對於物體角度進行預測，如圖 3-9。在模型最後預測輸出  
中，三種尺度下的特徵圖中，每個網格將負責預測一個邊界框，並附帶（物體類別數  
 $C+4+1+1$ ）個參數，如圖 3-10。而邊界框的位置由框的中心點  $(x, y)$  與框的寬高  $(w,$   
 $h)$  所組成，並包含該邊界框所匡列的物體之置信度與分類，最後還有我們所加入之角度  
預測值，代表物體的旋轉角度。

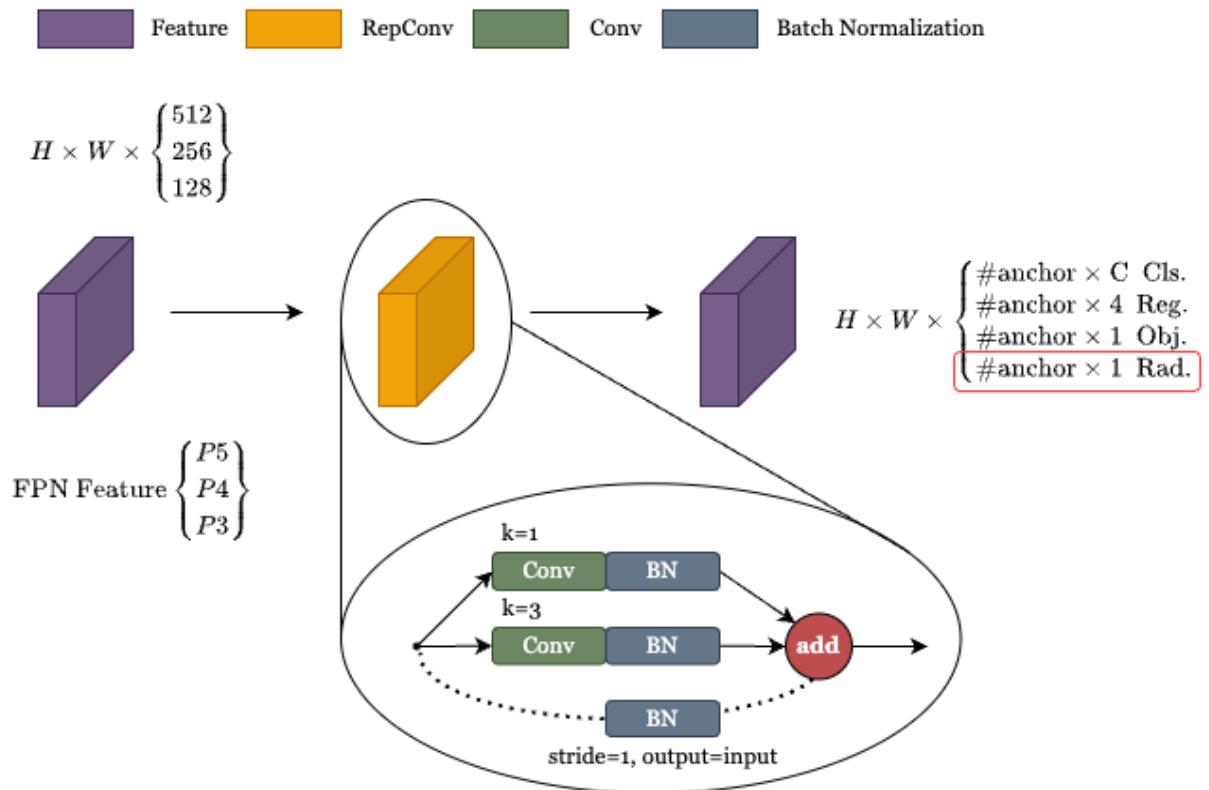


圖 3-9 YOLOv7 偵測頭以及我們所加入之角度偵測頭

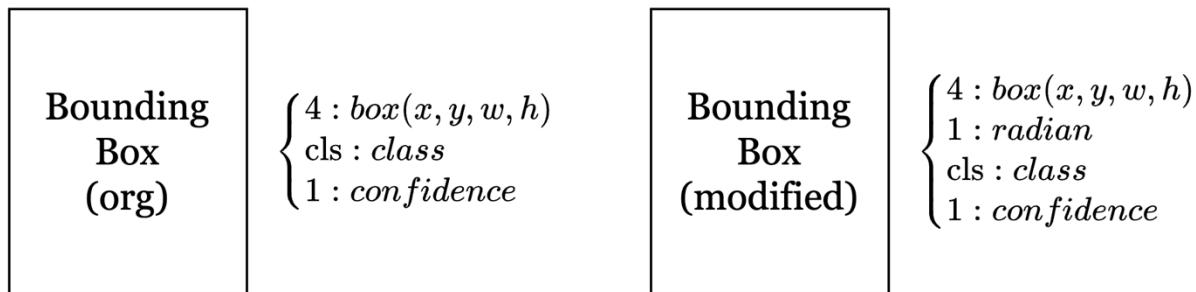


圖 3-10 邊界框預測資訊

### 3.3 數據增強

#### 3.3.1 馬賽克（Mosaic Data Augmentation）

馬賽克數據增強[19]由 Alexey Bochkoviskiy 等人於 2020 年提出，參考自 2019 年 Sangdoo Yun 等人提出之 CutMix[20]數據增強，在兩張拼貼的影像基礎上，如下圖採取四張影像進行拼貼，並進行隨機的縮放、剪裁、平移與傾斜等處理過後生成最終圖像。圖像隨機處理完後，再針對影像標註檔進行相對應的更新。如上述所說，在對資料集經過多重隨機加工後，大幅提升資料集的豐富性與數量。值得一提的是，在做縮放的同時也會將小物件放大，使得這些物件得以被模型學習，進而提升網路模型的性能。不過此數據增強本身建立於水平邊界框的基礎，對影像與標註檔的轉換，都是以水平邊界框的座標進行處理，而本篇論文的資料集均是採用旋轉擬合框，存在物件偏斜角度。所以我們也針對此部分進行程式的修改，使的旋轉物體在進行馬賽克數據增強過後，影像本身與標註檔依舊能維持對應的座標；輸入圖片經過馬賽克數據增強過後的結果如下圖 3-11。



圖 3-11 馬賽克數據增強示意圖

### 3.3.2 翻轉 (Flip up-down, Flip left-right Data Augmentation)

針對上段馬賽克數據增強過後的影像，將再進行隨機的水平翻轉或垂直翻轉如圖3-12，也因為旋轉邊界框跟水平邊界框在數值上轉換方式不同，在數據增強後，對角度值本身進行相對應的更新。

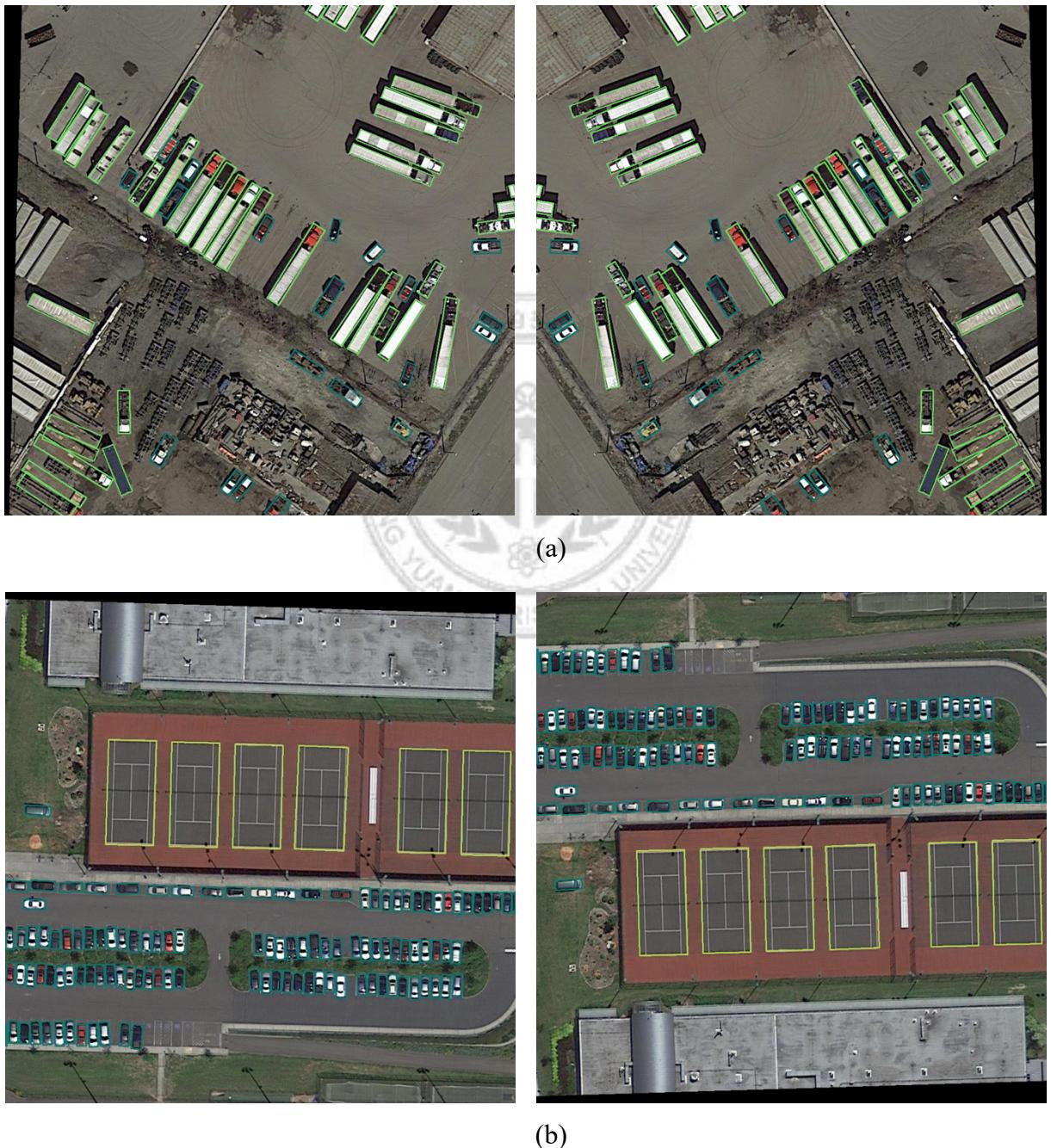


圖 3-12 翻轉資料增強示意圖，(a)為水平翻轉，(b)為垂直翻轉

### 3.4 基於 YOLOv7 之空拍單階段旋轉邊界框物體偵測器

本篇論文旨在基於 YOLOv7 此一單階段物體偵測方法，使模型輸出具備預測物體角度的能力。由於 YOLOv7 本身訓練採用的為常見的水平邊界框，但為了滿足具有方向性的空拍影像旋轉邊界框之需求，將水平邊界框轉為旋轉擬合框，進而減少偵測結果中的冗余背景資訊。我們提出基於 YOLOv7 模型的旋轉擬合框物體偵測方法，在訓練模型時，我們將訓練資料輸入進模型時的解析度定為  $640 \times 640$ ，並如圖 3-13。經過骨幹網路的特徵提取後，取得三種不同尺度的特徵圖。之後將上層特徵圖的特徵上採樣融合至下層，並在原先的特徵金字塔網路（FPN）上引入路徑聚合（Path Aggregation）[21]，對下層特徵圖進行下取樣融合至上層，以此結合下層精確的物體位置和上層豐富的語意特徵等特性，藉以提升整體模型的性能。

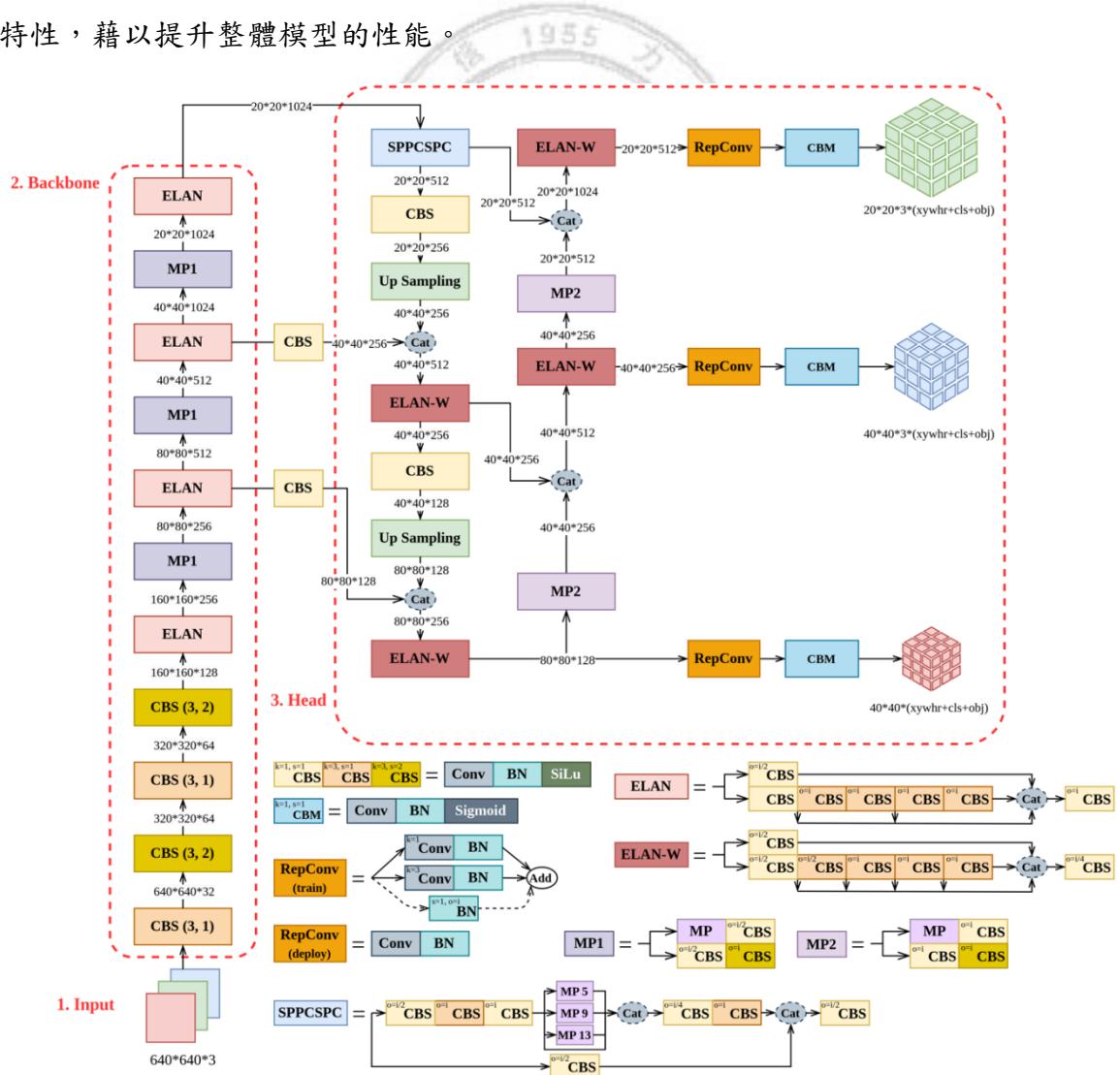


圖 3-13 本論文所使用之 YOLOv7 網路架構圖

最後提及每個神經網路架構中極其重要的角色：損失函數（Loss Function），它往往左右模型的收斂速度與梯度方向的正確與否。而我們在物件邊界框上使用的是交聯比損失函數（Intersection over Union Loss, IoULoss）；物體置信度與物體分類則是使用包含 Sigmoid 數字標準化的二元交叉熵損失函數（Binary Cross Entropy Loss, BCELoss），分別為底下的式(1)、式(2)與式(3)。二元交叉熵損失函數大部分用來評估二分類的問題；在基礎模型的角度值的損失計算上，我們選用平滑平均絕對誤差函數（Smooth L1 Loss, SL1Loss）。在平均絕對誤差與平均方差函數的基礎下，讓離群值的梯度影響力縮小，避免梯度爆炸，也使得梯度在原點可導，修正平均絕對誤差在原點之不平滑問題。但因為 Smooth L1 Loss 在損失計算上，是算預測值與真實值的絕對差值。但在如圖 3-14 的情況上為了使模型更能學習到物體角度值該迴歸的方向，我們希望模型去學習最小的角度差，數值計算如式(6)，而不是預測與真實值的絕對差值。所以我們將第一部分額外的角度損失去除，並額外計算預測值與真實值的最小角度差。並在模型計算交聯比損失時，分別嘗試利用最小角度差的額外計算來賦予當前模型預測的交聯比角度損失訊息。第一個嘗試我們直接利用最小角度差之倒數作為因子、第二個嘗試我們利用正切值在  $0 - \pi/2$  會逐漸從 0 趨近於無限大的特性，同樣利用導數作為因子，強化角度差初期的收斂趨勢。同時為了避免分母為 0，兩種嘗試我們都於分母的部分加一，添加完後交聯比之損失修改如表 3-1。

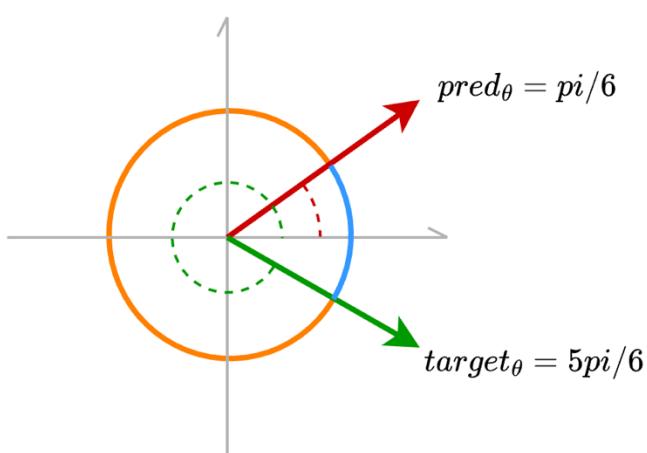


圖 3-14 絕對角度差與最小角度差示意圖

$$L_{IoU} = 1 - IoU \quad (1)$$

$$L_{confidence} = BCELoss(pred, target) \quad (2)$$

$$L_{class} = BCELoss(pred, target) \quad (3)$$

$$L_{radian} = SmoothL1(pred, target) \quad (4)$$

$$L_{total} = W_{iou} \times L_{iou} + W_{confidence} \times L_{confidence} + W_{class} \times L_{class} + W_{radian} \times L_{radian} \quad (5)$$

$$\text{min\_angle\_diff} = \min(|pred_\theta - target_\theta|, 2\pi - |pred_\theta - target_\theta|) \quad (6)$$

表 3-1 交聯比與角度損失權重組合表

	$L_{iou}$	$L_{radian}$
Baseline	$1 - IoU$	$SmoothL1(pred, target)$
IoU weight based on normalized angle	$1 - \left( \frac{1}{1 + \frac{\angle A(pred, target)}{\pi}} \times IoU \right)$	0
IoU weight based on tangent	$1 - \left( \frac{1}{1 + \tan(\frac{\angle A(pred, target)}{2})} \times IoU \right)$	0

## 第四章 實驗結果與分析

### 4.1 實驗環境

本實驗使用以 Docker 創建的 Container 作為實驗環境；硬體上 Host 主機為搭載 AMD Ryzen 9 5950X 3.4GHz 16 核，RAM 為 128G，顯卡為一張 NVIDIA GeForce RTX 3090 24G，Container 創建時 CPU 設定為 6 核，RAM 32G；軟體上 Python 版本為 3.10.9，PyTorch 版本為 2.0.0，CUDA 版本為 11.4。

### 4.2 實驗資料

本論文在訓練以及驗證上使用公開的 DOTA1.0，本資料集的圖片大小介於  $800 \times 800$  到  $20000 \times 20000$  像素，包含了各種長寬比、方向以及形狀的物體，並且分為 15 個常見類別。在總數 2,806 張影像中一共包含 188,282 個物體。資料集中初始的標注格式為旋轉擬合框的四個角點座標，而第一點為物體左上角的角點座標，且四個點按順時針的方向排列。如圖 4-1 為原始資料集的標注可視化，黃色頂點為旋轉擬合框的起點，分別對應圖飛機的左上角、圖大型汽車左上角、圖扇形棒球場之本壘位置。

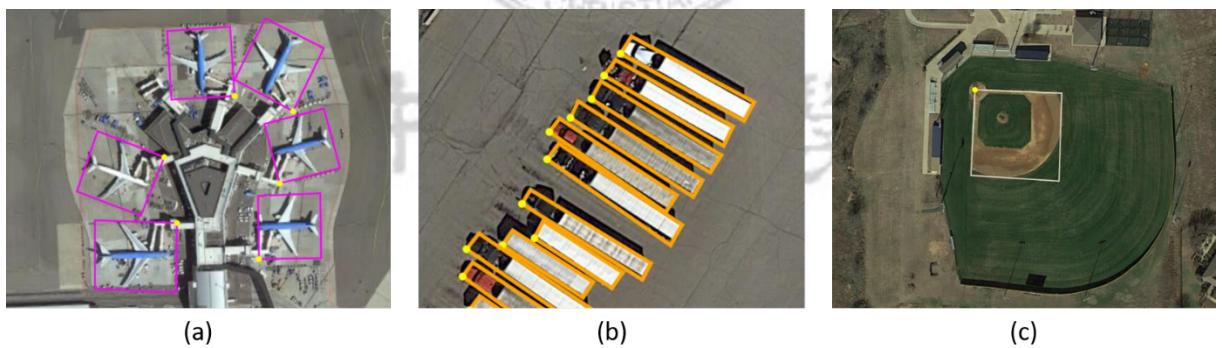


圖 4-1 DOTA-1.0[2]標註檔可視化影像

此外 DOTA 資料集這類的遙測空拍資料集，因為解析度過大，若直接輸入模型進行訓練，勢必會超越顯示卡記憶體的容量。為了符合顯示卡的記憶限制，必須對輸入影像大小進行內插縮小，這樣雖然滿足了顯示卡的記憶體限制，卻反而因為解析度的縮小而損失大量原始影像上的資訊，大幅降低對於小物體的檢測能力。因此我們事先針對資料集進行重疊裁切的前處理動作，將原本的影像切割為  $1024 \times 1024$  像素大小，水平與垂

直方向重疊部分取 200 像素，確保有一定的重疊區域，而切割前後物體數量的比對於表 4-1。從表中也可以發現 DOTA 的類別比例十分不均。最後在輸入進模型做訓練時，先將原始的 DOTA 標注格式 ( $x_0, y_0, x_1, y_1, x_2, y_2, x_3, y_3, \text{class}$ ) 轉換為類似 YOLO 的標注格式 ( $\text{class}, \text{center\_x}, \text{center\_y}, \text{object\_width}, \text{object\_height}, \text{object\_radian}$ )，再將訓練圖像縮放至 640x640 進行正式訓練。如圖 4-2，分別為飛機 (Plane, PL)、棒球場 (Baseball Diamond, BD)、橋墩 (Bridge, BR)、田徑場 (Ground Track Field, GTF)、小型車輛 (Small Vehicle, SV)、大型車輛 (Large Vehicle, LV)、船隻 (Ship, SH)、網球場 (Tennis Court, TC)、籃球場 (Basketball Court, BC)、儲油罐 (Storage Tank, ST)、足球場 (Soccer Ball Field, SBF)、圓環 (Roundabout, RA)、港口 (Harbor, HA)、游泳池 (Swimming Pool, SP) 以及直升機 (Helicopter, HC)。

表 4-1 DOTA 資料集分割前後類別數量比較表

		Images	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	Total
Before	Train Set	1411	8055	415	2047	325	26126	16969	28068	2367	515	5029	326	399	5983	1736	630	98990
	Valid Set	458	2531	214	464	144	5438	4387	8960	760	132	2888	153	179	2090	440	73	28853
After	Train Set	29457	31984	1397	6852	1501	76546	52248	104873	6197	1763	19624	1460	1461	22013	5991	2373	333910
	Valid Set	10132	10590	705	1637	574	13863	12355	32381	1967	453	10574	635	600	7152	1301	304	95237



圖 4-2 DOTA-v1.0 所有類別之範例圖

### 4.3 實驗結果

在初始參數的設定上，輸入的圖片大小我們固定為  $640 \times 640$ ，訓練次數為 150 個 Epoch，初始學習率為  $1e-3$ ，加上 PyTorch 的餘弦退火學習法（Cosine Annealing），使學習率透過餘弦函數的速度進行下降，其權重衰減為  $5e-4$ ，同時也使用隨機梯度下降法（Stochastic Gradient Descent, SGD）且動量設為 0.9。

數據擴增方面，由於遙測航拍中具有不同大小之物體並各自帶有角度資訊，因此如同上一章節所提到，我們使用馬賽克（Mosaic）與翻轉（Flip left-right & Flip up-down）兩種增強法來提升我們模型的整體性能。損失計算上，預測框使用交聯比損失，角度損失使用平滑平均絕對誤差損失，置信度與分類分別使用二元交叉熵損失。

第一部分的實驗我們以最直覺的方式直接迴歸物體的角度值，並另外賦予模型多一個角度預測的損失。在損失權重上  $W_{iou}$ 、 $W_{confidence}$ 、 $W_{classification}$  與  $W_{radian}$  上，我們分別使用 0.2、0.7、0.3 與 0.5。在檢測時，檢測輸入影像大小為  $640 \times 640$ ，並使用水平邊界框之非極大值抑制過濾冗余的框，交聯比的閥值設定為 0.3，在交聯比為 0.5 的標準下，平均精度均值為 65.74，檢測結果如圖 4-3。可以看到單純只加入一個通道讓模型直接回歸角度值，雖然可行，但是成果不是很理想，距離實際應用更是相差甚遠，需要加入更好的因子去給予模型更多更準確關於角度值的資訊。所以我們額外加入上一章節所提到的最小角度差資訊進行訓練，而模型分別的表現如表 4-2。

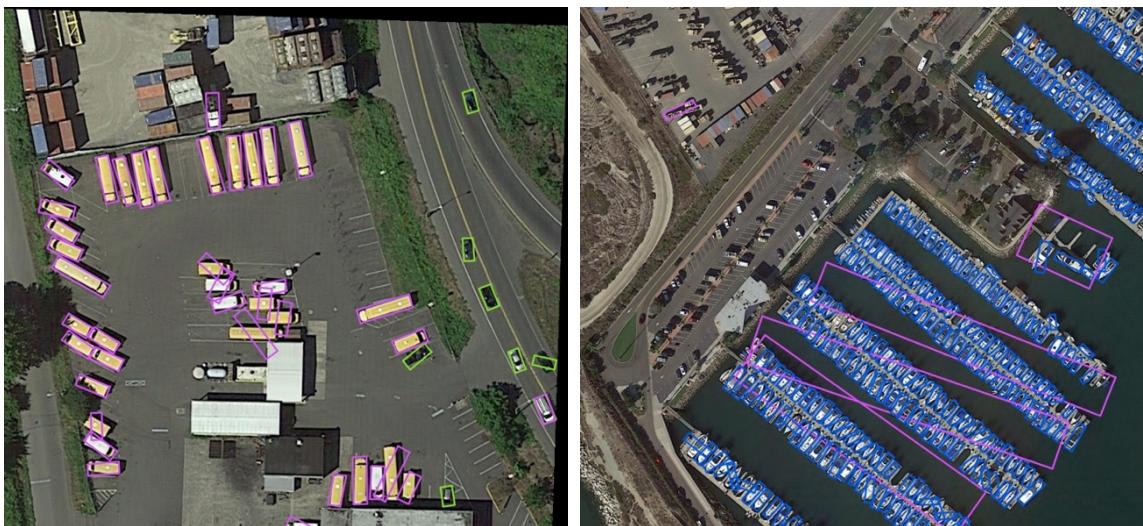


圖 4-3 使用基礎參數訓練的預測結果圖

表 4-2 初始參數與角度加權係數模型表現比較

IoU Multiply Factor	mAP@0.5	mAP@0.5:0.95	Precision	Recall
Baseline	65.74	38.6	74.08	63.13
IoU weight based on normalized angle	64.31	39.96	72.87	62.43
IoU weight based on tangent	<b>67.4</b>	<b>41.19</b>	<b>74.47</b>	<b>67.16</b>

可以從上表看出在加入最小角度差與最小角度差之正切值作為交聯比係數之後，平均精度均值皆有所提升。在為邊界框交聯比加入最小角度值之倒數係數之後，雖然在交聯比閥值為 0.5 的情況下，平均精度稍微下降，但在交聯比 0.5 到 0.95 的平均精度上，可以看到有稍微的提升，表示在高質量的預測結果上平均下來表現是比起初始參數的結果更為優秀的。之後我們繼續嘗試利用最小角度差的正切倒數值的倒數去強化預測角度與真實角度的損失資訊，訓練結果為所有結果中最好的，在角度預測的表現上，大幅度的得到進步，與初始參數訓練之模型比較結果如下頁圖 4-4 與圖 4-5。可以看到在我們加入了最小角度之正切值資訊後，對於旋轉物件角度偵測之其二難題：一、密集排列之物體，二、較大長寬比之物體，都得到良好的改善。

圖 4-6 為本模型之偵測結果錯誤情形，可以看到其中圈起來的部分為長寬比較大的物體，在這種情況下模型有時在物體中心點定位上會稍微有偏移的結果產生。而這樣的錯誤可能是利用加權係數將物體交聯比與角度資訊融合所產生的問題，如要解決的話，則可能需要將物體交聯比與角度資訊完全拆開，去以分類的方法處理角度資訊或以沒有角度迴歸問題之損失函數去計算角度損失。

最後第三部分為本篇論文第三個實驗。交聯比損失函數作為一個反應模型的預測框與真實框之間的誤差評分，在我們為交聯比損失函數添加完加權係數後，我們分別對三種不同計算交聯比的算法進行相關實驗，並觀察不同計算方法對角度偵測的影響，如表 4-3。實驗結果發現旋轉物體偵測在使用 DIoU 的交聯比計算方法上的表現是比較好的，也比較能適應額外的角度資訊與我們另外添加的加權係數，各類別之精度也如表 4-4。

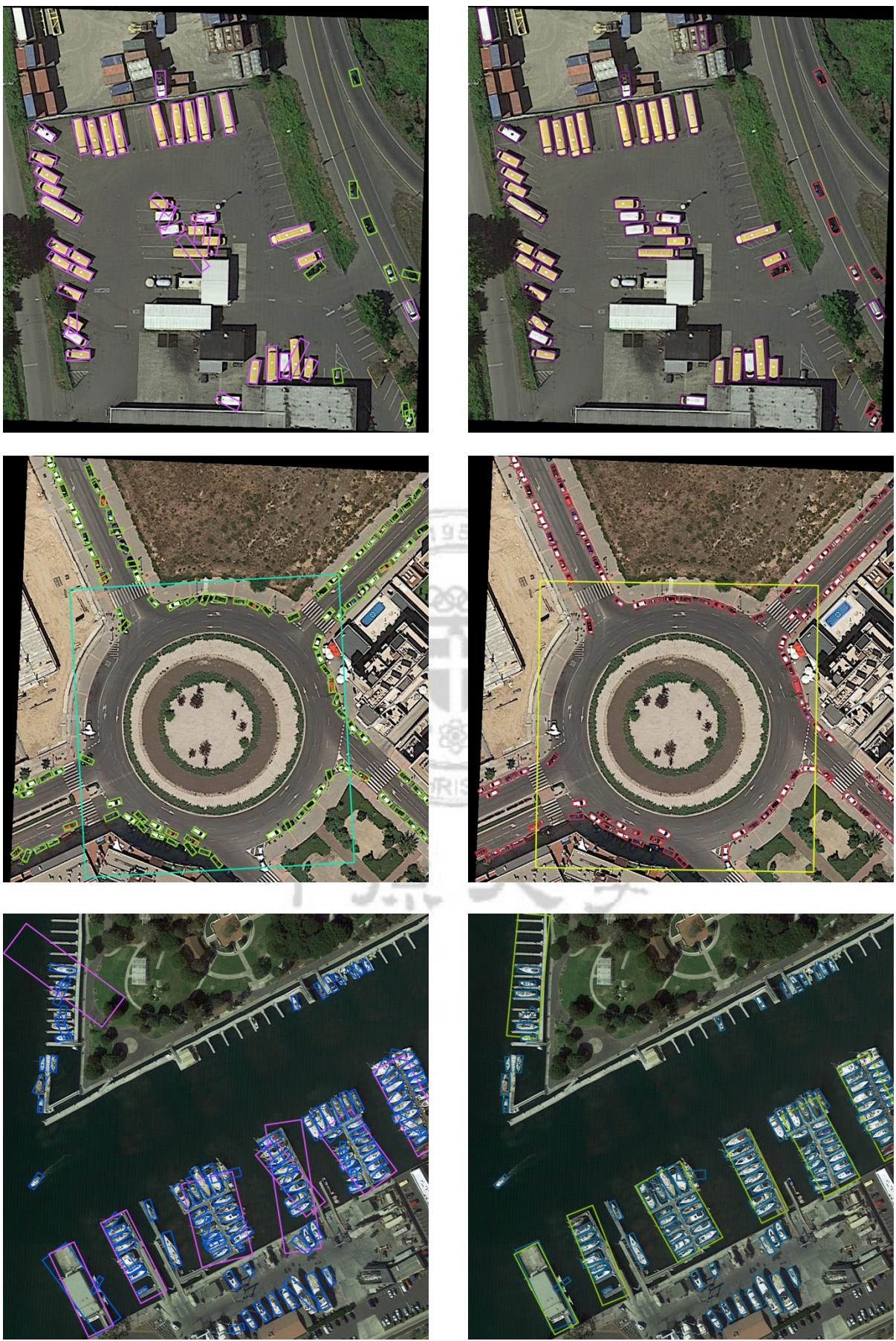


圖 4-4 初始參數（左）與加入正切倒數加權係數（右）模型表現比較（1）

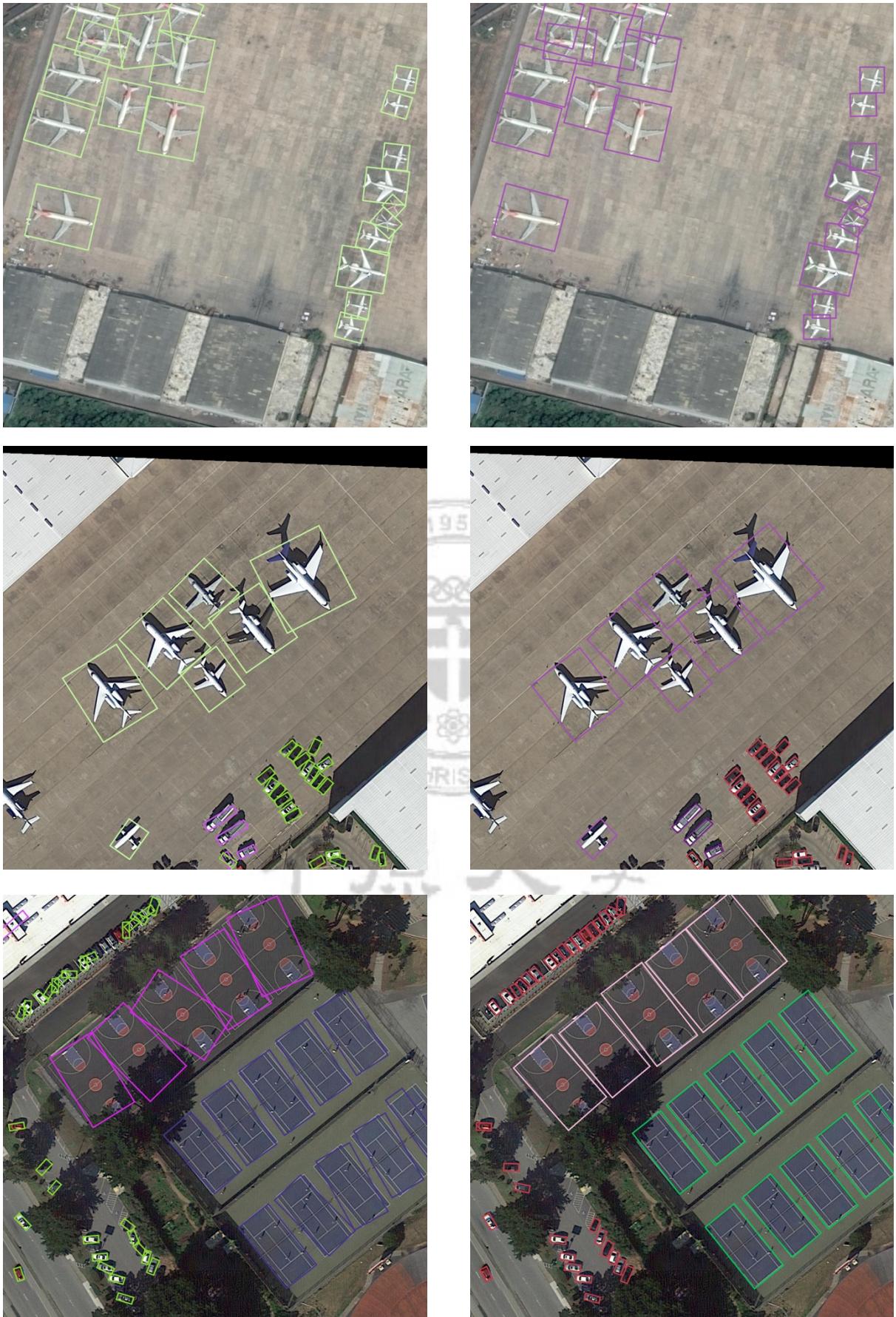


圖 4-5 初始參數（左）與加入正切倒數加權係數（右）模型表現比較（2）

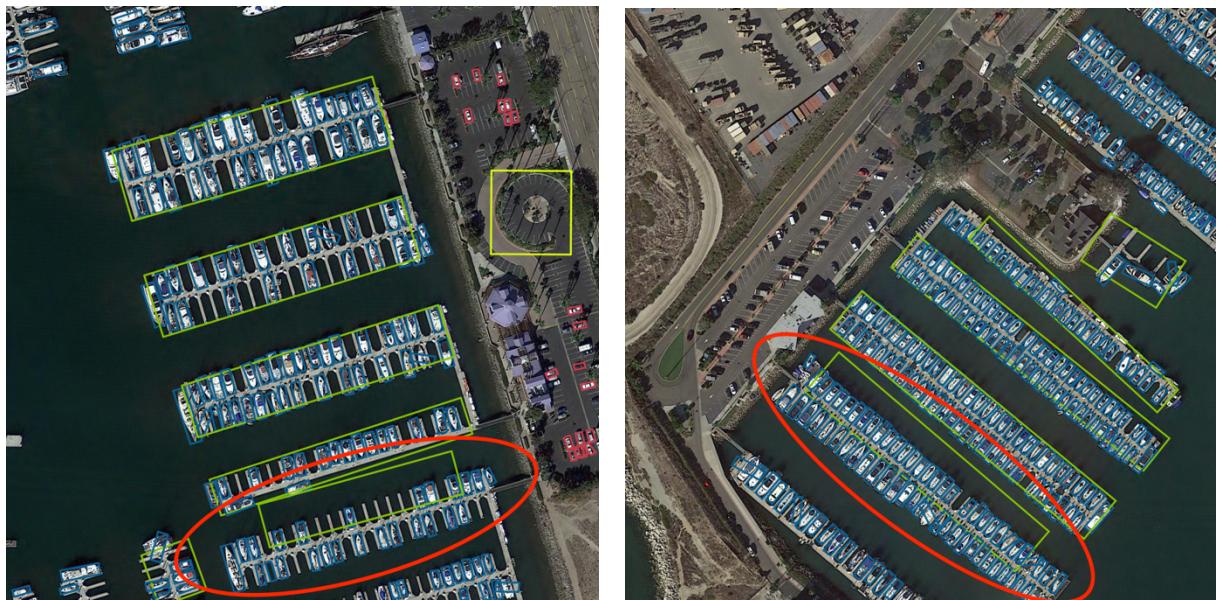


圖 4-6 偵測錯誤之結果

表 4-3 不同交聯比計算方法模型表現比較

IoU Method	mAP@0.5	mAP@0.5:0.95	Precision	Recall
CIoU	67.41	41.19	74.47	<b>67.16</b>
GIoU	66.63	41	76.53	64.57
DIoU	<b>67.9</b>	<b>41.62</b>	<b>76.58</b>	65.97

表 4-4 各類別之平均精度與平均精度均值

	AP																mAP
	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC		
Baseline	87.8	68.8	34.7	<b>64.1</b>	55.4	74.5	83.4	91.2	63.2	75.4	<b>55.3</b>	56.4	63.7	<b>63.6</b>	<b>49</b>	65.8	
min	89.9	65.3	39.6	54.3	60.4	<b>79.8</b>	86.8	<b>92.2</b>	62.7	75.5	54.2	50.7	67.2	63.2	23.4	64.4	
tangent	<b>90.4</b>	<b>74.4</b>	<b>42</b>	63.6	<b>61.5</b>	79	<b>86.9</b>	91.5	<b>63.5</b>	<b>76.3</b>	54.7	<b>57.4</b>	<b>69.4</b>	63.5	37.6	<b>67.4</b>	

## 第五章 結論與未來方向

深度學習與物件偵測在這幾年已經發展的十分優秀，而由此延伸出的旋轉物件偵測的方法也是百百種。有些人從損失函數著手，有些人將迴歸改成分類，也都取得不錯的表現，不過都是非常巨量的改動，牽涉到標註格式以及模型底層運作方式。基於以上的條件，我們想以一個更簡單的出發點去處理旋轉物件偵測任務，希望能以更輕鬆直覺的方式讓模型學習到正確的角度值。

在本篇論文中，我們為了不讓模型的偵測層過於厚重，選擇繼續沿用迴歸的方式獲得空拍影像中物體的角度值。從最後的實驗結果能看出，我們對模型提出的修改能有效的解決長短邊互換以及角度邊界週期性問題。在角度值的預測上，已有一定程度的準確率，代表在旋轉物件偵測的領域上，在修改上是有著適當著手點以及可能性的。利用簡單的角度差資訊因子，便能與交聯比損失做結合，使得模型更具備學習物體角度的能力。當然也可以從實驗的最後看出本修改方法對於長寬比的物體雖然能準確的預測角度值，但在少數情形下還是會有中心點偏移的狀況發生，希望未來的研究能繼續找出此問題的解決方案，進一步完善我們的成果。

中原大學

## 參考文獻

- [1] Wang, C., Bochkovskiy, A., & Liao, H.M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *ArXiv, abs/2207.02696*.
- [2] Xia, G., Bai, X., Ding, J., Zhu, Z., Belongie, S.J., Luo, J., Datcu, M., Pelillo, M., & Zhang, L. (2017). DOTA: A Large-Scale Dataset for Object Detection in Aerial Images. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3974-3983.
- [3] Yang, X., Liu, Q., Yan, J., & Li, A. (2019). R3Det: Refined Single-Stage Detector with Feature Refinement for Rotating Object. *AAAI Conference on Artificial Intelligence*.
- [4] Yang, X., Yang, J., Yan, J., Zhang, Y., Zhang, T., Guo, Z., Sun, X., & Fu, K. (2018). SCRDet: Towards More Robust Detection for Small, Cluttered and Rotated Objects. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8231-8240.
- [5] Ding, J., Xue, N., Long, Y., Xia, G., & Lu, Q. (2019). Learning RoI Transformer for Oriented Object Detection in Aerial Images. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2844-2853.
- [6] Girshick, R.B., Donahue, J., Darrell, T., & Malik, J. (2013). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp.580-587.
- [7] Girshick, R.B. (2015). Fast R-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440-1448.
- [8] Ren, S., He, K., Girshick, R.B., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*, pp. 1137-1149.
- [9] Redmon, J., Divvala, S.K., Girshick, R.B., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779-788.

- [10] Redmon, J., & Farhadi, A. (2016). YOLO9000: Better, Faster, Stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6517-6525.
- [11] Lin, T., Dollár, P., Girshick, R.B., He, K., Hariharan, B., & Belongie, S.J. (2016). Feature Pyramid Networks for Object Detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936-944.
- [12] Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *ArXiv*, [abs/1804.02767](https://arxiv.org/abs/1804.02767).
- [13] Yang, X., Yan, J., Ming, Q., Wang, W., Zhang, X., & Tian, Q. (2021). Rethinking Rotated Object Detection with Gaussian Wasserstein Distance Loss. *ArXiv*, [abs/2101.11952](https://arxiv.org/abs/2101.11952).
- [14] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778.
- [15] Huang, G., Liu, Z., & Weinberger, K.Q. (2016). Densely Connected Convolutional Networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261-2269.
- [16] Tan, M., & Le, Q.V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *ArXiv*, [abs/1905.11946](https://arxiv.org/abs/1905.11946).
- [17] Wang, C., Bochkovskiy, A., & Liao, H.M. (2020). Scaled-YOLOv4: Scaling Cross Stage Partial Network. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.13024-13033.
- [18] Zhang, X., Zeng, H., Guo, S., & Zhang, L. (2022). Efficient Long-Range Attention Network for Image Super-resolution. *European Conference on Computer Vision*.
- [19] Bochkovskiy, A., Wang, C., & Liao, H.M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. *ArXiv*, [abs/2004.10934](https://arxiv.org/abs/2004.10934).

- [20] Yun, S., Han, D., Oh, S., Chun, S., Choe, J., & Yoo, Y.J. (2019). CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6022-6031.
- [21] Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path Aggregation Network for Instance Segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8759-8768.

