

# Small Object Detection

April 18, 2025

課程名稱：研究方法

授課教授：Hong-Yuan Mark Liao (廖弘源)

課程助教：Hung-Shuo Chang (張翊碩)

報告人員：

5113050001 何承祐

5113050019 黃昱凱

5113050006 洪玄杞

5113050011 陳雅玲

# OUR TEAM

- 國立中興大學
- 人工智慧資科學程
- 碩專班 1年級



*Samuel*

何承祐

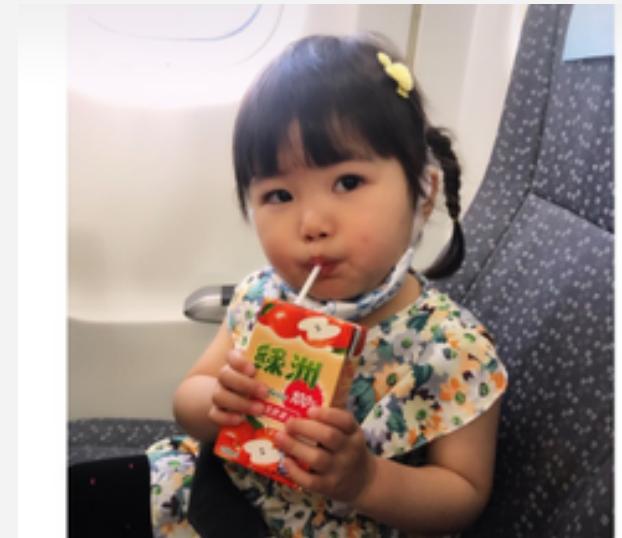
學號：5113050001



*Agnes*

洪玄杞

學號：5113050006



*Erin*

陳雅玲

學號：5113050011



*Dennis*

黃昱凱

學號：5113050019

# SPREAD THE WORKLOAD



- 網路參數調整

**Network architecture**  
• Yolov7 network Study  
• Adjust network parameters

- 網路架構調整

**Network architecture**  
• Data Exploratory  
• Yolov7 network Study  
• Paper Survey  
• Remove P4 from P2 P3  
P4 Architecture.

- 資料預處理

**Preprocessing**  
• Segment/Enhance /rotation  
• Add datasets  
• Use new network architecture

- 模型優化

**Optimization**

- method 1 - SAHI
- method 2 - Merge
- method 3 - CNN

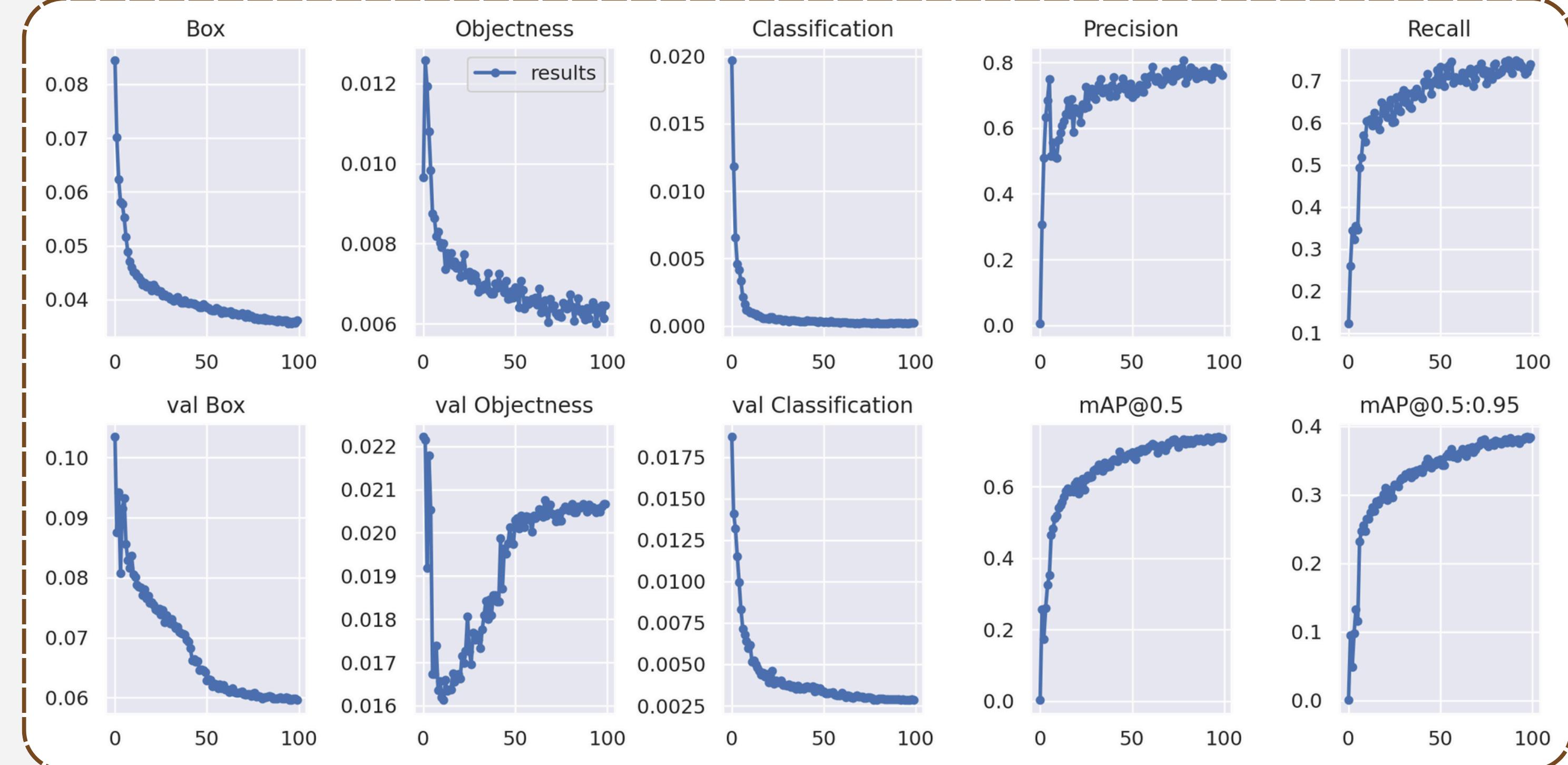
# 未做任何調整

Leaderboard

Public Score: 0.41603  
Total Vehicle: 18556

img : 640  
batch-size : 12  
workers : 8  
epoch : 100

train : 2094  
valid : 449



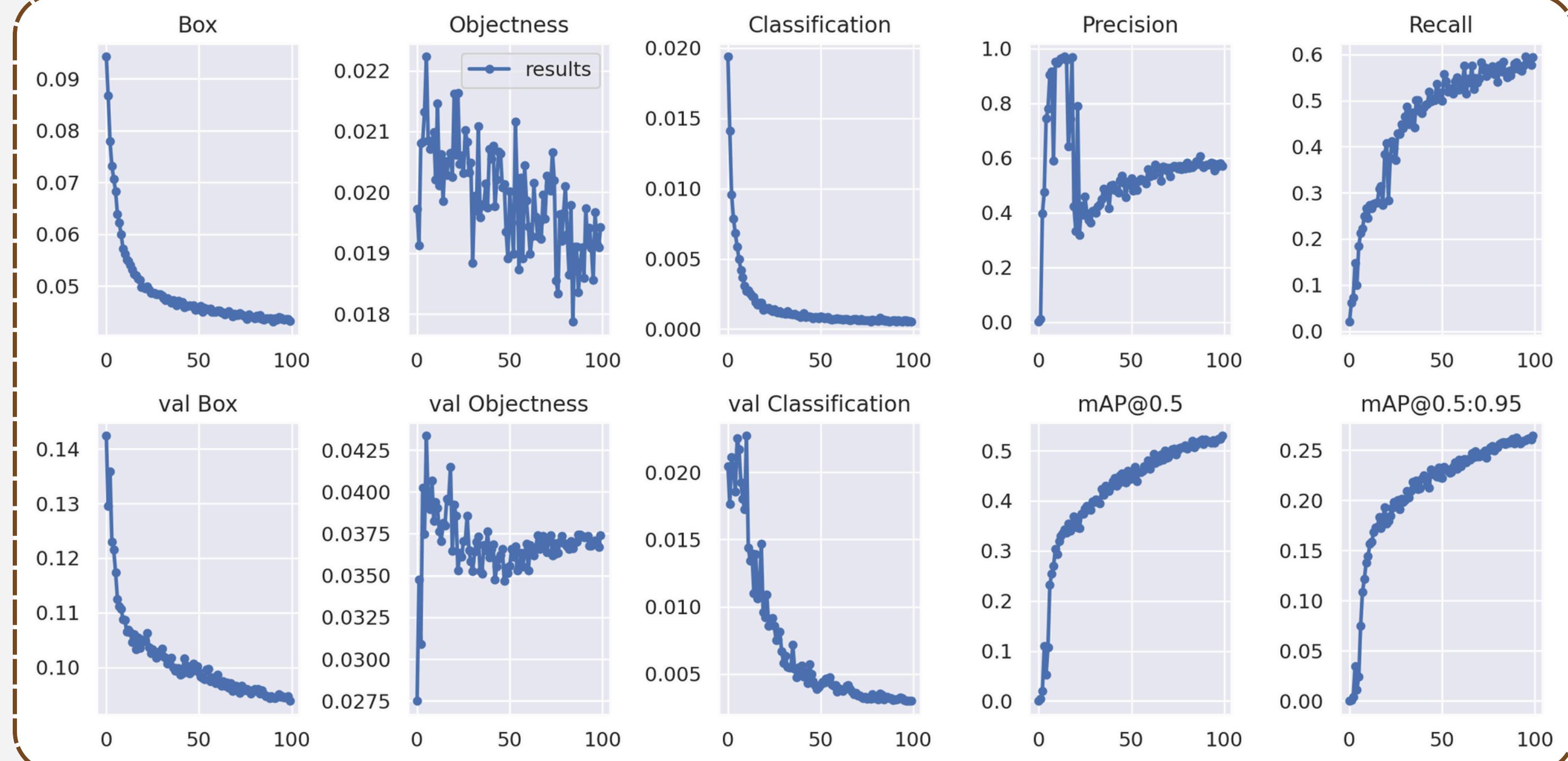
# yolov7-tiny.yaml

Leaderboard

Public Score: 0.11983  
Total Vehicle: 25100

img : 640  
batch-size : 12  
workers : 8  
epoch : 100

train : 2094  
valid : 449

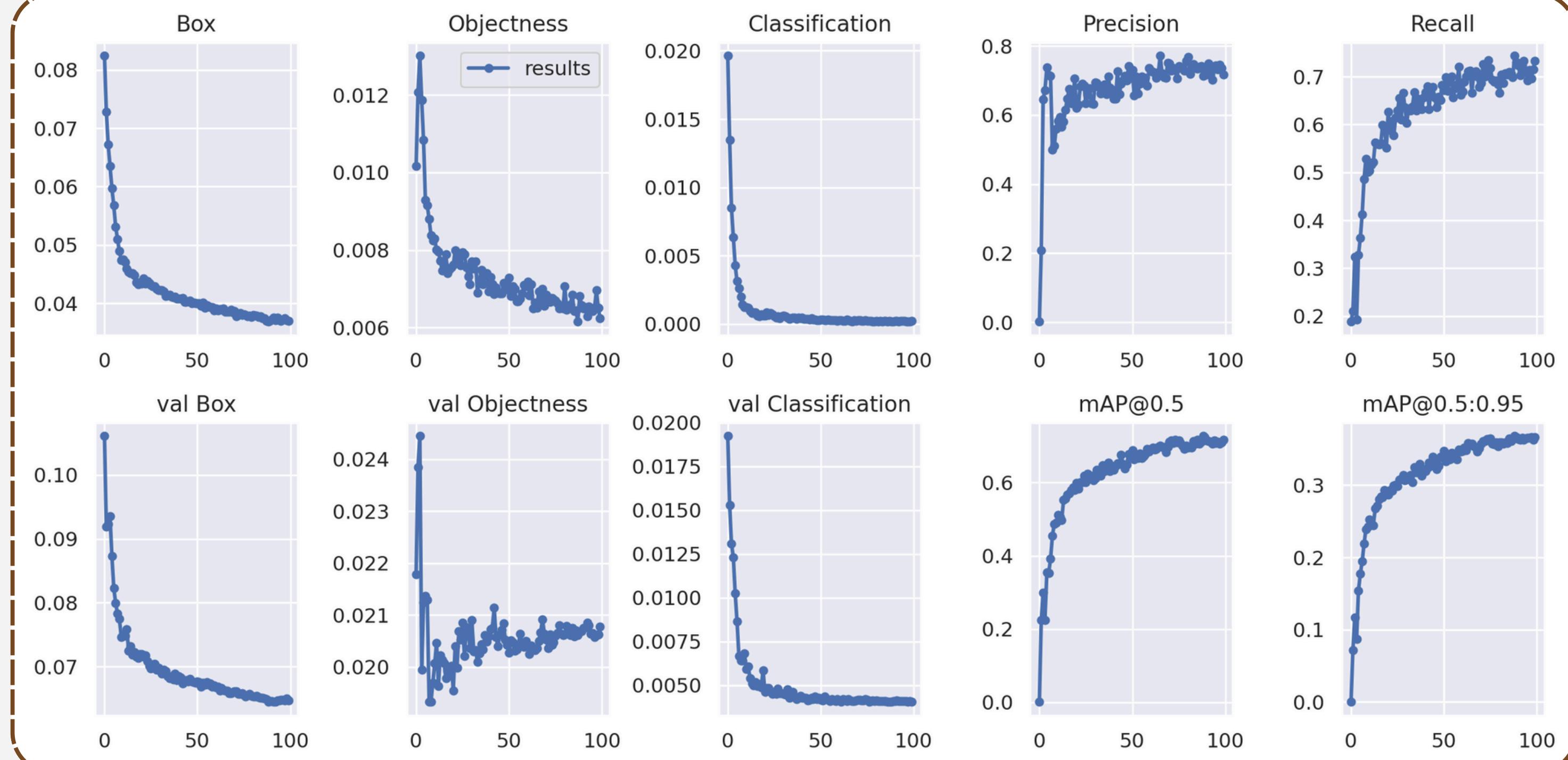


# 調整網路層參數

Leaderboard

Public Score: 0.42940  
Total Vehicle: 20527

img : 640  
batch-size : 12  
workers : 8  
epoch : 100  
  
train : 2094  
valid : 449

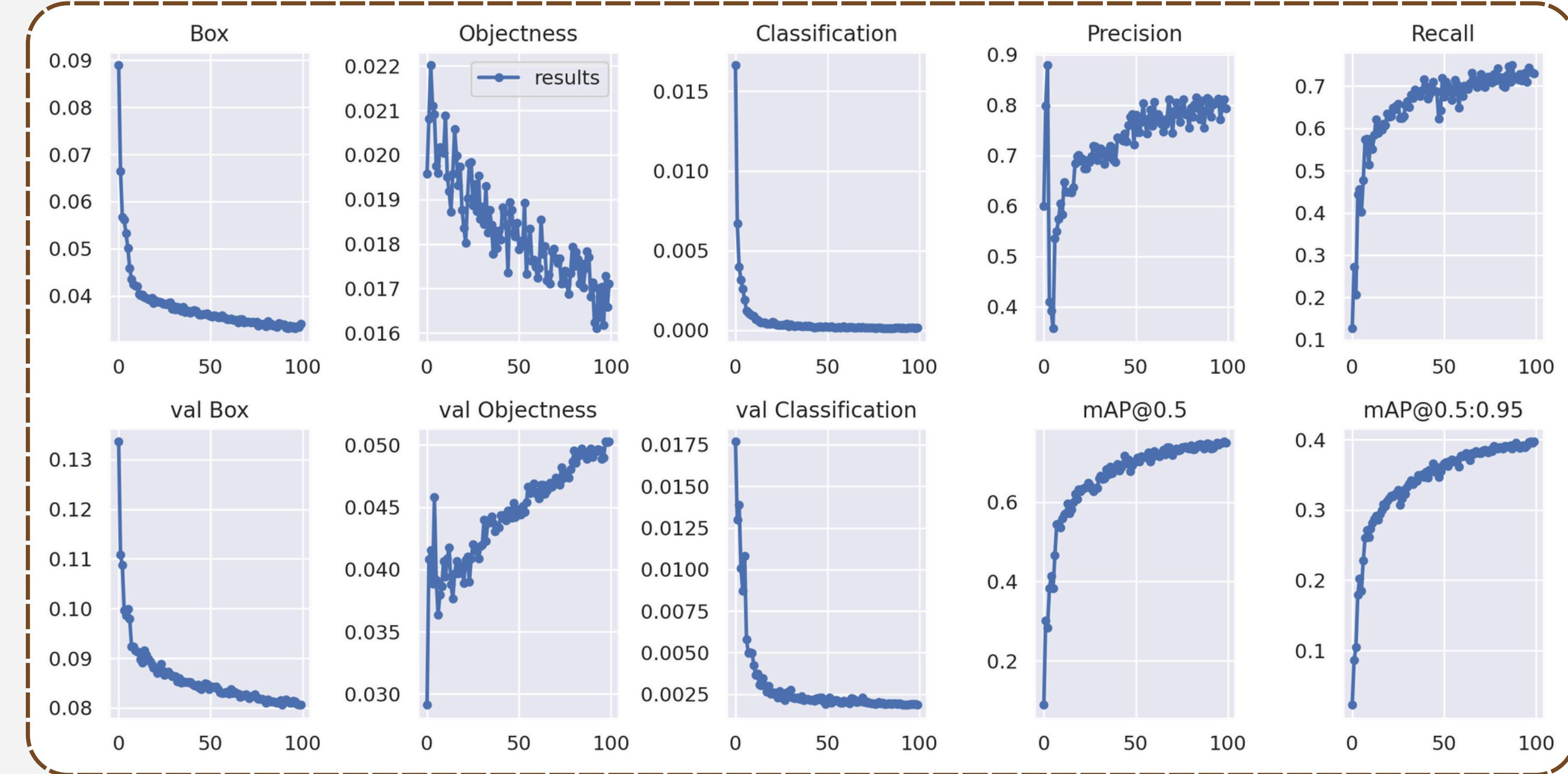


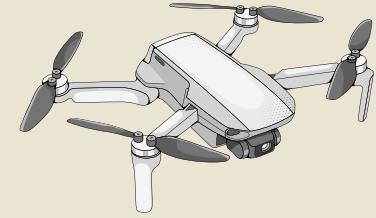
Leaderboard

Public Score: 0.44542  
Total Vehicle: 16878

# yolov7.yaml

img : 640  
batch-size : 12  
workers : 8  
epoch : 100  
  
train : 2094  
valid : 449





SkyFusion

# DATA EXPLORATORY

## FROM COCO JSON

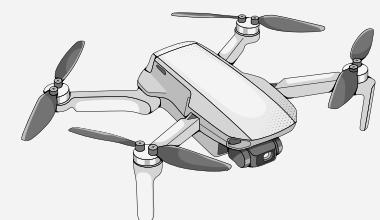
- large: area > 96x96
- medium: 32x32 < area < 96x96
- small: area < 32x32

object size	Train image sets			Test image sets			Valid image sets		
	large	medium	small	large	medium	small	large	medium	small
Aircraft	-	5,947	2,749	1	1,313	654	-	1,232	725
ship	-	56	1,427	-	15	391	-	11	277
vehicle	-	22	33,374	-	6	9,371	-	3	6,139
Summary	-	6,025 (13.8%)	37,550 (86.2%)	1	1,334 (11.4%)	10,416 (88.6%)	-	1,246 (14.9%)	7,141 (85.1%)

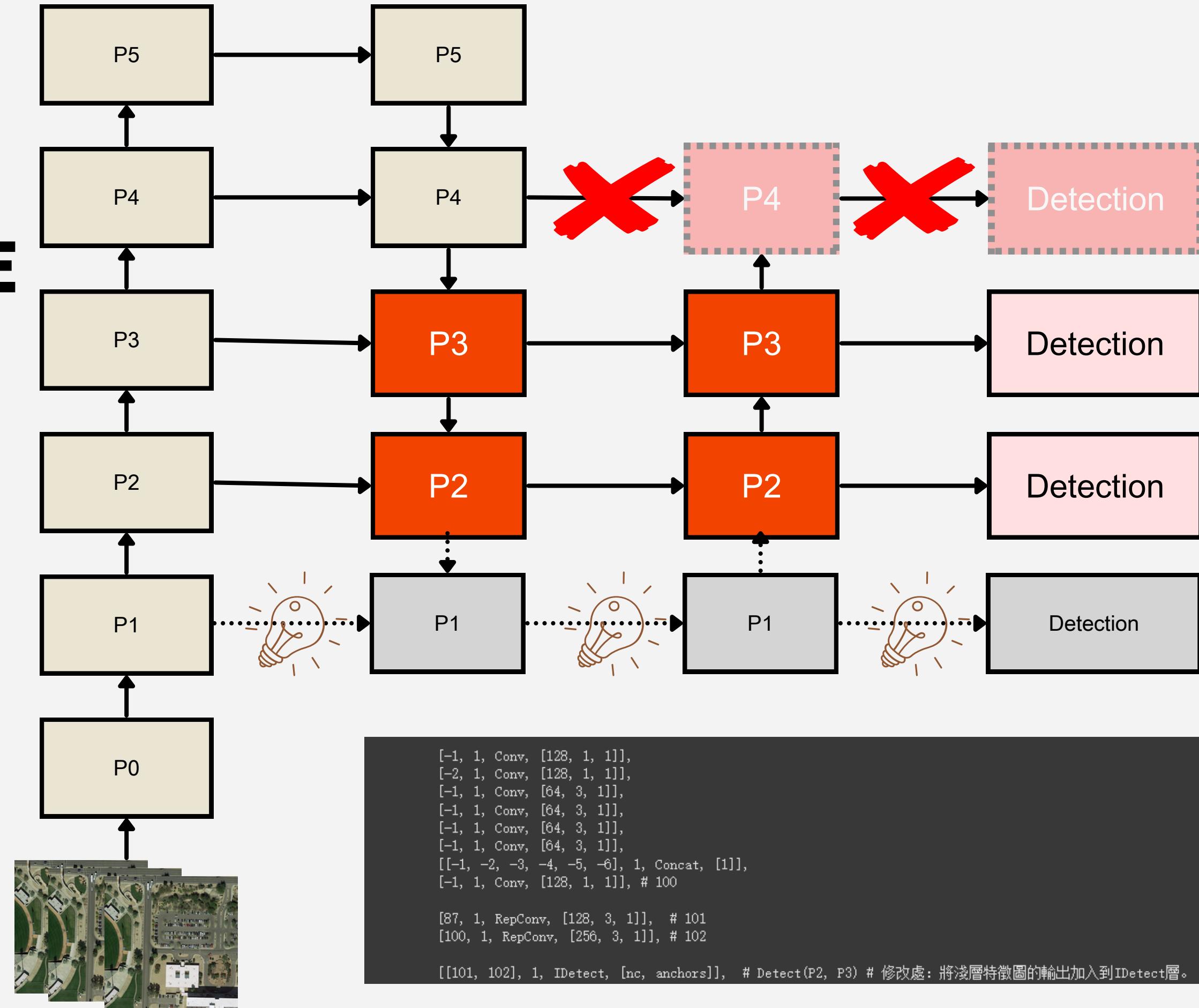


# MODIFY NETWORK ARCHITECTURE

源起：  
基於助教提供P2 P3 P4 架構  
再移除P4僅留下**P2 P3**！



# SkyFusion



# HANDS ON

01

NETWORK P2 P3 P4 --EPOCH60

```
python train.py --img 640 --batch-size 12 --workers 8 --weights 'yolov7_training.pt'  
--cfg cfg/training/yolov7_small_object.yaml  
--hyp data/hyp.scratch.p5.yaml  
--data data/skyfusion.yaml --device 0 --epoch 60
```

Epoch	gpu_mem	box	obj	cls	total	labels	img_size
59/59	14.3G	0.03723	0.00638	0.0001933	0.0438	220	640: 100% 175/175 [02:19<00:00, 1.26it/s]
Class	Images	Labels		P	R	mAP@.5	mAP@.5:95
all	449	8387		0.71	0.727	0.697	0.357
Aircraft	449	1957		0.955	0.949	0.94	0.654
ship	449	288		0.565	0.538	0.509	0.179
vehicle	449	6142		0.611	0.694	0.642	0.237

60 epochs completed in 2.570 hours.

02

WITH CBAM, MODIFY ANCHORS, ....

TRIAL RUN --EPOCH30

03

NETWORK P2 P3 --EPOCH60

Epoch	gpu_mem	box	obj	cls	total	labels	img_size
59/59	14.2G	0.03639	0.007592	0.0002324	0.04422	62	640: 100% 175/175 [02:09<00:00, 1.36it/s]
Class	Images	Labels		P	R	mAP@.5	mAP@.5:95
all	449	8387		0.743	0.74	0.724	0.366
Aircraft	449	1957		0.954	0.951	0.947	0.652
ship	449	288		0.656	0.576	0.581	0.209
vehicle	449	6142		0.619	0.692	0.645	0.237

60 epochs completed in 2.377 hours.

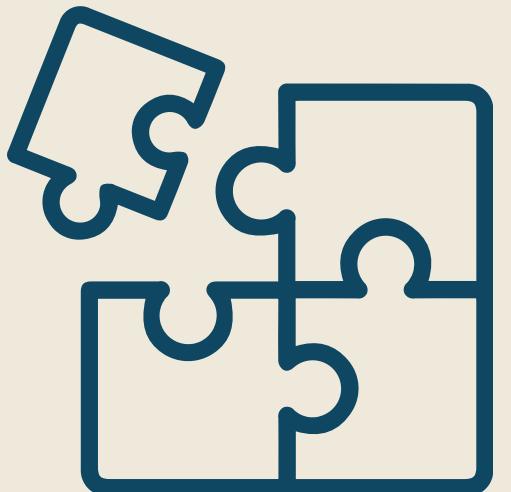
# GET HANDS DIRTY

## HANDS-ON & RESEARCH



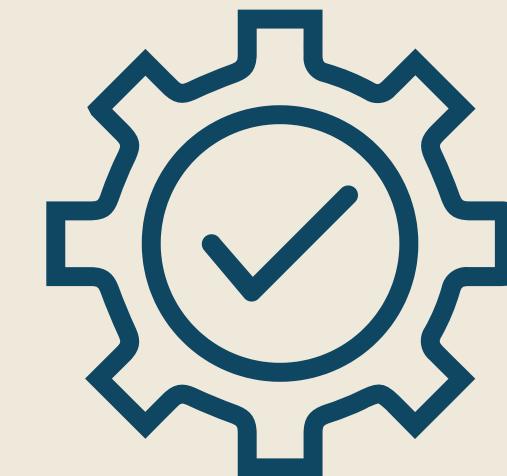
- Colab - install Yolov7 issue
- Network Study
- Survey (sliding window)
- image split with overlap

## DO IT MYSELF



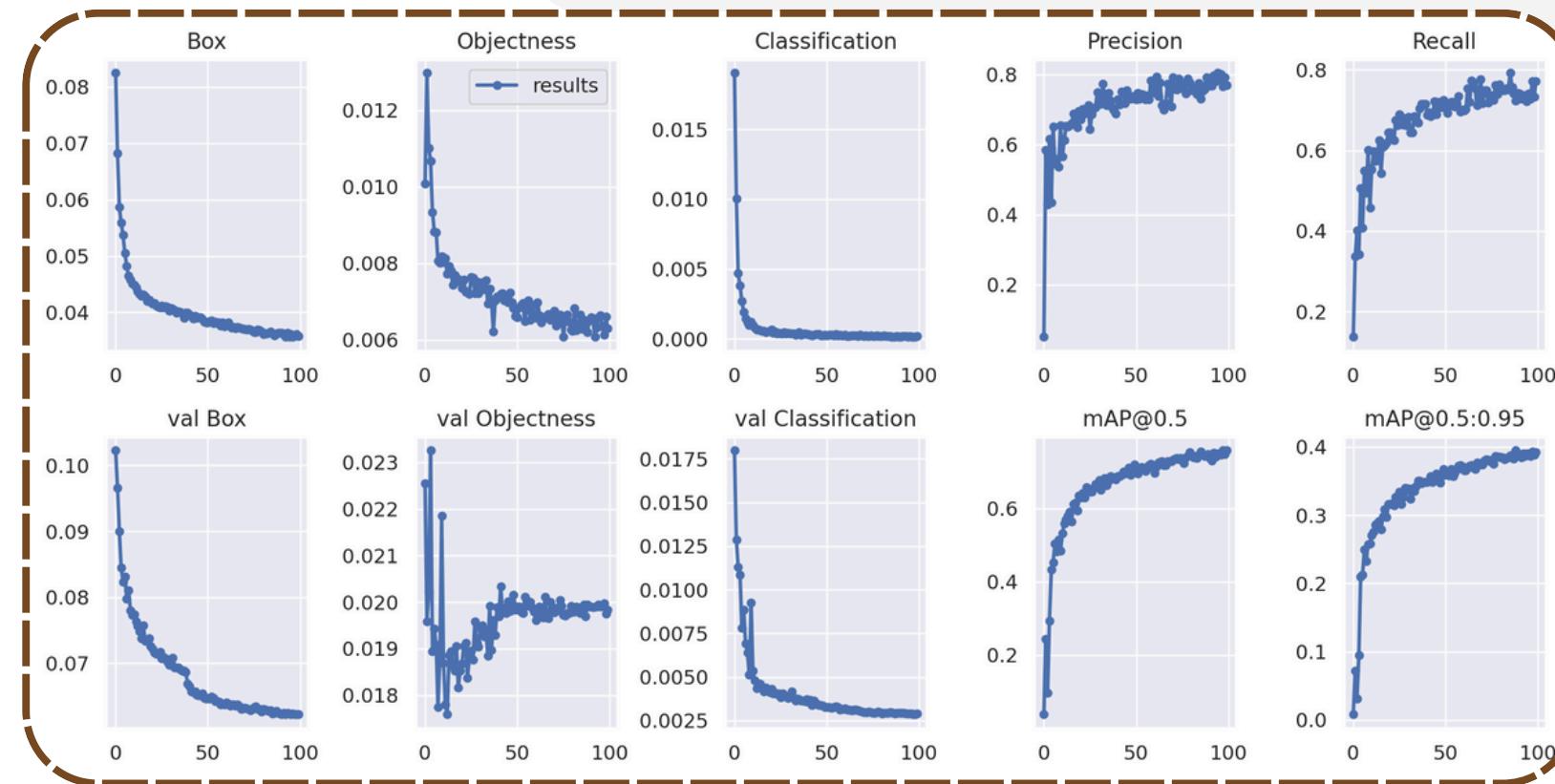
- CBAM in network
- Modify anchors
- Network with P2 & P3 Architecture.
- Try it repeatedly

## NOT ONLY A TEAM WE ARE A TEAM



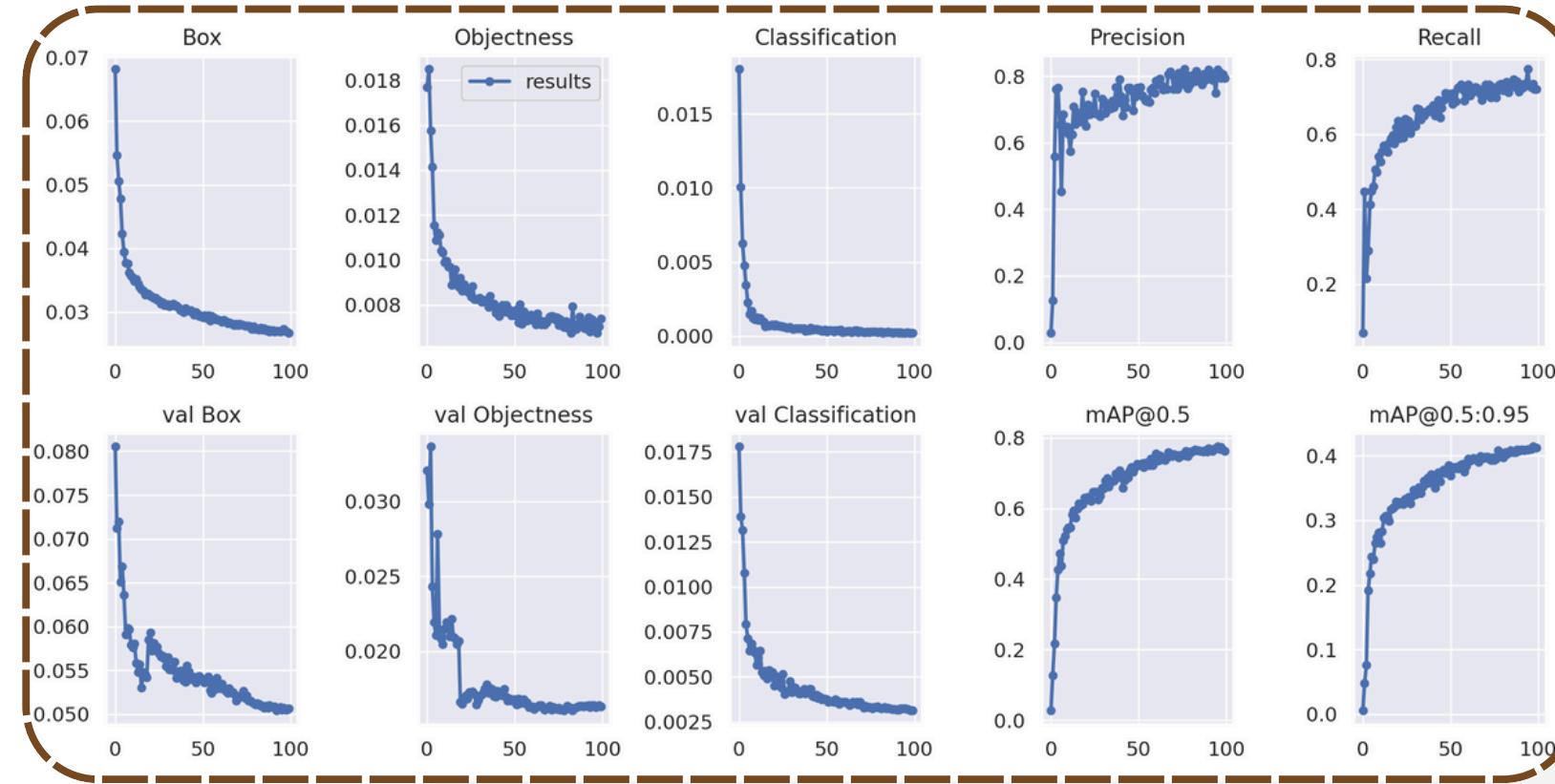
- Awesome Dennis
- Efficient Erin
- Hard-working Agnes
- Inspired Samuel

# Different image sizes



img 640

img : 640	batch-size : 12	train : 2543	all : 0.758
	workers : 8	valid : 449	Aircraft : 0.946
	epoch : 100		ship : 0.623
			vehicle : 0.707

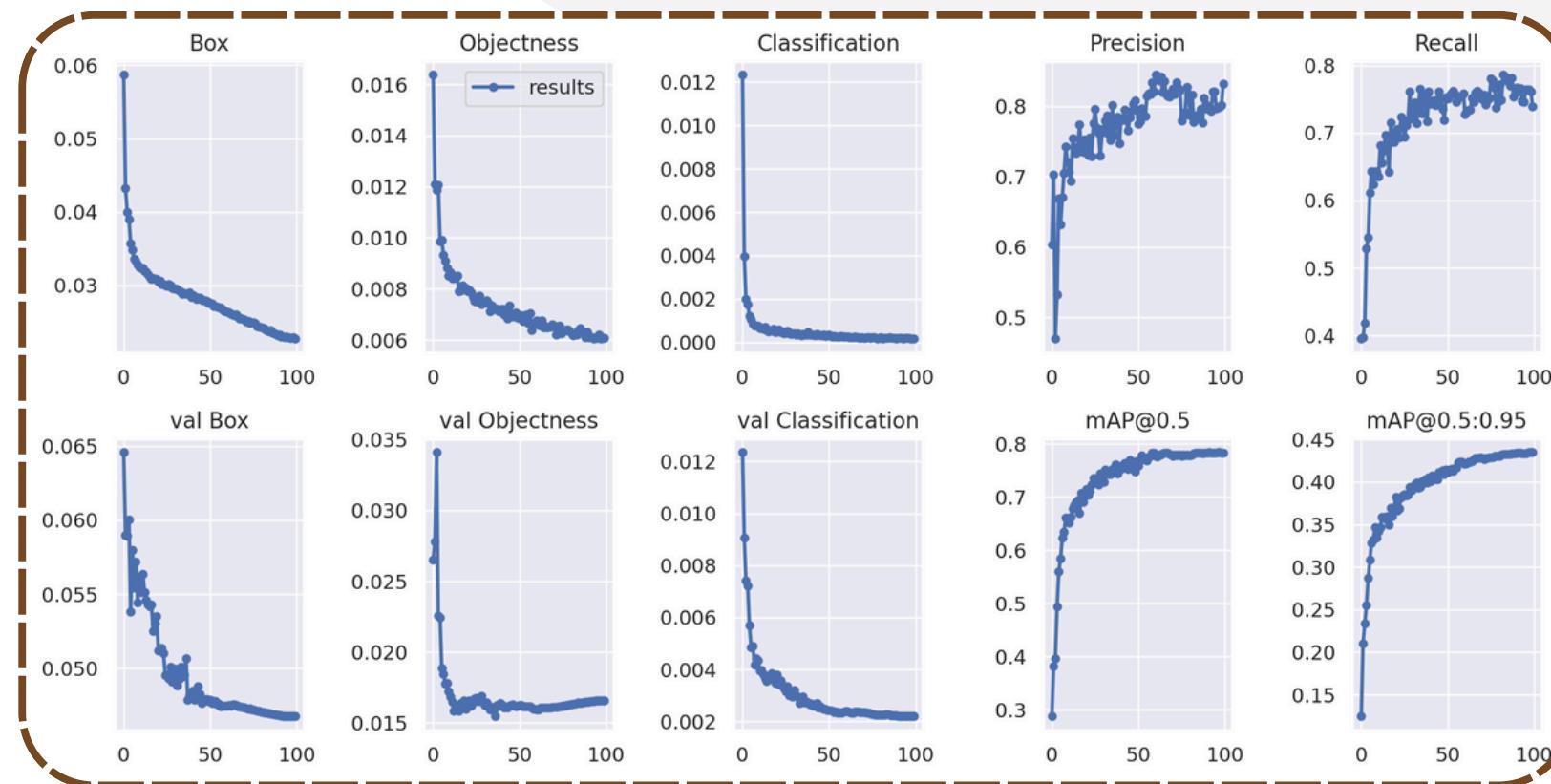


img 1280

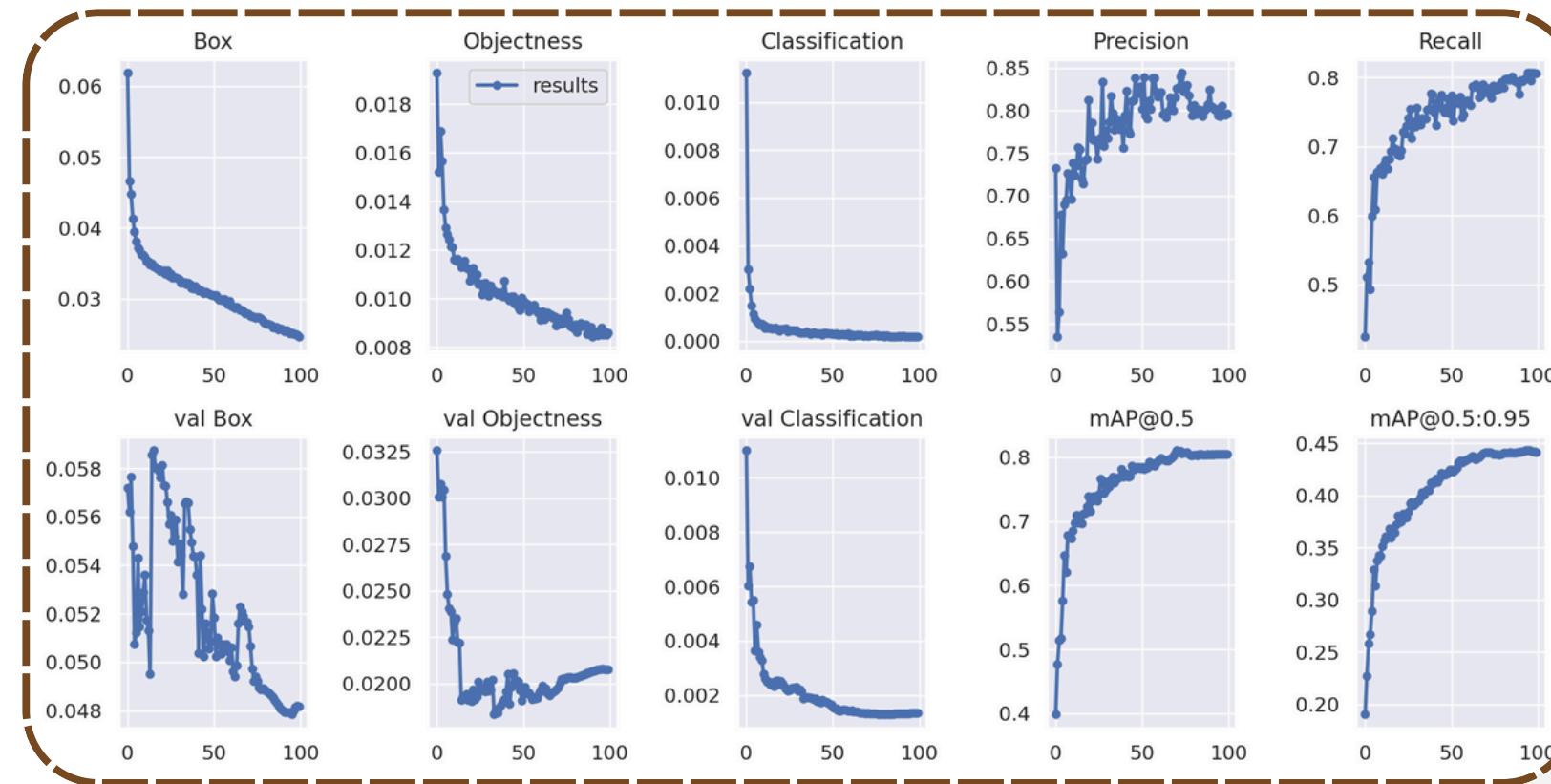
img : 1280	batch-size : 4	train : 2543	all : 0.763
	workers : 4	valid : 449	Aircraft : 0.945
	epoch : 100		ship : 0.582
			vehicle : 0.761



# Modify Network



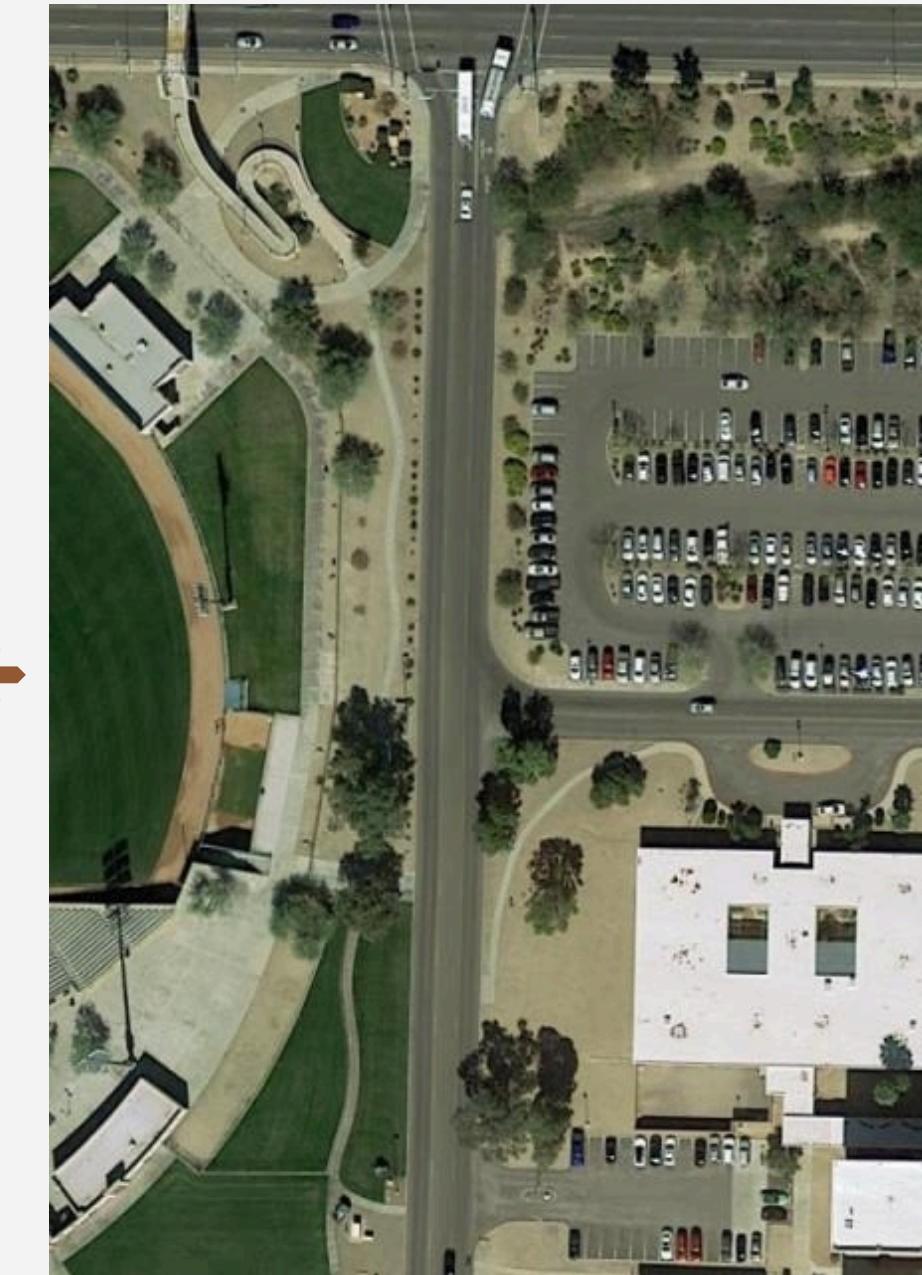
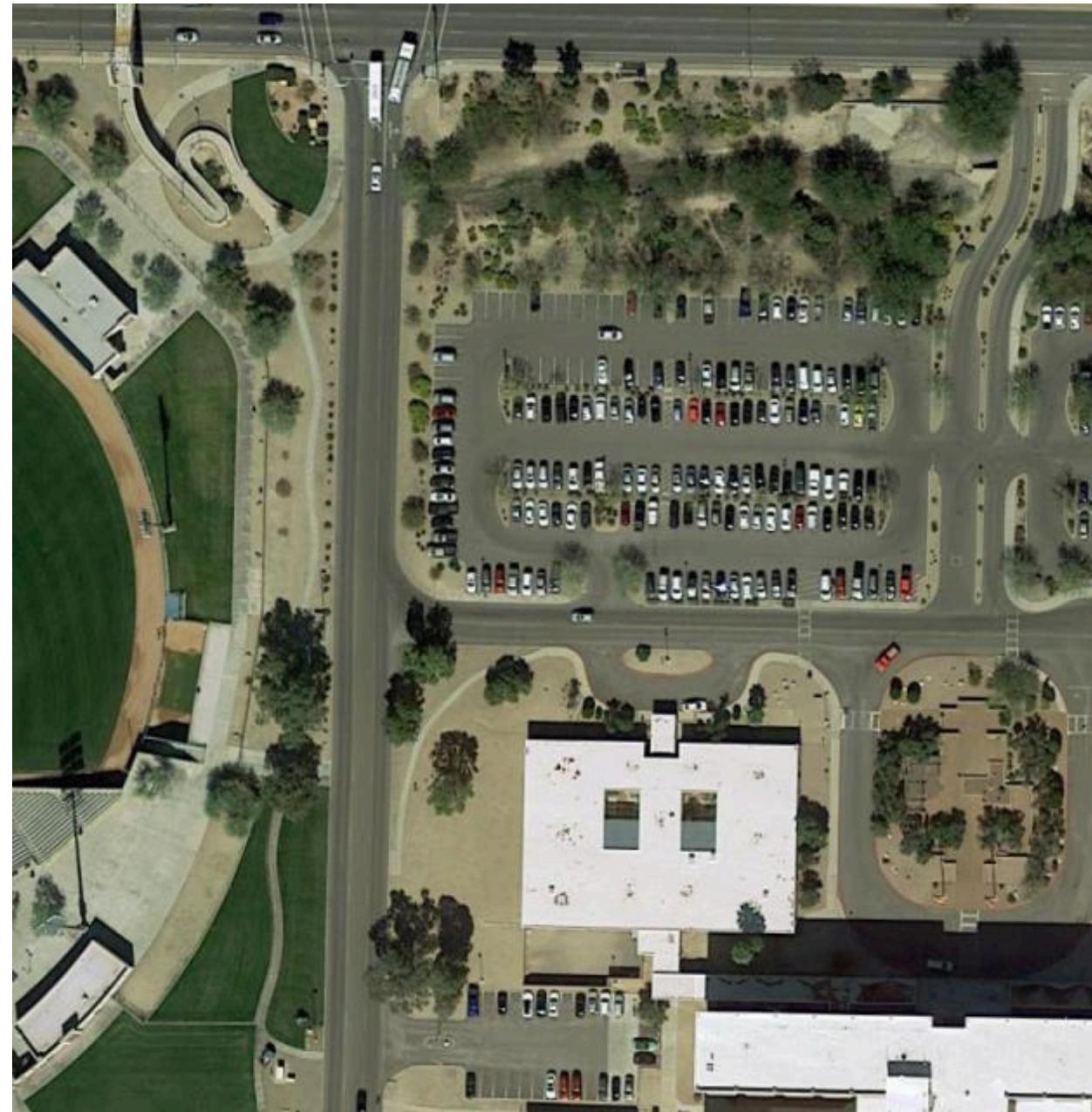
網路層：P2、P3、P4



網路層：P2、P3



# Preprocessing

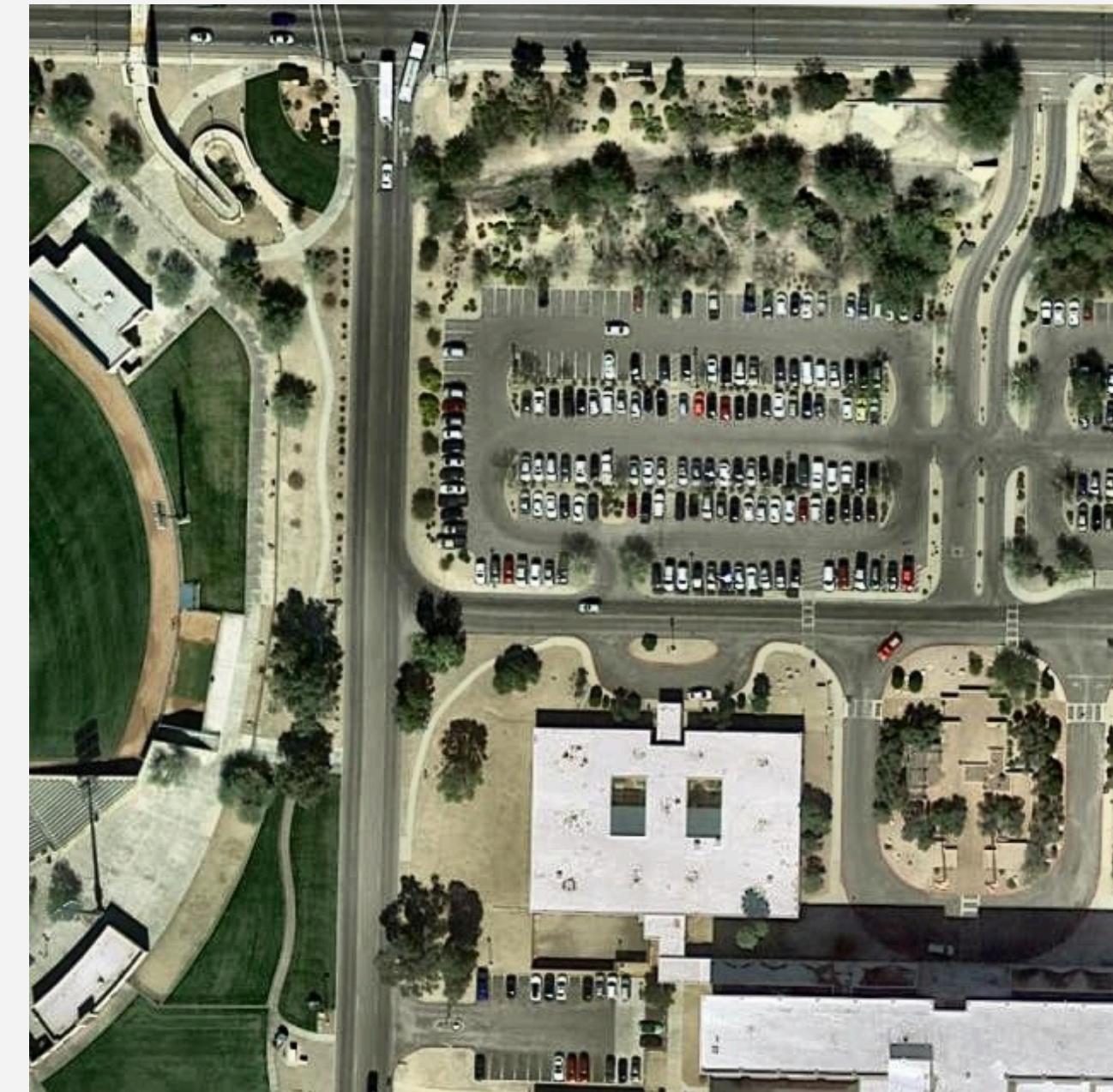


Segment the image  
Training image count : 3,173 張

# Preprocessing

Enhance color  
contrast

Training image count : 3,141 張



# Preprocessing

Histogram  
Equalization

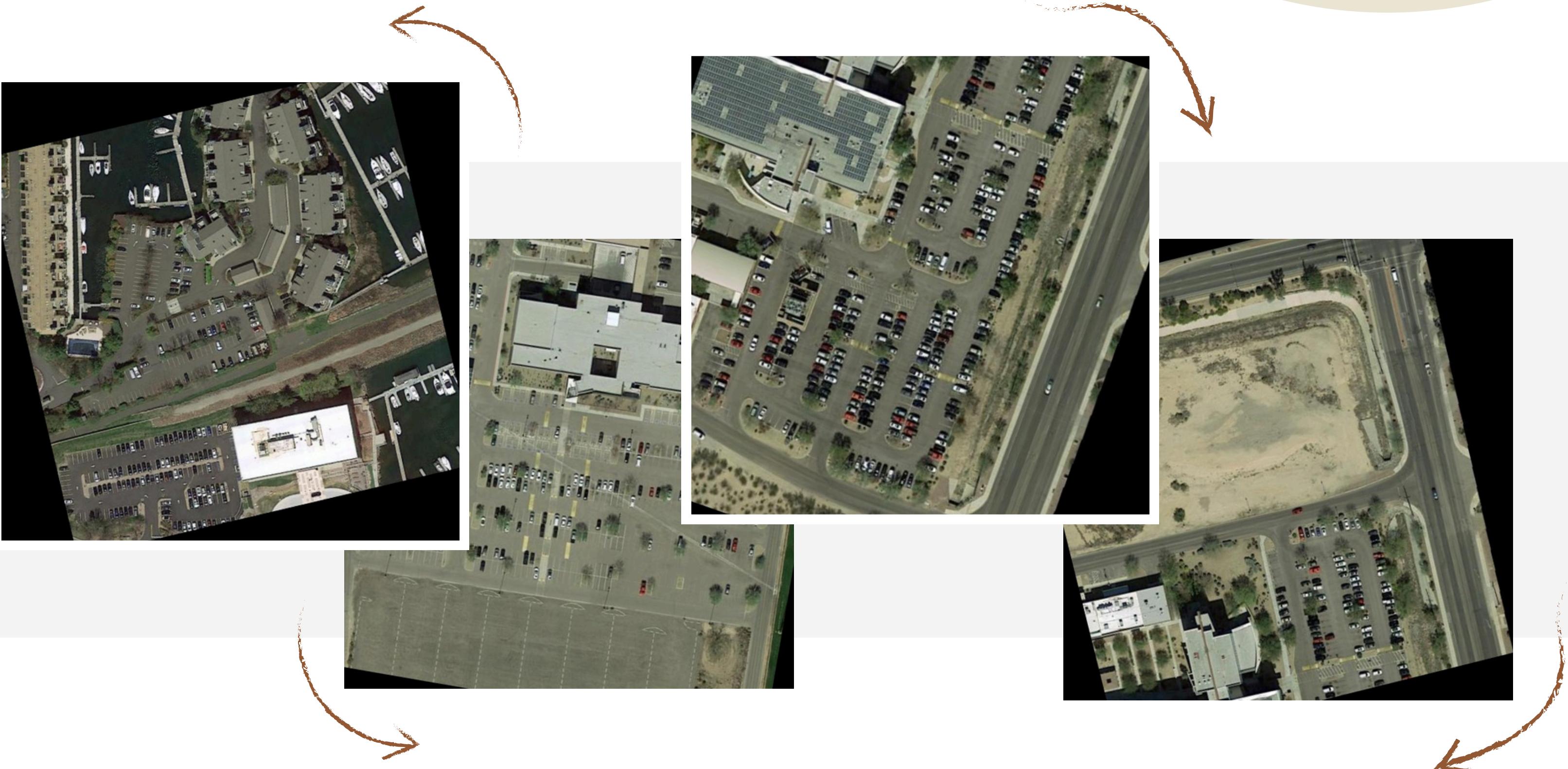
Training image count : 2,094 張



# Preprocessing

Image rotation

Training image count : 2,094 張

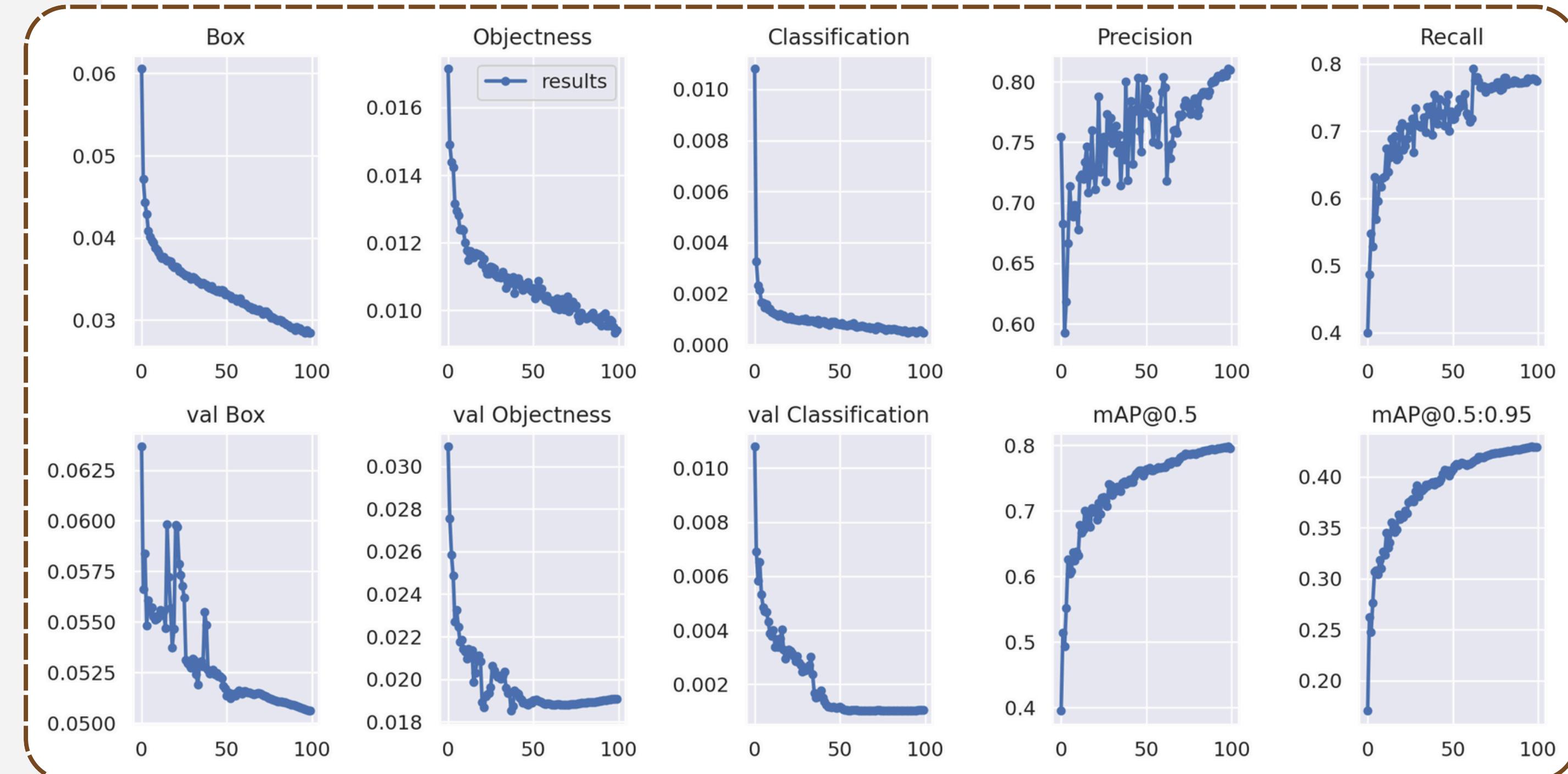


# 增加前處理資料集數量 調整網路層(只留P2、 P3)

Leaderboard

Public Score: 0.64922  
Total Vehicle: 16306

img : 1280  
batch-size : 4  
workers : 8  
epoch : 100  
  
train : 13045  
valid : 449



100 epochs completed in 30.634 hours.

# xView dataset

大規模的機載和衛星影像資料集，專門用於物件偵測任務

01

## 涵蓋範圍廣

資料集中的影像來自世界各地，包括城市、郊區、農村和海洋環境

02

## 高解析度影像

影像的解析度通常很高，允許模型學習到細緻的物件特徵

03

## 多樣性

涵蓋了各種不同的地理環境、天氣條件、光照條件和影像解析度

04

## 豐富的物件種類

涵蓋了 60 個不同的物件類別，例如建築物、飛機、車輛、船隻、橋樑、儲油罐等等



### 車輛 (vehicle)

Passenger vehicle - 客用車輛 / 轎車  
Small car - 小型汽車  
Bus - 公車 / 巴士  
Truck - 卡車 / 貨車  
Cargo truck - 貨運卡車  
Tractor - 拖拉機 / 農用車  
Trailer - 拖車 / 掛車  
Semi-trailer truck - 半拖車卡車 / 聯結車

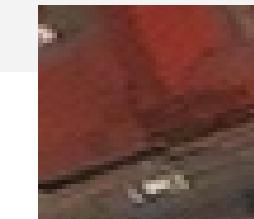
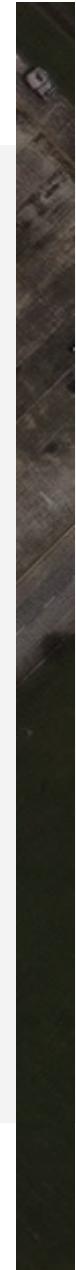
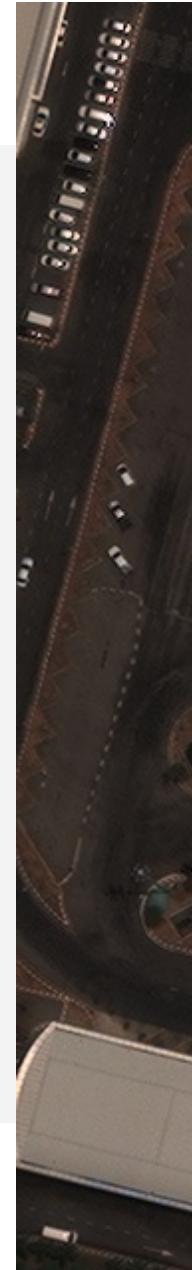
### 船 (ship)

Passenger ship - 客輪 / 遊輪  
Cargo ship - 貨輪 / 貨船  
Fishing vessel - 漁船  
Sailboat - 帆船  
Yacht - 遊艇  
Ferry - 渡輪  
Barge - 駁船

# Preprocessing

xView

Training image count : 9,283 張



# Preprocessing

將原圖和所有經過處理的圖像  
全部做為 train data

train : 22,326 張  
valid : 1,480 張



img 1280 : 100 epochs completed  
in 53.329 hours.

01 SkyFusion-train、test

原始資料集

02 Segment the image

圖片切割

03 Enhance color contrast

提升顏色對比度

04 Histogram Equalization

直方圖等化

05 Image rotation

影像旋轉

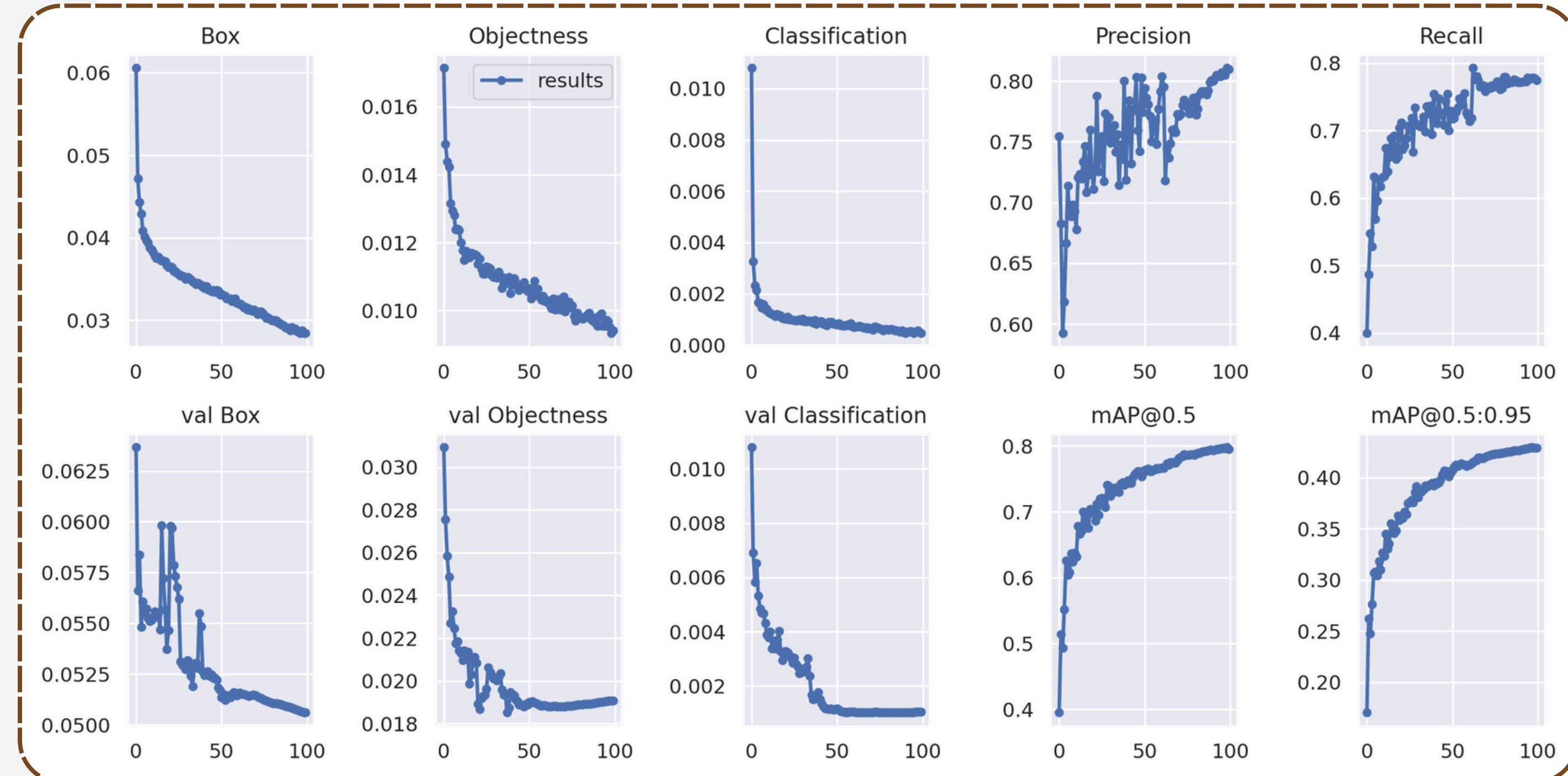
06 xView

# 再增加資料集數量(加 xView) 調整網路層(只留P2、 P3)

Leaderboard

Public Score: 0.821  
Total Vehicle: 21069

img : 1280  
batch-size : 4  
workers : 8  
epoch : 100  
  
train : 22326  
valid : 1480



100 epochs completed in 53.329 hours.

# INITIAL RESULT



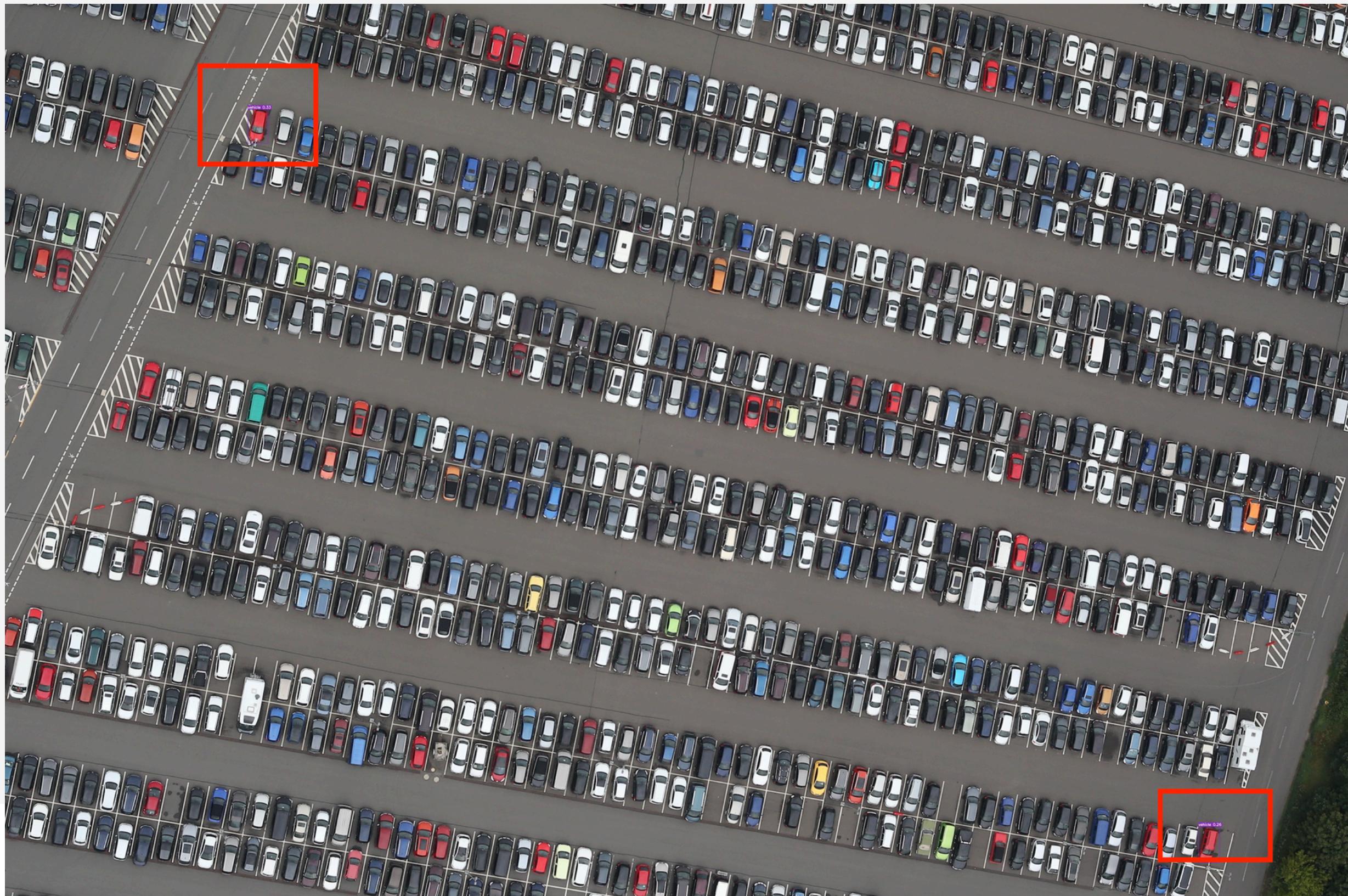
在模型信心水準 0.25

53 張測試集

Total vehicles : 21069

Public\_score : 0.821

# INITIAL RESULT

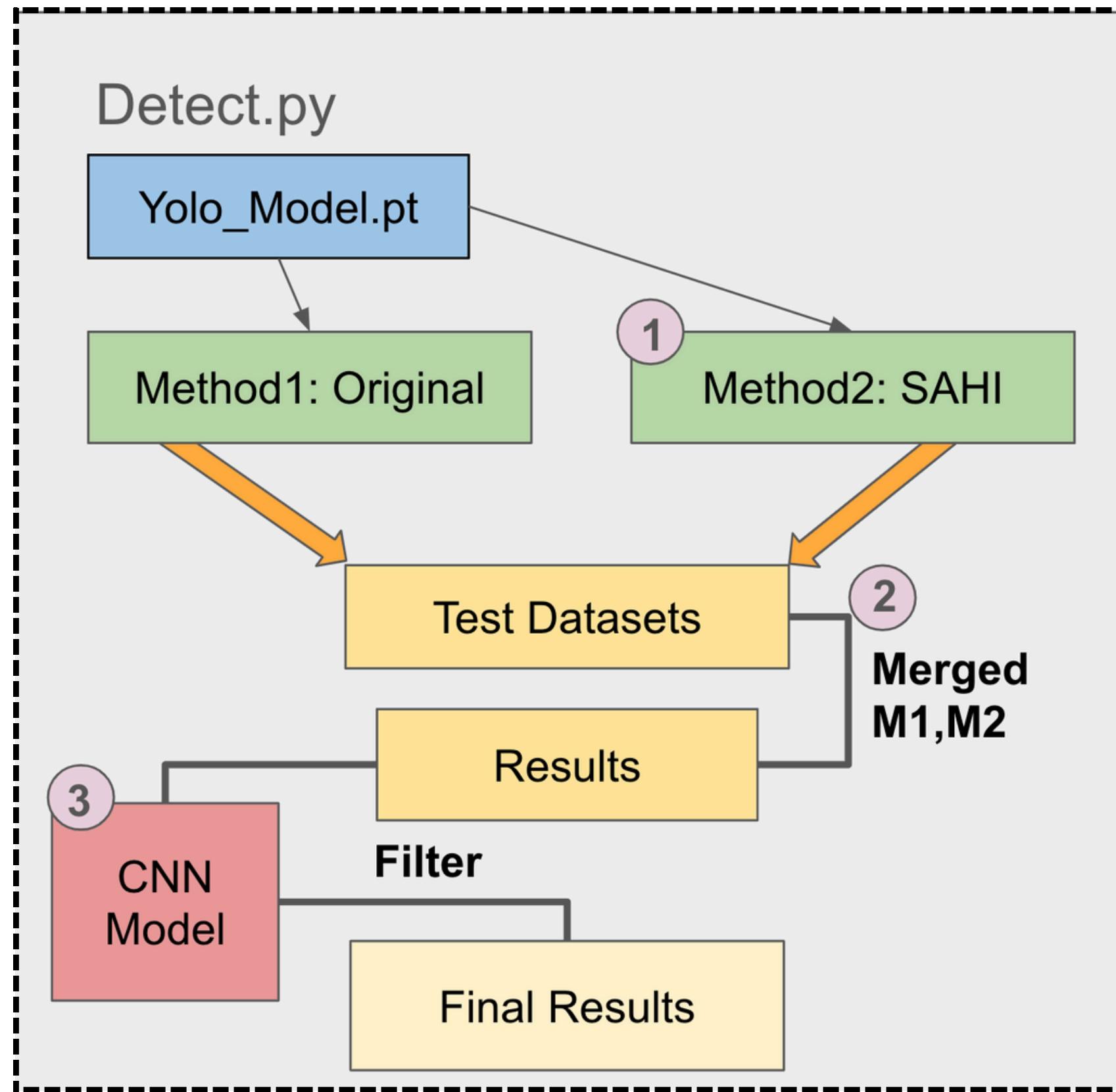


從偵測結果去檢查每張照片  
發現如果汽車很密集  
(例如：停車場)  
**漏檢非常多**

舉例：2016-08-02-wolfsburg-links-L0169.jpg(右圖)

Total Vehicles: 2

# Optimization structure



01

導入**SAHI** 的方法，針對漏檢進行優化。

02

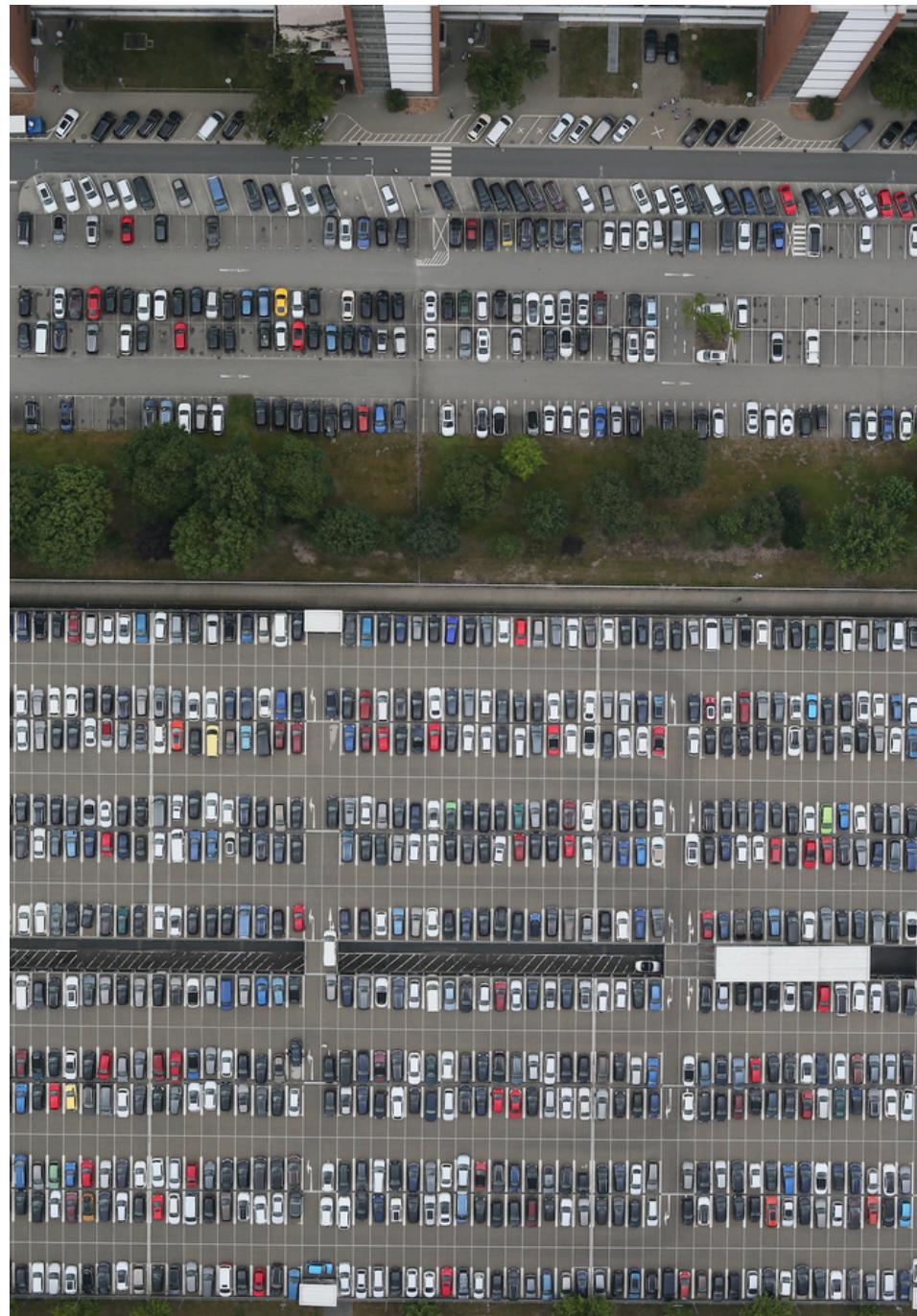
將原始方法與SAHI 方法進行合併。

03

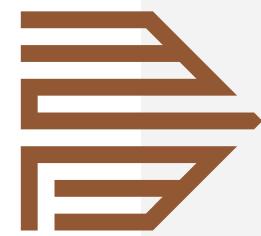
額外訓練**CNN** 模型，提高準確性。

# Optimization- method 1

## SAHI

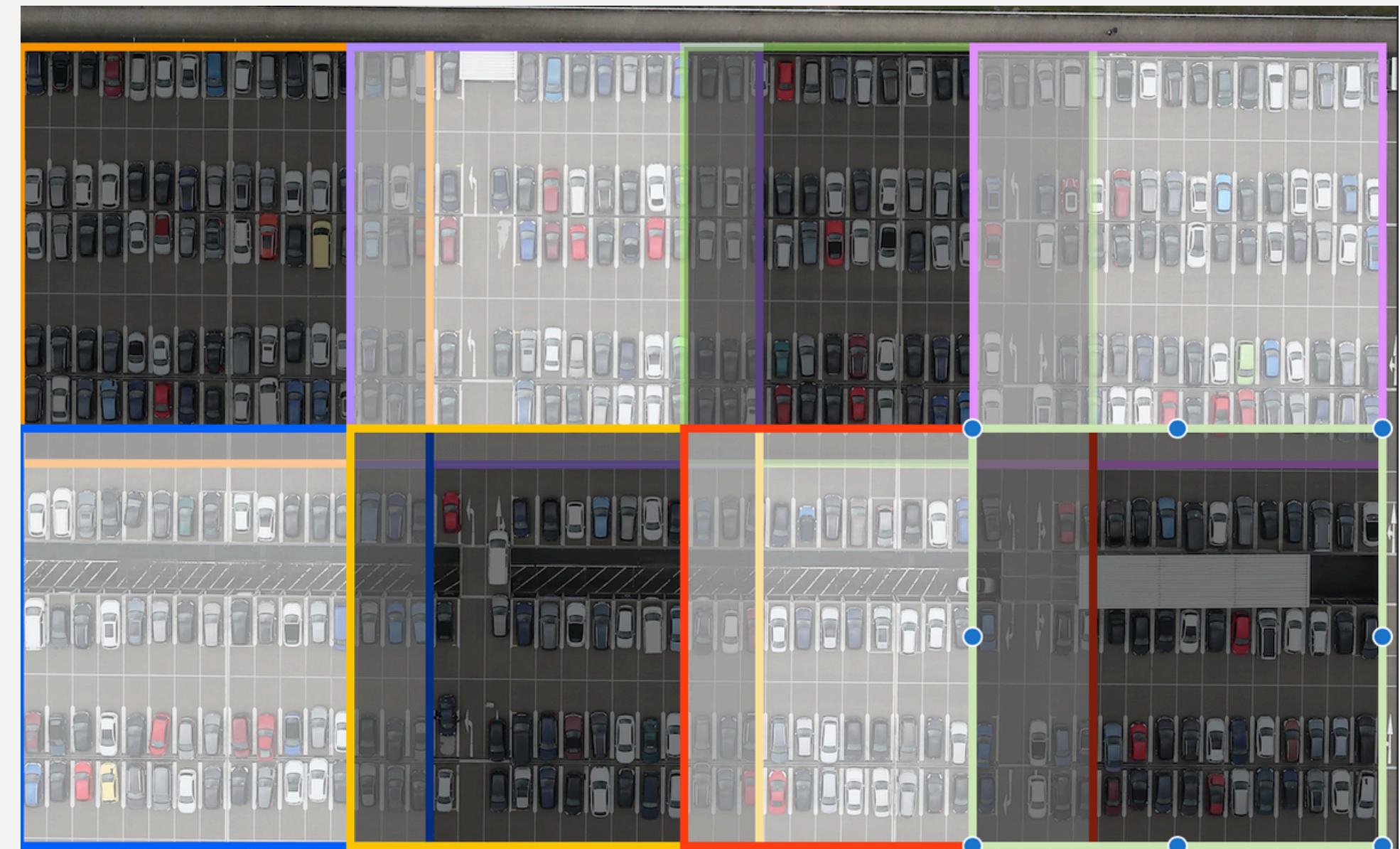


原尺寸 : 5184 x 3456



SAHI (Slicing Aided Hyper Inference)

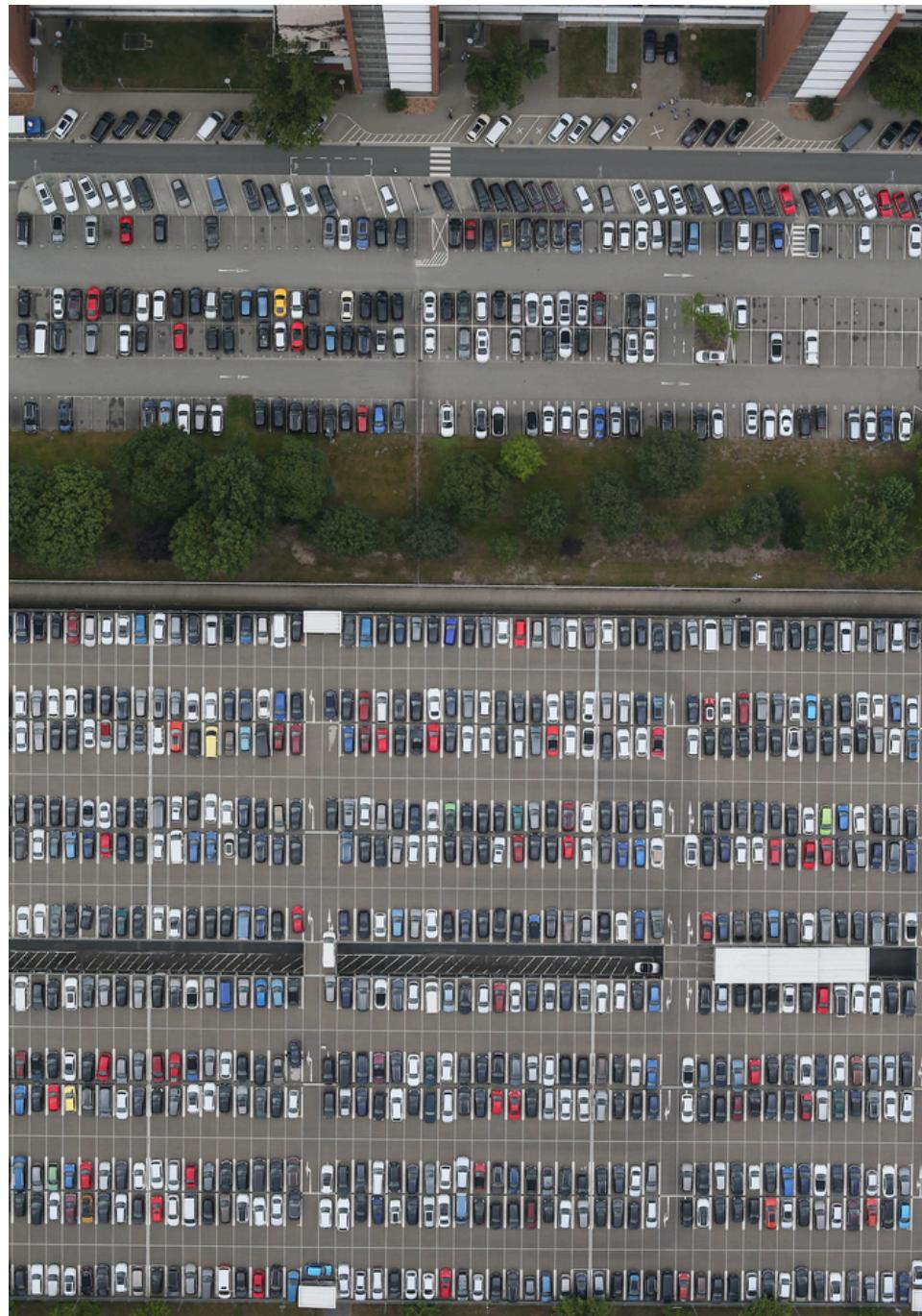
- 主要利用切割照片(示意圖如下)加強對小物件的偵測
- 基於滑窗(Sliding Windows) 理論由設定切割的尺寸對整張圖進行掃描
- 最後合併成結果



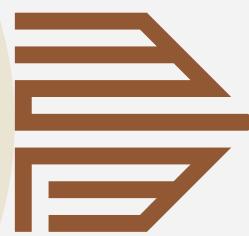
# Optimization- method 1

## SAHI

切割邏輯流程



輸入切割前  
先 Resize 成  
1280 x 1280  
(與訓練模型設定一樣)



進行切割  
(找最佳解)  
(ex: 640\* 640  
256\* 256)



原尺寸 : 5184 x 3456

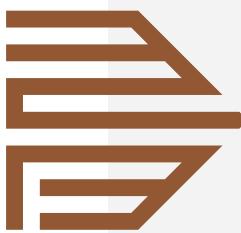
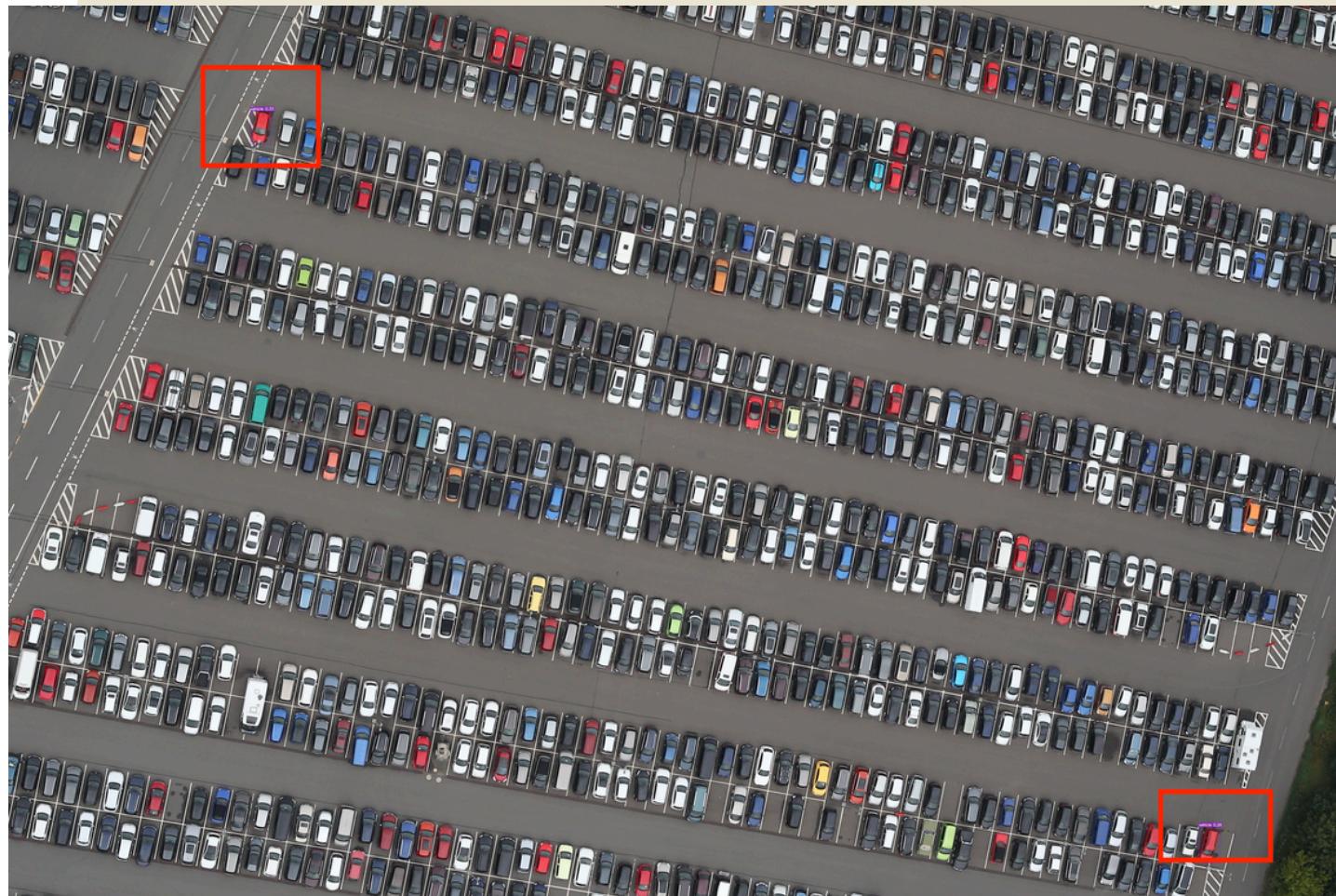
# Optimization-method 1

## SAHI

優化前

舉例：2016-08-02-wolfsburg-links-L0169.jpg

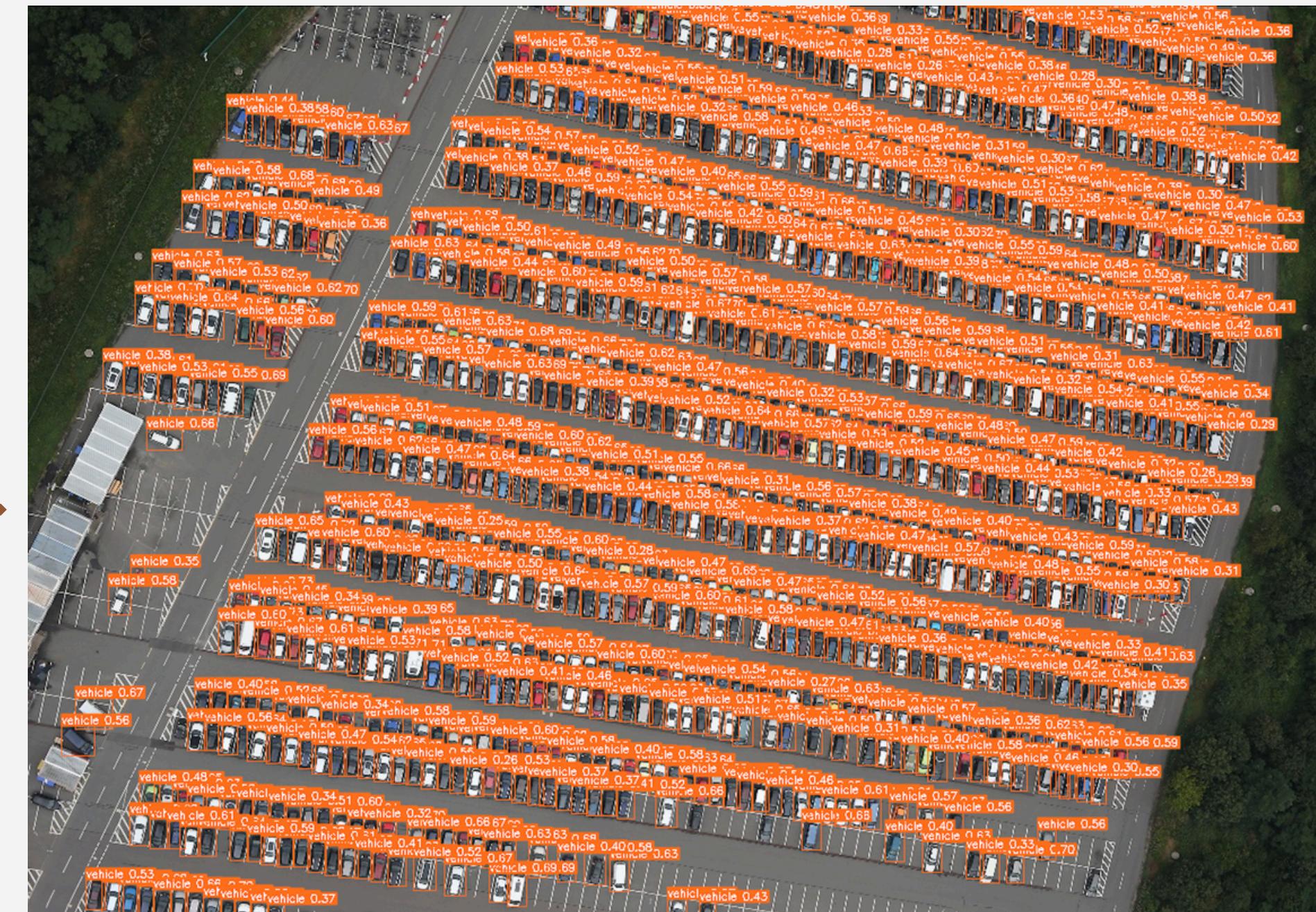
Total Vehicle: 2



優化後

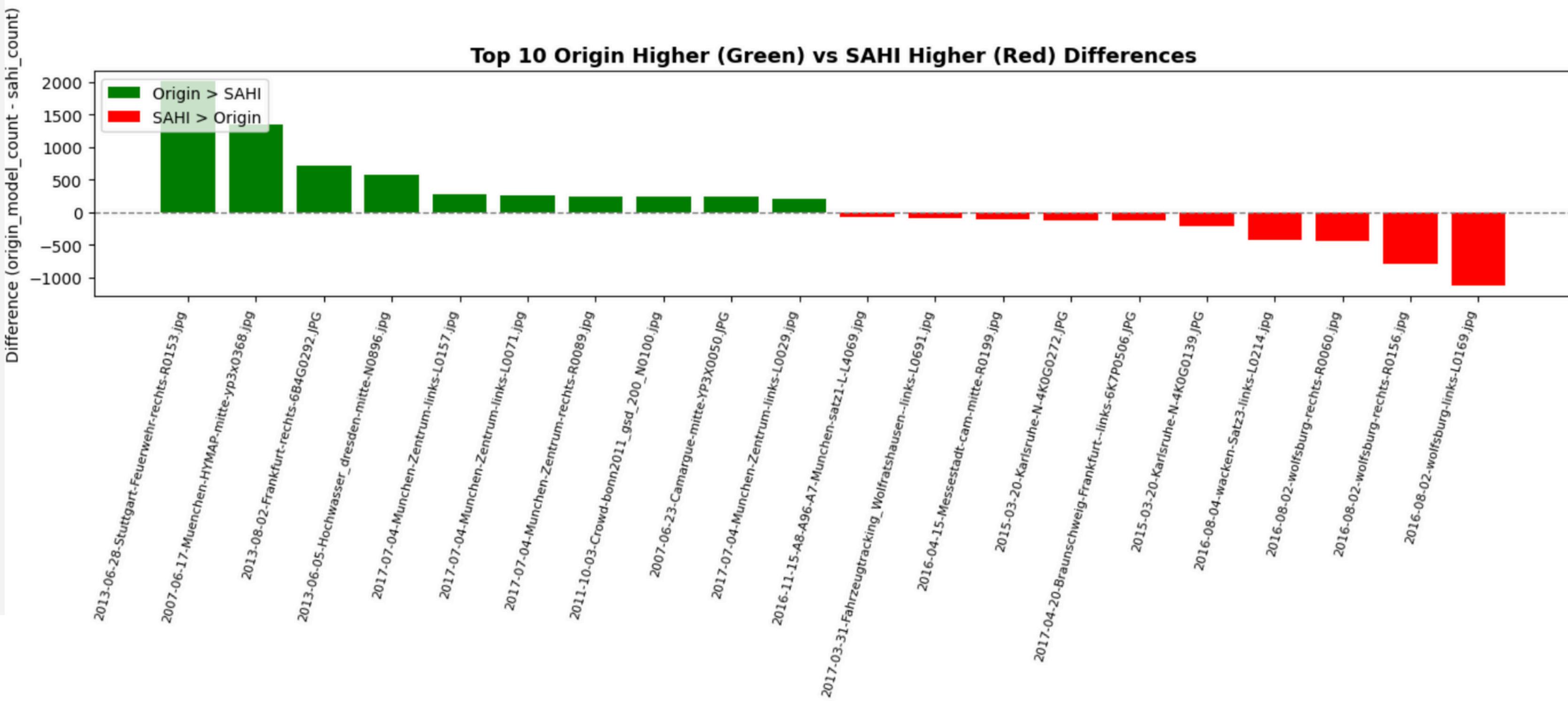
舉例：2016-08-02-wolfsburg-links-L0169.jpg

Total Vehicles: 1135



# Optimization- method 2

## Merge



# Optimization- method 2

## Merge

原本預期只用切割方法當作最後結果  
由下圖Dataframe可看出切割方法，對部份的圖表現更好、但部分圖相對原本方法表現差  
衍生出要將兩模型合併

	name	origin_model_count	sahi_count	merge_count	diff
0	2006-05-03-Allianz-links-yr7e0006.jpg	145	149	144	-4
1	2007-06-17-Muenchen-HYMAP-mitte-yp3x0368.jpg	1446	98	1450	1348
2	2007-06-23-Camargue-mitte-YP3X0050.JPG	259	24	260	235
3	2008-11-27-Muenchen-Verkehr-mitte-yp3x0014.jpg	77	66	81	11
4	2011-07-07-Tollwood--mitte-5V4B0094.JPG	745	699	779	46
5	2011-10-03-Crowd-bonn2011_gsd_200_N0100.jpg	370	129	379	241
6	2012-04-26-Muenchen-Tunnel_4K0G0051.JPG	236	227	255	9
7	2012-04-26-Muenchen-Tunnel_4K0G0130.JPG	636	545	694	91
8	2012-05-19-Champions-League-Final-rechts-R0025...	15	17	17	-2
9	2012-05-19-Champions-League-Final-rechts-R0202...	72	91	94	-19

# Optimization- method 2

## Merge

原本方法表現比較好的範例

初步組內討論或許是切割方法對分散的物件偵測能力較弱(對比停車場)



# Optimization- method 2

## Merge



利用兩種方法的結果進行合併  
(中間並利用iou去重複標籤)



# Optimization-2 Results

## Merge Method1 + Method2



在模型信心水準 0.25

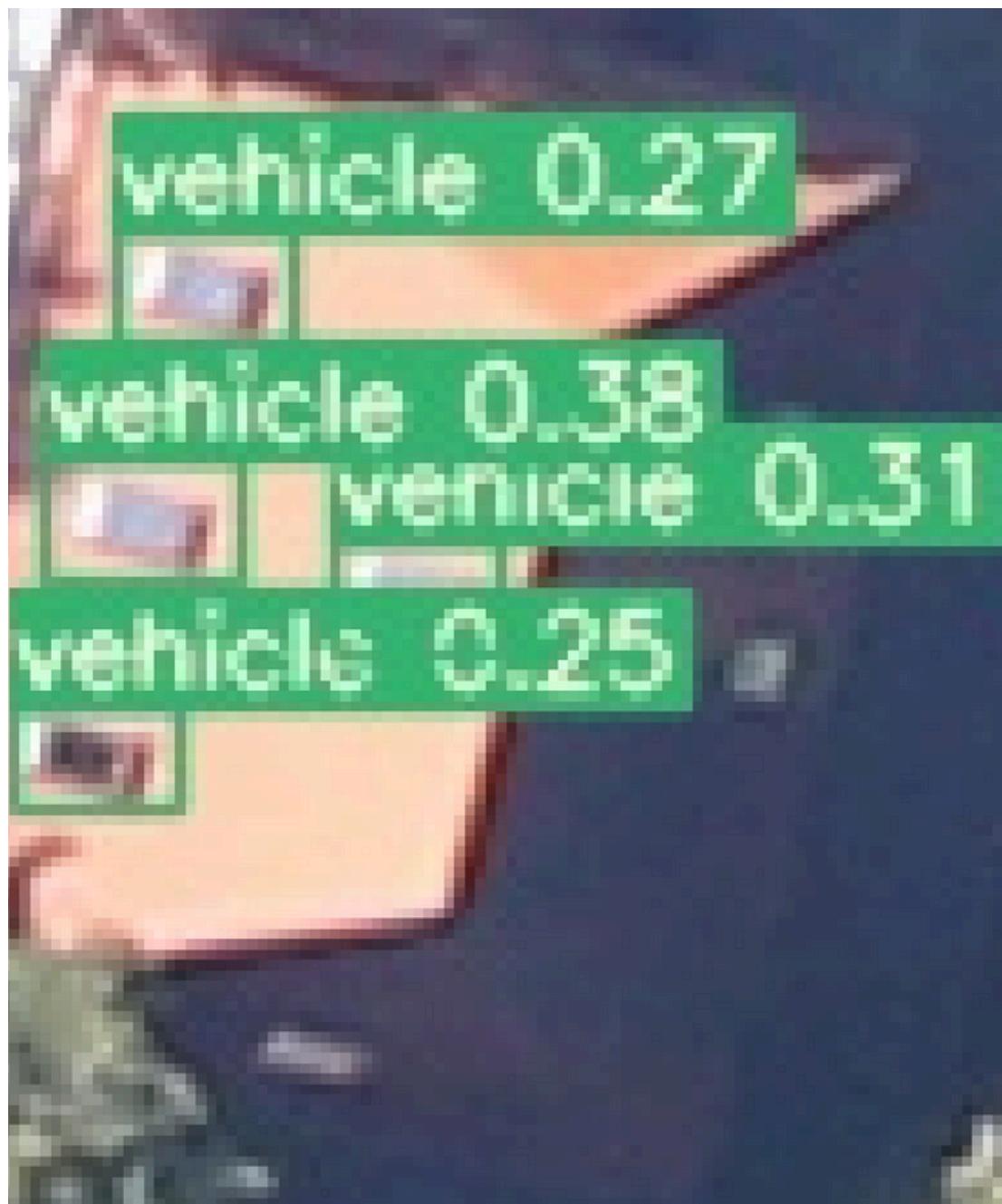
53 張測試集

Total vehicles : 25754

Public\_score : 0.823

# Optimization- method 3

## Using CNN models

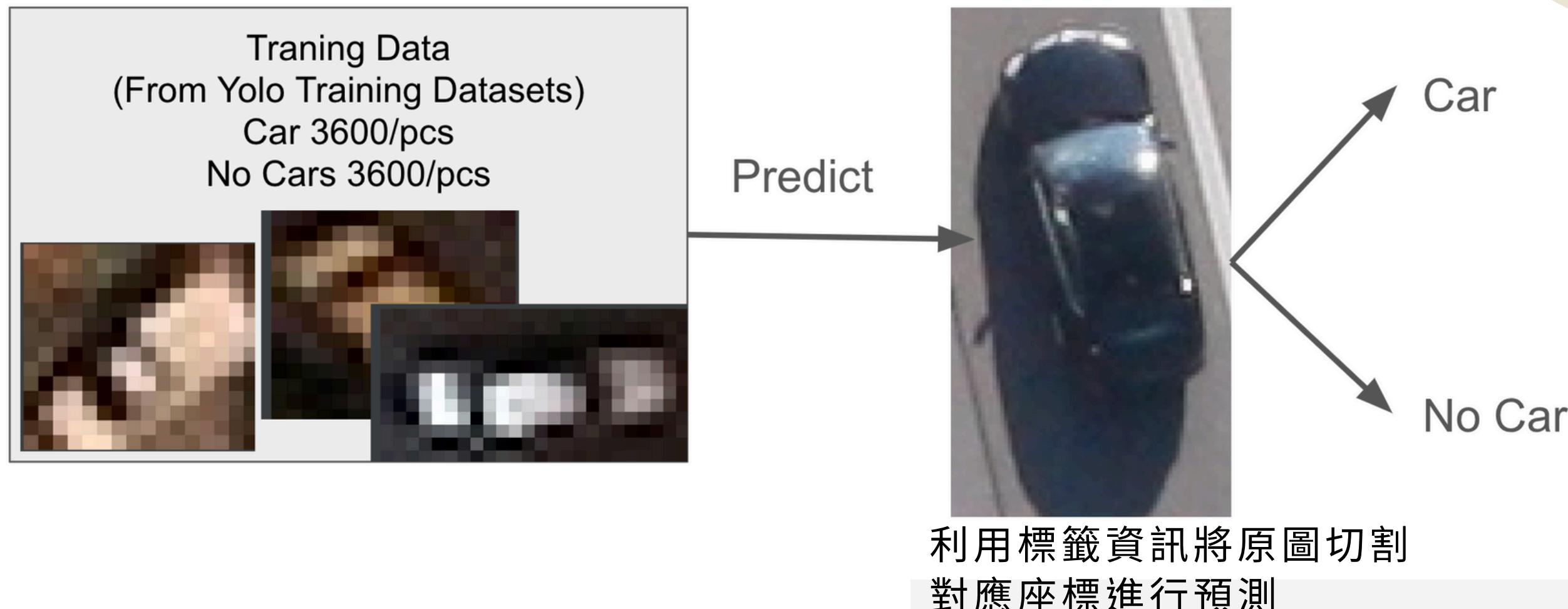


# Optimization- method 3

## Using CNN models

為了提高模型的準確率(Precision)  
將標籤好的照片單獨拉出進行二元分類

1. Car(車輛) 3600張
2. No\_Car (非車輛) 3600張



在模型信心水準 0.25

53 張測試集

Total vehicles : 24950

Public\_score : 0.826

# Conclusion and Reflections

經由不同的方式以及各結果如下，雖然嘗試的方法很多種，但我們認為資料前處理與調整網路層是最有效率提升指標的方式。也致謝助教提供資源以及廖老師提供一個有深度且具有挑戰的作業，讓我們用最短時間熟悉並實作yolov7。

方法	偵測物件數量	Kaggle Public Score
原始模型	21069	0.821
原始模型 + 切割	25754	0.823
原始模型 + 切割 + CNN	24950	0.826
原始模型 + 調整信心分數	104	0.926



Samuel, Dennis, Erin, Agnes

# THANK YOU

18 April, 2025