

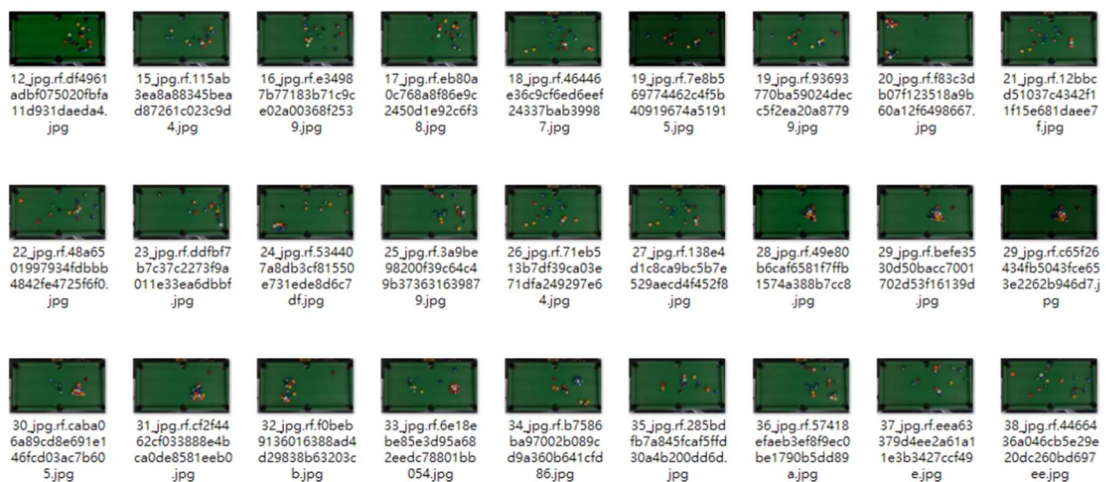
Roboflow + YOLOv8

使用 Roboflow 創立自定義模型

1. 自行蒐集資料集或使用網路上公開資料集(如: Kaggle、COCO、Roboflow

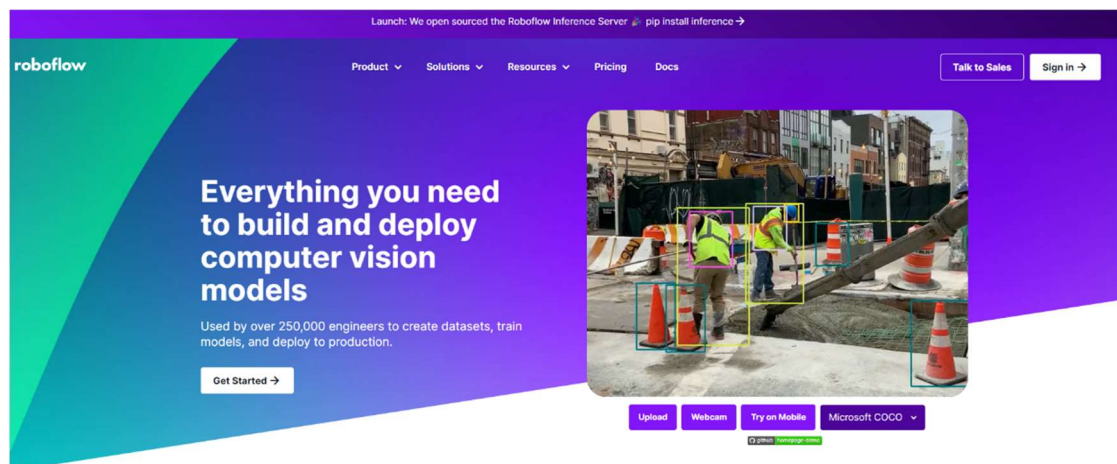
Universe 等) · 如下圖使用先前上銀機械手臂比賽的撞球影像共 50 張(數

量可以找 30、50 或以上)。

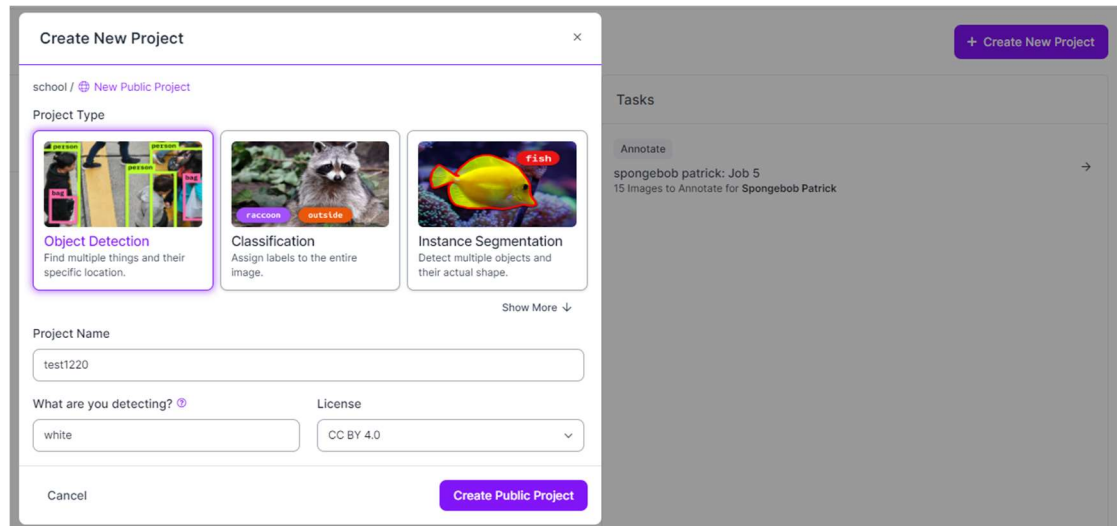


2. 進入 Roboflow 網站(<https://roboflow.com/>) · 點選右上 Sign up 註冊自

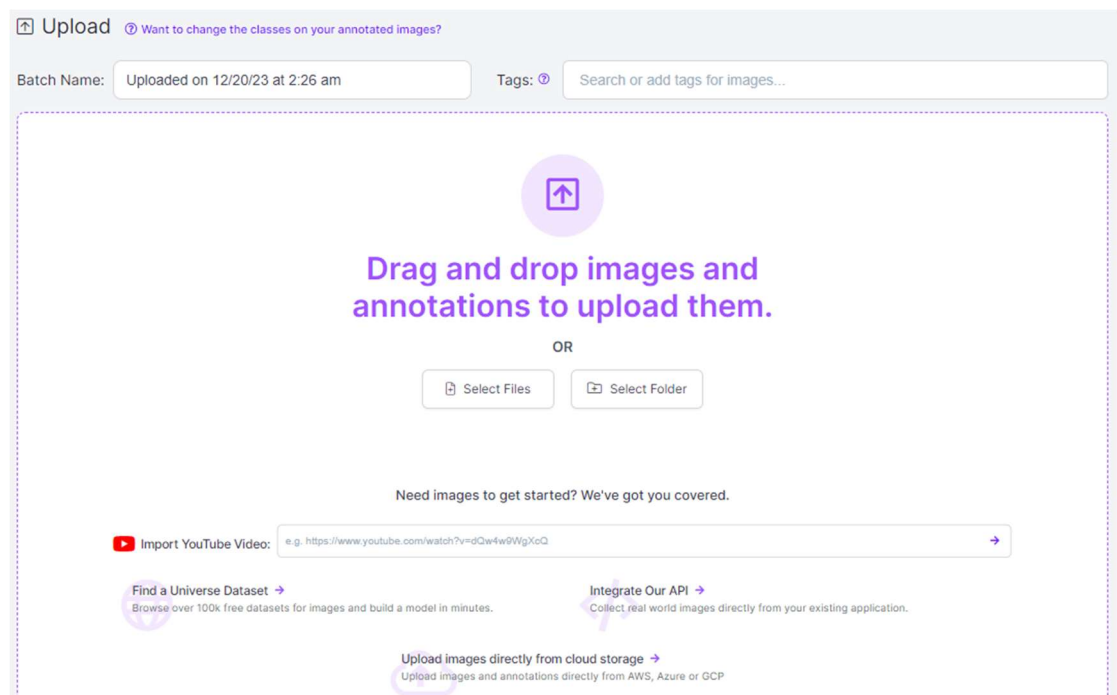
己的帳號 · 並進行登入。



3. 點選 Create New Project，依照自行需求選擇 Project Type 類型，這邊就以 Object Detection 作為示範。

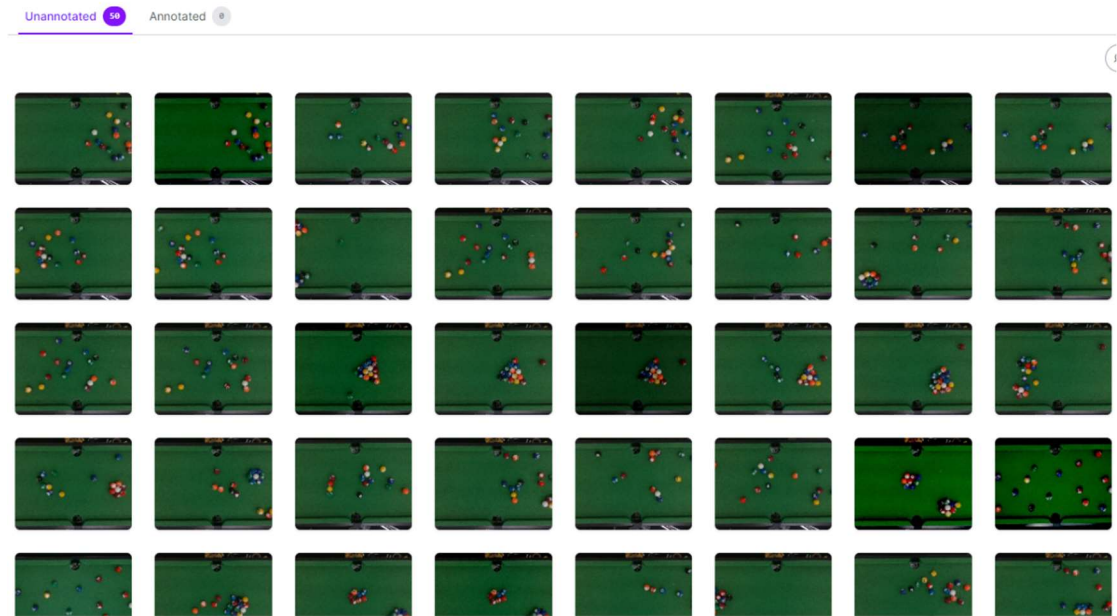


4. 建立完專案後，進行自定義資料集建置與上傳。

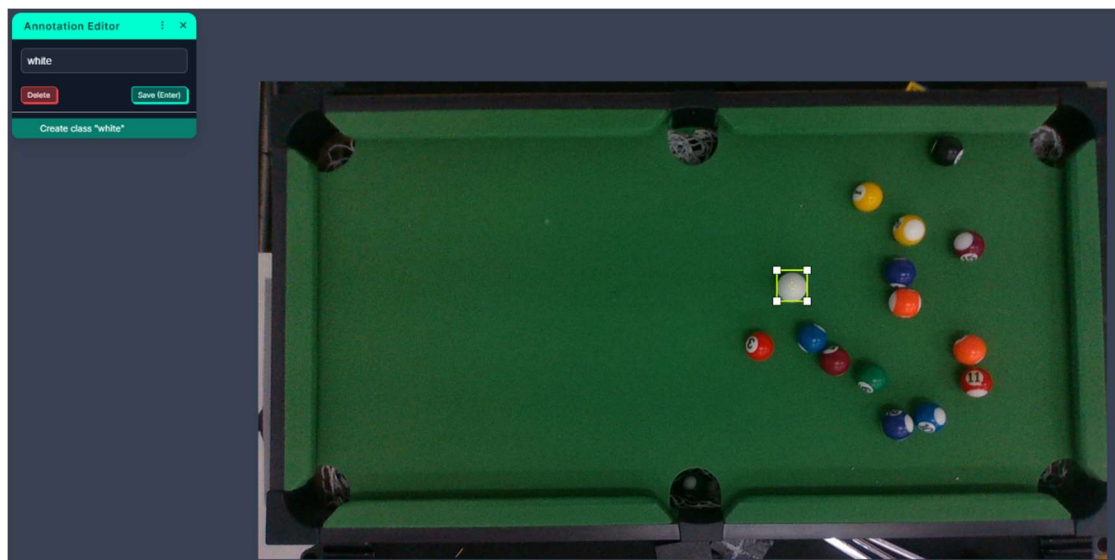


5. 上傳完資料之後就透過 Roboflow 工具進行資料標註(可自定義標籤)。

這裡測試使用標註分類白色母球「White」與黑色子球「Black」共兩種。



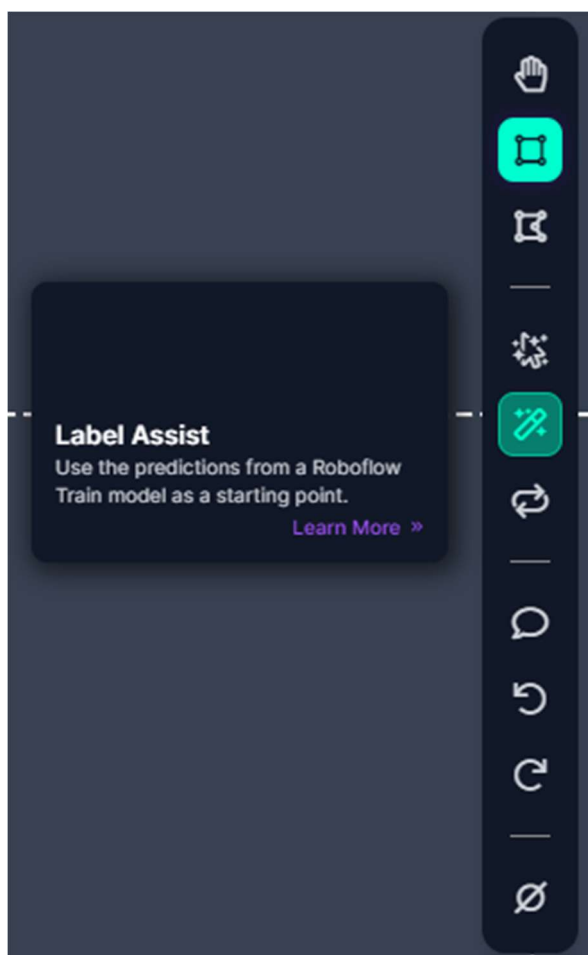
利用右邊 Bounding Box Tool 做物體的框選與新增類別。



畫面右側有工具列可做選擇使用如 Label Assist，一種使用模型的先前版

本來推薦標註的工具。因專案使用 Object Detection，避免使用其他

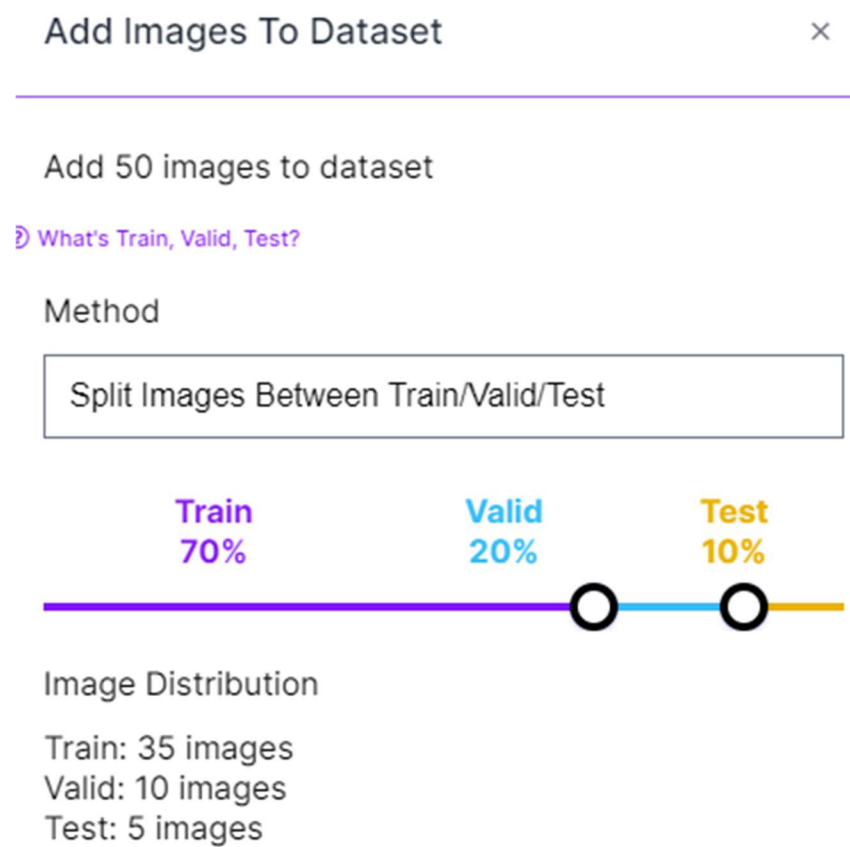
Segmentation 工具來描繪物體的整體輪廓，否則訓練模型會有問題。



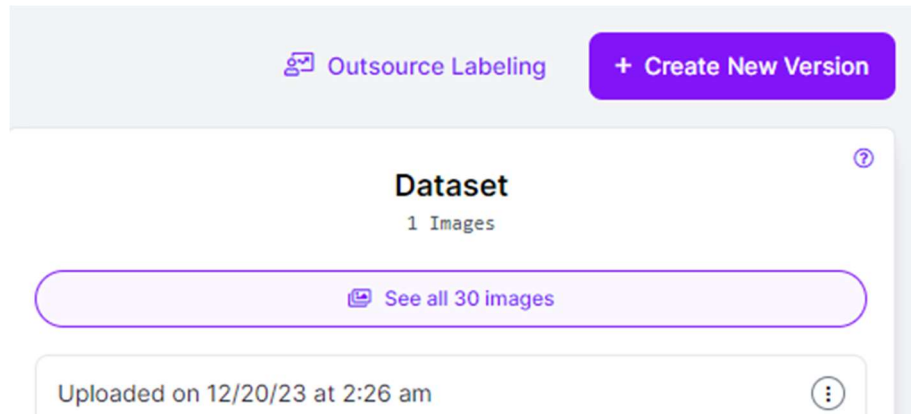
6. 標註完成後回到 Annotate 那頁，點選右上角 Add images to Dataset



7. 系統會先幫你設定訓練集、驗證集、測試集比率，也可以自行調整。



8. 點選 Create New Version，創立此資料集的版本，後續也可增加訓練資料或其他更動並創立更多新版本等。



9. 在版本中可以看到所使用的資料分配

- 影像數量(Images)
- 類別(Classes)
- 尚未標註(Unannotated)
- 訓練集 (Training Set)
- 驗證集 (Validation Set)
- 測試集(Testing Set)
- 預處理(Preprocessing)
- 資料增強(Augmentations)

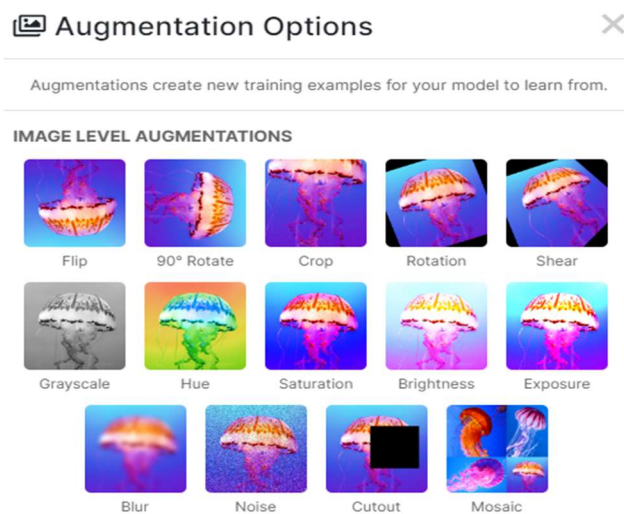
Creating New Version

Prepare your images and data for training by compiling them into a version. Experiment with different configurations to achieve better training results.

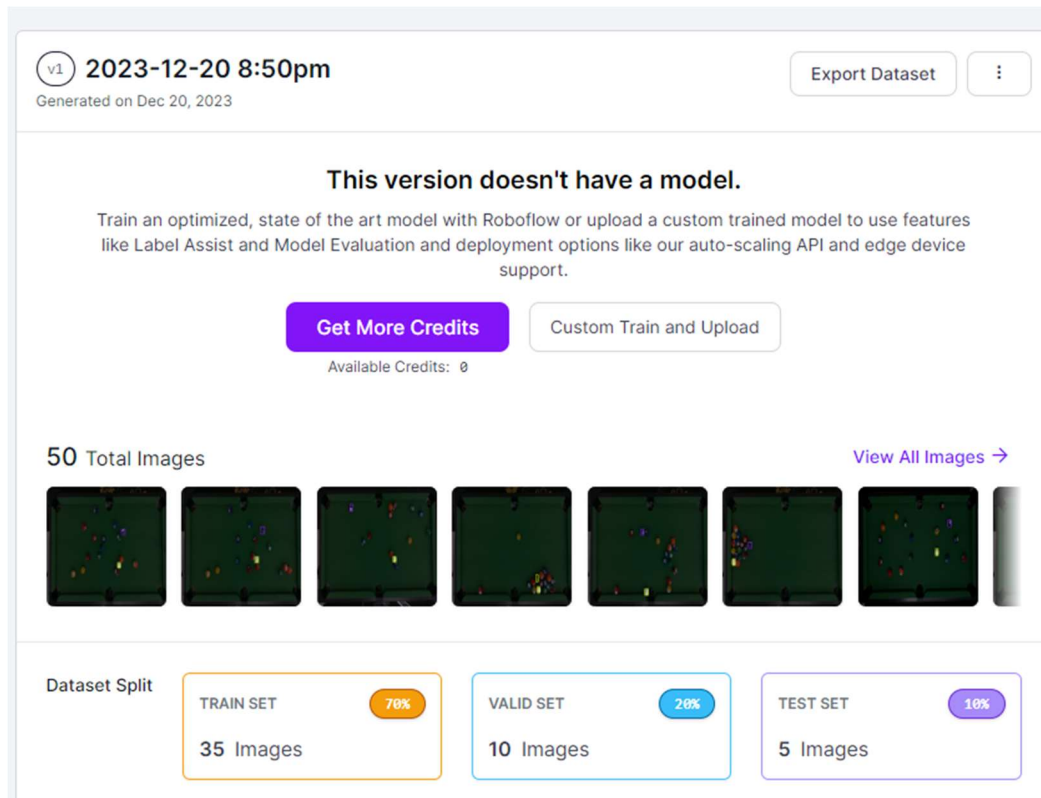
✓	Source Images	Images: 50 Classes: 2 Unannotated: 0
✓	Train/Test Split	Training Set: 35 images Validation Set: 10 images Testing Set: 5 images
✓	Preprocessing	Auto-Orient: Applied Resize: Stretch to 640×640
4	Augmentation	Create new training examples for your model to learn from by generating augmented versions of each image in your training set. <div><div>+</div> Add Augmentation Step</div> <div>Continue</div>
5	Create	

資料增強(Augmentation)使原始影像增加上下顛倒、90 度翻轉、模糊等

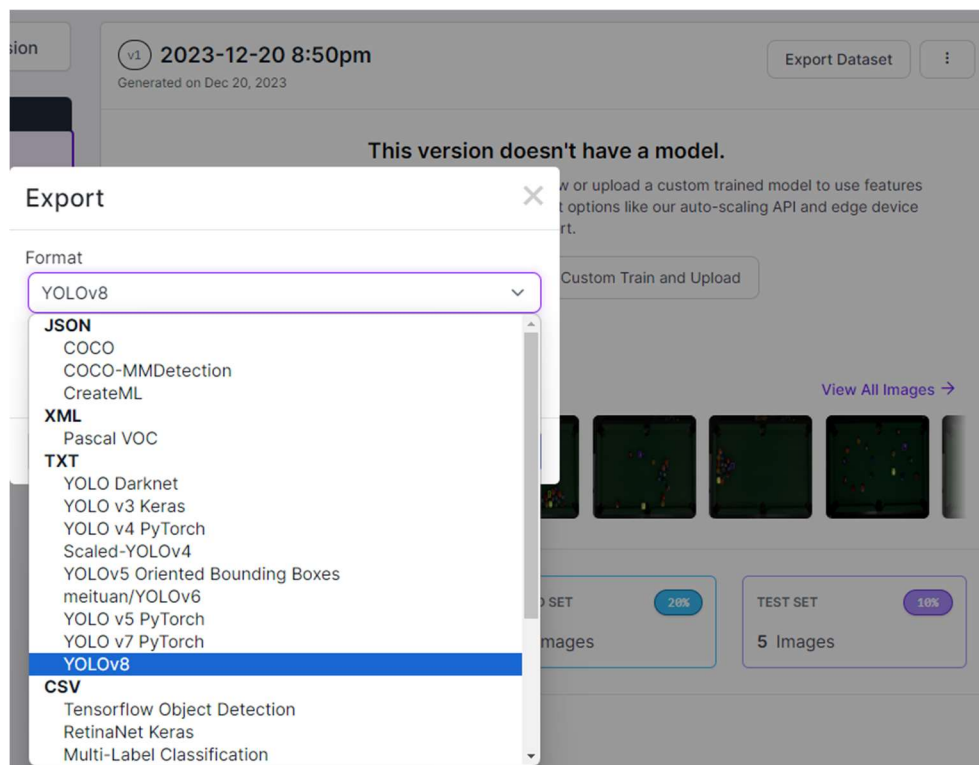
效果可以幫助提高模型的泛化能力，從而更有效地處理未見過的影像。



10. 創建完後版本會顯示日期時間，並點選右上 Export Dataset。



11. 選擇 YOLOv8(也可以依個人需求或實驗選擇其他配置)



12. 目前標註的專案到這裡就完成了，後續訓練模型的部分，教學會以

Google Colab 為主。接下來就先複製 Jupyter 中如下的部分。

Your Download Code



 Jupyter

>_ Terminal

[Raw URL](#)

Paste this snippet into [a notebook from our model library](#) » to download and unzip [your dataset](#) »:

```
!pip install roboflow
```

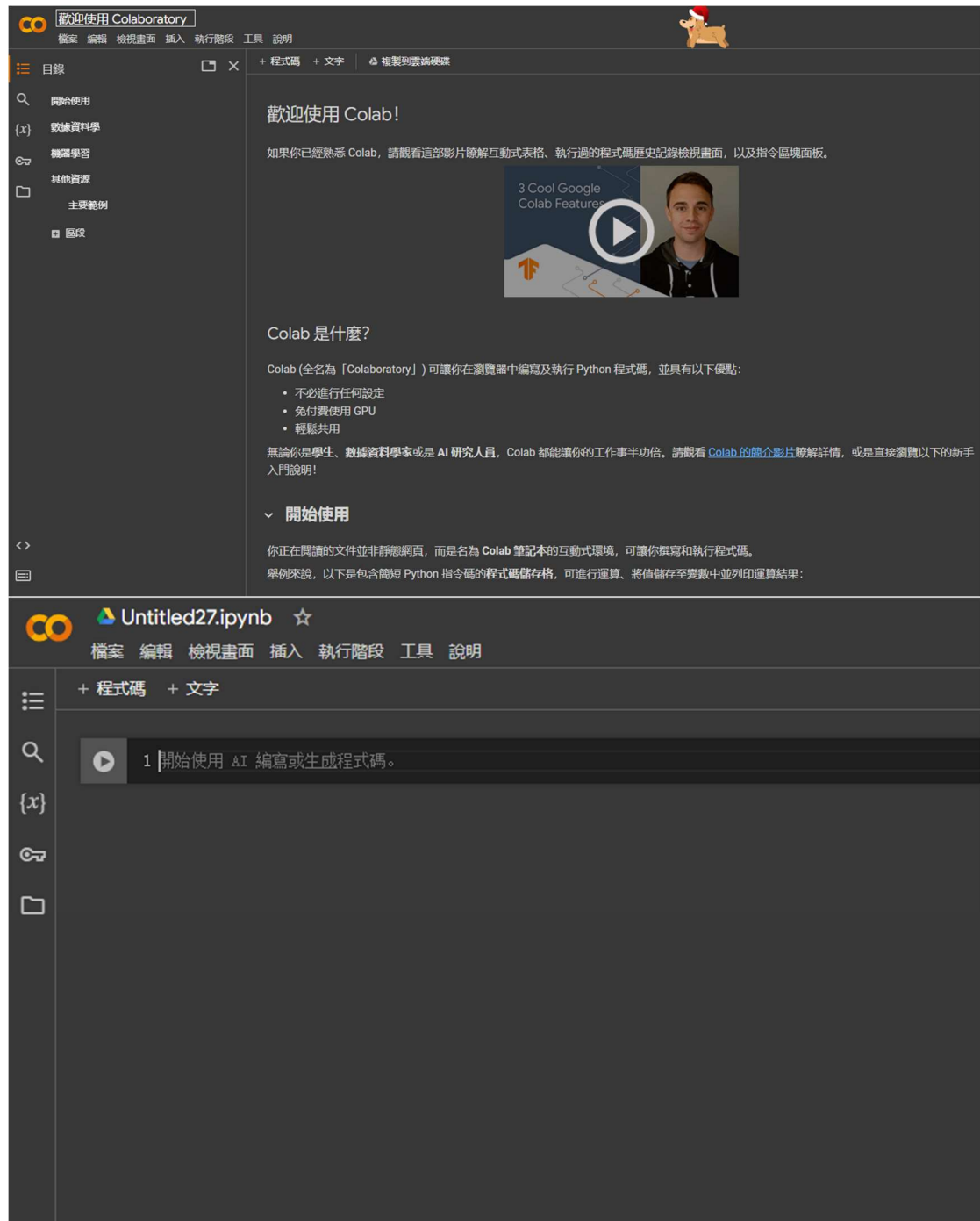


```
from roboflow import Roboflow  
rf = Roboflow(api_key="XXXXXXXXXXXXXXXXXXXX")  
project = rf.workspace("school-jxfxc").project("test1222")  
dataset = project.version(1).download("yolov8")
```

使用 Colab 訓練模型

13. 使用 Google Colab (<https://colab.research.google.com/>)，開啟記事本

並建立新專案。



14. 載入 YOLOv8 與 Python 影像套件

```
#YOLOv8整合至ultralytics並透過該框架就能夠進行YOLO系列相關操作
!pip install ultralytics==8.0.20
```

```
import ultralytics
ultralytics.checks()
```

```
from ultralytics import YOLO
```

```
from IPython.display import display, Image
```

將前面 Roboflow 建好的專案輸出貼到 Colab 中，如下

```
#有關於Dataset建置部分可以參考下方文章進行自定義資料收集
#影像處理Roboflow使用教學參考來源:https://www.wpgdadatong.com/blog/detail/71282
#創建資料集位置並透過API進行引用, API KEY部分登入帳號進行操作將會自行產生
!mkdir /content/datasets
%cd /content/datasets

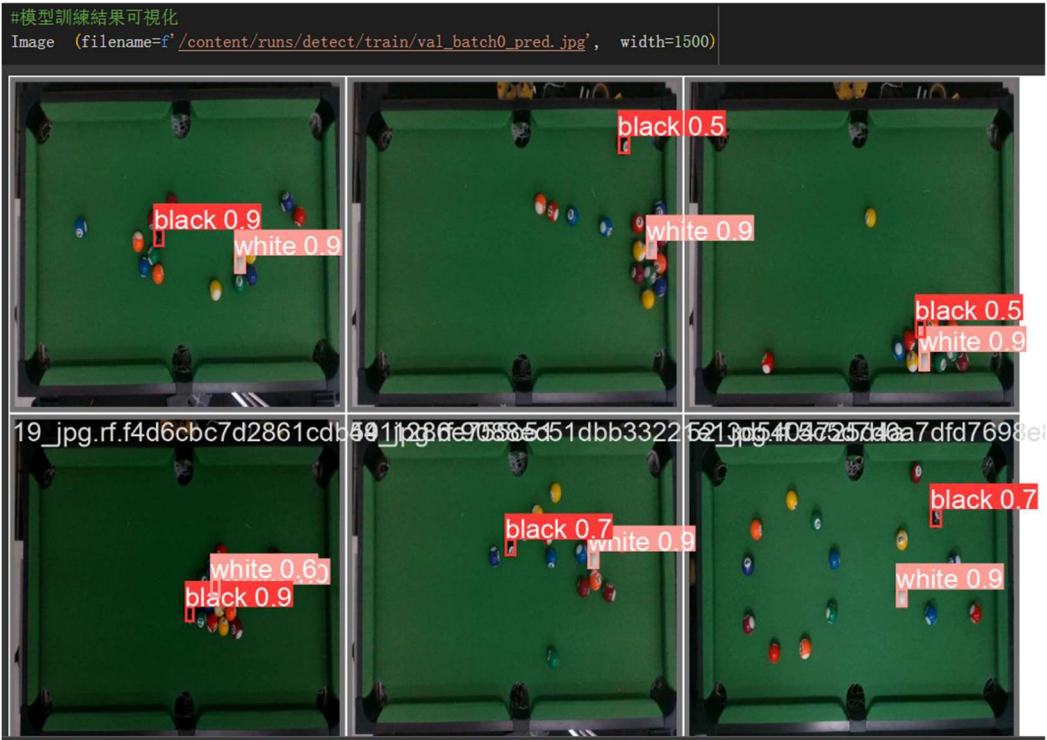
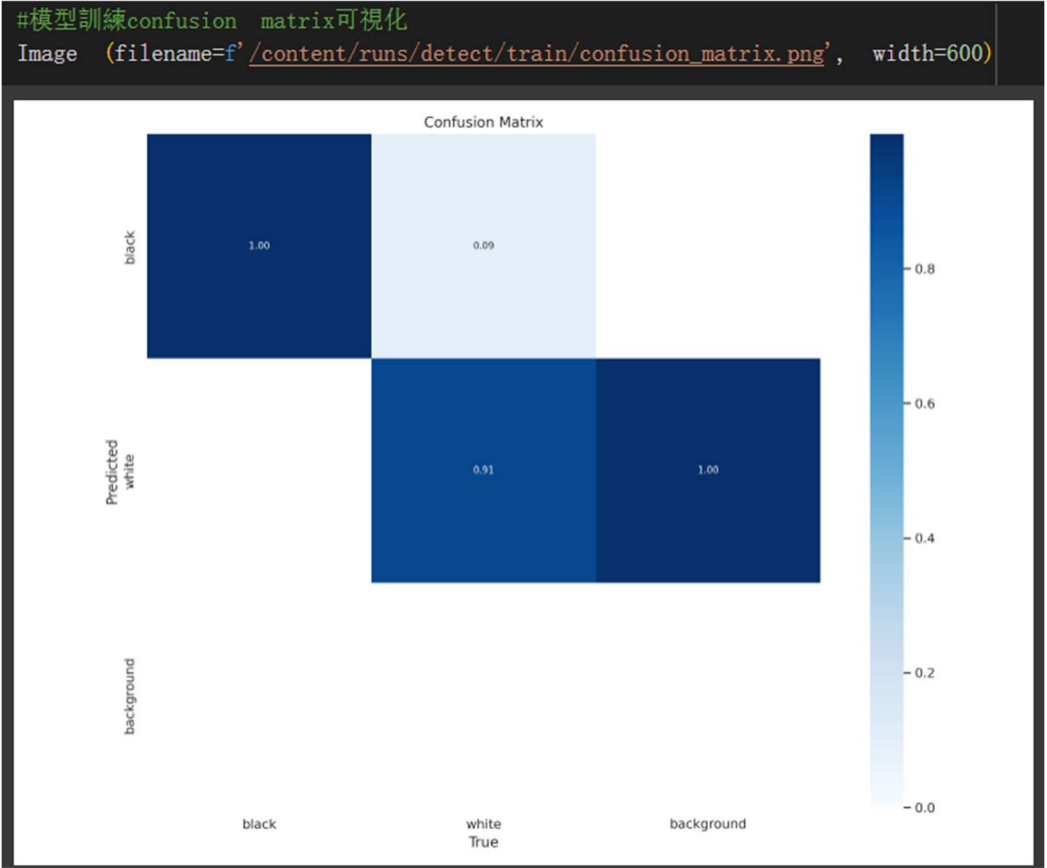
#=====Export Datasets=====
!pip install roboflow

from roboflow import Roboflow
rf = Roboflow(api_key="r9BpcvphquT6EPGSuIGB")
project = rf.workspace("school-jxfxc").project("test1220")
dataset = project.version(1).download("yolov8")
#=====Export Datasets=====
```

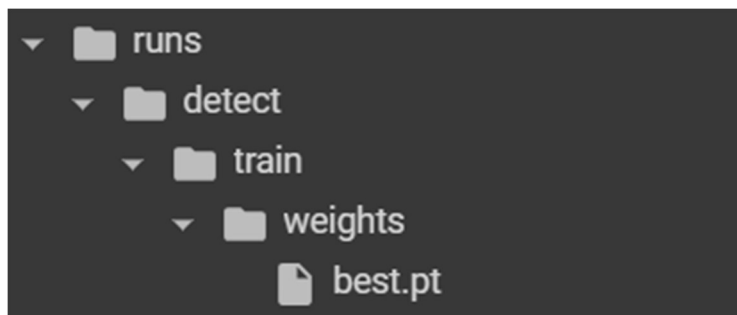
任務模式使用 detect，模型使用 yolov8s(若訓練時間太久可使用 GPU 或 yolov8n)，epoches 可依實驗需求調整，這裡使用 25 次為測試，並開始訓練模型。

```
%cd /content
!yolo task=detect mode=train model=yolov8s.pt data={dataset.location}/data.yaml epochs=25 imgsz=800
```

訓練完後可將模型中的混淆矩陣(Confusion Matrix)輸出顯示，評估模型效果。並將訓練結果可視化。



並使用訓練完的模型(best.pt)讀取影像測試結果。



```
import os
import matplotlib.pyplot as plt

# Load a model
model = YOLO("/content/runs/detect/train/weights/best.pt")

# Predict with the model
results = model.predict(source="/content/bk.jpg", save=True)

img = plt.imread('/content/runs/detect/predict/bk.jpg')
plt.imshow(img)
```



延伸應用

可將訓練完模型輸出，進行其他如撞球、水果、汽車或其他物件等即時辨識

識(Realtime)等相關應用，並針對實驗、比賽或專題方面發想。

如下是我測試的 Object Detection 模型，Classes 為「Spongebob」、

「Patrick」，影像總數量為 90 張。使用訓練完的模型檔(best.pt)與撰寫程


式來即時辨識現實物體影像。



檔案(F) 編輯(E) 選取項目(S) 檢視(V) 移至(G) 執行(R) ...

檔案總管 ... pre.py x README.md

Webcam



patrick

spongebob

東森幼幼

沒有人敢叫我胖呆

pre.py

README.md

YOLO

```
0)  
,"spongebob", ]  
read()  
, stream=True)
```

```
22  
23  
24 for box in boxes:
```

問題 輸出 偵錯主控台 終端機 連接埠

Confidence ---> 0.96
Class name --> patrick
Confidence ---> 0.47
Class name --> spongebob
0: 608x800 1 patrick, 1 spongebob, 148.5ms
Speed: 2.0ms preprocess, 148.5ms inference, 1.0

Confidence ---> 0.96
Class name --> patrick
Confidence ---> 0.45
Class name --> spongebob
0: 608x800 1 patrick, 1 spongebob, 143.7ms
Speed: 2.9ms preprocess, 143.7ms inference, 1.0