Lab09

Object Detection with YOLOv7

Dataset: https://reurl.cc/NyeQlx

Outline

安裝CUDA

安裝cuDNN

安裝pytorch

安裝yolo-v7

訓練自己的yolo-v7模型

Lab09

安裝CUDA

CUDA Toolkit 11.8 Downloads

這邊安裝
cuda11.8

[CUDA下載連結](#)

Select Target Platform

Click on the green buttons that describe your target platform. Only supported platforms will be shown. By downloading and using the software, you agree to fully comply with the terms and conditions of the CUDA EULA.

Operating System	Linux	Windows			
Architecture	x86_64				
Version	10	11	Server 2016	Server 2019	Server 2022
Installer Type	exe (local)	exe (network)			

Download Installer for Windows 11 x86_64

The base installer is available for download below.

> Base Installer   **Download (3.0 GB)**

Installation Instructions:

1. Double click cuda_11.8.0_522.06_windows.exe
2. Follow on-screen prompts

前面都一直點下一步或繼續

到安裝選項這邊點**自訂(進階選項)**

只要勾第一個CUDA

*Display Driver的部分，如果新版本大於目前的版本的話也要打勾

這邊記好安裝的路徑

這邊打勾並按NEXT開始安裝

NVIDIA 安裝程式

NVIDIA CUDA
版本 11.8

- ✓ 系統檢查
- ✓ 授權協議書
- 選項
- 安裝
- 完成

CUDA Visual Studio Integration

No supported version of Visual Studio was found. Some components of the CUDA Toolkit will not work properly. Please install Visual Studio first to get the full functionality.

☑ I understand, and wish to continue the installation regardless.

上一步(B)　　NEXT　　取消(C)

NVIDIA 安裝程式

NVIDIA CUDA
版本 11.8

- ✓ 系統檢查
- ✓ 授權協議書
- ✓ 選項
- ✓ 安裝
- **完成**

「NVIDIA 安裝程式」已完成

元件	版本	狀態
Occupancy Calculator	11.8	已安裝
CUFFT Development	11.8	已安裝
Demo Suite	11.8	已安裝
NVJPEG Runtime	11.8	已安裝
CUDA Profiler API	11.8	已安裝
nvcc	11.8	已安裝

關閉(C)

```
C:\Users\covis223b>nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2022 NVIDIA Corporation
Built on Wed_Sep_21_10:41:10_Pacific_Daylight_Time_2022
Cuda compilation tools, release 11.8  V11.8.89
Build cuda_11.8.r11.8/compiler.31833905_0

C:\Users\covis223b>nvidia-smi
Mon Nov 20 15:58:59 2023
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 526.47       Driver Version: 526.47       CUDA Version: 12.0      |
|-------------------------------+----------------------+----------------------+
| GPU  Name            TCC/WDDM | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA GeForce ... WDDM  | 00000000:08:00.0  On |                  N/A |
| 46%   31C    P8    N/A /  75W |    484MiB /  4096MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+
```

開啟終端機輸入

nvcc -V

nvidia-smi

確認第一個紅框的部分是你剛剛安裝的版本

且第二個紅框的數字大於等於第一個紅框

安裝cuDNN

安裝CUDNN

點擊Download cuDNN Library(需要登入NVIDIA會員才能下載)

依照剛剛安裝的CUDA版本以及作業系統選擇

cuDNN Download
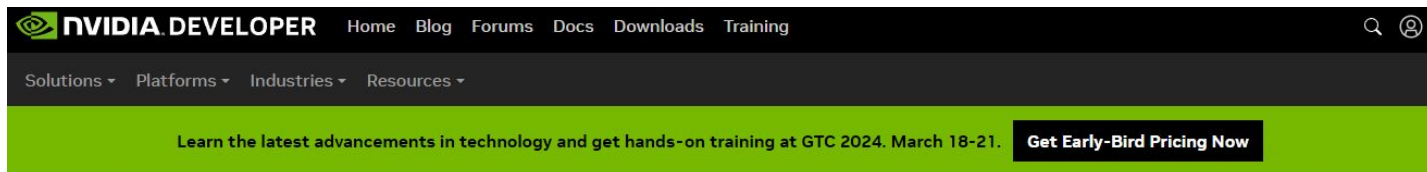
NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

☑ I Agree To the Terms of the cuDNN Software License Agreement

Note: Please refer to the Installation Guide for release prerequisites, including supported GPU architectures and compute capabilities, before downloading.

For more information, refer to the cuDNN Developer Guide, Installation Guide and Release Notes on the Deep Learning SDK Documentation web page.

Download cuDNN v8.9.6 (November 1st, 2023), for CUDA 12.x

Download cuDNN v8.9.6 (November 1st, 2023), for CUDA 11.x

Local Installers for Windows and Linux, Ubuntu(x86_64, armsbsa)

Local Installer for Windows (Zip)

Local Installer for Linux x86_64 (Tar)

Local Installer for Linux PPC (Tar)

Local Installer for Linux SBSA (Tar)

Local Installer for Debian 11 (Deb)

Local Installer for Ubuntu18.04 x86_64 (Deb)

Local Installer for Ubuntu20.04 x86_64 (Deb)

Local Installer for Ubuntu22.04 x86_64 (Deb)

Local Installer for Ubuntu20.04 aarch64sbsa (Deb)

Local Installer for Ubuntu22.04 aarch64sbsa (Deb)

下載完長這樣
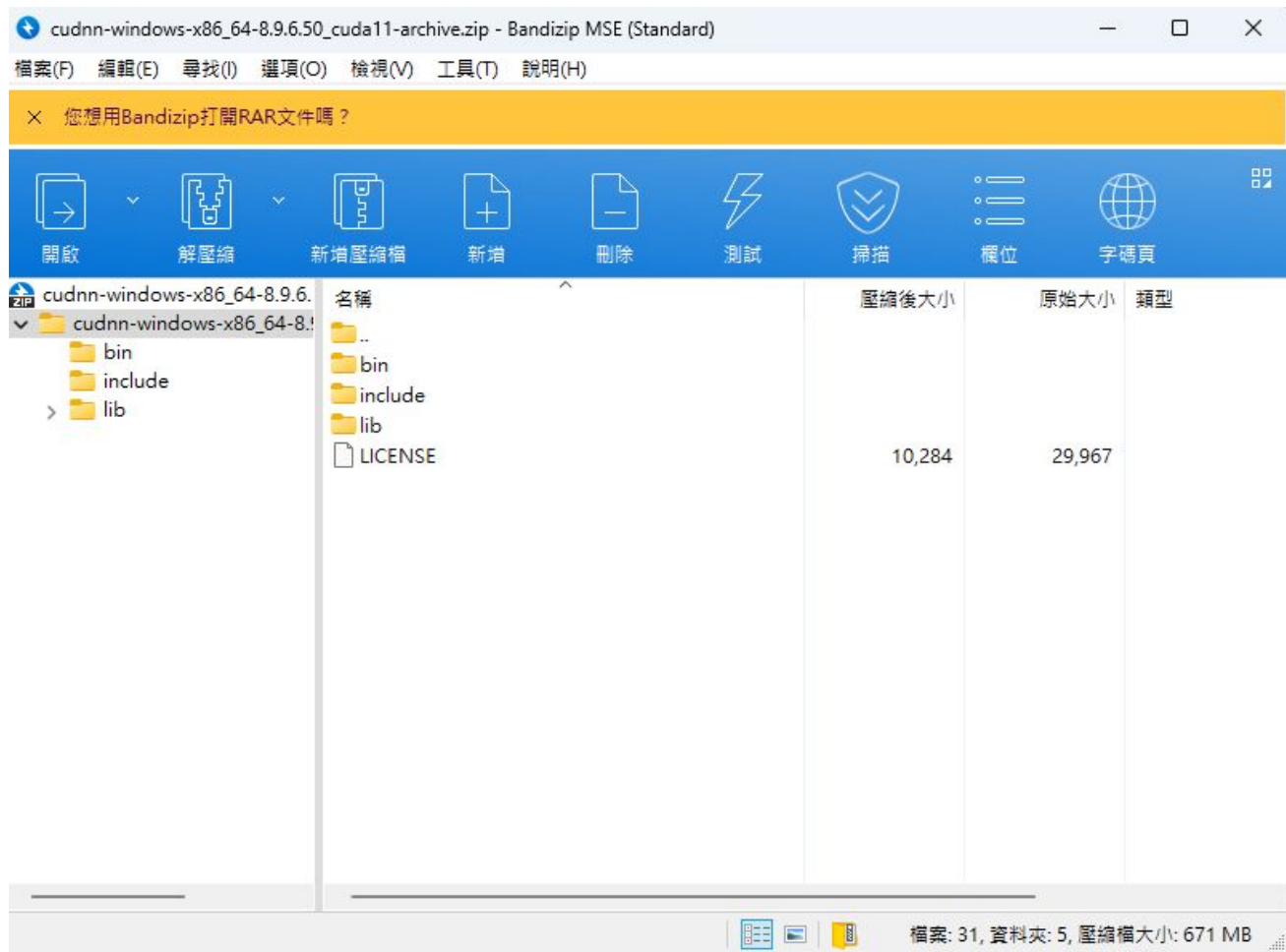
找到你前面安裝CUDA的路徑，預設安裝路徑:

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.8

將壓縮包內所有檔案丟進去覆蓋
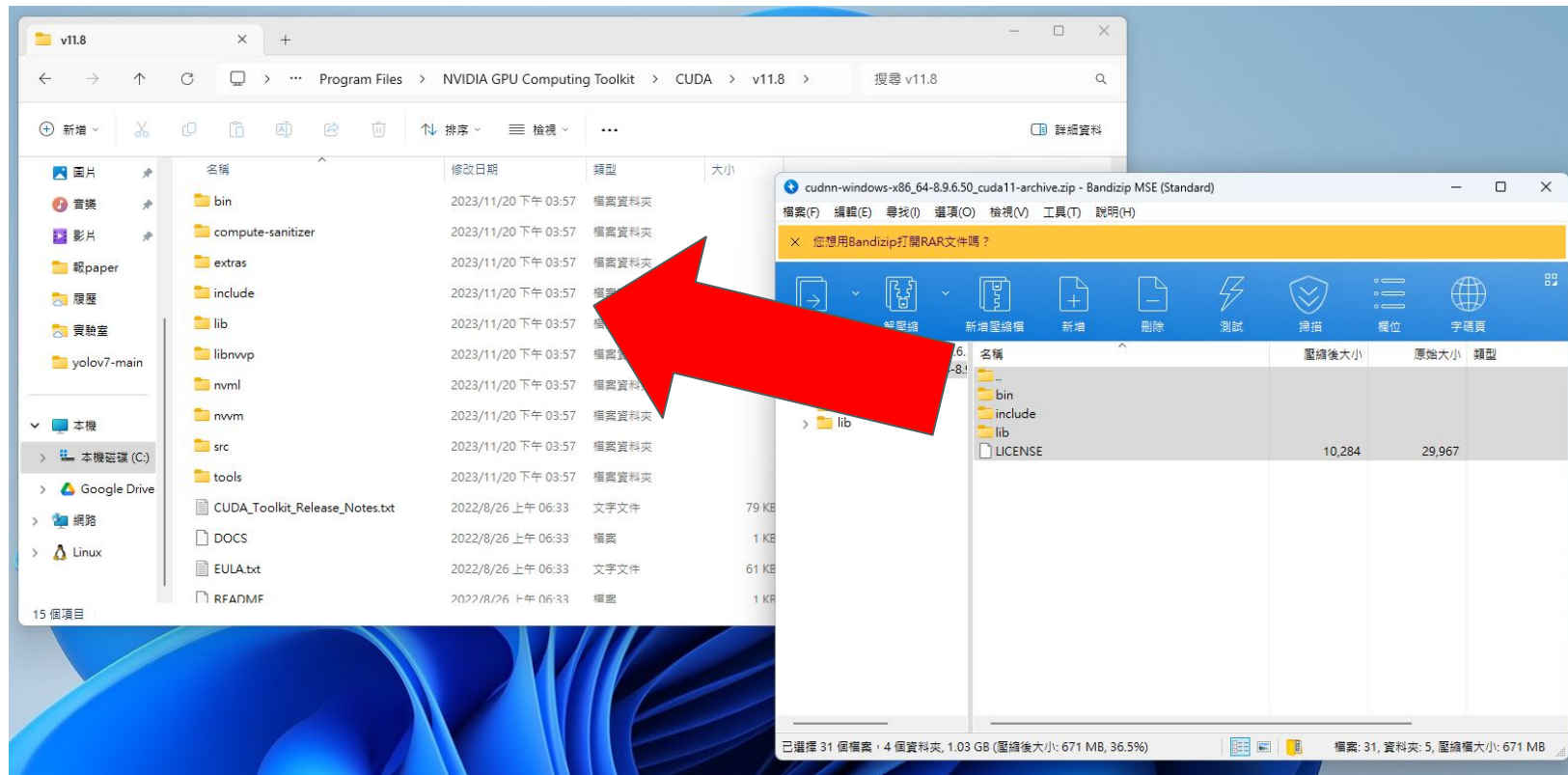
安裝Pytorch GPU版本

下載Pytorch

在你的conda環境輸入指令:

pip install torch torchvision torchaudio --index-url
https://download.pytorch.org/whl/cu118

PyTorch Build	Stable (2.1.1)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 11.8	CUDA 12.1	~~ROCm 5.6~~	CPU
Run this Command:	pip3 install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu118			

NOTE: PyTorch LTS has been deprecated. For more information, see this blog.

輸入pip list確認是否安裝成功

*最後面要有+cuXXX才是GPU版本的


```
torch                2.1.1+cu118
torchaudio           2.1.1+cu118
torchvision          0.16.1+cu118
```

開啟python確認輸出結果是否一樣

import torch

torch.cuda.is_available()

torch.zeros((3,3), device="cuda")


```
(uav) C:\Users\covis223b>python
Python 3.9.18 (main, Sep 11 2023, 14:09:
Type "help", "copyright", "credits" or "
>>> import torch
>>> torch.cuda.is_available()
True
>>> torch.zeros((3,3), device="cuda")
tensor([[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]], device='cuda:0')
```

安裝yolo-v7

下載yolo-v7

解壓縮，並在該目錄開啟終端
機來安裝必要的套件

pip install -r requirements.txt
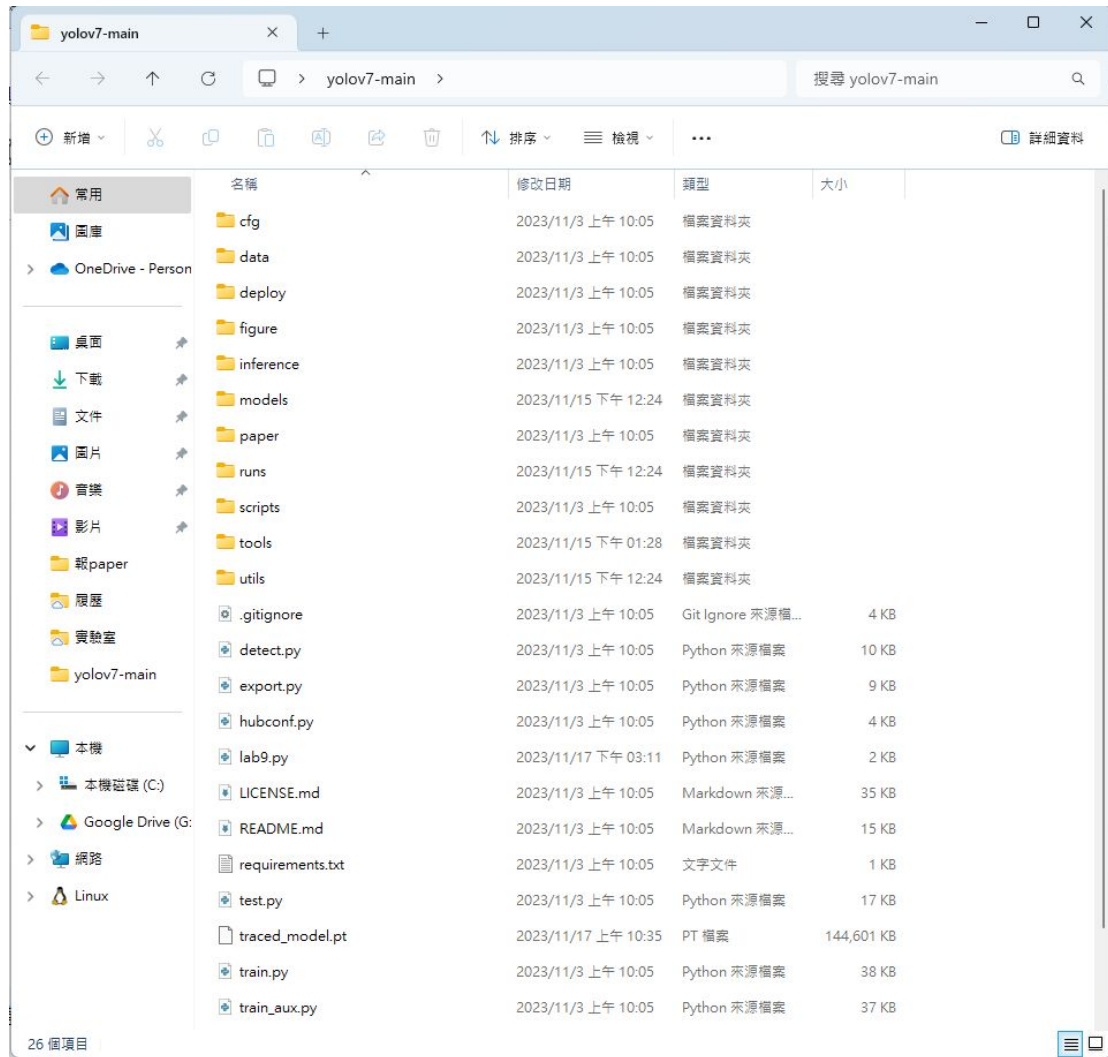
下載yolov7 tiny的[權重](#)並放入同一資料夾當中

python detect.py --weights yolov7-tiny.pt --conf 0.25 --img-size 640 --source inference/images/

確認是否能夠跑出結果


```
4 persons, 1 bus, Done. (18.0ms) Inference, (13.0ms) NMS
 The image with the result is saved in: runs\detect\exp7\bus.jpg
5 horses, Done. (13.0ms) Inference, (1.0ms) NMS
 The image with the result is saved in: runs\detect\exp7\horses.jpg
2 persons, 1 tie, 1 donut, Done. (16.0ms) Inference, (2.0ms) NMS
 The image with the result is saved in: runs\detect\exp7\image1.jpg
2 persons, 1 sports ball, Done. (13.0ms) Inference, (1.0ms) NMS
 The image with the result is saved in: runs\detect\exp7\image2.jpg
1 dog, 1 horse, Done. (15.0ms) Inference, (2.0ms) NMS
 The image with the result is saved in: runs\detect\exp7\image3.jpg
2 persons, 1 tie, Done. (11.0ms) Inference, (2.0ms) NMS
 The image with the result is saved in: runs\detect\exp7\zidane.jpg
Done. (0.625s)
```
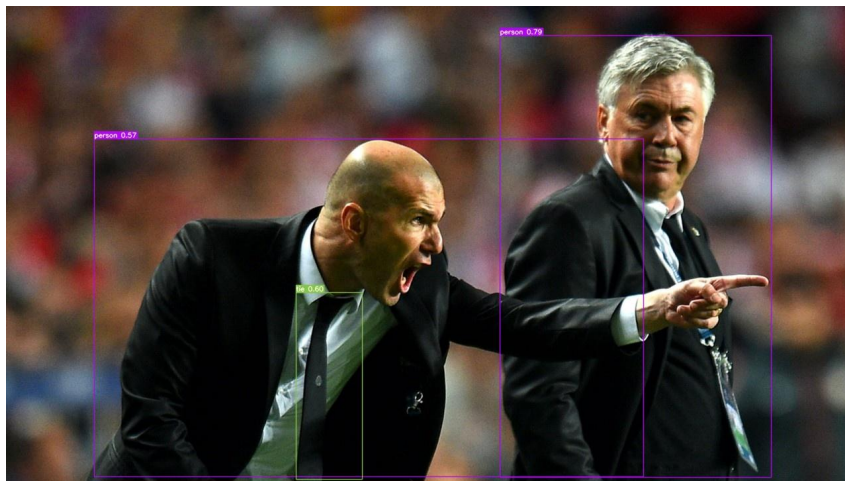
訓練自己的yolo-v7模型

標註照片

Dataset: https://reurl.cc/NyeQlx

Dataset共有三個資料夾

1. 只有卡娜赫拉:469張
2. 只有美樂蒂:481張
3. 兩個同時出現:589張

● 不需要全部都標
● 兩個種類的數量要差不多
● 怎麼標要講好(ex.框臉or全身)
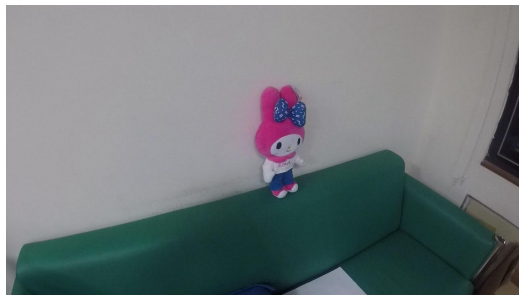
[roboflow](http://roboflow.com) 雲端共享Labeling的工具，請先註冊會員

Public
UAVlab09 ✎ 1 👥 ⌄

&+ Invite   ⚙ Settings   ⬆ Upgrade
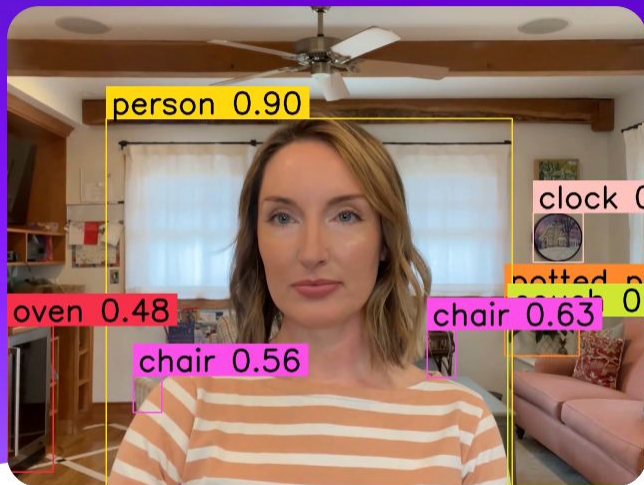
Projects   + Create New Project

上方的invite可以邀請成員

點擊Create New Project

Tasks

✧
This workspace has no projects.
Create new projects and they'll appear here.

+ Create New Project

☑ There are no tasks here yet!
Tasks will appear here once you have images to annotate, submit, or approve.

選擇Object Detection

Create New Project ✕

UAVlab09 / 🌐 New Public Project

Project Type

Object Detection
Find multiple things and their specific location.
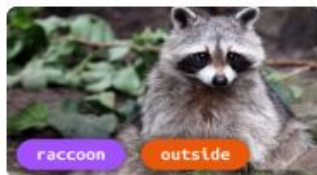
Classification
Assign labels to the entire image.

Instance Segmentation
Detect multiple objects and their actual shape.

Show More ↓

Project Name

lab09

What are you detecting? ⑦

dolls

License

MIT

Cancel

Create Public Project

點擊Select Folder將所有圖片上傳

點擊Assign
images來分配
Label的任務

UAVLAB09

🌐 **lab09** ⋮
Object Detection

☷ Sharing Page

↑ Upload

🔍 Unassigned

🖾 Annotate

🖾 Images   0

⇄ Generate

🗁 Versions

⌀ Deploy

◈ Health Check

🖸 Annotate   🖾 **Outsource Labeling**

Unassigned ⑦   **Annotating** ⑦   **Dataset** ⑦
1 Images   0 Images   0 Images

⊞ Upload More Images   Upload and assign images to an   Approve annotated images to add them
                               annotator.   to your dataset
🔍 View Unassigned Images

All data   ⋮
1533 unassigned images

                    Assign Images →

這邊可以分配要label的數量給團隊裡面的每一個人

Assign Images for Annotation ✕

TOTAL IMAGES TO ASSIGN: **1533** / 1533

Assign Images to Teammates

Choose teammates to label images. Images will be evenly divided between selected teammates.

You can assign specific images on the Unassigned images tab

🔍 Search for teammates...

曾正豪
zxcv3558160@gmail.com
1533 images

⚡ Outsource Labeling

👤+ Invite Teammate

分配完之後就可
以開始進行標記

這是標記的介面，先匡出物體的方框出來，然後輸入標記物體的名稱，若是有輸入過的可以直接用滑鼠選擇
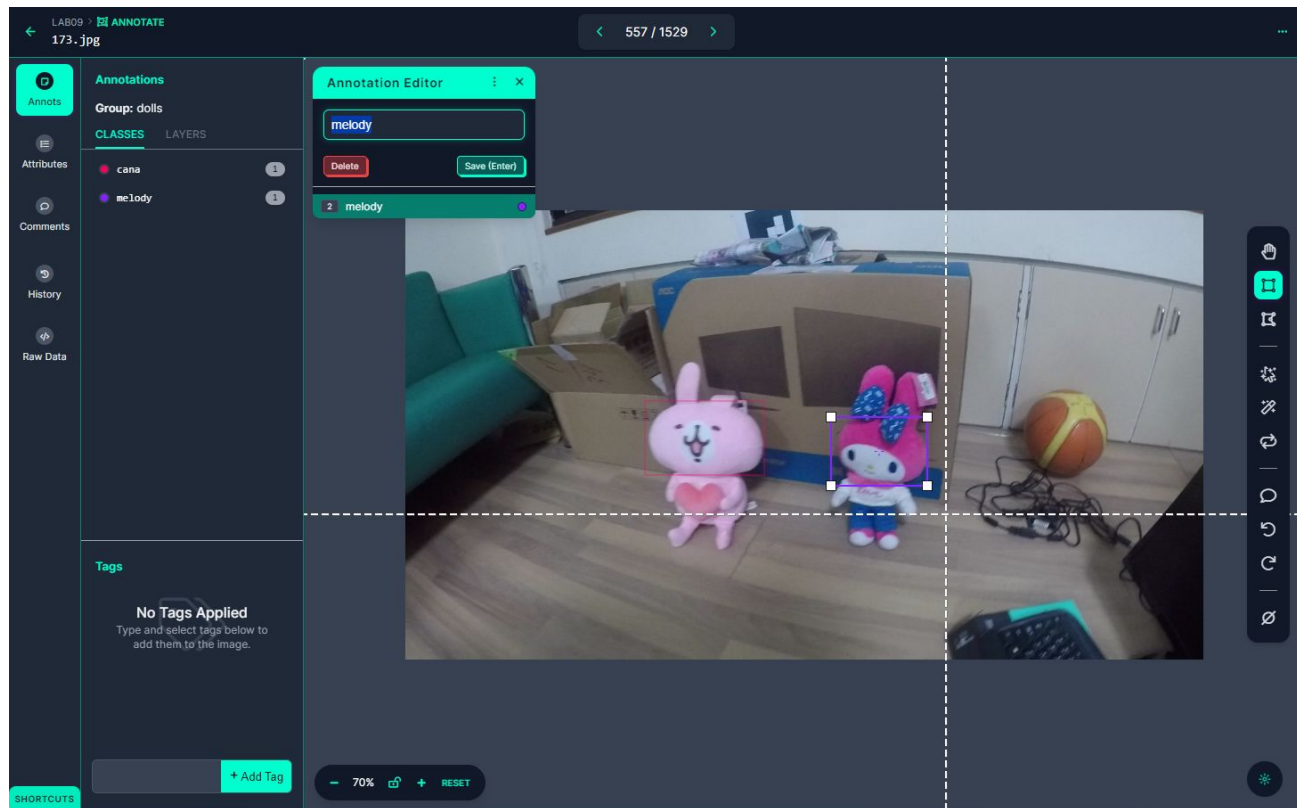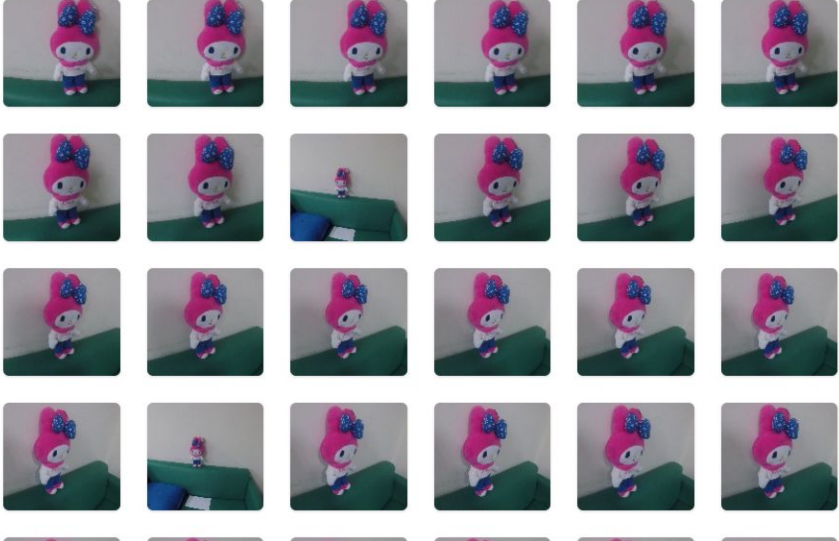
標記完之後點擊 Add images to Dataset

然後跟切分出訓練、測試資料集的數量出來

本次lab建議:

Train/Valid/Test

80%/20%/0%

Add Images To Dataset ✕

Add 40 images to dataset

⊘ What's Train, Valid, Test?

Method

Split Images Between Train/Valid/Test

Train 70%	Valid 20%	Test 10%

Image Distribution

Train: 28 images
Valid: 8 images
Test: 4 images

You are about to add 40 images to the dataset

1493 images will be sent back as part of a new job

Add Images

點擊左邊的Generate
來生成dataset出來，
這邊可以選擇資料前
處理或是增強。我們
這邊選擇都不要做。

UAVLAB09

lab09
Object Detection

📄 Sharing Page

↑ Upload

🔍 Unassigned

🖼 Annotate

🖼 Images   40

⚡ Generate

🗂 Versions

🚀 Deploy

🩺 Health Check

⊕ Generate New Version

VERSIONS

To train a model, you must
first generate a new version
of your dataset.

Choose your dataset settings
to get started.

Generating 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:** 40
                              Classes: 2
                              Unannotated: 0

✓ **Train/Test Split**   **Training Set:** 36 images
                              Validation Set: images
                              Testing Set: 4 images

3 **Preprocessing**
Decrease training time and increase performance by applying image
transformations to all images in this dataset.

⊕ **Add Preprocessing Step**

Continue

4 **Augmentation**

5 **Generate**

最後點擊
Generate

UAVLAB09

🌐 lab09 ⋮
Object Detection

▤ Sharing Page

↑ Upload

🔍 Unassigned

▣ Annotate

🖾 Images   40

⇄ Generate

🗀 Versions

✈ Deploy

♡ Health Check

⊕ Generate New Version

VERSIONS

To train a model, you must first generate a new version of your dataset.

Choose your dataset settings to get started.

Generating 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: 40
Classes: 2
Unannotated: 0

✔ **Train/Test Split**
Training Set: 36 images
Validation Set: images
Testing Set: 4 images

✔ **Preprocessing**
None

✔ **Augmentation**
Turned Off

5 **Generate**
Review your selections then click "Generate" to create a moment-in-time snapshot of your dataset with the applied preprocessing steps.

Maximum Version Size: 40
See how this is calculated »

Generate

點擊右上角
Export
Dataset

格式選擇Yolo v7

Export   ✕

Format

YOLO v7 PyTorch   ⌄

TXT annotations and YAML config used with **YOLOv7**.

◉download zip to computer   ○ show download code

Cancel   Continue

將下載下來的所有檔案複製到 data 資料夾當中
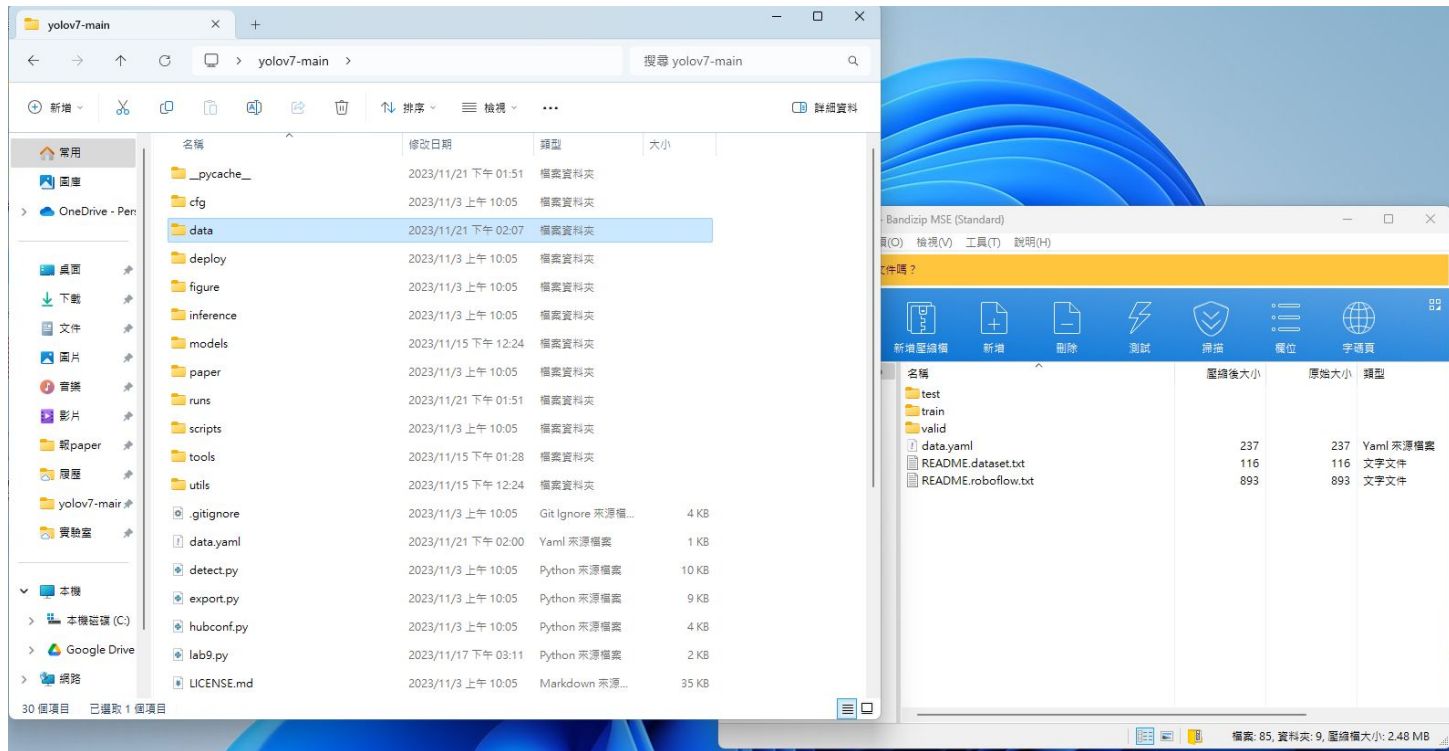
接著修改data.yaml的內容
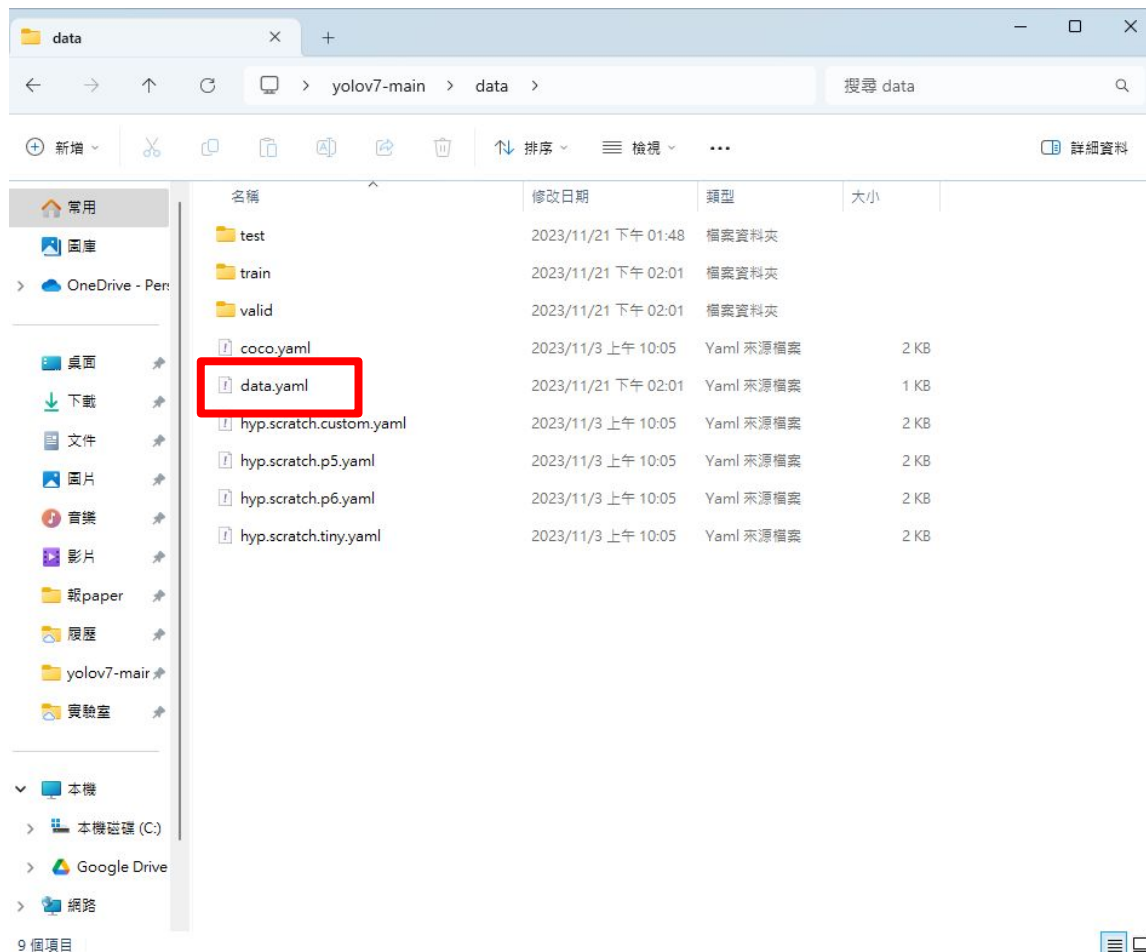
將前三行修改成資料集的路徑

```
1   train: ./data/train/images
2   val: ./data/valid/images
3   test: ./data/test/images
4   |
5   nc: 2
6   names: ['cana', 'melody']
7
8   roboflow:
9     workspace: uavlab09
10    project: lab09
11    version: 2
12    license: MIT
13    url: https://universe.roboflow.com/uavlab09/lab09/dataset/2
```

將 cfg\training\yolov7-tiny.yaml
複製一份出來並改名
yolov7-tiny-lab9.yaml

修改檔案第二行的nc，改為2

```
1    # parameters
2    nc: 2  # number of classes
3    depth_multiple: 1.0  # model depth multiple
4    width_multiple: 1.0  # layer channel multiple
5
6    # anchors
7    anchors:
8      - [10,13, 16,30, 33,23]  # P3/8
9      - [30,61, 62,45, 59,119]  # P4/16
10     - [116,90, 156,198, 373,326]  # P5/32
```
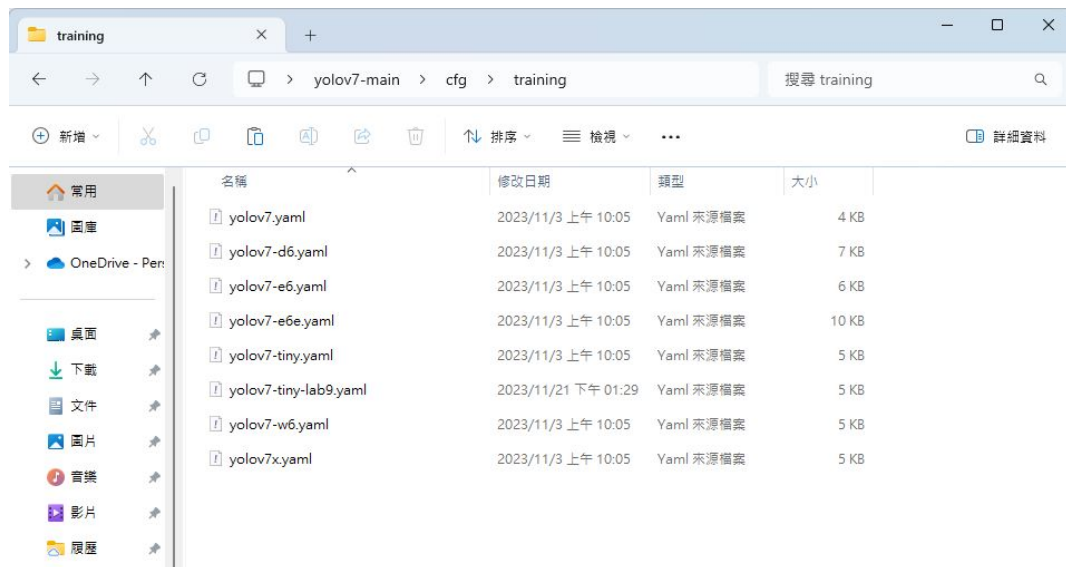
訓練的指令:

```
python train.py --device 0 --batch-size 8 --data data/data.yaml --img 1280 720 --cfg cfg/training/yolov7-tiny-lab9.yaml --weights 'yolov7-tiny.pt' --name yolov7-lab09 --hyp data/hyp.scratch.tiny.yaml --epoch 300
```

#batch-size 同時放多少張下去訓練(記憶體不夠有error就降低數量)

#data資料集的yaml檔　#img影像尺寸　#cfg 自定義模型的config檔

#name名字, 隨便取 #weights從哪個初始權重開始訓練

#hyp各種設定的超參數(可以自己調調看)

正常跑的話會像這個樣子


```
train: Scanning 'data\train\labels' images and labels... 36 found, 0 missing, 1 empty, 0 corrupted:
train: New cache created: data\train\labels.cache
val: Scanning 'data\valid\labels' images and labels... 1 found, 0 missing, 0 empty, 0 corrupted: 10
val: New cache created: data\valid\labels.cache

autoanchor: Analyzing anchors... anchors/target = 4.38, Best Possible Recall (BPR) = 1.0000
Image sizes 1280 train, 736 test
Using 2 dataloader workers
Logging results to runs\train\yolov7-lab0910
Starting training for 300 epochs...

     Epoch   gpu_mem       box       obj       cls     total    labels  img_size
     0/299     1.49G   0.06453   0.05509   0.01442     0.134         3      1280: 100%|█| 18/18 [00
               Class    Images    Labels         P         R    mAP@.5 mAP@.5:.95: 100%|
                 all         1         0         0         0         0           0

     Epoch   gpu_mem       box       obj       cls     total    labels  img_size
     1/299      1.5G   0.05584    0.0511   0.01278    0.1197         4      1280: 100%|█| 18/18 [00
               Class    Images    Labels         P         R    mAP@.5 mAP@.5:.95: 100%|
                 all         1         0         0         0         0           0

     Epoch   gpu_mem       box       obj       cls     total    labels  img_size
     2/299      1.5G   0.06299   0.04521   0.01357    0.1218         2      1280: 100%|█| 18/18 [00
               Class    Images    Labels         P         R    mAP@.5 mAP@.5:.95: 100%|
                 all         1         0         0         0         0           0

     Epoch   gpu_mem       box       obj       cls     total    labels  img_size
     3/299      1.5G   0.05625   0.04301   0.01272     0.112         9      1280: 100%|█| 18/18 [00
               Class    Images    Labels         P         R    mAP@.5 mAP@.5:.95: 100%|
                 all         1         0         0         0         0           0
```

訓練model - 調整訓練

- 框的時候盡量不要框進太多背景
- 選擇框臉或頭或身體
  - 框臉跟框身體的不能一起訓練，要統一
- 訓練時
  - 先學兩隻一起出現的，再訓練單一隻出現的
  - 全部放在一起訓練
  - 先訓練單一隻出現的，再訓練兩隻出現的
  - 觀看結果，假如美樂蒂效果較差，加強訓練美樂蒂的資料集
- 訓練時的learning rate
  - loss高時設大　ex. 1e-3
  - loss接近1時設小　ex. 1e-5 ~ 1e-7
- 一定要用pretrained weight, 不然效果很差
- 嘗試不同的.cfg檔及.weights檔
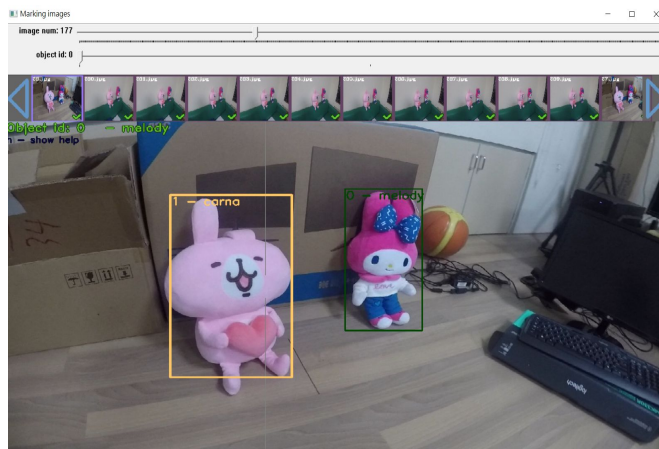
訓練完成的權重會放在.\runs\train\yolov7-lab09\weights

best是valid最好的那一個epoch

last是最後一個epoch

名稱	修改日期	類型	大小
best.pt	2023/11/21 下午 02:16	PT 檔案	11,989 KB
epoch_000.pt	2023/11/21 下午 02:13	PT 檔案	47,520 KB
epoch_024.pt	2023/11/21 下午 02:15	PT 檔案	47,526 KB
epoch_025.pt	2023/11/21 下午 02:15	PT 檔案	47,526 KB
epoch_026.pt	2023/11/21 下午 02:15	PT 檔案	47,526 KB
epoch_027.pt	2023/11/21 下午 02:16	PT 檔案	47,526 KB
epoch_028.pt	2023/11/21 下午 02:16	PT 檔案	47,527 KB
epoch_029.pt	2023/11/21 下午 02:16	PT 檔案	47,527 KB
init.pt	2023/11/21 下午 02:12	PT 檔案	23,720 KB
last.pt	2023/11/21 下午 02:16	PT 檔案	11,989 KB

Lab 09

Lab09.py

Lab09.py內是範例程式, 請放入YOLOv7的資料夾當中

將電腦攝像頭的影像來做實時的物件偵測

更改WEIGHT的檔案可以更換偵測的模型權重

```python
12  WEIGHT = './yolov7-tiny.pt'
13  device = "cuda" if torch.cuda.is_available() else "cpu"
14
15  model = attempt_load(WEIGHT, map_location=device)
16  if device == "cuda":
17      model = model.half().to(device)
18  else:
19      model = model.float().to(device)
20  names = model.module.names if hasattr(model, 'module') else model.names
21  colors = [[random.randint(0, 255) for _ in range(3)] for _ in names]
22
23
24  cap = cv2.VideoCapture(0)
25  while True:
26      ret, image = cap.read()
27      if not ret:
28          break
29
30      image_orig = image.copy()
31      image = letterbox(image, (640, 640), stride=64, auto=True)[0]
32      if device == "cuda":
33          image = transforms.ToTensor()(image).to(device).half().unsqueeze(0)
34      else:
35          image = transforms.ToTensor()(image).to(device).float().unsqueeze(0)
36      with torch.no_grad():
37          output = model(image)[0]
38      output = non_max_suppression_kpt(output, 0.25, 0.65)[0]
39
40      ## Draw label and confidence on the image
41      output[:, :4] = scale_coords(image.shape[2:], output[:, :4], image_orig.shape).round()
42      for *xyxy, conf, cls in output:
43          label = f'{names[int(cls)]} {conf:.2f}'
44          plot_one_box(xyxy, image_orig, label=label, color=colors[int(cls)], line_thickness=1)
45
46      cv2.imshow("Detected", image_orig)
47      cv2.waitKey(1)
48
49
```

Lab09.py

Lab09.py內是範例程式，請將範例程式改成輸入一段影片來進行物件偵測，並將偵測到的物體框出來標上Label與Confidence，最後輸出成一個新的影片檔