

vott_object_tracking 使用說明

一、 概要

程式	vott_object_tracking	
	vott_object_tracking_fpn	vott_object_tracking_iou
原理	IOU + FPN + ResNet50	IOU + Distance
適用情境	光源充足、畫面移動速度慢	所有
硬體需求	CPU + GPU (VRAM > 3GB)	CPU
運行速度	慢	非常快 (比 vott_object_tracking_fpn 運行一個 frame 的速度還快)
需要的檔案	Video (.mp4) Target (.vott 、 .json)	Target (.vott 、 .json)
參數預設值	reid_sim_threshold : 1.1 reid_iou_threshold : 0	Backward : 1 Distance : 200
備註	在符合適用情境下，效果可能會 比 vott_object_tracking_iou 好	如果大部分 frame 中 object 都很 密集，請優先考慮使用 vott_object_tracking_fpn
結論	如果沒特別原因，請直接選擇 vott_object_tracking_iou	

二、 vott_object_tracking_iou 使用說明

1. 調整參數

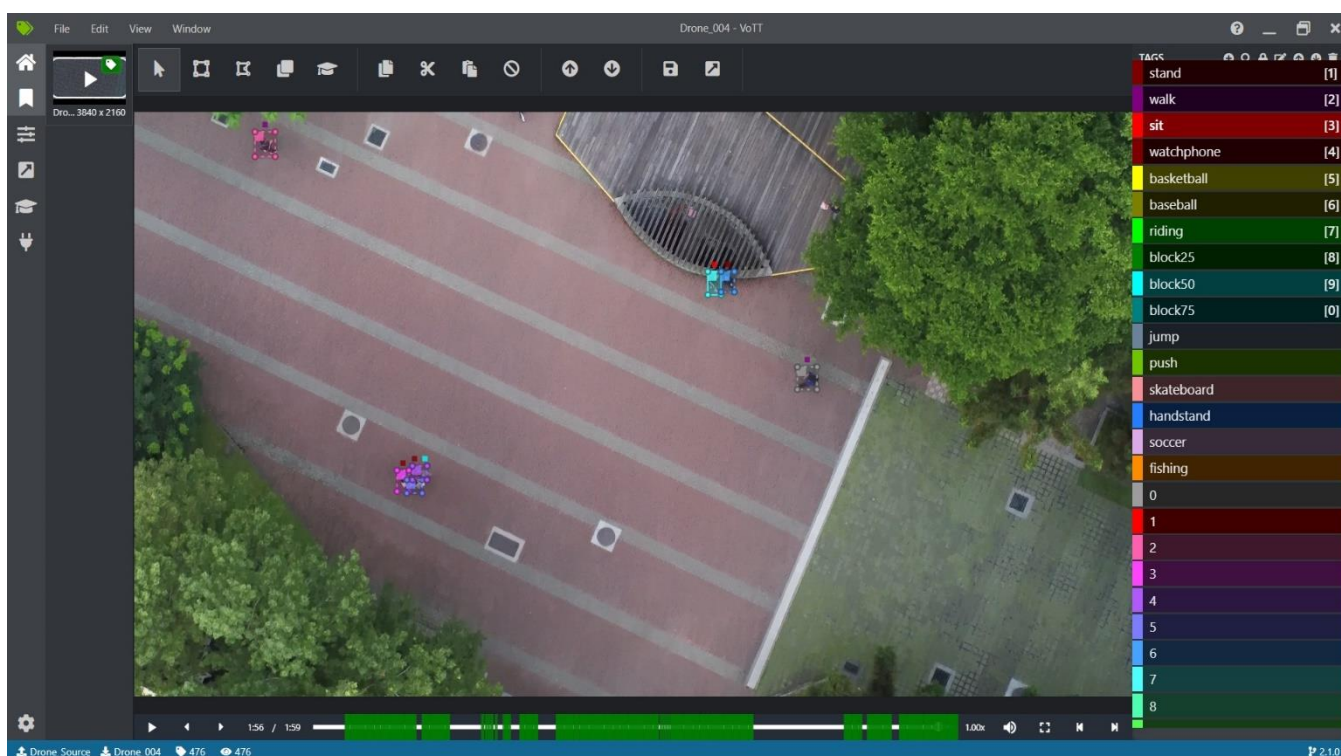
- 打開 `experiments/scripts/vott_object_tracking_fpn.py`，找到以下兩個參數並做適當調整
 - Distance (IOU 失敗時的搜索距離)
 - Backward (最多往後搜索幾個 frame)

2. 開始運行 vott_object_tracking_iou

- `python experiments/scripts/vott_object_tracking_iou.py`

```
Open D:\20190720_21\vott_object_tracking\Drone_004\Drone_004.vott  
475 frames were tagged  
Tracks found: 89  
Done
```

3. 在 VoTT 查看結果



三、vott_object_tracking_fpn 使用說明

4. 環境配置

- `conda create --name tracking python=3.7` 建立一個虛擬環境
- `conda activate tracking` 啟動該環境
- 打開 `requirements.txt` 將 `torch==1.4.0` 與 `torchvision==0.5.0` 刪除，因為會找不到，所以要用指定 channel 裝 (<https://pytorch.org/>)

`conda install pytorch torchvision cudatoolkit=10.1 -c pytorch`

- 請注意你電腦安裝的 NVIDIA CUDA 版本，如果不是 10.1，上方的 `cudatoolkit=`要改
- 如果你的 CUDA 版本比 <https://pytorch.org/> 中指示的版本要大，則 `cudatoolkit=`不改

```
ERROR: Could not find a version that satisfies the requirement torch==1.3.1 (from -r requirements.txt (line 79)) (from versions: 0.1.2, 0.1.2.post1, 0.1.2.post2)
ERROR: No matching distribution found for torch==1.3.1 (from -r requirements.txt (line 79))
```

- `pip install -r requirements.txt` 安裝必要的 packages
[Linux 下可以 conda 優先，pip 備用；使用以下指令代替上方指令]

```
while read requirement; do conda install --yes $requirement || pip install $requirement; done < requirements.txt
```

- `pip install -e .` 安裝 Tracktor
- `conda env export -n tracking` 確認該 env 中的 packages

5. 路徑修改

- 打開 `vott_object_tracking` 以下兩個 `.py` 並編輯設定 Image 資料夾所在的絕對路徑
 - `src\tracktor\datasets\ntut_wrapper.py` (17 行)
 - `src\tracktor\datasets\ntut_sequence.py` (53 行)

6. 調整參數

- 打開 `experiments\cfgs\tracktor.yaml`，找到以下兩個參數並做適當調整
 - `reid_sim_threshold` (AI 檢測是否同一人；越大越寬鬆)
 - `reid_iou_threshold` (IOU 閾值，不符合時會直接新增 ID，sim 會被忽略；越小越寬鬆，0 會關閉)

7. 資料準備

- 準備好已經用 vott 標註完的 Target (`.vott`、`.json`；不需要預輸出檔)

本機磁碟 (C:) > Drone_Target > Drone_004			
名稱	修改日期	類型	大小
 Drone_004.vott	2020/2/17 上午 02:43	VOTT 檔案	458 KB
 fef0320d971e43065cc386b12364556...	2020/2/13 下午 02:07	JSON File	2 KB
 fccb163260e1e1da15288172de99025...	2020/2/13 下午 02:07	JSON File	8 KB
 fcc4de2c746543e5191ed008639312e...	2020/2/13 下午 02:07	JSON File	3 KB
 fab58774f25849adf7126b858ec4ddfd...	2020/2/13 下午 02:07	JSON File	3 KB
 faa81ce14347b02f0bed0a4d1a977a4a...	2020/2/13 下午 02:07	JSON File	3 KB
 fa048212dca41d0c803008304bacca6a...	2020/2/13 下午 02:07	JSON File	8 KB
 fa381ec23917aaa376541f78f38cf1a3-...	2020/2/13 下午 02:07	JSON File	3 KB

- 準備好相應的影片，例如 `Drone_004.mp4`，放進 Video 資料夾中

8. 開始運行 vott_object_tracking_fpn

- python experiments/scripts/vott_object_tracking_fpn.py

```
WARNING - test_tracker - No observers have been added to this run
INFO - test_tracker - Running command 'main'
INFO - test_tracker - Started
```

- 填寫步驟 3 Target 所在位置

```
Please write your target folder path( /home/.../Drone_XXX ) :
```

- 接下來需要等待一段時間 (可以先去做別的事)

- 找到 .vott 檔並讀取

```
Open /home/ntut-drone/NTUTAction/Drone_004/Drone_004.vott
```

- 有多少 frame 是有被標記的 (這些 frame 將會被 extract 為圖片)

```
475 frames were tagged
```

- 圖片將會被保存為 6 位數檔名的.jpg

```
Image extract start
23 / 475
```

- 需要用到的圖片已經 extract 完成了

```
Image extract finish
```

- Tracktor 配置及 model 正在加載

```
Configuration (modified, added, typechanged, doc):
  seed = 453695
  reid:
    db_train = 'train'
    db_val = False
    description = 'all'
    lr_scheduler = True
    max_epochs = 69
    module_name = 'reid'
    name = 'res50-mot17-batch_hard'
    seed = 12345
    cnn:
      output_dim = 128
  dataloader:
    K = 4
    P = 18
    crop_H = 256
```

- Tracktor 程式需要的所有東西加載完成

```
INFO - main - Initializing object detector.
Tracktor初始化完成
```

- 開始一張一張圖片 tracking 並給出 ID，這些 ID 將會被寫回 Target 中

```
所有Drone_004加載完成, tracking開始
1% | 4/475 [00:02<04:42, 1.00 frames/s]
```

- 所有圖片跑完後會告知總共有幾個不同 ID (可以簡單的理解為找到了幾個不同的人)

```
INFO - main - Tracks found: 839
```

- 這個時候運行過程中解出來的圖片已經用不到了，如果沒特別需求的話就回答 y 來清理掉

Clean cache in /home/ntut-drone/NTUTAction/Image ? (y/n) :

9. 在 VoTT 查看結果

