* **物件檢測追蹤計數的目標 難點 和實現方法**
  + **目標**
    - 實現畫面上特定方向有多少人出現過就計數多少
  + **難點**
    - 圖像ai只能告訴我這frame或那frame有偵測到多少人
    - 即使使用追蹤模式它也不能去判斷這個人是否出現過
    - 在yolo眼中每個人只是檢測出的結果並無法分別這個人是否是誰
    - 如何從每frame的檢測出的人安排可計數的白名單和已計數的黑名單
  + **實現方法**
    - 使用方框檢測搭配追蹤 並將每個檢測的方框(左上跟右上)的終點當作人的點位
    - 使用兩個足夠細長的檢測框，較遠的稱為a，較近的稱為b，且這兩個方框需要足夠近又不能太近。
    - 人的點位被包含在a的時候，將這個人的id放入白名單
    - 人的點位被包含在b的時候，檢測id是否在白名單 有則踢出白名單，並count+=1。
    - 只偵測影片下半部
  + 方法說明
    - 使用白名單機制 讓同id不會被同時計數
    - 使用追蹤模式盡量不讓id有機會易主
    - 足夠近的方框機制要讓同id不會被重複計數(可能亂動一點點 就會在a b間擺動)之外，因為距離太遠的情形下在影片中碰到的frame會越遠id越有機會易主，這樣就不准了
    - 只偵測影片下半部：因為影片上半部的人物離攝影機太遠，這部分資訊沒有意義，反而拖慢推理速度跟Id易主。
    - Id易主的場景：人靠太近、接近影片邊緣，人太小等
    - 參考點以攝影機絕對看的到的地方才不會有太大擾動。
  + 仍然存在的問題
    - 如果有人刻意走回a點在走道b點 那會重複計數
    - 無法偵測反方向移動的人
    - 速度差異太大的人
* **環境搭配**
  + **建議有3060等級的顯示卡做GPU加速**
  + **Python3.9.19**
  + **pytorch 2.2.2 py3.9\_cuda12.1\_cudnn8\_0**
  + **opencv-python 4.7.0.72**
  + **ultralytics 8.1.43**
  + **其餘庫有含在readme裡**