**關於YOLO模型在Pi上透過NCS運算棒的執行，我找到兩個方案**  
  
**方案一目前能執行Tiny版的YOLOv3，但是沒有用到NCS運算棒**  
  
**方案二、三才會使用到運算棒**

**方案1(完成測試)**

可以不透過NCS運算棒成功執行YOLOv3 Tiny版本，推算一張水雉圖片的時間大約在1秒(有Bounding Box)，至於影片部分，似乎受限於Tiny的準確度關係，無法出現Bounding Box，FPS為2左右。

**方案2(還沒完成測試)**

(Intel NCS)使用Tiny YOLOv2搭配NCSDK

這個方法必須先在Pi上安裝NCSDK環境(完成)

https://github.com/duangenquan/YoloV2NCS

需將tiny-yolov2的.weights轉換成.caffemodel檔案

再將.caffemodel轉換成NCS支援的graph檔案

透過這個graph檔案進行偵測

**方案3(還沒完成測試)**

(Intel NCS)使用YoloV3 or Tiny YoloV3搭配OpenVINO

https://github.com/PINTO0309/OpenVINO-YoloV3

這個方法必須先在Pi上安裝OpenVINO環境(完成)

流程:

將yolov3 or yolov3-tiny的.weights轉換成tensorflow的.pb檔

再將.pb檔案轉換成OpenVINO支援的IR模型(FP16格式)

透過OpenVINO-YoloV3中的openvino\_yolov3\_MultiStick\_test.py進行偵測