

HW2 Report

0856149陳則佑 0856625郭毓梁 0856558李明峻

Camera Calibration

為求出物件深度，必須先得出蒐集data用相機的intrinsic matrix，利用7x5的棋盤格，由不同角度拍攝後便可輕易算出該相機的intrinsic matrix。以下為相機校正後所得之intrinsic matrix以及distortion coefficient。

```
<data>  
-4.4717322147312216e-01 2.2283292913763328e-01  
6.8860285697053150e-04 -7.1856180126169940e-04  
-6.1016350743055892e-02</data></distortion>
```

```
<data>  
9.5864930987687740e+02 0. 6.5478793324534320e+02 0.  
9.6603876479560995e+02 2.6366406139387527e+02 0. 0. 1.</data></intrinsic>
```

Object Detection

利用YOLOv3模型來偵測物件，使用DNN,FPN來達到Real-time object detection的效果，想法大概就是將整張圖片作為網路的輸入，用於直接預測bounding box座標位置，其中也包含了物體的confidence及物體的classification。

Distance Estimation

我們採用講義Homography的做法，去量照片中現實世界的距離，得到影像座標系統對應到世界座標的轉換矩陣(homography matrix)，利用這個轉換矩陣便能得知影像中距離換算回現實距離是多少。

以下公式中，(x, y)為我們找到物體所圈出來的bounding box底部中間點的影像座標，利用這個公式轉換，便能得到以相機為世界座標原點，物體的世界座標(x', y')，而我們深度則是以 $\sqrt{x'^2 + y'^2}$ 計算，即相機與物體的絕對距離。

$$s \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

我們也有嘗試使用相似三角形的方法，但出來的效果不彰，因為我們所拍攝的影片，並沒有讓鏡頭視角完美的水平於地面，這樣便會導致相似三角形方法出現極大誤差。

結果

詳細展示影片在zip檔中的result.avi，此照片為其中的截圖。



心得

這次學會了許多新的演算法，不僅了解相機內部參數和外部參數定義，也體驗到了YOLO深度網路的強大之處，快速精準的做到物體的偵測，雖然有嘗試在深度估計方面也使用深度網路的方法，但跑圖片的速度實在太慢了，也沒辦法直接跟bounding box做結合應用，一定得重跑這張照片進model，所以後來就沒繼續走這條路，而是採用傳統電腦視覺的方法，利用圖像幾何學來求深度，但是不同的方法有不同的優缺點，像是homography matrix的方法，在影像中間區域看起來都有不錯的結果，但如果物體出現影像的四周，偶爾就會有不合理的估計深度出現；相似三角形的做法也有極大的限制，只要相機視線稍微沒有水平於水平面，誤差便會很大。

經過這次實驗我們可以知道，即使用了好幾種演算法，深度的估計依然還存在著明顯的誤差，無法使用這些方法來達到商業化的水準，也因為如此才會有雙鏡頭(stereo camera)以及光達(Lidar)等更精準量測深度的技術，但這兩種成本都比較高，尤其光達雖然最為精準，但是價格非常高昂，想運用純軟體技術做到精準的深度估計，還需要努力做更多研究才行。