Team_7　R09631044 柯筆翔 R09522525 邱浩安

Paper title (proposal title):

3D position estimation from N–view images

Motivation:

近年來由於硬體與軟體設備發展，運動場上(不論是網球場還是羽球場)都有引入鷹眼系統(Hawk-Eye)來還原球於空間中的運動軌跡，並且協助裁判判定界內或者界外球，然而一套完整的鷹眼系統要價 70k USD！而鷹眼系統是由圍繞在運動場上的多台相機所組成，藉由多台相機所拍攝到的影像來獲得球於空間中的三維座標，並進一步取得運動軌跡，近年來判定之誤差已從約 3.6mm 降低至 2.6mm，這不禁讓我產生一個疑惑，「相機越多台真的能夠越準確嗎？」。因此，我想要於模擬軟體中實做鷹眼系統(Hawk-Eye)，藉由多張影像來還原球的軌跡，探討其精度極限，並進一步探討相機數量與相機架設對誤差的影響。

Procedure & Problem definition:

1. 於模擬軟體 V-rep(Virtual Robot Experiment Platform)建構實驗環境。
   環境物體包括球，多台相機以及場地。
2. 準確控制環境中的球，以獲得球的三維座標做為 ground truth。
   為了控制球於空間中的位置，我是利用 V-rep python API 來跟 python 溝通，並進一步使用 python 控制模擬環境。
3. 從模擬軟體中取得 N 台 Vision sensor 的影像。
   成功利用 python API 控制模擬環境後，亦可以取得環境中 Vision sensor 的影像。
4. 從取得的多張影像，根據 Algorithm 所述的方式來對影像進行處理，並且估測球的三維座標(為了簡化問題，假設 Vision sensors 角度與位置已知)。
5. 將結果與 ground truth 做比較，並優化演算法。
6. 探討系統之精度極限，並進一步探討相機數量與相機架設對誤差的影響。

Algorithm:

1. 於影像中找出球心：
   由於網球具有相當明顯的黃色特徵，因此首先在 RGB 色彩空間中篩選並保留黃色像素，具體做法是一一檢查影像中的每點 RGB 值，若該點 B 值<110 且 R 值>90 且 G 值>90(理想中黃色之 RGB 值為(255,255,0))則保留該點，否則設該點 RGB 值為(0,0,0)。
   接著，為了進一步取得球心之像素點位置，利用 Blob detection 並搭配恰當

的篩選參數(其中所採用的參數包刮 area, blob color, circularity, convexity & inertial，實際數值大小可以參考 source code)來找出球心於像素座標的位置。

2. 估測球的三維座標：

根據上一步驟可以得到在同一時刻每台相機拍攝到的影像中網球球心的像素位置，為了估測球的空間座標，這邊引入 Pin-hole model：

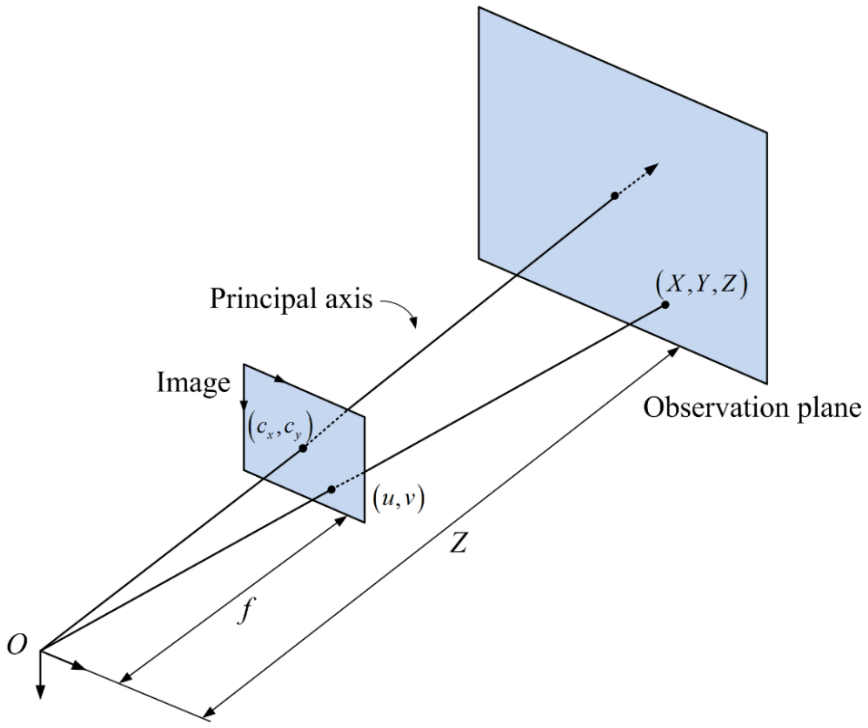
- Pin-hole model : $sm = \boldsymbol{K}M,\ m = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}, \boldsymbol{K} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}, M = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$

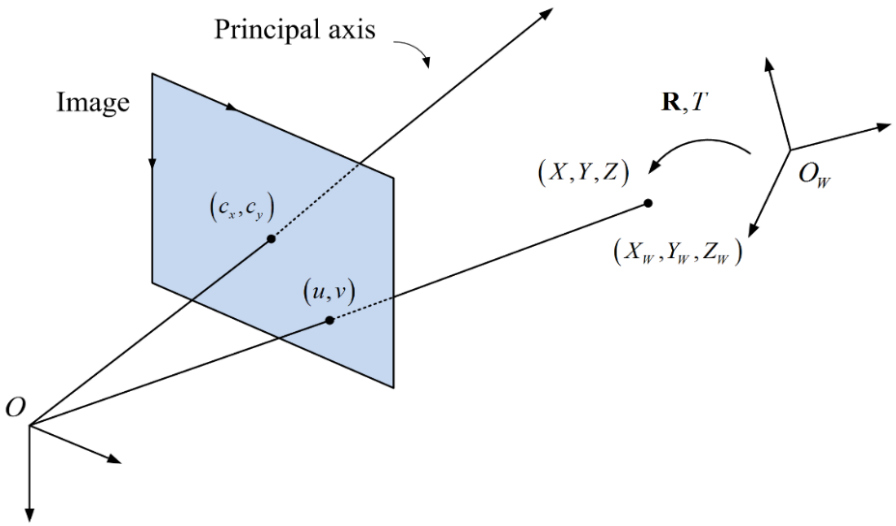
一台相機即具有一組 pin-hole model equations，其中 s 為 scale variable，m 為像素座標點位置，K 為相機內部參數，M 為球位於該相機座標系之座標。通常世界座標不會剛好與相機座標重疊，因此座標間的轉換包含一個剛體三維旋轉矩陣以及一個 translation vector：

- World coordinate ➔ Camera coordinate $M = \boldsymbol{R}W + T$

$$(R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}, W = \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix}, T = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix})$$

具體圖示如下，





若具有兩台相機，那麼就會有兩組 pin-hole model equations，即可求解球的世界座標。

- Two pin-hole models with two cameras:

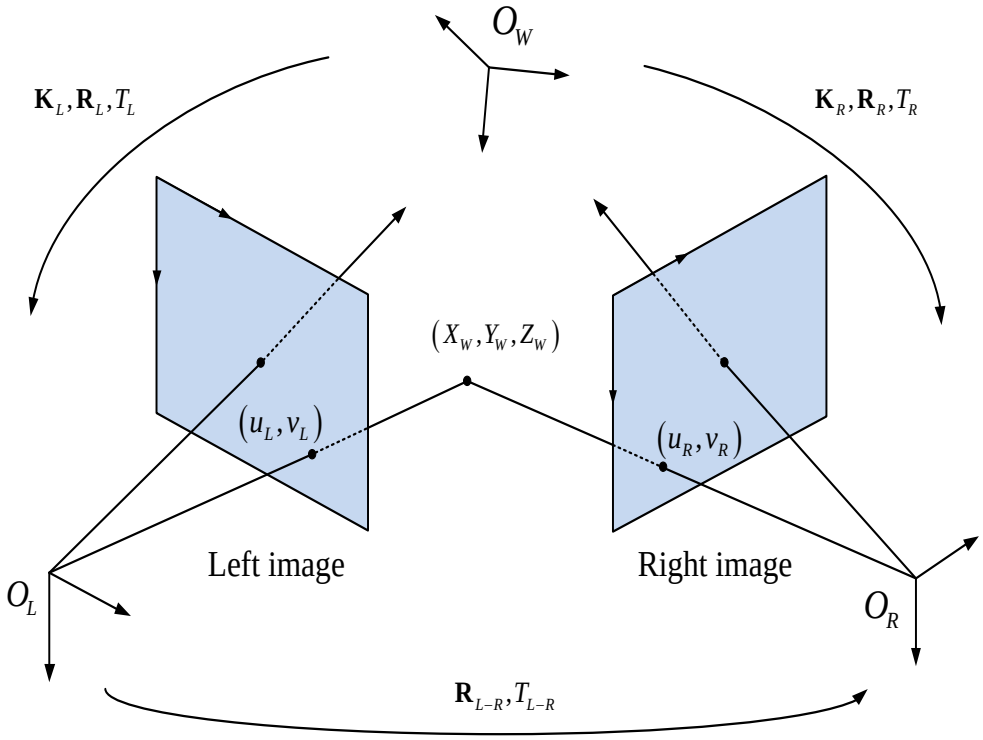
$$s\tilde{m}_L = \mathbf{K}_L(\mathbf{R}_L\tilde{W} + \tilde{T}_L)$$

$$s\tilde{m}_R = \mathbf{K}_R(\mathbf{R}_R\tilde{W} + \tilde{T}_R)$$

$$\tilde{m}_L = \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix},\ \mathbf{K}_L = \begin{bmatrix} f_{xL} & 0 & c_{xL} \\ 0 & f_{yL} & c_{yL} \\ 0 & 0 & 1 \end{bmatrix},\ \mathbf{R}_L = \begin{bmatrix} R_{11L} & R_{12L} & R_{13L} \\ R_{21L} & R_{22L} & R_{23L} \\ R_{31L} & R_{32L} & R_{33L} \end{bmatrix},\ \tilde{T}_L = \begin{bmatrix} T_{1L} \\ T_{2L} \\ T_{3L} \end{bmatrix}$$

$$\tilde{m}_R = \begin{bmatrix} u_R \\ v_R \\ 1 \end{bmatrix},\ \mathbf{K}_R = \begin{bmatrix} f_{xR} & 0 & c_{xR} \\ 0 & f_{yR} & c_{yR} \\ 0 & 0 & 1 \end{bmatrix},\ \mathbf{R}_R = \begin{bmatrix} R_{11R} & R_{12R} & R_{13R} \\ R_{21R} & R_{22R} & R_{23R} \\ R_{31R} & R_{32R} & R_{33R} \end{bmatrix},\ \tilde{T}_R = \begin{bmatrix} T_{1R} \\ T_{2R} \\ T_{3R} \end{bmatrix}$$

相關圖示如下所示：



更進一步，若有 N 台相機，那麼會有 N pin-hole models，2*N equations，而僅有 3 unknowns (X, Y, Z)也就是球的世界座標，藉由化簡且整理成線性方程組，即可利用 pseudo-inverse 去解得 least squares solution。

更加詳細說明如下：

若將 pin-hole model 寫得更加仔細，其中包含相機的三軸旋轉歐拉角，以及相機於世界座標中的位置，可以得到：

$$\begin{pmatrix} wu \\ wv \\ w \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} R_z(-\gamma) R_y(-\beta) R_x(-\alpha) \left(\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} \right)$$

$$R_z(\theta) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}, R_y(\theta) = \begin{pmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{pmatrix}, R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix}$$

(Xc,Yc,Zc) : coordinate of the camera

(X,Y,Z) : coordinate of the ball

$\alpha\beta\gamma$: rotation angle of the camera

(u,v) : image coordinate

因假設相機旋轉角度以及位置已知，所以未知數僅有 w,X,Y,Z 四個，為了方便表示且讓整體方程式更加簡潔，將旋轉矩陣以及相機內部參數乘開，並表示為 H 矩陣，若有兩台相機，那麼結果如下(其中元素上標為相機代號)：

$$\begin{pmatrix} w_0u_0 \\ w_0v_0 \\ w_0 \end{pmatrix} = \begin{pmatrix} h_{11}^0 & h_{12}^0 & h_{13}^0 \\ h_{21}^0 & h_{22}^0 & h_{23}^0 \\ h_{31}^0 & h_{32}^0 & h_{33}^0 \end{pmatrix} \left(\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - \begin{pmatrix} X_c^0 \\ Y_c^0 \\ Z_c^0 \end{pmatrix} \right),$$

$$\begin{pmatrix} w_1u_1 \\ w_1v_1 \\ w_1 \end{pmatrix} = \begin{pmatrix} h_{11}^1 & h_{12}^1 & h_{13}^1 \\ h_{21}^1 & h_{22}^1 & h_{23}^1 \\ h_{31}^1 & h_{32}^1 & h_{33}^1 \end{pmatrix} \left(\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - \begin{pmatrix} X_c^1 \\ Y_c^1 \\ Z_c^1 \end{pmatrix} \right)$$

將上述方程組全部乘開後，再將 w 進行代換，最後化簡整理成 XYZ 之線性方程組，如下：

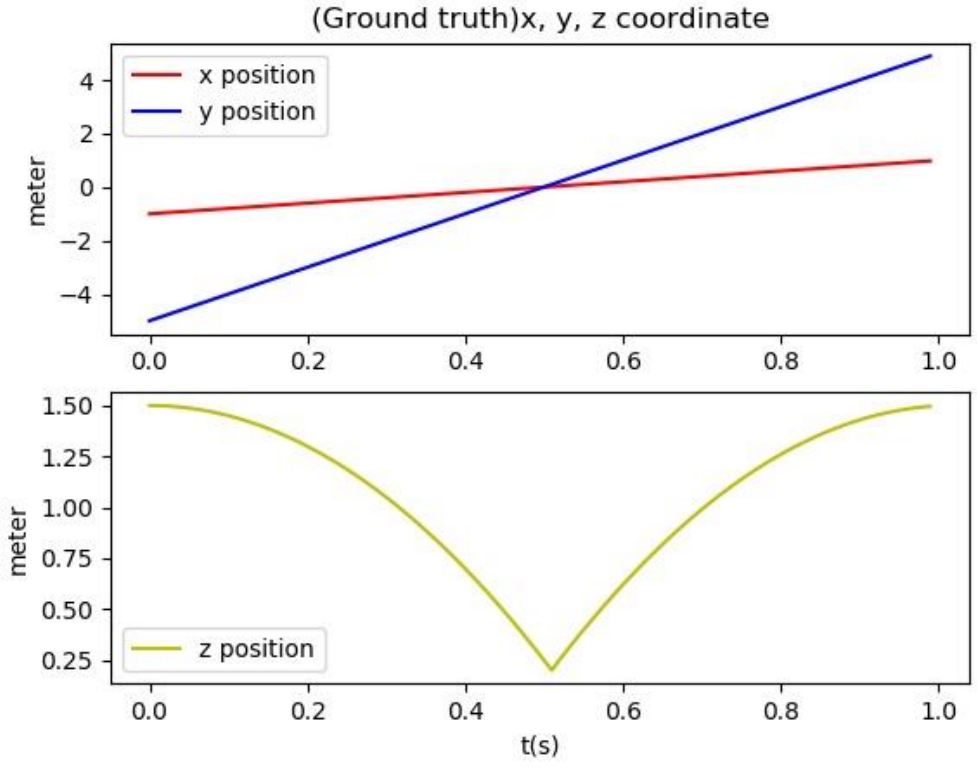
$$\begin{pmatrix} h_{11}^0 - u_0h_{31}^0 & h_{12}^0 - u_0h_{32}^0 & h_{13}^0 - u_0h_{33}^0 \\ h_{21}^0 - v_0h_{31}^0 & h_{22}^0 - v_0h_{32}^0 & h_{23}^0 - v_0h_{33}^0 \\ h_{11}^1 - u_1h_{31}^1 & h_{12}^1 - u_1h_{32}^1 & h_{13}^1 - u_1h_{33}^1 \\ h_{21}^1 - v_1h_{31}^1 & h_{22}^1 - v_1h_{32}^1 & h_{23}^1 - v_1h_{33}^1 \\ \vdots & \vdots & \vdots \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} -u_0(h_{31}^0X_c^0 + h_{32}^0Y_c^0 + h_{33}^0Z_c^0) + (h_{11}^0X_c^0 + h_{12}^0Y_c^0 + h_{13}^0Z_c^0) \\ -v_0(h_{31}^0X_c^0 + h_{32}^0Y_c^0 + h_{33}^0Z_c^0) + (h_{21}^0X_c^0 + h_{22}^0Y_c^0 + h_{23}^0Z_c^0) \\ -u_1(h_{31}^1X_c^1 + h_{32}^1Y_c^1 + h_{33}^1Z_c^1) + (h_{11}^1X_c^1 + h_{12}^1Y_c^1 + h_{13}^1Z_c^1) \\ -v_1(h_{31}^1X_c^1 + h_{32}^1Y_c^1 + h_{33}^1Z_c^1) + (h_{21}^1X_c^1 + h_{22}^1Y_c^1 + h_{23}^1Z_c^1) \\ \vdots \end{pmatrix}$$

每台相機可以提供兩條 equation，若有 N 台相機那麼會有 2*N equations，而僅有 X,Y,Z 三個未知數，利用 pseudo-inverse 來求解，即可獲得球的三維座標：

$$A\,\vec{x} = b,$$
$$\hat{x} = (A^TA)^{-1}A^Tb$$

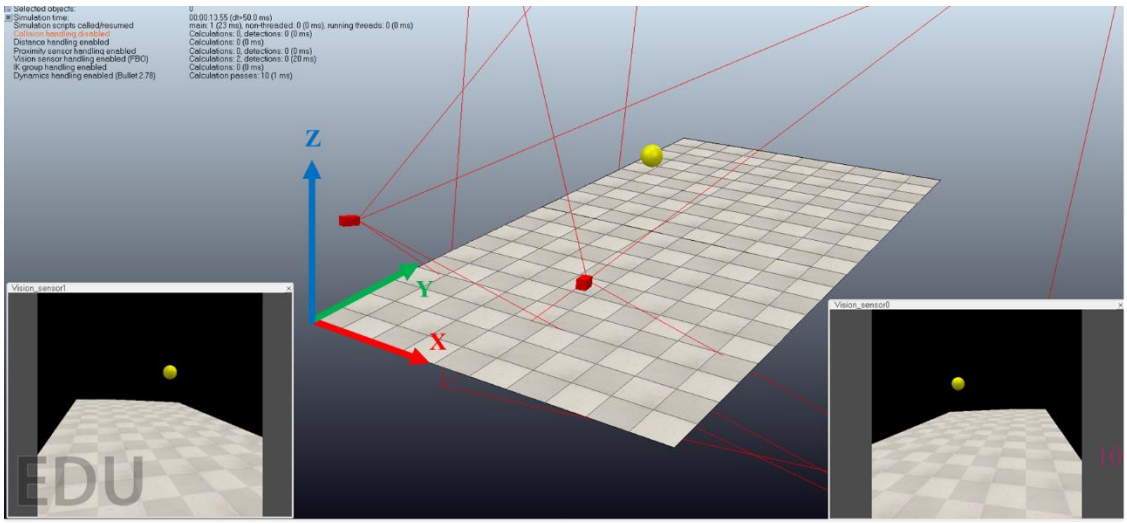
Experimental results:

(實驗一)為了控制變因，首先將測試軌跡(trajectory)統一訂定為以下路徑：

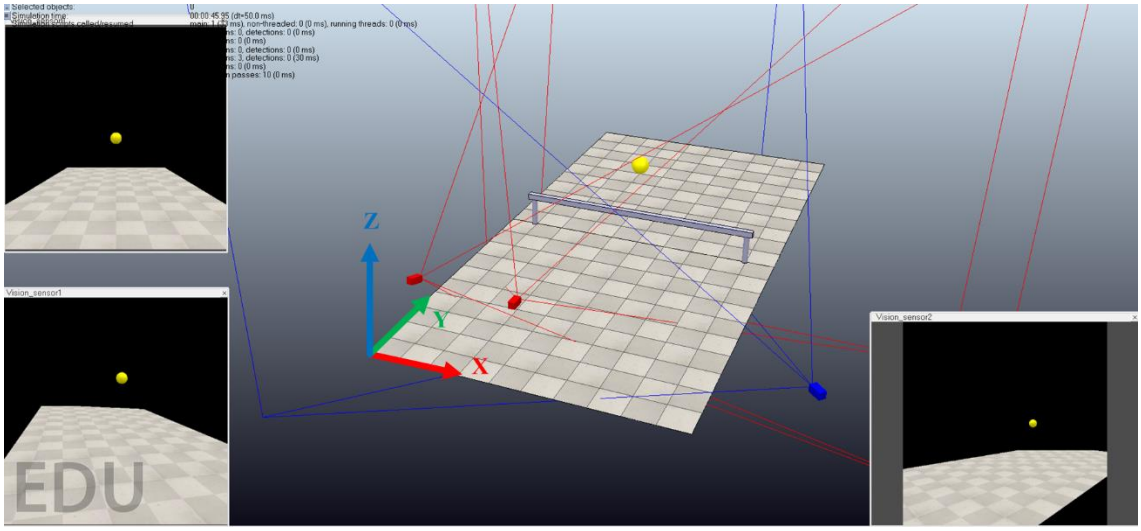


X 軸以及 Y 軸方向是做等速運動，而 Z 軸是做自由落體以及彈跳運動，是典型的水平拋射路徑。而實驗的相機數量是 2~5 台，其中個別架設方式如下：

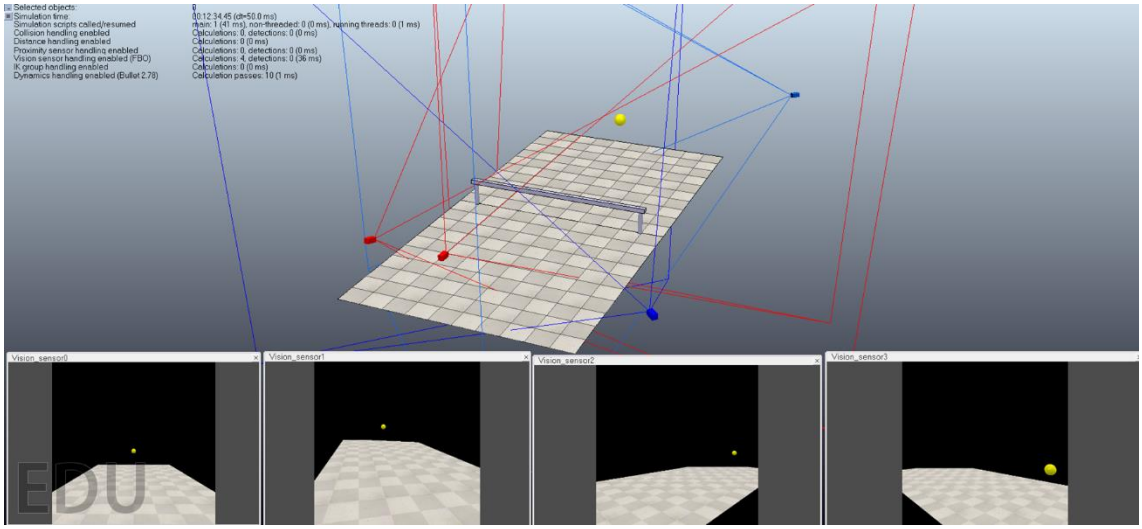
2 cameras



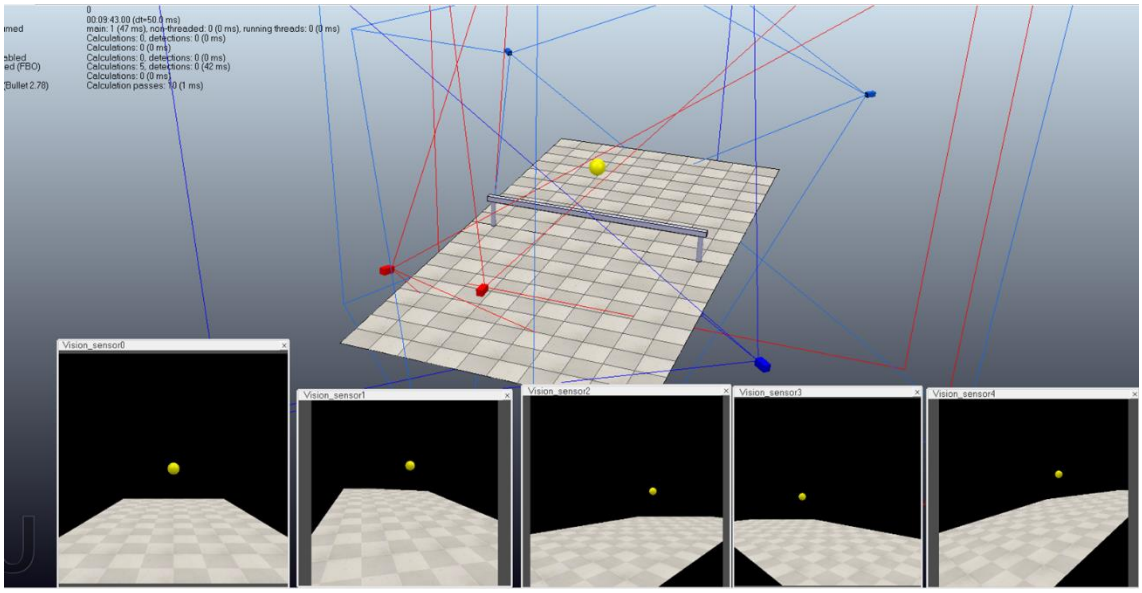
3 cameras



4 cameras



5 cameras



而誤差之計算方式為方均根誤差，以 5 cameras 之結果為例：

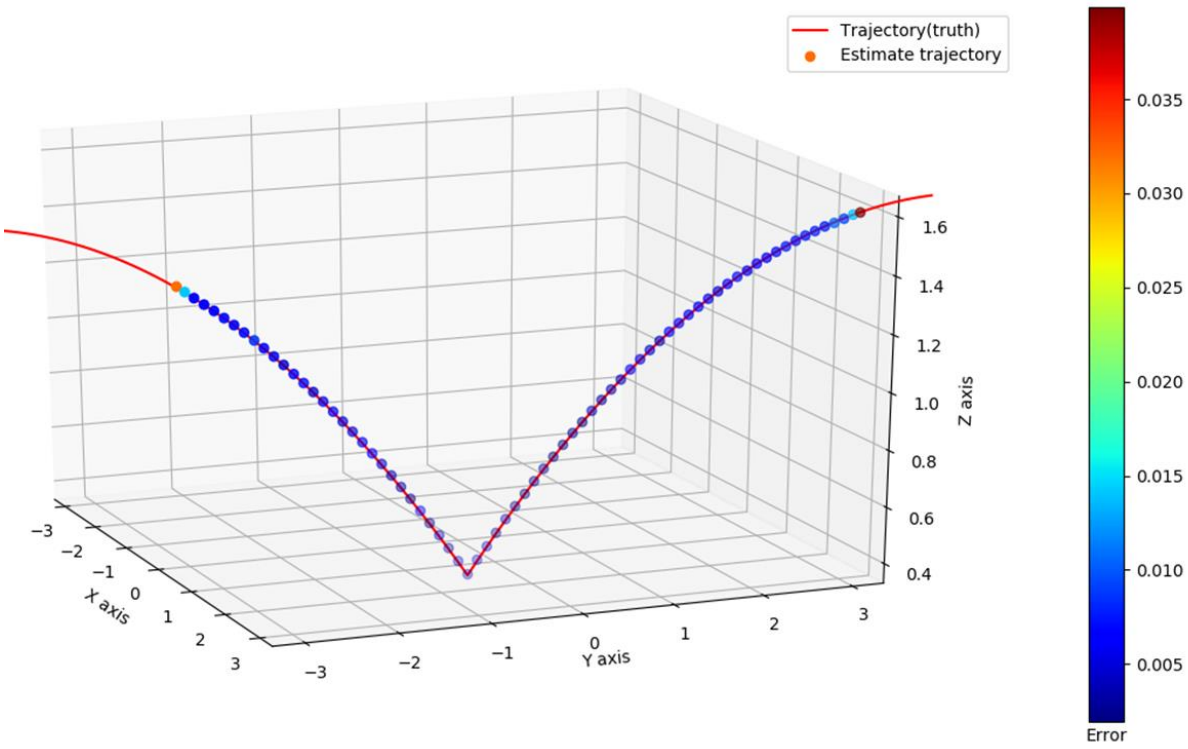
$$n = number\_of\_points$$

$$Error\_x = \sqrt{\frac{\sum_{i=1}^{n}\left(X_{estimate}^{i} - X_{truth}^{i}\right)^2}{n}} = 0.0035$$

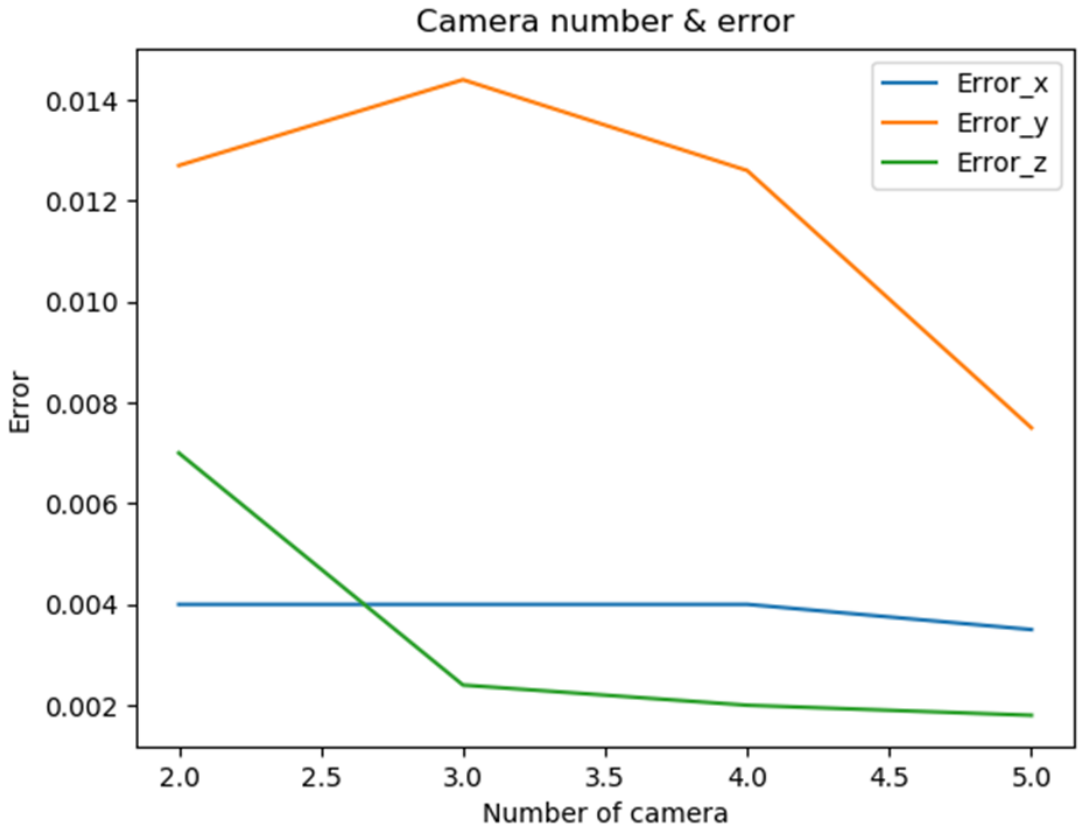
$$Error\_y = \sqrt{\frac{\sum_{i=1}^{n}\left(Y_{estimate}^{i} - Y_{truth}^{i}\right)^2}{n}} = 0.0075$$

$$Error\_z = \sqrt{\frac{\sum_{i=1}^{n}\left(Z_{estimate}^{i} - Z_{truth}^{i}\right)^2}{n}} = 0.0018$$

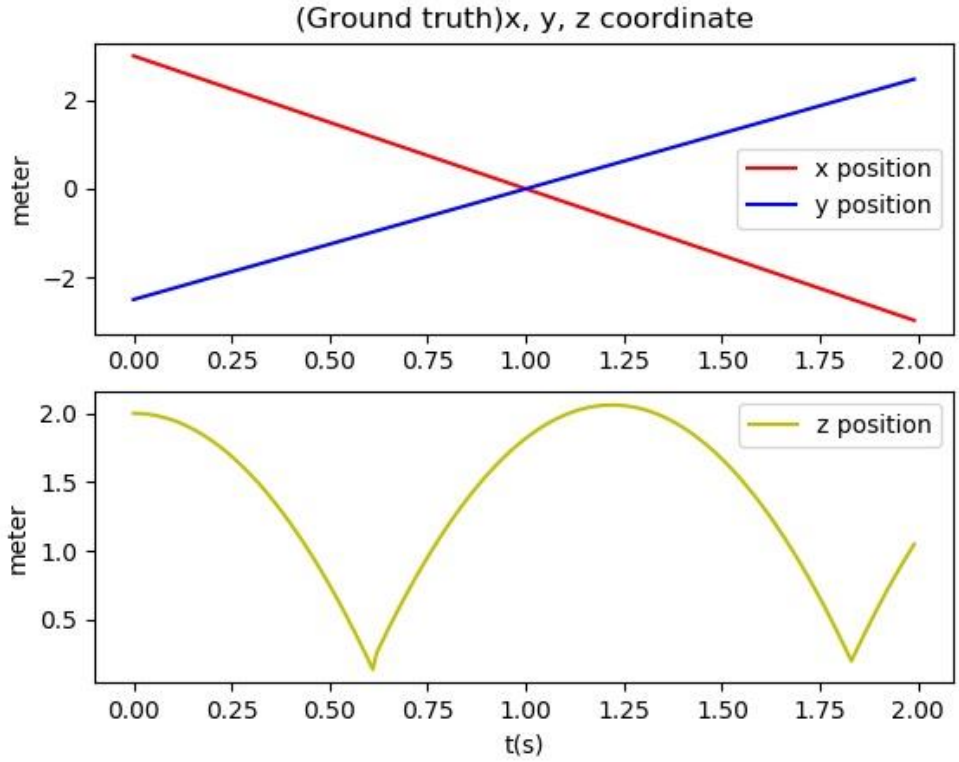
誤差是將所觀測到的每點座標值與正解(ground truth)去做相減後平方，並且相加，再除以總點數，最後開根號，而估測軌跡結果如下(以 5 cameras 為例)：



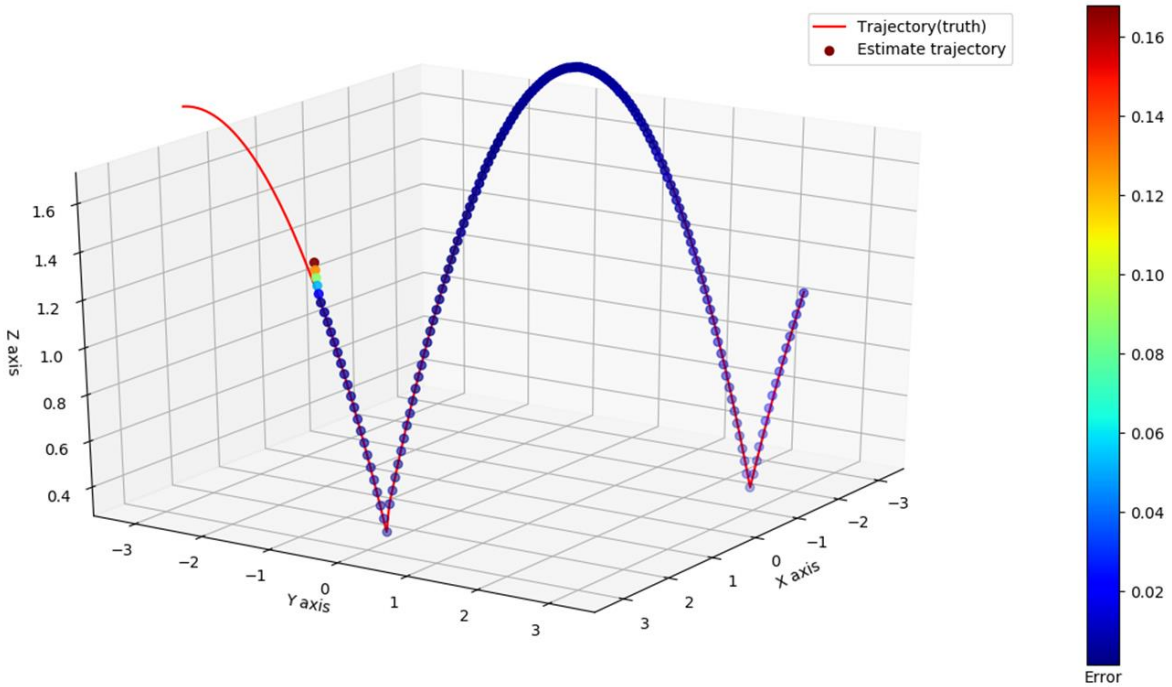
其中紅線為 ground truth 之軌跡，points 為預測之軌跡，誤差大小分布不管相機數量，基本上都差不多，都是在兩側端點處有最大誤差的發生，原因會在後續討論。而最終所得到誤差與相機數量間的關係圖如下：



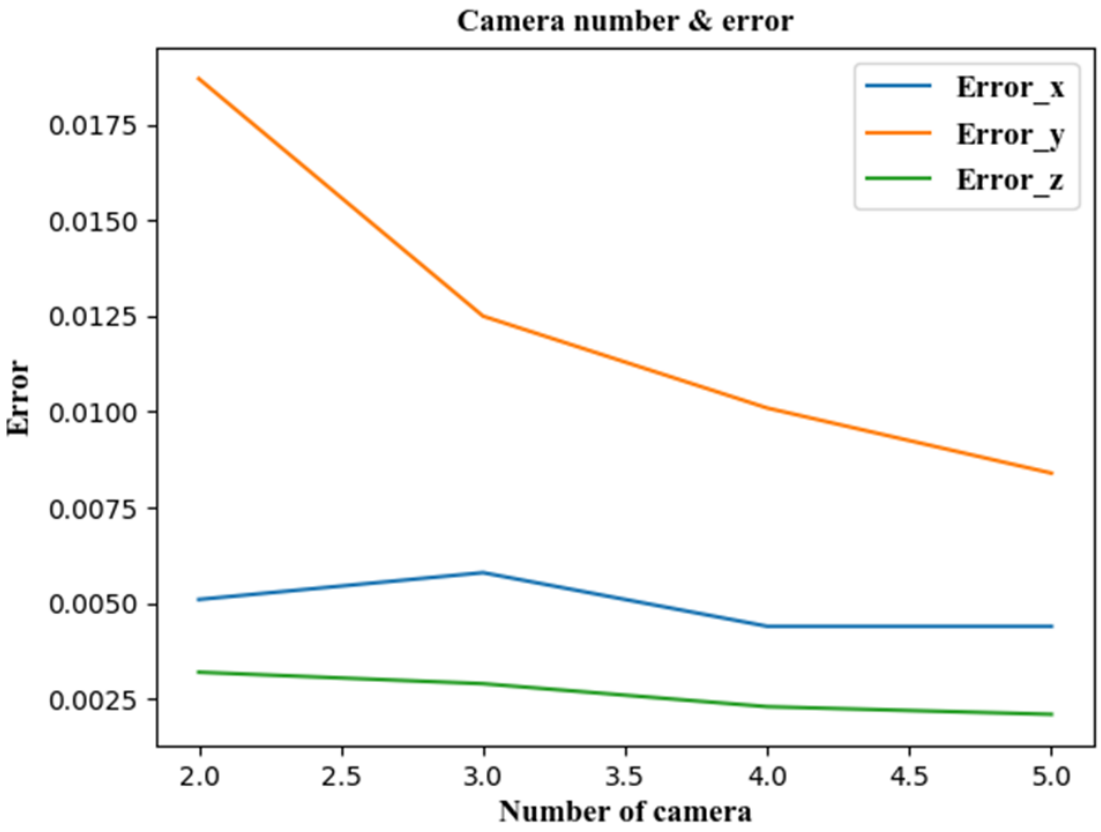
(實驗二)為了做進一步的驗證與確認，利用一樣的相關設置，但是不一樣的路徑軌跡來做測試，並觀察誤差之變化，新的軌跡設定如下圖：



在 X 軸方向以及 Y 軸方向做了速度上的調整，並且讓球做了兩次對地彈跳。以 5 cameras 之結果為例，其估測軌跡如下：

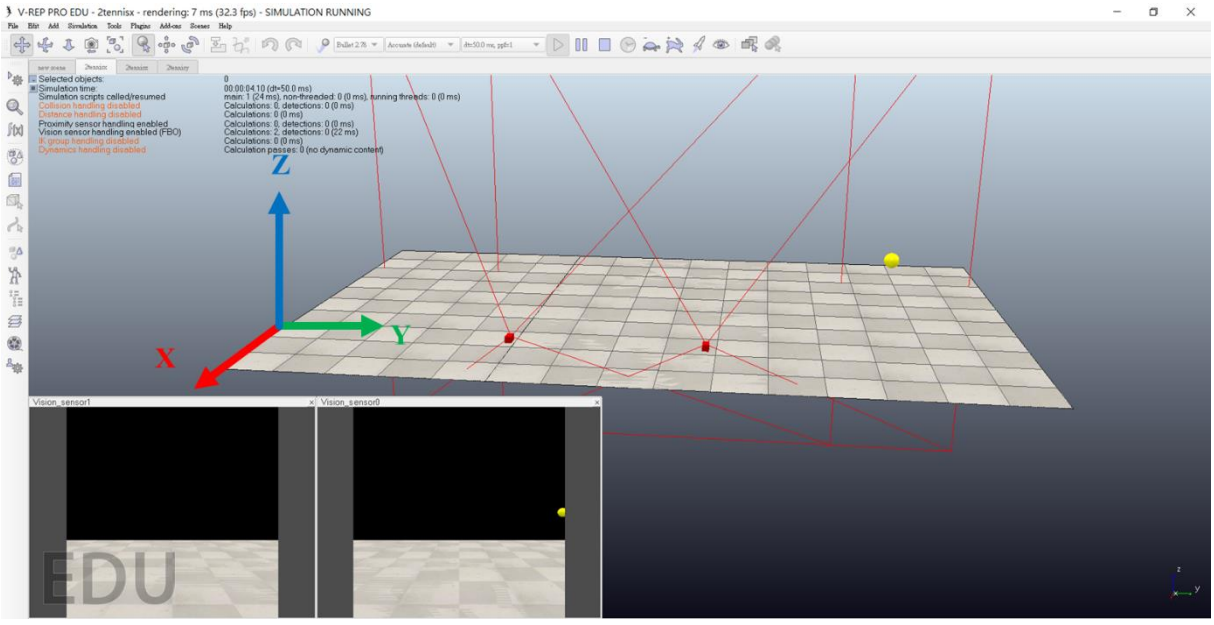


與上一個路徑類似，在端點處的誤差會來到最大，而相機數量與誤差的關係如下：



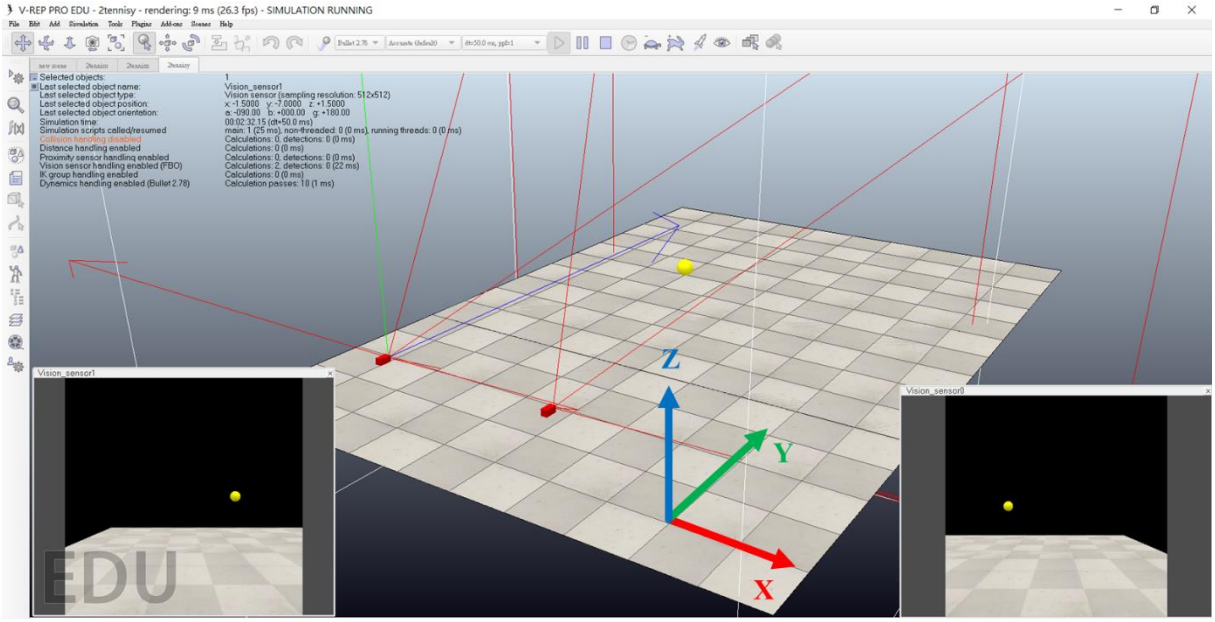
(實驗三)最後為了探討 XYZ 各軸誤差大小與相機架設間的關係，分別將相機沿著 XYZ 三軸進行拍攝，並且計算個別 error。

沿著 X 軸：



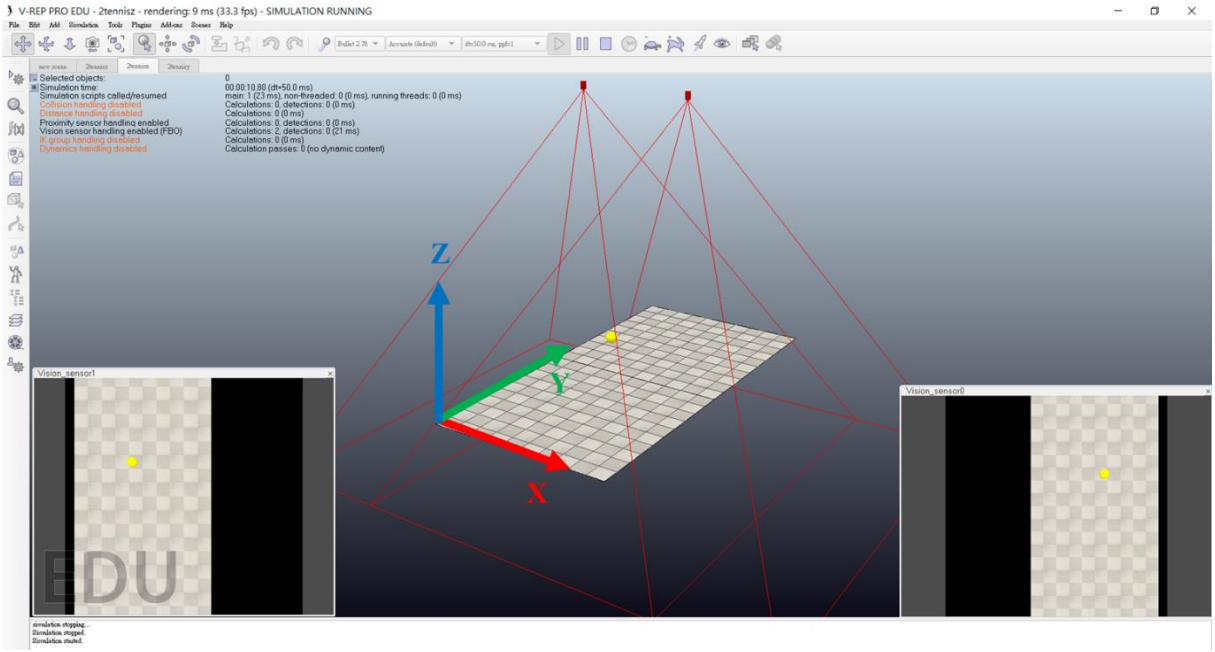
$$Error\_x = 0.0229$$
其誤差大小為，$Error\_y = 0.0008$
$$Error\_z = 0.0014$$

沿著 Y 軸：



$$Error\_x = 0.0009$$
其誤差大小為，$Error\_y = 0.0227$
$$Error\_z = 0.0013$$

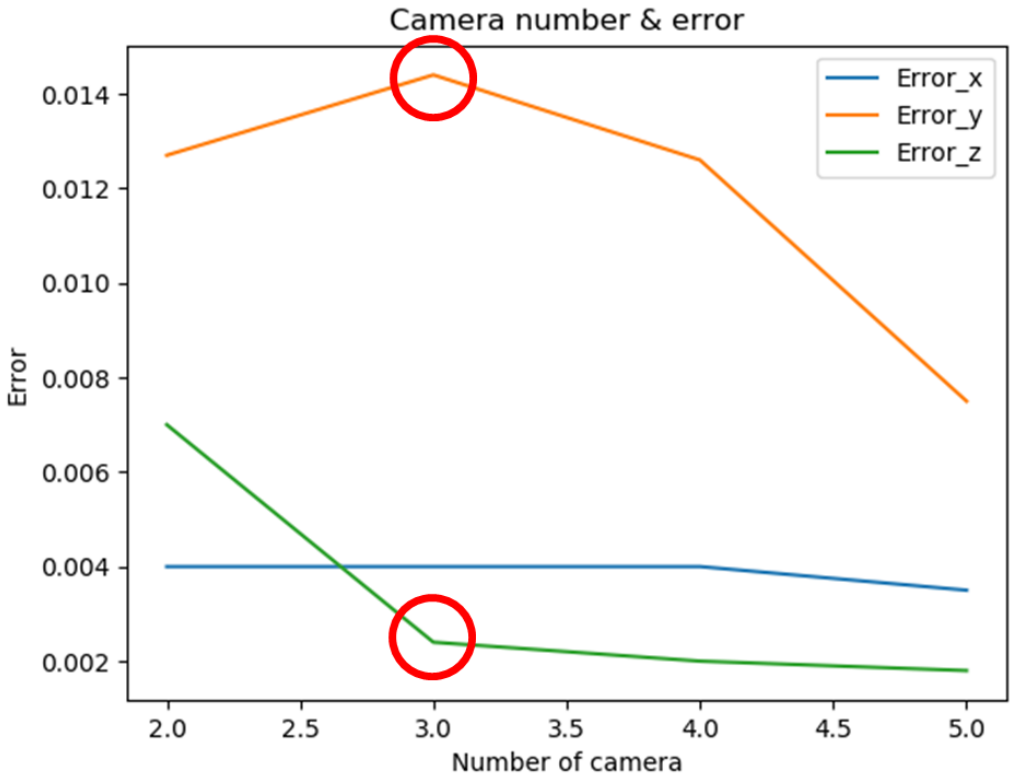
沿著 Z 軸：



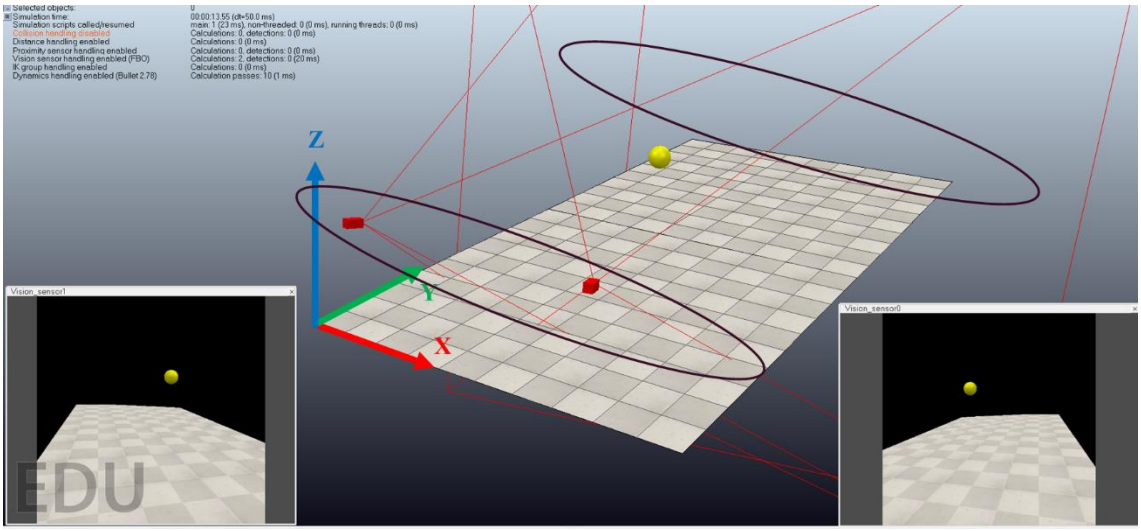
$$Error\_x = 0.0009$$
其誤差大小為，$Error\_y = 0.0012$
$$Error\_z = 0.0174$$

Discussions:

1. 根據實驗結果(不管是哪種路徑軌跡)，誤差確實有隨著相機數量上升而下降。
2. 相機的架設(包含位置與角度)非常重要，會很直接的影響各個方向的誤差，以下圖 3 cameras 的情況為例，因架設位置特殊，使得 y 方向誤差明顯上升，但是同時加強了 z 方向的準確度。



3. 追蹤球且找出球心於像素座標位置的方法也相當重要，以網球為例，可以簡單的利用 RGB space 的技巧以及 Blob detection 來找出球心，但是當球很靠近相機或是相當遠離相機時，此種演算法會有較大誤差產生(如上面所示之結果，兩端端點誤差較大)，且若是不同種球類如排球籃球等等，亦須客製化不同演算法找出準確球心，這點未來會想繼續嘗試看看。
4. 為何實驗結果中的兩種軌跡結果 y 方向的誤差都大於其他兩軸？
   (兩種軌跡之誤差與相機數量關係如上)
   這邊推論是由於相機架設的關係，可以發現實驗中的相機架設都是擺放在 x 軸上並且拍向 y 軸方向，如下圖所示：



也是為了驗證這個猜測，因此做了實驗三(沿著不同軸方向進行拍攝)，根據實驗三之結果，發現沿著哪軸拍攝，該軸之誤差就會有最大的誤差，誤差會大於其他兩軸一個 order 以上，因此可以得到以下推論：「對於相機來說，深度上的估測會有最大的誤差」，最終導向最後的結論，也就是最好的架設方式是從各個方向圍繞著運動場進行拍攝，相機越多台越好，且若希望誤差在空間中均勻分布，相機也必須均勻的分布在空間中，且拍攝方向須不盡相同。

Reference:

Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edition. Cambridge University Press, Cambridge (2004).