

# 國立交通大學

電控工程研究所

碩士論文

基於單眼視覺之擊球機器人設計與實現

Design and Implementation of Monocular Vision-based  
Ball-batting Robots

研究生：黃志誠

指導教授：蕭得聖 博士

中華民國九十九年九月

# 基於單眼視覺之擊球機器人設計與實現

Design and Implementation of Monocular Vision-based Ball-batting Robots

研 究 生：黃志誠

指 導 教 授：蕭得聖 博士

Student：Zhi-Cheng Huang

Advisor：Dr. Te-Sheng Hsiao

國立交通大學

電控工程研究所

碩士論文



Submitted to Department of Electrical and Control Engineering

College of Electrical Engineering

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

In

Electrical and Control Engineering

September 2010

Hsinchu, Taiwan, Republic of China

中華民國九十九年九月

# 基於單眼視覺之擊球機器人設計與實現

學生：黃志誠

指導教授：蕭得聖博士

國立交通大學電控工程研究所碩士班

## 摘要

本論文基於一個二軸平面的機械手臂實驗平台，再添加單眼系統使系統成為具有機器視覺的機器人。利用此單眼系統透過影像處理過程，可執行對高速運動下的特定顏色目標物進行追蹤，其中使用的影像處理方式主要為改良式背景相減法與 HSV 的顏色辨識方法，再經過一連串濾波、形態學、邊緣偵測等方式將目標物擷取出來，並且利用切換影像處理區域，來提高整體執行速度。透過校正攝影機使影像平面能與真實空間整合，並且利用轉換公式與拋物線軌跡方程式來求得目標物飛行時軌跡參數。克服硬體上速度的不足，計算出最佳擊球點位置，使機械手臂可以在旋轉最小角度的情況下便可以擊打目標物，並利用即時持續更新影像擷取點來不斷地修正目標物飛行軌跡，再透過 RS232 傳輸線將資料傳送給高效能數位訊號處理器來即時規劃機械手臂路徑，使機械手臂能在移動情況下更新路徑且不斷地修正位置來擊打目標物。

# Design and Implementation of Monocular Vision-based Ball-batting Robots

Student : Zhi-Cheng Huang

Advisor : Dr. Te-Sheng Hsiao

Department of Electrical and Control Engineering

National Chiao Tung University

## ABSTRACT

This thesis is dedicated to the integration of a monocular vision system into a two - joint robotic manipulator, such that the robot can track a target with specific color in high-speed motion. Firstly, candidates of the target are identified from the images, by a combination of an improved background subtraction method and the HSV color recognition method. Then a sequence of image processing such as filtering, morphological processing, and edge detection are applied to correctly locate the target in the images. To speed up these processes, the region of interest of the image is shrunk dynamically. Next a parabolic curve and pre-calibrated camera parameters are used to fit the trajectory of the target in the 3D space. Since the angular velocities of the robotic joints are limited, the best batting position is determined in the sense of least angular displacements of the robotic joints and is continuously updated whenever a new image is available. Through RS232 communication, the batting position is sent from the image process unit (a personal computer) to the motion controller (a high-performance DSP processor) of the robot for path planning such that the robot can hit the target at the desired position.

# 目錄

摘要 .....	i
ABSTRACT .....	ii
目錄 .....	iii
圖目錄 .....	v
表目錄 .....	vii
符號說明 .....	viii
第一章 緒論 .....	1
1.1 研究動機 .....	1
1.2 研究目的 .....	1
1.3 論文貢獻 .....	2
第二章 相關研究 .....	3
第三章 機器手臂控制與路徑規劃 .....	5
3.1 機械手臂的動力學模型 .....	5
3.2 改良型 LQR 控制器 .....	9
3.3 手臂路徑規畫 .....	13
3.3.1 手臂 D-H model .....	13
3.3.2 運動學 .....	14
3.3.3 反運動學 .....	16
3.3.4 路徑規劃 .....	18
第四章 影像擷取與追蹤 .....	21
4.1 顏色辨識 .....	21
4.1.1 RGB 法 .....	21
4.1.2 向量法 .....	23
4.1.3 HSV 法 .....	24
4.2 背景相減 .....	27
4.2.1 靜態背景模型 .....	27
4.2.2 閾值背景模型 .....	29
4.2.3 IRGABM 模型 .....	30
4.3 三角演算法 .....	31
4.4 中值濾波器 .....	31
4.5 形態學 .....	32
4.5.1 膨脹 .....	32
4.5.2 侵蝕 .....	33
4.6 影像擷取流程 .....	33

4.7 ROI(region of interest)選擇 .....	35
第五章 校正與最佳擊球點 .....	36
5.1 校正 .....	36
5.1.1 幾何相機模型 .....	36
5.1.2 幾何相機校正 .....	40
5.2 預測物體飛行軌跡 .....	42
5.3 影像擷取點數策略 .....	44
5.4 最佳擊球點 .....	46
第六章 實驗結果與分析 .....	53
6.1 硬體設施與流程 .....	53
6.2 座標系統 .....	54
6.3 機械手臂路徑規畫實驗 .....	55
6.4 影像處理實驗 .....	57
6.5 校正實驗 .....	62
6.6 機械手臂速度實驗 .....	63
6.7 實際擊球實驗 .....	65
6.8 實驗結果討論 .....	70
第七章 結論與未來展望 .....	72
7.1 結論 .....	72
7.2 未來展望 .....	73
參考文獻 .....	74



# 圖目錄

圖 3.1 兩軸平面機械手臂示意圖 .....	5
圖 3.2 直流馬達示意圖 .....	7
圖 3.3 不需要速度估測的控制器方塊圖 .....	10
圖 3.4 增大的受控廠 .....	11
圖 3.5 機械手臂 D-H model .....	14
圖 3.6 手臂幾何示意圖 .....	17
圖 3.7 手臂正反手示意圖 .....	17
圖 3.8 路徑規劃移動比較圖 .....	19
圖 3.9 路徑規劃流程圖 .....	20
圖 4.1 彩色影像 .....	22
圖 4.2 經過閾值後影像 .....	22
圖 4.3 AND 後二元影像 .....	22
圖 4.4 向量法示意圖 .....	23
圖 4.5 向量法處理後二元影像 .....	24
圖 4.6 HSV 示意圖 .....	24
圖 4.7 HSV 灰階影像 .....	25
圖 4.8 H 與 S AND 後二元影像 .....	25
圖 4.9 亮度不同時原始影像 .....	26
圖 4.10 改變場景後，RGB 法(左)與 HSV 法(中)與向量法(右).....	26
圖 4.11 改變場景及亮度後，RGB 法(左)與 HSV 法(中)與向量法(右).....	27
圖 4.12 灰階背景(左)與二元背景(右).....	28
圖 4.13 原始影像(左)與背景相減後結果(右).....	28
圖 4.14 震動後原始影像(左)與背景相減結果(右).....	28
圖 4.15 造成殘影結果 .....	29
圖 4.16 IRGABM 背景相減(左)與持續 3 秒後(中)與延遲 10 秒後(右).....	30
圖 4.17 三角演算法示意圖 .....	31
圖 4.18 集合 A(左)與結構元素 B(中)與膨脹後結果(右).....	32
圖 4.19 集合 A(左)與結構元素 B(中)與侵蝕後結果(右).....	33
圖 4.20 影像擷取流程圖 .....	34
圖 4.21 ROI 選擇流程圖.....	35
圖 5.1 針孔投影模型 .....	36
圖 5.2 距離對成像的影像 .....	37
圖 5.3 物體投影示意圖 .....	37
圖 5.4 實際相機內部座標系統 .....	38
圖 5.5 拋物線分量座標示意圖 .....	42

圖 5.6 物體飛行時間示意圖 .....	47
圖 5.7 手臂實作時原點平移圖 .....	48
圖 5.8 $\theta_1, \theta_2$ 平面 .....	50
圖 5.9 $h(\theta_1, \theta_2)$ 函式之角度代入 2-norm 結果 .....	50
圖 5.10 2-norm 與時間關係圖 .....	51
圖 5.11 真實空間擊打模擬圖 .....	52
圖 6.1 硬體設施 .....	53
圖 6.2 座標系統(正視圖) .....	54
圖 6.3 座標系統(側視圖) .....	55
圖 6.4 路徑規劃 單點模擬圖(左)與實際數據(右).....	56
圖 6.5 路徑規劃 多點模擬圖(左)與實際數據(右).....	56
圖 6.6 實驗原始背景 .....	58
圖 6.7 實驗原始影像 .....	58
圖 6.8 實驗 IRGABM 法.....	59
圖 6.9 實驗 HSV 法.....	59
圖 6.10 IRGABM 與 HSV And 後影像.....	59
圖 6.11 實驗中值濾波器濾波後影像 .....	60
圖 6.12 實驗膨脹後影像 .....	60
圖 6.13 實驗追蹤目標物影像 .....	60
圖 6.14 連續追蹤目標物影像 .....	61
圖 6.15 校正選取點 .....	62
圖 6.16 手臂速度測試 大軸(左)與小軸(右).....	63
圖 6.17 機械手臂 0.3 秒可移動範圍 .....	64
圖 6.18 擊球實驗 1 .....	69
圖 6.19 擊球實驗 2 .....	69
圖 6.20 擊球實驗 3 .....	70

## 表目錄

表 1 持續更新擷取點法 .....	45
表 2 機械手臂硬體參數 .....	55
表 3 距離與 pixel 關係表 .....	58
表 4 校正結果 .....	62
表 5 丟球實驗 1 .....	65
表 6 丟球實驗 2 .....	65
表 7 丟球實驗 3 .....	65
表 8 丟球實驗 4 .....	66
表 9 丟球實驗 5 .....	66
表 10 丟球實驗 6 .....	66
表 11 丟球實驗 6 .....	67
表 12 丟球實驗 7 .....	67
表 13 丟球實驗 8 .....	68



## 符號說明

$(x_a, y_a, z_a)$	機械手臂座標系
$(x, y, z)$	真實空間座標系
$P_{z_a}$	機械手臂座標系中 z 方向
$P_{y_a}$	機械手臂座標系中 y 方向
$P$	真實空間座標系中座標位置
$p$	影像平面中座標位置
$u$	影像平面中橫向座標位置
$v$	影像平面中縱向座標位置
$u_0$	影像平面中橫向座標原點位置
$v_0$	影像平面中縱向座標原點位置
$\theta_1$	機械手臂大軸角度
$\theta_2$	機械手臂小軸角度
$\theta_{10}$	機械手臂大軸初始角度
$\theta_{20}$	機械手臂小軸初始角度
$R$	RGB 格式中紅色的主要頻譜成分
$G$	RGB 格式中綠色的主要頻譜成分
$B$	RGB 格式中藍色的主要頻譜成分
$H$	HSV 格式中色調成分
$S$	HSV 格式中飽和度成分
$V$	HSV 格式中亮度成分
$B_{x,y}^k$	第 k 張影像中 x, y 座標的背景灰階值
$A \oplus B$	A 被 B 膨脹
$A \ominus B$	A 被 B 侵蝕
$f$	焦距
$M$	校正矩陣
$t_i$	第 i 個擷取點時間
$(x_i, y_i, z_i)$	第 i 個擷取點在真實空間座標系位置

# 第一章 緒論

## 1.1 研究動機

機器人自 1970 年開始，已大量運用於工業製造生產上，例如：汽車、機械與半導體等產業。經過數十年的科技發展，智慧型機器人的感測、認知、控制與通訊技術愈趨成熟，能夠與人類互動的智慧型機器人將不再是一個夢想。

機器人運動控制的設計與分析部分，已達到成熟的階段。故加入機器視覺部分，可增加機器人的功能性，甚至可做到與人類互動。在機器視覺領域中，也有許多優秀的論文被發表，然而大部份都使用昂貴的高速攝影機與雙眼系統來進行影像處理及距離估測，以及可高速移動的機械手臂實驗平台來呈現，如何使用市面上普遍可得的 USB 視訊鏡頭，加上單眼系統來擷取高速飛行下的物體影像，進而預測物體在 3D 空間中位置，並達到相同效果將是一個值得研究的問題。

## 1.2 研究目的

使用機械手臂擊球的方式來呈現機器視覺與機械手臂的整合。利用單眼機器視覺系統捕捉 3D 空間中快速飛行球體，並計算出物體飛行軌跡，再透過已建構之二軸平面機械手臂進行路徑規畫，尋找工作區內最佳打擊位置進行擊球。

### 1.3 論文貢獻

本論文貢獻為在已建立機器手臂平台上，進一步增加單眼機器視覺系統，使用市面上普遍看到的視訊攝影機經過影像處理技術達到追蹤高速物體的功能，並整合機械手臂，透過高效能的數位訊號處理器，規畫出最佳手臂路徑，使機器手臂在未來能夠有更多的發展空間。

實際進行即時投擲實驗，利用影像處理，可在複雜或不同背景下追蹤物體，並選取特定區域，減低影像處理範圍，進而提升程式計算速度，求出球體飛行軌跡參數，再經由增加擷取點，不斷地修正飛行拋物線軌跡，並找出機器手臂最少動作下可到達的最佳擊球位置。



## 第二章 相關研究

在現今世界上有許多對機器視覺有深入的研究，其中整合機械手臂來執行擊球動作的相關研究非常廣泛，如[10][14]使用雙眼視覺系統及高速機械手臂進行接球或擊球的工作，在實現成果上已達一定的成效，成功機率大約在 4 成~6 成 5 之間，也有人使用 Neural Network 的方式[23]來設計擊球所需之控制器，並且透過不斷地學習機制來擊球。但卻只能對某速度下進行擊球。

而在鑽研機械手臂擊球這一個研究當中，還包含了許多相關的技術。假設要進行擊球實驗，首當其衝的便是需要把目標物偵測出來。例如使用影像處理的方是將目標物擷取出來，其中有人採用背景相減的方式，如[16]中所分析的各種背景模型，包含 RGABM 法、Eigen-Background 模型、MoG 模型、KDE 模型以及作者所提出的 IRGABM 法，並且偵測陰影使目標物能更準確的被擷取出來。另外也有其他研究採用不使用顏色資訊針對某種形狀將目標物擷取出來[18]，或是使用形狀與顏色配合將複雜的足球比賽場景的足球擷取出來[17]。也有在複雜的顏色環境中，將欲選取之目標物擷取出來[15]。

在擷取目標物過程中，可能會需要設定閾值大小，[5] [6] [13][25]中提出了幾種不同的方式，包含 Background-symmetry algorithm、Isodata algorithm、triangle algorithm 幾種方式可以分割影像。使閾值可以呈現動態的方式選取。

欲擊打某目標，不可或缺的即是對距離的估測，而在估測距離之前，須先對攝影機進行校正工作，關於校正便有許多探討空間，如[2] [3] [19]中所提到的自動校正的方式，或是針對攝影機失真問題，該如何解決皆有明確說明。校正完畢後，即是進行距離的估測，估測方式如[1] [26]僅使用單眼視覺系統來對環境的估測方式，或是採用立體視覺系

統[11] [12]來進行估測。

介紹完擊球機器人所需的相關技術後，對於[10][14]皆使用雙眼視覺系統達到實現的效果，假使採用單眼視覺系統，即可使硬體造價上節省相當大的經費，其中[10]在距離估測上還仰賴雙眼視覺系統內部偵測到的距離，而[14]則是在硬體架構上，不論是機械手臂，或是攝影機，都是特製的超高速產品，似乎在研究上非常仰賴硬體設施，因此本論文將針對此問題嘗試使用造價低廉又垂手可得的視訊鏡頭，並且只利用單眼系統來估測 3D 空間中位置。在一般的硬體設備下，擊中高速飛行的物體。



## 第三章 機器手臂控制與路徑規劃

本章介紹機械手臂部分，主要分為兩大部分，第一部分為已建構完成的兩軸機械手臂系統，包含動力學與馬達模型，並將兩個模型歸納整理出完整的機器手臂動態模型，以及所使用的改良型 LQR 控制器介紹。第二部分則是為了使機械手臂能夠依照預想的揮動軌跡去擊打球，所以必需考量手臂的路徑規劃，其中還包括相關的 DH 模型、運動學及反運動學。

### 3.1 機械手臂的動力學模型

兩軸平面機械手臂如圖 3.1 所示，在本論文中稱 Link1 為大軸，Link2 為小軸。此座標原點位於大軸基座，水平  $z_a$  軸在真實空間中為縱深，垂直  $y_a$  軸在真實空間為高低。手臂歸零初始位置為  $-90$  度，也就是  $-y_a$  軸的位置。此種座標定義方式稱為手臂座標系。

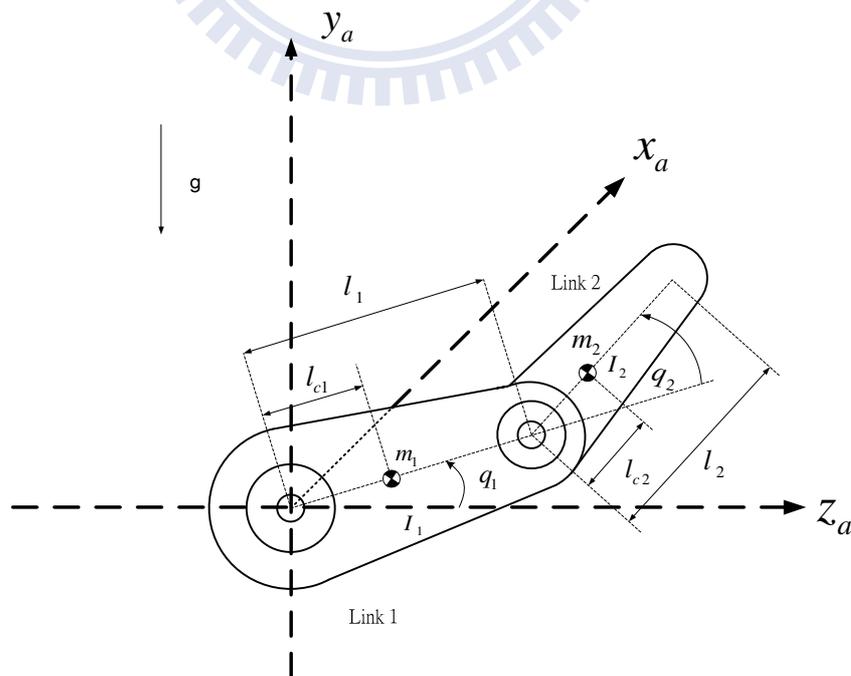


圖 3.1 兩軸平面機械手臂示意圖

由 Lagrangian Equation[8]可得到機械手臂的動力學方程式：

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}+\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\mathbf{G}(\mathbf{q})+\mathbf{B}(\dot{\mathbf{q}})=\boldsymbol{\tau} \quad (3.1.1)$$

$\mathbf{M}(\mathbf{q})$  為手臂的質量慣性矩陣， $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$  為向心力及柯氏力項矩陣， $\mathbf{G}(\mathbf{q})$  為重力項向量， $\mathbf{B}(\dot{\mathbf{q}})$  為摩擦力項向量， $\boldsymbol{\tau}$  為馬達傳至兩軸上的扭力。此處假設摩擦力  $\mathbf{B}(\dot{\mathbf{q}})$  為庫倫摩擦力與黏滯摩擦力，黏滯摩擦力為黏滯摩擦力係數  $\mathbf{b}$  乘上角速度，庫倫摩擦力為庫倫摩擦力係數  $\mathbf{f}_c$  乘上角速度的方向性，摩擦力係數皆設為常數。

其詳細函數如下：

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2) + I_1 + I_2 & m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2 \\ m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2 & m_2 l_{c2}^2 + I_2 \end{bmatrix}$$

$$\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \begin{bmatrix} -m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2 & -m_2 l_1 l_{c2} \sin(q_2) (\dot{q}_1 + \dot{q}_2) \\ m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 & 0 \end{bmatrix}$$

$$\mathbf{G}(\mathbf{q}) = \begin{bmatrix} (m_1 l_{c1} + m_2 l_1) g \cos(q_1) + m_2 l_{c2} g \cos(q_1 + q_2) \\ m_2 l_{c2} g \cos(q_1 + q_2) \end{bmatrix}$$

$$\mathbf{B}(\dot{\mathbf{q}}) = \begin{bmatrix} \mathbf{b}_1 \dot{q}_1 + \mathbf{f}_{c1} \text{sign}(\dot{q}_1) \\ \mathbf{b}_2 \dot{q}_2 + \mathbf{f}_{c2} \text{sign}(\dot{q}_2) \end{bmatrix}$$

在此系統中，角度  $\mathbf{q}$  與扭力  $\boldsymbol{\tau}$  為可量測之已知狀態，角速度  $\dot{\mathbf{q}}$  及角加速度  $\ddot{\mathbf{q}}$  為未知狀態。動力學模型的動態方程式可表示成下列型式：

$$\frac{d}{dt} \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \dot{\mathbf{q}}_1 \\ \dot{\mathbf{q}}_2 \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{q}}_1 \\ \dot{\mathbf{q}}_2 \\ \mathbf{M}(\mathbf{q}_1)^{-1} [\boldsymbol{\tau}(t) - \mathbf{C}(\mathbf{q}_1, \dot{\mathbf{q}}_1) \dot{\mathbf{q}}_1 - \mathbf{G}(\mathbf{q}_1)] \\ \mathbf{M}(\mathbf{q}_2)^{-1} [\boldsymbol{\tau}(t) - \mathbf{C}(\mathbf{q}_2, \dot{\mathbf{q}}_2) \dot{\mathbf{q}}_2 - \mathbf{G}(\mathbf{q}_2)] \end{bmatrix} \quad (3.1.2)$$

接著考慮馬達的部分。此兩軸機械手臂實驗系統的扭力是由兩顆直流馬達所供給，為了使控制更加準確，必須要將馬達的物理參數一併考慮進動力學模型之中。

典型的直流馬達驅動系統如圖 3.2 所示。

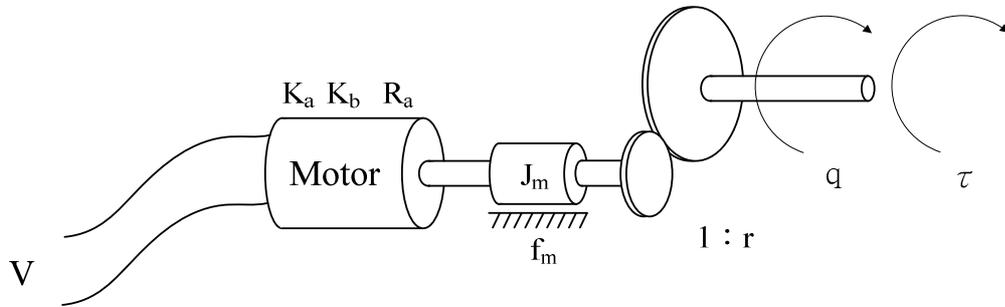


圖 3.2 直流馬達示意圖

一般可由二階線性微分方程式來表示其電氣機械等效電路模型，進而導出馬達電樞的輸入電壓與輸出力矩之間的動態方程式，如式 (3.1.1) 所示[8]。

$$J_m \ddot{q} + f_m \dot{q} + \frac{K_a K_b}{R_a} \dot{q} + \frac{\tau}{r^2} = \frac{K_a}{r R_a} v \quad (3.1.1)$$

若考慮此系統為兩軸機械臂，則可由(3.1.3)得

$$J \ddot{q} + F \dot{q} + \frac{\tau}{r^2} = K v \quad (3.1.2)$$

其中

$J_m$ : 轉子的轉動慣量( $kg \cdot m^2$ )

$K_a$ : 馬達力矩常數( $N \cdot m/A$ )

$R_a$ : 電樞電阻( $\Omega$ )

$K_b$ : 反電動勢常數( $V \cdot s/rad$ )

$f_m$ : 轉子的摩擦係數( $N \cdot m$ )

$\tau$ : 在齒輪組後的負載軸力矩( $N \cdot m$ ), 即機械手臂力矩

$q$ : 負載軸的角度( $rad$ )

$r$ : 齒輪減速比率

$v$ : 電樞電壓( $V$ )

其中

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_{m1} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_{m2} \end{bmatrix}$$

$$\mathbf{F} = \begin{bmatrix} \mathbf{f}_{m1} + \left( \frac{\mathbf{K}_a \mathbf{K}_b}{\mathbf{R}_a} \right) & \mathbf{0} \\ \mathbf{0} & \mathbf{f}_2 + \left( \frac{\mathbf{K}_a \mathbf{K}_b}{\mathbf{R}_a} \right) \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{f}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{f}_2 \end{bmatrix}$$

$$\mathbf{K} = \begin{bmatrix} \left( \frac{\mathbf{K}_a}{\mathbf{R}_a} \right) \frac{1}{r_1} & \mathbf{0} \\ \mathbf{0} & \left( \frac{\mathbf{K}_a}{\mathbf{R}_a} \right) \frac{1}{r_1} \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{K}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 \end{bmatrix}$$

將(3.1.4)式帶入(3.1.1)式後就可得到完整機械手臂動力學模型，如(3.1.5)式所示。

$$\left( \frac{\mathbf{M}(\mathbf{q}) + \mathbf{r}^2 \mathbf{J}}{\mathbf{r}^2 \mathbf{K}} \right) \ddot{\mathbf{q}} + \frac{\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})}{\mathbf{r}^2 \mathbf{K}} \dot{\mathbf{q}} + \frac{\mathbf{G}(\mathbf{q})}{\mathbf{r}^2 \mathbf{K}} + \frac{\mathbf{F}(\dot{\mathbf{q}})}{\mathbf{r}^2 \mathbf{K}} + \mathbf{B} \dot{\mathbf{q}} = \mathbf{v} \quad (3.1.3)$$

最後將(3.1.5)重寫，得到(3.1.6)式的形式。

$$\bar{\mathbf{M}}(\mathbf{q}) \ddot{\mathbf{q}} + \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \bar{\mathbf{G}}(\mathbf{q}) + \bar{\mathbf{F}}(\dot{\mathbf{q}}) + \bar{\mathbf{B}} \dot{\mathbf{q}} = \mathbf{v} \quad (3.1.4)$$

其中

$$\bar{\mathbf{M}}(\mathbf{q}) = \begin{bmatrix} \frac{1}{\mathbf{K}_1 r_1^2} (m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2) + I_1 + I_2) + \mathbf{J}_{m1} & \frac{1}{\mathbf{K}_1 r_1^2} (m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2) \\ \frac{1}{\mathbf{K}_2 r_2^2} (m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2) & \frac{1}{\mathbf{K}_2 r_2^2} (m_2 l_{c2}^2 + I_2) + \mathbf{J}_{m2} \end{bmatrix}$$

$$\bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \frac{1}{\mathbf{K}_1 r_1^2} (-m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2) & \frac{1}{\mathbf{K}_1 r_1^2} (-m_2 l_1 l_{c2} \sin(q_2) (\dot{q}_1 + \dot{q}_2)) \\ \frac{1}{\mathbf{K}_2 r_2^2} (m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1) & \mathbf{0} \end{bmatrix}$$

$$\bar{\mathbf{G}}(\mathbf{q}) = \begin{bmatrix} \frac{1}{\mathbf{K}_1 r_1^2} [(m_1 l_{c1} + m_2 l_1) g \cos(q_1) + m_2 l_{c2} g \cos(q_1 + q_2)] \\ \frac{1}{\mathbf{K}_1 r_1^2} (m_2 l_{c2} g \cos(q_1 + q_2)) \end{bmatrix}$$

$$\bar{\mathbf{B}}\dot{\mathbf{q}} = \begin{bmatrix} \frac{1}{\mathbf{K}_1 r_1^2} (b_1 \dot{q}_1 + r_1^2 f_1) \\ \frac{1}{\mathbf{K}_2 r_2^2} (b_2 \dot{q}_2 + r_2^2 f_2) \end{bmatrix}$$

$$\mathbf{F}(\dot{\mathbf{q}}) = \begin{bmatrix} \frac{f_{c1} \text{sign}(\dot{q}_1)}{\mathbf{K}_1 r_1^2} \\ \frac{f_{c2} \text{sign}(\dot{q}_2)}{\mathbf{K}_2 r_2^2} \end{bmatrix}$$

### 3.2 改良型 LQR 控制器

傳統計算力矩控制器須回授速度資訊，而速度資訊通常使用差分的方式來求得，但差分運算後容易把雜訊放大產生大幅度的震盪，使控制器效能降低，故改採用[20]新設計的改良型 LQR 控制器，如圖 3.3 所示。

此控制器特色除了結合 LQR 控制器強健性的優點外，更只需回授編碼器所量得的角度值，不需要額外做差分的運算。

首先在 LQR 控制器受控廠前加入一個濾波器，使其成為一個增大的受控廠，此濾波器功用是為了將 LQR 控制器狀態回授轉換成對輸出回授的控制器，避免以往設計控制器的方式，在加入觀察器後，改變其原本狀態回授控制系統。

由(3.1.6)式中可以看出， $\bar{\mathbf{M}}(\mathbf{q})$  矩陣內含有線性常數矩陣部分及非線性項部分，分別令為  $\bar{\mathbf{M}}_0$  及  $\bar{\mathbf{M}}_1(\mathbf{q})$ ，並將  $v$  拆成  $v_{Lin}$ 、 $v_{ff}$ ，可整理出：

$$\bar{\mathbf{M}}_0 \ddot{\mathbf{q}} + \bar{\mathbf{B}} \dot{\mathbf{q}} = v_{Lin} + v_{ff} - (\bar{\mathbf{M}}_1(\mathbf{q}) \ddot{\mathbf{q}} + \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \bar{\mathbf{G}}(\mathbf{q}) + \bar{\mathbf{F}}(\dot{\mathbf{q}})) \quad (3.2.1)$$

令  $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \bar{\mathbf{M}}_1 \ddot{\mathbf{q}} + \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \bar{\mathbf{G}}(\mathbf{q}) + \bar{\mathbf{F}}(\dot{\mathbf{q}})$ ，並將上式表示成動態方程式：

$$\frac{d}{dt} \underbrace{\begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \dot{\mathbf{q}}_1 \\ \dot{\mathbf{q}}_2 \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\bar{\mathbf{M}}_0^{-1} \bar{\mathbf{B}} \\ 0 & 0 & & \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \dot{\mathbf{q}}_1 \\ \dot{\mathbf{q}}_2 \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \bar{\mathbf{M}}_0^{-1} \end{bmatrix}}_{\mathbf{B}} (\mathbf{v}_{\text{Lin}} + \mathbf{v}_{\text{ff}} - \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})) \quad (3.2.2)$$

其中  $\mathbf{v}_{\text{ff}} = \bar{\mathbf{M}}_1(\mathbf{q}) \ddot{\mathbf{q}}_d + \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}_d) \dot{\mathbf{q}}_d + \bar{\mathbf{G}}(\mathbf{q}) + \bar{\mathbf{F}}(\dot{\mathbf{q}}_d)$ ，主要的作用在於消除非線性動態

$\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$  的前饋控制輸出。將(3.2.2)式，重新整理後便可得到

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{v}_{\text{Lin}} \quad (3.2.3)$$

$$\mathbf{y} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \dot{\mathbf{q}}_1 \\ \dot{\mathbf{q}}_2 \end{bmatrix} = \mathbf{C}\mathbf{x} \quad (3.2.4)$$

根據(3.2.3)式線性項部分所設計出的線性控制器輸出

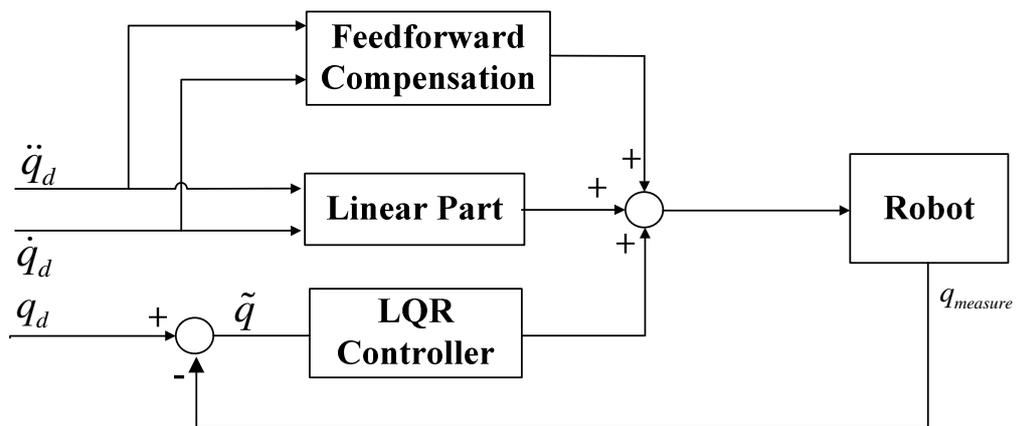


圖 3.3 不需要速度估測的控制器方塊圖

此兩軸機械臂系統控制器的目的為角度追蹤並追蹤其角速度，因此設角度及角速度的命令軌跡為  $\mathbf{x}_d = [q_{1d} \quad q_{2d} \quad \dot{q}_{1d} \quad \dot{q}_{2d}]^T$ ，並令  $v_d$  為：

$$\mathbf{v}_d = [\bar{\mathbf{B}} \quad \bar{\mathbf{M}}_0] \begin{bmatrix} \dot{q}_{1d} \\ \dot{q}_{2d} \\ \ddot{q}_{1d} \\ \ddot{q}_{2d} \end{bmatrix} \quad (3.2.5)$$

而目前的系統狀態為  $\mathbf{x} = [q_1 \quad q_2 \quad \dot{q}_1 \quad \dot{q}_2]^T$ ，所以可以得到角度及角速度的追蹤誤差  $\tilde{\mathbf{x}} = [\tilde{q}_1 \quad \tilde{q}_2 \quad \dot{\tilde{q}}_1 \quad \dot{\tilde{q}}_2]^T$ ，其中  $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d$ ， $\tilde{v}_{Lin} = v_{Lin} - v_d$ 。最後改寫(3.2.3)得到新的狀態空間表示如(3.2.6)式所示。

$$\dot{\tilde{\mathbf{x}}} = \mathbf{A}\tilde{\mathbf{x}} + \mathbf{B}\tilde{v}_{Lin} \quad (3.2.6)$$

$$\tilde{\mathbf{y}} = \mathbf{C}\tilde{\mathbf{x}} \quad (3.2.7)$$

接著針對(3.2.6)與(3.2.7)設計控制器，假設  $G(s)$  是上述系統的轉移矩陣。加入一個濾波器  $\Lambda(s)$  到電壓輸入端以增大受控廠  $G(s)$ 。如圖 3.4 所示。

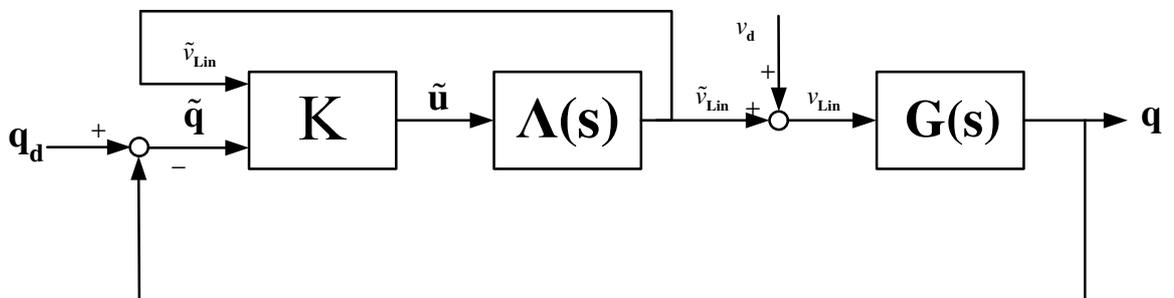


圖 3.4 增大的受控廠

其中

$$\Lambda(s) = \begin{bmatrix} \frac{1}{s+\lambda_0} & 0 \\ 0 & \frac{1}{s+\lambda_0} \end{bmatrix} \quad (3.2.8)$$

係數  $\lambda_0$  為設計的參數。加入  $\Lambda$  後可得到增大的受控廠的狀態空間為

$$\frac{d}{dt} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \\ \tilde{\boldsymbol{\tau}}_{\text{Lin1}} \\ \tilde{\boldsymbol{\tau}}_{\text{Lin2}} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\bar{\mathbf{M}}_0^{-1}\bar{\mathbf{B}} & -\bar{\mathbf{M}}_0^{-1} & -\bar{\mathbf{M}}_0^{-1} \\ \mathbf{0} & \mathbf{0} & -\lambda_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\lambda_0 \end{bmatrix}}_{\mathbf{A}_a} \underbrace{\begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \\ \tilde{\boldsymbol{\tau}}_{\text{Lin1}} \\ \tilde{\boldsymbol{\tau}}_{\text{Lin2}} \end{bmatrix}}_{\tilde{\mathbf{x}}} + \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}}_{\mathbf{B}_{a1}} \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \tilde{\mathbf{u}}_1 \\ \tilde{\mathbf{u}}_2 \end{bmatrix}}_{\tilde{\mathbf{u}}} \quad (3.2.9)$$

令狀態回授控制器的增益為  $\tilde{\mathbf{u}} = -\mathbf{K}\tilde{\mathbf{x}}$ ，則：

$$\tilde{\mathbf{u}} = -\mathbf{K}\tilde{\mathbf{x}} = -\begin{bmatrix} \mathbf{K}_1 & \mathbf{K}_2 & \mathbf{K}_3 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \\ \tilde{\boldsymbol{\tau}}_{\text{Lin}} \end{bmatrix} = -\mathbf{K}_1\tilde{\mathbf{q}} - \mathbf{K}_2\dot{\tilde{\mathbf{q}}} - \mathbf{K}_3\tilde{\boldsymbol{\tau}}_{\text{Lin}} \quad (3.2.10)$$

接著將狀態回授轉換成輸出回授的形式。而  $\hat{\tilde{\mathbf{u}}}$  及  $\hat{\tilde{\boldsymbol{\tau}}}_{\text{Lin}}$  分別表示  $\tilde{\mathbf{u}}$  和  $\tilde{\boldsymbol{\tau}}_{\text{Lin}}$  的拉氏轉換。

(3.2.10) 的拉氏轉換可以表示為

$$\hat{\tilde{\mathbf{v}}}_{\text{Lin}} = -\Lambda(s) \left( \mathbf{K}_1 \begin{bmatrix} \hat{\tilde{\mathbf{q}}}_1 & \hat{\tilde{\mathbf{q}}}_2 \end{bmatrix}^T + s\mathbf{K}_2 \begin{bmatrix} \hat{\tilde{\mathbf{q}}}_1 & \hat{\tilde{\mathbf{q}}}_2 \end{bmatrix}^T + \mathbf{K}_3 \begin{bmatrix} \hat{\tilde{\mathbf{v}}}_{\text{Lin1}} & \hat{\tilde{\mathbf{v}}}_{\text{Lin2}} \end{bmatrix}^T \right) \quad (3.2.11)$$

將上式移項整理後，可以得到輸出狀態回授控制方程式：

$$\hat{\tilde{\mathbf{v}}}_{\text{Lin}} = -[\Lambda^{-1}(s) + \mathbf{K}_3]^{-1} \left( (\mathbf{K}_1 + s\mathbf{K}_2) \begin{bmatrix} \hat{\tilde{\mathbf{q}}}_1 & \hat{\tilde{\mathbf{q}}}_2 \end{bmatrix}^T \right) \quad (3.2.12)$$

最後將(3.2.12)式展開可得

$$\begin{bmatrix} \hat{\mathbf{v}}_{Lin1} \\ \hat{\mathbf{v}}_{Lin2} \end{bmatrix} = \frac{1}{s^2 + (\lambda_0 + \lambda_0 + \mathbf{k}_{31} + \mathbf{k}_{34})s + (\lambda_0 + \mathbf{k}_{31})(\lambda_0 + \mathbf{k}_4) - \mathbf{k}_{32}\mathbf{k}_{33}} \begin{bmatrix} \mathbf{a}_{11} & \mathbf{a}_{12} \\ \mathbf{a}_{21} & \mathbf{a}_{22} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{q}}_1 \\ \hat{\mathbf{q}}_2 \end{bmatrix} \quad (3.2.13)$$

其中

$$\begin{aligned} \mathbf{a}_{11} &= \mathbf{k}_{21}s^2 + [\mathbf{k}_{11} + \mathbf{k}_{21}(\lambda_0 + \mathbf{k}_{34}) - \mathbf{k}_{32}\mathbf{k}_{23}]s + \mathbf{k}_{11}(\lambda_0 + \mathbf{k}_{34}) - \mathbf{k}_{32}\mathbf{k}_{13} \\ \mathbf{a}_{12} &= \mathbf{k}_{22}s^2 + [\mathbf{k}_{12} + \mathbf{k}_{14}(\lambda_0 + \mathbf{k}_{34}) - \mathbf{k}_{32}\mathbf{k}_{24}]s + \mathbf{k}_{12}(\lambda_0 + \mathbf{k}_{34}) - \mathbf{k}_{32}\mathbf{k}_{14} \\ \mathbf{a}_{21} &= \mathbf{k}_{23}s^2 + [\mathbf{k}_{13} + \mathbf{k}_{23}(\lambda_0 + \mathbf{k}_{31}) - \mathbf{k}_{33}\mathbf{k}_{21}]s + \mathbf{k}_{13}(\lambda_0 + \mathbf{k}_{31}) - \mathbf{k}_{33}\mathbf{k}_{11} \\ \mathbf{a}_{22} &= \mathbf{k}_{24}s^2 + [\mathbf{k}_{14} + \mathbf{k}_{24}(\lambda_0 + \mathbf{k}_{31}) - \mathbf{k}_{33}\mathbf{k}_{22}]s + \mathbf{k}_{14}(\lambda_0 + \mathbf{k}_{31}) - \mathbf{k}_{33}\mathbf{k}_{12} \end{aligned} \quad (3.2.14)$$

從上式可以觀察出，此控制器只需量得角度資訊，並配合命令的角速度和角加速度代入機械手臂的動力學方程式中，不需使用觀察器或速度估測器。

### 3.3 手臂路徑規畫

#### 3.3.1 手臂 D-H model

透過上節所設計之控制器，兩軸機械手臂實驗平台已建構完畢。只需下達角度、角速度與角加速度即可任意的控制機械手臂到達指定的角度。

接下來為了使機械手臂到達真實空間中指定的位置，採用機器人學上由 J. Denavit 和 R. S. Hartenberg 所提出來的 D-H model，將機械人參數由 6 個縮減成 4 個，以方便運算。

其中，機械手臂座標採用右手座標系統，原本 6 個參數分別為對  $x_a, y_a, z_a$  三軸旋轉的角度及對  $x_a, y_a, z_a$  三軸平移的位置，將其化簡為  $d, \theta, a, \alpha$ ，分別代表兩軸垂直線的距離，兩軸的角度差值也就是前幾節動力學模型中所提到的角度值  $q$ ，手臂軸的長度及兩軸旋轉軸的歪斜角度差值。

依照圖 3.1 可觀察出，此實驗平台有兩個活動軸，在 D-H model 中習慣將旋轉軸用 z 表示，此部分為機器人 DH model 座標系統，與圖 3.1 手臂座標系座標表示方法不同。由於兩軸在同一直線上也無歪斜角度，故 d 與  $\alpha$  皆為 0 度，將圖 3.1 轉換後，所得結果如圖 3.5 所示：

Joint	$\theta$	d	a	$\alpha$	variable
1	$\theta_1$	0	$l_1$	$0^\circ$	$\theta_1$
2	$\theta_2$	0	$l_2$	$0^\circ$	$\theta_2$

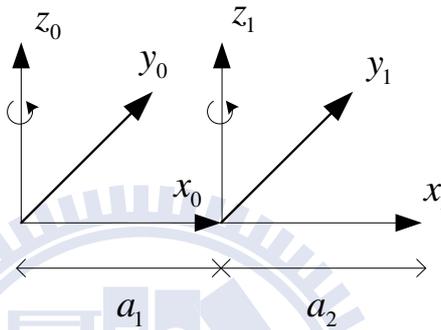


圖 3.5 機械手臂 D-H model

### 3.3.2 運動學

上小節得到 D-H model 後，透過旋轉矩陣及平移矩陣進行座標轉換，就可以由前一軸座標系統得到下一軸座標系統，藉由此運算即可得到機械手臂頂端座標位置。

以下為對  $x_a, y_a, z_a$  軸旋轉的旋轉矩陣(3.3.1)

$$\mathbf{R}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{R}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{R}_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3.1}$$

對  $x_a, y_a, z_a$  軸平移的平移矩陣(3.3.2)

$$\mathbf{Trans}(Q) = \begin{bmatrix} 1 & 0 & 0 & q_x \\ 0 & 1 & 0 & q_y \\ 0 & 0 & 1 & q_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3.2)$$

由 D-H model 可推得第一軸機械手臂座標轉換為

$$\mathbf{T}_1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & l_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & l_1 \cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & l_1 \sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3.3)$$

第二軸機械手臂座標轉換為

$$\mathbf{T}_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & l_2 \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & l_2 \sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3.4)$$

將兩軸座標轉換矩陣相乘後，可推得手臂頂端座標位置

$$\mathbf{T} = \mathbf{T}_1 \times \mathbf{T}_2 = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & l_2 \cos(\theta_1 + \theta_2) + l_1 \cos\theta_1 \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) & 0 & l_2 \sin(\theta_1 + \theta_2) + l_1 \sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3.5)$$

因機械手臂實驗平台參數  $l_1 = l_2$ ，故可以將(3.3.5)化簡為

$$\mathbf{T} = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & l_1(\cos(\theta_1 + \theta_2) + \cos\theta_1) \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) & 0 & l_1(\sin(\theta_1 + \theta_2) + \sin\theta_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3.6)$$

由(3.3.6)可求得機械手臂頂端於手臂座標系中座標位置為

$$\begin{aligned} P_{za} &= l_1(\cos(\theta_1 + \theta_2) + \cos\theta_1) \\ P_{ya} &= l_1(\sin(\theta_1 + \theta_2) + \sin\theta_1) \end{aligned} \quad (3.3.7)$$

### 3.3.3 反運動學

由於在控制器部分是需要下達機械手臂角度的命令值，故必須藉由反運動學將真實空間中任一位置座標反推回機械手臂各軸角度值。

先將(3.3.7)平方相加後可得

$$\begin{aligned} P_{za}^2 + P_{ya}^2 &= l_1^2(\cos^2(\theta_1 + \theta_2) + \cos^2\theta_1 + 2\cos(\theta_1 + \theta_2)\cos\theta_1) + \\ & l_1^2(\sin^2(\theta_1 + \theta_2) + \sin^2\theta_1 + 2\sin(\theta_1 + \theta_2)\sin\theta_1) \\ &= l_1^2(2 + 2\cos(\theta_1 + \theta_2 - \theta_1)) \\ &= 2l_1^2(1 + \cos\theta_2) \end{aligned} \quad (3.3.8)$$

經移項整理後可推得

$$\theta_2 = \cos^{-1}\left(\frac{P_{za}^2 + P_{ya}^2 - 2l_1^2}{2l_1^2}\right) \quad (3.3.9)$$

接著，使用幾何圖解法求解 $\theta_1$ 如圖 3.6 所示

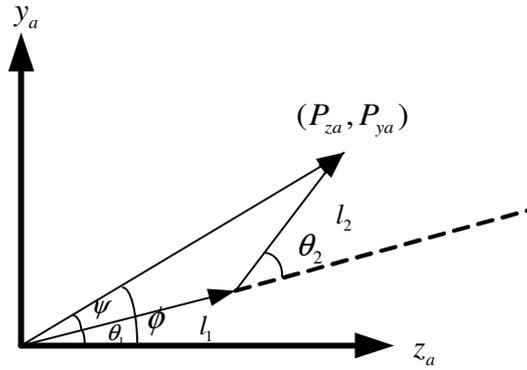


圖 3.6 手臂幾何示意圖

使用餘弦定理

$$l_2^2 = l_1^2 + (\sqrt{P_{za}^2 + P_{ya}^2})^2 - 2l_1(\sqrt{P_{za}^2 + P_{ya}^2})\cos\psi \quad (3.3.10)$$

經移項整理後

$$\psi = \cos^{-1}\left(\frac{l_1^2 - l_2^2 + P_{za}^2 + P_{ya}^2}{2l_1(\sqrt{P_{za}^2 + P_{ya}^2})}\right) = \cos^{-1}\left(\frac{\sqrt{P_{za}^2 + P_{ya}^2}}{2l_1}\right) \quad (3.3.11)$$

最後可求得

$$\theta_1 = \phi - \psi = \tan^{-1}\left(\frac{P_{ya}}{P_{za}}\right) - \cos^{-1}\left(\frac{\sqrt{P_{za}^2 + P_{ya}^2}}{2l_1}\right) \quad (3.3.12)$$

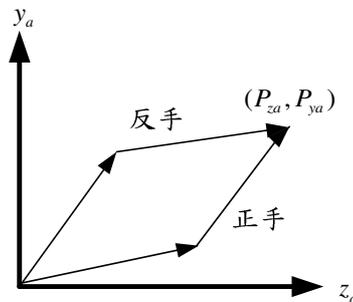


圖 3.7 手臂正反手示意圖

考慮正反手問題如圖 3.7 所示，可將(3.3.9)式與(3.3.12)式改寫為

$$\begin{aligned}\theta_1 &= \tan^{-1}\left(\frac{P_{ya}}{P_{za}}\right) \pm \cos^{-1}\left(\frac{\sqrt{P_{za}^2 + P_{ya}^2}}{2l_1}\right) \\ \theta_2 &= \pm \cos^{-1}\left(\frac{P_{za}^2 + P_{ya}^2 - 2l_1^2}{2l_1^2}\right)\end{aligned}\quad (3.3.13)$$

如(3.3.13)所示，則可得到 4 組角度解，再加上機械手臂角度正反轉結果後，理論上應該得到 8 組角度解，但由於機械手臂實驗平台在第一軸基座連接處有橫桿的關係，使第二軸無法跨越過去，使得實現上必須考慮避障問題。故機械手臂第一軸與第二軸夾角不得小於 10 度，換句話說，第二軸手臂角度必須小於  $\pm 170$  度，故實際上仍然只可得到 4 組角度解。

當求取反運動學時，有可能輸入真實座標位置無法到達，故必須先判斷是否在機械手臂工作區範圍內。由(3.3.13)式可觀察出， $\cos$  函數範圍僅在 -1~1 之間，如果超出範圍時，則反三角會出現虛數，故可以以此為判斷依據來辨別輸入真實座標位置是否在工作區內。

### 3.3.4 路徑規劃

將反運動學所得之角度，以三次多項式來描述其時間對角度的關係式。由於三次多項式有五個未知係數，故代入五個已知係數，分別為初始速度、終止速度、初始位置、終止位置及終止時間即可求得三次多項式各參數。

首先得到一個三次多項式

$$\theta = c_3 t^3 + c_2 t^2 + c_1 t + c_0 \quad (3.3.14)$$

代入已知常數  $\theta(0) = \theta_0, \theta(t_f) = \theta_f, v(0) = v_0, v(t_f) = v_f$ ，並寫成矩陣形式得到

$$\begin{bmatrix} 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} \theta_f \\ v_f \\ \theta_0 \\ v_0 \end{bmatrix} \quad (3.3.15)$$

再將求得之三次多項式輸入到控制器，使機械手臂依照此時間對角度關係運動。且在運動過程中，可以即時加入新位置，並可立即規畫出新路徑，使新舊路徑平順地銜接上。最後，在多點的路徑規畫中，為了使機械手臂可移動較少角度之路徑，故在選定下一點大小軸角度時，先將前一點所得大小軸角度與下一點所得大小軸四組角度，分別做 2-norm 運算，並且將運算結果最小值之大小軸角度選為下一點角度值，並依此角度產生關係式進行路徑規畫，如圖 3.8 所示。

比較任意選取角度值，如圖 3.8 左圖，及選擇最短距離，如圖 3.8 右圖，可清楚觀察出兩條路徑所移動的距離差距非常顯著。

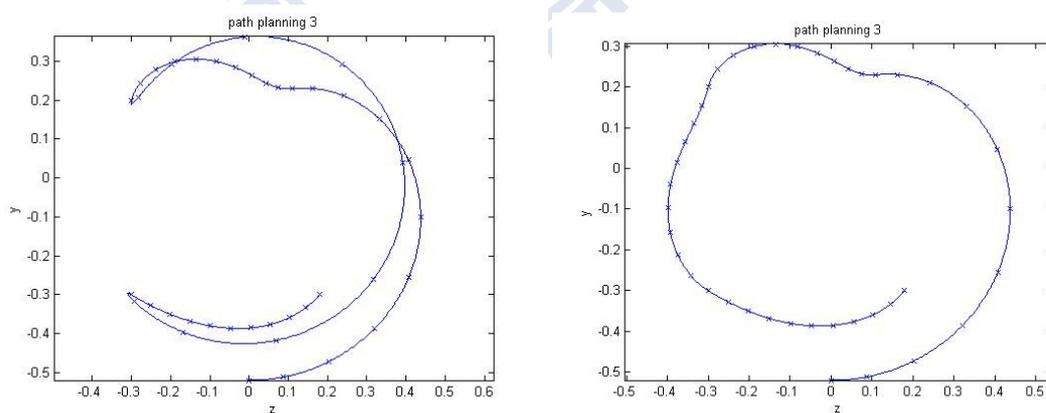


圖 3.8 路徑規畫移動比較圖

整體流程如圖 3.9，在真實空間中選取欲到達之座標位置，套入反運動學公式(3.3.13)求得 8 組角度解，將這 8 組角度經過避障考量後剩餘 4 組角度，篩選出可旋轉最少之角度，選擇出 1 組角度，再將此組角度利用三次式描述出時間對角度關係，透過控制器控

制即可使機械手臂依照規劃方式動作，運動途中如有插入新座標位置，也可立即規劃出新的路線取代原有路線。

路徑規劃流程圖如下所示：



圖 3.9 路徑規劃流程圖

## 第四章 影像擷取與追蹤

為了達到擊球之目的，必須先找到目標物，需要利用攝影機鏡頭來模擬人類的眼睛，確認目標物位置。在實現上，透過機器視覺與影像處理的部分，來追蹤飛行球體並在每張影像中標定其座標。

故在此章，介紹所使用的影像處理方法，為了使後續影像處理部分可以較為簡單且快速，故一開始先將彩色影像轉為灰階影像，再設法轉換成黑白二元影像，其中技術包含使用背景相減及顏色辨識的方式，結合兩種方法，可清楚地突顯出特定追蹤之物體，使影像追蹤上達到更好的效果，再加上動態閾值的演算法，使閾值不會只固定某個數值，可視情況而改變。於影像處理過程中再加入簡單的濾波器及形態學。

另一方面為了提高速度且不降低解析度，當影像追蹤成功的情況下，採用局部小畫面的處理方式。整體而言，希望能達到高速地影像處理並可以持續追蹤目標物，故演算法的計算效率也是需要考量的因素之一。

### 4.1 顏色辨識

本論文採用三種方式來將彩色影像轉為二元影像，再比較其效果並考量運算速度，選取最適宜的處理方式。

#### 4.1.1 RGB 法

此法相當容易，把輸入影像的紅色 R、綠色 G 及藍色 B 三種主要頻譜成分的二維灰階影像，分別選定特定閾值，將其二值化即可。此法優點在於不需要經過特殊運算，只要特定閾值即可，速度較快，但實作上，閾值選取容易受環境光照影響，較難調整出最佳閾值，效果較差。

以下圖 4.1 為輸入之彩色影像：



圖 4.1 彩色影像



圖 4.2 經過閾值後影像



圖 4.3 AND 後二元影像

在靜態彩色影像中，如圖 4.1 所示，分別選取閾值後結果，由左至右分別為  $R > 95$ ， $G < 110$ ， $B < 115$ ，可得到如圖 4.2 所示之影像，再將此三張影像執行 AND 動作，即可得到一張二維的二元影像，如圖 4.3 所示。

## 4.1.2 向量法

向量法是先取一組想要分割之「平均」色彩的估測值。令此平均色彩以 RGB 向量  $\mathbf{a}$  表示之。分割的方式是將彩色影像中每一個像素的 RGB 分量值取出，依照色彩是否在指定的範圍內分類。使用量測**歐基里德距離**來執行此判斷，如式(4.1.1)表示：

$$\begin{aligned} D(\mathbf{z}, \mathbf{a}) &= \|\mathbf{z} - \mathbf{a}\| = \left[ (\mathbf{z} - \mathbf{a})^T (\mathbf{z} - \mathbf{a}) \right]^{\frac{1}{2}} \\ &= \left[ (z_R - a_R)^2 + (z_G - a_G)^2 + (z_B - a_B)^2 \right]^{\frac{1}{2}} \end{aligned} \quad (4.1.1)$$

當  $D(\mathbf{z}, \mathbf{a}) \leq D_0$ ， $\mathbf{z}$  即為欲分割之 RGB 向量。

其中下標 R, G, B 代表向量  $\mathbf{a}$  和  $\mathbf{z}$  的 RGB 分量。 $D_0$  為指定區域，是一個半徑為  $D_0$  的實心球體。如圖 4.4 表示之

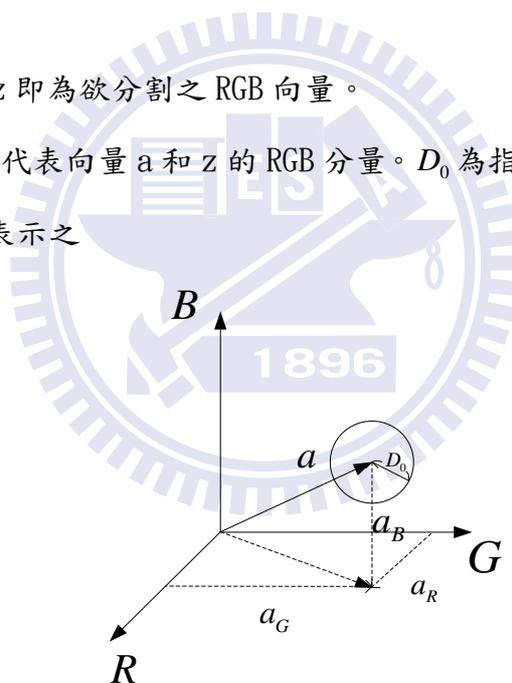


圖 4.4 向量法示意圖

使用向量法必須先找出欲分割的平均色彩值及範圍區域，較為麻煩。也由於需要計算歐基理德距離，故運算時間稍微較長，但效果明顯比 RGB 法好。

將圖 4.1 彩色影像透過向量法處理後，如圖 4.5 所示，其中設定參數  $\mathbf{a} = [255 \ 0 \ 0]$ ， $D_0 = 195$ 。



圖 4.5 向量法處理後二元影像

### 4.1.3 HSV 法

HSV 是根據色相、飽和度和明度來確定顏色的一種方法。色相是色彩的基本屬性，就是平常所說的顏色名稱，如紅色、黃色等，取 0-360 度的數值。因為色相不容易受到光的強弱變化影響，常常作為偵測人臉的依據。飽和度是指色彩的純度，越高色彩越純，低則逐漸變灰，取 0-100% 的數值。明度也叫亮度，取 0-100%。

其示意圖如圖 4.6 所示

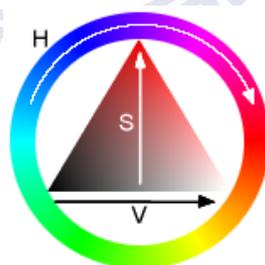


圖 4.6 HSV 示意圖

HSV 法與 RGB 法相似，先將輸入影像透過(4.1.2)式，轉換為 HSV 表示，再個別選取其閾值來達到二值化處理。

$$H = \begin{cases} 6 + \frac{G-B}{MAX-MIN} \times 60^\circ, & \text{if } R=MAX \\ 2 + \frac{B-R}{MAX-MIN} \times 60^\circ, & \text{if } G=MAX \\ 6 + \frac{R-G}{MAX-MIN} \times 60^\circ, & \text{if } B=MAX \end{cases} \quad (4.1.2)$$

$$S = \frac{MAX-MIN}{MAX}$$

$$V = MAX$$

將圖 4.1 彩色影像，經過(4.1.2)轉換後，如圖 4.7 所示，由左至右分別為 H，S，V 的二維灰階影像，將圖 4.7 前兩張 H 與 S 灰階影像，選取閾值，分別為  $H > 127, H < 20$ ， $S > 90$ ，執行 AND 動作，即可得到一張二維的二元影像，如圖 4.8 所示。



圖 4.7 HSV 灰階影像

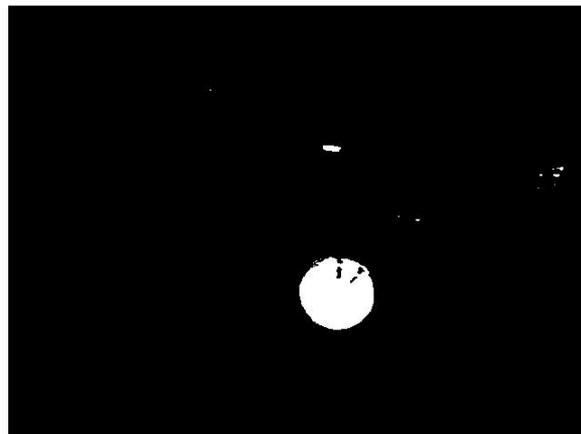


圖 4.8 H 與 S AND 後二元影像

由圖 4.8 可觀察出，其效果較優於 RGB 法，且與明度無關，故較不受光線亮度影響，只需利用色相與飽和度即可選取欲辨識之顏色，參數調整容易，且計算時間較向量法短，整體效果乍看之下與向量法相近，但使用同組閾值參數改變背景場景後，三者再進行比較後，可以明顯看出 HSV 法表現較好，故之後使用顏色來分割欲偵測物體，皆使用 HSV 法來辨識。

當圖 4.9 左圖較亮的原始影像分別使用圖 4.3、圖 4.5 及圖 4.8 之閾值，可得到如圖 4.10 之結果，明顯為 HSV 法表現較好。將圖 4.9 右圖較暗的原始影像代入相同閾值後，如圖 4.11 所示，比較後可發現，HSV 法即使在光線不足的情況下，仍然可以明顯表現出欲顯現之顏色，不受光線亮度所影響。



圖 4.9 亮度不同時原始影像

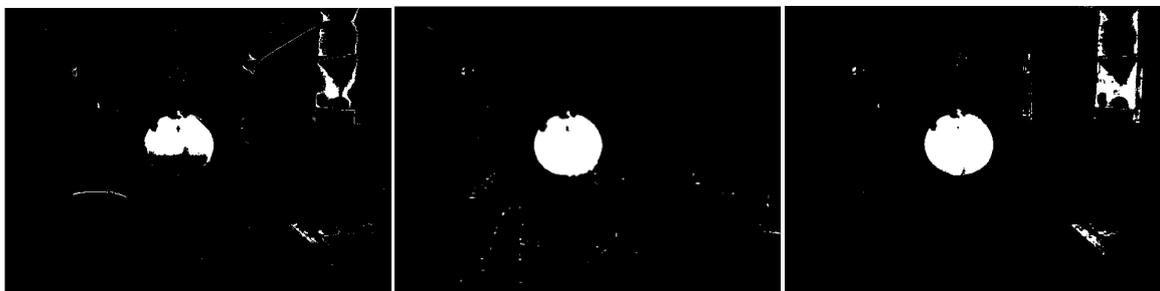


圖 4.10 改變場景後，RGB 法(左)與 HSV 法(中)與向量法(右)



圖 4.11 改變場景及亮度後，RGB 法(左)與 HSV 法(中)與向量法(右)

## 4.2 背景相減

由於實驗背景複雜，假使背景有與欲分割出的顏色相似，則無法辨識欲選取之目標物，為了避免此情形發生，故加入背景相減法與顏色辨識結合，使欲選取之目標物能更單純地被擷取出來。

### 4.2.1 靜態背景模型

採用背景相減法一開始先使用最簡單的方式，拍一張最初的影像，再將其轉為灰階影像，之後並以此影像做為固定的靜態背景，如式(4.2.1)。接著只需將新輸入影像與此影像相減，設定閾值，只要有相異過大之像素，即會被顯示出來。

$$\mathbf{B}_{x,y}^k = \mathbf{I}_{x,y}^0 \quad k=0,1,2,\dots \quad (4.2.1)$$

其中  $\mathbf{B}_{x,y}^k$  表示位於  $x, y$  像素的第  $k$  張背景灰階值， $\mathbf{I}$  則代表輸入之影像。

此法簡易，運算速度快，但只需要與最初的靜態背景相異像素皆會被顯現出來，假設新影像物體進入後，隔一段時間固定不動，理論上已成為新背景，此背景模型仍會將此物體顯示出來，或在實驗過程中，攝影機鏡頭受到外力干擾產生些微震動，都將使相減過程中，大受影響。

如圖 4.12 所示為靜態背景的灰階影像，轉二元影像後產生的雜訊，通常為燈光變化所影響。

以圖 4.12 為靜態背景，加入新物體得到背景相減結果，如圖 4.13 所示。使攝影機鏡頭受到些微振動後，雖然在原始影像無法觀察出任何變化，但背景相減結果，如圖 4.14 所示，產生相當大的差異。



圖 4.12 灰階背景(左)與二元背景(右)

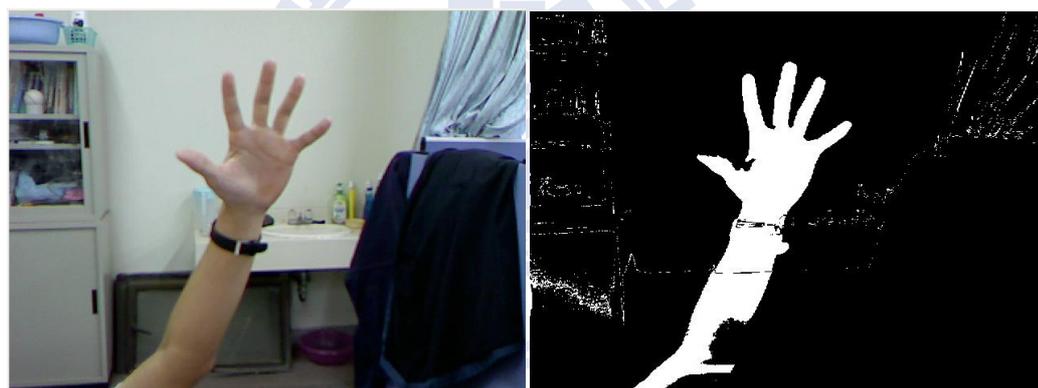


圖 4.13 原始影像(左)與背景相減後結果(右)

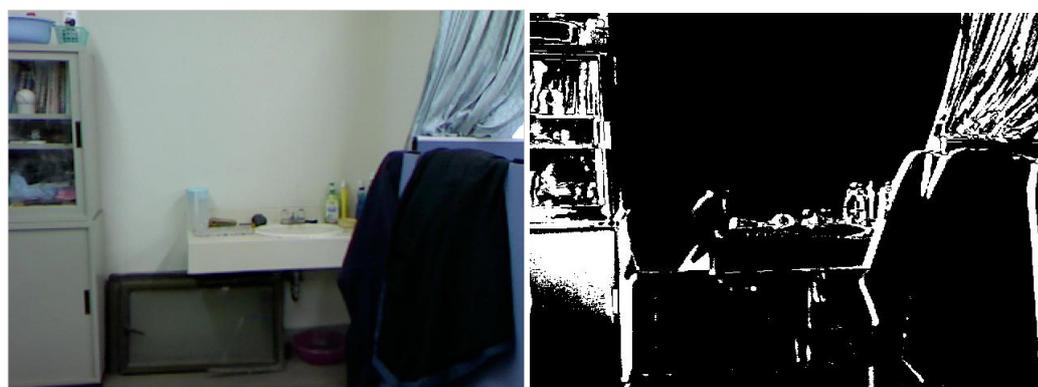


圖 4.14 震動後原始影像(左)與背景相減結果(右)

## 4.2.2 閾值背景模型

改良靜態背景法，使背景可以更新，透過閾值設定，將新影像與背景做絕對誤差運算後，小於閾值的像素更新為新的背景，反之，則使用原來的背景。其背景模型如式(4.2.2)表示

$$\mathbf{B}_{x,y}^k = \begin{cases} \mathbf{I}_{x,y}^k, & \mathbf{AD}_{x,y}^k < \mathbf{Th} \\ \mathbf{B}_{x,y}^{k-1}, & \mathbf{AD}_{x,y}^k > \mathbf{Th} \end{cases} \quad \mathbf{k}=0,1,\dots \quad (4.2.2)$$

其中  $\mathbf{AD}_{x,y}^k = |I_{x,y}^k - B_{x,y}^{k-1}|$ ，Th 為閾值，可依照需求自行調整大小

此法雖可即時更新影像背景，但由於在更新背景像素值時，於新加入物體邊緣處容易因為光照或其他雜訊等影響，使本來背景相減差值應該遠大於閾值，結果卻小於閾值，此時新的背景像素值被更新，之後，將新加入物體移走並得到新輸入影像，此時新輸入影像像素值等於最原始背景像素值，但此時背景像素值卻已被更新為另一像素值，於是背景相減後大於閾值並被顯現出來，造成類似殘影的情形發生。除非再加入新物體改變此背景像素值，否則將持續顯示下去。

原始背景如圖 4.12(左)所示，加入新物體(手)影像如圖 4.13(左)所示，移動新物體，得到背景相減殘影結果如圖 4.15 所示。



圖 4.15 造成殘影結果

### 4.2.3 IRGABM 模型

為了改良閾值背景模型，採用新型背景模型[16]所提出的 IRGABM(Improved Running Gaussian Average Background Model)法，其中提到產生干擾影像的雜點分為兩種，變化快但改變較小的雜訊效應，變化慢但改變較大的變化光照效應。此法利用兩個參數改變更新新背景像素值的比重，可改善閾值背景模型產生的殘影問題，也可去除雜訊效應，但變化較大的光照效應較難去除。假使有加入新物體經過一段時間後固定不動，將漸漸被更新成為背景，模型數學式如式(4.2.3)表示

$$B_{x,y}^k = \begin{cases} (1-\alpha_1)B_{x,y}^{k-1} + \alpha_1 I_{x,y}^k, & k > 0, AD_{x,y}^k < Th \\ (1-\alpha_2)B_{x,y}^{k-1} + \alpha_2 I_{x,y}^k, & k > 0, AD_{x,y}^k > Th \\ I_{x,y}^0, & k = 0 \end{cases} \quad (4.2.3)$$

其中  $\alpha_1, \alpha_2$  可以依照需求自行調整，數字越小背景更新越慢，通常  $\alpha_2 < \alpha_1$

原始背景如圖4.12(左)所示，背景相減結果如圖4.16(左)所示，在此  $\alpha_1 = 0.1, \alpha_2 = 0.05$



圖 4.16 IRGABM 背景相減(左)與持續 3 秒後(中)與延遲 10 秒後(右)

由圖4.16可以很明顯觀察出，此法除了在處理雜訊方面比靜態背景好以外，也可改善閾值背景模型所帶來的殘影效果。當新加入物體(手)持續一小段時間不動後，備更新為新背景漸漸消失，最後完全成為新背景，如此一來，即使攝影機鏡頭受到些微振動，隔一小段時間後，背景相減後也可以不受到影響。

### 4.3 三角演算法

G. W. ZACK 於 1977 年提出[25]，選取灰階值中的最高點與最低點並連成一線，再求得與這條線最遠距離的灰階值，即是閾值，如圖 4.17 所示。可以利用此演算法來自動選取閾值，而不是單單只選定一個固定的數值。其他還有許多種演算法如[6]所提到的 Background-symmetry algorithm、Isodata algorithm...等方式，但由於本實驗出現的直方圖與三角演算法要求的圖形相近，故採用三角演算法來設定動態閾值。

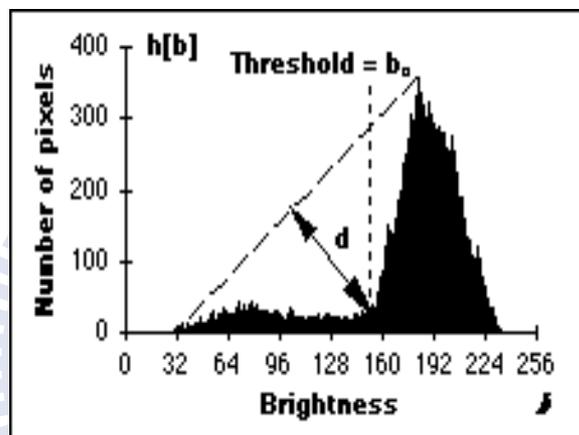


圖 4.17 三角演算法示意圖

### 4.4 中值濾波器

由於使用背景相減與顏色辨識後，仍然會有些許雜訊，故加入濾波器使處理畫面可以更簡潔，之後在追蹤目標物時，較不會被雜訊所干擾，在此採用中值濾波器來達到此功能。

中值濾波器是一種非線性的空間濾波器，對於處理胡椒鹽雜訊，提供了絕佳的雜訊降低效能，比起同樣大小的線性平滑濾波有顯著較小的模糊化。

其響應建立在由濾波器所包圍之影像區域中所含像素的順序上，然後由排序結果決定的值來取代中心像素的值。將像素的值用該像素(像素的原始值包含在中值的計算內)近鄰灰階的中間值來取代。為了在一幅影像中的一個點上執行中值濾波，首先把問題裡

的像素和其鄰近像素的值排序，並決定它們的中間值，然後把這個值指定到該像素。例如：在一個 $3 \times 3$ 的鄰域中，中間值是第五個最大的值，在 $5 \times 5$ 的鄰域中，中間值是第十三個最大值，以此類推。

## 4.5 形態學

影像處理上，有一些基本的處理方式可以使影像膨脹或收縮，在應用上也可利用這些處理來達到將影像斷開或閉合，或是可以用來處理雜訊。

由於原始畫面在經過先前所介紹到的許多方式處理過後，觀察畫面可察覺到，有些連接處出現不連續的情形發生，為了盡量避免此情形發生，故希望藉由形態學膨脹的方式，將影像中有些許斷開的部分連接起來。

### 4.5.1 膨脹

當 $A$ 和 $B$ 為 $Z^2$ 中的兩個集合， $A$ 藉由 $B$ 的膨脹表示為 $A \oplus B$ ，數學表示法為

$$A \oplus B = \left\{ z \mid \left( \hat{B} \right)_z \cap A \neq \emptyset \right\} \quad (4.4.1)$$

其中 $B$ 稱為結構元素， $\left( \hat{B} \right)_z$ 表示為對其原點反射並平移此反射  $z$  單位

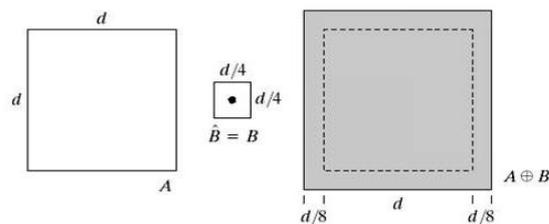


圖 4.18 集合  $A$ (左)與結構元素  $B$ (中)與膨脹後結果(右)

## 4.5.2 侵蝕

當A和B為 $Z^2$ 中的兩個集合，A被B的侵蝕表示為 $A \ominus B$ ，數學表示法為

$$A \ominus B = \{z \mid (B)_z \subseteq A\} \quad (4.4.2)$$

其中B稱為結構元素， $(B)_z$ 表示為對其原點平移 $z$ 單位

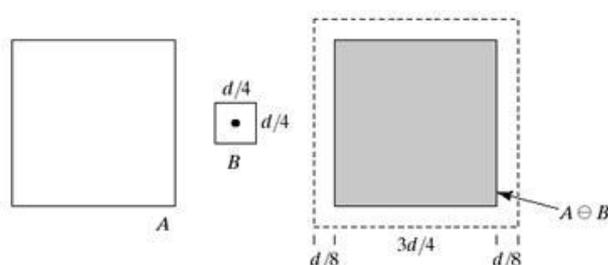


圖 4.19 集合 A(左)與結構元素 B(中)與侵蝕後結果(右)

## 4.6 影像擷取流程

經過前幾節個別介紹所使用到的影像處理方式後，在此說明如何把目標物顯現出來的完整流程。

首先，先將輸入影像透過顏色辨識的方式與背景相減法平行處理後，在畫面上即可得到目標物大致的形狀，接著再利用中值濾波器將不需要的雜訊點濾除乾淨，為了改善目標物有些許斷開的情形發生，故加入形態學將目標物膨脹些許大小，透過以上處理過後，希望能找出影像中目標物中心點，故使用影像處理常見的邊緣偵測方式，使畫面更簡潔，最後找到目標物之輪廓後，僅需在目標物邊緣找到四邊最大的點，將這些點組成矩形圖案把目標物框選起來並求得中心點。整體流程圖如圖4.20表示

利用前幾節介紹之影像處理法去擷取目標物流程圖如下：

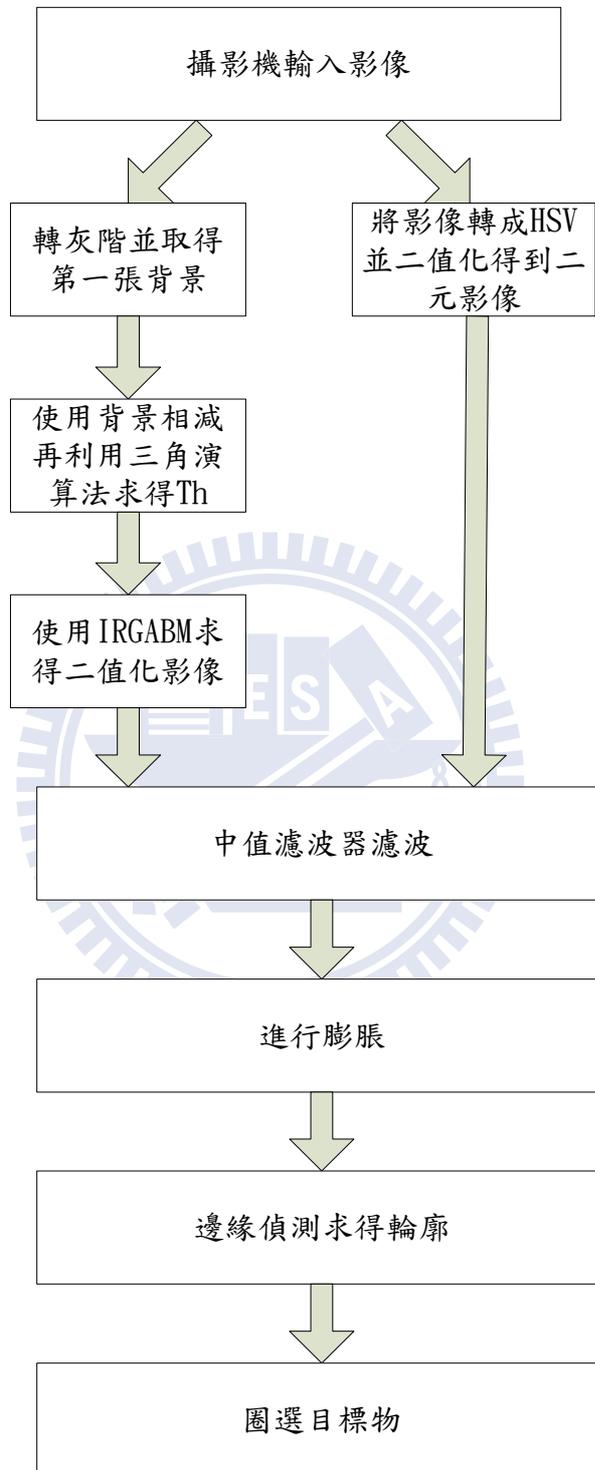


圖 4.20 影像擷取流程圖

## 4.7 ROI(region of interest)選擇

在整個影像擷取目標物過程中，處理像素範圍為 $640 \times 480$ ，處理時間經過計算大約為46ms，但是攝影機輸入影像頻率為每秒30張影像，換句話說，大約每隔33ms就會得到一張新影像，如此一來，單單只考慮影像處理部分就無法在一個取樣時間中完成，會造成新影像流失，故為了避免此情形發生，採用小畫面處理，即是減少ROI區域。

一開始採用全域( $640 \times 480$ )影像處理，當前一張影像已擷取到目標物，此時把ROI區域縮減並以偵測到目標物之中心為新ROI區域中點，故小畫面是會隨著追蹤到目標物而移動處理區域。由於目標物在縱向移動距離較大，故設定新ROI區域為 $100 \times 300$ ，如此一來即可使影像處理時間大幅縮短。假設在追蹤目標物過程中，出現追蹤失敗情形發生時，立即轉換為全域影像處理，以防止一次追蹤失敗後，即再也無法追蹤到目標物的情形發生。簡單流程圖如下：

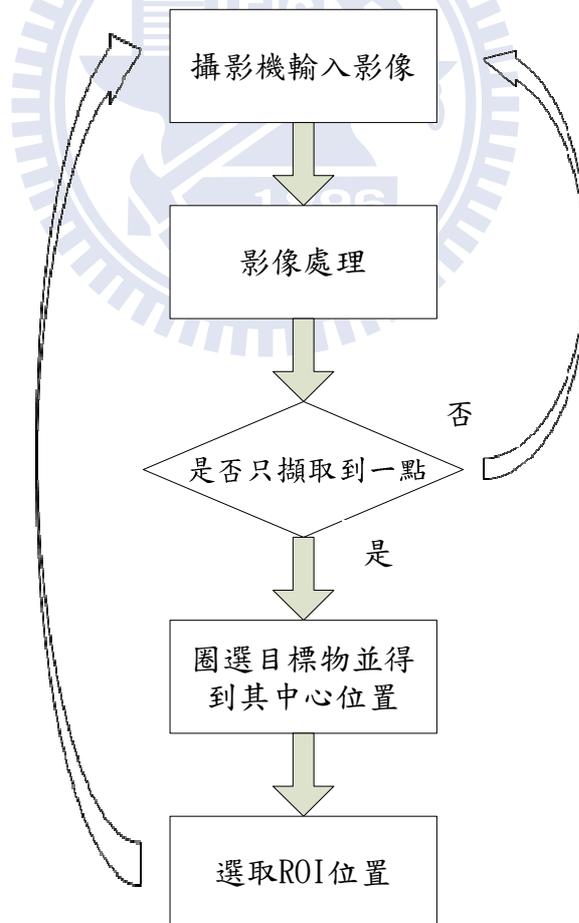


圖 4.21 ROI 選擇流程圖

## 第五章 校正與最佳擊球點

機械手臂與影像處理方面，個別都已運作正常的情況下，接著要將攝影機影像與真實空間做結合，需要先進行攝影機校正處理，本論文採用[3]所提到的校正方式，雖然不是最新穎的校正方式，但在校正成果上有不錯的表現，故採用此法即可。另外，在使用單眼系統的情況下，加入了拋物線方程式與校正矩陣一同求解物體在飛行時的拋物線軌跡參數預測其物體落點。由於物體飛行速度相對於機械手臂移動速度過快，受限於機械手臂速度上的限制，故以機械手臂在可擊打區域內移動最小角度為前提，計算出最佳擊球點的位置，並利用影像擷取點持續更新的方式，不斷地計算出新的參數並修正擊球點位置。

### 5.1 校正

每一部相機在硬體製作過程中會產生獨特的內部相關參數，而由於相機擺設方式的不同，也可以得到不同的外部參數。為了瞭解這些相關參數，本文先介紹幾何相機模型，採用常見的透視投影方式，再經由一連串運算後，求出其相關的校正矩陣，透過此校正矩陣，即可進行真實世界座標與相機二維影像平面座標的轉換。

#### 5.1.1 幾何相機模型

假設影像為針孔的方式投影，如圖 5.1 所示

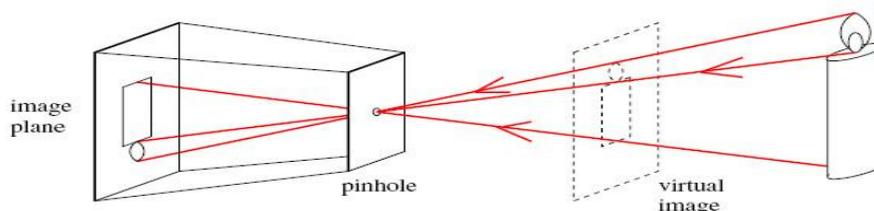


圖 5.1 針孔投影模型

在三維空間轉換至二維空間中，可能物體長度不同，在不同距離下，卻出現相同的成像如圖 5.2 所示，雖然 B 與 C 長度很明顯的不同，但由於 OB 距離是 OC 距離的兩倍，故最後成像時，B' 與 C' 長度卻是相等的。在這部分的錯誤，是很難避免的。由此可知，三維真實空間轉二維影像平面也許誤差不大，但當二維影像平面轉三維真實空間卻有其難度。

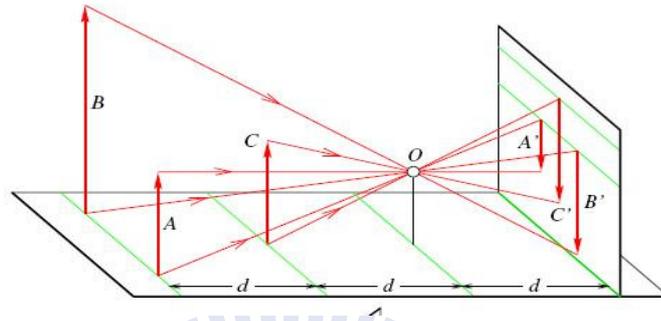


圖 5.2 距離對成像的影像

接著，如圖 5.3 所示，真實座標系統與圖 3.1 相同。物體與原點縱深距離為  $z$ ，橫向距離為  $x$ ，高低距離為  $y$ 。

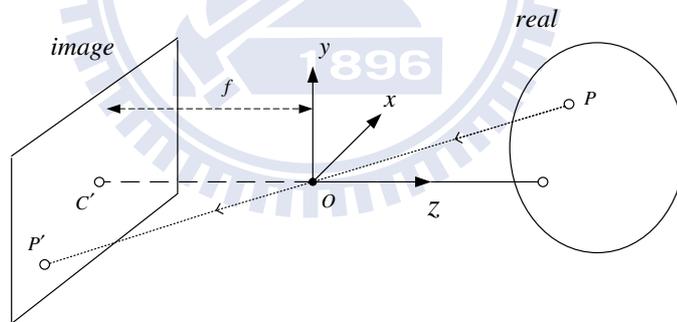


圖 5.3 物體投影示意圖

由圖 5.3 可知，真實物體  $P(x, y, z)$  點透過原點  $O$  投影至影像平面上  $P'(u, v, z')$ ，其中  $z' = f$ 。由圖中觀察可知， $\overline{OP'} = \lambda \overline{OP}$ ，故可推得(5.1.1)式

$$\begin{cases} \mathbf{u} = \lambda \mathbf{x} \\ \mathbf{v} = \lambda \mathbf{y} \\ \mathbf{f} = \lambda \mathbf{z} \end{cases} \Leftrightarrow \lambda = \frac{\mathbf{u}}{\mathbf{x}} = \frac{\mathbf{v}}{\mathbf{y}} = \frac{\mathbf{f}}{\mathbf{z}} \Rightarrow \begin{cases} \mathbf{u} = \mathbf{f} \frac{\mathbf{x}}{\mathbf{z}} \\ \mathbf{v} = \mathbf{f} \frac{\mathbf{y}}{\mathbf{z}} \end{cases} \quad (5.1.1)$$

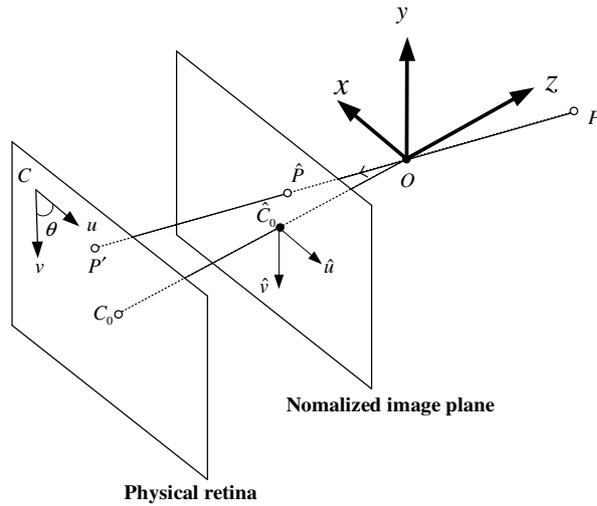


圖 5.4 實際相機內部座標系統

由圖 5.4 可觀察出，因為相機影像並非正方形的關係，所以長短軸不同，故將(5.1.1)式分別乘上增加兩個參數  $k$  與  $l$ ，單位為  $pixel/m$ ，並令  $\alpha = kf$  與  $\beta = lf$ ，可將(5.1.1)式改寫為

$$\begin{cases} \mathbf{u} = \alpha \frac{x}{z} \\ \mathbf{v} = \beta \frac{y}{z} \end{cases} \quad (5.1.2)$$

由於影像中心點會有所誤差，並非在正中心，故分別加入誤差值  $u_0$  與  $v_0$ ，改寫(5.1.2)式為

$$\begin{cases} \mathbf{u} = \alpha \frac{x}{z} + u_0 \\ \mathbf{v} = \beta \frac{y}{z} + v_0 \end{cases} \quad (5.1.3)$$

在相機製作過程中，影像正交軸  $u$  與  $v$  通常不會剛好呈現垂直的關係，而會呈現  $\theta$  度夾角的關係，由於此原因，修正(5.1.3)式後可得

$$\begin{cases} \mathbf{u} = \alpha \frac{x}{z} - \alpha \cot\theta \frac{y}{z} + u_0 \\ \mathbf{v} = \frac{\beta}{\sin\theta} \frac{y}{z} + v_0 \end{cases} \quad (5.1.4)$$

由於(5.1.4)式中有  $u_0, v_0$  的關係，希望能將式子化為簡單的  $Ax=b$  形式，故在此採用齊次座標系統。令影像平面座標  $p=(u, v, 1)$ ，真實座標  $P=(x, y, z, 1)$ ，將(5.1.4)式改寫為矩陣的形式

$$\mathbf{p} = \frac{1}{z} \begin{bmatrix} \alpha & -\alpha \cot \theta & u_0 & 0 \\ 0 & \frac{\beta}{\sin \theta} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{P} = \frac{1}{z} \mathbf{M} \mathbf{P} = \frac{1}{z} [\mathbf{K} \quad \mathbf{0}] \mathbf{P} \quad (5.1.5)$$

從(5.1.4)式可以了解相機內部有五個參數，分別為  $\alpha$ 、 $\beta$ 、 $\theta$ 、 $u_0$ 、 $v_0$ 。將這些內部參數再結合六個外部參數，分別為對  $x, y, z$  軸旋轉與對  $x, y, z$  軸平移，即為相機模型所涵蓋的 11 個參數。將(5.1.5)式結合外部參數後，可得到

$$\mathbf{p} = \frac{1}{z} \mathbf{M} \mathbf{P} = \mathbf{K} [\mathbf{R} \quad \mathbf{t}] \mathbf{P} \quad (5.1.6)$$

其中  $\mathbf{M}$  表示為  $3 \times 4$  校正矩陣， $\mathbf{K}$  為  $3 \times 3$  內部參數矩陣， $\mathbf{R}$  為  $3 \times 3$  對三軸的旋轉矩陣， $\mathbf{t}$  為  $3 \times 1$  對三軸平移向量。

最後將(5.1.6)簡化可得到影像平面座標與真實座標關係式為

$$\begin{cases} \mathbf{u} = \frac{\mathbf{m}_1 \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}} \\ \mathbf{v} = \frac{\mathbf{m}_2 \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}} \end{cases} \quad (5.1.7)$$

其中

$$\mathbf{M} = \begin{bmatrix} \alpha r_1^t - \alpha \cot \theta r_2^t + u_0 r_3^t & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \frac{\beta}{\sin \theta} r_2^t + v_0 r_3^t & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ r_3^t & t_z \end{bmatrix} \quad (5.1.8)$$

其中  $m'_1, m'_2, m'_3$  分別為 M 校正矩陣三個列向量， $r'_1, r'_2, r'_3$  分別為 R 旋轉矩陣三個列向量， $t_x, t_y, t_z$  分別為 t 平移向量， $z = m_3 \cdot P$ 。

### 5.1.2 幾何相機校正

在前一節(5.1.7)式得到真實世界座標與影像平面座標的關係式，故此節可利用此關係式將外部量得之真實座標值與相機影像中觀察得到的影像座標值代入關係式，即可求得相機參數，達到校正的目標。其中，影像平面座標原點為左上角處，在首先將(5.1.7)式移項改寫後可得

$$\begin{cases} (\mathbf{m}_1 - \mathbf{u}_i \mathbf{m}_3) \cdot \mathbf{P}_i = 0 \\ (\mathbf{m}_2 - \mathbf{v}_i \mathbf{m}_3) \cdot \mathbf{P}_i = 0 \end{cases} \quad i=1,2,\dots \quad (5.1.9)$$

整理過後寫成矩陣形式得到

$$\mathbf{Pm} = \mathbf{0} \Rightarrow \begin{bmatrix} \mathbf{P}_1^t & \mathbf{0}^t & -\mathbf{u}_1 \mathbf{P}_1^t \\ \mathbf{0}^t & \mathbf{P}_1^t & -\mathbf{v}_1 \mathbf{P}_1^t \\ \dots & \dots & \dots \\ \mathbf{P}_n^t & \mathbf{0}^t & -\mathbf{u}_n \mathbf{P}_n^t \\ \mathbf{0}^t & \mathbf{P}_n^t & -\mathbf{v}_n \mathbf{P}_n^t \end{bmatrix} \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} = \mathbf{0} \quad (5.1.10)$$

欲求得校正矩陣中  $m_1, m_2, m_3$ ，只是單純得到  $m = 0$  是沒有任何意義的，故必須求解  $\mathbf{Pm}$  絕對值平方的最小解，並設置  $m$  長度為 1，也就是  $\min \|\mathbf{Pm}\|^2$ ，利用求解  $P^t P$  的特徵值與特徵向量的方式即可求得其最小值。

更進一步求出每一個內部參數值，假設(5.1.10)式求出的  $M = [A \quad b]$ ，將其乘以一個

參數  $\rho$  得到

$$\rho[\mathbf{A} \quad \mathbf{b}] = \mathbf{K}[\mathbf{R} \quad \mathbf{t}] \Leftrightarrow \rho \begin{bmatrix} \mathbf{a}_1^t \\ \mathbf{a}_2^t \\ \mathbf{a}_3^t \end{bmatrix} = \begin{bmatrix} \alpha \mathbf{r}_1^t - \alpha \cot\theta \mathbf{r}_2^t + \mathbf{u}_0 \mathbf{r}_3^t \\ \frac{\beta}{\sin\theta} \mathbf{r}_2^t + \mathbf{v}_0 \mathbf{r}_3^t \\ \mathbf{r}_3^t \end{bmatrix} \quad (5.1.11)$$

其中  $\rho = \frac{\varepsilon}{|\mathbf{a}_3|}$  ,  $\varepsilon = \pm 1$  ,  $\mathbf{a}_1^t, \mathbf{a}_2^t, \mathbf{a}_3^t$  為  $\mathbf{A}$  矩陣的列向量

可推導得到其他內部參數分別為

$$\begin{cases} \mathbf{u}_0 = \rho^2 (\mathbf{a}_1 \cdot \mathbf{a}_3) \\ \mathbf{v}_0 = \rho^2 (\mathbf{a}_2 \cdot \mathbf{a}_3) \\ \alpha = \rho^2 |\mathbf{a}_1 \times \mathbf{a}_3| \sin\theta \\ \beta = \rho^2 |\mathbf{a}_2 \times \mathbf{a}_3| \sin\theta \\ \cos\theta = - \frac{(\mathbf{a}_1 \times \mathbf{a}_3) \cdot (\mathbf{a}_2 \times \mathbf{a}_3)}{|\mathbf{a}_1 \times \mathbf{a}_3| |\mathbf{a}_2 \times \mathbf{a}_3|} \end{cases} \quad (5.1.12)$$

在此需注意“ $\times$ ”表示為外積的意思，並非乘法。

而外部參數則為

$$\begin{cases} \mathbf{r}_3 = \rho \mathbf{a}_3 \\ \mathbf{r}_1 = \frac{\rho^2 \sin\theta}{\beta} (\mathbf{a}_2 \times \mathbf{a}_3) = \frac{1}{|\mathbf{a}_2 \times \mathbf{a}_3|} (\mathbf{a}_2 \times \mathbf{a}_3) \\ \mathbf{r}_2 = \mathbf{r}_3 \times \mathbf{r}_1 \end{cases} \quad (5.1.13)$$

$$\mathbf{t} = \rho \mathbf{K}^{-1} \mathbf{b} \quad (5.1.14)$$

## 5.2 預測物體飛行軌跡

將攝影機校正完畢後，即可進行真實空間與影像平面的轉換。接著為了可以進行擊球的動作，必須先預測物體在飛行時的軌跡，後續才有辦法可以選擇想要的擊球位置。

在做相機校正的時候，由於是從三維空間轉換到二維空間，故只需要兩條方程式(5.1.9)即可，但在預測物體飛行軌跡時，由於採用單眼視覺系統，在單一的二維影像平面轉換到三維真實空間的過程中，無法得到**縱深**的資訊，故光靠(5.1.9)兩條方程式是不夠的，必須再添加其他方程式輔助。考慮物體出手後，呈現拋物線的方式行進，所以再加入三條方程式一同運算，藉此彌補單眼視覺無法得到縱深資訊的缺點。

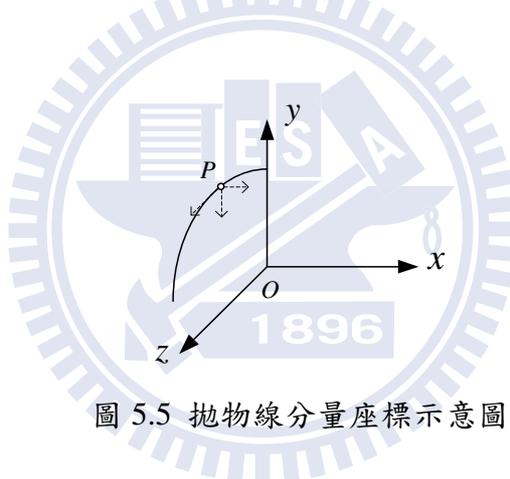


圖 5.5 拋物線分量座標示意圖

拋物線方程式

$$\begin{aligned}
 x_i &= c_{x1}t_i + c_{x0} & ,i=1\sim n \\
 y_i &= c_{y2}t_i^2 + c_{y1}t_i + c_{y0} & ,i=1\sim n \\
 z_i &= c_{z1}t_i + c_{z0} & ,i=1\sim n
 \end{aligned}
 \tag{5.2.1}$$

其中  $i$  代表所擷取的點數， $t_i$  則代表在第  $i$  個擷取點時的時間， $c_{y2} = -\frac{1}{2}g$  為常數，將(5.1.9)式與(5.2.1)式整合後，可觀察出變數有  $3n+6$  個，而方程式則有  $5n$  條，欲解出所有變數，必須滿足**變數**小於**方程式**的條件，故依照此條件推得，當擷取的點數大於等



經過化簡後，矩陣行數不會再隨著擷取點數的不同而變化，維持著  $2n \times 6$  比例變化，求解出  $x$  即為物體飛行軌跡之參數。當求取點數大於 3 點時，即採用 linear least-squares 的方式求解。

### 5.3 影像擷取點數策略

在實作中，取得越多點，可以減低其誤差，增加其軌跡預測參數的精確度，故盡量擷取越多點越好，但隨著擷取點數的提高，所花費的時間相對也越長。另外，隨著影像程式啟動後，攝影機透過影像處理即不斷地擷取到目標物，此時，目標物可能未出手，假使將此擷取點參入一併計算，則可能使物體飛行軌跡產生很大的誤差，故如何從影像平面座標數值中，判斷目標物是否已經出手，即是本節所要探討的部分。

首先嘗試最簡易的方法，利用影像中目標物的半徑來判斷其是否出手。

假使目標物朝攝影機投擲過來，毫無疑問的，目標物在影像平面中半徑將漸漸增加，故當影像平面中半徑數值變大時，就將此點列入計算，設定收集固定擷取點數 ( $n > 3$ )，一旦收集數目到達設定，及計算出飛行軌跡參數。此法優點為運算速度較快，缺點是當目標物在遠方時，目標物很小，影像平面中半徑數值可能由於雜訊干擾，使半徑變小或相同時，將產生問題。

有鑑於第一種方式缺點，故改用第二種方式，即為從一開始便設定固定擷取點數 ( $n > 3$ )，不斷地持續計算軌跡參數，由於在還未出手時，手部呈現左右搖晃動作，此時計算出來的飛行軌跡參數，在橫向速度上可能有較高的數值，而縱向距離與速度的數值顯得相對的小，如此一來，藉由一些簡單的條件式，如縱向速度須大於多少，或是橫向速度須小於多少...等來判斷計算出來的飛行軌跡參數是否符合實際狀況，即可判斷出目標物是否出手，或只是還在手上搖晃未出手。此法優點在於不考慮影像平面中目標物的

大小，故不會受到雜訊或其他的干擾，缺點則是因為需不斷地計算飛行軌跡參數，所以程式計算速度上較慢一些。

改進第二種方式，同樣從一開始便不斷地計算飛行軌跡參數，也採用條件式來判斷是否已經出手，唯一不同點在於，第二種方式擷取點數為固定的，假設一開始設定太少擷取點數如 3 點，將造成飛行軌跡參數估測誤差極大，如果設定擷取點數過大，則需要耗費太多計算時間，有時甚至無法擷取到該點數導致無法計算出飛行軌跡參數。在此考量下，使用第三種方式，一開始先設定少數點數便開始計算飛行軌跡參數，假設參數符合條件式，便加入新擷取到的 1 點一併計算，以此類推，如果不斷地新擷取到目標物且皆可以符合條件式，則可以反覆地修正飛行軌跡參數，假使加入到某個新擷取點後，就不符合條件式，即恢復至最一開始最低的擷取點數，重新計算。此法可比第二種耗費較少的運算時間，又可以不斷更新新的擷取點來修正飛行軌跡參數，故最後實驗時採用此種方式。

表 1 持續更新擷取點法

frame 數	第 n 個 Loop	第 n+1 個 Loop		第 n+2 個 Loop	
		符合條件式	不符合條件式	符合條件式	不符合條件式
1	x	x		x	
2	x	x	x	x	
3	x	x	x	x	x
4		x	x	x	x
5				x	x

在表 1 顯示第三種方法，其中 x 代表有擷取到並且加入計算飛行軌跡參數之點數。舉例說明：在第 n 個 loop 時，分別使用 frame 1,2,3 中擷取到的點數來計算飛行軌跡參數，假設初始設定最少擷取點數為 3 點，當第 n 個 loop 時已經擷取到 3 點，第 n+1 個

loop 時，假使有符合條件式，則加入新的 1 點擷取點，如果沒有，則重新計算 3 點擷取點，以此類推只要滿足條件式則加入新的擷取點，反之，不滿足則回到設定的最少擷取點重新計算。

## 5.4 最佳擊球點

當擷取點收集完成，預測出物體飛行軌跡參數時，即可求得目標物在任何一個時間點的三維真實空間座標位置。接下來須規劃擊球點位置，由於在硬體實現方面，機械手臂實驗平台速度上受到限制，目標物飛行時間大約 0.4~0.5 秒，而機械手臂大軸或小軸個別旋轉 90 度約需 0.5~0.6 秒，相較之下機械手臂速度偏慢，再扣掉影像擷取點最少需 3 點(約花費 0.1 秒)的情形下，機械手臂大約只有 0.3~0.4 秒的時間可以動作到達擊球點位置，故在此以機械手臂旋轉最少角度為前提來選擇最佳擊球點。另外，以機械手臂在 0.3 秒內所能移動範圍來設置其可擊打區域，故機械手臂初始擺放的擊打位置將影響機械手臂的可擊打區域，所以也必須列入考量範圍。

以圖 5.6 為例，此為真實空間座標側視圖，y 軸為機械手臂基座所在處，目標物從 4 米處朝向機械手臂投擲，影像處理擷取 3 點計算出飛行軌跡後，傳送資料給機械手臂，忽略傳送資料時間，機械手臂從 B 開始約有 0.3 秒可動作的時間，如果擊球點位置需花費 0.4 秒移動，則超出機械手臂可擊打範圍，即手臂動作太慢導致無法擊打到目標物。

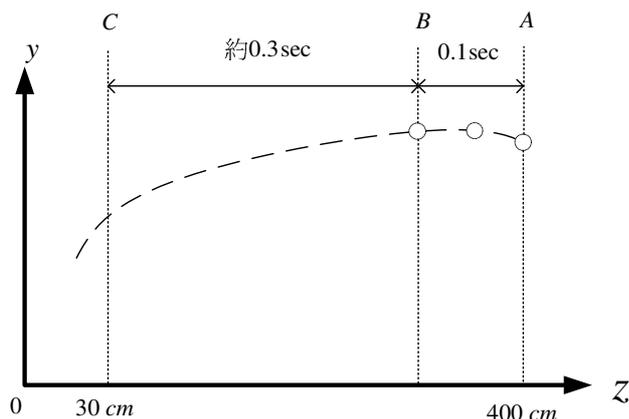


圖 5.6 物體飛行時間示意圖

以旋轉最小角度為考量，故在角度平面中考慮其最佳擊球點位置，當目標物飛入機械手臂工作區內時，可將真實空間座標位置 $(x,y,z)$ 轉換為機械手臂的角度座標 $(\theta_1, \theta_2)$ 。將拋物線方程式(5.2.1)式代入(3.3.13)式轉換為角度平面後，取 $\|\cdot\|_1$  再對時間微分嘗試求出一條公式解來得到旋轉最小角度解。

詳細過程如下表示，取 $\|\cdot\|_1$  對 $t$ 微分

$$\frac{d}{dt} [|\theta_1(t) - \theta_{10}| + |\theta_2(t) - \theta_{20}|] = 0 \quad (5.4.1)$$

為了使計算能更簡便，加上在實驗過程中觀察可發現，希望使機械手臂在進行擊打目標物時，皆只有向前擊打，也就是說機械手臂大小軸旋轉角度持續增加，故令 $\theta_1(t) > \theta_{10}$  ,  $\theta_2(t) > \theta_{20}$  則

$$\frac{d}{dt} [\theta_1(t) - \theta_{10} + \theta_2(t) - \theta_{20}] = 0 \quad (5.4.2)$$

將(3.3.13)式代入後可得到

$$\frac{d}{dt} \left( \tan^{-1} \left( \frac{P_{ya}}{P_{za}} \right) - \cos^{-1} \left( \frac{\sqrt{P_{ya}^2 + P_{za}^2}}{2l_1} \right) + \cos^{-1} \left( \frac{P_{ya}^2 + P_{za}^2 - 2l_1^2}}{2l_1^2} \right) - \theta_{10} - \theta_{20} \right) = 0 \quad (5.4.3)$$

其中(3.3.13)式中的  $P_{ya}, P_{za}$  代表機械手臂位置與(5.2.1)式的  $x, y, z$  代表目標物位置，兩者皆是基於真實座標系統，但在實作時，機械手臂原點分別向  $x$  方向移動  $Arm_x$  距離，向  $y$  方向移動  $Arm_y$  距離，向  $z$  方向移動  $Arm_z$  距離，如圖 5.7 表示，故  $P_{za} + Arm_z = z(t)$ ， $P_{ya} + Arm_y = y(t)$ ，將(5.4.3)式展開後可得

$$(4I_1^2 k - k^2)^{\frac{1}{2}} (P'_{ya} P_{za} - P'_{za} P_{ya}) - k (P'_{ya} P_{ya} - P'_{za} P_{za}) = 0 \quad (5.4.4)$$

其中  $k = P_{ya}^2 + P_{za}^2$

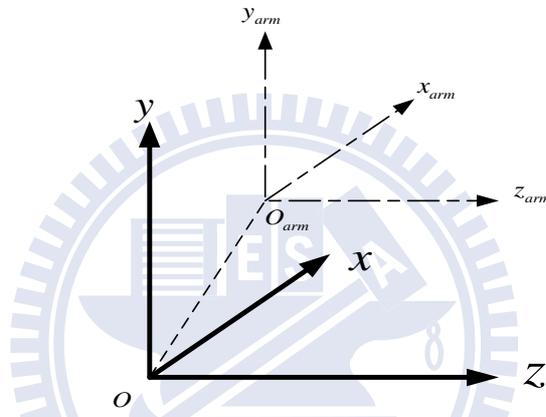


圖 5.7 手臂實作時原點平移圖

經運算過後，發現(5.4.4)式難以求解出  $t$  的公式解，故採用牛頓法疊代的方式並使用  $\|\cdot\|_2$  來求取最佳近似解。方程式如下：

$$\begin{bmatrix} \theta_{1,k+1} \\ \theta_{2,k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} \theta_{1,k} \\ \theta_{2,k} \\ \lambda_k \end{bmatrix} - \gamma H^{-1}(\theta_{1,k}, \theta_{2,k}, \lambda_k) \nabla F(\theta_{1,k}, \theta_{2,k}, \lambda_k) \quad (5.4.5)$$

其中  $\gamma$  代表 step，可自行調整，大小將影響收斂速度，且過大有可能造成無法收斂發生震盪，過小除了收斂速度慢以外，可能僅能收斂至區域最小值。

其中

$$\mathbf{F} = (\theta_1(t) - \theta_{10})^2 + (\theta_2(t) - \theta_{20})^2 - \lambda h \quad (5.4.6)$$

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 \mathbf{F}}{\partial \theta_1^2} & \frac{\partial^2 \mathbf{F}}{\partial \theta_1 \partial \theta_2} & \frac{\partial^2 \mathbf{F}}{\partial \theta_1 \partial \lambda} \\ \frac{\partial^2 \mathbf{F}}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 \mathbf{F}}{\partial \theta_2^2} & \frac{\partial^2 \mathbf{F}}{\partial \theta_2 \partial \lambda} \\ \frac{\partial^2 \mathbf{F}}{\partial \lambda \partial \theta_1} & \frac{\partial^2 \mathbf{F}}{\partial \lambda \partial \theta_2} & \frac{\partial^2 \mathbf{F}}{\partial \lambda^2} \end{bmatrix} \quad (5.4.7)$$

其中  $h$  是將拋物線方程式(5.2.1)經過座標原點平移後轉換為  $\theta_1, \theta_2$  的函式，由於存在限制條件式  $h$ ，故加入 Lagrange 參數  $\lambda$ 。限制條件式詳細表示為

$$\begin{aligned} h(\theta_1, \theta_2) = & \frac{l_1^2 c_{y2}^2}{c_{z1}^2} (\cos^2(\theta_1 + \theta_2) + \cos^2(\theta_1)) + \cos(\theta_1 + \theta_2) \left[ \frac{2l_1^2 c_{y2}^2}{c_{z1}^2} \cos\theta_1 \right. \\ & \left. - \frac{2l_1 c_{y2} (c_{z0} - \text{Arm}z)}{c_{z1}^2} + \frac{l_1 c_{y1}}{c_{z1}} \right] + \cos\theta_1 \left( \frac{l_1 c_{y1}}{c_{z1}} - \frac{2l_1 c_{y2} (c_{z0} - \text{Arm}z)}{c_{z1}^2} \right) \\ & + \left[ \frac{c_{y2} (c_{z0} - \text{Arm}z)^2}{c_{z1}^2} - \frac{c_{y1} (c_{z0} - \text{Arm}z)}{c_{z1}} + (c_{y0} - \text{Army}) \right] \\ & - l_1 (\sin(\theta_1 + \theta_2) - \sin\theta_1) = 0 \end{aligned} \quad (5.4.8)$$

雖然使用此牛頓法來求解最佳近似解，在結果方面有收斂到極值上，也就是微分等於 0 處，但是在觀察圖形後發現，如圖 5.8 與圖 5.9 表示，在限制條件式(5.4.8)的影響下，欲得到的最小值，容易因為初始角度值選取的不同，收斂至其他極值處，故此法雖然也可求得一組解，但卻可能不是真正最小解。

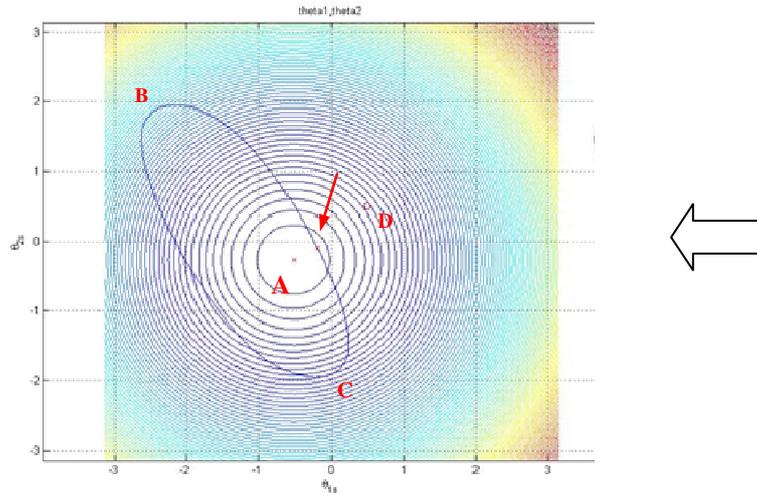


圖 5.8  $\theta_1, \theta_2$  平面

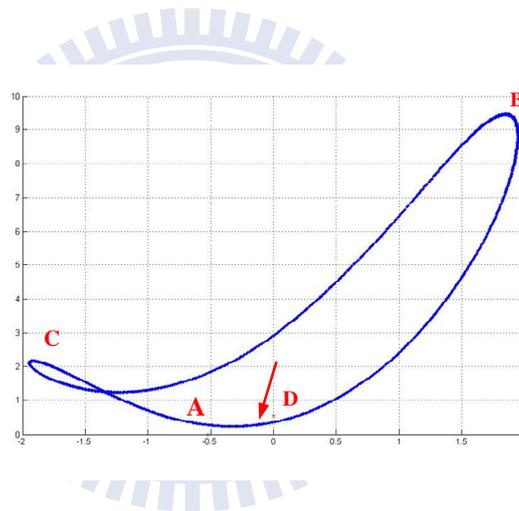


圖 5.9  $h(\theta_1, \theta_2)$  函式之角度代入 2-norm 結果

如圖 5.8 可看到由 A 點組成的彩色同心圓代表(5.4.9)式平方投影至  $\theta_1, \theta_2$  平面上，A 點為最低處越往外圍越高，呈現漏斗的形狀，藍色橢圓則代表將拋物線方程式轉換到  $\theta_1, \theta_2$  平面， $\theta$  單位皆為 rad/sec，也就是(5.4.8)式，紅色箭頭處為真正最小值。A 點為  $\theta_{10}, \theta_{20}$  手臂初始位置之角度，將圖 5.8 畫為立體側視圖(由圖 5.8 旁箭頭看入)表示為圖 5.9，橫軸為  $\theta_2$ ，縱軸則為  $\|\cdot\|_2$ 。由圖 5.9 可清楚看出 B 點與 C 點為微分為 0 處，D 點為疊代法起始代入的初始角度。由於牛頓法對初始角度 D 點相當敏感，選取過程中，假設較接近 B 點或 C 點，則可能造成收斂結果與欲得到結果不同的情況發生，加上  $\gamma$  參數設

定不能過大，使疊代至收斂次數遽增，以及計算(5.4.7)式需要花費大量的時間，故在追求執行速度的實驗過程中，採取別種方式計算最小角度解。

由於牛頓法運算過程較為耗時，故直接採用最簡易的方式，將時間分割成數千點，再分別將所有時間取樣點都代入 $\|\cdot\|_2$ ，直接比較最小的 $\|\cdot\|_2$ 值即可。

首先，將拋物線方程式(5.2.1)座標系統經過原點平移，即 $P_{za} = z(t) - Armz$ ， $P_{ya} = y(t) - Army$ ，接著代入(3.3.13)式使方程式 $\theta_1, \theta_2$ 成為 $t$ 的函數，再將 $\theta_1, \theta_2$ 代入

$$\theta_{\min} = \sqrt{(\theta_1(t) - \theta_{10})^2 + (\theta_2(t) - \theta_{20})^2} \quad (5.4.9)$$

將所有取樣時間點分別代入(5.4.9)式中，即可得到最小角度值及其時間點，並得到真實空間中座標位置，此座標位置也就是最佳擊球點位置。

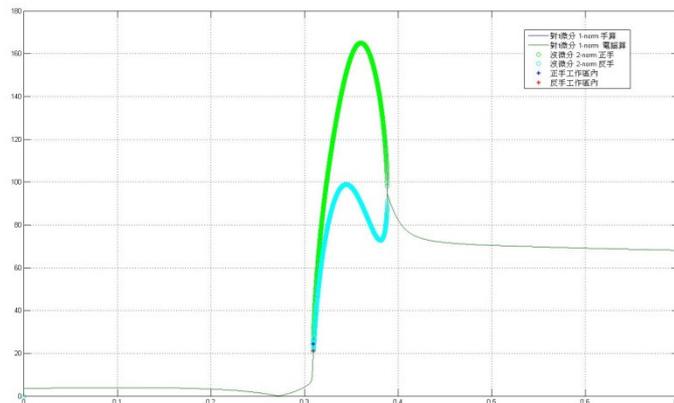


圖 5.10 2-norm 與時間關係圖

將時間取樣點代入(5.4.9)式，可畫出圖 5.10，其中橫軸為時間，單位為秒，縱軸為 $\|\cdot\|_2$ ， $\theta$ 單位為度，圖中粗線部分代表機械手臂可擊打之時間點，綠色則為機械手臂正手結果，淺藍色則為機械手臂反手結果，藍色”\*”與紅色”\*”分別為機械手臂正手與反手求得最小角度解，並將求得結果表示為真實空間模擬，如圖 5.11 表示

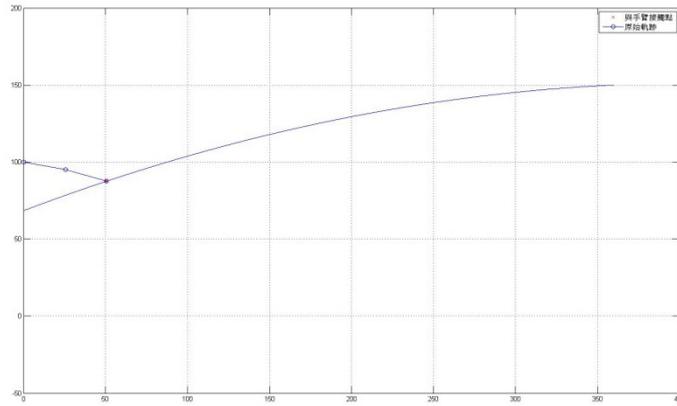


圖 5.11 真實空間擊打模擬圖

其中橫軸為 z 方向縱深，縱向為 y 方向高低，機械手臂基座設為圖中(0,100)處，擊球點大約在圖中(50,70)處，大小軸角度分別為 -11.1 度與 -5.5 度。



## 第六章 實驗結果與分析

將前幾章理論部分實現於實際機械手臂系統，並在這一章節介紹完整的實驗結果，以及各項硬體設施與座標軸系統。

### 6.1 硬體設施與流程

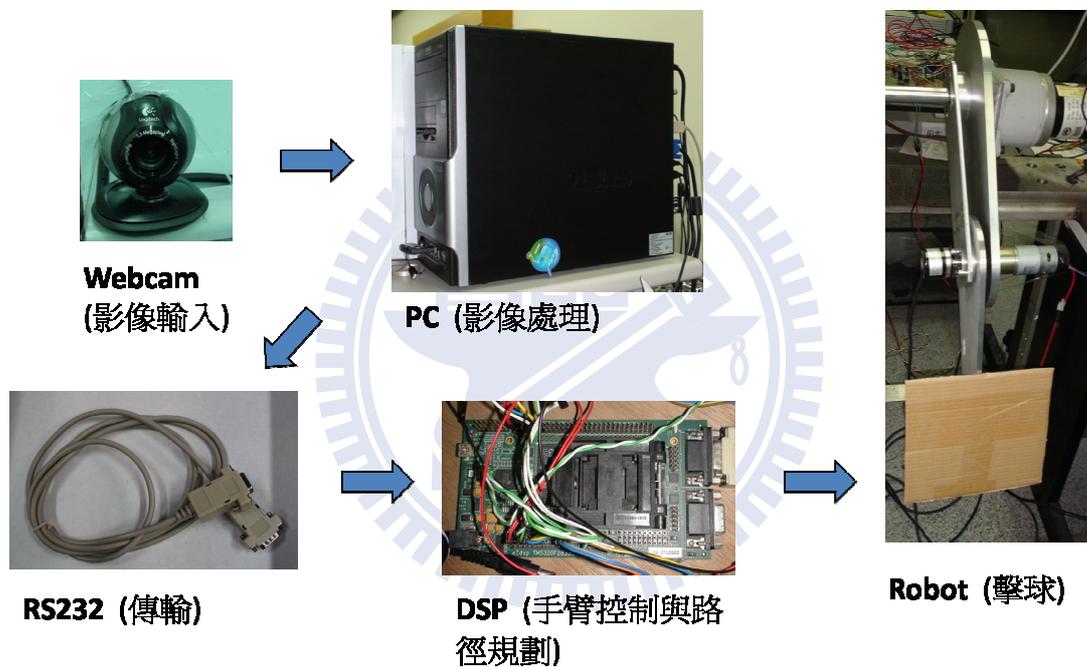


圖 6.1 硬體設施

相關硬體規格如下：

PC 效能	: cpu 單核心 3G Hz
	RAM 2GB
DSP 板	: TI TMS320F28335
視訊鏡頭	: 30 fps
機械手臂取樣時間	: 100 Hz
手臂全長	: 52 cm

實驗用球	: 直徑約為 7~8 cm
丟球距離	: 約 3.5~4 m
球拍大小	: $21 \times 20 \text{ cm}^2$

首先由攝影機鏡頭輸入影像畫面給 PC 進行影像處理，包含擷取出目標物及預測出最佳擊球點位置，並透過 RS232 傳輸給 DSP 板進行路徑規畫與手臂控制，再將計算出的手臂關節控制電壓發送 PWM 給機械手臂進行擊打的工作。

## 6.2 座標系統

真實空間中絕對原點位於圖 6.2 左下角處，以左邊牆壁為  $x=0$ ，地面為  $y=0$ ，攝影機鏡頭所在位置為  $z=0$ ，以牆壁為原點的座標系統，稱為真實座標系，用  $x-y-z$  表示，機械手臂基座中心位於真實座標系統  $x=106 \text{ cm}$ ,  $y=95 \text{ cm}$ ,  $z=-85 \text{ cm}$  之位置，將真實座標系統平移至以機械手臂基座中心為原點，此新的座標系統稱為手臂座標系，用  $x_a-y_a-z_a$  表示，如圖 6.3 所示，機械手臂在  $y_a-z_a$  平面上運動。

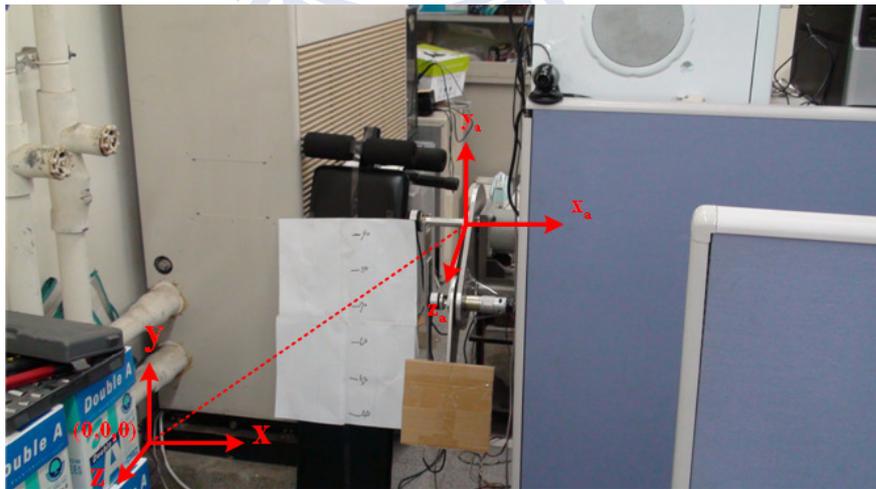


圖 6.2 座標系統(正視圖)



圖 6.3 座標系統(側視圖)

### 6.3 機械手臂路徑規畫實驗

將第三章所提到的路徑規畫，進行實際測試，包含輸入單一座標位置，或是多點座標位置，在不同初始速度與終止速度，時間不同的各種情況下進行測試。並且比較使用電腦模擬與實際執行結果，兩者是否能相同，以驗證理論是否有錯誤。

表 2 機械手臂硬體參數

種類	數值	單位
上臂大軸的長度	0.26	m
下臂小軸的長度	0.26	m
上臂大軸的質量	3.885	kg
下臂小軸的質量	0.867	kg

先測試單一座標位置，以機械手臂系為座標系統，輸入  $P_{za} = 0.26m$ ， $P_{ya} = 0.26m$ ，初始速度與終止速度皆為 0 度/秒，終止時間為 0.5 秒。

比較電腦模擬與實際執行後兩者結果：

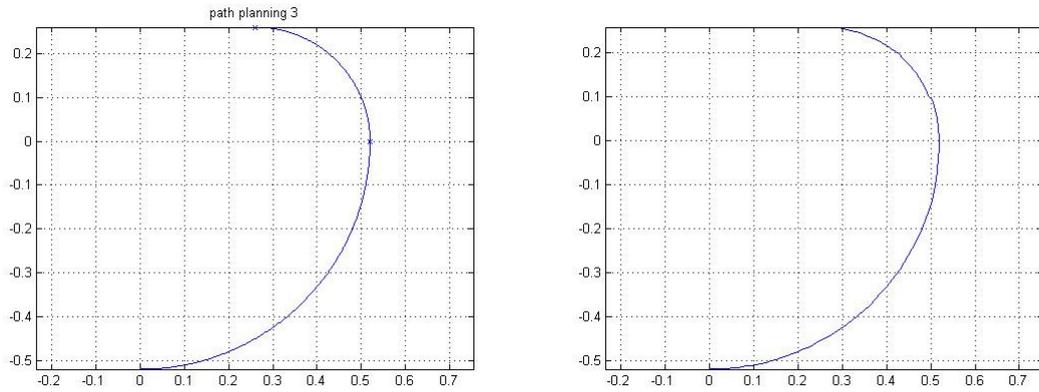


圖 6.4 路徑規劃 單點模擬圖(左)與實際數據(右)

其中橫軸為真實空間座標  $z_a$  方向，單位為公尺，縱軸則為  $y_a$  方向，單位為公尺。機械手臂基座中心為座標(0,0)，初始手臂角度大軸為-90 度，小軸為 0 度，換算成座標位置即是(0,-0.52)。

圖 6.4 左圖是使用 MATLAB 軟體模擬出來結果，右圖則是機械手臂移動出來的真實量測數據。觀察發現，兩者之間幾乎完全符合，沒有任何差別，故路徑規劃結果能如預期一般執行。

接著進行多點座標位置實驗， $P_{za}$  與  $P_{ya}$  依順序分別輸入 A(0.52,0)、B(0.15,0.35)、C(0.35,0.2)、D(0.35,-0.2)，座標單位皆為公尺，初始速度與終止速度則皆為 30 度/秒，終止時間訂為 0.5 秒，比較電腦模擬與實際執行後兩者結果：

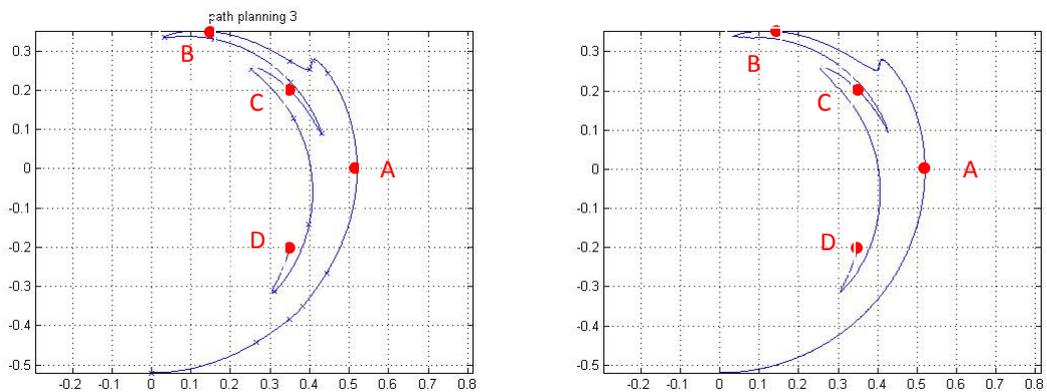


圖 6.5 路徑規劃 多點模擬圖(左)與實際數據(右)

圖 6.5 與圖 6.4 縱軸橫軸表示意義、單位皆相同，手臂初始座標位置也相同，同樣地，左圖為電腦模擬結果，右圖為實際量測數據結果，以上皆使用最短路徑的規畫方式，從結果中可觀察出，加入了初始速度與終止速度，機械手臂在快到達新的座標位置時，會因為有前一點終止速度與下一點初始速度的關係，產生超過再拉回來的現象。

比較電腦模擬結果與實際量測結果，兩者皆相同，故實驗結果相當成功。

另外，測試輸入非工作區內之座標位置，由於在 DSP 板中撰寫 C 語言，沒有虛數可以表示，利用反三角函數  $\cos$  來判斷是否在工作區內的方式，無法在 DSP 板上執行，故實作時，需要採取新的方式來判斷是否在工作區內，由於機械手臂最大範圍為半徑 0.52 公尺的圓形，故代入座標位置後，判斷是否滿足  $P_{za}^2 + P_{ya}^2 < 0.52^2$  即可。

測試結果，當輸入數據在工作區外時，則機械手臂不會有任何動作。因為測試結果，機械手臂即待在初始位置完全不動，數據沒有任何變化，所以沒有利用圖形表示之。

## 6.4 影像處理實驗

在第四章為了擷取到目標物介紹了許多影像處理的方式，但大多數為靜態影像實驗，在這裡進行真正的投擲目標測試，將影像處理中每個過程的畫面顯示出來。使用的解析度為(640×480)。

首先量測目標物在影像平面中的大小與距離關係，詳細數據如表 3，可以觀察出當距離到 400 cm 時，影像平面中半徑大小約 5~6 個 pixel，在影像中已經非常小，故雜訊濾除顯得相當重要。

表 3 距離與 pixel 關係表

距離 (公分)	RGB法面積 (pixel)	HSV法面積 (pixel)	RGB法半徑 (pixel)	HSV法半徑 (pixel)
20	17671	18146	75	76
40	5345	5608	41	42
60	2507	2642	28	29
80	1662	1662	23	23
100	1104	1104	19	19
120	804	855	16	17
140	573	615	14	14
160	433	471	12	12
180	380	398	11	11
200	299	346	10	11
220	269	346	9	11
240	226	227	8	9
260	201	214	8	8
280	165	201	7	8
300	165	165	7	7
320	133	143	7	7
340	133	143	7	7
360	104	133	6	7
380	103	132	6	7
400	87	113	5	6
420	79	95	5	6

一開始程式執行時，先拍下原始背景



圖 6.6 實驗原始背景



圖 6.7 實驗原始影像



圖 6.8 實驗 IRGABM 法

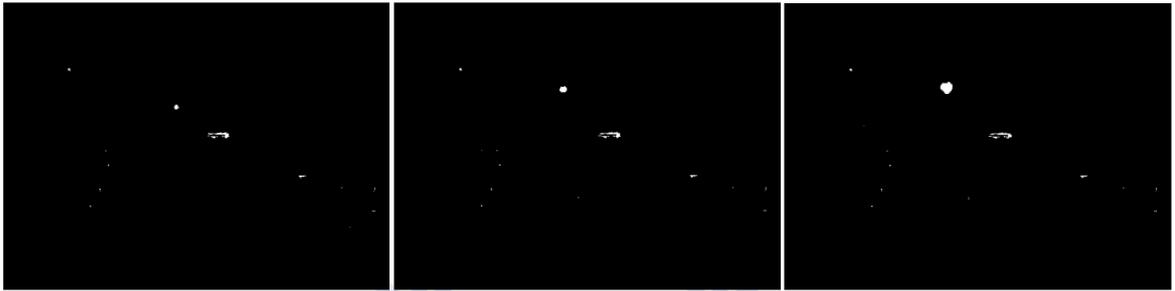


圖 6.9 實驗 HSV 法

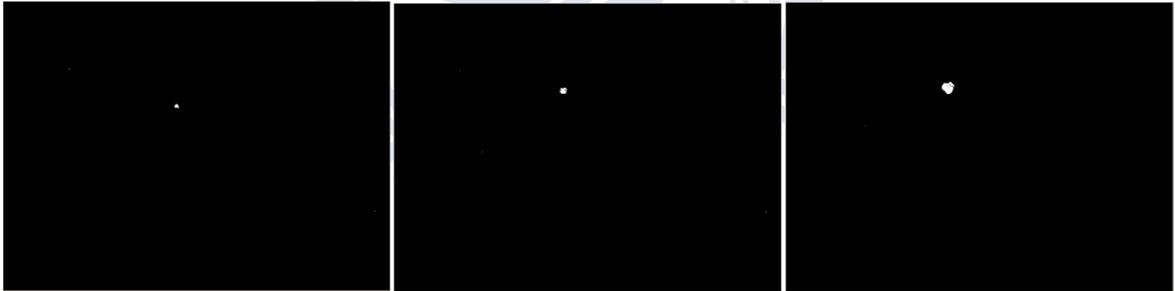


圖 6.10 IRGABM 與 HSV And 後影像

使用 IRGABM 法進行背景相減，如圖 6.8 所示，可看到由於背景太過複雜，使得光線在物體邊緣處產生亮度落差，造成的雜訊影響相當嚴重。設定  $\alpha_1 = 0.3$ ， $\alpha_2 = 0.15$ ，接著使用 HSV 法將紅色部分辨識出來，如圖 6.9 所示，閾值參數分別為  $H > 127, H < 10$ ， $S > 90$ 。將背景相減結果與顏色辨識結果兩者 AND 後，如圖 6.10 所示，可觀察出，已經可以明確的顯示出目標物，除了一些細小雜訊還無法濾除。

為了濾除剩餘細小雜訊，使用中值濾波器濾波後，如圖 6.11 所示，已看不出雜訊的出現。接著為了使影像在中間能有些許斷開點合併，故使用膨脹讓目標物產生些許放大但不影響整體效果，如圖 6.12 所示。最後，將影像處理之後結果，在原始影像上圈選出目標物。如圖 6.13 所示。

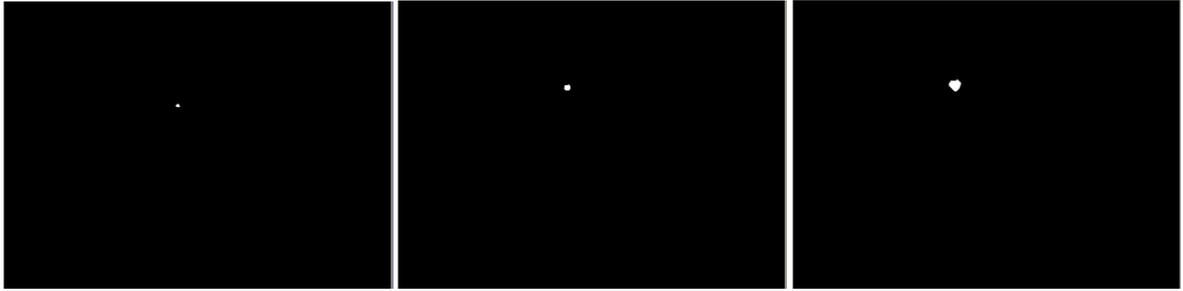


圖 6.11 實驗中值濾波器濾波後影像

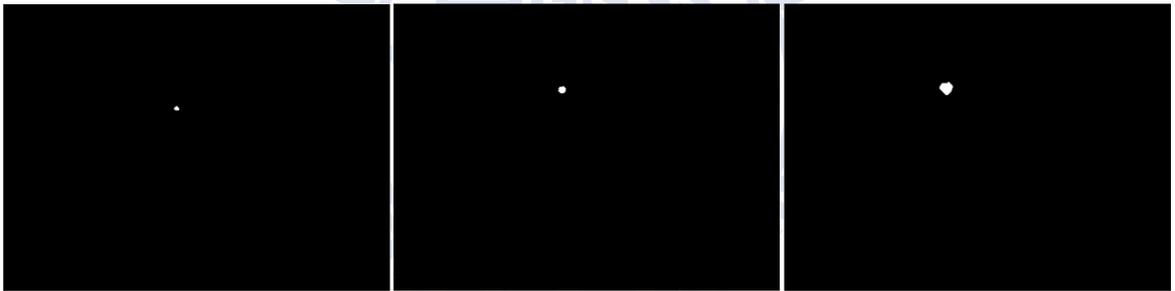


圖 6.12 實驗膨脹後影像

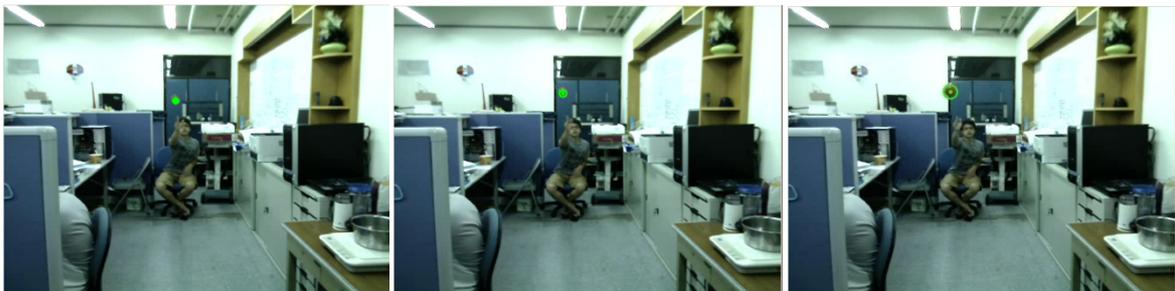


圖 6.13 實驗追蹤目標物影像

上述實驗由左至右分別是選取第 21 張影像，第 23 張影像及第 26 張影像，由於希望能在圖示上顯現影像處理時較大的不同處，故沒有選取連續影像，以下為連續影像追蹤到目標物之影像。



圖 6.14 連續追蹤目標物影像

在影像擷取目標的實驗中，經過多次觀察出，目標物飛行時間大約為 0.4 秒~0.5 秒，平均擷取點數 8~9 點，最多擷取點數可達到 10 點，當目標物越接近攝影機鏡頭時，y 方向移動速度相對過快時，容易造成殘影現象發生，較難以擷取到目標物，但實驗擊打球的時候，由於考慮球飛行時間與機械手臂動作所花費時間，故需要在球飛行軌跡前半部時就將球體飛行軌跡參數求出並傳輸資料給機械手臂準備進行擊打工作，假使太晚計算出來，將造成來不及擊打。故球體飛行後半部殘影造成的部分影像無法擷取較不影響結果。

## 6.5 校正實驗

採用真實座標系，在真實空間中任意選取目標點位置，通常選定實驗環境中較不會移動之固定式物品為目標點，如圖 6.15 所示，量測其 3D 空間中真實座標位置(x,y,z)，至少需 6 點，量測點數越多點，校正參數也越精準。接著觀察其影像平面中座標位置(u,v)，將這些座標位置代入第五章所提到的校正公式，即可求得校正矩陣 M。其中影像平面座標原點位於左上角處，u、v 分別為水平與垂直方向座標。



圖 6.15 校正選取點

表 4 校正結果

點數	校正參數	校正參數	校正參數	估測座標	實際座標	絕對誤差
6	$\alpha$ 577.9	$u_0$ 346.4	$\theta$ 92	u 286.3	284	2.3
	$\beta$ 563.5	$v_0$ 277.4		v 275.1	272	3.1
8	566.1	333.9	88.85	286.9	284	2.9
	524.6	186.4		274.4	272	2.4
17	579	351.1	89.1	337.3	337	0.3
	538.4	276.5		226	224	2
18	574.8	345.8	90.8	303	305	2
	536.5	257.3		279.9	280	0.1
25	563.3	333.9	89	275	274	1
	514.9	249.6		299.5	300	0.5
25	558.5	346.5	88.3	119	119	0
	515.4	230		164.6	163	1.6
27	574.6	317.2	88.9	251.4	251	0.4
	525.1	236.8		208.6	207	1.6
27	574.2	318.5	89.4	359.2	359	0.2
	523.3	250		305.7	305	0.7

如表 3 所示校正結果，其中估測座標是指校正完畢後，另外再選定 1 點做為估測點，利用計算出來之校正矩陣  $M$  將此估測點真實座標位置代入(5.1.7)式，求出其估測影像平面座標。實際座標則為影像平面中直接觀察求得影像平面座標。絕對誤差則是  $|\text{估測座標}-\text{實際座標}|$ 。

比較後可發現，利用此校正方式在絕對誤差值部份，量測點數較多時，絕對誤差值較小，換句話說，當量測點數越多，精確度越高，但也可以發現，並非無上限增加點數，精確度即可達到百分之百，由表 3 可知，當量測點數大於 20 點時，絕對誤差大約控制在 1 個 pixel 左右，以認知上來看，誤差 1 個 pixel 已經屬於相當精準。

## 6.6 機械手臂速度實驗

為了制定機械手臂可擊打目標物之工作區範圍，故需要先了解機械手臂大小軸最快旋轉速度實際數據為多少。

將機械手臂直接持續加入最大電壓 24V，測試其大小軸旋轉角度，實驗數次後，將所得數據平均後畫出圖 6.16，大軸初始角度為 -90 度，旋轉至 70 度停止供應電壓，小軸初始角度則為 0 度，旋轉至 90 度時停止供應電壓。結果如下：

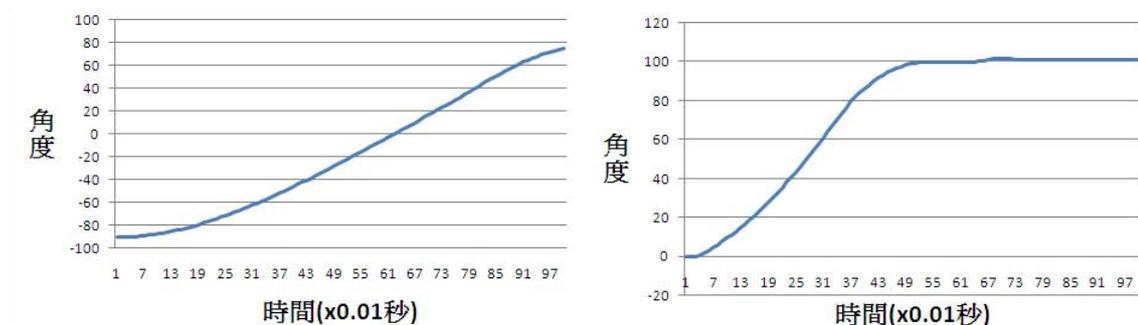


圖 6.16 手臂速度測試 大軸(左)與小軸(右)

從測試結果可發現，機械手臂在大軸部份，旋轉 90 度需花費約 0.6 秒，而小軸部份，旋轉 90 度需花費約 0.44 秒。

由於第五章圖 5.6 中所提到，欲使機械手臂能及時趕上擊球位置，故選擇機械手臂移動 0.3 秒範圍內為其可擊打區域，由測試結果發現，大軸 0.3 秒約可移動±26 度，小軸 0.3 秒約可移動±58 度。依照此數據可用電腦模擬出機械手臂工作區範圍，如圖 6.16 所示

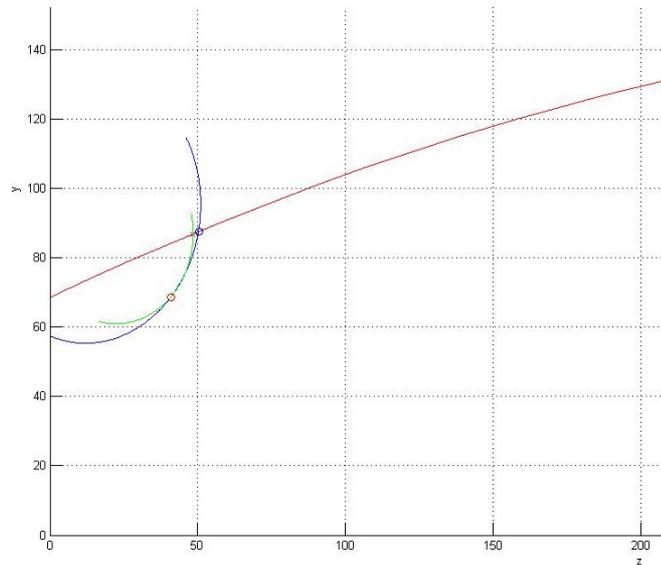


圖 6.17 機械手臂 0.3 秒可移動範圍

其中橫軸為真實空間  $z$  方向縱深單位為公分，縱軸為真實空間  $y$  方向高低，單位為公分，綠色線為僅動作小軸可到達位置，藍色線為同時動作大小軸可移動範圍，紅色線為任意模擬出的拋物線軌跡，機械手臂大小軸初始角度分別為-30 度及-15 度，即圖 6.17 中位於座標約(40,70)之紅點。由圖 6.17 模擬結果看出，此組機械手臂初始角度與機械手臂 0.3 秒可移動範圍所組成的區域，涵蓋球拋物線可能落點大致上的範圍，故實際測試丟球實驗時，採用此組機械手臂初始角度。

## 6.7 實際擊球實驗

實際丟球，即時擷取點數並推算球體飛行軌跡與預測最佳擊球點位置，將其數據記錄下來。

表 5 丟球實驗 1

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	Ya
319	170	0.0333	cx0	106.34	51.06	-5.48
322	158	0.0666	cx1	51.25		
324	149	0.0999	cy0	145.24		
327	141	0.1332	cy1	135.56		
330	137	0.1665	cz0	387.08		
335	132	0.1998	cz1	-668.3		
344	128	0.2331				

表 6 丟球實驗 2

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	Ya
319	168	0.0333	cx0	112	46.77	-20.01
322	152	0.0666	cx1	24.34		
325	145	0.0999	cy0	144.69		
329	138	0.1332	cy1	130.65		
339	131	0.1665	cz0	306.52		
347	134	0.1998	cz1	-487.02		
355	136	0.2331				
380	160	0.2664				

表 7 丟球實驗 3

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	Ya
307	179	0.0333	cx0	113.09	49.75	-12.07
309	169	0.0666	cx1	1.9		
312	163	0.0999	cy0	140.98		
320	154	0.1332	cy1	136.3		
325	151	0.1665	cz0	361.81		
332	150	0.1998	cz1	-611.51		
346	159	0.2331				

表 8 丟球實驗 4

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	Ya
311	183	0.0333	cx0	106.46	45.18	-23.01
315	175	0.0666	cx1	38.16		
317	169	0.0999	cy0	133.45		
322	163	0.1332	cy1	171.05		
327	161	0.1665	cz0	260.02		
339	165	0.1998	cz1	-651.34		
346	171	0.2331				

表 9 丟球實驗 5

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	Ya
311	183	0.0333	cx0	109.41	39.49	-31.23
315	175	0.0666	cx1	27.12		
317	169	0.0999	cy0	139.7		
322	163	0.1332	cy1	135.4		
327	161	0.1665	cz0	398.96		
339	165	0.1998	cz1	-647.27		
346	171	0.2331				

表 10 丟球實驗 6

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	Ya
320	177	0.0333	cx0	106.44	51.35	-2.46
324	173	0.0666	cx1	52.24		
330	171	0.0999	cy0	143.32		
335	171	0.1332	cy1	87.4		
341	175	0.1665	cz0	381.69		
356	191	0.1998	cz1	-780.54		
384	246	0.2331				

表 11 丟球實驗 6

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	ya
313	200	0.0333	cx0	111.62	50.87	-7.32
316	190	0.0666	cx1	28.35		
319	181	0.0999	cy0	132.52		
326	172	0.1332	cy1	163.5		
330	168	0.1665	cz0	336.82		
335	167	0.1998	cz1	-558.16		
350	173	0.2331				

表 5~11 為真實丟球實驗後，任意選取幾個擊球成功的相關數據表示出來，其中  $u, v$  為影像平面擷取到的座標， $t$  為真實時間，單位為秒， $z_a, y_a$  為手臂座標系之  $z_a$  軸與  $y_a$  軸之座標，單位為公分，觀察拋物線參數來判斷參數估算準確度。由於  $cx1, cy1, cz1$  為速度項，故通常選取  $cx0, cy0, cz0$ ，即丟球者初始所在位置來判斷估測結果準確性較為容易。丟球者投擲位置  $x$  座標約為 110~120cm 之間， $y$  座標約為 130~140cm， $z$  座標約為 350~400cm，觀察表 5~11 可察覺出， $x$  與  $y$  座標估測結果精準度相當高，而  $z$  座標則誤差較大。這是由於在 2 維影像平面中，對於 3 維真實空間縱深  $z$  較難辨別，假設球體朝鏡頭筆直飛過來，則在影像平面中，中心點可能皆為同一點座標，如此便很難估測出縱深距離，容易造成誤差過大。

表 12 丟球實驗 7

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	ya
305	99	3.681648	cx0	110.56	47.92	-17.46
306	91	1.356975	cx1	40.75		
310	82	6.034959	cy0	181.23		
313	73	0.77256	cy1	23.2		
316	67	14.77454	cz0	443.68		
321	62	-27.2131	cz1	-817.21		

表 13 丟球實驗 8

擷取點數			拋物線軌跡參數		預測擊球點	
u	v	t			Za	Ya
301	6	0.0333	cx0	116.66	48.04	-17.11
306	61	0.0666	cx1	35.52		
310	61	0.0999	cy0	180.87		
313	65	0.1332	cy1	-62.38		
318	82	0.1665	cz0	304.3		
324	107	0.1998	cz1	-641.92		
323	233	0.2331				

表 12 與表 13 則舉出兩個失敗的例子，觀看拋物線軌跡參數 x 方向及 z 方向數據與成功例子並無太大差別，唯一特別之處在於 y 方向，仔細觀察可發現擷取點數 v 的變化以及 y 方向初始位置 cy0，比平常高約 40~50 cm，這可能是投擲者出手時呈現高拋射方式投擲，導致出手瞬間 y 方向沒有擷取到，另外，在 y 方向速度 cy1 部份可看出，比起成功的例子，y 方向速度慢上許多，這將造成目標物飛行到後半段時，發生急速下降的情況，實際飛行距離可能比預測還要短。

在目標物飛行前半段時間時，飛行軌跡參數還未顯示出目標物呈現高拋物線行進方式，此時影像持續更新擷取點後，目標物飛行到達後半段時，拋物線軌跡參數已顯示不在機械手臂可擊打範圍內，但此時手臂已經朝擊球點移動一半以上，故最後實驗結果仍屬於已擊打，產生擊球失敗的結果。

在實驗過程中，每擷取到新的一點即需要判斷是否有滿足條件式，簡單設定為  $C_{x1} < 70$ ， $C_{y1} < 200$ ， $C_{z0} > 200$ ， $C_{z1} = -1200 \sim -200$ ，即橫向速度須小於 70 cm/sec，高低速度須小於 200 cm/sec，初始投擲位置須大於 200 cm，縱深速度在 -1200~200 cm/sec 之間，條件沒有非常嚴苛。可以依照需求自行調整數值。

通過上述條件式後，將拋物線軌跡代入，求取最佳擊球點，如果球飛行軌跡沒有進入機械手臂工作區內，則無法計算出最佳擊球點。

下圖為擊球真實影像畫面，影像順序由上而下由左至右顯示，紅色框框為機械手臂平台所在位置：

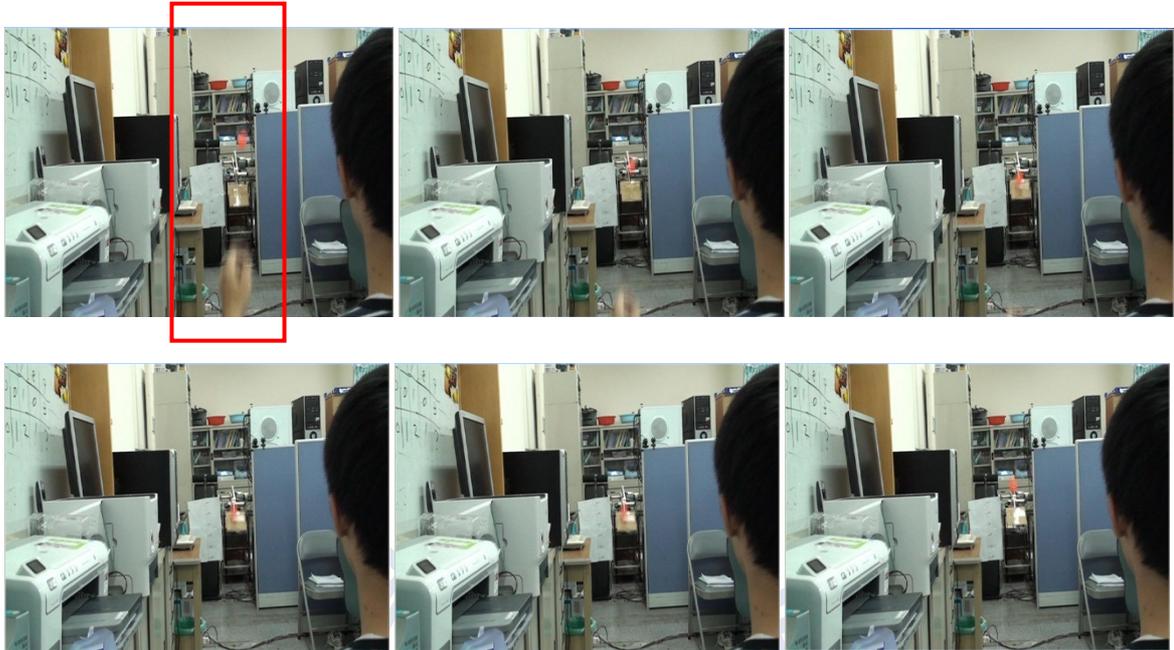


圖 6.18 擊球實驗 1

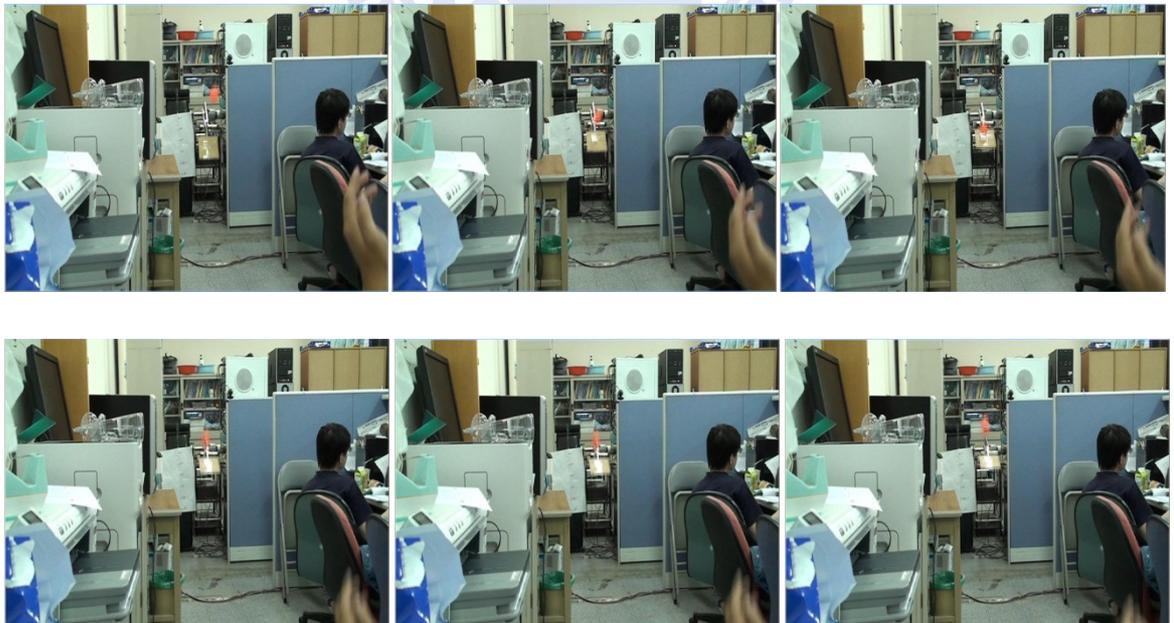


圖 6.19 擊球實驗 2

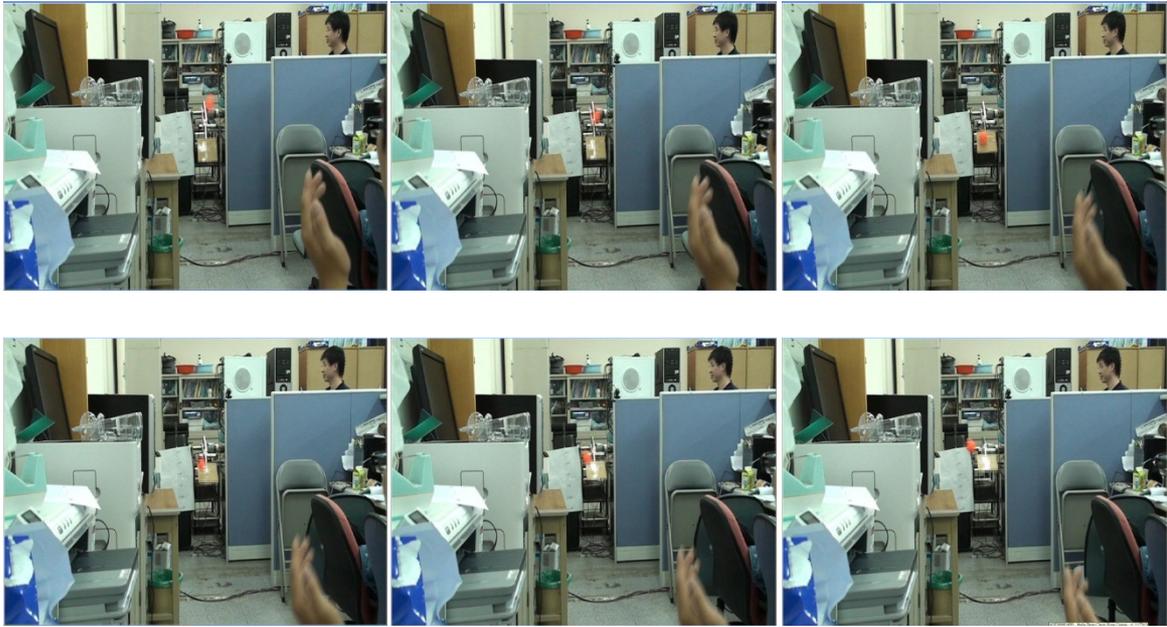


圖 6.20 擊球實驗 3

## 6.8 實驗結果討論

經過多次的丟球實驗，從數據結果可以發現， $z$  方向縱深的估測上較為不準，這是由於使用單眼系統僅有 2D 影像平面，欲利用 2D 影像平面來估測 3D 真實空間距離本有一定難度，可能在真實空間中距離變化很大，但在影像平面中卻可能很小甚至沒有變化。欲改善此問題，可以將攝影機鏡頭擺放位置與目標物飛行路徑呈現一個角度，使攝影機每個時間點拍攝到的縱深方向位移量變大，在估測方面就可以準確，但由於所使用之攝影機造價低廉，頻率僅 30 赫茲，假設攝影機鏡頭擺放位置與目標物飛行路徑夾角過大，則可能明顯出現殘影現象，如此一來在影像處理擷取目標物時，可能就產生相當大的誤差，故擺放位置也是一大考量。

另外，由於機器手臂實驗平台速度與工作區範圍的限制，投擲者在投擲過程中，很容易投偏，造成實驗數據收集較為困難。針對目標物的飛行軌跡，如果是高拋射型的軌跡，除了可能超出攝影機拍攝範圍外，在影像拍攝上容易造成高低  $y$  方向的殘影產生，針對這種高拋射型的飛行軌跡，則影像追蹤目標物效果較差。

整體實驗結果，在機械手臂方面，路徑規畫可以準確的依照規劃路徑動作。影像處理方面，目標物由 4 米處投擲，飛行時間大約在 0.4~0.5 秒之間，影像最多可連續追蹤 10 張影像。除了飛行後半段處落下速度太快產生殘影以至於無法追蹤，追蹤效果大約可達 70%。在最後擊球率方面，投擲 81 次，擊中 35 次，命中率大約 43%，還有改善的空間。



## 第七章 結論與未來展望

### 7.1 結論

使用此單眼視覺系統的機械人，在手臂路徑規畫的部份，控制器能達到相當好的響應，不論是單點或是多點路徑規畫，機械手臂都能依照模擬結果，準確地移動位置。在執行影像處理追蹤高速目標物方面，雖然僅使用市面上隨處可見的視訊鏡頭，但在實際測試上已經達到不錯的效果，且不需要使用相當複雜的演算法或是非常昂貴的硬體設施即可得到此成效。在校正方面，也可使影像平面中誤差值縮小到 1 個 pixel 左右。

利用簡單的拋物線軌跡方程式，並結合校正所推得的真實空間與影像平面轉換公式，即可利用單顆攝影機鏡頭來估測目標物飛行軌跡之相關參數，相較於其他使用昂貴的雙眼系統，本論文使用一顆便宜的硬體即可達到效果，雖然在縱深方向仍有加強的空間，但在橫向與高低的部分，估測參數顯得相當精準，克服許多硬體上限制，利用切換處理畫面來提高影像處理速度，利用求得最佳擊球點角度來彌補機械手臂在速度上的不足，並即時持續更新其運動路徑，盡其所能發揮到最大功效。

因為人工投擲的軌跡每次都有差異，無法百分之百重複實驗，故以多次投擲結果計算其擊中機率來評估系統效能。實驗結果顯示擊球率為 43%，與文獻[10][14]比較可發現，雙方在擊球成功率部份，[10]約為 6 成左右，而[14]約為 4 成左右，文獻在硬體設施方面皆使用雙眼視覺系統來進行估測，本論文卻只用單眼視覺系統即可實現約 4 成的命中率，而且文獻中所使用的機械手臂與攝影機鏡頭，在運作速度上遠遠超出本論文使用的硬體設施，其造價非常昂貴，需要特別訂製，而本論文使用之硬體設施皆造價低廉，甚至是市面上垂手可得的產品。故雙方比較結果，本論文在擊球率不至於差異太大的情況下，卻可以使用更簡單且便宜的方式實現，此為一大貢獻。

## 7.2 未來展望

改良影像處理部份，使追蹤目標物可以不用針對特定顏色；加快機械手臂實驗平台運作速度，使實驗過程中能有更多的彈性空間可以發揮，改善縱深方向估測準確度，進而提高擊球成功率。

接著可使用力回饋控制來利用機械手臂針對某特定方向或位置進行擊球的動作，使機械手臂能像棒球選手一樣做出推打、拉打、點球等動作。



## 參考文獻

- [1] J. Campbell, R. Sukthankar, I. Nourbakhsh, and A. Pahwa, "A Robust Visual Odometry and Precipice Detection System Using Consumer-Grade Monocular Vision," *2005 IEEE International Conference on Robotics and Automation*, pp. 3421-3427, 2005.
- [2] F. Devernay and O. Faugeras, "Straight Lines Have to Be Straight - Automatic Calibration And Removal of Distortion from Scenes of Structured Environments," *Machine Vision and Applications*, vol. 13, pp. 14-24, Aug 2001.
- [3] D. A. Forsyth and J. Ponce, "Computer Vision: A Modern Approach ", ed: Prentice Hall, 2002.
- [4] R. B. Gornes, L. M. G. Gonalves, and B. M. de Carvalho, "Real Time Vision for Robotics Using A Moving Fovea Approach with Multi Resolution," in *2008 IEEE International Conference on Robotics and Automation, Vols 1-9*, New York, 2008, pp. 2404-2409.
- [5] E. Hodneland, "Segmentation of Digital Images," Cand. scient thesis, Department of Mathematics, University of Bergen, Citeseer, 2003.
- [6] H. Iddamsetty, "Segmentation of Range Images for Modeling of Large Outdoor Scenes," Project in Lieu of Thesis For Masters Degree The University of Tennessee, Knoxville, Citeseer, 2005.
- [7] S. Kasiri-Bidhendi and R. Safabakhsh, "Effective Tracking of the Players and Ball in Indoor Soccer Games in the Presence of Occlusion," presented at the 14th International CSI Computer, 2009.
- [8] R. Kelly, V. Santibanez, and A. Loria, "Control of Robot Manipulators in Joint Space". Advanced Textbooks in Control and Signal Processing: Springer-Verlag, London, 2005.
- [9] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russell, "Towards Robust Automatic Traffic Scene Analysis in Real-Time," presented at the Proceedings of the 12th IAPR International Conference on Pattern Recognition - Conference A: Computer Vision & Image Processing, Los Alamitos, 1994.
- [10] C. Y. Lin and Y. P. Chiu, "The DSP Based Catcher Robot System with Stereo Vision," in *2008 IEEE/Asme International Conference on Advanced Intelligent Mechatronics, Vols 1-3*, New York, 2008, pp. 897-903.
- [11] D. Murray and C. Jennings, "Stereo Vision Based Mapping and Navigation for Mobile Robots," University of British Columbia, Vancouver, 1997.
- [12] S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, R. Schmidt, and T. Graf, "High Accuracy Stereo Vision System for Far Distance Obstacle Detection," in *intelligent Vehicles Symposium, 2004 IEEE*, 2004, pp. 292-297.
- [13] F. A. Peres, F. R. Oliveira, L. A. Neves, and M. F. Godoy, "Automatic Segmentation of

- Digital Images Applied in Cardiac Medical Images," in *Health Care Exchange (PAHCE), 2010 Pan American*, 2010, pp. 38-42.
- [14] T. Senoo, A. Namiki, and M. Ishihawa, "Ball Control in High-Speed Batting Motion Using Hybrid Trajectory Generator," presented at the 2006 IEEE International Conference on Robotics and Automation, New York, 2006.
- [15] N. Shibuya, Y. Shimohata, T. Harada, and Y. Kuniyoshi, "Smart Extraction of Desired Object from Color-Distance Image with User's Tiny Scribble," presented at the 2008 IEEE/Rsj International Conference on Robots and Intelligent Systems, Vols 1-3, Conference Proceedings, New York, 2008.
- [16] Y. Shu-Te Su, "Moving Object Segmentation Using Improved Running Gaussian Average Background Model," in *Digital Image Computing: Techniques and Applications*, 2008, pp. 24-31.
- [17] X. Tong, H. Lu, and Q. Liu, "An Effective and Fast Soccer Ball Detection and Tracking Method," in *17th International Conference on Pattern Recognition*, 2004, p. 798.
- [18] A. Treptow and A. Zell, "Real-Time Object Tracking for Soccer-Robots without Color Information," *Robotics and Autonomous Systems*, vol. 48, pp. 41-48, 2004.
- [19] R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-The-Shelf TV Cameras And Lenses," *IEEE Journal of Robotics and Automation*, vol. 3, pp. 323-344, Aug 1987.
- [20] M.-C. Weng, "Control and Actuator Fault Diagnosis of Robot Manipulators," National Chiao Tung University, 2009.
- [21] P. Wong, "Developing an Intelligent Table Tennis Umpiring System: Identifying the Ball from the Scene," in *Second Asia International Conference on Modelling & Simulation*, 2008, pp. 445-450.
- [22] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-Time Tracking of the Human Body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 780-785, Jul 1997.
- [23] Z. Xin-Zhi, W. Inamura, K. Shibata, and K. Ito, "Task Skill Formation via Motion Repetition in Robotic Dynamic Manipulation," in *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on*, 1999, pp. 1001-1006 vol.4.
- [24] D. Yow, B. Yeo, M. Yeung, and B. Liu, "Analysis and Presentation of Soccer Highlights from Digital Video," in *second asian conference on computer vision 1995*, p. 218.
- [25] G. W. Zack, W. E. Rogers, and S. A. Latt, "Automatic-Measurement of Sister Chromatid Exchange Frequency," *Journal of Histochemistry & Cytochemistry*, vol. 25, pp. 741-753, 1977.
- [26] Q. Zhan, S. R. Huang, and J. Wu, "Automatic Navigation for A Mobile Robot with Monocular Vision," in *2008 IEEE Conference on Robotics, Automation, and*

*Mechatronics, Vols 1 and 2, New York, 2008, pp. 531-536.*

