

# 一. 緒論

## 1.1 研究動機與目標

近幾年來人體動作捕捉 (motion capture) 技術已被廣泛的使用 [1-6]。遊戲業者與或電影業者為了使動畫的效果更加逼真細膩，紛紛使用人體動作捕捉的技術來製作動畫，而在醫學復健動作分析、運動員動作訓練上也都應用到人體動作捕捉技術。因此，人體動作捕捉在現代科技研發上扮演了非常重要的角色。

何謂人體動作捕捉 (motion capture) ? 概括的說，這個系統可以追蹤一些關鍵點，然後再根據關鍵點將人體的動作轉換成數位資料，供進一步的應用分析。除了捕捉人體全身的動作外，局部的動作諸如臉部、手部等細部動作，以至於其他動物的動作皆可以捕捉並加以分析。

然而，現今的人體動作捕捉技術都需要使用感應器、感應標誌，造成表演者表演時一些不必要的麻煩。本篇論文所提出的是不用感應標誌的人體動作捕捉技術，可將表演者的動作經由 3D 模型重新表現出來，讓表演者可以自在的展開動作而不受拘束，使表演能更加流暢、動畫更加逼真。

## 1.2 相關研究

目前人體捕捉系統從 20 世紀 70 年代起至今蓬勃發展，以其感測原理來區分的話，大致可以分為以下三類：

### 1. 機械式[9, 10]：

表演人員穿戴人形機械套件，當表演人員做動作時，這些人形機械套件在跟著人體移動的同時，身上的感應器也紀錄著每個關節的旋轉量。這種方法的準確度最高，但表演人員的活動易受機械套件限制，造成表演者的活動範圍縮小，且機械套件設備笨重，對表演人員亦是一大負擔，此外，機械套件也要經常校正，造成不必要的麻煩。

## 2. 電磁波式[9, 11]：

使用一個磁性發射器以產生磁場，表演人員穿戴磁場感應器，利用磁場效應，以感應力的大小來判斷物體在 3D 中的位置。所測量的位置和方向不同，經過後置處理即可直接的接受，成本也較低。缺點在於容易受空間中其他電磁波、雜訊以及金屬物質的影響，取樣頻率較低，造成分析上的誤差。

## 3. 光學式[9, 12, 13, 23]：

表演人員穿著貼有感應標誌的緊身衣，利用多台攝影機 (camera) 拍攝表演人員，經由相機校正及後置處理，求得表演人員穿戴之感應標誌的 3D 位置。運用此法不會有纜線的束縛，可活動空間較大取樣頻率亦較高，但需要複雜的後置處理，且感應標誌容易受到遮掩而使損失實驗數據。

# 1.3 研究方法介紹

本篇論文研究的方法如下：在使用二台攝影機，架設在適當的位置並做好相機校正的動作，將所拍攝的影像把人體的影像捕捉出來並求得 End\_effector 的和肢節向量的資訊後，再將這些資訊餵進本論文研究改善的逆向動力學解法，求得各個肢節點位置，。

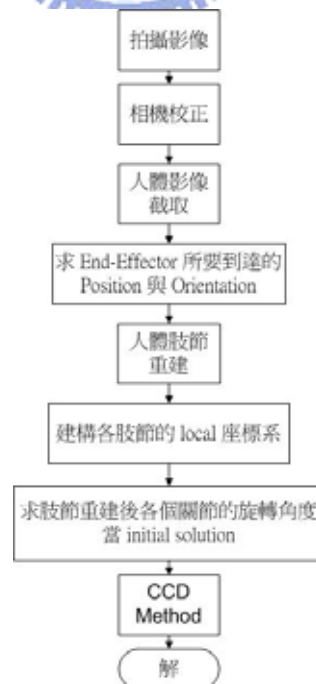


圖 1.1 研究流程

## 1.4 論文組織

本論文除本章為“緒論”外，第二章中將討論本論文需要用到的電腦視覺定位理論，包括相機校正的方法、平面投影幾何關係的理論，利用校正後的攝影機參數，將所得的二維影像資料轉換成三維的資訊。

在第三章中，介紹一些解決逆向動力學的方法，並將其中的 Cyclic Coordinate Decent (CCD) Method 加以改善，並比較改善前後的不同，再藉由模擬的結果來印證本論文的研究成果。

第四章介紹簡單快速的場影變化偵測，將人體從影像背景中捕捉出來。

第五章為研究過程中所進行的各項實驗，包括了模擬實驗結果來驗證本論文所提出的改善方法、用影像處理將人體從背景中擷取出來結果及將 CCD 方法應用在實際拍攝的影像所得到的結果。第六章是結論。



## 二. 理論基礎

在整個物體重建過程的三度空間座標系統中，使用二台以上的攝影機來拍攝三度空間中運動的人體，藉以計算出人體在 3D 空間中的位置 (position) 與方向 (orientation)，並推導三度空間物體和二維影像之間的投影關係，做為計算人體 3D 資訊的依據。

首先先定義一個共同的座標系統稱為「世界座標系統 (World Coordinate System, WCS)」，參考座標軸為  $(x_w, y_w, z_w)^T$ 。影像座標系統 (Image Coordinate System, ICS)，參考座標為軸  $(u_i, v_i)^T$ 。攝影機座標系統 (Camera Coordinate System, CCS)，參考座標軸為  $(x_c, y_c, z_c)^T$ 。三個座標系統的示意圖如下：

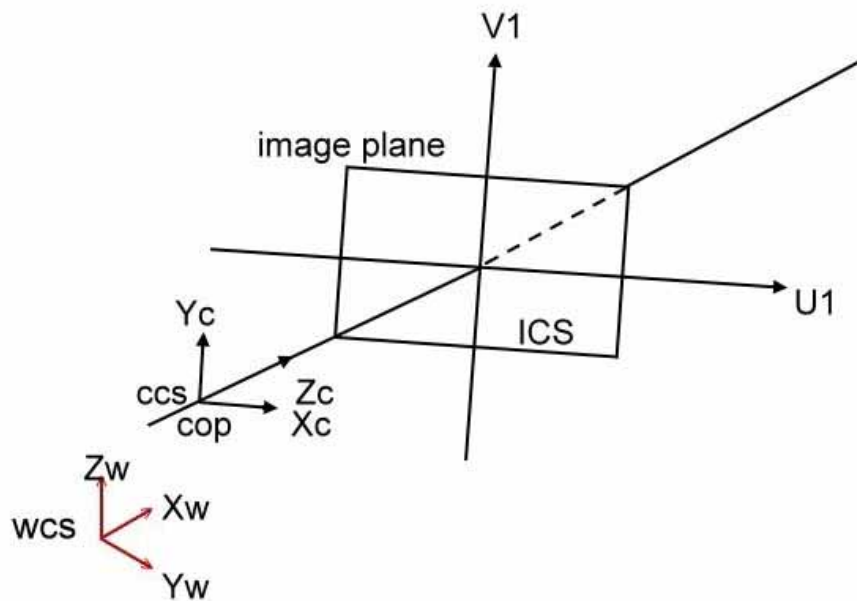


圖 2.1 WCS、CCS 及 ICS 間的關係

因經由攝影機拍攝所獲得的影像資料無法直接轉換，須將影像座標系統 (ICS) 轉換到攝影機座標系統 (CCS)，再轉換到世界座標系統 (WCS)。經過這個步驟，所有的座標值與向量才能互相進行各種計算。

傳統作法上，在空間中自訂定出世界座標系統的原點之後，對攝影機所處的世界座標系統中的位置，以及其內部的各項參數進行校正後，才能將攝影機所拍到的二維影像座標值轉換到三維空間的世界座標值，使所有的座標系統都能互相轉換。

## 2.1 投影原理

在電腦圖學的投影成像原理中，依不同的相機模型，主要分為平行投影 (Parallel Projection) 和透視投影 (Perspective projection)。

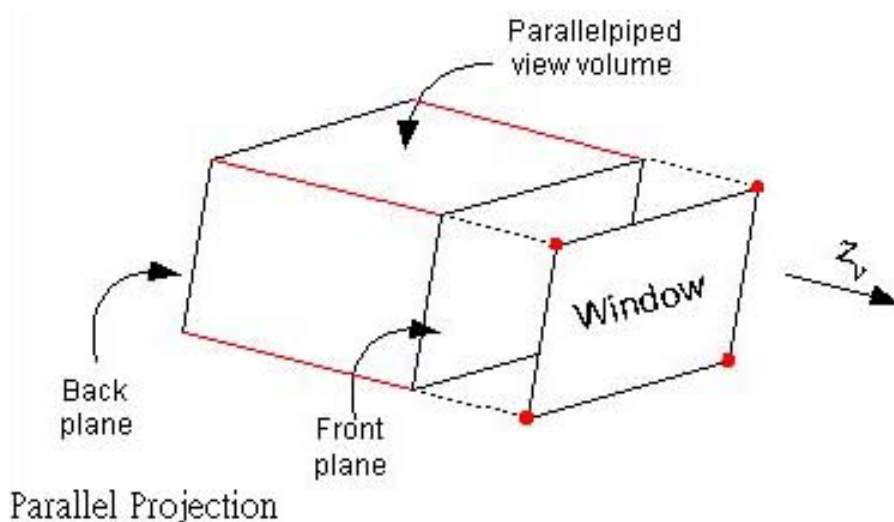


圖 2.2 平行投影示意圖

然而，平行投影方式並不符合真實世界中眼睛所看到的情況。在本論文所提出的系統中將採用針孔成像相機模型 (Pinhole Camera Model)，在透視投影的成像方式中，影像中所有點都會投影到 center of projection (COP)，其透視投影的成像方式示意圖如下：

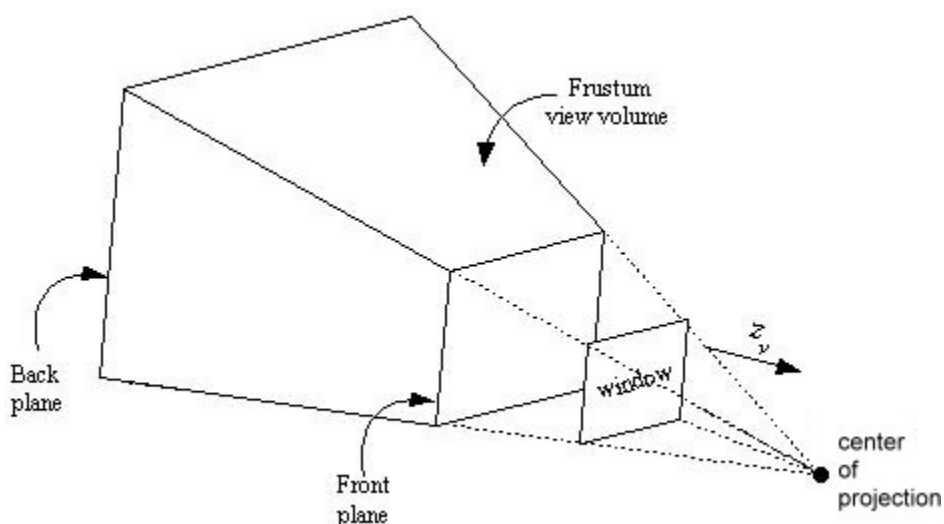


圖 2.3 透視投影示意圖

透視投影的成像方式，更加符合實際拍攝所得的影像，有助於提高整個系統的準確性。



## 2.2 相機參數校正

利用 homogeneous coordinate system，可以求得在世界座標系 (WCS) 中的 3D 座標點  $[X_w \ Y_w \ Z_w]^T$  和相對應的影像座標系統 (ICS) 上的 2D 座標點  $[u_i \ v_i]^T$  的關係式，兩者之間的關係可用數學式表示為：

$$\begin{bmatrix} u_i \cdot w \\ v_i \cdot w \\ w \end{bmatrix} = H \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \end{bmatrix} \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

其中  $w$  為一應變數， $H$  則是所要求的相機投影矩陣 (Projection matrix) 或稱為相機矩陣 (camera matrix)。矩陣  $H$  實際上包含了相機的六個內部參數， $f$  : effect focal length,  $\delta_u, \delta_v$  : aspect ration,  $p_u, p_v$  : principle point,  $\theta$  : skew angle 以及六個外部參數， $\theta_\omega, \theta_\phi, \theta_\kappa$  : rotation angle,  $[t_x \ t_y \ t_z]^T$  : translation。

在矩陣  $H$  中共有 12 個未知參數，以傳統的方法 [24, 26-29]，只要用 6 對以上不共面的 3D 點及其對應的 2D 投影點，再利用最小平方方法 (Least square method) 計算出矩陣  $H$  中的每一個參數，如此即可求得最接近的投影矩陣  $H$ 。

本論文所使用的方法是在攝影機可以拍攝到的範圍內，定出一個世界座標系統 (WCS) 的原點  $[0_w \ 0_w \ 0_w]^T$  當做基準點，接著在不同的平面上貼上特徵點，並且測量其在世界座標系統中的三維空間座標值，再經過攝影機拍攝的影像中，就可以得到多對的三維空間座標和相對應的二維影像座標。實際上，我們所使用的對應點越多，利用最小平方方法 (least square method) 所得到的矩陣  $H$  誤差就越小。



圖 2.4 相機校正用影像(a)



相機校正用影像(b)

## 2.3 三度空間中的線段方向

當知道了兩張影像上的射線 $\vec{v}_1$ 、 $\vec{v}_2$ ，可經由已知的H矩陣，來找出三度空間中的線段方向[24]，示意圖如下：

在下圖中，在射線 $\vec{v}_1$ 、 $\vec{v}_2$ 中各取二點 $a_1$ 、 $b_1$ 和 $a_2$ 、 $b_2$ ，將線段 $\overline{a_1b_1}$ 及線段 $\overline{a_2b_2}$ 投影到三度空間中，會形成平面 $A_1B_1C_1$ 及平面 $A_2B_2C_2$ ，而這兩個平面的交線即是所要求三度空間中的線段方向。其中 $C_1$ 、 $C_2$ 各為兩台攝影機COP在WCS的位置，因此，只要能求得平面 $A_1B_1C_1$ 及平面 $A_2B_2C_2$ 法向量 $\vec{N}_1$ 、 $\vec{N}_2$ ，即可做外積 (cross product) 找出兩平面的交線方向。

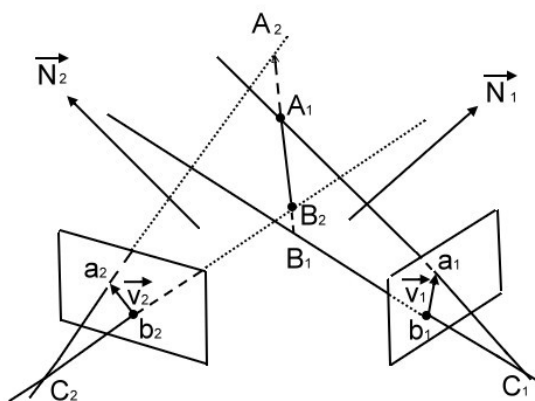


圖 2.5 求解三度空間線段方向示意圖

以求平面 $A_1B_1C_1$ 的向量 $\vec{N}_1$ 為例，從影像上的點 $a_1 = [u_{a_1} \ v_{a_1}]^T$ 與已知的H矩陣可得出關係式如下：

$$\begin{bmatrix} u_{a_1} w \\ v_{a_1} w \\ w \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \end{bmatrix} \cdot \begin{bmatrix} X_{A_1} \\ Y_{A_1} \\ Z_{A_1} \\ 1 \end{bmatrix}$$

，相乘展開之後可得：

$$u_{a_1} w = h_{11} X_{A_1} + h_{12} Y_{A_1} + h_{13} Z_{A_1} + h_{14} \quad (2.1)$$

$$v_{a_1} w = h_{21} X_{A_1} + h_{22} Y_{A_1} + h_{23} Z_{A_1} + h_{24} \quad (2.2)$$

$$w = h_{31} X_{A_1} + h_{32} Y_{A_1} + h_{33} Z_{A_1} + h_{34} \quad (2.3)$$

再將Eq.(2.3)式代回Eq.(2.1)式和Eq.(2.2)式，可將w變數消去，

$$(h_{11} - h_{31} u_{a_1}) X_{A_1} + (h_{12} - h_{32} u_{a_1}) Y_{A_1} + (h_{13} - h_{33} u_{a_1}) Z_{A_1} + (h_{14} - h_{34} u_{a_1}) = 0 \quad (2.4)$$

$$(h_{21} - h_{31} v_{a_1}) X_{A_1} + (h_{22} - h_{32} v_{a_1}) Y_{A_1} + (h_{23} - h_{33} v_{a_1}) Z_{A_1} + (h_{24} - h_{34} v_{a_1}) = 0 \quad (2.5)$$

Eq.(2.4)、Eq.(2.5)所代表的意義為空間中，通過 $a_1 = [u_{a_1} \ v_{a_1}]^T$ 的兩平面，將此二平面做外積 (cross product) 之後就可以得出兩平面的交線，其交線方向即為三度空間中，通過 $a_1$ 的射線方向 $\overrightarrow{a_1A_1}$ ；同理可求得通過 $b_1$ 、 $a_2$ 、 $b_2$ 的射線 $\overrightarrow{b_1B_1}$ 、 $\overrightarrow{a_2A_2}$ 、 $\overrightarrow{b_2B_2}$ ，有了直線向量，就可以很容易的求出通過兩直線的平面 $A_1B_1C_1$ 法向量 $\vec{N}_1 = \overrightarrow{a_1A_1} \times \overrightarrow{b_1B_1}$ ，平面 $A_2B_2C_2$ 法向量 $\vec{N}_2 = \overrightarrow{a_2A_2} \times \overrightarrow{b_2B_2}$ ，而所要求的三度空間中的線段方向為平面 $A_1B_1C_1$ 及平面 $A_2B_2C_2$ 的交線方向，即 $\vec{N}_1 \times \vec{N}_2$

## 2.4 三度空間中的點位置

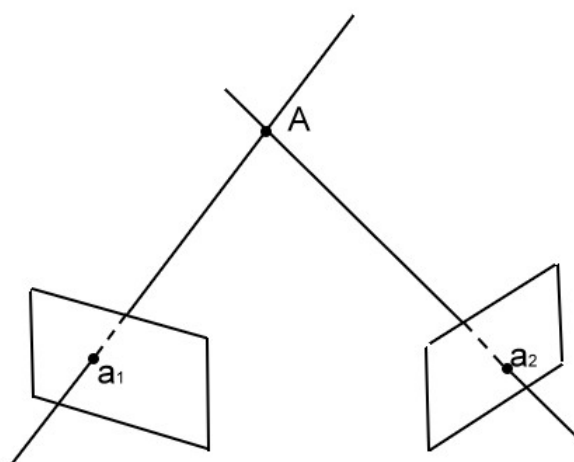


圖 2.6 求解三度空間點位置示意圖

上圖中，在三度空間中的點 A，經過透視投影 (Perspective projection) 後分別投影在二台相機影像平面上的 $a_1$ ，和 $a_2$ ，所以若能求得兩張影像上對應到同一個三度空間中的點，再經由已知的 H 矩陣，即可反推求得三度空間中的點位置[24]，所測量的投影座標及所求得的矩陣 H 中均有些許的誤差，所以不可能準確的交在一點，因此，也是利用最小平方法 (least square method) 來求得交點 A 在 WCS 中的三維座標值。

以上圖為例，已知兩張影像上的點 $a_1 = [u_{a_1} \ v_{a_1}]^T$ ， $a_2 = [u_{a_2} \ v_{a_2}]^T$ 為三度空間中的點 $A = [X_A \ Y_A \ Z_A]^T$ 在兩張影像上的投影點，且令兩張影像經由相機校正後求得的 H 矩陣各為



$$H' = \begin{bmatrix} h'_{11} & h'_{12} & h'_{13} & h'_{14} \\ h'_{21} & h'_{22} & h'_{23} & h'_{24} \\ h'_{31} & h'_{32} & h'_{33} & h'_{34} \end{bmatrix} \quad H'' = \begin{bmatrix} h''_{11} & h''_{12} & h''_{13} & h''_{14} \\ h''_{21} & h''_{22} & h''_{23} & h''_{24} \\ h''_{31} & h''_{32} & h''_{33} & h''_{34} \end{bmatrix}$$

有了以上的資訊，可以列出下列關係式

$$\begin{bmatrix} u_{a_1} w_1 \\ v_{a_1} w_1 \\ w_1 \end{bmatrix} = \begin{bmatrix} h'_{11} & h'_{12} & h'_{13} & h'_{14} \\ h'_{21} & h'_{22} & h'_{23} & h'_{24} \\ h'_{31} & h'_{32} & h'_{33} & h'_{34} \end{bmatrix} \cdot \begin{bmatrix} X_A \\ Y_A \\ Z_A \\ 1 \end{bmatrix}, \quad \begin{bmatrix} u_{a_2} w_2 \\ v_{a_2} w_2 \\ w_2 \end{bmatrix} = \begin{bmatrix} h''_{11} & h''_{12} & h''_{13} & h''_{14} \\ h''_{21} & h''_{22} & h''_{23} & h''_{24} \\ h''_{31} & h''_{32} & h''_{33} & h''_{34} \end{bmatrix} \cdot \begin{bmatrix} X_A \\ Y_A \\ Z_A \\ 1 \end{bmatrix}$$

如同上一節的方法，將上兩式展開後，再用代入消去法可將  $w_1$ 、 $w_2$  消掉，可得

$$\begin{aligned} (h'_{11} - h'_{31} u_{a_1}) X_A + (h'_{12} - h'_{32} u_{a_1}) Y_A + (h'_{13} - h'_{33} u_{a_1}) Z_A + (h'_{14} - h'_{34} u_{a_1}) &= 0 \\ (h'_{21} - h'_{31} v_{a_1}) X_A + (h'_{22} - h'_{32} v_{a_1}) Y_A + (h'_{23} - h'_{33} v_{a_1}) Z_A + (h'_{24} - h'_{34} v_{a_1}) &= 0 \\ (h''_{11} - h''_{31} u_{a_2}) X_A + (h''_{12} - h''_{32} u_{a_2}) Y_A + (h''_{13} - h''_{33} u_{a_2}) Z_A + (h''_{14} - h''_{34} u_{a_2}) &= 0 \\ (h''_{21} - h''_{31} v_{a_2}) X_A + (h''_{22} - h''_{32} v_{a_2}) Y_A + (h''_{23} - h''_{33} v_{a_2}) Z_A + (h''_{24} - h''_{34} v_{a_2}) &= 0 \end{aligned}$$

將上列四式寫成矩陣方式為

$$\begin{bmatrix} h'_{11} - h'_{31} u_{a_1} & h'_{12} - h'_{32} u_{a_1} & h'_{13} - h'_{33} u_{a_1} \\ h'_{21} - h'_{31} v_{a_1} & h'_{22} - h'_{32} v_{a_1} & h'_{23} - h'_{33} v_{a_1} \\ h''_{11} - h''_{31} u_{a_2} & h''_{12} - h''_{32} u_{a_2} & h''_{13} - h''_{33} u_{a_2} \\ h''_{21} - h''_{31} v_{a_2} & h''_{22} - h''_{32} v_{a_2} & h''_{23} - h''_{33} v_{a_2} \end{bmatrix} \cdot \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} = \begin{bmatrix} h'_{34} u_{a_1} - h'_{14} \\ h'_{34} v_{a_1} - h'_{14} \\ h''_{34} u_{a_2} - h''_{14} \\ h''_{34} v_{a_2} - h''_{14} \end{bmatrix}$$

為了方便，將上列等式表示為

$M_{4 \times 3} \cdot V_{3 \times 1} = b_{4 \times 1}$ ，利用最小平方方法 (least square method)，

$$M^T M V = M^T b$$

$$V = (M^T M)^{-1} M^T b = [X_A \quad Y_A \quad Z_A]^T$$

## 三. 逆向動力學 (Inverse Kinematics)

### 3.1 人體骨架模型

理想中的人體模型[15, 16, 25]包含了骨骼、肌肉、皮膚、衣服、毛髮還有臉部的表情等等，但在動作捕捉 (Motion Capture) 的電腦運算中，重點在於要能夠測得關節點的位置與方向。人體是由骨骼所組成的骨架堆成，而骨骼又受肌肉牽引而轉動，此為人體運動的機制。

然而，人體的骨架非常複雜，為了使 Motion Capture 的資料簡化，本文制定了既方便計算又不失人體基本結構的骨架架構，如下圖：

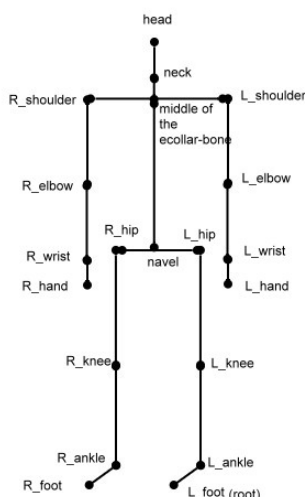


圖 3.1 制定人體骨架規格圖

### 3.2 肢節點位置的重定

由於影像處理所求得的肢節點位置不可能絕對正確，計算出來的肢節長度也會有所誤差。然而對於同一個人，肢節的長度是固定不變的，所以必須重定肢節點的位置，使得肢節的長度能符合人體模型。

肢節點的初始位置決定是利用影像處理所求得的肢節向量，再套上已知的肢節長度得出新的肢節點位置。此做法可使肢節的長度符合已知的人體模型，並沿用所估的肢節角度。

首先，經由影像處理求得 root (L\_foot) 在二張影像座標系統上 ( $ICS_1$  和  $ICS_2$ ) 的座標位置，接著利用第二章所提到 2D 到 3D 點座標的轉換方法，求出左腳尖 (L\_foot) 在三維空間中的座標位置，再以影像座標系統上的左腳尖當端點，計算出從左腳尖到左腳踝 (L\_ankle) 在  $ICS_1$  上的向量，利用同樣的方法同時求出左腳尖到左腳踝在  $ICS_2$  上的向量，再經由第二章所提到的 2D 到 3D 線段向量的轉換方法，可得到左腳尖到左腳踝在三維空間的向量。有了左腳尖的三維空間座標和左腳尖到左腳踝在三維空間的向量，只要再套上已知的肢節長度，便可得到左腳踝在三維空間中的座標位置了。接下來再:left腳踝當端點，求出左膝 (L\_knee) 和左髖骨 (L\_hip) 在三維空間中的位置。再使用同樣的方法，用已知的 End\_effector (L\_hand、R\_hand、R\_foot、head) 為起點，可陸續的求出手腕 (wrist)、手肘 (elbow)、肩膀 (shoulder)、脖子 (neck) 在三維空間的座標位置，示意圖如下。

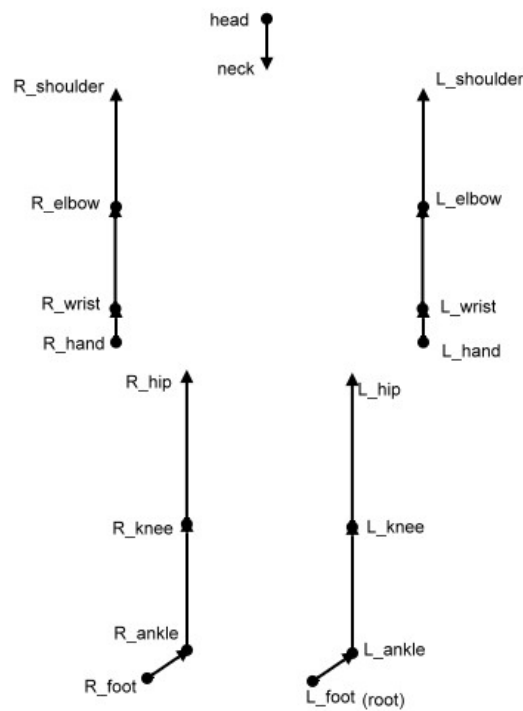


圖 3.2 重定肢節點位置重定示意圖(a)

然而，這個結果只能讓每個區塊的肢節長度固定不變，必需再將每個區塊整合起來，形成正確的人體模型。在此先整合雙腳和雙手，有了  $L\_hip$ 、 $R\_hip$ 、 $L\_shoulder$ 、 $R\_shoulder$  在三維空間中座標，就可以得知它們之間的向量  $\overrightarrow{hip_{LR}}$ 、 $\overrightarrow{shoulder_{LR}}$ ，再套上已知的肢節長度，就可以將雙腳、雙手整合起來，如下圖。

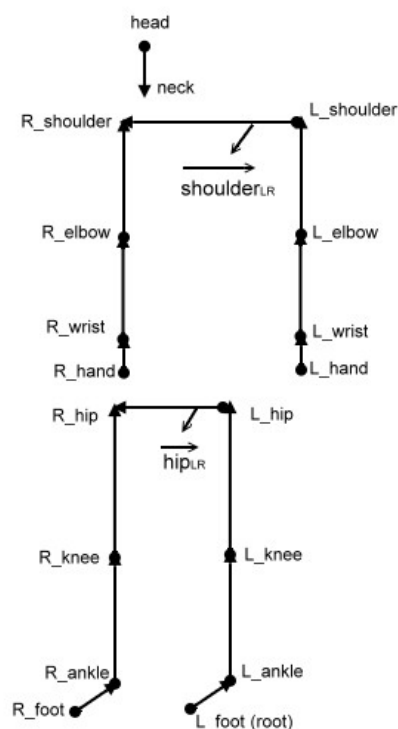


圖 3.2(續) 重定肢節點位置重定示意圖(b)

緊接著必須求得身體軀幹向量，才能將雙手和雙腳整合起來，而身體軀幹是由脊椎為骨架，在簡化的人體構造中，脊椎會通過肚臍，而脊椎和鎖骨是垂直的，即是身體軀幹向量會和兩肩所構成的向量垂直。首先求得肚臍的座標，在虛擬的人體骨架模型中，肚臍 (navel) 的座標為 L\_hip 和 R\_hip 的中心點位置  $P_1$ ，接下來求通過 L\_shoulder ( $P_{LS}$ )、R\_shoulder ( $P_{RS}$ ) 的直線方程式  $L_1$ ，利用  $L_1$ 、求得直線方程式  $\overline{P_1P_2}$ ，使得

$$\overline{P_1P_2} \perp L_1, P_2 \in L_1$$

身體軀幹向量和兩肩之間的向量垂直。因為人體是對稱的，所以必需延著  $L_2$  移動包含 L\_shoulder 的整個區塊和包含 R\_shoulder 的整個區塊，使得

$$\| \overline{P'_{LS} P_2} \| = \| \overline{P_2 P'_{RS}} \|; P'_{LS}, P'_{RS} \in L_1$$

如此一來便整合了雙手、雙腳和身體軀幹。最後再以  $P_2$  為端點，求得脖子到頭的向量，套上已知的肢節長度，那麼，整個人體的肢節就重定完成，人體架構的模型建構就完成了。如下圖：

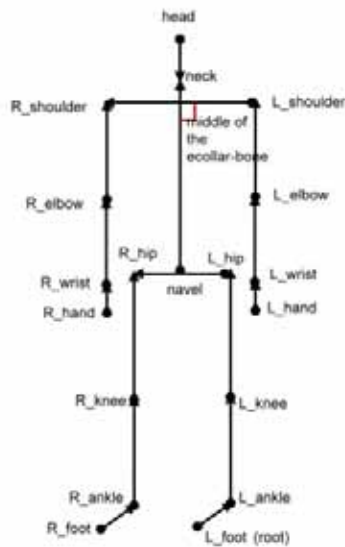


圖 3.2(續) 重定肢節點位置重定示意圖(c)

重定肢節點完成之後，雖然確保了肢節長度能固定不變，但也影響了末端肢節端點的位置產生了位移，因此必須利用重定好的人體模型將末端肢節端點的位置移到到所在的目標位置，以符合實際拍攝時所看到的末端肢節端點位置。

### 3.3 建構正向動力學(Forward Kinematics)的人體肢節點座標系及肢節角度

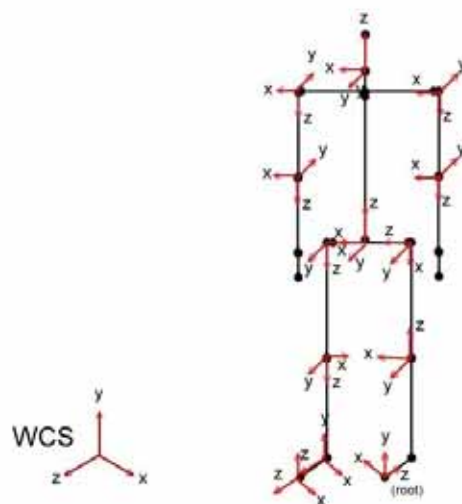


圖 3.3 定義人體肢節點區域座標系



為了求得每個肢節對於轉軸的旋轉角度，首先必需定義出人體每個肢節的區域座標系，有了區域的座標系後肢節才能經由定義出來的區域座標當參考，方便的求出對旋轉軸的角度。為了不混淆且方便計算，本論文將所有的肢節都定義成座標軸 Z 軸上，如上圖所定義。

在人體構造中，兩肢節相連接的部份稱之為關節，所以關節的旋轉角度取決於相鄰的兩個肢節方向，換句話說，求解關節角度即是求兩肢節的夾角。而本論文所有區域座標系中的 Z 軸都定義在肢節方向上。因此，求解關節角度的問題可以轉化為求解兩相鄰區域座標 Z 軸的夾角。利用所定義出來的區域座標系去求解兩座標系的旋轉角度，即可求出兩 Z 軸的夾角，解得所要的關節角度。

求解兩座標系的旋轉角度，首先必需先將兩座標座系轉到同一個座標系統，假設轉到同一個座標系統的兩座標系各為  $[OX_1 \ OY_1 \ OZ_1]$  及  $[OX_2 \ OY_2 \ OZ_2]$ ，可推得出下列關係式

$$[OX_1 \ OY_1 \ OZ_1] = M [OX_2 \ OY_2 \ OZ_2]$$

$$M = [OX_1 \ OY_1 \ OZ_1] * [OX_2 \ OY_2 \ OZ_2]^{-1}$$

所求得的 M 稱為旋轉矩陣，只要將旋轉矩陣 M 轉換成  $R_x(\phi_x) * R_y(\phi_y) * R_z(\phi_z)$  的型式就可以得到關節點的角度了；其中  $R_x(\phi_x)$ 、 $R_y(\phi_y)$ 、 $R_z(\phi_z)$  各為三個旋轉軸的旋轉矩陣。

### 3.4 逆向動力學(Inverse Kinematics)問題描述及定義

當重定完肢節點之後，雖然確保了肢節長度能固定不變，但也使末端肢節端點的位置產生了位移。因此必須利用重定好的人體模型將末端肢節端點的位置移到到所在的目標位置，以符合實際拍攝時所看到的末端肢節端點位置[25]。

末端肢節端點的位置，在正向動力學 (FK) 中，是使用者給定關節角度  $\Theta = (\phi_1, \phi_2, \dots, \phi_n)$  而決定出末端肢節端點的位置 X：

$$X = f(\Theta) \tag{3.1}$$

$$\Theta = (\phi_1, \phi_2, \dots, \phi_n)$$

在逆向動力學 (IK) 則是反向操作，由使用者給定一個目標位置，藉由這個目標位置和已知的肢節長度來反推決定出關節角度，亦即是求上面式子的反函數，

$$X = f^{-1}(\Theta) \tag{3.2}$$

在正向動力學(FK)中，只要給各個關節角度，則末端肢節點的位置即可確定，但在解逆向動力學(IK)並非如此簡單，因為函數  $f$  並非是線性函數，對於給定的目標位置，所求得的關節角度可能會有多組解 [15]，如下圖所示：

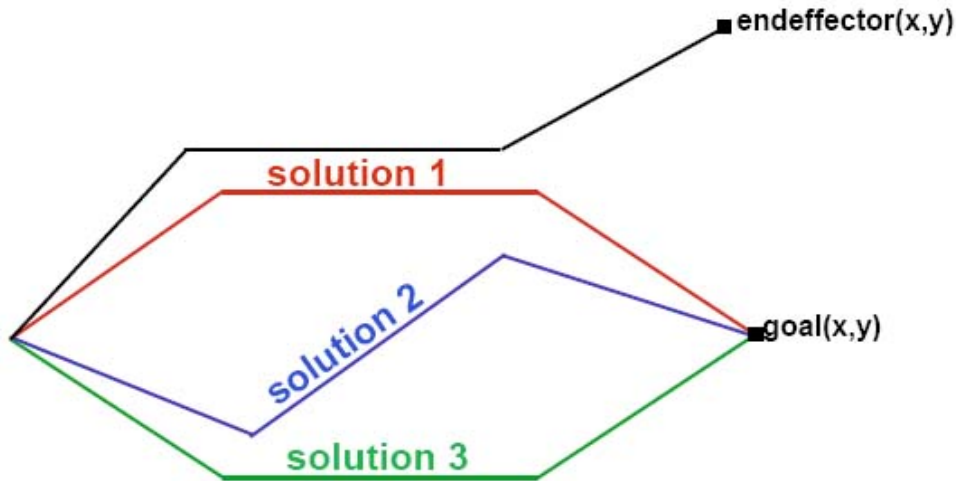


圖 3.4 逆向動力學多重解示意圖

### 3.5 解決逆向動力學的方法：Linearized solution

目前已有不少的逆向動力學(IK)的演算法[14-18]，但對於像人體複雜的肢節模型，要用逆向動力學(IK)來解出關節角度，通常都是利用數值方法來計算。

雖然求解逆向動力學(IK)的問題是非線性的，但如果假設在極短時間內，可以將末端肢節端點的位移量和角度變化量視為是線性的關係，所以將 Eq.(3.1)對角度微分之後可得

$$dX = J(\Theta)d\Theta \quad , \quad J = \frac{\partial f}{\partial \Theta} \quad (3.3)$$

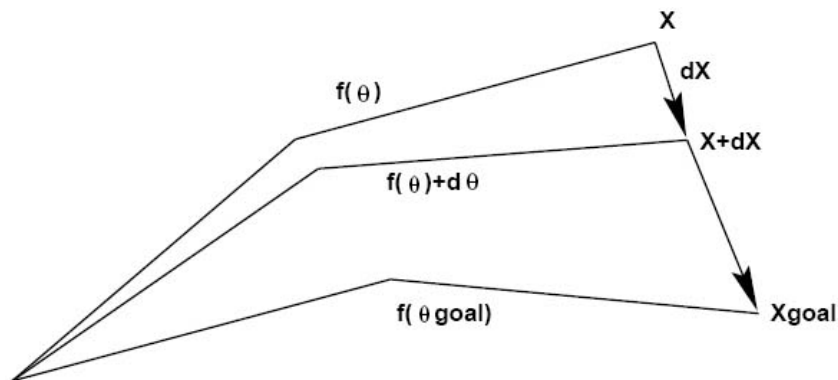


圖 3.5 解逆向動力學疊代過程

J 為函數對  $\Theta$  的偏微分，稱為 Jacobian 矩陣，將 Eq.(3.3) 移項，逆向動力學(IK)的問題就可以經由線性的方法來解。

$$d\Theta = J^{-1}(dX) \quad (3.4)$$

只要能解得 Jacobina 的反矩陣之後，便可求的各個關節的角度變化量，再經過多次的疊代之後，末端肢端點即可達到目標的位置。

然而函數 J 通常不是方陣，基本上來說，是沒有辦法求得反矩陣的。因此，Pseudo inverse 的技術就被應用來解決這個問題。為了和 Jacobina 的反矩陣有所區別，將 Jacobina 的 Pseudo inverse 用  $J^+$  來表示，

$$J^+ = J^T (JJ^T)^{-1} \quad (3.5)$$

則 Eq.(3.4) 變成

$$d\Theta = J^+(dX) \quad (3.6)$$

使用 Pseudo inverse 會使方陣  $J^+$  會導致 redundancy，因此必須讓 redundancy 符合一些次要的條件，在此可增加一多項式在 Eq.(3.6)：

$$d\Theta = J^+(dX) + (I - J^+J)dZ \quad (3.7)$$

$(I - J^+J)=0$  所以  $(I - J^+J)dZ$  並不會對 Eq. (3.6) 式的解有所影響， $(I - J^+J)$  稱為 homogeneous part of solution， $dZ$  是用來描述要符合的次要條件， $(I - J^+J)dZ$  稱為 penalty function。

此方法是用數值方法來解非線性方程式，但在計算的過程中，計算 Pseudo inverse 的時間複雜度相當的高，而且對於 Inverse kinematics 問題也會有解不出來的歧點(Singularity)。

有鑑於 Pseudo inverse 的時間複雜度相當的高，Wolovich 和 Elliot 在 1984[7] 年發表提出使用 Jacobian Transpose  $J^T$  來取代 Pseudo inverse 變成  $\Delta\Theta = J^T \Delta X$ ，可有效的減少計算時間。

在 Jacobian Transpose 方法中[25]，是從物理方面末端肢節點位移所做的功來探討，將目前末端肢節點的位置拉向目標位置之間的位移量為作用力，所作的功和作用在肢節上的力矩所產生的功是相等的(如下圖所示)，則可以得出兩者之間的關係式子：

功 = 力矩 \* 角度 = 作用力 \* 距離

$$\tau \bullet \Delta\Theta = F \bullet \Delta X$$

$$\tau^T * \Delta\Theta = F^T * \Delta X$$

當位移量  $\Delta X$  極小的時候，

根據 Taylor' s Theorem，

$$f(\Theta) = f(\Theta_0) + f'(\Theta_0)(\Theta - \Theta_0)$$

$$\Rightarrow \Delta X = J * \Delta\Theta$$

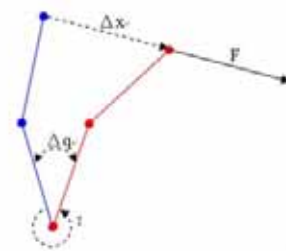


圖 3.6 外力與力矩關係示意圖

代入上式之後，

$$\tau^T \Delta\Theta = F^T * J * \Delta\Theta$$

$$\tau^T = F^T * J$$

$$\tau = J^T * F$$

將上式和  $\Delta\Theta = J^T \Delta X$  做比較，假設目前末端肢節點的位置為  $X_c(t)$ ，目標位置為  $X_d(t)$ ，則誤差估計函數 (error measure) 為：

$$e(t) = X_d(t) - X_c(t)$$

此誤差估計函數可視為是將末端肢節點的位置從  $X_c(t)$  拉到  $X_d(t)$  的作用力  $F$ ，且此作用力能使末端肢節點的位置隨著時間朝著  $X_d(t)$  的位置移動， $\tau$  可視為是角速度 [8]，也就是肢節點的旋轉角度，所以式子可以轉換成

$$\Delta\Theta = J^T \Delta X$$

如此一來，就可以利用每一次疊代產生一個將末端肢節點朝向  $X_d(t)$  的新向量  $\Theta$ ，直到末端肢節點達到目標位置，或是符合其他終止條件，就可求得  $\Theta$  值了。

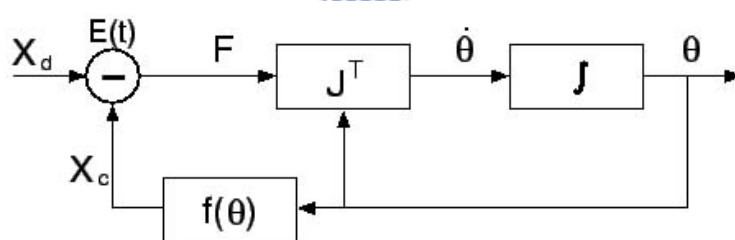


圖 3.7 利用 Jacobian Transpose 解逆向動力學疊代過程

雖然用 Jacobian Transpose 可以有效率的解決問題，但仍會發生非預期的表現。如圖 3.8 所示：

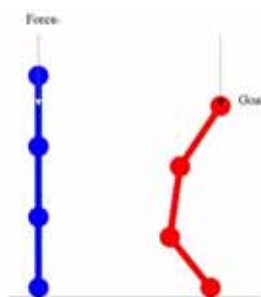


圖 3.8 Jacobian Transpose 發生歧點 (singularity) 示意圖

在圖 3.8 左邊的情況下，以 Jacobian Transpose 方法所得到的解其行為是保持原來狀態不動的。雖然符合物理行為，但卻非所預期的結果(圖 3.8 右邊)。

### 3.6 逆向動力學解法：Cyclic Coordinate Decent (CCD) Method

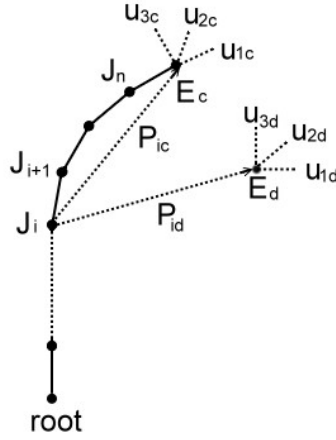


圖 3.9 CCD Method 定義

首先參考圖 3.9 定義

$\vec{w}E_c = ({}^w x_c, {}^w y_c, {}^w z_c)^T$  為目前 end\_effector 在世界座標系的位置。

$\vec{w}O_c = [{}^w u_{1c} \quad {}^w u_{2c} \quad {}^w u_{3c}]^T$  為目前最後一個關節的方向。

$\vec{w}E_d = ({}^w x_d, {}^w y_d, {}^w z_d)^T$  為 end\_effector 所希望到達的位置。

$\vec{w}O_d = [{}^w u_{1d} \quad {}^w u_{2d} \quad {}^w u_{3d}]^T$  為最後一個關節所希望的方向。

$\vec{w}J_j$  = 目前所要轉動關節在世界座標系的位置。

則可得

$$\vec{w}P_{ic} = \vec{w}E_c - \vec{w}J_i, \quad \vec{w}P_{id} = \vec{w}E_d - \vec{w}J_i。$$

$$\vec{i}P_{ic} = \left( \begin{bmatrix} \vec{w}x_i & \vec{w}y_i & \vec{w}z_i \end{bmatrix}^T \right)^{-1} \vec{w}P_{ic}, \quad i=1, 2, \dots, n$$

$$\vec{i}P_{id} = \left( \begin{bmatrix} \vec{w}x_i & \vec{w}y_i & \vec{w}z_i \end{bmatrix}^T \right)^{-1} \vec{w}P_{id}, \quad i=1, 2, \dots, n$$

$$\vec{i}E_c = \left( \begin{bmatrix} \vec{w}x_i & \vec{w}y_i & \vec{w}z_i \end{bmatrix}^T \right)^{-1} [\vec{w}E_c - \vec{w}J_i]^T, \quad i=1, 2, \dots, n$$

$$\vec{i}E_d = \left( \begin{bmatrix} \vec{w}x_i & \vec{w}y_i & \vec{w}z_i \end{bmatrix}^T \right)^{-1} [\vec{w}E_d - \vec{w}J_i]^T, \quad i=1, 2, \dots, n$$

$$\vec{i}O_c = \left( \begin{bmatrix} \vec{w}x_i & \vec{w}y_i & \vec{w}z_i \end{bmatrix}^T \right)^{-1} \vec{w}O_c = [{}^i u_{1c} \quad {}^i u_{2c} \quad {}^i u_{3c}]^T, \quad i=1, 2, \dots, n$$

$$\vec{i}O_d = \left( \begin{bmatrix} \vec{w}x_i & \vec{w}y_i & \vec{w}z_i \end{bmatrix}^T \right)^{-1} \vec{w}O_d = [{}^i u_{1d} \quad {}^i u_{2d} \quad {}^i u_{3d}]^T, \quad i=1, 2, \dots, n$$

其次，定義 penalty function：

$${}^i F_p(q) = \left\| \left( \vec{i}E_d - \vec{i}E_c \right) \right\|^2, \quad i=1, 2, \dots, n$$

$${}^i F_o(q) = \sum_{k=1}^3 \left( \left( \vec{i}u_{kd} \cdot \vec{i}u_{kc} \right) - 1 \right)^2, \quad i=1, 2, \dots, n$$

$q$  為參數(joint variables)。



目的：找出可以使  ${}_i F(q) = {}_i F_p(q) + {}_i F_o(q)$  為最小值的向量  $q$ 。  
就求取上述目的解而言，人體關節構造  $j_i$  可能的自由度分 1 DOF、2 DOF、3 DOF。

### CASE1：若 $j_i$ 為 1 DOF

令  $j_i$  的旋轉軸為  $axis_i$  (以區域座標系而言旋轉軸固定)，轉動角度為  $\phi$ ，

$$\begin{aligned} \overrightarrow{{}_i P'_{ic}}(\phi) &= R_{axis_i}(\phi) \overrightarrow{{}_i P_{ic}}, \quad i=1, 2, \dots, n \\ \overrightarrow{{}_i u'_{kc}}(\phi) &= R_{axis_i}(\phi) \overrightarrow{{}_i u_{kc}}, \quad k=1, 2, 3, \quad i=1, 2, \dots, n \end{aligned}$$

要求  ${}_i F_p(q) = \left\| \overrightarrow{{}_i E_d} - \overrightarrow{{}_i E_c} \right\|^2$  的最小值即是求

$$g_1(\phi) = \overrightarrow{{}_i P_{id}} \cdot \overrightarrow{{}_i P'_{ic}}(\phi) \text{ 的最大值} \quad (3.8)$$

同樣的要求  ${}_i F_o(q) = \sum_{k=1}^3 \left( \overrightarrow{{}_i u_{kd}} \cdot \overrightarrow{{}_i u'_{kc}} - 1 \right)^2$  的最小值即是求

$$g_2(\phi) = \sum_{k=1}^3 w_{ok} \left( \overrightarrow{{}_i u_{kd}} \cdot \overrightarrow{{}_i u'_{kc}}(\phi) \right) \text{ 的最大值} \quad (3.9)$$

結合(3.8)、(3.9)，且再加入權數(weight factor)  $w_p$  則

$$g(\phi) = w_p g_1(\phi) + g_2(\phi) \quad (3.10)$$

解 IK 的問題就變成如下：

找出  $\phi^*$ ，使得  $g(\phi^*) = \max \{ g(\phi) \mid \phi^l \leq \phi \leq \phi^u \}$

where  $\phi^l = \phi - \phi^l$ ， $\phi^u = \phi^u - \phi$ ，

$\phi^l$ 、 $\phi^u$  分別為關節的  $J_i$  轉動角度的下限和上限

$\phi$  則是為目前關節的  $J_i$  轉動角度

又

$$R_{axis_i}(\phi) \overrightarrow{{}_i P_{ic}} = \overrightarrow{axis_i} \left( \overrightarrow{{}_i P_{ic}} \cdot \overrightarrow{axis_i} \right) (1 - \cos_i \phi) + \overrightarrow{{}_i P_{ic}} \cos_i \phi + \left( \overrightarrow{axis_i} \times \overrightarrow{{}_i P_{ic}} \right) \sin_i \phi \quad (3.11)$$

將 Eq.(3.11) 代入 Eq. (3.8)

$$\begin{aligned} g_1(\phi) &= \overrightarrow{{}_i P_{id}} \cdot R_{axis_i}(\phi) \overrightarrow{{}_i P_{ic}} \\ g_1(\phi) &= \left( \overrightarrow{{}_i P_{id}} \cdot \overrightarrow{axis_i} \right) \left( \overrightarrow{{}_i P_{ic}} \cdot \overrightarrow{axis_i} \right) (1 - \cos_i \phi) + \left( \overrightarrow{{}_i P_{id}} \cdot \overrightarrow{{}_i P_{ic}} \right) \cos_i \phi + \overrightarrow{{}_i P_{id}} \cdot \left( \overrightarrow{axis_i} \times \overrightarrow{{}_i P_{ic}} \right) \sin_i \phi \end{aligned} \quad (3.12)$$

同樣的 Eq.(3.9) 可寫成

$$\begin{aligned} g_2(\phi) &= \sum_{k=1}^3 w_{ok} \left( \overrightarrow{{}_i u_{kd}} \cdot R_{axis_i}(\phi) \overrightarrow{{}_i u_{kc}} \right) \\ g_2(\phi) &= \sum_{k=1}^3 w_{ok} \left\{ \left( \overrightarrow{{}_i u_{kd}} \cdot \overrightarrow{axis_i} \right) \left( \overrightarrow{{}_i u_{kc}} \cdot \overrightarrow{axis_i} \right) (1 - \cos_i \phi) + \left( \overrightarrow{{}_i u_{kd}} \cdot \overrightarrow{{}_i u_{kc}} \right) \cos_i \phi + \overrightarrow{{}_i u_{kd}} \cdot \left( \overrightarrow{axis_i} \times \overrightarrow{{}_i u_{kc}} \right) \sin_i \phi \right\} \end{aligned} \quad (3.13)$$

將 Eq.(3.11)、Eq.(3.12) 整合到 Eq.(3.10)

$$g(\phi) = k_1(1 - \cos \phi) + k_2 \cos \phi + k_3 \sin \phi$$

where  $k_1$ 、 $k_2$ 、 $k_3$  皆為係數

$$k_1 = w_p \left( \overrightarrow{P_{id}} \cdot \overrightarrow{axis_i} \right) \left( \overrightarrow{P_{ic}} \cdot \overrightarrow{axis_i} \right) + \left( \sum_{k=1}^3 w_{ok} \left( \overrightarrow{u_{kd}} \cdot \overrightarrow{axis_i} \right) \left( \overrightarrow{u_{kc}} \cdot \overrightarrow{axis_i} \right) \right)$$

$$k_2 = w_p \left( \overrightarrow{P_{id}} \cdot \overrightarrow{P_{ic}} \right) + \sum_{k=1}^3 w_{ok} \left( \overrightarrow{u_{kd}} \cdot \overrightarrow{u_{kc}} \right)$$

$$k_3 = axis_i \cdot \left[ w_p \left( \overrightarrow{P_{ic}} \times \overrightarrow{P_{id}} \right) + \sum_{k=1}^3 w_{ok} \left( \overrightarrow{u_{kc}} \times \overrightarrow{u_{kd}} \right) \right]$$

若關節角度沒有限制的話，則  $g(\phi)$  的最大值成立於

$$\frac{dg(\phi)}{d\phi} = (k_1 - k_2) \sin \phi + k_3 \cos \phi = 0,$$

且

$$\frac{d^2g(\phi)}{d^2\phi} = (k_1 - k_2) \cos \phi - k_3 \sin \phi < 0$$

但實際上關節角度都是有限制的，

令上兩式所求的角度為  $\bar{\phi}$ ，

$$\text{if } \phi^l \leq \bar{\phi} \leq \phi^u$$

$$\text{then } \phi^* = \bar{\phi},$$

$$\text{elseif } g(\phi^u) > g(\phi^l)$$

$$\text{then } \phi^* = \phi^u$$

else

$$\phi^* = \phi^l$$



## CASE2：若 $j_i$ 為 2、3 DOF

可將 n DOF 轉換成 n 個 1 DOF，但在角度限制要另外檢查。

以 3 DOF 為例，令第  $j_i$  的關節為 3 DOF，將 3 DOF 的關節  $j_i$  轉換成 3 個 1 DOF，其關節限制的上下限各為  $\phi_x^u$ 、 $\phi_x^l$ 、 $\phi_y^u$ 、 $\phi_y^l$ 、 $\phi_z^u$ 、 $\phi_z^l$ 。

CCD 方法未執行時， $\overrightarrow{j_i j_{i+1}}$  在第  $j_i$  個關節區域座標系的向量為  $\overrightarrow{V}$ ，假設 CCD 方法執行到第 k 個疊代結束時，

$$\overrightarrow{V}' = {}_i M_{k_i} M_{k-1} \dots {}_i M_1 \overrightarrow{V} = {}_i M^k \overrightarrow{V}, \text{ 其中 } {}_i M^k = {}_i M_{k_i} M_{k-1} \dots {}_i M_1$$

又  ${}_i M^k = R_x(\phi_x) R_y(\phi_y) R_z(\phi_z)$ ，如此一來就可以得到各個旋轉軸的旋轉角度，接下來只要檢查並修正角度是在合理的範圍裡面，若有不合理的角度出現，就重新設定，使角度在合理範圍內而得到新的旋轉角度  $\phi_x^*$ 、 $\phi_y^*$ 、 $\phi_z^*$ ，然後再將角度轉換成旋轉矩陣  ${}_i M^k = R_x(\phi_x^*) R_y(\phi_y^*) R_z(\phi_z^*)$ ，繼續一下次的疊代。每次疊代的次序可以建構成樹狀結構圖如下：

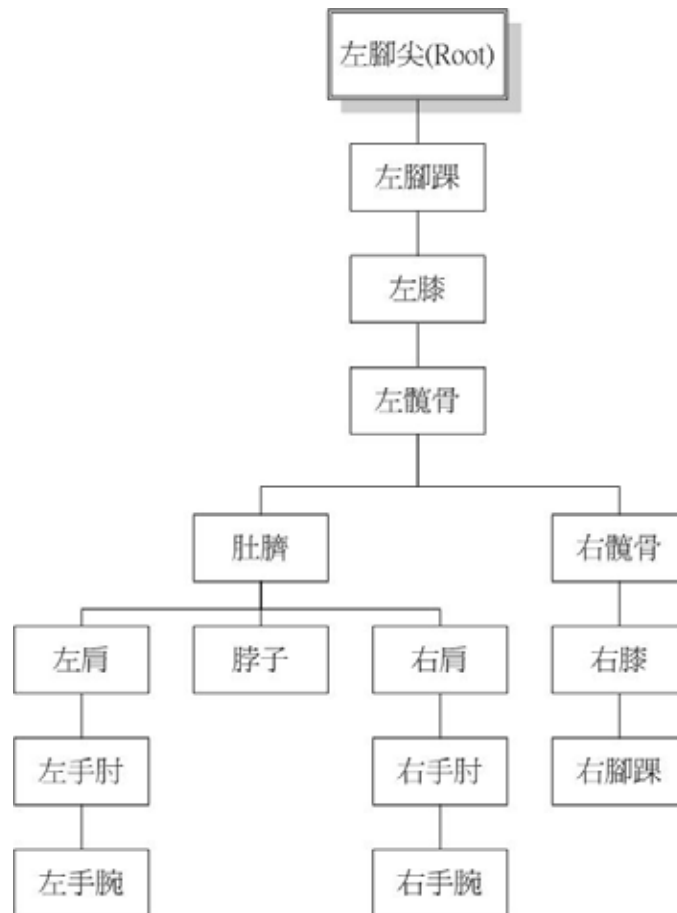


圖 3.10 CCD Method 疊代順序樹狀圖

每次疊代修正的關節順序由樹狀圖的 leaf 開始往 parent 修正，當所有的 children 都修正完之後能輪到本修做修正。假設要修正樹狀圖中的肚臍的話，必需等到樹狀圖所肚臍所有的 children(左肩、脖子、右肩、左手肘、左手腕、右手肘、右手腕)修正完後才能換肚臍做修正，下圖 3.11 為整個 CCD Method 的流程圖。

# CCD Method 流程圖

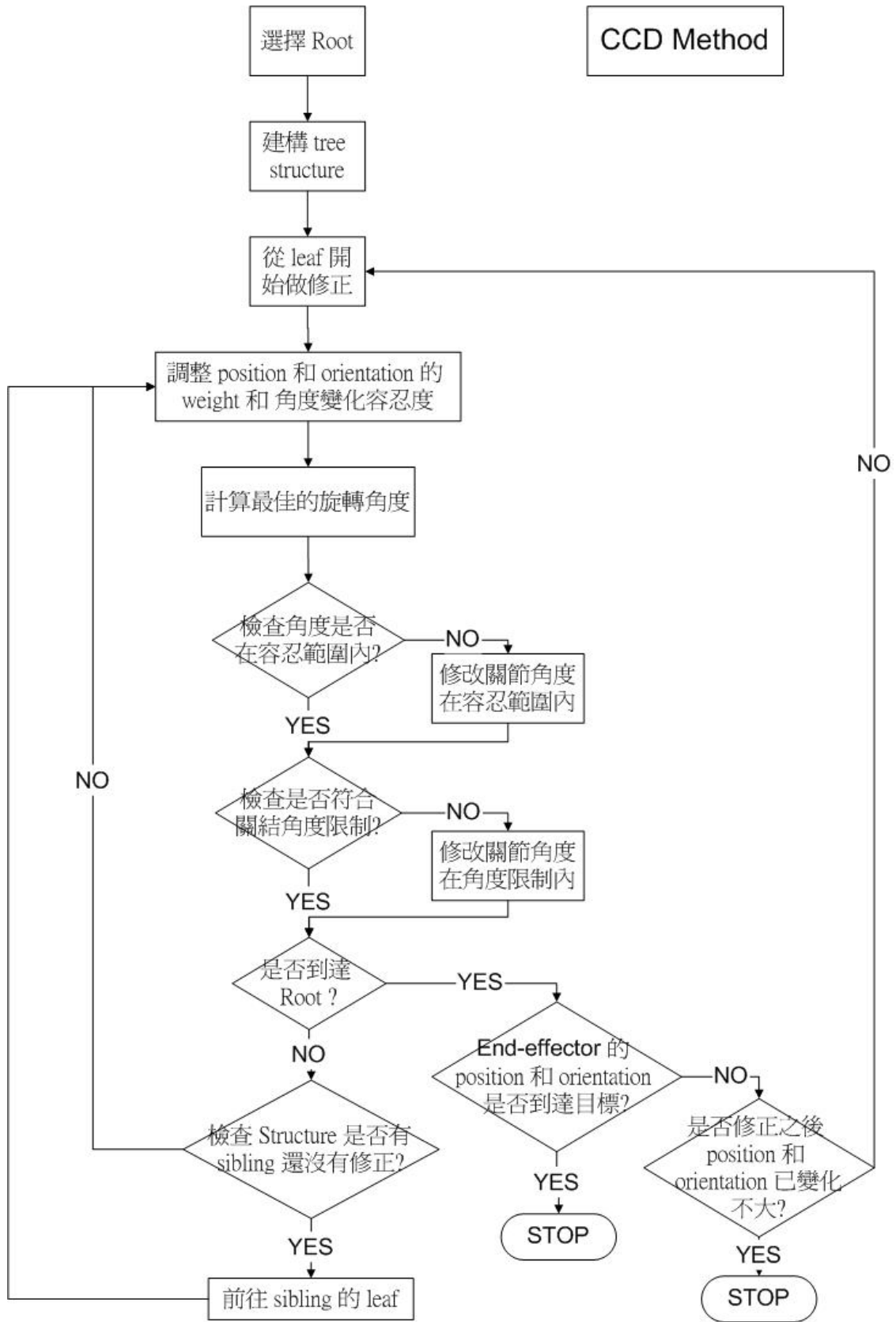


圖 3.11 CCD Method 流程圖

## 3.7 加強 CCD 方法

### 3.7.1 人體肢節角度限制：

因為求解逆向動力學(IK)單純只是數學上的運算求出肢節的旋轉角度，並沒有考量到人體的骨架結構，導致雖然肢節端點可以到達目標位置，卻不是人體所能擺出來的姿勢，所以必須考量人體運動時肢節所能轉動的角度範圍，確保不會有不合理的人體肢節角度出現。當計算出來的角度限制超過資料中人體肢節所能轉動的角度範圍，就將計算出來的角度重新設定在合理的肢節角度範圍內，使其動作能夠符合人體骨架的限制。

下表為美國骨外科學會及美國醫學會測量關節活動度的資料[30]

表 3.1 關節活動角度限制的測量 單位(角度)

關節	動作	美國骨外科學會	Kendall和 McCreary	Hopperfeld	美國醫學會
肩	屈曲	0-180	0-180	0-90	0-150
	伸展	0-60	0-45	0-45	0-50
	外展	0-180	0-180	0-180	0-180
	內轉	0-70	0-70	0-55	0-90
	外轉	0-90	0-90	0-45	0-90
肘	屈曲	0-150	0-145	0-150	0-140
前臂	旋前	0-80	0-90	0-90	0-80
	旋後	0-80	0-90	0-90	0-80
	伸展	0-70	0-70	0-70	0-60
	屈曲	0-80	0-80	0-80	0-60
	橈側偏移	0-20	0-20	0-20	0-20
	尺側偏移	0-30	0-35	0-30	0-30
拇指					
腕掌	外展	0-70	0-80	0-70	0-8cm
	屈曲	0-15	0-45		
	伸展	0-20	0		
	對掌	拇指的指尖到第五根手指的指尖或基部	拇指的指腹到第五根手指的指腹	拇指的指尖到手指的指尖	
掌指	屈曲	0-50	0-60	0-50	0-60
指間	屈曲	0-80	0-80	0-90	0-80



表 3.1(續) 關節活動角度限制的測量 單位(角度)

手指 2~5					
掌指	屈曲	0-90	0-90		
	過度伸展	0-45		0-90	0-90
	外展			0-45	0-20
近端指間	屈曲	0-100		0-100	0-100
遠端指間	屈曲	0-90		0-90	0-70
	過度伸展	0-10		0-10	0-30
腕	屈曲	0-120	0-125	0-135	0-100
	伸展	0-30	0-10	0-30	0-30
	外展	0-45	0-45	0-50	0-40
	內收	0-30	0-10	0-30	0-20
	外轉	0-45	0-45	0-45	0-40
	內轉	0-45	0-45	0-35	0-50
膝	屈曲	0-135	0-140	0-135	0-150
踝	背屈	0-20	0-20	0-20	0-20
	跖屈	0-50	0-45	0-50	0-40
	內翻	0-35	0-35		0-30
	外翻	0-15	0-20		0-20
距下	內翻	0-5		0-5	
	外翻	0-5		0-5	
橫腕	內翻	0-20		0-20	
	外翻	0-10		0-10	
腳趾					
第一橫趾間	屈曲	0-45		0-45	0-30
	伸展	0-70		0-90	0-50
第一趾間	屈曲	0-90			0-30
	伸展	0			0
2~5 橫趾間	屈曲	0-40			
	伸展	0-40			

表 3.1(續) 關節活動角度限制的測量 單位(角度)

第二指趾間	屈曲				0-30
	伸展				0-40
第三指趾間	屈曲				0-20
	伸展				0-30
第四指趾間	屈曲				0-10
	伸展				0-20
第五指趾間	屈曲				0-10
	伸展				0-10
2~5 近端趾間	屈曲	0-35			
2~5 遠端趾間	屈曲	0-60			
頸椎	屈曲	0-45	0-45	下巴碰到前胸	0-60
	伸展	0-45	0-45	看天花板	0-75
	側彎	0-45		0-45	0-45
	旋轉	0-60		下巴轉到肩部	0-80
胸椎	前彎				0-50
	旋轉				0-30
胸椎和腰椎	前彎	0-80 4 in.			
	後仰	0-25			
	側彎	0-35			
	旋轉	0-45			
腰薦椎	後仰				0-25
	側彎				0-25
顎頤	張嘴			三個手指的寬度	
	下頷前突			超過上排牙齒	
	下頷側面突出			超過下排牙齒	

### 3.7.2 角度變化容忍度：

因為經由影像處理所求得的肢節角度雖然會有誤差，但基本上與實際的肢節角度相去不遠，在做修正的時候肢節角度不可能會有大幅度的改變，若求解出來的角度有大幅度改變的話反而會影響到下一次的疊代 (iteration) 做修正。因此可以加入加角度變化容忍度，使得肢節角度於影像處理所求得的肢節角度不會有太大的變化，亦即是讓整個人體的姿勢不會有太大的改變，而實際上，影像處理求出來的人體姿勢原本就不會和實際上人體的姿勢有太大的改變。

基於上面所提到的理念，旋轉的角度過大，之後的疊代在做修正時，勢必會將角度再修正回來，如此一來有可能增加疊代的次數，也有可能導致做修正時陷入錯誤的區域最佳化裡面。所以加入了加角度變化容忍度除了可以使整個系統速度提升之外，也可以避免求得錯誤的肢節角度。由表 3.2 可看出，有加入角度變化容忍度的話，在疊代後的結果能使整個人體的姿勢不會有太大的改變下使 End\_effector 到達所要的目標。

表 3.2 有加角度變化容忍度與沒有加角度變化容忍度比(a)-(b)  
(a) 無角度變化容忍度。單位 (角度)

joint	X 軸旋轉角度差	Y 軸旋轉角度差	Z 軸旋轉角度差
R_shoulder	11.159300	2.110779	28.154774
R_elbow	5.775356	0.000000	0.000000
R_wrist	5.775356	4.832303	0.000000
L_shoulder	9.758395	10.877118	20.558036
L_elbow	11.033809	0.000000	0.000000
L_wrist	0.429275	4.422612	0.000000
Neck	4.030913	0.294248	23.764240
Navel	9.020453	11.776628	10.000000
R_ankle	14.786718	0.087431	0.000000
R_knee	12.422164	0.000000	0.000000
R_hipbone	3.000000	24.895279	31.813163
L_hipbone	34.202308	14.365972	1.497364
L_knee	0.377350	0.000000	0.000000
L_ankle	0.089097	0.000000	0.000000

(b) 有角度變化容忍度。單位 (角度)

joint	X 軸旋轉角度差	Y 軸旋轉角度差	Z 軸旋轉角度差
R_shoulder	1.254728	0.929658	5.717352
R_elbow	5.108931	0.000000	0.000000
R_wrist	5.798829	0.456462	0.000000

L_shoulder	4.230219	2.250052	5.087495
L_elbow	0.943912	0.000000	0.000000
L_wrist	4.123393	1.405526	0.000000
Neck	1.927801	0.145870	5.440050
Navel	3.880050	13.513374	4.095942
R_ankle	12.000000	0.025434	0.000000
R_knee	6.059808	0.000000	0.000000
R_hipbone	3.000000	4.279869	15.000000
L_hipbone	10.000000	14.178224	0.290439
L_knee	5.974182	0.000000	0.000000
L_ankle	3.807206	0.000000	0.000000

### 3.7.3 隨時更新位置(position)和方向(orientation)的權數(weight)

由於 CCD 方法在做修正時，位置誤差 (position error) 和方向誤差 (orientation error) 隨時都在變動，因此要隨時更新位置和方向的權數。當位置誤差修正變小時，有可能導致方向誤差變大，因此要將方向的權數加大，使得下次要做修正時在方向的考量方面加大。在位置誤差在權數調整方面，是以目前的 end\_effector 和目標的距離來做調整，當距離越遠，權數就會加的更大，距離越近，權數就越小。至於方向的權數調整方面，本論文的方向的權數不採用同一個值，而是三個方向向量各別考慮，當方向的夾角越大，權數就會加重，反之則變小。表 3.3 為模擬結果，由實驗數據中可看出，權數隨時更新能讓 CCD 方法收斂的更好。

表 3.3 權數固定和權數隨時更新比較(a)-(c)

(a) 修正前 End\_effector 與目標的位置誤差(cm)和方向誤差(角度)

joint	position error			orientation error		
	X 軸位置差	y 軸位置差	z 軸位置差	X 軸角度差	y 軸角度差	z 軸角度差
foot	28.292493	-4.427365	-3.938767	17.809257	19.458058	9.723260
R_hand	57.308943	14.950596	7.224642	17.268648	52.827193	55.006472
L_hand	53.508804	14.694903	13.463555	9.729935	49.280210	50.304584
head	61.728708	21.893489	-11.385678	18.896609	34.220754	38.819361

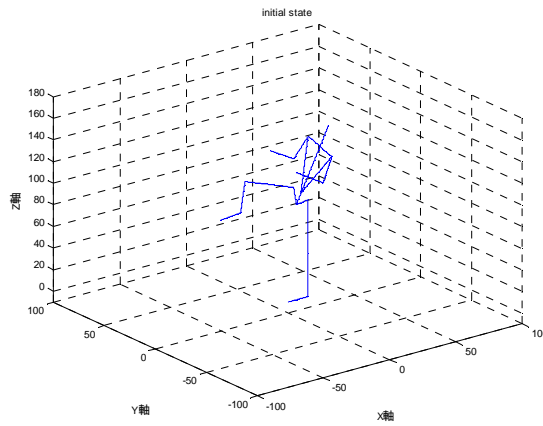
(b)修正後 End\_effector 與目標的位置誤差(cm)和方向誤差(角度)。權數固定

joint	position error			orientation error		
	x 軸位置差	y 軸位置差	z 軸位置差	x 軸角度差	y 軸角度差	z 軸角度差
foot	-0.010400	0.011438	-0.046322	14.535262	21.339514	16.403145
R_hand	-0.007486	-0.004247	0.006663	2.910135	2.874537	0.601138

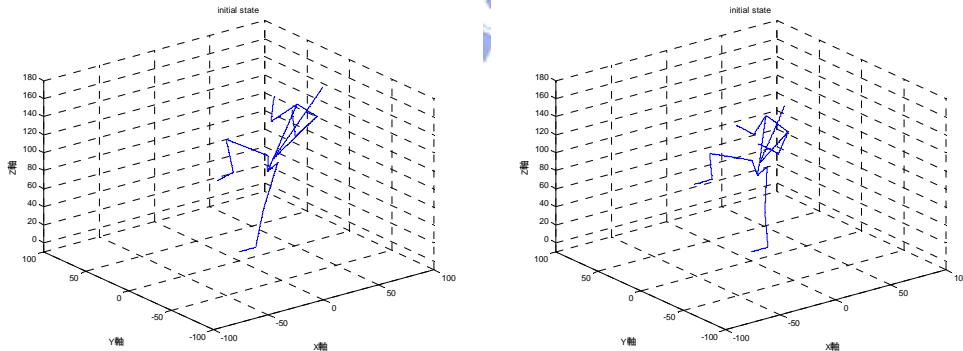
L_hand	-0.012744	-0.003156	0.003091	1.391934	1.524529	0.900296
head	0.030394	0.595016	-2.183021	4.359893	0.635652	4.406011

(c)修正後 End\_effector 與與目標的位置誤差(cm)和方向誤差(角度)。  
權數隨時更新

joint	position error			orientation error		
	x 軸位置差	y 軸位置差	z 軸位置差	x 軸角度差	y 軸角度差	z 軸角度差
foot	0.476309	0.137951	-0.583270	3.376447	7.307588	6.717927
R_hand	0.030684	0.007300	-0.024493	0.000172	0.000888	0.000872
L_hand	0.032817	0.007655	-0.017395	0.001616	0.001982	0.001192
head	-0.046843	0.634962	-1.794233	0.000848	0.007454	0.007421



(a) 模擬出正確的人體動作



(b) 角度加入雜訊的人體動作 (c) 修正完後的結果

圖 3.12 用 CCD 方法將錯誤的姿勢修正回來(a)-(c)

圖 3.12(a)為利用 MATLAB 給定各個關節角度，模擬出人體擺出的姿勢。圖 3.12(b)為圖 3.12(a)中各個關節角度加上雜訊，使得人體的整個姿勢偏掉，造成 End\_effector 位置和方向的誤差。圖 3.12(c)利用 CCD 方法，修正 End\_effector 位置和方向的誤差，使得人體的姿勢跟圖 3.12(a)相似。

## 四. 人體影像擷取

### 4.1 場影變化的偵測

要得到人體各個肢節點的位置，就必須先在 2D 的影像當中找出肢節點的位置及肢節方向，然後再經由第二章所提到空間中點位置及線段方向的求法將 2D 上的影像轉換成 3D 的資訊，因此在做完相機校正後，要先將人體從背景影像中擷取出來，才能進一步的獲取想要得到的資訊。

為了將系統簡化，本論文只考慮將人體從單純的背景影像中擷取出來。故本節參考[23]提出一套方法，在單純背景中，能簡易快速的將場景變化偵測出來，亦即將影像中將人體的部份擷取出來。

場景變化偵測[19-21]主要的目的是偵測場景中新植入的物件，亦即將攝影機影像中的前景(人或任何事物)與背景(固定不變的場景)分離。而要分離影像中的前景與背景部份，最簡單的方法就是將事先拍攝的背影圖與有人體在做運動的影像中的每個像點進行比對。如圖 4.1：



(a) Camera 1, 背景影像



(b) Camera 1, 人體動作影像



(c) Camera 2, 背景影像



(d) Camera 2, 人體動作影像

圖 4.1 攝影機所拍攝之影像(a)-(d)



然而如果單純的將兩張影像中每個相對應的像點分別對 R、G、B 的值做相減並無法去除光線變化的部份，影像中因光線變化或陰影蔽的部份會使該像點的 R、G、B 值增大或減小而被偵測為非背景。如圖 4.2：



圖 4.2 場影變化偵測

#### 4.1.1 去除光線變化

實際上，考慮光線變化或陰影遮蔽只是物理作用，並不會改變物體本身的顏色，只會使得物體的亮度增加或減少。基於此理論，可將影像中每像點的 R、G、B 值轉換成 H、S、I，在[22]中有詳細說明如何將 R、G、B 值轉換成 H、S、I，其轉換式子如下：

$$I = \frac{R+G+B}{3}$$

$$H = \cos^{-1} \frac{[(R-G)+(R-B)]/2}{[(R-G)^2+(R-B)(G-B)]^{1/2}}$$

$$S = 1 - \frac{3}{R+G+B} \min(R, G, B)$$

經過轉換之後就可得到該像點的亮度、色調、飽和度。如此一來，只要用色調來比對，就可以消除光線變化或陰影遮蔽的部份。



圖 4.3 去除陰影

### 4.1.2 去除雜訊

然而，並非所有經過顏色比對後的像點就是我們所預期的結果，會因為空間中的灰塵或微小粒子飄浮等雜訊造成像點顏色的改變，造成誤差而影響比對的結果。

由於所要偵測的部份為人體，而雜訊所造成的誤差在影像中只是小小一區塊，因此只要在之前比對完後的影像中找出最大一區塊，就可以將其他的小區塊重新設定為背景。

取區塊的方法參考[26]，在相鄰的判斷上採用的是 8-connected 的判斷法，只要是在該像點相臨的八個位置上即視為相鄰。每個區塊以不同的標記(Labeling)，進行區塊標記(component Labeling)，如下圖所示：

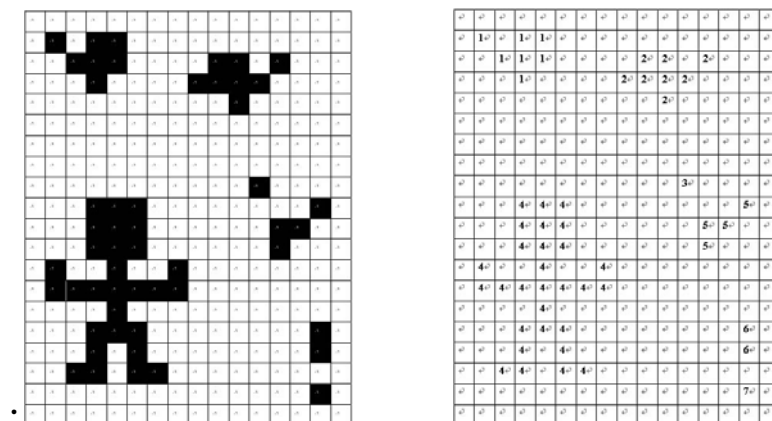


圖 4.4 component Labeling 示意圖

將區塊標記之後就可以計算每個標記的個數，找出標記最多的那一個區塊即是人體在影像中所佔的部份，之後只要再將區塊中的小破洞補好，就可以得到所要的人體區塊。如下圖：



圖 4.5 去除雜訊

## 4.2 肢節方向偵測

本論文在肢節偵測上是採用人工的方式，將肢節的大概方向畫出來，並且用不同的標誌標記不同的肢節，再由程式去讀取並將 2D 的資訊經由第二章所提到的空間中線段方向的求法得出肢節在 3D 的方向。

# 第五章 實驗數據與結果

## 5.1 模擬解決 IK 問題結果

在論文的研究過程中，為了檢測實驗的正確性，在實驗中，首先部份利用 MATLAB 6.5 模擬出可給定角度參數的人體架構，在角度的參數加入雜訊之後使的人體的動作改變，再利用本論文的方法將人體的動作修正回來，再和模擬出的人體動作、角度做比對來驗證實驗結果。

表 5.1 End\_effector 位置誤差和方向誤差收斂比較

(a) 修正前 End\_effector 與目標的位置誤差(cm)和方向誤差(角度)

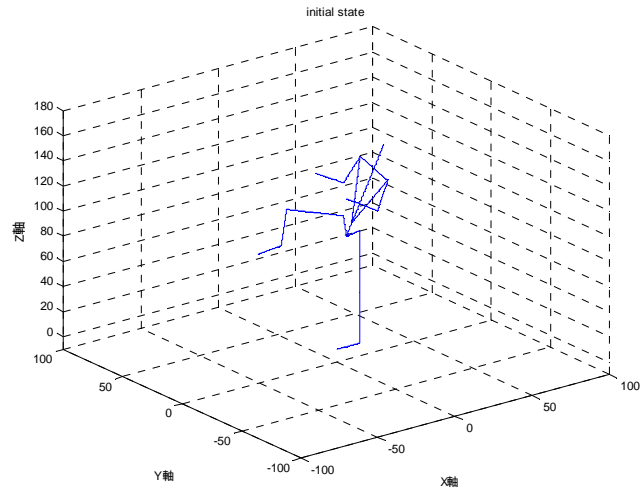
joint	position error			orientation error		
	X 軸位置差	y 軸位置差	z 軸位置差	X 軸角度差	y 軸角度差	z 軸角度差
foot	28.292493	-4.427365	-3.938767	17.809257	19.458058	9.723260
R_hand	57.308943	14.950596	7.224642	17.268648	52.827193	55.006472
L_hand	53.508804	14.694903	13.463555	9.729935	49.280210	50.304584
head	61.728708	21.893489	-11.385678	18.896609	34.220754	38.819361

(b)修正後 End\_effector 與目標的位置誤差(cm)和方向誤差(角度)。  
權數固定

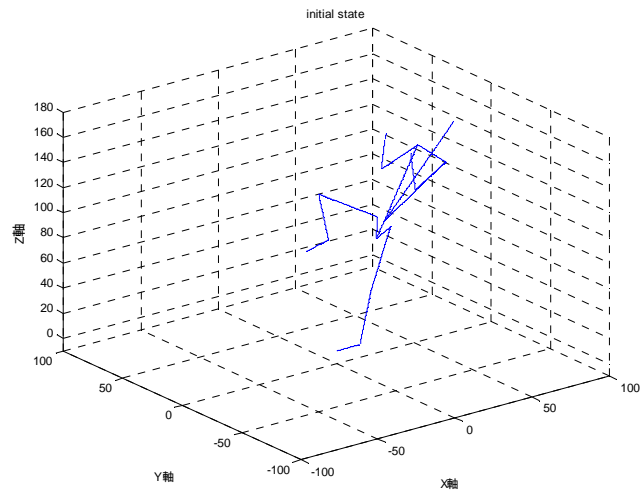
joint	position error			orientation error		
	x 軸位置差	y 軸位置差	z 軸位置差	x 軸角度差	y 軸角度差	z 軸角度差
foot	-0.010400	0.011438	-0.046322	14.535262	21.339514	16.403145
R_hand	-0.007486	-0.004247	0.006663	2.910135	2.874537	0.601138
L_hand	-0.012744	-0.003156	0.003091	1.391934	1.524529	0.900296
head	0.030394	0.595016	-2.183021	4.359893	0.635652	4.406011

(c)修正後 End\_effector 與目標的位置誤差(cm)和方向誤差(角度)。  
權數隨時更新

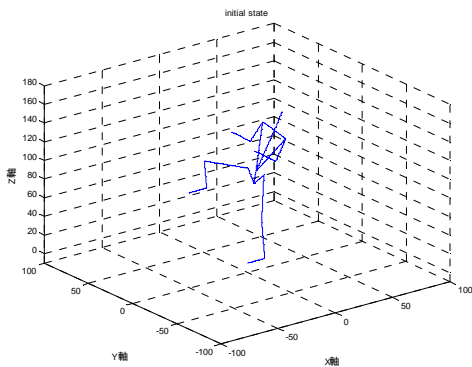
joint	position error			orientation error		
	x 軸位置差	y 軸位置差	z 軸位置差	x 軸角度差	y 軸角度差	z 軸角度差
foot	0.476309	0.137951	-0.583270	3.376447	7.307588	6.717927
R_hand	0.030684	0.007300	-0.024493	0.000172	0.000888	0.000872
L_hand	0.032817	0.007655	-0.017395	0.001616	0.001982	0.001192
head	-0.046843	0.634962	-1.794233	0.000848	0.007454	0.007421



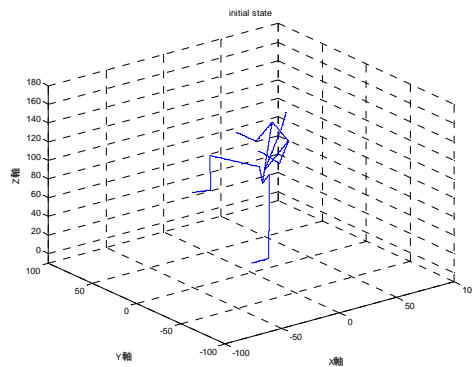
(a) 模擬出正確的人體動作



(b) 角度加入雜訊的人體動作

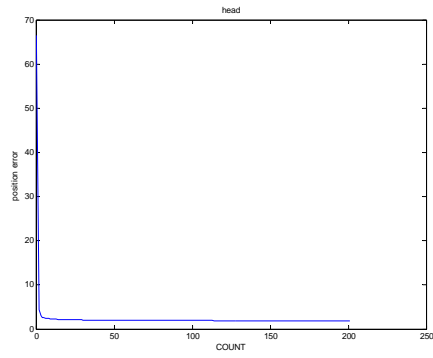


(c) 修正完後的結果  
隨時更新權數

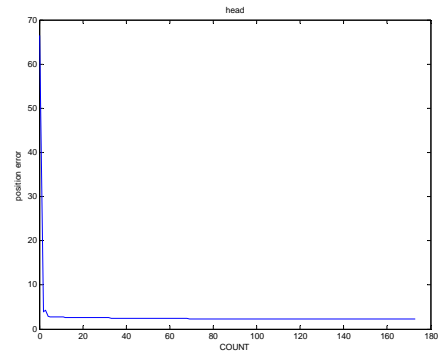


(d) 修正完後的結果  
固定權數

圖 5.1 模擬實驗結果

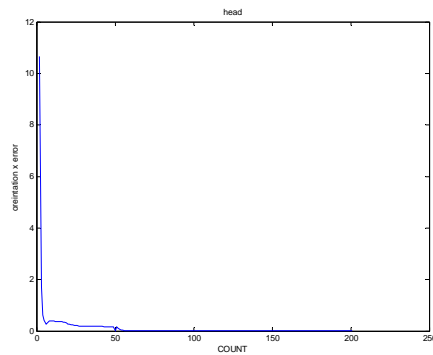


(a1) 隨時更新權數

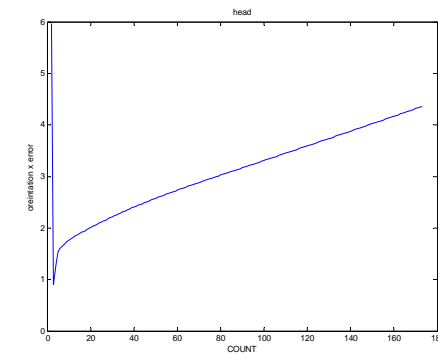


(a2) 固定權數

頭部位置誤差收斂比較圖

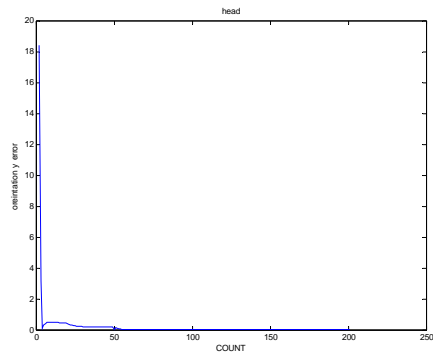


(b1) 隨時更新權數

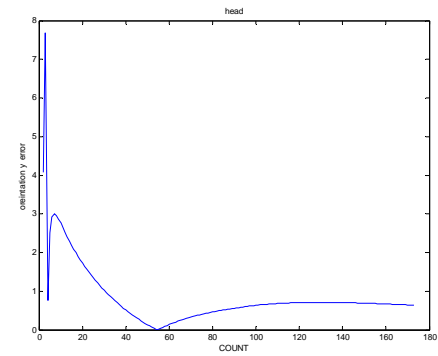


(b2) 固定權數

頭部 X 軸方向角度誤差收斂比較圖



(c1) 隨時更新權數

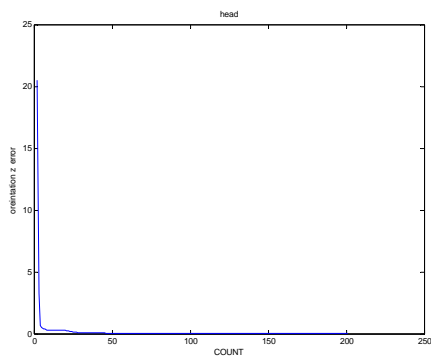


(c2) 固定權數

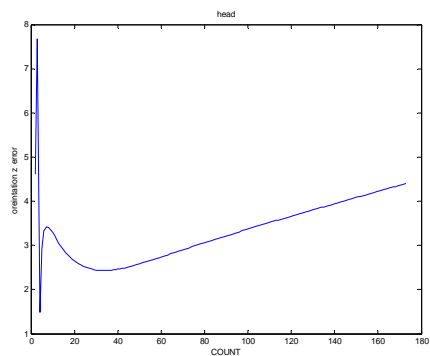
頭部 Y 軸方向角度誤差收斂比較圖

圖 5.2 收斂比較圖



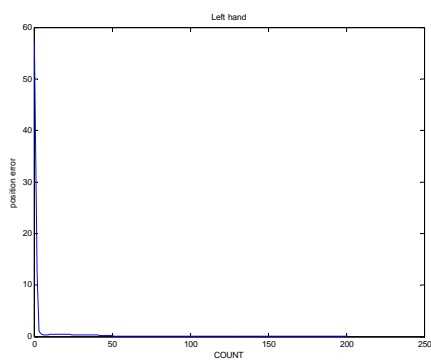


(d1) 隨時更新權數

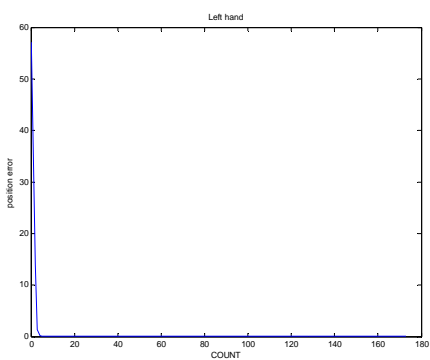


(d2) 固定權數

頭部 Z 軸方向角度誤差收斂比較圖

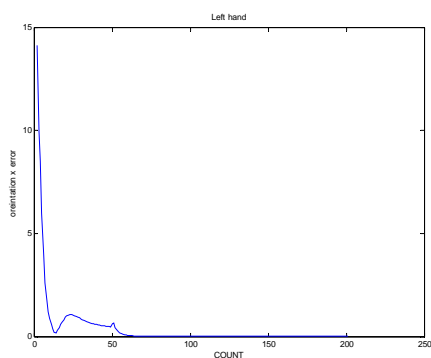


(a1) 隨時更新權數

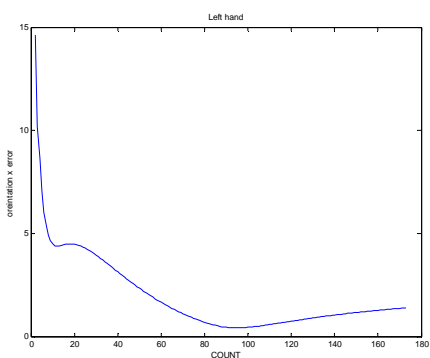


(a2) 固定權數

左手位置誤差收斂比較圖



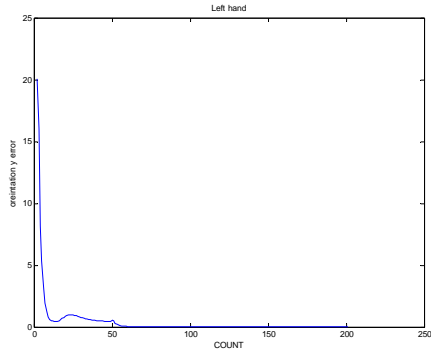
(b1) 隨時更新權數



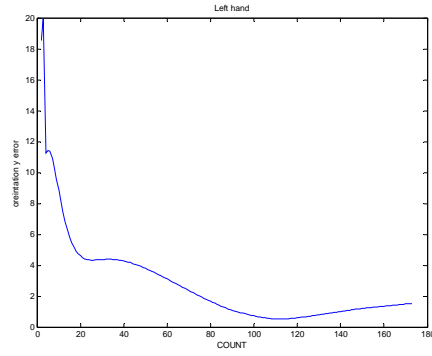
(b2) 固定權數

左手 X 軸方向角度誤差收斂比較圖

圖 5.2(續) 收斂比較圖

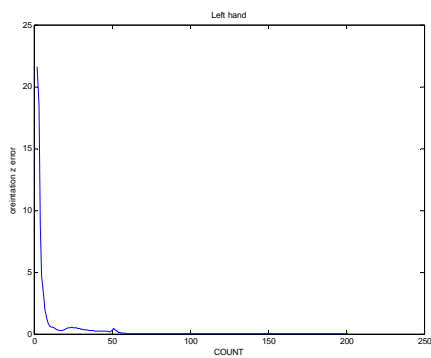


(c1) 隨時更新權數

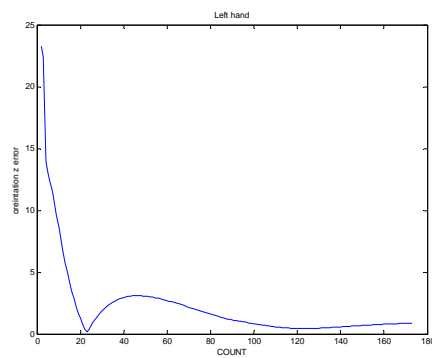


(c2) 固定權數

左手 Y 軸方向角度誤差收斂比較圖

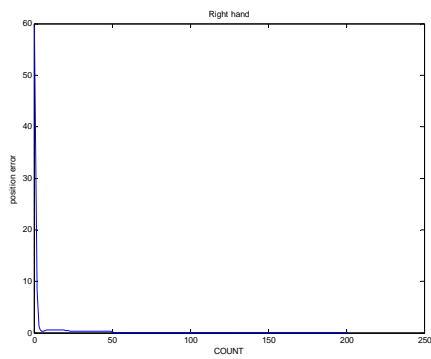


(d1) 隨時更新權數

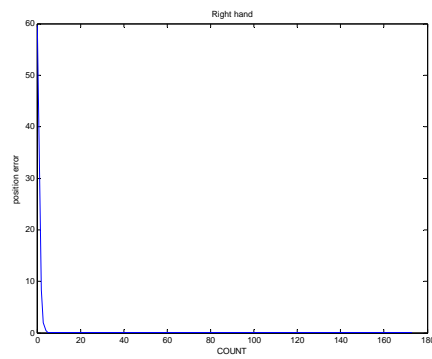


(d2) 固定權數

左手 Z 軸方向角度誤差收斂比較圖



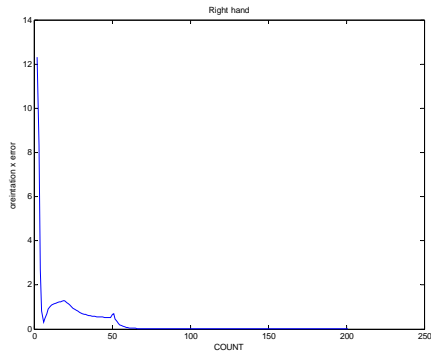
(a1) 隨時更新權數



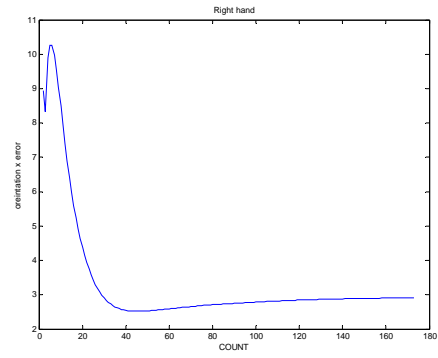
(a2) 固定權數

右手位置誤差收斂比較圖

圖 5.2(續) 收斂比較圖

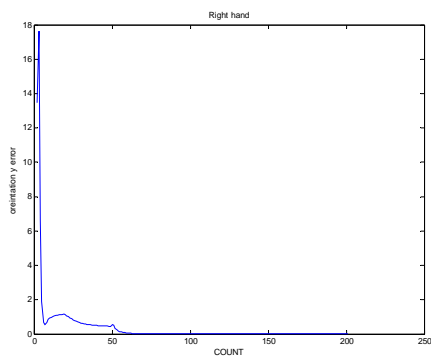


(b1) 隨時更新權數

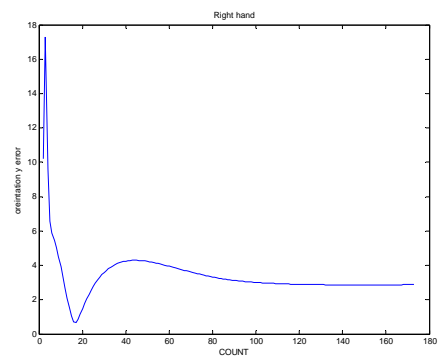


(b2) 固定權數

右手 X 軸方向角度誤差收斂比較圖

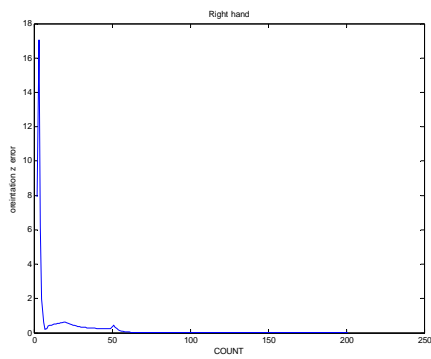


(c1) 隨時更新權數

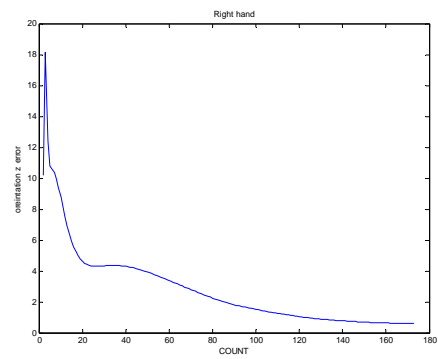


(c2) 固定權數

右手 Y 軸方向角度誤差收斂比較圖



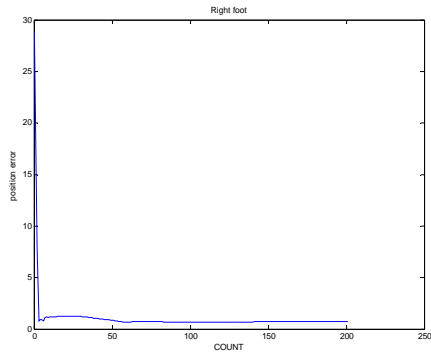
(d1) 隨時更新權數



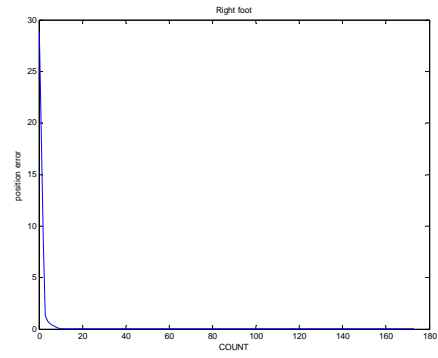
(d2) 固定權數

右手 Z 軸方向角度誤差收斂比較圖

圖 5.2(續) 收斂比較圖

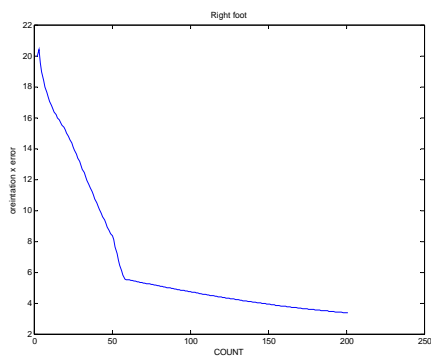


(a1) 隨時更新權數

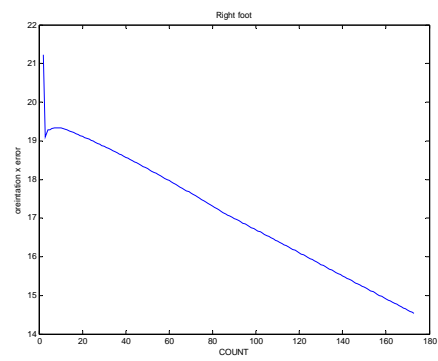


(a2) 固定權數

右腳位置誤差收斂比較圖

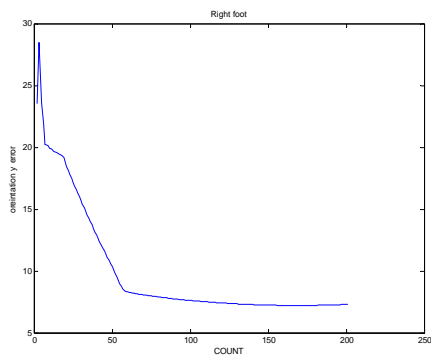


(b1) 隨時更新權數

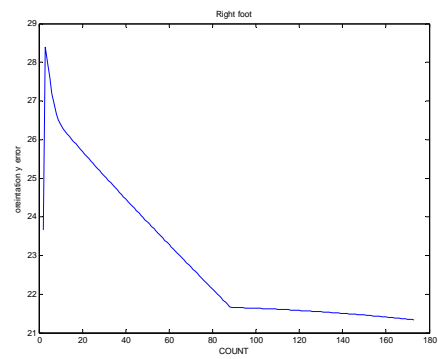


(b2) 固定權數

右腳 X 軸方向角度誤差收斂比較圖



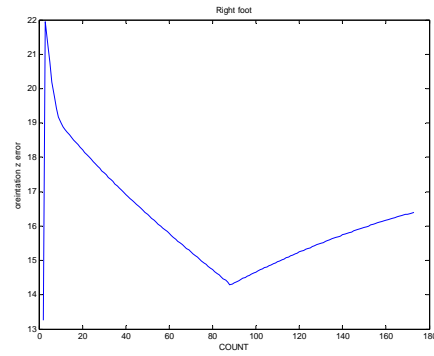
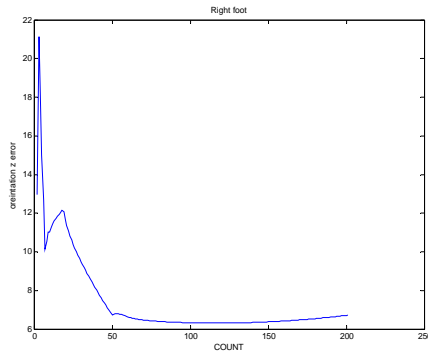
(c1) 隨時更新權數



(c2) 固定權數

右腳 Y 軸方向角度誤差收斂比較圖

圖 5.2(續) 收斂比較圖



(d1) 隨時更新權數

(d2) 固定權數

右腳 Z 軸方向角度誤差收斂比較圖

圖 5.2(續) 收斂比較圖

加入角度變化容忍度的有無雖然不會使 End\_effector 收斂的效果更好，但卻能讓內部隱藏的角度資訊變化不會太大；換句話說，就是能夠盡量在不改變姿勢的情形下將 End\_effector 收斂到所要達到的目標。

表 5.2 有加角度變化容忍度與沒有加角度變化容忍度的比較表(a)-(e)

(a)修正前 End\_effector 與目標的位置誤差(cm)和方向誤差(角度)

joint	position error			orientation error		
	x 軸位置差	y 軸位置差	z 軸位置差	x 軸角度差	y 軸角度差	z 軸角度差
foot	28.292493	-4.427365	-3.938767	17.809257	19.458058	9.723260
R_hand	57.308943	14.950596	7.224642	17.268648	52.827193	55.006472
L_hand	53.508804	14.694903	13.463555	9.729935	49.280210	50.304584
head	61.728708	21.893489	-11.385678	18.896609	34.220754	38.819361

(b)修正後 End\_effector 與目標的位置誤差(cm)和方向誤差(角度) (無角度變化容忍度)

joint	position error			orientation error		
	x 軸位置差	y 軸位置差	z 軸位置差	x 軸角度差	y 軸角度差	z 軸角度差
foot	0.194973	-0.130546	-0.081912	3.499468	2.218099	3.880775
R_hand	-0.087773	-0.206583	-0.084807	0.488538	0.414206	0.259697
L_hand	0.110021	-0.137182	0.015583	0.203689	0.203479	0.084310
head	-0.010669	0.516356	-1.461465	0.117631	0.117882	0.007709

(c)修正後 End\_effector 與目標的位置誤差(cm)和方向誤差(角度) (有角度變化容忍度)

joint	position error			orientation error		
	x 軸位置差	y 軸位置差	z 軸位置差	x 軸角度差	y 軸角度差	z 軸角度差
foot	0.194973	-0.130546	-0.081912	3.499468	2.218099	3.880775
R_hand	-0.087773	-0.206583	-0.084807	0.488538	0.414206	0.259697
L_hand	0.110021	-0.137182	0.015583	0.203689	0.203479	0.084310
head	-0.010669	0.516356	-1.461465	0.117631	0.117882	0.007709

foot	0.476309	0.137951	-0.583270	3.376447	7.307588	6.717927
R_hand	0.030684	0.007300	-0.024493	0.000172	0.000888	0.000872
L_hand	0.032817	0.007655	-0.017395	0.001616	0.001982	0.001192
head	-0.046843	0.634962	-1.794233	0.000848	0.007454	0.007421

(d)內部隱藏的角度資訊，無角度變化容忍度。單位（角度）

joint	X 軸旋轉角度差	Y 軸旋轉角度差	Z 軸旋轉角度差
R_shoulder	11.159300	2.110779	28.154774
R_elbow	5.775356	0.000000	0.000000
R_wrist	5.775356	4.832303	0.000000
L_shoulder	9.758395	10.877118	20.558036
L_elbow	11.033809	0.000000	0.000000
L_wrist	0.429275	4.422612	0.000000
Neck	4.030913	0.294248	23.764240
Navel	9.020453	11.776628	10.000000
R_ankle	14.786718	0.087431	0.000000
R_knee	12.422164	0.000000	0.000000
R_hipbone	3.000000	24.895279	31.813163
L_hipbone	34.202308	14.365972	1.497364
L_knee	0.377350	0.000000	0.000000
L_ankle	0.089097	0.000000	0.000000

(e)內部隱藏的角度資訊，有角度變化容忍度。單位（角度）

joint	X 軸旋轉角度差	Y 軸旋轉角度差	Z 軸旋轉角度差
R_shoulder	1.254728	0.929658	5.717352
R_elbow	5.108931	0.000000	0.000000
R_wrist	5.798829	0.456462	0.000000
L_shoulder	4.230219	2.250052	5.087495
L_elbow	0.943912	0.000000	0.000000
L_wrist	4.123393	1.405526	0.000000
Neck	1.927801	0.145870	5.440050
Navel	3.880050	13.513374	4.095942
R_ankle	12.000000	0.025434	0.000000
R_knee	6.059808	0.000000	0.000000
R_hipbone	3.000000	4.279869	15.000000
L_hipbone	10.000000	14.178224	0.290439
L_knee	5.974182	0.000000	0.000000
L_ankle	3.807206	0.000000	0.000000



## 5.2 實際拍攝結果



camera 1，拍攝原始影像

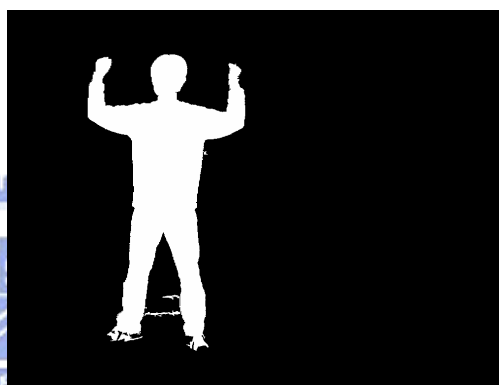


camera 2，拍攝原始影像

圖 5.3 原始影像，時間  $T=1$



camera 1，將人體從背景中擷取出來



camera 2，將人體從背景中擷取出來

圖 5.4 將人體從背景中擷取出來，時間  $T=1$



camera 1，標示 end\_effector 及肢節向量



camera 2，標示 end\_effector 及肢節向量

圖 5.5 標示 end\_effector 肢節向量，時間  $T=1$



(a)camera1 , 肢節點重定後



(b)camera 1 , 利用 CCD 方法修正結果

圖 5.6 IK 修正 End\_effector , 時間 T=1



(c) camera 2，肢節點重定後



(d) camera 2，利用 CCD 方法修正結果

圖 5.6(續) IK 修正 End\_effector，時間 T=1



camera 1，拍攝原始影像



camera 2，拍攝原始影像

圖 5.7 原始影像，時間 T=10



camera 1，將人體從背景中擷取出來



camera 2，將人體從背景中擷取出來

圖 5.8 將人體從背景中擷取出來，時間 T=10



camera 1，標示 end\_effector 及肢節向量



camera 2，標示 end\_effector 及肢節向量

圖 5.9 標示 end\_effector 肢節向量，時間 T=10



(a)camera1，肢節點重定後



(b)camera 1，利用 CCD 方法修正結果

圖 5.10 IK 修正 End\_effector，時間 T=10





(c) camera 2，肢節點重定後



(d) camera 2，利用 CCD 方法修正結果

圖 5.10(續) IK 修正 End\_effector，時間 T=10





camera 1，拍攝原始影像

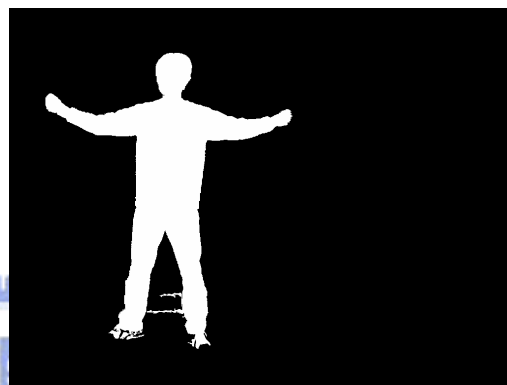


camera 2，拍攝原始影像

圖 5.11 原始影像，時間 T=18



camera 1，將人體從背景中  
擷取出來



camera 2，將人體從背景中  
擷取出來

圖 5.12 將人體從背景中擷取出來，時間 T=18



camera 1，標示 end\_effector  
及肢節向量



camera 2，標示 end\_effector  
及肢節向量

圖 5.13 標示 end\_effector 肢節向量，時間 T=18



(a)camera1，肢節點重定後



(b)camera 1，利用 CCD 方法修正結果

圖 5.14 IK 修正 End\_effector ，時間 T=18



(c) camera 2，肢節點重定後



(d) camera 2，利用 CCD 方法修正結果

圖 5.14(續) IK 修正 End\_effector，時間 T=18

camera1 原始影像    camera 2 原始影像    Motion Capture 骨架圖

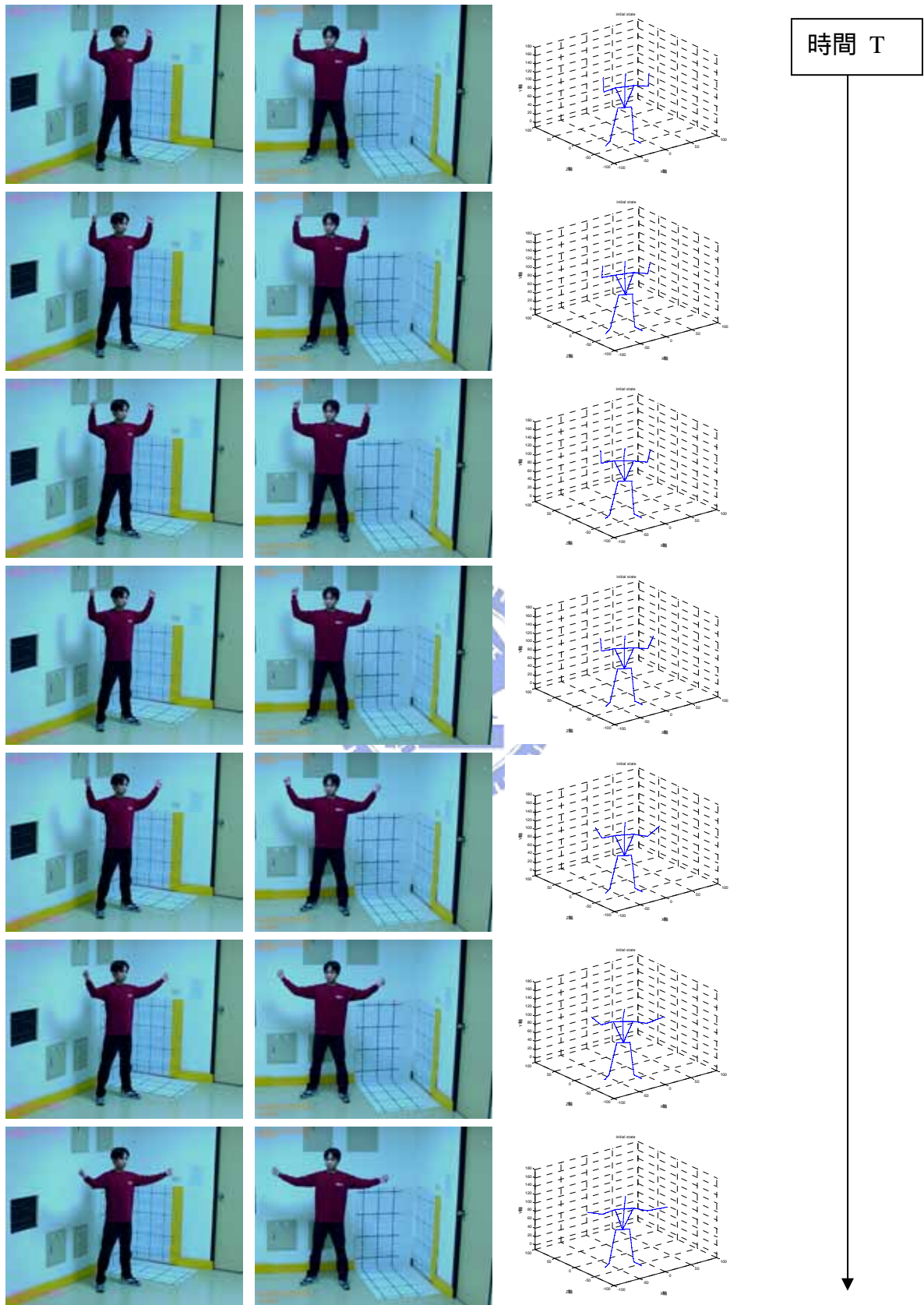


圖 5.15 一序列 Motion Capture 的骨架圖



# 第六章 結論與未來工作

## 6.1 結論

本篇論文使用二台攝影機，架設在適當的位置並做好相機校正的動作，將所拍攝的影像把人體的影像擷取出來並求得 End\_effector 的和肢節向量的資訊後，再將這些資訊餵進本論文研究改善的逆向動力學解法，求得各個肢節點位置。

本論文藉由基本 CCD 方法的概念，成功的將 CCD 方法的收斂結果改善了許多，也使得 CCD 方法對於應用在 n DOF 上能有更進一步的成果展現。將 CCD 方法成功套用在 3D 的人體肢節活動的辨識上，使本論文在不貼光點的情況下，只要能求得大概的肢節方向，依然能由 2D 的影像中取得 3D 人體動作的資訊。

## 6.2 未來工作



由於本系統只考慮在單純的環境底下進行人體動作捕捉的實驗，對於應用在複雜的環境仍需加入其他演算法。未來的工作期望可朝下列幾點改進。

1. 因實驗設備有限及拍攝場地的限制，只能使用二台攝影機拍攝，若未來能使用更多台攝影機拍攝，將能使整個系統更為精確。
2. 由於人體各肢節的長度因人而異，本論文採取人為介入設定的方式，以後若是能由自動辨視各肢節長度或是由 3D 掃描器來產生，就可以使本論文的應用更加方便有效率。
3. 另外在各肢節的方向判斷上，本論文目前是先以人工方式，在重要的幾個 frame 上畫出各肢節的向量。若未來能以自動化的方式來產生這些肢節的向量，必能使本論文的實用性大大提升。
4. 在人體影像捕捉方法中，為了簡化實驗及加速系統效能，只考慮了單純背景來實作，若未來能加以改善成有效且快速的方法能在複雜的背景中將人體影像捕捉出來，則能使本論文的方法應用更廣泛。

## 參考文獻

1. T. Moeslund "Summaries of 107 computer vision-based human motion capture papers", Technical Report LIA 99-01, University of Aalborg, March 1999.
2. T. Moeslund "Computer vision-based human motion capture—a survey", Technical Report LIA 99-02, University of Aalborg, March 1999.
3. D.M. Gavriila and L.S. Davis, "3-D model-based tracking of human in action: A multi-view approach," in Proc. IEEE Int. Conf. Computer Vision pattern Recognition, CVPR' 96, pp. 73-80, June 1996.
4. Gray R. Bradski and James Davis, "Motion Segmentation and Pose Recognition with Motion History Gradients" Proc IEEE Workshop on Applications of Computer Vision, pp. 238-244, Dec. 2000.
5. Jia-Ching Cheng and Jose M. F. Moura, "Capture and Representation of Human Walking in Live Video Sequences", IEEE Trans On Multimedia, Vol.1, No.2, pp. 144-156, June 1999.
6. M. K. Leung and Y.H. Yang, "First Sight: A Human Body Outline Labeling System" IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.17, no.4, pp. 359-377, April 1995.
7. W.A. Wolovich and H. Elliot. "A computational technique for inverse kinematics" In Proceedings of the 23rd Conference on Decision and Control, December 1984.
8. Z.R. Novakovic, B. Nemeč. "A solution of the inverse kinematics problem using the sliding mode." IEEE Transactions on robotics and automation, 1990.
9. Grigore Burdea, and Philippe Coiffect, "Virtual Reality Technology", John Wiley & Sons, New Yourk N.Y. 1994
10. Joe Gradecki, "The Virtual Reality Construction Kit", John Wiley & Sons, New York N.Y. 1994
11. Frederick H. Raab, Ernest B. Blood, Terry O. Steiner, and Herbert R. Jones, "Magnetic Position and Orientation Tacking System", IEEE Transcations on Aerospace and Electronic System, vol. AeS-15, no.5, pp.709-717. September 1979
12. Jih-fang Wang, Vernon Chi and Henry Fuchs, "A Real-time Optical 3D Tracker For Head-mounted Display System", Proc. Of IEEE Virtual Reality Annual International Symposium, Seattle,

- W.A., September 1993.
13. Jun Rekimoto, "A Vision-Based Head Tracker for Fish Tank Virtual Reality-VR without Head Gear", Proc. of 1995 IEEE Virtual Reality Annual International Symposium, Research Triangle Park, N.C., pp.94-100, March 1995.
  14. Li-Chun Tommy Wang and Chih Cheng Chen, "A Combined Optimization Methods for Solving the Inverse Kinematics Problem of Mechanical Manipulators", IEEE Transactions on Robotics and Automation, Vol. 7, NO. 4, August 1991
  15. A. Sezgin Abal, "Animation of human option with inverse kinematics using nonlinear programming", b\_ilkent university, master thesis, September, 2001
  16. Chris Welman, "Inverse kinematics and geometric constraints for articulated figure manipulation", Simon Fraser University, master thesis, September 1993
  17. Jianmin Zhao and Norman I. Badler, "Inverse Kinematics Positioning Using Nonlinear Programming for Highly Articulated Figures", University of Pennsylvania, ACM Transactions on Graphics, Vol. 13, No.4, October1994, Pages313 336
  18. Naganand Madhavapeddy and Stuart Ferguson, "Specialised Constraints for an Inverse Kinematics Animation System Applied to Articulated Figures", The Queen's University of Belfast
  19. E. Durucan and T. Ebrahimi, "Change Detection and Background Extraction by Linear Algebra," Proceedings of the IEEE, Vol. 89, No.10, pp.1368-1381, Oct. 2001.
  20. T. Aach, et al., "Statistical model-based change detection in moving video," Signal Processing, Vol.31, pp.165-180, 1993.
  21. A. Elgammal, D. Harwood, and L. Davis "Non-parametric Model for Background Subtraction", Proc IEEE FRAME-RATE Workshop,
  22. Rafael C. Gonzalez & Richard E. Woods "Digital Image Processing"
  23. 周宏隆, "虛擬實境中以真人帶領木偶做肢體動作之電腦視覺技術", 國立交通大學, 碩士論文, 民國 85 年
  24. 謝寧, "融入與互動式虛擬棒球遊戲中真人揮棒打擊", 國立交通大學, 碩士論文, 民國 86 年
  25. 張家銘, "三維動畫中動作重置之研究", 國立清華大學, 碩士論文,



民國 90 年

26. 黃聿文, "三維空間多攝影機視覺監控系統", 國立交通大學, 碩士論文, 民國 91 年
27. 謝耀賢, "環場全景製作技術及其在虛擬實境應用", 國立交通大學, 碩士論文, 民國 86 年
28. 張振宇, "利用人臉視覺資訊作為虛擬實境畫面瀏覽之控制", 國立交通大學, 碩士論文, 民國 88 年
29. 陳郁中, "藉由平面投射轉換執行影像嵌合及物体重建", 國立交通大學, 碩士文, 民國 89 年
30. Cynthia C. Norkin, 關節角度測量學的指導原則, 第 2 版, 翻譯社 Williams & Wilkins Asia-Pacific Ltd, 1995,

