

國立交通大學

資訊科學系

碩士論文



基於視覺的智慧型指向系統

A Vision-based Intelligent Pointer System

研究生：張柏仁

指導教授：莊仁輝 教授

中華民國九十四年六月

基於視覺的智慧型指向系統

學生：張柏仁

指導教授：莊仁輝 博士

國立交通大學

資訊科學系

摘要



由二維的影像資訊重建出三維的場景模型，在電腦視覺領域已被研究多年，這項技術若應用在指向系統上，便可以幫助我們進行場景的重建，以及計算使用者所指向的位置。目前有關於指向系統的研究，大部分都是分析手勢的影像，來計算所指的方向，但這類的方法精確度不高。本論文以一套現有的電腦視覺軟體系統，首先進行場景特徵點的三維重建，保留這個步驟所計算出來的基本矩陣，之後每次重建，不須再重算基本矩陣，直接用第一次重建時所保留的基本矩陣去重建指向物體，這個方法的優點是計算時間短且有較高的精確度。實驗方面，分別探討指向物體不同，基本矩陣不同，解析度不同以及指向方向不同的情況下，所重建出來的三維場景及指向位置，並列出幾點可能影響指向精確度的因素，以供日後改進做參考。

A Vision-based Intelligent Pointer System

Student : Po-Jen Chang

Advisor : Dr. Jen-Hui Chuang

Institute of Computer and Information Science

National Chiao Tung University

ABSTRACT

Computer vision techniques for 3D scene reconstruction from 2D images have been studied for decades. These technologies can be used to establish pointer systems to obtain the position users point at through 3D reconstruction. Most recent studies on pointer systems calculate the pointing direction and position by analyzing the images of gestures, which is not very accurate.

In this research, we develop a computer vision system which carries out the 3D reconstruction of the feature points in the scene and calculates the fundamental matrix. The obtained fundamental matrix is then used to perform the reconstruction. The advantage of this method is that only one calculation of the fundamental matrix is needed and with high accuracy.

Discussions of the accuracy of our 3D scene reconstruction and the obtained target position using different pointers, fundamental matrix, image resolution and pointing direction are given. Several possible methods that can improve the accuracy of our pointer system are also listed in this thesis.

目錄

摘要.....	II
ABSTRACT.....	III
目錄 IV	
圖目錄 VI	
表目錄 VIII	
第一章 簡介.....	1
1.1 多視角立體重建.....	2
1.2 單相機多視角重建.....	3
1.3 多視角立體重建的應用—指向系統.....	4
1.4 雷射筆指向系統.....	5
第二章 投影幾何.....	7
2.1 齊次座標表示法.....	7
2.2 階層轉換.....	8
2.2.1 投影層.....	8
2.2.2 仿射層.....	9
2.2.3 公制層.....	10
2.2.4 歐幾里德層.....	11
2.2.5 各階層的特性整理.....	11
2.3 相機模型.....	13
2.4 極線幾何與基本矩陣.....	17
2.4.1 極線幾何.....	17
2.4.2 基本矩陣.....	18
2.4.3 必要矩陣.....	19
2.5 相機校正.....	20
第三章 三維重建.....	21
3.1 三維重建之步驟.....	21
3.1.1 估算基本矩陣.....	21
3.1.2 估算投影矩陣.....	22
3.1.3 歐氏三維重建.....	24
3.2 估測相機位置.....	25
第四章 指向系統之精確度分析.....	27
4.1 使用不同的指向物體進行三維重建.....	28
4.2 使用不同的基本矩陣進行三維重建.....	31
4.3 使用不同的解析度進行三維重建.....	37

4.4 不同的指向方向進行三維重建.....	40
第五章 結論與未來工作.....	44
參考文獻.....	45



圖目錄

圖 1.1	3D Dome的示意圖(左)和實際照片(右)。	2
圖 1.2	(a)指尖的模板 (b)手指頭的模板。	4
圖 1.3	雷射筆簡報系統示意圖。	5
圖 2.1	不同幾何轉換下的立方體。	13
圖 2.2	針孔相機幾何 (a)針孔相機模型 (b)影像點的座標。	14
圖 2.3	極線幾何。	18
圖 3.1	由E 做重建後四種可能的幾何表示。	23
圖 3.2	相機座標與點座標的旋轉及平移關係。	26
圖 4.1	指向系統實驗流程圖。	28
圖 4.2	正交方塊圖板之 (a)側視影像 (b)俯視影像。	28
圖 4.3	(a)指向長方條 (b)指向雷射筆。	29
圖 4.4	使用指向長方條做指向的實驗結果 (a)重建場景的左影像 (b)重建場景的右影像 (c)利用尺的延伸去估計指向的目標點 (d)重建之後的目標點。	30
圖 4.5	使用指向雷射筆做指向的實驗結果 (a)重建場景的左影像 (b)重建場景的右影像 (c)重建場景的俯視影像 (d)重建之後的目標點。	31
圖 4.6	正交方塊圖板特徵點編號。	32
圖 4.7	(a)重建場景的左影像 (b)重建場景的右影像 (c)取八個特徵點所算出的基本矩陣之重建結果俯視圖 (d)取八個特徵點所算出的基本矩陣之目標點重建結果 (e)取十個特徵點所算出的基本矩陣之重建結果俯視圖 (f)取十個特徵點所算出的基本矩陣之目標點重建結果。	33
圖 4.8	(a)重建場景的左影像 (b)重建場景的右影像 (c)左右兩邊各取 32 個特徵點所算出的基本矩陣之重建結果俯視圖 (d)左右兩邊各取 32 個特徵點所算出的基本矩陣之目標點重建結果 (e)取左半邊 64 個特徵點所算出的基本矩陣之重建結果俯視圖 (f)取左半邊 64 個特徵點所算出的基本矩陣之目標點重建結果。	34
圖 4.9	取不同的八個特徵點去算基本矩陣所重建出來的結果。(上方的圖顯示所選取之特徵點，下方的圖為其重建結果)	35
圖 4.10	(a)重建場景的左影像 (b)重建場景的右影像 (c)取 128 個特徵點所算出的基本矩陣之重建結果俯視圖 (d)取 128 個特徵點所算出的基本矩陣之目標點重建結果 (e)取 130 個特徵點所算出的基本矩陣之重建結果俯視圖 (f)取 130 個特徵點所算出的基本矩陣之目標點重建結果。	36
圖 4.11	以人工方式加入場景的特徵點 (a)左影像 (b)右影像。	38
圖 4.12	待重建物體 (a)左影像 (b)右影像。	38

圖 4.13	使用 640 x 480 的影像做三維重建 (a)場景的左影像 (b)場景的右影像 (c)重建場景的俯視影像 (d)重建的目標點。	39
圖 4.14	使用 1280 x 960 的影像做三維重建 (a)場景的左影像 (b)場景的右影像 (c)重建場景的俯視影像 (d)重建的目標點。	40
圖 4.15	指向方向往右移動： (a)重建場景的左影像 (b)重建場景的右影像 (c)重建場景的俯視影像 (d)重建之後的目標點。	41
圖 4.16	指向方向往上移動： (a)重建場景的左影像 (b)重建場景的右影像 (c)重建場景的俯視影像 (d)重建之後的目標點。	42



表目錄

表 2.1	不同幾何轉換之比較.....	12
表 2.2	不同投影轉換層級的投影矩陣.....	12
表 4.1	使用輔助點與未使用輔助點的誤差比較表(單位:公分).....	43



第一章 簡介

由二維的影像重建出三維的室內場景，一直是電腦視覺領域一個很重要的研究主題，而根據不同的需求，便會有不同的研究重點。機器人視覺導航，便是利用兩張或者是多張的影像對所在的場景進行三維重建，進而可以得知場景中三維資訊，藉由這些資訊去判斷環境中障礙物的位置，提供給機器人路徑規畫時參考。機器人導航時並不需要詳細地重建出如相片般真實的（photo-realistic）場景，只需要重建出重要的環境結構，強調的是計算速度以及重建結果的精確度。電腦繪圖以及虛擬實境，都會利用到影像重建的技術，比如一個日常的物品，利用多個視角的照片，便可在電腦中建出它的三維模型，再輔以一些貼圖的技術，即可做出許多徐徐如生的模型，可供給動畫、電玩甚至電影來使用。在現實生活中大部分建築物的場景重建則可利用建築物常見的一些特性，比如牆面大多互相垂直，窗戶多為方形等等的特色找出許多垂直與平行的特徵，而大大地簡化重建時的計算複雜度。若將這些重建出來的模型，搭配上一個較大型的三維場景，則可以建立出一個虛擬實境的環境。在現有的指向系統中，有許多不同的實做方式，但是大部分都會利用三維重建計算出所指向的目標點，根據不同的應用目的，對於指向系統的精確度也會有不同的要求，若我們將指向系統當成是射擊遊戲中的瞄準系統，那麼對於指向系統精確度會有較高的要求，若是將指向系統應用在一般的簡報系統中，那麼對於指向系統的精確度要求就不會太高，只要所指向的目標點會隨著我們移動的方向而移動，我們就能指到想指的目標點。

本篇論文所提出的智慧型指向系統，即是利用三維重建，以取得指向物體的三維座標，進而計算出所指向的目標點。若我們一開始即先將場景的三維模型重建好，則當我們在判斷指向方向時，只需重建指向物體的兩個特徵點，不必再重建整個場景，也不需要再算一次基本矩陣，可以加速運算，簡化重建所需的步驟。在以下的幾個小節裡，我們將會介紹一些相關的三維重建的實例以及其他的指向

系統，並且簡單介紹其特色，而透視理論及影像三維重建的詳細原理，將會在下一章做說明。

1.1 多視角立體重建

多視角立體重建是從多個不同視角的位置取得影像資訊，將這些影像兩兩比較，透過一些相似度比較 (correlation-based) 的方法，找出影像中相對應的特徵，通常為點，進行三維重建。這種重建的好處是利用多台相機同時拍攝影像，在一個特定的時刻就可以取得足夠的資訊來進行重建，如此一來便可針對運動中的物體進行追蹤與軌跡重建。

Narayanan, Rander 和 Kanade [1] 實作了一個系統，他們稱為 3D Dome。如圖 1.1，3D Dome 是一個由 51 台同步且完全校正好 (fully calibrated) 的相機所結合而成的半球形環境，由於所有的相機皆是完全校正過的，於是每一台相機的投影矩陣均可以事先得到。

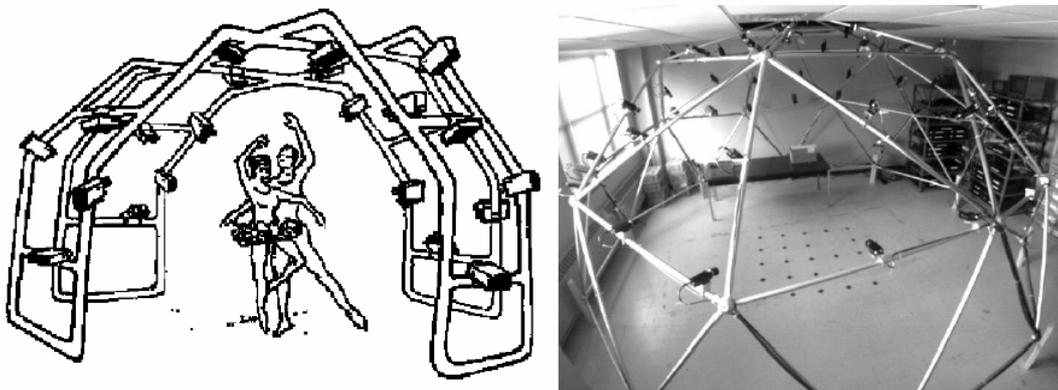


圖 1.1 3D Dome 的示意圖(左)和實際照片(右)。

當有人在 3D Dome 中心進行活動時，所有的相機會紀錄下許多不同視角的影像資訊，透過多基線立體重建 (multiple-baseline stereo reconstruction) [2] 的過程，對於每一台相機可以得到一張密集的深度影像 (dense depth graph)。接下來將原始影像上的材質資訊貼上去，便可得到一個簡單的重建模型，作者稱之為可見表面模型 (visible surface model, VSM)。但是由於 VSM 是每一台相機所

重建的結果，難免會因為重建物體本身的遮蔽關係導致有些地方會沒辦法重建出來，因此作者接下來又將許多的 VSM 結合為一，透過一些最佳化的選取及整合方式，將數個角度所重建出的場景結合成完整表面模型 (complete surface model, CSM)。最後同樣將材質貼上 CSM，得到一個完整重建的結果。

這個系統最大的特色就是相機已經完全校正好了，所以在三維重建中，只需要簡單的計算即可算出每一點的三維資訊。但是使用了高達51台相機，在應用上的彈性將會受到限制，相機間的不同步技術也將會是高難度的挑戰。

1.2 單相機多視角重建

單相機多視角重建是以單一的相機，由多個不同的視角來取得影像進行重建，這樣的重建有時候又稱為從運動中重建結構 (structure from motion)。這麼做的好處是只需要利用一台相機便可以進行重建，所花費的成本較少，其缺點是在某一個時刻只能取得一張影像。

Pollefeys 和 Van Gool [3] 便實作了這樣的一個系統，這個系統的輸入是一連串同一個場景的影像，經過了初步的特徵點擷取步驟，對每張影像分別找出一些特徵點，接下來進行相似度比較找出其對應關係。由於數位的相機在成像時一定會有些微的誤差，加上影像本身會有不穩定的雜訊，因此 Pollefeys 他們將所找出的特徵點，透過了 RANSAC (RANdom Sampling Consensus) 的方式，算出許多組可能的基本矩陣 (fundamental matrix)，再從這些基本矩陣中選出一個最穩定的。基本矩陣即代表了每兩張影像之中每個點的轉換關係，關於基本矩陣的定義與極線幾何將會在第三章中有詳細的介紹。

找出基本矩陣之後，即可進行投影重建 (projective reconstruction)。透過一些常見的相機內部參數假設，如 principle point 在影像中央、相機沒有 skew、相機的焦距 $f_x = f_y$ ，便可簡化相機的校正過程。經過校正以後，即得到完整的相機內部參數，於是重建出一個和真實場景只差一個比例常數的公制重建

(metric reconstruction) 結果。到了這個階段，每兩張影像的相機內外部參數以及投影矩陣都已經得到了，於是作者同樣地計算出密集的深度影像 (dense depth graph)，將影像中的每一個點都算出其深度資訊，最後貼上材質完成三維模型的建立。

1.3 多視角立體重建的應用—指向系統

在現有的指向系統當中，絕大多數都是利用手勢來判斷指向的方向，而用來判斷指向的兩個點，通常都是使用指尖與指根，也有使用指尖與手腕或是使用指尖與眼睛的做法。這些方法中，影響指向精確度最大的因素就是如何準確取得兩個決定指向的特徵點。

Shin Sato 和 Shigeyuki Sakane [4] 所提出的系統，即是利用指尖與指根來判斷指向的方向，而取得指尖與指根的方法，是利用一個模板 (template) 去近似手指頭的位置，如圖 1.2 (a) 即為指尖的模板，先在兩張影像中找指尖，找到指尖後，以圖 1.2 (b) 手指頭的模板去找指根的位置，找到指尖與指根之後就可以決定指向的方向，再利用三維重建去算出指向的位置。在這個方法中，找指尖只用了九個模板，只有九個方向，但是手指頭能指出的方向卻是無限多個，所以這個方法只能簡略地近似我們所要指向的方向。

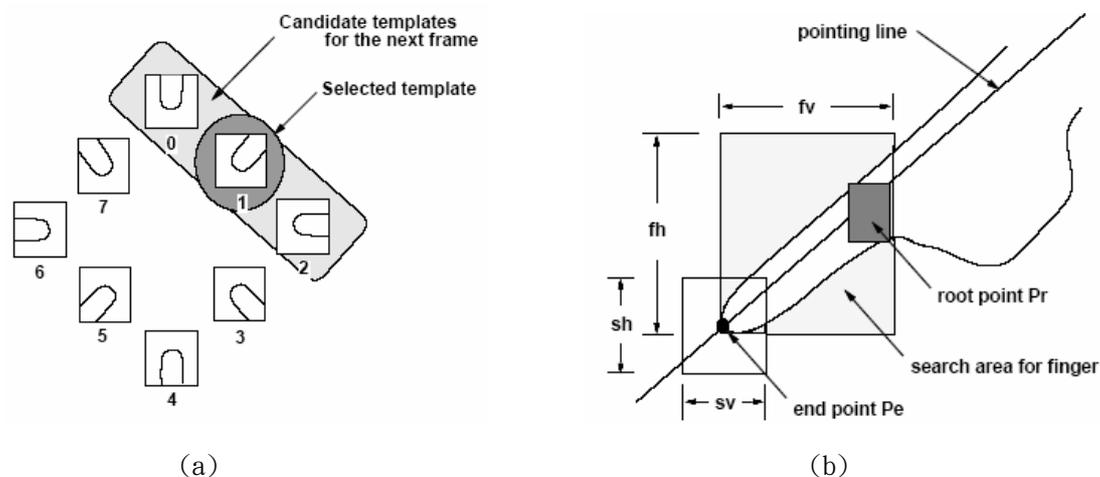


圖 1.2 (a)指尖的模板 (b)手指頭的模板。

而在 Yi-Ping Hung, Yao-Strong Yang, Yong-Sheng Chen, Ing-Bor Hsieh, and Chiou-Shann Fuh [5] 所提出的系統中，他們使用指尖與眼睛當成兩個指向點，在使用上會有所限制，因為影像中必須包含人的眼睛，所以人臉的朝向必須限制在某個範圍，使用上會有所不便。

1.4 雷射筆指向系統

在一般的會議中，最常使用雷射筆來當作我們的指向工具，用雷射筆所打出的光點資訊，如果可以輸入到電腦中，則可以做一些其他的應用，不再只有指向的功能，我們可經由雷射筆的使用對電腦做一些輸入指令的功能，使雷射筆成為使用者與電腦之間溝通的媒介。

Rahul Sukthakar, Robert G. Stockton and Matthew D. Mullin [6] 所提出的系統，是一個雷射筆簡報系統，它必須先經過系統校正，取得相機平面、投影平面以及顯示平面之間對應的關係，然後用相機取得光點的影像，來判斷指向的位置並且輸入到電腦中，當光點停留在特殊的區域時，代表某些特定的指令輸入（例如：繪圖、上一頁、下一頁）。圖1.3 為雷射筆簡報系統的示意圖。

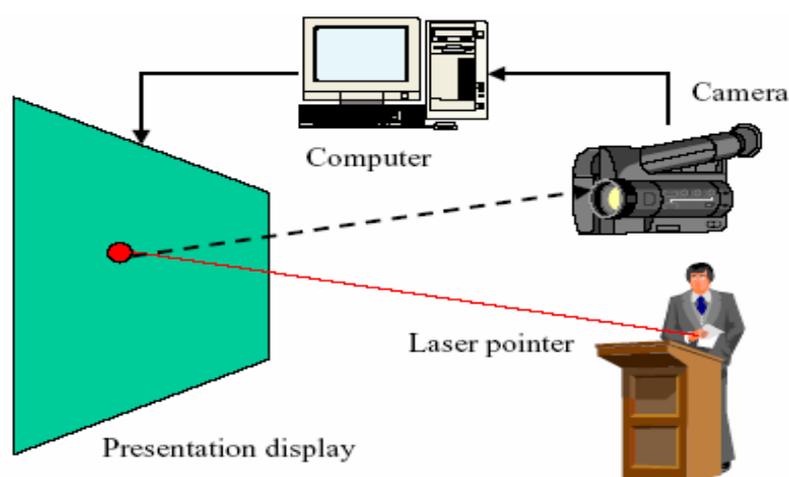


圖1.3 雷射筆簡報系統示意圖。

這個系統是屬於 2D to 2D 的對應，如果指向的範圍超過顯示平面，就無法判斷出指向的位置，況且雷射筆所打出來的紅光，如果射到人的眼睛上，會對眼睛造成傷害，有潛在性的危險，如果打在會反光的表面上，紅光的強度會變弱，可見性會降低，這些因素都使我們在應用上會有所限制。

本篇論文全文共分為五章，除了本章為簡介外，第二章介紹投影幾何、相機模型、極線幾何、基本矩陣以及必要矩陣。第三章介紹三維重建的步驟以及相機位置之估算。第四章介紹指向系統在不同的條件下做三維重建，並分析其指向精確度。第五章為結論並說明本研究接下來的一些方向。



第二章 投影幾何

本章將介紹與我們所採用電腦視覺系統有關的觀念。我們會依序介紹齊次座標表示法、階層轉換、相機模型、極線幾何與基本矩陣，最後說明此系統所使用的相機校正方法。由於此方面研究已有相當長遠的一段時間，在這裡僅選出與此系統較相關的部分，若想要深入探討本章沒有提到的部分，請參考電腦視覺相關文獻[7][8][9]。

2.1 齊次座標表示法

在平面上的一條直線可以用等式 $ax+by+c=0$ 來表示，而不同的 a 、 b 、 c 參數便代表了不同的直線，因此我們以向量的形式 $(a,b,c)^T$ 來代表平面上的一條直線。如果一個點 $x=(x_0,y_0)^T$ 在一條直線 $l=(a,b,c)^T$ 上，則將點座標 (x_0,y_0) 代入直線方程式，便可以得到 $ax_0+by_0+c=0$ 。為了能使用向量內積來表達這個式子，我們可將 x 座標重新寫成 $x=(x_0,y_0,1)^T$ 的形式，則：

$$ax_0+by_0+c=(x_0,y_0,1)(a,b,c)^T=x^Tl=0$$

其中 $(x_0,y_0,1)^T$ 和 $(a,b,c)^T$ 就是平面上點 x 和直線 l 的齊次座標表示法。

$ax_0+by_0+c=0$ 和 $kax_0+kby_0+kc=0$ 代表的是同一條直線，因此若在 $k \neq 0$ 的情況下，由向量形式表示的 $(a,b,c)^T$ 和 $(ka,kb,kc)^T$ ，代表相同的一條直線。在同樣的情形之下， $ka(x_0)+kb(y_0)+kc=0$ 和 $a(kx_0)+b(ky_0)+c(k)=0$ 的意義是相同的，因此在 $k \neq 0$ 的情況下， $(x_0,y_0,1)^T$ 和 $(kx_0,ky_0,k)^T$ 表示的是同一個點。也就是說，在 $x_3 \neq 0$ 的情況下， $(x_1,x_2,x_3)^T$ 可以對應到我們熟悉的歐幾里德平面上的 $(x_1/x_3,x_2/x_3)$ 這一點。另外，在齊次座標表示法中， $(0,0,0)^T$ 是沒有定義的。

2.2 階層轉換

在我們現實的生活環境中，所有我們看見的物體都可以用歐幾里德三度空間來表示，但是有時候我們不需要用到完整三度空間的歐幾里德座標轉換，而只需要針對投影幾何中比較簡單且限制較少的轉換來處理。我們可以把這些轉換看成是一個階層性的關係[10][11]。從最簡單的投影（projective）層，接下來是仿射（affine）層，公制（metric）層，最後是歐幾里德（Euclidean）層。這些階層性的觀念，彼此在幾何上有著一些轉換的關係，並且有一些不變性存在。和投影層有關的是投影轉換群，和仿射層有關的是仿射轉換群，和公制層有關的是相似群（similarities），最後和歐幾里德層有關的是歐幾里德轉換群。很重要的一點是，不同的轉換群之間有著父子群的關係，比方說相似群是仿射群的子群，而他們兩者都是投影群的子群。有關這些轉換群間的詳細介紹，請參考[8]。

在每一種轉換群中，有著一個很重要的特性，那就是不變性，是一種在某一個特定群中，不管經過什麼轉換，都依然保持不變的特性。不同的幾何轉換對應著不同的不變性。這些不變性通常和幾何實體有關，若能恢復並得知其值，便可將幾何轉換做些改變，通常是用來提昇成較高階層的轉換。例如，若我們能夠得知幾何實體的角度和相對長度的資訊，就可以將仿射結構提升到相似轉換。在本節中，我們將會個別對每一層做簡單的介紹，並且以圖例來表示在各個轉換層級下正方體可能的形變，並以表格表示各種轉換的不變性。

2.2.1 投影層

我們首先介紹投影層（projective stratum），它是所有層級中限制最少的一層，因此含有最少的不變性，而投影轉換（projective transformation）也是最普通的線性轉換。

三維空間的點轉換若滿足 $\rho y = T_p x$ ，則可稱為投影轉換。其中， ρ 為一個常數， x 、 y 分別為轉換前和轉換後空間點的齊次座標， T_p 為 4×4 的可逆矩陣：

$$T_p = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix} \quad (2.1)$$

由於這個轉換矩陣可以有一個比例常數，因此共有15個自由度。

在投影轉換群裡，除了共線的不變性質外，交比（cross-ratio）也是不會隨著投影轉換而改變的。假設有四個共線的點 M_1, M_2, M_3 和 M_4 ，則可將這四個點表示成 $M_i = M + \lambda_i M'$ ，其中 λ_i 為純量， M' 為該直線的方向向量，則交比便可定義為：

$$R(M_1, M_2, M_3, M_4) = \frac{(\lambda_1 - \lambda_3)(\lambda_2 - \lambda_4)}{(\lambda_2 - \lambda_3)(\lambda_1 - \lambda_4)} \quad (2.2)$$

由式(2.2)可知，交比和 M 或 M' 的選擇無關，只與共線之四點所形成的各線段之相對長度有關。



2.2.2 仿射層

仿射層（affine stratum）和投影層的不同點是，它建立在一個特別的參考平面上，這個平面我們稱之為無窮遠處平面（the plane at infinity）。若我們於投影空間 P^3 中選擇一平面，而在不失一般性的情況下，我們可令其齊次方程式為 $x_4 = 0$ ，則此平面便是無窮遠處的平面。

在 $X = [X \ Y \ Z]^T \rightarrow [X \ Y \ Z \ 1]^T$ 的對應關係下，我們可以將投影空間看成是包含了仿射空間。對於每個不在無窮遠處平面 $x_4 = 0$ 上的點，假設其空間座標為 (x_1, x_2, x_3, x_4) ，我們可以得到 $X = x_1/x_4$ ， $Y = x_2/x_4$ ， $Z = x_3/x_4$ 。

仿射轉換可以表示成：

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} a_{14} \\ a_{24} \\ a_{34} \end{bmatrix} \quad \text{with } \det(a_{ij}) \neq 0, 1 \leq i, j \leq 3 \quad (2.3)$$

若寫成齊次座標系則可改寫為 $M' = T_A M$ ，其中

$$T_A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \quad (2.4)$$

因此仿射層含有12個自由度，而在仿射的情況之下，平行性（parallism）是不變的。

2.2.3 公制層

公制層（metric stratum）相較於仿射層，多了角度與相對長度的限制，而其與歐幾里德層（Euclidean stratum）的轉換，其間只差了一個常數倍的比例關係。當我們無法得知物體實際尺寸大小時，我們便只能從影像中得到具有最多限制的這種幾何結構。公制轉換可表示如下：

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \sigma \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} t_{14} \\ t_{24} \\ t_{34} \end{bmatrix} \quad (2.5)$$

其中， $[X \ Y \ Z]^T$ 和 $[X' \ Y' \ Z']^T$ 分別代表轉換前後的三維點座標， $R = [r_{ij}]$ 是正交矩陣，它會滿足旋轉矩陣 $RR^T = I$ 且 $\det(R) = 1$ 的關係。將三維座標點以齊次座標表示，可再改寫成 $M' = T_M M$ ，其中

$$T_M = \begin{bmatrix} \sigma r_{11} & \sigma r_{12} & \sigma r_{13} & t_x \\ \sigma r_{21} & \sigma r_{22} & \sigma r_{23} & t_y \\ \sigma r_{31} & \sigma r_{32} & \sigma r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

因為正交矩陣只有3個自由度，再加上3個平移及1個常數，所以共有7個自由度。在相似轉換的情形下，有兩個重要的不變性就是相對長度及角度。若可以得到相對長度和角度的資訊，我們便可將仿射轉換提昇為相似轉換。

2.2.4 歐幾里德層

歐幾里德層 (Euclidean stratum)。它和公制層差不多，只不過把比例常數給固定下來，而使得相對長度成為絕對長度。歐幾里德轉換有6個自由度，包括了3個方位及3個平移。一個歐幾里德轉換可表示為：

$$T_E = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

其中 r_{ij} 是正交矩陣的元素。

2.2.5 各階層的特性整理

表2.1列出不同幾何轉換的特性，其中包含每個轉換的自由度、轉換矩陣及不變性。表2.2則列出了由三度空間投影到二度空間的投影矩陣[8]。另外，圖2.1則顯示出對於不同的幾何轉換下，立方體所可能呈現的樣子。由圖2.1可知，至少要到正規轉換以上的階層，才比較像立方體原來的樣子。

表2.1 不同幾何轉換之比較

幾何轉換	自由度	轉換矩陣	不變性
投影(projective)	15	$T_p = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix}$	交比 共線
仿射(affine)	12	$T_A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$	平行 共線、交比
公制(metric)	7	$T_M = \begin{bmatrix} \sigma r_{11} & \sigma r_{12} & \sigma r_{13} & t_x \\ \sigma r_{21} & \sigma r_{22} & \sigma r_{23} & t_y \\ \sigma r_{31} & \sigma r_{32} & \sigma r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$	角度、相對長度 平行 共線、交比
歐幾里德(Euclidean)	6	$T_E = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$	角度、絕對長度 平行 共線、交比

表2.2 不同投影轉換層級的投影矩陣

投影層級	投影矩陣	說明
投影(projective)	P	P:從三度空間投影到影像平面
仿射(affine)	$[Pp]$	P:無限遠平面和影像平面之間的 homography p:世界座標系原點的投影
歐幾里德(Euclidean)	$A[Rt]$	A:相機內部參數 (R, t):在世界座標中的相機位置

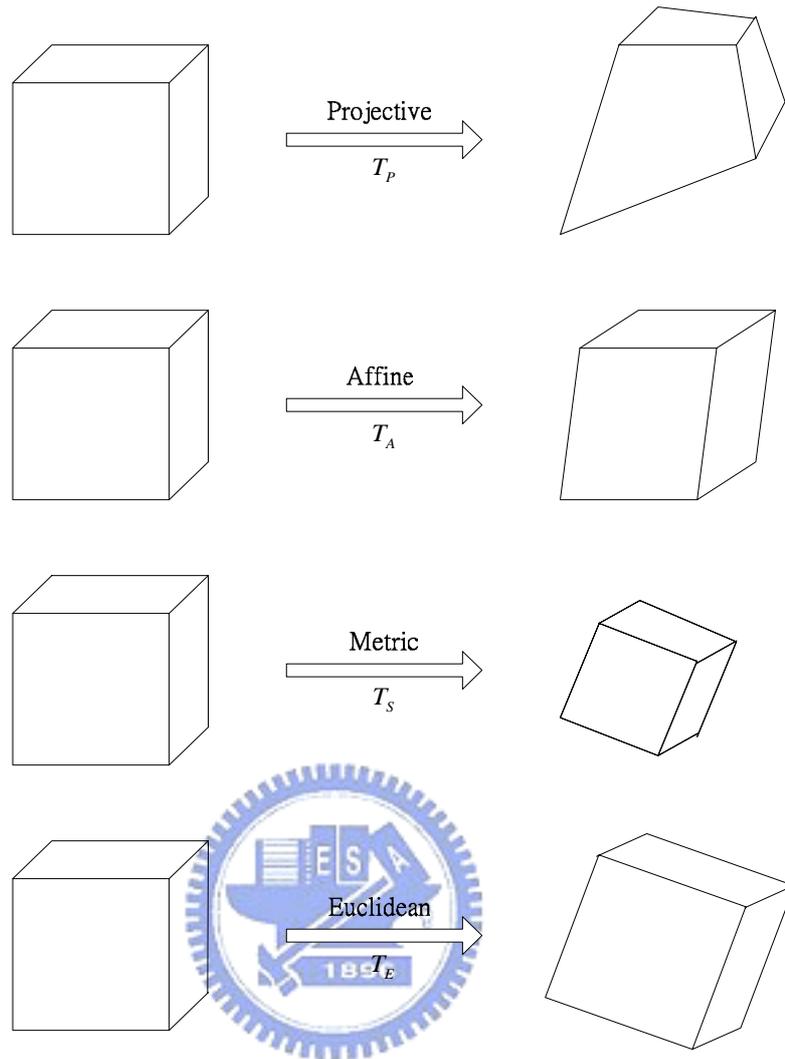


圖2.1 不同幾何轉換下的立方體。

2.3 相機模型

在本節中，我們將會透過數學算式的推導，介紹本系統所使用的一般化相機模型[7]。考慮用中心投影的方法，將一個空間中的點投影到一個平面，我們讓投影中心位在歐幾里德座標系的原點，並且讓成像的平面是 $Z = f$ ，稱為影像平面。考慮一個針孔相機的模型如圖2.2(a)，圖中 C 是相機光學中心、 P 是 principal point、 f 是焦距 (focal length)、 P 和 C 所形成的直線稱為 principal axis、通過 P 和 Z 軸垂直的平面就是影像平面。空間中的點 $X = (X, Y, Z)^T$ 在影像平面上的對應點，即 X 和相機中心 C 所形成的直線和影像平面的交點 x 。用相似

三角的幾何關係，參考圖2.2(b)，我們可以知道 $x = (fX/Z, fY/Z, f)^T$ 。如果不考慮最後一個座標，針孔相機可以把歐幾里德三維空間的點對應到歐幾里德二維空間的點：

$$(X, Y, Z)^T \mapsto (fX/Z, fY/Z)^T \quad (3.8)$$

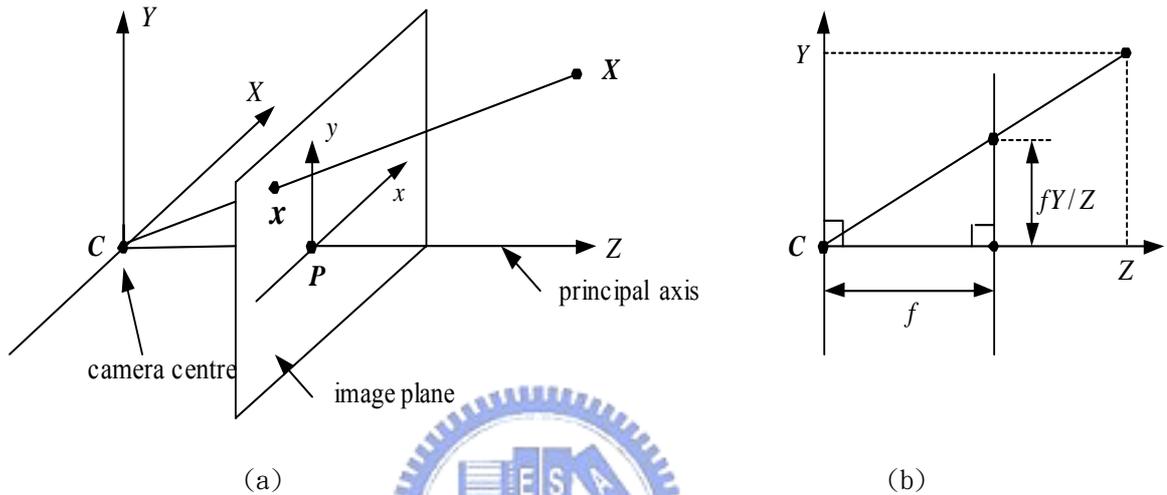


圖2.2 針孔相機幾何 (a)針孔相機模型 (b)影像點的座標。

如果將空間中的點和影像上的點用齊次座標來表示，式(2.8)可以寫成以下的線性方程式：

$$x = \begin{bmatrix} fX \\ fY \\ Z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = P X \quad (2.9)$$

其中， X 表示三維空間中的點的齊次座標， x 表示影像平面上的點的齊次座標， P 是一個 3×4 的矩陣，稱為投影矩陣。

在式(2.9)中，我們假設影像平面的原點剛好就是 principal point，但是實際上可能不是如此。若 $(p_x, p_y)^T$ 是影像上 principal point 的座標，則其歐幾里德對應關係為：

$$(X, Y, Z)^T \mapsto (fX/Z + p_x, fY/Z + p_y)^T \quad (2.10)$$

上述的對應我們可以用齊次座標表示如下：

$$x = \begin{bmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{bmatrix} = \begin{bmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = P X \quad (2.11)$$

若將相機內部參數定義為：

$$A = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2.12)$$

那麼，式子(2.11)變可以寫成：

$$x = A[I_3 | 0]X_{cam} \quad (2.13)$$

在式(2.13)中，我們把 $(X, Y, Z, 1)^T$ 寫成 X_{cam} ，主要是要強調這個座標是屬於相機座標系統。然而，在一般情況下，三維空間中的點是以世界座標系統表示，在世界座標系統和相機座標系統之間存在著旋轉和平移的關係。

假設 \tilde{X} 代表三維空間中一個點的歐幾里德座標， \tilde{X}_{cam} 代表在相機座標系統中同一個點的座標， \tilde{C} 代表相機中心在世界座標系統的位置， R 是一個 3×3 的旋轉矩陣，代表著相機座標系統的方向。那麼， \tilde{X} 和 \tilde{X}_{cam} 之間的關係可以表示成 $\tilde{X}_{cam} = R(\tilde{X} - \tilde{C})$ ，這個方程式可以齊次座標表示如下：

$$\tilde{X}_{cam} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \tilde{X} \quad (2.14)$$

將式(2.14)代入式(2.13)可以得到：

$$x = AR[I | -\tilde{C}]X \quad (2.15)$$

其中 X 已經是相對於實際的世界座標系統，而式(2.15)已經將理想的針孔相機模型推進為較具一般性的針孔相機模型了。

我們可以看出在一般性針孔相機模型中，相機投影矩陣 $P = AR[I | -\tilde{C}]$ 有9個自由度： A 有 f 、 p_x 、 p_y 共3個自由度， R 有3個自由度， \tilde{C} 有3個自由度。 A 之中的參數稱為相機內部參數， R 和 \tilde{C} 分別與相機在世界座標系統中的方向和位置有關，稱為外部參數。通常我們用 t 來表示 $-R\tilde{C}$ ，所以投影矩陣可以改寫成 $P = A[R | t]$ ，而式(2.15)的針孔模型可以替換如下：

$$x = A[R | t]X \quad (2.16)$$

上面所述的針孔模型是假設影像平面在 x 軸和 y 軸方向的比例是相等的。但是如果是用數位的 CCD 相機，它的影像平面上 x 軸和 y 軸方向的比例就有可能不是相等，也就是說經由 CCD 相機所得到的影像的每一個像素(pixel)可能不是正方形。我們假設在影像座標中 x 軸和 y 軸方向的單位長度內分別有 m_x 和 m_y 個像素。那麼，在式(2.12)的左邊乘上一個對角矩陣 $diag(m_x, m_y, 1)$ ，就可以把世界座標的點轉換到影像座標的點。因此，CCD 相機的內部校正矩陣就會變成：

$$A = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.17)$$

其中， $f_x = fm_x$ 、 $f_y = fm_y$ 、 $u_0 = p_x m_x$ 、 $v_0 = p_y m_y$ 。 f_x 和 f_y 代表相機在 x 軸和 y 軸方向上的焦距，單位為像素。 (u_0, v_0) 代表影像上用像素為單位的 principal point。因此，一個相機投影矩陣有10個自由度。

通常，更具一般性的相機內部參數矩陣還有一個參數 skew (γ)，這個參數在大部分的相機中的值都是0，只有在某些狀況下不為0，將它加入內部參數矩陣中，可得到式(2.18)：

$$A = \begin{bmatrix} f_x & \gamma & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

那麼，一個一般性的相機投影矩陣就有11個自由度， A 便成為一個完整的相機內部參數矩陣。

假設二維的點座標以 $m = [u \ v]^T$ 來表示，而一個三維的點座標以 $M = [X \ Y \ Z]^T$ 來表示。我們令他們的齊次座標表示法分別為 $\tilde{m} = [u \ v \ 1]^T$ 和 $\tilde{M} = [X \ Y \ Z \ 1]^T$ ，一部相機的則用針孔相機的模型來表示，那麼，一個三維的點 M 和該點投影在影像上的二維點座標 m 就有以下的關係存在：

$$s\tilde{m} = \begin{bmatrix} f_x & \gamma & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} [R | t] \tilde{M} = A [R | t] \tilde{M} \quad (2.19)$$

其中 s 是一個比例常數， $[R | t]$ 是由世界座標系統轉換到相機座標系統的旋轉矩陣和平移矩陣， A 則是相機的內部參數矩陣。

2.4 極線幾何與基本矩陣

2.4.1 極線幾何

在相同三維場景中，所取得兩張不同視角的影像之間，存在一個很重要的限制關係，稱為極線限制（epipolar constraint）。我們以圖2.3來說明這個限制。假設 M 是三維空間中的一個點，則它在第一張影像所成像的影像點是 m ，在第二張影像的影像點是 m' 。 C 和 C' 分別是第一個和第二個相機的光學中心。如圖所示，三維空間的點 M 和影像點 m 、 m' 以及相機的光學中心是共平面的，我們將這個平面稱為極線平面（epipolar plane） π 。 C 和 C' 的連線分別與第一和第二個影像平面相交於 e 和 e' ，我們稱這兩個點為極點（epipole）。

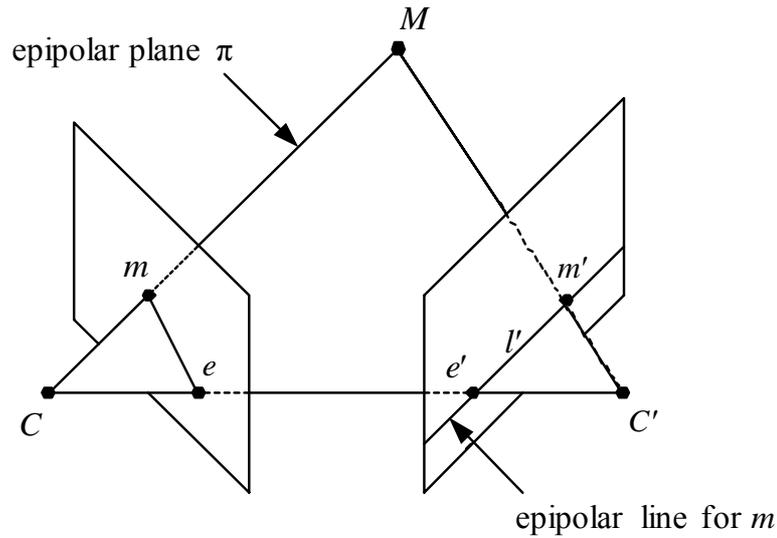


圖2.3 極線幾何。

根據極線幾何，可以發現一個特性，第一張影像上的任何一點和第一張影像的相機光學中心所形成的射線，投影在第二個影像平面上，將成像為一條線(極點除外)，這種線就稱做極線 (epipolar line)。如圖2.3中的線 l' ，就是 m 在第二張影像上的極線。不同的三維空間點對於 C 和 C' 會產生許多不同的極線平面，而這些極線平面將會交於一條線，也就是相機光學中心 C 和 C' 的連線，稱為基線 (baseline)。從影像平面來看，也就表示所有的極線都會通過極點。

極線限制對於影像特徵點的對應有著極大的輔助功能，可以將搜尋的範圍有效地從整張影像減少到極線附近。另外極線皆會交於極點這個性質，也可以用來當作一個誤差評量的標準。

2.4.2 基本矩陣

基本矩陣 (fundamental matrix) 是用來描述在極線幾何中，任一點與其極線之間的對應關係。由圖2.3中我們可以看出，在第一張影像中的任一點 m ，在第二張影像會有一條對應的極線 l' ，在第二張影像中和 m 點相對應的 m' 點一定會位在這條極線 l' 上。 m 點和極線 l' 透過基本矩陣 F ，以齊次座標表示

會有一個線性的關係：

$$l' = F m \quad (2.20)$$

因為 m 在第二張影像上的對應點 m' 會位在 l' 這條線上，所以 m' 的座標和 l' 的向量內積為零：

$$m'^T l' = m'^T F m = 0 \quad (2.21)$$

我們將兩張影像的角色互換，則基本矩陣會轉變為原來矩陣的轉置矩陣，也就是可以將上式轉換為：

$$m^T F^T m' = 0 \quad (2.22)$$

當第一張影像中的點是極點 e 的時候是一個特例，它所對應的極線退化成第二張影像的極點 e' 。所以：

$$l' = F e = 0 \quad (2.23)$$

除此之外，基本矩陣可寫成一個和相機內、外部參數有關的形式：

$$F \cong A'^T [t]_{\times} R A^{-1} = [A' t]_{\times} A' R A^{-1} \quad (2.24)$$

其中， $[t]_{\times}$ 為一個內含平移向量 $t = [t_1 \ t_2 \ t_3]^T$ 的斜對稱矩陣，也就是：

$$[t]_{\times} = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix} \quad (2.25)$$

3.4.3 必要矩陣

正規化影像座標的特別化基本矩陣即為必要矩陣 (essential matrix)，也可以看成是內含了相機內部參數的基本矩陣。

已知相機投影矩陣為 $P = A[R | t]$ ，且 $m = PM$ ， m 為影像上的點， M 為真實的三維空間點。若相機內部參數矩陣 A 已知，則我們可以在 m 左邊乘上 A 的反矩陣得到 $\tilde{m} = A^{-1}m$ ，則 $\tilde{m} = [R | t]M$ ， \tilde{m} 即為以正規化座標表示之影像點，可視為點 M 使用了具有單位校正矩陣 I 相機投影在影像上的二維點，投

影矩陣 $A^{-1}P = [R | t]$ 稱為正規化投影矩陣。

必要矩陣的定義如下：

$$\tilde{m}'^T E \tilde{m} = 0 \quad (2.26)$$

\tilde{m} 和 \tilde{m}' 分別代表相對於點 m 和 m' 之正規化影像座標。將 \tilde{m} 和 \tilde{m}' 分別用 m 和 m' 取代，則式(2.26)可改寫為 $m'^T A'^{-T} E A^{-1} m = 0$ 。與式(2.21)做比較，則可知必要矩陣與基本矩陣間的關係為：

$$E = A'^T F A \quad (2.27)$$

將上式代入(2.24)，得到：

$$E = [t]_x R \quad (2.28)$$

到這裡我們可以發現，當我們需要重建出相機的投影矩陣時，只要能算出基本矩陣，便能還原出旋轉矩陣 R 和平移向量 t ，詳細的重建過程我們將會在下一章中做詳細的介紹。



2.5 相機校正

在完全不知道場景的結構以及相機的資訊的情況下重建三維場景，所有的資訊都將隱含在基本矩陣中。然而，若我們限制在拍攝影像的過程中，不改變相機本身內部設定，如焦距，則相機內部參數並不會改變，因此我們便可以透過事先校正 (calibration) 的方式先算出這些參數，以降低重建時的複雜度。

從式(2.18)中我們可以知到，一般相機的內部參數有 f_x 、 f_y 、 γ 、 u_0 和 v_0 這五個， γ 、 u_0 和 v_0 這三個參數較為固定，而且可以有一些假設值比如 $\gamma = 0$ ， u_0 和 v_0 分別為影像中心等，但是相機的焦距 f_x 、 f_y 卻對於三維重建有很大的影響，因此相機校正也是近年來電腦視覺的一個重要的課題。

在本系統中所使用的相機，是一個事先校正好的相機。這個校正的方法是由 Zhang [12]所提出的，利用平面的 homography 性質來計算相機的內部參數，並且在計算出初始參數之後再進行一些微調，以達到最佳化。

第三章 三維重建

有了投影幾何的理論基礎，本章將會介紹三維重建的步驟，除了重建出三維場景之外，也會討論如何利用必要矩陣找出相機的旋轉矩陣及平移向量，進而還原出相機的三維位置資訊。

3.1 三維重建之步驟

在具備了投影幾何的基本觀念之後，本節將介紹由兩張影像重建場景的方法，其步驟包括：

1. 由對應點計算基本矩陣，見3.1.1 節。
2. 由基本矩陣計算投影矩陣，見3.1.2 節。
3. 對每組對應點，計算其三維空間之點座標，見3.1.3 節。

以下就針對各個步驟進行詳細的說明。



3.1.1 估算基本矩陣

三維重建的第一個步驟，是要估算出兩張影像間的基本矩陣。由式(2.22)中我們可以得知，在兩張影像上任意的對應點 m 和 m' 必定能滿足 $m^T F^T m' = 0$ 這個條件。假設兩張影像中的特徵點 m_i 和其對應點 m_i' 已知，則根據式(2.22)，每一組對應點可以產生一個線性方程式，由於基本矩陣 F 扣除了一個縮放的比例常數之後，會剩下八個未知數，因此只需要八組對應點，即可以線性方法解出一個允許改變大小比例的基本矩陣，這個方法稱為八點演算法 (8-point algorithm) [13]。

考慮到數位相機在成像時可能的取樣誤差，再加上影像本身會有不穩定的雜訊，甚至在特徵點的擷取與對應上發生錯誤，使得只利用八個點來計算基本矩陣，並不保證能得到正確的結果，因此有很多方法也被提出來改進基本矩陣。比

如使用超過八組對應點計算出多組基本矩陣，再透過最小平方 (least-square) 法來求出最佳的解，以增加基本矩陣的可靠度。有關基本矩陣的最佳化方法比較，可以參考[14][15][16]。

3.1.2 估算投影矩陣

利用前述之方法得到基本矩陣之後，第二個步驟我們要算出兩台相機各自的投影矩陣。由於本論文中，相機是已經校正完畢的，因此可以看成兩台相機的內部參數矩陣 A 和 A' 皆已知，加上在第一步中算出的基本矩陣 F ，透過式 (2.27)，我們便可以算出必要矩陣 E 。必要矩陣有以下的兩個性質：

1. 3×3 矩陣 $E = [t]_x R$ 為一必要矩陣若且唯若 E 的兩個奇異值相等，第三個為零[17]。因為必要矩陣 E 可分解為 $[t]_x R$ ，其中 $[t]_x$ 為斜對稱矩陣，考慮

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{且} \quad Z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.1)$$

則 3×3 斜對稱矩陣 $[t]_x$ 可分解為 $[t]_x = kUZU^T$ ， U 為正交矩陣，而 $Z = \text{diag}(1,1,0)W$ ，此即具有兩個相等奇異值之 E 的奇異值分解(SVD)。

2. 假設 E 的 SVD 為 $E = U \text{diag}(1,1,0)V^T$ ，則使用上述的表示法， $E = [t]_x R$ 有以下兩種可能的分解方式：

$$[t]_x = UZU^T, \quad R = UWV^T \quad \text{或} \quad R = UW^T V^T \quad (3.2)$$

上式中的 R 即為相機的旋轉矩陣，而平移向量 t 則包含在 $[t]_x$ 裡。由於 $[t]_x$ 的富氏長度為2，表示若 $[t]_x$ 包含比例，則 $\|t\|=1$ 。因為 $[t]_x t = 0$ ，可知 $t = U(0,0,1)^T = u_3$ ，也就是 U 的最後一行。

由以上兩個性質，我們可以找出可能的 R 和 t ，由於 R 有兩種可能性，而 t 因為沒有辦法決定其正負號，所以也會有兩種可能性，因此第二部相機相對於第一部相機的外部參數，便會有四種可能的選擇。也就是說，給一個必要矩陣 $E = Udiag(1,1,0)^T$ ，第一部相機的投影矩陣為 $P = [I | 0]$ ，則第二部相機的投影矩陣 P' 會有四種可能的選擇，分別為：

$$P' = [UWV^T | u_3] \text{ 或 } [UWV^T | -u_3] \text{ 或 } [UW^TV^T | u_3] \text{ 或 } [UW^TV^T | -u_3] \quad (3.3)$$

圖3.1說明由 E 做重建後四種可能的幾何表示， A 代表第一部相機中心， B 和 B' 代表第二部相機可能的相機中心， $M_1 \sim M_4$ 即為重建出之三維點。由於重建後的點只有可能會在兩部相機的前方，因此只要測試一個點是否落在相機前方，就可以從四種可能的相機位置中決定第二部相機的位置。

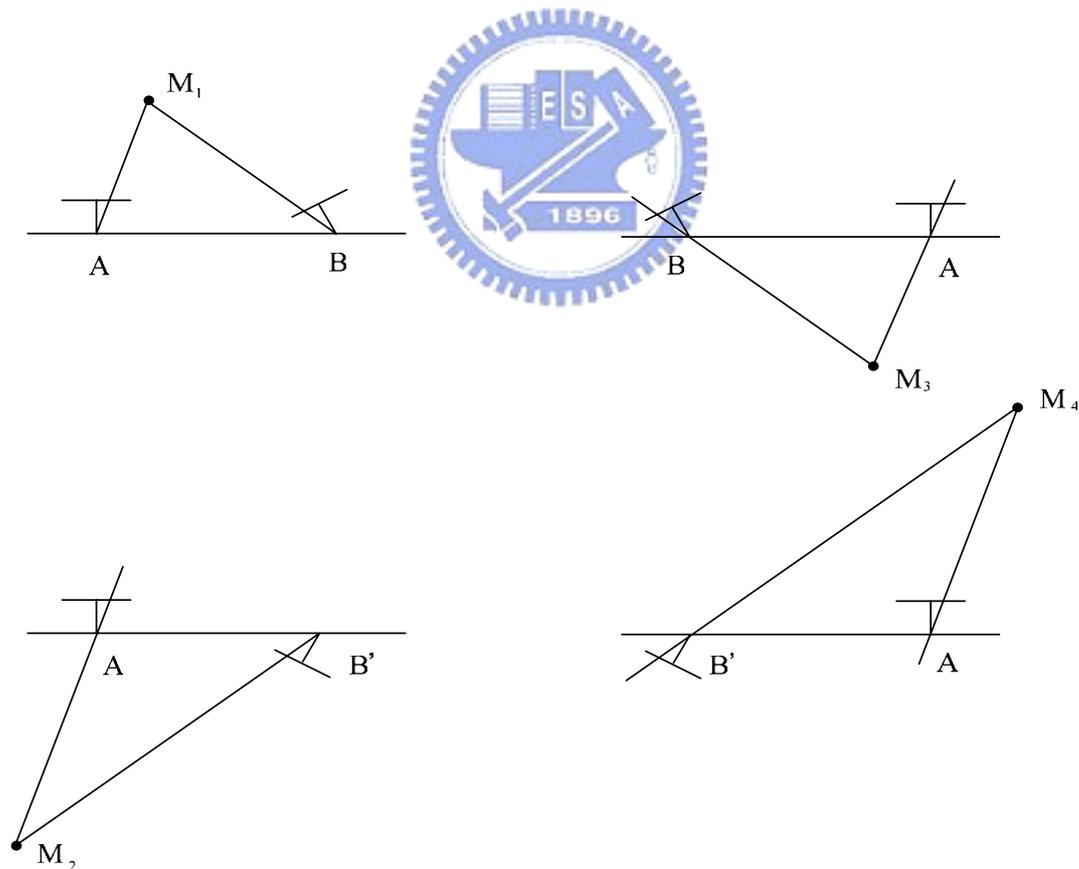


圖3.1 由 E 做重建後四種可能的幾何表示。

3.1.3 歐氏三維重建

在取得兩部相機的投影矩陣 P 和 P' 之後，接下來便可進行點的歐式三維重建 (Euclidean reconstruction)。首先將空間中的三維點 M 以及其在影像中對應的特徵點 m_1 、 m_2 寫成以下的關係式：

$$m_i = s_i P_i M, \quad i \in \{1, 2\} \quad (3.4)$$

其中 s_i 為一個代表大小的比例常數 (scale)。若我們將式(3.4)裡之影像點改寫成齊次座標系，並將投影矩陣以列向量表示，可得到：

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} p_1^T \\ p_2^T \\ p_3^T \end{bmatrix}_{3 \times 4} M_{4 \times 1} \quad (3.5)$$

由式(3.5)可推導出：

$$\begin{bmatrix} up_3^T - wp_1^T \\ vp_3^T - wp_2^T \end{bmatrix}_{2 \times 4} M_{4 \times 1} = 0 \quad (3.6)$$

令 $B_i = \begin{bmatrix} up_3^T - wp_1^T \\ vp_3^T - wp_2^T \end{bmatrix}_{2 \times 4}$ ，則對於每一台相機，皆有 $B_i M = 0$ 的關係存在。

每一個三維點 M ，皆有一組影像對應點，就可以獲得兩個如式(3.6)的關係式，將兩式合而為一，便為一滿足 $BX=0$ 的 4×4 矩陣 B 。欲求得三維點座標 M ，理論上可利用特徵值分解之對應特徵值 (eigenvalue) 為零的特徵向量 (eigenvector)，可是當有雜訊時，則需由對應於最小特徵值的特徵向量加以求得，求得的方法通常是利用矩陣的 SVD 分解，來得到三維點 M 。

經過了這些步驟，便可重建出特徵點的三維座標，到了這個階段，我們得到的是一個與歐式重建具有縮放比例關係的公制重建 (metric reconstruction)。接著，利用場景中任何一個已知的尺寸，便可以將這個比例固定下來，得到真正的歐式重建，也就是與實物等尺寸的三維重建。

3.2 估測相機位置

在電腦視覺中，姿勢估測（pose estimation）便是在估測相機本身的位置，歐式重建的過程裡，我們可以經從基本矩陣中直接抽取出兩台相機間的旋轉矩陣及平移向量，因此我們可以很直接地估算出每台相機相對於重建場景的位置。

我們在式(3.3)中提到，旋轉矩陣及平移向量的組合共有四種可能，而選擇的方法為判斷重建結果是否在兩台相機的前方。但是實際上在計算的時候，由於使用了SVD 分解法，因此所分解出來的旋轉矩陣和平移向量，對於正負符號的決定會有著不確定性，導致相機位置無法求出，也無法檢查場景是否在兩台相機前方。

為了解決這個問題，我們可以利用一些額外的限制，來找出正確的正負號。對於旋轉矩陣，我們可以利用旋轉矩陣一個重要的特性，也就是行列式（determinant）等於1，來決定出它的正負號。而平移向量則可從已知的相機位



移方向決定，比如事先知道相機是向右移動，則可知 x 方向的平移為正。確定了旋轉矩陣及平移向量的正負號之後，我們也就可以檢查四種可能的三維重建結果，哪一個是在兩台相機的前方了。

找出相機座標系間的旋轉及平移之後，也就代表著每一組三維重建的兩台相機相對位置已經找到，只要利用這些資訊，便可以重建出所有相機的位置。令 C_1 為第一台相機的相機中心， R_1 及 t_1 為第一台相機座標系與世界座標系間的旋轉和平移； C_2 為第二台相機的相機中心， R_2 及 t_2 為第二台相機座標系與世界座標系間的旋轉和平移，則我們可以求得兩台相機之間，座標系的相對旋轉及平移 R_{12} 及 t_{12} 分別為：

$$R_{12} = R_2 R_1^T \quad (3.7)$$

$$t_{12} = t_2 - R_2 R_1^T t_1 \quad (3.8)$$

若以平面的幾何上來看，如圖3.2所示，第二台相機的座標系即是第一台相機的座標系逆時針旋轉了 R_{12} 之後，再向右平移 t_{12} 。

對於所重建出的三維點，若它們的座標系不同，我們也可以利用這些已知的相機座標系關係，將重建點轉換為以同一個座標系為基礎。如圖3.2， M_1 代表重建物體於 C_1 座標系下的座標向量，若我們要計算同一個物體相對於 C_2 座標系下的座標向量 M_2 ，相當於算出 M_1 經過與相機座標系相反的旋轉及平移之後，所得到相對於 C_1 座標系的點座標，也就是圖中的 M_1' 。而由 M_1 到 M_1' 的這個過程，剛好是和相機座標系的旋轉和平移相反，如圖3.2 所示， M_1 先向左平移了 $-t_{12}$ 之後，再順時針旋轉 R_{12}^T ，得到 M_1' 。因此，我們可以將這樣的轉換寫成以下的關係式：

$$M_2 = M_1' = R_{12}^T(M_1 - t_{12}) \quad (3.9)$$

其中， M_1 為原物體相對於 C_1 座標系下的座標向量， M_2 為原物體相對於 C_2 座標系下的座標向量，而 M_1' 則為原物體經過與相機座標系相反的旋轉及平移之後，所得到相對於 C_1 座標系下的座標向量。反之，若我們已知 M_2 而欲求 M_1 ，則可透過相反的旋轉及平移過程，關係式如下：

$$M_1 = R_{12}M_2 + t_{12} \quad (3.10)$$

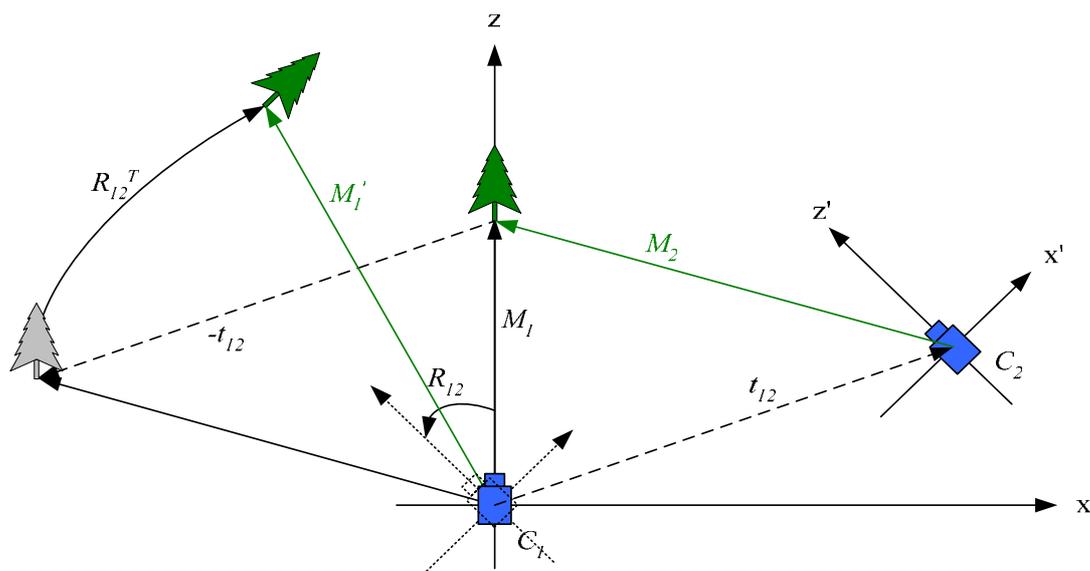


圖3.2 相機座標與點座標的旋轉及平移關係。

第四章 指向系統之精確度分析

由前面的理論基礎，我們已發展一套初步的視覺系統[18]，本章就是利用這一套已發展的系統，在不同的場景條件下，進行三維重建，並分析指向系統之精確度。

本篇論文所實作的智慧型指向系統，一開始需先將場景的三維模型重建好，並且保留計算好的基本矩陣，之後當我們在判斷指向方向時，只需重建指向物體的兩個特徵點，不必再重建整個場景，也不需要再算一次基本矩陣，可以加速運算，簡化重建所需的步驟。實驗流程圖如圖4.1，共分為相機校正、特徵點的選取、特徵點對應、三維重建、計算指向物體所指的目標點等五個步驟，前面四個步驟均是沿用[18]之程式以進行實驗。我們定義指向物體上代表指向方向的兩個特徵點，稱為指向特徵點，而兩個指向特徵點的延伸，在正交方塊圖板所指到的位置，稱為目標點。第五個步驟計算指向物體所指的目標點，即是利用三維重建，以取得兩個指向特徵點的三維座標，進而計算出所指向的目標點。在實驗結果的分析方面，我們分別探討基本矩陣不同，解析度不同，指向方向不同以及指向物體不同的情況下，所重建出來的三維場景及指向位置。實驗的環境為 Windows XP 作業系統之個人電腦，CPU為P4 2.8G， 512MB 記憶體，程式平台為 Matlab 6.5，所用的數位相機，右影像為 Sony P100 左影像為 Sony P10，經過相機校正，我們取得其相機內部參數如下：

$$\begin{bmatrix} 720.60 & -1.3817 & 308.62 \\ 0 & 722.92 & 233.28 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 723.11 & -1.6592 & 338.96 \\ 0 & 724.26 & 242.31 \\ 0 & 0 & 1 \end{bmatrix}$$

Sony P100

Sony P10

圖4.2 為指向系統實驗中的實驗場景，是由兩個夾角約為90度的平面所構成的正交方塊圖板，每個平面上有16個方塊。我們針對這些方塊圖案平面進行三維重建，並且藉由指向物體來判斷所指向的方塊。

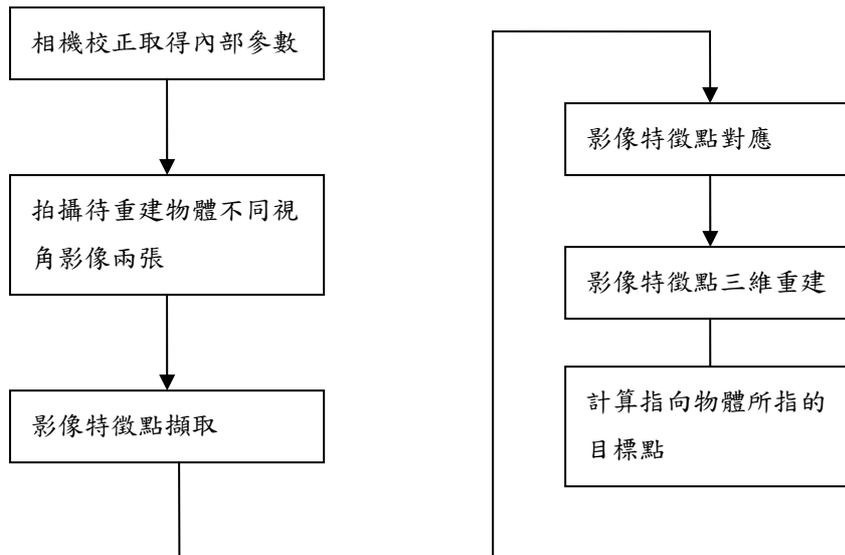
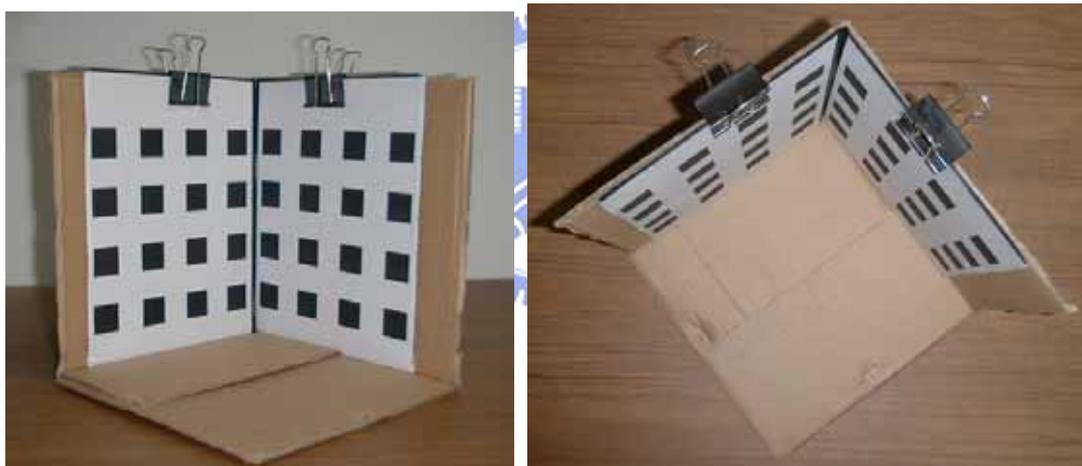


圖4.1 指向系統實驗流程圖。



(a)

(b)

圖4.2 正交方塊圖板之 (a)側視影像 (b)俯視影像。

4.1 使用不同的指向物體進行三維重建

指向物體對於指向系統的精確度有相當大的影響，一個好的指向物體可以幫助我們取得準確的二維指向特徵點，二維指向特徵點的準確性對於三維重建時，指向物體三維座標的計算相當重要，如果重建出來指向物體的兩個指向特徵點之三維座標跟真實座標相差了一個點距，對於計算指向的目標點可能會相差五個點

距以上。一個最理想的指向物體，就是具有可以很準確被我們取得的二維指向特徵點，且僅需極少量的計算。如圖 4.3(a)所示，我們以黑色長方條的右邊兩點之延伸當作我們的指向方向，如果能取得這樣準確的二維指向特徵點，對於指向系統的精確度有很大的幫助，但這樣準確的二維指向特徵點，在現實生活中並不容易取得，因為會有遮蔽以及角度的問題，使得我們不易取得這樣的兩個點。圖 4.3(b)中的指向雷射筆為本次實驗所採用的指向物體，其優點為不管在任何的角度都能準確的取得四個角點，上方兩個角點的中點與下方兩個角點的中點之延伸即為指向方向。

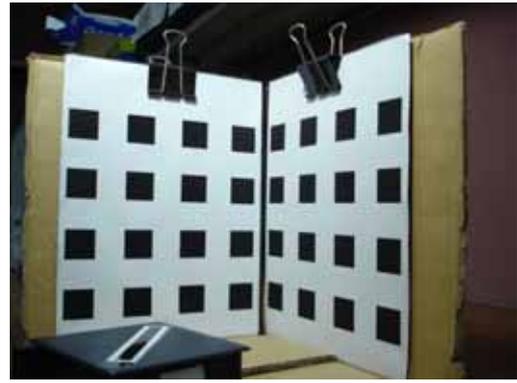


圖4.3 (a)指向長方條 (b)指向雷射筆。

圖4.4 為使用指向長方條做指向的實驗結果，由結果發現指向精確度很高，與我們所預期的位置非常接近，圖4.5 為使用指向雷射筆做指向的實驗結果，由結果發現指向精確度也很高，與我們所預期的位置也非常接近。由於指向雷射筆可以射出雷射光打在立體校正板上，方便我們使用肉眼判斷指向效果的準確性，因此，在接下來的實驗中，皆使用指向雷射筆當作我們的指向物體。



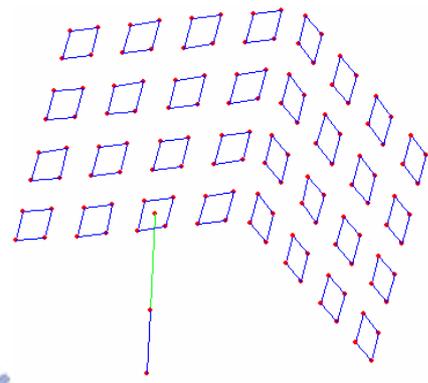
(a)



(b)



(c)



(d)

圖4.4 使用指向長方條做指向的實驗結果 (a)重建場景的左影像 (b)重建場景的右影像 (c)利用尺的延伸去估計指向的目標點 (d)重建之後的目標點。

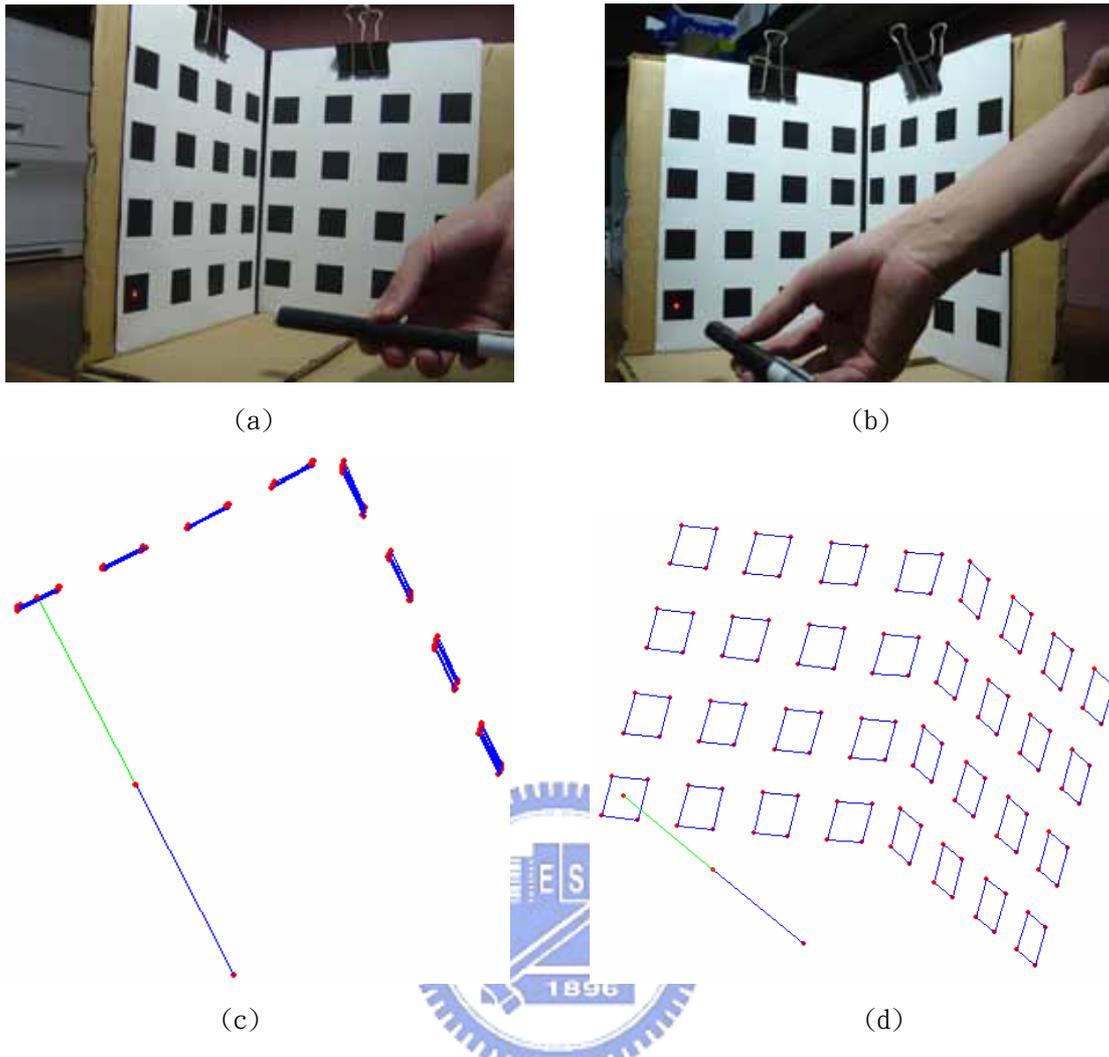


圖4.5 使用指向雷射筆做指向的實驗結果 (a)重建場景的左影像 (b)重建場景的右影像
(c)重建場景的俯視影像 (d)重建之後的目標點。

4.2 使用不同的基本矩陣進行三維重建

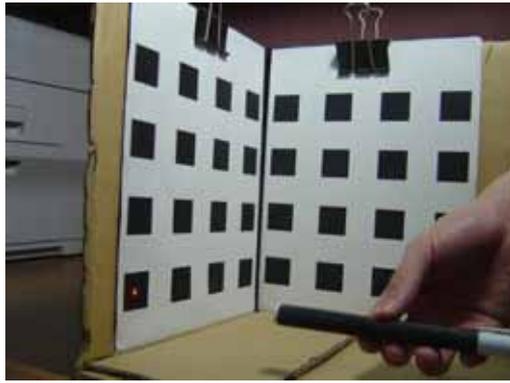
在三維重建中，基本矩陣的正確性，對於重建的結果有很大的影響，本節將會把正交方塊圖板上 128 個特徵點取編號如圖 4.6，左邊的校正版由左至右由上而下取編號 1 到 64，右邊的校正版由左至右由上而下取編號 65 到 128，我們取特定的特徵點去算基本矩陣，圖中被選到之特徵點將以圓圈做記號。我們將會經由結果，來探討不同特徵點取法所算出的基本矩陣，對於三維重建以及指向結果的影響。



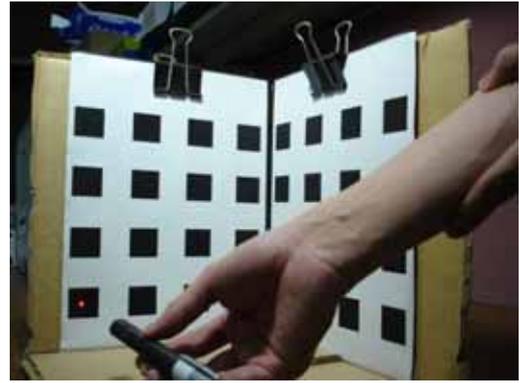
圖4.6 正交方塊圖板特徵點編號。

圖 4.7 為取八個點與取相同八個點再加上指向的兩個點共十個點去算基本矩陣所重建出來的場景與目標點，由重建之後的結果可以發現，使用八個點算基本矩陣所重建出來的場景比十個點算基本矩陣所重建出來的場景效果好很多，但是關於目標點的計算，使用十個點算基本矩陣所重建出來的指向點比八個點算基本矩陣所重建出來的目標點準確，由此結果我們可以推論，在此實驗中，我們所取的八個點是分布範圍廣且位置組合佳的點集合，而加入指向的兩個點之後，雖可以使得算出來的目標點較為準確，但是這十個點所成的點集合，位置組合不佳，所算出來的基本矩陣，正確性不足。

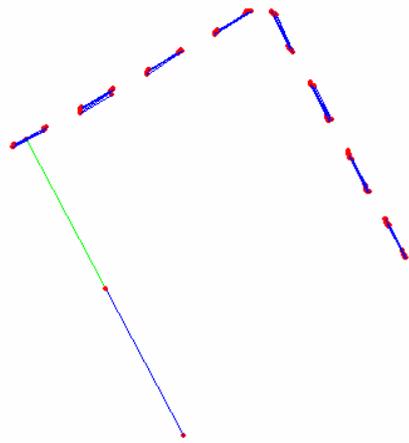
圖 4.8 為兩種取 64 個點去算基本矩陣所重建出來的場景與目標點，第一種取法為左右兩邊各取 32 個特徵點共 64 個點，第二種取法為取左半邊 64 個點，由重建之後的結果可以發現，使用第一種取法算基本矩陣所重建出來的場景比第二種取法算基本矩陣所重建出來的場景效果好很多，關於目標點的計算，由於第二種取法重建時就已經不正確，所以也無法正確的算出目標點。



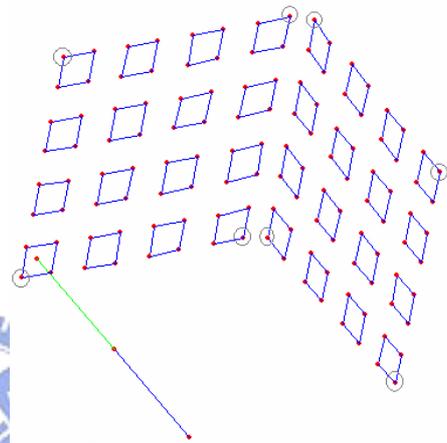
(a)



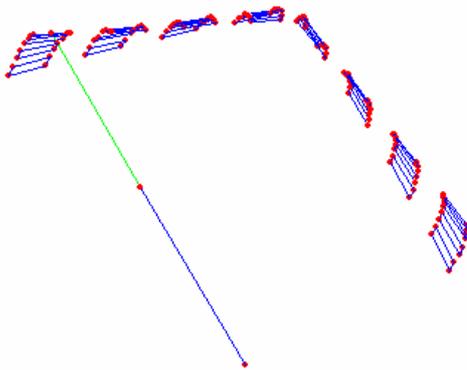
(b)



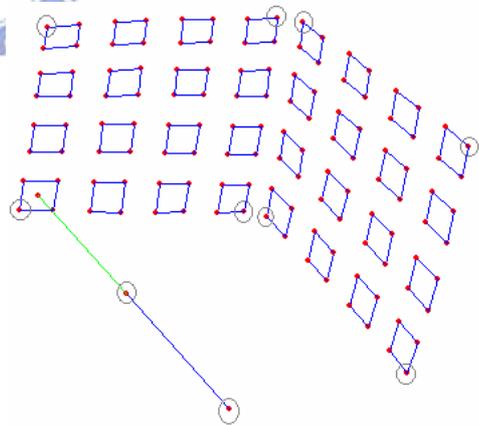
(c)



(d)

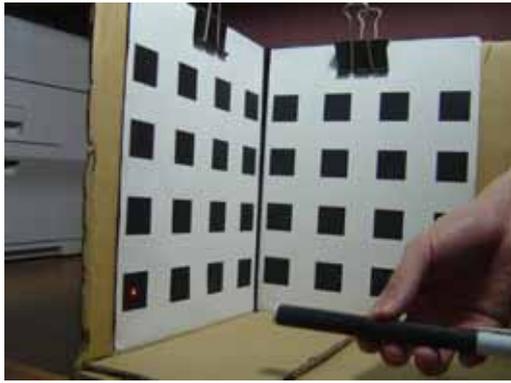


(e)

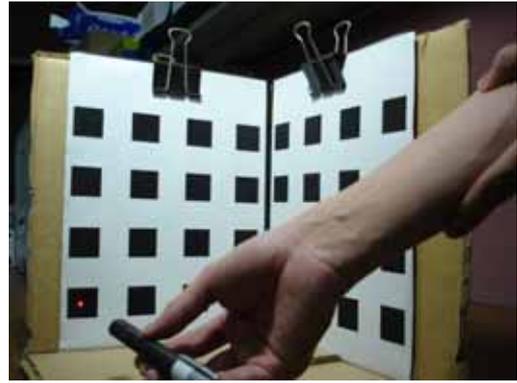


(f)

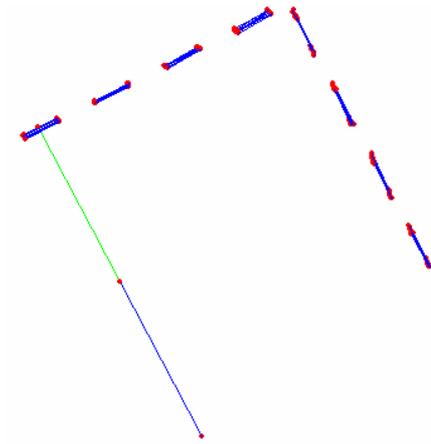
圖4.7 (a)重建場景的左影像 (b)重建場景的右影像 (c)取八個特徵點所算出的基本矩陣之重建結果俯視圖 (d)取八個特徵點所算出的基本矩陣之目標點重建結果 (e)取十個特徵點所算出的基本矩陣之重建結果俯視圖 (f)取十個特徵點所算出的基本矩陣之目標點重建結果。



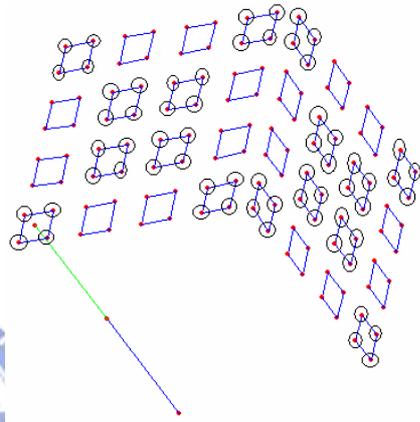
(a)



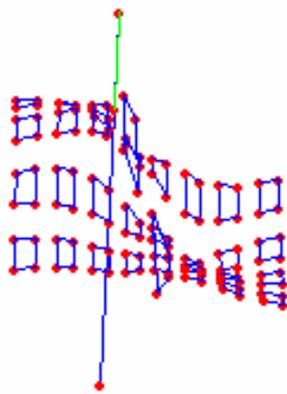
(b)



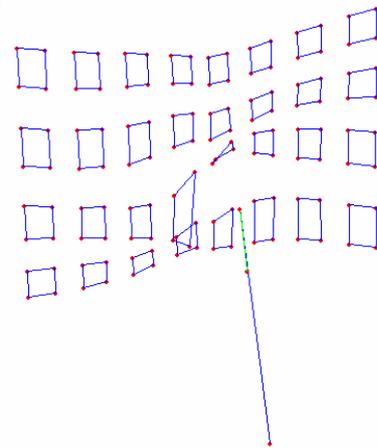
(c)



(d)



(e)



(f)

圖4.8 (a)重建場景的左影像 (b)重建場景的右影像 (c)左右兩邊各取32個特徵點所算出的基本矩陣之重建結果俯視圖 (d)左右兩邊各取32個特徵點所算出的基本矩陣之目標點重建結果 (e)取左半邊64個特徵點所算出的基本矩陣之重建結果俯視圖 (f)取左半邊64個特徵點所算出的基本矩陣之目標點重建結果。

由圖 4.7 與圖 4.8 我們可以得知，基本矩陣對於三維重建的影響很大，而基本矩陣的正確性決定於特徵點的選取。圖 5.7 (c)(d)中，只取了八個特徵點去算基本矩陣，就能得到相當好的重建結果，不過，如果特徵點的距離太近或者是特徵點皆位於同一平面，所算出的基本矩陣去做三維重建，都無法得到很好的重建結果。圖 5.9 為與圖 5.7 相同的重建場景但取不同的八個特徵點去算基本矩陣所重建出來的結果。

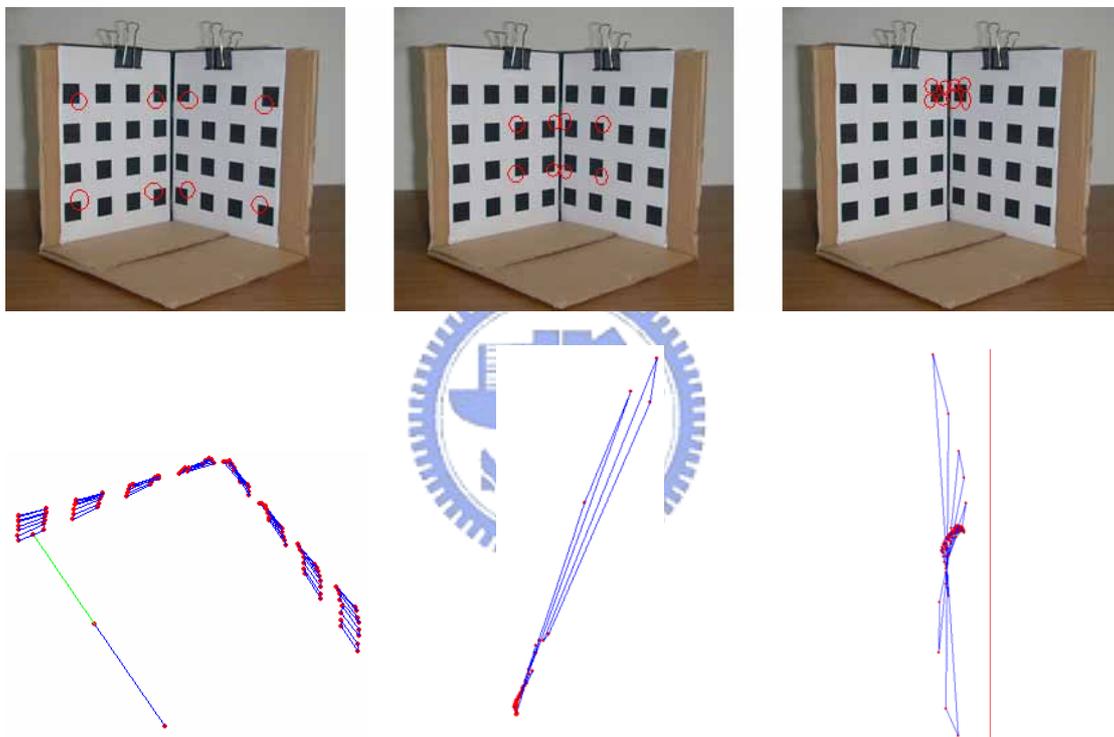
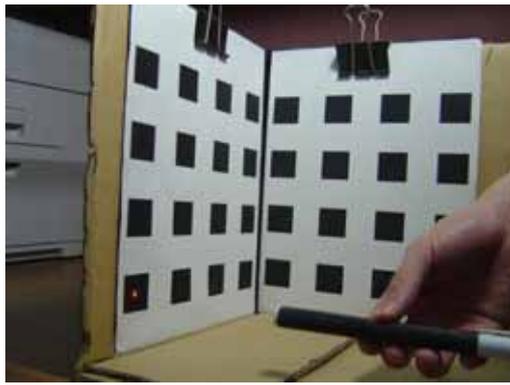
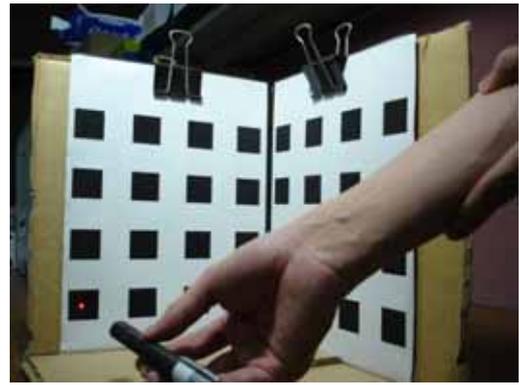


圖4.9 取不同的八個特徵點去算基本矩陣所重建出來的結果。（上方的圖顯示所選取之特徵點，下方的圖為其重建結果）

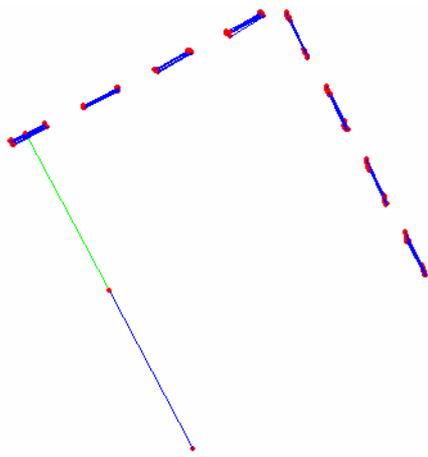
值得注意的是，雖然八點演算法理論上僅需要八組對應點即可求解，但是為了數值上的可靠度，通常取超過八組以上，這樣所算出來的基本矩陣穩定性較高。圖 4.10 為比較取編號 1 至 128 的特徵點與取編號 1 至 128 的特徵點在加指向的兩個特徵點所算出來的基本矩陣去做三維重建的結果。



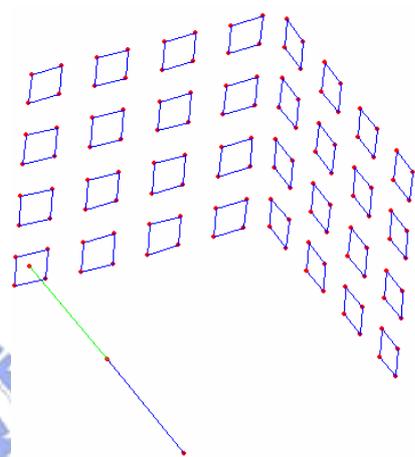
(a)



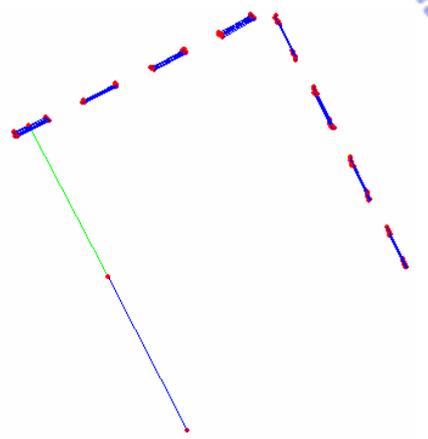
(b)



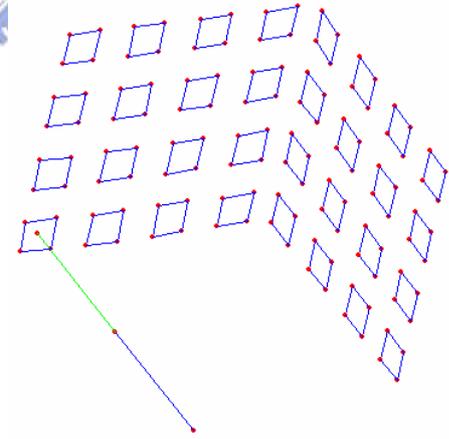
(c)



(d)



(e)



(f)

圖4.10 (a)重建場景的左影像 (b)重建場景的右影像 (c)取128個特徵點所算出的基本矩陣之重建結果俯視圖 (d)取128個特徵點所算出的基本矩陣之目標點重建結果 (e)取130個特徵點所算出的基本矩陣之重建結果俯視圖 (f)取130個特徵點所算出的基本矩陣之目標點重建結果。

經由圖 4.10 我們可發現，用 128 個特徵點與 130 個特徵點所算出來的基本矩陣重建效果都很好，目標點的計算也十分的接近，而這兩個方式所算出來的基本矩陣數值相差也不會太大，其中用 130 個特徵點算出的基本矩陣為

$$F_1 = \begin{bmatrix} -0.00000017137482 & -0.00000320505653 & 0.00116170512412 \\ -0.00000334586110 & -0.00000030556742 & -0.00598561095383 \\ 0.00088377341491 & 0.00861520178859 & -0.99994390848878 \end{bmatrix}$$

而用 128 個特徵點算出的基本矩陣為

$$F_2 = \begin{bmatrix} -0.00000002303801 & -0.00000308015675 & 0.00100317885327 \\ -0.00000359650070 & -0.00000029713941 & -0.00613448149988 \\ 0.00098197532663 & 0.00879334241111 & -0.99994153564993 \end{bmatrix}$$

兩個基本矩陣對應元素的差值可表示為

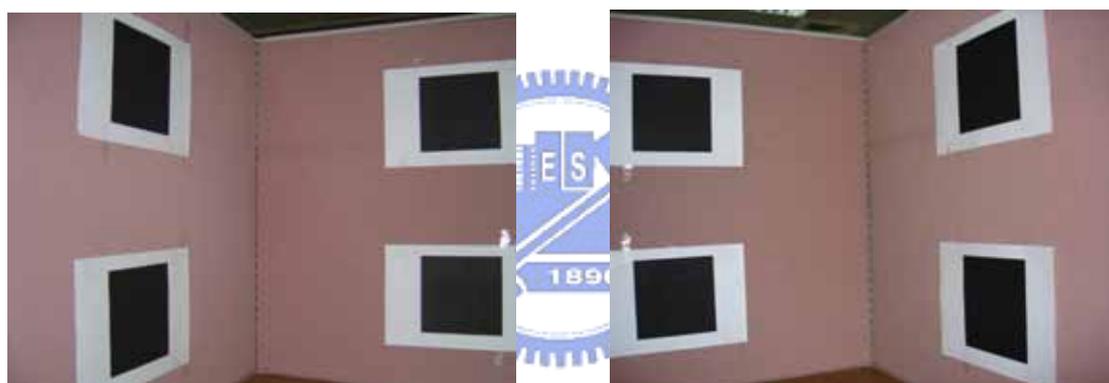
$$|F_1 - F_2| = \begin{bmatrix} 0.00000014833681 & 0.00000012489978 & 0.00015852627085 \\ 0.00000025063960 & 0.00000000842801 & 0.00014887054605 \\ 0.00009814191172 & 0.00017814062252 & 0.00000237283885 \end{bmatrix}$$

比較圖 4.7 (c)(d)、圖 4.8 (c)(d)和圖 4.10 (c)(d) 我們也可以發現，8、64 以及 128 個特徵點所算出的基本矩陣重建的效果都很好，所以由本節所有的實驗結果我們可以得知，當我們在取特徵點時，只要取到足夠多以及對應穩定的特徵點，即可得到很好的重建結果。

4.3 使用不同的解析度進行三維重建

二維影像特徵點的擷取，對於三維場景重建的正確性，有著很重要的影響，如果所擷取的特徵點誤差過大，重建出來的場景會有扭曲與變形的現象。本節將會以 640 x 480 和 1280 x 960 兩種不同的解析度去做三維重建，並分析其指向精確度。

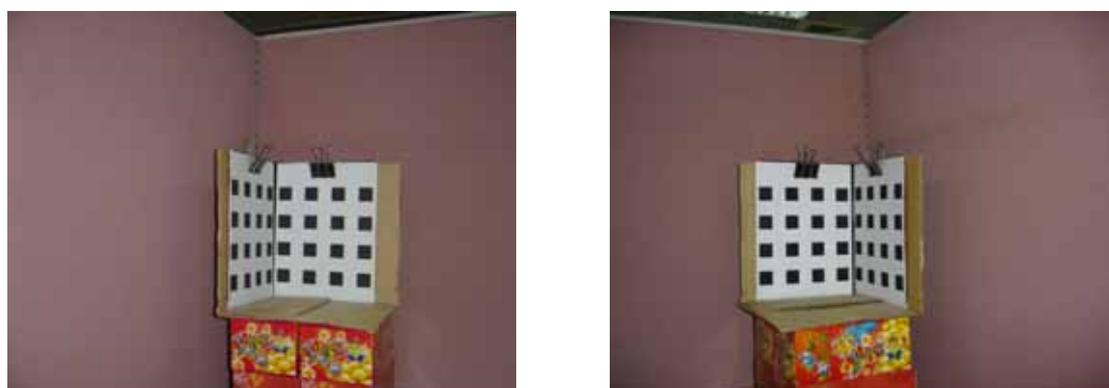
經由 4.2 節的討論我們可以得知，只要找到足夠多以及位置組合佳的特徵點，即可得到很好的重建結果，而在我們日常生活的場景中，其實不容易找到這樣的點，我們可以利用人工的方式，先使我們想要的特徵點出現在場景中，利用這些特徵點去算基本矩陣，然後以算出來的基本矩陣去重建場景中的物體。本節的實驗與之前的實驗有一點不同，之前實驗所用的基本矩陣都是以待重建物體上的特徵點去算的，現在我們以圖 4.11 所示之隔板為本節實驗的場景，圖中的四個方塊是以人工的方式加上去的特徵點紙板，我們稱這四個方塊的 16 個特徵點為輔助點是用來幫我們算基本矩陣，基本矩陣算出來之後，即把人工加上去的特徵點紙板移除，然後以這個基本矩陣去重建我們所要重建的物體如圖 4.12 所示。



(a)

(b)

圖4.11 以人工方式加入場景的特徵點 (a)左影像 (b)右影像。



(a)

(b)

圖4.12 待重建物體 (a)左影像 (b)右影像。

圖 4.13 為使用上述的方法以 640 x 480 的影像做三維重建之結果，由圖中我們可以發現，場景重建結果還不錯，只是方塊的形狀有些許的變形，而關於目標點位置的計算，大約向左偏移了半個方塊邊長的距離。而圖 4.14 也是使用上述的方法但以 1280 x 960 的影像做三維重建，由圖中我們可以發現，場景重建的結果很好，而目標點位置的計算也只向上偏移了一點。

經由本節的實驗我們可以發現，不同大小的解析度對於三維重建的結果以及所指向的目標點計算，有著舉足輕重的影響，對於重建的物體而言，如果解析度不夠大，無法準確的擷取特徵點，那麼所重建出來的物體，就容易產生扭曲以及形變的現象，而目標點的計算也會有產生誤差。

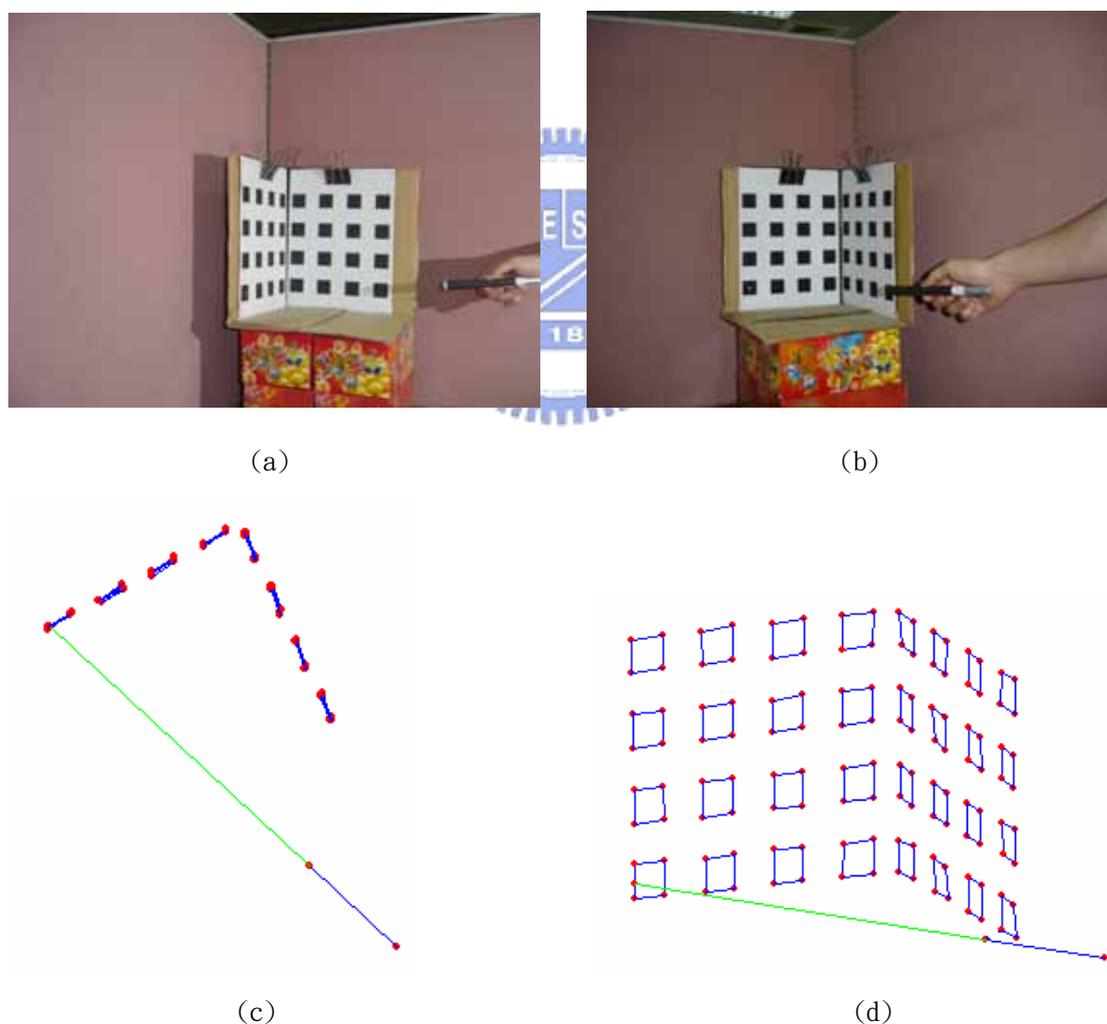


圖4.13 使用640 x 480的影像做三維重建 (a)場景的左影像 (b)場景的右影像 (c)重建場景的俯視影像 (d)重建的目標點。

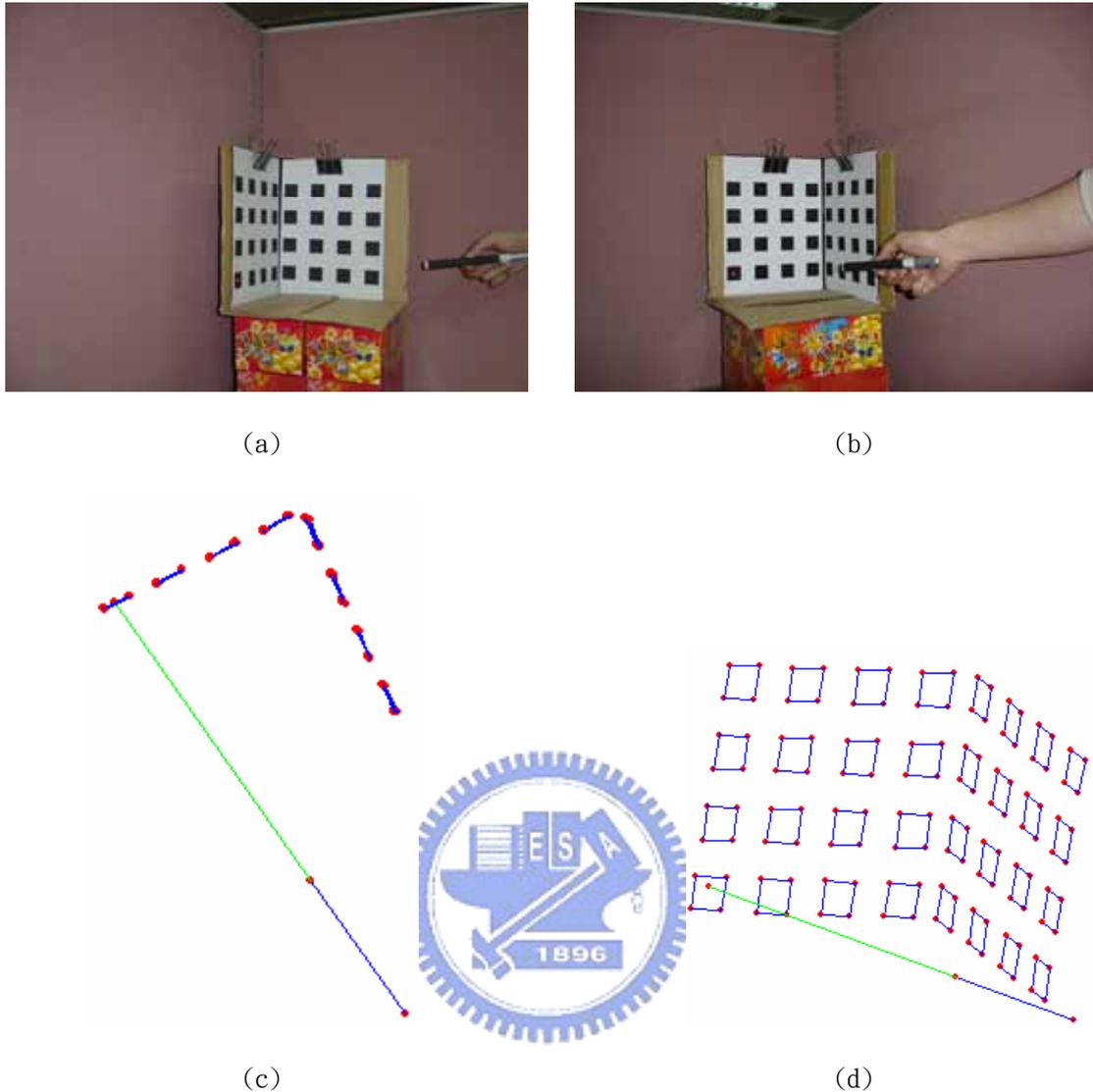


圖4.14 使用1280 x 960的影像做三維重建 (a)場景的左影像 (b)場景的右影像 (c)重建場景的俯視影像 (d)重建的目標點。

4.4 不同的指向方向進行三維重建

在即時 (real time) 的指向系統中，如果我們能夠將指向的結果，即時的顯示在我們人的眼前（即系統的 feedback），指向的精確度或許就變的沒那麼的重要了。假如顯示結果所指的目標點，不是我們所想要的位置，只要所指的方向不會差太多，我們可以移動指向物體往想去的方向去，而所指的方向會隨著指向物體的移動而移動，如此，指向系統應能順利達成指向的任務。

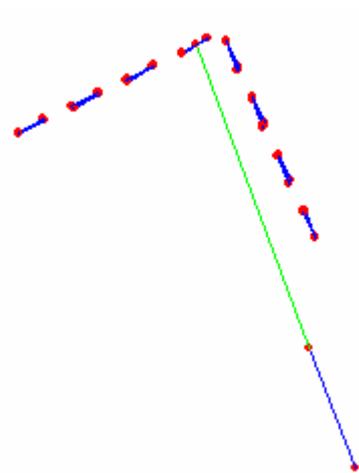
在本節的實驗中，我們將測試水平與垂直方向上的移動。我們以圖 4.14 為基準，若將指向物體向右移動三方塊大小，則結果如圖 4.15，若將指向物體向上移動三方塊大小，則結果如圖 4.16。



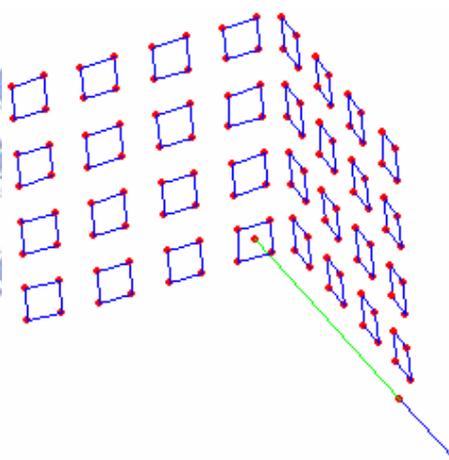
(a)



(b)



(c)



(d)

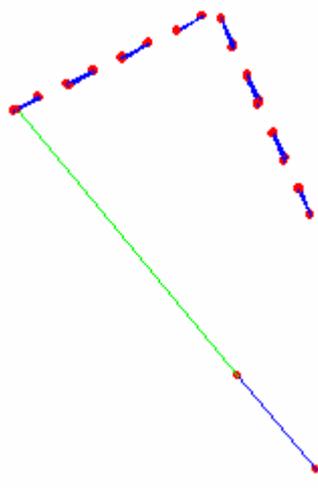
圖4.15 指向方向往右移動：(a)重建場景的左影像 (b)重建場景的右影像 (c)重建場景的俯視影像 (d)重建之後的目標點。



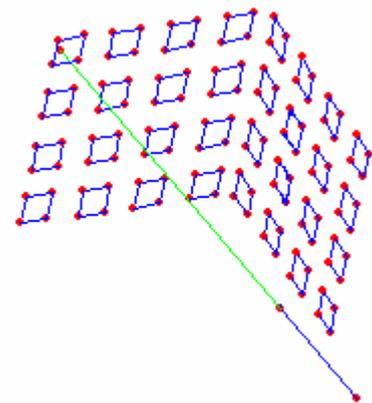
(a)



(b)



(c)



(d)

圖4.16 指向方向往上移動：(a)重建場景的左影像 (b)重建場景的右影像 (c)重建場景的俯視影像 (d)重建之後的目標點。

由實驗結果發現，本篇論文所提出的指向系統，不管是在水平或者是垂直方向上的移動，其所指的方向都會相當正確的隨著指向物體的移動而移動，也就說明了我們的指向系統，在各個不同的方向都能得到不錯的指向效果。表 4.1 為使用輔助點去算基本矩陣與未用輔助點去算基本矩陣（即以立體方塊圖版上 128 個點去算基本矩陣）的誤差比較表，由表中我們可以發現使用輔助點可以使我們的誤差減少三分之二，達成改善指向精確度的目標。

表4.1 使用輔助點與未使用輔助點的誤差比較表(單位:公分)

	圖 4.14	圖 4.15	圖 4.16	平均誤差
使用輔助點算基本矩陣，目標點的誤差	0.7	0.2	0.9	0.6
未使用輔助點算基本矩陣，目標點的誤差	1.4	3	1	1.8



第五章 結論與未來工作

在現有的指向系統中，大部分都是利用手勢來判斷指向方向，需要經過模板的比對，才能計算手指頭所指的方向，本論文以一套現有的電腦視覺軟體系統，使用三維重建去做指向的任務，我們所提出的方法，不需要經過模板的比對，即能取得準確的二維指向特徵點，減少計算量，而在三維場景重建的部份，我們也提出數種計算基本矩陣的方法，用以改善三維場景重建的效果，以及減少計算基本矩陣的時間。

在本論文的分析實驗裡，為了避免被特徵點擷取產生的誤差所影響，因此在特徵點的選擇與對應，是以人工介入的方式來達成，未來可進一步與前端的特徵點自動擷取系統結合，並實作出即時的系統，分析重建與指向的結果，並從影像品質、特徵點的選取與對應、視角與視角差之決定等方面設法改善，以滿足實際應用的要求。



參考文獻

- [1] P. Narayanan, P. Rander, and T. Kanade, "Constructing Virtual Worlds Using Dense Stereo," *Proc. International Conference on Computer Vision*, 1998.
- [2] M. Okutomi and T. Kanade, "A Multiple-baseline Stereo," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353-363, April 1993.
- [3] M. Pollefeys and L. Van Gool, "From Images to 3D Models," *Communications of the ACM*, vol. 45, no. 7, pp.50-55, July 2002.
- [4] Shin Sato and Shigeyuki Sakane, "A Human-Robot Interface Using an Interactive Hand Pointer that Projects a Mark in the Real Work Space," *IEEE International Conference on Robotics & Automation*, April 2000.
- [5] Yi-Ping Hung, Yao-Strong Yang, Yong-Sheng Chen, Ing-Bor Hsieh, Chiou-Shann Fuh, "Free-Hand Pointer by Use of an Active Stereo Vision System," *Proceedings of 14th International Conference on Pattern Recognition*, pp. 1244-1246, Brisbane, August 1998.
- [6] Rahul Sukthankar, Robert G. Stockton and Matthew D. Mullin, "Self-Calibrating Camera-Assisted Presentation Interface," *Proceedings of International Conference on Control, Automation, Robotics and Vision*, 2000.
- [7] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2001.
- [8] O. Faugeras and Q.-T. Luong, *The Geometry of Multiple Images*. The MIT Press, 2001.
- [9] M. Pollefeys, "Tutorial on 3D Modeling from Images," in conjunction with *European Conference on Computer Vision*, June 26, 2000.
- [10] O. Faugeras, "Stratification of Three-Dimensional Vision: Projective, Affine, and Metric Representations," *Journal of the Optical Society of America*, vol.12, no.3, pp. 465-484, 1995.
- [11] M. Pollefeys, "Self-Calibration and Metric 3D Reconstruction from

Uncalibrated Image Sequences," Ph.D. Thesis, Katholieke Universiteit Leuven, 1999.

- [12] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330-1334, November 2000.
- [13] H.C. Longuet-Higgins, "A Computer Algorithm for Reconstructing A Scene from Two Projections," *Nature*, vol. 293, pp. 133-135, 1981.
- [14] K.-L. Lee, "Optimization of Fundamental Matrix and Euclidean Reconstruction," Master Thesis, National Chiao Tung University, 2002.
- [15] R. Hartley, "In Defense of the Eight-point Algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580-593, June 1997.
- [16] Z. Zhang, "Determining the Epipolar Geometry and Its Uncertainty: A Review," *Research Report, No. 2927, INRIA Sophia-Antipolis*, July 1996.
- [17] T. Huang and O. Faugeras, "Some Properties of the E Matrix in Two-View Motion Estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 12, pp. 1310-1312, Dec. 1989.
- [18] L.-C. Guo, "Robot Vision for 3D Scene Reconstruction," Master Thesis, National Chiao Tung University, 2003.