

fterindentfalseaafterindenttrue aafterindenttrue



國立台灣科技大學  
資訊工程研究所

---

## 碩士論文

3D 點雲與 SIFT 特徵點室內影像定位之研究

3D Point Cloud and SIFT Descriptor Indoor Localization  
Research

研 究 生：陳致良  
學 號：M9915057

指 導 教 授：項天瑞博士

中華民國一百零一年七月十日





# 3D 點雲與 SIFT 特徵點室內影像定位之研究

學生：陳致良

指導教授：項天瑞博士

國立台灣科技大學資訊工程研究所

## 摘要

本篇論文探討如何利用單張彩色影像來重建出三維人臉模型。我們的方法是利用彩色影像來取代灰階值去建立張量模型 (tensor model)，在人臉資料庫中，是用典型相關分析 (Canonical correlation analysis) 是來建立彩色影像與深度資訊的對應關係，一旦建立好屬於各自的對應關係後，在重建人臉的過程中，就只需要單張彩色影像即可利用典型相關分析的對應關係來推算出正確的深度資訊。實驗中，我們的方法可以在不同的光線環境跟人臉角度得到不錯的效果。

# 3D Point Cloud and SIFT Descriptor Indoor Localization Research

Student: ZZhi-Liang Chen

Advisor: Dr. Tien-Ruey Hsiang

Submitted to Department of Computer Science and Information  
Engineering

College of Electrical Engineering and Computer Science  
National Taiwan University of Science and Technology

## ABSTRACT

This paper develops a tensor-based 3D face reconstruction approach from a single color image. Instead of the grayscale image, we also consider additional color factors in constructing the tensor model. Canonical correlation analysis is applied to establish the relationship between the color image and the depth information in the face database. During the face reconstruction, given a single color face image, the depth estimation is computed from the CCA-based mapping between the tensor models. Experimental results show our approach is better suited under different lighting conditions and poses.

# 誌謝

本論文能夠完成，首先要感謝的是指導教授項天瑞老師。老師嚴謹的治學態度，讓我不但在學術研究上學習到更謹慎的思考，也在日常生活上獲益良多。

感謝實驗室的同學們，實驗室的生活有苦有樂，有你們才讓我能撐得下這三年漫長的時間。感謝建群、松翰學長給我的指導。感謝益偉、崇峰、承志、誠儀、慶豪學長以及恩緯、盈樽、嘉駿、世寬學長給我的指導。感謝實驗室的學弟妹們，訓哲、立昂、致良、青緯、庭耀、宗博、雅筑，常常幫了我不少忙。更要感謝我的同梯們，貴彥、薇穎、冠佑，我們同甘共苦，一起奮鬥，特別是在弄計畫的過程中，冠佑幫助了我很多，有你們在我才能走到這一步。也感謝好朋友志強跟我一起修正文法與用詞。

最後我要把我最深的感謝留給我的家人。謝謝我的爸媽跟兩位姊姊，你們讓我沒有經濟壓力地讀完這這碩士學位，也常常給我很多鼓勵，今天我終於拿到這個學位，終於可以讓你們放下心上的一塊大石頭了。

# 內文目錄

論文指導教授推薦書	i
考試委員審定書	ii
摘要	iii
Abstract	iv
誌謝	v
內文目錄	vi
圖目錄	viii
表目錄	ix
1 虛擬環境建置與定位方法描述	1
1.1 方法大綱	1
1.2 利用 3D 點雲重建當時定位環境	1
1.2.1 重建初步模擬環境	3
1.2.2 將點雲環境限制範圍符合真實環境重建	4
1.3 虛擬照相機設置及影像資料庫建置	5
1.3.1 均勻分布設置虛擬攝影機	6
1.3.2 虛擬照相機成像原理	7
1.3.3 根據深度來調整攝影機角度	10
1.3.4 儲存虛擬照相機圖片	11
1.4 虛擬影像定位	12
1.4.1 前置作業處理	13
1.4.2 尋找特徵點並找出最多的特徵點投票選出位置	13

1.4.3 利用虛擬照相機位置來定位 . . . . .	13
2 定位實驗方法比較分析 . . . . .	18
2.1 實驗目的 . . . . .	18
2.2 可控制環境定位數據比較 . . . . .	18
2.2.1 可控制環境實驗方法 . . . . .	18
2.2.2 可控制環境實驗數據分析 . . . . .	19
Bibliography . . . . .	23

# 圖 目 錄

2.1 控制環境實驗參數設定 . . . . .	19
--------------------------	----

# 表 目 錄

1.1	3D 點雲環境座內定位整體流程圖	2
1.2	初步建置好的點雲環境	4
1.3	將點雲給予界限範圍	5
1.4	調整點雲坐標軸角度方法	5
1.5	在點雲上設置虛擬照相機位置	6
1.6	glfrustrum 攝影角錐示意圖	7
1.7	根據物體距離鏡頭遠近來調整方位	10
1.8	內差法補強前後的差異圖	12
1.9	特徵點比較差異圖	14
1.10	相機與特徵點的夾角示意圖	15
1.11	定位點夾角與長度向量關係示意圖	16
2.1	依照同心圓的覆蓋算出每個半徑內的每張圖的特徵點平均數，左 控制環境 1 右。控制環境 2	20
2.2	特徵點數量趨勢圖	20
2.3	間距定位平均誤差趨勢圖	21

# 第 1 章 虛擬環境建置與定位方法描述

為了提升定位成功的覆蓋率與精準度，我們方法在於改進影像定位取樣的不足，與增加相片角度取樣的範圍。傳統影像定位所拍出的相片只能片段的取得環境特徵，導致部分環境範圍定位準確，但沒有被拍照的範圍會定位誤差過大。我們的作法利用點雲模擬當時環境，藉由取得環境利用虛擬相機拍攝相片，將虛擬相片集合製作成資料庫，與待定位照片比較求出定位點。此舉可增加相片取樣的不足與拍攝角度的變化性。

## 1.1 方法大綱

整套虛擬影像室內定位作法我們分成三大步驟來敘述：(1). 利用 3D 點雲重建當時定位環境 (2). 虛擬照相機設置及影像資料庫建置 (3). 虛擬影像定位。利用 Kinect 紅外深度攝影機，取得深度資訊後幫助我們做出 3D 的點雲環境。透過 3D 的點雲環境，再根據點雲中的坐標系來作格狀均勻分布 (Grid Permutation)，藉由有規律的分布來決定虛擬照相機的位置。因為每個照相機間距相同，代表觀察到的區域都有固定範圍，藉由角度上的調整就可以取得比一般 2D 影像包涵更廣的視角空間與更大的覆蓋範圍，取出更好的照片，以減低之後定位所造成的誤差。圖 1.1 為整體 3D 點雲環境座內定位的過程，這些流程會在之後章節逐一解釋步驟與這些步驟的目的。

## 1.2 利用 3D 點雲重建當時定位環境

要重建環境，先利用 3D 點雲建立初步的模擬資料，再經過調整逼近原來定位環境。藉由這些 3D 點雲，作出虛擬照片建置影像定位所需的資料庫。在這章節首先介紹怎麼重建當時的定位環境，3D 環境的重建分成下列五個步驟：

- (1) 取得 Kinect 照片
- (2) 將偵測到照片中的特徵點作隨機抽樣一致演算法 (RANSAC)

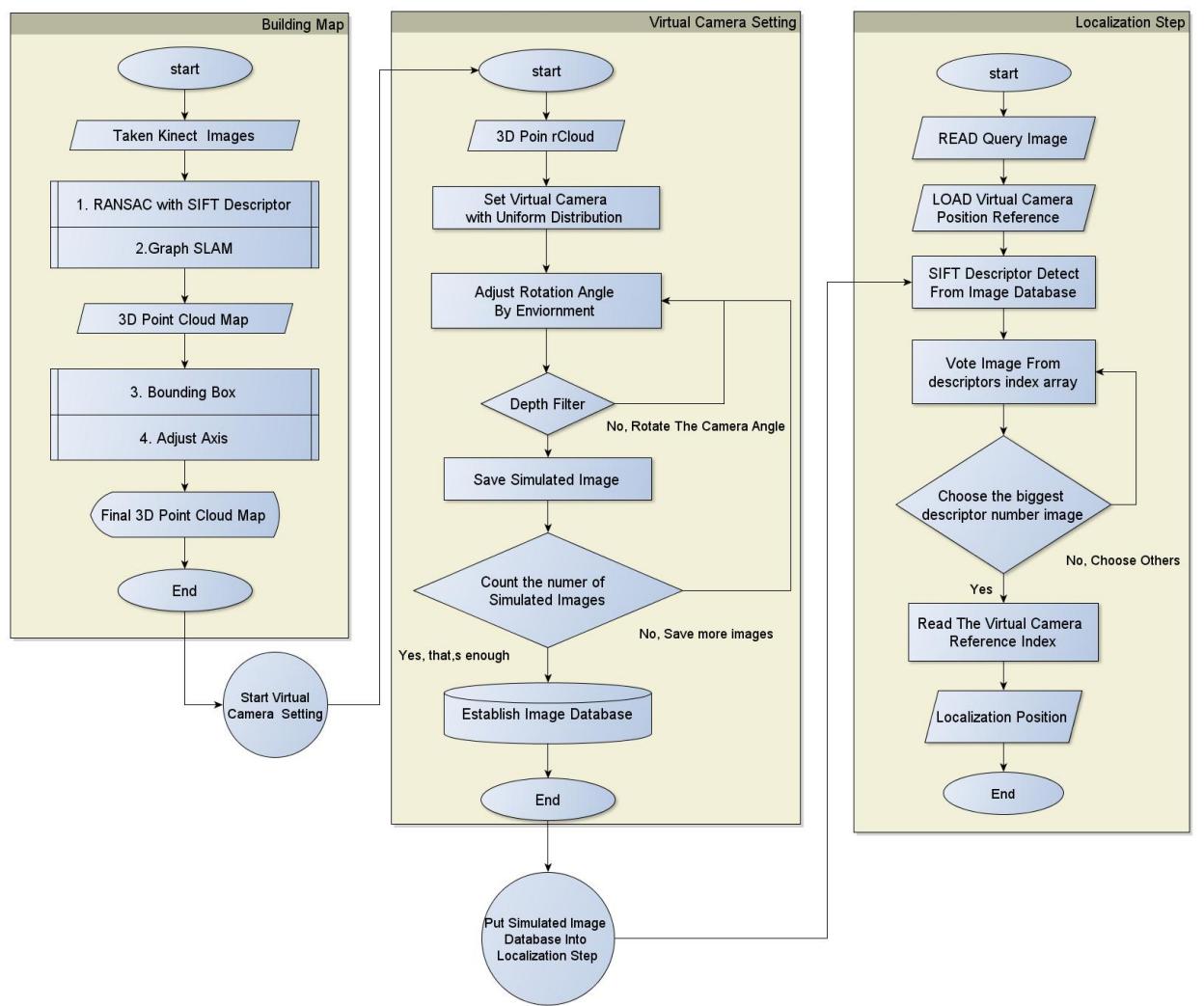


圖 1.1: 3D 點雲環境座內定位整體流程圖

- (3) 將組好的點雲作 Graph SLAM
- (4) 設置點雲涵蓋範圍的界線作界限範圍 (Bounding Box)
- (5) 調整坐標系

前面三個步驟主要是建置初步的環境，當環境完成模擬後，再透過剩下的兩個步驟對環境進行微調與限制。避免在四周看不到任何景物的地方，製造出沒有使用價值的虛擬相片。接下來將如何製作 3D 點雲環境分成兩大部分做說明。

### 1.2.1 重建初步模擬環境

一開始先用 Kinect 在環境中拍攝照片，照片取得的作法類似之前 [1] 的方法，在環境上對每個景物做一連串的拍攝，且每張拍攝出來的照片都需要有一些相似之處，環繞整個拍攝環境確保照片的組成具有連續性。當一個環境包含的景物越多代表所包含的特徵點越多，擁有豐富的特徵點數量在做 RANSAC 之後會有更好的重和效果。像之前在相關研究的章節所述，一個密集的點群所找到的平面會越接近真實點群的表現，所做出的轉移矩陣 (Transformation Matrix) 也會越精準，在進行重合時，不會有影像疊影或者是破碎導致點雲中空、物體歪斜扭曲的現象產生。連續的環繞拍攝是為了確保每一張影像相對位置關係沒有錯誤。RANSAC 最怕沒有順序的影像排列，順序不正確就無法找出影像的相對位置，也就是說根據轉移矩陣所做出的點雲位置可能會和實際景物在環境中的位置相差甚遠。因此拍攝照片在點雲建置的步驟中是影響最大的因素，其中可能光線的不足或是玻璃的反射等一些外在的因素都會導致之後在建置點雲的困難，所以在作拍攝時最好都避免這些不利的因素。

拍攝完照片組之後，再利用尺度不變特徵向量 (Scale Invariant Feature Transform) 在從這些照片中取得特徵點的位置，將這些特徵點的位置求取 RANSAC，使得每一張影像都能夠在 3D 坐標系中與正確的位置中重合。我們將每張圖片找出來的特徵點作配對，求出來配對關係後，再將這些配對關係作最小平方法 (Least Square Error) 求出想要的平面，根據不同平面求出轉移矩陣 (Transformation Matrix)，最後可以得到之前照片影像位置的絕對關係，我們稱為 Global Pose，Global Pose 包含照片在三維座標以及照片在當時拍攝的角度。之後 2D 影像的定位都需要以它作每張影像定位的起始原點，在其他地方像是



圖 1.2: 初步建置好的點雲環境

點雲之後需要調整亦或是找出界限範圍，都需要利用 Global Pose 的起始原點當做參考，在我們的研究中，我們是使用虛擬相機的 Global Pose，至於虛擬相機的 Global Pose 的設置方式會在之後詳加說明。有了 Global Pose 的位置後，最後我們利用 Graph SLAM 將點雲作最後的調整。

Graph SLAM 描述，記得加完整版上去。

### 1.2.2 將點雲環境限制範圍符合真實環境重建

前三個步驟將點雲環境完成之後，在我們需要在環境中制定界線範圍 (Bounding Box)，用照相機的 Global Pose 調整角度。目的是為了讓虛擬照相機能夠被擺放在限定的範圍內，而不會有照相機放置在點雲環境之外，因而無法產生出有效的虛擬影像作定位。具體的做法如下，當我們讀入整個點雲之後，找在點雲中最大及最小的  $X$  及  $Y$  座標，知道了這四個座標，可以求出位於 Bounding Box 的頂點座標。最後座標相減求出的長度即為 Bounding Box 的長寬。有了這些長度之後，就可以知道整個點雲環境的長寬距離為多少，在之後可防止虛擬照相機坐落在散布點雲以外的位置。

制定點雲環境的界限範圍之後，做出來的點雲可能會因為之前 Kinect 攝影機的 Global Pose 歪斜分布而使得點雲也會有歪斜的狀況產生，這時候就必須將點雲作角度上的調整。這個步驟是為了之後的虛擬相機在照相時不會因點雲角度歪斜而使得照出來的角度與真實相機的角度差異過大，減少許多對應的特徵點，導致定位的誤差產生。我們的做法是看出點雲分布的情況是往哪個方向歪斜，利用

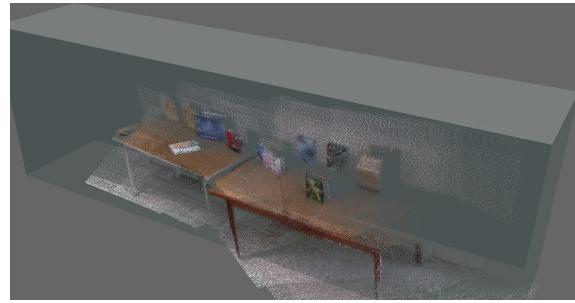
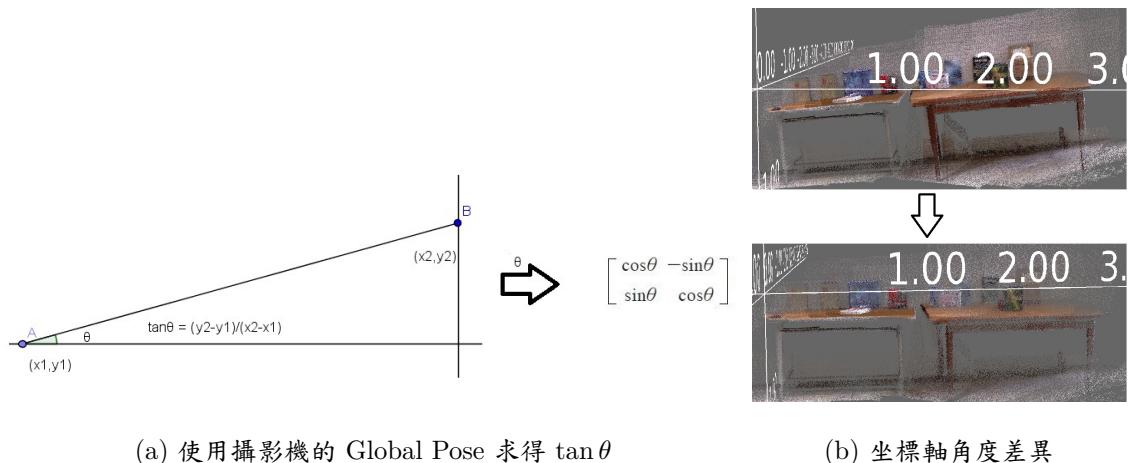


圖 1.3: 將點雲給予界限範圍



(a) 使用攝影機的 Global Pose 求得  $\tan \theta$

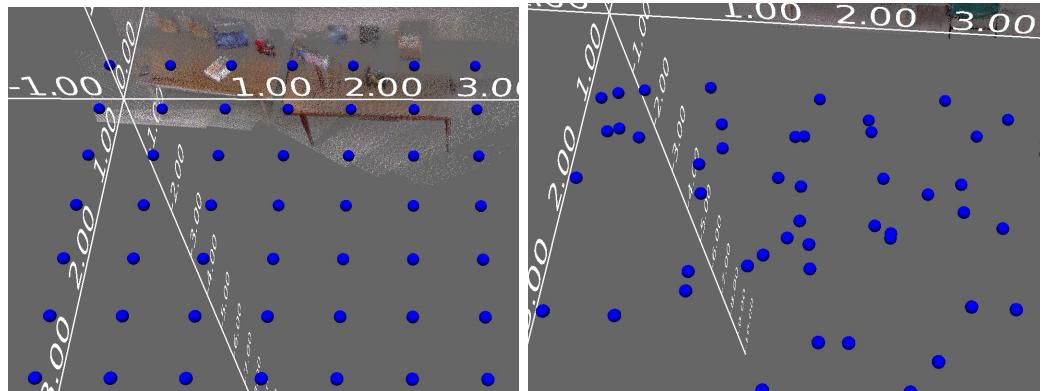
(b) 坐標軸角度差異

圖 1.4: 調整點雲坐標軸角度方法

Global Pose 來算出兩個 Kinect 攝影機角度的  $\tan \theta$ ，求出  $\theta$  之後，將點雲帶入求出來的  $\theta$  旋轉矩陣旋轉至與座標軸平行的角度。調整完角度後，點雲的界線範圍與旋轉角度都與座標軸方向一致，就可以準備虛擬相機的準備工作。

### 1.3 虛擬照相機設置及影像資料庫建置

在建置完環境之後，接下來利用這個章節來描述如何決定虛擬照相機的位置、角度以及虛擬照相機成像的原理。與一般 2D 影像定位方法不同的是，傳統的 2D 影像定位內的影像資料庫，大部分都是利用隨機位置取得影像資料，而在我們的作法是先利用格狀分布設置虛擬照相機的位置，再利用隨機分布的角度來決定照相機拍攝的角度。依照這樣的作法，我們能取得比一般影像定位更多的環境資訊，而這些資訊都是利用點雲所產生的，不必再額外人工存取 kinect 攝影機的影像資



(a) 格狀分布虛擬相機位置

(b) 均勻分布虛擬相機位置

圖 1.5: 在點雲上設置虛擬照相機位置

料，我們所要輸入的資料只需要點雲就可以了。之後我們將分成四個部分描述虛擬照相機的設置：

- (1) 均勻分布設置虛擬攝影機
- (2) 虛擬照相機成像原理
- (3) 根據深度來調整攝影機角度
- (4) 儲存虛擬照相機圖片

### 1.3.1 均勻分布設置虛擬攝影機

上一個章節中，我們完成了實驗環境的建置，也就是點雲環境的資料。在這個章節中為了環境內每個景物都有充分拍攝而取得足夠的特徵點，將虛擬相機位置設置成格狀分布，在每個區塊上設置一個虛擬照相機。格狀分布的好處在於能夠減少相機集中在某處的情形發生，以圖來說，格狀分布會均勻分布在環境內，但隨機分布卻過於集中在圓圈處，再之後實驗會分別進行定位比較。格狀分布作法依據環境而有所改變，為了希望環境內建置 50 部以上的虛擬相機，我們會將環境的長度分成 8 個等分、寬度分成 7 個等分，這樣每個等分都會有一樣的距離間隔，完成 56 部虛擬相機擺設的位置。接下來再隨機分布照相機角度，完成虛擬照相機布置的工作。隨機分布照相機角度比一般 2D 影像定位所建置的資料庫比較有更寬廣的角度。一般影像資料庫可能只針對特定區域的特徵點作取樣，而導致

特定區域內的影像定位效果非常好，但在其他區域卻沒有足夠的影像特徵點資料，使得定位誤差範圍過大，透過均勻分布不會有部分景物或場景沒有被拍攝到，也可以增加定位的覆蓋率。

### 1.3.2 虛擬照相機成像原理

當虛擬相機位置固定之後，接下來利用虛擬相機拍攝照片，透過攝影機的影像角錐來模擬相機的成像，將角錐內 3D 影像投影成 2D 影像，模擬照相機所照出的照片。透過 OpenGL 的坐標系，先將視野調整到虛擬照相機的位置，再利用 OpenGL 中 glFrustum 這個矩陣取得相機影像角錐，這個矩陣目的在於模擬相機光線經過透鏡成像，在投影到平面影像呈現看出來的照片，矩陣表示法如下：

$$glFrustum = \begin{pmatrix} \frac{2near}{right-left} & 0 & \frac{right+left}{right-left} & 0 \\ 0 & \frac{2near}{top-bottom} & \frac{top+bottom}{top-bottom} & 0 \\ 0 & 0 & \frac{far+near}{far-near} & \frac{2far\times near}{far-near} \\ 0 & 0 & -1 & 0 \end{pmatrix} \quad (1.1)$$

將座標轉為齊次座標後，利用攝影機的攝影近裁面 (near) 與遠裁面 (far) 的相似三角形來推出這個矩陣，這個矩陣會將角錐內的影像投影到 2D 平面上，即可完成虛擬照相機的建置。這部分攝影機的焦距設定，以及解析度都參照 Kinect 紅外深度攝影機的參數設定。要解釋如何求出 glfrustrum 矩陣，需要用到三項條件來說明：(1) 證明  $\frac{1}{z}$  為線性關係，(2) 求出位於頂點 Z 座標之間的線性關係，(3) 將 (1), (2) 所求出的公式帶入投影座標求出矩陣關係式。

(1) 證明  $\frac{1}{z}$  為線性關係：

根據圖所示，由相似關係三角形得出的關係式：

$$p = \frac{-n}{z} \times x \quad (1.2)$$

而我們知道直線關係視為  $y = ax + b$ ，將式 (1.2) 帶入直線關係式得出

$$p = \frac{n}{a} \left( \frac{z}{b} - 1 \right) \quad (1.3)$$

利用線性關係， $p_3 = tp_2 + (1-t)p_1$ ，帶入其中得出：

$$\frac{n}{a} \left( \frac{b}{z_3} - 1 \right) = t \frac{n}{a} \left( \frac{b}{z_1} - 1 \right) + (1-t) \frac{n}{a} \left( \frac{b}{z_2} - 1 \right) \quad (1.4)$$

化簡後得出：

$$\frac{1}{z_3} = t \frac{1}{z_1} + (1-t) \frac{1}{z_2} \quad (1.5)$$

(2) 求出位於頂點 Z 座標之間的線性關係：

同一條線相互的點座標關係可以寫為  $\frac{b_3-b_1}{b_2-b_1} = \frac{z_3-z_1}{z_2-z_1}$ ，其中  $z_3$  表示式為：

$$z_3 = \frac{1}{\frac{1}{z_1}t + \frac{1}{z_2}(1-t)} \quad (1.6)$$

將式 (1.6) 帶入關係式中得出

$$b_3 = z_3 \left[ \frac{b_1}{z_1}t + \frac{b_2}{z_2}(1-t) \right] \quad (1.7)$$

(3) 將 (1), (2) 所求出的公式帶入投影座標求出矩陣關係式：

在最後一個步驟中，要把之前求出的關係式都帶到投影座標內，我們假設  $(x, y, z, w)$  為攝影機座標， $(x', y', z', w')$  為投影座標，而  $(P_x, P_y, P_z, P_w)$  為攝影角錐內的座標，對於 x,y 其中關係式可以寫為：

$$x' = \frac{-nx}{z} y' = \frac{-ny}{z} \quad (1.8)$$

將其縮放到可視範圍  $[-1, 1]$  之間，得出：

$$\frac{1 - P_x}{1 - (-1)} = \frac{r - x'}{r - l} \frac{1 - P_y}{1 - (-1)} = \frac{t - y'}{t - b} \quad (1.9)$$

化簡後得出：

$$P_x = \frac{2x'}{r - l} - \frac{r + l}{r - l} P_y = \frac{2y'}{t - b} - \frac{t + b}{t - b} \quad (1.10)$$

帶入  $x', y'$ ：

$$P_x = \frac{2n}{r - l} \left( -\frac{x}{z} \right) - \frac{r + l}{r - l} P_y = \frac{2n}{t - b} \left( -\frac{y}{z} \right) - \frac{t + b}{t - b} \quad (1.11)$$

與真實相機的相機參數相同，再接下來實驗就相機拍出來的照片作對照，做為比較的依據。當我們截取下攝影機的圖片後，會先將角錐內影像的深度作平均，為的是求取平均深度來看虛擬相機取的位置會部會太逼近虛擬環境內的景物，或是虛擬相機位置太靠近點雲邊界，這部分會在之後的章節作說明。

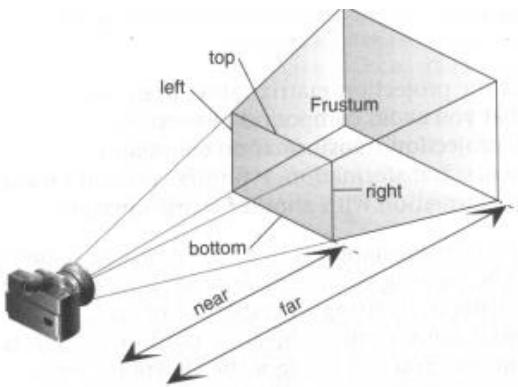


圖 1.6: 攝影機的攝影近裁面 (near) 與遠裁面 (far) 表示



圖 1.7: 根據物體距離鏡頭遠近來調整方位

### 1.3.3 根據深度來調整攝影機角度

當虛擬照相機的圖片擷取出來後，因為拍照的相機深度過淺，而導致拍攝的景物無法辨識，這時候我們利用深度過濾的機制來將照相機取得角度作過濾。一般深度 buffer 分為 z-buffer 與 w-buffer 兩種，先從兩種不同的深度分辨方式作探討：

首先作關於深度的計算，利用四維座標軸  $(x, y, z, w)$  表示三維座標軸  $(x', y', z')$  的點，空間關係的表示法為：

$$\begin{cases} x' = x/w \\ y' = y/w \\ z' = z/w \end{cases} \quad (1.12)$$

根據圖1.6(a)的示意圖表示， $Z_n = \text{near}$  面的 z 範圍， $Z_f = \text{far}$  面 z 範圍，

$w = \frac{2 \times Z_n}{right-left}$ ,  $Q = \frac{Z_f}{Z_f - Z_n}$  所以由 z 座標求得 w 縮放的比例，式子可以寫為：

$$w = \frac{Q \times Z_n}{(Q - Z)} \quad (1.13)$$

z-buffer 是保存經過 glFrustum 投影變換後的 z 坐標，投影後物體會產生近大遠小的效果，所以距離眼睛比較近的地方，z 坐標的分辨率比較大，而遠處的分辨率則比較小。換句話說，投影後的 z 坐標在其值得分布上，對於景物對眼睛的物理距離變化來說，不是線性變化的（即非均勻分佈），這樣的一個好處是近處的物體得到了較高的深度辨識，但是遠處物體的深度判斷可能會出錯。

w-buffer 保存的是經過投影變換後的齊次坐標系中的 w 坐標，而 w 坐標通常跟世界坐標系中的 z 坐標成正比，所以變換到投影空間中之後，其值依然是線性分佈的，這樣無論遠處還是近處的物體，都有相同的深度分辨率，這是它的優點，當然，缺點就是不能用較高的深度分辨率來表現近處的物體。

針對兩種不同的深度 Buffer 比較，因為我們的做法是來判別景物是否距離鏡頭過近，所以在深度判斷上是採用 z-Buffer 的作法，當我們判斷鏡頭與物體距離實際深度小於 80 公分時，我們會將照相機鏡頭角度轉向 180 度，也就是正後方來重新拍攝。

#### 1.3.4 儲存虛擬照相機圖片

當已經決定取好的照片之後，利用虛擬照相機將取出的照片來儲存至影像資料庫，來進行接下來定位的前置作業。虛擬影像儲存是透過虛擬照相機鏡頭裡的每一個像素寫入相片裡頭，主要做法如下。當從 z-buffer 讀出來的深度錯誤時，代表這個像素對應在點雲上是一個黑點或者是說根本沒有點雲的資訊，則以黑色為代表，當深度沒影錯誤時，則代表它具有實際點雲的資料，我們找出點雲對應點的顏色資訊，寫入圖檔裡，這樣即可完成初步的虛擬相片。根據上述的方法，還會遇到透視的問題，就是說原本不應該出現的景物因為深度有誤差，而原本在障礙物之後的物體卻跑在障礙物之前，像是穿透障礙物一樣，例如圖1.8原本不該出現桌子的地方，因為發生了透視的現象而出現了桌子。改進方法為根據周圍的深度來做內插補強。

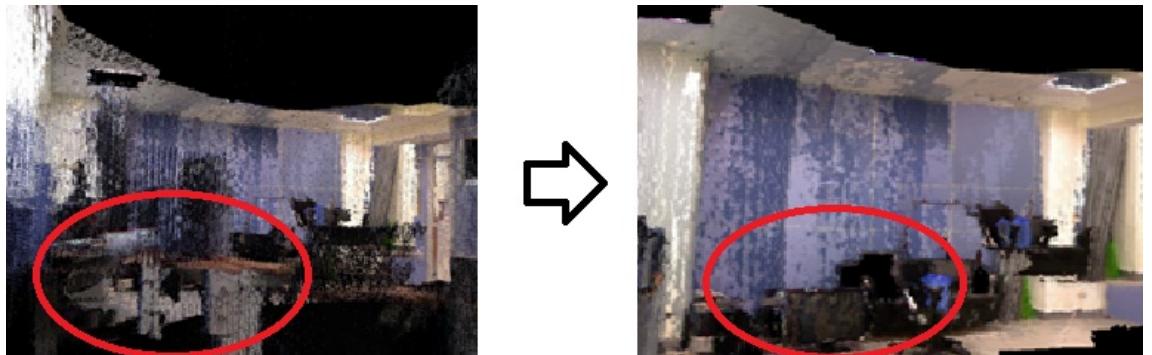


圖 1.8: 內差法補強前後的差異圖

虛擬影像的資料量因環境而變，主要根據 Global Pose 在每個位置上取出相隔 120 度的兩個不同角度的相片，在一般情況下環境中取出 50 點的 Global Pose，所以總共會有 100 張的虛擬相片。藉由這些虛擬相片，我們取得了環境所在內的不同位置與不同角度的資料，比起一般的影像定位資料多出了更豐富的特徵點資訊。之後的實驗可以比較出來，在不同位置以及距離特徵點的遠近對定位會帶來什麼樣的影響。到了最後定位的流程，將介紹虛擬影像的定位方法。

## 1.4 虛擬影像定位

在定位的流程中，利用讀取要定位的圖片，根據定位照片的特徵點比較找出最合適的虛擬影像，再參照虛擬影像所在的虛擬照相機的位置來定位。這樣可以節省利用三角定位 (Triangulation) 的時間，根據之前不同的影像資料庫的建置方法，又增加許多以前傳統影像定位所定位不到的地方，關於這種覆蓋率的數據比較，在之後的實驗方法有詳細的數據可以佐證增加覆蓋率的證明。

在定位的程序上，主要會分成 3 個階段：

- (1) 前置作業處理
- (2) 尋找特徵點並找出最多的特徵點投票選出位置
- (3) 利用虛擬照相機位置來定位

### 1.4.1 前置作業處理

在定位之前，先輸入待定位的照片以及虛擬照相機的位置。虛擬照相機的位置記錄檔格式包含每個虛擬相機的  $X, Y, Z$  座標以及每個攝影機的角度位置，當流程步驟做到特徵點定位時，就會需要參考到虛擬相機的位置。

### 1.4.2 尋找特徵點並找出最多的特徵點投票選出位置

在前置作業完成之後，接下來利用所有資料庫中的照片進行比對，並將每張照片所擁有找到與被定位照片相同的特徵點數量記錄下來。在這裡使用的的方法為尺度不變特徵向量 (Scale Invariant Feature Transform)，簡稱為 SIFT，我們利用 SIFT 找出與相片中相同的特徵點，並將找出的特徵點的數量給記錄下來。關於 SIFT 的作法在之前已經有了相關的敘述，所以可以得知當存在虛擬相片中特徵點的數量越多，代表與所要定位的照片有越密切的關係，當我們在所有照片中選出來擁有最多特徵點數量的虛擬照片時，我們參考這個拍攝虛擬照片的相機的編號，根據編號找出相機所在的位置，再利用這個虛擬相機的 pose 當作初步所在定位的定位位置。

特徵點的分布跟環境景物的分布有密切的關係，在我們的作法上藉由虛擬照片找到距離特徵景物遠的待定位照片，卻可以比一般的照片找出更多的特徵點。利用虛擬照片我們可以有效的找出更多的環境特徵，在之後的定位上不管是覆蓋率或是精準度都有一定程度的提升。

### 1.4.3 利用虛擬照相機位置來定位

最後定位我們利用虛擬相機的位置來當作定位的參考位置，在利用虛擬相機的位置來定位與一般影像定位不同的地方在於三角定位的使用。在這裡我們先解釋一般三角定位的流程：

- (1) 找出特徵點對於鏡頭的夾角
- (2) 利用夾角帶入餘弦定理 (Cosine Law) 求出特徵點所距離鏡頭位置的長度
- (3) 利用已知的長度求出待定位圖片的位置



(a) 根據虛擬相片找出的特徵點



(b) 根據實際相片找出的特徵點

圖 1.9: 特徵點比較差異圖

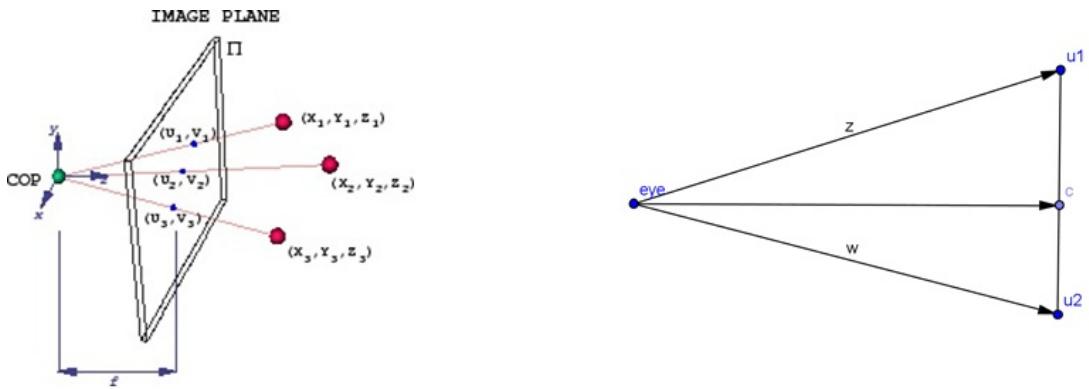


圖 1.10: 相機與特徵點的夾角示意圖

### 找出特徵點對於鏡頭圖片的夾角

在 1.10 當中我們要先求得  $\vec{z}$  與  $\vec{w}$  的長度，當我們知道  $\vec{U}_1$  與  $\vec{U}_2$  之後，帶入下列求解的算式：

$$\begin{cases} \vec{z} = (u_{1x} - c_x)\vec{u} + (v_{1y} - c_y)\vec{v} + \vec{f}d \\ \vec{w} = (u_{2x} - c_x)\vec{u} + (v_{2y} - c_y)\vec{v} + \vec{f}d \end{cases} \quad (1.14)$$

在這之中， $f$  為焦距向量， $d$  為深度。

我們得到  $|z|$  與  $|w|$  的長度，在圖 1.10 我們知道  $|u|$  的長度之後，再帶入餘弦定理求得角度  $\alpha$  的夾角：

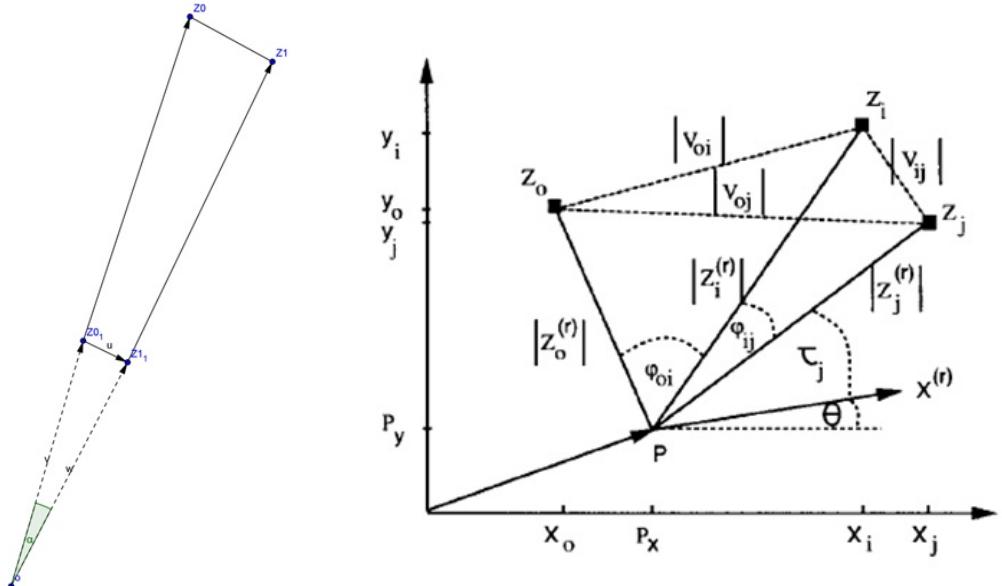
$$|\vec{u}|^2 = |\vec{z}|^2 + |\vec{w}|^2 - 2zw\cos\alpha \quad (1.15)$$

### 利用夾角帶入餘弦定理 (Cosine Law) 求出特徵點所距離鏡頭圖片位置的長度

由圖 1.11 我們可以知道  $\varphi_{oi}$  與  $\varphi_{ij}$  也知道  $|V_{oi}|$ ,  $|V_{oj}|$  與  $|V_{ij}|$  的長度，藉由餘弦定理可以推出下列算式：

$$\begin{cases} |V_{oi}|^2 = |Z_o^{(r)}|^2 + |Z_i^{(r)}|^2 - 2|Z_o^{(r)}||Z_i^{(r)}|\varphi_{io} \\ |V_{oj}|^2 = |Z_o^{(r)}|^2 + |Z_j^{(r)}|^2 - 2|Z_o^{(r)}||Z_j^{(r)}|\varphi_{jo} \\ |V_{ij}|^2 = |Z_i^{(r)}|^2 + |Z_j^{(r)}|^2 - 2|Z_i^{(r)}||Z_j^{(r)}|\varphi_{ij} \end{cases} \quad (1.16)$$

其中 (r) 代表從定位點 P 所觀測出的位置與視角。



(a) 角度  $\alpha$  對於夾角  $\vec{z}$  與  $\vec{w}$   
示意圖 (b) 利用夾角帶入餘弦定理 (Cosine Law) 求出特徵點所距  
離鏡頭圖片位置的長度

圖 1.11: 定位點夾角與長度向量關係示意圖

當我們解出  $|Z_o|, |Z_i|$  以及  $|Z_j|$  之後，根據圖上的座標表示法，我們最後帶入式子 (2.7) 中求解

### 利用已知的長度求出待定位圖片的位置

$$\begin{cases} |Z_o^{(r)}| = (x_o - p_x)^2 + (y_o - p_y)^2 \\ |Z_i^{(r)}| = (x_i - p_x)^2 + (y_i - p_y)^2 \\ |Z_j^{(r)}| = (x_j - p_x)^2 + (y_j - p_y)^2 \end{cases} \quad (1.17)$$

我們利用上述式子整理可得出下列式子：

$$\begin{cases} |Z_o^{(r)}|^2 - |Z_i^{(r)}|^2 = X_o^2 - X_i^2 + 2p_x(x_i - x_o) + y_o^2 - y_i^2 + 2p_y(y_i - y_o) \\ |Z_o^{(r)}|^2 - |Z_j^{(r)}|^2 = X_o^2 - X_j^2 + 2p_x(x_j - x_o) + y_o^2 - y_j^2 + 2p_y(y_j - y_o) \end{cases} \quad (1.18)$$

在依照 (2.7) 式子兩兩相減，可得出六項聯立方程組，(2.8) 為其中的兩項，在式子當中我們求出  $p_x$  及  $p_y$  則為我們想要定位之座標。當然所有的特徵點會超

過 3 點以上，這些將些的餘弦等式利用最小平方法求解，得出我們想要的定位結果。

上面為傳統 2D 平面影像根據特徵點的定位流程，我們根據虛擬影像也可以與待定位照片根據特徵點定位。但是虛擬影像為 3D 投影回 2D 的平面影像，在座標空間表示會面臨到投影所產生的誤差，再者點雲所見出的環境深度因為 Kinect 深度攝影機本身偵測的深度也會產生誤差，由虛擬影像跟平面影像特徵點利用式子求解比平面影像定位求解來的誤差更大。根據這點，我們利用虛擬相機的位置來當參考的定位點，當我們找出最多特徵點的虛擬照片後，我們還是利用虛擬影像作三角定位，當所求的定位點與虛擬相機絕對距離超過 70 公分，我們就利用虛擬相機位置做最後定位點，否則則三角定位的位置則為最後定位完成的結果。

會這樣做的理由基於我們每個相機的 x 軸距離與 y 軸距離是 50 公分，為均勻分布，所以假定最大的平均定位誤差就為  $\sqrt{x^2 + y^2} = 70.7$  公分，當定位超過這個平均誤差距離，我們最後還是捨棄三角定位的結果，改以最多特徵點的虛擬相機位置為最終的成果。

## 第 2 章 定位實驗方法比較分析

### 2.1 實驗目的

為了改善室內定位的覆蓋率以及定位的精準度，我們利用建置虛擬相片的方法來增加照片的範圍及廣度。有了更多的取樣範圍，藉由實驗來跟以前傳統的 SIFT 方法做比較。我們分成兩個實驗環境來說明方法所改善的定位數據：(1) 可以控制的實驗環境、(2) 一般室內定位的環境。

首先製造一個可以控制特徵點數量的環境，在這個環境中我們驗證每個固定距離內根據 SIFT 所涵蓋的特徵點數量作比較，增加與物體的固定距離算出每個距離中的平均定位誤差，再算出定位誤差範圍的覆蓋率與傳統的照片影像定位作比較。在可以控制的定位環境下我們根據這些實驗方法說明改善的成果，再把方法放建築一般實際的室內環境中作比較，最後呈現出改善的平均定位誤差與增加環境所能定位的覆蓋率。為了完成這些實驗，所用到的設備為 Intel Core I5 2.0GHz 的 CPU 與 8 GB 的 RAM，顯卡為了能夠使用 CUDA 平行運算加速，所採用的是 Nvidia 的顯示晶片。

### 2.2 可控制環境定位數據比較

#### 2.2.1 可控制環境實驗方法

建立可以控制的實驗環境主要目的為在可以控制的特徵點環境下與一般平面影像定位做比較，藉由實驗驗證出有更好的定位覆蓋率與更小的平均定位誤差。在根據有限景物數量的環境下，將待定位的照片依距離增加，生成出格狀的位置的照片定位點，每個定位點前後都有固定的間距距離，利用這些待定位照片，分別用三種方法比較實驗結果。

一開始建置實驗環境，我們將這些定位照片分布在 4 公尺 X 5 公尺的環境大小內，而景物的分布在 1.5 公尺 X 0.7 公尺的大小範圍內，Kinect 攝影機放置在距離景物 0.5 公尺前的地方，利用 Kinect 攝影機拍照取得深度照片與待定位的照

片。如圖2.1所示，在兩個控制環境的實驗中，利用不同的定位環境、不同的間距以及不同待定位圖片的數量來做實驗比較。

表 2.1: 控制環境實驗參數設定

實驗設定:	控制環境 1	控制環境 2
待定位照片數量:	45 張	54 張
間距寬度:	0.53m	0.5m
間距長度:	0.68m	0.5m

### 2.2.2 可控制環境實驗數據分析

為了證明在可以控制的環境下有比較好的定位結果，將實驗分成三個部分做討論：

- (1) 特徵點數量分布分析
- (2) 間距定位誤差分布分析
- (3) 定位精準度覆蓋率以及平均定位誤差

#### 特徵點數量分布分析

這個實驗目的是要看出根據景物的距離增加與特徵點數量的關係圖，實驗的作法是先在待定位圖片的第一排中間設為圓心，如圖 2.1所示。根據這個圓心將圓的半徑以 0.5 公尺的長度增加，依據每個圓之間所能覆蓋的特徵點位置做特徵點數量的總和並算出平均值，藉由圖中的趨勢看出特徵點數量的變化。

如圖2.2所示，橫軸表示與景物之間的距離增加，縱軸為特徵點的數量，三條線分別代表 1.2D 隨機排列的攝影機影像位置，2.3D 隨機排列的虛擬攝影機影像位置與 3.3D 格狀排列的虛擬攝影機影像位置，一開始距離越近根據 2D 影像所找出的影像特徵點也越多，符合實際的情況，但之後的趨勢分布呈現卻發現距離越遠平面影像所找出的特徵點減少許多，表示說 2D 平面影像因為受到攝影機位置取樣的關係都侷限在景物附近，導致距離越遠卻沒有好的匹配影像做比對，但是

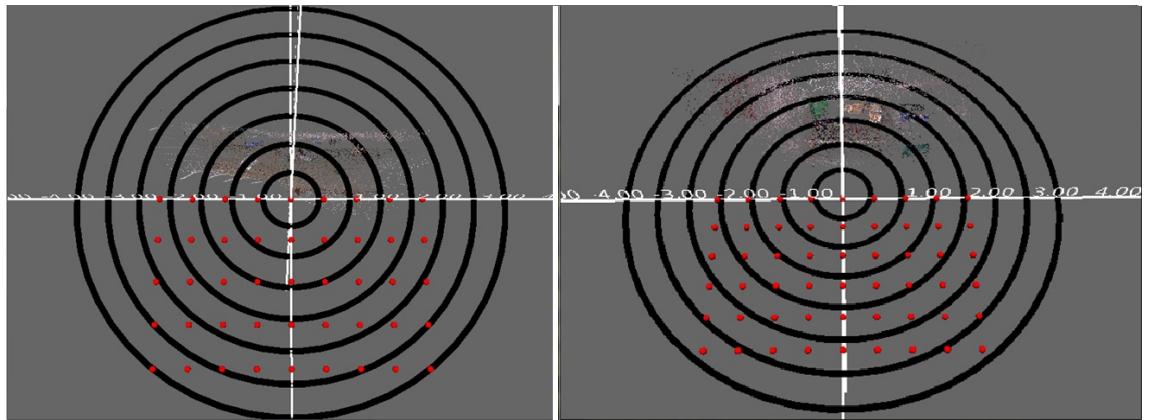


圖 2.1: 依照同心圓的覆蓋算出每個半徑內的每張圖的特徵點平均數，左. 控制環境 1 右. 控制環境 2

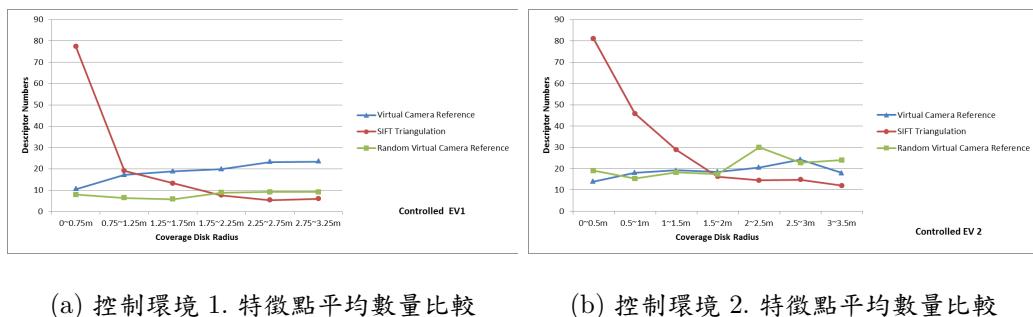
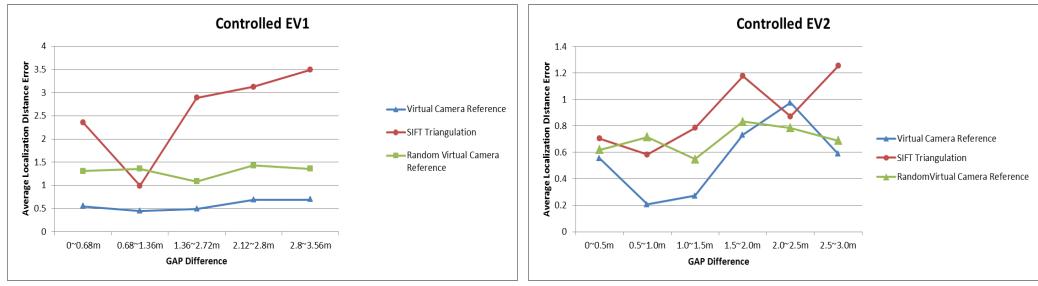


圖 2.2: 特徵點數量趨勢圖

3D 虛擬影像的好處在於可以分布在整個環境區域中模擬平面影像所照出的照片，所以特徵點的數量不會因距離增加而有太大的改變，對於之後的定位精準度也不會因為距離增加而導致定位誤差有明顯擴大的影響。

### 間距定位誤差分布分析

以圖2.3可看出，以每一橫排的待定位圖片的平均誤差分布，根據每一個不同的間距，虛擬相機圖片定位比一般相機所做出的影像定位誤差都有改善，尤其以格狀分布的 3D 虛擬影像改善最為明顯。根據我們的方法，我們想要模擬在不同的角度及位置產生 3D 虛擬影像，比起一般的傳統影像有更多可以做特徵點匹配的相片可供使用。傳統的平面影像定位所做出的資料庫中，對於景物比較遠的照片並沒有資訊提供，只能利用拘限在景物較近的照片可供定位，但是少許的特徵點使得定位誤差更加放大，所以照圖中趨勢來看，距離一增加，定位誤差就會加



(a) 控制環境 1. 間距定位平均誤差比較      (b) 控制環境 2. 間距定位平均誤差比較

圖 2.3: 間距定位平均誤差趨勢圖

大。但是在 3D 虛擬影像不會因為距離的增加，導致定位誤差增加。除了 3D 虛擬影像可以增加更多可以被匹配的相片以外，好的虛擬相機分布，也可以產生更多的特徵點可以被匹配。以隨機分布與格狀分布來說，隨機分布的 3D 虛擬相片雖然有改善，但沒有比格狀分布的虛擬相片改善來的明顯。在我們的方法中，我們定位會參考照相機所在的位置，所以定位的位置都會在相機位置的附近，隨機分布可能會造成某些區域的相機分布過於集中，某些相機卻又過於分散的情況發生。所以在隨機虛擬相機分布的定位其實就跟平面相機分布的定位分布差不多，差別在於相片角度會避開障礙物，但不會均勻分布在環境中；格狀平均分布的攝影機位置，就會均勻地分布在環境內，而對於影像特徵點匹配上比隨機分布來的更有幫助。

### 定位精準度覆蓋率以及平均定位誤差

當發現與景物的距離越遠在 3D 虛擬影像定位並不會影響誤差，整體來看，對定位的覆蓋率也有一定的提升。根據圖所示，在 (控制環境 1 裡的圖) 說明均勻分布的 3D 虛擬相機定位在誤差 0.5 公尺左右有超過 60% 的覆蓋率，但隨機分布的虛擬相機以及 2D 平面影像定位坐落在 20% 左右上下，表示相機的分布影響定位結果的好壞。這也是我們想讓照相機分布平均的原因，再回到圖 2.2(a)來看，在 1.75 公尺以後的隨機虛擬相片所找出特徵點平均數量比起格狀分布虛擬相片所找出的特徵點數量少上許多，而在圖 2.3(a)來看，每個間距的定位平均誤差，格狀分布都比隨機分布的誤差好上許多。所以當以誤差在 0.6 公尺以內的覆蓋率來說，隨機分布的虛擬相片並沒有產生比較好的定位覆蓋率，但以格狀分布的虛擬相片定位覆蓋率與 2D 影像定位覆蓋率相比，結果好上許多。平均定位的誤差以格狀

的虛擬影像為物誤差為最低，2D 影像定位的誤差為最高，改善了整體定位的平均誤差。

## 2.3 一般室內環境定位數據比較

在控制環境下，改善了定位的覆蓋率與精準度，在這章節我們將測試在一般室內環境下的定位測試。我們將室內定位環境分成三種情境：(1) 居家客廳，(2) 居家廚房與 (3) 居家房間，分別在這三種環境下進行定位實驗。在一般室內定位主要進行定位覆蓋率與定位平均誤差測試。

### 2.3.1 一般室內定位環境實驗方法

在室內定位的情況下，拍攝待定位照片的方法依據環境情況而定。待定位照片拍攝方法是依照人能夠活動的範圍作依據，在這些區域進行格點分布拍攝，如圖所示。每張待定位圖片的間距距離為 0.5 公尺，照片數量根據環境大小而定，平均在 20 張上下。虛擬照片依據間距距離，取出不同數量的虛擬相機。因為考量不同環境的景物分布，每組相機位置分別拍攝兩種不同的角度，最後根據虛擬相機拍攝出的照片作影像定位。除了與 2D 影像定位方法做比較之外，分別測試在不同虛擬照片資料數量的定位情形。在圖中，是客廳的點雲環境，如同之前實驗利用點雲環境拍攝虛擬照片，圖為定位平均誤差分布的情形

## Bibliography

- [1] H. Du, P. Henry, X. Ren, M. Cheng, D. B. Goldman, S. M. Seitz, and D. Fox, “Interactive 3D modeling of indoor environments with a consumer depth camera,” *Proceedings of the 13th international conference on Ubiquitous computing - UbiComp '11*, p. 75, 2011.