

第 1 章 前言

1.1 研究背景介紹

機器人相關領域研究在近年來相關廣泛，有許多的研究應用在生活上面，像是定位、導航、視覺感測等，在定位的研究上像是現在的 GPS 定位，在只要有衛星訊號的環境上就可以運用定位系統，在陌生的環境下可以藉由定位了解到自身所在位置相關的資訊。定位資訊的輔助下，藉由路徑規劃，即可完成運送、傳遞的任務，像是車用導航，輔助駕駛到達目的地。室外定位除了利用 GPS 外，也有利用紅外線及雷達等系統，但 GPS 最為廣泛，因為佈及全世界的視野，不須其他儀器輔助。其他像是紅外線及雷達等的儀器，遍布沒有這麼廣泛，較無法運用在日常生活中，較常用在私人用途，像是飛機、軍艦等的導航系統。

室內定位的技術多用於訊號接收定位，需要設備上的支援，像是紅外線、聲納 [1]、藍牙、Wi-Fi [2] 以及 RFID [3] 等室內定位的技術，這些技術的共同缺點都是訊號容易受到環境的干擾，使得無法精確地定出定位位置，誤差範圍過大 (510m)，為了替代訊號在定位上的使用，之後室內定位的研究轉往在影像上的特徵當作定位來源。影像定位藉由影像上的特徵作為定位處理的依據，影像特徵的擷取分為兩大部分，一為區域導向 (Region based)，一為特徵點導向 (feature based) [4]。區域導向代表在兩張相對圖片中，找出相對應畫素的關係性，像是以梯度向量 (Gradient) 強度，或是色彩像素上的強度作為特徵描述。另外一種特徵點導向是以 Lowe [5] 所提出尺度不變特徵轉換 (SIFT) 為前提，所使用擷取特徵的方法。

SIFT 特徵點須具備以下幾個條件：特徵點的識別度高，具有獨特性、容易匹配、對各種的破壞具有不變性。基於這上述三項理由，使得 SIFT 比一般區域導向的特徵更常用在定位演算法中。另外也有其他特徵點擷取的方法，像是 Harris 角點偵測法 [3]，不過其最大的缺點就是本身並非尺度不變，因為本身尺度上的限制，使得 Harris 角點偵測法鮮少出現在影像定位文獻中。SIFT 除了在特徵點擷取上效果優良外，在計算特徵點位置周圍區塊像素的影像梯度 (gradient) 後，求出方向性並與周圍區塊像素的方向統計過後濃縮成特徵點向量，使得 SIFT 比其他區域導向的特徵描述，在對抗破壞與干擾上有更好的效果。

影像定位除了在特徵點的擷取方法的不同上，定位方式也有所差異，像是利用機率方式 Monte Carlo Localization(MCL) 解決定位的問題 [6]，或者是傳統的三角定位 (Triangulation) [7]，都利用特徵描述的不同特性，完成定位。透過影像定位，除了讓機器人了解自己的位置外，也可以運用在建置環境地圖上，像是 SLAM(Simultaneously Localization And Mapping) 關於定位與建置地圖一系列的問題方向，都在影像定位與特徵點擷取的相關技術做研究，使得影像定位在不同的領域上有更多的應用發展。

3D 環境模型建置有許多種的方法，在傳統的方法像是利用雷射掃描深度，利用渲染 (Rendering) 的技術，透過材質 (texture) 拼貼 (mapping) 在立體維度中，達成立體視覺的效果。現在利用深度攝影機 (Kinect) 將深度記錄下來，與照片的像素作對應，使得每張照片的像素都有三維座標與其對應，這種三維座標的紀錄方式稱為點雲 (Point Cloud)。現在許多研究透過點雲應用在立體視覺上面，在 [8] 中，將點雲與機器人作結合，將點雲應用在建置環境地圖或是定位、導航等用途。這種 3D 環境建置的方法可以節省在 Rendering 所花費的計算時間，在者儀器的取得也比雷射或其他 IR 機器來的便利，更可將這些技術應用在日常生活上。

1.2 研究動機

室內定位訊號受到環境上的阻礙，在比較複雜的室內環境下，會因為訊號不穩定而導致定位精準度下降，又受限於設備上的限制，無法落實在日常生活上。我們想避免運用到需要訊號定位的設備，利用日常可隨身攜帶的相機當作我們定位的資訊，藉由影像特徵的幫助下完成室內定位。傳統的訊號定位，如：Wifi、RFID、等，都需要透過設備上存取訊號資料，但在一些特定區域可能環境不能夠放置設備，抑或是會受到其他訊號機器干擾，在室內環境下，我們把研究方向改為影像定位的方面做改進。

影像定位雖有比訊號定位較高的精準度，但是受限於拍照位置與角度上的限制，不能完整的涵蓋室內環境的資訊，在我們的研究想還原定位的 3D 環境，利用近年來常用的 Kinect 深度感測攝影機幫助我們建置 3D 環境，一來可以知道室內環境的全貌，其次也可以藉由環境資訊了解自身所在的位置，增進影像定位的利用與便利性。

1.3 研究目的

為使得影像定位能有更準確且更高覆蓋率的的定位，我們結合了 3D 環境的技術幫助我們達成目的，在重建的定位環境下設置虛擬相機取得照片，藉由均勻分布在定位環境下，取得比相機拍攝更好的影像來源，除了均勻的分布位置外，我們根據虛擬相機所在的位置作深度檢測，當距離障礙物過近的情況下，能夠自動的旋轉相機角度拍攝更好的虛擬影像。我們研究主對於室內影像定位要改善三個方針：

1. 均勻的相機分布取得充分的影像資料
2. 增加定位環境內可定位的覆蓋範圍
3. 改善整體定位的平均誤差

藉由改進這三個方針，使得影像資料能被更有效率的運用，改善因為隨機分布所導致影像資料的不完全，使得影像定位不會受到環境的阻隔而導致定位誤差過大，或是沒有相關資料導致無法定位的情形發生。

1.4 論文架構

在本篇研究中，第二章針對現在的影像定位技術相關研究與 SLAM 問題做探討，藉由之前影像定位研究的方法提出改進的目標，也分析解決 SLAM 問題的方法怎麼應用在建置 3D 環境上，藉由之前的研究參考與改進，提出我們研究的方法。第三章進入我們研究的核心，描述我們如何建置 3D 虛擬環境，並且說明虛擬照片取出的來源及過程，以及最後怎麼運用到三角定位上。第四章針對我們的研究與 SIFT 影像定位做比對，在針對固定放置物品的定位環境下，離物品由近而遠比較定位成果，說明在我們方法改善了那些目標，並針對一般室內環境之下，分別在客廳、廚房、與一般房間做室內定位比較，比較定位的精準度，以及定位在誤差下成功的覆蓋率。在最後一章，針對我們的貢獻以及未來能夠改進的目標做說明。

第 2 章 文獻探討

近年來許多不同的技術運用在室內定位上，像是無線網路、藍芽及 RFID 等相關技術領域，但都受到環境的阻隔與訊號間的干擾，使得無法做到精準的定位。之後在提出相關影像定位的研究，利用影像特徵替代訊號，解決了訊號干擾的問題，但不足的地方在於特定拍照的區域上才能做到有效的定位。在我們的方法裡，提出利用重建 3D 定位環境，將 3D 環境投影成平面虛擬影像，利用虛擬影像比對特徵定位。根據參考 [9] 的室內定位方式分析，根據表 2.1 比較各個室內定位方式所花費的成本做分析，可以得知，影像定位可以花較少的成本得到最高的精確度。在下面相關研究的章節裡，首先在 2.1 介紹先關影像定位的研究，在 2.2 節裡介紹 3D 環境重建的方法，2.3 介紹將 3D 影像投影成平面影像相關技術的應用，最後說明如何將這些研究帶到我們的方法裡。

表 2.1: 室內定位實驗所需設備成本與精確度比較

定位技術:	定位精確度	所需花費	實驗難易度
紅外線:	5-10 m	高	中等
RFID :	1-10 m	高	中等
Wi-Fi :	5 cm-5 m	高	高
手持攝影機:	1- 5 m	低	中等
監控攝影機:	10 cm-1 m	高	低 analysis

2.1 平面影像定位類型分析

定位研究有許多應用在影像視覺與機器人的領域上，不同的地方在於他們如何擷取特徵作應用。像是 Junqiu Wang 在 2004 年利用 Harris-Lapalace 角點偵測，將相片存取資料庫中，根據特徵點多寡投票找出最相似的照片，再將相機紀錄照片的座標位置作矩陣分解，推測出待定位照片的位置。Andreasson 在 2005 年 [10] 利用全景鏡頭大範圍且廣角的特性，把 SIFT 擷取的特徵點作為 Particle Filter 的輸入條件，追蹤照片位置做定位。Timothy Liu 在 2010 年 [11] 將雷射與攝影機作結合，蒐集深度資料作為之後定位參考的依據。藉由定位方法依據的不同，或是

使用輔助感測器提升定位的效果，使得影像定位在許多領域上有不同的運用。

更深入來看影像定位，有不同的定位方式，根據 Bonin-Font 等人的 [12] 的歷年來的研究分析，主要把定位方式分成三大類，三角定位 (Triangulation)，馬可夫鏈 (Markov) 與粒子濾波器 (Particle Filter)。三角定位傳統運用於 GPS 上，依據發射器訊號的角度與位置，透過餘弦定理找出這些位置的關係式，將這些關係式帶入最小平方法 (Least Square) 求解。應用在影像定位來看，把影像特徵點取代為 GPS 的訊號，根據特徵點與相機的角度及特徵點的位置當作三角定位的輸入條件。像是 Dana Cobzas [7]，利用雷射讀取影像的深度，比對 2D 影像的特徵點，根據特徵點與相機的夾角與位置等資訊求取初步的定位資訊，再根據影像深度的關係更新定位點所在的資訊，得到最後的定位成果。在這篇方法將三角定位與影像特徵比對作應用整合，但礙於特徵點比對的方法不夠精細，使得定位精確度受到影響，在我們的方法上使用 SIFT 更精確的特徵點比對方式，使得定位結果獲得改善。

也有的研究將馬可夫鏈 (Markov) 的特性運用在影像定位上，像是 Morita 等人 [13] 利用兩層 SVM(Support Vector Machine) 判斷出影像所在的位置區域，再利用馬可夫鏈根據歷史紀錄作推測出 Support Vector Machine(SVM) 所求的位置是否準確。馬可夫鏈是一套機率統計模型，藉由之前機率分布的狀況，推測出未來可能發生的機率分布。在這裡是增進 SVM 的定位效果，也可以減少 SVM 的訓練成本。大部分馬可夫鏈的研究室運用於導航上，因為需要大量的樣本數來推估機率模型，機器人導航藉由紀錄路徑中的大量影像來推測機率模型相當適合，但是少許的照片樣本數對定位效果的幫助並不顯著，所以對於室內定位小範圍且高密度的定位系統來看，Markov 較不適用於小範圍的定位應用。

粒子濾波器 (Particle Filter) 或稱為 Monte Carlo 定位方法，將拍攝影像存為樣本數，依據影像重複出現的特徵次數給予權重，根據不同的權重值分類，當下一次新的影像輸入時，比對相同的影像特徵所出現的次數，分類到所屬的權重值。在不同的 Monte Carlo 定位研究中，因為所取出特徵點的特性不同，而有不同方式的定位應用。像是在 Jürgen Wolf 等人的研究中 [14]，利用紅、綠、藍三原色的像素統計直方圖當作特徵點，依據不同的權重做分類找出定位的所在地。Monte Carlo 可以做大範圍的地圖定位，但是事先需要地圖的輔助，所以不適用我們的方法。

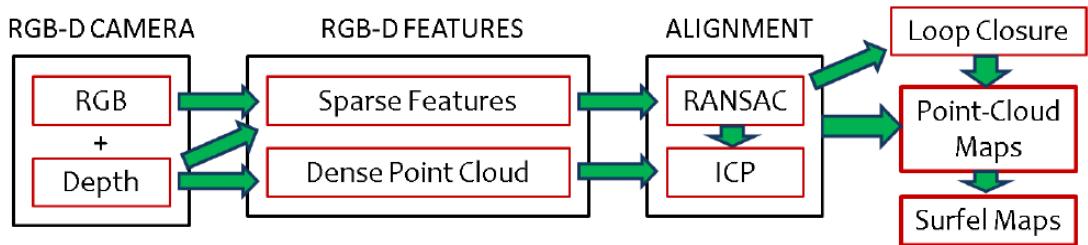


圖 2.1: [15]RGB-D 環境建置流程示意圖

根據三種不同的定位方式，我們找出三角定位最適合於我們的方法上，因為不需要大量的影像資料作輔助，也不用事先地圖作記錄。藉由點雲環境投影至平面的虛擬影像，當作資料庫的資料來源，這些影像位置不需事先的測量，節省人工紀錄的時間成本，也不用將影像事先作分類，減少訓練的成本。

2.2 3D 環境重建技術

3D 環境重建在視覺定位上的研究。在這小節裡，先說明利用 RGB-D 感測器建置 3D 環境的研究，接下來參考相關 Visual SLAM 的研究，以及說明我們所實作調整與最佳化點雲環境的演算法 -Sparse Pose Adjustment。

2.2.1 RGB-D Mapping 3D 環境建置

在 Peter Henry 等人 [15] 的研究中，利用可偵測深度的攝影機來建置 3D 點雲環境。以圖2.2說明整個點雲建置流程分為三大步驟，第一階段首先擷取攝影機畫面，第二階段取出畫面中的特徵與深度作比對，最後階段根據特徵點群先作 RANSAC(隨機一致演算法) 求取最好的轉換矩陣後，再根據 ICP 把點雲之間作最小化誤差。在我們的研究中，參考了 [15] 建置環境的作法，但為了減少運算時間，在我們的方法裡把每個影格作 Iterative Closet Point(ICP) 的時間，移置到最後階段作最佳化。在下面的小節裡，我們將介紹這篇研究中如何將 RGB-D 攝影機擷取的特徵作比對，以及最小化點雲距離誤差的方法。

演算法1說明如何將 RGB-D 攝影機擷取到的影格畫面，做點雲的重合與矯正。在上述的演算法中，透過前一秒的影格與下一秒的影格畫面當作輸入來源，

Algorithm 1 RGBD-ICP:

Input:RGB-D Frame, P_s ;The Target Frame, P_t ;**Output:**Transformation Matrix, t^* ;

- 1: $F = \text{Extract_RGBPoint_Feature}(P_s)$
 - 2: $F_{target} = \text{Extract_RGBPoint_Feature}(P_t)$
 - 3: $(t^*, A_f) = \text{Perform_RANSAC_Alignment}(F, F_{target})$
 - 4: **repeat**
 - 5: $A_d = \text{Compute_Closest_Points}(t^*, P_s, P_t)$
 - 6: $t^* = \arg\min_t \alpha \left(\frac{1}{|A_f|} \sum_{i \in A_f} w_i |t(f_s^i) - f_t^j|^2 \right) + (1 - \alpha) \left(\frac{1}{|A_d|} \sum_{j \in A_d} w_j |(t(P_s^i) - P_t^j) \cdot n_t^j|^2 \right)$
 - 7: **until** ($\text{Error_Change}(t^*) \leq \theta$) or (maxIter_reached)
 - 8: **return** t^* ;
-

先利用 RANSAC 作找出初始化的轉移矩陣，再進入迴圈利用 ICP 作轉移矩陣的最佳化，當誤差值小於等於 θ 值或達到最大迭代次數時，結束演算法。Step 1 與 Step 2 利用 SIFT 作特徵點比對，在擷取影格中對應的特徵點與深度值。當擷取完特徵點與深度值中，帶入 Step 3 中，求取初始的轉移矩陣，透過 RANSAC 的演算法，找出剩下特徵點群中屬於 inlier 的數量，把這個數量當作迭代次數當作其中一個 ICP 的中止條件。RANSAC 除了找出初始的轉移矩陣及 inliers 的數目之外，還可以找出存在這個轉移矩陣中最好的特徵點配對關係。根據這個配對關係，帶入接下來的 ICP 流程中求取點雲距離誤差最小化。

Step 4-Step 7進入到 ICP 的流程裡面，並根據權重找出最佳的轉移矩陣解。首先在 Steps 5 中根據 t^* 轉換矩陣的的轉換後，計算點雲跟點雲之間的配對關係，並找出在點雲 P_s 中跟點雲 P_t 距離最近的點群。當找出距離最近的點群之後，根據 Steps 6 的式子，計算點群間最小距離的誤差，判斷是否小於等於 θ 值當作離開 ICP 的條件。這項式子分成兩部分所組成，第一部分計算 RANSAC 轉換後點雲之間的距離，第二部分計算 ICP 算出來點群之間的誤差值，利用權重分配最適合的轉移矩陣解。為的使算出來誤差衡量具有公平性，在第二部分裡算點雲誤差乘上 normal 值 n_t^j ，使得與第一部分維持一樣的比例， n_t^j 是在點雲 P_t 根據主成分

分析 (Principle Component Analysis) 所算出來的集合值。當達到 Steps 7 的完成條件，最後回傳 t^* 轉換矩陣，利用轉換矩陣將點帶入矩陣中，進行點雲間的重和與調整。

[15] 的方法提供了 RGBD-ICP 演算法，這套演算法幫助了 RANSAC 達到了更精確的重合效果，但是也增加了運算的時間，在我們的方法裡參考了前半部 RANSAC 轉移矩陣的求解過程，之後對於點雲誤差距離最小化的問題，參考了解決 Visual SLAM 問題在誤差距離最小化過程的解法。下面小節先探討 Visual SLAM 問題描述，以及解決 Visual SLAM 問題在建置 3D 環境下有哪些幫助。

2.2.2 Visual SLAM 的問題描述與在建置 3D 環境的應用

Simultaneous Localization And Mapping(SLAM) 描述的是記錄拍攝路徑的連續影像，利用影像的相對關係來製作環境地圖，適合用來做探勘環境，機器人導航、定位等用途。Visual SLAM 將 SLAM 技術與電腦視覺作結合，透過雷射、kinect 或具備深度感測的儀器，達到建置環境地圖的目的，更能夠透過電腦視覺重現原來環境的樣貌。我們利用解決 Visual SLAM 問題的作法，透過拍攝室內環境的照片，利用照片重建定位環境。在下面的小節裡，描述現在那些技術能夠解決 Visual SLAM 的問題，及這些技術怎麼幫助在建置 3D 環境上。

在 Jorge Fuentes-Pacheo 等人的相關研究 [16] 中，根據 Visual SLAM 會遇到的問題，提出了三個分類：地圖封閉檢測 (Loop closure detection)、無法辨識所在位置 (Kidnapped robot) 與多重地圖重合 (Multi-session and cooperative mapping)。地圖封閉檢測描述的問題是，當機器人導航或者是利用攝影機建置地圖時，回到起始點時能否檢測出封閉性。舉一個例子來講，像是建房間地圖時，從大門開始環繞四周牆壁，當回到大門時，能否判斷大門的位置為一開始的起始點，完成地圖的封閉性。在 Kin Leong Ho 等人的研究中 [17] 根據影像辨識的方法，先將影像的特徵比對結果成對記錄成對稱矩陣，再利用 SVD(Single Value Decomposition) 分解進行主成分分析 (PCA)，辨識是否為相識的影像，如果是則完成建置地圖。Eade 等人在 2007 年的研究中 [18]，將他們解決 monocular Visual SLAM 的方法稱為 Graph SLAM，作法是將每個攝影機的位置作紀錄，在這些紀錄點儲存景物特徵點的資訊，將之前紀錄點建置機率模型，當下一個紀錄點在紀錄時，預測這個紀錄點是否出現在之前紀錄所在的位置，或是這個紀錄點

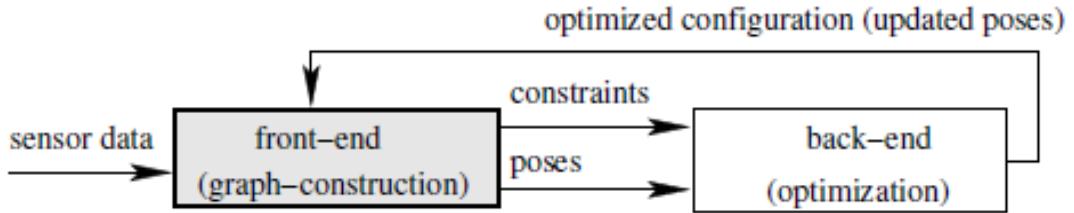


圖 2.2: [22]Graph-based SLAM 系統，透過新的 pose 帶入不斷更新

跟之前的紀錄點是否相關，來偵測封閉性或是減少建置地圖失敗的情形發生。

無法辨識所在位置，問題發生在機器缺乏之前環境位置的資訊，無法辨識目前位置，或是機器突然被轉移到別處，雖然已經有環境資訊，卻還是無法辨識現在的下落，也有的情形像是機器遇到障礙物，因為無法辨識障礙物，導致機器發生錯誤判斷的情形發生。在 Denis Chekhlov 等人的研究中 [19]，提出一個可以容錯且還原 SLAM 錯誤的系統機制。他們在紀錄點建立目次順序，並且在不同的解析度找出 SIFT 特徵點供辨識參考，在缺乏現有環境資訊的狀態下，能從之前的紀錄點推測出現在的位置，達到容錯的效果。

多重地圖重合，當在不只使用一個機器人或是攝影機的情況下，需要將多張地圖對齊拼貼成一張完整的地圖，在許多相關的研究，如 Ho 和 Newman 等人的研究 [17]、及 Gil 等人的研究 [20]，都曾在研究中探討過相關的問題，這些解法大部分都是找出最近的鄰居 (Nearest Neighbour) 或者是聯合相容分枝定界 (Joint Compatibility Branch and Bound) 的解法。這些解法相似的地方在於一開始必須對機器人的位置做很相近的猜測，如果與起始點差異過大，就會發生地圖碎裂或者是重疊的情況產生。

在我們的研究中，根據地圖封閉檢測這個問題作延伸，探討如何在建立點雲環境中能夠封閉重和。有關這類問題，在減少 3D 投影誤差及作出重合最佳化的研究，稱為 Bundle Adjustment(BA) [21]，我們針對這類的演算法，找出相關論文中最適用於我們的研究的演算法 – Sparse Pose Adjustment 演算法。

為了將 3D 模型做出效果能更逼近真實並且能達到封閉，Kurt Konnolige 等人將 BA 的概念運用在 SLAM 上 [22]。以圖 2.2 為例，將輸入的資料透過攝影機的條件予以限制，利用這些限制條件來達到 Graph SLAM 完成最佳化的條件做出了

更有效率的最佳化，稱為 Sparse Pose Adjustment(SPA)，藉由輸入攝影機或是感測器位置以及角度的資訊，作成協方差矩陣 (Covariance Matrix)，利用協方差矩陣為 Levenberg-Marquardt 的限制條件 (Constraint)，再利用迭代法將誤差降到最小，達到最佳化的條件。

2.3 3D 投影平面深度內插研究

在 3D 投影的相關研究上，有的研究將 3D 投影作補強 [23] [24]，對影像作深度內插法以及高斯濾波器 (Gaussian Filter) 作平滑處理，防止投影出來的平面碎裂化，在應用方面可以用來幫助 3D 模型作更好的影像渲染 (image rendering)。在我們的方法中也參考了深度內插法補強的部分，防止因為深度錯誤而導致圖形碎裂的情形發生，增加虛擬影像特徵辨識的能力。

在 Byong Mok Oh 的研究中 [25]，提供了一個很有趣的想法，利用畫素的深度來作視角轉換，透過這些深度作出轉移矩陣 (homography)，模擬這些畫面在視角上的變換，不過因為當時深度偵測的儀器並不普及，只能夠利用陰影模擬深度變化，所以作出來的模擬畫面與真實畫面有些差距，但是利用深度作照片編輯與改變照片視角，是一個很有趣的應用，在本篇研究中把這個想法運用到畫面深度的測量，當影像深度 buffer 過大時代表相機離物體距離過近，我們將相機視角隨機往左邊或右邊轉角度避過障礙物或環境邊緣，使得照片包含更多關於環境上的資訊。

也有的研究把平面影像投影成 3D 深度畫面，像是 Chenglei Wu 在 2012 年 [26] 追蹤影片中的物體的行動，在 non-keyframe 的時候取出追蹤物體取出深度，當在 key frame 時賦予追蹤物體深度，可以做出取出前景的效果，也可以把深度拿來製作 3D 景深。其中被物體擋住的深度利用內插法，將其他 non-keyframe 的深度補齊至 keyframe 中。把影像追中與深度偵測結合再一起，成為一個新的方法提出，但一旦背景複雜，就無法利用內插法補齊差值，所以容錯能力成為此類研究的改進目標。

在接下來的章節參考上述的相關研究方法，在室內影像定位做技術的整合與改良。在我們的方法利用內插法補族缺陷的照片深度，並參考照片的深度決定虛擬相機拍攝的角度，最後利用影像資料庫內的照片做定位比對。我們使用定位的

方法為三角定位，取決於不需要額外的訓練成本，在藉由深度儀器的輔助下，找出比一般照片更充足的影像特徵，也彌補拍攝影像位置取樣的不足，最後與傳統影像定位做比較，針對覆蓋率與精準度對室內環境做定位測試分析。

第 3 章 虛擬影像定位方法

為了提升室內定位的覆蓋率與精確度，我們方法在於增加影像定位取樣的資料，與改進傳統相機角度取樣的範圍。傳統室內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。本篇研究利用 Kinect 採集深度資料，藉由 RGB-D 點雲中透過虛擬相機拍攝相片，將相片集合成影像資料庫，與待定位照片比較求出特徵點角度及位置關係作定位。

3.1 虛擬影像定位流程

整套虛擬影像室內定位作法我們分成三大步驟來敘述：(1). 採集深度資訊及虛擬環境建置，將點雲資料重合形成虛擬環境提供資料蒐集 (2). 虛擬照相機設置及影像資料庫建置，拍攝虛擬相片提供定位環境資訊 (3). 虛擬影像定位，由相片中的特徵點對應關係求得定位位置。圖3.1為整體虛擬影像定位過程，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，在環境中來作格狀均勻分布 (Grid Permutation) 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，藉由規律的相機位置分布取得我們所需要的定位資料。虛擬相片的設置在角度上的調整由深度過濾相機角度，避免相機與景物過近導致無法辨識特徵的情形發生，根據相機位置規律分布與角度篩選機制，取得比一般平面影像定位更適合的角度與更全面的覆蓋，取出良好的定位資訊，減低之後定位所造成的誤差。

3.2 深度資料採集與建立虛擬環境

建立 RGB-D 點雲環境取得定位資訊，為我們研究方法中的第一步，此舉為了增加定位資訊中的完整性。平面影像定位中的相機的位置分布通常是隨機分布，而且需要人工紀錄相機位置與角度，如果能夠自動完成定位資料庫與增加相片的定位資訊，可以改善相片採集與建立資料庫的效率，對於定位品質的改善也有幫助。虛擬環境建立的方法首先從 Kinect 攝影機蒐集完深度照片後，利用

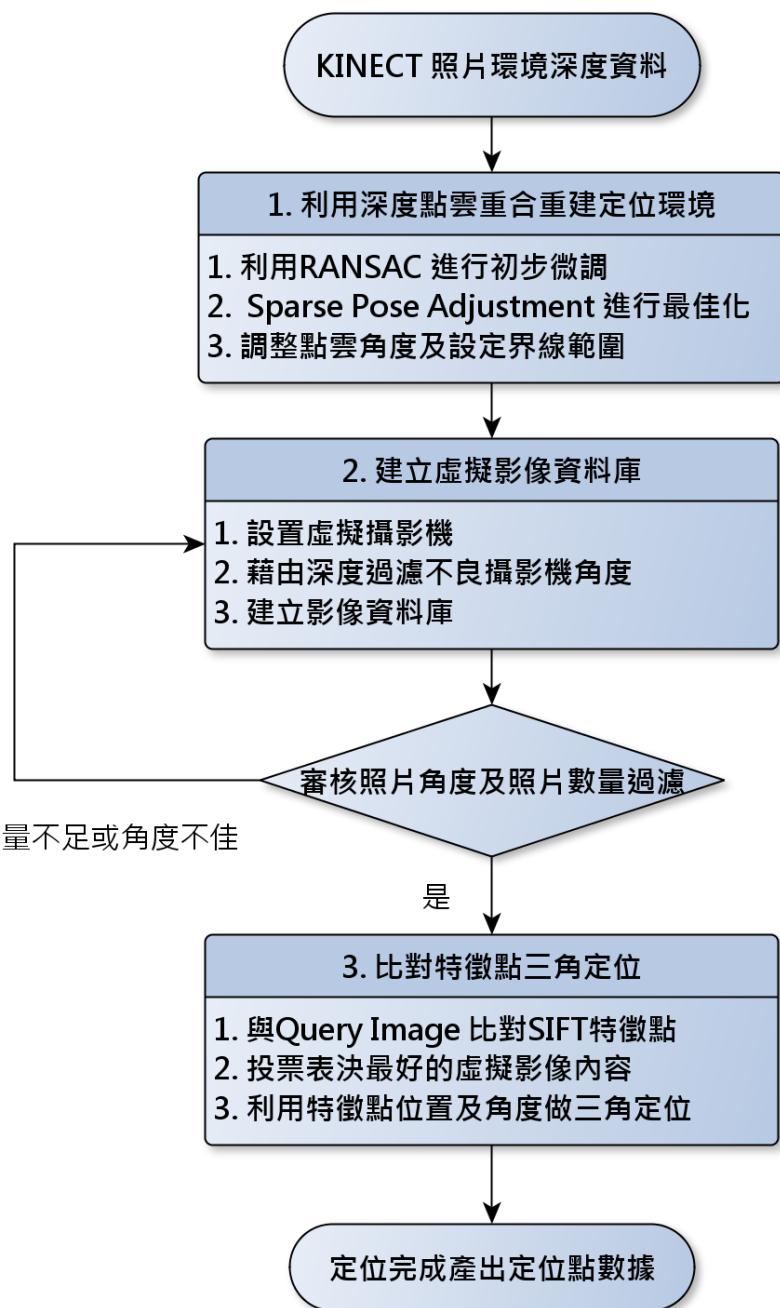


圖 3.1: 3D 點雲環境座內定位整體流程圖

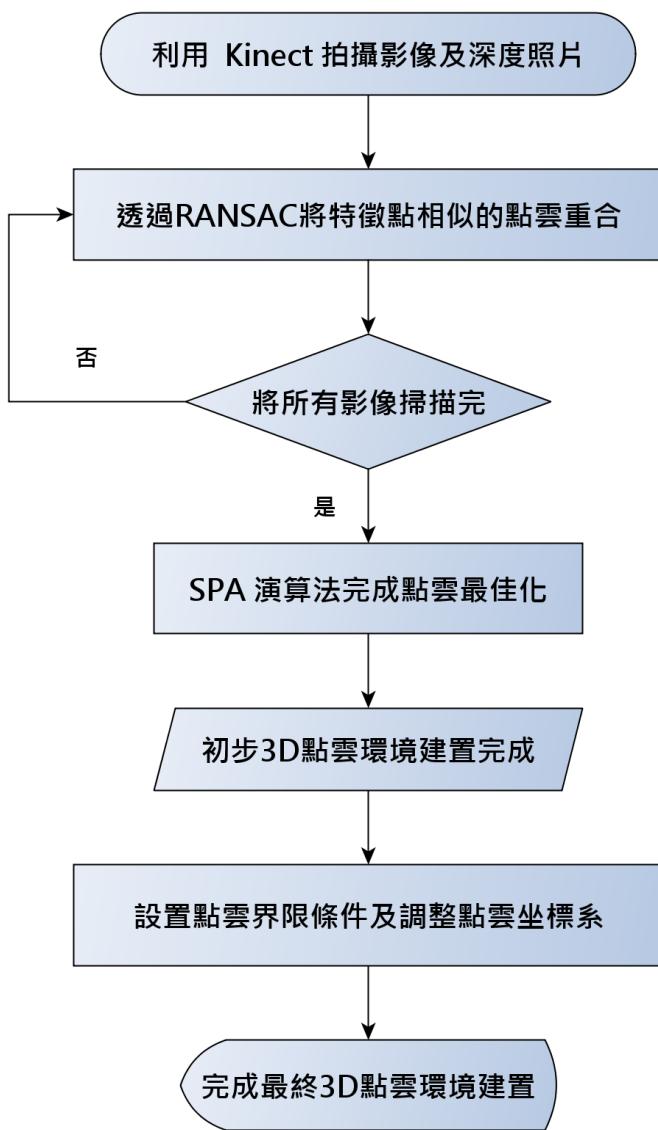


圖 3.2: 室內環境重建方法流程

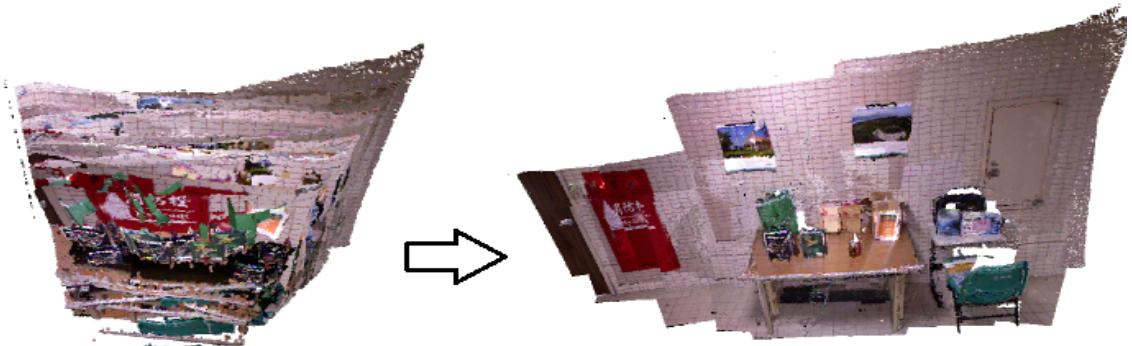


圖 3.3: 透過 SPA 演算法幫助 RGB-D 點雲最佳化，圖左為處理前，圖右為處理後

RANSAC 比對作深度資料重合後作成初步的 RGB-D 點雲建置，之後透過 SPA 演算法完成最後點雲整體最佳化。RANSAC 演算法只能做相對位置的對齊，會發生起始位置及最後位置無法重合的情況。SPA 演算法解決了 RANSAC 只能區域性對齊的問題，我們參考之前文獻探討中描述 Visual SLAM 問題中解決地圖封閉檢測的作法，將每個拍攝點的位置做紀錄找出最佳的重合路徑。重建環境方法我們分成下列幾個步驟：

- 將每張照片經過 SIFT 比對後找出相似的對應特徵
- 將這些對應特徵點帶入 RANSAC 演算法中作為輸入條件，其中包括對應點的特徵與角度以及相機拍攝位置與角度，設置重複計算的次數
- 出來的點雲資料為區域之間的重合，沒有經過整體最佳化，會發生不一致 (non-consistent) 的情形，如圖 3.3 所發生的情況
- 透過 SPA 將點雲做整體的最佳化，改善的情況如 3.3 所示
- 將點雲做最後微調的步驟，找出虛擬攝影機可以設置的範圍區域當作界線範圍，調整點雲角度與齊次坐標系平行

3.2.1 RGB-D 資料採集

資料採集的部分，首先利用 Kinect 在環境中拍攝照片，取得照片深度後透過 RANSAC 重合這些相片位置，製作初步的點雲。深度照片取得的作法類似之

前 [27] 使用的方法，在環境中對每個景物做連續性的拍攝，拍攝出來的照片都需要有一些相似之處，並環繞整個定位環境確保照片的組成具有連續性。當相片包含的景物越多顏色越豐富所代表所包含的特徵點越多，豐富的特徵點數量在做 RANSAC 之後會有更好的重合效果。對於 RANSAC 而言，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，不會因為順序錯誤而導致點雲破碎沒有一致性。RANSAC 怕影像排列的順序出錯，當順序不正確時會無法找出影像位置的相對關係，在之後點雲重合所做出的景物位置會和實際環境中出現的景物位置相差甚遠。因此拍攝照片在 RGB-D 資料採集的步驟中是影響最大的因素，其中因為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，我們要依據這些照片位置與深度關係帶入 RANSAC 計算出重和的位置。在這之中會利用到尺度不變特徵向量 (SIFT) 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齊次坐標系中在正確的位置中重合。作法是將每張採集圖片找出來的特徵點作配對，求出這些配對關係後，將這些配對作最小平方法 (Least Square Error) 求出想要的平面，根據不同平面帶入 homography 作出轉移矩陣 (Transformation Matrix)，最後得到相片位置的絕對關係，稱為 Global Pose。Global Pose 包含照片三維座標以及照片在當時拍攝的角度。在之後調整點雲角度以及制定點雲邊界都需要 Global Pose 的資訊，最後的影像定位過程，我們也需要 Global Pose 當作相片中特徵點的起始位置來協助定位。在初步的點雲建置完成後，最後我們利用之前文獻所提到解決地圖封閉的方法，透過 SPA(Sparse Pose Adjustment) 將點雲作最後的調整。

RANSAC [28] 由 1981 年被提出，大多運用在電腦視覺上，用來預測數學模型有可能分布的狀況，透過迭代的方法使得預測的精準度增加。RANSAC 的流程分為下面幾個步驟：

1. 設定隨機挑選的 inlier 數量 n ，以及迴圈的重複次數 k
2. 從所有的匹配中隨機挑選 n 個作為 inliers，數量至少能夠計算出轉移矩陣

3. 以 Levenberg-Marquardt 演算法從 inliers 計算出投影矩陣 R_{in}
4. 把 inliner 之外的匹配特徵帶入 R_{in} ，計算投影之後匹配特徵的距離，把距離小於設定門檻值得關配都加入 inliers
5. 紀錄所有 inliers，並比較其總數是否大於現在的最大值，如果是則更新最大值
6. 重複執行 k 次步驟 2. 到步驟 5.，最後留下的最大值就是過濾好的特徵點，再利用 Levenberg-Marquardt 從這些過濾好的批被去計算出最佳轉移矩陣 R_{best}

至於 n 和 k 如何決定，可以假設每個特徵匹配為良好的機率設 W，因此一開始隨機挑選的 n 個匹配的機率為 W^n ，其中設某個環節出錯的機率為 $1 - W^n$ ，而只要有一個是錯的，就找不到最佳的轉移矩陣，定義 p 為重複執行 k 次之後出現出錯的機率，至少選到一次 n 個都為良好匹配的機率即為 $1 - p$ ，可表示為：

$$1 - p = (1 - w^n)^k \quad (3.1)$$

$$p = 1 - (1 - w^n)^k \quad (3.2)$$

$$k = \frac{\log(1 - p)}{\log(1 - w^n)} \quad (3.3)$$

從 (3.2) 可看出，若希望 p 要盡可能大，則 k 要夠大 (3.3)，然而 k 越大表示所需計算的時間也越長，因此要先假設合理的 w 值，定義一個至少足以計算轉移矩陣的 n 值，來衡量 p 值與 k 值。

根據 RANSAC 所做出初步的點雲環境中，我們針對照片與照片之間相似的位置作重和，但是做出來的點雲之間會發生不一致的情形發生，這時候需要對原始深度照片拍攝的路徑作位置的最佳化。利用 SPA 演算法 [22]，這套演算法幫助我們減少點雲環境中不一致的情形發生，圖3.4為經過最佳化最後所產生的點雲環境。

3.2.2 座標系角度調整及制定邊界範圍

前三個步驟將點雲環境完成之後，在這個環節制定點雲的邊界範圍 (Bounding Box) 及調整點雲坐標系的角度，像圖3.5所示。這個流程的目的是為了讓虛擬照相



圖 3.4: 初步建置好的點雲環境

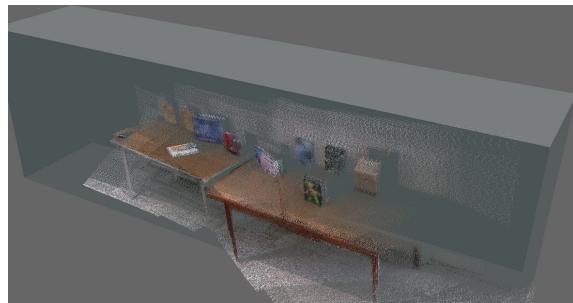


圖 3.5: 將點雲給予界限範圍

機能夠被擺放在定點雲的邊界範圍內，不會發生照相機放置在點雲環境之外而無法產生出有效的虛擬影像提供參考。尋找邊界範圍的具體做法如下，當讀入整個點雲之後，找點雲中最大及最小的 X 及 Y 座標，求出位於邊界範圍的頂點座標，最後將座標值相減求出的長度即為邊界範圍的長寬。有了這些長度之後，就可以知道整個點雲環境所在的位置以及長寬的距離為何，在之後設置虛擬相機位置可防止相機坐落在散布點雲以外的位置，使得虛擬照片無法提供相關的定位資訊。

在制定點雲的邊界範圍之後，點雲可能會因為之前 Kinect 相機位置的 Global Pose 歪斜分布導致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。這個步驟為了防止虛擬相機在照相時因點雲角度歪斜使得虛擬相機與真實相機的角度差異過大，減少對應的特徵點的配對關係而使得定位誤差產生。具體解決的辦法以圖3.6為例，觀察點雲是朝向哪個方向歪斜，運用 Global Pose 的相對位置求出 Kinect 攝影機角度的 $\tan \theta$ ，之後將點雲帶入求出來的 θ 的旋轉矩陣旋轉至與三維座標軸平行的角度，完成點雲角度調整。點雲調整的相關工作皆完成之際，就可以準備設置虛擬相機與建置資料庫的工作。

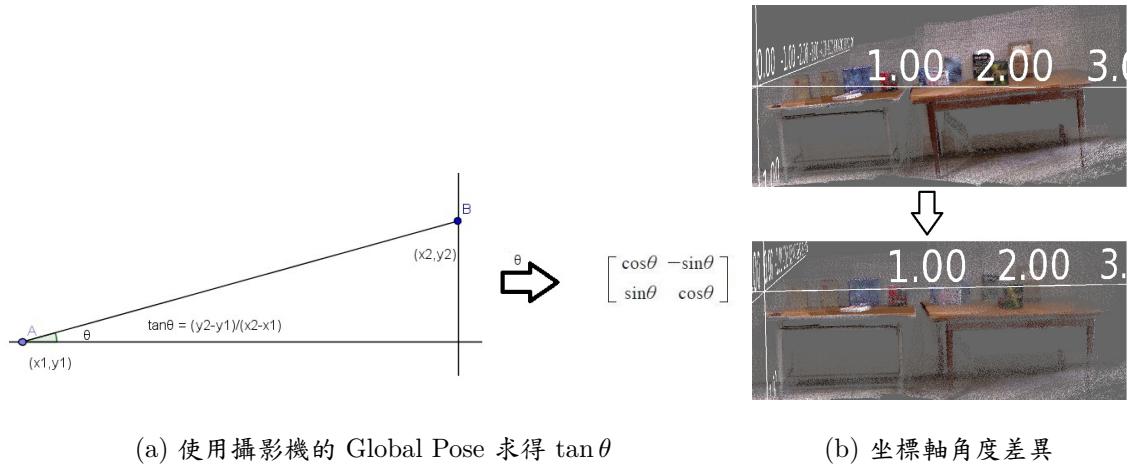


圖 3.6: 調整點雲坐標軸角度方法

3.3 虛擬相機設置及影像資料庫建置

在建置完環境之後，本章節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。與一般平面影像定位方法不同的是，一般平面影像定位的資料，絕大多數都是利用隨機位置取得影像資料，但在我們的研究中藉由格狀分布設置虛擬照相機的位置，再隨機分布照相機拍攝的角度。依照均勻設置相機位置的方式，能夠取得比一般影像定位更完整的環境資訊，這些資訊都是利用點雲所產生的，不必額外拍攝相片與提供相片位置資訊作輔助。圖3.7為拍攝虛擬相片與建置影像資料庫的流程，在本章節裡分為四個部分描述虛擬照相機的設置流程：

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

3.3.1 虛擬相機位置分布機制

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能被完整拍攝而取得足夠的特徵點，所以將虛擬相機位置設置成格狀分布，在每個區塊上設置虛擬相機。格狀分布的好處在於能

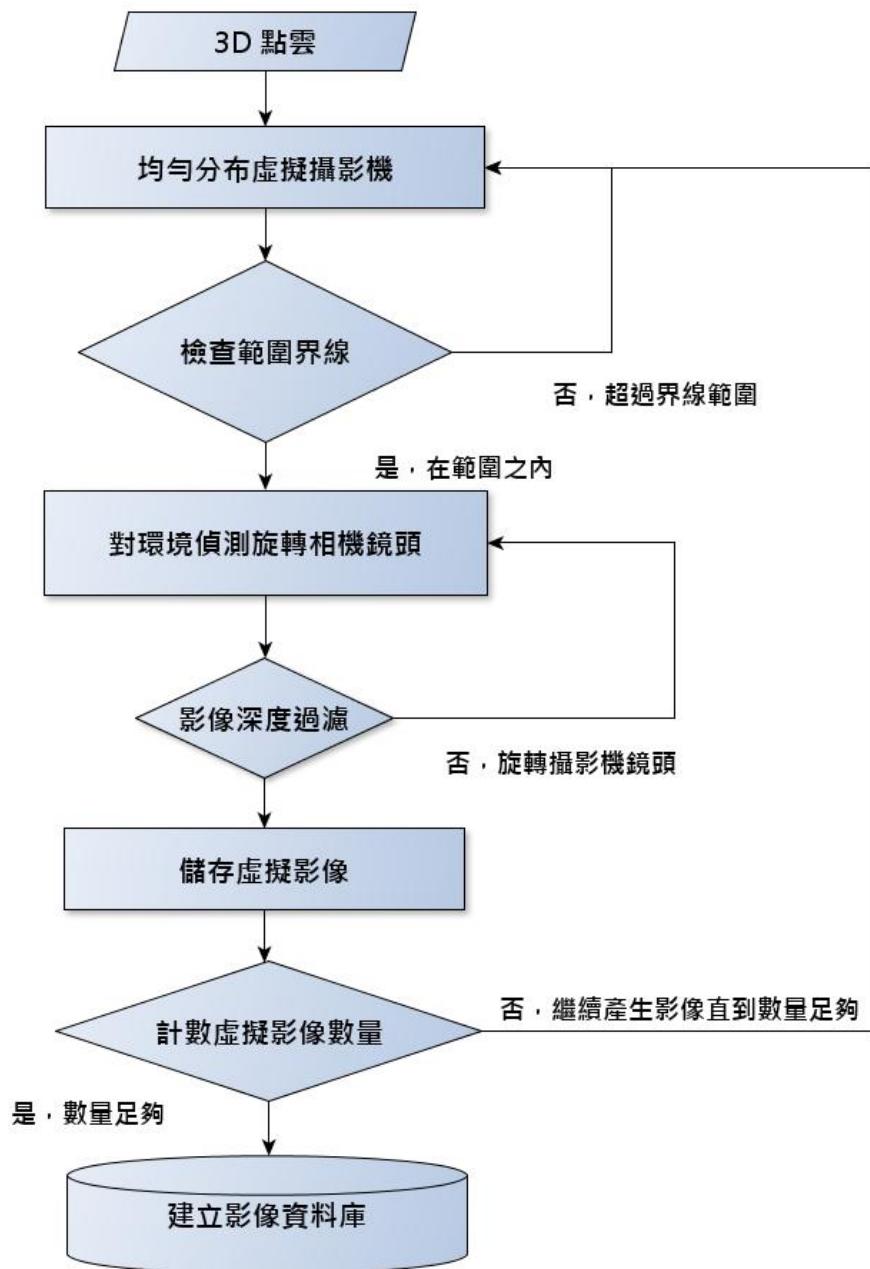
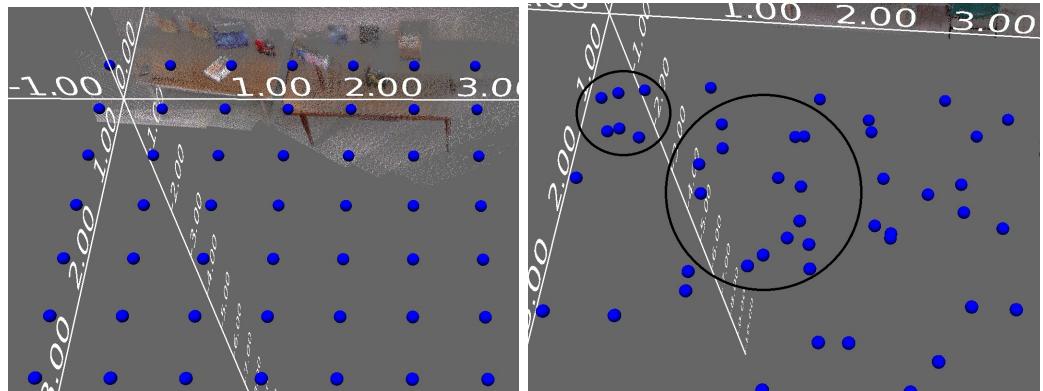


圖 3.7: 影像資料庫建置流程



(a) 格狀分布虛擬相機位置

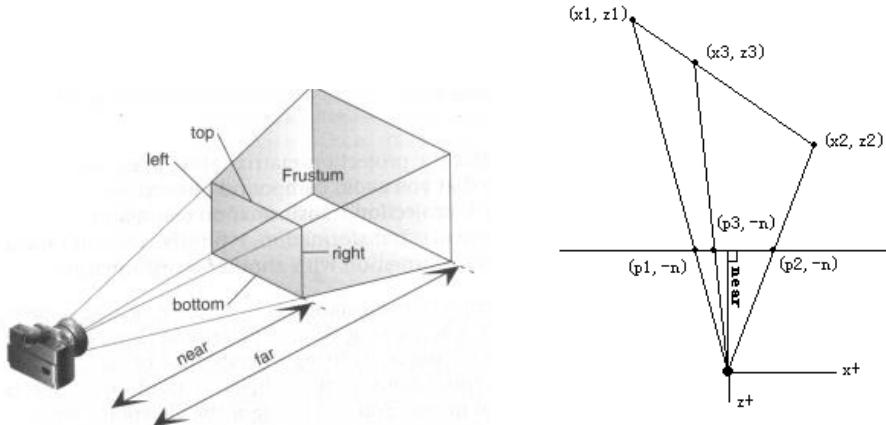
(b) 隨機分布虛擬相機位置

圖 3.8: 在點雲上設置虛擬照相機位置

夠減少相機集中在某處的情形發生，像是以圖3.8(a)為例，格狀分布中每個相機都有固定位置做隔。但圖 3.8(b)相機隨機分布位置卻過於集中在黑色圓圈處，使得每個相機位置間距都不同，導致有些地方相機分布過於分散，但有些地方相機位置分布卻過於集中。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，可以完成 56 部虛擬相機擺設的位置。一般影像定位的資料來源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，藉以增加定位的覆蓋率與改善定位的精確度。

3.3.2 虛擬相片成像原理

當虛擬相機位置固定之後，接下來拍攝虛擬相片，透過虛擬相機的影像角錐來模擬相機的成像，取出影像角錐內範圍的 3D 點雲，投射至平面模擬真實相機所拍出的相片。在這小節裡介紹虛擬相片的成像原理，透過齊次坐標系，將視野調整到虛擬照相機的位置，再利用相機成像的 perspective 矩陣取得相機影像角錐，這個矩陣目的在於模擬相機取得視野範圍內的影像，其矩陣表示法如式子 (3.4) 表示：



(a) 攝影機的攝影近裁面 (near) 與遠裁面 (far) 表示

(b) 相似三角形表示法

圖 3.9: glFrustum 矩陣圖示說明

$$glFrustum = \begin{pmatrix} \frac{2near}{right-left} & 0 & \frac{right+left}{right-left} & 0 \\ 0 & \frac{2near}{top-bottom} & \frac{top+bottom}{top-bottom} & 0 \\ 0 & 0 & -\frac{far+near}{far-near} & -\frac{2far\times near}{far-near} \\ 0 & 0 & -1 & 0 \end{pmatrix} \quad (3.4)$$

將三維座標轉為齊次座標後，如圖3.9(a)利用攝影機的攝影近裁面 (near) 與遠裁面 (far) 的相似三角形來推導出矩陣，這矩陣會將角錐內深度在 $[-1, 1]$ 之間的點雲投影至平面。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 perspective 矩陣，需要用到兩項條件來說明：(1) 證明 $\frac{1}{z}$ 為線性關係，(2) 將 (1) 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3.9(b)所示，由相似關係三角形得出的關係式：

$$p = \frac{-n}{z} \times x \quad (3.5)$$

而我們知道直線關係式為 $y = ax + b$ ，將式 (3.5) 帶入直線關係式得出

$$p = \frac{n}{a} \left(\frac{z}{b} - 1 \right) \quad (3.6)$$

由圖3.9(b)可得知 (p_1, p_2, p_3) 在同一條線上，利用其線性關係，將式 (3.6) 帶入 $p_3 = tp_2 + (1 - t)p_1$ 得出：

$$\frac{n}{a} \left(\frac{b}{z_3} - 1 \right) = t \frac{n}{a} \left(\frac{b}{z_1} - 1 \right) + (1 - t) \frac{n}{a} \left(\frac{b}{z_2} - 1 \right) \quad (3.7)$$

化簡式 (3.7) 後得出：

$$\frac{1}{z_3} = t \frac{1}{z_1} + (1 - t) \frac{1}{z_2} \quad (3.8)$$

在證明 $\frac{1}{z}$ 為線性關係後，接下來由 (1) 所推導的結果帶入 perspective 攝影角錐中

(2) 將 (1) 所求出的公式帶入投影座標求出矩陣關係式：

在這個階段中，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，以圖3.9(a)為例，t=top, l=left, r=right, b=bottom，對於 x,y 其中關係式可以寫為：

$$x' = \frac{-nx}{z} \quad y' = \frac{-ny}{z} \quad (3.9)$$

帶入攝影角錐內的投影座標 (P_x, P_y, P_z, P_w) ，將其縮放到可視範圍 $[-1, 1]$ 之間，得出：

$$\frac{1 - P_x}{1 - (-1)} = \frac{r - x'}{r - l} \quad \frac{1 - P_y}{1 - (-1)} = \frac{t - y'}{t - b} \quad (3.10)$$

化簡式 (3.10) 後得出：

$$P_x = \frac{2x'}{r - l} - \frac{r + l}{r - l} \quad P_y = \frac{2y'}{t - b} - \frac{t + b}{t - b} \quad (3.11)$$

帶入 x', y' 在式 (3.9) 中所得到的結果：

$$P_x = \frac{2n}{r - l} \left(-\frac{x}{z} \right) - \frac{r + l}{r - l} \quad P_y = \frac{2n}{t - b} \left(-\frac{y}{z} \right) - \frac{t + b}{t - b} \quad (3.12)$$



圖 3.10: 根據物體距離鏡頭遠近來調整方位

已知 P_Z 與 $\frac{1}{z}$ 呈現性關係，設 $P_z = \frac{a}{z} + b$ ，求 a 與 b。已知兩點 $(-n, -1), (-f, 1)$ 再近裁面中，所以得出下列結果：

$$a = \frac{2nf}{f-n} \quad b = \frac{f+n}{f-n} \quad (3.13)$$

$$P_z = \frac{2nf}{f-n} \left(\frac{1}{z} \right) + \frac{f+n}{f-n} \quad (3.14)$$

把 P_x, P_y 與 P_z 轉成齊次坐標系得出：

$$\begin{cases} -zP_x = \frac{2n}{r-l}x + \frac{r-l}{r+l}z \\ -zP_y = \frac{2n}{t-b}y + \frac{t+b}{t-b}z \\ -zP_z = -\frac{2nf}{f-n} - \frac{f+n}{f-n}z \\ w = -z \end{cases} \quad (3.15)$$

上述為 perspective 矩陣式子所推導的過程，我們將攝影角錐內的點雲投影成平面，存取虛擬影像。在存取完虛擬影像後，根據物體距離攝影機遠近或太靠近點雲邊界，判斷相機位置是否會太逼近點雲環境內的景物或太靠近點雲邊界。如何判斷利用影像內的平均深度，決定相機角度的選擇是否適當，增加資料庫內影像定位資料的關聯性，這部分會在下面的章節作說明。

3.3.3 攝影機角度過濾機制

在虛擬相機的圖片擷取出來後，因為拍照的相機深度過淺或者位於點雲的邊界範圍上，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 的表示方式分為 z-buffer 與 w-buffer 兩種，我們從兩種不同的深度分辨方式作探討：

關於深度的計算方式，利用齊次座標軸 (x, y, z, w) 表示三維座標軸 (x', y', z') 的點，以圖3.9(a)為例，t=top, l=left, r=right, b=bottom，其空間關係的表示法為：

$$\begin{cases} x' = x/w \\ y' = y/w \\ z' = z/w \end{cases} \quad (3.16)$$

根據圖3.9(a)的示意圖表示， $Z_n = near$ 面的 z 範圍， $Z_f = far$ 面 z 範圍， $w = \frac{2 \times Z_n}{right-left}$ ， $Q = \frac{Z_f}{Z_f - Z_n}$ 由 z 座標求得 w 縮放的比例，式子可以寫為：

$$w = \frac{Q \times Z_n}{(Q - Z)} \quad (3.17)$$

z-buffer 是保存經過 perspective 投影矩陣投影後的 z 坐標，投影後的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 坐標在其值的分布上，從景物對眼睛的物理距離變化為非線性變化（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，所以投影至平面之後，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，則缺點就是不能用較高的深度分辨率辨識近處的物體。

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 0.9 時，相機鏡頭會轉向 180 度，也就是正後方來重新拍攝，避免鏡頭距離物體過近或是拍攝到點雲邊界，導致影像模糊或殘缺的情形發生，在此也增加影像資料庫內定位資料的關聯性。

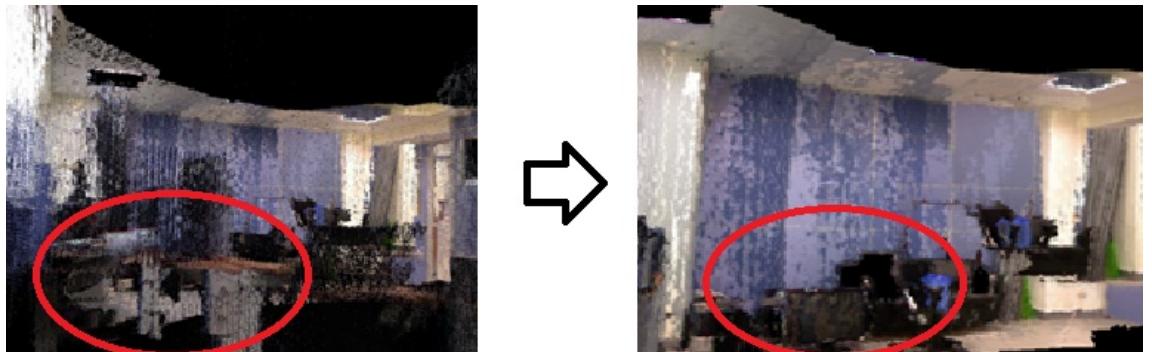


圖 3.11: 內差法補強前後的差異圖

3.3.4 儲存虛擬相機影像建立資料庫

當決定虛擬相機拍攝的位置與角度之後，拍攝虛擬影像將資料儲存至影像資料庫中，來進行影像定位的前置作業。虛擬影像儲存是透過虛擬相機鏡頭裡的每一個像素寫入相片裡頭，主要作法如下。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，以黑色作為代表，在深度辨識沒有錯誤有值讀入時，則代表它具有實際點雲的資料，我們找出點雲對應的顏色資訊，寫入圖檔裡，完成初步的虛擬相片。根據上述的方法，會遇到透視的問題，就是說原本不應該出現的景物會出現在正確位置之前，像圖3.11在圓圈中所表示，原本在之後的物體跑在正確位置之前，像是穿透障礙物一樣，圓圈中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。我們所改進的方法為根據周圍的深度來做內插補強。

虛擬影像的資料量因環境而變，主要根據虛擬相機的 Global Pose 在每個位置上相隔 120 度拍攝兩張不同相片，一般情況下點雲環境會有 40 組以上的虛擬相機位置，所以總共會有 80 張的以上的虛擬影像。藉由這些虛擬影像，取得定位環境在內的不同位置與不同角度的景物資料，根據先前所描述的相機設置方式與角度上的篩選，比起一般的平面影像定位資料庫多出了更豐富的特徵點資訊。之後會在室內定位實驗證實定位改善的成果，根據相機位置設置的方式以及距離特徵點的遠近會對定位帶來什麼樣的影響，留在實驗部分進行討論。在定位方法的最後一個章節，介紹如何利用虛擬影像特徵點進行三角定位。

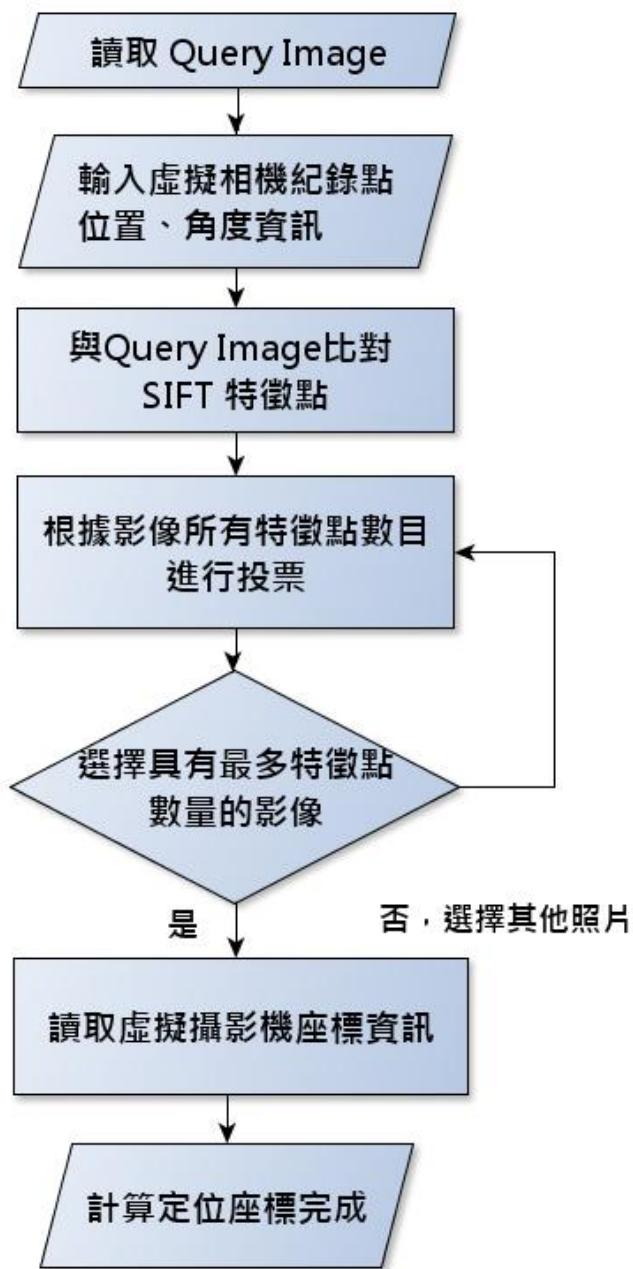


圖 3.12: 虛擬影像定位整體流程圖

3.4 虛擬影像特徵點三角定位

在定位的流程中，讀取待定位的圖片，根據定位照片中特徵點數量的多寡，投票表決找出最合適的虛擬影像，參照虛擬影像所在的相機位置來定位，節省利用三角定位 (Triangulation) 所需的時間。比較與之前不同的影像資料庫建置方法，改善許多平面影像所無法定位的地方，增加定位的覆蓋率。圖3.12為影像定位的整體流程，在定位的程序上，主要會分成 3 個階段：

1. 導入虛擬相機坐標位置作參考
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

3.4.1 導入虛擬相機坐標位置作參考

在定位之前，會先輸入虛擬照相機的 Global Pose。虛擬照相機的 Global Pose 記錄檔格式包含每個虛擬相機的 X, Y, Z 座標以及相機的角度位置，當流程執行到特徵點定位時，就會需要參考到虛擬相機的位置。這些 Global Pose 的紀錄為先前在制定相機位置分布與角度時所記錄下來的資訊，等同於平面影像定位中真實相機的位置及角度資訊。藉由這些資訊得知虛擬影像所在位置，提供虛擬相片中特徵點比對前參考的起始位置。

3.4.2 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，接下來利用資料庫中所有的虛擬相片進行特徵點配對，在配對的過程中，將每張照片所擁有配對關係的特徵點數量記錄下來。如先前所述，我們利用 SIFT 比對特徵找出與待定位相片中對應的特徵點，提供之後定位所需要的特徵點位置資訊。當資料庫片的照片中特徵點的數量越多，代表與待定位的照片有越密切的關係，表示相片出現的景物與待定位相片有很大的重複性，相當於相片在相機位置附近或是相片拍攝角度與當時待定位相片所拍攝的角度一致，最好的情形是都符合上述兩種情況，所以在資料庫中，我們選出最有幫助的相片，相當於最多特徵點數量的虛擬相片。選出虛擬相片後，參考當時



(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 3.13: 特徵點比較差異圖，從圖中差異得知一般相機在與景物距離較遠的情形下，會有較少的特徵比對結果，虛擬相片因為模擬當時待定位所拍攝相片的位置取景，有較高的特徵相似程度

拍攝虛擬照片的相機編號，根據編號比對找出相機所在的位置，利用虛擬相機的 Global Pose 當作初步定位位置的參考起始點。

如圖3.13所示，特徵點的分布跟環境景物的分布有著密切的關係，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。

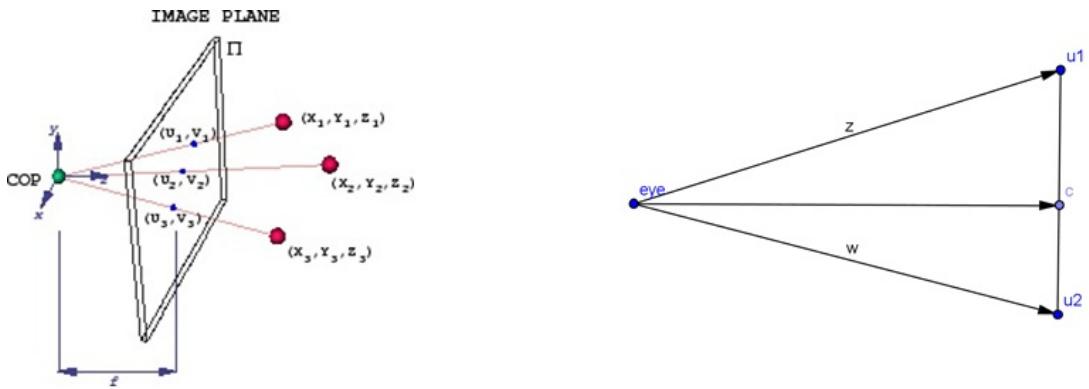


圖 3.14: 相機與特徵點的夾角示意圖

3.4.3 利用虛擬相機位置來定位

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

找出特徵點對於鏡頭圖片的夾角

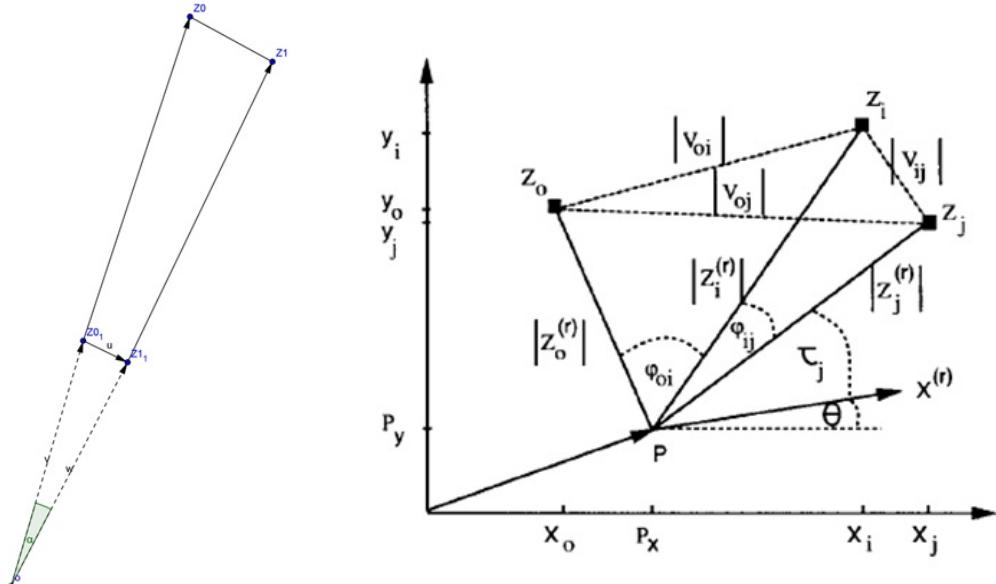
在 3.14 當中我們要先求得 \vec{z} 與 \vec{w} 的長度，當我們知道 \vec{U}_1 與 \vec{U}_2 之後，帶入下列求解的算式：

$$\begin{cases} \vec{z} = (u_{1x} - c_x)\vec{u} + (v_{1y} - c_y)\vec{v} + \vec{f}d \\ \vec{w} = (u_{2x} - c_x)\vec{u} + (v_{2y} - c_y)\vec{v} + \vec{f}d \end{cases} \quad (3.18)$$

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3.14我們知道 $|u|$ 的長度之後，再帶入餘弦定理求得角度 α 的夾角：

$$|\vec{u}|^2 = |\vec{z}|^2 + |\vec{w}|^2 - 2zw\cos\alpha \quad (3.19)$$



(a) 角度 α 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 3.15: 定位點夾角與長度向量關係示意圖

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3.15 我們可以知道 φ_{oi} 與 φ_{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：

$$\begin{cases} |V_{oi}|^2 = |Z_o^{(r)}|^2 + |Z_i^{(r)}|^2 - 2|Z_o^{(r)}||Z_i^{(r)}|\varphi_{io} \\ |V_{oj}|^2 = |Z_o^{(r)}|^2 + |Z_j^{(r)}|^2 - 2|Z_o^{(r)}||Z_j^{(r)}|\varphi_{jo} \\ |V_{ij}|^2 = |Z_i^{(r)}|^2 + |Z_j^{(r)}|^2 - 2|Z_i^{(r)}||Z_j^{(r)}|\varphi_{ij} \end{cases} \quad (3.20)$$

其中 (r) 代表從定位點 P 所觀測出的位置與視角。

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

$$\begin{cases} |Z_o^{(r)}| = (x_o - p_x)^2 + (y_o - p_y)^2 \\ |Z_i^{(r)}| = (x_i - p_x)^2 + (y_i - p_y)^2 \\ |Z_j^{(r)}| = (x_j - p_x)^2 + (y_j - p_y)^2 \end{cases} \quad (3.21)$$

我們利用上述式子整理可得出下列式子：

$$\begin{cases} |Z_o^{(r)}|^2 - |Z_i^{(r)}|^2 = X_o^2 - X_i^2 + 2p_x(x_i - x_o) + y_o^2 - y_i^2 + 2p_y(y_i - y_o) \\ |Z_o^{(r)}|^2 - |Z_j^{(r)}|^2 = X_o^2 - X_j^2 + 2p_x(x_j - x_o) + y_o^2 - y_j^2 + 2p_y(y_j - y_o) \end{cases} \quad (3.22)$$

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，當定位位置與相機距離超過平均誤差距離，我們會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 4 章 定位實驗成果比較分析

為了改善室內定位的覆蓋率以及定位的精確度，本研究利用建置虛擬相片的方法來增加照片的取樣範圍及過濾無法擷取照片特徵的不適角度，與平面 SIFT 影像定位方法做比較，利用數據及圖表做分析探討。我們分成兩個實驗環境來驗證定位方法所改善的數據：(1) 景物單一特徵固定的控制實驗環境、(2) 一般室內定位的環境。

首先製作一個景物單且特徵固定的環境情境，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，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。在特徵固定的環境下根據實驗情境與實驗方法說明改善的成果，再把方法放建築在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。

4.1 控制環境下定位結果分析

建立可以控制的實驗環境主要目的為在景物單一且可以控制的特徵點數環境下與一般平面影像定位做比較，藉由實驗驗證出更好的定位覆蓋率與更小的平均定位誤差。在根據有限特徵數量的環境下，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。

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作討論：

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

表 4.1: 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:	0.5m	0.5m
間距長度:	0.75m	0.5m

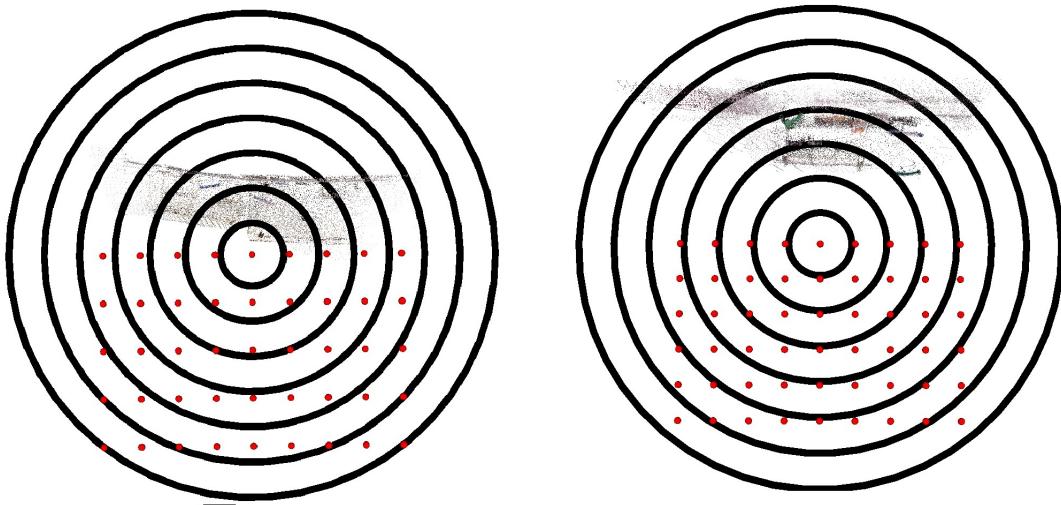
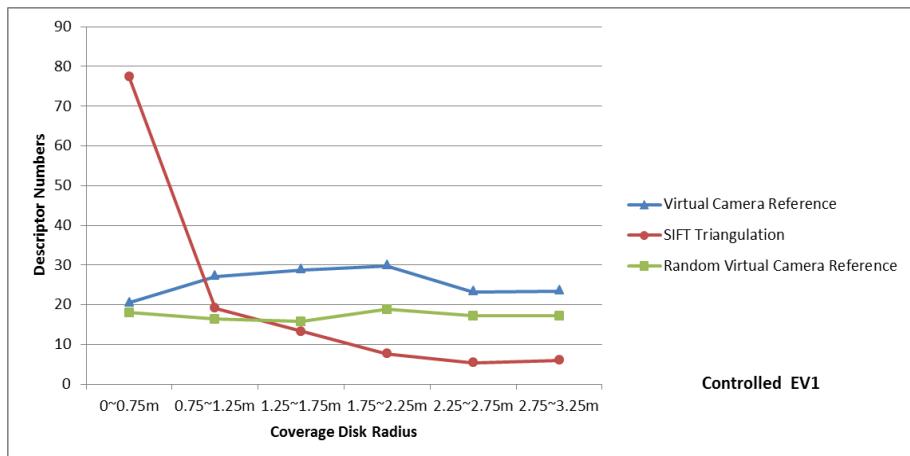


圖 4.1: 依照同心圓的覆蓋算出每個半徑內的每張圖的特徵點平均數，左. 控制環境 1 右. 控制環境 2

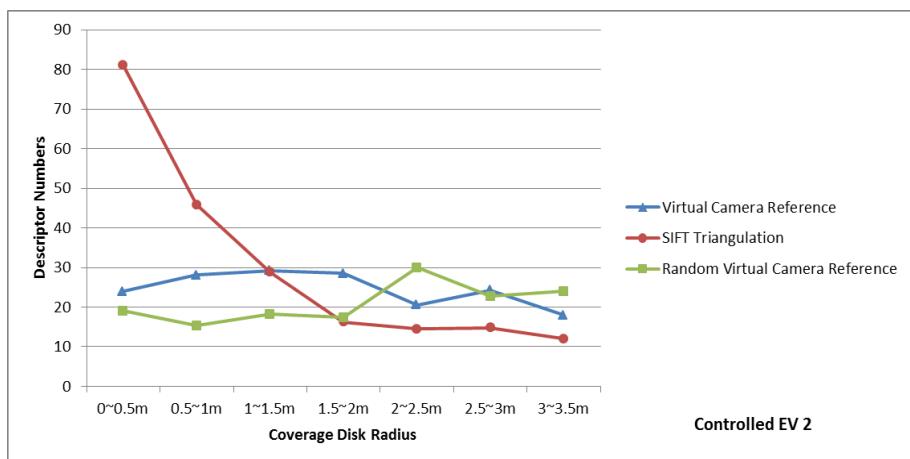
一開始建置實驗環境，我們將這些定位照片分布在 4 公尺 X 5 公尺的環境大小內，而景物的分布在 1.5 公尺 X 0.7 公尺的大小範圍內，Kinect 攝影機放置在距離景物 0.5 公尺前的地方，利用 Kinect 攝影機拍照取得深度照片與待定位的照片，待定位照片與虛擬照片解析度皆為 640 x 480。在表中 4.1 所示，在兩個控制環境的實驗中，利用不同的定位環境、不同的間距以及不同待定位圖片的數量來做實驗比較。

4.1.1 特徵點數量趨勢

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 4.1 所示。根據這個圓心將圓的半徑以間距距離的長度增加，像是控制環境 1. 中根據每個兼具距離增加 0.75m，控制環境 2. 中每個間距距離增加 0.5m，依據每個圓之間所能覆蓋的特徵



(a) 控制環境 1. 特徵點平均數量比較



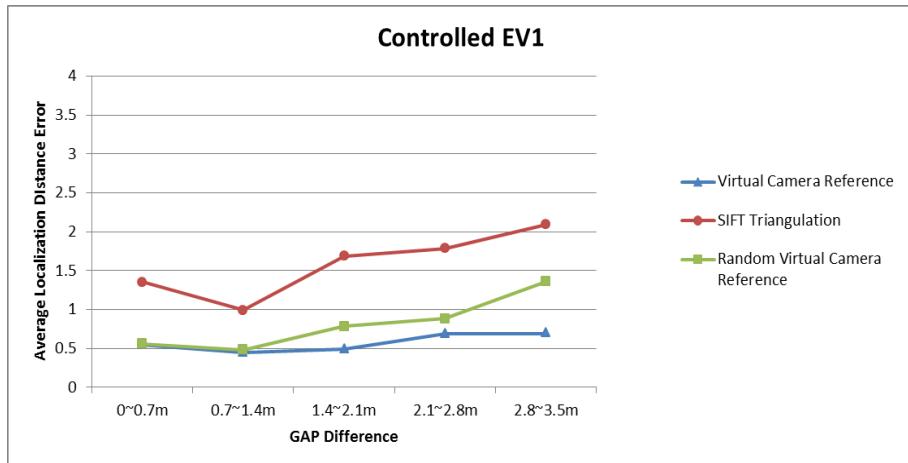
(b) 控制環境 2. 特徵點平均數量比較

圖 4.2: 特徵點數量趨勢圖，根據距離增加看出特徵點分布的數量差異情形

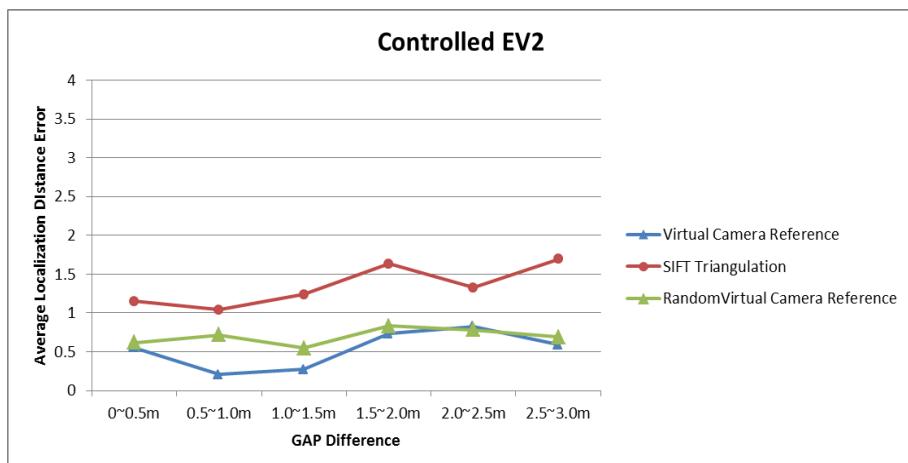
點位置做特徵點數量的總和並算出平均值，藉由圖中的趨勢看出特徵點數量的變化。根據圖4.2所示，特徵點的變化以虛擬影像格狀分布的數量區為最穩定的狀態，以圖4.2(a)在間距 0.75 m 的分布情形下，都有其對應的特徵點可供參考，反觀 SIFT 影像特徵雖然在一開始可以找到多數的特徵點，但是隨著距離增加，數量逐步遞減，最後只有些許的特徵可以找出。在圖4.2(a)因為間距距離較大，可以看出隨機分布與格狀分布的特徵點數量差異較為明顯，其中這些數量上的差異會影響之後間距定位的成果。

4.1.2 間距定位誤差趨勢

以圖4.3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，尤其以圖4.3(a)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。

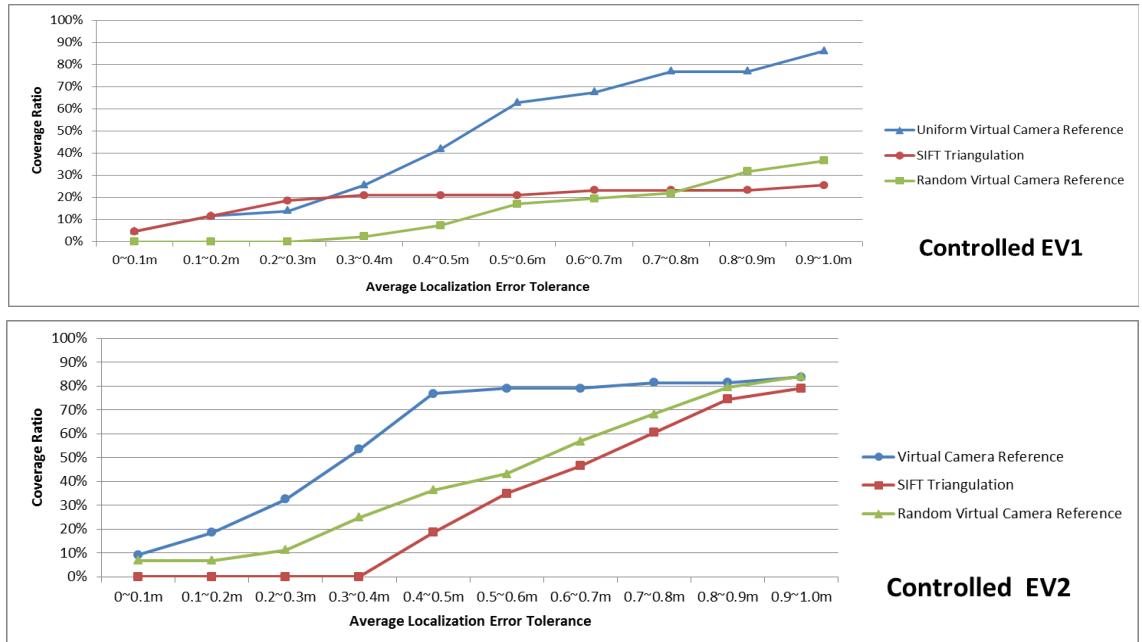


(a) 控制環境 1. 間距定位平均誤差比較

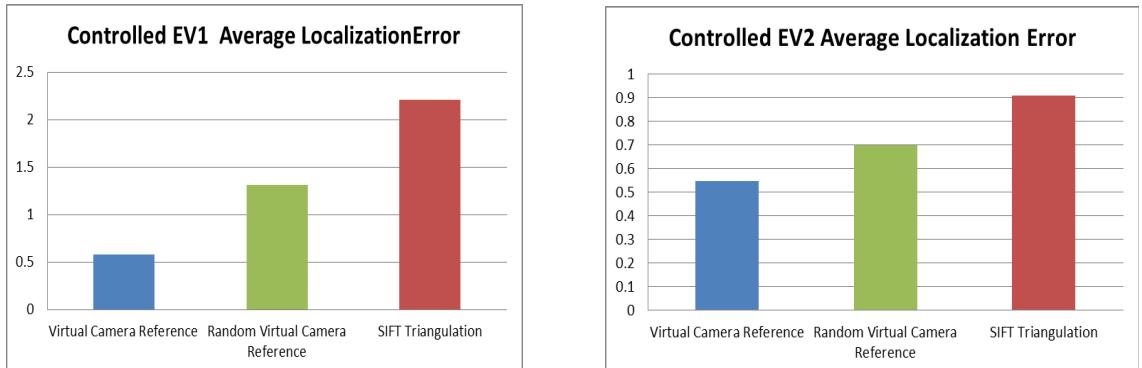


(b) 控制環境 2. 間距定位平均誤差比較

圖 4.3: 間距定位平均誤差趨勢圖，以每一橫排的待定位圖片平均誤差做計算，看出距離增加對定位誤差是否有影響



(a) 定位精準度覆蓋率



(b) 平均定位誤差

圖 4.4: 控制環境下定位精準度覆蓋率以及平均定位誤差成果分析，上圖為誤差容忍範圍下，可定位成功的覆蓋率分布情形，下圖整體的定位平均誤差

4.1.3 定位精準度覆蓋率以及平均定位誤差分布情況

研究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4.4(a)所示，在格狀分布的 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% 30% 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格狀分布平均的原因，再回到圖4.2(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4.3(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。這說明虛擬相機分布的位置，影響了虛擬相片的品質，平均定位誤差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。

4.2 一般室內環境定位實驗

在控制環境下，改善了定位的覆蓋率與精準度，在這章節將在一般室內環境下進行定位測試。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。

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4.5(c)與圖4.5(d)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 30 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。

表 2.2 記錄了不同室內定位環境的設定，分別在不同環境下進行實驗，每個環境的虛擬相機位置均採用格狀分布。圖4.6 記錄當時點雲環境的建置以及待定位

表 4.2: 室內定位環境大小分布

實驗設定:	客廳	廚房
待定位照片數量:	35 張	30 張
間距距離:	0.5m	0.5m
環境長度:	3.8m	4.1m
環境寬度:	4.1m	3.2m

照片的位置分布。點雲的建置是利用 Kinect 環繞室內環境四周所拍攝，再利用這些拍攝的圖片，當作平面 2D 影像定位所需的影像資料庫。接下來分別以不同虛擬相機照片的數量與平面 2D 影像做實驗比較分析。

4.2.1 定位覆蓋率與定位平均誤差分析比較

在室內定位的環境下，特徵點分布的數量，以及環境內觀測物的不同對定位結果增加許多變動因素。為了使定位結果能夠量化比較，我們將實驗結果分成(1). 定位精準度覆蓋率，與(2). 平均定位誤差兩個指標來分析成果。在圖4.6之中來看，橫坐標表示定位誤差的容忍範圍，縱座標代表在這個誤差範圍下的定位成功率，以三種不同虛擬照片的數量與傳統 2D 影像照片做實驗比較。

以覆蓋率來看，100 張虛擬照片的定位結果為最好，代表取得越多點雲環境的資料，定位的改善越明顯。因為在每個相機位置上，我們取出兩張虛擬照片，所以在環境中總共有 50 個虛擬相機位置作比較，分別與 40 個與 35 個相機位置相比，有更多定位參考的依據。則平面 2D 影像因為只有一部分的環境參考依據，所以在定位覆蓋率與平均誤差，都比 100 張與 80 張虛擬照片定位成果來的差。以 4.6(a)來看，誤差範圍在 $0.5m \sim 0.6m$ 之間的覆蓋率，以 100 張虛擬照片覆蓋率最高，有 80% 左右的定位成功率，但是平面 2D 影像只有 60% 左右的成功率，增加了 20% 的覆蓋率。以平均誤差來說也有改善，在上個章節中我們發現平面 2D 影像定位誤差會有不穩定的情況發生，這種情況增加了整體的平均定位誤差，而在虛擬影像定位因為誤差會穩定在 $1m$ 以內的範圍內，所以也會降低平均定位誤差。

從兩個定位環境下以圖 4.5(c)與圖 4.5(d)分別看兩個待定位圖片分布的情形，

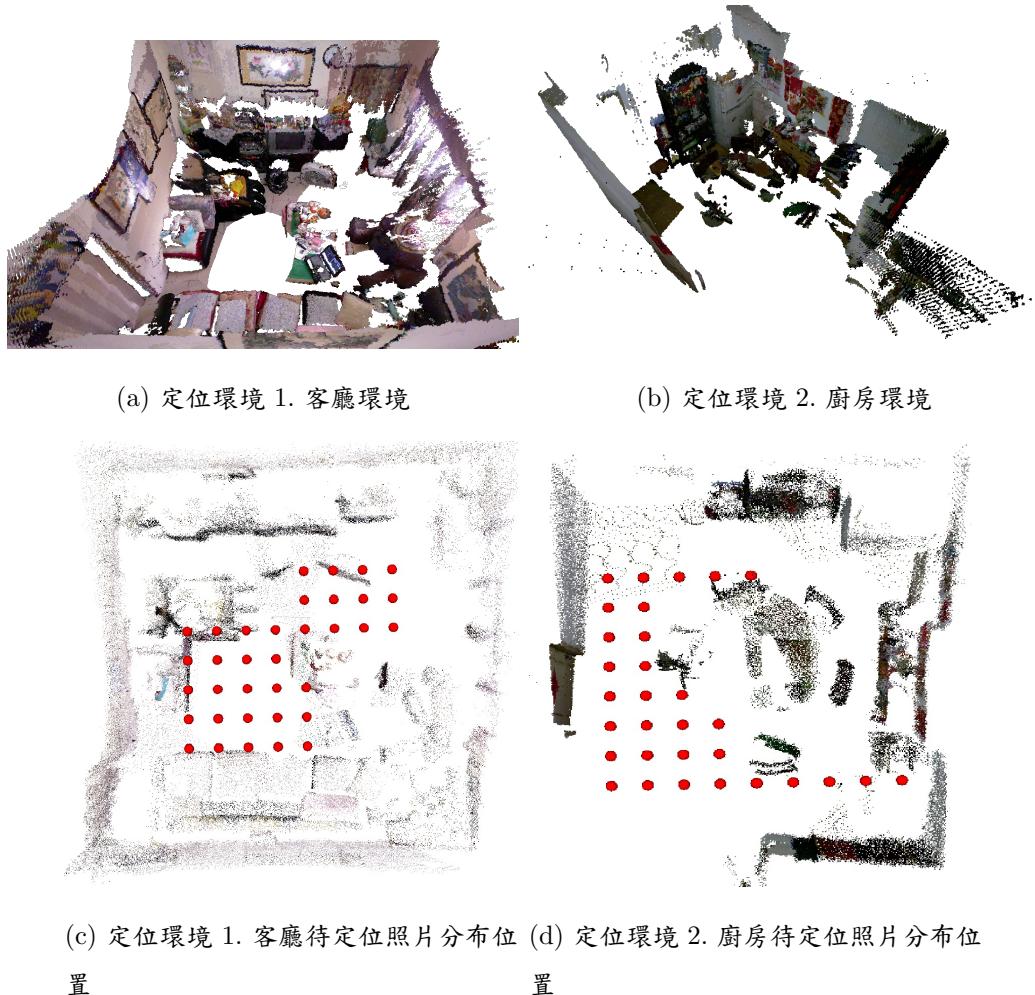


圖 4.5: 室內定位環境與待定位照片分布位置

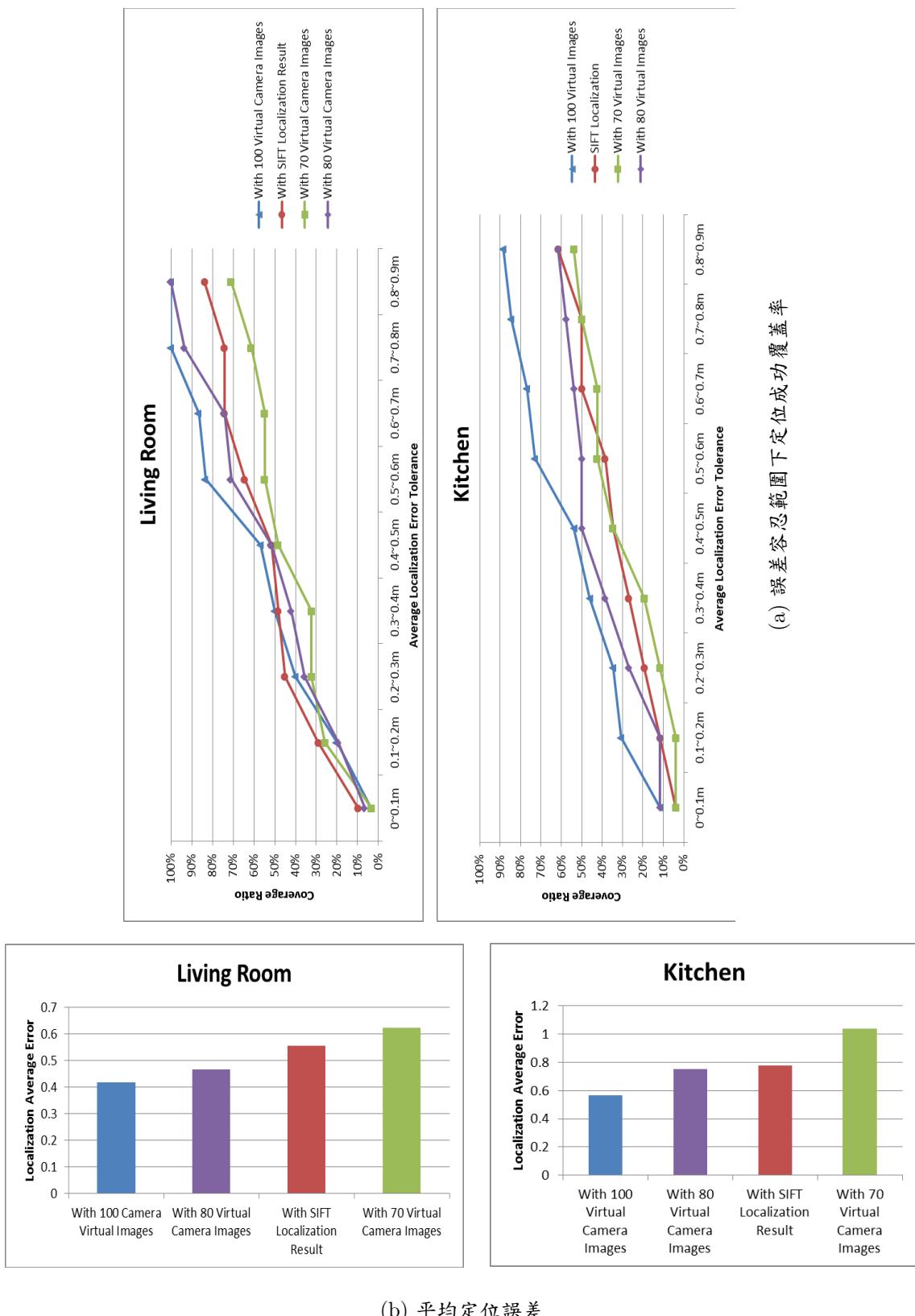


圖 4.6: 室內環境定位成果，上圖為誤差容忍範圍下定位成功覆蓋率，在容忍範圍下覆蓋率分布的變化，下圖為平均定位誤差，在整體環境下比較各方法定位誤差

得知在廚房的環境的待定位圖片分布偏向於同一側，另一側因為障礙物阻擋，對於實際相機取得較為困難。因為平面影像定位的環境資料缺乏，導致定位誤差上升，尤其以兩個定位環境互相比對最為明顯。但整體來說，100 張虛擬影像所取得定位成果都最為穩定，說明虛擬影像可以在一定的數量下可以補足環境資訊的缺乏，藉由優化相機角度增加影像的相關性，在相機無法拍攝的位置取得影像及避開障礙物，使得整體覆蓋率與誤差都能改善。

第 5 章 結論與未來展望

在室內定位環境下，我們的方法改善了平均定位誤差以及提升了整體定位成功的覆蓋率，證明了利用虛擬照片能夠取得在一般照片不容易取得的環境資訊，計算照片中的的平均深度過濾出距離物體過近的虛擬相片，以及透過均勻分布的虛擬相機位置，取得一般相機所遺漏的位置與角度。根據固定景物的定位環境與一般室內環境做實驗，藉由實驗證實我們研究中方法的可行性，希望藉由我們研究提供機器視覺與定位研究一個新的發展方向。在研究方法我們提供了新的定位程序希望藉此改善傳統定位諮詢發生的不足與缺陷，以下提供了我們以改進的目標：

- 本篇研究提出了重建環境 3D 點雲的方法來增加特徵點採集的成效，與一般平面相片做特徵點的數量比對，在不同的區域上發現虛擬相片不會因為與景物距離過遠導致特徵點數量減少的問題發生。
- 經過相片深度篩選後調整相機角度改善了與待定位照片的相關性，增加與待定位照片更多的相似性提升了整體室內定位的精確度。
- 虛擬相片的定位方法改善了定位的平均誤差也因此提升了整體定位成功的覆蓋率，增加室內環境內可定位的區域。

建置點雲環境比一般利用雷射透過渲染的方式建置 3D 環境速度快上許多，但是比較貼圖細節與紋理還是沒辦法與渲染的成效相比，希望在之後研究能夠改善點雲環境的精細程度，使得虛擬照片不只在位置與角度做改善，在相片內容也能提供更多的特徵加以比對，提升整體定位的精準度。之後也可以把這個定位系統移植到手機上，藉由 client 端收集待定位照片，交給 sever 端記算訂位資訊，完成雲端平台像是 Google 定位的服務，增進影像定位的便利性與實用性。下面幾項為我們方法可望改善的幾項建議：

- 希望透過更好的渲染效果增加虛擬相片的精細度，或者減少點雲建置所需要的時間。
- 在之後能夠利用雲端或是分散式架構減少定握所需要的時間成本，增加定位的效率。

- 能夠增加人性化的容錯機制，或是美化使用者介面的 UI，使得這套虛擬影像定位能夠更為便利化使用。

Bibliography

- [1] I. Bekkerman and J. Tabrikian, “Target detection and localization using mimo radars and sonars,” *Signal Processing, IEEE Transactions on*, vol. 54, no. 10, pp. 3873 –3883, oct. 2006.
- [2] C.-H. Lim, Y. Wan, B.-P. Ng, and C.-M. See, “A real-time indoor wifi localization system utilizing smart antennas,” *Consumer Electronics, IEEE Transactions on*, vol. 53, no. 2, pp. 618 –622, may 2007.
- [3] J. Zhou and J. Shi, “Rfid localization algorithms and applications—a review,” *Journal of Intelligent Manufacturing*, vol. 20, pp. 695–707, 2009.
- [4] B. Zitová and J. Flusser, “Image registration methods: a survey,” *Image and Vision Computing*, vol. 21, pp. 977–1000, 2003.
- [5] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
- [6] “Robust monte carlo localization for mobile robots,” *Artificial Intelligence*, vol. 128, pp. 99 – 141, 2001.
- [7] D. Cobzas, H. Zhang, and M. Jagersand, “Image-based localization with depth-enhanced image map,” *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 2, pp. 1570 – 1575 vol.2, sept. 2003.
- [8] R. Rusu and S. Cousins, “3d is here: Point cloud library (pcl),” *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 1 –4, may 2011.
- [9] K. Muthukrishnan, M. Lijding, and P. Havinga, “Towards smart surroundings: Enabling techniques and technologies for localization,” 2005.
- [10] H. Andreasson, A. Treptow, and T. Duckett, “Localization for mobile robots using panoramic vision, local features and particle filter,” *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 3348 – 3353, april 2005.

- [11] T. Liu, M. Carlberg, G. Chen, J. Chen, J. Kua, and A. Zakhori, “Indoor localization and visualization using a human-operated backpack system,” *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, pp. 1 –10, sept. 2010.
- [12] F. Bonin-Font, A. Ortiz, and G. Oliver, “Visual navigation for mobile robots: A survey,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 53, no. 3, pp. 263–296, 2008.
- [13] H. Morita, M. Hild, J. Miura, and Y. Shirai, “View-based localization in outdoor environments based on support vector learning,” *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pp. 3083–3088, 2005.
- [14] J. Wolf, W. Burgard, and H. Burkhardt, “Robust vision-based localization by combining an image-retrieval system with monte carlo localization,” *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 208–216, 2005.
- [15] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, “Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments,” *International Journal of Robotics Research*, vol. 31, no. 5, pp. 647–663, 2012.
- [16] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. Rendón-Mancha, “Visual simultaneous localization and mapping: a survey,” *Artificial Intelligence Review*, pp. 1–27, 2012.
- [17] K. Ho and P. Newman, “Detecting loop closure with scene sequences,” *International Journal of Computer Vision*, vol. 74, no. 3, pp. 261–286, 2007.
- [18] E. Eade and T. Drummond, “Monocular slam as a graph of coalesced observations,” *Proceedings of the IEEE International Conference on Computer Vision*, 2007.
- [19] D. Chekhlov, W. Mayol-cuevas, and A. Calway, “Appearance based indexing for relocalisation in real-time visual slam,” *In 19th Bristish Machine Vision Conference*, pp. 363–372, 2008.

- [20] A. Gil, scar Reinoso, M. Ballesta, and M. Julie, “Multi-robot visual slam using a rao-blackwellized particle filter,” *Robotics and Autonomous Systems*, vol. 58, no. 1, pp. 68 – 80, 2010.
- [21] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, “Bundle adjustment —a modern synthesis,” *Vision Algorithms: Theory and Practice*, vol. 1883, pp. 298–372, 2000.
- [22] K. Konolige, G. Grisetti, R. Kümmerle, W. Burgard, B. Limketkai, and R. Vincent, “Efficient sparse pose adjustment for 2d mapping,” *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pp. 22 –29, oct. 2010.
- [23] S. Zhang, “Recent progresses on real-time 3d shape measurement using digital fringe projection techniques,” *Optics and Lasers in Engineering*, vol. 48, no. 2, pp. 149 – 158, 2010.
- [24] C. Riechert, F. Zilly, M. Muller, and P. Kauff, “Advanced interpolation filters for depth image based rendering,” *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, 2012, pp. 1 –4, oct. 2012.
- [25] B. M. Oh, M. Chen, J. Dorsey, and F. Durand, “Image-based modeling and photo editing,” pp. 433–442, 2001.
- [26] D. Wang, J. Liu, J. Sun, W. Liu, and Y. Li, “A novel key-frame extraction method for semi-automatic 2d-to-3d video conversion,” *Broadband Multimedia Systems and Broadcasting (BMSB), 2012 IEEE International Symposium on*, pp. 1 –5, june 2012.
- [27] 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*, pp. 75–84, 2011.

- [28] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Commun. ACM*, vol. 24, pp. 381–395, jun 1981.