

國立臺灣大學工學院機械工程學系
碩士論文

Department of Mechanical Engineering
College of Engineering
National Taiwan University
Master Thesis



基於通訊信號強度預測距離之
未知初始相對姿勢自主多機器人協同探索
Autonomous Multi-robot Exploration with unknown
initial relative pose based on distance prediction using
communication signal strength

賴重叡
Chung-Jui Lai

指導教授：詹魁元博士
Advisor: Kuei-Yuan Chan, Ph.D.

中華民國 112 年 08 月
August, 2023



誌謝

就讀碩士班的這兩年，首先感謝指導教授詹魁元老師讓我加入 SOLab 大家庭，並且在學術、職涯都給了許多寶貴意見。研究上老師讓我們自由發揮，遇到困難時又給予許多思考方向，著實讓我們成長。嘿。對???

賴重叡 謹識於
國立臺灣大學 機械工程學系
中華民國一百一十二年八月



摘要

當機器人進行自主環境探索時，它會依靠感測器所收集到的環境資訊建立環境地圖，同時根據探索策略進行目標點分配。為了增加探索效率和範圍，通常會使用多台機器人進行協同探索，同步地建立環境地圖，以實現全面、高效的探索任務。現有的多機器人協同探索，通常機器人間初始相對關係是已知的，如果機器人間初始相對關係是未知的，就需要進行額外的地圖合併（Map Merge）處理，建立地圖之間的關聯性，使得各個局部地圖的座標系和比例尺一致，並整合成全域地圖。已有研究將地圖視作影像的一種，並使用影像縫合技術中的相同影像特徵來計算地圖的相對關係。然而，機器人地圖常採用格點佔據地圖格式，並只包含未知、自由和佔據區域3種數值，因此在工廠和圖書館等高重複性特徵的環境中，提取的特徵更難以進行匹配。因此，當機器人間初始相對關係是未知的情況，僅使用影像特徵來進行處理的方法可能無法實現高重複性特徵地圖的合併。本研究認為，欲合併高重複性特徵地圖，必須要有額外的資訊。因此，本研究將現有的多機器人協作探索修改為初始相對關係未知，並加入通訊信號強度預測出之機器人間相對距離作為額外的資訊，來進行地圖合併。透過本研究增加的機器人間相對距離資訊，可以建立個個局部地圖座標系的相對變換關係，再透過原有的特徵合併技術進行修正，有效地將地圖進行合併。此外，本研究之探索策略修改為，地圖未合併之機器人採用單機器人探索，地圖合併之機器人則採用多機器人協同探索，因為機器人間初始相對關係未知，無法直接進行多機器人協同探索。本研究以相同案例分別測試於原特徵擷取方法與加入額外資訊的方法，結果顯示經過本研究的方法不僅可成功合併，而且在判斷合併的指標分數上也有顯著的提昇。同時模擬結果表明，提出的策略可以在合理的時間內探索整個地圖，並與單機器人探索相比降低了探索成本。

關鍵字：地圖建構、地圖合併、多機器人、格點佔據地圖，多機器人探索、高重複性特徵環境



Abstract

When robots autonomously explore their environment, they rely on sensor data to build a map and use exploration strategies to allocate target points. To increase exploration efficiency and range, multiple robots are often used for collaborative exploration, synchronously building a map to achieve a comprehensive and efficient exploration task. Existing multi-robot collaborative exploration usually assumes that the initial relative relationship between robots is known. If the initial relative relationship between robots is unknown, additional map merging processing is required to establish the correlation between maps, making the coordinate systems and scales of each local map consistent and integrated into a global map. Some studies treat maps as images and use the same image features in image stitching techniques to calculate the relative relationship between maps. However, robot maps often use grid occupancy map formats and only include three values: unknown, free, and occupied areas. Therefore, it is more difficult to match extracted features in environments with high repetitive features, such as factories and libraries. To merge maps with high repetitive features, additional information is necessary. This study modifies existing multi-robot cooperative exploration to account for an unknown initial relative relationship and incorporates the predicted communication signal strength between robots as additional information for map merging. This additional relative distance information between

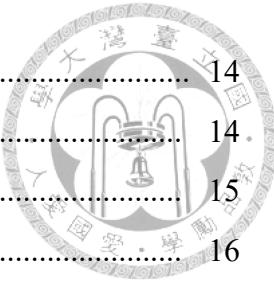
robots can establish a relative transformation relationship for each local map's coordinate system, effectively merging the maps using existing feature merging techniques. In addition, the exploration strategy is modified to use single-robot exploration for robots whose maps have not been merged and multi-robot collaborative exploration for robots whose maps have been merged. This is because the initial relative relationship between robots is unknown, and direct multi-robot collaborative exploration is not possible. The results show that the proposed method can successfully merge maps and significantly improve the score of the merging indicator compared to the original feature extraction method. The simulation results also demonstrate that the proposed strategy can explore the entire map in a reasonable time and reduce exploration costs compared to single-robot exploration.

Keywords: Mapping, Map Merge, Multi Agent, Occupancy Grid Map, Multi-robot exploration, high-repetitive feature environment

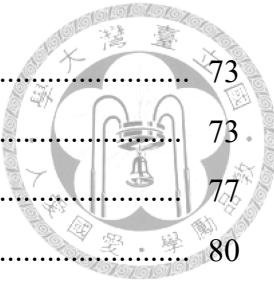


目錄

誌謝	i
摘要	ii
Abstract	iii
目錄	v
圖目錄	viii
表目錄	xi
第一章 緒論	1
1.1 前言	1
1.2 自主地圖探索	1
1.3 多機器人系統	3
1.4 研究動機與目的	3
1.5 本文架構	4
第二章 文獻回顧	5
2.1 機器人地圖資料型態	5
2.2 自主地圖探索	7
2.2.1 單機器人自主地圖探索	7
2.2.2 多機器人自主協作地圖探索	8
2.3 格點佔據地圖合併方法	9
2.3.1 直接類	9
2.3.2 間接類	10
2.4 文獻回顧總結	13



第三章 研究流程與架構	14
3.1 問題描述	14.
3.2 研究方法	15
3.2.1 流程與架構	16
3.2.2 測試案例	20
第四章 現有流程之技術	22
4.1 特徵辨識	23
4.1.1 尺度空間之極值檢測	23
4.1.2 特徵點定位	25
4.1.3 特徵點方向計算	26
4.1.4 特徵點描述	26
4.2 特徵點匹配	27
4.3 計算變換關係	27
4.3.1 隨機採樣共識演算法	28
4.3.2 以 RANSAC 演算法計算相對變換關係	28
4.4 評判相對變換關係	30
4.5 地圖銜接處理	31
4.6 各演算法效果討論與合併測試	34
4.6.1 演算法效用討論	34
4.6.2 合併測試	39
第五章 加入地圖修正之合併方法	44
5.1 影像二值化處理	44
5.2 雷登轉換與影像旋轉	46
5.3 閉運算處理	49
5.4 興趣點採樣	55
5.5 主體結構修正	59
5.6 影像修正後合併結果討論	65
5.6.1 影像修正對特徵之差異	65
5.6.2 合併測試	69

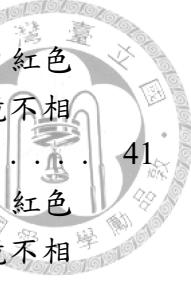


第六章 模擬環境測試	73
6.1 模擬環境建構	73.
6.2 合併測試與討論	77
第七章 總結與未來展望	80
7.1 結論	80
7.2 研究建議與未來展望	81
附錄A 格點佔據地圖之格點資訊處理	82
附錄B 程式碼	84
參考文獻	85



圖目錄

1.1	自主探索架構關係圖 [1]。	2
1.2	個體局部地圖之相對姿態關係示意圖， x, y, θ 表示個體的參考座標系， Z_{ij} 表示兩個體的相對姿態關係。	4
2.1	格點佔據地圖示例 [2]。	5
2.2	特徵地圖示例 [2]。	6
2.3	拓樸地圖示例 [2]。	6
3.1	光達誤差對於地圖建構之影響示意圖。	17
3.2	本文研究之參考方法	18
3.3	本文研究方法，主要在參考方法之前加入影像處理，進行格點佔據地圖的修正。	19
3.4	本文研究測試用地圖。座標軸刻度用於表示影像格點數量，解析度高的地圖格點數較多，解析度低的地圖格點數則較少。	21
4.1	影像金字塔與高斯差分函數計算示意圖 [3]	24
4.2	特徵點極值比對示意圖 [3]	24
4.3	影像梯度與角度統計 [3]。左圖為影像梯度計算，圓圈表示高斯函數權重，距離圓心越遠權重越小；右圖為統計各子區域的角度區間。此圖僅以 2×2 個子區域示意，實際計算時子區域數量為 4×4 個。	27
4.4	機率與熵值對應關係。	33
4.5	SIFT 演算法應用於局部測試地圖之特徵擷取效用呈現。	35
4.6	SIFT 特徵匹配示意圖。	38
4.7	熵濾波處理應用於局部測試地圖一與二合併之示意圖。	39
4.8	組合一之合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。	40



4.9 組合二之合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境不相符，代表計算的相對變換關係錯誤。	41
4.10 組合三之合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境不相符，代表計算的相對變換關係錯誤。	42
5.1 格點佔據地圖二值化處理。圖 5.1(a) 為原格點佔據地圖之影像，具有 3 種像素值（黑色：0、灰色：128、白色：255）來表示不同區域種類。圖 5.1(b) 則表示該地圖經過二值化處理後，佔據區域（白色：255）與非佔據區域（黑色：0）的分佈。	45
5.2 格點影像線段的扭曲示意圖。圖 5.2(a) 表示二值化地圖影像的佔據區域線段；圖 5.2(b) 為圖 5.2(a) 紅圈處放大示意圖，紅框處表示階梯狀的扭曲。	46
5.3 影像雷登轉換結果。圖 5.3(a) 為原平面影像 f ，紅框處表示積分值最大線段。圖 5.3(b) 為雷登轉換後的線性函數 Rf ，縱座標表示直線參數 r ，橫座標表示直線參數 θ 。紅圈處表示此點具有積分最大值，可對應至輸入影像中最長線段，即左圖紅框線段。	47
5.4 像素值插值計算示意圖。	48
5.5 影像旋轉之處理示意圖。圖 5.5(a) 為欲旋轉影像；圖 5.5(b) 為紅圈處放大影像；圖 5.5(c) 為紅圈處旋轉後，像素插值運算（式（5.5））的結果，部份格點顏色顯示灰色，表示像素值介於 0~255 之間；圖 5.5(d) 為再次經過二值化操作的結果，圖中白色線段產生許多毛邊與缺口。	49
5.6 影像與遮罩定義及平移示意圖。	50
5.7 膨脹操作示意圖。	52
5.8 侵蝕操作示意圖。	53
5.9 閉運算操作示意圖，綠點表示經過閉運算後額外新增之白色格點，與原影像（圖 5.6(a)）相比，部份缺口已受到填補。	54
5.10 地圖完成旋轉處理後，部份線段產生毛邊與缺口，故經過形態學之閉運算處理進行修補。圖 5.10(a) 為原旋轉處理後的影像區塊；圖 5.10(b) 為經過閉運算操作的影像區塊，可看見毛邊與缺口受到修補的效果。	55
5.11 興趣點採樣流程圖。	58
5.12 影像興趣點採樣示意圖。圖 5.12(a) 為原先影像旋轉與閉運算處理後的結果；圖 5.12(b) 為圖 5.12(a) 經過採樣後，紀錄興趣點的影像；圖 5.12(c) 與圖 5.12(d) 則分別為前兩圖的放大圖。	59
5.13 興趣點位置修正流程圖。	63



5.14 興趣點位置修正之情況 (3)	64
5.15 影像主體結構修正比較圖。圖 5.15(a) 為原先影像旋轉與閑運算處理後的結果；圖 5.15(b) 則是利用興趣點影像（圖 5.12(b)）進行結構修正後的結果；圖 5.15(c) 與圖 5.15(d) 則分別為前兩張影像的放大圖。從比較圖結果可見，主體線段修正後較為筆直且寬度統一。	65
5.16 表 3.2 之測試案例經過影像修正處理之結果。所有案例經過修正後，佔據區域的線段寬度皆為 1 格。因此相對於影像尺寸，地圖一、三、四的線段寬度比例過小，導致色彩表示上較不明顯，同時也將影響後續特徵擷取。	66
5.17 針對修正後影像進行特徵擷取之結果，紅點表示被擷取出的特徵點。	67
5.18 迭代最近點演算法之效果示意圖 [4]	68
5.19 經過影像修正之組合一合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。	70
5.20 經過影像修正之組合二合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。	71
5.21 經過影像修正之組合三合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。	72
6.1 臺大永齡生醫館之二樓平面圖。	74
6.2 臺大永齡生醫館二樓之模擬環境。	74
6.3 預計建構地圖之區塊。將模擬環境分為三區，並以 SLAM 方法建構局部地圖。	75
6.4 永齡生醫館模擬環境之局部地圖一。	75
6.5 永齡生醫館模擬環境之局部地圖二。	76
6.6 永齡生醫館模擬環境之局部地圖三。	76
6.7 局部地圖一（圖 6.4）與地圖二（圖 6.5）之合併可視化影像，綠色區塊表示正確匹配的格點，紅色區塊表示錯誤匹配的格點。	78
6.8 局部地圖二（圖 6.5）與地圖三（圖 6.6）之合併可視化影像，綠色區塊表示正確匹配的格點，紅色區塊表示錯誤匹配的格點。	78
6.9 根據局部地圖兩兩相對關係（圖 6.7、圖 6.8），可將局部地圖全部合併為全域地圖。	79



表目錄

3.1 測試案例之地圖組合	20
3.2 測試案例之地圖組合	20
5.1 不同方法對於表 3.2 案例的測試結果總整理。	72
6.1 測試案例之地圖組合	77
A.1 格點佔據地圖之格點資訊關係表	83



緒論

1.1 前言

自主探索環境一直是機器人領域中持續發展的研究議題。從最早的單一機器人自主探索未知環境，到近年來，多機器人系統 (Multi-Robot System, MRS) 在探索未知環境中受到越來越多的關注。MRS 可以利用多個機器人進行協作，實現高效率和高精度的地圖同時構建。但由於機器人個體的參考座標系皆不同，即便可與其他機器人進行通訊，所擷取的資料也需要建立相對關係並加以整合。換言之，每個機器人在建構出局部地圖後，必須使用一套結合各個機器人局部地圖的演算法來完成地圖合併。在自主探索不斷發展的今天，了解目前多機器人在自主探索及地圖合併中所遇到的問題，是相當必要的。因此，接下來將分別介紹自主地圖探索和 MRS 技術，再探討同時應用兩種技術時可能面臨的挑戰和限制，並提出本研究的動機和目的。

1.2 自主地圖探索

自主地圖探索可以被定義為穿越未知環境以構建後續導航所需的地圖的行為 [5]，而此過程應當高效進行，以在最短時間內獲取最多的信息。在地圖探索的過程中，機器人應避免已經探索過的區域，而傾向於未知區域。如圖1.1所示，探索位於層次結構的頂層，中層則是路徑規劃，而底層則是機器人的定位。

層次結構中的頂層是探索，探索未知區域需要機器人在全局坐標系中的位置和方向，且需要一條探索的路徑。探索包含探索策略及探索地圖建構，探索策略負責尋找探索的目標點，而探索地圖建構則負責更新地圖。探索可以讓機器人從未知的環境中獲取新的信息，提高機器人的自主性和適應性，因此，探索在自主地圖探索中被視為最重要的一步，是層次結構的頂層。



層次結構中的第二層是路徑規劃，探索策略在環境中尋找到目標點，機器人必須使用探索構建的地圖和機器人的當前位置和目標位置來計算最佳路徑，並規劃路徑以避開障礙物，到達目標點，同時使用感測器數據更新地圖，以確保地圖的準確性。

層次結構的底層為定位，因為要進行路徑規劃和探索，機器人需要具備自我定位的能力（即知道當前的位置和方向，定義為機器人的姿態），且目標和當前的姿態必須使用固定的全局框架來表示，因為構建地圖中使用的距離傳感器讀數（例如來自 LiDAR 的讀數）必須統一，也就是從傳感器坐標系轉換為全局坐標系。如果不使用絕對定位系統，地圖也可用於相對定位，這種方法被稱為同時定位和映射（SLAM），它需要將感測器數據與地圖匹配，以計算機器人的位置和方向，這通常使用濾波器（如卡爾曼濾波器）等技術實現 [6]，以將感測器噪聲和不確定性降到最低。

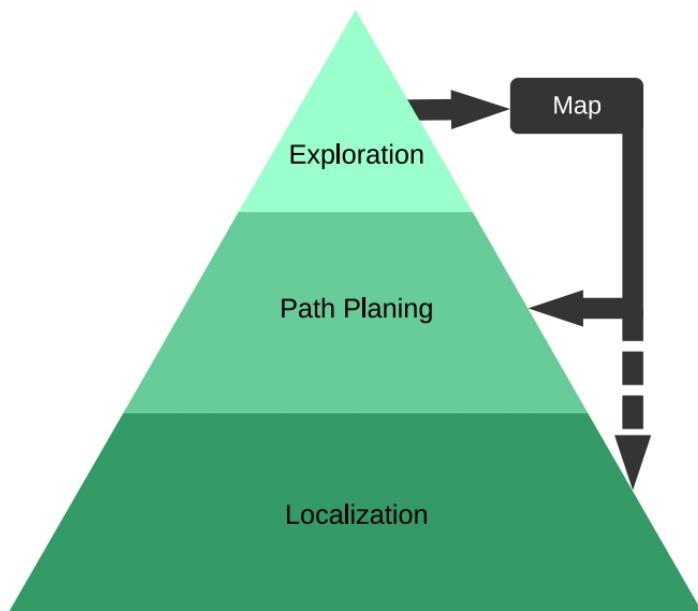


圖 1.1: 自主探索架構關係圖 [1]。

根據上述可以了解到，自主地圖探索是一個包含多個問題的複雜領域，包括機器人探索策略、自我定位、地圖構建、路徑規劃等。而下一節將介紹多機器人系統，以及其如何提昇探索效率。



1.3 多機器人系統

在探索環境中，提升效率是一個重要的議題。當個體的能力有限時，合作是提高效率的重要方式。這個觀念在人類生活中得到廣泛應用，同樣也可以在機器人領域中使用，即多機器人系統（MRS）。MRS 中，多個機器人協同工作共享信息，形成較多元的感測設備和多個策略方案，可以大幅提升探索環境的效率。在探索環境的案例中，多個機器人可以分散的方式，各自探索，大幅加速環境的覆蓋率。另外，MRS 也可提昇系統的可靠性，單一機器人的失效並不會直接造成任務的失敗，仍可藉由其他機器人的運行，完成任務，因此具有強健的特性。

MRS 中，機器人之間的通訊和協作是關鍵，但隨著個體數量的增加，所接收的感測資訊也變得龐大，資料間的整合、關聯處理也變得複雜與困難；而多機器人協作技術則包括集體決策、分工協作、任務分配等，必須根據任務定義必要的主從關係或者群體文化，因此增加了系統複雜度。

儘管多機器人系統已廣泛應用於工業、環境監測、救援等領域（例如，在救援領域中，多機器人系統可以協作完成搜救、救援等任務），但仍存在數個挑戰需要解決。因此，未來需要針對這些挑戰進行更深入的研究和探索，以實現 MRS 在各個領域的廣泛應用。下一節將對機器人自主探索與 MRS 的整合進行介紹，並說明本文研究目標及動機。

1.4 研究動機與目的

為了達成多台機器人同時自主探索環境的各項需求，必須將前兩節提及的機器人自主探索與 MRS 技術進行整合，可稱作「多機器人自主協同探索（autonomous multi-robot exploration）」。此整合技術中，機器人個體各自依據探索策略取得的目標點，以及利用通訊協定交換彼此感測資料並進行資料關聯的處理來更新地圖，以加快環境探索覆蓋率。但研究主要集中在通信和任務分配等方面，並假設機器人個體間已知彼此初始相對姿態，也就是一開始全部的局部地圖都可以轉為全域地圖合併。然而，在多機器人自主協同探索中，並無法預期個體已知彼此初始相對姿態（圖1.2中的 Z_{ij} ），也就是地圖合併是一個必要的步驟，地圖合併通常採用特徵合併技術來進行，可以使機器人更加全面地瞭解未知環境。

然而在高重複性特徵環境下，使用特徵合併技術來地圖合併變得更加困難，因為從地圖可能無法區分具有相同外型的特徵，導致在地圖合併過程中產生錯誤。因此，需要一種能夠應對高重複性特徵環境的地圖合併方法，以提高多機器人探索的效率和準確性。因此，本文希望探討機器人間未知初始相對姿勢，且在高重複性特徵環境的情況下，如何僅藉由個體所具有的資訊使局部地圖進行合併，並進行多機器人自主協同探索。

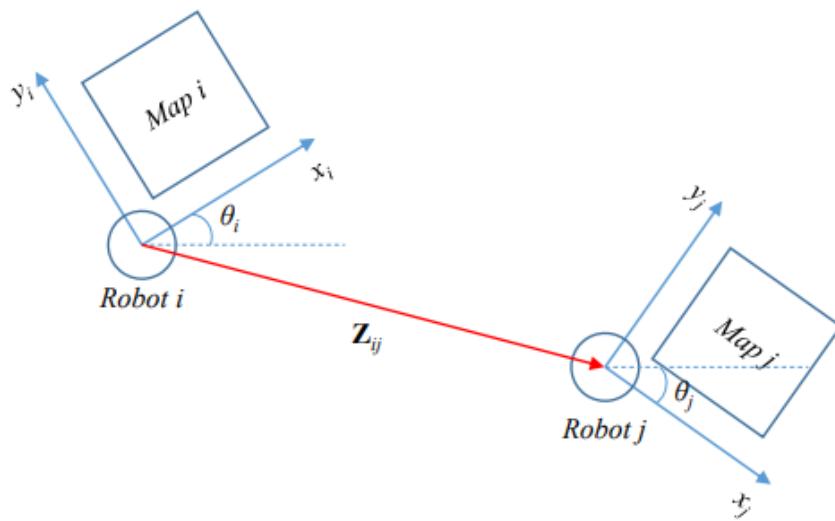


圖 1.2: 個體局部地圖之相對姿態關係示意圖， x, y, θ 表示個體的參考座標系， Z_{ij} 表示兩個體的相對姿態關係。

1.5 本文架構

本文一共分為七章，內容架構如下：

1. 第一章：緒論

介紹本研究主題的相關背景，說明研究動機與目的，並呈現全文架構。

2. 第二章：文獻回顧

探討本研究領域的文獻與相關方法。

3. 第三章：研究流程與架構

解釋本研究所使用方法的基本邏輯及相關流程。

4. 第四章：多機器人自主探索

說明現有用於多機器人自主探索的各項演算法。

5. 第五章：加入信號強度預測距離之地圖合併

說明本研究提出之地圖合併流程，並解釋演算法邏輯及其效用。

6. 第六章：模擬環境測試

以模擬環境測試本研究方法之效果，並加以分析。

7. 第七章：結論與討論

總結本研究內容，並提出未來研究可改進與發展之方向。



文獻回顧

在前一章節中提到，為了有效率地自主實現大型環境的地圖建構，必須使用多機器人技術。然而，在多機器人自主協同探索中，由於無法預期個體已知彼此初始相對姿態，所以必須使用地圖合併之技術。本文的研究重點是多機器人自主協同探索及地圖合併問題。因此接下來會探討自主探索及地圖合併的相關文獻，但在討論問題之前，必須先了解機器人地圖的不同格式種類，因此，下一節將先介紹應用於機器人地圖建構的數種資料型態。

2.1 機器人地圖資料型態

根據不同的環境資訊處理方式，地圖可以分為三種類型：

1. 格點佔據地圖 (Occupancy Grid Map)：是一種將環境劃分為一系列格點的地圖。每個格點都有一個機率值，通常分為未知、自由和佔據區域三種 [7,8]，如圖2.1所示。格點佔據地圖的優點在於解析度易於調整 [9]，並且可以快速、簡單地應用到任何環境 [8,10,11]。缺點在於在較大的場域或高解析度的情況下，地圖需要使用較多的格點，因此佔用較多的記憶體空間 [10,12]。

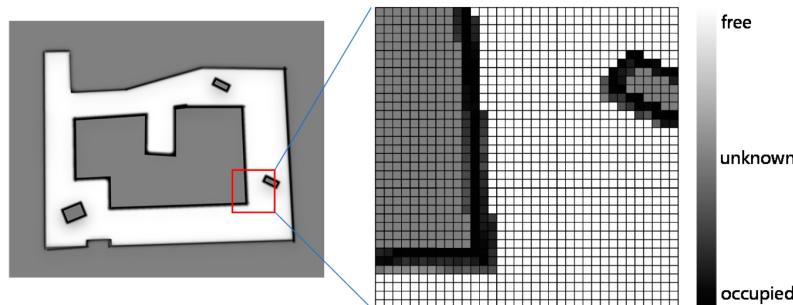


圖 2.1: 格點佔據地圖示例 [2]。

2. 特徵地圖 (Feature Map)：是一種根據收集的感測資料，擷取其中的特徵部分進行儲存的地圖，如圖2.2所示。相較於格點佔據地圖，特徵地圖只儲存部分感測資料，因此佔用較少的記憶體空間，減少多餘資料量。但是，此地圖需要一定的計算能力對感測資料進行過濾與擷取，因此計算成本較高。實際例子有：從二維光達資料中擷取出的環境直線特徵 [13] 及從深度相機的點雲資料中，擷取出的牆壁平面特徵 [14]

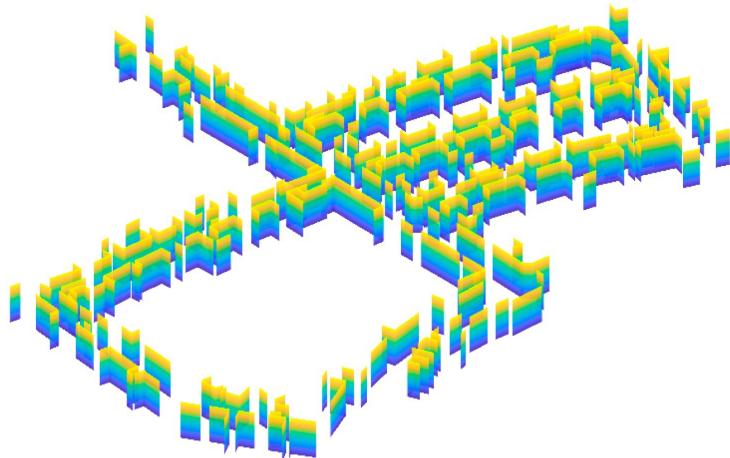


圖 2.2: 特徵地圖示例 [2]。

3. 拓樸地圖 (Topological Map)：是將環境中的醒目地標或特徵擷取出來，作為節點 (Node) 進行儲存，同時計算節點間的連結或路徑關係，儲存為邊 (Edge)，如圖2.3所示。透過節點與邊的概念，可以建構出環境地標的網絡關係，例如建築物拓樸地圖中，每個房間都可以表示為一個節點，而連接房間或區域的門或走廊則可以表示為邊。拓樸地圖不像其他地圖一樣表示環境樣貌，而是表達地標之間的相對位置和網絡關係。因此較常應用於特定任務，例如：定位 [15]、導航 [16] 及路徑規劃 [17, 18]。

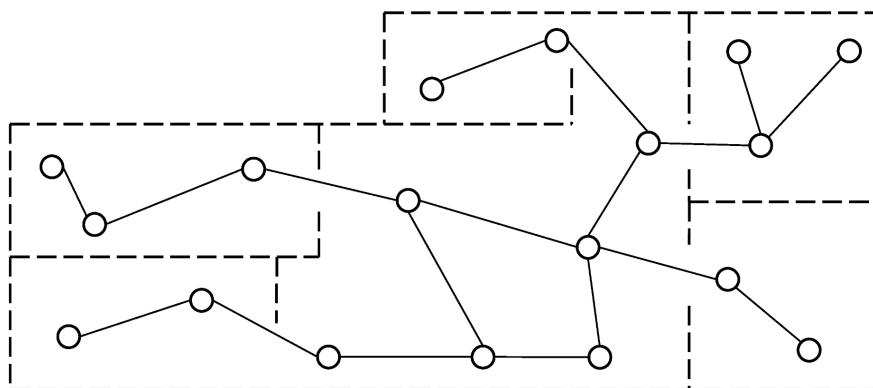


圖 2.3: 拓樸地圖示例 [2]。

三種地圖中，格點佔據地圖為最常使用的種類，具有簡易的表達形式；特徵地圖則希望改進格點佔據地圖的資料量問題，僅留下顯著特徵於各項任務使用；拓樸地圖雖無法直接表示出環境樣貌，但具有節點連結性關係對於應用任務則具有效率。本研究採以多數人使用的格點佔據地圖作為問題假設，因此接下來地圖的合併技術主要回顧格點佔據地圖。



2.2 自主地圖探索

直接類方法主要將單一機器人的技術擴展至多機器人系統上，藉由通訊找出個體感測資料的關聯性，依此建立相對關係與地圖合併。感測資料的關聯性則可依照觀察環境資訊實現，環境資訊可能包含(1)其他個體的觀察資訊、(2)個體能感測的共同物件或區域，以下就兩種不同環境資訊的方法分別進行介紹。

2.2.1 單機器人自主地圖探索

Yamauchi [5] 是最早使用基於邊界(frontier)的探索演算法的研究人員之一。它是一種可以自主探索和建構複雜環境的算法，比如堆滿家具和障礙物的辦公室。該演算法基於將機器人引向邊界，其定義是“開放探索空間和未探索空間之間的交界區域”[5]，機器人會被引導至會增加開放空間的未探索區域，直到整個區域探索。該演算法在一個真實的類似辦公室的環境中用一個機器人進行了測試。圖2顯示了檢測邊界的過程，其中機器人將嘗試探索其中最近的區域。圖中白色區域代表已知空間，虛線區域代表未知空間，檢測到的三個邊界區域用圓圈內的“+”標記。許多的探索策略沿襲 Yamauchi 的研究，因此出現了許多基於檢測邊界的探索演算法。

在[19]中，作者提出了兩種更快地檢測邊界的算法。分別是波前邊界檢測器(Wavefront Frontier Detector, WFD)和快速前沿檢測器(Fast Frontier Detector, FFD)。在WFD中，邊界搜索是通過僅掃描地圖的已知區域來完成的。WFD檢測器只搜索邊界點，當檢測到邊界邊緣上的點時，可以提取邊緣。在FFD中，只有雷射掃描包含在搜索中。此外，通過首先將雷射掃描轉換為輪廓，如果檢測到的邊界與任何先前存儲的邊界不匹配，則會創建並存儲新的邊界。

另一個與探索相關的研究方向是基於隨機運動規劃技術。這裡主要關注RRT算法。基於邊界的方法背後的主要思想是確保機器人探索未知空間並避免已經探索過的空間。這也可以通過利用基於RRT的隨機化技術的概率特徵來實現。機器人可以被引導到未探索的空間，而無需檢測邊界區域。使用RRT算法，機器人移動時形成的已探索節點樹會偏向於在未探索空間中擴展[20]。

在[21]中，作者提出了一種基於傳感器隨機樹(SRT)的探索策略，它是RRT的一種變體。在SRT中，目標探索目標是隨機生成的。這些隨機點在機器人周圍



的空間內生成，因此所有生成的點都位於用於掃描環境的傳感器的特定半徑內。這是 SRT 和 RRT 之間的一個關鍵區別，後者在整個地圖中隨機生成點。這使得 SRT 成為深度優先探索，這意味著將生成一系列隨機點，更像是一個節點鏈，而在 RRT 中，樹枝也向不同方向延伸並生長子樹枝。由於上述差異，基於 SRT 的方法需要回溯，即當分支停止增長時（例如，當機器人到達死胡同時），機器人必須返回並跟蹤當前節點的父節點。回溯過程中可能會多次訪問同一個地方。由於這是不可取的，研究人員提出了改進回溯的解決方案 [22]、[23]。

不遵循上述探索方法的不同探索算法可以歸類為基於信息的探索 [24]、[25]。他們在探索的同時考慮了定位和映射問題，因此目標探索目標的選擇方式可以降低機器人姿態估計的不確定性，從而有助於最大化地圖信息。

[26] 中提出了一種在探索中協調多個機器人的有趣方法。多機器人協調基於市場結構，其中機器人以最大化收入和最小化成本的方式被引導。信息的獲取就是收益，達到目標的預期距離就是成本。每個單獨的機器人都試圖最大化其利潤，整體效果是一個具有高生產力的機器人團隊（即高信息增益，因此更有效的探索）。

在 [27] 中，作者研究了機器人數量對探索效率的影響，他們根據探索所需的時間和團隊中所有機器人消耗的能量來衡量。結論是機器人越多越好。他們的研究中使用了基於邊界的探索算法。

2.2.2 多機器人自主協作地圖探索

在 [28] 中，Yamauchi 進一步將演算法應用於多個機器人，這些機器人以協作、分散的方式工作，並根據機器人之間共享的信息構建全局地圖，但該演算法可能會導致重複探索，因為每個機器人的探索獨立於其他機器人運行，並且邊界是在每個機器人的局部地圖計算的。因此，兩個機器人最終可能會探索相同的邊界 [19]。

在 [29] 中，作者提出了一種解決 Yamauchi 演算法在多機器人情況下存在的重複探索問題的方法。該解決方案基於將區域劃分為子區域，其中每個機器人都不會嘗試進入另一個機器人的子區域。機器人將繼續探索它的子區域，直到沒有檢測到額外的邊界區域（即子區域被完全探索）。然後機器人將進入所謂的“行走狀態”，機器人會嘗試找到下一個要探索的子區域。在 Yamauchi 的研究中 [28]，每次機器人到達目標點時都會進行邊界檢測。這降低了探索效率，因為檢測邊界邊緣的過程具有很高的計算成本 [30]，因為每次運行都必須掃描整個地圖。為了節省計算資源，通常避免頻繁計算邊界，在某些情況下，這會導致不必要的冗餘探索任務 [19]。

在 [31] 中，作者將之前基於 SRT 的方法應用於多個機器人。作者使用的基於 SRT 的方法是一種分散的方法，每個機器人首先構建自己的基於機器人初始姿勢的 SRT。當一個機器人不能再擴展自己的樹後，它會嘗試幫助其他機器人擴展他



們的樹，這就是所謂的“支持階段”。作者在 [32] 中進一步改進了這項研究，通過在樹上的可連接節點（稱為“橋”）之間引入直接快捷路徑，這有助於支持階段，並且所有機器人都會擴展一棵樹。

2.3 格點佔據地圖合併方法

在格點佔據地圖合併技術中，由於演算法和操作策略的不同，H.C Lee [33] 在其研究中將方法分為直接類和間接類。這兩種方法的邏輯架構有很大的不同，並且問題的發展也被劃分為早期和晚期。早期方法主要屬於直接類方法，而近年來提出的方法則主要屬於間接類方法。以下將介紹這兩種方法的背景、動機和方法流程。

2.3.1 直接類

直接類方法主要是透過通訊找出個體感測資料的相關性，進而建立相對關係和地圖合併。這些感測資料的相關性可以根據觀察環境資訊來實現，這些環境資訊可分為兩類：

1. **觀測個體資訊的方法**：為了解決來自不同機器人之間資料關聯的問題，最簡單直接的方法是觀察其他機器人與自己的相對位置和關係。透過感測器，機器人可以決定通訊範圍，觀察周圍的可合作對象，並利用感測資料處理與其他機器人的相對關係。

Konolige [34] 的研究中，機器人之間的互動可以被分為四個階段：無互動階段、建立假設階段、驗證假設階段和協作階段。在無互動階段，機器人之間沒有進行任何資訊交流。建立假設階段中，也就是當機器人觀察到彼此時，會開始建立通訊並開始接收其他機器人的感測數據，並嘗試建立兩者之間的相對關係。在驗證假設階段中，機器人會利用前一階段建立的相對關係，嘗試於某個地點進行會面，若會面成功，則代表假設得到驗證，兩者進入協作階段。最後，協作階段中的機器人會建立兩兩之間的相對關係，即使有些機器人未執行前述階段的步驟，仍然可以透過參照相同的機器人或協作網絡中的連鎖性來獲取相對資訊。

Zhou [35] 預先設定好會合地點，並使用全景相機偵測安裝在其他機器上的彩圓柱體，來建立地圖變換矩陣 (Map Transformation Matrix, MTM)，求得相對關係以進行地圖合併。然而，由於矩陣存在較大的誤差，Zhou 提出了一種修正地圖的方法，通過辨識環境特徵進行合併地圖的修正。

2. **辨識共同物件或區域的方法**：當機器人無法直接觀測到彼此時，可以透過參照相同物件來建立彼此之間的關係。透過參照相同物件，機器人可以形成三



角關係並計算彼此的相對位置。這些相同物件可能是環境中的物體，也可能是辨識出的相同區域。

Lee [36] 將機器人裝上向上的相機，用來觀測周圍的天花板影像，並從影像中提取和存儲影像特徵。當兩個機器人的地圖有重疊的部分時，這代表它們觀察到相同的影像特徵。因此可以使用最佳化演算法來搜尋這些地圖特徵之間的對應關係，以計算出兩個機器人之間的相對姿態差異，並完成地圖合併。

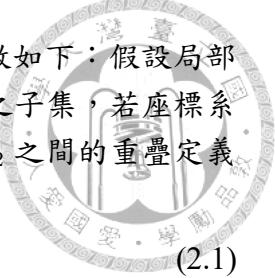
Tungadi [37] 使用全景相機和基於 Haar 小波的機率地區識別系統，來檢測機器人當前位置是否與先前存取或運行的地圖相符。如果是前者，這種方法用於閉環檢測，有助於地圖的建立；如果是後者，它可以與存取的地圖合併。Tungadi 的方法中，Haar 小波系統可以計算出當前位置相對於參考原點的角度，提高了掃描匹配演算法驗證系統輸出結果的收斂速度和準確度，因此提高了方法的穩健性。

直接類方法中，其優勢在於不受到環境特徵的影響，尤其適用於高重複性特徵環境。但是，由於觀察彼此位置要求機器人同時位於同一場域，這限制了時間分配，同時因為必須要觀測到彼此，限制了活動範圍及靈活性，進而降低了探索效率。相比之下，辨識共同環境資訊則可放寬活動範圍的限制，只要個體探索範圍曾重疊過，就可以計算相對關係，這種概念與間接類方法相似，但是直接類方法需要儲存大量資料。以 Lee [36] 的方法為例，需要儲存具有特徵的小部分影像圖片，隨著時間的推移，所需儲存的圖片數量也越來越多，因此佔用的儲存空間很大。

2.3.2 間接類

間接類方法是基於地圖影像進行處理的方法。地圖可以看作是圖片影像的一種，利用格點像素的縮放、旋轉、平移等方式進行對齊並完成地圖合併。對齊方式主要可以分為兩種，一種是利用最佳化演算法來搜尋地圖影像之間的相對關係，另一種是透過擷取與匹配地圖影像的特徵來計算相對關係。下面將介紹這兩種不同的對齊方式：

1. 最佳化搜尋方法：最佳化搜尋方法中，必須先定義地圖像素相關的目標函數，並使用演算法進行求解。根據格點佔據地圖的分類特性，目標函數可定義為格點像素種類的正確配對最大化或錯誤配對最小化，如「自由區-自由區」、「佔據區-佔據區」的正確配對，或「自由區-佔據區」的錯誤配對。而演算法的選擇則需根據問題的特性進行選擇，地圖合併的目標函數並非凸函數，可行解區域中可能具有數個極值點，且局部最大、最小值點可能非常鄰近。因此在常見的解決方法中常使用的是啟發式演算法。



以 Carpin et al. [?] 的方法為例，其首先定義了地圖重疊函數如下：假設局部地圖 m_1 與 m_2 屬於一大小為 $[N \times M]$ 的全域地圖 $m_{N \times M}$ 之子集，若座標系統一的情況下，Carpin et al. [?, 7] 將兩個局部地圖 m_1 與 m_2 之間的重疊定義如下：

$$\omega(m_1, m_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} Eq(m_1[i, j], m_2[i, j]) \quad (2.1)$$

其中函數 $Eq(m_1[i, j], m_2[i, j])$ 表示若格點 $[i, j]$ 位置的兩像素值相同，則輸出 1，反之輸出 0，根據上述函數定義，則可將目標函數定義為

$$\max_{x, y, \theta} \omega(m_1, T_{x, y, \theta}(m_2)) \quad (2.2)$$

其中操作變數為 x, y, θ ，地圖相對變換關係矩陣為

$$T_{x, y, \theta} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & x \\ \sin(\theta) & \cos(\theta) & y \\ 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

完成目標函數的定義後，Birk 則使用適應性隨機漫步（Adaptive Random Walk, ARW）[38] 演算法進行求解。

另一研究中，Li [8] 也使用了類似的目標函數，但改為使用基因演算法 [39] 進行求解。Li 同時也提出相關的應用情境，在室外環境中，車輛行進時可能會產生感測器的遮蔽而無法得知環境資訊，藉由地圖合併方法，可以求得車輛間的相對關係，進而從其他個體取得受遮蔽環境的資訊。

2. 特徵擷取方法：地圖影像大致描繪了環境的幾何關係，因此可擷取格點像素的特徵進而達到擷取環境特徵的效果。格點像素的特徵相當多元，可能是滿足空間結構關係的幾何特徵 [40]，也可能是線段角度轉換的頻譜特徵 [41]。根據使用的特徵不同，其演算方式也具有差異。

Park [40] 的方法於格點佔據地圖的自由區由大至小放置矩形，並將矩形分佈進行方位分區。於各區中將矩形兩兩連接生成直線，從中計算可代表該區斜率的直線角度，後將地圖旋轉至正交朝向並重新放置矩形，此時應只剩下數個大面積矩形。接著將最大的三個矩形連線形成三角形，以此作為匹配、對齊兩地圖的依據。根據三角的幾何特性，此方法不受限於旋轉與比例差異。

Carpin [41] 提出非迭代且快速的地圖合併演算法，針對格點佔據地圖，其使用了霍爾變換（Hough Transform）[42] 計算旋轉參數的候選集合，平移參數的候選集合則是把地圖影像投影後以直方圖進行計算。根據兩候選集合的組



合，使用接受指標（Acceptance Index，AI）[7] 挑選出用於地圖合併的變換關係。此研究中，對於格點佔據地圖的匹配，接受指標相當適合用於判斷地圖合併成功與否，概念與先前提及的區域種類的正確、錯誤配對類似，實際定義如下：

$$\omega(M_1, M_2) = \begin{cases} 0 & , \text{if } agr(M_1, M_2) = 0. \\ \frac{agr(M_1, M_2)}{agr(M_1, M_2) + dis(M_1, M_2)} & , \text{if } agr(M_1, M_2) \neq 0. \end{cases} \quad (2.4)$$

其中 $\omega(M_1, M_2)$ 為接受指標， $agr(M_1, M_2)$ 代表兩地圖 M_1 、 M_2 的正確格點種類配對的個數， $dis(M_1, M_2)$ 則代表錯誤配對的個數。Carpin 的方法因使用霍爾變換，對於具有特定空間結構的環境表現較為良好，如：方方正正的房間，因此較適合使用於室內環境。然而，此方法具有部分限制，例如不同地圖需具有相同的比例尺，同時也必須具有足夠的重疊區域。

Ferrão [43] 則使用尺度不變特徵轉換（Scale Invariant Feature Transformation，SIFT）[44] 演算法對格點佔據地圖進行特徵擷取。此演算法求取地圖影像中可作為穩健特徵的極值格點，常用於電腦視覺領域。Ferrão 的方法將不同地圖各自擷取特徵後，依照 SIFT 輸出的描述符進行特徵匹配。由於已建立兩地圖間的特徵配對集合，故可直接計算出相對變換關係並合併地圖。Ferrão 類似的方法常見於影像處理的全景圖合併、圖像縫合問題[45]，不同的是，一般影像圖片的紋理、明度、色彩特徵較多，可擷取出較多特徵；而格點佔據地圖的像素往往趨於三值（白、灰、黑，分別代表自由區、未知區、佔據區），因此可擷取的特徵較少。

間接類方法僅使用地圖影像儲存的資料，因此不受限於時間、探索範圍的限制，具有相當的靈活性。通訊上，不同於直接類方法傳遞感測資料，地圖影像的通訊要求較低，可避免資料遺失。近年來，許多最佳化演算法與特徵擷取演算法已被提出，演算效率與精確度都有良好表現，因此應用於間接類方法時可有效提高合併正確性。這種方法需要先將影像轉換成數字矩陣，然後使用數學模型來描述矩陣之間的相對關係。儘管最佳化演算法能夠有效地處理不同的地圖影像，但是在大規模地圖合併時需要消耗大量的計算資源，因此執行速度較慢。

然而，在間接類的方法中，最佳化搜尋方法使用時需要良好的初始估計值，儘管最佳化演算法能夠有效地處理不同的地圖影像，但是在大規模地圖合併時需要消耗大量的計算資源，因此執行速度較慢，且不保證全域最佳解的收斂。特徵擷取方法擷取特徵時包含複雜演算，與最佳化演算法相比，特徵擷取算法能夠快速地進行大規模地圖合併，因為它只需要對影像中的關鍵點進行計算，而不需要對整個影像進行處理，但應用於格點佔據地圖時，可取得的特徵數量較一般 RGB 影像少，同時運用在高重複特徵環境時，相似特徵太多造成無法成功配對。



2.4 文獻回顧總結

回顧與分析多機器人自主協作地圖探索及地圖合併的研究，可以發現現有的研究多著重在單一方面，缺乏將兩者結合在一起的探討。然而在實際應用中，無法預期各個機器人已知彼此的初始相對姿態，因此將探索和合併兩者結合在一起是必要的。同時如上述所提及，在格點佔據地圖進行特徵擷取時，尤其是在面對高重複特徵的環境時，可用特徵數量較少且難以配對，這是解決地圖合併問題的一大挑戰。因此，本研究旨在探討如何將未知初始相對姿勢的多機器人自主協作地圖探索和地圖合併結合在一起，並解決高重複特徵地圖的合併問題。在第三章中，首先會具體地針對本研究核心問題進行描述，給出完整數學定義，並可以看到本研究基於 [1] 的方法發展基於通訊信號強度預測距離之未知初始相對姿勢自主多機器人協同探索，因其有著完整的多機器人自主協作地圖探索架構，並在 ROS 中被廣泛使用。並在第四章詳細介紹 [1] 中的現有演算法，如 SIFT 演算法、RANSAC 演算法。而第五章將可以到本研究將未知初始相對姿勢加入考量，並針對高重複特徵地圖提出了 [1] 的流程修正，並說明效用。最後以模擬環境測試本研究方法，並進行討論與總結。



第三章

研究流程與架構

前兩章節已經說明了本文欲研究的格點佔據地圖合併之背景，此章節將針對格點佔據地圖合併進行更詳細的問題定義，並說明本文使用的測試案例與研究方法的流程及架構。

3.1 問題描述

處理合併格點佔據地圖的問題之前，我們必須先定義格點佔據地圖。由於格點佔據地圖可視作影像圖片的一種，因此本研究將其定義為二維矩陣 M_i ，矩陣大小為 $r_i \times c_i$ ，如式 (3.1)：

$$M_i = \begin{bmatrix} m_{11} & \cdots & m_{1c_i} \\ \vdots & \ddots & \vdots \\ m_{r_i1} & \cdots & m_{r_ic_i} \end{bmatrix} \quad (3.1)$$

其中，矩陣 M_i 的元素代表影像像素值，數值範圍如下：

$$M_i(x, y) = m_{xy} \in [0, 255], \quad x = 1, \dots, r_i, \quad y = 1, \dots, c_i \quad (3.2)$$

本研究以影像像素值表達格點佔據地圖的區域分類資訊，因此主要以三個整數作為表示，即

- 佔據區域：像素值為 0。
- 未知區域：像素值為 128。
- 自由區域：像素值為 255。



關於像素值與區域分類的詳細關係可參見附錄A。

定義完格點佔據地圖後，我們接著討論地圖之間的相對變換關係，因為建構地圖時，機器人的參考原點與地圖解析度（比例尺）不一定相同，所以必須藉由計算相對變換關係來將地圖進行旋轉、縮放與平移，使得地圖的比例尺統一，並且座標系對齊。因此，我們給定兩格點佔據地圖為 M_1 、 M_2 ，假設存在一相對變換關係 $T_{s,\theta,t_x,t_y}^{12}$ 可將地圖 M_1 進行變換，使其座標系與比例尺可與地圖 M_2 對齊及統一：

$$T_{s,\theta,t_x,t_y}^{12} = \begin{bmatrix} s \cos \theta & -s \sin \theta & t_x \\ s \sin \theta & s \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

式(3.3)中，以地圖 M_2 作為參考， s 表示地圖 M_1 與 M_2 的比例差異， θ 表示兩者的旋轉差異， t_x 與 t_y 則表示兩者的平移差異。

經過相對變換後，兩地圖 M_1 與 M_2 已統整比例與座標系關係，因此可看出兩地圖已有部份重疊的格點。對於每一重疊的格點，其格點像素值皆具有兩個數值，分別來自地圖 M_1 與 M_2 。因此，可根據此兩個像素值將重疊格點的合併進行區分，區分方式如同第二章提及的正確匹配、錯誤匹配，若

- 兩像素值所代表的區域類型相同，且並非未知區域，即「佔據區域—佔據區域」、「自由區域—自由區域」，則屬於正確匹配 (*agr*)。
- 兩像素值所代表的區域類型相異，且並非未知區域，即「佔據區域—自由區域」，則屬於錯誤匹配 (*dis*)。
- 任一像素值所代表的區域中為未知區域，則不予計算。

根據上述的正確、錯誤匹配計算，可加以採用 Birk [7] 的接受指標（式(2.5)）來評估地圖 M_1 、 M_2 在經過相對變換後的重疊結果。理論上，地圖 M_1 的格點若經過正確的變換後應與相同的區域類型重疊，即正確匹配。因此，通過正確的變換，可使得正確匹配數量多，錯誤匹配數量少，使得接受指標 ω 接近於 1。根據這個概念，我們可藉由設定一閾值，評估是否採用此相對變換，並且將兩地圖的重疊情況作為地圖合併的結果：

$$\omega(M_1, M_2) > \omega_{thres} \quad (3.4)$$

3.2 研究方法

前一節已針對格點佔據地圖進行定義，同時也給定評判合併地圖結果的量化指標。接下來將介紹本文研究方法的流程與架構，我們將簡介研究方法的基礎背

景以及延伸發展。最後展示數個作為測試用的地圖，並且說明每個地圖具有的特性。

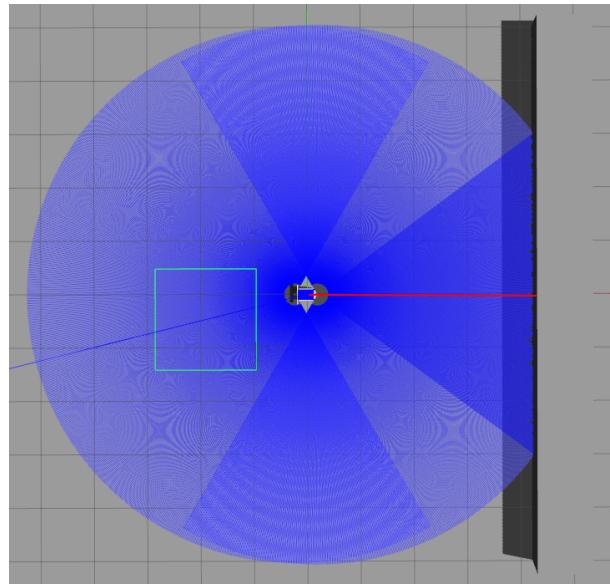


3.2.1 流程與架構

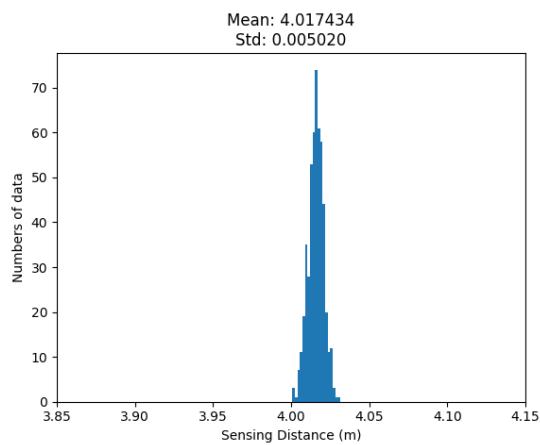
本文研究方法主要參考 Ferrão [43] 以及其他影像縫合 [46] 中所使用的概念，如圖3.2，意即使用特徵擷取演算法取得格點佔據地圖的特徵，並搭配匹配演算法進行特徵的配對，最後計算出地圖的相對變換關係。求得相對變換關係後，則可使用前一節提到的接受指標（式（3.4））判斷是否接受該變換關係，若接受則可採用熵濾波技術輸出合併地圖的結果。上述作為本文研究的參考方法將於第四章進行演算法的詳細介紹，並且搭配測試用地圖展示各演算法的效果。

然而，在之前的章節已提及，格點佔據地圖作為影像進行處理時，其像素值主要為3個整數（0、128、255），用於表示地圖的3個格點區域種類（佔據、未知、自由區域）。因此，相較於一般影像包含的256個整數像素值（0~255），像素變化來得較小，所以特徵擷取演算法可取得的特徵數量相對較少。另外，SLAM方法在建構格點佔據地圖時，可能受到感測器的誤差以及演算的數值誤差影響，使得格點佔據地圖產生扭曲與雜訊，如圖3.1。因此，若直接採用圖3.2的方法時，可能造成特徵的錯誤匹配，使得計算出的相對變換關係無法通過接受指標，並且完成地圖合併。

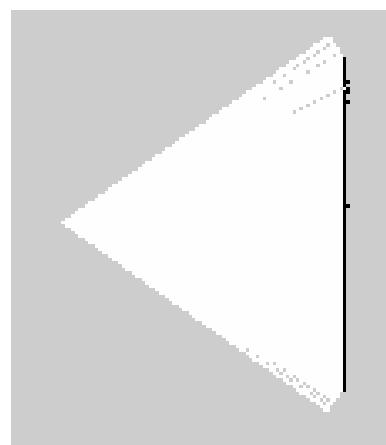
根據上述提及的原因，本研究在參考方法之前加入影像處理方法，如圖3.3。首先將格點佔據地圖的像素值利用二值化處理進行轉換，將像素區分為0與255，分別表示非佔據區域與佔據區域。雖然二值化處理降低了像素數值的差異，但是剔除了未知區域的不確定性，並且強化了佔據區域的特色。接著採用雷登變換（Radon Transform）擷取出地圖線段主要的方向，並將多數斜線旋轉至與座標軸平行。由於斜線以格點進行表示時是階梯狀，因此在特徵擷取時可能產生不必要的特徵點。接著我們利用形態學的概念，將影像進行閉運算處理，彌補旋轉時造成的瑕疵缺口。根據旋轉後的影像，我們在佔據區域的線段中，擷取滿足特定遮罩的興趣點，最後利用興趣點進行主體結構線段的修正。經過此影像處理流程，可將SLAM方法產生的誤差進行修正，促進後續計算出正確的相對變換關係以及完成地圖合併。本文研究方法將於第五章進行詳細的說明，並且搭以測試用地圖展示每個步驟的效用。



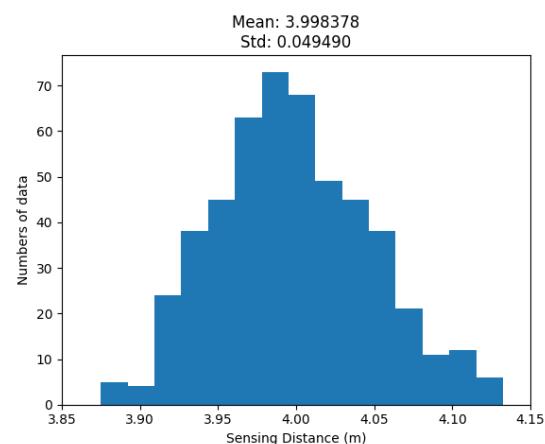
(a) 作為展示感測誤差之模擬環境。中心處為機器人本體，藍色區域為光達掃描範圍，紅色線為測量光達誤差的距離，右側長條物體為環境之牆壁。



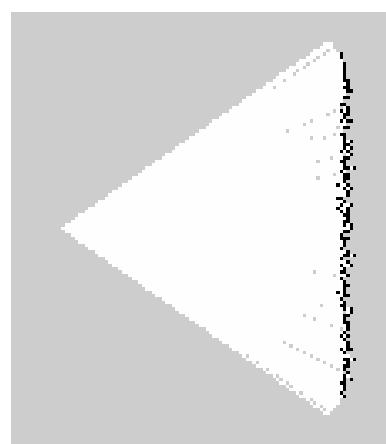
(b) 調整光達誤差之標準差為 0.005 的距離測量情況



(c) 調整光達誤差之標準差為 0.005 的地圖建構情況。誤差較小時，較為筆直。



(d) 調整光達誤差之標準差為 0.05 的距離量測情況。



(e) 調整光達誤差之標準差為 0.05 的地圖建構情況。誤差較大時，較為扭曲。



參考方法 現有影像縫合技術

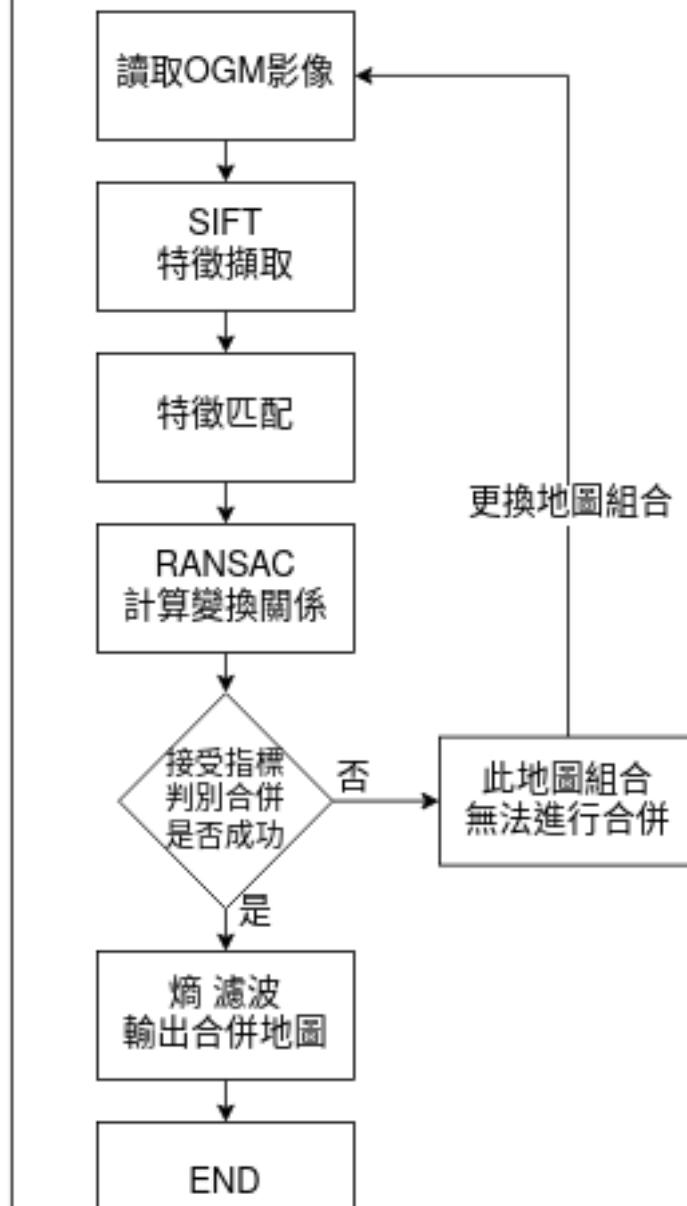


圖 3.2: 本文研究之參考方法

研究方法 地圖修正之合併方法

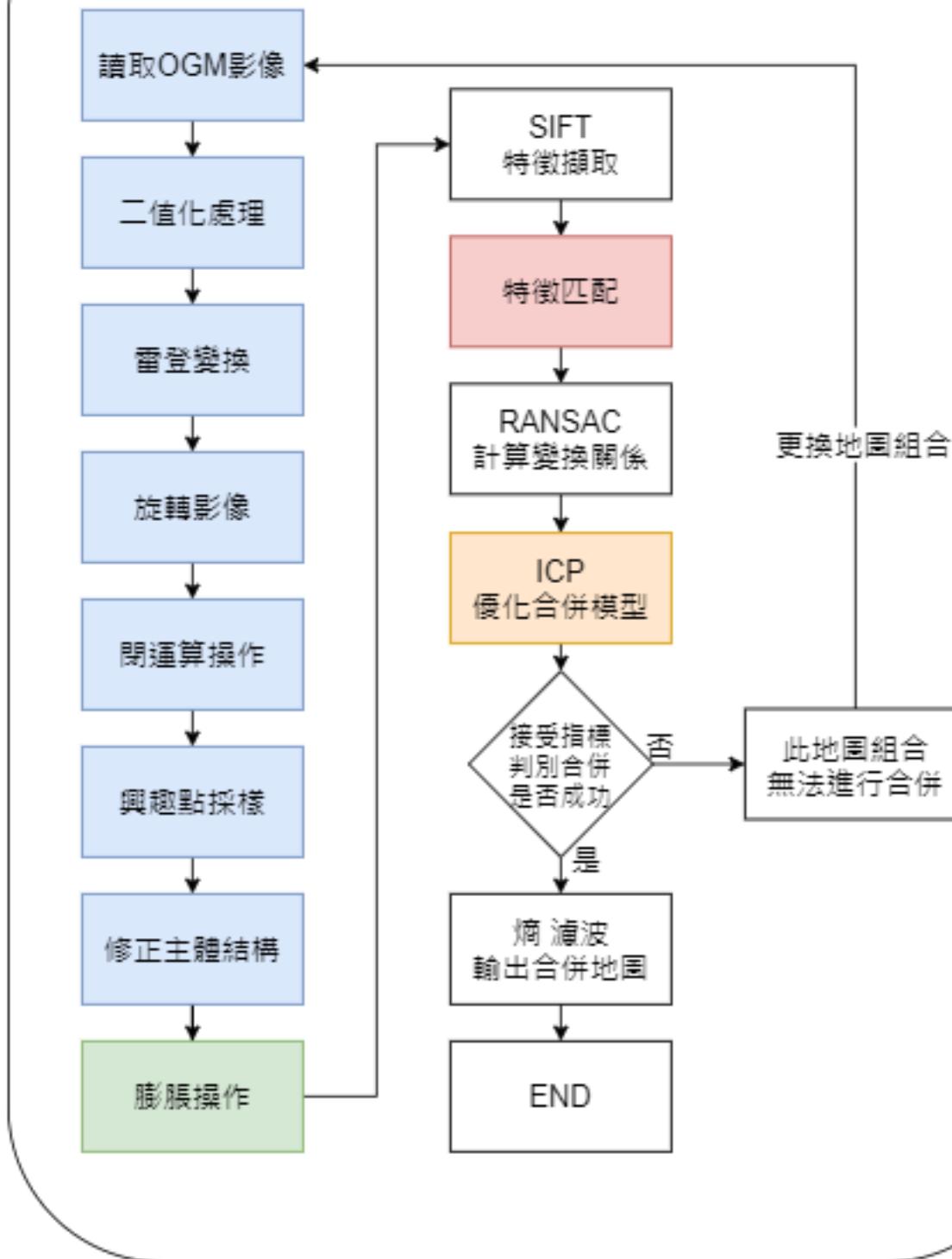


圖 3.3: 本文研究方法，主要在參考方法之前加入影像處理，進行格點佔據地圖的修正。



3.2.2 測試案例

前一節已說明本研究參考的方法背景以及我們額外加入的影像修整步驟，分別會在第四、第五章進行詳細說明。針對其中所使用到的每一個演算法，我們將個別應用於下列測試案例上，展示演算法效用。下列的測試案例（圖3.4）中，每一個局部測試圖（圖3.4(b)～3.4(e)）皆於模擬環境（圖3.4(a)）中進行建構，並且分別加入了不同參數，例如調整感測器誤差、解析度改變，使得每一測試用地圖具有不同的特性，可有效檢測出目前演算方法的不足之處，詳細的參數將條列於表3.1：

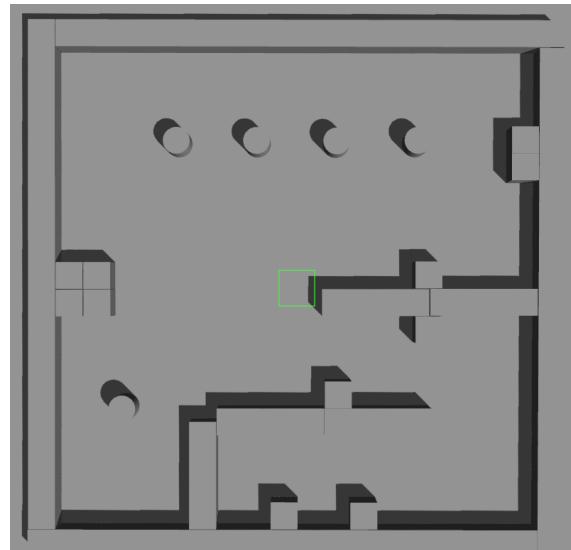
測試地圖	解析度（比例尺） (格點數/公尺)	光達感測誤差 (標準差(公尺))
局部測試圖一（圖3.4(b)）	20	0.005
局部測試圖二（圖3.4(c)）	20	0.005
局部測試圖三（圖3.4(d)）	10	0.005
局部測試圖四（圖3.4(e)）	20	0.05

表 3.1: 測試案例之地圖組合

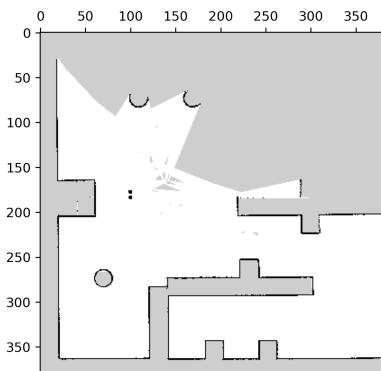
在測試案例地圖中，我們需將地圖進行組合進行地圖合併的測試，地圖組合可見於表3.2。此表格中，我們也列出每個組合的參數差異關係，因此測試時必須考量所列出的參數。

	輸入地圖一	輸入地圖二	參數關係
組合一	局部測試圖一 (圖3.4(b))	局部測試圖二 (圖3.4(c))	旋轉、平移差異
組合二	局部測試圖一 (圖3.4(b))	局部測試圖三 (圖3.4(d))	旋轉、平移、縮放差異
組合三	局部測試圖一 (圖3.4(b))	局部測試圖四 (圖3.4(e))	旋轉、平移、感測誤差差異

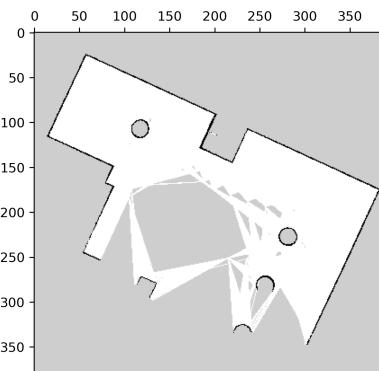
表 3.2: 測試案例之地圖組合



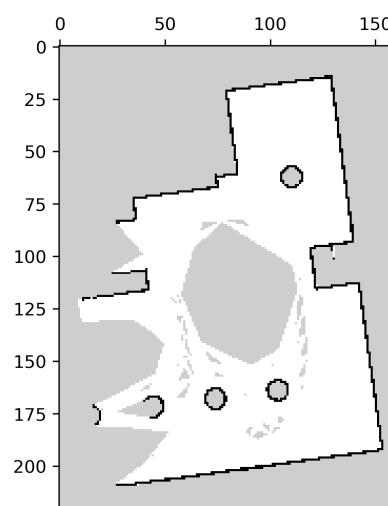
(a) 研究測試案例 - 模擬環境



(b) 研究測試案例 - 局部測試圖一



(c) 研究測試案例 - 局部測試圖二



(d) 研究測試案例 - 局部測試圖三
(調整解析度)



(e) 研究測試案例 - 局部測試圖四
(調整感測誤差)

圖 3.4: 本文研究測試用地圖。座標軸刻度用於表示影像格點數量，解析度高的地圖格點數較多，解析度低的格點數則較少。



現有流程之技術

第三章已簡述本研究參考方法的背景，即影像縫合相關技術 [46] 與 Ferrão [43] 的方法。在 Wang [46] 對於影像縫合問題的整理中，我們發現將兩影像進行縫合時，可將影像進行特徵擷取與匹配，並以此計算影像的相對變換關係。在第二章回顧地圖合併的方法中，Ferrão 也提出類似的概念。然而，在影像的重疊區域處理上，影像縫合技術希望將兩影像的銜接部分進行平滑、潤化的處理，而 Ferrão 則是將重疊格點的像素值進行平均。本研究認為，格點佔據地圖的格點訊息來自於感測資料的存取，因此不適宜進行影像縫合的平滑處理。而平均處理則可能造成像素值的訊息混亂，即佔據區域（像素值 0）與自由區域（像素值 255）平均處理後趨於未知區域（像素值 128），我們認為平均處理會使得地圖的不確定性加大。

基於上述原因，本研究另外參考 Saeedi [47] 方法中的熵濾波處理（Entropy Filter）[48]。熵濾波處理可針對機率計算出對應的熵值，並以此判斷不確定性的大小。由於格點佔據地圖的格點資訊也包含對應的機率數值，因此我們採用此技術進行合併處理。

接下來的部份，我們將介紹參考方法（圖3.2）中所使用的各項演算法，即

- 用於擷取特徵的尺度不變特徵轉換演算法（SIFT）
- 用於匹配特徵的歐氏距離（Euclidean Distance）
- 用於計算相對變換關係的隨機採樣共識演算法（RANSAC）
- 用於評判重疊情況的接受指標（Acceptance Index）
- 用於合併地圖影像的熵濾波演算法（Entropy Filter）

最後，我們將以圖3.4的測試案例，進行各演算法的展示，並且說明效用與需改善之處。



4.1 特徵辨識

本小節將介紹如何利用尺度不變特徵轉換（Scale Invariant Feature Transform，SIFT）對影像進行特徵擷取，並詳細說明 SIFT 演算法的原理與流程。

SIFT 於 1999 年由 David Lowe [44] 提出，其針對影像進行高斯差分與建立影像金字塔，在影像的尺度空間中尋找可作為特徵的極值點，計算該特徵點周遭位置的梯度向量，並依直方圖進行統計後，作為該特徵點的特徵向量。此演算法所計算出的特徵點具有尺度縮放、旋轉、亮度之不變性，對於影像雜訊亦相當穩健。下列將介紹 SIFT 演算法的流程，主要可分為四個步驟：

1. 尺度空間之極值檢測
2. 特徵點定位
3. 特徵點方向計算
4. 特徵點描述

4.1.1 尺度空間之極值檢測

為了在尺度空間中檢測穩定的特徵點，須對影像進行模糊化以取得影像於不同尺度空間的資訊。Koenderink [49] 與 Lindeberg [50] 說明在合理假設下，高斯函數是唯一可用於尺度空間的方法。因此，檢測特徵點位置前，必須針對影像進行高斯模糊。影像的尺度空間可以函數表示：

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (4.1)$$

其中 $I(x, y)$ 表示原始影像，運算子 $*$ 表示卷積操作，而

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (4.2)$$

表示高斯函數。

為完整搜尋尺度空間，影像將經過影像金字塔與高斯模糊的處理。影像金字塔以層級 (Octave) 進行分層，數量根據影像大小決定，每一層又包含數個不同的高斯模糊影像。舉例來說，最底層為原始影像大小，包含數個 $L(x, y, \sigma)$, $L(x, y, k\sigma)$, $L(x, y, k^2\sigma)$, ... 尺度空間函數。每上一個層級，原始影像進行降採樣，並重複進行高斯模糊，而參數改為前一層級的 2 倍，即 $L(x, y, 2\sigma)$, $L(x, y, 2k\sigma)$, $L(x, y, 2k^2\sigma)$, ... 尺度空間函數。

建立完影像金字塔後，為實現尺度不變性，Lindeberg [50] 說明必須使用參數為 σ^2 的歸一化拉普拉斯算子，同時此算子可使用高斯差分函數 (DoG) 進行近似。

故在極值檢測時，Lowe [44] 應用了高斯差分函數，運算僅需將兩個不同尺度空間的影像進行相減：

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \quad (4.3)$$



相關影像金字塔與高斯差分函數的操作可見於圖4.1。

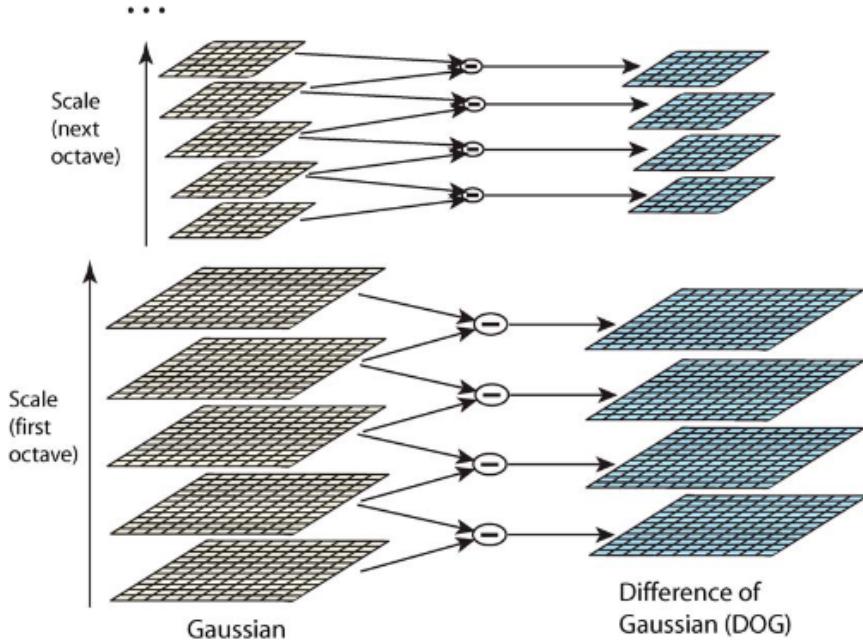


圖 4.1: 影像金字塔與高斯差分函數計算示意圖 [3]

接著，搜尋各層高斯差分函數的影像極值點作為特徵點，特徵點須為同尺度空間周圍 8 個點與上、下兩尺度空間鄰近各 9 點中的最大或最小值，如圖4.2。

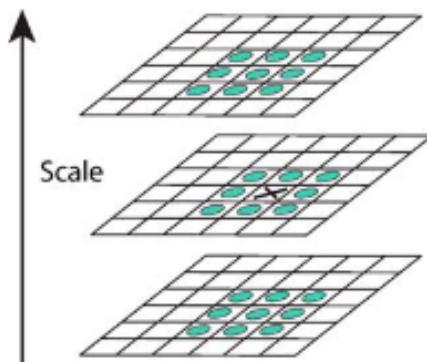


圖 4.2: 特徵點極值比對示意圖 [3]



4.1.2 特徵點定位

根據前一小節擷取的特徵點為離散空間的極值點，位置並非完全精確，同時可能包含對影像雜訊敏感的低對比度或受到高斯差分函數增強邊緣響應而較不穩定的特徵點，因此通過擬合三維二次函數進行處理。

根據高斯差分函數的泰勒展開式（已將原點平移至特徵點）：

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}} + \frac{1}{2} \mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2} \mathbf{x} \quad (4.4)$$

其中， D 表高斯差分函數特徵點， $\mathbf{x} = (x, y, \sigma)$ 表相對偏移量。

將式 (4.4) 對 \mathbf{x} 微分後設為零，可求得極值偏移量 $\hat{\mathbf{x}}$:

$$\hat{\mathbf{x}} = -\frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1} \frac{\partial D}{\partial \mathbf{x}} \quad (4.5)$$

若 $\hat{\mathbf{x}}$ 中任一變數大於 0.5，則代表有另一特徵點更接近極值，因此捨去此關特徵點。將式 (4.5) 代回式 (4.4)，得更新後高斯差分函數特徵點:

$$D(\hat{\mathbf{x}}) = D + \frac{1}{2} \frac{\partial D^T}{\partial \mathbf{x}} \hat{\mathbf{x}} \quad (4.6)$$

另外，若 $D(\hat{\mathbf{x}}) < 0.03$ (像素值範圍假設為 $[0,1]$)，表示其對比度過低，為不穩定特徵點，故捨去。另外，位於邊緣的特徵點，其高斯差分函數也有較強的響應，因此可能較為不穩定。不穩定邊緣特徵點的高斯差分函數在橫跨邊緣時具有較大的曲率，在垂直方向上則較小。藉由比對曲率的比例關係，則可評判是否捨去該位於邊緣的特徵點。曲率的比例關係與該特徵點位置的 Hessian 矩陣特徵值比例相同，因此計算 Hessian 矩陣:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (4.7)$$

假設 Hessian 矩陣的特徵值為 λ_1 與 λ_2 ，其中 $\lambda_1 > \lambda_2$ ，則可利用矩陣特性計算特徵值總和與特徵值乘積:

$$\begin{aligned} Tr(H) &= D_{xx} + D_{yy} = \lambda_1 + \lambda_2 \\ Det(H) &= D_{xx}D_{yy} - (D_{xy})^2 = \lambda_1 \lambda_2 \end{aligned} \quad (4.8)$$

假設兩特徵值的比例關係為 $\lambda_1 = r\lambda_2$ ，則利用式 (4.8) 關係可得:

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\lambda_1 + \lambda_2)^2}{\lambda_1 \lambda_2} = \frac{(r\lambda_2 + \lambda_2)^2}{r(\lambda_2)^2} = \frac{(r+1)^2}{r} \quad (4.9)$$



若比例 r 越大，式 (4.9) 的值也越大。因此僅須設定一閥值 r_{th} 便可決定是否捨去該邊緣特徵點。

4.1.3 特徵點方向計算

完成特徵點定位後，根據周圍影像資訊計算該特徵點方向，使其具有旋轉不變性。影像資訊來自於該關鍵點同一尺度空間的模糊影像，以便維持尺度不變性。對模糊影像中各點 $L(x, y)$ 計算梯度：

$$\begin{aligned}m(x, y) &= \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \\ \theta(x, y) &= \tan^{-1} \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\end{aligned}\quad (4.10)$$

並以特徵點為中心，將周遭像素的梯度方向進行直方圖統計，每 10 度為一區間，共統計為 36 個區間。每一像素方向進行統計時，會根據其梯度大小與高斯函數（標準差為該尺度空間標準差的 1.5 倍）進行加權，使較遠的像素影響降低，並平滑直方圖。直方圖中最大值為該特徵點主方向，若有其他極大值大於最大值的 80% 則作為輔助主方向，因此一特徵點可能具有多個主方向，但僅佔少數。

4.1.4 特徵點描述

求得特徵點的位置、方向資訊後，接著需以一組向量描述，稱為描述符 (Descriptor)，使特徵點可去除亮度、視角變化造成的影響。

首先以特徵點為中心，將周遭 16×16 個像素的梯度方向相對於特徵點主方向進行旋轉修正，確保旋轉不變性。接著分為 4×4 個子區域，每一子區域像素的梯度方向進行直方圖統計，每 45 度為一區間，共 8 個區間，如圖4.3。統計過程中，同樣採取高斯函數進行加權，降低距離較遠的像素造成的影響，並平滑化。將 16 個子區域的直方圖統計結果定義為 128 維的描述符向量，為了避免亮度影響，該向量最後會經過正規化，成為單位向量。

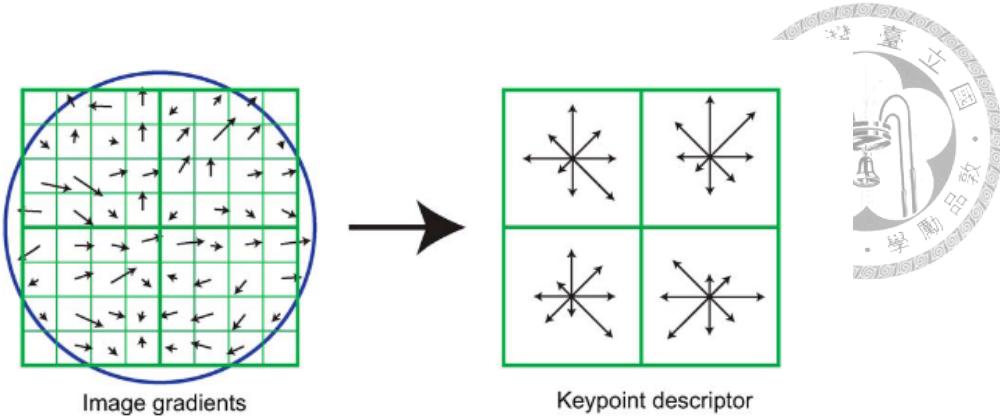


圖 4.3: 影像梯度與角度統計 [3]。左圖為影像梯度計算，圓圈表示高斯函數權重，距離圓心越遠權重越小；右圖為統計各子區域的角度區間。此圖僅以 2×2 個子區域示意，實際計算時子區域數量為 4×4 個。

4.2 特徵點匹配

前一小節已介紹了利用 SIFT 演算法進行影像特徵點擷取，假設地圖影像 M_1 所擷取出的特徵點集合為 $P = \{p_1, p_2, \dots, p_n\}$ ，地圖影像 M_2 所擷取出的特徵點集合為 $Q = \{q_1, q_2, \dots, q_m\}$ 。在兩個特徵點集合中，每一個特徵點皆具有 128 維的描述符向量，因此可利用歐氏距離比對特徵點向量的相似度：

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} \quad (4.11)$$

式 (4.11) 為歐式距離的計算，代數 x, y 為欲計算相似度的兩個向量，有下標的代數 x_i, y_i 分別表示向量 x, y 的分量。

接著我們利用式 (4.11) 進行地圖特徵點的匹配。首先，以集合 P 的各特徵點 $\{p_i\}_{i=1,\dots,n}$ 作為參考，一一在集合 Q 中利用歐式距離查找最相似的特徵點，因此可找出 n 組特徵點匹配 $\{p_i, q_i\}_{i=1,\dots,n}$ 。接著，再以集合 Q 的特徵點 $\{q_j\}_{j=1,\dots,m}$ 作為參考，一一在集合 P 中查找最相似特徵點，因此可找出 m 組特徵點匹配 $\{p_j, q_j\}_{j=1,\dots,m}$ 。由於兩組特徵點匹配中可能具有相同的配對關係，因此需要剷除重複的部份，即 $\{p_i, q_i\}_{i=j} = \{p_j, q_j\}_{j=i}$ 。最後將剩餘部份統整，可得出特徵點匹配 $\{p_i, q_i\}_{i=1,\dots,N}$ ，可用於下一節的相對變換關係計算。

4.3 計算變換關係

本小節將利用歐式距離組合出的特徵點配對 $\{p_i, q_i\}_{i=1,2,\dots,N}$ 計算地圖影像間的相對變換關係 $T_{\Delta s, \Delta \theta, \Delta x, \Delta y}$ 。

由於特徵點擷取時，可能受到影像背景影響以及數值運算的誤差，即便相

同的物件也可能產生描述符向量的誤差。因此進行匹配時，可能產生特徵點的錯誤配對。所以利用特徵點配對來計算地圖的相對變換關係時，必須排除錯誤配對的特徵點，現有方法中常導入隨機採樣共識演算法 [51] (RANdoN SAmples Consensus, RANSAC) 進行處理。此方法可將輸入資料點中不符合模型的離群 (Outlier) 資料排除，有效找出符合模型的內群 (Inliner) 資料及其模型。RANSAC 演算法利用多次隨機採樣進行資料點測試，以下將介紹基本概念，並說明如何應用於本文研究。

4.3.1 隨機採樣共識演算法

給定一資料集合 X ，欲計算符合該資料集合的模型 $h(X)$ 。資料集合 X 中含有符合模型 $h(X)$ 的內群資料 $\{x_i^{Inlier}\}_{i=1,2,\dots,n_I}$ 與不符合模型的離群資料 $\{x_i^{Outlier}\}_{i=1,2,\dots,n_O}$ 。若隨機選取集合 X 中任意一點，其屬於內群的機率是內群資料點數量與所有資料點數量的比值，即

$$w = \frac{n_I}{n_I + n_O} \quad (4.12)$$

根據內群機率 w ，可定義下列幾項機率：

- w^n ：表示任選 n 點皆為內群的機率。
- $1 - w^n$ ：表示含有至少一個離群點的機率。
- $(1 - w^n)^k$ ：表示重複 k 次選取皆含有離群資料點的機率。

假設重複選取 k 次後，未含有離群資料點的成功機率為 p ，則

$$\begin{aligned} 1 - p &= (1 - w^n)^k \\ p &= 1 - (1 - w^n)^k \end{aligned} \quad (4.13)$$

，由於一般情況往往無法得知 w 的實際值，因此若希望成功機率 p 越大，則可以調整參數 n 與 k 。參數選擇上， n 基本是依照模型 $h(X)$ 選定，主要以調整 k 為主。然而， k 值大小決定 RANSAC 演算法計算模型的正確性與迭代次數， k 值越大，計算出的模型越可能符合內群資料，故參數選擇上主要以正確性與迭代次數進行權衡。

4.3.2 以 RANSAC 演算法計算相對變換關係

前一節中，我們已取得兩地圖影像的特徵點配對 $\{p_i, q_i\}_{i=1,2,\dots,N}$ ，其中 p_i 、 q_i 分別表示 M_1 、 M_2 的第 i 個已配對特徵。令此配對作為輸入 RANSAC 演算法的資料集合 X ，即 $X = \{p_i, q_i\}_{i=1,2,\dots,N}$ ，接著可計算兩地圖影像的相對變換關係



$T_{s,\theta,t_x,t_y}^{12}$ (式3.3)。此相對變換關係 $T_{s,\theta,t_x,t_y}^{12}$ 含有四個未知變數，即縮放 s 、旋轉 θ 、平移 (t_x, t_y) ，因此每次需從資料集合 X 中採樣至少 2 組點對（可建立 4 條方程式）。計算模型的詳細步驟如下：

- 首先根據式 (3.3) 定義猜測模型 $T_{s,\theta,t_x,t_y}^{12,guess}$ ：

$$T_{s,\theta,t_x,t_y}^{12,guess} = \begin{bmatrix} s \cos \theta & -s \sin \theta & t_x \\ s \sin \theta & s \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4.14)$$

接著計算猜測模型的四個參數 s 、 θ 、 t_x 、 t_y 。假設隨機選取 2 組特徵點對，點對與對應的二維座標為

$$\begin{aligned} \{p_a = (x_{pa}, y_{pa}), q_a = (x_{qa}, y_{qa})\} \\ \{p_b = (x_{pb}, y_{pb}), q_b = (x_{qb}, y_{qb})\} \end{aligned} \quad (4.15)$$

則縮放參數 s 為

$$s = \frac{|\sqrt{(x_{qb} - x_{qa})^2 + (y_{qb} - y_{qa})^2}|}{|\sqrt{(x_{pb} - x_{pa})^2 + (y_{pb} - y_{pa})^2}|} \quad (4.16)$$

角度參數 θ 為

$$\theta = \arctan\left(\frac{y_{qb} - y_{qa}}{x_{qb} - x_{qa}}\right) - \arctan\left(\frac{y_{qb} - y_{pa}}{x_{pb} - x_{pa}}\right) \quad (4.17)$$

兩個位移參數 t_x 與 t_y 為

$$\begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} x_{qa} \\ y_{qa} \end{bmatrix} - s \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_{pa} \\ y_{pa} \end{bmatrix} \quad (4.18)$$

- 根據猜測模型，測試所有特徵點對是否符合該模型，令第 i 個特徵點對及其二維座標為：

$$\{p_i = (x_{pi}, y_{pi}), q_i = (x_{qi}, y_{qi})\}_{i=1,2,\dots,N} \quad (4.19)$$



根據式 (4.16~4.18)，可將每一點對進行變換：

$$v_{qi} = \begin{bmatrix} x_{qi} \\ y_{qi} \end{bmatrix}$$

$$v'_{pi} = s \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_{pi} \\ y_{pi} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (4.20)$$

變換後，若符合

$$|v_{qi} - v'_{pi}| < \epsilon \quad (4.21)$$

則視該點對符合猜測模型，其中 ϵ 為一調整閥值。

記錄符合模型的點對數量，若數量超過先前記錄的最高數量，則更新目前模型為最佳模型。

3. 重複 k 次步驟 1、2，輸出最佳模型 $T_{s,\theta,t_x,t_y}^{12,best}$

4.4 評判相對變換關係

根據前一節使用 RANSAC 演算法計算出的最佳模型 $T_{s,\theta,t_x,t_y}^{12,best}$ ，我們可將地圖 M_1 進行變換：

$$M'_1 = T_{s,\theta,t_x,t_y}^{12,best} M_1 \quad (4.22)$$

此時地圖 M_1 的座標系、比例尺已經以地圖 M_2 作為參考進行變換。變換後的地圖 M'_1 應與 M_2 具有部份重疊格點，我們另外定義此重疊部份的格點集合為 M_3 ，作為重疊地圖。在重疊地圖 M_3 中，每一格點皆具有來自 M_1 與 M_2 的格點像素值，即

$$M_3(x, y) = \{m'_{1,xy}, m_{2,xy}\} \quad (4.23)$$

依據兩個像素值分別所代表的區域，我們可以定義重疊地圖 M_3 的格點為正確匹配 (agr) 或錯誤匹配 (dis)：

$$M_3(x, y) = \begin{cases} agr & , \text{if } m'_{1,xy} = m_{2,xy} = 0. \\ agr & , \text{if } m'_{1,xy} = m_{2,xy} = 255. \\ dis & , \text{if } m'_{1,xy} = 0, m_{2,xy} = 255. \\ dis & , \text{if } m'_{1,xy} = 255, m_{2,xy} = 0. \end{cases} \quad (4.24)$$



定義完地圖 M_3 的格點後，我們可根據式 (2.5) 計算目前重疊情況的指標分數，令 $agr(M_1, M_2)$ 為地圖 M_3 中 agr 格點的總數， $dis(M_1, M_2)$ 為地圖 M_3 中 dis 格點的總數，則指標分數為

$$\omega(M_1, M_2) = \begin{cases} 0 & , \text{if } agr(M_1, M_2) = 0. \\ \frac{agr(M_1, M_2)}{agr(M_1, M_2) + dis(M_1, M_2)} & , \text{if } agr(M_1, M_2) \neq 0. \end{cases} \quad (2.5)$$

若此指標分數 $\omega(M_1, M_2)$ 大於設定的閾值 (式 (3.4))

$$\omega(M_1, M_2) > \omega_{thres} \quad (3.4)$$

則接受最佳模型 $T_{s,\theta,t_x,t_y}^{12,best}$ ，並以此進行下一部份的地圖銜接處理。

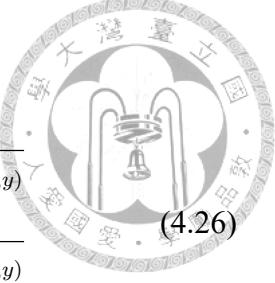
4.5 地圖銜接處理

從前一小節評判相對變換關係的處理中，我們已定義出重疊地圖 M_3 ，此地圖不只是用於判斷是否接受相對變換關係，同時也是兩地圖 M_1 、 M_2 需進行銜接處理的部份。銜接處理的方法在先前已提及，我們採用熵濾波處理 (Entropy Filter) [48]，可將格點佔據地圖的機率資訊列入考量，計算對應熵值，再篩選出該重疊格點的最終區域種類。

在附錄A中，我們詳細說明了 SLAM 技術建構格點佔據地圖時的機率處理，每個格點皆具有一個佔據機率值，若機率大於佔據閾值 (Occupied Threshold, $p_{occupied}$ ，本研究設定為 0.65)，則該格點於影像上紀錄為佔據區域；若機率小於自由閾值 (Free Threshold, p_{free} ，本研究設定為 0.19)，則該格點紀錄為自由區域；若皆不滿足上述兩種情況，則紀錄為未知區域。換句話說，佔據區域的機率較高 (接近 1)，自由區域的機率較低 (接近 0)，未知區域的機率為中間值 (接近 0.5)。地圖銜接時，熵濾波處理考量的是機率資訊而非像素值代表的區域種類，因此必須改為使用佔據機率值。由於重疊地圖 M_3 的格點皆可取得來自地圖 M_1 、 M_2 的 2 個佔據機率，因此我們可以先依照附錄A的方法計算出合併機率，再將原先 2 個機率與合併機率 (共 3 個機率) 帶入熵濾波進行後續處理，相關處理如下：

1. 假設重疊地圖 M_3 中，每個格點對應的兩個佔據機率值為

$$p(s=1)_{M_3(x,y)} = \{p(s=1)_{M'_1(x,y)}, p(s=1)_{M_2(x,y)}\} \quad (4.25)$$



2. 將佔據機率轉換為機率比值，並以對數形式表示

$$\begin{aligned} Odd(s = 1)_{M'_1(x,y)} &= \frac{p(s = 1)_{M'_1(x,y)}}{p(s = 0)_{M'_1(x,y)}} = \frac{p(s = 1)_{M'_1(x,y)}}{1 - p(s = 1)_{M'_1(x,y)}} \\ Odd(s = 1)_{M_2(x,y)} &= \frac{p(s = 1)_{M_2(x,y)}}{p(s = 0)_{M_2(x,y)}} = \frac{p(s = 1)_{M_2(x,y)}}{1 - p(s = 1)_{M_2(x,y)}} \\ \log Odd(s = 1)_{M_3(x,y)} &= \{\log Odd(s = 1)_{M'_1(x,y)}, \log Odd(s = 1)_{M_2(x,y)}\} \end{aligned} \quad (4.26)$$

3. 轉換為對數機率比值 $\log Odd(s = 1)$ 後，可直接以相加減合併對數機率比值，因此合併後的對數機率比值為

$$\log Odd(s = 1)_{M_{combined}(x,y)} = \log Odd(s = 1)_{M'_1(x,y)} + \log Odd(s = 1)_{M_2(x,y)} \quad (4.27)$$

4. 得到式 (4.27) 的結果後，轉換回機率形式，可得整合後的佔據機率為

$$p(s = 1)_{M_{combined}(x,y)} = \frac{Odd(s = 1)_{M_{combined}(x,y)}}{1 + Odd(s = 1)_{M_{combined}(x,y)}} \quad (4.28)$$

依照式 (4.25~4.28) 的處理，對於重疊地圖的每一格點，我們共可取得 3 個機率值，即

$$p(s = 1)_{M_3(x,y)} = \{p(s = 1)_{M'_1(x,y)}, p(s = 1)_{M_2(x,y)}, p(s = 1)_{M_{combined}(x,y)}\} \quad (4.29)$$

接著，我們可將此 3 個機率帶入熵濾波進行篩選。

篩選之前，我們先簡單說明熵濾波的基本概念。熵濾波處理中，熵值與機率資訊的關係定義為

$$H(X) = - \sum_{i=1}^n p(X = i) \log p(X = i) \quad (4.30)$$

其中 X 為事件隨機變數， $p(X = i)$ 為某一情況 i 下的機率， $H(X)$ 則表示事件對應的熵值。根據 Fox [48] 說明，熵值越高，表示事件不確定性越大。若將格點佔據地圖的機率訊息套用至式 (4.30)，可令事件隨機變數為格點佔據的可能性 ($X = s$)，其中包含兩種情況，即格點為佔據的可能性 ($s = 1$) 與非佔據的可能性 ($s = 0$)，並計算對應熵值為

$$\begin{aligned} H(s) &= -p(s = 1) \log p(s = 1) - p(s = 0) \log p(s = 0) \\ &= -p(s = 1) \log p(s = 1) - (1 - p(s = 1))(\log 1 - p(s = 1)) \end{aligned} \quad (4.31)$$

在式 (4.31) 中，若以佔據機率 $p(s = 1)$ 為橫座標、熵值 $H(s)$ 為縱座標，可得下

圖：

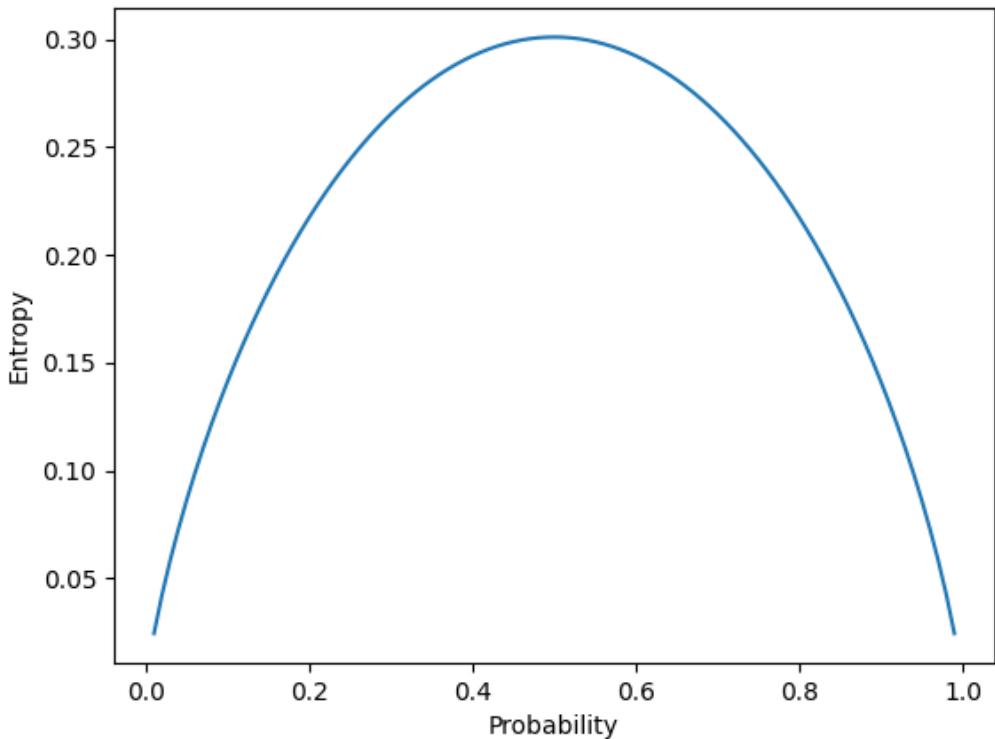


圖 4.4: 機率與熵值對應關係。

在圖4.4中，可看出機率值偏向0與1時，對應的熵值較低，若以 Fox 的說明解釋，其不確定性應較低，而機率介於中間值（0.5）時，熵值較高，不確定性應較高。另外，在格點佔據地圖中，屬於不確定性較低的自由區域與佔據區域，其原先的佔據機率亦分別接近0與1，而不確定性較高的未知區域，原先的佔據機率則接近中間值0.5。因此，機率在熵濾波的處理中與格點佔據地圖的概念相符合，所以我們可使用熵濾波進行佔據機率的篩選。

接著，回到原先佔據機率的篩選，在式（4.29）中，我們已對重疊地圖 M_3 的



每一個格點計算出 3 個佔據機率值，分別使用式 (4.30) 的概念計算對應熵值：

$$\begin{aligned}
 H(s)_{M'_1(x,y)} &= -p(s=1)_{M'_1(x,y)} \log p(s=1)_{M'_1(x,y)} \\
 &\quad - p(s=0)_{M'_1(x,y)} \log p(s=0)_{M'_1(x,y)} \\
 &= -p(s=1)_{M'_1(x,y)} \log p(s=1)_{M'_1(x,y)} \\
 &\quad - (1-p(s=1)_{M'_1(x,y)}) \log (1-p(s=1)_{M'_1(x,y)}) \\
 H(s)_{M_2(x,y)} &= -p(s=1)_{M_2(x,y)} \log p(s=1)_{M_2(x,y)} \\
 &\quad - p(s=0)_{M_2(x,y)} \log p(s=0)_{M_2(x,y)} \\
 &= -p(s=1)_{M_2(x,y)} \log p(s=1)_{M_2(x,y)} \\
 &\quad - (1-p(s=1)_{M_2(x,y)}) \log (1-p(s=1)_{M_2(x,y)}) \\
 H(s)_{M_{combined}(x,y)} &= -p(s=1)_{M_{combined}(x,y)} \log p(s=1)_{M_{combined}(x,y)} \\
 &\quad - p(s=0)_{M_{combined}(x,y)} \log p(s=0)_{M_{combined}(x,y)} \\
 &= -p(s=1)_{M_{combined}(x,y)} \log p(s=1)_{M_{combined}(x,y)} \\
 &\quad - (1-p(s=1)_{M_{combined}(x,y)}) \log (1-p(s=1)_{M_{combined}(x,y)}) \\
 \end{aligned} \tag{4.32}$$

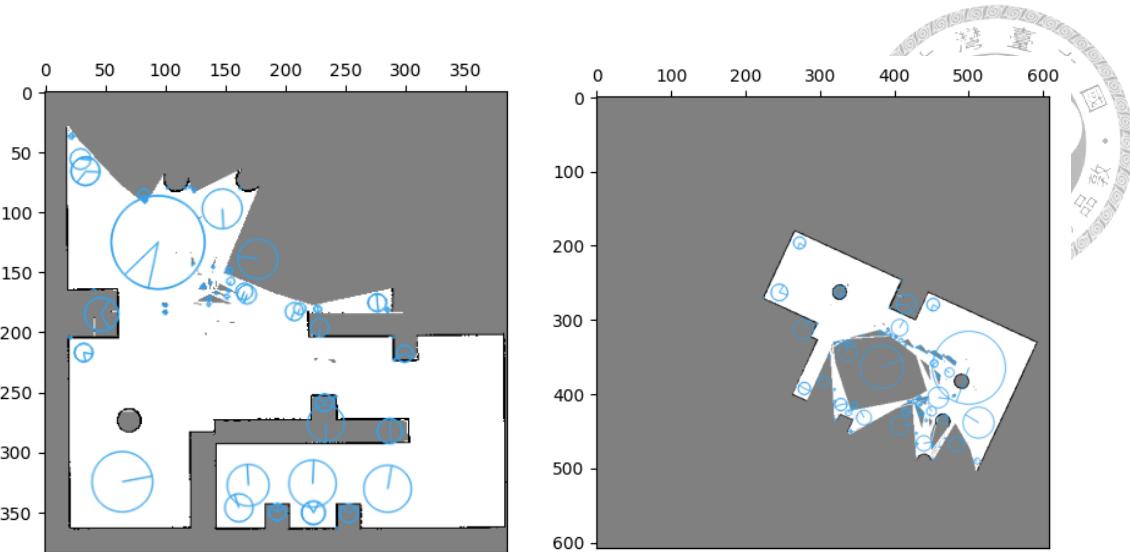
熵值計算完成後，我們選擇對應熵值較小的機率作為重疊格點的佔據機率，並將此機率根據佔據、自由閥值輸出重疊格點 $M_3(x,y)$ 的區域種類（式 (A.8)），進而完成地圖銜接處理。

4.6 各演算法效果討論與合併測試

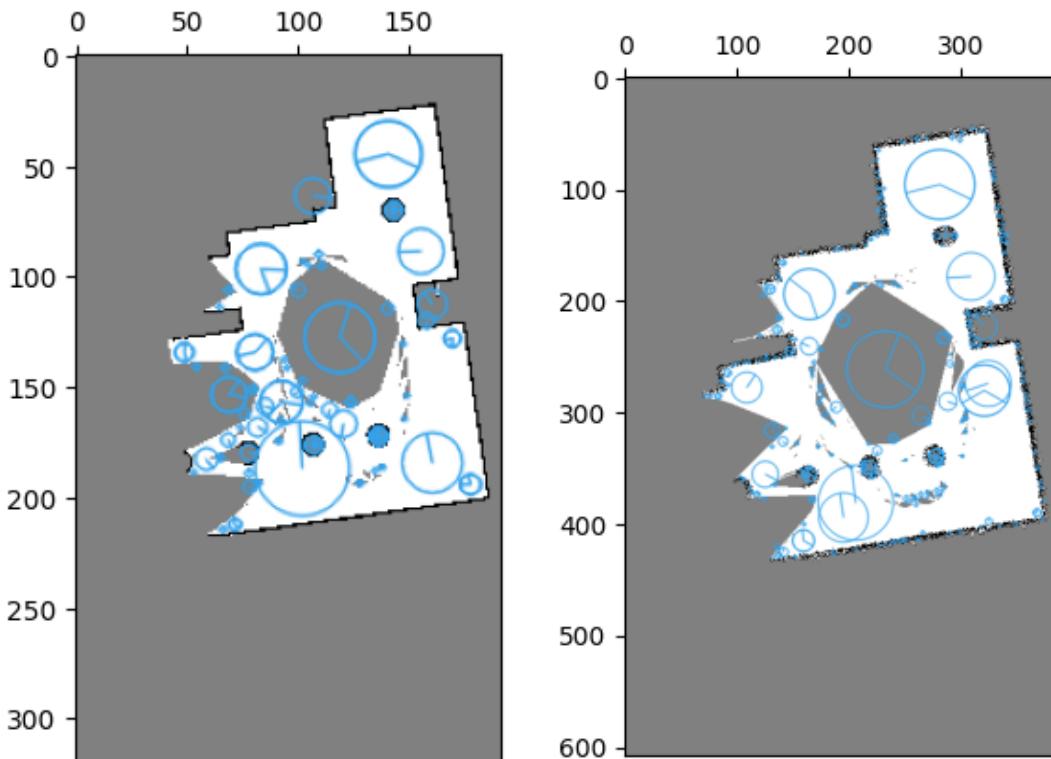
上述已介紹完參考方法（圖3.2）中 5 個流程步驟所使用到的演算法，接著我們將以圖3.4的測試案例（主要使用局部測試圖二，即圖3.4(c)，因其線段明顯不平行座標軸，故在第五章修正步驟中可進行比較）說明各演算法實際操作上的效用，最後再以表3.2的組合進行參考方法的合併測試。

4.6.1 演算法效用討論

首先，我們針對測試案例使用 SIFT 演算法進行特徵的擷取，其結果如圖4.5。從圖中可看出，每一張地圖影像皆擷取出數個以藍色圓圈表示的特徵點，圓圈中的線段表示該特徵主方向。



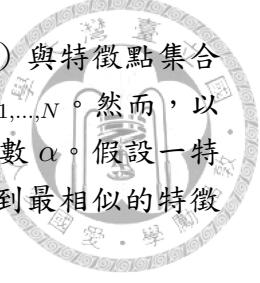
(a) 局部測試地圖一（圖3.4(b)) 於 SIFT 演算法 (b) 局部測試地圖二（圖3.4(c)) 於 SIFT 演算法之效用，藍色圓圈表示 SIFT 特徵位置與特徵主方向。藍色圓圈表示 SIFT 特徵位置與特徵主方向。



(c) 局部測試地圖三（圖3.4(d)) 於 SIFT 演算法 (d) 局部測試地圖四（圖3.4(e)) 於 SIFT 演算法之效用，藍色圓圈表示 SIFT 特徵位置與特徵主方向。藍色圓圈表示 SIFT 特徵位置與特徵主方向。

圖 4.5: SIFT 演算法應用於局部測試地圖之特徵擷取效用呈現。

根據圖4.5的結果，我們可接著使用地圖的特徵點描述符向量進行匹配，建立特徵點之間的關係。匹配過程已在第4.2節說明，以歐式距離（式(4.11)）計算向



量相似度，查找特徵點集合 $P = \{p_1, \dots, p_n\}$ (來自地圖影像 M_1) 與特徵點集合 $Q = \{q_1, \dots, q_m\}$ (來自地圖影像 M_2) 的最相似配對，即 $\{p_i, q_i\}_{i=1, \dots, N}$ 。然而，以歐式距離的相似度篩選方式上，Lowe [3] 認為必須加入一篩選參數 α 。假設一特徵點 p_i (來自集合 P)，根據歐式距離的計算，可於集合 Q 中找到最相似的特徵 q_j 與第二相似的特徵 q_k 。令 p_i 與 q_j 、 q_k 分別的歐式距離為

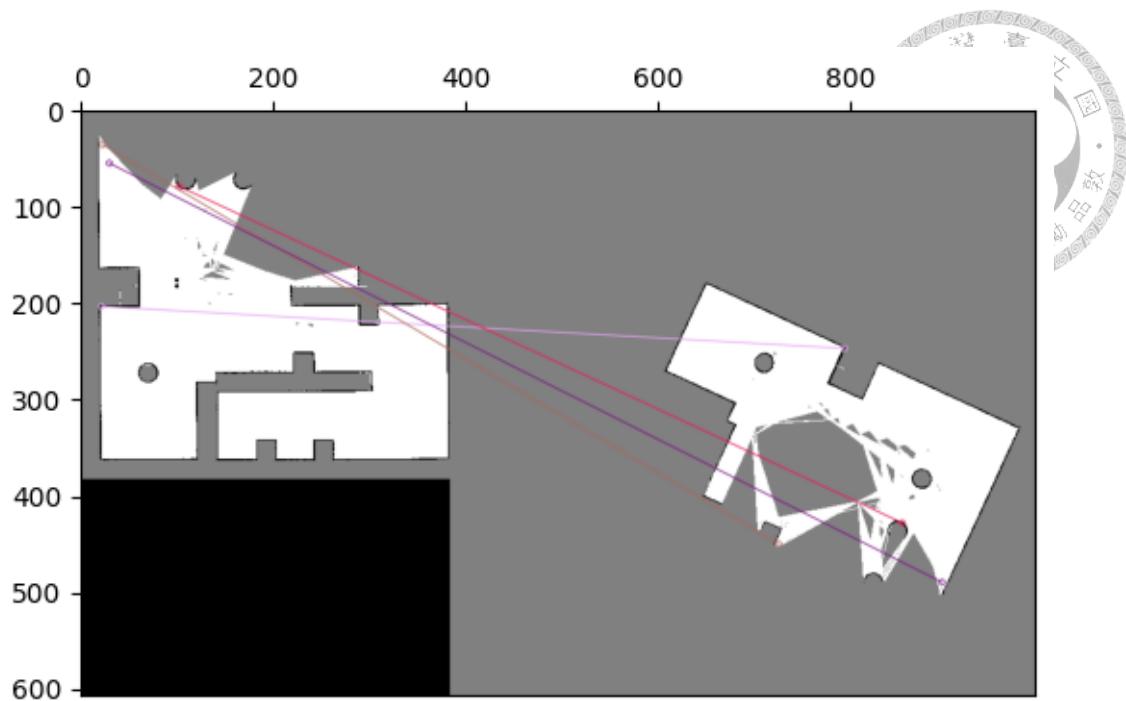
$$d(p_i, q_j) = \sqrt{(p_{i1} - q_{j1})^2 + \dots + (p_{i128} - q_{j128})^2} \quad (4.33)$$

$$d(p_i, q_k) = \sqrt{(p_{i1} - q_{k1})^2 + \dots + (p_{i128} - q_{k128})^2} \quad (4.34)$$

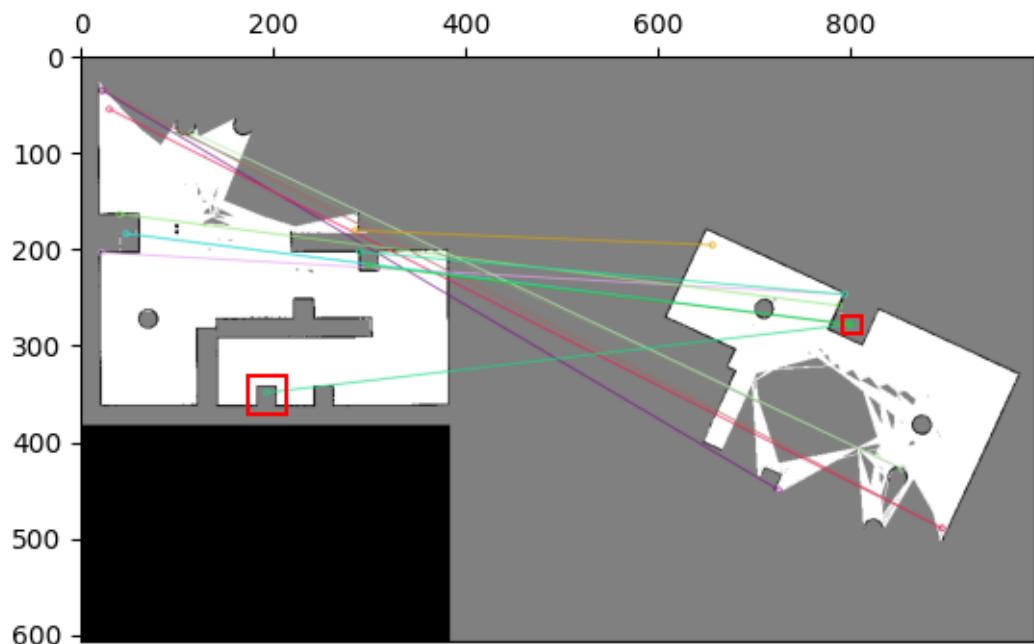
Lowe 認為，若 p_i 與最相似特徵 q_j 可被選作特徵配對，其必須滿足

$$d(p_i, q_j) \leq \alpha \times d(p_i, q_k) \quad (4.35)$$

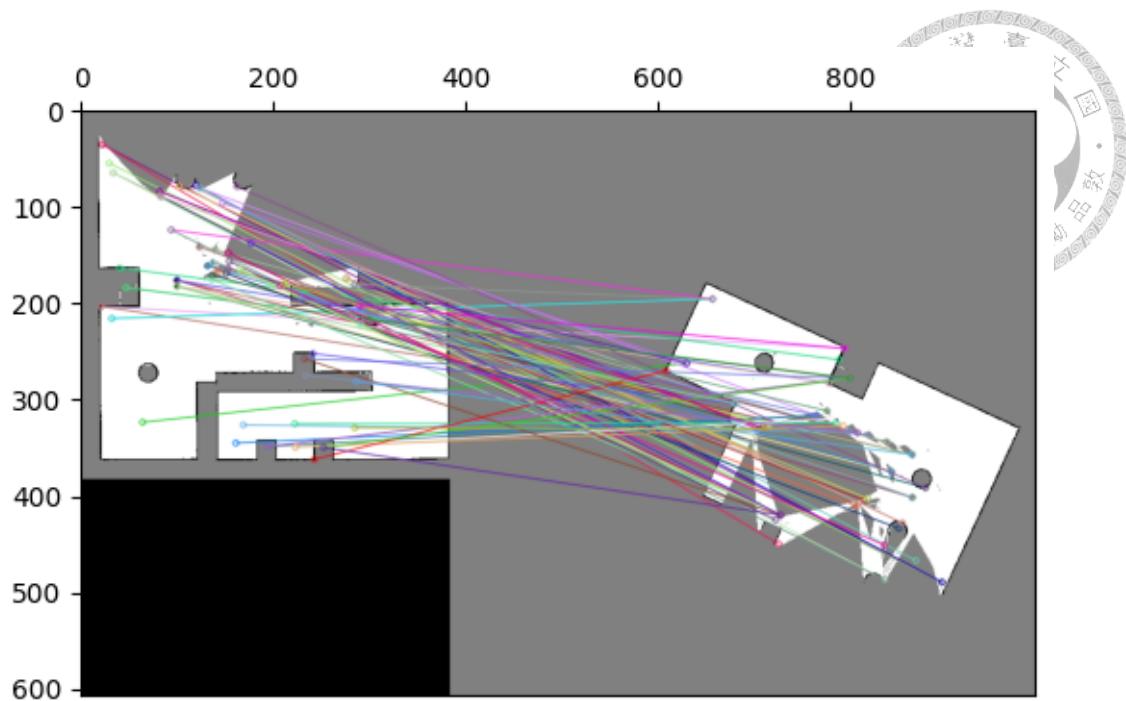
Lowe 設式 (4.35) 的篩選參數 α 為 0.8。因為 SIFT 演算法計算時可能受到影像遮擋、背景混亂產生部份並無匹配關係的特徵點，若不使用篩選參數可能產生數對不必要的錯誤特徵配對；若使用則可進行排除，同時建立的特徵配對也符合唯一性，操作上亦較為穩定。此參數 α 設定越小，排除效果越好，但同時特徵配對數量也會越少。根據上述說明，我們針對圖4.5(a)~4.5(b)的特徵點，以不同的篩選參數進行配對，如圖4.6。從圖中亦證明，參數設定越小，建立的特徵配對數越少。由於後續的 RANSAC 演算法必須具有一定數量的正確匹配關係，因此我們的參數先以設定為 $\alpha = 0.75$ 為主，錯誤特徵配對則作為離群值 (Outlier)，並仰賴 RANSAC 演算法進行排除。



(a) 局部測試地圖一與二之特徵匹配示意圖，匹配篩選參數 $\alpha = 0.5$ 之情況。



(b) 局部測試地圖一與二之特徵匹配示意圖，匹配篩選參數 $\alpha = 0.75$ 之情況，紅框表示錯誤配對的特徵匹配。



(c) 局部測試地圖一與二之特徵匹配示意圖，不使用匹配篩選參數之情況。

圖 4.6: SIFT 特徵匹配示意圖。

將已建立的特徵配對輸入 RANSAC 演算法中，我們可得到地圖合併的相對變換關係。根據該變換關係，可將地圖利用熵濾波進行合併處理，如圖4.7。從圖中可見，根據相對變換關係與熵濾波技術，兩地圖影像已進行合併。

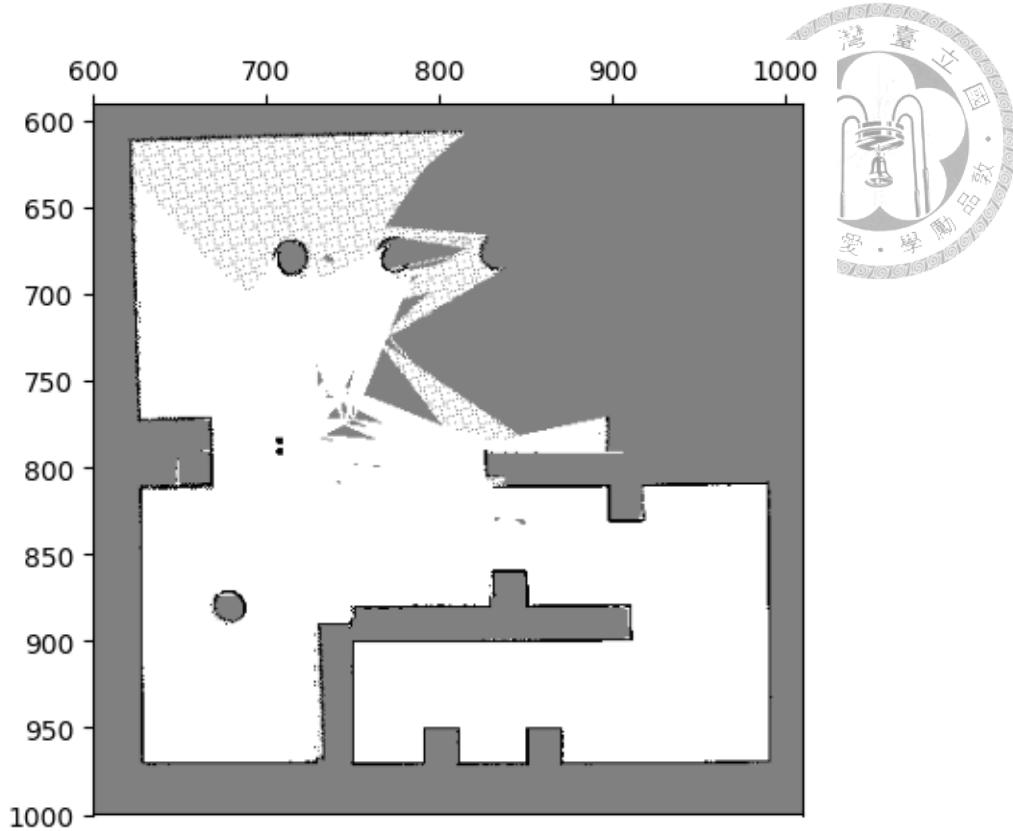


圖 4.7: 熵濾波處理應用於局部測試地圖一與二合併之示意圖。

介紹完實際應用每個步驟流程的效用圖後，我們將在下一小節中討論表3.2案例的合併結果，並根據測試結果得出用於判斷是否接受合併的接受指標閾值 ω_{thres} 。

4.6.2 合併測試

本小節根據已介紹的參考方法（圖3.2）流程，我們將實測於表3.2的每個案例組合。然而，在討論測試結果前，我們必須先說明表3.2的參數關係。該表中每個案例組皆列出相對參數關係，例如旋轉、平移、縮放差異，故使用參考方法時可對參數的使用進行調整。舉例來說，每個組合皆具有旋轉、平移差異，因此必須考量式 (4.17~4.18) 的三個參數 (θ 、 t_x 、 t_y)。而僅有組合二額外考量縮放差異，因此縮放參數 s 也必須在 RANSAC 演算法中使用式 (4.16) 計算。其他未考量縮放參數的組合案例，參數 s 則可直接人為設定。縮放差異可選擇是否考量的原因為，機器人建構地圖時可精確設定解析度參數，因此可從此資訊得知正確的地圖縮放關係；而旋轉、平移參數皆倚靠量測得知，故無法直接得知正確的相對關係。在參數的使用上，考量的參數越多，欲計算出正確的相對關係就越困難。

說明完組合案例的參數使用後，我們接著討論組合案例的測試合併結果。測試結果我們不使用熵濾波的輸出圖作為呈現，而是另外使用其他色彩來表示格點



合併的正確匹配 (agr , 綠色)、錯誤匹配 (dis , 紅色) 關係。以組合一的結果為例，如圖4.8，圖中可見大部分的重疊區域為綠色，少部份線段顯示為紅色，表示大部分格點為正確匹配，僅有少量錯誤匹配。若以人眼主觀判斷，可看出該合併結果與圖3.4(a)的模擬環境相符，故先以主觀判斷為成功合併，而此情況下的接受指標為（下標1表示組合一）

$$\omega_1 = \frac{agr_1}{agr_1 + dis_1} = 0.966292 \quad (4.36)$$

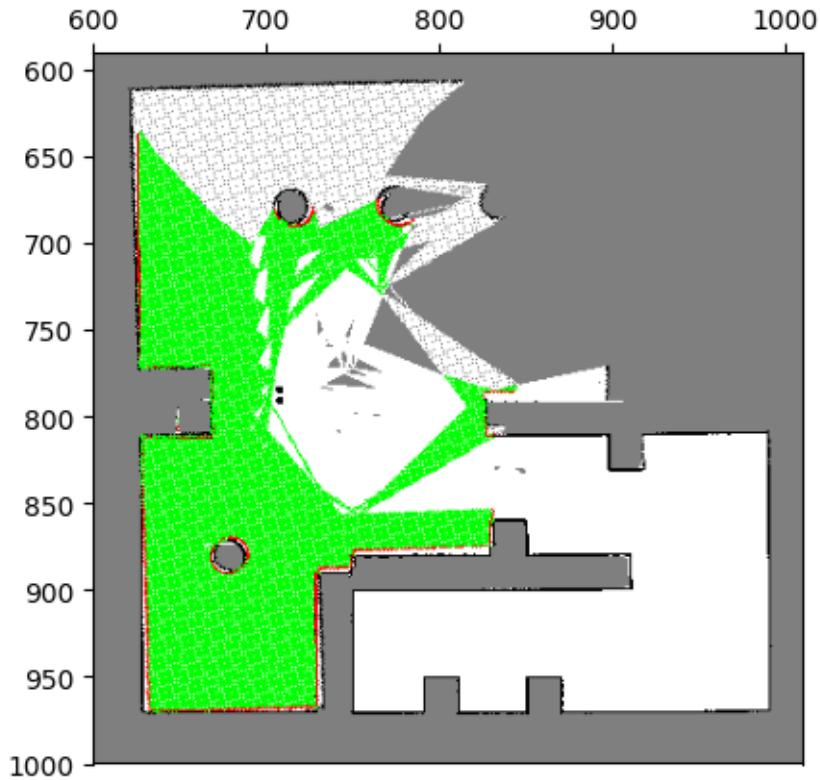


圖 4.8: 組合一之合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。

另外，不同於組合一成功合併的情況，組合二與組合三的結果分別如圖4.9與圖4.10。從兩圖中，可明確的看出合併結果與模擬環境（圖3.4(a)）完全不同，同時代表錯誤配對的紅色格點比例也相對較多，故兩組合測試結果屬於失敗。此時兩結果的接受指標分別為

$$\begin{aligned} \omega_2 &= \frac{agr_2}{agr_2 + dis_2} = 0.935030 \\ \omega_3 &= \frac{agr_3}{agr_3 + dis_3} = 0.903026 \end{aligned} \quad (4.37)$$

根據此三圖（圖4.8~4.10）的接受指標（式（4.36~4.37）），我們認為判斷合併屬於成功或失敗的閥值 (ω_{thres}) 應設定於 0.93 至 0.96 之間，故本研究暫且設定 $\omega_{thres} = 0.95$ 。

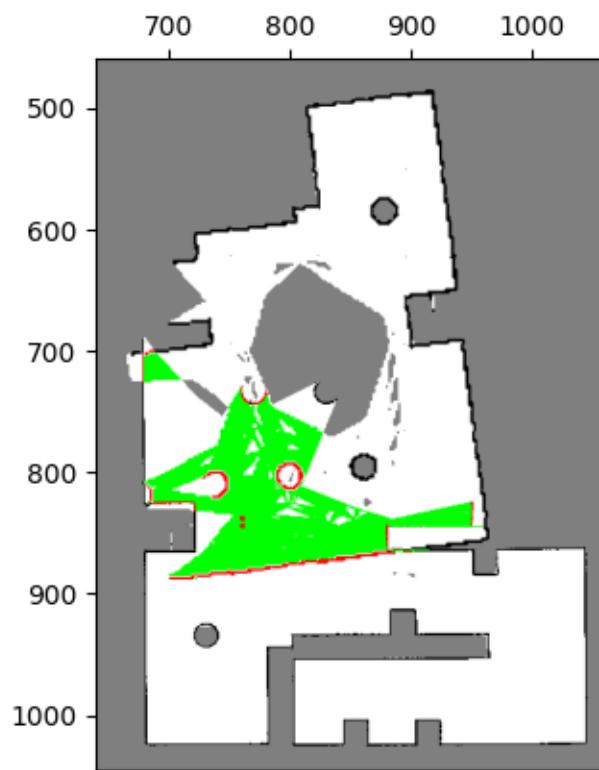


圖 4.9: 組合二之合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境不相符，代表計算的相對變換關係錯誤。

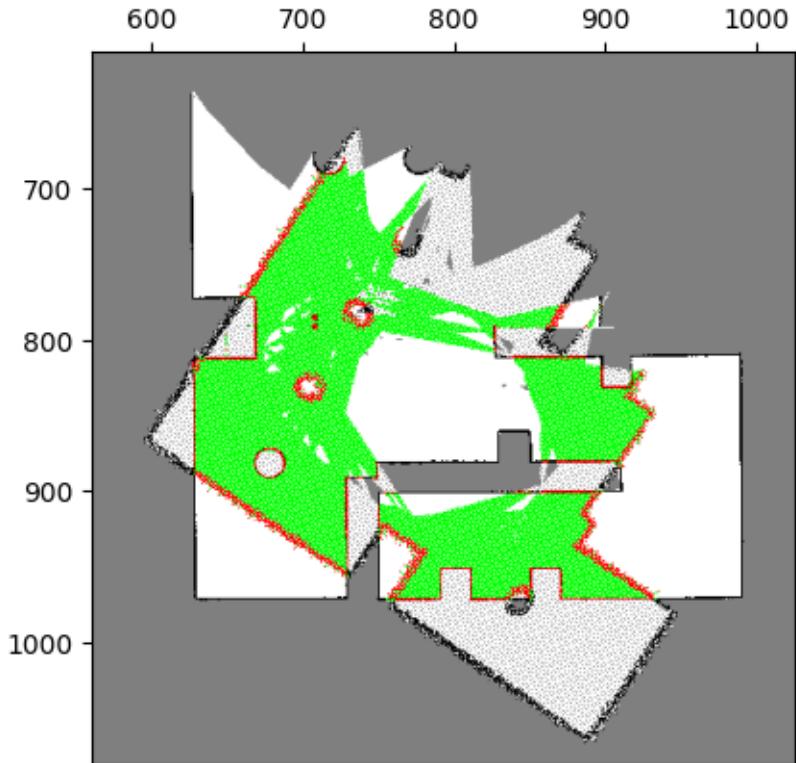


圖 4.10: 組合三之合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境不相符，代表計算的相對變換關係錯誤。

根據參考方法的測試，我們認為測試失敗主要來自於以下幾個原因：

1. SIFT 演算法雖可擷取穩定、不受尺度與旋轉影響的特徵點，但格點佔據地圖的未知區域代表資訊目前尚不清楚，若影像特徵擷取時考量該區域種類的像素值，可能影響特徵點的擷取，如圖4.5中部份特徵點出現在未知區域（灰色區域）上。
2. 格點佔據地圖的像素值差異過少（僅 3 種數值），因此計算描述符向量並匹配時可能產生錯誤配對，如圖4.6(b)，部份配對連線的特徵點明顯來自不同區域（紅框）。
3. 建構格點佔據地圖時，可能因感測器誤差影響使得佔據區域扭曲，例如圖3.4中，圖3.4(e)的佔據區域線段較其他三圖粗，同時對於圓、線段的表達也較為模糊。導致在擷取特徵時，更容易產生許多不具匹配關係的特徵點，例如圖4.5中，圖4.5(d)中出現許多位在佔據區域線段的特徵點，與其他三圖不同。若以此進行匹配，則可能產生錯誤的特徵配對，使得後續計算相對關係錯誤。

根據上述幾個原因，我們認為使用特徵擷取方法來解決地圖合併問題，必須具備良好的特徵匹配。由於感測誤差、格點不確定性皆可能對特徵點造成影響，故我們希望對地圖影像進行額外處理，將地圖的不確定性排除，同時也將感測誤差造成的扭曲進行更正。因此我們將在下一個章節介紹本研究提出的地圖修正前處理方法，我們將對格點佔據地圖進行一系列影像處理，使其修正佔據區域的表達，再重新經過參考方法的流程，進而得到較好的合併效果。



第五章

加入地圖修正之合併方法

第四章中我們不僅介紹了參考方法，同時也分析了該方法中可能產生問題的原因，並且提出了如何進行改善。如第四章說明，我們希望將地圖的不確定性（未知區域的訊息）進行排除，並且對感測誤差造成的佔據區域扭曲進行修正。為達成此兩項目標，本研究提出了用於地圖修正的影像處理方法，如圖3.3，在進行特徵擷取、匹配與計算相對關係前，先將地圖影像進行處理（圖3.3藍色方框之步驟），待處理完成後，影像便剔除不確定性因素與區域扭曲現象。再接著經過原先參考方法的流程，則可有效提昇合併的效果。因此，本章節將接著介紹上述提及的地圖修正之影像處理方法。該方法中主要分為5大項目，分別為

1. 影像二值化處理
2. 雷登變換與影像旋轉
3. 閉運算處理
4. 興趣點採樣
5. 主體結構修正

以下小節將就每個項目先進行說明，並且應用於局部測試地圖二（圖3.4(c)）來展示其效用，最後則將以表3.2之案例重新進行合併測試。

5.1 影像二值化處理

本研究所使用的格點佔據地圖中，像素值的三個整數分別代表佔據（0）、自由（255）、未知（128）區域，此三個區域分別對應著高低不同的不確定性，故用於影像處理時，不確定性高的像素格點可能對特徵擷取造成影響。為避免上述影響，並且同時兼顧地圖的環境結構，例如牆壁，我們將採用影像的二值化處理，



把地圖影像分為「佔據區域」與「非佔據區域」，前者為原先的佔據區域，後者則包含原先的自由區域與未知區域。影像二值化處理主要設定一閥值 (B_{thres})，並以此將所有像素區分為兩個種類（本研究定為佔據與非佔據區域），即格點像素值小於閥值者，該格點定為佔據區域；大於等於閥值者，格點定為非佔據區域：

$$M(x, y) = \begin{cases} \text{Occupied} & \text{if } M(x, y) < B_{thres} \\ \text{Non-occupied} & \text{else.} \end{cases} \quad (5.1)$$

式 (5.1) 中 $M(x, y)$ 為格點像素值， B_{thres} 為閥值參數。

本研究中，經過式 (5.1) 處理的格點已分為佔據與非佔據區域，如圖5.1。然而，影像處理上仍需要給定數值才能進行運算，因此我們將佔據區域以新的像素值 255 表示，圖5.1中以白色表示，而非佔據區域的新像素值則定為 0，圖5.1以黑色表示，以數學表達則為

$$M(x, y) = \begin{cases} 255 & (\text{Occupied}) \quad \text{if } M(x, y) < B_{thres} \\ 0 & (\text{Non-occupied}) \quad \text{else.} \end{cases} \quad (5.2)$$

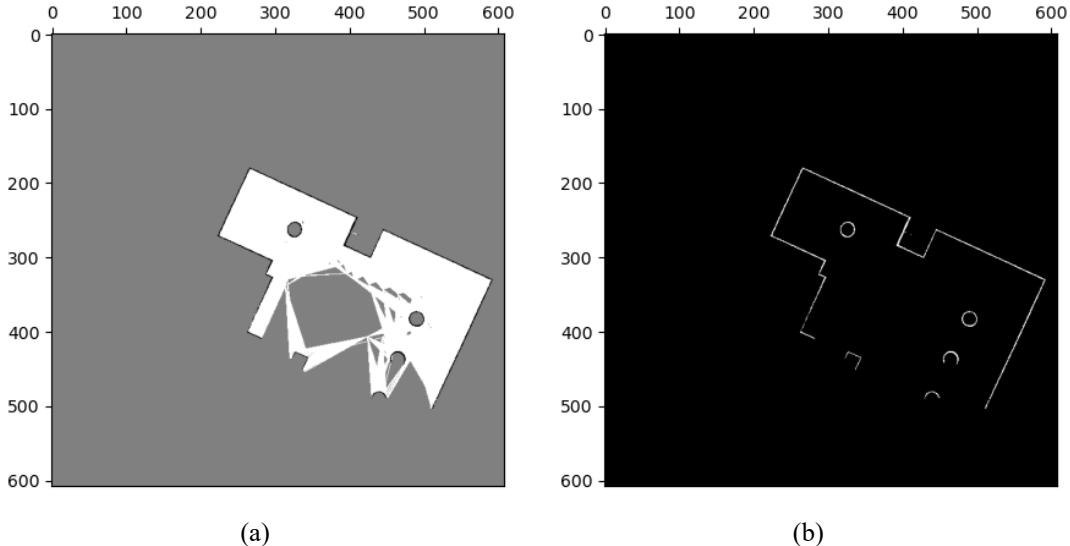


圖 5.1: 格點佔據地圖二值化處理。圖5.1(a)為原格點佔據地圖之影像，具有 3 種像素值（黑色：0、灰色：128、白色：255）來表示不同區域種類。圖5.1(b)則表示該地圖經過二值化處理後，佔據區域（白色：255）與非佔據區域（黑色：0）的分佈。



5.2 雷登轉換與影像旋轉

將地圖影像進行二值化處理後，影像顯示上僅剩下黑、白色的格點。由於格點佔據地圖的環境結構中，經常出現線段類型的佔據區域，因此在格點影像中，若該直線並非與兩個座標軸平行或垂直，格點顯示上則會具有階梯狀的扭曲情況，如圖5.2。另外，前面章節也已說明，格點佔據地圖作為影像處理時，可使用的特徵本就不多，若影像具有額外的階梯狀的扭曲情況，也可能對特徵擷取產生影響。因此，為避免扭曲情況，本研究採用雷登轉換（Radon Transform）對二值化處理後的地圖影像進行變換，並依變換結果將地圖進行旋轉。

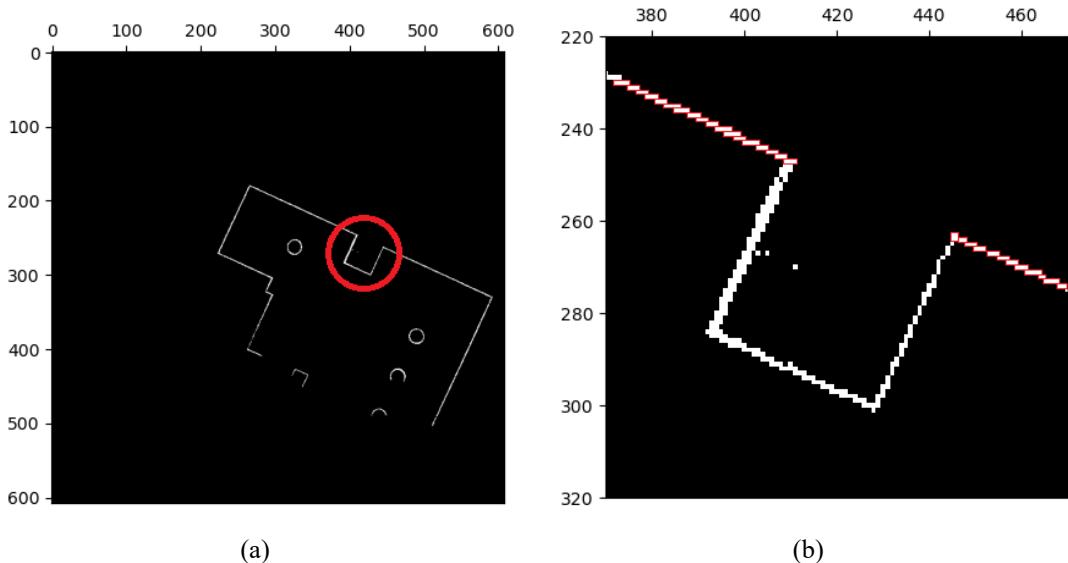


圖 5.2: 格點影像線段的扭曲示意圖。圖5.2(a)表示二值化地圖影像的佔據區域線段；圖5.2(b)為圖5.2(a)紅圈處放大示意圖，紅框處表示階梯狀的扭曲。

雷登轉換（Radon Transfrom）於 1917 年由 Johann Radon 提出，可將二維平面函數 f 轉換為二維空間的線性函數 Rf 。其原理在於將平面函數 f 進行不同直線參數的線積分，將積分結果轉換至線性函數 Rf 。直線參數為 r 與 θ ，分別表示該直線與平面原點的距離和直線與平面座標軸的夾角（本文使用 y 軸）。雷登轉換的詳細數學表達可見於 [52]。

將格點佔據地圖影像經過雷登轉換後，可從線性函數 Rf 中查找出積分值最大的部份，若對應回原地圖影像則是最長線段，如圖5.3所示。利用此特性，可找出目前地圖佔據區域線段與座標軸的夾角，並作為後續影像欲旋轉的角度。

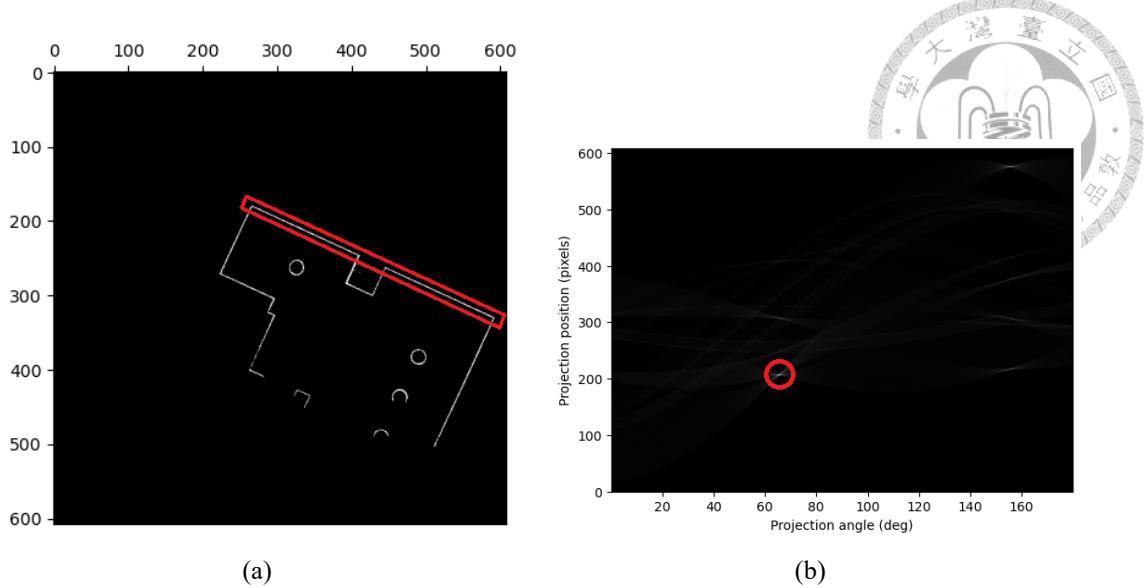


圖 5.3: 影像雷登轉換結果。圖5.3(a)為原平面影像 f ，紅框處表示積分值最大線段。圖5.3(b)為雷登轉換後的線性函數 Rf ，縱座標表示直線參數 r ，橫座標表示直線參數 θ 。紅圈處表示此點具有積分最大值，可對應至輸入影像中最長線段，即左圖紅框線段。

根據雷登變換計算出的角度，接著我們將地圖影像進行旋轉，如式 (5.3)。

$$M' = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} M \quad (5.3)$$

而地圖進行旋轉時，格點像素值的計算必須利用插值法進行計算。舉例而言，假設一地圖 M 旋轉 $\theta = 30^\circ$ 後為地圖 M' 。對於旋轉後的格點 $M'(30, 30)$ 而言，若旋轉 $\theta = -30^\circ$ 則求得旋轉前的格點，即 $M(40.98, 10.98)$ ，如式 (5.4)

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos -30^\circ & -\sin -30^\circ \\ \sin -30^\circ & \cos -30^\circ \end{bmatrix} \begin{bmatrix} 30 \\ 30 \end{bmatrix} = \begin{bmatrix} 40.98 \\ 10.98 \end{bmatrix} \quad (5.4)$$

然而，格點 $M(40.98, 10.98)$ 的座標具有小數點，並不包含於地圖 M 中，因此像素值必須利用 $M(40.98, 10.98)$ 的周圍 4 個點 ($M(40, 10)$ 、 $M(40, 11)$ 、 $M(41, 10)$ 、 $M(41, 11)$) 以插值法計算。說明利用插值法計算影像像素值前，為使說明清楚，我們先定義所需使用到的格點，即

- 旋轉後格點 $M'(x, y)$ 。
- 旋轉前格點 $M(x, y)$ 。

- 旋轉前的四個周圍格點 $M(x^+, y^+)$ 、 $M(x^-, y^+)$ 、 $M(x^-, y^-)$ 、 $M(x^+, y^-)$ 。

利用上述定義格點與插值法可計算 $M(x, y)$ 為

$$\begin{aligned} M(x, y^-) &= \frac{x^+ - x}{x^+ - x^-} M(x^-, y^-) + \frac{x - x^-}{x^+ - x^-} M(x^+, y^-) \\ M(x, y^+) &= \frac{x^+ - x}{x^+ - x^-} M(x^-, y^+) + \frac{x - x^-}{x^+ - x^-} M(x^+, y^+) \\ M(x, y) &= \frac{y^+ - y}{y^+ - y^-} M(x, y^-) + \frac{y - y^-}{y^+ - y^-} M(x, y^+) \end{aligned} \quad (5.5)$$

在插值運算（式（5.5））過程中，格點的關係可見圖5.4。計算出 $M(x, y)$ 的像素值後，同時也可將該值作為旋轉後格點 $M'(x, y)$ 的像素值，即 $M'(x, y) = M(x, y)$ 。

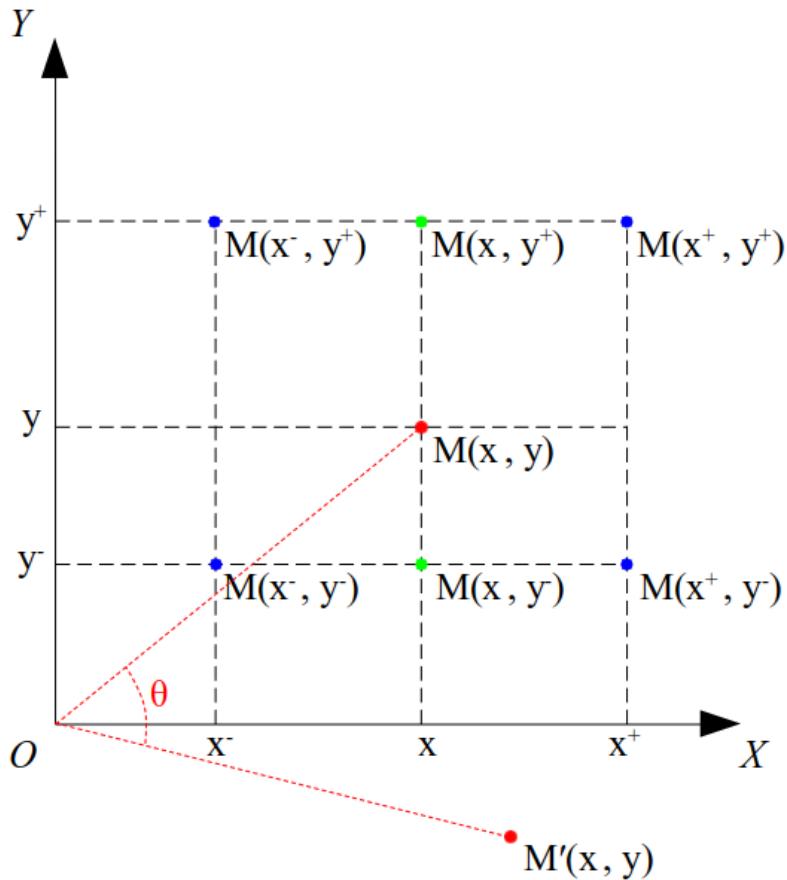


圖 5.4: 像素值插值計算示意圖。

然而，根據式（5.5）的插值運算，此時旋轉後格點 $M'(x, y)$ 的像素值可能並非原先定義的兩個數值（0、255），因此我們將再次通過二值化處理（式（5.2））。關於旋轉時，插值運算與二值化處理對格點的影響可見圖5.5。從圖中可知，上述

操作將使線段產生毛邊或缺口（圖5.5(d)），影響地圖的線段主體。為進行毛邊與缺口的修補，我們將接著以下一節介紹的閉運算進行處理。

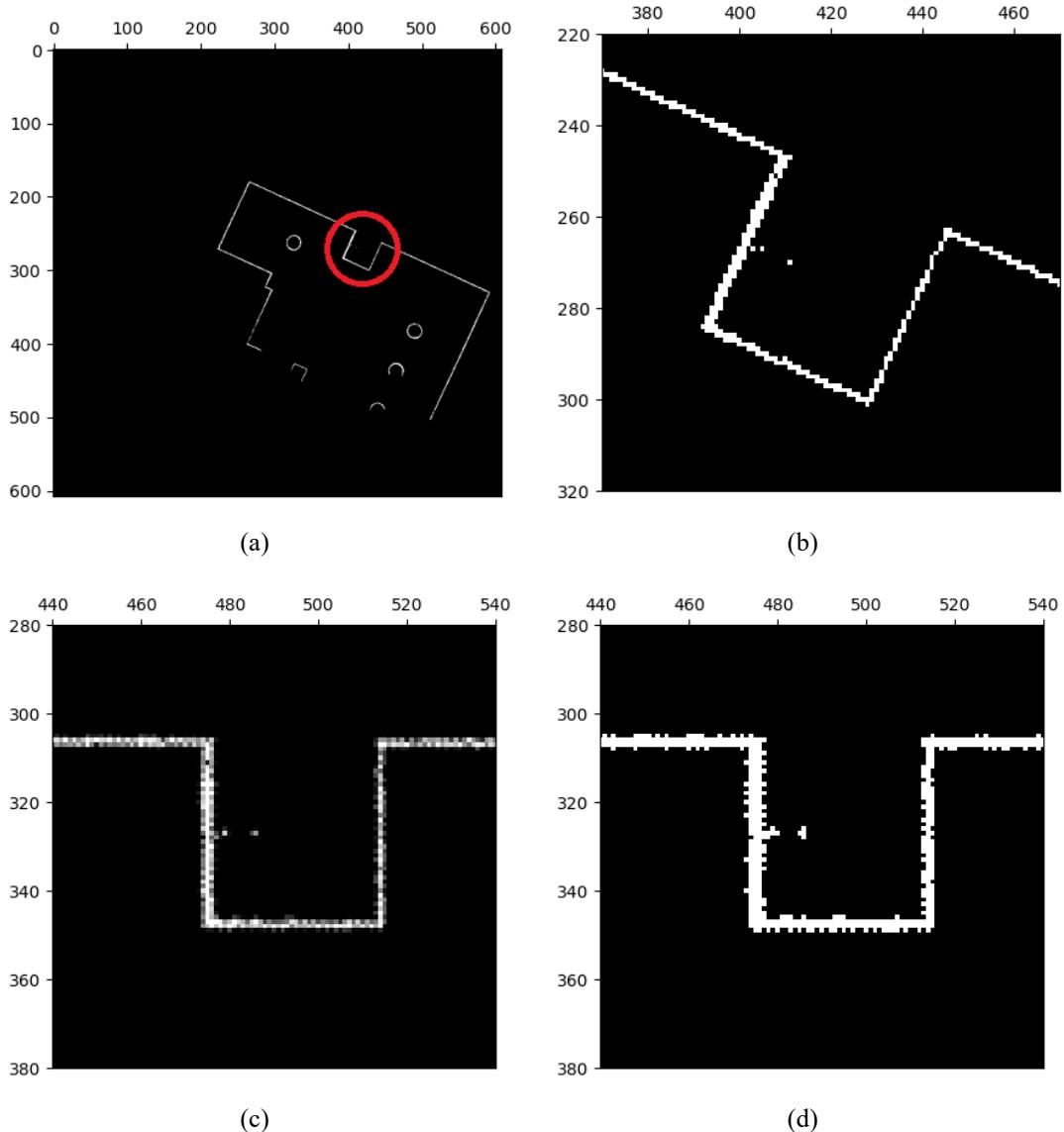
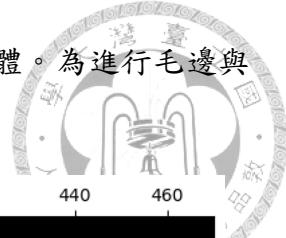


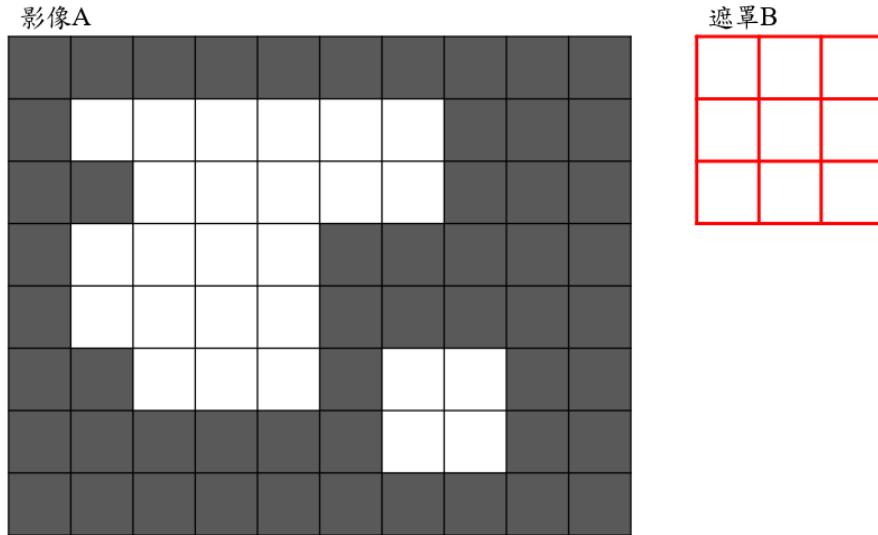
圖 5.5: 影像旋轉之處理示意圖。圖5.5(a)為欲旋轉影像；圖5.5(b)為紅圈處放大影像；圖5.5(c)為紅圈處旋轉後，像素插值運算（式（5.5））的結果，部份格點顏色顯示灰色，表示像素值介於 0~255 之間；圖5.5(d)為再次經過二值化操作的結果，圖中白色線段產生許多毛邊與缺口。

5.3 閉運算處理

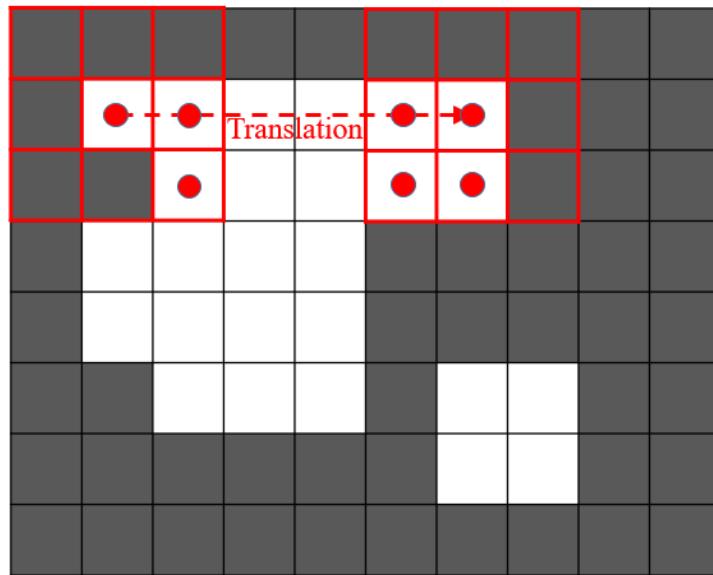
由於前一節將影像旋轉後會產生毛邊或缺口，故我們採用形態學影像處理的閉運算進行修補。形態學處理中，閉運算（Closing）為兩項集合運算的組合，即

膨脹 (Dilation) 與侵蝕 (Erosion)，故以下先介紹膨脹與侵蝕的運算方式，再說明組合為閉運算操作時的效果。

形態學主要利用集合運算對二值化影像進行處理，假設二值化影像 A 為大小 $m \times n$ 的二維矩陣，另外定義一運算遮罩 B 為大小 $(2k + 1) \times (2k + 1)$ 的二維矩陣，且 $2k + 1 \leq m, 2k + 1 \leq n$ 。由於遮罩 B 的大小較小，因此可將其在影像 A 中進行平移，如圖5.6所示。



(a) 影像 A 與遮罩 B 之示意圖。影像 A 具有白色格點（像素值 255）與黑色格點（像素值 0），遮罩 B 僅具有白色格點（像素值 255）。



(b) 將遮罩 B 移動於影像 A 中，紅點表示每次移動情況下的交集情形。

圖 5.6: 影像與遮罩定義及平移示意圖。

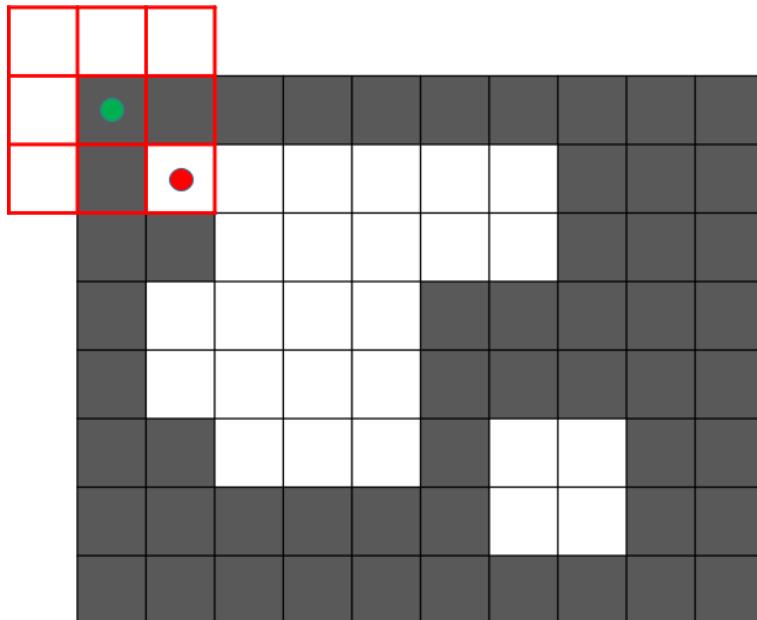


當遮罩 B 在影像 A 的各格點平移時，兩者可計算格點的集合關係，例如交集、聯集等。而形態學則是利用此類集合關係對影像進行修改，以前面提到的膨脹與侵蝕為例：

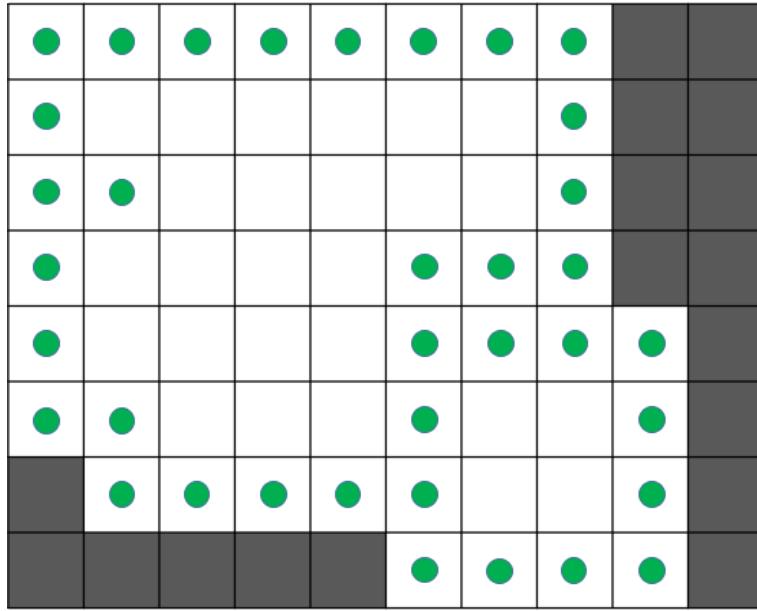
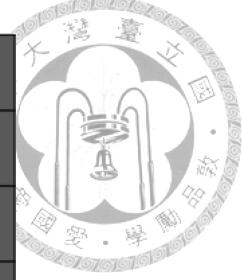
1. 膨脹 (Dilation)：將遮罩以中心點 $B(i, j)$ 為基準點，依序移動於影像 A 的各個格點 $A(x, y)$ 。若 $A(x, y)$ 鄰近的 $(2k + 1) \times (2k + 1)$ 個格點與遮罩 B 的格點交集為空集合，即無任何格點像素值相同，則令格點 $A(x, y)$ 為 0；若交集並非空集合，則令格點 $A(x, y)$ 為 255。數學定義如下

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\} \quad (5.6)$$

式 (5.6) 中， \oplus 表示膨脹操作運算子。 z 表示每次移動遮罩時，被賦值 255 的格點 $A(x, y)$ ，其條件必須滿足「交集並非空集合」。以圖 5.7 為例，此圖說明膨脹操作的示意圖概念。圖 5.7(a) 中，綠點表示遮罩以基準點 $B(i, j)$ 移動於影像的格點 $A(x, y)$ ，紅點表示兩者該移動下的交集情況。當遮罩移動完影像的每個格點後，即膨脹操作完成，影像 A 將會有額外新增的白色格點，如圖 5.7(b) 的綠點。從圖 5.7 可知，影像 A 經過膨脹處理後，原先像素值為 255 的格點周遭皆有擴大的現象。



(a) 將遮罩移動於影像之每個格點，綠點表示遮罩基準點，紅點表示該移動下之交集情況。



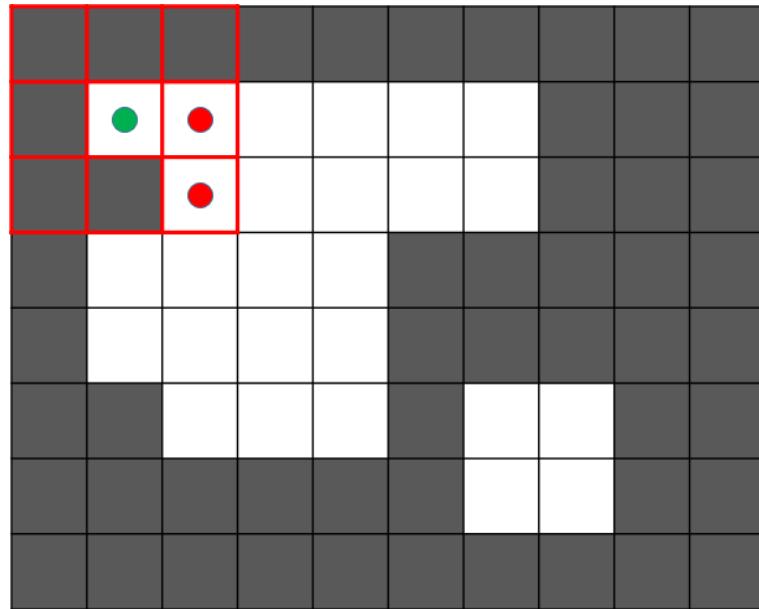
(b) 完成遮罩在每個格點的移動後，影像新增部份額外被賦值 255 的格點（綠點處）。

圖 5.7: 膨脹操作示意圖。

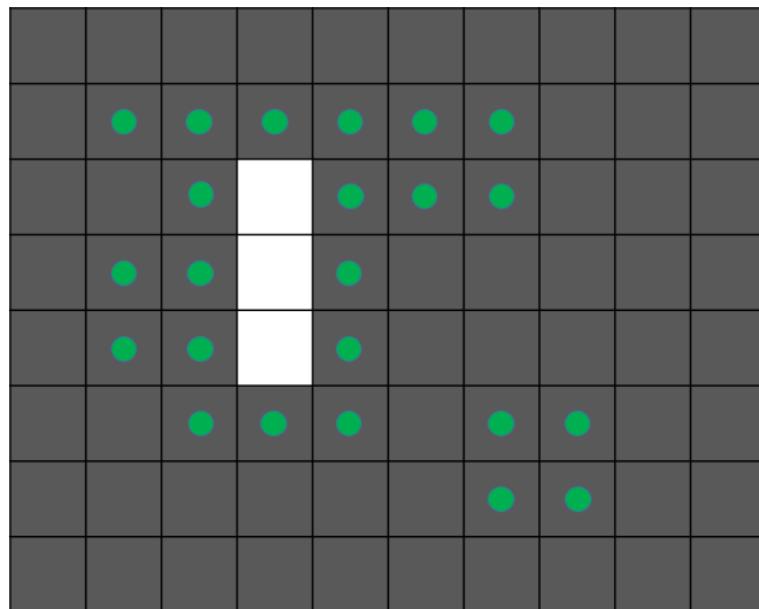
- 侵蝕 (Erosion): 將遮罩以中心點 $B(i, j)$ 為基準點，依序移動於影像的 A 的各個格點 $A(x, y)$ 。若遮罩 B 相等於 $A(x, y)$ 鄰近的格點，即 $(2k+1) \times (2k+1)$ 個格點像素值完全相同，則令格點 $A(x, y)$ 為 255；不完全相同則令 $A(x, y)$ 為 0。數學定義如下

$$A \ominus B = \{z | (\hat{B})_z \subseteq A\} \quad (5.7)$$

式 (5.7) 中， \ominus 表示侵蝕操作運算子。 z 表示每次移動遮罩時，被賦值 255 的格點，其條件必須滿足遮罩 B 包含於影像 A ，即 $(2k+1) \times (2k+1)$ 個格點像素值完全相同。圖5.8以圖像方式說明侵蝕操作概念，圖5.8(a)中，綠點表示遮罩以基準點 $B(i, j)$ 移動於影像的格點 $A(x, y)$ ，紅點表示兩者該移動下的交集情況。當遮罩移動完影像的每個格點後，即侵蝕操作完成，原先影像 A 的部份白色格點像素值因不滿足式 (5.7) 條件而被賦值 0，並且變為黑色格點，如圖5.8(b)的綠點。從圖5.8可知，影像 A 經過侵蝕處理後，原先像素值為 255 的格點周遭可能產生縮減的現象。



(a) 將遮罩移動於影像之每個格點，綠點表示遮罩基準點，紅點表示該移動下之交集情況。



(b) 完成遮罩在每個格點的移動後，部份原先影像的白色格點像素值被賦值 0 而變為黑色格點（綠點處）。

圖 5.8: 侵蝕操作示意圖。

3. 閉運算 (Closing)：將影像 A 先進行膨脹後，再進行侵蝕，數學定義如式 (5.8)，符號 \bullet 表示閉運算之運算子。閉運算處理中，先進行膨脹操作可使白色格點變多、區域擴大，填補部份缺口。再進行侵蝕操作則可使擴大的白色格點縮減，使其約略符合原先白色格點之數量。如圖5.9示意，綠點表示



之格點為經過閉運算處理後，受到填補的部份。

$$A \bullet B = (A \oplus B_1) \ominus B_2$$

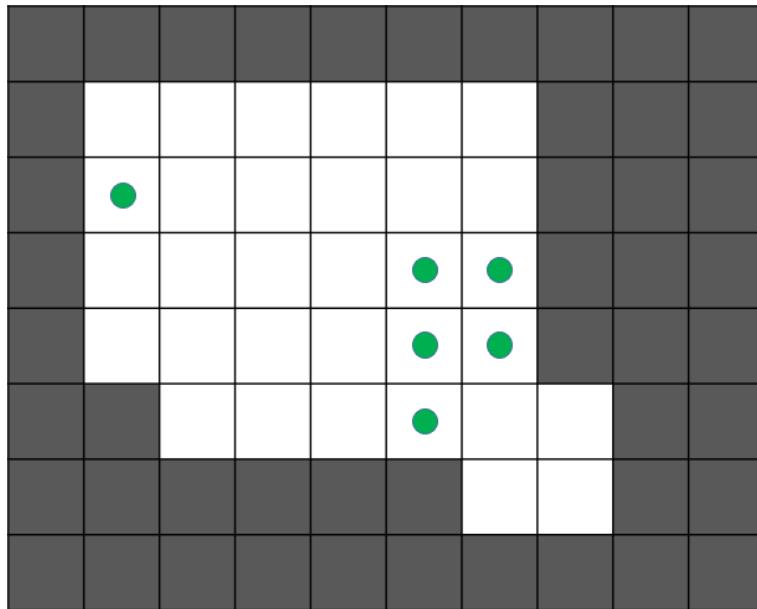


圖 5.9: 閉運算操作示意圖，綠點表示經過閉運算後額外新增之白色格點，與原影像（圖5.6(a)）相比，部份缺口已受到填補。

根據上述形態學操作的介紹，若我們應用於圖5.5的旋轉處理結果，則可如先前所說，將毛邊與缺口進行修補，如圖5.10所示。在圖中，可發現紅框處線條的毛邊與缺口已減少許多。

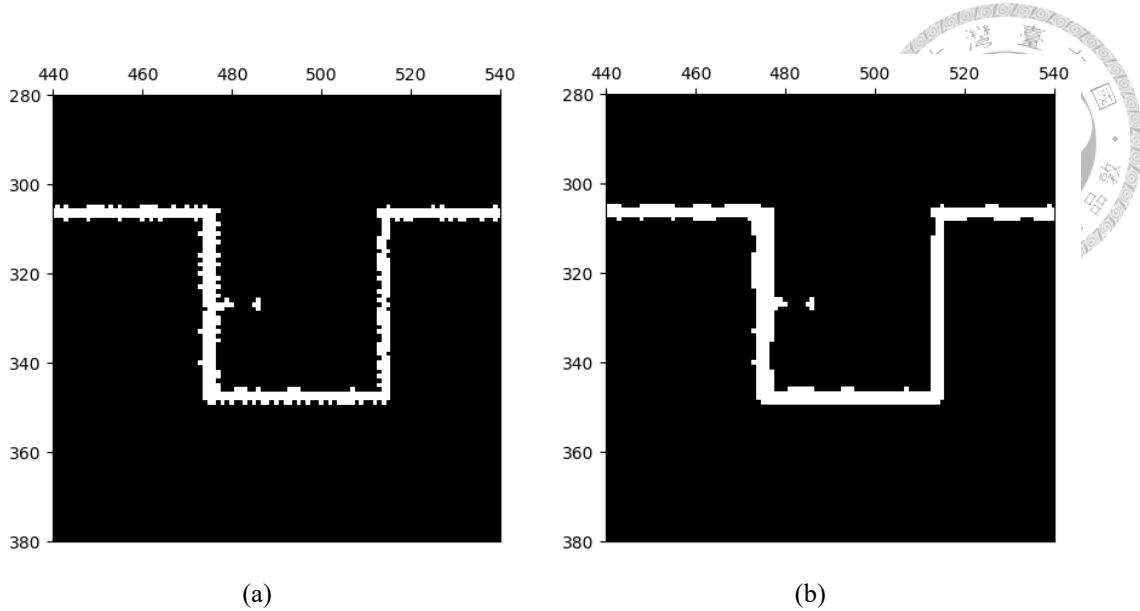


圖 5.10: 地圖完成旋轉處理後，部份線段產生毛邊與缺口，故經過形態學之閉運算處理進行修補。圖5.10(a)為原旋轉處理後的影像區塊；圖5.10(b)為經過閉運算操作的影像區塊，可看見毛邊與缺口受到修補的效果。

5.4 興趣點採樣

經過前一小節的閉運算操作後，格點佔據地圖影像的線段主體已進行初步修補。但在使用特徵擷取演算法之前，我們仍希望線段主體可更加筆直、無任何缺口，因此我們仍需將目前的主體線段進一步修正。修正的方式分為兩個步驟，首先在目前線段中採樣興趣點，接著以興趣點的位置將線段重新繪製，作為新的線段主體。根據上述說明，此小節將介紹如何在線段主體中擷取興趣點。

假設二值化影像 A 為大小 $m \times n$ 的二維矩陣，另外定義一運算遮罩 B 為大小 $(2k + 1) \times (2k + 1)$ 的二維矩陣，且 $2k + 1 \leq m$ 、 $2k + 1 \leq n$ 。我們採用類似形態學的概念，將遮罩於影像中平移，並依照特定條件篩選興趣點，篩選方式如下

步驟 1 新建一大小與影像 A 相同的二維矩陣 A' ，其中像素值皆設定為 0。

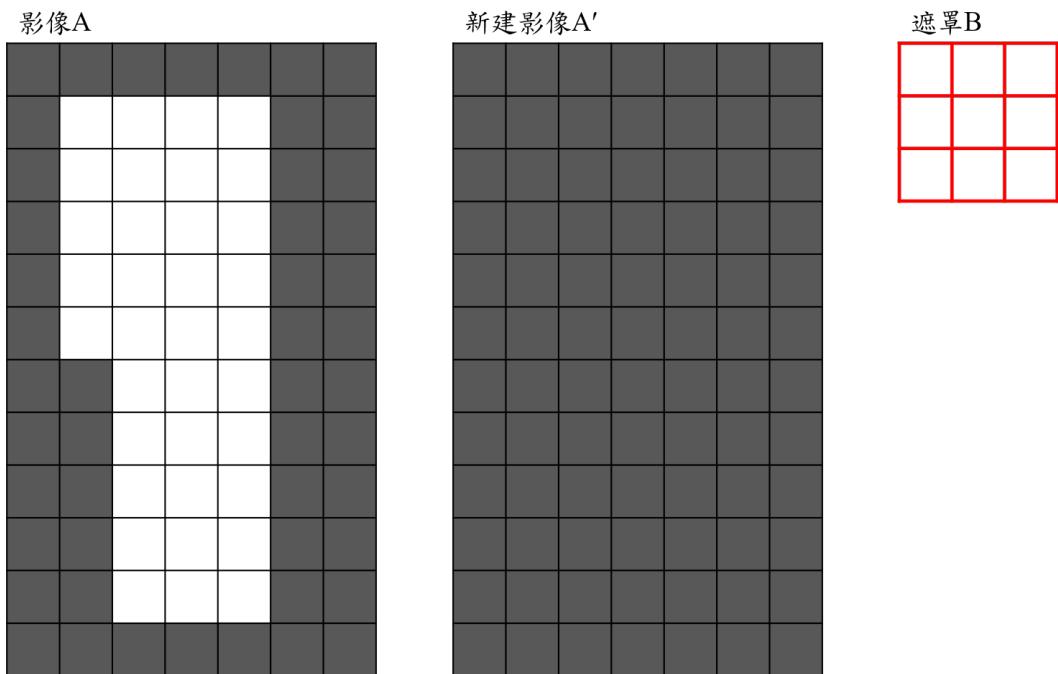
步驟 2 接著，將遮罩 B 以中心點為基準點，依序移動於影像的格點 $A(x, y)$ 。若遮罩 B 與影像 A 的關係可滿足

$$(\hat{B})_z \subseteq A \quad (5.9)$$

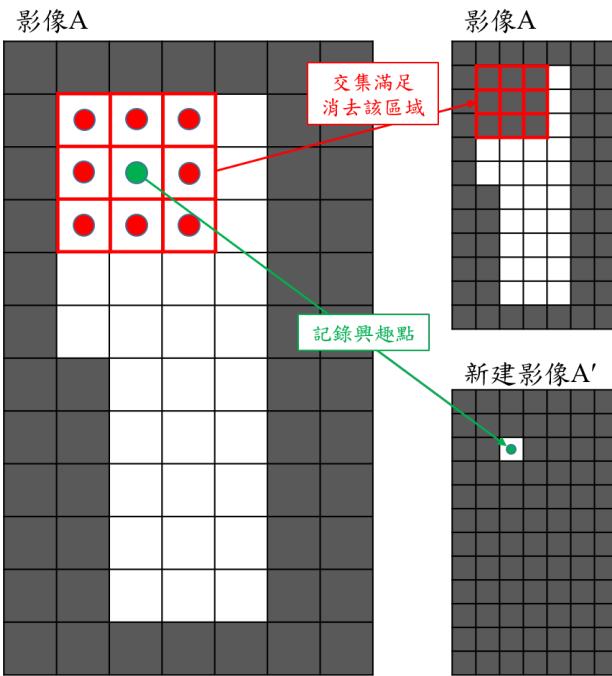
即遮罩 B 與格點 $A(x, y)$ 周遭的格點像素值全部相同，表示格點 $A(x, y)$ 為興趣點，故使新影像格點 $A'(x, y)$ 像素值為 255 來紀錄該興趣點位置。另外，為使此採樣方法不同於侵蝕操作，在平移遮罩於下一位置前，先將格點 $A(x, y)$ 及其周遭共 $(2k + 1) \times (2k + 1)$ 個格點像素值全部賦值為 0。

步驟3 反覆操作第2點直到遮罩移動完影像A的每個格點，便結束興趣點的採樣。

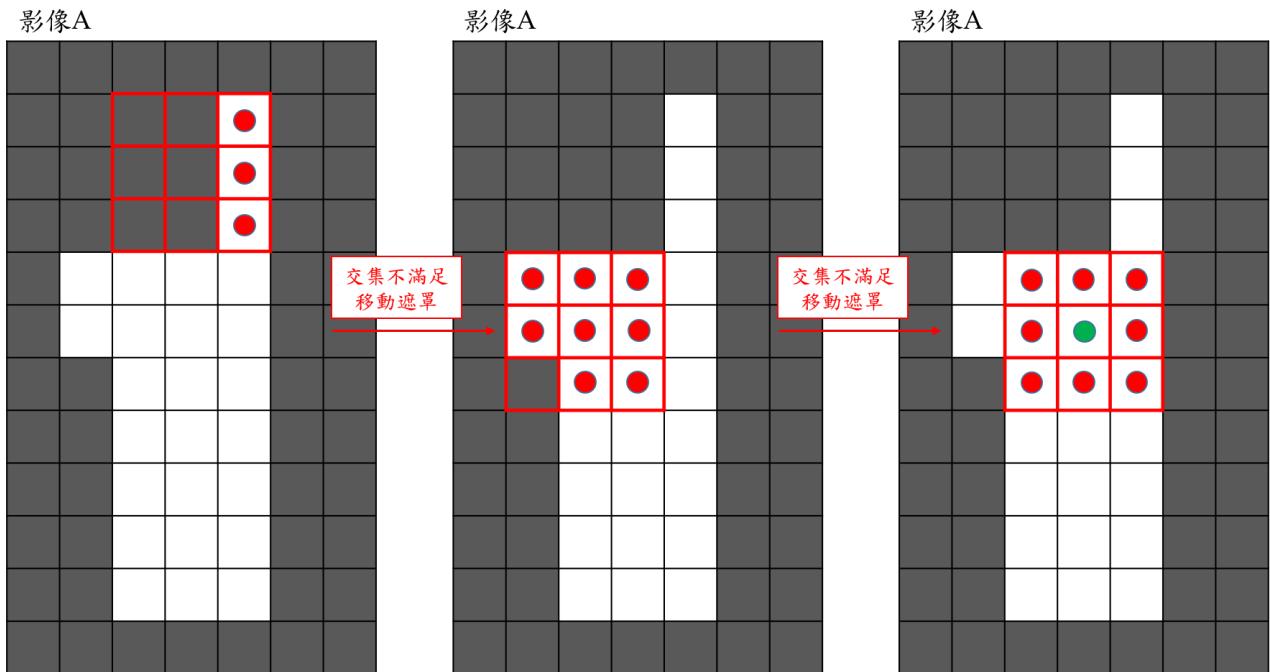
圖5.11將上述步驟轉為圖像進行說明。圖5.11(b)~5.11(d)詳細圖像化了步驟2中，遮罩如何在影像中檢測格點的交集關係，並紀錄符合條件（式(5.9)）的興趣點於新建影像A'。在本研究的興趣點採樣方法下，原先寬度不一的線條（圖5.11(a)之影像A）可擷取出數個興趣點（圖5.11(e)之新建影像A'）。



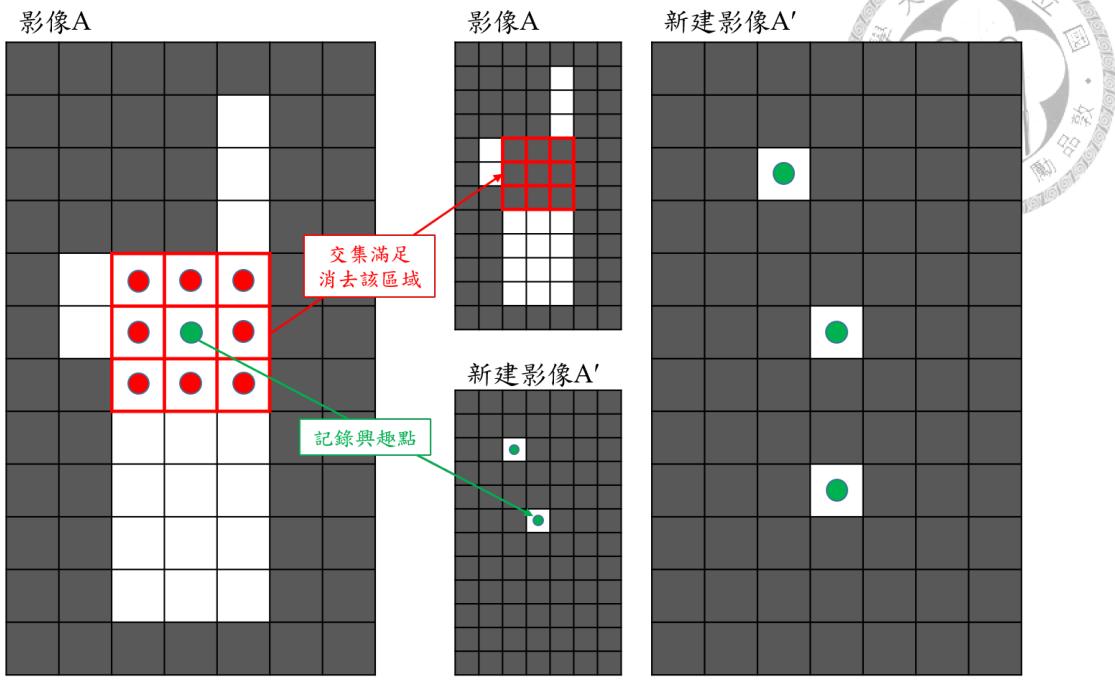
(a) 興趣點採樣之各二維矩陣。影像A中各列的佔據區域寬度可能不相同，因此進行興趣點採樣。



(b) 遮罩 B 於影像 A 查找興趣點，滿足交集情況便記錄興趣點位置於新建影像 A' (右下圖之綠點處)，並於影像 A 中消除交集佔據區域 (右上圖之紅框處)。



(c) 持續將遮罩 B 移動於影像 A 之各格點，搜尋興趣點位置。



(d) 再次取得興趣點位置，紀錄於新建影像 A' 與消除
交集格點。

(e) 興趣點採樣完成之示意圖。

圖 5.11: 興趣點採樣流程圖。

同樣地，我們以原先圖5.5的影像為例，經過閉運算處理後（圖5.10），再以上述步驟進行興趣點採樣，可見結果如圖5.12。從圖中可見先前描述，原先寬度不一的線條，現已被轉換、擷取出數個興趣點。接著在下一節中，我們將介紹如何使用已擷取的興趣點對原先影像的主體線段進行修正。

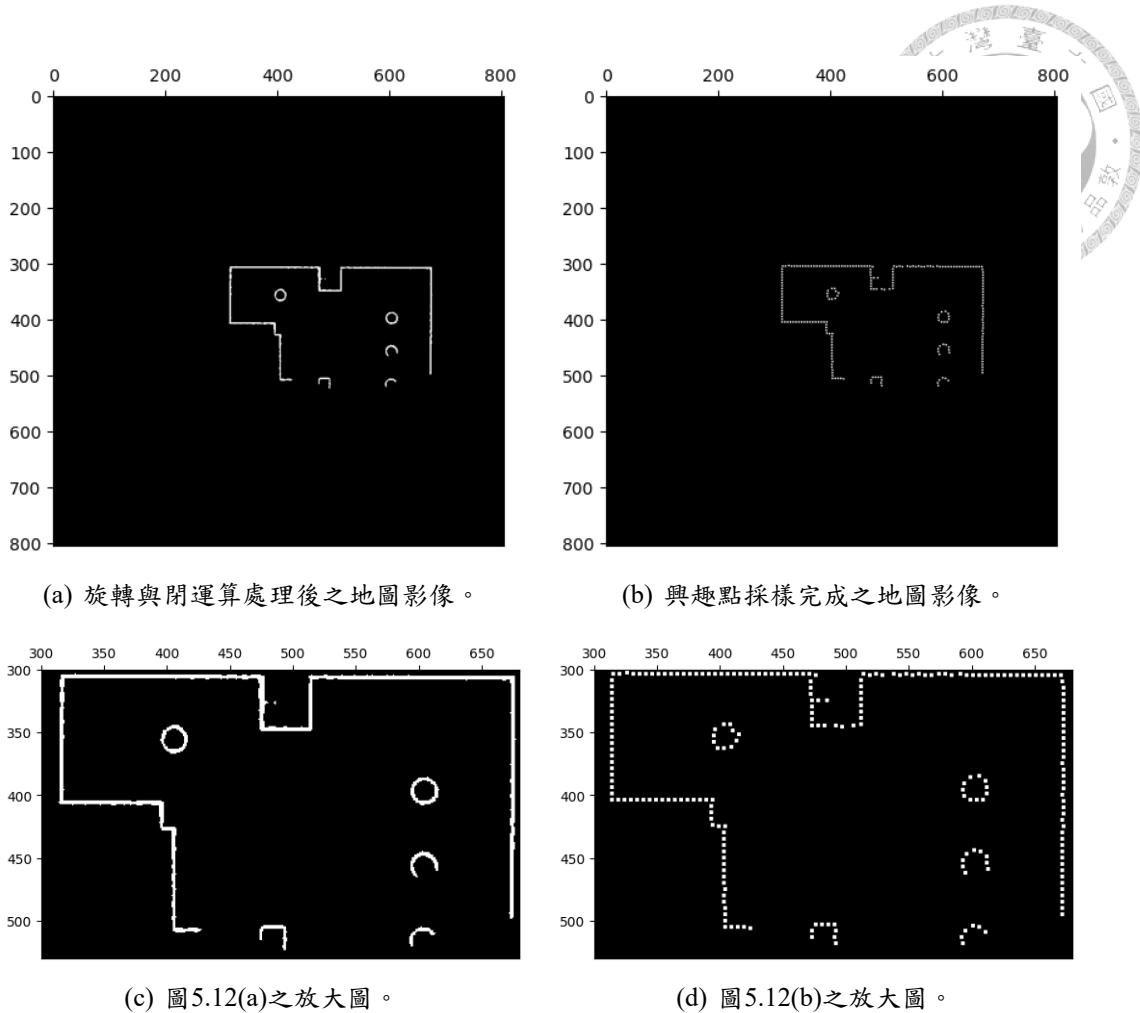


圖 5.12: 影像興趣點採樣示意圖。圖5.12(a)為原先影像旋轉與閉運算處理後的結果；圖5.12(b)為圖5.12(a)經過採樣後，紀錄興趣點的影像；圖5.12(c)與圖5.12(d)則分別為前兩圖的放大圖。

5.5 主體結構修正

前一節中，我們已將所有興趣點位置紀錄於影像 A' 中，接著將繼續以這些興趣點重新繪製線段來修正原先格點佔據地圖的主體結構。

步驟1 首先以曼哈頓距離作為標準，選擇距離影像原點最近的興趣點作為起始點 $A'(x_i, y_i)$ 。假設兩點座標為 $P_1(x_1, y_1)$ 、 $P_2(x_2, y_2)$ ，則曼哈頓距離為兩點水平距離、垂直距離的總和，即

$$d(P_1, P_2) = |x_1 - x_2| + |y_1 - y_2| \quad (5.10)$$

為在運算過程中避免小數點運算，故採以曼哈頓距離作為選擇標準。



步驟2 以點 $A'(x_i, y_i)$ 為參考點，再次以曼哈頓距離搜尋周遭距離最近的點 $A'(x_t, y_t)$ 作為目標點。

步驟3 將兩點 $A'(x_i, y_i)$ 、 $A'(x_t, y_t)$ 連線，並取出連線所通過的全部 n 個格點 $\{A'(x_1, y_1), A'(x_2, y_2), \dots, A'(x_n, y_n)\}$ 。根據取出的格點位置，在原地圖影像 A 的相同位置上計算白色格點（佔據區域，像素值 255）數目與總數 n 的比值 R_{it} （下標 i 表示參考點， t 表示目標點）。計算方式為像素值總和與 $255 \times n$ 的比值，即

$$R_{it} = \frac{\sum_{k=i}^n A(x_k, y_k)}{255 \times n} \quad (5.11)$$

若比值 R_{it} 大於一閾值 R_{thres} ，如式 (5.12)，則可確認兩點 $A'(x_i, y_i)$ 、 $A'(x_t, y_t)$ 在原先影像上本為線段，因此可加以修正；若不滿足式 (5.12)，則重新選擇目標點。

$$R_{it} > R_{thres} \quad (5.12)$$

步驟4 根據步驟3的確認，若兩點 $A'(x_i, y_i)$ 、 $A'(x_t, y_t)$ 進行修正，則可計算向量 v_{it} 為

$$v_{it} = (x_t - x_i, y_t - y_i) \quad (5.13)$$

根據式 (5.13) 的向量 v_{it} ，以點 $A'(x_i, y_i)$ 為參考點，將點 $A'(x_t, y_t)$ 的位置進行平移。平移方式如下：

首先確認 $|x_t - x_i|$ 、 $|y_t - y_i|$ 的量值，因此可定義 3 種關係為

- (1) 若 $|x_t - x_i| < |y_t - y_i|$ ，則將點 $A'(x_t, y_t)$ 進行 x 向平移。
- (2) 若 $|x_t - x_i| > |y_t - y_i|$ ，則將點 $A'(x_t, y_t)$ 進行 y 向平移。
- (3) 若 $|x_t - x_i| = |y_t - y_i|$ ，則依先前方向 (x 或 y 向) 額外新增一格點，再重新進行平移判斷。

若進行 x 向平移，則將格點 $A'(x_t, y_t)$ 的像素值改為 0，並將格點 $A'(x_i, y_t)$ 的像素值改為 255，意即將格點 $A'(x_t, y_t)$ 平移至 $A'(x_i, y_t)$ ；若進行 y 向平移，則將格點 $A'(x_t, y_t)$ 的像素值改為 0，並將格點 $A'(x_t, y_i)$ 的像素值改為 255，意即將格點 $A'(x_t, y_t)$ 平移至 $A'(x_t, y_i)$ ；若需額外新增格點，如情況 (3)，則視先前平移方向而定。若

- 先前方向為 x 向，則新增格點 $A'(x_i, y_t)$ 。
- 先前方向為 y 向，則新增格點 $A'(x_t, y_i)$ 。
- 無先前方向，不進行偏移並重新選取參考點與目標點。

上述方式進行了格點平移或格點新增。然而，為確保此格點位置修正不會過度影響原先影像主體，我們再次仿照第 3 點，將參考點 $A'(x_i, y_i)$ 與修正（或



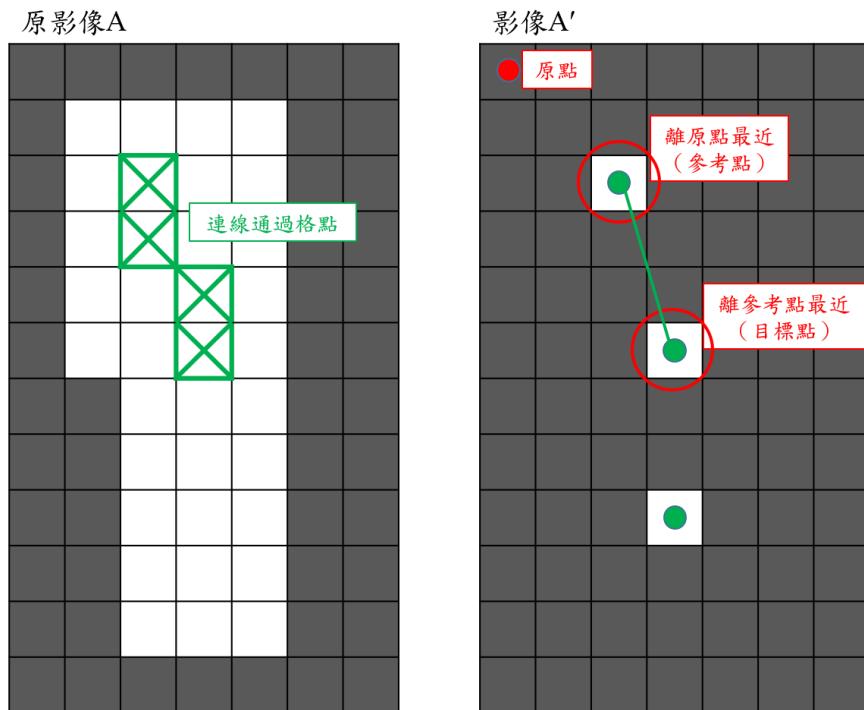
新增) 點 $A'(x_i, y_t)$ 或 $A'(x_t, y_i)$ 連線，並取出連線所通過的全部格點，以式(5.11)計算白色格點所佔比值 $R_{it'}$ (下標 i 表示參考點， t' 表示修正點)。若比值可大於式(5.12)的閾值 R_{thres} ，即

$$R_{it'} > R_{thres} \quad (5.14)$$

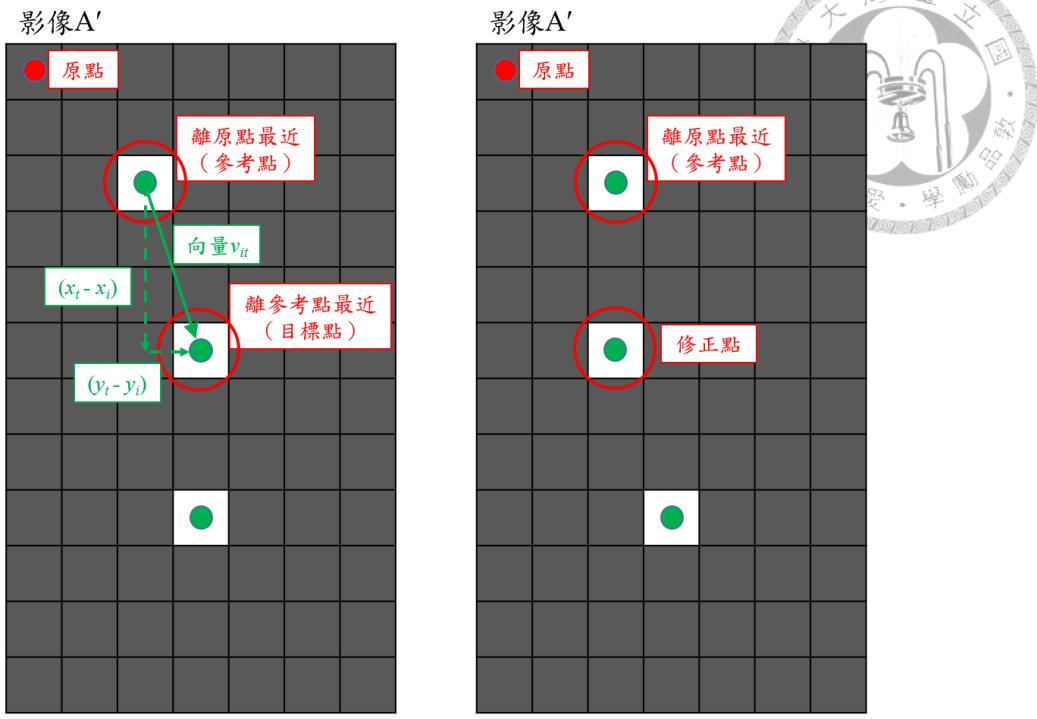
表示此修正不會過度影響原先生主體結構，因此接受該修正。若不滿足式(5.14)，則表示該修正可能過度改變主體結構，故拒絕此修正格點。

步驟5 根據步驟4的結果，可將修正點 ($A'(x_i, y_t)$ 或 $A'(x_t, y_i)$) 重新作為參考點，查找鄰近目標點，並重複進行步驟3~5，直至所有興趣點皆完成修正檢驗。倘若步驟4未計算出修正點，則使原先目標點 $A'(x_t, y_t)$ 為參考點進行演算。

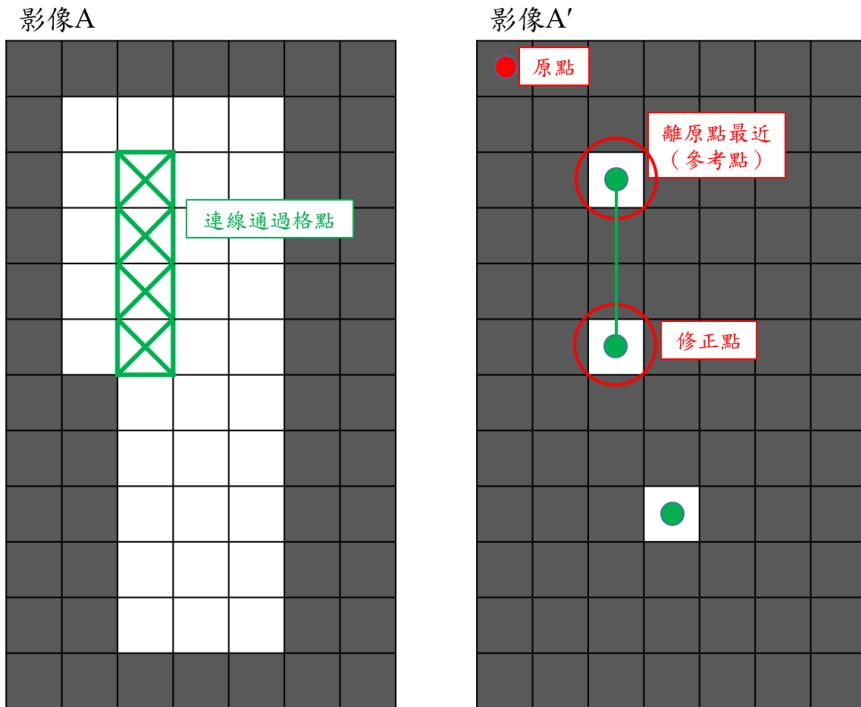
圖5.13的流程中，以圖5.11(a)中影像 A 與圖5.11(e)中紀錄興趣點的影像 A' 為例，將上述步驟以圖像化方式進行說明。圖5.13(a)顯示步驟1~3如何選取參考點、目標點，並以連線的白色格點比例確認是否進行修正(式5.12)。若進行修正，則計算兩點向量 v_{it} ，並決定修正方向與修正點位置，如圖5.13(b)(步驟4)。修正點位置決定後，再次與參考點連線確認是否接受修正(式5.14)，如圖5.13(c)。修正完成後，重新選取參考點與目標點，如圖5.13(d)。反覆操作，直到所有興趣點皆修正完成。



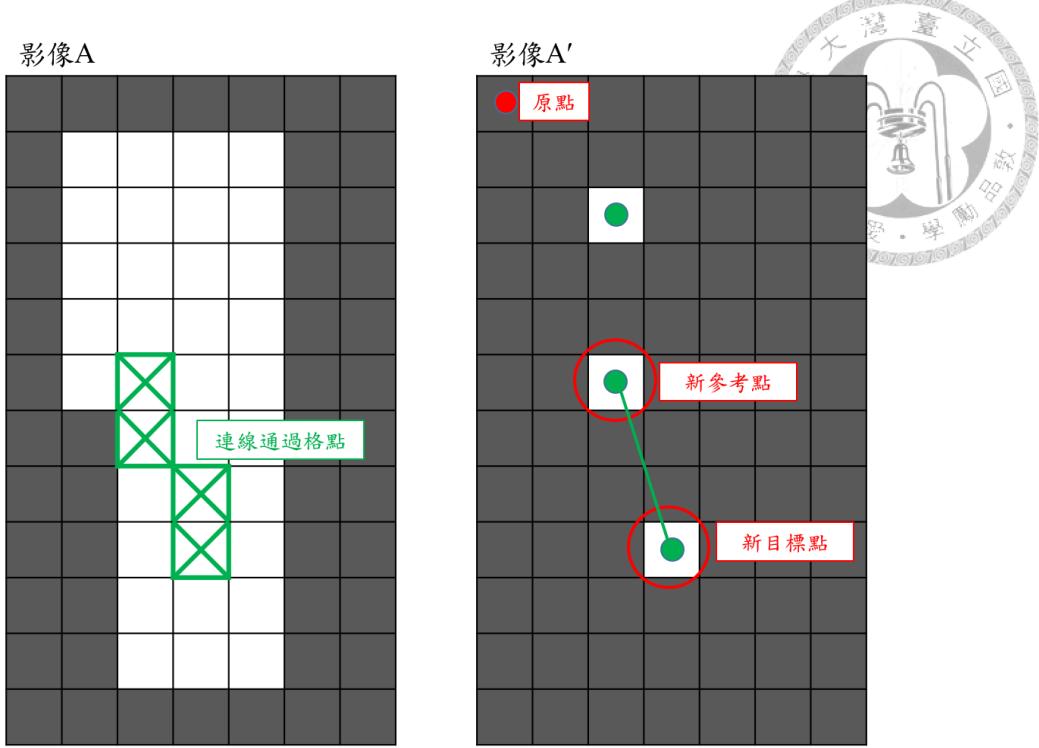
(a) 依曼哈頓距離選擇參考點 ($A'(x_i, y_t)$) 與目標點 ($A'(x_t, y_i)$)，並連線計算白色格點所佔比值 R_{it} ，並決定是否進行修正。(步驟1~3)



(b) 計算參考點與目標點的向量 v_{it} ，並根據步驟 4 決定目標點的修正位置，圖中因 $|y_t - y_i|$ 較小，屬情況 (2)，故依照 y 向分量 $(y_t - y_i)$ 將目標點向左平移。(步驟 4)



(c) 選定目標點的修正位置 (修正點) 後，再次將參考點與修正點連線，並計算白色格點所佔比例 $R_{it'}$ ，並決定是否接受修正。(步驟 4)



(d) 接受修正後，將前修正點作為參考點，並重新選擇目標點再次進行修正。(步驟 5)

圖 5.13: 興趣點位置修正流程圖。

另外，對於步驟 4 中提到的情況 (3)，如圖5.14，將依照先前的修正方向 (x 或 y 向)，新增修正點 (如紅色虛線圓圈)，同時也不刪除現目標點。接著以參考點與新增修正點的連線與佔據格點比例，確認是否接受該新增格點。若接受，則以新增修正點為參考點，修正現目標點的位置；若不接受，則直接以現目標點為參考點，搜尋下一鄰近目標點進行修正。

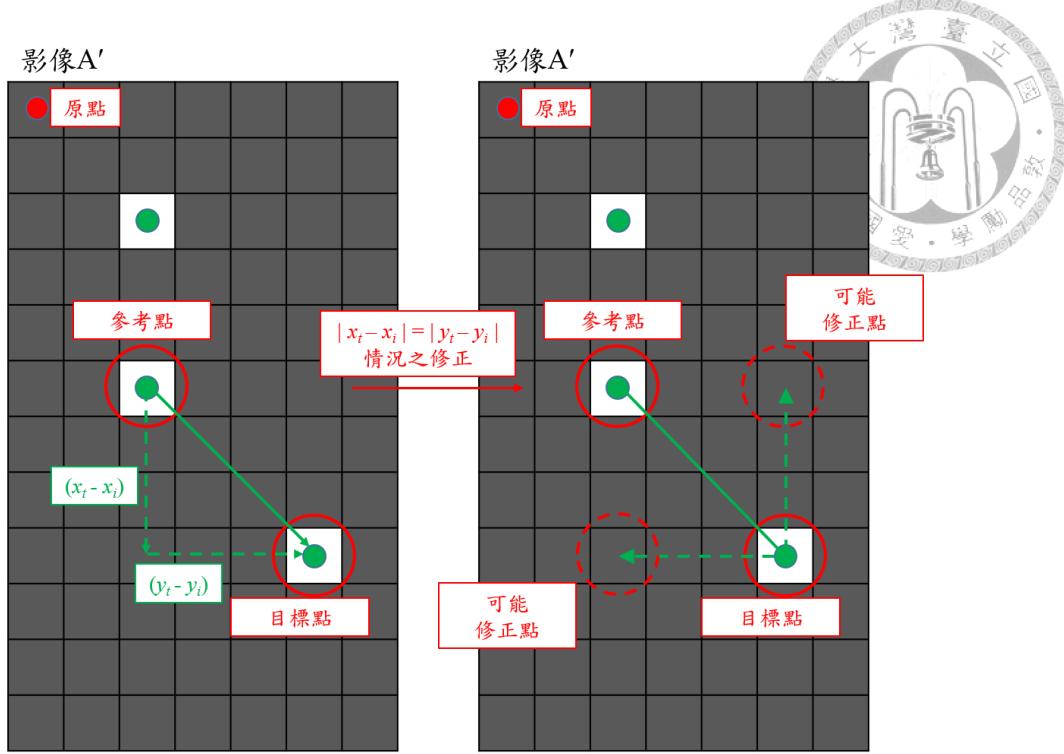


圖 5.14: 興趣點位置修正之情況 (3)。

介紹完上述主體線段修正步驟後，我們以圖5.12的興趣點影像為例，重新繪製主體線段，完成影像修正，如圖5.15。與原先地圖影像相比，各線段寬度皆已統一，同時線段也較為筆直。然而，由於修正仰賴於興趣點的位置，因此在某些區域可能產生微小轉折，如圖5.15(d)紅圈處。另外，同樣從圖5.15(d)中也可見，以興趣點進行修正，對於原先為圓圈、弧線的部份，會使其形狀大幅改變（紅框處），這是由於修正演算法中皆以平行座標軸（ x 、 y 軸）為主而造成的，此問題將留待第七章再進行詳細討論。

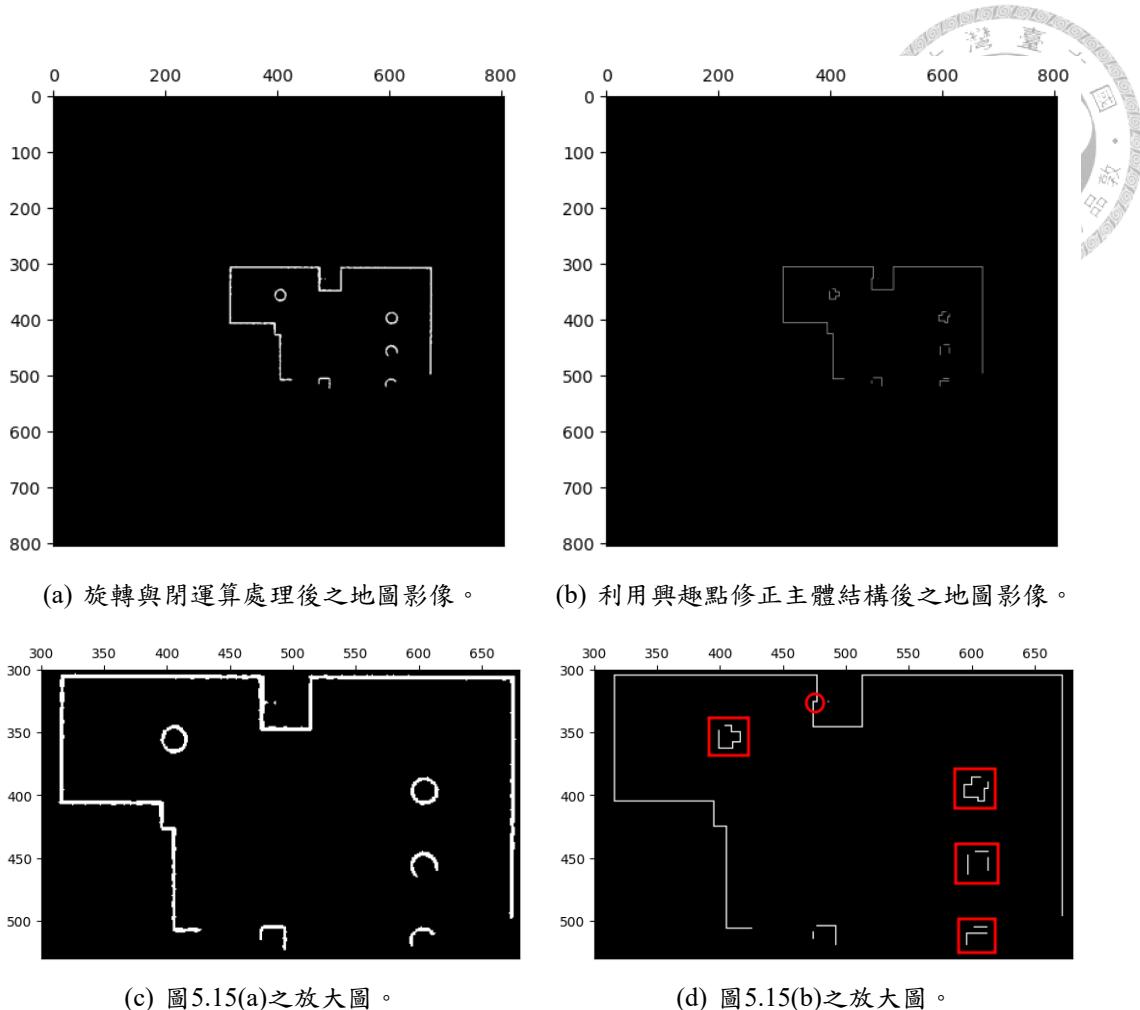


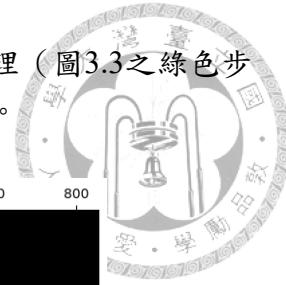
圖 5.15：影像主體結構修正比較圖。圖5.15(a)為原先影像旋轉與閉運算處理後的結果；圖5.15(b)則是利用興趣點影像（圖5.12(b)）進行結構修正後的結果；圖5.15(c)與圖5.15(d)則分別為前兩張影像的放大圖。從比較圖結果可見，主體線段修正後較為筆直且寬度統一。

5.6 影像修正後合併結果討論

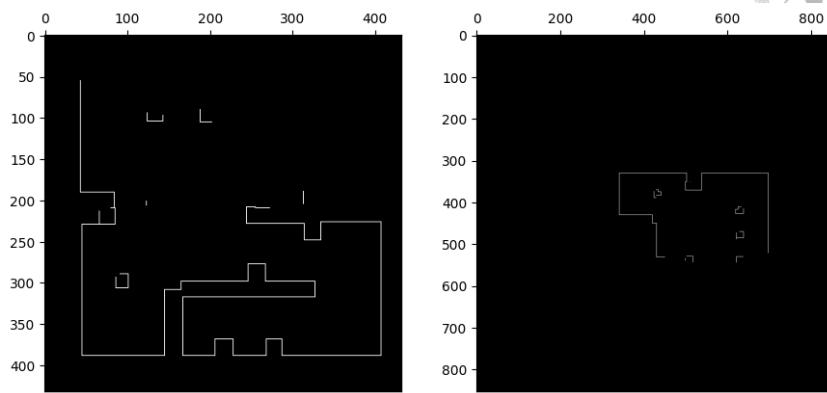
介紹完本研究提出的地圖影像修正之前處理步驟與概念後，我們接著將實際應用至表3.2的測試案例。接下來，我們會先說明經過影像修正後，地圖的特徵擷取及匹配產生了何種影響，再接著展示地圖的合併成果。

5.6.1 影像修正對特徵之差異

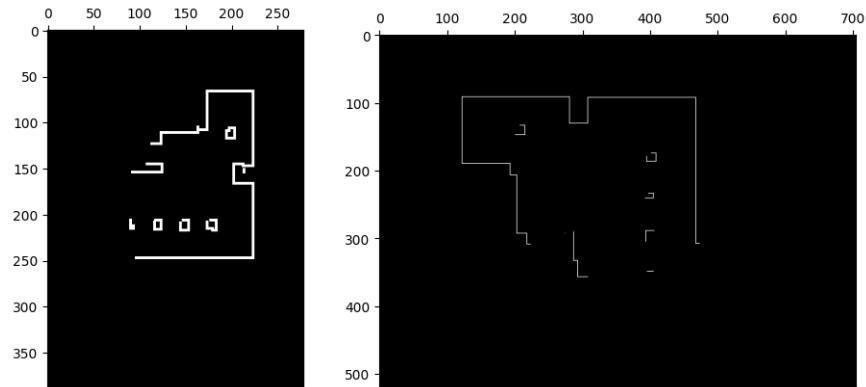
首先，我們先依照影像修正步驟（圖3.3之藍色步驟）對每個測試案例（圖3.4(b)～3.4(e)）進行修正，如圖5.16，接著再對修正後的影像進行特徵擷取。但我們發現，若直接應用 SIFT 演算法，因目前佔據區域線段寬度在影像中的比例



過小，可能無法擷取出特徵。因此我們必須再加入一個膨脹處理（圖3.3之綠色步驟），使佔據區域的比例增加，才可有效擷取特徵點，如圖5.17。

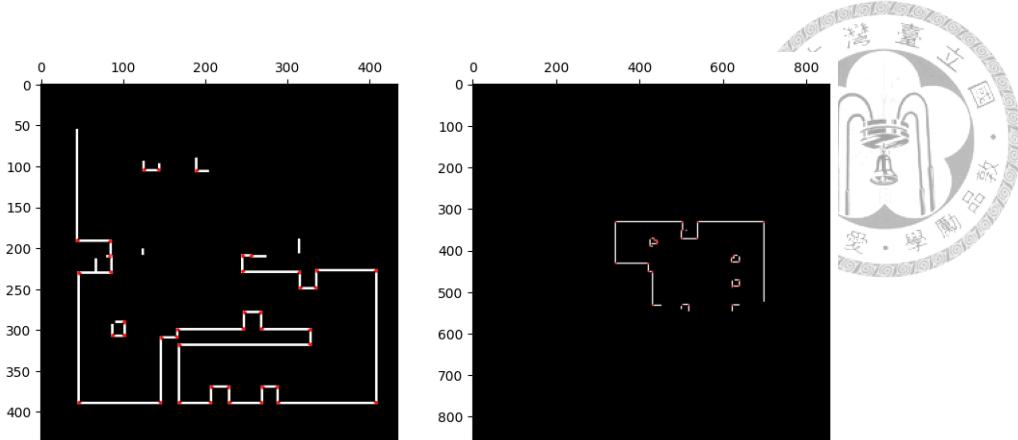


(a) 局部測試地圖一（圖3.4(b)）經過影像修正處理之結果。
(b) 局部測試地圖二（圖3.4(c)）經過影像修正處理之結果。

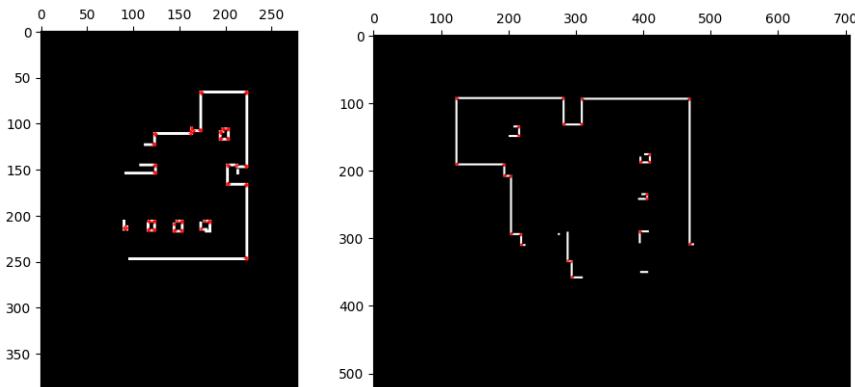


(c) 局部測試地圖三 (d) 局部測試地圖四（圖3.4(e)）經過影像修正處理之結果。

圖 5.16: 表3.2之測試案例經過影像修正處理之結果。所有案例經過修正後，佔據區域的線段寬度皆為 1 格。因此相對於影像尺寸，地圖一、三、四的線段寬度比例過小，導致色彩表示上較不明顯，同時也將影響後續特徵擷取。



(a) 圖5.16(a)之特徵擷取結果。案例(b) 圖5.16(b)之特徵擷取結果。案例一經過影像修正後，因線段寬度於二經過影像修正後，因線段寬度於影像中的比例過小，必須經過額外膨脹處理後才能有效擷取特徵。膨脹處理後才能有效擷取特徵。



(c) 圖5.16(c)之特徵擷取結果。案例四經過影像修正後，雖寬度亦為1格，但因須經過額外膨脹處理後才能有效擷取特徵。
影像尺寸較其他案例小，故不必進行額外的膨脹處理。

圖 5.17: 針對修正後影像進行特徵擷取之結果，紅點表示被擷取出的特徵點。

從圖5.17中可發現，影像修正後的特徵點擷取，其位置大多位於佔據區域的線段轉折（角點）或線段端點，與先前第四章的測試結果（圖4.5）相比，我們可以確保特徵出現在特定位置。根據已擷取出的特徵點，我們將繼續進行特徵匹配。然而，第四章已說明過因格點佔據地圖的扭曲、不確定性因素，使用特徵的描述符向量進行匹配可能會導致匹配錯誤。本研究的影像修正雖排除上述的兩項原因，但擷取的特徵點因周遭影像相似（都是直角轉折），使得描述符向量無法突顯唯一性，導致無法使用向量進行特徵匹配。但我們上述也提及，本研究方法確保特徵出現在特定位置，因此我們將特徵匹配改為窮舉匹配方式（圖3.3之紅色步驟）。舉例而言，假設兩特徵集合 $P = \{p_1, p_2, \dots, p_n\}$ （來自地圖影像 M_1 ）與特徵集合 $Q = \{q_1, q_2, \dots, q_m\}$ （來自地圖影像 M_2 ），則配對關係為

$\{\{p_1, q_1\}, \dots, \{p_1, q_m\}, \dots, \{p_n, q_1\}, \dots, \{p_n, q_m\}\}$ ，共 $n \times m$ 個特徵配對。

建立匹配後，則可依照的 RANSAC 演算法計算相對變換關係。然而，經過修正後的地圖與原本的格點佔據地圖有些微差異，故相對變換的計算上可能也含有誤差。因此，基於 RANSAC 演算法計算出的合併模型，我們將使用原本地圖的佔據區域格點搭配迭代最近點演算法（Iterative Closest Point，ICP）[53] 來優化合併模型（圖3.3之黃色步驟）。

ICP 是一種點雲（Point Cloud）匹配、對齊方法，主要基於最小平方和來求解一空間變換，使得兩個輸入點雲的距離差最小化進而達到點雲匹配與對齊，其效果如圖5.18所示，關於 ICP 的原理與數學證明、推導可參見 [53,54]。

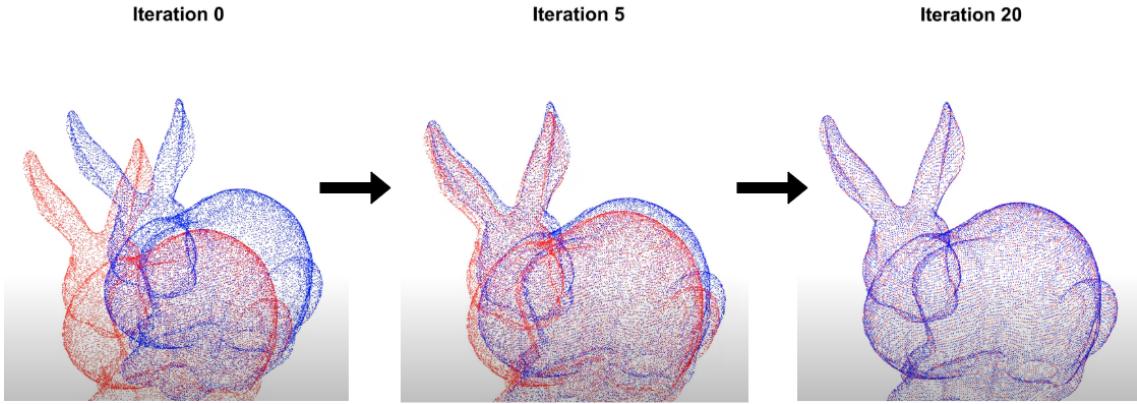


圖 5.18: 迭代最近點演算法之效果示意圖 [4]

接著，我們將介紹如何利用 ICP 進行合併模型優化，相關步驟如下：
令欲合併的兩局部地圖 M_1 、 M_2 之佔據區域格點集合（如圖5.12(c)之白色格點）分別為 $P_{occ} = \{p_{occ,1}, \dots, p_{occ,n}\}$ 、 $Q_{occ} = \{q_{occ,1}, \dots, q_{occ,m}\}$ ，經過 RANSAC 演算法計算出的合併模型可初步將地圖 M_1 變換為 M'_1 ，並對齊地圖 M_2 ，即

$$M'_1 = T_{RANSAC} M_1 \quad (5.15)$$

此時佔據格點集合 P_{occ} 也應變換為

$$P'_{occ} = T_{RANSAC} P_{occ} = \{p'_{occ,1}, \dots, p'_{occ,n}\} \quad (5.16)$$

接著，將集合 P'_{occ} 與 Q_{occ} 代入 ICP 演算法計算二次對齊模型。迭代過程中，點集 P'_{occ} 的各點會在點集 Q_{occ} 中搜尋鄰近點，並建立對應關係為 $\{p'_{occ,i}, q_{occ,i}\}_{i=1,\dots,N_p}$ ，



並計算模型 T_{ICP} 為

$$R^*, t^* = \arg \min_{R,t} \sum_{i=1}^{N_p} \|q_{occ,i} - (Rp_{occ,i} + t)\|^2$$

$$T_{ICP} = \left[\begin{array}{c|c} R^* & t^* \\ \hline \mathbf{0} & 1 \end{array} \right] \quad (5.17)$$

其中， R^* 與 t^* 為對齊點集 P'_{occ} 、 Q_{occ} 的旋轉與平移矩陣。接著，將 RANSAC 的合併模型 T_{RANSAC} 與 ICP 的二次對齊模型 T_{ICP} 合併，則可輸出優化模型為

$$T_{optimized} = T_{ICP} \times T_{RANSAC} \quad (5.18)$$

此優化模型可將局部地圖 M_1 變換為 $M_1^{optimized}$ ，並與地圖 M_2 精準對齊，即

$$M_1^{optimized} = T_{optimized} M_1 \quad (5.19)$$

最後，通過熵濾波處理合併為全域地圖。說明完影像修正與計算變換關係的流程後，我們將於下一小節展示測試案例在本文方法下的合併結果。

5.6.2 合併測試

依照本研究方法，我們再次對表3.2的案例組合進行測試。同樣地，我們對於每個組合的參數關係皆依照第四章的說明進行調整，即組合一、三考量旋轉、平移參數 (θ 、 t_x 、 t_y)，組合二考量旋轉、平移、縮放參數 (θ 、 t_x 、 t_y 、 s)。經過前一小節說明的特徵點匹配方式，我們接著輸入至 RANSAC 與 ICP 演算法來計算地圖間的相對變換關係，最後也以合併可視化示意圖作為結果呈現。各組合測試結果如下：

- 組合一之合併結果如圖5.19，該合併結果與模擬環境（圖3.4(a)）相符。另外，與利用參考方法的結果相比（圖4.8），本文方法可計算出較為精準的相對變換關係，重疊區域的格點較為貼合。此時的接受指標為（下標 1 表示組合一，上標' 表示經過影像修正處理）

$$\omega'_1 = \frac{agr'_1}{agr'_1 + dis'_1} = 0.992072 > \omega_{thres} = 0.95 \quad (5.20)$$

不僅符合成功合併的判斷標準（大於 0.95），也較先前的接受指標 $\omega_1 =$

0.966292 提昇了 2.67% 的分數，如式 (5.21)。

$$\frac{\omega'_1 - \omega_1}{\omega_1} = \frac{0.992072 - 0.966292}{0.966292} \approx 0.0267$$

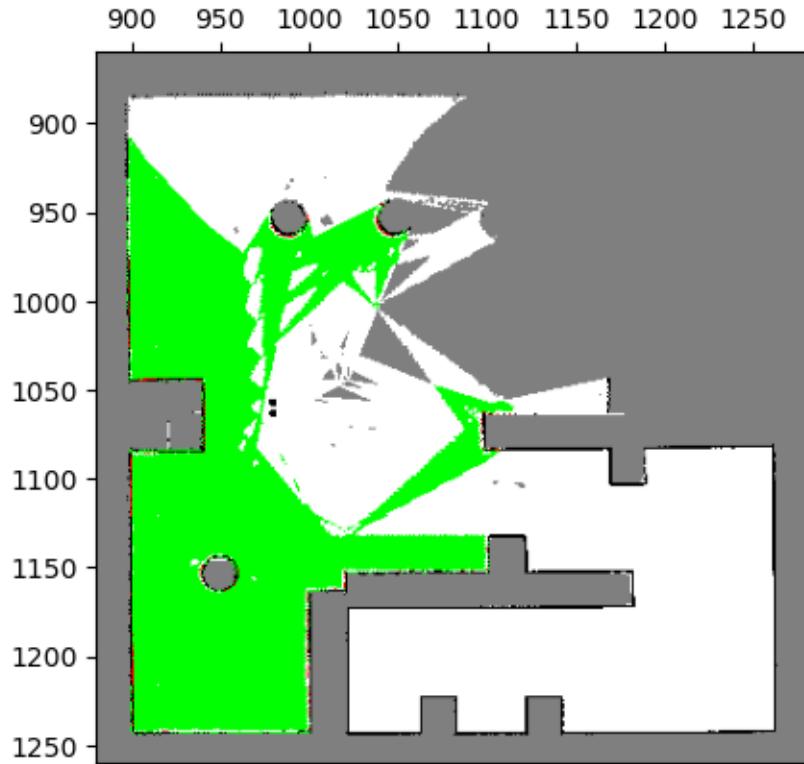


圖 5.19: 經過影像修正之組合一合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。

2. 組合二之合併結果如圖5.20，該示意圖與模擬環境（圖3.4(a)相符。此案例的接受指標則為（下標 2 表示組合二，上標' 表示經過影像修正處理）

$$\omega'_2 = \frac{agr'_2}{agr'_2 + dis'_2} = 0.989327 > \omega_{thres} = 0.95 \quad (5.22)$$

亦符合正確合併的標準。與使用參考方法的結果相比（圖4.9，指標分數 $\omega_2 = 0.935030$ ），本文方法不僅可正確合併測試案例，同時也提昇了 5.81% 的分數，如式 (5.23)。

$$\frac{\omega'_2 - \omega_2}{\omega_2} = \frac{0.989327 - 0.935030}{0.935030} \approx 0.0581 \quad (5.23)$$

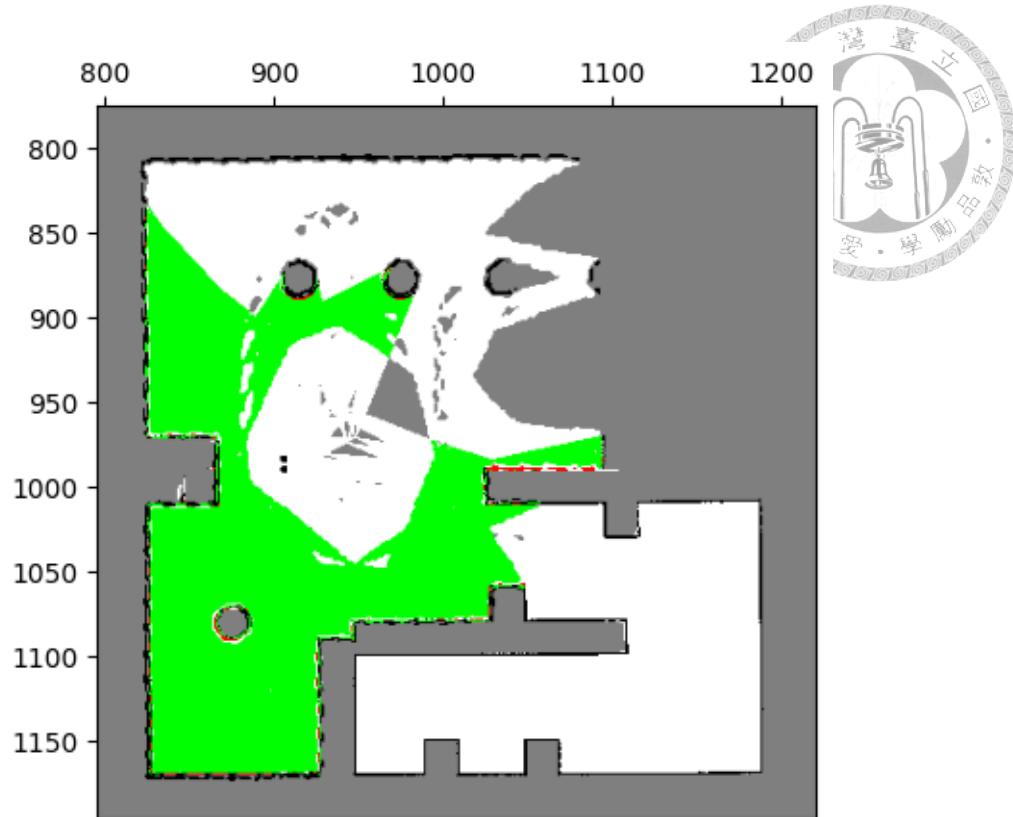


圖 5.20: 經過影像修正之組合二合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。

3. 組合三之合併結果如圖5.21，接受指標分數為（下標 3 表示組合三，上標 ' 表示經過影像修正處理）

$$\omega'_3 = \frac{agr'_3}{agr'_3 + dis'_3} = 0.979052 > \omega_{thres} = 0.95 \quad (5.24)$$

與先前兩個案例相同，組合三的結果符合模擬環境（圖3.4(a)），也大於接受指標的閾值，因此屬於成功合併。與參考方法的結果相比（圖4.10，接受指標 $\omega_3 = 0.903026$ ），我們提升了 8.42% 的分數，如式 (5.25)。

$$\frac{\omega'_3 - \omega_3}{\omega_3} = \frac{0.979052 - 0.903026}{0.903026} \approx 0.0842 \quad (5.25)$$

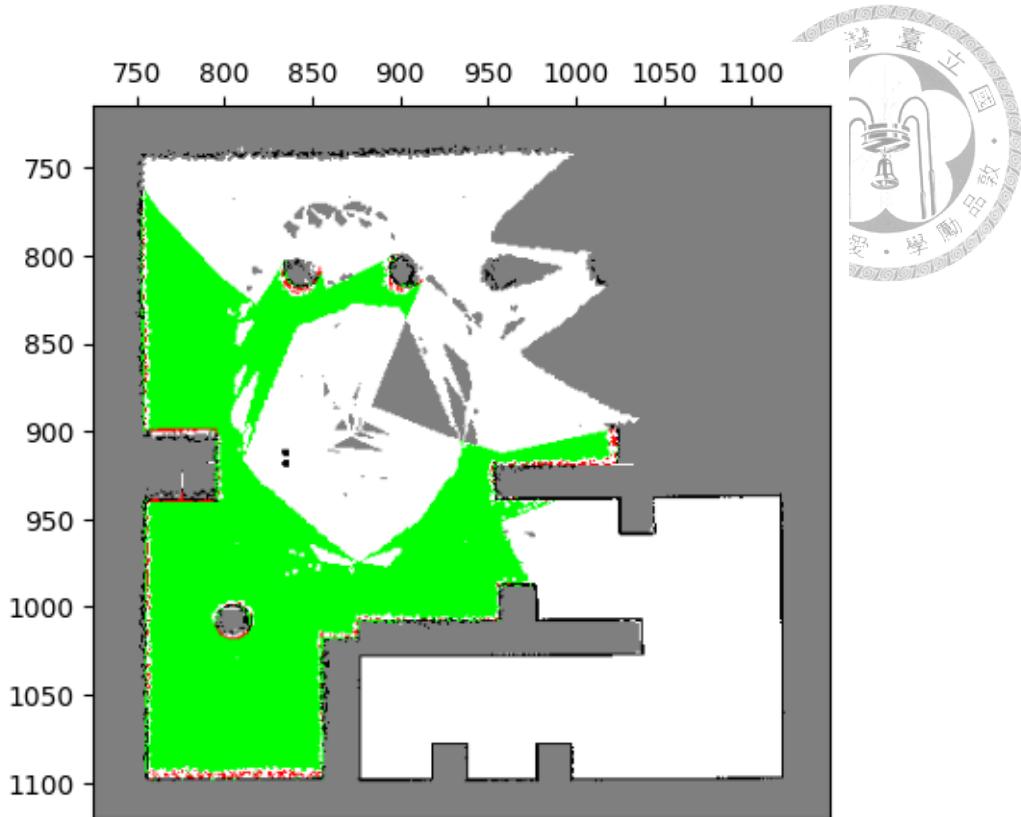


圖 5.21: 經過影像修正之組合三合併結果可視化示意圖。綠色區域為正確匹配格點，紅色區域為錯誤匹配格點，此合併情況下，人眼判斷與模擬環境相符。

上述的測試結果，我們一併整理於表5.1中。經過相同案例的測試，本研究方法不僅可有效的完成格點佔據地圖的合併任務，在合併效果上也較參考方法有顯著的提昇。然而，僅使用一種地圖環境進行測試，無法有效證明本研究方法的可行性。因此，我們將在下一章中額外對不同的地圖環境進行合併測試。

	參考方法之測試結果	本文方法之測試結果	接受指標提昇百分比
組合一	接受指標 $\omega_1 = 0.966292$ 圖4.8 (成功)	接受指標 $\omega'_1 = 0.992072$ 圖5.19 (成功)	+2.67%
組合二	接受指標 $\omega_2 = 0.935030$ 圖4.9 (失敗)	接受指標 $\omega'_2 = 0.989327$ 圖5.20 (成功)	+5.81%
組合三	接受指標 $\omega_3 = 0.903026$ 圖4.10 (失敗)	接受指標 $\omega'_3 = 0.979052$ 圖5.21 (成功)	+8.42%

表 5.1: 不同方法對於表3.2案例的測試結果總整理。



模擬環境測試

在第三章中，我們已列出用於本研究方法測試的數張格點佔據地圖（圖3.4），並於第四章及第五章皆有進行測試以及討論。然而，為證明本文研究方法具有一定的可行性，我們加入不同的模擬環境地圖，並再次測試研究方法的結果。下列將先介紹模擬環境地圖的建構，再討論應用本文方法時的結果。

6.1 模擬環境建構

為了證明本研究方法的可行性，我們認為測試用的地圖不應太過單一，意即只使用一個環境來源進行地圖建構與地圖合併，因此研究方法須額外測試於不相同的環境。然而，測試用的模擬環境須具有相當的特性或依據，才足以作為研究參考，因此我們決定仿照本研究人員日常辦公環境進行模擬環境建構。本研究人員日常辦公環境為臺灣大學永齡生醫館之二樓，我們以該環境的樓層圖（圖6.1）作為參考，並於模擬軟體中進行相同環境的建構，如圖6.2。

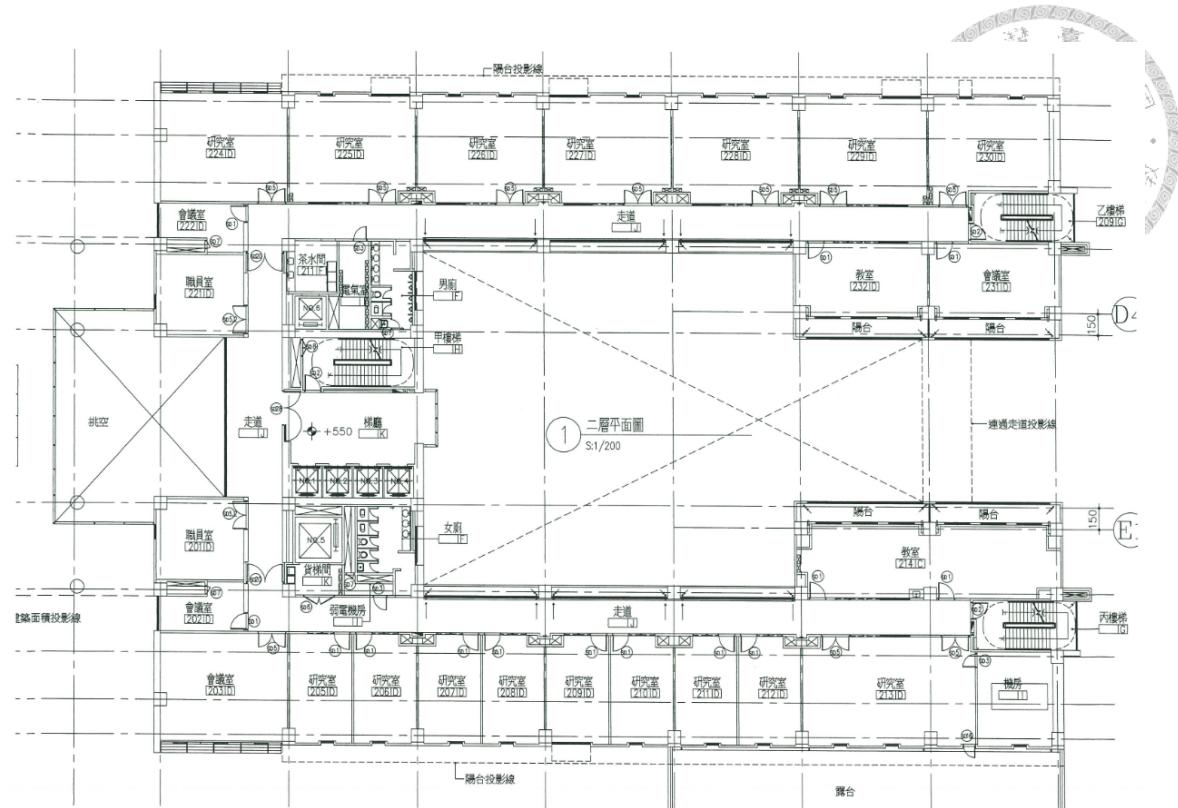


圖 6.1: 臺大永齡生醫館之二樓平面圖。

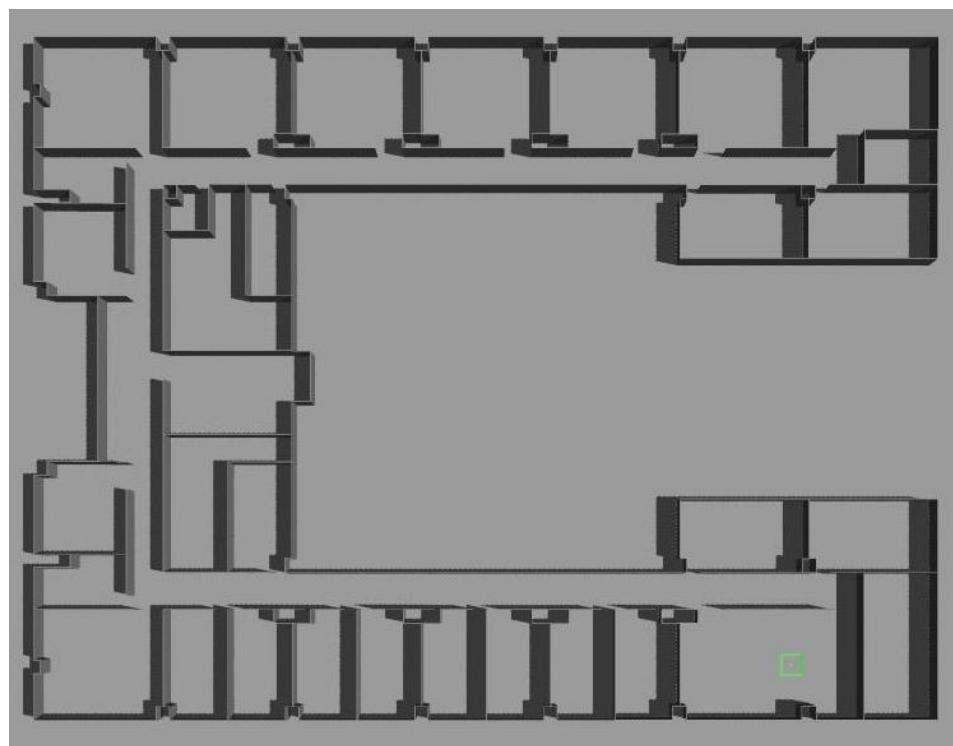


圖 6.2: 臺大永齡生醫館二樓之模擬環境。



根據永齡生醫館的模擬環境，我們將接著進行局部地圖的建構，主要將該環境分為三個區域，如圖6.3的紅、綠、藍框區域。分割區域後，便以不同初始位置及姿態，進行各自區域的地圖建構，結果如圖6.4~6.6。地圖建構完成後，將於下一節進行合併的測試討論。

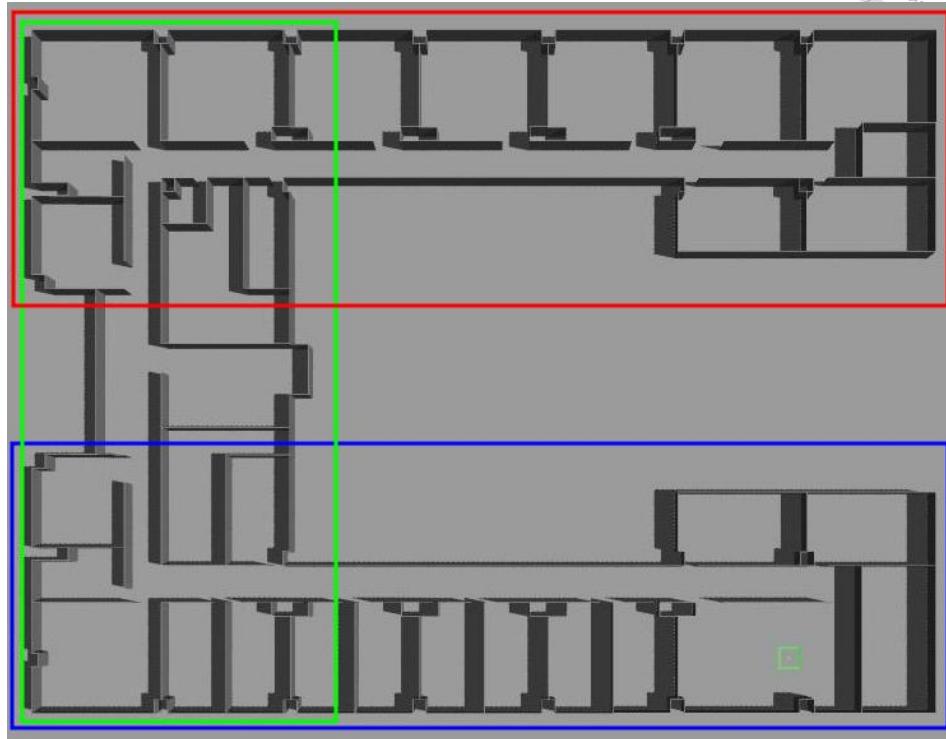


圖 6.3: 預計建構地圖之區塊。將模擬環境分為三區，並以 SLAM 方法建構局部地圖。

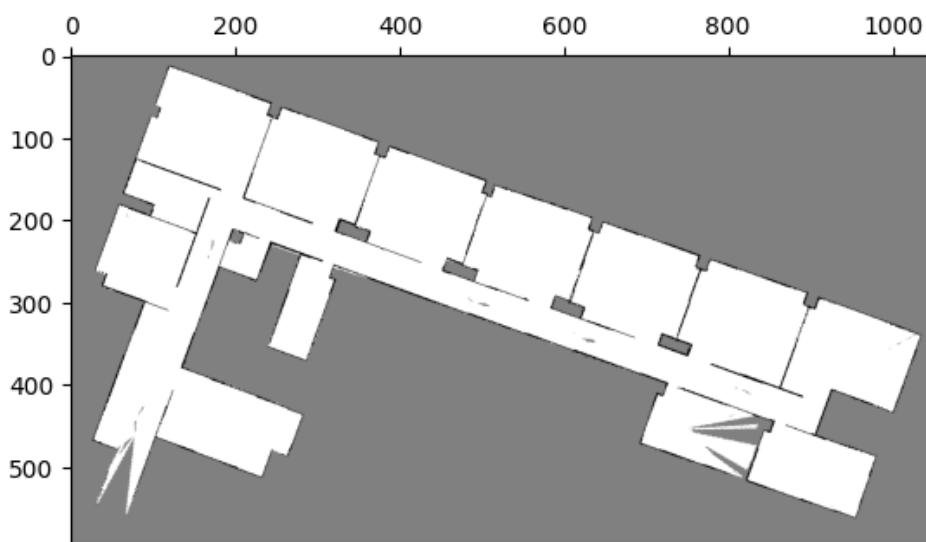


圖 6.4: 永齡生醫館模擬環境之局部地圖一。



圖 6.5: 永齡生醫館模擬環境之局部地圖二。

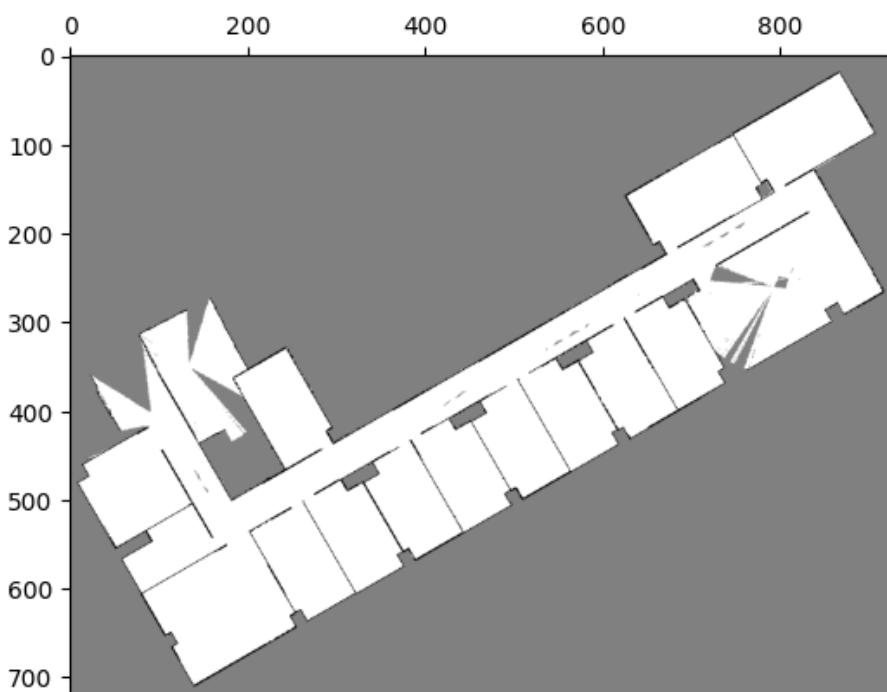


圖 6.6: 永齡生醫館模擬環境之局部地圖三。



6.2 合併測試與討論

根據前一小節的局部地圖，本節將繼續進行地圖合併的測試與討論。首先我們必須定義地圖合併的組合關係，如表6.1。

	輸入地圖一	輸入地圖二	參數關係
組合一	局部地圖一（圖6.4）	局部地圖二（圖6.5）	旋轉、平移差異
組合二	局部地圖二（圖6.5）	局部地圖三（圖6.6）	旋轉、平移差異

表 6.1: 測試案例之地圖組合

根據表6.1的組合，我們將依序進行合併測試。局部地圖一與局部地圖二的合併測試結果如圖6.7，此圖如先前第四章所介紹，我們採用其他色彩來顯示格點合併的可視化效果（綠色格點表示正確匹配，紅色格點表示錯誤匹配）。從圖中可見，大部分格點屬綠色區域，為格點的正確匹配。而屬於格點錯誤匹配的紅色區域則為少數，原因則來自影像旋轉造成的些許誤差。根據圖6.7的可視化結果，局部地圖一、二的合併結果符合模擬環境（圖6.3）的上半區域（紅、綠框）。另外，該合併情況下的接受指標為（下標1表示組合一）

$$\omega_1 = 0.984169 > \omega_{thres} = 0.95 \quad (6.1)$$

式(6.1)的接受指標 ω_1 大於先前於第四章設定的接受指標閾值 ($\omega_{thres} = 0.95$)，因此可以判斷此測試結果屬於成功合併。

接著，我們說明組合二的測試結果，如圖6.8，該圖也採用相同可視化方式呈現。圖中顯示，格點合併亦是大部分屬綠色的正確匹配，少量為紅色的錯誤匹配，與模擬環境（圖6.3）相比，也符合環境的下半區域（綠、藍框）。另外，組合二的測試結果中，接受指標為

$$\omega_2 = 0.984952 > \omega_{thres} = 0.95 \quad (6.2)$$

式(6.2)中，接受指標 ω_2 也大於閾值 ($\omega_{thres} = 0.95$)，因此該測試結果屬於成功合併。

依照表6.1的兩個組合測試結果，我們現在可利用計算出的相對變換關係，將三個局部地圖（圖6.4~6.6）進行合併，並以熵濾波處理輸出全域地圖，如圖6.9。

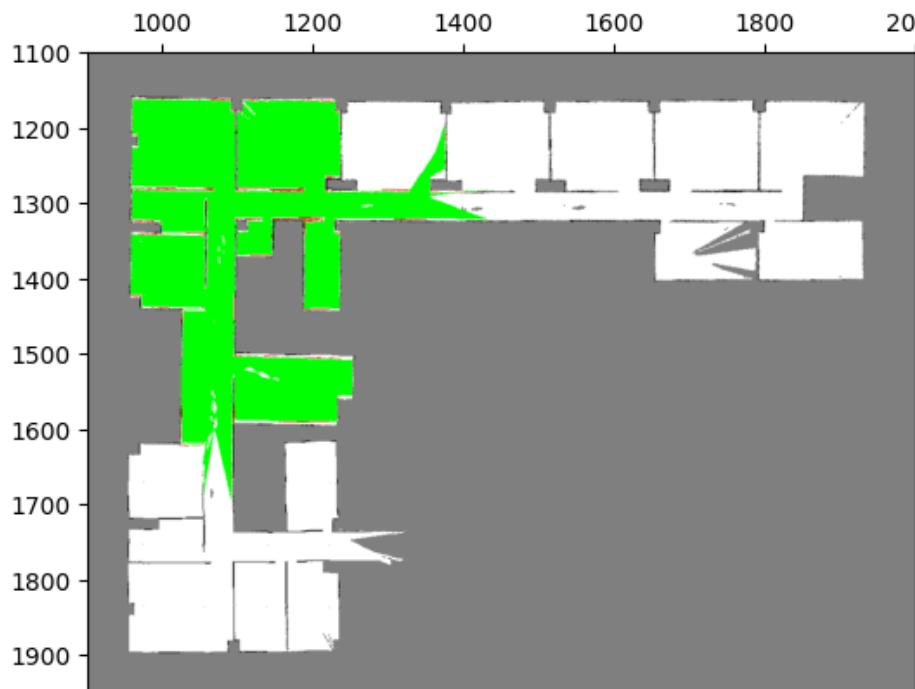
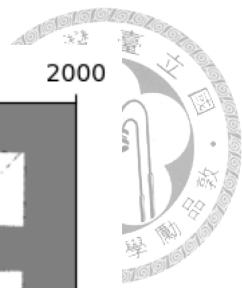


圖 6.7: 局部地圖一（圖6.4）與地圖二（圖6.5）之合併可視化影像，綠色區塊表示正確匹配的格點，紅色區塊表示錯誤匹配的格點。

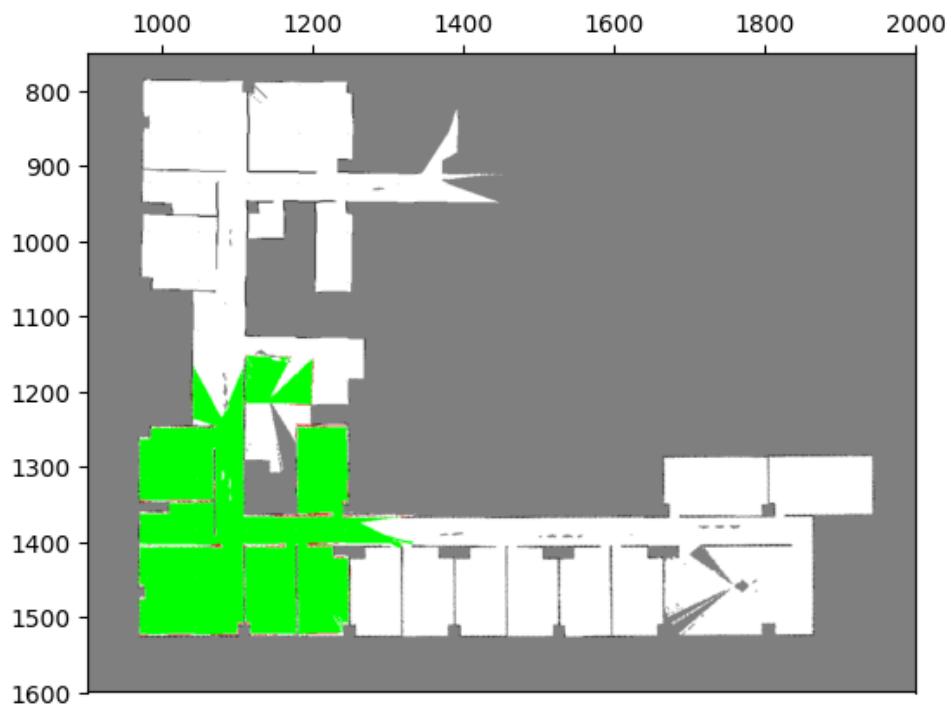


圖 6.8: 局部地圖二（圖6.5）與地圖三（圖6.6）之合併可視化影像，綠色區塊表示正確匹配的格點，紅色區塊表示錯誤匹配的格點。

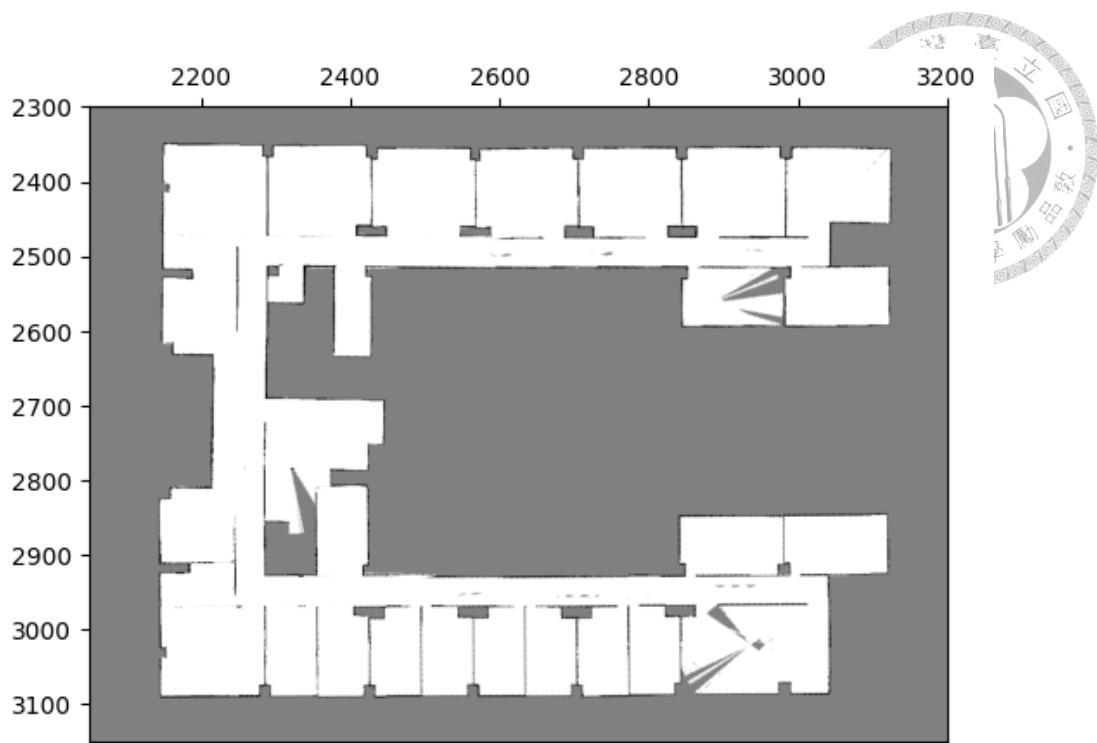


圖 6.9: 根據局部地圖兩兩相對關係（圖6.7、圖6.8），可將局部地圖全部合併為全域地圖。

根據此模擬環境的測試結果，我們深信本文研究方法足以解決格點佔據地圖的合併問題，因此具有相當程度的可行性。在下一章節中，我們將進行本文總結，並討論本研究貢獻與未來值得持續研究的方向。



總結與未來展望

在前面的章節中，我們介紹了格點佔據地圖合併問題的背景（第一章）以及相關解決方法的回顧（第二章），同時我們也針對該問題進行了詳細的問題定義（第三章）。針對本文要解決的地圖合併問題，我們首先說明本研究參考的合併方法，解釋該方法所使用的演算法，並以本文的案例進行測試（第四章）。測試案例中，我們呈現了參考方法的結果，並對合併失敗的案例進行問題分析。分析問題原因後，我們便提出本研究的處理流程來進行改善。本研究流程涵蓋數個影像處理步驟，主要針對地圖影像的佔據區域進行修正。我們發現，同樣的案例在本文方法下可取得較好的結果（第五章）。然而，為增加本文方法的可行性，我們另外測試了不同的地圖環境，同樣取得了良好的成果（第六章）。最後，我們將於此章節總結本文研究方法，並討論值得持續研究的方向。

7.1 結論

本研究旨在利用影像特徵擷取與匹配的方式，以格點佔據地圖的相同特徵計算相對變換關係，進而完成地圖合併任務。我們採用特徵擷取方法處理，不要求機器人系統於同個時間內進行任務，也不要機器人彼此進行會面，使系統更為靈活，具有彈性。另外，特徵擷取方法並非以最佳化搜尋方式進行操作，因此無需良好的初始條件，同時也避免最佳化的收斂問題。然而，在本文的說明中，若僅仰賴普通的特徵擷取方法於格點佔據地圖，可能因地圖像素值分佈較少致使可使用的特徵不多，或受到地圖建構的各種誤差影響，使得相對變換關係計算錯誤，導致地圖合併失敗。因此，本研究在特徵擷取的操作步驟前加入對地圖影像的前處理修正，可使地圖在相同環境下的主體結構線段變為一致。因結構線段變為一致，特徵擷取方法則可在不同地圖影像中，擷取相同環境位置的特徵點，並建立正確匹配關係，使後續方法可計算出正確的相對變換關係，得以成功將地圖合併。對於本文研究的案例，我們分別測試了兩種不同環境的地圖合併問題，其



中包含日常生活中可能出現的場景，經由不同環境的測試，確認了本研究方法的可行性。依照本文方法，可將僅帶有局部資訊的格點佔據地圖進行合併，並輸出為完整的全域地圖。

7.2 研究建議與未來展望

本研究提出應用影像修正與特徵匹配的格點佔據地圖合併方法，並以不同案例證實了可行性。然而，在本研究中仍有許多值得持續研究與改進的地方，以下詳列說明：

1. 嚴謹的實際環境之案例測試

本研究案例皆來自於模擬環境的地圖建構，雖然可調整模擬時的各種誤差，但與實際情況可能仍具有落差。因此，仍需實際環境的案例進行測試，才能確保研究方法的穩健性。

2. 建立參數自動化調整機制

本研究的影像處理步驟中，涉及參數的調整，例如形態學遮罩的大小。不同的地圖影像可能使用不同的操作參數，且參數可能需要反覆調整才能取得較好的效果。因此，欲應用本文方法於實際的機器人系統中，則必須建立參數自動化調整的機制，才可穩健地進行地圖合併。

3. 主體結構修正方法之修改

第五章中，我們提到主體結構修正主要以平行座標軸為原則，因此對於弧、圓圈等曲線，將造成結構上的變形。我們認為，若可建立保有曲線特徵的修正原則，不僅可排除地圖格點的不確定性及感測誤差，同時也能保有更為完整的主體結構，使特徵點匹配更為可靠，提昇合併效果。



附錄 A

格點佔據地圖之格點資訊處理

格點佔據地圖以機率區分每個格點的區域種類，假設一格點佔據地圖為 M ，其每個格點 $M(x, y)$ 皆含有兩個佔據機率 $\{p(s = 0), p(s = 1)\}$ ，前者代表該格點屬於自由區域（或稱非佔據區域）的機率，後者則代表屬於佔據區域的機率，兩者相加為一，如式 (A.1)

$$p(s = 0) + p(s = 1) = 1 \quad (\text{A.1})$$

根據式 (A.1)，可將每個格點的兩個機率化簡為機率比值 $Odd(s)$ 進行表示，化簡方式為

$$Odd(s) = \frac{p(s = 1)}{p(s = 0)} \quad (\text{A.2})$$

當感測器取得格點新的感測資訊時，可將機率比值 $Odd(s)$ 更新為

$$Odd(s|z) = \frac{p(s = 1|z)}{p(s = 0|z)} \quad (\text{A.3})$$

其中 z 表示新的感測資訊。根據貝氏定理， $p(s = 0|z)$ 與 $p(s = 1|z)$ 可等同於

$$\begin{aligned} p(s = 1|z) &= \frac{p(z|s = 1)p(s = 1)}{p(z)} \\ p(s = 0|z) &= \frac{p(z|s = 0)p(s = 0)}{p(z)} \end{aligned} \quad (\text{A.4})$$

根據式 (A.4)，式 (A.3) 可重新改寫為

$$\begin{aligned} Odd(s|z) &= \frac{p(s = 1|z)}{p(s = 0|z)} = \frac{\frac{p(z|s = 1)p(s = 1)}{p(z)}}{\frac{p(z|s = 0)p(s = 0)}{p(z)}} \\ &= \frac{p(z|s = 1)}{p(z|s = 0)} Odd(s) \end{aligned} \quad (\text{A.5})$$



將式 (A.5) 等號兩邊取對數 \log 後，得

$$\log Odd(s|z) = \underbrace{\log \frac{p(z|s=1)}{p(z|s=0)}}_{lomeas} + \log Odd(s) \quad (A.6)$$

從式 (A.6) 可看出，機率比值的更新只需將原本的 $\log Odd(s)$ 與 $\log \frac{p(z|s=1)}{p(z|s=0)}$ 相加。在格點佔據地圖中， $\log \frac{p(z|s=1)}{p(z|s=0)}$ 為一測量模型，稱為 lomeas。lomeas 為定值且具有兩個數值，即 lofree 與 looccu，分別表示當接收格點的感測資訊為自由區域與佔據區域的情況，可定義為

$$\begin{aligned} \text{lofree} &= \log \frac{p(z=0|s=1)}{p(z=0|s=0)} \\ \text{looccu} &= \log \frac{p(z=1|s=1)}{p(z=1|s=0)} \end{aligned} \quad (A.7)$$

因此僅需要設定感測器測量模型 (lomeas) 的 lofree 與 looccu 數值，則可以式 (A.6) 快速更新機率比值 $Odd(s|z)$ 。更新完畢後，依照式 (A.1) 與式 (A.2) 轉換為佔據機率 $p(s)$ ，並依照格點區域閾值輸出格點的區域種類，即

$$M(x, y) = \begin{cases} 0 & (\text{occupied}) \quad , \text{if } p(s=1) > p_{occupied} \\ 255 & (\text{free}) \quad , \text{if } p(s=1) < p_{free} \\ 128 & (\text{unknown}) \quad , \text{else} \end{cases} \quad (A.8)$$

式 (A.8) 中， $p_{occupied}$ 表示佔據閾值 (Occupied Threshold)， p_{free} 表示自由閾值 (Free Threshold)，根據兩個閾值可將格點區分為佔據 (Occupied)、自由 (Free)、未知 (Unknown) 區域，以像素值表示則分別為 0、255、128 三個整數值。

根據以上的說明，當格點佔據地圖 M 作為影像進行處理時，每一個格點皆含有「區域種類、影像像素值、佔據機率」的資訊，因此必須建立三者的相互關係，如表 A.1。

區域種類	影像像素值	佔據機率
佔據	0	$p(s=1) \in (p_{occupied}, 1]$
自由	255	$p(s=1) \in [0, p_{free})$
未知	128	$p(s=1) \in [p_{free}, p_{occupied}]$

表 A.1: 格點佔據地圖之格點資訊關係表



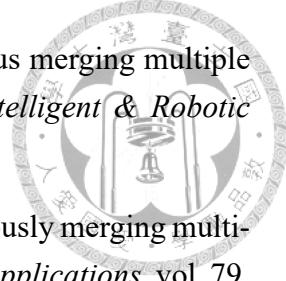
程式碼

本論文所有程式碼公佈於軟體原始碼代管服務平台 GitHub，可於本實驗室之 GitHub 網站下載 (<https://github.com/solab-ntu>)。

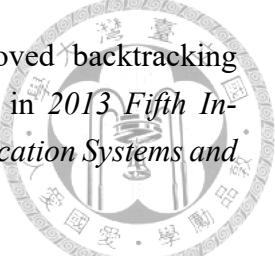


參考文獻

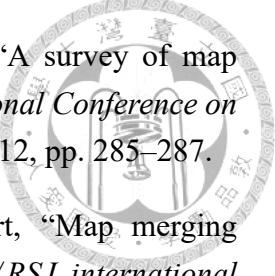
- [1] H. Umari and S. Mukhopadhyay, “Autonomous robotic exploration based on multiple rapidly-exploring randomized trees,” 09 2017, pp. 1396–1402.
- [2] S. Yu, C. Fu, A. K. Gostar, and M. Hu, “A review on map-merging methods for typical map types in multiple-ground-robot slam solutions,” *Sensors*, vol. 20, no. 23, 2020. [Online]. Available: <https://www.mdpi.com/1424-8220/20/23/6988>
- [3] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [4] P. Glira, “Iterative Closest Point,” 2016. [Online]. Available: https://youtu.be/uzOCS_gdZuM
- [5] B. Yamauchi, “A frontier-based approach for autonomous exploration,” in *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. 'Towards New Computational Principles for Robotics and Automation'*, 1997, pp. 146–151.
- [6] S. Wang, Z. Wu, and W. Zhang, “An overview of slam,” in *Proceedings of 2018 Chinese Intelligent Systems Conference: Volume I*. Springer, 2019, pp. 673–681.
- [7] A. Birk and S. Carpin, “Merging occupancy grid maps from multiple robots,” *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1384–1397, 2006.
- [8] H. Li, M. Tsukada, F. Nashashibi, and M. Parent, “Multivehicle cooperative local mapping: A methodology based on occupancy grid map merging,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2089–2100, 2014.
- [9] S. Yu, C. Fu, A. K. Gostar, and M. Hu, “A review on map-merging methods for typical map types in multiple-ground-robot slam solutions,” *Sensors*, vol. 20, no. 23, p. 6988, 2020.



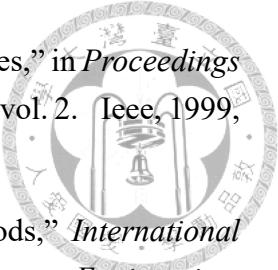
- [10] Z. Jiang, J. Zhu, Y. Li, J. Wang, Z. Li, and H. Lu, “Simultaneous merging multiple grid maps using the robust motion averaging,” *Journal of Intelligent & Robotic Systems*, vol. 94, no. 3, pp. 655–668, 2019.
- [11] Z. Jiang, J. Zhu, C. Jin, S. Xu, Y. Zhou, and S. Pang, “Simultaneously merging multi-robot grid maps at different resolutions,” *Multimedia Tools and Applications*, vol. 79, no. 21, pp. 14 553–14 572, 2020.
- [12] L. Carlone, M. K. Ng, J. Du, B. Bona, and M. Indri, “Simultaneous localization and mapping using rao-blackwellized particle filters in multi robot systems,” *Journal of Intelligent & Robotic Systems*, vol. 63, no. 2, pp. 283–307, 2011.
- [13] F. Amigoni, S. Gasparini, and M. Gini, “Building segment-based maps without pose information,” *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1340–1359, 2006.
- [14] J. Weingarten and R. Siegwart, “3d slam using planar segments,” in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 3062–3067.
- [15] H. Choset and K. Nagatani, “Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization,” *IEEE Transactions on robotics and automation*, vol. 17, no. 2, pp. 125–137, 2001.
- [16] O. Booij, B. Terwijn, Z. Zivkovic, and B. Kroese, “Navigation using an appearance based topological map,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 3927–3932.
- [17] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, “Probabilistic roadmaps for path planning in high-dimensional configuration spaces,” *IEEE transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.
- [18] S. M. LaValle and J. J. Kuffner Jr, “Randomized kinodynamic planning,” *The international journal of robotics research*, vol. 20, no. 5, pp. 378–400, 2001.
- [19] M. Keidar and G. Kaminka, “Robot exploration with fast frontier detection: Theory and experiments,” vol. 1, 06 2012, pp. 113–120.
- [20] S. M. LaValle, “Rapidly-exploring random trees : a new tool for path planning,” *The annual research report*, 1998.
- [21] G. Oriolo, M. Vendittelli, L. Freda, and G. Troso, “The srt method: randomized strategies for exploration,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 5, 2004, pp. 4688–4694 Vol.5.



- [22] H. El-Hussieny, S. F. M. Assal, and M. Abdellatif, “Improved backtracking algorithm for efficient sensor-based random tree exploration,” in *2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks*, 2013, pp. 19–24.
- [23] A. Franchi, L. Freda, G. Oriolo, and M. Vendittelli, “The sensor-based random graph method for cooperative robot exploration,” *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 2, pp. 163–175, 2009.
- [24] C. Stachniss, G. Grisetti, and W. Burgard, “Information gain-based exploration using rao-blackwellized particle filters,” 06 2005, pp. 65–72.
- [25] F. Bourgault, A. Makarenko, S. Williams, B. Grocholsky, and H. Durrant-Whyte, “Information based adaptive robotic exploration,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, 2002, pp. 540–545 vol.1.
- [26] R. Zlot, A. Stentz, M. Dias, and S. Thayer, “Multi-robot exploration controlled by a market economy,” vol. 3, 02 2002, pp. 3016 – 3023.
- [27] Z. Yan, L. Fabresse, J. Laval, and N. Bouraqadi, “Team size optimization for multi-robot exploration,” in *Simulation, Modeling, and Programming for Autonomous Robots*, D. Brugali, J. F. Broenink, T. Kroeger, and B. A. MacDonald, Eds. Cham: Springer International Publishing, 2014, pp. 438–449.
- [28] B. Yamauchi, “Frontier-based exploration using multiple robots,” in *Proceedings 1998 Second International Conference on Autonomous Agents AGENTS'98.*, 1998, pp. 47–53.
- [29] Y. Wang, A. Liang, and H. Guan, “Frontier-based multi-robot map exploration using particle swarm optimization,” in *2011 IEEE Symposium on Swarm Intelligence*, 2011, pp. 1–6.
- [30] P. Senarathne, D. Wang, Z. Wang, and Q. Chen, “Efficient frontier detection and management for robot exploration,” in *2013 IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*, 2013, pp. 114–119.
- [31] A. Franchi, L. Freda, G. Oriolo, and M. Vendittelli, “A randomized strategy for cooperative robot exploration,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 768–774.
- [32] ———, “A decentralized strategy for cooperative robot exploration,” 01 2007, p. 7.



- [33] H.-C. Lee, S.-H. Lee, T.-S. Lee, D.-J. Kim, and B.-H. Lee, “A survey of map merging techniques for cooperative-slam,” in *2012 9th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*. IEEE, 2012, pp. 285–287.
- [34] K. Konolige, D. Fox, B. Limketkai, J. Ko, and B. Stewart, “Map merging for distributed robot navigation,” in *Proceedings 2003 IEEE/RSJ international conference on intelligent robots and systems (IROS 2003) (Cat. No. 03CH37453)*, vol. 1. IEEE, 2003, pp. 212–217.
- [35] X. S. Zhou and S. I. Roumeliotis, “Multi-robot slam with unknown initial correspondence: The robot rendezvous case,” in *2006 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2006, pp. 1785–1792.
- [36] H. S. Lee and K. M. Lee, “Multi-robot slam using ceiling vision,” in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 912–917.
- [37] F. Tungadi, W. L. D. Lui, L. Kleeman, and R. Jarvis, “Robust online map merging system using laser scan matching and omnidirectional vision,” in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 7–14.
- [38] S. Carpin and G. Pillonetto, “Motion planning using adaptive random walks,” *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 129–136, 2005.
- [39] K.-F. Man, K. S. Tang, and S. Kwong, *Genetic algorithms: concepts and designs*. Springer Science & Business Media, 2001.
- [40] J. Park, A. J. Sinclair, R. E. Sherrill, E. A. Doucette, and J. W. Curtis, “Map merging of rotated, corrupted, and different scale maps using rectangular features,” in *Proceedings of IEEE/ION PLANS 2016*, 2016, pp. 535–543.
- [41] S. Carpin, “Fast and accurate map merging for multi-robot systems,” *Autonomous robots*, vol. 25, no. 3, pp. 305–316, 2008.
- [42] R. O. Duda and P. E. Hart, “Use of the hough transformation to detect lines and curves in pictures,” *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.
- [43] V. T. Ferrão, C. D. N. Vinhal, and G. da Cruz, “An occupancy grid map merging algorithm invariant to scale, rotation and translation,” in *2017 Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE, 2017, pp. 246–251.



- [44] D. G. Lowe, “Object recognition from local scale-invariant features,” in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.
- [45] S. Arya, “A review on image stitching and its different methods,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 5, pp. 299–303, 2015.
- [46] Z. Wang and Z. Yang, “Review on image-stitching techniques,” *Multimedia Systems*, pp. 1–18, 2020.
- [47] S. Saeedi, L. Paull, M. Trentini, M. Seto, and H. Li, “Group mapping: A topological approach to map merging for multiple robots,” *IEEE Robotics & Automation Magazine*, vol. 21, no. 2, pp. 60–72, 2014.
- [48] D. Fox, W. Burgard, S. Thrun, and A. B. Cremers, “Position estimation for mobile robots in dynamic environments,” *AAAI/IAAI*, vol. 1998, pp. 983–988, 1998.
- [49] J. J. Koenderink, “The structure of images,” *Biological cybernetics*, vol. 50, no. 5, pp. 363–370, 1984.
- [50] T. Lindeberg, “Scale-space theory: A basic tool for analyzing structures at different scales,” *Journal of applied statistics*, vol. 21, no. 1-2, pp. 225–270, 1994.
- [51] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [52] S. Helgason and S. Helgason, *The radon transform*. Springer, 1980, vol. 2.
- [53] P. J. Besl and N. D. McKay, “Method for registration of 3-d shapes,” in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586–606.
- [54] E. Huang, TS and Blostein, SD and Margerum, “Least-squares estimation of motion parameters from 3-D point correspondences,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1986, pp. 112—115.