

國立臺東大學資訊管理學系

碩士論文

Department of Information Science and Management Systems

National Taitung University

Master Thesis

地理標記語言之資料隱藏技術

A Study of Data Hiding Techniques for
Geography Markup Language

研究生：林延錄

Yan-Lu Lin

指導教授：王聖銘 博士

Advisor: Sheng-Ming Wang, Ph.D.

中華民國九十九年七月

July, 2010

國立臺東大學
學位論文考試委員審定書
系所別： 資訊管理學系

本班 林延錄 君

所提之論文地理標記語言之資料隱藏技術

業經本委員會通過合於 碩士學位論文 條件

論文學位考試委員會：

許志堅

(學位考試委員會主席)

郭晏升

王聖銘

(指導教授)

論文學位考試日期：99 年 7 月 13 日

國立臺東大學

附註：1. 本表一式二份經學位考試委員會簽後，正本送交系所辦公室及註冊組或進修部存查。

2. 本表為日夜學制通用，請依個人學制分送教務處或進修部辦理。

博碩士論文電子檔案上網授權書

(提供授權人裝訂於紙本論文書名頁之次頁用)

本授權書所授權之論文為授權人在 國立臺東大學 資訊管理學系碩士班 _____組 98 學年度第 二 學期取得 碩士 學位之論文。

論文題目：地理標記語言之資料隱藏技術

指導教授：王聖銘 博士

茲同意將授權人擁有著作權之上列論文全文(含摘要)，非專屬、無償授權國家圖書館及本人畢業學校圖書館，不限地域、時間與次數，以微縮、光碟或其他各種數位化方式將上列論文重製，並得將數位化之上列論文及論文電子檔以上載網路方式，提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印。

- 讀者基非營利性質之線上檢索、閱覽、下載或列印上列論文，應依著作權法相關規定辦理。

授權人：林延錄

簽 名： 林 延 錄

中華民國 99 年 07 月 23 日

博碩士論文授權書

本授權書所授權之論文為本人在 國立臺東大學 資訊管理 系(所)
組 98 學年度第 2 學期取得 碩 士學位之論文。
論文名稱：地理標記語言之資料隱藏技術

本人具有著作財產權之論文全文資料，授權予下列單位：

同意	不同意	單 位
☒	☐	國家圖書館
☒	☐	本人畢業學校圖書館
☒	☐	與本人畢業學校圖書館簽訂合作協議之資料庫業者

得不限地域、時間與次數以微縮、光碟或其他各種數位化方式重製後散布發行或
上載網站，藉由網路傳輸，提供讀者基於個人非營利性質之線上檢索、閱覽、下
載或列印。

☒ 同意 ☐ 不同意 本人畢業學校圖書館基於學術傳播之目的，在上述範圍內得再授
權第三人進行資料重製。

本論文為本人向經濟部智慧財產局申請專利(未申請者本條款請不予理會)的附件之一，申請
文號為：_____，請將全文資料延後半年再公開。

公開時程

立即公開	一年後公開	二年後公開	三年後公開
		☒	

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行
權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。上述同意與
不同意之欄位若未勾選，本人同意視同授權。

指導教授姓名：王聖銘 (親筆簽名)

研究生簽名：林延銘 (親筆正楷)

學 號：9701302 (務必填寫)

日 期：中華民國 99 年 7 月 23 日

1.本授權書(得自 <http://www.lib.ntu.edu.tw/theses/> 下載)請以黑筆撰寫並影印裝訂於書名頁之次頁。

2.依據 91 學年度第一學期一次教務會議決議:研究生畢業論文「至少需授權學校圖書館數位化，並至遲
於三年後上載網路供各界使用及校內瀏覽。」

授權書版本:2008/05/29

致謝

研究所兩年的時光過得飛快，在整個求學期間與論文撰寫的過程中，最高興的不單單只是論文的完成，更包含在台東求學兩年的過程。這期間所遇到的人事物、以及學習到的知識與經驗，都將成為我美好的回憶與日後受用不盡的寶藏。

首先感謝恩師王聖銘教授與陳富美教授這兩年來在生活、課業與專案上悉心指導與啟發，才能讓我在這兩年中不斷有所進步，更在王老師優越的專業能力、洞悉問題核心的洞察力與豐富的實務經驗指導下，讓我如入寶山，滿手而歸，使本論文得以順利完成，在此致上最高的謝意與敬意。

同時也要感謝資管系所有老師與行政助理平時的指導與照顧，不但讓我收穫良多，更讓離鄉背井的我倍感溫暖。另外也要感謝口試委員許志堅教授與邱景升教授給予我寶貴的建議及指導，使得本論文內容更加豐富且完整。

另外也要感謝紹永學長幫忙校稿，使我的論文更加完整。還有感謝龍富、聖熙、文翔、牧隴、冠廷、慧文、千秦與弘侑，除了學業上的相互扶持及討論外，更彼此相互鼓勵與關心，使我能安心、專心於論文的研究。也感謝學弟名哲、昀峰、瀚品、啟章、政鴻與俊德，幫忙處理實驗室的大小事務。

最後感謝家人的體諒與關懷，並在背後默默的支持我，讓我無後顧之憂。再次鄭重向所有老師、親朋好友以及曾經幫助過我的人，致上本人最高的謝意與敬意，謝謝。

林延錄 敬上

國立台東大學資訊管理學系碩士班

中華民國 99 年 7 月

摘要

本研究主要在研究如何將不同類型的資料隱藏(Data Hiding)技術導入地理標記語言(Geography Markup Language, GML)的空間資料格式中，分析與探討 GML 空間資料格式在導入資料隱藏技術後的成效，並提出改善可擴展之可逆式向量地圖資料隱藏演算法的方法。本研究中提出了無失真資訊隱藏法、PCA 排序法、投影排序法與直接投影法等資訊隱藏處理方法，並藉此評估提升地理標記語言之資料隱藏技術在地理資訊系統(Geographic Information System, GIS)的應用性。

本研究架構在既有的可擴展之可逆式向量地圖資料隱藏演算法與無失真資料隱藏演算法上，並導入 GML 空間資料格式做應用評估。其目標除了求得可嵌入的訊息量最大化之外，也希望能根據不同的應用目的、秘密訊息的容量大小與 GML 地圖的精度條件，規劃其所對應的資料隱藏演算法、前置處理流程與嵌入資料的位置。是故，本研究針對可擴展之可逆式向量地圖資料隱藏技術提出三種不同的前置處理流程：PCA 排序法、投影排序法與直接投影法，同時嵌入資料的位置包含幾何圖形中的頂點、曲線與表面，並且將資料嵌入其所屬的圖徵當中。其結果不但可以提升 GML 檔案中資料的隱密性、完整性與安全性，並且增加地理標記語言之資料隱藏技術的可應用性，更可做為 GML 檔案的認證機制。

經由可嵌入訊息量、不可察覺性、可逆性、以及資料嵌入與擷取所需執行時間等評估項目的比較，本研究的結果比直接應用可逆式向量地圖資料隱藏及無失真資料隱藏法於傳統向量地圖資料格式的結果有更高的可嵌入資料量，同時所需執行時間也較短。但本研究所提出的直接投影資料處理法在不可察覺性及可逆性的評估項目中表現較差，是故，該方法只能適合於少量資料嵌入需求的應用。而本研究也顯示，應用 GML 空間資料格式於向量地圖資訊隱藏的應用中，可降低原先陳富美學者所提出之可逆式向量地圖資料隱藏演算法中資料前處理需先經 PCA 轉換的繁複程序，增加其應用的效率。本研究的結果可做為應用資料隱藏

技術於 GML 空間資料格式而達到空間資訊系統創新應用的基礎。針對未來研究，建議可進一步探討 GML 空間資料格式中有關拓撲相關的特性，以及 GML 各種標籤元素的應用。

關鍵字：資料隱藏、地理標記語言、空間資訊系統、向量地圖、資料擷取



Abstract

This research is focused on analyzing and evaluating the results of applying different data hiding technologies to Geography Markup Language (GML) spatial data format and to propose the possible modification of existing extensive reversible data hiding algorithms. The data hiding processing methods, which include distortion free method, PCA sorting method, projection sorting method, and direct projection method, are proposed and evaluated in this research. The results of this research are also used for the evaluation of how data hiding technologies can be used for the enhancements of geographic information system (GIS) application based on GML spatial data format.

The existing extensive reversible data hiding algorithms to vector maps proposed by Chen and the distortion free data hiding algorithms proposed Bogomjakov are used in this research. The main purpose of this research is not only to maximize the capacity of embedding messages but also to construct the corresponding data hiding algorithms, data processing methods and data embedding position based on different application objectives, size of the secret message, and the precision of maps. Thus, in addition to the distortion free methods, there are three different processing methods for extensive reversible data hiding algorithms for vector maps: PCA sorting method, projection sorting method, and direct projection method; are applied to hide data in the points, curve (lines), and surface(polygons) data type in GML respectively. The results show that the mechanisms and methods proposed in this research not only can strengthen the privacy, integrity and security of the data hidden in GML but can also enhance the application of applying data hiding technology for the GML. Moreover, it can be used for GML authentication.

Case studies are implemented for the comparison on data hiding capacity, invisibility effects, reversibility, and execution time for data embedding and extraction are made for evaluating the results in this research to the results that applying extensive reversible data hiding algorithm and distortion free data hiding algorithm to traditional vector data format. The results show that the mechanism proposed in this research is superior in data hiding capacity and executing time. However, the performance of the direct projection processing method proposed in this research is worse than the other methods. It indicate the method can only been used by the applications that need low data hiding capacity. As conclude by the case studies, the application of data hiding technologies in GML can reduce the complicated PCA pre-processing procedure in the algorithm proposed by Chen and promote the data hiding efficiency. The mechanism the apply data hiding technologies in GML proposed in this research can be used as the base the innovation applications of spatial information system. Moreover, the topological features and related Tag elements in GML can be used for the applications of data hiding in future studies.

Keywords: Data Hiding, Geography Markup Language, Geographic Information System, Vector Maps, Data Extraction.

目錄

摘要	I
Abstract.....	III
目錄	V
圖目錄	VI
表目錄	VIII
第一章 緒論.....	1
1.1 研究動機.....	1
1.3 研究流程.....	5
第二章 文獻探討.....	8
2.1 GML 介紹	8
2.2 資料隱藏技術及應用	11
2.2.1 向量地圖資料隱藏演算法	15
2.2.2 可逆式向量地圖資訊隱藏系統.....	19
2.2.3 可擴展之可逆式向量地圖資料隱藏演算法	21
2.2.4 無失真資料隱藏演算法	23
第三章 研究方法.....	25
3.1 GML 空間資料格式.....	26
3.1.1 GML 頂點格式.....	28
3.1.2 GML 曲線格式.....	29
3.1.3 GML 表面格式.....	30
3.2 資料嵌入流程	33
3.2.1 頂點之圖徵類型.....	35
3.2.2 曲線之圖徵類型.....	38
3.2.3 表面之圖徵類型.....	40
3.3 資料擷取流程	44
第四章 研究結果與分析	46
4.1 實驗設計	46
4.2 系統介紹.....	48
4.2 可嵌入訊息量(Capacity)評估	52
4.3 不可察覺性(Imperceptibility)評估	54
4.4 可逆性(Reversibility)評估.....	56
4.5 執行時間(Execute Time)評估	58
第五章 結論與未來研究建議.....	60
5.1 結論.....	60
5.2 未來研究建議	62
參考文獻	63

圖目錄

圖 1-1：研究流程.....	6
圖 2-1：GML 文件組成.....	10
圖 2-2：GML 文件說明.....	10
圖 2-3：資料隱藏示意圖.....	11
圖 2-4：資訊隱藏的分類.....	12
圖 2-5：資訊隱藏應用目的.....	13
圖 2-6：資訊隱藏與加密技術的比較.....	14
圖 2-7：奇/偶數序列.....	16
圖 2-8：定義區間.....	17
圖 2-9：嵌入資訊「0」.....	18
圖 2-10：嵌入資訊「1」.....	18
圖 2-11：頂點位於狀態「0」之嵌入動作.....	20
圖 2-12：頂點位於狀態「1」之嵌入動作.....	20
圖 2-13：分割變數 s 與切割變數 c 與子區間個數 v 之關係.....	21
圖 2-14：嵌入 2 位元資訊.....	22
圖 2-15：置換端點排列順序嵌入訊息.....	23
圖 2-16：無失真資料隱藏演算法.....	24
圖 3-1：研究流程.....	25
圖 3-2：圖徵之拓撲與幾何屬性.....	27
圖 3-3：頂點語法示意圖.....	28
圖 3-4：曲線語法示意圖.....	29
圖 3-5：表面語法示意圖.....	30
圖 3-6：資料嵌入流程.....	33
圖 3-7：嵌入資料之圖徵類型分類.....	34
圖 3-8：台灣五直轄市所在位置.....	36
圖 3-9：讀取頂點標籤內容演算法.....	37
圖 3-10：讀取曲線標籤內容演算法.....	39
圖 3-11：空間資料儲存比較.....	41
圖 3-12：讀取表面標籤內容演算法.....	42
圖 3-13：資料擷取流程.....	44
圖 4-1：台灣鄉鎮地圖.....	47
圖 4-2：系統介面.....	49
圖 4-3：資料嵌入介面.....	49
圖 4-4：資料擷取介面.....	50
圖 4-5：軟體架構.....	50

圖 4-6：系統特色.....	51
圖 4-7：可嵌入訊息量.....	53
圖 4-8：嵌入資料之 RMSE Ratio.....	55
圖 4-9：座標值與其鄰近座標值之距離.....	55
圖 4-10：還原資料之 RMSE Ratio.....	57
圖 4-11：分割變數影響座標值差異.....	57
圖 4-12：執行時間.....	59



表目錄

表 2-1：子分割區間.....	19
表 3-1：地理標記語言頂點屬性語法與說明	28
表 3-2：地理標記語言曲線屬性語法與說明	29
表 3-3：地理標記語言表面屬性語法與說明	31
表 4-1：GML 檔案基本資料	46
表 4-2：可嵌入訊息量.....	52
表 4-3：嵌入資料之 RMSE Ratio.....	54
表 4-4：還原資料之 RMSE Ratio.....	56
表 4-5：執行時間.....	58
表 5-1：研究結果比較.....	60



第一章 緒論

1.1 研究動機

本研究主要在研究如何將不同類型的資料隱藏技術導入地理標記語言 (Geography Markup Language, GML) 的空間資料格式中，根據不同應用目的，選擇對應的資料隱藏演算法，並提出不同的前置處理流程與可選擇嵌入資料的位置，藉此提升 GML 檔案中資料的隱密性、完整性與安全性，同時比較不同資料隱藏技術導入 GML 空間資料格式後與原研究結果的差異，分析與探討 GML 空間資料格式在資料隱藏技術導入後的成效，提出改善資料隱藏技術應用的方法與 GML 地理資訊系統 (Geographic Information System, GIS) 上的創新應用。

因全球資訊化的腳步越來越快，人們可以輕易地從網路上竊取資料或重製散播 (Niu, Shao, & Wang, 2006)，導致著作權 (Copyright) 以及資訊安全 (Information Security) 等相關問題產生，因此許多專家學者展開資訊隱藏 (Information Hiding) 的相關研究 (Chang & Tseng, 2009; Hong, Chen, & Shiu, 2008)。目前資料隱藏技術在地理資訊上的相關研究非常少，主要針對向量地圖做研究 (陳凱崑，2007；劉秋良，2008；陳富美 2009)，而向量地圖主要的優點在於以向量的方式來表現圖徵，不用侷限於特定的螢幕解析度 (Resolution)，但缺點為詮釋與屬性資料須分開儲存，同時一個檔案僅可表示一種圖徵類型，因此需要多個檔案來描述真實現象，且傳遞時需一併傳送其詮釋資料，容易造成外部資料與內部資料不一致的情況。

Google 於 2008 年提出的地理資訊資料格式：鎖孔標記語言 (Keyhole Markup Language, KML) 是一種基於 XML 語法的檔案格式，可用於塑模與儲存例如點、

線、圖像、多角形與模型等地理特性，用以顯示在 Google Map、Google Earth 與其他 GIS 上，使 GIS 不再只是單純的描繪地形或真實現象，透過結合不同的應用程式，便可提供不同形式的服務，應用於不同領域(X. T. Wang, Shao, Xu, & Niu, 2007)，例如實境導覽、教學或防災等應用(S.-M. Wang & Chen, 2008; 王聖銘 & 陳富美, 2009)。但目前 KML 的應用大多架構在 Google Map 與 Google Earth 上，並未廣泛應用於不同的 GIS 中(何燦群，2010)。

因此本研究將探討一個地理資訊的標準資料格式：地理標記語言(Geography Markup Language, GML)，是由 OGC(Open Geospatial Consortium)所認定的空間資訊編碼規範，同時也是 ISO 國際標準，是一種基於 XML 技術發展的開放式標記語言，透過標籤(Tag)來描述真實世界中的現象，以屬性的值來描述圖徵性質。其中屬性的值可以是整數、浮點數、幾何或圖徵等型態，同時因具備自訂標籤的特性，提供製圖者更彈性的方式來描繪地理特徵。與一般 VML(Vector Markup Language)和 SVG(Scalable Vector Graphics)地理資訊檔案格式不同，GML 將地理資訊與其屬性封裝在一起，同時提供分散式的地理資料儲存。

而如何保護 GML 中的地理資訊與其詮釋資料(Meta-Data)在進行傳遞時不易受到懷疑、甚至是竊取或竄改，確保資訊的隱密性和完整性，是一個相當重要且值得探討的研究領域。有別於一般的金鑰加密技術，資料隱藏提供另外一種保護秘密訊息的機制。目前在向量地圖相關的資料隱藏演算法中(S.-M. Wang, Chen, Chen, & Yang, 2009；陳凱威、王聖銘、張聖杰等，2007；陳凱威、王聖銘、鄭友銘、王宗銘，2007)，進行前置處理時，均利用主成分分析(Principal Component Analysis, PCA)將端點(Vertex)座標值轉換，其優點在於使用簡單，並且可提升安全性與強韌性，但座標值一經轉換，其圖徵的精度就會降低，可應用的領域有限。

所以為了提升 GML 資料的安全性與 GML 資料隱藏技術之可應用性，同時改善可擴展之可逆式向量地圖資料隱藏演算法造成精度降低的問題，因此本研究主要研究如何將既有的可擴展之可逆式向量地圖與無失真資料隱藏技術導入 GML 空間資料格式中，透過標籤語法的特性，直接嵌入與擷取所需的資料，根據不同應用目的選擇對應的資料隱藏演算法、前置處理流程與嵌入資料的位置，藉此提升 GML 中資料的隱密性、完整性與安全性，同時增加 GML 資料隱藏技術的可應用性。並比較資料隱藏技術導入 GML 空間資料格式後與可擴展之可逆式向量地圖資料隱藏演算法之研究結果的差異，分析與探討 GML 空間資料格式在資料隱藏技術導入後的成效，提出改善可擴展之可逆式向量地圖資料隱藏演算法應用的方法與 GML 資料隱藏技術在 GIS 上的創新應用。

1.2 研究目的

本研究提出 GML 空間資料格式之資料隱藏技術應用，並評估不同資料隱藏技術導入 GML 空間資料格式之成效。GML 空間資料格式之資料隱藏技術，透過標籤語法的特性，分析圖徵的特性，根據不同應用目的選擇適用的資料隱藏演算法、前置處理流程與嵌入資料的位置，藉此提升 GML 資料隱藏技術之可應用性，同時增加 GML 資料的隱密性、完整性與安全性，並且提出改善資料隱藏技術應用的方法。其中嵌入資料的位置包括不同的圖徵類型與不同的圖徵，而嵌入的資料包含 GML 中圖徵的屬性資料與詮釋資料，將之直接嵌入 GML 空間資料格式中，藉此避開竊取者的注意力，同時降低 GML 的檔案容量，不但達到保護秘密訊息的目的，更提出創新的 GML 資料隱藏技術應用。因此以 GML 空間資料格式做為掩護媒體，除了了解 GML 檔案的特性，並實際將可擴展之可逆式向量地圖資料隱藏演算法(陳富美，2009)與無失真資料隱藏演算法(Bogomjakov, et al., 2008)導入 GML 空間資料格式之外，本研究的目標如下：

(1) 增加 GML 資料的安全性

將可擴展之可逆式向量地圖與無失真資料隱藏技術導入 GML 空間資料格式中，藉此隱藏 GML 中圖徵的屬性資料與詮釋資料，並據根據不同應用目的選擇資料隱藏演算法、前置處理流程與嵌入資料的位置，提升 GML 資料的在各方面的安全性。

(2) 提升 GML 資料隱藏技術的可應用性

利用標籤語法的特性，直接擷取所需的端點座標值，根據不同應用目的選擇適用的資料隱藏演算法，並針對可擴展之可逆式向量地圖資料隱藏技術提出不同的前置處理流程：PCA 排序法、投影排序法與直接投影法，藉此提升可嵌入的訊息量與降低圖形的變形量，以應用於不同精度條件與比例尺(Scale)需求的 GIS 中。

(3) 評估資料隱藏導入 GML 空間資料格式的結果

分別評估可擴展之可逆式向量地圖與無失真資料隱藏演算法導入 GML 空間資料格式後的結果，並比較不同前置處理流程與不同嵌入資料的位置產生的影響，探討 GML 空間資料格式做為掩護媒體的成效。

(4) 透過 GML 特性改善原資料隱藏技術之結果

透過 GML 標籤語法的特性，根據不同的前置處理流程與嵌入資料的位置，並利用分段處理的方式，藉此增加可嵌入的訊息量，同時不增加圖形的變形量。

故本研究針對 GML 空間資料格式導入可擴展之可逆式向量地圖與無失真資料隱藏演算法，並滿足上述的目標。在對 GML 空間資料格式嵌入高訊息容量的同時，根據適當的前置處理流程與嵌入資料的位置，透過分段處理的方式，減少失真並降低秘密訊息被竊取的機率。另外比較與評估可擴展之可逆式向量地圖與無失真資料隱藏演算法導入 GML 空間資料格式之結果，探討可應用的領域，提出改善資料隱藏技術應用的方法與創新的 GML 資料隱藏技術應用。

1.3 研究流程

本研究以 GML 空間資料格式為掩護媒體，實際導入可擴展之可逆式向量地圖與無失真資料隱藏技術，提出改善資料隱藏技術應用的方法與 GML 在 GIS 上的創新應用，流程如下圖 1-1 所示，依序在下列章節詳細說明。

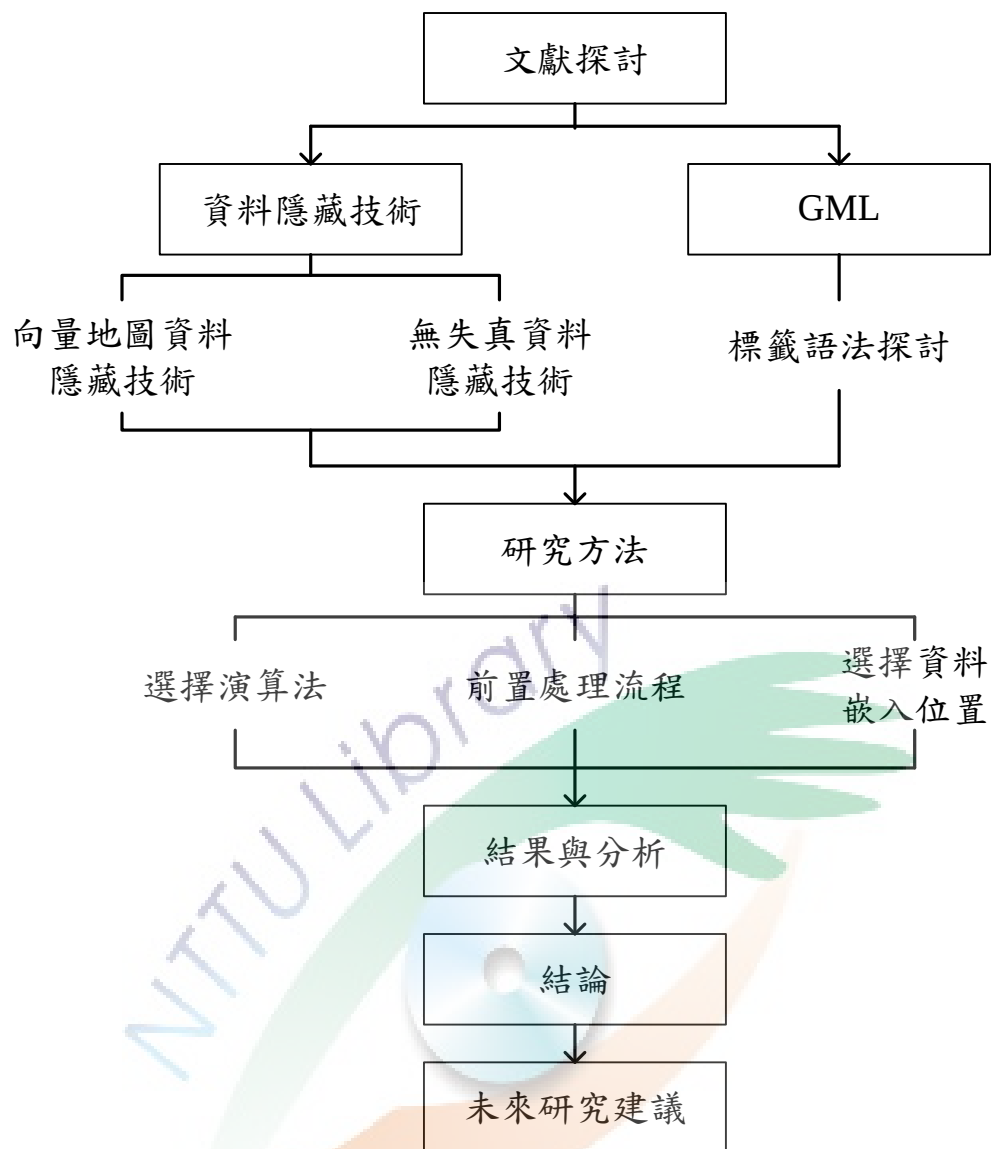


圖 1-1：研究流程

首先第二章文獻探討說明本研究針對的檔案格式：GML，和其他資料隱藏相關研究。由於目前沒有 GML 資料隱藏技術之相關研究，但 GML 的座標系統與向量地圖相同，因此將向量地圖的資料隱藏技術導入 GML 空間資料格式，同時為了比較不同資料隱藏演算法導入之成效，因此將無失真資料隱藏演算法導入，比較其特性與成效。

第三章研究方法詳細說明本研究提出的導入資料隱藏技術的方法，透過詳細了解 GML 空間資料格式，擷取出可用的屬性資料，根據不同應用目的與地圖精度條件選擇對應的資料隱藏演算法、前置處理流程與嵌入資料的位置，其中針對可擴展之可逆式向量地圖資料隱藏演算法提出三種前置處理流程：PCA 排序法、投影排序法與直接投影法，並且詳細說明資料嵌入與擷取流程的各個步驟。

第四章研究結果與分析說明本研究的結果，根據可嵌入訊息量、不可察覺性、可逆性與執行時間這四項指標來評估不同資料隱藏技術導入 GML 空間資料格式之結果，並比較與可擴展之可逆式向量地圖資料隱藏技術之研究結果的差異，討論本研究提出的 GML 空間資料格式之資料隱藏技術在各方面的表現成果。

最後第五章結論與未來研究建議，針對第三章所提出的研究方法與第四章的研究結果做綜合的比較與總結，陳述本研究之研究貢獻，並提出後續研究可改進或加強的部分，討論未來可研究之方向。

第二章 文獻探討

在本章中，首先在 2.1 節說明本研究針對的掩護媒體：地理標記語言 (Geography Markup Language, GML)。其次在 2.2 節說明什麼是資料隱藏，並釐清資訊隱藏、資料隱藏與浮水印之間的關係與相異之處，由於 GML 目前沒有資料隱藏的相關文獻，故研究相同性質的向量地圖資料隱藏技術來探討在 GML 空間資料格式上可應用的資料隱藏演算法，最後探討無失真的資料隱藏演算法，藉此比較不同資料隱藏技術在 GML 空間資料格式上的差異。

2.1 GML 介紹

地形圖是一種重要的基本圖形資料，應用廣泛，如都市規劃、旅遊導覽等等，然而不同的製圖單位所製作的地形圖，其資料格式不盡相同，使用地形圖的單位也因其需求而使用不同的 GIS，導致地形圖必須轉換檔案格式，造成檔案因轉換而產生資訊的漏失，影響地形圖的正確性。因此追求地理空間資料的共通共享，一直是 GIS 使用者共同的理想(陳勇仁，2007)。

地理標記語言(Geography Markup Language, GML)是 OGC(Open Geospatial Consortium)所認定的空間資訊編碼規範，同時也是 ISO 國際標準，是一個公開且供應商中立(Vendor-Neutral)的標準，不被任何一家 GIS 軟體廠商所限制。GML 是一種基於可擴展標記語言(eXtensible Markup Language, XML)技術發展的開放式標記語言，用以描述生活環境中之地理物件，以圖徵(Feature)為基礎，記錄地理物件之空間位置或相關屬性內容，並且達成下列目的(鄧東波，2006)：

- (1) 透過網際網路分享和交換已編碼的地理資訊。
- (2) 表達不同領域的地理語彙論述。

(3) 表達地理性網絡服務訊息元素。

現行版本 GML3.2.1 於 2007 發佈，已成為 ISO19136 標準，納入 TC/211 系列標準之一(林信助，2008)。該版本加強了地理空間資料之表達所需的型態與方式，支援多種物件(Objects)以描述地理資訊之相位關係、幾何性質、座標參考系統、時間屬性值、多種比例尺、詮釋資料、網格(Grid)資料、和對地形及區域做視覺化處理所需的預設樣式。

對於空間資訊而言，GML 是一個標準的詮釋語言(Meta-Language)，可以完整表達空間資訊的複雜性。以往以關連式資料庫為主的 GIS，雖然在分析及資料處理上能力強大，但因為地理資料龐雜，導致需要更多的詮釋資料來說明地理資料，而 GML 本身不但是詮釋語言，而且核心綱目提供大量地理語彙，因此可以用來建構地理資料的語義(Semantic)和邏輯(Logic)模式，此外 GML 還有具有下列特性，使得地理資料易於整合網路資源(鄧東波，2006)：

- (1) GML 資料可以直接被使用者讀取和了解。
- (2) GML 可實現分散式空間資料集(Dataset)並且連結這些資料集，降低資料成本。
- (3) GML 資料可簡單混合非空間的資料，例如文字或影像等。

GML 文件是由三個部份所組成，分別為 OGC 所公佈的 GML 綱目(Schema)、不同應用領域根據其需求所定義的 GML 應用綱目(Application schema)和來自資料生產單位並根據 GML 應用綱目(Application schema)之架構產生之文件(GML Instance Data)，該文件並提供給使用者，如下圖 2-1、圖 2-2 所示(鄧東波，2004)。

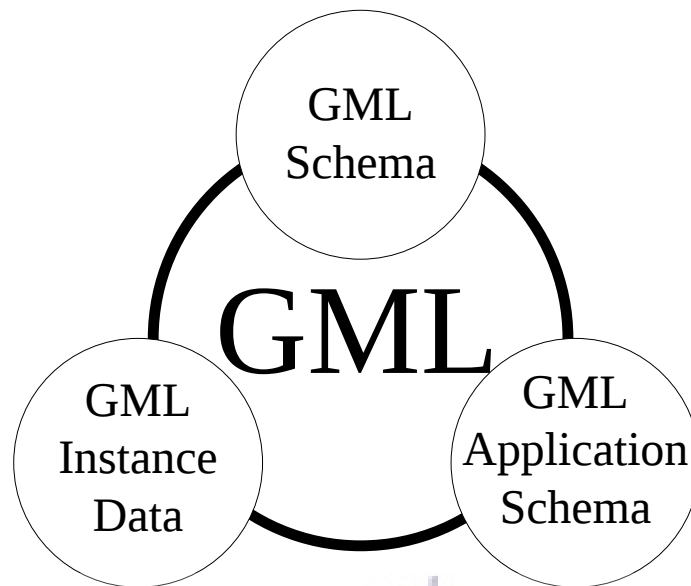


圖 2-1：GML 文件組成
資料來源：(鄧東波，2004)

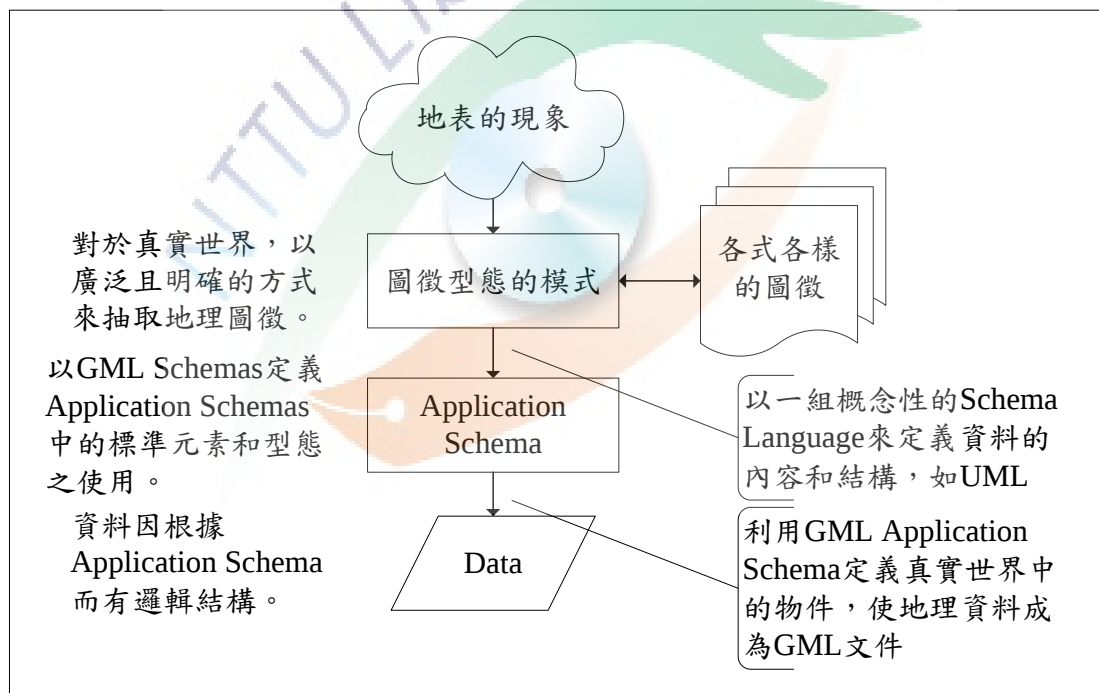


圖 2-2：GML 文件說明
資料來源：(鄧東波，2004)

其中 GML 綱目是平行的，且不針對任何一個應用領域，提供不同應用領域的一般性構想和觀念。而 GML 應用綱目則如同堆積木，同樣規格的 GML 綱目

可以組合成不一樣的 GML 應用綱目。GML 除了提供已定義好的標籤來描述外，更可讓使用者自訂標籤來描述現實的現象。

因此本研究希望藉由將資料隱藏技術導入 GML 空間資料格式中，透過不同的圖徵特性、應用目的與地圖精度條件使用不同的資料隱藏演算法與其對應的前置處理流程，以及選擇要嵌入資料的位置，同時嵌入的資料除了圖徵的屬性資料與詮釋資料外，更可嵌入自訂標籤的屬性資料，除了達到資訊安全與資料一致性的目的之外，更提出創新的 GML 資料隱藏在 GIS 上的應用方式。

2.2 資料隱藏技術及應用

資訊隱藏是一種秘密通訊的藝術與科技(Kahn, 1996)，亦可稱為掩護寫入(Covered Writing)或是在一般訊息內隱藏訊息的技術(Rabah, 2004)，是一種將秘密訊息隱藏到掩護媒體中的技術(Chao, Lin, Yu, & Lee, 2009)，如下圖 2-3 所示。

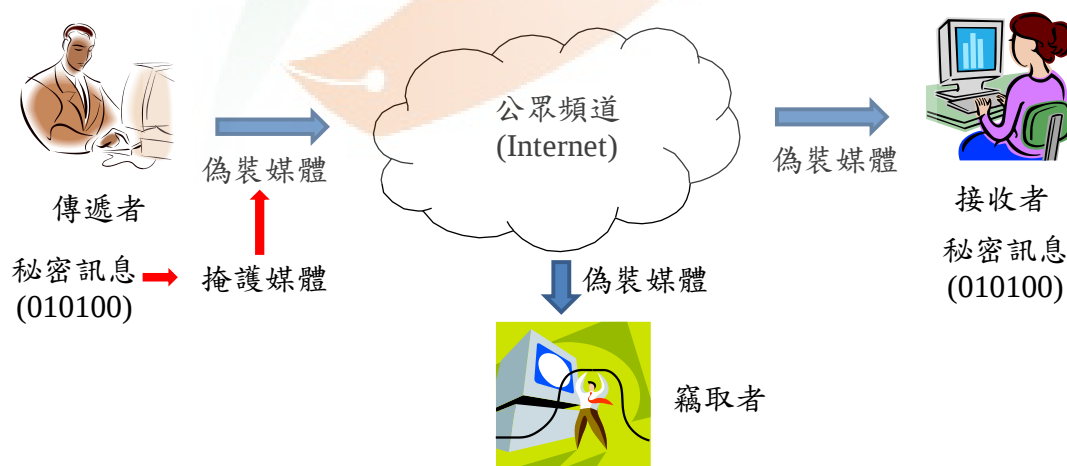


圖 2-3：資料隱藏示意圖
參考資料來源：(劉秋良，2008)

傳遞者將秘密訊息嵌入掩護媒體後產生偽裝媒體，利用偽裝媒體進行秘密通訊的動作，當竊取者在公眾頻道中擷取偽裝媒體時，不知道媒體內藏有秘密訊息，因此無法竊取到秘密訊息，當真正的接收者收到偽裝訊息時，即可將秘密訊息擷取出來，達到秘密通訊的目的。因此資訊隱藏泛指將訊息、圖片或文字等資訊藏匿在一媒體中。同時因目的不同，可以延伸出許多應用與分類(Pfitzmann, 1996)，如圖 2-4 所示。

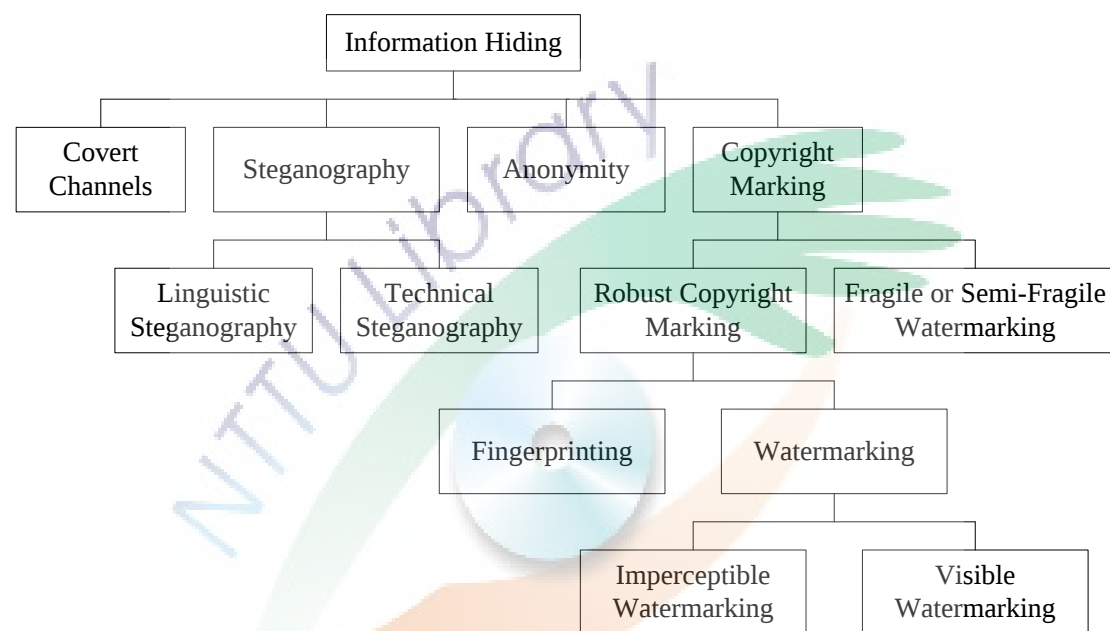


圖 2-4：資訊隱藏的分類

資料來源：(Pfitzmann, 1996)

資訊隱藏依其應用的目的不同，可以分為兩大類：浮水印(Watermarking)與資料隱藏(Data Hiding)(Popa, 1998)，如下圖 2-5 所示。底下將分別對浮水印與資料隱藏的目的與應用做簡單說明。

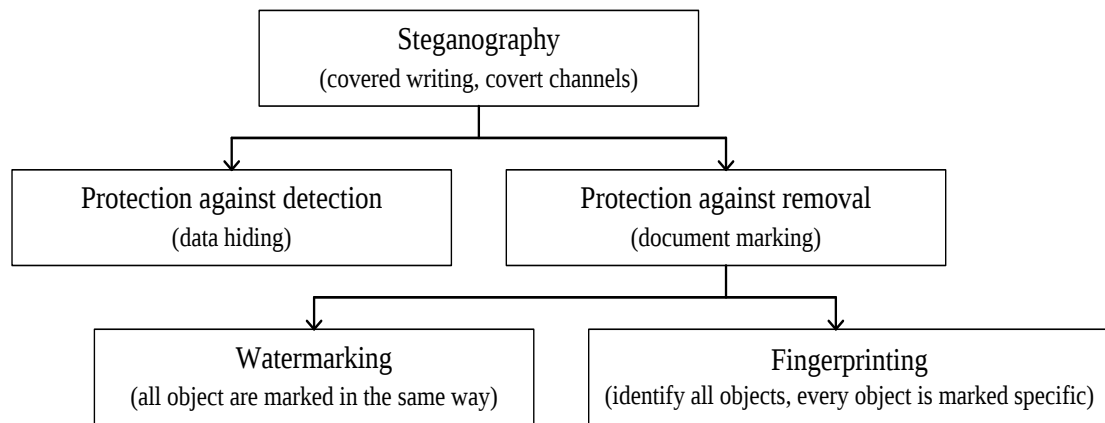


圖 2-5：資訊隱藏應用目的

資料來源：(Popa, 1998)

浮水印是指將具代表性的文字或影像，稱之為浮水印(Watermark)，嵌入需要保護的數位媒體之中。當數位媒體遭受竊取者盜用或攻擊(Attack)時，原創者可以藉由嵌入其中的浮水印證明該數位媒體的合法原始擁有者，其目的在於保護著作權，因此浮水印的技術必須具備抵抗蓄意攻擊的能力，強調其強韌性(Robustness)；反觀資料隱藏則是將欲傳送的秘密訊息嵌入一個不受他人注意的掩護媒體上，藉此迷惑竊取者的注意力，透過傳遞此媒體來達到秘密通訊的目的，故資料隱藏演算法著重於偽裝而非強韌性，強調嵌入訊息的容量。

資料隱藏是指將欲傳送的秘密訊息(Secret Message)嵌入(Embed)掩護媒體中(Cover-Media)，透過傳送已嵌入秘密訊息的偽裝媒體(Stego-Media)來間接傳遞秘密訊息，達到秘密通訊的目的。與傳統對訊息加密(Encryption)再傳送的作法不同，資料隱藏是利用偽裝的概念來提供維護資訊安全的另外一種機制，如下圖 2-6 所示。

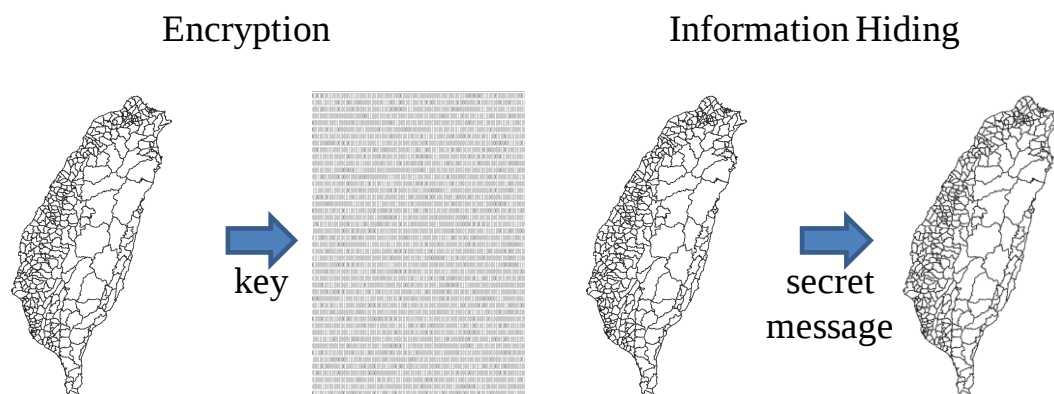


圖 2-6：資訊隱藏與加密技術的比較

資料隱藏依嵌入訊息的方式可區分為兩大領域(陳凱威，2007)：空間域(Spatial Domain)與轉換域(Transform Domain)。空間域的嵌入方式係透過改變媒體本身具備的特徵來嵌入資訊，其處理速度較快但較不具備強韌性，例如掩護媒體為一般圖片時，可改變其像素(Pixel)的數值；若為二維或三維模型時，可改變模型上的座標(Coordinates)、法向量(Normal)或顏色(Color)等數值。轉換域的嵌入方式則是先將媒體經過複雜的數學運算產生相對應的係數值，透過改變係數值來嵌入資訊。其優點為強韌性高，可分散變形量，缺點為處理速度慢。

另外根據嵌入訊息時是否會改變掩護媒體的外觀，導致容易被竊取者發現，可分為失真(Distortion)與無失真(Distortion-Free)。失真為嵌入訊息後，偽裝媒體與原本的掩護媒體在幾何或拓撲上產生差異，導致竊取者容易察覺，但其優點為可嵌入的資訊量較大。無失真為當嵌入訊息後，偽裝媒體與原本的掩護媒體在幾何或拓撲上是相同的，但其缺點為可嵌入的資料量較小。

最後資料隱藏在擷取訊息時，可依其需要的資訊，分為告知擷取(Informed detection)與盲擷取(Blind detection)兩種方式。告知擷取需要原始掩護媒體輔助才可以擷取資訊，而盲擷取則不需要原始掩護媒體，也不需要嵌入訊息的原始資訊，即可擷取訊息。在秘密通訊的過程中，若要在傳送已嵌入訊息的偽裝媒體

之外，再另外傳送掩護媒體或其資料，很容易引起竊取者的懷疑。因此資料隱藏技術發展至今，盲擷取是一重要項目，也可以方便接收方擷取秘密訊息。

目前在資料隱藏技術這方面，以圖片、音訊、視訊或是二維/三維模型為掩護媒體所發展的資料隱藏技術已經有不少的研究，但在地理資訊的研究上，目前僅針對向量地圖的部分。由於 GML 沒有資訊隱藏的相關研究，因此底下以與 GML 空間資料格式相同的向量地圖資料隱藏演算法，來探討導入 GML 空間資料格式的資料隱藏技術。

2.2.1 向量地圖資料隱藏演算法

陳凱威學者於 2007 年提出向量地圖資料隱藏演算法(陳凱威，2007)，針對 2D 向量地圖提供空間域的資料隱藏演算法。首先將掩護向量地圖中所有端點的座標擷取出來，經過主成分分析(Principle Component Analysis, PCA)後，得到兩個正交的向量，分別為第一主軸：PCA1，以及第二主軸：PCA2。然後以這兩個主軸為基底，如式(2-1)和(2-2)，形成一個以第一主軸為 X 軸和第二主軸為 Y 軸的 PCA 座標空間，透過此基底將原向量地圖上所有端點的座標轉換到 PCA 座標空間中，得到新的座標值 $P_{PCA,i}=(x_{PCA,i}, y_{PCA,i})$ ，其轉換式如式(2-3)和(2-4)，其中 n 為向量地圖端點的總數。經過空間轉換後，根據 PCA 座標空間中的 X 軸和 Y 軸分別對所有的頂點 $P_{PCA,i}$ 由小到大作排序，得到兩個排序序列(Sorting List)，分別是 X 軸排序序列與 Y 軸排序序列。

$$PCA1 = \begin{bmatrix} x_{PCA1} \\ y_{PCA1} \end{bmatrix}, PCA2 = \begin{bmatrix} x_{PCA2} \\ y_{PCA2} \end{bmatrix} \quad (2-1)$$

$$PCA \text{ 座標空間的基底} = [PCA1|PCA2] \quad (2-2)$$

$$P_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, P_{PCA,i} = \begin{bmatrix} x_{PCA,i} \\ y_{PCA,i} \end{bmatrix}, i=1, \dots, n \quad (2-3)$$

$$\begin{bmatrix} x_{PCA1} & x_{PCA2} \\ y_{PCA1} & y_{PCA2} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} x_{PCA,i} \\ y_{PCA,i} \end{bmatrix} \quad (2-4)$$

嵌入資訊時，分別對 X 軸排序序列與 Y 軸排序序列嵌入資訊，假設 $P'_1, P'_2, \dots, P'_n$ 為 X 軸排序序列中的頂點順序， $X_1, X_2, \dots, X_n$ 為這些頂點的 X 軸座標值， n 為向量地圖端點的總數。利用分層的概念，首先將這些座標值分為兩大類，分別是奇數序列(Odd list)與偶數序列(Even list)。奇數序列定義為將固定編號為奇數的頂點不動，偶序列則相反，定義為將固定編號為偶數的頂點不動，如下圖 2-7 所示。

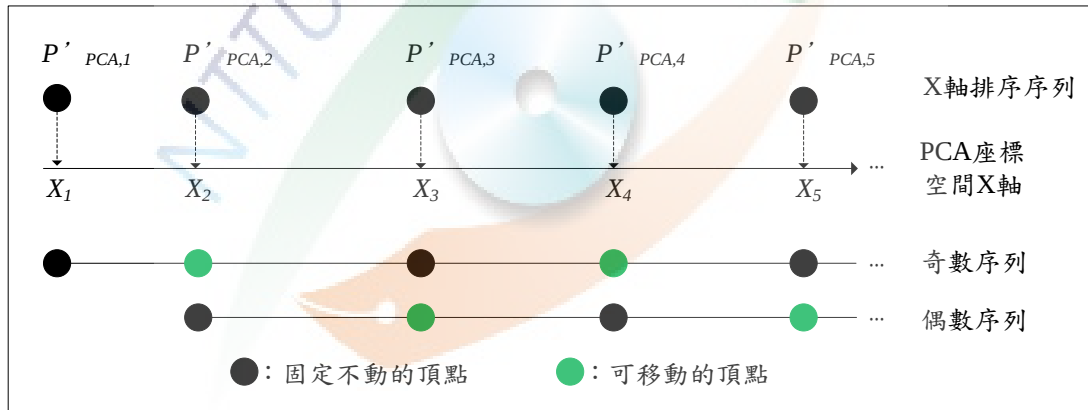


圖 2-7：奇/偶數序列
資料來源：(陳凱威，2007)

然後先對奇數序列、而後再對偶數序列進行嵌入資訊的動作，首先固定所有編號為奇數的頂點之 X 軸座標值，並定義 X_m 與 X_{m+2} 為一區間，再將每一區間平分為兩等分，區間的左半部份定義為狀態「0」，區間的右半部份定義為狀態「1」，如下圖 2-8 所示。

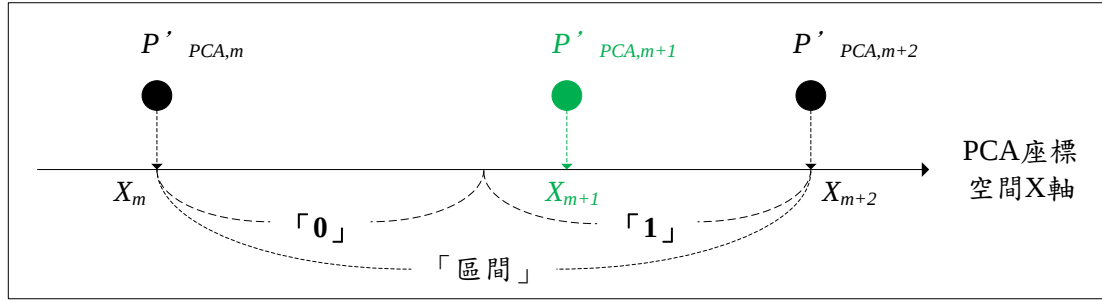


圖 2-8：定義區間
資料來源：(陳凱威，2007)

定義了區間及「0」、「1」的狀態後，逐一對每個區間 $X_m X_{m+2}$ 中的頂點 $P'_{PCA,m+1}$ 之 X 軸座標值 X_{m+1} 嵌入一個位元的資訊。為了達到「可逆」需求，記錄了 X_{m+1} 到 H_L 或 H_R 所移動的距離，嵌入公式如式(2-5)，其中 H_L 和 H_R 分別為左半部分和右半部分區間的中點， R 為 X_{m+1} 和 H_L 的差距， d 為欲嵌入的資訊。因此若嵌入的資訊為 0 時， $P'_{PCA,m+1}$ 的新 X 軸座標值定義為 $X'_{m+1} = H_L + R$, $R = (X_{m+1} - H_L)/4$ ，如下圖 2-9 所示。

$$X'_{m+1} = \begin{cases} X_L + R, R = (X_{m+1} - H_L)/4, d = "0" \\ X_R + R, R = (X_{m+1} - H_R)/4, d = "1" \end{cases} \quad (2-5)$$

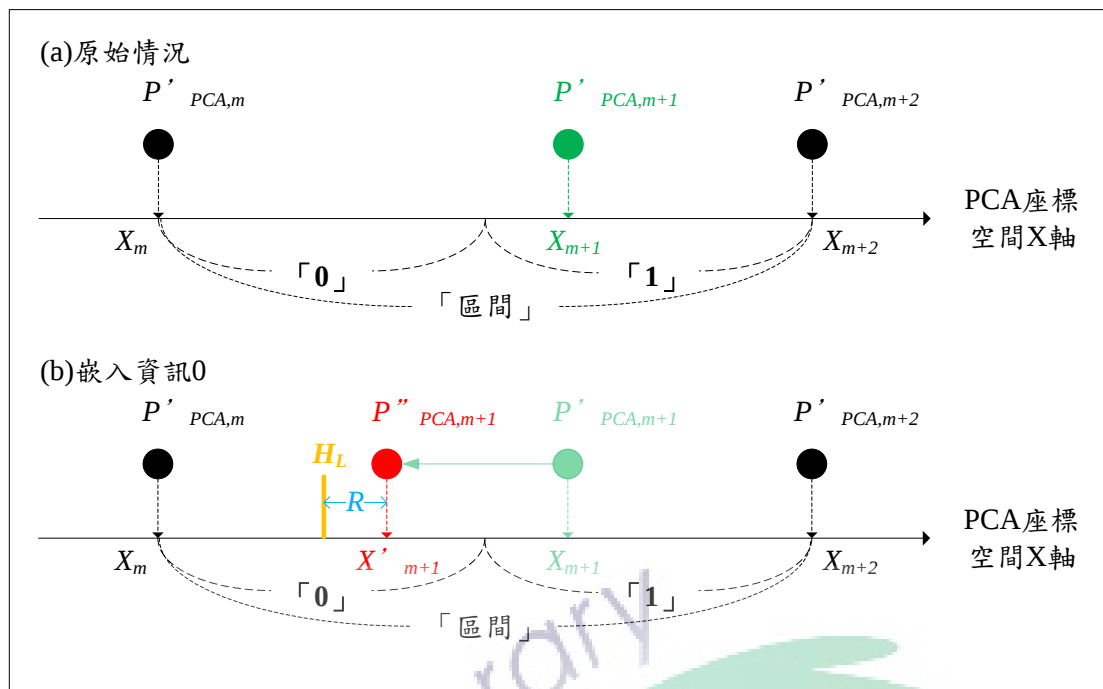


圖 2-9：嵌入資訊「0」
資料來源：(陳凱威，2007)

同理，當嵌入的資訊為 1 時， $P'_{PCA,m+1}$ 的新 X 軸座標值定義為 $X'_{m+1}=H_R+R$, $R=(X_{m+1}-H_R)/4$ ，如下圖 2-10 所示。

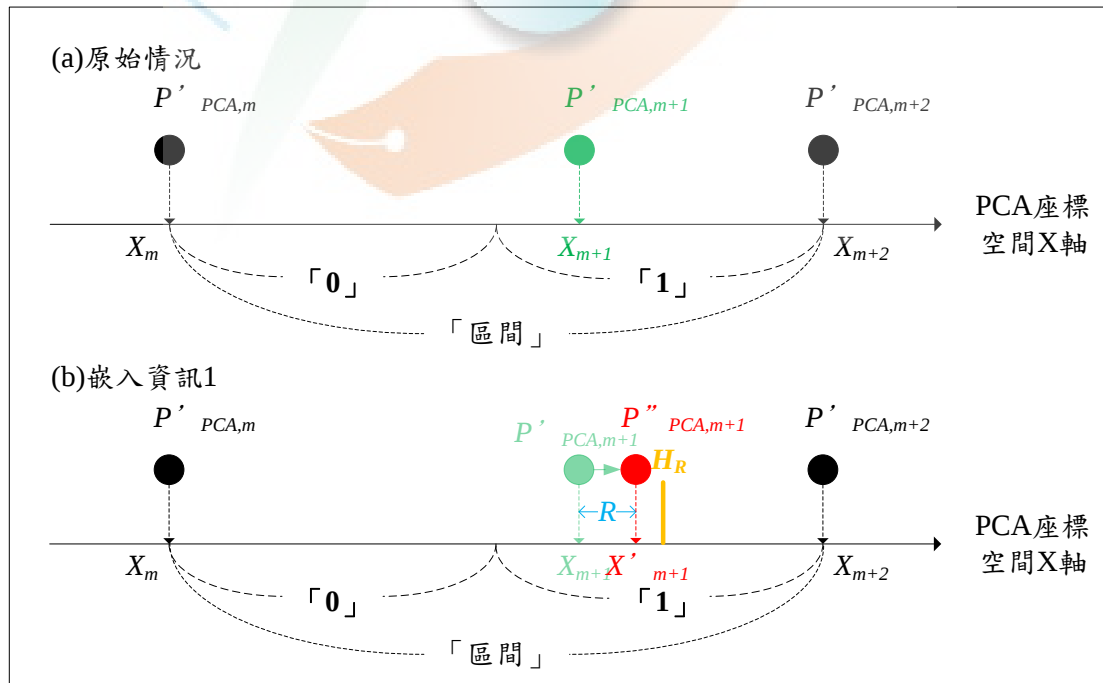


圖 2-10：嵌入資訊「1」
資料來源：(陳凱威，2007)

因此一 PCA 座標空間軸最大可嵌入(n-2)個位元，整個向量地圖最多可嵌入 2(n-2)個位元。

2.2.2 可逆式向量地圖資訊隱藏系統

劉秋良學者於 2008 年提出發展可逆式向量地圖資訊隱藏系統(劉秋良，2008)，與前一節提到的演算法不同，雖然可嵌入的資訊量仍為 2(n-2)個位元，卻降低向量地圖的變形量。首先利用 PCA 座標空間中的 X 軸和 Y 軸分別對所有的頂點 PPCA_i 由小到大作排序，得到 X 軸排序序列與 Y 軸排序序列。在嵌入資料時，利用式(2-6)將區間分割成 4 個子分割，如下表 2-1 所示，r 為頂點原本所在的區間，d 為欲嵌入的資訊。

$$s = 2 \times r + d \quad (2-6)$$

表 2-1：子分割區間
資料來源：(劉秋良，2008)

原本所在的區間 r	欲嵌入的資訊 d	新的子分割區間 s
0	0	0
0	1	1
1	0	2
1	1	3

然後利用式(2-7)和(2-8)進行嵌入資訊的動作，區間中頂點位於狀態「0」時如下圖 2-11 所示；區間中頂點位於狀態「1」時如下圖 2-12 所示。

$$X'_{m+1} = X_m + s \times \frac{X_{m+2} - X_m}{4} + \frac{\lambda}{2} \quad (2-7)$$

$$\lambda = X_{m+1} - X_m - r \times \frac{X_m + X_{m+2}}{2} \quad (2-8)$$

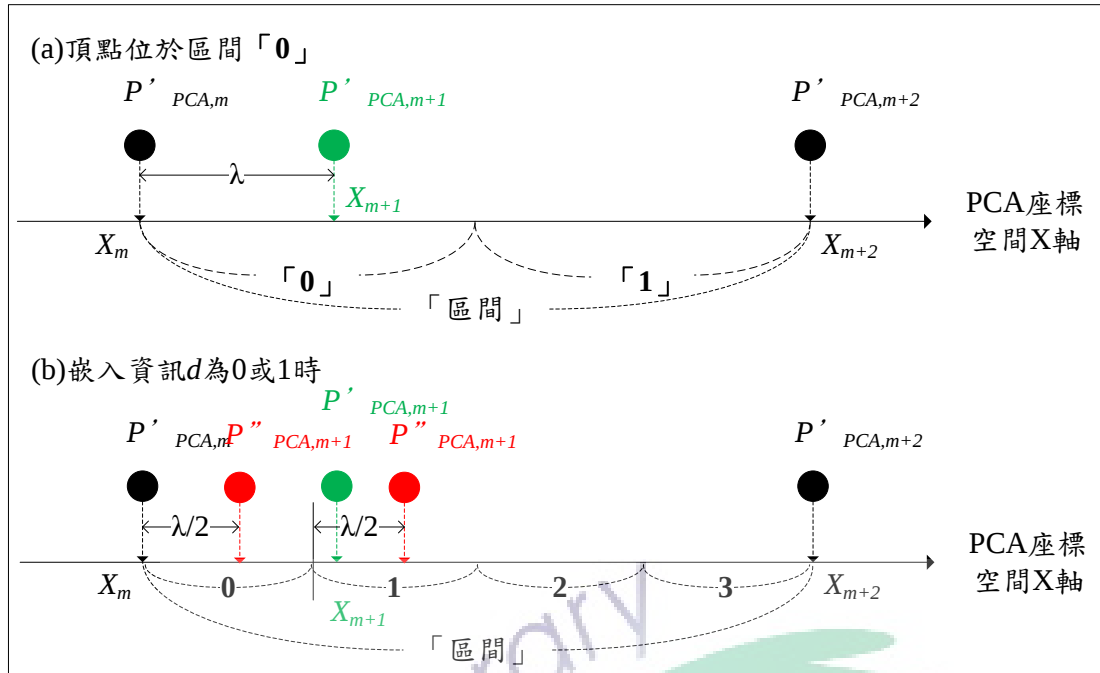


圖 2-11：頂點位於狀態「0」之嵌入動作
資料來源：(劉秋良，2008)

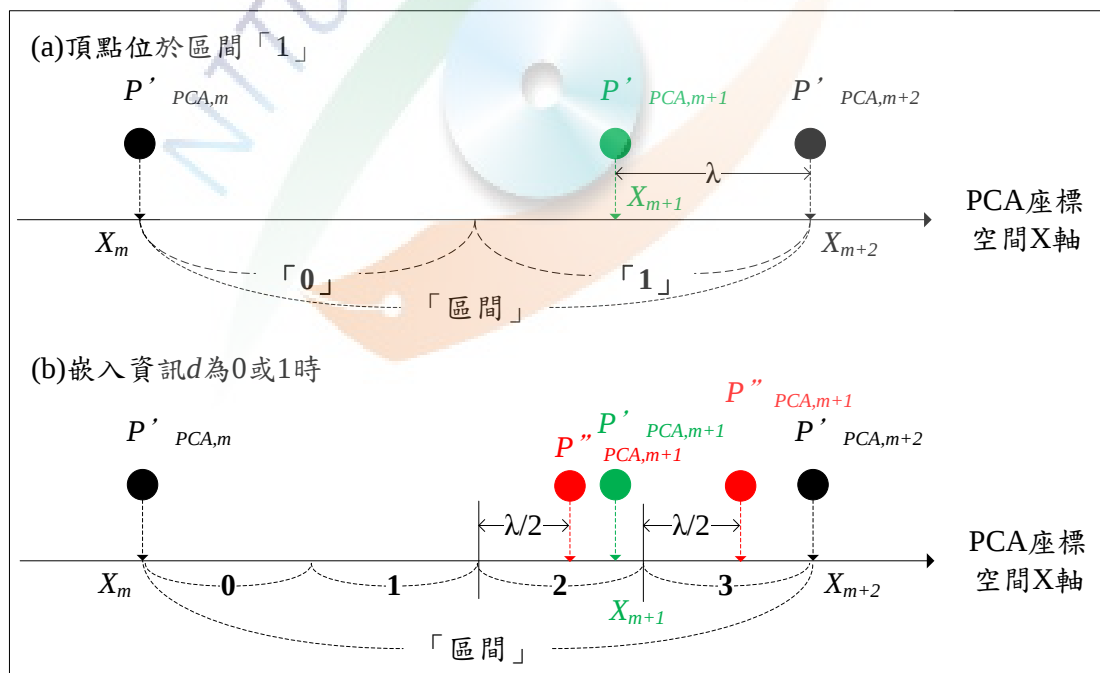


圖 2-12：頂點位於狀態「1」之嵌入動作
資料來源：(劉秋良，2008)

2.2.3 可擴展之可逆式向量地圖資料隱藏演算法

陳富美學者於 2009 年提出可逆式資訊隱藏技術應用於向量地圖後設資料嵌入之研究(陳富美, 2009), 當中發表了可擴展之可逆式向量地圖資料隱藏演算法, 改善了前兩種資訊隱藏演算法可嵌入的資訊量固定的問題, 提出嵌入量可變動的資料隱藏演算法, 其可嵌入的資訊量為 $2(n-2) \times s$ 。首先利用 PCA 座標空間中的 X 軸和 Y 軸分別對所有的頂點 $P_{PCA,i}$ 由小到大作排序, 得到 X 軸排序序列與 Y 軸排序序列。透過分割變數 s 決定切割變數 c , 切割變數 c 決定子區間的個數 v , 如下式(2-9)和(2-10), 同時 γ 為子區間之狀態, 如下圖 2-13 所示, 來決定子區間的個數和嵌入的資訊量。

$$c = (2^s - 1) \quad (2-9)$$

$$v = (c + 1) \quad (2-10)$$

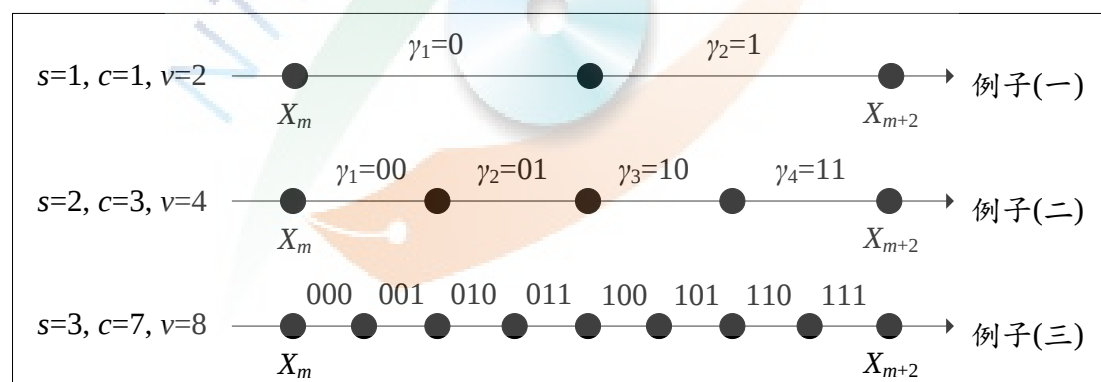


圖 2-13：分割變數 s 與切割變數 c 與子區間個數 v 之關係

資料來源：(陳富美, 2009)

然後利用式(2-11)和(2-12)進行嵌入資訊的動作, 其中 H_γ 為子區間之中點。

$$R_\gamma = \frac{(X_{m+1} - H_\gamma)}{2v} \quad (2-11)$$

$$X_{m+1}^\gamma = (H_\gamma + R_\gamma) \quad (2-12)$$

以嵌入 2 位元的訊息為例, 如下圖 2-14 所示。

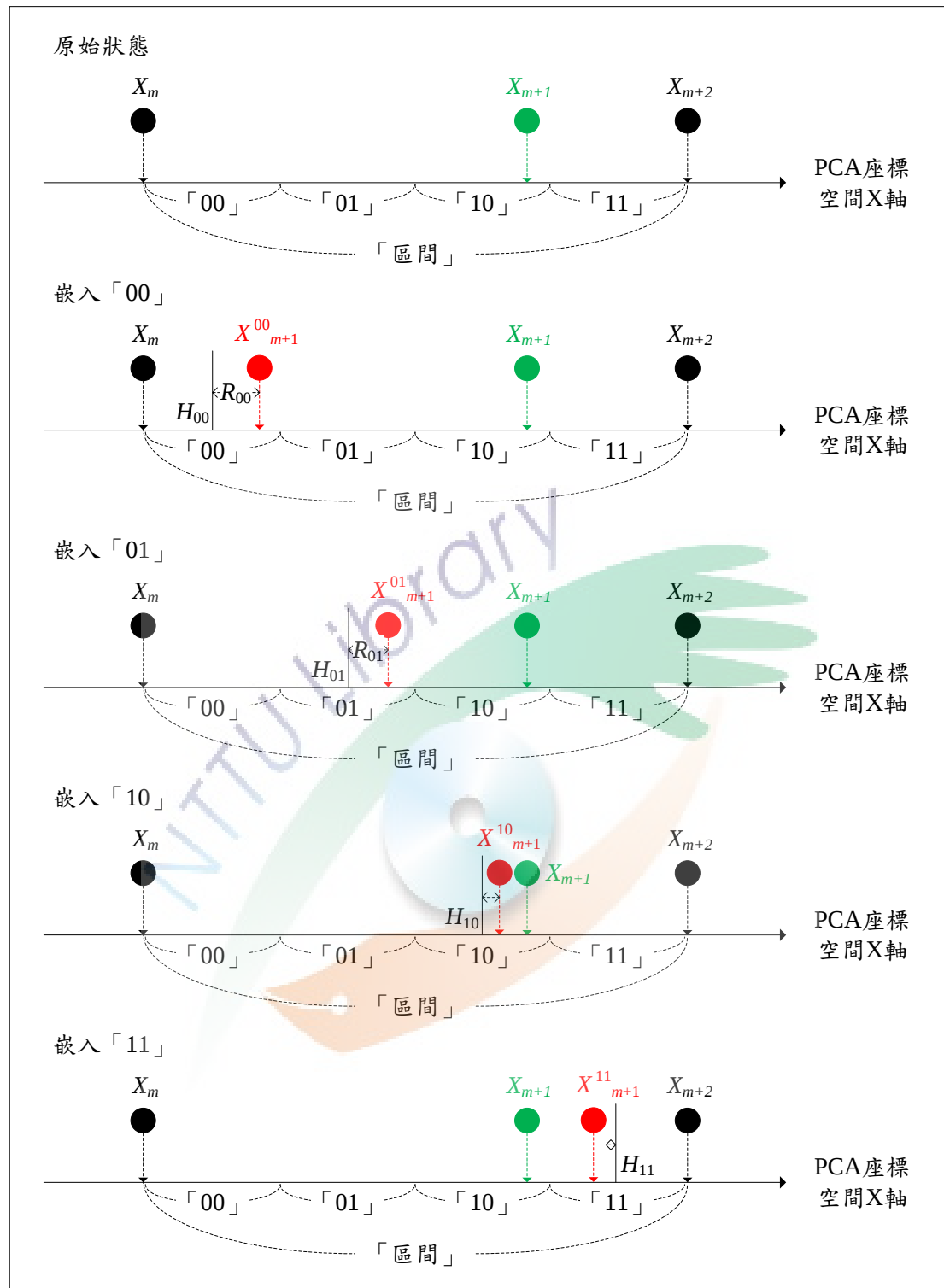


圖 2-14：嵌入 2 位元資訊

資料來源：(陳富美，2009)

綜觀上述不同的向量資料隱藏演算法，皆是透過區間的方式嵌入資料，同時所使用的座標值均經過 PCA 轉換，因此為了提升 GML 資料隱藏技術的可應用

性，本研究將針對不同的應用目的，提出不同的前置處理流程，分別是 PCA 排序法、投影排序法與直接投影法。其中可擴展之可逆式向量地圖資料隱藏演算法，除了可嵌入的訊息量最大之外，其圖形的變形量也控制在一定範圍內。故本研究將以陳富美於 2009 年提出的向量地圖資料隱藏技術導入 GML 空間資料格式中，透過 GML 檔案的特性，提出改善向量地圖資料隱藏技術的方法與 GML 在地理資訊系統(Geographic Information System, GIS)上的創新應用。

2.2.4 無失真資料隱藏演算法

Bogomjakov 等學者於 2008 年提出無失真資料隱藏用於多邊形網絡(Polygon Mesh)(Bogomjakov, et al., 2008)，利用置換端點排列順序的方式來嵌入訊息，如下圖 2-15 所示，應用於容許任意置換其成分的資料集合，改善了一般無失真資料演算法的執行時間，同時利用參照排序(Reference Ordering)的方式，可抵抗所有幾何類型上的攻擊，但也因此降低了可嵌入的資料量。

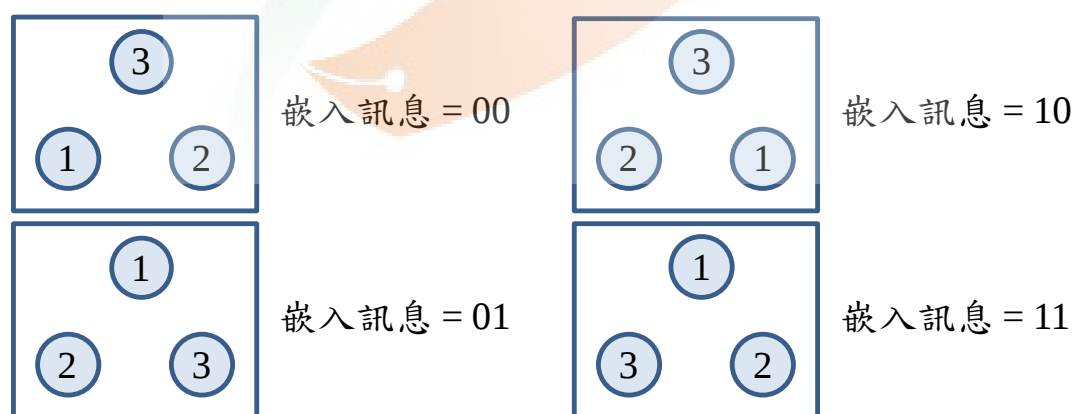


圖 2-15：置換端點排列順序嵌入訊息

首先計算出網絡端點的參照排序，並將該端點存入 ref[] 陣列中，並根據下圖 2-16 進行資料嵌入的動作，並在底下簡單介紹。

```

for i = 0,...,n-1
  k ← ⌊log2(n - i)⌋
  b ← peek(k + 1)  //peek at next k + 1 bits
  if 2k ≤ b and b < n - i
    advance(k + 1)  //advance by k+1 bits
  else
    b ← peek(k)  //peek at next k bits
    advance(k)  //advance by k bits
  end
  perm[i] ← ref[b]  //vertex b to permutation
  pef[b] ← ref[n - i - 1]  //replace with last vertex
end

```

圖 2-16：無失真資料隱藏演算法

資料來源：(Bogomjakov, et al., 2008)

定義 n 為初始參照排序的端點總數， i 為已經重新排列的端點數，從 $n - i$ 個端點中挑選一個位置 b 的端點，並且將它輸出為下一個置換(Permutation)的端點， b 為一個介於 $(0, \dots, n - i - 1)$ 的整數，可編碼輸入訊息 $k = \lfloor \log_2(n - i) \rfloor$ 個位元。一開始先從輸入訊息中讀取 $k + 1$ 個位元，若其 b 值小於 $n - i$ 並且大於或等於 2^k ，此時透過挑選位置 b 的端點可編碼 $k + 1$ 個位元，否則直接用 k 的 b 值當作挑選端點的位置，此時可編碼 k 個位元。

被挑選的端點輸出完畢後，便將該端點從參照排序中刪除，並且移動參照排序中最末尾的端點到該位置，表示在參照排序中剩下的端點以連續的方式儲存於陣列之中，因此時間複雜度為 $O(n)$ ，同時可嵌入的訊息量為 $\log_2(n!) - n + 1$ 個位元。

第三章 研究方法

根據文獻回顧本研究發現，前述資料隱藏演算法都僅針對向量座標或特定概念架構做研究，並未實際應用於地理資訊的檔案格式中，另外向量地圖資料隱藏演算法所使用的座標值都經過 PCA 的轉換，導致座標值改變，使地圖的精度降低，造成失真擴大的現象。因此本研究希望藉由將不同類型的資料隱藏演算法導入 GML 空間資料格式，根據不同的應用目的提出不同的前置處理方法，進一步探討可發展的 GML 資料隱藏技術應用，研究流程如下圖 3-1 所示。

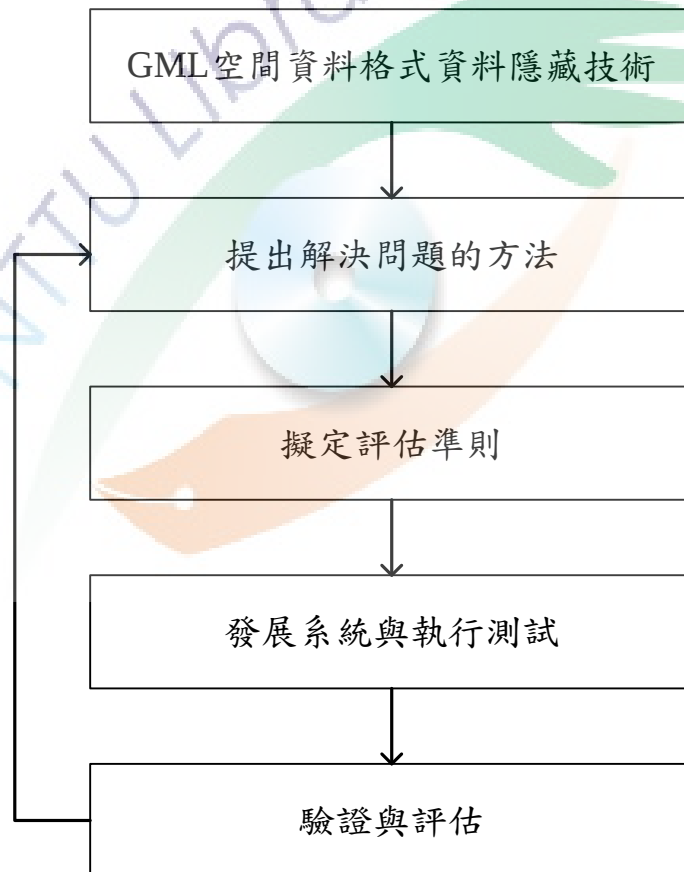


圖 3-1：研究流程

本研究主要針對 GML 檔案提出資料隱藏技術之應用，除了實際將可擴展之可逆式向量地圖資料隱藏演算法(陳富美，2009)與無失真資料隱藏演算法(Bogomjakov, et al., 2008)導入 GML 空間資料格式中，以提升 GML 資料的安全性外，並根據不同應用目的與地圖精度條件，針對可擴展之可逆式向量地圖資料隱藏演算法提出三種不同的前置處理流程：PCA 排序法、投影排序法與直接投影法，增加 GML 資料隱藏技術之可應用性，同時利用標籤語法的特性，將秘密資料分別嵌入不同的圖徵類型與圖徵當中，藉此提升秘密資料的安全性，並且增加可嵌入的資料量，達到 GML 空間資料格式之資料隱藏技術創新應用。同時利用可嵌入訊息量、不可察覺性、可逆性與執行時間等評估指標，來衡量不同資料隱藏演算法與不同前置處理方法導入之結果，評估 GML 空間資料格式資料隱藏技術之成效。

因此底下第一節詳細分析 GML 空間資料格式，找出可應用的領域；第二節將提出本研究的 GML 空間資料格式之資料隱藏技術，除了實際導入可擴展之可逆式向量地圖資料隱藏演算法(陳富美，2009)與無失真資料隱藏演算法(Bogomjakov, et al., 2008)外，並提出本研究之前置處理與資料嵌入流程。

3.1 GML 空間資料格式

GML 是以 XML 為發展基礎，提供多種地理資料的編碼方式，解決了異質性資料整合的問題(林峰田，2006)。透過標籤來描述欲表達的現象，利用標籤的屬性值來描述圖徵性質，而圖徵代表的是真實世界中的實際現象，其中屬性的值可以是整數、浮點數或幾何等型態，甚至其他的圖徵也可以做為屬性的值。GML 空間資料格式中，圖徵可以分為兩種屬性，分別是拓撲(Topology)與幾何(Geometry)，如下圖 3-2 所示(鄧東波，2004)。

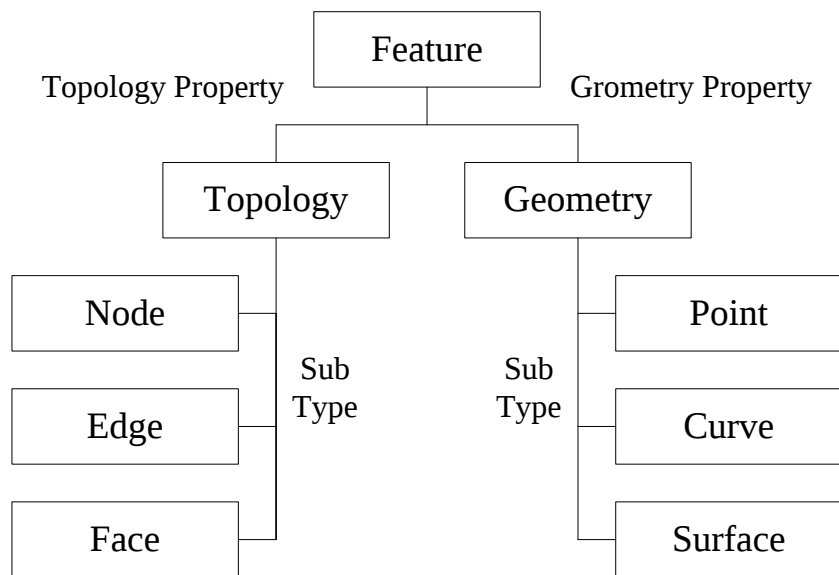


圖 3-2：圖徵之拓樸與幾何屬性
資料來源：(鄧東波，2004)

簡單來說，拓樸是一種擺脫測量(Measurement)和距離(Distance)的一種幾何數學上運用的方法，因為擺脫了計量法(Metrics)的約束，而能夠允許變形的發生。在空間上，所有的扭曲、伸長或壓縮，從拓樸的角度來看，並不會對原本的空間造成影響，因此在這個條件下，圓形和橢圓形是一樣的，球體也等於橢圓體。且拓樸最大的特點在於：在一個已經發生變形的狀態下，仍然保有某些特定的性質，而那些不變量(Invariants)便是拓樸所要研究的重點。有別於一般傳統的幾何容易聯想到可測量的量，例如角度、距離、空間等等，拓樸則是著重其連續性(Continuity)以及相對位置(Relative position)的關係。

由於本研究將導入的可擴展之可逆式向量地圖資料隱藏演算法(陳富美，2009)與無失真資料隱藏演算法(Bogomjakov, et al., 2008)，其方法架構於幾何上的向量座標與多邊形網絡，與拓樸較無直接相關，因此本研究主要針對幾何屬性做為導入資料隱藏技術的部分，其子類型分別為頂點(Point)、曲線(Curve)與表面(Surface)，詳述如下。

3.1.1 GML 頂點格式

首先頂點的部分，是以點的方式來描繪圖徵，其順序沒有意義，僅為名目上之定義，例如標示城市位置、捷運站或疾病案例等資訊。其示意圖如下圖 3-3 所示，語法如下表 3-1 所示。

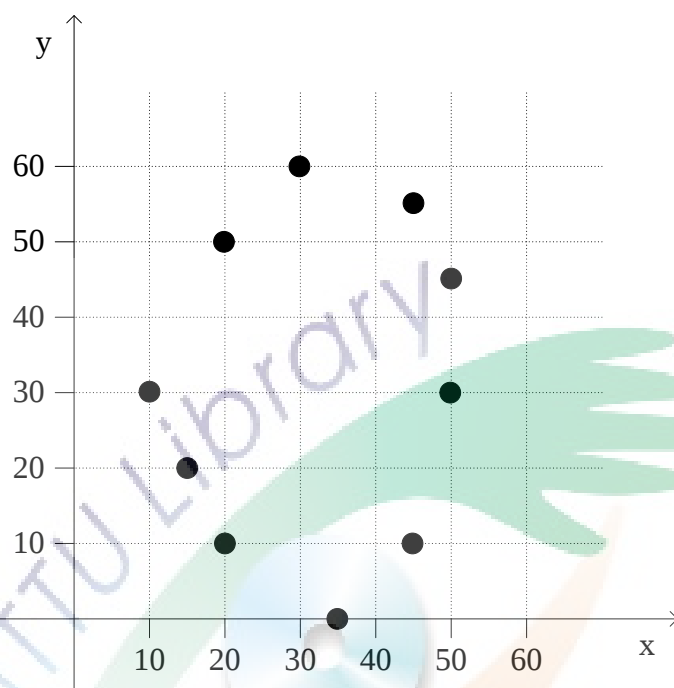


圖 3-3：頂點語法示意圖

表 3-1：地理標記語言頂點屬性語法與說明

語法	說明
(1) <code><gml:pointArrayProperty></code>	(1) 以陣列的方式儲存頂點資料
(2) <code><gml:Point></code>	(2) 定義元素為頂點
(3) <code><gml:coordinates>30,60 20,50 10,30 15,20 20,10 35, 0 45,10 50,30 50,45 45,55 </gml:coordinates></code>	(3) 頂點的座標值，分別為 X 軸與 Y 軸座標值，中間以「,」區隔
(4) <code></gml:Point></code>	
(5) <code></gml:pointArrayProperty></code>	

3.1.2 GML 曲線格式

其次曲線的部分，是以線的方式來描繪圖徵，例如描繪公路線或下水道管線等資訊。其示意圖如下圖 3-4 所示，語法如下表 3-2 所示。

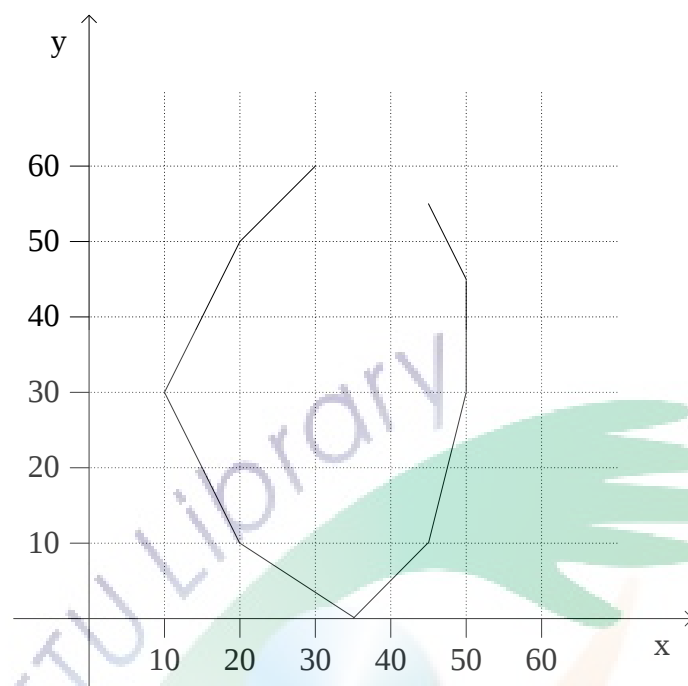


圖 3-4：曲線語法示意圖

表 3-2：地理標記語言曲線屬性語法與說明

語法	說明
(1) <gml:curveProperty>	(1) 定義曲線，其值域可以是幾何元素或參照 XLink 的幾何元素
(2) <gml:Curve>	(2) 定義元素為曲線，由一或多個曲線片段組成
(3) <gml:segments>	(3) 為封裝曲線片段
(4) <gml:LineStringSegment>	(4) 一種曲線片段，由二維或多維座標組成
(5) <gml:coordinates>30,60 20,50 10,30 15,20 20,10 35, 0 45,10 50,30 50,45 45,55	(5) 頂點的座標值，分別為 X 軸與 Y 軸座標值，中間以「,」區隔
(6) </gml:coordinates>	
(7) </gml:LineStringSegment>	
(8) </gml:segments>	
(9) </gml:Curve>	
</gml:curveProperty>	

3.1.3 GML 表面格式

最後表面的部分，是以面的方式來描繪圖徵，例如描繪地形或行政區等資訊。其示意圖如下圖 3-5 所示，語法如下表 3-3 所示。

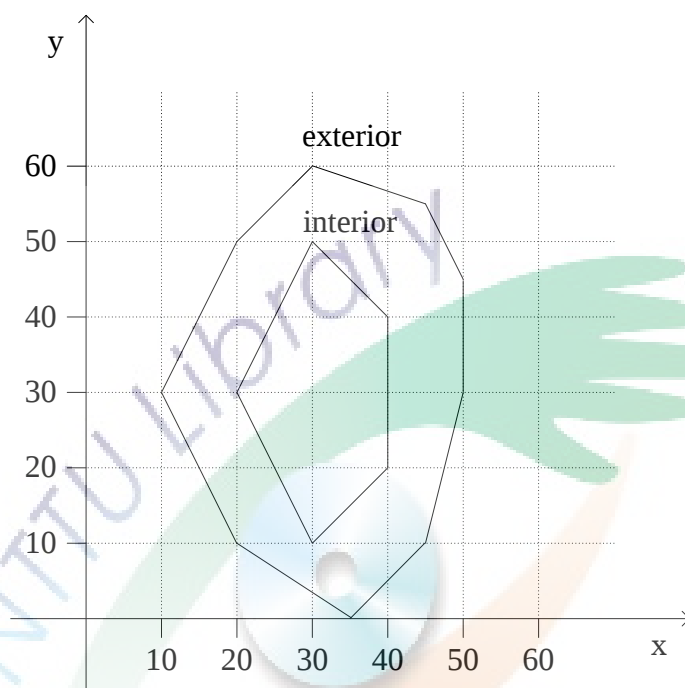


圖 3-5：表面語法示意圖

表 3-3：地理標記語言表面屬性語法與說明

語法	說明
(1) <gml:polygonProperty>	(1) 定義多邊形，其值域可以是幾何元素或參照 XLink 的幾何元素
(2) <gml:Polygon>	
(3) <gml:exterior>	(2) 定義元素為多邊形
(4) <gml:Ring>	(3) 定義為多邊形之外框，若要定義中空的多邊形之內框，則可用
(5) <gml:curveMember>	
(6) <gml:Curve>	<gml:interior>
(7) <gml:segments>	(4) 連接不同的邊，形成環狀結構
(8) <gml:LineStringSegment>	(5) 成分為經由 XLink 屬性的曲線或曲線成分
(9) <gml:coordinates>30,60 20,50 10,30 15,20 20,10 35, 0 45,10 50,30 50,45 45,55 30,60	(6) 定義元素為曲線，由一或多個曲線片段組成
(10) </gml:coordinates>	(7) 為封裝曲線片段
(11) </gml:LineStringSegment>	(8) 一種曲線片段，由二維或多維座標組成
(12) </gml:segments>	
(13) </gml:Curve>	(9) 頂點的座標值，分別為 X 軸與 Y 軸座標值，中間以「,」區隔
(14) </gml:curveMember>	
(15) </gml:Ring>	(17) 定義中空的多邊形之內框
(16) <gml:exterior>	
(17) <gml:interior>	
(18) <gml:Ring>	
(19) <gml:curveMember>	
(20) <gml:Curve>	
(21) <gml:segments>	
(22) <gml:LineStringSegment>	
(23) <gml:coordinates>30,50 20,30 30,10 40,20 40,40 30,50	
(24) </gml:coordinates>	
(25) </gml:LineStringSegment>	
(26) </gml:segments>	
(27) </gml:Curve>	
(28) </gml:curveMember>	
(29) </gml:Ring>	
(30) </gml:interior>	
(31) </gml:Polygon>	
(32) </gml:polygonProperty>	

其中幾何屬性中所有子類型均架構於座標系統上，透過不同的標籤來定義不同的圖徵型式，同時因具備自訂標籤的特性，使用者可以自由描述現象。因此本研究認為 GML 空間資料格式可用於可擴展之可逆式向量地圖、無失真以及與頂點相關的資料隱藏演算法，並且可根據應用目的與地圖精度條件來進行不同的前置處理方式與選擇嵌入訊息的圖徵類型與圖徵，同時可嵌入的資訊包含詮釋資料或 GML 空間資料格式之屬性資料。第二節將詳細介紹本研究提出之 GML 空間資料格式資料隱藏技術之嵌入資料流程，最後第三節介紹擷取資料之流程。



3.2 資料嵌入流程

本研究之資料嵌入流程如下圖 3-6 所示。

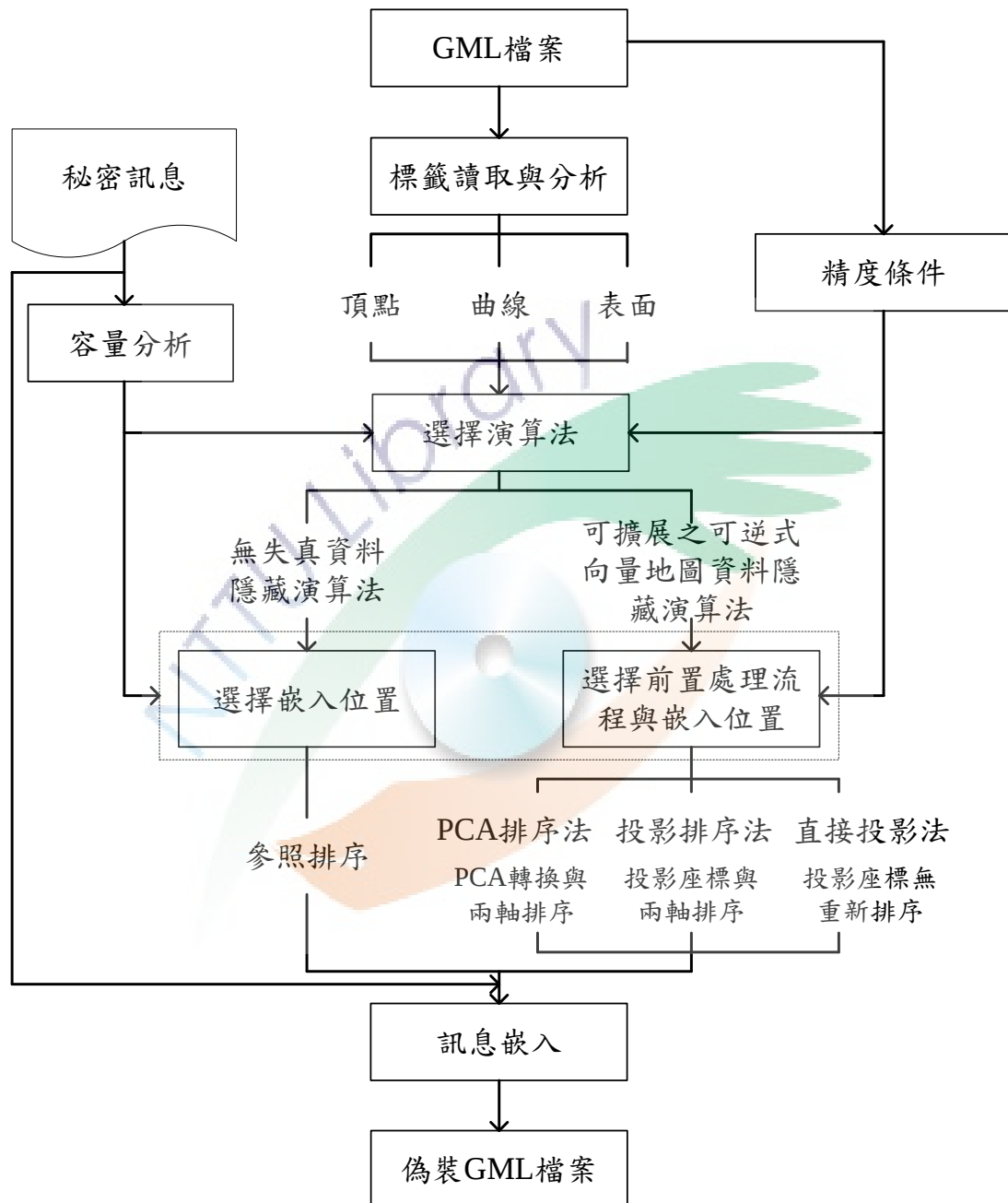


圖 3-6：資料嵌入流程

首先開啟 GML 檔案，除了擷取標籤進行分析外，並記錄該地圖資料的精度條件，例如 1/5,000、1/10,000 或 1/25,000 等比例。然後擷取「頂點」、「曲線」與「表面」三種類型的屬性資料後，根據秘密訊息的容量大小與 GML 檔案的精度條件來選擇使用的資料隱藏演算法、前置處理的流程與嵌入資料的圖徵類型與圖徵，最後進行秘密訊息嵌入的動作，得到偽裝 GML 檔案。

其中可擴展之可逆式向量地圖資料隱藏演算法的前置處理流程共分為三種：「PCA 排序法」，將端點座標進行 PCA 轉換，得到兩軸資料，並分別進行排序，得到兩軸序列；「投影排序法」，將端點的座標值投影到兩軸上，得到兩軸資料，並分別進行排序，得到兩軸序列；「直接投影法」，將端點的座標值投影(Projection)到兩軸上，得到兩軸資料，不進行排序的動作。而無失真資料隱藏演算法的前置處理流程，僅需計算參照排序即可。另外嵌入資料的圖徵類型為幾何屬性中之子類型，分別為：頂點、曲線與表面，分別進行資料隱藏的動作，分類如下圖 3-7 所示，並分述如下。



圖 3-7：嵌入資料之圖徵類型分類

3.2.1 頂點之圖徵類型

頂點之圖徵類型可分為「多個單一頂點」與「多個頂點」兩種子類別，其中多個單一頂點的部分，例如台灣地圖中五個直轄市的位置，如下圖 3-8 所示。每個頂點代表不同的意義，可分開獨立使用，因此頂點的屬性值都跟隨著頂點，同時只有一組頂點座標值。若針對累積多個單一頂點進行資料隱藏的動作，因頂點數不足，若產生變形就更容易被察覺，而且需要移動大量的屬性資料，可能造成資料損壞，因此本研究不對此類別進行資料隱藏的動作。例如針對標示台北市位置的頂點進行資料隱藏的動作，因為只有一組頂點座標，故無法套用於可擴展之可逆式向量地圖與無失真資料隱藏演算法；若將標示五都所在位置的頂點座標集合起來進行資料嵌入的動作，可嵌入的訊息量非常少，同時需移動大量的屬性資料，因此本研究不針對此類別嵌入資料。

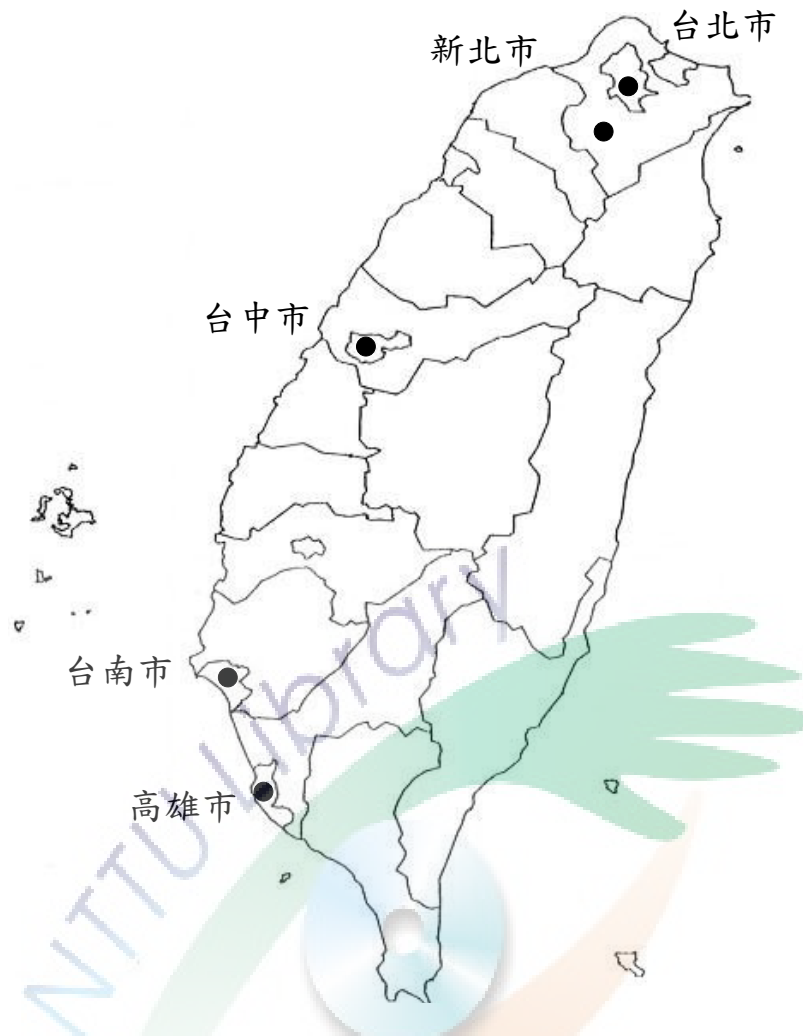


圖 3-8：台灣五直轄市所在位置

至於多個頂點類別，因頂點之間沒有直接相關，同時頂點的排列順序無特別意義，故擷取各頂點之座標後，選擇適當的隱藏演算法，並依情況進行對應的前置處理流程。

讀取標籤內容演算法如下圖 3-9 所示，其中「model」用來儲存 GML 空間資料格式內的資料，「n」為記錄圖徵順序與編號。

```

FileInputStream File = new FileInputStream(gmlFile);
InputStreamReader ISReader = new InputStreamReader(File);
StreamTokenizer ST = new StreamTokenizer(ISReader);
Model2D model = new Model2D();
int n = 1;
scan:
while (true) {
    switch (ST.nextToken()) {
        case StreamTokenizer.TT_EOL:
            break;
        case StreamTokenizer.TT_EOF:
            break scan;
        case StreamTokenizer.TT_WORD:
            if ("gml:pointArrayProperty".equals(ST.sval)) {
                ST.nextToken();
                if ("gml:Point".equals(ST.sval)) {
                    ST.nextToken();
                    if ("gml:coordinates".equals(ST.sval)) {
                        ST.nextToken();
                        while (ST.ttype != StreamTokenizer.TT_WORD) {
                            if (ST.ttype == StreamTokenizer.TT_NUMBER) {
                                x = ST.nval;
                                ST.nextToken();
                                y = ST.nval;
                                ST.nextToken();
                                Model.addVertex(x,y,n);
                                n++;
                            }
                        }
                    }
                    else
                        break;
                }
            }
        }
    }
}
ISReader.close();

```

圖 3-9：讀取頂點標籤內容演算法

詳細步驟如下：

- Step 1: 開啟 GML 檔案，讀取標籤內容。
- Step 2: 搜尋<gml:pointArrayProperty>標籤。
- Step 3: 判斷子標籤，找出<gml:Point>標籤。
- Step 4: 判斷子標籤，找出<gml:coordinates>標籤。
- Step 5: 擷取<gml:coordinates>與</gml:coordinates>標籤所包覆的座標值。
- Step 6: 將 X 軸與 Y 軸座標分開儲存。

例如：<gml:coordinates>與</gml:coordinates>標籤所包覆的屬性值為

「30,60 20,50 10,30 15,20 20,10 35, 0 45,10 50,30 50,45 45,55」，因此
X 軸座標序列為「30, 20, 10, 15, 20, 35, 45, 50, 50, 45」， Y 軸座標序
列為「60, 50, 30, 20, 10, 0, 10, 30, 45, 55」。

Step 7: 重複執行上述步驟，直到所有的座標值都讀取完畢，才執行底下步
驟。

Step 8: 將得到的兩軸座標序列，根據所選擇的資料隱藏演算法與前置處理
流程，進行資料嵌入的動作。

3.2.2 曲線之圖徵類型

曲線之圖徵類型可分為「單一曲線」與「多條曲線」兩種子類別，首先單一
曲線的類別，因曲線當中的端點之間有關連性，同時不可任意更換端點的排列順
序，因此擷取各端點之座標值後，先記錄起始點與終點的座標值與原始排列順
序，接著選擇資料隱藏演算法，並依情況進行對應的前置處理流程，演算法如下
圖 3-10 所示。

```

FileInputStream File = new FileInputStream(gmlFile);
InputStreamReader ISReader = new InputStreamReader(File);
StreamTokenizer ST = new StreamTokenizer(ISReader);
Model2D model = new Model2D();
int n = 1;
scan:
while (true) {
    switch (ST.nextToken()) {
        case StreamTokenizer.TT_EOL:
            break;
        case StreamTokenizer.TT_EOF:
            break scan;
        case StreamTokenizer.TT_WORD:
            if ("gml:curveProperty".equals(ST.sval)) {
                ST.nextToken();
                if ("gml:Curve".equals(ST.sval)) {
                    ST.nextToken();
                    if ("gml:segments".equals(ST.sval)) {
                        ST.nextToken();
                        if ("gml:LineStringSegment".equals(ST.sval)) {
                            ST.nextToken();
                            if ("gml:coordinates".equals(ST.sval)) {
                                ST.nextToken();
                                while (ST.ttype != StreamTokenizer.TT_WORD) {
                                    if (ST.ttype == StreamTokenizer.TT_NUMBER) {
                                        x = ST.nval;
                                        ST.nextToken();
                                        y = ST.nval;
                                        ST.nextToken();
                                        Model.addVertex(x,y,n);
                                        n++;
                                    }
                                }
                                else
                                    break;
                            }
                        }
                    }
                }
            }
        }
    }
    ISReader.close();
}

```

圖 3-10：讀取曲線標籤內容演算法

詳細步驟如下：

Step 1: 開啟 GML 檔案，讀取標籤內容。

Step 2: 搜尋<gml:curveProperty>標籤。

Step 3: 判斷子標籤，找出<gml:Curve>標籤。

Step 4: 判斷子標籤，找出<gml:segments>標籤，根據此標籤來判定是否為多條曲線類別。

Step 5: 判斷子標籤，找出<gml:LineStringSegment>標籤。

Step 6: 判斷子標籤，找出<gml:coordinates>標籤。

Step 7: 擷取<gml:coordinates>與</gml:coordinates>標籤所包覆的座標值。

Step 8: 將 X 軸與 Y 軸座標分開儲存。

例如：<gml:coordinates>與</gml:coordinates>標籤所包覆的屬性值為「30,60 20,50 10,30 15,20 20,10 35, 0 45,10 50,30 50,45 45,55」，因此 X 軸座標序列為「30, 20, 10, 15, 20, 35, 45, 50, 50, 45」， Y 軸座標序列為「60, 50, 30, 20, 10, 0, 10, 30, 45, 55」。

Step 9: 紀錄起始點與終點座標值，分別為：「30,60」與「45,55」，然後重複執行上述步驟，直到所有的座標值都讀取完畢，才執行底下步驟。

Step 10: 將得到的兩軸座標序列，根據所選擇的資料隱藏演算法與前置處理流程，進行資料嵌入的動作。

3.2.3 表面之圖徵類型

表面之圖徵類型可分為「單一表面」與「多個表面」兩種子類別，首先單一表面類別，因表面是由多個端點所組成，故端點之間有關連性，同時不可任意更換端點的排列順序。因此可將一個表面當作是一條環繞的曲線，其起始點與終點為同一端點，座標值相同，擷取各端點之座標值後，先記錄起始點與終點的座標值與原始的排列順序，接著選擇資料隱藏演算法，並依情況進行對應的前置處理流程。

而多個表面類別的部分，與一般空間資料格式不同，GML 儲存空間資料時，是將圖徵分割成各個獨立的子圖徵，並依照需求不同組合使用。例如台灣地圖在一般空間資料儲存時，是將各縣市的圖徵資料全部儲存在一起，透過索引來搜尋各自的資料；但使用 GML 空間資料格式儲存時，可將每個縣市的圖徵與屬性資

料分開儲存，因此可以個別呈現各縣市的圖徵與資料，亦可組合成完整的台灣地圖加以顯示，如下圖 3-11 所示。

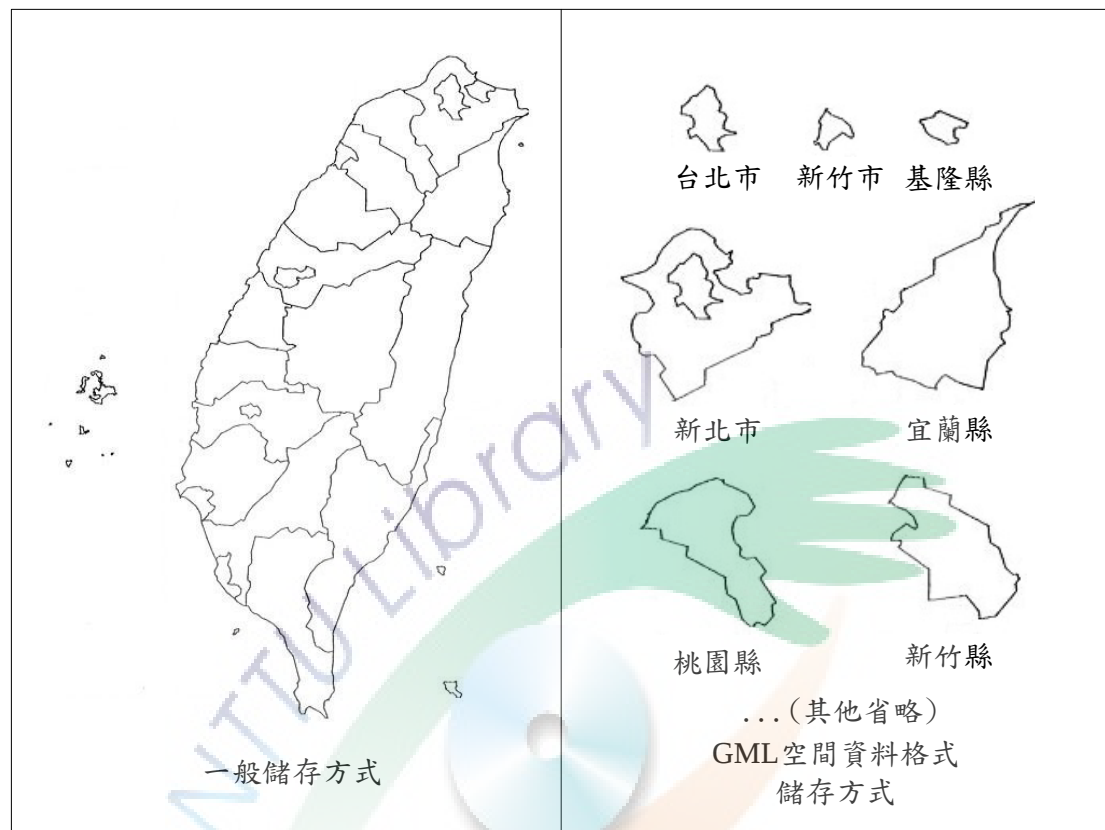


圖 3-11：空間資料儲存比較

因此不同表面需分別記錄起始點的座標值，先記錄起始點與終點的座標值與原始的排列順序，接著選擇資料隱藏演算法，並依情況進行對應的前置處理流程，演算法如下圖 3-12 所示。

```

FileInputStream File = new FileInputStream(gmlFile);
InputStreamReader ISReader = new InputStreamReader(File);
StreamTokenizer ST = new StreamTokenizer(ISReader);
Model2D model = new Model2D();
int n = 1;
scan:
while (true) {
    switch (ST.nextToken()) {
        case StreamTokenizer.TT_EOL:
            break;
        case StreamTokenizer.TT_EOF:
            break scan;
        case StreamTokenizer.TT_WORD:
            if ("gml:polygonProperty".equals(ST.sval)) {
                ST.nextToken();
                if ("gml:Polygon".equals(ST.sval)) {
                    ST.nextToken();
                    if ("gml:exterior".equals(ST.sval) || "gml:interior".equals(ST.sval)) {
                        ST.nextToken();
                    }
                    if ("gml:Ring".equals(ST.sval)) {
                        ST.nextToken();
                    }
                    if ("gml:curveMember".equals(ST.sval)) {
                        ST.nextToken();
                    }
                    if ("gml:Curve".equals(ST.sval)) {
                        ST.nextToken();
                    }
                    if ("gml:segments".equals(ST.sval)) {
                        ST.nextToken();
                    }
                    if ("gml:LineStringSegment".equals(ST.sval)) {
                        ST.nextToken();
                    }
                    if ("gml:coordinates".equals(ST.sval)) {
                        ST.nextToken();
                        while (ST.ttype != StreamTokenizer.TT_WORD) {
                            if (ST.ttype == StreamTokenizer.TT_NUMBER) {
                                x = ST.nval;
                                ST.nextToken();
                                y = ST.nval;
                                ST.nextToken();
                                Model.addVertex(x,y,n);
                                n++;
                            }
                        }
                    }
                    else
                        break;
                }
            }
            }}}}}}}}}}}
ISReader.close();

```

圖 3-12：讀取表面標籤內容演算法

詳細步驟如下：

Step 1: 開啟 GML 檔案，讀取標籤內容。

Step 2: 搜尋<gml:polygonProperty>標籤。

- Step 3: 判斷子標籤，找出<gml:Polygon>標籤。
- Step 4: 判斷子標籤，找出<gml:exterior>和<gml:interior>標籤。根據<gml:exterior>標籤來判定為多邊形之外框；根據<gml:interior>標籤來判定為多邊形之內框。首先對<gml:exterior>標籤進行 Step 5 的步驟，當結束嵌入流程後，再對<gml:interior>標籤進行 Step 5 的步驟。
- Step 5: 判斷子標籤，找出<gml:Ring>標籤。
- Step 6: 判斷子標籤，找出<gml:curveMember>標籤。
- Step 7: 判斷子標籤，找出<gml:Curve>標籤。
- Step 8: 判斷子標籤，找出<gml:segments>標籤。
- Step 9: 判斷子標籤，找出<gml:LineStringSegment>標籤。
- Step 10: 判斷子標籤，找出<gml:coordinates>標籤。
- Step 11: 擷取<gml:coordinates>與</gml:coordinates>標籤所包覆的座標值。
- Step 12: 將 X 軸與 Y 軸座標分開儲存，同時去除最後一組座標值。
例如：<gml:coordinates>與</gml:coordinates>標籤所包覆的屬性值為「30,60 20,50 10,30 15,20 20,10 35, 0 45,10 50,30 50,45 45,55 30,60」，因此 X 軸座標序列為「30, 20, 10, 15, 20, 35, 45, 50, 50, 45」，Y 軸座標序列為「60, 50, 30, 20, 10, 0, 10, 30, 45, 55」。
- Step 13: 紀錄起始點座標值，為：「30,60」，然後重複執行上述步驟，直到所有的座標值都讀取完畢，才執行底下步驟。
- Step 14: 將得到的兩軸座標序列，根據所選擇的資料隱藏演算法與前置處理流程，進行資料嵌入的動作。

3.3 資料擷取流程

本研究之資料擷取流程如下圖 3-13 所示。

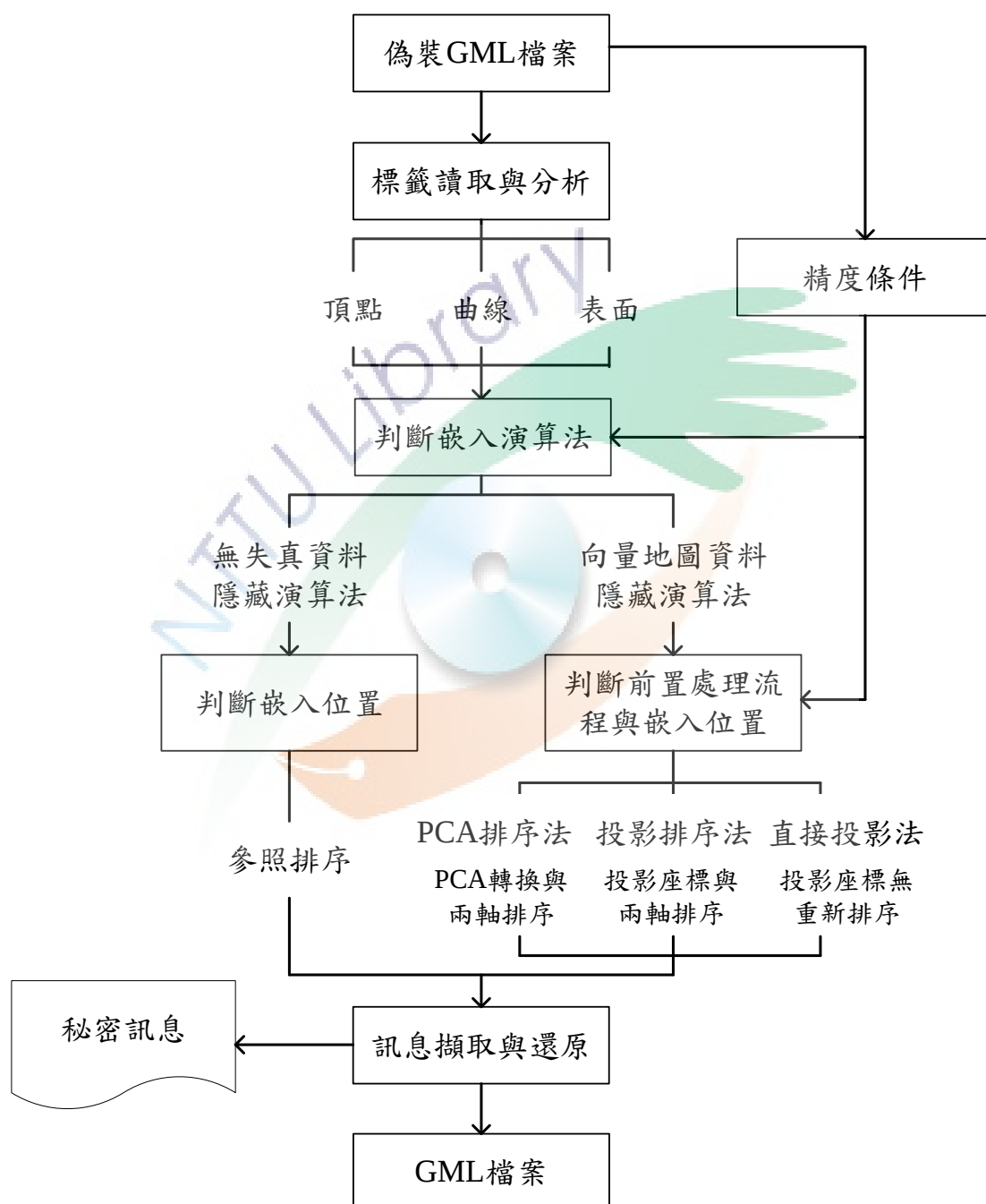


圖 3-13：資料擷取流程

首先開啟偽裝 GML 檔案，除了擷取標籤進行分析外，並記錄該地圖資料的精度條件。然後擷取「頂點」、「曲線」與「表面」三種類型的屬性資料後，根據 GML 檔案的精度條件、嵌入演算法與嵌入位置參數來判斷嵌入資料時使用的資料隱藏演算法、前置處理的流程與嵌入資料的圖徵類型，最後進行擷取訊息與還原的動作，得到秘密訊息與 GML 檔案。



第四章 研究結果與分析

本章節將提出地理標記語言之資料隱藏技術的實驗結果與分析，首先第一節說明本研究實驗設計，其次在第二節說明本研究開發的系統介面，並於後續章節分別說明地理標記語言之資料隱藏技術在可嵌入訊息量、不可察覺性、可逆性與執行時間等評估指標上的結果。

4.1 實驗設計

本研究實驗的硬體環境為個人電腦，作業系統為 Microsoft Windows 7，搭配 Intel Core 2Duo CPU T5750 2.0GHz 中央處理器以及 3 GB 的記憶體。程式的開發語言為 JAVA，在單機環境下測試，實驗的地圖資料格式為 GML 檔案格式，共有 1 個 GML 檔案。GML 檔案共有 65,828 個頂點，頂點座標的精度到小數點下第 6 位，另外 GML 檔案的基本資料如下表 4-1 所示，其顯示的圖形如下圖 4-1 所示。

表 4-1：GML 檔案基本資料

檔案名稱	比例尺	頂點個數(不重複)
Taiwan Village	1:1,000,000	65,828

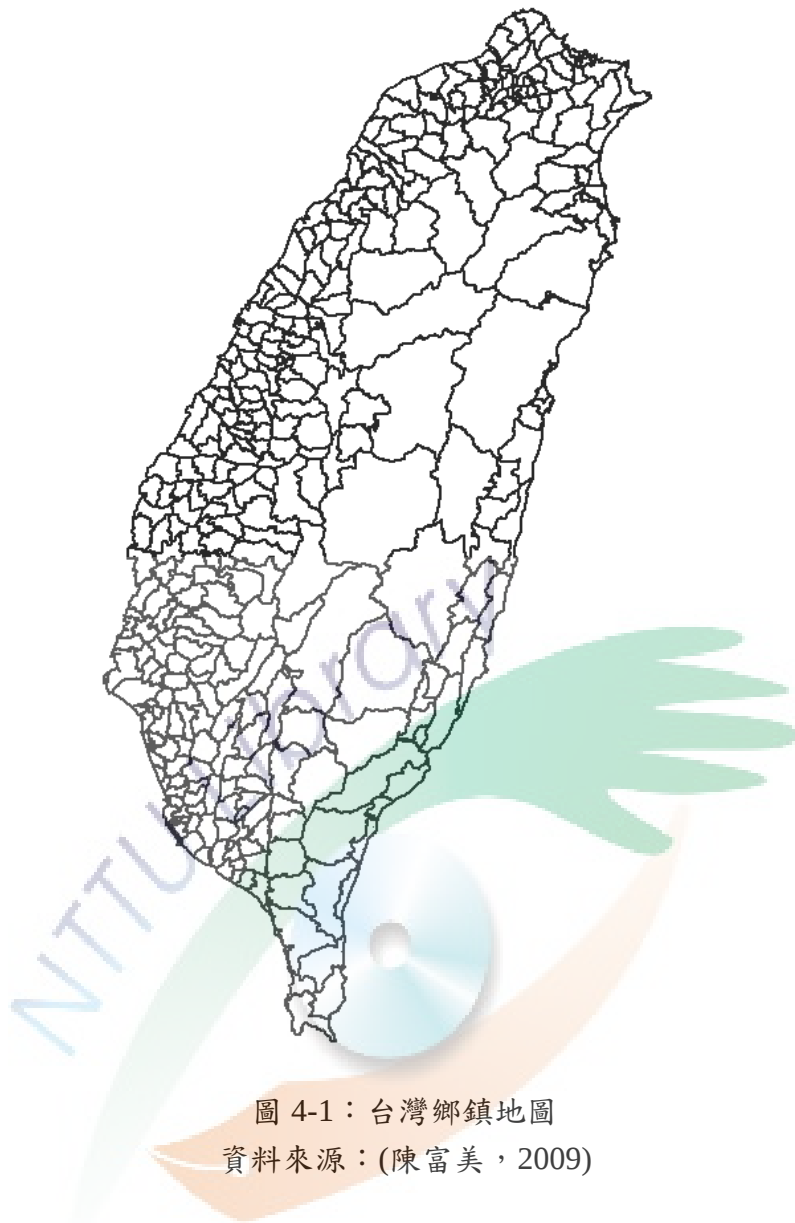


圖 4-1：台灣鄉鎮地圖
資料來源：(陳富美，2009)

用來評估不同資料隱藏技術導入 GML 空間資料格式之成效的指標包括：可嵌入訊息量(Capacity)、不可察覺性(Imperceptibility)、可逆性(Reversibility)與執行時間(Execute Time)。

首先可嵌入訊息量越大，代表 GML 檔案可以隱藏的訊息量越多，可應用性也越高；其次不可察覺性的部分，其數值越低，代表掩護 GML 圖徵與偽裝 GML 圖徵在幾何上的差異越小，可應用性與秘密訊息的安全性也越高；可逆性的部分，其數值越低，表示偽裝 GML 圖徵與掩護 GML 圖徵相似程度越高，代表還

原 GML 檔案後，該 GML 檔案可以再利用，可應用性也越高；最後執行時間的部分，時間越低代表效率越好。

其中不可察覺性是以 RMSE(Root Mean Square Error)當作評估變形量的指標，如下式(4-1)。但由於 GML 檔案之地圖大小不相同，因此將得到的 RMSE 數值除以邊界方塊 (Bounding Box)的對角線長，再取百分比做為評估變形量的指標，如下式(4-2)。

$$RMSE = \sqrt{\sum_{i=1}^n (\bar{X}_i - X_i)^2 + (\bar{Y}_i - Y_i)^2 / n} \quad (4-1)$$

$$RMSE \text{ ratio} = (RMSE / \text{對角線長}) \times 100\% \quad (4-2)$$

4.2 系統介紹

本研究之系統介面如下圖 4-2 所示，資料嵌入介面如下圖 4-3 所示，資料擷取介面如下圖 4-4 所示，最後軟體架構如下圖 4-5 所示。

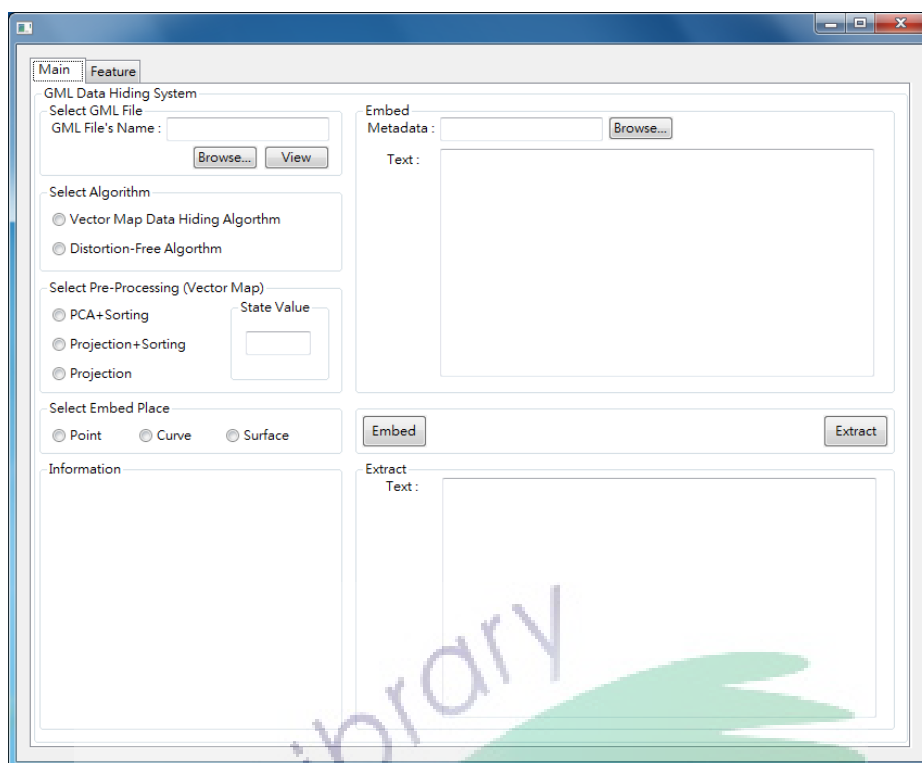


圖 4-2：系統介面

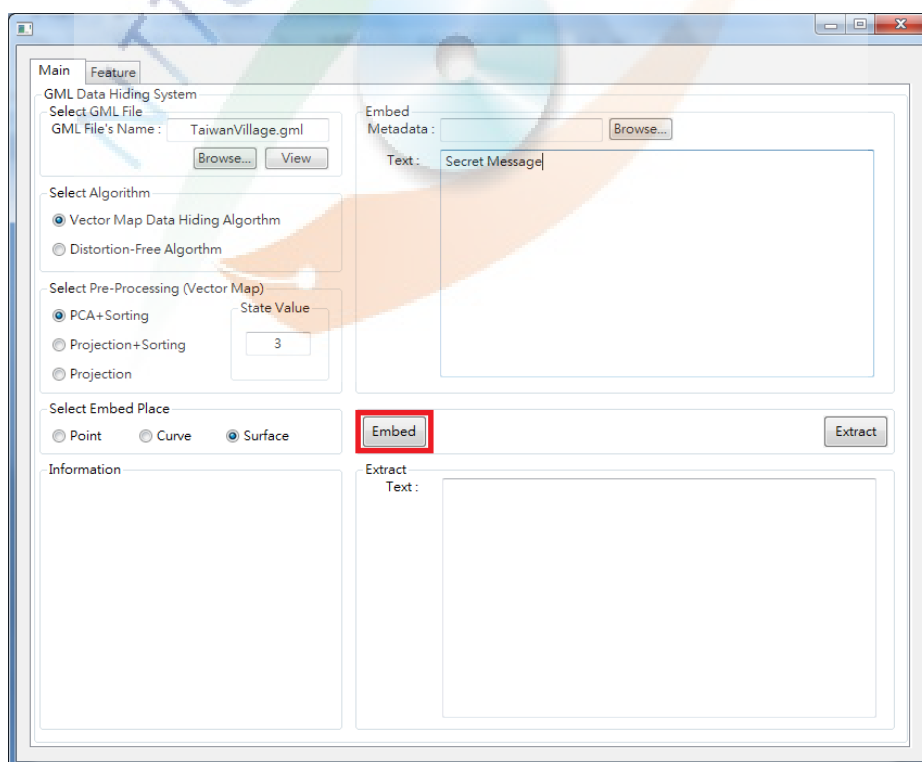


圖 4-3：資料嵌入介面

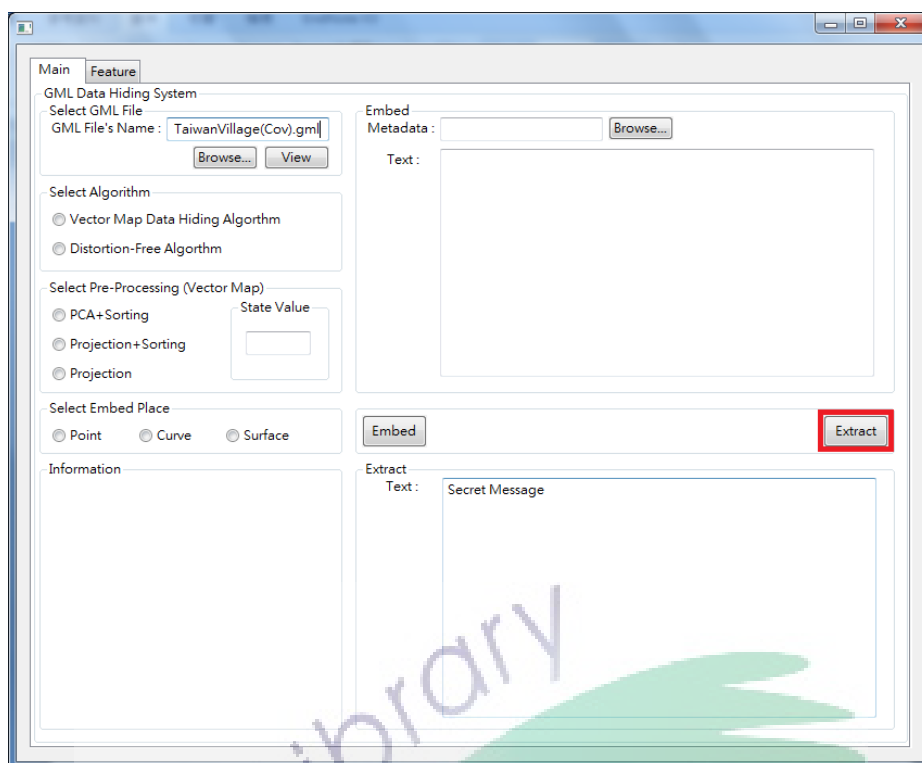


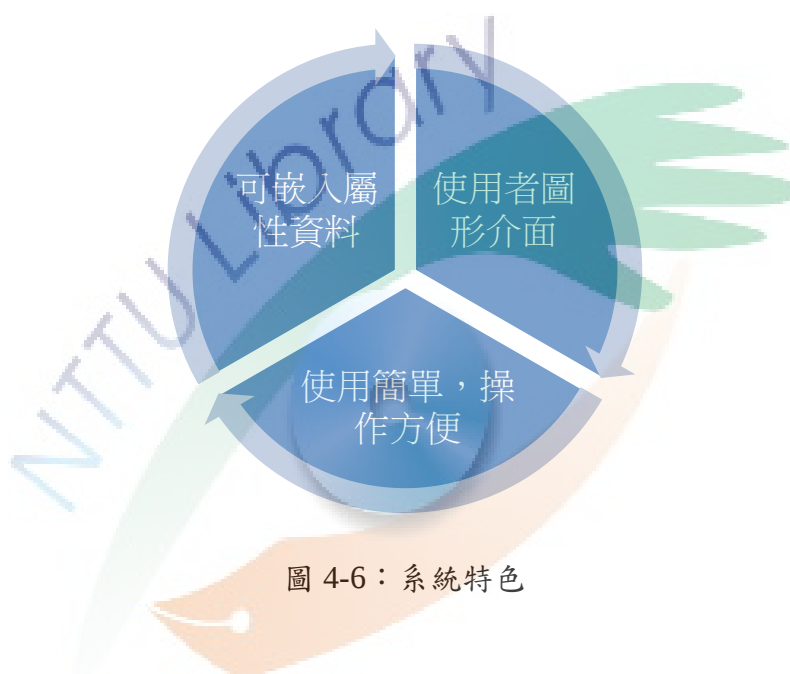
圖 4-4：資料擷取介面

地理標記語言之資料隱藏技術			
Eclipse Java IDE			
JavaDBF		JMathTools	
Jigloo GUI Builder (SWT)			
I/O			
Java 虛擬機器			
Windows	Linux	Solaris	Other

圖 4-5：軟體架構

參考資料來源：(劉秋良，2008)

本研究建立使用者圖形介面，讓使用者可選擇要進行資料隱藏動作的 GML 檔案與要嵌入的秘密訊息或詮釋資料，並可根據需求選擇要使用的資料隱藏演算法、前置處理流程與嵌入資料的位置，最後產生偽裝的 GML 檔案。當要擷取訊息時，僅需選擇要讀取的偽裝 GML 檔案，按下擷取的按鈕，即可擷取秘密訊息，同時將偽裝 GML 檔案還原。此應用機制不但可用於保護秘密訊息、詮釋資料或 GML 屬性資料，更可做為 GML 檔案的認證簽章，因此增加了地理標記語言之資料隱藏技術的可應用性，本系統的特色如下圖 4-6 所示。



- 使用者圖形介面

所有的地理標記語言之資料隱藏技術操作，包含嵌入訊息、擷取訊息、還原檔案與呈現地圖均可在一個圖形介面上完成，同時透過嵌入資料的方式，達到保護秘密訊息或檔案認證的目的，增加使用者使用資料隱藏技術的意願與資料隱藏技術的可應用性。

- 使用簡單，操作方便

僅需透過一個簡單步驟，即可將秘密訊息或詮釋資料嵌入到 GML 檔案格式中所選擇的位置，並新增一個偽裝的 GML 檔案。另外從偽裝的

GML 檔案中擷取資料時，僅需按一個按鈕，即可取得秘密訊息，同時將 GML 檔案還原。

- 可嵌入屬性資料

除了秘密訊息與詮釋資料外，更可將 GML 檔案中之屬性資料嵌入 GML 檔案之空間資料格式中，當從偽裝的 GML 檔案擷取完資料後，即新增一個完整的 GML 檔案。

4.2 可嵌入訊息量(Capacity)評估

在可嵌入資料量的部分，因為利用 GML 檔案標籤語法的特性，未捨棄相同的頂點座標值，所以可嵌入較多的訊息，詳細資料如下表 4-2 與圖 4-7 所示。

表 4-2：可嵌入訊息量

資料隱藏演算法	可嵌入訊息量	
	本研究	原研究
可擴展之可逆式向量地圖資料隱藏演算法		
s = 1	197,480	131,652
s = 2	394,960	263,304
s = 3	592,440	394,956
s = 4	798,920	526,608
s = 5	987,400	658,260
無失真資料隱藏演算法	976,374	650,893

單位：位元(bits)

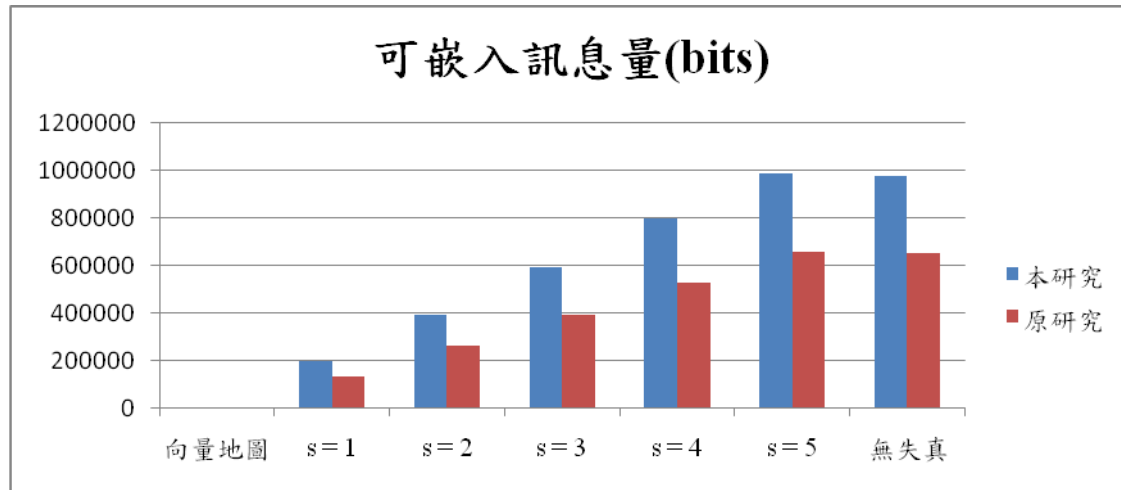


圖 4-7：可嵌入訊息量

本研究之資料隱藏技術所使用的資料隱藏演算法，其可嵌入的訊息量均受到頂點個數多寡影響，端點個數越多，則可嵌入越多訊息。在資料嵌入頂點與曲線位置的部分與原研究結果相同，可擴展之可逆式向量地圖資料隱藏演算法的可嵌入訊息量為 $2 \times (n-2) \times s$ ，無失真向量地圖資料隱藏演算法的可嵌入訊息量為 $\log_2(n!) - n + 1$ 。

但在嵌入表面位置的部分，因 GML 檔案儲存多邊形資料時，可將多邊形分成多個子多邊形來儲存，使重複的端點數增加，與可擴展之可逆式向量地圖資料隱藏演算法將重複的頂點座標捨棄不用不同，透過 GML 標籤語法的特性，讓不同子多邊形可分開進行資料嵌入的動作，使可嵌入的資料量增加，所以當子多邊形重疊的邊越多時，代表可嵌入訊息的端點數增加，因此利用 GML 空間資料格式，可充分利用所有的座標值，嵌入更多的訊息。而且在可擴展之可逆式向量地圖資料隱藏演算法中多了分割變數 s ，透過設定分割變數來調節一個區間的可嵌入訊息量，當分割變數越大時，可嵌入越多的訊息，藉此提供較為彈性的選擇。

因此在嵌入一般的訊息時，可選擇將訊息優先嵌入頂點或曲線的位置，但當

需要嵌入大量訊息時，則將訊息嵌入表面的部分，同時視情況設定分割變數 s ，藉此嵌入大量訊息，使地理標記語言之資料隱藏技術在應用上更加靈活。

4.3 不可察覺性(Imperceptibility)評估

在不可察覺性的部分，因在進行嵌入資料時，會改變頂點的座標值或是排列順序，造成與原頂點座標值的差異。本研究利用 RMSE ratio 當作評估變形量的指標，但因無失真資料隱藏演算法採用的方法是改變頂點的排列順序，並沒有改變頂點的座標值，其 RMSE ratio 的值均為 0，因此底下探討本研究針對可擴展之可逆式向量地圖資料隱藏演算法所提出的三種前置處理流程，在變形量上的差異，如下表 4-3 與圖 4-8 所示。

表 4-3：嵌入資料之 RMSE Ratio

s 值	向量地圖可擴展之可逆式向量地圖資料隱藏演算法		
	PCA 排序法	投影排序法	直接投影法
1	0.0000223411	0.0000230388	0.0001382328
2	0.0000322013	0.000032888	0.0001874616
3	0.0000381767	0.0000388997	0.00024895808
4	0.000041417	0.0000421684	0.00025722724
5	0.0000430995	0.0000438674	0.0002632044

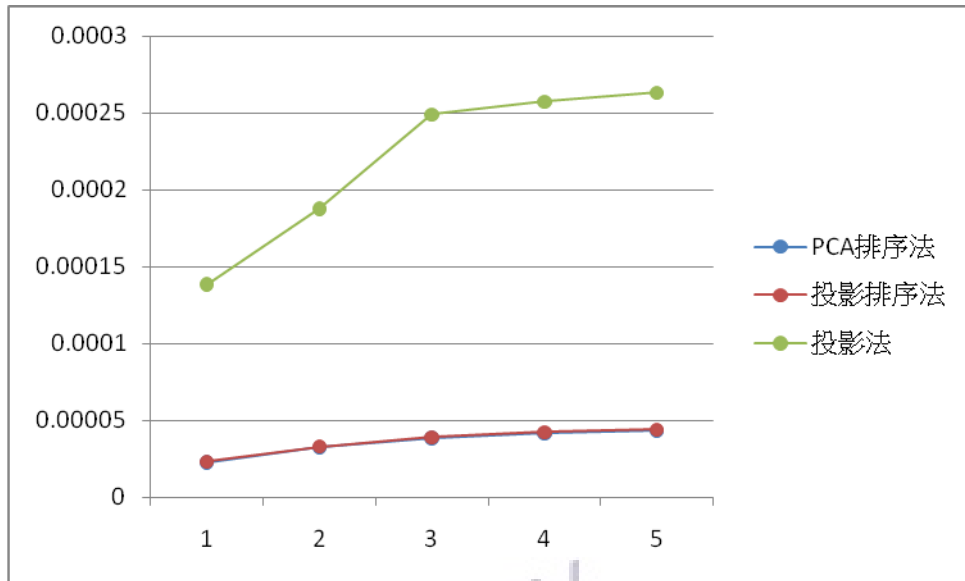


圖 4-8：嵌入資料之 RMSE Ratio

在可擴展之可逆式向量地圖資料隱藏演算法中，前置處理「PCA 排序法」與「投影排序法」的 RMSE Ratio 值近乎相同，同時與可擴展之可逆式向量地圖資料隱藏演算法的研究結果差異不大，代表在此比例尺(1:1,000,000)的 GML 地圖上，是否使用 PCA 轉換對於嵌入資料時端點座標的改變沒有顯著的差異，但「直接投影法」的部分，因為所使用的端點座標值未經大小排序的轉換，因此要嵌入資料的座標值與鄰近的座標值，其距離並非最短，因此造成 RMSE Ratio 值明顯上升，示意圖如下圖 4-9 所示，距離之正負值代表方向，因此未排序的數列之總差距為 55，但排序後的數列之總差距為 40，相差 15。

數列 a：	30, 20, 10, 15, 20, 35, 45, 50, 50, 45	(未排序)
距離：	-10, -10, 5, 5, 15, 10, 5, 0, -5	
數列 a'：	10, 15, 20, 20, 30, 35, 45, 45, 50, 50	(排序後)
距離：	5, 5, 0, 10, 5, 10, 0, 5, 0	

圖 4-9：座標值與其鄰近座標值之距離

同時本研究發現，雖然分割變數 s 越大時可嵌入越多訊息，但同時產生的變形量也越大，但還在可容許的誤差範圍內。因此在高精度與高強韌度需求的 GIS 應用上，若要嵌入大量的秘密訊息，可選用前置處理「PCA 排序法」與「投影排序法」，而前置處理「直接投影法」，因嵌入資料後之 RMSE ratio 值較高，所以適合嵌入小量的秘密訊息與應用於低精度需求的 GIS 服務中。

4.4 可逆性(Reversibility)評估

在可逆性的部分，在擷取完資料後，需將座標值還原回嵌入資料前的數值，從可逆程度的好壞可以判斷該資料隱藏技術的應用性之高低。本研究也利用 RMSE ratio 當作評估還原程度的指標，但因無失真資料隱藏演算法是透過改變頂點的排列順序嵌入訊息，並沒有改變頂點的座標值，還原時也僅是改變頂點的排列順序，故 RMSE ratio 的值為 0，因此底下探討本研究針對可擴展之可逆式向量地圖資料隱藏演算法所提出的三種前置處理流程，在可逆性上的差異，如下表 4-4 與圖 4-10 所示。

表 4-4：還原資料之 RMSE Ratio

s 值	可擴展之可逆式向量地圖資料隱藏演算法		
	PCA 排序法	投影排序法	直接投影法
1	1.72E-10	1.75E-10	5.34E-9
2	5.61E-10	5.83E-10	4.98E-8
3	2.27E-8	2.40E-8	1.71E-7
4	6.73E-8	6.77E-8	9.34E-7
5	7.56E-7	7.61E-7	4.29E-6

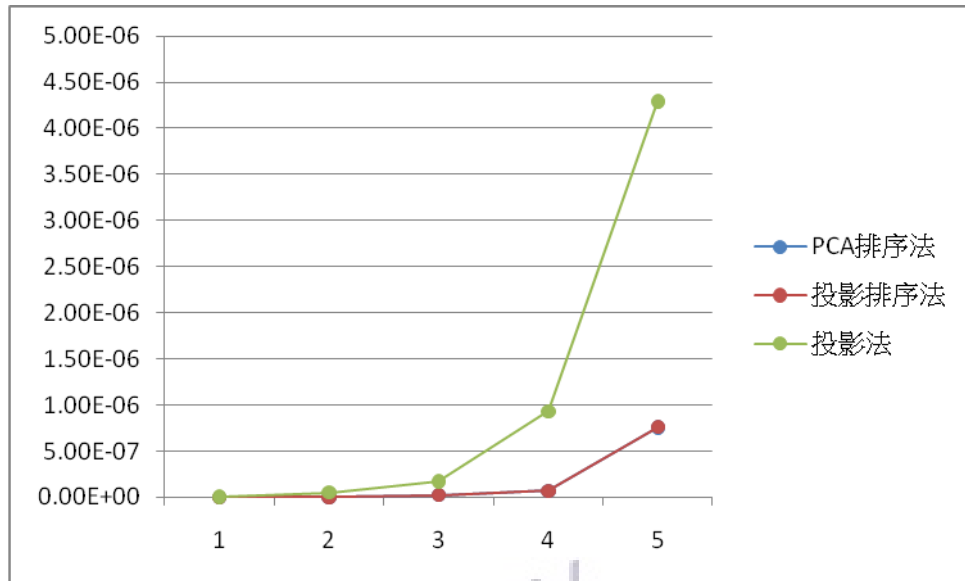


圖 4-10：還原資料之 RMSE Ratio

在可擴展之可逆式向量地圖資料隱藏演算法的部分，前置處理「PCA 排序法」與「投影排序法」的 RMSE Ratio 值近乎相同，代表是否使用 PCA 轉換對於嵌入資料時端點座標的改變沒有顯著的差異，但「直接投影法」的部分，因為所使用的座標值未經大小排序的轉換，因此 RMSE Ratio 值同樣偏高。同時本研究發現，當分割變數 s 越大，區間內切割出越多子區間，因此可能造成部分差異值擴大，導致嵌入與還原之 RMSE Ratio 值變高，如下圖 4-11 所示。

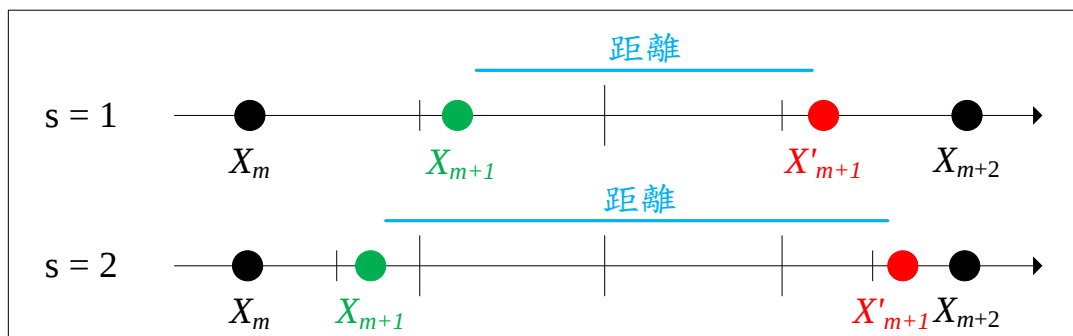


圖 4-11：分割變數影響座標值差異

假設嵌入資料時為最遠距離，當分割變數 s 為 1，區間內 X_{m+1} 由狀態「0」移動到狀態「1」時，最遠距離大約為 X_m 與 X_{m+2} 區間的二分之一，但當分割變數 s 為 2，區間內 X_{m+1} 由狀態「00」移動到狀態「11」時，最遠距離大約為 X_m 與 X_{m+2} 區間的四分之三，相差四分之一，因此分割變數 s 越高時，則座標值之距離就越大。

最後可擴展之可逆式向量地圖資料隱藏演算法在可逆性的部分，三種前置處理流程在還原後與原資料的 RMSE ratio 值均小於 10 的負 5 次方，因此有很高的可逆性，代表可實際應用於一般 GIS 上。

4.5 執行時間(Execute Time)評估

最後執行時間的部分，嵌入資料與擷取資料的所需時間如下表 4-5 與圖 4-12 所示，端點個數在 7 萬點的情形下，可擴展之可逆式向量地圖資料隱藏演算法的嵌入資料所需時間最多不超過 30 秒，擷取資料所需的時間不超過 10 秒；無失真資料隱藏演算法的嵌入資料所需的時間最多不超過 30 秒，擷取資料所需的時間不超過 30 秒。

表 4-5：執行時間

端點個數	可擴展之可逆式向量地圖		無失真	
	嵌入	擷取	嵌入	擷取
10,000	2.8	1.9	4.0	4.2
20,000	5.1	3.7	8.6	8.6
40,000	16.7	6.3	17.1	16.9
70,000	28.3	9.4	28.5	28.6

單位：秒(second)

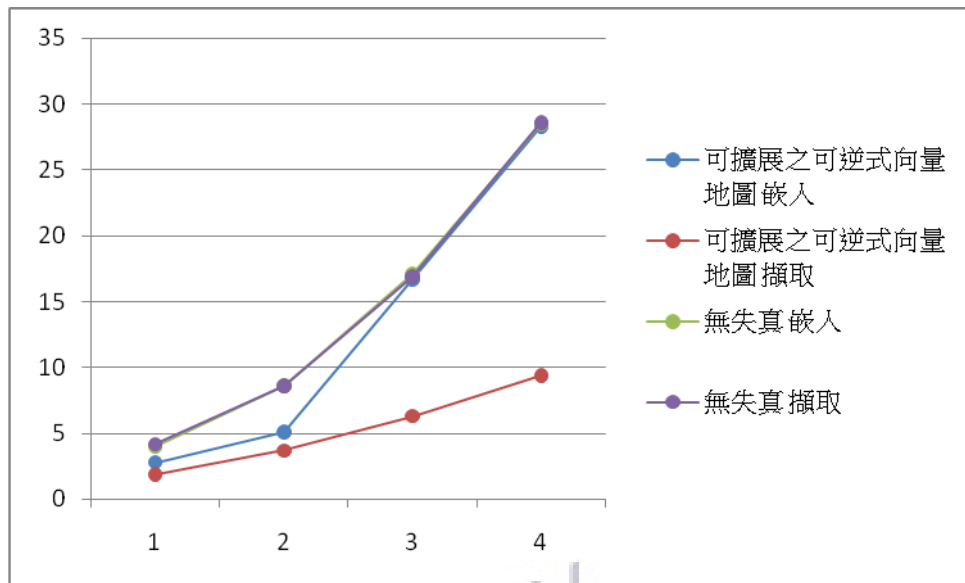


圖 4-12：執行時間

本研究發現可擴展之可逆式向量地圖資料隱藏演算法在嵌入資料所花的時間比擷取資料所花的時間還要多，乃因為嵌入區間與嵌入位置是在嵌入資料時決定，因此需要多一點時間計算，而無失真資料隱藏演算法因嵌入與擷取之時間複雜度相同，因此所需的時間差不多。另外可擴展之可逆式向量地圖資料隱藏演算法與無失真資料隱藏演算法的時間複雜度均為 $O(n)$ ，代表在執行時間上沒有太大的差異，但在可嵌入訊息量的部分，可擴展之可逆式向量地圖資料隱藏演算法可嵌入較多訊息。

第五章 結論與未來研究建議

5.1 結論

本研究針對可擴展之可逆式向量地圖資料隱藏演算法提出的三種不同前置處理流程，與可擴展之可逆式向量地圖資料隱藏演算法以及無失真資料隱藏演算法的研究結果比較，如下表 5-1 所示。

表 5-1：研究結果比較

評估指標	本研究			可擴展之可逆式向量地圖資料隱藏演算法	無失真資料隱藏演算法
	PCA 排序法	投影排序法	直接投影法		
可嵌入訊息量	較佳	較佳	較佳	佳	較差
不可察覺性	佳	佳	較差	佳	較佳
可逆性	較佳	較佳	較差	較佳	較佳
執行時間	佳	佳	佳	佳	佳

可擴展之可逆式向量地圖資料隱藏演算法導入 GML 檔案在各項評估指標上的表現，首先可嵌入訊息量的部分，因充分利用表面當中多邊形重複的邊，同時可彈性設定要嵌入的位元數，因此可嵌入訊息量較多；其次不可察覺性的部分，三種前置處理流程的 RMSE ratio 值均小於 0.0003，其中因直接投影法未經過重新排序的流程，因此變形量較大，但也發現座標值是否經過 PCA 轉換對於變形量沒有顯著的影響；最後在可逆性的部分，三種前置處理流程的 RMSE ratio 值均小於 10 的負 5 次方，代表可應用性高。

而無失真資料隱藏技術導入 GML 檔案在各項評估指標的表現上，在不可察覺性與可逆性上，因嵌入資料是透過改變頂點的排列順序，因此 RMSE ratio 均為 0，代表不但沒有產生變形量，其可應用性更高。在可嵌入訊息量的部分，相較於向量地圖資料隱藏演算法，其可嵌入量較低，同時較無彈性，同時執行時間較長。

本研究主要針對 GML 檔案提出資料隱藏技術之應用，除了實際將可擴展之可逆式向量地圖資料隱藏演算法(陳富美，2009)與無失真資料隱藏演算法(Bogomjakov, et al., 2008)導入 GML 空間資料格式中之外，更根據不同應用目的、地圖的精度條件與秘密訊息的容量大小來選擇不同的資料隱藏演算法與對應前置處理流程，同時透過標籤語法的特性，將欲嵌入的資料分別嵌入不同的圖徵類型與圖徵當中，以提升 GML 檔案的安全性與可嵌入的資料量，避免秘密訊息、GML 屬性資料或詮釋資料在傳輸時遭到竊取或竄改，最後在擷取完偽裝 GML 檔案的資料後，可將偽裝 GML 檔案還原成原 GML 檔案，增加地理標記語言之資料隱藏技術之可應用性。

透過針對可擴展之可逆式向量地圖資料隱藏演算法提出的三種不同前置處理流程，除了提供高精度與高強韌性、高精度與低強韌性以及低精度與低強韌性需求的 GIS 應用外，更透過無失真資料隱藏演算法，使嵌入資料時不會產生變形量，可應用於不同應用目的之 GIS 服務。其中 PCA 排序法與投影排序法適合嵌入大量訊息，同時可搭配原端點的順序表當作金鑰，使可應用性增加，而直接投影法適合嵌入小量訊息，若嵌入大量訊息時，其變形量較高，可應用性降低。

利用 GML 檔案中的標籤語法，可將訊息嵌入不同的圖徵類型或圖徵，藉此

提升秘密訊息、詮釋資料與 GML 屬性值的安全性、可嵌入的訊息量與 GML 資料隱藏技術之可應用性。同時除了嵌入秘密訊息外，透過使用簡單的使用者圖形介面，更可讓資料隱藏技術成為 GML 檔案的驗證機制，達到類似數位簽章的效果，提供地理標記語言之資料隱藏技術在 GIS 上的創新應用。

5.2 未來研究建議

首先可針對 GML 中圖徵之拓樸屬性，探討導入資訊隱藏技術的方法。因為本研究導入的可擴展之可逆式向量地圖與無失真資料隱藏演算法，均架構於幾何上，同時幾何與拓樸在應用上的著重點不同，所以本研究未探討拓樸屬性的資料在資料隱藏技術方面的應用，因此後續研究可針對 GML 中圖徵之拓樸屬性，找出適合導入的資料隱藏演算法，藉此增加 GML 檔案中資料隱藏技術可應用的元素，以符合更多 GIS 應用上的需求。

其次可根據 GML 檔案格式，提出專門用於 GML 空間資料格式之資料隱藏演算法。由於本研究是導入其他學者所提出的資料隱藏演算法，為了增加應用上的方便，是根據其特性來選擇要使用的資料隱藏演算法、相對應的前置處理流程與要嵌入資料位置。但該嵌入與擷取資料的流程，並未完全使用到 GML 檔案中所有可用的元素，同時無法完善發揮 GML 資料格式的特性。

參考文獻

- 王聖銘、陳富美 (2009)。利用 open api 的技術發展互動式空間資訊服務：國立東華大學鄉土文化學系出版。
- 何燦群 (2010)。Google earth 創新應用。在國土資訊系統與民間資訊整合運用發展趨勢系列研討會。
- 林信助 (2008)。開放式時空地理資料描述基本架構之研究。國立成功大學測量及空間資訊學系碩士論文，台南市。
- 林峰田 (2006)。開放式地理資訊系統於地方政府的應用-以台北市政府為例，開放式地理資訊系統暨網路地理資訊系統研討會。
- 陳勇仁 (2007)。以 gml/xml 為基礎建置一套駕駛路徑資訊系統。靜宜大學資訊管理學系碩士論文，台中縣。
- 陳凱威 (2007)。向量地圖資料隱藏演算法。國立中興大學資訊科學系所碩士論文，台中市。
- 陳凱威、王聖銘、張聖杰、王宗銘、鄭友銘、蔡淵裕 等 (2007)。向量地圖資料隱藏演算法。興大工程學刊，13 (2)，111-124。
- 陳凱威、王聖銘、鄭友銘、王宗銘 (2007)。轉換域上的數位地圖資料隱藏演算法。資訊安全通訊，13 (1)，112-130。
- 陳富美 (2009)。可逆式資訊隱藏技術應用於向量地圖後設資料嵌入之研究。國立東華大學企業管理學系博士論文，花蓮縣。
- 劉秋良 (2008)。發展可逆式向量地圖資訊隱藏系統。國立東華大學企業管理學系碩士論文，花蓮縣。
- 鄧東波 (2004)。Gml 原理概述及應用。國土資訊系統交流共享相關標準制度規劃建置作業教育訓練講義。
- 鄧東波 (2006)。Gml 原理概述及其應用。國土資訊系統基礎人才研討會課程講義。
- Bogomjakov, A., Gotsman, C., & Isenburg, M. (2008). Distortion-free steganography for polygonal meshes. *The Eurographics Association and Blackwell Publishing*, 27(2), 637-642.
- Chang, C.-C., & Tseng, H.-W. (2009). Data Hiding in Images by Hybrid LSB Substitution. *IEEE Computer Society*.
- Chao, M.-W., Lin, C.-H., Yu, C.-W., & Lee, T.-Y. (2009). A High Capacity 3D Steganography Algorithm. *IEEE Transactions on Visualization and Computer Graphics*, 15(2), 274-284.
- Hong, W., Chen, T.-S., & Shiu, C.-W. (2008). Reversible Data Hiding Based on Histogram Shifting of Prediction Errors. *2008 International Workshop on Education Technology and Training & 2008 International Workshop on*

- Geoscience and Remote Sensing*, 2, 578-581.
- Kahn, D. (1996). The history of steganography. *Lecture Notes In Computer Science: Proceedings of the First International Workshop on Information Hiding*, 1174, 1-5.
- Niu, X. M., Shao, C. Y., & Wang, X. T. (2006). A Survey of Digital Vector Map Watermarking. *ICIC International*, 2(6), 1301-1316.
- Pfitzmann, B. (1996). Information Hiding Terminology-Results of an Informal Plenary Meeting and Additional Proposals. *Lecture Notes In Computer Science: Proceedings of the First International Workshop on Information Hiding*, 1174, 347-350.
- Popa, R. (1998). *An Analysis of Steganographic Techniques*. Universitatea POLITEHNICA din Bucuresti Department of Computer Science and Software Engineering, Timisoara.
- Rabah, K. (2004). Steganography-The Art of Hiding Data. *Information Technplogy Journal*, 3(3), 245-269.
- Wang, S.-M., & Chen, F.-M. (2008). The Development of a Ubiquitous Geographic Information Service by Using Service-Oriented Architecture. *Proceeding of 2008 IEEE Asia-Pacific Services Computing Conference*.
- Wang, S.-M., Chen, F.-M., Chen, K.-W., & Yang, W.-P. (2009). Using Reversible Steganography Algorithm to Enbed Metadata in Vector Maps. *Journal of Information Management*, 16(4).
- Wang, X. T., Shao, C. Y., Xu, X. G., & Niu, X. M. (2007). Reversible Data-Hiding Scheme for 2-D Vector Maps Based on Difference Expansion. *IEEE Transactions on Information Forensics and Security*, 2(3), 311-320.