10/14 朝陽科技大學 陳忠信 副教授

A Study of Efficient GNSS Coordinate Classification Strategies for Epidemic Management

Introduction

- Many infectious diseases, such as COVID-19, monkeypox, chickenpox, and influenza, are highly contagious and seriously affect human health, economic activities, education, sports, and leisure.
- Restricting, tracing, and isolating the movement of people (targets) during an epidemic is an effective way to slow its spread.
- Knowing the areas of targets can help allocate medical treatment or protection-related materials or personnel.
- GNSS technology is also very mature and can accurately provide the current geographical coordinates of targets.
- These coordinates can be converted into the areas where the targets are located.
- In this study, we proposed a coordinate classification strategy, which uses the GNSS coordinates of the current targets and KNN technology to classify (predict) the areas where the targets are located.

這場演講主要探討如何利用 GNSS (全球衛星導航系統) 進行座標分類以支援疫情管理在疫情期間，地理位置資料可用來追蹤人員移動、判定風險區域、並進一步協助防疫決策。

研究核心目標在於設計一種高效的座標分類策略，能準確判斷使用者位置是否位於某個疫情風險之內。

相關研究

- PIP (Point-In-Polygon)

  概念：用於判斷一個點是否在多邊形內部。

  主要方法：1. Ray Casting (射線法)
  透過從測試點向任意方向畫一條射線，計算與多邊形邊界的交點數量
  [若交點為奇數 → 點在多邊形內
  [若交點為偶數 → 點在多邊形外

  2. Winding Number (環繞數法)
  根據多邊形邊對測試點的環繞角度判定內外。

  數學特性與複雜度：時間複雜度為 $O(n)$，其中 n 為多邊形的邊數。
  優點是判定準確；缺點是資料量大時效率可能受限

- KNN (K-Nearest Neighbors)

  一種常用於分類的机器學習演算法
  透過訓練資料集中最近的 k 個點來決定測試點的分類

  在 GNSS 資料應用上，可以用於：[根據地理分佈分類不同的地點狀態 (ex: 安全區與風險區)
  [結合 AI 技術進行自動化疫情區域劃分

  優點：直觀、容易實作
  缺點：當样本量很大時，分類速度慢，記憶體需求高

Related Work：Traditional KNN

KNN傳統流程共三步驟：

1. 計算距離：
   - 使用歐氏距離(Euclidean distance)比較測試點與樣本間的距離
   - 時間複雜度：$O(nTD)$，n為資料量、D為維度數
2. 排序距離：
   - 根據距離大小對所有樣本排序
   - 時間複雜度：$O(nTD^2)$
3. 分類預測：
   - 取最近的k個鄰居，依多數決定分類結果
   - 時間複雜度：$O(k)$

Related Work：One-Area and Cell / Rectangle Base PIP

重點說明：

目的，將整個目標區域切割成多個小的網格(cell)，以加快PIP判斷

Cell Allocation 技術：

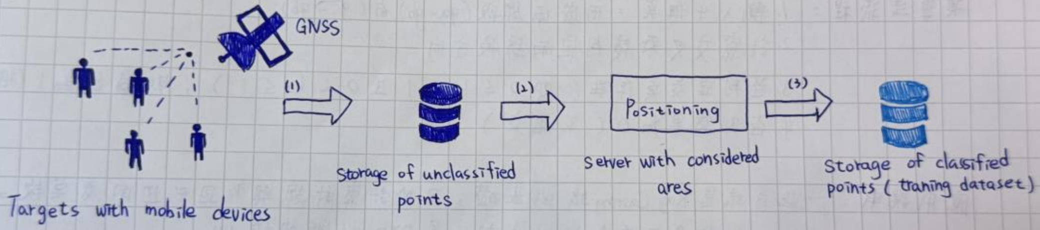
- 內部cell：完全在區域內 → 不需要再檢查
- 交界cell：與邊界相交 → 需進行PIP判斷

優勢：

- 降低演算法的計算負擔
- 僅在必要區域進行判斷，提高效率
- 適合應用於地圖分區與疫情熱區分析

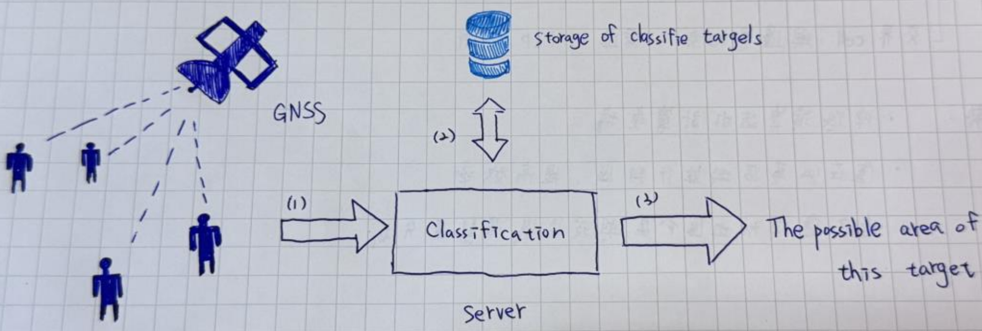
System Model

1. The data sent from the mobile devices of targets will be stored as unclassified geo-graphic points.
2. The sever will individually take out a point from the unclassified points and position which polygon this point is inside.
3. The server will store these geographical points with their classes.



(a). Positioning

1. The real-time data sent from targets' mobile devices arrive to the server.
2. The server extracts candidate points from the storage of classified targets.
3. The server will classify the targets into their located areas according to the candidate points.



(b) Classification

Proposed Strategies : PIP Implementation

主要內容：投影片展示 PIP 實作中的一個關鍵演算法：
SegSegInt (Segment - Segment Intersection)

→ 用於判斷兩線段是否相交

演算法流程：
1. 輸入 4 個點：形成兩線段 (ga - gb) 和 (gc - gd)
2. 計算交叉乘積判定兩線段方向
3. 若判定交點存在（即 $0 \le k1 \le 1$ 且 $0 \le k2 \le 1$），則輸出為 1（相交）
4. 否則結果為 0（不相交）

應用說明：
- 此方法是 Ray Casting 法的基礎，用於計算射線與多邊形邊的交點數。
- 幫助判斷點是否在多邊形內部，是 PIP 判斷的核心

Proposed Strategy : KNN Classification Implementation

- For KNN classification will make statistics on the class g'_{pc} of g' using $I(\cdot)$, find the class with the largest number, and then assign it to g_{cc}.

$$g_{cc} = \arg_i \left(\max \sum_{g' \in NB} I(g'_{pc} = i) \right)$$

- We also contain the weighting KNN
- The Euclidean distance of two points ga and gb is as follow, where (ga_x , ga_y) is the coordinate value of point ga and (gb_x , gb_y) is the coordinate value of point gb.

$$d(ga, gb) = \sqrt{(ga_x - gb_x)^2 - (ga_y - gb_y)^2}$$

- The weighting KNN is

$$g_{cc} = \arg_i \left(\max \sum_{g' \in NB} I(g'_{pc} = i) \times d(g, g')^{-1} \right)$$

Proposed Strategy

- Algorithm 4 employs the technology of the weighting KNN classification for classifying points into ares.
- In addition, Step 1 of this algorithm calculates the candidates of k neighbors based on a numerical value r. When necessary, the value of r will be adaptively adjusted until the number of candidates is greater than or equal to k.
- So, the candidates of k neighbors in Steps 2. 3, and 4 are k or slightly more than k data points, not the total training dataset.
- In this way, we improve the classification time.
- property 1. Given a point g and a polygon set P_A with size m, if point g is inside on of set P_A, Algorithm PtPos positions point g in $O(m \times n_{max})$ time, where n_{max} is polygon's larget edge number of this polygon set.
- Property 2: Given a point g and a training dataset T with size n_T, algorithm AdaptKNN classifies point g in $O(n_T)$ time.

Conclusions

- In this paper, we have planned a strategy, including positioning and classification phases, which can be used when epidemic management or other applications need to track the location of some targets or people.
- We hope this research can help epidemic management understand the spread of these pathogens and enable us to make predictions and preparations earlier, significantly as the infection numbers rapidly increase.