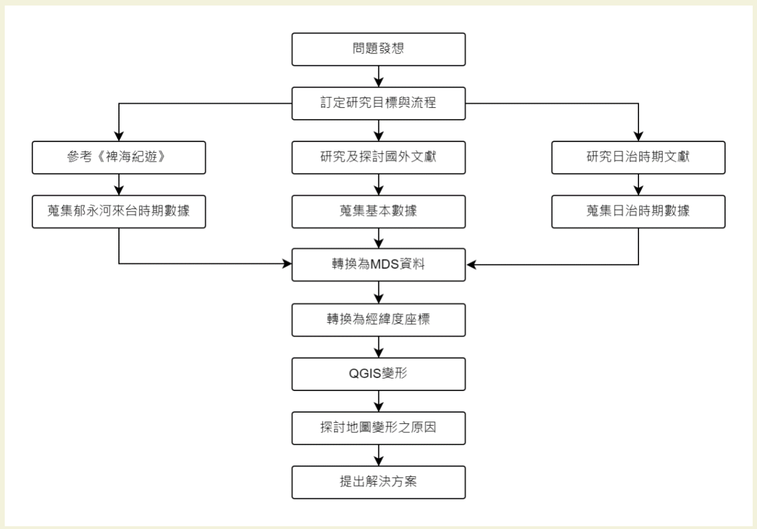
Taiwan Time Space Map

台灣時空地圖研究與製作

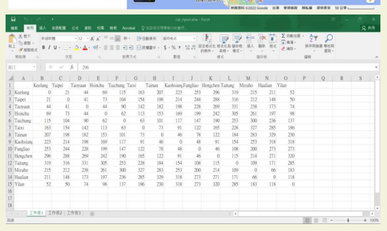
洪浩恩、鄭詠育

1.研究流程



2.地圖取樣點與時空紀錄

1) 14個取樣點分別為主要城市：基隆、台北、桃園、新竹、台中、台南、高雄、台東、花蓮、宜蘭與維持地圖變形後的台灣輪廓形狀用的：台西、枋寮、恆春、瑞穗。且取點以當地政府為代表：如台北市政府、枋寮鄉公所等。
2) 資料查詢使用Google Maps來進行取得。
3) 資料紀錄使用Excel來進行記錄。



3.Multidimensinal Scaling(MDS) 之運算

1) Multidimensinal Scaling 就是多維尺度分析工具，能保持地圖架構完整不被破壞，以保持拓樸學（Topology）上的一致性與穩定性。
2) 我們以 Multidimensinal Scaling 中的最小平方法來決定地點的位置。

4.MDS 的運作原理與座標轉換

利用 Google Map 提供的資訊：譬如 i 地至 j 地，最快的交通時間 t 為資料來源，而後利用 Vscode Python 所支援的MDS函式程式進行資料處理

1) 若 i 地的座標為(p,q)，則兩地距離為：

$$d_{ij} = \sqrt{(p_i - p_j)^2 + (q_i - q_j)^2}$$

2) 任意距離d與對應的交通時間的誤差要越小越好，若時空地圖共取M個地點，則可用最小平方法決定這M個點的座標(p,q)，算式如：

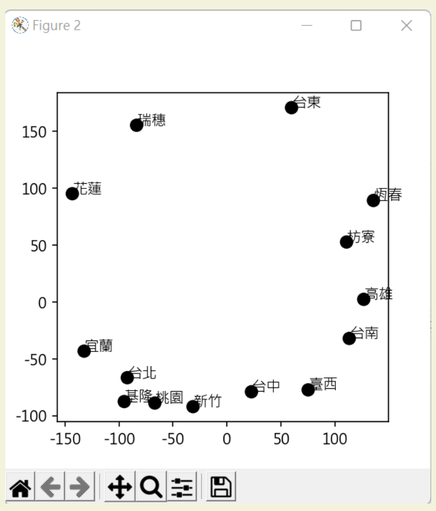
$$\min_{(p,q)} \sum_{i=0}^{M} \sum_{j>i}^{M} (d_{ij} - t_{ij})^2$$

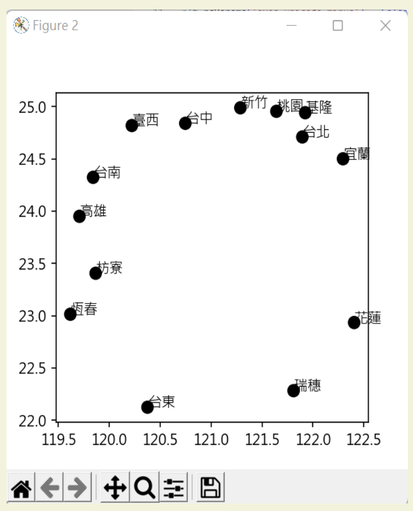
3) 運用 Interative Method (疊代法) 重複計算來的到最小的插植平方和，並利用 Skerlan 所提供的 Python 函式程式 MDS 來完成。如下圖。

```python
from csv import reader
from matplotlib.offsetbox import OffsetImage, AnnotationBbox
from sklearn.metrics.pairwise import manhattan_distances, euclidean_distances
from sklearn.manifold import MDS
from matplotlib import pyplot as plt
import sklearn.datasets as dt
import seaborn as sns
import numpy as np
from matplotlib.font_manager import FontProperties
plt.rcParams['font.sans-serif'] = ['Microsoft JhengHei']
plt.rcParams['axes.unicode_minus'] = False


path = "檔案路徑"
with open(path, 'r') as csv_file:
    csv_reader = reader(csv_file)
    list_of_rows = list(csv_reader)
    for i in range(len(list_of_rows)):
        list_of_rows[i] = list(map(float, list_of_rows[i]))
    X = np.array(list_of_rows)


mds = MDS(dissimilarity='precomputed', random_state=0)
X_transform_1i = mds.fit_transform(X)
colors = ['black','black','black','black','black','black','black','black','black','black','black','black','black','black']
size = [60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60]
fig = plt.figure(2, (8, 3))
cityName = ["基隆","台北","桃園","新竹","台中","臺西","台南","高雄","枋寮","恆春","台東","瑞穗",
            "花蓮","宜蘭"]
plt.scatter(X_transform_1i[:, 0], X_transform_1i[:, 1], s=size, c=colors)
textHere = np.array(X_transform_1i[:, 0], X_transform_1i[:, 1])
print(np.size(textHere, 1))
for j in range(textHere.shape[1]):
    plt.text(textHere[0][j], textHere[1][j],
             cityName[j], fontsize=10)


print(X_transform_1i)
plt.show()
```

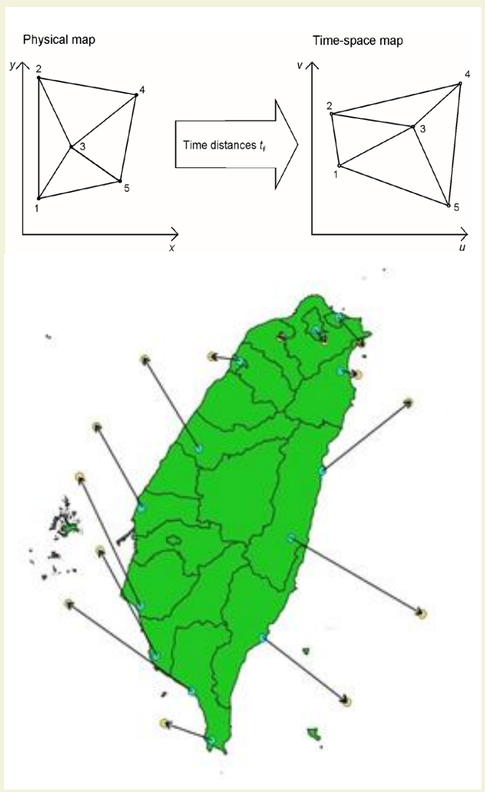
4) 由於MDS透過下列算式將時空地圖座標(p,q)轉換成經緯度座標(u,v)，算式如下：

$$u = p/100 + 120.9738819$$
$$v = q/100 + 23.97565$$





5.地圖變形原理



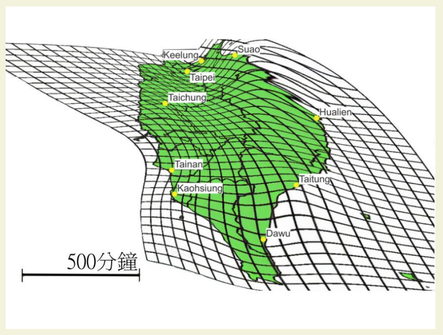
1) 將轉換後的時空地圖座標(u,v)與實際地圖座標(x,y)對位後可以建立轉換關係：

$$u = f(x,y) \qquad v = g(x,y)$$

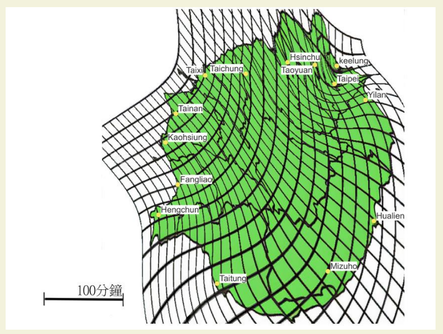
2) 得到轉換關係後就能利用 2-D interpolation (二維內插法)來求出非原先取樣地點的時空地圖座標。
3) 上述二維內插法可利用 QGIS 軟體內之 Georeferencer 工具所提供的薄板曲線(Thin plate spline)。

5.地圖變形實例

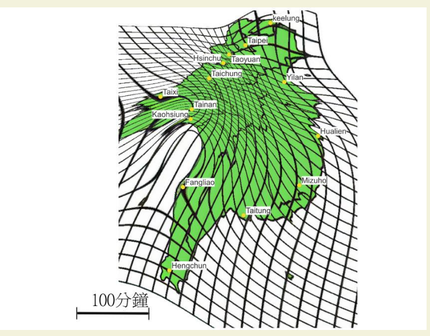
1) 日治時期時空地圖



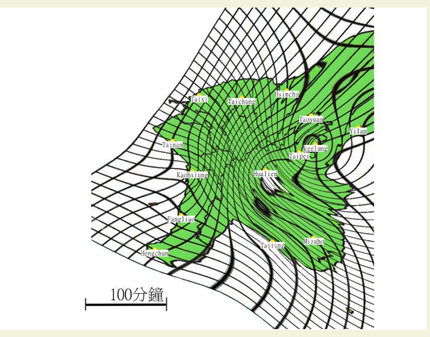
2) 現代公路時空地圖



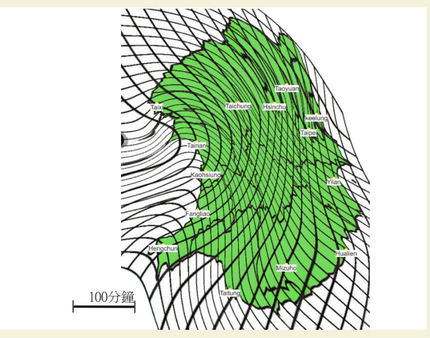
3) 火車節點時空地圖

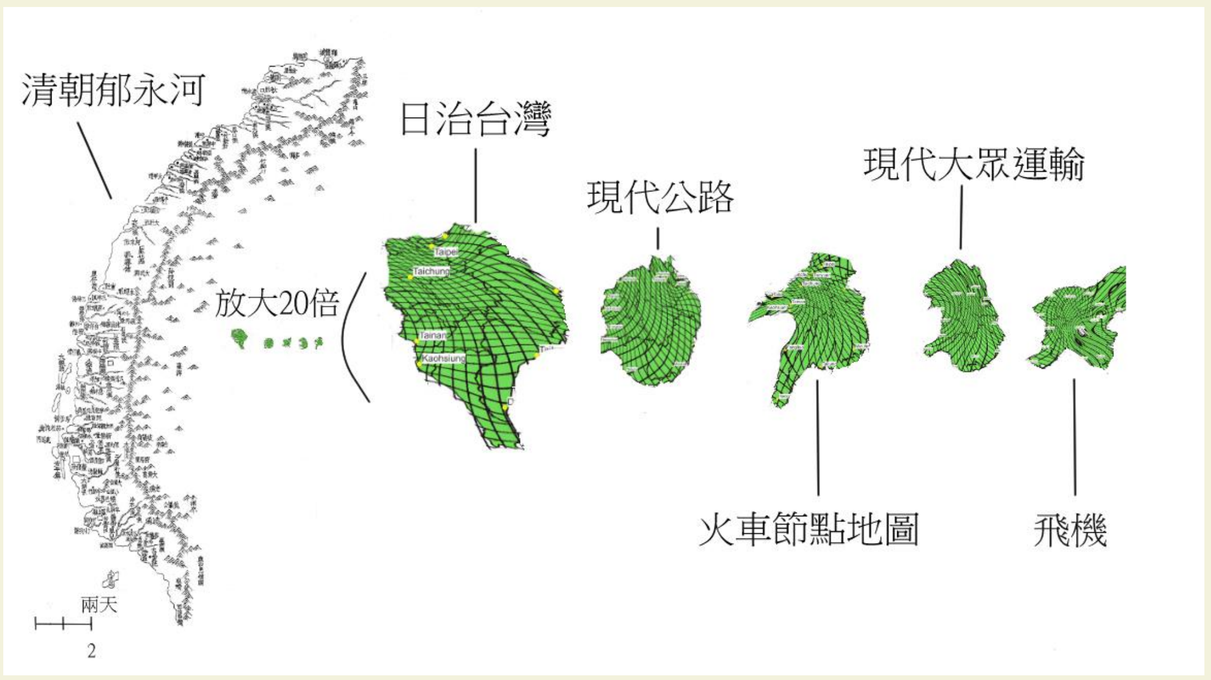


4) 現代大眾運輸時空地圖



5) 飛機時空地圖



7.研究成果與結論

1) 清朝沒有明顯的交通革新，移動以步行為主
2) 不同的交通工具，收縮幅度不同
3) 日治時期因工業革命產生交通革新，交通時間被超大幅縮短
4) 中央山脈對於路上東西向的交通影響極大
5) 不同的交通工具，收縮幅度不同
6) 飛機對於東部交通有顯著提升
7) 高鐵造成西部時空收縮，東部相對時空輻散