

LAB3 結報

(一) merge.py 程式碼說明：

程式碼主要分成兩大部分，步驟一負責將人體骨架進行合併，而步驟二則會將合併後的骨架進行篩選，只留下較完整且可信的數據。

a. 合併重疊骨架

1. num_skeletons表示骨架的數量
2. 用雙重迴圈來逐一檢查每對骨架是否有重疊的部分。
 - For i in range(num_skeletons): 外層迴圈選取一個骨架i。
 - for j in range(i + 1, num_skeletons): 內層迴圈檢查骨架i後面的骨架j，避免重複比較。
3. 如果骨架i或j被標記為已刪除（即subset[x, -1] == -1），則跳過該骨架。
4. common_parts檢查這兩個骨架是否有共同的部位（座標相同），如果有，則認為這些部位是重複的。
5. 如果骨架i和骨架j有重疊部位，我們做以下操作：
 - 用np.maximum取兩者座標的最大值更新subset[i, :-2]，這樣我們可以保留最可靠的座標。
 - 把骨架j的總分和部位數量加到骨架i上，更新它的「總分」和「部位數量」。
 - 將骨架j標記為 -1，表示它已經被合併，不再需要獨立存在。
6. 最後，我們將標記為 -1 的骨架刪除，只保留合併後的骨架。

b. 刪除不符合要求的骨架

1. parts_num: 計算骨架中的部位數量（例如有幾個部位是有效的）。
2. total_score: 計算骨架的總分數。
3. avg_score: 我們用總分數除以部位數量，得到每個部位的平均分數。如果某骨架的部位少於min_num_body_parts或者平均分數小於min_score，它就會被標記為不合格。
4. 我們將不合格的骨架添加到delete_idx列表中，然後利用np.delete一次性把它們刪除。

最終輸出會將篩選後的骨架集合顯示出來，以方便我檢查剩下的骨架是否符合我們的要求。

(二) output subset 結果

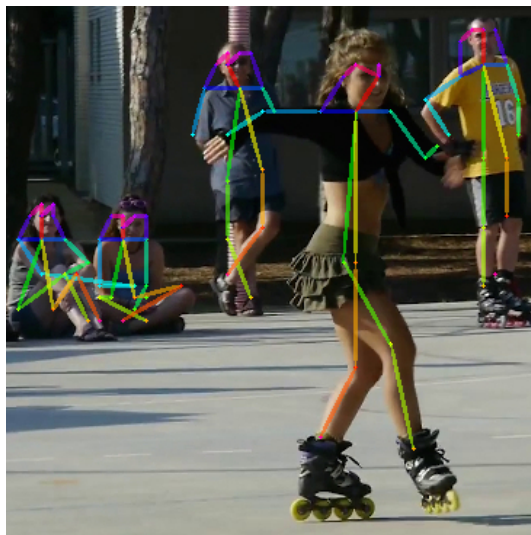
Output1 :

```
最終的 subset:
骨架 0: [ 0.          1.          2.          3.          4.          5.
  6.          7.          8.          9.         10.         11.
 12.         13.         14.         15.         17.         18.
41.1386306 38.          ]
```

Output2 :

```
最終的 subset:
骨架 0: [ 3.          8.          13.         18.         24.         28.
 33.         37.         44.         47.         52.         59.
 62.         67.         73.         78.         83.         87.
44.87914342 40.          ]
骨架 1: [ 0.          5.          10.         15.         21.         25.
 30.         35.         40.         45.         50.         55.
 60.        -1.         70.        -1.         80.         85.
38.3965891 32.          ]
骨架 2: [ 2.          7.          12.         16.         20.         27.
 32.         36.         42.         49.         54.         57.
 64.         69.         72.         77.         82.        -1.
37.66937214 34.          ]
骨架 3: [ 1.          6.          11.         17.         -1.         26.
 31.        -1.         41.         46.         51.         56.
 61.         65.         71.         76.         81.         86.
42.08062637 34.          ]
骨架 4: [ 4.          9.          14.         19.         24.         29.
 34.         39.         43.         48.         53.         58.
 63.         68.         74.         79.         84.         88.
45.56413941 40.          ]
```

(三) 兩張圖生成結果截圖



左圖為 output1.png，右圖為 output2.png

（四）過程中遇到的問題及解決方法

在這次作業中，最具挑戰的部分是要讀懂所有重要的程式碼檔，並深入理解整個人體姿態估計的運作過程。每個函式的作用、在流程中如何運用，以及其輸入輸出和陣列的意義，都需要逐一釐清。我會和同學們一起討論，交流彼此的理解，這樣在理解每個函式在各部分中的作用後，撰寫merge.py時便輕鬆不少，讓我能更順利地掌握程式邏輯。經過這樣的討論和思考，面對程式碼中的細節更游刃有餘，也更確信自己對人體姿態估計的理解。