**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。

1. 合併重疊骨架
2. num\_skeletons 表示骨架的數量
3. 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。

1. 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。
2. 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。
3. 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。

1. 最後，我們將標記為 -1 的骨架刪除，只保留合併後的骨架。
2. 刪除不符合要求的骨架
3. parts\_num: 計算骨架中的部位數量（例如有幾個部位是有效的）。
4. total\_score: 計算骨架的總分數。
5. 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。
6. 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 :

一張含有 文字, 螢幕擷取畫面, 黑色, 字型 的圖片

自動產生的描述

Output2 :

一張含有 文字, 螢幕擷取畫面 的圖片

自動產生的描述

（三）兩張圖生成結果截圖

 一張含有 足部穿著, 人員, 體育, 服裝 的圖片

自動產生的描述

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。