2025 嵌入式多核心系統與軟體 - Pthread專題

A1105534 張宏宇

# 結果

一張含有 行, 圖表, 繪圖, 文字 的圖片

AI 產生的內容可能不正確。

# 問題討論

## 請問每組資料各需要多少時間才能做完整張圖片的Template Matching?

這邊我平行運算皆拿最多線程數量的結果

|  |  |
| --- | --- |
| 資料 1 (3x3, 3750x4320)   * 無平行計算:   + PCC: 0.281529 秒   + SSD: 0.146937 秒 * 平行計算 (32 線程):   + PCC: 0.028171 秒   + SSD: 0.016721 秒   資料 2 (5x5, 7750x1320)   * 無平行計算:   + PCC: 0.488946 秒   + SSD: 0.226849 秒 * 平行計算 (32 線程):   + PCC: 0.035001 秒   + SSD: 0.024310 秒 | 資料 3 (3x3, 8140x9925)   * 無平行計算:   + PCC: 1.411686 秒   + SSD: 0.753369 秒 * 平行計算 (32 線程):   + PCC: 0.103491 秒   + SSD: 0.062905 秒   資料 4 (5x5, 50x50)   * 無平行計算:   + PCC: 0.000096 秒   + SSD: 0.000045 秒 * 平行計算 (32 線程):   + PCC: 0.001916 秒   + SSD: 0.002194 秒 |

## 你搭配出怎麼樣的thread count達到最好的效能?

下面是取了每組資料中PCC和SSD的最快計算時間與對應的thread count、加速倍率：

* 資料 1:
  + SSD: 最佳線程數 29，時間 0.014221 秒（加速 10.33 倍）
  + PCC: 最佳線程數 29，時間 0.021896 秒（加速 12.86 倍）
* 資料 2:
  + SSD: 最佳線程數 28，時間 0.019380 秒（加速 11.71 倍）
  + PCC: 最佳線程數 30，時間 0.034592 秒（加速 14.13 倍）
* 資料 3:
  + SSD: 最佳線程數 30，時間 0.061053 秒（加速 12.34 倍）
  + PCC: 最佳線程數 32，時間 0.103491 秒（加速 13.64 倍）
* 資料 4:
  + SSD: 最佳線程數 1，時間 0.000045 秒（加速 48.76 倍）
  + PCC: 最佳線程數 1，時間 0.000096 秒（加速 19.96 倍）

對於較大資料量的case（資料 1、2、3），最佳線程數通常在 28-32 之間，平行計算顯著提升效能，至於為啥不是32，可能是因為加速後的運算時間已經快小於平行化的而外開銷了，所以從關係圖看也可以發現，後面幾乎趨近於同一個時間。

但是資料較少的4，最佳效能反而出現在無平行計算的時候，線程越多反而越慢，主要問題還是在平行化的而外開銷所消耗的時間已經大於執行時間了。

## 除了thread count的參數之外，你還有做什麼特別的效能優化嗎?

* 預分配子矩陣的記憶體空間。

雖然在讀取資料上可以多線程讀取，但從Disk讀資料進Mem的IO性能不再此專題討論中就沒實作了。