

醫療、運動及職業安全應用的運動分析演算法開發

文/ Edgar Charry, dorsaVi 公司

優秀的運動員都希望在受傷後可以快速復原返回運動場。因此，即便尚未完全康復，他們常常試圖去說服他們的醫生以及自己認定他們其實已經可以運動了；而運動員和訓練員也利用一些先進的科技，諸如使用光學追蹤系統來擷取動作，或是透過測力板(force plates)測量地面反作用力，來協助運動員及訓練員確認其何時可以安全的回到運動場。雖然這些技術所提供的資訊及評估，遠超過單單透過觀察運動員去了解其狀況，但仍有一些缺點。除了昂貴之外，基本上也需具備專業訓練的技術人員協助，以及長時間的設置程序。再者，因為這些設備只能讓運動員在比較有限的環境下執行，比方說在跑步機上，因此難以評估平常運動情境的狀況。

因此，在 dorsaVi，我們的團隊開發了一個無線可穿戴的運動分析裝置，不論運動員在任何情況下自由的活動，我們都可以精確的測量並且追蹤其動作（圖 1）。



圖 1. 跑步測試使用的 ViMove 感測器

ViPerform 結合了慣性測量單元 (IMU)、磁力計，以及肌電感測器 (electromyography sensors)，用以測量肌肉活動。感測數據會被傳到一個記錄反饋裝置 (RFD)，使用者可將此裝置戴在手臂或放入口袋，之後 RFD 會將量測到的數據傳送至電腦進行數據處理，並透過 dorsaVi 軟體顯示出來。

dorsaVi 的技術核心是一些我們自行研發的演算法，可以過濾及分析原始的感測數據，進而對膝蓋控制、腰部運動、腿部筋絡運動、臀部和核心肌群控制以及跑步性能等運動提供資訊。我們在 MATLAB 環境開發並測試演算法，並使用 MATLAB Coder 轉為可攜式 C 程式碼，與過去直接撰寫 C#相比，開發時間可以縮短近一半的時間。

目標分析的演算法開發

目標分析是經由處理原始感測數據，並以精密的演算法讓處理結果具有意義。例如，某一種演算法可以識別當運動員跑步時，從加速器測得其每一個步伐產生的負尖峰。藉由這些尖峰的幅度和時機，該演算法可以分別計算跑者的節奏及每一步的地面反作用力。這種分析可以偵測兩腿不對稱的節奏或不平均的地面反作用力，進而顯示出運動員無法或較難平均地施力於兩腿的情況。同樣地，當運動員深蹲或跳躍時，也可分析膝蓋偏差，屈曲和旋轉，以引導曾受膝傷、前十字韌帶 ACL 斷裂或其他膝蓋傷害的運動員進行復健。即使是健康的運動員，也能夠受益於這樣的分析，比方說利用演算法的回饋，來改進他們的技巧或優化關鍵動作的效率。

直接透過 C# 來實現演算法時，我們必須自己去開發低階訊號處理以及圖像函數。若轉由 MATLAB，我們可以藉由它資料視覺化以及離散傅立葉轉換的內建函數來加快開發速度。現在，我們更進一步使用訊號處理工具箱 (Signal Processing Toolbox™) 以及小波工具箱 (Wavelet Toolbox™) 來精簡同步開發流程。例如，我們用訊號處理工具箱來進行設計，並運用 Butterworth 以及無限脈衝響應 (IIR) 濾波器去感測數據；此外，我們使用小波工具箱 (Wavelet Toolbox) 去計算訊號中的連續小波轉換係數 (CWT)，以辨別感測訊號的波峰和波谷。這些訊號型態的辨識對於定位步伐活動 (pinpointing gait events) 相當重要，就跑步分析而言，能精確的分析如腳跟著地及腳趾離地等訊號。

驗證並測試演算法

在 MATLAB 環境下開發演算法之後，再將其結果與經由設備 - 例如光學追蹤系統或測力板產生的結果相互比較進行驗證。比方說，我們會比對由演算法偵測到的步伐活動時機，以及由測力板產生的地面反作用力尖峰的時機，在此階段，檢查由 MATLAB 產出的圖像，以判別並修正可能潛藏的演算法缺陷 (圖 2)。

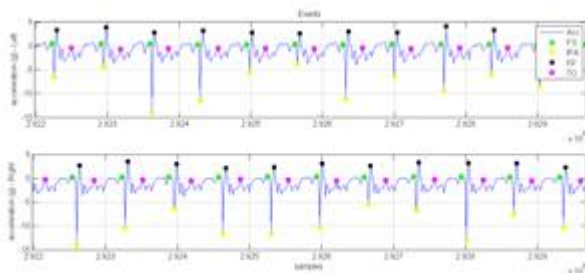


圖 2 MATLAB 圖形顯示跑步測試的結果

在限定的測試範圍內，當滿意演算法的效能之後，就會使用世界各地運動員的數據紀錄，繼續在 MATLAB 上進行超過 1000 組的測試。之後，則持續的調校並優化我們的演算法，直至其在效能及精確度上符合需求。

將 MATLAB 演算法轉為 C++程式碼

DorsaVi 軟體是利用 C#所撰寫的，可提供一個檢視感測數據及評估運動員運動表現的主要操作介面，最初使用 MATLAB 來開發訊號處理演算法時，我們的工作流程仍然倚賴 C++及 C#的程式撰寫者將演算法實現成生產程式碼(production code)並整合至 dorsaVi 上；不過這個方式非常沒有效率，包含許多重複的工作，還可能需要多花一個月去重新撰寫、測試已經在 MATLAB 開發測試完成的演算法。

為了減少無效率的工作方式並且縮短專案交付的時間，我們決定從已在 MATLAB 驗證過的演算法上，直接使用 MATLAB Coder 來產生 C++。先初始化所有變數和尋求可以進行優化的迴圈，以準備將演算法轉為 C 程式碼，然後驗證我們的演算法準備進行轉碼，再藉由產生一個 MEX 函式將編譯完成的程式碼包起來，再從原先的 MATLAB 演算法中去呼叫該 MEX 函式。而從演算法轉為 C++程式碼後，將演算法編譯成一個 DLL 檔，C#的程式開發者再將此 DLL 檔下載至 dorsaVi 軟體上。

之前，需要花一個月的時間將 MATLAB 演算法轉譯成生產程式碼，透過 MATLAB Coder，現在只需要一兩天。最後一個步驟，我們對 dorsaVi 的演算法進行全系統測試，到目前為止，在程式碼產生過程中，我們還沒有發現有任何缺失。我們在 MATLAB 上所施作的延伸測試，可以讓我們的 C#開發人員花更多時間去開發 dorsaVi 的新功能，而不是將演算法錄製一遍再重新測試它。

重複使用演算法並回應客戶需求

我們已經將最初用來開發在 ViMove 及 ViSafe 產品的 ViPerform 的幾種演算法，分別重新使用在運動員、牙醫應用，以及職業健康應用上。我們團隊現正從事開發新的演算法，以利擷取不同身體部位的感測資料進行分析，並且由現有客戶需求去強化演算法開發。在最近的版本中，我們更新了兩項演算法模組、新開發兩組演算法、並且自信滿滿地全數交付。