

- Introduction/Motivation

- 插值方法、正向運動學

- ◆ 四元數、asf 檔、acm 檔

- Fundamentals

- Describe local & global coordinate in your word

- ◆ local coordinate

- 可以視為空間中的空間，有自己獨立的座標系統（原點、軸向、單位）。

- ◆ global coordinate

- 所有物體都共用的座標系統。

- local coordinate 下的物體可以藉由原本的區域座標系統參數（原點、軸向、單位）轉換成 global coordinate。

- Implementation

- 計算四元數為弧度，angular_vector 儲存則是角度。

- Time warping

- 以 desired_catch_frame_idx 為界分為前和後，前者根據與 drop_ball_catch_frame_idx 的比例，後者根據 desired_catch_frame_idx 後剩下 frame 數與 drop_ball_catch_frame_idx 剩下 frame 數的比例，乘以 j 後得到上下兩個 frame index。接著對這兩個 frame 進行插值（slerp & linear interpolation）。第 j 個 frame 的 vector6d_t 即為插值結果。

- Forward Kinematics

- ◆ ConstructArticPath()

- 設定 root，對每個 bone 設定 parent，push 進 artic_path_內。

- ◆ ComputeSkeletonPose()

- 舊 pose 的 end_pos 是新 pose 的 start_pos。

根據上課講義和 `acclaim_FK_IKnote.pdf` 的式子將每個向量、矩陣都化為四元數做運算。

因為四元數乘法沒有交換律，需要注意乘法順序。

● Question & Solution

- 老師上課的 ppt 跟作業的 ppt 骨架圖 id 都只有到 29，但實際上是到 30，如果用 ppt 給的骨架圖做的話在 Time warping (punch_kick) 會很明顯看到球卡在手的骨架上（原本應該會在骨架末端接到球）。
- 一開始在計算 asf 的 quaternion 時沒有把原本參數得出的 rotation matrix 做 transpose，弄了很久才發現 `acclaim_FK_IKnote.pdf` 裡面的表格中有寫是用 transpose 格式儲存的…
- 每多跑一次資料夾的大小就會加差不多 100MB 左右，一開始用寫完程式的資料夾壓縮後超過 500MB 不能上傳，後來用原本的壓縮檔解壓縮後跑過一次的版本再壓縮才上傳完成。

● Result & Discussion

- 在做 forward kinematics 的時候偶然發現即使旋轉量多乘一個 root 的旋轉量，在 punch_kick 內雖然姿勢很怪也會接到球（但我沒有試過多乘其他的會不會一樣）。
- 改變 `desired_catch_frame_idx` 並觀察動作變化，會根據 frame 數決定動作的快慢程度。

● Conclusion

- 這次有附上滿多一些 library 的一些變數、函數解釋，對這些東西就不用再花太多時間弄懂。但有些在實作過程中會用到的變數、函數，甚至是原本 call 該 function 的函數都要另外找滿久的，不過畢竟是個大程式所以也沒辦法。
- 四元數的轉換及運算需要注意的地方滿多的，而且可以用的 function 也比矩陣少，但運算上比矩陣方便直觀很多。