

# 逢甲大學學生報告 ePaper

報告題名：

結合慣性測量單元(IMU)與 Deep Learning 演算法  
進行汽車姿態預估

Combining inertial measurement unit (IMU) and depth  
learning algorithm for vehicle attitude estimation

作者：黃俊穎、黃正毓

系級：電子工程學系乙班

學號：D0343302、D0343613

開課老師：陳冠宏

課程名稱：專題研究(一)

開課系所：電子工程學系

開課學年：105 學年度 第 2 學期



## 中文摘要

在交通發達的時代中，高交通事故已漸漸成為社會上重視的問題，駕駛人行駛車最大的優點是可以依據開車的經驗累積，可以隨時應變道路上各種的突發狀況，然而交通事故的發生，卻有九成以上是人為造成。

交通事故發生的主要原因不外乎是因駕駛人容易受到外在因素干擾、疲勞駕駛或是使用智慧型手機而導致車禍的發生，例如：土地面積廣大的國家，往返兩地往往就要數十小時甚至跨日，對駕駛人而言是重大的負擔，因此提出了自動駕駛，主要考量為機器無精神上的考量。

在此報告中，我們提出基於加速度和角速度的車輛姿態識別方法，並且使用深度學習演算法卷積神經網路 CNN (Convolution Neural Network)來建構神經網路，和使用慣性測量單元 IMU (Inertial Measurement Unit)來收集汽車的六軸訊號。特別的是我們建構了特別的 CNN 模型來學習六軸 IMU 的訊號，使其效能精準度兼具。此外，我們總共收集了 800 個訓練資料，並經由特殊的預處理方法，得到 59200 筆的訓練資料。最終實驗結果證明，此 CNN 網路架構是相當良好的，可以在完整動作的 20%以內完成預估，精準度達到 98%(測試來自於 800 訓練樣本，當中有 10%是尚未訓練的)，當中並沒有經過任何的特徵擷取，並且此深度學架構可以在樹梅派 3 上運行，而每次的預估時間均小於 0.5 秒。

**關鍵字：**卷積神經網路、深度學習、慣性測量單元、機器學習、車輛姿態預估

## 英文摘要

In the paper, we propose an acceleration-based and angular-velocity-based vehicle attitude recognition method by using a popular deep learning algorithm, i.e., Convolution Neural Network (CNN). We use an Inertial Measurement Unit (IMU) to collect six-axial signal of a vehicle. In particular, we construct a CNN model to learn the characteristics of six-axial IMU signal and the model can be used to predict vehicle attitudes. We constructed training data consists of 800 package from six attitudes. In addition, we preprocess the 800 package that each package will be broken down. Finally, our training data is 59200 sample-train. The experiment results show that the CNN works well, which can reach an average accuracy of 98% by the time of 1/5 of the overall action without any feature extraction methods. Due to our optimization of the convolution kernel number, this work can reach real-time processing capability. The estimated time of each action is less than 0.5 sec based on the raspberry pi3 development board.

**Keyword : CNN 、 Deep Learning 、 IMU 、 Machine Learning 、 Vehicle Attitude**

## 目 次

|   |    |
|---|----|
| 一、作品功能與構想 .....                             | 5  |
| 1.1 研究動機與設計構想 .....                         | 5  |
| 1.2 分析比較現行既有技術能力 .....                      | 6  |
| 1.3 設計流程 .....                              | 7  |
| 二、系統統體平台簡述與資源運用分析 .....                     | 8  |
| 2.1 軟硬體平台簡介 .....                           | 8  |
| 2.2 資源運用 .....                              | 8  |
| 三、系統實作內容 .....                              | 8  |
| 3.1 所提出系統示意圖 .....                          | 8  |
| 3.2 SparkFun 9DoF Razor IMU M0 訊號波型分類 ..... | 9  |
| 3.3 訓練資料處理 .....                            | 13 |
| 3.4 資料預處理 .....                             | 14 |
| 3.5 深度學習網路架構 .....                          | 16 |
| 3.6 訓練結果 .....                              | 16 |
| 四、實驗與分析 .....                               | 17 |
| 五、設計創意 .....                                | 17 |
| 六、遭遇之困難點及解決方法 .....                         | 17 |
| 七、工作分工 .....                                | 18 |
| 八、結論 .....                                  | 18 |
| 九、實際測試影片擷取 .....                            | 19 |
| 十、參考文獻 .....                                | 22 |

## 圖目錄

|   |    |
|---|----|
| 圖. 1 系統設計流程圖 .....                      | 7  |
| 圖. 2 系統示意圖 .....                        | 8  |
| 圖. 3 各軸所代表的物理意義 $w$ :角速度 $a$ :加速度 ..... | 9  |
| 圖. 4 原始資料轉換成波形圖 .....                   | 10 |
| 圖. 5 左轉資料擷取 .....                       | 10 |
| 圖. 6 右轉資料擷取 .....                       | 11 |
| 圖. 7 左切換道路資料擷取 .....                    | 11 |
| 圖. 8 右切換道路資料擷取 .....                    | 12 |
| 圖. 9 停紅綠燈資料擷取 .....                     | 12 |
| 圖. 10 直線行駛資料擷取 .....                    | 13 |
| 圖. 11 IMU 擺放位置 .....                    | 13 |
| 圖. 12 靠左行駛及靠右行駛數據圖 .....                | 14 |
| 圖. 13 數據分解說明圖 .....                     | 15 |
| 圖. 14 CNN model .....                   | 16 |
| 圖. 15 沒有校正陀螺儀的車輛姿態識別系統 .....            | 17 |
| 圖. 16 靠左行駛及靠右行駛數據圖 .....                | 18 |
| 圖. 17 實地測試影片正常行駛 .....                  | 19 |
| 圖. 18 實地測試影片左轉 .....                    | 20 |
| 圖. 19 raspberry pi 與 IMU 最終成品 .....     | 21 |

## 表目錄

|                                      |    |
|--------------------------------------|----|
| 表. 1 IMU、GPS、影像辨識功能評比.....           | 6  |
| 表. 2 訓練姿態種類.....                     | 9  |
| 表. 3 訓練樣本分解.....                     | 15 |
| 表. 4 CNN model of test results ..... | 16 |
| 表. 5 動作完成度與精準度.....                  | 17 |
| 表. 6 遭遇之困難點及解決方法.....                | 18 |



## 一、 作品功能與構想

### 1.1 研究動機與設計構想

在自動駕駛方面，汽車的姿態是相當的重要；在控制方面，必須知道自己正處在甚麼行為，就像是明明輪胎右轉，但因打滑而尚未右轉；汽車姿態也可以提升主動安全設備的精準度和提醒駕駛汽車狀態，像汽車正朝右前方偏離，但 GPS 顯示此為直線公路，意旨汽車正朝護欄方向前進，這時如果可以知道汽車的移動姿態就可以提早警惕駕駛，例如：近期國道五號遊覽車翻覆事件，如果這時有主動安全系統可以偵測汽車姿態並適時的介入煞車或控制，將可以把人為駕駛的操作失誤降到最低，並使汽車的安全性提高。

本計畫想藉由汽車 IMU 訊號配合 Deep Learning 來預估汽車現在的動作，加以輔助自動駕駛以及行車監控，或是輔助難以用攝像鏡頭得知的姿態。IMU 優點為其價格較低、功耗低、體積小和儲存的資料也較小，非常適合用來當車用電子裝置，再加上 Deep Learning 演算法來訓練並且判斷其 IMU 訊號，用來提前預估汽車將要做的動作，即時甚至提前告知電腦或者駕駛。

這次的專題計畫有實地採集一些數據，並經由文獻中發現，資料採集的頻率不同對於得到收集良好的訊號是有影響的，本計畫也會經由實地實驗，找出較良好的採集頻率，分別使用不同頻率所得到的特徵做訓練，研究哪種頻率的訓練對成果判斷的精準度較佳。計畫最終將找出最佳的演算法與採集頻率，使其能在汽車未完成完整姿態的 25% 內，預估汽車當下的行為，而不是等到汽車行駛結束才做預估。應用方面，輔助自動駕駛做判斷及處理、使用 Deep Learning、協助車用主動安全系統做判斷，甚至還能當作汽車的行車紀錄與輔助 GPS 使其定位更加精準。

## 1.2 分析比較現行既有技術能力：

以目前車輛動態預估，使用的方法有建立二自由度自行車模型及三自由度側傾模型，再以INS及GPS來估測偏轉角速度、側傾角、車身側滑角及側向加速度，此做法必須在車上加裝GPS或INS成本也相對較高。根據影像及路面3D模型來估測偏轉角、側傾角及俯仰角等車輛動態，此作法雖將車輛有效預估，但也必須加裝影像系統。

影像系統預估車輛動態會因雨天或與陰霾起霧等影響影像辨識，而IMU並不受天氣狀況所影響，即使在雪地、雨天、高山行駛等皆可運作不受干擾。影像系統比起IMU訊號的資料量更大許多，對於長路程來說會造成資料量負載問題，對於影像系統搭配影像辨識的預估裝置也是非常昂貴且還未成熟，目前影像辨識搭配光流來做辨識，但是大部分已車牌辨識、距離監測、號誌路標辨識為主，較無法針對車輛姿態做預估。因此本專題不只方法新穎使用了Deep Learning 結合IMU做車輛預估外，其價格低廉、功耗低以及體積小的特性非常適合安裝於車用電子的系統平台，與影像辨識的最大差別是不用看到外頭的世界，意旨說可以安裝在汽車內部。

表1. IMU、GPS、影像辨識功能評比

|              | IMU6050     | Garmin<br>DriveAssist50<br>車用衛星導航 | 鏡頭             |
|--------------|-------------|-----------------------------------|----------------|
| Sensor price | NT\$200     | NT\$2000~12000                    | NT\$1000~10000 |
| Sensor size  | 3*3cm       | 14*8.2*2.1cm                      | 2*2CM          |
| Data         | 10~20kb/sec | 200~300kb/sec                     | 400~500kb/sec  |



### 1.3 設計流程

本專題使用了IMU感測器，安裝在車上並且實地採集車子姿態的各種訊號(例：左右轉、切換道路、上下交流道、直線加速、停紅綠燈等)，接著使用Matlab擷取其特徵訊號，並丟入類神經網絡的輸入層，加以訓練；訓練完畢以後，再丟輸入各種訊號讓Deep Learning做判斷，並且能在訊號的前25%做出車輛動態預估，精準度高達95%以上，其流程圖如下

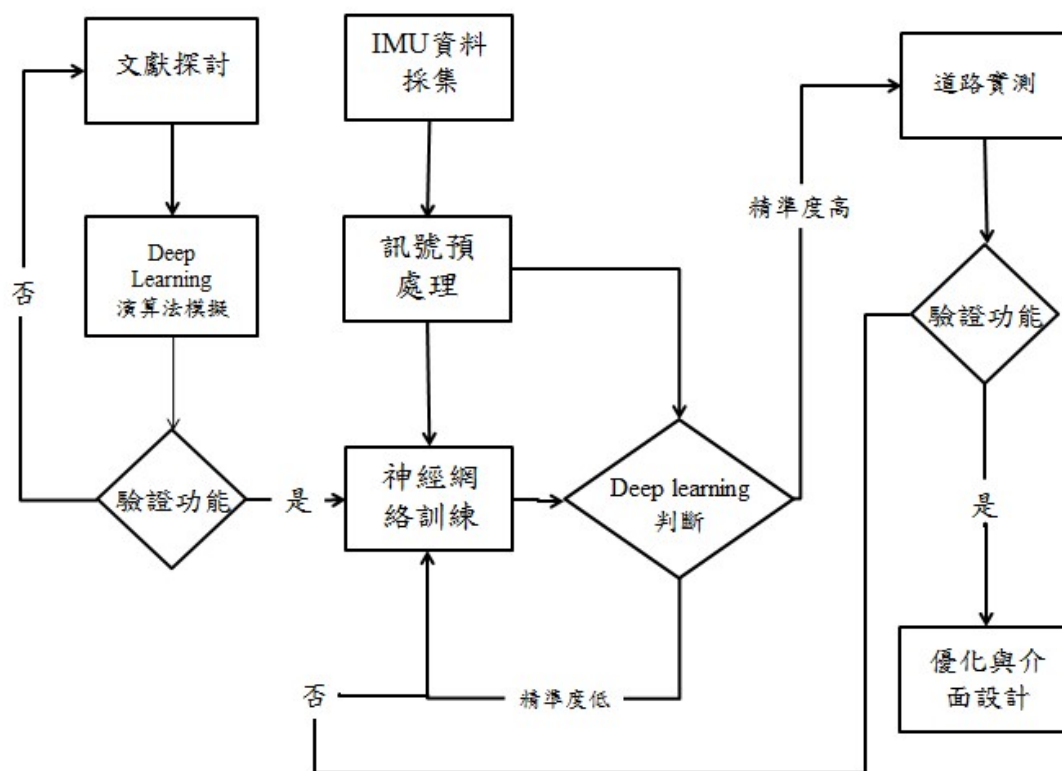


圖 1. 系統設計流程圖

## 二、 系統軟硬體平台簡述與資源運用

### 2.1 軟硬體平台簡介

軟體: Matlab、Theano 0.9、Tensorflow1.0.1、Keras 2.0.0、Python 2.7、anaconda2

硬體: SparkFun 9DoF Razor IMU M0

### 2.2 資源應用

本專題使用 python 當作開發語言，以 theano、keras library 創建深度學習網路，並搭配 arduino 來當作 IMU (Inertial measurement unit)的擷取平台。PC 端會同時顯示出此時訊號的波形以及預估姿態。

## 三、 系統實作內容

### 3.1 所提出之系統示意圖

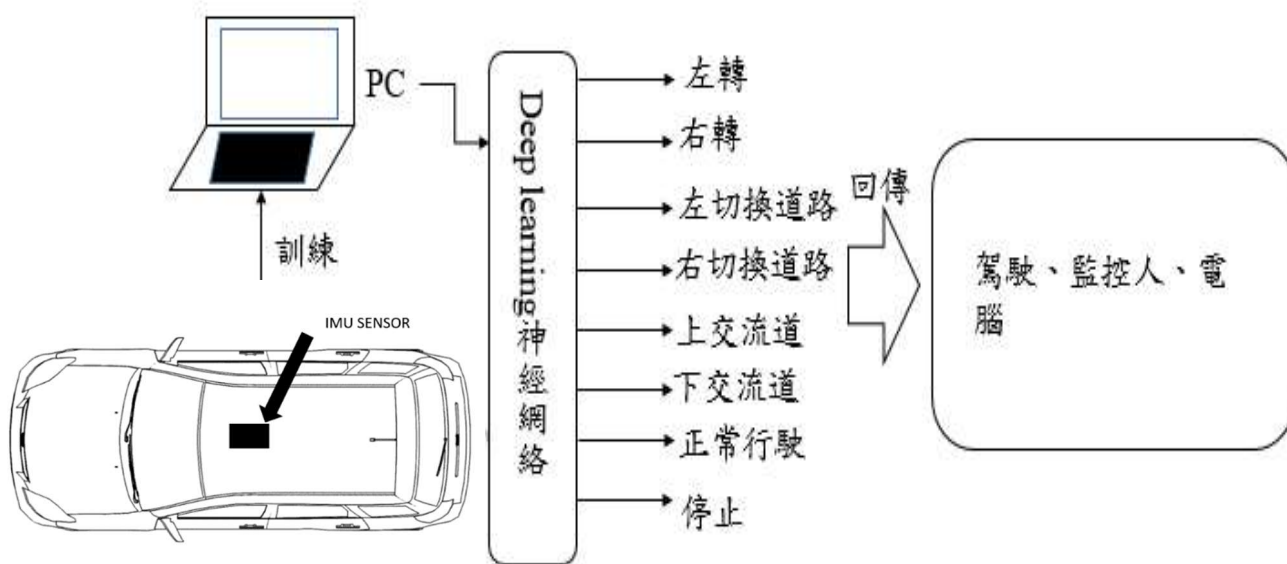


圖 2. 系統示意圖

### 3.2 SparkFun 9DoF Razor IMU M0訊號波型分類

使用SparkFun 9DoF Razor IMU M0感測器接收車輛六軸數據，製成波形圖後，將其各種動作加以擷取200筆並且分類，最後放入CNN類神經網絡架構裡做訓練，藉此訓練神經網絡分別出車輛各種姿態的波型，加以辨認。

表2. 訓練姿態種類

| 目前6種訓練姿態 |       |      |
|----------|-------|------|
| 左轉       | 左切換道路 | 停車   |
| 右轉       | 右切換道路 | 直線行駛 |

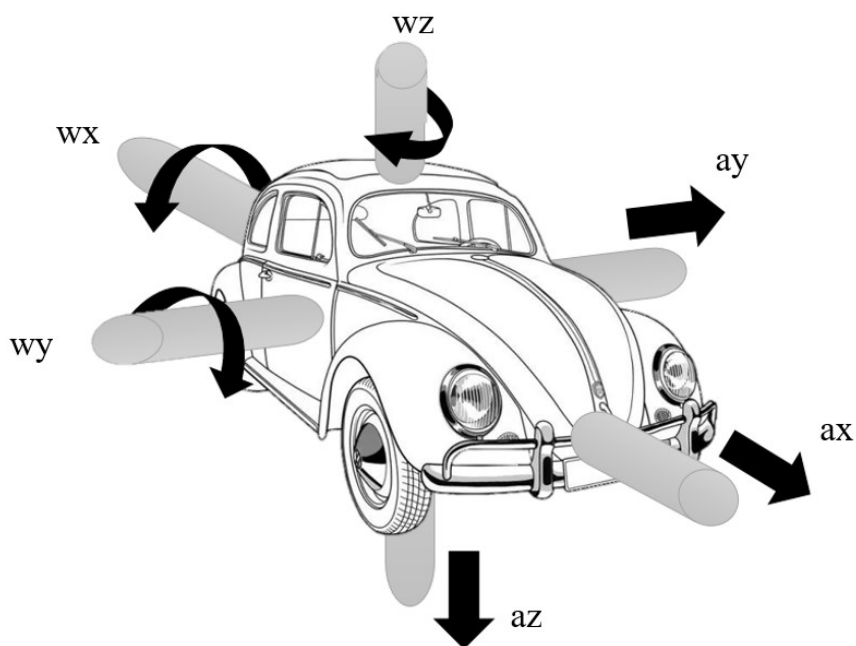


圖 3. 各軸所代表的物理意義w:角速度 a:加速度

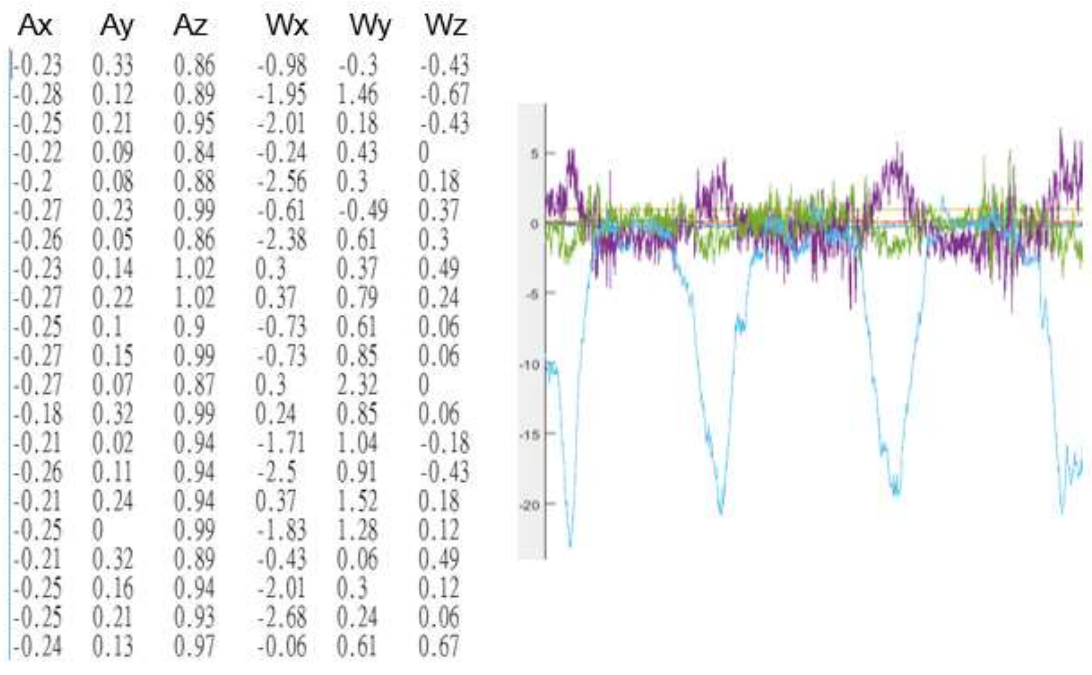


圖4. 原始資料轉換成波形圖

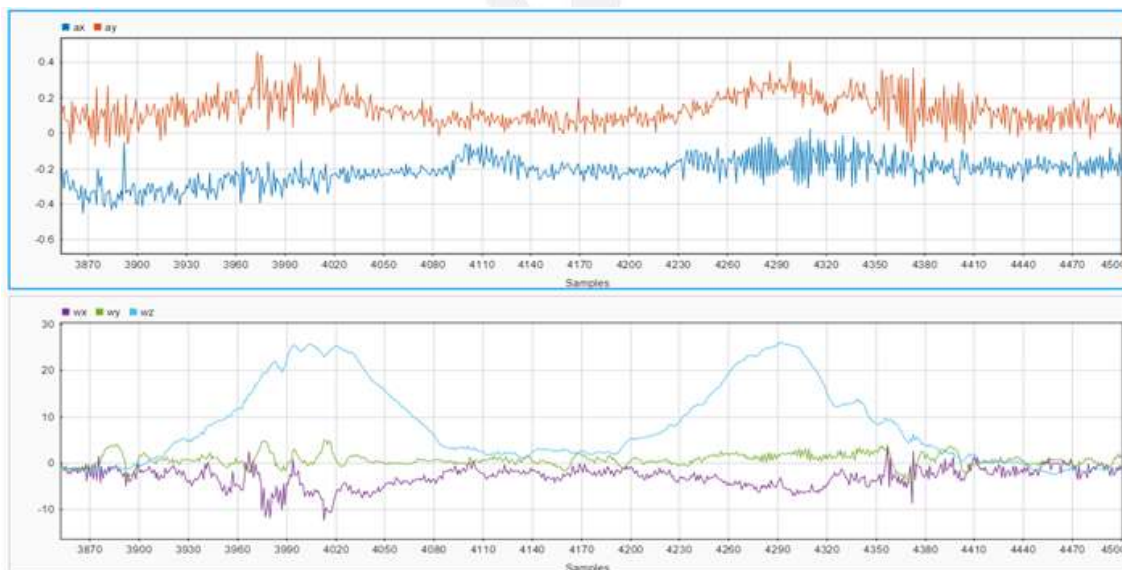


圖5. 左轉資料擷取

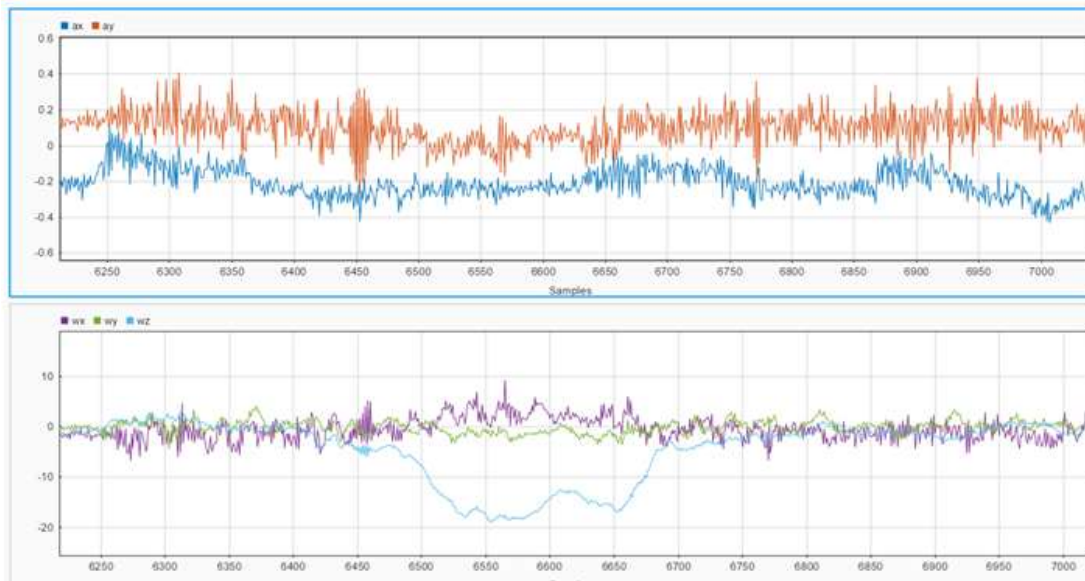


圖6. 右轉資料擷取

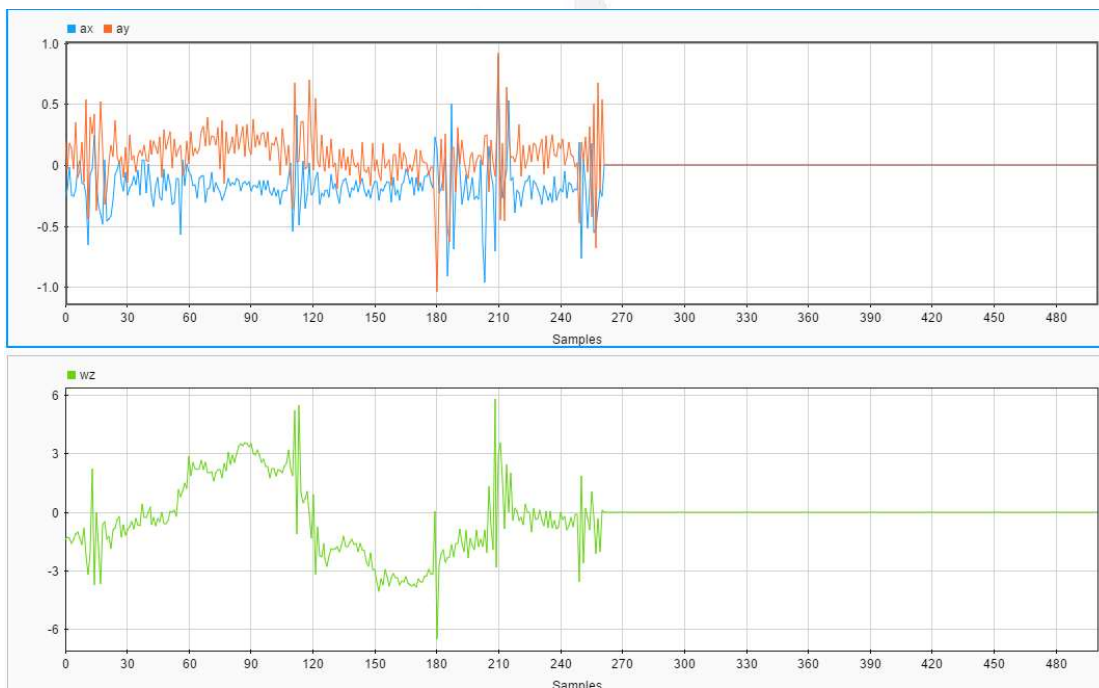


圖7. 左切換道路資料擷取

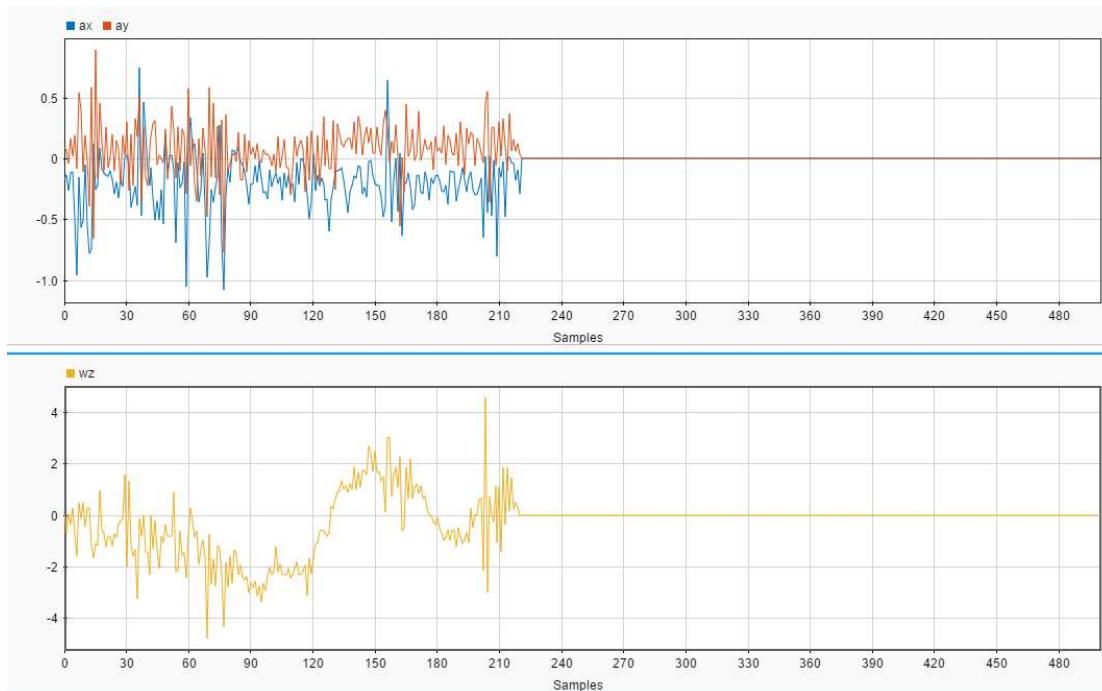


圖8. 右切換道路資料擷取

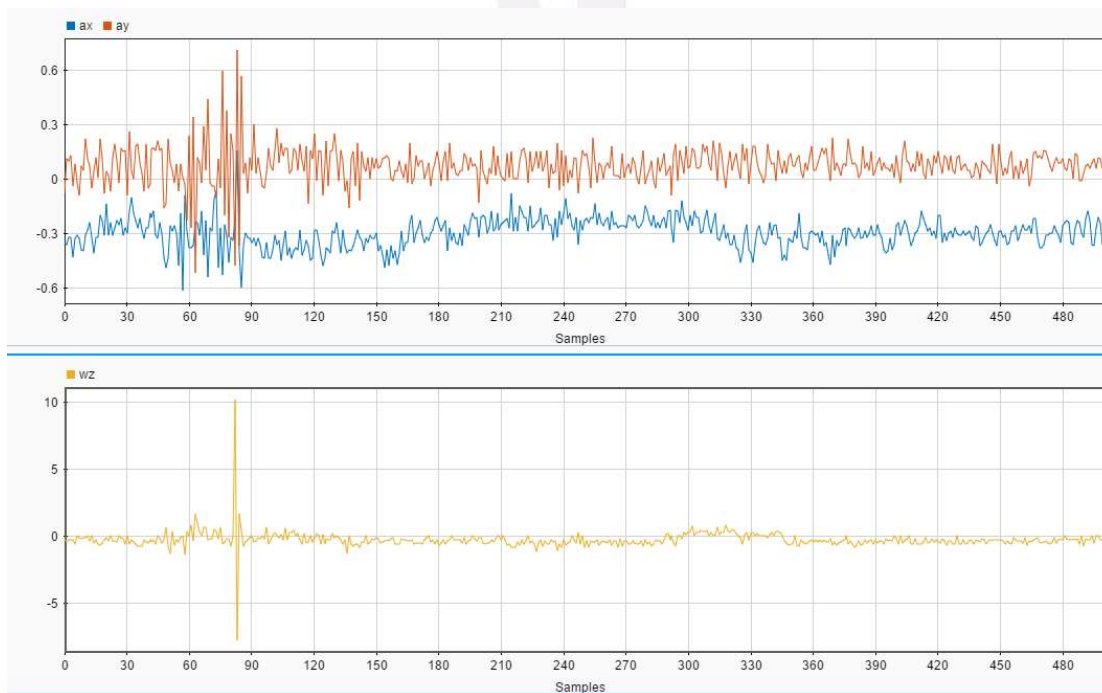


圖9. 停紅綠燈資料擷取

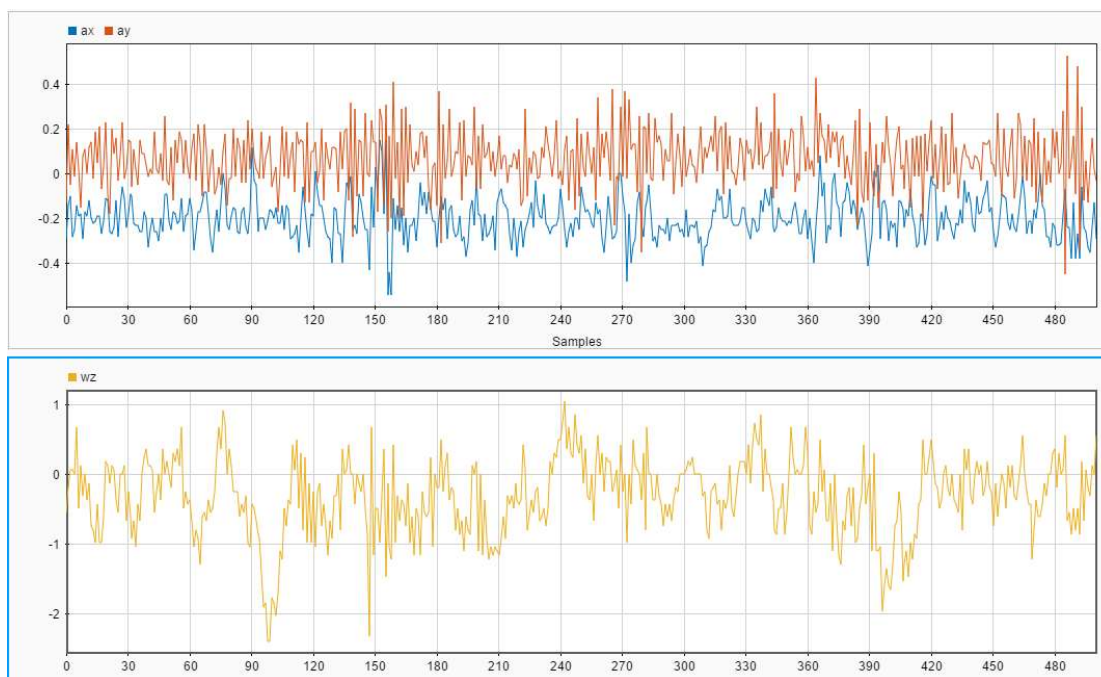


圖10. 直線行駛資料擷取

### 3.3 訓練資料收集

要得到一個成功的深度學習訓練網路，他必須要有良好的訓練資料，在資料收集的這部分，我們將IMU放置在汽車中央(如圖10)，並且使用相機同步地紀錄汽車當時的動作狀態，以方便事後做資料預處理與驗證時使用，IMU感測器會以30Hz蒐集6軸的訊號，分別是3軸的加速度與3軸的角速度(如圖3)，因為這六軸都有它所代表的物理意義，所以對於汽車姿態的辨識是有相當大的幫助。我們收集了汽車的6種姿態(如表3)。

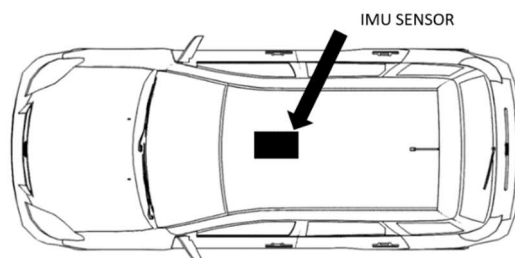


圖 11. 這張圖說明著 IMU 擺放的位置，在收集這部分我們發現汽車的震動頻率的干擾相當的嚴重，所以最終我們選擇了振動頻率較低的地方，正式汽車的正中央。

### 3.4 資料預處理

經過多次測試我得到汽車的動作(ex: 右轉...)時間大約落在於 6.6 秒至 10 秒之間，並且我們發現右切換道路與左切換道路的訊號有些地方相當的雷同(如圖 12)，因我們是使用一個固定的掃描視窗(window)來掃描波型，但左右切換有波型雷同缺陷使得我們失敗多次，所以選適當的 window 是很重要的，這將會影響到訓練結果。因為左切換與右切換是汽車動作中最快的，也代表他的數據長度最短，因此我們設計了一個公式，將數據對最短的大小帶入公式(1)，便可得到最佳的掃描 window。

$$y = x + \Delta y, \quad \Delta y \doteq 10\%y \quad (1)$$

(y 是最佳的掃描視窗大小，x 是最短數據的一半)



圖12. 靠左行駛及靠右行駛數據圖

為了使 window 掃描每個地方都能夠辨識出汽車當下動作，訓練資料必須經過預處理，首先我們先假設正常的駕駛行為，在開車時不管做做哪一種動作的前後都會是直線，這代表汽車動作開始會先以直線動作，結束會以直線結束，因此我們在做資料預處理時，會將每筆動作資料的前後都加上直線的數據(如圖 13)，並在將以 window 120\*6 間隔為 5 的為每筆資料進行分解為訓練資料方法如(圖 13)，

最終將會得到 59200 筆的訓練資料(表 3 .訓練樣本分解)。



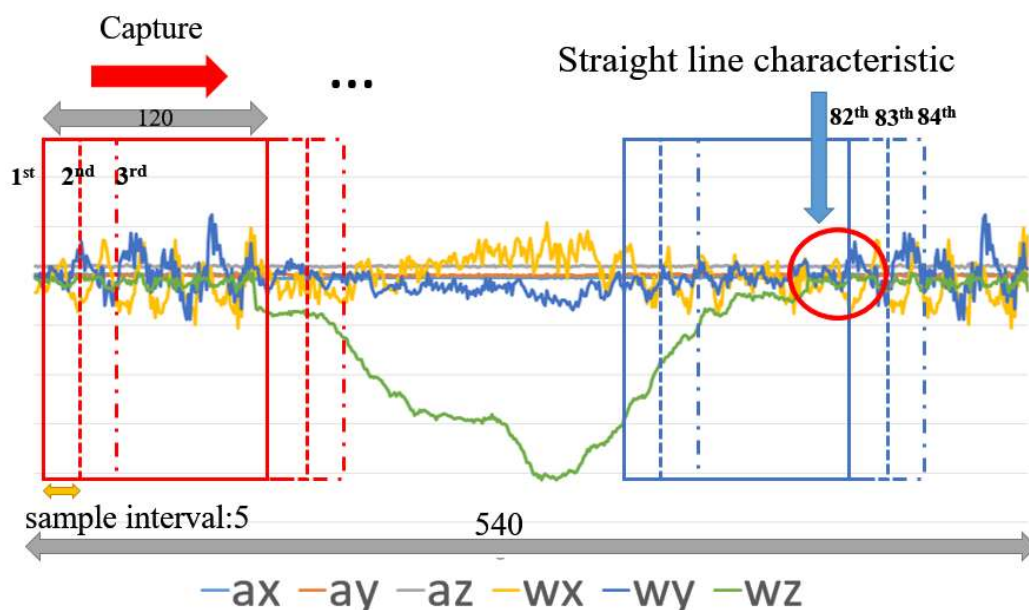


圖 13. 圖中的波行事一個右轉的波形，當我們在擷取練資料時，紅色框框代表右轉，藍色框框代表直線，所以當我們在分解一個在分解一個右轉波形時，會分解出右轉訓練資料與直線訓練資料，同樣地我們也將左轉、左切換道路、右切換道路做一樣的處理，而直線與停則直接做分解即可，並不需要在訊號的頭跟尾接上直線訊號。

表 3. 訓練樣本分解

|               | Stop | Shift left | Shift right | Turn left | Turn right | Go Straight |
|---------------|------|------------|-------------|-----------|------------|-------------|
| Signal length | 440  | 440        | 440         | 540       | 540        | 440         |
| Package       | 100  | 100        | 100         | 200       | 200        | 100         |
| Sample        | 6400 | 6400       | 6400        | 16800     | 16800      | 6400        |

### 3.5 深度學習網路架構

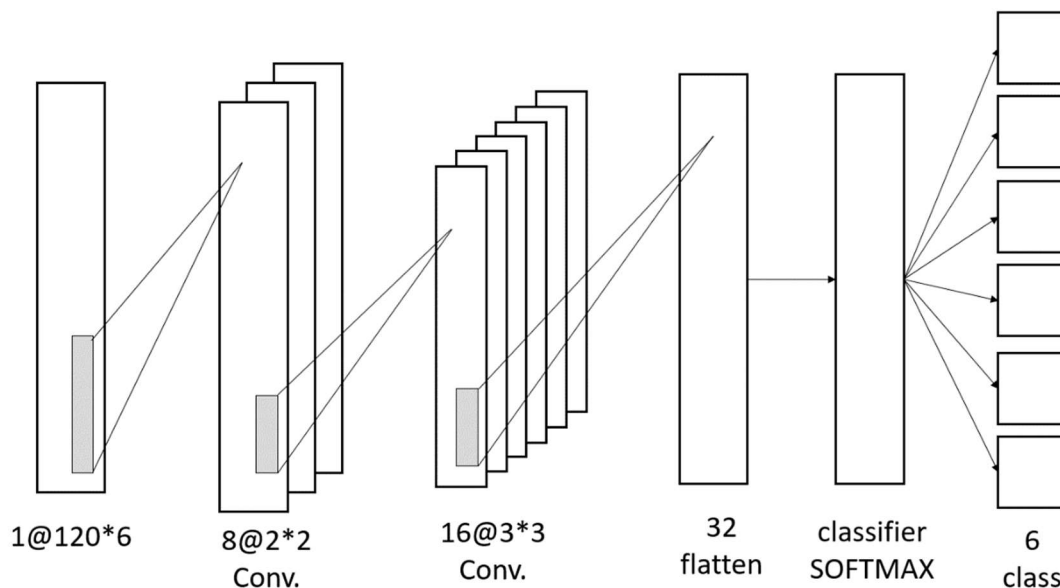


圖 14. CNN model

### 3.6 訓練結果

在深度學習當中，卷積核的選擇是相當重要的，這不只影響到系統的預估速度，也影響到整體的精準度，而在這方便我們經過多次的實驗，選擇了一個精準度較高CNN model。

表 4. CNN model of test results.

The accuracy and loss based on 59200 samples.

| 1 <sup>st</sup> CONV | 2 <sup>nd</sup> CONV | Test_accuracy    | Test_loss        |
|----------------------|----------------------|------------------|------------------|
| (2,2)                | (2,2)                | 0.9988243        | 0.0047272        |
| <b>(2,2)</b>         | <b>(3,3)</b>         | <b>0.9984375</b> | <b>0.0037726</b> |
| (2,2)                | (4,4)                | 0.9983597        | 0.0048684        |
| (2,2)                | (5,5)                | 0.9980816        | 0.0041436        |
| (3,3)                | (2,2)                | 0.9985156        | 0.0004889        |
| (3,3)                | (3,3)                | 0.9977526        | 0.0058891        |
| (3,3)                | (4,4)                | 0.9984968        | 0.0056904        |
| (4,4)                | (2,2)                | 0.9837343        | 0.0045667        |
| (4,4)                | (3,3)                | 0.9981055        | 0.0053838        |
| (5,5)                | (2,2)                | 0.9985924        | 0.0046172        |

#### 四、 實驗與分析

表 5.動作完成度與精準度

50%: half action, 100%: complete action, the accuracy is based on 800 package.(When the vehicle turns right, turn to half on behalf of 50%)

| 完整度  | Stop | Shift left | Shift right | Turn left | Turn right | Strain |
|------|------|------------|-------------|-----------|------------|--------|
| 7.5% | 100% | 86%        | 88%         | 0%        | 1%         | 100%   |
| 10%  | 100% | 62%        | 100%        | 0%        | 13%        | 100%   |
| 15%  | 100% | 98%        | 100%        | 57%       | 80%        | 100%   |
| 20%  | 100% | 98%        | 100%        | 91%       | 100%       | 100%   |
| 50%  | 100% | 98%        | 100%        | 100%      | 100%       | 100%   |
| 80%  | 100% | 99%        | 98%         | 100%      | 100%       | 100%   |
| 100% | 100% | 99%        | 98%         | 100%      | 100%       | 100%   |

#### 五、 設計創意

1. 使用 IMU 原始訊號不須經過校正陀螺儀
2. 使用 Deep Learning 由充滿雜訊的 IMU 訊號精準地偵測出車輛動態
3. 方法具強健性，不受天氣狀況以及視線的影響

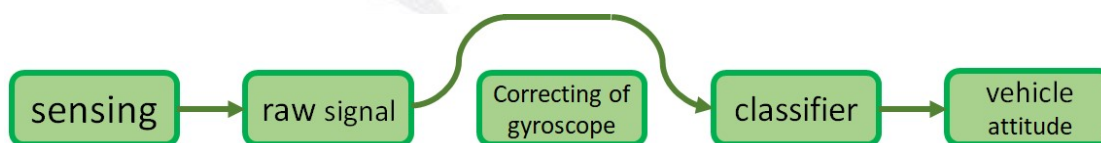


圖 15. 沒有校正陀螺儀的車輛姿態識別系統

#### 六、 遭遇之困難點及解決方法

為了使我們的深度學習訓練得更加的完整以及增加到最好的精準度，我們尋找出最適合的資料筆數以及訓練樣本選取。一開始只為了訓練車輛動作而擷取的波形未經過修飾，導致深度學習在動作25%時的精準度過低，最後調整波型擷取的範圍，終於找出最適合的訓練大小以及能夠在前25%資料即能判斷出的波形。

訓練資料的框架選擇問題，多次測試發現汽車的一個動作大約落於6.66秒到10秒之間，並且左切與右切有部份波形相同，如果掃描window太小容易誤判，最終將選取稍大於靠左行駛一半data的window當作掃描window。



圖16. 靠左行駛及靠右行駛數據圖

表6. 遭遇之困難點及解決方法

|        |          |
|--------|----------|
| 改善程式判斷 | 資料擷取位置改善 |
| 波形相似   |          |

## 七、 分工合作

**黃俊穎:**深度學習演算法模擬與驗證、python 編程、收集資料、文獻探討

**黃正毓:**資料預處理、收集資料、文獻探討

## 八、 結論

本文介紹了基於加速度和角速度以及特別的訓練資料擷取方式來訓練料，並以較簡化的 CNN 架構完成，最終並可以在汽車尚未完成動作時，以達到在整體動作 20%時，可以預知汽車正在進行的動作，預測汽車姿態不僅可用於自動駕駛，

結合慣性測量單元(IMU)與 Deep Learning 演算法進行汽車姿態預估

還可以用掌握駕駛人的行為，並糾正駕駛行為以避免車禍發生。

在一般的做法當中，大部分都通過陀螺儀得到汽車姿態，但陀螺儀必須經由事先的校正才能使用，而我們提出的方法是使用原始信號預估汽車正在做的動作。並且可以在 raspberry pi 3 上達成 realtime 預估(每次預估時間小於 0.35 sec)。

## 九、 實際測試影擷取

我們採取實地把 IMU 架在車上，並實地做測試

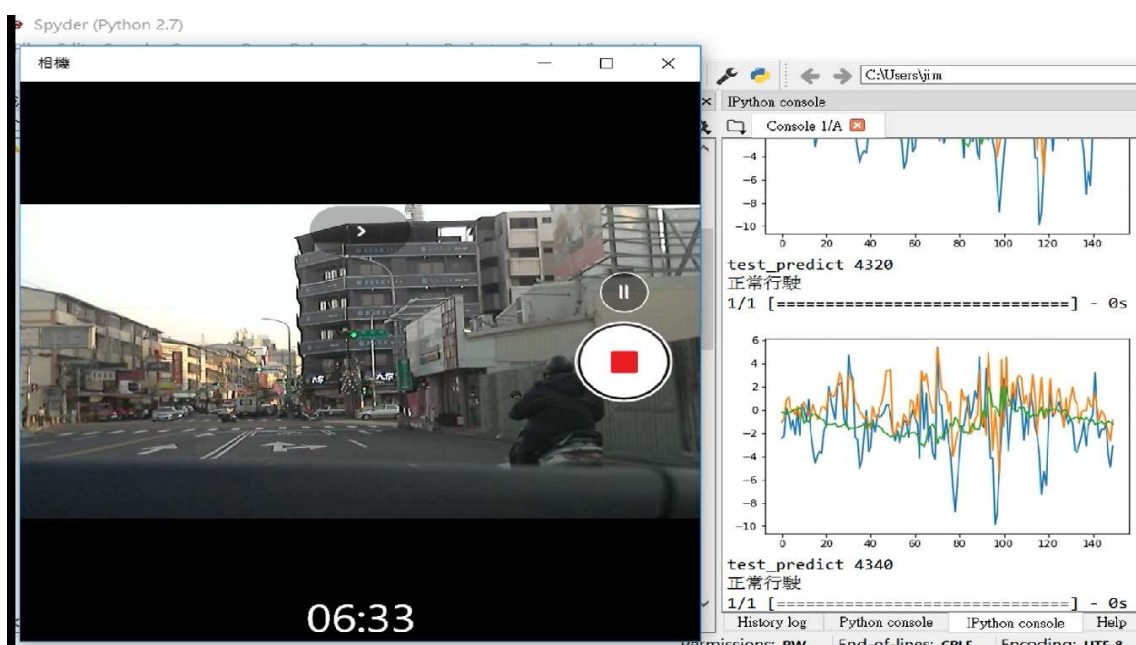


圖17. 實地測試影片正常行駛

結合慣性測量單元(IMU)與 Deep Learning 演算法進行汽車姿態預估

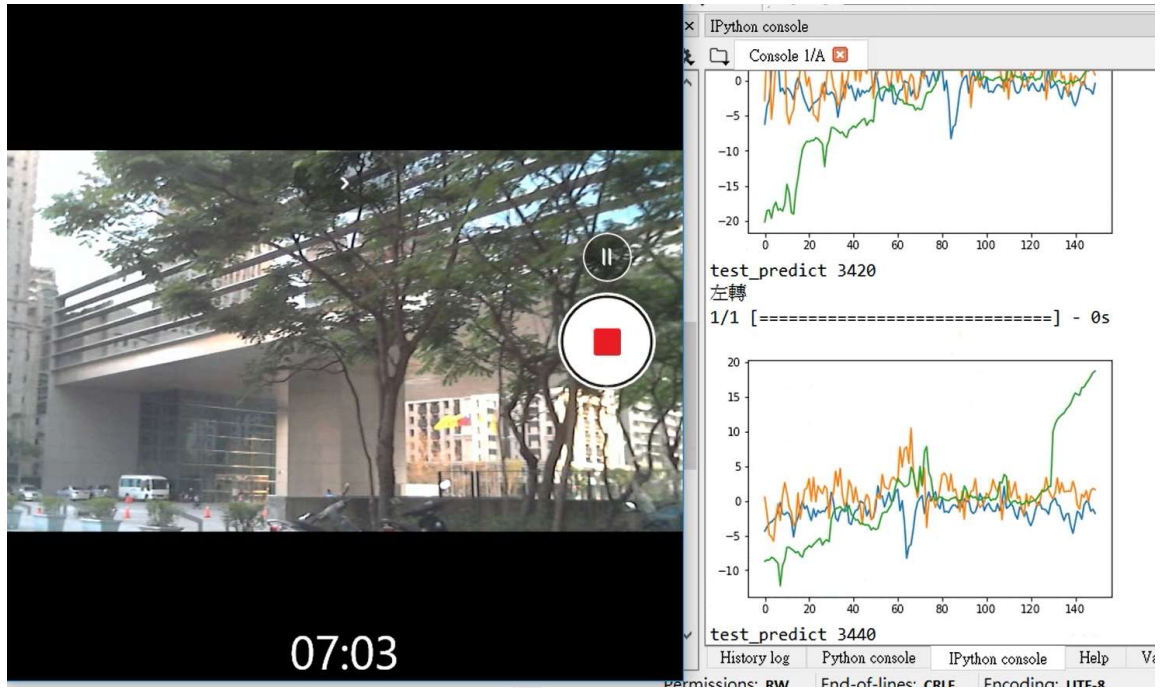


圖18. 實地測試影片左轉





圖19. Raspberry Pi與IMU最終成品

DEMO影面網址：

<https://www.youtube.com/watch?v=wesrS3rAIQQ&feature=youtu.be>

## 十、 參考文獻

- [1] S. Chen, X. Huang, W. Zhang. "Intelligent vehicle collision avoidance system based on DSP research" Automation technology and application, 7 (I) , 2005
- [2] H. Xiang et al, "Evaluation and Prediction of Driver's Bad Driving Behavior Based on the Moving Vehicle Attitude" International Journal of Security and Its Applications, Vol.9, No.11, pp.63-76, 2015
- [3] P. S. Bhagavatula et al. "Optic flow cues guide flight in birds." Current Biology 21.21 (2011): 1794-1799.
- [4] Y. Wu et al, "A monocular vision-based attitude estimation approach for small Unmanned Aerial Vehicles and its experimental verification", Guidance, Navigation and Control Conference (CGNCC), 2016.
- [5] M. H. Tehrani, M. A. Garratt, S. Anavatti. "Gyroscope offset estimation using panoramic vision-based attitude estimation and Extended Kalman Filter", Communications, Computing and Control Applications (CCCA), 2012 2nd International Conference on, 2012.
- [6] H. Ahmed, M. Tahir. "Accurate Attitude Estimation of a Moving Land Vehicle Using Low-Cost MEMS IMU Sensors", IEEE Transactions on Intelligent Transportation Systems, vol.99, 2016.
- [7] Y.Chen, Y.Xue, "A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer", IEEE International Conference on Systems, Man, and Cybernetics, pp. 1489-1491, 2015.
- [8] M. Zeng et al. "Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors Ming", International Conference on mobile computing, Applications and Services (MobiCASE), pp. 197-203, 2014.
- [9] C. Crema et al, "IMU-based Solution for Automatic Detection and Classification of Exercises in the Fitness Scenario", IEEE International Conference on Sensors Applications Symposium (SAS), 2017.