

## 第 19 屆行動計算研討會

### 使用機器學習辨識駕駛情境

## Recognizing Driving Scenario with Machine Learning Techniques

李佳福, 汪晨騏, 蔡欣穆

國立台灣大學 資訊工程學系

r02922135@csie.ntu.edu.tw

r01922140@csie.ntu.edu.tw

hsinmu@csie.ntu.edu.tw

李朝陽

財團法人車輛研究測試中心

綠能車輛發展處

nelsorn@artc.org.tw

### 摘要

駕駛情境辨識的技術對很多車輛應用系統,如車輛省油系統、駕駛輔助系統、及未來的自動駕駛等,皆為一重要之子系統。它可以利用分析收集的車輛資訊,幫助這些應用系統做出更好的決策。本研究致力於使用機器學習的方式建立一個行車情境辨識模型,我們定義的行車情境指的是一個較長時間範圍的情境,而不是細微的、如加速、減速等行為。例如從交流道上高速公路前的一段路程必須要加速到某個速度,以匯入車流中,這樣的一個情境我們會定義為加速。我們所建立的辨識模型,可以根據由手機上的感測器及目前車輛均配備的車載自動診斷系統(OBD-II)所收集的資料,辨識共六種情境。我們收集來自不同車輛的資料來建立及測試模型,並利用行車紀錄器記錄影像作為判斷行車情境的基準。為了確保即時的車輛情境辨識在行動裝置上是可行的,我們對 29 種收集的特徵進行特徵值選取以降低資料維度。驗證結果顯示當我們減少特徵值的數量之後,仍可維持很高的辨識正確率。

**關鍵詞:** 駕駛情境辨識(Driving Scenario Recognition)、機器學習(Machine Learning)、特徵選取(Feature Selection)

駕駛情境辨識技術的重要性在於其為多種車輛應用系統,如車輛省油系統、駕駛輔助系統及未來的自動駕駛等應用的重要子系統。過去的研究[7, 8]大多將行車情境歸類為行駛環境的不同,如郊區、市區等,或是將其定義為道路類型和交通壅塞程度的不同,並採用機器學習的方式[11]來加以辨識。亦有一部份的研究著重在同樣的行為分辨上,但採取機率分析的方式,如 Amardeep [12]等人提出使用隨機馬可夫模型(Hidden Markov Model)來進行行車行為的分析。除了行車情境的辨識外,亦有研究使用機器學習中的支持向量機來偵測駕駛者未專心駕駛車輛之情境[9]。

本研究則著重於分析駕駛情境,藉由收集車輛實際上路的真實環境的資料包含行車紀錄器影像、OBD-II 及 Android 裝置感測器等,並根據影像進行分類,建立行車情境資料庫,並且透過特徵選取的方式來降低特徵向量的維度,以避免辨識時所產生的大量運算,確保在行動裝置上可即時進行辨識。然後利用機器學習的方式訓練出穩定的行車情境辨識模型。



### 一、簡介

駕駛情境辨識的技術對很多車輛應用系統,如車輛省油系統、駕駛輔助系統、及未來

表 1. 行車紀錄器規格

項目	規格
顯示螢幕	2.4" LCD 彩色液晶螢幕
感光元件	Full HD 低照度感測器 CMOS 感應器
鏡頭	127 度高廣角 焦距=2.0mm
GPS 接收模組	u-blox6 最新一代衛星接收模組 陶瓷天線 25x25x4mm
重力感測器	3 軸重力感測器
儲存裝置	SDHC 卡(最大支援 32GB)

## 二、相關研究

過去車輛資料分析相關的研究中，部分研究著重在道路類別的辨識[7, 8, 11]。其中 Jeon 等人在[7]中採用類神經網路(Neural network)的方式來進行道路類別的分類，他們將道路分成六種類別，每個類別用類神經網路的方式對不同參數取不同的權重，每次要判別道路種類時，將輸入的資料與每種類別的權重做內積運算，將六個內積的結果比較後取出最為相似的作為最後的判別結果。有的則同樣也定義了他們感興趣的行為，並對這些行為進行辨識。Amardeep [12]等人定義了三種駕駛行為，包含左轉、右轉和變換車道，並使用三種特徵：速度、方向盤角度、煞車力度，來建立隨機馬可夫模型(Hidden Markov Model, HMM)進行辨識。他們認為 HMM 在連續時間訊號分析的處理有足夠的準確度，並且認為 HMM 的隨機性符合駕駛行為的駕駛者習慣的不同、道路的影響等等不確定性。

另一類研究則同樣採用機器學習的方法來減少車輛的油耗。主要方法是對不正確的行為進行糾正，以實現節能目的。這部份的研究[1, 4-6, 10, 11, 13]對於最佳油耗量的數學模型是由理論的方法建立，採用包括線性規劃(Linear Programming)、最佳化控制(Optimal Control)和動態規劃(Dynamic programming)等方法。[11]則採用機器學習的方法，分類出 11 種道路類別和壅塞程度組合，然後根據不同的道路環境建立不同的最佳化省油模型。但是這些方法大多採用電腦模擬，缺乏實際道路測試，難以驗證其在真實世界中的效能。採用電腦模擬之主要缺點為電腦模擬無法將真實的環境因素考量進去。除了道路的實際狀況、車輛的狀況外，最重要的是駕駛人的行為難以使用電腦作準確的模擬。因此可能造成電腦模擬的實驗結果過於樂觀，而實際上路測試的效果較差。本研究因此採用實際上路時收集的資料建立辨識模型。

表 2. 感測器介紹

感測器類別	來源	描述
加速度	手機	測量裝置 x, y, z 三方向的加速度 ( $m/s^2$ )
重力	手機	測量裝置 x, y, z 三方向的重力 ( $m/s^2$ )
陀螺儀	手機	測量裝置 x, y, z 三方向的轉速 ( $rad/s$ )
方向	手機	測量裝置 x, y, z 三方向的選轉的角度
GPS	手機	紀錄經緯度、時間及速度等資訊
速度	OBD-II	車輛本身速度資訊(km/hr)
節氣門位置	OBD-II	測量節氣門位置(%)
踏板位置	OBD-II	測量踏板位置(%)
引擎負擔	OBD-II	測量引擎負擔(%)
引擎轉速	OBD-II	測量引擎轉速(rpm)
旅程油耗	OBD-II	測量一趟旅程耗費的油耗(公升)
其他	OBD-II	其他 OBD-II 可收集之資訊

多數辨識問題皆需面對資料特徵值的選取問題。根據 Erission [3]所提出的與行車行為相關的參數共計有 62 種，因此無論是面對道路類別的判別或是擁塞狀況還是行車行為，過多的特徵都有可能導致維度災難(Curse of dimensionality)，因此本研究只選定了 29 種較為重要、且現有設備即可收集到之特徵值。

## 三、研究方法

### 3.1 資料收集

資料收集所使用的實驗車輛以小客車為主。車上配備一個行車紀錄器及一個 Vgate (OBD-II adapter)及一台 Android 裝置。行車紀錄器採用 Papago P2X，規格如表 1 所示。用於拍攝影像作為判斷行車情境的基準，Vgate 則透過藍芽將 OBD-II 資料傳給 Android 裝置紀錄。同時 Android 裝置也會收集 GPS 及加速度等裝置上感測器的資料，再透過網路上傳到資料庫。

手機及 OBD-II 感測器的介紹如表 2。其中除了 GPS 資料的取樣頻率為每秒 1 次之外，其他感測器的資料皆為每秒 10 次。

行車紀錄器:收集影像資料作為標記基準。



#### OBD-II:

收集速度、節氣門位置、踏板位置、引擎負擔、引擎轉速和旅程油耗等資訊。

#### Android 裝置:

收集加速度、GPS 資訊等手機感測器資料，並記錄 OBD-II 透過藍芽傳過來的資料。

圖 1. 實驗情境示意圖

目前所收集的車輛資料以兩台為主，其詳細規格如表 3 所示。

表 3. 實驗車輛規格

	2011 Subaru Legacy Wagon 2.0i	Mitsubishi Lancer Fortis 2.0
汽缸設計	水平對臥四缸	直列四缸
汽門數	16	16
排氣量	1994cc	1998cc
最大馬力	150hp@6000rpm	157hp@6000rpm
最大扭力	20.0kgm@3000rpm	20.9kgm@4250rpm
驅動型式	四輪驅動	前輪驅動
變速系統	CVT 6 速手自排	CVT 6 速手自排
車長	4775mm	4585mm
車寬	1780mm	1760mm
車高	1535mm	1490mm
車重	1528kg	1400kg
軸距	2750mm	2635mm

目前所收集的資料可區分成以下四種資料集。各個資料集資料如表 4 所示。其中 2011 Subaru Legacy Wagon 2.0i 所收集的資料為  $A_n$ 、 $A_d$  和  $A$ 。 $A_n$  為駕駛者平時固定的上班路線， $A_d$  則為平時上班路線以外的資料， $A$  則為  $A_d$  和  $A_n$  資料的混合代表同一車種和同一駕駛的資料，Mitsubishi Fortis 收集的則為  $B$ 。

表 4. 資料集說明

名稱	道路類型	里程數 (km)	累計油耗量 (l)
$A_n$	駕駛者平日上班路線，實際上路符合真實情境。	39.34	5.45
$A_d$	駕駛者非平日上班路線	12.97	2.20
	將 $A_n$ 及 $A_d$ 整合成一個，表示為同一駕駛但不考慮不同路線的影響。	52.32	7.56
$B$	實際上路資料符合真實情境。	279.92	23.07



圖 2. 遇到紅燈時的減速行為

### 3.2 資料人工標記流程

我們的研究中，將辨識六種分類，分別是加速、減速，左轉和右轉，上坡和下坡。收集資料時，OBD-II 及手機感測器的資料及行車紀錄器的影像都會標記。在人工標記時，我們會根據行車紀錄器的影像，記錄事件發生的起始和結束時間，然後依據標記的時間在 OBD-II 及手機感測器收集的資料上進行標記。



圖 3. 遇到綠燈時的加速行為



圖 4. 左轉行為

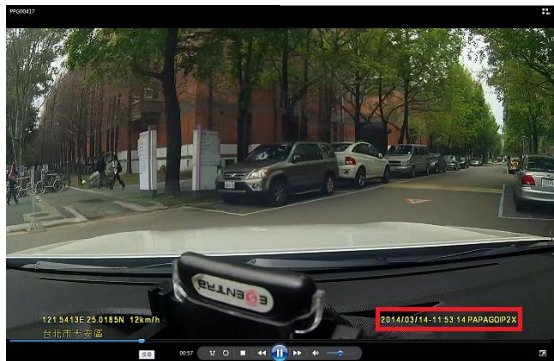


圖 5. 右轉行為

### 3.3 資料時間同步問題

當要以行車紀錄器的時間來標記手機端資料紀錄的時間時，會面臨的是時間同步的問題，如果行車紀錄器的時間和手機的時間不同步，那麼在資料標記的部分就會產生錯誤，如此便會影響我們最後建立行車辨識模型的結果。為此我們採用 GPS 時間同步的方式，因為 GPS 時間的偏差在毫秒等級，遠小於我們資料收集的頻率，因此我們可以以 GPS 時間來同步與手機端收集到的資料。具體操作中如下面兩個圖所示，手機端資料如圖 6。標記時會根據行車紀錄器影像的 GPS 時間(如圖 7 中顯示的時間)來進行標記。

	A	B	C	D	E	F	G
GPS Time	Longitude	Latitude	GPS Speed	Horizontal	Altitude	Bearing	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:27 GMT+08:00 2014	121.5096	24.99802	0	48	21.7	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:28 GMT+08:00 2014	121.5096	24.99801	0	48	25.3	49.4	
Fri Mar 14 11:38:29 GMT+08:00 2014	121.5097	24.99802	0	32	36.5	49.4	
Fri Mar 14 11:38:29 GMT+08:00 2014	121.5097	24.99802	0	32	36.5	49.4	
Fri Mar 14 11:38:29 GMT+08:00 2014	121.5097	24.99802	0	32	36.5	49.4	

圖 6. 手機端資料的 GPS 時間



圖 7. 行車紀錄器紀錄的 GPS 時間

### 3.4 特徵擷取方法

由於我們的資料是透過手機上面感測器和 OBD-II 擷取的，其中手機的感測器如 GPS 資料是每 1 秒收集一次，而 OBD-II 資料則是每 0.1 秒就收集一次，為了盡可能保存完整的資料，我們以 OBD-II 的收集頻率 0.1 秒為主來擷取資料。

myWeb

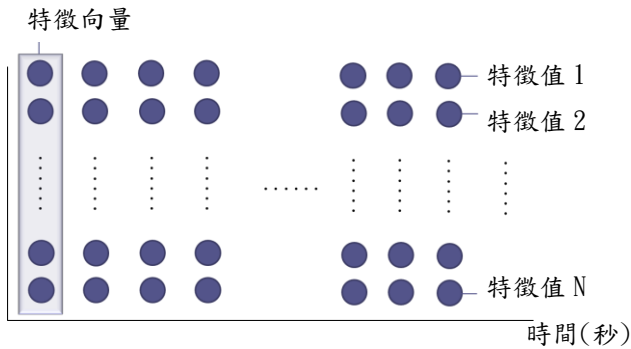


圖 8. 每 0.1 秒取得的特徵向量示意圖

目前我們共可得到 29 種特徵，其中包含收集的 GPS 資訊、加速度以及 OBD-II 的速度、引擎、節氣門和踏板深度等資訊。我們的特徵向量則如圖 8 所示。圖中每一行表示每 0.1 秒得到的特徵向量，而其中的每一個圓圈表示 29 種特徵中的其中一種特徵值。

然而由於我們所表達的駕駛情境為連續一段時間的駕駛行為，所選取的特徵向量必須包含這段時間的資訊。為此我們選取了連續一段時間的所有特徵值串接起來作為表達連續時間的特徵向量(如圖 9 所示)，如此一來特徵向量所包含的資訊量也隨之增加，可提高準確性。但是同時因為特徵向量維度的增加，可能導致訓練分類模型的難度增加，可能導致「維度災難」(Curse of dimensionality)，因此合理的選擇連續時間的窗口大小來決定維度大小，以及利用降維的演算法是必要的。

### 3.5 特徵選取

#### 3.5.1 窮舉法

窮舉法將所有可能的特徵值組合全部都計算一次，找出結果最好的組合，但是這樣的計算量非常大需要計算  $2^N - 1$  次，其中 N 表示特徵值數量。由於我們的資料特徵值數目較多，因此不宜使用此方法。

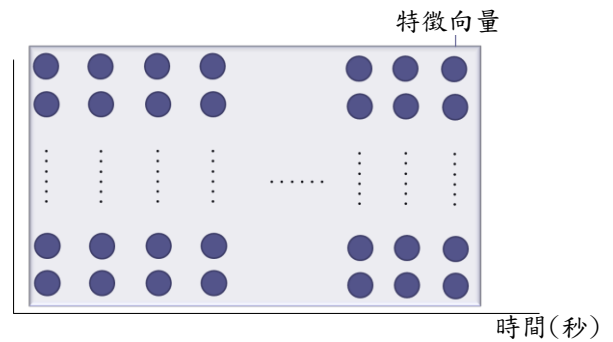


圖 9. 連續時間的特徵向量示意圖

#### 3.5.2 循序向前演算法

循序向前選取法(Sequential forward selection, SFS)是一種貪婪演算法，計算量相較窮舉法小很多。演算法可分為以下四個步驟：

步驟一、使用 k 近鄰演算法(K Nearest Neighbor Classification, KNNC)和一次挑一個(Leave-One-Out, LOO)辨識率預估法。

步驟二、第一個挑選的特徵一定是辨識率最高的特徵。

步驟三、下一個挑選的特徵一定是和原本已選取的特徵合併後，辨識率最高的特徵。

步驟四、重複步驟三，直到出現準確率下降的情況出現，即可獲得最大值，用這樣的方式即可挑選出最好的一組特徵組合。

此方法最大的優點為計算量較小，但是由於是單獨針對各個特徵值去做計算，因此其缺點是並沒有考慮特徵值之間的關係，容易造成局部最佳化的情況。

### 3.6 分類器演算法

#### 3.6.1 支持向量機

支持向量機(Support Vector Machine, SVM)，為一種監督式學習(Supervised Learning)的演算法，而監督式學習是一種機器學習方法，是指由訓練資料中建立一個模型，而後即可依此模型來判別未被分類的資料。訓練資料通常由特徵向量及預期對應的輸出所組成，輸出部分可以是一個連續的值(Regression Analysis 迴歸分析)或是預測一個分類標籤(Classification, 分類)。而 SVM 的主要方法為使用訓練資料找出一個具有最大邊界的超平面(Maximum Marginal

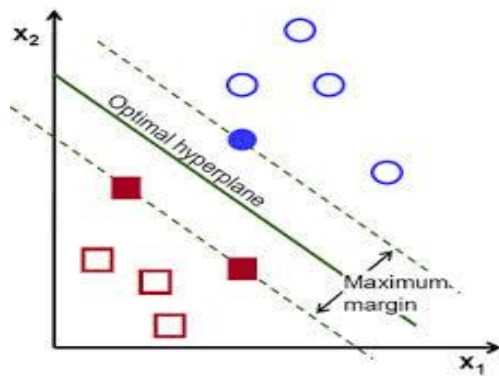


圖 10. 支持向量機二維示意圖

Hyper-Plane), 將不同類別的集合分開, 如圖 10 所示。之後便以此超平面作為分類的基準。稱為支持向量機之主要原因是因為支持向量(Support Vector)是使用訓練資料建立模型時找到的最大邊界上的點, 因此其也具有較高的分類準確性。以二維的例子來說, 我們希望找到一條線能夠將紅點與藍點分開, 而且還希望邊界越大越好, 如此以來我們才能夠很明確的去分辨這些資料是屬於不同的類別。

### 3.6.2 決策樹

決策樹是一種建構出樹狀結構來達成分類的效果。通常藉由計算熵(Entropy)和資訊量(Information Gain)的方式來決定分類樹的節點, 分類的結果則會出現在葉節點(Leave Node)上。從根(Root)走到葉節點的路徑就是分類樹演算法的分類規則。在本研究中採用的是 J48 演算法, 其優點為簡單、快速且準確率高等。

## 四、實驗結果

我們使用前面所提到的演算法建立了多類分類辨識(multi-class classification)模型。

我們將針對交叉驗證正確率及測試正確率兩個數值來看模型的穩定度。其中交叉驗證正確率我們採用的是 10 次交叉驗證(10-fold Cross Validation), 而測試正確率則是使用不同的兩個資料集一個做為建立模型使用, 另一個用來做測試模型的準確率。

表 5. 支持向量機交叉驗證正確率

訓練資料集	資料筆數 (個)	交叉驗證正確率(%)	建立模型時間(秒)
A	6653	93.04	14.3
A <sub>n</sub>	4483	91.86	7.55
A <sub>d</sub>	2050	95.95	1.89
B	4350	95.77	6.73
A+B	10883	94.13	66.03

表 6. 支持向量機測試正確率

訓練資料集	測試資料集	測試正確率 (%)
A	B	32.18
B	A	26.51
A <sub>d</sub>	A <sub>n</sub>	42.85
A <sub>n</sub>	A <sub>d</sub>	44.88

### 4.1 支持向量機

這部份我們採用 LIBSVM[3], 核心函數使用 Polynomial kernel。支持向量機的交叉驗證正確率如表 5, 從表 5 可以發現, 交叉驗證正確率皆高於 90%, 這表示我們有辦法針對同一台車收集一段時間的資料, 之後將那些資料用來建立模型, 然後給同一台車使用, 該模型能夠有很高的正確率; 同樣的, 針對同一路徑來看, 若今天模型的使用情境為一貨運司機或上班族固定的路線, 我們的模型同樣也很適合這樣的情況。另一部分測試正確率如表 6 所示, 從中我們可以觀察到的現象有當以不同車種收集來的資料(A、B)或是由不同路徑(A<sub>d</sub>、A<sub>n</sub>)互相作為建立及測試模型的資料, 其測試正確率都不高, 從這樣的結果我們可以知道, 當不同車種或是不同路徑時, 其所收集到相同情境的資料可能具有非常不同的特性, 也因此導致模型無法準確辨認這些情境。若將來想針對這部分進行處理, 可以採取的將路徑及車種同樣作為一個特徵值放進特徵向量中來建立辨識模型, 但目前我們所收集的資料較為有限, 無法驗證此一部份。

表 7. 決策樹交叉驗證正確率

訓練資料集	資料筆數 (個)	交叉驗證正確率(%)	建立模型時間(秒)
A	6653	97.92	0.41
A <sub>n</sub>	4483	98.57	0.54
A <sub>d</sub>	2050	98.68	0.12
B	4350	99.43	0.16
A+B	10883	98.47	1.17

表 8. 決策樹測試正確率

訓練資料集	測試資料集	測試正確率 (%)
A	B	5.75
B	A	22.30
A <sub>d</sub>	A <sub>n</sub>	58.02
A <sub>n</sub>	A <sub>d</sub>	55.80

#### 4.2 決策樹

我們採用的是 J48 決策樹。決策樹的交叉驗證正確率結果如表 7。其正確率較 SVM 的結果高且建立模型的速度非常快。從測試正確率來看，當資料集為同一台車時，如 A<sub>d</sub>、A<sub>n</sub>，其正確率相較 SVM 來說提高了 10% 左右，然而針對不同的車種時，其測試正確率卻大幅降低。由此可見，在訓練資料集及測試資料集車種不同的情況下，決策樹並不適合用來作為建立模型的演算法。但從交叉驗證正確率的結果來看，決策樹較適合做為針對同一車種或同一路徑時，建立模型的演算法。

#### 4.3 特徵選取

我們嘗試使用前面提到的循序向前選取法在資料集 A 上，其選取出的四種特徵分別是扭力(Torque)及三種油耗相關的參數(使用油耗量、MPG)，可將 29 維的特徵向量降成 4 維。但交叉驗證正確率僅從 97.78% 降成 93.49%，仍有相當不錯的正確率。

### 五、結論

在此研究中，我們嘗試使用機器學習的方式來建立一個行車情境辨識模型，其中，我們收集了不同的兩台車的資料利用行車紀錄器及 OBD-II 和 Android 裝置，收集了多種感測器的資料及行車紀錄器影像來建立一個行車情境資料庫。從實驗的結果發現，我們可以

針對同一車種或同一路段能有很高的辨識正確率，適用於貨車司機或是上班族等有固定行車路線的使用者。另外針對不同路線或不同車種時，我們未來收集較多車種和路段的資料時，可以採取將路線或車種同樣作為特徵放入特徵向量中的方式來做處理，預期可以提高辨識率。另外從特徵選取的結果發現，我們仍能維持 90% 以上的正確率，但維度可以從 29 維降低到 4 維，這對將來移植行車情境辨識模型到 Android 裝置上時能夠有很大程度的幫助，可以大幅降低計算量並且可以降低需要的感測器個數，減少安裝的成本。另外本研究所提出的行車情境辨識模型未來也可以幫助節能系統、輔助駕駛系統和自動駕駛等研究的發展。

### 參考文獻

- [1] J. A. Crossman, H. Guo, Y. L. Murphey and J. Cardillo, "Automotive Signal Fault Diagnostics-Part I: Signal Fault Analysis, Signal Segmentation, Feature Extraction and Quasi-Optimal Feature Selection," *IEEE Trans. on Vehicular Technology*, vol. 52, pp. 1063-1075, 2003
- [2] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," *ACM Trans. on Intelligent Systems and Technology*, 2011, Software available at the following url: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- [3] E. Ericsson, "Independent Driving Pattern Factors and Their Influence on Fuel-Use and Exhaust Emission Factors," *Transportation Research Part D: Transport and Environment, Elsevier*, vol. 6, pp. 325-345, 2001
- [4] E. Ericsson, H. Larsson and K. Brundell-Freij,

- “Optimizing Route Choice for Lowest Fuel Consumption--Potential Effects of a New Driver Support Tool,” *Transportation Research Part C: Emerging Technologies, Elsevier*, vol. 14, pp. 369-383, 2006
- [5] F. Ferri, P. Pudil, M. Hatef and J. Kittler, “Comparative Study of Techniques For Large-Scale Feature Selection,” *Machine Intelligence and Pattern Recognition, Citeseer*, vol. 16, pp. 403-403, 1994
- [6] J. N. Hooker, “Optimal Driving for Single-Vehicle Fuel Economy,” *Transportation Research Part A: General, Elsevier*, vol. 22, pp. 183-201, 1988
- [7] S. Jeon, S. T. Jo, Y. Park and J. M. Lee. “Multi-Mode Driving Control of a Parallel Hybrid Electric Vehicle Using Driving Pattern Recognition”, *Dynamics System Measurement Control*, vol. 124, pp. 141-149, 2002
- [8] C. C. Lin, S. Jeon, H. Peng and J.M. Lee, “Driving Pattern Recognition for Control of Hybrid Electric Trucks,” *Vehicle System Dynamics*, Vol. 42, nos. 1-2, pp. 41-58, 2004
- [9] Y. Liang, M. L. Reyes and J. D. Lee, “Real-time detection of driver cognitive distraction using support vector machines,” *IEEE Trans. Intelligence Transportation System*, vol. 8, no. 2, pp. 340–350, Jun. 2007
- [10] Y. L. Murphey and H. Guo, “Automatic Feature Selection-a Hybrid Statistical Approach.” *Proc. International Conference on Pattern Recognition*, vol. 2, pp. 382-385, 2000
- [11] J. Park, Z.Chen, L. Kiliaris, M. L. Kuang, A. Masrur, A. M. Phillips and Y. L. Murphey, “Intelligent Vehicle Power Control Based on Machine Learning of Optimal Control Parameters and Prediction of Road Type and Traffic Congestion,” *IEEE Trans. on Vehicular Technology*, vol. 58, pp. 4741-4756, 2009
- [12] A. Sathyanarayana, P. Boyraz and J. H. Hansen, “Driving Behavior Analysis and Route Recognition by Hidden Markov Models,” *Proc. IEEE Int. Conf. Vehicle Electron Safety*, pp. 276-281, 2008
- [13] A. Schwarzkopf and R. Leipnik, “Control of Highway Vehicles for Minimum Fuel Consumption over Varying Terrain,” *Transportation Research, Elsevier*, vol. 11, pp. 279-286, 1977

