

# 「國研盃 i-ONE 儀器科技創新獎」 得獎作品介紹 2020 年中學組首獎

## 醫療代步車-AI 輔助煞車系統

### Mobility Scooter AI Braking Assistant

桃園市立內壢高級中等學校團隊：羅敦允、賴宥蓁、李文翔、  
陳彥儒、陳安儒、呂盈臻

指導老師：廖經益、曾秀燉

#### 一、摘要

本研究因考量數據統計與分析的可視性和易觀察性，因此使用自製改裝腳踏車模擬醫療用代步車，並針對圍繞著輔助煞車系統的多項功能進行實驗與研發。其中功能有：(1) 多個感測器與攝影機結合，提高系統識別精準度。(2) 道路視覺辨識，輔助長輩判別道路即時路況。(3) 軟硬體整合，App 顯示車體即時資訊。(4) 馬達作動轉角與煞車力道關係。(5) 即時測量前方與兩側車距，並在意外突發時，啟動輔助煞車。(6) 依據不同環境參數，計算煞車力道、與作動滑行距離與時間。

#### 二、背景介紹

臺灣於 2018 年邁入「高齡社會」，老年人口達到總人口的 14%。國發會推估在 2026 年臺灣將邁入「超高齡社會」老年人口達到總人口的 20%。交通部於 2017 年起開始管理高齡者的駕照，因此不需駕照又同樣是四輪的醫療用電動代步車，成了高齡長輩的新選擇。但依照交通部解釋，「醫療用電動代步車」、「動力式輪椅」屬於醫療器材，視為「行人」活動之輔助器材。但許多長者卻將醫療電動代步車當成汽、機車般的直駛在道路上。因長輩視聽力等生理機能已相較退化，反應不如常人，老人醫療代步車，成了道路上的潛在危機。

因此，本研究的理念是製作一款：AI 輔助煞車系統。利用各式感測器及攝影機的影像辨識功能偵測周遭環境路況，並與軟體整合，讓輔助煞車系統可以應對騎乘中的突發狀況，並有效降低行車致死率、提升駕駛騎乘的安全性。

### 三、研究目的

本研究的主旨為透過設計、自製出「輔助煞車系統」並有效降低醫療代步車的車速。藉由電動腳踏車蒐集數據，我們可針對更快速的車速進行數據分析與測量。透過自製傳動、供電系統，我們將車體硬體智慧化，將各部件連網，並於每次啟動前進行自身車體檢測，同時利用手機軟體開發智能儀表板，透過語音和圖示的提醒，更有效的提升長輩的用路安全，並輔助駕駛。此外，我們為了解決家屬擔憂長輩行車安全的問題，我們同時開發家屬專用軟體，家屬可透過軟體得知長輩的即時車況與位置，為了使家屬能在意外發生時掌握即時狀況，我們透過車體上的多重感測器比對，推測意外狀況，並透過簡訊通知家屬。

### 四、作品實現之過程與步驟

本次作品之研究可分為四大架構，分別為：「輔助煞車時的原理與運用」、「輔助煞車運作流程與設計」、「感測器與車體檢測」、「影像辨識」。

#### 1. 輔助煞車的物理原理與運用

為了確切的計算出騎乘的驅動力與煞車時的制動力，本節針對輪胎與地面所產生的滾動阻力、騎乘時的空氣阻力、非水平面所產生的坡度阻力進行討論與研究，並透過感測器取得即時環境參數，最終得出加速阻力，透過加速阻力，我們可計算出制動時所產生的負加速度，並推測出制動所需時間以及車體滑行距離。

##### 1.1 輔助煞車：車體行駛時的受力分析

- 驅動時所產生的阻力

根據達蘭伯原理 (D'Alembert's principle) 對車體行駛時的力圖分析可得知，移動的行駛方程為：

$$F_t = F_f + F_i + F_w + F_j \quad (1)$$

其中加速阻力與其它阻力共同構成平衡力系，並與車體本身的驅動力相反且相等，如 (圖 1)。

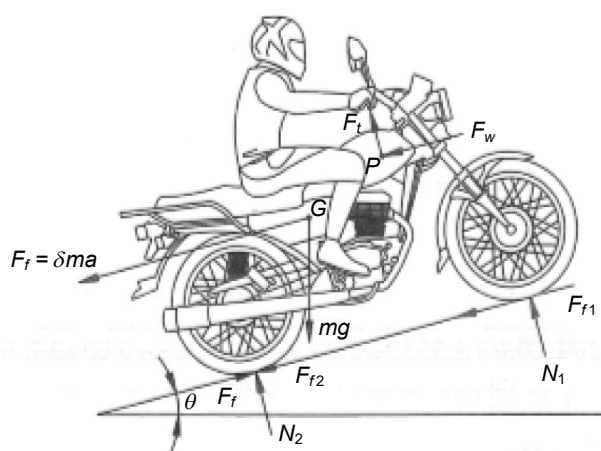


圖 1. 車體行駛時的力圖分析圖。

## 1.2 輔助煞車：移動行駛方程中的各力分析

### 1.2.1 滾動阻力

$F_f$  (滾動阻力)：如圖 2，滾動阻力是因為車輪並非正圓，當車體在移動時會因為重力產生阻礙輪胎向前滾動的阻力，計算公式為：

$$F_f = F_{f1} + F_{f2} = N_1 f + N_2 f = (N_1 + N_2) f = W f \quad (2)$$

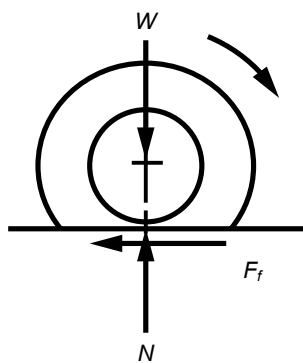


圖 2. 滾動阻力。

### 1.2.2 坡度阻力

$F_i$  (坡度阻力)：如圖 3，坡度阻力會依照所處環境的傾角，造成移動時的正或負加速度，我們利用三軸感測器，去感測環境坡度，並透過程式計算出因坡度所產生的正、負加速度，計算公式為：

$$F_i = mg \sin \theta \quad (\theta = 5\% , a \pm 0.5g) \quad (3)$$

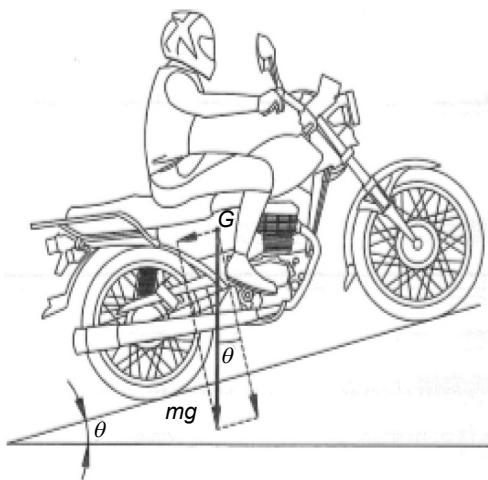


圖 3. 坡度阻力。

### 1.2.3 空氣阻力

$F_w$  (空氣阻力)：空氣阻力與人車迎風受力面積、環境空氣密度、溫度、風速有關，因影

響與其他阻力相比較小，故我們假設所處環境為室溫 25 °C，和無風狀態，進行計算，計算公式為：

$$F_w = \frac{1}{2} \rho A C_d V^2 (N) \quad (4)$$

#### 1.2.4 加速阻力

$F_j$  (加速阻力)：如圖 4，根據達蘭伯原理摩托車在加速過程中，會受到一個與加速度  $a$  相反的力，因此稱為加速或慣性阻力，它屬於車體移動時平衡力系的一環，當車速提升時，會形成阻力偶。

將轉動件的旋轉慣性換成等效的平移慣性： $\delta$  [旋轉質量換算係數 (Mass Factor)]  $\delta = 1.08$ ，計算公式為：

$$F_j = \delta ma \quad (5)$$

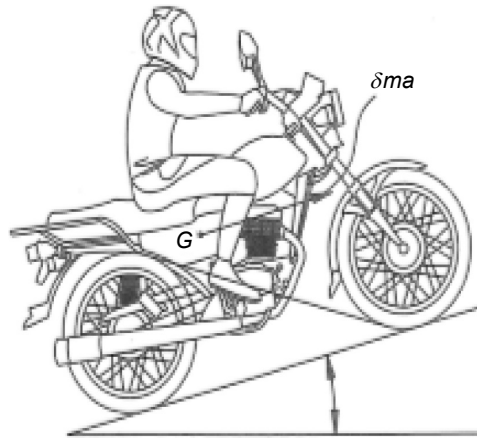


圖 4. 加速阻力。

### 1.3 輔助煞車：車體制動時的受力分析

· 加速阻力與煞車力度的計算

煞車時，先前在驅動力中所提到的阻力反而會變成煞車力道的助力，換句話說在煞車時，除了制動力為主要作用力之外，滾動、空氣、坡度阻力也能增加其煞車力道。我們透過公式計算出加速阻力與煞車力度，並確保車體能在安全時間內有效煞車。由公式可看出：

$$\begin{aligned} -(F_{b1} + F_{b2}) - (F_{f1} + F_{f2}) - F_w + mg \sin \theta + F_j &= 0 \\ \rightarrow F_b + F_f + F_w &= F_i + F_j \end{aligned} \quad (6)$$

透過以上公式，並利用車上的獨立系統運算，我們可得出加速阻力的負加速度，同時算出輔助煞車系統所需作動的時間以及煞車距離，並與前車距離比對，系統將計算出煞車所需的制動力  $F_b = T/R \Rightarrow F = \mu N \Rightarrow 0.5N \Rightarrow N(\text{kg})/2$  (平均一組制動器所需施加的力道)。

#### 1.4 輔助煞車：制動時地面正向反作用力的變化

• 地面作用於前、後輪的正向反作用力公式推導：

當車體制動時會因為慣性的緣故產生載荷轉移，影響車體制動時的穩定性，為了計算出前、後輪的最佳制動力，本節參考圖 5，當車體制動時透過對前、後輪與路面接觸中心 A、B 取矩，可得出地面作用於前、後輪的正向反作用力：

$$N_1 = \frac{W \times \cos \theta \times \ell_2 + \delta m a \times h - F_w \times h_w + W \sin \theta \times h}{L} \quad (7)$$

$$N_2 = \frac{W \times \cos \theta \times \ell_1 + \delta m a \times h - F_w \times h_w + W \sin \theta \times h}{L} \quad (8)$$

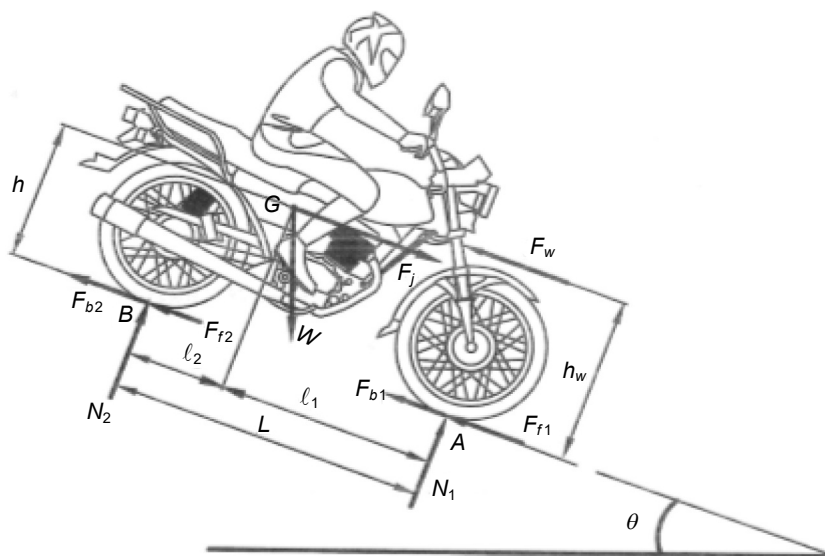


圖 5. 車體煞車制動時的力圖分析。

假設空氣阻力通過「人—車系統」的重心，此時  $h_w = h$ 。因此地面作用於前後輪的正向反作用力變成：

$$N_1 = \frac{1}{L} [W \cos \theta \ell_2 + h(\delta m a - F_w \times h_w + W \sin \theta)] \quad (9)$$

$$N_2 = \frac{1}{L} [W \cos \theta \ell_1 + h(\delta m a - F_w \times h_w + W \sin \theta)] \quad (10)$$

由方程式 (9)、(10) 中的後項  $(\delta m a - F_w + W \sin \theta)h/L$  又稱為制動載荷轉移力，若車體在平地行駛且忽略空氣阻力，則  $F_w = 0$ ， $\cos \theta = 1$ ， $\sin \theta = 0$ ，上述兩式則變成：

$$N_1 = \frac{1}{L} \left( W \ell_2 + h \frac{w}{g} \delta a \right) = \frac{W}{L} \left( \ell_2 + \frac{\delta a}{g} \right) \quad (11)$$

$$N_2 = \frac{1}{L} \left( W \ell_1 + h \frac{w}{g} \delta a \right) = \frac{W}{L} \left( \ell_1 + \frac{\delta a}{g} \right) \quad (12)$$

由上述方程式可得知，當  $a$  值越大 (制動力越大)，前輪地面正向反作用力  $N_1$  越大，而後輪地面正向反作用力  $N_2$  越小。設輪胎的附著係數相同 ( $\varphi_1 = \varphi_2 = \varphi$ ) 且前後輪同時達到附著極限，車體最大制動減速度為  $a = \varphi g$ ，帶入方程式 (11)、(12)，得：

$$N_1 = \frac{W}{L}(\ell_2 + h\delta\varphi) \quad (13)$$

$$N_2 = \frac{W}{L}(\ell_1 + h\delta\varphi) \quad (14)$$

由上述公式可得知，在最大附著力時，附著係數  $\varphi$  越大，制動時前輪地面正向反作用力越大，後輪地面正向反作用力愈小。

### 1.5 輔助煞車：最佳制動力的推導與運用

• 前、後輪制動百分比

因為車體兩輪中心與重心的水平距離不同，因此造成兩輪有不同的附著係數，為了計算出兩輪的最佳制動力，帶入方程式 (6)，得：

$$F_i + F_j - F_w = F_b + F_f \quad (15)$$

帶入上述阻力公式，可寫成：

$$\delta ma - F_w + W \sin \theta = F_b + fW \quad (16)$$

將方程式 (16) 帶入方程式 (9)、(10)，可得：

$$N_1 = \frac{1}{L}[W \cos \theta \ell_2 + h(F_b + fW)] \quad (17)$$

$$N_2 = \frac{1}{L}[W \cos \theta \ell_1 + h(F_b + fW)] \quad (18)$$

設兩輪的附著係數相同 ( $\varphi_1 = \varphi_2 = \varphi$ ) 且前後輪同時達到可能的最大制動力，此時車體的最大制動力為：

$$F_{b \max} = F_{b1 \max} = F_{b2 \max} = \varphi_1 N_1 + \varphi_2 N_2 = \varphi(N_1 + N_2) = \varphi W \cos \theta \quad (19)$$

將方程式 (19) 代入方程式 (17)、(18)，可得前、後輪最大制動力：

$$F_{b1 \max} = \varphi N_1 = \frac{\varphi W}{L}[\cos \theta \ell_2 + h(\varphi \cos \theta + f)] \quad (20)$$

$$F_{b2 \max} = \varphi N_2 = \frac{\varphi W}{L}[\cos \theta \ell_1 + h(\varphi \cos \theta + f)] \quad (21)$$

經過公式推導並將其程式化後，我們的輔助煞車系統能有效地依照所處的環境參數，計算出前後碟煞所需的最佳制動力比例，並透過調整馬達轉角，分別制動前後輪的煞車模組，達到最高效益的煞車。

最佳制動力：

$$\frac{K_{b1}}{K_{b2}} = \frac{F_{b1 \max}}{F_{b2 \max}} = \frac{N_1}{N_2} = \frac{\cos \theta l_2 + h(\varphi \cos \theta + f)}{\cos \theta l_1 - h(\varphi \cos \theta + f)} \quad (22)$$

(其中  $K_{b1}$  與  $K_{b2}$  分別為前、後輪制動力所佔全部制动力的百分比)

## 2. 輔助煞車運作流程與設計

### 2.1 制動流程

我們透過將感測器與攝影機的資料交互比對以提升辨識精準度，並針對感測器與影像辨識的數值做危險程度析。當危險發生時，系統會參考周遭環境數值，經由程式計算，並啟動漸進式的輔助煞車，同時透過智能儀表板來警示駕駛。

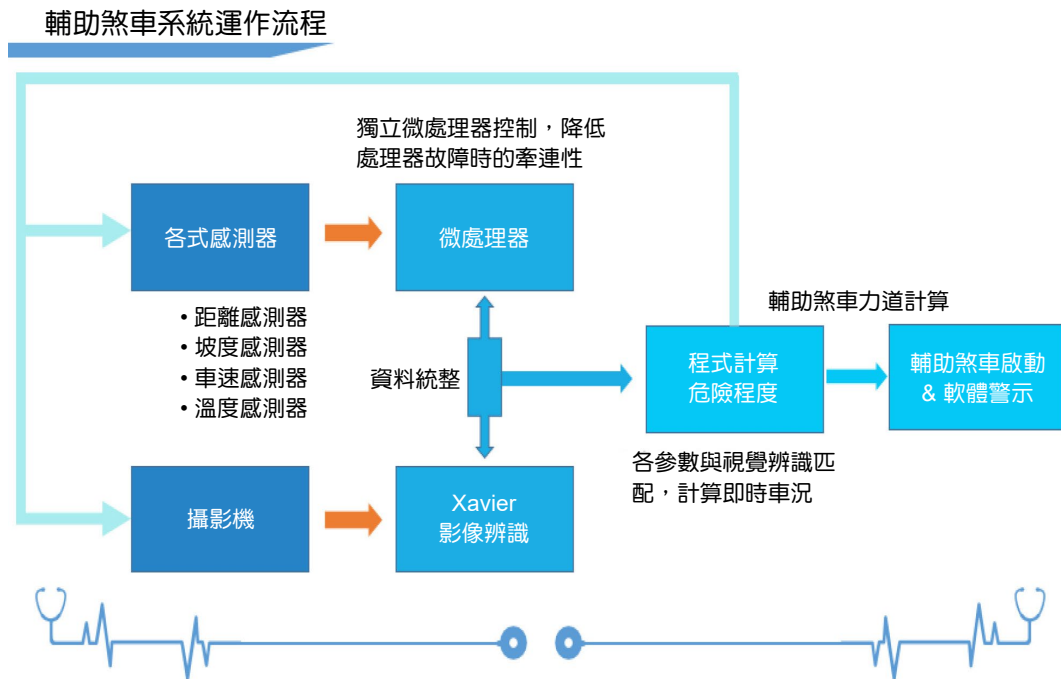


圖 6. 系統運作流程圖。

### 2.2 輔助煞車設計與配置

#### 2.2.1. 主動式煞車和協助式煞車

我們將輔助煞車系統分成兩大功能:主動式煞車和協助式煞車。其中主動式煞車是當危險情況發生時，駕駛未及時制動，系統將啟動輔助煞車並緩速；而協助式煞車是當駕駛按下煞車後，系統若判斷車體無法在有效時間內緩速，將計算出最佳的煞車力道並輔助緩速。

#### 2.2.2. 設計與配置

因前輪有避震器，導致前輔助煞車系統支點較少且集中於前叉。而紅色及藍色圈處為主要支點，綠色圈處則因避震前叉相對於地面傾角較大，僅能作為輔助支撐來穩定結構。因腳

踏車後半部支點較多，故結構較前輔煞穩定。紅色圈處的扣環可抵消前後的分力，綠色圈處的扣環固定在後上叉，可抵銷軸力及彎矩，藍色圈處的支點可抵銷上下的分力及穩定碟煞。

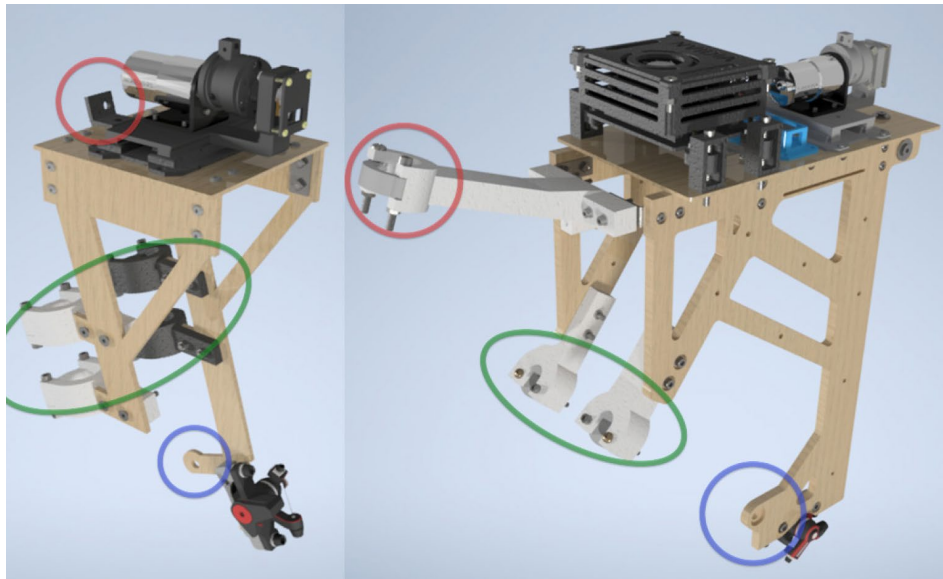


圖 7. 前後輔助煞車系統。

### 3. 感測器與車體檢測

#### 3.1 系統穩定性

##### 3.1.1. 感測器分區

透過獨立的 MCU 做資料處理與分析，我們將各式感測器回傳之數值透過中央資料處理開發板彙整，接著與影像處理器 Xavier 做資料比對，如圖 8。

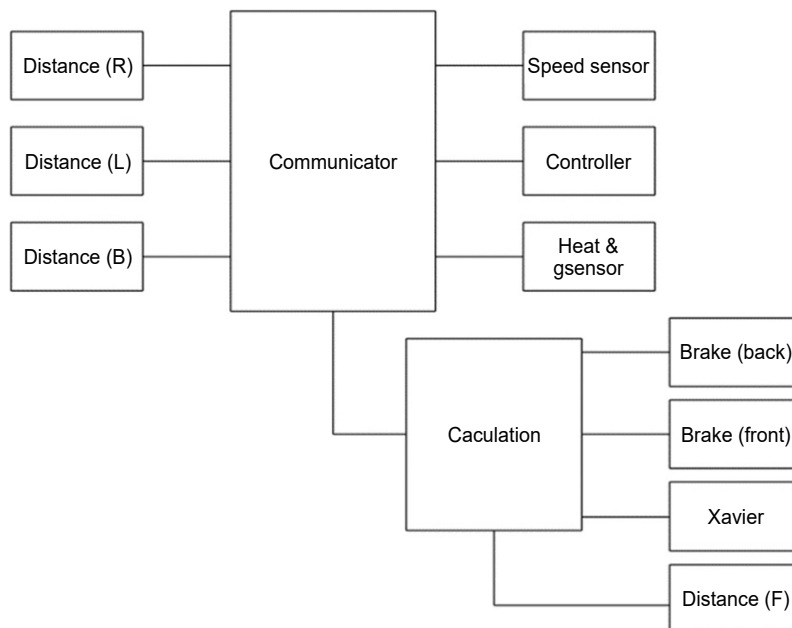


圖 8. 各式 MCU 架構圖。



如此一來，整個系統的運行速度與系統穩定性皆會提升，如圖 9

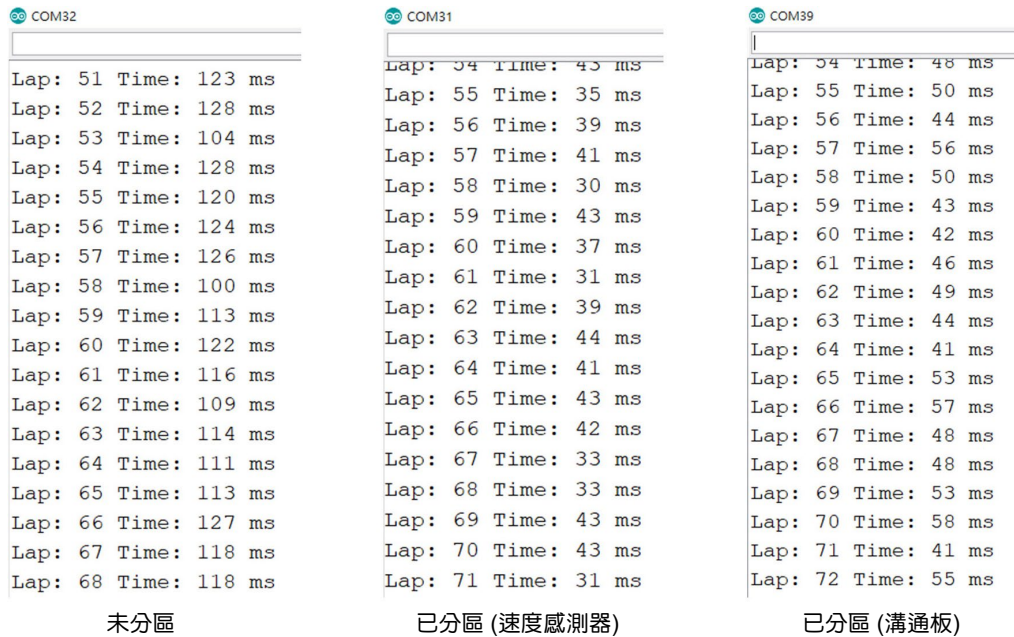


圖 9. 微處理器未分區與分區的速度比對。

### 3.2 車體檢測

長者的行車安全除了道路上的危險之外，還有可能因車體任一感測器故障而導致的事。為了避免這種情況發生，我們使代步車在開機時先做全車檢查，確保每個感測器模組無誤且正常運作。當有故障情況發生的時候，系統會馬上通知家屬，檢查流程如下。

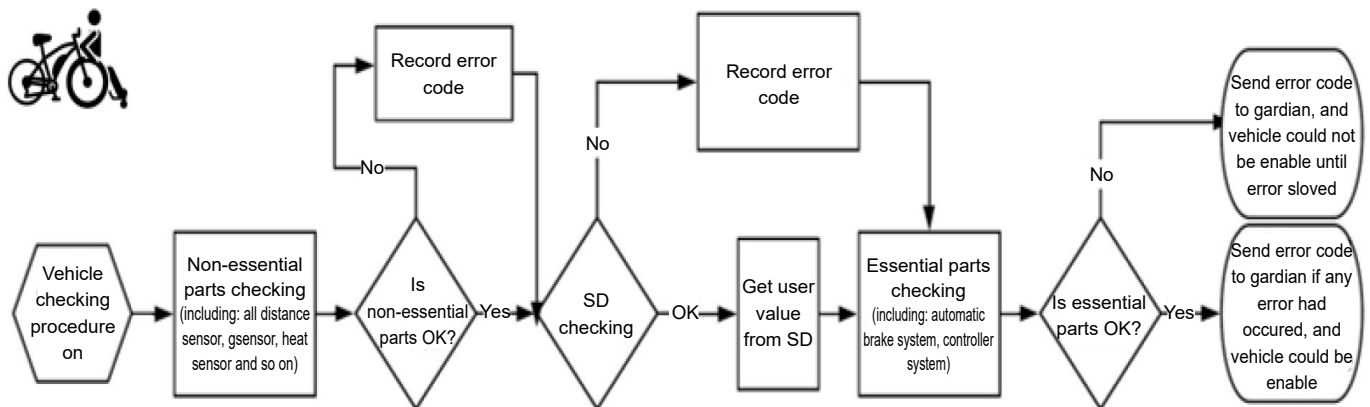


圖 10. 系統檢查流程圖。

## 4. 影像辨識

### 4.1 車體偏移偵測

我們先將圖片轉成灰階並先做初步的模糊處理，而我們使用高斯模糊，以減少圖片的雜

訊、降低細節層次，求出一個較為平滑的圖再做 Canny 邊緣檢測。接著使用 fillPoly 遮罩非路線位址的區域 (ROI)，減少不必要的資料，在這裡我們取可能為道路的梯型區域，如 (圖 24)。至於不在讀入圖片的第一步就 ROI 的原因是，避免切割的梯形邊緣被 Canny 檢測出來造成誤差。

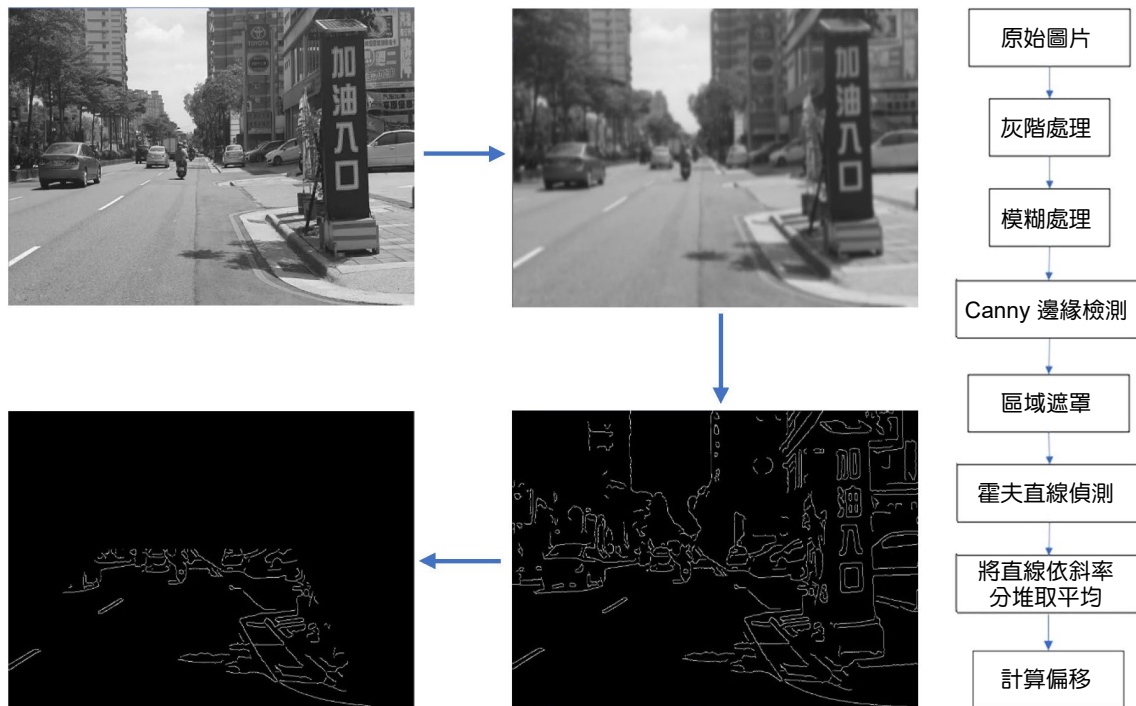


圖 11. 邊緣檢測前置流程圖。

### 原理 1. 高斯模糊 (Gaussian blur)

相較於簡單的平均模糊，高斯模糊使用常態分布的方式來分配卷積的權重，也就是越靠近的點關係越密切、權重越大；越遠離的點關係越疏遠、權重越小。而常態分布的密度是使用高斯函式，透過高斯函數  $G(x, y)$  和圖片  $F(x, y)$  做卷積，以分配權重的方式來保留較重要的數值和細節，避免被較遠的極值所影響。

$14 \times 0.0947416$	$15 \times 0.118318$	$16 \times 0.0947416$
$24 \times 0.118318$	$25 \times 0.147761$	$26 \times 0.118318$
$34 \times 0.0947416$	$35 \times 0.118318$	$36 \times 0.0947416$

圖 12. 空間和權重的關係。

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

圖 13. 高斯函數。(圖片來源：Itread<sup>01</sup>)

## 原理 2. Canny 邊緣檢測 (Canny Edge Detection)

Canny 是一個複合式演算法，結合了梯度偵測、非最大值抑制 (non- maximum suppression) 與邊界判別。並具有低錯誤率、定位準確等優點。

### Step1: 梯度偵測

一開始會使用 Sobel filter 來計算  $x$ 、 $y$  方向分別的梯度值，同時計算出梯度的方向 ( $\theta$ ) 和強度 ( $G$ )。

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad G = \sqrt{G_x^2 + G_y^2}$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \quad \theta = \arctan\left(\frac{G_y}{G_x}\right)$$

圖 14. Sobel filter 梯度計算。(圖片來源：hby coding academic)

### Step2: 非最大值抑制 (non- maximum suppression)

為了簡化計算，將梯度分成  $0^\circ$ 、 $45^\circ$ 、 $90^\circ$ 、 $135^\circ$  四個主要的方向，再做非最大值抑制，保留該方向中梯度變化 (強度  $G$ ) 最大的點作為邊緣。

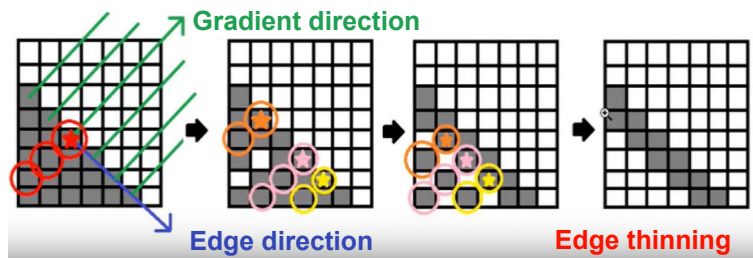


圖 15. 非最大值抑制。(圖片來源：hby coding academic)

### Step3: 邊界判別

設定高邊界和低邊界來篩選出正確的邊線：

1. 高於高界線：一定是邊緣。
2. 低於低界線：一定不是邊緣。
3. 介於高、低界線之間：若附近有高界線的点，則此點也視為邊緣。

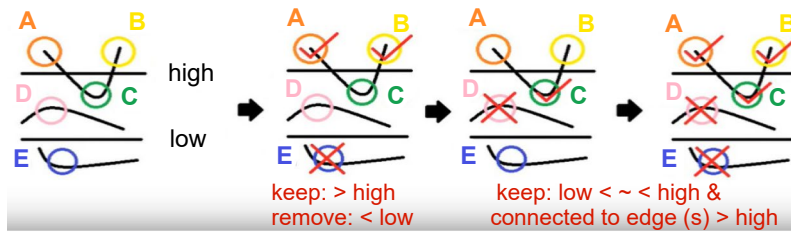


圖 16. Canny 邊界設定值篩檢示意圖。(圖片來源：hby coding academic)

因為臺灣的交通環境較複雜，故需較明顯的降噪效果，經過多次的修正和測試，最終我們使用高斯模糊範圍較大 kernel (模糊效果較顯著) 加上較低閾值的邊緣檢測參數。參數數值如下：

```
GaussianBlur (img_gray, (17, 17), 0)
Canny (img_gray_blur, 30, 60)
```

圖 17. 高斯模糊及 Canny 邊緣檢測參數。

接著我們利用 Canny 檢測出邊線的點做霍夫直線偵測 (Hough line detection)，得到許多條可能符合的線之後，再依照斜率、位置等資料設立篩選條件，以移除錯誤的結果減少誤差，如：

1. 斜率絕對值大於 0.3、小於 10。  
→ 移除偵測到車體或樹葉陰影的橫線
2. 斜率為正 (左側邊線) 且最左邊的位置在畫面中心右側。
3. 斜率為負 (右側邊線) 且最右邊的位置在畫面中心左側。  
→ 移除不可能為路線的直線

最後再依照斜率正負，分別取其平均區分為左線及右線。計算左右邊線中點和畫面中心的差以及左右線段距離的比值得出車體的偏移值。

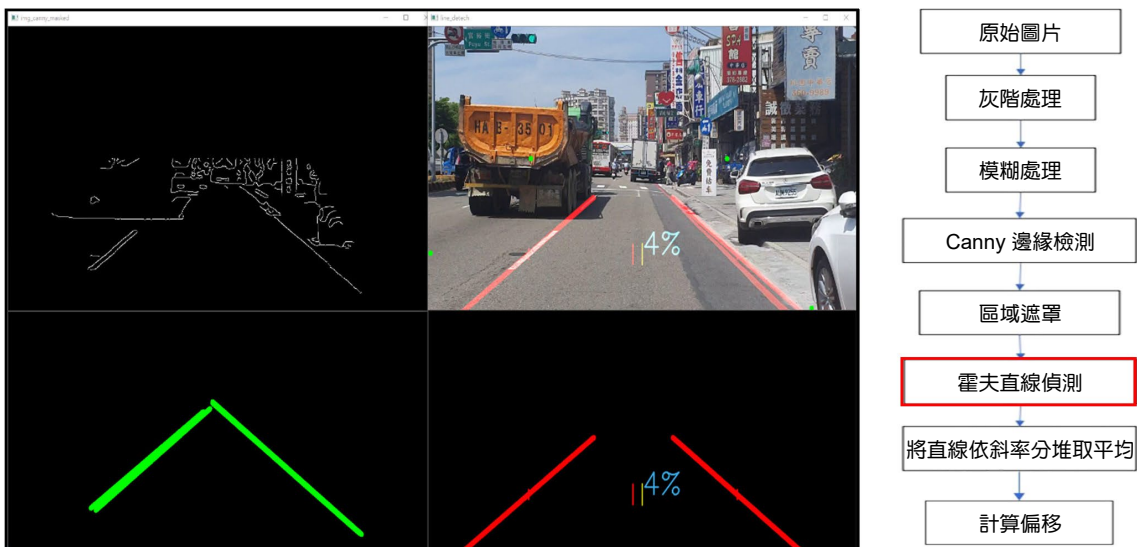


圖 18. 路線偵測和計算車體偏移。

## 4.2 紅綠燈辨別

為了加快處理器的運算速度並減少路面的干擾，我們在一開始讀入圖片時就擷取上方紅綠燈可能出現的位子(紅框)，聚焦在少量且必要的資料上。

因為要做顏色的特徵擷取，故我們先將圖片轉成 HSV 表示法，並且設定紅、黃、綠燈的 HSV 顏色範圍，之後分別將三個顏色以 inRange 二元黑白的方式表示(如圖 19 左側三個黑底的視窗，由上至下分別為綠、黃、紅)。



圖 19. 紅綠燈辨別。

## 4.3 人車辨識

本次研究在物件辨識的主要應用是用於即時路況偵測，因此在一開始研究時，我們首要考量的需求便是「運算速度」。故我們使用了 YOLO 這個演算法，搭配 darkflow 的框架。

YOLO (You Only Look Once) 相較於：

1. R-CNN 使用 Selective Search 來篩選 2,000 個可能區域 (region proposals)，再依照這些區域去分類，跑過至少 2,000 次的 CNN。

2. Fast R-CNN 使用 RoIPooling 的方式，讓這 2,000 多個區域共用，全部只算一次 CNN。
3. Faster R-CNN 直接從 feature map 上選出可能區域的 RPN (region proposal network)。

R-CNN 家族使用先提出可能區域，再針對這些提出的區域使用 CNN 分類，最後以 regression 修正 bounding box 位置這樣的方式，速度快了許多。

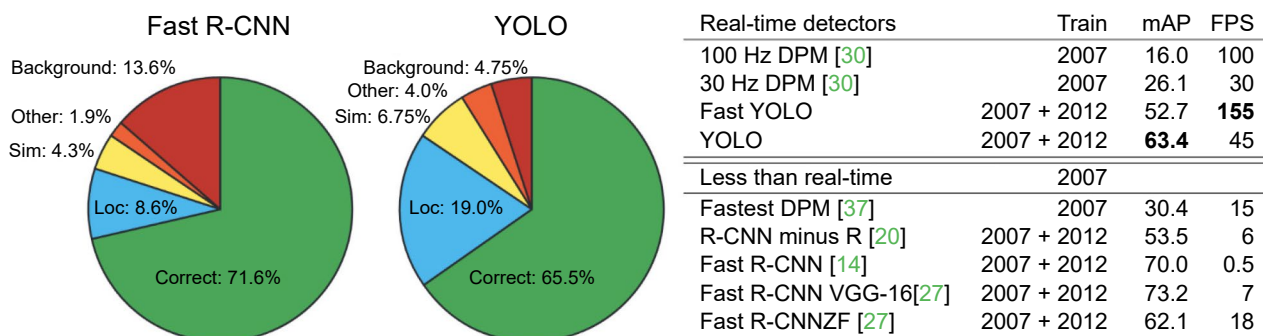


圖 20. 演算法效能比較。

透過圖表可看出，YOLO 相比於 R-CNN 速度明顯快了許多，同時和 Fast R-CNN 相比，準確度 (mAP) 也僅略低一點。根據官方實測在 Titan X GPU 上可以達到 45 FPS，而在我們專題所使用的 Jetson Xavier NX 上約為 3.8 FPS。

#### 4.4 影像辨識小結——針對臺灣交通開發的系統

本次研究中的影像辨識，我們參照了國外自駕車的特點和許多輔助駕駛的功能，作為我們的參考對象。但與國外道路相比，臺灣的交通相對複雜，同時文化的差異也使辨識課題不盡相同，像是紅綠燈在顏色和外型上就與歐美的不同，辨識也常受到臺灣道路上繽紛的廣告刊版影響，外加路標多元且複雜，人車眾多等問題，因此常造成路線被遮蔽、影像辨識效果不佳等問題。

因此，本次研究在影像辨識的領域最精華且特別的部分在於，我們不僅只是將方法實現出來，而是特別針對臺灣的交通做了許多的研究和改變。我們嘗試使用不同的演算法、透過實測，比較不同的方式所造成的成效差異，利用數據分析，開發出專屬於臺灣道路的篩選方式，像是在路線偵測時，透過自行設定的直線方程式作為篩選條件、紅綠燈辨別則使用二次篩檢提升準確率，盡可能地實現我們期望的辨識效能與結果。

除了針對臺灣交通道路客製化之外，我們影像辨識的功能也緊扣在老人較常發生危險事故的原因，同時針對代步車的使用習慣進行調整。例如代步車通常行駛在道路的右側，因此將辨識紅綠燈的範圍略為左移、當系統偵測不到右線時，則去判別左線的斜率是否接近垂直線，作為車體偏移檢測的依據。

另外我們也致力於研究每個演算法背後的原理，讓我們能知道如何更有效的調整參數，以及成效不彰時要如何修正，使我們的系統能更精準的辨識臺灣的交通路況。然而，因臺灣的交通相較國外複雜許多，更是增加了本次研究的困難度，也同時展現了此研究的價值。

## 五、作品成果展現

### 1. 車體照片



圖 21. 車體照片。

### 2. 影像辨識照片

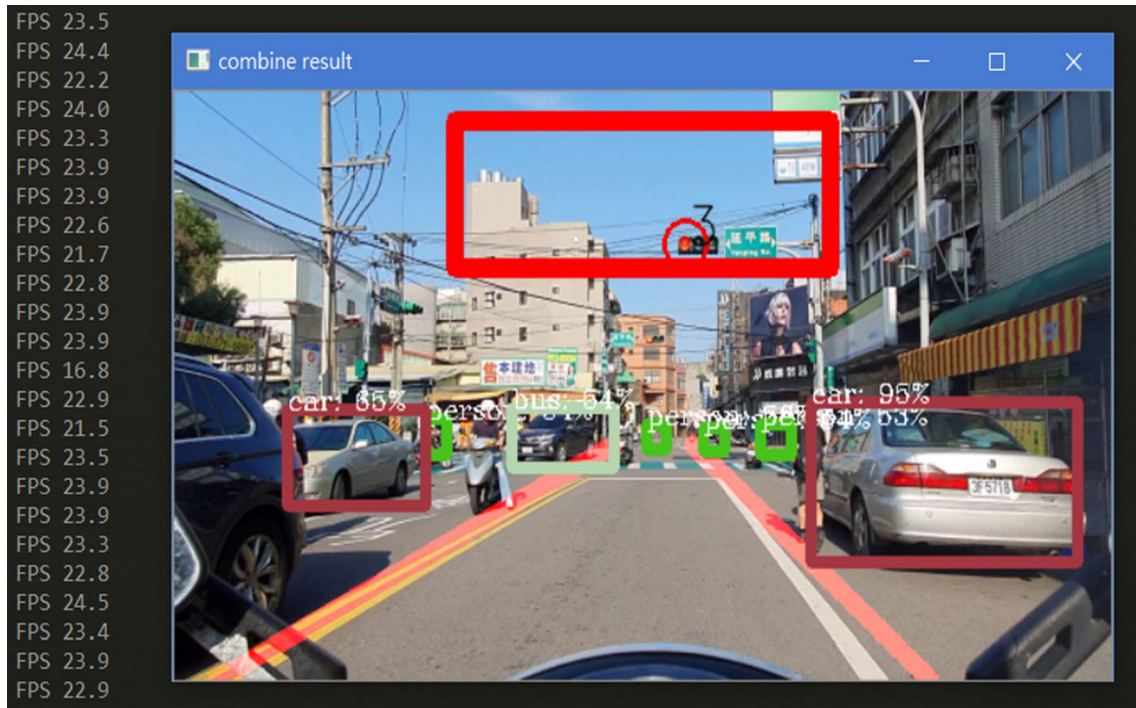


圖 22. 影像辨識。

### 3. 手機軟體成果

表 1. 雙軟體簡介

軟體名稱	AutoBrake Panel	AutoBrake Family
中文暱稱	智能儀表板	家屬專用軟體
使用對象	代步車駕駛	駕駛的家屬
設計目的	使駕駛得知行車資訊、危險警示	幫助家屬掌握長者行車安全
功能	顯示車速、電量、地圖掌握危險警示、震動、語音提醒、緊急聯絡人、家屬貼心 Note	長輩行車資訊、即時位置、行車資訊數據統計、發生事故通知、地點抵達通知、設定儀表板緊急聯絡人和家屬貼心 Note

#### 3.1 AutoBrake Panel ( 智能儀表板 )

我們透過手機 App 當作智能儀表板，手機本身是個非常高階的顯示器，相較於傳統儀表板，我們的儀表板能夠支援多點觸控、GPS、語音播報、遠端軟體更新。因此我們使用這個世代人人皆有的手機，來取代價格高昂的顯示器，讓駕駛能與系統互動、同時節省額外的開發成本，而我們也利用這樣的優勢，設計出能和家屬專用軟體及車體遠端互動的功能。



圖 23. AutoBrake Panel 顯示介面

- 當開啟 App 的時候，便會自動連線到煞車系統的藍芽模組，並顯示車體回傳的開機狀態，同時停用開始鍵 (暗燈)，直到車體完成開機。
- 車體安全鎖：我們的儀表板同時也是車體的馬達安全鎖，按下 Start 即進入一般行駛狀態、解鎖車體馬達，顯示車速、驅動比例、馬達即時溫度等運行資訊；反之在行駛狀態按下 Pause 則進入安全模式，避免誤觸油門。
- 家屬貼心 Note：家屬可以透過 AutoBrake Family 軟體，遠端更改想要顯示在儀表板上的



文字，叮嚀老人需要帶的東西或是代辦事項。

- 緊急聯絡人：在儀表板右下角，我們也製作了緊急連絡人的快捷按鈕，幫助老人可以在緊急的時候直接連絡家屬，而家屬可以透過 AutoBrake Family 軟體，隨時更改緊急聯絡人的電話號碼。
- 危險警示：當發生危險情況時軟體會顯示相對應的警示圖，並且發出震動、語音警示駕駛，同時提醒用路人。
- 掌握路況：此 App 也整合了 Google Map，方便使用者在路況不熟的時候可以透過簡單的左滑，快速的打開地圖掌握路況。



圖 24. 危險警告圖示

### 3.2 AutoBrake Family 家屬專用軟體

為了讓家屬能掌握家中長輩的即時動向與狀態，我們額外開發了一款軟體，並將車體車況、GPS 定位、事故提醒等資料回傳給家屬，以免意外發生時家屬無法掌握長輩實際位置及狀況，並加快救護時間，同時也可以讓家屬透過此軟體幫助老人設定智能儀表板資料，減少長者使用軟體的操作難度。



圖 25. AutoBrake Family 顯示介面

- 即時狀態頁面：顯示車體當前的行駛狀態，如剩餘電量、馬達溫度、目前車速，並在電量過低或是馬達溫度異常時通知家屬。

- 目前位置頁面：顯示老人的即時位置，以及最近一次更新位置的時間，按下標記點則會連接至 Google Map App。
- 圖表統計頁面：我們會將上述的行駛資料統計成圖表，透過這些圖表可以讓家屬了解長者的騎乘習慣，根據電量走勢也可以得知電池的使用狀態和推估長者的生活作息。
- 個人設定頁面：
  1. 設定儀表板的家屬貼心 Note 要顯示的文字。
  2. 設定儀表板的緊急聯絡人電話號碼。
  3. 設定長輩的常去地點。
- 出發抵達通知：家屬可以預先設定常去地點，在長者出發或是抵達常去地點時通知家屬。
- 事故提醒：們透過車體感測是否發生傾倒或碰撞等事故，在第一時間透過此 App 通知家屬長者可能發生事故，以避免錯過搶救時機，導致憾事發生。

## 六、結論

本次研究透過自製電動腳踏車模擬醫療代步車，成功實現了機電整合、影像辨識、輔助煞車機構設計、手機雙軟體開發等功能，透過實測與實驗探討「醫療代步車-AI 輔助煞車系統」的可實現性。

### 1. 機電整合

透過獨立微處理器，將各感測器模組化，並制定危險指標。

1. 獨立微處理器：已實測證實獨立微處理器的優點。
2. 感測器模組化：自製 PCB 板，增加感測器穩定度。
3. 危險指標制定：條件被觸發時，軟、硬體進行相對應作動。

### 2. 影像辨識

利用影像辨識實現：車體偏移偵測、紅綠燈辨別、人車辨識等功能。

1. 車體偏移偵測：易受到路上車體、路面標誌影響。
2. 紅綠燈辨別：有效範圍內偵測號誌，相當準確。
3. 人車辨識：可有效結合感測器偵測危險。

### 3. 輔助煞車機構設計

以結構力學為設計考量，自製前後輔助煞車系統、主控避震系統。

1. 輔助煞車：依照車體結構，以三點以上固定方式使結構更穩固。
2. 避震設計：將四個彈簧並聯，保護主控系統。
3. 車體機構：分析各部件支點位置之分力。

### 4. 手機雙軟體開發

考量家屬與長輩不同的需求開發相關功能。

1. AutoBrake Panel (智能儀表板)：以簡單的功能符合老人的需求。
2. AutoBrake Family (家屬專用軟體)：開發多項功能掌握長輩的行車狀況。

## 參考文獻

1. Association Adilca, BRAKING FORCE, (2020). Please refer to the web site:  
<http://www.adilca.com/files.htm>
2. Twiflex, Industrial Disc Brakes, (2018). Please refer to the web site:  
<https://www.twiflex.com/-/media/Files/Literature/Brand/twiflex-limited/Catalogs/P-1648-TF.ashx?la=en&hash=1A2A38BAF28709DFFE22D0D2B89FBB5570FFC75C>
3. Huco Dynatork , Formulae and Conversion Factors. Please refer to the web site: <https://www.twiflex.com/-/media/Files/Literature/Brand/twiflex-limited/Catalogs/P-1648-TF.ashx?la=en&hash=1A2A38BAF28709DFFE22D0D2B89FBB5570FFC75C>
4. 原口秀昭, 圖解結構力學練習入門, 台北: 臉譜, (2015).
5. 高木任之, 超簡單結構力學, 台北: 世茂, (2014).
6. Robert Keim, How to Choose the Right Microcontroller for Your Application, (2019). Please refer to the web site:  
<https://www.allaboutcircuits.com/technical-articles/how-to-choose-the-right-microcontroller-for-your-application/>
7. Maurizio Di Paolo Emilio, 140-year-old secret in the Hall effect discovered, (2019). Please refer to the web site:  
<https://www.eetimes.com/140-year-old-secret-in-the-hall-effect-discovered/#>
8. Joseph Redmon , Santosh Divvala , Ross Girshick & Ali Farhadi, You Only Look Once: Unified, Real-Time Object Detection, (2016). Please refer to the web site:  
[https://pjreddie.com/media/files/papers/yolo\\_1.pdf](https://pjreddie.com/media/files/papers/yolo_1.pdf),
9. Jędrzej Świączewski, YOLO Algorithm and YOLO Object Detection: An Introduction, (2020). Please refer to the web site:  
<https://appsilon.com/object-detection-yolo-algorithm/>
10. Steven Shen, 關於影像辨識, 所有你應該知道的深度學習模型, (2018). Please refer to the web site:  
<https://medium.com/cuboi/%E7%89%A9%E9%AB%94%E5%81%B5%E6%B8%AC-object-detection-740096ec4540>
11. R. Fisher, S. Perkins, A. Walker and E. Wolfart., “Hough Transform”, (2003 ) Please refer to the web site:  
<http://homepages.inf.ed.ac.uk/rbf/HIPR2/hough.htm>
12. 天道酬勤, 邊緣偵測懶人包-Canny演算法, (2019). Please refer to the web site: <https://medium.com/@bob800530/open-cv%E5%AF%A6%E4%BD%9C%E9%82%8A%E7%B7%A3%E5%81%B5%E6%B8%ACcanny%E6%BC%94%E7%AE%97%E6%B3%95-d6e0b92c0aa3>

## 作者簡介

羅敦允先生畢業於桃園市立內壢高級中等學校，現為東海大學資訊管理學系學生。

Tun-Yun Lo received his diploma from Taoyuan Municipal Nei Li Senior High School. He is currently an undergraduate student in Information Management at Tung Hai University.

李文翔先生畢業於桃園市立內壢高級中等學校，現為國立中正大學資訊工程學系學生。

Wen-Shiang Li received his diploma from Taoyuan Municipal Nei Li Senior High School. He is currently an undergraduate student in Computer Science and Information Engineering at National Cheng Chung University.

陳彥儒先生畢業於桃園市立內壢高級中等學校，現為長庚大學電機工程學系學生。

Yan-Ru Chen received his diploma from Taoyuan Municipal Nei Li Senior High School. He is currently an undergraduate student in Electrical Engineering at Chang Cung University.

賴宥蓁先生畢業於桃園市立內壢高級中等學校，現為國立宜蘭大學資訊工程學系學生。

Yu-Chen Lai received his diploma from Taoyuan Municipal Nei Li Senior High School. He is currently an undergraduate student in Computer Science and Information Engineering at National Ilan University.

陳安儒先生畢業於桃園市立內壢高級中等學校，現為國立海洋大學機械與機電工程學系學生。

An-Ju Chen received his diploma from Neili Junior High School. He is currently an undergraduate student in Mechanical and Mechatronic Engineering at National Taiwan Ocean University.

呂盈臻小姐畢業於桃園市立內壢高級中等學校，現為國立台北科技大學機械工程學系學生。

Ying-Chen Lu received her diploma from Neili Junior High School. She is currently an undergraduate student in Department of Mechanical Engineering at National Taipei University of Technology.

廖經益先生為數學科教師，曾任教務主任，現為桃園市立內壢高中圖書館主任。

Ching-Yi Liao is a math teacher, he has served as director of the Academic Affairs, and is now the director of Taoyuan Municipal Nei-Li High School Library.

曾秀燉為臺灣師範大學化學碩士，現為桃園市立內壢高中化學科教師。

Hsiu-Tun Tsen received her M.S. in Department of Chemistry from National Taiwan Normal University. She is currently a Chemistry teacher at Taoyuan Municipal Nei Li Senior High School.