

2024 年【科學探究競賽-這樣教我就懂】

普高組 成果報告表單

題目名稱：開拓路徑～三寶駕駛挑戰神作 AI

一、摘要

本次探究我們專注於利用人工智慧技術改善輔助駕駛系統，特別是在交通號誌辨識方面的應用。透過機器學習，我們成功提升了系統的智能化和準確性，並建立了交通號誌資料庫以支持系統辨識。然而，我們也意識到機器學習的精準度需要更多資料的支持，因此未來的研究將集中在提高辨識準確性和擴展辨識種類上。同時，我們將注重加強系統的安全性和隱私保護，以確保駕駛者和使用者的安全，我們希望這些努力能夠為交通安全技術的發展和社會的進步做出積極貢獻。

二、探究題目與動機

隨著車輛普及，交通安全日益成為全球焦點。雖然輔助駕駛系統有助於提高駕駛安全，但目前仍存在許多缺陷，例如未能提供交通號誌警示功能。因此，我們研究開發一款能夠辨識交通號誌和交通號誌燈的智能化輔助駕駛系統，及時提醒駕駛人注意交通狀況，並根據號誌指示執行相應操作，從而期望降低交通事故發生率，保障行人和駕駛人的安全。

三、探究目的與假設

我們在日常生活中經常面臨駕車時的種種不便，即使輔助駕駛技術不斷進步，仍然存在許多挑戰。因此，我們的研究旨在透過人工智慧，尤其是利用機器學習技術，來改進輔助駕駛系統。例如，我們致力於開發能夠辨識交通號誌牌和號誌燈的技術，以進一步提高駕駛人的安全和便利。

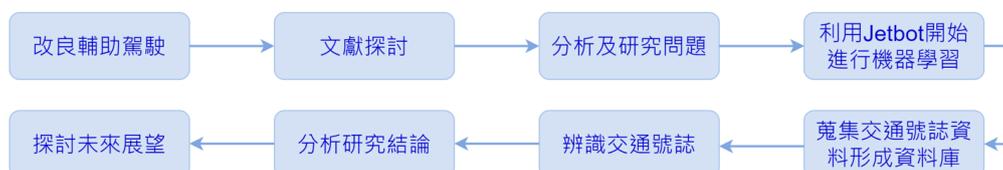
- (一) 學習如何利用機器學習技術
- (二) 蒐集交通號誌資料形成資料庫
- (三) 利用機器學習辨識交通號誌
- (四) 輔助駕駛 SWOT 分析與發展可能性

四、探究方法與驗證步驟

一、研究方法

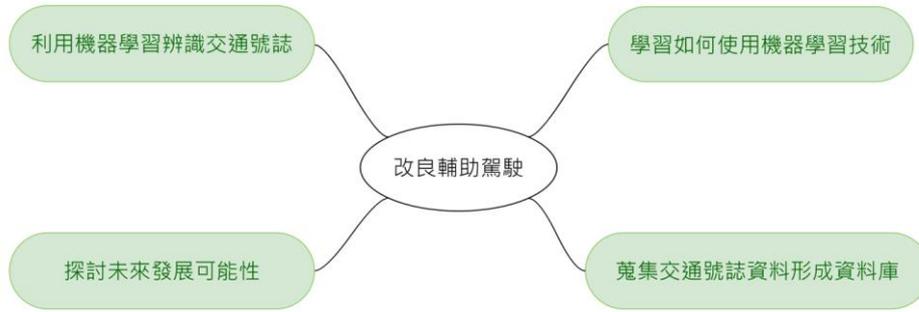
- (一) 文獻探討法：蒐集相關領域資料、論文，深入了解有關人工智慧等的知識。
- (二) 實作分析法：以 Jetbot 自走車作為實驗對象，實際改良輔助駕駛系統，模擬不同路況，實作辨識交通號誌及其對應動作。
- (三) 優劣分析法：根據我們改良過後的輔助駕駛系統，分析其優點及其有待改進的部分。

二、研究流程



圖一：研究流程圖

三、研究架構

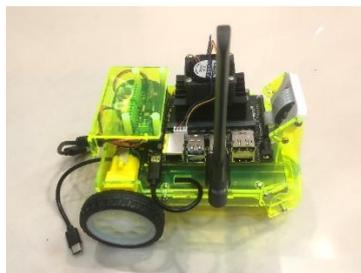


圖二：研究架構圖

四、研究設備

表一：本研究使用硬體設備表

硬體名稱	規格
Jetson Nano	<ul style="list-style-type: none"> ● GPU：NVIDIA Maxwell™ 架構配備 128 個 NVIDIA CUDA® 核心 ● CPU：四核心 ARM® Cortex®-A57 MPCore 處理器 ● 記憶體：4 GB 64-bit LPDDR4 ● 大小：69.6 mm x 45 mm ● 顯示器：HDMI 2.0 或 DP1.2 eDP 1.4 DSI (1x2) 2 個同步
Jetbot	<ul style="list-style-type: none"> ● 鏡頭：800 萬高清像素，160 度廣角鏡頭 ● 馬達規格：TT 減速馬達(齒輪比 1:120) ● 電源：雙孔輸出 5V / 3A 行動電源 ● USB 網卡：支援 2.4GHz 和 5GHz 頻段無線網卡 ● 車子尺寸(L x W x H)mm：約 135 x 125 x 113 mm



圖三：Jetbot 組裝後成品

表二：軟體設備表

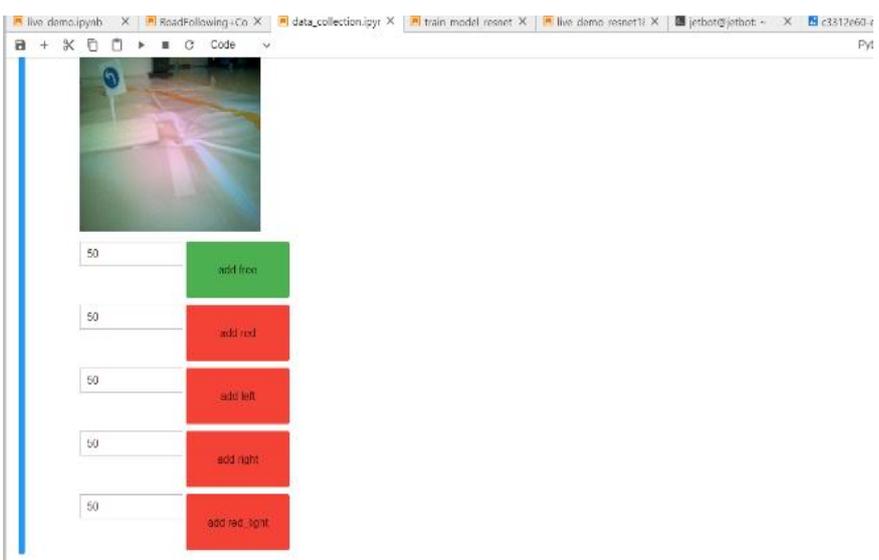
軟體名稱	規格
程式語言	Python 3.10.10 64-bit
開發環境	Jupyter Notebook
Balenaetcher	v1.18.12

五、研究步驟

(一) 蒐集交通號誌資料形成資料庫

我們先模擬交通號誌可能會出現在鏡頭畫面上的各種角度及位置。此外，我們在蒐集資料的過程中，須執行另外一段程式碼以形成資料庫，這段程式碼使我們拍的照片能存入另外一個資料夾，以利後續訓練時使用，如下表三所示。

表三：蒐集交通號誌程式碼及程式碼表格說明

實作程式碼截圖	程式碼說明
	<p>對不同的交通號誌種類擷取畫面，在之後的機器訓練過程中做使用。</p>
<pre data-bbox="193 1126 1072 1514"> import os left_dir = 'dataset/turn_left' red_sign_dir = 'dataset/red_sign' right_dir = 'dataset/turn_right' free_dir = 'dataset/free' red_light_dir = 'dataset/red_light' # we have this "try/except" statement because these next functions can throw an error if the directories exist already try: os.makedirs(free_dir) os.makedirs(blocked_dir) except FileExistsError: print('Directories not created because they already exist') </pre>	<p>根據可以行進和不能行進分成 “free” 和 “block”，並進一步在兩種型態內分成不同種類的交通號誌</p>

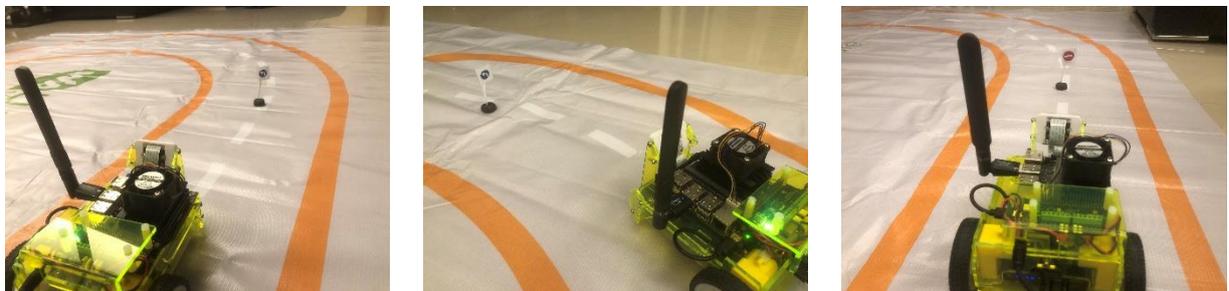
(二) 利用機器學習技術辨識交通號誌

將上述蒐集的資料標記後，將交通號誌資料傳入並利用機器學習技術重複訓練 30 次，使 Jetbot 能正確辨識交通號誌。Jetbot 根據辨識出的結果做出相對應的反應，如下表四所示。

表四：訓練辨識交通號誌程式碼及程式碼表格說明

實作程式碼截圖	程式碼說明
<pre data-bbox="193 1816 922 1899"> NUM_EPOCHS = 30 BEST_MODEL_PATH = 'best_model_test1.pth' best_accuracy = 0.0 </pre>	<p>宣告訓練次數為 30 次、訓練後模型保存途徑、以及準確率</p>
	<p>每個訓練 epoch 中，從訓練數據集中取出批次的圖像和標籤。通過模型進行前向傳播以獲得預測輸</p>

<pre> outputs = model(images) loss = F.cross_entropy(outputs, labels) loss.backward() optimizer.step() </pre>	<p>出。計算模型輸出和實際標籤之間的損失。使用反向傳播計算梯度並更新模型參數。</p>
<pre> test_error_count = 0.0 for images, labels in iter(test_loader): images = images.to(device) labels = labels.to(device) outputs = model(images) test_error_count += float(torch.sum(torch.abs(labels - outputs.argmax(1)))) test_accuracy = 1.0 - float(test_error_count) / float(len(test_dataset)) print('%d: %f' % (epoch, test_accuracy)) </pre>	<p>在每個訓練 epoch 結束後，使用測試數據集進行模型的評估。通過模型進行預測，並計算模型的測試準確率。如果測試準確率優於先前的最佳準確率，則保存當前模型為最佳模型。</p>
<pre> if test_accuracy > best_accuracy: torch.save(model.state_dict(), BEST_MODEL_PATH) best_accuracy = test_accuracy </pre>	<p>如果測試準確率大於先前記錄的最佳準確率（best_accuracy），則保存當前模型的參數為最佳模型。更新 best_accuracy 為當前的測試準確率。</p>
<pre> def update(change): global blocked_slider, robot x = change['new'] x = preprocess(x) y = model(x) </pre>	<p>從 change 中獲取新的圖像</p>
<pre> y = F.softmax(y, dim=1) prob_blocked = float(y.flatten()[0]) blocked_slider.value = prob_blocked if prob_blocked < 0.5: robot.forward(speed_slider.value) else: robot.left(speed_slider.value) time.sleep(0.001) update({'new': camera.value}) </pre>	<p>prob_blocked 為偵測交通號誌的偵測率。若超過預設的則判定為前方有障礙物，使車子左轉</p>



圖四：Jetbot 辨識交通號誌實作

六、SWOT 分析

表五：輔助駕駛之 SWOT 分析

<p style="text-align: center;">Strengths</p> <p>提醒駕駛人目前路況以及周遭交通號誌，提升駕車時的行車安全，也能減少因忽略交通號誌而造成的交通事故。此外，輔助駕駛也能提高駕駛舒適性以及降低駕駛的疲勞感。</p>	<p style="text-align: center;">Opportunities</p> <p>為提高輔助駕駛對駕駛人的影響性，可針對所辨識出的交通號誌做出相對應的反應，例如：若辨識出紅燈，則協助車子減速，同時也能減少因為闖紅燈而發生的交通事故。</p>
<p style="text-align: center;">Weaknesses</p> <p>全世界汽車的輔助駕駛系統僅對自動駕駛的程度進行程度上的規範諸如 Level 1~Level 6 車輛自動化。對於駕駛系統的標準化尚未完成，因此駕駛人駕駛不同車型需先了解不同輔助駕駛系統，否則反而會造成事故。</p>	<p style="text-align: center;">Threats</p> <p>輔助駕駛系統確實可以減輕駕駛的負擔，但可能造成駕駛人過度依賴輔助駕駛系統，而將輔助駕駛誤認為自動駕駛，進而釀成大禍。</p>

五、結論與生活應用

一、結論

- (一) 成功辨識交通號誌並讓 Jetbot 做出相對應的動作
- (二) 發現機器學習需要更多資料才能提高辨識的精準度
- (三) 未來目標是提高辨識準確性並增加辨識的號誌種類
- (四) 未來計劃將研究應用於真實車輛上，提升駕駛安全

二、未來展望與應用

- (一) 輔助駕駛之未來發展方向

表六：輔助駕駛之未來發展方向及說明

發展方向	說明
自動駕駛技術的提升	更先進的感知技術，例如：雷達、攝影機、激光雷達等的使用，提高對周圍環境的感知能力。此外，強化車輛的機器學習和人工智慧能力，使其能夠更容易理解複雜的交通狀況和不同的駕駛風格。
通訊技術的整合	促進車輛之間的即時通訊，提高交通流的效率，減少交通事故。將車輛和交通基礎設施（如交通號誌燈、路標）連接起來，實現更智慧的交通管理。
能源效能和環保	透過智慧導航系統，考慮交通情況、氣象等因素，使行車路線更節能和高效。同時，推動共享的概念，除了能降低車輛擁有率，同時也能減少交通壅塞並且達到節能減碳的效果。
法規和保險制度的發展	配合自動駕駛技術的發展，制定相對應的法規和標準，確保其合法、安全的運行。根據自動駕駛的風險和責任分配，調整相

	應的汽車保險模型。
城市規劃的改進	將自動駕駛技術納入城市交通規劃，提高城市交通系統的效率。促進公共交通與自動駕駛技術的結合，實現更便利和經濟的出行方式。

(二) 輔助駕駛之生活應用

表七：輔助駕駛之生活應用及說明

生活應用	說明
自動跟車系統	利用車子外圍的攝影機偵測前方車子的車距、車速等等，並自行調整至安全速度的範圍，保持和前方車子的車距。
自主停車	運用感應器和控制系統的協助，辨識出適合停車的空間。讓駕駛即使在下窄的空間也能停車
盲區提醒	在大型卡車或貨車中常會有視覺死角。運用雷達或鏡頭偵測盲區，在車子或行人出現時發出警告。
哨兵模式	哨兵模式可偵測在駕駛離開期間，若車子周圍出現可疑人士或外界物品的碰撞時，啟動哨兵模式，可以用周圍的攝影機記錄當時的狀況。
疲勞駕駛 危險警示	在長期的路途中，透過偵測駕駛的行車行為，若有異常的行車模式，會警示駕駛以免發生意外。

參考資料

1. 劉湘瑤、張震興、張璣勻、趙恩、李思賢 (2020)。人工智能倫理的挑戰與反思：文獻分析。資訊社會研究/Journal of Cyber Culture and Information Society，27-64。
2. 鄭弘毅 (2021)。深度學習與機器學習於應用程式識別之比較。銘傳大學：碩士論文
3. 曾品嘉 (2022)。AI 深度學習影像辨識之應用。樹德科技大學：碩士論文
4. 永豐金證投顧 (2022年3月21日)。自動駕駛是什麼？ADAS 扮演關鍵角色；車用電子概念股一次看。<https://www.sinotrade.com.tw/richclub/industry/-64114f50476b232e18c54c5e>