運輸計劃季刊
第四十九卷 第三期
民國一〇九年九月
頁 235～頁 258

Transportation Planning Journal
Vo1. 49 No. 3
September 2020
PP. 235～258

基於空拍影像之人車軌跡抽取技術[1]

PEDESTRIAN AND VEHICLE TRAJECTORY EXTRACTION BASED ON AERIAL IMAGES

蘇志文 Chih-Wen Su[2]
黃家耀 Walter Ka Io Wong[3]
張開國 Kai-Kuo Chang[4]
葉祖宏 Tsu-Hurng Yeh[5]
孔垂昌 Chui-Chang Kung[6]
黃明正 Ming-Cheng Huang[7]
溫基信 Chi-Sin Wen[8]

(108 年 12 月 20 日收稿，109 年 09 月 16 日第一次修改，109 年 09 月 30 日定稿)

摘 要

車輛與行人軌跡在許多交通分析應用上是相當重要的參考資訊。本研究針對繁雜的路口交通環境，透過深度學習自動偵測空拍影像中的人車位置，進而取出大量豐富的人車軌跡資訊。本研究的主要貢獻有三：(1) 結合空拍機與深度學習、影像處理等多項跨領域技術，以實際的十字路口為測試對象，完成了一個高度自動化的人車軌跡抽取技術。(2) 利用 Mask R-CNN

1. 本文為路口無人機交通攝影及衝突分析技術開發 (計畫編號:MOTC-IOT-108-SEB006) 部分研究成果，特此致謝。
2. 中原大學資訊工程系專任助理教授(聯絡地址：桃園市中壢區中正路 200 號；電話：03-2654730；E-mail：lucas@cycu.edu.tw)。
3. 國立交通大學運輸與物流管理學系專任副教授。
4. 交通部運輸研究所運輸安全組組長。
5. 交通部運輸研究所運輸安全組副組長。
6. 交通部運輸研究所運輸安全組研究員。
7. 交通部運輸研究所運輸安全組研究員。
8. 訊力科技股份有限公司副總經理 (研究案計畫主持人)。

與 YOLOv3 等物件偵測方法，解決偵測框貼合車輛形狀以及行人等小物件偵測問題；(3) 將傳統軌跡資訊從線拓展為面的行經區域資訊，進一步豐富軌跡資訊。未來透過自動提取出 2D 平面上的軌跡資訊，可提供交通分析上的有利參考。

關鍵詞： 空拍影像、深度學習、卷積神經網路、物件偵測、物件追蹤

ABSTRACT

Vehicle and pedestrian trajectories are important references for many traffic analysis applications. In this study, we focus on the complex traffic environment at intersections and use deep learning to automatically locate the positions of vehicles and pedestrians in aerial images, and then extract a large amount of rich information on vehicle and pedestrian trajectories. There are three main contributions in this study: (1) Combining aerial camera with several cross-domain technologies such as deep learning and image processing, a highly automated human/vehicle trajectory extraction technology is accomplished using real intersections as test data. (2) Solve the problem of fitting bounding boxes to vehicles and detecting small objects such as pedestrians by using Mask R-CNN and YOLOv3, respectively; (3) Expanding traditional trajectory information from lines to travel area information to further enrich trajectory information. By automatically extracting the trajectory information on 2D maps, it helps to enrich the information needed for advanced traffic analysis.

Key Words: *Aerial image; deep learning; convolutional neural network; object detection; object tracking*

一、前　言

隨著都市人口的愈加密集化，重要道路於交通尖峰時段所需承載的運輸量大幅提升，交通事故發生之數量、機率、後果亦有巨量化、多元化、及嚴重化趨勢，交通事故已成為交通運輸過程中難以預測之意外事件，為有效防制道路交通事故發生，降低交通事故傷亡人數與嚴重程度，各國政府無不致力於交通安全防制的課題。依據交通部 105 年頒訂「機車交通政策白皮書」中統計 96 年至 100 年交通事故資料顯示，事故地點為交岔路口比例最高 (高達 60.5%) [1] ；美國研究分析顯示都會區與郊區交通事故中，分別超過 1/2 與 1/3 發生集中於交岔路口範圍內，在澳洲則分別為 43%及 11% [2]。

本研究主要以都會區交通環境為背景，進行十字路口影像分析與調查，鑒於一般路口監視攝影機之拍攝角度易造成不同車輛大小、方向、形狀之差異，且易有車輛影像遮蔽之問題，難以將路口監視影像直接應用到車流影像分析之領域。因此透過無人機蒐集交岔路

口之人車流動空拍影像，使用深度學習並結合影像分析技術進行人車偵測、分類及追蹤，找出人車所在之範圍以取得人車之行經軌跡。本研究相關技術除複雜之十字路口外，亦能普遍適用於更為單純之一般道路情況。

二、文獻回顧

2.1 基於卷積神經網路 (Convolutional Neural Network) 之物件偵測技術

一般而言，多數基於卷積神經網路的物件偵測方法可以分為兩類。一類是以 Faster R-CNN [3] 為首的兩階段 (two-stage) 物件偵測方法，先透過訓練提出大量潛在的物體外接框 (bounding box)，再將框內的特徵圖利用卷積神經網路進行調整框的位移、分類並保留信心度高的結果。2017 年由 He 等人提出的 Mask R-CNN [4] 是這類方法中的另一著名架構，除了延續 Faster R-CNN 的基本架構外，更透過 FCN (Fully Convolutional Network) 卷積層 (如圖 1 中兩個卷積層) 對外接框中的每個像素分類為物件與背景，進而可切割出物體輪廓區域，同時達成物件偵測、分類、分割三大工作。然而，由於兩階段方法中的同一區域可能經過多次重複處理，因此計算量通常明顯較大導致處理速度較慢。

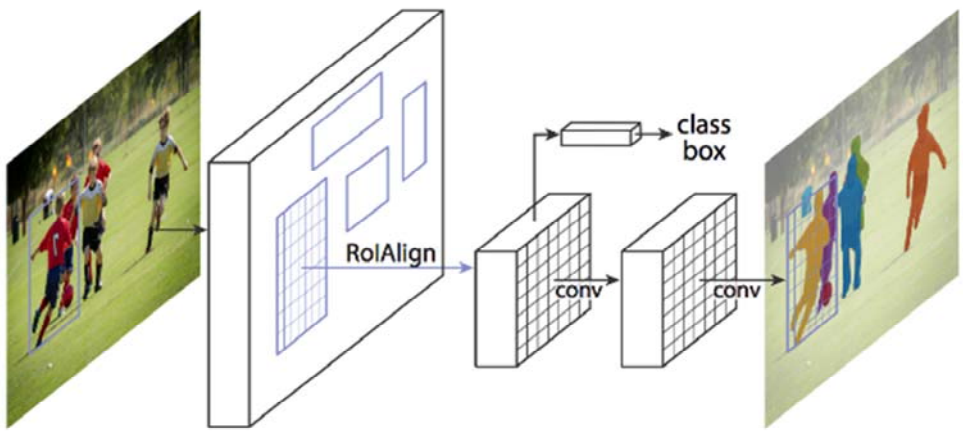


圖 1　Mask R-CNN 的物件分割概念 [4]

相對於兩階段物件偵測方法，另一類方法是將影像分區並迴歸計算出各區的最佳代表物件位置與分類。這類單一階段 (one-stage) 物件偵測方法以 Redmon 等人 [5-7] 從 2015 年開始提出的一系列 YOLO (You Only Look Once) 架構最為著名。相對於先預測大量的物件候選區域，再針對每個候選區域去檢測物件分類的兩階段物件偵測方法，YOLO 系列主要概念是將影像分割成 N×N 個區塊 (grid) 後，考量物件中心落在各區塊內時的外接框資訊

—包含外接框中心對每個區塊左上角的偏移量 (x, y) 與及外接框的寬高 (w, h) —與屬於各分類的信心度，並直接透過神經網路訓練學習出前述外接框資訊，最後排除掉信心度低、不屬於各分類的外接框作為偵測結果。YOLO 由於預設了單一物件落在各區塊的情形，因此可以針對這些均勻分布且為數不多的區塊進行更有效率地處理。

Redmon 等人 [7] 在 2018 年提出了 YOLOv2 的改進版本 YOLOv3。不同於 YOLOv2 的特徵圖只有 13×13 一種尺度，YOLOv3 使用了 FPN (Feature Pyramid Network) [8]多尺度預測架構，增加了 26×26、52×52，共 3 種尺度的特徵圖，每格用 3 個錨框 (anchor box) 進行外接框的預測。由於 52×52 等高解析尺度的特徵層，雖然在尺度上較適合偵測大量的小物件，但也因為經過較少的卷積處理而導致特徵資訊不足，難以正確區分小物件類別。因此需要透過 FPN 將具備豐富資訊的低解析尺度特徵圖，放大並加成至高解析尺度的特徵圖中，以彌補其短缺的複雜特徵資訊。因外接框在數量上比 YOLOv2 多了 4.2 倍，加上 FPN 可幫助在 52×52 尺度層取得更豐富的特徵資訊，因此提升了對小物件的偵測效果。另一方面，作者也對 YOLO 原先 19 層的 Darknet 架構做了改進，提出了新一代 53 層的 Darknet 網路架構以取得更有效的影像特徵，並做為 YOLOv3 架構的骨幹 (backbone)。

由於 YOLOv3 在小物件偵測上有不錯表現，且其架構在改寫上亦較有彈性，較能針對特定的小物件對象來優化架構，因此本研究採用 YOLOv3 來偵測空拍影片中的小物件，如行人、自行車與機車等。後續再透過追蹤技術，針對已偵測出之對象擷取出移動軌跡。

2.2 物件追蹤技術

物體追蹤是影像處理上長久以來的課題之一，其原本目的在於早期物體偵測的速度非常緩慢，因此希望藉由較快速的物體追蹤技術來取代後續的偵測動作，提升系統整體的效率。然而，隨著硬體效能與軟體演算法的改進，如今的物體偵測技術已能兼具準確與速度，因此目前的追蹤技術目的也從取代偵測來進行加速，演變為確認偵測結果是否失誤並校正物體位置。按照追蹤所使用的策略不同，大致可區分為下列三類追蹤概念：

1. 建立外觀、運動模型

此類方法主要針對目標物體的外觀一致性或運動趨勢建立模型，在影像中尋找最可能的位置，例如早期追蹤方法常使用的卡爾曼濾波 (Kalman filter) [9] 可利用假定的運動模式來預測後續位置，例如投手投出球的拋物線軌跡由於服從重力的影響，因此相當適合卡爾曼濾波來進行預測。卡爾曼濾波器由一組數學式所組成，在 1960 年由 Kalman 提出，之後便有許多人對它進行廣泛地研究並將它應用在自動導航的領域上。卡爾曼濾波器可以精準並有效率的估算一個系統過去、現在、甚至未來的狀態，即使在本研究無法明確觀察量測到整個系統的所有狀態參數時，它也一樣有效。卡爾曼濾波器系統可大致分為兩階段，如圖 2 所示，一為預測下一個時間點的系統狀態，稱為預測機制 (time update)，二為依實際

測量得到的資訊以對卡爾曼濾波器的各項參數做修正調整，稱為修正機制 (measurement update)。

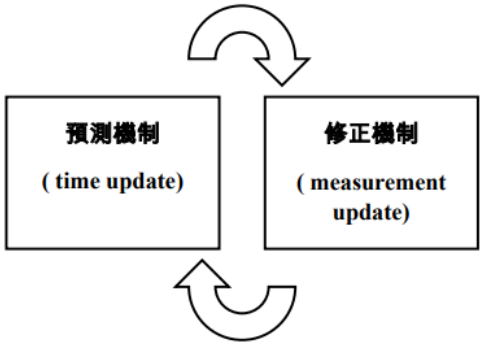


圖 2　卡爾曼濾波器遞迴示意圖

又例如粒子濾波器 (particle filter) [10] 透過預測與更新機率密度函數，來預測非線性的下一次物體位置。近年來，如 KCF (Kernelized Correlation Filters) [11] 追蹤法等利用連續追蹤結果進行即時訓練，並找尋目標位置的方法亦開始興盛，但粒子濾波器與 KCF 追蹤法這些方法主要著重在難度較高的非剛體追蹤問題上，繁複的運算使其較適合少量、甚至是單一物體的追蹤問題上，因此並不適合在利用空拍影像追蹤大量目標且少有形變問題發生的車輛偵測問題上。

2. 計算局部運動資訊

此系列方法係利用影像上的局部特徵，透過局部運動連接物體位置，相對於前述針對物體本身特性進行追蹤的策略，這類方法則是蒐集廣布於影像上的局部特徵與運動向量 (Motion Vector)，針對前一張影像的局部特徵找尋下一張影像中鄰近位置的相似局部特徵並加以匹配，估算出相似特徵在前後影像的位移作為追蹤結果。例如對影像中的角點 (corner) 進行偵測追蹤，或是透過光流法 (optical flow) [12] 取出密集的局部運動資訊，透過這些局部運動資訊並配合前張影像之偵測／追蹤結果，可以利用前張影像車輛區域內之特徵點推估出其後續的位移情形。局部特徵追蹤演算法可分為以下幾個步驟：

(1) 讀取相鄰的兩個影像，至少包含一移動目標 (Moving Object)。

(2) 檢測第一影像中的特徵點，尋找影像中具有大特徵的角點。

(3) 計算第二影像中的特徵點。計算各特徵點在相鄰影像間形成的光流，找出屬於運動目標的特徵點，

(4) 用以估計第一影像特徵點在第二影像中的位置，同時過濾位置不變的特徵點。

由於空拍影像中的物體基本上不會因攝影機角度變化而產生形變，因此本研究過去曾將局部特徵追蹤法應用於空拍影像中的四輪車輛追蹤問題，在效率上也遠比前一類方法更

有效率。然而此類方法並不適用於複雜環境下的小物體追蹤，主要原因在於小物體所包含的特徵點勢必也較少，特別在路面斑馬線等對比強烈的背景下，容易被背景中大量而強烈之特徵點所埋沒，只有當物體具有足夠面積時，才能透過較多的特徵點計算出較穩定一致的運動軌跡。

3. 基於偵測結果進行追蹤 (Tracking by Detection)

近年來由於物件偵測技術的日趨成熟，許多基於偵測結果的追蹤技術 (tracking by detection) 應運而生，例如 2016 年由 Bewley 等人所提出之 SORT (Simple Online and Realtime Tracking) 演算法 [13] 即為其中經典方法之一。此方法透過卡爾曼濾波器將追蹤中之多個目標對象，透過線性速度的假設來預測後續位置，並與物件偵測結果進行匹配。在多目標間的匹配問題上，則採用了俗稱匈牙利演算法 [14] 的匹配策略，以偵測目標 D 與追蹤預測目標 T 間的外接框重疊程度 (IoU, Intersection over Union) 為基準，計算整體上最低的匹配損失，若追蹤預測目標 T 超過一定幀數尋找不到適合匹配 (預設為 IOU > 0.3) 之偵測目標對象，則停止其後續追蹤。另一方面，由於偵測結果並不是百分之百可靠，因此對於無法匹配任何預測目標之新偵測結果，在對其建立新的追蹤目標後，需連續成功追蹤 (找到適合匹配之偵測結果) 數次，才會將其視為可靠的新出現物體。

SORT 演算法特點在於計算快速的同時，也能達到多物體追蹤的良好表現，而缺點則主要發生於遮蔽嚴重時，帶來的追蹤對象編號改變問題。由於這一類基於偵測結果的追蹤技術主要考量的是物體位置的關聯性，而非物體外觀上的相似性，因此適用在畫面間的物體位移較小的情況，這點一般可透過提高影片幀率來達成。2017 年，一個基於 SORT 的新追蹤方法 Deep SORT 被提出 [15]，以改善前述問題。Deep SORT 主要特點在於加入了深度學習，對行人外觀資訊進行訓練，藉此改善偵測與追蹤對象在遮蔽發生時的匹配效果。

三、研究方法

在複雜的都會區交通環境下，一般槍型監視器畫面中的車輛往往由於所在位置不同，進而造成影片中車輛的大小、形狀都不相同，加上攝影機視角的差異與監控影像品質之問題，造成傳統視訊監控技術難以直接應用到不同攝影機所拍攝的影像內容，往往需要事前對系統參數進行人工調整，才能應用不同攝影機所拍攝之影像。

近年來崛起的深度學習風潮，雖然在電腦視覺領域上有突破性的發展，但對於相對小型物體的辨識率仍有先天上的限制，在遠距離車輛偵測上較容易產生誤判，不如單純車輛分類表現優異。爰此，本研究利用無人機收集空拍影片，再透過深度學習對可能的車輛做進一步的驗證與分類。其中，空拍影像資料蒐集與影像處理技術兩大部分為本研究之基礎，主要目的在於從空拍影片中自動擷取並記錄車輛軌跡資訊。

在本章節中，將詳述如何透過深度學習與部分傳統影像處理，自原始空拍影像中擷取

每部車輛之軌跡，包含車種、進出路口時間與路口方向、單張畫面中的車輛位置等資訊，具體流程如圖 3 所述。首先，我們會從空拍影片中挑選適當的訓練影像，進行人工標註後再透過深度學習技術，訓練出四輪車輛以及行人、兩輪車輛之偵測模型。另一方面，我們也對空拍影片的晃動進行校正，以確保所有車輛在畫面上的位移是來自其自身移動，而非無人機飛行造成的背景晃動。本研究中我們採用了 Mask R-CNN 與 YOLOv3 兩種架構，分別處理面積較大且呈明確矩形的四輪車輛以及面積明顯較小的行人、自行車與機車。在取得單張下的初步偵測結果後，再透過特徵點追蹤與 SORT 追蹤兩種不同的追蹤機制，依據目標特性進行時間上的車輛追蹤，並對未偵測出之車輛進行預測，以擷取完整之車輛軌跡並記錄歸檔。

圖 3　本研究方法流程圖

3.1 空拍影像資料蒐集

據本研究的調查空拍結果 (如表 1 所示)，以一般雙向三線道的交岔路口為例，飛行高度 50 公尺拍攝縱向車道停止線後的範圍容易過小，不利追蹤車輛進出交岔路口情形；飛行高度 100 公尺則縱向車道停止線後的範圍過大，間接降低空拍影像中交岔路口區域範圍大小，不利車輛偵測與辨識；而飛行高度 75 公尺可涵蓋交岔路口外之車輛停止線後約兩部車輛長度的範圍，使影像中的車輛在進出交岔路口時能被完整追蹤，並保有較大的車輛偵測面積，為適合進行交岔路口車流分析之高度。

本研究選定臺灣北、中、南多處地點，蒐集多部 1080p/4K 高解析空拍影片，作為系統開發與訓練測試所需之影像資料庫。為了盡量避免攝影機不同設定對影片的影響，除了讓無人機定點飛行於 75 公尺高度進行垂直拍攝外，攝影機也都安裝於無人機的 3D 雲台以初步減輕影片晃動程度，同時關閉攝影機的曝光、白平衡、對焦等自動調節功能。拍攝畫面中之主要道路建議盡量以水平方式呈現於畫面中，有助於提供更一致的車輛形狀，降低車輛偵測的難度。

表 1　飛行高度拍攝範圍比較

拍攝高度				
	50m			
	75m			
	100m			

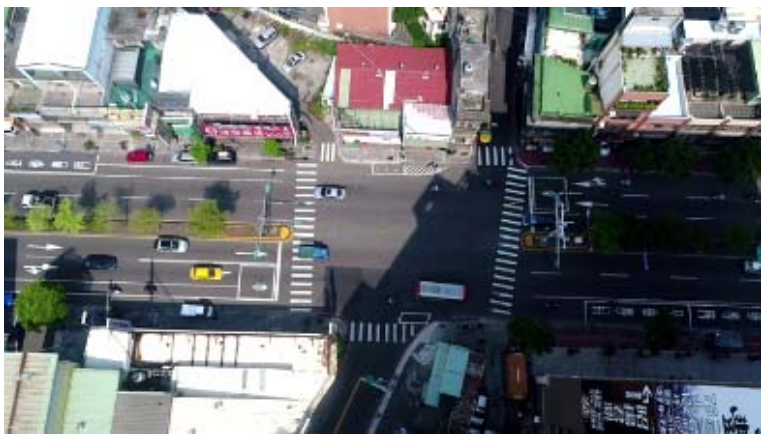
3.2 影片穩定校正與前處理

在利用空拍影像展開行人車輛偵測之前，由於無人機本身易受到陣風影響，導致影像中的街道位置偏離，因此須先對影片中每一張影像進行對齊校準，以確保所有影像中的道路、車輛物件皆維持在同一個平面座標系統。本研究採用 2008 年由 Evangelidis 與 Psarakis 提出的 ECC (Enhanced Correlation Coefficient) 影像對齊技術 [16]，讓輸入影片中的所有畫面都能重新經過旋轉縮放，投影到與人工選定的樣板畫面一致的平面座標系統，如圖 4 所示。

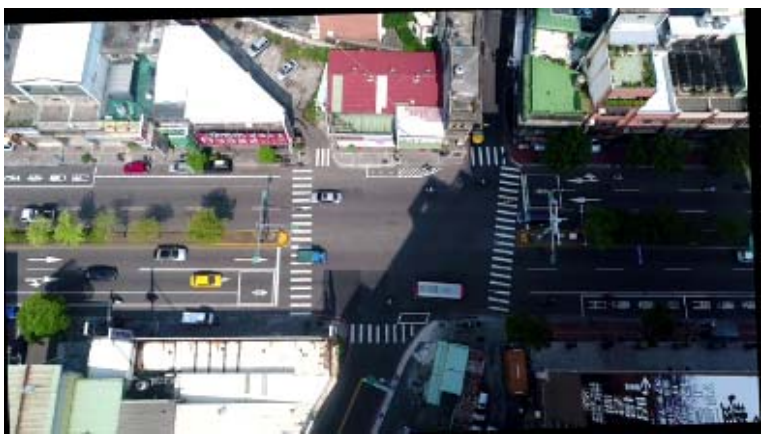
傳統上的影像校正多透過局部特徵點的匹配關係，找尋出共通的轉換矩陣來實現校正。然而，空拍影像中的建築物與道路景深不同，在影像晃動時，特徵點會因景深造成不同程度的位移，加上特徵點容易集中在複雜建築物區域，因此難以估算出共通的轉換矩陣。相對地，ECC 影像對齊技術是找尋一個最佳的 3×3 轉換矩陣，以確保經矩陣轉換後的影像能與樣板影像間，在整體像素亮度上有著最大的相關係數，大幅減低了前述的景深問題。

考量後續透過深度學習偵測車輛需耗費相當大的電腦計算與記憶體空間，為了有效減輕計算負擔，所有對齊校正完之 HD 影片 (解析度 1920×1080) 會以路口為中心裁切為 1024×1024 之影片。若為 4K 影片 (解析度 3840×2160)，則將路口為中心之 2048×2048 範圍，縮小為 1024×1024 之影片，如圖 5 所示。此一 1024×1024 之尺寸大致是 Mask R-CNN

(a) 原首張空拍影像



(b) 原第 5 分鐘空拍影像



(c) 經對齊校正後之第 5 分鐘空拍影像

圖 4　空拍影像對齊示意圖

圖 5　前處理完之影片範圍示意圖

架構在頂級消費型顯示卡 (如 GeForce GTX 1080Ti 與 GeForce RTX 2080Ti) 上能執行的最大影像尺寸。

為了進一步加速處理效率，本研究也對影片畫面進行等間隔採樣，以減少非必要的多餘計算。由於國內一般高解析影片為每秒 29.975 幀，所有影片會以每 3 張畫面間隔進行取樣，得到每秒幀數為 9.99 張的影片。

3.3 四輪車輛偵測與追蹤

在車輛偵測方面，本研究採用 2017 年由 He 等人所提出的深度學習架構 Mask R-CNN，以偵測車輛的矩形範圍。由於一般基於深度學習的物件偵測方法是偵測出物件最上、下、左、右邊緣所成之矩形框，如圖 6 (a) 的 Faster R-CNN 偵測結果所示，然而，這樣的偵測結果無法正確表示轉彎中車輛的傾斜矩型範圍，因此需要能提供更精確車輛外型的車輛偵測架構。相較於 Faster R-CNN 的外接框區域，Mask R-CNN 架構除了外接框外，更進一步提供了像素級的物件分割區域。本研究在此利用了可傾斜之矩形去擬合每個車輛的像素集合，進而成功取出可貼合轉彎中車輛外緣的矩形範圍，這樣的做法改善了車輛位置在表示上的精確度，有助於更準確地估測車輛間距等資訊。另一方面，由於 Mask R-CNN 在偵測出車輛位置時，也同時預測了其所屬的車輛種類，因此可同步完成偵測與分類工作，有效簡化整體處理流程。圖 6 (b) 中，每部車之外接框以虛線表示、每個顏色分割區域代表一部四輪車輛所占區域，從路口中心的轉彎中車輛，可以明顯發現其分割結果非常貼近實際車輛區域。

透過 Mask R-CNN 提供的像素級 (pixel-level) 物件區域範圍，再利用可傾斜之矩形對其進行擬合，進而成功取出可貼合轉彎中車輛外緣的矩形範圍，這樣的做法優於過往以單

一車輛中心代表車輛位置的做法，提供了每部車輛在道路上所佔的實際區域，將原本以線呈現的車輛中心軌跡資訊，提升至以面呈現的車輛行經區域資訊。

(a) (b)

圖 6 (a) 利用 Faster R-CNN 與 (b) Mask R-CNN 進行車輛偵測與分類

在四輪車輛的軌跡方面，本研究是針對偵測出的車輛區域，利用光流法對車輛上之特徵點進行追蹤，再透過同車多個特徵點的位移向量，估算出轉換矩陣與及其在下一張畫面的預測框。之後，再將預測框與 Mask R-CNN 在下一畫面的所有偵測框比對，若預測框與某個偵測框具有高度重疊，則表示極可能同一部車輛在下一畫面的預測與偵測結果相符。由於透過 Mask R-CNN 獲得的偵測框，往往較特徵點計算出的預測框更為精準可靠，因此我們將此偵測框視為是該部車在下一畫面的實際位置。相反地，若下一畫面的所有偵測框都無法與預測框高度重疊，代表下一畫面可能該車偵測失敗或離開了畫面，則暫且將預測框當作該車在下一畫面的實際位置；若之後連續數張畫面之預測框皆無法與偵測框高度重疊，則代表該車很可能之前就離開了畫面，應把前述連續的預測框視為無效。圖 7 為四輪車輛追蹤情形，紅色框為前一張車輛位置、青色框為當前畫面車輛位置。

由於同一部車輛在不同畫面可能被誤分類為不同車種，因此會再透過投票機制，統計出每條車輛軌跡出現最多次數的車種作為代表。例如若被分類為小客車之次數最多，則該部車輛在所有畫面中的分類結果統一為小客車。此外，由於一般卷積神經網路架構之物件偵測方法對小面積物體表現普遍不佳，因此對於面積甚小的行人與自行車、機車等兩輪小型車輛，本研究採 YOLOv3 架構進行偵測，並對偵測結果進行追蹤以產生軌跡。

3.4 行人與兩輪車輛偵測與追蹤

由於深度學習的發展歷史尚淺，過去 CNN 物件偵測方法主要著重在對所有尺度物體的整體表現，直到 2018 年起小物件偵測才逐步獲得了重視。本研究採用 YOLOv3 作為小

物件偵測的基本架構，主要原因有三：

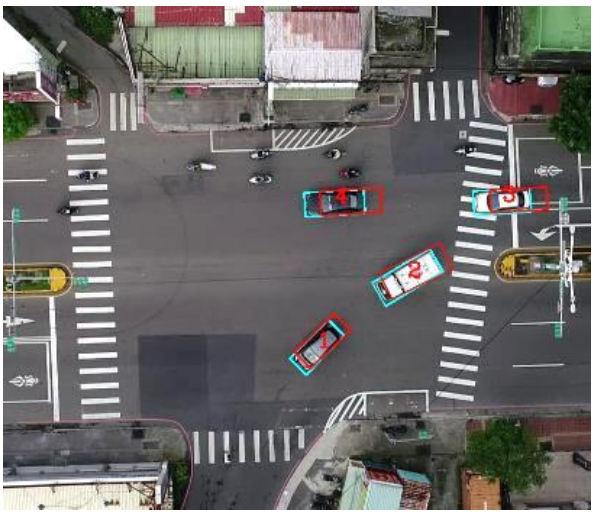


圖 7　四輪車輛追蹤情形

1. YOLOv3 在小物件偵測上有著較佳表現。
2. YOLOv3 程式在架構更改上遠較 Faster R-CNN、Mask R-CNN 等容易，方便針對實際需求加以改進。
3. YOLOv3 在執行效率上也遠較 Faster R-CNN、Mask R-CNN 等架構更為快速，可避免追加過多的計算時間負擔。

傳統上在標註一般訓練物件的外接框時，是指定外接框的左上角與右下角來取得位置資訊，有鑑於空拍影像中行人與兩輪小型車輛等並非矩形，加上面積小難以準確決定其實際範圍，因此本研究在行人與自行車、機車的訓練上，統一採用正方形外接框進行標註。如此僅需指定小物件中心位置，配合固定邊長即可快速對大量相關物件進行標註，大幅減輕標註的困難度。

圖 8 為透過 YOLOv3 偵測行人與自行車、機車之結果圖，其中，紫色、紅色與褐色框分別代表偵測出之行人、自行車與機車，框上數字則為物件編號，從圖中可以發現行人面積相當小，僅占約 10×10 的像素範圍，人眼已不易分辨，另一方面，自行車與機車的差異也很難透過人眼進行快速的判別。以目前訓練結果而言，緊鄰的人群與部分自行車有時仍會被誤認為機車，但機車由於收集的樣本數較大，因此單張畫面上的機車失偵 (miss detection) 與誤偵 (false detection) 情況相較自行車與行人較低。

在行人與兩輪小型車輛的追蹤問題上，由於 YOLOv3 偵測出的小物件區域難以包含足夠多的特徵點以進行可靠追蹤，因此本研究利用 SORT 演算法對偵測出之行人與兩輪小型車輛進行追蹤。由於 SORT 演算法主要基於卡爾曼濾波器進行線性運動預測，將預測結果

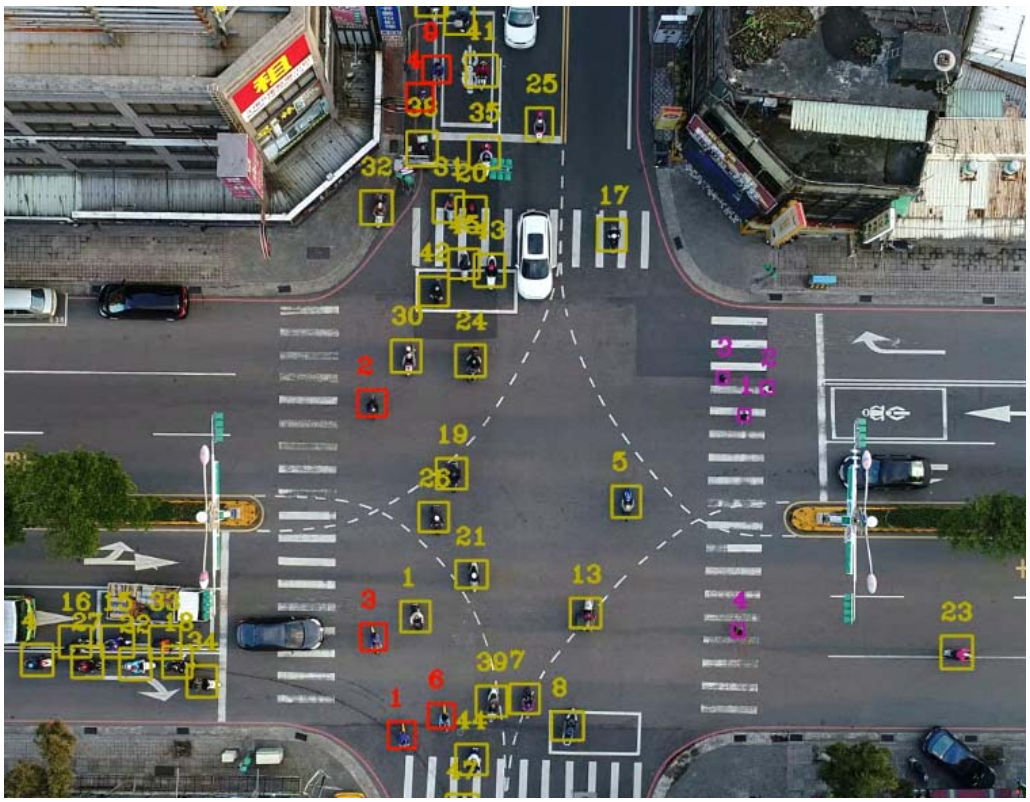


圖 8　YOLOv3 偵測行人與自行車、機車

與偵測結果的偵測框進行匹配，取出高重疊率的匹配組合來完成追蹤，因此不會受路面線道等顯著外觀變化影響。考量本研究採用之空拍影像不會有車輛間的遮蔽問題，且經過晃動校正排除攝影機運動 (camera motion) 影響後，一般兩輪車輛與行人在短期間內的運動模式亦多符合線性假設，因此採用 SORT 演算法來追蹤小物件的移動軌跡。另一方面，Deep SORT 雖然改善了一般監視器角度下，容易因遮蔽引起的物體編號改變問題，但由於本研究採用之空拍畫面在先天上已避免了傳統攝影機的遮蔽問題，加上 Deep SORT 原始訓練出的外觀資訊來自一般攝影機視角之側拍行人資料庫，因此也不適合直接用於本研究空拍影像中的俯視行人與機、腳踏車等情形。

在軌跡表現上，由於小物件難以利用 Mask R-CNN 取得其實際範圍，因此我們利用 YOLOv3 架構偵測以小物件中心為基準的正方形外接框，並對行人、自行車、機車賦予特定的不同邊長大小。此外接框雖有利於資料標註上的方便性，但卻明顯小於對象的實際所佔區域 (特別是對自行車與機車而言)。為了提取出更精準的小物件範圍，本研究針對小物件訂定了對應的矩形範圍，並根據下一刻的位移方向來決定行人／車體方向，若下一刻位移量太小，則再往後續軌跡點尋找，直到找出一個與現今軌跡點間有足夠明顯位移 (例如機車車身長度之一半) 的軌跡點，作為車體方向之參考。以圖 9 為例，軌跡上的一連串圓

點代表時間點 t_0 至 t_7 的連續機車中心位置，黃色矩形則代表了機車範圍，紅線代表機車車體主軸的參考方向。其中，t_1 與 t_2 由於定點等待等原因，造成中心位置十分接近。為了讓車體主軸不會因短距離的位移而產生不合理的大幅度改變，因此我們利用較遠的 t_3 時間點位置與 t_1 時間點位置所成之連線向量，做為 t_1 時車體方向之參考。這樣的做法有助於提供慢速或靜止機車一個較穩定一致的車體主軸方向。此外，由於最後一個軌跡點並沒有後續軌跡點可供參考，因此使用前一個足夠明顯位移的軌跡點，作為車體主軸方向之參考，如圖中所示時間點 t_7 的車體主軸。

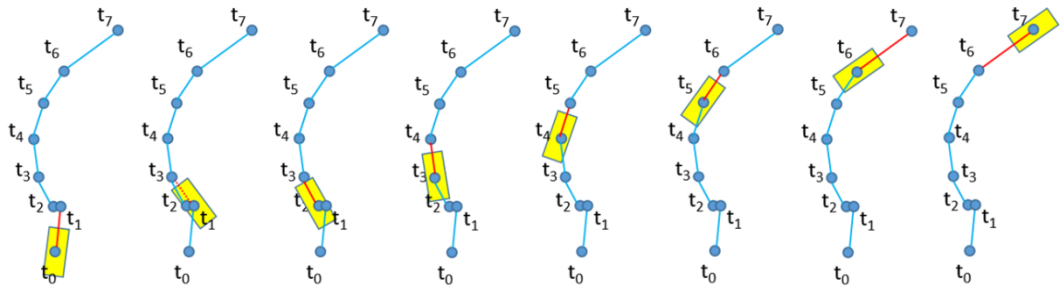


圖 9　透過小物件之前後中心位置決定小物件範圍

原本透過 YOLOv3 偵測出的行人、自行車與機車等方形區域，經過 SORT 追蹤取出中心點之連續移動軌跡後，重新以 11×11、11×25、15×35 等 3 種矩形範圍來分別代表行人、自行車與機車所佔空間，並根據軌跡點的後續移動方向來決定車身方向。圖 10 顯

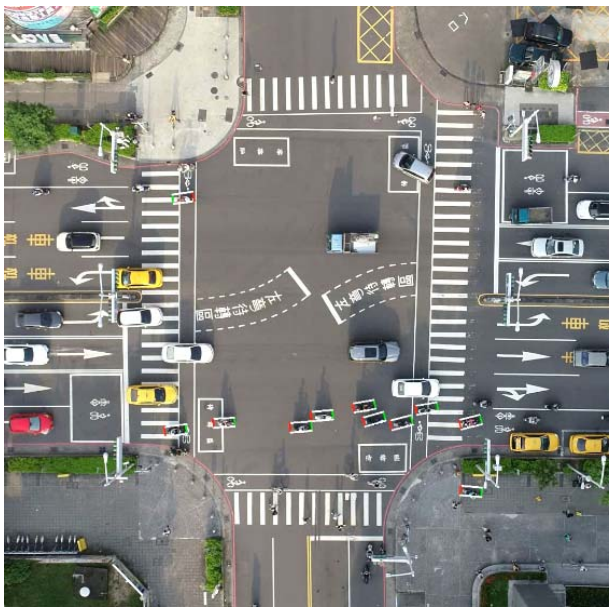


圖 10　透過小物件之前後中心位置決定小物件範圍

示了行經路口之機車代表區域，每部機車車頭、車尾分別以綠線與紅線表示。由圖中可見機車即便是通過斑馬線亦能有效追蹤，線道不會對車輛追蹤之結果造成影響明顯。

在 YOLOv3 架構下，錯誤的軌跡資訊來自三個方面：

1. YOLOv3 在偵測上的失偵，這類情況可能造成追蹤上找不到適當繼承者的問題。目前透過 SORT 演算法可以透過線性運動假設來對失偵車輛進行短期位置預測，因此可以容忍短期的失偵現象，並給予重新偵測到之小物件同樣編號。一旦發現該編號之小物件重新被追蹤成功，再透過線性內插方式填補失偵期間的小物件位置。
2. YOLOv3 在偵測上的誤偵／誤分類。一般來說，單純的誤偵情況遠較失偵情況更少，影響甚小。由於誤偵對象一般為道路上不具移動性的物品，且不易被連續偵測出，因此實務上會被 SORT 演算法以新物件無法連續追蹤成功 3 次之條件加以排除，難以形成軌跡。另外一種誤分類情況，通常是行人或自行車被誤認為機車車輛。由於目前本研究在追蹤上是對予行人、自行車、機車三者統一進行追蹤，如同四輪車輛作法先產生軌跡，再根據同一軌跡在不同時間的分類情況進行投票，決定該軌跡所屬的車種，因此可減少因誤分類而產生的追蹤不連續問題。
3. 追蹤對象的混淆導致同一物件編號改變，這類情況在機車擁擠的情況下較容易發生。在採用 SORT 演算法後，由於其引入了卡爾曼濾波器來區分不同運動趨勢的物件，並採用匈牙利演算法優化物件群間的整體配對誤差，因此表現相當良好。惟有部分慢速劇烈轉向車輛在失偵時，可能因為前後運動趨勢的差異性而改變了追蹤編號。

考慮以上各種可能的錯誤軌跡成因，在引入 SORT 演算法後多數獲得了一定程度的改善，未來可考慮讓卡爾曼濾波器引進非線性之運動趨勢估算，抑或是利用 Deep SORT 針對空拍畫面車輛偵測框進行外觀特徵之訓練，嘗試透過外觀特徵修正原本小物件基於運動的單純匹配模式，取得更可靠之物體預測位置，以強化追蹤效果。

四、實驗分析

為了驗證行人、自行車、機車、小轎車、貨車、巴士、聯結車等偵測與追蹤之實際效果，本研究在臺南市永康區中華路與勝華街交會路口 (簡稱 A 路口)、雲林縣斗六市大學路三段與中山路交會路口 (簡稱 B 路口)、高雄市鼓山區裕誠路與博愛二路交會路口 (簡稱 C 路口)、以及新竹縣竹北市自強北路與勝利七八街交會路口 (簡稱 D 路口) 等四個不同尺度大小路口，從高度 75 公尺處拍攝約 100 秒空拍影片作為實驗對象。每部影片原始規格為解析度 3840×2160、每秒 29.97 幀之 4K 影片，在經過影片穩定化處理後，以路口為中心選取 2048×2048 區域，重新縮小壓製為 1024×1024、每秒 9.99 幀 (即每 3 幀保留 1 幀) 之影片，以節省處理時間並滿足 Nvidia 1080/2080 ti 之 11G 記憶體限制需求。圖 11 依序顯示了 A、B、C、D 四個路口的樣態與單張偵測結果。紫色包圍路口之線段為路口區域邊界，僅進出邊界路口車輛會被記錄，其他始終停滯於邊界內或邊界外之車輛將不被記錄，並排

圖 11　實驗驗證用之四個路口樣態; 左上、右上、左下、右下分別為 A、B、C、D 路口

除在需要偵測追蹤之對象外。四輪車輛分別以不同顏色表示車身邊界，包含小轎車 (紅)、貨車 (藍)、巴士 (綠)、聯結車 (車頭黃色、拖車部分水藍色)。對於較小的偵測對象，實驗中主要以固定寬度方型區域配合不同顏色作為代表，如行人 (水藍)、自行車 (黃)、機車 (白)，並以紅色代表正面方向。此外，在每個框中心可見一條白色線段，指向同 ID 物件的前一幀位置，以便檢查車輛彼此交錯時是否發生 ID 互換的追蹤錯誤情形，每部實驗影片通過路口之車輛數如表 2 所示，每部影片長度約 100 秒，包含 1000 張連續畫面做為測試影片。

在測試影片中，每部車輛需花費數秒乃至數分鐘時間行經路口，為了驗證每部車輛在

行經路口期間的偵測追蹤情況，我們統計了 1000 張畫面中所有車輛的總出現次數，也就是同一部車若在 1000 張畫面中有 100 張畫面在路口區域內，則該部車總出現次數為 100 次。在 4.1 與 4.2 節中，我們將分別討論可偵測追蹤出的車輛數量、整體查準率 (Precision) 與查全率 (Recall) 等數據。

表 2　四部實驗影片路口經過車輛數

	行人與兩輪車輛			四輪車輛				
代號	行人	自行車	機車	小轎車	貨車	巴士	聯結車頭	聯結車身
路口 A	**0**	**3**	**114**	**52**	**9**	**0**	**0**	**0**
路口 B	**2**	**9**	**81**	**59**	**3**	**0**	**2**	**2**
路口 C	**27**	**5**	**161**	**65**	**2**	**2**	**0**	**0**
路口 D	**0**	**0**	**61**	**76**	**2**	**0**	**1**	**0**

由於行人／兩輪車輛以及四輪車輛分別是透過 YOLOv3 以 Mask R-CNN 兩種方法進行偵測，因此我們收集了約 1500 張人工註解影像，其中包含了 5014 位行人、1120 輛自行車、18076 部機車用於訓練 YOLOv3 模型；25835 部小轎車、2896 部貨車、189 部公車、303 部聯結車頭與 306 部聯結車身用於訓練 Mask R-CNN 模型。我們的訓練資料在行人、機車、小轎車等方面收集了可觀的數量，惟自行車、貨車、公車、乃至聯結車礙於道路車輛比例問題較難收集。

4.1 四輪車輛偵測與追蹤

由於四輪車輛與較小之行人／兩輪車輛在偵測與追蹤上皆採用了不同架構與機制，兩者間幾乎不會產生相互誤認之情形，因此我們可以將小物件與較大型的四輪車輛之偵測追蹤結果分別進行討論。為釐清車種間的誤認關係，本研究以混淆矩陣 (confusion matrix) 來評估單張畫面下的車種分類結果。在四輪車輛方面，表 3 代表各測試影片中四輪車輛分類的混淆矩陣情況。1000 張測試畫面中的每部車輛都會針對其分類情況進行獨立計數，以 A 路口 (臺南市永康區中華路與勝華街交會路口) 為例，1000 幀測試畫面中，52 部小轎車 (見表 2) 被成功追蹤的次數為 1509 次，僅 10 次被誤分類為貨車；9 部貨車被成功追蹤出 139 次，其中某部大型貨車車頭剛進路口時有 4 幀被誤分類為聯結車頭。

由於貨車、巴士、聯結車相較於小轎車的出現比例不高，因此我們先就數量最多、較具代表性的小轎車進行說明。四個路口中，偵測追蹤出之小轎車幾乎皆為正確分類，僅有總計 18 次誤偵發生在 C、D 路口，此外，有 97 次失偵發生在 B 路口的兩部小轎車上 (全程未被偵測追蹤到)，與及 57 次被錯誤分類為貨車。在貨車方面，除前述 A 路口有大型貨

表 3　偵測追蹤四輪車輛之混淆矩陣; 左上、右上、左下、右下分別為 A、B、C、D 路口

台南市永康區中華路與勝華街交會路口		偵測追蹤結果					
		小轎車	貨車	巴士	聯結車頭	聯結車身	未認出
實際車輛	小轎車	1509	10 誤分	0 誤分	0 誤分	0 誤分	0 失偵
	貨車	0 誤分	139	0 誤分	4 誤分	0 誤分	0 失偵
	巴士	0 誤分	0 誤分	0	0 誤分	0 誤分	0 失偵
	聯結車頭	0 誤分	0 誤分	0 誤分	0	0 誤分	0 失偵
	聯結車身	0 誤分	0 誤分	0 誤分	0 誤分	0	0 失偵
	非四輪車	0 誤偵	0 誤偵	0 誤偵	0 誤偵	0 誤偵	-

雲林縣斗六市大學路三段與中山路交會路口		偵測追蹤結果					
		小轎車	貨車	巴士	聯結車頭	聯結車身	未認出
實際車輛	小轎車	4024	47 誤分	0 誤分	0 誤分	0 誤分	97 失偵
	貨車	0 誤分	102	0 誤分	0 誤分	0 誤分	0 失偵
	巴士	0 誤分	0 誤分	0	0 誤分	0 誤分	0 失偵
	聯結車頭	0 誤分	0 誤分	0 誤分	67	4 誤分	19 失偵
	聯結車身	0 誤分	0 誤分	0 誤分	1 誤分	48	4 失偵
	非四輪車	0 誤偵	0 誤偵	0 誤偵	0 誤偵	0 誤偵	-

高雄市鼓山區裕誠路與博愛二路交會路口		偵測追蹤結果					
		小轎車	貨車	巴士	聯結車頭	聯結車身	未認出
實際車輛	小轎車	3571	0 誤分	0 誤分	0 誤分	0 誤分	0 失偵
	貨車	0 誤分	267	0 誤分	0 誤分	0 誤分	0 失偵
	巴士	0 誤分	0 誤分	87	0 誤分	0 誤分	0 失偵
	聯結車頭	0 誤分	0 誤分	0 誤分	0	0 誤分	0 失偵
	聯結車身	0 誤分	0 誤分	0 誤分	0 誤分	0	0 失偵
	非四輪車	6 誤偵	0 誤偵	0 誤偵	0 誤偵	0 誤偵	-

新竹縣竹北市自強北路與勝利七八街交會路口		偵測追蹤結果					
		小轎車	貨車	巴士	聯結車頭	聯結車身	未認出
實際車輛	小轎車	7360	0 誤分	0 誤分	0 誤分	0 誤分	0 失偵
	貨車	0 誤分	134	0 誤分	0 誤分	0 誤分	0 失偵
	巴士	0 誤分	0 誤分	0	0 誤分	0 誤分	0 失偵
	聯結車頭	0 誤分	0 誤分	0 誤分	485	0 誤分	0 失偵
	聯結車身	0 誤分	0 誤分	0 誤分	0 誤分	0	0 失偵
	非四輪車	12 誤偵	0 誤偵	0 誤偵	0 誤偵	0 誤偵	-

車車頭剛進路口時有 4 幀被誤分類為聯結車頭外，與公車同樣並無失偵誤偵情形發生。相對前述車輛，聯結車較容易發生誤分類與失偵情形，聯結車由於車頭與拖車車身在轉彎時的移動方向不一致，因此在偵測上區分為兩個獨立矩形區域，其中車頭部分較容易與其他車輛混淆或產生誤分類 (A 路口)、失偵等問題 (B 路口)，其主要原因在於難以訓練樣本數難以提高，且車頭型態較無一致性所致。此外，D 路口影片中包含一部聯結車頭 (見圖 11 右下 D 路口空拍圖)，雖然可追蹤出該聯結車頭在影片中之完整軌跡，但由於訓練資料中對於加掛拖板車下之車頭部分，無法標示車頭後半部車體 (如圖 12 所示)，因此在偵測聯結車頭時，有時會產生如圖 12 兩種偵測情況，未來需增加單一聯結車頭之訓練樣本以強化其後半部車體偵測，或透過程式判斷自動補足後半車體。

圖 12　單一聯結車頭偵測結果之差異

我們透過查準率 (Precision) 與查全率 (Recall) 來評估 4 部影片中，各類車輛的整體偵測追蹤情形。根據預測對象與實際結果之不同情況 (如表 4)，查準率與查全率可以分別定義為式 (1) 與式 (2)。其中，查準率代表所有被系統偵測為該類車輛的正確比例、查全率則代表實際該類車輛可被系統正確偵測出之比例。各類型在四輪車輛的查準率與查全率如表 4 所示方面，除聯結車效果稍差、部分載人箱型車有時會被歸類為貨車外，表現相當可靠穩定。

表 4　預測對象與實際結果之排列組合情況

	預測為目標對象 (Positive)	預測為非目標對象 (Negative)
預測正確 (True)	TP (True Positive)	TN (True Negative)
預測錯誤 (False)	FP (False Positive)	FN (False Negative)

$$\text{Precision} = TP/(TP + FP) \tag{1}$$

$$\text{Recall} = TP/(TP + FN) \tag{2}$$

表 5　四輪車輛整體查準率與查全率表現

	小轎車	貨車	巴士	聯結車頭	聯結車身
Precision	99.9%	91.8%	100%	99.1%	92.3%
Recall	99.1%	99.4%	100%	96.0%	90.6%

4.2 行人、自行車、機車追蹤偵測

相對於四輪車輛，行人、自行車、機車等顯著較小物件之偵測，在偵測率上通常較一般大小物件要來得低，隨著近年來深度學習研究課題的不斷擴張，以包含 YOLOv3 為主的多種物件偵測架構正逐漸致力於改善前述情況。另一方面，隨著物件偵測效果的逐步提升，SORT 等基於 tracking-by-detection 的追蹤技術也越加受到重視，甚至結合了深度學習模型，發展出 Deep SORT 等緩解遮蔽問題的進階追蹤方法。但由於本研究採用的空拍影像不存在車輛間的遮蔽問題，因此直接透過 SORT 即可取得相當理想的結果。表 6 為針對 A、B、C、D 四個路口偵測追蹤行人、自行車與機車之混淆矩陣，在行人方面，C 路口有 27 位行人行經路口，較具代表性，在 C 路口中，行人誤分類為其他車輛之比例低於 1%、偵測追蹤失敗率則約占 20%。事實上，由於俯視下行人極度缺乏外觀特徵，因此需要仰賴 SORT 的持續追蹤效果，即使連續 3 至 4 幀未偵測到行人，也可透過線性假設對移動緩慢

表 6　針對 A、B、C、D 四個路口偵測追蹤行人、自行車與機車之混淆矩陣

台南市永康區中華路勝華街交會路口		偵測追蹤結果			
		行人	自行車	機車	未認出
實際人車	行人	0	0 誤分	0 誤分	0 失偵
	自行車	0 誤分	0	173 誤分	117 失偵
	機車	19 誤分	0 誤分	5917	519 失偵
	非小物件	39 誤偵	0 誤偵	13 誤偵	-

雲林縣斗六市大學路三段與中山路交會路口		偵測追蹤結果			
		行人	自行車	機車	未認出
實際人車	行人	996	0 誤分	0 誤分	64 失偵
	自行車	0 誤分	245	621 誤分	28 失偵
	機車	0 誤分	39 誤分	6845	9 失偵
	非小物件	0 誤偵	0 誤偵	0 誤偵	-

高雄市鼓山區裕誠路與博愛二路交會路口		偵測追蹤結果			
		行人	自行車	機車	未認出
實際人車	行人	11404	42 誤分	0 誤分	2829 失偵
	自行車	112 誤分	1311	0 誤分	178 失偵
	機車	6 誤分	0 誤分	17437	1446 失偵
	非小物件	0 誤偵	0 誤偵	0 誤偵	-

新竹縣竹北市自強北路與勝利七八街交會路口		偵測追蹤結果			
		行人	自行車	機車	未認出
實際人車	行人	0	0 誤分	0 誤分	0 失偵
	自行車	0 誤分	0	0 誤分	0 失偵
	機車	0 誤分	0 誤分	6572	526 失偵
	非小物件	0 誤偵	0 誤偵	0 誤偵	-

之行人進行預測，大幅改善了原本的偵測效果。同樣地，機車也因為 SORT 追蹤法的引進而有顯著改善，即使在機車擁擠且數量龐大的 C 路口 (平均每張畫面約有 19 部機車)，失偵率亦僅約 8%。相對於行人與機車，自行車有著明顯較高的誤分類比例，平均約 28.5%的自行車會被誤分類為機車；與及 11.6%的自行車無法被偵測追蹤出來。主要原因有二；一為自行車與機車外型較為接近，且自行車樣本數量明顯偏少，較容易被誤認為機車；二為目前收集之自行車樣本多為 Youbike 等特定外觀之自行車為大宗，故不利於其他非都會區域自行車輛之偵測。

表 7 為行人、自行車、機車整體查準率與查全率表現。整體而言，行人雖然面積小且缺乏外觀特徵，仍可偵測出 80%的行人。自行車雖然可正確偵測出之比例僅接近 6 成，但多數被誤判為相近外觀的機車。若將行人、自行車、機車視為同一小物件類型，有 90%小物件可被偵測出，只是分類上有一定比例的錯誤。

表 7　行人、自行車、機車整體查準率與查全率表現

	行人	自行車	機車	不分類小物件
Precision	98.6%	95.1%	97.9%	99.9%
Recall	80.9%	55.9%	93.5%	90.1%

4.3 實驗討論

整體而言，本研究在四輪車輛偵測與追蹤上可以達到可靠的追蹤結果，在行人、自行車與機車等小物件偵測上，在不分類前提下亦可達到 90%的正確追蹤結果，惟自行車礙於訓練樣本數量與外觀種類上的劣勢，容易被誤分為機車車輛。由於行人、自行車、機車一般有著截然不同的移動速率差距，因此事實上也可以透過計算軌跡中移動時的平均速率，對分類結果進行重新校正。此外，各車輛誤偵比例皆非常低，且在機車方面獲得還算不錯的偵測追蹤效果，YOLOv3 雖然在小物件偵測上仍有進步空間，但因為誤偵情況低，儘管在斑馬線等複雜線道上，仍可透過 SORT 等追蹤方法彌補數張畫面內的失偵問題，並重新透過線性差補取得完整之軌跡資訊。圖 13 展示了路口 C 中部分軌跡抽取結果，可以清楚看見軌跡以行經區域的方式呈現，有助於反映大型車內輪差、計算車輛間距等目的。

五、結論

本研究應用了多種深度學習與影像處理技術以取出空拍影像中之車輛軌跡，包含空拍影片穩定化、基於深度學習的行人與車輛偵測分類、應用傳統影像處理技術進行車輛追蹤等。本研究主要有下列三項貢獻：

圖13　路口C中部分軌跡視覺化結果。每部人車以不同色表示軌跡，亮度深淺表時間近遠

1. 結合空拍機與深度學習、影像處理等多項跨領域技術，以實際的十字路口為測試對象，實作完成一個高度自動化的人車軌跡抽取技術，此技術有助於協助交通資訊的收集，減低傳統所需的人力成本。
2. 利用 Mask R-CNN 與 YOLOv3 等深度學習物件偵測技術，分別針對四輪車輛與行人/兩輪車輛特性發展適合的偵測/追蹤策略，有效改善單一方法難以解決的小物件偵測與偵測框貼合四輪車輛等問題。
3. 將傳統以車輛中心點所連線段而成的軌跡表示方式，提升至以車輛行經區域表示。在軌跡資訊上從點與線提升至面的階段，有助於提供更完整的車輛連續位置資訊，可進一步達到反映內輪差、計算車輛間間距等功能。

本研究目前所發展之技術已有一定的可靠程度，未來隨著空拍交通影像資料的日漸增加，配合日新月異的深度學習技術發展，相信針對空拍影像的人車軌跡抽取技術未來可以成為交通分析上不可或缺的重要工具之一。

參考文獻

1. 交通部運輸研究所，**機車交通政策白皮書**，民國 105 年。
2. 黎俊彬，「號誌化平面路口對向直行左轉車輛安全通行之研究」，國立交通大學運輸科技與管理學系碩士論文，民國 94 年。
3. Ren, S., He, K., Girshick, R. and Sun, J., “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 6, 2017, pp.1137-1149.

4. He, K., Gkioxari G., Dollár, P. and Girshick, R., “Mask R-CNN”, IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 2980-2988.
5. Redmon, J., Divvala, S., Girshick, R., Farhadi, A., “You Only Look Once: Unified, Real-Time Object Detection”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp.779-788.
6. Redmon, J. and Farhadi, A., “Yolo9000:better,faster,stronger”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 6517-6525.
7. Redmon, J. and Farhadi, A., “YOLOv3: An Incremental Improvement”, ArXiv e-prints, 1804.02767, Apr. 2018.
8. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., “Feature Pyramid Networks for Object Detection”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 936-944.
9. Welch, G. and Bishop, G., “An introduction to the kalman filter”, Technical Report TR 95-041, University of North Carolina, Department of Computer Science, Chapel Hill, 2006
10. Ristic, B., Arulampalam, S. and Gordon, N., “*Beyond the Kalman Filter: Particle Filters for Tracking Applications*”, Artech House, London , 2004.
11. Henriques, J. F., Caseiro, R., Martins, P. and Batista, J., “Exploiting the Circulant Structure of Tracking-by-Detection with Kernels”, European Conference on Computer Vision, Vol. 7575, 2012, pp.702-715.
12. Farneback, G., “Two-Frame Motion Estimation Based on Polynomial Expansion”, *Lecture Notes in Computer Science*, Vol. 2749, 2003,pp.363-370.
13. Bewley, A., Ge, Z., Ott, L., Ramos, F. and Upcroft, B., “Simple Online and Realtime Tracking”, IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, 2016, pp. 3464-3468.
14. Kuhn, H. W., “The Hungarian Method for The Assignment Problem”, *Naval Research Logistics Quarterly*, Vol.2, Iss.1-2,1955, pp.83-97.
15. Wojke, N., Bewley, A. and Paulus, D., “Simple Online and Realtime Tracking with a Deep Association Metric”, IEEE International Conference on Image Processing (ICIP), Beijing, 2017, pp. 3645-3649.
16. Evangelidis, G. D., Psarakis, E. Z., “Parametric Image Alignment Using Enhanced Correlation Coefficient Maximization”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 30, No.10, 2008, pp.1858-1865.