

# 基於行車攝錄視訊之移動物偵測方法

## The Moving Object Detection Method for Car-Camera Video

陳昭和、陳聰毅、林俞均、駱冠銘

國立高雄應用科技大學電子工程系

E-mail: {thouho, chentso, 1101305140, 1102305119}@kuas.edu.tw

### 摘要

本文提出一種適用於車用型視覺系統的移動物偵測方法。本文方法主要分成三個部分：(1)前處理：須先設置出感興趣區塊以用來設限移動物偵測範圍，並判定車子是否在移動中以進行特徵點分類處理；(2)特徵點分類：使用角點偵測與光流法來取得攝影畫面的光流向量，根據該向量的方向、長度以及分布區塊來做為前景特徵點、背景特徵點與誤判光流的分類基準；(3)移動物偵測：利用背景特徵點以計算出單應性矩陣並將前一幀的影像進行透視變換，再與目前影像做影像相減以得到一個差值影像以用來做移動物偵測處理。實驗結果顯示本文方法在晴天正常天候情況下可有效地偵測出移動物，平均偵測率達81%以上，誤判率則約14%，明顯優於其他方法。

**關鍵詞：**移動物偵測、光流法、角點偵測、單應性矩陣、透視變換。

### Abstract

This paper proposed a moving object detection method for vehicle vision system, which can effectively detect the moving objects shown in the image sequence. The detection system included three steps: (1) A region of interest (ROI) is set to limit a range of the moving-object detection for checking whether a vehicle is moving or not, and then followed by the classification of feature points. (2) Both corner detection and optical flow are utilized to obtain motion vectors of feature points, and the foreground feature points, background feature points, and misjudgment of optical flow are classified according to the motion vector's direction, length, and distribution. (3) Background feature points are employed to calculate a homography matrix which is used to perform a perspective transform in the previous frame, and then the moving-object detection is based on the difference between the transformed result and the current frame. Experimental results demonstrate the proposed method can provide a detection rate of 81% and false detection rate of 14% on average, which is superior to other methods.

**Keywords:** Moving object detection, Optical flow, Corner detection, Homography matrix, Perspective transformation.

### 1. 前言

現在行車紀錄器非常普遍，若能透過行車紀錄器所拍攝的畫面並藉由影像處理，抓出車前的移動物，這對未來安全駕駛輔助系統是一大進展。目前常見的前方移動物偵測為使用硬體感測器來做輔助，但相對的成本就會提升，故利用攝影機中的影像資訊來做移動物切割處理，可節省這些硬體的成本支出和與行車紀錄器整合之效益。若以影像處理的領域來看，機器學習方式大多都以HOG為基礎[1]，雖具偵測準確之優點，然HOG計算耗時而不適合即時應用。目前移動攝影機的移動物切割，就近年來的兩篇文獻[2]、[3]都適用在手持攝影機、機器人視覺的部分，發展原理是將雙眼視角套用在單攝影機上去實作，這在移動速度較為緩慢的情形下，偵測結果會較合乎要求，但當載具的速度變快，偵測結果並不理想。文獻[4]是利用車道線去做前方移動物偵測，只是在市區街道場景中，並非常有可截取到車道線的機會。為克服上述問題，本文方法仿效文獻[2]、[3]之架構，然將雙眼視角原理改成單視角的原理以達成車用型視覺系統的移動物偵測功能。

### 2. 系統架構與流程

本文所提之車用型視覺系統的移動物偵測方法主要分成三個部分：前處理、特徵點分類、移動物偵測，其系統流程如圖1所示。

#### 2.1 前處理

前處理主要是將影像縮減、讀取ROI之相關資訊以及判定汽車是否處在移動狀態。影像縮減主要是為了減低運算量，ROI設定的部分，由於每一台行車紀錄器被架設之位置、角度不同，且每台行車紀錄器的可拍攝角度、畫質也不同，外加每個人想要抓取的區域範圍也不同，所以將系統中的移動物偵測區塊交給使用者自行設定，當區塊範圍設定完畢之後，系統會保留住所設定的資訊；再來會做影像相減，並利用剛剛所設置的區塊資訊取出區塊以外之上半部差值影像，以判定車子是否處在移動狀態，如圖2所示。判定車子是否處在移動狀態的程序是：首先，先統計出判定範圍內的像素總數( $total\_n\_p$ )，如公式(1)，當中的 $y_1$ 是在設定移動物抓取的區域中，最靠近 $y=0$ 的那一個 $y$ 值(即 $y_1$ )，剩下的 $w$ 分別為影像的寬；再來，會使用一個暫存器

(count)來計數相鄰相減後的含有差值的總數，利用圖像中有差異的地方與式(1)所得之結果，可計算出差值所佔的百分比，如式(2)；最後則判定  $s\_percent$  值是否低於一個門檻值；若否，則表示在靜止狀態，如式(3)。

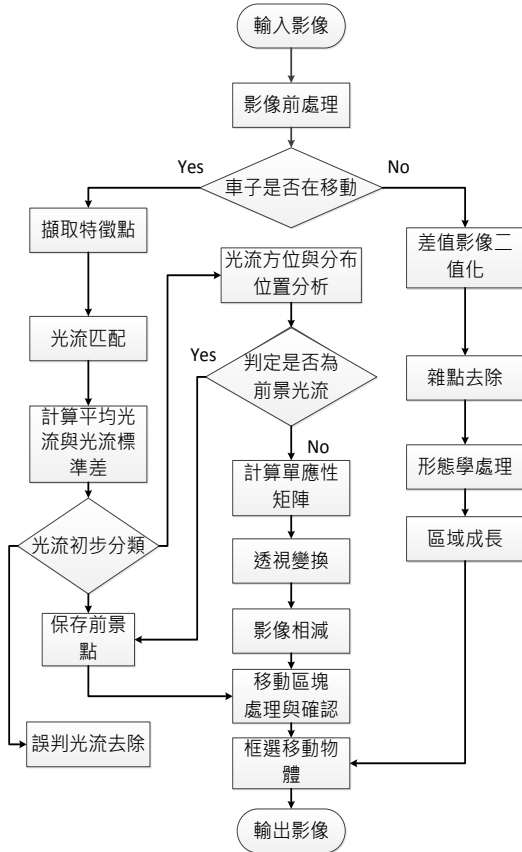


圖1. 系統流程圖

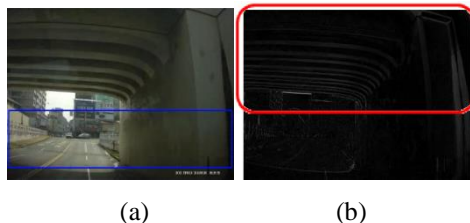


圖2. ROI與ROI之外區塊示意圖；(a)原圖，藍色框圍ROI區域；(b)相減後的圖像，紅色區塊為判定車子移動區域。

$$total\_n\_p = w * y1 \quad (1)$$

$$s\_percent = \frac{count}{total\_n\_p} \times 100\% \quad (2)$$

$$Status = \begin{cases} move & , if (s\_percent) > Thr \\ static & , if (s\_percent) \leq Thr \end{cases} \quad (3)$$

式(3)中的  $Thr$ ，經實驗統計，發現在0.5%的時候會有一個較好的判定結果，當中或許會有一些雜訊會

干擾判定結果，所以需設置一個狀態暫存器，來存放一秒內的所有結果，當這一秒內多數都是呈現在靜止的時候，最終則判定靜止情況，這樣就可以避免掉系統一下判定靜止，一下判定移動的情況。

## 2.2 特徵點分類

首先將影像進行邊緣偵測(edge detection)，這邊是採用Sobel邊緣偵測的方式，其邊緣偵測後的結果如圖3所示。有了邊緣偵測的圖像後，再來就是利用角點偵測找出邊緣影像中的角點，其做法原理是根據某一像素點檢視周遭的色階變化，當落在平滑區的時候，遮罩內的資訊並無方向的變化，所以所抓出的角點就是特徵點部分，其特徵點抓取結果如圖4所示，影像中的藍點為特徵點。接著採用LK金字塔光流法找出每一個特徵點於下一時間點的影像位置(即光流資訊)，匹配後的結果如圖5所示，圖中綠色為匹配後的特徵點，紅色為前一幀之特徵點，若只有單一綠色表無光流長度。



圖3. Sobel 邊緣偵測結果



圖4. 特徵點偵測結果

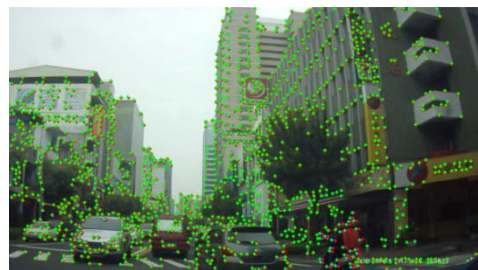


圖5. 特徵點匹配後之結果

從所得之光流資訊可以分為3種光流:前景光流、背景光流、誤判光流。首先，必須先把誤判光流給剔除，作法是先計算出所有光流的平均長度 ( $F\_avg$ )與標準差 ( $F\_std$ )，如公式(4)與(5)，之後再藉由這兩個值得到兩個門檻值  $flow\_thr1$ 、 $flow\_thr2$

，其定義如式(6)、式(7)，當中的 $F_k$ 是介於0~1之間的參數，其定義如式(8)，而 $flow\_max$ 為所有光流中最大的光流。特徵點分類的結果如圖6所示。

$$F\_avg = \frac{1}{n} \sum_{i=1}^n Fi \quad (4)$$

$$F\_std = \sqrt{\frac{1}{n} \sum_{i=1}^n (F\_avg - Fi)^2} \quad (5)$$

$$flow\_thr1 = F\_avg + F\_k \times F\_std \quad (6)$$

$$flow\_thr2 = \frac{flow\_thr1 + flow\_max}{2} \quad (7)$$

$$F\_k = \frac{mf - n}{n} \quad (8)$$



圖 6. 特徵點分類的結果

由於是針對車用型視覺系統來切割移動物的狀態下，必須了解當攝影機有移動的時候，並不是每個位置的光流都是相等長度的，而這個移動量在攝影機中的影像就會有『越遠之移動光流越似靜止光流』的情況，所以當前一幀的影像要校正成當前影像，就需要做形變的處理，作法如公式(9)。

$$P'_i = H \cdot P_{i-1} \quad (9)$$

(9)式中的 $H$ 是一個變換矩陣-單應性矩陣， $P_{i-1}$ 為時間 $i-1$ 的影像中任一點， $P'_i$ 為 $P_{i-1}$ 對應至 $P_i$ ；我們可用另一種形式來表示每個符號的內容，如式(10)。

$$P_{i-1} = \begin{bmatrix} x_{i-1} \\ y_{i-1} \\ 1 \end{bmatrix}, P'_i = \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix}, H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \quad (10)$$

從(10)式中可以得知 $H$ 矩陣最少需要匹配4組點，即8個未知數。而從前面特徵點分類過後所得的背景特徵點，然後再利用這些特徵點計算出所需的變換矩陣，主要是使用了隨機抽樣一致演算法(RANSAC, RANdom Sample Consensus) [5]來求得 $H$ 中的未知數，再利用 $H$ 將前一幀影像校正回來，校正後的影像再與目前影像相減所得的結果可以消除很多背景的資訊了。

## 2.3 移動物偵測

汽車可能會處在移動狀態或靜止狀態，所以須判定車子是否處在移動或是靜止，判定後的結果分別會有對應的移動物偵測處理，而動態中需要有特徵點的擷取與分類。當系統判定是靜止的時候，就會利用前面所得之差值影像進行移動物偵測，為了避免破碎多個物體，所以會先將二值化影像做形態學中的膨脹處理，最後再將經過處理的連續兩張差值影像累加，結果如圖7所示(紅色框選的)。



圖 7. 靜態的移動物偵測結果(紅色框選的)

動態的移動物資訊處理主要是在透視相減之後，有些時候它可以很齊全的將圖像轉變過去，但有時候邊邊角角是沒有圖像資訊的，這個在文獻[3]也有提到這個問題，所以這樣的情況到了最後都會導致相減的時候邊邊角角的資訊會遺失掉。雖然這部分並非要偵測之移動物範圍，但如果直接取二值化並做後續的形態學處理，就會導致邊緣被誤判成移動物，而本文方法利用ROI區域的二值影像解決邊緣的問題，再利用形態學處理以及MER就可以將移動物給框選出來了，結果如圖8所示(紅色框選的)。



圖 8. 動態的移動物偵測結果(紅色框選的)

## 3. 實驗結果

本研究所採用的影像擷取設備為DOD F880LHD FULL HD，架設於車內，影像的錄製採用1920\*1080之解析度，每秒30張的速度來錄製影像，本研究之系統程式是在Microsoft Visual Studio 2008之Windows Form應用程式專案環境建構下暨搭配OpenCV 2.3.1之函式庫所建造完成。本實驗測試影片有5個，偵測結果(紅色框選的)如圖9所示，test1為停紅燈時的影片；test2為單純直行，偶爾停個紅燈的影片；test3為先直行後停紅燈，停紅燈時，大多為左右來向之移動物，之後再直行；test4為直行後稍微右轉；test5為直行後稍微左轉。

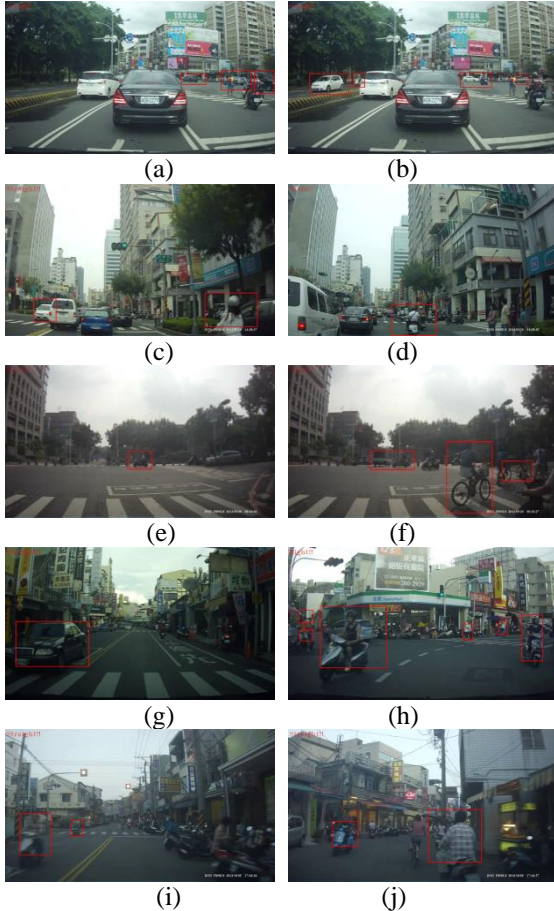


圖 9. 本文方法偵測結果: (a)與(b)為 test1 的實驗結果; (c)與(d)為 test2 的實驗結果; (e)與(f)為 test3 的實驗結果; (g)與(h)為 test4 的實驗結果; (i)與(j)為 test5 的實驗結果。

本文方法之移動物偵測率，如表 1 所示，當中的偵測率與誤判率算法如公式(11)和(12)。

$$\text{偵測率(DR)} = \left( \frac{DN}{MN} \right) \times 100\% \quad (11)$$

$$\text{誤判率(FR)} = \left( \frac{FN}{DN+FN} \right) \times 100\% \quad (12)$$

MN 是移動目標數量，DN 是偵測成功數量，FN 是偵測誤判數量，DT 是偵測時間。

表 1. 本文方法之移動物偵測率

名稱	MN	DN	FN	DR	FR
test1	5619	5146	243	91.6%	4.2%
test2	10363	8548	567	82.5%	5.2%
test3	10742	10054	710	93.6%	6.2%
test4	4272	3426	873	80.2%	17 %
test5	1717	1364	377	79.4%	18 %

從實驗結果得知，本文方法雖然無法將所有移動物給判定出來，但是至少不會把靜止物抓出，而

文獻[1]的部分，只有訓練了汽車的資訊，所以只有偵測到汽車的部分，包含靜止的車輛，只要訓練樣本夠，偵測率是很高的，但相對的誤判率也會跟著提高。

#### 4. 結論

本文提出一種適用於車用型視覺系統的移動物偵測方法，主要可分成:前處理、特徵點分類、移動物偵測等三個部份，透過本文方法，可明確地偵測出多個移動物於影片中。由於背景環境是處在持續變化的狀態，若只用過去的移動物切割是無法適用的，習知的相關偵測技術，大多也都是使用分類器，然其偵測時間太久，所以本文方法的優勢就是能夠及時處理，且準確率也不至於太差。

#### 參考文獻

- [1] N. Dalal, and B. Triggs, "Histograms of Oriented Gradients for Human Detection", 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, USA, Vol.1, pp. 886-893, 2005.
- [2] C. H. Chen, Z. C. Wu, T. Y. Chen, and C. M. Chen, "Moving Object Segmentation Method for Video Captured by a Moving Camera", The 3<sup>rd</sup> National Conference on Web Intelligence and Applications (NCWIA), pp.226-229, 2013.
- [3] C. H. Chen, T. Y. Chen, W. C. Hu, Z. C. Wu, and Z.C. Jao, "Multiple Moving Targets Detection Method for Image Sequences Captured by a Moving Camera", The 4<sup>th</sup> National Conference on Web Intelligence and Applications (NCWIA), 2014.
- [4] C. H. Chen, Y. R. Chen, T. Y. Chen, D. J. Wang, and C. T. Li, "A Video Processing Driver-Assistance System by Lane Detection", The 6th Intelligent Living Technology Conference, Taichung, Taiwan, pp.2173-2179, 2011.
- [5] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography", Comm. of the ACM, vol.24, no. 6, pp.381-359, 1981.