

以 DSP 平台實現之前方車道與車輛偵測系統

A DSP-based Lane Departure and Vehicle Detection System

吳炳飛

國立交通大學教授

bwu@cssp.cn.nctu.edu.tw

張家維

國立交通大學研究生

chiawei@cssp.cn.cntu.edu.tw

陳昭榮

國立交通大學研究助理教授

cjchen@cssp.cn.cntu.edu.tw

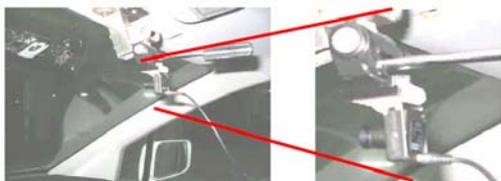
摘要

本文將呈現一個以 DSP 平台實現之適應性車道偏離警示與前方車輛偵測系統，此系統可應用於高速公路、隧道和市區道路，以及在白天和晚上的晴天、陰天和雨天均可運行良好，利用視覺偵測演算法可偵測各種不同型態的車道線，包含實線、虛線以及雙實線等。此系統的 DSP 影像處理器的頻率為 600 MHz，輸入影像尺寸為 CIF，路面標線與車輛的偵測速度超過每秒 25 張影像。本文所呈現的快速適應性車道與車輛偵測演算法為有效率且強健的並適用於白天及晚上。本系統裝設於智慧車 TAIWAN iTS-1，並且成功地於國道 3 號高速公路及 68 號快速道路實現。

關鍵詞：電腦視覺、車輛偏離、車輛偵測、嵌入式平台

一、系統介紹

本系統的特色在於以 DPS 平台的影像處理器 (DSP Image Processor, DSPIP) 來實現即時的前方車道與車輛偵測演算法，此演算法是利用預測證實與更新的準則，並且使用改良過的 3D 座標轉換為影像座標的模型。本系統利用具有 6mm 透鏡的單眼 CCD 來偵測路面的標線並估測前方車輛在其車道線內的位置，見圖一，此 CCD 被安裝於智慧車 TAIWAN iTS-2 的擋風玻璃後方，見圖二。



圖一 前方攝影機架設位置



圖二 Taiwan iTS-2

本文的組織架構如下，第二節介紹車道線偵測演算法，第三節描述車輛偵測和車輛位置估測，實驗成果於第四節說明，第五節為結論與未來展望。

二、車道偵測

本文提出的車道偵測演算法是使用兩階段處理程序來增進效能，第一階段處理的影像是用於缺乏先前車道線像素和車道趨勢的資訊，第二階段是建立在得知車道線像素和車道趨勢資訊下所處理的影像，但若出現無法辨識車道線的情況，則回到第一階段的處理。第一階段是在所設定的範圍內作全域搜尋，被偵測到的車道線會逼近於二次多項式，其在影像中的曲率被視為車道趨勢的資訊。在第二階段的處理中，由於具有第一階段提供參考資訊，因此搜尋車道線的範圍較第一階段的搜尋範圍小。

(一)座標轉換

在最初的偵測期間，在世界座標的點 (x, y, z) 被投影到影像座標 (u, v) ，見圖三，參考世界座標所得到的投影平面可由下式描述：

$$u = e_u \frac{x}{y} \quad (1)$$

$$v = e_v \frac{z - H}{y} \quad (2)$$

其中 $e_u = f/du$ ， $e_v = f/dv$ ，而 du 與 dv 分別為影

像像素的寬與高。本文所提出的演算法是將車道線近似成參數化的二次曲線模型，如下式所述：

$$x = k \cdot y^2 + m \cdot y + b \quad (3)$$

利用(1)與(2)解聯立方程式，可得到(4)和(5)如下，其中 $z = m_\theta \cdot y$ ：

$$x = \frac{u \cdot H}{e_v \cdot m_\theta - v} \cdot \frac{e_v}{e_u} \quad (4)$$

$$y = e_v \frac{H}{e_v \cdot m_\theta - v} \quad (5)$$

將(4)與(5)代入(3)可得到(6)：

$$u = \frac{ke_u e_v H}{e_v m_\theta - v} + me_u + \frac{be_u}{He_v} (e_v m_\theta - v) \quad (6)$$

此即為在影像座標 (u, v) 中包含道路斜率 m_θ 的道路模型。

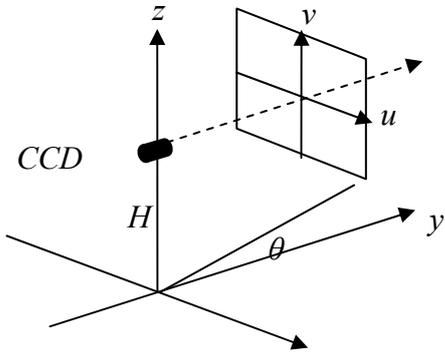


圖 3 世界座標與影像座標關係圖

$P_i(x_i, y_i, z_i)$ 為世界空間中車道線的位置座標，其中 $i \in \{L, M, R\}$ ， L 、 M 和 R 分別代表車道線左、中及右邊界，車道線中點 P_M 定義為 $P_M = (P_L + P_R)/2$ ，在影像平面中，座標點 (u_i, v_i) 為車道線資訊，利用(1)式和(2)式，可得到

$$\frac{1}{2}(u_R + u_L) = u_M = e_u \frac{x_M}{y_M} \quad (7)$$

$$(u_R - u_L) = e_u \frac{x_R - x_L}{y_M} = e_u \frac{W}{y_M} \quad (8)$$

$$v_R = v_L = v_M = e_v \frac{z_M - H}{y_M} \quad (9)$$

其中 W 為車道線寬度。 x_M 、 y_M 和 z_M 可由(7)、(8)和(9)式得到

$$x_M = \frac{u_M \cdot W}{u_R - u_L} \quad (10)$$

$$y_M = e_u \frac{W}{u_R - u_L} \quad (11)$$

$$z_M = H + \frac{v_M \cdot W}{u_R - u_L} \cdot \frac{e_u}{e_v} \quad (12)$$

將(10)與(11)代入(3)式，可得

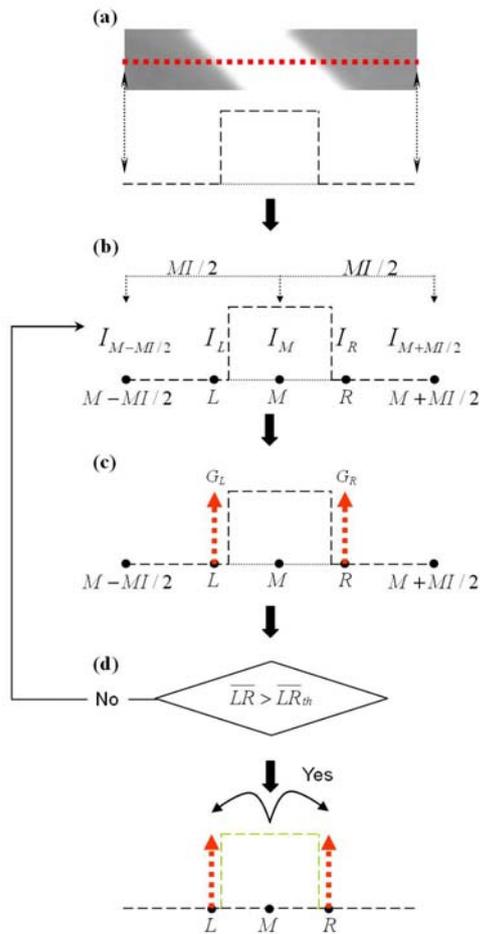
$$u_M(u_R - u_L) = ke_u^2 W + me_u(u_R - u_L) + \frac{b}{W}(u_R - u_L)^2 \quad (13)$$

(6)式和(13)式表示影像座標中的車道線模型，當 m_θ 已知，可得到(6)式的值，而(13)式是假設在 W 為給定常數的條件，在本文的演算法中，將結合(6)和(13)來預測(3)式中的 k 、 m 和 b 參數。在給定所偵測到的車道線參數 (u_L, u_R) 的條件下，(13)式可由加權最小平方來近似，因此 k 、 m 和 b 參數可由計算得知。

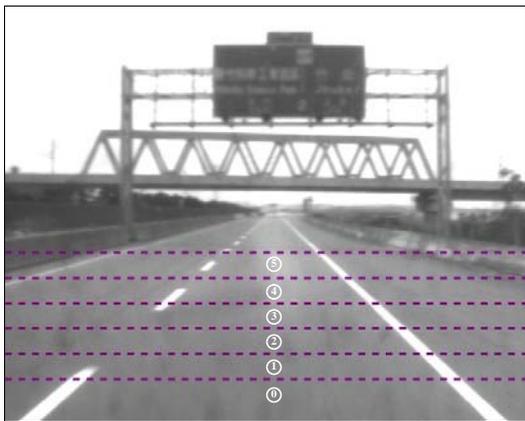
(二)車道線偵測

本文所提供的車道線偵測演算法遵循預測證實與更新的準則。在預測的部份，當缺乏前一張的車道線資訊時，使用一階泰勒展開式來估測下一張影像車道線資訊的候選點；當前一張影像偵測到足夠的車道線點數，則利用最小平方近似法來逼近車道線。此演算法不但可預測下一張影像的估測，而且下一張影像搜尋的興趣區間亦決定。

車道線的灰階強度具有暗亮暗(dark-light-dark, DLD)的特性，見圖四(a)；首先搜尋一中心點 M 位於 DLD 過渡帶，此 M 點的灰階強度 I_M 超過左右兩邊延伸車道線半寬 $MI/2$ 的灰階強度，見圖四(b)；車道線的邊緣點 L 及 R 被定義為在區間 $[M - MI/2, M)$ 與 $(M, M + MI/2]$ 中梯度最大的兩點，見圖四(c)；當 L 與 R 被偵測出來， \overline{LR} 很可能是車道線，如圖四(d)；若 $\overline{LR} > \overline{LR}_{th}$ ，其中 \overline{LR}_{th} 為預設的車道線寬度門檻值。車道線的搜尋為由影像下方開始，逐列逐區間向上搜尋，見圖五，當某些區間完成車道線偵測時，便以此車道線趨勢預測下一個區間車道線可出現的範圍，最後利用最小平方近似法來逼近車道線。



圖四 車道線偵測流程

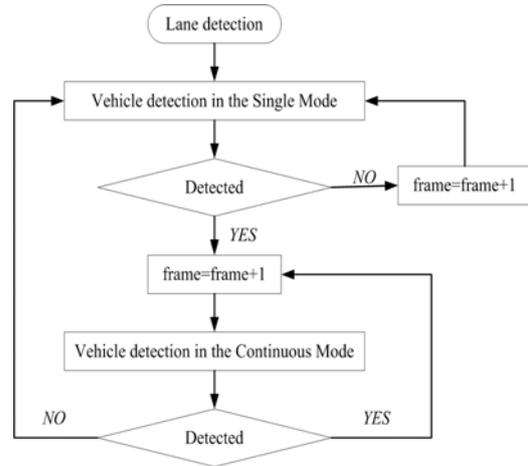


圖五 偵測區間

三、車輛偵測

本系統另一核心演算法主要基於影像辨識技術來偵測前方車輛與其相對距離，本演算法的流程主要分為兩大部份，分別為 Single Mode 與 Continuous Mode，見圖六。Single Mode 主要用於

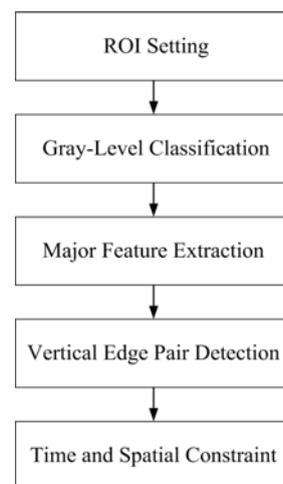
沒有先前偵測資訊或前次偵測錯誤時，採用較廣域的搜尋範圍來偵測前方車輛；在 Single Mode 中偵測到前方車輛後，即進入 Continuous Mode，利用前次車輛偵測資訊，將搜尋範圍縮小，以加速演算法運算時間。



圖六 前方車輛偵測流程圖

(一) Single Mode

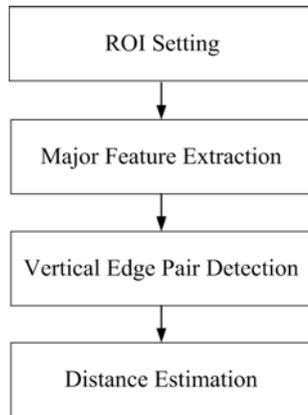
在 Single Mode 中演算法偵測流程，見圖七，由於沒有前次的偵測結果，若針對整幅影像搜尋前方車輛，車輛偵測範圍太廣且耗時，為了減少搜尋範圍，參考車道偵測資訊來降低前方車輛偵測的時間，因此需設定前方車輛偵測 ROI，接著為使系統全天候運作，統計路面灰階值來作為車輛偵測的重要參考指標，再來針對不同天候狀況，利用影像辨識技術來擷取車輛特徵，並檢查車輛成對邊緣特性，最後利用高取樣週期下，車輛的時間與空間連續性來降低誤偵測的情形。



圖七 Single Mode 車輛偵測流程圖

(二) Continuous Mode

在 Single Mode 偵測到車輛後，即進入 Continuous Mode，其偵測流程見下圖八，由於系統保留前次車輛偵測資訊，為加快系統偵測時間，針對前次搜尋到影像中車輛位置附近設定 ROI，接著偵測 ROI 中車輛特徵並比對車輛垂直邊緣特性，然後利用 3D 座標轉換為影像座標的模型來估測前方車輛距離。

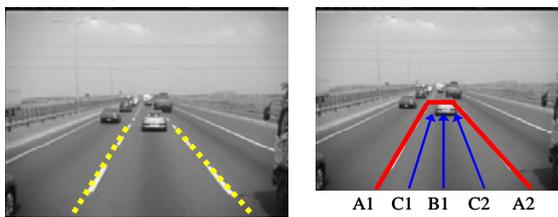


圖八 Continuous Mode 車輛偵測流程圖

以下將介紹 Single Mode 與 Continuous Mode 中的車輛偵測演算法。

(一)ROI 設定

在 Single Mode 中，由於沒有前次偵測資訊可供參考，因此為了增進系統效能，我們將車輛偵測範圍設定在前方車道範圍內，在偵測到車道線後，如下圖八(a)，在車道區域內，我們利用三條偵測線由畫面下往上搜尋車輛，見圖八(b)，其中 $B_1 = (A_1 + A_2)/2$ ， $C_1 = (A_1 + B_1)/2$ ， $C_2 = (B_1 + A_2)/2$ 。



(a)車道線偵測 (b)ROI 設定

圖九 車輛偵測 ROI 設定

由於採用二次曲線來近似車道線，但在遠端無窮處逼近時會有不穩定的擺動現象，此時若更新車輛偵測 ROI，容易造成誤偵測的情形，因此利用消失點位置來輔助車輛偵測 ROI 的更新，見圖十。 V_1 與 V_2 為影像消失點，當出現圖十(a)的狀況時，消

失點 V_1 偏離設定區域，若此時更新車輛偵測區域，很有可能偵測到隔壁車道的車輛，為了解決此問題，本演算法採用一濾波器來過濾車道線偵測結果，當消失點 V_1 偏離合理區域範圍時，如圖十(a)，則不更新車輛偵測 ROI 範圍；當消失點 V_2 在合理範圍內時，如圖十(b)，便更新車輛偵測的 ROI 範圍。



(a)車道線不穩定 (b)車道線穩定

圖十 車輛偵測 ROI 更新

在 Continuous Mode 中，已知先前車輛的偵測資訊，包括車輛底部位置、車輛寬度以及距離等，如圖十一(a)，在畫面取樣率達 30 frames/sec.時，車輛在相鄰畫面中的移動距離很小，利用此特性，以預測的方式來估計一下張畫面車輛位置的範圍，車輛偵測 ROI 設定如圖十一(b)所示。



(a)前次車輛偵測結果 (b)ROI 設定

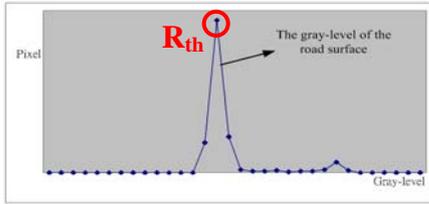
圖十一 Continuous Mode 車輛偵測 ROI

(三) 路面灰階值統計

為達成全天候車輛偵測系統，包含晴天、陰天、雨天和夜晚，其車輛與路面特徵均有明顯的不同，因此必需動態調整參數，來達到全天候偵測的目的。由於在下半張畫面中，路面佔大部份的比例，並且天候的特徵會表現在路面上，因此路面灰階值可作為本演算法重要的參考值。為了獲得路面灰階值，本演算法統計所偵測到車道線附近的區域，見圖十二(a)，針對灰階值作個數的統計，採用點數最多的灰階值 R_{th} 作為路面灰階值的代表，如圖十二(b)，利用 R_{th} 即可動態調整車輛偵測的臨界值，達到全天候車輛偵測。



(a)路面灰階值統計區域



(b)路面灰階值統計

圖十二 路面灰階值統計

(四) 車輛影像特徵

在不同天候下，車輛所產生的特徵也會有不同的變化。在晴天和陰天時，車輛偵測主要是將車輛底部的陰影作為主要特徵，而雨天影像則是有模糊化的特性，最後夜間車輛的主要特徵為車尾燈的亮塊區域。

在晴天和陰天的狀況下，車輛底部會有黑色陰影，而以一般狀況來說，此塊黑色陰影之明亮度會比由陽光斜射造成的陰影還暗，如圖十三(a)，圖十三(b)為原影像的水平 Sobel 邊緣特性。

本演算法所提出的自動臨界值設定方法，利用此臨界值可將原始影像轉為圖十三(c)的二值化影像，接著再針對圖中藍色區域對 v 軸作橫向投影，見式(14)。

$$V = (V_1, V_2, K, V_n) \quad (14)$$

$$= \left(\sum_{i=1}^m B(u_i, v_1), \sum_{i=1}^m B(u_i, v_2), K, \sum_{i=1}^m B(u_i, v_n) \right)$$

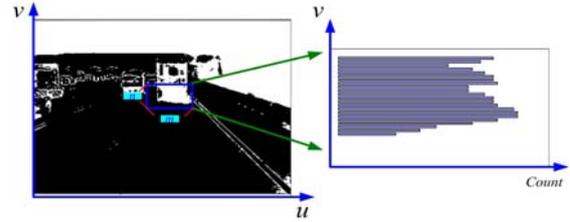
其中 $B(u_m, v_n)$ 為二值化影像的灰階值， (V_1, V_2, K, V_n) 為投影向量。投影的統計結果如圖十三(d)，接著找出 V 向量的最大值 $V_{\max} = \max\{V_i | 1 \leq i \leq n\}$ ，定出二分之一的 V_{\max} 為一車底陰影搜尋的臨界值 B_{th} ，由下往上搜尋，當符合 $V_i > B_{th}$ 時，即判斷 i 為真正車輛底部位置。以圖十三(d)為例，當偵測到連續水平陰影邊緣時，圖中綠線部份，但影像中實際車輛底部位置為藍線部份，藉由上述的修正機制，可正確偵測出影像中車輛底部位置。



(a)原始影像



(b)Sobel 水平邊緣



(c)陰影二值化影像



(d)車輛底部偵測



(e)車輛偵測結果

圖十三 晴天前方車輛偵測

在夜間的狀況下，本演算法主要會針對車尾燈為主要特徵擷取，圖十四(a)為夜間受隔壁車道車輛的車燈影響之前方車道影像，本演算法利用水平 Sobel 邊緣運算子作為前處理，圖十四(b)與圖十四(c)分別為水平 Sobel 邊緣遮罩與其邊緣二值化圖。在夜間車輛偵測演算法中，首先我們由影像下方往上搜尋，找到連續水平邊緣，如圖十四(d)，但此邊緣並非前方車輛所造成的，因此會造成誤偵測的情形，本演算法利用以下步驟來解決此誤偵測狀況：

步驟一：利用水平 Sobel 遮罩來產生水平 Sobel 邊緣二值化影像。

步驟二：由影像下往上搜尋連續水平邊緣，其連續長度大於二分之一路寬。

步驟三：偵測車輛尾燈。

步驟四：產生垂直邊緣二值化影像。

步驟五：將垂直邊緣二值化影像對 v 軸作投影，見圖十一(e)。

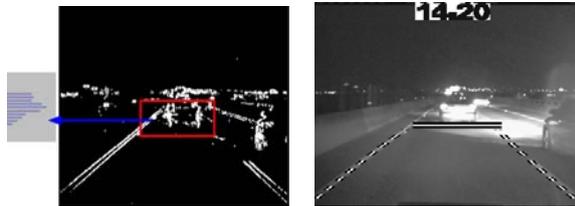
步驟六：利用車燈影響所產生較少的垂直邊緣特性來輔助判斷，達到修正正確車輛底部的目的，如圖十一(f)。



(a)原始影像 (b)水平 Sobel 遮罩



(c)Sobel 水平邊緣(d)誤偵測情形

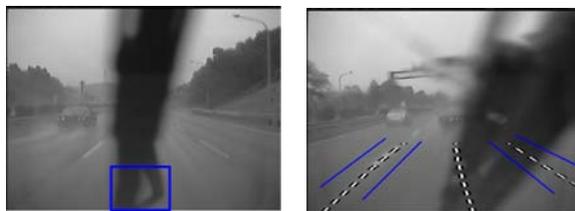


(e)Sobel 垂直邊緣投影 (f)修正情形

圖十四 夜間前方車輛偵測

在雨天狀況下，因為影像被模糊化的緣故，車輛特徵較不易擷取，因此需要相較於其他天候狀況的前處理。

首先在一般小客車的前擋風玻璃上，會有雨刷的影響，如圖十五(a)和十五(b)，這會使得偵測上有困難與錯誤情形發生，因此在這種情形下，輸入的影像必需由一濾波器來進行濾除的工作，本演算法在雨刷偵測的 ROI 為畫面中央以及車道線附近區域，如圖十五(c)與十五(d)，當偵測區域出現大量黑色灰階值，即將此張影像不作偵測，僅將上一張影像的偵測結果資訊顯示出來。

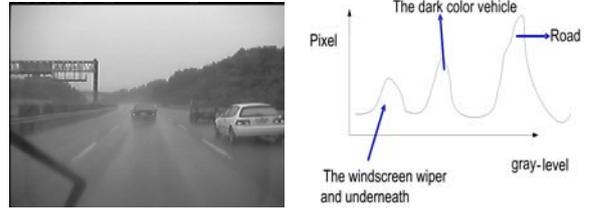


(a)影像中央 (b)車道線附近

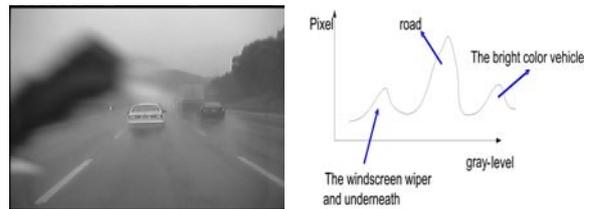
圖十五 雨刷偵測 ROI

接著介紹雨天車輛偵測演算法，包含偏暗與偏亮的車輛，在偏暗車輛的部份，如圖十六(a)，其畫面灰階值統計特性見圖十六(b)，利用路面灰階值統計方法，即可找出相較於路面暗的物體，同

理，在偏亮車輛的部份，如圖十六(c)，其畫面灰階值統計特性見圖十六(d)，車輛主要特徵為相較於路面較亮的部份，最後利用式(1)來得到車輛底部位置。



(a)暗車 (b)暗車影像灰階值統計



(c)亮車 (d)亮車影像灰階值統計

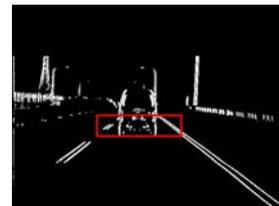
圖十六 雨天前方車輛偵測

(五) 車輛成對垂直邊緣

垂直成對邊緣是車輛偵測一項重要特徵，不論是貨車、轎車、暗車或亮車在大部份的天候狀況下，均會有成對邊緣的特徵，以圖十七(a)為例，圖十七(b)與十七(c)分別為垂直 Sobel 遮罩與垂直邊緣二值化圖，在各天候下，擷取車輛主要特徵後，可得到車輛位置，針對此位置，車輛成對邊緣特性會出現在圖十八(c)紅色區域中。



(a)原始影像 (b)垂直 Sobel 遮罩



(c)成對垂直邊緣區域

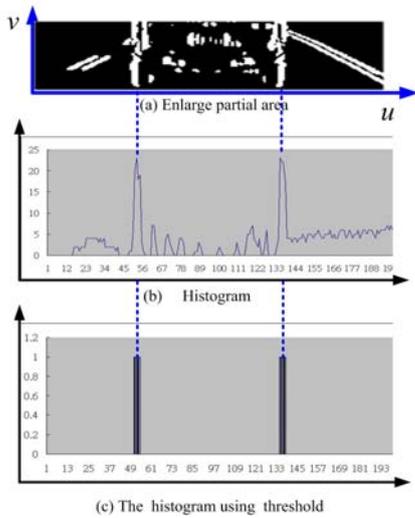
圖十七 車輛成對垂直邊緣

圖十八(a)是將圖十八(c)中紅色區域放大，並對u軸作投影，見式(15)。

$$U = (U_1, U_2, K, U_m)$$

$$= \left(\sum_{i=1}^n V(u_1, v_i), \sum_{i=1}^n V(u_2, v_i), K, \sum_{i=1}^n V(u_m, v_i) \right) \quad (15)$$

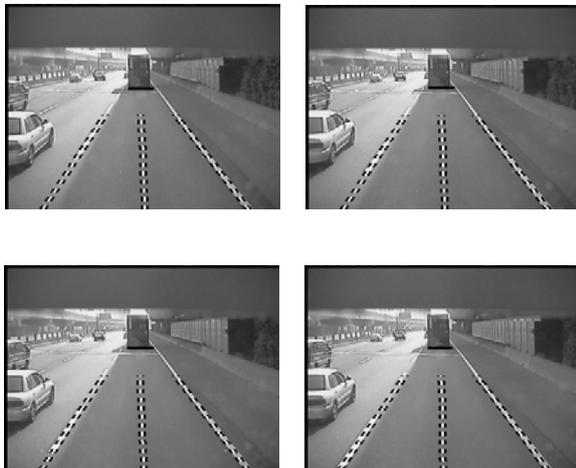
其中 $V(u_m, v_n)$ 和 (U_1, U_2, K, U_m) 分別為二值化垂直邊緣灰階值以及投影向量。其投影結果如圖十八(b)所示，最大向量 $U_{\max} = \max\{U_i | 1 \leq i \leq m\}$ ，取 U_{\max} 的一半為臨界值為主要垂直邊緣特徵，其擷取結果見圖十八(c)。



圖十八 車輛垂直邊緣特性分析

(六) 車輛在時間與空間的連續性

在 Single Mode 中偵測到車輛後，會記錄其位置，在短時間的取樣率中，車輛的移動範圍很小，圖十九為連續四張畫面的車輛移動情形，當偵測到一車輛物體時，在連續畫面中移動量很小，若移動量太大時，則認為是誤偵測的情形。



圖十九 連續前方車輛影像

(七) 車輛距離估測

利用 3D 與 2D 座標轉換推導公式，並將偵測到的車道線參數代入，可求得攝影機至前方車輛底部的距離 Y_{Mr} ，見式(16)。

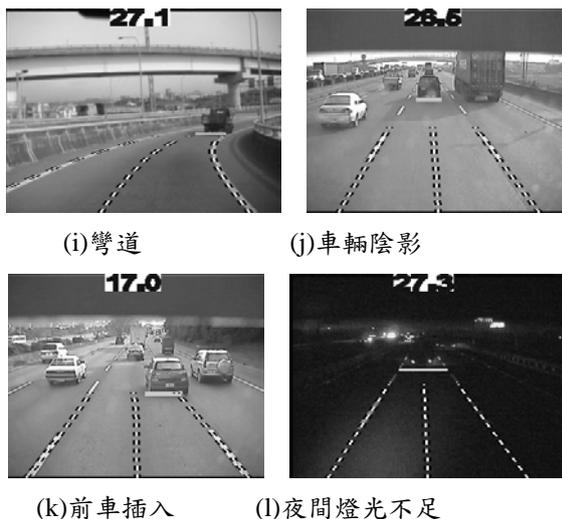
$$Y_{Mr} = e_u \frac{W}{u_R - u_L} \quad (16)$$

其中 e_u 為攝影機參數， W 為道路寬度， u_R 與 u_L 分別為前方車輛底部所在 v 座標的右車道線 u 座標與左車道線 u 座標。

(八) 實驗結果與分析

本系統之車輛偵測演算法可應用於全天候狀況，包括晴天、陰天、雨天和夜晚，本系統在小客車與大客車上均運作良好，並且在國道三號與 68 號快速道路做過數百公里的測試，圖十七為本系統在各種不同條件下的偵測結果，例如路面上的標字、橋樑連接縫、高架橋下的陰影以及夜間車燈的影響。





圖二十 全天候車輛偵測結果

四、結論與未來展望

本系統已於國道三號以及 68 號快速道路上做過上百公里的測試。其中車道線與車輛偵測在各種不同的天候條件下，運作良好。在輸入影像為 CIF，並實現最佳化於 DPS 平台時，本系統運作時間為 25 f.p.s 以上，並可在駕駛距離前車過近時，給予駕駛人警示，可避免意外的發生，保護駕駛者與用路人的安全。

五、參考文獻

- [1] M. Nashman and H. Schneiderman, "Real-time visual processing for autonomous driving," *Proceedings IEEE Intelligent Vehicle Symp.*, pp. 373-378, 1993.
- [2] C. Stiller, J. Hipp, C. Roessing and A. Ewald, "Multisensor obstacle detection and tracking," *Image and Vision Computing*, vol. 18, no. 5, pp. 389-396, April, 2000.
- [3] L. Giubolini, "A multistatic microwave radar sensor for short range anti-collisionwarning," *IEEE Trans. on Vehicular Technology*, vol. 49, no. 6, pp. 2270-2275, Nov., 2000.
- [4] A. Giachetti, M. Campani, and V. Torre, "The use of optical flow for road navigation," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 34-48, Feb. 1998.
- [5] A. Giachetti, M. Campani, and V. Torre, "The Use of Optical Flow for Road Navigation," *IEEE Trans. Robotics and Automation*, vol. 14, pp. 34-48, Feb. 1998.
- [6] R. Alix, F. Le Coat, and D. Aubert, "Flat World Homography for Non-Flat World On-Road Obstacle Detection," in *Proc. IEEE, on Intelligent Vehicles Symposium*, pp. 310-315, 9-11 June 2003.
- [7] R. Okada, Y. Taniguchi, K. Furukawa, and K. Onoguchi, "Obstacle Detection Using Projective Invariant and Vanishing Lines," in *Proc. IEEE, on Computer Vision*, vol. 1, pp. 330-337, 2003.
- [8] R. Okada and K. Onoguchi, "Obstacle Detection Based on Motion Constraint of Virtual Planes," in *Proc. IEEE/RSJ, on Intelligent Robots and System*, vol. 1, pp. 61-66, 30 Sept.-5 Oct. 2002.
- [9] C. Demonceaux, A. Potelle, and D. Kachi-Akkouche, "Obstacle Detection in a Road Scene Based on Motion Analysis," *IEEE Trans. Vehicular Technology*, vol. 53, pp. 1649-1656, Nov. 2004.
- [10] H. Mallot, H. Bulthoff, J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical flow computation and obstacle detection," *Biol. Cybern.*, vol. 64, no. 3, pp. 177-185, 1991.