

路邊車輛車牌 快速辨識

郭英哲、劉品佑、林智宣、顏子翔 國立勤益科技大學電機工程系

kuoyc@ncut.edu.tw

摘要

藉由警察在執行巡邏勤務時,攜帶嵌入式 車牌辨識裝置,針對停放於路邊之車輛進行 車牌自動辨識,並連結警務無線網路進行贓 車查察,可快速追查贓車,有效嚇阻車輛失 竊。本研究所提出的車牌辨識系統,包括車 牌定位、車牌傾斜與旋轉校正、字元分割及 類神經網路辨識。在車牌定位部份,首先將 視訊影像進行灰階可減少後續程式運算量, 接續使用垂直邊緣偵測、閥值轉換、濾波、 連通物件法找出車牌位置。取出車牌影像後 ,進行傾斜校正使車牌邊緣與X軸平行,接 著進行旋轉校正,使扭曲車牌四個邊緣呈現 90度,並判斷車牌中區域黑像素比例,將字 元切割出來。最後,將字元丟入先前就訓練 過的反向傳遞式神經網路架構(Back Propagation Neural Network) 辨識出車牌號 碼。

一、簡介及系統開發環境建立

隨著汽車數量成長迅速,針對交通違規 及汽車失竊等問題也隨著增加。在都會區 ,警察常使用機車行駛於大街小巷執行巡 羅仟務, 這樣的巡邏仟務可涵蓋大部分都 會區域。藉由警察執行巡邏任務時,配備 車牌辨識系統針對停放於巡邏路線的車輛 即時進行車牌辨識,並透過警務無線網路 系統確認是否為失竊車輛,以達到即時查 察嚇阴車輛失竊。

此車牌辨識系統使用ODROID-XU4 嵌入式 微電腦平台(圖一),外接一個攝影機(圖二) 。同時,安裝Linux作業系統,以C++語言搭 配OpenCV電腦視覺資料庫實現車牌辨識影 像處理演算法則。



硬體架構		Samsung Exynos5422 Cortex TM -A15 2Ghz and Cortex TM -A7 Octa core CPUs \ Mali-T628 MP6 \ 2Gbyte LPDDR3 RAM	
作業系統		Linux Kernel 4.9 LTS	
介	面	HDMI 'Gigabit Ethernet port' eMMC5.0 '2 x USB 3.0 Host, 1 x USB 2.0 Host	
尺	寸	83 x 58 x 20 mm	

-、ODROID-XU4嵌入式微電腦平台



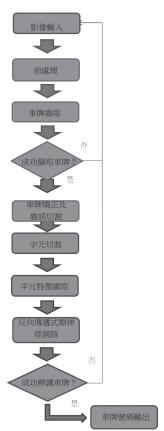


	Windows® 7 \ Mac OS 10.6 \ Chrome			
小小儿而小	OS™、Android v 5.0或更新版本。			
硬體介面	體介面 USB 連接埠、網際網路連線。			
	最高可達1500萬畫素、Full HD 1080p			
+日+欠+土 45	視訊通話、H.264 視訊壓縮功能、內建			
規格特色	兩個具自動降低雜音功能的立體聲麥克			
	風、光源不足自動調整。			

圖二、Logitech C920 HD Pro 網路攝影機

二、車牌辨識影像處理演算法

架設於警用機車之攝影機以車道線方向朝 右約60度,擷取道路右邊路邊停放車輛的影 像,先判斷影像中是否有車牌影像,然後再進 行後續的傾斜旋轉校正、字元擷取、車牌辨識 。若非車牌輸入影像,則拋棄輸入影像,繼續 進行視訊串流。最後於LCD液晶顯示器顯示辨 識後的車牌號碼。處理流程(圖三)。



圖三、車牌辨識處理流程

1.前處理

前處理為先將影像灰階化,再利用Sobel 處理、影像模糊、二值化等步驟。

(1) 灰階化

經過攝影機攝影後,輸入的影像格式 為RGB(圖四(a)),為了方便後續擷取車牌的 處理,於是轉成只剩亮度的灰階影像(圖四 (b)) °

實際上人眼對綠色的亮度感最大,而對藍 色最小,於是由下面公式得到比較適合人類 眼睛的灰階影像,久而久之已成為影像處理 界所用來彩色轉灰階的標準。



圖四(a)、原始彩色影像



圖四(b)、灰階處理結果

(2) Sobel 邊緣強化處理

Sobel是一種獲得影像梯度的手法,常用 於邊緣檢測,分別為Gx和Gy模板。Gx用來 檢測垂直邊緣,Gy用來檢查水平邊緣,通常 會分別對影像進行水平和垂直模板的運算, 得到像素的梯度。梯度為一個有距離和方向 的二維向量,距離表示變化幅度,方向表示 強度變化最大的方向。

科投新知

經過實驗後,發現車牌的垂直強度較為強 烈,所以使用Gx模板來偵測車牌,如此可以 消除更多不必要的雜訊,提升擷取車牌的成功 率,Sobel結果如圖4,車牌附近會較為明顯。

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \qquad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$

圖五(a) \ Sobel Gx \ Gy模板



圖五(b)、Sobel邊緣強化處理結果

(3) 二值化

二值化是將影像依照某個標準值區分為黑 與白,此標準值也稱作閥值(threshold),當 灰階影像的像素值大於閥值則變為灰度極大 值(白色);反之則變為灰度極小值(黑色)。 閥值的算法主要分為固定閥值和自適應閥值 (OTSU)兩類。此研究採用固定閥值。



圖六、 二值化處理結果

(4) 形態學運算

開運算、閉運算為形態學中的兩種運算方 式。開運算為先侵蝕 (erode)後膨脹 (dilate) ,可在纖細點處將物體分離,且可平滑較大

物體的輪廓;閉運算為先膨脹後侵蝕,於影 像中填充物體內細小空洞,連接鄰近物體, 平滑邊界且不明顯改變其面積。如圖七(a)。

侵蝕是一種消除邊界點,使邊界向內部收 縮的過程,可以用來消除小而無意義的物體 ;膨脹是將與物體接觸的背景點合併到該物 體中·使邊界向外擴張的過程,可以用來填 補物體中的空洞。(圖七(b))。

本專題先利用閉運算使二值影像中車牌的 區域補滿其內部細小的空洞,再利用開運算 使二值影像中細小的雜訊消除,以便後續於 物件連通法的處理。





圖七(b)、開運算結果

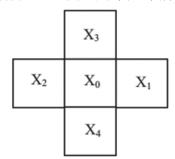
2.車牌擷取

(1) 物件連通法

影像經過二值化後,影像中會出現大小不 一的區塊,而只有一塊是欲得的車牌區域, 我們利用物件連通法的方式使得相鄰的區塊 可以視為同一個物件且給予一個唯一的標籤 符號。物件連通法可分為兩種類型:4-連通 (圖八(a))、8-連通(圖八(b)), 我們使用4-連



通,這樣可以避免雜訊與車牌誤連,造成判 斷錯誤,接著進一步做區塊分析以找出車牌 的位置,分析方法是將車牌從中間做水平掃 描,如圖九的黃線,將像素黑白變化的次數 相加起來,以變換次數最少的1為例子,如 果變換次數小於12次,則不是車牌。





圖八(a)、4-連通

X ₈	X ₃	X5
X_2	X_0	X_1
X ₇	X ₄	X ₆

圖八(b)、8-連通



圖九、 黑白變換示意圖

3.車牌矯正

(1) 車牌水平矯正

當我們找到車牌的位置,此時車牌有可能 不是水平的(圖十(a))。為了切割順利,我們 利用物件連通法,利用長寬比和密度比,將 字元的區塊留下,接著將這些字元的區域利 用斜矩形框起來,此時就能找出車牌的傾斜 角度。再利用仿射變換(Affine Transformation) 來矯正傾斜的車牌,矯正後 的車牌(圖十(b)),與X軸呈現水平。



圖十(a)、傾斜的車牌



圖十(b)、水平處理後的車牌

仿射變換是一種二維座標間的線性變換, 保持二維圖形的平直性(即變換後直線依然 為直線不會彎曲,圓弧還是圓弧)和平行性(即變換後二維圖形間的相對關係不變,平行 線依然平行,相交兩線的交角不變),來實 現平移、縮放、翻轉、旋轉。

(2) 車牌上下切割

當車牌水平矯正後,為了減少後續的運算 量,這時可以先切割上下雜訊,此時會發現 有些車牌上下會有螺絲及中文(圖十一(a))。 為了分開這時將車牌進行水平直方圖統計(圖十一(b))後可以發現車牌上下的邊緣,接 著將車牌切割,結果(圖十一(c)),會只剩下 字元的高度。

科技新知

圖十一(a)、 車牌號碼切割前



圖十一(b)、水平直方圖統計

圖十一(a)、切割完成的車牌號碼

(3)車牌傾斜矯正

由於攝影機是以車輛行駛方向偏右方60 度架設擷取影像,會造成車牌實際距離與投 影出來的距離會有差距。所以經過水平矯正 後會造成車牌傾斜(圖十二(a)),影響後續切 割及特徵提取。所以我們利用斜矩形找出傾 斜的偏移量,再用透視變換(Prespective Transform)來矯正傾斜的車牌,矯正後的車 牌(圖十二(b))。



圖十二(a) 、車牌傾斜矯正前

圖十二(b) 、車牌傾斜矯正後

(4) 字元切割

接下來使用傾斜矯正後的車牌做垂直直方 圖統計,由(圖十三(a))可看出在垂直直方圖 中字元之間不存在黑像素,以及可藉由統計 各區域的黑像素比例判別,若黑像素過多或 過少,則可判別各區域是否為雜訊,一般字 元黑像素比例約0.25~0.75,最後即可成功切 割出字元(圖十三(b))。



圖十三(a)、垂直直方圖

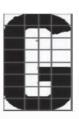
6280LD

圖十三(b) 車牌號碼分割結果

(5) 字元的特徵值擷取

我們將正規化後的字元分成4*8的區塊(圖 十四),並分別計算每格區塊裡的黑像素數 目,再除上此字元總共的黑像素,作為各區 塊的特徵值,供給類神經網路做字元的訓練 與預測。

區塊的分割方法為左右兩排寬度為總寬度 的0.3倍,中間兩排寬度為總寬度的0.2倍, 長度則平分為八份。



圖十四、字元特徵值

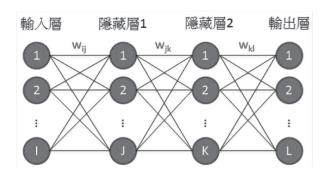


三、車牌號碼辨識

類神經網路 (artificial neural network, ANN)是一種模擬人類神經網路運 作的數學模型,神經網路的數值傳遞是藉由 各個神經元間連接的權重值與誤差值來決定 輸出,神經網路架構以外界給予的資訊來改 變內部結構,可説是一種自適應的系統。

多層感知器類神經網路架構(Multilayer Perception, MLP), 屬於前向網路, 一組輸入 向量對應一組輸出向量。MLP是由多個節點 層所組成,除輸入節點外,每個神經節點中 都帶有一個非線性的激勵函數,激勵函數可 為任意函數,但必須是可微(differentiable)。

倒傳遞法以錯誤更正學習法則來訓練網路 的鍵結值,藉由錯誤鍵結值的修正,以使網 路的輸出值趨向於期望輸出值,也可以視為 是最小均方法 (LMS 演算法) 的一種推廣,(圖十五)為倒傳遞法類神經網路架構示意圖。



圖十五、反向傳遞式類神經網路架構

1.倒傳遞式多層感知器網路-學習階段

於學習初期,給於網路各個節點間的權重 值是產生範圍介於0~0.1的亂數。為解決傳 統迴歸中,特徵值大小影響分類結果的問題 ,在MLP神經節點中所使用的激勵函式為 symmetrical sigmoid function,也稱作羅吉 斯函數(logistic function),如方程式(1)。 其作用會使輸出值漸漸收斂至-1~+1之間。

$$f(x) = \beta * (1 - e^{-\alpha x})/(1 + e^{-\alpha x})$$
 (1)

輸入層有32個神經元,即為字元特徵取 出的特徵值。隱藏層有兩層,且每層有33個 神經元,33為輸出入層神經元的總和除二。 輸出層有34個神經元,最後,學習階段所得 出各個節點間的權重輸出值(圖十六)。

```
<weights>
< >
  5.0018196671624671e-01 -9.5518678190171225e-01
  6.5695457584327344e-01 1.7159580552067331e+00
   1.1000204316048323e+00 1.4724691904468215e-01
  1.7857218339829921e+00 1.5955361573867179e+00
   -6.5179831339413710e+00 -2.9414680454448078e-01
   -2.8185192815585580e+00 1.0010802061245323e+00
   -2.5900006583395060e-03 4.6637873981651062e-01
  2.4963661044143990e+00 -1.2132317664712235e+00
   2.8182815957009866e+00 -1.4772015693172278e-02
  1.5195334540852126e+00 -1.4426958346770677e+00
  4.5802417506576799e-02 4.3692065211132514e+00
   -3.2393116828558552e+00 1.8263582346318689e-02
```

2.4886860539894027e+00 -3.3008141532327046e-01

圖十六、 訓練後各節點間的權重值

2.倒傳遞式多層感知器網路-預測階段

輸入特徵給學習好的多層感知器網路架構 後,會得到一組對應的輸出(預測的字元), 隨即將預測結果顯示於終端機上,且在 Linux中創建一個新的資料夾放入辨識結果 與圖片。

四、實驗結果

在一般的嵌入式微電腦開發版上若需要處 理影像串流,則會花費過多的時間造成程式 效率不佳,且容易有過熱的現象,通常可能 是處理器核心任務分配不佳,或是RAM過少 ,抑或是缺乏圖形處理單元。

在此使用Linux系統中可指定特定核心執 行任務的方法,可使四核心或八核心處理器



同時執行指定任務,分攤執行任務,故使 用Util-Linux套件中的taskset指令來加速影像 處理程式,大部分的 Linux 發行版預設都會 安裝這個套件。

圖像加速有很多方法,例如:CUDA、AMD Stream,各個廠商都不支援其他廠商的圖像 加速應用,因此OpenCL就是可為各個廠商 的產品進行平行化運算的API。OPENCL有兩 種平行化模式,包括 task parallel 以及 data parallel。所謂的 data parallel,指的是有大 量的資料,都進行同樣的處理,而影像處理 上常常使用,例如,遮罩運算,經常要對一 個影像的每個 pixel 進行同樣的動作(例如 Gaussian blur),這類工作很適合 data parallel 的平行處理模式。

影像處理程式透過 OpenCL API 取得顯示 晶片(GPU)的相關資料,並希望在GPU上執 行影像處理程式(使用 OpenCL 程式語言撰 寫),藉GPU單元的影像處理能力,達到圖 像加速的功用。

(圖十七)為實驗結果,由圖片中紅框可知其 輸出結果與輸入影像中的車牌號碼皆相同。



圖十七、車牌辨識結果

