**讀書計畫**

**即將完成中學六年的學習階段，來到大學鑽研自己喜歡的學問，讓人感到興奮且期待。中學階段，很多的學習都是為了考試。相對的，大學階段應該會有更彈性且自由的學習環境。為了讓自己在未來四年不要虛度光陰，我相信做好事先的規劃是很重要的。**

**對於資訊工程學系的專業，我有初淺的認知：電腦硬體、電腦軟體、及各種因電腦而衍生的技術，都是屬於資訊工程系研究的範圍。為了學好相關技術，基礎知識的建構是當務之急。電腦硬體的基礎學科，包括數位系統設計，計算機組織等，我相信非常值得花心力把它學好。這些科目不僅是未來硬體工程師的必備工具，對於軟體工程師而言，瞭解軟硬體介面也是提升軟體品質的有效方法。在軟體課程方面，物件導向程式語言、資料結構、演算法等則是基礎工具，絕對不能輕忽。**

**好的數學能力對於資訊工程師及科學家來說也是必備工具，因此除了高中時期就已經接觸過的微積分之外，線性代數、離散數學、機率統計、甚至工程數學中的傅立葉變換、小波變換等，都已證明在許多資訊技術中扮演舉足輕重的角色，因此我也會多花些時間在上面。**

**到了三、四年級，我想花時間在專題製作及準備研究所或留學考試。由於目前我所知有限，較難具體規劃出屆時專題製作的題目與方向。但是從最近的時事可窺知，機器人、大數據、及物聯網是目前熱門的議題。而我個人則是對於機器人情有獨鍾。台灣的交通秩序相較於許多先進國家，真的不是太好。如果未來有一天能讓自動駕駛機器人主導交通，秩序應該能大幅改善。這也是我的夢想！參與這方面的研究，是令人興奮的！**

**除了專業科目之外，博雅的通識課程是使得大學精神得以彰顯的重要因素之一。二十一世紀的人才絕對不應只侷限在自己的專業領域之中，跨領域整合的能力日益重要。台大（或清大、交大，改成推甄的學校）令我嚮往的因素之一，就是在專業之外，也提供豐富完善的通識課程，因此我將把握這個機會，除了讓自己成為資訊專業人才之外，也要具備跨領域知識整合的能力。**

**為了與國際接軌，好的外語能力已是基本。英語是我從幼兒時期就已經接觸的語言，未來會持續精進自不在話下。但是除了英語之外，具備第二甚至第三外國語將使得自己更具國際競爭力。因此，修習第二、三外國語也是我大學時期的規劃之一。**

**最後，我以自動駕駛車技術中重要的一環：「道路標線偵測」做為主題，呈現我最近在資訊技術中研究的成果。**

**專題研究：道路標線偵測的研究**

1. **研究動機：交通事故常造成人員的傷亡。而交通事故發生的原因，「駕駛行為不當」又居首位。因此，若能開發無人自動駕駛車，且讓無人駕駛車主導大部分的交通，勢必可以減少交通事故的發生，進而降低無謂的人員傷亡。**
2. **研究範圍：為了實現無人駕駛車，需要許多技術整合。例如：自動控制、號誌辨識、路徑規劃、定位等不勝枚舉。而我認為其中一項重要技術，就是如何讓車子認得道路的範圍，並安全地行駛在道路上。這其中的首要工作，就是如何讓車子認得道路的標線。**

**在此研究中，將限縮討論的範圍如下：**

* **車子行走於高速公路，車前方之行車記錄器記錄下之道路狀況做為研究對象。**
* **目的是利用較簡單的影像處理技術，找出道路邊緣的標線。**
* **先僅限於筆直道路的部分。至於彎曲轉彎的道路，將留待後續研究。**
* **找出道路標線之後，將用紅色線標示其邊線，以利後續自動控制時使用。**

1. **研究方法：針對使用的程式語言、資訊技術等說明如下：**

* **程式語言的選擇： 在此研究中，我使用Python程式語言。由於Python是一種互動式的程式語言，且提供完善的函式庫，對於高中生而言是很親和的程式語言。另外，此研究的目的主要在於瞭解資訊技術解決問題的程序，並不是產品開發，因此不注重軟體效能，Python語言已足夠。**
* **動態影像轉靜態影像： 行車記錄器的錄像，可分解成多張靜態影像。先從分析靜態影像開始，決定使用的方法，測試，等一切滿意之後，再將這些方法應用於動態的錄像檔。**
* **彩色轉灰階： 錄像為彩色。初步認為色彩的資訊暫時可不考慮，因此將彩色影像轉換成灰階影像。**
* **影像裁切： 由於行車記錄器的鏡頭是廣角鏡，車道左右兩條標線將匯聚於遠方。因此在匯聚點之上的影像基本上已不重要。影像裁切有助於減少多餘資訊，有利於後續處理。**
* **邊緣偵測：道路標線通常與柏油或水泥路面的顏色有強烈對比。利用此特性，可以很容易找出標線的邊緣。一般常用的邊緣偵測演算法，例如：Canny edge detection，在Python的opencv函式庫中都有提供。**
* **Hough transform: 在「邊緣偵測」所找到的結果，視覺上看起來即使是一條線，但是在資訊表示上仍然是點陣圖。因此，為了進一步獲得「直線」的資訊，要進一步進行Hough transform。Hough transform的結果，會給我們一堆直線段的訊息，每條直線段以兩個端點來表示。有了這些端點，即表示已經找到了標線。**

1. **研究步驟：**

**首先我們看一下原始從行車記錄器獲得的影像：**

****

**現在車子是走在最外側車道，可以看得出來車道相當筆直。**

**將彩色影像轉換成灰階影像：**

****

**為了適當地去除雜訊，我們可以將灰階影像做模糊化，得到下列影像：**

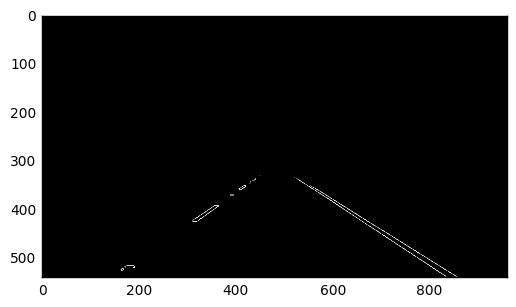
****

**模糊化之後的結果，用肉眼看起來差異並不大。**

**接下來，我們要開始進行邊緣偵測了，用的方法是opencv裡面的Canny edge detection。**

****

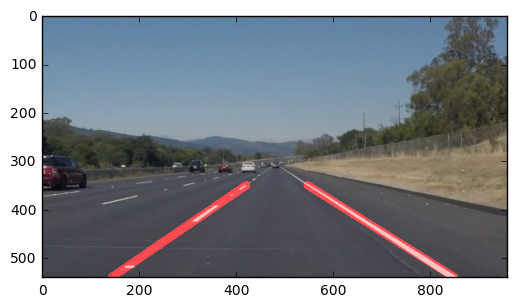
**從起結果看得出來，道路的確被偵測到了。但是後面的風景造成我們的困擾。如果能適當地裁切，會有利於後續工作。裁切的方法是這樣的：觀察到影像大小寬x高為960x540。切割一個梯形，將可涵蓋車道兩邊標線。根據影像大小，設計一梯形，其四個端點為（96,540）, (460,324), (500,324), (912,540)。這邊特別說明一下，在影像處理的領域中，座標的定義與解析幾何不同。座標原點定在影像的左上角，x座標往右越大，y座標往下越大。裁切結果如下圖。**



**從這張圖很明顯可以發現，背後雜訊都不見了。剩下的只是車道線而已。**

**然而，我們的工作還沒完成。目前看到圖中的線，其實仍然是由一堆點所構成的。這樣的資訊對於後續使用仍然不足。因此需要進一步做Line detection。在影像處理及電腦視覺的領域中，提到其中一個方法叫做Hough Transform，而我們即是要用此方法。**

**經過Hough Transform之後，得到一堆線段。線段由兩個端點來表示。若將端點連成一線畫在原圖上，會得到下圖。**

****

**至此，已成功地偵測到道路兩邊的標線了。**

1. **結論與研究心得：**

**以上研究，是我在高三這一年，有機會在大學老師的指導之下所完成的一個小專題。在這過程中，我接觸到程式語言。使用程式語言方便的函式庫使得寫程式並不像想像中那麼困難。**

**另外，標線偵測這個原本也以為會很困難的議題，在抽絲剝繭之下，竟也迎刃而解。**

**相信在未來學習更多相關知識之後，可以解決更複雜的問題。**