

第 1 章 影像辨識簡介

1-1 我認為的影像辨識是…

影像辨識最簡單的定義就是：「從影像中找到我們需要的資訊」！由於多數動物都很依賴視覺提供周遭環境的資訊，經過幾億年的演化，應該都已「內建」有非常智慧高效率的影像辨識「演算法」，隨時都在我們的大腦中運作！目前所有的影像辨識專家都還無法完全理解這些演算法的精確邏輯，如果可以！我們就能按照藍圖寫出跟人類的視覺能力完全一樣的影像辨識軟體了！

在個人的理念上，影像辨識應該是對於人腦視覺演算法的一個逆向工程！事實上較早期的影像科學研究也多以人眼及大腦為師，很多有關顏色亮度的數學模式都以人眼的感知程度為依據，譬如明度、彩度與亮度等等。但是當我們開始使用電子元件取得數位化影像之後，RGB 三原色就變成資料型態的主流，也讓影像辨識的研究慢慢偏離以人腦視覺為師的方向，所有實作方法都受到數位資料形態的引導與限制了！

我認為，即使人類在近幾百年的科學研究上有很好的成就，可以運用很多數學方法，做出類似人類視覺判斷的影像辨識結果。但是面對來自自然環境中的影像，我不覺得以純數學角度研究所得的影像辨識演算法，其辨識能力與效率，可以超越數億年演化形成的生物視覺演算法！本公司研發的影像辨識產品，辨識能力與速度都獲得好評，基本上都是來自模仿人的視覺認知過程所得到的成果。

但與此同時，目前影像辨識的主流技術方向已偏向機器學習與深度學習，就是相信以數學或統計方式可以製作出等同，或超越人眼視覺能力的影像辨識軟體！其理念就是相信：從人類數百年來發明的數學方法組合之中，應該可以讓電腦自行學習並「演化」出超越數億年生物演化出來的視覺演算法！

目前機器學習或深度學習宣稱的目標尚未達成，個人認為：放棄對於基礎科學的研究，寄望電腦的自行演化可以達到(或超越)人類智慧的程度，是不切實際的想法。所以包括本書在內，我的所有研發產品都未曾使用過任何機器學習或深度學習的方法與概念。對於不知而行的統計學模式的研究方式，我們會盡量避免使用。

我想證明的是：基於模仿人眼視覺認知過程設計演算法，就可以完成大多數影像辨識應用軟體的研發製作。根據已知事實建立出明確的演算法，其實才是傳統科學的精髓。事實上也一定會比寄望於以嘗試錯誤為基礎的機器學習或深度學習，更快達到高辨識率，讓影像辨識普及於各個事業領域的目標！

本書雖然提供非常完整的程式範例，但請不要誤以為可以直接套用到其他問題或類似影像上，就能「無入而不自得」！我們製作的高辨識的商業化軟體規模都大於這些範例十倍以上！因為真實的影像辨識例外狀況極多，那些由「經驗值」累積產生的程式碼並不困難，但是需要你能充分理解每一個辨識的步驟，分析每個案例錯誤的原因，才能

見招拆招建構出高辨識率的軟體。所以每一範例與步驟「為何要如此做？」的理解比「怎麼實作？」更為重要！如果你只是想找影像辨識的「公式」或「模組」？就不必購買本書了！你應該會失望的。

總之，本書完全不會有機器學習或深度學習的相關內容，也不會使用 OpenCV 或 Yolo 等流行的影像辨識工具模組。只會包含傳統的影像辨識方法實作，以及模擬視覺過程設計的一些自有演算法。只要有 Windows 作業系統的電腦，下載免費的 Visual Studio 軟體，有基礎的 VB 程式設計能力，具有高中程度的數學能力，就可以輕鬆閱讀本書，學會實作影像辨識。

1-2 影像辨識的過程—以車牌辨識為例

影像辨識就是從影像中獲得我們所需要的資訊，如下照片中，有車子、浪板、收費機、與斑駁的路面。但我們想辨識出來的，只是其中的一個車牌號碼！如何用數學方法寫出程式逐步完成資料擷取的工作，最終得到「EZ-9528」這個文字模式的答案，就是影像辨識必須完成的工作。



EZ-9528 這個字串之所以對我們有意義，是因為我們腦中已經有這些字元的模型，與其所代表的文字意義。我們其實是以大腦中既有的認知，去比對影像中有沒有符合這些資訊特徵的目標？如果人腦判斷的答案符合電腦處理的結果，就是「辨識成功」了！



摘自：http://portal.ptivs.ptc.edu.tw/sys/lib/read_attach.php?id=1039

目前絕大多數的車牌辨識研究都還是以「**車牌**」作為第一階段的搜尋目標，如上圖就是嘗試在複雜背景中使用車牌的高強度對比，以及車牌是近似「**水平矩形**」的特徵找到車牌位置。但是隨著影像畫素越來越高，背景越來越複雜，加上車牌從側面或高角度拍攝時呈現的形狀**並不是水平矩形**，此類演算法已日漸落伍，傾斜超過 10 度的車牌就無法辨識，很難被百萬畫素時代的新使用者接受。

事實上，這種「**找車牌**」的演算法並非典型的 **OCR**(Optical Character Recognition，光學字元辨識)流程，是受限於早期影像品質與電腦效能較差，迫不得已的選擇。只有這樣才能即時快速找到車牌，速度太慢的車牌辨識演算法是沒有實用價值的！但隨著數位影像品質與電腦效能的大幅躍進，直接以標準 **OCR** 流程辨識車牌**字元**的演算法，已經可以實用於商業產品了！

本公司的各種車牌辨識產品核心就是以此為基礎！你可以預期：本書內容將與你之前看過的所有車牌辨識演算法公開文獻都明顯不同！基本上就可以克服辨識傾斜車牌的這個大難題，傾斜二三十度應該都沒問題，而且演算速度並不會因此變得太慢，保持高效率的演算效能也是本書內容的重點之一。

1-3 簡化影像

我們要找的車牌目標只是原始影像中的一小部分，任何生物的眼睛，包括我們現在使用的攝影機，在取得影像之前都不確定目標在哪裡？甚至沒有預期要看甚麼東西！所以都是以自身俱備的感光能力盡量收集最多的資訊。因此，原始影像通常都是**資訊過量**的狀態，我們作影像辨識最困難的問題，其實就是如何簡化影像資訊？

人眼通常都能很快在視野中找到我們想找的目標，但是誰都說不清楚詳細的過程，為什麼影像中的東西那麼多，我們卻馬上就能鎖定那個車牌？這就是影像辨識需要研究的重點了！我們必須設計出**簡化資訊，聚焦於目標**的演算法，還要減少運算量，讓它們很快完成，如果電腦找到車牌的時間比人的反應還慢，影像辨識產品的價值就不高了！

在此，我們首先會從 **RGB** 三原色中選取一種對於辨識預期目標最有利的灰階亮度，譬如自然光中綠光對於整體亮度的代表性最高，所以一般車牌辨識可以直接選擇綠光，拋棄紅與藍光來做辨識，也可以計算出 **RGB** 的整體亮度，總之就是將 **RGB** 三個

陣列，簡化成一個陣列作為後續運算的基礎。

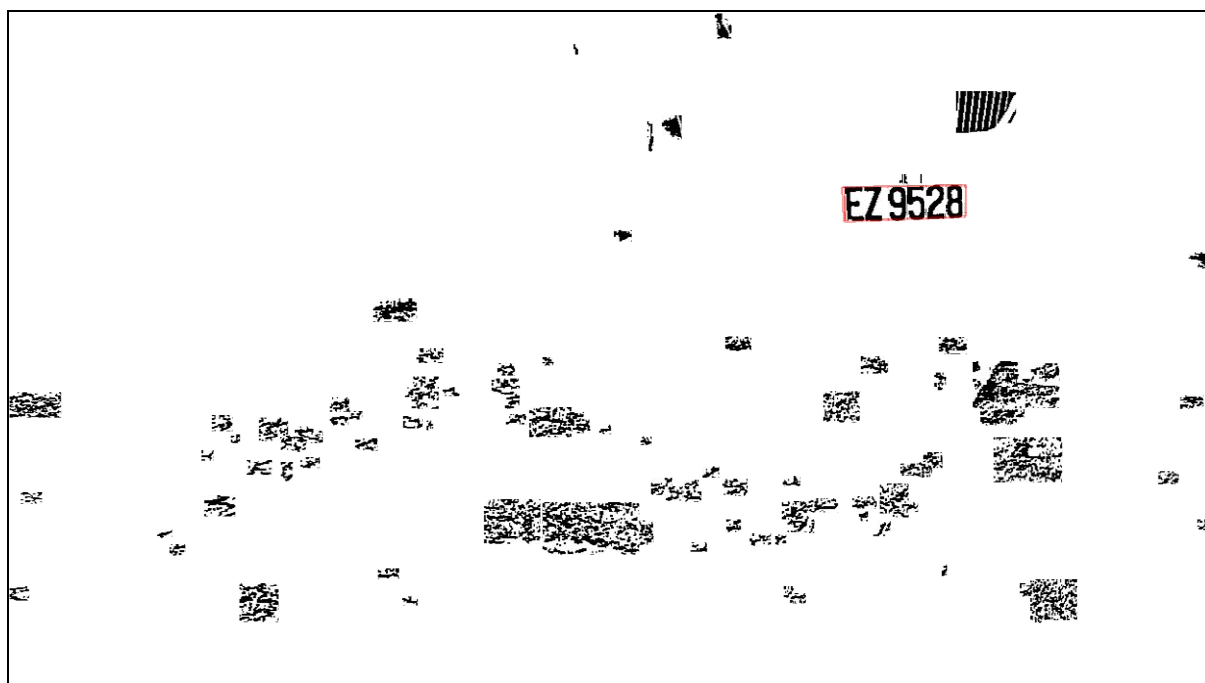


因為在此例中我們要找的是字元，特性是獨立塊狀的目標，所以接下來是進一步簡化影像成為黑白圖，希望要找的字元可以被獨立切割出來，字元是全黑的，字元背景是全白的，這就是所謂的二值化過程。



再來是畫出它們的輪廓線，以驗證這些黑色塊狀目標的大小形狀像不像我們要找的字元目標？就是用演算法找出所有封閉曲線，曲線範圍太大或太小的就忽略掉的意思。

這好像很複雜？但不必擔心，只要有適當的演算法，就可以像吹掉灰塵一樣，很快清理出可能的目標。



在此必須強調的是：簡化資訊的過程不是可以輕易封裝固定的！每一個簡化程序都會影響到能否成功辨識到我們需要的目標，必須有針對性的設計，譬如用綠光做辨識時，會造成綠色車牌的工程車辨識較差，綠光用在地磅站的車牌辨識系統就會變成劣質產品，改成紅光辨識就會特別靈敏準確，但是碰到紅字的計程車又看不清楚了！

多數影像辨識初學者都很害怕這一階段涉及的複雜條件與數學邏輯，寄望有神奇的黑盒子可以直接提供篩選後的合格目標，也因此會有 **YOLO** 之類的演算模組興起，但是如果想做出針對特定目標最佳化的辨識產品，就絕對不能依賴這類演算模組！其實不會有任何模組可以提供適合所有辨識目的的資訊簡化結果。

想像一下，我們用眼睛找東西時不是也會有「**預設立場**」嗎？因為這樣才會有效率！找計程車時，你會注意到眼前黑色的賓士車嗎？一定是對黃色的大目標才有反應！找不同的目標當然會有不同的**過濾器**，即使你用 **YOLO** 這類工具模組，也必須正確且精確的操控它，才會有好的效果，任何影像辨識的研發應該都是針對目標設計，沒有捷徑的！

無可避免的，任何簡化影像擷取特徵的演算法，包括本書使用的方法，其實都會減損事實真相。重點是在這些簡化資訊擷取特徵的過程中，我們是不是始終能充分掌握我們最終的辨識目的？不讓這些過程減損我們最終辨識成功需要的關鍵資訊？這是**物理**的問題，不是**數學**的問題，**數學方法**始終不應該凌駕**物理事實**，這是本書的核心理念。

1-4 正規化目標影像

當人眼在辨識字元的時候，就是拿大腦中記得的字型去比對，所以我們寫程式時也一定會先建立這種資訊。就是字型的資料庫，裡面會有每一個字的標準大小圖形。但是我們從原始影像中找到的字元目標，不會剛好與我們事先建立好的字模圖形一樣大，還可能會有傾斜與變形。所以必須有個正規化的過程，以車牌辨識來說，就是將從原圖切割的字元影像，經過幾何計算縮放變形到與標準的字元模型完全一樣大小！



因為我們知道真正的實體車牌與字元邊界框線一定是矩形的，雖然在立體空間中車牌與字元影像會因為拍攝角度而變形，只要將此任意四邊形投影到與車牌或字元模型一樣的直角矩形即可。要理解這部分的演算法，需要一些高中程度的幾何學，也就是本書中會用到最深奧的數學概念了！所以讀者不必擔心自己的數學不夠好而很難閱讀本書。

1-5 字模比對確認資料

完成上述步驟，接下來就可以用個別的字元模型去比對出車牌內容是哪幾個字了！一旦確認是哪些字，影像資訊就正式變成文字資訊了！以車牌辨識來說，這些文字資訊就可以與車牌資料庫比對，做很多的後續應用。譬如確認為可以開柵欄通行的白名單，或必須攔截查緝的黑名單等等。

一般來說，在辨識流程中「找字」是主要的難事，「認字」則是簡單的工作。但是通常必須引進很多「**影像之外**」的資訊與邊界條件，來對你找出的答案做檢驗。譬如在台灣不會有「EZ-952B」這種車牌，考量真實世界的物理或法規限制，你可以直接認定那個 **B** 一定是 **8** 的誤認，必須將答案改成「EZ-9528」。

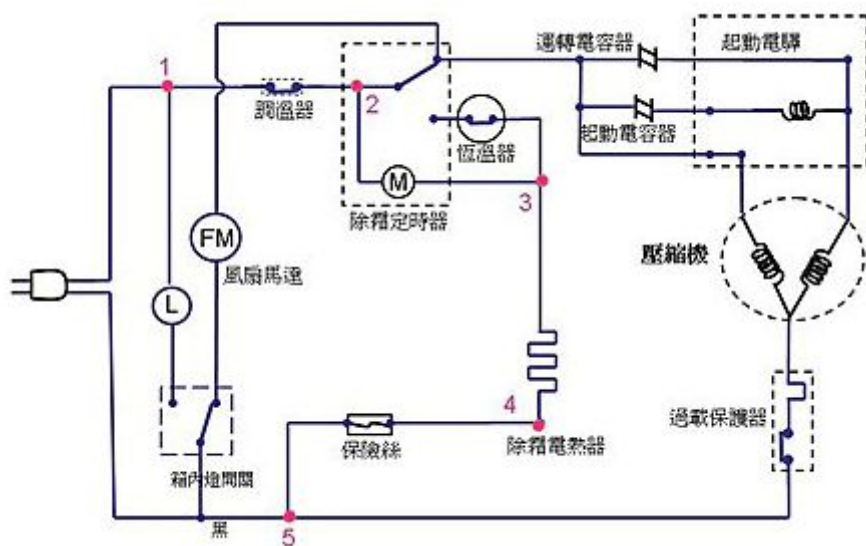
我常說：「影像辨識不只是辨識**影像**！」我們的目的是**辨識出合理有意義的答案**供使用者應用，通常被辨識的目標都會受到一些既定的規則限制，當辨識結果與這些非影像的條件衝突時，我們必須用程式做合理的評估判斷，甚至修改！這不是作弊！因為那些規則比經過拍攝過程被扭曲改變的影像更「**真實**」！如上例如果你堅持「EZ-952B」的「誠實」辨識結果，才是違背影像辨識目的的錯誤決定。

1-5 目標辨識只是影像辨識的一種

以上的影像辨識程序基本上是遵循傳統 OCR 的程序，它們通常是指文件影像上的字元辨識，但車牌也是字元組成，算是廣義 OCR 的一種。那是不是所有的影像辨識都必須照這個程序進行呢？絕對不是的！譬如指紋辨識就不行！



切割出上圖中曲折交錯的線狀目標，對於辨識是誰的指紋幫助不大？因為肉是軟的，那些線條很容易會在壓指紋時略微移位，差一點點就會對不準模型影像了！通常的作法是找出這些線條的**分岔點**，然後依據這些點的分布型態做分類，建立個人的特徵值，作為最後目標比對的依據。所以資訊簡化的方向就不是找到塊狀的目標，也不是線狀目標，而是**分岔點的位置、種類與分布**！使用的辨識程序當然會不同！



又如上面的電路圖，如果要辨識出其中的電子元件，因為全圖元件都是**相連**的，你根本無法以 OCR 的概念切出**獨立**的元件目標！必須發展出其他的程序才能抽離元件。所以影像辨識的流程不會是通用的！必須依據目標特性與辨識目的做針對性的設計，本書限於篇幅，只會介紹以 OCR 為基礎的最常見程序，但影像辨識的學問絕對不止於此！

1-6 眼見為憑，看到過程很重要

本書除了介紹可用的影像辨識方法實作，也非常注重讓讀者充分理解每一個辨識處理的過程與意義。一般專業等級的電腦書籍普遍有兩個缺點：第一是**缺少完整的程式範例**，讀者很難自行拼湊出專家級的完整程式，常常因為一個小細節卡關就無法完整實作，只能放棄繼續鑽研。所以本書強調每一章節介紹之功能都有完整的專案程式碼，你絕對可以完整複製做出書中介紹的所有動作。

第二是複雜的程式流程即使可以實作，但是很多過程細節**沒有足夠的說明**或圖示，較抽象的程序，對於讀者來說還是一個黑盒子！這一點我們會盡量在範例專案中，使用影像或文字介面顯示所有**過程資料**，讓讀者可以充分追蹤理解每一個辨識過程，以及每一個影像角落裡發生的事情。

如果我們的程式專案提供的分析影像功能還是不夠，我們會讓每一處理過程的影像都可以逕行輸出為檔案，讓讀者可以使用其他專業的影像處理軟體，譬如 PhotoShop 進行更多的細節檢視、分析或實驗。事實上，這就是我們公司團隊日常研究影像辨識的方式！希望本書讀者都能享受到跟我們一樣愉快的影像辨識工作經驗。