

顏色辨識在即時視訊上的研究與實作

賴岱佑 台北科技大學 t8599004@ntut.edu.tw
郭忠義 台北科技大學 jykuo@csie.ntut.edu.tw
黃福助 台北科技大學 t7599002@ntut.edu.tw

摘要

對於即時視訊監控系統，物件顏色的識別相當重要。例如每當發生犯罪案件，警方迅速就監視器拍攝到犯罪事件，分析出有用的特徵。其中顏色是重要的特徵之一，不受物體的大小、位置、型變和時間等限制。影響即時識別顏色的效能與精確度，包含硬體因子，如鏡頭材質、鏡頭參數等；以及軟體因子如演算法效率、準確性、實用度等。目前有許多靜態顏色分析的方法被提出，但卻無法有效的改善即時視訊顏色識別的問題。本研究提出一個動態分析演算法，其步驟包括顏色降維、色彩轉換、色彩分群、與即時色彩辨識。最後，本研究提出的方法能有效改善即時辨識效能與正確率。

關鍵詞：模糊類聚演算法、明確集合演算法、顏色識別。

Abstract

In the real-time video surveillance system, the recognition of object color is very important. For example, police need to rapidly analyze the video record to get some useful features after some criminal events happen. The "Color" is one of the important features of the video, and it does not vary with some object variables including size, location, type, and changing timing. There are many factors which can influence the efficiency and accuracy of the color recognition. The factors of hardware include the material of camera lens, the parameters of camera lens, the parameters of infrared, and so on. The factors of software embrace the efficiency of algorithms, accuracy and practical degree, etc. Although many static analysis methods have been proposed, they cannot effectively solve the color recognition in the real-time video system. In this paper, a dynamic analysis approach was proposed for improving the efficiency and accuracy of color recognition in the real-time video system. The analysis steps sequentially include the reduction of color dimension, color transform, color classification and real-time color recognition. Finally, we applied a case study to verify our proposed approach.

Keywords: Fuzzy C-Means Algorithm, Hard C-Means Algorithm, Color Recognition,

1. 前言

顏色識別已被應用在許多領域，大部分的研究 [1][4] [6][8] [10][11][12][11][12] 都屬靜態影像分析，甚至需要人工處理，才能獲得較佳的效果。辨識靜態影像的顏色沒有時間軸的變化，而辨識視訊中的物件顏色，須要考量時間軸上的變化，所以使用靜態影像演算法來辨識視訊物件的顏色是有困難。



圖 1 色偏暖色系的影像



圖 2 色偏冷色系的影像

由於攝影鏡頭 (Lens) 參數會影響光量及色偏的變化，因此視訊影像的品質也有所不同。例如圖 1 與圖 2 是在同樣的情境 (路段相同，角度不同) 下，鏡頭參數不同，所產生的影像，可以很明顯的看出圖 1 與圖 2 影像色偏有所不同。圖 1 是偏暖色系的影像，圖 2 是偏冷色系的影像。此外，鏡頭參數本身會針對外在環境光影變化自動修正，導

致每次輸入顏色不同造成同一物件在不同時間會有不同顏色的變化；因此，靜態影像與視訊影像的顏色識別面對的問題有許多的差異。如果直接使用靜態影像顏色識別演算法針對即時視訊處理，除了演算時間較長，無法達到即時演算外，即使只針對同一個物體或場景，靜態影像顏色識別演算法會不斷送出不同的顏色分析結果。



圖 3 於四到六公尺處架設監視攝影機

本論文研究環境為室外，依照標準室外監視攝影機架設位置而定，地點是市區主要道路，為了是完全模擬實際情況。本研究處理即時視訊動態顏色識別，發展的方法經過實地測試，架設於四至六公尺，鏡頭使用 60mm~120mm。使用兩架網路攝影機，架設於車道旁。本研究主要目的是驗證所提出方法，能容許不同硬體參數的變化，及在不穩定的連續影像中，辨識物件正確的顏色。一般監視攝影機一般都架設於二至三樓牆面，或紅綠燈桿、路燈桿、電線桿等四到六公尺處，如圖 3。會遇到風吹、車輛經過、其他因素干擾，因此視訊畫面會隨機產生震動而非完全穩定。辨識顏色必須把握即時擷取的畫面立刻進行運算並輸出結果。



圖 4 顏色識別問題

物件顏色判斷難處在車輛體本身就具有反光特性，一台黃色車輛，車體可能呈現白色、淺黃、橘黃色、深黃色、土色、咖啡色等。車窗倒影可成呈現黑色、白色等。而車牌顏色也

是一個色塊。保險桿、輪胎、車燈等，附屬配件，都是顏色辨識的問題，如圖 4。

2. 各種顏色辨識方法探討

本章探討顏色識別領域上的目前使用的一些研究方法。

2.1. 靜態顏色辨識方法

在[3]的研究中使用 Fuzzy C-Means Algorithm 規劃顏色色塊的組合，且使用此組合和以雪普爾 (M.E. Chevreul) 調和論、[2]奧斯德華德 (Wilhelm Ostwald, 1853-1932, 1909 諾貝爾化學獎得主, 1922 年發表調和論) 調和論、孟謝爾 (Munsell) 調和論、[3]夢·斯本沙 (Moon. Spence) 調和論的審美量化和色面平均值評估，將理論與設計師整合，建構電腦諮詢系統。設計師先輸入圖片，由系統獲得客觀評價，接著設定主色調及調和感覺，最後獲得數組建議配色並設計出符合需求的色彩。顏色識別通常需要人為介入，每個人對於顏色的感覺及定義都不同。

研究[4]提出一套地理資訊系統，處理於空照圖的彩色影像分割。其中分色是一項重要任務，分色必須清楚且乾淨，否則會導致許多錯誤輸出。該研究處理的是單張、靜態的影像，且必須人為調整，更新一張地圖的步驟非常繁瑣。以顏色擷取和分離方法處理色層分離與加工 (Colour separation and layer processing) 議題，依照每一個不同層次的地圖進行處理，提取單獨的顏色，清楚定義該層的範圍；而每一層的結果都會影響到下一層的資訊；例如水文的資料與綠地的資料要分離且加工。

本研究主要處理物件顏色識別，分割物件內的色塊，藉由色塊分離，獲得區域顏色資訊，與上述色層分離概念類似，但執行過程中採用自動演算方法而無須人工介入。

[10]影像檢索領域 (CBIR、Content-Based Image Retrieval) 需要更適合的色彩空間，研究 [10]以 RGB 處理 SAR (Synthetic Aperture Radar)，顏色被視為主要的判斷屬性之一；其研究[6]選擇 CIE $L^*a^*b^*$ (CIE Lab Color Space) 運用歐基里德距離計算附近的色彩。

研究[1]提到 RGB 是三組強度數值，若單獨取出雖可近似得到影像輪廓，亦會遺漏原始影像在其他軸向的資訊，因此需要分割影像做前處理，將 RGB 座標轉換到其他具有線性特

性的色彩座標空間；例如 HIS、CIE、HLS 等色彩空間，以得到更能代表影像特性的特徵值，使以後分割更具效能。

研究[9]利用 CIE 色彩空間辨別不同顏色，例如：紅，橙，黃，綠，藍，紫。該論文提到校準的問題，光譜內容的散射光取決於材料表面分子結構、與光源的光譜分佈，散射光的強度依賴表面光澤度，以及入射光強度的期間不同的測量方法，因此須要一個準確的顏色識別演算法，以校準表面材料和光澤度。光源穩定的控制是任何感測器會遇到的實作校正問題。

研究[6]著重於影像分類，研究對象是室內影像及室外影像。色彩空間使用 YCbCr 能夠有效分析藍色天空以及綠色草地。使用 YCbCr 前處理，細節部分再額外使用 RGB 色彩空間參數處理。這個研究探討的演算法，包括 Grey World (GW) algorithm、Shades of Gray (SG) algorithm、General Grey World (gGW) algorithm、Gray Edge (GE1) algorithm、Second Order Gray Edge (GE2) algorithm，皆不使用 RGB 色彩空間。只有演算法 White Point (WP) 使用 RGB 色彩空間，在實驗數據中 GW、WP 必須以手動修正。

研究[12]探討彩色影像識別，研究對象是全世界網際網路上的靜態影像，共計 79,302,017 張影像。其中包含彩色分析及物件識別，甚至於作影像分類。與本研究有關的是採用彩色縮圖 (tiny images) 的方式以最近鄰居法 (nearest neighbor methods) 處理大量 (large data sets) 影像資料的物件識別 (Object recognition)。

2.2 動態顏色識別

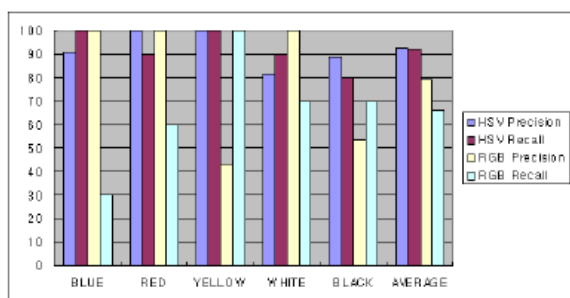


圖 5 HSV 與 RGB 色彩空間比較[11]

RGB 色彩空間，任何一個值變動，就會影響整個顏色的變化。使用 HSV 好處在於可以控制色相 (Hue)；當色相固定也就是顏色固定。當飽和度 (Saturation) 及明度 (Value) 改變，也不會改變顏色。類似於 HSV 的色彩空間還有

HSL。色彩空間線性與非線性研究中，研究[11]提到 RGB 直接做顏色辨識的準確率不好，轉換成 HSV (Hue, Saturation, and Value) 可以增加其準確度，如圖 5 對照使用 RGB 作顏色辨識的準確度遠輸於 HSV 色彩空間；此與本研究論點類似。研究[8]提出將 RGB 轉換到九個顏色模型 (NCM、Nine Color Model)。在 NCM 中， v 代表亮度。

$$v = \frac{0.299 \times R + 0.587 \times G + 0.114 \times B}{255} \quad (1)$$

$$h_{360} = 360 \times h \quad (2)$$

$$n_h = \begin{cases} 1 & \text{if } (v \leq 0.2 \text{ or if } s \leq 0.2) \text{ and } v < 0.8 \\ 2 & \text{if } s \leq 0.2 \text{ and } v \geq 0.8 \\ 3 & \text{if } v > 0.2 \text{ and } s > 0.2 \text{ and } \\ & h_{360} \geq 330 \text{ and } h_{360} < 20 \\ 4 & \text{if } v > 0.2 \text{ and } s > 0.2 \text{ and } \\ & h_{360} \geq 20 \text{ and } h_{360} < 35 \\ 5 & \text{if } v > 0.2 \text{ and } s > 0.2 \text{ and } \\ & h_{360} \geq 35 \text{ and } h_{360} < 65 \\ 6 & \text{if } v > 0.2 \text{ and } s > 0.2 \text{ and } \\ & h_{360} \geq 65 \text{ and } h_{360} < 165 \\ 7 & \text{if } v > 0.2 \text{ and } s > 0.2 \text{ and } \\ & h_{360} \geq 165 \text{ and } h_{360} < 200 \\ 8 & \text{if } v > 0.2 \text{ and } s > 0.2 \text{ and } \\ & h_{360} \geq 200 \text{ and } h_{360} < 270 \\ 9 & \text{if } v > 0.2 \text{ and } s > 0.2 \text{ and } \\ & h_{360} \geq 270 \text{ and } h_{360} < 330 \end{cases}$$

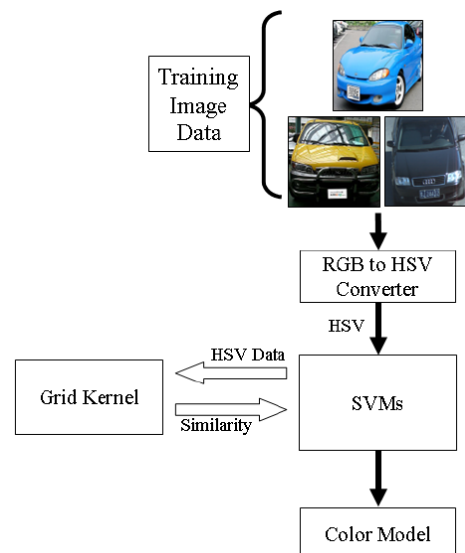


圖 6 模型產生程序[11]

研究[11]在訓練問題上，必須先將訓練影像資料轉換到 HSV 色彩空間，運用支援向量機 (Support Vector Models, SVM) 輸入到該論文提出的 Grid Kernel 計算相似度，最後由 SVM

輸出所屬的顏色模組，如圖 6。其研究可辨識藍色、紅色、黃色、白色、黑色的車輛，正確率為 90.9%、100%、100%、81.8%、88.8%。由於光線影響，加上車窗有偏白及偏黑情形，還有白色烤漆有鏡面效果，有時會倒映到黑色影像。黑色烤漆也有鏡面效果，有時會呈現白色的像素。該論文實驗影像不能太大，採用 150x150 Pixels 影像，影像不能包含太多背景，以人工切割方式，因此無法達成即時影像顏色辨識。另外，研究[11]說明顏色識別會被外在的光線、氣候環境和不同的情境所影響，本研究論文針對這個問題提出解決方法。

3. 動態視訊顏色辨識方法

本研究對即時視訊顏色辨識的問題，提出三點解法。首先使用縮圖，減少計算量。其次是轉換影像顏色的色彩空間。最後透過統計方法，以數量最多的顏色代表物件顏色。詳細演算法於第二段描述。

3.1 縮圖處理

基於[12]研究，如圖 7，縮圖提供足夠的資訊做顏色識別，因處理的資料量減少而有助於運算效率。本研究採用 D1 Size(720x480 Pixels)，長與寬各除以 4，縮小為 180x120 Pixels，資料量是原始的 1/16 倍，較適合做即時演算。



圖 7 Humans can correctly recognize and segment objects at very low resolutions[12]

3.2 物件自動抽取

識別物件的顏色，如圖 8，可採取研究[11]的方法人工定義物件範圍並切圖。本演算法則透過自動物件追蹤以定義物件範圍，自動處理顏色識別。

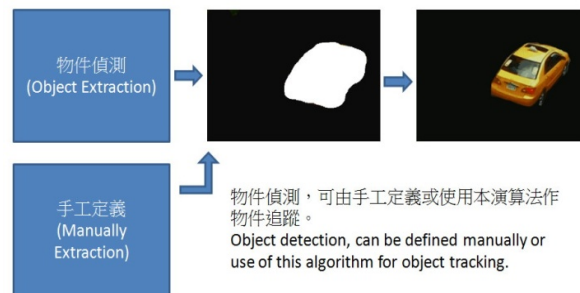


圖 8 手工定義或使用本演算法作物件偵測追蹤

3.3 線性色彩空間轉換

影像原始資訊是 RGB，因為 RGB 是非線性，直接辨識效果不理想，因此需轉換到線性空間。線性色彩空間有 YUV、YIQ、HSV 等，如此有助於顏色群聚演算。轉換色彩空間的演算速度越快越好，這才有助於即時運算。

3.4 C-Means Algorithm

為即時演算效率及精確，選擇 C-Means[4]處理顏色群聚的分析，以計算顏色數目。實驗 Fuzzy C-Means (FCM) 和 Hard C-Means (HCM)，兩種演算法的區別是 HCM 採用明確 (Crisp Set) 集合，而 FCM 採用模糊 (Fuzzy) 集合理論，其中 HCM 實驗效率及精確率較好。若 HCM 仍無法達到即時演算效率，本論文提出兩點方法以達到快速收斂目的。第一是提升分群數及分群中心值準確率，若一開始分群中心值就非常接近收斂位置，就可快速收斂。第二使用演算較快速的距離法，因為距離法針對每一個像素演算，善演算效率能大幅提升整體效率。第三段將詳細描述此方法。



圖 9 演算法自動分析，且輸出為黃色

3.5 語意定義與顏色匹配

圖案顏色青色看起來可能是綠色，也可能是藍

色。對沒有受過色彩學訓練的人答案都不同。考量人性及友善度，本研究允許使用者定義顏色代表的語意，輸出的顏色將會匹配使用者定義的語意。好處是輸出的結果符合使用者預期，使用者可先定義出顏色範圍並設定語意。依據使用者設定的範圍，使用統計方法，找出數量最多的顏色為代表色。在即時辨識時，物件顏色與使用者定義的顏色匹配時，就會顯示或輸出定義後的顏色語意。

3.6 靜態與動態處理

本研究方法略作調整可處理靜態與動態視訊影像。靜態影像中，定義色彩轉換後，依據物件範圍內統計數量最多的顏色，作為輸出代表色，並統計物件範圍內各種顏色所佔百分比。動態視訊影像中，基於時間軸統計多個畫格 (Frame) 內物件範圍的顏色，將所有時間軸顏色數量作累，決定該物件的顏色。

3.7 即時視訊顏色辨識演算法

3.7.1 Hard C-Means

[5]使用明確集合論依據明確分割 (Hard C-Partitions) 演算法則，每一個像素所屬的類別歸屬程度只有 1 或 0，如表 1。

表 1 Hard C-Means (HCM) 示意圖

Classification Pixels	Classification					$\sum_{j=1}^c \mu_{ik}$
	C1	C2	...	Cc		
P1	0	1	...	0	1	
P2	1	0	...	0	1	
...	
Pn	0	0	...	1	1	

$$\mu_i : X \rightarrow \{1,0\}$$

$$x_k \rightarrow \mu_i(x_k) = \mu_{ik} \in \{1,0\}$$

$$1 \leq i \leq c$$

$$1 \leq k \leq n$$

μ_{ik} 代表第 k 個像素於第 i 個群組的歸屬值

x_k 代表整張影像的第 k 個像素

c 代表群組數

n 代表影像像素總數

假設有 c 個群組，群組中心向量通常是隨機選定[5]。其他像素點通常使用用歐基里德距離 (Euclidean distance) 計算歸屬值，以迭代法求出新的群組中心，若不收斂則繼續計算歸屬

值求取新的群組中心，若收斂則結束演算。收斂條件可以自行決定。為證明 HCM 計算速度比 FCM 快，採用一張 24bit 全彩、720x480 Pixels (D1 Size) 的影像比較。分群中心以隨機 (Randomization) 決定，設定為 19 群，19 群是本範例圖最佳群數。表 2 是比較使用 FCM 和 HCM 的執行速度，單位是秒。

表 2 使用 FCM 與 HCM 的執行效率比較

Method Number	FCM	HCM
1	7.02	4.618
2	11.716	6.955
3	9.36	1.447
4	7.066	4.08
5	7.051	3.433
6	9.375	2.112
7	7.02	2.791
8	9.344	2.098
9	9.376	2.108
10	9.516	2.784
Average	8.6844	3.2426
	second	second

顏色分群準確度也可透過影像輸出判斷。圖 10 一輛黃色的車子，透過隨機設定分群中心方式，無論 FCM 或 HCM 都有計算分群錯誤的機率，圖 10 可看出黃色車輛被轉換為綠色，這在即時影像分析上會造成很大的錯誤，因為即時影像分析在時間軸上，相關連的物件顏色必須要相近，需要用到顏色資訊以判斷兩個畫格 (Frame) 中的物件是同一個。使用傳統 FCM 或 HCM 在效率及準確度都不適合使用在即時影像分析。本研究針對此一問題改良並敘述於下一小節。

Origin Image



False Image(FCM) False Image(HCM)



圖 10 FCM 和 HCM 使用隨機分群計算後錯誤影像

3.7.2 決定群組數及群組中心

要使 HCM 收斂快速，本研究提出第一個方法是，設定接近收斂條件的群組中心。將顏色分群才可使用統計法計算顏色數量，由於 RGB 是非線性空間，因此轉換線性色彩空間獲得較好的分群結果。由於 HSV 的 Hue (色相) 描述 360 種顏色量，將 RGB 空間轉換到 HSV 空間時，建立 Hue 長條圖 (Histogram)，可明確知道各種顏色使用數量。為了更接近群組中心，計算飽和度 (Saturation)、數值 (Value) 平均值。圖 11 長條圖為圖 12。



圖 11 全彩 24bit 720x480 Pixels 影像

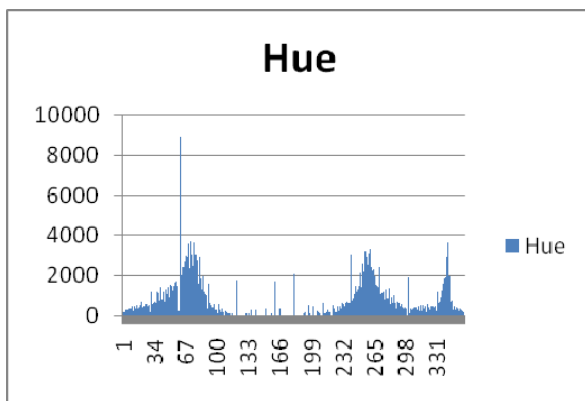


圖 12 圖 11 的長條圖

依長條圖，人類直覺知道最少有三群顏色，但不可能用三種顏色就可以明確描述圖 11，必須找出數量足以代表群組性質的顏色值。若單純使用門檻值，會造成很多鄰近相似的顏色值被規劃為群組中心，十分不準確。本範例長條圖平均值 750 當門檻值有 111 筆，代表 111 群組，這樣組數太多，不適合顏色統計。使用標準差，衡量觀測資料與平均數之間的差異量。

$$\text{標準差公式} \sqrt{\frac{\sum (x - \bar{x})^2}{(n-1)}}$$

為避免太多相似群組產生，不直接使用門檻值，採用相鄰數相減並取絕對值大於標準差。

$$c_n = \begin{cases} 1, & |x - \bar{x}| > \sqrt{\frac{\sum (x - \bar{x})^2}{(n-1)}} \\ 0, & \text{otherwise} \end{cases}$$

本範例標準差是 950.9402941 樣計算結果分群數降為 32 筆。但仍有相鄰分群值產生。原因是當長條圖數值為孤立且量大於標準差時，就會產生這種情形。採用統計的方法處理孤立值

產生的相鄰分群，當有連續值 (相鄰分群值) 產生就開始累計，直到不連續，就將該筆資料設定為單一分群值，也就是分群中心。演算結果降為 19 組，代表 19 組的顏色群組中心。這 19 組 Hue 值，與平均飽和度 (Saturation)、數值 (Value) 結合轉換回 RGB 值，這 19 組 RGB 值是本研究提出最接近群組中心的色彩。表 3 是 19 組 HSV 值及轉換後的 RGB 值。

表 3 19 組 HSV 值及轉換後的 RGB 值

Saturation	60		
Value	153		
Hue	Red	Green	Blue
0	255	82	82
30	255	169	82
59	255	252	82
70	226	255	82
72	220	255	82
75	212	255	82
79	200	255	82
90	168	255	82
120	82	255	82
160	82	255	197
180	82	255	255
240	82	82	255
252	117	82	255
255	125	82	255
259	137	82	255
270	169	82	255
300	255	82	255
340	255	82	140
343	255	82	131

為證明效能提升，使用亂數決定分群中心，隨機建立分群中心法來比較。為求客觀，分別演算十次求取平均值。表 4 是本研究所提方法與隨機分群法效率比較，可觀察出隨機分群中心法會因隨機分佈情形不同而影響效率，以及精確度。本實驗使用隨機法有 3/10 機率會將黃色車輛錯轉換為綠色。使用本研究方法，時間較為穩定，處理一張 720x480 Pixels (D1 Size)、24bit color 的影像，平均約 0.7425 秒。

表 4 本研究所提方法與隨機分群法效率比較

Method Number	Randomization	Proposal
1	6.755	0.811
2	2.684	0.749
3	4.883	0.733
4	2.668	0.733
5	1.373	0.717
6	2.683	0.733
7	3.323	0.749
8	2.933	0.734
9	2.043	0.733
10	4.836	0.733
Average	3.4181	0.7425
	second	second

以上改進仍然無法完全滿足即時演算的需求，因此需要透過縮圖來降低處理資料量，本

研究參考[12]的研究，縮圖後的資訊仍足以判斷色彩，將 D1 Size(720×480 Pixels)，長與寬各除以 4，縮小為 180×120 Pixels，資料量是原始的 1/16 倍。表 5 是使用縮圖 180×120 Pixels，並用本研究所提之分群中心演算法，處理的時間平均為 0.109 秒。

表 5 180×120 Pixels 縮圖處理時間

Method Number	Proposal
1	0.109
2	0.109
3	0.109
4	0.109
5	0.109
6	0.109
7	0.109
8	0.109
9	0.109
10	0.109
Average	0.109
	second

FPS 是每秒鐘輸入畫格張數，常見的規格是 1 FPS~30FPS，一秒鐘除以格數可得每一畫格輸入時間，可計算出視訊演算法可以運算的時間。表 6 列出常見的 FPS 與每張畫面送出時間，等同於演算法處理時限。演算法若能在時限內處理，理論上不會有延遲情形，達成即時演算的目標。

表 6 常見 FPS 與時間表

FPS	Time Second
1	1.0000
3	0.3333
5	0.2000
6	0.1667
7	0.1429
10	0.1000
15	0.0667
25	0.0400
29	0.0345
30	0.0333

表 6 顯示，本研究適合於 7FPS 以下的視訊設備，因為 $0.1429 - 0.109 = 0.0339$ (秒)，還有 0.0339 秒可以處理其他演算。實際運作時，理論上顏色識別不需要每一張畫格都計算，當物件偵測演算法 (Object Tracking Algorithm) 發現物件後，才需要做顏色識別演算法 (Color Recognition Algorithm)。因此延遲時間可以被人為 (因人而異) 接受，實際測試結果設定 15FPS 視訊設備，處理顏色識別演算沒有感覺延遲。

3.7.3 顏色準確度

在顏色準確度上，先做靜態影像比較，將隨機群組中心與本研究作相鄰畫格像素差異數量比較，使用隨機群組中心方法轉換顏色後明顯的不穩定，幾乎相鄰畫格都有差異，無法用在智慧型視訊分析如：物件追蹤等。本研究無論轉換多少次，其中心是穩定的，相鄰畫格顏色並不會有所差異。

表 7 本研究與隨機中心相鄰畫格差異像素統計

Method Number	Randomization's Difference Pixels	Proposal's Difference Pixels
1	21554	0
2	21592	0
3	21592	0
4	21591	0
5	21587	0
6	21537	0
7	21523	0
8	21592	0
9	21576	0
10	21589	0
Average	21573.3	0
Total : 21600	Pixels	Pixels

表 7 是十張 180×120 Pixels 的影像，總像素數量為 21600，因此隨機群組中心幾乎是每張皆有差異。

4. 實驗結果

4.1 台北市警局鄰里道路監視系統標案為例

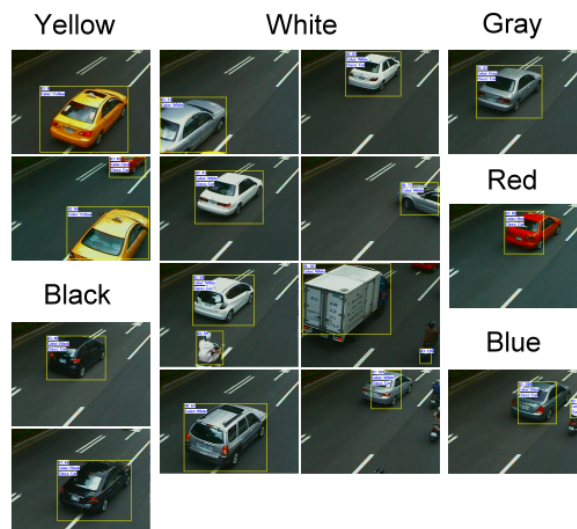


圖 13 台北市警局鄰里道路標案顏色辨識比賽檔案測試擷圖

台北市警局於 2009/6/29 舉辦智慧型監視系統演算法標案競賽，在顏色辨識上，規定偵測第二車道，並辨識白色及黃色車輛，辨識的車輛除了有主辦單位準備的白色及黃色車輛，還有道路上隨機出現的各種顏色車輛。本研究取得比賽錄製的即時影像，進行實驗，實驗擷圖如圖 13。該檔案資訊如圖 14。

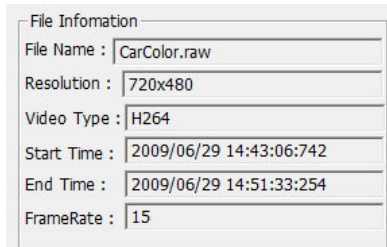


圖 14 台北市警局鄰里道路標案顏色辨識影像檔案資訊

本研究除了偵測到黃色車輛與白色車輛，另外也偵測到黑色、灰色、紅色、藍色車輛。且由圖 13 可見本研究可同時偵測多個物件的顏色，而非只限定於單物件偵測。比賽影片中，黃色車輛應有 15 車次，白色車輛應有 15 車次。本研究偵測結果為黃色 15 輛車，白色 15 輛車，符合比賽要求。

4.2 自行道路即時視訊實驗結果

本研究先探討道路即時測試不佳的範例及原因，規格採用正面拍攝、雙車道，影像壓縮分別採用 H.264 與 MPEG4，影像大小為 720x480 (D1 Size)，時間分別選在中午 11 點半左右及下午 16 點半左右，FPS 選用 15 及 30FPS。

表 8 辨識率不佳的範例表

No.	Snapshot	Resolution	Video Type	Start Time	The length of a film (minute)	FPS (Frame Per Second)	number of vehicle	degree of accuracy	number of error
1		720x480	H.264	2009/02/11 11:32:19:503	06:20	15	71	88%	8
2		720x480	H.264	2009/02/11 11:38:40:883	04:05	15	40	87.50%	5
3		720x480	MPEG4	2009/01/22 16:35:36:890	01:35	30	38	84%	6
4		720x480	MPEG4	2009/01/22 16:45:31:470	02:26	30	54	92%	4

表 8 是辨識不佳的範例表，本研究檢討原因，範例 1、2 共通原因是中午時段陽光較強，會產生陰影，而本研究沒有做陰影消除，因此有些車輛若太靠近，會變成單一物件，此時計算顏色會不準確。此外，越接近陽光的車輛面，容易產生反光，也是誤判原因之一。範例 1、2 準確度約在 88%、87.5%。

範例 3、4 為下午 4 點半左右，沒有陽光影響，天氣陰天稍微偏暗，但這不是造成誤判的原因。造成誤判原因是車燈，此時有些車輛開車燈，大部分車輛沒有開車燈。因此車燈是造成顏色誤判的主要原因。

實驗片段是每十分鐘左右錄影，總時數累積八小時。拍攝角度有車輛正面、背面及側面。實驗在一般正常環境，氣候穩定如晴天、光源平均、沒有人為的光源影響下，辨識率約有 9 成以上。經過參數調整，本研究可在兩天進行顏色辨識，準確度約在 9 成以上。

4.3 演算法規格設定

本研究的規格：

1. 自行定義欲辨識的顏色。
 - (1) 使用人工自定義顏色。
 - (2) 使用系統預設顏色（灰、白、黑、紅、黃、藍、綠）。
2. 由 RGB 色彩空間轉換後仍可辨識顏色。
 - (1) 可使用 YUV 色彩空間。
 - (2) 可使用 YIQ 色彩空間。
 - (3) 可使用 HSV 色彩空間。
3. 提供物件顏色數量百分比。
 - (1) 提供所定義顏色在該物件中所佔比例。
 - (2) 使用者可自行依顏色所百分比決定物件顏色。
4. 隨環境或鏡頭變化調整。
 - (1) 隨環境如天候影響變化調整。
 - (2) 隨鏡頭如不同廠牌鏡頭參數變化調整。
5. 不受限影像來源。
 - (1) 影像來源可以是靜態如數位相機、無線口服內視鏡。
 - (2) 影像來源可以是動態如類比或數位攝影機或監視器。
6. 可內建於嵌入式系統：演算法具有即時效率，可於嵌入式系統演算。

5. 結論

本研究提出適合於視訊的顏色辨識演算法，具有九成以上的辨識率。透過即時電影視訊畫格輸入，直接輸出分析結果的擷圖及時間，做回放、找尋等功能，可應用於追查以顏色為特徵的物件，如追查贓車。未來對於演算法效率的提升、辨識準確度的提升、人為光線干擾的問題如車燈、解決天候影響如車身反光、物件陰影的問題如物件影子過長等的研究改善，可以更有有效的運用於智慧型監視系統。

致謝

This work was supported by the National Science Council under grant NSC-98-2752-E-008-001-PAE and NSC-98-2220-E-027-009.

參考文獻

- [1] 李建德、黃鐘賢，”Fuzzy C-means 演算法於彩色影像自動分割之研究”，**1999 年 NCS 全國計算機會議**，2006。
- [2] 陳昱帆、林銘泉，”配色面積對於色彩意象之影響”，**國立成功大學、工業設計學系、碩士論文**，2008。
- [3] 陳崇憲、蕭世文，”應用色彩審美度於手機設計之研究”，**國立成功大學、工業設計學系、碩士論文**，2005。
- [4] Ablameyko, S., Beveisbik, V., Homenko, M., Paramonova, N., Patsko, O., ” Interpretation of colour maps. A combination of automatic and interactive techniques,” **Computing & Control Engineering Journal**, Vol. 12, Issue 4, pp. 188-196, 2001.
- [5] Bezdek, J.C., Pal, N.R., "Some new indexes of cluster validity," **IEEE Transactions on Systems, Man, and Cybernetics, Part B**, Vol. 28, Issue 3, pp. 301-315, 1998.
- [6] Bianco, S., Ciocca, G., Cusano, C., Schettini, R., "Improving Color Constancy Using Indoor-Outdoor Image Classification," **IEEE Transactions on Image Processing**, Vol. 17, Issue 12, pp.2381-2392, 2008.
- [7] Commission internationale de l'eclairage, ” CIE Colorimetry — Part 4: 1976 L*a*b* Colour Space,” **Joint ISO/CIE Standard**, ISO 11664-4:2008(E)/CIE S 014-4/E:2007
- [8] Li Jinwei, Liao Guiping, Xiao Fen, “Rapeseed seeds colour recognition by machine vision,” **27th Chinese Control Conference**, pp.146-149, 2008.
- [9] Nguyen, D. Thong, Kennedy, Hamish, Cornwall, Jeff, “A Vocalized Color Recognition System for the Blind,” **IEEE Transactions on Instrumentation and Measurement**, Vol. 33, Issue 2, pp. 122-126, 1984.
- [10] Puzicha, J. Buhmann, J.M. Rubner, Y. Tomasi, C., "Empirical evaluation of dissimilarity measures for color and texture", **The Proceedings of the 7th IEEE International Conference on Computer Vision** , Vol. 2, pp. 1165-1172, 2002.
- [11] Son, Jeong-Woo, Park, Seong-Bae, Kim, Ku-Jin, “A Convolution Kernel Method for Color Recognition,” **6th International Conference on Advanced Language Processing and Web Information Technology**,, pp. 242-247, 2007.
- [12] Torralba, A., Fergus, R., Freeman, W.T., "80 Million Tiny Images: A Large Data Set for Nonparametric Object and Scene Recognition", **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Vol. 30, Issue 11, pp. 1958 - 1970, 2008.