

# 臺灣二〇〇六年國際科學展覽會

科 別：電腦科學

作 品 名 稱：彩色數位影像資料庫檢索架構-以國立自然科學博物館為例

學校 / 作者：國立臺中第一高級中學 林子禾  
國立臺中第一高級中學 張君豪

## 作者簡介



我的名字叫做林子禾，生於一個幸福美滿的家庭。受到父親的影響，從小就有一點完美主義的個性。國中時有幸能遇到一位很有耐心且很照顧我的導師，在她的教導下，養成了凡事堅持不懈的人生及學習態度。

升上中一中後，由於自己的興趣，參加了資訊人才培育班，開始投身於程式語言這個領域。高二在電腦老師帶領下報名了科展，雖然最後沒有得名，學到的經驗及作學問的態度讓我受益良多。如果我有任何成就，我願將所有的榮耀歸功於家人的支持及所有幫助過我的人。



我是張君豪，目前就讀於台中一中二年級。

自從國小五年級買了第一台電腦之後，就對電腦非常有興趣。國小時，很幸運的遇到一位很有耐心的老師，他教的居然是 **FLASH**，於是我就一頭栽進了電腦世界。國中三年期間，班級網頁全部由我製作。

升上高一後，參加了學校舉辦的人才培育班，教的是我從來沒有碰過的程式語言，興趣加上天分，讓我學習得很快。希望將來能繼續的學習。

# 摘要

## 中文摘要

國立自然科學博物館推出《數位博物館》，內含大量生物圖鑑及豐富館藏文物之圖文資料，無論用於資料查詢或提供進階研究資料皆有極大貢獻。但經使用後發現，其檢索架構仍只提供關鍵字搜尋及分類瀏覽，無法精確檢索出欲查詢之資料，尚有改進之處。

於是針對現有之影像檢索系統，我們跳脫出傳統以文字為檢索之依據，而設計出一連串之改進方案，如下：

1. **IRHI 色調辨識影像檢索**，針對色調相近之影像類群提供良好檢索架構。
2. **IRCI 輪廓辨識影像檢索**，針對輪廓相近之影像類群提供良好檢索架構。
3. **IRHCI 色調暨輪廓辨識影像檢索**，綜合前述兩種方法之優點所設計。

## **Abstract**

A while ago, the National Museum of Natural Science put forth the “Digital Museum,” which contains great amount of biological pictographs and abundant collections of textual as well as pictorial materials. This has contributed tremendously to information search and advanced research. However, after employing them, we come to discover that its retrieval paradigm only provides key-word search and categorization browse, without enabling us to precisely pick out the desired data. Thus, this paradigm leaves something to be desired.

To make up for the insufficiency of the existing system, we have escaped from the concept of searching by texts. Instead, we have designed a series of improvements. They are as follows:

1. IRHI(Image Retrieval by Hue Identification): Providing a sound paradigm for the image groups composed of similar hues.
2. IRCI(Image Retrieval by Contour Identification): Providing a sound paradigm for image groups composed for similar contours.
3. IRHCI(Image Retrieval by Hue and Contour Identification): Combining the strengths of the above two paradigms.

# 壹、緒論

## 一、研究動機

隨著多媒體技術及網路快速地發展，整個網際網路匯聚成一個相當巨大的分散式影像資料庫。網際網路的普遍也促成博物館、美術館等機構將大量的影像數位化，建立所謂的『數位圖書館』。然而龐大的影像資料，如果沒有一個簡便快速的檢索方法，數位化不但沒有發揮其優點，反而會使影像資料難以尋找。

例如國立自然科學博物館（以下簡稱科博館）之數位博物館而言，仍以輸入關鍵字進行檢索，若有一張動物影像，一般使用者不知其分類依據，在龐大的影像資料庫裡，根本無從找起。

因此本研究希望能以科博館現有影像資料為實驗材料，並研究出一個新的《彩色影像檢索系統》，改進科博館現有以文字檢索繁瑣沒有效率的缺點，進而發展一個使用簡便、快速、有效且能準確地找到所需的影像資料的檢索技術，使科博館的影像檢索系統更能符合社會大眾的使用需求。

## 二、研究目的

目前市面上已經有好幾套影像檢索系統，大部分都是應用在保全系統，如指紋辨識、人臉辨識等。但這些影像辨識主要是以單色為主，因此彩色影像的檢索技術已成爲當今最熱門的檢索技術研究主流。其原理主要是透過影像本身的視覺內容特徵來判斷圖像的相似性，篩選出相似性最大的影像。

因此本研究的目的是以使用者的需求爲主，設計出一種簡便有效『以圖找圖』的檢索技術，進而捨去傳統以輸入關鍵字爲主的檢索方法。

## 三、研究方法

1. 設計一個有效萃取彩色影像資料特徵值的演算法。
2. 設計一個使用最小的記憶體空間儲存特徵值的方法。
3. 設計一個準確率及回收率最佳化的檢索技術。

## 貳、文獻探討

### 一、數位影像簡介

在人類的文化發展過程中，影像（Image）就一直扮演著非常重要的角色。人類不僅使用語言、文字，更學會使用圖像來彼此溝通、紀錄各種歷史。直到今天，資訊科技的突飛猛進加上網路的普及，使得數位化的影像資料充斥著整個網路世界。人類對於這麼龐大的影像資料，運用的需求是一天勝過一天。

影像（Image）是現實世界的視覺圖像在平面上呈現出來的結果；人類的視覺圖像、繪畫照片、電影及電視畫面等，都屬於影像。廣義的「影像」可定義為「依某人或某物的外型作成的複製品」，它不僅涵蓋動態影像(Moving Images)，也涵蓋靜態影像(Still Images)。而一般靜態影像的表現形式又可分為二維平面(Two-Dimensional)與三維立體(Tri-Dimensional)。狹義的靜態影像所指的是二維平面的靜態圖像，其涵蓋的範圍相當廣泛，例如文件影像、照片、影像、插畫、人工繪圖、電腦繪圖…等。如圖 2.1.1 所示。

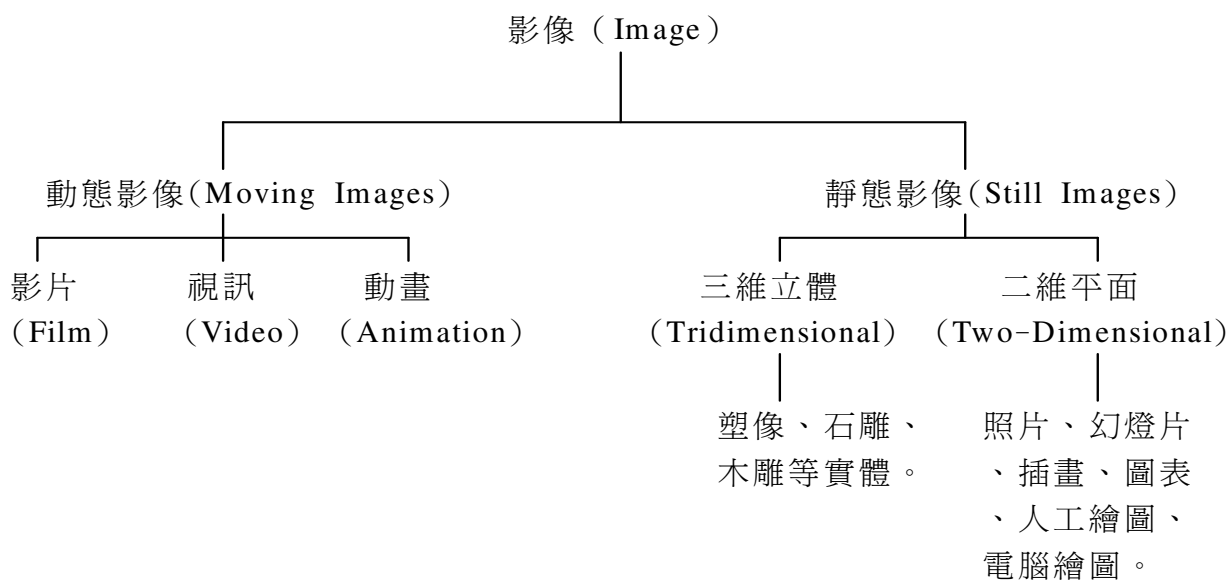


圖 2.1.1 影像的定義

## 1. 有關數位影像的專有名詞

一般常見的數位影像相關的專有名詞有像素、解析度、影像模式、影像尺寸、列印尺寸、檔案大小、檔案格式等等，茲分別簡要說明如下：

**像素 (Pixel)：**所謂的像素就是圖像元素(Picture Element)的意思。由於點陣影像是由像素所組成，因此，對於一個以點陣影像格式存放在電腦中的影像而言，點陣影像中所包含的像素愈多，影像檔案就會愈大，能表現的影像內容也越豐富。

**影像尺寸 (Image size)：**影像尺寸指的是影像中所包含像素的個數，通常以“寬 × 高”來表示影像的大小。例如：800 × 600、640 × 480 等。影像尺寸表示的方法如下：

$$\text{影像尺寸} = \text{影像寬度的像素數量} \times \text{影像長度的像素數量}$$

**檔案格式：**檔案格式 (file format) 指的是電腦系統如何把影像的像素值儲存在記憶設備 (磁碟、硬式磁碟、光碟、磁帶等) 中，檔案格式與壓縮方式及檔頭 (file head) 的設計方式有關，常見的檔案格式有 PSD(Photoshop)、GIF、TIFF、TGA、JPEG、PNG、EPS(Enhanced Postscript)等等。

## 2. 影像色彩的類型

在電腦中，影像是以點陣影像的方式表示。點陣影像是由許多像素所構成，每個像素含有各自的色彩，狀似方格。如果同一尺寸大小的影像中，所含的像素愈多，圖像的色彩就愈豐富，影像的效果也就顯得愈細緻。點陣影像的色彩，是由組成點陣影像的每一個像素點的色彩綜合呈現出來；點陣影像要呈現的色彩愈多，就必須使用愈多的位元數來表示，因此，色彩也會影響點陣影像的呈現效果和檔案大小。

每個像素能夠表現的色彩範圍稱為色階(Color-Depth)，也就是說每個像素有多少種顏色可供選擇，使它呈現在畫面上。數位影像可以依據影像中所包含的色彩數目，區分為黑白影像、灰階影像、16 色、256 色及全彩等，分別說明如下：

**黑白：**用 1 bit 來記錄顏色，因此最多只能表示兩種顏色，即黑與白。

**灰階：**用 8 bit 來記錄顏色，可以記錄  $2^8=256$  種顏色；在這種類型的影像中，除了純黑與純白之外的 254 種顏色，都被用來表示不同程度的灰色，因此稱為“灰階”影像。

**16 色**：用 4 bit 來記錄顏色，可以記錄  $2^4=16$  種顏色。

**256 色**：用 8 bit 記錄顏色，可以記錄  $2^8=256$  種顏色；雖然灰階同樣是以 8 bits 來記錄顏色，但是 256 色影像除了純黑與純白外，其餘的色彩並不只是黑與白之間的灰色，因此色彩較灰階影像豐富。

**全彩**：用 24 bit 來記錄顏色，可以記錄  $2^{24}=16,777,216$  種顏色；由於色彩非常豐富，在數位影像資料中被稱為全彩。

由上列的幾種色彩類型我們可以知道，運用越多位元來記錄色彩，則記錄的色彩內容就越豐富，但是相對的，影像檔案所佔用的儲存空間也會越大。

### 3. 影像檔案的格式

**BMP**：適用於 Windows 與 OS/2 作業平台的點陣圖檔案格式，大多數的繪圖、影像處理軟體都會支援此格式圖檔，但缺點是儲存時使用的空間較大。

**JPEG**：為網頁常用的圖檔格式，JPEG 檔案具有極高的壓縮率，這種格式會將許多圖片中的像素捨棄，因此也較容易造成影像失真，屬於破壞性壓縮圖檔格式。一般使用掃描器掃描照片時，常儲存成此種圖檔格式。

**GIF**：適用於網頁中的圖檔格式，可以製作網頁中的動畫。由於 GIF 圖檔格式最高只有 256 色的色彩階數，所以將影像儲存成 GIF 格式時，這 256 種色彩以外的其他色彩，都會被相似色取代。此外，GIF 圖檔格式還可製作透明背景的圖案，因此製作網頁時，經常使用此種圖檔格式。

**PNG**：這是一種結合 GIF 以及 JPEG 優點的檔案格式，不但可以顯示 RGB 全彩影像，還可以支援交錯顯示與透明背景的效果。此外，它採用非破壞性壓縮，不但可以有效減小檔案大小，圖片也不會因壓縮而失真。

不同的影像格式都有不同的特性，我們可以根據影像的用途，選擇適當的格式來儲存影像。

## 二、影像檢索系統

隨著電腦科技的發達，各種不同形式的影像、圖片、動畫，在電子文件中隨處可見。全球資訊網(World Wide Web，簡稱 WWW)的興起、發燒，整個網際網路更形成一個龐大的分散式數位影像資料庫。這些成千上萬的影像圖片如能加以適當的儲存、管理與應用，對人類而言是一項非常珍貴的資產。因此如何透過影像檢索系統的研發並且改進目前影像檢索的瓶頸，將影像資料快速且有效率的提供給使用者，是相關研究者的當務之急。

所謂的影像檢索(Image Retrieval)，簡單地說就是在一個影像資料庫中尋找出自己所需要的影像，通常一個影像檢索系統大多含有三個基本的概念，查詢(Query)、比對(Match)和結果(Results)。

查詢包括輸入資料和選擇使用者想要使用的工具或條件，對影像檢索而言，輸入資料可以是一張已經存在的影像，而使用的工具可以用顏色、外形、紋理和空間關係等。至於比對就是在資料庫中找尋符合查詢的資料，通常會根據一些檢索的演算法來找，而針對不同的查詢，使用的演算法也都不相同。結果就是將符合的資料輸出給使用者，通常檢索的結果是依據資料間的相似性(Similarity)來決定的。

因此，對影像檢索而言，上述的三個基本概念缺一不可。影像檢索的模式，如圖 2.2.1 所示。

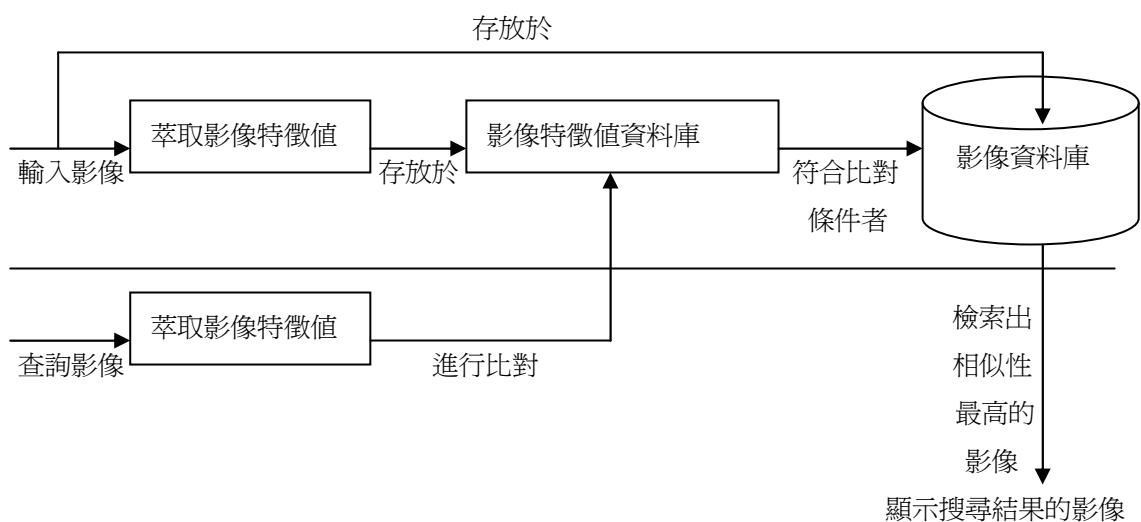


圖 2.2.1 影像檢索模式

影像資訊的檢索技術目前發展出四種查詢介面，其方法如下：

- 1.文字查詢 (Query by Text)
- 2.特徵查詢 (Query by Feature)
- 3.範例圖片查詢 (Query by Image Example)
- 4.瀏覽 (Browsing)

影像檢索系統需要提供兩大功能：

- 1.因為使用者無法很精確地描述影像資料，所以影像資料檢索必須要有相似性檢索 (Similarity Retrieval)的功能，使用者藉此功能可以找出相似的影像，或將影像依照相似度(Similarity Measure)來排名(Ranking)。
- 2.因為使用者往往無法完全描述一張影像，只記得影像的某個部分，所以影像資料的檢索必須要有提供影像本身資料擷取的各種特徵值作為檢索所需要影像的依據，例如顏色、紋路、形狀、空間關係等等。

### 近代以影像本身擷取特徵值作為依據的技術

傳統以文字來檢索影像的檢索技術，如果使用者的認知與分類者所描述不一致時，如此要檢索出一張使用者所需的影像，猶如大海中撈針一樣的困難，因此以影像內容的檢索技術才應運而生。

所謂以影像本身資料擷取各種特徵值的影像檢索系統(Image Retrieval System)，主要是指以影像或利用介面的輔助工具所繪出的圖形當作輸入資料，藉由擷取輸入資料中某些特徵值或向量，再與影像資料中事先存著經由某種演算法所得到的特徵值來進行相似性的比對，然後將最相似的數張影像以及一些相關資訊回傳給檢索者。近代以影像內容檢索的技術，其資料庫管理的查詢方式可分為下列三種不同的層級：

#### 1. 原始像素

這是最低階的一種方式。影像資料的標注即為影像本身，查詢時使用者需要輸入或指定查詢影像 (Query Image)，系統再將查詢影像與資料庫中的影像作對比，找出最相似的影像。此種比對的方式是直接比較兩張影像中相對位置的像素 (Pixels)。

## 2. 特質

這是屬於較高階的一種查詢方式，因為每張影像的內容特徵，在存入資料庫時都會被系統自動萃取出來作為特徵值。使用者在查詢時只需指定影像的特徵值是什麼，即可找到所需要的影像。而最常被用來檢索的特徵值條件包括影像的顏色、紋理及形狀，以下就這三項檢索的特徵值條件來加以介紹：

### a. 以顏色為基礎的影像檢索系統

一般影像檢索系統都會選取顏色作為特徵，因為對人類的視覺來說，顏色是最自然的影像特徵之一，人眼用來辨識影像最常使用的就是顏色。影像的顏色特徵可分為整張圖片的顏色分布比例、主要顏色以及出現位置的比對。顏色分布圖（Color Histogram）則是影像檢索系統最常用到的顏色特徵，利用顏色特徵作為檢索的依據，其過程可分為四個步驟：選擇適當的顏色格式、減少色階、計算顏色分布圖以及比對顏色分布圖。

### b. 以紋理為基礎的影像檢索系統

紋理是影像中最常見到的特徵之一，它可根據下列三種特性加以分類：(a)粗糙度(Coarseness)、(b)對比(Contrast)及(c)方向性(Directionality)，這些特性可經由傳統的影像處理方法擷取得到，不過現今分析紋理的研究最常用的方式是將影像經過轉換後，從中擷取紋理特徵。

### c. 以形狀為基礎的影像檢索系統

形狀是指影像中相同顏色或紋理的區域的形狀，或是指影像中的物體的形狀。形狀演算法的基礎是建立在將影像加以邊緣增強(Edge Enhancement)、邊緣檢測(Edge Detection)與邊緣連結等。使用者只要描繪出一個物件的粗略形狀，系統會以資料庫中的影像與之比對，找出具有符合該物件的影像。

## 3. 語意 (Semantic)

這是最高階的一種查詢方式。使用者可真正的以影像內容或意義，作為查詢的要求而查到所需的影像，在影像資料鍵入資料庫時，需由人工對影像內容詮釋並作文字標注。

**以關鍵字爲主的檢索技術，其優點爲：**

- 1.文字易於表達人類高階的認知與抽象的概念。
- 2.相較於內涵檢索，文字檢索較簡單，技術也均已成熟，系統容易實現。
- 3.適合需要人爲審核或評鑑的場合。

**而以關鍵字爲主的檢索技術，其缺點爲：**

- 1.較費時費力，資料庫中所儲存的是圖片本身相關的代碼或文字片語資料，這些資料必須人爲方式事先建置。
- 2.文字敘述有時會流於主觀，不同人會有不同的認知或描述，有時文字又難以作完整表達或陳述。
- 3.無法直接以某一圖例作查詢（Query by example）。

**內容式的影像檢索技術，其優點爲：**

- 1.電腦直接萃取出影像的特徵元素，如形狀、顏色、紋理等資訊，作爲相似比對的內容，不需太多的前置作業、人力的介入。
- 2.減少人爲的主觀差異。
- 3.適合不需要人爲審核或評鑑的場合。

**而內容式的影像檢索技術，其缺點爲：**

- 1.目前技術尚未成熟，檢索出的圖形，往往無法讓使用者感到滿意。
- 2.低階的技術特性，很難突破與人類較深層抽象意境的差距。
- 3.抽取特徵或進行特徵比對時，需花較多時間作繁雜的運算。

內容式的影像檢索技術並非文字檢索的替代技術，反而應是它的互補技術。只有兩者相互整合成功才能獲得令人滿意的檢索成效。至於未來的影像檢索系統應該會走向多元檢索模式的發展，結合各種影像搜尋模式，讓使用者可以多重選擇符合本身認知的影像檢索方法來進行影像的查詢。

## 參、研究方法

### 一、JL 影像檢索技術的定義

本研究秉持著縮小影像檢索時間及加大影像檢索成功率的基本精神，提出了 JL 影像檢索技術(JL Image Retrieval Technique，以下簡稱 JLIRT)。

在我們的實驗中發現，單純以色調來辨識影像，許多依人的觀點認為相似之影像，檢索成功率並不是很高；若依輪廓加以區分這些影像，則檢索成功率大為提升。有鑑於此，根據不同影像特性，本研究將 JLIRT 分為三大類：第一大類為色調辨識影像檢索 (Image Retrieval by Hue Identification，以下簡稱 IRHI)，IRHI 利用調色盤萃取影像特徵值以進行影像檢索。第二大類為輪廓辨識影像檢索 (Image Retrieval by Contour Identification，以下簡稱 IRCI)，IRCI 利用像素相對值將影像轉成相對輪廓以進行影像檢索。第三大類為色調暨輪廓辨識影像檢索 (Image Retrieval by Hue and Contour Identification，以下簡稱 IRHCI)，IRHCI 各取 IRHI 與 IRCI 兩套影像檢索演算法之優點而加以合併。

以下 F 代表 JLIRT，M 代表 JLIRT 之函式(JLIRT Function)。M 代表輸入欲進行檢索之影像，F(M)代表經過 JLIRT 運算後所得之輸出結果，即為最相似之影像群 G，則  $F(M)=G$ 。如圖 3.1.1 所示：

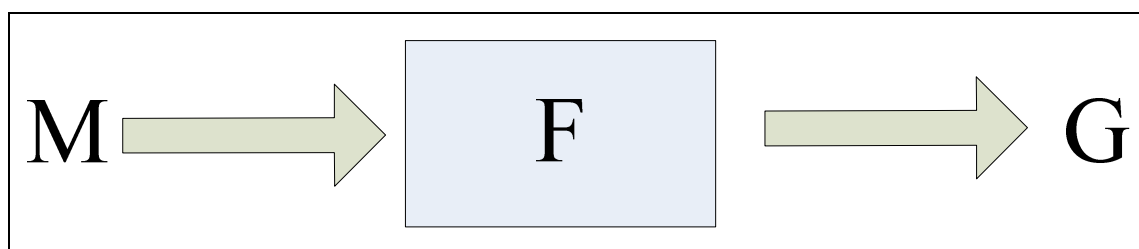


圖 3.1.1 JL 影像檢索技術

## 二、IRHI 色調辨識影像檢索

### 1. 萃取影像身分證演算法

爲了達到最小檢索時間及最大成功率，本研究設計出萃取影像身分證演算法 (Distill Image Identification Algorithm，以下簡稱 DIIA)，利用 DIIA，可將一張影像轉換爲由 16 個特徵值所組成之影像身分證(Image Identification)。

首先，將一張影像  $M$  分割爲 16 個區塊  $m_1$ 、 $m_2$ 、 $m_3$ 、 $m_4$ …… $m_{16}$ ，如圖 3.2.1.1 所示：

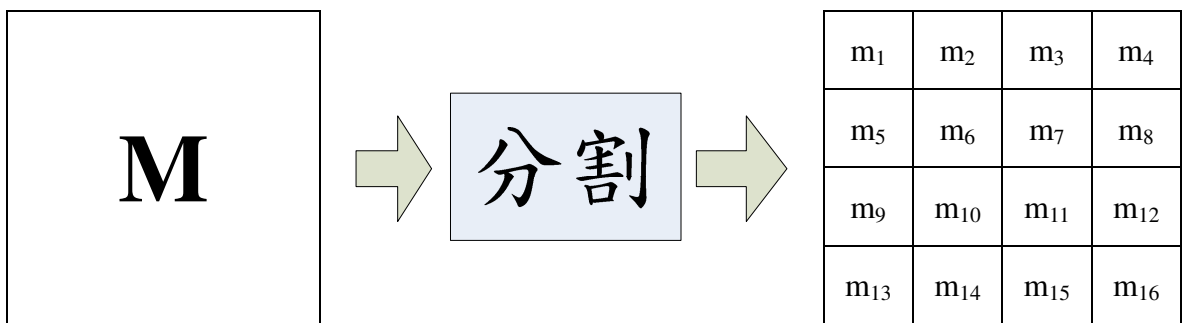


圖 3.2.1.1 影像分割

再利用色彩分析函式(Color Analysis Function，以下簡稱 CAF)分別計算出每個影像區塊出現色彩頻率最高之像素索引值(Index For Pixel)，而定義此像素索引值爲此區塊之影像特徵值，分別爲  $C_1$ 、 $C_2$ 、 $C_3$ 、 $C_4$ …… $C_{16}$ ，如圖 3.2.1.2 及圖 3.2.1.3 所示：

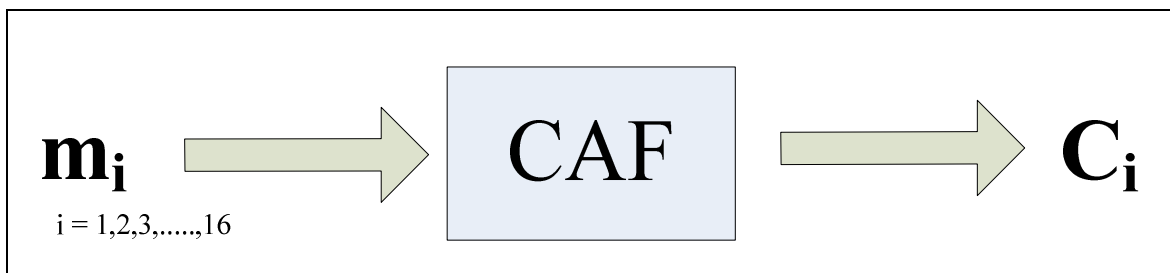


圖 3.2.1.2 CAF 函式

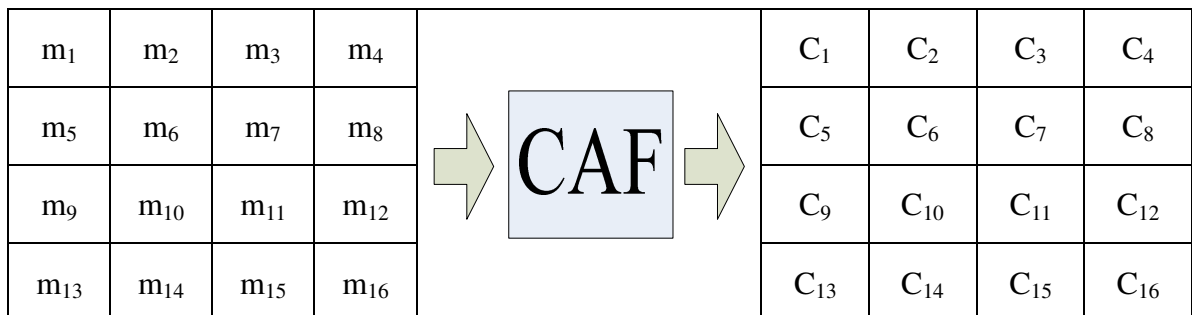


圖 3.2.1.3 CAF 函式

最後，將所產生之 16 組影像特徵值組合成此張影像之影像身分證  $I = \{ C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}, C_{15}, C_{16} \}$ ，如表 3.2.1.1 所示：

表 3.2.1.1 影像身分證

M															
$m_1$	$m_2$	$m_3$	$m_4$	$m_5$	$m_6$	$m_7$	$m_8$	$m_9$	$m_{10}$	$m_{11}$	$m_{12}$	$m_{16}$	$m_{14}$	$m_{15}$	$m_{16}$
$C_1$	$C_2$	$C_3$	$C_4$	$C_5$	$C_6$	$C_7$	$C_8$	$C_9$	$C_{10}$	$C_{11}$	$C_{12}$	$C_{13}$	$C_{14}$	$C_{15}$	$C_{16}$
$I = \{ C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}, C_{15}, C_{16} \}$															

## 2. 建立影像身分證資料庫

利用 DIIA 將影像資料庫中  $n$  張影像  $M_1, M_2, \dots, M_n$  皆轉換為相對應之影像身分證  $I_{d1}, I_{d2}, \dots, I_{dn}$  以建立影像身分證資料庫，如圖 3.2.2.1 及圖 3.2.2.2 所示：



圖 3.2.2.1 建立影像身分證資料庫

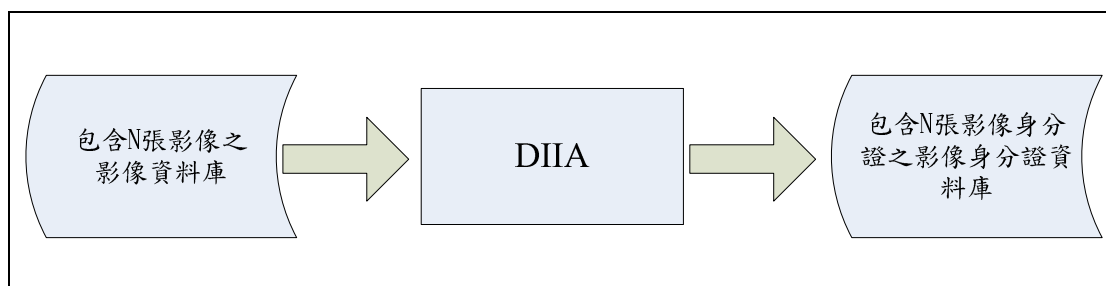


圖 3.2.2.2 建立影像身分證資料庫

### 3. 計算相似分數

爲了辨識一張影像與影像資料庫中影像之相似度，我們設計出相似分數 (Similarity Mark，以下簡稱 SM)，SM 係先計算出相對像素之間的幾何距離，再將所有距離加總起來，以茲判斷該張影像與影像資料庫中影像之相似度；若數值越大，則表示兩張影像相似度越小，反之則相似度越大。

欲進行辨識之影像  $M_p$  經過 DIIA 轉換後得到一組影像身分證  $I_p = \{C_{p1}, C_{p2}, C_{p3}, C_{p4}, C_{p5}, C_{p6}, C_{p7}, C_{p8}, C_{p9}, C_{p10}, C_{p11}, C_{p12}, C_{p13}, C_{p14}, C_{p15}, C_{p16}\}$ ，接著計算此張影像與影像資料庫中第一張影像之 SM：首先，從影像身分證資料庫中讀取第一組影像身分證  $I_{d1} = \{C_{d11}, C_{d12}, C_{d13}, C_{d14}, C_{d15}, C_{d16}, C_{d17}, C_{d18}, C_{d19}, C_{d110}, C_{d111}, C_{d112}, C_{d113}, C_{d114}, C_{d115}, C_{d116}\}$ ，將  $I_p$  中第一個元素  $C_{p1}$  與  $I_{d1}$  中相對應之第一個元素  $C_{d11}$  相減並取其絕對值  $A_1$ ，接著將  $I_p$  中第二個元素  $C_{p2}$  與  $I_{d1}$  中相對應之第二個元素  $C_{d12}$  相減並取其絕對值  $A_2$ ，接下來按照此步驟依序取得絕對值  $A_3, A_4, \dots, A_{16}$ ，如下列(3.2.3.1)所示：

$$A_1 = |C_{p1} - C_{d11}|, A_2 = |C_{p2} - C_{d12}|, \dots, A_{16} = |C_{p16} - C_{d116}| \quad (3.2.3.1)$$

最後將此 16 個絕對值相加即得到欲進行比對之影像  $M_p$  與影像資料庫中第一張影像之 SM 即  $SM_1$ ，如下列(3.2.3.2)式所示：

$$\begin{aligned} SM_1 &= A_1 + A_2 + A_3 + \dots + A_{16} \\ &= |C_{p1} - C_{d11}| + |C_{p2} - C_{d12}| + |C_{p3} - C_{d13}| + \dots + |C_{p16} - C_{d116}| \end{aligned} \quad (3.2.3.2)$$

取得影像  $M_p$  之  $SM_1$  後，依照上式之演算法可依序取得  $M_p$  與影像資料庫中其他影像之 SM，則可整合出下列相似分數之公式(3.2.3.3)：

$$SM_{i(i=1,2,\dots,n)} = |C_{pi} - C_{dki}| + |C_{pi} - C_{dki}| + |C_{pi} - C_{dki}| + \dots + |C_{pi} - C_{dki}|_{(k=1,2,\dots,n)} \quad (3.2.3.3)$$

### 4. 排序

經由上述方法取得比對影像  $M_p$  與影像資料庫中  $n$  張影像之所有相似分數  $SM(SM_1, SM_2, \dots, SM_n)$  後，我們利用快速排序(Quick Sort)將所有 SM 由小到大做排序，最後排序結果中數值最小之  $SM_i$  所代表的影像  $i$  即爲影像資料庫所有影像中與比對影像  $M_p$  最相似者。而排序結果中其他依序從第二小到最大的 SM 所代表之影像，則分別爲影像資料庫中與比對影像  $M_p$  第二相似到最不相似者。

最後依使用者要求列出最相似之影像群組 G，如圖 3.2.4.1 所示：

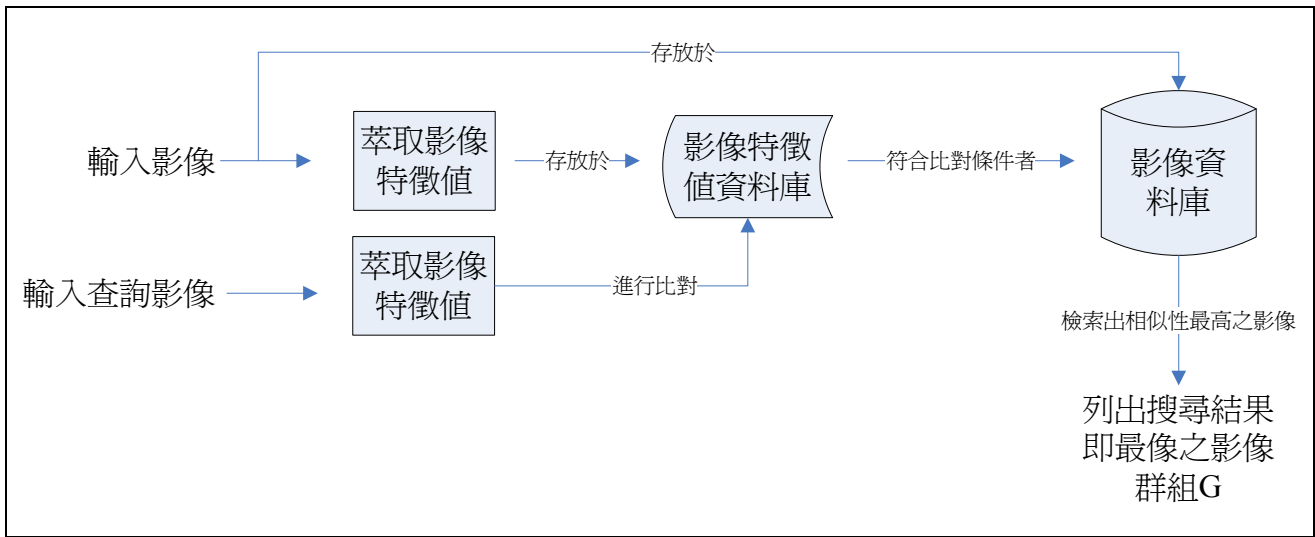


圖 3.2.4.1 IRHI 檢索流程

### 三、IRCI 輪廓辨識影像檢索

#### 1. 萃取影像輪廓演算法

由於本研究秉持著縮小影像檢索時間的基本精神，在如何將一張影像轉換成輪廓這一部份，若直接從彩色影像 RGB 值下手處理，運算量會過於龐大而拖慢執行速度。

因此我們做實驗將同樣的一張影像分別以(1)全彩 (2)256 色 (3)16 色 (4)8bit 灰階 (5)3bit 灰階五種色調模式表示，然後再個別轉成輪廓，結果顯示同一張影像由這五種色調模式所轉出的輪廓幾乎沒有差別！

而以運算速度而言，3bit 灰階影像轉輪廓的時間顯然最短，故最後我們選擇先將欲萃取輪廓之影像，利用影像繪圖程式轉為 3bit 灰階，再利用我們所設計出的萃取影像輪廓演算法(Distill Image Contour Algorithm，以下簡稱 DICA)萃取出影像輪廓。利用 DICA，可將 3bit 灰階影像轉換成由數值“0”與“1”所代表之影像輪廓(Image Contour)，其中 1 代表黑色，0 代表白色。

首先，利用影像繪圖程式將一張  $n \times n$  維之影像 M 轉為 3bit 灰階格式。接著利用索引值讀取函式(Index Catching Function，以下簡稱 ICF)讀取此張灰階影像之所有像素索引值，如圖 3.3.1.1 所示：

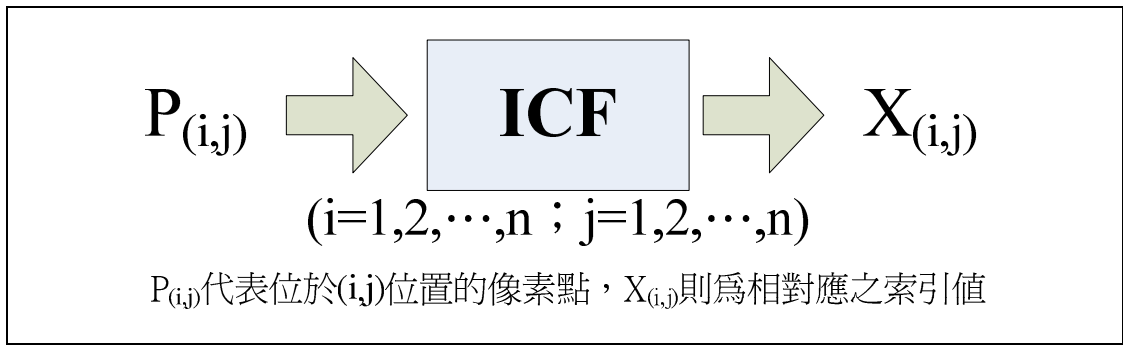


圖 3.3.1.1 ICF 函式

取得  $n \times n$  個索引值  $X$ (從  $X_{(1,1)}$  到  $X_{(n,n)}$ )後，將這  $n \times n$  個  $X$  全部加總起來，再除以  $n \times n$ ，則得到一個色調平均值  $J$ ，如圖 3.3.1.2 所示：

$$J = \frac{X_{(1,1)} + X_{(1,2)} + \dots + X_{(1,n)} + X_{(2,1)} + X_{(2,2)} + \dots + X_{(2,n)} + \dots + X_{(n,n)}}{n \times n}$$

圖 3.3.1.2 色調平均值  $J$

下一步將  $n \times n$  個  $X$ (從  $X_{(1,1)}$  到  $X_{(n,n)}$ )分別與  $J$  做比較，若  $X_{(i,j)}$ 大於  $J$ ，則將  $X_{(i,j)}$ 所對應的像素點 $(i,j)$ 指定為黑色、索引值設定為 1，反之若  $X_{(i,j)}$ 小於  $J$ ，則將  $X_{(i,j)}$ 所對應之像素點 $(i,j)$ 指定為白色、索引值設定為 0，如圖 3.3.1.3 所示：

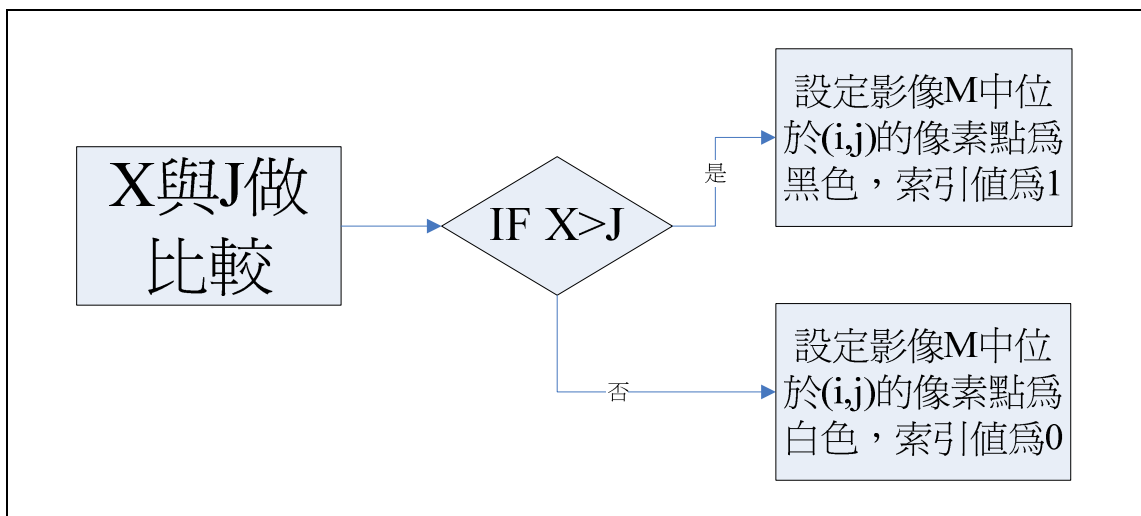


圖 3.3.1.3 判斷流程

最後，產生一個由數值“0”與“1”所構成之 2 維陣列 T，即稱 T 為影像 M 之影像輪廓，如圖 3.3.1.4 所示：

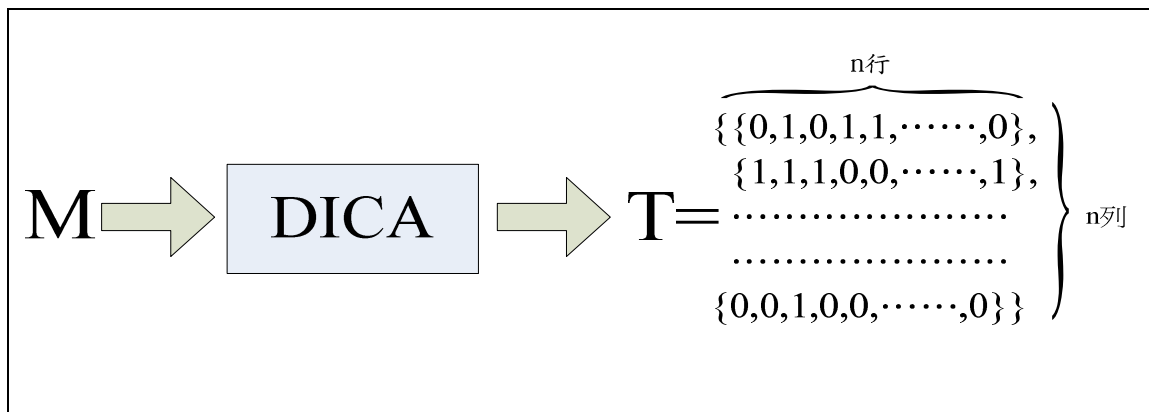


圖 3.3.1.4 DICA 函式與影像輪廓 T

## 2. 建立影像輪廓資料庫

利用 DICA 將影像資料庫中  $n$  張影像  $M_1$ 、 $M_2$ ..... $M_n$  皆轉換為相對應之影像輪廓  $T_{d1}$ 、 $T_{d2}$ .....  $T_{dn}$  以建立影像身分證資料庫，如圖 3.3.2.1 及圖 3.3.2.2 所示：

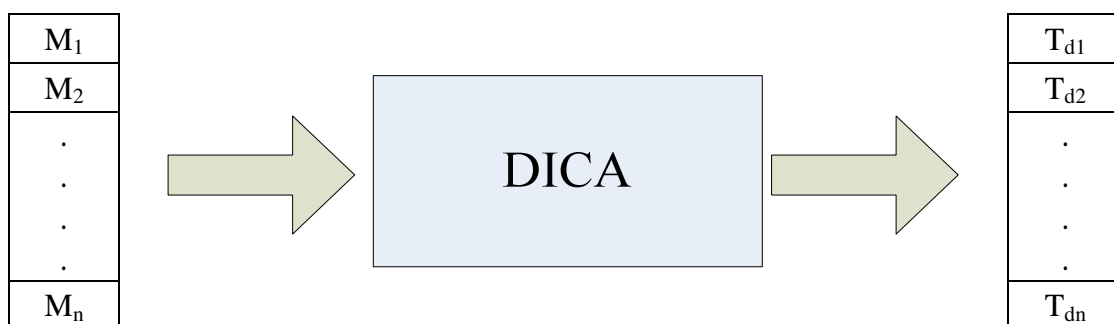


圖 3.3.2.1 建立影像身分證資料庫

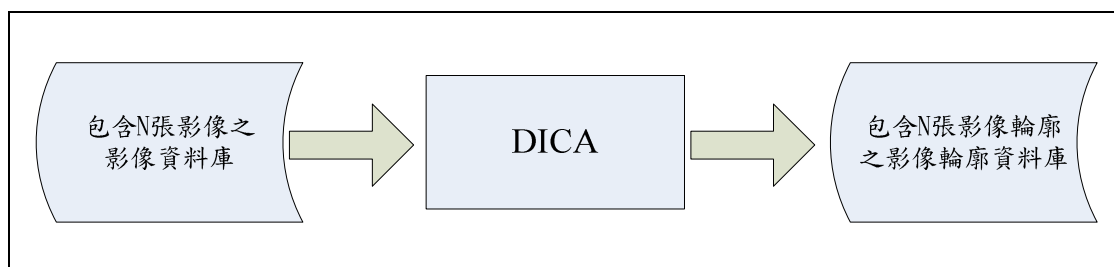


圖 3.3.2.2 建立影像身分證資料庫

### 3. 計算相似分數

欲進行辨識之影像  $M_p$  (設大小為  $n \times n$ ) 經過 DICA 轉換後得到一組影像輪廓  $T_p = \{T_{p1}, T_{p2}, T_{p3}, \dots, T_{p(n \times n)}\}$ ， $T_p$  為由 0 與 1 所組成之 2 維陣列，共含  $n \times n$  個數值；接著計算此張影像與影像資料庫中第一張影像之 SM：首先，從影像輪廓資料庫中讀取第一組影像輪廓  $T_{d1} = \{T_{d11}, T_{d12}, T_{d13}, \dots, T_{d1(n \times n)}\}$ ，將  $T_p$  中第一個元素  $T_{p1}$  (可能為 0 或 1) 與  $T_{d1}$  中相對應之第一個元素  $T_{d11}$  (可能為 0 或 1) 相減並取其絕對值  $A_1$ ，接著將  $T_p$  中第二個元素  $T_{p2}$  (可能為 0 或 1) 與  $T_{d1}$  中相對應之第二個元素  $T_{d12}$  (可能為 0 或 1) 相減並取其絕對值  $A_2$ ，接下來按照此步驟依序取得絕對值  $A_3, A_4, \dots, A_{n \times n}$ ，如下列(3.2.3.1)式所示：

$$A_1 = |T_{p1} - T_{d11}|, A_2 = |T_{p2} - T_{d12}|, \dots, A_{n \times n} = |T_{p(n \times n)} - T_{d1(n \times n)}| \quad (3.3.3.1)$$

最後將此  $n \times n$  個絕對值相加即得到欲進行比對之影像  $M_p$  與影像資料庫中第一張影像之 SM 即  $SM_1$ ，如下列(3.2.3.2)式所示：

$$SM_1 = A_1 + A_2 + A_3 + \dots + A_{n \times n} \quad (3.3.3.2)$$

取得影像  $M_p$  之  $SM_1$  後，依照上式之演算法可依序取得  $M_p$  與影像資料庫中其他影像之 SM，則可整合出下列相似分數之公式(3.3.3.2)：

$$SM_{i(i=1,2,\dots,n)} = |T_{pi} - T_{dki}| + |T_{pi} - T_{dki}| + |T_{pi} - T_{dki}| + \dots + |T_{pi} - T_{dki}|_{(k=1,2,\dots,n \times n)} \quad (3.3.3.3)$$

### 4. 排序

經由上述方法取得比對影像  $M_p$  與影像資料庫中  $n$  張影像之所有相似分數  $SM(SM_1, SM_2, \dots, SM_n)$  後，我們利用快速排序(Quick Sort)將所有 SM 由小到大做排序，最後排序結果中數值最小之  $SM_i$  所代表的影像  $i$  即為影像資料庫所有影像中與比對影像  $M_p$  最相似者。而排序結果中其他依序從第二小到最大的 SM 所代表之影像，則分別為影像資料庫中與比對影像  $M_p$  第二相似到最不相似者。

最後依使用者要求列出最相似之影像群組  $G$ ，如圖 3.3.4.1 所示：

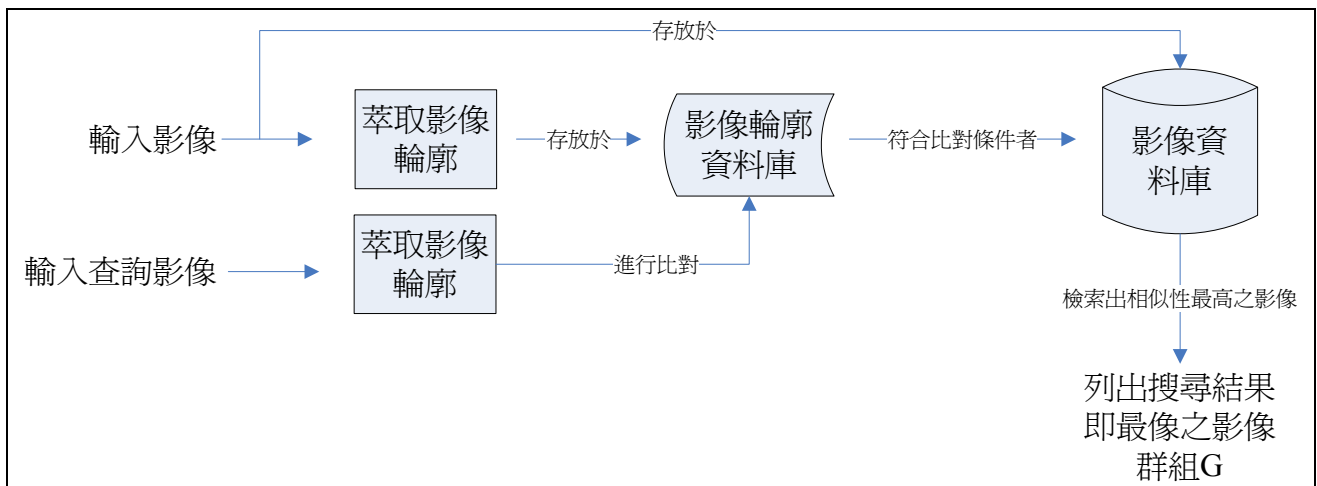


圖 3.3.4.1 IRCI 檢索流程

#### 四、IRHCI 色調暨輪廓辨識影像檢索

我們研究將 IRHI 與 IRCI 兩套影像檢索演算法合併，各取優點而設計出色調暨輪廓辨識影像檢索 (Image Retrieval by Hue plus Contour Identification，以下簡稱 IRHCI)。

##### 1. IRHCI

首先，我們輸入欲進行比對影像  $M$ ，利用 IRHI 從影像資料庫中檢索出與  $M$  最為相似之影像群組  $G_1$ ，如圖 3.4.1.1 所示：



圖 3.4.1.1 IRHI 產生  $G_1$

接著，再利用 IRCI 從影像資料庫中檢索出與  $M$  最為相似之影像群組  $G_2$ ，如圖 3.4.1.2 所示：



圖 3.4.1.2 IRCI 產生  $G_2$

最後，從  $G_1$  與  $G_2$  中選取出 IRHI 與 IRCI 皆有尋找到之影像，形成另一新影像群組  $G_3$ 。因此  $G_3$  可能含有  $a$  ( $a$  為小於或等於 200 之任一正整數) 張影像，並同時含有  $a$  個各自與影像  $M$  的  $SM_{G_{3n}}$  ( $1 < n < a$ )，而每個  $SM_{G_{3n}}$  分別為在  $G_1$  中之  $SM_{G_{1x}}$  ( $x$  為小於 200 之任一整數) 與  $G_2$  中之  $SM_{G_{2x}}$  相加再除以 2 所得之，如下列式 3.4.1.1 及圖 3.4.1.3 所示：

$$SM_{G_{3n}} (1 < n < a) = (SM_{G_{1x}} + SM_{G_{2x}}) / 2 \quad (3.4.1.1)$$

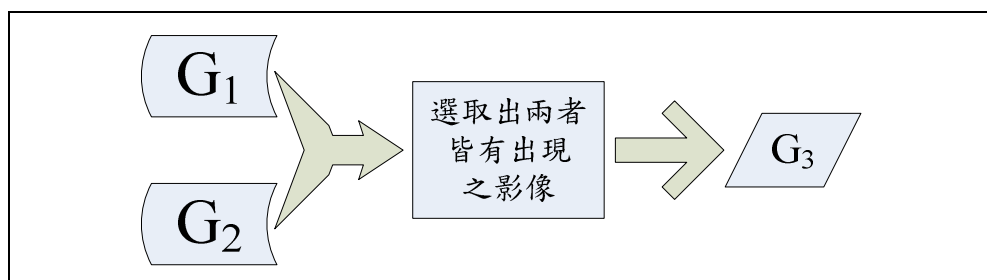


圖 3.4.1.3 尋找出  $G_1$  與  $G_2$  中皆有出現之影像

## 2. 排序

經由上述方法取得比對影像  $M$  與  $G_3$  中  $a$  張影像之所有相似分數  $SM_{G_3n(1 < n < a)}$  後，我們利用快速排序(Quick Sort)將所有  $SM$  由小到大做排序，最後排序結果中數值最小之  $SM_i$  所代表的影像  $i$  即為影像資料庫所有影像中與比對影像  $M$  最相似者。而排序結果中其他依序從第二小到最大的  $SM$  所代表之影像，則分別為影像資料庫中與比對影像  $M$  第二相似到最不相似者。最後依使用者要求列出最相似之影像群組  $G$ ，如圖 3.3.2.1 所示：

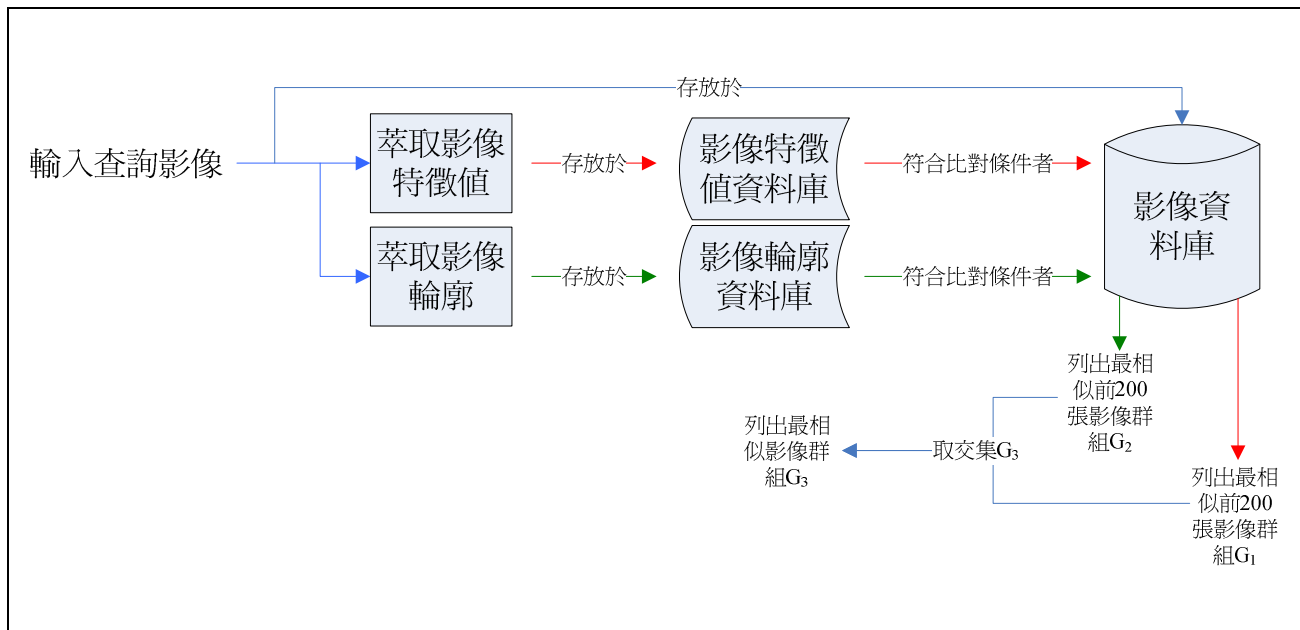


圖 3.4.2.1 IRHCI 檢索流程

## 肆、研究過程及比較

在本研究中，我們使用 PHP 程式語言撰寫 IRHI、IRCI 與 IRHCI，以下是我們的實驗平台：

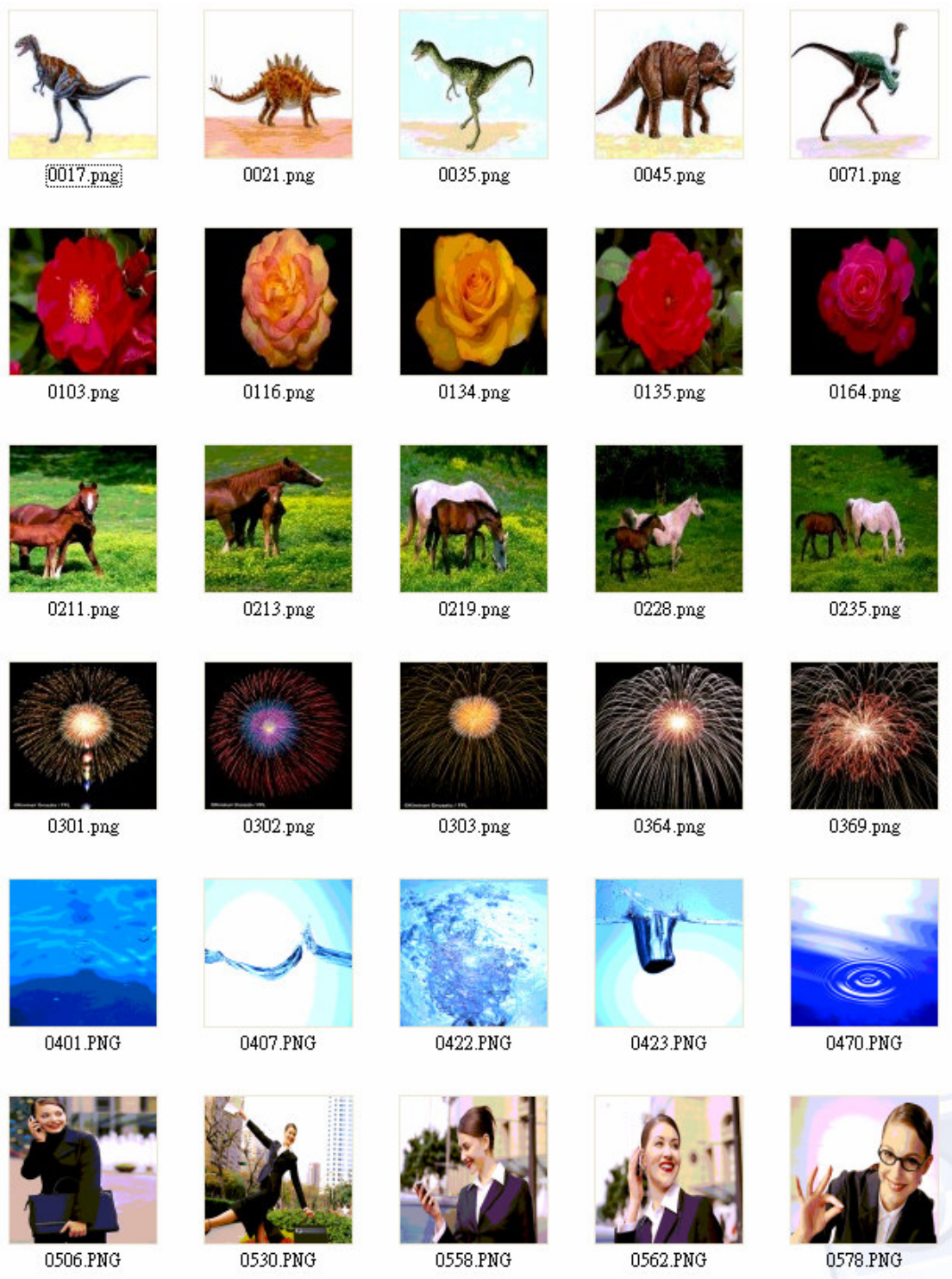
實驗平台：

<b>CPU</b>	<b>Intel Pentium 4 3.0G (FSB 800 Hz, 200 外頻)</b>
<b>RAM</b>	<b>512MB × 4</b>
<b>GPU</b>	<b>NVIDIA GeForce 6600</b>
<b>HDD</b>	<b>Seagate 7200.8 200G</b>
<b>OS</b>	<b>Microsoft Windows XP sp2</b>
<b>伺服器端</b>	<b>AppServ v2.5.4a</b>

## 一、圖片分類

在本篇研究報告中，我們將所使用之彩色影像資料庫分為六大類，每類群有 100 張影像，共 600 張影像，其依序為：

1.恐龍 2.花 3.馬 4.煙火 5.水紋 6. 人物



〈圖 4.1.1〉六大類影像資料庫中之部分影像

## 二、IRHI 研究過程

### 1. 萃取影像特徵值

首先：我們定義所有進行萃取之影像格式皆為

(1) 標準 256 色

(3) PNG 圖片格式

如此即可避免因不同影像格式而造成比對時之不預期錯誤。

第二：將每張影像分割成  $4 \times 4$  個區塊。

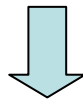
第三：利用 CAF 函式依序由左而右、由上而下讀取 16 區塊中所有單獨像素之索引值，並同時計算出每一區塊中使用次數最多之顏色的索引值，而此值即為此一區塊之特徵值。

第四：重複第三步驟直至 16 個區塊皆找出其各自所代表之特徵值，接著我們將這 16 個特徵值依序排成一個一維陣列，並稱之為此張影像的影像身分證。

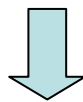
以下為萃取影像特徵值之示意圖（圖 4.2.1.1）。



原圖



將影像轉成  
1.標準256色  
2.PNG圖片格式



將每張影像分割成4 × 4個區塊。



256色色盤

利用ICF及RBG函式依序由左而右、由上而下讀取16區塊中所有單獨像素之RGB值，並同時計算出每一區塊中使用次數最多之RGB的索引值，而此值即為此一區塊之特徵值。



<b>232</b>	<b>232</b>	<b>230</b>	<b>78</b>
<b>232</b>	<b>232</b>	<b>64</b>	<b>230</b>
<b>253</b>	<b>250</b>	<b>232</b>	<b>230</b>
<b>232</b>	<b>232</b>	<b>230</b>	<b>230</b>

特徵值

重複第三步驟直至 16 個區塊皆找出其各自所代表之特徵值，接著我們將這 16 個特徵值依序排成一個一維陣列，並稱之為此張影像的影像身分證。

圖 4.2.1.1 萃取影像特徵值之流程圖

## 2. 建立影像身分證資料庫

利用上述之萃取影像特徵值演算法(DIIA)將影像資料庫中  $n$  張影像  $M_1, M_2, \dots, M_n$  皆轉換為相對應之影像身分證  $I_{d1}, I_{d2}, \dots, I_{dn}$  以建立影像身分證資料庫。

## 3. 與影像資料庫進行比對

取得欲進行比對影像之影像身分證後，將之與影像身分證資料庫中 600 張影像身分證分別進行計算比對，比對方法為：

第一：欲比對影像身分證中第一個特徵值與影像身分證資料庫中第一張影像身分證第一個特徵值相減，取其絕對值；再來比對影像身分證第二個特徵值與影像身分證資料庫中第一張影像第二特徵值相減，亦取其絕對值……，接著依此類推，比對影像身分證中剩下的 14 個特徵值，依序與影像身分證資料庫中第一張影像剩下的 14 個特徵值相減並各取其絕對值，最後把取得之 16 個絕對值相加，此數值即為比對影像是否與影像資料庫中第一張影像相似之依據(即為 SM)。

第二：取得比對影像與影像資料庫中第一張影像之 SM 後，接著比對影像與影像身分證資料庫中第二個影像身分證進行同步驟一之運算，取得比對影像與影像資料庫中第二張影像之 SM……，依此類推讓比對影像取得與影像資料庫中 600 張影像各自的 SM。

第三：取得比對影像與影像資料庫中 600 張影像各自的 SM 後，將此 600 個 SM 利用快速排序法進行由小而大的排序，而排序結果第一項必為 0，即為比對影像本身，最後再依使用者要求列出最為相似之影像，即完成整個比對過程。


以下為與彩色影像資料庫進行比對之示意圖（圖 4.2.3.1）。

依序讀入影像身分證



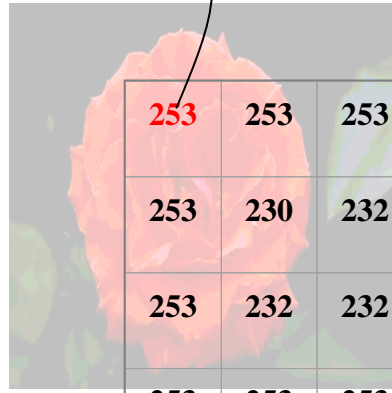
兩數值相減再取絕對值

$$A_1 = |232 - 253|$$



<b>232</b>	232	230	78
232	232	64	230
253	250	232	230
232	232	230	230

欲比對圖像之影像身分證




<b>253</b>	253	253	253
253	230	232	254
253	232	232	253
253	253	253	253

讀入之第一張影像身分證

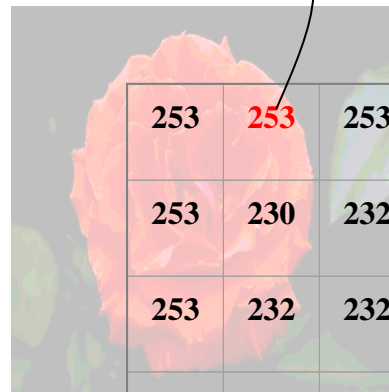


兩數值相減再取絕對值

$$A_2 = |232 - 253|$$

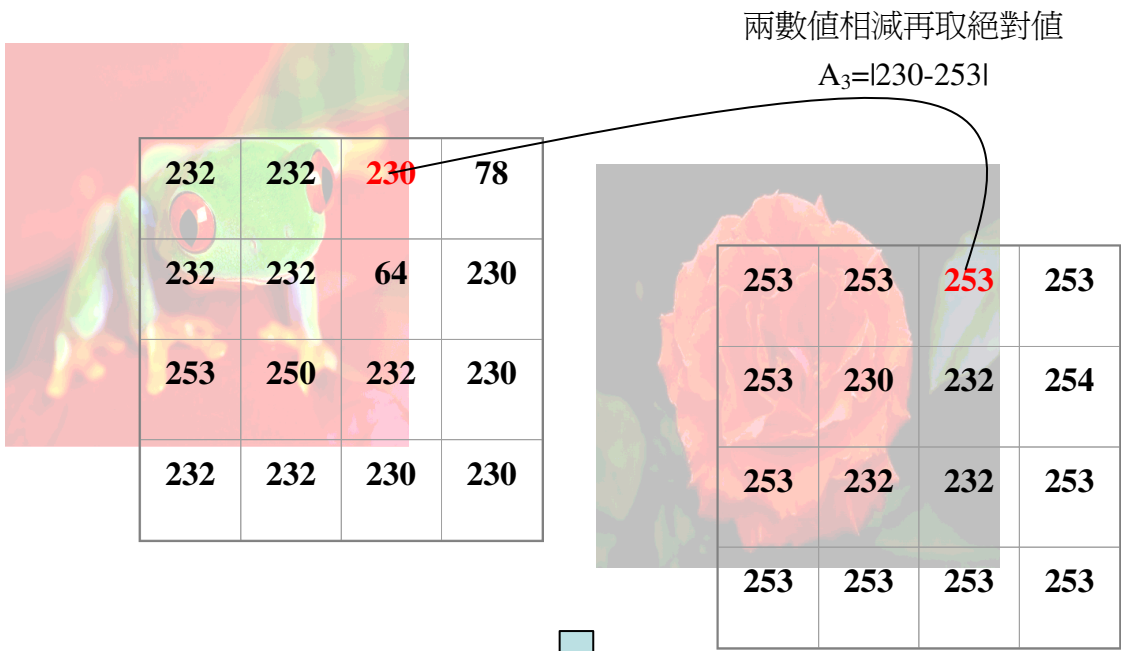


232	<b>232</b>	230	78
232	232	64	230
253	250	232	230
232	232	230	230



253	<b>253</b>	253	253
253	230	232	254
253	232	232	253
253	253	253	253

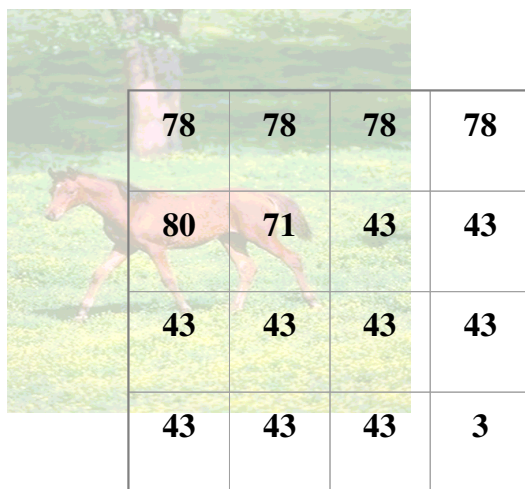


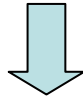


依此類推，最後再把所有的絕對值相加起來，即為此影像與影像資料庫中影像之SM，  
 即  $SM = A_1 + A_2 + \dots + A_{16}$

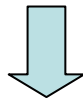


讀入第二張影像身分證





依照同樣的方法找出所有SM



彩色影像檢索

研究動機 流程展示 實例展示 未來展望

瀏覽...

列出前 20 項

- 色調比對
- 輪廓比對
- 色調+輪廓比對

開始上傳

第1像:322  
分數:0

第11像:372  
分數:4

http://127.0.0.1 - 322.jpg (JPEG 圖片, 650x348 像素) - Mozilla Firefox

<b>Name</b>	Coelophysis
<b>Meaning</b>	hollow form
<b>How to say it</b>	see-oh-fee-sis
<b>Where found</b>	USA
<b>Length</b>	3 m
<b>Height</b>	2 m
<b>Mass (weight)</b>	27 kg
<b>How it moved</b>	on 2 legs
<b>Teeth</b>	small and sharp
<b>Type of feeder</b>	CARNIVOROUS
<b>Food</b>	vertebrates
<b>Period</b>	Upper Triassic
<b>When it lived</b>	225-220 mya
<b>Type of hip</b>	lizard-hipped
<b>Dinosaur group</b>	Theropoda
<b>Other info</b>	One of the earliest known

將所有SM做排序後  
列出搜尋結果

圖4.2.3.1 與影像資料庫進行比對

### 三、IRCI 研究過程

在 IRHI 的實驗中，發現某些特定類群影像，利用以色調為基礎的演算法來做檢索辨識的成功率並非很高；我們曾經嘗試過將程式更改為以 512 色為色盤、甚至以全彩為色盤，或將演算法調整成許多不同型態，例如取整張影像的像素平均值、取中央像素平均、做分區加權……等等，卻始終無法很有效率的提高這些特定類群影像的檢索成功率。

為了解決此一惱人問題並提高整體成功率，我們著手研究這些特定類群影像的性質，希望能找出它們共同的特性，然後就這些特性加以改進我們程式。最後我們發現這些影像都有某些共同特徵：

- A. 非影像主體的背景部份，色調皆紊亂不規律，即使是相同類群影像亦無規則可循。
- B. 影像主體部分，雖屬於同類群影像，卻分成許多不同種色調。
- C. 影像主體部分的輪廓大致相似。

綜合以上特徵，我們以為應該是 A 與 B 兩點造成以色調為基礎的檢索成功率不高，而針對這一部份，很明顯地單純調整色盤或演算法是無法解決此方面的問題，因此我們利用 C 特徵，專門為這一特定影像類群設計出一種新的影像檢索系統：輪廓辨識影像檢索(Image Retrieval by Contour Identification，簡稱 IRCI)，IRCI 利用像素相對值將影像轉成相對輪廓以進行影像檢索。

#### 1. 萃取影像輪廓

首先：我們定義所有進行萃取之影像格式皆為

- (1) 3bit 灰階
- (3) PNG 圖片格式

如此即可避免因不同影像格式而造成比對時之不預期錯誤。

第二：利用 ICF 讀取此張灰階影像之所有索引值

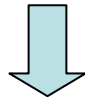
第三：取得  $n \times n$  個索引值  $X$ (從  $X_{(1,1)}$  到  $X_{(n,n)}$ )後，將這  $n \times n$  個  $X$  全部加總起來，再除以  $n \times n$ ，則得到一個判斷分數  $J$ 。

第四：將  $n \times n$  個  $X$ (從  $X_{(1,1)}$  到  $X_{(n,n)}$ )分別與  $J$  做比較，若  $X_{(i,j)}$  大於  $J$ ，則將  $X_{(i,j)}$  所對應的像素點  $(i,j)$  指定為黑色、索引值設定為 1，反之若  $X_{(i,j)}$  小於  $J$ ，則將  $X_{(i,j)}$  所對應之像素點  $(i,j)$  指定為白色、索引值設定為 0。

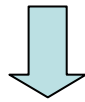
以下為萃取影像輪廓之示意圖 (圖 4.3.1.1)。



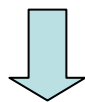
原圖

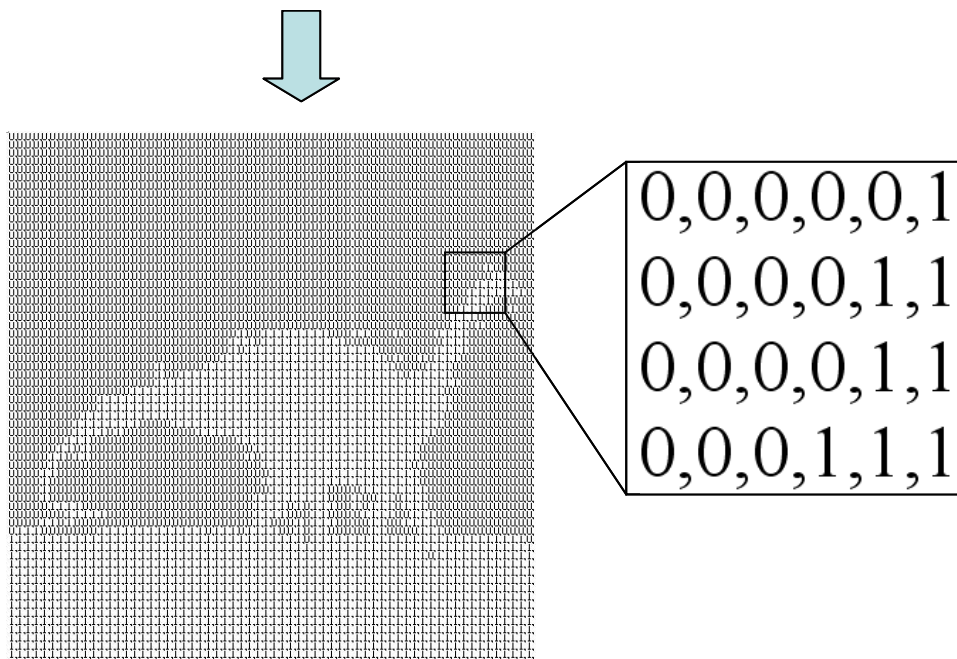


影像轉成3bit灰階



取得 $n \times n$ 個索引值後，將這些索引值全部加總起來，再除以 $n \times n$ ，則得到一個色調平均值 $J$





將全部索引值分別與J做比較，若大於J，則將所對應的像素點(i,j)指定為黑色(數字1)，反之若小於J，則將所對應之像素點(i,j)指定為白色(數字0)。

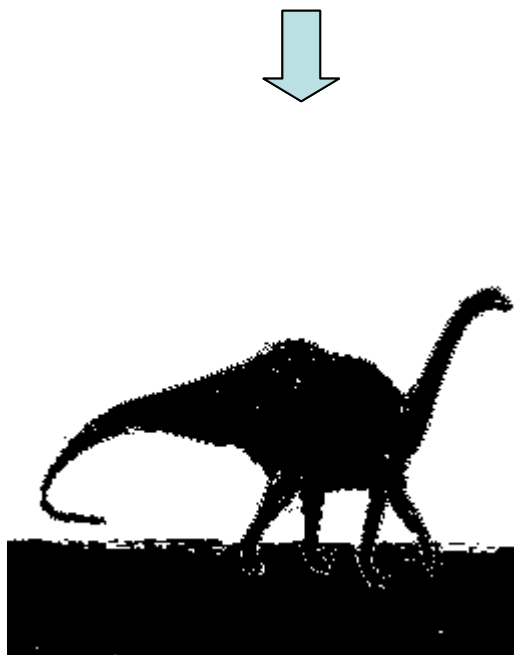


圖4.3.1.1 萃取影像輪廓

## 2. 建立影像輪廓資料庫

利用上述之萃取影像輪廓演算法(DICA)將影像資料庫中  $n$  張影像  $M_1、M_2……M_n$  皆轉換為相對應之影像輪廓  $T_{d1}、T_{d2}……T_{dn}$  以建立影像輪廓資料庫。

## 3. 與影像資料庫進行比對

取得欲進行比對影像之影像輪廓後，將之與影像輪廓資料庫中 600 張影像輪廓分別進行計算比對，比對方法為：

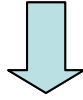
第一：欲比對影像輪廓中第一個數值與影像輪廓資料庫中第一張影像輪廓之第一個數值相減，取其絕對值；再來比對影像輪廓第二個數值與影像輪廓資料庫中第一張影像第二數值相減，亦取其絕對值……，接著依此類推，比對影像身分證中剩下的  $n \times n - 2$  個數值，依序與影像輪廓資料庫中第一張影像剩下的  $n \times n - 2$  個數值相減並各取其絕對值，最後把取得之  $n \times n$  個絕對值相加，此數值即為比對影像是否與影像資料庫中第一張影像相似之依據(即為 SM)。

第二：取得比對影像與影像資料庫中第一張影像之 SM 後，接著比對影像與影像輪廓資料庫中第二個影像輪廓進行同步驟一之運算，取得比對影像與影像資料庫中第二張影像之 SM……，依此類推讓比對影像取得與影像資料庫中 600 張影像各自的 SM。

第三：取得比對影像與影像資料庫中 600 張影像各自的 SM 後，將此 600 個 SM 利用快速排序法進行由小而大的排序，而排序結果第一項必為 0，即為比對影像本身，最後再依使用者要求列出最為相似之影像，即完成整個比對過程。

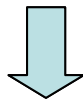
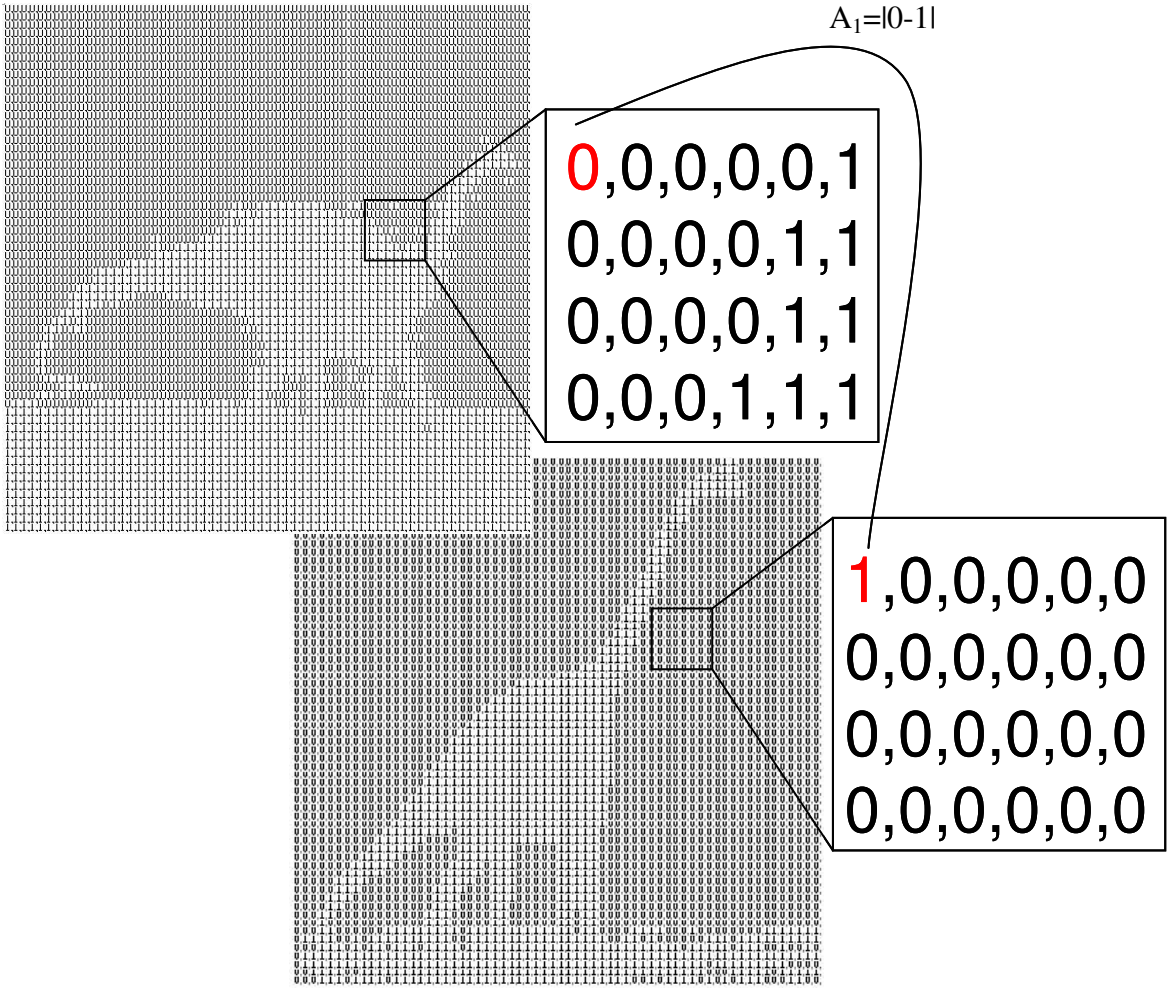
以下為與影像資料庫進行比對之示意圖（圖 4.3.3.1）。

讀入影像輪廓



兩數值相減再取絕對值

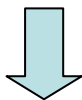
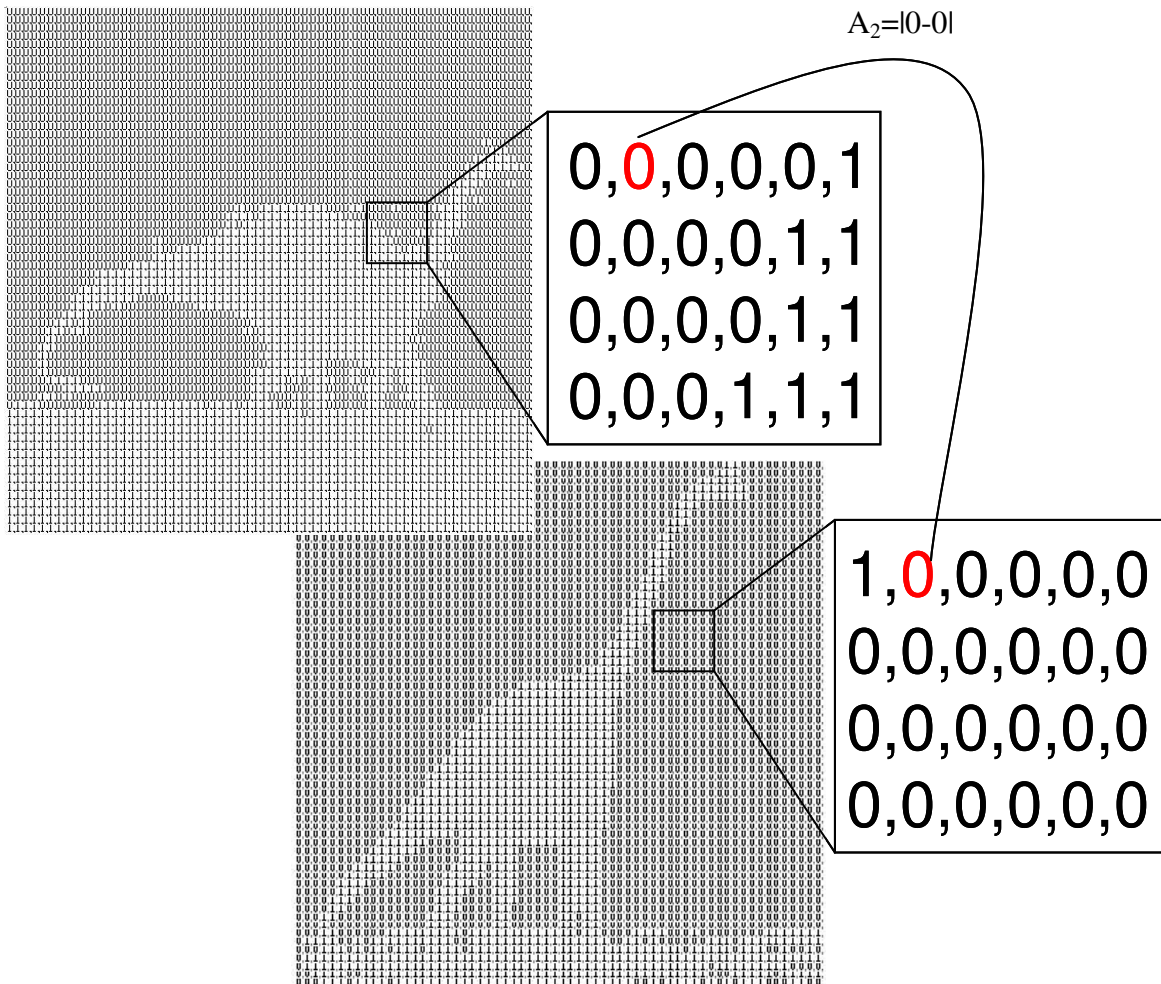
$$A_1 = |0 - 1|$$





兩數值相減再取絕對值

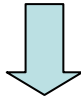
$$A_2 = |I - O|$$



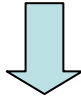
依此類推，最後再把所有的絕對值相加起來，即為此影像與影像資料庫中影像之SM



取得比對影像與影像資料庫中600張影像各自的SM後，將此600個SM利用快速排序法進行由小而大的排序



依照同樣的方法找出所有SM



彩色影像檢索技術

研究動機 流程展示 實例展示 未來展望

C:\App\Server\www\pic0306.png 瀏覽...

列出前 20 項

- 色調比對
- 輪廓比對
- 色調+輪廓比對

開始上傳

---

彩色影像檢

研究動機 流程展示 實例展示 未來展望

Image ID	Score
第4像:400	分數:797
第5像:18	分數:814
第14像:400	分數:797
第15像:18	分數:814

將所有SM做排序後  
列出搜尋結果

圖4.3.3.1 與影像資料庫進行比對

## 四、IRHCI 研究過程

原始設計出 IRCI 的理念是針對改進 IRHI 中檢索成功率不高的類群而來，但最後 IRCI 實驗結果卻沒有預期中的好。經過多次討論與改進，IRCI 檢索成功率依然不是很高，其中某些類群的成功率甚至比 IRHI 低。有鑑於此，我們研究將 IRHI 與 IRCI 兩套影像檢索演算法合併，各取優點而設計出 IRHCI。以下為實驗流程：

1. 利用前述之 IRHI 萃取影像特徵值。
2. 利用影像特徵值透過 IRHI 與影像資料庫進行比對，找出 G1。
3. 利用前述之 IRCI 萃取影像輪廓。
4. 利用影像輪廓透過 IRCI 與影像資料庫進行比對，找出 G2。
5. 找出 G1 與 G2 之交集，即為 G3。
6. 利用快速排序法將 G3 之所有 SM 進行排序。

經由上述方法取得比對影像 M 與 G<sub>3</sub> 中影像之所有相似分數 SM 後，我們利用快速排序(Quick Sort)將所有 SM 由小到大做排序，最後排序結果中數值最小之 SM<sub>b</sub> 所代表的影像 b 即為影像資料庫所有影像中與比對影像 M 最相似者。而排序結果中其他依序從第二小到最大的 SM 所代表之影像，則分別為影像資料庫中與比對影像 M 第二相似到最不相似者。

## 五、實驗結果與比較

爲了比較出所設計之檢索系統能否精確搜尋出相似影像，我們定義一個比較數值：

$$\text{檢索成功率} = \frac{\text{回傳影像群中找到同類群之影像張數}}{\text{回傳給使用者之影像張數}} \times 100\%$$

利用前述方法設計出來的程式和處理流程，經由多次的實驗和偵錯後，實驗的結果所得數據如下所示：

### 1. IRHI 實驗結果

表 4.5.1.1 爲 IRHI 檢索系統對各種類群之檢索成功率

表 4.5.1.1 IRHI 檢索系統對各種類群之檢索成功率

類群	檢索出同一張成功率	前二張檢索出同類群成功率	前三張檢索出同類群成功率	前四張檢索出同類群成功率	前五張檢索出同類群成功率
煙火	100 %	98.5 %	96.7 %	97 %	96.8 %
水紋	100 %	99.5 %	99.3 %	99.3 %	99.2 %
人物	100 %	94.5 %	89 %	85 %	81.8 %
恐龍	100 %	99.5 %	99 %	99.8 %	97 %
花	100 %	88.5 %	81.7 %	71.6 %	74.8 %
馬	100 %	97 %	94.7 %	93 %	92.4 %

## 2. IRCI 實驗結果

表 4.5.2.1 為 IRCI 檢索系統對各種類群之檢索成功率

表 4.5.2.1 IRCI 檢索系統對各種類群之檢索成功率

類群	檢索出同一張成功率	前二張檢 索出同類 群成功率	前三張檢 索出同類 群成功率	前四張檢 索出同類 群成功率	前五張檢 索出同類 群成功率
煙火	100 %	89.5 %	90.3 %	91.3 %	92.2 %
水紋	100 %	89 %	80.3 %	77 %	75 %
人物	100 %	76.5 %	63.7 %	58 %	51.8 %
恐龍	100 %	88 %	83 %	81.5 %	78.4 %
花	100 %	79.5 %	74 %	69 %	66 %
馬	100 %	55.5 %	38 %	29.3 %	23.8 %

### 3. IRHCI 實驗結果

表 4.5.3.1 為 IRHCI 檢索系統對各種類群之檢索成功率

表 4.5.3.1 IRHCI 檢索系統對各種類群之檢索成功率

類群	檢索出同一張成功率	前二張檢索出同類群成功率	前三張檢索出同類群成功率	前四張檢索出同類群成功率	前五張檢索出同類群成功率
煙火	100 %	98 %	97 %	96.8 %	97.2 %
水紋	100 %	99.5 %	99.4 %	99.3%	99 %
人物	100 %	91%	83.4%	79 %	75.4 %
恐龍	100 %	100 %	100 %	98.8 %	98 %
花	100 %	83 %	73.7 %	67 %	63 %
馬	100 %	99 %	97.4 %	95.8 %	94.2 %

### 4. IRHI、IRCI、IRHCI 比較

表 4.5.4.1 為 IRHI、IRCI、IRHCI 比較：

表 4.5.4.1 IRHI、IRCI、IRHCI 比較

檢索系統	檢索速度	適用影像	檢索成功率	優點	缺點
<b>IRHI</b>	最快	色調相似者	高	比對運算量最小，故檢索速度最快	只適用於色調相似影像檢索
<b>IRCI</b>	第三	主體與背景反差大者	低	對於主體明顯之影像，成功率較高	檢索速度稍慢
<b>IRHCI</b>	第二	皆可	中	可廣泛適用於各種影像	只有單純色調明顯或輪廓明顯者，成功率較差

表 4.5.4.2 為 IRHI、IRCI、IRHCI 與灰階影像內容之檢索技術之綜合比較：

表 4.5.4.2 IRHI、IRCI、IRHCI 與灰階影像內容之檢索技術之綜合比較

檢索系統	搜尋出本身影像成功率	平均檢索成功率	優缺點	適用影像
<b>IRHI</b>	100 %	90.3 %	已有完整整合之線上檢索系統	彩色影像
<b>IRCI</b>	100 %	64.6 %		
<b>IRHCI</b>	100 %	87.8 %		
灰階影像內容之檢索技術[4]	99.3 %	74.3 %	實際應用程式尚未做完整之整合	灰階影像

## 伍、討論與分析

### 一、IRHI 討論與分析

由表 4.4.1.1 可知，IRHI 對某些類群的檢索成功率極低，如人物類及花類。探討其原因，主要是由於：

- 1.由於 IRHI 的架構是以色調相近為搜尋依據，又因一般影像主體多在影像中央，故對於中央區域有加權，所以對於主體不在中央之影像，其檢索成功率會下降，希望未來能設計出自動辨識影像主體區域之演算法，進而判斷加權之區域。
- 2.為了達到使用最少記憶體空間來儲存影像身分證之目的，IRHI 採用 256 色之圖像，故會與全彩圖像有所誤差，期望未來能發展出使用全彩圖片，而記憶體使用依舊極低之演算法。
- 3.IRHI 在萃取影像身分證時，為了顧及效率及速度，將影像分割成 4x4 個區塊，故影像中大量之特徵只縮減成 16 個特徵值，期待在不失效率及速度之前提下，仍能將影像分割成更多區塊，以使原本影像中豐富之特徵皆發揮最大用處，如此應可降低失真率。
- 4.從研究過程中得知，由於計算特徵值與比對相似圖片時，計算量過於龐大，造成搜尋時間過長，期許能發展出速度更快的檢索架構，且不失其準確率。

## 二、IRCI 討論與分析

在 IRHI 的實驗中，某些特定類群之檢索成功率一直不是很高，探討其原因，發現這些類群之影像皆有若干共同特徵：

1. 影像背景部分皆雜亂無規律，色調上亦無一致性。
2. 同一類群內，影像主體色調差異度大。

以上兩點可能是造成以色調為基礎之系統 IRHI 對於此類群影像檢索成功率不高之因素。有鑑於此，我們針對此類影像設計出 IRCI。而經過多次實驗及改正，由表 4.4.1.2 可知，IRCI 對某些類群的檢索成功率仍然極低，如人物類及馬類。探討其原因，主要是由於：

1. 某些影像之背景色調與主體色調相近，造成 IRCI 判斷輪廓錯誤，如馬類。
2. 同類群內主體位置不一，即使輪廓判斷正確，疊合比對後 SM 差異極大，期望能發展出自動尋找影像主體位置之演算法，以正確比對。
3. 經由 IRCI 轉換出之輪廓僅一細線，造成比對時條件過於嚴苛，需要適當加入模糊比對的模式。
4. 目前 IRCI 須先將全彩影像降色為灰階方能轉換成輪廓，如此會延長執行時間，並佔用較多記憶體，若能由全彩影像直接轉成輪廓，不但可以節省執行時間，亦可降低伺服器負荷量。

### 三、IRHCI 討論與分析

經過 IRHCI 實驗後，發現對於 IRCI 而言，除了花這個類群之外，其他類群檢索成功率皆有不小之提升，探討其原因，我們歸納出下面幾點因素：

- 1.由於 IRHCI 結合了 IRHI 檢索成功率不受背景色調差異而下降的優勢，及 IRCI 檢索成功率不受主體色調差異而下降的優勢，對於背景及主體色彩不規律之影像，例如馬類，檢索成功率會較高。
- 2.水紋在 IRCI 之架構下檢索成功率不是很高，但經 IRHCI 檢索後成功率卻大幅度提升，探究其原因，應為水紋之色調大致上相似，輪廓卻不盡相同，經由融合了 IRHI 之 IRHCI 執行後，成功率固然會上升。
- 3.由於我們在設計 IRHCI 時，將 IRCI 部分改為模糊比對，故 IRHCI 雖經過 2 次運算及 3 次比對，執行速度並無明顯提升，甚至略比 IRCI 快。
- 4.由於花類群的輪廓與煙火類群的輪廓相似，多為圓形，且煙火類群與花類群之色調差距大，故檢索成功率下降，期望能找出改進之方法。
- 5.IRHI 與 IRCI 之加權計分模式比例不同，若不作調整而直接平均計分，在實驗結果上會有所偏頗，需再多實驗以循找出最恰當之比例。

## 四、應用及展望

由實驗結果得知，本篇研究之檢索架構有著許多的優點，若能應用於科博館之檢索系統，應可增加數位博物館之使用率與普及率，使豐富資源能得到充分應用，發揚其原本創立此資料庫之宗旨。

由於此系統可以 100 % 找到本身圖像，故可以應用於檔案庫內尋找重複圖像，除了前面所描述的應用外，還有以下應用：

1. 圖書館封面檢索系統
2. 線上百科全書檢索系統
3. 指紋辨識
4. 瞳孔位置辨識
5. 車牌辨識
6. 葉脈辨識
7. 證件照片之規格檢驗
8. 互動玩具設計

在本篇論文中，我們提出了一套基於色調及輪廓的彩色影像檢索技術。雖然影像檢索的時間、比對運算速度、檢索成功率高低…等情形，我們均做了相當多的考量與改進，但在未來研究的方向上，不可避免地還是存在許多可以改進的空間。

## 陸、參考文獻

- [1]陳同孝、黃國峰、張真誠，數位影像處理技術，初版，臺北市，旗標，第3、7、9章，  
2003
- [2]Majid Rabbani、Paul W. Jones，Digital Image Compression Techniques，  
Bellingham,Washington，seventh printing，chapter 1、3、12，1991
- [3]Khalid Sayood，Digital Image Processing，NEW YORK，SECOND EDITION，chapter 1、  
2、9、13，2000
- [4]吳冠龍、尤孝庭，灰階影像內容之檢索技術，台灣 2005 年國際科學展覽會研究報告

## 評語

1. 提出一個影像檢索系統，相當快速地找出一定數目符合相似度要求的影像，可以用在數位博物館中之資料檢索。
2. 改進方向：檢索動率應有改善空間。
3. 應注意是否已經有夠有效的方法已經問過。