

應用遺傳程式規劃之影像自動分類系統

An Automatic Image Classification System Using Genetic Programming

林忠億 錢炳全 林欣靜 謝逸忠 黃永豐
Jung-Yi Lin, Been-Chain Chien, Shin-Jing Lin,
Yi-Chung Hsieh and Yung-Feng Hwang

義守大學 資訊工程學系
840 高雄縣大樹鄉學城路一段 1 號
Department of Information Engineering
I-Shou University
{m893310m, cbc, b8703056, b8703038, b8703072}@isu.edu.tw

摘要

在有關影像技術研究與應用中，影像分類是一個相當重要的研究主題。影像分類通常包括影像特徵分析與分類演算法兩個主要課題。本論文將提出一無特定領域之一般性影像自動分類系統，利用遺傳程式規劃演算法產生分類函數做為分類器，進而建構一自動影像分類系統。經由給定樣本影像之特徵向量及一個有效率的動態遞增訓練法，遺傳程式規劃將產生各類別之分類函數。本系統能在非特定的影像特徵向量中，得到相當高的分類準確率。在實驗中，我們測試我們所提出的影像自動分類系統，實驗結果將證明我們所提出分類系統能得到高準確率的分類結果。

【關鍵字：影像分類；遺傳程式規劃；自動分類】

ABSTRACT

Image classification plays an important role on applications of image processing. It is a challenge to classify general images precisely. In this paper, we propose a new automatic image classification system based on genetic programming. In the proposed system, we design an approach to learn the classification functions from a set of feature vectors. The main characteristic of the proposed system is that classification functions can be generated automatically according to the selected features. The distinct features are selected and evaluated together such that an effective classifier can be achieved. In the experiment, we found that the proposed automatic image classification system is accurate and efficient for general images.

【Keywords: image classification; genetic programming; automatic classification】

1. 簡介

在影像的各種研究與應用中，影像分類是一個相當重要的研究主題。有許多針對這個主題的研究方法已經陸續被提出，並且成功地應用在許多場合，例如身份辨識、臉孔辨識與地圖辨識等等。影像是複雜的資料集合，它可視為一個二維陣列，在 RGB 色彩空間中，陣列的每個元素值即為紅色、藍色與綠色三種顏色的值所組成。影像資料所表達的內容通常可分為兩方面來討論，第一是影像的視覺內容所表達的概念意義，例如風景、人物等等，另一方面則是不具有語意的影像像素，如顏色成份與像素分佈多寡等。因為影像資料本身的特性複雜，如果想直接以影像像素來判別其所代表的概念意義，是相當不可行的。通常我們會先利用特徵擷取 (Feature Extraction) 的演算法，將所需影像有意義的特性以特徵向量表示，然後經由此特徵向量來做為分類判斷的依據，配合適當的函數或演算法來進行分類的動作。因此，影像分像可分為兩個部份來討論，第一部份為特徵擷取，討論如何將代表影像內容的特徵擷取出來，再以特徵向量表示之，第二部份為識別部份，以影像之特徵向量做為依據，找出相對的比對方法，然後將影像分類至預先定義的各個類別中。

在影像分類的研究中，有許多方法被提出 [1][8][9][12]，例如 Nearest Feature Line (NFL)[1]，是將欲分類影像之特徵向量對應至特徵空間中，而原先在特徵空間中的所有樣本影像之特徵向量所形成的軌道 (Trajectory) 即可視之為類別，而欲分類之影像與這些軌道之間的距離將會決定此影像物件的類別為何。M. Szummer [12] 將影像對應到 Ohta 色彩空間、使用 Multi-resolution

Simultaneous Autoregressive (MSAR) 模型與 Shift-Invariant DCT 的係數來得出不同的特徵向量，進而使用 k -nearest neighbor 等分類演算法進行影像分類。其它方法如例用 Probabilistic Reasoning [9] 與 Sequential Resolution Nearest Neighbor (SRNN) [8] 等。

在本篇論文中，我們針對不特定範圍 (domain) 的一般影像特徵向量設計一具學習能力的分類函數學習系統，對於任何不同影像擷取演算法所擷取出來之影像物件特徵向量，皆能有效地自動學習出最適當的分類函數並準確地進行分類，而不必擔心如何計算特徵向量間相似度的問題。我們主要所使用的學習方法為遺傳程式規劃演算法 (Genetic Programming)，遺傳程式規劃是一種自動調整的演算法，能夠動態地依照不同的密合度 (fitness) 來找尋最符合需求的函數，因此，利用遺傳程式規劃演算法的特性，當影像物件經由特徵的擷取並以多維度特徵向量來表示之後，我們即可利用遺傳程式規劃演算法來學習並產生出對應於每個影像類別的分類函數，而經由這些分類函數，可以使我們的系統達成高分類準確率的目標。

本篇論文的結構如下：第二節中將簡單介紹遺傳程式規劃的基本步驟與方法，第三節介紹我們所用以展示的影像資料集合、影像分類系統架構與分類函數之動態訓練法，第四節說明分類實驗與所得到的結果，最後則是結論與未來的研究方向。

2. 遺傳程式規劃演算法

將遺傳的操作方式帶入資訊工程應用的是遺傳演算法 (Genetic algorithm)，遺傳演算法是將參數做編碼的動作，然後將編碼後的碼代入函數做運算，目的是求出最佳的函數值；經由對編碼後的參數做複製 (Reproduction)、交配 (Crossover) 與突變 (Mutation) 的動作，可以將參數做進化，於是在淘汰掉某些表現不佳的參數後，代入有最佳表現的參數後，即可得出最佳的解。

遺傳程式規劃 (Genetic programming，或稱遺傳規劃) 是由 J. R. Koza 在 1989 年所提出的一種自動函數產生方法，它能由給定的參數，自動的找出能符合這些參數的函數。和遺傳演算法一樣，遺傳程式規劃利用了遺傳演化的概念，包含了複製、交配與突變等操作方法。而經由演化的方式，可以逐次得到最佳的函數。然而遺傳程式規劃與遺傳演算法所適用的問題不同，遺傳演算法所求的是最佳解 (Optimum solution)，但是遺傳程式規劃所求的是最佳函數 (Fitness function)。相對於對參數做改變的遺傳演算

法，遺傳程式規劃的參數是固定的，然後對函數的多項式做演化運算，將函數的部份多項式保留或淘汰，使得經由進化的動作，找到能夠符合我們給定的參數的函數。

遺傳程式規劃的輸出為一函數運算式 (Functional expression)，它將我們所輸入的資料屬性加以運算成為一個數值，函數中可以使用各種不同的運算元，例如： $+$ 、 $-$ 、 \times 與 \div ，甚至 IF、ELSE 與 OR 等邏輯判斷與 Sine、Cosine 等三角函數。然後以二元樹的資料結構來表示運算式，一棵二元樹即為一個函數個體 (Individual)，使用者可以決定這棵樹的深度，樹葉的內容為資料屬性或常數運算元。許多二元樹將形成一個族群 (Population)，同樣的，使用者可以決定族群的數目。遺傳程式規劃的進化準則在於合適度 (Fitness) 的設定，亦即以個體的表現如何來決定個體是否需要進化，所以遺傳程式規劃最重要的一個環節就是設計合適度的判斷方式與程序。

遺傳程式規劃的初始化是隨機產生足夠的二元樹族群，這些族群中的二元樹將會經由複製 (Reproduction)、交配 (Crossover) 與突變 (Mutation) 等過程來演化出更符合要求的下一代函數，遺傳程式規劃的主要流程如圖 1 所示。主要演化下一代新個體的方法說明如下：

複製：是選擇具有較優良表現的個體給予保留，然後複製同樣的個體到新的族群。

交配：為選擇二個個體後，隨機產生一個交配點 (Crossover point)，將這兩個個體從交配點後的子樹做交換，於是產生二個新的個體，如圖 2 所示，運算式 $(5+X)+X$ 與 $(X+X)-2$ 經由交配後得到兩個新的運算式： $(X+X)+X$ 與 $(5+X)-2$ 。

突變：有兩種主要的做法，一種是單點突變，即直接對某一節點以新的內容取代，如圖 3，運算式 $(5+X)+X$ 經由突變後得出新運算式： $(5-X)+X$ 。另一種是子樹突變，則是將個體的某個節點的子樹以一棵新的、隨機產生的子樹取代。突變可以產生新的個體，以避免區域最佳值 (Local optimum) 的產生，如圖 4，運算式 $(5+X)+X$ 突變後為 $(7+X)$ 。

遺傳程式規劃可以經由增加或減少二元樹節點或樹末終端的方法來去除不必要的資料屬性，例如：假設我們輸入的資料屬性數目為 19，則在遺傳程式規劃輸出的運算式裡，可能只出現 8 個特徵向量，亦即有 11 個資料屬性的值是無意義或是不重要的，並不能做為分類的判斷，在影像物件特徵向量維度較高的時候，這種選擇重要特徵的優點將能使分類函數處理在運算時，處理較少的特

徵，進而快速地得出分類結果。

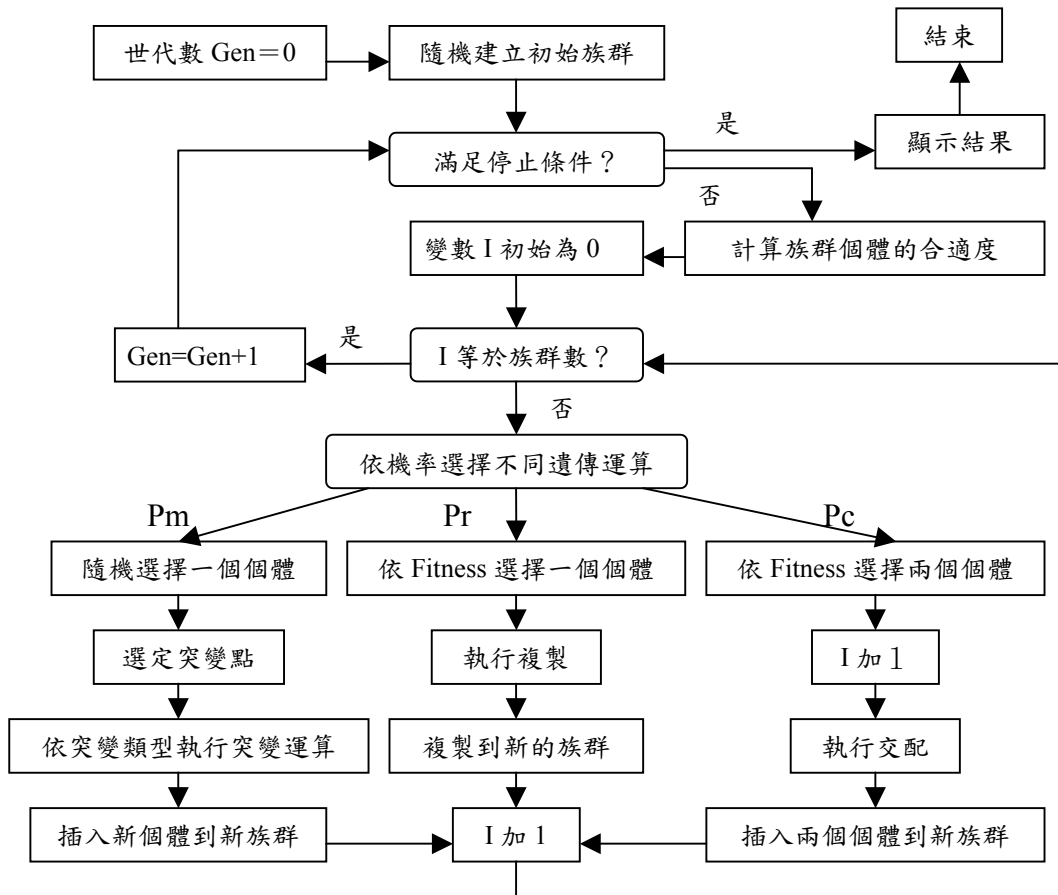


圖 1: 遺傳程式規劃流程圖

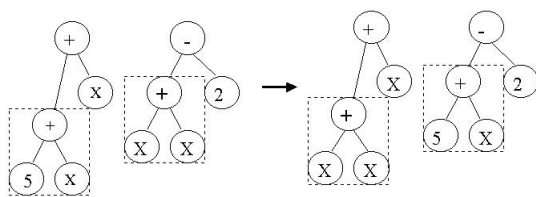


圖 2: 交配運算

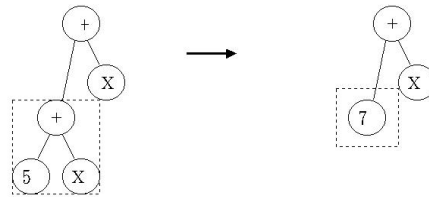


圖 4: 子樹突變

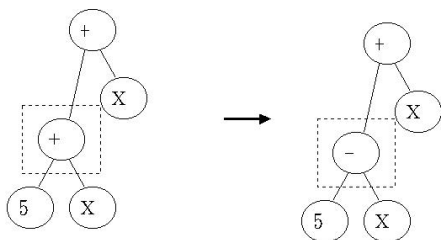


圖 3: 突變運算

3. 影像自動分類系統

本節中首先我們將提出一個影像分類系統架構，並詳述如何使用遺傳規劃演算法應用在影像分類。

3.1 影像分類之系統架構

在這一節，我們將對我們所提出之影像分類系統架構做一說明。對於一已知類別的影像物件，我們將其稱為樣本影像物件，經過影像特徵擷取 (Feature extract) 的處理之

後，樣本影像物件將以多維向量表示。對於所有影像所擷取出之特徵向量，我們建立一資料庫，將所有特徵向量儲存在特徵資料庫 (Feature DB) 中，而將樣本影像物件儲存於媒體資料庫 (Media DB)，而樣本影像物件的類別與檔名等資訊則儲存在資訊資料庫 (Information DB)。

然後，所有在特徵資料庫中的影像特徵向量將被視為訓練資料 (Training Data)，而訓練資料將應用在遺傳程式規劃中以得出對應於各類別之分類函數。在利用遺傳程式規劃並得到每個類別的函數之後，我們建立一資料庫稱為函數資料庫 (Function DB)，並將分類函數儲存於函數資料庫中，以便於分類時取出使用。

我們的系統流程介紹如下。於使用者介面中，在使用者輸入一個未知類別的影像物件之後，經過特徵擷取演算法處理，將得到影像物件之特徵向量，然後將各分類函數由函數資料庫中取出，對此特徵向量進行運算，經過各分類函數的運算與判斷之後，此影像物件的類別將被決定，於是此影像物件分類至適當類別，並將此影像物件儲存於媒體資料庫中，特徵向量與其它資料則分別儲存於特徵資料庫與資訊資料庫。最後系統將會顯示分類結果於使用者介面。

在媒體資料庫的影像增加之後，分類函數可能發生分類率下降的問題，此時可經由系統的函數資料庫與特徵資料庫，對所有分類函數再次進行遺傳程式規劃，使得各類別之分類函數能保持高度的分類準確率。

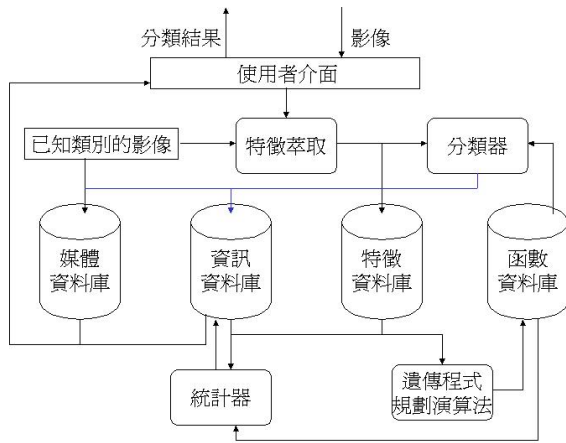


圖 5: 影像自動分類系統架構

3.2 問題定義

在這一節，我們將分類問題與我們所使用的符號做正式的定義。

首先我們定義影像的類別共有 K 類，每張影像物件可表示為 n 維特徵向量，而 K 個類別則可表示成集合 C

$$C = \{C_1, C_2, \dots, C_K\},$$

其中 C_1, C_2, \dots, C_K 分別代表第 K 類的影像物件。對於用來代表影像物件的影像特徵向量，在擷取後則表示為：

$$x_{jt} = (v_{j1}, v_{j2}, \dots, v_{jt}, \dots, v_{jn})$$

其中 $1 \leq t \leq n$, $v_{jt} \in \mathbf{R}$ 代表影像物件 x_j 之特徵向量的第 t 個特徵。

將已知類別的影像物件，即樣本影像物件，稱之為訓練資料 (training data)，以 TR 表示，而欲分類之未知類別的影像物件，稱為測試資料 (testing data)。假設 TR 的資料個數 $|TR| = m$, TR 中的每一個影像物件 x_j 均對應到已知類別其中的一類。我們將其表示為

$$\langle x_j, c_j \rangle, 1 \leq j \leq m, c_j \in C$$

TR 則可表示為

$$TR = \{\langle x_j, c_j \rangle | x_{jt} = (v_{j1}, \dots, v_{jn}), 1 \leq j \leq m, c_j \in C\}$$

然後我們可以定義分類函數 f_i 為對應類別 C_i 且由 \mathbf{R}^n 對映至 \mathbf{R} 的函數：

$$f_i: \mathbf{R}^n \rightarrow \mathbf{R}$$

對於給定的常數 a 與訓練集中影像特徵向量 $\langle x_j, c_j \rangle$ ，分類函數 f_i 必須能滿足以下條件：

$$\begin{cases} f_i(x_j) \geq a, & \text{if } c_j = c_i \\ f_i(x_j) < a, & \text{if } c_j \neq c_i \end{cases}, 1 \leq i \leq k, 1 \leq j \leq m$$

也就是說，以給定之常數 a 做為判斷之準則，若對於 $c_j = C_i$ 之影像特徵向量 x_j ，分類函數 f_i 能夠將此影像特徵向量對映至一個大於 a 之實數，則我們說分類函數 f_i 分類正確；若 $c_j \neq C_i$ ，則分類函數 f_i 應該能正確地得出一小於 a 的實數。對於所有的分類函數，我們定義一個分類函數集合 F ，並將其表示為

$$F = \{f_i | f_i: \mathbf{R}^n \rightarrow \mathbf{R}, 1 \leq i \leq K\}.$$

3.3 分類函數之動態遞增訓練法

我們在遺傳程式規劃對訓練資料作訓練的過程，並非一次即將全部的資料給予訓練，而是採用漸進式遞增資料的訓練。首先，

我們定義動態遞增訓練法所需要的變數 g, m', p, w 與 r ; g 是一整數，為每次動態遞增訓練法所進行的世代數， p 為一實數且 $0 < p < 1$ ， w 是一個實數，用以判斷分類函數的密合度 (Fitness) 優劣，最後定義 r 是一個整數，用以調整動態遞增訓練法所增加的訓練資料筆數。在初始階段，我們設定 $m'=0, r=1$ ，然後初始的訓練資料項目數，將其設定為 $m' \times p \times r$ ，在遺傳程式規劃每經過 g 個世代之後，我們判斷所得出的分類函數之密合度是否優於 w ，若是，則將 r 的值增加。然後，我們將參與演化的訓練資料筆數 m' 加上 $m' \times p \times r$ ，重覆以上步驟直到訓練資料全部訓練完為止。詳細的動態遞增訓練法之步驟整理如下：

- 步驟一：設定 g, p, w 之初始值，令 $r = 1$ 且 $m' = 0$ 。
- 步驟二：設 $m' = m \times p \times r + m'$ ，若 $m' \geq m$ ，則 $m' = m$ 。
- 步驟三：進行遺傳程式規劃，世代數為 g 。在 g 個世代結束之後，我們將得到一個分類函數。
- 步驟四：若 $m' = m$ ，則所得到的分類函數為最後結果，輸出此函數並停止動態遞增訓練法。否則執行步驟五。
- 步驟五：利用密合度函數計算所得到之分類函數的密合度，若此分類函數的密合度優於設定的 w ，則增加 r 的大小。否則 r 為 1。回步驟二。

在挑選訓練資料時，通常必須包含正例與負例，才能達到訓練目的，對於正例資料必須能正確地辨認 (Recognize)，而對於負例資料，必須要能正確地排除 (Discard)。我們將正負例之定義如下：對於一個影像物件 $\langle x_j, c_j \rangle \in TR, 1 \leq j \leq m$ ，對於某一類別 C_i ，若 $c_j = C_i$ ，則我們稱影像物件 $\langle x_j, c_j \rangle$ 為類別 C_i 之正例，反之，若 $c_j \neq C_i$ ，則此影像物件稱為類別 C_i 之負例。

3.4 密合度的判斷

在遺傳程式規劃中，一個分類函數是族群中的一個個體，而對於個體的密合度，是以一密合度函數來計算，在本論文中，我們所設計的密合度函數是基於誤差度量的方式的完成。考慮類別 C_i 之分類函數 f_i ，依 3.2 節所述，若影像物件 $\langle x_j, c_j \rangle$ 是正例，則我們期望 $f_i(x_j) \geq a$ ，而若 $\langle x_j, c_j \rangle$ 屬於負例資料，則希望 $f_i(x_j) < a$ 。為達成此要求，我們另外設定了兩數 α 與 β ，令 $\alpha > a, \beta < a$ ，且 $\alpha + \beta = 2a$ 。

對於正例資料，定義密合度誤差 $D_{positive}$ 為：

$$\begin{aligned} \text{If } c_j = C_i \text{ and } f_i(x_j) \geq a, D_{positive} &= 0, \\ \text{If } c_j = C_i \text{ and } f_i(x_j) < a, D_{positive} &= [\alpha - f_i(x_j)]^2. \end{aligned}$$

對於負例資料，定義密合度誤差 $D_{negative}$ 為：

$$\begin{aligned} \text{If } c_j \neq C_i \text{ and } f_i(x_j) \geq a, D_{negative} &= [f_i(x_j) - \beta]^2, \\ \text{If } c_j \neq C_i \text{ and } f_i(x_j) < a, D_{negative} &= 0. \end{aligned}$$

然後，個體的密合度定義為 *Fitness*，為正例密合度誤差與負例密合度誤差的總合。

$$Fitness = -D_{positive} + D_{negative}$$

由以上的密合度誤差，我們即可判斷在所有族群當中，何者表現較好，何者表現較差，在進行遺傳運算的交配運算、突變運算與複製等演化動作時，給予誤差小、表現好的個體較大的被選擇機率，而對於誤差大、表現不好的個體，則給予較低的被選擇機率，於是較佳的個體較有可能被演化，進而在新的一代的族群產生較佳的新個體，然後進行下一次的評估，最後即可得到密合度最佳的個體，而此個體即為我們所需要的分類函數。

4. 實驗

為了展示自動化影像分類系統，我們使用 UCI 影像資料集 Image Segment[1] 來做為我們系統的測試與展示。Image Segment 資料分為兩個資料集合，第一個資料集合是訓練資料，用以得出合適的分類函數，而第二個資料集合則是供分類函數測試其分類準確率之測試資料。Image Segmentation Data 共有 7 個類別，分別是 brickface、sky、foliage、cement、window、path 與 grass，在訓練資料集合中，各類別之影像物件數目為 30 筆，共 210 筆，而測試資料集合中各類別含 300 筆，共 2100 筆。每一筆影像萃取出 19 個特性形成特徵向量來表示。各特徵的萃取方法及所代表的意義如列於表一之說明。

我們使用 GP Quick 2.1[10] 來做為實驗的工具，並修改部份程式碼內容以符合實驗上的需求，使用的參數如表 2 所示，而資料項目的 19 個屬性值以 $F1, F2, \dots, F19$ 來表示。在密合度函數中，使用的參數定義為： $\alpha = 100, \beta = -100, a = 0$ ，在動態遞增訓練法中，參數設定為 $g = 500, p = 0.1, r = \{1, 2\}, w = 0$ 。實驗流程如圖 6 所示。

為了評估分類函數的分類準確度，我們採取以下的評估方式：各分類函數的準確率之計算分為兩個部份，一為 Precision，另一為 Recall。對於所有測試資料，考慮類別 C_i 之分類函數 $f_i \in F$ ，對於所有屬於測試資料之資料項目 x_j ，定義兩數 N_i 與 N_i' ， N_i 為所有被分類函數 f_i 辨認的資料筆數， N_i' 為 $x_j \in C_i$

且被分類函數 f_i 辨認的資料筆數。則 Precision 與 Recall 定義如下：

$$\text{Precision} = \frac{N'_i}{N_i}$$

$$\text{Recall} = \frac{N'_i}{|C_i|}$$

分類函數若具有較高的 Precision，則表示此分類函數的誤認率 (misclassification rate) 較低，而若具有較高的 Recall，則表示此分類函數的辨識率 (Recognition rate) 較高。

我們實驗得出的分類函數如表 3 所示。為了避免電腦計算除法時產生的誤差問題，我們在計算時並不簡化函數所有的除法部份。由我們所得出之分類函數可以知道，各類別中的影像物件若彼此具有複雜的關係，例如相似的特徵向量，則遺傳程式規劃所產生的分類函數將相對地更加複雜，這也代表了分類函數與各類別之間的關係相當密切，能真正地代表各個類別。對於 Image Segmentation 測試資料集合的分類結果，我們將其列於表 4。由分類的結果可以知道，每一條分類函數皆有相當高的 Recall 與 Precision，在 Window 與 Cement 兩個類別中，我們發現 Precision 較低，這是因為這兩個類別並不能很有效地以兩個分類函數區分，在 Window 類別與 Cement 類別中的部份影像物件特徵向量，在我們的方法只使用一條分類函數來代表一個類別的情形下，與其它類別的物件特徵向量極難分辨，因而造成 Precision 較低的情形。然而，實驗結果仍然證明了我們所提出的影像分類系統可以達到準確率相當高的分類結果。

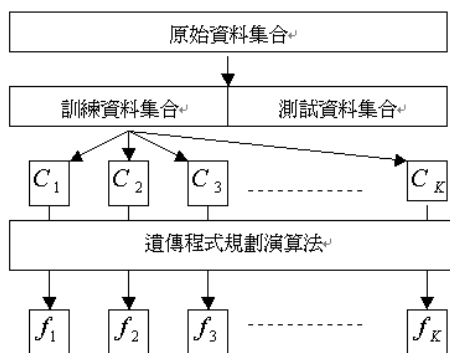


圖 6: 實驗示意圖

5. 結論與未來方向

本論文應用了遺傳程式規劃演算法與資

料庫的結合，成功地對影像物件進行分類，我們的實驗可以證明，遺傳程式規劃演算法所得到的分類函數，應用於影像分類上可以達到相當好的表現與效率。然而，使用分類函數仍然有缺點，主要的缺點在於，對於要分類的影像物件而言，同一個影像物件可能同時被二個以上的分類函數所辨認，於是將產生同一影像卻分屬於多個類別的錯誤結果，本論文中已注意到這個問題，但並未提供有效的解決的方法，在[5]中所使用的 SA (Strength of Association) measure 雖然具有可行性，但是仍不能完善地解決這個問題，因此，我們將在未來的研究中解決這一問題，以改善本研究之自動分類系統，並使其能更有效與準確地解決影像分類問題。

另外，我們在 Segmentation 資料庫的實驗結果中，發現在同一類別的物件特徵向量彼此差異較大時，或是不同類別的物件特徵向量相似時，只使用一條分類函數做為分類器所得到的分類結果，依然有可改善的空間。例如可以使用二條以上的函數來代表一個類別，在少量分類函數數目的前提下，來改善分類結果與分類準確率。

未來我們除了解決上述發現的問題之外，我們還將應用遺傳程式規劃演算法於影像回授，提供一系統可自動尋找符合使用者主觀意識中合適的影像。

參考文獻

- [1] Blake, C., Keogh, E. & Merz, C.J. (1998). UCI Repository of machine learning databases, <http://www.ics.uci.edu/~mllearn/MLRepository.html>, Irvine, University of California, Department of Information and Computer Science.
- [2] C. C. Bojarczuk, H. S. Lopes, A. A. Freitas, "Discovering comprehensible classification rules using genetic programming: a case study in a medical domain", Proc. Genetic and Evolutionary Computation Conference (GECCO -99), Orlando, FL, USA, pp. 953 - 958, 1999.
- [3] A. A. Freitas, "A genetic programming framework for two data mining tasks: classification and generalized rule induction", Proc. 2nd Annual Conference Morgan Kaufmann, pp. 96-101, 1997.
- [4] Stan Z. Li, K. L. Chan, Changliang Wang, "Performance Evaluation of the nearest feature line method in image classification and retrieval", IEEE Transactions On Pattern Analysis and Machine Intelligence, Vol. 22, No. 11, pp. 1335-1339, 2000.
- [5] J. K. Kishore, L. M. Patnaik, V. Mani, V. K. Agrawal, "Application of genetic

- programming for multicategory pattern classification”, IEEE Transactions On Evolutionary Computation, Vol. 4, No. 3, pp. 242-258, 2000.
- [6] J. R. Koza, Genetic Programming: “On the programming of computers by means of Natural Selection”, MIT Press, 1992.
- [7] J. R. Koza, “Introductory genetic programming tutorial”, Genetic Programming 1996 Conference, Stanford University, 1996.
- [8] T. Kuyel, W. Geisler, J. Ghosh, “Fast image classification using a sequence of visual fixations”, IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics, Vol. 29, No. 2, pp. 304 –308, 1999.
- [9] S. Paek, S. F. Chang, “A knowledge engineering approach for image classification based on probabilistic reasoning systems”, IEEE International Conference on Multimedia and Expo, Vol. 2, pp. 1133 –1136, 2000.
- [10] J. Sherrah, R. E. Bogner and Abdesselam Bouzerdoum,” Automatic selection of features for classification using genetic programming”, in Proc. Australian New Zealand Conference On Intelligent Information Systems, 1996, pp. 284-287.
- [11] A. Singleton, “Genetic programming with C++”, Byte, pp. 171-176, 1994.
- [12] M. Szummer, R. W. Picard, “Indoor-Outdoor image classification”, IEEE International Workshop on Content-Based Access of Image and Video Database. Proc. pp. 42 –51, 1998

表 1: 特徵向量之特徵名稱與意義

次序	名稱	意義
1	region-centroid-col	the column of the center pixel of the region.
2	region-centroid-row	the row of the center pixel of the region
3	region-pixel-count	the number of pixels in a region = 9
4	short-line-density-5	the results of a line extractoin algorithm that counts how many lines of length 5 (any orientation) with low contrast, less than or equal to 5, go through the region.
5	short-line-density-2	same as short-line-density-5 but counts lines of high contrast, greater than 5.
6	vedge-mean	measure the contrast of horizontally adjacent pixels in the region. The mean are given. This attribute is used as a vertical edge detector
7	vegde-sd	measure the contrast of horizontally adjacent pixels in the region. The standard deviation are given. This attribute is used as a vertical edge detector
8	hedge-mean	measures the contrast of vertically adjacent pixels. Used for horizontal line detection.
9	hedge-sd	measures the contrast of vertically adjacent pixels. Used for horizontal line detection.
10	intensity-mean	the average over the region of $(R + G + B)/3$
11	rawred-mean	the average over the region of the R value
12	rawblue-mean	the average over the region of the B value
13	rawgreen-mean	the average over the region of the G value
14	exred-mean	measure the excess red: $(2R - (G + B))$
15	exblue-mean	measure the excess blue: $(2B - (G + R))$
16	exgreen-mean	measure the excess green: $(2G - (R + B))$
17	value-mean	3-d nonlinear transformation of RGB. (Algorithm can be found in Foley and VanDam, Fundamentals of Interactive Computer Graphics)
18	Saturatoin-mean	3-d nonlinear transformation of RGB. (Algorithm can be found in Foley and VanDam, Fundamentals of Interactive Computer Graphics)
19	hue-mean	3-d nonlinear transformation of RGB. (Algorithm can be found in Foley and VanDam, Fundamentals of Interactive Computer Graphics)

表 2: GP Quick 參數設定

參數	值	參數	值
Population	2000	Crossover weight	28%
Node mutate weight	43.5%	Crossover weight annealing	20%
Mutate constant weight	43.5%	Mutation weight	80%
Mutate shrink weight	13%	Mutation weight annealing	40%
Selection method	Tournament	Terminal	$F1, F2, \dots, F19$
Tournament size	7	Function set	$+, -, \times, \div$

表 3: 各類別之分類函數

$f_{\text{Brickface}} = ((F12/F16)-(F16-(F14-(F3-1))))$
$f_{\text{Sky}} = F12-112$
$f_{\text{Foliage}} = (((((((((120/F14)-(-71*F8))/(F11 \times F11 \times F11))+((F15+F16) \times (F17-F14))+((F19 \times -38)-((F8-F11) \times F8))-((F8-F11) \times (F8-F17)))))))+(F8-F11))-35)/20)-F12) \times 15)$
$f_{\text{Grass}} = (F19+(1/(F12-120)))$
$f_{\text{Window}} = (- (/ F3 (\times F18 (- F15 F11))) (/ (\times (/ F3 (\times F18 (- F15 F11))) (- F15 (/ (- F15 (- F15 (- F15 (- F15 (/ F11 (- F15 F11)))))) F9))) (- (- F11 F16) F16)))$
$f_{\text{Cement}} = (/ (\times (+ (+ (- (/ (+ F12 (+ F3 F2)) (+ F12 (+ (- (/ (+ (/ F12 (+ F12 (+ F3 -21))) (+ F3 (+ F8 F2))) F3) F12) -25))) F12) (/ (/ F12 F2) (+ F9 -27))) 94) F3) (+ F12 -27))$
$f_{\text{Path}} = (((((((((F19-F2)+F10) \times ((F9+F12)/(-73/F12)))/F17)+F10)+((((((((((F19-F2)+F10) \times ((F9+F12)/(-74/F12)))/F17)+F17)+F6)-F9)-126)+(F8-F9)))-120)+(F19-F9))$

表 4: Segmentation 之測試資料分類結果

	$f_{\text{Brickface}}$	f_{Sky}	f_{Foliage}	f_{Grass}	f_{Window}	f_{Cement}	f_{Path}
Brickface	283	0	0	0	87	97	0
Sky	0	300	0	0	15	0	1
Foliage	2	0	273	0	43	34	0
Grass	0	0	0	297	3	68	0
Window	1	0	44	0	265	48	0
Cement	1	0	1	0	73	281	32
Path	0	0	1	0	3	5	299
Precision	98.6%	100%	85.6%	100%	54.2%	52.7%	90.1%
Recall	94.3%	100%	91.0%	99.0%	88.3%	93.7%	99.7%