

棒球視訊檢索與事件偵測方法

Video Retrieval and Event Detection for Baseball Video

杜維昌 (Wei-Chang Du)

義守大學資訊工程系

wcd�@isu.edu.tw

李健銘 (Jian-Ming Lee)

義守大學資訊工程系

ptljm@ms9.hinet.net

摘要

隨著電腦技術不斷提升與網路環境的日漸成熟,各種不同形式的數位媒體也因應而生。特別在最近幾年間,利用數位攝影機拍攝各種場景已不再是那麼遙不可及的事。但就從視覺來說,使用攝影機所得的數位視訊資料只是忠實記錄場景影像,由於缺乏語意資訊,因而仍需仰賴人類大腦的認知。為了提升視訊的應用價值與範疇,自動生成高階意涵是必要的過程。在考量各種運動視訊中,移動物件的運動行為是球賽內涵的重要特徵。在此一研究中,我們以棒球球賽的視訊資料為研究標的,目的是希望透過架設固定式數位攝影機可將拍得的視訊畫面自動萃取出運動特徵,亦即將前景物件從背景影像分離出來,以求得移動物件在場景中的運動軌跡。接著,對運動行為與特徵設計相關索引與比對方法,以利於特徵的有效使用。為了進一步擷取高階意涵,開發有關棒球球賽的事件模式,以提供在時間軸上得到較為高階的內涵。

關鍵詞: 棒球視訊、物件追蹤、運動軌跡、視訊檢索、事件偵測。

一、簡介

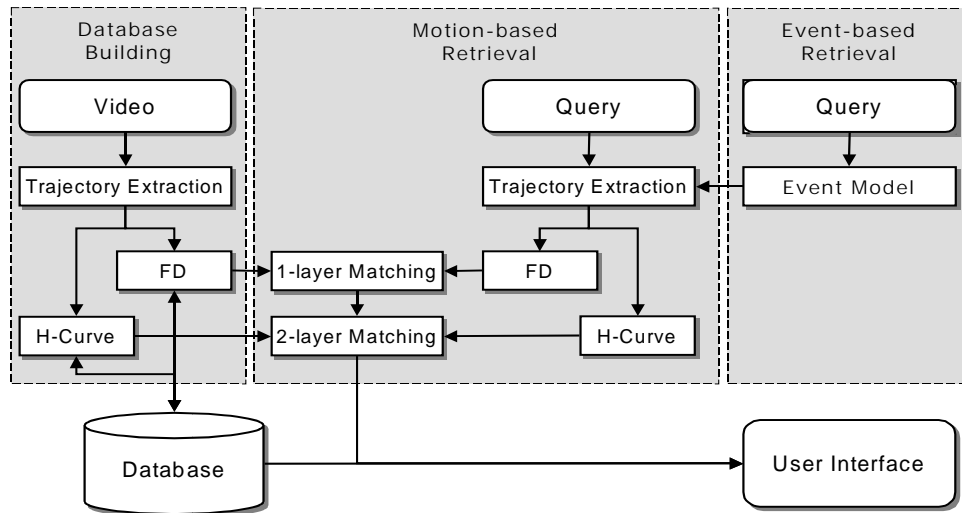
由於儲存媒體與壓縮技術的大幅進步,使得大量的視訊資料得以使用數位方式儲存,因而利用電腦來對視訊資料作自動化管理的構想與需求也因應而生,常見的應用包括監視器系統[5][9][11]、隨選視訊(video on demand, VOD)[10]、數位圖書館(digital library)[8]等。隨著視訊擷取介面和數位攝影機的普及,人們可以輕易將數位視訊資料儲存在電腦儲存媒體之中。一般而言,數位視訊除聲音以外,主要包括一連串的數位影像。以 NTSC 標準為例,視訊是由每秒 30 張畫面所組成,而每張畫面由無數代表顏色的取樣像素所構成。這樣的表示方法僅僅表達視覺效果,因而仍然需要仰賴人類大腦的認知。但在人類思維中,思考的主題經常是較為深沉的語意部分。如何將數位視訊資料轉化為以內涵為基礎的描述方式,於是成為提升電腦智能的重要步驟。自從動畫專家群組制定 MPEG-7 藉以提供影音多媒體內容的描述介面(multimedia content description interface),然而對於描述資料的生成與搜尋技術則急待進一步開發。由於多媒體視訊資訊用文字的方式不足以充分表達其涵意,取而代之是以內涵為基礎的方式來描述。

為了解決此一問題,學者提出各種方法將多媒體資料轉化成以內涵為基礎的描述方式,希望能夠在多媒體資料上自動萃取出符合人類感官的特徵,乃至於高階語意的生成,使電腦對數位多媒體資料的認知能提升至與人腦相近的程度。在影像資料上,其視覺特徵包括了紋理、顏色、外形等,基於視訊資料是由一連串有序的影像所構成,因此視訊資料庫也可以應用影像特徵來建立[14]。另一方面,視訊資料不單只是一張張獨立的影像而已,這些影像之間還具有時間上的相關性,此一性質包括攝影機的運鏡與物件的移動等動態行為[6]。這些動態的特徵對視訊資料而言是相當重要的視覺特徵,同時也是推演出高階語意的重要關鍵,我們可以藉此一動態特徵來對特定類型影片歸納出事件模式。

由於一套視訊檢索系統可涵蓋多個不同層次的議題,而這些議題之間往往存有相依性,因此必須分別從不同因素來加以考量[1][3]。以視訊資料的動態特徵為例,我們將會面對下列幾個問題:物件動態特徵的表達方式、特徵的比對方法、檢索效率的考量、使用者查詢介面的制定[14]。對於上述問題,在本研究中,我們將以固定視點之監視器視訊為對象,提出通用性的雙階層比對架構,以尋求在檢索效率與擷取成效間取得較佳的平衡點。在查詢介面上,我們以手繪軌跡作為基本輸入方式,並在其上架構事件式查詢方法,讓使用者能選擇系統管理者預先建立好的事件定義,或是以自行設計的事件來作查詢。基於上述檢索平台,我們將以棒球視訊為例,建立主要的事件模式,並測試相關事件擷取的實際成效,圖一為整體系統架構圖。

二、運動分離與特徵表達

一般而言,物件可由不同特徵來作分離,主要包括空間性特徵和時間性特徵兩類。在空間的視覺特徵上,我們可利用前景物件與背景影像在色調上的不連續性,以及物件內部顏色與紋理的相依性質,藉此對影像作區域性分割。可以想見的是,單獨使用此一類特徵,不容易得到穩定的擷取成效,這是因為電腦不像人腦對事物的認知,因而如能事先得知外形的資訊,將有助於有效分離。在時間上的運動特徵方面,乃依據物件在時間軸上變動的位移資訊為基礎,來尋找出前景物件所在的區域。由於兩相鄰畫面間格很短,因而利用兩畫面間的差異是偵測移動物件的重要途徑。



圖一、系統架構圖

在此，在色調上不作任何假設。但由於單純只靠物件在空間上的特徵，不足以充分表達其資訊。因此，我們也採用移動物件的運動特徵為基礎，並且使用差分影像和區域性運動分離方法來處理。在背景固定不動的前提下，可使用影像差值法將兩張相鄰畫面相減來偵測畫面變動的部分，進而分離出運動物件。假設第 t 張畫面上座標 (x, y) 處的像素 RGB 值分別為 $I_R(x, y, t)$ 、 $I_G(x, y, t)$ 及 $I_B(x, y, t)$ ，那麼相鄰兩畫面座標 (x, y) 處的差異度可定義為

$$D(x, y, t) = \sqrt{\Delta I_R^2 + \Delta I_G^2 + \Delta I_B^2}$$
，其中 ΔI_R 、 ΔI_G 、 ΔI_B 是像素點 (x, y) 上在 t 時間的變化量。若 $D(x, y, t)$ 大於門檻值時則可簡單判定該像素為移動物體的一部分。當找出整個畫面中移動物件的像素點之後再作內部填充與雜訊濾除的動作，接著將剩餘的像素群聚起來 (clustering) 便可得出移動物件的外形。最後計算其最小包圍矩形 (minimal bounding rectangle) 的重心，即可得到該物件的座標。將一個物件在連續畫面上的座標依序記錄起來，即構成所謂的運動軌跡。

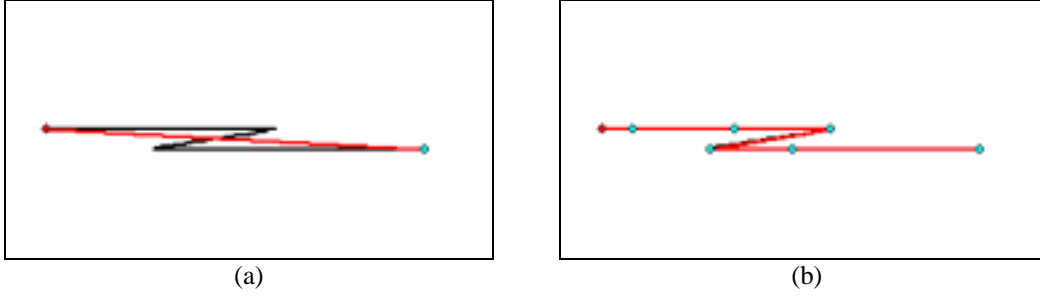
視訊物件軌跡的建立，是依據時間軸上連續畫面中的物件，取出一連串的重心座標而成，所得到的軌跡具有位置與時間特徵。在多數的影片格式中，畫面與畫面之間的時間間隔通常是固定的，因此我們常會省略時間軸的表示，而只以 XY 座標平面來表示物件軌跡。假設一物件在第 i 張畫面的座標為 P_i ，若一段視訊有 n 張畫面，則該物件的運動軌跡為 $T = \{P_0, P_1, \dots, P_{n-1}\}$ ，此座標序列又可稱為數位化曲線 (digitized curve)。由於軌跡除了位置特徵外，還有方向、速率等其它特徵。在過去文獻中，速率特徵經常被忽略或是獨立於軌跡特徵之外的形式來表示 [4]。為了充分運用速率特徵，並與時間上相互對應的位置特徵作緊密結合，我們提出「混合特徵曲線」作為軌跡表示方式。所謂「混合特徵曲線」即是將原本的二維曲線結合其它特徵形

成高維度的曲線，那麼物件的軌跡特徵將與其它特徵建立起更緊密的相關性。在本研究中，我們將速率特徵與原來的二維數位化曲線結合而成三維的軌跡曲線，並將其稱為 H-Curve (hybrid curve)。

一般情況下，視訊物件的移動速率並非太快，如此一來在高畫格率的影片下所擷取出的軌跡曲線經常是由高密度點所構成，如果能對曲線作適當的簡化，只保留較重要的特徵點，將可在比對中降低運算量。重複疊代法 (簡稱 IRM) [2] 為一多邊形近似法，其作用可以儘量保留高曲率的像素點，簡化後的曲線外形可用來近似原數位曲線。由於計算簡單，同時也容易應用於高維度的曲線上，因此我們利用 IRM 來簡化「混合特徵曲線」。對於一段高維度數位曲線，IRM 的運算步驟如下：

1. 設曲線的首尾兩個點為 P_0 與 P_{n-1} 。
2. 在曲線上所有的點當中，找出距離 $\overline{P_0 P_{n-1}}$ 最遠的點，若其距離大於門檻值，則記錄該點為近似後的頂點，並設其為 P_s ；若是小於門檻值則停止。
3. 將 P_0 至 P_s 與 P_s 至 P_{n-1} 兩段子曲線分別視為新的曲線，再由步驟一作起。

當運算結束後，首尾兩點與每次疊代中所記錄下來的所有 P_s 點即為簡化後的曲線。步驟 2 中所使用的門檻值 為控制近似程度的參數，設定越大則留下的特徵點數越少，但外形失真度也就越大。IRM 主要應用於外形的近似，並沒有考慮到物件軌跡的時間性質，例如一物體沿著一條直線來回移動，這種情況下 IRM 會將一些重要的軌跡資訊剔除，如圖二。但一般而言，視訊影片中的物件多屬於自然物體，在這種極端的轉折處，其物體的移動速率通常也會隨著變化，因此加入了速率特徵的 H-Curve 正好可以讓 IRM 偵測出速率變化大的點，因而得以保留原曲線的外形。



圖二、經 IRM 簡化結果，其中黑色曲線為原始軌跡，紅色則為簡化後的曲線：(a)只考慮空間性軌跡情形，(b)同時考量空間與速率特徵之情形

三、運動軌跡檢索方法

在這一節裡，我們將利用物件軌跡來建立軌跡檢索系統。為了在保有精確性的前提下提升比對效率，我們使用雙階層的比對架構。第一層使用傅立葉描述子(Fourier descriptors, 簡稱 FD)的比對方法，以利於快速篩選出候選資料。經由傅立葉轉換所得出的係數中，低頻係數項即代表曲線的大致外形，因此只需比對少數的幾個低頻係數，便能初步判定該軌跡曲線是否吻合，而且每次比對的資料為固定維度的特徵值，所以在資料檢索上可利用高維度索引技巧以得到高效率的搜尋成效。第二層則是以較精確的方式計算出資料庫軌跡的排名，使用動態規劃法(dynamic programming)計算軌跡與軌跡間在空間上的歐基里德距離(Euclidean distance)，以作為衡量軌跡相似與否的主要依據。

3.1 傅立葉描述子的索引與比對

傅立葉描述子是對一整段軌跡曲線作描述，因而將軌跡曲線作適當分段將有助於兩曲線間的有效比對。首先，定義軌跡分段的規則。基於考量使用者在查詢上的習慣與比對效率問題，所以訂定的規則必須依據高階的意涵以符合事件描述的完整性。在觀察影片中物體的運動行為後可發現，當物體停止移動時，通常為一基本事件的中斷點，因此我們定義若某一段軌跡的速度接近 0 且持續一小段時間，則可將此一小段軌跡切除，其中速度與持續時間的門檻值應配合不同影片的物件運動特性而作適當的調整。

切割後的軌跡片段可經傅立葉轉換得到頻率係數，但軌跡片段的取樣點數目往往不會一樣。不同數目的資料，在經過傅立葉轉換後得出來的係數也會不同而難以進行比對，所以必須先將所有曲線片段重新取樣成相同點數才行。常見的重新取樣方式，有「等比例時間取樣」及「等比例線段長度取樣」。由於等比例時間取樣會受到物件移動速度的影響，使得取樣後的資料會帶有速度特徵。在考慮到檢索時速度特徵是選擇性條件，因此我們使用等比例線段長度取樣法。假設一曲線片段的長度為 L ，共需取樣 N 個點，則將在曲線上每隔 $L/(N-1)$ 的長度取樣一次。

可應用於傅立葉轉換的外形特徵有複數座標(complex coordinates)、質心距離(centroid distance)、曲率(curvature)及累加角度(cumulative angular)等特徵，其中以質心距離在轉換成 FD 後擁有較強韌的比對效果[15]。質心距離的定義如下：設 (x_c, y_c) 為軌跡的質心座標， $(x(t), y(t))$ 為軌跡的第 t 個座標，則每個座標點的質心距離 $r(t)$ 為

$$r(t) = \sqrt{(x(t) - x_c)^2 + (y(t) - y_c)^2},$$

接著，便可將外形特徵 $r(t)$ 代入離散傅立葉轉換：

$$FD_k = \frac{1}{N} \sum_{t=0}^{N-1} r(t) e^{-i2\pi kt/N}, \quad k = 0, 1, \dots, N-1$$

其中 FD_k 即為傅立葉描述子。由於質心距離特徵是實數值，因此 FD_{N-k} 與 FD_k 是共軛複數，所以只需記錄 FD 的前 $N/2 + 1$ 項。另外，考慮到我們將提供位移與縮放不變量的比對，故必須將 FD 轉成這些不變量。求得方式是對 FD 取絕對值，縮放不變量則可藉由將交流係數除以直流係數而得到：

$$FD_{inv} = \left(\frac{|FD_1|}{|FD_0|}, \frac{|FD_2|}{|FD_0|}, \dots, \frac{|FD_{N/2}|}{|FD_0|} \right)$$

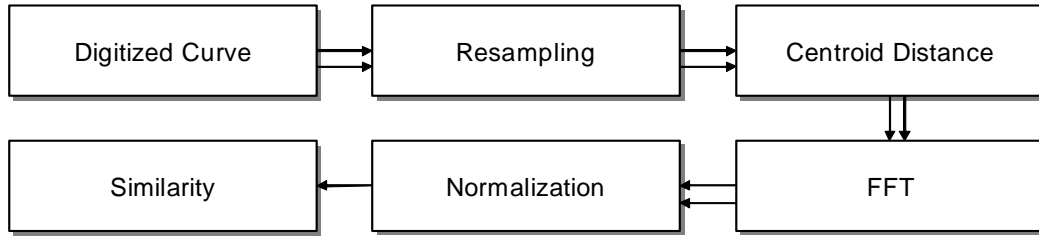
此時， FD_{inv} 即為一組具有位移、縮放不變量特徵。最後，假設兩軌跡曲線的 FD_{inv} 分別為 $(F_Q^0, F_Q^1, \dots, F_Q^{N-1})$ 與 $(F_S^0, F_S^1, \dots, F_S^{N-1})$ ，並且只取前 M 項 FD_{inv} 作比對，那麼我們可定義兩曲線差異度為

$$D_{FD} = \sqrt{\sum_{i=0}^M (F_Q^i - F_S^i)^2}, \quad M \leq N-1$$

此乃因為低頻係數項代表了曲線的大致外形，圖三為比對流程。

3.2 混合特徵曲線相似度比對

在特徵相似度比對上，我們以「混合特徵曲線」在空間上的歐基里德距離作為差異度計算方式，假設使用者查詢曲線為 $Q = (P_0^Q, P_1^Q, \dots, P_{m-1}^Q)$ ，資料庫中某段曲線為 $S = (P_0^S, P_1^S, \dots, P_{n-1}^S)$ ，則其差異度可表為



圖三、使用傅立葉描述子之比對流程

$$D(Q, S) = \max \left\{ \begin{array}{l} \max_{i=0}^{m-1} \left(\min_{j=0}^{n-1} (d(P_i^Q, P_j^S)) \right) \\ \max_{j=0}^{n-1} \left(\min_{i=0}^{m-1} (d(P_i^Q, P_j^S)) \right) \end{array} \right\}$$

其中 $d(P_i^Q, P_j^S)$ 代表 P_i^Q 與 P_j^S 的歐基里德距離。運動軌跡曲線是有序的点集合，上述的差異度計算方式並沒有考慮到匹配的前後次序，如此一來軌跡的比對將與單純的圖形比對無異。例如兩物件所經過的路徑相同，但位移方向卻是完全相反的兩軌跡，在此一比對方法下將分辨不出這種差異，因此必須進一步限制曲線的比對順序。此外，每段視訊資料的特徵數目不一定相等，況且數目相等也不代表曲線 Q 的第 i 點就一定對應到 S 的第 i 點。若要在限制比對順序下套用上述差異度的計算式，則必須要列出兩軌跡的特徵點所有可能的配對，並從這些配對中找出最佳的配對結果。若以一般列舉方式求解將會花費指數次方的計算時間，而動態規劃法則可有效解決此一問題[7][13]。假設 $DP(Q, S)$ 為最佳解，那麼

$$DP(Q, S) = w(m-1, n-1)$$

其中

$$w(i, j) = \min \left\{ \begin{array}{l} \max \{ w(i, j-1), d(P_i^Q, P_j^S) \} \\ \max \{ w(i-1, j), d(P_i^Q, P_j^S) \} \\ \max \{ w(i-1, j-1), d(P_i^Q, P_j^S) \} \end{array} \right\}$$

where $w(k, l) = 0$ if $k < 0$ or $l < 0$.

運算方式是在 $m \times n$ 晶格點上找出由 $(0, 0)$ 走到 $(m-1, n-1)$ 的路徑，且其路徑花費(path cost)要最小。晶格上每個節點 (i, j) 擁有節點花費(node cost)與路徑花費兩個值，其中節點花費值定為 P_i^Q 與 P_j^S

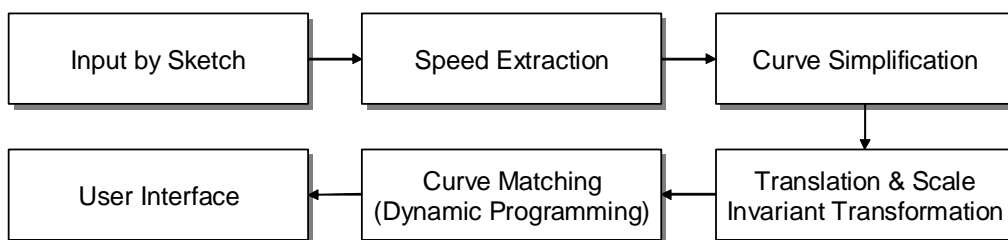
之間的幾何距離，即 $d(P_i^Q, P_j^S)$ ；路徑花費則儲存由 $(0, 0)$ 走到 (i, j) 的最小花費值，即 $w(i, j)$ ，同時每個節點也都會有一額外的欄位記錄此最小值是由哪個路徑得來。因此表格只要按順序由左至右、由上而下，一直到 $w(m-1, n-1)$ 便可完成計算，最後節點 $(m-1, n-1)$ 的 $w(m-1, n-1)$ 值即為兩軌跡的最佳配對之路徑花費，也就是我們所求的兩軌跡之間的歐基里德距離，其時間複雜度為 $O(mn)$ 。若自節點 $(m-1, n-1)$ 的路徑來源記錄追溯回去，則可得到配對的內容，圖四即為本節所述之混合特徵曲線比對的流程。

四、事件偵測與檢索

在前一節中，我們介紹了雙階層軌跡比對方法，但在實際的應用上，使用者更希望能以高階語意來與電腦溝通，並用以查詢出具有事件意涵的視訊片段。以運動視訊為例，只有少數人會去查詢某球員從畫面上方跑到下方這類單調的畫面，但對於絕大多數的一般使用者而言，比賽中的安打、得分等精彩畫面才是他們最感興趣的部分。

4.1 單一物件與多物件的事件檢索

視訊的事件有可能只是單一個物件的運動行為，也有可能是多個物件的互動所構成。在上一節所敘述的軌跡比對系統中，顯然可以滿足單一物件運動行為的查詢，我們只需在原有的主系統上，再加上一層作為輸入關鍵字的事件式查詢架構。此子架構對於管理人員或是進階使用者來說，他們可以為一般使用者預先定義每一事件所代表的軌跡關係模型，而一般使用者在查詢時只需簡單選擇檢索的關鍵字，那麼此子架構可將輸入轉換成對應的軌跡，輸入至主系統查詢，藉以達到單一物件的事件檢索方式。



圖四、軌跡曲線處理流程圖

由多個物件的互動所構成的事件上，首先是擴充上述的事件查詢架構，使其能管理包含多個軌跡的查詢，接下來還必須能判別出多個物件間互動關係的差異。每個物件本身皆具有相對於自己的空間與時間關係，而物件間的互動需要物件間的空間與時間關係。由於軌跡檢索就己含有物件的空間關係，因此我們最後為這層架構加上時間關係的比對功能，那麼即可具備多個物件事件的查詢功能。

4.2 事件檢索系統設計

時間關係的比對上，我們設定每個軌跡都有兩個可供比對的時間點，一為軌跡的起點時間，一為軌跡的終點時間。在關係上，我們設定有「之前」、「同時」、「之後」三種關係，並且可以指定其時間的間隔大小，基本單位為秒。為了使定義能更加明確，我們為系統制定了查詢語法，每一段基本描述式可表為

$obj.tp [op t] rel_op obj.tp [op t] [e:error_time] [l:limit_time]$

分別說明如下：

- *obj* 為輸入軌跡的代號，代表資料庫裡軌跡與 *obj* 相似的物件所構成的集合，同一段描述式內的兩個 *obj* 不得相同。tp 為時間點屬性，可為 start 或 end，start 代表起點時間，end 代表終點時間。中括號內為選用語法，可以不使用。
- *op* 可為加法(+)或是減法(-)運算子，*t* 為秒數。
- *rel_op* 為關係描述子，可用的關係描述子有大於(>)、等於(=)、小於(<)、大於等於(>=)、小於等於(<=)、不等於(<>)。
- *error_time* 為容許誤差時間，單位為秒，例如在使用等於描述子(=)時可指定容許誤差時間，以避免檢索條件過於嚴苛。*limit_time* 是為了使用等於(=)以外的關係描述子時，用於限制最大容許的時間差。以“a.start > b.end”描述式為例，若是不指定 *limit_time*，則 a.start 比 b.end 還遲上一個小時這類組合也仍符合描述式，但在一個事件裡，很少有物件互動的時間會相隔這麼長，此時便可設定 *limit_time* 以限制最大容許的時間差。若 *limit_time* 與 *error_time* 同時使用，則最大容許時間差將變為 *limit_time*+*error_time*。
- 描述式可以是複合式的表達方式，描述與描述之間以邏輯運算子結合，可以使用的邏輯運算子為 AND 與 OR，其中 AND 運算優先權高於 OR。若使用了兩個以上的邏輯運算子，可使用小括號來改變運算優先權。

我們舉些查詢範例：假設手繪輸入三個軌跡，分別設定其代號為 *a*、*b*、*c*，若要查詢符合「軌跡像 *a* 的物件，在軌跡像 *b* 物件停止 5 秒後才開始動」的影片片段，則語法為“a.start > b.end + 5”。同理，要查詢「*a* 開始動之後，*b* 才開始動，且 *a* 與 *c* 同時停止，並容許有前後 3 秒鐘的誤差」，則語法為“b.start > a.start AND a.end = c.end e:3”。當使

用者輸入多物件事件資訊並開始檢索後，事件索引系統會將每個輸入的軌跡曲線送至主系統查詢，並分別為每個軌跡的查詢回傳結果存成一個集合。當所有集合都收集完畢後，依所輸入的時間關係查詢語法，開始至各個集合內挑選符合語法描述的軌跡組合，並將結果資訊回傳給使用者。

接下來說明描述式的時間關係比對與多描述式的邏輯運算方法。假設一描述式 *expr* 裡指定了 Q_a 與 Q_b 兩個使用者手繪輸入的軌跡，且 Q_a 不得等於 Q_b 。假設 Q_a 與 Q_b 於雙階層軌跡檢索分別得出 S_a 與 S_b 的物件集合，則 *expr* 的時間關係比對結果應滿足：(1) $s = \{x_a, x_b\}$ ，其中 x_a 與 x_b 的時間關係符合 *expr* 所述，(2) $x_a \neq x_b$ 。多描述式的邏輯運算，在 AND 運算上，假設有兩單一描述式 *expr_l* 與 *expr_r*，並以 AND 運算子連結成 *expr_l* AND *expr_r* 之複合描述式。設兩軌跡 Q_l 與 Q_r 各自代表在 *expr_l* 與 *expr_r* 中所指定的使用者輸入之軌跡之一，並參照前述單一描述式比對方法中的定義為 Q_l 、 Q_r 定義出對應的物件軌跡 x_l 與 x_r 。於是，*expr_l* AND *expr_r* 的運算結果 $S_{(expr_l, expr_r)}^c$ 應滿足：(1) s^c 是由 $S(expr_l)$ 與 $S(expr_r)$ 中各取一組集合，經聯集後所成的一個集合，(2) 存在至少一組 $Q_l = Q_r$ ，(3) 對於所有的 x_l 與 x_r ，如果 $Q_l = Q_r$ ，則 $x_l = x_r$ ；如果 $Q_l \neq Q_r$ ，則 $x_l \neq x_r$ 。上述定義的 AND 運算，也可將 *expr_l* 與 *expr_r* 推廣至複合式描述間的運算。在 OR 運算上，設 *expr_a*、*expr_b*、*expr_c* 為單一或複合描述式，我們將任何符合

$$expr_a \text{ AND } (expr_b \text{ OR } expr_c) \\ (expr_b \text{ OR } expr_c) \text{ AND } expr_a$$

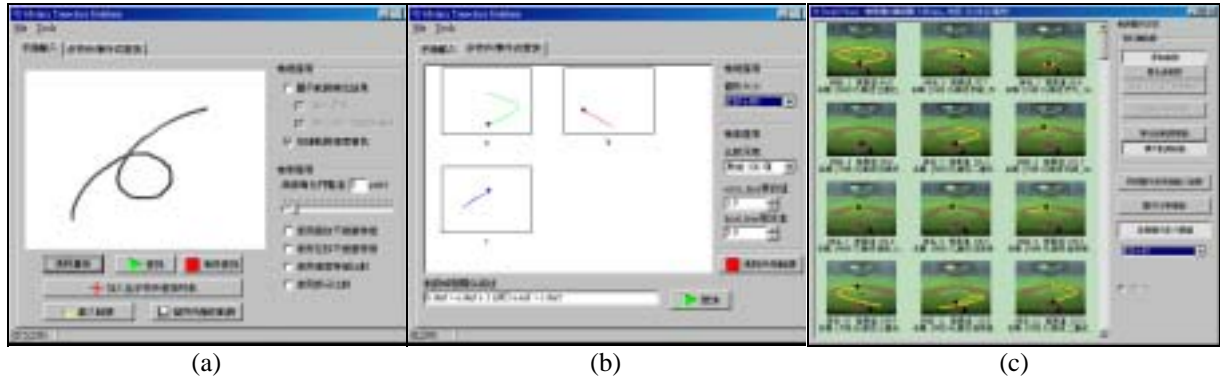
形式的式子展開為

$$(expr_a \text{ AND } expr_b) \text{ OR } (expr_a \text{ AND } expr_c)$$

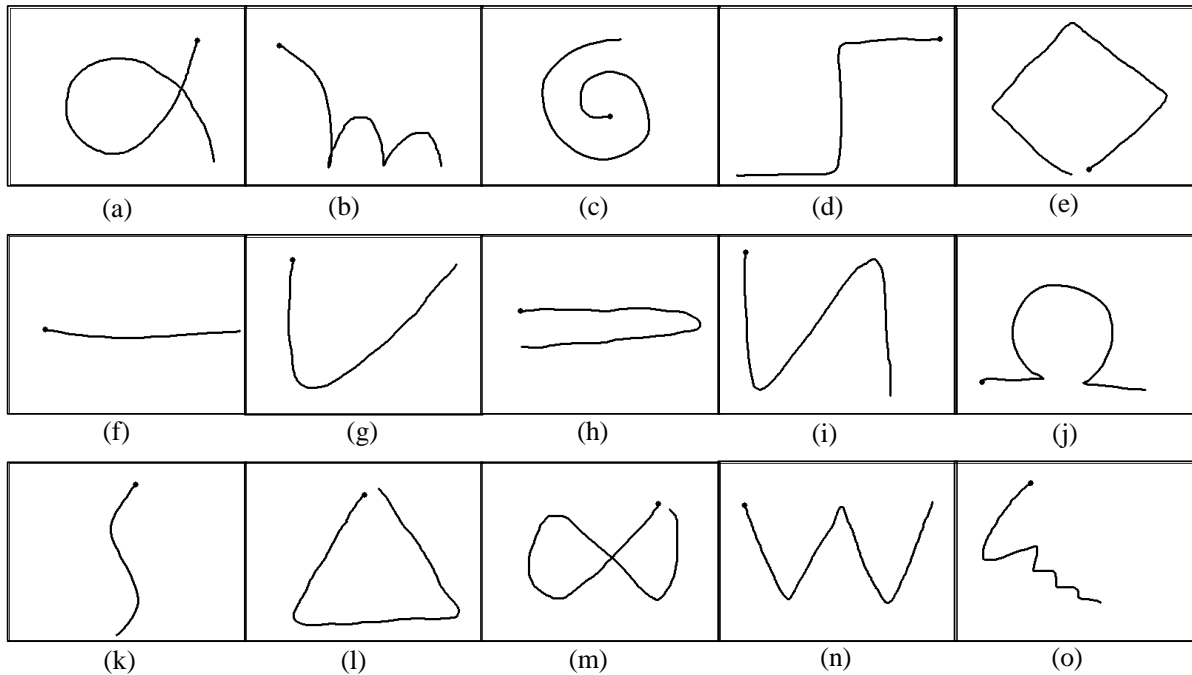
，並為每一個被 OR 運算子分隔的式子個別輸出比對結果給使用者。比對出來的每個一筆結果都是含有多個軌跡的子集合，子集合內所有軌跡中所存在的最早的一個時間點，與最晚的一個時間點所構成的時間區間，便是該筆結果所代表的影片片段。

五、實驗結果

對於上述所提的方法，我們實際在電腦上進行測試。我們首先設計視訊檢索測試平台的兩類查詢介面如圖五所示：第一類主要供使用者以手繪方式輸入欲查詢的軌跡，第二類則可輸入物件關係描述式以作為事件檢索的檢索介面。接著，則為檢索結果瀏覽介面，此介面會將資料庫中的軌跡依比對的相似程度由高至低列出。為瞭解視訊檢索各種功能實際的擷取成效，我們對個別方法作一測試：



圖五、使用者輸入介面：(a)手繪輸入查詢介面，(b)事件式查詢介面，(c)檢索結果瀏覽



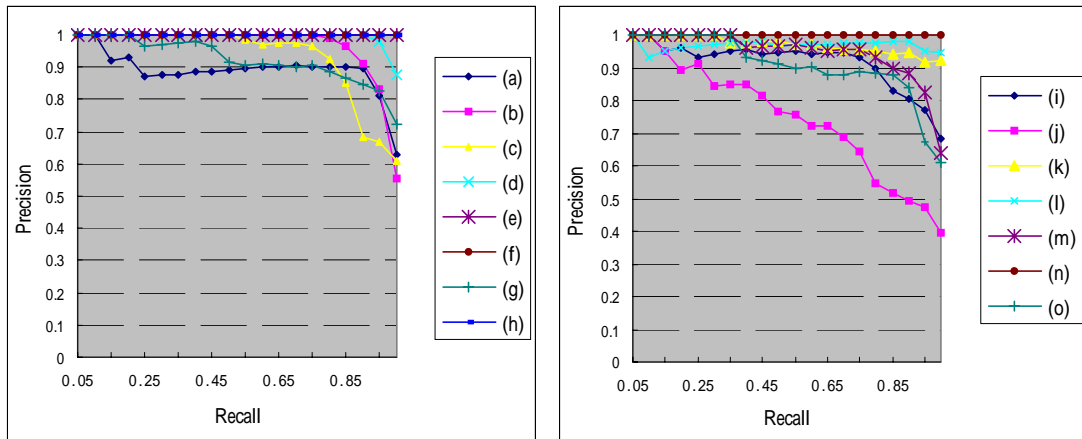
圖六、軌跡主要類型

5.1 傅立葉描述子檢索測試

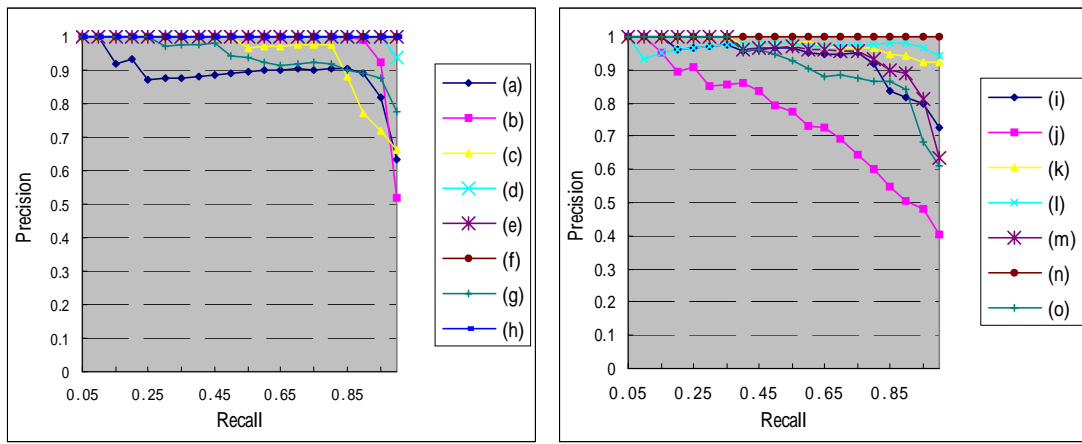
在傅立葉描述子的檢索測試中，我們以人工手繪的方式來建立軌跡資料庫，其中軌跡分為 15 個類型，每類 20 筆，總共 300 筆資料以供測試，每類軌跡的外形如圖六所示。每個類型各檢索 5 次，同時計算取 5 項 ($M = 5$) 與 10 項 FD_{inv} ($M = 10$) 的比對結果，並統計它們在各個 Recall 下的平均 Precision 值，圖七為所得出來的統計圖。我們可以觀察到，相對於只取 $M = 5$ 作比對， $M = 10$ 並不會得到顯著的進步。我們進一步找出在 $M = 5$ ，Recall 為 100% 但 Precision 未達 100% 的軌跡，與它們在 $M = 10$ 的比對結果作比較，發現 $M = 10$ 只使 Recall 為 100% 時的 Precision 平均上升 1.75% 而已，因此作為第一階層的比對，取 $M = 5$ 已足以表達大致的軌跡。

5.2 混合特徵曲線檢索測試

此部分的測試中，我們將曲線簡化門檻值定為 5 像素的誤差容忍值，然後分別手繪輸入同樣外形，但是不同速率的軌跡，藉以同時測試速率特徵檢索的成效。但由於原始軌跡速率的變化幅度不如座標資料來得大，因此我們將所有的軌跡速率值皆乘以 10 倍權重值，以強化速率資訊對檢索的影響力。圖八即為使用速率特徵比對的檢索結果，圖的左方為輸入的軌跡，右方為比對後排名前 15 的軌跡(由左而右，由上而下)，軌跡顏色由黑-白-紅分別代表速率的慢至快。由圖中可觀察到，速率特徵的使用，明顯反應在比對結果上，部分“ ”形狀的軌跡會被檢索進來則是因為其外形有一定程度的相像，加上速率特徵非常吻合所致。若是不使用速率比對，在 Recall 為 100% 的條件下，Precision 皆高於 90% 的成效。

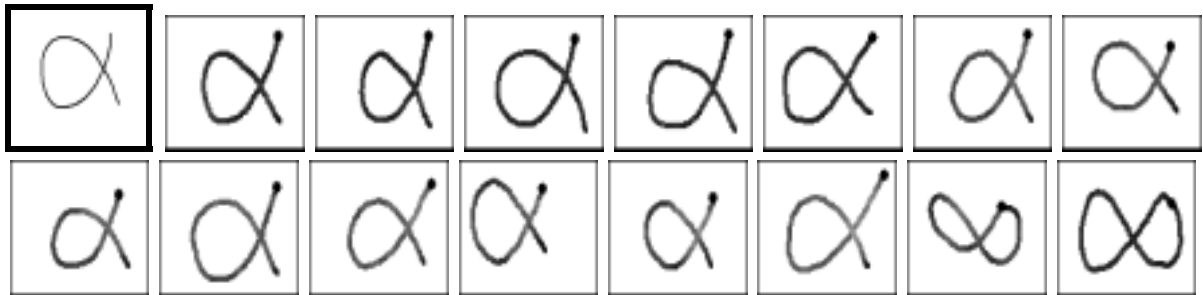


(a)

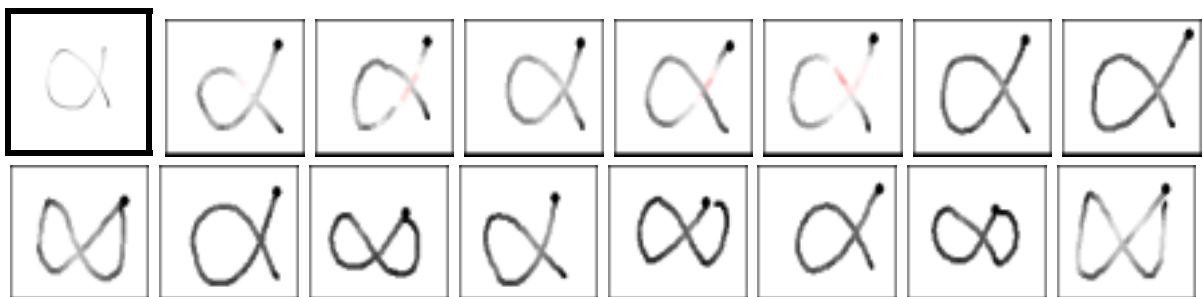


(b)

圖七、傅立葉描述子檢索實驗結果：(a) $M = 5$, (b) $M = 10$ 。



(a)



(b)

圖八、混合特徵曲線實驗結果，其中粗邊框為查詢軌跡，檢索結果由左至右依序排列：(a)低移動速率軌跡、(b)高移動速率軌跡

5.3 事件檢索測試

在此實驗裡，我們使用實際的影片來作為實驗對象。鑑於運動影片含有較多的事件，而其中棒球比賽又是目前最受國人喜愛的運動球賽之一，因此我們採用棒球影片為例，實地至職業棒球比賽場地，以數位攝影機架設於本壘正後方之看台高處，並固定視點拍攝全球場。然而受限於設備解析度的不足，影片中的物件需以人工方式協助追蹤，同時考慮到我們所提出的檢索方法是屬通用性方法，並無針對特定類型的影片作最佳化。影片裡，可擷取出的物件有攻方球員、守方球員與裁判共三個種類，首先利用球衣顏色[12]可將畫面上的三類物件區別出來。我們自影片中擷取出的軌跡，共有 121 段，若以打擊者的跑壘動作為主來分類，概略可分出 68 個事件，其中包括安打 24 件、全壘打 1 件、保送 9 件、刺殺出局 13 件、接殺出局 20 件、界外球 1 件。我們將測試部分比對、移動速率比對，以及多物件事件在棒球事件偵測的應用。我們以得分事件來作為偵測目標，由於得分必定是有跑者經由三壘到達本壘，因此便可使用部分比對來檢索出所有會經過這一段路徑的軌跡。我們輸入的軌跡與檢索結果如圖九所示。在結果中，僅有兩個非得分事件的片段被檢索進來，但其跑者的運動仍完全符合所輸入的軌跡，只是因為同一時間中的打擊者遭刺殺或接殺並達到三人出局，而沒構成得分條件。我們以保送事件來作為偵測目標，被保送的球員通常會從容地跑步至一壘，因此我們可繪入由本壘至一壘的慢速軌跡，並使用速率比對來檢索。輸入的軌跡與檢索結果如圖九所示，在達到 100% Recall 的範圍中，有 3 個非保送的事件，分別為遭高飛接殺及內野接殺的打擊者及在二壘遭刺殺的跑者，前者由於在球擊出後即了解自己必定出局，因此放慢速度跑向一壘的移動軌跡也正好符合檢索條件，後兩者則是因為已出局而慢慢地走回球隊休息區，雖然移動路徑與所輸入檢索的軌跡不是很類似，但其慢速移動的速率而使得兩者意外檢索進來。

此外，我們以多個在同時期發生的事件來模擬多物件事件測試。這裡以「壘上有跑者時擊出外野高飛球」的事件作為偵測目標。在打擊手擊出外野高飛球並離壘後，位於其他壘上的跑者需在球落地或是被接住後才能離壘，我們以這個規則來嘗試偵測這類特定事件。檢索條件與結果如圖十，其中

c.start>a.start 此一段描述式並無法檢索出期望的目標影片。檢視資料庫後得知，在目標影片片段中，軌跡 c 所代表的二壘跑者正如同一般情況，預先離壘以拉近與下個壘包的距離，遇到高飛接殺會有回壘動作，然後才再離壘，但此小段影片中的跑者回壘後停留的時間太短，因此軌跡切割方法並沒有偵測到此一時間點，於是回壘與前進下一壘的軌跡連在一起沒分開造成軌跡的起始時間點比打擊者還要早，故無法檢索出來。

六、結論

在此一研究中，主要探討物件運動軌跡為基礎的視訊檢索與事件偵測方法。在視訊檢索方面，採用質心距離的傅立葉描述子作為資料庫的初步篩選；所提的混合式曲線用以改善傳統多特徵分離比對，結合動態規劃法可完成兩軌跡的精確比對。結合上述而成的雙階層比對方法，得以在檢索效率與擷取成效間取得較佳的平衡點。在事件偵測方面，則是利用視訊檢索平台開發事件的描述工具。藉由運動軌跡查詢語言，對特定視訊內涵建構事件模式，以期可自動偵測出高階意涵。

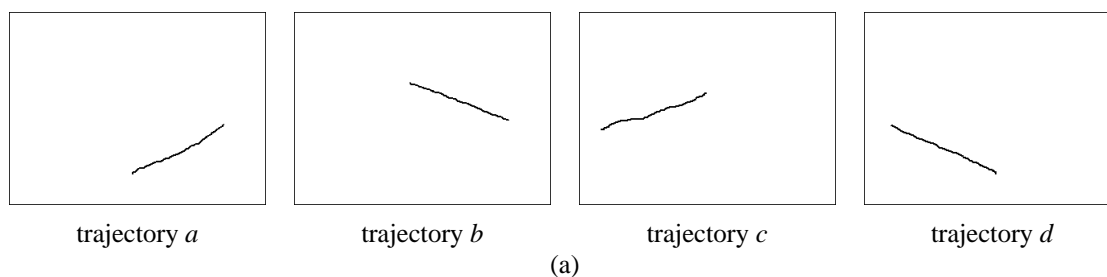
此外，為了驗證所提方法的可行性，我們首先以手繪軌跡資料庫進行測試。實驗結果顯示，對於一般性軌跡仍有不錯的效果。接著，我們以實際棒球視訊為例進行棒球事件的測試。由於我們所提出的方法是以通用性為研究目標，然而部份棒球賽裡的物件行為卻是有著相當的多樣性與不定性，因此只對棒球影片作最低限度的處理下，難免沒辦法對所有的棒球事件都訂定出相對應的查詢模型。但在實驗中得到的結果，仍可看出我們提出的方法在較為簡易的事件上能有一定程度的適用性。少數無法成功偵測的主要原因是因為無法單獨使用局部的運動軌跡特徵來充分描述事件的內容，因此未來的研究重點將擴及其它特徵，以進一步提升電腦在視覺的認知能力。

七、致謝

本研究承行政院國科會(NSC94-2213-E-214-034)、義守大學(ISU 94-01-17)經費補助，特此致謝。

比對方式	輸入的軌跡		檢索結果	
			正確筆數	Precision (於 Recall 為 100%時)
部分比對			13	86.67%
速度比對			9	75%

圖九、單一移動物件的事件偵測



檢索描述	檢索目標於資料庫中的筆數	Precision (於 Recall 為 100%時)
b.start > a.start e:0 l:8	0	
c.start > a.start e:0 l:8	1	failed
d.start > a.start e:0 l:8	1	100%

(b)

圖十、多物件事件測試：(a)輸入的軌跡與對應的代號，(b)檢索結果

參考文獻

- [1] William Chen and Shih-Fu Chang, "Motion trajectory matching of video objects," *SPIE 13th Annual International Symposium (SPIE)*, 2000.
- [2] Y.S. Cho, S.H. Lee, J.S. Shin and Y.S. See, "Shape coding tool: using polygonal approximation and reliable error residue sampling method," *ISO/IEC JTC1/SC29/WG11 MPEG 95/565*, Munich, Germany, 1996.
- [3] Serhan Dagtas, Wasfi Al-Khatib, Arif Ghafoor and Ashfaq Khokhar, "Trail-based approach for video data indexing and retrieval," *International Conference on Multimedia Computing and Systems (ICMCS)*, 1999.
- [4] Zhe Gu, "Video database retrieval based on trajectory analysis," master thesis, Fudan University, 1999.
- [5] Young-Kee Jung, Kyu-Won Lee and Yo-Sung Ho, "Content-based event retrieval using semantic scene interpretation for automated traffic surveillance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 2, Issue 3, pp.151-162, 2001.
- [6] John Z. Li, M. Tamer Ozsu and Duane Szafron, "Modeling of moving objects in a video database," *IEEE International Conference on Multimedia Computing and Systems*, pp. 336-343, 1997.
- [7] Wei-Bin Liang, "Video retrieval based on object's motion trajectory," master thesis, I-Shou University, 2002.
- [8] Michael R. Lyu, Edward Yau and Sam Sze, "Video and multimedia digital libraries: A multilingual, multimodal digital video library system," *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, 2002.
- [9] Cina Motamed, "Video indexing based on object motion in video-surveillance context," *Conference RIAO 2000 (Content-Based Multimedia Information Access)*, 2000.
- [10] Lawrence A. Rowe, John S. Boreczky and Charles A. Eads, "Indexes for user access to large video databases," *Storage and Retrieval for Image and Video Databases (SPIE)*, 1994.
- [11] Emile Sahouria and Avidah Zakhor, "A trajectory based video indexing system for street surveillance," *IEEE International Conference on Image Processing (ICIP)*, 1999.
- [12] Lei Wang, Boyi Zeng, S. Lin, Guangyou Xu and Heung-Yeung Shum, "Automatic extraction of semantic colors in sports video," *IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, 2004.
- [13] Chikashi Yajima, Yoshihiro Nakanishi and Katsumi Tanaka, "Querying video data by spatial-temporal relationships of moving object traces," *VDB*, pp. 357-371, 2002.
- [14] Atsuo Yoshitaka, Masanori Yoshimitsu, Masahito Hirakawa and Tadao Ichikawa, "V-QBE: video database retrieval by means of example motion of objects," *International Conference on Multimedia Computing and Systems (ICMCS)*, 1996.
- [15] D. S. Zhang and G. Lu, "A comparative study on shape retrieval using Fourier descriptors with different shape signatures," *In Proc. of International Conference on Intelligent Multimedia and Distance Education (ICIMADE01)*, pp.1-9, Fargo, ND, USA, June 1-3, 2001.