

一個表示影片中空間事件之字串

A String for Representing Spatial Events in a Video

楊振偉

淡江大學資訊管理學系

ericwei.yang

@mail.im.tku.edu.tw

陳彥睿

淡江大學資訊管理學系

focke.chen

@mail.im.tku.edu.tw

梁恩輝

淡江大學資訊管理學系

ehliang

@mail.im.tku.edu.tw

摘要

影片中每個畫面都可視為一張靜態影像，2D String 是一個影像的空間表示法。利用這些 2D String 進行推論，可以得到影片中物件間的空間關係變化，我們稱為空間事件。本論文利用一空間事件表(Spatial Event Table, SET)記錄兩相鄰畫面間中所發生的空間事件。在此表中，每一空間事件均會由一唯一的事件編號代表；接著，將每張空間事件表中的這些事件編號依大小順序組成為事件字串(Event String)。因此，當查詢影片時，便可根據查詢的空間事件，以其事件編號，對事件字串進行二元搜尋(Binary Search)，查詢事件字串中是否包含欲查詢之空間事件，達到以空間事件搜尋影片的目的。最後我們將分析說明本論文提出的方法是非常有效率的。

關鍵字：2D String、空間事件、空間事件表、事件字串

Abstract

In a video, each frame can be regarded as a static image. 2D String is a spatial representation for an image. The change of spatial relationship between objects in a video can be inferred from 2D String. The change is named the Spatial Event. In this paper, a Spatial Event Table is proposed to record spatial events occur between two adjacent frames. In this table, each spatial event is represented by a unique number. Afterwards, those numbers in the table are exacted and arranged as an Event String in increasing order. Therefore, when the user queries video with a spatial event, binary search is performed on the Event String to check if the Event String contains the number for the spatial event. Hence, the task of querying video with spatial event can be achieved. Finally we will analyze and explain the efficiency of the proposed method.

Keyword: 2D String, spatial event, spatial event table, event string

1. 緒論

由於資訊技術的迅速發展，許多的多媒體資料被建立出來，然而，多媒體資料存在著許多型態，例如：靜態影像(image)、音訊(audio)，以及影片(video)。資料庫中存放著大量的多媒體資料，這些資料已經被應用在許多層面上，但其中包含了許多資訊及個人導向的內容，導致使用者在使用上的困難，也增加了在影像資料庫中進行影片尋取的複雜度。由於龐大的資料量，也使得使用者無法快速地找到自己所需要的資料，此時如果有一個合適的尋取方法，將會大幅度地提升這些資料的可用性。因此，多媒體資料的索引(index)及尋取(retrieve)的技術，變的越來越重要。

由於影片資料不易直接處理，因為其中包含的資訊量非常大，且不具任何結構化格式，因此，如何在影像資料庫中快速地且有效率地進行搜尋與存取，成為了現今研究中相當重要的主題。以往對資料庫的查詢大多都是使用文字為基礎的檢索，這樣的方式對於一般文件是非常有效率的，但應用於影像資料的檢索會有很多的缺點，若當影片本身資訊不足，或資訊與內容不相符時，將無法建立正確的索引，無法幫助使用者進行影片尋取的工作；近年來已將研究趨勢轉向使用以基於內容的尋取方式(Content-Based)，根據影片中的內容來建立索引，利用分析影片中的特徵來協助索引的建立，影片資料的內容特徵大致上可以歸納為兩類：低階的視覺特徵(low-level visual features)以及高階的關係特徵(high-level relationship features) [12]這兩類，低階視覺特徵包括影像中物件的顏色、紋理、形狀等；一般使用高階的關係特徵來衡量影像相似度之技術多採用物件間之二元空間關係(binary spatial relationships)，物件之間的空間關係包含拓撲關係(topology)、方向關係(direction)、距離關係

(distance)等。使用低階視覺特徵做為索引的例子包括有 QBIC(Query By Image Content)[6]、VisualSEEK[20]等系統，使用高階關係特徵做為索引技術的例子則包括了有 2D 字串[3]、2D C 字串[8]、[9]以及[16]。

在近年來的研究中，「基於內容的相似尋取」常被應用於影像相似尋取上，此種技術通稱為「基於內容的影像相似尋取(Content-Based Image Retrieval, CBIR)」，這樣的觀念被延伸到影片資料庫的相似尋取問題上[6]。因此，我們可以將 CBIR 的技術，應用於影片相似尋取的問題上，我們可以利用範例搜尋(Query-by-Example)的方式來進行影片相似尋取，或者使用者只須提供欲搜尋的查詢條件，便可以從影像資料庫中找尋與查詢條件相符的影片。

根據影片內容做為搜尋條件的方法，概略上能夠分為三種類型[4]：

- Type 1：基於視覺特徵的搜尋：將整部影片拆解成片段或單張畫面，逐張比對相似度較高的 Key Frames，並將其整合，利用 Key Frames 做為此片段的索引。
- Type 2：基於關鍵字或自由文字的搜尋：以影片內容的文字描述做為索引，使用者可以透過文字來進行影片搜尋。
- Type 3：基於影片中物件及物件關係的搜尋：透過分析影片中的物件動作及關係，來呈現事件或概念的搜尋。

在三種搜尋型態當中，第三型態最常被用來做語意搜尋(Semantic Search)，因為它能夠從整部影片中取出具有意義的事件或概念來做為搜尋條件，讓來自各種背景的使用者都可以

藉由更友善的方式來進行影片搜尋，也能夠提供更多樣化的搜尋條件組合。

然而，影片資料的索引及相似尋取的方法，也可以從影像資料的索引及相似尋取技術延伸而來[2]，過去的研究當中也有以影像相似尋取的方法，而應用於影片相似尋取的問題上的也在過去的研究中被提出來，例如：在[15]中，利用 3D-List 來進行影片相似度之計算；而在[13]中，則是將一段影片切割成多段，並利用 3D 矩量保持法(Moment Preserving)，來找出影片中關於內容的時間或空間特徵；而在[1]的研究中，提供了三種不同的相似度上的成本計算方式，目的是去比對影片中每個畫面的相似度，藉此來達到相似尋取的功能。

相似尋取在影像資料庫系統中是一項相當重要的功能。在影像資料庫中，空間關係是常被用來對影像資料做索引及相似尋取的方式之一，由 Chang 等人提出的 2D String 就是其中的一個例子。2D String 是將影像中的物件分別投影至空間座標的 x 、 y 軸，以各種不同意涵的符號記錄投影資料，而用這些符號代表兩兩物件的空間關係。依照相同的概念，有許多改良 2D 字串的方法被提出，例如 2D C-string、2D B-string[14]及 2D C^+ -string[11]等。

以往當我們在做影片的相似尋取時，使用的方法是線性搜尋，依照時間序列來對影像進行比對，這樣並非是一個有效率的方式，因此建立一個新的字串及索引結構，利用此字串就能表達出在該段影片中，物件彼此之間移動的空間關係，及使用索引結構進行搜尋，以提升搜尋效能、減少相似比對次數，皆為本論文之重點。

在現有的影片索引技術中，基於語意描述的高階搜尋技術越來越受推崇，但當開始利用人工的方式來建立影片索引時，卻在整合數種索引時發生了索引不一致的問題，因此，如何

建立良好且有效的語意索引，成為了當前的重點課題。

我們可以從影片中建立出語意索引，來協助我們進行一些基於語意的影片尋取。其中以影片中物件間之空間關係定義出之空間事件是一常用的查詢條件。我們可以將影像視為由一連串的畫面(Frame)所組成，然而每一個畫面又是一個靜態影像[5]。在眾多的影像表示法中，2D String 為其中一種最常被使用的方法，我們將其應用於影片上，使得每一個畫面都會有一組 2D String。我們可以從影片中的表示字串推論得知影片中的物件空間關係發生變化，我們將其稱之為空間事件(Spatial Event)，並且利用空間事件表(Spatial Event Table, SET)來記錄這些空間事件，我們利用空間事件來做為語意索引，即可利用事件來做為搜尋影片時所使用的搜尋條件，進階地，我們將這些空間事件表分別各轉換成一事件字串(Event String)，建立本論文所提出之 2D E-String 及搜尋方法，藉此達到有效率地影片尋取之目的。

本論文的架構如下：第二節是相關研究的探討；第三節將討論如何利用空間事件表記錄影片中的空間事件，並產生事件字串，來執行影片尋取；最後第四節是本研究的結論。

2. 相關研究

本章將介紹在影像資料庫和影片資料庫中與物件之空間關係相關之研究。首先會介紹在影像資料庫領域裡的 2D String 表示法以及 2D B-String 表示法。其中 2D String 為本論文所採用的空間關係字串表示法，所以特別在這一章做介紹。最後將介紹在影片尋取的領域中，根據各種情況來對影片中的內容做索引的研究。

2.1 2D String

2D String是一種以物件為導向，根據物件之間的空間關係來查詢影像的方法，最早是由Chang等人所提出。在分析影像之前，一張影像必須經過影像處理和圖形辨識等兩階段處理，並從中確認包含於各張影像中的所有物件後，確保影像可以被表示成符號圖(symbolic picture)，其中，每個符號皆代表原圖中的一個物件，並且每個物件皆是以最小邊界矩形MBR(Minimum Bounding Rectangle, MBR)來代表整個物件以及其位置。

2D String意指在2D中是一維字串的有序序對(ordered pair)，我們將它表示成(u, v)。它將影像中的物件先以MBR的方式做概略化的動作，再計算出MBR的重心，並分別對x-軸及y-軸做投影。其中，第一個一維字串u，表示物件投影至x-軸上的空間關係，第二個一維字串v，表示物件投影至y-軸上的空間關係。2D String的結構可以表示為(S, R)，S與R的意義如下：

- S：符號圖中各個符號的集合。
- R：空間關係符號的集合，用來描述物件與物件之間的空間關係，包括“=”、“<”及“:”
 1. 「=」用以表示兩物件在x-軸或y-軸投影到同一個位置上。
 2. 「<」用以表示兩物件在x-軸上有“左右”的空間關係，或者是在y-軸有“上下”的空間關係。
 3. 「:」用以表示兩物件於二個維度上有同一位置，即表示兩物件有“重疊”或“覆蓋”的情形。

DE		
	B	
A		C

2D u-String：A=D:E<B<C

2D v-String：A=C<B<D:E

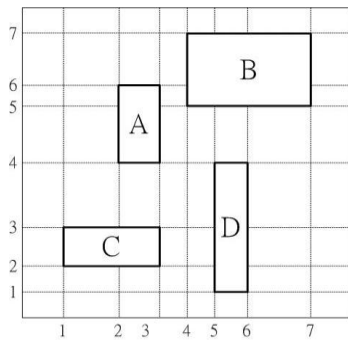
圖一、2D String之範例

以圖一為例，此圖的2D u_String表示法為A=D:E<B<C，2D v_String表示法為A=C<B<D:E。先將所有物件投影於x-軸上，物件D和物件E有相同的位置空間關係，以「:」來表示兩物件有重疊，即以D:E表示，物件A向右延伸可遇到物件B和物件C，以「<」來表示左右的空間關係，即以A<B<C表示，物件A和物件D和物件E有相同的空間關係，以「=」來表示投影於同一位置，即以A=D:E表示。同理，所有物件投影於y-軸上，物件A和物件C有相同的空間關係，以「=」來表示投影於同一位置，即以A=C表示，物件D和物件E有相同的位置空間關係，以「:」來表示兩物件有重疊，即以D:E表示。

2.2 2D B-String

2D B-String表示成(u, v)，其中u和v分別代表影像中物件投影至x-軸和y-軸後，物件間之起始邊界和結束邊界所形成的字串。它簡化空間關係的符號，僅保留“=”符號，“=”於此代表物件投影在同一點的邊界，例如圖二之2D B_u-String為CAA=CBDDDB，2D B_v-String為DCCD=A BAB。

2D B-String是利用物件的起始邊界與結束邊界的序值之比較來推論空間關係。在一維字串u和v中，各個符號序值的定義為該符號在字串中的位置減去在此符號前出現的“=”個數，且計算該符號在字串的位置時，不包括“=”在內；也就是影像中各個物件投影至x-軸和y-軸的起始邊界和結束邊界的排列順序。



圖二、2D B-String 之範例

投影之後的物件以兩個符號表示，分別為代表物件的起始邊界和結束邊界，因此一個物件在一維字串 u 和 v 中各有兩個序值，分別是起始邊界的序值和結束邊界的序值。在圖二中，令 $\text{begin}(X)$ 和 $\text{end}(X)$ 分別代表物件 X 的起始邊界之序值和結束邊界之序值。

因此圖二的 u 字串各物件之序值如下：

$\text{begin}(A)=2$ 、 $\text{end}(A)=3$

$\text{begin}(B)=4$ 、 $\text{end}(B)=7$

$\text{begin}(C)=1$ 、 $\text{end}(C)=3$

$\text{begin}(D)=5$ 、 $\text{end}(D)=6$

v 字串各物件之序值如下：

$\text{begin}(A)=4$ 、 $\text{end}(A)=6$

$\text{begin}(B)=5$ 、 $\text{end}(B)=7$

$\text{begin}(C)=2$ 、 $\text{end}(C)=3$

$\text{begin}(D)=1$ 、 $\text{end}(D)=4$

利用這些序值，我們更能詳細地推導物件間的各種空間關係，如：分離、接合、包含等空間關係，幫助我們得到更多空間資訊。

2.3 影片空間索引

在影片相似尋取的領域中，根據不同的情境，發展出各式各樣的索引方法，例如：在[10]

的研究中，為了解決基於語意(Semantic-Based)的搜尋問題，利用系統來分析影片中之片段畫面的動作，根據不同的動作標記上不同的關鍵字，並透過活動模組來將這些資訊自動學習進系統內，來協助建立影片索引，當使用者在搜尋時所輸入的搜尋條件與系統內的關鍵字相符時，便可以尋取到該段影片。

在[21]的研究中，則是根據影片中的各項低階特徵來做偵測，例如像顏色、畫面邊界、二階化等，系統自動地將影片中的部份片段做分類，並將相似的片段整理為一群，而在搜尋時，使用者便可以藉由系統建立好的分類來搜尋相關片段；而在[7]中，更是提供了更多的環境條件來供使用者做設定，例如：搜尋精準度、取影格的間隔等，提供使用者做更進一步的搜尋。

在[17]的研究中，則是將基於內容的影片尋取應用在語意網路上，相對於一般的關聯性資料庫，語意網路能夠更有彈性地將資料儲存起來，並且可以在不更動底層的資料模式及結構的情況下，加入從影片中搜尋出來的新影片特徵，能夠更容易的擴展系統的規模及應用。

而在[18]和[19]的研究中，利用2D B-String來表示影片中的物件空間關係，並且利用編輯字串的方式，記錄每張畫面(Frame)的空間關係變化，並透過物件排序表(Object Rank Table, ORT)來還原每張畫面的空間關係。在第一張畫面時記錄完整的2D B-String，而從第二張畫面之後便開始記錄前後相鄰兩張畫面中的2D B-String之差異，並將字串變化的過程儲存下來，而這些變化的過程稱之為編輯操作(edit operations)，讓我們可以從第一張畫面開始，並藉由各畫面之間的編輯操作，能夠推導出第 N 張畫面的2D B-String。編輯操作包含有 Insert(新增符號)；Move(移動符號)；Delete(刪除符號)，但這種表示法無法直接表示出一些空

間資訊，例如影像中物件之間的空間關係，使得我們無法直接從編輯操作得知該畫面中的物件空間關係，所以必須參照物件排序表來還原每張畫面中物件間的空間關係，物件排序表中將會記錄每個物件在投射於x-軸與y-軸上的之參考字串中的起始位置與結束位置，以及每個符號的排序值(Rank)，利用這些資訊，配合上每張畫面的編輯操作，以達到推論空間關係的功能。

更重要的是，這種方式只能呈現每張畫面的物件空間關係，無法表現出影片中所發生的物件間空間關係之變化，當空間關係發生變化時，也無法從各畫面之間的編輯操作推導得知，當以空間事件做為搜尋的條件時會相當複雜及困難。所以本論文提出了以空間事件做為搜尋條件的概念，並且以二元搜尋法來改善在影片查詢時的效率。

3. 影片中空間事件及其字串表示法

這節將會介紹如何利用前後相鄰的兩張畫面之 2D String 來定義空間事件，再利用空間事件表(Spatial Event Table, SET)來記錄我們從影片中推論出來的物件間之空間事件，接著再產出事件字串(Event String)。我們可以藉由輸入欲查詢之特定空間事件，將所輸入的空間事件對應其在空間事件表中所對應之事件編號，而後對事件字串做二元搜尋(Binary Search)，目的是要查詢事件字串中是否有包含欲查詢之空間事件編號，我們藉由搜尋字串的方式，取代以往直接搜尋影片的方式，改善線性搜尋的速度，提升影片查詢的效率。

3.1 空間事件

影片中的每個畫面都可視為一個靜態影像，我們可以使用 2D String 表示法來表示每張

畫面，使得每張畫面都有獨立的表示字串。在影片中，不論是人物、車子、或其他物體，都存在著運動行為，在物件進行運動行為時，會改變物件本身在畫面中的位置，而當位置改變時，便會影響到物件之間的空間關係。但當我們用 2D String 來表示這些畫面時，由於表示的是單獨一張畫面的物件空間關係，並無法直接表示在前後相鄰的兩張畫面中物件所發生的空間關係改變，所以我們必須根據前後相鄰兩張畫面的表示字串進行推論，方能得知其空間關係之改變，我們將這種變化稱之為空間事件(Spatial Event)。

本論文中所提到的空間事件，限制定義為兩物件間之”超越事件”。在 2D String 的定義中，其字串中，物件 A 和物件 B 之空間關係若為”A<B”表示物件 B 在物件 A 的右方(或前方)。假設在第 i 張畫面中物件 A 和物件 B 的空間關係不是”A<B”，而在第 i+1 張畫面中，物件 A 和物件 B 的空間關係變成了”A<B”。則我們定義在第 i 張畫面和第 i+1 張畫面之間，發生了物件 B 超越了物件 A 的事件。相同的概念也適用於 v 字串。我們用箭頭符號連接兩物件符號，用來表示超越事件，例如：B→A，即代表物件 B 超越物件 A。

3.2 空間事件表

為了有效率的記錄，並且方便地存取這些空間事件，本論文利用了空間事件表(SET)來存取這些空間事件，讓我們能夠直接且立即地從表格中，便可得知有哪些物件之間發生了空間關係變化。

在每兩張前後相鄰的畫面之間，便存在著兩張空間事件表，我們分別以 SET_u 及 SET_v 表示，我們以 SET_u 表示由兩張畫面之 2D u-String 推論所得出的空間事件表，而 SET_v 則表示由 2D v-String 推論所得出的空間事件表，分別記錄空間中 x-軸及 y-軸上的空間事件。

空間事件表是一張尺寸大小為 $n \times n$ 的表格，其中， n 為影片中同時出現最多物件數，以表一為例， $n=7$ ，代表著影片中同時最多出現七個物件，則代表空間事件表的尺寸大小為 7×7 ，在此我們將其依序編列為物件 A、物件 B、物件 C、物件 D、物件 E、物件 F、以及物件 G。

我們在空間事件表的每一個空格中，皆將付予流水號視為每一空格的編號，空格編號將會從 1 號編至 49 號，用流水號的目的是確保其唯一性，不會有編號重覆的情況。每一個空格的編號皆有其定義，各自代表特定的空間事件，例如：編號為 4 的空格，代表著 x-軸上索引 D 及 y-軸上索引 A 的位置，若該空格中有被記錄，則代表在此相鄰兩個畫面之間有發生物件 A 超越物件 D 的事件。相同的，編號 30，則代表了物件 E 超越物件 B，以此類推。

接下來將以圖三為範例來說明本論文所提出的方法，圖三為一個含有四個畫面的影片，我們可以從圖三中看出，影像中同時出現的最多物件數為 7 個，在四個畫面中，我們可以分別對每一張畫面中的物件取其 MBR，我們利用紅點標記出 MBR 的重心，接著再將每個 MBR 的重心投射在 x-軸上，便可以獲得一串 2D u-String，其即代表著在一個畫面中物件投影在 x-軸上的空間關係。我們先以 x-軸之處理做說明，而 y-軸也可依相同方法處理。從圖三的四個畫面中，我們可以發現前後相鄰的兩個畫面中，物件之間發生了空間關係的變化。

表一、SET 範例

	A	B	C	D	E	F	G
A	1	2	3	4	5	6	7
B	8	9	10	11	12	13	14
C	15	16	17	18	19	20	21
D	22	23	24	25	26	27	28
E	29	30	31	32	33	34	35
F	36	37	38	39	40	41	42
G	43	44	45	46	47	48	49

從四個連續的畫面中，我們可以得知物件的確存在著運動行為，藉由移動及靜止來改變物件自身在畫面中的空間位置。

我們依照時間順序，分別將前後相鄰的兩張畫面之 2D u-String，拿來做事件推論的動作，我們先以第一個畫面及第二個畫面的 2D u-String 做為範例來說明。

在這兩串 2D u-String 中，藉由事件推論的方式，我們可以得知這兩個畫面間發生了以下空間事件：物件 A 超越了物件 C、物件 B 超越了物件 C、以及物件 F 超越了物件 D，而其他物件則沒有發生空間關係的變化，而我們可以將其表示如下：

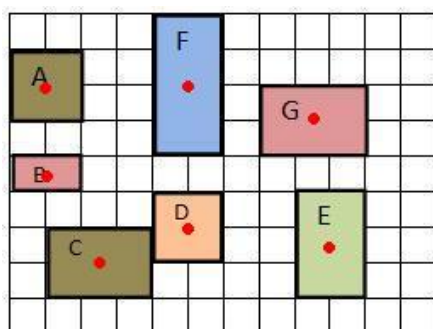
Frame1~Frame2：A→C，B→C，F→D

同樣的，Frame2 和 Frame3 之間，以及 Frame3 和 Frame4 之間所發生的空間事件也可以表示如下：

Frame2~Frame3：A→B，A→D，A→F，C→A，C→D，D→F，G→E

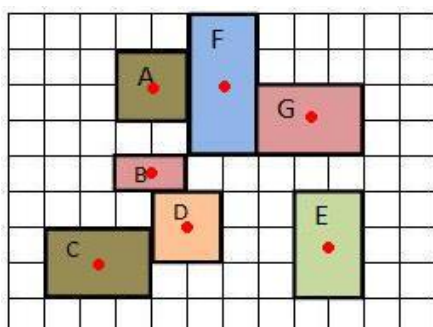
Frame3~Frame4：B→F，C→F，D→A，D→E，D→G

接著我們將利用空間事件表 SETu(i, i+1) 來記錄發生在 Frame i 及 Frame i+1 之間的空間事件。而此範例可得到 SETu(1,2)，SETu(2,3) 及 SETu(3,4)，分別為表二、表三及表四。



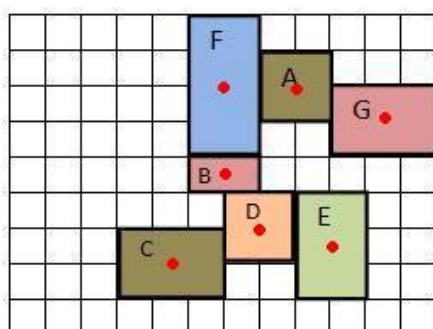
2D_u-string : A=B<C<F=D<G<E

(a)Frame1



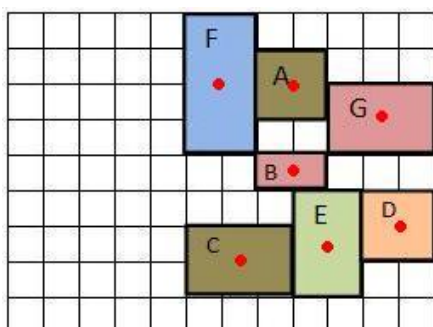
2D_u-string : C<A=B<D<F<G<E

(b)Frame2



2D_u-string : C<F=B<D<A<E<G

(c)Frame3



2D_u-string : F=C<A=B<E<G<D

(d)Frame4

圖三、範例影片

表二、SETu(1,2)

	A	B	C	D	E	F	G
A	1	2	V_3	4	5	6	7
B	8	9	V_{10}	11	12	13	14
C	15	16	17	18	19	20	21
D	22	23	24	25	26	27	28
E	29	30	31	32	33	34	35
F	36	37	38	V_{39}	40	41	42
G	43	44	45	46	47	48	49

表三、SETu(2,3)

	A	B	C	D	E	F	G
A	1	V_2	3	V_4	5	V_6	7
B	8	9	10	11	12	13	14
C	V_{15}	16	17	V_{18}	19	20	21
D	22	23	24	25	26	V_{27}	28
E	29	30	31	32	33	34	35
F	36	37	38	39	40	41	42
G	43	44	45	46	V_{47}	48	49

表四、SETu(3,4)

	A	B	C	D	E	F	G
A	1	2	3	4	5	6	7
B	8	9	10	11	12	V_{13}	14
C	15	16	17	18	19	V_{20}	21
D	V_{22}	23	24	25	V_{26}	27	V_{28}
E	29	30	31	32	33	34	35
F	36	37	38	39	40	41	42
G	43	44	45	46	47	48	49

3.3 空間事件字串及其查詢

從表二 SETu(1,2)中可以得知，分別在表中編號為 3、10、39 的方格中被記錄了超越事件，我們將這三個編號取出，按照編號順序編列成一串由阿拉伯數字以及逗號所組成的字串，我們稱之為事件字串(Event String)，進一

步地將其定義為 2D E-String，而此後皆以 2D E-String 來表示此字串。以表二 SETu(1,2)為例，我們所得到的 2D E u-String 為”3,10,39”。得到 2D E u-String 之後，我們也可以參照空間事件表，從 2D E u-String 中得知在表格中，編號為 3、10、39 的方格被記錄了超越事件，便可以推論出在兩張前後相鄰的畫面中，分別出現了物件 A 超越了物件 C、物件 B 超越了物件 C、以及物件 F 超越了物件 D 等以上三項事件，而這便是我們利用空間事件表與 2D E u-String 相互參照之後，所能達到的事件記錄功能。

有了 2D E u-String 之後，我們便可以利用空間事件來進行影片尋取。我們讓使用者利用介面輸入想要查詢的空間事件(例如：物件 D 超越物件 F，以 $D \rightarrow F$ 表示之)，系統將會利用上述所提及的相互參照功能，利用參照空間事件表的方式，將使用者所輸入之欲查詢的空間事件轉換成事件編號，而後利用這些事件編號來對我們所取得之每一串 2D E u-String 做二元搜尋(Binary Search)；目的是尋找 2D E u-String 中是否有包含欲搜尋之事件編號。我們用的搜尋方式是二元搜尋，能夠大幅度減少搜尋次數，以提升在搜尋 2D E u-String 內容時的效率，搜尋完所有 2D E u-String 後，便可以知道哪些畫面的 2D E u-String 包含欲搜尋之事件編號，也就是可以快速找到包含欲搜尋事件之畫面。

利用這種方式來完成以空間事件為搜尋條件的影片查詢，以圖三範例影片為例，我們總共可以從這段影片中取得三段事件字串，分別是：

2D E u-String in SETu(1,2)：”3,10,39”

2D E u-String in SETu(2,3)：“2,4,6,15,18, 27,47”

2D E u-String in SETu(3,4)：“13,20,22,26, 28”

假定我們想要搜尋的空間事件為物件 D 超越物件 F，藉由系統參照空間事件表之後，轉換成的事件編號為”27”，此時系統將會依序

到三段 2D E u-String 中使用二元搜尋法去尋找是否有包含”27”這項編號。結果我們只在 Frame2 和 Frame3 之間的 2D E u-String 中搜尋到我們所輸入的空間事件，便可得出結論，Frame2 到 Frame3 之間發生了物件 D 超越物件 F 的空間事件，即是我們想要搜尋的結果。在此我們只討論及應用 x-軸上的 2D E u-String，事實上，在許多應用上同時需考慮 y-軸上的 2D E v-String，而其推論及查詢方式與 x-軸相同。

3.4 分析與應用

二元搜尋法是一種非常快速的搜尋方法，當包含 n 個數值的數列按照由小至大排序之後，其時間複雜度則為 $O(\log_2 n)$ 。在上述文章中，我們所提出的事件字串便是由一張大小為 $n*n$ 的空間事件表轉換而來，其中 n 為影片中同時出現最多物件數，而又因為兩物件之間的空間事件皆以互斥方式存在，例如：當物件 A 超越物件 B 的時候，便不可能同時發生物件 B 超越物件 A，因為這並不符合物理現象，因此 2D E-String 最大可能長度為 $n^2/2$ 。而我們提出的影片查詢方法也是利用二元搜尋法的，而此影片搜尋方法的時間複雜度則為 $O(\log_2 n^2)$ ，等於 $O(2\log_2 n)$ ，也就是等於 $O(\log_2 n)$ ，所以能夠說明我們所提出的影片搜尋的方法是非常有效率的。

3.5 未來研究

上一節中所提到的範例為利用單一事件來做為搜尋條件，然而，我們也提供了利用多個事件來做為搜尋條件的搜尋方式，使用者可以一次選擇輸入多個空間事件來做為搜尋條件，並且可以選擇要進行串聯(AND)查詢，或是並聯(OR)查詢。若要一次搜尋多個空間事件，則系統會依序將各空間事件帶入 2D E-String 中去執行二元搜尋，以串聯(AND)為

例，如果搜尋成功，則將前一項空間事件搜尋完成的結果拿來接著做後一項空間事件的搜尋，以此類推，直到得出結果為止；若以並聯(OR)為例，則視為個別獨立的兩空間事件來做搜尋；以這樣的方式來達成以多個空間事件為搜尋條件的影片查詢。

4. 結論

我們提出了一個利用二元搜尋法來進行影片搜尋的方式，而這個方式是非常有效率的。在影片中，每一個畫面都可以視為一張靜態影像，所以我們可以利用 2D String 表示法來表示這些畫面，方便我們針對影像來做索引以及搜尋。2D String 只能表示在單一畫面中物件之間的空間關係，而無法記錄其在前後相鄰的兩個畫面之空間關係變化，所以我們必須利用事件推論的方式來推論前後相鄰兩個畫面的 2D String，以便於找出物件之間在影片中的空間關係變化，我們將這種變化稱之為空間事件(Spatial Event)，並且利用空間事件表(Spatial Event Table, SET)來記錄這些空間事件。而後，將這些空間事件表轉換為事件字串(Event String)，我們將其定義為 2D E-String，而其便代表了在影片中的前後相鄰兩畫面裡所發生的空間事件。有了 2D E-String 後，使用者便可以利用介面輸入欲查詢之空間事件，經由系統參照空間事件表之後，轉換為事件編號，利用事件編號來針對每一串 2D E-String 中做二元搜尋(Binary Search)，目的是查詢 2D E-String 中是否有包含欲查詢之空間事件的編號。

利用二元搜尋法來實作影片查詢的方式，其平均時間複雜度為 $O(\log_2 n)$ ，可以成功縮減一般在做影片尋取時所花費的線性時間，不但提供了以空間事件為搜尋條件的搜尋方式，並且也能夠大幅度地提升搜尋影片時的效率。

參考文獻

- [1] 黃悌維，影像資料庫中一個相似尋取之方法，淡江大學資訊管理研究所碩士論文，民國 91 年
- [2] Arndt, T., and Chang, S.K., "Image Sequence Compression by Iconic Indexing," 1989 IEEE Workshop on Visual Languages, The Institute of Electrical and Electronic Engineers, IEEE Computer Society, Silver Spring, MD, pp. 177-182, Oct. 1989.
- [3] Chang, S.K., Shi, Q.Y., and Yan, C.W., "Iconic indexing by 2D-strings," IEEE Trans. On Pattern Analysis and Matching Intelligence, PAMI-9, May, 1987, pp. 413-428.
- [4] Chen, L., Oria, V., Tamer Özsu, M., "A Multi-Level Index Structure for Video Databases", Multimedia Information Systems 2002: 28-37.
- [5] Deb, S., "Video Data Management and Information Retrieval," Idea Group Inc. 2005.
- [6] Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Streele, D., and Yanker, P., "Query by Image and Video Content: The QBIC System," Computer, vol. 28, no. 9, pp. 23-32, Sept. 1995.
- [7] Hauptmann, Alexander G., Christel, Michael G., and Yan, Rong., "Video Retrieval Based on Semantic Concepts" Proceedings of the IEEE | Vol. 96, No. 4, April 2008

- [8] Hsu, F. J., and Lee, S.Y., "Spatial Reasoning and Similarity Retrieval of Images Using 2D C-String Knowledge Representation," *Pattern Recognition*, vol. 25, no. 3, pp. 305-318, March 1992.
- [9] Hsu, F.J., and Lee. S.Y., "Similarity Retrieval by 2D C-Trees Matching In Image Database," *Journal of Visual Communication And Image Representation Vol.9, No. 1, March, 1998,pp.87-100.*
- [10]Hu, Weiming., Xie, Dan., Fu, Zhouyu., Zeng, Wenrong., and Maybank, Steve., "Semantic-Based Surveillance Video Retrieval" *IEEE Transactions On Image Processing*, VOL. 16, NO. 4, APRIL 2007
- [11]Huang, P.W., and Jean, Y.R., "Using 2D C+-string as spatial knowledge representation for image database systems," *Pattern Recognition*, 1994,pp.1249-1257.
- [12]Huang, P.W., and Lee, C.H., "Image Database Design Based on 9D-SPA Representation for Spatial Relations," *IEEE Trans. on Knowledge and Data Engineering*, vol. 16, no. 12, 2004.
- [13]Huang, Wei-Kan., Chung, Chi-Han., Cheng, Shyi-Chyi., and Hsieh, Jun-Wei., "A Fast Cube-based Video Shot Retrieval Using 3D Momentpreserving Technique" 2009 IEEE.
- [14]Lee, S.Y., Yang, M.C., and Chen, J. W., "2D B-string: a spatial knowledge representation for image database systems," *Proc.ICSC'92 Second Int. Computer Sci. Conf.,1992,pp.609-615.*
- [15]Liu, C. C., and Chen, A.L.P., "3D-List: A Data Structure for Efficient Video Query Processing," *IEEE Trans. on Knowledge and Data Engineering*, vol. 14, no. 1, 2002.
- [16]Nabil, M., Ngu, A.H.H., and Shepherd, J., "Picture Similarity Retrieval Using the 2D Projection Interval Representation," *IEEE Trans. Knowledge and Data Eng.*, vol. 8, no. 4, pp. 533-539, Aug. 1996.
- [17]Sebastine, Sancho, C., Thuraisingham, Bhavani., Prabhakaran, Balakrishnan., "Semantic Web for Content Based Video Retrieval" 2009 IEEE International Conference on Semantic Computing
- [18]Shearer, K.R., Kieronska, D., and Venkatesh, S., "Resequencing Video Using Spatial Indexing," *J. Visual Languages Comput.*, vol. 8, pp. 193-214, 1997.
- [19]Shearer, K.R., Venkatesh, S., and Kieronska, D., "Spatial Indexing for Video Databases," *J. Visual Commun. Image Representation*, vol. 7, pp. 325-335, 1996.
- [20] Smith, J.R., and Chang, S.F., "VisualSEEK: A Full Automated Content-Based Image Query System," *Proc. Fourth ACM Int'l Multimedia Conf.*, pp. 87-98, 1996.
- [21]YU, Jun-qing., HE, Yun-feng., SUN, Kai., Wang, Zhi-fang., WU, Xiang-mei., "Semantic Analysis and Retrieval of Sports Video" *IEEE Computer Society* 2006.