

# 行動商務應用之動態使用者查詢系統

翁頌舜，輔仁大學資訊管理研究所

周羿禎，輔仁大學資訊管理研究所

---

## 摘要

本研究運用「移動中的物體」、「座標位置」與「時間」三個主軸，提出一個移動物體對靜態資料動態查詢的系統架構—動態使用者查詢系統(Dynamic User Query System)。在本系統中，所有靜止或移動中的物體都可以自行決定起始查詢地點的位置、速度、時間區間、距離及想查詢的項目，成為下查詢指令的使用者，並且可以支援：(1)多重動態查詢；(2) 移動式查詢範圍；本研究的實驗應用領域為行動服務業，藉由動態使用者查詢系統支援移動中的物體依當時的需求進行動態查詢，以期獲得更精確的查詢結果。本研究實做領域為手機通訊，藉由手機的普遍性及可攜帶性，落實讓使用者動態查詢的行動服務目的。

**關鍵字：**定址化服務、動態查詢、查詢系統、行動服務

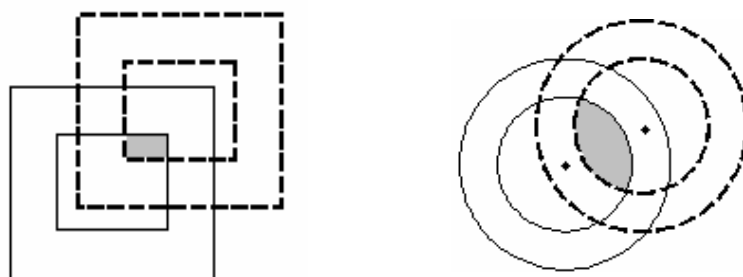
---

## 1. 緒論

無線通訊產業及行動商務能有效提昇行動工作者的效率、解決有線設備鋪設線材的問題，降低維護成本等優點，市場成長快速，許多相關的加值行動服務（Value-added Mobile Services）以及與「定址化服務」（Location-based Services, LBS）相關的產品和服務將能增加使用者的便利性，並且有效提升其產品的應用價值。本研究運用「移動中的物體」、「座標位置」與「時間」三個主軸，讓使用者可以自行決定查詢的目標、時間和範圍，增加查詢方式的彈性，以期得到更符合使用者期望的動態查詢結果。

本研究的目的是為以下幾點：1.任何移動或靜止狀態的物體皆可被視為使用者，並可在任何時間及任何地點查詢。2.考慮使用者的移動狀態，使計算出的查詢區域和查詢的結果更加精確，並能即時將查詢結果回覆給使用者參考。3.讓使用者可以自行選擇要查詢的速度、地點、時間區間、查詢範圍及查詢的種類項目，讓查詢的方式更加彈性化。因此，本研究欲探討的問題有：1.如何才能在任何時間及任何地點進行查詢？ 2.如何才能計算得到精確的查詢區域？

本研究的假設與限制條件為：1. 使用者的位置可以連結 GPS 系統得到定位的服務。2. 查詢的移動區域為正方形：本研究之所以假設物體的查詢範圍以正方形型的區域移動，是因為計算移動區域比較容易、快速；此外，本研究的目的是期望能即時的將查詢結果提供給使用者，因此不考慮在查詢的時間區間內，使用者會改變其移動的方向和速度。3. 資料庫中已有所有營業商店的座標資料：假設已有一存放所有營業商店資料的資料庫，讓使用者可以直接選出各類別中需要查詢的項目。



圖一 正方形和圓形移動區域比較

## 2. 文獻探討

定位系統的發展可追溯至二次世界大戰時的軍用無線電，而定址化服務 (Location-based Services, LBS) 是在 1996 年時，美國通過一項 E911 強制措施，讓緊急救援機構能夠確認利用手機撥 911 求救人士的詳細位址的系統，而定址化服務的特色在於必須依照個別使用者目前的環境給予回覆，目前發展的定址化服務的應用領域包含手機、汽車等行動裝置，內容包括電子地圖、救援定位服務與交通控制等。

連續性查詢 (Continuous Query) 是為了讓使用者得到可被利用的資料而產生的查詢方式 [3]。「Continuous Query」於 1992 年第一次提出，當時這種查詢建立在一種只容許附加資料的靜態資料庫中 [11]。Chen et al. [3] 依其使用動機將其大致分為「改變型」(Change-based) 和「計時型」(Timer-based) 兩種，Sistla et al. [7] 則將傳統的查詢方式分為即時、連續和持續三種；Zheng and Lee [13] 則認為傳統的位置取向的查詢 (Location-dependent Query) 已經越來越難以回覆具有連續性性質的查詢問題。

Tao and Papadias [8] 認為連續型查詢的動態性及非單一查詢結果的特性很適合應用在時空資料庫中，而時空資料庫管理系統 (Spatio-temporal database management systems) 近年來應用在交通管理或飛航控制等，都是對持續移動物體的應用。視窗式查詢包括存在式查詢 (Exist Query) 和回覆式查詢 (Report Query)，

查詢的區域可以下式表示[2,5,12]:  $w = \{(x, y) / (x, y) \in [x_1, x_2] \times [y_1, y_2]\}$ 。Kriegel et al. [6] 認為應用空間資料在地理資訊資料庫的研究將會越來越受到重視，其查詢方式及其準確度和速度等相關需求作有效的執行也很重要。

時空視窗查詢法(Spatio-Temporal Window Query)，使用MinSkew演算法對動態物體做靜態查詢，而時空視窗查詢法的主要概念是針對靜止或動態的查詢區域 $q_R$ ，預測在未來的時間 $q_T$ 時，其他被偵測的物體是否會經過 $q_R$ ；其查詢的區域 $q_R$ 面積是固定的，之後再根據兩項條件判別移動的物體是否在未來時間內會經過指定的查詢區域[1,4,9]：

1. 物體移動的軌跡投射至各象限軸的區間是否都和查詢區域 $q_T$ 有重疊的狀況？
2. 各象限重疊交叉最長和最短的時間區間是否也和查詢的時間間隔 $q_T$ 重疊？

Tao et al. [9] 提出將時空視窗查詢法的選擇性估計分成三種：(1)對靜態的點進行查詢(Static Point Data)；(2)對動態的點進行查詢(Dynamic Point Data)；(3)對動態的方形區域進行查詢(Dynamic Rectangle)。對靜態的點做查詢時，將查詢區域定為方形的可移動區塊，查詢的區域可隨時間的變化而變動邊長，改變其面積，因此必須先計算在未來的某一查詢時間時，查詢區域移動所包含的所有面積，之後才對這面積內所涵蓋的點做查詢。

時間參數視窗查詢(Time-Parameterized Window Query)是連續時空查詢(Continuous Spatio-Temporal Query)的一部分，時間參數查詢考慮(R, T, C)，R是滿足即時查詢的物件集合，T是從下指令查詢開始至得到結果R的時間，C是在時間T中會影響R結果的物件集合；R是一般查詢常見的因素，而(T, C)是時間參數查詢的影響變數因子，代表下達查詢的指令到得到回覆的時間內，因環境改變而對結果正確性的影響，Tao and Papadias [8]定義一個物體o對查詢q的影響時間為 $T_{INF}(o, q)$ ，將 $T_{INF}(o, q)$ 視為距離的概念，最終目標是找出最會影響結束時間的物體，再根據C去修正即時查詢的物件集合(R)的內容，最終才能得到最正確的結果[10]。

### 3. 研究方法

#### 3.1 Dynamic User Query System 之理論基礎

本研究參考 Choi and Chung [4]、Tao et al.[10]及 Tao and Papadias [8]的概念，並綜合其理論概念加以創新改良，參考移動物體對查詢區域變動的計算，考慮查詢區域因時間改變所造成的空間變動，本研究提出 Dynamic User Query System 架構，改良目前研究中對移動中物體動態查詢的限制，以建立更有彈性的查詢方式，達成個人化查詢的效益。

表一 常用符號定義表

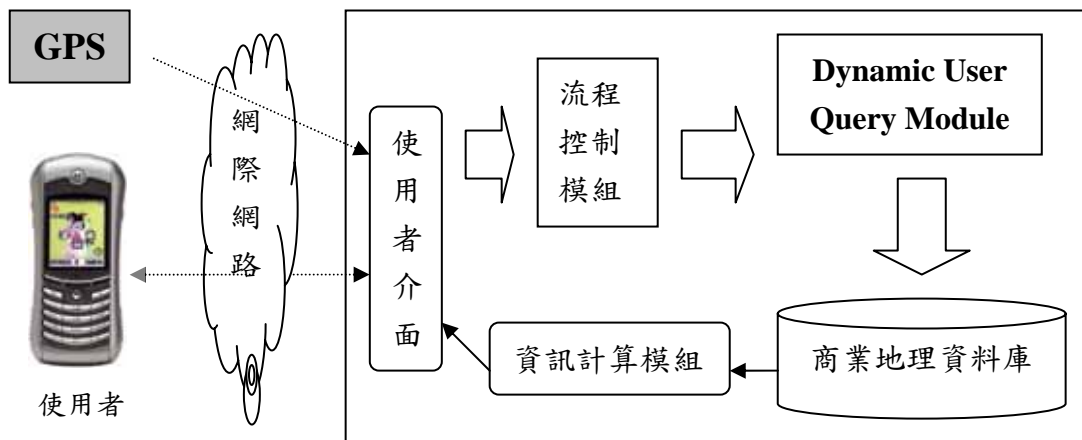
符號	表示意義	符號	表示意義
$O(x,y)$	加入系統計算的時間誤差後，物體目前所在的座標	$V[Vx, Vy]$	物體移動的2-dimension速度向量
$t'$	系統計算、搜尋資料和傳回資料所需要的時間	$t[t-, t+]$	使用者查詢的時間區段： $t-$ 是設定為開始查詢的時間區段： $t+$ 是設定為結束查詢的時間區段
$r[r_1, r_2]$	使用者查詢的半徑範圍區間	$R$	以 $r_2$ 算出的正方形區域邊長
$R'$	以 $r_1$ 算出的正方形區域邊長	$q_R$	以使用者座標為中心，最大邊長為 $R$ 的query region
$P_{qt+}$	$t = t+$ 時，結束查詢的區域座標	$P_{qt-}$	$t = t-$ 時，開始查詢的區域座標
$P_{q0}$	$t = 0$ 時查詢區域的起始座標	$(Sx, Sy)$	資料庫的商店座標

因為本研究的架構是應用在移動物體對靜態資料的動態查詢，計算查詢時間內物體移動的區間是使用正方形查詢區域的方式，應用Choi and Chung [4]的研究，將移動的區域範圍投射至x和y軸線上找出移動區域，考量方形區域移動的特性，並考慮方形和六邊形兩種不同的移動區域面積，並加入其他的限制條件，使得區域中所有符合條件的點皆為查詢所需要的答案，以提高精確度，且本研究提出的Dynamic User Query System架構，以能即時回覆查詢結果為目的，因此不考量物體會改變其移動速度；本研究參考Tao et al.[10]，考慮 $q_R$ 是以方形為移動區域，因為圓形的移動區域對本研究在計算移動不完全覆蓋的移動區域時會產生較大的誤差，本研究參考Tao and Papadias [8]，加入查詢輸出時間的誤差考量，將誤差值降低。

### 3.2 系統架構

Dynamic User Query System 的系統架構如圖二所示，以「移動中的物體」、「座標位置」與「時間」為主軸，並根據物體的移動速度與方向，讓物體在可以移動的狀態中，依目前的位置、速度或由使用者指定的條件下，在訂定的查詢時間區間、區域範圍中查詢；本系統的優點在於：

1. 支援多重動態查詢：使用者可以自行輸入各項查詢的條件，更有彈性的下達查詢指令，達到動態查詢的目的。
2. 移動式查詢範圍：計算查詢範圍時採用正方形的移動區域，這樣可以精確的在使用者查詢的時間區間內，計算查詢的距離區間範圍。



圖二 Dynamic User Query System 系統架構圖

以下說明本研究系統架構各部分的主要功能：

1. **使用者**：所有移動中的物體都可以被視為下查詢指令的使用者，使用者可選擇經由 GPS 找出目前所在的位置座標及速度，或是直接選取某一定點。
2. **GPS (Global Position System)**：依需求定出使用者所在的兩個座標，而後端的 Dynamic User Query Module 會依照該座標在時間區段中的改變計算出移動的速度和方向。
3. **使用者介面**：移動中的物體可透過前端的使用者介面輸入各項參數進行查詢，查詢結束後，使用者介面會將符合查詢的項目資料及資訊顯示給使用者。
4. **流程控制模組**：主要功能在於判斷使用者是否自己設定座標和速度及參數設定的條件控制，流程控制模組會判斷使用者的設定，並分別以兩種不同的狀況進入 Dynamic User Query Module 中做個別的判斷。
5. **Dynamic User Query Module**：Dynamic Users Query Module 提供速度向量計算和不同的演算模式計算功能，所提供的動態查詢功能包括：

- (1). 依 GPS 傳遞的兩項座標計算出速度向量  $V[V_x, V_y]$ ，假設 GPS 系統在 0.01 秒的時間間距中得到移動物體的座標為  $B(bx, by)$  和  $B'(bx', by')$ ，則可得到移動物體的速度向量為：

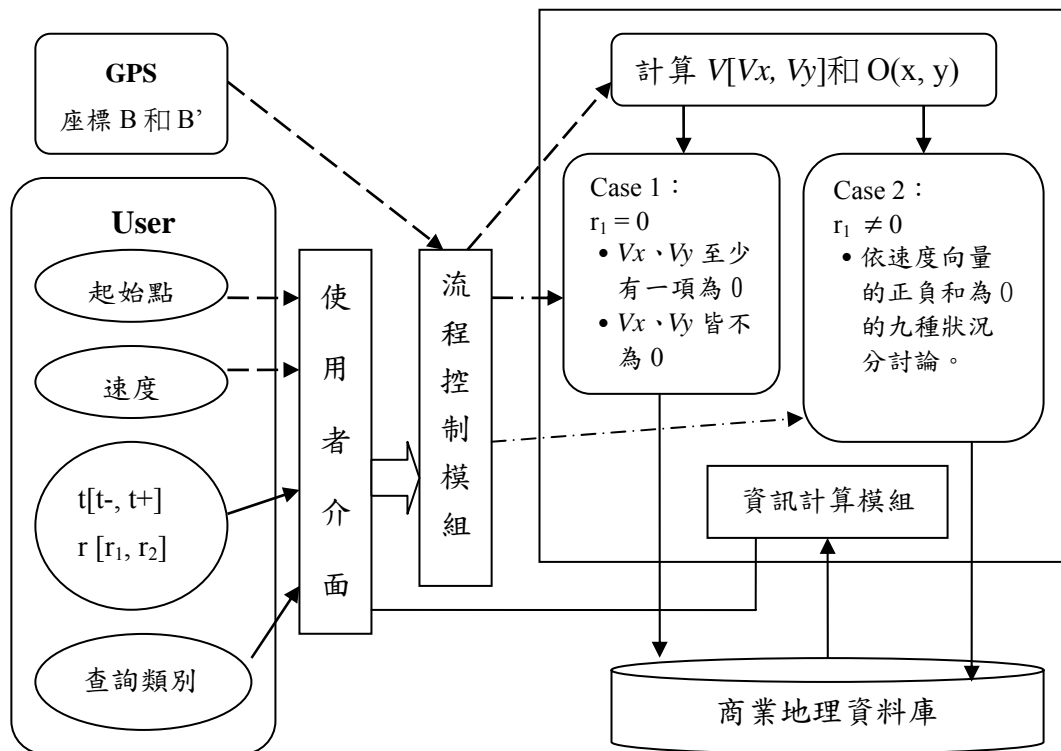
$$V[V_x, V_y] = \left[ \frac{bx - bx'}{0.01}, \frac{by - by'}{0.01} \right] \quad (1)$$

再加入計算時間的誤差  $t'$ ，算出物體的起始位置，則可得到物體移動的速度和方向及目前所在的起始位置座標  $O$ 。

$$O(x, y) = (bx + V_x \times t', by + V_y \times t') \quad (2)$$

- (2). 根據不同的演算模式及條件限制，計算符合查詢區域的座標範圍，並將座標範圍和使用者查詢的種類項目傳至資料庫做搜尋。

6. **資訊計算模組**：資料庫將符合查詢的資料座標傳送給資訊計算模組後，資訊計算模組會計算出資料點離物體的距離和方向，並依距離的長短將資料一併傳至前端使用者介面給使用者。
7. **商業地理資料庫**：儲存有關各種不同類別營業用店舖資料和記錄，資料項目包含座標、地址、電話...等資訊。



圖三 Dynamic User Query System 系統流程圖

#### 4. Dynamic User Query Module 演算模式

本系統依使用者查詢的需求，提出不同的演算模式，表二說明依兩種不同型態的查詢範圍區間和不同的速度向量，將有不同的演算模式。

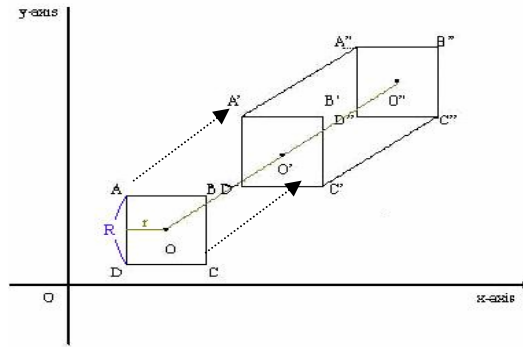
表二 演算模式分類表

使用情境	查詢區域	時間區段	速度向量
Case 1	正方形區域	使用者動態設定	$V_x, V_y$ 兩者至少有一項為0
			$V_x, V_y$ 皆不為0
Case 2	正方形框格	使用者動態設定	依速度向量的正負和為0的狀況將分成九種情況討論。

## 4.1 查詢區域為正方形區域

由前面系統架構的說明中已知O點座標 $O(x, y)$ 為起始點，透過使用者選取的半徑  $r$ ，我們可以計算出離物體最短的距離為  $r$ ，邊長為 $R$ 的正方形，如圖四所示，且 $R = 2r$ ；查詢區域 $q_R ABCD$ 的座標以向量 $P_{q0}$ 的方式呈現，因此，

$$\begin{aligned} P_{q0} &= [Ax, Ay, Bx, By, Cx, Cy, Dx, Dy] \\ &= [x-R/2, y+R/2, x+R/2, y+R/2, x+R/2, y-R/2, x-R/2, y-R/2] \end{aligned} \quad (3)$$



圖四 正方形查詢區域查詢示意圖

已知在開始查詢時間( $t = t-$ )時每秒在x軸向的位移為 $v_{xt-}$ ，每秒在y軸向的位移為 $v_{yt-}$ ；設在開始查詢時間( $t = t-$ )時物體的座標為 $O' = (x', y')$ ，查詢區域為 $A'B'C'D'$ ，則：

$$O' = (x', y') = (x + V_{xt-}, y + V_{yt-}) \quad (4)$$

$A'B'C'D'$ 的座標向量 $P_{qt-}$ 為：

$$P_{qt-} = [x'-R/2, y'+R/2, x'+R/2, y'+R/2, x'+R/2, y'-R/2, x'-R/2, y'-R/2] \quad (5)$$

因此在結束查詢時間( $t = t+$ )時每秒x軸向的位移為 $V_{xt+}$ ，每秒y軸向的位移為 $V_{yt+}$ ，查詢區域為 $A''B''C''D''$ ，則：

$$O'' = (x'', y'') = (x + V_{xt+}, y + V_{yt+}) \quad (6)$$

而 $A''B''C''D''$ 的座標向量 $P_{qt+}$ 則為：

$$P_{qt+} = [x''-R/2, y''+R/2, x''+R/2, y''+R/2, x''+R/2, y''-R/2, x''-R/2, y''-R/2] \quad (7)$$

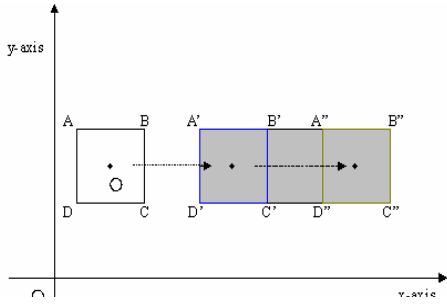
由上面的式子可以發現，速度向量 $V[V_x, V_y]$ 可以表示出物體移動的方向，而物體移動的範圍會影響查詢的區域，其影響的結果可以分為兩種狀況：

1. 當 $V_x$  和 $V_y$  有其中一項為零或兩項都為零時，代表物體是平行移動，其移動的區域範圍為一個四邊形。因此平移的物體，依查詢區域查詢符合需求的目標商店座標時，需滿足下列條件：

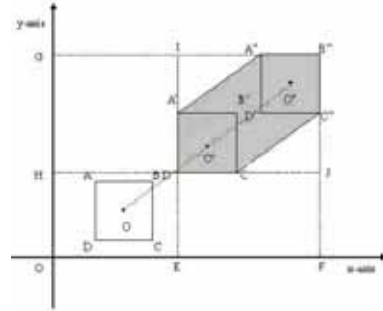
**Condition 1**：假設目標商店的座標為 $(S_x, S_y)$ ，則：

$$(1). S_x \in [x-R/2 + V_{xt-}, x+R/2 + V_{xt+}] = [x'-R/2, x''+R/2], \text{ 且}$$

$$(2). S_y \in [y-R/2 + V_{yt-}, y+R/2 + V_{yt+}] = [y'-R/2, y'+R/2].$$



圖五 物體平行移動查詢區域圖示



圖六 物體非平行移動查詢區域圖示

2. 當  $V_x \neq 0$  且  $V_y \neq 0$  時，物體不是平行移動，其移動的區域範圍為一個六邊形。將六邊形所圍成的灰色區域投射至  $x$  軸和  $y$  軸上，得到區間  $[E, F]$  和  $[H, G]$ ，但灰色區域才是真正的查詢區域。已知線段  $\overline{A'A''}$  與線段  $\overline{C'C''}$  平行，則  $\overline{A'A''}$  和  $\overline{C'C''}$  的斜率相同，將四邊形  $ID'JB''$  中灰色區域中的某一點代入和  $\overline{A'A''}$ 、 $\overline{C'C''}$  相同的斜率方程式後，其常數項會介於  $\overline{A'A''}$  和  $\overline{C'C''}$  兩個方程式的常數項之間，則：

$$\text{令 } \max(\overline{A'A''} \text{ 的常數項}, \overline{C'C''} \text{ 的常數項}) = C1$$

$$\min(\overline{A'A''} \text{ 的常數項}, \overline{C'C''} \text{ 的常數項}) = C2$$

因此，符合查詢需求的目標商店座標需滿足下列條件：

**Condition 2:** 假設目標商店的座標為  $(S_x, S_y)$ ，則：

(1).  $S_x \in [E, F] = [x - R/2 + V_{xt-}, x + R/2 + V_{xt+}]$ ，且

(2).  $S_y \in [G, H] = [y - R/2 + V_{yt-}, y + R/2 + V_{yt+}]$ ，而

(3).  $C2 \leq \frac{V_y}{V_x} S_x - S_y \leq C1$ 。

## 4.2 查詢區域為正方形框格

在此情境之下，使用者除了之前設定可以輸入的項目之外還可以自行設定查詢範圍的半徑；邊長  $R$  的正方形  $A'B'C'D'$  至邊長  $R'$  的正方形  $E'F'G'H'$  為使用者設定需要查詢的範圍，而  $E'F'G'H'$  的座標為：

$$[E'x, E'y, F'x, F'y, G'x, G'y, H'x, H'y] \\ = [x' - R'/2, y' + R'/2, x' + R'/2, y' + R'/2, x' + R'/2, y' - R'/2, x' - R'/2, y' - R'/2] \quad (8)$$

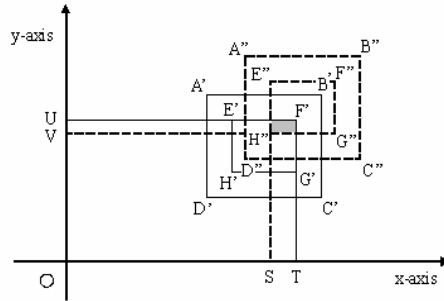
當物體移至  $O''$  時，邊長為  $R$  的正方形  $A''B''C''D''$  至邊長為  $R'$  的正方形  $E''F''G''H''$  為在查詢時間結束 ( $t+$ ) 時的查詢範圍，而  $E''F''G''H''$  的座標為：

$$[E''x, E''y, F''x, F''y, G''x, G''y, H''x, H''y] \\ = [x'' - R'/2, y'' + R'/2, x'' + R'/2, y'' + R'/2, x'' + R'/2, y'' - R'/2, x'' - R'/2, y'' - R'/2] \quad (9)$$

因此正方形  $A'B'C'D'$  移至  $A''B''C''D''$  的面積減去正方形  $E'F'G'H'$  至  $E''F''G''H''$  的面積即為使用者所查詢的範圍，但是在移動的過程中，可能會發生



不能被完全覆蓋的情況，如圖七所示，此時已不能再用六邊形  $A'A''B''D''D'C'$  做為查詢區域。



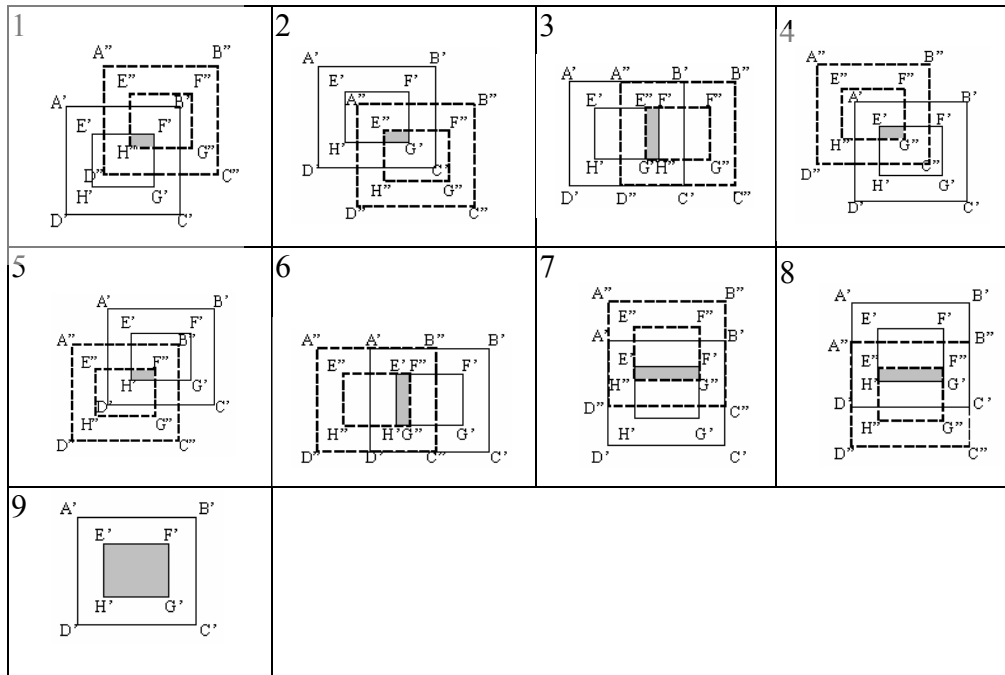
圖七 方形框格查詢區域產生無法完全覆蓋的情況

假設起始的實線正方形為  $A'B'C'D'$  及  $E'F'G'H'$ ，終點為虛線方格  $A''B''C''D''$  及  $E''F''G''H''$ ，速度向量  $V[V_x, V_y]$ ， $V_x$ 、 $V_y$  會依方向的不同而呈現正負或是為零。因此依照移動的方向可分為以表三之九種情況來討論：

表三 查詢區域無法被完全覆蓋的九種情境表

	移動方向	速度方向	條件	需扣除的區域
1	右上	$V_x > 0$ 且 $V_y > 0$	$F'x > H''x$ 且 $F'y > H''y$	$[H''x, F'x] \times [H''y, F'y]$
2	右下	$V_x > 0$ 且 $V_y < 0$	$G'x > E''x$ 且 $G'y < E''y$	$[E''x, G'x] \times [G'y, E''y]$
3	右	$V_x > 0$ 且 $V_y = 0$	$F'x > E''x$ 且 $F'y = E''y$	$[E''x, F'x] \times R'$
4	左上	$V_x < 0$ 且 $V_y > 0$	$E'x < G''x$ 且 $E'y > G''y$	$[E'x, G''x] \times [G''y, E'y]$
5	左下	$V_x < 0$ 且 $V_y < 0$	$H'x < F''x$ 且 $H'y < F''y$	$[H'x, F''x] \times [H'y, F''y]$
6	左	$V_x < 0$ 且 $V_y = 0$	$E'x < F''x$ 且 $E'y = F''y$	$[E'x, F''x] \times R'$
7	上	$V_x = 0$ 且 $V_y > 0$	$E'x = H''x$ 且 $E'y > H''y$	$[H''y, E'y] \times R'$
8	下	$V_x = 0$ 且 $V_y < 0$	$H'x = E''x$ 且 $H'y < E''y$	$[H'y, E''y] \times R'$
9	原地靜止	$V_x = 0$ 且 $V_y = 0$	無	四邊形 $E'F'G'H'$

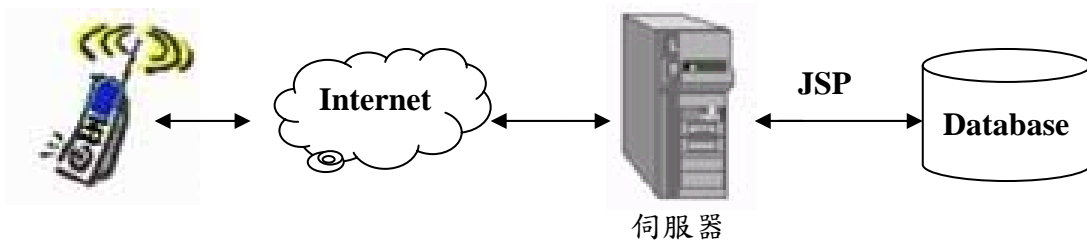
如果發生圖七中的灰色區域時，代表產生不完全覆蓋的查詢區域，所以在計算此類型的查詢區域時，必須先依照前面的限制條件 **Condition 2** 找出六邊形  $AEFGCD$  的區域，再扣除灰色區域的面積，即可得到最終正確的查詢區域。因此在依照使用者設定的查詢種類在資料庫中進行搜尋時，因  $R$  和時間間隔皆為使用者動態設定，此時若輸入查詢起始的範圍不為 0 時，則必須考慮是否為上述九種狀況之一，若不符合或是查詢起始的範圍為 0 時，則直接以 **Condition 1** 和 **Condition 2** 的條件判定至資料庫中尋找符合的資料，若為上述九種情況之一，則需將資料庫中符合灰色區域中的點扣除，才能真正找出符合查詢需求的目標。



圖八 九種無法完全覆蓋的查詢區域

## 5. 動態使用者查詢系統實作

由於目前的資訊科技非常發達，加上網路傳輸的能力無遠弗屆，為了達到本研究的目的並解決其研究問題，本研究使用 J2ME Wireless Toolkit JavaPhone 手機模擬器來開發移動資訊裝置的雛型系統，以 Tomcat 為網路伺服器，並用 JSP 網頁設計來聯結模擬器和資料庫，其架構基本上可分為用戶端、前端處理程序、網路伺服器、資料庫及後端處理程序等，如圖九所描述：



圖九 雛型系統架構圖

本系統的運作流程為，使用者在其所擁有的行動電話中選擇查詢的項目以及下達查詢的參數，需要設定的參數如前所述，包括查詢項目、查詢位置、查詢速度、查詢範圍及時間區間等等，之後動態使用者查詢系統會依據設定的參數計算

符合查詢的位置座標區間，並透過網路至資料庫中搜尋需要的資料後，再傳回給使用者參考，其使用者輸入介面如圖十所示。



圖十 使用者輸入介面

本系統的特色在於使用者可以依照目前的狀況和喜好設定查詢的參數，或是藉由定位系統的幫助，自動偵測所在位置及速度，讓回覆的查詢結果可以更為準確的提供給使用者，而本研究選擇目前發展快速且普及率廣泛的手機作為模擬的工具，藉由網路的傳輸，讓每個使用者可以在任何地區、任何時間查詢他們所想要的資料。

## 6. 結論

無線通訊產業及配合定位系統的行動商務應用將在未來民眾生活中佔有重要的一席之地，本研究提出的動態使用者查詢系統中，任何移動或靜止的物體皆可被視為使用者，並讓使用者在任何時間及任何地點進行查詢，查詢系統會考慮使用者目前的移動狀態，計算出更精確的查詢區域和查詢的結果，並將查詢結果回覆給使用者參考，讓查詢的方式更加地彈性化，讓這種查詢的應用系統實際呈現後再結合行動式服務，相信可以提供使用者更便利的服務。

因為本研究的應用的商業資料皆屬於靜態資料，因此未來延伸的研究方向還可以針對動態使用者查詢動態物體的目標繼續研究，若能準確的預測動態的適用者及動態物體，並將其服務實際應用在生活中，在交通或尋人等行動式服務上將有更大的發展，例如，發生緊急意外事故時，可以搜尋最近距離的醫護人員而不再只是查詢醫院的行動服務，將可讓行動查詢服務功能更加強大。

## 7. 參考文獻

1. Acharya, S., Poosala, V. and Ramaswamy, S., Selectivity Estimation in Spatial Databases, ACM SIGMOD Conference on Management of Data, 1999, pp. 13-24.

2. Aref, W. G. and Samet, H., Efficient Processing of Window Queries in the Pyramid Data Structure, The 9th ACM PODS Conference, 1990, pp. 265-272.
3. Chen, J., DeWitt, D. J., Tian, F., and Wang, Y., NiagaraCQ- a Scalable Continuous Query System for Internet Databases, ACM SIGMOD on Management of Data, 2000, pp. 379-390.
4. Choi, Y. and Chung, C., Selectivity Estimation for Spatio-Temporal Queries to Moving Objects, ACM SIGMOD Conference on Management of Data, 2002, pp. 440-451.
5. Klein, R., Nurmi, O., Ottmann, T. and Wood, T., Optimal Dynamic Solutions for Fixed Windowing Problems, The 2nd Annual Symposium on Computational Geometry, June 1986, pp. 109-115.
6. Kriegel, H.-P., Brinkhoff, T. and Schneider, R., Efficient Spatial Query Processing in Geographic Database Systems, IEEE Data Engineering Bulletin, Vol. 16, No. 3, 1993, pp. 10-15.
7. Sistla, P., Wolfson, O., Chamberlain, S. and Dao, S. Modeling and Querying Moving Objects, International Conference on Data Engineering (ICDE), April 1997, pp. 422-432.
8. Tao, Y. and Papadias, D., Spatial Queries in Dynamic Environments, ACM Transactions on Database Systems, Vol. 28, No. 2, June 2003, pp. 101-139.
9. Tao, Y., Sun, J. and Papadias, D., Selectivity Estimation for Predictive Spatio-Temporal Queries, International Conference on Data Engineering (ICDE), Mar. 2003, pp. 417-428.
10. Tao, Y., Sun, J. and Papadias, D., Analysis of Predictive Spatio-Temporal Queries, ACM Transactions on Database Systems, Volume 28, Issue 4, 2003, pp. 295-336.
11. Terry, D., Goldberg, D., Nichols, D. and Oki, B., Continuous Queries over Append-only Databases, ACM SIGMOD Conference on Management of Data, June 1992, pp. 321-330.
12. Wilde, M. D., Stroobandt, D. and Campenhout, J. V., AQUASUN: Adaptive Window Query Processing in CAD Applications for Physical Design and Verification. The 12<sup>th</sup> ACM Great Lakes Symposium on VLSI, 2002, pp. 153-159.
13. Zheng, B. and Lee, D., Semantic Caching in Location-dependent Query Processing, Symposium on Spatial and Temporal Databases, 2001, pp. 97-116.