Towards Derail of Global Pandemics via Patient Trajectory

Daniel Adu-Gyamfi[1,2,*] and Fengli Zhang[1]

[1] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
[2] Department of Computer Science and Informatics, University of Energy and Natural Resources, P O Box 214 Sunyani, Ghana
[*] Corresponding author. E-mail: daniel.adu-gyamfi@uenr.edu.gh

Received: Sep. 24, 2019, Accepted: Feb. 13, 2020

In recent decades, the public healthcare settings have devoted their attention to the derail of global pandemics. As a result, the public health professionals have adopted patients monitoring as one of the immediate measures to combat disease spreading. Incorrectness of data is a disadvantage of the traditional monitoring system as it is unable to efficiently handle the complex dynamic behavior of patients. The research community seeks to provide compelling techniques or algorithms that can be used to detect the location and travel of potentially hazardous and/or contagious patients in the case of pandemics. Heuristically, the mobility dynamics and activity records of patients are vital resources to support health researches. The activity records of patients contain their whereabouts in time, and that may provide some relevant knowledge for the analysis of disease spreading. This article presents a trajectory data mining technique to support the detection of outdoor mobile patient. The proposed technique examines the spatio-temporal trajectories of the patient via monitoring strategies with the aid of a global position system (GPS) device. The result of the experiment, using GeoLife big dataset has proven the proposed technique as efficient for monitoring an outdoor mobile patient, and it is easy to integrate into mobile health information systems that are intended for outdoor monitoring purposes towards derail of global pandemics.

Keywords: Health Information System; Mobile Data Analysis; Mobile Outdoor Patient; Patient Monitoring Technique; Spatio-Temporal Trajectory

http://dx.doi.org/10.6180/jase.202009_23(3).0005

1. Introduction

Mobile health information systems, such as patient monitoring system (PMS), etc. have become very important resources in the public healthcare settings. PMS is used to support the study of the physiological condition of mobile patients and their whereabouts in time. This is due to the support that it derives from the proliferation of ubiquitous mobile positioning systems and devices, for example Smartphones, Sensors, etc., that are used everywhere for collecting data on patients. In recent times, public health professionals have widely used PMS to collect vital information on patients, which has also aided them to report several health cases with ease and as early as possible. For example, a case like pandemics. Pandemics can drastically reduce the productivity of highly skilled workers, and correspondingly result to socio-economic disruption. Moreover, it is quite easier to report pandemic cases with the aid of PMS. Particularly, a report on an asymptomatic patient who is diagnosed with a contagious disease, but has no noticeable symptoms. This patient can easily spread the disease without adequate monitoring. Nonetheless, obtaining information with a traditional patient monitoring system can result to data incorrectness. This is due to the complex dynamic activity and behavior of patients. This situation can result to some adverse effects on producing the monitoring results efficiently. Nonetheless, it is a very challenging task to tag a patient with a particular interest in dynamic conditions, and thus to do so incrementally in continuous time intervals. This will require excessive computations, due to the high dimensional data type it involves [1]. Also, a lot of software and hardware resources are required [2, 3] to carry out the task. Thereby, it has become imperative for researchers to develop techniques and/or algorithms that can be appropriately applied to capture the complex dynamic activity and behavior of patients so as to enhance the features of mobile health information systems. In other words, researchers seek to produce some compelling techniques and/or algorithms that can be used to detect the location and travel of potentially hazardous and/or contagious patients in the case of pandemics. Moreover, the daily activity records [4–6] of patients are very resourceful data which can aid the public health professionals to obtain relevant information on patients [7, 8]. Such data, upon deeper analysis can provide some useful knowledge to handle pandemic cases and their chances of reoccurring. Let's consider asymptomatic patients as an example. In spite of medically diagnosing such patients with certain contagious chronic diseases, it is also not easy to physically identify them. This is due to the no symptoms condition, and hence their health status cannot be revealed. Therefore, it has become imperative that the public healthcare settings may have to devise a mechanism to put such categories of patients under special monitoring, and thus right after they are diagnosed with any potentially hazardous and/or contagious diseases. Such mechanism will aid the public health professionals to derail any possible global pandemics. The introduction of information system technologies in the healthcare settings has contributed a lot to patient monitoring towards a fight against disease spreading. Nowadays, patients can wear clothes that are equipped with sensors to record their health data [8]. Pawar et al. [9] have proposed a framework to compare mobile patient monitoring systems. Further, the authors defined mobile patient monitoring as the continuous (or periodic) measurement and analysis of a mobile patient biosignals from a distance by employing mobile computing, wireless communications, and networking technologies. Or et al. [9] have proposed that modern health information systems still face challenges with efficiency due to their under-explored design techniques. Moreover, Singh et al. [10] have introduced a deep learning model for recognizing the activities of people. The model has been used to learn to classify the activities of people without any prior knowledge. Heuristically, the model will not well fit for investigating the behavior dynamics of outdoor mobile patient in continuous instances. This is due to the fact that, in practice there is the need for a prior knowledge about the health status of the patients before monitoring can proceed. In respect of monitoring the mobility of objects, Wu et al. [11] identified two pressing issues. These include, how to correlate spatial and temporal information on objects in order to obtain accurate location predictions, and also how to effectively develop and deploy large scale predictive systems. In this article, it is envisaged that obtaining a prior knowledge on the health status of patients prior to monitoring is an ideal condition to consider. This present study is motivated by the proliferation of ubiquitous global position system (GPS) mobile devices, such as mobile phones, sensors, etc. and the vast available methods of computing spatial positions of objects. Hence, this article has utilized the spatial locations, positions, time stamp, and place of visit of an outdoor mobile patient to introduce monitoring technique that is based on the concept of trajectory data mining. The focus of this present study is to advance the design of mobile health information systems that are aimed at monitoring outdoor mobile patients towards derail of global pandemics. Similar algorithmic designs [2, 12–14] and methods which are incorporated with a grid based index data structure have been adopted in this present study. These include, concurrent object localization and recognition techniques, branch and bound formalism, and multi-object instance. The ultimate goal of adopting these fore going methods is to considerably determine the current time-stamp position and location of outdoor mobile patient during monitoring. This present study has been experimented using the publicly available GeoLife big dataset [15]. The result has proven the proposed technique as efficient, and easy to integrate into state-of-the-art PMS. Nonetheless, the key setback of this study is authorization to access the mobile device data. This is because, access to mobile data may vary from one country to another, and thus depending on the corresponding laws of every country. Therefore, the study recommends users and/or commercial projects to devise the necessary access control policy and/or consent as a mechanism to directly access the mobile phone data of the patients.

2. Theory and formula

Trajectory data mining is an act of exploring and discovering knowledge from the trajectories that are generated by objects. Here, we define trajectory as a tuple given by the sequence $P = (x_1, y_1, tax_1, ts_1), ..., (x_k, y_k, tax_k, ts_k)$ [11–13]. Where x and y denote any location coordinates. The ta and ts represent the respective arrival time and stay time of the objects. Such that $ta_1 <, ..., < ta_k$ and $ts_1 <, ..., < ts_k$, where k represents the size or length of data. Essentially, P which is the trajectory of the objects is subjected to a unit positive weight, as well as other attributes or environmental features such as altitude [13]. However, it is considerably insignificant to consider the altitude in trajectory analysis. Moreover, in every location and time, the mobile objects usually exhibit certain basic attributes during mobility, such as place of interest and duration of travel, etc. and these are meaningful in trajectory analysis. These basic attributes are classified as spatio-temporal (spatial and temporal) properties. Therefore, a trajectory which involves these basic attributes of the objects is termed as spatio-temporal trajectory. Spatio-temporal trajectory can be simplified as the activity that is spatially time-bounded which is carried out by a mobile object. Let $P(n)$ be denoted as any countably-infinite position data of a patient (or object). Let i be denoted as any identified position data of a patient. It can be deduced that $P(i) = (x, y, w)$, such that $P(i) * w \epsilon R^+$, where $P(i) \epsilon P(n)$. Heuristically, we can establish Eq. 1 as sum of the identified position data of a patient. Such that w is represented as the positive weight which is tagged to the monitoring location.

$$P(i)w = \sum P(i) * w \tag{1}$$

Supposing in every location, we want to find the $top - k$ place of visit of a patient. The location can be seen as smaller search space r and large space R . Such that it can represent the relation $r < R$. Thereby, if $r = R$ then it is not relevant to execute any task, since the search space should not be the same as the large space (i.e. which represents the entire monitoring location). Also, if $r > R$, then the condition is practically not possible to execute, since the search space must be within the large space. Heuristically, we can establish in 2, the maximized weighted-sum of any identified position of the patient.

$$r(w) = \underset{i \epsilon n \cap R}{argmax} P(i) * W \tag{2}$$

Further, mathematically we can deduce how to find the location and position of a patient, and show how to monitor such a patient accordingly. We approach the problem with the concept of concurrent localization and recognition of objects [2]. The concept seeks to device a model to find a bounding rectangle (r) in the case of localization, in which r has to enclose some of the identified position data $P(i)$. Also, in the case of recognition the same model has to maximize the prediction score (L) for the $P(n)$. Let n be a number of location features, and further consider that as any input given by the triplets, $F =: (ax_k, by_k, cL_k)|k : 1, ..., n$. Such that L becomes an M-dimensional vector. The a, b, and c represent any labels of the location features which are constants. If $L_{(i)}$ is the ith element of $L_{(k)}$, then $i < k$. Therefore, we can heuristically establish the objective function in Eq. 3 to obtain the prediction score for any position data of the patient by considering the $top - k$ visited places, and thus within the monitoring or search location.

$$r(w) = \underset{i, r \epsilon R}{argmax P} \sum_{k \epsilon F(r)} L_{\vec{(k)}}(i) \tag{3}$$

Moreover, classification can be used to identify the $top - k$ positions of the patient, that will serve as the influential place among the rest of the places of visit or travel. In this case, the position data can be taken as input into the SVM classifier, and a testing sample is compared with that of a training sample. The weighted position data which is considered as the support vector can be compared with the aggregated partial score of the entire position data. Thereafter, any similarity score can be obtained. Usually, the similar scores can be selected. Hence, summing up the similar score based on some SVM offset value τ, will lead to a chosen margin. Based on the margin, the aggregated score can be expressed as the weighted-sum of partial similarity scores obtained from the $top - k$ position data. Therefore, we summarize the whole process and establish a formula in Eq. 4. Where $\mu_{\vec{(p)}}$ is a vector which carries a support vector weight of the P training set. In this case, if there exists any ith element in the P training set, then this will denote the weight of the ith SVM.

$$L_{\vec{(k)}}(i) = \sum_{i \epsilon k} h(l_p = i) c_p \mu_{\vec{(p)}}(i) \tag{4}$$

In addition, we can employ branch and bound strategy to break down the search space into disjoined subspaces. This is to obtain an optimal value to represent the $top - k$ position data. Subsequently, by applying the branch technique, we let d denotes a subspace, d_1 and d_2 to denote any two smaller components of the subspace. Let's represent m as the object model of R , and z_1 and z_2 as any two subsets of R. By so doing, we can transform d as $d(i) =: (-x_{(i)}, +x_{(i)}, -y_{(i)}, +y_{(i)}), m_{(i)}$. Where $-x$ and $+x$ represent the extreme x-coordinates of R. Such that $i = 1, 2$ is a binary. Consider the y-coordinate of R, we split that into d_1 in which the points in z_1 are removed from the $-y$, including all the points in the x_1. On the other hand, we can obtain $+b$ and $-b$ as the largest and smallest rectangle to represent the monitoring location by applying the bound technique. Thus, by using the subspace d and extremities of the x-coordinates, y-coordinates and m. Heuristically, if a which represents any label of the location feature satisfies that $a \epsilon m$, then in Eq. 5 and 6, we use a and consider the $+b$ and $-b$ to locate a search space of the patient. Thus, the search space must contain more positive weighted position data, such that $z_{\vec{(k)}}(a) > 0$. By so doing, we try to exclude many of the negative values, such that $z_{\vec{(k)}}(a) < 0$. In addition, let $+j$ denotes an upper-bound as a function of subspace d which is computed via Eq. 5. On the other hand, let $-j$ denotes a lower-bound as a function of subspace d which is computed via Eq. 6. In view of that $+h(x)$ and $-h(x)$ are known functions. Such that $+h(x)$ is evaluated whenever the (x) is positive, and $-h(x)$ is evaluated where negative. Otherwise, the function is evaluated to null (or zero).

$$+j(d) = \sum_{j\epsilon d(+b)} h^{+}(z_{(k)}^{\rightarrow}(a)) + \sum_{j\epsilon d(-b)} h^{-}(z_{(k)}^{\rightarrow} \quad (5)$$

$$-j(d) = \sum_{j\epsilon d(+b)} h^{+}(z_{(k)}^{\rightarrow}(a)) + \sum_{j\epsilon d(-b)} h^{-}(z_{(k)}^{\rightarrow} \quad (6)$$

$$+j = j + \tau \quad (7)$$

$$-j = j + \tau \quad (8)$$

The study of network topology coupled with graph theory are used in various fields to find solution to complex dynamic network problems [16]. These include the analyses of the network connectivity behavior, patterns and impacts of network systems, continuous spatio-temporal trajectory joins and among others [16]. Subsequently, in this present study, a generalised grid based index data structure is developed to process the trajectory of a patient by using concepts of graph theory and networking. This is to prove that the proposed grid based index data structure is very efficient to handle mobility dynamics of the patient in the case of continuous instances. Consider the Fig. 1, where there exist ten different position instances of an outdoor mobile patient. These are enclosed within rectangles which are labeled as A, B, C, D, E, F, G, H. The red broken-line rectangle D represents the optimal position instance (i.e. it has a total of three position data) which represents the maximized weighted-sum of the position data. Assume that $G = (V, E)$ is a dynamic graph. The vertices V will correspond to any position data, and it is transcribed as rectangle as shown in Fig. 1. Therefore, when any two vertices overlap, we obtain an edge E. In other words, a one-to-one mapping of vertices will generate an edge E. Subsequently, by using the concept of graph analysis, a dynamic tree structure is constructed as shown in Fig. 2, and that is based on the dynamic graph G. Further, by using the dynamic tree structure, we derive a grid based index data structure as shown in Fig. 3. Let r_i and r_j be any individual nodes. Such that $r_i, R_j \epsilon V$, in which r_i is generated first (or later), followed by r_j. It follows that $r_i \neq r_j$ in the case of vertices. Also, in the case of nodes or edges, $e_i \neq e_j$. We provide the relationship of the dynamic graph G in Table 1. We make use of origin-destination matrix at this point, and further construct a list adjacency matrix, and that is based on the Fig. 2. In the Fig. 2, any $p_{(i,j)}$ corresponds to the position data. Furthermore, the grid based index data structure as shown in Fig. 3 is based on an NXN matrix, which is to take care of the activity and/or behavior dynamics of patient in continuous instances of mobility. When developing the grid based index data structure, the $AdjMatrix[i][j]$ and $AdjMatrix[j][i]$ need to be set to either 0 for overlap or 1 for non-overlap in the case of each pair of position data. Assume that M is any newly generated position data, and N is the rate of generating the position data. We obtain the time complexity for implementing the ordinary data structure as $O(MN)$. This is not quite efficient, hence we apply network dynamics to further modify the graph $G = (V, E)$ to a continuous dynamic graph $G_{(i,j)} = (V_{(i,j)}, E_{(i,j)}$. This is to enable the Fig. 2 to cater for continuous position instances of the patient. Thereby, we can obtain cell $g_{(i)}$ and/or $g_{(j)}$ as a subset of the graph $G_{(i,j)}$, and that is classified as a continuous dynamic grid structure or cell. As a result, if m is considered as newly generated vertices of $G_{(i,j)}$, then the cells $g_{(i)}$ and $g_{(j)}$ can be checked where there is any overlaps. This is to cover any update cell $G^u(i)$ observed in the grid index data structure. Where i represents a single factor of the update. In view of that, the overall time complexity when the data structure is optimized will be $O(|G^u(i)|M^u, N^u)$. The M^u is the average number of vertices that are updated after the overlap, and N^u is the average number of updated overlap. Such that $M^u << M$, and $N^u << N$. At this stage, the time complexity is theoretically reasonable and efficient for practical implementation. Moreover, the cost of storage is $O(|V| + |E|)$ by considering all the cells in $G_{(i,j)} = (V_{(i,j)}, E_{(i,j)}$, and for every $r_{(i)}$ (i.e. those vertices that are maintained as position data and not discarded). The worst space complexity of storage is $O(|V|^2)$, by considering all the n rounds of overlap which is equivalent to that of $O(|O|^2)$ [13]. However, it is not practical to evaluate such a worst space [13]. At this point, it is proven that the proposed grid based index data structure is very efficient, and can handle the complex dynamics (i.e. behavior/activity) of the patient in the case of continuous instances of mobility. The various trajectory records [17, 18] of the patient can be incorporated into the grid based index data structure for implementation as provided in the proposed algorithm. In addition, it is simple to execute the patient monitoring in real time with the algorithm. This is to aid to identify the place of visit or travel together with the various activities and/or behavior dynamics that are exhibited by the outdoor mobile patient. Notwithstanding, the maximized weighted-sum of position data can be deduced from the $top - k$ visited locations. This can further be used to compute any future place of visit [19] of the patient in continuous time intervals. Subsequently, we elaborate on how the proposed algorithm will implement this phenomenon. Assume that $l(w)$ represents any newly identified place of visit, and $p(i)$ is any identified position data. Supposing there exists a cell $g_{(i,j)} \epsilon G$. If $g_{(i,j)} * w > l(w)$, then $l(w)$ will not be part of the vertices $V_{(i,j)}$ (i.e.). Hence

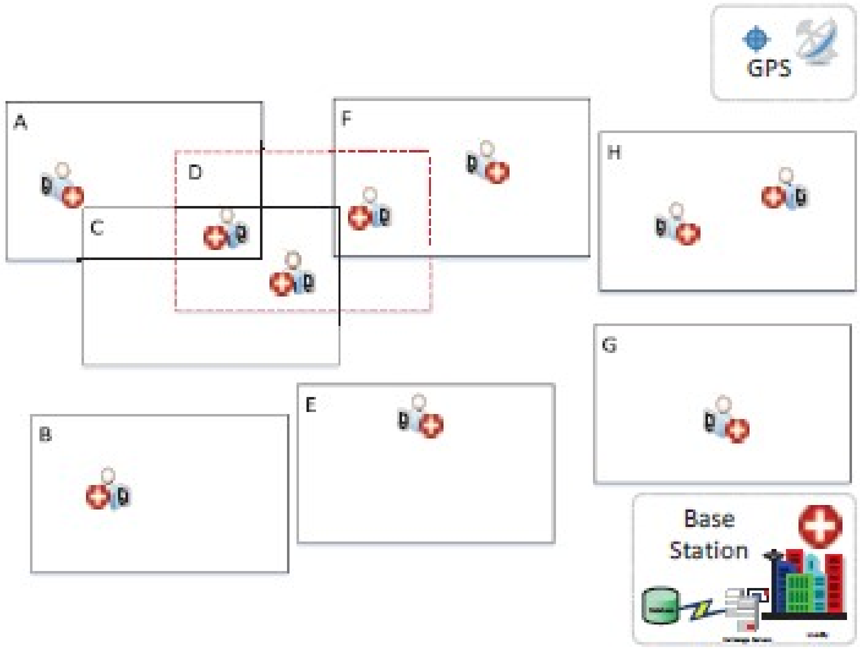

Fig. 1. A schematic view of tracking the position instances of a patient. Assume that a patient is traveling with a GPS enabled mobile device. The rectangles with labels A, B, C, D, E, F, G, H are used to capture the respective position instances of an outdoor mobile patient over time. The rectangle D with red broken-line denotes the optimal solution. Thus, among all the eight rectangles the D contains a total of three position instances of the patient representing the maximized weighted-sum of the position data. The "Base Station" is the medical centre where the GPS data are processed by the data analytic experts.

there is no need to compute the exact solution (i.e. exact place of visit). This condition also implies that, if $g_{(i,j)} \epsilon G$, then $g_{(i,j)} * w \geq (w)$. Furthermore, consider $r_{(i)} \epsilon V_{(i,j)}$ as a new node. We can establish that $l(w) < P(i)$. This implies that $P(i) \exists r_{(i)}$. Therefore, in order to find the newest $top - k$ visited place as per the positions of the patient, one can consider the iteration from Eq. 9 through to Eq. 10. The optimization function is established in Eq. 11. Intuitively, to summarize the procedure as a whole and cater for the implementation, we present that in *algorithm 1*, *algorithm 2*, and *algorithm 3*.

$$l(w) = l(w) + \sum_{i}^{k} p(i) \tag{9}$$

$$l(w) = l(w) + \sum_{i}^{k} l(i) \tag{10}$$

$$g = \underset{g_{i,j} \epsilon G \cap R}{argmax g_{i,j} w} \tag{11}$$

3. Experimental setup

We have experimented and evaluated the proposed algorithm using python 2.7 platform on personal computer (PC). The specifications of the PC include Intel Core i5, CPU 4.1 GHz, 8.00 GB RAM, and Windows 10 OS. The basic scientific software and packages that were used include Scikit library, Numpy, and Microsoft Visio. These tools were mounted with the GeoLife dataset [15]. GeoLife dataset is published by the Microsoft Asia Beijing Research Centre [15]. The GeoLife dataset is for research purposes, and it is publicly available for free download [15]. The GeoLife dataset is recorded with GPS mobile devices, and it is a type of trajectory data which consists of social networking records of outdoor mobile users [15]. In our experiment, the trajectory data were divided into training and testing samples. The dataset was executed using our technique

Table 1. Relationship of dynamic graph G.

Vertex $(V_{(ri)})$	Edge $(e_{(i)})$	Next-neighbor Vertex $(NV_{(ri)})$
1	e1,e2	2.3
2	e3	3
3	e4	4
4	e6	6
6	e7	7
7	null	null

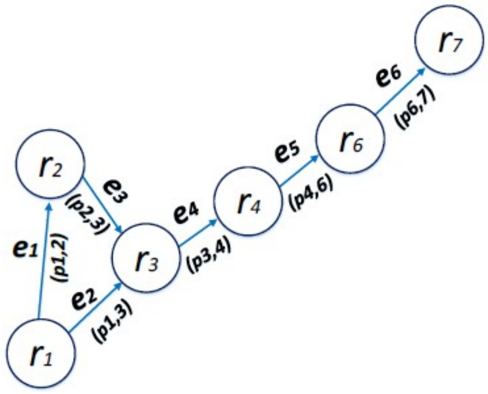

Fig. 2. A tree structure of dynamic graph G. The r_i denotes any vertices which is transformed as the continuous position instances of the patient. Where r_1 is generated later than r_2 through to r_7 as the current position instance. The e_i denotes an edge which is an overlap of any two vertices. This is further transformed as $(p_{i,j})$ to represent the various corresponding position data of the patient.

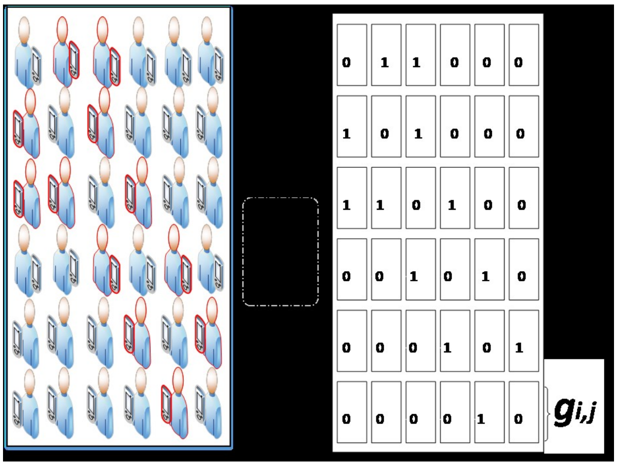

Fig. 3. A grid based index data structure of dynamic graph G. Based on the concept of origin-destination matrix, the $AdjMatrix[i][j]$ and $AdjMatrix[j][i]$ are set to either 0 in the case of overlap or 1 in the case of non-overlap, for each pair of position data (or vertices). Where $g_{i,j}$ represents a cell or any generated position instance for a continuous mobility.

and the G2 method [13]. The G2 method is a streaming data technique which seeks to similarly monitor mobile objects. Our proposed technique is represented as TMAXWS in the legend, whereas G2 method is represented as G2. The location used for the monitoring was represented as a rectangle, and this was set to 1000 x 1000 meter squared by default. The other parameters for the experiment are provided in Table 2. Thus, by default the size of position data was set to 500 in number, the rate of generating position data was set at 100 seconds, the user tolerance was varied from 0 to 0.4 marginal error, and the $top - k$ query answer was set to 10 minutes. The various results of the experiment are shown in Fig. 4.

In this present study, access to mobile data may vary from one country to another, and thus depending on the corresponding laws of every country. Hence, the authorization to access the mobile device data is a major challenge. This is especially, in the case of accessing private data of the patients which will require their consent.

4. Result discussions

The average computation time of each parameter in Table 2 was considered. Comparatively, the experiment verified that our technique has performed very well in line with the G2 method [13]. This is with regards to outdoor mobile object monitoring. Whereby the proposed technique seeks to detect the location and travel of potentially hazardous and/or contagious patients in the case of a pandemic. This is evidenced in the log scale of the graphs provided in Fig. 4, in which the proposed technique is represented as TMAXWS in the legend, whereas G2 method is represented as G2. The impact of k which is the size of position data of a patient was tested. The computation time for finding each position data was increased linearly as shown in Fig. 4 (a). There were several overlaps, and the situation overburdened the functions given as ComputeOverlap $(\cdot)$ and ComputeExact. Weight $(\cdot)$ that are used to compute the overlaps and the exact weight of the position data, respectively. This means that when there is a larger location size, there will be more time to produce results on the visited place of the patient. The impact of d as rectangle size which represents visited location was tested, as shown in Fig. 4 (b). Here a skewed distribution was observed, in which the rectangle size also increased quite sharply as more overlaps occur. This situation suggest that it is reasonable to monitor the outdoor mobile patient in a sizeable location, and thus based on the available resources during implementation. The impact of which represents the rate of generating position data was tested. Here, less than 50 positions were generated per second, as shown in Fig. 4 (c). The rate of generating the position data was increased, and that directly improved the execution time. This behavior suggests the technique and/or algorithm as relatively faster in terms of response time of querying any position instance of the patient. The impact of tol which is defined as user tolerance was tested. This was to obtain the ap-

Algorithm 1 Algorithm for position data extraction and execution

function $Loc.Records\ (P)$
$-x_0, +x_0, -y_0, +y_0 \longleftarrow$ the points in P // Obtaining SubSpace
Sort out xs' and ys' along coordinates x and y
$a_0 \longleftarrow$ labels of objects models
$d_0 \longleftarrow -x_0, +x_0, -y_0, +y_0, a_0$
$L \longleftarrow$ a null priority queue initialize
$L \longleftarrow$ set d_0
while $V \longleftarrow POP(L)$ **do**
    if $\forall Z \in V, |Z| = 1$ **then**
        return V
    end if
    $(d_1, d_2) \longleftarrow SPLIT(d)$
    push d_1 to J with key $L_m(d_1)$
    push d_2 to J with key $L_m(d_2)$
end while
$(-x, +x) \bigcup (-y, +y) \bigcup a \longleftarrow$ the points in d //Obtaining UpperBound
Let Z be largest set of candidate
if $Z = m$ **then**
    Let $d_m \longleftarrow ((-x, +x) \bigcup (-y, +y) \bigcup m)\ w$
    $wSort$ all $m \in Z$ by $L_m\ (d_m)$
    $Z_1 \longleftarrow$ push all $m_{(i)}$ top half
    $Z_2 \longleftarrow$ push all $m_{(i)}$ bottom half
else
    $Z_1 \longleftarrow$ map first half of all Z
    $Z_2 \longleftarrow$ map second half of all Z
end if
$d_1 \longleftarrow$ retrieve d copy with Z_1 removed
$d_2 \longleftarrow$ retrieve d copy with Z_2 removed
return (d_1, d_2)
end function

Table 2. Experimental parameters.

Param	Default	Variable	Indicator
n	500	100; 250; 500; 750; 1000	Size of position data
d	1000	100; 500;1000; 1500; 2000	Sizes of rectangle
m	100	50; 100; 200; 500; 800	Rate of generating data
Tol	-	0; 0.10; 0.20; 0.30; 0.40	User tolerance
$Top - k$	-	10; 20; 30; 40; 50	Query answer

Algorithm 2 Algorithm for monitoring evolving position data

1: Input R^u // R^u newly generated position data
2: Map (R^u)
3: $G^u(i) \longleftarrow$ update cell $g_{i,j}$ at m-rectangle overlap
4: $G_{i,j} \longleftarrow$ update $(G^u(i))$ // Computing overlap for rectangles
5: **for** $\forall\ g_{i,j} \in G^u(i)$ **do**
6: **for** $\forall\ V_{(ri)} \in g_{i,j} \ldots V_{i,j}$ when new $e_i \longleftarrow r$ **do**
7: $p_{(i)} \longleftarrow$ Loc.L-Sweep $(V(NV_{(ri)} \cup V_{(ri)}$ // Table 1 in-memory tree update, O(n) space, exec.(t) of O(n log n)
8: $l(w) \longleftarrow l(w) = \arg\max_{(i\in n)\cap R}\ p(i)w$
9: **end for**
10: **end for**
11: **return** $l(w)$

proximation of quality results when querying any place of visit. The *tol* was varied as per the executed position data of the patient. The running time was seen as indirectly proportional to the *tol* as shown in Fig. 4 (d). This behavior of the graph suggests a trade-off between the quality of result and the efficiency of processing the query to obtain the place of visit of the patient. The Fig. 4 (e) provides the results of the $top-k$ place of visit of the patient based on the position data. When the $top-k$ data was increased, there was a proportional increased in the execution time of the algorithm. This behavior of the graph suggests that $top-k$ visited place has a direct influence on the execution time of the algorithm (i.e. when querying for the future positions of the patient). Therefore, *tol* must always be set as a marginal error rate. Therefore, it is not necessary to obtain the exact location of the patient in the case of practical applications. For instance, it will be enough to implement the appropriate monitoring measures by the public health settings upon knowing the $top-k$ places (i.e. influential places) that were visited by the patient. This will facilitate early detection of hazardous and/or contagious patients in the case of a pandemic.

5. Conclusions

In this paper, we have designed and evaluated a trajectory data mining based technique for the purpose of monitoring outdoor mobile patients in the case of pandemics. The technique has made use of grid-based index data structure, and that is based on network topology, graph theory and trajectory analysis. The technique also adopted the current object localisation and recognition, branch and bound formalism, and multi-object instances strategies to arrive at the solution. An outdoor mobile patient was emulated as ordinary GPS outdoor mobile phone user based on the well-defined GeoLife trajectory dataset. The technique has correlated data on the activity and behavior dynamics of the patient via spatial and temporal information during her mobility. The proposed technique has not only proven efficient, and that is based on execution time of producing query results, but it is also easy to deploy into practical applications. The proposed technique and/or algorithm can offer considerable enhancement on the existing public health information systems towards monitoring outdoor mobile patients in order to derail global pandemics.

The limitation of this present study is the authorization to access the mobile device data of the patients which will require their consent, and also access to mobile data may vary from one country to another depending on the corresponding laws of the country. Our future work will include a comparative study of the proposed technique and/or algorithm with that of other state-of-the-art outdoor patient monitoring techniques in the case of pandemics.

Algorithm 3 Training algorithm for discovery of continuous influential location

Require: Execute *Algorithms* 1 & Steps: 1−8 in *Algorithm* 2

1: **for** $\forall\, r \in R^{u}$ **do**
2:   **if** $g_{i,j} \longleftarrow r$ **then**
3:     $g_{i,j}\dot{w} \longleftarrow g_{i,j}\dot{w} + r\dot{w}$
4:     $R_{(i,j)} \longleftarrow R_{(i,j)} \cup r$
5:   **end if**
6: **end for**
7: $g \longleftarrow g_{i,j} | l \in g_{i,j}$
8: **if** l obsolete $(g \longleftarrow null)$ **then**
9:   $g \longleftarrow \arg\max_{g_{i,j} \in G \cap R} p(i)w$
10: **end if**
11: ComputeOverlap (g)
12: $l \longleftarrow$ ComputeExact.Weight $(l\ ,\ g)$
13: **for** $\forall\, g_{i,j} \in G \setminus g$ **do**
14:   **if** $g_{i,j}\dot{w} > l(w)$ **then**
15:     ComputeOverlap$(g_{i,j})$
16:   **end if**
17: **end for**
18: **if** $g_{i,j}\dot{w} > l(w)$ **then**
19:   $l(w) \longleftarrow$ ComputeExact.Weight $(l, g_{i,j})$
20: **end if**
21: Compute $top-k$ Weighted-sum $(l(w))$ // set $l(w^{)}$ of the k spaces maximum weighted-sum
22: Execute lines $1 \rightarrow 6$
23: $G^{u}(i) \longleftarrow\ \forall\, g_{i,j} \mid \exists\, p(i)w \in l(w) \in g_{i,j}$
24: **if** $G^{u}(i) = null$ (obsolete of all spaces in $l(w)$) **then**
25:   $G^{u}(i)$ $\arg\max_{g_{i,j} \in G \cap R} G^{u}(i)$
26: **end if**
27: ComputeOverlap$(G^{u}(i))$
28: $l(w) \longleftarrow$ ComputeExact.Weight$(l(w)\ ;\ G^{u}(i))$
29: **for** $\forall\, g_{i,j} \in G \setminus G^{u}(i)$ **do**
30:   Execute lines $14 \rightarrow 19$ 21
31: **end for**
32: **return** $l(w)$

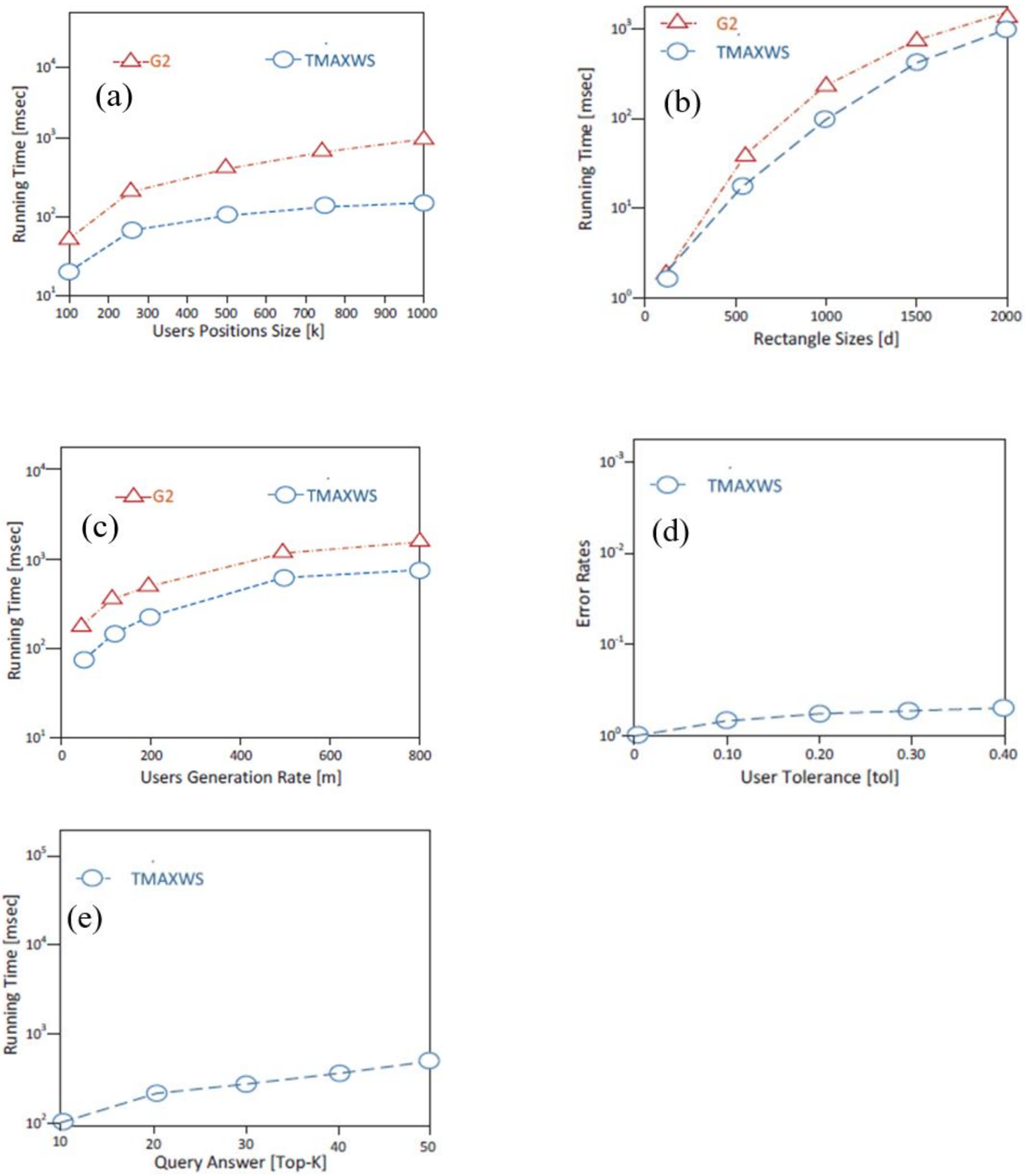

Fig. 4. Graphs showing the results of experiments. In (a) the computation time for finding each position data has increased linearly with increasing position data. This suggests that larger location size will require more time to produce results on the visited place of the patient. In (b) the distribution has been skewed. The rectangle size has increased quite sharply due to more overlaps occurring. This suggests that the outdoor mobile patient should be monitored in a location that has reasonable size. In (c) the rate of generating the position data has increased. There has been a corresponding increase in the execution time. This suggests that the response time of the algorithm on querying any position instance of the patient was relatively faster. In (d) the running time has decreased and that is indirectly proportional to tol. This suggests that obtaining the quality of result and the efficiency of query processing of the place of visit of the patient will be a trade-off. In (e) the $top - k$ position data has increased proportionally with the execution time. This suggests that querying for the exact location of the patient will not be necessary in the case of practical implementation.

Acknowledgment

This research was supported by the National Natural Science Foundation of China [Grant No.61602097, No.61502087, and No.61472064]

References

[1] Sangkyun Lee and Andreas Holzinger. Knowledge discovery from complex high dimensional data. In *Solving Large Scale Learning Tasks. Challenges and Algorithms*, pages 148–167. Springer, 2016.

[2] Tom Yeh, John J Lee, and Trevor Darrell. Fast concurrent object localization and recognition. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 280–287. IEEE, 2009.

[3] Francesco Lettich, Salvatore Orlando, and Claudio Silvestri. Processing streams of spatial k-NN queries and position updates on manycore GPUs. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 1–10, 2015.

[4] Mohammad Mehedi Hassan, Shamsul Huda, Md Zia Uddin, Ahmad Almogren, and Majed Alrubaian. Human activity recognition from body sensor data using deep learning. *Journal of medical systems*, 42(6):99, 2018.

[5] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *nature*, 453(7196):779–782, 2008.

[6] Thomas Beltrame, Robert Amelard, Alexander Wong, and Richard L Hughson. Extracting aerobic system dynamics during unsupervised activities of daily living using wearable sensor machine learning models. *Journal of Applied Physiology*, 124(2):473–481, 2018.

[7] Pedro Nogueira, Joana Urbano, Luís Paulo Reis, Henrique Lopes Cardoso, Daniel Castro Silva, Ana Paula Rocha, Joaquim Gonçalves, and Brígida Mónica Faria. A Review of Commercial and Medical-Grade Physiological Monitoring Devices for Biofeedback-Assisted Quality of Life Improvement Studies. *Journal of medical systems*, 42(6):101, 2018.

[8] Massimo Esposito, Aniello Minutolo, Rosario Megna, Manolo Forastiere, Mario Magliulo, and Giuseppe De Pietro. A smart mobile, self-configuring, context-aware architecture for personal health monitoring. *Engineering Applications of Artificial Intelligence*, 67:136–156, 2018.

[9] Calvin Or, Ellen Tong, Joseph Tan, and Summer Chan. Exploring factors affecting voluntary adoption of electronic medical records among physicians and clinical assistants of small or solo private general practice clinics. *Journal of medical systems*, 42(7):121, 2018.

[10] Deepika Singh, Erinc Merdivan, Ismini Psychoula, Johannes Kropf, Sten Hanke, Matthieu Geist, and Andreas Holzinger. Human activity recognition using recurrent neural networks. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 267–274. Springer, 2017.

[11] Ruizhi Wu, Guangchun Luo, Junming Shao, Ling Tian, and Chengzong Peng. Location prediction on trajectory data: A review. *Big data mining and analytics*, 1(2):108–127, 2018.

[12] Yu Zheng. Trajectory data mining: an overview. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3):1–41, 2015.

[13] Daichi Amagata and Takahiro Hara. A general framework for MaxRS and MaxCRS monitoring in spatial data streams. *ACM Transactions on Spatial Algorithms and Systems (TSAS)*, 3(1):1–34, 2017.

[14] Daniel Adu-Gyamfi, Fan Zhou, Fengli Zhang, Kittur Philemon Kibiwott, and Victor Dela Tattrah. Realtime Monitoring of Mobile User using Trajectory Data Mining. In *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pages 1–8. IEEE, 2019.

[15] Yu Zheng, Xing Xie, and Wei-Ying Ma. GeoLife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.*, 33(2):32–39, 2010.

[16] Petko Bakalov and Vassilis J Tsotras. Continuous spatiotemporal trajectory joins. In *International conference on GeoSensor Networks*, pages 109–128. Springer, 2006.

[17] Fa Li, Xi Long, Shenglan Du, Jiawen Zhang, Zizheng Liu, Moying Li, Feng Li, Zhipeng Gui, and Hanruo Yu. Analyzing campus mobility patterns of college students by using GPS trajectory data and graph-based approach. In *2015 23rd International Conference on Geoinformatics*, pages 1–5. IEEE, 2015.

[18] Negin Ebadi, Jee Eun Kang, and Samiul Hasan. Constructing activity–mobility trajectories of college students based on smart card transaction data. *International Journal of Transportation Science and Technology*, 6(4):316–329, 2017.

[19] Eunjoon Cho, Seth A Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090, 2011.