Journal of Information Technology Research
Volume 15 • Issue 1

Mobility and Trajectory-Based Technique for Monitoring Asymptomatic Patients

Daniel Adu-Gyamfi, C K Tedam University of Technology and Applied Sciences, Navrongo, Ghana
https://orcid.org/0000-0003-3164-0409

Fengli Zhang, University of Electronic Science and Technology of China, China

ABSTRACT

Asymptomatic patients (AP) travel through neighborhoods in communities. The mobility dynamics of the AP makes it hard to tag them with specific interests. The lack of efficient monitoring systems can enable the AP to infect several vulnerable people in the communities. This article studied the monitoring of AP through their mobility and trajectory towards reducing the stress of socio-economic complications in the case of pandemics. Mobility and trajectory-based technique for monitoring asymptomatic patients (MTT-MAP) was established. The time-ordered spatial and temporal trajectory records of the AP were captured through their activities. A grid-based index data structure was designed based on network topology, graph theory, and trajectory analysis to cater for the continuous monitoring of the AP over time. Also, concurrent object localisation and recognition, branch and bound, and multi-object instance strategies were adopted. The MTT-MAP has been shown to be efficient when experimented with GeoLife dataset and can be integrated with state-of-the-art patient monitoring systems.

KEYWORDS

Asymptomatic, Data, Health, Mining, Mobile, Monitoring, Patient, System, Technique, Trajectory

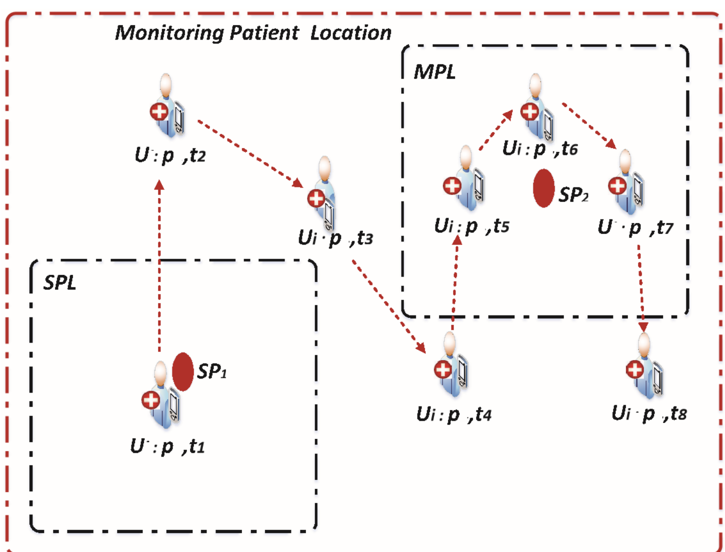
1. INTRODUCTION

In recent times, the attention of the public healthcare settings has been drawn to the eradication of pandemics. As a result of pandemics, patients are often monitored (Crepaldi, et al., 2018) by a team of healthcare professionals. This is to allow them to check the daily health status and/or record the routine activities (Hassan, et al., 2018; Gonzalez, et al., 2008) of the patients. For instance, when patients are visiting and/or travelling to interesting and/or important places, such as theatre, tourist centers, local market, shopping mall, airport, and so forth. It has become necessary to keep records of these activities in order to obtain relevant data (Smyth, 2018; Nogueira, et al., 2018) on the patients to support health research, planning and decision making in the society in the case of pandemics. For example, real-time data which contain the whereabouts of patients with respect to time (Adu-Gyamfi, et al., 2019) will be more appealing to assist public healthcare workers on their decision to derail spreading of contagious diseases. Moreover, there

DOI: 10.4018/JITR.2022010109

have been several reports on the increasing number of patients who are affected on daily basis with chronic health complications (El-Sappagh, et al., 2019). However, managing chronic disease is considerably a global issue which places pressure on the patients, healthcare settings, and the affected communities at large (Esposito, et al., 2018). Asymptomatic patients (AP) have been one of the serious pandemic contributing patients. This is because the AP are usually diagnosed with chronic diseases, but they do not show any noticeable symptoms. Therefore, it is hard for anyone to casually recognized them as they can equally be seen as active as any ordinary persons. Furthermore, irrespective of the health education given to the AP, they can spread diseases directly to vulnerable people, such as family members, co-workers, travelers, etc. This situation can result to possible pandemic. Therefore, it is prudent to establish compelling novel techniques to facilitate the development of efficient mobile health information systems for monitoring the AP when they are outdoor in order to derail pandemics. Several techniques have been developed over the years towards the advances in patients' health monitoring strategies (Smyth, 2018). Due to the proliferation of outdoor mobile technologies and devices, many people can wear shirts equipped with sensors for taken records of their health status (Esposito, et al., 2018). Moreover, a comparative study on sensor and real laboratory data in (Smyth, 2018) has revealed that it is quite easier to predict the health status of patients with mobile sensor data. According to (El-Sappagh, et al., 2019), it is also easier to track the fitness of patients using mobile sensor data. Nonetheless, in (Or, et al., 2018) it is proposed that most of the modern and/or state-of-the-art health information systems still face challenges in terms of their output which is less efficient, and that is due to their under-explored techniques (or approaches) underlying the design of the systems (Nogueira, et al., 2018). In (Rajkumar, et al., 2018), there is a proposed improved software intelligent system towards enhancing the performance of a hearing impairment system in which the system is aimed to provide solution to audio logical problems. In (Singh, et al., 2017) is an introduced deep learning model towards recognition of human activity in which the model is able to learn to classify human activities without prior knowledge. Heuristically, the model will be practically not well fit for investigating the AP who are in continuous traveling routine. This is because prior to real-time monitoring of the AP, this paper identifies the initial health statuses as relevant knowledge to obtain efficient outcome. Moreover, in (Nogueira, et al., 2018) is a proposed biofeedback system technique for evaluating the quality of life of patients. In (Crepaldi, et al., 2018) is evaluated health information systems based on software engineering metrics. The work has laid emphasize on the analyses of system methods and designs, and has proposed that many existing techniques (or approaches) are not efficient as a results of over use of resources, etc. In (Wu, et al., 2018) is identified peculiar issues on monitoring of general object mobility. The first issue is on how to correlate spatial and temporal information on objects such as patients towards location predictions. The second issue is on how to develop effective and efficient practical oriented predictive systems such as health information decision support systems on a large scale. To closely examine the technicalities involved in the two issues. It can be said that it is quite harder to find readily available solution to them. Nonetheless, in this paper a strong motivation is derived from the proliferation of the current ubiquitous built-in GPS mobile devices and the available methods for determining the location of objects (Li, et al., 2015) (Lee & Holzinger, 2016). Therefore, with the aid of these devices and technologies, it is quite simpler to find the geographical location, spatial position, time stamp, place of visit of mobile outdoor patients. In (Pawar, et al., 2012), patient monitoring can be defined as the continuous (or periodic) measurement and analysis of a mobile patient bio-signals from a distance by employing mobile computing, wireless communications, and networking technologies. To clearly outline the research question, supposing we want to monitor the various places where the AP often travel or visit in a community. We classify or map this question to that of Maximize Range Sum (MaxRS) problem (Amagata & Hara, 2017) which is a common problem in spatio-temporal databases. It is not trivial to achieve solution to this problem due to the dynamics and varying degree of interests of the AP. The varying interest could be attributed to either inherent or acquired circumstances of the present location of the AP. Therefore, tagging the AP with a particular interest incrementally in continuous time intervals is a challenge. This challenge is associated with the excessive computational needs to execute the high dimensional data (Lee & Holzinger, 2016) as well as the software and hardware resource requirements needed to be met. The focus of this paper is mainly to delve into mathematical modelling coupled with that of trajectory data mining techniques in order to establish the MTT-MAP (i.e. Mobility and Trajectory based Technique for Monitoring Asymptomatic Patients) as an efficient data dependent monitoring technique to be used in the healthcare settings. The general objective of MTT-MAP seeks to ensure proper and efficient monitoring of the AP in a community. Whereas the specific objective seeks to facilitate the stay place (SP) detection of the AP via place of interests (POIs). Therefore, the POIs are determined based on the spatio-temporal trajectories (Yu, 2015) of the AP which consists of their routine activities during mobility. To determine the POIs, there exist two categories of SP (Yu, 2015) to be considered. These are single-point location (SPL) which refers to a location where the AP travelled or visited and spent inordinate amount of time as illustrated in Figure 1, and multi-point location (MPL) which refers to a location where AP travelled or visited and spent relatively a short time as illustrated in Figure 1. Heuristically, both types of SP can significantly contribute to trajectory analysis towards monitoring and predicting of object's location (Yu, 2015). This article offers a considerable novelty in healthcare as a contribution towards improving state-of-the-art mobile patient monitoring system applications in order to aid informed decision towards pandemic eradication. MTT-MAP has been experimented using GeoLife big dataset and proven efficient for monitoring the AP. The remainder of this paper is as follows. The method underlying the MTT-MAP is detailed under Materials and Methods section. Under Results and Discussion section, we present plotted graphs of the simulation results of the MTT-MAP algorithm and highlight on the discussions of the findings and possible improvement. Finally, we conclude the paper with some future research directions under Conclusions section.

Figure 1. Category of Stay place (SP). Showing SP1: single-point location (SPL), and SP2: multi-point location (MPL) of patient based on POIs. U_i represents user identifier for the AP, whereas p and t represent the spatial position (POIs) and time-stamp respectively.



2. MATERIALS AND METHODS

This section outlines the materials and methods that are used to establish the MTT-MAP. Given a set of POIs, unit weight and user-specified rectangle r . Suppose that we want to monitor the various places where the AP will travel or visit in a community. This problem can be map to that of MaxRS problem which is a commonly occurring problem in spatio-temporal databases (Amagata & Hara, 2017). Especially, when analyzing mobility patterns to determine SP. An example of MaxRS approach includes the G2 algorithm (Amagata & Hara, 2017). This present paper tries to find an enclosing r which contains the maximised weighted-sum of POIs, and locates a centroid of r as SP. The SP is combined with other methods to determine the disease spreading potentials of the AP. Concurrent object localisation and recognition, branch and bound formalism, and multi-object instance (Yeh, et al., 2009) are adopted as far more efficient methods to ensure proper monitoring of the AP in any streaming environments. Where the current position and the time-stamp of the AP are considered in a monitoring location. Further, spatio-temporal trajectory analysis which is a sub field of trajectory data mining has been applied to find solution to many real life problems (Lettich, et al., 2015; Costa, et al., 2017; Yang, et al., 2013; Giannotti, et al., 2009; Adu-Gyamfi, et al., 2019) including this present study, and hence making this study not an entirely new research in computing.

We used GeoLife (Yu, et al., 2010) dataset for the experimental evaluation. It is a GPS based dataset which is open source provided by the Microsoft Asia, Beijing. The details as to how to find it can be obtained in the reference (Yu, et al., 2010). The dataset consists of trajectory of mobile users' outdoor mobility and social networking. Upon testing the algorithms with the dataset, a single AP was emulated as ordinary mobile user. The AP was considered as performing daily life activities, such as spotting, travelling, shopping, hiking, sight-seeing, visiting, and so forth (Adu-Gyamfi, et al., 2019). Together with the AP, the neighborhood and which corresponds to the specific location was represented in the model as rectangle r. Whereas the entire community was represented as R. The default size of the R was 1000 x 1000. The other parameters are defined in Table 2. The graphical results of the simulation are provided in Figure 8, Figure 9, Figure 10, Figure 11, and Figure 12. In the various graphs legend, the compared G2 algorithm has been indicated as G2, and TMAXWS for the proposed MTT-MAP.

2.1 Trajectory

First of all, we define trajectory as a sequence of tuple given by $T = \left\{\left(x_1, y_1, ta_1, ts_1\right), \ldots, \left(x_k, y_k, ta_k, ts_k\right)\right\}$. Such that $ta_1 < ta_2 <, \ldots, < ta_k$ and $ts_1 < ts_2 <, \ldots, < ts_k$, where k denotes length (size) of data. The x and y are the location (or spatial position) coordinates, whereas ta and ts represent arrival time and stay time respectively. It is essential to note that T is subject to a unit positive weight, and other effects such as altitude. However, altitude is insignificant so far as trajectory analysis is concerned (Yu, 2015). Here, we refer to spatio-temporal trajectory as the time bounded spatial activity carried out by outdoor mobile object such as AP. Trajectory monitoring instances. Assume that $P\left(n\right)$ are countably-infinite POIs. If $p\left(i\right)$ is any identified POIs then it follows that $p\left(i\right) = \left\{x, y, w\right\}$, such that $p\left(i\right) \in P\left(n\right)$ and $p\left(i\right) * w \in \mathbb{R}^+$, where x and y are spatial position coordinates respectively, and w is a positive weight defining the features of the monitoring location. Supposing we want to find SP, this can be done by considering the most visited and stay place within the monitoring location. In that regards the monitoring location becomes the community where AP lives. Using Eq.1, we can obtain $p\left(i\right)w$ is any weighted identified POIs. Let's represent m as search space and n as large location under monitoring, such that $i = 1, \ldots, m$ and $m \in n$. This implies that, if $m = n$ then the task is not relevant to execute, and thus m , which represents the search space is equal to n which represents the entire monitored location. Moreover, if $m > n$ then this condition cannot be satisfied as it is not practical to execute such a task:

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

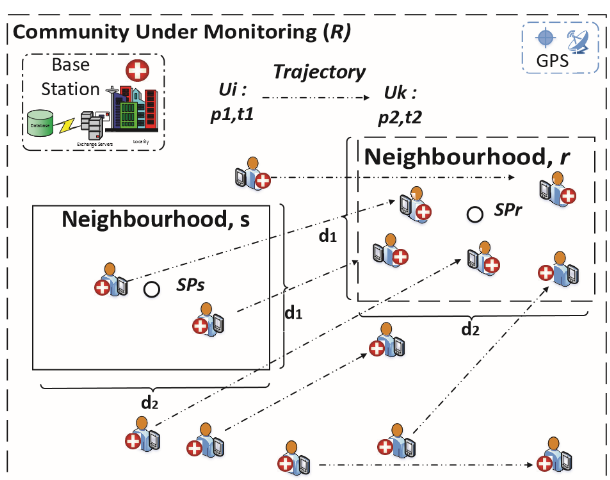
Assuming R is a community. Let's denote r as monitoring location (i.e. specific) within location R (i.e. general) and p(i) as individual POIs, such that r ∈ R. Therefore, we can heuristically monitor r continuously to obtain POIs. Let $p\left(i\right)w \in P\left(n\right)$ be the weighted-sum of POIs. We need to ensure that the weighted-sum of POIs is maximised in r. Therefore, we heuristically establish the objective function in Eq. 2. Where p(i)w represents a weighted POIs. As a result, monitoring $r\left(w\right)$ is the same as monitoring the SP. We have illustrated the process in Figure 2:

$$r\left(w\right) = \underset{i \in n \cap R}{\arg\max}\, p\left(i\right)w \tag{2}$$

2.2 Concurrent Localisation and Recognition of POIs

In this section, we present how to locate, recognise and monitor AP in any community. Such that we can identify the POIs in order to obtain SP which will intern be used to guide the decision making on pandemic eradication. By so doing, we adopt concurrent localization and recognition (Yeh, et al., 2009) strategy to find a bounding rectangle r (i.e. localization) that encloses the various POIs. Also, we try to find a model, say $p\left(i\right)$ that maximizes a prediction score L for the POIs (i.e. recognition). Assuming there are n location features and s models. At any instance of predicting the POIs, there would be prediction partial score $L_{(i)}$ which can either be an output of support-vector machine (SVM) classifier (i.e. identification) or tree-based index (i.e. categorization). Alternatively, L can be obtained by accumulating only the $L_{(i)}$ considering those from various POIs. We approach that by computing the L incrementally when the POIs has been discovered (or where any POIs get obsoleted or discarded). Assuming we have an anticipated prediction score of $s_{(k)}$ when searching for subsequent

Figure 2. A derived SP. Showing the trajectory of asymptomatic patient based on POIs, where SPs and SPr are the obsolete and updated centroid indicating SP respectively. Base station ensures the acquisition, storage and management of the monitoring data from the GPS device.



POIs (i.e. evolution). We compute $s_{(k)}$ by creating an upper bound. Therefore, we employ branch and bound formalism (Yeh, et al., 2009; Lampert, et al., 2008) to enable us to obtain an $\arg\max_k s\left(k\right)$ for all the POIs. Drawing on the method of plane-sweep mechanism (Yeh, et al., 2009; Lampert, et al., 2008), we create an interval that captures all the POIs. An iteration is then performed to split the interval into two disjoints. Using the two disjoint intervals, we apply bounding strategy to decide on the next POIs location. The iteration is terminated when k becomes a singleton, and hence we consider that at this stage the optimal location is found indicating the POIs. Now let's express as input the n location features given by the triplets, $F =: \{\left(ax_k, by_k, cL_{(k)}\right) \mid k : 1, \ldots, n\}$, where L corresponds to the location feature which is an M-dimensional vector. Whereas a, b and c are constant coefficients. If $L_{(i)}$ becomes the ith element of $L_{(k)}$, and thus i < k, then we can now find the partial scores of $L_{(i)}$ which the various POIs contribute to the total L. Heuristically, we can establish the objective optimization in Eq.3, where r(w) is any specified location. It follows from Eq.2 that $F_{(r)}$ represents the POIs that are spatially bounded by r. Therefore, we can obtain the POIs by computing $\vec{L}_{(k)}$, where, r and R correspond to the search space (e.g. neighborhood) and large monitored location (e.g. community) respectively, such that $r \in R$. This implies that we can have more than one neighbourhood within a community, such that the AP is being monitored whenever moving from one neighborhood to another within the same community, and based on the SP in the community:

$$r\left(w\right) = \underset{i, r \in R}{\arg\max} \sum_{k \in F\left(r\right)} \vec{L}_{(k)}\left(i\right) \tag{3}$$

Suppose we want to train SVM classifier for any POIs. The POIs is taken as input to evaluate the SV M classifier margin by comparing the testing sample to each of the training sample. We compare the weighted POIs that has the support vector with other aggregated partial scores in order to obtain a set of similarity scores. We sum up the similarity scores based on a certain SVM offset value δ and express the margin as weighted-sum (i.e. all the partial similarity scores). This approach is very similar to the one used in the reference (Yeh, et al., 2009). Let $\vec{\mu}_{p(i)}$ be any vector with support vector weights for sth training set of POIs, such that $w \in w\left(k\right)$, where w is the positive weight attributed to the location. Assuming there is ith element of s which corresponds to the weight of the ith SVM. Heuristically, consider any $p\left(i\right)$ selected from the POIs. It implies that the model label and feature score can respectively be given as $l_{p(i)}$ and $c_{p(i)}$. Therefore, the anticipated partial score can be computed using Eq.4. Note that $h\left(\cdot\right)$ evaluates to 1 when $\left(\cdot\right)$ is true, and thus it implies that there is a solution. Otherwise $h\left(\cdot\right)$ evaluates to 0, and thus it implies that there is no solution:

$$\vec{L}_{(k)}\left(i\right) = \sum_{p(i) \in w_{(k)}} h\left(l_{p(i)} = i\right) c_{p(i)} \vec{\mu}_{p(i)}\left(i\right) \tag{4}$$

where:

$\vec{L}_{(k)}\left(i\right)$ = any identified partial score of position data

$p\left(i\right)$ = any identified position data of the AP

w = any selected support vector weight based on the monitored location
$w\left(k\right)$ = aggregated support vector weight based on the monitored location

$l_{p\left(i\right)}$ = any identified model label

$c_{p\left(i\right)}$ = any identified feature score

$\vec{\mu}_{p\left(i\right)}$ = any single/identified vector considered in the position data

2.3 Branch and Bound Formalism

Drawing on the concept of branch and bound formalism (Yeh, et al., 2009), we ensure that search space is disintegrated into disjoint subspaces. Through that we discover optimal value to represent the POIs. Heuristically, we assume w_0 and w_1 as maximum weighted-sum POIs and weight of monitored location respectively. Let tol represents user error tolerance, we conjecture that $w0 \geq w1\left(1 - tol\right)$. Therefore, there would be trade-off between query efficiency and tol. However, tol can be tuned such that it will be slightly higher than the realistic (or actual) error rate.

2.4 Branch Technique

In this section, we discuss how our algorithm applies the branch technique, see algorithm 1 to find the optimal location as POIs. Let d be a subspace, and d_1 and d_2 be any two smaller subspaces. Base on upper bound estimation, and thus by considering the x and y coordinates of the general monitoring location (i.e. R, and m object models, we obtain Z_1 and Z_2 as two subsets of equal-size (i.e. as candidates). We transform subspace d as:

$$d\left(i\right) =: \left\{\left(-x\left(i\right), +x\left(i\right)\right), \left(-y\left(i\right), +y\left(i\right)\right), m\left(i\right)\right\}$$

such that $i = 1,2$ (i.e. binary), and y is split as d_1 (i.e. points in Z_1 are removed from $-y$ as well as all x_1.

2.5 Bound Technique

In this section, we discuss how our algorithm applies bounding operation in order to determine the largest and smallest bounding locations. We do that by considering a given subspace d using the $-x$, $+x$, $-y$, $+y$ and m parameters. Considering the x and y extremities, we obtain $+b$ (i.e. $+L\left(i,j\right)$) and −b (i.e. $-L\left(i,j\right)$) as the largest and smallest rectangle (i.e. location) sizes respectively. Let a represent label for object model m, such that $a \in m$. Using the dimension of a, we consider $-b$ and $+b$ intervals, and locate a space which includes as many positive weighted POIs (i.e. $\vec{Z}_k\left(a\right) > 0$), and thus excluding as many negative weighted POIs (i.e. $\vec{Z}_k\left(a\right) < 0$). At this stage the Eq. 2 can heuristically apply. If there are more positive values than $+b$ and few negative values than $-b$ in the rectangle (i.e. location), then optimal solution can be assumed for POIs. Otherwise it will be irrelevant to compute or query any new rectangle (i.e. as optimal or best location to be assumed as the POIs). We can heuristically compute the upper-bound say $+J$ via Eq. 5. Similarly, lower-bound say $-J$ can be computed via Eq. 6, and thus by adding many negative values, and subtracting many positive values. Both $+J$ and $-J$ are assumed to have object model a as a function of subspace d:

$$+J_a\left(d\right) = \sum_{j \in d\left(+b\right)} h\left(\vec{Z}_k\left(a\right)\right) + \sum_{j \in d\left(-b\right)} h\left(\vec{Z}_k\left(a\right)\right) \tag{5}$$

$$-J_a\left(d\right) = \sum_{j \in d\left(-b\right)} h\left(\vec{Z}_k\left(a\right)\right) + \sum_{j \in d\left(+b\right)} h\left(\vec{Z}_k\left(a\right)\right) \tag{6}$$

where $+h\left(x\right)$ and $-h\left(x\right)$ are functions. If x is positive then $+h\left(x\right)$ is evaluated. On the other hand, if x is negative then $-h\left(x\right)$ is evaluated. Otherwise, the function evaluates to 0 when x is neither positive nor negative. Finally, based on the subspace d we compute the overall upper bound $+J$ and lower bound $-J$ for the object model a as given in Eq. 7 and Eq. 8. Therefore, any SV M classification problems can be trained by adjusting the bounding estimates using offset values, say δ_a if the respective SV M classifier via Eq.7. We have constructed the algorithmic pseudocode for the POIs execution in Algorithm 1:

$$+J_a = J_a + \delta_a \tag{7}$$

$$-J_a = J_a + \delta_a \tag{8}$$

where:

$+J_a$ = overall upper bound

$-J_a$ = overall lower bound

J_a = bounded space with object model

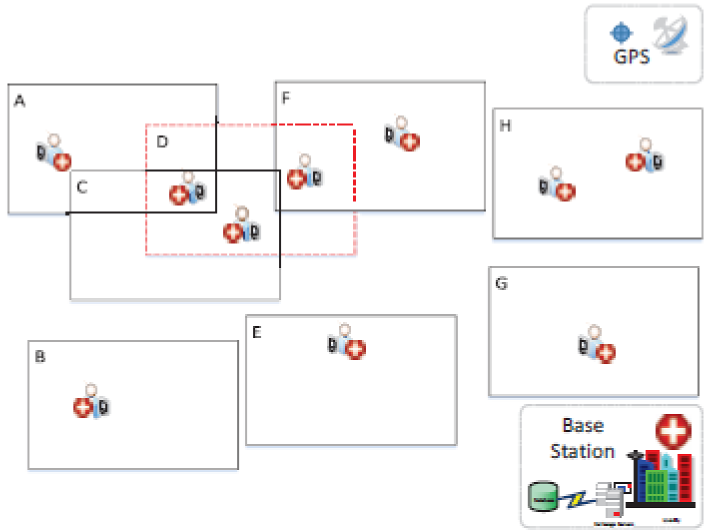
δ_a = Offset value with object model

2.6 Graph Analysis and Index Data Structure

Graph theory and network topology (Lee & Holzinger, 2016) have been used in various domain of researches to find solution to complex and dynamic network related problems. Thus, graph base and topological data mining provide knowledge representation base on graph analysis to construct novel queries for the analysis of complex and dynamic network related problems. Such as analyses of network connectivity behavior (Hu, et al., 2019), patterns and impacts of network systems, continuous spatio-temporal trajectory joins (Bakalov & Tsotras, 2006), etc. Here we discuss the construction of a generalised index data structure for processing the trajectory of AP based on graph analysis. Let's consider Figure 3 which is an illustration of tracking instances of POIs. Assuming there are ten positions instances of the AP representing POIs enclosed within rectangles A, ..., H (i.e. representing individual places of visit or neighborhood). As shown in Figure 3, it can be seen that red broken-line rectangle provides the optimal solution for having three of the total POIs as the maximised weighted-sum.

Let $G = \left(V, E\right)$ denote a dynamic graph. Vertices V represents the position data leading to the discovery of POIs. As shown in Figure 3, vertices are further transcribed as rectangles. An overlap of two vertices generate edges E. Therefore, one-to-one mapping of vertices corresponds to edges. As shown in Figure 3, we perform transformation of POIs into vertices. As a result, we can now construct the dynamics of monitoring POIs in continuous time intervals via graph analysis. Following

Figure 3. Example of tracking position instance of asymptomatic patient. Showing the various instances of POIs in monitoring location or neighborhood identified by A, B, C, D, E, F, G and H. The neighborhood D has the maximised weighted-sum of 3 POIs and it is classified as the optimal solution (or neighborhood) to represent SP of the patient. Base station ensures the acquisition, storage and management of the monitoring data from the GPS device.



the dynamic process, we can derive the trajectory instances of AP based on Figure 4. Therefore, we construct a tree graph to represent the dynamics of trajectory instances of AP (Feng, et al., 2015) based on POIs as shown in Figure 5. Further, we construct a tree data structure by concentrating on the core or usual trajectory of AP as shown in Figure 6. It should be noted that graph $G = \left(V, E\right)$ is dynamic, hence it can be reconstructed to reflect the dynamics of mobility. Therefore, if $r_i, r_j \in V$

Figure 4. Transformation of POIs into rectangles. Showing rectangles r_1, …, r_8 and intersection generated by p(i,j) and p'(i,j).

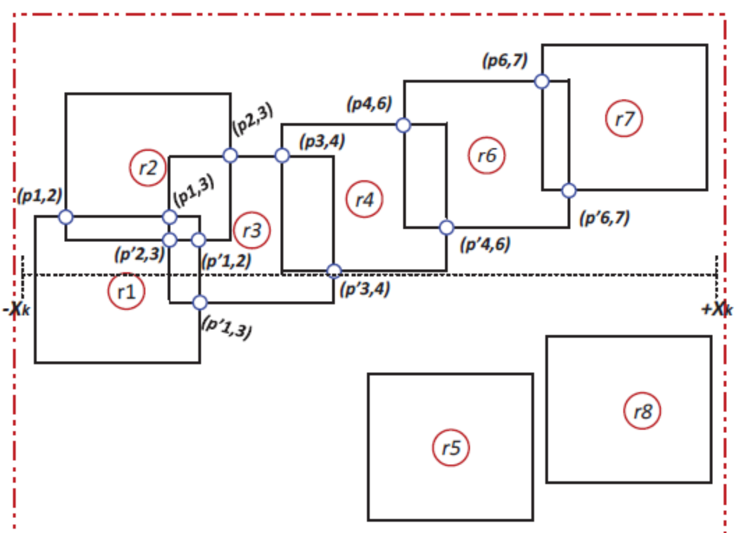
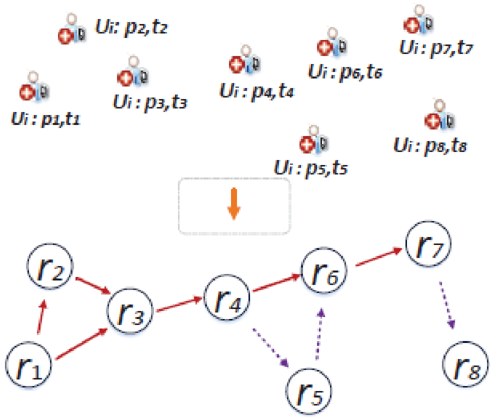


Figure 5. A tree graph of POIs. Showing trajectory instances of AP based on POIs. Ui represents user identifier for AP, whereas p and t represent the spatial position (POIs) and time-stamp respectively.



is generated, then by convention r_i is older than r_j. We present the relationship of graph $G = (V, E)$ in Table 1. We apply the concept of Origin-Destination (OD) matrix to construct a list adjacency matrix. By so doing, it is now possible to design a generalised grid based index data structure to cater for the continuous evolution of POIs leading to discovery of SP. Thereby, we construct a grid based index data structure as shown in Figure 7.

Drawing on the study of graph theory, three properties can be defined based on Figure 6 (Amagata & Hara, 2017). Firstly, if $r_i, r_j \in V$, then $p(i) \neq p(j)$. Thus, if and only if there exists an overlap between $V(r_i)$ and $V(r_j)$. For instance, where $V(r_i)$ is older than $V(r_j)$, there exists $(V(r_i), V(r_j)) = r_{ij} = e(i) \in E$. Also, if $e(i) \neq e(j)$, then $NV(r_i) \neq NV(r_j)$, and thus making $p(i)$, a subspace of $V(r_i)$. Secondary, if P grid cells or spaces contain a set of p_i enclosed in r_i, then Eq. (2) is verifiable. It means any obsoleted vertices are no longer needed to undergo maintenance as a result of the corresponding edges being held by the existing vertices. Thirdly, if $r_i \in V$, then vertex r_i is considered as obsoleted in circumstances where there are other older vertices, and hence any p_i becomes the current position of AP. When AP moves from one place to another, the position data need to be updated, and thus represented R^u. Therefore, we compare R^u with all the existing POIs so as to update the grid cell or data structure accordingly via Figure 5. If m represents newly

Table 1. A relationship of vertices, edges, and next neighbour vertices based on graph G

Vertex ($V(r_i)$)	Edge (e(i))	Next Neighbour Vertex($NV(r_i)$)
1	e1,e2	2,3
2	e3	3
3	e4	4
4	e6	6
6	e7	7
7	null	null

generated POIs and n represents the rate, then the time complexity for the process becomes $O(mn)$, and that is quite high. Therefore, we need to find more suitable solution to improve on the complexity. By so doing, we modify $G=(V,E)$ as illustrated in Figure 5 to obtain $G(i,j)=(V(i,j),E(i,j))$ as illustrated in Figure 6. Thereby, cell $g(i)$ or $g(j)$ can be maintained in the graph $G(i,j)$. As a result, if m new vertices are generated and added to $G(i,j)$, then overlaps need to be checked with the corresponding g_i and g_j in order to update the grid cell data structure by a factor of $G^u(i)$. Therefore, the time complexity improves from $O(mn)$ to $O(m)$ for the total overlap at this stage. Therefore, the overall time complexity of the proposed algorithm becomes $O(|G^u(i)|m^u, n^u)$, which is now reasonable. Where m is the average number of vertices undergoing overlap, and n is the average number of overlap. It should be noted that $m^u << m$ and $n^u << n$. Hence, the cost of storage becomes $O(|V|+|E|)$ for all $G(i,j)=(V(i,j),E(i,j))$, and for every r_i to maintain $p(i)$. Also, the worst space complexity becomes $O(|V|^2)$ or n sounds of overlap, and that is almost equal to $O(|O|^2)$ which is quite impractical to evaluate as worst space (Amagata & Hara, 2017).

2.7 Continuous Monitoring of POIs

Making use of Figure 7, we can perform real time monitoring of POIs towards achieving SP. This is done by following the trajectory of the AP (Ebadi, et al., 2017). We deduce the maximised weighted-sum of POIs and obtain a centroid as SP. Knowing the SP at some instance, we can compute newer SP by understanding AP mobility pattern (Gonzalez, et al., 2008; Cho, et al., 2011). Let $l(w)$ and $p(i)'$ represent new SP and new POIs respectively. Supposing there is an existing cell $g \in G$. If $g * w > l'(w)$, then all the vertices in $V(i,j)$ (i.e. R) do not contain $l(w)$. Therefore, no need to compute the exact solution for $p(i)$. Moreover, with $g \in G$ and $g * w \geq l(w)$, where vertices (or rectangle) $r_i \in V(i,j)$, if $l'(w) < p(i)w$ (i.e. in term of time of generating position data), then any newly generated ri' does not contain $p(i)w$. Therefore, it is not necessary to compute the exact

Figure 6. A tree data structure of POIs. Showing the core or usual trajectory instances of AP based on POIs.

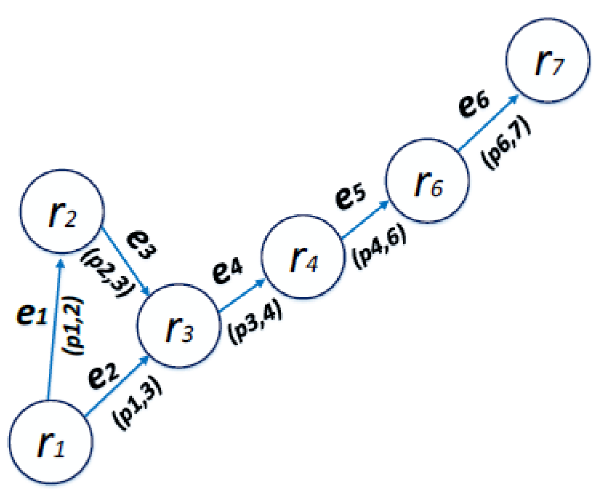
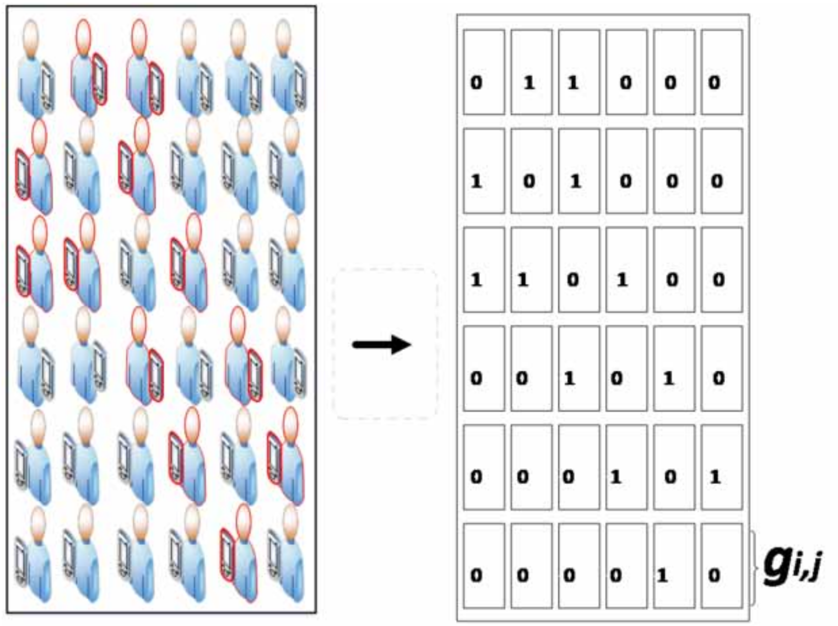


Figure 7. A schematic view of data structure for maintaining any answers for a Patient's POIs query in continuous case



solution for $p(i)w$. Therefore, Eq. 11 can be used to evaluate evolving SP. Intuitively, set k-th largest weight as threshold upon continuously monitoring the top k of the POIs. Also it is important to bound any error rate based on a defined user-tolerance (tol). Heuristically, establish the optimization function of that as follows:

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

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

$$g = \mathop{\arg\max}_{g_{i,j} \in G \cap R} g * w \tag{11}$$

where:

$l'(w)$ = updated location data

$l(w)$ = current location data

$p\left(i\right)$ = any identified position of the AP

The algorithmic pseudocode for monitoring the POIs is given in Algorithm 2 and Algorithm 3.

3. RESULTS AND DISCUSSION

3.1 Results

Heuristically, a compelling grid based index data structure as shown in Figure 7 and algorithm as shown in Algorithm 1, Algorithm 2 and Algorithm 3 have been provided in the present study to be used for monitoring AP. Relatively, MTT-MAP has generally proven better for monitoring naturally occurring phenomena in mobile objects spatial and temporal databases.

3.2 Experimental Setup

The MTT-MAP algorithm was implemented on Windows 10 OS personal computer with Intel Core i5, CPU 4.1 GHz and 8.00 GB RAM. Programming and simulation were done with Python language using Python version 2.7 software. Scikit library, Numpy, Microsoft Visio, and among others were mounted. Experimental parameters are provided in Table 2.

The parameters n, d and m as shown in Table 2 were used for the simulation and evaluation. The average computation times were measured for each of the parameters with simulation results shown in Figure 8, Figure 9, Figure 10, Figure 11, and Figure 12. As a limited study, there were other similar approaches and/or properties that the MTT-MAP was not compared with them due to the massive size of the dataset and the huge computing resources involved.

3.3 Analysis and Discussion

The previous techniques most often over use resources, concepts, and among others (Costa, et al., 2017), as they do not consider the natural dynamics of objects mobility in a case of continuous spatial data streams. In the present study, the MTT-MAP approach was mainly evaluated on the execution time of continuous queries of the location and position of the AP. The results have proven fruitful and also have testified that MTT-MAP is very efficient to be used for monitoring a continuous outdoor mobile patient that is regarded as a maximised range-sum problem.

The impact of the number of spatial positions defined as n in Table 2 was tested. The computation time for finding each position data increased linearly with increasing position data as shown in Figure 8. That resulted to several overlaps of the POIs, whereby the OverlapComputation(.) and ExactWeightComputation(.) functions of the MTT-MAP algorithm heavily got overburdened to execute. That implied the larger the size of monitoring location, the more time will be needed to produce all the position data for the AP and to subsequently obtain the various POIs.

Table 2. Experimental parameters

Param	Default	Variables	Indicators
n	500	100; 250; 500; 750; 1000	No. of position
d	1000	100; 500;1000; 1500; 2000	Rectangle sizes
m	100	50; 100; 200; 500; 800	Generation rate
tol	-	0; 0.10; 0.20; 0.30; 0.40	User tolerance
k	-	10; 20; 30; 40; 50	Query answers

Figure 8. A graph of running time against position size. Showing the impact of n as amount of position data available for execution.

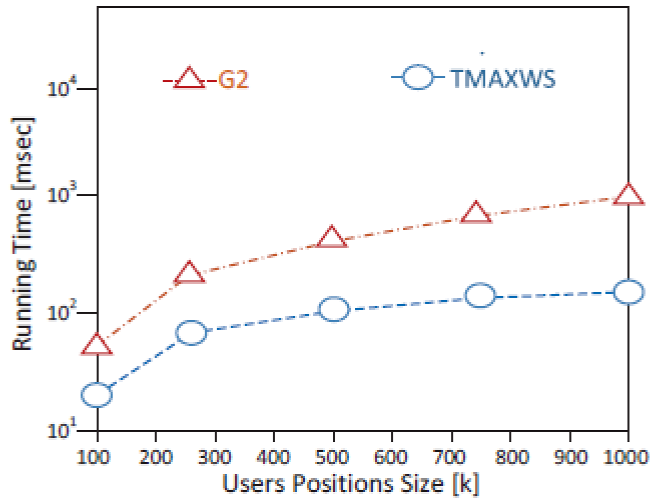
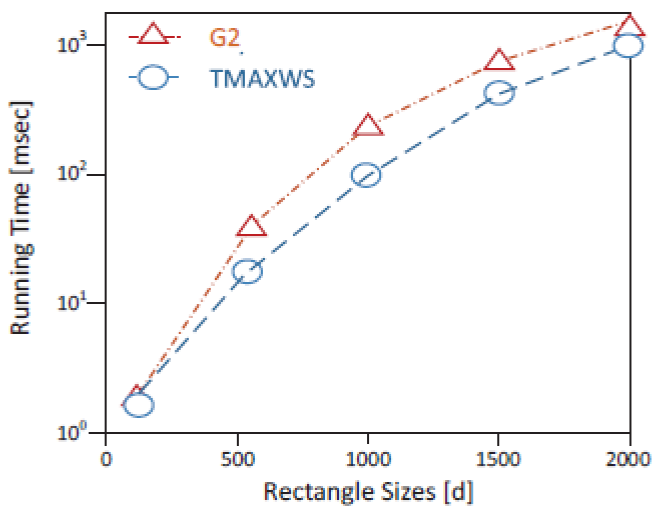


Figure 9. A graph of running time against rectangle sizes. Showing the impact of d as the increasing size of location data.



The impact of rectangle size defined as d in Table 2 was tested. A skewed distribution was observed as shown in Figure 9. There was a proportionate increased in the running time of the algorithms with more overlaps when the rectangle size was increased. That implied how appropriate it was to monitor the AP in a sizeable location.

The impact of rate of generating the position data defined as m in Table 2 was tested. Less than 50 position data were generated per second by the algorithms as shown in Figure 10. That implied how faster the algorithms completed the POIs queries execution for the AP.

The impact of user tolerance defined as tol in Table 2 was tested. This was done only with the MTT-MAP to observe the error rate and computation time required for executing the queries as shown in Figure 11. The tol was indirectly proportional to the running time of the algorithm. That implied a trade-off between the query processing efficiency and result quality. The more the user fine tune

Figure 10. A graph of running time against user generation rate. Showing the impact of m as the rate of increasing position data.

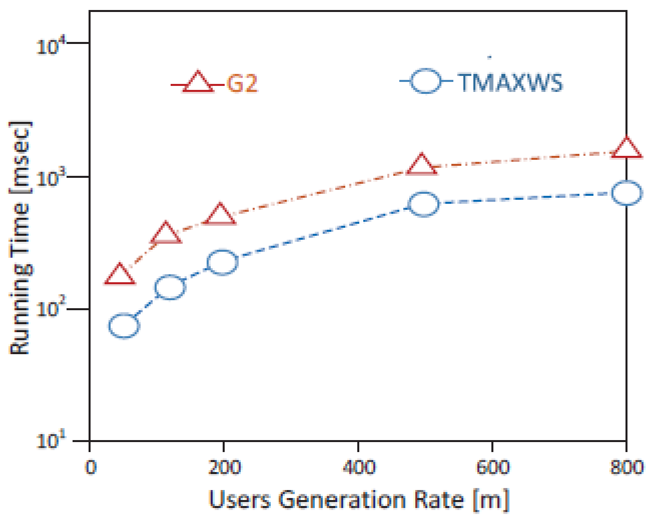
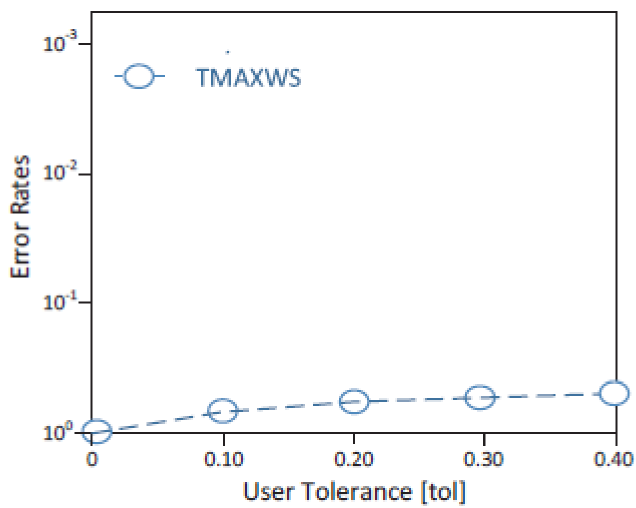


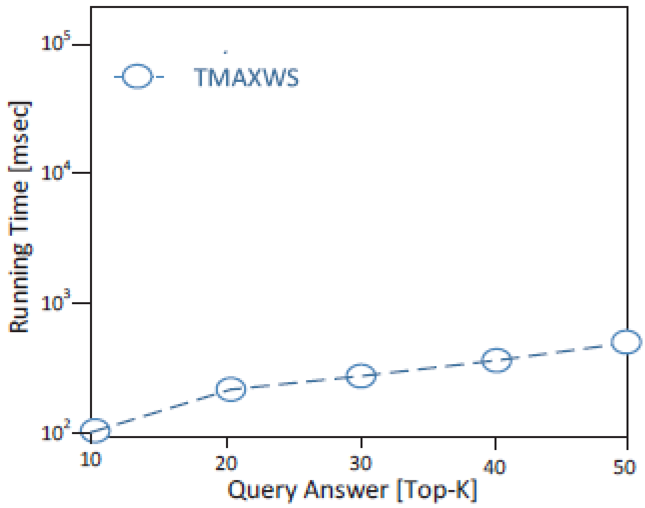
Figure 11. A graph of error rates against user tolerance. Showing the impact of varying user tolerance tol for approximate solution.



the POIs queries towards quality result, the more time the algorithm took to efficiently produce the expected outcome for obtaining the SP and vice versa.

Finally, the impact of generating top−k position data as defined in Table 2 was tested. This was done only with the MTT-MAP to obtain the optimal SP for the AP. When k was increased, the execution time was proportionately increased as shown in Figure 12. That implied the k as directly proportional to the execution time for obtaining newer updates of SP. Therefore, it will always be advisable to approximate the queried results based on marginal error rate determined by the user as tolerance (tol). In effect, it will not be necessary to examine exact query solution of the SP so far as real life applications are concerned. Thus an approximated SP will be enough to an informed decision as to whether the AP can be a disease spreading potential in the present location and position.

Figure 12. A graph of running time against query answer. Showing the impact of top k query results.



4. CONCLUSION

This paper has established a Mobility and Trajectory Technique for Monitoring Asymptomatic Patient (MTT-MAP). The Asymptomatic Patients (AP) were monitored through their mobility and trajectory in order to reduce the stress of socio-economic complications so far as pandemic is concerned. Notably, the AP were people of potentially spread chronic diseases and may cause pandemic. As such, the POIs of outdoor mobile AP who traveled through different neighborhoods in a specific community were studied. The mobility patterns of the AP were analyzed with the goal of finding the places where they spend most of their time known as stay place (SP) in order to determine possible pandemics. The problem of mobility patterns analysis and determining SP was mapped to that of Maximize Range Sum (MaxRS) problem which is a commonly occurring problem in spatial and temporal databases so far as queries of objects location are concerned. MTT-MAP as a data dependent technique was derived from the methods of concurrent localisation and recognition, branch and bound formalism, graph theory and network topology that were intertwined with spatial and temporal (i.e. spatio-temporal) trajectory data mining. The MTT-MAP algorithm was fundamentally produced from a grid based index data structure to cater for the continuous monitoring of mobility dynamics of AP. The study provided an overview of the technicality of monitoring the AP as a clue for enhancing the design and efficiency of prospective mobile health monitoring systems in the case of pandemics. The MTT-MAP algorithm was rigorous simulated with an open source GeoLife big dataset and it was proven efficient with respect to computation time of execution over that of the previous G2 algorithm which has similarly been applied to spatial data streams. The present study was devoted to the healthcare settings in particular and the computing community at large. The motive was to establish a compelling technique to aid the health professionals to find efficient solution to the challenges involved in patients monitoring in the case of pandemics.

The subject of future work will include further improvement and test of scalability of the MTT-MAP to cater for extra computation and storage of POIs by considering tight upper bound. It will be useful to also intensify the study of trajectory data mining techniques, as they involve natural processes and artificial intelligence concepts to achieve better and efficient solution to present societal challenges in general.

REFERENCES

Adu-Gyamfi, D., Zhang, F., & Zhou, F. (2019). Finding influential location via user mobility and trajectory. In *Proceedings of ICAECT 2019 Advances in Electrical and Computer Technologies*. Springer.

Adu-Gyamfi, D., Zhou, F., Zhang, F., Kibiwott, K. P., & Tattrah, V. D. (2019). Real-time monitoring of mobile user using trajectory data mining. In *2019 Third IEEE International Conference on Electrical, Computer and Communication Technologies*. IEEE. doi:10.1109/ICECCT.2019.8869336

Amagata, D., & Hara, T. (2017). A general framework for maxrs and maxcrs monitoring in spatial data streams. *ACM Trans. Spatial Algorithms Syst.*, *3*(1), 1–34. doi:10.1145/3080554

Bakalov, P., & Tsotras, V. J. (2006). Continuous spatiotemporal trajectory joins. In S. Nittel, A. Labrinidis, & A. Stefanidis (Eds.), Lecture Notes in Computer Science: Vol. 4540. *GeoSensor Networks. GSN 2006* (pp. 109–128). Springer.

Cho, E., Myers, S., & Leskovec, J. (2011). Friendship and mobility: User movement in location-based social networks. *17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. doi:10.1145/2020408.2020579

Costa, C., Chatzimilioudis, G., Zeinalipour-Yazti, D., & Mokbel, M. (2017). Efficient exploration of telco big data with compression and decaying. In *2017 IEEE 33rd International Conference on Data Engineering*, (pp. 1332–1343). IEEE Computer Society. doi:10.1109/ICDE.2017.175

Crepaldi, N. Y., de Lima, I. B., Vicentine, F. B., Rodrigues, L. M. L., Sanches, T. L. M., Ruffino-Netto, A., Alves, D., & Rijo, R. P. C. L. (2018). Towards a clinical trial protocol to evaluate health information systems: Evaluation of a computerized system for monitoring tuberculosis from a patient perspective in brazil. *Journal of Medical Systems*, *42*(6), 101. doi:10.1007/s10916-018-0968-8 PMID:29737418

Ebadi, N., Kang, J. E., & Hasan, S. (2017). Constructing activity-mobility trajectories of college students based on smart card transaction data. *International Journal of Transportation Science and Technology*, *6*(4), 316–329. doi:10.1016/j.ijtst.2017.08.003

El-Sappagh, S., Ali, F., Hendawi, A., Jang, J.-H., & Kwak, K.-S. (2019). A mobile health monitoring-and-treatment system based on integration of the ssn sensor ontology and the hl7 fhir standard. *BMC Medical Informatics and Decision*, *19*(1), 1–36. doi:10.1186/s12911-019-0806-z PMID:31077222

Esposito, M., Minutolo, A., Megna, R., Forastiere, M., Magliulo, M., & De Pietro, G. (2018). A smart mobile, self-configuring, context-aware architecture for personal health monitoring. *Engineering Applications of Artificial Intelligence*, *67*, 136–156. doi:10.1016/j.engappai.2017.09.019

Feng, Y., Ji, M., Xiao, J., Yang, X., Zhang, J., Zhuang, Y., & Li, X. (2015). Mining spatial-temporal patterns and structural sparsity for human motion data denoising. *IEEE Transactions on Cybernetics*, *45*(12), 2693–2706. doi:10.1109/TCYB.2014.2381659 PMID:25561602

Giannotti, F., Nanni, M., Pedreschi, D., Renso, C., & Trasarti, R. (2009). Mining mobility behavior from trajectory data. *IEEE 2009 International Conference on Computational Science and Engineering*. doi:10.1109/CSE.2009.542

Gonzalez, M. C., Hidalgo, C. A., & Barabasi, A. L. (2008). Understanding individual human mobility patterns. *Nature*, *453*(7196), 779–782. doi:10.1038/nature06958 PMID:18528393

Hassan, M. M., Huda, S., Uddin, M. Z., Almogren, A., & Alrubaian, M. (2018). Human activity ecognition from body sensor data using deep learning. *Journal of Medical Systems*, 42.

Hu, H., Zhai, X., Wang, M., & Hu, G. (2019). Graph analysis of network flow connectivity behaviors. *Turkish Journal of Electrical Engineering and Computer Sciences*, *27*, 861–877. doi:10.3906/elk-1808-148

Lampert, C. H., Blaschko, M. B., & Hofmann, T. (2008). Beyond sliding windows: object localization by efficient subwindow search. *Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition*, 1–8. doi:10.1109/CVPR.2008.4587586

Lee, S., & Holzinger, A. (2016). Knowledge discovery from complex high dimensional data. In S. Michaelis, N. Piatkowski, & M. Stolpe (Eds.), Lecture Notes in Computer Science: Vol. 9580. *Solving Large Scale Learning Tasks. Challenges and Algorithms* (pp. 148–167). doi:10.1007/978-3-319-41706-6_7

Lettich, F., Orlando, S., & Silvestri, C. (2015). Processing streams of spatial k-nn queries and position updates on manycore gpus. *SIGSPATIAL '15 Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. doi:10.1145/2820783.2820803

Li, F., Long, X., Du, S., Zhang, J., Liu, Z., & Li, M. (2015). Analyzing campus mobility patterns of college students by using gps trajectory data and graph based approach. *2015 23rd International Conference on Geoinformatics*.

Nogueira, P., Urbano, J., Reis, L. P., Cardoso, H. L., Silva, D. C., Rocha, A. P., Gonçalves, J., & Faria, B. M. (2018). A review of commercial and medical-grade physiological monitoring devices for biofeedback-assisted quality of life improvement studies. *Journal of Medical Systems*, *42*(6), 113. doi:10.1007/s10916-018-0946-1 PMID:29667028

Or, C., Tong, E., Tan, J., & Chan, S. (2018). 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. doi:10.1007/s10916-018-0971-0 PMID:29845400

Pawar, P., Jones, V., van Beijnum, B. J. F., & Hermens, H. (2012). A framework for the comparison of mobile patient monitoring systems. *Journal of Biomedical Informatics*, *45*(3), 544–556. doi:10.1016/j.jbi.2012.02.007 PMID:22406009

Rajkumar, S., Muttan, S., Sapthagirivasan, V., Jaya, V., & Vignesh, S. S. (2018). Development of improved software intelligent system for audiological solutions. *Journal of Medical Systems*, *42*(7), 127. doi:10.1007/s10916-018-0978-6 PMID:29860544

Singh, D., Merdivan, E., Psychoula, I., Kropf, J., Hanke, S., & Geist, M. (2017). Human activity recognition using recurrent neural networks. In A. Holzinger, P. Kieseberg, A. Tjoa, & E. Weippl (Eds.), *Machine Learning and Knowledge Extraction. CD-MAKE 2017* (pp. 267–274). Lecture Notes in Computer Science. Springer. doi:10.1007/978-3-319-66808-6_18

Smyth, P. (2018, May 16). *Researchers combine wearable technology and ai to predict the onset of health problems*. University of Waterloo News.

Wu, R., Luo, G., Shao, J., Tian, L., & Peng, C. (2018). Location prediction on trajectory data: A review. *Big Data Mining and Analytics*, *1*(2), 108–127. doi:10.26599/BDMA.2018.9020010

Yang, Y., Du, B., & Jiang, X. (2013). A human trajectory estimate based on individual mobility pattern library. IEEE GreenCom/iThings/CPScom 2013, 1181–1185. doi:10.1109/GreenCom-iThings-CPSCom.2013.205

Yeh, T., Lee, J. J., & Darrell, T. (2009). *Fast concurrent object localization and recognition*. IEEE CVPR. doi:10.1109/CVPR.2009.5206805

Yu, Z. (2015). Trajectory data mining: An overview. *ACM Trans. IST*, *6*, 1.

Yu, Z., Xie, X., & Ma, W.-Y. (2010). Geolife: a collaborative social networking service among user location and trajectory. IEEE Data (base) Engineering Bulletin, 32–39.

Daniel Adu-Gyamfi received the B.Sc. degree in Computer Science with honors from the University for Development Studies, Ghana in 2010. He holds M.E. degree in Computer Science and Technology in 2014, and also pursued the Ph.D. degree in Software Engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China in 2020. He has published more than twenty papers in both reputable Journals and Conferences. He holds a faculty position as a Lecturer in the Department of Computer Science and Informatics of the University of Energy and Natural Resources, Ghana in 2014-2020, and currently a Senior Lecturer at the Department of Cyber Security and Computer Engineering Technology at the School of Computing and Information Sciences in C. K. Tedam University of Technology and Applied Sciences (CKT-UTAS) in Navrongo, Ghana. He is a member of SIGCHI, Association of Computing Machinery (ACM), etc.

Fengli Zhang received her Ph.D. degree from the University of Electronic Science and Technology of China (UESTC) in 2007, M.S. degree in 1986. She joined the DataBase System and Mobile Computing (DBMC) Lab, University of Illinois at Chicago, USA, during 2000-2002 as a visiting scholar. She has published more than sixty papers and translated or edited 5 books as a partaker. In 1986, She joined UESTC as a teacher. She is a professor in UESTC now.