IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 27 February 2024, accepted 21 May 2024, date of publication 31 May 2024, date of current version 11 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3407776

RESEARCH ARTICLE

Spatio-Temporal Contact Mining for Multiple Trajectories-of-Interest

ADIKARIGE RANDIL SANJEEWA MADANAYAKE[1], KYUNGMI LEE[2], (Member, IEEE), AND ICKJAI LEE[2], (Member, IEEE)

[1]Department of Information Technology, James Cook University, Townsville, QLD 4814, Australia
[2]Department of Information Technology, James Cook University, Cairns, QLD 4870, Australia

Corresponding author: Ickjai Lee (Ickjai.Lee@jcu.edu.au)

ABSTRACT Spatio-temporal trajectory is a movement of an object in space over a certain time period, represented by a series of nodes composed of geospatial location and corresponding timestamp. A large amount of spatio-temporal trajectory data is being gathered through various trajectory acquiring devices by tracking the movement of objects such as people, animals, vehicles and natural events. Various trajectory data mining techniques have been proposed to discover useful patterns to understand the behaviour of spatio-temporal trajectories. One unexplored pattern is to identify potential contacts of targeted trajectories which can be defined as contact mining, that is useful for many applications. One such example would be to identify potential victims from known infected humans or animals, especially when the victims are asymptomatic in a rapid spread of infectious disease environments. Another one would be to identify individuals who have been close contacts with known terrorist networks or law breakers. This paper proposes a robust contact mining framework to efficiently and effectively mine contacts of multiple trajectories-of-interest from a given set of spatio-temporal trajectories. Experimental results demonstrate the efficiency, effectiveness and scalability of our approach. In addition, parameter sensitivity analysis reveals the robustness and insensitivity of our framework.

INDEX TERMS Contact mining, data mining, multiple trajectories-of-interest, spatio-temporal trajectories.

I. INTRODUCTION

A spatio-temporal trajectory refers to a movement of an object through geographical space which is represented by a series of geospatial coordinates, latitude and longitude over a period of time [1]. Massive amounts of data are being generated over a long period of time using different types of trajectory acquiring devices. This will be terabytes of data for a small city with about half a million population, when each individual's trajectories are obtained at every second for a period of a month. This raw spatio-temporal trajectory data contains various types of spatial uncertainties and inaccuracies due to the nature of trajectory acquiring devices which require pre-processing prior to data mining [2]. Trajectory clustering, classification and trajectory pattern mining are several widely studied domains in trajectory data mining, and have been applied to discover interesting patterns in many applications [3], [4], [5], [6], [7], [8].

Contact mining is to find potential contacts from spatio-temporal trajectories which are a beneficial domain where contacts in a close proximity are of interest. In a pandemic situation such as COVID outbreak, identifying contacted individuals of known infected people and isolating them would minimise the rapid spread of disease until a medical solution is discovered. Identifying individuals who have been close contacts with known terrorist networks and law breakers is also vital to decrease criminal activities. It is interesting to study whether spatio-temporal trajectory data can be used to identify these forms of contacts. Even though spatio-temporal data can be used to identify contacts, mining algorithms can take significant amount of processing time due to the massiveness of data.

To address these issues, this study proposes a multi-step contact mining framework to identify contacts from Other Trajectories (OT) (Definition 3) of multiple

The associate editor coordinating the review of this manuscript and approving it for publication was Qiang Yang.

Trajectories-of-Interest (ToI) (Definition 2) and undertakes empirical analyses with various settings.

Initially raw spatio-temporal data is pre-processed in order to identify and rectify the inaccuracies and inconsistencies of data. Then with the availability of user specified attributes, several approaches are proposed to identify contacts for multiple ToI using Minimum Bounding Rectangle (MBR). The aim of this paper is to investigate a scalable, efficient and effective contact mining approach, and compares its performance against baselines modified from existing data mining techniques.

Main contributions of this study are as follows:

- Formulation of contact mining for multiple ToI;
- Proposal of a scalable, efficient, effective and robust contact mining algorithm for multiple ToI;
- Provision of extensive experimental results for performance analysis including accuracy, efficiency, scalability, parameter sensitivity and applicability.

The rest of the paper is organised as follows. Section II reviews relevant studies to identify the literature gap. Section III describes definitions and illustrates the proposed framework for contact mining. Section IV presents a framework of multiple ToI contact mining. Section V examines experimental results and presents major findings to identify the most appropriate approach to find contacts for multiple ToI. Section VI draws conclusive remarks and suggests possible future directions.

II. LITERATURE REVIEW

Three broader domains are analysed and reviewed in this section to assess whether they can be employed to identify potential contacts, and what hinders their applicabilities. Initially a review is conducted to examine techniques which can be used to identify inaccuracies of spatio-temporal data as well as approaches to resolve these problems. Trajectory data mining techniques are widely used in discovering interesting knowledge such as finding anomalies, patterns and correlations within large spatio-temporal datasets for better decision making. Hence the suitability of solving the contact mining problem using these techniques is investigated. Finally, collision detection techniques which can be used to detect the intersections of geometric models are examined to determine whether they can be applied to find contacts from spatio-temporal trajectories.

A. SPATIO-TEMPORAL DATA PRE-PROCESSING

Varying location accuracy levels are witnessed due to various types of trajectory acquiring devices, surrounding barriers, lack of satellites in certain areas and weather conditions which result in measurement inaccuracies [2]. Multipath is another problem caused by reflecting satellite signals which leads to positional errors [9]. Longer paths can be lessened using the Real-Time Kinematic Precise Point Positioning (PPP-RTK) systems [10]. Trajectory acquiring devices capture data in a certain resolution of time and when this sample rate is lower than the required minimum sample, it is identified as spatial uncertainty. Trajectory data may also contain oversampled data as well as inconsistent data. This data has to be resampled into regular time intervals using trajectory simplification techniques [11]. These solutions are being used in the pre-processing stage of this paper to extract correct spatio-temporal data prior to using other trajectory mining methods to identify contacts.

B. TRAJECTORY DATA MINING

In this subsection, we will review major trajectory data mining approaches.

1) TRAJECTORY CLUSTERING

Trajectory clustering is an unsupervised learning method which categorises spatio-temporal trajectory datasets into clusters by identifying similarities of intra-cluster trajectories from dissimilarities of inter-cluster trajectories [12]. This is being used in applications such as object motion prediction, traffic monitoring, activity understanding, abnormal detection and weather forecasting [12]. Trajectory clustering algorithms can be generally categorised into partitioning based, density based, hierarchical based, model based and grid based [13]. Partitioning based algorithms are more popular as they are relatively simple and have the ability of handling large datasets [14]. On the other hand, they have disadvantages such as needing a predefined number of clusters prior to clustering and the impact of outliers [15]. Density based clustering algorithms overcome these issues, but they have their own challenges such as the requirement of predefined parameters and inability to perform well with higher dimensional data and clusters with varying densities [16]. Although, hierarchical based algorithms overcome these issues by considering more attributes in each level, they consume more computational time [17]. Model based clustering computes internal relationships by analysing similar matrix and thus is more efficient in processing data together [18]. Hence these clustering methods can be used to divide the space into several highly populated regions, it is interesting to use these methods as a preprocessing approach to examine whether this can be used to identify contacts.

2) TRAJECTORY CLASSIFICATION

Trajectory classification is a supervised learning technique which categorises trajectories into pre-defined classes [19]. This is useful when there exist predefined labels and a prediction is required. Trajectory classification is used in many applications such as trip recommendations, sharing life experiences, hurricane prediction, security alert triggers and context-aware computing [20]. Trajectory classification could be used as a post-process step for contact mining but cannot be deployed for contact data mining due to the lack of ground-truth training data.

3) TRAJECTORY PATTERN MINING

Trajectory pattern mining describes discoveries of significant, interesting or unexpected patterns in a movement of trajectories [21]. Trajectory pattern mining is categorised into periodic/repetitive pattern mining, frequent/sequential pattern mining and moving together/group pattern mining. Periodic or repetitive pattern mining refers to a moving object which repeatedly follows the same route in constant time periods such as daily, monthly or annually in the same trajectory [22]. These behavioural patterns are useful to predict future movements. This method has uncertainties since the period affects the clustering output. The specification of a period in advance was overcome by the Periodica algorithm [22]. Frequent or sequential pattern mining focuses on multiple moving objects who visit approximately the same place in the same order in relative time [23]. Frequent Spatiotemporal Sequential Pattern (FSSP) mining and Generalized Sequential Pattern (GSP) mining are some of the methods found in frequent pattern mining [24]. Finding important regions from the trajectories and then applying sequential mining is a common approach to mine frequent patterns. Group pattern mining is numerous moving objects staying close in space and visiting the same places at the same time [1]. These patterns can be categorised depending on the shape and density of the group and the duration of movement of objects. There are different types of trajectories which move together in a certain time period such as flock, convoy and swarm patterns [1]. Trajectory pattern mining is designed to find frequent or regular movement patterns and is not designed to detect contacts.

4) TRAJECTORY MONITORING

Recently, few studies [25], [26] have been proposed to monitor trajectories to identify asymptomatic patients and to find potential zones for daily activities and movement dynamics. For a given target trajectory, [25] attempts to identify dense areas where potential interaction activities occur. The approach was tested with various clustering algorithms but it was for a single trajectory and not general to be applied to multiple ToI. Another study was conducted to identify potential endemic zones monitoring asymptomatic patient's movements by identifying Points of Interest (PoI) using spatio-temporal trajectories [25]. This study was further extended to a continuous monitoring of asymptomatic patients [26]. Even though these studies focus on potential patient's interactions, these approaches are limited to the stay point detection, and cannot be used to mine general contacts.

In another study [27], mining activity chains of individuals was proposed by identifying stops in spatio-temporal data. This is an extension to pattern mining discussed in Section II-B3 but not directly related to contact mining. Another study investigated movement dynamics in urban areas using inflows and outflows of trajectories [28]. This study focuses on flow traces revealing monocentric flow patterns and changes of functionalities which are different from contact mining. A Privacy Protection Technique (PPT) developed for COVID-19 pandemic [29] investigated on safeguarding individual's privacy in order to protect the intended uses of personal data. Another investigation conducted on modelling travel behaviour [8] is to mine the similarity amongst trajectories based on their activities.

In general, these trajectory monitoring approaches are designed to monitor trajectories based on clusters or stay points to find behavioural dynamics and interactions for a certain target trajectory, and they are not designed to mine contacts for multiple ToI.

C. COLLISION DETECTION

Collision detection is to detect the intersection of geometric models when objects are static as well as moving [30]. This is used in areas such as computer graphics, manufacturing, automation, robotics, computer animation and computer simulated environments [31]. There are many collision detection algorithms available which can be categorised into two phases such as the broad phase followed by the narrow phase [32]. To optimise the speed, broad phase algorithms are initially used to identify objects that can potentially collide and exclude objects that are not colliding with certainty. Then only those objects with a possibility of colliding are used to find out which objects are colliding each other in the narrow phase. The two phases allow much more efficient collision detection than using one phase [32]. The separation of these two phases was introduced by Hubbard [33] and followed by others. Collision detection methods are designed to detect two-dimensional and three-dimensional objects and cannot be used with spatio-temporal trajectories. Also, the scalability of these algorithms is questionable hence it is designed to handle small datasets.

In summary, none of these techniques can be utilised to find contacts for multiple ToI. Even though the contact mining technique [25] could be extended to identify multiple ToI, it becomes inefficient in handling multiple ToI. A comparative literature review table is given in Table 1.

TABLE 1. Comparison of literature review.

Technique	Patterns	ST Data	Contact Mining	Multiple ToI
Clustering	Similar groups	Yes	No	No
Classification	Prediction model	Yes	No	No
Pattern mining	Sequential/ periodic	Yes	No	No
Trajectory monitoring	Stay points/ PoI	Yes	Yes/ No	No
Collision detection	None	No	No	No

III. DEFINITIONS OF MULTIPLE TOI CONTACT MINING

Definition 1 (Spatio-Temporal Trajectory): A spatio-temporal trajectory (T_a) in a given spatio-temporal trajectory database $T = [T_a, T_b, \ldots, T_n]$ is a list of trajectory nodes representing longitude, latitude and corresponding timestamp, denoted by $T_a = \{(x_{a1}, y_{a1}, t_{a1}), (x_{a2}, y_{a2}, t_{a2}), \ldots, (x_{an}, y_{an}, t_{an})$, where $x_{ai}, y_{ai} \in \mathrm{R}^2$ and $t_{ai} \in \mathrm{R}^+$ for $i = \{1, 2, \ldots, n\}$ and $t_{a1} < t_{a2} < \ldots < t_{an}$.

Definition 2 (Trajectory of Interest): ToI is a user specified subset of the spatio-temporal trajectories ($\subseteq T$).

Definition 3 (Other Trajectories): OT are the remaining trajectories in T other than the ToI. That is, OT $= T \backslash$ ToI.

Definition 4 (Spatial s-Neighbourhood): The spatial s-neighbourhood of a trajectory node $n \in T_a$ for a given trajectory $T_i \in (\mathrm{T} \backslash T_a)$, denoted by $N_s^{Ti}(n)$, is defined by $N_s^{Ti}(n) = \{n_j \in T_i \mid dist(n_j, n) \leq s\}$, where $dist(.,.)$ is a distance function, but it is the Euclidean distance by default in this paper.

Definition 5 (Temporal t-Neighbourhood): The temporal t-neighbourhood of a trajectory node $n \in T_a$ for a given trajectory $T_i \in (\mathrm{T} \backslash T_a)$, denoted by $N_t^{Ti}(n)$, is defined by $N_t^{Ti}(n) = \{n_j \in T_i \mid diff(n_j, n) \leq s\}$, where $diff(.,.)$ is a time difference function, that measures the difference between the two timestamps.

Definition 6 (Spatio-Temporal st-Neighbourhood): The spatio-temporal st-neighborhood of a trajectory node $n \in T_a$ for a given trajectory $T_i \in (\mathrm{T} \backslash T_a)$, denoted by $N_{st}^{Ti}(n)$, satisfies both Definition 4 and Definition 5.

Definition 7 (Contact Duration d-Neighbourhood): Let N be a set $\{n_i, n_{i+1}, \ldots, n_{i+k}\}$ (where $i, k \in \mathrm{R}^+$) of consecutive nodes in a trajectory $T_i \in (\mathrm{T} \backslash T_a)$. The contact duration d-neighbourhood of a trajectory node $n \in T_a$ for a given trajectory T_i, denoted by $N_d^{Ti}(n)$, is defined by $N_d^{Ti}(n) = \{N \mid diff(n_i, n_{i+k}) \leq d\}$.

Definition 8 (Contact Detectable): A trajectory $T_i \in (\mathrm{T} \backslash T_a)$ is contact detectable by T_a *iff* $N_d^{Ti}(n)$ for a given d for a node $n \in T_a$ is not $\emptyset$.

Definition 9 (Multiple ToI Contact Mining From Spatio-Temporal Trajectories): For a given set of ToI ($Ts \subset T$), multiple ToI contact mining from a set $T = \{T_a, T_b, \ldots, T_n\}$ of spatio-temporal trajectories is to find all contact detectable trajectories from $T \backslash T_s$ (Definition 8).

IV. FRAMEWORK OF MULTIPLE TOI CONTACT MINING

A multi-step hierarchical contact mining framework is proposed to identify contacts using multiple ToI as shown in Figure 1.

Primarily, different types of datasets are artificially generated and another dataset is downloaded to perform a diverse set of experiments. First, these datasets are pre-processed to identify and rectify the inaccuracies of the raw spatio-temporal trajectory data. Subsequently by employing the user defined attributes, different types of approaches are used to identify contacts in various situations. A brute-force approach is initially applied to identify ground truth contacts which will be used as the baseline to compare the accuracy of other approaches. Then several types of clustering approaches are applied to obtain clustering results that are utilised to identify contacts. Thereafter MBR-s approach (an extension of single ToI contact mining [25]) is used to identify contacts, serially going through each ToI for contact mining. These algorithms will be used as baselines to evaluate the comparative performance analysis of our proposed approaches with respect to accuracy, efficiency, scalability, parameter sensitivity and applicability.

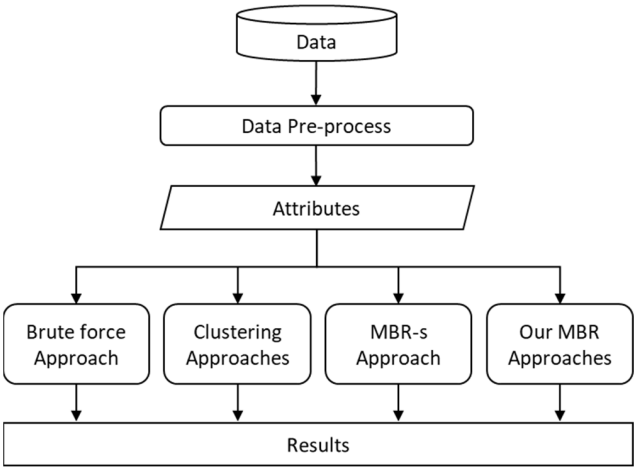

FIGURE 1. Multi-step hierarchical multiple ToI contact mining framework.

A. DATA GATHERING

A spatio-temporal trajectory consists of a series of nodes where each node is denoted by trajectory *id*, latitude, longitude and a corresponding timestamp. Three datasets were artificially generated with varying sizes and different complexities to explore the wide spectrum of trajectories, and one real-world dataset is downloaded from the Web in order to carry out our experiments.

Generated dataset 1 (denoted by Gd1) is artificially generated composed of 100 trajectories (among them 20 as ToI) with varying number of nodes in each trajectory. This dataset is further subdivided into four sub-datasets having 500, 1000, 2500 and 5000 nodes per each trajectory to carry out dedicated experiments. This dataset is used to compare the accuracy of each approach against the ground truth obtained by the brute-force approach. This dataset is also used to compare the efficiency against the brute-force, clustering, MBR-s and MBR-m (MBR based multiple ToI) approaches. Please note that this is a relatively small dataset as the brute-force and clustering approaches consume a considerable amount of processing time, it becomes infeasible to run these traditional approaches with large spatio-temporal trajectories.

Generated dataset 2 (denoted by Gd2) is artificially generated consisting of 100 trajectories and having 20 ToI. This dataset is then subdivided into four sub-datasets having 1000, 2000, 5000 and 10000 nodes per each trajectory to

compare the efficiency amongst MBR-m and our proposed approaches. Since these approaches are more efficient, relatively large dataset is generated. This dataset is also utilised to perform parameter sensitivity experiments.

Generated dataset 3 (denoted by Gd3) is artificially generated composed of varying trajectories having 1000 nodes per trajectory. Several experiments were conducted to analyse the efficiency of each approach having 100, 200, 500 and 1000 trajectories where 20 trajectories are used as ToI. These experiments are performed to see how efficiency varies with an increasing number of trajectories.

Generated dataset 4 (denoted by Gd4) is similar to Gd2 but having 30 ToI. These datasets are used to perform experiments to observe the efficiency variation with regards to the number of ToI.

Generated dataset 5 (denoted by Gd5) is generated with 100 trajectories where 20 trajectories are used for ToI. This dataset is then subdivided into four sub-datasets having 10000, 20000, 50000 and 100000 nodes per each trajectory to compare the scalability of the approaches. This is a larger dataset compared to other datasets hence it requires to perform a scalability experiment.

Generated dataset 6 (denoted by Gd6) is generated with 10000 nodes per trajectory. Experiments were conducted to analyse the scalability of each approach having 100, 200, 500 and 1000 trajectories when 20 trajectories are used as ToI. These experiments are performed to see how scalability is affected by an increasing number of trajectories.

Downloaded dataset 1(denoted by Dd1) is downloaded from Microsoft Geolife and utilised to examine the applicability of the approaches with real data. This dataset contains 100 trajectories which include 20 ToIs. This dataset is subdivided into four sub-datasets having 1000, 2000, 5000 and 10000 nodes per each trajectory to analyse efficiency.

B. DATA PREPROCESSING

Raw spatio-temporal trajectory data has inaccuracies due to the nature of trajectory acquiring devices. These inaccuracies such as measurement inaccuracies and spatial uncertainties such as over-sampled complexity and under-sampled simplicity must be resolved prior to experiments. Initially, inconsistencies of different formats of data, due to various types of data acquiring devices are processed. Then, measurement inaccuracies are handled by identifying and removing inaccurate and incomplete data. Then spatial uncertainties are addressed by finding the linear movement of trajectories and correcting the sampling rates.

C. USER SPECIFIED ATTRIBUTES

User specified attributes are used to find the contacts in various types of situations. This is performed to illustrate our approaches are parameter insensitive and applicable to various applications. Users can define spatial s-neighbourhood threshold, temporal t-neighbourhood threshold and a contact duration d-neighbourhood threshold in order to identify contacts. For instance, in relation to contagious disease situations, to identify potentially affected humans from known infected humans, a user may define spatial s-neighbourhood threshold as maximum 2 meters (the effective range of the virus), temporal t-neighbourhood threshold as maximum 15 minutes (effective lifetime of the virus) and a duration d-neighbourhood as at least 5 seconds (minimum contagion duration). Users may define these attributes according to the type of contagious disease. Furthermore, in relation to criminal network activities, contacts of known criminals may be identified by defining spatial s-neighbourhood threshold as at most 2 meters (effective range of the meeting), temporal t-neighbourhood threshold as at most 5 seconds (arrival time), and a duration d-neighbourhood threshold as at least 1 minute (effective meeting time). Various types of experiments have been conducted in Section V-E to illustrate the sensitivity of parameters with different values.

D. MULTIPLE TOI CONTACT MINING APPROACHES

This section covers three multiple ToI contact mining approaches and four proposed new methods.

1) BRUTE-FORCE APPROACH

Initially, a naive brute-force approach was performed to identify ground-truth contacts. All nodes in multiple ToI are compared against all nodes in OT to find true positive contacts. Given a set $T = \{T_1, T_2, \ldots, T_n\}$ of spatio-temporal trajectories and ToI $T_a \notin T$, this approach requires $|T_a|$ x $|T_1|$ x $|T_2|$ x...x $|T_n|$ operations. Even though this is a time-consuming operation, this approach is required as there are no datasets with true ground-truth contacts available.

2) CLUSTERING METHODS

As clustering is to identify a set of nodes in trajectories exhibiting similar spatio-temporal similarities, clustering could be a potential approach to detect contacts. This clustering prunes the search space and focuses on areas of high densities, thus it is a solid candidate to improve efficiency. Different categories of clustering methods were discussed in the literature and four clustering methods from most suitable for trajectories have been utilised in this paper. This includes DBSCAN and OPTICS clustering from the density based category, k-Means clustering from the distance based category, and BIRCH clustering from the hierarchical based category. The choice is intentionally to cover the wide spectrum of clustering categories and also to investigate which category of clustering methods is more suitable for multiple ToI contact mining for spatio-temporal trajectories. As clustering needs to be undertaken prior to contact mining, an overhead is involved in this approach.

3) MBR-s APPROACH

MBR-s approach utilises a MBR which is computed using the spatial s-neighbourhood threshold for each node of trajectory in ToI. Once computed, each node of OT is compared against to see if each node in OT is within the MBR which indicates

st-neighborhood. In multiple ToI, we need to iterate each trajectory in ToI to find all multiple ToI contacts. This is an extension of single ToI contact mining [25].

4) MULTIPLE ToI CONTACT MINING APPROACHES

Four approaches are proposed in this paper to find multiple ToI contacts.

a: APPROACH 1 – MBR-m

A straightforward extension of MRB-s is to consider all MBR of ToI at the same time to efficiently process contact mining. Similar to MBR-s, this approach creates a MBR initially for all nodes of each trajectory in ToI. Then nodes in OT are compared to see whether they are within MBRs of all trajectories in ToI. Instead of going through each node of each trajectory in ToI, this approach compares all nodes of all trajectories in ToI.

b: APPROACH 2 – MBR-mm

Initially the minimum and maximum of latitude, longitude and time-stamp of all nodes in ToI are obtained. Then each node of OT is compared against to see if it is within these boundaries (the minimum and maximum) as an initial pruning phase. Thereafter only those nodes within the boundaries are compared to see whether it is within MBRs. Hence this approach uses only the nodes within the minimum and maximum boundary, it is named as MBR-mm. Hence all nodes in OT are not compared against all MBRs this approach will be more efficient than MBR-m approach.

c: APPROACH 3 – MBR-sn

In this approach, initially each node of OT is compared to find the nearest node of ToI and this distance is obtained. Thereafter using the average timestamp interval, the number of nodes to reach this distance is calculated. Hence this is the shortest path to the node in ToI, the result can be used to skip the number of nodes in OT and hence to further prune the search space. The details of this approach are described in Algorithm 1.

Algorithm 1 Find_Contacts_MBRsn

```
Input:
dsTOI :Trajectories of interest dataset;
dsOT :Other trajectories dataset;
dsAttributes: User provided attributes;
Output
dsContactsFound: Contacts found;
function Find_Contacts_MBRsn(dsTOI, dsOT, dsAttributes)
    Create an empty list dsContactsFound;
    Assign time taken move to next node to timeNode;
    while not end of dsOT
        Assign to dsOT[id] to id;
        while dsOT[id] = id
            Find the nearest node of dsTOI;
            Assign distance to nearest node to distanceToNode;
            Assign distanceToNode / timeNode to nodesToSkip;
            if nodesToSkip <= 1
                Find contact using dsAttributes;
                if contact found
                    Append dsOT[id] to dsContactsFound;
                    Skip to next id
                    Exit while loop
                else
                    Skip to next node
                end if
            else
                Skip nodesToSkip;
            end if
        end while
    end while
    return dsContactsFound;
end function
```

d: APPROACH 4 – MBR-ms

Both MBR-mm (Approach 2) and MBR-sn (Approach 3) are combined to further reduce the search space, as this approach inherits the benefits from both approaches, it will be the most efficient approach.

V. EXPERIMENTAL RESULTS

A. COMPUTER SPECIFICATIONS

A workstation with an Intel(R) Core (TM) i7-8750H @ 2.20 GHz processor and 20 GB unallocated memory is used to perform all experiments in this paper. Python programming language is utilised to implement all algorithms.

B. ACCURACY ANALYSIS

This experiment is conducted with all approaches to compare the accuracy amongst each of them which can be observed in Figure 2. Initially brute-force approach is conducted to identify ground-truth data, and the result is used as a baseline to compare the accuracy of other approaches. A relatively small dataset (Gd1) is used to perform this experiment since the brute-force approach consumes a considerable amount of processing time to find the ground-truth true positive contacts. The dataset consists of 100 trajectories, and four experiments were conducted with each trajectory having 500, 1000, 2500 and 5000 nodes per trajectory. Out of 100 trajectories, 20 trajectories are identified as ToI. This is conducted to observe how an accuracy level varies with the number of nodes in each trajectory. It is noted that in general clustering approaches are computationally fast at the expense of effectiveness exhibiting a varying level of accuracies. Density and hierarchical based clustering methods show reasonably accurate results while the distance based clustering method shows less accurate results. The reason behind this is the density based and hierarchical based clustering methods group nodes spatio-temporally close to each other whilst the distance based clustering method does not. Also, the distance based clustering may identify nodes in potential contact areas as outliers. One interesting point to note is that the clustering based approaches improve their accuracies in general as there are more nodes in trajectories. This is not surprising as there will be more nodes falling in the clustered areas (however

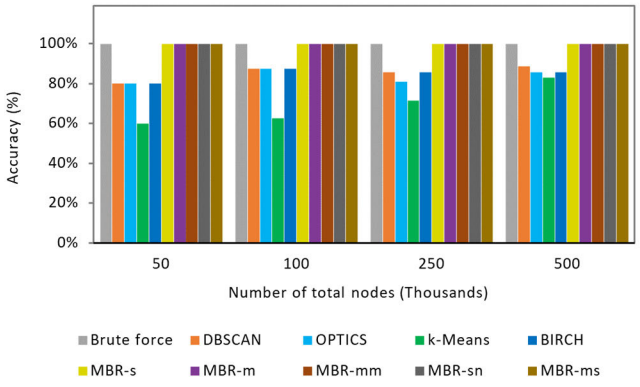

FIGURE 2. Comparison of approaches: accuracy percentage for 100 trajectories each having 500, 1000, 2500 and 5000 nodes.

it will be achieved at the expense of efficiency as there will be more nodes to check for possible contacts). In conclusion, MBR-s and our four proposed approaches are able to identify all true positive ground-truth contacts.

C. EFFICIENCY ANALYSIS

This experiment is conducted utilising the same dataset (Gd1), used in previous experiment to analyse the efficiency of each approach. Initially clustering methods, MBR-s and MBR-m methods are compared against the brute-force baseline approach as shown in Figure 3. It is observed that certain clustering approaches consume more processing time than the baseline approach. This is due to the additional requirement of processing time for clustering. This clearly demonstrates that clustering cannot be directly applied to contact mining as it fails to detect all ground-truth contacts and also even its extra time to compute clusters places an additional computational burden to contact mining. Particularly, the density based approaches (DBSCAN and OPTICS) suffer from inefficiencies even though they perform better in accuracy than the distance based clustering approach. Even

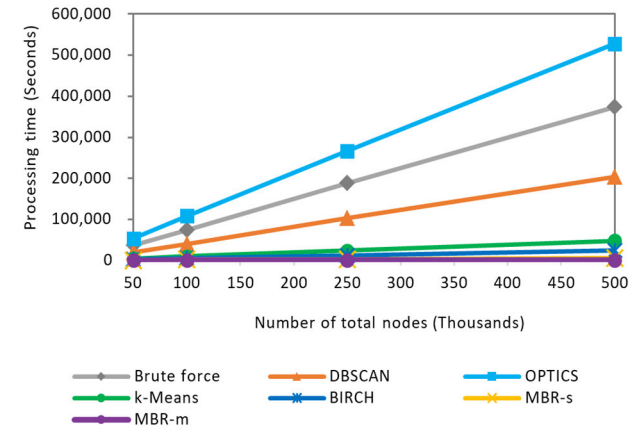

FIGURE 3. Comparison of approaches: processing time in seconds for 100 trajectories having 500, 1000, 2500 and 5000 nodes in each trajectory.

though the distance based clustering performs the worst in effectiveness among other approaches, it performs better in efficiency than other clustering approaches. Undoubtedly, MBR-s and MBR-m approaches demonstrate promising results as shown in Figure 3. Hence processing times of these two approaches are not noticeable in Figure 3, it is shown separately in Figure 4.

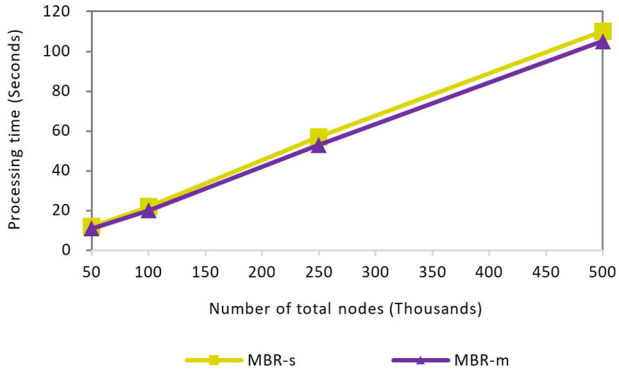

FIGURE 4. Comparison of MBR-s and MBR-m approaches: processing time in seconds for 100 trajectories having 500, 1000, 2500 and 5000 nodes in each trajectory.

A relatively large dataset (Gd2) is used to perform the efficiency analysis of proposed four approaches. This dataset has 100 trajectories with 20 ToI, and four experiments are conducted with each trajectory having 1000, 2000, 5000 and 10000 nodes. The most efficient MBR-m approach in previous experiment is utilised here as the baseline to compare against our three other approaches. The result is shown in Figure 5.

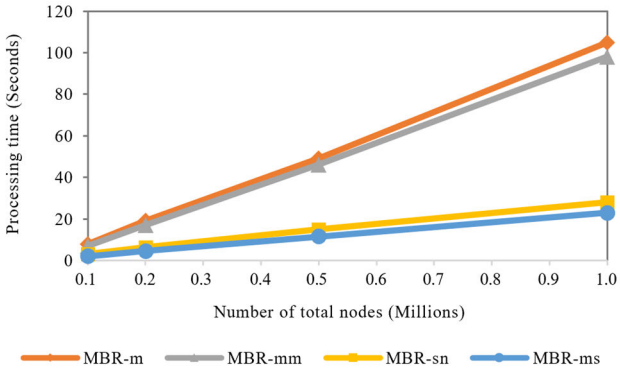

FIGURE 5. Comparison of MBR-m with proposed approaches: processing time in seconds for 100 trajectories with 20 ToI and 1000, 2500, 5000 and 10000 nodes in each trajectory.

It is observed that MBR-mm is slightly more efficient than the baseline approach MBR-m. But approach MBR-sn is far more efficient than MBR-mm. The combination (MRB-ms) of approaches MBR-mm and MBR-sn even further improves the efficiency of MBR-sn. The efficiency of approach MBR-mm is dependent on how dense the nodes are. When the nodes of trajectories are spread out in relation to the

geographical study region and the time span, the efficiency is higher and vice versa. However, this dependency is lessened with MBR-ms as it consistently performs better than all other approaches in various settings.

The experiment shown in Figure 6 is conducted to investigate how efficiency varies with the number of trajectories. Dataset (Gd3) is used with a fixed number of nodes per trajectory, which is 1000 with an increasing number of trajectories 100, 200, 500, 1000 to perform this experiment.

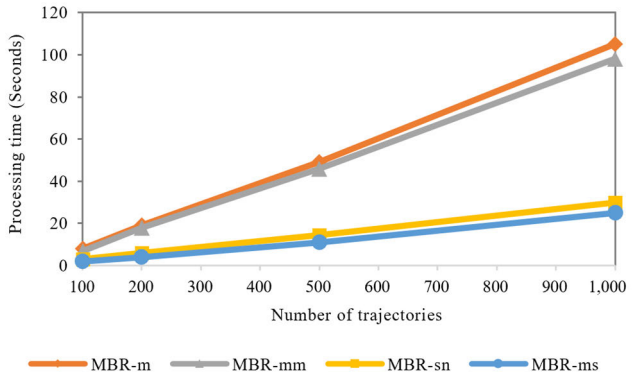

FIGURE 6. Comparison of MBR-m with our approaches: processing time in seconds for 100 trajectories with 20 ToI and 10000, 20000, 50000 and 100000 nodes in each trajectory.

Please note that both experiments shown in Figure 5 and Figure 6 have the same number of total nodes (with different arrangements), we can induce some findings in both results. It is observed that efficiencies of all approaches in Figure 6 (varying the number of trajectories) are slightly better than those shown in Figure 5 (varying the number of nodes per trajectory). This suggests that our proposed algorithms are more robust in efficiency with a growing number of trajectories than a growing number of nodes per trajectory.

All experiments above were conducted with 20 ToI out of 100 trajectories. The following experiment shown in Figure 7 is performed with dataset (Gd4) having 30 ToI. It is observed that efficiency in all the approaches is slightly lower than

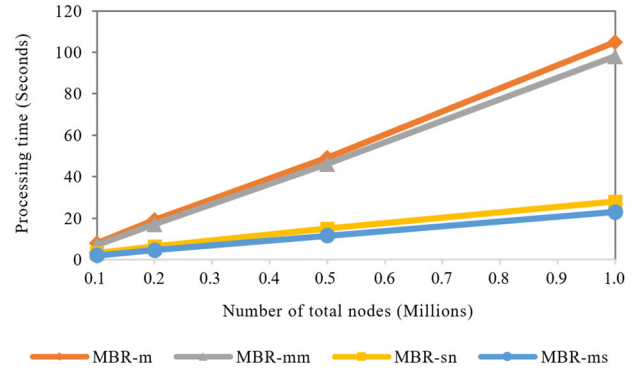

FIGURE 7. Comparison of MBR-m with proposed approaches: processing time in seconds for 1000 nodes per trajectory with 20 ToI and 100, 200, 500, 1000 trajectories.

the previous experiment. This is due to the fact that when there are more nodes in ToI, the requirement of comparison amongst each node is more.

D. SCALABILITY ANALYSIS

This experiment is conducted to demonstrate the scalability of our approaches with a comparatively larger dataset (Gd5). The dataset consists of 100 trajectories with 20 trajectories as ToI and having 10000, 20000, 50000 and 100000 nodes per trajectory. All approaches as seen in Figure 8 display a linear growth indicating that the scalability of proposed approaches.

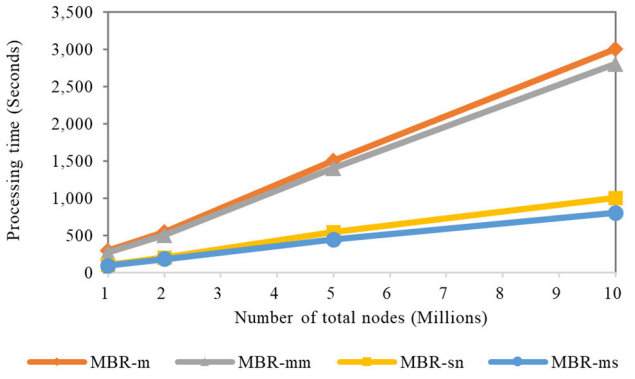

FIGURE 8. Comparison of MBR-m with our approaches: processing time in seconds for 100 trajectories with 30 as ToI having 1000, 2500, 5000 and 10000 nodes in each trajectory.

Another experiment was conducted using the dataset Gd6 having 10000 nodes per trajectory with 100, 200, 500 and 1000 trajectories as shown in Figure 9. This was performed to see how an increasing number of trajectories effects the scalability. This also exhibits a similar trend as shown in Figure 8 illustrating the scalability of proposed approaches.

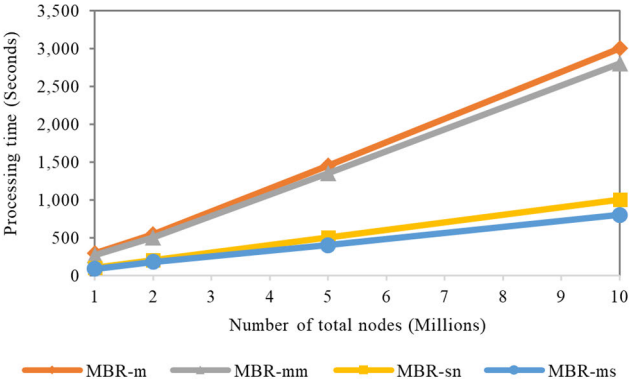

FIGURE 9. Comparison of MBR-m with our approaches: processing time in seconds for 10000 nodes per trajectory with 20 ToI and having 100, 200, 500, 1000 trajectories.

E. PARAMETER SENSITIVITY ANALYSIS

This experiment is conducted to demonstrate the parameter sensitivity of proposed approaches. Users can have different parameter values for various applications. Dataset Gd2 is

used here with 100 trajectories where 20 ToI and 1000 nodes per trajectory used. Also, different parameter values are used to explore the parameter sensitivity analysis. It is observed in Figure 10 that all approaches have a similar efficiency trend with varying parameter values hence it can be concluded that our proposed approaches are insensitive to parameter values, and robust to various parameter settings and real-world scenarios.

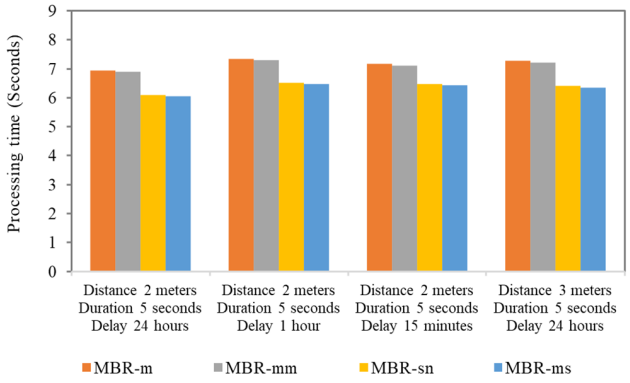

FIGURE 10. Comparison of MBR-m with our approaches: processing time in seconds for 100 trajectories with 10 ToI and each having 10000 nodes.

F. APPLICABILITY ANALYSIS

A downloaded dataset (Dd1) is used to experiment the applicability of our approaches. This was conducted using 100 trajectories and a varying number of nodes per trajectory ranging from 1000, 2000, 5000 and 10000.

As shown in Figure 11, it is observed that efficiency results are similar to previous experiments exhibiting an outperforming trend, and also note that these approaches are able to detect all true positive contacts from dataset Dd1 that can be found by the brute-force approach as ground-truth contacts. This indicates that these approaches are applicable to real-world data.

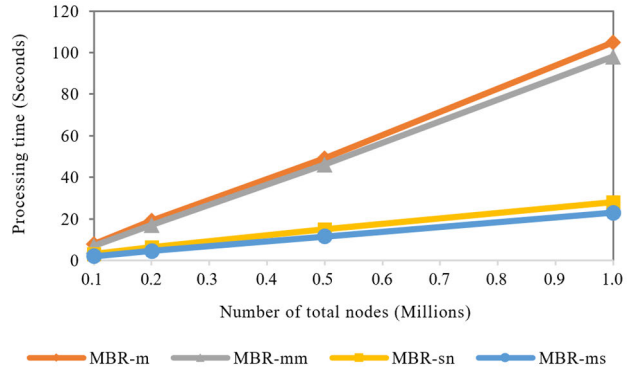

FIGURE 11. Comparison of MBR-m with our approaches: processing time in seconds for 100 trajectories with 20 ToI and 1000, 2500, 5000 and 10000 nodes in each trajectory.

VI. CONCLUSION

Contact data mining is an interesting topic as it investigates potential contacts involving interactions with others. It could be used to identify suspicious interactions in criminal networks and to find potential interactions in pandemic disease situations. In many situations, we are required to find contacts from multiple ToI as there would be more than one criminal in the criminal network analysis and one patient in the epidemic disease analysis.

This paper introduces this new multiple ToI contact mining and proposes a number of approaches that are able to identify all true positive ground-truth contacts. First, this paper designs and implements clustering based approaches, extends the single ToI contact mining, and proposes several MBR based approaches. A various set of experiments demonstrates that our proposed multiple ToI contact mining approaches are able to detect all ground-truth contacts, more efficient than clustering based and extended single ToI approaches, and also robust and insensitive to various parameter settings demonstrating the efficiency, effectiveness, scalability and applicability to various real-world settings and scenarios.

Future directions are in two folds. First, an investigation into multi-level contact mining is of interest as it is evident in real-world where a criminal is contacted by potential people who will be in contact with other people in a later stage. Second, as there is no known dataset with ground-truth contacts available. A generation of datasets with ground-truth contacts is an area to explore.

REFERENCES

[1] Y. Zheng, "Trajectory data mining: An overview," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 3, pp. 1–41, 2015.

[2] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle, *GNSS—Global Navigation Satellite Systems: GPS, GLONASS, Galileo, and More*. Vienna, Austria: Springer, 2007.

[3] I. Ardakani, K. Hashimoto, and K. Yoda, "Understanding animal behavior using their trajectories: A case study of gender specific trajectory trends," in *Proc. 6th Int. Conf. Distrib., Ambient Pervasives Interact., Technol. Contexts (DAPI)*, Las Vegas, NV, USA, 2018, pp. 3–22.

[4] Z. Duan, L. Tang, X. Gong, and Y. Zhu, "Personalized service recommendations for travel using trajectory pattern discovery," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 3, Mar. 2018, Art. no. 155014771876784.

[5] Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, "TrafficPredict: Trajectory prediction for heterogeneous traffic-agents," in *Proc. AAAI Conf.*, 2019, pp. 6120–6127.

[6] A. K. Miltenberger, S. Pfahl, and H. Wernli, "An online trajectory module (version 1.0) for the nonhydrostatic numerical weather prediction model COSMO," *Geosci. Model Develop.*, vol. 6, no. 6, pp. 1989–2004, Nov. 2013.

[7] B. Qu, W. Yang, G. Cui, and X. Wang, "Profitable taxi travel route recommendation based on big taxi trajectory data," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 653–668, Feb. 2020.

[8] W. Yang, Y. Zhao, B. Zheng, G. Liu, and K. Zheng, "Modeling travel behavior similarity with trajectory embedding," in *Proc. 23rd Int. Conf. DASFAA*, Gold Coast, QLD, Australia, May 2018, pp. 630–646.

[9] P. K. Enge, "The global positioning system: Signals, measurements, and performance," *Int. J. Wireless Inf. Netw.*, vol. 1, no. 2, pp. 83–105, Apr. 1994.

[10] P. J. G. Teunissen and A. Khodabandeh, "Review and principles of PPP-RTK methods," *J. Geodesy*, vol. 89, no. 3, pp. 217–240, Mar. 2015.

[11] M. Liu, G. He, and Y. Long, "A semantics-based trajectory segmentation simplification method," *J. Geovis. Spatial Anal.*, vol. 5, no. 2, pp. 1–15, Dec. 2021.

[12] S. Wang, Z. Bao, J. S. Culpepper, and G. Cong, "A survey on trajectory data management, analytics, and learning," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–36, Mar. 2022.

[13] L. Bermingham and I. Lee, "A general methodology for n-dimensional trajectory clustering," *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7573–7581, Nov. 2015.
[14] J.-G. Lee, J. Han, and K.-Y. Whang, "Trajectory clustering: A partition-and-group framework," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2007, pp. 593–604.
[15] D. Zhang, K. Lee, and I. Lee, "Hierarchical trajectory clustering for spatio-temporal periodic pattern mining," *Expert Syst. Appl.*, vol. 92, pp. 1–11, Feb. 2018.
[16] P. Bhattacharjee and P. Mitra, "A survey of density based clustering algorithms," *Frontiers Comput. Sci.*, vol. 15, no. 1, Feb. 2021, Art. no. 151308.
[17] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: An overview, II," *WIREs Data Mining Knowl. Discovery*, vol. 7, no. 6, Nov. 2017, Art. no. e1219.
[18] P. D. Mcnicholas, "Model-based clustering," *J. Classification*, vol. 33, pp. 331–373, Nov. 2016.
[19] C. L. da Silva, L. M. Petry, and V. Bogorny, "A survey and comparison of trajectory classification methods," in *Proc. 8th Brazilian Conf. Intell. Syst. (BRACIS)*, Salvador, Brazil, Oct. 2019, pp. 788–793.
[20] D. Patel, C. Sheng, W. Hsu, and M. L. Lee, "Incorporating duration information for trajectory classification," in *Proc. IEEE 28th Int. Conf. Data Eng.*, Arlington, VA, USA, Apr. 2012, pp. 1132–1143.
[21] F. Giannotti, M. Nanni, F. Pinelli, and D. Pedreschi, "Trajectory pattern mining," in *Proc. 13th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Jose, CA, USA, Aug. 2007, pp. 330–339.
[22] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye, "Mining periodic behaviors for moving objects," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2010, pp. 1099–1108.
[23] C. Körner, M. May, and S. Wrobel, "Spatiotemporal modeling and analysis—Introduction and overview," *Künstliche Intelligenz*, vol. 26, no. 3, pp. 215–221, Aug. 2012.
[24] H. Cao, N. Mamoulis, and D. W. Cheung, "Mining frequent spatio-temporal sequential patterns," in *Proc. 5th IEEE Int. Conf. Data Mining (ICDM)*, Houston, TX, USA, Nov. 2005, p. 8.
[25] D. Adu-Gyamfi and F. Zhang, "Mobility and trajectory-based technique for monitoring asymptomatic patients," *J. Inf. Technol. Res.*, vol. 15, no. 1, pp. 1–18, Nov. 2021.
[26] D. Adu-Gyamfi, F. Zhang, and A. K. K. Ansah, "EDDAMAP: Efficient data-dependent approach for monitoring asymptomatic patient," *BMC Med. Informat. Decis. Making*, vol. 20, no. 1, pp. 1–16, Dec. 2020.
[27] L. Yin, N. Lin, and Z. Zhao, "Mining daily activity chains from large-scale mobile phone location data," *Cities*, vol. 109, Feb. 2021, Art. no. 103013.
[28] X. Xing, Y. Yuan, Z. Huang, X. Peng, P. Zhao, and Y. Liu, "Flow trace: A novel representation of intra-urban movement dynamics," *Comput., Environ. Urban Syst.*, vol. 96, Sep. 2022, Art. no. 101832.
[29] A. Majeed and S. O. Hwang, "A comprehensive analysis of privacy protection techniques developed for COVID-19 pandemic," *IEEE Access*, vol. 9, pp. 164159–164187, 2021.
[30] R. Weller, "A brief overview of collision detection," in *New Geometric Data Structures for Collision Detection and Haptics*. Heidelberg, Germany: Springer, 2013, pp. 9–46.
[31] S. Kockara, T. Halic, K. Iqbal, C. Bayrak, and R. Rowe, "Collision detection: A survey," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Montreal, QC, Canada, Oct. 2007, pp. 4046–4051.
[32] S. Kockara, T. Halic, C. Bayrak, K. Iqbal, and R. A. Rowe, "Contact detection algorithms," *J. Comput.*, vol. 4, no. 10, pp. 1053–1063, Oct. 2009.
[33] P. M. Hubbard, "Interactive collision detection," in *Proc. IEEE Res. Properties Virtual Reality Symp.*, San Jose, CA, USA, Oct. 1993, pp. 24–31.

ADIKARIGE RANDIL SANJEEWA MADANAYAKE received the bachelor's degree in IT degree (Hons.) from James Cook University, Townsville, QLD, Australia, where he is currently pursuing the Ph.D. degree with the Information Technology Discipline, College of Science & Engineering, under the supervision of Dr. Joanne Lee and Prof. Ickjai Lee. His research interests include criminal network analysis, contact mining, trajectory data mining, spatio-temporal analysis and mining, and applied artificial intelligence.

KYUNGMI LEE (Member, IEEE) received the Ph.D. degree in computer science from Griffith University, Australia, in 2007. She started her academic career as a Lecturer with the School of Business and Information Technology, Charles Sturt University, Australia, and continued her academic pursuit after moving to James Cook University, QLD, Australia. She is currently an Active Researcher. Her research interests include machine learning, algorithm optimization, neural networks, data mining, and applied artificial intelligence. She has been involved in various projects developing a real-world scheduling optimization system for fly-in-fly-out mining employee scheduling, designing and implementing a non-destructive neural network-based classification for ultrasonic signals, and developing spatio-temporal mining algorithms for moving objects. Recently, she participated in various industry projects, including automatic detection of measuring of abalones and automation of marine monitoring.

ICKJAI LEE (Member, IEEE) is currently a Professor with the Information Technology Discipline, James Cook University, QLD, Australia. He is also actively involved in teaching and research and the Head of the Information Technology Discipline. He has published more than 150 articles in academic forums and has been involved in a number of research projects. His research interests include data mining, geospatial databases, space tessellations, geo-visualization, sentiment analysis, deep learning, and trajectory data mining. He has served as a program committee member for various conferences and an Associate Editor for *Machine Learning with Applications*.