BIG TRAJECTORY DATA: A DISTRIBUTED COMPUTING PERSPECTIVE

B. Anbaroğlu [1], Y. Alter [2,3]
[1] Dept. of Geomatics Engineering, Hacettepe University, Ankara, Turkey – banbar@hacettepe.edu.tr
[2] Graduate School of Science and Engineering, Hacettepe University, Ankara, Turkey – yavuzalter@hacettepe.edu.tr
[3] Aselsan Inc., Ankara, Turkey – yalter@aselsan.com.tr

Commission IV, WG IV/I

KEY WORDS: Distributed computing, big spatial data, cloud computing, trajectory, 3V

ABSTRACT:

Trajectory data constitute location of objects at specified time intervals. The continuous availability of GNSS signals, or discrete availability of sensor systems such as license plate recognition cameras are used to generate trajectory data. Consequently, in a smart city context, big trajectory data are being generated on a daily basis. The analysis of big trajectory data entails the use of a distributed environment to conduct analysis, and at least two data sources. The literature review conducted in this paper shows that the two *V*s of big data, *Volume* and *Variety*, may not be satisfied since researchers usually rely on a centralised computing environment, and analyse data coming from a single data source. Out of the 17 papers published from 2020 in Scopus, only five of them relied on a distributed computing environment, and two of them utilised more than one data source.

1.INTRODUCTION

The ubiquitous use of sensors coupled with the development in information and communication technologies contribute to the explosive growth of geospatial data. Geospatial data is usually considered as remotely sensed data, since petabytes of images have been collected and managed with the launch of various satellites including Landsat and Sentinel. These images could be processed, relatively easily, by relying on services such as Google Earth Engine or Amazon NASA NEX for various purposes ranging from estimating crop yields to cultural heritage management (Agapiou 2017; Warren et al. 2015). Geospatial data in a smart city context; however, is often collected as vector data, and more specifically trajectory data.

Zheng (2015) defines a spatial trajectory as '*a trace generated by a moving object in geographical spaces, usually represented by a series of chronologically ordered points*'. The moving object can be both living, such as an animal or a person, or non-living, such as a bus or a ship. Kong et al. (2018) provide a thorough review on various use-cases and research areas that rely on trajectory data ranging from detecting rapid accelerating vehicles to address fuel consumptions and carbon emissions, to the location selection of electric charging stations in order to maximise vehicle-miles travelled. They also classify trajectory data into two as i) *explicit*, and ii) *implicit*. Explicit trajectory data relies on continuous receival of GNSS signals to record the location and time of an object. Consequently, data are collected at regular time intervals. Implicit trajectory data, on the other hand, cannot collect location data at regular time intervals, since sensors such as Automatic Number Plate Recognition (ANPR) cameras, Wi-Fi probes or cellular network are used to collect location data. Therefore, locational information relies on the spatial distribution of these sensors. Furthermore, the activity of the object, such as making a phone-call or passing through an ANPR camera, defines the data collection interval. Consequently, implicit data are collected at irregular time intervals.

Zheng (2015) provided a thorough coverage on trajectory data mining by emphasizing various interrelated topics on data preprocessing and management, data transformation and provided exemplar datasets, most of which can be associated within an urban context. It is reasonable to envision a scenario where buses / taxis equipped with GNSS receivers collect large amounts of time-stamped location information in a smart city. For example, Taxi & Limousine Commission (TLC) of New York City have been collecting and distributing the origin-destination of taxi trips since 2009 (https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page). This valuable source of data has been utilised in various research areas ranging from taxi demand prediction (Xu et al. 2018) to developing effective ways to realise taxi ridesharing (Barann, Beverungen, and Müller 2017). Consequently, collected data are often used by public transportation authorities to improve transportation services as well as by passengers to ease their commute or reduce costs. Even though, origin-destination data constitute only a subset of trajectory data, some researchers still consider it as *Big Data* as millions of taxi trips usually occur within metropolitan cities on a monthly basis (Anbaroğlu 2021; Zhu et al. 2016).

Big Data has been defined in different contexts, but this paper relies on the definition provided by Khan, Uddin, and Gupta (2014), which is as follows: '*according to many researchers and writers, big data is a form of data that exceeds the processing capabilities of traditional database infrastructure or engines'*. The emphasis here is on the lack of ability of traditional database infrastructure to handle big data. More precisely, a *distributed computing environment* is required to manage and store big data. The necessity to scale out computations from single high-performance servers to multiple low-cost commodity-servers (tens to thousands) has led to the development of effective technologies like Apache Spark (Zaharia et al. 2016), YARN (Vavilapalli et al. 2013) or Apache Tez (Saha et al. 2015), most of which are licensed under the Apache Software Foundation. Although these technologies have been effective to employ analytics on textual / numeric data, their utilisation on spatial data, and more specifically, trajectory data have remained limited. Those that focused on trajectory data have overlooked the progress in distributed-computing environments (Kong et al. 2018; D. Wang, Miwa, and Morikawa 2020).

The aim of this paper is to review the literature on the use of distributed technologies to handle big trajectory data. The organisation of this paper is as follows. Second section explains the spatial indexing methods, and current technological tools and technologies to handle big geospatial data. Third section provides the results obtained from the literature review on keyword search *big trajectory data* in Scopus. Fourth section provides a Strengths-Weaknesses-Opportunities-Threats (SWOT) analysis on the utilisation of distributed computing technologies to manage big trajectory data. Finally, conclusions and future research directions are presented.

2. HANDLING BIG GEOSPATIAL DATA

This section first describes the spatial indexing methods, and then the cloud infrastructure that can support management of big trajectory data.

2.1 Spatial Indexing Methods

Spatial indexing is used to enable fast access to spatial data. A spatial index usually corresponds to a tree data structure to reduce the time of searching the query data. Traditional indexing methods such as B-trees fail to index spatial data; since it is not possible to order spatial data in which the number of dimensions is usually two or three. Most spatial database management systems comprise a spatial indexing method to increase the performance on spatial queries such as *k*-nearest neighbour or point-in-polygon. The spatial indexing can be primarily divided into two for different geographical features: i) point, and ii) line and polygon. Indexing point datasets is usually achieved by converting point coordinates into grids by Space Filling Curves (SFCs), which are used to organise points in 2D space into bounding-box hierarchies by preserving the spatial adjacency. It was initially proposed by Giuseppe Peano in 1890, and simplified and improved by David Hilbert in 1891 (Haverkort and van Walderveen 2010; Moon et al. 2001).

Usually a two-step procedure is carried out to realise a spatial query involving lines and polygons: i) filter and ii) refinement. In the filter step, geospatial entities such as lines or polygons are represented as their Minimum Bounding Rectangle (MBR), and in the second step refinement the true representation of the geographical data is used. The filter step would dramatically reduce the number of computations. Manolopoulos, Theodoridis, and Tsotras (2009) provided an excellent summary on the historical progress and various methods of spatial indexing including, but not limited to, R-trees, quadtree, and LSD tree. Many variants of the renowned R-tree have been proposed, but probably the one that is most related to a distributed computation architecture is the SD-Rtree (du Mouza, Litwin, and Rigaux 2007).

The theoretical foundations of indexing methods may not be clearly transferred to a database management system (DBMS). Specifically, the implementation details may be hidden from the users, and even if the code is open, it might be difficult to trace the code except for the developers of the DBMS themselves. For example, the spatial index used in PostGIS, the spatial extension of PostgreSQL –a renowned relational DBMS– is referred to as the generalised search tree (i.e. gist) and relies on R-tree. On the other hand, the spatial index used in MongoDB –a renowned NoSQL DBMS– is referred to as the 2Dsphere (https://www.mongodb.com/docs/manual/core/2dsphere/), but the data structure that it relies on has not been explicitly stated. Researchers have identified that the same query on the same dataset may return different outcomes on Postgres and MongoDB (Anbaroğlu 2021; Bartoszewski, Piorkowski, and Lupa 2019). Consequently, the differences in the ways in which a spatial indexing method has been implemented has crucial effect on the performance and effectiveness of spatial queries.

2.2 Cloud Infrastructure

Leading IT companies provide cloud services, where it is possible to store and analyse big trajectory data. The well-known of these services include Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure and AliBaba Cloud. These infrastructures have also been used in research, and in order to provide a proxy of their prevalence, Scopus is used to search the keywords in two different settings: i) using all fields, and ii) using only the article title, abstract and keywords. The search is conducted on 22 June 2022. The former is abbreviated as 'A', and the latter as 'T, A, K' in Table 1. The former search provides a prevalence of the cloud service in the published outcomes, whereas the latter provides research articles focusing on the specified cloud service. The second number in each cell denotes the same search while adding the keywords 'spatial' and 'trajectory' to the search.

	A	**T, A, K**
AliBaba Cloud	676 / 6	157 / 0
Amazon Web Services (AWS)	**6394** / 26	**1399** / 0
IBM Cloud	1273 / 3	167 / 0
Microsoft Azure	3987 / 21	942 / 0
Google Cloud	5190 / **52**	1047 / **2**

Table 1. Prevalence of the cloud platforms, and in their use with trajectory data in Scopus

The number of publications suggest that AWS is more common to research community. However, once the focus is on trajectory data analysis, apart from Google Cloud Platform (GCP), none of them was utilised. The two papers on trajectory analysis that relied on GCP are indeed relevant to this review. First, Ghosh and Ghosh (2019) developed a GCP-based trajectory management system. Although the proposed system's performance exceeds the centralised way to realise map-matching (i.e. matching GPS records to a road network), the experimental setup should be re-evaluated on larger datasets to understand the full-potential. Second, Jitkajornwanich et al. (2017) utilised high-frequency radar observations to predict the direction and velocity of sea current values. The high-frequency radar data can be considered as an implicit trajectory data, since observations are recorded at predefined grids rather than a continuous space. Moreover, the effectiveness of the use of GCP have not been investigated. To realise a global model, one might be interested in performance issues.

On the other hand, other researchers might setup their own distributed system, and conduct big trajectory analysis. For example, Maguerra et al. (2020) designed a reactive system on an Akka cluster. The designed system comprises high concurrency, responsiveness, and elasticity while handling exceptions efficiently. The technologies they have relied on include Play Framework, Nginx, AngularJS, D3.js and

https://doi.org/10.5194/isprs-archives-XLVIII-4-W3-2022-21-2022

MongoDB. The components of such systems may differ, and consequently the performance and effectiveness may change. Furthermore, user-interface design is another issue that have not been addressed in the paper. An interactive dashboard allows users interact and analyse data (Lwin et al. 2019; Oktay et al. 2021).

Geospatial big data framework solutions have also been developed that can be classified under three categories: i) GeoWave, ii) GeoMesa, and iii) GeoTrellis. GeoWave can manage different data stores such as Redis, Kafka, Cassandra or Accumulo, and visualise spatial data on top of GeoServer. GeoWave takes advantage of Hilbert space filling curve for dimensionality reduction, and to index spatial data. In this way, contiguity in single dimensional keys of a datastore would be preserved. Similarly, GeoMesa, allows querying and analytics on big geospatial data, and it provides indexing to various database management systems including Accumulo, HBase, Google Bigtable and Cassandra. Stream processing can also be realised via Apache Kafka and Apache Camel integration (Hughes et al. 2015, 2016). GeoMesa has been leveraged by Li et al. (2020) due to its capability to manage big spatio-temporal data over distributed NoSQL data stores. However, these are emerging technologies, and some issues may arise in their installation. For example, GeoWave could not be installed on all three operating systems as of 30 June 2022 as illustrated in Figure 1, probably due to an issue at the Amazon S3 environment in which the associated files are hosted. In our previous attempts, we were successful to install GeoWave, but then configuring the Maven environment proved to be difficult.

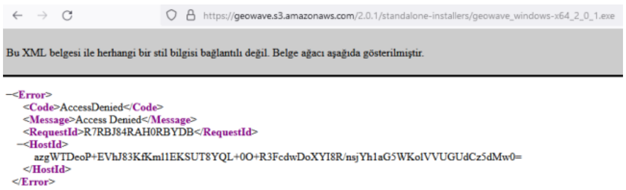

Figure 1. Installation error of GeoWave – 30 June 2022

Scopus is used to search these frameworks to understand their prevalence in research community, and the results are presented in Table 2.

	A	T, A, K
GeoMesa	**114**	15
GeoTrellis	74	3
GeoWave	71	**25**

Table 2. Prevalence of big geospatial data analysis frameworks in Scopus

The results suggest that GeoWave is more prevalent when titles, abstract and keywords are searched, whereas GeoMesa is more prevalent when the search is relaxed to the entire paper. Nevertheless, these frameworks are still at their infancy but have a great potential to facilitate big trajectory analysis.

3. SURVEY ON BIG TRAJECTORY DATA

This section describes the literature review conducted on Scopus by searching the phrase “big trajectory data” on 24 June 2022. The outcome of this inquiry revealed that the search phrase was present in 75 papers’ title, abstract or keyword. The number of papers published each year are illustrated in Figure 2.

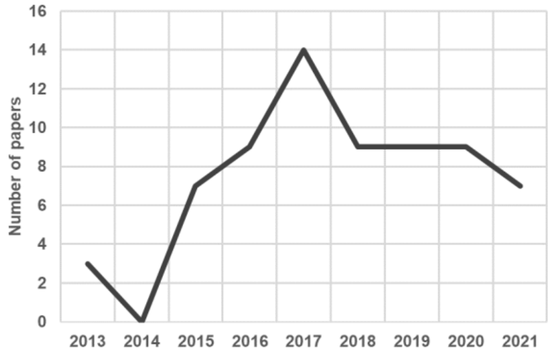

Figure 2. Number of papers appear on Scopus by searching ‘big trajectory data’

Due to space limitations, 24 papers published in 2020, and onwards, were collected and analysed. Three of those papers were written in Chinese, which are entitled: i) ‘*Fast and Distributed Map-Matching Based on Contraction Hierarchies*’, ii) ‘*Integrating Human Mobility into the Epidemiological Models of COVID-19: Progress and Challenges*’, and iii) ‘*kNN Query Processing for Trajectory Big Data Based on Distributed Column-Oriented Storage*’. These papers were not understood; hence, they were removed from the analysis. Out of the remaining 21 papers, two of them were review papers (Chekol and Fufa 2022; D. Wang et al. 2020), and there was no access to two papers (G. Wang et al. 2020; L.-W. Wang et al. 2022). The remaining 17 papers were analysed in detail, and the results are illustrated in Table 3. The table reveals information about the type of the trajectory (Implicit or Explicit), a brief purpose, whether a Centralised or Distributed system was used, whether a spatial index was utilised, and information about the 3Vs of Big Data: Volume (or data size), Velocity (temporal granularity of the trajectory data) and Variety (data source(s) utilised).

Majority of the papers relied on implicit trajectory data (8/17), which is motivating; since having the opportunity to observe an object’s location on a continuous space using GNSS data would provide more insights than observing object’s location at places where some specific sensors are installed. Research on big trajectory data was on several research areas ranging from smart transportation governance to animal movement analysis. Data obtained by Automatic Identification System (AIS) constitute as a valuable research in understanding marine traffic.

One of the staggering outcomes of the survey is that the majority of the papers actually relied on a centralised system to manage their data. Some of the research have not stated the computational environment and they are indicated as Not Available, N.A. For instance, even though Yongdong et al. (2020) have not stated the details of the computational environment, one could actually infer that they relied on a centralised environment as it is the traditional method for data analysis, and also that they relied on Postgres. Therefore, including the N.As, only almost one-third (5/17) of the papers relied on a distributed environment.

Paper	Type	Purpose	C/D	Spatial Index	Volume	Velocity	Variety
Fang et al. (2022)	I	Data cleaning on trajectory data.	C	✗	~10K taxi trajectories	~3 min	GPS records*
Rong et al. (2022)	E	Navigation accuracy, marine lane identification	N.A.	N.A.	~8K ship trajectories	N.A.	AIS
S. Wang, Ding, and Cheng (2022)	I	Comparison of actual and shortest paths.	N.A.	N.A.	~2.8K trajectories	5 seconds	GPS records
Yang et al. (2022)	I	human mobility restriction policies to mitigate Covid-19.	N.A.	✓	~1.6M users, ~30B GPS records	~ 5 min/record/user	City POI, human mobility
Pan et al. (2021)	E	Trajectory recommendation based on historical user profiles.	C	Geohash	~50K users, ~215K POI	N.A.	Foursquare check-in records
Fang and Xu (2021)	I	Repairing abnormal GPS records.	C	✓	Three different data sets: 15M, 45M, 1.07M records	10s, 1s, 3 min	GPS records
Wang et al. (2021)	I	Analysing driving behaviours	N.A.	✗	~10K taxi trajectories	~ 3min	GPS records*
Yin, Lin, and Zhao (2021)	E	Understanding daily activity patterns within a city	N.A.	✗	~5.8M mobile phone users, ~6K cell towers	~1h/user	GSM location, POI, travel survey
Ramadhan and Kwon (2021)	I	Enhancing the trajectory KNN search.	N.A.	✗	~50K trajectories	N.A.	GPS records
Niu et al. (2021)	I	Trajectory clustering.	D	N.A.	~196K trajectories	N.A.	GPS records
Yongdong et al. (2020)	E	Travel behaviour analysis	N.A. (Postgres)	✗	~1M records	N.A.	Vehicle origin-destination
Maguerra et al. (2020)	I	Reactive system design	D	✗	N.A.	N.A.	GPS records**
G. Wang et al. (2020)	I	Extracting marine lane information	D	Geohash, quadtree	~510GB	~5 sec. to ~10 min.	AIS
Chen et al. (2020)	I	Improving taxi service.	D	✗	~10K taxi trajectories	~3min	GPS records*
Zhang et al. (2020)	I	Urban trajectory modeling and anomaly detection	C	✗	~10K taxi trajectories	~3min	GPS records*
Eldawy and Mokhtar (2020)	I	Identifying anomalous (sub) trajectories	C	✗	~ 60 trajectories	N.A.	Animal movement data set
Li et al. (2020)	I	Developing an efficient big trajectory management system.	D	Geomesa	~1360GB	N.A.	GPS records

Table 3. The analysis of the recent papers (published in 2020 and onwards) on 'big trajectory data'

***** Open T-drive data set: https://www.microsoft.com/en-us/research/publication/t-drive-trajectory-data-sample/

** Open Geolife data set: https://www.microsoft.com/en-us/research/publication/geolife-gps-trajectory-dataset-user-guide/

It should also be noted that the details of the distributed environment may not be specified as well. For instance, Chen, Fu, and Zhu (2020) relied on Hadoop Distributed File System (HDFS), whereas they have not specified the number of nodes utilised in their experiments. Volume of the analysed data ranges substantially but the largest dataset was analysed by Li et al. (2020) amounting to almost 1.5 TB. Velocity of the data also varied substantially from one second to one hour.

The other important outcome is that majority of the research on big trajectory data relied on a single data source. Consequently, the *variety* component of big data has almost always been overlooked.

4.DISCUSSION

In a smart city context trajectory data are being collected to manage road or marine transportation more efficiently, reduce costs and improve safety, and provide reliable journey times. The literature review conducted in this paper revealed that two important components of 'Big Data' have usually been overlooked by researchers. Most of the research relied on a centralised environment (violating the Volume principle), and relied on a single data source (violating the Variety principle). It can also be argued that the Velocity principle is by default supported when working with implicit trajectory data due to the continuous collection of location of the object.

There may be literature relevant to this paper, but not included in Table 3 due to the use of different terminology. For example, Mao et al. (2021) relied on the keyword '*distributed trajectory streams*' and indeed utilised Spark to detect outlier trajectories. Furthermore, trajectory data analysis is actually an interdisciplinary research endeavour. For example, Gudmundsson and Horton (2017) provided a thorough coverage on the use of trajectories in sports sciences, and specifically in invasion sports such as football or basketball. Similarly, Niu et al. (2021) stated that *streaming trajectory*, *incremental trajectory* or *parallel trajectory* are all used to address research conducted on a distributed environment. Finally, researchers may not have explicitly emphasised the *trajectory* keyword, and only used 'big data' in their papers, which may also lead to the omission of relevant papers. The Strengths-Weaknesses-Opportunities-Threats of research on big trajectory data is summarised in Table 4.

Strengths	**Weaknesses**
• Big trajectory data sets are readily available. • There is a growing research community designing distributed systems to handle big trajectory data.	• Lack of educational material to configure a distributed environment. • Keeping up with literature written in languages apart from English. • Requires a research budget.
Opportunities	**Threats**
• Leading companies support research and education by providing credits.	• Governments restricting the distribution of big trajectory data (for privacy concerns or economic issues).

Table 4. SWOT analysis on research utilising big trajectory data

5.CONCLUSION

Big geospatial data are being collected on a daily basis. One of the common forms of big geospatial data is trajectory data, which is often encountered in a smart-city context. Various domains ranging from intelligent transportation systems to animal movement analysis, and to sports science rely on trajectory data. Openly available datasets provide a good starting point; however, it is difficult to setup a distributed computational environment. Recent technological advances (e.g. Citus extension to Postgres) enable researchers to scale-out computations from a centralised computing environment to a distributed environment. However, there is lack of effective learning material for newcomers, and research on distributed computing environment usually requires a research budget.

ACKNOWLEDGEMENTS

The authors would like to thank Earth on AWS for the credit they provided, and the anonymous reviewers for their constructive feedback.

REFERENCES

Agapiou, Athos. 2017. 'Remote Sensing Heritage in a Petabyte-Scale: Satellite Data and Heritage Earth Engine© Applications'. *International Journal of Digital Earth* 10(1):85–102. doi: 10.1080/17538947.2016.1250829.

Anbaroğlu, Berk. 2021. 'Spatial Query Performance Analyses on a Big Taxi Trip Origin–Destination Dataset'. Pp. 37–53 in *Open Source Geospatial Science for Urban Studies: The Value of Open Geospatial Data*, *Lecture Notes in Intelligent Transportation and Infrastructure*, edited by A. Mobasheri. Cham: Springer International Publishing.

Barann, Benjamin, Daniel Beverungen, and Oliver Müller. 2017. 'An Open-Data Approach for Quantifying the Potential of Taxi Ridesharing'. *Decision Support Systems* 99:86–95. doi: 10.1016/j.dss.2017.05.008.

Bartoszewski, Dominik, Adam Piorkowski, and Michal Lupa. 2019. 'The Comparison of Processing Efficiency of Spatial Data for PostGIS and MongoDB Databases'. Pp. 291–302 in *Beyond Databases, Architectures and Structures. Paving the Road to Smart Data Processing and Analysis*, *Communications in Computer and Information Science*, edited by S. Kozielski, D. Mrozek, P. Kasprowski, B. Małysiak-Mrozek, and D. Kostrzewa. Cham: Springer International Publishing.

Chekol, Ayele Gobezie, and Marta Sintayehu Fufa. 2022. 'A Survey on next Location Prediction Techniques, Applications, and Challenges'. *EURASIP Journal on Wireless Communications and Networking* 2022(1):29. doi: 10.1186/s13638-022-02114-6.

Chen, Yuepeng, Qingwen Fu, and Jiahui Zhu. 2020. 'Finding Next High-Quality Passenger Based on Spatio-Temporal Big Data'. Pp. 447–52 in *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*.

Eldawy, Eman O., and Hoda M. O. Mokhtar. 2020. 'Clustering-Based Trajectory Outlier Detection'. *International Journal of Advanced Computer Science and Applications (IJACSA)* 11(5). doi: 10.14569/IJACSA.2020.0110520.

Fang, Chenglong, Feng Wang, Bin Yao, and Jianqiu Xu. 2022. 'GPSClean: A Framework for Cleaning and Repairing GPS Data'. *ACM Transactions on Intelligent Systems and Technology (TIST)*. doi: 10.1145/3469088.

Fang, Chenglong, and Jianqiu Xu. 2021. 'GPSClean: An Embedded Tool for Cleaning GPS Data'. Pp. 229–32 in *2021 22nd IEEE International Conference on Mobile Data Management (MDM)*.

Ghosh, Shreya, and Soumya K. Ghosh. 2019. 'Traj-Cloud: A Trajectory Cloud for Enabling Efficient Mobility Services'. Pp. 765–70 in *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*.

Gudmundsson, Joachim, and Michael Horton. 2017. 'Spatio-Temporal Analysis of Team Sports'. *ACM Computing Surveys (CSUR)*. doi: 10.1145/3054132.

Haverkort, Herman, and Freek van Walderveen. 2010. 'Locality and Bounding-Box Quality of Two-Dimensional Space-Filling Curves'. *Computational Geometry* 43(2):131–47. doi: 10.1016/j.comgeo.2009.06.002.

Hughes, James N., Andrew Annex, Christopher N. Eichelberger, Anthony Fox, Andrew Hulbert, and Michael Ronquest. 2015. 'GeoMesa: A Distributed Architecture for Spatio-Temporal Fusion'. Pp. 128–40 in *Geospatial Informatics, Fusion, and Motion Video Analytics V*. Vol. 9473. SPIE.

Hughes, James N., Matthew D. Zimmerman, Christopher N. Eichelberger, and Anthony D. Fox. 2016. 'A Survey of Techniques and Open-Source Tools for Processing Streams of Spatio-Temporal Events'. Pp. 1–4 in *Proceedings of the 7th ACM SIGSPATIAL International Workshop on GeoStreaming*. Burlingame California: ACM.

Jitkajornwanich, Kulsawasd, Peerapon Vateekul, Upa Gupta, Teeranai Kormongkolkul, Arnon Jirakittayakorn, Siam Lawawirojwong, and Siwapon Srisonphan. 2017. 'Ocean Surface Current Prediction Based on HF Radar Observations Using Trajectory-Oriented Association Rule Mining'. Pp. 4293–4300 in *2017 IEEE International Conference on Big Data (Big Data)*.

Khan, M. Ali-ud-din, Muhammad Fahim Uddin, and Navarun Gupta. 2014. 'Seven V's of Big Data Understanding Big Data to Extract Value'. Pp. 1–5 in *Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education*.

Kong, Xiangjie, Menglin Li, Kai Ma, Kaiqi Tian, Mengyuan Wang, Zhaolong Ning, and Feng Xia. 2018. 'Big Trajectory Data: A Survey of Applications and Services'. *IEEE Access* 6:58295–306. doi: 10.1109/ACCESS.2018.2873779.

Li, Ruiyuan, Huajun He, Rubin Wang, Sijie Ruan, Yuan Sui, Jie Bao, and Yu Zheng. 2020. 'TrajMesa: A Distributed NoSQL Storage Engine for Big Trajectory Data'. Pp. 2002–5 in *2020 IEEE 36th International Conference on Data Engineering (ICDE)*.

Lwin, Ko Ko, Yoshihide Sekimoto, Wataru Takeuchi, and Koji Zettsu. 2019. 'City Geospatial Dashboard: IoT and Big Data Analytics for Geospatial Solutions Provider in Disaster Management'. Pp. 1–4 in *2019 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*.

Maguerra, Soufiane, Azedine Boulmakoul, Lamia Karim, Hassan Badir, and Ahmed Lbath. 2020. 'Towards a Reactive System for Managing Big Trajectory Data'. *Journal of Ambient Intelligence and Humanized Computing* 11(10):3895–3906. doi: 10.1007/s12652-019-01625-3.

Manolopoulos, Yannis, Yannis Theodoridis, and Vassilis J. Tsotras. 2009. 'Spatial Indexing Techniques'. Pp. 2702–7 in *Encyclopedia of Database Systems*, edited by L. LIU and M. T. ÖZSU. Boston, MA: Springer US.

Mao, Jiali, Jiaye Liu, Cheqing Jin, and Aoying Zhou. 2021. 'Feature Grouping–Based Trajectory Outlier Detection over Distributed Streams'. *ACM Transactions on Intelligent Systems and Technology (TIST)*. doi: 10.1145/3444753.

Moon, B., H. V. Jagadish, C. Faloutsos, and J. H. Saltz. 2001. 'Analysis of the Clustering Properties of the Hilbert Space-Filling Curve'. *IEEE Transactions on Knowledge and Data Engineering* 13(1):124–41. doi: 10.1109/69.908985.

du Mouza, Cedric, Witold Litwin, and Philippe Rigaux. 2007. 'SD-Rtree: A Scalable Distributed Rtree'. Pp. 296–305 in *2007 IEEE 23rd International Conference on Data Engineering*.

Niu, Xinzheng, Yunhong Zheng, Philippe Fournier-Viger, and Bing Wang. 2021. 'Parallel Grid-Based Density Peak Clustering of Big Trajectory Data'. *Applied Intelligence*. doi: 10.1007/s10489-021-02757-w.

Oktay, Talha, Erdenay Yoğurtçuoğlu, Ramazan Nejdet Sarıkaya, Ali Recep Karaca, Mehmet Fırat Kömürcü, and Ahmet Sayar. 2021. 'Multimodel Anomaly Detection on Spatio-Temporal Logistic Datastream with Open Anomaly Detection Architecture'. *Expert Systems with Applications* 186:115755. doi: 10.1016/j.eswa.2021.115755.

Pan, Xiao, Lei Wu, Fenjie Long, and Ang Ma. 2021. 'Exploiting User Behavior Learning for Personalized Trajectory Recommendations'. *Frontiers of Computer*

Science 16(3):163610. doi: 10.1007/s11704-020-0243-2.

Ramadhan, Hani, and Joonho Kwon. 2021. ‘Enhancing Learned Index for A Higher Recall Trajectory K-Nearest Neighbor Search’. Pp. 6006–7 in *2021 IEEE International Conference on Big Data (Big Data)*.

Rong, Yu, Zhong Zhuang, Zhengwei He, and Xuming Wang. 2022. ‘A Maritime Traffic Network Mining Method Based on Massive Trajectory Data’. *Electronics* 11(7):987. doi: 10.3390/electronics11070987.

Saha, Bikas, Hitesh Shah, Siddharth Seth, Gopal Vijayaraghavan, Arun Murthy, and Carlo Curino. 2015. ‘Apache Tez: A Unifying Framework for Modeling and Building Data Processing Applications’. Pp. 1357–69 in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, *SIGMOD ’15*. New York, NY, USA: Association for Computing Machinery.

Vavilapalli, Vinod Kumar, Arun C. Murthy, Chris Douglas, Sharad Agarwal, Mahadev Konar, Robert Evans, Thomas Graves, Jason Lowe, Hitesh Shah, Siddharth Seth, Bikas Saha, Carlo Curino, Owen O’Malley, Sanjay Radia, Benjamin Reed, and Eric Baldeschwieler. 2013. ‘Apache Hadoop YARN: Yet Another Resource Negotiator’. Pp. 1–16 in *Proceedings of the 4th annual Symposium on Cloud Computing*. Santa Clara California: ACM.

Wang, Di, Tomio Miwa, and Takayuki Morikawa. 2020. ‘Big Trajectory Data Mining: A Survey of Methods, Applications, and Services’. *Sensors* 20(16):4571. doi: 10.3390/s20164571.

Wang, Guiling, Jinlong Meng, Zhuoran Li, Marc Hesenius, Weilong Ding, Yanbo Han, and Volker Gruhn. 2020. ‘Adaptive Extraction and Refinement of Marine Lanes from Crowdsourced Trajectory Data’. *Mobile Networks and Applications* 25(4):1392–1404. doi: 10.1007/s11036-019-01454-w.

Wang, Li-Wen, Du Li, Wan-Chi Siu, and Daniel Pak-Kong Lun. 2022. ‘Robust Lane Detection through Automatic Trajectory Analysis with Deep Learning and Big Data Environment’. Pp. 425–30 in *International Workshop on Advanced Imaging Technology (IWAIT) 2022*. Vol. 12177. SPIE.

Wang, Pengyang, Xiaolin Li, Yu Zheng, Charu Aggarwal, and Yanjie Fu. 2021. ‘Spatiotemporal Representation Learning for Driving Behavior Analysis: A Joint Perspective of Peer and Temporal Dependencies’. *IEEE Transactions on Knowledge and Data Engineering* 33(2):728–41. doi: 10.1109/TKDE.2019.2935203.

Wang, Shiguang, Heng Ding, and Zeyang Cheng. 2022. ‘Understanding the Shortest Route Selection Behavior for Private Cars Using Trajectory Data and Navigation Information’. *Journal of Advanced Transportation* 2022:e4100704. doi: 10.1155/2022/4100704.

Warren, Michael S., Steven P. Brumby, Samuel W. Skillman, Tim Kelton, Brendt Wohlberg, Mark Mathis, Rick Chartrand, Ryan Keisler, and Mark Johnson. 2015. ‘Seeing the Earth in the Cloud: Processing One Petabyte of Satellite Imagery in One Day’. Pp. 1–12 in *2015 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*.

Xu, Jun, Rouhollah Rahmatizadeh, Ladislau Bölöni, and Damla Turgut. 2018. ‘Real-Time Prediction of Taxi Demand Using Recurrent Neural Networks’. *IEEE Transactions on Intelligent Transportation Systems* 19(8):2572–81. doi: 10.1109/TITS.2017.2755684.

Yang, Chuang, Zhiwen Zhang, Zipei Fan, Renhe Jiang, Quanjun Chen, Xuan Song, and Ryosuke Shibasaki. 2022. ‘EpiMob: Interactive Visual Analytics of Citywide Human Mobility Restrictions for Epidemic Control’. *IEEE Transactions on Visualization and Computer Graphics* 1–1.

Yin, Ling, Nan Lin, and Zhiyuan Zhao. 2021. ‘Mining Daily Activity Chains from Large-Scale Mobile Phone Location Data’. *Cities* 109:103013. doi: 10.1016/j.cities.2020.103013.

Yongdong, Wang, Xu Dongwei, Peng Peng, and Zhang Guijun. 2020. ‘Analysis of Road Travel Behaviour Based on Big Trajectory Data’. *IET Intelligent Transport Systems* 14(12):1691–1703. doi: 10.1049/iet-its.2019.0785.

Zaharia, Matei, Reynold S. Xin, Patrick Wendell, Tathagata Das, Michael Armbrust, Ankur Dave, Xiangrui Meng, Josh Rosen, Shivaram Venkataraman, Michael J. Franklin, Ali Ghodsi, Joseph Gonzalez, Scott Shenker, and Ion Stoica. 2016. ‘Apache Spark: A Unified Engine for Big Data Processing’. *Communications of the ACM* 59(11):56–65. doi: 10.1145/2934664.

Zhang, Haiyan, Yonglong Luo, Qingying Yu, Liping Sun, Xuejing Li, and Zhenqiang Sun. 2020. ‘A Framework of Abnormal Behavior Detection and Classification Based on Big Trajectory Data for Mobile Networks’. *Security and Communication Networks* 2020:e8858444. doi: 10.1155/2020/8858444.

Zheng, Yu. 2015. ‘Trajectory Data Mining: An Overview’. *ACM Transactions on Intelligent Systems and Technology* 6(3):1–41. doi: 10.1145/2743025.

Zhu, Yuan, Kaan Ozbay, Kun Xie, and Hong Yang. 2016. ‘Using Big Data to Study Resilience of Taxi and Subway Trips for Hurricanes Sandy and Irene’. *Transportation Research Record* 2599(1):70–80. doi: 10.3141/2599-09.