Zayed University

ZU Scholars

10-1-2022

Augmented Reality and GPS-Based Resource Efficient Navigation System for Outdoor Environments: Integrating Device Camera, Sensors, and Storage

Saravjeet Singh
Chitkara University, Punjab

Jaiteg Singh
Chitkara University, Punjab

Babar Shah
Zayed University, babar.shah@zu.ac.ae

Sukhjit Singh Sehra
Wilfrid Laurier University

Farman Ali
Sejong University

Recommended Citation

Article

Augmented Reality and GPS-Based Resource Efficient Navigation System for Outdoor Environments: Integrating Device Camera, Sensors, and Storage

Saravjeet Singh [1,†]**, Jaiteg Singh** [1,*,†]**, Babar Shah** [2]**, Sukhjit Singh Sehra** [3] **and Farman Ali** [4,*,†]

1   Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab 140401, India
2   College of Technological Innovation, Zayed University, Dubai 19282, United Arab Emirates
3   Department of Physics & Computer Science, Wilfrid Laurier University, Waterloo, ON N1048, Canada
4   Department of Software, Sejong University, Seoul 05006, Korea
*   Correspondence: jaiteg.singh@chitkara.edu.in (J.S.); farmankanju@sejong.ac.kr (F.A.)
†   These authors contributed equally to this work and are first co-authors.

Abstract: Contemporary navigation systems rely upon localisation accuracy and humongous spatial data for navigational assistance. Such spatial-data sources may have access restrictions or quality issues and require massive storage space. Affordable high-performance mobile consumer hardware and smart software have resulted in the popularity of AR and VR technologies. These technologies can help to develop sustainable devices for navigation. This paper introduces a robust, memory-efficient, augmented-reality-based navigation system for outdoor environments using crowdsourced spatial data, a device camera, and mapping algorithms. The proposed system unifies the basic map information, points of interest, and individual GPS trajectories of moving entities to generate and render the mapping information. This system can perform map localisation, pathfinding, and visualisation using a low-power mobile device. A case study was undertaken to evaluate the proposed system. It was observed that the proposed system resulted in a 29 percent decrease in CPU load and a 35 percent drop in memory requirements. As spatial information was stored as comma-separated values, it required almost negligible storage space compared to traditional spatial databases. The proposed navigation system attained a maximum accuracy of 99 percent with a root mean square error value of 0.113 and a minimum accuracy of 96 percent with a corresponding root mean square value of 0.17.

Keywords: location awareness; prospective memory; embedded navigational intelligence; vision services; sustainable urban innovation

1. Introduction

Emerging technologies such as augmented and virtual reality (AR and VR) have resulted in research tools that may offer a decent balance between ecological validity and experimental control for research studies. Affordable high-performance mobile consumer hardware and smart software have resulted in the popularity of AR and VR technologies [1,2] (Pan and Hamilton). The primary difference between these two technologies is their intertwining of the virtual and real worlds. VR offers a synthetic view of the content presented with the help of stereoscopic displays, whereas AR imbues virtual and real-world spatial structures. Navigation systems are one domain where augmented reality may contribute to the maximum. Navigation systems use positioning systems such as the Global Positioning System (GPS) to determine the position of any person or object. Fortunately, GPS technology has become integral to current mobile devices [3,4]. Such integration has resulted in numerous context-aware and location-based services (LBS). Typical LBS have four major components, and these are explained below:

- Location Information: The location information component provides the location information. These components can be a direct or indirect source of location providers, e.g., GPS receivers, inertial sensors, deduced reckoning techniques, cellular systems, and Wi-Fi devices. GPS receiver provides output in the form of longitude and latitudes but sometimes lacks in performance due to shielding effects.
- Reference data: Reference data provides digital map information for the geographical entities. This information includes location, topological, Arial, and content information. The sources of reference data can be proprietary or non-proprietary. Proprietary data sources are Google Maps, Bing Maps, TomTom map, etc. Non-proprietary data sources include OpenStreetMap data, GPX records, etc.
- Processing Unit: The processing unit is responsible for information retrieval and knowledge generation. This system fetches the information from the location information component and reference data source and uses its supporting tool to retrieve the knowledge.
- Output Unit: Any audio-visual device that can communicate knowledge to the user. The output unit takes the help of the graphical user interface or some storage medium for displaying the output.

Reference data is crucial for any navigation system. The more precise your reference data, the more accurate your navigation system is. Though proprietary datasets charge a license fee to avail of their services, they do not allow the use of their data for analysis and research purposes. On the other hand, OpenStreetMap (OSM) shares its data under a GNU license and does not restrict its use. With crowdsourced data, everyone can contribute, update, correct, and share spatial data. Many commercial navigation systems rely on OSM data for their reference data [5–9]. Although the OSM community has described some fundamental attributes for the data, it does not mandate full compliance.

Further, contributors may use nonstandard hardware and nomenclature to contribute data, which may add real challenges to maintaining data quality. Owing to the complexity of data structures used for capturing and developing reference data, analysing spatial data may require many generalisations due to the volume of data to be analysed [10–12]. If the navigation system is meant to offer offline services, then the size of reference data itself is a prime challenge. Reference data may not only require huge storage space, but it may also slow down the system itself [13,14]. Further, the renderer services responsible for showing graphics to assist route guidance are also responsible for guzzling hardware resources. The more spatial information added to reference data, the more would be the size of reference data. There is a need to revamp the traditional architecture of navigation systems to make these navigation systems more scalable, robust, and user-friendly. This paper proposes a novel approach for location identification and route finding for offline outdoor navigation systems. It uses minimal storage and processing resources and is not reliant on traditional spatial databases. The proposed navigation system merely uses GPS as a location provider and a comma separated file (CSV) file as a spatial data source. The proposed system supports the device camera's augmented graphical user interface (GUI). There is no need to process heavy shape files to generate the GUI effect.

The rest of this paper is organised as follows: Sections 2 and 3 provide the literature study and proposed methodology, respectively. The proposed system is presented in Section 4, a case study and experimental results are explained in Section 5. The future scope and conclusion are provided in Section 6.

2. Literature Studies

Location-based services deal with the user's current location and provide different functionality based on the user's location. Firm growth in the development of LBS is possible due to the easy availability of small GPS receivers [15]. First, LBS (Active Badge) was based on a sensor network to find the user's location [16]. Since 2000, many GPS-based applications have come into existence; LBS has become an interest for both business and educational purposes, and navigation system is one of them [17].

According to the area of usage, navigation systems are of two types: indoor and outdoor. Indoor navigation systems are used to navigate inside a building or any enclosed area, whereas outdoor navigation systems provide navigation options in a road network or any open area. According to visual utility, outdoor navigation systems are further categorised into map-based, map-less, and mapbuild-based [18–22].

Mapbuild-based systems are based on the concept of Simultaneous Localisation and Mapping (SLAM) systems. Due to the unavailability of maps, considerable progress in SLAM systems for navigation was achieved during the last decade. SLAM system uses image stream from the camera, map information such as sparse and dense information, and a visual odometer for navigation purposes. These systems estimate and map the location by using the camera images. Typically, SLAM systems are based on a teach-and-repeat process [23]. SLAM system uses bundle adjustment (BA) for localisation using camera and sensing devices. BA application for the SLAM system was initially used using the visual odometer [24]. These systems perform localisation, frame selection, feature selection, mapping, tracking, and navigation [25,26].

Further accuracy in the SLAM system is enhanced using the concept of convergence theorem. The convergence theorem states that if any sequence grows and is bounded by the least upper bound, then the sequence is converted to the least upper bound. Similarly, if any sequence decreases and is bounded by the greatest lower bound, the sequence converts to the greatest lower bound. Errors in distance calculations and landmark estimations were controlled and adjusted using the convergence estimation [27]. The coordinate transformation was eliminated, and navigation was performed using the non-coordinate system to reduce the complexity of the SLAM system. In a non-coordinate system, the device is trained before the actual navigation using navigation components such as path, obstacle, and places. This system does not require a map and uses implicit substitutes based on cause–effect correspondence using associative memory. Robot navigation is an example of non-coordinate-based navigation. Causes are the predefined action of the robot. Camera images and sensor data define the view and act as the robot's input. Based on views and causes, the robot takes navigational steps. These steps act as effects on the robot [28].

Due to rapid development in road infrastructure, digital maps are not updated accordingly (primarily for rural areas). This causes many challenges during autonomous navigation. A mapless driving framework was proposed to reduce the complexity of the SLAM system. This framework uses topological maps with sensor-based systems to provide local perception. In this approach, the system first generates local and global perception using waypoint and local perception [29].

Further, a 3M ("multi-robot, multi-scenario, and multi-stage")-based framework was designed to improve the accuracy of mapless navigation. In reinforcement learning, robot and environmental data are used as training data to generate the navigation steps. This work uses gradient-based reinforcement learning to train the robot using complex crowd environmental data. The reinforcement environment was set up using the 4-tuple Markov decision process [30]. A hidden Markov model with an optimal transition matrix was used to provide the route in a road-less environment [31].

The favourite route recommendation (FAVOUR) algorithm has been developed to select routes in a multi-model environment. This algorithm provides routes using situation-aware and personal data [32]. Optimal route selection using eco-friendly factors was designed. This route selection algorithm uses the traffic-light signal timing, traffic at the intersection, and a star route finding algorithm to provide the optimal route with less energy consumption [33]. To provide routes in both online and offline scenarios, the Graphium Map Matching (GraphiumMM) algorithm was proposed using both topological and geometrical features of the road network and GPS trajectories [34].

The research community proposed many navigation systems and map matching algorithms to increase the accuracy and performance of the mapping process [35,36]. One of the prime reasons for poor accuracy is an error in spatial data [37]. Although online spatial data are being handled by big ventures and have fewer errors, offline, non-proprietary

spatial data have many errors and quality issues [6,7], [38,39]. These quality issues are due to less contribution by society and volunteer access. Navigation based on these offline spatial dataset causes poor accuracy and a high error rate in location mapping [35–37].

In past research, using different techniques, the concept of augmented reality was introduced with navigation in indoor and outdoor networks. Location information and digital photographs of the location were used as markers for indoor navigation. These markers act as a label for the input for augmented-reality-based navigation. For augmented reality in indoor navigation, markers were created using building labels, particular points, turn points, and floor information. For providing navigation inside the building, QRcode, Beacon messages, points on the floor, special codes for markers, device cameras, and graph databases were used [40–44]. The OpenStreetMap dataset used a graph database to add augmented reality to outdoor navigation. This graph database stores information in the form of nodes and edges. Nodes contain data in the form of text information and images. Edges define the link between two or more images. In this approach, location images of the navigational area were stored in a graph database, and their mapping was performed with location information. Different images were linked to each other in order of their sequence in the actual route. While navigating, images from the graph database were fetched using the location information received from the GPS device. These images were displayed to the user as output [45].

Further, camera images and points of interest were mapped to provide outdoor augmented reality. In outdoor navigation, location images with GPS and inertial sensors were used to add augmented reality. Inertial sensors compute any device or body's orientation, force, and angular velocity. The inertial sensor comprises a gyroscope and an accelerometer for the relative movement estimation. The different researchers also addressed the use of sensors in outdoor networks and body area networks' impact on spatial data quality [46–48].

Map-based navigation requires prior knowledge of environment structure (such as geometric and topological information of the road network) for navigation. In contrast, a map-less navigation system aims to recognise the road network during motion and decide based on the non-map element. A mapbuild-based system builds the map itself for localisation and routing. This mapbuild-base system is known as SLAM (Simultaneous Localisation and Mapping). These systems do not require a map for localization and navigation. For localization, these systems use sensor and movement measurements. Similarly, these systems build routing and navigation maps using cameras and sensors. For example, room cleaning robots. These robots use sensors and wheel movement information for obstacle avoidance and wall detection. These robots use cameras and sensors to create a map for routing [23]. Table 1 provides comparisons among map-based, map-less, and mapbuild-based navigation systems. According to the studied literature, considerable research is performed to improve the accuracy and performance of map-less, map-based, and map-build-based navigation systems. According to the studied literature, less work has been done to reduce the size of ever-increasing spatial databases [44,45,49].

Further, the impact of massive storage on processing resources and rendering of information has not been investigated to the extent. Furthermore, the integration of augmented reality with navigation systems has only been available for licensed proprietary systems, which also demands huge storage and processing power. In the purview of the previously mentioned observations, there was a need to propose an efficient, reliable, scalable navigation system with augmented-reality support for navigation instructions. Complete details of the proposed system are provided in subsequent sections.

Table 1. Comparision of different navigation systems.

Map-Based	Map-Less	Mapbuild-Based
Requires prior knowledge of environment structure (such as geometric and topological information of the road network) for navigation [13,33,50].	Recognises the road network during motion and makes the decision based on the non-map element [22,51,52].	Builds map itself for the localization and routing, also called SLAM (Simultaneous Localization and Mapping) system [24,25,28,53].
These systems use predefined maps that include all geometrical and topological attributes. Routing and navigation algorithms are used to find the path and localization of the device.	These systems do not require a map for routing and localization. For navigation, these systems are trained on predefined routes, and after training, the device can move based on the training instruction.	These systems do not require a predefined map for localization and navigation but create a map while navigating. These systems use sensor and movement measurements for localization, but for routing and navigation, these systems build maps using cameras and sensors.
GPS-based routing is an example of a map-based system. These systems generate the route using map information, starting address, and ending address.	Data transmission robots are an example of a map-less system. These systems are trained on a predefined route and after training, the robot can move on the same route without any map information.	Room cleaning robots are an example of mapbuild-based navigation. These robots use sensors and wheel movement information for obstacle avoidance and wall detection. These robots use cameras and sensors to create a map for navigating.

3. Research Methodology

This section covers the system description and details the adopted methodology for this experiment. There is a need to understand a few fundamental concepts and definitions, as discussed below, to elaborate the entire methodology:

3.1. Fundamental Terms

- Node: A node is the fundamental element of the map. If we consider a map a graph, then the node is equivalent to a vertex of the graph. For example, M (V, E) is a map, V comprises all the map nodes, and E comprises all the map edges. In Figure 1, A-I is the map node, and set V comprises [A, B, C, D ... I].
- Edge: An edge is a link between two nodes of the map. If vi, vj are the nodes of map M (V, E), then edge eij is the direct link between vi and vj. Whereas vi, vj € V and eij € E. In a map, the collection of edges is called a road. A road consists of finite connected edges. Figure 1 shows the edge as a link between two nodes.
- Route: A route is a path between two directly or indirectly connected nodes. If two nodes are directly connected, the route is the same as edge, whereas if two nodes are not directly connected, the route is a collection of adjacent edges. The route always has a finite length. In Figure 1, A-B-D-F-H-I is an example of a route.
- GPS receiver output: GPS receiver output provides the positional and temporal information of the device. It comprises latitude and longitude information with a timestamp.
- GPS Trajectory: GPS trajectory is formed by collecting continuous GPS points with respect to time. Figure 1 shows a sample scenario of a road network where A–I are the road nodes and GPS trajectory collected from GPS receiver by traversing path A-C-E-G-I is shown using a dotted line. Each trajectory element includes latitude, longitude, and time information.
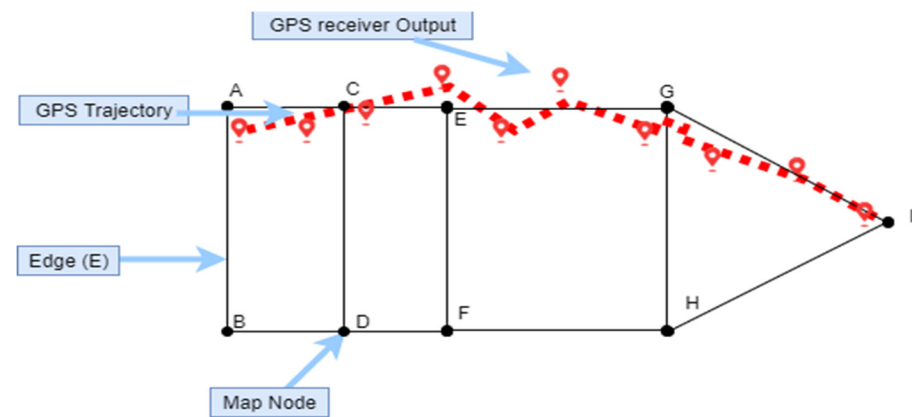
Figure 1. Basic map elements.

3.2. A Star Algorithm

A star (A*) algorithm provides the shortest path between source and destination node of a graph. A star algorithm uses Equation (1) to calculate the path cost [54,55].

$$Path\_Cost(i) = \text{Traversed\_cos}\,t(i) + Estimation(i) \tag{1}$$

where i is the current node, Traversed_cost(i) is the path cost between s and i; Estimation(i) provides the minimum cost between node i and destination node d.

3.3. Nearest Neighbor Search Algorithm

The nearest neighbour search (NNS) algorithm finds the closest point within a given set to any given point. NNS based on proximity search is used for solving the optimisation problem [56]. The NNS problem is defined as: giving a set of P points from a complete set S (where P subset of S). Action point A € S would find the nearest point in P to A. The generalisation of NNS is k-NNS, which finds the k nearest points to a given point. Let A be a sample input point having n features (A1, A2.....An), and P a set of points for finding the closest point. Each point in P has n features (P11, P12, . . . , P1n), then NNS finds the closest point using the Euclidean distance as given in Equation (2). A point with the minimum Euclidean distance is considered the closest point.

$$D(A, P_i) = \sqrt{(A_1 - P_{i1})^2 + (A_2 - P_{i2})^2 + \cdots \ldots\ldots + (A_n - P_{in})^2} \tag{2}$$

3.4. The Proposed System

The proposed system (resource-efficient navigation—REN) provides route information between source and destination with minimal storage and computational resources. The basic component of the system is shown in Figure 2. REN operates with the help of four basic subsystems responsible for data preparation, location identification, path generation, and a visual output component. The functioning of these subsystems is detailed below and depicted in Figure 3.

3.4.1. Data Preparation (S1)

This subsystem is responsible for generating navigation reference spatial data (map) for the navigation using the trajectories of moving entities and basic map components. This component (S1) generates a map in the format of CSV files and will be used later for routing and navigation purposes. Location information of the user was fetched using the GPS receiver output. Further, these GPS trajectories were used to enhance the quality of data.
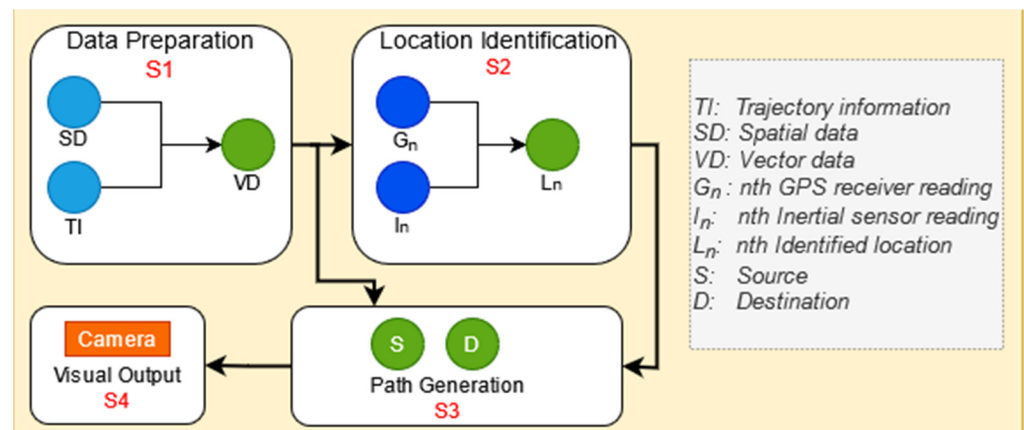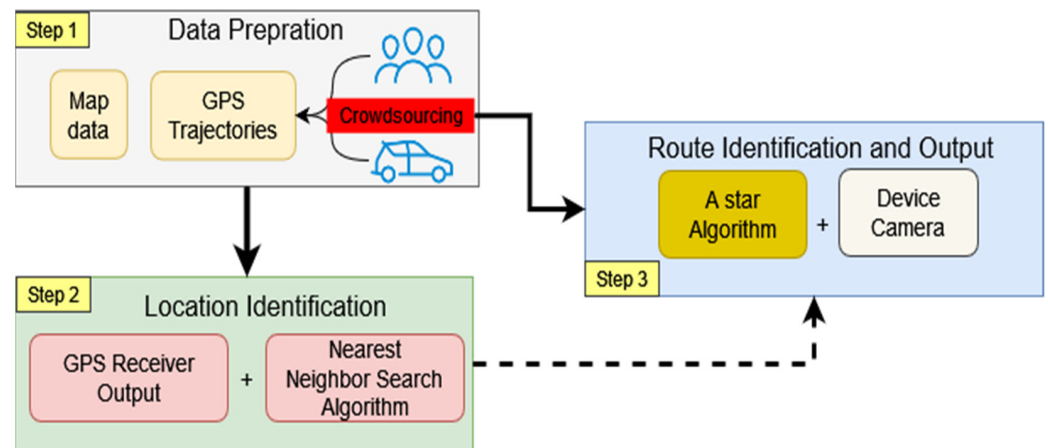
Figure 2. Overview of the proposed REN system.

Figure 3. Complete description of resource-efficient system.

3.4.2. Location Identification (S2)

S2 records the immediate location information of the user. The location information fetched from the GPS receiver may have a particular drift from the actual location due to transmission delay and associated errors. S2 maps the recorded immediate GPS receiver output to the actual location on the digital map using the k-nearest neighbour algorithm with kd tree and support vector machine [49]. S2 subsystem is responsible for location identification by using the GPS receiver output and reference spatial data. Due to transmission delay and associated errors with the receiver, the location information fetched from the GPS receiver may have a certain drift from the actual location, so this subsystem keeps records of the previous location information of the user to process the current location.

3.4.3. Route Identification (S3)

For navigation from one location to another location, it is mandatory to have a route. S3 generates the route between the start and end location, passing through a fixed location. Finding the single route from all possible is called route identification. For route identification, S3 relies on the A star algorithm to find the shortest path.

3.4.4. Visual Output (S4)

S4 provides output to the user using the device camera and screen. In this system, augmented reality has been provided using the device camera output. While navigating, the device's back camera provides a real-time view of the road, which will be used with the output of the S3 component to provide user output. This component replaces the heavy shapefiles of spatial data to provide the map GUI.

4. The REN System

The REN system identifies location and path with optimised space and processing requirements. According to the studied literature, navigation systems primarily use either online spatial data (API support) or offline spatial data for navigation purposes. The proposed REN stored spatial data in a comma-separated file system. This data contained only the geometrical and topological information about the road network. In contrast to other offline storage systems, the REN dataset does not contain data for the graphical user interface and map visualisation (like .shp, .shx files). We used an integrated mobile camera and screen to render navigation instructions for graph visualisation.

A Kd tree-based K nearest neighbour algorithm was used for the location identification, and A star algorithm was deployed to calculate the shortest path calculation. The complete process for the navigation between source to destination using REN is shown in Figure 4. Step-by-step working of the REN system is shown in Algorithm 1.

Algorithm-1: REN (G, S, E)

Input: Finite connected map data comprises V vertices and E edges.
Output: Route from source to destination

---

1. Start
2. Set t = 0
3. Take the output of GNSS receiver ft and direction of motion of the device
4. If (ft is the first observation and ft =~ S)
5.   Set ft to S
6. Else
7.   Map ft using K-NN search algorithm to the resultant node that is Sn.
8. Set C = Sn
9. Update shortest path SP using A star algorithm between current node C and E
10. Do
11. Take next GPS receiver output ft
12. For each previous fix ft-1 update shortest path SP using A star algorithm between the starting point (S) and ending point (E) passing through ft-1
13. Identify the direction of movement using the point orientation algorithm
14. Identify actual map node using kd tree based nearest node search algorithm
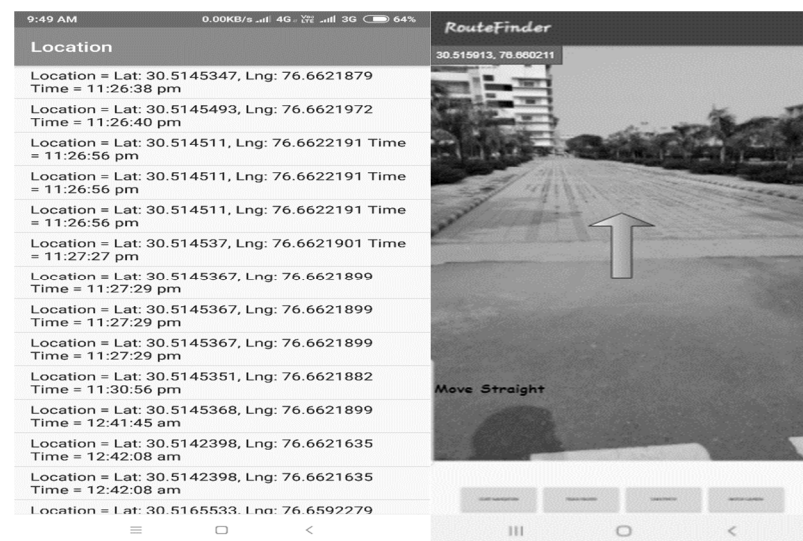15. While (ft-1! = E)
16. End

Figure 4. Designed android application for the experiment.

5. Case Analysis

The proposed REN system was verified and validated using a case study. For the case study, the road networks of Punjab and Haryana (two states of India) were considered. The considered road networks include urban, suburban and rural areas. Different types of roads were considered for the experiment based on the selected area. For this experiment, the longitude and latitudes of the road networks were considered. Due to its user-friendly interface and easy availability, mobile phone-based applications are frequently used for navigation. Therefore, for this experiment, the REN prototype was developed as an Android application. During the experiment, two modes of travel were considered: pedestrian and travel modes. In travel mode, the device speed was from 5 km/h to 80 km/h (based on allowed traffic limits).

A mobile application (as shown in Figure 4) was developed for data collection and to analyse the accuracy and performance of the REN system. The frequency of the inertial unit was about 30 Hz (it depends on the mobile device used to implement REN). The sample rate for the GPS receiver was between 1 to 10 seconds, depending on the type of travelling mode (pedestrian or vehicle). Data from predefined routes were first collected to create the spatial reference data (i.e., map in CSV format). A few volunteers were selected, and their GPS trajectories were captured while moving, as shown in Table 2.

Table 2. Route Specification.

Route Number	Distance (in km)	Road Beginning and Ending Position (Lat, Lon) in Degree Unit	Route Type
1	0.3	(30.516913,76.660170)(30.514242,76.660846)	Pedestrian
2	0.65	(30.512529,76.658809)(30.517021,76.660418)	Pedestrian
3	7.2	(30.518102,76.659002)(30.487103,76.603105)	Road
4	32.4	(30.518102,76.659002)(30.338219,76.832267)	Road
5	16	(30.281446,76.834756)(30.335909,76.834155)	Road
6	45.3	(30.515920,76.658856)(30.258633,76.852609)	Road
7	48.2	(30.257539,76.852845)(30.478197,76.578609)	Road
8	60.3	(30.173239,76.861614)(30.532606,76.678967)	Road

For this experiment, we considered eight road networks (shown in Table 2) covering pedestrian and vehicle movement. The mobile phone's built-in GPS and GLONASS module was used to receive the GPS points.

5.1. Comparison Metrics

The proposed system was evaluated based on accuracy, performance, and space matrices. The proposed approach calculates accuracy by identifying the correct location mapping for POI's. Accuracy determines the percentage of correctly identified locations concerning the recorded GPS points. Equation (3) provides the formula to calculate the accuracy. Further, to identify the accuracy, matched/unmatched node count (MaUC) was calculated based on the wrong and correct mapping of GPS points. If the received GPS receiver points are $T = \langle T_1, T_2, \ldots, T_u \rangle$ and $P = \langle P_1, P_2, \ldots, P_m \rangle$ are the road points, then MaUC returns the number of T nodes correctly and wrongly mapped with P.

$$Accuracy = \frac{Total\ number\ of\ matched\ nodes}{Total\ number\ of\ node\ considered} \times 100 \qquad (3)$$

Similarly, the root mean square error (RMSE) metric was used to identify the error rate in the mapping process. RMSE calculates the deviation between obtained mapping results and standard mapping results. The RMSE of received GPS receiver points

$T = \langle T_1, T_2, \ldots, T_u \rangle$ can be calculated using Equation (4), where SD $= \langle SD_1, SD_2, \ldots, SD_u \rangle$ is the standard mapping series.

$$RMSE = \sqrt{\frac{1}{u} * \sum_{i=1}^{u} (SD_i - T_i)^2} \tag{4}$$

Storage requirement is also calculated using the total storage space required on secondary storage for application data. To identify the performance of the proposed system, execution time of location identification and route-finding algorithm is calculated using Equation (5). It includes the total time elapsed by an algorithm to generate the results. Where ET is the end time of the process and ST is the start time of the process.

$$\text{Total Execution Time (TT)} = \text{ET} - \text{ST} \tag{5}$$

5.2. Experimental Results

To evaluate the performance of the proposed system in real time, we used mobile devices in an outdoor environment. During the experiment, the mobile device was kept in such a position that *x*-axis of the device is along the forward direction and *y*-axis on the right side with the perpendicular to the *x*-axis. The Z-axis is perpendicular to the plane formed by x and *y*-axis, which is an x-y plane. We performed experiments in an outdoor environment to obtain the maximum precision value of the GPS receiver and good heading information of the inertial unit. The experiments were executed two–five times by eight different users/participants on pre-decided routes (as shown in Table 2). The distance covered by the eigjt users is shown in Table 3.

Table 3. User information on test.

Route Number	User1	User2	User3	User4	User5	User6	User7	User8
Distance Covered (in km)	80	105	186	33	93.5	108	134	77

The accuracy and RMSE of the location identification process were calculated by analysing the experimental data received from eight users. The average accuracy of 97.75 per cent and an average RMSE of 13.2 per cent (0.132 units) were recorded. The value of accuracy and RMSE achieved by eight users is shown in Figure 5 According to Figure 5, the maximum recorded accuracy and RMSE were 99 per cent and 17 per cent (0.17 units), respectively, whereas the minimum value of accuracy and RMSE were 96 and 11.3 per cent, respectively.
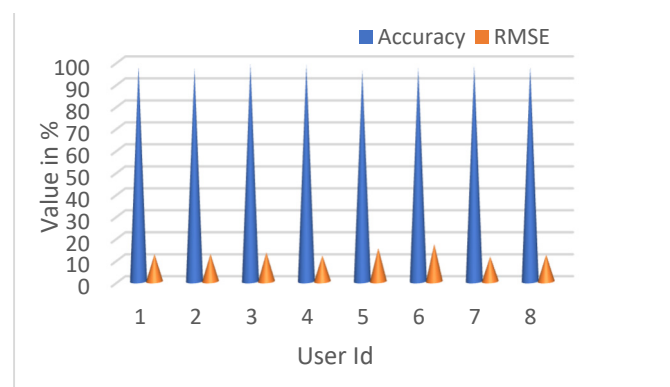

Figure 5. Accuracy and RMSE value achieved by eight different users.

Further, to check the impact of the density of road nodes on the accuracy, we experimented on a pedestrian track (of distance 300 m) having different nodes count (in the range of 10 to 100). The experiment was repeated 10 times for 30 GPS points each time with a sampling rate of 10 seconds and observed the accuracy in terms of accuracy percentage, RMSE, and matched/unmatched node count (MaUC). Figures 6–8 shows the accuracy achieved concerning different node densities.
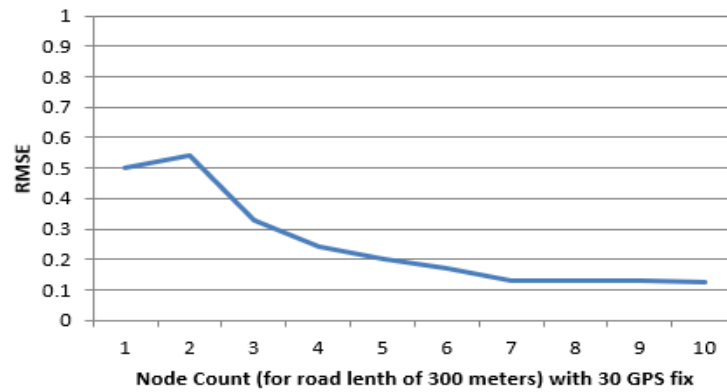

Figure 6. Accuracy analysis of REN system based upon RMSE vs. node count (for road length of 300 m) with 30 GPS fix.
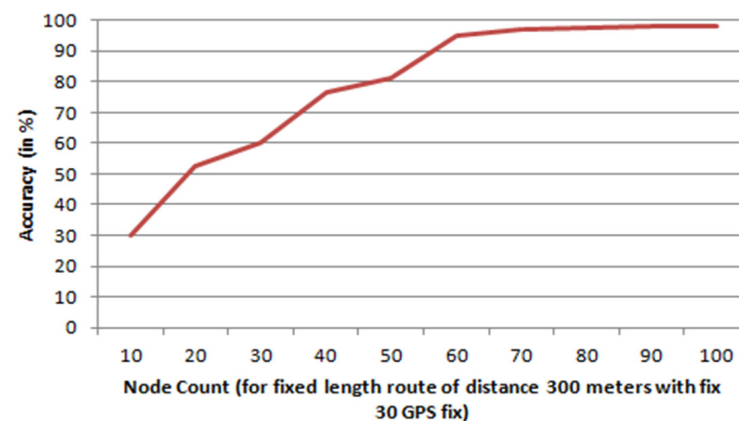

Figure 7. Accuracy analysis of REN system based upon accuracy vs. node count (for road length of 300 m) with 30 GPS fix.
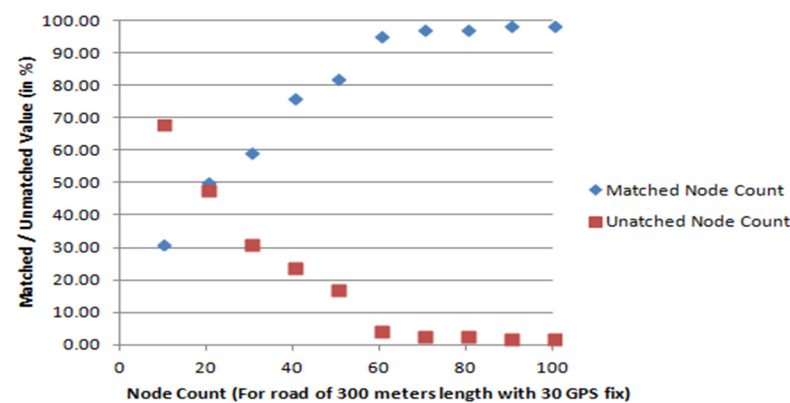

Figure 8. Accuracy analysis of REN system based upon MaUC vs. node count (for road length of 300 m) with 30 GPS fix.

Road node density was set between 0.033 and 0.333 nodes per meter to analyse the influence of node density on mapping accuracy. For the considered track of 300 m, the maximum accuracy to map 30 GPS points was 98.3 per cent, and the minimum accuracy was 30.1 per cent. Similarly, the maximum and minimum value of RMSE is 0.54 and 0.127, respectively. The number of road nodes was lower and far apart in sparse areas (due to low node density). As a result, GPS locations were mapped to nodes that were far away, resulting in inaccurate mapping.

Similarly, road node density was sufficiently good for the dense area, resulting in better GPS location mapping to road nodes. In this experiment, it was also observed that after a certain node density, accuracy and RMSE remain constant, and after that, increasing node density has no significant influence on the mapping process' accuracy. This accuracy behavior will remain the same for the different types of GPS.

If we make the node density very dense, then instead of improving the accuracy, it results in inappropriate results and poor performance. According to Figure 8, it is observed that after the threshold level of node density, MaUC provides improved results. Therefore, for good mapping, the track must have a sufficient number of nodes. If the track has fewer nodes which are far from each other, it creates wrong mapping and poor matching results.

To demonstrate the usefulness of the REN data storage approach for navigation, it was compared with OpenStreetMap (OSM). Figures 9 and 10 shows the comparison of REN data storage approach and OSM dataset based on the accuracy and storage size. To analyse the accuracy, the road was selected, and the navigation process was performed using both datasets. According to this analysis, it was observed that the REN approach gives approximately the same accuracy for both datasets. In some ways, the REN approach has a better result. The probable reason is that the OSM dataset has data-quality issues and location identification provides incorrect mapping due to the low quality of data.
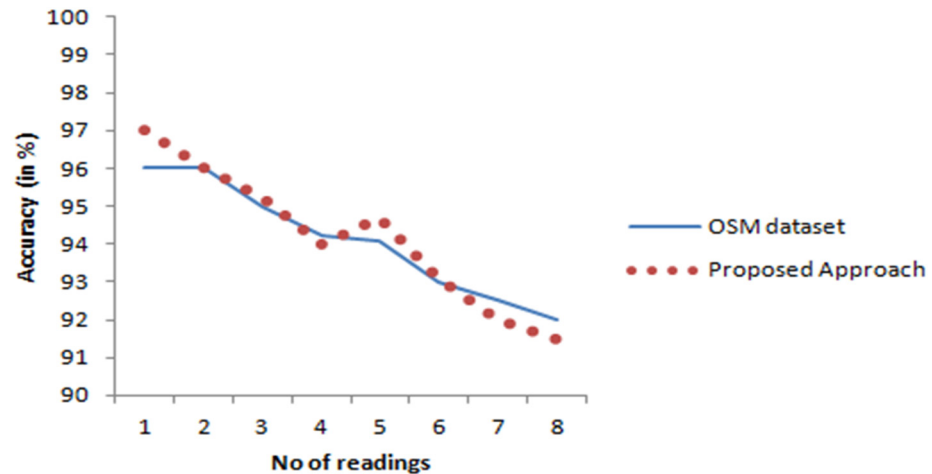

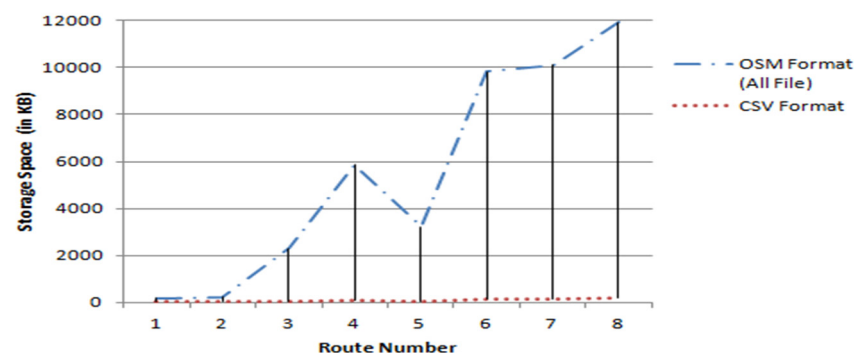**Figure 9.** Comparison of OSM and REN system based on achieved accuracy.

Figure 10. Comparison of OSM and REN system based on storage space.

Table 4 provides the comparison of the proposed REN with already existing technologies. The REN system uses the concept of a device camera to replace the heavy shapes files of maps to visualise the navigation output. REN achieved maximum accuracy of 99 per cent with RMSE of 13.2. At the same time, the minimum value of RMSE is 0.113. According to Table 4, AR-based navigation systems use an RCB-D camera, augmented reality and SLAM system for localisation and navigation in hybrid maps. This system attained maximum accuracy of 99 percent in an indoor area (in a room) [45]. Our proposed system also attained 99 percent accuracy in an outdoor area without using the GUI-based map information. The REN system was implemented using the road's basic location attributes, and these basic features are present for each and every road. Due to the requirement of general geospatial properties such as location information and turn identification, this case study can be implemented for other roads also.

Table 4. Comparison of proposed system with state-of-the-art techniques.

Algorithm	Accuracy	Error	Key Features	Reference
ORB-SLAM	93 percent	46.58 and 1.59 m are the maximum and minimum RMSE	Uses ORB features to perform tracking, mapping, localisation and loop closing.	[25]
GraphiumMM	93.1 Percent	Minimum RMSE is 0.12 and maximum RMSE is 0.38	Graphical and Topological features were used for localisation	[34]
AR based Navigation	99 percent for indoor environment	4 percent error rate	RCB-D camera, Augmented reality and SLAM system was used to provide the localisation and navigation in hybrid maps.	[45]
Proposed navigation system	Maximum accuracy of 99 percent and minimum accuracy is 96 percent	0.113 and 0.17 are the minimum and maximum root mean square error values.	Uses CSV files as source of reference data. Device camera, A Star algorithm and NNS are used for the localisation, routing and result visualizaton	

Figure 11 shows the CPU usage trends of the REN system (Figure 11a) and OSM-based navigation application (Figure 11b). In Figure 10, the blue line shows maximum CPU frequency and for the selected system it was 41 percent, the green line shows CPU utilisation of system process and routes and other services, the orange line shows CPU usage of the selected navigation applications (OSM-based or REN). It was observed that OSM-based navigation requires more CPU than the REN system. According to Figure 11, the CPU usage of the REN system is 29 percent less than the OSM-based system.

Similarly, memory usage is shown in Figure 12. Memory usage and memory faults for the REN system were less in comparison to an OSM-based navigation application. Memory usage for OSM based navigation application were more in comparison to the REN system. From the memory usage trends, it was observed that REN system have 35 percent less memory requirements in comparison to OSM-based applications.
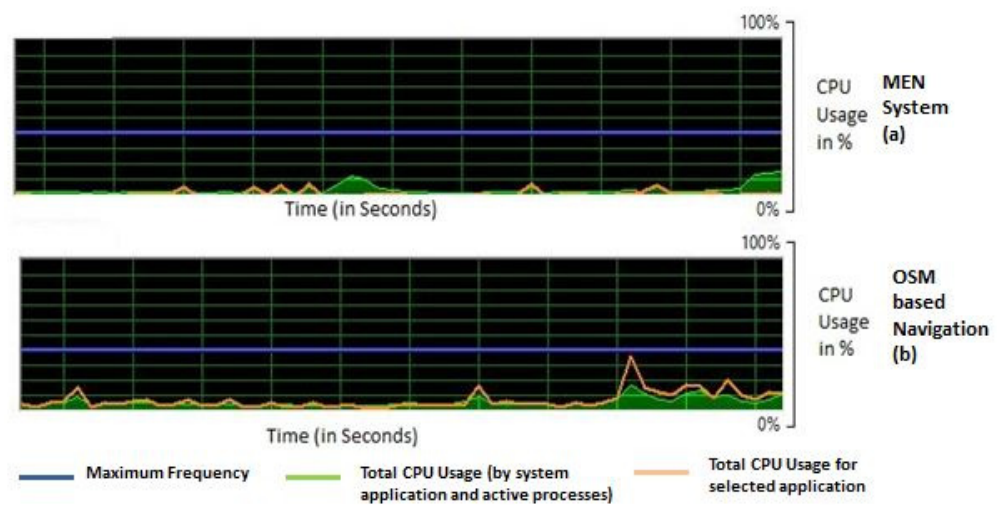
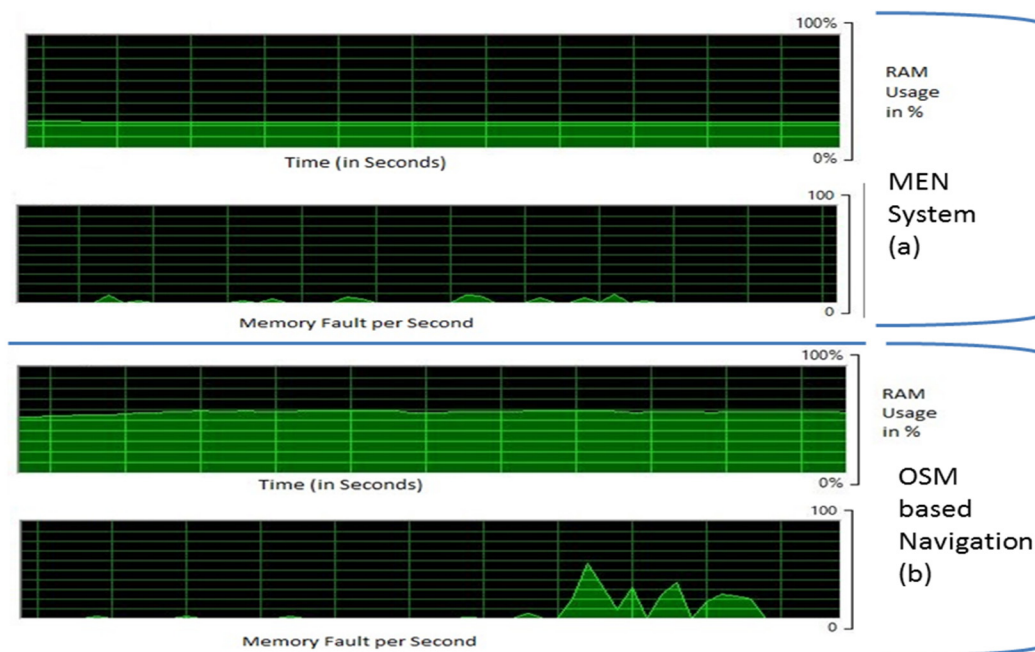**Figure 11.** CPU-usage-based analysis of OSM-based application and MEN system.

Figure 12. Memory utilisation and fault-based analysis of OSM based application and MEN system.

6. Future Scope and Conclusions

Mobile devices are used for many daily activities and have become an integral part of our life. Almost all desktop-based activities can be operated through the mobile phone. We use mobile phones for essential communication and complex activities such as banking, entertainment, art, audio/visual development, navigation, and searching. All these activities use large storage space and processing capacities. Therefore, in this study, we attempted to optimally use the mobile phone's storage space and processing power for a very frequently used application, i.e., a navigation application. The experiment was conducted in different traffic conditions and speed limits between 5–80 km/h. The proposed methodology can also be applied to other roads with a different speed limit.

In this study, we proposed a resource-efficient navigation system, which optimally uses device resources and storage for navigation purposes. In the proposed system, instead of using a heavy spatial dataset for navigation, a crowdsourcing-based light CSV dataset was used for reference data. The CSV dataset comprises only the topological and geometrical

information of the map. In contrast to the existing dataset (such as OSM), the REN system does not use shapefiles for the graphical interface. A device camera was used to augment the real-life navigation scenario to provide the graphical interface. The proposed system used the K nearest neighbour algorithm with the Kd tree and support vector machine for location identification. For finding the shortest distance, A star algorithm in recursive mode was used. Further, a case study was performed to test the accuracy and performance of the REN system. To validate the performance and accuracy of the proposed system, a comparison analysis is performed between REN and OSM datasets. The comparison shows that the REN system has approximately the same accuracy as the OSM dataset provides for navigation.

Apart from the accuracy, the REN system storage scheme requires very little space compared to the offline OSM dataset. Due to requiring less storage space, the REN system provides better processing capabilities and optimal resource utilisation. According to the experiments performed, the REN system requires less CPU and memory requirements. Further, the REN system ensured a maximum accuracy of 99 percent with the lowest RMSE of 0.11. REN system can work effectively with spatial data in CSV files format. If spatial data is presented in an OSM format or in graphical shapes file format, then the REN system will not work and requires some additional data processing mechanism. In the future, additional spatial data conversion mechanisms can be added to the REN system to provide compatibility with different spatial data formats.

References

1. Papagiannakis, G.; Singh, G.; Magnenat-Thalmann, N. A survey of mobile and wireless technologies for augmented reality systems. *Comput. Animat. Virtual Worlds* **2008**, *19*, 3–22. [CrossRef]
2. Wiley, J.; Pan, X.; Hamilton, A.F.d.C. Why and how to use virtual reality to study human social interaction: The challenges of exploring a new research landscape. *Br. J. Psychol.* **2018**, *109*, 395–417. [CrossRef]
3. Panzieri, S.; Pascucci, F.; Ulivi, G. An outdoor navigation system using GPS and inertial platform. *IEEE/ASME Trans. Mechatronics* **2002**, *7*, 134–142. [CrossRef]
4. Maddison, R.; Mhurchu, C.N. Global positioning system: A new opportunity in physical activity measurement. *Int. J. Behav. Nutr. Phys. Act.* **2009**, *6*, 73. [CrossRef]
5. Colpaert, P.; Abelshausen, B.; Melendez, J.A.R.; Delva, H.; Verborgh, R. Republishing OpenStreetMap's roads as Linked Routable Tiles. In Proceedings of the European Semantic Web Conference, Portorož, Slovenia, 2–6 June 2019; pp. 13–17.
6. Singh, S.; Singh, J. Intrinsic Parameters based Quality Assessment of Indian OpenStreetMap Dataset using Supervised Learning Technique. In Proceedings of the 2020 Indo—Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN), Rajpura, Punjab, India, 7–15 February 2020; pp. 52–57.
7. Singh, S.; Singh, J. Analysis of GPS Trajectories Mapping on Shape Files Using Spatial Computing Approaches. In Proceedings of the International Conference on Big Data Analytics, Sonepat, India, 15–18 December 2020; pp. 91–100.
8. Sehra, S.S.; Singh, J.; Rai, H.S. Assessing OpenStreetMap Data Using Intrinsic Quality Indicators: An Extension to the QGIS Processing Toolbox. *Futur. Internet* **2017**, *9*, 15. [CrossRef]

9.   Sehra, S.S.; Singh, J.; Rai, H.S. Assessing the Topological Consistency of Crowdsourced OpenStreetMap Data. *Hum. Comput.* **2014**, *1*, 265–280. [CrossRef]

10. Sehra, S.S.; Rai, H.S.; Singh, J. Quality Assessment of Crowdsourced Data against Custom Recorded Map Data. *Indian J. Sci. Technol.* **2015**, *8*, 1–6. [CrossRef]

11. Zacharopoulou, D.; Skopeliti, A.; Nakos, B. Assessment and Visualization of OSM Consistency for European Cities. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 361. [CrossRef]

12. Pazoky, S.H.; Pahlavani, P. Developing a multi-classifier system to classify OSM tags based on centrality parameters. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *104*, 102595. [CrossRef]

13. Geisberger, R.; Sanders, P.; Schultes, D.; Delling, D. Contraction hierarchies: Faster and simpler hierarchical routing in road networks. In Proceedings of the International Workshop on Experimental and Efficient Algorithms, Provincetown, MA, USA, 30 May–1 June 2008; pp. 319–333.

14. Bolliger, P.; Scherrer, M. Analysis Platform for OpenStreetMap. PhD Thesis, FHO Fachhochschule Ostschweiz, Buchs, Switzerland, 2020.

15. Scott-Young, S.; Kealy, A. An intelligent navigation solution for land mobile location-based services. *J. Navig.* **2002**, *55*, 225–240. [CrossRef]

16. Wang, E.; Li, Y.; Sun, A.; Gao, H.; Yang, J.; Fang, Z. Road detection based on illuminant invariance and quadratic estimation. *Optik* **2019**, *185*, 672–684. [CrossRef]

17. Huang, H.; Gartner, G.; Krisp, J.M.; Raubal, M.; van de Weghe, N. Location based services: Ongoing evolution and research agenda. *J. Locat. Based Serv.* **2018**, *12*, 63–93. [CrossRef]

18. Teipel, S.; Babiloni, C.; Hoey, J.; Kaye, J.; Kirste, T.; Burmeister, O.K. Information and communication technology solutions for outdoor navigation in dementia. *Alzheimer's Dement.* **2016**, *12*, 695–707. [CrossRef] [PubMed]

19. Brahmbhatt, S.; Hays, J. Deepnav: Learning to navigate large cities. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5193–5202.

20. Li, Y.; Tong, G.; Sun, A.; Ding, W. Road extraction algorithm based on intrinsic image and vanishing point for unstructured road image. *Rob. Auton. Syst.* **2018**, *109*, 86–96. [CrossRef]

21. Bonin-Font, F.; Ortiz, A.; Oliver, G. Visual Navigation for Mobile Robots: A Survey. *J. Intell. Robot. Syst.* **2008**, *53*, 263–296. [CrossRef]

22. Mirowski, P.; Grimes, M.; Malinowski, M.; Hermann, K.M.; Anderson, K.; Teplyashin, D.; Simonyan, K.; Kavukcuoglu, K.; Zisserman, A.; Hadsell, R. Learning to Navigate in Cities Without a Map. *Adv. Neural Inf. Process Syst.* **2018**, *31*. [CrossRef]

23. Krajnik, T.; Majer, F.; Halodova, L.; Vintr, T. Navigation without localisation: Reliable teach and repeat based on the convergence theorem. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1657–1664.

24. Holmes, S.; Klein, G.; Murray, D.W. A square root unscented Kalman filter for visual monoSLAM. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 3710–3716.

25. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [CrossRef]

26. Klein, G.; Murray, D. Parallel tracking and mapping for small AR workspaces. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, 13–16 November 2007; pp. 225–234.

27. Hashim, H.A.; Eltoukhy, A.E.E. Landmark and IMU Data Fusion: Systematic Convergence Geometric Nonlinear Observer for SLAM and Velocity Bias. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 3292–3301. [CrossRef]

28. Figueroa-Gonzalez, J.; Khachaturov, G.; Martinez-Hernandez, J.M. Generation of Non-Coordinate Navigation Knowledge from a Flow of Input Views. *Res. Comput. Sci.* **2019**, *148*, 139–151. [CrossRef]

29. Ort, T.; Paull, L.; Rus, D. Autonomous vehicle navigation in rural environments without detailed prior maps. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 2040–2047.

30. Fan, T.; Cheng, X.; Pan, J.; Manocha, D.; Yang, R. Crowdmove: Autonomous mapless navigation in crowded scenarios. *arXiv* **2018**, arXiv:1807.07870.

31. Zhou, X.; Gao, F. Route Matching Research Based on Roadless Navigation Data Improvements on Hidden Markov Model. In Proceedings of the 3rd International Conference on Computer Science and Application Engineering, Sanya, China, 22–24 October 2019; pp. 1–6.

32. Campigotto, P.; Rudloff, C.; Leodolter, M.; Bauer, D. Personalized and Situation-Aware Multimodal Route Recommendations: The FAVOUR Algorithm. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 92–102. [CrossRef]

33. Hu, L.; Zhong, Y.; Hao, W.; Moghimi, B.; Huang, J.; Zhang, X.; Du, R. Optimal route algorithm considering traffic light and energy consumption. *IEEE Access* **2018**, *6*, 59695–59704. [CrossRef]

34. Rehrl, K.; Gröchenig, S.; Wimmer, M. Optimization and evaluation of a high-performance open-source map-matching implementation. In Proceedings of the The Annual International Conference on Geographic Information Science, Wageningen, The Netherlands, 9–12 May 2018; pp. 251–270.

35. Singh, S.; Singh, J.; Sehra, S.S. Genetic-inspired map matching algorithm for real-time GPS trajectories. *Arab. J. Sci. Eng.* **2020**, *45*, 2587–2603. [CrossRef]

36. Singh, J.; Singh, S.; Singh, S.; Singh, H. Evaluating the performance of map matching algorithms for navigation systems: An empirical study. *Spat. Inf. Res.* **2019**, *27*, 63–74. [CrossRef]
37. Singh, S.; Singh, J. Map Matching Algorithm: Empirical Review Based on Indian OpenStreetMap Road Network Data. *Int. Arab J. Inf. Technol.* **2022**, *19*, 143–149.
38. Singh, S.; Singh, J.; Singh, S. Mitigating Spoofed GNSS Trajectories through Nature Inspired Algorithm. *Geoinformatica* **2020**, *25*, 581–600. [CrossRef]
39. Singh, S.; Singh, J. Location Driven Edge Assisted Device and Solutions for Intelligent Transportation. In *Fog, Edge, and Pervasive Computing in Intelligent IoT Driven Applications*; Wiley: Hoboken, NJ, USA, 2020; pp. 123–147. [CrossRef]
40. Rajeev, S.; Wan, Q.; Yau, K.; Panetta, K.; Agaian, S.S. Augmented reality-based vision-aid indoor navigation system in GPS denied environment. *Mob. Multimed. Image Process. Secur. Appl.* **2019**, *10993*, 143–152. [CrossRef]
41. Joshi, R.; Hiwale, A.; Birajdar, S.; Gound, R. Indoor Navigation with Augmented Reality. *Lect. Notes Electr. Eng.* **2020**, *570*, 159–165. [CrossRef]
42. Phupattanasilp, P.; Tong, S.R. Augmented Reality in the Integrative Internet of Things (AR-IoT): Application for Precision Farming. *Sustainability* **2019**, *11*, 2658. [CrossRef]
43. Choi, J.; Lee, K.; Kim, H.; An, S.; Nam, D. Classification of Inter-Urban Highway Drivers' Resting Behavior for Advanced Driver-Assistance System Technologies using Vehicle Trajectory Data from Car Navigation Systems. *Sustainability* **2020**, *12*, 5936. [CrossRef]
44. Chidsin, W.; Gu, Y.; Goncharenko, I. AR-Based Navigation Using RGB-D Camera and Hybrid Map. *Sustainability* **2021**, *13*, 5585. [CrossRef]
45. Amirian, P.; Basiri, A.; Gales, G.; Winstanley, A.; McDonald, J. *The Next Generation of Navigational Services Using OpenStreetMap Data: The Integration of Augmented Reality and Graph Databases*; Springer: Berlin/Heidelberg, Germany, 2015.
46. Pereira, F.C.; Costa, H.; Pereira, N.M. An offline map-matching algorithm for incomplete map databases. *Eur. Transp. Res. Rev.* **2009**, *1*, 107–124. [CrossRef]
47. Caruso, D.; Eudes, A.; Sanfourche, M.; Vissière, D.; le Besnerais, G. A Robust Indoor/Outdoor Navigation Filter Fusing Data from Vision and Magneto-Inertial Measurement Unit. *Sensors* **2017**, *17*, 2795. [CrossRef] [PubMed]
48. Tyagi, N.; Singh, J.; Singh, S. Sensor Based Wearable Devices for Road Navigation. *ECS Trans.* **2022**, *107*, 8863–8869. [CrossRef]
49. el Najjar, M.E.; Bonnifait, P. A road-matching method for precise vehicle localization using belief theory and Kalman filtering. *Auton. Robot.* **2005**, *19*, 173–191. [CrossRef]
50. Alvarez, J.M.; Gevers, T.; Lopez, A.M. Vision-based road detection using road models. In Proceedings of the 2009 16th IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, 7–10 November 2009; pp. 2073–2076.
51. Tennety, S.; Kumar, M. Autonomous navigation in unknown road-like environments using support vector machines. In Proceedings of the ASME 2010 Dynamic Systems and Control Conference, Cambridge, MA, USA, 12–15 September 2010; pp. 727–733.
52. Cao, K.; Wang, L.; Zuo, Z.; Sun, X. A Map Matching Algorithm Combining Twice Gridding and Weighting Factors Methods. In Proceedings of the International Symposium for Intelligent Transportation and Smart City, Shanghai, China, 9–11 May 2019; pp. 63–73.
53. Xiao, X.; Liu, B.; Warnell, G.; Stone, P. Motion planning and control for mobile robot navigation using machine learning: A survey. *Auton. Robot.* **2022**, *46*, 569–597. [CrossRef]
54. Tang, G.; Tang, C.; Claramunt, C.; Hu, X.; Zhou, P. Geometric A-Star Algorithm: An Improved A-Star Algorithm for AGV Path Planning in a Port Environment. *IEEE Access* **2021**, *9*, 59196–59210. [CrossRef]
55. He, Z.; Liu, C.; Chu, X.; Negenborn, R.R.; Wu, Q. Dynamic anti-collision A-star algorithm for multi-ship encounter situations. *Appl. Ocean Res.* **2022**, *118*, 102995. [CrossRef]
56. Li, J.; Ni, C.; He, D.; Li, L.; Xia, X.; Zhou, X. Efficient kNN query for moving objects on time-dependent road networks. *VLDB J.* **2022**, 1–20. [CrossRef]