



Article

Development of Multilayer-Based Map Matching to Enhance Performance in Large Truck Fleet Dispatching

Ching-Yun Mu ¹, Tien-Yin Chou ¹ , Thanh Van Hoang ^{1,*} , Pin Kung ², Yao-Min Fang ¹, Mei-Hsin Chen ¹ and Mei-Ling Yeh ¹

¹ GIS Research Center, Feng Chia University, Taichung 40724, Taiwan; jackie@gis.tw (C.-Y.M.); jimmy@gis.tw (T.-Y.C.); frankfang@gis.tw (Y.-M.F.); ivy@gis.tw (M.-H.C.); milly@gis.tw (M.-L.Y.)

² SkyEyes GPS Technology Co., Ltd., Taichung 40667, Taiwan; kevin@skyeyes.tw

* Correspondence: van@gis.tw; Tel.: +886-42451-6669

Abstract: Spatial information technology has been widely used for vehicles in general and for fleet management. Many studies have focused on improving vehicle positioning accuracy, although few studies have focused on efficiency improvements for managing large truck fleets in the context of the current complex network of roads. Therefore, this paper proposes a multilayer-based map matching algorithm with different spatial data structures to deal rapidly with large amounts of coordinate data. Using the dimension reduction technique, the geodesic coordinates can be transformed into plane coordinates. This study provides multiple layer grouping combinations to deal with complex road networks. We integrated these techniques and employed a puncture method to process the geometric computation with spatial data-mining approaches. We constructed a spatial division index and combined this with the puncture method, which improves the efficiency of the system and can enhance data retrieval efficiency for large truck fleet dispatching. This paper also used a multilayer-based map matching algorithm with raster data structures. Comparing the results revealed that the look-up table method offers the best outcome. The proposed multilayer-based map matching algorithm using the look-up table method is suited to obtaining competitive performance in identifying efficiency improvements for large truck fleet dispatching.

Keywords: spatial information technology; large truck fleet dispatching; multilayer-based map matching algorithm; spatial data structure; dimension reduction technique; spatial data mining



Citation: Mu, C.-Y.; Chou, T.-Y.; Hoang, T.V.; Kung, P.; Fang, Y.-M.; Chen, M.-H.; Yeh, M.-L. Development of Multilayer-Based Map Matching to Enhance Performance in Large Truck Fleet Dispatching. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 79. <https://doi.org/10.3390/ijgi10020079>

Academic Editors: Wolfgang Kainz and Giuseppe Borruo

Received: 27 November 2020

Accepted: 9 February 2021

Published: 17 February 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Spatial information technology (SIT) has been extensively applied to vehicle navigation. Intelligent transport systems (ITS) have been successfully applied to transportation management, and the map matching algorithm plays an important role in such systems [1–5]. The efficiency and accuracy of map matching have been discussed for many years. Large trucks (i.e., those over fifteen tons) with freight containers, and therefore, large turning circles are commonly used in fleet dispatching. Although previous studies have provided turning radius studies for general vehicles [6,7], the driving tracks of large trucks are different from those of ordinary vehicles. It is difficult for the map matching procedure to efficiently handle situations such as when a driver makes a U-turn on a complex and narrow road, or when dealing with large trucks using large-scale positioning data, which can reduce computing accuracy in determining their positions. In general, large fleet positioning is monitored by global positioning systems (GPS), which have rapidly gained importance in accurately locating the position of large trucks in logistics applications. Not all roads and road sections are suitable for a large truck fleet and, since actual road networks are highly complex, it is very important to be able to display real-time location information. Improvements in fleet dispatching are therefore imperative and in this paper we propose a new, more efficient system to assist vehicle navigation and to rapidly determine the positions of large truck fleets.

Map matching involves placing the positions from a GPS on a digital map. Many studies utilize computers to perform such map matching procedures, which use coordinate computations received by the GPS tracking information from the vehicles that correspond to the road on the map. Moreover, the map matching method involves using the road network information in a digital map as the classification basis for pattern recognition to correct the GPS positioning errors according to the identified results [1,2]. Obtaining accurate large-scale location data using currently available technology is the first step toward determining the most appropriate driving route. In the context of location-based services (LBS), this paper employed SIT in ITS to collect large-scale data from various sensors in order to handle the routing problems of large trucks.

Geographic information systems (GIS) have developed using projection technology to convert three-dimensional data into two-dimensional maps. Since map projection is conducted using three-dimensional space, the procedures for coordinate conversion used by projection technology can take considerable time. To limit this timeframe, it is necessary to complete the procedure of converting all coordinates using a map matching algorithm. GPS point data of maximum size, depending on the amount of data, are continuously received for all processing operations [5,8]. Logistics services are now commonly employed and smart transportation management has emerged, which uses state-of-the-art fleet management for automatic vehicle dispatching, together with GIS, which provides real-time information on the actual road network. Intelligent fleet management and dispatching involves an interconnected information system that is applied to allow for more effective completion of logistics tasks. With the rapid development of information and communication technology (ICT), the Internet of Vehicles (IoV) has become the most popular smart transportation tool for information collection from sensors and the analysis of vehicle dispatching data to provide full services for intelligent transportation. With the ongoing increase in road transport globally, effective fleet management has thus become imperative. Fleet management systems use computer-based technology to handle large-scale datasets for analysis [2,3,5] and combine SIT with ICT for vehicle dispatching. This system needs to be able to continuously monitor the fleet location and frequently match the position to ensure that the vehicles operate at optimum capacity. Large fleet dispatching strives for efficiency and to offer the most appropriate fleet management services. However, complex road networks result in distance calculation errors so the calculated precise position of the fleet may not always be accurate. As a consequence, many studies have been conducted on position error correction using map matching algorithms.

Probe vehicles (or floating vehicles) are now used to collect floating data to obtain real-time road information and can also be used in a variety of effective ways with ITS data. Probe vehicles communicate at a fixed frequency with the control center and track the vehicles' positions on the road. These large-scale data are immediately used for reception, converting different types of data structures into useful traffic information for processing the map matching algorithm [9–11]. Data structures include vector-based and raster-based formats for the map matching algorithm. The processed spatial data can be converted into both data formats, while data from different structures can be converted into each other [5]. Therefore, in this paper, we used the probe vehicle concept in large truck fleet management. Traffic information was collected from the current traffic situation, and the driving time was also provided for each segment of the road to facilitate urgent dispatching.

In general, many types of traffic information can be displayed in digital road networks. Map matching algorithms focus on position error correction and efficiency improvements to show the current positions of the vehicles. However, most studies on map matching algorithms discuss position accuracy. Moreover, the road network is becoming increasingly more complex. For the practical application of transportation, road network planning affects the performance of map matching which, in turn, influences the paths of large truck fleet dispatching. In this paper, we present a novel method to develop an effective map matching application. With limited resources to upgrade the hardware and software for improving the map-matching algorithm to implement map positioning, two key factors

including time and money (cost) had to be considered in this study. The aim was to enhance the efficiency of large fleet dispatching, provide efficient map positioning based on large fleet data volumes, and solve the relevant big data problems to improve computational efficiency.

The rest of this paper is organized as follows. Section 2 provides a brief review of map matching algorithms and their spatial data structure. In Section 3, we develop the multilayer-based map matching approach. Section 4 provides comparison results and analysis using our presented multilayer-based map matching algorithm with two different types of data structures. Section 5 reflects on the results, on the basis of which we present the most important discussion and conclusions.

2. Algorithms and Spatial Data Structures in Geographic Information Systems (GIS)

GIS is commonly applied to vehicle management, and vehicle navigation for logistics tasks has been a research interest for many years [2–5]. Unlike large truck fleets, typical vehicle dispatching problems have been solved in various studies, and map matching algorithms have drawn attention in transportation management for intelligence logistics. Statistical methods have also been used for the analyses of vehicle management with GIS applications. More spatial data mining information, statistical models, classification methods, and spatial regression analyses are found in [8,12–14].

2.1. Map Matching Algorithms

Many map matching algorithms are used in navigation studies for matching GPS positions and many studies have been conducted on position error corrections. For positioning deviations, many researchers have extended existing map matching algorithms to solve the imprecise positioning problems and adjust for errors. Geometric information can use point-to-point, point-to-curve, or curve-to-curve matchings [15], and these three methods have been extended to develop map matching studies for transportation and other areas to develop efficiency in spatial data modeling, road network matching, and visualization for transport in a range of applications [16–18].

The development of transportation and SIT are closely interrelated. Map matching approaches are arranged into four main categories: geometric, topological, probabilistic, and advanced algorithms [4,19,20]. Velaga and Quddus [21] improved map matching performance using error detection and correction. Quddus et al. [13] employed statistical analysis to compare the performance of geometric, probabilistic, and fuzzy logic map matching algorithms. For more statistical applications, the probabilistic method is based on the historical matching of results to match the current position of the vehicles. Probability and statistics were used to build error models by receiving GPS data, and statistical methods were applied for SIT. In order to improve matching position accuracy, Pfoser et al. [19] used a fuzzy map matching method to handle uncertainty in a fleet management system. Luo et al. [22] proposed a hidden Markov model (HMM) map matching method to improve mobile phone positioning efficiency and accuracy and made comparisons using GPS data to yield better performance and demonstrate that such models could help to resolve road matching problems.

Recently, OpenStreetMap (OSM) has been extensively discussed in many studies and applied to achieve accuracy. OSM is a type of digital map that can be freely edited to revise the deviation between the trajectory of positioning and actual roads. Liu et al. [23] provided buffer radius calculations to update a new road map with OSM data. Fan et al. [24] applied polygon matching as a map matching approach for OSM road networks, providing high precision and matching efficiency. Brovelli and Zamboni [25] applied the map matching method to verify the spatial precision of point locations and to detect position errors in building footprints using OSM vector maps. To integrate the road network information, the above methods search all OSM road combinations around the area of vehicle tracks to match a combined route, and then use the combined road with the best match as the

current driving route. Since road networks feature various complicated combinations in urban areas, most studies have focused on positioning accuracy.

Other related approaches seek to enhance positional accuracy using different combinations of points, lines, and curves. White et al. [1] calculated the error of a GPS position point and road network using curve-to-point distance, curve-to-curve distance, and curve-to-curve angle by applying the concepts in [15]. Geometry-based map matching can use low-sampling or high-sampling frequencies as the basis for improving the trajectory-based matching process to obtain greater positioning accuracy [1,26]; lower sampling frequencies for GPS signals may have large positioning errors and signal losses. The Fréchet distance is used in the curve-to-curve matching process, which determines the path for using trajectory data to perform positioning accuracy [27], and the global map matching algorithm includes a map matching method based on this distance. Local map matching is another method and uses point-to-line matching, although its matching accuracy is lower than that of global map matching. Liu et al. [22] proposed a local map matching method and an intersection segment matching method to enhance efficiency and accuracy in urban road networks. The extended approach involves adding a weighting method in the map matching algorithm, for example, as shown by Kuijpers et al. [20] and Quddus et al. [13]. Hashemi and Karimi [28] presented a weight-based map matching method using urban road network data, while Kong and Yang [29] studied complex networks with weighting scores to ensure accuracy, and various map matching methods were applied in [30]. However, these methods have proved successful in solving map matching issues for vehicles in general, but large truck fleets are more challenging.

ITS technology is deployed in the analysis of floating vehicle data (FCD), which are extracted using advanced intelligent transport technology. Li et al. [9] discussed using large-scale FCD for traffic surveillance. Given data volumes, cloud-computing technologies including Bigtable and MapReduce can be effectively adapted to handle FCD. Zheng et al. [31] used the MapReduce technology to enhance floating car efficiency with map matching methods. Chen et al. [11] presented a multi-criteria dynamic programming map matching algorithm that could be successfully used for matching FCD. This probe car technique, which collects FCD car data, can obtain significant road information from real-time traffic. Thus, floating car data can be used to update the road networks and explore points of interest.

As noted above, there are many approaches to resolve position error correction [21,22,26,27,32], but there are few studies on efficiency improvements for map matching methods in complex road networks and for large truck fleet management. In this paper, efficiency improvement is of vital concern for practical large truck fleet management.

2.2. Spatial Data Structure

Spatial data structures comprising vector-based and raster-based formats have been well studied. These two formats have various advantages in different scenarios [2,5]. In recent years, spatial data structures and characteristics have become more sophisticated as data volume has increased, requiring higher levels of computing power. In response, various widely applied technologies have been adopted to handle large-scale spatial data [12,18,33–35].

The vector data structure provides a representation of real-world spatial information. The real-world landscape includes surface, ground, and underground areas along with various geographical features. Vector data are traced by points, lines, or polygons based on the spatial environment and geographical information, while the attribute data refer to the descriptive material, which is composed of text or numbers. Points can be positioned with x and y coordinates, while road networks can be represented by lines. Polygons are the most common type of spatial information, and a bounding box can represent a complicated spatial area. A bounding based on vector-based data is included in the GIS platform. Suppose that a target point (s) falls within a polygon, which means that the point is

captured. When s has a coordinate (x_s, y_s) , the area of the polygons of Equations (1) and (2), described in [5], is expressed as

$$X_{min} < x_s < X_{max}, \quad (1)$$

$$Y_{min} < y_s < Y_{max}, \quad (2)$$

where X_{min} , X_{max} , Y_{min} , and Y_{max} denote the eigenvalues of the vertices. A road network described on the basis of the vector-based format features a group of linked segments and nodes.

The raster-based format provides the spatial position for each grid by labelling the grid with a row and column serial number, and encodes each grid based on the property that fully represents all the geographic information. The actual area represented by a grid cell on the ground is the spatial resolution of the grid. The higher the resolution of the image, the greater the spatial recognition rate. Moreover, for raster-based applications, the digital differential analyzer (DDA) method can be used to capture the grid polygon boundary. Boundary algebra filling (BAF), which is based on the concept of integration, is an algorithm for a vector to raster conversion approach; moreover, the grid segment of the polygon boundary can be captured by BAF [5]. Common tree structures such as R-tree and Quadtree are discussed in [5,36]. R-tree is the solution used to overcome the limit that a node can only have two sub-nodes. Quadtree is a hierarchical data structuring technique that encodes space by dividing it into four quadrants until the property encoding value of the sub-quadrant becomes only one. Quadtree is a popular data structure and image compression method in the application of GIS. The Quadtree method provides a very useful data structure for many spatial database applications. Sun et al. [36] proposed a unique identification method using a Quadtree grid, and Yao and Li [18] listed other progressive techniques such as Hadoop architecture for Quadtree.

Data compression is a coding technique that reduces the amount of space required for data storage. It is an important technique to reduce the storage space of raster data. Since a large volume of raster data is collected from the GPS-based vehicle probe, a suitable approach for data compression is used to improve the efficiency of data processing. There are two main compression methods. Run-length encoding (RLE) is one application of data compression algorithms and offers highly efficient compression that can achieve one-dimensional data compression. The number of random samples is increased when the maps become too complex. The extended popular method is Quadtree Morton digital code (Quadtree MD code), which is used to achieve two-dimensional compression. Quadtree MD code is used to decrease the number of raster data retrieved and reduce the time need for spatial identification. For instance, RLE and Quadtree MD code can be used if the polygons do not have extremely slender shapes. The aim of mode selection and coding is to achieve efficient compression [5,36].

The vector-based format requires complex mathematical procedures, while the raster-based format does not use difficult mathematical equations or programming languages. In general, the vector-based format has high accuracy but in the case of large fleet management, it is necessary to consider the processing costs, which may be high due to data volumes. Digital maps based on raster data can also be useful over extended periods if constructed appropriately.

3. Research Methodology

Since the return of coordinate data from vehicles is continuous, the fleet dispatching management platform may quickly become overloaded. This paper uses a point-to-polygon method to deal with the high volumes and rates of location GPS data returned by large truck fleets. We also used the same technology to process large fleet location data and found that multiple layer grouping combinations can be used to reduce loading. Figure 1 shows four steps of the process to achieve efficiency improvements with a new multilayer-based algorithm for map matching.

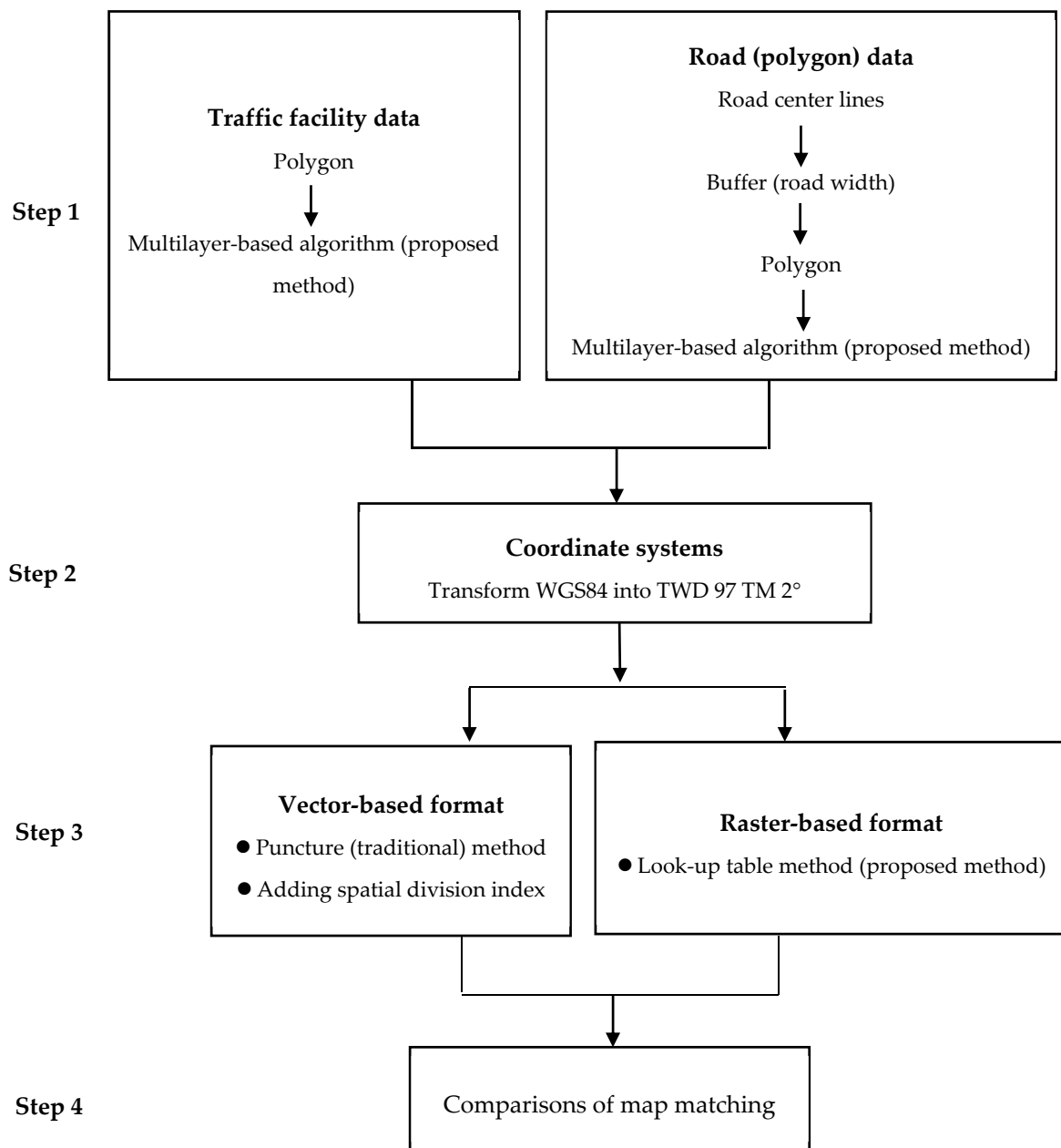


Figure 1. Flowchart for the multilayer-based map matching process.

Step 1: Initially, there are two procedures for traffic facility and road data. The traffic facility can be constructed as a polygon. For road data, center lines using vector data are collected from ArcGIS [37]. These center lines were used to produce buffers, and then the road can be constructed as a polygon. For the proceeding multilayer-based map matching, we constructed a large fleet management system based on a structured query language (SQL) database for several segments from the road network data to process map matching.

Step 2: Truck fleet data were collected from onboard GPS and plane coordinates of TWD 97 TM 2° obtained for sample testing.

Step 3: The puncture (traditional) method using vector data that has more relevant but complex structures was used to partition and compose the digital map. Thereafter, we used an improvement method, adding a spatial division index, which still requires complex mathematical procedures. Since more large-scale data are used for reception and cause a computational bottleneck, we proposed a look-up table method using raster data to improve efficiency.

Step 4: A comparison of the methods based on the two data types was made to provide guidance for large truck fleet management.

3.1. Multilayer-Based Map Matching

To construct the framework of the multi-layer structure in Step 1, we assumed that the digital map of the national road network took a single-line network in a vector-based format based on the road center lines. The road center lines were used to complete the conversion procedures of the segments. These procedures include road data stratification, defining the segments and facility ranges, and determining the surrounding point distance.

In the past, road networks were described according to each section of the road but were not considered altogether. Large truck fleets usually have common routes for dispatching. To enhance the technology of map matching efficiency in a fleet management system, we proposed a multi-layer method with road networks for large fleet dispatching. We suggest that the constraint of road networks can be divided into two categories, viz. traffic facilities and roads, and can be written as

$$QTY(GPS) = QTY(TF_i(GPS)) + QTY(R_j(GPS)), i \geq 0 \text{ and } j \geq 0, \quad (3)$$

where GPS is the quantity (QTY) of GPS data. TF denotes the quantity of GPS data for each traffic facility and there are i categories of TF . R denotes the quantity of GPS data for each road and there are j categories of R . Usually, traffic facilities have smaller categories than roads. It is very important to enhance efficiency when the large truck fleet data volume increases in size. Several layers are taken from road network information. For the planning of large truck dispatching routes, multi-layer selection can be associated with the road network. This study considered the road network as the interplay between traffic facilities and roads. Different combinations can affect the performance of the map matching algorithm. The multilayer-based map matching can have an important influence on the efficiency of the map matching algorithm; therefore, we proposed this approach to enhance the overall efficiency for determining map positioning.

3.2. Coordinate Conversion

In executing map matching, this study also used coordinate conversion to improve time efficiency. In Step 2, we used linear coordinate transformation to enhance map matching performance and used the area of coordinate conversion with an interval of 2 arcminutes for the transformation procedure. Here, we adopted the formulae for linear coordinate transformation as described in [38]; this is written as:

$$X = x_0 + a_\gamma(\gamma - \gamma_0) + a_\theta(\theta - \theta_0), \quad (4)$$

$$Y = y_0 + b_\theta(\theta - \theta_0) + b_\gamma(\gamma - \gamma_0), \quad (5)$$

where X and Y , respectively, denote the values of the plane coordinates; γ and θ are the values of the longitude and latitude; x_0 and y_0 are the initial values of the transformation of coordinate areas; γ_0 and θ_0 are the initial values of the longitude and latitude; and a_γ , a_θ , b_θ and b_γ are undetermined coefficients. When applying the transformation procedures, the values of a_γ , a_θ , b_θ , and b_γ may be different due to the location and size of the coordinate conversion area. a_γ and b_θ are the key conversion coefficients from the longitude and latitude to the X and Y coordinates. a_γ and b_θ have a proportional relationship. a_θ and b_γ are auxiliary transformation coefficients, where a_θ indicates an increase of the same latitude with longitude, and b_γ indicates an increase of the same longitude with latitude. Moreover, $\Delta\gamma$ and $\Delta\theta$ are the increments of longitude and latitude from the origin to the end of the coordinate conversion area. The ranges of four points, (γ_0, θ_0) , $(\gamma_0 + \Delta\gamma, \theta_0)$, $(\gamma_0 + \Delta\gamma, \theta_0 + \Delta\theta)$, and $(\gamma_0, \theta_0 + \Delta\theta)$, provide an approximately trapezoidal area. Since $\Delta\gamma$ and $\Delta\theta$ are very small, the longitude and latitude can be regarded as approximately parallel. There is a linear correspondence between the geographical coordinates of points and the increase in the longitude and latitude of points relative to (γ_0, θ_0) and the plane

coordinates. Moreover, the transformation errors can be controlled within a certain range for the coordinate conversion area [38,39]. Using the dimension reduction technique, linear transformation can simplify complicated conversion and enhance traditional coordinate transformation.

3.3. Comparative Methods

This study evaluates our proposed multilayer-based map matching algorithm under both spatial data structures, vector and raster (Step 3), in order to compare performance (Step 4). The point-in-polygon algorithm described in [40,41] was applied to verify performance efficiency using the puncture (traditional) method to determine whether the point falls within the polygon. In this scenario, the puncture method can be used with vector data to process the mapping between coordinates and map features and can be applied to map matching algorithms [39].

This paper extends the puncture method to multilayer complex road network design. Assume that ray \vec{AB} extends $A(x_1, y_1)$ to point $B(x_2, y_2)$, and $P(x_3, y_3)$ is the initial point to make a ray from left to right. For each edge of the polygon, the puncture method is used to determine whether P is on \vec{AB} or if P and \vec{AB} are intersected. We then distinguish this into four regions for the puncture method, as shown in Figure 2. The lines of P_1 , P_2 , and P_3 and \vec{AB} do not intersect, while P_4 and \vec{AB} are intersected. P_5 is on the ray \vec{AB} and P_5 is within the polygon. To provide all edges of the vertices on the polygon, the difference-product's y-coordinate is given by

$$y - \text{Value} = (y - y_i)(y - y_{i+1}), \quad (6)$$

where y is for P ; y_i is for A ; and y_{i+1} is for B . As stated above, the polygonal areas of A , B , and P can be shown as

$$S = \begin{vmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{vmatrix} = (x_1 - x_3)(y_2 - y_3) - (x_2 - x_3)(y_1 - y_3), \quad (7)$$

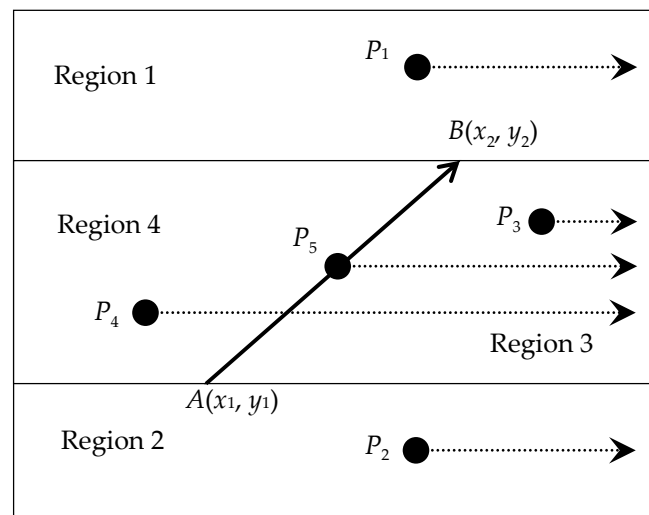


Figure 2. An example of the puncture method.

If $S > 0$, P is on the left of \vec{AB} ;
 If $S < 0$, P is on the right of \vec{AB} ;
 If $S = 0$, when $y - \text{Value} \leq 0$, P is on ray \vec{AB} . Otherwise, P is on the extended line of \vec{AB} .

A segment polygon was built in the continuous-node data table using the puncture method. A polygon is a set of node coordinates built on the feature. The proposed approach uses the polygon features as the road segments. This functions like an electronic fence, where the number corresponding to the feature is the segment code. The surrounding point coordinates are composed of the surrounding points on the perimeter line of the segments. The number of surrounding points is set according to the size of the segment region, and the density of the surrounding coordinates affects the fineness of the segments. This study used 50 m as the criterion for the collection of the surrounding points. When considering the length of a road segment, the goal is to make the large truck fleet travel at a normal speed on the road, where each return can be on a different segment. Performance bottlenecks may occur when applying the puncture method to process large amounts of data. In general, sophisticated mathematical and computational techniques are required for vector data structures and, when the data sources are themselves complex, it becomes difficult to solve the problems.

The second approach involves adding the spatial division index in the puncture method. In this way, a simple division operation can be used to quickly determine which point coordinates of the square block are located. This multilayer-based map matching algorithm uses a method with a spatial division index. However, large-scale data can seriously reduce the retrieval efficiency. If the spatial division index table is well established, it can more efficiently determine the improvements to the multilayer-based map matching.

According to the literature in [40], the authors concluded that the polygon in raster format using the grid method was more efficient due to a few mathematical computations. Therefore, we extended the idea and proposed a third approach that involves a look-up table method using a raster-based format. The raster data structure has the same square size as the grid, and no complicated mathematics are required. For the grid procedures, the DDA is considered to take the grids of the polygon boundary. The polygon boundary grid point of the extraction operation is shared among the segments. The DDA method uses the slope of a line segment to take a grid point on a line at a unit interval (the minimum grid size) to determine the corresponding integer value on the other axis of the closest line. The polygon boundary of the grid data can be extracted without using complicated trigonometric functions. BAF is used to determine the entire grid within the polygon. The whole polygon grids are achieved using DDA and BAF. Encoding and compression are the last steps needed to complete the raster-based procedure before the grid spatial database is ready for use.

The above three spatial data structure approaches can be used for comparing performance using multilayer-based map matching. Based on Equation (3), we constructed an equation to calculate the time as below:

$$T = \begin{cases} QTY(GPS)_{-S_{Pu}} \\ QTY(GPS)_{-S_{Pu}^{Ind}} & n_L < n < n_U, \\ QTY(GPS)_{-Lt} \end{cases} \quad (8)$$

where T denotes the computation time; QTY of GPS includes the QTY of traffic facilities and roads of GPS data; Pu is the puncture method; Ind is the puncture method with the spatial division index; S is based on Equation (7); Lt is the look-up table method; and n denotes the sample, which has lower bound (n_L) and upper bound (n_U) samples. In general, the sample is based on the scale of large truck fleet. This study aimed to use this multilayer-based method with different data structures to effectively achieve improvement for map matching under existing limited software and hardware resources.

4. Results and Analysis

This section illustrates the comparison results using multi-layer map matching algorithms. Based on spatial map matching of a complex road network, this study used the national electronic map road center single-line vector data produced by the GIS Research

Center, Feng Chia University (GIS.FCU), and processes the segments by buffers according to the width of the road. This method then utilizes the surrounding points on the perimeter of the road segments to generate the nodes of the road segments. We adopted different categories of segments in Taiwan's road networks for the case study. This study provides an applicable map matching algorithm and efficiently matches geographic coordinates to an actual route. The empirical data were actual GPS car driving records from Taiwan's large truck fleet dataset, which were collected from car kits on large trucks. The coordinates were taken from large trucks ready to travel, and the proportion of GPS coordinates was chosen to match the actual road for navigation.

We considered that traffic facilities consisted of interchange and service stations. Roads included national highways, express highways, provincial highways, county highways, district roads, and normal roads. Our large fleet management system included the segment data arranged as in Table 1. Traffic facilities and roads as eight-segment layers on Taiwan main island were collected from GPS instruments installed on large trucks. Table 2 shows that the quantity (QTY) of GPS data has one million empirical data points. Transmission was the fourth generation of mobile phone mobile communication technology standards (4G). The source data were collected from 35 trucks with eight working hours per day for one month, and the sampling interval was 30 s. Plane coordinates were used in the large fleet management system, thus type of orography was not considered in this study. Table 3 gives an example of an interchange, which includes one center point and 99 vertices from our system. A polygon was composed of 99 vertices. Figure 3 illustrates the example from Table 3 on a map using ArcGIS software. When a point falls within this polygon, the point information can be displayed in the caption field.

Table 1. Segment data.

Category	Total
Center point	441,287
Layer description	Number of vertices
Interchange	15,934
Service station	353
National highway	58,337
Express highway	39,495
Provincial highway	220,689
County highway	151,213
District road	525,619
Normal road	1,637,631
Total	2,649,271

Table 2. Data collection.

Source	Transmission	QTY of Trucks	Period	Interval	Total
GPS data	4G	35	1 month (8 h per day)	30 s	1,000,000

Table 3. An interchange example.

Category	Node_ID	X (TM2°)	Y (TM2°)	Caption	
Center point	1000238	320,518	2,776,740	Dahua System Interchange, National Highway No. 1, Qidu Dist., Keelung City	
Category	Order	Node_id	Node_sequence	X (TM2°)	Y (TM2°)
Vertices	1	1000238	0	320,631.54598	2,776,896.05615
	2	1000238	1	320,679.75089	2,776,909.23300
	3	1000238	2	320,729.08397	2,776,917.19750
	⋮	⋮	⋮	⋮	⋮
	99	1000238	98	320,611.60494	2,776,899.41189

**Figure 3.** Illustrative example of the interchange.

For the map matching comparison of the two data structure types, it is necessary to convert the geodetic coordinates to plane coordinates. For all GPS coordinates, the WGS 84 longitude and latitude geodetic coordinates can be immediately used via linear transformation formulae to transform them into Taiwan TWD 97 TM 2° plane coordinates. The scope of implementation and testing was based on Taiwan's main island. The scope ranged from 120° to 122° eastern longitude, 21°55' to 25°31' northern latitude, spanned south and north by 216 arcminutes, and crossed east and west by 120 arcminutes. According to 2 arcminutes sections, there were 60 × 108 coordinate conversion areas. The two orders in Table 4 are expressed as:

$$N_{\theta} = \left\lceil \frac{\theta - 7200}{2} \right\rceil, \quad (9)$$

$$N_{\gamma} = \left\lceil \frac{\gamma - 1315}{2} \right\rceil, \quad (10)$$

where N_{θ} is the longitude order, and θ denotes the longitude value. N_{γ} is the latitude order, and γ denotes the latitude value (units: arcminutes). Here, 120° was 7200 arcminutes, 21°55' was 1315 arcminutes, 2 denotes arcminutes sections, and square brackets denote Gauss notation.

Table 4. Two arcminute interval linear coordinate transformation.

$N_\gamma \backslash N_\theta$		N_θ		
		1	...	60
N_γ	1	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(1,1)}$...	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(60,1)}$
	2	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(1,2)}$...	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(60,2)}$
	\vdots	\vdots	...	\vdots
	107	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(1,107)}$...	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(60,107)}$
	108	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(1,108)}$...	$(X, Y, a_\gamma, a_\theta, b_\theta, b_\gamma)_{(60,108)}$

This paper used the two arcminute interval linear coordinate transformation formulae with six parameters, as shown in Table 4. The notations have already been described in Equations (9) and (10). There were 60 longitude serial numbers, and 108 latitude serial numbers, representing a total of 6480 points. The results showed that the conversion errors could be controlled within one meter, which represents the overall efficiency improvement of map matching.

For example, the linear transformation formulae employed in this study used two arcminute intervals to transform WGS84 coordinates into TWD 97, which is the commonly used coordinate projection system used in Taiwan. The transformation errors were within one meter, both latitudinally and longitudinally. The linear transformation computation time was about 1.737×10^{-5} s. For the traditional coordinate transformation using the GIS software tool, the computing time was 2.784×10^{-5} s [39]. Therefore, linear transformation was used in this study to enhance the efficiency of our large truck fleet case study.

Moreover, the space is divided horizontally and vertically over 5000 m. Using Taiwan's main island as the range, the origin coordinates are (135,000; 2,400,000) from the lower left corner coordinates (which are origin coordinates (135,000; 2,400,000)) to the upper right corner coordinates (355,000; 2,800,000), which can be divided into 44×80 square blocks. In the empirical operating procedure, the longitude and latitude coordinates are converted into TM 2° plane coordinates, and the plane coordinates are subtracted from the origin coordinates (135,000; 2,400,000). The coordinate values are divided by the division distance of 5000 m, and (X_i, Y_i) are index values using Equations (11) and (12). For example, assuming that X is 230,529 and Y is 2,621,656, the values of index code can be given as follows:

$$X_i = \left\lceil \frac{X - 135,000}{5000} \right\rceil = 19, \quad (11)$$

$$Y_i = \left\lceil \frac{Y - 2,400,000}{5000} \right\rceil = 44 \quad (12)$$

$$\text{Index}(X_i, Y_i) = (19, 44). \quad (13)$$

Furthermore, the raster-base format consists of horizontal encoding, vertical encoding, and Quadtree MD code. As most of Taiwan's roads run in a north and south direction, vertical encoding and compression obtain the best performance. The procedure for encoding and compression may take time, but it can be replaced by subsequent efficiency improvements since digital maps are not changed frequently.

For the output of Equation (8), the computing time (T) divided by 50 runs is the average processing time (i.e., the average case), which is calculated by the following formula:

$$\text{Average case} = \frac{\sum_{i=1}^{50} T_i}{50}. \quad (14)$$

For the outcomes, the performance of the best and worst case have been shown. The experiments included 100,000 samples with 50 runs, and the computational results are shown in Table 5. According to the performance of the individual segment layers for the puncture method, normal roads offered the worst results. All the values were large. An individual segment layer for the puncture method with the addition of a spatial division index showed significantly improved performance. The values for computational time

were between 24 and 1076 s. Adding the spatial division index was more efficient than using the puncture method, except for service stations.

Table 5. The individual segment layer using 100,000 samples with 50 runs.

Method	Layer Description	Computation Time (s)		
		Average Case	Best Case	Worst Case
Puncture method	Interchange	147.54	145	152
	Service station	17.82	17	19
	National highway	365.32	364	366
	Express highway	365.52	363	368
	Provincial highway	2007.22	1996	2012
	County highway	1391.02	1381	1429
	District road	4782.36	4717	4878
	Normal road	11,765.76	11,654	11,999
Puncture method supplemented with the spatial division index	Interchange	25.12	24	27
	Service station	27.66	26	32
	National highway	52.94	52	54
	Express highway	47.72	46	48
	Provincial highway	71.70	70	74
	County highway	62.48	60	64
	District road	172.70	171	173
	Normal road	1073.50	1072	1076
Look-up table method	Interchange	11.06	11	12
	Service station	11.08	11	12
	National highway	12.04	12	13
	Express highway	12.06	12	13
	Provincial highway	12.86	12	13
	County highway	12.84	12	13
	District road	13.64	13	14
	Normal road	16.80	14	18

According to this 50-run outcome, as Table 5 shows, the values for the normal road case were slightly higher than those of the other seven layers. The look-up table method yielded the excellent results listed in Table 5. The best case for the normal road using the look-up table method yielded good performance, from 11,654 down to 14 s. Since the quantity of service station had less data and the position distribution was scattered, service stations with a spatial division index may not be efficient. Figure 4 shows that the puncture method provided a steep increase from the county highway and included 100,000 samples with 50 runs. Overall, the look-up table method with individual segment layers could produce a more stable performance than the other two methods.

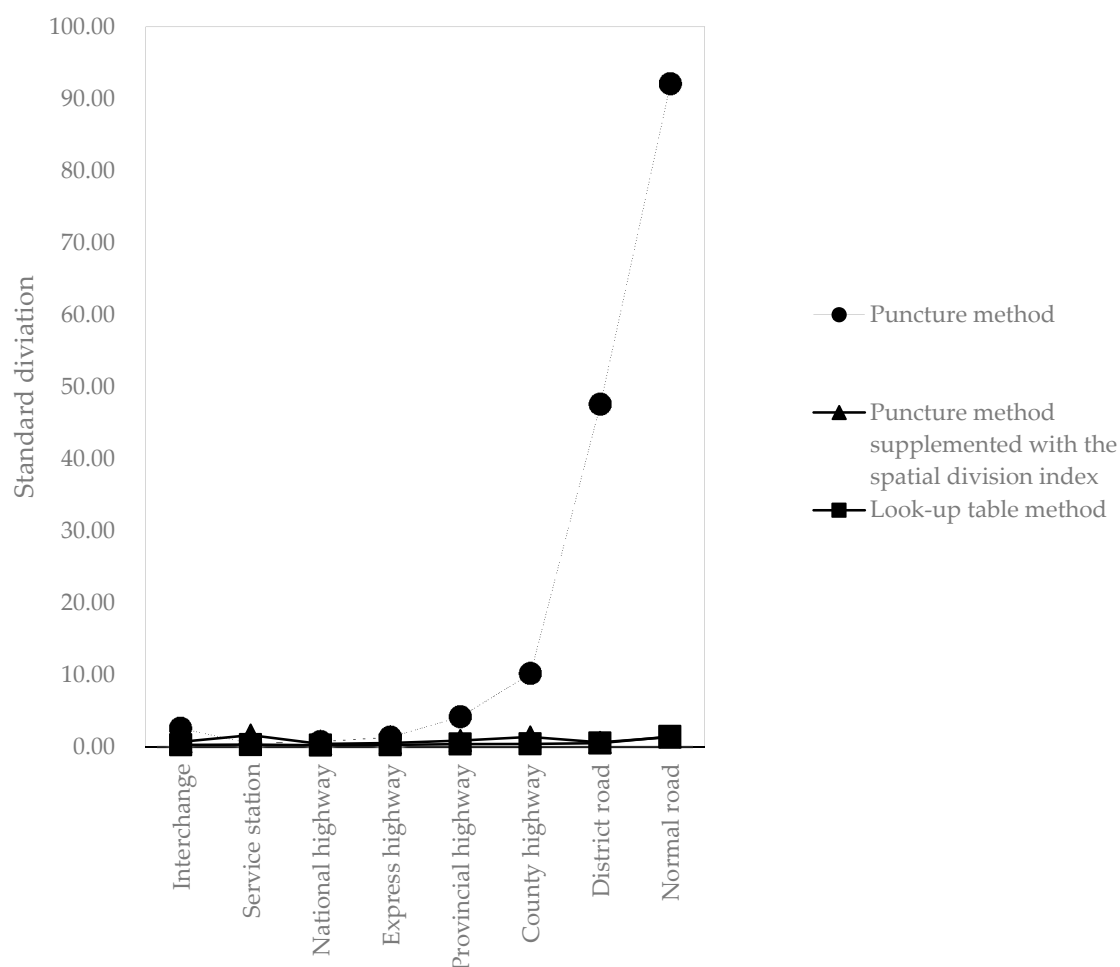


Figure 4. Standard deviations of three methods using 100,000 samples with 50 runs.

As stated above, if a normal road is chosen as the first layer, the computational process may take a great deal of time; thus, a normal road should not be chosen as the first layer. Moreover, once the individual computation of each layer is done, then the combination of all layers together may also be time-consuming. To improve efficiency, we grouped interchanges and service stations for traffic facilities and the other six layers for roads.

In Table 6, the experiments still featured 100,000 samples with 50 runs. To compare individual calculations with the groupings, Table 6 performed better than Table 5. The grouping of roads took more time than the grouping of traffic facilities. In this case, the puncture method reduced the efficiency of process execution and affected the output performance. The look-up table method still required much less computation time than the other two methods. The look-up table method also produced a stable performance. The three categories of standard deviations are summarized in Table 7. For traffic facilities and roads, the puncture method had larger standard deviations. Grouping traffic facilities and roads provided better results than using individual layer calculations. All eight-segment layers were grouped together (i.e., traffic facilities and roads). The puncture method provided a very large standard deviation value of 49.8420, whereas the other two methods had low standard deviations. Therefore, different grouping combinations can affect the efficiency of multilayer-based map matching in different ways. The category of roads is also more complicated than traffic facilities, so applying the puncture method with vector data may yield excess computing loads and delays. However, the grouping using the puncture method supplemented with the spatial division index and look-up table method can retain small standard errors of measurements.

Table 6. Two groups using 100,000 samples with 50 runs.

Method	Category	Average Case	Best Case	Worst Case	Range
Puncture method	Traffic facilities	151.9800	148	164	16
	Roads	2902.3200	2876	2931	55
Puncture method supplemented with the spatial division index	Traffic facilities	43.88	43	46	3
	Roads	222.40	221	224	3
Look-up table method	Traffic facilities	14.04	14	15	1
	Roads	16.66	15	18	3

Table 7. Standard deviation of the eight-segment layers for the three methods.

Standard Deviation			
Category	Puncture Method	Puncture Method Supplemented with the Spatial Division Index	Look-Up Table Method
Traffic facilities	5.5420	1.0999	0.1979
Roads	14.6377	0.8806	1.0616
Traffic facilities plus roads	49.8420	0.7887	0.8485

Computation time using all eight-segment layers together and the average processing time for every point was considered. The experiments outlined in Table 8 still featured 100,000 samples with 50 runs. As shown in Table 8, the puncture method took 2756.76 s, and its average processing time was 0.0275676 s. The computation time for the puncture method supplemented with the spatial division index was 217.52 s, and its average processing time was 0.0021752 s. The computation time of the look-up table method was 15.12 s, and its average processing time was 0.0001512 s. The puncture method supplemented with the spatial division index improved the efficiency by 12.67 times compared to the method without a spatial division index. Using the look-up table method with raster as the data structure along with the data compression technology improved the average map matching processing time by 14.39 times compared to the puncture method supplemented with the spatial division index and by 182.33 times compared to the puncture method. The computation time of the look-up table method was the shortest. The percentage of deviation (PoD) was between the best-solution (BS) and the solution-so-far (SSF):

$$\text{PoD} = \frac{BS - SSF}{SSF} \times 100\%. \quad (15)$$

Table 8. The comparisons of grouping eight-segment layers using three different methods.

	Puncture Method	Puncture Method Supplemented with the Spatial Division Index	Look-Up Table Method
Samples		100,000	
Computation time (s)	2756.76	217.52	15.12
Average processing time for every point (s)	0.0275676	0.0021752	0.0001512

Based on Equation (15), comparing other puncture methods with the puncture method supplemented with the spatial division index, the improvement was 92.11%; thus, the efficiency was significantly improved. This indicates that the spatial division index can enhance the efficiency of map matching. The highest improvement was 99.45% between

the puncture method and the look-up table method. In terms of average processing time, the puncture method with the spatial division index was the second-best approach. Thus, map matching based on a raster data structure using the look-up table method was more efficient than the other two methods. Figure 5 shows three samples of 10,000, 50,000, and 100,000, and illustrates that the line of the puncture method was higher than the other two lines when the sample increased. This study employed a statistical test method using the independent samples test [42] and arranged the values in SPSS software. Table 9 shows that the P -value for Levene's test was 0.000, which is statistically significant at a $P < 0.05$ level. Thus, the variances were not equal. The P -value of the independent T -test was 0.000, which is statistically significant at a $P < 0.05$ level. Therefore, the puncture method supplemented with the spatial division index and the look-up table method offer significant improvements.

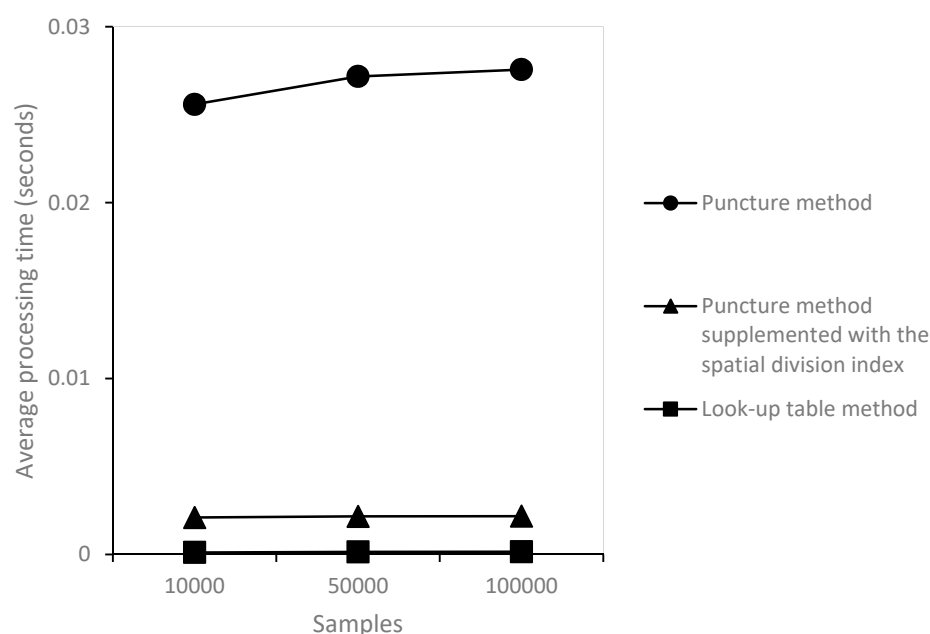


Figure 5. Performance of the vector and raster-based formats using the three different methods.

Table 9. Independent samples test.

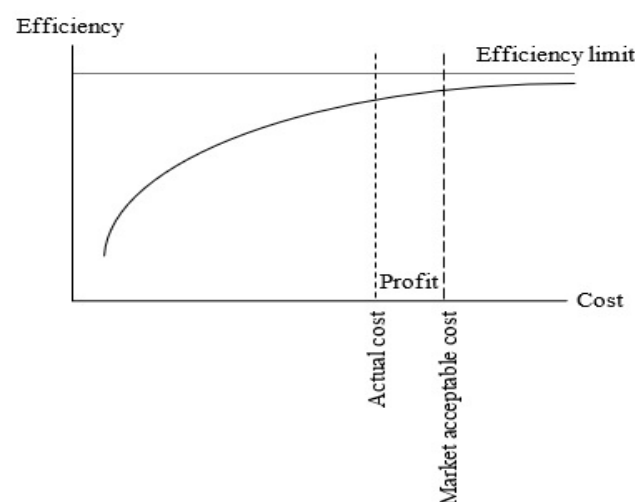
Independent Samples Test	Levene's Test		T-Test for Equality of Means		
	F-Value	P-Value	DF	T-Value	P-Value
Puncture method vs. puncture method supplemented with the spatial division index	120.059	0.000	49	360.196	0.000
Puncture method vs. look-up table method	120.628	0.000	49	388.899	0.000

To evaluate the look-up table method's performance, four samples of 10,000, 50,000, 100,000, and 500,000 were used. Table 10 demonstrates that the average processing time for every point with four samples was less than 0.0002. Thus, the look-up table method is a competitive solution for using a multilayer-based map matching approach to group eight-segment layers. In the row for the average processing time of every point, the look-up table method provides stable performance using different samples. To evaluate the performance, the multilayer-based map matching algorithm using a look-up table with raster data provided the best efficiency for large truck fleet dispatching.

Table 10. Comparisons of grouping the eight-segment layers with different samples using the look-up table method.

	Look-Up Table Method			
Samples	10,000	50,000	100,000	500,000
Computation time (s)	1.1	7.16	15.12	75.6
Average processing time for every point (s)	0.00011	0.0001432	0.0001512	0.0001512

The computer used for the empirical tests was a desktop PC running Microsoft Windows 10, with an Intel Core i5-9400F CPU at 2.9 GHZ and 32 GB of RAM. Figure 6 shows the trade-off between efficiency and cost [39]. First, when the data structure and algorithm procedures are the same, marginal efficiency improvements can be realized by improving the hardware and software. After the hardware and software are upgraded to a certain extent, the marginal efficiency increase will gradually decrease. The efficiency limit is shown in Figure 6. When determining the investment costs of hardware and software, the actual cost, the market acceptable cost, and the profit should be considered. The best option may not be to upgrade only the hardware and software.

**Figure 6.** Trade-off between efficiency and cost.

Ultimately, this study used different data structures to effectively improve map matching and maximize the efficiency of map matching under existing software and hardware constraints. The number and complexity of the road sections will obviously affect the efficiency of multilayer-based map matching with the puncture method, but the number of GPS coordinate data do not greatly affect the computation time when using the look-up table method. The look-up method uses a database to sort the raster data and build an index for search and analysis. It is possible to reduce the length of the raster data and improve searching efficiency through coding and compression. The method of the spatial division index involves dividing the data into several layers and large blocks, which effectively reduces the number of data to be searched and improves the efficiency of map matching. Although multilayer-based map matching using raster data loses the partial geometry elements of the road networks, which negatively affects accuracy, the efficiency is also increased.

5. Discussion and Conclusions

The aim of this paper was to enhance the efficiency of large truck fleet dispatch to achieve timely deliveries. Our proposed multilayer-based map matching method can provide appropriate grouping combinations to handle complicated road networks for managing large truck fleet dispatch. The linear transformation process was verified for

handling GPS coordinates in a short time and rapidly providing all transformation coordinates in the case study. Overall, the computation time for individual layers was higher than that for the grouping layers.

The multilayer-based map matching approach presented here provided different outcomes with the three approaches used to compare the efficiency. Employing computers to perform the puncture method is time-consuming. The puncture method with multilayer-based map matching, however, can reduce the average processing time for every point. Establishing a spatial division index for the puncture method can improve performance, since indexing is a technical approach to reduce the scope of information data. Although indices speed up queries when many of the fields are indexed, they require even more space than the original data table. By maintaining the organization of the index tables, it takes more time for the data to change. Adding a spatial division index is recommended as the second-best approach. Raster-based compression can also take a great deal of time unless the map undergoes large revisions. Thus, raster maps cannot be frequently redrawn in practice.

Previous studies have focused on improving vehicle positioning accuracy and used the map matching method for general vehicle cases. This paper focused on map positioning and efficiency performance. Vector-based and raster-based formats have been shown to have advantages depending on different scenarios and so in this study, we used the point-to-polygon approach and then integrated these technologies and methods for improving performance. Based on the conclusion for polygons in raster-based format from [40], we extended the idea to solve large trucks with large-scale positioning data. The method contributes to large truck fleet management and shows that multilayer-based map matching with a raster-based data structure can improve fleet dispatch. Furthermore, a multilayer-based map-matching algorithm using the look-up table method does not require complicated mathematical equations or programming languages and can yield the lowest average processing time. When the sample size increased, the look-up table method offered a stable performance. With the increase in large truck fleets, our proposed approach can handle the computational bottleneck. Generally, mobile phones have limited computational ability, but installing a multilayer-based map-matching algorithm with a look-up table in the dispatching system can reduce load, and logistics technicians can operate obtain fleet coordinate information via mobile phone.

A limitation of our study is that we did not consider the overlapping of facilities. Generally, actual road networks contain various irregular intersections. Future research could address the intersections in facilities based on extending the center of the road a certain distance in the horizontal and vertical directions, rather than using a polygonal design. Internal consistency, which is a reliability topic [43], should be achieved in preliminary evaluations to reduce sample and measurement errors. In this study, the smaller truck fleet scenario was not our objective to assess the financial viability, and economies of scale are such that it is probably more feasible economically in large truck fleets, but this was not a consideration in this paper. Therefore, this study only focused on large trucks. In the near future, a similar approach can be applied to other regions and countries, and the types of orography can be considered for further research. The multilayer-based map matching approach can also determine fleet locations for further comparisons with different logistics vehicle sizes. This multilayer-based map matching algorithm using the look-up table method can serve as a useful reference in enhancing the performance of large truck fleet dispatching. Ultimately, by improving efficiency, our proposed map matching method can help large truck fleet drivers to quickly obtain the relevant point coordinates and complete their dispatching tasks.

Author Contributions: Conceptualization, Ching-Yun Mu; Data curation, Ching-Yun Mu, Pin Kung; Formal analysis, Ching-Yun Mu, Thanh Van Hoang; Investigation, Ching-Yun Mu, Pin Kung, Yao-Min Fang, Mei-Hsin Chen and Mei-Ling Yeh; Methodology, Thanh Van Hoang, Pin Kung; Resources, Thanh Van Hoang, Pin Kung; Supervision, Tien-Yin Chou, Thanh Van Hoang; Writing—original draft,

Ching-Yun Mu, Pin Kung, and Thanh Van Hoang; Writing—review & editing, Thanh Van Hoang and Tien-Yin Chou. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Skyeyes company, and Geographic Information Systems Research Center, Feng Chia University, Taiwan.

Data Availability Statement: <http://www.skyeyes.tw/en/> (accessed on 27 November 2020).

Acknowledgments: We thank ME Meadows for editorial assistance in compiling the final version of the manuscript and the Geographic Information System Research Center, Feng Chia University, Taiwan and Taiwan SkyEyes Company for their support.

Conflicts of Interest: The authors declare no conflict of interest.

References

- White, C.E.; Bernstein, D.; Kornhauser, A.L. Some map matching algorithms for personal navigation assistants. *Transp. Res. Part C Emerg. Technol.* **2000**, *8*, 91–108. [\[CrossRef\]](#)
- Goodchild, M.F. GIS and transportation: Status and challenges. *GeoInformatica* **2000**, *4*, 127–139. [\[CrossRef\]](#)
- Quddus, M.A.; Ochieng, W.; Zhao, L.; Noland, R.B. A general map matching algorithm for transport telematics applications. *GPS Solut.* **2003**, *7*, 157–167. [\[CrossRef\]](#)
- Quddus, M.A.; Ochieng, W.; Noland, R. Current map matching algorithms for transport applications: State-of-the art and future research directions. *Transp. Res. Part C Emerg. Technol.* **2007**, *15*, 312–328. [\[CrossRef\]](#)
- Zhao, Z.M.; Chou, T.Y.; Yan, T.L. *Spatial Information Technique Theorem and Its Application-Basic Theory*, 2nd ed.; Scholars Books; Scholars Publisher: Taipei, Taiwan, 2017.
- Arvind, V. Optimizing the turning radius of a vehicle using symmetric four wheel steering system. *Int. J. Sci. Eng. Res.* **2013**, *4*, 2177–2184.
- Janssens, D.; Yasar, A.-U.-H.; Knapen, L. *Data Science and Simulation in Transportation Research*; IGI Global: Charleston, SC, USA, 2013.
- Blazquez, C.A.; Miranda, P.A.; Ponce, A.D. Performance of a new enhanced topological decision-rule map matching algorithm for transportation applications. *J. Appl. Res. Technol.* **2012**, *10*, 929–940. [\[CrossRef\]](#)
- Li, Q.; Zhang, T.; Yu, Y. Using cloud computing to process intensive floating car data for urban traffic surveillance. *Int. J. Geogr. Inf. Sci.* **2011**, *25*, 1303–1322. [\[CrossRef\]](#)
- Miwa, T.; Kiuchi, D.; Yamamoto, T.; Morikawa, T. Development of map matching algorithm for low frequency probe data. *Transp. Res. Part C* **2012**, *22*, 132–145. [\[CrossRef\]](#)
- Chen, B.Y.; Yuan, H.; Li, Q.; Lam, W.H.K. Map matching algorithm for large-scale low-frequency floating car data. *Int. J. Geogr. Inf. Sci.* **2014**, *28*, 22–38. [\[CrossRef\]](#)
- Zhao, Z.M.; Chou, T.Y.; Yan, T.L. *Spatial Information Technique Theorem and Its Application-Technical Application*, 1st ed.; Scholars Books; Scholars Publisher: Taipei, Taiwan, 2015.
- Quddus, M.A.; Noland, R.B.; Ochieng, W.Y. The effects of navigation sensors and spatial road network data quality on the performance of map matching algorithms. *GeoInformatica* **2009**, *13*, 85–108. [\[CrossRef\]](#)
- Lopes, S.B.; Brondino, N.C.M.; da Silva, A.N.R. GIS-based analytical tools for transport planning: Spatial regression models for transportation demand forecast. *ISPRS Int. J. Geo-Inf.* **2014**, *3*, 565–583. [\[CrossRef\]](#)
- Bernstein, D.; Kornhauser, A. *An Introduction to Map Matching for Personal Navigation Assistants*; New Jersey Institute of Technology: Newark, NJ, USA, 1998; pp. 1–16.
- Loidl, M.; Wallentin, G.; Cyganski, R.; Graser, A.; Scholz, J.; Haslauer, E. GIS and Transport Modeling—Strengthening the Spatial Perspective. *ISPRS Int. J. Geo-Inf.* **2016**, *5*, 84. [\[CrossRef\]](#)
- Xiong, D. A three-stage computational approach to network matching. *Transp. Res. C Emerg. Technol.* **2000**, *8*, 71–89. [\[CrossRef\]](#)
- Yao, X.; Li, G. Big spatial vector data management: A review. *Big Earth Data* **2018**, *2*, 108–129. [\[CrossRef\]](#)
- Pfoser, D.; Tryfona, N.; Jensen, C.S. Indeterminacy and spatiotemporal data: Basic definitions and case study. *GeoInformatica* **2005**, *9*, 211–236. [\[CrossRef\]](#)
- Kuijpers, B.; Moelans, B.; Othman, W.; Vaisman, A. Uncertainty-based map matching: The space-time prism and k-shortest path algorithm. *ISPRS Int. J. Geo-Inf.* **2016**, *5*, 204. [\[CrossRef\]](#)
- Velaga, N.R.; Quddus, M.A.; Bristow, A.L. Improving the performance of a topological map matching algorithm through error detection and correction. *J. Intell. Transp. Syst.* **2012**, *16*, 147–158. [\[CrossRef\]](#)
- Luo, A.; Chen, S.; Xv, B. Enhanced map matching algorithm with a hidden markov model for mobile phone positioning. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 327. [\[CrossRef\]](#)
- Liu, C.; Xiong, L.; Hu, X.; Shan, J. A progressive buffering method for road map update using OpenStreetMap data. *ISPRS Int. J. Geo-Inf.* **2015**, *4*, 1246–1264. [\[CrossRef\]](#)
- Fan, H.; Yang, B.; Zipf, A.; Rousell, A. A polygon-based approach for matching OpenStreetMap road networks with regional transit authority data. *Int. J. Geogr. Inf. Sci.* **2016**, *30*, 748–764. [\[CrossRef\]](#)

25. Brovelli, M.A.; Zamboni, G. A new method for the assessment of spatial accuracy and completeness of OpenStreetMap building footprints. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 289. [[CrossRef](#)]
26. Zhang, T.; Yang, D.; Li, J.; Lian, X. A trajectory-based map matching system for the driving road identification in vehicle navigation systems. *J. Intell. Transp. Syst.* **2016**, *20*, 162–177. [[CrossRef](#)]
27. Brakatsoulas, S.; Pfoser, D.; Salas, R.; Wenk, C. On map matching vehicle tracking data. In Proceedings of the 31st International Conference, Rondheim, Norway, 30 August–2 September 2005; pp. 853–864.
28. Hashemi, M.; Karimi, H.A. A weight-based map matching algorithm for vehicle navigation in complex urban networks. *J. Intell. Transp. Syst.* **2016**, *20*, 573–590. [[CrossRef](#)]
29. Kong, X.; Yang, J. A scenario-based map matching algorithm for complex urban road network. *J. Intell. Transp. Syst.* **2019**, *23*, 617–631. [[CrossRef](#)]
30. Hashemi, M.; Karimi, H.A. A critical review of real-time map matching algorithms: Current issues and future directions. *Comput. Environ. Urban Syst.* **2014**, *48*, 153–165. [[CrossRef](#)]
31. Zheng, L.; Sun, M.; Luo, Y.; Song, X.; Yang, C.; Hu, F.; Yu, M. Utilizing MapReduce to improve probe-car track data mining. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 287. [[CrossRef](#)]
32. Liu, M.; Zhang, L.; Ge, J.; Long, Y.; Che, W. Map Matching for Urban High-Sampling-Frequency GPS Trajectories. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 31. [[CrossRef](#)]
33. Rodrigue, J.P. *The Geography of Transport Systems*, 3rd ed.; Routledge: New York, NY, USA, 2013.
34. Guan, Q.; Clarke, K.C. A general-purpose parallel raster processing programming library test application using a geographic cellular automata model. *Int. J. Geogr.-Inf. Sci.* **2010**, *24*, 695–722. [[CrossRef](#)]
35. Hackeloeer, A.; Klasing, K. Georeferencing: A review of methods and applications. *Ann. GIS* **2014**, *20*, 61–69. [[CrossRef](#)]
36. Sun, Y.; Yao, C.; Zhao, S.; Xu, M. Unique identification of vector geographic objects by location and geometry. *Int. J. Geogr. Inf. Sci.* **2014**, *28*, 2495–2510. [[CrossRef](#)]
37. ArcGIS. Available online: <https://www.arcgis.com/index.html> (accessed on 31 January 2021).
38. Ye, S.J.; Yan, T.L.; Yue, Y.L.; Lin, W.Y.; Li, L.; Yao, X.C.; Mu, Q.Y.; Li, Y.Q.; Zhu, D.H. Developing a reversible rapid coordinate transformation model for the cylindrical projection. *Comput. Geosci.* **2016**, *89*, 44–56. [[CrossRef](#)]
39. Mu, C.-Y. Performance Enhancement of Spatial Big Data Map Matching by Using Linear Transformation of Coordinates and Distinctive Spatial Index Structures. Ph.D. Dissertation, Feng Chia University, Taichung, Taiwan, 2015.
40. Huang, C.-W.; Shih, T.-Y. On the complexity of point-in-polygon algorithms. *Comput. Geosci.* **1997**, *23*, 109–118. [[CrossRef](#)]
41. Hao, J.; Sun, J.; Chen, Y.; Cai, Q.; Tan, L. Optimal reliable point-in-polygon test and differential coding boolean operations on polygons. *Symmetry* **2018**, *10*, 477. [[CrossRef](#)]
42. Ott, L.; Longnecker, M. *An Introduction to Statistical Methods and Data Analysis*, 7th ed.; Cengage Learning: Boston, MA, USA, 2015.
43. Webb, N.; Shavelson, R.; Haertel, E. Reliability Coefficients and Generalizability Theory. *Handb. Stat.* **2006**, *26*, 81–124.