

Improvement of an Automatic Vehicle Location System for Route Bus Service with OpenStreetMap Data

Piao Tailong, Masaki Ito, Takao Kawamura, and Kazunori Sugahara
Department of Information and Electronics, Graduate School of Engineering
Tottori University, 4-101 Koyama Minami Tottori 680-8552, Japan
Email: {k112005, masaki, kawamura, sugahara}@ike.tottori-u.ac.jp

Abstract—This paper discusses integration of route bus map and Busnet. We developed the shortest-path-planning system for route buses and railways called "Busnet" in 2006 and have been operating the system as a web service in Tottori prefecture, Japan for six years. The system provides the appropriate itinerary to the destination in using route buses and railways. In this paper, we will integrate route bus map with Busnet. With the route bus map, we can provide rich services that increase the awareness and findability of route buses for the mobile users with advanced visualization of the map. We can also predict the arrival time of route buses based on the location reported from the running buses. This paper introduces the basic technique to manage the route bus map for Busnet, and discuss future possible services with the map.

Index Terms—database, bus service, OpenStreetMap, parse log.

I. INTRODUCTION

Public transport such as a route bus is important for people who do not have a private car. However, since it is hard to obtain information about bus, route bus system is not convenient for practical use. Therefore, in order to promote the use of public transport, we have developed an information system to support the use of public transportation system called "Busnet". Also, we have been operating the system as a web service in Tottori prefecture since 2006. Busnet has a route search function. It can find the best itinerary to the destination considering distance between different bus stops and the delay of the buses. Busnet also provides the function of searching timetable for each bus stop. Busnet is available for a PC, a mobile phone or information terminal called "Intelligent bus stop". The information terminal is available in various public places such as Tottori Station and Tottori prefectural office.

Today, the development of the basic services of Busnet is completed. However, it can be more convenient. For example, Busnet has route search function considering the delay of the bus. But the accuracy of the delay can be improved. Also the system is difficult to understand because it mainly provides information with characters. To solve the above problems, this research builds a database of bus route with OpenStreetMap(OSM) and the bus location system

II. IMPROVEMENT OF AUTOMATIC VEHICLE LOCATION SYSTEM USING THE DATA OF BUS ROUTE

Busnet has a system called "bus location system". The server of the system acquires the current GPS locations of the buses in real time from the smart phone. The smart phone reports its current location every 1 minute. Using the information, Busnet displays the current location of the bus on a map, and estimates the amount of the delay of each running bus in order to improve the result of the route search.

A. Correction of the Location of the Bus

We show the current position of running buses on a map to those people who are waiting for a bus. However, GPS is not so accurate when the vehicle is moving. On the map, an icon of the bus sometimes locates in wrong position, as shown in Figure 1. The icon, which should have been on the road, is outside the road.

We utilize information of the routes to correct the bus location. In other words, we put the icon on the nearby road. So that will look more reasonable.



Fig. 1. The location of the icon of the bus is wrong.

B. calculate the delay more accurately

The bus cannot always travel precisely according to timetable due to road conditions, bad weather and unexpected accidents. It is uncertain for a waiting user when the bus will come. If a user needs to transfer other bus, he/she may miss the next bus. Therefore we have developed a route search function considering the delay of the bus. Bus location system calculates the delay of the bus based on the location and timetable of the running bus. The calculation method of the bus delay is as follows.

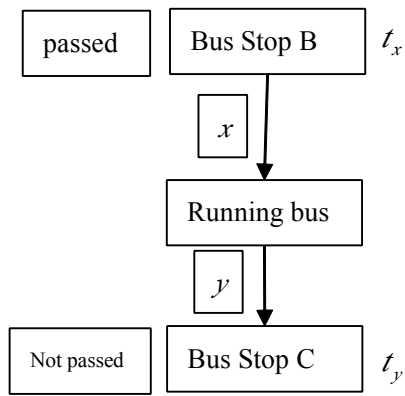


Fig.2. Calculation of delay.

1. Find the last bus stop the bus passed and the next bus stop. x is the distance between the current location of the bus and bus stop B. y is the distance between the current location of the bus and the bus stop C. t_x is the time when the bus should have passed bus stop B according to the timetable, and t_y is the time the bus will pass the bus stop C according to timetable.
2. p is the percent that bus has run between the bus stop B and the bus stop C.

$$p = \frac{x}{x+y}$$

3. t_p is the time bus should have passed current location according to the timetable.

$$t_p = t_x + (t_x - t_y)p$$

4. Use t_p and current time to calculate the delay of the bus.

$$\text{Delay} = t_p - \text{current time}$$

This method calculates the delay accurately if the distance x and y is correct.

In current Busnet, we take the straight-line distance as the distance between bus stops instead of the actual distance. If the shape of the road between bus stops B and C is straight, the delay is correct. However, not every path between the bus stops is straight.

Also, this method does not calculate the future delay, only calculate the latest delay. What a user really wants to know is the delay when he/she will take a bus. Therefore prediction of the future delay is important.

We utilize the route of buses to increase the accuracy of the x and y mentioned in 2-B in order to increase the accuracy of the delay. Bus routes can also be useful to predict the delay in the future. The method is as follows.

- 2.1 Record the transit time of a bus on a certain section.
- 2.2 Assume that the next bus will take the same transit time to run through the same section.
- 2.3 Compare the estimated arrival time with the time from the timetable, and then we predict the delay in the future.

C. Make the information service richer

The information from Busnet is difficult to understand. Because it is mainly consist of characters. Even in the bus location system, which illustrates the current locations of the buses on a map as icons, it is not easy to understand the direction and destination of the bus just by looking at the

screen. We can say that the information provided on the Busnet is not thoughtful. Only with character and icon of bus, we cannot provide attentive services.

If we show the routes of buses on the map in addition to the icons of the buses representing current location of the buses, a user can know the relationship between the routes and the bus icons intuitively. Then the service will be more thoughtful.

III. BUILDING THE BUS ROUTE DATABASE

This paper proposes the method that builds the bus route database to meet the needs of the above-mentioned function and support editing function of the database. In order to build the database mentioned above, we need to solve the following problems.

A. How to collect information of bus routes

We must survey the geographic locations and shapes of the bus routes precisely and digitize the bus routes in order to create the bus route database. Since there is many bus routes and it occasional changes, to create and maintain the database is difficult.

In this research, we will use the data of the roads from OSM and the locations of the running buses from the bus location system to develop a semiautomatic building system of database of bus routes.

OSM is a collaborative project to create a free editable map of the world. Many participants of the project upload their trajectories captured by handy GPS devices, trace the trajectory to create the road data, and add tags such as names of places, names of roads and names of shops to the data. As the result fresh and detailed map data are available in vector format with open license. We can acquire the data of the road from OSM via the API. The form of OSM data is a list of primitives such as nodes, ways, and relations, and is represented as XML.

In the log of the bus location system, there are not only the current locations of the buses, but also the data about the originating station or bus stop, the destination station or bus stop, stations and bus stops on the path, departure time, passed bus stations, routes, etc. Thus, we can appoint one of the current locations of the bus.

B. The format of the bus route data and amount of information

The more the route data is available, the better delay of the route bus can be calculated. However, if the data is too large, the processing time of the data will be too long. Thus, the balance is important.

In the function that shows the current location of the bus and route of the bus on the map, we must fit the location of the bus icons to the bus routes on the map. We do not have to worry too much about time to process the data on the client, because smart phone and PC performance is good right now. So the more data is available, the better map will be generated.

In the function that calculates the delay we need a lot of data in order to calculate the length of road accurately. However, if the data is too much, the speed to calculate the delay will be slow. So balance is important.

In the function that predicts the delay in the future, we need only the time that the bus runs through the section, and we do not need the points that express the detailed shape of the route within the section.

In order to satisfy different requirement to the route database mentioned above we will design the structure of the database to extract the necessary information flexibly.

C. The problem about editing the database of the bus route

After the building database of bus routes, we must make a function to add data, to edit data and to delete data. So, we must design the database to be suitable to edit. To do that, the database must meet following two conditions. One is that there must be no side effect to other bus routes when editing a portion of the data of bus routes. The other is that when updating the data of the road, the needed edit of the route of the bus on the road should be smaller.

IV. THE CURRENT STATE OF THE DEVELOPMENT

We need to do the following three things in order to create the database of the bus location. First, we parse the log file of the bus location system and put the information into the database. Second, we put the road data from OSM into the database. Third, we build the database of routes with the log and OSM.

A. Parse the log of the bus location system

We have already developed a bus location system in order to grasp the current location of the bus. We parsed the log of the bus location system in order to acquire the information of bus route. Bus location system leaves the history of the current locations reported from smart phones on route buses as a log file. The interval of the report is one minute. Parsing the log, we can find the data of running buses including the times and the locations. The figure 3 illustrates the 1000 location information of the bus on the OSM. The icons represent the history of location of the bus that start at 15:30 and whose name is "juroppon matsu (ge)" going to "hattyome". We will make a program that selects the road on OSM on which the buses run, and put it into the bus route database.

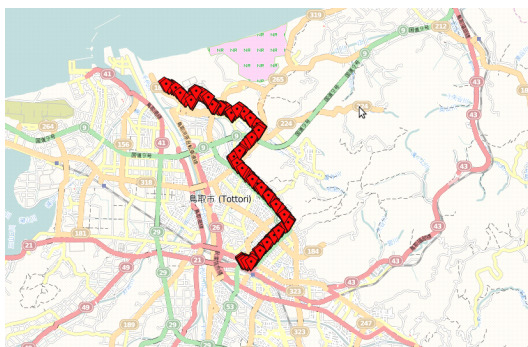


Fig. 3. Trace of bus location on "juroppon matsu (ge)"

B. Acquire road data from OSM

The form of OSM data is a list of primitives such as Nodes, Ways, and Relations. A node defines a single geospatial point using a latitude and longitude. The way is an ordered list of between 2 and 2000 nodes. Ways can be used to represent linear features (vectors) or polygons (areas). A Relation consists of an ordered list of nodes, ways and sometimes also other relations as member of the new relation.

The figure 4 is a map around Tottori University. The figure is expressed with the data structure of OSM. In the data, a node represents a location, the line represents a way or an

area, and a relationship represents the connection between nodes or lines even when the nodes or lines are invisible. In OSM, a curve is expressed by collection of short straight lines.

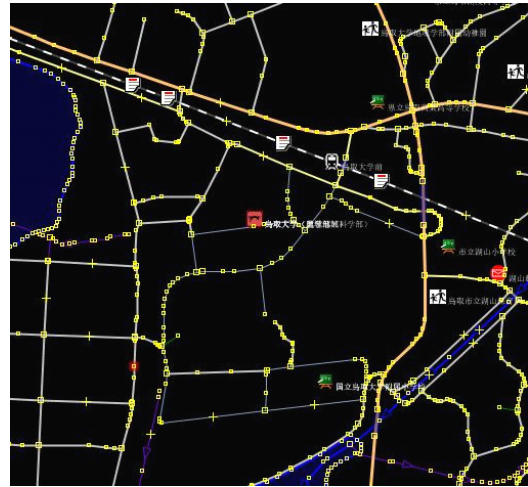


Fig. 4. The data structure of OSM

We acquire road data from osm file of OSM, and put it to the following database. We called it OSM_database;

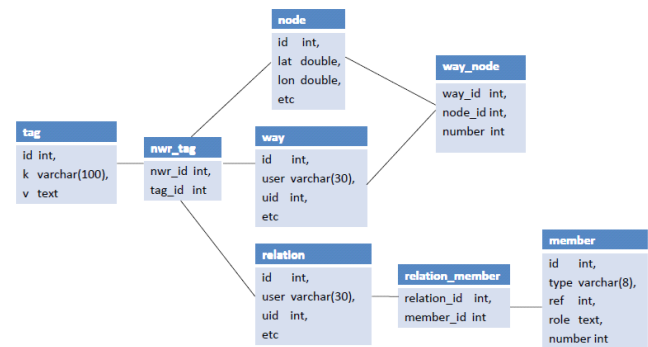


Fig.5. OSM_database

C. Schema of bus route database

The figure 6 is a ER model of the bus route database. The model is designed to solve the problem mentioned in III-B and III-C.

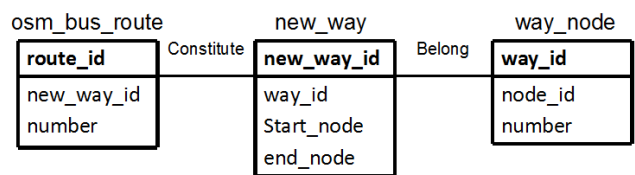


Fig.6. ER model of the bus route database

The "way_node" table from OSM_database.

In the "way_node", the "way_id" is the id of the way in OSM. The "node_id" is the id of node in OSM. The "number" is the sequence number of the node in "way_node".

The "new_way" is created by modifying the "way_node".

We search the first node and last node in the "way_node". Then they are transformed to the "start_node" and the "end_node" in the "new_way".

"Osm_bus_route" is a collection of "new_way". It represents a route.

With this schema have following advantages.

1. Get details easily

In the function to show the current location of the bus and

bus route on the map, we must fit the bus icon to the bus route on the map. In this database we can get the detail of the route easily from the way_node table.

2. Calculate the delay quickly

The more data is available, the more accurate the estimation of the delay will be. However, large data requires amount of processing time. By sampling the nodes in the way_node table we can decrease size of the data.

We need the information of the Interval to predict the delay in the future. With the current location of the bus and the id of the bus route, we can find the Start/End_node of the Interval in route from "new_way". Then if we know the time that the previous bus cost to pass the certain section, we could predict the time this bus will take to pass the same section. As we propose it will take the same time.

3. Make a program that semiautomatic builds the bus route database easily

During searching the bus route, the problem focus on searching the road bus gets through instead of searching the point bus passed. With the new_way, we can find the roads instead of the points.

4. Edit bus route data

In this research, when a user edits a bus route data, another bus route data should not be affected. As the new_way_id in the osm_bus_route is foreign key. When the osm_bus_route is changed there is no influence to the map data. Also there is no need to consider the nodes.

D. The algorithm of building bus route database

The figure 7 shown bus locations on the OSM. The red points in the map represent bus locations.

With statistic bus location number nearby one bus location, we can judge whether a bus location is noise. The figure 8 shown after the filter.

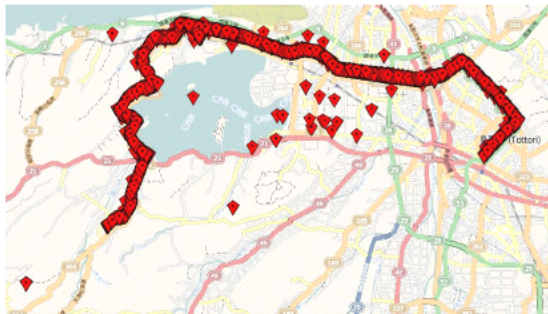


Fig.7. The bus locations on OSM

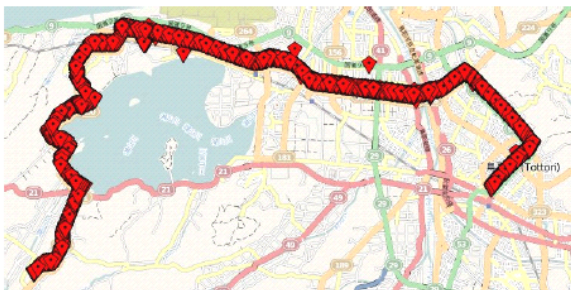


Fig.8. After the filtered bus locations on OSM

The log of the bus location system has the information of bus location. Utilizing the bus location and the new_way table, we develop a semiautomatic program that builds a bus route database easily. The Program flow chart is as figure 9.

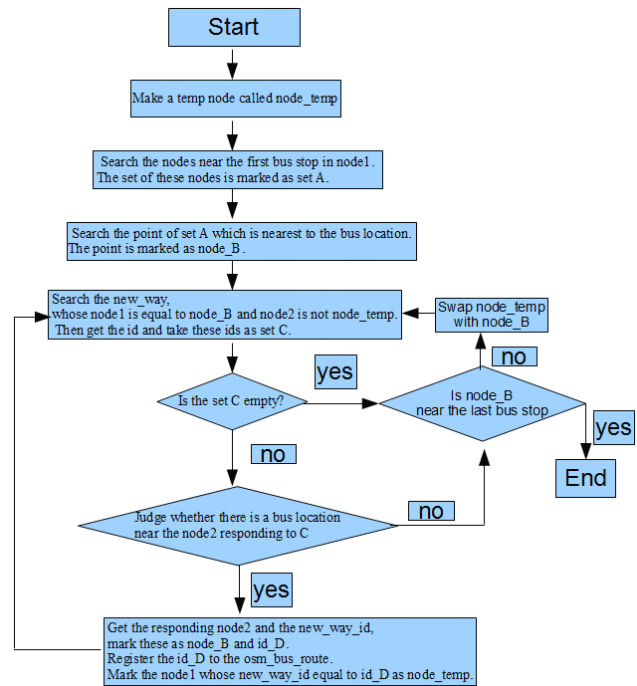


Fig.9. Searching bus route

The figure 10 is the route named "juroppon matsu (ge)". It is the result with the above method.

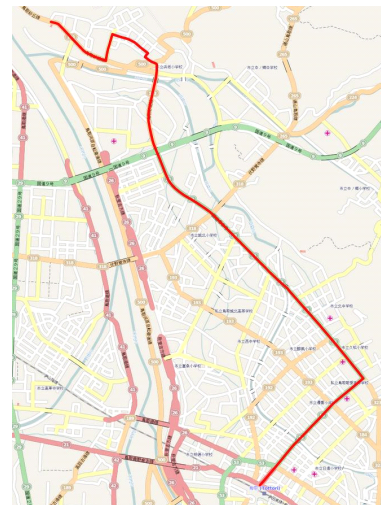


Fig.10. The route named "juroppon matsu (ge)"

V. CONCLUSIONS

In this paper, we discussed the method to build a database of bus routes to improve the bus service. We defined the format of the bus route data and developed the system that parse log of bus location system. We store the road data to the database acquired from OSM. Then we integrate the log and road data to create the bus route information. As the next step of this research, we hope to develop the algorithm to be more accurate. Also we would like to develop the API to inquire and edit bus route data. Finally we will operate and improve the system.

REFERENCES

[1]Busnet :<http://www. ikisaki. jp/>
 [2]OpenStreetMap:<http://www. openstreetmap. org/>
 [3]OpenStreetMap elements:
<http://wiki. openstreetmap. org/wiki/Elements>
 [4]bus location system:
http://www. ikisaki. jp/gps/bus_map_all. rhtml