OSM-SVG Converting for Open Road Simulator

Rajashree S. Sokasane, Kyungbaek Kim

Department of Electronics and Computer Engineering
Chonnam National University
Gwangju, Republic of Korea
sokasaners@gmail.com, kyungbaekkim@jnu.ac.kr

Abstract

Monitoring the flow and congestion of vehicular traffic is essential for efficient road systems in cities; therefore traffic monitoring has received a significant amount of attention. Recently, sensor techniques have been used to collect real-time traffic information, such as loop detectors, cameras, etc. However these sensor techniques have some issues; first these sensors are limited with their coverage area to collect traffic information. Second, to collect traffic information of big city road network, we need number of sesors; which costs very high. To address such problems related to traffic information we proposed a model with the help of simulator. "OSM-SVG" conversion model is develoed to use any arbitrary region for simulation. Trajectory collector module is designed to gather trajectory logs of moving objects. Clustering module is used to get dense roads in a given roadmap of city. From our experimental study it is observed that, we acquire 70-80% exactly similar results as compared with real world data like GPS traces.

Keywords: openstreetmap; SVG; Density Based Clustering

I. Introduction

Roads and vehicular traffic are a key part of the day-to-day lives of people. The analysis of moving objects behavior on road network is recently getting more attention in research area. Traffic Simulation can be applied to both transportation planning and to transportation design and operations. In transportation planning the simulation models evaluate the impacts of regional urban development patterns on the performance of the transportation infrastructure. Regional planning organizations use these models to evaluate what-if scenarios in the region, such as air quality to help develop land use policies that lead to more sustainable travel.

Analyzing the behavior of moving objects (i.e. entities whose positions or geometric attributes changes over time) over provided road network, helps to accumulate traffic information like which road segments are crowded; this information is helpful for managing traffic flows. Moving objects are modeled as moving points, whose trajectories (i.e. paths through space and time) can be analyzed. Most of existing research works in this direction focus on the sensors to collect data for transportation planning of city. To develop good transportation plannibg we must know about the road network of the city and traffic information of city such as which roads are crowded.

In recent years, for making good transportation planning, many advanced sensor techniques have been adopted to collect real-time traffic information, such as loop detectors, cameras, etc. These sensor techniques all endure their advantages and disadvantages. With the sensors like loop detectors, cameras [1][2] are used to gather traffic information of specific location where they installed; these are not able to gather traffic information beyond the location of installation and their coverage area. Second, in case of large scale urban road network it is not a practical scheme to use fixed detectors to gather the traffic information on every link, because it will incur a huge financial cost in both the initial construction and the subsequent maintenance.

To address such problems related to traffic information, we proposed a model "OSM-SVG Converting for Open Road Simulator" with the help of traffic simulator. Simulation of road network traffic is good option to overcome such problems. In this paper we focused on open road simulator. Simulator which uses the roadmap as an input and set some random moving objects with given roadmap. We designed one conversion scheme "OSM-SVG" with the help of JOSM to use roadmap information for simulation. JOSM editor [5] is used for conversion of osm to svg; but this svg format is not suitable for simulator as an input. Due to which we used to convert resulted svg file into an input svg format. Trajectory collector module is implemented along with simulator, due to which we may able to collect GPS logs of moving objects. This information is used with density based clustering technique (clustering module) to get most dense roads in a given roadmap of city. To check our obtained results are correct or not, we collected GPS tracecs of city from openstreetmap site [3] and compared it.

Our contributions are as follows:

- With proposed conversion model, we can select any arbitrary region's roadmap from the world for analysis of road network traffic.
- Gather mobile user's (moving objects) trajectories moving on arbitrarily provided roadmap using simulator.
- Provides frequently used road segments, of any arbitrary selected region by using density based clustering.

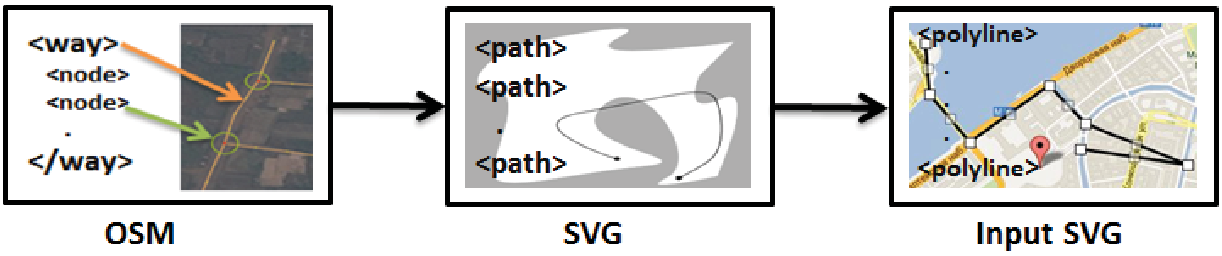

Figure 1: OSM-SVG Conversion Model

The rest of the paper is structured as fallows. Section II provides background of openstreetmap, svg, density based clustering techniques and the simulator. Section III describes the proposed conversion model in detail. Section IV presents the implementation, evaluations and results. Finally, Section V covers conclusion and discussions.

II. Background

A. Openstreetmap

Openstreetmap is a global, collaborative effort to create a free and universal map of the world. OpenStreetMap is built by a community of mappers that contribute and maintain data about roads, trails, cafés, railway stations, and much more, all over the world. OpenStreetMap is an initiative to create and provide free geographic data, such as street maps, to anyone. It is dedicated to encouraging the growth, development and distribution of free geospatial data and to providing geospatial data for anyone to use and share [3].

Basically OSM (Open Street Map) is a list of instances of our data primitives (nodes, ways, and relations) that are the architecture of the OSM model [5]. A <way> is an ordered list of nodes, which normally also has at least one tag or is included within a Relation. A <way> can have between 2 and 2,000 nodes, although it's possible that faulty ways with zero or a single node exist. A <way> can be open or closed. A closed way is one whose last node on the way is also the first on that way. A closed way may be interpreted either as a closed polyline, or an area, or both.

A <node> is one of the core elements in the OpenStreetMap data model. It consists of a single point in space defined by its latitude, longitude and node id. Nodes can be used to define standalone point features, but are more often used to define the shape or "path" of a <way>. Many nodes form part of one or more ways. Where ways intersect at the same altitude, the two ways must share a node (for example, a road junction).

B. Scalable Vector Graphics

Scalable Vector Graphics (SVG) is an XML-based vector image format for two-dimensional graphics with support for interactivity and animation. SVG [6] images and their behaviors are defined in XML text files. This means that they can be searched, indexed, scripted, and compressed. As XML files, SVG images can be created and edited with any text editor, but are more often created with drawing software. SVG is a widely-deployed royalty-free graphics format developed and maintained by the W3C SVG Working Group [6].

The <path> element is the generic element to define a shape. All the basic shapes can be created with a path element. A path can be used to describe the same figures as line, polyline, polygon, circle, and rect elements. The syntax is a little more complex than those elements, but is also more general. The significant attribute in the <path> element called 'd'. The 'd' attribute contains drawing commands; and is assigned a string of text which describes the path to be created.

The <polyline> element is an SVG basic shape, used to create a series of straight lines connecting several points. Typically a <polyline> is used to create open shapes. This element is used to draw multiple connected lines. The multiple lines are identified by points. Each point is listed as x,y in the 'points' attribute. A <polyline> can only draw connected line segments.

C. Simulator

We adapt the public event-based simulator GTMobiSIM [4] to generate thousands of mobility traces on those road networks for a large-scale evaluation. By using this simulator we may well able to generate mobility traces of the loaded roadmap file. On the other hand we can say that, this simulator is helpful for generating mobility traces and query traces for large numbers of mobile users moving in a road network. Through the simulator we may able to provide number of mobile users as well as time for simulations by using configuration file, which is one of the feature of this simulator.

D. Density Based Clustering

Clustering algorithms are attractive for the task of class identification in spatial databases. Density-based clustering methods are an important category of clustering methods that are able to identify areas with dense clusters of any shape and size. DBSCAN (Density Based Spatial Clustering of Applications with Noise) is one of them. It is popular because of the ability of discovering clusters with arbitrary shapes for providing much interesting information.DBSCAN is efficient even for large spatial databases [7].

III. OSM-SVG conversion model for Open Road Simulator

In this paper, we presented a "OSM-SVG" conversion model. "OSM-SVG" conversion model is helpful in converting roadmap data (OSM) to the input format of simulator (SVG). Fig. 1 shows “OSM-SVG” conversion model. “OSM-SVG” conversion model has 3 main modules; OSM, SVG and input SVG respectively. These modules represent how roadmap data is transformed from one format to other. Working of this model is carried out in three steps given below:

- Select a region to generate a roadmap with the help of openstreetmap.
- Change data format of map obtained from openstreetmap
- Conversion of map for passing to simulator

Step 1: Select a region

In this step we need to select a region (for e.g. city, part of city) according to requirement. Fig. 2 shows a one of the e.g. roadmap of Gwangju city. With the help of openstreetmap’s website, we can acquire any region’s roadmap. We need to save this roadmap information in a file with extension .osm, which is

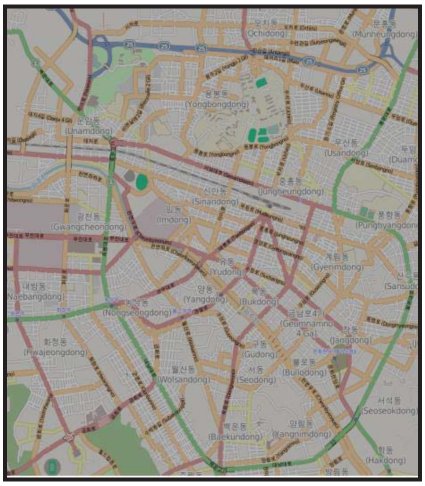

Figure 2: Gwangju roadmap (OSM)

generated by openstreetmap. It contains nodes (A node represents a specific point on the earth's surface defined by its latitude and longitude. Each node comprises at least an id number and a pair of coordinates.), ways (A way is an ordered list of nodes. Ways are used to represent linear features such as roads and river).

Step 2 : Change data format of map obtained from openstreetmap

Achieved roadmap file contains information related to nodes and ways of selected region. In this step we need to convert the node and way information in to the format of path. Path is an element which defines a link/path between two/multiple points or nodes. In this step we can create a file with .csvg extension to keep track of variations in data formats. We achieved these changes in data formats with the help of osm-rendering. After executing this step, the file contains information of paths instead of nodes and ways. Fig. 3 shows converted roadmap of gwangju city.

In this step we are using JOSM editor to convert osm into svg (save file with extension .csvg). However this file is not in suitable format of simulator. To convert resulted file into an input svg format of simulator we must follow step 3.

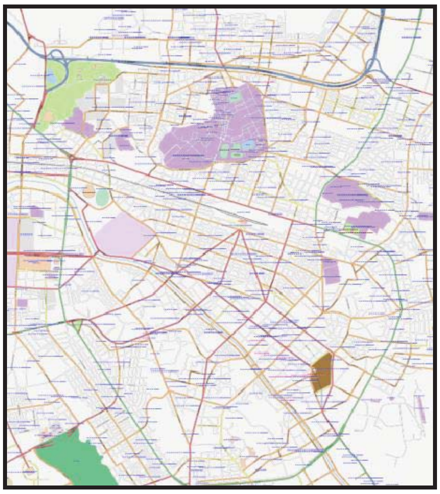

Figure 3: Gwangju roadmap (Converted to SVG)

Step 3: Conversion of map for passing to simulator

This step involves in transforming .csvg file into a .svg (Scalable Vector Graphics) file. Being "Scalable" means that the viewer can scale the SVG image up and down in size without loss of quality. This is possible because the graphics are defined as numbers instead of pixels. To provide a roadmap as an input to the simulator, it is essential that, roadmap file should be in .svg format. For this conversion we created a parser to parse .csvg file into .svg file. In this step we can create .svg file with polyline element. The polyline element is an SVG basic shape element. Typically a polyline is used to create open shapes. After executing this step the roadmap file is in appropriate format to pass as an input for simulator.

After passing the roadmap file to simulator, it starts simulating a map with mobile users (moving object) moving on provided roadmap as shown in fig. 4.

Figure 4: Gwangju roadmap with mobile users

IV Implementation and Evaluation

This section covers implementation, evaluation results. In this paper, we proposed Conversion Model, Trajectory Collector Module, and Clustering Module.

A. Implementation

Figure 5 shows an implementation and work- flow of proposed modules. Conversion Model Provides suitable input to the GTMobiSIM simulator. With Conversion Model described in section 3, we can provide any region's roadmap file as an input to GTMobiSIM simulator. JOSM editor is used as an intermediate step of conversion from osm to svg. However this step help us to convert the roadmap with osm extension to svg extension, we can not pass this svg file as an input to simulator; we need to convert the resulted svg file (obtained from step 2 of conversion model) into input svg format (achieved by using stp 3 in conversion model). Trajectory Collector module is implemented together with GTMobiSIM simulator.

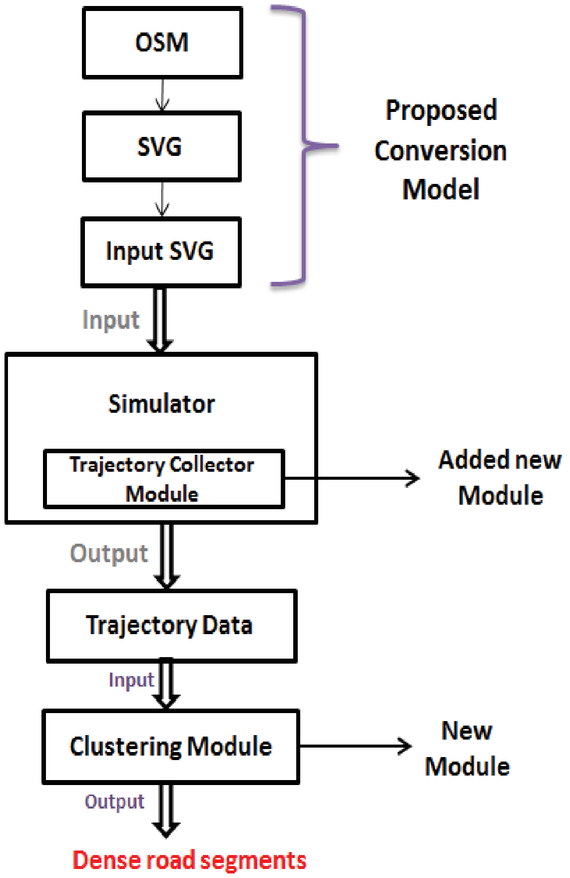

Figure 5: Flow for Simulation of road network traffic

Trajectory Collector module is designed and implemented as a part of simulator. Trajectory Collector module is used to gather each user's trajectory data; with the help of this module, we may able to create trajectory file of every user, which contains GPS co-ordinates associated with it. This trajectory data is used as input for Clustering module. Clustering module is designed with the help of a well-known density based clustering technique. Clustering module is used to cluster trajectory data, which is obtained by using Trajectory Collector module along with simulator.

Clustering module uses DBSCAN clustering technique to cluster trajectory data. By using DBSCAN we may able to find out most frequently used roads in given region of roadmap. In the Figure 6 given below, we can see that the points with black color, showing most frequently used road segments (or the road segments with high density) of Gwangju city. This information is helpful for managing traffic state in Gwangju city. Like this with the help of proposed modules in this paper, we can simulate the road network traffic of any region without installing fixed sensors and without paying plentiful cost for maintenances.

Figure 6: Clustered trajectory data

B. Results

In this paper we focused on open road simulator, to collect gps logs of moving objects; and also processed that GPS logs with clustering modules to get most frequently used roads in the given region. DBSCAN were applied on two datasets (delhi and daejon); trajectory data collected by using simulator and GPS traces collected from openstreetmap site. To check the performance with proposed modules, we compared the clustering results obtained by using simulator and proposed module with GPS traces obtained by using openstreetmap site. From the observed results fig. 7 (b) after applying density based clustering shows 70 % similar output which we observed from simulator shown in fig. 7 (a). And also the experimental study shows that, results obtained with proposed modules are better than it obtained with GPS traces. Fig. 7 and 8 shows the different region's roadmap delhi and daejon respectively, and comparison between results obtained with different datasets. Fig. 7(a) shows roadmap of Delhi with number of mobile users. Fig. 7(b) shows results after applying DBSCAN on trajectory data obtained with proposed modules; and fig. 7(c) shows GPS traces obtained from openstreetmap site. In the same way Fig. 8 (a) shows roadmap of Daejon with number of mobile users. Fig. 8 (b) shows results after applying DBSCAN on trajectory data obtained with proposed modules; and fig. 8 (c) shows GPS

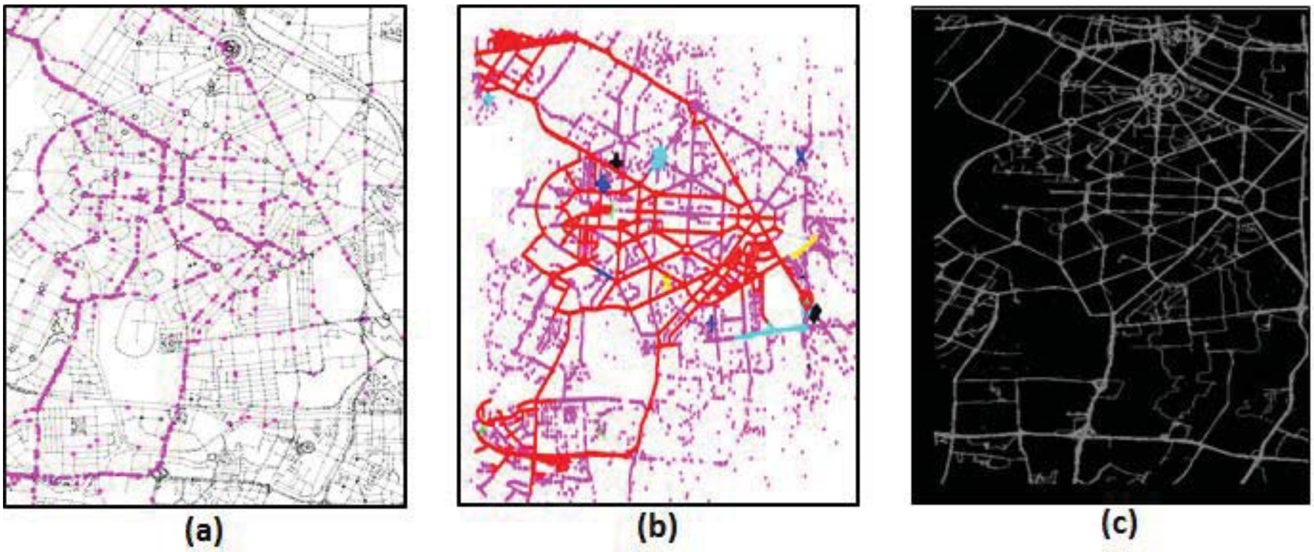

Figure 7: Delhi roadmap data

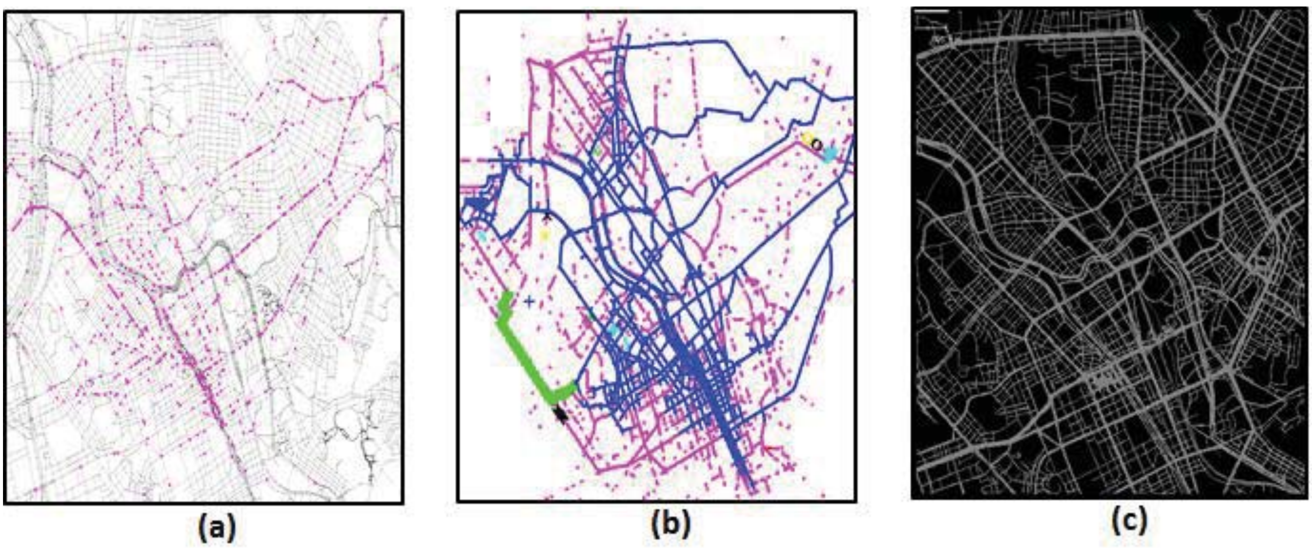

Figure 8: Daejon roadmap data

traces obtained from openstreetmap site; however fig 8 (c) shows gps traces of daejon city, but the emphasized traces shows frequently used roads in a city. From this it is observed that the results obtained with our proposed approach and GPS traces obtained from openstreetmap site are almost similar. Our proposed approach gives 70-80% exactly similar results as compared with real world data GPS traces.

V Conclusion

In this paper, we presented the Conversion Model, Trajectory Collector Module and Clustering Module to analyzing road network traffic of world's any regions. The proposed Conversion Model is able to provide any selected region's roadmap file as an input to the simulator. Trajectory Collector Module is implemented with simulator, which is helpful to generate trajectory files of each user. These trajectory files are used as input data for clustering module. By applying Density based clustering with the help of Clustering Module, we can able to find out most frequently used road segments of a given city (or region); which will helpful in managing traffic flows in the city.

Acknowledgment

This research was supported by the MSIP(Ministry of Science, ICT and Future Planning), Korea, under the ITRC(Information Technology Research Center) support program (NIPA-2014-H0301-14-1014) supervised by the NIPA(National IT Industry Promotion Agency). This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government (NRF-2014R1A1A1007734)

References

[1] Y. Wang, P. Coppola, A. Tzimitsi, A. Messmer, M. Papageorgiou, and A. Nuzzolo, “Real-time freeway network traffic surveillance: Large-scale field-testing results in southern Italy,” IEEE Trans. Intell. Transp. Syst., vol. 12, no. 2, pp. 548–562, Jun. 2011.

[2] Q.-J. Kong, Z. Li, Y. Chen, and Y. Liu, “An approach to urban traffic state estimation by fusing multisource information,” IEEE Trans. Intell.Transp. Syst., vol. 10, no. 3, pp. 499–511, Sep. 2009.

[3] http://openstreetmap.org

[4] GTMobiSIM, http://code.google.com/p/gt-mobisim/.

[5] Wikipedia. http://en.wikipedia.org/wiki/OSM

[6] Wikipedia. http://en.wikipedia.org/wiki/Scalable_Vector_Graphics

[7] Ester, M., Kriegel, H.-P., Sander, J., and Xu, X., \A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," In Proc. 2nd Int'l Conf. on Knowledge Discovery and Data Mining, Portland, Oregon, pp. 226-231, Aug. 1996.