

# Estimation of Urban Traffic State Using Simulation of Urban Mobility(SUMO) to Optimize Intelligent Transport System in Smart City

Mega Ayu Dian Khumara, Lubna Fauziyyah, Prima Kristalina

*Department of Electrical Engineering  
Politeknik Elektronika Negeri Surabaya  
Surabaya, Indonesia*

khumaramega@gmail.com, lubnafauzy@gmail.com, prima@pens.ac.id

**Abstract -** Traffic flow is not a strange phenomenon anymore. Currently, traffic density is a big problem faced in big cities. Increasing the vehicle amount which is not balanced with the capacity of road impact on the transportation system, especially the density of vehicles on the road. Thus, traffic monitoring is needed to minimize congestion and traffic accidents and also develop ITS (Intelligent Transport System). This paper discusses realistic vehicle mobility simulation based on traffic surveys of a city using a microscopic model simulation by using Simulation of Urban Mobility (SUMO). This paper discusses realistic vehicle mobility simulation based on traffic surveys of a city using a microscopic model simulation by using Simulation of Urban Mobility (SUMO). SUMO is an open source road traffic simulator program that allows users to build simulations of vehicle movement on network topology. In particular, the SUMO network has some information like each edge as a collection of paths including the position, shape and speed limit of each path, the logic of traffic lights sourced from the crossroads, the correct junction rules, the relationship between the lanes at the intersection. The input format which is used is demand data of survey results from the vehicles number at a certain time in each way. It can be known the pattern the traffic distribution. This traffic distribution pattern will be modeled on SUMO applications so that this application can simulate the mobility of the original vehicle in the big city. The data traffic will be managed and computed with Kalman Filter to enhance location accuracy. Then, it will be displayed on Website. Based on analysis of vehicle traffic volume in Central Surabaya, it has normal distribution traffic which is the rush hour in 4 pm. - 5 pm Based on SUMO result, Surabaya map from OSM could be integrated with SUMO GUI. adjust the speed of the road limit at Central Surabaya is the maximum speed limit on the motor, car and taxi is 33.33 m / s and from the test results obtained parameter value  $Q = 1E-04$  and  $R = 1E-04$  on Kalman filter seen that in estimating vehicle speed has the biggest percentage of error up to 49,17%.

**Keywords—** *Simulation of Urban Mobility, Survey Data, Microscopic Model*

## I. INTRODUCTION

Smart city is a "smart city" concept that focuses daily operations on Information and Communication Technology (ICT). Thus, there is an increase of effective and efficient public services. This concept has been echoed and run by several cities in Indonesia, such as Jakarta, Surabaya, and Bandung. However, the implementation of this concept on a large scale is still difficult due to constrained funding problems.

Traffic is a means to move from one place to other places. Currently, traffic density is a big problem faced by big city areas. This is caused by increasing volume of vehicles in each year that is not balanced with road capacity [1]. Some problems that arise due to traffic density include the occurrence of congestion and inconvenience in driving. It causes a delay to achieve certain goals. Not only that, the negative effect of this traffic density is the accident incidences.

We recommend a new technology for the problem solution. The solution technology is Estimation of Traffic Density in Large City Areas using Urban Mobility Simulation based on Intelligent Transport System (ITS). ITS is a communication system used in Information and Communication Technology (ICT). The presence of ITS technology can improve the convenience, security, and efficiency in the delivery of traffic. Along with the development of technology, enabling website being a solution to watch over the traffic information. Traffic information can be applied in the Smart City concept to support the development of ITS on Smart Traffic Systems.. SUMO is a realistic traffic simulator. Motion models that accurately model the behavior of individual vehicles in their environment [2]. Simulation allows handling a large number of traffic problems. Each vehicle is explicitly modeled that has its own path and moves itself through the network. Traffic information is obtained integration between Open Street Map digital network adjusting and request data. Open Street Map is digital map that can be developed for free. Position information such as longitude and latitude of each path is used as vehicle's location data estimation. Then it will be conducted filtering process by Kalman Filter method where a vehicle location data will continuously updated in order to retrieve unnecessary data. Because of that, a stable vehicle location is resulted. This last result will be shown via website.

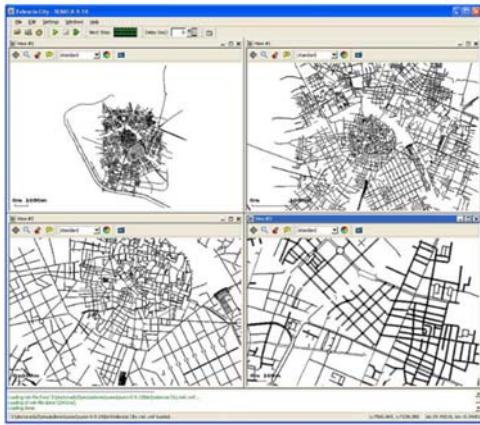


Fig. 1. Graphical User Interfaces (GUI) SUMO[3]

TABLE I. APPLICATION PACKAGE IN SUMO[3]

| Application Name      | Description  |
|-----------------------|--|
| SUMO                  | Microscopic simulation without visualization; command line application.  |
| SUMO – GUI            | Microscopic simulation with GUI  |
| NETCONVERT            | Network importer and generator; read the network path from a different format and convert it into SUMO format                            |
| NETEDIT               | Graphical network editor   |
| NETGENERATE           | Generate abstract network for SUMO simulation  |
| DUAROUTER             | Calculates the fastest route over a network, imports various types of demand descriptions. Do DUA  |
| JTRROUTER             | Calculates a route by using the percentage change of intersection  |
| DFROUTER              | Calculates the route from the induction loop measurement   |
| MAROUTER              | Perform a macroscopic task   |
| OD2TRIPS              | Decomposition of the O / D matrix into a single vehicle trip   |
| POLYCONVERT           | Importing interesting points and polygons from various formats and translating them into descriptions that can be visualized by SUMO-GUI |
| ACTIVITYGEN           | Generate demand based on mobility desires of the population model  |
| EMISSIONSMAP          | Generate emission maps.  |
| EMISSIONSDRIVINGCYCLE | Calculates emission values based on given driving cycle.   |
| Additonal Tools       | There are several tasks to write a large unnecessary application. Some solutions to different problems can be covered by this tool       |

#### A. OSM (Open Street Map)

Open Street Map (OSM) is a joint project to create an example of a world map that can be freely changed by anyone. Two supporting factors in OSM for using and

developing are the lack of availability of map information on most areas of the world and the emergence of affordable portable navigation tools. OSM is a prime example of freely assigned geographical information. The most important thing is the OSM map can be stored on the internet, and anyone can access the map anytime, for free.

OSM is described as the "Map of wikipedia", which is inseparable from the availability of mechanisms where volunteers or anyone can contribute directly to change or update geographic data to create more accurate, detailed and up-to-date maps. This happens to the arena of more than two million registered OSM editors who are constantly making changes to maps around the world. Almost every minute there is an update so the map is often more detailed than the commercial map and even a more complete area than Google Maps [4]. The example of OSM are shown at Fig. 2 and 3. Here's the command to import OSM data on SUMO:

```
Netconvert --osm-files filename.osm.xml -o
filename.net.xml [5]
```

#### B. Python Programming Language

Python is one of the open source programming language is simple, concise, and can be used in some operating systems. This recording language is becoming common to engineers around the in software generation, even some companies use python as a commercial software maker.

#### C. XML Programming Language

XML (Extensible Markup Language) is a public-purpose markup language suggested by the W3C to create markup documents for the exchange of data across multiple systems.

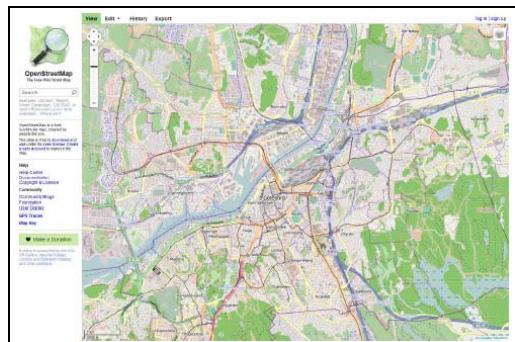


Fig. 2. OpenStreetMap Network of Gothenborg[5]

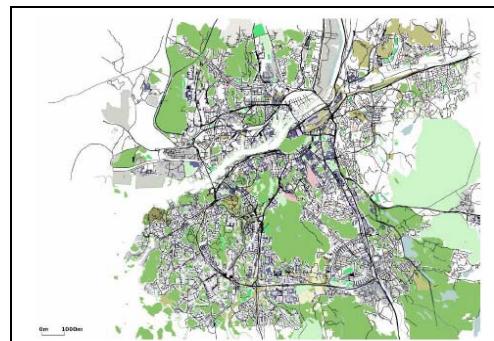


Fig. 3. Gothenborg Network is imported into SUMO[5]

XML is a continuation of HTML (HyperText Markup Language) which is the standard language for tracking the Internet. XML is designed to be able to store data in a simple and easy to be managed. The primary keywords of XML are data (plural of datum) which if processed can provide information. XML provides a standardized yet modifiable way of describing the contents of the document.

## II. METHODS

In this research, a traffic density estimation system in big city area is in Surabaya city, Central Surabaya using microscopic model simulation. Stages of design in the manufacture of this system is the creation of traffic generation system traffic using the application SUMO.

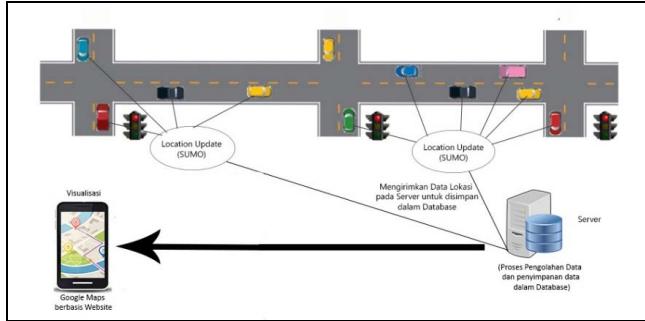


Fig. 4. System Illustration

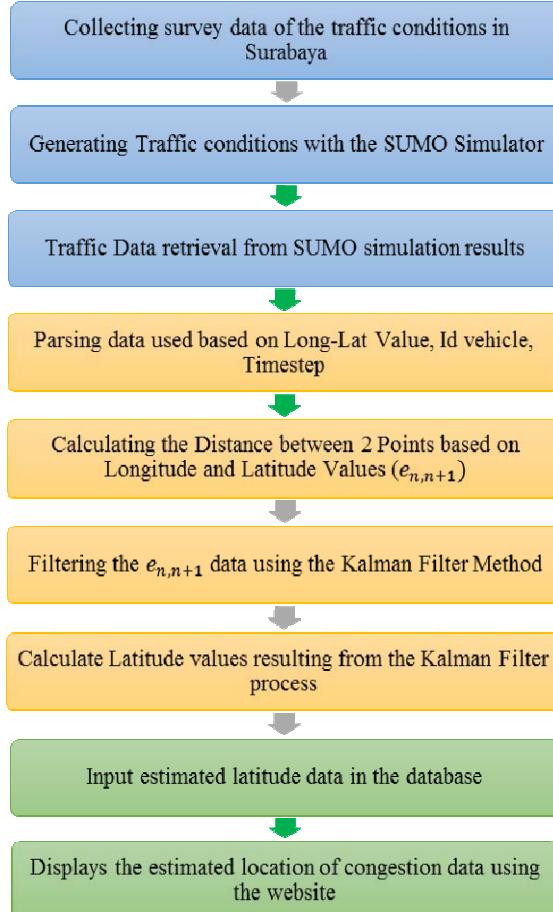


Fig. 5. Block Diagram of Proposed System

From the illustration shown in Fig. 4, the proposed system is distinguished to 2 parts, firstly SUMO is used as a simulator to generate traffic simulation of vehicles that are useful for determining the position and state of traffic, and secondly server is used to process the data obtained from the SUMO so it can be displayed on the website. In this research, the system design focuses on the generation of vehicle traffic with SUMO.

After the traffic can be generated, the frame is a time step, vehicle id, speed and position of the vehicle will be sent to the server side, so that the output of this data can be done though by kalman filter and estimation of traffic density in Surabaya can be displayed on website At the design stage of this system there are several stages of work to be done, as shown in Fig. 5.

## III. IMPLEMENTATION, MEASUREMENT AND ANALYSIS

Vehicle speed test is a vehicle speed test that is generated in vehicle traffic simulation. Conducted 10 times test for vehicle traffic simulation. In this test taken 3 samples of motor vehicles, cars, and taxis.

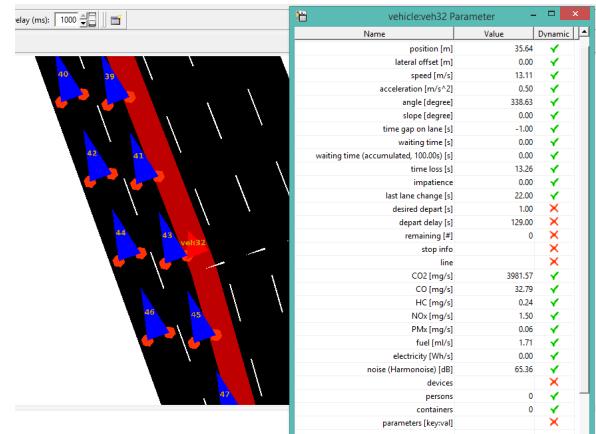


Fig. 6. The Parameter Vehicle Description of veh32.

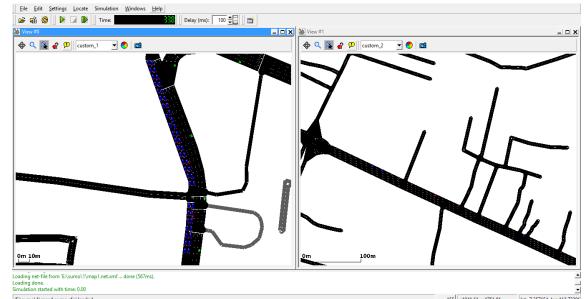


Fig. 7. Traffic Density in 3235.

From the illustration in Fig. 6. At the time of simulation for 1 hour, it can be seen that the density of occurred at a certain time occurred in Basuki Rahmat Street and Embong Malang. Here is sampled every 10 minutes. Here is the Vehicle density picture that occurred in Central Surabaya at 1 hour.

In Fig. 7. it could be seen that the generation of the road network is done in SUMO GUI app. The Road network is comprised of junctions and edges corresponding to real estate network topologies in major road areas in Central Surabaya. Road network testing is done by running or opening map.net.xml file. So that when tested repeatedly the output of the network will remain the same. Network generation is

said to succeed if the street name matches the street name is prearranged.

In the next step, the examination on how many variations of R and Q values are most appropriate. This examination was using data for 1 vehicle id only as sample. With the change of values on the parameters contained in the Kalman filter. In a filter implementation, measurement covariant noise (R) is usually measured before the filter is operated. Measurement of measurement covariance error (R) is generally possible because it can take an offline measurement sample to determine the variant of the measurement noise. The determination of the covariate noise of the Q process is generally more difficult. Sometimes a simple process model can produce good estimates with proper Q selection, when the measurement of the process is reliable.

To determine the best value of both parameters above, tested using vehicle speed data, where the value of R and Q will be varied from 10<sup>-4</sup>, 10<sup>-5</sup>, 10<sup>-6</sup>, 10<sup>-7</sup> and 10<sup>-8</sup>. This variation is done to obtain optimal estimation of the given noise. The combination of variations of parameters used can be seen in Table II.

Based on this table it can be noted that the smallest error presentation is 0.0%, that is the parameter R = 1E-08 and Q = 1E-04. So if it is shown in the graph as in Fig. 9.

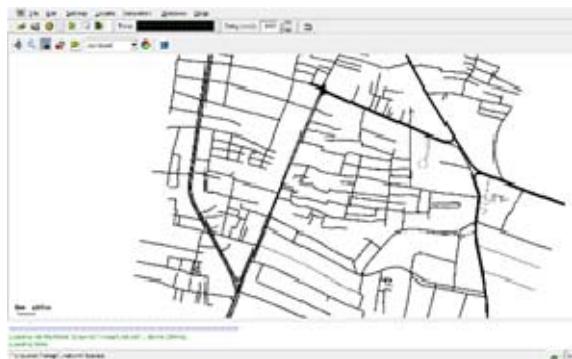


Fig. 8. SUMO Networking

TABLE II. KALMAN FILTER ESTIMATION RESULTS USING PARAMENTER VARIATION R AND Q AT SPEED 11.49 M / S FOR ID 6

| Kovarian Noise Proses (Q) | Kovarian Noise Pengukuran (R) |       |                  |       |                  |       |                  |       |                  |       |
|---------------------------|-------------------------------|-------|------------------|-------|------------------|-------|------------------|-------|------------------|-------|
|                           | 10 <sup>-4</sup>              |       | 10 <sup>-5</sup> |       | 10 <sup>-6</sup> |       | 10 <sup>-7</sup> |       | 10 <sup>-8</sup> |       |
|                           | speed                         | % Err | speed            | % Err | speed            | % Err | speed            | % Err | speed            | % Err |
| 10 <sup>-4</sup>          | 10.086                        | 12.2% | 11.277           | 1.85% | 11.466           | 0.2%  | 11.487           | 0.02% | 11.489           | 0.0%  |
| 10 <sup>-5</sup>          | 7.241                         | 36.9% | 10.086           | 12.2% | 11.277           | 1.85% | 11.466           | 0.2%  | 11.487           | 0.02% |
| 10 <sup>-6</sup>          | 5.978                         | 47.9% | 7.241            | 36.9% | 10.086           | 12.2% | 11.277           | 1.85% | 11.466           | 0.2%  |
| 10 <sup>-7</sup>          | 5.806                         | 49.4% | 5.978            | 47.9% | 7.241            | 36.9% | 10.086           | 12.2% | 11.277           | 1.85% |
| 10 <sup>-8</sup>          | 5.788                         | 49.6% | 5.806            | 49.4% | 5.978            | 47.9% | 7.241            | 36.9% | 10.086           | 12.2% |

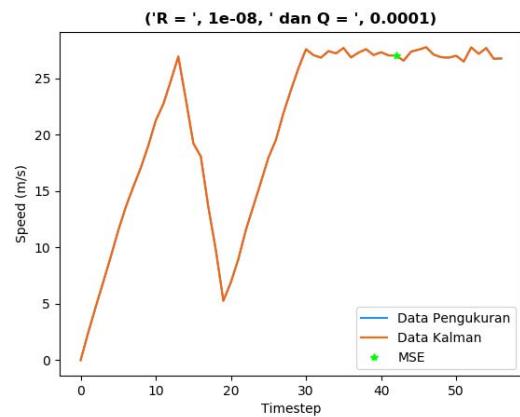


Fig. 9. Speed Estimation Results with R = 1E-08 and Q = 1E-04

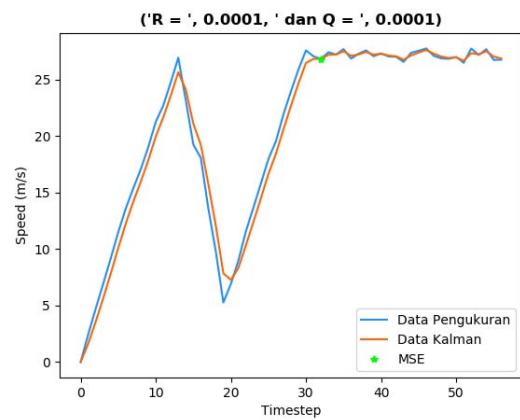


Fig. 10. Graph of Kalman Filter with parameters R = 0.0001 and Q = 0.0001



Fig. 11. Graph of Comparison between% Error with Vehicle Speed Increase

However, the principle of the Kalman Filter algorithm used in this Final Project is to process the simulated data to obtain a more constant value and with small error. So the best variation of R and Q parameters to be applied in this Final Project is R = 0.0001 and Q = 0.0001. After testing the value of R and Q Kalman Filter, added the search value closest to the calculation value, ie MSE (Mean Square Error). In Fig.10 the most closely approximated value of measurement is when it is at a speed of 26.85 m / s, which is the smallest MSE value. With this MSE it can compare the data of the kalman process with the measurement data.

After testing the best combination of R and Q values that will be used in the estimation process with Kalman Filter, filtering process is done on the data  $e_{(n, n+1)}$  by applying R and Q in this Final Project of R = 0.0001 and Q = 0.0001. Then the system will generate a new  $e_{(n, n+1)}$  value,

which is the output of the Kalman Filter process. Based on the value of  $e_{-}(n, n + 1)$  recently will do the process of selecting new Latitude values. So the value of this new latitude will be the predicted value of the vehicle location.

At this stage, simulation samples have been conducted for 10 types of vehicles, namely for motorcycles, cars and taxis. Testing data is done in the span of 1 hour (city rush hour surabaya) that is at 16.00-17.00 hours. Fig.12 is the result of the comparison of actual Latitude value with Latitude prediction from 1 motor sample.

Based on Fig. 12, the most closely matching value search is MSE (Mean Square Error). The value closest to the measurement value is when it is at latitude -7.255987, which is the smallest MSE value. With this MSE it can compare predictive latitude data with actual latitude data.

Fig.13 shows the comparison graph between the latitude with longitude so that it can be known that the route has been traversed by motor in the span of 1 hour. From this figure it can be noted that the actual route through which the motor over an hour of solid Surabaya is not much different from the prediction route. Motor 1 is flown from Jl. Basuki Rahmat then passes Embong Malang Street, and last through Blauran Street. Furthermore, from the data and graphics above will be easier to be mapped in a Web that is integrated with Google Maps.

Fig. 14. is illustrating the output of the actual Latitude value with Latitude prediction from 1 sample of car.

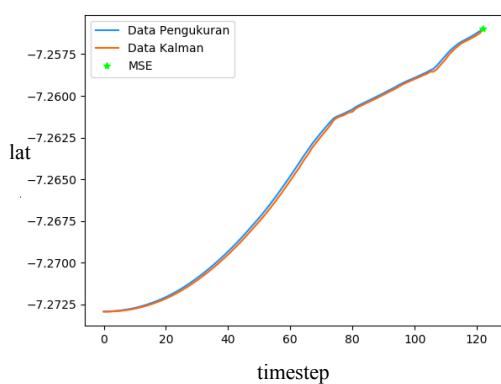


Fig. 12. Graph of the actual Latitude score with Latitude prediction

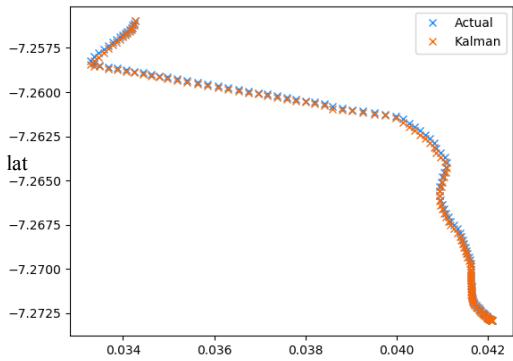


Fig. 13. Graph between Longitude-Latitude for motor

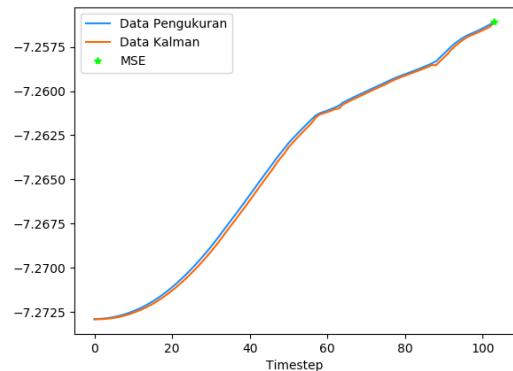


Fig. 14. Graph of the actual Latitude score with Latitude prediction

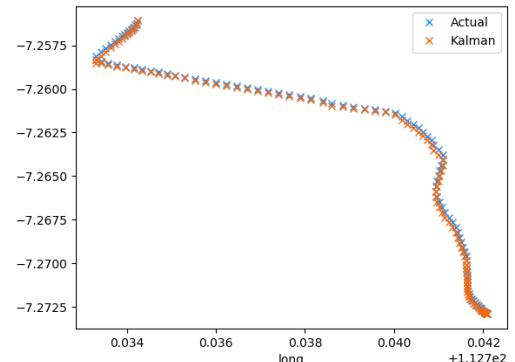


Fig. 15. Graph between Longitude-Latitude car

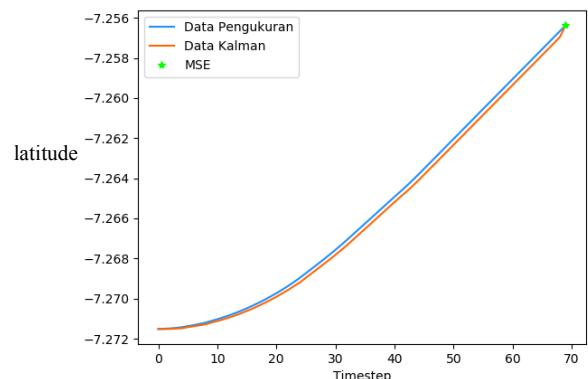


Fig. 16. Graph of the actual Latitude score with Latitude prediction

Based on Fig. 14., the most closely approximated value search is MSE (Mean Square Error). In Fig.14 the value closest to the measurement value is when it is latitude -7.256086, which is the smallest MSE value. With this MSE it can compare predictive latitude data with actual latitude data.

Then in Fig.15 shows the comparison graph between the latitude with longitude so it can be known route that has been dilelwati car in the span of 1 hour. From this figure it can be noted that the actual route that the car passes for 1 hour in Surabaya is not much different from the prediction route. Car 1 is reddish from Jl. Basuki Rahmat then passes Jl. Embong Malang, and last through Jl. Blauran. So from the data and graphics above will be easier to be mapped in a Web that is integrated with Google Maps.

Fig.16 is the result of the comparison of actual Latitude values with Latitude predictions from 1 taxi sample. Based on this figure, we add the search value closest to the calculation value, ie MSE (Mean Square Error). In Fig.16 the value closest to the measurement value is when it is at

latitude -7.256367, which is the smallest MSE value. With this MSE it can compare predictive latitude data with actual latitude data.

Then in Fig.17 shows the comparison graph between the latitude with longitude so that it can be known that the route has been dilelwati taxis within the span of 1 hour. From this figure it can be observed that the actual route through which the motor passes for 1 hour in Surabaya is not much different from the prediction route. Car 1 is reddish from Jl. Basuki Rahmat then passes Jl. Embong Malang, and last through Jl. Blauran. So from the data and graphics above will be easier to be mapped in a Web that is integrated with Google Maps.

The result of the targeted target position is visualized on the Website. This test is performed to determine the accuracy of the coordinates at the actual location. Testing is done using a Website that has been integrated with Google Maps to know the position of the vehicle.

The input for the map is the value of the calculations that have been done before. The value is stored in \*.Txt file which is then inputted to the website. The file will be stored in the mysql database. The data stored in the database and displayed on the website in the form of data longitude-latitude taken from the database.

The scenario of this test consists of 3 sample vehicles each for 1 motor, 1 car and 1 taxi. Where the routes passed by each vehicle are as follows:

1. Motor 1 through: Basuki Rahmat Street- Embong Malang Street- Blauran Street
2. Car 1 via: Basuki Rahmat Street- Embong Malang Street- Blauran Street
3. Taxi 1 via: Pasar Kembang Street- Arjuno Street

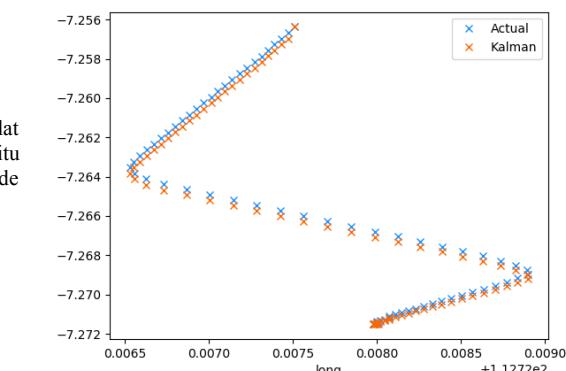


Fig.17. Graph between Longitude-Latitude motor

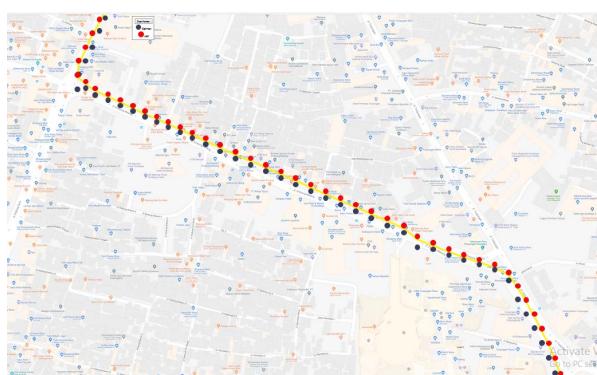


Fig.18. Routes by motor 1



Fig.19. Route by car 1

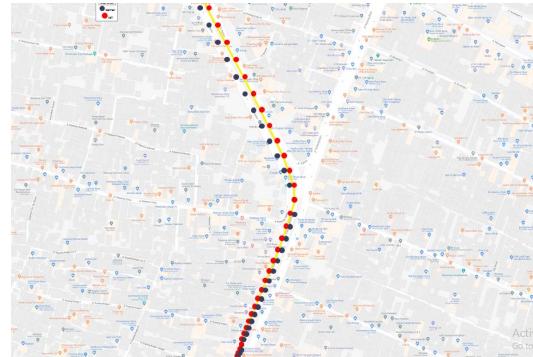


Fig.20. Routes bypassed by taxi 1

TABLE III. ACCURACY ROUTES WHICH IS PASSED BY MOTOR 1

| No | Accuracy             | Route |
|----|----------------------|-------|
| 1  | Basuki Rahmat Street | ✓     |
| 2  | Empong Malang Street | ✓     |
| 3  | Blauran Street       | ✓     |

TABLE IV. ACCURACY ROUTES WHICH IS PASSED BY CAR 1

| No | Accuracy             | Route |
|----|----------------------|-------|
| 1  | Basuki Rahmat Street | ✓     |
| 2  | Empong Malang Street | ✓     |
| 3  | Blauran Street       | ✓     |

TABLE V. ACCURACY ROUTES WHICH IS PASSED BY TAXI 1

| No | Accuracy             | Route |
|----|----------------------|-------|
| 1  | Pasar Kembang Street | ✓     |
| 2  | Arjuno Street        | ✓     |

The evaluation has been done by using 3 types of vehicles, then compared the data of longitude and latitude with the actual condition at the location passed at the time of simulation. Comparison data can be seen in the table as follows:

#### IV. CONCLUSION

From the results of implementation, testing and system analysis in the previous chapter, it can be taken some conclusions such as maps used in SUMO are derived from global topolgi data retrieval from OpenStreetMaps (OSM). To generate traffic using SUMO it is necessary to manually adjust the settings for the julman of the vehicle, the speed of

the vehicle and the route to be bypassed from the vehicle. based on the test results obtained parameter value  $Q = 1E-04$  and  $R = 1E-04$  on Kalman filter seen that in estimating vehicle speed has the biggest percentage of error up to 49,17%. And coordinate accuracy with the original condition is appropriate, then change the data longitude and latitudenya depending on the speed of the vehicle and traffic conditions on the street.

#### REFERENCES

- [1] Dinas Perhubungan Pemerintah Kota Surabaya. (2012). “Dibalik Berdirinya SITS”. <http://sits.dishub.surabaya.go.id/ver2/tentang-sits>. Accessed on Thu, June 1, 2017
- [2] Francisco J. Martinez, Chai Keong Toh, Juan-Carlos T. Calafate, Pietro Manzoni, “A Survey and Comparative study of Simulator for Vehicular ad hoc Networks (VANET)”, Wireless Communications and Mobile Computing, 2009.
- [3] <http://SUMO.dlr.de>, official site SUMO, Accessed on Thu, Jul 13, 2017.
- [4] Rahardjo Ilmal Alfriansyah, “Studi Kinerja 802.11 pada Protokol Adhoc On-Demand Distance Vector(AODV) di Lingkungan Vehicle AD HOC Network (VANET) menggunakan Network Simulator NS2” Institut Teknologi Sepuluh November Surabaya, 2017.
- [5] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker, “Recent Development and Applications of SUMO – Simulation of Urban Mobility”, International Journal Advances in Systems and Measurements, December 2012.
- [6] Ahmad Faisal Abidin, Mario Kolberg, ”Towards Improved Vehicle Arrival Time Prediction in Public Transportation: Integrating SUMO and Kalman Filte”, International Conference on Modeling and Simulation, 2015.
- [7] Lara Codeca, Raphael Frank, Sebastian Faye dan Thomas Engel, “Luxembourg SUMO Traffic (LuST) Scenario: Traffic Demand Evaluation”, IEEE Intelligent Transportation System Magazine, 2017.
- [8] Carolina, Anita., “Data Hasil Survei Traffic Accounting”, Institut Teknologi Sepuluh November Surabaya, 2014.