Level Of Services Impact Prediction with Multiclass SVM On Vehicle Speed

Khusnan Hadi Eka Panca D.[1], Surya Michrandi Nasution[2], Agus Virgono[3]
[1]Telkom University,Indonesia,khusnanhadi@student.telkomuniversity.ac.id
[2]Telkom University,Indonesia,michrandi@telkomuniversity.ac.id
[3]Telkom University,Indonesia, avirgono@gmail.com

ABSTRACT

As times develop, more and more vehicle users in big cities. But this was not well anticipated by the city government. So congestion is a common thing in big cities. There have been many solutions to overcome congestion, but it is still not effective because the application is still not perfect. But congestion can be avoided if the congestion can be predicted. Traffic prediction can be done with the SUMO application (Simulation Urban Mobility). The simulation results will be processed by python with the SVM multiclass method. The processed product will become a reference for predicting congestion. The best result from prediction is 100% accuracy in scenario with 90% train data and 10% of test data.

Key words :congestion, Multiclass SVM, SUMO,Traffic.

1. INTRODUCTION

Traffic is a problem for developing countries, especially Indonesia. Traffic in major cities in Indonesia is in the spotlight due to congestion that is difficult to overcome, especially in Kota Bandung. Kota Bandung has complicated traffic. The complexity is caused by the layout of the city that is not well ordered and coupled with the number of residents in the city of Bandung. The population movement in the city of Bandung is very massive, especially at certain times such as rush hour in the morning.

Congestion in Kota Bandung gives a drawback to residents and the environment in the Kota Bandung. The forms of losses that occur are increasing travel time to reach the destination and increasing air pollution in the city of Bandung so that it disrupts the health of the population. therefore we need a real solution to reduce congestion in the city of Bandung[1].
In several major cities in the world such as Sao Paulo and Shanghai, several optimal solutions have been found to overcome congestion. In the city of Sao Paulo, a solution was found to reduce congestion by simulating the transfer of private vehicle users to public transportation, in this case, is Subway[2]. In the city of Shanghai, a simulation was carried

out to rearrange the placement of traffic signs in the city. This rearrangement aims to increase the average speed of vehicles on the city's streets. From those two studies, it was found that the results were quite significant for breaking down congestion and reducing pollution in these two cities[3]. Not only to free the traffic flow, by reducing congestion can have an impact to reduce the number of accidents. High accident rates not only cause casualties, but also affect a country's economy. In India, it lost 2.5 percent of GDP due to road accidents [4]. In the future many electric vehicles will also replace conventional vehicles that use fossil fuels. With the development of electric vehicles in the future, more intelligent systems are being implemented in the vehicle, one of which is a system to predict traffic flow [5].

2. LITERATURE STUDY

2.1 Simulation Of Urban Mobility (SUMO)

SUMO is an application to simulate traffic in urban areas. SUMO is an open-source application developed by employees of the Institute of Transportation Systems at the German Aerospace Center. SUMO was first developed in 2000. The purpose of this development is to support the traffic research community with the ability to implement their algorithms in their use [6].
This application will later be used as a simulation tool for traffic. Sumo has the advantage of each unit being simulated explicitly, which means that each unit has different characteristics among the other units. This makes it easy to conduct simulations that require high accuracy. SUMO has several features such as vehicle movement based on space-continuous and time-discrete. SUMO can also simulate up to 10,000 roads. This application has been used to evaluate traffic lights, determine route and route search, as well as simulate communication between vehicles [7].

2.2 Multiclass Support Vector Machine (M-SVM)

Multiclass Support Vector Machine (M-SVM) is one of the supervised algorithms in machine learning. M-SVM can be applied to classification and regression. SVM is used to divide datasets into two conditions, but with M-SVM datasets can be

divided into more than two conditions. An important component in M-SVM is a hyperplane. A hyperplane is a dividing line that divides the dataset into two conditions[8].

M-SVM itself is divided into two types, indirect and direct. In the indirect classification process must divide several processes, while indirect classification can be done in one efficient process. M-SVM is also divided into two types in the prediction process, namely One Vs One and One Vs Rest. In SVM there is also a function called kernel.

The reason for using msvm is a pretty good accuracy based on a journal that discusses spam detection[9]. In the journal the results obtained from using svm produce an accuracy value of 97.33%. This value is the highest value compared to the accuracy value in other methods such as KNN. Not only based on the journal, the reason for using MSVM is also based on the journal that discusses the prediction of the arrival of public transportation[10]. The similarity about traffic demand variation in that paper is the main reason for using M-SVM in this project.

2.3 Level Of Services

Level of Services (LOS) is a benchmark to assess the performance of a road section. Determination of a LOS based on the value of volume/capacity ratio (V / C ratio). If the V / C ratio is getting closer to 1 then the performance on a road segment will be lower.

Each road section has a different V / C ratio [11]. The type of LOS used in this Final Project is the Level of Service depending on the flow. This level of service assesses a road segment based on vehicle speed on a road or road facility [12].
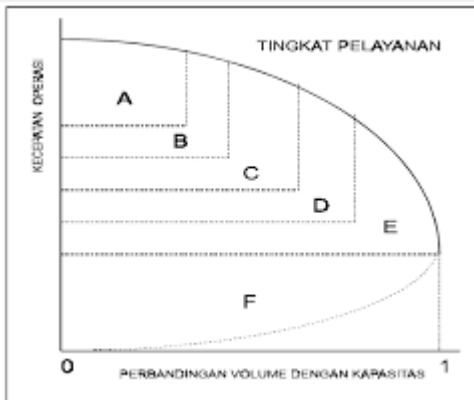

Figure1: Level Of Services [8]

3. Design And Testing

3.1 System Overview

In this project, a simulation will be created to simulate the impact of each level of services on vehicle speed. To run the simulation we need vehicle data that captured from Jl. R. E. Martadinata Kota Bandung. The data must be processed first before used it into the simulation. to achieve the level of services from the data, the prediction system must be created. The method for the prediction system is Multiclass SVM. Here the overview of how the system work:
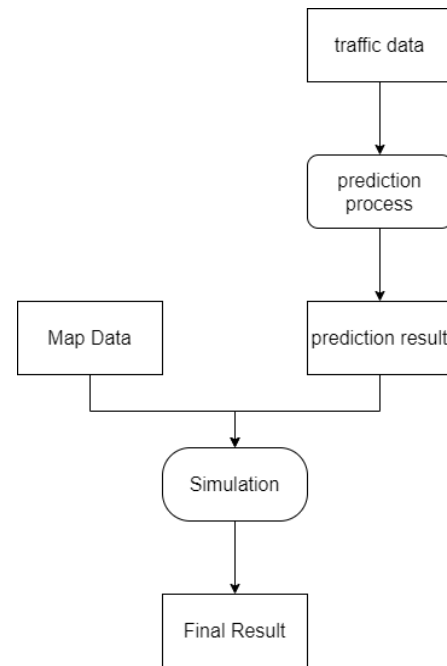

Figure 2: System Overview

3.2 Simulation Creation In SUMO Application.

In the process of traffic simulation in the SUMO application is done in several steps. In the first stage to carry out simulations the necessary data were collected, such as the R.E Martadinata road map, as well as historical traffic data on the road. To get the R.E Martadinata road map data, go to the OpenStreetMap website.

Maps that have been obtained in OpenStreetMap must be changed to the format .net.xml. this is intended so that the SUMO application can read the map. After the map file has been converted to the .net.xml format, route creation can be done. In making this route file can be done in several ways, one of which is using tools from the SUMO application, namely "randomtrips.py". The output from the process of determining this route is a file with the format ".rou.xml". To combine the file ".net.xml" with ".rou.xml" must be made SUMO config file. The file creation also determines the duration of the simulation process that will run on the SUMO application

3.3 Implementation Of Prediction System

The first process carried out is the creation of a system for the prediction process. engine created using the SVM multiclass method. In making engines with the SVM Multiclass method, we must pay attention to the distribution of data to be used. Because if you do not pay attention to the spread of data, the engine accuracy results are not good. If the data distribution is known, then it can determine the proper use of the kernel.
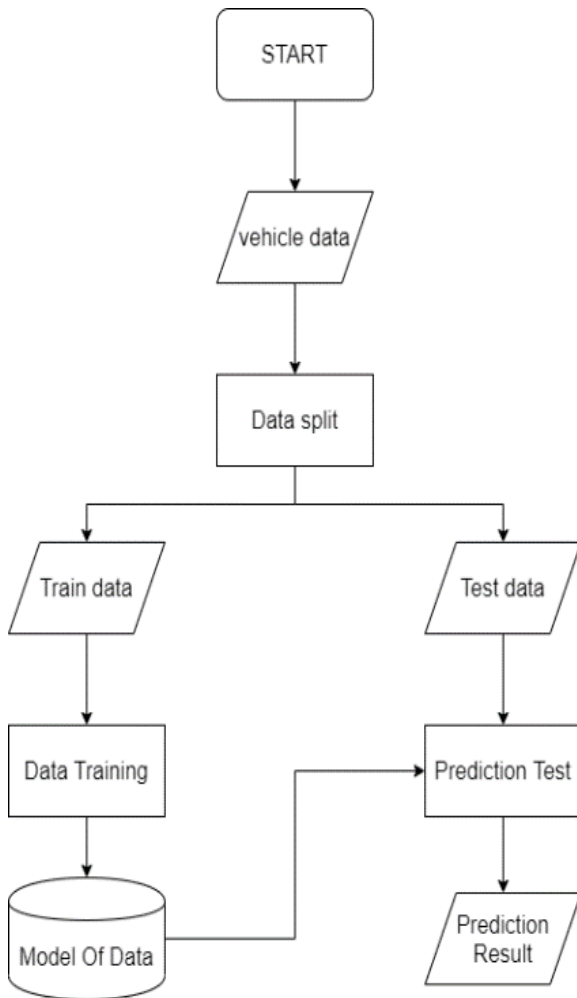
Figure 3: Flowchart of Prediction System

After making the engine is complete the process to split the dataset into training data and test data can be done. In SVM the more training data, the better the prediction results. At the training data processing stage, determining the kernel is very important. The kernel used must be able to create hyperplanes that share data well. So the data that has been divided into classes does not have a lot of noise. When this process is complete, the data will be stored for testing the test data

3.4 Combining The Result Of Prediction System Into SUMO

After the prediction system can generate a result, the simulation can create a vehicle route based on the prediction result. The prediction result must be converted into a route file so the SUMO application can use it to running in the simulation. here how the system work:
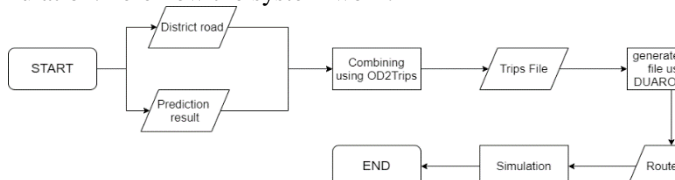

Figure 4: Flowchart of Combining Process

To create the district road file, just assign a road id from the map in the simulation. to combine those files, the OD2Trips tools can handle the job. OD2Trips generate a Trips file. The trips file must be converted again into the route file. To do this, use DUAROUTER tools provided by the SUMO application. From here the simulation can be running. To running the simulation using the route file from the prediction result, just insert the route file name into the SUMO configuration file. The SUMO configuration can be running more than one route file simultaneously.

3.5 Result

Table 1: Result from Prediction System

No	Train Data	Test Data	Kernel	Accuracy
1	50%	50%		99.6%
2	60%	40%		99.4%
3	70%	30%	Linear	99.3%
4	80%	20%		99.61%
5	90%	10%		100%

The result from the prediction system proves that the prediction process using multiclass SVM with Linear kernel gives good accuracy in scenario with 90% of train data and 10% of test data. But there are signs of overfitting in the prediction process. To see more clearly about the overfitting, the decision boundaries graphic can describe it.
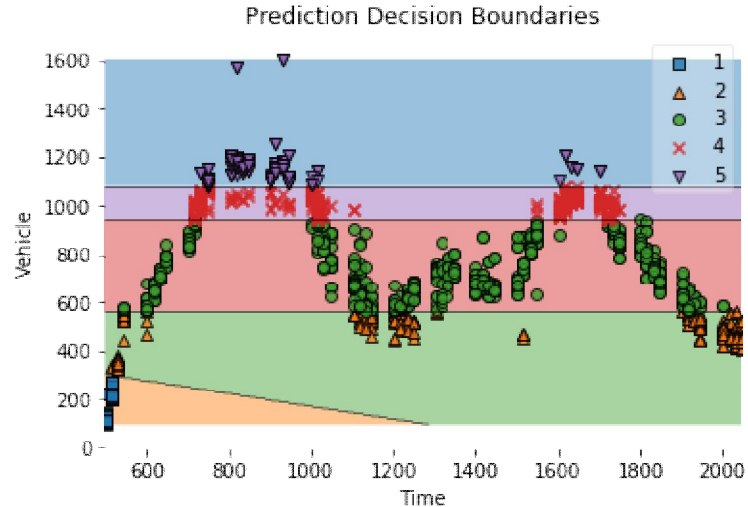

Figure 5: Decision Boundaries

The decision boundaries give a result that the classification process can make a good hyperplane to separate every class. This is the reason why the accuracy nearly perfect. But it gives a glimpse that the data has an overfitting condition.
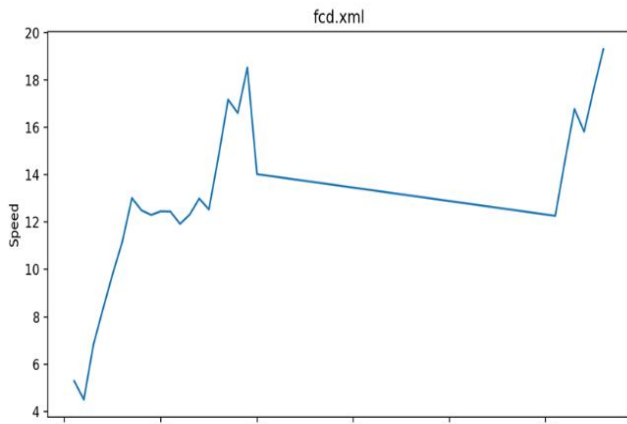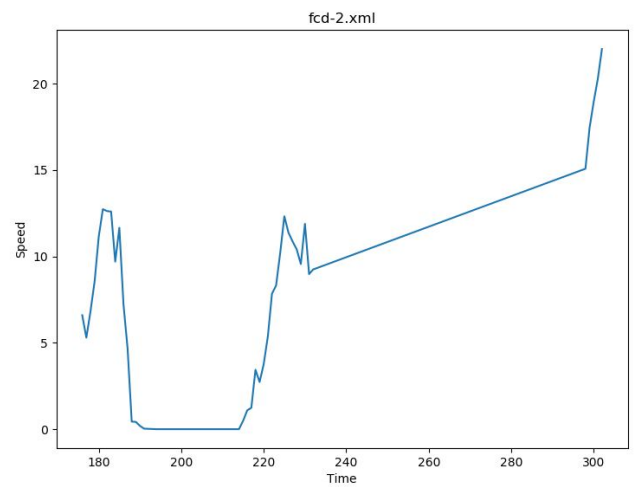
Figure 6: Speed VS Time of LOS A
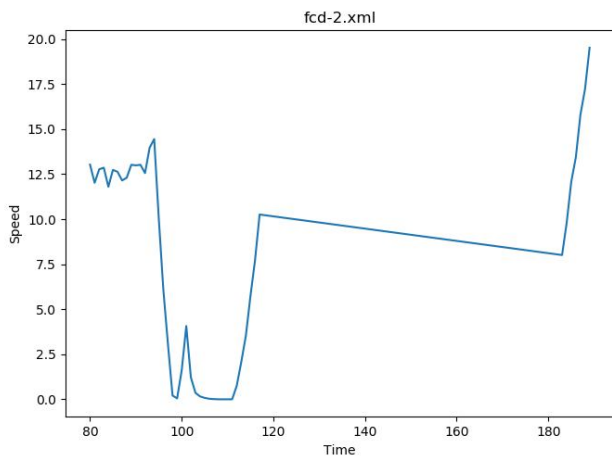

Figure 9: Speed VS Time of LOS D

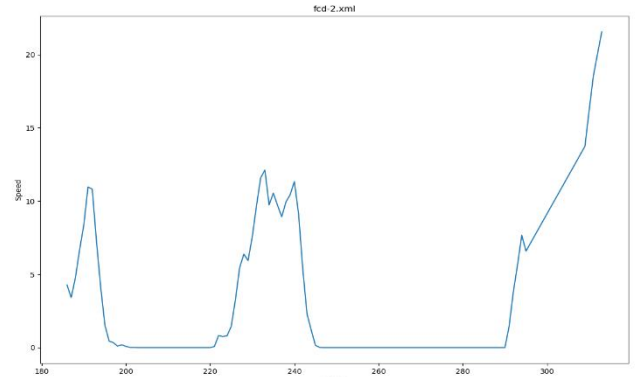**Figure 7:** Speed VS Time of LOS B
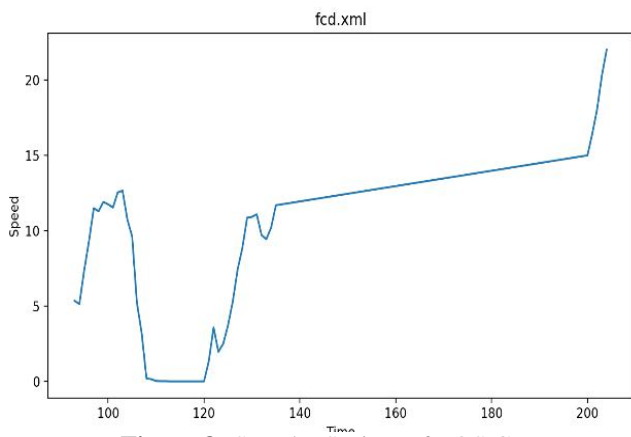

Figure 10: speed VS time of LOS E

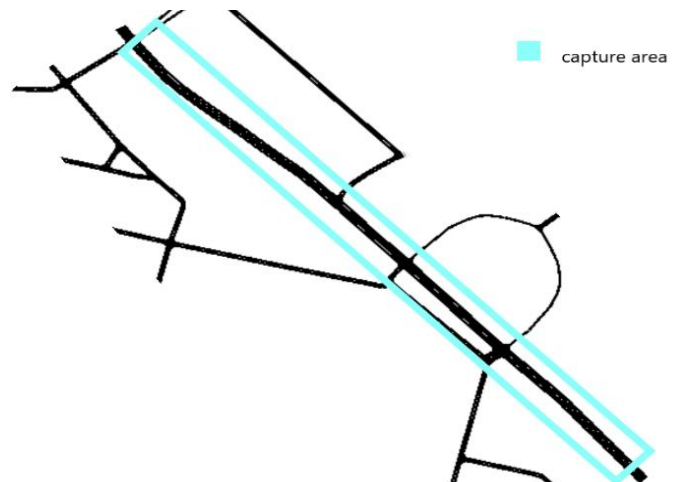**Figure 8:** Speed VS Time of LOS C

Figure 11: Capture Area for Creating The Graphic of LOS

From the graph above, it shows every difference in speed and time of each class from the level of services. But for maximum speed in each class it looks almost the same at around 20. But the average speed of each class is different. This can be seen from the rise or fall of the graph in each class. To make these graphs done by recording data from one of the cars that will be simulated in the SUMO application. The car used as the sample has several different attributes from other cars that will be simulated. The difference lies in the ID name

used and the time of deployment. For the deployment time is set at Time 50. So if during the process of simulating the vehicle speed on the road smoothly, the sample car will enter the simulation at Time 50. However, if the vehicle speed on the road is not smooth or even jammed, then the sample car will experience a delay to enter the road.
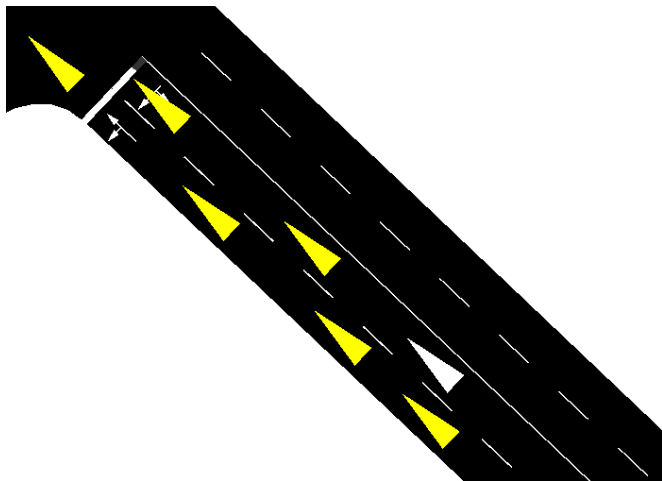

Figure 12: the sample car in simulation (white triangle)

Every delay that occurs in every LOS is different. For los A The travel time starts from 50 to 100. Whereas for LOS E it has a travel time that starts from 180 to 300. From the comparison of the two graphs it can be seen that LOS E has a lot of delay.

Not only delay can be seen from each of the los graph above, for the average speed can also be seen from a decrease in speed. In the Los B chart there can be seen speeds that are 0 from Time 100 to 110. While in the Los D chart it is seen that the velocity is 0 starting from 190 to 210. From the two comparisons of the LOS charts it can be concluded that the LOS D graph has many stops.
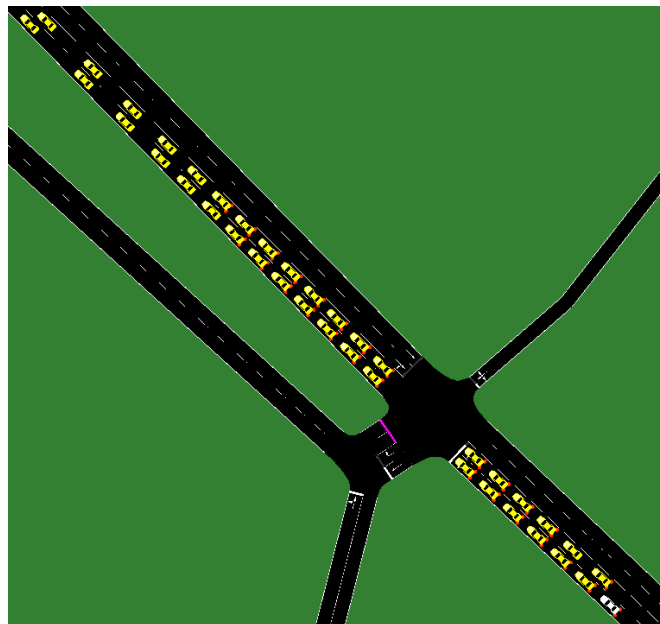

Figure 13: simulation process for LOS B

Figure 14: simulation process for LOS D

4. CONCLUSION

From the results of simulations performed by the sumo application, graphs can be obtained for each LOS class. Each graph generated from a sumo simulation application has a different value - different. For charts with good travel time and with very little delay owned by the LOS A graph. For graphs that have poor travel time and lower average speeds owned by the LOS E. graph this proves that the number of vehicles is very influential in service from a road. For prediction systems using the M-SVM method the results are obtained with 100% accuracy with 90% training data sharing conditions and 10% test data. From these results it can be seen that there is an overfitting condition caused by the data used.

REFERENCES

1. A. J. Jawaami, "PenyebabKemacetan Utama di Kota Bandung," 30 Desember 2017. [Online]. Available: https://www.ayobandung.com/read/2017/12/30/26964/ini-penyebab-kemacetan-utama-di-kota-bandung.
2. M. B. L. B.-W. Pablo Alvarez Lopez, "Microscopic Traffic Simulation using SUMO," dalam IEEE Intelligent Transportation Systems Conference (ITSC), Maui, 2018..
3. E. F. Z. Santana, L. Kanashiro dan B. Tomasiello, "Analyzing Urban Mobility Carbon Footprint with Large-scale, Agent-based Simulation," dalam 7th International Conference on Smart Cities and Green ICT Systems , Madeira, 2018.
4. C. R. Keerthi, G.Shanmukh and R.Sivaran, "Various Accident Detection Technologies and Recovery," International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), vol 2, pp. 7 - 12, 2013.

5. J. Ahmad, I. Musirin, M. K. M. Zamani, S. A. S. Mustafa, N. Aminuddin, S. A. Shaaya and H. Suyono, "Evolutionary Programming Based Technique for Plug-in-Hybrid Electric," International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), vol. 8, pp. 425 - 430, 2019.

6. T. Azevedo, P. J. M. de Araújo, R. Rossetti dan A. Rocha, "JADE, TraSMAPI and SUMO: A tool-chain for simulating traffic light control," dalam Proceedings of the 8th International Workshop on Agents in Traffic and Transportation, ATT'14, held at the Thirteenth International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS'14, Volume: 1, paris, 2014.

7. D. Krajzewicz, G. Hertkorn, C. Feld dan P. Wagner, "SUMO (Simulation of Urban MObility); An open-source traffic simulation," dalam 4th Middle East Symposium on Simulation and Modelling, Sharjah, 2002Systems, I. B. (2012) 'Improving Barcode Detection with Combination of Simple Detectors', pp. 0–6. doi: 10.1109/SITIS.2012.52.

8. V. Vapnik, "Stastical Learning Theory.," dalam Wiley-Interscience, New York, 1998.

9. S. I. Manzoor and D. J. Singla, "A Comparative Analysis of Machine Learning Techniques for Spam Detection," hfaq Manzoor et al., International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), vol. 8, pp. 810-814, 2019.

10. Y. Bin, Y. Zhongzhen and Y. Baozhen, "Bus Arrival Time Prediction Using Support Vector Machines," Journal of Intelligent Transportation Systems, vol. 10, no. 4, pp. 151-158, 2006.

11. DirektoratJenderal Bina Marga, Manual Kapasitas Jalan Indonesia, Jakarta: Bina Karya, 1997.

12. O. Z. Tamin, Perencanaan dan PemodelanTransportasi, Bandung: Penerbit ITB, 2000