IEIE Transactions on Smart Processing and Computing, vol. 6, no. 4, August 2017
https://doi.org/10.5573/IEIESPC.2017.6.4.281

Traffic Flow Estimation System using a Hybrid Approach

Swe Sw Aung*, Itaru Nagayama, and Shiro Tamaki

Information Engineering Department, University of the Ryukyus, Okinawa, Japan

***** Corresponding Author: Swe Sw Aung, sweswe@ie.u-ryukyu.ac.jp

Received February 7, 2017; Revised April 11, 2017; Accepted May 31, 2017; Published August 30, 2017

** Regular Paper*

** Extended from a Conference: Preliminary results of this paper were presented at ITC-CSCC 2016. This paper has been accepted by the editorial board through the regular review process that confirms the original contribution.*

***Abstract*:** Nowadays, as traffic jams are a daily elementary problem in both developed and developing countries, systems to monitor, predict, and detect traffic conditions are playing an important role in research fields. Comparing them, researchers have been trying to solve problems by applying many kinds of technologies, especially roadside sensors, which still have some issues, and for that reason, any one particular method by itself could not generate sufficient traffic prediction results. However, these sensors have some issues that are not useful for research. Therefore, it may not be best to use them as stand-alone methods for a traffic prediction system. On that note, this paper mainly focuses on predicting traffic conditions based on a hybrid prediction approach, which stands on accuracy comparison of three prediction models: multinomial logistic regression, decision trees, and support vector machine (SVM) classifiers. This is aimed at selecting the most suitable approach by means of integrating proficiencies from these approaches. It was also experimentally confirmed, with test cases and simulations that showed the performance of this hybrid method is more effective than individual methods.

***Keywords*:** Decision tree, Logistic regression, Support vector machine

1. Introduction

Traffic congestion is not an unusual problem; however, it is an everyday issue to people in their daily lives as they rely on the road network for transportation. Consequently, people gradually become impatient and feel "road rage."

From the perspective of healthcare, facing traffic jams every day is similar to a kind of disease that erodes human health. For that reason, many researchers have been seeking brilliant solutions that aim at reducing road traffic jams, as well as giving more detailed traffic information that is timely and correct.

For the purpose of providing traffic information to the public in more detail and greater accuracy, this paper estimates traffic conditions at three levels, ***Red***, ***Yellow*** and ***Green***. By knowing real-time detailed traffic information about a road, people can make the right decision before taking a specific road to their destination. In this work, we first explore the prediction accuracy of three algorithms (decision trees, support vector machine [SVM] classifiers, and multinomial logistic regression) and then take deep learning in estimation of the three traffic levels to get a perspective insight into traffic conditions. After that, based on those facts, this paper proposes a hybrid approach to advance the prediction accuracy of traffic conditions ***Red***, ***Yellow***, and ***Green*** to a higher level.

Moreover, to improve prediction accuracy, this system estimates the traffic conditions based on real traffic data, as well as other factors that can cause heavy traffic jams (for example, weather conditions, special days, rush hour or normal time, weekdays and weekends). If traffic systems operate only based on the data from only one resource, like roadside cameras, the outcome is unlikely to produce better results and will then fail at prediction accuracy. Especially for situations like heavy rain, correct and vivid data may not be captured by a camera. To overcome the issue discussed above, this paper took into account many characteristics and attributes of traffic jams.

These attributes are labeled in the format in expression (1):

<NumberofVehicles, SpecialDay, WeatherCondition, CurrentTime, TypeofDay > (1)

For a more detailed discussion of traffic evidence, there are many situations that cause traffic jams to happen. Among them are bad weather conditions, which are most likely to cause heavy traffic jams. Another is special days, such as festivals. During these days, traffic jams can occur due to the large number of people on the streets. For example, during Naha Festival in the Okinawa region in Japan, the roads connecting to this area are almost completely blocked by cars. From our experience, traffic congestion is more likely to occur on weekdays than on weekends. The traffic situations described above seriously affect downtown roads and disturb drivers. According to the analysis above, there is a link between traffic jams and the environment. Because of the data from roadside cameras, under these circumstances, better results could be predicted than by using stand-alone functions.

The rest of this paper is organized as follows. Section 2 describes related work. Section 3 presents data collection and representations. Section 4 details theoretical descriptions of machine learning models, and how to apply these models to our traffic prediction system, and Section 5 discusses experimental results of each model, comparing these models to the hybrid approach. Finally, Section 6 is the conclusion and suggests future work.

2. Related Work

Researchers have been implementing traffic flow estimation models by employing many different kinds of appropriate approaches, such as supervised or unsupervised learning, image processing, etc., taking into account different kinds of factors, such as weather conditions, weekdays or weekends, rush hour times and special days, and employing data from different kinds of sources, like roadside cameras, loop detectors, and GPS-enabled vehicles and mobile devices. Among them, the camera detector plays an essential role in tracking real-time traffic information in the intelligent transportation system research field. Moreover, the video streaming processing for traffic congestion is a critical one in controlling city traffic. Osenbaum et al. [1] designed a system that extracts traffic data from a roadside camera with previously recorded information from a road database of the approximate location of road axes in geo-referenced and ortho-rectified images. First, vehicle detection is executed, and then velocity is obtained by applying a vehicle tracking technique. In this case, it is obvious that these authors mainly focused on only one resource in order to predict the traffic condition. Niksaz [2] proposed a system that estimates the amount of traffic on highways by utilizing one of the image processing techniques, background subtraction, to determine the number of cars. Before this task, each frame is compared with the first frame, and if the number of cars is more than a threshold, it assumes there is traffic. However, the system predicted traffic congestion by not considering any other effect that can cause it. Hashemi et al. [3] proposed classification models in terms of if-then rule sets for short-term traffic prediction of a highway section by applying supervised data-mining learning algorithms: classification tree, random forest, naïve Bayes, and CN2. According to the results in that paper, classification tree and random forest methods have the highest quality in prediction accuracy compared to naïve Bayes and CN2. Average prediction accuracy by utilizing these classification methods was 80%. Lu et al. [4] designed a system for tracking vehicles based on speeded-up robust features (SURF) and the local binary pattern aiming at providing more efficient and robust results than the classical approaches. Gao et al. [5] proposed a robust algorithm to detect real moving vehicles and eliminate the influence of shadows and vehicle headlights by using a support vector machine (SVM).

Short-term traffic prediction systems are very popular in intelligent transportation systems aimed at precisely and rapidly providing traffic information to traffic control and management systems. Wang et al. [6] proposed a Bayesian combination method by updating the old one introduced by Petridis et al. [7] for short-term traffic flow forecasting by taking three single predictors: autoregressive integrated moving average, a Kalman filter, and a back propagation neural network. Qi and Ishak [8] and Habtemichael and Cetin [9] also designed similar short-term traffic flow systems by applying a hidden Markov model and K-nearest neighbor, respectively.

Sarin et al. [10] developed a forecasting model named JamBayes, for traffic flow and congestion in the Greater Seattle area, aimed at use in different versions on different devices, smartphones, and desktops. The model of this work was constructed by employing a Bayesian network that has powerful approaches for approximating traffic flows.

There are two differences taken into consideration for traffic prediction between the systems discussed above and this paper. The first difference is in utilizing history data. The systems mentioned above made traffic prediction only based on one point of view in using the history from one resource. In other words, the data were collected from only one resource, for example, a loop detector station, a camera, and so on. However, this paper takes into account multiple points of consideration in using real-time history data that might influence heavy traffic jams, such as weather data and time conditions, festivals, etc. The second difference is in traffic prediction style. The systems discussed above predicted the traffic condition in general, and did not go into further detailed estimation. The core difference between the previous systems and our system is that we intend to accurately provide more detailed traffic information to the public. On this account, we predict traffic conditions in three stages, Red, Yellow, and Green, and then make a detailed analysis in these three classes.

This system models traffic flow prediction by combining traffic data from roadside cameras and other traffic influencing factors, such as weather conditions, rush hour times, special days, weekdays, and weekends, aiming at complementing each other perfectly. This paper mainly references S.S.Aung et al. [12] and we implemented the system by using a combination approach.

Table 1. Attribute Table.

No		Attribute Name	Value
1	*Inputs*	*NumberofVehicles*	High / Medium / Low
2		*SpecialDay*	Yes / No
3		*WeatherCondition*	1 / 2 / 3
4		*CurrentTime*	High / Medium / Low
5		*TypeofDay*	Weekday / Weekend
6	*Output*	*Traffic*	Red/Yellow/Green

Fig. 1. Ojana Junction, Okinawa, Japan.

3. Data Collection and Representation

Table 1 illustrates the attributes and their related values. The input attributes are *NumberofVehicles, SpecialDay*, *WeatherCondition*, *CurrentTime* and *TypeofDay*. The output attribute is *Traffic*. *NumberofVehicles* means the total number of vehicles on the road, and it has three values, *High*, *Medium*, and *Low*. Moreover, it also has to get live weather data from a weather station (*WeatherCondition*) with values of 1, 2, and 3. In this case, 1 means sunny, 2, cloudy, and 3, raining. In addition, it analyses the current time as to whether it is rush hour or not (*CurrentTime*). Those values are *High, Medium*, and *Low*. Furthermore, it considers the current day for whether it is a weekend or a weekday (*TypeofDay*). In rush hour, the traffic jam is usually high or medium. The rest of the time, it is low. The traffic situation is defined in one of three conditions: *Red*, *Yellow*, and *Green*. *Red* is the most congested situation, a heavy traffic jam. This condition mostly happens at rush hour and sometimes under bad weather conditions. *Yellow* status means be aware that it can change to *Red* or *Green*, and *Green* means no heavy traffic and smooth conditions.

Traffic data and related information have been mainly collected from Ojana Junction, Okinawa, Japan, for five months from Google Maps as well as a video camera. Fig. 1 is Ojana Junction, Route 58.

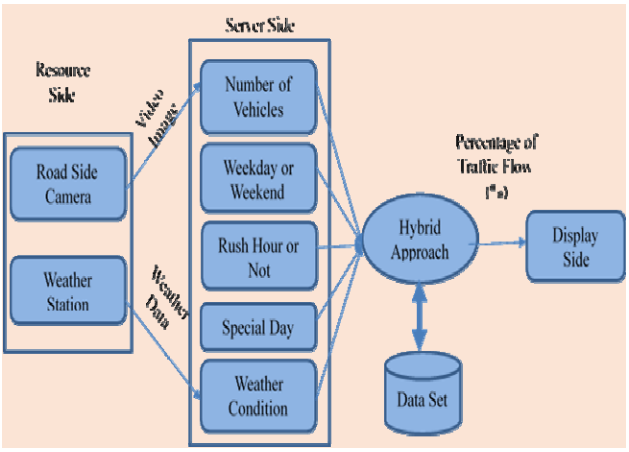

Fig. 2. General Architecture of the Traffic Prediction System.

4. Traffic Prediction Model

This section has three subsections. The first describes the overall system architecture; the second is a detailed explanation of each classifier, and the last is our proposed approach, the hybrid approach.

4.1 Overall System Architecture

Fig. 2 illustrates the overall architecture of the traffic prediction system. The system is composed of three main parts, the Resources Side (RS), the Traffic Prediction Server (TPS), and the Display Side. RS has responsibility for tracking traffic data and then transmitting it to the TPS. The Traffic Prediction Server plays a vital role in the traffic prediction system. Its responsibility is to filter the raw data into usable data, and then to integrate the refined data with a dataset by applying estimation methods. After that, it sends the final estimation result to the Display Side. This paper assumes that it has already received the data from a roadside camera and weather station, and does not emphasize the detailed calculations of image processing.

4.2 Theoretical Description of Each Machine Learning Model

The aim of machine learning is to generate a model that is able to automatically predict future events based on experience and information from past events (history data).

4.2.1 Theoretical Description of Each Machine Learning Model

Decision tree learning is one of the most commonly used and practical approaches for real-world problems in machine learning as a classifier. It constructs a tree by using a top-down search through a set of attributes.

The following paragraphs explain how to apply Decision Tree approach to our traffic system given real traffic data set, with Red, Yellow and Green target concepts. An attribute having the highest information gain,

the entropy *H(S), is computed by* using the Eq. (1). Suppose S is a collection of *N* examples of target concepts, Red, Yellow and Green, the entropy of S relative to this classification is

$$Entropy(S) = -(p_{Red})Log_2(p_{Red}) - (p_{Yellow})Log_2(p_{Yellow}) - (p_{Green})Log_2(p_{Green}) \quad (1)$$

where p_{Red} is the proportion of *Red* examples, p_{Yellow} is the proportion of *Yellow* examples, and p_{Green} is the proportion of *Green* examples in *S*.

Suppose S is a collection of *N* examples of target concepts Red, Yellow, and Green. Then the entropy of *S* relating to this traffic status classification can be calculated by using the following equations:

$$\text{Gain}(S, \text{Current Time}) = Entropy(S) - \sum_{v-1}^{v} \frac{CurrentTime_v}{N} Entropy(CurrentTime_v) \quad (2)$$

where Values (*A*) is the set of all possible values for attribute A, and S_v is the subset of S for which attribute *A* has value *v*.

The information gain for all attributes, CurrentTime, NumbersofCar, SpecialDay, WeatherCondition, Typeof Day, can be derived from Eq. (2) as following:

$$Gain(S, CurrentTime) = Entropy(S) - \sum_{v\in\{High,Medium,Low\}} \frac{|S_v|}{|S|} Entropy(S_v) \quad (3)$$

$$Gain(S, NumbersofCar) = Entropy(S) - \sum_{v\in\{High,Medium,Low\}} \frac{|S_v|}{|S|} Entropy(S_v) \quad (4)$$

$$Gain(S, SpecialDay) = Entropy(S) - \sum_{v\in\{Yes,No\}} \frac{|S_v|}{|S|} Entropy(S_v) \quad (5)$$

$$Gain(S, WeatherCondition) = Entropy(S) - \sum_{v\in\{Sunny,Cloudy,Raining\}} \frac{|S_v|}{|S|} Entropy(S_v) \quad (6)$$

$$Gain(S, TypeofDay) = Entropy(S) - \sum_{v\in\{Weekday,Weekend\}} \frac{|S_v|}{|S|} Entropy(S_v) \quad (7)$$

where *v* is the value of the attributes. Information gain is a measurement to choose the best attribute at each level in the growing tree.

Fig. 3 demonstrates a typical learned decision tree for estimating traffic conditions. In this figure, the attribute, *NumberofVehicles*, which has the highest information gain, is the root for the first level. According to the figure, heavy traffic jams can mostly occur when the number of cars on the road is high. If *NumberofVehicles* is low, traffic jam status is *Green*. Usually, the almost exact result of counting the number of cars using an image processing technique can be examined during good weather conditions. Otherwise, to get an accurate result, the system needs to consider weather condition and time. It is obvious that when *WeatherCondition* is greater than 2 (raining) the traffic condition can be concluded to be heavy from this evidence. If *NumberofVehicles* is at medium, the system needs to consider Type*ofDay*, *SpecialDay*, and *WeatherCondition* in advance.

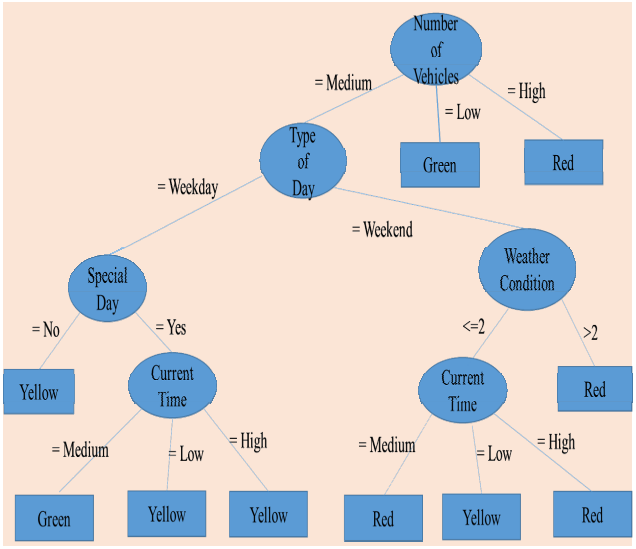

Fig. 3. Decision Tree Representation for the Traffic Prediction System.

4.2.2 Support Vector Machine

A support vector machine is one of the supervised learning algorithms and classifies the observed data by using a set of training examples belonging to categories. For the data classification process, SVM first establishes the model by learning the given training examples, and the second stage is to estimate the observed data by using the model. A novel type of learning machine, the SVM has been receiving increasing attention in areas ranging from its original application in pattern recognition to the extended application of regression estimation [13].

To predict the traffic condition on the target road or place, SVM estimates traffic flow over the training dataset of *m* points in the following form:

$$S = (x_1, a_1), (x_2, a_2), \ldots, (x_m, a_m)$$

where a_i represents traffic status (*Red*, *Yellow* or *Green*), and x_i is traffic attribute values in the following form:

<*NumberofVehicles* = *1*, *SpecialDay* = *0*,
WeatherCondition = *2*, *CurrentTime* = *1*, *TypeofDay* = *1*>

Consider a pattern classifier that uses a hyperplane to separate the classes of patterns based on given examples, *S*. The hyperplane is defined by (ω, *b*), where ω is a weight

vector, and b is bias. The linear function of the hyperplane can be written as

$$f(x) = <\omega.\, x_i> + b = \sum_{i=1}^{m} \omega_i\, x_i + b \tag{8}$$

where ω is a weight vector, x_i maps input x to a vector in a feature space, and b is bias [14].

4.2.3 Multinomial Logistic Regression

Multinomial logistic regression is used for data in which the dependent variable is unordered with more than two possible discrete outcomes [15].

The following paragraphs discuss how to apply multinomial logistic regression to our traffic prediction system.

In this work, to estimate the probability of each traffic state, *Red*, *Yellow*, and *Green*, the following multinomial logistic regressions are applied. Therefore, it has three kinds of outcome that can be defined as dependent variables. Independent variables are a set of N observed data points. Each data point consists of a set of M independent variables (*CurrentTime*, *WeatherCondition*, *NumberofVehicles*, *TypeofDay*). In this system, the *Green* condition is defined as the reference category. Then, a model for Logit voting on the *Red* condition can be computed by using the following equation: a model for log odds voting on the *Red* condition is Eq. (8).

$$\begin{aligned} L^{(Red)} &= \log = \left(\frac{\pi_i^{(Red)}}{\pi_i^{Yellow}} \right) \\ &= \beta_0^{Red} + \beta_1^{Red}\, CurrentTime_i \\ &\quad + \beta_2^{Red}\, WeatherCondition_i \\ &\quad + \beta_3^{Red}\, NumberofVehicle_i + \beta_4^{Red}\, TypeofDay_i \end{aligned} \tag{9}$$

A second model for log odds voting on the **Yellow** condition is as follows:

$$\begin{aligned} L^{(Yellow)} &= \log = \left(\frac{\pi_i^{(Yellow)}}{\pi_i^{Red}} \right) \\ &= \beta_0^{Yellow} + \beta_1^{Yellow}\, CurrentTime_i + \\ &\quad \beta_2^{Yellow}\, WeatherCondition_i + \\ &\quad \beta_3^{Yellow}\, AmountofVehicle_i + \\ &\quad \beta_4^{Yellow}\, TypeofDay_i \end{aligned} \tag{10}$$

An estimation model for the ***Red*** condition is:

$$\pi^{Red} = \frac{e^{\left(L^{(Red)}\right)}}{1 + e^{\left(L^{(Red)}\right)} + e^{\left(L^{(Yellow)}\right)}} \tag{11}$$

An estimation model for the ***Yellow*** condition is:

Algorithm 1: **Hybrid Approach**

```
vj ∈ V = {Red, Yellow, Green} where (j=1,2,3)
Input: current time, kind of day, special day,
   amount of car, weather data
Output: percentage of current traffic condition
   (Red, Yellow and Green)
1) Accept new input instance from resource
   side and calculate current time and current
   day.
2) Read history data from traffic database.
3) Predict traffic flow by using Decision
   Tree, SVM and Multinomial Logistic
   Regression.
4) Predict traffic situation by using Majority
   Rule using equation 14, 15, and 16.
5) If the results of each algorithm are equal,
   the system automatically set SVM's
   prediction as a final result based on the
   past prediction experience.
6) Store the final result in traffic history
   database for future prediction, and go to
   step 1.
```

$$\hat{\pi}^{Yellow} = \frac{e^{\left(L^{(Yellow)}\right)}}{1 + e^{\left(L^{(Red)}\right)} + e^{\left(L^{(Yellow)}\right)}} \tag{12}$$

An estimating model for the ***Green*** condition is:

$$\hat{\pi}^{Green} = \frac{1}{1 + e^{\left(L^{(Red)}\right)} + e^{\left(L^{(Yellow)}\right)}} \tag{13}$$

4.2.4 Proposed Hybrid Approach

Fig. 4 gives a description and flow chart of the proposed approach. The detailed processes of our proposed algorithm are in **Algorithm 1**.

The main task of this system is to learn and predict the level of current traffic condition, *Red*, *Yellow*, *and Green* by applying hybrid approach. According to the study of the algorithms descried above, each has a unique prediction skill which could not be occurred in other methods. By taking this advantage, we integrate the individual skill of each algorithm aiming at giving the best estimation skill.

The following Eqs. (14)-(16), are for calculating the traffic level for Green, Yellow, and Red condition respectively by applying majority rule.

$$Green = \frac{LG_{green} + DT_{green} + SVM_{green}}{n} \tag{14}$$

$$Yellow = \frac{LG_{yellow} + DT_{yellow} + SVM_{yellow}}{n} \tag{15}$$

$$\mathrm{Re}\,d = \frac{LG_{red} + DT_{red} + SVM_{red}}{n} \tag{16}$$

where LG means Multinomial Logistic Regression, DT is Decision Tree and SVM is Support Vector Machine. If the output of LG is green, LG_{green} is one and so on. After that,

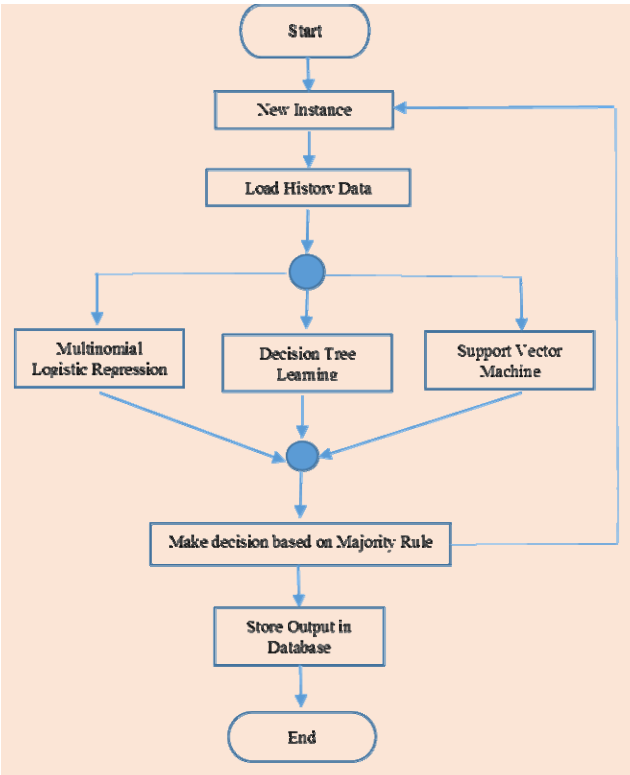

Fig. 4. Flow Diagram of Traffic Prediction System.

choose the class that has the highest value.

Fig. 4 gives the description and flow chart of propose approach. The detail processes of our proposed approach are represented in **Algorithm 1**.

In this work, hypothesis X is a vector of six constraints, specifying the values of the six attributes, ***CurrentTime, TypeofDay, SpecialDay, NumberofVehicles,*** and ***WeatherData***. V is the set of target values, ***Red***, ***Yellow*** and ***Green***.

Thus, the combination approach based on majority rule promotes prediction accuracy by acquiring the specialty of each method. Majority rule is a decision-making system for choosing a vote between two or more options, and the option with 50% of the vote wins. However, there are circumstances in which there can be no winner, and the hybrid approach cannot make a final decision in that case.

In this issue, the system yields a prediction result as the final estimation by taking into account the classifier that has the highest prediction accuracy. For that case, according to our simulation experience, SVM has the highest prediction accuracy from among these three algorithms. Therefore, whenever the system encounters the same situation, it automatically gives SVM first priority for the final estimation result. For example, the result of logistic regression might be ***Green***, but from decision tree, ***Yellow***, and from SVM, ***Red***. Under this condition, there is no winner according to majority rule, and the system automatically assumes the traffic state is ***Red*** because SVM has the highest priority, based on past experience.

5. Experimental Results

This section will discuss and compare the performance of the three estimation methods against the combination approach based on real traffic data. Now, the system is focusing on the Route 58 Ojana Junction in Okinawa, Japan. Before going to a prediction of each section, let us measure how much error is included in the dataset. As the system very much depends on history data, the accuracy of the hypothesis is also very important in order to get better future results. In this work, the accuracy of the hypothesis is measured by using standard deviation for *errors(h)*, where *errors(h)* means errors of hypothesis *h* with respect to target function *f* and data sample *S*:

$$errors(h) = \frac{r}{n} \tag{17}$$

where r means the number of instances from S misclassified by h, and n means the number of instances in sample S.

$$\delta_{errors(h)} \approx \sqrt{\frac{errors(h)(1-errors(h))}{n}} \tag{18}$$

In this case, error for the hypothesis was observed to be 0.03 (3%). Now let us explore each individual performance for prediction accuracy.

5.1 Experimental Result using Decision Tree Learning

In estimating traffic jams by utilizing three estimators (decision tree, SVM, and logistic regression), SVM has the highest accuracy among these methods. The experiment was carried out by applying Eqs. (1)-(6).

The *Y* axis gives the number of points that were correctly predicted. The *X* axis expresses the time in minutes. To measure how much it can accomplish, the ground truth data related to this junction have already been gathered. The prediction result is divided into three parts: *Best*, *Medium*, and *Worst*. *Best* is when the predictor can predict a traffic jam, with the highest performance defined between numbers 15 and 20. *Medium* is expressed with numbers from 8 to 15. The last is the *Worst* case, with numbers from 0 to 7. In other words, if the predictor can only predict the result correctly somewhere between 0 and 7, the accuracy is defined as *Worst*. There are 60 minutes in an hour, but measurements were made every 20 minutes. The blue circle shows the level of the prediction result. In the first hour, the prediction result reaches *Medium* once. After that, the estimator's result remains stable in the best condition until more than 3 hours. Then it goes to *Medium* three times. After 8 hours, it hit *Worst* once and then went up to the *Best* level until the last time. It can be clearly seen that there are only four points at *Medium* and one point at the *Worst* accuracy over nine hours.

Fig. 5 gives the correctness prediction points for the decision tree. As illustrated by the graph, three-fourths of

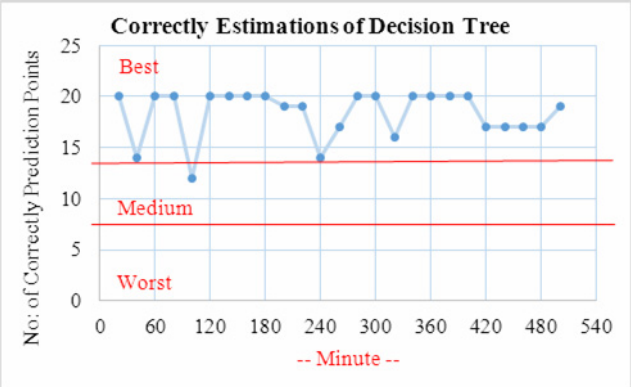

Fig. 5. Correct Estimations of the Decision Tree Approach over Nine Hours.

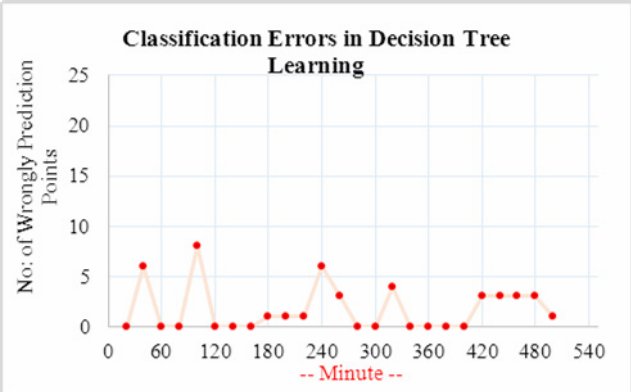

Fig. 6. Wrong Estimations by Decision Tree Learning over Nine Hours.

Table 2. Prediction Accuracy of Decision Tree Learning.

Class	Precision	Recall	F-measurement
Red	90%	90%	90%
Yellow	94%	82%	88%
Green	87%	98%	94%

the total are at the *Best* level. However, the approximation points fluctuate according to Fig. 5, whereas in Fig. 14, for the combination approach, the prediction points remain stable until the last time. In the decision tree approach, only three estimation points are at *Medium*. But this approach does not reach *Worst* status over this time duration.

Fig. 6 depicts the number of incorrect predictions with the decision tree approach. During the first two hours, it makes many estimation mistakes. The most incorrect predictions is 8, the second highest is 6, and the third highest is 4. After two hours, it dramatically falls until, over the next hour, it estimated 12 points correctly. However, its estimations did not fall to *Worst*. According to the figure, decision tree learning made more mistakes than logistic regression.

Table 2 describes the estimation accuracy of decision tree learning. It can be clearly seen that Red could be predicted better than Yellow and Green. According to the

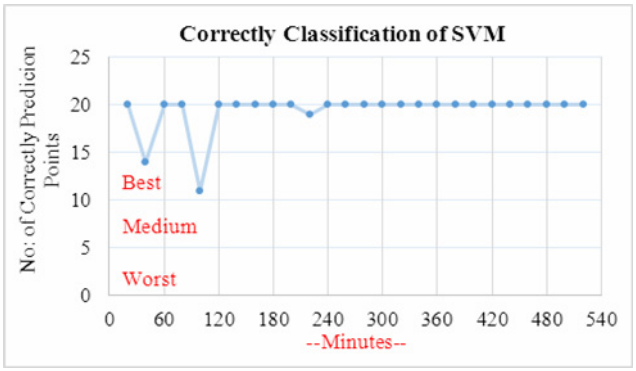

Fig. 7. Correct Estimations by SVM over Nine Hours.

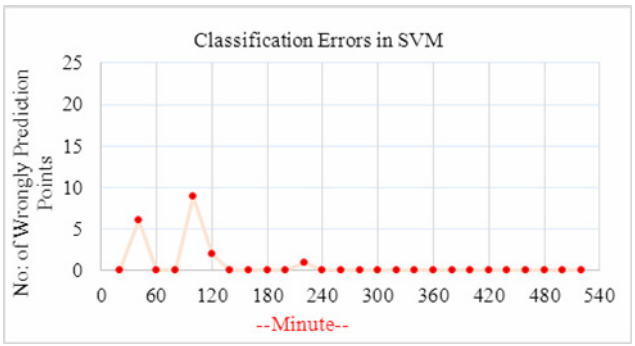

Fig. 8. Wrong Estimations by the SVM Classifier over Nine Hours.

table, the accuracy for predicting Green status is better than for Yellow. For Green status, the accuracy percentage is over 90% in Recall and F-measurement columns. For Yellow, the accuracy is over 90% only in Precision. Recall and F-measurement results are slightly lower than Precision. The overall accuracy of the decision tree approach is 91%. The accuracy of this approach is a little better than multinomial logistic regression.

5.2 Experimental Results using Support Vector Machine

The support vector machine approach is a highly practical learning method and a good classifier for many real-world problems. Therefore, the performance with SVM shows that its prediction accuracy is higher than logistic regression and decision tree learning in this work.

Fig. 7 illustrates the correctness of traffic estimation over nine hours. According to the figure, most points could be predicted well. However, it reached ***Medium*** accuracy only two times, and there was no point at the ***Worst*** level. According to the figure, SVM is better than logistic regression and decision tree at traffic jam forecasting.

Fig. 8 illustrates the prediction points that were incorrectly estimated. This classifier could forecast most points exactly the same as the ground truth data. The worst situation can be seen in the first and second columns, from 40 minutes to 120 minutes.

Table 3 shows traffic estimation accuracy of the SVM classifier computed by Precision, Recall, and F-measurement. The estimation accuracy for Green status is

Table 3. Prediction Accuracy of the SVM Approach.

Class	Precision	Recall	F-measurement
Red	93%	100%	96%
Yellow	97%	93%	95%
Green	99%	98%	98%

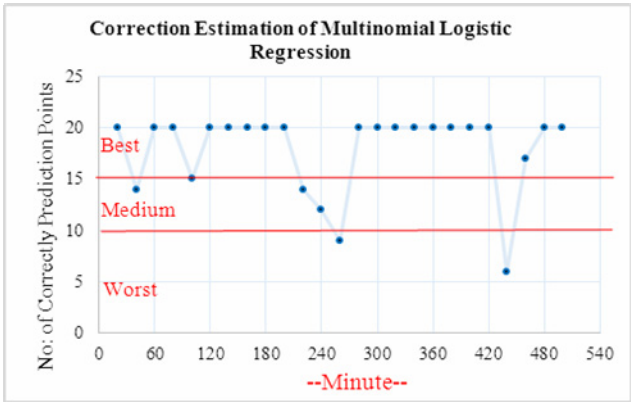

Fig. 9. Correct Estimations of the Multinomial Logistic Regression Approach over Nine Hours.

98% and 99% in all measurements. It is obvious that Green is the highest accuracy among these classes. The second is Red status, and Yellow is at the lowest accuracy level. In this case, Red and Green conditions rarely change, but the Yellow condition is still in fluctuation. Therefore, prediction accuracy is slightly lower than for the other conditions.

5.3 Experimental Result using Multinomial Logistic Regression

The logistic regression approach unexpectedly did better at traffic jam estimation than decision tree by utilizing Eqs. (8)-(12). Fig. 9 demonstrates how many points at which the estimator could correctly predict traffic jams over nine hours. According to these results, the logistic regression approach is the second best estimator among these three approaches.

Fig. 9 displays how many points were predicted correctly, whereas Fig. 10 shows how many mistakes the predictor makes every 20 minutes over about nine hours. From the graph, it is clear that the largest number of mistakes is less than 14. The second highest number of mistakes is 11, and the third highest is less than 8. As shown by the graph, it made only seven mistakes over nine hours.

Table 4 describes the prediction accuracy for the different classes over nine hours. It can be clearly seen that the predictor could estimate Red, Yellow, and Green well, according to the measurement results. Among them, the accuracy of Red status remains stable at over 92%. The accuracy of Red status is 100% computed by Recall, whereas it is 93% and 96% for Precision and F-measurement, respectively. In the case of Yellow status, the accuracy measured for Precision is 99%, whereas it is 78% and 87% as measured by Recall and F-measurement, respectively. The accuracy for Yellow status is the lowest

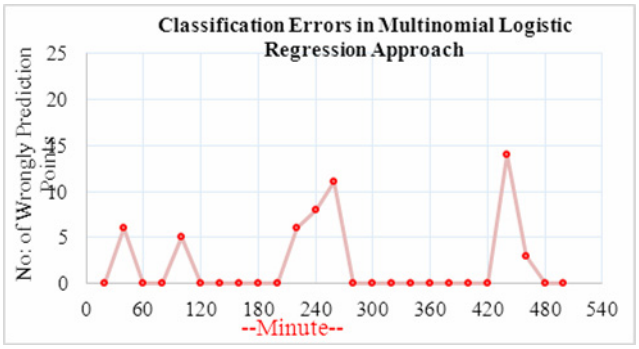

Fig. 10. Incorrect Estimations by the Multinomial Logistic Regression Approach over Nine Hours.

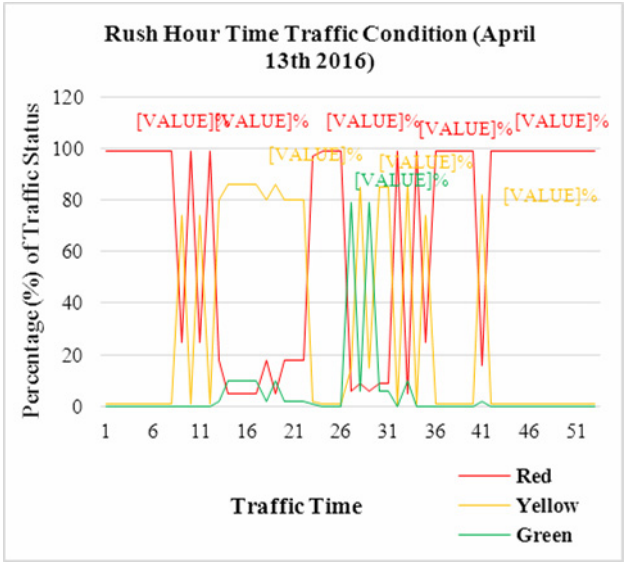

Fig. 11. Decision Tree Traffic Prediction for Rush Hour.

Number 1 to 51 represent traffic time, where number 1, 6, and 11 denote morning rush hour time from 6:30 AM to 8:30 AM, number 16, 21, 26, and 31 are for normal time, number 36, 41,46, and 51 symbolize evening rush hour time from 4 PM to 8:30 PM.

Table 4. Prediction Accuracy of Multinomial Logistic Regression Approach.

Class	Precision	Recall	F-measurement
Red	93%	100%	96%
Yellow	99%	78%	87%
Green	74%	98%	84%

among these methods in Recall. The overall accuracy of multinomial logistic regression is 90%.

Figs. 11-13 visually show the prediction results for decision tree, SVM, and multinomial logistic regression classifiers for rush hour. According these figures, the traffic jams remain stable at the Red condition during rush hour at Ojana Junction, Okinawa, Japan.

For the rest of the time, the traffic state is mostly at Yellow and Green. According to these figures, Red is a heavy traffic condition, Yellow is a situation where drivers must be aware, and Green is a light traffic condition. It can be divided into three parts: morning rush hour from 6:30

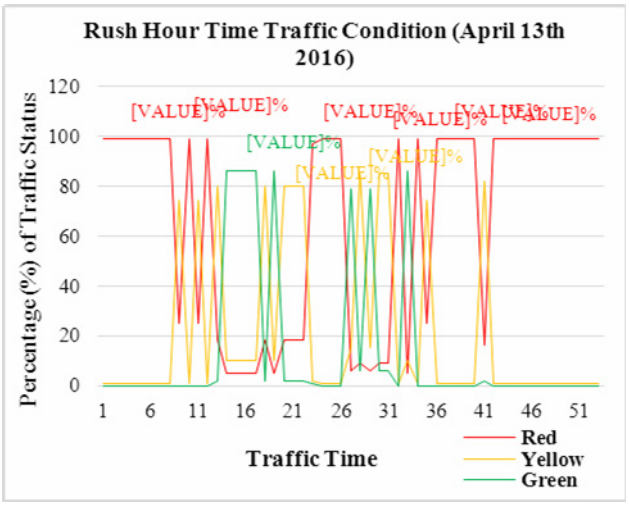

Fig. 12. SVM Traffic Prediction for Rush Hour.

Number 1 to 51 represent traffic time, where number 1, 6, and 11 denote morning rush hour time from 6:30 AM to 8:30 AM, number 16, 21, 26, and 31 are for normal time, number 36, 41,46, and 51 symbolize evening rush hour time from 4 PM to 8:30 PM.

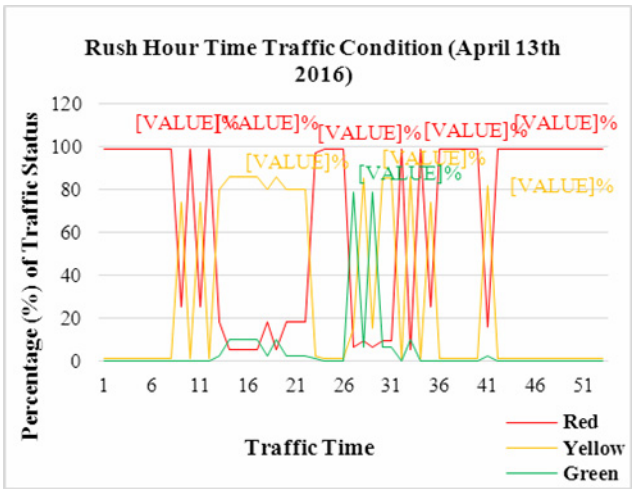

Fig. 13. Multinomial Traffic Prediction for Rush Hour.

Number 1 to 51 represent traffic time, where number 1, 6, and 11 denote morning rush hour time from 6:30 AM to 8:30 AM, number 16, 21, 26, and 31 are for normal time, number 36, 41,46, and 51 symbolize evening rush hour time from 4 PM to 8:30 PM.

AM to 8:30 AM, normal time from 8:30 AM to 4:00 PM and evening rush hour from 4:00 PM to 8:30 PM. In morning rush hour, most traffic was at Red status for over 90% of the prediction results. At the end of the period, it fluctuated just before normal time. In normal time, it is obvious that Red status was seen only once. Most times were at Yellow and Green. Evening rush hour is the same situation as morning rush hour. The traffic condition remained at Red until 8:30 PM.

During normal time, the traffic condition is mostly at Yellow and Green, while the heavy traffic condition remains stable in rush hour. According to Table 2, Tables 3 and 4, the accuracy for Yellow status is less than

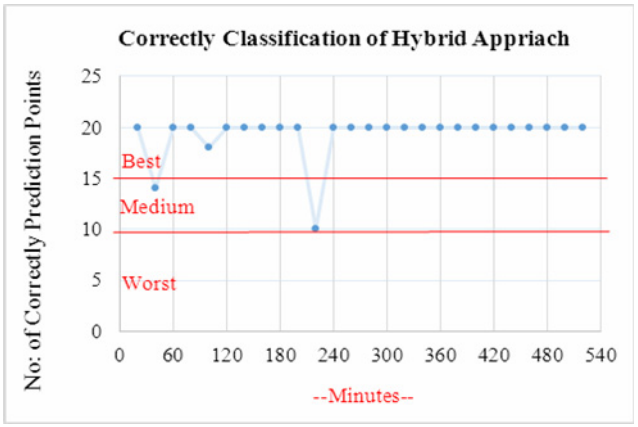

Fig. 14. Correctly Estimation of Hybrid Approach over 9 hours.

that for Red and Green The Yellow condition is between Red and Green. It means that if traffic is at Red, it hardly ever changes to Green immediately, and also, Green hardly ever changes to Red directly. However, it can easily change from Yellow to Red or Green most of the time. For this reason, the accuracy of Yellow is slightly less than for Red and Green.

5.4 Experimental Result using Hybrid Approach

Section 5.1 explained the analytical performance of decision trees, Section 5.2, SVM, and Section 5.3, the multinomial logistic regression approach. This section is going to describe the analytical performance of the combination approach and compare it with the decision tree, SVM, and logistic regression approaches.

The system predicts traffic conditions by applying **Algorithm 1** described in Section 4.3, and the prediction accuracy is measured by Precision, Recall, and F-measurement.

Fig. 14 shows the number of correct prediction points from the hybrid approach. According to the figure, most prediction points remain stable in the ***Best*** range. It means that the system could predict the traffic conditions accurately. It is obvious that only two points are in the ***Medium*** range. The graph clearly shows that the accuracy of the combination approach did not reach ***Worst***. Compared with the accuracy of the three approaches described above, it can be seen that the hybrid approach could predict more accurately with the best performance. Fig. 15 also describes the number of wrong predictions with the hybrid approach. The largest number of mistakes is only 8, but no more than that.

Compared with other approaches, the combination approach makes fewer mistakes than the SVM approach, multinomial logistic regression, and decision tree, and also promotes prediction accuracy, as shown in Table 5. For ***Yellow*** status, Recall accuracy is 93%, and the accuracy of the other two methods is over 99%. According to the analytical results of the three approaches (decision tree, SVM, and multinomial logistical regression), these estimators had a little bit of difficulty forecasting the

Table 5. Prediction Accuracy of Combination Approach.

Class	Precision	Recall	F-measurement
Red	93%	100%	96%
Yellow	99%	93%	96%
Green	98%	99%	98%

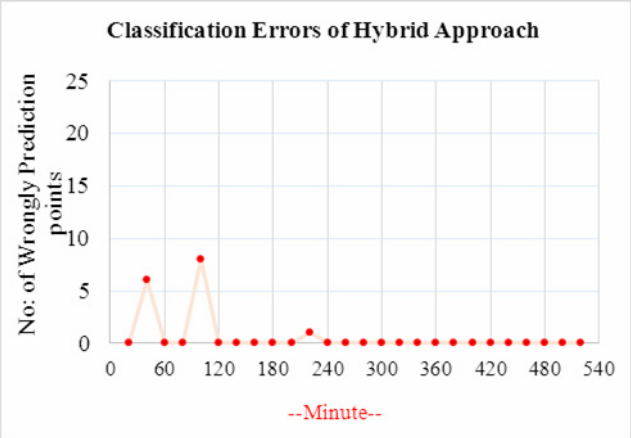

Fig. 15. Wrongly Estimation of Combination Method over 9 hours.

Yellow condition because it is easy for it to change to another state. However, Green and Red are mostly stable.

As described above, Yellow status is the most difficult situation for accuracy because it can change to another state frequently. On this issue, the hybrid approach did much better than the other three approaches because it integrates proficiencies from the other approaches, especially for Yellow status. Consequently, it also upgrades the whole forecasting accuracy of the hybrid approach unexpectedly. It is obvious that the very changeable condition, Yellow, decreases the prediction accuracy of the decision tree, SVM, and logistic regression classifiers. However, it can be clearly seen that the combination method achieved the highest accuracy for the prediction of Yellow status by combining good points of each method. Therefore, it is a big help in improving overall prediction accuracy.

In detail, Recall values for Yellow status under logistic regression and decision tree are 78% and 82%, respectively, while the combination approach predicted it with 93% accuracy. In F-measurement, the combination approach reached the highest prediction accuracy of 96% among these methods. In Precision, the exactness of the combination approach is much higher than decision tree and SVM.

Table 6 compares the prediction accuracy of each traffic status of hybrid approach with Decision Tree, SVM, and hybrid approach. It is obvious that hybrid approach is the best predictor with the average accuracy 97% among these approaches.

6. Conclusion

This paper describes a traffic prediction system that was implemented at Ojana Junction, Route 58, Okinawa, Japan. Some factors, such as weather conditions, data from cameras, and analyzing day and time are considered in approximating traffic conditions. According to the analytical results, the combination of three methods is the best estimator, with 97% accuracy; the second best estimator is SVM with 96% accuracy, the third is the multinomial logistic regression approach with 92% estimation accuracy, and the last is decision tree with 91% accuracy. Although these three methods could predict traffic conditions with acceptable accuracy, they failed in achieving good results with the very changeable *Yellow* status. Compared with the combination method, it could carry out this difficulty with the highest accuracy (97%) by exploiting majority rule. For future work, we are going to explore and measure the traffic state of road links adjacent to Ojana Junction, which indirectly impact its heavy traffic jams.

Table 6. Comparing Prediction Accuracy.

Traffic Status	Decision Tree	SVM	Multinomial Logistic Regression	Hybrid Approach
Green	90%	96%	96%	96%
Yellow	88%	95%	88%	96%
Red	93%	98%	85%	98%
Average Accuracy	90%	96%	89%	97%

References

[1] D. osenbaum, J. Leitloff, F. Kurz, O. Meynberg, and T. Reize, “Real-Time Image Processing for Road Traffic Data Extraction from Aerial Images”, Vienna, Austria, July 5-7, 2010, IAPRS, Vol.XXXVIII, Part 7B.

[2] P.Niksaz, “Automatic Traffic Estimation Using Image Processing”, International Journal of Signal Processing, Image Processing and Pattern Recognition, vol.5, No. 4, December, 2012.

[3] S.M. Hashemi, M.Almasi, R.Ebrazi, M.Jahanshahi, “Predicting the Next State of Traffic by Data Mining Classification Techniques”, International Journal of Smart Electronical Engineering. Vol.1,No.3,2012, pp.181:193.

[4] X.Lu, T.Izumi, L.Teng and L.Wang, “Particle Filter Vehicle Tracking Based on SURF on SURF Feature Matching”, IEEJ Journal of Industry Applications, Vol.3 No.2 pp. 182-191, May 15, 2013.

[5] D.Gao, J.Zhou, and L.Xin, “SVM-based Detection of Moving Vehicles for Automatic Traffic Monitoring”, IEEE Intelligent Transportation Systems Conference Proceedings-Oakland (CA) , Page(s):745 – 749, August 25-29, 2001.

[6] J.Wang, W.Deng, and Y.Guo, Nextrans Center, Purdue University, West Lafayette, IN 47906, USA, "New Bayesian combination method for short-term traffic flow forecasting", Transportation Research Part C 43 (2014) 79-94.

[7] V.Petridis et al., "A Bayesian Models Combination Method for Time Series Prediction ", Journal of Intelligent and Robotic Systems, Volume 31, Issue 1, pp 69-89, May 2001.
[8] Y.Qi and S.Ishak, "A Hidden Markov Model for short term prediction of traffic conditions of freeways", Transportation Research Part C 43 (2014) 95-111.
[9] F.G. Habtemichael and M.Cetin, "Short-term traffic flow rate forecasting based on identifying similar traffic patterns", Transportation Research Part C 66 (2016) 61-78.
[10] R. Sarin, E. Horvittz, J. Apacible and L. Liao., "Prediction, expectation, and surprise: Methods, designs, and study of a deployed traffic forecasting service", In Twenty-First Conference on Uncertainty in Artificial Intelligence, UAI-2005, UAI-P-2005-PG-275-283.
[11] E. Bolshinsky and R. Freidman, "Traffic Flow Forecast Survey", Technion-Computer Science Department- Technical Report CS-2012-06-2012, June 3, 2012.
[12] S.S. A, S.Tamaki, I.Nagayama, "Advanced Traffic Prediction System by Socio-Technical Sensor Fusion using Machine Learning", International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSSC-2016), pp. 709 – 712, July 13, 2016.
[13] L.J. Cao and F.E.H. Tay, Department of Mechanical Engineering, the national University of Singapore, "Support Vector Machine with Adaptive Parameters in Financial Time Series Forecasting", IEEE Transactions on Neural Networks, Vol.14, No.6, November 2003.
[14] M.C.Neto, Y.S.Jeong, M.K.Jeong, and L.D.Han,Department of Civil and Environmental Engineering, University of Tennessee, USA, "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions", Expert System with Applications Journal, Vol 36, Issue 3, Pages 6164-6173, April 2009.
[15] Y.Wang, Center for Outcomes Research and Evaluation, Yale University and Yale New Haven Health System, USA, "A multinomial logistic regression modeling approach for anomaly intrusion detection", Computer & Security Journal, Volume 24, Issue 8, Pages 662-674, 13 May 2005.

Swe Swe Aung (Non-member) received a Bachelor of Computer Science in 2004 from Computer University, Loikaw, Myanmar, and a Master of Computer Science in 2009 from the Computer University, Taunggyi, Myanmar. She is currently working toward a PhD at University of the Ryukyus, Okinawa, Japan. Her research interests are in transportation technologies, image processing, mobile computing, and cloud computing.

Itaru Nagayama (Member) received a BSc, MSc and PhD in Information Science and Systems Engineering from the University of Tokushima. He is now an associate professor in the Department of Information Engineering in the Faculty of Engineering at University of the Ryukyus in Okinawa, Japan. His research interests include digital image processing, machine intelligence, evolutionary engineering, and social safety systems. He published some handbooks for intelligent information systems and its applications. He published some handbooks for intelligent information systems and their application.

Shiro Tamaki (Member) was born in Okinawa, Japan. He received an M.Eng. from Tokushima University. Then, he received a PhD in Engineering Science from Osaka University. Currently, he is a professor in the College of Engineering, University of the Ryukyus. His research interests include Natural Energy applications and IoT for agricultural systems, application of control theory to horticultural facilities, and development of agricultural robotics. He is a member of IEEE and IPSJ.