PAPER

Travel Time Prediction System Based on Data Clustering for Waste Collection Vehicles

Chi-Hua CHEN[†a)], *Member*, Feng-Jang HWANG[††], *and* Hsu-Yang KUNG[†††], *Nonmembers*

SUMMARY In recent years, intelligent transportation system (ITS) techniques have been widely exploited to enhance the quality of public services. As one of the worldwide leaders in recycling, Taiwan adopts the waste collection and disposal policy named "trash doesn't touch the ground", which requires the public to deliver garbage directly to the collection points for awaiting garbage collection. This study develops a travel time prediction system based on data clustering for providing real-time information on the arrival time of waste collection vehicle (WCV). The developed system consists of mobile devices (MDs), on-board units (OBUs), a fleet management server (FMS), and a data analysis server (DAS). A travel time prediction model utilizing the adaptive-based clustering technique coupled with a data feature selection procedure is devised and embedded in the DAS. While receiving inquiries from users' MDs and relevant data from WCVs' OBUs through the FMS, the DAS performs the devised model to yield the predicted arrival time of WCV. Our experiment result demonstrates that the proposed prediction model achieves an accuracy rate of 75.0% and outperforms the reference linear regression method and neural network technique, the accuracy rates of which are 14.7% and 27.6%, respectively. The developed system is effective as well as efficient and has gone online.

key words: *travel time prediction, arrival time prediction, intelligent transportation system, waste collection vehicle, data clustering*

1. Introduction

In recent years, intelligent transportation system (ITS) techniques have been widely exploited to enhance the quality of public services, e.g. mass rapid transit [1], railway traffic [2], bus system [3], bicycle-sharing system [4], and garbage truck fleet management [5], etc. One of these applications which have not been extensively studied could be the utilization of transportation information in waste collection service. As one of the worldwide leaders in recycling, Taiwan adopts the waste collection and disposal policy named "trash doesn't touch the ground", which requires the public to deliver garbage directly to the specific locations for awaiting garbage collection during specific time periods. Under this effective waste collection and disposal system, the waste collection vehicle (WCV) collects garbage from the collection points (CPs) along the route that is defined by a specific sequence of CPs. Denote by CP$_n$ the n-th collection point in a considered waste collection route. Define the time when a WCV arrived at CP$_n$ by t_n. Consider the route segment from CP$_{n-1}$ to CP$_{n+1}$ as shown in Fig. 1. The realized travel time of WCV from CP$_{n-1}$ to CP$_n$ is defined by $x_{n,n+1} = t_{n+1} - t_n$. When the WCV arrived at CP$_n$ at time t_n, the predicted travel time of WCV from CP$_n$ to CP$_{n+1}$, which is denoted by $x'_{n,n+1}$, can be generated by the travel time prediction methods [1], [2], [6]–[11]. Then the predicted arrival time of WCV at CP$_{n+1}$, i.e. $t'_{n+1} = t_n + x'_{n,n+1}$, can be obtained. Then the predicted arrival time t'_{n+1} can be announced to the public through the mobile application so that the public can avoid suffering a long waiting for garbage disposal or missing the garbage collection.

Several approaches to travel or arrival time prediction have been proposed, e.g. statistical mean value (SMV) method, linear regression (LR) method, neural networks (NNs), etc [1], [2], [6]–[11]. Although the statistical methods such as SMV and LR can provide the predicted information quickly, the accuracy of predicted information would be lowered when a large variation exists in historical records. The NN and deep learning techniques can provide relatively precise prediction, but a relatively high computational cost is required. To develop an effective and efficient WCV arrival time prediction system, this study devises a travel time prediction model utilizing the adaptive-based clustering technique coupled with a data feature selection procedure. Our utilized clustering method can effectively merge high-similarity data clusters to extract and analyze the characteristics of traffic information and further enhance the accuracy of the devised prediction model. The developed travel time prediction system consists of mobile devices (MDs), on-board units (OBUs), a fleet management server (FMS), and a data analysis server (DAS). While receiving inquiries from users' MDs and relevant data from WCVs' OBUs through the FMS, the DAS performs the devised travel time prediction model to yield the predicted arrival time of WCV, which is then sent to users' MDs via the FMS.

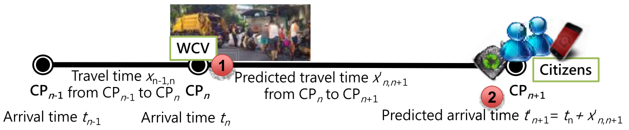
WCV
1
Citizens
Travel time $x_{n-1,n}$
CP$_{n-1}$ from CP$_{n-1}$ to CP$_n$ CP$_n$
Predicted travel time $x'_{n,n+1}$
from CP$_n$ to CP$_{n+1}$
2
CP$_{n+1}$
Arrival time t_{n-1}
Arrival time t_n
Predicted arrival time $t'_{n+1} = t_n + x'_{n,n+1}$

Fig. 1 The WCV route segment from CP$_{n-1}$ to CP$_{n+1}$.

Manuscript received August 25, 2018.
Manuscript revised January 15, 2019.
Manuscript publicized March 29, 2019.

[†] The author is with the College of Mathematics and Computer Science, Fuzhou University, China.

[††] The author is with the School of Mathematical and Physical Sciences, Transport Research Centre, University of Technology Sydney, Australia.

[†††] The author is with the Department of Management Information Systems, National Pingtung University of Science and Technology, Pingtung, Taiwan, R.O.C.

a) E-mail: chihua0826@gmail.com

DOI: 10.1587/transinf.2018EDP7299

The remainder of the paper is as follows. Various exiting travel time prediction methods are reviewed in Sect. 2. The devised travel time prediction model utilizing the adaptive-based clustering technique coupled with a data feature selection procedure is introduced in Sect. 3. The experimental results and online implementation are presented in Sect. 4. Finally, the conclusions are given in Sect. 5.

2. Literature Reviews

This section discusses the existing approaches for travel time prediction in literature and the relevant machine learning techniques for data clustering. The SMV, LR, NN, recurrent neural network (RNN) and auto-encoder (AE) methods are demonstrated in Sect. 2.1. The partition-based clustering (PBC), density-based clustering (DBC), grid-based clustering (GBC), and adaptive-based clustering (ABC) methods are introduced in Sect. 2.2.

2.1 Existing Travel Time Prediction Methods

The SMV, LR, NN, RNN and AE methods proposed to estimate and predict the travel time [1], [2], [6]–[10] are discussed and compared as follows.

2.1.1 Statistical Mean Value (SMV) Method

The SMV method for travel time prediction [6] contains the following two steps.

Step 1: This method generates the predicted travel time $x'_{n,n+1}$ based on the mean value of historical travel times from CP_n to CP_{n+1} within a specific recent time period. Denote the realized travel time from CP_n to CP_{n+1} in the k-th record by $x_{n,n+1,k}$ for $k = 1, 2, \ldots, m$. Then the predicted travel time $x'_{n,n+1}$ can be estimated by Eq. (1). To demonstrate the SMV method, we consider a real-world example from Hsinchu City WCV route, where $n = 2$ and $m = 10$ as showed in Fig. 2. The predicted travel time $x'_{2,3}$ is about 364 seconds calculated by Eq. (1).

$$x'_{n,n+1} = \overline{x}_{n,n+1} = \frac{\sum_{k=1}^{m} x_{n.n+1,k}}{m} \tag{1}$$

Step 2: The predicted arrival time at CP_3, i.e. $t'_3 = t_2 + x'_{2,3}$, can then be obtained.

To evaluate the performance of the SMV method, the mean absolute error (MAE), i.e. the absolute value of the difference between the predicted and realized values, is used. In this Hsinchu WCV example, the MAE of the predicted travel time $x'_{2,3}$ by the SMV method is equal to 218 seconds. Since the SMV method only employs the mean value of historical records to predict travel time, the significant inaccuracy of estimation would be incurred due to the dynamic traffic conditions.

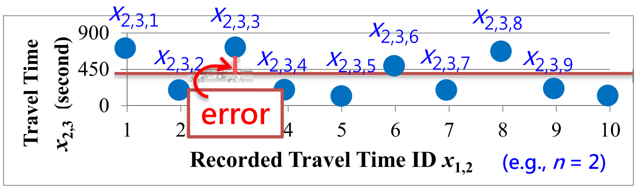

Fig. 2 Ten historical travel times from CP_2 to CP_3 and the SMV.

2.1.2 Linear Regression (LR) Method

The application of LR method in analyzing real-time traffic conditions for travel time prediction can be found in previous studies [1], [6], [11]. In the setting of LR method, the realized travel time $x_{n-1,n}$ is used as the input for estimating the predicted travel time $x'_{n,n+1}$, i.e. the output. The LR procedure consists of the following two steps.

Step 1: An LR model can be defined by Eq. (2), where the intercept parameter b and the slope parameter w can be calculated by Eqs. (3) and (4). Consider again the Hsinchu City WCV example with $n = 2$ and $m = 10$ as shown in Fig. 3. The LR equation (5) can be obtained using Eqs. (2)–(4).

$$x'_{n,n+1} = b + w \times x_{n-1,n} \tag{2}$$

$$w = \frac{\sum_{k=1}^{m} (x_{n-1,n,k} - \overline{x}_{n-1,n})(x_{n,n+1,k} - \overline{x}_{n,n+1})}{\sum_{k=1}^{m} (x_{n-1,n,k} - \overline{x}_{n-1,n})^2} \tag{3}$$

$$b = \overline{x}_{n,n+1} - w \times \overline{x}_{n-1,n} \tag{4}$$

$$x'_{2,3} = -77.242 + 1.3895 x_{1,2} \tag{5}$$

Step 2: Since the realized travel time $x_{1,2}$ is collected as the input of the LR equation, the predicted travel time $x'_{2,3}$ can be estimated by Eq. (5). Then the predicted arrival time $t'_3 = t_2 + x'_{2,3}$ can then be obtained.

The MAE of the predicted travel time $x'_{2,3}$ by the LR equation (5) is equal to 150 seconds. Although the accuracy of LR model is better than the SMV method, it could not be effectively applied to the practical traffic conditions with the property of non-linearity.

2.1.3 Neural Networks (NNs)

The NN techniques were proposed to analyze the interaction effects among the input parameters and produce the non-linear prediction models [2], [6], [10]. A NN consists of an input layer, hidden layers (e.g., dense layers, convolutional layers, or recurrent layers), and an output layer. The

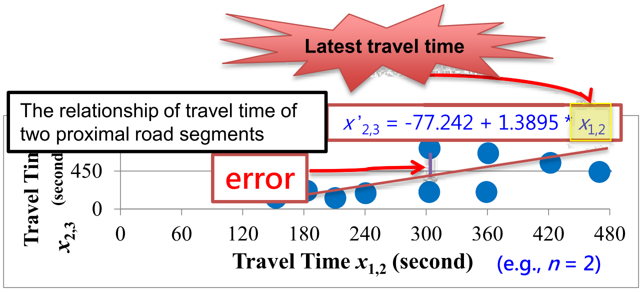

Fig. 3 Ten pairs of historical travel times and the LR equation.

sigmoid function can be applied as the activation function for obtaining a non-linear model, and the gradient descent method can be applied in the optimization of each weight in the NN [12], [13]. A NN model consists of the following two stages.

Stage 1 (Training stage): The structure of a NN is defined and the NN model is trained at this stage. The realized travel time $x_{n-1,n}$ is used as the input layer of the NN, and the predicted travel time $x'_{n,n+1}$ is set as the output layer. The dense layers can be defined as hidden layers for the analyses of the interaction effects among parameters. Then the historical records can be used to train the NN, and the optimized value of each weight in the NN model can be calculated by the gradient descent method.

Stage 2 (Runtime stage): At the runtime stage, the trained NN model can be exploited to predict the travel time $x'_{n,n+1}$ in accordance with the input parameter $x_{n-1,n}$. Then the predicted arrival time $t'_{n+1} = t_n + x'_{n,n+1}$ can be obtained.

Although the NN model can yield non-linear solutions, it does not take into account different data features, which may influence the accuracy of WCV travel time prediction according to our preliminary study.

2.1.4 Recurrent Neural Network (RNN)

In recent years, the RNNs and long short term memory (LSTM) networks have been proposed to analyse the time series datasets. The features of the past sequence elements can be encoded as several neurons (i.e. vectors) by recurrent layers in RNNs and LSTM networks. The records of traffic flow and travel time can be expressed as time series data and sequential vectors which can be adopted into RNNs and LSTM networks for extracting the significant features of sequence elements and improving the accuracies of traffic information [14]–[18]. However, the higher computation time and cost are required by RNNs and LSTM networks. Furthermore, the prediction results of RNNs may be the same as the prediction results of NNs if the length of sequential input vectors is short.

2.1.5 Auto-Encoder (AE)

For data generalization and over-fitting prevention, the AE method has been proposed to reduce dimensions and extract the significant features. For performing AE method, a multilayer NN model can be constructed with a hidden layer which has lower dimensions, and the input vectors are the same as the output vectors in the NN model. The input vectors can be encoded as significant features from the input layer to the hidden layer, and the significant features can be decoded as the output vectors from the hidden layer to the output layer accordance with trained NN model [18]–[20]. Therefore, the significant features of traffic information may be extracted and encoded by AE method for the improvement of prediction accuracies. However, performing AE method needs higher computation time and cost.

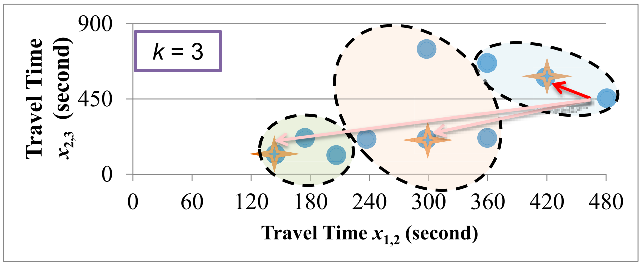

Fig. 4 PBC with $m = 10$ and $k = 3$ for the travel time prediction.

2.2 Data Clustering Methods

Data clustering techniques are the important tool to extract and analyze the characteristics of traffic information. The clustering methods can be categorized into four classes, i.e. partition-based clustering (PBC) [21], [22], density-based clustering (DBC) [23], [24], grid-based clustering (GBC) [25], [26], and adaptive-based clustering (ABC) [27], [28].

2.2.1 Partition-Based Clustering (PBC)

The PBC methods (e.g., the k-means clustering) can cluster data into several groups in accordance with the similarities or distances between the data and cluster centers [21], [22]. The four steps of the PBC are shown as follows.

Step 1: Given m data points, we determine the number of clusters denoted by k. Consider the example with $m = 10$ and $k = 3$ as shown in Fig. 4. Each realized travel time record is depicted as a dot, and each cluster center is marked with a star.

Step 2: Among the m data points, k points are randomly selected as the centers of the k individual clusters.

Step 3: The similarity or distance between each of the m data points and each of the k cluster centers is calculated.

Step 4: Each data point is grouped into a cluster in accordance with the highest similarity between the data point and cluster center. The center of each cluster is then updated after grouping. Steps (3) and (4) will be repeated until each cluster center remains unchanged.

Although the data records can be clustered with the relatively low time complexity $O(mk)$ by the PBC, the outliers cannot be filtered out. Furthermore, it could be difficult to determine the value of k, which influences the clustering quality.

2.2.2 Density-Based Clustering (DBC)

The DBC methods (e.g., the density-based spatial clustering of applications with noise) can analyze the density of each considered data group to determine the clusters [23], [24]. The DBC consists of the following three steps.

Step 1: A radius r and the minimum number of points denoted by q are predefined for clustering. Consider an example with $r = 60$ and $q = 2$. As shown in Fig. 5, each travel

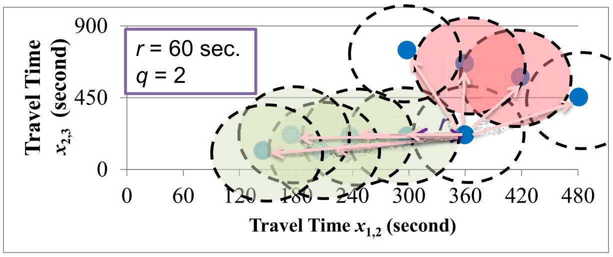

Fig. 5 DBC with m = 10, r = 60, and q = 2 for the travel time prediction.

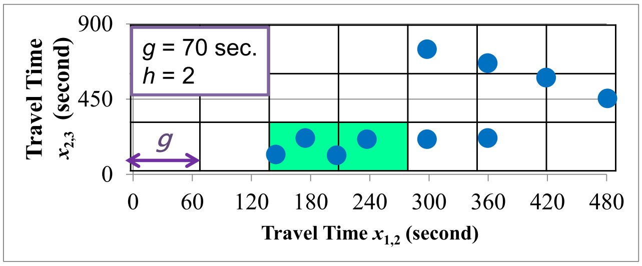

Fig. 6 GBC with m = 10, g = 70, and h = 2 for the travel time prediction.

time record is depicted as a dot and the circle of each data point has a radius r = 60 seconds.

Step 2: The similarity or distance between each pair of the m records is calculated.

Step 3: If the number of data points in a circle is larger than q, the circle is marked as a significant circle. Any adjacent significant circles are grouped into a cluster.

Although outliers can be filtered out by the DBC, its time complexity $O(m^2)$ is relatively high. Furthermore, it again could be difficult to determine the values of q and r, which influence the clustering quality.

2.2.3 Grid-Based Clustering (GBC)

The GBC methods utilize grids to filter out the outliers and to reduce the time complexity [25], [26]. The GBC contains the following four steps.

Step 1: The grid length g and the minimum number of points h are predetermined for clustering. Consider an example with g = 70 and h = 2. As shown in Fig. 6, each travel time record is depicted as a dot.

Step 2: Denote by o the number of grids. All the m points can be classified into the o girds according to the grid length g.

Step 3: If the number of points in a grid is larger than h, the grid is marked as a significant grid. Any adjacent significant grids are then grouped into a cluster.

Although the time complexity of the GBC is $O(mo)$, the values of g and h would influence the clustering quality and determining their values is an optimization problem.

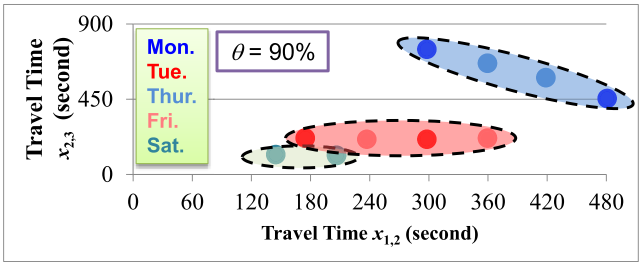

Fig. 7 ABC with m = 10, v = 6, and θ = 90% for the travel time prediction.

2.2.4 Adaptive-Based Clustering (ABC)

The ABC methods analyze the similarities or distances between the data and cluster centers and group these data into several clusters in accordance with a threshold [27], [28]. The ABC consists of the following four steps.

Step 1: The data can be initially grouped into v clusters according to the date features. Consider an example with v = 6 using the data feature of weekdays (including Saturday) as shown in Fig. 7.

Step 2: A similarity threshold θ is selected. Consider θ = 90% in our example.

Step 3: The similarity between each pair of clusters can be calculated using a similarity function.

Step 4: If the similarity between some pair of clusters is higher than θ, they are grouped into a new cluster. Steps 3 and 4 are repeated until all the clusters remain unchanged.

Since our preliminary study indicates that some data features, e.g. weekdays and humidity, may influence the travel time prediction, the ABC would be a suitable technique for yielding an accurate travel time prediction model.

3. The Proposed Travel Time Prediction System

This study develops a travel time prediction system based on data clustering for providing real-time information on the arrival time of waste collection vehicle (WCV). The components of the proposed system are presented in Sect 3.1, and the proposed WCV travel time prediction method is illustrated in Sect. 3.2.

3.1 Components of the Proposed System

The architecture of the proposed system includes MDs, OBUs, an FMS, and a DAS as shown in Fig. 8. Each component is introduced as follows.

3.1.1 Mobile Devices (MDs)

The mobile application of the WCV arrival time information needs to be installed in the public's MDs. The users can send inquiries about the current location of WCV and the predicted WCV arrival time at each CP via the mobile application. The FMS can immediately reply with the relevant

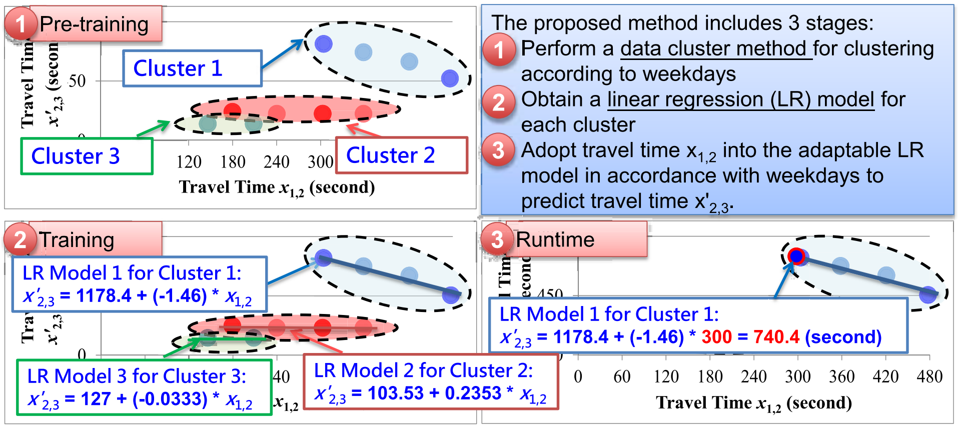
1 Pre-training
Travel Time x'2,3 (second)
50
Cluster 1
Cluster 3
Cluster 2
120 180 240 300
Travel Time x1,2 (second)
The proposed method includes 3 stages:
1 Perform a data cluster method for clustering according to weekdays
2 Obtain a linear regression (LR) model for each cluster
3 Adopt travel time x1,2 into the adaptable LR model in accordance with weekdays to predict travel time x'2,3.
2 Training
LR Model 1 for Cluster 1:
x'2,3 = 1178.4 + (-1.46) * x1,2
LR Model 3 for Cluster 3:
x'2,3 = 127 + (-0.0333) * x1,2
LR Model 2 for Cluster 2:
x'2,3 = 103.53 + 0.2353 * x1,2
3 Runtime
450
LR Model 1 for Cluster 1:
x'2,3 = 1178.4 + (-1.46) * 300 = 740.4 (second)
0 60 120 180 240 300 360 420 480
Travel Time x1,2 (second)

Fig. 9 The three stages of the proposed travel time prediction model.

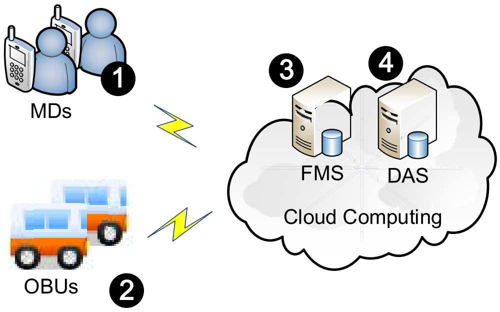

Fig. 8 The architecture of the proposed travel time prediction system.

traffic information of WCV to MDs.

3.1.2 On-Board Units (OBUs)

The OBUs including the global positioning system (GPS) and network modules are equipped into the WCV for detecting and reporting the WCV location. The location information is then sent to the FMS via cellular networks [29] or vehicular ad hoc networks (VANETs) [30], [31].

3.1.3 Fleet Management Server (FMS)

The FMS can receive and show the location information of each WCV in a geographic information system. Furthermore, the FMS can detect the arrival events of WCVs at CPs. The realized arrival times are sent to the DAS for the travel time prediction. Then the FMS receives the predicted WCV arrival times from the DAS and sends the information to the users' MDs.

3.1.4 Data Analysis Server (DAS)

Receiving the realized arrival times of WCVs at CPs, the DAS performs the proposed travel time prediction model to estimate the WCV arrival times at the next CPs. The predicted arrival times are then sent to the FMS for broadcasting.

3.2 The Proposed Travel Time Prediction Model

For improvement of travel time prediction, this study uses clustering method to evaluate and select significant features. Furthermore, a linear regression method is considered to be implemented and obtain predicted travel time for lower computation time and cost. The proposed travel time prediction model contains the three stages: (1) the pre-training stage, (2) training stage, and (3) runtime stage, as shown in Fig. 9.

3.2.1 Pre-Training Stage

The pre-training stage performs the ABC method for clustering the historical travel time data. Two data features including weekdays and humidity are considered for feature extraction in this study. The revised chi-squared distribution is utilized as the similarity function. The chi-squared distribution can be used to test the significant difference between two data groups. The cumulative distribution function of the chi-squared distribution given in Eq. (6) can then estimate the probability of difference [32]. We note that the chi-squared value between the i-th cluster and the j-th cluster is defined as $v_{i,j}$. The degree of freedom between the i-th cluster and the j-th cluster is denoted by $d_{i,j}$. Furthermore, the function $\gamma\left(\frac{d_{i,j}}{2}, \frac{v_{i,j}}{2}\right)$ is the lower incomplete gamma function, and the function $\Gamma\left(\frac{d_{i,j}}{2}\right)$ is the ordinary gamma function. Therefore, the similarity $s_{i,j}$ between the i-th cluster and the j-th cluster can be estimated by Eq. (7). Moreover, the value of the similarity threshold θ is set as 90%. The analyses of data features are presented in the following subsections.

$$F(v_{i,j}, d_{i,j}) = \frac{\gamma\left(\frac{d_{i,j}}{2}, \frac{v_{i,j}}{2}\right)}{\Gamma\left(\frac{d_{i,j}}{2}\right)} = \frac{\int_{b=0}^{\frac{v_{i,j}}{2}} b^{\frac{d_{i,j}}{2}-1} e^{-b}\, db}{\int_{b=0}^{\infty} b^{\frac{d_{i,j}}{2}-1} e^{-b}\, db} \tag{6}$$

$$s_{i,j} = 1 - F(v_{i,j}, d_{i,j}) \tag{7}$$

3.2.1.1 Data Feature of Weekdays

As aforementioned, we consider a real-world case with Hsinchu City WCVs, where the garbage collection days include Monday, Tuesday, Thursday, Friday, and Saturday. Thus, the historical travel time records can be initially grouped into five clusters. The similarity between each pair of clusters can be estimated by Eq. (7). The results of the similarities in the first run are shown in Table 1. As the value of $s_{2,4}$ is higher than the similarity threshold $\theta = 90\%$ and is the highest similarity, the cluster 2 and cluster 4 can be merged into a new cluster. The similarity results updated in the second run are shown in Table 2. The value of $s_{1,3}$ which is higher than $\theta = 90\%$, so the cluster 1 and 3 can be merged into a new cluster. Then the third run is performed to update the similarity as shown in Table 3, where no similarity is higher than 90%. Therefore, the historical WCV travel time records can be grouped into three clusters, i.e. (i) Monday and Thursday, (ii) Tuesday and Friday, and (iii) Saturday.

Our clustering result is exactly consistent with the practical insights. If the public's behaviors are different in different weekdays, and the travel times in different weekdays could be differentiated. In our real-life instance of Hsinchu City WCVs, Sunday and Wednesday are the days without garbage collection service. It implies that relatively more trash would be accumulated for disposal on Monday and Thursday. It is then expected that a longer WCV travel time is required on Monday as well as Thursday, which perfectly explains the reason why the data of Monday and Thursday are grouped into Cluster 1 as shown in Fig. 9. Furthermore, most residents may choose to travel on Saturday, so less trash would be collected by WCVs, which again clarifies why the data of Saturday individually form Cluster 3 as shown in Fig. 9. The remaining data of Tuesday and Friday are grouped into Cluster 2.

Table 1 The similarity results for weekdays in the first run.

j \ i	1 (Monday)	2 (Tuesday)	3 (Thursday)	4 (Friday)	5 (Saturday)
1 (Monday)	-	-	-	-	-
2 (Tuesday)	$s_{1,2}$=0.00%	-	-	-	-
3 (Thursday)	$s_{1,3}$=92.84%	$s_{2,3}$=0.00%	-	-	-
4 (Friday)	$s_{1,4}$=0.00%	$s_{2,4}$=99.47%	$s_{3,4}$=0.00%	-	-
5 (Saturday)	$s_{1,5}$=0.00%	$s_{2,5}$=87.35%	$s_{3,5}$=0.00%	$s_{4,5}$=0.00%	-

Table 2 The similarity results for weekdays in the second run.

j \ i	1 (Monday)	2 (Tuesday & Friday)	3 (Thursday)	4 (Saturday)
1 (Monday)	-	-	-	-
2 (Tuesday & Friday)	$s_{1,2}$ = 0.00%	-	-	-
3 (Thursday)	$s_{1,3}$ = 92.84%	$s_{2,3}$ = 0.00%	-	-
4 (Saturday)	$s_{1,4}$ = 0.00%	$s_{2,4}$ = 86.31%	$s_{3,4}$ = 0.00%	-

3.2.1.2 Data Feature of Humidity

We then investigate whether the humidity is an appropriate data feature for clustering the WCV travel time records. The humidity can be classified into four levels, i.e. Level 1 (0%–25%, sunny day), Level 2 (25%–50%), Level 3 (50%–75%), and Level 4 (75%–100%, raining day). Therefore, the historical travel time records can be initially grouped into four clusters. The similarity between each pair of clusters can be estimated by Eq. (7), and the similarity results are shown in Table 4. Since there is no similarity higher than 90%, the humidity cannot be used to perform the data clustering for the WCV travel time prediction.

3.2.2 Training Stage

In order to develop our model with the low computational complexity, we adopt the LR method to train the prediction model for each cluster yielded in the pre-training stage. Then the historical travel time records in each cluster are used to generate the linear regression equation according to Eqs. (2), (3), and (4). Given the data in Fig. 9, the linear regression equations for Clusters 1, 2, 3 are yielded as Eqs. (8), (9), and (10), respectively.

$$\text{Cluster 1: } x'_{2,3} = 1178.4 - 1.46x_{1,2} \tag{8}$$

$$\text{Cluster 2: } x'_{2,3} = 103.53 + 0.2353x_{1,2} \tag{9}$$

Table 3 The similarity results for weekdays in the third run.

j \ i	1 (Monday & Thursday)	2 (Tuesday & Friday)	3 (Saturday)
1 (Monday & Thursday)	-	-	-
2 (Tuesday & Friday)	$s_{1,2}$ = 0.00%	-	-
3 (Saturday)	$s_{1,2}$ = 0.00%	$s_{2,3}$ = 86.31%	-

Table 4 The similarity results for humidity in the first run.

j \ I	1 (Level 1)	2 (Level 2)	3 (Level 3)	4 (Level 4)
1 (Level 1)	-	-	-	-
2 (Level 2)	$s_{1,2}$ = 0.00%	-	-	-
3 (Level 3)	$s_{1,3}$ = 0.00%	$s_{2,3}$ = 0.00%	-	-
4 (Level 4)	$s_{1,4}$ = 0.00%	$s_{2,4}$ = 0.00%	$s_{3,4}$ = 0.00%	-

$$\text{Cluster 3: } x'_{2,3} = 127 - 0.0333x_{1,2} \tag{10}$$

The travel time prediction models between each two consecutive CPs are trained in the training stage. For instance, the travel time prediction model from the n-th CP to the $(n+1)$-th CP is expressed as $x'_{n,n+1}$ (shown in Eq. (11)) in accordance with historical records.

$$x'_{n.n+1} = b_{n,n+1} + w_{n,n+1} \times x_{n-1,n} \tag{11}$$

3.2.3 Runtime Stage

At the runtime stage, the yielded linear regression model can be selected according to the weekdays and thus clusters. Then the realized travel time $x_{1,2}$ can be brought into the selected linear regression model to predict the travel time $x'_{2,3}$. For instance, if on Monday the DAS receives the information of realized travel time $x_{1,2} = 300$ seconds, it generates the predicted travel time $x_{2,3} = 740.4$ seconds as shown in Fig. 9. Furthermore, the predicted travel time $x'_{n,n+1}$ can be adopted into Eq. (12) to predict the travel time $x'_{n+1,n+2}$ from the $(n+1)$-th CP to the $(n+2)$-th CP when the WCV arrives at the n-th CP.

$$x'_{n+1,n+2} = b_{n+1,n+2} + w_{n+1,n+2} \times x'_{n,n+1} \tag{12}$$

4. Computational Experiment and Online Implementation

The experimental results and discussions are provided in Sect. 4.1 for evaluating the proposed predicted model. The online implementation of the developed system is demonstrated in Sect. 4.2.

4.1 Computational Experiment

This study considers LR, NN, RNN and AE for travel time prediction. This section discusses the comparisons of accuracy and computation time in 8 cases (i.e., Case 1: LR without clustering; Case 2: NN without clustering; Case 3: RNN without clustering; Case 4: using AE in pre-training stage and using NN in training stage without clustering; Case 5: LR with clustering; Case 6: NN with clustering; Case 7: RNN with clustering; Case 8: using AE in pre-training stage and using NN in training stage with clustering). Section 4.1.1 presents the accuracies of each case, and Sect. 4.1.2 shows the computation time of each case.

4.1.1 Accuracy

In our computational experiments, the records of the Hsinchu City WCV travel times from April to October were collected to evaluate the performance of the proposed prediction model. To justify the accuracy of travel time prediction, we adopted the accuracy formula devised by the Ministry of Transportation and Communications of Taiwan (MTOC) as shown in Eq. (13). If the error ratio of a predicted travel time record is larger than 20%, it is labelled as error. Otherwise, the record is labelled as precise. This accuracy formula is used to calculate the ratio of precise records.

$$\text{Accuracy} = 100\% - \frac{\sum_{k=1}^{m} f\left(\frac{x_{n,n+1,k} - x'_{n,n+1,k}}{x_{n,n+1,k}}\right)}{m},$$
$$\text{where } f(z) = \begin{cases} 1, & \text{if } |z| > 20\% \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

The accuracy results of the LR, NN, RNN and AE are shown in Tables 5 and 6. The accuracies in Cases 1, 2, 3 and 4 are 14.7%, 27.6%, 27.6% and 75%, respectively. This study considered one timestamp for RNN, so the accuracy of RNN was the same as the accuracy of NN. For the evaluation of the proposed method, the accuracy results of different methods with clustering are showed in Table 6. The accuracies in Cases 5, 6, 7 and 8 are 75.00%, 76.72%, 76.72% and 77.59%, respectively. Therefore, the accuracy of travel time prediction can be improved based on clustering method. Figures 10 or 11 show that the distances between actual data and the predicted equation are relatively large, which is reflected by the low accuracy value. Figure 12 illustrates that

Table 5 The accuracy comparisons of different methods without clustering.

	LR	NN	RNN	AE+NN
Accuracy	14.70%	27.60%	27.60%	37.93%

Table 6 The accuracy comparisons of different methods with clustering.

	Cluster + LR (the proposed model)	Cluster+NN	Cluster+RNN	Cluster+AE+NN
Accuracy	75.00%	76.72%	76.72%	77.59%

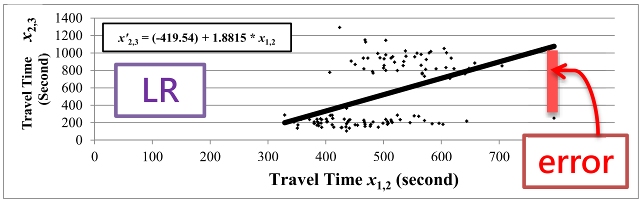

Fig. 10 The travel time prediction equation by the LR method.

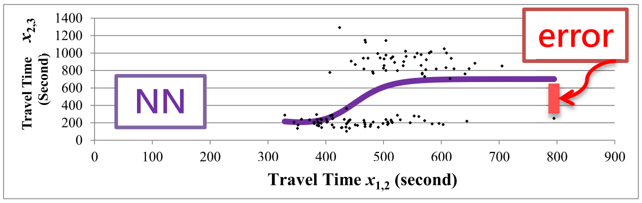

Fig. 11 The travel time prediction equation by the NN.

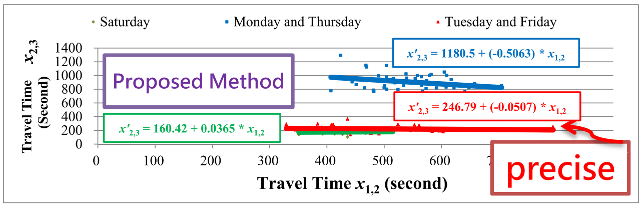

Fig. 12 The travel time prediction equation by the proposed model.

Table 7 The computation time comparisons of different methods without clustering. (unit: milliseconds)

	LR	NN	RNN	AE+NN
Time	15.01	161.48	185.31	323.21

Table 8 The computation time comparisons of different methods with clustering. (unit: milliseconds)

	Cluster + LR (the proposed model)	Cluster+NN	Cluster+RNN	Cluster+AE+NN
Time	18.92	165.39	189.22	327.12

the distances between actual data and the predicted equations are relatively small, which results in a high accuracy of 75%. Our computational experiment demonstrates that the proposed clustering method can effectively extract the critical data feature for data clustering. Furthermore, our proposed WCV travel time prediction model can fulfil the accuracy of 70% required by MTOC in practice. Therefore, the solutions of travel time prediction with clustering method can be accepted by MTOC.

4.1.2 Computation Time

For the evaluation of computation time, this study used a server with a GPU module (i.e., GeForce GTX 1080). The regression and deep learning models based on TensorFlow and Keras libraries were implemented to obtain predicted travel time.

In the runtime stage, the number of runs is 20,000 for each case, and the means of computation time are presented in Tables 7 and 8. The practical experimental results show that the lower computation time was needed for LR. Furthermore, the higher computation time is required in the cases of NN and RNN. The computation times in Cases 1, 2, 3 and 4 are 15.01 milliseconds, 161.48 milliseconds, 185.31 milliseconds, and 323.21 milliseconds, respectively. For the evaluation of the proposed method, the computation time comparisons of different methods with clustering are showed in Table 8. The computation times in Cases 5, 6, 7 and 8 are 18.92 milliseconds, 165.39 milliseconds, 189.22 milliseconds, and 327.12 milliseconds, respectively. The lower computation time is about 4 milliseconds for clustering method in the runtime stage.

Although the accuracy in Case 8 is higher than the accuracy in Case 5, the computation time in Case 8 is higher than the computation time in Case 5. Furthermore, the accuracy in Case 5 which is higher than the accuracy of 70% can be accepted by MTOC. Therefore, the linear regression method with clustering can be adopted for travel time prediction with lower computation time and cost.

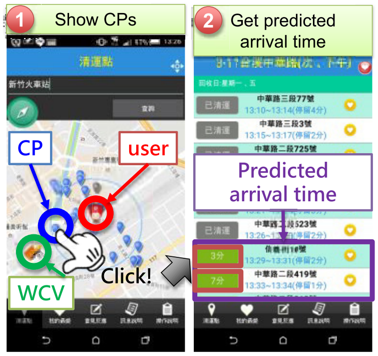

Fig. 13 The user interfaces of the WCV mobile application.

4.2 Online Implementation

The proposed travel time prediction system has gone online for providing the arrival time information of WCVs in Hsinchu City, where there are 157,000 households and 434,000 residents. The developed WCV mobile application which was published in the site of Google Play has been downloaded over 15,000 times [33].

The usage of the WCV mobile application includes the following two steps (as shown in Fig. 13).

Step 1: The mobile application can get the locations of proximal CPs according to user's location.

Step 2: The user can click the icon of CP, and the mobile application can show the predicted WCV arrival time at each CP in the selected route.

In Fig. 13, the red circle indicates the user's location and the blue circle illustrates the location of CP. The green circle indicates the current location of WCV. The user can query the locations of proximal CPs, and then select a suitable CP for obtaining the predicted WCV arrival time. When the user clicks the icon of CP, and the mobile application shows the predicated WCV arrival time at the CP.

5. Conclusions and Future Work

This study developed a travel time prediction system based on data clustering for providing real-time information on the arrival time of waste collection vehicle (WCV). The developed system consists of MDs, OBUs, an FMS, and a DAS. A travel time prediction model utilizing the adaptive-based clustering technique coupled with a data feature selection procedure is devised and embedded in the DAS. Our experiment result demonstrated that the proposed prediction model achieves an accuracy rate of 75.0% and outperforms the reference linear regression method and neural network technique, the accuracy rates of which are 14.7% and

27.6%, respectively. The developed system is effective as well as efficient and has gone online.

For further extension of this research, the NN model or deep learning techniques instead of the LR method can be utilized in the training stage of the devised prediction model for further enhancing the prediction accuracy. Moreover, our prediction model can be applied to different transportation services such as bus and logistics systems.

Acknowledgments

This research was funded by Fuzhou University, grant number 510730/XRC-18075. This research was also funded by the Ministry of Science and Technology, ROC, grant number MOST 107-2637-E-020-007. The authors thank the Telecommunication Laboratories of Chunghwa Telecom Company in Taiwan for the kind support.

References

[1] S.-L. Chen, S.-C. Hsu, C.-T. Tseng, K.-H. Yan, H.-Y. Chou, and T.-M. Too, Analysis of rail potential and stray current for Taipei Metro, IEEE Trans. Veh. Technol., vol.55, no.1, pp.67–75, 2006. DOI: 10.1109/TVT.2005.861164

[2] F. Corman and L. Meng, A review of online dynamic models and algorithms for railway traffic management, IEEE Trans. Intell. Transp. Syst., vol.16, no.3, pp.1274–1284, 2015. DOI: 10.1109/TITS.2014.2358392

[3] C.-H. Chen, An arrival time prediction method for bus system, IEEE Internet of Things Journal, vol.5, no.5, pp.4231–4232, 2018. DOI: 10.1109/JIOT.2018.2863555

[4] A. Schlote, B. Chen, and R. Shorten, On closed-loop bicycle availability prediction, IEEE Trans. Intell. Transp. Syst., vol.16, no.3, pp.1449–1455, 2015. DOI: 10.1109/TITS.2014.2365492

[5] C.-H. Chen, Y.-T. Yang, C.-S. Chang, C.-M. Hsieh, T.-S. Kuan, and K.-R. Lo, The design and implementation of a garbage truck fleet management system, South African Journal of Industrial Engineering, vol.27, no.1, pp.32–46, 2016. DOI: 10.7166/27-1-982

[6] M.G. Karlaftis and E.I. Vlahogianni, Statistical methods versus neural networks in transportation research: Differences, similarities and some insights, Transportation Research Part C: Emerging Technologies, vol.19, no.3, pp.387–399, 2011. DOI: 10.1016/j.trc.2010.10.004

[7] A. O'Sullivan, F.C. Pereira, J. Zhao, and H.N. Koutsopoulos, Uncertainty in bus arrival time predictions: Treating heteroscedasticity with a metamodel approach, IEEE Trans. Intell. Transp. Syst., vol.17, no.11, pp.3286–3296, 2016. DOI: 10.1109/TITS.2016.2547184

[8] H. Yu, Z. Wu, D. Chen, and X. Ma, Probabilistic prediction of bus headway using relevance vector machine regression, IEEE Trans. Intell. Transp. Syst., vol.18, no.7, pp.1772–1781, 2017. DOI: 10.1109/TITS.2016.2620483

[9] L. Moreira-Matias, J. Mendes-Moreira, J.F. de Sousa, and J. Gama, Improving mass transit operations by using AVL-based systems: A survey, IEEE Trans. Intell. Transp. Syst., vol.16, no.4, pp.1636–1653, 2015. DOI: 10.1109/TITS.2014.2376772

[10] L. Zhu, F.R. Yu, Y. Wang, B. Ning, and T. Tang, Big data analytics in intelligent transportation systems: A survey, IEEE Trans. Intell. Transp. Syst., vol.20, no.1, pp.383–398, 2019. DOI: 10.1109/TITS.2018.2815678

[11] W.-J. Chen, C.-H. Chen, B.-Y. Lin, and C.-C. Lo, A traffic information prediction system based on global position system-equipped probe car reporting, Advanced Science Letters, vol.16, no,1, pp.117–124, 2012. DOI: 10.1166/asl.2012.2192

[12] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, Learning representations by back-propagating errors, Nature, vol.323, pp.533–536, 1986. DOI: 10.1038/323533a0

[13] Y. LeCun, Y. Bengio, and G.E. Hinton, Deep learning, Nature, vol.521, pp.436–444, 2015. DOI: 10.1038/nature14539

[14] J. Pang, J. Huang, Y. Du, H. Yu, Q. Huang, and B. Yin, Learning to predict bus arrival time from heterogeneous measurements via recurrent neural network, IEEE Trans. Intell. Transp. Syst., 2018. early access, DOI: 10.1109/TITS.2018.2873747

[15] L. Wu, C.-H. Chen, and Q. Zhang, A mobile positioning method based on deep learning techniques, Electronics, vol.8, no.1, Article ID 59, 2019. DOI: 10.3390/electronics8010059

[16] J. Mackenzie, J.F. Roddick, and R. Zito, An evaluation of HTM and LSTM for short-term arterial traffic flow prediction, IEEE Trans. Intell. Transp. Syst., 2018, early access. DOI: 10.1109/TITS.2018.2843349

[17] P. He, G. Jiang, S.-K. Lam, and D. Tang, Travel-time prediction of bus journey with multiple bus trips, IEEE Trans. Intell. Transp. Syst., 2018, early access. DOI: 10.1109/TITS.2018.2883342

[18] P. Duan, G. Mao, W. Liang, and D. Zhang, A unified spatio-temporal model for short-term traffic flow prediction. IEEE Trans. Intell. Transp. Syst., 2018, early access. DOI: 10.1109/TITS.2018.2873137

[19] C.H. Chen, Reducing the dimensionality of time-series data with deep learning techniques, Science, 2018, eLetter. Available online: http://science.sciencemag.org/content/313/5786/504/tab-e-letters (accessed on 12 Jan. 2019).

[20] L. Zhu, F.R. Yu, Y. Wang, B. Ning, and T. Tang, Big data analytics in intelligent transportation systems: A survey, IEEE Trans. Intell. Transp. Syst., vol.20, no.1, pp.383–398, 2019. DOI: 10.1109/TITS.2018.2815678

[21] Z.H. Huang, N. Li, K.F. Rao, C.T. Liu, Y. Huang, M. Ma, and Z.J. Wang, Development of a data-processing method based on Bayesian *k*-means clustering to discriminate aneugens and clastogens in a high-content micronucleus assay, Human & Experimental Toxicology, vol.37, no.3, pp.285–294, 2017. DOI: 10.1177/0960327117695635

[22] S. Khanmohammadi, N. Adibeig, and S. Shanehbandy, An improved overlapping k-means clustering method for medical applications, Expert Systems with Applications, vol.67, pp.12–18, 2017. DOI: 10.1016/j.eswa.2016.09.025

[23] L. Ni, C. Li, X. Wang, H. Jiang, and J. Yu, DP-MCDBSCAN: Differential privacy preserving multi-core DBSCAN clustering for network user data, IEEE Access, vol.6, pp.21053–21063, 2018. DOI: 10.1109/ACCESS.2018.2824798

[24] D. Ienco and G. Bordogna, Fuzzy extensions of the DBScan clustering algorithm, Soft Computing, vol.22, no.5, pp.1719–1730, 2018. DOI: 10.1007/s00500-016-2435-0

[25] B. Wu and B.M. Wilamowski, A fast density and grid based clustering method for data with arbitrary shapes and noise, IEEE Trans. Ind. Informat., vol.13, no.4, pp.1620–1628, 2017. DOI: 10.1109/TII.2016.2628747

[26] C. Deng, J. Song, R. Sun, S. Cai, and Y. Shi, GRIDEN: An effective grid-based and density-based spatial clustering algorithm to support parallel computing, Pattern Recognition Letters, vol.109, pp.81–88, 2018. DOI: 10.1016/j.patrec.2017.11.011

[27] F. Ferstl, M. Kanzler, M. Rautenhaus, and R. Westermann, Time-hierarchical clustering and visualization of weather forecast ensembles, IEEE Trans. Vis. Comput. Graphics, vol.23, no.1, pp.831–840, 2017. DOI: 10.1109/TVCG.2016.2598868

[28] A.-A. Liu, Y.-T. Su, W.-Z. Nie, and M. Kankanhalli, Hierarchical clustering multi-task learning for joint human action grouping and recognition, IEEE Trans. Pattern Anal. Mach. Intell., vol.39, no.1, pp.102–114, 2017. DOI: 10.1109/TPAMI.2016.2537337

[29] C.-H. Chen, J.-H. Lin, T.-S. Kuan, and K.-R. Lo, A high-efficiency method of mobile positioning based on commercial vehicle operation data, ISPRS International Journal of Geo-Information, vol.5,

no.6, Article ID 82, 2016. DOI: 10.3390/ijgi5060082

[30] R. Huang, J. Wu, C. Long, Y. Zhu, B. Li, and Y.-B. Lin, SPRCA: Distributed multisource information propagation in multichannel VANETs, IEEE Trans. Veh. Technol., vol.66, no.12, pp.11306–11316, 2017. DOI: 10.1109/TVT.2017.2726159

[31] D. Su and S. Ahn, In-vehicle sensor-assisted platoon formation by utilizing vehicular communications, International Journal of Distributed Sensor Networks, vol.13, no.7, pp.1–12, 2017. DOI: 10.1177/1550147717718756

[32] H. Cui, G, Yan, and H. Song, A novel curvelet thresholding denoising method based on chi-squared distribution, Signal, Image and Video Processing, vol.9, no.2, pp.491–498, 2015. DOI: 10.1007/s11760-014-0685-9

[33] Chunghwa Telecom Co. Ltd., Waste collection vehicle mobile application, Google Play 2018. Available online: https://play.google.com/store/apps/details?id=com.cht.cleanbuspublic&hl=zh_TW (accessed on 12 Jan. 2019).

Chi-Hua Chen serves as a professor (a Minjian scholar and a Qishan scholar) for the College of Mathematics and Computer Science at Fuzhou University in China. He received his Ph.D. degree from the Department of Information Management and Finance of National Chiao Tung University (NCTU) in 2013. He has published over 200 journal articles, conference articles, and patents. His recent research interests are in the Internet of things, big data, deep learning, cellular networks, data mining and intelligent transportation systems.

Feng-Jang Hwang received his Ph.D. degree in Information Management from NCTU in 2011. He is the Senior Lecturer of Operational Research at the School of Mathematical and Physical Sciences, Transport Research Centre, University of Technology Sydney. His research interests center around Operations Research, Industrial Optimization, Management Information Science, Data Science, Intelligent Transportation and Logistics, and Computational Intelligence.

Hsu-Yang Kung received his Ph.D. degrees in Computer Science and Information Engineering from National Cheng-Kung University, Taiwan, ROC. He is currently a distinguished professor in the Department of Management Information Systems, National Pingtung University of Science and Technology, Taiwan, ROC. Prof. Kung published around 300 academic papers and obtained 14 the best paper awards. He obtained the special talent award 8 times and the excellent team award 8 times in the Opens Software Development Project Plan from Ministry of Science and Technology. He obtained the outstanding performance and special contributions award in the Innovative Talents Promotion Program for the Communication Software from Ministry of Education. He dominated more than 100 industrial and academic research projects and owned 26 patens. He also served as a Guest Editor for the journals of Agronomy, Electronics, Mathematical Problems in Engineering, Journal of Electrical and Computer Engineering, Advances in Multimedia, and IEEE TEE. His research interests include IoT middleware, cloud computing, wireless and mobile communications, and embedded multimedia applications.