Use of acceleration data for transportation mode prediction

Muhammad Awais Shafique · Eiji Hato

Abstract   Most smartphones today are equipped with an accelerometer, in addition to
other sensors. Any data recorded by the accelerometer can be successfully utilised to
determine the mode of transportation in use, which will provide an alternative to con-
ventional household travel surveys and make it possible to implement customer-oriented
advertising programmes. In this study, a comparison is made between changes in pre-
processing, selection methods for generating training data, and classifiers, using the
accelerometer data collected from three cities in Japan. The classifiers used were support
vector machines (SVM), adaptive boosting (AdaBoost), decision tree and random forests.
The results of this exercise suggest that using a 125-point moving average during pre-
processing and selecting training data proportionally for all modes will maximise pre-
diction accuracy. Moreover, random forests outperformed all other classifiers by yielding
an overall prediction accuracy of 99.8 % for all three cities.

Keywords   AdaBoost · SVM · Random forest · Decision tree · Transportation Mode

Introduction

Travel survey methods can be broadly classified into two categories. In the first category,
respondents are asked to provide details of their trips based on memory. The second
category is the most recent approach, where travel data is automatically recorded by
devices either placed at fixed locations or carried by the respondents themselves (Hato

M. A. Shafique (✉)
Department of Transportation Engineering and Management, University of Engineering and
Technology, GT road, Lahore 54890, Pakistan
e-mail: awais@trip.t.u-tokyo.ac.jp

M. A. Shafique · E. Hato
Transportation Research and Infrastructure Planning Laboratory, Department of Civil Engineering,
The University of Tokyo, 3-1-7, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

2010). In recent times, researchers have been focussing on the second approach to determine travel patterns, primarily due to its significant benefits compared to conventional travel diaries and questionnaires. Such traditional methods are usually expensive, time consuming and require considerable effort on the part of the respondents. Moreover, the start and end times of the trips reported are usually approximate, while small trips are often left unreported. An additional factor is that people's perceptions of in-vehicle time vary according to different modes of travel. For example, a person travelling by car will underestimate the travel time, whereas the same person travelling via public transport will overestimate it (Ettema et al. 1996; Stopher 1992). This decreases the accuracy of the data collected and, in turn, affects subsequent transportation planning and design.

For the purpose of automatic travel data collection, researchers now employ sensors such as global positioning systems (GPS) and accelerometers, among others. A GPS can locate the position of a device anywhere in the world with varying accuracy, depending on factors such as the number of satellites in view. An accelerometer, on the other hand, measures the acceleration of a device in three directions with respect to gravitational force. This means that when the device is placed on a flat surface, an acceleration of 1 g is detected in a downward direction, whereas zero acceleration is recorded in the other two directions. Modern smartphones are now equipped with both of the above sensors, so any methodology developed using either or both sensors can be very easily applied via smartphones. The rapid global increase in smartphone usage, especially in developed countries, offers a perfect opportunity to utilise them to collect travel data.

A major concern for researchers is to accurately detect the mode of transportation used by a person carrying a device (either a smartphone or a purpose-built device with the necessary embedded sensors). Mode determination will not just prove beneficial for the transportation sector, but will also pave the way for a new and effective means of advertising. For example, if a user's location and mode of transportation are known in real time, a message can be sent to his or her mobile phone advertising the nearest facilities available in connection with the mode detected. In addition, products relating to a particular mode used can be advertised directly to the user. In this way, the data accumulated can be used to implement a targeted customer-oriented advertising programme.

For the purpose of mode detection, the analyst can currently avail of different types of classification algorithms. Some of these are listed in Table 1, along with their advantages and disadvantages and the methodologies associated with each. A number of researchers have compared the various algorithms in comparative studies, some of which are summarised in Table 2. The four classifiers used in this study, namely, AdaBoost, SVM, decision tree and random forests, were selected based on the results derived from existing comparative studies. Those algorithms have exhibited good performance in numerous studies, and their respective advantages and disadvantages can be seen in Table 1. The current study compares the algorithms with a view to ascertaining the one most appropriate for transportation mode detection.

Related work

The related work can be divided into three sections depending on the sensors used to determine the travel mode, as follows: GPS only, accelerometer only and GPS with accelerometer. Each section is detailed below. GSM communications (Anderson and Muller 2006; Sohn et al. 2006) and local area wireless technology (Wi-Fi) (Mun et al.

Table 1 A comparison among major classification algorithms

Algorithm	Approach	Advantages	Disadvantages
Nearest neighbor (For details see Besag (1972))	For a test example the closest training example is detected and the same label is used	• Easy implementation • Understandable results • Complex functions	• Every example need to be stored • Choosing similarity matric • Slow classification
Decision Tree (for details see Safavian and Landgrebe (1991))	A tree of decisions is constructed to classify the test data	• Easy to understand • Good for identifying critical features • Fast classification	• Slow to train • Over-fitting • Hard to represent functions like Parity and XOR
Artificial neural networks (for details see Mitchell (1997))	A neural network is developed where the input/output nodes are defined and the hidden internal nodes are learned	• Applicable for non-linear functions • Multiple outputs possible	• Difficult to interpret
Support vector machines (for details see Boser et al. (1992))	Given the training data an optimization problem is defined and the optimum margin classifier is attained.	• Good performance • Efficient learning • Powerful kernel-based framework	• Difficult to interpret • No direct multiclass SVM • Slow to train for large problems
Adaptive boosting (for details see Freund and Schapire (1997))	Strong classifier is constructed by linear combination of simple weak classifiers	• Easy implementation • Good generalization • Feature selection on very large sets of features	• Suboptimal solution • Sensitive to noisy data
Fuzzy systems (for details see Wang (1994))	Computation is done with words	• Robust approach • Can be used in complex systems • Low computational cost • Easily understandable • Easy representation of expert knowledge	• Number of rules grow exponentially inverse with the accuracy level • Must be combined with an adaptive system if some heuristics is desired

Table 1 continued

Algorithm	Approach	Advantages	Disadvantages
Random forest (for details see Breiman (2001))	A group of decision trees is grown which then randomly uses the data for prediction.	• Good accuracy • Can handle many input variables • No pruning required • Cross validation is unnecessary • Identifies important features	• Loss of interpretability
Naïve bayes (for details see Lewis (1998))	Bayes' theorem is applied with strong independence assumptions between the features.	• Easy implementation • Fast learning • No problem regarding dimensionality	• Poor performance if independence assumptions do not hold • Over-fitting

Table 2 Some previous algorithm comparison studies

Study	Algorithms compared	Best Algorithm
Caruana and Niculescu-Mizil (2006)	Support vector machines, neural nets, logistic regression, naïve bayes, memory-based learning, random forest, decision trees, bagged trees, boosted trees, boosted stumps	Boosted trees
Nick et al. (2010)	Naïve bayes, support vector machines	Support vector machines
Reddy et al. (2010)	Decision trees, K-means clustering, naïve bayes, nearest neighbor, support vector machines, continuous hidden markov model, decision trees with discrete hidden markov	Decision tree with discrete hidden markov model
Stenneth et al. (2011)	Naïve bayes, bayesian network, decision trees, random forest, multilayer perceptron	Random forest
Yu et al. (2013)	Decision trees, AdaBoost, support vector machines	Support vector machines

2008), are also employed for the purpose of mode detection, but due to their relatively low accuracy, they will not be mentioned here.

GPS only

Various studies confirm that the use of GPS data loggers has resulted in greater data accuracy compared to conventional paper diaries and telephone surveys (Forrest and Pearson 2005; Ohmori et al. 2005; Wolf et al. 2003). Accuracy is further enhanced when GPS data is used on a geographic information system (GIS) application (Chung and Shalaby 2005; Schönfelder et al. 2002; Tsui and Shalaby 2006; Wolf et al. 2001).

In their study, Tsui and Shalaby 2006 recorded the GPS logs of participants based in Toronto. The average and maximum speed, in addition to the acceleration, were deduced from the GPS data. This data, along with information relating to public transport routes, was used to determine the transport modes. The prediction accuracy achieved was more than 90 %, a figure slightly higher than the method that did not use GIS information. Chung and Shalaby (2005) asked one participant to repeat 60 trips for the Toronto 'Transportation Tomorrow Survey', carrying a GPS device. The recorded GPS data was used in combination with GIS data to achieve a mode prediction accuracy of 92 %.

A study by Stopher et al. (2008) used a probability matrix to differentiate between travel modes. Trip characteristics such as bicycle ownership, maximum speed, average speed and most frequent speed defined whether a person was walking, cycling or using motorised transport. Further GIS data was utilised to distinguish between motorised transportation modes. In another study by Bohte and Maat (2009), a similar methodology was used for mode detection. Firstly, the average and maximum speeds were used to determine whether the respondent was walking, cycling or driving a car. Secondly, in line with the rules of interpretation, GPS data was plotted on GIS maps to determine whether the motorised trip was by car or by train. A prediction accuracy of 70 % was achieved.

Stenneth et al. (2011) took a different approach to solving the problem. They used GPS data, along with ground conditions, to extract the features to be used in learning algorithms. The features included the average accuracy of the GPS coordinates, average speed, average heading change, average acceleration, bus location proximity, rail line trajectory proximity, bus stop proximity rate and zip code. The classification algorithms used were as follows: (1) naïve Bayes, (2) Bayesian network, (3) decision tree, (4) random forests and (5) Multilayer Perceptron (MLP). The results suggested that random forests is the best classifier, with an average prediction accuracy of 93.7 %.

GPS and GIS information was also used by Chen et al. (2010) to distinguish between six different transportation modes in the city of New York. Prediction accuracy ranged from 60 to 95 %.

Our study is different in the sense that GPS data is not used at all. Although GPS data has been shown to work well for mode detection, certain disadvantages are associated with it. The main difficulty is the drop in accuracy due to signal loss or degradation during warm or cold starts, and in 'urban canyons' (Gong et al. 2012; Schuessler and Axhausen 2009; Stopher et al. 2008). Warm and cold starts happen when a GPS logger requires between 5 and 30 s more to find enough satellites for accurate location detection after being off (or underground) for a long period of time. In densely built central business districts (CBDs), satellite signals do not generally reach the GPS device directly but are bounced off tall buildings. This is known as the urban canyon effect. The above drawbacks associated with GPS use tend to decrease the accuracy of the results extracted from GPS data. Furthermore, respondents' privacy concerns are also a problem in this area. If a smartphone is used as

the data collection instrument, developing a methodology using acceleration data alone will not only address the above problems but will also extend the battery time of the smartphone during data collection, as the GPS sensor will not be activated.

Accelerometer only

Much of the available research focusses on using accelerometer data for the classification of physical activity (Bao and Intille 2004; Lester et al. 2006; Tapia et al. 2007), including research conducted using iPhone accelerometer data (Nham et al. 2008). In that case, data from only three participants was used to predict the mode of travel. For classification purposes, the LIBSVM framework (Chang and Lin 2011) was used, where the first 70 % of the data set for each mode was selected as the training set and the remaining 30 % was used as the test set. While the prediction results were reasonably accurate but highly varied among the participants, ranging from 88 to 97 %, the overall validity of the study is questionable in light of the small amount of data used. Nick et al. (2010) collected acceleration data for three modes including walking, car and train. In total, 90 % of the entire data set was used to train two classification algorithms, namely naïve Bayes and SVM, whereas the remaining 10 % was used as test data. According to the results, SVM outperformed naïve Bayes and achieved a classification accuracy of 97.32 %.

In a recent study by Hemminki et al. (2013), 16 participants from four countries collected accelerometer data spanning more than 150 h and covering six modes of transportation. A mean recall accuracy of 82.4 % was achieved.

Our work provides a comparison between the main classifiers used in research on this area. Although only four modes were classified (due to data constraints), the accuracy achieved was outstanding (mean 99.8 %). Therefore, this methodology is also expected to work quite satisfactorily for additional modes.

GPS with accelerometer

The use of GPS data accompanied by accelerometer data is a relatively novel approach, and few studies have reported methodologies utilising both types of sensor data. For instance, Reddy et al. (2010) used the decision tree followed by the discrete hidden markov model (DHMM) to identify transportation modes, including stationary, walking, running, biking and motorised transport. The classification system was tested on a data set obtained from sixteen participants and an accuracy of 93.6 % was achieved.

A comparison between the various pre-processing techniques used in several studies was carried out by Figo et al. (2010). Data for prediction and comparison purposes was obtained for three activities, walking, running and jumping. Almost 50 % of the data was used to train the algorithm. The results suggest that for the three-activity scenario, the best frequency-domain techniques yielded comparable results to the best time-domain techniques. But for the two-activity scenario, the best time-domain techniques prevailed.

Moreover, Nitsche et al. (2012) gathered 266 h of travel data with the help of 14 test participants and extracted 72 features for use in probabilistic classifiers. The results ranged from 50 to 98 % over different modes of transportation.

Feng and Timmermans (2013) carried out a study comparing the following three approaches: GPS data only, accelerometer data only and GPS combined with accelerometer data. The study used the Bayesian belief network model for classification purposes. The results showed that the acceleration only approach, with a mean validation accuracy of 88.87 %, works better than GPS only (mean 78.4 %), but the combined data approach

outperforms both of them, with a mean validation accuracy of 91.7 %. The use of bicycle ownership, motorcycle ownership and car ownership variables presents a small constraint to the goal of collecting data automatically without putting any burden on the respondents.

Data collection

The data was collected from three cities in Japan, namely, Niigata, Gifu and Matsuyama. In Niigata, the surveys were conducted during January and February 2011 and involved 12 participants; in Gifu, they were conducted in December 2010 and January 2011 and involved 8 participants; and in Matsuyama, they were conducted in November 2010 and January 2011 and involved 26 participants. The data collected can be classified into location data and trip data.

Collection method

The location data was recorded using Behavioural Context Addressable Loggers in the Shell (BCAL) (Hato 2010). BCALs, shown in Fig. 1, are purpose-built wearable devices equipped with different sensors, in addition to a GPS and an accelerometer. They can record location as well as acceleration in three directions, a task that is now possible using modern smartphones. The BCALs observed the various sensors' readings at a frequency of 16 Hz or 16 readings per second, but the readings transmitted to the server were spaced out at an average of 5 s. Hence, the maximum, minimum and average readings were calculated by the device for each 5 s interval and then recorded by the server. The wearable devices were kept in the same position throughout the trip so that accelerations in different directions could be judged easily.

The trip data was collected using paper-based travel diaries in which the respondents were asked to record the details of their everyday trips. Feedback calls were made to the respondents to correct any mistakes made during reporting. Again, this is a task that can be fulfilled using smartphones, a method used by many researchers. A simple application developed for the smartphone can be utilised to record the start and end of a trip, as well as the mode of transportation used.

Data description

The location data comprised GPS data and accelerometer data. The accelerometer data recorded was the minimum, maximum and average acceleration in movement, crosswise and vertical directions. Moreover, resultant acceleration and average resultant acceleration were also noted. The trip data covered the information regarding each trip, i.e., the date, start time, end time and mode used.

Amount of data

Table 3 presents the raw location data and the mode-assigned data (discussed in "Data collection" section: Mode assignment) for each city. The table also shows the assignment of the data to various modes. Table 4 displays the trip share for each mode.

Due to data limitations, the analysis was carried out for four modes only. Acceleration data relating to the bus as a fifth mode was either non-existent or so small that it was not

Fig. 1 BCALs equipped with various sensors

Table 3 Amount of data collected through BCALs

Data\City	Niigata	Gifu	Matsuyama
Raw location data	341,712	258,388	507,014
Mode assigned data	229,502 (100.00 %)	143,626 (100.00 %)	269,375 (100.00 %)
Walk	164,078 (71.49 %)	83,645 (58.24 %)	168,687 (62.62 %)
Bicycle	3,214 (1.4 %)	24,559 (17.09 %)	15,404 (5.72 %)
Car	61,785 (26.92 %)	34,678 (24.14 %)	81,653 (30.31 %)
Train	425 (0.19 %)	744 (0.52 %)	3,631 (1.35 %)

Table 4 Number of trips recorded

Mode\City	Niigata	Gifu	Matsuyama
Walk	662	342	861
Bicycle	12	180	90
Car	306	219	386
Train	4	3	40
Total	984	744	1,377

treated separately but simply merged with the car travel data. Similarly, only one trip was recorded for Shinkansen (the high-speed train), and instead of adding a new mode, it was included with the train data.

Mode assignment

The location data was filtered in terms of the trip data. For example, if accelerometer data was recorded with respect to a user for a specific day, but the user had not registered any trips for that particular day in the trip data, then the accelerometer data recorded was of no use. Moreover, data sets with zero acceleration ('rest' position) were also discarded.

Using the departure and arrival times listed in the trip data, the corresponding data sets in the location data were assigned the respective mode of transportation, as shown in Fig. 2. After the mode of transportation was assigned to the location data, the remaining data sets were disposed of. The reason some data remained unassigned is that the accelerometer data may have contained data sets recorded before the start of the trip or after the end of the trip. The remaining data was used in subsequent pre-processing and analysis.

Methodology

Elementary analysis

A distinction between the modes was detected upon careful examination of the acceleration data. For instance, Figs. 3, 4, 5 and 6 show part of the acceleration data for each mode. It can be observed that walking has maximum variability, followed by cycling. This could be due to excessive movement by the traveler carrying the device. On the other hand, the car and train modes showed relatively small acceleration variability, probably due to the smooth travelling environment. Therefore a clear distinction can be perceived between the different modes by just inspecting the acceleration data.
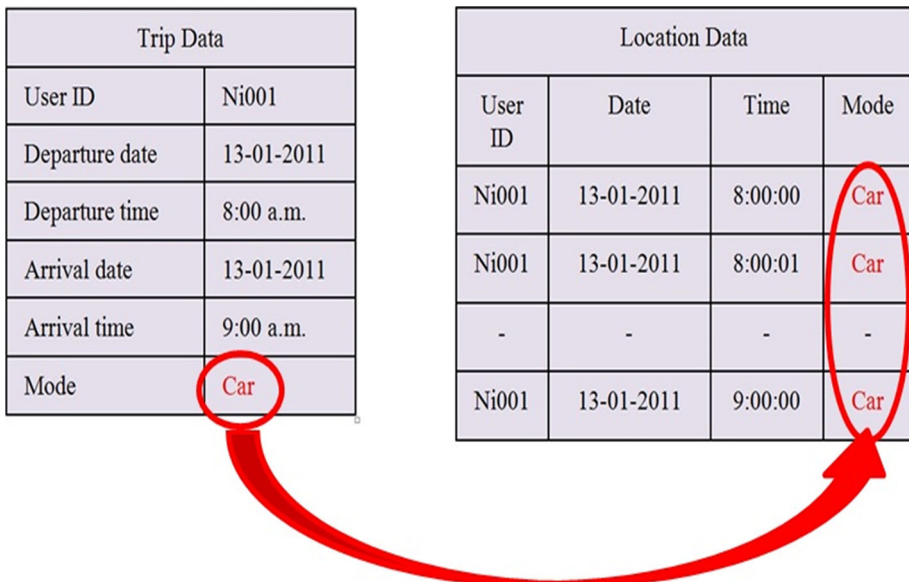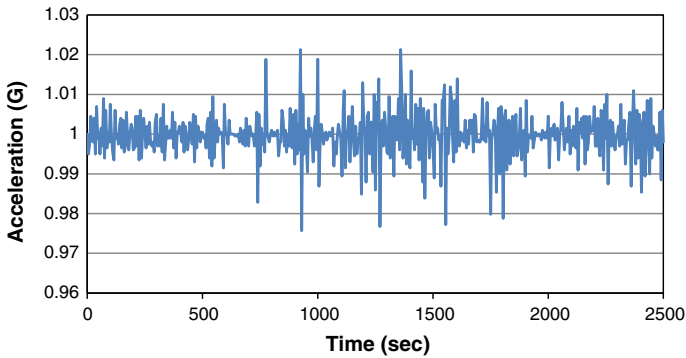

Fig. 2 Example of mode assignment

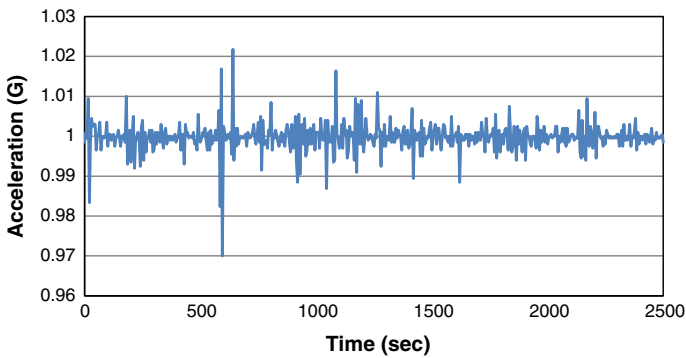**Fig. 3** Average resultant acceleration for walking

Fig. 4 Average resultant acceleration for bicycling

Pre-processing

Pre-processing was applied in two stages. First, the moving average was calculated, followed by the differences between each mode.

The moving average was calculated at 25 point, 50 point, 75 point, 100 point and 125 point in order to identify the trend most likely to maximise classification accuracy.

In this case, x denotes the various data entries for acceleration in any direction, n is the total number of data entries and k is the window size (25, 50, 75, 100 and 125) for calculating the moving average. At any position i within the data, the window will cover x_j entries to calculate the moving average. The window will keep the reference entry x_i at the centre, except at the start and end of the data set. As the reference entry x_i moves closer to the start or end of the data set, the window will be suppressed. As a solution to this, the window was halved at the start and end of the data set, with the reference entry kept at one end of the window rather than placed in the centre. The following Eqs. 1 and 2 were formulated for the calculation of the k point average. Equation 2 was used only for average resultant acceleration.
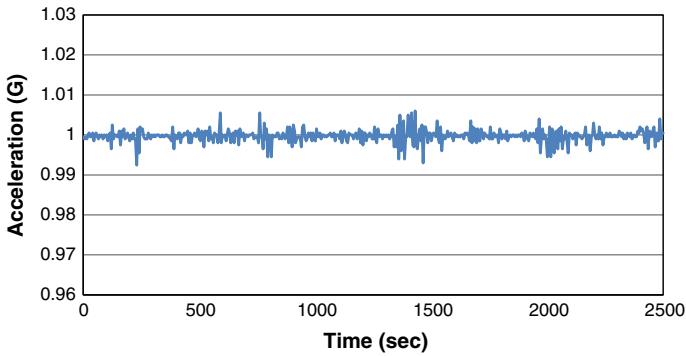
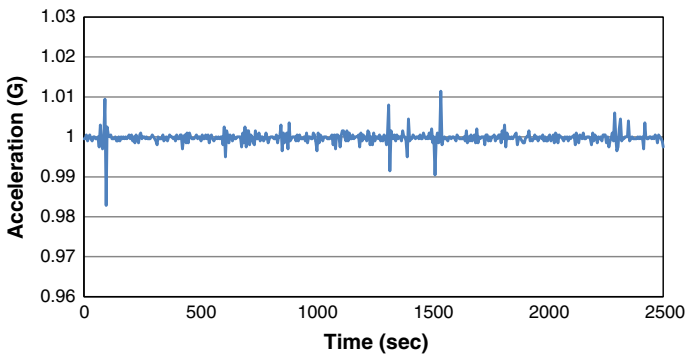Fig. 5 Average Resultant Acceleration for automobile travel

Fig. 6 Average resultant acceleration for train travel

$$(k\,point\,Avg)_i = \begin{cases} \dfrac{2}{k}\displaystyle\sum_{j=i}^{i+k/2} x_j & if\ i \le k/2 \\[2em] \dfrac{1}{k}\displaystyle\sum_{j=i-k/2}^{i+k/2} x_j & if\ k/2 < i < n - k/2 \\[2em] \dfrac{2}{k}\displaystyle\sum_{j=i-k/2}^{i} x_j & if\ i \ge n - k/2 \end{cases} \tag{1}$$

$$(k\,point\,Avg)_i = \begin{cases} \dfrac{2}{k}\displaystyle\sum_{j=i}^{i+k/2} |x_i - x_{i-1}|_j & if\ i \le k/2 \\[2em] \dfrac{1}{k}\displaystyle\sum_{j=i-k/2}^{i+k/2} |x_i - x_{i-1}|_j & if\ k/2 < i < n - k/2 \\[2em] \dfrac{2}{k}\displaystyle\sum_{j=i-k/2}^{i} |x_i - x_{i-1}|_j & if\ i \ge n - k/2 \end{cases} \tag{2}$$

Equation 1 shows that at the start of the data set, that is, when the reference position i had not yet exceeded the $k/2$ mark, a window of size $k/2$ was used to calculate the average value, with the reference value at the start of the window. Similarly, at the end, the $k/2$-sized window was used, keeping the reference value at the end of the window. Between these two extremes, the window size was increased to k, with $k/2$ before i and $k/2$ after i.

In this way, moving averages were calculated for maximum, minimum and average accelerations in the movement, crosswise and vertical directions. Furthermore, moving averages were also calculated for resultant and average resultant acceleration (acc_{res}, $acc_{avg.res}$). After the original values were replaced with the moving averages, the differences between maximum and minimum accelerations (acc_{max}, acc_{min}) were calculated for all three directions ($cross, vert, mov$), and their differences subsequently calculated. Moreover, the differences between average accelerations (acc_{avg}) along the three directions were also calculated. Equations 3–9 show the complete procedure used for the difference calculations.

$$D_d = acc_{max.d} - acc_{min.d} \ for \ d = cross, vert, mov \tag{3}$$

$$D_1 = D_{cross} - D_{vert} - D_{mov} \tag{4}$$

$$D_2 = D_{vert} - D_{mov} - D_{cross} \tag{5}$$

$$D_3 = D_{mov} - D_{cross} - D_{vert} \tag{6}$$

$$D_{a1} = acc_{avg.cross} - acc_{avg.vert} - acc_{avg.mov} \tag{7}$$

$$D_{a2} = acc_{avg.vert} - acc_{avg.mov} - acc_{avg.cross} \tag{8}$$

$$D_{a3} = acc_{avg.mov} - acc_{avg.cross} - acc_{avg.vert} \tag{9}$$

Figure 7 shows the entire pre-processing method. After pre-processing, the final features were as follows: maximum, minimum and average acceleration along the three directions; differences between maximum and minimum (D_x, D_y, D_z); their differences (D_1, D_2, D_3); differences between average accelerations (D_{a1}, D_{a2}, D_{a3}); resultant acceleration and average resultant acceleration. In addition, moving averages were calculated for all values.

Training and test data selection

As the data for each mode was different, the training data was randomly selected in the following two ways:

(1)   Equal number selection
(2)   Equal proportion selection

While equal number selection ensures that all the modes are equally represented in the training data set, the algorithm lacks sufficient training for the most frequently occurring mode in the test data set. Conversely, equal proportion selection ascertains that training is done proportionally for the test data set, but the modes are not represented equally in the training data set. This variation may affect the prediction results.
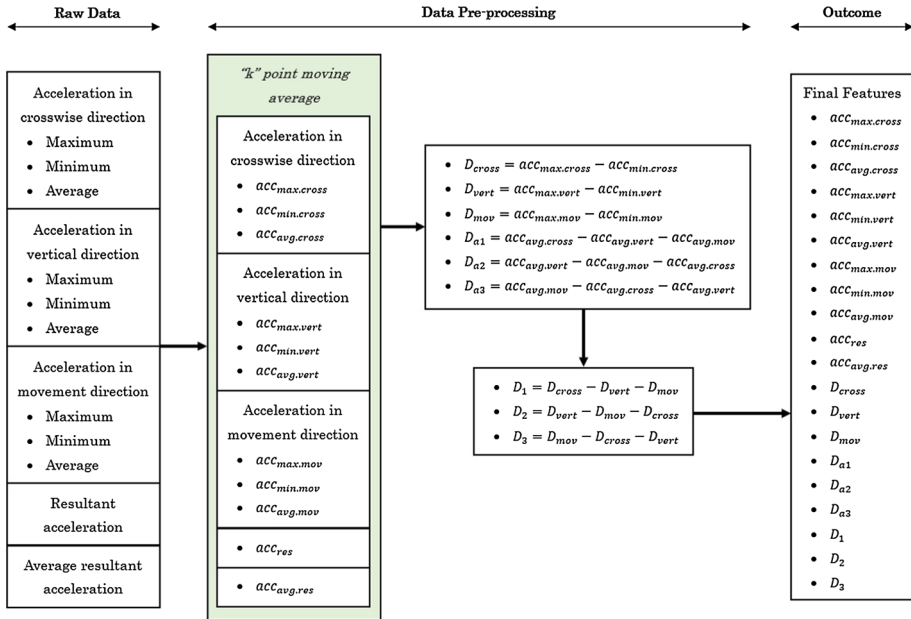
Fig. 7 Pre-processing and feature extraction

Table 5 Amount of training data used for travel mode classification

City	Data selection method	Mode				
		Walk	Bicycle	Car	Train	Total
Niigata	Total	164,078	3,214	61,785	425	229,502
	Equal number	298	298	298	298	1,192
	Equal proportion	114,855	2,250	43,250	298	160,653
Gifu	Total	83,645	24,559	34,678	744	143,626
	Equal number	521	521	521	521	2,084
	Equal proportion	58,552	17,191	24,275	521	100,539
Matsuyama	Total	168,687	15,404	81,653	3,631	269,375
	Equal number	2,542	2,542	2,542	2,542	10,168
	Equal proportion	118,081	10,783	57,157	2,542	188,563

Equal number selection

For each city, the mode with the least data was selected and the number corresponding to 70 % of that data was calculated. The data equal to that number was then randomly selected from each mode to form the training data set, leaving the rest as a test data set.

In this way, no matter how much difference was present between the modes, the training data always comprised equal numbers from each. Table 5 shows the amount of training data selected for each city.

Equal proportion selection

A total of 70 % of data for each mode was randomly selected to form the training data and the remaining 30 % was used to test the algorithms. This method yielded a much larger quantity of training data, which can be seen in Table 5.

Classifiers

In order to determine the classifier that most accurately predicts transportation mode, a comparison was made between (a) Support Vector Machines (SVM); (b) Adaptive Boosting (AdaBoost); (c) decision tree using rpart, and (d) random forests. These classifiers were selected due to their frequent and established use in existing literature. The aim was to identify the best performing algorithm by carrying out a comparison between them.

Support vector machine

SVM is a state-of-the-art classification method that was introduced by Boser et al. (1992). SVM has a vast range of applications in bioinformatics, text recognition, image recognition, robotics and many other fields.

SVM fits into the kernel methods category (Shawe-Taylor and Cristianini 2004). Kernels allow the use of linear methods to solve non-linear problems. However, the efficient use of SVMs depends largely on knowledge of how this classifier works, and the user first needs to decide what pre-processing method to use.

A suitable kernel must then be selected, after which the user faces the difficulty of setting parameters for SVM and the selected kernel. A comprehensive guide for this purpose is provided by Ben-Hur and Weston (2010).

SVM is a linear two-class classifier. For simplification purposes, it is assumed that the two classes are labelled as $+1$ (positive examples) and -1 (negative examples). In the sample case below, x_i is an ith example in a data set $(x_i, y_i)_{i=1}^{n}$, where y_i is the class label associated with that example; boldface $\boldsymbol{x}$ is a vector with components x_i. The linear classifier is specified by a dot product and is defined by the function, as below:

$$\boldsymbol{w}^T\boldsymbol{x} = \sum_i w_i x_i \tag{10}$$

$$f(x) = \boldsymbol{w}^T\boldsymbol{x} + b \tag{11}$$

The vector $\boldsymbol{w}$ is known as the weight vector and b is called the bias. The set of points x, for which $f(x) = 0$, constitutes a hyperplane. This hyperplane, shown in Fig. 8, divides the space into two regions so as to separate the data into two classes.

The circled data points are the points closest to the hyperplane and are called the 'support vectors'. The margin is the distance from the support vectors to the hyperplane. The aim is to maximise the geometric margin $1/\parallel \boldsymbol{w} \parallel$, which is equivalent to minimising $\parallel \boldsymbol{w}^2 \parallel$. This leads to the following optimisation problem:

$$\min \parallel \boldsymbol{w}^2 \parallel \textit{ subject to } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 \; i = 1, \ldots, n \tag{12}$$

With the help of kernels, the concept of linear classifiers can be extended to non-linear problems. Kernels are used because direct computation of non-linear features is very
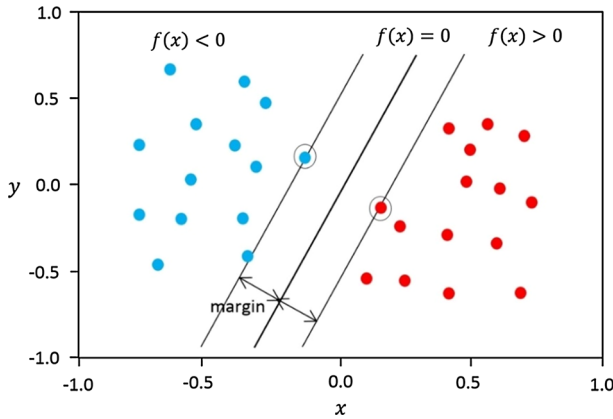
Fig. 8 Linear Classifier with support vectors shown as *circled* data points

expensive in the case of a huge quantity of data. Some famous kernels are shown in Eqs 13–15 below:

$$\text{Linear Kernel } k\left(x, x^{'}\right) = x.x^{'} \tag{13}$$

$$\text{Gaussian Kernel } k\left(x, x^{'}\right) = \exp\left(-\gamma \parallel x - x' \parallel^{2}\right), \quad \gamma > 0 \tag{14}$$

$$\text{Polynomial Kernel } k\left(x, x^{'}\right) = (x.x^{'} + 1)^{d}, \quad d \in N \tag{15}$$

As SVM is a binary class classifier, the one-against-one technique was employed and the correct class was determined using a voting mechanism.

Adaptive boosting (AdaBoost)

As a solution to many of the difficulties of earlier boosting algorithms, AdaBoost was first introduced by Freund and Schapire (1997). Using the same example as that mentioned above for SVM, AdaBoost takes the training data and calls a weak classifier repeatedly. Starting with equal weights for all the examples, the weights for incorrectly classified examples are increased after each round so that the algorithm can focus more on the difficult examples. Consequently, a strong classifier is constructed as shown in Fig. 9.

In this example, the initial weights are $w_i^{(1)} = 1$ for all data points x_i. In order to generate a set of M classifiers, the same number of iterations is done. At each iteration, W is the sum of the weights of all data points, whereas W_e is the sum of the weights of misclassified data points.

For $m = 1 \, to \, M$

- Select the classifier k_m which minimizes W_e

$$W_e = \sum_{k_m(x_i) \neq y_i} w_i^{(m)} \tag{16}$$

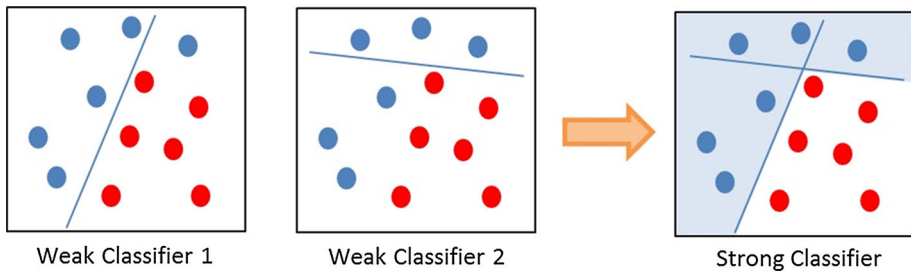- Set the weight α_m of the classifier

Fig. 9 Concept of AdaBoost

$$\alpha_m = \frac{1}{2} ln \left(\frac{1 - e_m}{e_m} \right) \tag{17}$$

Where $e_m = W_e/W$

- Update the weights of the data points for the next iteration.

$$w_i^{(m+1)} = \begin{cases} w_i^{(m)} e^{\alpha_m} & if \ k_m(x_i) \neq y_i \\ w_i^{(m)} e^{-\alpha_m} & if \ k_m(x_i) = y_i \end{cases} \tag{18}$$

Similar to SVM, AdaBoost can only solve binary class problems, so the one-against-all technique was used and the correct class was assigned only when there was one unique answer. Consequently, some of the data remained unclassified, and this was again investigated using AdaBoost in a similar way. In the end, SVM was used to classify any remaining unclassified data.

Decision trees using rpart

A decision tree is a classifier that employs recursive partitioning to arrive at a decision. The data set is split into branch-like segments, and those segments form an inverted decision tree that originates from a starting node called the root. The root has no incoming edge, whereas all the other nodes in the tree have exactly one incoming edge. Nodes with outgoing edges are known as internal or test nodes, while the rest are known as leaves, terminals or decision nodes. Each internal node splits the data into two or more segments according to certain rules, which depend on the attribute values.

Each terminal node corresponds to a target class. The data is classified while navigating from the root down to the leaves. Along the way, internal nodes decide the path of the decision in light of certain rules, which are also defined by the algorithm. Figure 10 presents a simple decision tree for a sample trip with only two features.

Rpart is an acronym for 'recursive partitioning', a statistical package written in the R programming language, which applies Classification and Regression Trees (CART), as discussed by Breiman et al. (1984). This partitioning method can be applied to many different kinds of data. In this case, there was a classification problem and pruning was carried out to eliminate the effects of over fitting.

Random forests

Random forest, developed by Breiman (2001), is an ensemble classification and regression method that constructs a number of decision trees at the training level, predicts the class using each tree and outputs the final class as the mode of the individually predicted classes. Because the classification method involves tree-like structures, and because randomness is inherently built into the procedure, the method is named 'random forests'. One of the major advantages of random forests is that pruning is not required.

Each unpruned tree is grown using CART. At each node, a subset of features from the data is randomly selected and the best split is made using that subset. A large number of trees can be grown, and each tree uses nearly 63 % of the given training data randomly selected. The remaining 37 %, known as 'Out of Bag' or OOB data, is used to test each tree. Obviously, it will be different for each tree. Trees are grown by means of binary partitioning. At each node, a subset of the predictors or features is randomly selected. Typically the subset is $\sqrt{k}$, where k is the total number of features. Among the subset features, the best feature is used for the split. For the resulting nodes, new feature subsets are selected randomly. New data is predicted using all the trees and the result is finalised by taking the mode of the individual results (classification problem) or their average (regression problem). Figure 11 presents the general procedure involved in random forests.

Results and discussion

The overall classification results of the classifiers for the different moving averages, as well as the two types of training data selection methods, are summarized in Figs. 12, 13 and 14. From the figures, it is evident that maximum prediction accuracy can be achieved by employing a 125-point moving average at the pre-processing stage. For the 125-point moving average, Table 6 shows the overall classification accuracies, while Table 7 gives the detailed results. The accuracy calculated can be considered producer accuracy. For example, if the prediction accuracy is 85 %, this means that 85 % of the known data carrying a certain class label (ground truth) is returned with the same label by the algorithm. The accuracies were calculated after creating confusion matrices and dividing the number of correct predictions for each mode by the total quantity of data in the test data set that is linked to that mode.

It can also be observed from the figures, as well as from the results listed in the tables, that the equal proportion method is better than the equal number method, but some of the detailed results show differently. For instance, in the equal proportion method, SVM and AdaBoost seem to perform well, with overall accuracy exceeding 85 % in all cases. However, a breakdown of the accuracies at mode level reveals that the accuracy in terms of train transport prediction is very poor, in fact zero in case of Niigata and Matsuyama. This is because the amount of data corresponding to train transportation in the training data is relatively very small, which results in a zero prediction accuracy, even for the training data itself.

Random forests performs best in all cases. In particular, its accuracy is very high, at 99.8 %, for the 125-point moving average using the equal proportion method. Even in the equal number method, the overall accuracy is greater than 91 %, which is quite impressive. The next best performer is decision tree, followed by AdaBoost and then SVM.
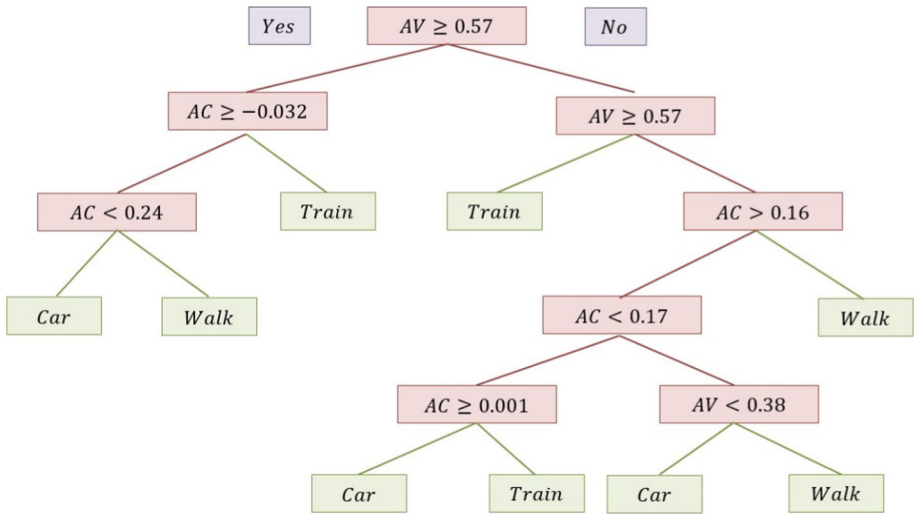
AV = Average acceleration in vertical direction (G)
AC = Average acceleration in cross-wise direction (G)

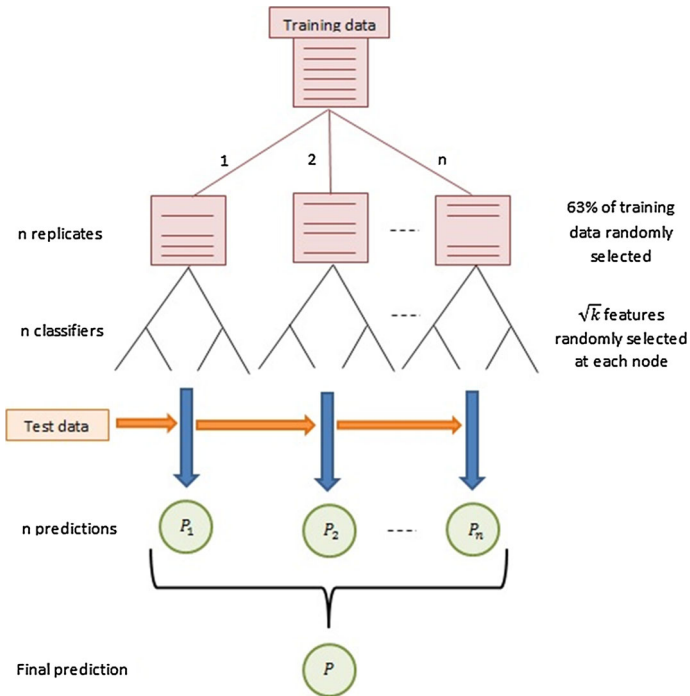Fig. 10 Decision tree presenting mode classification using acceleration data

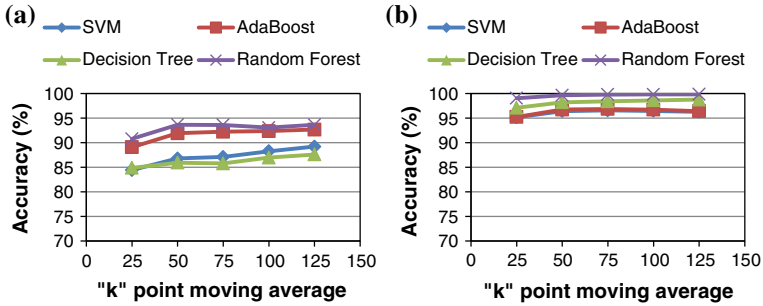Fig. 11 General procedure of random forest

Fig. 12 Prediction accuracy for Niigata city using (**a**) equal number method and (**b**) equal proportion method
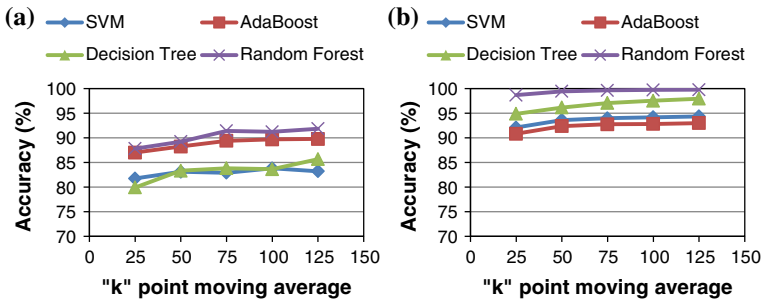

Fig. 13 Prediction accuracy for Gifu city using (**a**) equal number method and (**b**) equal proportion method
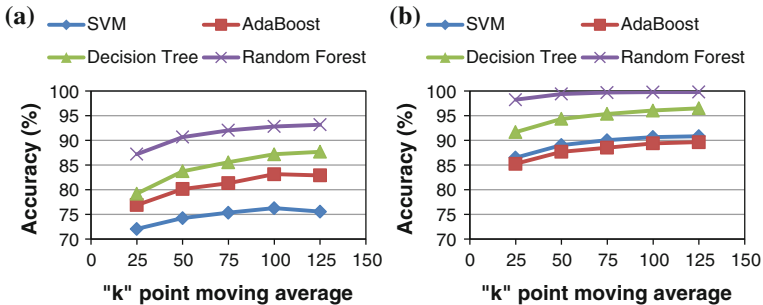

Fig. 14 Prediction accuracy for Matsuyama city using (**a**) equal number method and (**b**) equal proportion method

The developed methodology was tested for three cities in order to establish the stability as well as the broader applicability of the approach. The results suggest that similar classification accuracy was achieved for the three cities. This is an indication that the approach is stable and might yield a good level of accuracy for other cities in Japan. But to confirm this, more data is required.

A careful examination of the results reveals that when using random forests, the prediction accuracy of the train transportation mode is the highest of all the modes, in fact

Table 6 Overall classification results at 125 point moving average

Selection method	City	SVM		AdaBoost		Decision tree		Random forest	
		Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
Equal number	Niigata	90.44	89.21	98.49	92.67	100.00	87.59	100.00	93.64
	Gifu	85.80	83.22	96.74	89.77	100.00	85.67	100.00	91.85
	Matsuyama	81.25	75.54	90.96	82.91	99.28	87.68	100.00	93.17
Equal proportion	Niigata	96.37	96.30	96.61	96.42	99.20	98.78	100.00	99.86
	Gifu	94.41	94.36	93.15	93.00	98.76	97.97	100.00	99.79
	Matsuyama	90.84	90.85	89.89	89.65	97.42	96.52	100.00	99.81

100 %, in the case of the equal number method. However, the same mode is predicted with the least accuracy relatively for the equal proportion method. This suggests that the prediction accuracy of the train mode can easily be improved by collecting more data so as to increase its representation in the training data set. Therefore, the optimum solution is to collect a comparable amount of data for each mode so that both selection methods will yield a training data set of a similar size.

Conclusion

This study shows that by using only the acceleration data, the transportation mode being used by the device carrier can be detected with a high level of accuracy. The developed methodology has the potential to complement or partially replace conventional travel survey methods. Furthermore, the data required for the developed approach can be collected using smartphones, which will increase its applicability. Automatic mode detection will assist transportation planners in studying and modelling people's travel behaviour more easily and with higher accuracy. This, in turn, will improve subsequent planning and design works.

Apart from a good classification algorithm, the training sample size and appropriate pre-processing are also vital for achieving better results, which is a primary focus of this study. The data was collected by respondents in three Japanese cities, namely Niigata, Gifu and Matsuyama. The training sample size was set based on two data selection methods. Of the two data selection methods tested, the equal proportion method performed better. Moreover, regarding the pre-processing phase, varying window sizes were used to calculate moving averages. A 125-point moving average improves the prediction accuracy relative to others, although the variation is minimal. Finally, of the four algorithms used in this study, random forests outperformed all the others. A combination of all of the optimum conditions described above yielded an overall prediction accuracy of 99.8 %. The surveyed cities exhibited similar classification accuracies, indicating that this approach might also be applied to other areas with the expectation of good results.

This study highlights a limitation with respect to the SVM and AdaBoost algorithms. Minimal representation of the train transportation mode in the training data set following the equal proportion selection method resulted in the total misclassification of train data during prediction. This shows that the training of SVM requires equal or comparable representation from all classes, and the same is true for AdaBoost. On the other hand, no

Table 7 Classification results at 125 point moving average

Selection method	City	Mode	SVM Train (%)	SVM Test (%)	AdaBoost Train (%)	AdaBoost Test (%)	Decision Tree Train (%)	Decision Tree Test (%)	Random Forest Train (%)	Random Forest Test (%)
Equal number	Niigata	Walk	92.95	92.91	97.32	93.99	100.00	87.85	100.00	94.39
		Bicycle	90.94	90.05	98.99	96.54	100.00	91.50	100.00	97.81
		Car	77.85	79.27	97.65	88.96	100.00	86.69	100.00	91.43
		Train	100.00	100.00	100.00	100.00	100.00	97.64	100.00	100.00
		All	90.44	89.21	98.49	92.67	100.00	87.59	100.00	93.64
	Gifu	Walk	85.03	85.92	95.20	90.80	100.00	85.37	100.00	93.06
		Bicycle	90.98	87.16	96.93	89.82	100.00	83.36	100.00	90.88
		Car	74.09	73.78	94.82	87.16	100.00	87.92	100.00	89.57
		Train	93.09	94.62	100.00	100.00	100.00	99.55	100.00	100.00
		All	85.80	83.22	96.74	89.77	100.00	85.67	100.00	91.85
	Matsuyama	Walk	77.50	77.07	81.83	80.91	98.47	86.26	100.00	91.81
		Bicycle	87.25	86.71	94.26	91.10	99.76	93.40	100.00	98.27
		Car	71.75	70.32	88.28	85.56	98.94	89.60	100.00	95.11
		Train	88.51	87.51	99.49	99.54	99.96	97.52	100.00	100.00
		All	81.25	75.54	90.96	82.91	99.28	87.68	100.00	93.17
Equal proportion	Niigata	Walk	97.95	97.84	98.17	97.92	99.66	99.45	100.00	99.96
		Bicycle	77.96	79.88	84.76	86.41	94.13	92.22	100.00	99.38
		Car	93.81	93.70	93.55	93.37	98.32	97.44	100.00	99.64
		Train	0.00	0.00	30.54	37.01	89.60	85.04	100.00	99.21
		All	96.37	96.30	96.61	96.42	99.20	98.78	100.00	99.86
	Gifu	Walk	97.82	97.70	97.55	97.31	99.28	98.75	100.00	99.84
		Bicycle	88.10	88.04	82.18	81.49	97.91	96.73	100.00	99.78
		Car	91.46	91.57	90.93	91.43	98.23	97.14	100.00	99.71

Table 7 Classification results at 125 point moving average

Selection method	City	Mode	SVM		AdaBoost		Decision Tree		Random Forest	
			Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
	Matsuyama	Train	56.05	56.05	63.92	61.88	92.90	89.69	100.00	98.65
		All	94.41	94.36	93.15	93.00	98.76	97.97	100.00	99.79
		Walk	94.10	94.09	94.10	94.13	98.72	98.14	100.00	99.88
		Bicycle	64.50	63.75	51.79	49.12	89.58	87.06	100.00	99.70
		Car	93.12	93.29	91.47	91.17	96.97	95.95	100.00	99.80
		Train	0.00	0.00	20.38	19.47	80.45	74.38	100.00	97.43
		All	90.84	90.85	89.89	89.65	97.42	96.52	100.00	99.81

such constraints exist in the case of decision tree and random forests. A further observation was made regarding the computational time required by the algorithms. SVM and Ada-Boost are very time consuming when it comes to large data sets like those used in this study, whereas decision tree and random forests outmatch them in this respect also.

However, the ideal scenario is to have a nearly equal amount of data for each contributing mode and then use the equal proportion method. In this manner, the strengths of both methods will be combined and yield even better prediction results. One of the limitations of this study relates to the fixed positioning of the data collection device while its carrier was travelling. The positioning should be flexible, especially in cases where purpose-built devices are replaced by smartphones. The newly developed methodology needs to be modified and extended to incorporate varying placement of the device. Furthermore, the new approach should also be checked for additional modes. To this end, behaviour models can also be incorporated into the analysis in order to enhance accuracy, and may be especially beneficial in the case of insufficient collected data.

References

Anderson, I., Muller, H.: Practical Activity Recognition using GSM Data. Department of Computer Science. University of Bristol, Bristol (2006)

Bao, L., Intille, S.: Activity recognition from user-annotated acceleration data. In: Ferscha, A., Mattern, F. (eds.) Pervasive Computing, Lecture Notes in Computer Science, vol 3001, pp. 1–17. Springer Berlin Heidelberg (2004). doi:10.1007/978-3-540-24646-6_1

Ben-Hur, A., Weston, J.: A user's guide to support vector machines. Methods mol. biol. (Clifton, NJ) **609**, 223–239 (2010). doi:10.1007/978-1-60327-241-4_13

Besag, J.E.: Nearest-neighbour systems and the auto-logistic model for binary data. J. R. Stat. Soc. Ser. B (Methodol.) **34**, 75–83 (1972)

Bohte, W., Maat, K.: Deriving and validating trip purposes and travel modes for multi-day GPS-based travel surveys: a large-scale application in the Netherlands. Transp. Res. Part C **17**(3), 285–297 (2009). doi:10.1016/j.trc.2008.11.004

Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. Paper presented at the proceedings of the fifth annual workshop on computational learning theory, Pittsburgh, Pennsylvania, USA, (1992)

Breiman, L.: Random Forests. Mach. Learn. **45**(1), 5–32 (2001). doi:10.1023/A:1010933404324

Breiman, L., Freidman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. The Wadsworth Statistics/Probability Series. Wadsworth International Group, Belmont (1984)

Caruana, R.: Niculescu-Mizil A An empirical comparison of supervised learning algorithms. In: Proceedings of the 23rd international conference on machine learning, ACM, pp 161–168 (2006)

Chang, C–.C., Lin, C.-J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 1–27 (2011). doi:10.1145/1961189.1961199

Chen, C., Gong, H., Lawson, C., Bialostozky, E.: Evaluating the feasibility of a passive travel survey collection in a complex urban environment: lessons learned from the New York City case study. Trans. Res. Part A **44**(10), 830–840 (2010)

Chung, E.-H., Shalaby, A.: A Trip Reconstruction Tool for GPS-based Personal Travel Surveys. Transp. Plan. Technol. **28**(5), 381–401 (2005). doi:10.1080/03081060500322599

Ettema, D.F., Timmermans, H.J.P., Van Veghel, L.: Effects of Data Collection Methods in Travel and Activity Research. European Institute of Retailing and Service Studies, Washington (1996)

Feng, T., Timmermans, H.J.: Transportation mode recognition using GPS and accelerometer data. Transp. Res. Part C **37**, 118–130 (2013)

Figo, D., Diniz, P.C., Ferreira, D.R., Cardoso, J.M.: Preprocessing techniques for context recognition from accelerometer data. Pers. Ubiquitous Comput. **14**(7), 645–662 (2010). doi:10.1007/s00779-010-0293-9

Forrest, T.: Pearson D (2005) Comparison of Trip Determination Methods in Household Travel Surveys Enhanced by a Global Positioning System. Trans. Res. Rec. **1**, 63–71 (1917). doi:10.3141/1917-08

Freund, Y., Schapire, R.E.: A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. J. Comput. Syst. Sci. **55**(1), 119–139 (1997). doi:10.1006/jcss.1997.1504

Gong, H., Chen, C., Bialostozky, E., Lawson, C.T.: A GPS/GIS method for travel mode detection in New York City. Comput. Environ. Urban Syst. **36**(2), 131–139 (2012). doi:10.1016/j.compenvurbsys.2011.05.003

Hato, E.: Development of behavioral context addressable loggers in the shell for travel-activity analysis. Transp. Res. Part C **18**(1), 55–67 (2010). doi:10.1016/j.trc.2009.04.013

Hemminki, S., Nurmi, P., Tarkoma, S.: Accelerometer-based transportation mode detection on smartphones. In: Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems, 2013. ACM, p 13

Lester, J., Choudhury, T., Borriello, G.: A Practical approach to recognizing physical activities. In: Fishkin, K., Schiele, B., Nixon, P., Quigley. A. (eds.) Pervasive Computing, Lecture Notes in Computer Science, vol 3968, pp. 1–16. Springer Berlin, Heidelberg (2006). doi:10.1007/11748625_1

Lewis, D.D.: Naive (Bayes) at forty: The independence assumption in information retrieval. In: Machine Learning: ECML-98. Springer, pp 4–15 (1998)

Mitchell, T.M.: Artificial neural networks. Machine Learning, pp. 81–127 (1997)

Mun, M., Estrin, D., Burke, J., Hansen, M.: Parsimonious mobility classification using GSM and WiFi traces. Paper presented at the 5th workshop on embedded networked sensors, (2008)

Nham, B., Siangliulue, K., Yeung, S.: Predicting mode of transport from iphone accelerometer data, Machine Learning Final Projects. Stanford University, California (2008)

Nick, T., Coersmeier, E., Geldmacher, J., Goetze, J.: Classifying means of transportation using mobile sensor data. In: Neural Networks (IJCNN), The 2010 International Joint Conference on, pp. 1–6, 18–23 July (2010). doi:10.1109/IJCNN.2010.5596549

Nitsche, P., Widhalm, P., Breuss, S., Maurer, P.: A strategy on how to utilize smartphones for automatically reconstructing trips in travel surveys. Procedia-Soc. Behav. Sci. **48**, 1033–1046 (2012)

Ohmori, N., Nakazato, M., Harata, N.: GPS mobile phone based activity diary survey. In: Proceedings of the Eastern Asia Society for Transportation Studies, pp. 1104–1115 (2005)

Reddy, S., Mun, M., Burke, J., Estrin, D., Hansen, M., Srivastava, M.: Using mobile phones to determine transportation modes. ACM Trans. Sen. Netw. **6**(2), 1–27 (2010). doi:10.1145/1689239.1689243

Safavian, S.R., Landgrebe, D.: A survey of decision tree classifier methodology. IEEE Trans. Syst. Man Cybern. **21**(3), 660–674 (1991)

Schönfelder, S., Axhausen, K.W., Antille, N., Bierlaire, M., Axhausen, K.W., Axhausen, K.W., Bierlaire, M., Bierlaire, M.: Exploring the Potentials of Automatically Collected GPS Data for Travel Behaviour Analysis: A Swedish Data Source. ETH, Eidgenössische Technische Hochschule Zürich, Institut für Verkehrsplanung, Transporttechnik, Strassen-und Eisenbahnbau IVT (2002)

Schuessler, N., Axhausen, K.: Processing Raw Data from Global Positioning Systems Without Additional Information. Transp. Res. Rec. **2105**(1), 28–36 (2009). doi:10.3141/2105-04

Shawe-Taylor, J., Cristianini, N.: Kernel methods for pattern analysis. Cambridge University Press, Cambridge (2004)

Sohn,T., Varshavsky, A., LaMarca, A., Chen, M., Choudhury, T., Smith, I., Consolvo, S., Hightower, J., Griswold, W., Lara, E.: Mobility detection using everyday GSM traces. In: Dourish, P., Friday, A. (eds.) UbiComp 2006: Ubiquitous Computing, Lecture Notes in Computer Science, vol 4206, pp. 212–224. Springer Berlin, Heidelberg, (2006). doi:10.1007/11853565_13

Stenneth, L., Wolfson, O., Yu, P.S., Xu, B.: Transportation mode detection using mobile phones and GIS information. Paper presented at the Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Chicago, Illinois, (2011)

Stopher, P., FitzGerald, C., Zhang, J.: Search for a global positioning system device to measure person travel. Transp. Res. Part C **16**(3), 350–369 (2008). doi:10.1016/j.trc.2007.10.002

Stopher, P.R.: Use of an activity-based diary to collect household travel data. Transportation **19**(2), 159–176 (1992). doi:10.1007/BF02132836

Tapia, E.M., Intille, S.S., Haskell, W., Larson, K., Wright, J., King, A., Friedman, R.: Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In: Wearable Computers, 2007 11th IEEE International Symposium on, pp. 37–40, 11–13 Oct 2007. doi:10.1109/ISWC.2007.4373774

Tsui, S.Y.A.: Shalaby AS (2006) Enhanced system for link and mode identification for personal travel surveys based on global positioning systems. Transp. Res. Rec. **1**, 38–45 (1972)

Wang, L.-X.: Adaptive fuzzy systems and control: design and stability analysis. Prentice-Hall, Inc., Upper Saddle River (1994)

Wolf, J., Guensler, R., Bachman, W.: Elimination of the travel diary: experiment to derive trip purpose from global positioning system travel data. Transp. Res. Rec. **1768**(1), 125–134 (2001)

Wolf, J., Oliveira, M., Thompson M.: Impact of Underreporting on Mileage and Travel Time Estimates: results from Global Positioning System-Enhanced Household Travel Survey. Transp. Res. Rec. **1854**(1), 189–198 (2003)

Yu, M.-C., Yu, T., Lin, C., Chang, E.: Low power and low cost sensor hub for transportation-mode detection. Studio Engineering, HTC (2013)

Muhammad Awais Shafique is currently a Ph.D. student in Transportation Research and Infrastructure Planning Laboratory at the University of Tokyo and an Assistant Professor in the department of Transportation Engineering and Management at the University of Engineering and Technology, Lahore. His research interests include application of machine learning and pattern recognition in transportation.

Eiji Hato is currently a Professor in Behavior in Networks Studies Unit at the University of Tokyo. He works in the area of travel behavior modelling, multi scale parallel simulation based on data assimilation and electric vehicle sharing services. He has received numerous awards including honorable mention in 2002 WCTR young prize from WCTRs and 2011 Kometani-Sasaki Award from ISSR.