



Travel mode identification using bluetooth technology

Shu Yang & Yao-Jan Wu

To cite this article: Shu Yang & Yao-Jan Wu (2018) Travel mode identification using bluetooth technology, Journal of Intelligent Transportation Systems, 22:5, 407-421, DOI: [10.1080/15472450.2017.1384698](https://doi.org/10.1080/15472450.2017.1384698)

To link to this article: <https://doi.org/10.1080/15472450.2017.1384698>



Accepted author version posted online: 27 Sep 2017.
Published online: 23 Oct 2017.



Submit your article to this journal [↗](#)



Article views: 215



View Crossmark data [↗](#)



Citing articles: 3 View citing articles [↗](#)



Travel mode identification using bluetooth technology

Shu Yang ^a and Yao-Jan Wu^b

^aPostdoctoral Scholar Center for Urban Transportation Research(CUTR), University of South Florida, 4202 E. Fowler Ave., CUT 100, Tampa, Florida,, USA; ^bDepartment of Civil Engineering and Engineering Mechanics, The University of Arizona, 1209 E 2nd St. Room 324F, Tucson, AZ, USA

ABSTRACT

Bluetooth technology has been widely used in transportation studies to collect traffic data. Bluetooth media access control (MAC) readers can be installed along roadways to collect Bluetooth-based data. This data is commonly used to measure traffic performance. One of the advantages of using Bluetooth technology to measure traffic performance is that travel time can be measured directly with a certain level of error instead of by estimation. However, travel time outliers can commonly be observed due to different travel mode on arterials. Since travel mode information cannot be directly obtained from the raw Bluetooth-based data, a mathematical methodology is in need to identify travel mode. In this study, a genetic algorithm and neural network (GANN)-based model was developed to identify travel mode. GPS-enabled devices were used to collect ground truth travel time. In order to additionally compare the model performance, K nearest neighbor (KNN) and support vector machine (SVM) were also implemented. N-fold cross validation was applied to statistically assess the models' results. Since the model performances depend on the model inputs, seven collections of model inputs were tested in order to achieve the best travel mode identification performance. An arterial segment with four consecutive links and three intersections was selected to be the study segment. The results suggested that correctly identifying the three travel modes successfully every time was not possible, although the GANN based model had low misidentification rates. In our study, 6.12% of autos were misidentified as bikes and 10.53% of bikes were misidentified as autos using three links.

KEYWORDS

bluetooth; genetic algorithm; neural network; short corridor; travel mode

Introduction

Bluetooth technology has been widely used in transportation studies to collect traffic data (e.g., travel time and partial Origin-destination data). Bluetooth media access control (MAC) readers can be installed along roadways to collect Bluetooth-based data. This data is commonly used to measure traffic performance (e.g., Araghi, Christensen, Krishnan, Olesen, & Lahrman, 2013; Barceló, Montero, Bullejos, Serch, & Carmona, 2013; Khoei, Bhaskar, & Chung, 2013; Qiao, Haghani, & Hamedi, 2013; Aliari & Haghani, 2012; Barceló, Montero, Marqués, & Carmona, 2010; Quayle, Koonce, DePencier, & Bullock, 2010; Haghani et al., 2010; and Wasson, Sturdevant, & Bullock, 2008). One of the advantages of using Bluetooth technology to measure traffic performance is that travel time, one of the most important traffic performance measures, can be measured directly instead of by estimation. It is worth noting that Bluetooth technology only can identify Bluetooth-enabled devices and cannot capture data from all traffic, and thus travel time calculations have to be rendered as estimates with a certain level of error. In recent years, the number of

personal Bluetooth devices (e.g., laptops, smart phones and smart watches) has grown significantly, enlarging the size of Bluetooth-based data samples (increased penetration rate). Therefore, travel time can be more accurately measured since the penetration rate has increased.

Bluetooth-based travel time accurately represents ground truth on long freeway segments in most circumstances (Haghani et al., 2009). However, in urban environments, more effort is required to use Bluetooth-based data to measure arterial travel time. This is because traffic on arterials is controlled by traffic signals, and therefore, traffic conditions are more complex than on freeways. Additionally, heterogeneous traffic is observed, and multiple travel modes travel simultaneously, including transit, bicyclists, and pedestrians. Several travel time outlier detection algorithms have been developed to clean Bluetooth-based data before use (e.g., Moghaddam, & Hellinga, 2013(a); Van Boxel, Schneider, & Bakula, 2011; Bachmann, Roorda, Abdulhai, & Moshiri, 2013; Porter, Kim, Magaña, Poocharoen, & Arriaga, 2013; Yang, Wu, Marion, & Moses, 2015). In the work as stated by Moghaddam, and Hellinga, 2013 (2013(a)), the authors

CONTACT Shu Yang  shuyang@usf.edu  Center for Urban Transportation Research(CUTR) University of South Florida 4202 E. Fowler Ave., CUT 100 Tampa, Florida 33620.

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/gits.

© 2017 Taylor & Francis

tested several outlier detection algorithms based on autos and buses. The work suggested that if travel modes can be identified, then outliers can be eliminated and Bluetooth-based travel time can be more precisely estimated. Therefore, knowing the travel mode of Bluetooth-based data would help practitioners and researchers develop mode-specific travel time outlier detection algorithms and accurately estimate mode-specific travel time.

Our study demonstrated herein is built on the previous scientific finding in terms of Bluetooth technology and applications of Bluetooth-based data, and then identify travel mode in the context of urban environment using Bluetooth-based data collected from the existing Bluetooth-based hardware and software infrastructure. The rest of the paper is organized as follows: first, relevant Bluetooth technology is overviewed. Next, a study segment in Tucson, AZ, and its corresponding dataset is presented. A genetic algorithm neural network (GANN) based model is introduced to identify travel modes using the Bluetooth-based data. The model performance is presented before drawing final conclusions.

Bluetooth technology overview

In contrast to conventional data sources (e.g., radar-based traffic data and GPS-based data) and their applications (e.g., Yang, An, Wu, & Xia, 2017a; Yang, An, Wu, & Xia, 2017b; Chen, Yang, Hu, & Wu, 2016; Yang & Wu, 2016; Yang et al., 2015a; Yang, Malik, & Wu, 2014), Bluetooth technology produces unique data sources and bring up with technical and application-wise concerns.

Privacy concerns

Bluetooth technology has been widely used for short-range wireless communication. For example, data can be shared between two Bluetooth-enabled devices, and certain devices (e.g., smart phones) can be remotely controlled by other devices (e.g., smart watches) via Bluetooth connections. These wireless operations require a shared or controlled agreement between the two devices (Bluetooth.com, 2016). These agreements are typically unnecessary in transportation studies, because: 1) only the Bluetooth signals broadcasted from devices that have Bluetooth turned on are detected (Bhaskar & Chung, 2013). The detected signals are usually encrypted before analysis. 2) No communication between the detecting and detected devices can be established since no agreement is initialized. 3) The MAC addresses of the detected devices are used anonymously (e.g., Araghi et al., 2013) and are not connected to specific individuals. Therefore, the privacy concerns regarding detected Bluetooth-enabled devices are not an issue.

Bluetooth-based data

Although Bluetooth-based data can be collected by many wireless communication protocols, such as WiFi, Zig-Bee (Zigbee.org, 2016) and radio-frequency identification (Technovelgy.com, 2016), three primary types of Bluetooth-based data are used in transportation studies:

- Media access control (MAC) addresses: every electronic device with a Bluetooth module built in has a global unique identifier. Travel times are usually estimated by matching identical MAC addresses detected at upstream and downstream MAC readers.
- Timestamp: the time of detection for the Bluetooth-enabled device.
- Location identifier: the location where the Bluetooth MAC reader is installed.

The Bluetooth received signal strength indication (RSSI) can also be used. The RSSI could be recorded depending on the functionality and configuration of the Bluetooth MAC readers. Bluetooth-enabled devices may be detected multiple times by a MAC reader. The increasing RSSI values indicate that Bluetooth-enabled devices are closer to MAC readers. Therefore, travel times can be more accurately estimated by knowing the upstream and downstream timestamps with the greatest RSSI value. A few studies have used the RSSI in traffic studies to estimate or predict travel time (see, for example, Araghi et al., 2013; and Saedi, Park, Kim, & Porter, 2013). These authors concluded that estimating travel times using RSSI values may be a better representation of ground truth travel time.

Detection range

Bluetooth MAC readers detect Bluetooth-enabled devices within a certain range. Several factors influence detection range, including the types and power gains of Bluetooth antennas and the antenna installation position. Additionally, previous studies have also shown that the Bluetooth signal strength inside cars may be half of the normal range due to the vehicle's metal body (Quayle et al., 2010), resulting in a smaller detection range. Therefore, detection range depends on both hardware and travel mode. This characteristic could be helpful in travel mode determination.

Multiple detections

Bluetooth-enabled devices may be detected multiple times within a particular detection range. However, the specific locations of the devices remain unknown and the location identifier is the only known spatial information. For example, consider a Bluetooth MAC reader (named R) installed at an intersection. The information collected

by R includes the MAC addresses of the detected devices, multiple timestamps for each device due to repeat detections, and the location identifier of R. The timespan of repeat detections is also called duration. A few studies have tried to explore the value of duration at intersections and establish the relationship between the duration and traffic congestion (Tsubota, Bhaskar, Chung, & Billet, 2011). However, the relationship remains largely undefined.

Limitations on bluetooth-based data applications

Most existing Bluetooth-based data applications are focused on two areas: 1) travel time estimation and prediction (e.g., Araghi et al., 2013; Khoei et al., 2013; Qiao et al., 2013; Aliari & Haghani, 2012; Quayle et al., 2010; Haghani et al., 2009; and Wasson et al., 2008). 2) Origin-destination matrix estimation (e.g., Barceló et al., 2013; and Barceló et al., 2010). Several researchers have conducted work zone analysis (e.g., Haseman, Wasson, & Bullock, 2010) or route choice analysis (e.g., Hainen et al., 2011) based on either estimated or predicted Bluetooth travel time. Few studies have used Bluetooth-based data to study bike travel time (Mei, Wang, & Chen, 2012). A recent study by Araghi, Krishnan, and Lahrmann (2016) has used Bluetooth-based data to estimate mode-specific travel time. The Bluetooth MAC readers have been deployed not only in the setting of urban environment but also on freeways. The locations of these readers serve as the foundation of filtering gates of travel modes. In addition, two filtrations, including travel time and Class of Device (CoD) filtration, primarily compose the mode-specific travel time estimation approach. Due to the limitation of funding resources or transportation authorities' jurisdiction, the deployment of Bluetooth MAC readers may be limited to a certain region. In order to address the issue of identifying travel mode, location-free approaches may provide a more general solution. Unlike other traffic data sources (e.g., loop sensors or GPS), which have been applied to various subjects, Bluetooth-based data applications have been limited. Therefore, Bluetooth-based data has been considered as complementary transport data (Bhaskar & Chung, 2013). Errors, mainly caused by detection range, multiple detections, and various travel modes, are common. For example, the accuracy of travel time measurement using Bluetooth-based data would be improved with the increase in the distance between two Bluetooth MAC readers (Haghani, Hamedi, Sadabadi, Young, & Tarnoff, 2010). In addition, Araghi, Olesen, Hammershøj, Krishnan, Tørholm Christensen, & Lahrmann (2015) used 1000 trips of both Bluetooth-based and GPS data to examine the reliability of travel time estimation using

Bluetooth-based technology. However, travel time estimation using the data from loop sensors is highly independent on sensor locations. Many mathematical models have been developed to estimate travel time using conventional traffic sensors (e.g., Yang et al., 2016). Since GPS-enabled devices send out geographical locations with a certain level of offset at a certain time interval (e.g., 5, 10, 30 seconds), the accuracy of GPS-based travel time would be primarily caused by both the accuracy of GPS locations and built-in time interval in GPS-enabled devices.

Because of the limited applications of Bluetooth data, Bluetooth-based data types are few in number, and three major types of traffic data are difficult to collect with Bluetooth technology:

- Traffic volume information and turning movements: previous studies have shown that only 2.0% to 3.4% of the total traffic volume is detected by the average Bluetooth system (Aliari & Haghani, 2012); therefore accurate traffic volume cannot be estimated.
- Lane-by-lane information is practically difficult using Bluetooth technology with omnidirectional antennas. Detailed lane-by-lane information is valuable for studying driving behavior, such as lane changing maneuvers. Lane-by-lane information is critical for locating vehicles in a connected vehicle environment. Additionally, Haghani et al. (2010) concluded that Bluetooth technology was not suitable for facilities with managed lanes. However, lane-by-lane information might be collected using directional antenna.
- Travel mode information. On freeways, autos are the primary travel mode, including passenger cars, motorcycles, trucks, etc. However, in an urban context, multiple travel modes, including autos, bikes, and pedestrians, are mixed and share the roadway. Since Bluetooth-enabled devices are independent of travel mode and device privacy is protected, travel mode information is unavailable. Our study seeks to enable mode identification using modeling methods.

Errors created by detection range and multiple detections are difficult to correct. Moghaddam and Hellinga (2013b) categorized the characteristics of Bluetooth measurement errors into three types: sampling error, sampling bias, and measurement error. These errors were mainly caused by detection range, multiple detections, and different travel modes. For example, without knowing the travel mode of detected devices, estimated travel time could be biased. Slower modes, such as pedestrians, could result in a longer estimated travel time. A recent study quantified the relationship between estimated speed errors and arterial segment length (Haghani et al., 2010). The authors concluded that estimated speed errors increased with decreasing arterial segment length. However, few

studies have developed models to mathematically correct these errors.

Bluetooth data and study segment

Bluetooth collection system in Tucson, Arizona

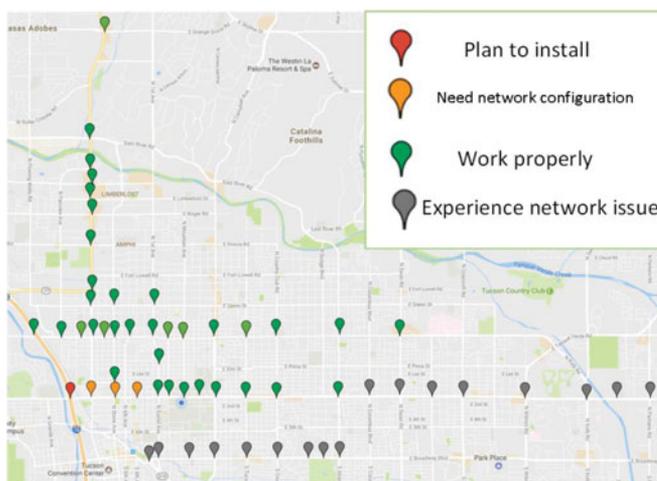
With the help of the City of Tucson, the Pima Association of Governments (PAG), and the Arizona Department of Transportation (ADOT), a Bluetooth-based data collection system has been developed and maintained in the Tucson, Arizona, area since 2013. Figure 1(a) shows the locations of Bluetooth MAC readers in the Tucson area. These MAC readers were installed inside traffic control cabinets located at major intersections. Instead of using commercial MAC readers, custom MAC readers with 9 dbi omnidirectional antennas, Bluetooth external adapters, and mini PCs were created. The Bluetooth channel scan time interval was programmed to be 3.84 seconds instead of the default value of 10.24 seconds in order to more precisely record entering and exiting times for each detector (Saeedi et al., 2013, p.92). After completing a Bluetooth scan time interval, the MAC readers sent the detected MAC addresses back to a central computer server located at the University of Arizona (UA) using user datagram protocol (UDP). Figure 1(b) shows the data collection system architecture.

Study segment and ground truth data collection

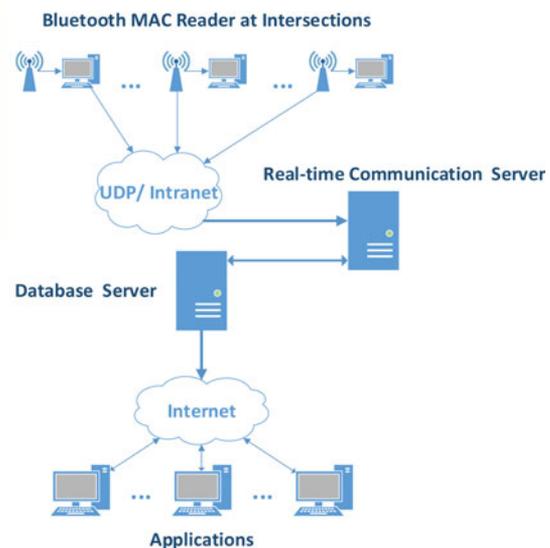
Speedway Boulevard is one of the busiest roadways in Tucson. Since UA is located next to Speedway and Tucson is a bicycle-friendly city, sidewalks and exclusive bike lanes had been built along the segment. Because of the high volume of multiple transportation modes along the

route, Speedway between Park Avenue and Campbell Avenue was chosen as the study segment. This segment included four intersections, three westbound links, and three eastbound links. Each intersection was configured with a custom MAC reader. To identify travel modes using Bluetooth-based data, ground truth data was collected. Several components of the data collection plan are shown below.

- 1) Three travel mode categories were classified, including autos (passenger cars, motorcycles, and trucks), bicyclists, and pedestrians. Our research team conducted a test on whether Bluetooth signals can be stably and reliably detected from inside transit buses, and the results showed it was negative. In addition, a preliminary study showed that Bluetooth signals from devices in transit buses were weak and could not be reliably detected by the MAC readers, possibly due to the metal body of transit buses (Quayle et al., 2010). Therefore, transit was not included in our study.
- 2) Table 1 shows the ground truth data collection plan. A trip was defined as either eastbound Speedway from Park to Campbell or westbound Speedway from Campbell to Park. The data was split into two parts: data used for travel mode identification model calibration (training data) and data used for verification (testing data).
- 3) The Bluetooth devices used for ground truth collection included two Blackberry cellphones, a Samsung cell phone, an iPhone, and an iPad. The GPS module for each device was also enabled to track the device's location every second. The GPS data was used only to estimate the MAC reader detection ranges.



(a) Bluetooth MAC reader locations



(b) Data collection system architecture

Figure 1. Bluetooth-based data collection system in Tucson, AZ, U.S.

Table 1. Ground truth data collection (GPS enabled during data collection).

Date	Mode		
	Autos Number of trips	Bike Number of trips	Pedestrian Number of trips
2015-08-31			4
2015-09-01		12	8
2015-09-04	18		8
2015-09-11	16		8
2015-09-16		12	8
2015-09-17	20*	12	
2015-09-18		12	
2015-09-28		14	
2015-09-29	20		
2015-10-01	20		
2015-10-05	20	14	
2015-10-06		14	8
2015-10-10			8
2015-10-12			8

*Five trips ended in non-through movements at the intersection with Campbell.

- As noted in Table 1, to account for real vehicle behavior, some trips ended in right, left, or U-turns at Campbell and Park, rather than continuing straight on Speedway beyond the study segment.

Detection range examination

Although previous studies have determined theoretical MAC reader detection ranges (e.g., 100–300 m (Araghi et al., 2013) or 300 ft. (Haghani et al., 2010)), few studies have physically examined actual MAC reader detection ranges. Our study used both GPS and Bluetooth-based data to examine the MAC reader detection ranges by matching the timestamps collected from both data. Figure 2 shows the detection ranges for each of the three travel modes at two intersections. The red areas in Figure 2 represent the regions where most of the tested devices were detected. Two findings were noted: 1) most of these detection ranges were less than 300 m (985 ft.) in our study; 2) the detection ranges varied depending on the intersection and travel mode. Note that the asymmetric detection ranges may be caused by trees and buildings along the Speedway Blvd.

Genetic algorithm and neural network-based mode identification

Travel mode identification can be seen as a classification issue (Araghi et al., 2016). Neural networks, the K-nearest neighbor (KNN) and the support vector machine (SVM) are three primary approaches to address classification issues (Qiao et al., 2013, p.166; Yang et al., 2016). Our study, therefore, used neural networks, KNN and SVM to identify travel mode using the data demonstrated in Table 1.

Justification for using a genetic algorithm to train a neural network

Neural networks are widely used for data classification and prediction because of their high accuracy. Three common types of neural networks include feed-forward, recurrent, and high-order. One of the most popular neural network structures is the single hidden layer feed-forward neural network (SHLFFNN). Many commercial and open source software implementations of the SHLFFNN can be found, such as the neural network toolbox in MATLAB and the “RANN” package in R language. SCHLFFNNs are composed of three layers in the following order: input, hidden, and output layers. Each layer contains one or more neurons. The neuron connections strictly follow these rules: 1) connections are only made between two consecutive layers, such as the input layer to the hidden layer; or the hidden layer to the output layer. 2) Neurons in a layer must fully connect to every single neuron in the consecutive layer. 3) During the neural network training procedure, only the connection weights can be updated.

With regards to the SHLFFNN, many previous studies have found that: 1) several factors may affect the accuracy and efficiency of training SHLFFNNs, including learning rate, number of iterations, and initial connection weights (Michalewicz, 1996; and Koehn, 1994). 2) The back-propagation algorithm (BP) (Rojas, 2013) is commonly used to train the SHLFFNNs. However, the connection weight “often gets trapped in a local minimum of the error function and is incapable of finding a global minimum” (Yao, 1999, p. 1425). 3) Optimal combinations of the three abovementioned training factors are typically found by trial-and-error, making this a time consuming experiment. 4) Not only the connection weights but also the topology of neural networks can be updated during the training procedure. Updating neural network topology can improve accuracy and find near-optimal solutions (Yao, 1996).

The genetic algorithm is a near-optimal search algorithm commonly used for solving problems that are difficult to solve using mathematical equations. Therefore, the Genetic Algorithm Neural Network (GANN) was used in our study to obtain more accurate results through changing connection weights, topology, and to avoid the time consuming trial-and-error approach, since 21 neural networks were trained. Details regarding these networks are provided in the following section.

Genetic algorithm and neural network (GANN)

Topology in the genetic algorithm and neural network

Essentially, the neural network used in our study was a SHLFFNN. The implementation details can be found on the authors’ website (Yang, 2015). Based on Yao’s (1996) suggestions, the topology of our neural network

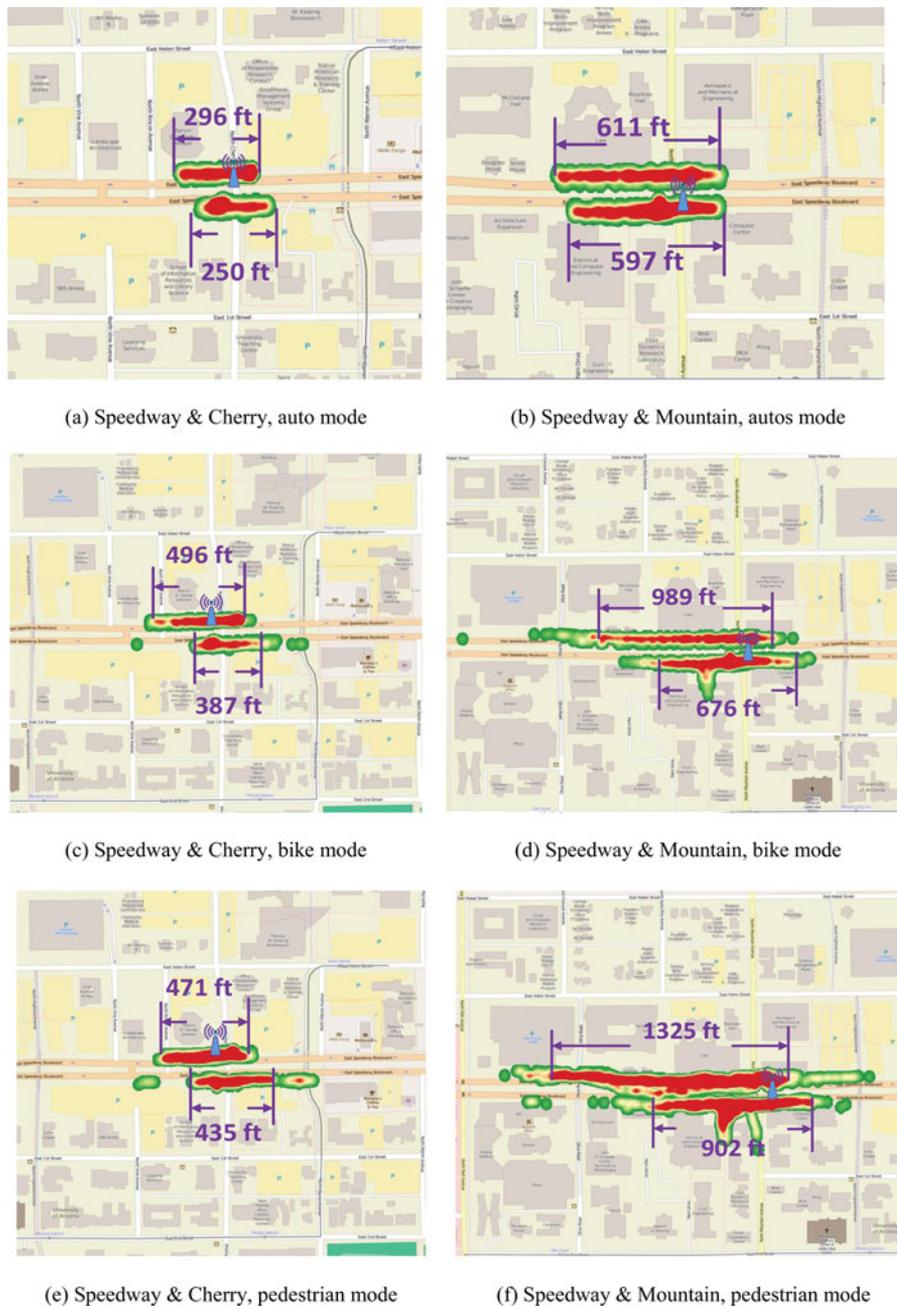


Figure 2. Detection range by three travel modes (background images from OpenStreetMap).

did not follow the strict rules listed in Section 4.1 and was organized in a more flexible manner: 1) the three layers were connected to each other and 2) connections between two neurons (connectivity) were not required to be established. Figure 3(a) shows an example of a typical SHLFFNN topology with two neurons of input (labeled as A and B), eight neurons of hidden layer (labeled as 1 – 8), and three neurons of output (labeled as I, II, and III). The neurons of input only connect towards the neurons of hidden layer, and thus 16 ($2 * 8$) connections have to be established between the input and hidden layer. Likewise, a fully 24 connections have to be established between the

hidden and output layer. Figure 3(b) shows an example of GANN topology with the same numbers of neurons in input, hidden, and output layers. The black links represent the connections in a typical SHLFFNN; and the blue links represent the connections from the input to the output layer. It is worth noting that the neuron labeled “8” in the hidden layer can be seen as unnecessary neuron, because the neuron does not forward the information from the neuron B to the output layer. In this example, seven neurons in the hidden layer may be enough to translate the information from the input layer to the output layer.

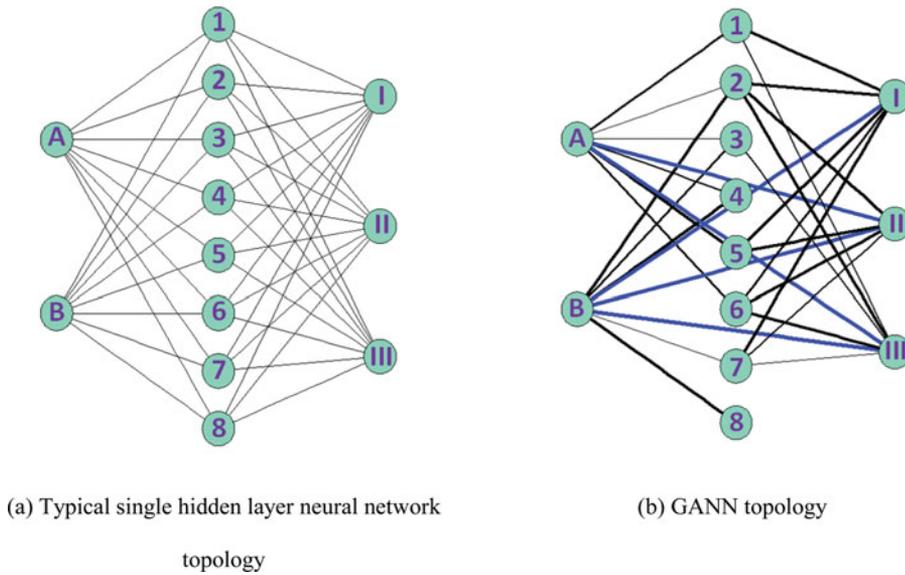


Figure 3. Feed-forward GANN model with single hidden layer.

Connection weights and connectivity encoding scheme in the genetic algorithm

Solutions to the connection weights and connectivity are usually coded as string-based schema in genetic algorithms. The string-based schema is designed to easily operate the crossover and mutation solutions. A sparse-matrix-based encoding scheme is proposed in order to update the connectivity and weights. Figure 4 shows the proposed scheme. The labels (i.e. A, B, 1–8, and I–III) in Figure 4 represent the corresponding neurons in Figure 3(b). Each cell in the sparse-matrix-based shows the connection between two neurons. For example, the cell “A→1” represents the connection information between neuron A and neuron 1. If no connections

between A and 1 are established, the value of the cell is set as zero; while, if the connection is established, the value of the cell would be non-zero, indicating that the two neurons are connected with the non-zero connection weight. An instance of a sparse matrix with connection information (zeros and non-zeros) is a candidate optimal solution for a GANN. The crossover and mutation operations are executed on instanced sparse matrixes.

Crossover and mutation operations

The crossover and mutation operations in the proposed genetic algorithm are used to generate new candidate optimal solutions by switching and modifying the connection information at the same position in existing two

Connections from input to hidden layer	A -> 1	B -> 1								
	A -> 2	B -> 2								
	A -> 3	B -> 3								
	A -> 4	B -> 4								
	A -> 5	B -> 5								
	A -> 6	B -> 6								
	A -> 7	B -> 7								
	A -> 8	B -> 8								
		Connections from hidden to output layer								
A -> I	B -> I	1 -> I	2 -> I	3 -> I	4 -> I	5 -> I	6 -> I	7 -> I	8 -> I	
A -> II	B -> II	1 -> II	2 -> II	3 -> II	4 -> II	5 -> II	6 -> II	7 -> II	8 -> II	
A -> III	B -> III	1 -> III	2 -> III	3 -> III	4 -> III	5 -> III	6 -> III	7 -> III	8 -> III	

Figure 4. Connectivity encoding scheme: sparse-matrix-based encoding scheme (cells in black: connections in a typical SHLFFNN; cells in blue: connections from input layer to output layer).

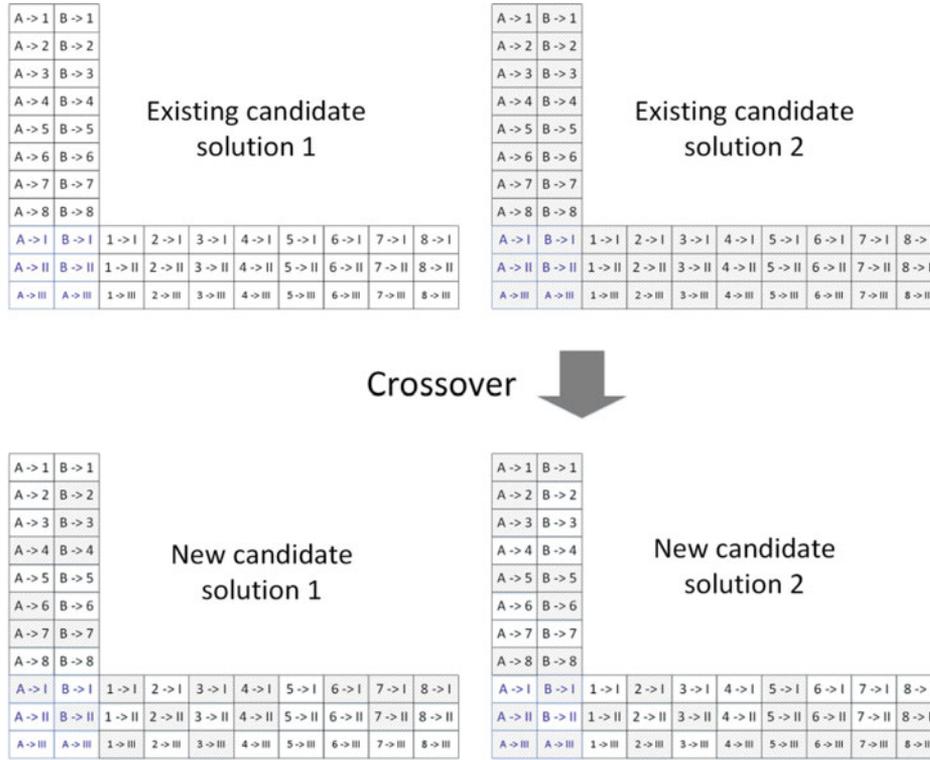


Figure 5. Crossover operation on two sparse matrixes.

sparse matrixes. Figure 5 shows that two new candidate solutions are produced by applying the crossover operation on two existing solutions. In order to distinguish the existing candidate solution 1 and 2, the cells in existing candidate solution 2 are designed with the upward diagonal pattern. The uniform crossover technique was used to implement the crossover operation. The mixing ratio was set to be 0.5 in our study, meaning that the corresponding cells in the two existing candidate solutions have 50% chance to switch and then produce new solutions.

The mutation operation is executed after switching corresponding cells. Since the connection information in a cell consists of connectivity and connection weight value, two sub-mutation operations are executed in order, including 1) modifying connection weight values by adding a random number sampled from a Gaussian distribution with (μ, σ) . The standard Gaussian distribution with $(\mu = 0, \sigma = 1)$ was used to generate random numbers; and 2) modifying connectivity by changing non-zero values to zeros with a probability γ . γ was set as 0.05 in our study, indicating that established connections between neurons had 5% chance to be disconnected.

Error calculation in GANN

The traditional SHLFFNN uses the BP algorithm to minimize output error. Due to the requirements of the BP algorithm, the output error is calculated based on a single input and is reduced back to the connection

weights. Equation 1 shows the error calculation if the activation function is based on a binary function, while Equation 2 shows the error calculation if the activation function is based on the Sigmoid function. Equation 3, unlike Equations 1 and 2 which only consider the error of a single input, calculates the error in GANN-based models by summing up the errors of all inputs (Yao, 1996, p. 1425). Equation 3 also serves as the fitness function in the genetic algorithm. The objective of the genetic algorithm is to minimize the error calculated from the inputs.

$$Error\ r_{O_i} = Calculated_{O_i} - Target_{O_i} \quad (1)$$

$$Error\ r_{O_i} = (Calculated_{O_i})(1 - Calculated_{O_i}) \times (Calculated_{O_i} - Target_{O_i}). \quad (2)$$

$$Error = \frac{\sum_{n=1}^N \sum_{m=1}^M (Calculated_{O_i} - Target_{O_i})^2}{N} \quad (3)$$

Where: $Error\ r_{O_i}$ represents the error of the i_{th} neuron in the output layer;

$Calculated_{O_i}$ is the calculated value of the i_{th} neuron in the output layer;

$Target_{O_i}$ is the ground truth value of the i_{th} neuron in the output layer;

N is the number of training data samples; and

M is the number of classifications. In our study $M = 3$.

Figure 6 shows the genetic algorithm flowchart used to train the sparse-matrix encoding schema neural network. In addition to the aforementioned uniform

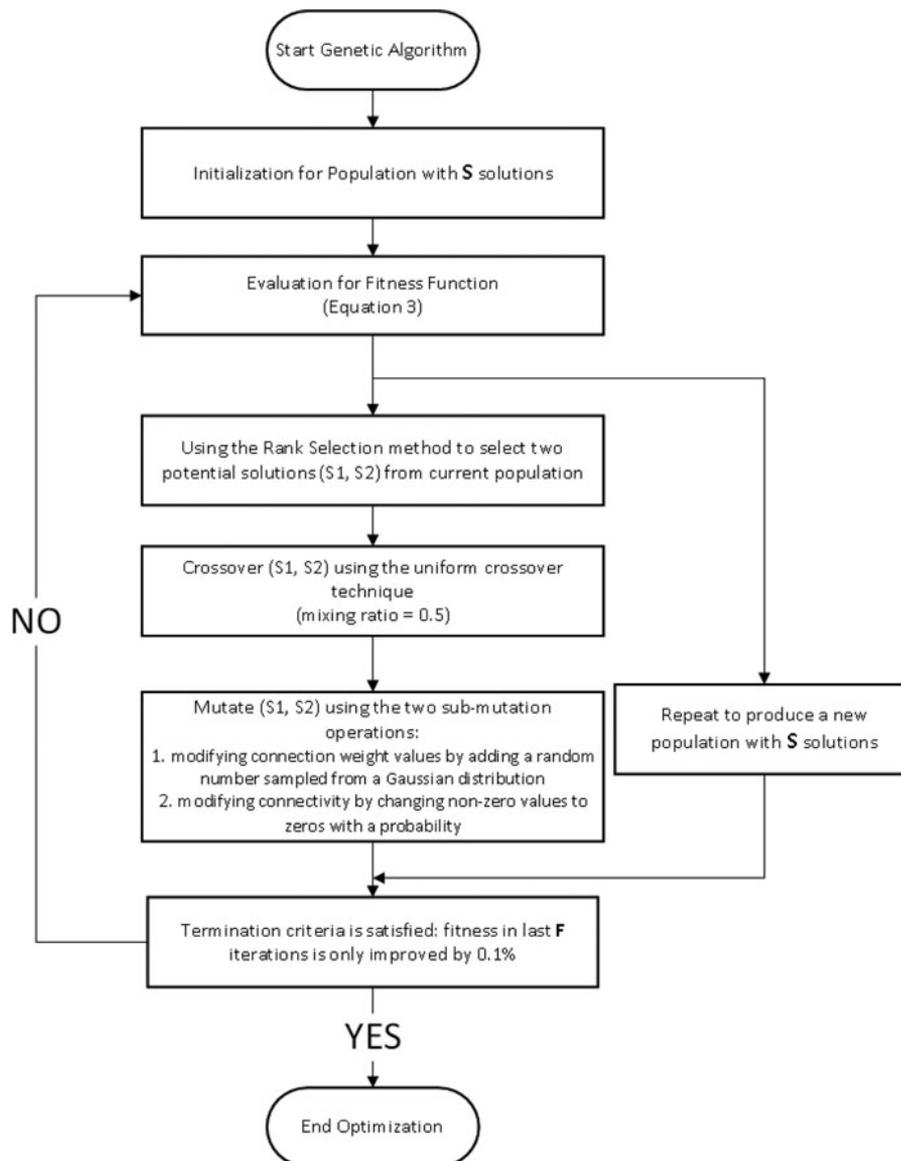


Figure 6. Genetic algorithm flowchart.

crossover technique and two sub-mutation operations, a termination criterion was set as: the GANN model stopped when the best fitness in the last F iterations was only improved by 0.1%. F in our setting was selected to be 100.

Population initialization and computing environment

Genetic algorithms are inherently compatible with parallel computing because each solution in a population can be individually evaluated. Our study used five desktops with Intel(R) Core(TM) i7-4770 central processing units (CPU). Each CPU is configured to be eight logic cores. In order to ensure the five desktops unfrozen during high tension computing, seven logic cores in a desktop were used. 35 cores in total, therefore, were setup to perform parallel computing. In order to leverage the computing power to maximum usage, the population consists of 70

candidate solutions. The first-generation solutions were initialized by assigning random numbers ranging from -1 to 1. Each GANN-based model was well trained within four minutes using the five internet-connected desktops.

K-Nearest neighbor and support vector machine models

In addition to the proposed GANN model for travel mode identification, the K-nearest neighbor (KNN) and the support vector machine (SVM) were used in our study for the comparison purpose. N-fold cross validation was used to assess these models' performance.

K-nearest neighbor

The K-nearest neighbor is a non-parametric approach designed for classification and regression (Altman, 1992).

When KNN is applied for classification, the attributes associated with an object are determined by k nearest neighbor objects. Various distance measures are available to search for k nearest neighbors given an object. Euclidean distance was used and K was experimentally set as 5 in our study after testing the range from 2 to 10 for the selection of K .

Support vector machine

Support vector machine has been commonly used for classification and image recognition (Cortes & Vapnik, 1995). SVM could be seen as a high dimension version of linear classification on an X-Y plane. In linear classification, data can be separated by a line on a two dimension plane. Instead of using a line, hyperplanes are applied to handle high dimension data. Many hyperplanes may be available to classify data; an optimal hyperplane is the one that classifies data with largest separation. The standard statistical package R provides the packages for KNN and SVM models, and the relevant R built-in functions were used in our study to classify travel mode. The Gaussian distributions are selected as the standard kernel function in the SVM model.

N-fold cross validation

Cross validation (Geisser, 1993) has been widely used to statistically assess models' results. Several strategies are available to implement cross validation, such as leave-p-out and N-fold. N-fold cross validation was used to validate the three aforementioned models in our study. The N-fold cross validation primarily consisted of two steps: 1) N partitions with equal data size were randomly generated; and 2) one out of N subsamples was kept for validation and the rest of $N-1$ subsamples were used to train the three models. Since N-fold was applied, the two steps were repeated N times. The average misidentification error and

the associated standard error can be calculated every step. The final statistical assessment of misidentification errors can be calculated using Equations 3 and 4. N was set to be 20 in our study.

$$\bar{e} = \frac{1}{N} \sum_{n=1}^N e_i \quad (3)$$

$$\text{Standard error} = \sqrt{\frac{\sum_{n=1}^N (e_i - \bar{e})^2}{N(N-1)}} \quad (4)$$

Where: e_i is the i th misidentification error.

Input selection

Traffic performance measures, such as travel time and speed, can help identify travel modes. One of the measures obtained from Bluetooth-based data is travel time, which can then be used as the primary performance measure to identify travel modes. For example, autos usually travel faster than both bikes and pedestrians on arterials. However, travel time is dependent on the specific link, and therefore, a single model is required to identify travel mode on that link. Speed (normalized travel time divided by link length) is an alternative measure that can be used to develop link-independent mode identification models.

First-to-first (FF) and last-to-last (LL) travel time were favored in previous studies (for example, Araghi et al., 2013 and Saeedi et al., 2013). Since the RSSI is unavailable, peak-to-peak travel time (Araghi et al., 2013) was not calculated in our study. Based on the detection range determined in Section 3.3, the FF and LL distances for the three travel modes are listed in Table 2. Then, the FF and LL speeds were calculated. In order to examine whether the detection ranges could significantly affect travel mode identification, a baseline speed was also calculated using the intersection-to-intersection distances.

Table 2. Measured distance by mode.

From	To*	# of Links	Autos Mode (ft)		Bike Mode (ft)		Pedestrian Mode (ft)		Intersection-Intersection (ft)
			FF	LL	FF	LL	FF	LL	
1	2	One	1526.0	1639.1	1346.3	1456.5	1289.1	1575.1	1602.7
1	3	Two	3234.3	3110.2	3307.3	3037.3	3256.2	3015.3	3181.8
1	4	Three	4665.6	4804.3	4767.7	4753.0	4694.5	4536.0	4665.6
2	1	One	1474.9	1398.3	1467.6	941.9	1598.9	798.0	1602.7
2	3	One	1781.5	1522.2	1985.9	1584.3	2026.1	1596.0	1579.1
2	4	Two	3205.6	3169.1	3431.9	3285.8	3548.4	3168.8	3062.9
3	1	Two	3095.6	3132.1	2956.9	3095.6	2869.3	3183.2	3181.8
3	2	One	1606.2	1850.8	1482.2	2161.1	1361.6	2445.9	1579.1
3	4	One	1457.0	1599.3	1467.8	1752.5	1372.7	1566.3	1483.8
4	1	Three	4607.0	4650.8	4468.2	4409.8	4431.8	4329.5	1483.8
4	2	Two	3110.5	3380.7	3051.9	3548.3	3052.1	3687.0	3062.9
4	3	One	1533.3	1486.0	1551.1	1387.4	1586.0	1241.2	1483.8

*1: Speedway Blvd. & Park Ave.

2: Speedway Blvd. & Mountain Ave.

3: Speedway Blvd. & Cherry Ave.

4: Speedway Blvd. & Campbell Ave.

Table 3. Input selection candidates scenarios.

Scenario Name	Input	
	Input Parameters	Number of Inputs
Scenario 1	FF speeds using measured distances for the three travel modes	3
Scenario 2	FF speed using intersection-to-intersection length	1
Scenario 3	LL speeds by the three travel modes	3
Scenario 4	LL speed using intersection-to-intersection length	1
Scenario 5	Using both FF and LL speeds by the three travel modes	6
Scenario 6	Using both FF and LL speeds based on intersection-to-intersection length	2
Scenario 7	Adding duration data to the best-performing scenario among Scenarios 1–6	Determined based on Scenario 1–6 results

Model performances could be significantly different depending on inputs. Seven scenarios designed to identify the best collection of inputs are presented in Table 3. Scenarios 1 and 2 used FF speeds. Scenario 1 was based on measured distances between detection ranges, while Scenario 2 was based on the intersection-to-intersection length. Scenario 2 could be used in situations where measured distances were unknown or not available. Similarly, Scenarios 3 and 4 used LL speeds, and Scenarios 5 and 6 used both FF and LL speeds. Scenario 7 was designed based on the assumption that the travel mode identification accuracy is improved by adding duration data, which is determined as a result of multiple detections. The duration data at upstream and downstream locations was included in Scenario 7, while the other input parameters depended on the best-performing scenario among Scenarios 1–6.

To investigate the effect of the number of links on travel mode identification, data from a single link, two links, and three links was collected and used in each scenario. Since KNN, SVM, and GANN were used, a total of 63 models were developed and tested, including 21 KNN-based models, 21 SVM-based models, and 21 GANN-based models. Note that real-value inputs in the GANN-based models were scaled between zero and one for computing convenience.

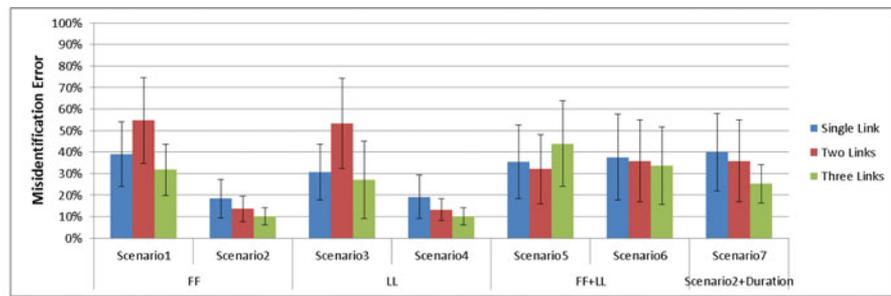
Model performance and comparisons

Best input selection for KNN, SVM, and GANN models

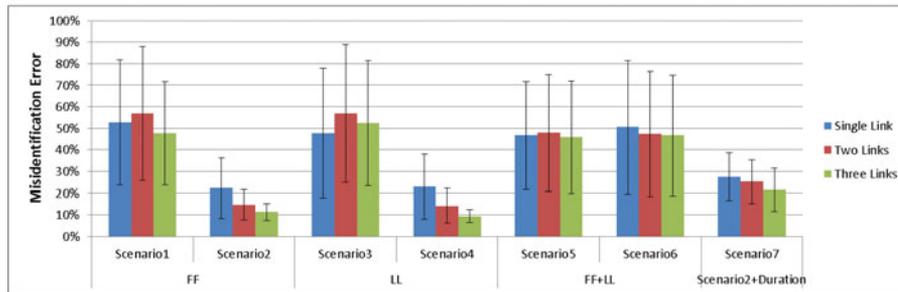
Different combinations of inputs and models would lead to significantly different travel mode identification results. Additionally, the best input for the KNN models may not be the best input for the GANN models. The misidentification rate and the corresponding standard errors resulted from the cross-validation technique were selected to measure the model performance. Figure 7 shows the misidentification rate and its corresponding standard errors for the 63 models. Overall, the GANN-based models outperformed KNN-based and SVM-based models with significant accuracy improvement of travel mode identification, and the training errors decreased

with an increasing number of links regardless of the model selection. The best inputs for KNN, SVM, and GANN models (regardless of number of links) and their impacts on model performances are listed below:

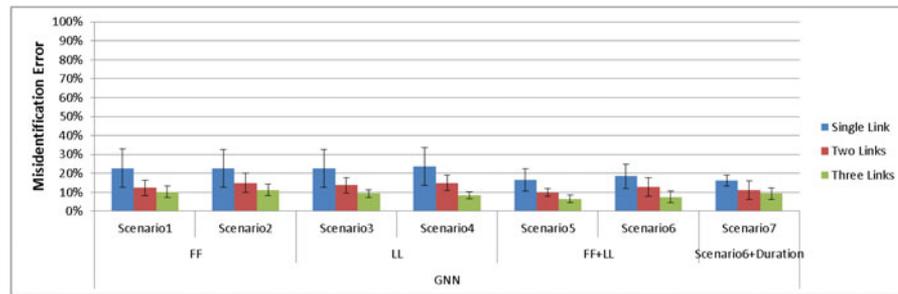
1. Best inputs for KNN and SVM models
 - a. Given a fixed segment (regardless of the number of links), the model performance of using either FF or LL speeds alone was better than using both FF and LL speeds. Since the misidentification errors of using FF speed alone were slightly lower than that using LL speed, Scenario 2 (FF speed using intersection-to-intersection length) was considered as the best input for the KNN and SVM models.
 - b. Scenario 7 was therefore designed to jointly use Scenario 2 and the upstream and downstream durations. Figures 7(a) and (b) shows that the misidentification errors with different number of links using Scenario 2 input were lower than that using Scenario 7 input.
 - c. Detection ranges by travel mode had significant impacts on travel mode identification. For example, in Figure 7(a), the misidentification errors using either Scenarios 1 or 3 input ranged from approximately 30% to 54%; while, the errors using Scenarios 2 or 4 input were below 20%.
 - d. One of the summative findings from (1.b) and (1.c) was that adding more information (i.e. detection ranges by travel mode and the durations) may not improve the accuracy of travel mode identification using either KNN or SVM models. Specifically, the model performances using Scenario 7 were even worse than those using single speed alone.
2. Best input for GANN models
 - a. Given a fixed segment (regardless of the number of links), using both FF and LL speeds was better than using either speed type alone. The GANN-based models in Scenario 5 and Scenario 6 outperformed the models in Scenarios 1–4. Considering the minor performance differences between Scenario 5 and Scenario 6 and overall model complexity, Scenario 6 was identified as the best input because the number of input parameters (FF and



(a) Peer Comparisons among 21 KNN Models



(b) Peer Comparisons among 21 SVM Models



(c) Peer Comparisons among 21 GANN-based Models

Figure 7. Peer comparisons.

- LL speeds) was less than that in Scenario 5 (FF and LL speeds by the three travel modes).
- Scenario 7 was therefore designed to jointly use Scenario 5 and the duration information. Figure 7(c) shows that the misidentification errors using Scenario 7 input were 16.21% (single link), 11.11% (two links), and 9.35% (three links); while, the corresponding errors using Scenario 5 were 16.48%, 9.83%, and 6.54%. Overall, the model performances of accuracy using Scenario 5 were slightly better than that using Scenario 7.
 - Detection ranges by travel mode had limited impacts on travel mode identification in Figure 7(c). For example, the misidentification errors using Scenarios 1 and 3 inputs were 22.8% and 22.52 (single link), 12.39% and 13.68 (two links), and 10.28% and 9.35% (three links). However, the corresponding errors using Scenarios 2 and 4 inputs were 22.53% and 23.63%

(single link), 14.96% and 14.96% (two links), and 11.21% and 8.41 (three links). The corresponding differences of misidentification errors were approximately $\pm 1\%$.

- One of the summative findings from (2.b) and (2.c) was that adding information to the GANN-based models had fairly limited impacts on travel mode identification. This conclusion may imply that the GANN-based models were less sensitive than KNN-based and SVM-based models.

Model performances with best inputs

The best inputs were identified for the KNN, SVM, and GANN models in Section 5.1. This section will take a closer look at the model performance through demonstrating misidentification errors by travel mode. Figure 8 quantifies the three model performances by showing the misidentification errors by three travel modes. Several

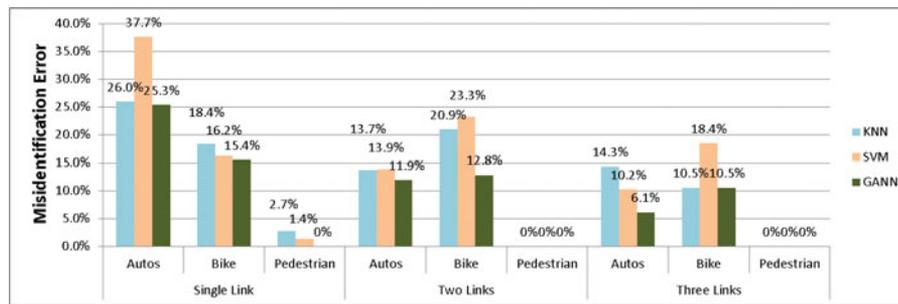


Figure 8. Best scenario performance differences of KNN, SVM, and GANN note: due to the graphical canvas limitation, the misidentification errors are rounded with one digit after decimal point.

findings are summarized below: Overall, the GANN-based model outperformed both the KNN-based and SVM-based models. For example, in the three links case, 6.12% of autos were misidentified as bikes and 10.53% of bikes were misidentified as autos using the GANN-based model, while the corresponding misidentification rates were 14.29% and 10.53% using the KNN-based model, and 10.20% and 18.42% using the SVM-based model. In the single link case, the KNN-based and SVM-based models could not adequately distinguish the pedestrian mode from other modes. Intuitively, pedestrians should be easily distinguishable due to their low speed (approximately 1.5–3 mph). The GANN-based model successfully identified pedestrians with a 0% misidentification rate. In addition, the KNN failed to identify pedestrians in many cases.

Discussion

Similar traffic performance measures on short segments

Due to the impacts of signalized intersections, bike speeds are very likely to be overlapped with auto speeds, especially on a short segment, indicating that travel times by bike and auto (travel time + waiting time at intersections) on a signalized segment are similar. The inherent difficulties of travel mode identification may explain this effect. The speeds of autos, bikes, and pedestrians on Speedway were approximately 35 mph, 12 mph, and 2.5 mph, respectively. Intuitively, they should be distinguishable. However, several factors can significantly affect speed estimation using Bluetooth-based data:

1) Short segment length: Haghani et al. (2010) showed that estimated speed errors were approximately 4.5 mph for an arterial link length of 0.5 miles and speed limit of 30 mph. However, the link length in our study site was 0.32 miles, indicating that the estimated speed errors may have been greater than 4.5 mph. These speed errors could have resulted in speeds of different

travel modes overlapping. However, our study also proved that the misidentification rates in the three-link segment were lower than those in the single-link segment, suggesting that speeds were more accurately estimated in the three-link segment.

- 2) Poorly coordinated traffic signals: if two consecutive traffic signals are not well coordinated, an auto stopped at the first signal may have to stop again at the second signal. Low speed travel modes, such as bikes, would have enough time to catch up to the auto as it waited at the second signal. In this case, both the travel time and average speed of the auto and the low speed mode would be similar.
- 3) Traffic congestion: since bikes and pedestrians often travel on bike lanes and sidewalks, they are much less affected by vehicular traffic congestion. However, auto speeds would be lower due to the delay caused by traffic congestion. Therefore, average bike speeds were sometimes faster than auto speeds in our study segment.

Potential applications

Travel mode identification on arterials can assist estimation of mode-specific traffic performance measures (e.g., travel time and speed). If a Bluetooth-enabled device is identified as an auto during a relatively short time period, the data from this device could be used for further traffic measure estimation. Since Bluetooth-based data has been also widely used to estimate origin-destination (OD) matrix, the general OD matrix could be further developed to be mode-specific OD matrix. Additional modes (e.g., truck) could be considered to polish the overall picture regarding people and vehicle movement.

Conclusion

Bluetooth technology has been popular in transportation studies. Bluetooth MAC readers are often used to detect Bluetooth devices and store data. Many previous studies have utilized Bluetooth-based data to measure traffic performance. However, travel time on arterials may be

inaccurate because of mixed travel modes traveling at different speeds. Therefore, travel mode identification becomes necessary before further data processing. Our study proposed a genetic algorithm neural network (GANN) based model to identify travel modes on the study segment in Tucson, Arizona. Twenty-one groups of input candidates were tested. To calibrate and verify these GANN-based models, Bluetooth-based data with known travel modes were collected. The Bluetooth-based infrastructure on the study segment, which had been developed and maintained since 2013, facilitated the data collection.

Several important findings from our studies are summarized below:

- Using both First-to-First (FF) and Last-to-Last (LL) speed as inputs performed better than using FF or LL speed alone.
- The detection ranges of the travel modes had little impact on travel mode identification using the GANN model.
- The travel mode misidentification rate can be decreased by considering higher numbers of arterial links.
- Duration data may not improve the rate of successful travel mode identification using the GANN model.
- The GANN based model outperformed both the KNN and SVM models. Using the KNN, even pedestrians were sometimes misidentified as other modes.

Correctly identifying the three travel modes successfully every time was not possible, although the GANN based model had low misidentification rates. In our study, 6.12% of autos were misidentified as bikes and 10.53% of bikes were misidentified as autos using three links. The GANN-based travel mode identification model showed its potential to detect travel time outliers and further clean Bluetooth-based data. Future studies will focus on the following areas: first, development of the outlier detection algorithm based on the GANN model. Obtaining the percentages of bike and autos in reality could help further improve the GANN model; second, investigating contribution factors (e.g., intersection type, traffic volume, and vehicle speed) would help further improve the model.

Acknowledgments

The authors would like to thank the University of California Center on Economic Competitiveness in Transportation (UCCONNECT), City of Tucson and the Regional Transportation Authority (RTA) for funding support. Also, the authors would like to thank PAG and ADOT for the Bluetooth MAC reader installation. The authors are grateful for the technical support from Mr. Mike Hicks in the City of Tucson and Mr. Simon Ramos of ADOT. Paul Casertano from PAG also provides useful advice on the development of the proposed system. Special thanks go to Mr. Payton Cooke for editing assistance.

ORCID

Shu Yang  <http://orcid.org/0000-0002-6690-7081>

References

- Altman, N. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3), 175–185. doi:10.1080/00031305.1992.10475879 10.2307/2685209.
- Aliari, Y., & Haghani, A. (2012). Bluetooth sensor data and ground truth testing of reported travel times. *Transportation Research Record: Journal of the Transportation Research Board*, 2308, 167–172. doi:10.3141/2308-18.
- Araghi, B. N., Christensen, L. T., Krishnan, R., Olesen, J. H., & Lahrmann, H. (2013). Use of low-level sensor data to improve the accuracy of bluetooth-based travel time estimation. *Transportation Research Record: Journal of the Transportation Research Board*, 2338, 29–34. doi:10.3141/2338-04.
- Araghi, B. N., Olesen, H., Hammershøj, J., Krishnan, R., Tørholm Christensen, L., & Lahrmann, H. (2015). Reliability of bluetooth technology for travel time estimation. *Journal of Intelligent Transportation Systems*, 19(3), 240–255. doi:10.1080/15472450.2013.856727.
- Araghi, B. N., Krishnan, R., & Lahrmann, H. (2016). Mode-specific travel time estimation using bluetooth technology. *Journal of Intelligent Transportation Systems*, 20(3), 219–228. doi:10.1080/15472450.2015.1052906.
- Bachmann, C., Roorda, M. J., Abdulhai, B., & Moshiri, B. (2013). Fusing a Bluetooth traffic monitoring system with loop detector data for improved freeway traffic speed estimation. *Journal of Intelligent Transportation Systems*, 17(2), 152–164. doi:10.1080/15472450.2012.696449.
- Barceló, J., Montero, L., Bulles, M., Serch, O., & Carmona, C. (2013). A kalman filter approach for exploiting bluetooth traffic data when estimating time-dependent OD matrices. *Journal of Intelligent Transportation Systems*, 17(2), 123–141. doi:10.1080/15472450.2013.764793.
- Barceló, J., Montero, L., Marqués, L., & Carmona, C. (2010). Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring. *Transportation Research Record: Journal of the Transportation Research Board*, 2175, 19–27. doi:10.3141/2175-03.
- Bhaskar, A., & Chung, E. (2013). Fundamental understanding on the use of bluetooth scanner as a complementary transport data. *Transportation Research Part C: Emerging Technologies*, 37, 42–72. doi:10.1016/j.trc.2013.09.013.
- Bluetooth.com. (2016). How It Works? Retrieved January 31, 2017, from <https://www.bluetooth.com/what-is-bluetooth-technology/how-it-works>
- Chen, Y., Yang, S., Hu, M., & Wu, Y.-J. (2016). A reliability-based transit trip planning model under transit network uncertainty. *Public Transport*, 8(3), 1–20. doi:10.1007/s12469-016-0134-y.
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297. doi:10.1023/A:1022627411411 10.1007/BF00994018.
- Geisser, S. (1993). *Predictive inference* (Vol. 55). Boca Raton, Florida: CRC press.

- Haghani, A., Hamed, M., Sadabadi, K. F., Young, S., & Tarnoff, P. (2010). Data collection of freeway travel time ground truth with bluetooth sensors. *Transportation Research Record: Journal of the Transportation Research Board*, 2160, 60–68. doi:10.3141/2160-07.
- Hainen, A., Wasson, J., Hubbard, S., Remias, S., Farnsworth, G., & Bullock, D. (2011). Estimating Route Choice and Travel Time Reliability with Field Observations of Bluetooth Probe Vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, 2256, 43–50. doi:10.3141/2256-06.
- Haseaman, R., Wasson, J., & Bullock, D. (2010). Real-time measurement of travel time delay in work zones and evaluation metrics using bluetooth probe tracking. *Transportation Research Record: Journal of the Transportation Research Board*, 2169, 40–53. doi:10.3141/2169-05.
- Khoei, A. M., Bhaskar, A., & Chung, E. (2013). Travel time prediction on signalised urban arterials by applying sarima modelling on bluetooth data. In *36th Australasian transport research forum (ATRF) 2013*. Brisbane, QLD: Queensland University of Technology.
- Koehn, P. (1994). *Combining genetic algorithms and neural networks: The encoding problem*. The University of Tennessee. Retrieved from <http://homepages.inf.ed.ac.uk/pkoehn/publications/gann94.pdf>.
- Mei, Z., Wang, D., & Chen, J. (2012). Investigation with bluetooth sensors of bicycle travel time estimation on a short corridor. *International Journal of Distributed Sensor Networks*, 2012(1), 1–7.
- Michalewicz, Z. (1996). *Genetic algorithms + data structures = evolution programs* (3rd, rev.). Salmon Tower Building New York City: Springer.
- Moghaddam, S. S., & Hellinga, B. (2013a). Evaluating the performance of algorithms for the detection of travel time outliers. *Transportation Research Record: Journal of the Transportation Research Board*, 2338, 67–77. doi:10.3141/2338-08.
- Moghaddam, S. S., & Hellinga, B. R. (2013b). Quantifying measurement error in arterial travel times measured by bluetooth detectors. *Transportation Research Record: Journal of the Transportation Research Board*, 2395, 111–122. doi:10.3141/2395-13.
- Qiao, W., Haghani, A., & Hamed, M. (2013). A nonparametric model for short-term travel time prediction using bluetooth data. *Journal of Intelligent Transportation Systems*, 17(2), 165–175. doi:10.1080/15472450.2012.748555.
- Porter, J. D., Kim, D. S., Magaña, M. E., Poocharoen, P., & Arriaga, C. A. G. (2013). Antenna characterization for Bluetooth-based travel time data collection. *Journal of Intelligent Transportation Systems*, 17(2), 142–151. doi:10.1080/15472450.2012.696452.
- Quayle, S. M., Koonce, P., DePencier, D., & Bullock, D. M. (2010). Arterial performance measures with media access control readers. *Transportation Research Record: Journal of the Transportation Research Board*, 2192(-1), 185–193. doi:10.3141/2192-18.
- Rojas, R. (2013). *Neural networks: a systematic introduction*. Salmon Tower Building New York City: Springer Science & Business Media.
- Saeedi, A., Park, S., Kim, D. S., & Porter, J. D. (2013). Improving accuracy and precision of travel time samples collected at signalized arterial roads with bluetooth sensors. *Transportation Research Record: Journal of the Transportation Research Board*, 2380(-1), 90–98. doi:10.3141/2380-10.
- Technovelgy.com. (2016). What is RFID? Retrieved January 31, 2017, from <http://www.technovelgy.com/ct/technology-article.asp>
- Tsubota, T., Bhaskar, A., Chung, E., & Billot, R. (2011). Arterial traffic congestion analysis using bluetooth duration data. *Australasian Transport Research Forum*, (September), 28–30.
- Van Boxel, D., Schneider, W. H., & Bakula, C. (2011). Innovative real-time methodology for detecting travel time outliers on interstate highways and urban arterials. *Transportation Research Record: Journal of the Transportation Research Board*, 2256, 60–67. doi:10.3141/2256-08.
- Wasson, J. S., Sturdevant, J. R., & Bullock, D. M. (2008). Real-time travel time estimates using media access control address matching. *ITE Journal*, 78(6), 20–23.
- Yang, F., Yao, Z., Cheng, Y., Ran, B., & Yang, D. (2016). Multitime trip information detection using personal trajectory data. *Journal of Intelligent Transportation Systems*, 2450, 449–460. doi:10.1080/15472450.2016.1151791.
- Yang, S. (2015). Teach traffic smart. Retrieved Oct. 12, 2015, from <https://onthejeep.wordpress.com/2015/10/12/feedforward-neural-network-development-single-layer/>.
- Yang, S., An, C., Wu, Y.-J., & Xia, J. (2017a). Taxicab Availability: Empirical Study of Kunshan, China. *Transportation Research Record: Journal of the Transportation Research Board*, 2650. doi: <https://doi.org/10.3141/2650-06>
- Yang, S., An, C., Wu, Y.-J., & Xia, J. (2017b). Origin-destination based travel time reliability. *Transportation Research Record: Journal of the Transportation Research Board*, 2643, 139–159. doi:10.3141/2643-16.
- Yang, S., Liu, X., Wu, Y.-J., Wooschlager, J., & Coffin, S. L. (2015). (a). Can Freeway Traffic Volume Information Facilitate Urban Accessibility Assessment? *Journal of Transport Geography*, 44, 65–75. doi:10.1016/j.jtrangeo.2015.03.006.
- Yang, S., & Wu, Y.-J. (2016). Mixture models for fitting freeway travel time distributions and measuring travel time reliability. *Transportation Research Record: Journal of the Transportation Research Board*, 2594, 95–106. doi:10.3141/2594-13.
- Yang, S., Wu, Y. J., Marion, B., & Moses, I. E. (2015). Identification of transit farebox data errors: Impacts on transit planning. *Public Transport*, 7(3), 457–473. doi:10.1007/s12469-015-0107-6.
- Yang, S., Wu, Y.-J., Yin, Z., & Feng, Y. (2016). Estimating freeway travel times using general motors model. *Transportation Research Record: Journal of the Transportation Research Board*, 2594, 83–94. doi:10.3141/2594-12.
- Yang, S., Malik, A., & Wu, Y. (2014). Travel time reliability using hasofer lind-rackwitz fiessler algorithm and kernel density estimation. *Transportation Research Record: Journal of the Transportation Research Board*, 2442, 85–95. doi:10.3141/2442-10.
- Yao, X. (1999). Evolving artificial neural networks. *Proceedings of the IEEE*, 87, 1423–1447. doi:10.1109/5.784219.
- Zigbee.org. (2016). What is ZigBee? Retrieved January 31, 2017, from <http://www.zigbee.org/>.