Speed Estimation Using Smartphone Accelerometer Data

Ilyas Ustun (Corresponding author)
Data Scientist
Transportation Research Group
Department of Civil Engineering
5050 Anthony Wayne Dr.
Wayne State University
Detroit, MI 48202
ilyasustun@wayne.edu

Mecit Cetin
Associate Professor
Department of Civil and Environmental Engineering
Director, Transportation Research Institute (TRI)
Kaufman Hall 135
Old Dominion University
Norfolk, VA 23529
(757)-683-6700
mcetin@odu.edu

Submission Date: 8/1/2018
Revision Date: 11/15/2018

Total Word Count: 4,568 words text + 2 table x 250 words (each) = 5,068 words

Key Words: Speed estimation, smartphone, accelerometer, sensors, machine learning

Paper submitted for presentation at the 98th Annual Meeting of the Transportation Research Board and consideration for Publication in Transportation Research Record

ABSTRACT
This paper is focused on developing an algorithm to estimate vehicle speed from accelerometer data generated by an onboard smartphone. The kinetic theory tells that the integration of acceleration gives the speed of a vehicle. Thus, the integration of the acceleration values collected with the smartphone in the direction of motion would theoretically yield the speed. However, speed estimation by the integration of accelerometer data will not yield accurate results, since the accelerometer data in the direction of motion is not pure acceleration, but involves white noise, phone sensor bias, vibration, gravity component, and other effects. To account for these sources of noise and error, a calibration method that can adjust the speed at certain times or points is needed. The exact times when the vehicle stops and starts are identified and used to calibrate the estimated speed. Based on the collected sample data, the proposed method yields that the estimated speed is on average within 10 mph of the actual speed with a lower margin at the street-level driving. This suggests that with more information to calibrate the speed, the model accuracy can be improved further.

INTRODUCTION

The growth in mobile consumer devices, such as smartphones and tablets, has opened alternative ways to collect useful data to monitor traveler behavior and traffic flow. According to a February 2018 survey by the Pew Research Center, 77% of Americans own a smartphone, up from 35% in 2011 (*1*). Data aggregated from many mobile devices onboard vehicles as they navigate the transportation network provide a cost-effective opportunity to observe the performance of large-scale transportation networks.

Smartphone-based vehicular applications have been used in various fields such as traffic monitoring (*2*), accident detection (*3*), vehicle localization (*2; 4; 5*), and driving behavior analysis (*6-8*). For the most part, the use of smartphones as a mobile platform for sensing traffic conditions on the roadways has been primarily focused on the use of GPS data (*9-11*). Travel mode detection is a field where accelerometer and other sensors have been utilized for identifying the travel mode of the phone user. Travel modes include a wide variety of cases such as driving a car, riding a bicycle, taking a bus, walking, running, riding a metro, riding on light rail, and riding a train (*12; 13*). Acceleration data are generally used together with GPS data in mode detection (*14*). Using different sensor data in combination with GPS, or in essence speed data, have proven to yield more accurate results in these studies.

Here, the authors propose a speed estimation algorithm that uses smartphone accelerometer data. The kinetic theory tells that the integration of acceleration gives the speed of a vehicle. Thus, the integration of the acceleration values collected with the smartphone in the direction of motion would theoretically yield the speed. However, speed estimation directly by integration of accelerometer data is not possible, since the accelerometer data in the direction of motion is not pure acceleration, but involves white noise, phone sensor bias, vibration, gravity component, and other effects. These get integrated together with the motion data and produce inaccurate results. However, the linear trend in estimated speed shows that the behavior is consistent as can be seen in FIGURE 4 and proves that the speed can be corrected at certain calibration points. The motion stop and start points of a vehicle can be used to calibrate the estimated speed. The authors' previous research used smartphone accelerometer data and machine learning methods to come up with models that predict vehicle stop and start points with high accuracy (*15; 16*).

The focus of this paper is not on developing the best model to predict speed but rather building the groundwork for developing an algorithm that can reliably predict speed from smartphone sensory data. In addition, the research aims to predict speed without relying on GPS sensor within the smartphones for several reasons. GPS sensor has several limitations including low accuracy of GPS in urban areas with tall buildings because of the multi-path interference (*17*), low precision of GPS localization, and high-power consumption when the GPS is in use (*18-21*). Because of this high-power consumption, running an application that relies on the continuous use of GPS receiver depletes the phone battery quickly. However, the GPS can be turned on occasionally for a very short duration (e.g., one-two seconds) to locate the vehicle in a transportation network or a network link whereas the accelerometer data can be collected almost continuously to predict the stop and start points and to estimate the speed of a vehicle. The speed estimation using accelerometer provides an alternative to the generally applied case of using GPS. It can also be used to adjust and improve the GPS speed in cases of urban canyons and poor reception.

This paper contributes to the literature by proposing a speed estimation algorithm that use accelerometer data. The stop and start points detected using accelerometer data are used to calibrate the estimated speed. The usage of only the accelerometer sensor in the omnipresent smartphones has advantages compared to speed estimation methods that rely on loop detectors (*22-24*), or cameras (*25; 26*) as it gets rid of the need of infrastructure investments or maintenance related issues. The Methodology Section describes the speed estimation algorithm and how it is implemented using the high-resolution accelerometer data. While the developed method is not yet tested on very large datasets from multiple smartphones, the testing is done on field data collected by an Android phone while driving on typical urban and suburban roads. The model can be easily extended and applied on different smartphones and vehicles.

The organization of the paper is as follows: The next section describes the details of the data collected, followed by the methodology of speed estimation algorithm. This is followed by the application of the method on field data and presenting the results. Conclusions and potential improvements are presented at the end.

DATA

Field data are collected by a smartphone, and an on-board diagnostics (OBD) device. The OBD device used is capable of transmitting data via Bluetooth to the smartphone. An Android application is developed which records the data from the smartphone sensors, logs GPS readings and the speed data transmitted from the OBD. The GPS or OBD speed is the speed of the individual vehicle the phone is placed in. Each sensor has its own data sampling rate, and the app is set to log data from each sensor at the highest rate allowed. From the field data collected, it is found that the sampling rate for accelerometer sensor can be as low as 1 sample per second and as high as 238 samples per second, with the majority at around 15 samples per second. In general, the more the forces exerted on the phone, the higher the sensor activity is. The GPS and OBD data collection rate is 1 sample per second. To fix the time intervals between samples to a certain duration, these separate datasets from accelerometer, GPS, and OBD are first interpolated with a common start and end time and are resampled with a chosen sampling rate, and then appended together which yields the raw data. The rows represent each sample point, and the columns represent each sensory data of that sample. The OBD speed data are used as ground truth for model training and testing purposes. The data collection process involved normal driving on arterial streets with signalized intersections. The summary of training and testing trips are provided in TABLE 1 which includes the trip duration in minutes, number of time the vehicle stops in each trip, max speed in mph, trip distance in miles, and percentage of time the vehicle is in motion. These trips are from driving a Toyota Camry 2012 on arterials and city streets in Hampton Roads area of Virginia. The phone used throughout the data collection is a LG G4 android phone. This is important as each phone has a different sensor accuracy, sensitivity, and quality which might affect model accuracy if mixed with other phone types.

TABLE 1. Aggregate summary of datasets used.

	Dataset	# of Trips	Total Length(min)	Standstill Duration(min)	% Stop Duration	# of Stops	MaxSpeed (mph)	AvgSpeed (mph)
Camry	Train	10	204	34	17	74	74	30
	Val	10	182	18	10	36	82	40
	Test	20	488	77	16	137	78	33

METHODOLOGY

The time series data generated by the smartphone sensors can be either processed in real-time (e.g., every second) or offline. The nature of the application dictates whether an online (real-time) or an offline algorithm is needed. For example, for assessing the performance of a signalized intersection for signal timing design and planning, offline analysis would be sufficient. While online vs. offline distinction is important for algorithm design, the focus of this paper is investigating the feasibility of estimating speed using accelerometer data, not necessarily designing a fast algorithm for online applications. However, the lessons learned, and the presented methods could be extended to online applications with some modifications to estimate speed and to detect when a vehicle stops and when it starts accelerating back to its desired speed from a stop.

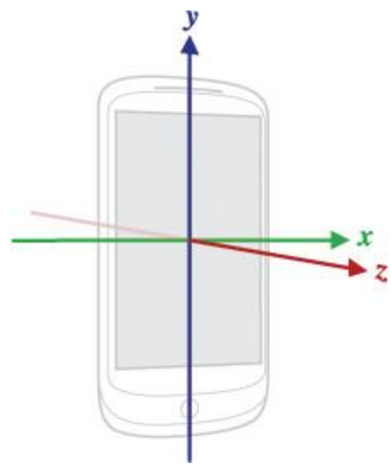

FIGURE 1 Smartphone sensor axes directions.

Algorithm for Estimating Speed

In this section, the process for estimating speed will be explained. One of the trips will be used to illustrate the case. The route taken during the trip and the vehicle's speed are shown in FIGURE 2. The trip was approximately 14 miles long and took about 22 minutes. The smartphone accelerometer sensor is a sensor with x, y, and z axes orthogonal to each other as shown in FIGURE 1. To be able to estimate speed, the accelerometer measurements taken in the direction of motion is needed. During the trips, the phone is positioned on its side a little tilted towards its bottom, such that the y-axis of the phone is oriented along the direction of the movement of the vehicle. The y axis or the longitudinal direction is the one axis that is least affected by gravity. This can be seen by checking the acceleration measurements during the standstills of the vehicle. The y axis is around 0 m/s^2, while x axis measures close to 8.0 m/s^2, and the z axis is around 5.2 m/s^2. This allows us to use the y axis acceleration obtained from the accelerometer to estimate speed. If the phone is positioned in the vehicle in a random orientation, orientation correction methods are needed (*11; 27-29*). The raw accelerometer values in the three axes logged from the smartphone, the magnitude of the acceleration 3D vector, and the GPS and OBD speeds of the vehicle are shown in FIGURE 3.

First, the kinematic equation shown in Eq. (1) is used to estimate speed.

$$V_f = V_i + a\Delta t \tag{1}$$

Here,

V_f Final velocity,

V_i Initial velocity,

a Acceleration,

Δt Time interval.

The frequency of interpolated data is held at a constant rate of 10Hz. Thus, the time interval will be 0.1 seconds between the measurements. Initial velocity is taken as the first speed value of the GPS, and then the vehicle speed at each instance is estimated recursively, based on Eq. (1).

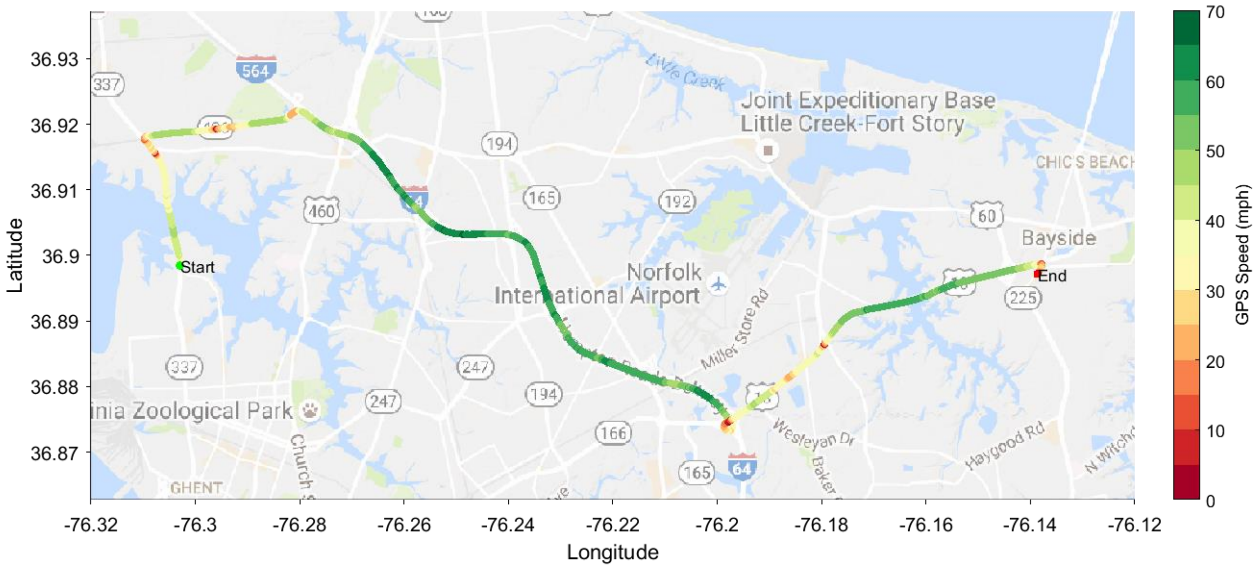

FIGURE 2 The route and speed of a Toyota Camry driven from Norfolk to Virginia Beach.

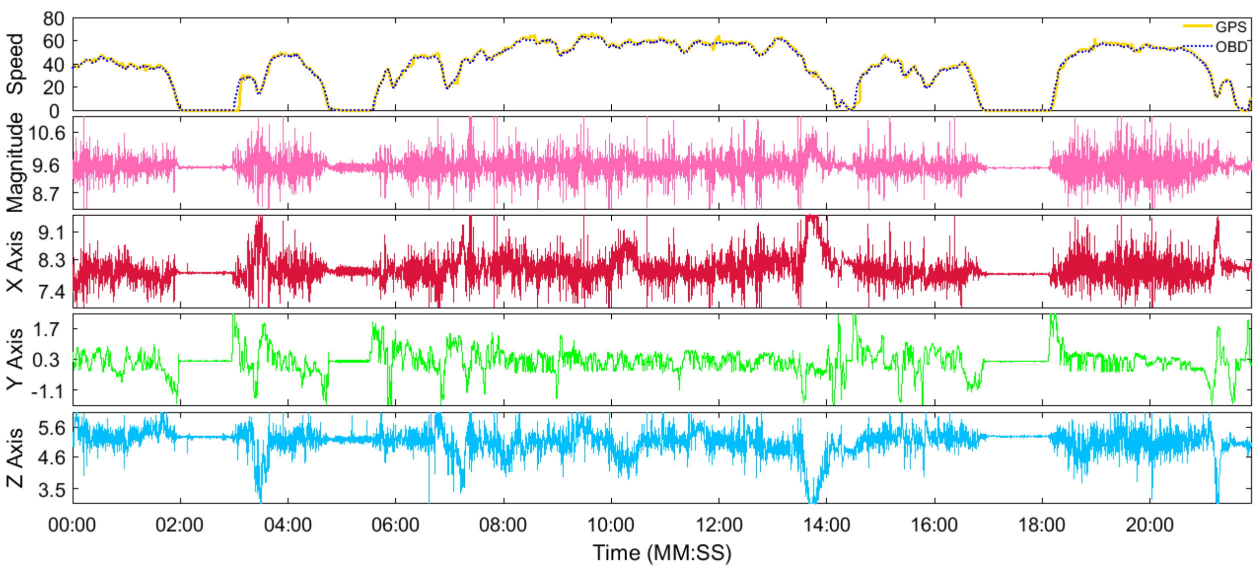

FIGURE 3 Raw accelerometer, magnitude of acceleration, and speed of GPS and OBD.

Normally, once this step is completed, one would expect that the speed estimate would be correct, based on the kinematic equation above. However, as mentioned before, because the phone is not oriented perfectly in the direction of motion, and other noise factors get accumulated as well during the integration process, the result is an upwardly (or downwardly) sloped monotonically increasing curve, as can be seen in FIGURE 4. As is evident, the monotonic increase can be identified by a slope within each motion and standstill segment. A segment is defined as the region between a stop and start point (standstill), or between a start and stop point (motion), which are shown in FIGURE 5. Here, the red vertical line signifies the stopping of the vehicle, while the green vertical line represents the start point. These points are referred to as change points. The state of each point *i* of a trip is denoted as a set *Q* containing values of ones and zeroes:

$$Q = \{1,0,0, \dots ,1,1,1,1, \dots ,1\} \quad (2)$$

The zeros (0) denote vehicle standstill at each instance, and the ones (1) denote the vehicle being in motion. To detect the change points (the stop and start points of a vehicle), the algorithm shown below is applied:

$$Q_i - Q_{i-1} \in \{0,1,-1\} \quad (3)$$

The difference between point *i* and point *i*-1 can only have three distinct values. "0" denotes no change in the state of the vehicle. "1" denotes that the vehicle was in standstill and started moving. The index of this point is stored in the Motion Start (*M*) set. A value of "-1" denotes that the vehicle was in motion and has stopped at this instance. The index of this point is stored in the Stop (*S*) set.

$$\begin{gathered} M = \{i | Q_i - Q_{i-1} = 1\} \\ S = \{i | Q_i - Q_{i-1} = -1\} \\ C = S \cup M \end{gathered} \quad (4)$$

Where $i = 1,2,3, \dots , |Q|$.

Thus, *M* is the set of indices when vehicle starts to move from a standstill, and *S* is the set of indices when the vehicle stops from being in motion. *C* is the sorted union of the stop and start change points in the trip, where C_k would represent one of the change points' index. There are *K + 1* change points, including the very first and end points of a trip, and there are *K* segments.

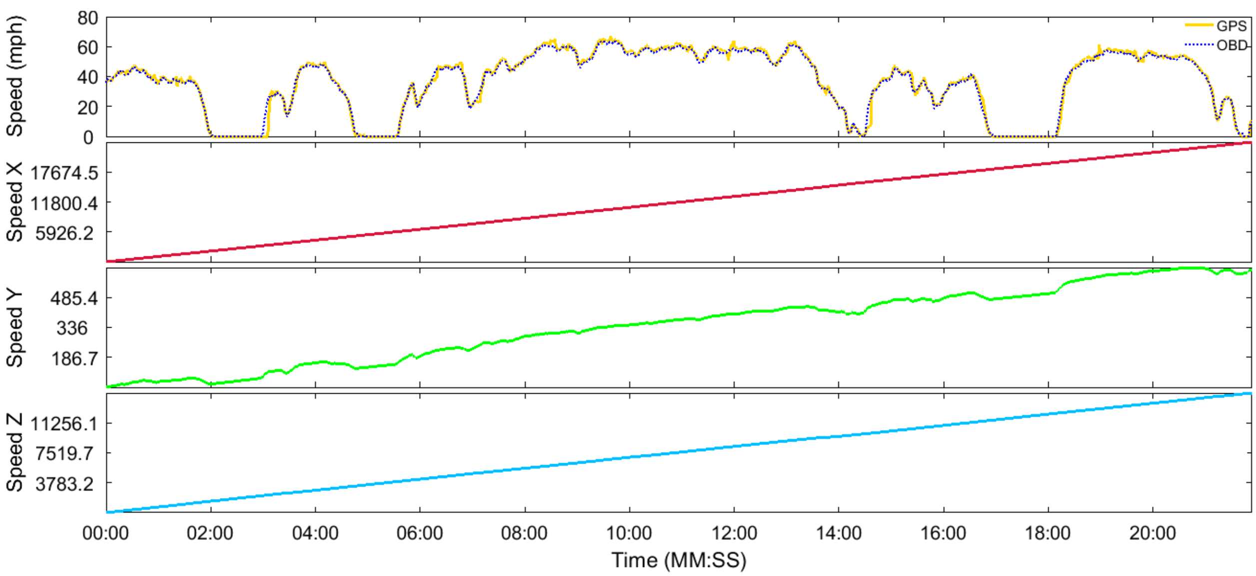

FIGURE 4 Speed estimations for each axis using the kinematic equation of $V_f = V_i + a\Delta t$.

Once the state detection phase is complete and the stop and start points are detected, the slope of each segment is computed, except for the first and last segments, unless these segments are a standstill. This is because, if the first and last segments start with motion, the calculated slope will be wrong, as is evident in FIGURE 5. The slope for each segment is calculated as the difference in speed between the last and first points of a segment, divided by the number of points in the segment. Since the change points represent the first point of each segment, the index of the last point of a segment is one less than the index of the next change point.

$$m_k = \big(V(C_{k+1} - 1) - V(C_k)\big)/(C_{k+1} - C_k) \tag{5}$$

Where,

m_k Slope of segment k

$V(C_k)$ Speed at the index of the state change point C_k

C_k The index of the change point k in the set C

Once the slope within each region is computed, the median is taken and the median $\widetilde{m}$ is used in the rest of the computations. If so desired, the individual slope values can be used in each segment as well. However, this might be a little noisy. The stop and start points are used to calibrate the speed estimation at standstill segments, where the speed is set to zero.

$$V = \{0 | Q_i = 0, for\ i = 1,2,3, \dots, |Q|\} \tag{6}$$

The speed at the beginning of each motion segment and the slope will be used to calibrate the estimated speed in the motion segments. The speed of each point is subtracted by the speed of the segment's first point and the product of the slope and the number of points from the beginning of the segment. The remainder is the calibrated speed estimation, which is the top part of the black vertical line shown in FIGURE 5. Here, by getting rid of the $V_{Initial}$ and V_{Slope} portions from V_i, the actual speed is left, which is denoted by $V_{Calibrated}$. In a sense, each point in the motion segment is pulled down to the expected speed level.

$$\begin{gathered} V = \left\{V_i - \widetilde{m}(i - C_k) - V(C_k) \,\middle|\, Q_i = 1, k > 1\right\} \\ V = \left\{V_i - \widetilde{m}(i - C_k) \,\middle|\, Q_1 = 1, k = 1 \right\} \end{gathered} \tag{7}$$

Where $i = 1,2,3, \dots, |Q|$.

The calibration phase finalizes the speed estimation process. The calibrated speed estimation for each axis is shown in FIGURE 6. Here, it becomes obvious that the motion direction of the vehicle aligns mostly with the Y axis of the phone, as the speed estimation obtained on this axis is the best among the three. It can be seen that the performance of the speed estimation is quite good and that it mimicked the actual speed very closely.

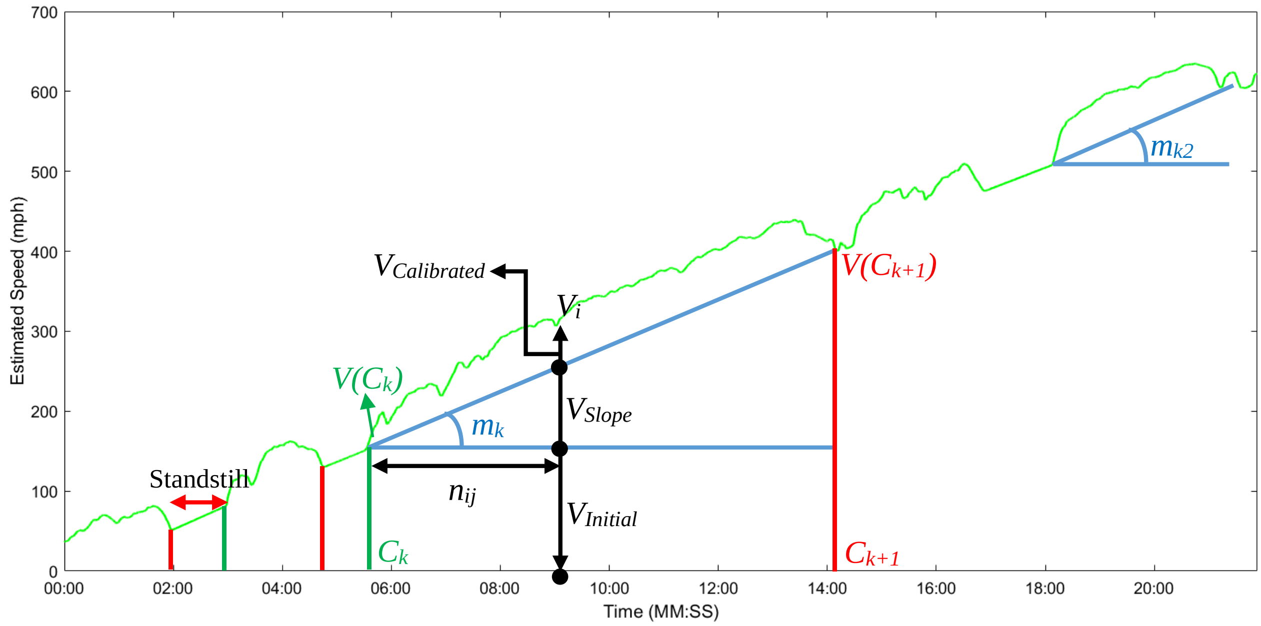

FIGURE 5 Calculating the slope and finding the beginning and ending speeds of segments.

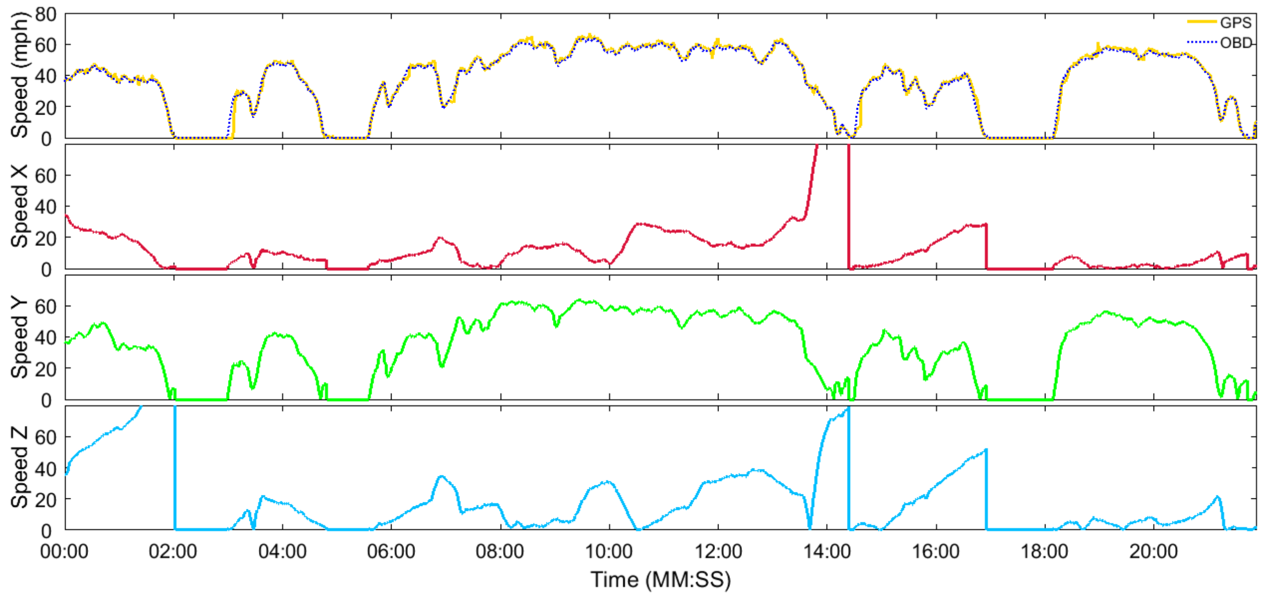

FIGURE 6 Calibrated speed estimation on each axis using the observed change points.

EMPIRICAL ANALYSES AND RESULTS

An important issue that needs to be mentioned is that, until now, the change points from observed set are used for calibration. The high accuracy of speed estimation using observed points proves that the speed estimation algorithm is a feasible method. However, the real performance of the speed estimation process can be obtained by testing with predicted change points. Here, the estimated speed will be calibrated by using predicted stop and start points which are obtained by implementing machine learning techniques in which the smartphone accelerometer data is utilized (*15; 16*), making the whole algorithm purely accelerometer dependent, without any extra knowledge.

The estimated speed obtained by using the predicted change points is shown in FIGURE 7. Visually, it can be deduced that the method still performs very well, with some overestimation between the minutes 6 and 14, in which the vehicle is travelling on the freeway. Some other dynamics might be causing this overestimation. The RMSE of estimated speed with respect to GPS speed using the observed and predicted points are shown in TABLE 2. As can be seen, both achieve good accuracy, with an RMSE of ~6, meaning approximately 6 mph difference in speed compared to the actual one. Using the predicted points is not very different than using the observed ones, and only degrades the accuracy by ~1.2 mph.

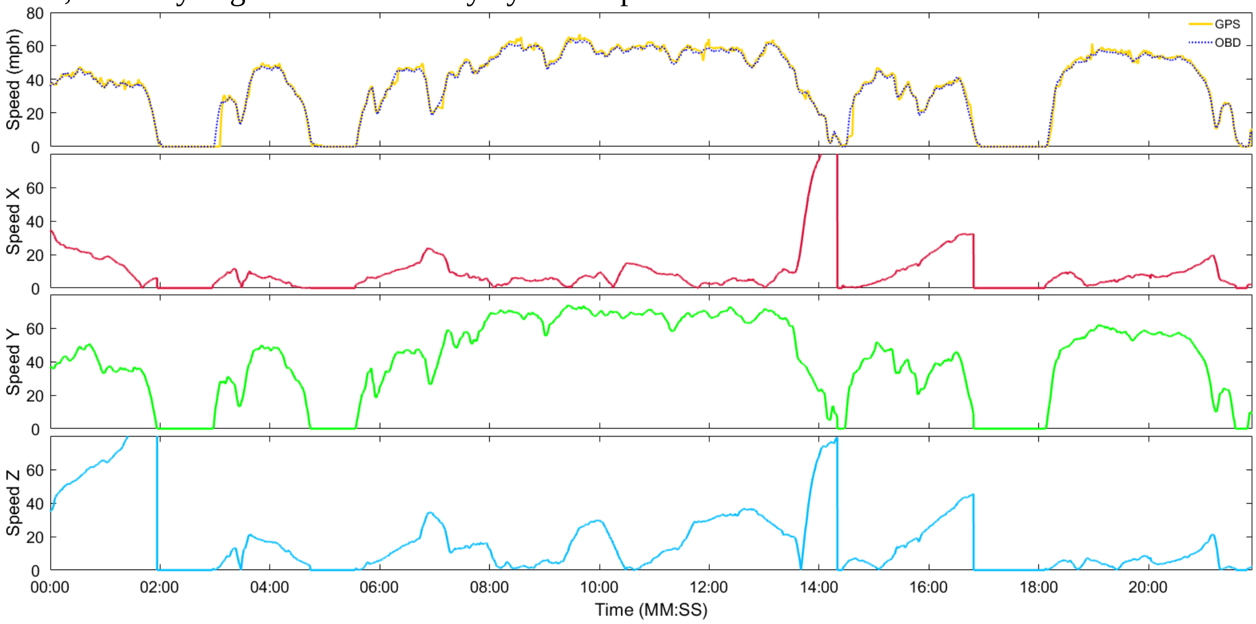

FIGURE 7 Calibrated speed estimation on each axis using the predicted change points.

The mean RMSE of speed estimation algorithm applied to all the test trips is provided on the right of TABLE 2. The stop and start point predictions of fifteen of the test trips are shown in FIGURE 8. Overall accuracy decreased when many trips are considered. However, the speed estimation whereby calibration is done with the predicted points still performed close to using the observed points, with slightly more error.

TABLE 2. The RMSE of estimated speed w.r.t. GPS speed.

	Single Trip RMSE	Testing Set Mean RMSE
Observed Change Points	5.49	9.78
Predicted Change Points	6.68	11.16

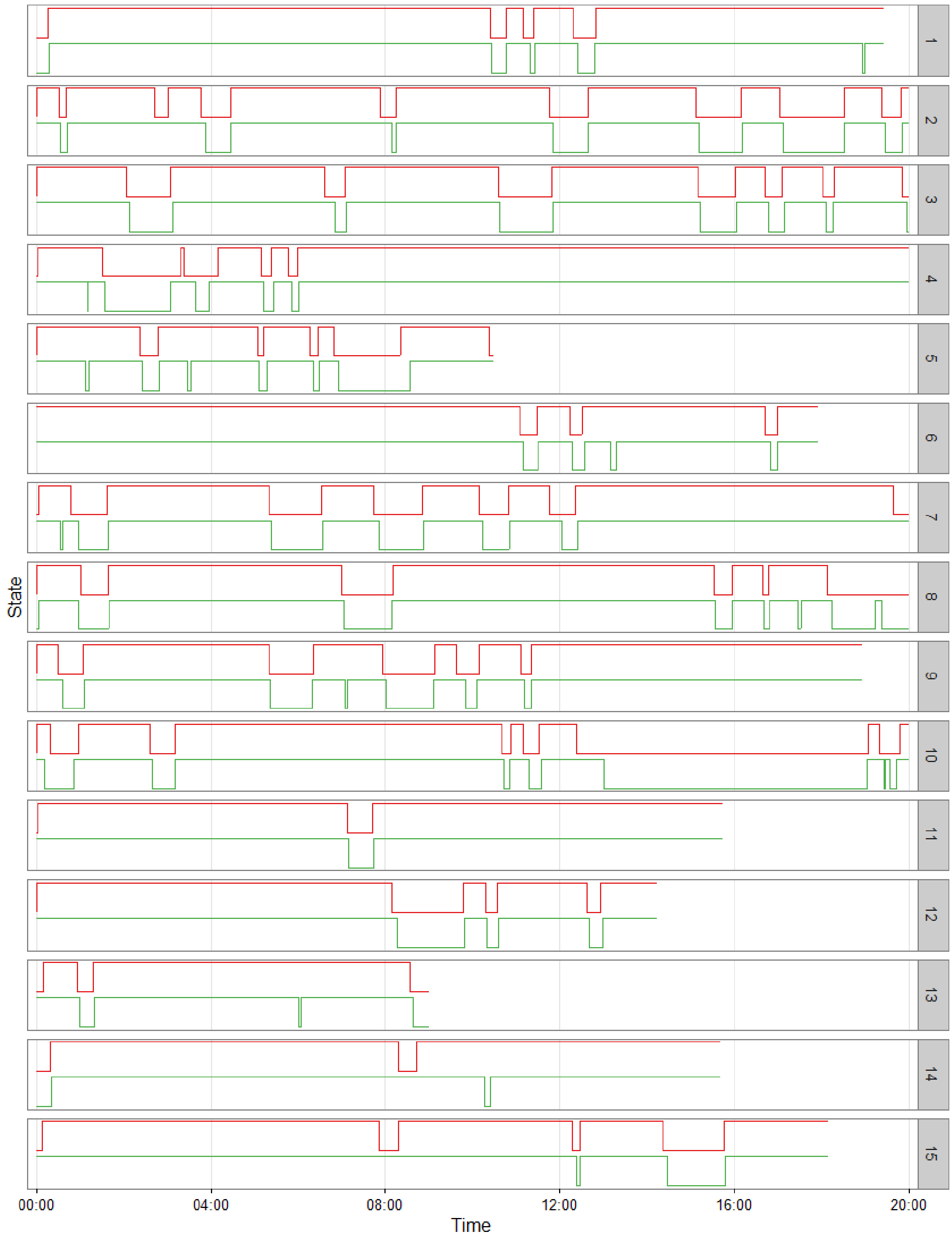

FIGURE 8 Camry stop and start prediction of test trips. Red: Prediction, Green: Observed

CONCLUSIONS

This paper demonstrates how speed can be estimated from smartphone accelerometer data and calibrated by using the stop and start points of a vehicle. The integration of the acceleration values collected with the smartphone in the direction of motion would theoretically yield the speed. However, integration of accelerometer data in the direction of motion also accumulates other components such as white noise, phone sensor bias, vibration, gravity component, and other effects. However, the linear trend in estimated speed proves that the speed can be corrected at some calibration points. The stop and start points of a vehicle provide the necessary calibration points. The speed estimation by using only accelerometer data and the predicted stop and start points used for calibration performs very well on surface street driving and slightly less so on highway driving. Based on the collected sample data, the proposed method yields that the estimated speed is on average within 10 mph of the actual speed. The main factor here is the frequency of stops as they are used for calibration. This suggests using more information to calibrate the speed can improve the model accuracy.

Some alternatives for improvement include the use of heuristic methods, such as increasing or decreasing the estimated speed by a certain amount if it under or overestimates consistently in certain type of segments. One such approach could be defined for segments over highways, which have higher speeds for longer durations. A better approach could be to make use of other motion information such as going over potholes, whereby the accelerometer signatures created at the front and back tires can be used to further calibrate the speed. Here, the phone was oriented in a way such that the longitudinal direction would face the direction of motion. However, this is not always possible. Making use of the gyroscope and re-orienting the mobile device to align with the vehicle orientation can also be done to further improve the speed estimation.

The importance of the model presented in the paper is to show that with minimal information and only single sensory data the speed estimation can still be reliably done and can be an alternative to the generally applied case of using GPS which has its own disadvantages. However, it should be noted that speed estimation using accelerometer is done using a very noisy data source and the re-orientation of a phone might add to the problem as well. The speed estimation algorithm presented in this paper can serve to correct and supplement the GPS speed in cases of urban canyons and poor reception.

ACKNOWLEDGEMENT

The research presented in this paper was funded by the TranLIVE University Transportation Center which is led by the University of Idaho.

AUTHOR CONTRIBUTION STATEMENT

The authors confirm contribution to the paper as follows: study conception and design: Ustun and Cetin; data collection: Ustun and Cetin; analysis and interpretation of results: Ustun; draft manuscript preparation: Ustun. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

[1] Center, P. R. *Mobile Fact Sheet*. http://www.pewinternet.org/fact-sheet/mobile/. Accessed November 2018.

[2] Mohan, P., V. N. Padmanabhan, and R. Ramjee. Trafficsense: Rich monitoring of road and traffic conditions using mobile smartphones. *Tech. Rep. no. MSR-TR-2008–59*, 2008.

[3] White, J., C. Thompson, H. Turner, B. Dougherty, and D. C. Schmidt. Wreckwatch: Automatic traffic accident detection and notification with smartphones. *Mobile Networks and Applications*, Vol. 16, No. 3, 2011, pp. 285-303.

[4] Bo, C., X.-Y. Li, T. Jung, X. Mao, Y. Tao, and L. Yao. Smartloc: Push the limit of the inertial sensor based metropolitan localization using smartphone.In *Proceedings of the 19th annual international conference on Mobile computing & networking*, ACM, 2013. pp. 195-198.

[5] Ahmed, U., O. Sahin, and M. Cetin. Minimizing GPS Dependency for a Vehicle's Trajectory Identification by Using Data from Smartphone Inertial Sensors and Onboard Diagnostics Device. *Transportation Research Record: Journal of the Transportation Research Board*, No. 2644, 2017, pp. 55-63.

[6] Johnson, D. A., and M. M. Trivedi. Driving style recognition using a smartphone as a sensor platform.In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, IEEE, 2011. pp. 1609-1615.

[7] Hong, J.-H., B. Margines, and A. K. Dey. A smartphone-based sensing platform to model aggressive driving behaviors. Presented at Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Toronto, Ontario, Canada, 2014.

[8] Paefgen, J., F. Kehr, Y. Zhai, and F. Michahelles. Driving behavior analysis with smartphones: insights from a controlled field study.In *Proceedings of the 11th International Conference on mobile and ubiquitous multimedia*, ACM, 2012. p. 36.

[9] White, J., C. Thompson, H. Turner, B. Dougherty, and D. Schmidt. WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones. *Mobile Networks and Applications*, Vol. 16, No. 3, 2011, pp. 285-303.

[10] Killick, R., K. Haynes, and I. A. Eckley. changepoint: An R package for changepoint analysis: R package version 2.0.1.In, 2015.

[11] Mohan, P., V. N. Padmanabhan, and R. Ramjee. Nericell: Rich monitoring of road and traffic conditions using mobile smartphones.In *6th ACM Conference on Embedded Networked Sensor Systems, SenSys 2008, November 5, 2008 - November 7, 2008*, Association for Computing Machinery, Raleigh, NC, United states, 2008. pp. 323-336.

[12] Manzoni, V., D. Manilo, K. Kloeckl, and C. Ratti. Transportation mode identification and real-time CO2 emission estimation using smartphones.In, 2010.

[13] Lester, J., P. Hurvitz, R. Chaudhri, C. Hartung, G. Borriello, and MobileSense - Sensing Modes of Transportation in Studies of the Built Environment. Presented at International Workshop on Urban, Community, and Social Applications of Networked Sensing Systems - UrbanSense08, 2008.

[14] Feng, T., and H. J. P. Timmermans. Transportation mode recognition using GPS and accelerometer data. *Transportation Research Part C: Emerging Technologies*, Vol. 37, No. 0, 2013, pp. 118-130.

[15] Ustun, I., A. Alasadi, M. Cetin, and T. Nadeem. Detecting vehicle stops from smartphone accelerometer data Presented at ITS World Congress, Detroit, 2014.

[16] Cetin, M., I. Ustun, and O. Sahin. Classification Algorithms for Detecting Vehicle Stops from Smartphone Accelerometer Data. Presented at Transportation Research Board, Washington DC, 2016.
[17] Parkinson, B. W. GPS error analysis.In *Global Positioning System: Theory and applications, No. I*, 1996. pp. 469-483.
[18] Ballantyne, W., G. B. Turetzky, G. Slimak, and J. Shewfelt. Achieving low energy-per-fix in cell phone. *GPS World*, Vol. 17, No. 7, 2006, pp. 24-32.
[19] Raskovic, D., and D. Giessel. Battery-Aware Embedded GPS Receiver Node.In *Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, 2007. pp. 1-6.
[20] Simjee, F., and P. H. Chou. Accurate battery lifetime estimation using high-frequency power profile emulation.In *Low Power Electronics and Design, 2005. ISLPED '05. Proceedings of the 2005 International Symposium on*, 2005. pp. 307-310.
[21] Lane, N. D., Y. Chon, L. Zhou, Y. Zhang, F. Li, D. Kim, G. Ding, F. Zhao, and H. Cha. Piggyback CrowdSensing (PCS): energy efficient crowdsourcing of mobile sensor data by exploiting smartphone app opportunities.In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, ACM, 2013. p. 7.
[22] Lao, Y., G. Zhang, J. Corey, and Y. Wang. Gaussian Mixture Model-Based Speed Estimation and Vehicle Classification Using Single-Loop Measurements. *Journal of Intelligent Transportation Systems*, Vol. 16, No. 4, 2012, pp. 184-196.
[23] Jain, M., and B. Coifman. Improved Speed Estimates from Freeway Traffic Detectors. *Journal of Transportation Engineering*, Vol. 131, No. 7, 2005, pp. 483-495.
[24] Coifman, B. Using Dual Loop Speed Traps To Identify Detector Errors. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 1683, 1999, pp. 47-58.
[25] Schoepflin, T. N., and D. J. Dailey. Dynamic camera calibration of roadside traffic management cameras for vehicle speed estimation. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 4, No. 2, 2003, pp. 90-98.
[26] Dailey, D. J., F. W. Cathey, and S. Pumrin. An algorithm to estimate mean traffic speed using uncalibrated cameras. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 1, No. 2, 2000, pp. 98-107.
[27] Engelbrecht, J., M. J. Booysen, G. v. Rooyen, and F. J. Bruwer. Survey of smartphone-based sensing in vehicles for intelligent transportation system applications. *IET Intelligent Transport Systems*, Vol. 9, No. 10, 2015, pp. 924-935.
[28] Bhoraskar, R., N. Vankadhara, B. Raman, and P. Kulkarni. Wolverine: Traffic and road condition estimation using smartphone sensors.In *2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012)*, 2012. pp. 1-6.
[29] Tundo, M. D., E. Lemaire, and N. Baddour. Correcting Smartphone orientation for accelerometer-based analysis.In *2013 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, 2013. pp. 58-62.