Automated transportation transfer detection using GPS enabled smartphones

Leon Stenneth [#1], Kenville Thompson [*2], Waldin Stone [*3], Jalal Alowibdi[#4]

[#] *Department of Computer Science, University of Illinois*
Chicago, IL, USA

[1] lstennet@cs.uic.edu,[4]jalowibd@cs.uic.edu

[*]*The City College of New York (CUNY)*
New York, USA

[2] kthompson@gc.cuny.edu,[3]waldin@gmail.com

Abstract— **Understanding the mobility of a traveller from mobile sensor data is an important area of work in context aware and ubiquitous computing. Given a multimodal GPS trace, we will identify where in the GPS trace the traveller changed transportation modes. For example, where in the GPS trace the traveller alight a bus and boards a train, or where did the client stop running and start walking. Using data mining schemes to understand mobility data, in conjunction with real world observations, we propose an algorithm to identify mobility transfer points automatically. We compared the proposed algorithm against the state of the art that is used in the previously proposed work. Evaluation on real world data collected from GPS enabled mobile phones indicate that the proposed algorithm is accurate, has a good coverage, and a good asymptotic run time complexity.**

I. Introduction

Determining a traveller's context from wearable computers is essential in pervasive computing and activity recognition. The mobility transfer points of a traveller denote some characteristics of his behaviour or travel pattern. In [1, 3], it was pointed out that detecting the transportation modes travellers from their uploaded GPS traces becomes easier with knowledge of a person's mobility transfer points .

Another motivation for mobility transfer detection is mobility surveys. Travel demand surveys have taken multiple formats, such as telephone interviews and questionnaires. These data collection strategies rely on manual labelling of data after the completed trip, thus, these data collected manually retain inaccuracies. For example, one may not recall the exact time that they had exit a bus, or boarded a cab, or transfers from a bus to a train, etc. Using GPS enabled mobile phones is more reliable for reporting accurate location, trip time, and trip transfers [10]. Thus, if the precise mobility facts of individual users are recognized, it is possible to provide a more realistic travel demand picture.

In this paper, we are not concerned about distinguishing the traveller's mode of transportation, such as whether they are traveling by car, bus, or bike. Instead, we focus on identifying the points where she changed from one transportation mode to another. The proposed approach outperformed previously proposed algorithm for the same problem. More specifically, we compared the proposed work against the change point segmentation based algorithm that was proposed in [1, 3] and employed in the GEOLIFE project [6].

The proposed scheme is able to detect the modal transfer points automatically without knowledge of the mobile client's travel history (e.g. where the traveller parked) or external indexes such as GIS data (e.g. bus stop locations or road geometries). The detection algorithm is based on real world observations and knowledge extracted from collected GPS sensor data using data mining algorithms. From labelled collected GPS sensor data, we used data mining algorithms to study the correlations in the data. Based on this study, we determine certain threshold values. Using these pre-determined threshold values, in addition to real world rational, we developed the algorithm.

The proposed algorithm is compared against the change point segmentation based algorithm [1, 3] that is utilized in Microsoft's GEOLIFE project [6]. In [1, 3], the authors used only speed and acceleration to determine the transition points in the change point algorithm. However, depending on them (i.e. speed and acceleration) solely is not enough, in some cases such as traffic or extreme weather, the speed and acceleration of walking, car, and bus are the same. Thus, in addition to speed and acceleration, we also utilize the heading change and accuracy of the GPS points within the GPS trace to accurately detect mobility transitions.

II. Related Work

Transportation mode detection from sensor data has been documented in the literature [1, 2, 3, 4, 10, 13, 14, 15]. The general principle for mode detection is to apply machine learning algorithms on the collected sensor data, then for an unlabelled submission, the transportation mode is detected probabilistically. The states of the art are: (1) Reddy et al [5] combined GPS sensor data with accelerometer data in order to detect which mode of transportation a user is currently using (online system). In this work, the transportation mode detection is done in real time every second. Hence, there is no need to detect modal transfer points. (2) Stenneth et al [4] combined GPS sensor data with GIS information about the underlying transportation network to infer transportation modes. This proposed work is different as we only cater for transportation transition points and not transportation mode detection for the entire GPS trace.

The closest work to the proposed approach is Zheng et al [1, 3]. The transportation transfer algorithm proposed [1, 3] is utilized in Microsoft's GEOLIFE project [6]. In [1, 3, 6] they proposed and considered a *change point segmentation* method to determine where the mobile client (i.e. traveller) changes from one transportation mode to another. We compare the proposed work with the *change point* scheme in [1, 3] since it is the closest work to this paper. Our proposed algorithm is different from the change point segmentation method used [1, 3]. For example, in the proposed approach, from the traveller's GPS trace, we extracted speed, acceleration, GPS accuracy, and heading change. On the other hand, the change point method does not consider heading or GPS accuracy. Additionally, the change point method does not account for inaccurate GPS reports.

Several other papers have considered the accelerometer sensor on the traveller's mobile phone for transportation mode activity recognition [5, 13, 14]. This work is different since we did not use the accelerometer sensor. The GPS sensor is more common than the accelerometer. Further, [14, 15] utilizes Wi-Fi and cell tower technology for mobility detection. This proposed work by us did not consider Wi-Fi or cell tower information. Consequently, the proposed algorithm is different.

III. DATA MODEL

In this section, we discuss the data model. *Definition 1. **GPS sensor report.*** A GPS sensor report p_i, represents data submitted by the mobile client from their mobile device's GPS sensor. The format on the report is < lat, lon, t, v, h, acc > where: lat represents the latitude; lon represents longitude; t represents the timestamp of the sensor report; v represents the current ground speed of the device; h represents the direction of travel; acc is the accuracy level of the latitude and longitude coordinates.

From each GPS report we need to compute two extra variables, heading change and acceleration. Given a finite set of GPS reports $\{p_1, p_2, p_3, p_4 \ldots p_n\}$, the heading change of the i^{th} GPS report is given by

$$p_i^{heading\ change} = (|p_i^h - p_{i-1}^h|)$$

$$\forall 2 \le i \le n \ and\ p_1^{heading\ change} = p_1^h$$

Likewise, the acceleration of the i^{th} GPS report is given by

$$p_i^{acceleration} = (|p_i^v - p_{i-1}^v|)/(|p_i^t - p_{i-1}^t|)\ \forall 2 \le i \le n$$

Definition 2. **GPS Trace**. A GPS trace T in our model is a sequence of time-stamped related GPS reports, $T = p_0 \rightarrow p_1 \rightarrow \cdot\cdot\cdot \rightarrow p_k$, $\forall 0 \le i \le k$, $p_{i+1}^t > p_i^t$.

Definition 3. **GPS Slow Point**. The slow points on a trajectory T is a point p_i, where p_i^v < *speed threshold* and the acceleration of p_i is below some *acceleration threshold*. The acceleration of p_i, is $p_i^{acceleration} = (p_i^v - p_{i-1}^v)/(p_i^t - p_{i-1}^t)$.

In our experiments, the *speed threshold* = 1.88 m/s and the *acceleration threshold* is 0.4 m/s^2. These thresholds were determined based on data mining techniques (see Section VI). On the contrary, if a GPS point is not a GPS Slow Point, it is then called a GPS fast Point.

Definition 4. **Single Modal GPS Trace**. Given a labelled GPS trace T, with points $p_0 \rightarrow p_1 \rightarrow \cdot\cdot\cdot \rightarrow p_k$, T is considered a single modal GPS trace if $\forall 0 \le i \le k$, $p_i^{transportation\ mode}$ is the same. Intuitively, this means that all the points $p_0 \rightarrow p_1 \rightarrow \cdot\cdot\cdot \rightarrow p_k \,\mathcal{E}\, T$, are labelled with the same transportation mode.

Definition 5. **Multi Modal GPS Trace**. Given a labelled GPS trace T, with points $p_0 \rightarrow p_1 \rightarrow \cdot\cdot\cdot \rightarrow p_k$, T is considered a multi modal GPS trace if $\forall i,j\ 0 \le i, j \le k$, $\exists i \exists j (p_i^{transportation\ mode} \ne p_j^{transportation\ mode})$. Intuitively, this means that there exists at least one GPS point from the set of all GPS points $\{p_0, p_1, \ldots p_{k-1}, p_k\}$ that has a different transportation modal label.

Definition 6. **GPS Trace Leg**. Given a trajectory T, as defined in *definition 2*. A GPS trace leg is subset of the continuous points $\{p_n \rightarrow {}_{pn+1} \rightarrow p_{n+2} \cdot\cdot\cdot\ p_m\}$, n>=0, m <= k from $\{p_0 \rightarrow p_1 \rightarrow \cdot\cdot\cdot \rightarrow p_k\}$, such that the subset corresponds to a *Single Modal GPS Trace*. Also, $\{p_0 \rightarrow p_{n-1}\}$ could represent zero or more *GPS Trace Legs*, likewise the subset $\{p_{m+1} \rightarrow p_k\}$.

A. *GPS sensor sample rate*

The sample rate of the GPS sensor is important in mobile sensing because sensor sampling consumes energy [14]. Since the algorithms are for mobile devices, it makes sense to adjust the sensor sample rate in accordance to the available battery power. In other words, the proposed algorithms should be energy-aware. In this work, we utilized a fix GPS sensor sample rate of once per second. Energy-aware dynamic sampling is a subject of future work.

B. *Mobile phone's GPS accuracy*

In this work, three types of mobile devices (Samsung Galaxy, IPhone 3G, and HP IPAQ) are considered for validating the proposed algorithms (see Appendix). Based on previous studies by Miller et al [11, 12], these devices possess a GPS accuracy of within 10m 95% of the time.

IV. PROPOSED ALGORITHM

In our modal transfer algorithm, we aim to discover start and end points of *GPS Trace Legs* where the corresponding modal label is walking. Given the multimodal GPS trace, we extract four properties; speed, acceleration, accuracy, and heading. In the change point method used in [1, 3], heading and accuracy was not considered.

The algorithm has three phases (1) Remove spurious data points that were introduced by GPS uncertainty (2) From the set of GPS points { $p_0 \rightarrow p_1 \rightarrow \cdots \rightarrow p_k$ } in the trajectory T, we form *GPS Trace legs*, where each *GPS Trace leg* modal label is walking or not. (3) Finally, we validate each *GPS Trace leg* whose modal label is walking by observing the *average heading change* for that *GPS Trace leg.*

C. *Phase 1 – Uncertainty pruning*

In phase one of the proposed work, we prune inaccurate GPS points. GPS systems may return inaccurate location results for several reasons, such as GPS sensor in building or in canyon. Pruning of inaccurate GPS points is done in two steps.

First, given a GPS trace T, with points $p_0 \rightarrow p_1 \rightarrow \cdots \rightarrow p_k$, we remove uncertain GPS reports by observing the accuracy parameter of the GPS report. We suppress GPS points whose accuracy values are greater than 40m. We ascertain points whose accuracy readings are high by scanning linearly through the GPS trace T and observing the accuracy attribute. For any point p_i in the GPS trace T, if $p_i^{accuracy}$ >40m, p_i is suppressed.

Next, starting from p_0, if a *GPS fast Point* lies between two *GPS slow Points*, we prune the *GPS fast Point.* Subsequently, if a *GPS slow Point* lies between two *GPS fast Points* we prune the *GPS slow point*. This stage will remove some amount of the uncertainty that was introduced by the GPS. For example, if a person is walking, an incorrect positioning may lead to an increase in the velocity computation, hence irrelevant *GPS fast Point* are introduced in the trajectory. This stage 1 prunes unauthentic points of this nature.

D. *Phase 2 – GPS trace leg construction*

The set of GPS points {$p_n \rightarrow p_{n+1} \rightarrow \cdots \rightarrow p_m$}, n>=0 and m<=k, remaining in T from stage 1 is called T′, T′ ⊂ T. Starting from p_n in T′, for consecutive sets of *GPS slow Points* or *GPS fast Points*, we prune all intermediary points except the first and last points of each set. This stage is the first step in forming trip legs. After this stage, the first and last point of each set form possibly *GPS Trace Legs*. It should be clear to the reader that these possible *GPS Trace Legs* contain only the start and end point of the leg. The set of remaining points is called T″, T″ ⊂ T′.

E. *Transportation transition point identification*

The remaining points from the set T′ in stage 2 is referred to as T″. Since it is highly likely that a modal transfer point is represented by a walk leg as observed in [1, 3], in this stage we identify and verify walk legs. A walk leg is identified by any consecutive points that are *GPS Slow Points* in T″. For each walk leg *WL* in T″, we revisit the trajectory T′, each GPS point in T′ between and inclusive of the start and end point of *WL* forms the candidate set for that walk leg. Next, for each candidate set, we compute the average heading change. If the average heading change is greater than the *heading threshold*, the walk leg's start and end points are added to T‴. We are interested in heading change to concur if a leg is a walk leg; the heading change of a person walking is greater than if they are using motorized modes [1, 3, 4]. Hence, at this stage we are able to distinguish walking from motorized modes in traffic.

Each walk leg's start point and end point in T‴ are now modal transfer points. In our previous work [4], we used classification models such as Decision Trees to infer a traveller's transportation mode probabilistically from the set {still, walk, bike, car, bus, above ground train}. This work is different; we did not use classification for mobility transfer point detection.

Our modal transfer algorithm is different from change point method in [1, 3] in a number of ways. First, in the proposed work we consider the speed, acceleration, heading, and accuracy. For example, in phase 1, we suppress inaccurate points by observing the accuracy reading of the GPS sensor reports. On the other hand the change point method only uses speed and acceleration.

Second, we validate all walk legs in stage 3 by considering another dimension of context, namely the heading change of the traveller. This was not the case for [1, 3]. Additionally, to remove uncertainties, we do not merge segments into its backward segment. Furthermore, the change point method in [1, 3] does not work well in congestion. Instead, in heavy and persistent traffic, even though one may be driving slowly or on a bus, Microsoft's change point algorithm concludes that the traveller is transferring, hence misclassifies the mobility pattern as a possible change point.

Our proposed algorithm solves this congestion problem by considering the heading change as another facet to validate walk legs in stage 3. Persons who are walking have a higher heading change rate than driving because cars and buses are constrained by the road network, and cannot change their heading as flexibly. Likewise, trains are constrained by the rail line.

V. ALGORITHM DEMONSTRATION

Below in Table 1 and Figure 2, we demonstrate our modal transfer algorithm for clarity. The *GPS trace* T consisting of ten GPS reports {a, b, c, d, e, f, g, h, i, j} are submitted and the accuracy of the reports are below 40m. The properties of the GPS reports are highlighted in Table 1 and are only for discussion purposes only.

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**
speed (m/s)	1	18	1.2	0.6	8	17	0.45	5	1.1	1.23
acceleration (m/s^2)	0.3	2.3	0.39	0.42	1	3.2	0.19	2.1	0.15	0.25
heading (degrees)	32	38	59	113	117	119	119.4	116.6	126.5	134.3

TABLE 1- SENSOR REPORT PROPERTIES

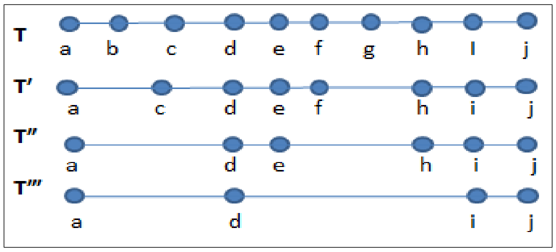

FIGURE 2- ALGORITHM TRACE

In phase 1 of the algorithm, given GPS trace T (see Figure 2), both GPS point b and GPS point g are removed from T to form T′. The points b and g are removed since b's speed is 18 m/s (fast point) that is between two slow points (i.e. a and c), and g is a slow point between two fast points. Next in phase 2 of our algorithm, we form possible GPS Trace legs in T″ by removing point c and f from T' since they are intermediary points of their trip legs. For example, in T' points a, c, and d forms a trip leg since they are consecutive slow points. We are only interested in the start and end points of each trip leg. Hence we can remove c. Observe that removing c is not because of uncertainty of the GPS report. Likewise, in T' points e, f, and h forms a trip leg since they are consecutive fast points. Again, we are interested in the start and end points of each leg. Thus, f is removed from T'. Observe, in T″ we have three possible GPS Trace legs {{a, d}, {e, h}, {i, j}}. Then for phase 3, in T‴ we only select the sets {a, d} and {i, j} for two reasons; (1) {a, d} and {i, j} are walk legs because they start and end with GPS slow points. (2) The average heading change of all points between GPS point a and GPS point d {a, d} also, GPS point i and GPS point j {i, j} in T′ is greater than the heading change threshold. In the experiments, we configured the heading change threshold system parameter to 1.5 degrees. Finally, GPS Trace Legs (walk legs {a, d} and {i, j}) in T‴ start and end points are modal transfer points. Below, we explain the strategy that we used to determine the thresholds used in the proposed work.

VI. CLUSTERING GPS TRACES TO DETERMINE THRESHOLDS

In the data model discussed earlier, we defined a GPS slow point using two thresholds. The first is the speed threshold (i.e. 1.88 m/s) and the second the acceleration threshold (i.e. 0.4 m/s^2). These thresholds are utilized as system parameters in the final algorithm. Below we give the motivation for assigning specific values for these thresholds. Using K-means clustering on collected GPS traces, we receive an insight on suitable values for these thresholds. We refer readers to [8] for details on this data mining algorithm. Using a clustering scheme, we can form meaningful groups from the dataset. In this way objects of the same clusters have high similarities, and objects of different clusters have big differences.

For this section, to determine the clusters, the dataset used is dataset 1 (see Appendix for further details on this dataset). To apply the K-Means algorithm on the dataset, we used the WEKA machine learning toolkit [9]. K-Means will partition the input into K sets. K was predetermined since data set 1 has only one mode of transportation (i.e. walking mode). Therefore, $K = 1$.

Given K, the K-Means algorithm then calculates a mean point (i.e. centroid) of each set. Given the centroid, it then associates each GPS report with the closest centroid. New centroids are computed, and the process is repeated until convergence. K-means minimizes the following objective function.

$$J = \sum_{j=1}^{K} \sum_{i=1}^{n} ||x_i^{(j)} - c_j||^2$$

Where $||x_i^{(j)} - c_j||^2$ is distance between data point $x_i^{(j)}$ (i.e. GPS report p_i) and its centroid c_j. However, we are only interested in data points or GPS reports that corresponds to walking, therefore $x_i^{(walk)}$ and also the centroid of the walking cluster (i.e.c_{walk}). Given c_{walk} , we can extract the acceleration $c\,^{acceleration}_{walk}$ and speed $c\,^{speed}_{walk}$ as the thresholds. Therefore,

$$acceleration\ threshold = c\,^{acceleration}_{walk}$$

$$speed\ threshold = c\,^{speed}_{walk}$$

Though we considered K-Means for threshold determination, we are aware that K-means is primarily for unsupervised learning. In general, any algorithm that computes the mean can be utilized for determining suitable thresholds.

VII. EVALUATION BY EXPERIMENT

First, we encourage readers to see the Appendix for a description of the data collection procedure and the datasets that we utilized when evaluating the algorithms. Both datasets (i.e. dataset 1 and dataset 2) are real world datasets collected by travellers from their mobile phones (see Appendix).

A. Evaluation matrices

Coverage - The coverage matrix for the algorithms is the ratio, number of modal transfer points in labeled ground truth for a multimodal GPS trace divided by the number of correct modal transfer points returned by algorithm that corresponds to the labeled ground truth. For example, if an experiment participant recorded 10 transportation transition points in their GPS trace and the algorithms only detected 8 correctly, the coverage is therefore 8/10=80%. The coverage of the algorithm is a measure of accuracy and effectiveness.

Number of irrelevant transfer points – This is a count of the number of false mobility transfer points return by the algorithm. This metric is also a measure of the accuracy of the algorithms. For example, for a given multimodal GPS trace $T^{multimodal}$, if the ground truth for the number of mobility transfer points is X, and the algorithm returns Z mobility transfer points such that $Z \geq X$. Then, the number of irrelevant transfer points in $T^{multimodal}$ is Z-X.

B. Results

In the Figures (Figure 3, Figure 4) we show preliminary results of our proposed modal transfer algorithm and the change point [1, 3] methodology that is used in GEOLIFE [6]. For evaluation purposes, to validate the proposed algorithm, 13 multimodal *GPS Traces* from thirteen different individuals, with transportation modes from the set {*still*, *walk*, *car*, *bus*, *train*, *cycle*} is considered. Description of this data set from the 13 individuals is presented in the Appendix. In Figure 3 the horizontal axis represents the trace number and the vertical axis represents the *percentage accuracy* of the transportation mode change detection.

Figure 3 illustrates that the modal transfer algorithm proposed in this paper has good coverage. In most cases, all the mobility transfer points were discovered in the multimodal GPS trace. In general, the proposed mobility transfer algorithm has a better coverage than the change point segmentation scheme. From Figure 3, in some cases we don't achieve 100%, for example for trajectory 3 in the labelled ground truth we have four modal transfer points. However our algorithm only detected three of these points. For the case of trajectory 3, the mobile client alights one bus and boards another at the same bus stop.

In general, Figure 3 indicates that the proposed modal transfer algorithm is more consistent than the change point method in [1, 3] since the expected number of modal transition points are always identified.

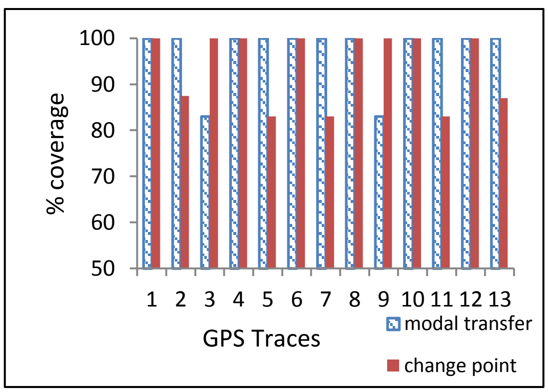

FIGURE 3– COMPLETENESS

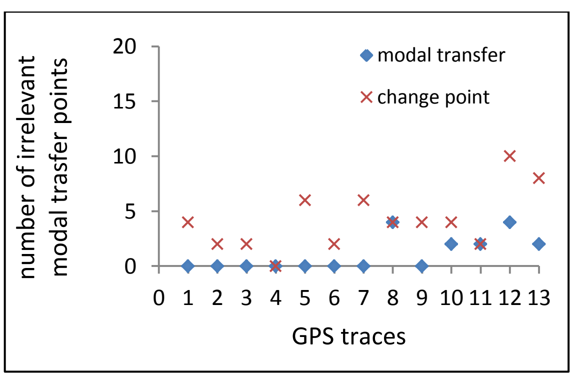

FIGURE 4 – IRRELEVANT TRANSPORTATION TRANSITION POINTS

We study the irrelevant modal transfer points in Figure 4. The vertical axis represents the number of irrelevant modal change points and the horizontal axis represents the *multimodal GPS traces*. In most cases, our algorithm detected the correct amount of transportation mode change points.

However, in some cases such as *GPS trace 8* and *GPS trace 12*; the proposed algorithm produces excessive transfer points. We observe that the location of the two traces *GPS trace 8* and *GPS trace 12* were collected in the city centre downtown Chicago, where GPS uncertainty is high because of the skyscrapers. Likewise, traces 10, 11, 12, and 13 were collected in heavy traffic. For the change point method used in [1, 3], in most cases the algorithm returns much more than the number of change points.

The fundamental reason why the change point algorithm will always return more transfer points than expected is due to the fact that in traffic and heavy weather, motorized vehicles move slowly. Hence, since the motorized vehicles are moving slowly, they may be misclassified as transfer points. For the proposed modal transfer algorithm, we solved this problem by considering mobility patterns derived from other parameters such as heading change and GPS accuracy.

VIII. COMPLEXITY ANALYSIS

In the analysis of the complexity of the proposed mobility transfer algorithm, we observe that the run time complexity is O(n). The complexity of step 1 is bounded by the input of modal traces represented by $|T| = n$. Also, in step 1, T' is derived, which in the worst case $|T'| = |T| = n$, which makes step 2 also executing n times. In step 2, T'' is used as input for step 3. In step 3, the worst case $|T''| = n$ and step 3 is executed n times. Overall, the complexity is bounded by n, hence O(n).

In [1, 3] the change point segmentation algorithm, the runtime is bounded by the input size n, which makes the overall complexity of the algorithm also O(n). However, the algorithm has an extra step (step 4) and performs a backward merger of consecutive uncertain segments. In our analysis we find that backward segment merging is done with n-1 comparisons. However, if sorting is involved with the merger then the complexity could be absorbed by the complexity of the sorting operation. Based on our observation, the merging is a simple concatenation and only needs n-1 comparisons. Thus, the overall complexity of change point method of [1, 3] is O(n) time. The space complexity of both algorithms is O(n).

IX. CONCLUSION

In this document, we propose a method to detect transfer points in multimodal GPS traces. Our algorithm is based on real world observations and parameter value selection via a K-means clustering strategy. Results indicate that the proposed approach is more accurate and effective for detecting mobility transfer points than the previously proposed change point algorithm [1, 3] used in Microsoft's GEOLIFE project [6]. The proposed algorithm is robust and simple; it can detect transfer points under extreme weather and traffic conditions.

Using two different real world datasets consisting of GPS traces collected via mobile phones, we evaluated the algorithms under varying traffic conditions (see appendix). These datasets were collected under varying traffic conditions because we wanted to understand the algorithm's effectiveness in the real world. While in the prior art the authors only considered speed and acceleration to detect transportation mode transition points, we consider more parameters. We believe that using these additional parameters on GPS accuracy and heading change will give us an enhanced picture in terms of understanding the mobility of the traveller. This approach has a better coverage in the sense that it will find all the mobility transfer points. Additionally, it produces less false transition points than [1, 3], hence it is more accurate.

The run time complexity of the proposed algorithm is O(n), where n is the number of GPS reports in the GPS trace. This is asymptotically the same as the previously proposed work.

X. REFERENCES

[1] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W. Ma, Understanding mobility based on GPS data. *ACM UbiComp, 2008.*

[2] Liao, D. Patterson, D. Fox, and H. Kautz. Learning and inferring transportation routines. *Artificial Intelligence, 2007.*

[3] Y. Zheng, L. Liu, L. Wang, X. Xie. Learning Transportation Mode from Raw GPS Data for Geographic applications on the Web. *WWW, 2008*

[4] L. Stenneth, P. Yu, O. Wolfson, B. Xu. Transportation Mode Detection from Mobile Phones and GIS Information. *ACM SIGSPATIAL GIS 2011*.

[5] S. Reddy, M. Mun, J. Burke, D. Estrin, M Hansen, and M. Srivastava. Using Mobile Phones to Determine Transportation Modes. *ACM Transactions on Sensor Networks, Vol. 6, No. 2, Article 13, 2010.*

[6] Microsoft's GEOLIFE project: http://research.microsoft.com/en-us/projects/geolife/

[7] The Geo-Location API: http://dev.w3.org/geo/api/spec-source.html

[8] A. Jain, M. Murty, P. Flynn. Data Clustering: A review. *ACM Computing surveys Vol. 31, No. 3, 1999.*

[9] I. Witten, E. Frank, Data Mining: Practical machine learning tools and techniques. *Morgan and Kaufmann, 2005*.

[10] E. Murakami, and D. P. Wagner, Can using global positioning system (GPS) improve trip reporting? *Transportation Research Part C, 7(2/3):149-165, 1999.*

[11] T. Menard, J. Miller, M. Nowak, D. Norris. Comparing the GPS Capabilities of the Samsung Galaxy S, Motorola Droid X, and the Apple IPhone for Vehicle Tracking using FreeSim_Mobile. *14th IEEE Intelligent Transportation Systems Conference, 2011*.

[12] T. Menard, J. Miller. Comparing the GPS Capabilities of the IPhone 4 and IPhone 3G for Vehicle Tracking using FreeSim_Mobile. *7th IEEE Intelligent Vehicle Symposium, 2011.*

[13] L. Bao, S. Intille. Activity recognition from user-annotated acceleration data. *In Lecture Notes in Computer Science. Springer-Verlag, 2004.*

[14] Y. Wang, J. Lin, M. Annavaram, Q. Jacobson, J. Hong, B. Krishnamachari, N. Sadeh. A Framework of Energy Efficient Mobile Sensing for Automatic User State Recognition. *ACM MobiSys, 2009.*

[15] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, P. Boda. PEIR, the Personal Environmental Impact Report, as a Platform for Participatory Sensing Systems Research. *ACM MobiSys, 2009.*

XI. APPENDIX

In this section, we will discuss the datasets that is utilized to evaluate the algorithms and the data collection strategy.

Data Collection

The discussion of the data collection methods are separated in two sections (data set 1, data set 2). Both data sets were collected in Chicago, Illinois, USA.

Dataset 1

This is the data set that was considered to determine the threshold values used by our algorithm. The GPS traces in the dataset is single modal of type walking was collected by 6 individuals, 3 females and 3 males over a 3 week period. On this dataset we performed K-means clustering to pre-determine system parameter settings. Additionally, three types of mobile devices were used for data collection. The three devices considered are: (1) HP IPAQ PDA, (2) Android based Samsung Galaxy, and (3) IPhone 3G. These devices are shown below in Figure 5. In total, this dataset has 91 minutes of GPS trace single modal data.

FIGURE 5- MOBILE DEVICES UTILIZED FOR DATA COLLECTION

Dataset 2

This data set was used to evaluate the algorithms. This data set consists of 13 multimodal GPS traces, supplied by thirteen different participants (seven males and six females). The sample participants were different in both experiment datasets, to ensure a level of robustness and adaptability. Three different types of mobile phones were used for data collection and proposed algorithm. The three mobile devices are Samsung Galaxy, HP IPAQ, and the IPhone 3G (see Figure 5).

Four of the thirteen traces were collected in heavy traffic in the Downtown Metropolitan region of Chicago. We considered heavy traffic because we wanted to understand how the algorithm performs in the real world. Dataset 2 was collected over a one week period and consists of trips with transportation modes from the set walking, bus, car, still, aboveground train and bike. This data is also labeled with transportation mode and mode transition point ground truth. For algorithm evaluation, the mode transition labels are pruned, and next when the algorithms generate the transition points, the results are compared with the mode transition ground truth. We studied the algorithms' effectiveness in the real world such as varying traffic. In total, dataset 2 has 264 minutes of multimodal GPS traces.