Understanding Transportation Modes Based on GPS Data for Web Applications

YU ZHENG
Microsoft Research Asia
YUKUN CHEN
Tsinghua University
QUANNAN LI
Huazhong University of Science and Technology
and
XING XIE and WEI-YING MA
Microsoft Research Asia

User mobility has given rise to a variety of Web applications, in which the global positioning system (GPS) plays many important roles in bridging between these applications and end users. As a kind of human behavior, transportation modes, such as walking and driving, can provide pervasive computing systems with more contextual information and enrich a user's mobility with informative knowledge. In this article, we report on an approach based on supervised learning to automatically infer users' transportation modes, including driving, walking, taking a bus and riding a bike, from raw GPS logs. Our approach consists of three parts: a change point-based segmentation method, an inference model and a graph-based post-processing algorithm. First, we propose a change point-based segmentation method to partition each GPS trajectory into separate segments of different transportation modes. Second, from each segment, we identify a set of sophisticated features, which are not affected by differing traffic conditions (e.g., a person's direction when in a car is constrained more by the road than any change in traffic conditions). Later, these features are fed to a generative inference model to classify the segments of different modes. Third, we conduct graph-based postprocessing to further improve the inference performance. This postprocessing algorithm considers both the commonsense constraints of the real world and typical user behaviors based on locations in a probabilistic manner. The advantages of our method over the related works include three aspects. (1) Our approach can effectively segment trajectories containing multiple

This article is an expanded version of Zheng et al. [2008], which appears in Proceedings of the 4th International Conference on the World Wide Web, 247–256.
Authors' addresses: Y. Zheng, X. Xie, and W. Y. Ma, Microsoft Research Asia, 4F Sigman Building, 49 Zhichun Road, Haidian District, Beijing 100190, China; email: {yuzheng,xingx,wyma}@microsoft.com; Y. Chen, Department of Computer Science, Tsinghua University, Beijing 100084, China; email: chenyukun03@hotmail.com; Q. Li, Huazhong University of Science and Technology, Wuhan 430074, China.

DOI 10.1145/1658373.1658374 http://doi.acm.org/10.1145/1658373.1658374

transportation modes. (2) Our work mined the location constraints from user-generated GPS logs, while being independent of additional sensor data and map information like road networks and bus stops. (3) The model learned from the dataset of some users can be applied to infer GPS data from others. Using the GPS logs collected by 65 people over a period of 10 months, we evaluated our approach via a set of experiments. As a result, based on the change-point-based segmentation method and Decision Tree-based inference model, we achieved prediction accuracy greater than 71 percent. Further, using the graph-based post-processing algorithm, the performance attained a 4-percent enhancement.

Categories and Subject Descriptors: I.5.2 [**Pattern Recognition**]: Design Methodology—*Classifier design and evaluation*; H.2.8 [**Database Management**]: Database Applications—*Data mining*

General Terms: Algorithm, Design, Experimentation

Additional Key Words and Phrases: Spatial data mining, GPS trajectory, ubiquitous computing, understanding user behavior, GeoLife, user mobility, transportation modes

ACM Reference Format:

Zheng, Y., Chen, Y., Li, Q., Xie, X., and Ma, W.-Y. 2010. Understanding transportation modes based on GPS data for Web applications. ACM Trans. Web, 4, 1, Article 1 (January 2010), 36 pages.
DOI = 10.1145/1658373.1658374 http://doi.acm.org/10.1145/1658373.1658374

1. INTRODUCTION

The increasing pervasiveness of location-acquisition technologies, such as GPS and GSM network, is leading to the large collection of spatio-temporal datasets. Such datasets have supported a variety of novel Web applications, in which locality and mobility usually connect to one another closely. For instance, people can tag user-generated contents like photos with locations [Toyama et al. 2003]; trace their outdoor mobility [Ashbrook and Starner 2003]; and use location-based services [Chen and Kotz 2000]. Recently, a branch of GPS-track-sharing applications using Web maps appeared on the Internet. In this category of Web applications [Counts and Smith 2007], people can record their travel routes using a GPS-equipped device and then share travel experiences among each other by publishing these GPS tracks in a Web community. GPS-track-sharing offer a more fancy and interactive approach than text-based articles to better express people's travel experiences, which provide users with valuable references when planning a travel itinerary.

However, so far, these applications require people either to manually label their own trajectories or to use raw GPS data such as GPS coordinates and timestamps without much understanding. Neither of these methods is optimal to the development of such applications. Actually, users become easily frustrated by the additional data-labeling effort, and then give up uploading their data. Moreover, people intend to understand an individual's mobility, and learn information about user behaviors as well as user intentions behind the raw data.

Being an important kind of human behavior, transportation modes, such as walking, driving, and taking a bus, can enrich their mobility with knowledge and provide pervasive computing systems with more contexts.

—For users. The information of transportation modes helps individuals effectively reflect on their past events, and deeply understand their own life pattern as well. Also, with the transportation modes of a GPS track, people are facilitated to share life experience among each other in Web communities, and obtain more reference knowledge from others' trajectories. Users can know not only where other people have been but also how these people reach each location.

—For the service providers. Such knowledge enables the application systems to classify GPS tracks into different categories of transportation mode. Therefore, systems are capable of performing smart route recommendations/designs for a person based on the person's needs. For instance, a system should return a bus line rather than a driving route to an individual intending to move to somewhere by a bus.

Moreover, a transportation mode can feature many pervasive computing systems aiming to recognize human activities from GPS data. Such high-level activities would both enable the creation of new computing services that autonomously respond to a person's unspoken needs and support more accurate predictions about future behaviors, such as moving direction [Liao et al. 2004], destination [Krumm et al. 2003], and life pattern [Patterson et al 2003]. In turn, the knowledge learned from these works can be leveraged to further enhance many innovative local/mobile applications on the Web.

Unfortunately, the identification methods based on simple rules, such as a velocity-based approach, cannot handle this problem with great effect due to the following reasons: 1) people usually change their transportation modes during a trip, that is, a GPS trajectory may contain multiple modes. 2) The velocity of different transportation modes is usually vulnerable to traffic conditions and weather. Intuitively, in congestion, the mean velocity of driving would be as slow as riding a bike.

In this article, we aim to automatically infer transportation modes, including driving, walking, taking a bus, and riding a bicycle, from raw GPS logs based on supervised learning. It is a step toward recognizing human behavior and understanding user mobility for pervasive computing systems. Also, it is a step toward improving local/mobile applications on the Web and enhancing the connection between mobility and locality by mining knowledge from raw GPS data with minimal user efforts. The contributions of this work lie in the following three areas.

First, we propose a change point-based segmentation method. This method aims to partition each GPS trajectory into separate segments of different transportation modes, while maintaining a segment of one mode as long as possible. In addition, this segmentation method is capable of enhancing the reliability of our methodology facing the variable traffic conditions.

Second, from each segment, we identify a set of sophisticated features, such as direction change rate, velocity change rate, and stop rate. These features have few correlations with the velocity, hence are not affected by differing traffic conditions. These set of features can also be extended to other pervasive computing systems aiming to recognize human behavior and understand user mobility.

Third, we conduct a graph-based postprocessing algorithm to further improve the inference performance. In this algorithm, we mine the commonsense constraints of the real world and typical user behaviors on a location from user-generated GPS logs. Therefore, we are able to leverage this location-constrained knowledge as probabilistic cues, while maintaining our methodology being independent of an additional database of road networks or points of interests.

Overall, the advantages of our method over the related works include two parts. (1) Our method is independent of other sensor data like GSM signal and heart rate, and map information, for example, road networks and bus stops, etc. Thus, it is generic to be deployed in a broad range of Web applications. 2) The model learned from the dataset of some users can be applied to infer GPS data from others; that is, it is not a user-specific model.

The rest of this article is organized as follows. In Section 2, we justify the motivation of inferring transportation mode using two application scenarios. Section 3 first describes the framework of our approach, and then introduces each component of the proposed method in details. In Section 4, we conduct a set of experiments, which evaluate our approach based on a GPS dataset collected by 65 people over a period of 10 months. The major experimental results, as well as the corresponding discussion, are also reported here. Finally, after introducing some related works in Section 5, we draw our conclusion in Section 6.

2. APPLICATION SCENARIOS

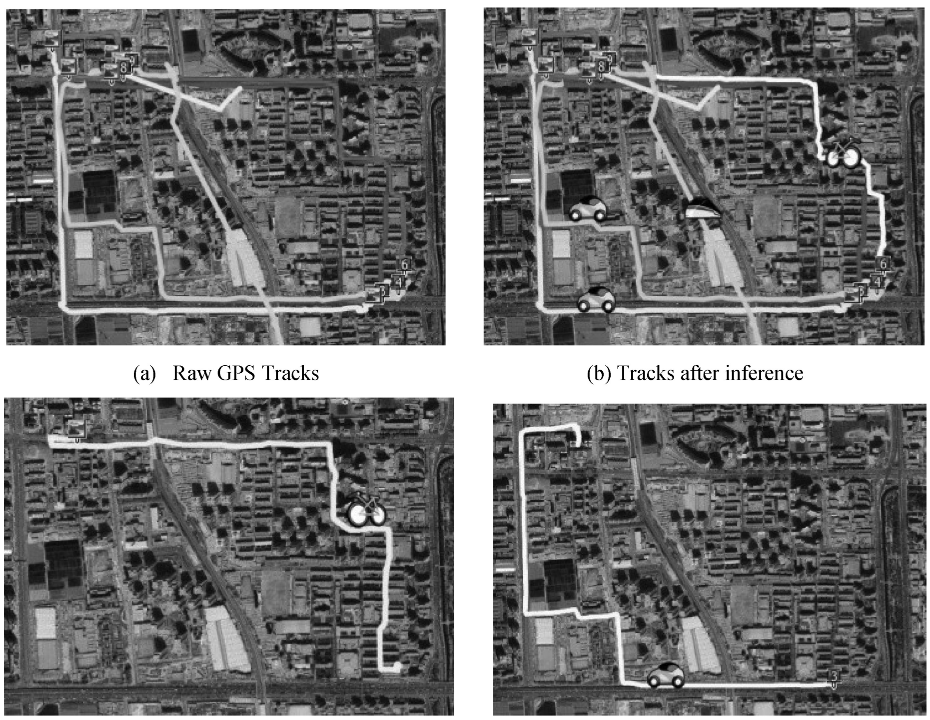
The work reported in this article is a part of research into our project GeoLife, which focuses on lively visualization [Zheng et al. 2008c, 2008d], fast retrieval [Wang et al. 2008] and a deep understanding [Zheng et al. 2008a, 2008b] of GPS track logs for both personal and public use. Our approach has been deployed in the website of GeoLife to automatically tag transportation modes to GPS tracks submitted by users. Leveraging the following two cases, we differentiate between the significance of a GPS track with and without the information of transportation modes.

2.1 Improving Sharing and Interactions between Users

Using a GPS trajectory generated by an individual, Figure 1 presents an example to distinguish the different Web experiences with and without transportation modes. Without the tag of transportation modes, the track shown in the Figure 1(a) provides us nothing but a ploy-line. However, as illustrated in Figure 1(b) and (c), after conducting inference, we realize that the individual first drives downtown, and then switches to walking at a parking lot. Based on this observation, a place where we are allowed to park a car was discovered, and how long we might spend on the way by driving was correctly suggested. Meanwhile, this track may also offer a reference experience to walking downtown from the parking lot. In this way, we are more likely to avoid heavy traffic, and enjoy shopping when passing the street side. Regardless of the fact that this track is a compound trajectory containing driving and walking, the mean velocity of the whole track would be quite slow given the relatively long duration consumed by walking. Therefore, the route might be deemed as a way that suffered from heavy traffic. In other words, we might ignore it when searching for an efficient way to drive downtown.

(a) Before inference (b) Stop at a parking lot (c) Switch to walking

Fig. 1. An example of inferring transportation modes from raw GPS data.



(a) Raw GPS Tracks (b) Tracks after inference

(c) A track generated by a user riding a bike (d) A track generated by a user driving

Fig. 2. Route recommendation according to users' preferences on transportation modes.

2.2 GPS Trajectories Classification

Using a set of GPS trajectories generated in the real world, Figure 2 further demonstrates the contribution of our work to route planning/recommendation systems. In Figure 2 (a), there are many route candidates for our selection when we attempt to find a way from the bottom right to the top left. Intuitively, people have various preferences on transportation modes when planning a travel route. For instance, some individuals like riding a bike, while others prefer

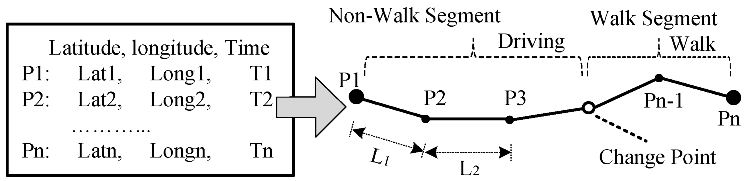


Fig. 3. GPS log, segment and change point.

driving or taking buses. Unfortunately, these routes are less discriminative from one another before being inferred. Actually, as shown in Figure 2(b), these routes are generated by different users taking different transportation modes, such as driving and riding a bike. Thus, if a user wants to ride a bike to their destination, we should recommend the route shown in Figure 2(c). Likewise, when the person needs to drive, we should present the route depicted in Figure 2(d).

3. INFER TRANSPORTATION MODES

In this section, we first define some preliminary concept about GPS data, and then give an overview on the framework of our methodology. Subsequently, the four steps, consisting of segmentation, feature extraction, inference and post-processing, of our approach are respectively described in detail.

3.1 Preliminary

Before describing the framework of our approach, we have to define the following terms: GPS log, GPS trajectory, *Walk Segment*, *non-Walk Segment,* and change point. Basically, as depicted in the left part of Figure 3, a GPS log is a sequence of GPS points, $p_i \in P$, $P = \{p_1, p_2, \ldots, p_n\}$. Each GPS point p_i contains latitude, longitude and a timestamp. On a two dimensional plane, we can sequentially connect these GPS points into a GPS trajectory, and then divide the GPS trajectory into trips if the time interval between the consecutive points exceeds a certain threshold. A change point stands for a place where a user changes their transportation mode in a trajectory. For instance, in the right part of Figure 3, a change point was generated when an individual transferred from driving to walking.

For simplicity's sake, we name the segments users traveled on foot as *Walk Segments,* while the segments of other transportation modes are called *non-Walk Segments*. Further, we call the GPS point from a *Walk Segment*, such as P_{n-1}, a *Walk Point,* while the GPS points like P_2 from *non-Walk Segments* are coined in *non-Walk Points*. In short, as depicted in Figure 3, a trip can be partitioned into a *Walk Segment* and a *non-Walk Segment* by a change point where the user transfers from driving to walking. In the remainder of this article, we use {*Bike, Bus, Driving, Walk*} to respectively represent the following four kinds of transportation modes: riding a bike; taking a bus; driving or taking a taxi; traveling on foot.

Figure 4 depicts how we calculate features from GPS logs. Given two consecutive GPS points, for example, p_1 and p_2, we can calculate the spatial distance L_1, temporal interval T_1 and heading direction ($p_1.head$) between them.

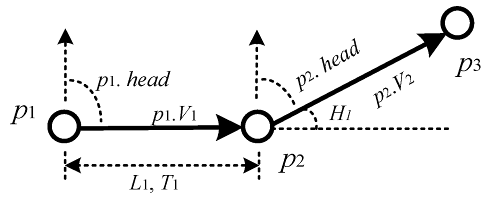


Fig. 4. Feature calculation based on GPS logs.

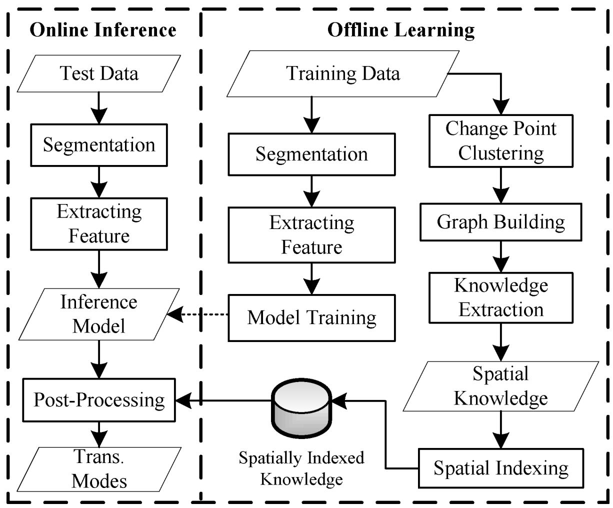


Fig. 5. Architecture of our approach.

Subsequently, the velocity of p_1 can be computed as Equation (1).

$$p_1.V_1 = L_1/T_1. \tag{1}$$

Then, the heading change, such as H_1,of three consecutive points like p_1, p_2 and p_3 can be calculated as Equation (2).

$$H_1 = |p_1.head - p_2.head|. \tag{2}$$

Further, more features, such as acceleration and expectation of velocity, can be calculated in this manner.

3.2 Architecture of Our Approach

As shown in Figure 5, the architecture of our approach includes two parts, offline learning and online inference.

In the offline learning section, on one hand, we first partition GPS trajectories into segments based on change points and extract features from each segment. Then, the features and corresponding ground truths are employed to train a classification model for online inference.

On the other hand, using a density-based clustering algorithm, we group the change points detected from all users' GPS logs into clusters. Subsequently, a graph based on these clusters and user-generated GPS trajectories is built.

Table I. Transition Matrix among Transportation Modes

Transportation Modes	Walk	Driving	Bus	Bike
Walk	/	41.0%	49.0%	9.0%
Driving	99.7%	/	0%	0.3%
Bus	98.7%	0.8%	/	0.5%
Bike	99.8%	0%	0.2%	/

From this graph we can mine some location-constrained knowledge, such as the probability distribution of different transportation modes on each edge. The knowledge can be employed as probabilistic cues to improve the inference performance in the postprocessing. In addition, a spatial index is built over the detected spatial knowledge to enhance the processing efficiency.

In the online operation, when a GPS trajectory comes, like the offline training process, we first partition it into segments and extract the same features from each segment. Second, given the features, the generative inference model will predict the transportation mode of each segment in a probabilistic manner. Third, given the probabilities of a segment being different transportation modes, a post-processing algorithm is used to improve the inference accuracy by leveraging the spatial knowledge mined from the training data. Finally, the transportation mode with maximum posterior probability will be selected as the ultimate result.

3.3 Change Point-Based Segmentation

In this section, we will demonstrate how change points can be detected automatically from a given GPS log. This detecting approach is derived from the following commonsense knowledge of the real world, and is justified by the data shown in Table I.

—Walking should be a transition between different transportation modes. In other words, the start point and end point of a *Walk Segment* could have a very high probability to be a change point.

—Typically, people must stop and then go when changing their transportation modes, i.e., there must be some GPS points whose velocities are close to zero during a transition.

Therefore, we first retrieve *Walk Segments* from a GPS trajectory, and then partition the trajectory into several portions with these *Walk Segments*.

Table I shows the transition matrix among different transportation modes. This matrix was summarized from the ground truth of the GPS data collected by 65 people over a period of 10 months. In almost all of the cases, *Driving*, *Bus* and *Bike* transfer to *Walk* before changing to one another. On a few occasions, a person might take a taxi immediately after getting off a bus. When the person labels the GPS data, the very short *Walk Segment* between these two transportation modes is easy to neglect. That is the cause of the direct transition between *Driving* and *Bus*. However, a *Walk Segment* essentially exists in this situation although its distance is quite short.

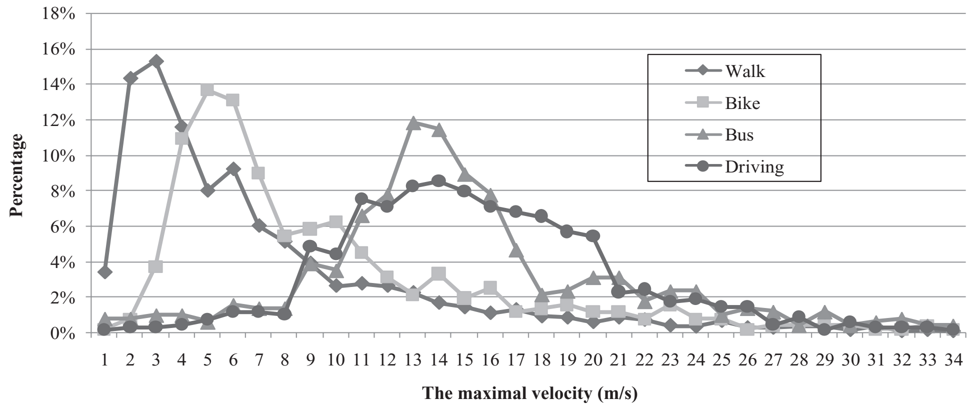


Fig. 6. The distribution of the maximum velocity of a segment.

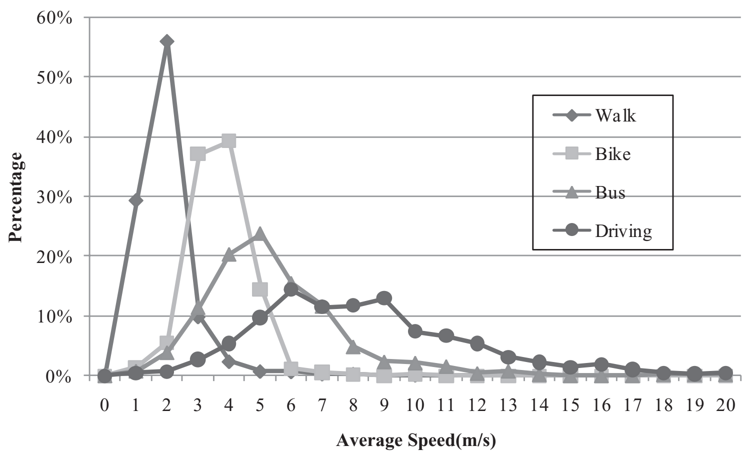


Fig. 7. The distribution of the average speed of a segment.

However, over the same dataset, Figures 6, 7, and 8 respectively show the distribution of maximum velocity, average velocity and maximum acceleration of different transportation modes. The data shown in these figures paints a rich picture about how difficult it is to give some simple rules to directly distinguish between the segments of different transportation modes. Without knowing how many modes a trip contains, it is especially difficult to tackle the problem using simple rules.

Enlightened by the previously-mentioned commonsense knowledge as well as the information mined from GPS data, we first find the change points by detecting *Walk Segments* from a trip. Then, using these change points, we are able to partition the trip into alternate *Walk Segments* and non-*Walk Segments*. Since segments from a trip are first categorized into two classes {*Walk*, *non-Walk*} rather than four classes {*Bike, Bus, Driving, Walk*}, the complexity of segmentation has been reduced greatly. Subsequently, we can extract the features of each segment and further infer its specific transportation mode. Using

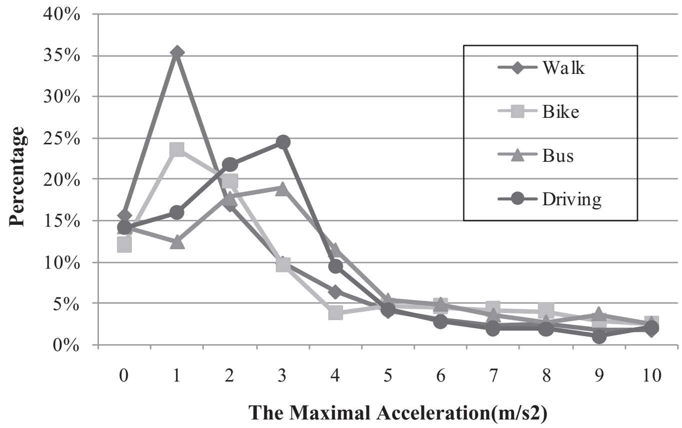


Fig. 8. The distribution of the maximum acceleration of a segment.

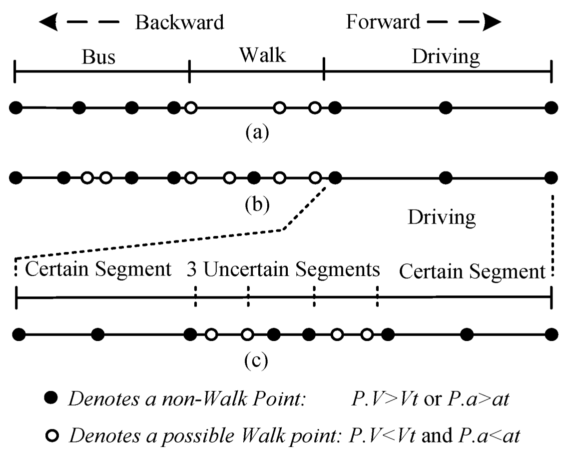


Fig. 9. An example of detecting change points.

Figure 9 as an example, in which an individual transfers from *Bus* to *Driving* using *Walk* as a transition, we describe the detecting procedure as follows.

—*Step* 1. Using a loose upper bound of velocity (V_t) and that of acceleration (a_t) to distinguish all possible *Walk Points* from *non-Walk Points*.

—*Step* 2. If the distance or time span of a segment composed by consecutive *Walk Points* or *non-Walk Points* less than a threshold, merge the segment into its backward segment.

—*Step* 3. If the length of a segment exceeds a certain threshold, the segment is regarded as a *Certain Segment*. Otherwise it is deemed as an *Uncertain Segment*. If the number of consecutive *Uncertain Segments* exceeds a certain threshold, these *Uncertain Segments* will be merged into one *non-Walk Segment.*

—*Step* 4. The start point and end point of each *Walk Segment* are potential change points, which are used to partition a trip.

As depicted in Figure 9, each possible *Walk Point* (white points) is a GPS point whose velocity ($P.V$) and acceleration ($P.a$) are both smaller than the given bound. Ideally, as demonstrated in Figure 9(a), only one *Walk Segment* would be detected from this trip. However, as illustrated in Figure 9(b), when a vehicle moves slowly in some transient occasions, a few GPS points from *non-Walk Segments* may be detected as possible *Walk Points*. Also, because of the locative error, a few points from the *Walk Segment* will exceed the bound, and become *non-Walk Points* (black points in Figure 9). To improve the precision of the segmentation method, we require that the distance of each retrieved segment must exceed a certain distance. Otherwise, it will be merged into its backward segment. For instance, in Figure 9(b), the two *Walk Points* contained in the *Bus* segment cannot construct a stand-alone segment due to the short distance between them. Therefore, it will be merged into its backward segment. The same criterion is also applied to handle the outlier points in *Walk Segment*.

After step 1 and step 2 are conducted, the trip is partitioned into a series of alternate *Walk Segments* and non-*Walk Segments*. As demonstrated in Figure 9(c), unfortunately, in some occasions, in which a user meets congestion or heavy traffic, a *Driving* segment may be composed of many alternate *Walk Segments* and *non-Walk Segments*. It is not appropriate for the classification model to conduct inference using the features extracted from such trivial segments. Intuitively, the longer a segment is the more accurate features a segment might express. Thus, we are more likely to infer its transportation mode correctly. However, the shorter a segment is, the higher the uncertainty might be.

To avoid a trivial partition, which will lead to further inference errors, we take the following policy to merge, to some extent, the consecutive *Uncertain Segments*. We define a segment whose length exceeds a threshold (e.g., 200 meters used in the experiments) as a *Certain Segment*. Otherwise, we deem it as an *Uncertain Segment*. In other words, we are not sure about the transportation mode of this segment even if it holds the condition of a *Walk Segment*. If the number of consecutive *Uncertain Segments* exceeds a certain threshold, for example, two we find out in experiments, we still deem all these *Uncertain Segments* as one *non-Walk Segment*. It is not difficult to understand that common users will not frequently change their transportation modes within such a short distance. For instance, as depicted in Figure 9(c), within a certain distance, it is impossible for a person to take the following transition, *Driving* → *Walk* → *Driving* → *Walk* → *Driving*. So, we believe the three segments between the two *Certain Segments* are also non-*Walk Segments*, *Driving* here. Thus, we are able to merge these *Uncertain Segments* into one segment and perform a further inference. By maintaining the consecutive GPS points of the same transportation mode in one segment, we are more likely to reduce the affection caused by the congestion.

3.4 Feature Extraction

Table II shows the features we extracted from each segment. Two categories of features including *Basic Features* and *Advanced Features* are identified.

Table II. The Features We Explored in the Experiment

Category	Features	Significance
Basic Features	Dist	Distance of a segment
	MaxVi	The ith maximal velocity of a segment
	MaxAi	The ith maximal acceleration of a segment
	AV	Average velocity of a segment
	EV	Expectation of velocity of GPS points in a segment
	DV	Variance of velocity of GPS points in a segment
Advanced Features	HCR	Heading Change Rate
	SR	Stop Rate
	VCR	Velocity Change Rate

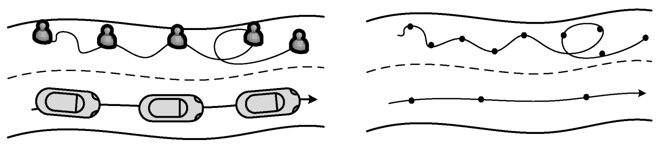


Fig. 10. Heading change rate of different modes.

Following the way we presented in Figure 3 and Figure 4, the *Basic Features* are easily calculated, while the *Advanced Features* is described in the following sections. Beyond the *Basic Features*, the *Advanced Features* is more robust to variable traffic conditions.

3.4.1 *Heading Change Rate (HCR).* As shown in Figure 10, typically, being constrained by a road, people driving a car or taking a bus cannot change their heading direction as flexibly as if they are walking or cycling, no matter what traffic conditions they meet. Moreover, regardless of traffic conditions and the weather, people walking or riding a bicycle inevitably and unintentionally wind their way to a destination, although they attempt to create a straight route.

In other words, the heading directions of different transportation modes differ greatly in being constrained by the real routes while being independent of traffic conditions. Thus, *HCR* modeling this principle is defined as Equation (3).

$$HCR = |P_c|/Distance, \tag{3}$$

where P_c stands for the collection of GPS points at which a user changes his/her heading direction exceeding a certain threshold (Hc). $|P_c|$ represents the number of elements in P_c. After dividing $|P_c|$ by the distance of the segment, *HCR* can be regarded as the frequency that people change their heading direction to some extent within a unit distance.

3.4.2 *Stop Rate (SR).* Figure 11 presents the typical trend of velocity when people take different transportation modes. We observe that, within a similar distance, people taking a bus are likely to stop more times than driving. Intrinsically, besides waiting at traffic lights, a bus would take passengers on or off at bus stops. Meanwhile, people walking on a route would become more likely than they would in other modes to stop somewhere on their journey, such as when talking with passersby, being attracted by surrounding environments, waiting for a bus. These observations motivate us to define two features to differentiate

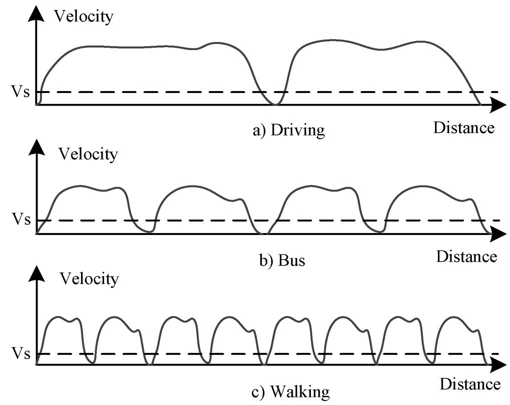


Fig. 11. Velocity change rate and stop rate.

various transportation modes; one is the stop rate (*SR*); the other is the velocity change rate (*VCR*).

The *SR* stands for the stop frequency of a moving object within a unit distance. As depicted in Figure 11, using a velocity threshold V_s, we can detect several groups of GPS points (*PS*), where $PS = \{P_{s1}, P_{s2}, \ldots, P_{sn}\}, PS_{si} = \{P_i | p_i \in P, p_i.V < V_s\}$. In each group of points (P_{si}), the velocity of each GPS point (p_i) is smaller than V_s. Then, we can calculate the *SR* as Equation (4).

$$SR = |PS|/Distance. \tag{4}$$

Obviously, as shown in Figure 11, we can see *SR* (*Walk*) > *SR* (*Bus*) > *SR* (*Driving*).

3.4.3 *Velocity Change Rate (VCR).* The other hint we sensed from Figure 11 is the *VCR*. First, we can calculate the *VRate* of each GPS point as Equation (5). Then, we can get the statistics of the number of GPS points whose *VRate* is greater than a certain threshold V_r, and calculate *VCR* according to Equation (6).

$$p_1 \cdot VRate = |V_2 - V_1|/V_1; \tag{5}$$

$$VCR = |P_v|/Distance, \tag{6}$$

where $P_v = \{p_i | p_i \in P, p_i \cdot VRate > V_r\}$. Consequently, we can understand the *VCR* as the number of GPS points, whose velocity change percentage over its prior point exceeds a certain threshold, within a unit distance. *SR* and *VCR* clearly capture the difference among various transportation modes, and gets support from later experimental results.

3.5 Inference Model

Given the features X of each segment, we are able to infer its transportation mode based on a supervised learning method. Using a training corpus, which contains a set of features-class pairs, we can train an inference model to classify the coming segments in the future. The training dataset was collected by multiple users over a long period. Thus, the model we built is generic to

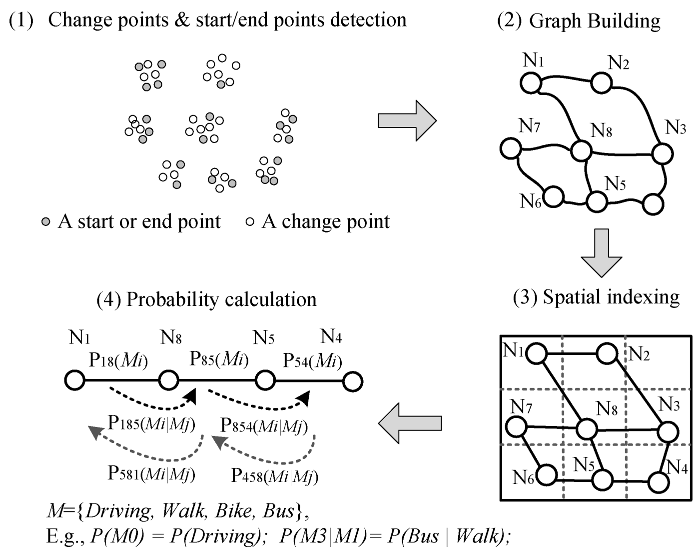


Fig. 12. Mining spatial knowledge from GPS logs.

infer the GPS trajectories for a variety of people. A group of classification algorithms, including a Decision Tree, a Support Vector Machine, a Bayesian Net, and a Conditional Random Field, have been tested in our previous experiments [Zheng et al. 2008a]. The results show that the Decision Tree outperforms others based on the change point-based segmentation. Consequently, we still use this inference model in this article. Meanwhile, a Bootstrap aggregating (bagging) [Breiman 1996] is employed as a meta-algorithm to improve the accuracy of the model by reducing variance and overfitting.

3.6 Spatial Knowledge Extraction

Figure 12 illustrates the four steps toward mining spatial knowledge from users' GPS logs. The knowledge includes a change point-based graph and the probability distribution on each edge of the graph.

First, given GPS logs with labeled ground truths, we can get the special points consisting of change points and the start/end points of each GPS trajectory. These special points were subsequently grouped into several nodes (clusters) using a density-based clustering algorithm. The reasons we prefer to use density-based clustering instead of agglomerative methods, such as K-Means, lie in two aspects. First, a density-based approach is capable of detecting clusters with irregular structures, which may stand for bus stops or parking places. Second, it can discover popular places where most people change their transportation modes while removing sparse change points representing places with low access frequency.

Second, with the GPS trajectories from multiple users' GPS logs, we can construct an undirected graph. In such a graph, a node represents a cluster of the special points mentioned above, and an edge denotes users' transitions between two nodes. Here, we do not differentiate various trajectories with similar start/end points; that is, all the trajectories passing two graph nodes are regarded as similar trajectories.

Third, we build a grid-based spatial index [Sahr et al. 2003] over the graph to improve the efficiency of accessing the information of each node and each edge. The space covered by the graph is partitioned into many disjoint grids. Then, the graph nodes falling in different grids are associated with the grids. Therefore, when a new GPS trajectory comes to be inferred, we only need to match the special points detected from the trajectory against the graph nodes pertaining to the grids where these special points falling in. Of course, this step is optional unless the scale of the GPS dataset is quite large.

Fourth, we are able to calculate the probability distribution of different transportation modes on each edge. For instance, as depicted in the fourth step of Figure 12, $P_{18}(Bus)$ stands for the likelihood of the event that people take buses on the edge between node 1 and node 8. Further, the conditional probability between different transportation modes can also be computed based on the graph, for example, $P_{185}(Bus|Walk)$ represents the transition probability from *Walk* to *Bus* between edge 18 and edge 85. In other words, it denotes the likelihood of the event that a user walks from node 8 to node 5 based on the observed occurrence that the user takes a bus from node 1 to node 8.

Such previously mentioned knowledge is promising in improving the inference accuracy due to the following reasons. 1) This implies people's typical behaviors among different places. The clusters of change points represent the places many people change their transportation modes. Usually, these places could be bus stops, parking lots, and railway stations. We can take into account user behavior among these nodes as probabilistic cues when we infer other trajectories passing these two nodes. Naturally, for example, if most users take a bus between two nodes, we can suggest that the two nodes could be bus stops, and the edge between them could have a very high probability of being a bus line. 2) The probability on each edge implies constraints of the real world. For instance, buses only take passengers on at bus stops, cars are left in parking lots, and cars and buses only travel on streets, etc. This knowledge mined from multiple users' GPS logs take advantages of the location constraints while keeping our method independent of an additional map database. In such a way, it is not necessary to match each GPS trajectory against the road network. Meanwhile, we do not need to maintain a database of bus stops, railway stations and parking lots.

3.7 Graph-Based Postprocessing

The graph-based postprocessing algorithm takes the preliminary inference result and the spatial knowledge mentioned above as inputs, and aims to generate an improved prediction result. The graph-based postprocessing is comprised of three components: normal post-processing, prior probability-based enhancement and transition probability-based enhancement. The main idea of the postprocessing lies in using the correctly inferred segments to revise the false predictions.

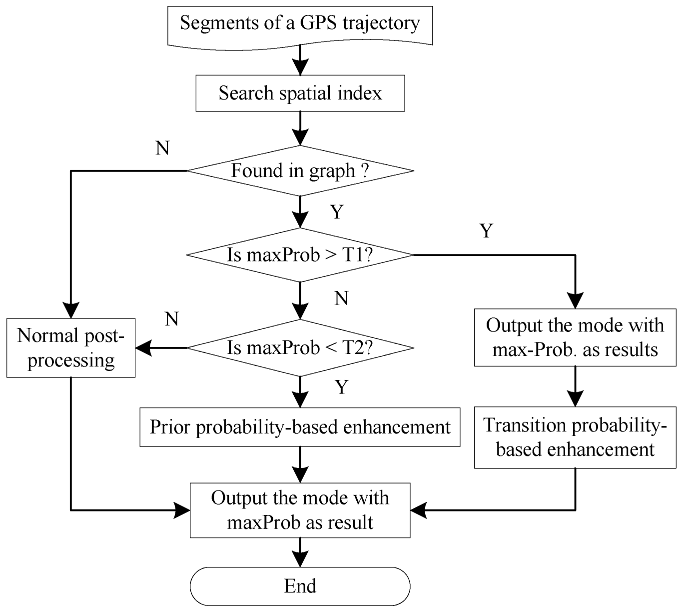


Fig. 13. Flowchart of the graph-based postprocessing.

3.7.1 *Framework of the Postprocessing.* Figure 13 shows the flowchart of the graph-based post-processing we designed. When an inferred segment of GPS trajectory appears, we first search the spatial index to quickly match its vertexes against graph nodes. If we cannot find graph nodes close to the vertex, the normal postprocessing algorithm is employed to enhance the inference performance. Otherwise, the prior probability-based enhancement or the transition probability-based method would be used. If the maximum posterior probability of the segment being a kind of transportation given feature X exceeds a certain threshold T_1, we believe the transportation mode corresponding to the maximum probability would be a correct inference. Subsequently, we can leverage this segment to revise its adjacent segments (including backward and forward segments) using the transition probability-based enhancement. If the situation is opposed to the above assumption, we will further check whether its maximum probability is smaller than another threshold T_2. If the condition holds, the prior probability-based enhancement method is performed. Otherwise, the normal postprocessing method is used. When a segment holds the conditions being processed by both the prior probability-based and the transition probability-based enhancement approaches, the latter is used. Finally, we select the transportation mode with maximum probability as the prediction result of a segment.

The idea of postprocessing is motivated by the observation that when a segment's maximum $P(m_i|X)$ exceeds a threshold (T_1 in Figure 13), it is very likely to be a correct prediction. Thus, it can be used to fix the potentially false inference adjacent to it in a probabilistic manner. Instead, if the maximum posterior probability of a segment is less than a certain threshold T_2, it could have

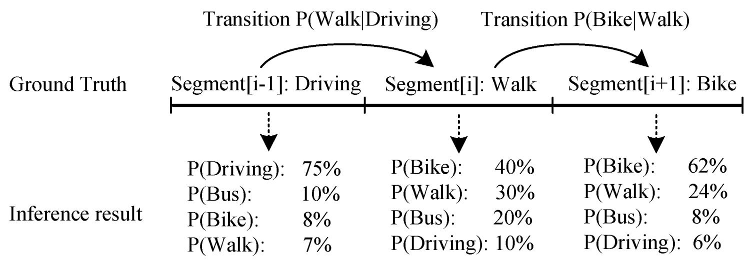


Fig. 14. Performing normal post-processing on a GPS trajectory.

a very high probability of being a false inference. Consequently, it deserves our revision. With the threshold T_1 and T_2, we are more likely to correct the false prediction while maintaining accurate inference results. (See Figure 29 for evidences.)

3.7.2 *Normal Postprocessing.* Normal post-processing aims to improve the prediction accuracy by considering the conditional probability between different transportation modes. Using the trajectory depicted in Figure 14 as an example, we introduce the normal post-processing algorithm. After implementing the preliminary inference, we can get the predicted posterior probability of each segment being different transportation modes given feature X. If we directly select the transportation mode with maximum posterior probability as the final result, the prediction would be *Driving→Bike→Bike*, while the ground truth is *Driving→Walk→Bike*; that is, a prediction error occurred. On this occasion, if a segment, for example, segment$[i-1]$, whose posterior probability being a kind of transportation mode exceeds threshold T_1 (0.6 used in our experiment), we select the transportation mode as the final prediction. Later, if the probability of its adjacent segments, for example, segment$[i]$, is less than a threshold T_2, the inference of segment$[i-1]$ can be used to revise the prediction of segment$[i]$. Therefore, the posterior probability of segment$[i]$ being different transportation modes conditioned by the transportation mode of segment$[i-1]$ can be re-calculated according to Equations (7) and (8).

$$Segment[i].P(Bike) = Segment[i].P(Bike) \times P(Bike|Driving), \tag{7}$$
$$Segment[i].P(Walk) = Segment[i].P(Walk) \times P(Walk|Driving) \tag{8}$$
$$\cdots\cdots$$

Here $P(Bike|Driving)$ stands for the probability of an event that a person directly transfers transportation modes from driving to riding a bike. Likewise, $P(Walk|Driving)$ denotes the transition probability from *Driving* to *Walking*. As shown in Table I, these probabilities can be summarized from the user-labeled data. $Segment[i].P(Bike)$, which represents the probability of the segment$[i]$ being a *Bike* segment, is the output of the inference model.

After the calculation, we use the transportation mode with maximum probability as the final result. In the case depicted in Figure 14, since the transition probability from *Driving* to *Bike* is very small, $Segment[i].P(Bike)$ will drop behind $Segment[i].P(Walk)$ after the multiplications shown in

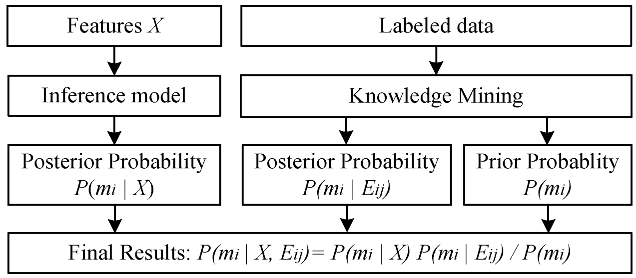


Fig. 15. Prior probability-based enhancement method.

Equations (7) and (8). In other words, the inference result of *Bike* will be substituted with *Walk*. Consequently, a false inference is revised based on the correct prediction of its adjacent segment. The normal postprocessing is performed in a propagated manner until the conditions (T_1 and T_2) are not held any longer.

3.7.3 *Prior Probability-Based Enhancement on Graph.* When a segment is detected to pass two graph nodes (i, j), we can use the prior probability distribution on the corresponding edge (E_{ij}) to reduce the incorrect prediction. Actually, what we want to predict is the posterior probability of transportation mode (m_i) on E_{ij} based on the observed feature X, i.e., $P(m_i|X, E_{ij})$. Since E_{ij} is involved in the condition part, the probability is more discriminative and informative than $P(m_i|X)$ used in the preliminary inference model. As shown in Equation (9), by applying the Bayesian rule, we decompose $P(m_i|X, E_{ij})$ into prior probability and the likelihood of seeing (X, E_{ij}) given transportation mode m_i. Then, using the assumption that X is independent of E_{ij}, we can further decompose $P(X, E_{ij}|m_i)$ into $P(X|m_i)$ and $P(E_{ij}|m_i)$.

$$\begin{aligned} P(m_i|X, E_{ij}) &= \frac{P(X, E_{ij}|m_i)P(m_i)}{P(X, E_{ij})} \\ &= \frac{P(X|m_i)P(E_{ij}|m_i)P(m_i)}{P(X)P(E_{ij})} \\ &= \frac{P(m_i|X)P(X)}{P(m_i)} \cdot \frac{P(m_i|E_{ij})P(E_{ij})}{P(m_i)} \cdot \frac{P(m_i)}{P(X)P(E_{ij})} \\ &= \frac{P(m_i|X) \cdot P(m_i|E_{ij})}{P(m_i)}. \end{aligned} \tag{9}$$

Figure 15 presents how each element of Equation (9) is calculated after the transformation. As we can see, $P(m_i|E_{ij})$ and $P(m_i)$ can be summarized from multiple users' datasets while $P(m_i|X)$ is difficult to be directly calculated since the elements of X may not be independent of each other. So, in this work, we use the posterior probability generated by the preliminary inference model as an approximate substitute. From the theoretic perspective, we would face two challenges in computing $P(m_i|X, E_{ij})$. One is the assumption of the independence between X and E_{ij}. The other lies in that $P(m_i|X)$ is substituted by an approximate value generated by an inference model. Thus, we need to use it

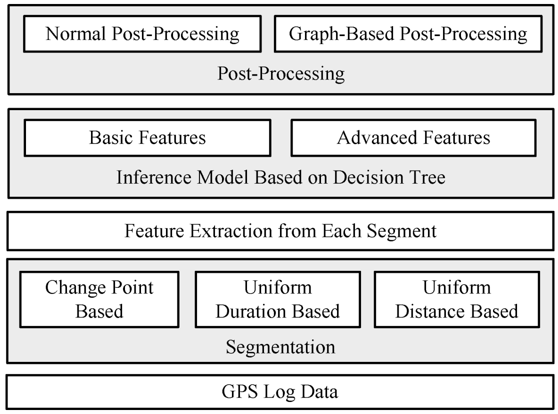


Fig. 16. Framework of our experiments.

carefully to ensure the effectiveness of this approximate calculation. That is another reason why we need threshold T_1 and T_2.

3.7.4 *Transition Probability-Based Enhancement on the Graph.* This process is performed only when the following three conditions hold: 1) we find consecutive segments on the graph; 2) One segment's $P(m_i|X)$ exceeds the threshold T_1; 3) The maximum $P(m_i|X)$ of its adjacent segments is less than T_2. The process is similar to the normal post-processing algorithm while the difference is that the transition probability between different transportation modes is based on the graph. In other words, the probability is location-constrained and contains more commonsense information of the real world. Therefore, it is more useful and informative than the normal transition probability summarized from all users' ground truth regardless of locations.

4. EXPERIMENTS

In this section, we first describe the framework of the experiments we performed. Second, we present the experiment setup including GPS devices, GPS data, and toolkits we used. Third, the evaluation approach, including how we get ground truths and what criteria we used are reported. Fourth, we respectively introduce how the parameters are selected for each algorithm. Finally, we report detailed experimental results with corresponding discussions.

4.1 Framework of Experiments

Figure 16 shows the framework of the experiments. Here, we focus on evaluating the effectiveness of the change point-based segmentation method, exploring the performance of the *Advanced Features* and testing the effect of the graph-based post-processing.

4.1.1 *Segmentation.* To validate the effectiveness of the change point-based segmentation, two baseline methods are selected to partition the trips into segments. They are uniform duration-based segmentation and uniform

Fig. 17. GPS devices used in our experiments.

distance-based segmentation. In other words, each segment will have the same time spans after being partitioned by the former method or have the same distances after being partitioned by the latter one.

4.1.2 *Inference.* Four inference models, including Decision Tree, Supported Vector Machine, Bayesian Net, and Conditional Random Field, are studied in our previous experiments reported in Zheng and Liu [2008]. Based on the change point-based segmentation method, Decision Tree outperforms other competitors by offering higher inference accuracy. Therefore, in this article, we aim to investigate the effectiveness of the *Advanced Features* we identified, and make a difference between the advanced features and the basic ones.

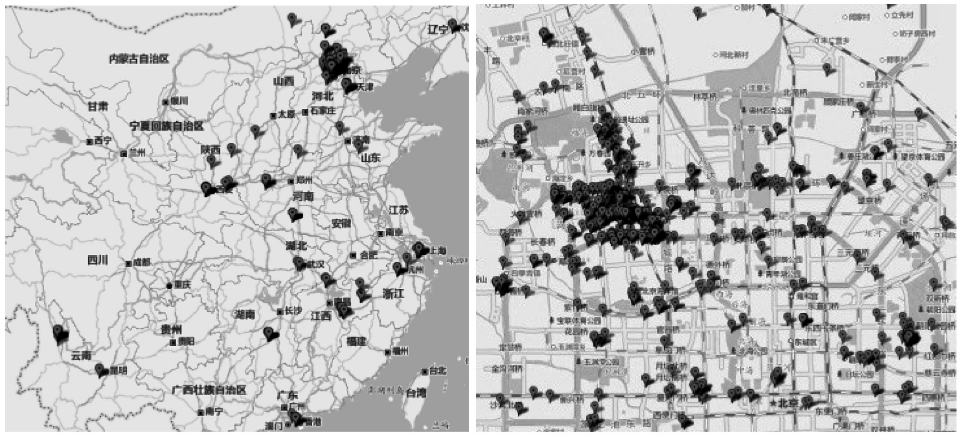
4.1.3 *Postprocessing.* In this step, we aim to evaluate the effectiveness of graph-based postprocessing, and compare it with the normal post-processing algorithm. At first, we group multiple users' change points using OPTICS [Ankerst et al 1999], a density-based clustering algorithm, which can detect data clusters with irregular structures. Further, a graph is built based on these clusters and GPS trajectories. Then, over each graph edge, we are able to calculate the probability distribution of different transportation modes and the transition probability among them.

4.2 Settings

4.2.1 *GPS Devices.* Figure 17 shows the GPS devices we chose to collect data. They are composed of stand-alone GPS receivers (Magellan Explorist 210/300, G-Rays 2 and QSTARZ BTQ-1000P) and GPS phones. Except for the Magellan 210/300, these devices are set to receive GPS coordinates every two seconds. Regarding the Magellan devices, we configure their setting to record GPS points as densely as possible. When an individual changes his/her heading direction or speed to some extent, a GPS point is recorded with such devices.

4.2.2 *GPS Data.* Figure 18 shows the distribution of the GPS data we used in the experiments. Carrying a GPS-enabled device, 65 users recorded their outdoor movements with GPS logs over a period of 10 months. The dataset covers 28 big cities in China and some cities in the USA, South Korea, and Japan. We pay each data collector based on the distance of GPS trajectories they collected and labeled, and use this dataset anonymously.

As shown in Table III, the total distance of these GPS logs exceeds 30,000 kilometers, and their total duration equates to more than 2,000 hours. From



(a) Data distribution in China (b) Data distribution in Beijing

Fig. 18. Distribution of GPS data.

Table III. The Information of Each Activity in the GPS Dataset We Used in the Experiments

Transportation Modes	Number of Trajectory	Distance (km)	Duration (h)
Walk	3037	2877.52	769.8
Bike	1306	4287.53	395.6
Bus	1145	5738.65	313.6
Driving	1624	17169.85	557.2
Total	**7112**	**30073.55**	**2036.2**

each user's GPS logs, we select about 70 percent of the data to construct a training set while the rest is used as a test dataset. Meanwhile, in both the training set and the test set, we try to keep the number of segments of different transportation modes as balanced as possible. When processing the dataset, each GPS log is divided into trajectories if the temporal interval between consecutive GPS points exceeds 20 minutes. Further, each GPS trajectory will be partitioned into segments of different modes based on the change point-based method.

4.3 Evaluation Approach

4.3.1 *Criteria.* When exploiting the performance of our approach, we consider the following two aspects of criterions: the accuracy of inferred transportation modes and the accuracy of the detected change points.

Criterion for transportation modes. With regard to the prediction accuracy of transportation modes, we focus on *Accuracy by Segment* (A_S) and *Accuracy by Distance* (A_D), which are defined in equation (10) and (11) respectively.

$$A_S = m/N : \quad (10)$$

$$A_D = \frac{\sum_{j=0}^{m} CorrectSegment\ [j].Distance}{\sum_{i=0}^{N} Segment\ [i].Distance}; \quad (11)$$

where N stands for the total number of the segments, while m denotes the number of segments being correctly predicted. Intuitively, it is more important to correctly infer a segment with long distance than that with short distance. So, A_D is more objective to measure the inference accuracy.

Criterion for change points. Regarding the inference accuracy of change points, we explore their recall and precision in two stages. In the first stage, after being partitioned by the change point-based segmentation method, a set of change points are detected from each trajectory. At this moment, the change points are vertexes of the estimated *Walk-Segments*. Here, we can use the criterion to measure the effectiveness of the segmentation method. In the second stage, we evaluate the precision and recall of change points after the inference process. At this moment, a change point occurs if the inferred transportation modes of consecutive segments are different.

If the distance between an inferred change point and its ground truth is within 150 meters, we regard the change point as a correct inference. Intrinsically, if a change point is missed in a segmentation process, two segments of different transportation modes will be deemed as one segment of the same mode. Therefore, a false inference will definitely occur no matter what kind of inference model we employed later. However, even if a trajectory containing only one transportation mode is carelessly partitioned into several segments, we still have chances to infer these separated segments correctly. Later, these segments having the same inference results can be merged into one trajectory. However, in the segmentation phase, a very poor precision of change points will cause too much trivial segments with short distance. Subsequently, the short distance will damage the inference accuracy of transportation modes of a segment. At the same time, a very low precision of change points will bring bad user experiences. Consequently, we claim that although the recall of a change point has relatively higher priority over the precision, we still need to keep the balance between them.

4.3.2 *Ground Truth.* In each day of the data collection, to help data creators manually label their GPS trajectories, we respectively visualize each trace on a map for each creator. Therefore, these data-creators can view the timestamp of each GPS point, and reflect on when and where they change transportation modes. Given the short time span between data created and data labeled, we believe users are able to accurately label their GPS trajectory based on their reliable memory and the visualized geographic clues.

Figure 19 presents a case in which a user travels on foot from 20:30:15 to 20:35:56, and then transfers to driving at 20:35:58. The GPS trajectory can be labeled in a manner of "DateTime1-DateTime2 transportation mode". In this case, the labeled ground truth should be

"*2008-06-22 20:30:15 to 2008-06-22 20:35:56 Walk*";
"*2008-06-22 20:35:58 to 2008-06-22 20:55:28 Driving*."

Later, a change point is easy to be detected based on such labels.

Fig. 19. A case showing how a GPS trajectory was labeled by its creator.

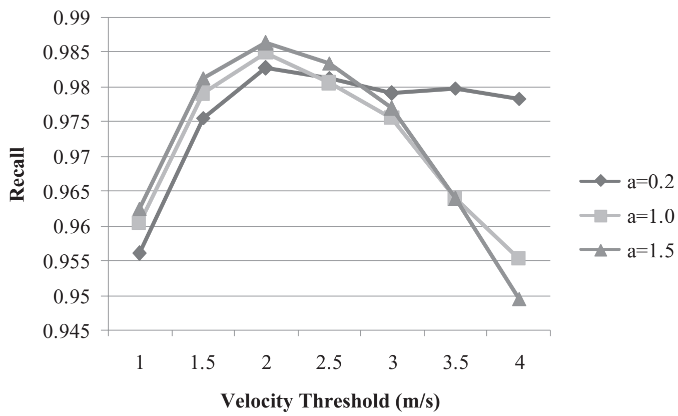


Fig. 20. Recall of the change points after conducting the first step of segmentation.

4.4 Parameter Selection

4.4.1 *Change Point-Based Segmentation.* The change point-based segmentation method aims to accurately partition a GPS trajectory into segments of different transportation modes while maintaining each segment as long as possible. In the following paragraphs, we study the performance of our segmentation method step by step, choose proper parameters for each step and justify the significance of each step.

Performance of Step 1. Figure 20 and Figure 21 respectively show the recall and precision of detected change points after we conduct the first step of the segmentation method. Here, we attempt to retrieve all the *Walk Segments* using a looser upper bound. A set of possible values of velocity within [1, 4] m/s have been studied, and three values of acceleration (0.2, 1.0, 1.5) m/s^2 have been tested. As a result, when v = 2.5 m/s and a = 1.5 m/s^2, we can obtain the relatively high recall with an acceptable precision. In the first step, the recall is slightly more important than the precision as we still have chances to improve the precision in the following steps.

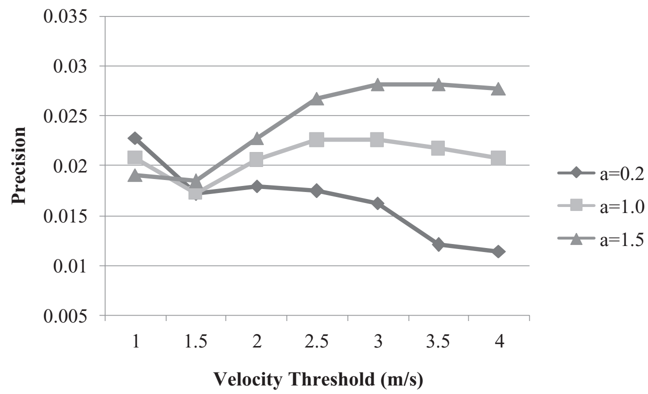


Fig. 21. Precision of the change points after conducting the first step of segmentation.

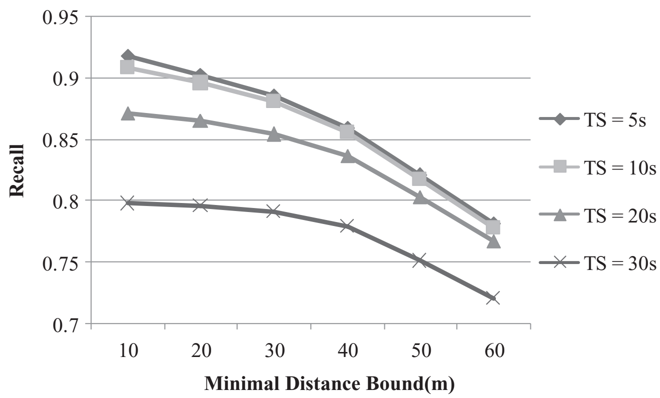


Fig. 22. Recall of the change points after conducting the second step of segmentation.

Performance of Step 2. After being processed by the first step of our method, a GPS trajectory is partitioned into many *Walk Segments* and *non-Walk segments*. Two parameters, minimal distant bound (*MDB*) and minimal time span (*TS*) of a segment, are employed in the second step to improve the precision of the segmentation by merging these trivial segments. From the data depicted in Figure 22 and Figure 23, when $MDB = 20$ meters and $TS = 10$ seconds, we obtain a relatively high recall with an acceptable precision of detected change points. In other words, if the distance of a segment is smaller than 20 meters or the time span between its start time and end time is less than 10 seconds, this segment will be merged into its backward segments.

Performance of Step 3. Figure 24 and Figure 25 present the segmentation performance of the third step of our method. In this step, if the number of consecutive *Uncertain Segments* exceeds a threshold *SN*, these *Uncertain Segments* will be merged into one segment. Therefore, two parameters need to be studied. One is the distance threshold of a *Certain Segment* (*DT*). The other is the number of consecutive *Uncertain Segments*. As a result, we find that when $SN = 2$ and $DT = 200$ meters, the segmentation method achieves an acceptable

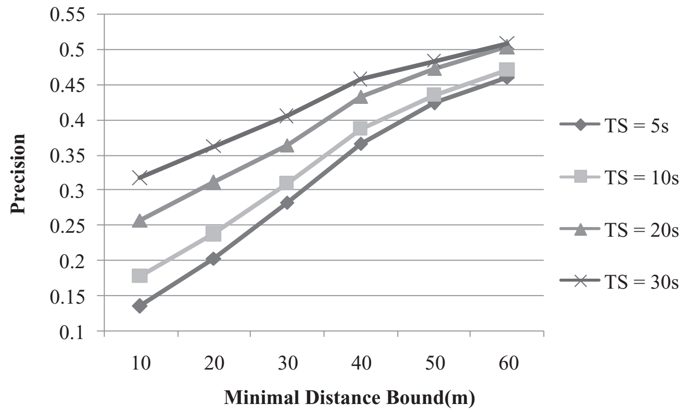


Fig. 23. Precision of the change points after conducting the second step of segmentation.

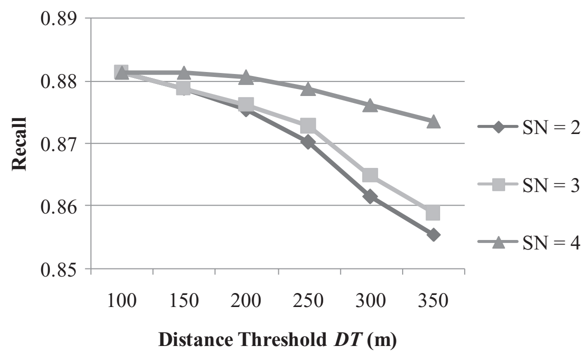


Fig. 24. Recall of the change points after conducting the third step of segmentation.

recall with relatively high precision. In other words, if the distance of a segment is less than 200 meters, it is regarded as an *Uncertain Segment*. If the number of consecutive *Uncertain Segments* exceeds two, these segments will be merged into one segment.

4.4.2 *Advanced Features.* Figure 26 shows the inference accuracy changing over the threshold (*Hc*) when *HCR* is used alone to differentiate different transportation modes. The curve painted in Figure 26 presents us with the prior knowledge that *HCR* becomes the most discriminative when *Hc* is set to 15 degrees. In other words, when a user changes his/her heading direction by a angle greater than 15 degrees, the corresponding GPS point will be sampled into the collection *Pc*, and further calculate *HCR* according to Equation (3).

As depicted in Figure 27, we study the effect of *SR* in stand-alone predicting transportation modes. Obviously, we can get the suggestion that, as compared to other candidates, when *Vs* equals to 2.5, *SR* shows its greatest advantages in classifying different modes.

Figure 28 depicts the inference accuracy changing over the threshold V_r when we employ *VCR* alone. We found evidence that when V_r is set to 0.7, that is, a

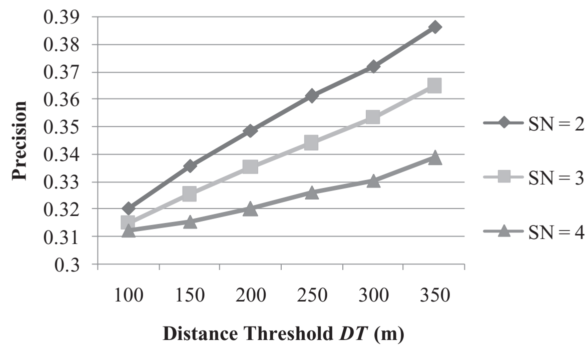


Fig. 25. Precision of the change points after conducting the third step of segmentation.

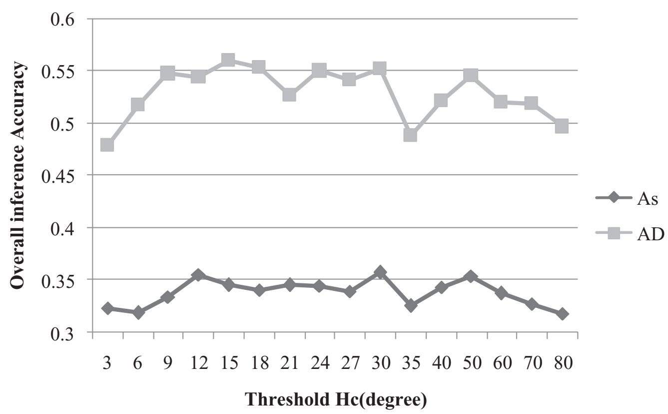


Fig. 26. Selecting threshold *Hc* for *HCR*. The inference is performed only based on *HCR* without post-processing.

GPS point would be sampled into collection *Pv* if its velocity changes beyond 70 percent over its predecessor, *VCR* becomes the most powerful beyond other candidates.

4.4.3 *Postprocessing*. To ensure the effectiveness of the graph-based post-processing, two thresholds (T_1 and T_2) are studied in the experiment. Figure 29 illustrates the statistical results we performed based on the preliminary inference results without postprocessing. Here, $P(m_i|X)$ stands for the posterior probability of a segment being a kind of transportation mode given feature X. Using the relationship between the value of maximum $P(m_i|X)$ and its inference result, we found the following evidence. When the value of maximum $P(m_i|X)$ is smaller than 0.36, the rate of false inference exceeds 60 percent. Instead, when the value is greater than 0.6, the rate of correct inference outscores 90 percent. Thus, we set T_1 to 0.6 and T_2 to 0.36 to reduce the risk of modifying a correct prediction while ensuring a false inference would be revised.

Using the GPS data within a given geographic region, Figure 30 presents the clustering results of OPTICS changing over the number of GPS trajectories. As a result, the number of clusters within the region does not keep on increasing

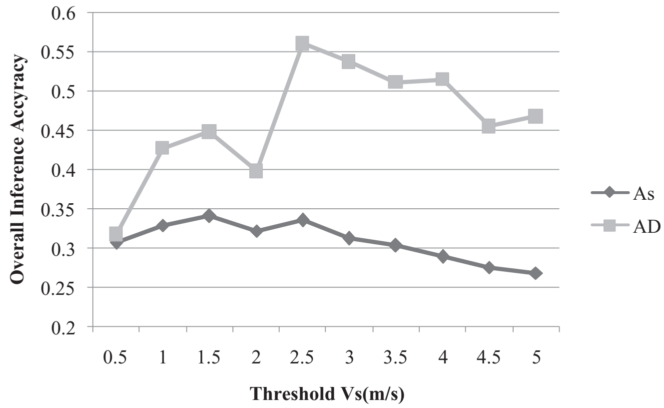


Fig. 27. Selecting threshold (*Vs*) for *SR*. *SR* is the only feature used in the inference model without post-processing.

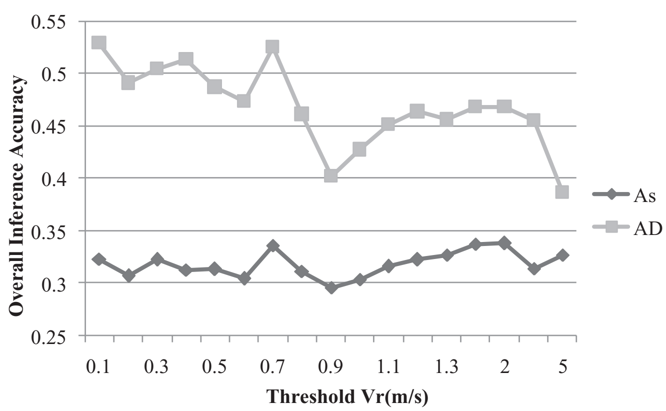


Fig. 28. Selecting threshold (*Vr*) for *VCR*. *VCR* is the only feature used in the inference model without post-processing.

with the incrementally added GPS trajectories. It proves that the number of places where most people change their transportation modes in a given region is limited, and is constrained by the real world. This observation also provides positive support on the feasibility of our graph-based postprocessing. With regard to the OPTICS algorithm, its result depends on two parameters, core-distance (*CorDist*) and minimal points (*minPts*) within the core-distance. According to the commonsense knowledge of real world and experimental evaluation, we found that when *CorDist* = *25* and *minPts* = *5,* the distribution of clusters makes more sense than that based on other parameters.

Figure 31 paints a case of a change point-based graph on a map, and displays the most popular transportation mode on each graph edge. Each circle stands for a cluster and the line between two circles represents a graph edge.

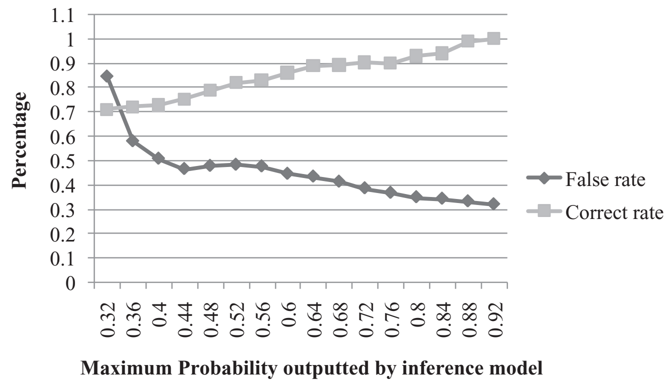


Fig. 29. The statistic results of relationship between maximum $P(m_i|X)$ and inference accuracy.

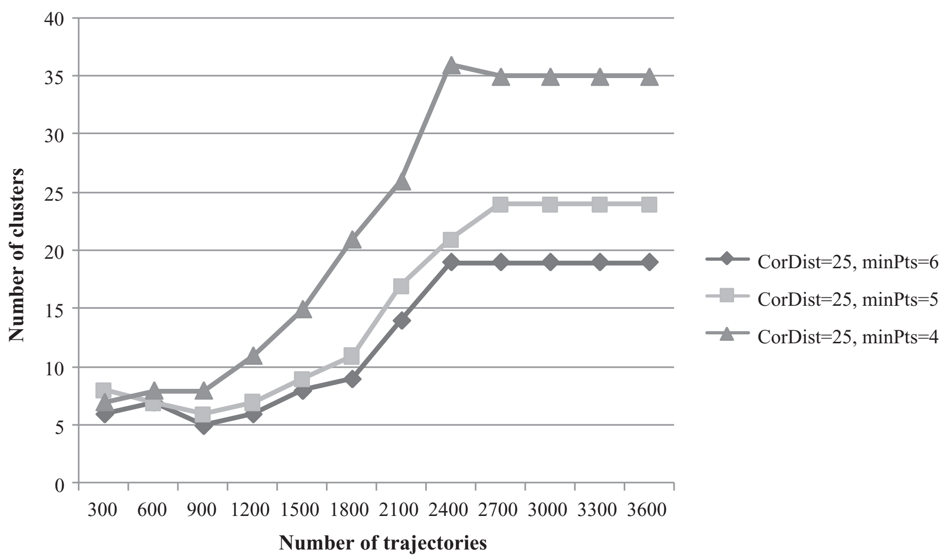


Fig. 30. Number of clusters changing over the number of trajectories in a given region.

4.5 Results

4.5.1 *Single Feature Exploration.* Using the inference accuracy without postprocessing, Table IV shows the capability of each feature in stand-alone differentiation of transportation modes. We observe that *HCR*, *SR* and *VCR* clearly outperform other features. These results justify our claim that the *Advanced Features* is more robust to the variable traffic conditions beyond velocity-based features.

4.5.2 *Effectiveness of Feature Combination.* Table V presents the results of the inference model without performing postprocessing. Using a subset feature selection method, we evaluate the performance of different feature combinations. As we mentioned in Section 3.4, the *Basic Feature* includes *MaxV1, MaxV2, MaxV3, MaxA1, MaxA2, MaxA3, AV, EV, DV* and *Dist.* For simplicity's

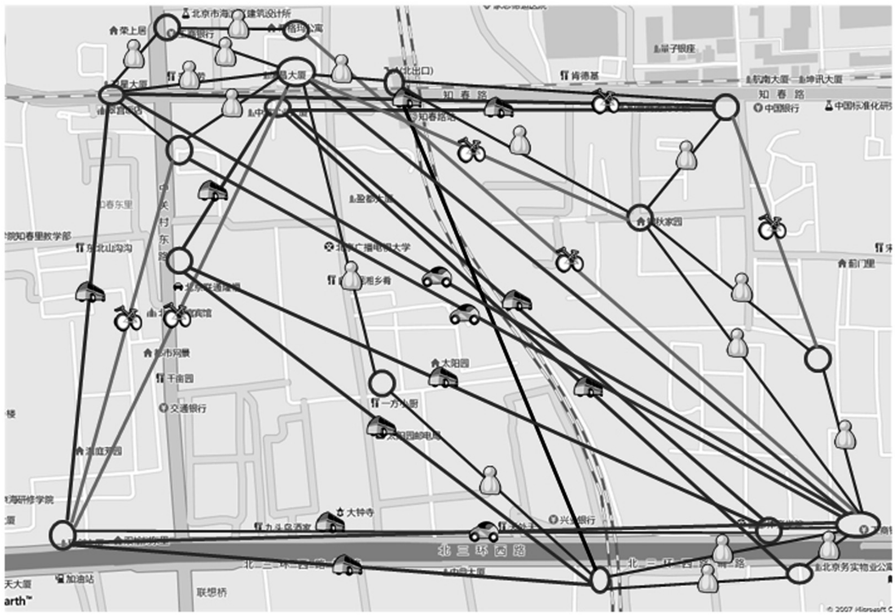


Fig. 31. A change point-based graph within a region.

Table IV. Overall Inference Accuracy Using Each Feature Alone

Rank	Features	A_S	A_D	Rank	Features	A_S	A_D
1	*HCR*	0.345	0.561	8	*DV*	0.269	0.357
2	*SR*	0.335	0.561	9	*MaxV2*	0.322	0.344
3	*AV*	0.382	0.547	10	*MaxV1*	0.294	0.257
4	*VCR*	0.336	0.526	11	*MaxA2*	0.239	0.217
5	*EV*	0.375	0.523	12	*MaxA1*	0.259	0.208
6	*Dist*	0.302	0.499	13	*MaxA3*	0.256	0.197
7	*MaxV3*	0.334	0.365				

sake, the combination of the *Basic Features* and the *Advance Features* is called *Full Feature*.

From the results shown in Table V, we can make two observations. First, the combination of the *Advanced Features* (*SR+HCR+VCR*) is more effective than that of velocity and acceleration in predicting users' transportation modes. It justified our statement that these three features are more robust to traffic conditions than the *Basic Features*. Second, by combining *SR+HCR+VCR* with the *Basic Features*, we attain the highest accuracy. This evidence further proves that the *Advanced Feature* is discriminative and has little correlation between *Basic Features*. In this experiment, the inference results of *Basic features* are slightly less than those reported in paper [Zheng et al. 2008a] due to the increased test data.

4.5.3 *Effectiveness of Segmentation.* Using the *Full Features*, we evaluate the performance of the change point-based segmentation method. We compare our method with two baseline trajectory-partition approaches, including

Table V. Inference Performance of Combined Features Without Performing Postprocessing

Feature Combinations	Transportation Mode		Change Point	
	A_S	A_D	Precision	Recall
MaxA1 + MaxA2 + MaxA3	0.297	0.283	0.118	0.584
MaxV1 + MaxV2 + MaxV3	0.480	0.526	0.142	0.687
Distance + EV + AV	0.480	0.550	0.227	0.582
Distance + EV + MaxV1	0.548	0.597	0.217	0.55
AV + EV + MaxV1	0.558	0.621	0.253	0.603
MaxV3 + MaxA3 + AV	0.511	0.632	0.138	0.669
SR + HCR + VCR	0.575	**0.644**	0.286	0.643
Basic Features	0.618	0.673	0.284	0.681
Enhanced Features	**0.635**	**0.715**	**0.373**	**0.724**

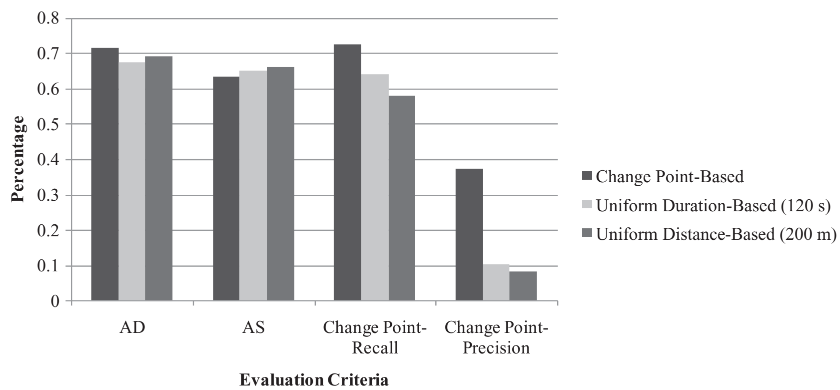


Fig. 32. Comparison among different segmentation methods.

uniform distance-based and uniform duration-based methods. A set of parameters have been studied for these two baseline methods. As a result, we find that when the unit distance is set to 200 meters, the uniform distance-based segmentation method achieved a relatively high performance among the selected parameter candidates. Meanwhile, when we configure the unit duration as 120 seconds, the uniform-based method obtained a relatively high performance. Therefore, Figure 32 shows only the comparison results between our method and the baseline method with the best performance. As we can see, except for the accuracy by segment (A_S), the change point-based segmentation method outperforms its competitors in all the rest of the evaluation criteria.

4.5.4 *Effectiveness of Postprocessing*. Figure 33 shows the inference performance with and without post-processing. Based on the inference model using *Full Features* (*Advanced Features* + *Basic Features*), the normal postprocessing has achieved almost 2 percent improvement in accuracy (A_D) beyond the preliminary results. Further, the graph-based postprocessing outperforms the normal method by bringing a 4-percent promotion over the preliminary inferences. Additionally, the graph-based postprocessing algorithm provides a 5-percent improvement to the precision of change points over the preliminary inference results while maintaining almost the same recall of change points.

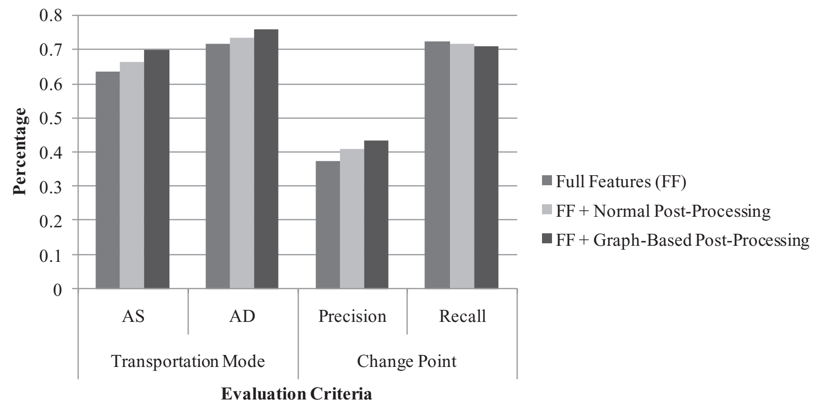


Fig. 33. Evaluation of the post-processing algorithm.

Table VI. Confusion Matrix of Final Inference Results with Graph-Based Postprocessing

		Inferred Results (KM)					
		Walk	*Driving*	*Bus*	*Bike*		
Ground Truth	*Walk*	752.8	107.4	81.5	78.3	**0.738**	Recall
	Driving	41.7	2867.5	563.0	88.3	**0.805**	
	Bus	58.6	425.7	1489.8	105.1	**0.717**	
	Bike	58.9	21.8	290.0	833.1	**0.692**	
		0.825	**0.837**	**0.615**	**0.754**	**0.756**	
		Precision					

Table VI presents detailed inference results, including precision and recall of each kind of transportation mode by distance. With a large-scale dataset, we believe that the graph-based postprocessing would bring a greater improvement to current experimental results. First, with more GPS data, the change point-based graph could cover more places where people log their trajectories with GPS data. Thus, more inferred GPS trajectories can be matched on the graph, and further be processed by the graph-based method. Second, the probability distribution on the graph would become more capable of representing typical user behavior on locations.

4.6 Discussion

4.6.1 *Discussion of the Segmentation Method.* Given the data shown in Figure 32, we made the following observations. Regarding the uniform distance-based and uniform duration-based methods, they are more likely to put consecutive segments of different transportation modes into one segment, and generate many trivial segments with short distances. Thus, as compared to the change point-based approach, it is inevitable that some change points will be missed and more false inferences will be generated. Further, these false inferences will damage the precision of the change points and bring a very bad user experience of browsing a trajectory. Although, these baseline methods also provide relatively good A_S, their poor precision of change points reveal the fact that the correct inferences and false inferences alternate in a trajectory. As a consequence, people are easily confused by the inference result when browsing a trajectory's transportation mode.

The advantages of the change point-based segments lie in two aspects. One is that this approach is capable of maintaining a segment of one transportation mode as long as possible. Therefore, we are more likely to extract discriminative features from each segment and obtain a correct inference. The other is that its merging strategy can handle, to some extent, the vulnerability of features facing variable traffic conditions. In congestion, a user reluctantly moves fast and slow in an alternative manner. If the change point-based segmentation method is employed, these segments might be merged into one segment. However, this trajectory will be partitioned into many trivial segments by the uniform distance or uniform duration-based method.

4.6.2 *Discussion on the Advanced Features.* The results reported in Table IV and Table V justified the effectiveness of *Advanced Features* in classifying transportation modes. Whether being used alone or in combination, the *Advanced Features* always shows its advantages over the *Basic Features*. It is not difficult to understand that *Advanced Features* extracted from a user's trajectory depends more on the characteristics of the vehicle the user selected rather than the traffic conditions. For instance, buses and cars cannot change their moving direction as flexibly as people traveling on foot. This fact would not vary with traffic conditions. Therefore, the reasons why our approach is capable of tackling the affects of heavy traffic include three parts: 1) the change point-based segmentation methods; 2) the *Advanced Features* we identified; and 3) the graph-based post-processing method.

4.6.3 *Discussion on the Graph-Based Postprocessing.* The data shown in Figure 33 and Table VI has shown the contributions that the graph-based post-processing makes in improving the inference accuracy. Here, we mine an implied road network from user-generated GPS logs rather than directly employing a database of map information. Actually, it is impractical to directly match a user's trajectory against a road network and bus stops due to the following reasons. First, as depicted in Figure 34(a), the locations of bus stops are usually close to crossroads. Hence, it is really difficult to judge whether a person was driving or taking a bus based on the observation that the user has passed a region containing a bus stop. In other words, it is also possible that the user might drive a car and wait a traffic light at a crossroad near the bus stop. Second, as demonstrated in Figure 34(b), bikes, buses and cars usually move on the same road surface in an urban area. That may sometimes contradict the assumption that if an individual's trajectory aligns with a highway, the individual might be driving at that moment. Third, matching a trajectory against a given road network would need lots of computational efforts, which easy in theory but difficult in practice.

As opposed to the straightforward method already mentioned, we mine an invisible graph from user-generated GPS logs. The advantages of this graph consist of two parts. First, in this graph, each node just represents a place where

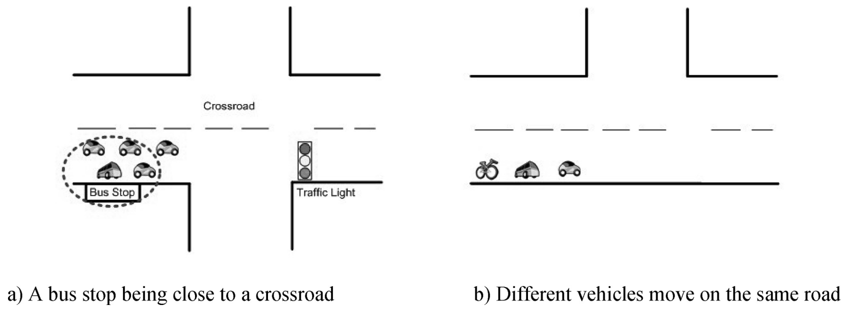


Fig. 34. Some cases opposing the idea of using road network directly.

multiple users change their transportation modes. We neither need to recognize whether a node is a bus stop or railway station, nor match a trajectory against the road network. This data-driven approach takes into account the location constraints of the real world while keeping the post-processing algorithm very efficient. Second, the typical user behaviors based on locations are employed as probabilistic cues to improve the inference results. On each edge, we calculate the probability distribution of different transportation modes as well as the transitions probability among them. In short, we do not fix a false inference based on some pre-defined rules.

5. RELATED WORKS

5.1 GPS Track Sharing

In the application scenarios of GPS-trajectory-sharing [Counts et al. 2007 and SportsDo 2007], some communities have been established to help people share their historical life experience based on GPS data. However, these applications either provided people with raw GPS tracks or required users to manually label their tracks. For instance, some systems tell the users about the basic information, such as distance and duration, of a particular route. Alternatively, the transportation modes of each track are manually tagged by the individual who uploads the GPS log. Due to the extra efforts of user-labeling, many people are frustrated and give up contributing their GPS data to the community. The essential difference between our work and the work mentioned above is that we aim to automatically understand each individual's GPS tracks, and leverage the mined knowledge to improve the applications on the Web.

5.2 User Behavior Recognition Based on Sensor Data

Parkka et al. [2006] and Ermes et al. [2006] aim to recognize human activity, such as walking and running, using the data collected by more than 20 kinds of wearable sensors on a person. A user's body condition, such as temperature, heart rate, and GPS position, as well as environment situation, like environmental humidity and light intensity, are employed as input features of a classification model to differentiate the person's everyday activities. However, it is somehow obtrusive and complex for normal users to carry extra sensors in their daily lives. The major difference between the technique mentioned above and our work is in that we understand human activities only based on raw GPS data. Thus, it is more promising to be deployed in people's GPS phone without increasing their burden in wearing extra sensors. Additonally, as GPS data can be a part of sensor ensemble, our approach is also useful to improve the recognition performance of such methods.

LOCADIO [Krumm et al. 2004] used a Hidden Markov Model to infer motion of a device using 802.11 radio signals, while Timothy et al. [2006] attempted to detect the mobility of a user based on GSM signal. Unfortunately, the observations of such radio signals vary in many conditions, such as time, space and the number of users in a radio cell. As a consequence, the locations estimated based on such signals are quite coarse; so only three simple motions including stationary, walking and driving can be inferred in Timothy's work [2006]. Likewise, LOCADIO can only differentiate two fundamental types of activities, Still and Moving. The essential difference between our approach and this strand of work lies in not only the GPS data with higher locality accuracy but also the spatial information we mined from GPS data to improve the recognition performance. Given the fact that a GPS receiver will lose signal indoors, it is promising to enable a more sophisticated approach to recognize user activities by combining the technique mentioned above with ours.

5.3 GPS Data-Driven Mining

GPS data-based activity recognition has received considerable attention during the past years. These works include extracting significant places of an individual [Ashbrook et al. 2003; Hariharan and Toyama 2004], predicting a person's movement [Krumm et al. 2007; Liao et al 2005] and modeling a user's transportation routine [Liao et al. 2004 and Patterson et al. 2003]. Patterson et al. [2003] use GPS tracks to classify a user's mode of transportation as either "bus," "foot," or "car," and to predict his or her most likely routes. Similarly, Liao et al. [2004] aim to infer an individual's transportation routine given the individual's GPS data. Their system first detects a user's set of significant places, and then recognizes the activities like shopping and dining that could take place at those significant places.

As compared to our approach, these works have the following constraints. 1) It requires the information regarding road networks, bus stops and parking lots. 2) The model learned from a particular user's historical GPS data is customized for the user. Thus, it is not generic to be deployed in ubiquitous computing systems. However, we mine the knowledge from the raw GPS data collected by multiple users while the knowledge can contribute to both personal use and public use. What's more, we do not need additional information from other sensors or map information like bus stops. This information was automatically mined from user-generated GPS logs.

6. CONCLUSION

In this paper, we move towards understanding user mobility based on GPS data. A work aiming to infer transportation modes from GPS logs based on supervised learning is reported. Using our approach, four kinds of transportation modes, consisting of walking, driving, taking a bus and riding a bike, are differentiated from one another. Such transportation modes can feature a user's mobility and support a variety of pervasive computing systems, such as human behavior recognition, trajectory sharing and smart route recommendation. Using the GPS logs collected by 65 people over a period of 10 months, we evaluated our approach via a set of experiments and produced the following results.

First, the change point-based segmentation approach outperforms the baseline methods, including uniform distance-based and uniform duration-based segmentation, in inferring the transportation modes and detecting change points of a GPS trajectory. This method can effectively partition a trajectory into segments of different transportation modes, while maintaining a segment of one mode as long as possible. Consequently, this approach is more likely to provide a better foundation for the inference model and achieve a relative higher inference performance.

Second, beyond the *Basic Features* directly using velocity and acceleration, the *Advanced Features*, including heading change rate, stop rate, and velocity change rate, is more discriminative in differentiating between transportation modes. No matter being used alone or in combination, the *Advanced Features* always shows its advantages over the *Basic Features*. It is not difficult to understand that the *Advanced Features* depends more on the characteristics of the vehicle a user selected rather than the traffic conditions the user experienced. Based on the change point-based segmentation method and Decision Tree-based inference model, we achieved an inference accuracy greater than 0.71 by combining the *Advanced Features* with *Basic Features*.

Third, beyond the normal postprocessing, the graph-based post-processing algorithm brought a 4 percent improvement over the preliminary inference results. In this method, we mined an implied graph from user-generated GSP logs. This graph contains the probabilistic cues, including the commonsense constraints of the real world and typical user behaviors based on locations, while keeping our method independent of additional map information.

ACKNOWLEDGMENTS

Many thanks to the volunteers who helped us collect GPS data in the past years. We also appreciate the input of Like Liu and Longhao Wang, who helped us conduct some experiments in the beginning of this project.

REFERENCES

ANKERST, M., BREUNIG, M. M., KRIEGEL, H., AND SANDER, J. 1999. OPTICS: Ordering points to identify the clustering structure. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, A. Delis, C. Faloutsos, and S. Ghandeharizadeh, Eds. ACM Press, New York. 49–60.

ASHBROOK, D. AND STARNER, T. 2003. Using GPS to learn significant locations and predict movement across multiple users. *Personal Ubiquitous Comput. 7*, 5, 275–286.

BREIMAN, L. 1996. Bagging Predictors. *Machine Learn. 24,* 123–140

CHEN, G. AND KOTZ D. 2000. A survey of context-aware mobile computing research. Tech. rep. TR2000-381, 2000, Dartmouth College Hanover, NH.

Counts, S. and Smith M. 2007. Where were we: Communities for sharing space-time trails. In *Proceedings of the 15th International Symposium on Advances in Geographic Information Systems*, Samet H., Shahabi C. and Schneider M., Eds. New York. 10–18.

Ermes, M., Parkka, J., Mantyjarvi, J., and Korhonen, I. 2006. Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions. *IEEE Trans. Inform. Tech. Biomed. 12*, 1, 20–26.

Hariharan, R. and Toyama, K. 2004. Parsing and modeling location histories In *Proceedings of the 3rd International Conference on Geographic Information Science*. Springer 106–124.

Krumm, J., and Horvitz, E. 2006. Predestination: Inferring destinations from partial trajectories. In *Proceedings of the 8th International Conference on Ubiquitous Computing*. Springer-Verlag 243–260.

Krumm, J., and Horvitz, E. 2004. LOCADIO: Inferring motion and location from wi-fi signal strengths. In *Proceedings of the 1st International Conference on Mobile and Ubiquitous.* Computing. IEEE Press, 4–13.

Liao, L., Patterson, D. J., Fox, D., and Kautz, H. 2005. Building personal maps from GPS data. *Annals N.Y. Acad. Sci. 1093*, 249–265.

Liao, L., Fox, D., and Kautz, H. 2004. Learning and Inferring Transportation Routines. In *Proceedings of the National Conference on Artificial Intelligence*. AAAI Press, 348–353.

Patterson, D., Liao, L., Fox, D., and Kautz, H. 2003. Inferring high-level behavior from low-level sensors. In *Proceedings of the 8th International Conference on Ubiquitous Computing*. Springer, 73–89.

Parkka, J., Ermes, M., Korpipaa, P., Mantyjarvi, J., and Peltola, J. 2006. Activity classification using realistic data from wearable sensors. *IEEE Trans. Inform. Tech. Biomed. 10*, 1, 119–128.

Sahr, K., White, D., and Kimerling, A. J. 2003. Geodesic discrete global grid systems. *Cartography and Geog. Inform. Sci. 30*, 2, 121–134

SportsDo. 2007. http://sportsdo.net/Activity/ActivityBlog.aspx

Toyama, H. 2004. Project lachesis: Parsing and modeling location histories, In *Proceedings of the 3rd International Conference on Geographic Information Science*. Springer, 106–124.

Toyama, K., Ron, L., and Roseway, A. 2003. Geographic location tags on digital images. In *Proceedings of the 11th ACM International Conference on Multimedia*. Lawrence A. Rowe, Harrick M. Vin, Thomas Plagemann, Prashant J. Shenoy and John R. Smith, Eds. 156–166.

Timothy, S., Varshavsky, A., Lamarca, A., Chen, M. Y., and Choudhury, T. 2006. Mobility detection using everyday GSM traces. In *Proceedings of the 8th International Conference on Ubiquitous Computing.* Springer. 212–224

Wang, L., Zheng, Y., Xie, X., and Ma, W. Y. 2008. A flexible spatio-temporal indexing scheme for large-scale GPS track retrieval. In *Proceedings of the 9th International Conference on Mobile Data Management*. IEEE Press. 1–8.

Zheng, Y., Li, Q., Chen, Y., Xie, X., and Ma, W. Y. 2008a. Understand mobility based GPS data. *In Proceedings of the 10th International Conference on Ubiquitous Computing*. ACM Press, 312–321.

Zheng, Y., Liu, L., Wang, L., and Xie, X. 2008b. Learning transportation mode from raw GPS data for geographic applications on the Web. In *Proceedings of the 11th International Conference on World Wide Web*. ACM Press, 247–256.

Zheng, Y., Wang, L., Zhang, R., Xie, X., and Ma, W. Y. 2008c. GeoLife: Managing and understanding your past life over maps. In *Proceedings of the 9th International Conference on Mobile Data Management*. IEEE Press, 211–212.

Zheng, Y., Xie, X., and Ma, W. Y. 2008d. Search your life over maps. In *Proceedings of the International Workshop on Mobile Information Retrieval*. 24–27.

Received August 2008; revised February 2009; accepted August 2009