The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-2, 2014
ISPRS Technical Commission II Symposium, 6 – 8 October 2014, Toronto, Canada

TRAJECTORY CLUSTERING FOR PEOPLE'S MOVEMENT PATTERN BASED ON CROWD SOURING DATA

Jiangping Chen, Ting Hu, Pengling Zhang, Wenzhong Shi
School of Remote Sensing and Information Engineering ,Wuhan University, chen_jp@whu.edu.cn
129 Luoyu Road ,Wuhan ,China ,430079

Technical Commission II

KEY WORDS: GPS trajectories data; Spatial-temporal clustering algorithm; Movement pattern

ABSTRACT:

With the increasing availability of GPS-enabled devices, a huge amount of GPS trajectories recording people's location traces have been accumulated and shared freely on the Web. In this area, one of the most important research topics is to exploit trajectory-movement pattern about where and when people clustered based on the raw GPS data. In order to solve this problem, clustering is a good way to perform data mining tasks on trajectory data.

This paper provides a clustering algorithm which aims at mining people's movement pattern about the clustered location and their temporal evolution characteristics. Firstly, the characteristic points of GPS trajectories were chosen. Based on the characteristic points, a trajectory has been partitioned into a group of line segments. These line segments can represent the movement pattern of trajectories much better than that of track points. Secondly, an improved density-based line clustering method was used for the individual partitioned line segments to find out individual clusters with similar track segments. In this step, the absolute time spot of people's trajectories was taking into account as a characteristic for the temporal evolution of people's trajectories. Finally, the representative clustered hot spots of multiple users' line segments achieved by above steps were output. Experiments were conducted with GPS trajectories data downloaded from the web to verify the effectiveness of the algorithm in this paper. According to the results, the spatial distribution and temporal evolution characteristics of people's stay hot spots were effectively discovered from people's GPS trajectories data.

1. INTRODUCTION

The increasing availability of GPS-enabled devices is changing the way people interact with the Web, and has facilitated people to record their location trace with GPS trajectories. More and more users start recording their outdoor activities with GPS trajectories for many reasons, such as travel experience sharing, life logging, sports activity analysis and multimedia content management, etc[1]. Therefore, a huge amount of GPS trajectories representing people's location information have been accumulated on the Web. For example, according to the OSM official website statistics[2], there have been 1.5million users registered on the website, with 380 billion track points shared and more than 10000 cities over the world included, by the end of 2013. Another case is the project GeoLife proposed by Zheng Yu et al.[3] from Microsoft Research Asia. The team conducted experiments using GPS data collected from 181 volunteers during a period of 2 years in the real world to mine valuable knowledge like individual life pattern[4], transportation mode[5] and user similarity[6], etc. Moreover, these GPS data are all shared freely on the Web. It brings us challenges as well as opportunities to discover the valuable knowledge that we need from the massive amounts of GPS trajectories. One of the most important research topics in this area is to exploit trajectory-movement pattern about where and when people clustered or stayed based on the raw GPS data.

In order to discover people's trajectory movement pattern, clustering is a good way to perform data mining tasks on trajectory data. Jae-Gil Lee et al.[7] have proposed a trajectory clustering algorithm TRACLUS based on a partition-and-group framework which partition the trajectories into sub track segment using the minimum description length (MDL) principle. They made experiments with hurricane trajectories and animal movement data to discover the sub track segment with the same movement pattern in space. Since the movement pattern of human GPS trajectories is quite different from

This contribution has been peer-reviewed.
doi:10.5194/isprsarchives-XL-2-55-2014

hurricanes' or animal movements', the spatial-temporal characteristics of people's trajectories are essential to be considered in the clustering process. Zhang Yanling et al.[8] have proposed a new method T-CLUS, which partitions a trajectory into line segments based on characteristic points. T-CLUS identifies cluster structure of sub-trajectories by means of reachability plot. With experiments on hurricane trajectories they have obtained the movement pattern of the sub-trajectories. Although the research has calculated the time interval between two points as time distance, it may lose some inner properties of human GPS trajectories since they discarded the absolute time spot contained by each log point in GPS trajectories data. The absolute time spot is quite important for analyzing the temporal evolution of people's movement pattern.

In this paper, an advanced clustering method by taking more spatial-temporal information into account is provided. It aims at mining people's movement pattern about the clustered location and their temporal evolution characteristics.

2. PROBLEM STATEMENT

Based on a GPS trajectory data set collected in the real world, this paper develop a spatial-temporal clustering algorithm in order to discover people's trajectory movement pattern. Given a set of trajectories, our algorithm generates a set of clusters *C* from a group of line segments which is constituted by the characteristic points chosen from the raw trajectories. And a representative clustered hotspot is computed for each cluster *Ci* to stand for a hotspot region people clustered. Now the track point, trajectory, characteristic point, cluster and representative clustered hotspot are defined as follows.

Definition 1 (Track point) a time-stamped track point is defined as *P (Lat, Lon, T)*, wherein a *Lat* for latitude, *Lon* for longitude, *T* for the timestamp.

Definition 2 (GPS trajectory) a GPS trajectory *Tra* is denoted as a set of track points according to the time sequence, *Tra* = {*p1, p2,...,pn*}, which *pi.T* < *pi+1. T.*

Definition 3 (Characteristic point) characteristic points represent the points where the behavior of a trajectory changes rapidly. Given a direction angle threshold θd, a velocity threshold θv and a time threshold θt, we choose characteristic points from a trajectory *Tra* = {*p1, p2,...,pn*} when the change of the trajectory is greater than the threshold values. We determine characteristic points as *cp*, a set of characteristic points *CP* = {*cp1, cp2,...,cpn*} can be chosen from the trajectory. Two consecutive characteristic points constitute a line segment of the trajectory, and it represents the trajectory's partial characteristic.

Definition 4 (Cluster) a cluster is a density-connected set of trajectory partitions. A trajectory partition is a line segment constituted by the characteristic points chosen from the same trajectories. On the basis of distance measure, the line segments which belong to the same cluster are close to each other. Since a trajectory is partitioned into multiple line segments and clustering is performed over these line segments, a trajectory can belong to multiple clusters.

Definition 5 (Representative clustered hotspot) a representative clustered hotspot is the center point of a cluster. It is a computed track point that represents the clustered trajectory segments of the cluster. Given a cluster $Ci = \{LC_1, LC_2, ..., LC_n\}$, here *LC*(*Pstart, Pend*) is a line segment and *n* is the number of line segments in *Ci*, the representative clustered hotspot *Pc* is:

$$Pc(\mathrm{Lat}, \mathrm{Lon}, \mathrm{T}) = \frac{1}{2\mathrm{n}} \sum_{\mathrm{i}=1}^{\mathrm{n}} (Psi + Pei) \tag{1}$$

3. TRAJECTORY PARTITIONING

In this section, a trajectory partitioning algorithm is proposed to find out the characteristic points we defined in Definition 3. After the characteristic points are chosen, the trajectory *Tra* can be partitioned at every characteristic point, and two consecutive characteristic points constitute a line segment of the trajectory. Therefore, *Tra* is partitioned into a set of line segments {*cp1cp2, cp2cp3, ..., cpend-1cpend*}. Literature[7] called such a line segment a trajectory partition, and it is also suitable for this paper. Figure 1 shows an example of a trajectory and its trajectory partitions.

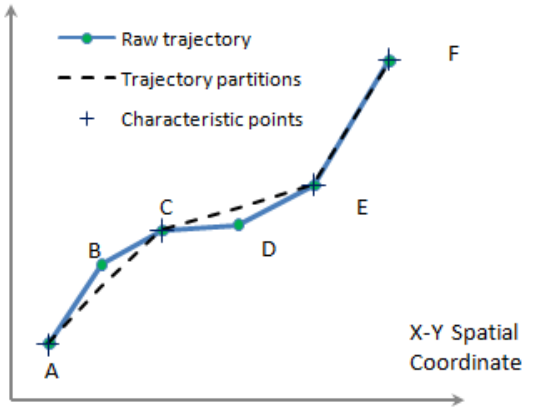

Figure 1. An example of trajectory partitioning

Since we aim at mining people's movement pattern from the real travel trajectories of some volunteers, we add the analysis of real trajectories into our trajectory partitioning algorithm, and the characteristic points can be divided into three categories: (1) direction characteristic points, which mean the direction angle of a track point changes obviously, such as a road crossing; (2) velocity characteristic points, which are related to the travel modes; (3) time characteristic points, most time the track points of a trajectory are recorded every 5-10 seconds but sometimes the time interval between two consecutive points is more than several minutes, we determine these points as time characteristic points. The trajectory partitioning algorithm is described as follows.

Initially, we input a trajectory $Tra = \{< p1, ..., pi, ..., pn> |(1 \le i \le n)\}$ and 3 threshold values: a direction angle threshold θd, a velocity threshold θv, a time threshold θt. And we aim to output a set *CP* of characteristic points finally. The steps of the algorithm are:

First of all, the starting point *p1* was added to the characteristic points set *CP*, then in the next cycle the points which change the trajectory characteristic rapidly are find out to add into the set *CP*. We compute direction changes *dc*, velocity changes *vc* and time interval *ti* between each adjacent line segments. If one of the variations exceeds the corresponding threshold value, then the track point which connects the two adjacent line segments is chosen as a characteristic point. The time complexity of the algorithm is *O(log n)*.

4. TRAJECTORY CLUSTERING

People's trajectories usually have irregular shapes and they are always clustered when people have the same activities or movements, therefore, the density-based clustering algorithm is quite applicable for us to perform clustering task on trajectory data. In this section, we develop a line-oriented density-based clustering algorithm based on the algorithm DBSCAN[10]. Firstly, the distance function used in clustering line segments is introduced in Section 4.1. Next, a spatial-temporal line segment clustering algorithm is proposed for the partitioned trajectory segments to find out clusters with similar track segments in Section 4.2. Finally we get the representative clustered hot spots of each trajectory segments cluster. Because compared with line segments clusters, clustered hot spots can show the spatial distribution and temporal evolution characteristics more intuitively.

4.1 Distance Function

In our study, people's trajectories in the real world are taken as our research object to analyze people's movement behaviors. For instance, Figure 2 shows a trajectory in real world, it can be seen that the trajectory is from place A to place B, and there are some segments clustered near A and B. We can infer that a person have some activities at A and B, such as working, shopping, jogging etc. If a person have some trajectories that always clustered somewhere at a regular time, then we can extract the places as a clustered hot spot by our clustering algorithm. Therefore, we need to define a distance function in order to measure the distance between these trajectory segments in our clustering algorithm. Typically, the distance function between line segments is put forward by considering the positional relationship or the similarities of shapes. However, most of the existing distance measurement methods only consider the spatial attribute and ignore the temporal attribute. Thence, we define the distance function by taking advantage of the trajectory segments' temporal characteristics as well as the spatial characteristics.

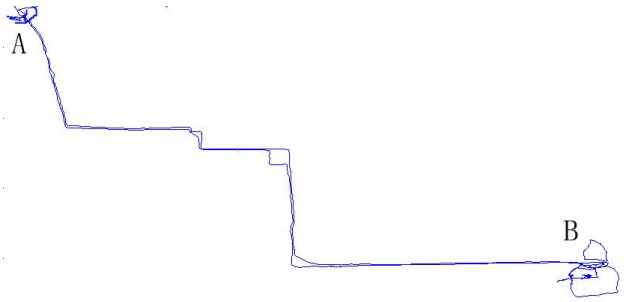

Figure 2. An example of a trajectory in real world

The distance function used in clustering line segments consists of two parts: (1) spatial distance (*ds*); (2) temporal distance (*dt*). The data model of sub-track segments is expressed as: *Li* = <*si, ei*>, *Lj* = <*sj, ej*>. Here, *s* and *e* respectively represent a start and an end of a line segment. The distance function is described as follows:

(1) The spatial distance *ds*. According to the trajectories' spatial characteristics, spatial distance *ds* is consisted of the positional distance (*dp*) and speed distance (*dv*).

The positional distance *dp* represents the absolute positional relationship between the trajectory segments. We measure the positional distance using Hausdorff distance formula (2).

$$dp(Li, Lj) = \max(h(Li, Lj), h(Lj, Li)) \tag{2}$$

Herein, $h(Li, Lj) = \max_{a \in Li}(\min_{b \in Lj}(dist(a,b)))$ is the direct Hausdorff distance between *Li* and *Lj*. It is the maximum distance from the points of *Li* to the nearest point of *Lj*. *dist (a, b)* represents the Euclidean distance between the points.

The speed distance *dv* can represent the differences of people's motion state and travel mode. Mostly, people are going on foot when they have daily activities around somewhere. The speed distance *dv* is defined using Formula (3).

$$dv(Li,Lj) = |Li.speed - Lj.speed| \tag{3}$$

Here, $L.speed = dist(L.start, L.end)/\Delta t$ is the speed of a line segment, Δt is the time interval between the start point and end point.

Finally the spatial distance *ds* is represented using Formula (4). Without losing generality, we set ξ equal to 0.5 in default.

$$ds = \xi * dp + (1-\xi)dv \tag{4}$$

(2) The temporal distance *dt*. Each track point of a trajectory has an absolute timestamp like "*Year/Month/day Hour: Minute: Second*", we take the information of hour and minute to measure the time distance of two trajectories but not the date. As Figure 3 shows, t_{si} and t_{ei} are respectively the start time and end time of line segment *Li*, t_{sj} and t_{ej} are respectively the start time and end time of line segment *Lj*. The time interval of different trajectories may intersect or not, showed in Figure 3 (a), (b) respectively. Therefore, we measure the temporal distance in two cases using Formula (5). If there is no intersection between the time intervals of two segments, then the distance is from the first segment's start time to the second segment's end time. Otherwise, the distance is from the first segment's start time to the second segment's end time and minus their overlap portion. The unit of *dt* is minute.

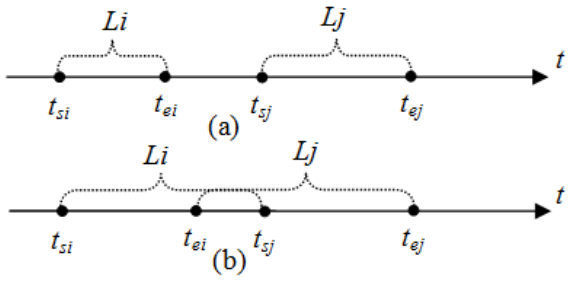

Figure 3. Time interval of different trajectories

$$dt = \begin{cases} |t_{si} - t_{ej}|, t_{ei} \le t_{sj} or t_{ej} \le t_{si} \\ |t_{si} - t_{ej}| - |t_{sj} - t_{ei}|, t_{ei} > t_{sj} or t_{ej} > t_{si} \end{cases} \tag{5}$$

4.2 Line Segment Clustering Algorithm

Since the line segment and the point in space are two different vector graphics, and the parameter thresholds defined in traditional DBSCAN algorithm are only applicable for points, we define the related conceptions as follows to describe our line segment clustering algorithm. Let *D* denote the set of all line segments, *C* the set of clusters and *S* the set of representative clustered hot spots. The algorithm requires three parameters *Eps_space*, *Eps_time* and *MinLength*.

Definition 6 The *spatial-temporal neighborhood Ne(Li)* of a line segment $Li \in D$ is defined by $Ne(Li) = \{Lj \in D \mid ds(Li,Lj) \leq Eps\_space \text{ and } dt(Li,Lj) \leq Eps\_time\}$. Herein, *Eps_space* is the spatial neighborhood threshold and *Eps_time* is the temporal neighborhood threshold. The line segments in *Ne(Li)* are marked as neighbors of *Li*, they are adjacent not only in space but also in time.

Definition 7 A line segment $Li \in D$ *is* called a *core line segment* w.r.t. *Ne(Li)* and *MinLength* if $|Ne(Li)| \geq MinLength$. $|Ne(Li)|$ denotes the total length of the neighbor segments in *Ne(Li)*.

Definition 8 A line segment $Li \in D$ is *directly density-reachable* from a line segment $Lj \in D$ w.r.t. *Eps_space*, *Eps_time* and *MinLength*., if $Li \in Ne(Lj)$ and $|Ne(Lj)| \geq MinLength$.

Definition 9 A line segment $Li \in D$ is *density-reachable* from a line segment $Lj \in D$, if there is a chain of line segments $Lj, Lj-1, \ldots, Li+1, Li \in D$ such that *Lk* is directly density-reachable from *Lk+1* w.r.t. *Eps_space*, *Eps_time* and *MinLength*.

Definition 10 A line segment $Li \in D$ is *density-connected* to a line segment $Lj \in D$ w.r.t. *Eps_space*, *Eps_time* and *MinLength*. if there is a line segment $Lk \in D$ such that both *Li* and *Lj* are density-reachable from *Lk* w.r.t. *Eps_space*, *Eps_time* and *MinLength*.

Definition 11 Given a line segments set *D*, if there is a non-empty subset $C \subseteq D$, *C* is called a *density-connected set* if it satisfies the following two conditions:

(1) $\forall Li,Lj \in D$, if $Li \in C$ and *Lj* is density-reachable from Li w.r.t. *Eps_space*, *Eps_time* and *MinLength*, then $Lj \in C$.

(2) $\forall Li,Lj \in C$, *Li* is density-connected to *Lj* w.r.t. *Eps_space*, *Eps_time* and *MinLength*;

Now our density-based spatial-temporal clustering algorithm for line segments is presented. Although our algorithm shares the basic characteristics with the algorithm DBSCAN, we improve the DBSCAN to spatial-temporal clustering algorithm so that we can generate a set of clusters $C = \{LC1, LC2, \ldots, LCm\}$ from a set of line segments $D = \{L1, L2, \ldots, Ln\}$. The line segments in one cluster are adjacent both in space and time. Our algorithm requires three parameters: the spatial neighborhood threshold *Eps_space*, the temporal neighborhood threshold *Eps_time* and the minimal length *MinLength*. The process of our clustering algorithm is described as follows:

(1) The algorithm scans each segment of the database *D*, if the current line segment *Li* has not yet been classified, the neighborhood of *Li Ne(Li)* is computed. Each line segment of *D* is scanned and computed with *Li*. The computation of *Ne(Li)* is divided into two parts: First the line segments are filtered according to *Eps_time*, if the time interval between current line segments exceeds *Eps_time*, then the line segment are not allowed to enter the next computation of spatial distance; otherwise, compute the spatial distance between the two line segments according to the distance function, if it is not more than *Eps_space*, the line segment will be added in *Ne(Li)* as a neighbor of *Li*.

(2) *Ne(Li)* we acquired in step (1) is used to judge whether the segment is the core segment or not. The sum of all segments in *Ne(Li)* is calculated, and if it is greater than the threshold *MinLength*, *Li* is a core line segment. Otherwise, *Li* is classified as a noise.

(3) If *Li* is a core line segment, each neighbor in *Ne(Li)* is expanded outward to find out all the line segments *density-connected* to *Li*. As a result, a density-connected set of the core line segment is obtained as a cluster, and all the members in this set are marked as the same cluster.

(4) The above process is repeated until all the line segments have been scanned.

(5) At last we compute the center point of each cluster as the representative clustered hot spots and output them.

5. EXPERIMENT RESULTS AND DISCUSSION

5.1 Experimental Setting

We use a real trajectory data set downloaded on the Internet: the *GeoLife* data set. *GeoLife* is a project proposed by Zheng Yu et al.[3] from Microsoft Research Asia. It contains the GPS trajectory data collected from 181 volunteers during a period of 4 years in the real world. The information of a raw

GPS trajectory includes the track point's latitude, longitude, elevation, and time stamp. We choose the trajectory data of 15 volunteers through 2009 to 2010 from the *GeoLife* for experiments, which has 651 trajectories and 504172 points in total. We choose them for the reason that their trajectories are mostly in a closed region so that our study area is relatively concentrated. We extract the information of the track points' latitude, longitude and timestamp for experiments. All the experiments are conducted on a Pentium(R) Dual-Core 2.70 GHz PC with 2 GBytes of main memory, running on Windows7. Our algorithms are implemented in C# using Microsoft Visual Studio 2010.

5.2 Experiment Results

The aim of our trajectory segment clustering analysis is to find spatial-temporal adjacent track clusters, which is a representation of people's movement patterns. Since individuals always have regular travelling activities, we conduct our line segment clustering algorithm experiments on personal trajectory data of every user to find out each individual trajectory clusters. In the line segment clustering algorithm, there are three important parameters: the spatial neighborhood threshold *Eps_space*, the temporal neighborhood threshold *Eps_time* and the minimal length *MinLength*. The choice of different threshold value has an impact on the results of clustering. According to repeated experiments, it is found that when *Eps_space* is set during 0.2 to 0.6, *Eps_time* is set 30 minutes and *MinLength* is equal to 5-7 times the length of the longest track segment, we can get a preferable clustering result. We take a test on a trajectory data set of a user who had 30 trajectories through 3 months. According to the clustering method, we finally get 8clusters. Figure 4 shows a part region of the clustering result. Thin black dotted lines display trajectories, and thick blue lines display clustered track segments. We observe that when people have travelling activities, their trajectory segments are clustered around some place at a certain moment.

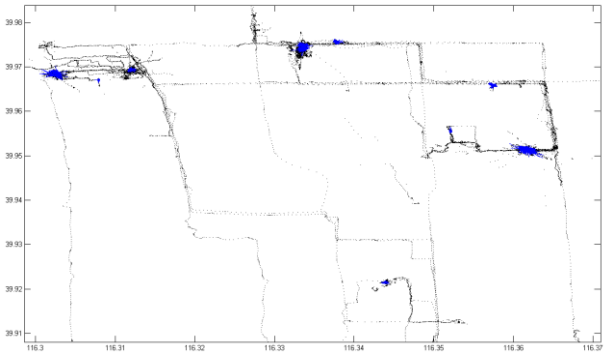

Figure 4. Clustered line segments of an individual

However, the segments clusters are so highly overlapped in space that it is difficult to understand the spatial-temporal clustering results by the expression of the above-described two-dimensional plane. For that, we display the representative clustered hot spots in a vision of three-dimension. Figure 5 shows the result. We gather 15 users' clustered hot spots which were achieved by our clustering algorithm. X axis represents the latitude, Y the longitude and Z the time. Black lines in the X-Y coordinate space display trajectories, and blue points in three-dimensional space-time represent clustered hot spots. Here, the number of clusters is that of blue points. Since the absolute time spot of people's trajectories was taking into account, we know that the points at the same Z axis (time) in the vertical direction stand for people's clustered hot spots which are at the same location but different moments. It can be a representation for the temporal evolution of people's trajectories hot spots. Moreover, the points on the same horizontal plane at one time are on behalf of the spatial distribution of people's clustered hot spots at that moment.

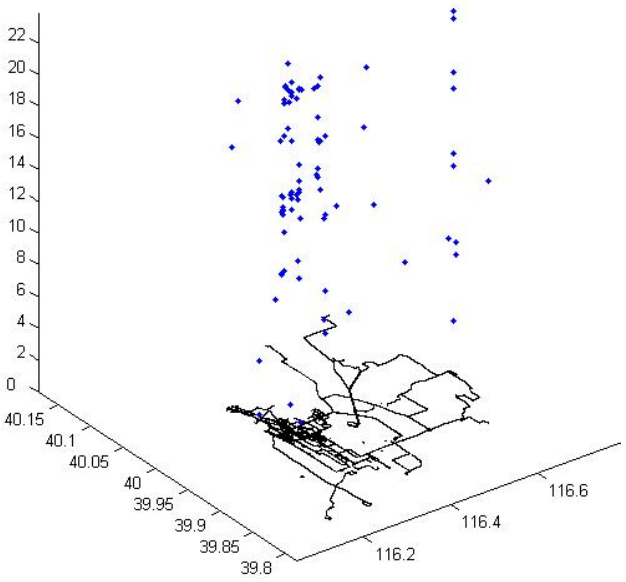

Figure 5. clustered hotspots of multipeople

5.3 Efficiency of Algorithms

Figure 6 shows the changes of the algorithms' execution time when the number of trajectories is increased. As we can see, the curve presents a linear growth, indicating that the method in this paper has scalability. Here, we compare the algorithm in this article with original DBSCAN method. The algorithm in this article is segment-oriented, on the line segment covers a plurality of points, which will greatly reduce the number of scans. Therefore, compared with original DBSCAN method, our algorithm reduced the execution time significantly.

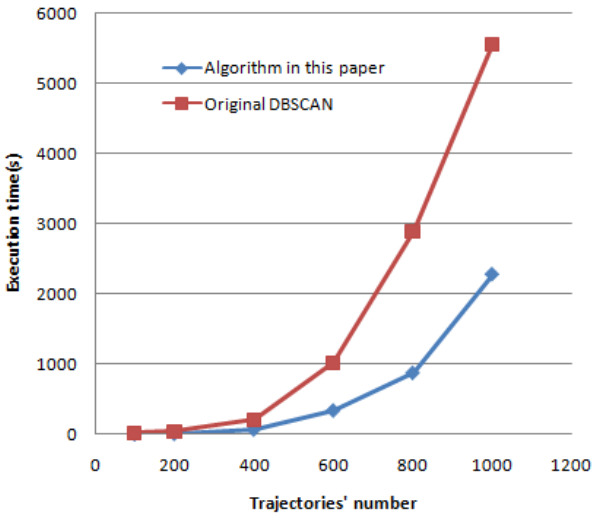

Figure 6. comparison of the algorithms' execution time

6. CONCLUSIONS

In this paper, a clustering algorithm which aims at mining people's movement pattern about the clustered location and temporal evolution characteristics is provided. As the algorithm progresses, a trajectory is partitioned into a set of line segments at characteristic points, and then, the individual partitioned line segments are clustered to find out individual clusters with similar track segments. Eventually the representative clustered hot spots of multiple users' line segments were output. In our algorithm, the absolute time spot of people's trajectories was taking into account as a characteristic for the temporal evolution of people's trajectories. To show the effectiveness of our algorithm, we have performed extensive experiments using a real GPS trajectory data set: *GeoLife*. The visual inspection results of clustering results have demonstrated that our algorithm effectively identifies the spatial distribution and temporal evolution characteristics of people's clustered hot spots. It should be noted that the effects of parameter values is not analyzed enough and quantitatively. We will focus on the quantitative evaluation analysis in further study.

7. REFERENCES

[1] Y. Zheng, L. Zhang, X. Xie, W. Ma, Mining interesting locations and travel sequences from GPS trajectories, *In Proceedings of International conference on World Wild Web (WWW 2009)*, Madrid Spain.

[2] OpenStreetMap. http://wiki.openstreetmap.org/wiki/Stats.

[3] Y. Zheng, Yukun Chen, Xing Xie, Wei-Ying Ma. GeoLife2.0: A Location-Based Social Networking Service. *In proceedings of the International Conference on Mobile Data Management 2009 (MDM 2009)*.

[4] Y. Ye, Y. Zheng, Y. Chen, X. Xie, Mining Individual Life Pattern Based on Location History, *In proceedings of the International Conference on Mobile Data Management 2009 (MDM 2009)*.

[5] Y. Zheng, L. Liu, L. Wang, and X. Xie, Learning Transportation Mode from Raw GPS Data for Geographic Applications on the Web, *17th International World Wide Web Conference (WWW 2008)*, Beijing, China, Apr. 2008.

[6] Q. Li, Y. Zheng, Y. Chen, X. Xie, Mining user similarity based on location history, *In Proceedings of ACM SIGSPATIAL conference on Geographical Information Systems (ACM GIS 2008)*, Irvine, CA, USA.

[7] J.-G. Lee, J. Han, and K.-Y. Whang, Trajectory clustering: A partition-and-group framework, *in Proc. of SIGMOD*, 2007, pp. 593–604.

[8] Y. Zhang, J. Liu，B. Jiang, Partition and clustering for sub-trajectories of moving objects. *Computer Engineering and Applications*, 2009, 45（10）:65-68.

[9] Chen, J., Leung, M. K. H., and Gao, Y., Noisy Logo Recognition Using Line Segment Hausdor® Distance, *Pattern Recognition*, 2003, 36(4): 943-955.

[10] Ester, M., Kriegel, H.-P., Sander, J., and Xu, X., A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, *In Proc. 2nd Int'l Conf. on Knowledge Discovery and Data Mining*, Portland, Oregon, pp. 226-231, Aug. 1996.