Direction Clustering for Characterizing Movement Patterns

Wenjun Zhou, Hui Xiong, Yong Ge
MSIS Department, Rutgers University
Newark, NJ 07102 USA
hxiong@rutgers.edu, {yongge,wjzhou}@pegasus.rutgers.edu

Jannite Yu, Hasan Ozdemir, K.C. Lee
Panasonic System Solutions Development Center of USA
Princeton, NJ 08540 USA
{jyu,timucin,kclee}@research.panasonic.com

Abstract

The increasing availability of motion data creates unprecedent opportunities to change the paradigm for characterizing movement patterns. While cluster analysis is usually a useful starting point for understanding and exploring data, conventional clustering algorithms are not designed for handling trajectory data. Therefore, in this paper, we propose a direction-based clustering (DEN) method, which aims to group trajectories by moving directions. A key development challenge is how to transform direction information into a data format which is appropriate for traditional clustering algorithms to explore. To this end, we partition the space into grids and turn the movement statistics in a grid into a vector which represents the probabilities of moving directions within the grid. With such data transformation, we are able to develop a grid-level K-means clustering method for direction clustering. We illustrate the use of DEN for showing movement patterns and detecting outliers on real-world data sets.

Keywords: Clustering, Data mining, Outlier detection, Trajectory analysis

1. Introduction

Advances in sensors, wireless communication, and information infrastructures such as GPS, WiFi, Video Surveillance, and RFID have enabled us to collect and process realtime massive amounts of fine-grained location traces (trajectory data) from multiple sources. For instance, from GPS trace data, a vehicle's speed and direction of driving can be obtained. Also, the movement of people or cargos within a building or a given area can be observed from the digital traces produced by door access control, video monitors, or RFID tags. Indeed, there is an opportunity to explore location traces to automatically discover useful knowledge, such as identifying suspicious activities, understanding the behaviors of drivers as well as the patterns of transportation networks under extreme conditions, which in turn delivers intelligence for real-time decision making in various fields, such as urban planning, cargo shipment, transportation management, and video surveillance.

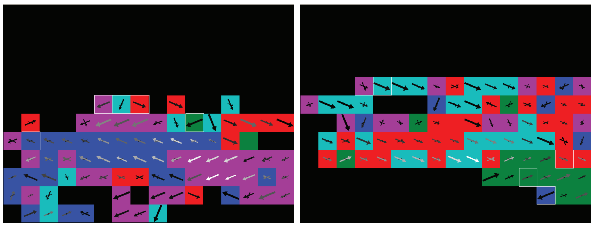
In recent years, recognition of the importance of clustering trajectories has produced a growing body of research [5, 8, 9, 14], aiming to enhance people's understanding about the movement patterns. Several issues, however, have not been well addressed by those studies. First, the above mentioned work for clustering trajectories have the hidden assumption that a complete trajectory belonging to an individual user is available for data analysis. In reality, location traces will be anonymized before they are shared with the researchers for data analysis. For an anonymous location dataset, it is usually not possible for knowing complete trajectories. In most cases, only a collection of directed line segments are available for data analysis. Second, the direction information is not fully utilized in the current cluster analysis. However, the direction information usually indicates the users' real interests. Studying moving directions will help us better understand movement behaviors. Also, the direction information can help to reveal some movement patterns which are difficult to be captured by other geographic information. For instance, it is possible to capture local movement outliers whose location traces are co-located with a large group of other location traces, but have a reverse direction.

In response to the issues noted above, in this paper, we exploit the direction information of each movement and propose a direction-based clustering (DEN) method which aims to divide all the movements into clusters in a way such that movements in a cluster have similar directions and movements in different clusters have different directions. A key development challenge for direction clustering is how to transform direction information into a data format which is appropriate for use in a traditional clustering framework. Along this line, we partition the space into grids and introduce a probabilistic model to turn the direction information of line segments in a grid into a vector with eight values to indicate the probabilities of moving towards eight directions within the grid. With this data transformation, we are able

IEEE IRI 2010, August 4-6, 2010, Las Vegas, Nevada, USA
978-1-4244-8099-9/10/$26.00 ©2010 IEEE

to develop a grid-level constrained clustering method based on K-means for direction clustering.

To illustrate the utility of DEN, we evaluate our direction clustering approach on real-world data sets. As a case study, we first exploit DEN for finding within-cluster outlier traces which have reverse directions with most other traces in the same cluster. As described above, before location traces are published, the anonymization scheme will remove unique ID associated with each trace. However, within-cluster outlier traces identified by our direction clustering method allow people to reconstruct a location trace belonging to an individual user. This raises a new privacy-preserving challenge for publishing data that contain outliers.



(a) morning patterns (b) evening patterns

Figure 1. Sample movement patterns.

Moreover, we show different clustering patterns at different time periods. For instance, Figure 1 shows two direction clustering results in the morning and evening for the same monitoring area. In the figure, we can clearly observe that the moving trends in the morning are opposite to the moving patterns in the evening.

2. The DEN Problem

To illustrate the usefulness of direction-based clustering, we discuss the following examples, as shown in Figure 2.

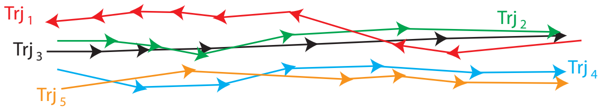


Figure 2. A motivating example: Scenario I.

In Figure 2, there are 5 trajectories, all of which have similar shapes and are located closely. If we do not consider direction information, these trajectories can be clustered together based on Euclidean distance. However, Trj_1 would be an obvious outlier, considering its direction is opposite to all other trajectories. This indicates that distance-based trajectory outlier detection methods cannot capture this type of direction-based outliers.

Furthermore, real-world location traces will be anonymized before these traces become available for use. In other words, we may not have complete trajectories as shown in Figure 2 to use. Indeed, the anonymization algorithms usually go beyond removing the identity information of location traces [3, 7]. Existing anonymization algorithms also remove substantial number of line segments from location traces to avoid the reconstruction of any complete trajectory belonging to a user. However, as shown in Figure 3, the direction-based clustering algorithms allow us to identify direction-based outliers, which in turn, lead to reconstructing traces belonging to the outlier users. As a result, these existing anonymization techniques cannot provide sufficient protection to the privacy of the outlier users and their behaviors can be captured as the direction-based outliers. The above example motivates the study of the direction-based clustering (**DEN**) problem, which is formally defined as follows.

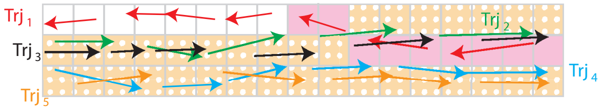


Figure 3. A motivating example: Scenario II.

2.1. Problem Statement

The direction-based clustering (DEN) problem can be formulated as follows. Given a specific time window $\mathcal{T}$, a finite monitoring area $\mathcal{A}$, and a set of time-stamped data points $\mathcal{P}$ recorded in the monitoring area, where $\mathcal{P} = \{(t, x, y) | t \in \mathcal{T}, (x, y) \in \mathcal{A}\}$. The objective of DEN is to partition the monitoring area $\mathcal{A}$ into regions $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_k$, such that the moving directions in the same region are more similar to each other than moving directions in different regions. In the DEN problem, the identity information of location traces may not be available. Also, some portion of observed traces may be removed by the anonymization algorithms to protect privacy. However, in the most cases, the direction information for each movement is available after the data anonymization.

2.2. Related Work

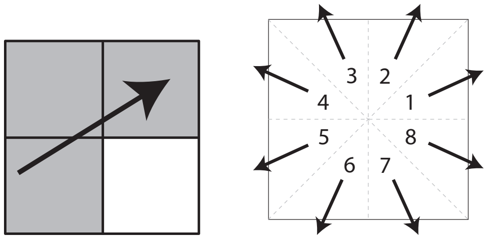
Related work can be roughly grouped into two categories. In the first category, people are more interested in finding interesting patterns in trajectory data by exploring unsupervised learning methods. For example, Giannotti et al. [6] proposed mining frequent behaviors in trajectory data, which are called T-patterns. Also, the ROAM [10] framework transforms original trajectories into a sequence of pattern fragments named motifs, and takes a rule-based approach to discover patterns at multiple levels. In addition, Lee et al. [9] proposed a partition-and-group framework for trajectory clustering. Finally, treating trajectories as documents, and positions as words, [14] proposes co-clustering trajectories and semantic regions with a Bayesian model called Dual Hierarchical Dirichlet Process ($Dual\text{-}HDP$). In the second category, people are interested in trajectory classification. For instance, Lee et al. [8] proposed extracting discriminative features by clusters of trajectory segments, first by geographical regions, if they are identified as homogeneous; and then by distances among trajectory segments using the same distance measure as in [9].

In summary, the related work mentioned above assume that complete traces of moving objects are available. In reality, the identity information of location traces may not be available. Also, some portion of observed traces may be removed by the anonymization algorithms to protect privacy. In contrast, the study in this paper provides a creative way to explore direction information and introduces a new way for direction clustering, even in the anonymized motion data. Furthermore, the clustering results can also be used to identify local and regional outliers.

3. Direction Vectors

In this section, we illustrate how to transform direction information of the movements into a data format which is appropriate for use in a traditional clustering framework. [1]

Let us consider a scenario that a number of moving objects are observed in a fixed region within a certain time window. For many other cases, even if the object IDs are not available, we still have the direction information for each movement, as shown in Figure 4 (a). Then we transform the direction information into vectors by discretizing the continuous direction information as well as the continuous space. Specifically, the monitoring area is divided into small grids.



(a) grid counting (b) direction discretization

Figure 4. Illustrations of data preprocessing.

Once we have partitioned the space into grids, we further partition each grid into 8 direction bins, as shown in Figure 4(b), the angle of each bin has a range of $\pi/4$.[2] Next, we will transform each grid into a direction vector $g = (p_1, p_2, p_3, \ldots, p_8)$, where p_i is the probability of moving towards direction i within this grid. To compute p_i, we first count the frequency f_i of moving objects which have passed this grid and has the direction along the direction i. For example, in Figure 4(a), for the whole monitoring are, a vector is across three grids $(1, 1)$, $(1, 2)$, and $(2, 2)$ along the direction 1 as shown in Figure 4(b). Therefore, the frequency of direction 1 will increase by one for all these three grids. Then, $p_i = f_i / \sum_{k=1}^{8} f_k$. Therefore, p_i is the probability of all the moving objects towards the direction i within this grid.

4. Direction Clustering

In the above section, we have partitioned the monitoring area into grids. Each grid contains a collection of movements and the direction information of all the movements within a grid has been transformed into a vector by a probabilistic model. In other words, the task of clustering trajectories based on their direction information can be transformed to cluster the grids which have been represented by vectors with eight values to indicate the probabilities of moving towards eight directions within the grid. Once the direction information has been transformed into vectors, we can exploit constrained K-means clustering on these vectors and produce a grid-level direction clustering.

4.1. The Distance Measure

We introduce two ways to select the representative unit vectors in each bin: the simple form and the weighted form. Given two direction vectors, $g_1 = (p_1^1, p_2^1, \ldots, p_8^1)$ and $g_2 = (p_1^2, p_2^2, \ldots, p_8^2)$, we have th following difinitions.

Definition 1 *(Simple Distance)* $\mathcal{D}(g_1, g_2) = \sum_{k=1}^{8} |q_k^1 - q_k^2|$*, where* $q_k^j = \sum_{i=1}^{8} p_i^j \cdot cos(v_i, v_k), j \in \{1, 2\}$.

Definition 2 *(Weighted Distance)* $\mathcal{D}(g_1, g_2) = \sum_{k=1}^{8} |q_k^1 - q_k^2| \cdot cos(\theta/2)$*, where* $q_k^j = \sum_{i=1}^{8} p_i^j \cdot cos(v_i^j, v_k^j), j \in \{1, 2\}$ *and* $\theta = \angle(v_k^1, v_k^2)$.

4.2. Constraint-Based K-Means Clustering

K-means [11] is a prototype-based, simple partitional clustering technique which attempts to find the user-specified K clusters. These clusters are represented by their centroids (a cluster centroid is typically the mean of the data objects in that cluster). K-means has an objective function:

$$\mathcal{F}_{kmeans} = \sum_{x_i \in \mathcal{X}} \|x_i - \mu_{l_i}\|^2 \qquad (1)$$

where $l_i (l_i \in \{1, \cdots, K\})$ is the cluster assignment of point x_i and μ_{l_i} represents the centroid of cluster l_i.

[1] Some ideas illustrated in this paper were filed as patent in July, 2006 and are currently patent pending.

[2] Note that we have eight bins for each grid in this paper. However, it is possible to have a different number of bins for each grid.

The performances of K-means clustering can be improved by some enforced appropriate constraints [12]. In general, there are two types of constraints: CANNOT-LINK and MUST-LINK. MUST-LINK means that two objects must be in the same cluster, while CANNOT-LINK means that two objects cannot be in the same cluster. In the real world, this type of partial pairwise constraint information is more practical than providing class labels, because true class labels may be unknown. It can be easier to generate constraints according to background knowledge of the domain. For example, speaker identification in a conversation [1] and lane-finding from GPS data [13].

With constraint information, we can have the objective function of constrained K-means. Let us use $\mathcal{M}$ be a set of must-link pairs where $(x_i, x_j) \in \mathcal{M}$ indicates x_i and x_j should be within the same cluster. Also we use $\mathcal{C}$ to denote a set of cannot-link pairs where $(x_i, x_j) \in \mathcal{C}$ indicates x_i and x_j should be in different clusters. Let $W = w_{ij}$ and $\bar{W} = \bar{w}_{ij}$ be penalty cost for violating the constraints in $\mathcal{M}$ and $\mathcal{C}$ respectively. Then the objective function of constraint-based clustering can be formulated [2] as the following:

$$\begin{aligned} \mathcal{F}_{constr} &= \sum_{x_i \in \mathcal{X}} \|x_i - \mu_{l_i}\|^2 + \sum_{(x_i,x_j)\in\mathcal{M}} w_{ij}\mathbb{I}[l_i \neq l_j] \\ &\quad + \sum_{(x_i,x_j)\in\mathcal{C}} \bar{w}_{ij}\mathbb{I}[l_i = l_j] \qquad (2) \end{aligned}$$

where $\mathbb{I}$ is the indicator function, $\mathbb{I}[true] = 1$ and $\mathbb{I}[false] = 0$. Several heuristic methods are available for finding the optimal solution of this objective function [4].

4.3. Constraints Generation

In the real-world applications, we can use semantic information to generate constraints. For instance, there are many twists and turns in the road systems. We can enforce all the grids along the turns as the MUST-LINK constraints. However, in this study, we do not have the semantic information which are ready to exploit for generating constraints. Instead, we explore geographical proximity and direction consistencies for generating some MUST-LINK constraints. Geographical proximity means that grids in the same cluster are adjacent geographically. Specifically, for one grid, we only consider the 8 adjacent grids of this grid. Direction consistencies refer to grids of the same cluster should have similar direction. In other words, these grids should have high similarity as measured by the distance metrics proposed in subsection 4.1. To maintain direction consistence, we use a parameter to control the selection of grids as the MUST-LINK constraints.

Finally, Figure 5 shows the pseudo-code of the direction based cluster using the constraint-based K-means.

```
ALGORITHM GridClustering($\mathcal{G}$,$\lambda$,$k$)
Input:  $\mathcal{G}$: the fine-cut grids with direction vectors and
            average directions;
        $\lambda$: the threshold for determining initial constraints;
        $k$: the target number of clusters.
Output: $C$: the cluster labels for all non-zero grids.

Initializations;
for each neighboring grid pair $g_1, g_2$ do
    if $\mathcal{D}(g_1, g_2) >= \lambda$ then
        Record $\langle g_1, g_2 \rangle$ as a MUST-LINK
    end
end
$C \leftarrow ConstrKMeans(\mathcal{G}, k,$MUST-LINKs$)$
```

Figure 5. Grid-Level Direction Clustering by Constrained K-means

5. Experimental Results

In the experiments, we have used two real-world motion data sets: one from MIT [14] and another one is from a major IT vendor. Each data set contains a number of trajectories derived from several surveillance cameras. Each trajectory is a sequence of time-stamped positions, indicating the location of an object at a certain time. In our experiments, we first anonymize the data and extract the direction information for each movement left after the data anonymization. Also, we keep the identifications of original trajectories for validating the results of outlier detection and location trace reconstruction.

With the movement vectors, we count the frequency of each direction of each grid, and transform the original data into a number of grids, each of which has a probabilistic direction vector.

5.1. Movement Patterns

The direction clustering results at different time periods allow us to quickly summarize the overall movement trends. At the same location, different moving directions can be observed during different time windows. For instance, Figure 1 shows two direction clustering results, in the morning and in the evening, respectively. The movement data is derived from the video collected by a surveillance camera over a parking lot. For the purpose of privacy protection, we do not show the background image. In the morning, most of the moving objects are moving to the left, while in the afternoon, the majority move towards the right. This indicates that most people come to the office about the same morning time period and leave the office about the same evening time period. The direction clustering results also show that less moving activities have been observed from 9:30AM to 11:30AM and the moving activities during this time period have weak clustering effects. This may indicate a high work-efficient time period.

In addition, as illustrated by the colors of the grids, the clustering results show possible paths and other semantic regions. The edges of the clusters are likely to be important landmarks. For example, the edges, the turning point, and the double yellow line of the road, etc. Let us take Figure 6 as an example, the grids colored in purple have scattered directions, which may represent pedestrian area, where moving objects do not follow a uniform direction. While the subregions in the rectangles and ellipses are segments on the road. From the background image of Figure 6, we can clearly see that there are double yellow lines in the center of the road, so the traffic on both sides tend to have opposite moving directions. Finally, since there is a turning point on the road, the same traffic following one side of the road is cut into two part, due to the change of moving directions.

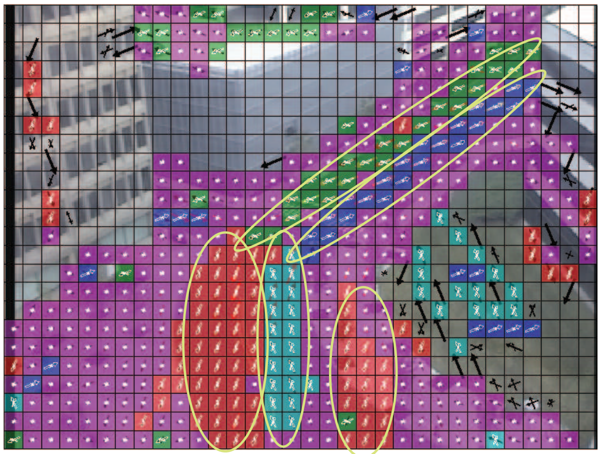


Figure 6. Possible landmarks.

5.2. Outlier Detection: A Case Study

There are two different levels of outliers as shown in Section 2. Local outliers are determined within each grid. If the total frequency of a grid is very low, we consider this grid as a grid outlier; on the other hand, if the grid has considerable observations, which has an overall moving trend, we can evaluate the likelihood of one moving segment and see if it follows the major movement direction. Regional outliers are movement segments that are different from the cluster direction of a direction cluster. In Figure 7, we show the effectiveness of direction clustering for identifying outliers.

In Figure 7, the left part (a) shows the overall clustering results based on the MIT motion data (C200_nw2). It is shown that in the center there is a cluster colored in bright blue. We specifically zoom into this region, and show all the movement segments in the selected region in Figure (b). The normal ones, which represent the majority, are shown in the gray color. We can observe the dense region

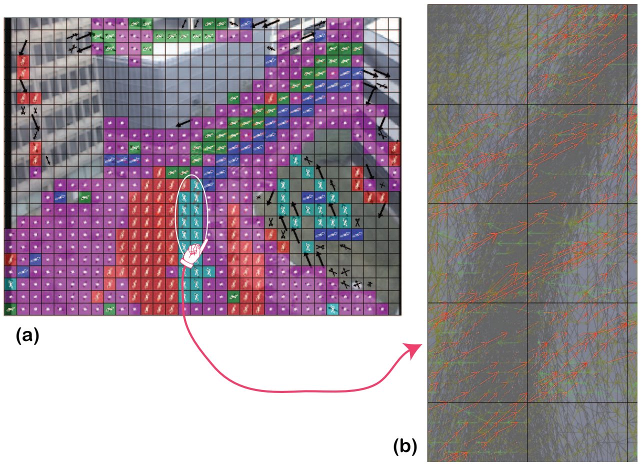


Figure 7. Outlier detection.

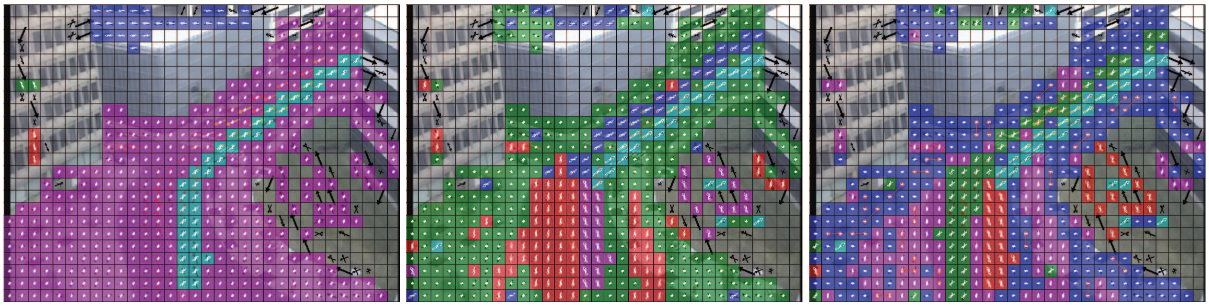
of gray, showing the pattern of moving upwards, slightly to the right. Local outliers (but not regional outliers) are shown in bright green, which overall move towards the left, are almost perpendicular to the gray ones. Regional outliers (but not local ones) are shown in red, which interestingly reconstruct a number of connected paths, moving towards northeast. These local and regional outliers can allow the users to reconstruct location traces belonging to specific users. Indeed, if a outlier trace has been combined together with some semantic maps, it is possible to identify a real person in the real-life. This certainly raise serious privacy issues for even the anonymized data.

5.3. A Comparison of Algorithms

In Figure 8, we illustrate a comparison of different direction clustering methods including $Greedy$, K-$means$, and $Constrained\ K$-$Means$.

The greedy method works in a typical way of hierarchical clustering. Initially, each grid is an individual “cluster”. After calculating the similarities between each pair of neighboring grids, we merge two grids with the highest similarity, and recompute their combined direction vector by a weighted mean. The weights are determined by their original cluster sizes. This process goes iteratively until the number of clusters reduces to a target number. The grid clustering results by the greedy method is shown in Figure 8 (a). Meanwhile, Figure 8 (b) shows the grid-level direction clustering results using K-means clustering.

By comparing Figures 8 (a) and 8 (b), we can see that the greedy method merges the grids into 5 connected groups. The cluster colored in purple spread over a large range and do not provide enough details. This greedy method is sensitive to noise, since the outlier grids (the isolated ones on the top and the left) are put into stand-alone clusters. K-means, however, can separate the region with more details, such as



(a) greedy method (b) K-means (c) constrained K-means

Figure 8. A comparison of clustering algorithms.

the two directions of the road, and the turning of the road. Although the K-means algorithm is specified to produce 5 clusters, these clusters are based simply on the similarity of direction vectors instead of Euclidean distance. So the same cluster may distribute into several disconnected subregions. In real-world applications, we can split the geographically isolated subregions into separate clusters, as a post processing step. Or we can specify some CANNOT-LINK constraints for K-means clustering.

Finally, Figure 8 (c) illustrates an example of K-means clustering with MUST-LINK constraints. The constraints are shown with red connections between grids. The ones shown are generated by connecting grids pairs that are neighboring each other and their distance is below the threshold 0.005. Our observation indicates that it is more useful to generate MUST-LINK constraints along the twist and turns in the road systems. These constraints avoid breaking a line shape direction cluster into several isolated smaller clusters.

6. Concluding Remarks

In this paper, we introduced a direction-based clustering method (DEN) for characterizing movement patterns in motion data. This research moves beyond past studies of trajectory data by filling two research gaps. First, DEN can deal with a collection of directed line segments without the requirement of knowing the identity of a complete trajectory. Second, we exploit a probabilistic model to transform the direction of line segments into a form that can be easily explored by traditional clustering algorithms (e.g. K-means).

Our empirical studies, which apply DEN to two real-world motion data sets, suggest the need of direction-based clustering. Specifically, the results show that DEN can effectively capture direction outliers in motion data and help us understand movement patterns. Finally, as a case study, we point out a new privacy-preserving challenge for publishing location traces that contain direction outliers. In other words, direction outliers identified in clusters can allow the users to reconstruct some outlier location traces.

Acknowledgements This research was partially supported by Panasonic Inc. and the National Science Foundation (NSF) via grant number CNS 0831186.

References

[1] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relations. In *ICML*, pages 11–18, 2003.

[2] S. Basu, A. Banerjee, and R. J. Mooney. Active semi-supervision for pairwise constrained clustering. In *SDM*, 2004.

[3] A. R. Beresford and F. Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[4] I. Davidson, S. S. Ravi, and M. Ester. Efficient incremental constrained clustering. In *KDD*, pages 240–249, 2007.

[5] S. Gaffney and P. Smyth. Trajectory clustering with mixtures of regression models. In *KDD*, pages 63–72, 1999.

[6] F. Giannotti, M. Nanni, F. Pinelli, and D. Pedreschi. Trajectory pattern mining. In *KDD*, pages 330–339, 2007.

[7] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady. Preserving privacy in gps traces via uncertainty-aware path cloaking. In *CCS*, pages 161–171, 2007.

[8] J.-G. Lee, J. Han, X. Li, and H. Gonzalez. Traclass: Trajectory classification using hierarchical region-based and trajectory-based clustering. *VLDB Endowment*, 1(1):1081–1094, 2008.

[9] J.-G. Lee, J. Han, and K.-Y. Whang. Trajectory clustering: a partition-and-group framework. In *SIGMOD*, pages 593–604, 2007.

[10] X. Li, J. Han, S. Kim, and H. Gonzalez. Roam: Rule-and motif-based anomaly detection in massive moving object data sets. In *SDM*, 2007.

[11] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Berkeley Sympo. on Math. Stats. and Prob.*, pages 281–297, 1967.

[12] W. Tang, H. Xiong, S. Zhong, and J. Wu. Enhancing semi-supervised clustering: a feature projection perspective. In *KDD*, pages 707–716, 2007.

[13] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. In *ICML*, pages 577–584, 2001.

[14] X. Wang, K. T. Ma, G.-W. Ng, and W. E. L. Grimson. Trajectory analysis and semantic region modeling using a non-parametric bayesian model. In *CVPR*, 2008.