Spatio-temporal clustering

Slava Kisilevich, Florian Mansmann, Mirco Nanni, Salvatore Rinzivillo

Summary. Spatio-temporal clustering is a process of grouping objects based on their spatial and temporal similarity. It is relatively new subfield of data mining which gained high popularity especially in geographic information sciences due to the pervasiveness of all kinds of location-based or environmental devices that record position, time or/and environmental properties of an object or set of objects in real-time. As a consequence, different types and large amounts of spatio-temporal data became available that introduce new challenges to data analysis and require novel approaches to knowledge discovery. In this chapter we concentrate on the spatio-temporal clustering in geographic space. First, we provide a classification of different types of spatio-temporal data. Then, we focus on one type of spatio-temporal clustering - trajectory clustering, provide an overview of the state-of-the-art approaches and methods of spatio-temporal clustering and finally present several scenarios in different application domains such as movement, cellular networks and environmental studies.

44.1 Introduction

Geographic and temporal properties are a key aspect of many data analysis problems in business, government, and science. Through the availability of cheap sensor devices we have witnessed an exponential growth of geo-tagged data in the last few years resulting in the availability of fine-grained geographic data at small temporal sampling intervals. Therefore, the actual challenge in geo-temporal analysis is moving from acquiring the right data towards large-scale analysis of the available data.

Clustering is one approach to analyze geo-temporal data at a higher level of abstraction by grouping the data according to its similarity into meaningful clusters. While the two dimensional geographic dimensions are relatively manageable, their combination with time results in a number of challenges. It is mostly application dependent how the weight of the time dimension should be considered in a distance metric. When tracking pedestrians, for example, two geographically close sample points co-occurring within a minute interval could belong to the same cluster, whereas two sample points at near distance within a time interval of a few nanoseconds in a physics experiment might belong to different clusters. In addition to this, representing temporal information on a map becomes extremely challenging.

When considering a group of points in time as a single entity, more complex data types such as trajectories emerge. Analysis questions might then deal with the correlation of these trajectories among each others, resulting in extraction of patterns such as important places from trajectories or clustering of trajectories with common features.

Yet on a higher level, the problem of moving clusters arises. An exemplary analysis question might therefore be if there are groups of commuters within a city that move from one area of the city to another one within a particular time frame. This kind of analysis can give meaningful hints to city planners in order to avoid regular traffic jams.

The rest of this chapter first details basic concepts of spatio-temporal clustering and then lists a number of applications for spatio-temporal clustering found in the literature. Afterwards, we identify open issues in spatio-temporal clustering with a high need for future research. Finally, the last section summarizes our view on spatio-temporal clustering.

44.2 Spatio-temporal clustering

Whatever the analysis objective or the computational schema adopted, the clustering task heavily depends on the specific characteristics of the data considered. In particular, the spatio-temporal context is a large container, which includes several kinds of data types that exhibit extremely different properties and offer sensibly different opportunities of extracting useful knowledge. In this section we provide a taxonomy of the data types that are available in the spatio-temporal domain, briefly describe each class of data with a few examples taken from the spatio-temporal clustering literature, and finally report in detail the state-of-art of clustering methods for a particular kind of data – trajectories – that constitute the main focus of this chapter.

44.2.1 A classification of spatio-temporal data types

Several different forms of spatio-temporal data types are available in real applications. While they all share the availability of some kind of spatial and temporal aspects, the extent of such information and the way they are related can combine to several different kinds of data objects. Figure 44.1 visually depicts a possible classification of such data types, based on two dimensions:

- the *temporal dimension* describes to which extent the evolution of the object is captured by the data. The very basic case consists of objects that do not evolve at all, in which case only a static snapshot view of each object is available. In slightly more complex contexts, each object can change its status, yet only its most recent value (i.e., an updated snapshot) is known, therefore without any knowledge about its past history. Finally, we can have the extreme case where the full history of the object is kept, thus forming a time series of the status it traversed;
- the *spatial dimension* describes whether the objects considered are associated to a fixed location (e.g., the information collected by sensors fixed to the ground) or they can move, i.e., their location is dynamic and can change in time.

In addition to these two dimensions, a third, auxiliary one is mentioned in our classification, which is related to the spatial extension of the objects involved. The simplest case, which is also the most popular in real world case studies, considers point-wise objects, while more complex cases can take into consideration objects with an extension, such as lines and areas. In particular, Figure 44.1 focuses on point-wise objects, while their counterparts with spatial extension are omitted for the sake of presentation.

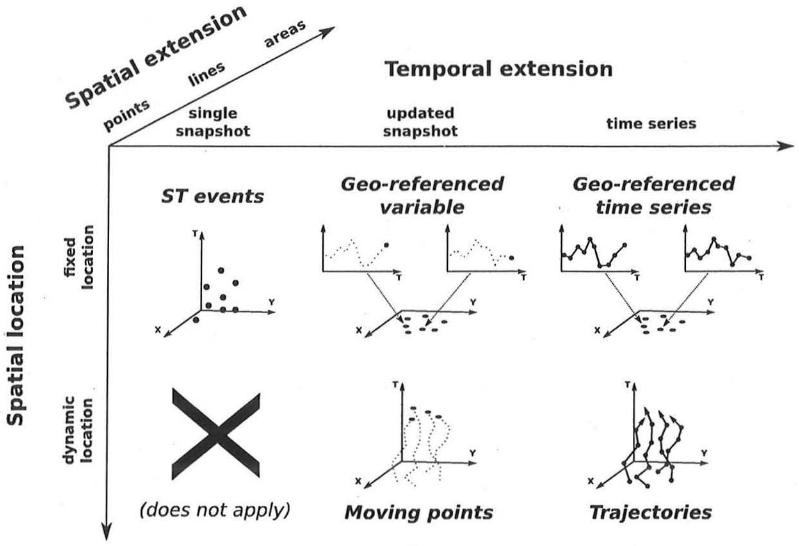

Fig. 44.1. Context for ST Clustering

In the following we briefly describe the main classes of data types we obtain for point-wise objects.

ST events. A very basic example of spatio-temporal information are spatio-temporal events, such as earth tremors captured by sensors or geo-referenced records of an epidemic. Each event is usually associated with the location where it was recorded and the corresponding timestamp. Both the spatial and the temporal information associated with the events are static, since no movement or any other kind of evolution is possible. Finding clusters among events means to discover groups that lie close both in time and in space, and possibly share other non-spatial properties. A classical example of that is (Kulldorff(1997))'s spatial scan statistics, that searches spatio-temporal cylinders (i.e., circular regions considered within a time interval) where the density of events of the same type is higher than outside, essentially representing areas where the events occurred consistently for a significant amount of time. In some applications, such as epidemiology, such area is expected to change in size and location, therefore extensions of the basic scan statistics have been proposed that consider shapes different from simple cylinders. For instance, (Iyengar(2004)) introduces (reversed) pyramid shapes, representing a small region (the pinpoint of the pyramid, e.g. the origin of an epidemic) that grows in time (the enlarging section of the pyramid, e.g. the progressive outbreak) till reaching its maximal extension (the base of the pyramid). From another viewpoint, (Wang et al(2006)Wang, Wang, and Li) proposed two spatio-temporal clustering algorithms (ST-GRID and ST-DBSCAN) for analysis of sequences of seismic events. ST-GRID is based on partitioning of the spatial and temporal dimensions into cells. ST-DBSCAN is an extension of the DBSCAN algorithm to handle spatio-temporal clustering. The k-dist graph proposed in (Ester et al(1996)Ester, Kriegel, Sander, and Xu) as a heuristic for determination of the input parameters was used in both approaches. Hence, in the first step, the *k*-dist graph was created using spatial and temporal dimensions. By means of the graph, the analyst could infer the suitable thresholds for the spatial and temporal cell lengths. In the second step, the inferred cell lengths are provided to ST-GRID algorithm as an input and the dense clusters are extracted. ST-DBSCAN introduced the second parameter of the neighborhood radius in addition to the spatial neighborhood radius ε, namely temporal neighborhood radius ε_t. These two parameters were determined using *k*-dist graph and provided to ST-DBSCAN as an input. Thus, point p is considered as *core* when the number of points in the neighborhood is greater or equal to the threshold $MinPts$ within spatial and temporal thresholds.

Geo-referenced variables. When it is possible to observe the evolution in time of some phenomena in a fixed location, we have what is usually called a geo-referenced variable, i.e., the time-changing value of some observed property. In particular, the basic settings might allow only to remember the most recent value of such variable. In this case, the clustering task can be seen as very similar to the case of events discussed above, with the exception that the objects compared refer to the same time instant (the actual time) and their non-spatial features (variables) are not constant. A typical problem in this context consists in efficiently computing a clustering that (i) takes into account both the spatial and non-spatial features, and (ii) exploits the clusters found at the previous time stamp, therefore trying to detect the relevant changes in the data and incrementally update the clusters, rather than computing them from scratch.

Geo-referenced time series. In a more sophisticated situation, it might be possible to store the whole history of the evolving object, therefore providing a (geo-referenced) time-series for the measured variables. When several variables are available, they are usually seen as a single, multidimensional time series. In this case, clustering a set of objects requires to compare the way their time series evolve and to relate that to their spatial position. A classical problem consists in detecting the correlations (and therefore forming clusters) among different time series trying to filter out the effects of spatial autocorrelation, i.e., the mutual interference between objects due to their spatial proximity, e.g., (Zhang et al(2003)Zhang, Huang, Shekhar, and Kumar). Moreover, spatio-temporal data in the form of sequences of images (e.g., fields describing pressure and ground temperature, remotely sensed from satellites) can be seen as a particular case where location points are regularly distributed in space along a grid.

Moving objects. When (also) the spatial location of the data object is time-changing, we are dealing with moving objects. In the simplest case, the available information about such objects consists in their most recent position, as in the context of real-time monitoring of vehicles for security applications, and no trace of the past locations is kept. As in the case of geo-referenced variables, a typical clustering problem in this context consists in keeping an up-to-date set of clusters through incremental update from previous results, trying to detect the recent changes in the data (in particular, their recent movements) that were significant or that are likely to be followed by large changes in the close future, e.g., due to a change of heading of the object. An example is provided by the work in (Li et al(2004a)Li, Han, and Yang), where a *micro-clustering* technique based on direction and speed of objects is applied to achieve a large scalability.

Trajectories. When the whole history of a moving object is stored and available for analysis, the sequence of spatial locations visited by the object, together with the time-stamps of such visits, form what is called a *trajectory*. Trajectories describe the movement behavior of objects, and therefore clustering can be used to detect groups of objects that behaved in a similar way, for instance by following similar paths (maybe in different time periods), by moving consistently together (i.e., keeping close to each other for long time intervals) or by sharing other properties of movement. Recent literature is relatively rich of examples in this area, which will be the focus of this chapter and will be described in detail in the following sections.

Analogous classes of data types can be obtained through similar combination of the temporal and spatial properties on objects that possess a spatial extension, such as lines (e.g., road segments) and areas (e.g., extension of a tornado). In these cases, a dynamic spatial attribute can result not only to movement, but also to a change of shape and size. Due to the limited availability of this form of information in real scenarios and the absence of studies of specific analysis methods – especially for the dynamic cases – these contexts will not be further examined in this chapter, which instead will focus on point-wise objects in the richest setting, i.e., trajectories of moving objects.

44.2.2 Clustering Methods for Trajectory Data

Here we will focus on the context of moving objects that can be traced along the time, resulting in trajectories that describe their movements. On one hand, trajectories represent the most complex and promising (from a knowledge extraction viewpoint) form of data among those based on point-wise information. On the other hand, point-wise information is becoming nowadays largely available and usable in real contexts, while spatio-temporal data with more complex forms of spatial components are still rarely seen in real world problems – exception made for a few, very specific contexts, such as climate monitoring.

Clustering is one of the general approaches to a descriptive modeling of a large amount of data, allowing the analyst to focus on a higher level representation of the data. Clustering methods analyze and explore a dataset to associate objects in groups, such that the objects in each groups have common characteristics. These characteristics may be expressed in different ways: for example, one may describe the objects in a cluster as the population generated by a joint distribution, or as the set of objects that minimize the distances from the centroid of the group.

Descriptive and generative model-based clustering

The objective of this kind of methods is to derive a global model capable of describing the whole dataset. Some of these methods rely on a definition of multivariate density distribution and look for a set of fitting parameters for the model. In (Gaffney and Smyth(1999)) it is proposed a clustering method based on a mixture model for continuous trajectories. The trajectories are represented as functional data, i.e. each individual is modeled as a sequence of measurement given by a function of time depending on a set of parameters that models the interaction of the different distributions. The objects that are likely to be generated from a core trajectory plus gaussian noise are grouped together by means of the EM algorithm. In a successive work (Chudova et al(2003)Chudova, Gaffney, Mjolsness, and Smyth), spatial and temporal shift of trajectories within each cluster is also considered. Another approach based on a model-based technique is presented in (Alon et al(2003)Alon, Sclaroff, Kollios, and Pavlovic), where the representative of a cluster is expressed by means of a Markov model that estimates the transition between successive positions. The parameter estimation task for the model is performed by means of EM algorithm.

Distance-based clustering methods

Another approach to cluster complex form of data, like trajectories, is to transform the complex objects into features vectors, i.e. a set of multidimensional vectors where each dimension represents a single characteristic of the original object, and then to cluster them using generic clustering algorithms, like, for example, k-means. However, the complex structure of the trajectories not alway allows an approach of this kind, since most of these methods require that all the vectors are of equal length. In contrast to this, one of the largely adopted approach to the clustering of trajectories consists in defining distance functions that encapsulate the concept of similarity among the data items.

Using this approach, the problem of clustering a set of trajectories can be reduced to the problem of choosing a generic clustering algorithm, that determines how the trajectories are joined together in a cluster, and a distance function, that determines which trajectories are candidate to be in the same group. The chosen method determines also the "shape" of the resulting clusters: center-based clustering methods, like *k-means*, produce compact, spherical clusters around a set of centroids and are very sensitive to noisy outliers; hierarchical clusters organize the data items in a multi-level structure; density-based clustering methods form maximal, dense clusters, not limiting the groups number, the groups size and shape.

The concepts of similarities of spatio-temporal trajectories may vary depending on the considered application scenario. For example, two objects may be considered similar if they have followed the same spatio-temporal trajectory within a given interval, i.e. they have been in the same places at the same times. However, the granularity of the observed movements (i.e. the number of sampled spatio-temporal points for each trajectory), the uncertainty on the measured points, and, in general, other variations of the availability of the locations of the two compared objects have required the definition of several similarity measures for spatio-temporal trajectories. The definition of these measures is not only tailored to the cluster analysis task, but it is strongly used in the field of Moving Object Databases for the similarity search problem (Theodoridis(2003)), and it is influenced also by the work on time-series analysis (Agrawal et al(1993)Agrawal, Faloutsos, and Swami, Berndt and Clifford(1996), Chan and chee Fu(1999)) and Longest Common Sub Sequence (LCSS) model (Vlachos et al(2002)Vlachos, Kollios, and Gunopulos, Vlachos et al(2003)Vlachos, Hadjieleftheriou, Gunopulos, and Keogh, Chen et al(2005)Chen, Özsu, and Oria). The distance functions defined in (Nanni and Pedreschi(2006), Pelekis et al(2007)Pelekis, Kopanakis, Marketos, Ntoutsi, Andrienko, and Theodoridis) are explicitly defined on the trajectory domain and take into account several spatio-temporal characteristics of the trajectories, like direction, velocity and co-location in space and time.

Density-based methods and the DBSCAN family

The density-based clustering methods use a density threshold around each object to distinguish the relevant data items from noise. DBSCAN (Ester et al(1996)Ester, Kriegel, Sander, and Xu), one of the first example of density-based clustering, visits the whole dataset and tags each object either as *core object* (i.e. an object that is definitively within a cluster), *border object* (i.e. objects at the border of a cluster), or *noise* (i.e. objects definitively outside any cluster). After this first step, the core objects that are close each other are joined in a cluster. In this method, the density threshold is espressed by means of two parameters: a maximum radius ε around each object, and a minimum number of objects, say *MinPts*, within this interval. An object p is defined a *core object* if its neighborhood of radius ε (denoted as $N_\varepsilon(p)$) contains at least *MinPts* objects. Using the core object condition, the input dataset is scanned and the status of each object is determined. A cluster is determined both by its core objects and the objects that are reachable from a core object, i.e. the objects that do not satisfy the core object condition but that are contained in the Eps-neighborhood of a core object. The concept of "reachable" is express in terms of the *reachability distance*. It is possible to define two measures of distances for a core object c and an object in its ε-neighborhood: the *core distance*, which is the distance of the *MinPts*-th object in the neighborhood of c in order of distance ascending from c, and the *reachability distance*, i.e. the distance of an object p from c except for the case when p's distance is less than the *core distance*; in this case the distance is normalized to the *core distance*. Given a set of *core* and *border* object for a dataset, the clusters are formed by visiting all the objects, starting from a core point: the cluster formed by the single point is extended by including other objects that are within a reachability distance; the process is repeated by including all the objects reachable by the new included items, and so on. The growth of the cluster stops when all the border points of the cluster have been visited and there are no more reachable items. The visit may continue from another core object, if avaiable.

The OPTICS method (Ankerst et al(1999)Ankerst, Breunig, Kriegel, and Sander) proceeds by exploring the dataset and enumerating all the objects. For each object p it checks if the core object conditions are satisfied and, in the positive case, starts to enlarge the potential cluster by checking the condition for all neighbors of p. If the object p is not a core object, the scanning process continues with the next unvisited object of D. The results are summarized in a reachability plot: the objects are represented along the horizontal axis in the order of visiting them and the vertical dimension represents their reachability distances. Intuitively, the reachability distance of an object p_i corresponds to the minimum distance from the set of its predecessors p_j, $0 < j < i$. As a consequence, a high value of the reachability distance roughly means a high distance from the other objects, i.e. indicates that the object is in a sparse area. The actual clusters may be determined by defining a reachability distance threshold and grouping together the consecutive items that are below the chosen threshold in the plot. The result of the OPTICS algorithm is insensitive to the original order of the objects in the dataset. The objects are visited in this order only until a core object is found. After that, the neighborhood of the core object is expanded by adding all density-connected objects. The order of visiting these objects depends on the distances between them and not on their order in the dataset. It is also not important which of density-connected objects will be chosen as the first core object since the algorithm guarantees that all the objects will be put close together in the resulting ordering. A formal proof of this property of the algorithm is given in (Ester et al(1996)Ester, Kriegel, Sander, and Xu).

It is clear that the density methods strongly rely on an efficient implementation of the neighborhood query. In order to improve the performances of such algorithms it is necessary to have the availability of valid index data structure. The density based algorithms are largely used in different context and they take advantages of many indices like R-tree, kd-tree, etc. When dealing with spatio-temporal data, it is necessary to adapt the existing approaches also for the spatio-temporal domain (Frentzos et al(2007)Frentzos, Gratsias, and Theodoridis) or use a general distance based index (e.g. M-tree, (Ciaccia et al(1997)Ciaccia, Patella, and Zezula))

The approach of choosing a clustering method and a distance function is just a starting point for a more evolute approach to mining. For example, in (Nanni and Pedreschi(2006)) the basic notion of the distance function is exploited to stress the importance of the temporal characteristics of trajectories. The authors propose a new approach called *temporal focusing* to better exploit the temporal aspect and improve the quality of trajectory clustering. For example, two trajectories may be very different if the whole time interval is considered. However, if only a small sub-interval is considered, these trajectories may be found very similar. Hence, it is very crucial for the algorithm to efficiently work on different spatial and temporal granularities. As mentioned by the authors, usually some parts of trajectories are more important than others. For example, in rush hours it can be expected that many people moving from home to work and viceversa form movement patterns that can be grouped together. On weekends, people's activity can be less ordered where the local distribution of people is more influential than collective movement behavior. Hence, there is a need for discovering the most interesting time intervals in which movement behavior can be organized into meaningful clusters. The general idea of the time focusing approach is to cluster trajectories using all possible time intervals (time windows), evaluate the results and find the best clustering. Since the time focusing method is based on OPTICS, the problem of finding the best clusters converges to finding the best input parameters. The authors proposed several quality functions based on density notion of clusters that measures the quality of the produced clustering and are expressed in terms of average reachability (Ankerst et al(1999)Ankerst, Breunig, Kriegel, and Sander) with respect to a time interval I and reachability threshold ε'. In addition, ways of finding optimal values of ε' for every time interval I were provided.

Visual-aided approaches

Analysis of movement behavior is a complex process that requires understanding of the nature of the movement and phenomena it incurs. Automatic methods may discover interesting behavioral patterns with respect to the optimization function but it may happen that these patterns are trivial or wrong from the point of view of the phenomena that is under investigation. The visual analytics field tries to overcome the issues of automatic algorithms introducing frameworks implementing various visualization approaches of spatio-temporal data and proposing different methods of analysis including trajectory aggregation, generalization and clustering (Andrienko and Andrienko(2006), Andrienko et al(2007)Andrienko, Andrienko, and Wrobel, Andrienko and Andrienko(2008), Andrienko et al(2009)Andrienko, Andrienko, Rinzivillo, Nanni, Pedreschi, and Giannotti,Andrienko and Andrienko(2009)). These tools often target different application domains (movement of people, animals, vehicles) and support many types of movement data (Andrienko et al(2007)Andrienko, Andrienko, and Wrobel). The advantages of visual analytics in analysis of movement data is clear. The analyst can control the computational process by setting different input parameters, interpret the results and direct the algorithm towards the solution that better describes the underlying phenomena.

In (Rinzivillo et al(2008)Rinzivillo, Pedreschi, Nanni, Giannotti, Andrienko, and Andrienko) the authors propose progressive clustering approach to analyze the movement behavior of objects. The main idea of the approach is the following. The analyst or domain expert progressively applies different distance functions that work with spatial, temporal, numerical or categorical variables on the spatio-temporal data to gain understanding of the underlying data in a stepwise manner. This approach is orthogonal to commonly used approaches in machine learning and data mining where the distance functions are combined together to optimize the outcome of the algorithm.

Micro clustering methods

In (Hwang et al(2005)Hwang, Liu, Chiu, and Lim) a different approach is proposed, where trajectories are represented as piece-wise segments, possibly with missing intervals. The proposed method tries to determine a *close time interval*, i.e. a maximal time interval where all the trajectories are pair-wise close to each other. The similarity of trajectories is based on the amount of time in which trajectories are close and the mining problem is to find all the trajectory groups that are close within a given threshold.

A similar approach based on an extension of *micro-clustering* is proposed in (Li et al(2004b)Li, Han, and Yang). In this case, the segments of different trajectories within a given rectangle are grouped together if they occur in similar time intervals. The objective of the method is to determine the maximal group size and temporal dimension within the threshold rectangle.

In (Lee et al(2007)Lee, Han, and Whang), the trajectories are represented as sequences of points without explicit temporal information and they are partitioned into a set of quasi-linear segments. All the segments are grouped by means of a density based clustering method and a representative trajectory for each cluster is determined.

Flocks and convoy

In some application domains there is a need in discovering group of objects that move together during a given period of time. For example, migrating animals, flocks of birds or convoys of vehicles. (Kalnis et al(2005)Kalnis, Mamoulis, and Bakiras) proposed the notion of *moving clusters* to describe the problem of discovery of sequence of clusters in which objects may leave or enter the cluster during some time interval but having the portion of common objects higher than a predefined threshold. Other patterns of moving clusters were proposed in the literature: (Gudmundsson and van Kreveld(2006), Vieira et al(2009)Vieira, Bakalov, and Tsotras) define a flock pattern, in which the same set of objects stay together in a circular region of a predefined radius, while (Jeung et al(2008)Jeung, Yiu, Zhou, Jensen, and Shen) defines a convoy pattern, in which the same set of objects stay together in a region of arbitrary shape and extent.

(Kalnis et al(2005)Kalnis, Mamoulis, and Bakiras) proposed three algorithms for discovery of moving clusters. The basic idea of these algorithms is the following. Assuming that the locations of each object were sampled at every timestamp during the lifetime of the object, a snapshot $S_{t=i}$ of objects' positions is taken at every timestamp $t = i$. Then, DBSCAN (Ester et al(1996)Ester, Kriegel, Sander, and Xu), a density-based clustering algorithm, is applied on the snapshot forming clusters $c_{t=i}$ using density constraints of *MinPts* (minimum points in the neighborhood) and ε (radius of the neighborhood). Having two snapshots clusters $c_{t=i}$ and $c_{t=i+1}$, the moving cluster $c_{t=i}c_{t=i+1}$ is formed if $\frac{|c_{t=i} \cap c_{t=i+1}|}{|c_{t=i} \cup c_{t=i+1}|} > \theta$, where θ is an integrity threshold between 0 and 1.

(Jeung et al(2008)Jeung, Yiu, Zhou, Jensen, and Shen) adopts DBSCAN algorithm to find candidate convoy patterns. The authors proposed three algorithms that incorporate trajectory simplification techniques in the first step. The distance measures are performed on the segments of trajectories as opposed to commonly used point based distance measures. They show that the clustering of trajectories at every timestamp as it is performed in moving clusters is not applicable to the problem of convoy patterns because the global integrity threshold θ may be not known in advance and time constraint (lifetime) is not taken into account, which is important in convoy patterns. Another problem is related to the trajectory representation: Some trajectories may have missing timestamps or be measured at different time intervals. Therefore, the density measures cannot be applied between trajectories with different timestamps. To handle the problem of missing timestamps, the authors proposed to interpolate the trajectories creating virtual time points and apply density measures on segments of the trajectories. Additionally, the convoy was defined as candidate when it had at least k clusters during k consequent timestamps.

Five on-line algorithms for discovery flock patterns in spatio-temporal databases were presented in (Vieira et al(2009)Vieira, Bakalov, and Tsotras). The flock pattern Φ is defined as the maximal number of trajectories and greater or equal to density threshold μ that move together during minimum time period δ. Additionally, the disc with radius $\varepsilon/2$ with the center $c_k^{t_i}$ of the flock k at time t_i should cover all the points of flock trajectories at time t_i. All the algorithms employ the grid-based structure. The input space is divided into cells with edge size ε. Every trajectory location sampled at time t_i is placed in one of the cells. After processing all the trajectories at time t_i, a range query with radius ε is performed on every point p to find neighbor points whose distance from p is at most ε and the number of neighbor points is not less than μ. Then, for every pairs of points found, density of neighbor points with minimum radius $\varepsilon/2$ is determined. If the density of a disk is less than μ, the disk is discarded otherwise the common points of two valid disks are found. If the number of common points is above the threshold then the disk is added to a list of candidate disks. In the basic algorithm that generate flock patterns, the candidate disk at time t_i is compared to the candidate disk at time t_{i-1} and augmented together if they have the common number of points above the threshold. The flock is generated if the augmented clusters satisfy the time constraint δ. In other four proposed algorithms, different heuristics were applied to speed up the performance by improving generation of candidate disks. In one of the approaches called *Cluster Filtering Evaluation*, DBSCAN with parameters μ as a density threshold and ε for neighborhood radius is used to generate candidate disks. Once candidate disks are obtained, the basic algorithm for finding flocks is applied. This approach works particularly well when trajectory dataset is relatively small and many trajectories have similar moving patterns.

Important places

In the work of (Kang et al(2004)Kang, Welbourne, Stewart, and Borriello), the authors proposed an incremental clustering for identification of important places in a single trajectory. Several factors for the algorithm were defined: arbitrary number of clusters, exclusion of as much unimportant places as possible and being not computationally expensive to allow running on mobile devices. The algorithm is based on finding important places where many location measurements are clustered together. Two parameters controlled the cluster creation - distance between positions and time spent in a cluster. The basic idea is the following. Every new location measurement provided by a location-based device (Place Lab, in this case) is compared to the previous location. If the distance between previous location is less than a threshold, the new location is added to the previously created cluster. Otherwise, the new candidate cluster is created with the new location. The candidate cluster becomes a cluster of important places when the time difference between first point in a cluster and the last point is greater than the threshold. Similar ideas of finding interesting places in trajectories were used in later works (Alvares et al(2007)Alvares, Bogorny, Kuijpers, de Macedo, Moelans, and Vaisman, Zheng et al(2009)Zheng, Zhang, Xie, and Ma).

A similar task was performed in (Palma et al(2008)Palma, Bogorny, Kuijpers, and Alvares), this time by using speed characteristics. For this, the original definition of DBSCAN was altered to accommodate the temporal aspect. Specifically, the point p of a trajectory called *core* point if the time difference between first and last neighbor points of p was greater or equal to some predefined threshold $MinTime$ (minimum time). This definition corresponds to the maximum average speed condition $\varepsilon/MinTime$ in the neighborhood of point p. Since original DBSCAN requires two parameters to be provided for clustering: ε - radius of the neighborhood and $MinPts$ - minimum number of points in the neighborhood of p, similarly, the adopted version required providing two parameters: ε and $MinTime$. However, without knowing the characteristic of the trajectory it is difficult for the user to provide meaningful parameters. The authors proposed to regard the trajectory as a list of distances between two consecutive points and obtain means and standard deviations of these distances. Then, Gaussian curve can be plotted using these parameters that should give some information about the properties of the trajectory and inverse cumulative distribution function can be constructed expressed in terms of mean and standard deviation. In order to obtain ε, the user should provide a value between 0 and 1 that reflects the proportion of points that can be expected in a cluster.

Borderline cases: patterns

Patterns that are mined from trajectories are called *trajectory patterns* and characterize interesting behaviors of single object or group of moving objects (Fosca and Dino(2008)). Different approach exist in mining trajectory patterns. We present two examples. The first one is based on grid-based clustering and finding dense regions (Giannotti et al(2007)Giannotti, Nanni, Pinelli, and Pedreschi), the second is based on partitioning of trajectories and clustering of trajectories' segments (Kang and Yong(2009)).

(Giannotti et al(2007)Giannotti, Nanni, Pinelli, and Pedreschi) presented an algorithm to find frequent movement patterns that represent cumulative behavior of moving objects where a pattern, called *T-pattern*, was defined as a sequence of points with temporal transitions between consecutive points. A *T-pattern* is discovered if its spatial and temporal components approximately correspond to the input sequences (trajectories). The meaning of these patterns is that different objects visit the same places with similar time intervals. Once the patterns are discovered, the classical sequence mining algorithms can be applied to find frequent patterns. Crucial to the determination of *T-patterns* is the definition of the visiting regions. For this, the *Region-of-Interest (RoI)* notion was proposed. A *RoI* is defined as a place visited by many objects. Additionally, the duration of stay can be taken into account. The idea behind *Roi* is to divide the working region into cells and count the number of trajectories that intersect the cell. The algorithm for finding popular regions was proposed, which accepted the grid with cell densities and a density threshold δ as input. The algorithm scans the cells and tries to expand the region in four directions (left, right, up, down). The direction that maximizes the average cell density is selected and the cells are merged. After the regions of interest are obtained, the sequences can be created by following every trajectory and matching the regions of interest they intersect. The timestamps are assigned to the regions in two ways: (1) Using the time when the trajectory entered the region or (2) Using the starting time if the trajectory started in that region. Consequently, the sequences are used in mining frequent *T-patterns*. The proposed approach was evaluated on the trajectories of 273 trucks in Athens, Greece having $112,203$ points in total.

(Kang and Yong(2009)) argues that methods based on partition of the working space into grids may lose some patterns if the cell lengths are too large. In addition, some methods require trajectory discretization according to its recorded timestamps which can lead to creation of redundant and repeating sequences in which temporal aspects are contained in the sequentially ordered region ids. As a workaround to these issues, the authors proposed two refinements: (1) Partitioning trajectories into disjoint segments, which represent meaningful spatio-temporal changes of the movement of the object. The segment is defined as an area having start and end points as well as the time duration within the area. (2) Applying clustering algorithm to group similar segments. A *ST-pattern* (Spatio-temporal pattern) was defined as a sequences of segments (areas) with time duration described as a height of 3-dimensional cube. Thus, the sequences of *ST-patterns* are formed by clustering similar cubes. A four-step approach was proposed to mine frequent *ST-patterns*. In the first step, the trajectories are simplified using the DP (Douglas-Peucker) algorithm dividing the trajectories into segments. The segments are then normalized using linear transformation to allow comparison between segments having different offsets. In the next step, the spatio-temporal segments are clustered using the BIRCH (Zhang et al(1996)Zhang, Ramakrishnan, and Livny) algorithm. In the final step, a DFS-based (depth-first search) method is applied on the clustered regions to find frequent patterns.

44.3 Applications

The literature on spatio-temporal clustering is usually centered around concrete methods rather than application contexts. Nevertheless, in this chapter, we would like to bring examples of several possible scenarios where spatio-temporal clustering can be used along with other data mining methods.

For the sake of simplicity, we divide spatio-temporal data into three main categories according to the way these data are collected: *movement, cellular networks* and *environmental*. Movement data are often obtained by location based devices such as GPS and contain id of an object, its coordinates and timestamp. Cellular network data are obtained from mobile operators at the level of network bandwidth. Environmental data are usually obtained by censor networks and RFID technology.

The specificity of properties of these data require different approaches for analysis and also result in unique tasks. For example, in movement data the possible analysis tasks could be analysis of animal movement, their behavior in time, people's mobility and tracking of group objects. Phone calls that people make in a city can be used in the analysis of urban activity. Such information will be valuable for local authorities, service providers, decision makers, etc. Environmental processes are analyzed using information about locations and times of specific events. This information is of high importance to ecologists and geographers.

Table 44.1 summarizes the categories of spatio-temporal data, tasks considered in these categories, examples of applications, the basic methods used for solving tasks and the selected literature.

44.3.1 Movement data

Trajectory data obtained from location-aware devices usually comes as a sequence of points annotated by coordinate and time. However, not all of these points are equally important. Many application domains require identification of important parts from the trajectories. A single trajectory or a group of trajectories can be used for finding important parts. For example, in analysis of people's daily activities, some places like home or work could be identified as important while movement from one place to another would be considered as not important. Knowledge of such places can be used in analysis of activity of an object or group of objects (people, animals). Moreover, the information can be used in personalized applications (Kang et al(2004)Kang, Welbourne, Stewart, and Borriello). Usually, the place can be considered as important if the object spends in it considerable amount of time or the place is visited frequently by one or many objects. GPS-based devices are the main source of movement trajectories. However, the main disadvantage is that they loose signal indoor. A new approach to data collection using Place Lab (Schilit et al(2003)Schilit, LaMarca, Borriello, Griswold, McDonald, Lazowska, Balachandran, Hong, and Iverson) was proposed in which a WiFi enabled

Table 44.1. Overview of spatio-temporal methods and applications

Category	Problem	Application	Method based on	Selected literature
Movement	Trajectory clustering	Cars, evacuation traces, landings and interdictions of migrant boats	OPTICS	(Nanni and Pedreschi(2006)) (Rinzivillo et al(2008)Rinzivillo, Pedreschi, Nanni, Giannotti, Andrienko, and Andrienko) (Andrienko and Andrienko(2008))
	Trajectory aggregation			(Andrienko and Andrienko(2009))
	Trajectory generalization			(Andrienko et al(2009)Andrienko, Andrienko, Rinzivillo, Nanni, Pedreschi, and Giannotti)
	Moving clusters	Migrating animals, flocks,	DBSCAN	(Kalnis et al(2005)Kalnis, Mamoulis, and Bakiras)
		convoys of vehicles	Gridding, DBSCAN	(Jeung et al(2008)Jeung, Yiu, Zhou, Jensen, and Shen)
			DBSCAN	(Vieira et al(2009)Vieira, Bakalov, and Tsotras)
	Extracting important places from trajectories	People's trajectories	DBSCAN,	(Palma et al(2008)Palma, Bogorny, Kuijpers, and Alvares)
			Incremental clustering	(Kang et al(2004)Kang, Welbourne, Stewart, and Borriello)
	Trajectory patterns	Fleet of trucks	Density of spatial regions	(Giannotti et al(2007)Giannotti, Nanni, Pinelli, and Pedreschi)
		Synthetic data	BIRCH	(Kang and Yong(2009))
Cellular networks	Urban activity	Phone calls	k-means	(Reades et al(2007)Reades, Calabrese, Sevtsuk, and Ratti)
Environmental	Oceanography	Seawater distribution	DBSCAN	(Birant and Kut(2006), Birant and Kut(2007))
	Seismology	Seismic activity	Gridding, DBSCAN	(Wang et al(2006)Wang, Wang, and Li)

device can get location positions from various wireless access points installed in cities. This approach can be used by mobile devices in real time applications even when the person is inside a building. In the example presented by (Kang et al(2004)Kang, Welbourne, Stewart, and Borriello), the mobile device should identify the important place and act according to some scenario. For example it can switch to a silent mode when the person enters a public place. For this, incremental spatio-temporal clustering was used to identify important places.

Two fictitious but possible scenarios of analysis of movement were proposed at VAST 2008 mini challenge (Grinstein et al(2008)Grinstein, Plaisant, Laskowski, OConnell, Scholtz, and Whiting) and addressed in (Andrienko and Andrienko(2009)). In the first scenario called *Evacuation traces*, a bomb, set up by a religious group, exploded in the building. All employees and visitors in the building wore RFID badges that enabled recording location of every person. Five analytical questions were asked: *Where was the device set off, Identify potential suspects and/or witnesses to the event, Identify any suspects and/or witnesses who managed to escape the building, Identify any casualties, Describe the evacuation.* Clustering of trajectories comes in handy for answering the second and third questions. In order to find suspects of the event, the place of the explosion epicenter was identified and people's trajectories were separated into *normal* (trajectories not passing through the place of explosion) and *suspected.* In order to answer the third question, trajectories were clustered according to the common destination. This enabled to find people who managed to escape the building and those who didn't. The second scenario called *Migrant boats*, described a problem of illegal immigration of people by boats to the US. The data consisted, among the others, of the following fields: location and date where the migrant boat left the place and where the boat was intercepted or landed. The questions were to *characterize the choice of landing sites and their evolution over the years* and *characterize the geographical patterns of interdiction.* Spatio-temporal clustering with different distance functions was applied on the data and the following patterns were found: landings at the Mexican coast and period of migration started from 2006 and increased towards 2007, while the number of landings at the coast of Florida and nearby areas was significantly smaller during 2006-2007 than on 2005. It was shown that the strategy of migration changed over the years. The migration routes increased and included new destinations. Consequently, the patrolling extended over larger areas and the rate of successful landings increased.

44.3.2 Cellular networks

Until recently, surveys were the only data collection method for analysis of various urban activities. With the rapid development of mobile networks and their global coverage, new opportunities for analysis of urban systems using phone call data have emerged. (Reades et al(2007)Reades, Calabrese, Sevtsuk, and Ratti) were one of the first who attempted to analyze urban dynamics on a city level using *Erlang* data. Erlang data is a measure of network bandwidth and indicate the load of cellular antenna as an average number of calls made over specific time period (usually hour). As such, these data are considered spatio-temporal, where the spatial component relates to the location of a transmitting antenna and temporal aspect is an aggregation of phone calls by time interval. Since the data do not contain object identifiers, only group activity can be learned from it.

The city of Rome was divided into cells of $1,600m^2$ each, $262,144$ cells in total. The Erlang value was computed for every cell taking into consideration the signal decay and positions of antennas. For each cell, the average Erlang value was obtained using 15 minutes interval during 90 day period. Thus, every cell contained seven (for every day of the week) observations of phone call activities during 90 days and 96 measurements for each day (using 15 minutes interval). Initially, six cells corresponding to different parts of the city and types of activities (residential areas, touristic places, nighttime spots) with significantly different Erlang values were selected. The analysis of these places revealed six patterns in the daily activity when there were rapid changes in cellular network usage: $1a.m.$,$7a.m.$,$11a.m.$,$2p.m.$,$5p.m.$, and $9p.m.$. To check this hypothesis, k-means was applied on all $262,144$ cells using 24-dimensional feature vector of six daily periods averaged for Monday through Thursday and separate six daily periods for Friday, Saturday and Sunday. The result of clustering suggested that the phone call activity is divided into eight separate clusters. The visual interpretation of these clusters revealed the correspondence of places to expected types of people's activity over time.

44.3.3 Environmental data

Very early examples of spatio-temporal analysis of environmental data, including clustering, are given in (Stolorz et al(1995)Stolorz, Nakamura, Mesrobian, Muntz, Santos, Yi, and Ng) as applications of an exploratory data analysis environment called CONQUEST. The system is specifically devoted to deal with sequences of remotely-sensed images that describe the evolution of some geophysical measures in some spatial areas. A most relevant application example is cyclones detection, i.e., extracting locations of cyclones and the tracks (trajectories) they follow. Since cyclones are events rather than physical objects, and there is not a straightforward way to locate them, cyclone detection requires a multi-step analysis process, where spatio-temporal data is subject to transformations from a data type to another one. First, for each time instant all candidate cyclone occurrences are located by means of a local minima heuristics based on sea level pressure, i.e. spatial locations where the sea level pressure is lower than their neighbourhood (namely, a circle of given radius) are selected. The result is essentially a set of spatio-temporal events, so far considered as independent from each other. Then, the second step consists in spatio-temporally clustering such cyclone occurrences, by iteratively merging occurrences that are temporally close and have a small spatial distance. The latter condition is relaxed when the instantaneous wind direction and magnitude are coherent with the relative positions of the occurrences – i.e., the cyclone can move fast, if wind conditions allow that. The output of this second phase is a set of trajectories, each describing the movement in time of a cyclone, which can be visually inspected and compared against geographical and geophysical features of the territory. In summary, this application shows an interesting analysis process where original geo-referenced time series are collectively analyzed to locate complex spatio-temporal events, and such events are later connected – i.e., associated to the same entity – to form trajectories.

More recently (Birant and Kut(2006), Birant and Kut(2007)) studied spatio-temporal marine data with the following attributes: sea surface temperature, the sea surface height residual, the significant wave height and wind speed values of four seas (the Black Sea, the Marmara Sea, the Aegean Sea, and the eastern part of the Mediterranean). The authors proposed ST-DBSCAN algorithm as an extension of classical DBSCAN to find seawater regions that have similar physical characteristics. In particular, the authors pursued three goals: (1) to discover regions with a similar sea surface temperature (2) to discover regions with similar sea surface height residual values and (3) to find regions with significant wave height. The database that was used for analysis contained measurements of sea surface temperature from 5340 stations obtained between 2001 and 2004, sea surface height collected over five-day periods between 1992 and 2002 and significant wave height collected over ten-day periods between 1992 and 2002 from 1707 stations. The ST-DBSCAN algorithm was integrated into the interactive system to facilitate the analysis.

44.4 Open Issues

Spatio-temporal properties of the data introduce additional complexity to the data mining process and to the clustering in particular. We can differentiate between two types of issues that the analyst should deal with or take into consideration during analysis: general and application dependent. The general issues involve such aspects as data quality, precision and uncertainty (Miller and Han(2009)). Scalability, spatial resolution and time granularity can be related to application dependent issues.

Data quality (spatial and temporal) and precision depends on the way the data is generated. Movement data is usually collected using GPS-enabled devices attached to an object. For example, when a person enters a building a GPS signal can be lost or the positioning may be inaccurate due to a weak connection to satellites. As in the general data preprocessing step, the analyst should decide how to handle missing or inaccurate parts of the data - should it be ignored, tolerated or interpolated.

The computational power does not go in line with the pace at which large amounts of data are being generated and stored. Thus, the scalability becomes a significant issue for the analysis and demand new algorithmic solutions or approaches to handle the data.

Spatial resolution and time granularity can be regarded as most crucial in spatio-temporal clustering since change in the size of the area over which the attribute is distributed or change in time interval can lead to discovery of completely different clusters and therefore, can lead to the improper explanation of the phenomena under investigation. There are still no general guidelines for proper selection of spatial and temporal resolution and it is rather unlikely that such guidelines will be proposed. Instead, ad hoc approaches are proposed to handle the problem in specific domains (see for example (Nanni and Pedreschi(2006))). Due to this, the involvement of the domain expert in every step of spatio-temporal clustering becomes essential. The geospatial visual analytics field has recently emerged as the discipline that combines automatic data mining approaches including spatio-temporal clustering with visual reasoning supported by the knowledge of domain experts and has been successfully applied at different geographical spatio-temporal phenomena ((Andrienko and Andrienko(2006), Andrienko et al(2007)Andrienko, Andrienko, and Wrobel, Andrienko and Andrienko(2010))).

A class of application-dependent issues that is quickly emerging in the spatio-temporal clustering field is related to exploitation of available background knowledge. Indeed, most of the methods and solutions surveyed in this chapter work on an abstract space where locations have no specific meanings and the analysis process extracts information from scratch, instead of starting from (and integrating to) possible a priori knowledge of the phenomena under consideration. On the opposite, a priori knowledge about such phenomena and about the context they take place in is commonly available in real applications, and integrating them in the mining process might improve the output quality (Alvares et al(2007)Alvares, Bogorny, Kuijpers, de Macedo, Moelans, and Vaisman, Baglioni et al(2009)Baglioni, Antonio Fernandes de Macedo, Renso, Trasarti, and Wachowicz, Kisilevich et al(2010)Kisilevich, Keim, and Rokach). Examples of that include the very basic knowledge of the street network and land usage, that can help in understanding which aspects of the behavior of our objects (e.g., which parts of the trajectory of a moving object) are most discriminant and better suited to form homogeneous clusters; or the existence of recurring events, such as rush hours and planned road maintenance in a urban mobility setting, that are known to interfere with our phenomena in predictable ways.

Recently, the spatio-temporal data mining literature has also pointed out that the relevant context for the analysis mobile objects includes not only geographic features and other physical constraints, but also the population of objects themselves, since in most application scenarios objects can interact and mutually interfere with each other's activity. Classical examples include traffic jams – an entity that emerges from the interaction of vehicles and, in turn, dominates their behavior. Considering interactions in the clustering process is expected to improve the reliability of clusters, yet a systematic taxonomy of relevant interaction types is still not available (neither a general one, nor any application-specific one), it is still not known how to detect such interactions automatically, and understanding the most suitable way to integrate them in a clustering process is still an open problem.

44.5 Conclusions

In this chapter we focused on geographical spatio-temporal clustering. We presented a classification of main spatio-temporal types of data: *ST events, Geo-referenced variables, Moving objects* and *Trajectories*. We described in detail how spatio-temporal clustering is applied on trajectories, provided an overview of recent research developments and presented possible scenarios in several application domains such as movement, cellular networks and environmental studies.

References

Agrawal R, Faloutsos C, Swami AN (1993) Efficient Similarity Search In Sequence Databases. In: Lomet D (ed) Proceedings of the 4th International Conference of Foundations of Data Organization and Algorithms (FODO), Springer Verlag, Chicago, Illinois, pp 69–84

Alon J, Sclaroff S, Kollios G, Pavlovic V (2003) Discovering clusters in motion time-series data. In: CVPR (1), pp 375–381

Alvares LO, Bogorny V, Kuijpers B, de Macedo JAF, Moelans B, Vaisman A (2007) A model for enriching trajectories with semantic geographical information. In: GIS '07: Proceedings of the 15th annual ACM international symposium on Advances in geographic information systems, pp 1–8

Andrienko G, Andrienko N (2008) Spatio-temporal aggregation for visual analysis of movements. In: Proceedings of IEEE Symposium on Visual Analytics Science and Technology (VAST 2008), IEEE Computer Society Press, pp 51–58

Andrienko G, Andrienko N (2009) Interactive cluster analysis of diverse types of spatiotemporal data. ACM SIGKDD Explorations

Andrienko G, Andrienko N (2010) Spatial generalization and aggregation of massive movement data. IEEE Transactions on Visualization and Computer Graphics (TVCG) Accepted

Andrienko G, Andrienko N, Wrobel S (2007) Visual analytics tools for analysis of movement data. SIGKDD Explorations Newsletter 9(2):38–46

Andrienko G, Andrienko N, Rinzivillo S, Nanni M, Pedreschi D, Giannotti F (2009) Interactive Visual Clustering of Large Collections of Trajectories. VAST 2009

Andrienko N, Andrienko G (2006) Exploratory analysis of spatial and temporal data: a systematic approach. Springer Verlag

Ankerst M, Breunig MM, Kriegel HP, Sander J (1999) Optics: ordering points to identify the clustering structure. SIGMOD Rec 28(2):49–60

Baglioni M, Antonio Fernandes de Macedo J, Renso C, Trasarti R, Wachowicz M (2009) Towards semantic interpretation of movement behavior. Advances in GIScience pp 271–288

Berndt DJ, Clifford J (1996) Finding patterns in time series: a dynamic programming approach. Advances in knowledge discovery and data mining pp 229–248

Birant D, Kut A (2006) An algorithm to discover spatialtemporal distributions of physical seawater characteristics and a case study in turkish seas. Journal of Marine Science and Technology pp 183–192

Birant D, Kut A (2007) St-dbscan: An algorithm for clustering spatial-temporal data. Data Knowl Eng 60(1):208–221

Chan KP, chee Fu AW (1999) Efficient time series matching by wavelets. In: In ICDE, pp 126–133

Chen L, Özsu MT, Oria V (2005) Robust and fast similarity search for moving object trajectories. In: SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data, ACM, New York, NY, USA, pp 491–502

Chudova D, Gaffney S, Mjolsness E, Smyth P (2003) Translation-invariant mixture models for curve clustering. In: KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, New York, NY, USA, pp 79–88

Ciaccia P, Patella M, Zezula P (1997) M-tree: An efficient access method for similarity search in metric spaces. In: Jarke M, Carey M, Dittrich KR, Lochovsky F, Loucopoulos P, Jeusfeld MA (eds) Proceedings of the 23rd International Conference on Very Large Data Bases (VLDB'97), Morgan Kaufmann Publishers, Inc., Athens, Greece, pp 426–435

Cohen S., Rokach L., Maimon O., Decision Tree Instance Space Decomposition with Grouped Gain-Ratio, Information Science, Volume 177, Issue 17, pp. 3592-3612, 2007.

Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. Data Mining and Knowledge Discovery pp 226–231

Fosca G, Dino P (2008) Mobility, Data Mining and Privacy: Geographic Knowledge Discovery. Springer

Frentzos E, Gratsias K, Theodoridis Y (2007) Index-based most similar trajectory search. In: ICDE, pp 816–825

Gaffney S, Smyth P (1999) Trajectory clustering with mixtures of regression models. In: KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, New York, NY, USA, pp 63–72

Giannotti F, Nanni M, Pinelli F, Pedreschi D (2007) Trajectory pattern mining. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, p 339

Grinstein G, Plaisant C, Laskowski S, OConnell T, Scholtz J, Whiting M (2008) VAST 2008 Challenge: Introducing mini-challenges. In: Proceedings of IEEE Symposium, vol 1, pp 195–196

Gudmundsson J, van Kreveld M (2006) Computing longest duration flocks in trajectory data. In: GIS '06: Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems, ACM, New York, NY, USA, pp 35–42

Hwang SY, Liu YH, Chiu JK, Lim EP (2005) Mining mobile group patterns: A trajectory-based approach. In: PAKDD, pp 713–718

Iyengar VS (2004) On detecting space-time clusters. In: Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining (KDD'04), ACM, pp 587–592

Jeung H, Yiu ML, Zhou X, Jensen CS, Shen HT (2008) Discovery of convoys in trajectory databases. Proc VLDB Endow 1(1):1068–1080

Kalnis P, Mamoulis N, Bakiras S (2005) On discovering moving clusters in spatio-temporal data. Advances in Spatial and Temporal Databases pp 364–381

Kang J, Yong HS (2009) Mining Trajectory Patterns by Incorporating Temporal Properties. Proceedings of the 1st International Conference on Emerging Databases

Kang JH, Welbourne W, Stewart B, Borriello G (2004) Extracting places from traces of locations. In: WMASH '04: Proceedings of the 2nd ACM international workshop on Wireless mobile applications and services on WLAN hotspots, ACM, New York, NY, USA, pp 110–118

Kisilevich S, Keim D, Rokach L (2010) A novel approach to mining travel sequences using collections of geo-tagged photos. In: The 13th AGILE International Conference on Geographic Information Science

Kulldorff M (1997) A spatial scan statistic. Communications in Statistics: Theory and Methods 26(6):1481–1496

Lee JG, Han J, Whang KY (2007) Trajectory clustering: a partition-and-group framework. In: SIGMOD Conference, pp 593–604

Li Y, Han J, Yang J (2004a) Clustering moving objects. In: Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining (KDD'04), ACM, pp 617–622

Li Y, Han J, Yang J (2004b) Clustering moving objects. In: KDD, pp 617–622

Maimon O., and Rokach, L. Data Mining by Attribute Decomposition with semiconductors manufacturing case study, in Data Mining for Design and Manufacturing: Methods and Applications, D. Braha (ed.), Kluwer Academic Publishers, pp. 311–336, 2001.

Miller HJ, Han J (2009) Geographic data mining and knowledge discovery. Chapman & Hall/CRC

Nanni M, Pedreschi D (2006) Time-focused clustering of trajectories of moving objects. Journal of Intelligent Information Systems 27(3):267–289

Palma AT, Bogorny V, Kuijpers B, Alvares LO (2008) A clustering-based approach for discovering interesting places in trajectories. In: SAC '08: Proceedings of the 2008 ACM symposium on Applied computing, pp 863–868

Pelekis N, Kopanakis I, Marketos G, Ntoutsi I, Andrienko G, Theodoridis Y (2007) Similarity search in trajectory databases. In: TIME '07: Proceedings of the 14th International Symposium on Temporal Representation and Reasoning, IEEE Computer Society, Washington, DC, USA, pp 129–140

Reades J, Calabrese F, Sevtsuk A, Ratti C (2007) Cellular census: Explorations in urban data collection. IEEE Pervasive Computing 6(3):30–38

Rinzivillo S, Pedreschi D, Nanni M, Giannotti F, Andrienko N, Andrienko G (2008) Visually driven analysis of movement data by progressive clustering. Information Visualization 7(3):225–239

Rokach L. and Maimon O., Feature Set Decomposition for Decision Trees, Journal of Intelligent Data Analysis, Volume 9, Number 2, 2005b, pp 131–158.

Rokach L., Genetic algorithm-based feature set partitioning for classification problems,Pattern Recognition, 41(5):1676–1700, 2008.

Rokach L., Maimon O. and Lavi I., Space Decomposition In Data Mining: A Clustering Approach, Proceedings of the 14th International Symposium On Methodologies For Intelligent Systems, Maebashi, Japan, Lecture Notes in Computer Science, Springer-Verlag, 2003, pp. 24–31.

Schilit BN, LaMarca A, Borriello G, Griswold WG, McDonald D, Lazowska E, Balachandran A, Hong J, Iverson V (2003) Challenge: ubiquitous location-aware computing and the "place lab" initiative. In: WMASH '03: Proceedings of the 1st ACM international workshop on Wireless mobile applications and services on WLAN hotspots, ACM, New York, NY, USA, pp 29–35

Stolorz P, Nakamura H, Mesrobian E, Muntz RR, Santos JR, Yi J, Ng K (1995) Fast spatio-temporal data mining of large geophysical datasets. In: Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD'95), AAAI Press, pp 300–305

Theodoridis Y (2003) Ten benchmark database queries for location-based services. The Computer Journal 46(6):713–725

Vieira MR, Bakalov P, Tsotras VJ (2009) On-line discovery of flock patterns in spatio-temporal data. In: GIS '09: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, New York, NY, USA, pp 286–295

Vlachos M, Kollios G, Gunopulos D (2002) Discovering similar multidimensional trajectories. In: Proceedings of the International Conference on Data Engineering, pp 673–684

Vlachos M, Hadjieleftheriou M, Gunopulos D, Keogh E (2003) Indexing multi-dimensional time-series with support for multiple distance measures. In: KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, New York, NY, USA, pp 216–225

Wang M, Wang A, Li A (2006) Mining Spatial-temporal Clusters from Geo-databases. Lecture Notes in Computer Science 4093:263

Zhang P, Huang Y, Shekhar S, Kumar V (2003) Correlation analysis of spatial time series datasets: A filter-and-refine approach. In: In the Proc. of the 7th PAKDD

Zhang T, Ramakrishnan R, Livny M (1996) BIRCH: an efficient data clustering method for very large databases. ACM SIGMOD Record 25(2):103–114

Zheng Y, Zhang L, Xie X, Ma WY (2009) Mining interesting locations and travel sequences from gps trajectories. In: WWW '09: Proceedings of the 18th international conference on World wide web, pp 791–800