A Survey on Clustering Techniques for Situation Awareness*

Stefan Mitsch[2], Andreas Müller[1], Werner Retschitzegger[1], Andrea Salfinger[1], and Wieland Schwinger[1]

[1] Johannes Kepler University Linz, Altenbergerstr. 69, 4040 Linz, Austria
[2] Computer Science Dept., Carnegie Mellon University, Pittsburgh, PA-15213, USA

Abstract. Situation awareness (SAW) systems aim at supporting assessment of critical situations as, e.g., needed in traffic control centers, in order to reduce the massive information overload. When assessing situations in such control centers, SAW systems have to cope with a large number of heterogeneous but interrelated real-world objects stemming from various sources, which evolve over time and space. These specific requirements harden the selection of adequate data mining techniques, such as clustering, complementing situation assessment through a data-driven approach by facilitating configuration of the critical situations to be monitored. Thus, this paper aims at presenting a survey on clustering approaches suitable for SAW systems. As a prerequisite for a systematic comparison, criteria are derived reflecting the specific requirements of SAW systems and clustering techniques. These criteria are employed in order to evaluate a carefully selected set of clustering approaches, summarizing the approaches' strengths and shortcomings.

1 Introduction

Situation awareness (SAW). SAW systems are increasingly used in large control centers for air or road traffic management in order to reduce the information overload of operators induced by various data sources. This is done by automatically assessing critical situations occurring in the environment under control (e. g., an accident causing a traffic jam) [3].

Data Mining (DM) for SAW. The definition of relevant critical situations is provided in current SAW systems (e.g., [3,26]) explicitly by domain experts during a *configuration phase*, representing a time-consuming task. This effort of providing explicit knowledge could be complemented and eased by a data-driven approach in terms of DM techniques making direct use of the observed data, through detecting "interesting" or uncommon relationships, which the domain experts might not be explicitly aware of (i.e., intrinsic knowledge). Furthermore, during *runtime* of the SAW system ongoing changes and additional uncommon

* This work has been funded by the Austrian Federal Ministry of Transport, Innovation and Technology (BMVIT) under grant FFG FIT-IT 829589, FFG BRIDGE 838526 and FFG Basisprogramm 838181.

relationships can be spotted that have not even been explicitly defined so far. Especially *clustering techniques* are considered to be beneficial, since they require neither a-priori user-created test data sets nor other background knowledge and also allow anomaly detection, such as uncommon relations. If we consider a road traffic SAW system, clustering might for example be used to detect (ST) hotspots on a highway, i. e., road segments and time windows where atypically many accidents occur, or to reveal current major traffic flows in an urban area.
Specific Requirements of SAW. The nature of SAW systems, however, poses specific requirements on the applicability of existing clustering techniques. SAW systems have to cope with a large number of heterogeneous but interrelated real-world objects stemming from various sources, which evolve over time and space, being quantitative or qualitative in nature (as, e.g., demonstrated in [3]). The focus of this paper is therefore on evaluating existing spatio-temporal (ST) clustering techniques with respect to their ability to complement the specification of critical situations in SAW systems.
Contributions. In this paper, we first systematically examine the requirements on clustering techniques to be applicable in the SAW domain. Then, we survey several carefully selected approaches according to our criteria stemming from the fields of SAW and ST clustering, and compare them in our lessons learned, highlighting their advantages and shortcomings.
Structure of the Paper. In the following section, we compare our survey with related work (cf. Section 2). Then, we introduce our evaluation criteria (cf. Section 3), before we present lessons learned (cf. Section 4). Due to space limitations, the in-depth criteria-driven evaluation of each approach surveyed can be found online[3] only.

2 Related Work

Despite a plethora of work exists in the field of data mining, to the best of our knowledge no other survey has dealt with the topic of clustering ST data for SAW applications so far. However, there exist surveys about ST clustering (e. g., [21]) and ST data mining in general (e. g., [19], [29]), surveys about spatial (e. g., [11]) and temporal (e. g., [34]) clustering and various surveys on clustering in other domains (e. g., [15], [36]). Furthermore the field of stream data mining has received a lot of attention over the last few years and as a result several surveys on stream data clustering (e. g., [20]) and on stream data mining (e. g., [8], [13]) in general have been conducted.

In the following paragraphs these mentioned surveys are briefly discussed with respect to the contributions of our survey.
ST Clustering. Kisilevich et al. [21] propose a classification of ST data and focus their survey of clustering techniques on *trajectories*, being the most complex setting in their classification. The main part of the survey is a discussion of several groups of clustering approaches including different example algorithms for

[3] `http://csi.situation-awareness.net/stc-survey`

each group. Finally, they present examples from different application domains where clustering of trajectory data is an issue, like studying movement behavior, cellular networks or environmental studies. In contrast to our survey, Kisilevich et al. focus on presenting various groups of ST data mining approaches, rather than systematically analyzing different techniques backed by a catalog of evaluation criteria. Furthermore, we considered the applicability of the techniques to our domain on basis of SAW-specific criteria, while they conducted their analysis in a more general way. Nevertheless, their work represented a valuable starting point for our survey, from which we especially adopted their classification of ST data, like trajectories or ST events.

Spatial, Temporal and General Clustering. Several surveys on spatial clustering (e. g., [11]) and temporal clustering (e. g., [34]) have been conducted for different application domains. Nevertheless, none of the surveyed approaches specifically deals with the unique characteristics that arise from the domain of SAW respectively from ST data in general (cf. Section 3.1), but take only either spatial- or temporal characteristics into account.

Finally, there exist numerous and comprehensive surveys on non-ST clustering (e. g., [15], [36]). These approaches cannot simply be used to work with ST data, but have to be adopted manually to deal with its particular nature.

ST Data Mining. Kalyani et al. [19] describe the peculiarities of ST data models and the resulting increase of complexity for the data mining algorithms. In their survey they outline different data mining tasks compared to their spatial counterparts and motivate the need for dedicated ST data mining techniques. Geetha et al. [29] present a short overview of challenges in spatial, temporal and ST data mining. However, none of the above focuses on concrete data mining techniques, but rather gives an overview of the field of ST data mining.

Stream Data Mining. In their survey on clustering of time series data streams, Kavitha et al. [20] review the concepts of time series and provide an overview of available clustering algorithms for streaming data. More general surveys of stream data mining were conducted by Gaber et al. [8] and Ikonomovska et al. [13]. Both review the theoretical foundations of stream data analysis and give a rough overview of algorithms for the various stream data mining tasks and applications. Furthermore, they mention that stream data clustering is a major task in stream data mining and lots of algorithms have been adapted to work on streaming data (e. g., CluStream [1], DenStream [7], ClusTree [22]).

However, in contrast to our SAW systems which store a history of the data received, thus allowing an arbitrary number of read accesses, such stream data mining approaches do not store the huge amount of data they process. Even though the elements of a data stream are temporally ordered, they might arrive in a time-varying and unpredictable fashion and do not necessarily contain a timestamp, which does not allow for the identification of temporal patterns (especially cyclic ones). Furthermore, most stream data mining approaches surveyed by these authors do not deal with spatial data at all and hence are not applicable here.

Consequentially, we conduct this survey on ST clustering to investigate the applicability of various clustering techniques for the field of SAW.

3 Evaluation Criteria

In this section, we derive a systematic criteria catalog, (methodologically adhering to some of our previous surveys, e.g., [35]), which we use for evaluating selected ST clustering techniques. The criteria catalog consists of two sets of criteria, as depicted in Fig. 1, comprising SAW-specific criteria (cf. Section 3.1) and clustering-specific criteria (cf. Section 3.2), which are detailed in the following. Each criterion is assigned an abbreviation for reference during evaluation.

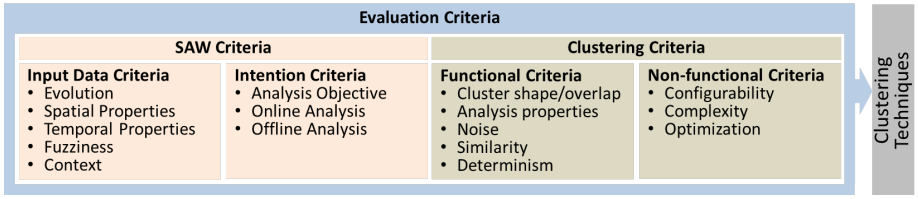

Fig. 1. The evaluation criteria at a glance

3.1 SAW-specific Criteria

The rationale behind the following SAW-specific criteria, which are illustrated by examples from the domain of road traffic control, is based on considering the kind of input data a clustering technique has to cope with ("What do we have") and further on reflecting on the intent of the analysis ("What do we want").

Spatial, Temporal and Other Input Data Properties. As already mentioned, SAW systems employed in control centers need to monitor a large number of interrelated objects anchored in space and time, both either with an extent or without. In particular the *temporal properties* (`TP`) range from instants to intervals, whereas the *spatial properties* (`SP`) comprise points, one-dimensional intervals (e. g., lines) or two-dimensional intervals (e. g., regions). Non-ST properties can be of qualitative (e.g., freezing temperature) or quantitative (e.g., 0 °C) nature. Regarding `SP` and `TP`, however, we assume quantitative data, as typically required by SAW applications to monitor the ST environment.

Heterogeneity of Input Data. Another SAW-specific requirement is that objects constituting a certain critical situation often exhibit a mixture of different ST properties. This can be exemplified by a situation involving three objects, (i) a bus, sending location points at time instants via GPS to the control center (i. e., a trajectory), (ii) a traffic jam, comprising a spatial and temporal interval, both evolving over time (i. e., growing, shrinking, moving) and (iii) a fog area

being spatially extended over a certain region and characterized by a temporal interval (e. g., predicted for two hours). The more different ST properties data a clustering technique supports, i.e., allows for *heterogeneous input data* (`HID`), the better it is applicable to the SAW domain.

Evolution of Input Data. As indicated in the examples above, the temporal dimension furthermore captures the potential *evolution* of the observed objects (`E`) with respect to their spatial (e.g., location or length) or non-spatial (e.g., temperature) properties. An evolution along the spatial dimension corresponds to a *mobile*, i. e., moving, object, whereas an object solely evolving with respect to the non-spatial properties represents an *immobile*, i.e., static, object.

Context of Input Data. The input data used in SAW systems often comprises objects which are bound to a certain *context* (`CID`), enforcing constraints on the interpretation of the input data (e. g., in road traffic, the majority of objects cannot move around in space freely, but is bound to the underlying road network), or further requirements on the input data (e. g., necessity of velocity information). We evaluate if and what kind of context information is required.

Fuzzy Input Data. SAW systems often have to cope with *fuzzy input data* (`FID`), for example incidents reported by humans who only have a partial overview of the situation. Fuzzy input data might address the spatial properties (`SP`) (e.g., uncertainty about the exact location of an object), the temporal properties (`TP`) (e.g., an accident has occurred within the last half an hour), or the non-ST properties (e.g., it cannot be defined exactly what has happened).

Besides the criteria mentioned above, which detail the nature of the input data, the following further deal with the goal of the analysis.

Intention. This criterion (`I`) reflects the *objective of the analysis*, i.e., the kind of implicit knowledge that should be extracted by the clustering technique. Possible intentions are clustering of *events* or *regions* with similar ST characteristics, clustering of *trajectories*, or the detection of *moving clusters*.

Online or Offline Analysis. The requirements imposed on the clustering techniques differ with respect to the phase they should be employed. During the configuration phase of a SAW system, we have to perform an analysis on a complete, historical data set, i. e., *offline analysis*. However, since an SAW system typically operates on a real-time environment, we also want to perform an analysis at runtime, i. e., *online analysis*. Since clustering techniques devoted to online analysis often rely on optimizations and approximations in order to deliver fast results (e.g., compute only locally optimal clusters), these approaches are less suited for configuration tasks where computation time is not a major issue, whereas an exact result is preferred. Thus, this criterion (`OOA`) reflects whether the clustering technique is only suited for the offline configuration phase, or if it is also applicable to or favored for runtime analysis.

3.2 Criteria Imposed by Clustering Techniques

Whereas in the previous section, the criteria have been derived from the nature of data and the intention of the analysis stemming from the SAW domain,

the present section focuses on assessing *how* these goals can be achieved with dedicated ST clustering techniques.

We first give a short description of the main contribution (`MC`) and distinguish the clustering techniques according to their *algorithm class* (`AC`), describing the method how the clusters are obtained. Following [12] this can be, due to *partitioning* (i.e., the data space is partitioned as a whole into several clusters), *hierarchical clustering* (i. e., clusters are created by merging close data points bottom-up or splitting clusters top-down), *density-based clustering* (i. e., clusters are defined as exceeding a certain density of data points over a defined region), and *grid-based clustering* (i. e., the data space is overlaid by a grid, grid cells containing similar structures are merged).

The remaining criteria are structured into functional and non-functional ones.
Functional Criteria. The distinct algorithmic methods yield different *cluster shapes* (`CS`), distinguishing *spherical*, *rectangular* or *arbitrarily* shaped clusters. Besides, it is investigated if a technique can handle *clusters which overlap* (`CO`). For *analysis properties*, we consider *spatial analysis properties* (`SA`), i. e., what kind of spatial data is internally processed by the algorithm, *temporal analysis properties* (`TA`), i. e., time instants or intervals, *temporal patterns* (`TAP`), i. e., linear or cyclic time patterns, and the *focus* of the analysis (`F`), i. e., if spatial and temporal aspects are handled equally or if one is favored without ignoring the other. We evaluate if the techniques support dedicated handling of *noise* (`N`) within the data set and furthermore investigate the employed *similarity measures* (`SM`) and the *determinism* of the produced results (`D`).
Non-functional Criteria. These criteria comprise the *configurability* (`C`) of the approaches, reflecting to which degree the clustering technique can be tweaked, and the computational *complexity* (`CC`). Furthermore, we examine whether and which *optimization strategies* (`O`) are provided, which would be beneficial in reducing runtime, however might also affect the quality of the obtained clustering result.

4 Survey of ST Clustering Techniques

In the following we present our selection of techniques, and group and analyze them according to our criteria.

4.1 Rationale Behind the Selection of Techniques

We carefully selected ST clustering approaches ranging from very recent ones on the one hand, to several more mature ones on the other hand to provide a broad overview of the field of ST clustering. We did not include *visually aided approaches* (e. g., Andrienko et. al [2]) in our survey as they contradict our approach to complement the configuration of SAW systems in a more data-driven way because a purely user-guided clustering approach again has to be steered by a domain expert.

In the following, we structure our discussion into three groups according to the *kind of evolution* the techniques consider. The rationale behind these categories is that algorithms from the different groups share several properties, which is reflected in the evaluation criteria tables.

The first group comprises techniques that *do not entail any evolution* (`NE`), like clustering ST events (i. e., grouping events in close or similar regions) or ST groups (i. e., finding regions sharing similar physical properties over time). Techniques for trajectory clustering (i. e., grouping similar trajectories) deal with the *evolution of objects* that move along these trajectories (`OE`). And finally, the area of moving-object clustering (i. e., discovery of groups of objects moving together during the same time period, so called moving clusters) and the detection of spatio-temporal trends (e.g., disease outbreaks) deal with the *evolution of clusters* over time (`CE`).

		Input Data									Intention (I)					
		Evolution (E)		Properties (HID)					Fuzziness (FID)	Context (CID)	No Evolution				Analysis (OOA)	
		Spatial		Spatial (SP)			Temporal (TP)									
		Immobile	Mobile	Point	Line	Region	Instant	Interval			Cluster ST-Events	Cluster ST-Regions	Cluster Trajectories	Moving Clusters	Online	Offline
No Evo.	Wang, 2006	✓	-	✓	-	-	✓	-	-	-	✓	-	-	-	-	✓
	Birant, 2006	✓	-	✓	-	-	✓	-	-	-	-	✓	-	-	-	✓
	Tai, 2007	✓	(✓)	✓	-	-	✓	-	-	-	✓	-	-	-	✓	✓
Object Evo.	Gaffney, 1999	-	✓	✓	-	-	✓	-	-	-	-	-	✓	-	-	✓
	Nanni, 2006	-	✓	✓	-	-	✓	-	-	-	-	-	✓	-	-	✓
	Li, 2010	-	✓	✓	-	-	✓	-	-	-	-	-	✓	-	✓	✓
	Lu, 2011	-	✓	✓	-	-	✓	-	-	-	-	-	✓	-	-	✓
	Gariel, 2011	-	✓	✓	-	-	✓	-	-	-	-	-	✓	-	✓	✓
Cluster Evolution	Kalnis, 2005	-	✓	✓	-	-	✓	-	-	-	-	-	-	✓	✓	✓
	Iyengar, 2004	✓	-	✓	-	-	✓	-	-	-	✓	-	-	-	-	✓
	Neill, 2005	✓	-	✓	-	-	✓	-	-	-	✓	-	-	-	-	✓
	Li, 2004	-	✓	✓	-	-	✓	-	-	requires velocity	-	-	-	✓	-	✓
	Jensen, 2007	-	✓	✓	-	-	✓	-	✓	requires velocity	-	-	-	✓	✓	-
	Chen, 2007	-	✓	✓	-	-	✓	-	-	spatial network & velocity	-	-	-	✓	✓	-
	Rosswog, 2008	-	✓	✓	-	-	✓	-	-	-	-	-	-	✓	-	✓
	Jeung, 2008	-	✓	✓	-	-	✓	-	-		-	-	✓	✓	-	✓
	Rosswog, 2012	-	✓	✓	-	-	✓	-	-	-	-	-	-	✓	-	✓
	Zheng, 2013	-	✓	✓	-	-	✓	-	-	-	-	-	✓	✓	(✓)	✓

Fig. 2. Situation Awareness Criteria Table

4.2 Lessons Learned

Our evaluation of ST clustering approaches for the field of SAW has revealed interesting peculiarities of current clustering techniques. In the following, we explain our findings grouped according to our evaluation criteria (cf. also Figure 2, 3 and 4). Note that a tick in parentheses means that the criteria is only partly fulfilled by the approach.

		General Information					Non-functional Criteria		
			Algorithm Class (AC)						
		Main Contribution (MC)	Partitioning	Hierarchical	Density-based	Grid-based	Configurability (C)	Computational Complexity (CC)	Optimization strategies (O)
No Evo.	**Wang, 2006**	DBSCAN-based / grid based ST-clustering	-	-	✓	✓	•calculated with k-dist graph	O(n) / O(n^2)	fast vs. precise
	Birant, 2006	DBSCAN-based ST-clustering	-	-	✓	-	•min # points •density •similarities	O(n*log n)	improved R-Tree, filters to reduce search space
	Tai, 2007	incremental ST-clustering of dual data	✓	-	-	-	•mostly determined automatically	not specified	NeiGraph; fast vs. precise
Object Evo.	**Gaffney, 1999**	clustering trajectories through regression	✓	-	-	-	•# clusters	O(N K I) (K…# clust., I…# iter.)	-
	Nanni, 2006	clustering trajectories with focus on time	-	-	✓	-	•min # points •time window •similarities	O(n*log n) / O(n_iter*n*log n)	M-Tree
	Li, 2010	incremental clustering trajectories with micro clusters	-	-	✓	-	•distance threshold •parameters for macro-clustering	not specified	micro-clusters, kinetic heap
	Lu, 2011	clustering trajectories through semantic regions	-	-	✓	-	•density threshold •distance radius •min # points	not specified	-
	Gariel, 2011	detection of aircraft waypoints and nominal trajectories, aircraft anomaly detection	-	-	✓	-	•DBSCAN parameters •# PCs •# sample points	not specified	-
Cluster Evolution	**Kalnis, 2005**	detecting moving clusters	-	-	✓	-	•DBSCAN parameters •integrity threshold Θ		pruning redundant cluster comb./approximations
	Iyengar, 2004	detecting evolving space-time clusters	-	-	-	✓	•max. population candidate size •nr. of search iterations	not specified	-
	Neill, 2005	detecting evolving space-time clusters	-	-	-	✓	•temporal window size •level of data aggregation	O(N_ST+N^4T)	-
	Li, 2004	clustering moving objects into micro-clusters to enable generic clustering	-	✓	-	-	•split threshold	O((N+U)*log(N+U)* log(N)+N*log^3(N))	heuristics
	Jensen, 2007	continuous moving-object clustering	-	✓	-	-	•max. update time/cluster capacity •time window size	not specified	disk-based data structure, event queue, hash tab.
	Chen, 2007	clustering moving-objects on a network	✓	-	✓	-	•max distances	not specified	cluster blocks
	Rosswog, 2008	detecting and tracking spatio-temporal clusters with adaptive history filtering	✓	-	-	-	•order of the filter, filter type (flat, adaptive, adaptive+stability)	not specified	results improve as history grows
	Jeung, 2008	clustering of convoys, i.e., groups of objects that travel together for at least a min. duration of time	-	-	✓	-	•min # of cluster objs. •distance value e •lifetime k, length of time partition λ •tolerance of trajectory simplification δ	not specified	clustering on simplified trajectories for detection of convoy candidates
	Rosswog, 2012	DBSCAN-based ST-clustering backed by a SVM	-	-	✓	-	•parameterized kernel •DBSCAN •training data set required	Relationship graph: O(n*log n)	R-Tree
	Zheng, 2013	clustering of gatherings	-	-	✓	-	•DBSCAN parameters •variation threshold δ •lifetime threshold of crowd/participator •support threshold of crowd/gathering	not specified	diff. spatial indexing techniques, pruning

Fig. 3. Clustering Criteria Table (part 1)

Evolution mostly supported (E). Algorithms from the NE group work with objects in fixed locations and thus do not support spatial evolution at all. All other techniques allow for moving objects.

Spatial and temporal extent not handled (SP, TP). None of the surveyed techniques can handle anything but spatial points or temporal instances — lines, rectangles or temporal intervals are not dealt with. Hence, currently none of the approaches is able to directly deal with the heterogeneity (HID) criterion.

Fuzziness of data is not an issue (FID). All of the approaches treat objects as facts and do not consider any uncertainties, except Jensen et al. [16] who consider that objects *might disappear* without notifying the server and reduce the confidence in the assumed object movement as time passes.

Almost no context knowledge supported (CID). Most of the algorithms work without any knowledge of context and do not allow any further information than a data set and parameter values. Exceptions are techniques that cluster the objects backed by a network graph (e. g., [5]), or techniques that make use of velocity information for moving objects (e. g., [16], [23]).

Techniques mainly focus on offline clustering (OOA). While the surveyed techniques mainly aim at offline clustering of closed data-sets, only few exceptions (from each group) allow online clustering of ever-changing data-sets (e. g.,

		Functional Criteria															
		Cluster		Analysis Properties												Determinism (D)	
				Spatial (SA)			Temporal (TA)				Focus (F)						
							Pattern (TAP)										
S... spherical R... rectangular A... arbitrary		Shape (CS)	Overlap (CO)	Point	Line	Region	Linear	Cyclic	Instant	Interval	spatial	temporal	spatio-temp.	Noise (N)	Similarity Measures (SM)	deterministic	heuristic
No Evo.	Wang, 2006	A	-	✓	-	-	✓	-	✓	-	-	-	✓	-	n.a. / arbitrary	✓	-
	Birant, 2006	A	-	✓	-	-	✓	✓	✓	-	✓	-	-	density parameter	arbitrary (ED used for evaluation)	✓	-
	Tai, 2007	A	-	✓	-	-	✓	-	✓	-	✓	-	-	-	cost function	✓	-
Object Evo.	Gaffney, 1999	A	-	-	✓	-	✓	-	✓	-	-	-	✓	-	Membership probabilities	-	✓
	Nanni, 2006	A	-	-	✓	-	✓	-	-	✓	-	✓	✓	-	average ED between objects	✓	-
	Li, 2010	A	-	-	✓	-	✓	-	✓	-	-	-	✓	micro-cluster extent	composed three-part distance	✓	-
	Lu, 2011	A	-	-	✓	-	✓	-	✓	-	-	-	✓	-	weighted squared ED	✓	-
	Gariel, 2011	A	✓	-	✓	-	✓	-	✓	-	-	-	✓	projecting trajectories to PCs	dist. betw. projections on PCs	✓	-
Cluster Evolution	Kalnis, 2005	A	-	✓	-	-	✓	-	✓	-	✓	-	-	-	spatial distance measure (e.g., ED)	✓	-
	Iyengar, 2004	R	-	✓	-	-	✓	-	✓	-	-	-	✓	-	n.a.	-	✓
	Neill, 2005	R	-	-	-	✓	✓	✓	-	✓	-	-	✓	-	n.a.	-	✓
	Li, 2004	S	-	✓	-	-	✓	-	✓	-	-	-	✓	-	ED (exchangable)	✓	-
	Jensen, 2007	S	-	✓	-	-	✓	-	✓	-	-	-	✓	-	weighted squared ED over time period	✓	-
	Chen, 2007	A	-	✓	-	-	✓	-	✓	-	✓	-	-	-	network distance	✓	-
	Rosswog, 2008	S	✓	✓	-	-	✓	-	✓	-	✓	-	-	-	filtered ED squared over time period	-	(✓)
	Jeung, 2008	A	-	-	✓	-	✓	-	-	✓	-	-	✓	-	specialized measure of ED between simplified trajectories	✓	-
	Rosswog, 2012	A	✓	✓	-	-	✓	-	✓	-	-	-	✓	SVM	n.a.	✓	-
	Zheng, 2013	A	-	-	✓	-	✓	-	-	✓	-	-	✓	-	Hausdorff distance	✓	-

Fig. 4. Clustering Criteria Table (part 2)

[32], [10], [16]).

Majority of algorithms is density-based (AC). Our survey comprises clustering techniques from all classes of algorithms, whereby `OE` techniques are usually density based, while `CE` techniques cover all algorithm classes.

Predominance of arbitrarily shaped clusters (CS). `OE` and `NE` techniques mostly result in arbitrary clusters, whereas clusters produced by `CE` techniques are often restricted to spherical or rectangular shape (e. g., [14], [30]).

Cluster overlap in CE techniques (CO). Overlapping clusters are only considered by few of the techniques mostly from the `CE` group (e. g., Kalnis et al. [18]). These approaches are able to detect different clusters moving through each other and can keep them separated until they split again.

Spatial analysis properties highly dependent on intent of analysis (SA). While `NE` and `CE` techniques are focused on clustering data points, `OE` techniques mostly deal with clustering of lines, since a trajectory can be interpolated to a line. Li et al. [23] first group the objects into micro-clusters (i. e., small regions) and then combine these regions to complete clusters. However, as they cluster the micro-clusters center points, they also deal with points only.

Only temporal instants (TA) and linear patterns (TAP). Linear patterns for time instants are predominant in the surveyed approaches, while other temporal properties like cyclic time patterns or intervals are unhandled in the majority of cases, especially by `CE` techniques. Only Birant et al. [4] include cyclic time patterns, and Nanni et al. [27] consider time intervals.

Focus on ST data (F). Most of the techniques are focused on handling the specific nature of ST data. However, there are several spatial-data dominant

techniques originally stemming from the field of spatial clustering, enriched with the ability of dealing with temporal data aspects (e. g., [4], [32]). Only Nanni et al. [27] presented a temporal-data dominant variation of their algorithm.

Sparse explicit noise handling (N). As most of the proposed techniques extend well-known clustering algorithms to work with ST data, the handling of noise highly depends on the abilities of the underlying base algorithm. An example for an often used algorithm is DBSCAN [6], which is generally tolerant to noise, although it is prone to errors if clusters of different densities exist. Only Birant et al. [4] and Rosswog et al. [30] explicitly address the topic of noise and inconsistencies by proposing an additional density parameter or a stability filter. As detection of noise might be used to find ST outliers, approaches that deal with anomalies might be applicable in different ways within an SAW system.

Euclidean Distance as similarity measure (SM). Regarding similarity measures, most authors make use of an adjusted version of the Euclidean distance (ED) (e. g., weighted ED, squared ED). Obvious exceptions are Chen et al. [5] who operate on a network graph and thus use a network distance and some special approaches that do not require a similarity measure at all (e. g., [33]). A border case is the similarity measure used by Tai et al. [32], who define their own cost function looking similar to the ED, considering only (what they call) optimization attributes, thus excluding spatial components.

Mostly deterministic techniques (D). Only few of the techniques use heuristic approaches to deliver fast, but more inexact results (e. g., Iyengar et al. [14] who propose a heuristic cluster search because of the huge search space).

Various levels of configurability (C). The number of parameters ranges from none or automatically approximated to up to seven parameters. For instance, DBSCAN-based algorithms usually require a minimum number of points per cluster and a desired cluster radius, while grid based clustering techniques require a grid-cell border length. We highlight the approach by Wang et al. [33], where parameters can be chosen arbitrarily, but might be approximated by an optional technique, thus offering a hybrid approach of parameter settings.

Computational complexity rarely included (CC). Most authors did not provide the computational complexity of their approaches. Noticeable is the grid based algorithm proposed by Wang et al. [33] that has the advantage of a linear runtime complexity, as a single iteration of the grid is sufficient to discover ST clusters.

Few optimization strategies (O). Only some of the authors suggest usage of special optimization strategies, like indexes or pruning. Some authors suggest different variants of their algorithms, offering a trade-off between execution efficiency and quality of results. For example Tai et al. [32] suggest an exact technique for cluster discovery, and a second variant that is more efficient in execution, but might not yield the best results.

4.3 Conclusion.

In this paper we focused on ST clustering approaches in the domain of SAW. We proposed evaluation criteria stemming from the field of SAW on the one

hand and from the field of ST data mining on the other hand, to evaluate the approaches with respect to their applicability to the field of SAW.

Summing up, we state that none of the surveyed approaches fulfills all the criteria stemming from the special nature of data in SAW systems. Spatial and temporal extent (`SP`) as well as cyclic time patterns (`TAP`) are not the focus of ST clustering techniques. Also fuzzy input data (`FID`) is not handled in the predominant number of cases and only few online approaches (`OOA`) exist. As long as no appropriate techniques are available, we suggest transformation of the input data to enable application of the clustering techniques reviewed in this survey (e. g., only the starting points of traffic jams could be used for clustering, in order to apply techniques operating on point input data only and thus discarding the information about their extent).

References

1. Aggarwal, C.C., et al.: A framework for clustering evolving data streams. In: Proc. of 29th Int. Conf. on Very Large Data Bases. pp. 81–92. VLDB Endowment (2003)
2. Andrienko, G., Andrienko, N.: Interactive cluster analysis of diverse types of spatio-temporal data. ACM SIGKDD Explorations Newsletter 11(2), 19–28 (2009)
3. Baumgartner, N., et al.: Beaware!—situation awareness, the ontology-driven way. Int. Journal of Data and Knowledge Engineering 69(11), 1181–1193 (2010)
4. Birant, D., Kut, A.: St-dbscan: An algorithm for clustering spatial and temporal data. Data & Knowledge Engineering 60(1), 208–221 (2007)
5. Chen, J., et al.: Clustering moving objects in spatial networks. In: Proc. of 12th Int. Conf. on Database Systems for Advanced Appl. pp. 611–623. Springer (2007)
6. Ester, M., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proc. of 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD). pp. 226–231. AAAI Press (1996)
7. Feng Cao, et al.: Density-based clustering over an evolving data stream with noise. In: SIAM Conf. on Data Mining. pp. 328–339 (2006)
8. Gaber, M.M., et al.: Mining data streams: a review. ACM SIGMOD Record 34(2), 18–26 (2005)
9. Gaffney, S., Smyth, P.: Trajectory clustering with mixtures of regression models. In: Proc. of the 5th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. pp. 63–72. KDD '99, ACM (1999)
10. Gariel, M., et al.: Trajectory clustering and an application to airspace monitoring. Trans. Intell. Transport. Sys. 12(4), 1511–1524 (2011)
11. Han, J., et al.: Spatial clustering methods in data mining: A survey. Geographic Data Mining and Knowledge Discovery pp. 1–29 (2011)
12. Han, J., et al.: Data Mining Concepts and Techniques. Morgan Kaufmann, 3 edn. (2011)
13. Ikonomovska, E., et al.: A survey of stream data mining. In: Proc. of 8th National Conf. with Int. Participation (ETAI) (2007)
14. Iyengar, V.S.: On detecting space-time clusters. In: Proc. of 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD). pp. 587–592. AAAI Press (1996)
15. Jain, A.K., et al.: Data clustering: a review. ACM Computing Surveys 31(3), 264–323 (1999)
16. Jensen, C., et al.: Continuous clustering of moving objects. IEEE Transactions on Knowledge and Data Engineering 19(9), 1161–1174 (2007)

17. Jeung, H., et al.: Discovery of convoys in trajectory databases. Proc. VLDB Endow. 1(1), 1068–1080 (2008)
18. Kalnis, P., et al.: On discovering moving clusters in spatio-temporal data. In: Proc. of the 9th Int. Conf. on Advances in Spatial and Temporal Databases (SSTD). pp. 364–381. Springer (2005)
19. Kalyani, D., Chaturvedi, S.K.: A survey on spatio-temporal data mining. Int. Journal of Computer Science and Network (IJCSN) 1(4) (2012)
20. Kavitha, V., Punithavalli, M.: Clustering time series data stream - a literature survey. Int. Journal of Computer Science and Inf. Sec. (IJCSIS) 8(1) (2010)
21. Kisilevich, S., et al.: Spatio-temporal clustering: a survey. Data Mining and Knowledge Discovery Handbook pp. 1–22 (2010)
22. Kranen, P., et al.: The clustree: Indexing micro-clusters for anytime stream mining. Knowledge and Information Systems Journal 29(2), 249–272 (2011)
23. Li, Y., et al.: Clustering moving objects. In: Proc. of the 10th ACM Int. Conf. on Knowledge Discovery and Data mining (KDD). pp. 617–622. ACM (2004)
24. Li, Z., et al.: Incremental clustering for trajectories. In: Proc. of the 15th Int. Conf. on Database Systems for Advanced Appl. (DASFAA). pp. 32–46. Springer (2010)
25. Lu, C.T., et al.: A framework of mining semantic regions from trajectories. In: Proc. of the 16th Int. Conf on Database Systems for Advanced Applications. pp. 193–207. DASFAA'11, Springer (2011)
26. Matheus, C., et al.: Sawa: An assistant for higher-level fusion and situation awareness. In: Proc. of SPIE Conf. on Multisensor, Multisource Information Fusion. pp. 75–85. Architectures, Algorithms, and Applications (2005)
27. Nanni, M., Pedreschi, D.: Time-focused clustering of trajectories of moving objects. Journal of Intelligent Information Systems 27(3), 267–289 (2006)
28. Neill, D.B., et al.: Detection of emerging space-time clusters. In: Proc. of the 11th ACM SIGKDD Int. Conf. on Knowledge discovery in Data Mining. pp. 218–227. KDD '05, ACM (2005)
29. R.Geetha, et al.: A survey of spatial, temporal and spatio-temporal data mining. Journal of Computer Applications 1(4), 31–33 (2008)
30. Rosswog, J., Ghose, K.: Detecting and tracking spatio-temporal clusters with adaptive history filtering. In: Proc. of 8th IEEE Int. Conf. on Data Mining, Workshops (ICDMW). pp. 448–457 (2008)
31. Rosswog, J., Ghose, K.: Detecting and tracking coordinated groups in dense, systematically moving, crowds. In: Proc. of the 12th SIAM Int. Conf. on Data Mining. pp. 1–11. SIAM / Omnipress (2012)
32. Tai, C.H., et al.: Incremental clustering in geography and optimization spaces. In: Zhou, Z.H., Li, H., Yang, Q. (eds.) Advances in Knowledge Discovery and Data Mining, Lecture Notes in Computer Science, vol. 4426, pp. 272–283. Springer Berlin Heidelberg (2007)
33. Wang, M., et al.: Mining spatial-temporal clusters from geo-databases. In: Advanced Data Mining and Applications, Lecture Notes in Computer Science, vol. 4093, pp. 263–270. Springer Berlin Heidelberg (2006)
34. Warren Liao, T.: Clustering of time series data-a survey. Pattern Recogn 38(11), 1857–1874 (2005)
35. Wimmer, M., et al.: A survey on uml-based aspect-oriented design modeling. ACM Computing Surveys 43(4), 28:1–28:33 (2011)
36. Xu, R., Donald C. Wunsch II: Survey of clustering algorithms. IEEE Transactions on Neural Networks 16(3), 645–678 (2005)
37. Zheng, K., et al.: On discovery of gathering patterns from trajectories. In: IEEE Int. Conf. on Data Engineering (ICDE) (2013)