BibTeX:

On Event Detection from Spatial Time series for Urban Traffic Applications

Gustavo Souto[1] and Thomas Liebig[2]

[1] Fraunhofer Institute for Software and Systems Engineering, Fraunhofer ISST
Gustavo.Souto@isst.fraunhofer.de
[2] TU Dortmund University, Artificial Intelligence Group
thomas.liebig@tu-dortmund.de

Abstract. Since the last decades the availability and granularity of location-based data has been rapidly growing. Besides the proliferation of smartphones and location- based social networks, also crowdsourcing and voluntary geographic data led to highly granular mobility data, maps and street networks. In result, location-aware, smart environments are created. The trend for personal self-optimization and monitoring named by the term 'quantified self' will speed-up this ongoing process. The citizens in conjunction with their surrounding smart infrastructure turn into 'living sensors' that monitor all aspects of urban living (traffic load, noise, energy consumption, safety and many others). The "Big Data"-based intelligent environments and smart cities require algorithms that process these massive amounts of spatio-temporal data. This article provides a survey on event processing in spatio-temporal data streams with a special focus on urban traffic.

1 Introduction

Early detection of anomalies in spatio-temporal data streams provides many applications for smart cities and is a major research topic since the availability and granularity of location-based data has been rapidly growing in the last decades.

Besides, the proliferation of smartphones and location- based social networks, also crowdsourcing and voluntary geographic data led to highly granular mobility data, maps and street networks. In result, location-aware, smart environments are created. The trend for personal self-optimization and monitoring named by the term 'quantified self' will speed-up this ongoing process. The citizens in conjunction with their surrounding smart infrastructure turn into 'living sensors' that monitor all aspects of urban living (traffic load, noise, energy consumption, safety and many others).

The "Big Data"-based intelligent environments and smart cities require algorithms that process these massive amounts of spatio-temporal data in real-time. But key challenges for streaming analysis are (1) one-pass processing (2) limited amount of memory and (3) limited time to process [6].

Spatio-temporal data comes in a variety of forms and representations, depending on the domain, the observed phenomenon, and the observation method.

In principle, there are three types of spatio-temporal data streams [19]: *spatial time series*, *events*, and *trajectories*.

– A *spatial time series* consists of tuples $(attribute, object, time, location)$.
– An *event* of a particular type $event_i$ is triggered from a spatial time series under certain conditions and contains the tuples verifying these conditions $(event_i, object_n, time_n, location_n)$.
– A *trajectory* is a spatial time series for a particular $object_i$. It contains the location per time and is a series of tuples $(object_i, time_n, location_n)$.

The increasing availability of massive heterogeneous streaming data for public organizations, governments and companies pushes their inclusion in incident recognition systems. Leveraging insights from these data streams offers a more detailed and real-time picture of traffic, communication, or social networks, to name a few, which still is a key challenge for early response and disaster management. Detecting events in spatio-temporal data is a widely investigated research area (see e.g. [1] for an overview). Depending on the application, the event detection can analyze single trajectories (e.g. of persons or vehicles), group movements, spatio-temporal measurements, or heterogeneous data streams. Following examples highlight capabilities of these approaches:

– *Individual Mobility:* Within airports (or other security region) it is valuable to monitor whether individuals enter some restricted area. The analysis of stops or of sudden decelerations allows detection of unusual behaviour. Sequences of such events can be matched against predefined mobility patterns [12], e.g. to identify commuters.
– *Group Movement:* During public events the early detection of hazardous pedestrian densities gains much attention. The patterns one could distinguish and detect in group movement are *encounter*, *flock* or *leadership* pattern [10].
– *Spatio Temporal Measurements:* A spatio-temporal value spans a whole region. This could be traffic flow, air pollution, noise, etc. The sudden rise or decline of these values indicates an anomaly.
– *Heterogeneous Data Streams:* The combination of previously described types of anomalies provides event filters in an urban environment based on heterogeneous data (e.g. GPS data of pedestrians, traffic loop data, mobile phone network data).

In the paper at-hand we provide a introductory survey on (1) functions on heterogeneous spatio-temporal data streams, Section 2, (2) pattern matching, Section 3, (3) anomaly detection in spatio-temporal time series, Section 4, and (4) streaming frameworks, Section 5. All four aspects are relevant for implementing real-world event detection systems that process heterogeneous data streams.

2 Function Classes on Heterogeneous Spatio-Temporal Time Series

In general functions for event detection from heterogeneous data streams can be classified using a former concept of raster-geography, namely *map-algebra* [5].

Both, raster geography and heterogeneous spatio-temporal data analysis consider data which is provided in multiple layers (i.e. one layer per data stream). Functions can be applied to one or multiple layers. Thus, spatial functions split into four groups: *local*, *focal*, *zonal* and *global* ones [5], illustrated in Figure 1.

- *Local functions* operate on every single cell in a layer. And the cell is processed without reference to surrounding cells. An example is a map transformation, the multiplication with a constant, or the comparison with a threshold.
- *Focal functions* process cell data depending on the values of neighboring cells. The neighborhood can be defined by arbitrary shapes. Example functions are moving averages and nearest neighbor methods.
- *Zonal functions* process cells on the base of zones, these are cells that hold a common characteristic. Zonal functions allow the combination of heterogeneous data streams in various layers by application of functions to one layer if another layer already fulfills another condition.
- *Global functions* process the entire data. Examples are distance based operations.

For heterogeneous data streams analysis, expressiveness of these four function types is important to *derive* low-level events (incidents), to *combine* low-level events (e.g. aggregation, clustering, prediction etc.) and to *trigger* high-level events.

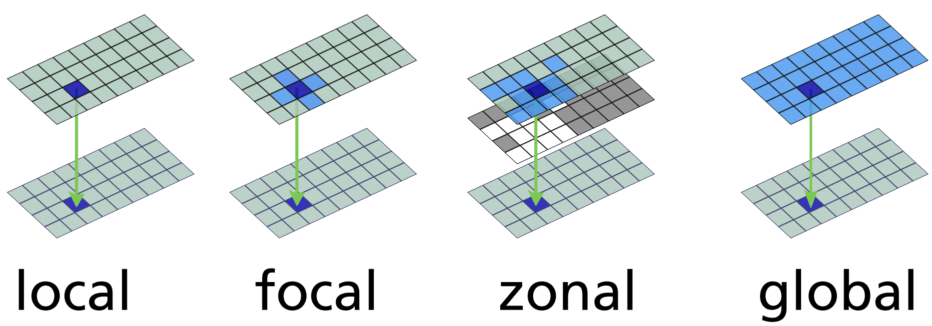

Fig. 1. Function classes on Spatio-Temporal data, Dark blue highlights the currently processed location. Light blue cells indicate the regions whose values are used for computation. Best viewed in color.

3 Event Pattern Matching

The exploitation of spatio-temporal event patterns is a major research field in mobility mining. Event pattern matching focuses on the task to match sequences of events against event patterns and to trigger another event (which is raised for further analysis) in case the sequence matches. Recently, pattern-graphs were introduced in [27], their pattern description is capable to express the temporal relations among various occurring events following the interval-calculus [2]. As an example the co-occurrence of two low-level events may trigger any high-level event. With spatio-temporal data streams also spatial relations are important to consider. The region connection calculus [28] lists relations of spatial events that are essential for a spatio-temporal pattern matcher.

Possible frameworks for event pattern matchers are the event calculus [32], finite automaton [12] and other pattern matcher [8, 27] or even complex frameworks which allow application of local, focal, zonal and global functions e.g. [30, 14]. The requirements for spatio-temporal pattern matcher in a smart city scenario are:

- to operate in real time,
- to incorporate spatial [28] and temporal [2] relations
- to provide local, focal, zonal, and global [5] predicates on the attributes, and
- to pose arbitrary queries formed of these elements (regular language [23], Kleene closure [18]).

In Table 1 we compare the features of state-of-the-art event detection frameworks. The temporal expressiveness is split into the following four categories:

- *Pattern Duration* is a constraint on the temporal distance of first and last condition in a pattern.
- *Condition Duration* is a constraint on the duration of a condition to get matched.
- *Inter-Condition Duration* is a constraint on the temporal distance among succeeding conditions.
- *Complete* indicates the complete integration of the temporal relations [2].

The Table also compares the approaches from the literature against the INSIGHT architecture, we introduced in [31]. This approach is inspired by the TechniBall system [14], previous works on stream data analysis [13] and follows the Lambda architecture design principles for Big Data systems [22]. A sketch of the architecture and the interconnection among the components is presented in Figure 2. Every data stream is analysed individually for anomalies. In this detection functions (e.g. clustering, prediction, thresholds, etc.) on the data streams can be applied. The resulting anomalies are joined at a round table. A final Complex Event Processing component allows the formulation of complex regular expressions on the function values derived from heterogeneous data streams.

Approach	**Time Algebra [2]**				**Spatial Algebra [28]**	**Regular Expressions**	**Spatial Functions [5]**				**Stream Processing**
	complete	**condition duration**	**inter-condition duration**	**pattern duration**			**local**	**focal**	**zonal**	**global**	
Mobility Pattern [23]	-	✓	-	-	-	✓	-	-	-	-	-
SASE [17]	-	✓	-	✓	-	✓	✓	✓	✓	✓	✓
$SASE^+$ [9]	-	✓	-	-	-	✓	✓	✓	✓	✓	✓
Cayuga [8]	-	-	-	✓	-	✓	✓	✓	✓	✓	✓
Spatio-Temporal Pattern Queries [30]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
Mobility Pattern Stream Matching [12]	-	✓	-	✓	-	✓	-	-	-	-	✓
Event calculus [32, 3]	✓	✓	-	✓	-	✓	✓	✓	✓	✓	✓
Temporal Pattern Graphs [27]	✓	✓	✓	✓	-	✓	✓	-	-	-	✓
INSIGHT architecture [31]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1. Comparison of Spatio-Temporal Event Detection Frameworks

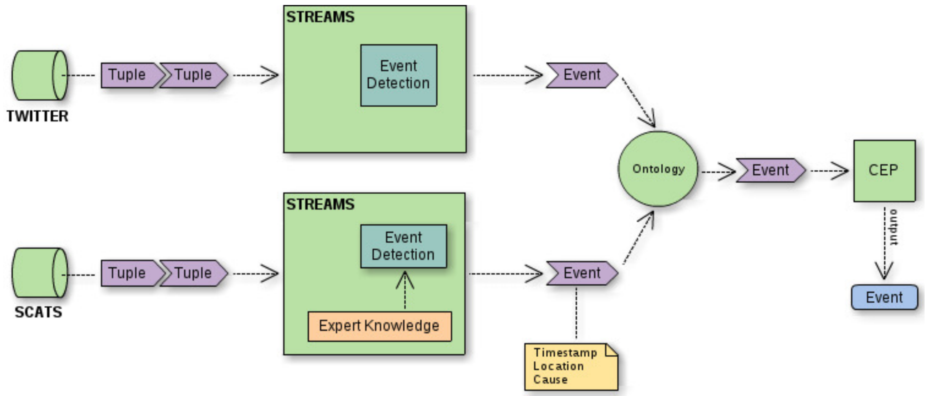

Fig. 2. INSIGHT Architecture for event detection from heterogeneous data streams exemplified with two input streams Twitter and traffic loop data derived by SCATS, compare [31, 4].

4 Anomaly Detection on Spatial Time Series

This section discusses state-of-the-art of anomaly detection in traffic condition data streams as this paper focuses on smart cities and traffic is a major aspect of a smart city. However, some techniques generalize also to other spatio-temporal phenomena as noise, pollution, etc. For a comprehensive survey on outlier detection from spatio-temporal data streams we point the reader to [16].

4.1 Statistical Approach

Pang et. al proposed an approach [25] which extends the Likelihood Ratio Test (LRT) framework to detect abnormal traffic patterns in taxi trajectory data (GPS trajectories). The approach partitions the road network of Beijing into a spatial grid, regions (R), to deal better with the problem of finding abnormal patterns. The extended LRT uses statistical models which are Persistent Spatiotemporal Model (PSTO) and Emerging Spatiotemporal Outlier Model (ESTO) to compute the likelihood of "anomalousness" of a region and detect the emerging spatio-temporal outliers, respectively. In addition, the proposed statical model works with the Maximum Likelihood Estimation (MLE) and Upper-bounding strategy to estimate the parameters of models and prune the non-outliers, respectively. However, this approach does not use other source of data (e.g. weather, list of events in the city, social network) to reduce the uncertainty of detected events, as well as it does not present a good ratio of adaptability to face natural changes in the data stream over time.

In [34], Yang et. al present a non-parametric Bayesian method, or Bayesian Robust Principal Component Analysis (RPCA) - BRPCA, to detect traffic events on a road. This method takes the traffic observations as one dimensional data and converts it into a matrix format which in turn decomposes it into a superposition of low-rank, sparse, and noise matrices. In addition, this method proposed an extended BRPCA to deal with multiple variables/time series/data streams. The idea of that extended BRPCA is to improve the traffic detection by sharing a sparsity structure among multiple data streams affected by the same events. Such an approach uses multiple homogeneous data streams and a static weather data source in the detection process.

In [26], although the major goal of this work is not detect outlier itself, the authors propose a novel adaptive Artificial Neural Network (ANN) based filter to detect and remove them to build a training data. The ANN filter uses the training set (i.e., usually the 3 months of historical data - information from street loops) as incoming and thus analyzes whether the readings are twice the maximum value, if it holds true, then the method marks it as anomaly, otherwise removed.

In [36], the authors propose an approach to estimate the traffic which uses mobile probes to detect outliers in Handover Data of a suburban freeway. The approach detects anomalies in 2 steps. The first step applies Least Squares Support Vector Machine (LS-SVM) ensemble classifier to identify whether each individual handover link is an outlier or not, and the second step employs a statistical-based algorithm which evaluates whether the detected outlier holds any locally handover link which is anomalous as well.

Trilles et al. [33] propose a variation of the CUmulative SUM (CUSUM) algorithm to detect anomalies in data streams near to real-time. This approach is only applied when the observations are in-control, that is, the data is normally distributed. In the anomaly detection process the CUSUM is obtained by computing $S_i = S_{i-1} \cdot z_i$, where z_i is a standard normal variable which is computed as follows $z_i = \frac{x_i - \bar{x}}{s}$, where the s is the standard deviation of the time series, and x_i is the i-th data point of the time series. The events are detected by the Equation 1, if S_{H_i} exceeds a predefined threshold (CUSUM control charts) $\hat{A} \pm h\sigma_x$ ($h = 5$ and σ_x is the standard deviation), then it is an *Up-Event* due to its increase and if S_{L_i} is greater than threshold (CUSUM control charts) $\hat{A} \pm h\sigma_x$ ($h = 5$ and σ_x is the standard deviation), then it is an *Down-Event* due to its decrease. The variable k is a slack-variable and denotes the reference value which is usually set to be one half of the mean. The advantages of this work are the application of a simple approach for Real-Time anomaly detection and the dashboad application to visualize the detected events. However, the work does not present experiments with a data source which has high refresh rate such as SCATS data stream.

$$\begin{aligned} S_{H_i} &= MAX[0, (z_i - k) + S_{H_i} - 1] \\ S_{L_i} &= MIN[0, (z_i - k) + S_{L_i} - 1] \end{aligned} \tag{1}$$

4.2 Human/Driver's Behavior

Pan [24] proposes a new method to detect disruptions in typical traffic patterns (traffic anomalies) using crowd-sourcing and social media. This approach detects anomalies according to drivers' routing behaviour instead of traffic volume-based and speed on roads. In addition, it provides a view of congested road segments and their relationships among these segments. It also provides to the end-user a detour router to avoid or escape the congestions. This method also makes use of a historical tweets associated with the spatial region to represent the normal occurrences of each region. In order to retrieve only the relevant contents, this approach applies a simple filtering technique which compares the frequency of current tweets with historical tweets and apply a weight to each term according to its frequency, as well as the location and time information.

4.3 Unsupervised

Yang [35] investigates the problem of outlier detection on large-scale collective behaviors. His work extracts features from high-dimensional data streams using K-Nearest Neighbors (KNN) method to detect the anomalies. This method performs the anomaly detection in 3 phases as follows: (1) observations from multiple sensors, this phase organizes more than 400 sensors as high-dimensional time series; (2) manifold learning, it applies Locally Linear Embedding (LLE) computes and Principal Component Analysis (PCA) to obtain a feature at a higher abstraction level; and (3) outlier detection, this phase performs the outlier detection through the K-Nearest Neighbors. The approach works good since special days, or holidays, which might generate an abnormal flow are known in advance. For instance, New Year and Independence Day. However, from this characteristics, it indicates that the method cannot handle historical data as well as adapt itself according to the changes.

Guo et al. [15] propose a traffic flow outlier detection approach which focuses on the pattern changing detection problem to detect anomalies in traffic conditional data streams. The traffic data comes from inductive loop sensors of four regions in United State and United Kingdom as well as this works makes use of a short-term traffic condition forecasting system to evaluate the proposed approach. This approach performs the analysis of the incoming data point after the data point be processed by Integrated Moving Average filter (IMA) which captures the seasonal effect on the level of traffic conditional series, and then Kalman filter picks up the local effect flow levels after IMA, and GARCH filter models and predict time-varying conditional variance of the traffic flow process. These filters constitute together the integrated forecast system aforementioned.

4.4 Tree Approach

Liu et al. [21] present an approach based on features analysis to detect outliers points as well as trees which detects the relationship among anomalies in traffic data stream. This work uses taxi trajectory data (GPS trajectories) on the road network of Beijing. The approach presents a model with 3 main steps which processes the traffic data to build a region graph. The 3 main steps are (1) Building a region graph, (2) Detect outliers from graph edges, and (3) Discover relations among outliers (building a tree). Then, this method partitions the map of traffic into regions by employing Connected Components Labeling. Each region holds a link to other region and a link is anomalous whether its features have the largest difference from both their temporal and spatial neighbors, and the STOTree algorithm captures the causal relationship among outliers. Although this work presents an interesting work about correlated anomalies in traffic data streams, the work does not provide experiments under an online setting for the traffic anomaly detection. Instead, it describes a set of algorithms which could be applied in such a setting.

4.5 Discussion

Although these works present some substantial advances in the field of anomaly detection in data streams, the field is still in its early stage, and therewith it is possible to see that such works hold some drawbacks as well as open tasks. Examples of open tasks are incorporate heterogeneous data streams, keep tracking of historical data (local and global), apply adaptive data stream models, use expert knowledge, develop straightforward and lightweight approaches for data stream analysis. These open tasks aim to improve the anomaly detection in data streams (in general), that is, decreasing the uncertainty whether the detected event is a true anomaly.

The use of heterogeneous data streams improves detection of anomalies by reducing the uncertainty about the events veracity, this issue has been little exploited in the traffic conditions domain. Outlier detectors should take into account external factors (e.g., weather and social events), such an issue has been exploited more than heterogeneous data streams, but their applications only refer to sources which provide static information, or general information from online forecast sources (e.g., wind speed, amount of rainfall, humidity), instead of precise information about what is happening around the city by using local sensors (e.g., flood in a specific region of a city). The works [31] and [4] use heterogeneous data streams to detect anomalies in a smart city, but it is still some open questions which need answers such as "*How to merge the flow of heterogeneous data streams to obtain a good result?*" and "*How to join the result of the analysis of each flow to detect the true anomaly detection?*".

Adaptive classification models react to the natural changes of data stream. The change of the target variable value in which the model is trying to predict is well-known as *Concept Drift*. A model which adapts itself over time holds more chances to find a true anomaly than another model without such characteristic. Therefore, this feature is also important to find true anomalies, for more details, see [29] and [11]. Except [4] which applies adaptive function in its complex event processing (CEP), none of the other works we discussed in this work holds this characteristic in their approaches.

The expert knowledge data issue addresses interesting challenges for the anomaly detection in traffic condition. The expert knowledge along with a base of knowledge acquired during the detection process in traffic conditions data stream is an interesting challenge which should receive more attention from now on, because this topic has not been well explored in traffic conditions data streams domain and its use can raise the rate of true anomalies by reducing the uncertainty in the data. None of works we present in this work approach such a concept, the exception are [31] and [20] which use traffic network data from OpenStreetMap[3] (OSM).

Straightforward and lightweight anomaly detection approaches lead to the data stream analysis in critical environments (e.g. old devices, or even mobile ones in smart cities). This open task is important in traffic conditions field since data emitters might apply some privacy constraints, and therewith the device next to the sensor (e.g. SCATS - region computer) around the city, or user mobile device (i.e., small agent running in smartphone), (pre-)processes part of the data stream before send it to a central server. Therefore, anomaly detection approaches must also satisfy such resource constraints on consumption of energy/battery, CPU and memory.

5 Streaming Frameworks for Anomaly Detection

The implementation of previously presented real-time event detection algorithms (Section 4) and event pattern matchers (Section 3) is usually done in a streaming framework. A streaming framework models the data flow in the analysis process and therefore the connections of the streams to the individual process steps. The data from one step to the next is transferred as messages. In general, a streaming framework is characterized by the following features [7]:

- *Message Processing Semantics* describes how often a message is processed in the framework, and which ordering of the messages is assumed by the framework.
- *State Handling and Fault Tolerance* describing how the streaming framework provides fault tolerance. Usually, a streaming framework provides fault tolerance by resending data that has not been acknowledged by the recipient.
- *Scalability* describes how the streaming frame work scales out in case of increasing resources.
- *Portability* describes whether the execution is bound to a specific platform, or whether it could also be executed in other, e.g. embedded, environments.

In [7] the state-of-the-art streaming frameworks are compared according to this feature list.

Acknowledgements This research was supported by the National Council for Scientific and Technological Development (CNPq), the European Union's Seventh Framework Programme under grant agreement number FP7-318225, INSIGHT. Additionally, this work has been supported by Deutsche Forschungsgemeinschaft (DFG) within the Collaborative Research Center SFB 876, project A1.

[3] openstreetmap.org

References

1. Aggarwal, C.C.: Outlier Detection. Springer, New York, NY, USA (2013)
2. Allen, J.F.: Maintaining knowledge about temporal intervals. Commun. ACM 26(11), 832–843 (Nov 1983), http://doi.acm.org/10.1145/182.358434
3. Artikis, A., Weidlich, M., Gal, A., Kalogeraki, V., Gunopulos, D.: Self-adaptive event recognition for intelligent transport management. In: Big Data, 2013 IEEE International Conference on. pp. 319–325 (Oct 2013)
4. Artikis, A., Weidlich, M., Schnitzler, F., Boutsis, I., Liebig, T., Piatkowski, N., Bockermann, C., Morik, K., Kalogeraki, V., Marecek, J., Gal, A., Mannor, S., Gunopulos, D., Kinane, D.: Heterogeneous stream processing and crowdsourcing for urban traffic management. In: Proc. 17th International Conference on Extending Database Technology (EDBT), Athens, Greece, March 24-28, 2014. pp. 712–723. OpenProceedings.org (2014)
5. Berry, J.K.: Gis modeling and analysis. In: Madden, M., for Photogrammetry, A.S., Sensing, R. (eds.) Manual of Geographic Information Systems, pp. 527–585. American Society for Photogrammetry and Remote Sensing (2009), http://books.google.de/books?id=ek-IQAAACAAJ
6. Bifet, A., Kirkby, R.: Data stream mining a practical approach (2009)
7. Bockermann, C.: A survey of the stream processing landscape. Tech. Rep. 6, TU Dortmund University (5 2014), http://www-ai.cs.uni-dortmund.de/PublicPublicationFiles/bockermann_2014b.pdf
8. Demers, A., Gehrke, J., Panda, B., Riedewald, M., Sharma, V., White, W.: Cayuga : A General Purpose Event Monitoring System. Publish pp. 412–422 (2007)
9. Diao, Y., Immerman, N., Gyllstrom, D.: Sase + : An agile language for kleene closure over event streams. Analysis (UM-CS-07-03), 1–13 (2007)
10. Dodge, S., Weibel, R., Lautenschütz, A.K.: Towards a taxonomy of movement patterns. Information visualization 7(3-4), 240–252 (2008)
11. Dongre, P.B., Makik, L.G.: A review on real time data stream classification and adapting to various concept drift scenarios. IEEE International Advance Computing Conference 1, 533537 (February 2014)
12. Florescu, S., Körner, C., Mock, M., May, M.: Efficient mobility pattern stream matching on mobile devices. In: Proc. of the Ubiquitous Data Mining Workshop (UDM 2012). pp. 23–27 (2012)
13. Fuchs, G., Andrienko, N., Andrienko, G., Bothe, S., Stange, H.: Tracing the german centennial flood in the stream of tweets: First lessons learned (2013)
14. Gal, A., Keren, S., Sondak, M., Weidlich, M., Blom, H., Bockermann, C.: Grand challenge: The techniball system. In: Proceedings of the 7th ACM International Conference on Distributed Event-based Systems. pp. 319–324. DEBS '13, ACM, New York, NY, USA (2013)
15. Guo, J., Huang, W., Williams, B.M.: Real time traffic flow outlier detection using short-term traffic conditional variance prediction. Transportation Research Part C: Emerging Technologies pp. 1–13 (July 2014)
16. Gupta, M., Gao, J., Aggarwal, C., Han, J.: Outlier detection for temporal data. Synthesis Lectures on Data Mining and Knowledge Discovery 5(1), 1–129 (2014)
17. Gyllstrom, D., Diao, Y., Wu, E., Stahlberg, P., Anderson, G.: SASE : Complex Event Processing over Streams. Science 1, 407–411 (2007)
18. Gyllstrom, D., Agrawal, J., Diao, Y., Immerman, N.: On supporting kleene closure over event streams. In: Alonso, G., Blakeley, J.A., Chen, A.L.P. (eds.) ICDE. pp. 1391–1393. IEEE (2008)

19. Liebig, T., Morik, K.: Report on end-user requirements, test data, and on prototype definitions. Tech. Rep. FP7-318225 D5.1, TU Dortmund and Insight Consortium Members (August 2013)
20. Liebig, T., Piatkowski, N., Bockermann, C., Morik, K.: Route planning with real-time traffic predictions. In: Proceedings of the 16th LWA Workshops: KDML, IR and FGWM. pp. 83–94 (2014)
21. Liu, W., Zheng, Y., Chawla, S., Yuan, J., Xing, X.: Discovering spatio-temporal causal interactions in traffic data streams. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1010–1018. KDD '11, ACM, New York, NY, USA (2011), http://doi.acm.org/10.1145/2020408.2020571
22. Marz, N.: Big data : principles and best practices of scalable realtime data systems. O'Reilly Media (2013), http://www.amazon.de/Big-Data-Principles-Practices-Scalable/dp/1617290343
23. du Mouza, C., Rigaux, P., Scholl, M.: Efficient evaluation of parameterized pattern queries. In: Herzog, O., Schek, H., Fuhr, N., Chowdhury, A., Teiken, W. (eds.) CIKM. pp. 728–735. ACM (2005)
24. Pan, B., Zheng, Y., Wilkie, D., Shahabi, C.: Crowd sensing of traffic anomalies based on human mobility and social media. In: Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. pp. 344–353. SIGSPATIAL'13, ACM, New York, NY, USA (2013), http://doi.acm.org/10.1145/2525314.2525343
25. Pang, L.X., Chawla, S., Liu, W., Zheng, Y.: On detection of emerging anomalous traffic patterns using gps data. Data Knowl. Eng. 87, 357–373 (Sep 2013), http://dx.doi.org/10.1016/j.datak.2013.05.002
26. Passow, B.N., Elizondo, D., Chiclana, F., Witheridge, S., Goodyer, E.: Adapting traffic simulation for traffic management: A neural network approach. IEEE Annual Conference on Intelligent Transportation Systems (ITSC 2013) pp. 1402–1407 (October 2013)
27. Peter, S., Höppner, F., Berthold, M.R.: Learning pattern graphs for multivariate temporal pattern retrieval. In: Hollmen, J., Klawonn, F., Tucker, A. (eds.) Advances in Intelligent Data Analysis XI, Lecture Notes in Computer Science, vol. 7619, pp. 264–275. Springer Berlin Heidelberg (2012)
28. Randell, D.A., Cui, Z., Cohn, A.G.: A spatial logic based on regions and connection. In: Nebel, B., Rich, C., Swartout, W.R. (eds.) KR. pp. 165–176. Morgan Kaufmann (1992)
29. Richard Kirkby Albert Bifet, G.H., Pfahringer, B.: Data Stream Mining: A Pratical Approach. The university of Waikato, The address of the publisher (May 2011)
30. Sakr, M.A., Güting, R.H.: Spatiotemporal pattern queries. GeoInformatica 15(3), 497–540 (2011)
31. Schnitzler, F., Liebig, T., Mannor, S., Souto, G., Bothe, S., Stange, H.: Heterogeneous stream processing for disaster detection and alarming. In: IEEE International Conference on Big Data. pp. 914–923. IEEE Press (2014)
32. Skarlatidis, A., Paliouras, G., Vouros, G.A., Artikis, A.: Probabilistic event calculus based on markov logic networks. In: Olken, F., Palmirani, M., Sottara, D. (eds.) RuleML America. Lecture Notes in Computer Science, vol. 7018, pp. 155–170. Springer (2011)
33. Trilles, S., Schade, S., Óscar Belmonte, Huerta, J.: Real-time anomaly detection from environmental data streams. In: Bacao, F., Santos, M.Y., Painho, M.

(eds.) AGILE 2015, pp. 125–144. Lecture Notes in Geoinformation and Cartography, Springer International Publishing (2015), http://dx.doi.org/10.1007/978-3-319-16787-9_8

34. Yang, S., Kalpakis, K., Biem, A.: Detecting road traffic events by coupling multiple timeseries with a nonparametric bayesian method. IEEE Transactions on Intelligent Transportation Systems 15(5), 1936–1946 (March 2014)
35. Yang, S., Liu, W.: Anomaly detection on collective moving patterns. IEEE International Conference on Privacy, Security, Risk, and Trust, and IEEE International Conference on Social Computing 7, 291–296 (October 2011)
36. Yueming Yuan, W.G.: Outlier detection of handover data for innersuburban freeway traffic information estimation using mobile probes. IEEE Vehicular Technology Conference (VTC Spring) pp. 1–5 (May 2011)