Event Recognition Challenges and Techniques: Guest Editors' Introduction

ALEXANDER ARTIKIS, NCSR "Demokritos", Greece
AVIGDOR GAL, Technion – Israel Institute of Technology, Israel
VANA KALOGERAKI, Athens University of Ecenomics and Business, Greece
MATTHIAS WEIDLICH, Imperial College London, UK

ACM Reference Format:
Alexander Artikis, Avigdor Gal, Vana Kalogeraki, and Matthias Weidlich. 2014. Event recognition challenges and techniques: Guest Editors' Introduction. ACM Trans. Internet Technol. 14, 1, Article 1 (July 2014), 9 pages.
DOI: http://dx.doi.org/10.1145/2632220

1. INTRODUCTION

The concept of event processing is established as a generic computational paradigm in various application fields, ranging from data processing in Web environments, over logistics and networking, to finance and medicine [Cugola and Margara 2012]. Events report on state changes of a system and its environment. Event recognition (event pattern matching [Luckham 2002]), in turn, refers to the detection of events that are considered relevant for processing, thereby providing the opportunity to implement reactive measures. Examples consist of the recognition of attacks in computer network nodes [Dousson and Maigat 2007], human activities on video content [Brendel et al. 2011], emerging stories and trends on the Social Web[1], traffic and transport incidents in smart cities [Artikis et al. 2014b], fraud in electronic marketplaces [Schultz-Møller et al. 2009], cardiac arrhythmias [Callens et al. 2008], and epidemic spread [Chaudet 2006]. In each scenario, event recognition allows one to make sense of large data streams and react accordingly.

Event recognition systems become increasingly important as we move from an information economy to an "intelligent economy," where it is not only the accessibility to information that matters but also the ability to analyse and act upon information, creating a competitive advantage in commercial transactions, enabling the sustainable management of communities, and promoting the appropriate distribution of social, healthcare, and educational services [Vesset et al. 2011]. Current businesses tend to be unable to make sense of the amounts of data that are generated by the increasing number of distributed data sources that are available daily [Manyika et al. 2011], and they need to rely more and more on automated event recognition. As an example, consider traffic management in smart cities that needs to make use of data from an increasing number and variety of sensors.

[1] `https://www.recordedfuture.com/`.

This work is funded by the EU FP7 projects SPEEDD (619435) and INSIGHT (318225).
Authors' addresses:

DOI: http://dx.doi.org/10.1145/2632220

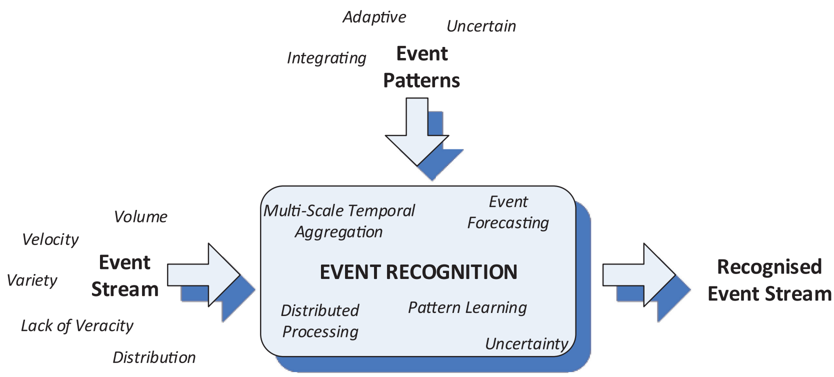

Fig. 1. Event recognition.

An intelligent economy that makes use of big data can extract actionable knowledge from it by employing event recognition systems that detect events/activities of special significance given extremely large amounts of data spreading over various geographical locations. The goal of this article is to provide an overview of the open research issues of event recognition as an introduction to this special issue. The rest of this article is structured as follows. Section 2 presents a number of research challenges of event recognition, Section 3 briefly introduces the articles in this special issue, and Section 4 concludes the article.

2. EVENT RECOGNITION RESEARCH CHALLENGES

Figure 1 outlines the key aspects of the event recognition input in terms of the event stream and event patterns. On the one hand, the characteristics of the event stream, as they are described by the four "V"s, and the distribution of events, impose challenges. That is, velocity (number of events per time unit), volume (overall amount of events), variety (differently structured events), lack of veracity (uncertainty of event occurrence), and the distribution of event sources potentially involving mobility, complicate event recognition.

On the other hand, the properties of the event patterns are largely orthogonal to the characteristics of an event stream and also add to the complexity of the event recognition task. Patterns may be required to adapt to dynamic environments. Further, patterns may integrate various event sources. Finally, patterns may be inherently uncertain in the sense that certain events can be recognised only within the bounds of a confidence interval.

Taking the dimensions outlined in Figure 1 as a starting point, five research challenges of event recognition are discussed in this section, namely, multiscale temporal aggregation, uncertainty, distribution, pattern learning, and event forecasting.

2.1. Multiscale Temporal Aggregation of Events

Composite events evolve over multiple scales of time and space [Vespier et al. 2013]. The variety of the event stream may be reflected by sources that report events at different time scales ranging from (milli-)seconds to days. For example, videos depict events at a small scale, whereas tweets may describe events at a much larger scale. Moreover, it is often the case that historical data spanning over long periods of time need to be taken into consideration [Dindar et al. 2011]. Consider, for instance, the recognition of traffic incidents using streaming bus probe data as well as several months of archived data [Bacon et al. 2011] and credit-card fraud detection using streaming and archived transactions [Artikis et al. 2014a]. Once event patterns integrate events from sources that greatly vary in their adopted scales for time and space, event recognition becomes challenging. A recognition system should be adaptable, dynamically computing the appropriate lengths of multigranular windows of varying levels of detail, and remain accurate, being able to recognise composite events from lower-level events of varying spatio-temporal granularity without, of course, compromising efficiency [Maier et al. 2012; Patroumpas 2013; Lijffijt et al. 2012].

2.2. Event Recognition under Uncertainty

Event recognition applications exhibit various types of uncertainty. Input event streams are often incomplete and include erroneous information. Sensor networks introduce uncertainty due to reasons that range from inaccurate measurements through local network failures to unexpected interference of mediators. For all of these reasons, input event streams lack veracity.

Furthermore, the rules expressing an event pattern can be imprecise, meaning that there is also uncertainty related to the events that are recognised. In many application domains, we only have imprecise knowledge about the definition of a composite event, or the available events and context information are insufficient for expressing a composite event. Consider, for example, the recognition of a fight between two people in a system of event sources that cannot distinguish between abrupt and non-abrupt human movement.

Surprisingly, uncertainty has been largely overlooked by the event processing community [Cugola and Margara 2012]. Gal et al. [2011] and Wasserkrug et al. [2012] are but a few exceptions. However, in computer vision several approaches that deal with uncertainty in event recognition have been suggested. (An overview of event recognition under uncertainty may be found in Skarlatidis et al. [2014].) Since event recognition requires the processing of streams of time-stamped events, numerous approaches are based on sequential variants of probabilistic graphical models, such as hidden markov models, dynamic bayesian networks and linear-chain conditional random fields. Such models handle uncertainty but their propositional structure provides limited representation capabilities. To address this issue, probabilistic graphical models have been extended to capture long-term dependencies between states [Hongeng and Nevatia 2003] and model the hierarchical composition of events [Natarajan and Nevatia 2007]. The lack of a formal representation language, however, makes composite event definition complicated and the use of background knowledge very hard.

Recently, statistical-relational learning techniques have been used for event recognition. These techniques combine logic with probabilistic models in order to perform reasoning under uncertainty in the presence of complex relational structures. Markov Logic Networks (MLN)s [Domingos and Lowd 2009], a generic statistical relational learning framework that subsumes several probabilistic graphical models, have been attracting attention. For example, Kembhavi et al. [2010] use MLNs to take into account the confidence values of the input events. Morariu and Davis [2011] use MLNs in combination with Allen's interval algebra [Allen 1983] to determine the most consistent sequence of composite events, based on the observations of low-level classifiers. Sadilek and Kautz [2012] propose a method based on hybrid-MLNs [Wang and Domingos 2008] in order to recognise (un)successful human interactions using noisy GPS streams. Skarlatidis et al. [2014] express the event calculus [Kowalski and Sergot 1986] in MLNs to perform event recognition.

Although there is considerable work on optimising probabilistic reasoning techniques, the imposed overhead does not allow for real-time performance in a wide range of applications. This is then a key challenge of event recognition in uncertain environments.

2.3. Distributed Event Recognition

The volume and velocity of event streams is continuously growing and the increasing scale at which events are required to be processed poses challenges both in terms of computational resources and in terms of communication resources. Computational scalability issues are addressed by distributing event recognition tasks among multiple nodes, while communication scalability issues are addressed by algorithms that perform as much of the processing as possible on the event sources, thus minimising the amount of information transferred between nodes (a discussion about these issues may be found in Artikis et al. [2014a]).

Several approaches have been proposed for distributing an event recognition task among multiple nodes. Brenna et al. [2009], for example, evaluate empirically strategies for distributing the execution of automata expressing queries in the Cayuga event language [Brenna et al. 2007]. Semantic dependencies between event queries are used by Lakshmanan et al. [2009] to identify strata of independent queries that are deployed on different (sets of) processing units. Then profiling is applied to guide the assignment of the processing units in a stratum to the respective queries in order to maximise throughput. Schultz-Møller et al. [2009] employ query rewriting for efficient query execution and distribute the recognition process to enable the system to scale to the rate of incoming events. Queries are compiled into detection automata, and the system deploys new automata in a greedy manner. Balkesen et al. [2013] propose a method for distributing input events that belong to individual run instances of the finite state machine of an event pattern to different processing units, thereby providing fine-grained partitioned data parallelism [Hirzel 2012] that is independent from the event pattern.

As mentioned in Section 2.2, the task of event recognition is inherently uncertain and, therefore, deterministic techniques are often unsuitable. The challenge thus is to develop methods for distributing *probabilistic* event recognition tasks, which are fundamentally different from deterministic event recognition tasks.

In addition to the computational scalability issues just discussed, the increasing distribution of event sources requires that network resources are utilised efficiently. Managing bandwidth usage is also a key requirement for mobility-aware event recognition where the events of interest are detected in sensor networks including mobile devices such as smart phones and tablets. For instance, it is necessary to deal with highly dynamic event consumers whose interests change with their location. Since communication efficiency reduces the volume of data sent to a data centre for processing, it also supports computational efficiency. Moreover, communication efficiency supports the privacy of event sources.

Communication-efficient distributed recognition has been an active research field. The proposed methods minimise communication by decomposing the event recognition task into a set of local constraints on the data generated at the event sources. The constraints are such that as long as all of them are satisfied, it is guaranteed that the event of interest has not occurred. Therefore, as long as all constraints are satisfied, no communication is required. The event to be recognised is typically defined using a function over aggregate values derived at the event sources. Research on recognising such types of event includes sketching [Papapetrou et al. 2012], in which data summaries are sent, thus reducing communication overhead and geometric methods for expressing constraints at the event sources [Sharfman et al. 2006; Giatrakos et al. 2014; Keren et al. 2014].

Most of the communication minimisation literature has focused on events defined as functions over aggregate values. Event recognition also requires matching logical, temporal, and spatial event patterns. A key challenge thus lies in developing communication minimisation techniques covering all types of event patterns.

2.4. Event Pattern Learning

Machine learning techniques may be used for constructing or adapting event patterns in a dynamic environment. Both supervised and unsupervised techniques have been employed to automatically adapt and construct event patterns. A widely used unsupervised learning technique is the frequency-based analysis of sequences of events (e.g., Yu et al. [2004], Lee and Lee [2005], Vautier et al. [2007], Álvarez et al. [2010], and Calders et al. [2014]). Frequency-based analysis is a promising approach for discovering unknown events in databases or logs, but it is limited to propositional learning. Moreover, this technique may not be adapted to learning the structure of patterns of composite events that are not frequent in data. In some applications, including credit-card fraud management, the recognition of such events (fraud) is of utmost importance.

A common supervised learning technique for constructing composite event patterns concerns the use of Inductive Logic Programming (ILP) [Muggleton and Raedt 1994] (e.g., Callens et al. [2008]). ILP may construct patterns that capture exceptional cases in an event stream, but does not handle numerical reasoning, which is crucial in the representation of event patterns. When supervision is partial, ILP may be combined with abduction [Denecker and Kakas 2002] in order to learn an event pattern [Ray 2009; Corapi et al. 2011; Athakravi et al. 2013]. This combination of techniques, however, does not scale to the volume and velocity of event streams observed in practice.

In addition to learning the structure of an event pattern, the weight (confidence value) of the pattern can be learned from data [Domingos and Lowd 2009]. The tasks of structure learning and weight learning are often separated. First, the structure of an event pattern is constructed; then the weight of the pattern is learned. Separating these learning tasks, however, may lead to suboptimal results, as structure learning needs to make assumptions about the weights, which have not yet been estimated.

2.5. Event Forecasting

Rapid social, economic, and political changes are leading organisations to shift their thinking from reactive to proactive in order to detect opportunities and threats that could affect their business [Burton et al. 2010]. Changing traffic light policies and speed limits to avoid forecast traffic congestions, for instance, will reduce carbon emissions and optimise public transportation. In energy management, there is a need for real-time optimisation of power consumption in households equipped with renewable energy sources. This requirement may be addressed by forecasting energy consumption and production and making decisions about load adjustments.

Proactive event-driven computing systems exhibit the ability to eliminate or mitigate anticipated problems or to capitalise on forecast opportunities by event forecasting and decision-making [Engel and Etzion 2011; Engel et al. 2012; Feldman et al. 2013]. Event forecasting may be achieved by *forward* event recognition—the ability of an event processing system to recognise events incrementally. The system reports partially recognised events, that is, events for which a subset of the constraints expressing the event pattern are satisfied. Such events may or may not be completely recognised in the future.

Perhaps the most successful system for forward event recognition is the Chronicle Recognition System (CRS) [Dousson and Maigat 2007]. CRS has proven to be very efficient and scalable. It is a purely temporal reasoning system, however, and therefore, cannot be directly applied to any application requiring spatial reasoning. Furthermore, CRS does not deal with uncertainty. Consequently, CRS is insufficient for applications that lack veracity. Note also that event forecasting needs to indicate the probability of a forecast event, as well as the probability of when an event will happen—a probability distribution over the event occurrence time must be provided.

3. IN THIS SPECIAL ISSUE

This section briefly introduces each of the six articles that were selected for this special issue.

3.1. Approximate Semantic Matching of Events for The Internet of Things

Souleiman Hasan and Edward Curry propose an approach for deriving semantic similarity and relatedness of events that do not require a prior agreement between event providers and event consumers on the schema of events. The article is motivated by the increasing number of events created through the Internet of Things. In the absence of an agreement on event schema, participants may loosely agree on topics represented in large corpora of texts, which, in turn, are used for approximate event matching. The authors tackle the challenges of event recognition under uncertainty, using top-K matchers along with a probability model and event pattern learning. The article offers a formal framework and empirical validation for time efficiency, and presents insights on the effect of the degree of approximation on the model.

3.2. PADUA: Parallel Architecture to Detect Unexplained Activities

Cristian Molinaro, Vincenzo Moscato, Antonio Picariello, Andrea Pugliese, Antonino Rullo, and V. S. Subrahmanian present an approach for identifying situations that cannot be satisfactorily explained by any of the known event patterns. Their motivation stems from the fact that, in many applications, such as public space surveillance, credit card transactions, and computer networks, there is a need to recognise irregularities, that is, behaviours for which there is no known pattern. Molinaro et al. first propose probabilistic penalty graphs, an extension of stochastic automata, to deal with the inherent uncertainty of event recognition. Then they propose parallel coordination algorithms for distributed event recognition using probabilistic penalty graphs. The empirical evaluation consists of recognition over 160 CPUs using real-world datasets from video surveillance and network traffic.

3.3. Adaptive Speculative Processing of Out-of-Order Event Streams

Christopher Mutschler and Michael Phlippsen propose an adaptive speculative processing technique for out-of-order event streams in distributed event recognition. Out-of-order event arrival has been handled by using buffering techniques to delay events. This work combines buffering with speculative processing, and adapts the degree of speculation at runtime to fit the available system resources so that detection latency becomes minimal. The empirical evaluation shows improvement over existing approaches both on synthetic data and real data from a real-time locating system with several thousands of out-of-order sensor events per second.

3.4. Decentralised Fault-Tolerant Event Correlation

Gregory Aaron Wilkin, Patrick Eugster, and K. R. Jayaram argue that many use cases for event recognition require fault-tolerant processing in terms of guarantees of event delivery. Providing these guarantees in a decentralised infrastructure is challenging though. The article addresses this issue of distributed event recognition by means of FAIDECS, a model and system that provides guarantees of event delivery. In particular, the authors present a formal analysis of the interplay of the choices of event matching and disposal semantics and the guarantees on safety, liveness, agreement, and order properties of event recognition. In addition, the article links these findings to the StreamSQL, EQL, CEL, and TESLA languages, making it possible to assess the guarantees provided by these languages. Finally, FAIDECS is experimentally shown to scale better than centralised processing or existing solutions for decentralised, fault-tolerant processing.

3.5. MCEP: A Mobility-Aware Complex Event Processing System

Beate Ottenwälder, Boris Koldehofe, Kurt Rothermel, Kirak Hong, David Lillethun, and Umakishore Ramachandran present a middleware that efficiently handles highly dynamic mobile consumers whose interests change with their location. The motivation of this work stems from the proliferation of mobile devices and sensors that offer increasing amounts of information. The proposed distributed Mobile Complex Event Processing (MCEP) system automatically adapts the processing of events according to a consumer's location. MCEP reduces latency, network utilisation, and processing overhead by providing on-demand and opportunistic adaptation algorithms to dynamically assign event streams and computing resources to the MCEP operators. MCEP is evaluated on traffic accident detection, video friend finder, and vehicle speed detection.

3.6. Efficient Stream Provenance via Operator Instrumentation

Boris Glavic, Kyumars Sheykh Esmaili, Peter M. Fischer, and Nesime Tatbul introduce an approach that uses operator instrumentation, that is, modification of the behaviour of operators, to generate and propagate fine-grained provenance through several operators of a query network. The motivation of this work stems from the diagnostic and assurance requirements of a wide range of applications. In addition to computing provenance eagerly during query execution the authors study how to decouple provenance computation from query processing in order to reduce runtime overhead and avoid unnecessary provenance retrieval. Ariadne, the provenance-aware extension of the Borealis stream management system, implements the proposed techniques and manages provenance with minor overhead.

4. CONCLUSIONS

The new era of big data brings with it opportunities with the potential for advancing world technology to new heights. One can envision a world where one wakes up in a smart home, networked with green technology, dresses up in clothes that can monitor day-to-day health, and uses smartly monitored public transportation to get safely and quickly to work or a leisure place. Growing older has the promise of careful health monitoring, improved medical devices, avoiding helpless, embarrassing last years on earth. This special issue provides a few small steps in acquiring the technology for this world and unlocks the potential of big data, which undoubtedly awaits us.

ACKNOWLEDGMENTS

We have benefited from the description of the event recognition challenges of Elias Alevizos.

REFERENCES

J. Allen. 1983. Maintaining knowledge about temporal intervals. *Comm. ACM* 26, 11 (1983), 832–843.

M. R. Álvarez, P. Félix, P. Cariñena, and A. Otero. 2010. A data mining algorithm for inducing temporal constraint networks. In *Proceedings of the International Conference on Information Processing and Management of Uncertainty (IPMU)*. 300–309.

A. Artikis, C. Baber, P. Bizarro, C. C. de Wit, O. Etzion, F. Fournier, P. Goulart, A. Howes, J. Lygeros, G. Paliouras, A. Schuster, and I. Sharfman. 2014a. Scalable proactive event-driven decision-making. *IEEE Technology and Society Mag*.

A. Artikis, M. Weidlich, F. Schnitzler, I. Boutsis, T. Liebig, N. Piatkowski, C. Bockermann, K. Morik, V. Kalogeraki, J. Marecek, A. Gal, S. Mannor, D. Gunopulos, and D. Kinane. 2014b. Heterogeneous stream processing and crowdsourcing for urban traffic management. In *Proceedings of the International Conference on Extending Database Technology (EDBT)*. 712–723.

D. Athakravi, D. Corapi, K. Broda, and A. Russo. 2013. Learning through hypothesis refinement using answer set programming. In *Proceedings of the International Conference of Inductive Logic Programming (ILP)*.

J. Bacon, A. I. Bejan, A. R. Beresford, D. Evans, R. J. Gibbens, and K. Moody. 2011. Using real-time road traffic data to evaluate congestion. In *Dependable and Historic Computing*, 93–117.

C. Balkesen, N. Dindar, M. Wetter, and N. Tatbul. 2013. RIP: run-based intra-query parallelism for scalable complex event processing. In *Proceedings of the International Conference on Distributed Event-Based Systems (DEBS)*. 3–14.

W. Brendel, A. Fern, and S. Todorovic. 2011. Probabilistic event logic for interval-based event recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3329–3336.

L. Brenna, A. J. Demers, J. Gehrke, M. Hong, J. Ossher, B. Panda, M. Riedewald, M. Thatte, and W. M. White. 2007. Cayuga: a high-performance event processing engine. In *Proceedings of the ACM SIGMOD Conference*. 1100–1102.

L. Brenna, J. Gehrke, M. Hong, and D. Johansen. 2009. Distributed event stream processing with non-deterministic finite automata. In *Proceedings of the International and Conference on Distributed Event-Based Systems (DEBS)*.

B. Burton, Y. Genovese, N. Rayner, R. Casonato, M. Smith, M. A. Beyer, T. Austin, B. Gassman, and D. Sommer. 2010. Pattern-based strategy technologies and business practices gain momentum. Gartner Report G00208030.

T. Calders, N. Dexters, J. J. M. Gillis, and B. Goethals. 2014. Mining frequent itemsets in a stream. *Inf. Syst.* 39, 233–255.

L. Callens, G. Carrault, M.-O. Cordier, É. Fromont, F. Portet, and R. Quiniou. 2008. Intelligent adaptive monitoring for cardiac surveillance. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*. 653–657.

H. Chaudet. 2006. Extending the event calculus for tracking epidemic spread. *Art. Intell. Medicine* 38, 2.

D. Corapi, A. Russo, and E. Lupu. 2011. Inductive logic programming in answer set programming. In *Proceedings of the International Conference on Inductive Logic Programming (ILP)*. 91–97.

G. Cugola and A. Margara. 2012. Processing flows of information: From data stream to complex event processing. *ACM Comput. Surv.* 44, 3, 15.

M. Denecker and A. Kakas. 2002. Abduction in logic programming. In *Computational Logic: Logic Programming and Beyond*, A. Kakas and F. Sadri, Eds., Lecture Notes in Computer Science, vol. 2407, Springer, 99–134.

N. Dindar, P. M. Fischer, and N. Tatbul. 2011. DejaVu: a complex event processing system for pattern matching over live and historical data streams. In *Proceedings of the Distributed Event-Based Systems (DEBS)*. 399–400.

P. Domingos and D. Lowd. 2009. *Markov Logic: An Interface Layer for Artificial Intelligence*. Morgan & Claypool Publishers.

C. Dousson and P. L. Maigat. 2007. Chronicle recognition improvement using temporal focusing and hierarchisation. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 324–329.

Y. Engel and O. Etzion. 2011. Towards proactive event-driven computing. In *Proceedings of the Distributed Event-Based Systems (DEBS)*. 125–136.

Y. Engel, O. Etzion, and Z. Feldman. 2012. A basic model for proactive event-driven computing. In *Proceedings of the Distributed Event-Based Systems (DEBS)*. 107–118.

Z. Feldman, F. Fournier, R. Franklin, and A. Metzger. 2013. Proactive event processing in action: a case study on the proactive management of transport processes (industry article). In *Proceedings of the Distributed Event-Based Systems (DEBS)*. 97–106.

A. Gal, S. Wasserkrug, and O. Etzion. 2011. Event Processing over uncertain data. In *Reasoning in Event-Based Distributed Systems*, S. Helmer, A. Poulovassilis, and F. Xhafa, Eds., Springer, 279–304.

N. Giatrakos, A. Deligiannakis, M. Garofalakis, I. Sharfman, and A. Schuster. 2014. Distributed geometric query monitoring using prediction models. *ACM Trans. Database Syst.*

M. Hirzel. 2012. Partition and compose: parallel complex event processing. In *Proceedings of the Distributed Event-Based Systems (DEBS)*. 191–200.

S. Hongeng and R. Nevatia. 2003. Large-scale event detection using semi-hidden markov models. In *Proceedings of the International Conference on Computer Vision (ICCV)*. 1455–1462.

A. Kembhavi, T. Yeh, and L. S. Davis. 2010. Why did the person cross the road (there)? scene understanding using probabilistic logic models and common sense reasoning. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 2, 693–706.

D. Keren, G. Sagy, A. Abboud, D. Ben-David, A. Schuster, I. Sharfman, and A. Deligiannakis. 2014. Geometric monitoring of heterogeneous streams. *IEEE Trans. Knowl. Data Eng.*

R. Kowalski and M. Sergot. 1986. A Logic-based calculus of events. *New Generation Comput.* 4, 1, 67–96.

G. T. Lakshmanan, Y. G. Rabinovich, and O. Etzion. 2009. A stratified approach for supporting high throughput event processing applications. In *Proceedings of the Distributed Event-Based Systems (DEBS)*, A. S. Gokhale and D. C. Schmidt, Eds., ACM.

D. Lee and W. Lee. 2005. Finding maximal frequent itemsets over online data streams adaptively. In *Proceedings of the International Conference on Data Mining (ICDM)*. IEEE Computer Society, 266–273.

J. Lijffijt, P. Papapetrou, and K. Puolamäki. 2012. Size matters: Finding the most informative set of window lengths. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD).* 2, 451–466.

D. Luckham. 2002. *The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems*. Addison-Wesley.

D. Maier, M. Grossniklaus, S. Moorthy, and K. Tufte. 2012. Capturing episodes: may the frame be with you. In *Proceedings of the Distributed Event-Based Systems (DEBS)*. 1–11.

J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers. 2011. Big data: The next frontier for innovation, competition, and productivity.

V. I. Morariu and L. S. Davis. 2011. Multi-agent event recognition in structured scenarios. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3289–3296.

S. Muggleton and L. D. Raedt. 1994. Inductive Logic Programming: Theory and Methods. *J. Logic Program.* 19/20, 629–679.

P. Natarajan and R. Nevatia. 2007. Hierarchical multi-channel hidden semi markov models. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 2562–2567.

O. Papapetrou, M. N. Garofalakis, and A. Deligiannakis. 2012. Sketch-based querying of distributed sliding-window data streams. *Proc. VLDB Endow.* 5, 10, 992–1003.

K. Patroumpas. 2013. Multi-scale window specification over streaming trajectories. *J. Spatial Info. Science* 7, 1, 45–75.

O. Ray. 2009. Nonmonotonic abductive inductive learning. *J. Appl. Logic* 7, 3, 329–340.

A. Sadilek and H. A. Kautz. 2012. Location-based reasoning about complex multi-agent behavior. *J. Artif. Intell. Res.* 43 (2012), 87–133.

N. Poul Schultz-Møller, M. Migliavacca, and P. R. Pietzuch. 2009. Distributed complex event processing with query rewriting. In *Proceedings of the Distributed Event-Based Systems (DEBS)*.

I. Sharfman, A. Schuster, and D. Keren. 2006. A geometric approach to monitoring threshold functions over distributed data streams. In *Proceedings of the ACM SIGMOD Conference*. 301–312.

A. Skarlatidis, G. Paliouras, A. Artikis, and G. Vouros. 2014. Probabilistic event calculus for event recognition. *ACM Trans. Computat. Logic*. (Preprint available from http://arxiv.org/abs/1207.3270.)

A. Vautier, M.-O. Cordier, and R. Quiniou. 2007. Towards data mining without information on knowledge structure. In *Proceedings of the Principles and Practice of Knowledge Discovery in Databases (PKDD)*. 300–311.

U. Vespier, S. Nijssen, and A. J. Knobbe. 2013. Mining characteristic multi-scale motifs in sensor-based time series. In *Proceedings of the Conference on Knowledge Management (CIKM)*. 2393–2398.

D. Vesset, M. Flemming, and M. Shirer. 2011. Worldwide decision management software 2010–2014 forecast: A fast-growing opportunity to drive the intelligent economy. IDC report 226244.

J. Wang and P. Domingos. 2008. Hybrid markov logic networks. In *Proceedings of the AAAI Conference (AAAI)*. 1106–1111.

S. Wasserkrug, A. Gal, O. Etzion, and Y. Turchin. 2012. Efficient processing of uncertain events in rule-based systems. *IEEE Trans. Knowl. Data Eng.* 24, 1, 45–58.

J. Xu Yu, Z. Chong, H. Lu, and A. Zhou. 2004. False positive or false negative: Mining frequent itemsets from high speed transactional data streams. In *Proceedings of the International Conference on Very Large Databases (VLDB)*, Mario A. Nascimento, M. Tamer Özsu, Donald Kossmann, Renée J. Miller, José A. Blakeley, and K. Bernhard Schiefer, Eds., Morgan Kaufmann, 204–215.