High-Level Situation Recognition Using Fuzzy Metric Temporal Logic, Case Studies in Surveillance and Smart Environments

David Münch
Fraunhofer IOSB
Gutleuthausstraße 1, 76275 Ettlingen
david.muench@iosb.fraunhofer.de

Joris IJsselmuiden
Fraunhofer IOSB
Fraunhoferstraße 1, 76131 Karlsruhe
joris.ijsselmuiden@iosb.fraunhofer.de

Michael Arens
Fraunhofer IOSB
Gutleuthausstraße 1, 76275 Ettlingen
michael.arens@iosb.fraunhofer.de

Rainer Stiefelhagen
Karlsruhe Institute of Technology /
Fraunhofer IOSB
Adenauerring 2, 76131 Karlsruhe
rainer.stiefelhagen@kit.edu

Abstract

Although computer vision and other machine perception have made great progress in recent years, corresponding high-level components have not progressed that fast. We present a general purpose framework for high-level situation recognition that is suited for arbitrary application domains and sensor setups. Our approach is hierarchical as opposed to monolithic and we focus on modeling expert knowledge with Fuzzy Metric Temporal Logic and Situation Graph Trees rather than learning from training data. To demonstrate the power and flexibility of our approach, we present case studies in two different settings: guiding the operator's attention in video surveillance and automatic report generation in smart environments. Our results show that this approach can yield a conceptually exhaustive situation recognition for diverse input modalities and application domains.

1. Introduction

As the power and flexibility of computer vision and other machine perception increases, so does the need for high-level components that fuse multiple modalities into a single world model. We aim to bridge the gap between outputs from machine perception and a semantic understanding of the observed scene. First, the collection of low-level information provided by perception is converted into atomic predicates. Then, these predicates are fed into a reasoning engine containing the domain knowledge to deduce corresponding facts on a semantic level. Our system can handle arbitrary application domains and sensor setups, two of which are presented as case studies in this paper.

The goal in the surveillance case study is to guide an operator's focus of attention. Predefined situations should be highlighted in large amounts of video data in order to reduce the operator's workload. The experiments for this case study are based on the well-established CAVIAR dataset [1]. It contains situations with annotated ground truth for people walking alone, meeting with others, window shopping, entering and exiting shops, fighting, passing out, and leaving a bag in a public place. Figure 1, Figure 4, and Table 1 correspond to the `lbpugt` sequence from CAVIAR where somebody leaves a bag and picks it up later. Figure 5, Figure 3, and Table 2 correspond to the `ms3ggt` sequence where three people walk around and meet each other.

The smart environments case study is situated in a control room context. A control room is a place where high-ranking officers work together to lead their forces in managing a crisis situation. Our goal in this case study is to automatically generate reports about the situation in the control room. For the corresponding experiments, we created synthetic data based on a real fire brigade control room exercise. The data consists of the control room interior, person tracks, orientations, gestures, and speech for six officers (see Figure 7). We recognize situations related to the group formations, phases, and activities that are typical to control room operation.

The proposed system for high-level situation recognition can handle other applications and sensor setups as well. Furthermore, our system can provide perception with top-down information and predict situations in the near future. We discuss related work in Section 2 and our methods are described in Section 3. The experiments are discussed in Section 4 and we conclude this contribution in Section 5.

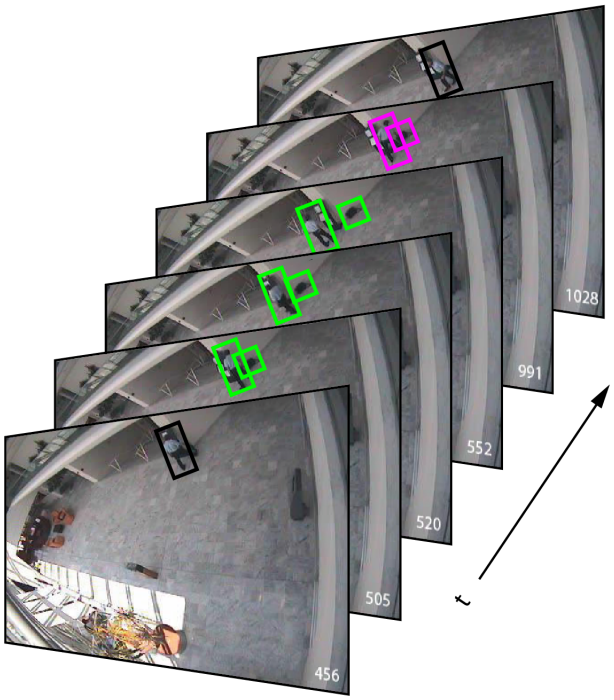

Figure 1. An example sequence from the CAVIAR dataset [1]. A person leaves a bag and walks away (green), upon which our system raises an alarm. Over 400 frames later the person returns and picks up his bag and the alarm is withdrawn (magenta).

2. Related Work

Recent surveys dealing with the high-level recognition of situations in image sequences are [3, 17, 25]. The field can be divided into three types of approaches. (1) Statistical approaches compute the likelihood of a situation given an image sequence by learning graph models (e.g. dynamic Bayesian networks) from labeled training data [2, 7, 8, 21]. (2) Syntactic approaches apply nested production rules as used in formal grammars. Subsequently, they parse the generated situation strings [12, 15]. (3) Description-based approaches are built upon the formulation of temporal and spatial properties of situations [14]. Temporal relations are usually expressed using interval temporal logic [4]. High-level situation recognition was further improved by contributions such as [26], increasing efficiency, and [22], increasing the complexity of recognized situations using hierarchical structures. To deal with uncertainty in input data, [16] and [24] combined logical and probabilistic approaches. They use first order predicate logic with weighted rules as input for a Markov Logic Network. Advanced methods for image and video understanding and the subsequent report generation in natural language are presented in [13] and [27].

Our own approach uses the conceptual description of situations with Fuzzy Metric Temporal Logic (FMTL) and Situation Graph Trees (SGTs) described in [20]. FMTL and SGTs were applied to traffic scenarios in [6, 10] and [11, 9] apply them to human behavior. Our framework offers several advantages when compared to approaches that learn from training data. Formalizing expert knowledge is easy and existing sets of rules can be extended with minimal effort. Furthermore, the reasons for arriving at a certain conclusion are human understandable. And most importantly, no training data is needed. Especially in complex domains, high-dimensional search spaces are difficult or impossible to cover without an enormous amount of training data.

The contribution of this paper consists of several parts. First, we developed a domain independent rule dictionary that can be applied to both the surveillance case study and the smart environment case study. These rules were then combined in two sets of SGTs using temporal and hierarchical composition. Furthermore, we apply a fuzzy exhaustive graph traversal on the SGTs to recognize multiple situations in parallel. Furthermore, our work is not limited to the description of a person's motion behavior. Instead, we have incorporated other input modalities and we will continue to do so in the future. Finally, we will show that our framework can deduce sophisticated situations in various application domains.

3. Methods

Nagel described a generic layered model for cognitive vision systems in [19]. It was modified in [5] to satisfy the needs of multimodal sensors and actuators. Figure 2 depicts our specific instantiation of this model.

The *Interactive Subsystem* (IS) encompasses the *Sensor Actuator Level* (SAL) which can contain data in various modalities. Sensors can be cameras, microphones, haptic sensors, or anything else that is available. Actuators can include displays, speakers, and motors. As video is the dominant source of perception in our research, we explain this layered model for the vision case only. When using audio and other signals, the model can be extended accordingly.

The image signal information from SAL is passed to the *Quantitative Layer* (QL). It is comprised of the *Image Signal Level* (ISL), the *Picture Domain Level* (PDL), and the *Scene Domain Level* (SDL). In the ISL, video information is encoded in pixels. Adding information about local features, edges, or any other construct that uses more than one pixel leads to the *Picture Domain Level* (PDL). It allows tasks such as person detection and object detection. In the subsequent *Scene Domain Level* (SDL), information such as person tracks, orientations, and gestures is stored. Additionally, scene knowledge is added in this level, e.g. scene geometry. This information can be either generated automatically or added manually.

Up to this point all information is of a quantitative na-

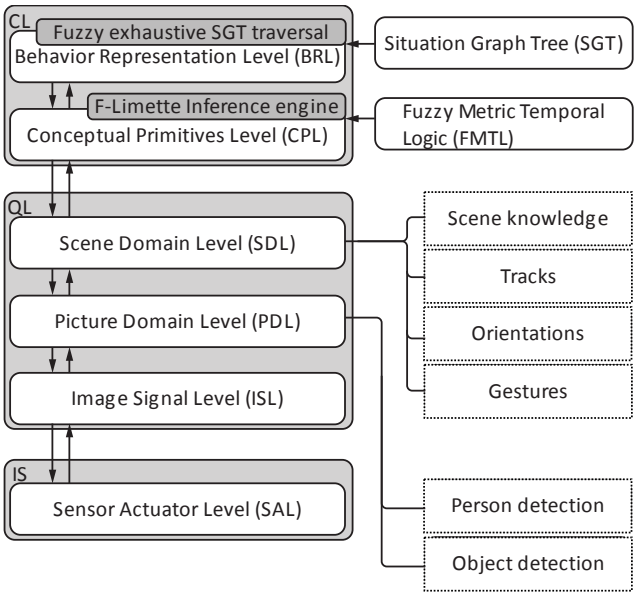

Figure 2. Cognitive vision system adapted from [5, 19]. The Interactive Subsystem (IS) stores the information from sensors and actuators, the Quantitative Layer (QL) consists of 1D pixel information (ISL), 2D image information (PDL), and 3D scene geometry information (SDL). The dotted boxes represent the perceptual modalities that are currently used. The Conceptual Layer (CL) stores conceptual knowledge as FMTL rules and SGTs.

ture. In order to reach a semantic understanding from these large amounts of noisy quantitative information, the system needs formalized expert knowledge about the domain in question. In the *Conceptual Layer* (CL), this knowledge is split up into primitive knowledge and high-level knowledge. The primitive knowledge is represented as a set of domain independent Fuzzy Metric Temporal Logic (FMTL) rules in the *Conceptual Primitives Level* (CPL). It contains rules about spatiotemporal relations in the physical world. Finally, the *Behavior Representation Level* (BRL) contains high-level knowledge represented as Situation Graph Trees (SGT). These SGTs employ the domain independent FMTL rules to recognize high-level situations. The SGTs are domain specific because they require conceptual, abstract interpretations that go beyond the spatiotemporal physical domain.

Each level of the model is bidirectionally connected to the next. Typically, most information flows bottom-up, but there are also cases where the top-down direction plays an important role. A good example is the guided deployment of sensors and other resources to improve coverage of interesting situations. We continue with a detailed description of the *Conceptual Layer* (CL).

3.1. Conceptual Primitives Level

On the transition from SDL into CPL, quantitative information is transformed into primitive conceptual knowledge. The fuzziness that is involved here exists in two forms: vagueness and uncertainty. Vagueness means that number intervals cannot be mapped directly to concepts as concepts are often vague. There is no clear distinction between moving fast and moving very fast for example, the transition is smooth. Uncertainty in this context typically stems from the lower levels supplying uncertain information. If the lower levels supply confidence values, the higher levels should be able to handle these appropriately in their reasoning. Both vagueness and uncertainty are expressed by a number between 0 and 1. To turn low-level confidence values and high-level conceptual vagueness into combined truth values, several s-norms and t-norms are available: Zadeh, Lukasiewicz, Product, and Gödel, among others [18]. FMTL is a temporal predicate logic with fuzzy truth values [10] that can handle vagueness and uncertainty in combined truth values.

The inference on FMTL predicates is performed with the inference engine F-LIMETTE [23]. We developed a rule dictionary for our case studies in surveillance and smart environments. Many rules in this dictionary are used in both case studies, demonstrating rule reusability. Other rules have to be adapted to the given domain. Two example rules are shown below. The $\Box$ is the *always* operator and the other operators have the conventional meaning from first order logic.

$$\begin{aligned}\Box \{ [\Diamond_{-2}\ & Distance(p,q,d_{-2}) \wedge \\ \Diamond_{-1}\ & Distance(p,q,d_{-1}) \wedge \\ & Distance(p,q,d_0) \wedge \\ \Diamond_{1}\ & Distance(p,q,d_1) \wedge \\ \Diamond_{2}\ & Distance(p,q,d_2) \wedge \\ & Derivative(d_{-2},d_{-1},d_0,d_1,d_2,d') \wedge \\ & DistanceChangeCategory(d',c)] \rightarrow \\ & DistanceChange(p,q,c) \}\end{aligned}$$

$$\begin{aligned}\Box \{ \forall\, p\, [\, & HasType(p,agent) \rightarrow \\ \exists\, q\, [\, & HasType(q,chair) \wedge AtSeat(p,q) \rightarrow \\ & EverybodyAtSeat()]] \}\end{aligned}$$

The first rule infers $DistanceChange(p,q,c)$, telling us how the distance is changing between two agents or objects. The rule requests the Euclidian distance between p and q for the surrounding time interval from $t = -2$ to $t = 2$ (indicated by $\Diamond_{-2}$ through $\Diamond_2$). Then, the derivative d' of these five distances is calculated and fuzzily mapped to the values for c: *decreasing*, *constant*, *increasing*. The mapping is performed by three displaced trapezoidal truth functions with

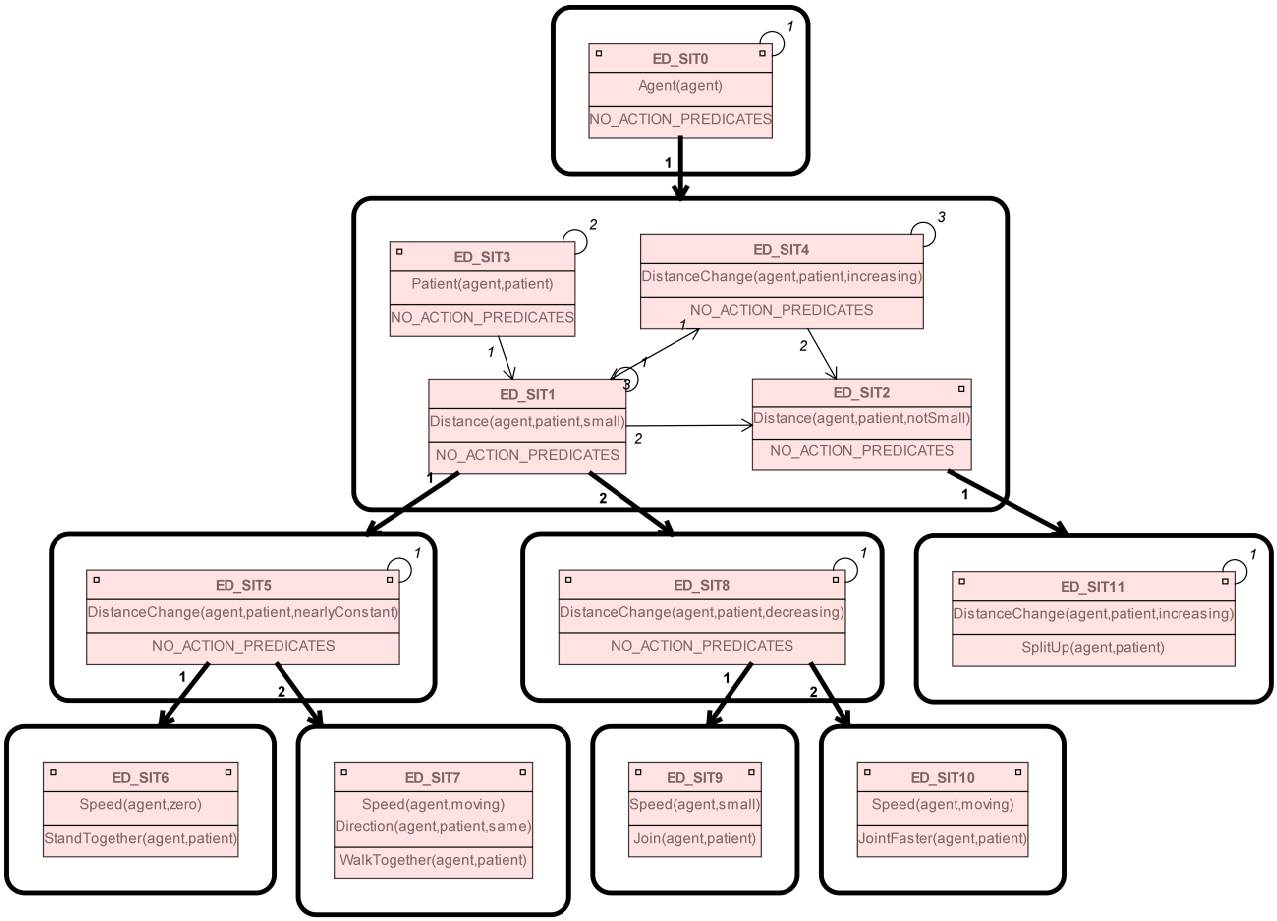

Figure 3. Part of the Situation Graph Tree (SGT) used in the surveillance scenario described in Section 4.1. An SGT represents the expert knowledge about the situations to be detected. In this case $WalkTogether$, $StandTogether$, $SplitUp$, $JoinFaster$, and $Join$. Each box represents a situation scheme that can be specialized conceptually (thick edges) and temporally (thin edges). Boxes at the top left and top right of a situation scheme indicate that it is a start situation scheme and end situation scheme respectively. The numbers on the edges assign their priority for the traversal.

the one for *constant* centered around 0. The second rule infers $EverybodyAtSeat()$. Its condition $AtSeat(p, q)$ also uses a trapezoidal truth function so that the rule also fires if an agent is close to a chair. The rules in the CPL are mostly concerned with spatial relations and temporal relations on short time intervals.

3.2. Behavior Representation Level

The BRL is the highest level in the layered model in Figure 2. The Situation Graph Trees (SGTs) used in this level can be edited with the graphical user interface described in [5]. This tool also performs the fuzzy exhaustive graph traversal described in Section 3.2.2

3.2.1 Situation Graph Trees

At this level the primitive logic predicates from the CPL are aggregated and structured in SGTs to model high-level conceptual situations. Figure 3 depicts an SGT which represents the expected behavior of agents. It consists of situation schemes which can be start and/or end nodes in the SGT. Each situation scheme has a unique name, a state scheme and an action scheme. The state scheme consists of state predicates in Fuzzy Metric Temporal Logic (FMTL) as described in Section 3.1 which is used as a precondition for the instantiation of the given situation scheme. If a situation has been instantiated for a certain agent or object, the corresponding action scheme is instantiated. In SGTs it is possible to specialize each situation in a conceptual and temporal manner. Prediction edges are used to link a situation with

a possible subsequent situation to model knowledge about possible temporal developments of situations. Specialization edges connect more general situations to more specific ones in a hierarchical structure. For conceptual specialization additional state predicates are added to the more specific situation schemes whereas for temporal specialization the situation is broken up into several situation schemes.

3.2.2 Exhaustive Situation Analysis

A situation graph traversal algorithm performs the situation analysis. Until now the algorithm has found only one instantiation of a situation for each agent or object at each point in time [6, 11]. Considering the fact that several situations can be an adequate description simultaneously and that the uncertainty from lower levels requires multiple hypotheses, this algorithm is insufficient for our purpose. Therefore, we extended it to a fuzzy exhaustive graph traversal. It starts with the instantiation of a new agent or object in the first start situation scheme of the root graph. In case the state scheme of a particular situation scheme can be instantiated, the algorithm traverses along its specialization edges. If a new start situation scheme is found the traversal continues recursively from there. If this fails, the algorithm tries to proceed at one time-step ahead while crossing prediction edges to instantiate situation schemes until an end situation is reached. After having finished the traversal of the specialization it continues in the more general situation until the end situation scheme in the root graph is reached. The traversal algorithm follows all existing alternatives in specialization and temporal development meaning that it is concurrently considering different situation schemes with different instantiations. This allows a fuzzy exhaustive recognition of situations and their different instantiations.

4. Experiments

We present experimental results from two different case studies. Section 4.1 describes our experiments on video surveillance data and Section 4.2 describes the smart environments case study.

4.1. Video Surveillance

For evaluating video sequences in surveillance applications the well-known CAVIAR dataset is chosen [1]. The scenes of this dataset are challenging, there are multiple people and objects involved, and a variety of different actions is performed: walking alone, meeting with others, window shopping, entering and exiting shops, fighting, passing out, and leaving a bag in a public place. The hand-labeled ground truth annotation per frame consists of names, positions, orientations, roles, and more for the

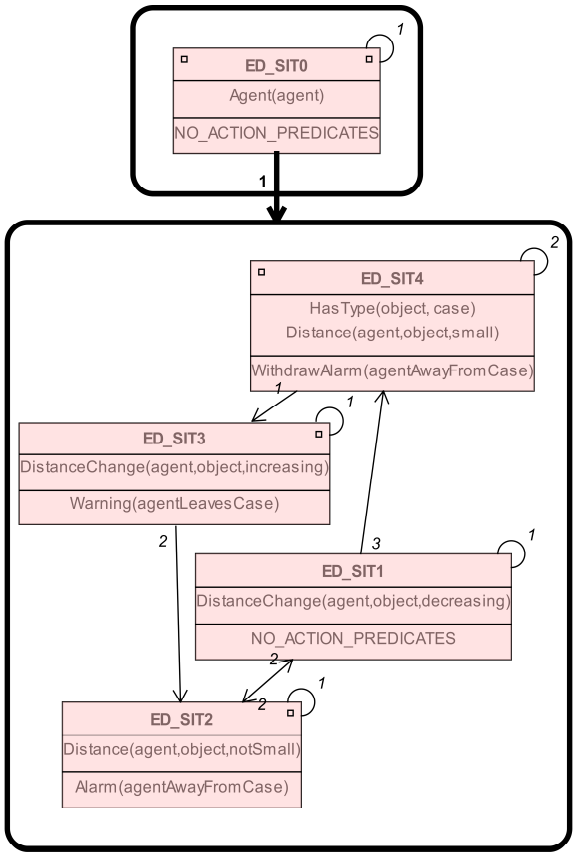

Figure 4. Part of an SGT representing the knowledge about leaving a bag and raising an alarm and picking the bag up later and withdrawing the alarm. The situation scheme ED_SIT0 gets specialized if the distance between the agent and the bag is small.

agents and objects involved. As our aim is to evaluate high-level situation recognition we assume fair results in the QL. This is why the CL only uses the available hand-labeled information of the position of each agent and object and their names.

Our system can recognize situations involving one agent or object (e.g. moving slowly to the right), multiple agents and objects (e.g. picking up a bag or two people meeting), and situations involving agents, objects, and locations (e.g. leaving a bag in a specific area). Figure 4 shows part of an SGT that can recognize interactions between a person and a bag. ED_SIT4 is instantiated if a person and a bag are detected close to each other. Then, ED_SIT3 raises a warning if the distance between the person and the bag increases. As soon as their distance is $notSmall$ (ED_SIT2), an alarm is raised. And if the person returns to the object (ED_SIT1 to ED_SIT4), the alarm is withdrawn.

Table 1 shows an output snippet for CAVIAR's `lbpugt` sequence (see Figure 1). Our system recognizes a person at

Time	Truth value	Predicate
456	1.000	Agent(a)
504	0.735	HasType(c, bag)
505	0.242	DistanceChange(a, c, increas.)
505	0.242	Distance(a, c, notSmall)
519	0.154	HasType(c, bag)
520	1.000	DistanceChange(a, c, increas.)
520	0.564	Distance(a, c, notSmall)
552	1.000	DistanceChange(a, c, increas.)
552	1.000	Distance(a, c, notSmall)
991	0.627	HasType(c, bag)
1028	1.000	Agent(a)

Table 1. Output and intermediate predicates of the situation recognition corresponding to Figure 1. Note the increasing truth value of *Distance* while the agent leaves the bag.

frame 456 and a bag at frame 504. At 505, the person is leaving the bag, but only with a low truth value. At 552, the truth values increase as the person leaves the bag proper, causing an alarm. And finally, at 991, the alarm is withdrawn as the person picks up his bag again. These results reflect the situations in Figure 1 very well, and we achieve comparable results on other CAVIAR sequences involving luggage.

In the `ms3ggt` sequence of the CAVIAR dataset people are walking together, standing still, splitting up, and joining each other (see Figure 5). Figure 3 depicts part of an SGT which can recognize the behavior of small groups. In this experiment we apply this SGT to the `ms3ggt` sequence and Table 2 and Figure 5 show a meaningful excerpt of the obtained result. At time point 350 person 1 and 2 are walking together, at 390 they are standing together, and at 402 both are splitting up. Then person 3 approaches, at 410 person 1 and 3 are standing together, and at 441 person 1 and 3 are walking together. Our system successfully recognized the high-level situations occurring in the CAVIAR dataset with a few minor errors due to imprecise ground-truth.

4.2. Smart Environments

Collecting the complex experimental data that we need for the smart environments case study is very laborious, and we did not find any suitable existing datasets. Because synthetic data can serve the purpose of enhancing high-level algorithms very well, we decided to generate XML data with a dedicated PyQt tool (see Figure 7). It allows us to generate realistic data with very little effort. In the future, additional challenges will be added by adding interpolation, noise, uncertainty, and a more sophisticated array of perceptual modalities.

The simulation is based on an actual fire brigade control room exercise (see Figure 6). From around twenty participants in the exercise, we modeled the six most prominent roles, each with his own table and chair: commanding officer (CO), messenger (M), units officer (S1), maps officer (S2), strategy officer (S3), and supplies officer (S4). Furthermore, the simulation contains two doors, a message hatch, and two planning boards. The area of the room is scaled down from $8m \cdot 8m$ to $800px \cdot 800px$ and divided into 15 zones.

Person tracks, orientations, gestures, and speech are manipulated through mouse manipulation. We created 360 snapshots this way, corresponding to 30 minutes of data with one snapshot every five seconds. This should be increased to at least one snapshot per second in order to recog-

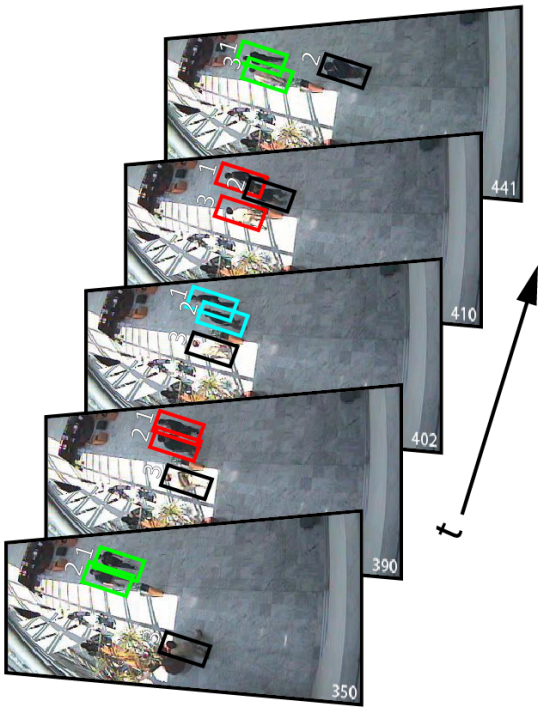

Figure 5. Sequence with multiple people and their recognized behavior (see Table 2). At frame 350 person 1 and 2 are walking together (green), at frame 390 they are standing together (red), at 402 they split up (cyan), at 410 person 3 stands together with person 1 (red) and finally, at 441 person 1 and 3 are walking together (green).

Time	Truth value	Predicate
350	0.550	WalkTogether(a1, a2)
390	0.552	StandTogether(a1, a2)
402	1.000	SplitUp(a1, a2)
410	1.000	StandTogether(a1, a3)
441	1.000	WalkTogether(a1, a3)

Table 2. Output of the situation recognition for the sequence displayed in Figure 5.

nize temporal developments such as accelerating and moving towards an object. From Figure 7, the system recognizes that CO is working individually in the CO zone, S4 and M are having a conversation in the S4 zone, and S1, S2, and S3 are doing teamwork in the map zone (compare to Figure 6). These situations, among others, have been successfully recognized in a running system.

The rule $Teamwork(p, q, r, z)$ is shown below. It looks for occurrences of three people working together in a zone z (e.g. $Teamwork(s1, s2, s3, mapZone)$ in Figure 7). The condition $In(p, z)$ needs to be satisfied for all three agents and $Interacts(p)$ for at least one of them. For $In(p, z)$, p and q get their types checked and e is set to $0.5m$, which should be interpreted as half the width of the zone z's vague border. Then, the position of p and the position and size of z are retrieved. These are used in $IR(m, a, b, c, d)$ which uses a trapezoidal function to determine to what degree x_p and y_p are in the correct range. For x_p (and equivalently for y_p), the truth value rises to 1 between $x_z - e$ and $x_z + e$ and it falls back to 0 between $x_z + w_z - e$ and $x_z + w_z + e$. Effectively, the predicate $In(p, z)$ evaluates to 1 if p is at least $0.5m$ inside the zone z and $In(p, z)$ evaluates to 0 if p is at least $0.5m$ outside z. For $Interacts(p)$, agent p has to be speaking or pointing at least once in the immediate temporal surroundings.

$$\begin{aligned} \Box\ \forall\, p, q, r, z\ \{ \\ [\ In(p,z)\ \wedge\ In(q,z)\ \wedge\ In(r,z)\ \wedge \\ (\ Interacts(p)\ \vee\ Interacts(q)\ \vee\ Interacts(r)\ )\ ]\ \rightarrow \\ Teamwork(p,q,r,z)\ \} \end{aligned}$$

$$\begin{aligned} \Box\ \forall\, p, z\ \{\ Type(p, agent) \wedge Type(z, zone) \wedge e = 0.5\ \wedge \\ Pos(p, x_p, y_p) \wedge Pos(z, x_z, y_z)\ \wedge\ Size(z, w_z, h_z)\ \wedge \\ IR(x_p,\ x_z - e,\ x_z + e,\ x_z + w_z - e,\ x_z + w_z + e)\ \wedge \\ IR(y_p,\ y_z - e,\ y_z + e,\ y_z + h_z - e,\ y_z + h_z + e)\ \rightarrow \\ In(p, z)\ \} \end{aligned}$$

$$\begin{aligned} \Box\ \forall\, p\ \{\ [\ \Diamond_{-1}\ (\ Speaks(p)\ \vee\ Points(p)\ )\ \vee \\ (\ Speaks(p)\ \vee\ Points(p)\ )\ \vee \\ \Diamond_{1}\ (\ Speaks(p)\ \vee\ Points(p)\ )\ ]\ \rightarrow \\ Interacts(p)\ \} \end{aligned}$$

5. Conclusion

We presented our ongoing work on a general purpose framework for high-level situation recognition. To show the power and flexibility of our approach, we discussed two case studies: one for guiding the operator's attention in video surveillance and one for automatic report generation in smart environments. Our framework fuses the output from all available perceptual components into a single world model. Then, low-level data is fed into FMTL rules to deduce spatiotemporal relations. SGTs containing the domain knowledge combine these relations in hierarchical and temporal tree structures to deduce high-level situations.

This work contains the following contributions. We developed a domain independent dictionary of Fuzzy Metric Temporal Logic rules that is applicable to the two cases studies presented in this paper. In the first case study, the goal is to guide the operator's attention to predefined situations in video surveillance. The second case study is concerned with automatic report generation in smart environ-

Figure 6. The smart environments case study is based on an actual fire brigade control room exercise. This particular snapshot shows the following situations: teamwork, one-on-one conversation, and individual work.

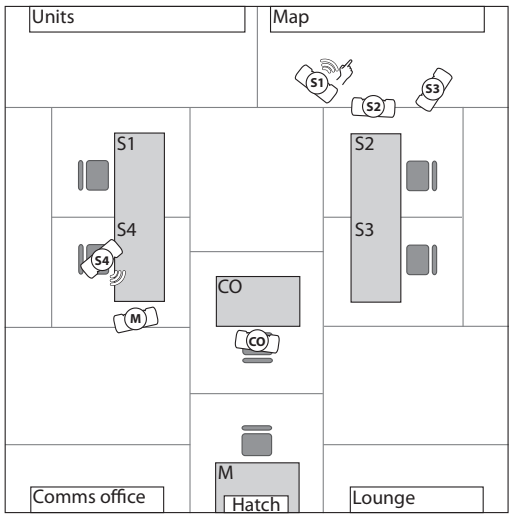

Figure 7. Our simulation tool for generating experimental data in the smart environments case study. Our system can deduce all three activities visible in this snapshot *concurrently*: teamwork at the map display, one-on-one conversations, and individual work.

ments, particularly for control room operations. Furthermore, the situations that are to be recognized in these case studies are modeled in Situation Graph Trees, incorporating the domain independent dictionary of FMTL rules. In our experiments, we have successfully evaluated all examples discussed above. Corresponding quantitative evaluations will be performed in the near future and we will extend our input modalities with 3D pose estimation, moving objects, and display interaction. The joint handling of uncertainty and vagueness will also play an important role in our future work.

Acknowledgments.

This work is supported by the FhG Internal Programs under Grant No. 692 026.

References

[1] EC Funded CAVIAR project / IST 2001 37540, url: http://homepages.inf.ed.ac.uk/rbf/caviar/, 2001. 1, 2, 5

[2] J. K. Aggarwal and S. Park. Human motion: Modeling and recognition of actions and interactions. *3DPFT*, 0:640–647, 2004. 2

[3] J. K. Aggarwal and M. S. Ryoo. Human Activity Analysis: A Review. *ACM Computing Surveys*, 2011. 2

[4] J. Allen and G. Ferguson. Actions and events in interval temporal logic. *Journal of logic and computation*, 4(5):531, 1994. 2

[5] M. Arens. Repräsentation und Nutzung von Verhaltenswissen in der Bildfolgenauswertung, Fakultät für Informatik der Universität Karlsruhe (TH), Dissertationen zur Künstlichen Intelligenz (DISKI) 287, Akademische Verlagsgesellschaft Aka GmbH, 2004. 2, 3, 4

[6] M. Arens, R. Gerber, and H.-H. Nagel. Conceptual representations between video signals and natural language descriptions. *IVC*, 26(1):53–66, 2008. 2, 5

[7] O. Brdiczka, M. Langet, J. Maisonnasse, and J. Crowley. Detecting human behavior models from multimodal observation in a smart home. *IEEE T-ASE*, 6(4):588 –597, oct. 2009. 2

[8] H. Buxton. Learning and understanding dynamic scene activity: a review. *IVC*, 21(1):125 – 136, 2003. 2

[9] C. Fernndez, P. Baiget, X. Roca, and J. Gonzalez. Interpretation of complex situations in a semantic-based surveillance framework. *Signal Processing: Image Communication*, 23(7):554 – 569, 2008. Special Issue on Semantic Analysis for Interactive Multimedia Services. 2

[10] R. Gerber and H.-H. Nagel. Representation of occurrences for road vehicle traffic. *Artificial Intelligence*, 172(4-5):351 – 391, 2008. 2, 3

[11] J. Gonzalez, D. Rowe, J. Varona, and F. X. Roca. Understanding dynamic scenes based on human sequence evaluation. *IVC*, 27(10):1433 – 1444, 2009. Special Section: Computer Vision Methods for Ambient Intelligence. 2, 5

[12] G. Guerra-Filho and Y. Aloimonos. A language for human action. *Computer*, 40(5):42–51, 2007. 2

[13] A. Gupta, P. Srinivasan, J. Shi, and L. Davis. Understanding videos, constructing plots, learning a visually grounded storyline model from annotated videos. In *Proc. CVPR*, pages 2012–2019, 2009. 2

[14] J. IJsselmuiden and R. Stiefelhagen. Towards high-level human activity recognition through computer vision and temporal logic. In *Proceedings of the 33rd Annual German Conference on Advances in Artificial Intelligence*, 2010. 2

[15] Y. Ivanov and A. Bobick. Recognition of visual activities and interactions by stochastic parsing. *PAMI*, 22(8):852–872, 2000. 2

[16] A. Kembhavi, T. Yeh, and L. Davis. Why did the person cross the road (there)? scene understanding using probabilistic logic models and common sense reasoning. In K. Daniilidis, P. Maragos, and N. Paragios, editors, *Proc. ECCV*, volume 6312, pages 693–706. Springer, 2010. 2

[17] G. Lavee, E. Rivlin, and M. Rudzsky. Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video. *IEEE Trans Syst Man Cybern C Appl Rev*, 39(5):489 –504, 2009. 2

[18] T. Lukasiewicz and U. Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):291 – 308, 2008. Semantic Web Challenge 2006/2007. 3

[19] H.-H. Nagel. Image sequence evaluation: 30 years and still going strong. *Pattern Recognition, International Conference on*, 1:1149, 2000. 2, 3

[20] H.-H. Nagel. Steps toward a cognitive vision system. *AI Mag.*, 25(2):31–50, 2004. 2

[21] S. Park and J. Aggarwal. A hierarchical bayesian network for event recognition of human actions and interactions. *Multimedia Systems*, 10(2):164–179, 2004. 2

[22] M. Ryoo and J. Aggarwal. Semantic representation and recognition of continued and recursive human activities. *IJCV*, 82:1–24, 2009. 2

[23] K. H. Schäfer. Unscharfe zeitlogische Modellierung von Situationen und Handlungen in Bildfolgenauswertung und Robotik, Fakultät für Informatik der Universität Karlsruhe (TH), Dissertationen zur Künstlichen Intelligenz (DISKI) 135, Akademische Verlagsgesellschaft Aka GmbH, 1996. 3

[24] S. Tran and L. Davis. Event modeling and recognition using markov logic networks. In D. Forsyth, P. Torr, and A. Zisserman, editors, *Proc. ECCV*, volume 5303, pages 610–623. Springer, 2008. 2

[25] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *CSVT*, 18(11):1473 –1488, 2008. 2

[26] V.-T. Vu, F. Bremond, and M. Thonnat. Automatic video interpretation: a novel algorithm for temporal scenario recognition. In *Proc. IJCAI*, pages 1295–1300, 2003. 2

[27] B. Z. Yao, X. Yang, L. Lin, M. W. Lee, and S.-C. Zhu. I2T: Image Parsing to Text Description. *Proceedings of the IEEE*, 98(8):1485–1508, Aug. 2010. 2