Incorporating Online Process Mining Based on Context Awareness into Human-Robot-Cooperation Framework

Stephan Puls, Benjamin Lotspeich, and Heinz Wörn
Institute of Process Control and Robotics (IPR)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
{puls, lotspeich, woern}@ira.uka.de

***Abstract*—The framework, in which this work is embedded, focuses on the cooperation of humans and robots in productive scenarios. Based on action recognition and situation awareness, conclusions about human behavior can be drawn. The presented work utilizes an online process mining method based on the Heuristic Miner in order to identify a process model regarding context and recognized actions. The process model represents temporal and spatial dependencies of events and is used for detecting recurring behavior sequences.**

Keywords – cognitive robotics; online process mining; heuristic miner; ambient intelligence; context awareness; situation and action recognition; human-robot-cooperation.

I. INTRODUCTION

Assistance through technology can be found in many areas of daily living. Nowadays, many, mostly trivial, work processes are conducted by machinery and robots. Due to their strength, dependability and accuracy, there is a wide variety of uses for robots. Nevertheless, human labor is still more flexible, dynamic and robust due to cognitive abilities to identify processes and errors and react accordingly. Bringing these strengths together in human and robot cooperation is an important research field.

Industrial robotics is a challenging domain for cognitive systems, especially when close interactions between solid machinery and human intelligence is wanted. We are conducting research on recognition of and reasoning about human actions, situations and spatio-temporal context in a human centered productive environment, in order to enable interactive and cooperative scenarios. For this purpose a framework for human-robot-cooperation (MAROCO) was introduced [1, 2], in which human pose reconstruction and context awareness is achieved (see Fig. 1).

This paper focuses on using this extracted information for online process mining based on the Heuristic Miner [3]. Process mining aims at extracting knowledge from event logs consisting of process events and their respective temporal order. Thus, the framework is extended by identifying a sequence model of performed human actions and detecting recurring sequences.

The remainder of this paper is organized as follows. In Section II, some related research work on process mining and sequence detection will be presented. In Section III, the framework will be introduced, which enables the sensor data processing and subsequent knowledge based reasoning. Also the modeled situations, activities and context are explained. In Section IV, the Heuristic Miner will be briefly introduced and the module realizing the online process mining will be presented in detail. Section V discusses experimental results which have been carried out for both, predetermined test cases and under real-life conditions. In Section VI, a summary is given. Finally, some hints for future work are also mentioned.

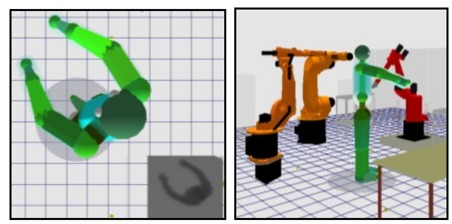


Figure 1. Human kinematics reconstruction based on depth information from time-of-flight camera (left) and the overall scene (right).

II. RELATED WORKS

For identifying work sequences in complex processes, patterns of variable length need to be recognized in a time series of discrete state variations. Many fundamental methods in the area of artificial intelligence are devised to recognize sequences. In this context, Sun and Giles distinguish between recognition and detection of patterns and present some possible solutions [4].

One such solution can be inductive logical programming (ILP). In [5], ILP is used learn and recognize symbols. Based on positive and negative examples, rules are learned, which enable a subsequent classification of images.

In [6], temporal logics are used to detect frequent patterns in temporal databases by using temporal operators and constructing temporal logic programs. These programs were specific to the pattern and the overall system performance was argued to be too slow for practical applications by the authors.

Hidden Markov Models (HMMs) are a widely used means for learning and recognizing temporal patterns (e.g., [7, 8, 9]). For detecting unknown patterns this method cannot be used, due to its supervised learning strategy. That is also the case for the above mentioned procedures. On the contrary, discovering unknown patterns in a time series is an unsupervised task. In [10], an approach is presented which

 ISBN: 978-1-61208-269-1

uses HMMs for the pattern discovery task. The length of the patterns and structure of the HMMs has to be known a priori, which modifies the task of discovery of unknown patterns to a distinguishing task of similar patterns.

The research field of process mining (PM) wants to extract knowledge from event logs, which are comprised of process events including their temporal order. PM is mainly used to identify and optimize business processes. Different methods have been devised to identify processes, their internal structure and to model them in so called Workflow-Nets [11, 12].

The Alpha-Algorithm is one of the first developed methods for process modeling [13]. It models temporal relationships of events based on their occurrence in the event log. In [14], it is shown that the Alpha-Algorithm is prone to incomplete or noisy event logs and produces erroneous process models. The Tsinghua-Alpha-Algorithm is a further development of the Alpha-Algorithm and takes the duration of activities into account and allows the modeling of parallelism [15]. It also requires complete and noise-free data logs [14].

The Heuristic Miner models a process based on frequency and order of events [3]. In contrast to the Alpha and Tsinghua-Alpha Algorithms, it can deal with noisy data sets, which is a prerequisite in sensor based cognitive systems, hence, its adoption in the research framework MAROCO (see Section IV).

The basic methods for process mining mentioned above are offline algorithms due to their main application in analyzing business processes. Thus, data logs are recorded over some period of time and analyzed afterwards. Accordingly, for the use of process mining in human-robot-cooperation scenarios the adaption to online data analysis is necessary.

III. THE MAROCO FRAMEWORK

The process mining presented in this paper is embedded into a general human robot cooperation software framework. In this section, an overview of the framework is given. Also, the generated information, which is processed during mining, is presented.

A. Overview

In our previous research a framework for human-robot-cooperation (MAROCO) was developed [1]. The goal is to realize a comprehensive cognition cycle, which consists of (1) sensing, (2) cognition, and (3) acting. The sensing is done by a photon mixer device (PMD) camera, which captures directly a 3D point cloud of the scene. Through 3D image sequence analysis the work space of an industrial robot is observed and the kinematics of a human worker is reconstructed (see Fig. 1).

The reconstruction of the human kinematics is based purely on depth image analysis and does not require markers attached to the human agent [16]. During the process of reconstruction many parameters of the human kinematics are estimated, e.g., head orientation, upper body orientation, arm configurations, etc. Due to the position of the camera above the work cell some parameters, like leg configurations, cannot be identified.

The camera fixation at the ceiling is mandatory in respect to safety concerns. Dirt, which is inevitable in a production environment, is less prone to stick to the lens system of the camera. Furthermore, manipulation of the camera position is difficult to achieve. Thus, its calibration in the environment is constant. Also, occlusions of the human through movement of the robots and production materials are avoided.

In order to achieve seamless interaction and an intuitive human-machine interface an activity recognition and situation awareness system based on the combination of HMMs and Description Logics (DLs) was implemented [2, 17]. Fast action recognition is implemented based on HMMs, whereas higher cognitive reasoning about situations and activities is accomplished with the slower DL-based reasoning system. Also, due to the sensing of the human kinematics in the scene, a localization module identifies the current semantic position of the human co-worker in the scene. The overall system regarding action recognition and situation awareness is depicted in Fig. 2.

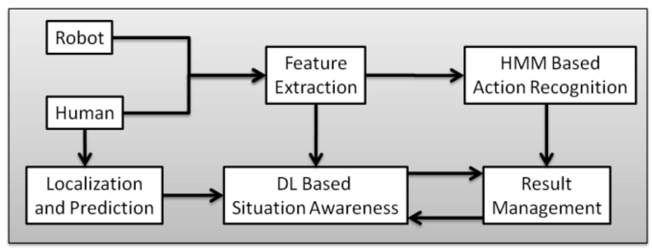


Figure 2. System overview of the different components involved regarding action recognition and situation awareness.

B. Actions, Places, and Situations

The generated information from the scene analysis is used as input for the online process mining. So far, there are 33 modeled activities and actions, 9 modeled situations and 5 different places. Some examples are shown in Table I.

For the process mining, each occurring action, situation or place information is handled as an event.

Based on the behavior of the human co-worker different events and combinations thereof are recognized and made available through the result management module (see Fig. 2).

TABLE I. EXAMPLES FOR MODELED ACTIONS, PLACES, AND SITUATIONS

Actions and Activities	Places	Situations
Taking Tool	Cooperative Workplace A/B	Monitoring
Putting Tool Away	Manual Workplace C	Distraction
Sitting	Outside of Working Area	Communication
Crouching	Place in Between	Cooperation
Walking		Partially Attentive

IV. The Process Mining Module

This section is dedicated to discuss the online process mining module incorporated into the framework after a very short introduction to the underlying Heuristic Miner.

A. *Heuristic Miner*

The Heuristic Miner analyses a data log based on frequency and order of occurring events. Thus, temporal relations and dependencies are modeled depending on how often events occur after one another (see Fig.3).

According to [3], different temporal relations can be distinguished:

- A > B – Event B is direct successor of A,
- A → B – Event A comes generally before B,
- A >> B – There is a cycle of the form A-B-A,
- A >>> B – Marks *long-distance-dependencies*.

The norm, given as |·|, marks the number of instances that conform to the specified relation.

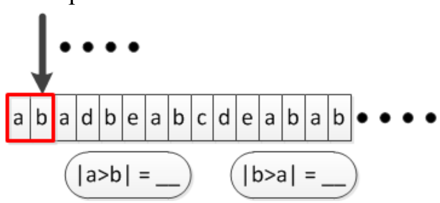


Figure 3. Frequency and temporal order analysis of occurring events.

Using these relations a dependency-graph is extracted. For each relation and combination of event pairs a coefficient based on the data log L is computed, for example:

$$a \Rightarrow b = \frac{|a >_L b| - |b >_L a|}{|a >_L b| + |b >_L a| + 1} \qquad (1)$$

$$a \Rightarrow_2 b = \frac{|a >>_L b| + |b >>_L a|}{|a >>_L b| + |b >>_L a| + 1} \qquad (2)$$

The value of the coefficient, similarly to a correlation coefficient, is in the interval [-1, 1], where 1 shows complete direct dependency and -1 complete inverse dependency. Using a threshold for the coefficient, the dependency-graph can be determined.

The advantage of this method is that a small amount of noise does not lead to significant changes in the process model. The role and influence of noise is closely examined in Section V.

In [3], the information of the dependency-graph is augmented by further analyzing different parameters in order to identify causal dependencies, which are modeled in a *Causal Net*. In the here presented work, only temporal dependency is assumed for occurring events. Thus, necessary computations to identify a causal net are left aside.

B. *The Online Mining Module*

The recognition and situation awareness results are stored as a triple: (Timestamp, Descriptor, Class). The descriptor is a string containing the identifier of the event, e.g., "Taking Tool". The class value represents if the event is an action, situation, location, etc. Based on the timestamp, the temporal order can be defined.

In order to accelerate computations during comparisons of events, a look-up table is defined to map descriptors onto integer identifiers. This mapping is done bidirectional before invocation of the mining algorithm and afterwards.

Also, the isolated analysis or combination of different event classes leads to either one-dimensional or multi-dimensional data logs. For the online analysis three different sets of event classes are considered:

- Actions alone,
- Actions and situations,
- Actions in combination with their occurring location.

The first two sets are considered one-dimensional, due to their subsequent occurrences. The third set is two-dimensional because actions are related to locations. Thus, the same action can occur at different places and the combination can characterize a sequence.

Another adaptation lies in the amount of analyzed data. It is not wanted to collect an event log of a week or day and perform process mining afterwards. Thus, a smaller amount of data needs to be accumulated. The actual number of logged events is dependent on the complexity of the process and can influence the process model significantly, which is shown in Section V.

The presented approach uses a sliding window to extract a current log from the history of recorded events. Events that are too old are discarded from the history. Also, if the event log is large enough the Heuristic Miner is triggered to analyze the new data. Afterwards, part of the event log is freed-up to allow accumulation of new events (see Fig. 4).

The mining algorithm is running in its separate thread in parallel to the overall framework. Thus, new events can be recognized and the event log buffer can be filled.

It would also be possible to invoke the mining algorithm with each new event but the computation load would increase drastically. Hence, the buffered approach is incorporated into this work. Details on runtime are presented in Section V.

In Fig. 5, an example of a process model is depicted. The visualization is done with the software library GraphViz. In order to reduce the complexity of the process model graph a graph-cycle search is used to extend the mining algorithm. As can be seen in Fig. 6, the extracted cycles ease the examination of the overall process model. This enables subsequent process analysis, e.g., for detecting production faults or inefficiencies.

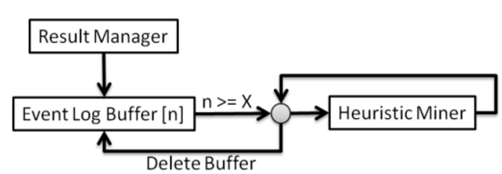


Figure 4. Online implementation of Heuristic Miner: Parallel to the framework the buffer is filled and subsequently analyzed.

 ISBN: 978-1-61208-269-1

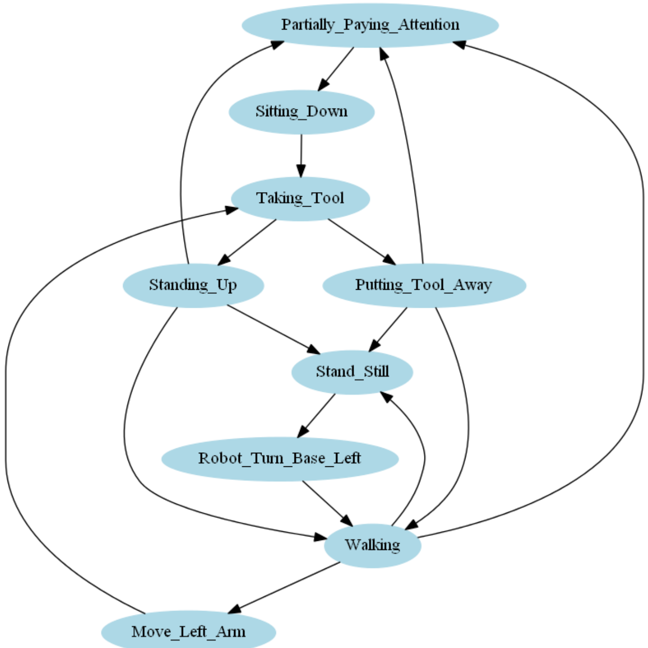


Figure 5. Example of a process model regarding recognized actions and activities only.

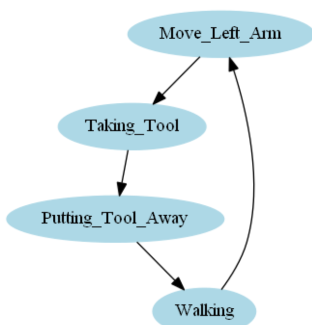


Figure 6. Example of extracted process cylce for better analysis of the overall process.

V. EXPERIMENTAL RESULTS

For experimental analysis of the online process mining module, different courses of actions were executed and results were recorded.

The evaluation was divided into two parts. Firstly, for analyzing different parameter influences efficiently means of automated event presetting were implemented. Secondly, actual recognition results based on sensor data processing were used during online execution to determine online behavior of the mining module. Thus, sensor noise resulting in recognition errors influences the mining procedure.

In Fig. 7, the overall setup of the working area is depicted, which is used throughout the experimental evaluation of the mining module. The figure shows the different work places, the reconstructed human kinematics and the robot. As mentioned in Section III.B the robots' movements and actions are not included into the event log.

In this section, analyzed parameters and mining result are illustrated and discussed.

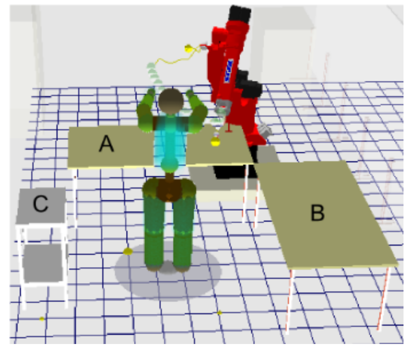


Figure 7. The overall setup of the working area.

A. Quality Criterion

For benchmarking the mining results a quality criterion was devised. Based on the comparison of mining results and expected outcome it tries to answer the following questions:

- Are the correct transitions identified?
- Are there too many nodes identified?
- Are all correct nodes identified?

Thus, the overall criterion Q is assessed by three different criteria:

$$C_1 = \frac{\textit{Correct Number of Transitions}}{\textit{Number of Found Transitions}} \Rightarrow Q_1 = |C_1 - 1|$$

$$C_2 = \frac{\textit{Number of Correctly Identified Nodes}}{\textit{Number of Identified Nodes}} \Rightarrow Q_2 = |C_2 - 1|$$

$$C_3 = \frac{\textit{Number of Correctly Identified Nodes}}{\textit{Correct Number of Nodes}} \Rightarrow Q_3 = |C_3 - 1|$$

$$\Rightarrow Q = max(Q_1, Q_2, Q_3) \qquad (3)$$

Using the maximum of the separate criteria defines the value which influences the outcome most negatively. Thus, a Q value of 0 is defined to be optimal.

B. Analysis Based on Synthetic Data

For efficiently analyzing different parameters a set of cyclic event log presets were defined. They included different courses of actions in different situations and at different places. Depending on the analyzed parameter, these presets could be tested in an isolated manner or in arbitrary combination.

The following parameters were evaluated using synthetic event logs:

- Number of different cycles in the event log,
- Length of the cycles,
- Influence of noise in the event log,
- Number of events in the event log buffer.

 ISBN: 978-1-61208-269-1

The variation of the number of different cycles in the event log resulted in a more complex process model. As expected, the value of the quality criterion stayed optimal. The same effect is also true for the variation of cycle lengths.

Robustness against noise is an important property when dealing with sensor based systems. For evaluation purposes, noise was added to the defined event log presets in two ways: (1) addition of random events after a complete event cycle, marked as R_1, and (2) addition of a random event in a cycle, marked as R_2. These two noise processes were also analyzed in combination. In Table II, results are presented. When using a combined noise ratio of 13% or greater the quality of the identification of temporal relations is insufficient. Also, the noise R_2 has generally a lower impact on the quality of the process model.

When considering an online process mining, it is important to know how much data needs to be processed. In Table III, results are shown for varying the event log buffer size regarding noisy event logs. Thus, using a buffer size of about 800 event entries yields acceptable results. Due to the greater influence of R_1 the impact of R_2 is discarded.

TABLE II. VARIATION OF ADDITIONAL NOISE. THE QUALITY CRITERION WAS EVALUATED FOR DIFFERENT NOISE VALUES.

R_2 \ R_1	0%	3%	5%	8%	12%	20%
0%	0	0.05	0.12	0.19	x	x
3%	0	0.12	0.19	0.12	x	x
5%	0.05	0.05	0.12	x	x	x
8%	0.05	0.12	x	0.26	x	x
12%	x	x	x	x	0.26	x
20%	x	x	x	x	x	0.46

TABLE III. VARIATION OF EVENT LOG BUFFER SIZE. THE QUALITY CRITERION WAS EVALUATED FOR DIFFERENT SIZES AND NOISE VALUES.

R_1 | R_2 \ Buffer Size	100	200	400	800
0% | 0%	8.5	0.72	0.35	0
3% | 0%	8.5	0.72	0.58	0.19
5% | 0%	8.5	0.9	0.9	0.19

C. *Analysis Based on Sensor Data Processing*

For experimental analysis using actual recognition results from sensor data processing different courses of action were defined (see Fig. 8). Different scenarios of differing complexity were used. Also, each scenario consists of sequences of varying complexity. Moreover, to simulate process faults and unexpected events, deliberately executed defined actions were introduced into the scenarios at random time (see Fig. 8 bottom row).

The result of the process mining for the second sequence of the second scenario (see Fig. 8 middle column and row) is shown in Fig. 9. The overall structure of the sequence is clearly visible and well captured.

Similar to synthetic event logs, the size of the event log buffer is of interest. In contrast, the buffer size resulting in best quality criterion is 400 log entries (see Table IV).

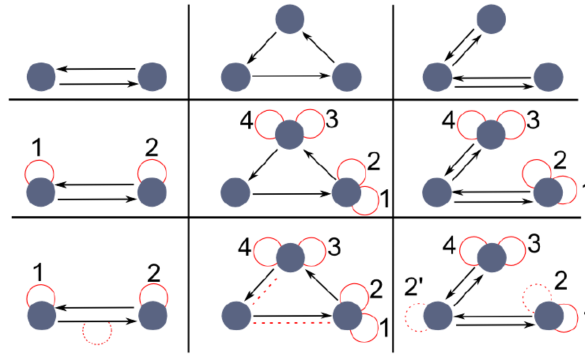
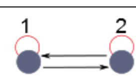
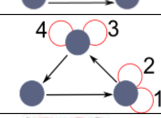
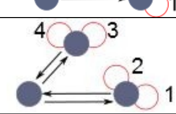


Figure 8. Different courses of action. Each column depicts a scenario, which is extended with each row. Full dots represent places, black arrows place changes, red circles actions, and dotted red lines deliberately executed noise.

Figure 9. Identified process model of the second sequence of the second scenario. Different colors represent different places: Yellow – A, Blue – B, Green – C, Grey – In Between, and Red – Outside.

TABLE IV. VARIATION OF EVENT LOG BUFFER SIZE. THE QUALITY CRITERION WAS EVALUATED FOR DIFFERENT SIZES AND SCENARIOS.

Scenario \ Buffer Size	100	200	400	800	1200
	4	0.5	0.29	0.29	0.4
	x	4	0.25	0.55	0.63
	x	1.5	0.2	0.42	0.47

Also of interest is the threshold for the dependency coefficient (see Section IV.A). In Table V, the corresponding results are presented listing the quality criterion for a buffer size of 400 event log entries. Q_1 represents the quality criterion for the first sequence and Q_2 of the second sequence of the second scenario. A threshold of 0.9 yields best results. This might change with varying complexity of the process but allows best results so far. Moreover, using a lower threshold allows for faster adaptation when the underlying process changes. Thus, using dynamic threshold adaption needs to be further investigated.

TABLE V. VARIATION OF COEFFICIENT THRESHOLD.

Threshold	0.85	0.875	0.9	0.925
Q_1	0.5	0.29	0.285	1
Q_2	0.55	0.36	0.25	0.43

The runtime behavior is another important characteristic for online processing methods. In Table VI, some results of the runtime measurements are shown. In tested scenarios, the needed computation time is well below one second. Thus, fast process mining is possible. Still, invocation of the mining with each new recognition result is not feasible so far. Considering that processes are not subject to change frequently and rapidly, this result is very promising.

TABLE VI. VARIATION OF COEFFICIENT THRESHOLD.

# Events	100-500	500-1000	1000-1500
Runtime [ms]	~ 300	~ 450	~ 650

VI. CONCLUSION

In this paper, an online process mining method was incorporated into a human-robot-cooperation framework. It is capable of identifying recurring event sequences, which consist of recognized actions, situations and locations.

Temporal relations of events are extracted and a process model is generated. In order to ease model analysis the mining algorithm is extended with a subsequent process cycle search. Thus, complex and unclear models are split into connected cyclic components.

The online mining method was experimentally evaluated regarding different parameters using defined event log presets and actual recognition results from sensor data processing. It was shown that the identified processes resemble the structure of the test scenarios. Also, a quality criterion was proposed for allowing comparison of different results.

The next steps will be the exploitation of the identified process model. Extracted cycles can be generalized and used for concept learning in the Description Logics based reasoning system. Thus, the foundation for action plan identification and recognition is laid.

The mining results are thus far only represented as dependency graphs. This does not allow for discrimination of major work cycles and processes that occur only eventually. There is further research needed to remedy that constraint.

Moreover, investigation of using an adaptive dynamic dependency coefficient threshold might allow faster adoption of changing work processes.

REFERENCES

[1] S. Puls, J. Graf, and H. Wörn, "Cognitive Robotics in Industrial Environments," Human Machine Interaction – Getting Closer, InTech, I. Maurtua(Ed.), Rijeka, Croatia, 2012, pp. 213-234.

[2] S. Puls and H. Wörn, "Combining HMM-Based Continuous Human Action Recognition and Spatio-Temporal Reasoning for Augmented Situation Awareness," Proc. of Int. Conf. on Interfaces and Human Computer Interaction, 2012, pp. 133-140.

[3] A.J.M.M Weijters, W.M.P. van der Aalst, and A.K.A. de Medeiros, "Process mining with the HeuristicsMiner algorithm," Report, Technische Universiteit Eindhoven, 2006.

[4] R. Sun and C.L. Giles, "Sequence Learning: From Recognition and Prediction to Sequential Decision Making," IEEE Intelligent Systems, 16(4), 2001, pp. 67-70.

[5] K.C. Santosh, B. Lamiroy, and J.-P. Ropers, "Inductive Logic Programming for Symbol Recognition," 10th Int. Conf. on Document Analysis and Recognition, 2009, pp. 1330-1334.

[6] B. Padmanabhan and A. Tuzhilin, "Pattern Discovery in Temporal Databases: A Temporal Logic Approach," Proc. of the Int. Conf. on Knowledge Discovery and Data Mining (KDD), 1996, pp. 351-354.

[7] F. Niu and M. Abdel-Mottaleb, "HMM-Based Segmentation and Recognition of Human Activities from Video Sequences," IEEE Int. Conf. on Multimedia and Expo, 2005, pp. 804-807.

[8] J. Krajcovic, M. Hrncar, and L. Muzikarova, "Hidden Markov Models in Speech Recognition," Advances in Electrical and Electronic Engineering, 7(1-2), 2011.

[9] N. Arica and F.T. Yarman-Vural, "An overview of character recognition focused on off-line handwriting," IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2001, pp. 216-233.

[10] D. Chudova and P. Smyth, "Pattern discovery in sequences under a Markov assumption," Proc. of the 8th Int. Conference on Knowledge Discovery and Data Mining (KDD), 2002, pp. 143-162.

[11] W.M.P. van der Aalst, "Process Mining – Discovery, Conformance and Enhancement of Business Processes," Springer, 2011.

[12] W.M.P. van der Aalst, "Process Mining: Discovering and Improving Spaghetti and Lasagna Processes," IEEE Symposium on Computaational Intelligence and Data Mining (CIDM), 2011, pp. 1-7.

[13] W.M.P. van der Aalst, A.J.M.M. Weijters, and L. Maruster, "Workflow mining: discovering process models from event logs," IEEE Transactions on Knowledge and Data Engineering, 16(9), 2004, pp. 1128-1142.

[14] M. Waimer, "Integration Adaptiver Prozess-Management-Technologie und Process Mining," Diploma Thesis, University Ulm, 2006.

[15] L. Wen, J. Wang, W. van der Aalst, B. Huang, and J. Sun, "A novel approach for process mining based on event types," Journal of Intelligent Information Systems, 2009, pp.163-190.

[16] J. Graf and H. Wörn, "Safe Human-Robot Interaction using 3D Sensor," Proc. of VDI Automation, 2009, pp. 1-12.

[17] S. Puls, J. Graf, and H. Wörn, "Design and Evaluation of Description Logics based Recognition and Understanding of Situations and Activities for Safe Human-Robot Cooperation," International Journal on Advances in Intelligent Systems, Vol. 4, 2011, pp. 218-227.