Cardiac Rehabilitation: an IoT Architecture Exploring Situation Awareness Based on Open-Source Technologies

Douglas Scheunemann, Wemerson D. Parreira, Adenauer Yamin, João Lopes and Cláudio Geyer

Abstract— **The IoT is influencing how computational systems are developed, enabling new application scenarios and more intelligent user interaction. The inference and prediction of user situation is a research challenge in IoT, because a large number of sensors information need to be collected and processed to implement those features. To handle this scenario, middleware systems are generally applied. This paper presents a view of EXEHDA, a middleware conceived to support the execution of IoT situation-aware applications. Particularly, we highlight an architecture integrated to EXEHDA middleware, named EXEHDA-HP, created for supporting hybrid processing of context data, aiming at detection of situations. To evaluate the architecture was carried out a case study in health area, focusing on cardiac rehabilitation. To prototyping the EXEHDA-HP were employed widely available and open-source technologies, potentializing the use of the architecture in different segments of society.**

Keywords— **Internet of Things, Situation Awareness, Cardiac Rehabilitation.**

I. Introduction

Advances in technologies of sensors, actuators, computers and wireless protocols allows the development of many types of connected devices. Things like home appliances, cars, medical devices, and cell phones can be connected and interact anytime, anyplace, with anything and anyone through an unified network, called Internet of Things (IoT) [1].

The IoT enables the deployment of a large number of sensors, increasing the quantity of raw data collected. To many applications the parameters of interest are obtained from the fusion of sensor data, establishing a more abstract view of actual context, allowing prediction and inference about system and user situation [2].

In this sense, one of the main research problems in IoT area is the situation awareness. Situation corresponds to a high-level and comprehensive view of contexts of interest of applications, which can be used by them in their adaptation process. This view is resultant of situations detection that can be composed by context data collected through different sensors distributed in the environment [3], [4].

The detection of situations in IoT can be a complex task, since in many cases the variables needed to determine the situations are not directly accessible throught the data collected. Furthermore, complex relationships among data may be required to recognize the situations of interest, which prevents the application of processing techniques based only in specifications, requiring also learning-based techniques. This way, it can be employed a hybrid approach to context processing, which combines different techniques to situations identification [5], [6].

The main contribution of this work is a software architecture, called EXEHDA-HP (Execution Environment for Highly Distributed Applications - Hybrid Processing), that provides situation awareness to the IoT applications, enabling a hybrid processing of context data, which combines specification-based and learning-based approaches for detection of situations. This work contributes specially to the Context Recognition and Adaptation Subsystem of EXEHDA middleware [7]. The prototyping of EXEHDA-HP architecture employed widely available technologies. In this sense, the information is collected through an Android smartphone, while the servers use Linux operating system. We expect that this technological approach potentializes the use of the architecture in different segments of society.

Through the case study on cardiac rehabilitation, we sought to demonstrate how the proposed architecture can be used in applications that contribute to the digital inclusion, enabling remote patient monitoring through a proactive application based on the identification of situations.

The paper is organized as follows. Section 2 describes the design of the proposed architecture. Section 3 presents the assessment of the architecture. Section 4 discusses the related work. Finally, section 5 presents the conclusion and future work.

II. Architecture Design

The EXEHDA-HP encompasses two types of servers: Border Server, responsible for interaction with the environment through sensors and actuators, and Context Server, responsible for processing the contextual information. These servers are located in cells of the environment managed by EXEHDA, where each cell has one Context Server and can contain several Border Servers, as shown in Figure 1.

The focus of this paper is the processing of context data, aimed at identifying situations of interest of applications. So, the next section describes the features and functionalities of the Context Server, highligthing the Processing Module.

Douglas Scheunemann, Wemerson D. Parreira and Adenauer Yamin. Catholic University of Pelotas, Pelotas - RS, Brazil, E-mails: scheunemann.d.a@gmail.com, wemerson.parreira@ucpel.edu.br, adenauer.yamin@ucpel.edu.br.

João Lopes and Cláudio Geyer¸ Federal University of Rio Grande do Sul, Porto Alegre - RS, Brazil, E-mails: jlblopes@inf.ufrgs.br, geyer@inf.ufrgs.br.

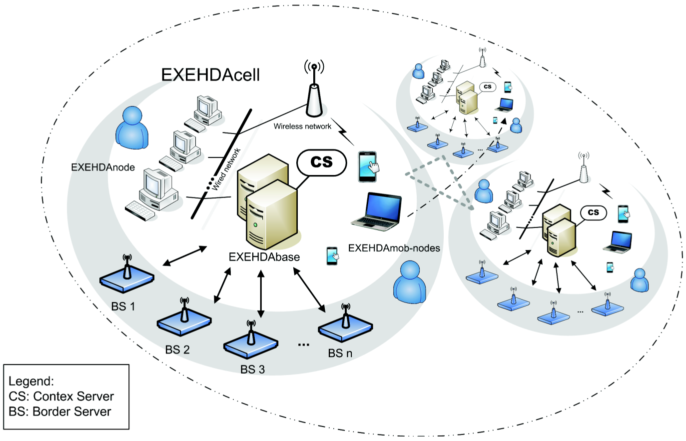

Fig. 1. Environment Managed by EXEHDA

A. Context Server

The modules of Context Server, following described, interoperate in the provision of the functionalities for situation awareness services. An overview of the Context Server architecture is illustrated in Figure 2, characterizing the relationship with Border Servers, other EXEHDA middleware services, other remote Context Servers, and applications.

Aquisition Module: provides support for the capture of contextual information, collected by Border Servers, considering logic (software interfaces) and/or hardware sensors. This module presents a server behavior whose functionality is implemented through an Enterprise Service Bus, allowing Border Servers publish context data, whenever there are significant variations in these data.

Actuation Module: in charge of the actuators control (activation, deactivation, and configuration), after being notified by other Context Server modules. This module receives actuator's identifier and operational parameters, and interoperates with Border Servers for trigger the actuators, enabling the use of EXEHDA in applications for automation and control.

Processing Module: processes contextual information, considering the situations of interest of the applications. Section II-B provides more detailed information about this module.

Notification Module: deals with notifying the result of context processing performed by the Processing Module. This module receives, through the Communication Manager, subscriptions from all services and/or applications that require notifications about the context state or situations.

Communication Module: used by remote Context Servers and applications to request situational/contextual data, and to trigger actuators.

B. Processing Module

The **Processing Module**, shown in Figure 3, has as main function perform the processing of contextual information, aiming at identifying situations of interest of the applications. Its hybrid approach enables that specification-based techniques (rules) may be employed in conjunction with learning-based techniques. This combination of techniques increases the number of classes of situations that can be inferred, improving the quality of the IoT applications managed by EXEHDA middleware.

The **Context Manager** block is responsible for providing the functionality for processing of contextual information, comprising the steps of interpretation, fusion, storage and query. This process aims at raising the level of abstraction of the collected context data, improving their availability and usability in situations identification process. The fusion and interpretation steps can be performed using models based on learning, rules, or both, thus characterizing a hybrid model.

One of the main challenges of the proposed hybrid model is to allow learning algorithms can be parameterized and used in a transparent way for the EXEHDA application layers. With this, the objective is to prevent that details of its operation, often complex, require treatment by the administrator of the application. To add this functionality to EXEHDA was created the components of **Learning-based Engine** block.

One of the features used to enable the abstraction of the processing techniques was the inclusion of a service for training, which uses the same data model regardless of the chosen algorithm. Thus, the user provides the sample data for training, chooses the processing technique and rules for extraction of features that must be performed on the input data. After the execution of the training, obtained parameters are stored in the **Learned Models** repository. These parameters are accessed through the functions of **Use API** component, used during the execution of the application.

Processing algorithms are stored in the form of software components in the middleware repository, and new components can be included. The interfaces of the new components must be built according to the interoperability model, standardized to the module in question.

The **Features Processing** component is responsible for preprocessing the context variables or signals read directly from sensors, extracting the set of relevant features for training and subsequent implementation of the learning algorithm. The routines for preprocessing, as well as learning algorithms are software components stored in the repository. Its execution is done by a rule created by EXEHDA administrator. As example of algorithms for feature extraction may be cited statistical methods as mean, variance, covariance, maximum and minimum, and signal processing methods like FFT.

The **Situation Classifier** block allows inferences of situations, considering the context information generated by the hybrid processing model. In this activity are employed models of fuzzy reasoning together with learning-based techniques.

The choice of fuzzy logic for situation inference layer was motivated by its capabilities of mapping computational data for a linguistic variables domain, which facilitates the establishment of rules for high-level specialists. In addition, fuzzy logic can treat the uncertainties of information and discrepancies between context variables [5].

To perform the inference of situations can also be applied to learning-based algorithms. These algorithms may be configured and accessed through the same APIs used in the programming of context processing model.

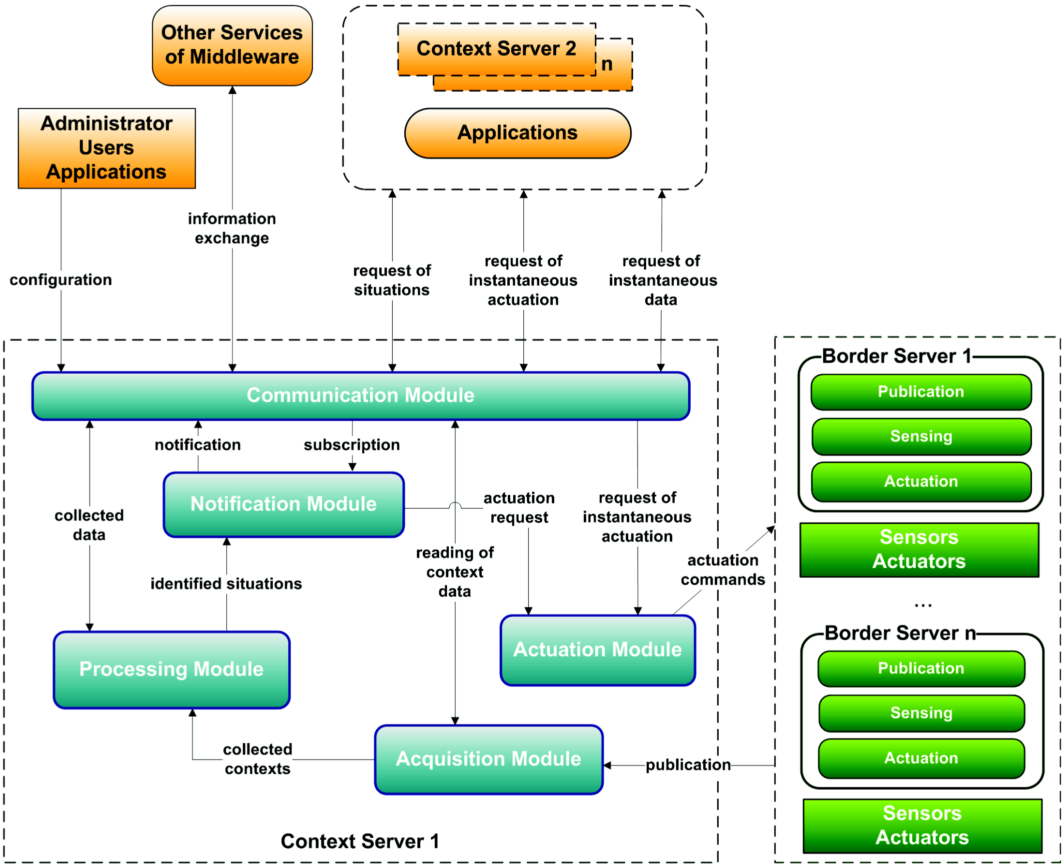

Fig. 2. Context Server

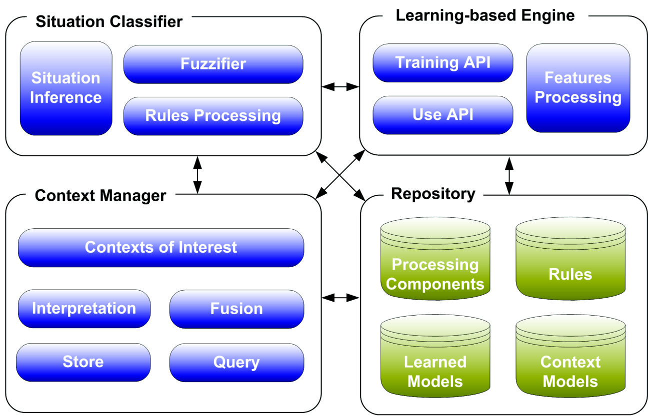

Fig. 3. Processing Module

III. Architecture Evaluation

In order to evaluate the hybrid processing approach presented in this paper, we designed an application to assess the functionalities of the EXEHDA-HP architecture.

The chosen application was the simulation of monitoring patients in rehabilitation after a stroke. Cardiac rehabilitation can involve various therapies, including administering medications, nutritional advice and also the prescription of physical activity. Cardiac rehabilitation through physical activity is considered a central therapy. Studies indicate that rehabilitation based on physical exercise was associated with a reduction of 20 to 30% in mortality rates when compared to care without exercise [8].

One aspect that should be considered in the therapy through exercises is the disproportionate responses in heart rate. This may indicate a risk for the patient [9]. Activities of daily life such as walking, going up and down stairs, can also lead to situations of risk depending on the patient's condition. Thus, the recognition of physical activity and its correlation with the heart rate level allows a safer recovery.

The structure for context processing used to identify situations can be seen in the figure 4. The Border Server was made using an Android smartphone, applied to collect data of heart rate of the patient and also the sign of the smartphone accelerometer used to determine the physical activity of the patient. The Context Server was prototyped using a computer with Linux operating system, which do the context processing and situations identification.

The learning method used to identify the activity of the patient will be explained in the section III-A and the identification of situations through fuzzy adaptation and rules are shown in section III-B.

A. *Inference of Activities*

The inference of the physical activity applying machine learning algorithms is an application widely discussed in the literature. The application of models that use accelerometers

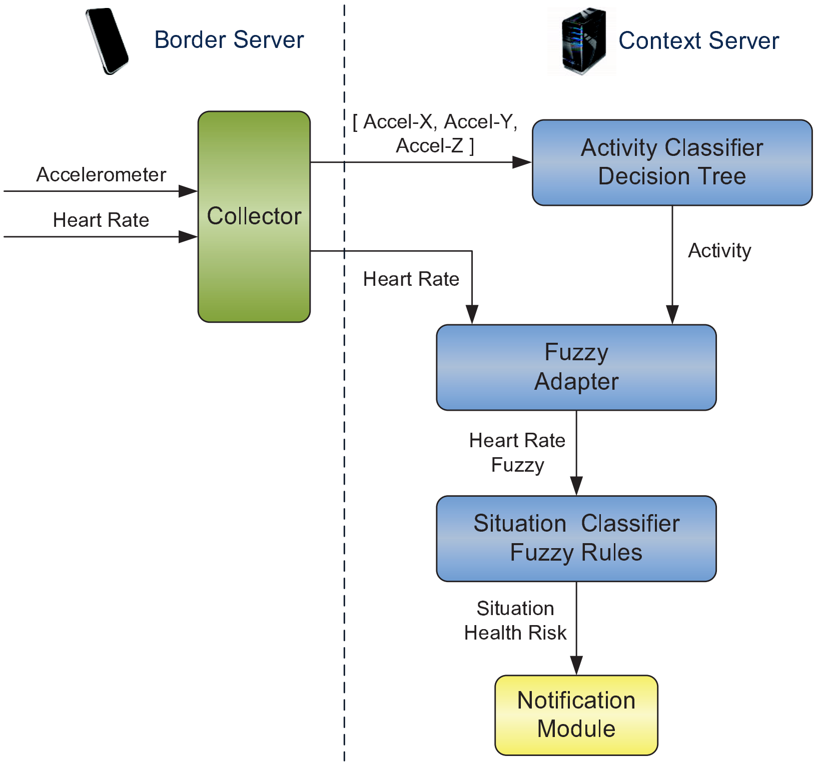

Fig. 4. Processing diagram of the study case in cardiac rehabilitation.

TABLE I
% OF ACTIVITIES CORRECTLY PREDICTED.

	J48 Decision Tree	Bayes Network	Multilayer Perceptron	Logical Regression
Walking	87.33%	82.94%	89.23%	92.37%
Jogging	95.80%	94.19%	98.89%	97.41%
Upstairs	59.21%	39.31%	52.96%	24.51%
Downstairs	54.66%	30.64%	40.17%	10.35%
Sitting	98.66%	94.98%	98.33%	95.32%
Standing	97.56%	84.96%	93.90%	91.06%
Overall	84.88%	77.50%	84.39%	79.05%

of smartphones has gained notoriety in recent years because of technological advances these devices [10].

In the study by Kwapisz et al. [11] were captured accelerometer signals from 29 volunteers during the physical activities: walking, jogging, ascending stairs, descending stairs, sitting, and standing. The signs were made available in a database, which was used in this work to verify the operation of the training and use APIs. The results after training of different learning algorithms can be found in Table I.

Learning machine algorithms were accessed through a Java API for Weka software. In the **Processing Components** repository of EXEHDA-HP was stored the Java codes that providing access to each algorithm. The Weka software must be installed in Context Server to run the API calls.

Among the tested processing methods, which had the highest overall accuracy was Decision Tree. Moreover, this method also required a minor processing time for his training when compared with other methods. The results were consistent with those obtained by Kwapisz et al. [11]. Thus, it can be demonstrated the operation of the proposed architecture for training and execution of learning-based algorithms.

B. *Inference of Situations*

We use the information about recognition of patient activity, described in Section III-A, in conjunction with other context data for composing situations in **Situation Classifier** block of EXEHDA-HP. For the use case discussed in this paper, we aim at determining a situation that represents the level of risk for physical activity performed by the patient, classified in the following linguistic domain: low, moderate and high.

The level of heart rate appropriate for each physical activity must be previously informed by the physician. The linguistic domain to heart rate is given for very low, low, normal, high and very high. In **Fuzzifier** component of EXEHDA-HP the numerical values obtained from the heart rate sensor are transformed to fuzzy domain applying a triangular membership function. In Figure 5 can be seen an example of membership functions used for the conversion of the heart rate to the fuzzy domain. In the Figure Figure 6, the pseudocode shows the fuzzy reasoning model used to determine the risk situation of the patient.

IV. RELATED WORK

The study of related work has been done considering the main design assumptions of EXEHDA-HP: (a) architecture used in different application domains; (b) distributed architecture; (c) hybrid model for processing the context, considering specification-based and learning-based algorithms; (d) support for the training of learning algorithms; and (e) identification of

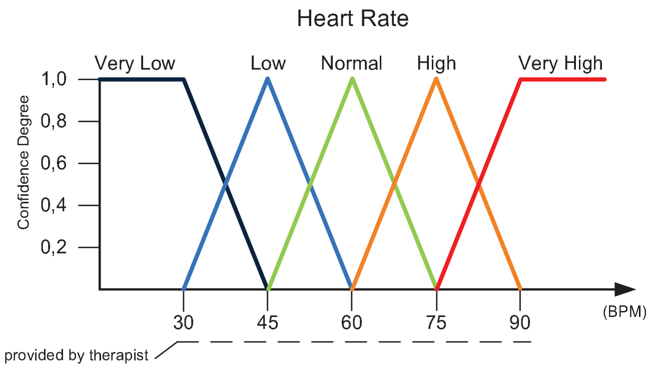

Fig. 5. Representation of heart rate through fuzzy logic.

```
GET activity
GET activity_normal_params
IF heart_rate IS very_high OR heart_rate IS very_low THEN
   risk IS high
IF heart_rate IS high OR heart_rate IS low THEN
   risk IS moderate
IF heart_rate IS normal THEN
   risk IS low
```

Fig. 6. Situation inference algorithm.

situations. Table II shows a comparison between related work and EXEHDA-HP.

TABLE II
COMPARISON BETWEEN RELATED WORK AND EXEHDA-HP.

	a	b	c	d	e
[12]	no	yes	no	no	yes
[13]	no	no	yes	no	yes
[14]	yes	yes	no	no	yes
EXEHDA-HP	yes	yes	yes	yes	yes

Yuan and Herbert [12] propose an architecture for remote health services to support the monitoring and recognition of activities and habits, with the consequent generation of alerts. This work allows the use of only a learning algorithm, which is based on cases. Unlike, in EXEHDA-HP four algorithms was tested. Furthermore, EXEHDA-HP provides support for inclusion of other algorithms, in the form of software components.

Monitoring of the health of hypertensive patients is the application that gave rise to the model proposed by Haghighi et al. [13]. In this work, the authors consider a hybrid model of context processing, analyzing the patient's activity and some physiological parameters to compose the health situation. The architectural model proposed by the authors is associated with a specific application and has no support to distributed processing. In turn, the EXEHDA-HP is designed to be independent of application domain, and the architectural model is structured in a distributed way, at all stages of handling context information.

Cimino et al. [14] propose a generic software architecture for situation awareness management. The proposal aims at providing a general approach to situation awareness, in which both architecture as behavioral knowledge can be integrated into a distributed environment for many types of applications. The hybrid model for situation inference proposed considers the use of rules and semantic models. In turn, the hybrid model of EXEHDA-HP employs rules and machine learning algorithms.

V. Conclusion and Future Work

The main contribution of EXEHDA-HP is an architecture to provide situation awareness for IoT applications, with support to hybrid processing of context data. Developed software components allow the use of specification-based and learning-based approaches to detect situations. The evaluation showed promising results, both in terms of the architectural model design as the technologies used. The prototyping with IoT resources, widely available hardware, and open-source software solutions allow that the developed architecture could be employed in several segments of society, enhancing the digital inclusion.

As future work we intend to improve the routines for training the learning algorithms, including two more features: automatic features selection, and automatic selection of the learning technique.

References

[1] M. A. Razzaque, M. Milojevic-Jevric, A. Palade, and S. Clarke, "Middleware for Internet of Things: a Survey," *Internet of Things Journal, IEEE*, vol. PP, no. 99, p. 1, 2015.
[2] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context Aware Computing for The Internet of Things : A Survey," *IEEE COMMUNICATIONS SURVEYS & TUTORIALS*, vol. X, no. X, pp. 1–41, 2013.
[3] P. Bellavista, A. Corradi, M. Fanelli, and L. Foschini, "A survey of context data distribution for mobile ubiquitous systems," *ACM Comput. Surv.*, vol. 44, no. 4, pp. 24:1–24:45, Sep. 2012.
[4] M. Knappmeyer, S. Kiani, E. Reetz, N. Baker, and R. Tonjes, "Survey of context provisioning middleware," *Communications Surveys Tutorials, IEEE*, vol. 15, no. 3, pp. 1492–1519, 2013.
[5] J. Ye, S. Dobson, and S. McKeever, "Review: Situation identification techniques in pervasive computing: A review," *Pervasive Mob. Comput.*, vol. 8, no. 1, pp. 36–66, Feb. 2012.
[6] S. E. Bibri, "Context modeling, representation, and reasoning: An ontological and hybrid approach," in *The Human Face of Ambient Intelligence*, ser. Atlantis Ambient and Pervasive Intelligence. Atlantis Press, 2015, vol. 9, pp. 197–257.
[7] J. Lopes, R. Souza, C. Geyer, C. Costa, J. Barbosa, A. Pernas, and A. Yamin, "A middleware architecture for dynamic adaptation in ubiquitous computing," *Journal of Universal Computer Science*, vol. 20, no. 9, pp. 1327–1351, sep 2014.
[8] D. Rabelo, C. Gil, and S. D. Araújo, "Reabilitação cardíaca com ênfase no exercício: uma revisão sistemática," *Revista Brasileira de Medicina do Esporte*, vol. 12, no. 5, pp. 279–285, 2006.
[9] C. E. Negrão and A. C. P. Barreto, *Cardiologia do Exercício: do Atleta ao Cardiopata*, 3rd ed. Barueri, SP - Brazil: Manole, 2010.
[10] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P. Havinga, "A survey of online activity recognition using mobile phones," *Sensors*, vol. 15, pp. 2059–2085, 2015.
[11] J. R. Kwapisz, G. M. Weiss, and S. a. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explorations Newsletter*, vol. 12, p. 74, 2011.
[12] B. Yuan and J. Herbert, "Context-aware hybrid reasoning framework for pervasive healthcare," *Personal and Ubiquitous Computing*, vol. 18, no. 4, pp. 865–881, 2014.
[13] D. P. Haghighi, P. Averi, M. Indrawan-Santiago, and H. T. Minh, "Situation-aware mobile health monitoring," *ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)*, pp. 248—-256, 2014.
[14] M. G. C. A. Cimino, B. Lazzerini, F. Marcelloni, and A. Ciaramella, "An adaptive rule-based approach for managing situation-awareness," *Expert Syst. Appl.*, vol. 39, no. 12, pp. 10 796–10 811, 2012.