Perception Assistance for the Visually Impaired Through Smart Objects: Concept, Implementation, and Experiment Scenario

JEAN CONNIER[1], **HAIYING ZHOU**[2], **CHRISTOPHE DE VAULX**[1], **JIAN-JIN LI**[1], **HONGLING SHI**[1], **PHILIPPE VASLIN**[1], **AND KUN MEAN HOU**[1]

[1]Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes (LIMOS), UMR6158 UCA-CNRS, 63170 Aubière, France
[2]Hubei University of Automotive Technology, Shiyan 442002, China

Corresponding author: Haiying Zhou (zhouhy_dy@huat.edu.cn)

ABSTRACT The last few years have seen substantial progress in the field of smart objects (SOs): their number, diversity, performance and pervasiveness have all been quickly increasing and this evolution is expected to continue. To the best of our knowledge, little work has been made to leverage this abundance of resources to develop assistive devices for Visually Impaired People (VIP). However, we believe that SOs can both enhance traditional assistive functions (i.e. obstacle detection, navigation) and offer new ways of interacting with the environment. After describing spatial and non-spatial perceptive functions enabled by SOs, this article presents the SO2SEES, a system designed to be an interface between its user and neighboring SOs. The SO2SEES allows VIP to query surrounding SOs in an intuitive manner, relying on knowledge bases distributed on Internet of Things (IoT) cloud platforms and the SO2SEES's own back-end. To evaluate and validate the exposed concepts, we have developed a simple working implementation of the SO2SEES system using semantic web standards. A controlled-environment test scenario has been built around this early SO2SEES system to demonstrate its feasibility. As future works, we plan to conduct field experiments of this first prototype with VIP end users.

INDEX TERMS Internet of Things cloud framework, OWL, smart objects, semantic web rule language (SWRL), visually impaired people (VIP).

I. INTRODUCTION

Perception of the environment is essential for human daily life activities and vision appears to be the best sensory channel for acquisition of spatial information because it provides relatively simultaneous perception of large spatial fields [1]. Navigating in an unknown environment while avoiding obstacles is thus a big challenge for visually impaired people (VIP) who were estimated to be 252.6 million in 2017 [2], among whom 36.0 million were blind, to which 188.5 million people suffering from mild visual impairment (i.e. with a corrected visual acuity less than 6/12 but 6/18 or more in their best eye) can be added. To assist blind people in their daily trips, different tools are available, such as white canes, guide dogs and, recently, electronic travel aids (ETAs). Whereas a lot of research in the area of assistive devices for VIP has been focused on improving sensory ability of systems (e.g. computer vision or obstacle detecting range sensors), our research aims at integrating assistive systems into the developing framework of the Internet of Things (IoT). In this paper, we present a novel system that associates an integrated assistive device to Smart Objects (SOs) and their own cloud back-ends to improve environmental perception for VIP.

A. PERCEPTION

The definition of perception varies depending on sources. For Efron [3], it is "[…] *man's primary form of cognitive contact*

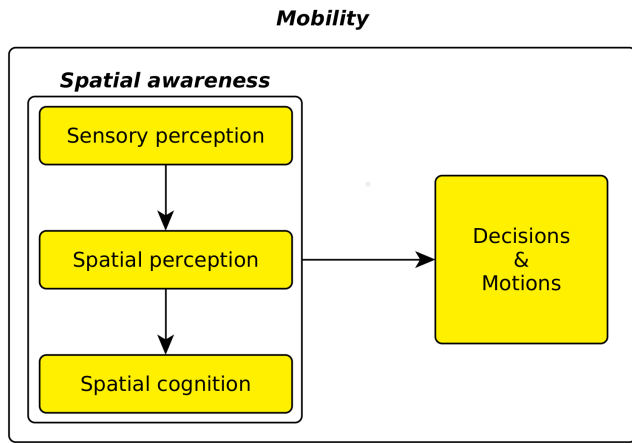The associate editor coordinating the review of this manuscript and approving it for publication was Chin-Feng Lai.

FIGURE 1. One way of seeing mobility.

FIGURE 2. Simple model of communication paths used by assistive devices.

with the world around him'', while for Schacter *et al.* [4], it is "[…] *the organization, identification, and interpretation of a sensation in order to form a mental representation.*'' The progress in neuroscience has refined the questions around perception. These questions often relate to the boundaries defining perception, such as the relationships between motor functions and perception [5]. In that spirit, many discussions arise from the level at which perception should be defined: some authors see it as a broad field, covering all perceptual experience, while other restrict its meaning to a well-defined field of information processing [6]. In this article, we adopted the first definition, because it allows us to associate with cognitive and conscience issues more simply than with the second one. Indeed, perception of VIP raises several questions about spatial and non-spatial perception. Spatial perception is the mechanism of acquisition of spatial information, i.e. the spatial configuration and shape of objects. It does heavily rely on vision [7] but not exclusively. A major part of the literature is devoted to visual perception, which is the sensory modality that sighted people use the most, because it is the sense that conveys the highest volume of information since it has a higher bandwidth than other senses [8]. Marr [9] regards the ability of vision to give geometry and spatial information as its capital function, considering that *visual perception* is above all *spatial perception*. Visual perception has been widely studied because it is of paramount importance for daily tasks, mainly mobility. In a first-order approach, mobility is the displacement of one's body. Such displacement is not random and requires decisions that in turn rely on perception, in particular spatial perception (Fig. 1). Since mobility is a vital task and heavily depends on spatial perception, it is essential for assistive systems to take into account their relation and interactions with the users' spatial perception of their environment. Some definitions of perception, such as the one from Schacter *et al.* [4], do not restrict perception to spatial perception. Non-spatial processes equally require formation of a mental representation; the difference is that this representation may relate to non-spatial, non-physical or
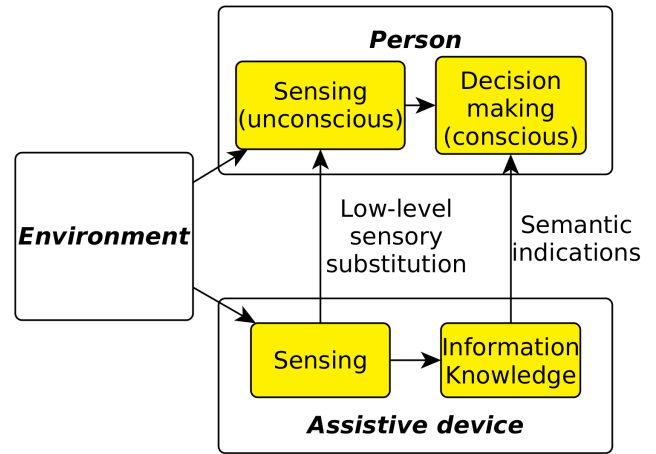
virtual objects. For instance, the perception of other people's emotions is a non-spatial form of perception, as is. Although interesting, this topic is beyond the purpose of the present article.

B. STATE-OF-THE-ART ASSISTIVE DEVICES

One way to look at assistive systems is to analyze how they communicate with their users. The nature of this human-machine interface is a structuring characteristic of devices. In this paper, we only considered the flow of information going from the device to the user – and not the opposite way – since most existing devices work this way. Two "channels" are mainly used (Fig. 2): the system can send information to the user either through a *conscious* path or through an *unconscious* path. The *conscious* path is a way of communicating information to the user through explicit, conscious cognitive processes. In this modality, the information is generally encoded in a verbal form. The nature of verbally-encoded information should be of higher level than that of the direct or preprocessed sensor information that is given by sensory substitution devices: if the user gets messages describing non-contextualized and plain scalar data, the verbosity induced by the encoding will increase the cognitive load. Such an operation, which amounts to verbally communicating raw information, is inefficient. The *unconscious* path is essentially the sensory substitution path, introduced by Bach-y-Rita *et al.* [10], which is the communication of information through another sensory modality than the one that would normally be used. In the case of assistance to VIP, it means the acquisition of visual information by a device (e.g., a camera) and, after a pre-processing stage, the communication of this information to the person through another sensory modality (e.g., tactile or non-linguistic hearing). In a broad sense, most assistance to blind people amounts to sensory substitution, since information supplementing the loss of sight has to go through another channel. But here, we only use the term for devices transmitting an isomorphic representation of what is

perceived by the sensor, as described by Loomis *et al.* [8]. Successful sensory substitution devices rely on sensorimotor loops [11]: the user must be continuously moving the sensor in order to be able to get what is perceived by the device. Recent examples of this approach are given by Tactile Vision Substitution Systems (TVSS), as those developed by Hsu *et al.* [12] and Hiroyuki Kajimoto *et al.* [13], for instance. These systems are able to extract relevant information from videos and to convert them into tactile stimulation but their resolution is still far below that of vision. The distinction between conscious and unconscious cognitive paths is not easy, neither is the mapping between these paths and assistive systems. For example, when using a white cane, one can both directly "feel" the distance to a wall (an unconscious cognitive process) while sometimes needing explicit thinking to understand the shape of the wall, when it becomes less intelligible (a conscious cognitive process). Furthermore, these two modalities lead to different levels of fatigue and cognitive load: constant conscious processes generate a high cognitive load and fatigue, while unconscious cognitive processes are less tiring even though they are not always without burden. Over the years, many assistive systems have been developed. They rely on a wide variety of principles and mechanisms to provide several functions to VIP. We analyzed the existing scientific literature from the point of view of the paradigms that, taken individually, interest us the most. First, numerous devices have aimed at improving spatial sense of distant objects. The traditional white cane and its electronic derivatives belong to this category. Among these, remarkable examples in research are the EyeCane [14], and the Guide-Cane [15]. The Eye Cane only offers obstacle detection but its interface is intuitive and easy to learn. The GuideCane is of particular interest because it integrates two functions: obstacle detection and navigation assistance. Second, there are systems that contextually interact with the environment, like those that are able to give information about transportation [16]. This is the case of remote-controlled auditory signals, such as those used in traffic signals. The Handisco Sherpa is a commercial smart cane [17] that integrates two examples of contextual interactions with the environment: navigation functionalities together with the ability to give information about public transportation timetables and to activate sound beacons at intersections. A new field in the domain of context-relevant information is the automated scene description, such as done by Microsoft's Seeing AI smartphone application [18]. This application is able to perform many contextual tasks that require vision such as analyzing and describing scenes, reading text and identifying people. A third category gathers devices that rely on infrastructures, which can be pre-existing, as is the case with many localization systems relying on WiFi access points [19], or which may be designed and deployed specifically for the assistive system, like the Remote Infrared Signage System (RISS) or Talking Signs ® [20], [21]. Developing a specific infrastructure for assistance allows to offer a wide range of functionalities to VIPs, but the deployment and the maintenance of a dedicated infrastructure increase its complexity and its cost. On the other hand, using a pre-existing infrastructure means that the assistive system can be designed to work wherever the infrastructure is available (faster scaling), but restricts the developers' freedom, since the assistive system has to adapt to the infrastructure functional constraints. It should be noted that infrastructure does not always have to embed electronics: for instance, indoor signage can be directly recognized by a standard video camera [22]. To the best of our knowledge, no assistive systems for VIPs use SOs present in the environment. Some come somewhat close, such as the Talking Signs ®, but they do not fully fit the definition of SOs. In this context, the system proposed in this paper is new as it is at the intersection of several paradigms: (i) it relies on existing infrastructure, (ii) it works with externally managed SOs, (iii) it collects information from the environment, and (iv) it provides both spatial and context-related high-level information to VIP users.

C. THE 2SEES CONCEPT

The 2SEES project aims to build an Electronic Travel Aid (ETA) able of helping VIP to cope with mobility issues. It is an improvement of the former SEES project [23] and its first key concept point is the gathering of several functionalities for mobility in a single system and object: initially, *walking* assistance and *orientation* assistance. In order to be able to *walk* safely, blind people must have several abilities: detecting obstacles, estimating the traveled distance, identifying the nature of the ground by feeling its surface (e.g. tactile paving surfaces), and estimating the state of traffic lights. Moreover, in order to achieve a successful *orientation*, VIP have to estimate the direction they must follow to reach the targeted destination, which usually requires them to know their current location. Together, these two functions should allow VIP to move from their current location to a desired destination. In order to perform these functions simultaneously, the 2SEES hardware platform must embed several types of sensors (see section III below). However, continuous and simultaneous use of all sensors would result in prohibitive energy consumption by both the sensors themselves and the computing resources required to process their data. *Multi-sensor active context awareness* is an essential concept and design element: context information is used to dynamically determine which sensors provide the most meaningful data in a given situation. Therefore, embedded sensors can be enabled or disabled. Network connectivity is another essential feature of the 2SEES: it allows for storing tailor-designed maps and the presence of a cloud back-end in the system allows the offloading of computationally expensive processes.

D. SMART OBJECTS AND THE INTERNET OF THINGS

The idea behind the Internet of Things (IoT) is to extend the connectivity of the Internet to the physical world through network-connected physical objects: Cyber Physical System (CPS). The *things* were initially RFID-identified plain objects [24], but the concept has evolved and merged to some degree with the neighboring concept of pervasive

computing [25]. Consequently, the generally accepted definition of what an *IoT object* is leans more and more towards *smart objects* (SOs). SOs are autonomous objects embedding sensors and actuators to interact with their physical environment [26]. They also embed hardware and software allowing them to run cognitive algorithms and make decisions using the information they handle [26]. Then, to be part of the IoT cloud platform, they must have networking capabilities and be connected to the Internet. In this work, we chose not to rely on a specific, hard definition of the IoT and instead opted for a *de facto* approach: we based our IoT integration on a vastly shared common characteristic of IoT environments, the reliance on cloud computing back-ends to harvest the data generated by deployed objects. In this paper, all IoT nodes and communicating objects are defined as SOs. To further allow easy integration into existing and future ecosystems, we tried to adopt widespread standards. In particular, the W3C standards (i.e. RDF, RDFs and OWL) serve as the basis for the representation, integration and extraction of information and knowledge. The key point of the system is also based on an upcoming W3C standard: the Web of Things-Things Description (WoT-TD) standard [27] that defines a formal model for objects description, from the point of view of interactions. WoT-TD fits our application rather well because it is build upon W3C semantic web standards and apparently oriented towards consumer objects and usages. It corresponds to our usage – automatic discovery of surrounding objects – while staying away from low-level considerations, i.e. avoiding additional constraints about network connectivity or the hardware of devices.

II. SO2SEES PERCEPTION

In this paper, we propose a new concept based on SOs and IoT cloud infrastructure and framework, in addition to the sensors commonly found in assistive devices (e.g., ultrasonic sensors) to provide VIP with a complementary perception of the environment.

A. ARCHITECTURE

The SO2SEES architecture involves several "actors", each one being composed of several components (Fig. 3) Each of these actors has a local tier and a remote cloud back-end. The main actor is the 2SEES system; its local tier includes the sensor platform, called Local Mobility Smart Cane (LMSC), and a smartphone running a home-made application acting as a data processing node, user interface and Internet link to the cloud back-end. The other actors are the vendors and managers of SOs. For them, we consider a scenario of the "vertical, open, standardized" type: each manufacturer, vendor or manager is supposed to have its own low-level protocols and standards up to a cloud platform, hence the "vertical" characteristic. However, at least some of the information generated by the fleet should be shared on a cloud platform and be reachable through a standardized mechanism based on Internationalized Resource Identifiers (IRIs) – denoting the open and standard characteristics, respectively.
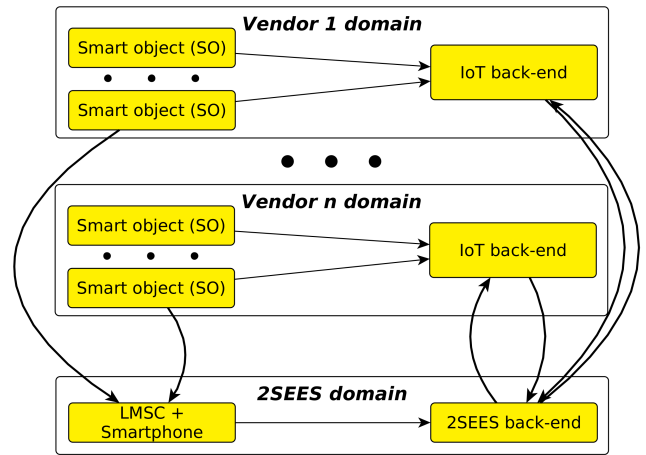

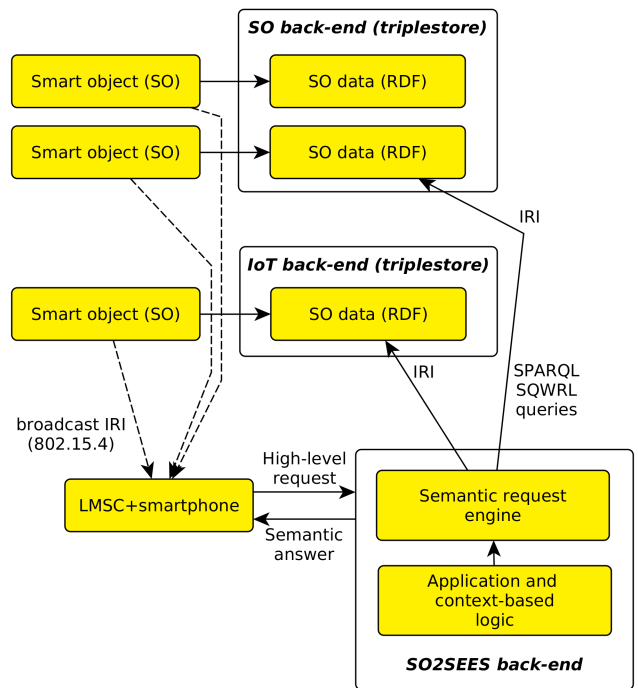**FIGURE 3.** Actors and relationships in the SO2SEES architecture.

FIGURE 4. SO2SEES communication model (RDF: Resource Description Framework; IRI: Internationalized Resource Identifiers, an extension of URI/Uniform Resource Identifiers; LMSC: Local Mobility Smart Cane, a component of the 2SEES; SPARQL: SPARQL Protocol and RDF Query Language).

Furthermore, formal descriptions of the SOs are obtainable through local discovery (Fig. 4). When navigating into an environment, the local tier encounters SOs and acquires their IRIs through a local discovery process. It sends these IRIs to the 2SEES cloud back-end, which uses them to obtain the formal descriptions of the objects. The cloud back-end can then analyze these descriptions to extract context cues that are used to determine which queries are useful with respect to the current "total context". This "total context" is the union of the SO context, which is extracted from
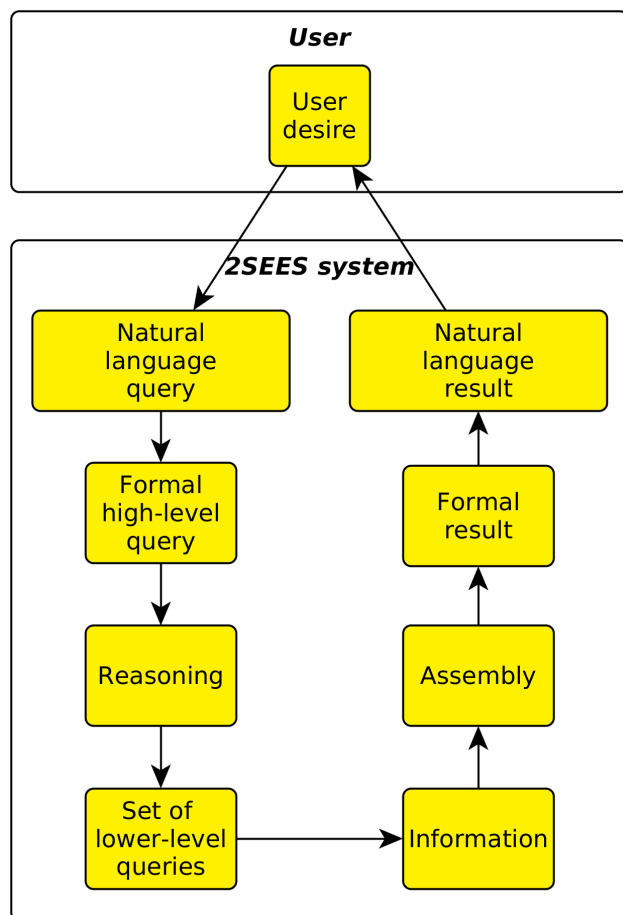
FIGURE 5. SO2SEES query model.

the retrieved descriptions of the surrounding SOs, and the more general context of the user's habits, explicit preferences and current whereabouts. The resulting queries can either be constructed from the bottom up with artificial intelligence-backed linguistic semantic techniques, or selected in a set of existing queries using predefined context-matching rules. Once the cloud (back-end) has selected or generated context-relevant queries, it sends them to the user interface running on the smartphone, making them available for the user. The execution of the queries flows back to the cloud, which runs the query procedures, querying the corresponding SO clouds if further information about the SOs they manage is required. The queries and context evaluation are described in the following section.

B. FUNCTIONALITIES, QUERIES AND RULES
Information obtained through communications with SOs allows the 2SEES to both enhance its basic functionalities and provide new ones. The general querying process is the following: the user queries the system in natural language; this query is translated into a formal high-level query, then split into a set of lower-level queries made to SOs' cloud back-ends. Once these have been answered, the information

obtained is assembled according to the logic of the high-level query to answer the natural-language query (Fig. 5). This type of query can be used both for assistance to spatial and non-spatial perception.

1) SOs FOR SPATIAL PERCEPTION
Spatial perception is a central function of the 2SEES system and, more generally, of the mobility problem of VIP. There are multiple ways in which surrounding SOs can prove useful for localization. Firstly, as radio positioning is a widely used method for localization, SOs can be used as radio beacons. Outdoors, global navigation satellite systems (GNSS) such as Galileo and the ubiquitous Global Positioning System (GPS) are large-scale implementations of this principle. Indoors, since devices are unable to receive signal from GNSS because of obstacles (e.g., roofs, walls) to radio frequency signals, local radio beacon infrastructures are often the chosen method to achieve accurate localization [19]. For instance, Wi-Fi access points are often used for indoor positioning. Although not all buildings are yet equipped with specific localization, because the deployment and maintenance of these infrastructures is costly, leveraging existing, pre-deployed or externally planned RF infrastructures, such as Wi-Fi access points, appears to be a wise choice for indoor radio localization. And, given the ever-increasing prevalence of SOs, it is likely that they will constitute in the near future a dense mesh, perfect for localization. Secondly, SOs could be used to directly improve the spatial mental representation of users, through several mechanisms. The function of a location is tightly linked to the objects it contains, making the link between objects and places easier to remember. For instance, one component of the identification of locations is their social function (e.g. a room can be a kitchen, an office, or a shop) [28]. Associating places with their role helps to remember them, in particular if one can relate oneself to this role. SOs thus be used as location function markers. For instance, an average western kitchen will contain several cooking appliances, whereas computers and printers are mostly found in an office. If these objects are detected by the 2SEES, the execution of a simple cognitive algorithm can lead to the function of the place. Detection can be done through optical means (e.g. camera and object recognition), but these methods are computationally expensive, prone to error and not always applicable because objects should be in the line-of-sight of the camera. In comparison, the generalization of SOs would allow almost certain identification, with little computational cost and fewer physical constraints. Connectivity of all objects is not necessary, as only one or two can give a good idea of the function of a room. What is necessary, though, is the ability for the system to individuate spaces and determine how they are linked to SOs. In addition to being markers of location function, SOs can be used as decision point landmarks. Typically, street intersections are navigational decision points where travelers have to choose between several possible paths, and humans usually encode the objects they see at these navigational decision

points as landmarks [29]. These encodings can occur and be used unconsciously, through the involvement of the para-hippocampal gyrus [29]. They can also occur consciously, through the use of language (e.g. "*I need to turn left after the bus stop*"). The context of VIP and SOs is, however, slightly different from that of visual objects. The perception of the position of a SO by a VIP through the SO2SEES is much less precise than the perception of its position by a sighted person. It is therefore likely that SOs information would rather be used as a means to confirm navigation decisions than a means to make the decisions in the first place. Furthermore, if their positions are known, SOs can be used to provide a detailed spatial insight into the organization of a room. Upon entering a room, a VIP will get useful knowledge by receiving the list of the closest spatially relevant SOs. Spatial relevance can be estimated, as an initial approach, by examining the physical geometry of the considered objects: desks, tables, and furniture or bulky objects in general tend to be relevant. The names of the SOs should be accompanied with their positions relatively to the VIP, yielding a descriptive list such as "*There is a desk in front of you; There is a table on your right; There is a cupboard on your left; etc.*"

2) SOs FOR NON-SPATIAL PERCEPTION

One interesting possibility with SOs is to query high-level information in the way of virtual assistants (e.g. Apple's Siri, Amazon's Alexa or Microsoft's Cortana): the user asks a question and the system answers. For instance, the user can feel cold and ask "*Why is it cold?*" The system will check local SOs and try to find reasons that could explain the low temperature. An assistive system can do more than replace abilities lost due to handicap: it can offer new possibilities of interactions with the world. A rich IoT environment is a very interesting ground for new forms of interactions. In the context of visual impairment, it is even more interesting, as information that SOs manipulate does not only consist of physical quantities, which are difficult to communicate without the help of visual displays, but also of symbolic information or of knowledge. Exploiting low and high-level information, the 2SEES can act as an interface between objects and their individual functionalities. But more can be done: the 2SEES could exploit the SOs information and holistically combine them to provide high-level knowledge to the VIP. The general querying process is the following: based on their wish, the VIP expresses a query in natural language, the system converts it into a formal high-level query, which is broken down into several sub-queries. Reasoning is then performed on queried data to answer the high-level query (Figure 5). For instance, to answer the "*Why is it cold?*" question, the 2SEES system needs to:

- Identify objects that have a relationship with the local temperature: it could be a thermometer, the heating system, or a window;
- Identify their relationship with the temperature: the thermometer measures it, the heating system measures and

can influence it, and the window can influence it by being open or closed;
- Query these objects to retrieve data about that relationship: the temperature can be obtained from the thermometer, the state of the heating system (i.e. on/off, setpoint temperature), and the state of the window (i.e. open or closed);
- For instance, the system can check the underlying judgment of the question using the temperature value, and then check possible reasons: the window may be open and/or the heating system may be turned off.

Since the 2SEES acts as a gateway between the user and the surrounding SOs, a lot of information can be transmitted to the VIP. However, because of the limitations of transmission channels (see section I), it is not possible to exchange large quantities of data between the VIP and SOs. Thus, reducing the amount of exchanged data is a central issue of the design of the 2SEES-user interface. In order to provide only meaningful, concise and intelligible information, available data have to be filtered. If a low-level querying model were to be adopted, users would have to verbally navigate and access SOs data, and most of their effort would be spent telling the system which SOs and attributes it must query, which is a rather tedious communication process. On the opposite, high-level queries offer a great benefit compared to basic ones (data without reasoning) because they move that burden from the VIP to the system. One important tool to achieve relevant data selection is the use of contextual information. The word *context*, here, refers to any information that is not directly related to the fact being examined: i. VIP's personal characteristics and profile; ii. short-time history (e.g. what the VIP was looking for before: a form of personal purpose); iii. social purpose of the current location; iv. explicit and implicit language rules, and so on. The most essential and perhaps simplest piece of context information is the current location of the user. Indeed, locality is tightly linked to SOs as they are interactions between the Internet and the physical world (e.g. measures, actuations, user interfaces). However, it is not the only determinant context clue: for instance, the query "*I am looking for an oven*" will not have the same meaning whether it is pronounced by a client in a home appliance store or by a seller in the same shop who is looking for a way to warm their meal during lunch break. A prominent advantage of localization as context information is that it is easy to use. On the one hand, it is a basic functionality of the system that needs to be computed whatever the level of progress of the SO communication functionality. On the other hand, spatial filtering, of which ambient SO connectivity is a simple implementation, allows the pruning of a lot of irrelevant data. A reasonable query evaluator should therefore take the context into account (Fig. 6). The topics about which users may want to query are influenced by the context: e.g. if they are at a tramway station they might want to know when the next tramway will come to that particular station. A representation of the context inside the system
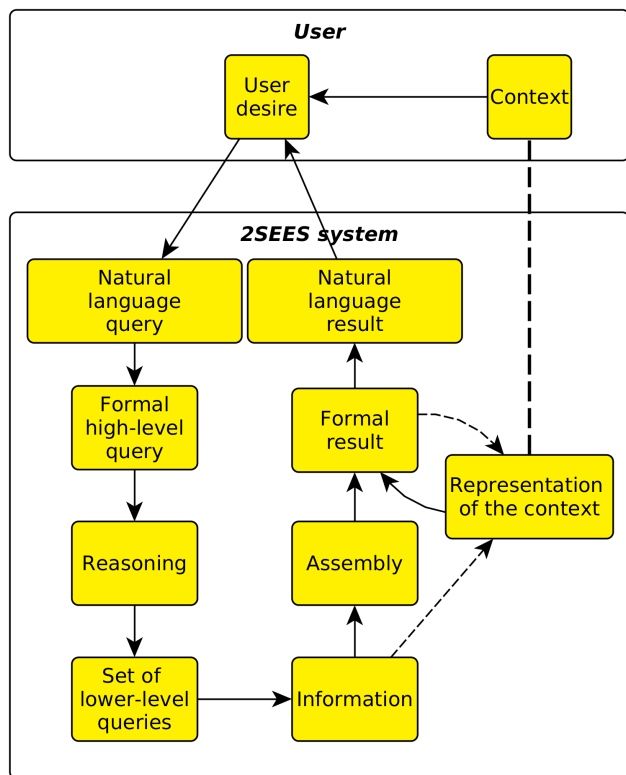
FIGURE 6. SO2SEES query model taking context into account.

will be needed to understand the query (e.g. which tramway station needs to be checked?). Query evaluation depends on the correct translation of the user's wording, expressed in natural language, into a formal query containing symbols that are meaningful for the system. This translation is dependent on certain forms of context, such as language rules. For instance, the query "*Why is it cold?*" involves at least two rules: the fact that "*it*" does not refer to a particular object is controlled by an explicit rule of English language, and the fact that the sought location is controlled by an implicit rule (i.e. "here"). To implement the query and context module, we had the choice between two approaches: a trained Artificial Intelligence (AI) system that answers queries, or offering pre-defined queries to users. We chose the latter, in order to avoid having to developed an AI able to answer natural language queries. A difficulty with that approach is to keep a balance between the expressiveness of the system (i.e. the expressiveness of the union of all queries), the ease of access of queries for the VIP (e.g. selecting a query in a list offering hundreds of choices is impractical), and the upstream work to define queries (as there is an infinity of possible queries). To solve this non-trivial problem, we mainly rely on the context of information on SO neighborhood: queries are bound to specific objects or specific properties of objects, and the system only proposes requests when the particular devices are near the person – that is, generally, when they are likely to be requested. The queries are backed by rules,

and our choice was also motivated by the availability of the Semantic Web Rule Language (SWRL), another standard of the W3C [30]. SWRL extends the logic of OWL with rules: each rule is composed of a *body* (the antecedent) and a *head* (the consequent), which are in turn composed of conjunctions of *atoms*. Atoms are elementary verifiable facts on variables, e.g., "*This variable belongs to that class*", or "*These two variables are linked by that property*". A rule is evaluated the following way: if the antecedent holds, then the consequent holds. A canonical example of such a rule (in this example, x, y and z denote people) is:

$$hasParent(?x,?y) \; \hat{} \; hasBrother(?y, \; ?z) \; = \; > \; hasUncle(?x, \; ?z)$$

It can be formulated in natural language as "*If y is a parent of x, and if y and z are brothers, then z is necessarily x's uncle*". In the SO2SEES system, such rules are part of the knowledge base, extending the descriptive knowledge encoded in OWL and RDFs. In fact, the rules form a large part of the knowledge required to answer queries. The system needs a given quantity of semantic logic (i.e. world modeling) to be able to answer queries. This logic can be either situated on the query side or on the knowledge base side, but it is mandatory: the use of SWRL rules implies to locate the logic on the knowledge base side. This alternative has the strong advantage of allowing the sharing of logic rules among all the queries, which leads to a better scalability in expressiveness of the queries. Another advantage is that the burden of defining the logic linking handled concepts moves away from the SO2SEES system to SOs manufacturers, location managers and general ontology creators. Moreover, in case of logic missing to implement specific queries, rules can be added by the administrators of the SO2SEES system: the knowledge base on which query reasoning is performed is built on the SO2SEES cloud anyway. In a straightforwardly developed system, the problem of answering queries is dynamic: the system needs to understand the logic subtending the formulated query and then to use the logic to formally query the knowledge base. However, with enough rules in the knowledge base, the problem becomes a static one: matching the formulated query to a rule or a set of rules leads to an answer. The first problem – i.e. finding a correspondence between natural language words and known concepts – is still an open research issue, but the nature of the second problem has changed. Moving complexity away from execution time allows for higher performance in answering complex queries. Having rules encoded as parts of the knowledge base is also interesting in the case where the rules are generated by an AI system (e.g., by mining conversation to extract logical relationships between known concepts). Beside the above facts, which still hold, explicit rules have the advantage of providing a human-understandable representation of the logic governing the system. This characteristic is important when working on a system that must be dependable. In order to simplify both the front-end (recognition of natural language and building of concept correspondence) and the semantic logic part of the query evaluator, we can limit ourselves to

a reduced set of queries. By using a simplified model of natural language queries, we can further lower the cost of queries translation from natural language to semantic formal language.

III. TEST SCENARIO

In this section, we describe the hardware and software used to perform the demonstration scenario, which is aimed at validating the concepts detailed above. For that purpose, we had to demonstrate (i) their feasibility, and (ii) their usefulness. As the latter implies the former, the system could not be tested if it did not work: so, a fortiori, its usefulness could not be evaluated. Therefore, the scenario has been focused on demonstrating the usefulness of the concept of environment perception assistance through SOs: spatial and non-spatial perception. To establish the benefit of *spatial perception*, we can investigate the effect of the usage of the system during orientation-needing tasks. The first criterion is the objective performance, i.e. the time required to accomplish the task and the obstacle avoidance. Moreover, it is important to evaluate the assistance level as perceived by VIPs, which could be done through a survey. The contributions of SO-sensing to the localization function of the SO2SEES have not been evaluated here, only the added perception gained by users when directly interacting with SOs. The utility of *non-spatial perception* is harder to evaluate because it does not affect commensurate and measurable variables, at least not in a simple way. Indeed, it is a rather new way of interacting with the world and gives access to information that was previously generally unreachable. Perceived gains may though be assessed through surveys.

A. HARDWARE

To test and validate the proposed concept, seven SOs were deployed in an office of the laboratory. The SO hardware is based on the uSu-Edu board [31] powered by a standard battery and equipped with the following sensors (Fig. 7): a 3-axis accelerometer, a 3-axis gyrometer, a 3-axis compass, a light sensor, an air temperature sensor and a barometric pressure
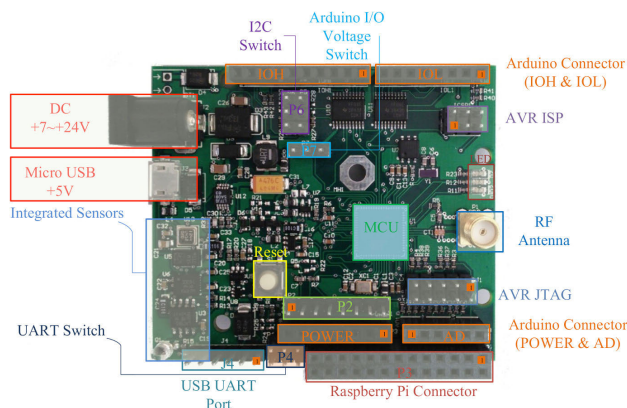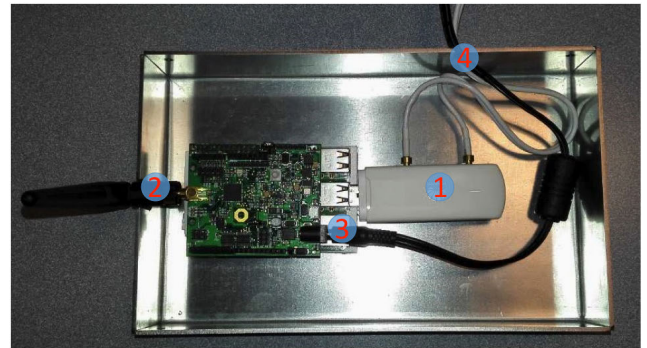

FIGURE 7. The uSu-Edu board.

FIGURE 8. Multi–support SO Local server equipped with 3/4G modem.
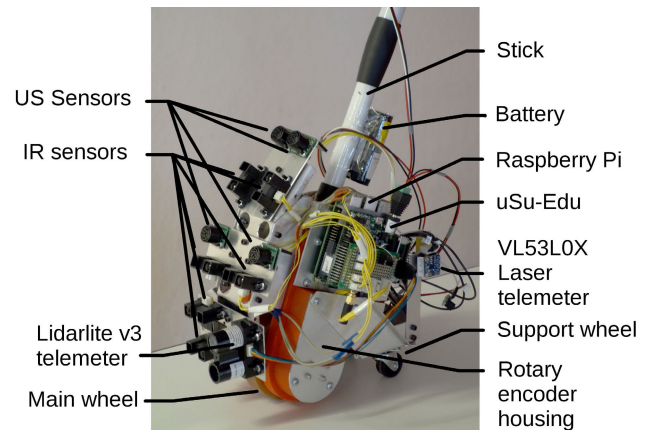

FIGURE 9. The 2SEES prototype.

sensor. This board has an UART port, an extension port that enables connections to different Arduino shields, and a port that enables a direct connection with a Raspberry Pi. Each SO is connected to the Internet through a multi-support local server using the IEEE802.15.4 (ZigBee) wireless access medium. The uSu-Edu board is therefore versatile enough to implement different type of SOs. To meet the application requirements, it is important to implement a multi-support SO local server to connect SOs to their back-ends. This local server node is build on a Raspberry Pi 3 and a uSu-Edu board, which offer different communication protocols such as Wi-Fi, IEEE802.15.4 (ZigBee), Ethernet and 3G/4G mobile network (Fig. 8). Each multi–support SO server can manage up to 20 SOs with a 0 dBi low cost antenna within a range of 200 m. This range may be increased by using an appropriate antenna. We implemented the specific firmware running on the uSu-Edu board to serve as a coordinator to setup a star topology network. The 2SEES prototype (Fig. 9) has been used, together with a smartphone running Android, to implement the local platform of the SO2SEES system. The 2SEES has originally been designed as a demonstrator for obstacle detection, localization and navigation functionalities [32]. Like our SOs, it is built upon the uSu-Edu and Raspberry Pi 3 boards. In the context of this experiment, the 2SEES
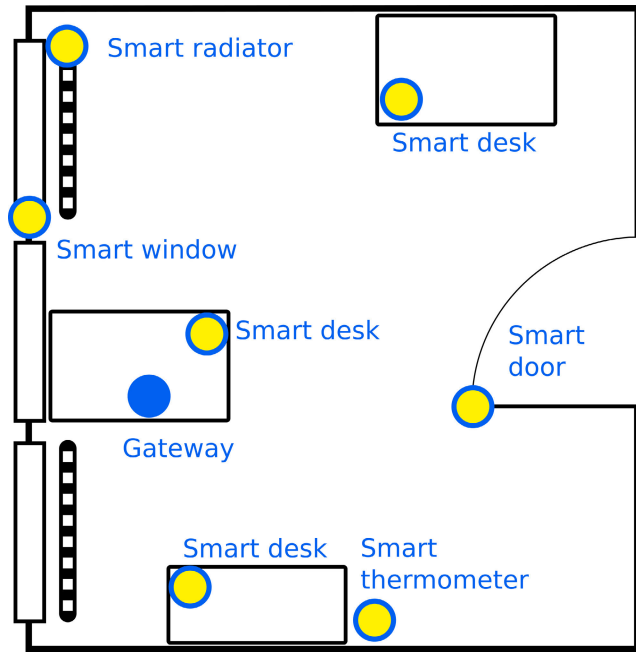
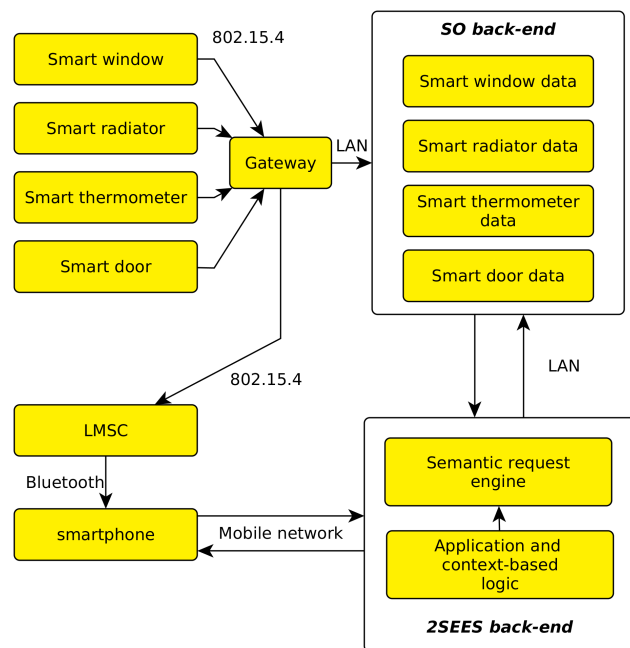**FIGURE 10.** Map of the room where the scenario took place.

FIGURE 11. Smart window (left) and smart door (right).

FIGURE 12. SO2SEES elements and communications between them in the test scenario.

FIGURE 13. Display of some available queries, in French (a: *What is the current temperature?* b: *Why is it cold?*) on the SO2SEES smartphone application.

prototype only acts as a bridge between the gateway of the SOs (IEEE 802.15.4) and the smartphone (WiFi).

B. SCENARIO SETUP

We have set up a limited experiment scenario in order to show the feasibility of the contextual querying system concept described above, i.e. spatial and non-spatial perception. The scenario unfolds in the following way: a VIP enters a room; at that moment, their SO2SEES system receives a list of the objects present in the room, analyses it, and then proposes a list of context-meaningful queries to the user. Finally, the user can choose a query and execute it. The scenario took place in a room containing seven SOs, and was structured around one high-level semantic query and one low level. The seven SOs were the entry door of the room, a window, a thermometer, a radiator, and three desks representing obstacles (Fig. 10). The door and the window (Fig. 12) were able to

know their status (open or closed) using a gyroscope and a state-machine; the thermometer periodically measured the ambient temperature; the radiator integrated a thermometer and adjusted its power output to try to match the temperature setpoint defined by the user. All these SOs sent the data they generated to a back-end platform acting as the cloud via a multi-support gateway (IEEE802.15.4 and IEEE802.3y) (Fig. 10, 11). The gateway also acts as the local communication link between the SO environment and the SO2SEES system: it advertises the IRIs of the SOs to the SO2SEES system. When the VIP enters the room, they open and close the door. The gateway detects an entrance in the room by monitoring the status of the door: it triggers the local, wireless transmission of the list of the SOs' IRIs to the local platform of the 2SEES (LMSC). The LMSC transmits this list to the smartphone, which in turn transmits it to the 2SEES back-end, which was part of the same LAN as the other back-end and the gateway. The IRIs point to files containing WoT-TD descriptions of the objects, hosted on the back-end of the SOs. The 2SEES back-end downloads these files and analyses them: individual RDF nodes (context cues) are extracted and used to select context-relevant queries. To link queries with context cues, a simple bipartite graph is progressively built over the life cycle of the system: when new objects are encountered, context cues are extracted from their descriptions and linked to relevant queries; if necessary, new queries are added by SO2SEES human operators. That way, more and more queries become available as the graph expands. The high-level queries available in the experiment are: "*Why is it cold?*", and its symmetric: "*What is the current temperature?*" (Fig. 13). A query to list obstacles ("*Are there obstacles in the room?*") is also offered to users. To answer the two high-level queries, the system needs to observe the status of the associated SOs: an open window or a turned off radiator are possible reasons among others. Therefore, the query execution program consists in retrieving the status of every object in the room related to the temperature – and transmitting it to the user: "*The thermometer measures 14 °C; the window is closed; the radiator is turned off.*" This process induces context relevance constraints: the only properties to be checked must (i) be related to temperature and (ii) belong to objects located inside the room. In our implementation, the first constraint was solved by the local transmission mechanism: the gateway knew where SOs were and which room had been entered. Embedding objects location in their descriptive metadata could also solve this constraint. The temperature relevance has to be embedded in the knowledge base, either vendor's side or 2SEES side. To illustrate the full functionality of the SO2SEES, a specific ontology for VIPs still needs to be developed, but in this work we only focused on the feasibility of using SOs to improve the perception of VIPs' environment. Thus, a model has been designed by integrating with the Web of Things – Thing Description ontology and the SWRLTab framework [33]. The SWRLTab is a Protégé plugin that provides a development environment for working with SWRL rules and SQWRL queries [34]. A room

is identified by its identifier (*id*) and has a number. A room can be equipped with several SOs. A SO is characterized by its identifier (*title*), and has several properties. A property is characterized by its type (*title*) and its value. A value can be a constant (*const*), a variable (*value*) or a state (*state*). Some of these SOs may be associated with objects considered as obstacles (e.g., *desk*). Therefore, obstacle detection is performed automatically by the inference engine. In addition, one or more alarms can be defined for each room. An alarm is characterized by its identifier (*id*) and has a type (*type*). Each alarm can have several causes; the triggering of the alarms and the identification of their possible causes is done automatically by the inference engine. We defined four rules:

1. A rule to specify that the desks present in the current room are obstacles:

Thing(?thing) ^ hasProperties(?thing, ?prop)
^ hasPropertyAffordanceTitle(?prop, ?title)
^ swrlb:stringEqualIgnoreCase(?title, "type")
^ hasPropertyAffordanceConst(?prop, ?const)
^ swrlb:stringEqualIgnoreCase(?const, "desk")
^ hasThing(?room, ?thing) => hasObstacle(?room, ?thing)

2. A rule to trigger low temperature alarms in a room (i.e. when temperature is lower than 15 degrees):

Thing(?thing) ^ hasProperties(?thing, ?prop1)
^ hasPropertyAffordanceTitle(?prop1, ?title1)
^ swrlb:stringEqualIgnoreCase(?title1, "type")
^ hasPropertyAffordanceConst(?prop1, ?const1)
^ swrlb:stringEqualIgnoreCase(?const1, "thermometer")
^ hasProperties(?thing, ?prop2)
^ hasPropertyAffordanceTitle(?prop2, ?title2)
^ swrlb:stringEqualIgnoreCase(?title2, "value")
^ hasPropertyAffordanceValue(?prop2, ?value2)
^ swrlb:lessThanOrEqual(?value2, 15)
^ hasThing(?room, ?thing) ^ hasAlarm(?room, ?alarm)
^ hasAlarmType(?alarm, ?type)
^ swrlb:stringEqualIgnoreCase(?type, "lowTemperature")
=> hasTriggedAlarm(?room, ?alarm)

3. A rule to identify that an open window is a possible cause of a low temperature alarm:

Room(?room) ^ hasTriggedAlarm(?room, ?alarm)
^ hasAlarmType(?alarm, ?type)
^ swrlb:stringEqualIgnoreCase(?type, "lowTemperature")
^ hasThing(?room, ?thing) ^ hasProperties(?thing, ?prop1)
^ hasPropertyAffordanceTitle(?prop1, ?title1)
^ swrlb:stringEqualIgnoreCase(?title1, "type")
^ hasPropertyAffordanceConst(?prop1, ?const1)
^ swrlb:stringEqualIgnoreCase(?const1, "window")
^ hasProperties(?thing, ?prop2)
^ hasPropertyAffordanceTitle(?prop2, ?title2)
^ swrlb:stringEqualIgnoreCase(?title2, "state")
^ hasPropertyAffordanceState(?prop2, ?state2)
^ swrlb:stringEqualIgnoreCase(?state2, "open")
=> hasPossibleOrigin(?alarm, ?thing)

4, A rule to highlight that a turned off radiator is a possible cause of cold:

Room(?room) ˆ hasTriggedAlarm(?room, ?alarm)
ˆ hasAlarmType(?alarm, ?type)
ˆ swrlb:stringEqualIgnoreCase(?type, "lowTemperature")
ˆ hasThing(?room, ?thing) ˆ hasProperties(?thing, ?prop1)
ˆ hasPropertyAffordanceTitle(?prop1, ?title1)
ˆ swrlb:stringEqualIgnoreCase(?title1, "type")
ˆ hasPropertyAffordanceConst(?prop1, ?const1)
ˆ swrlb:stringEqualIgnoreCase(?const1, "radiator")
ˆ hasProperties(?thing, ?prop2)
ˆ hasPropertyAffordanceTitle(?prop2, ?title2)
ˆ swrlb:stringEqualIgnoreCase(?title2, "state")
ˆ hasPropertyAffordanceState(?prop2, ?state2)
ˆ swrlb:stringEqualIgnoreCase(?state2, "off")
= > hasPossibleOrigin(?alarm, ?thing)

To illustrate the SO2SEES functionality, these four rules have been implemented in Protégé together with the scenario parameters, i.e. the seven smart objects. Then, the following scenario has been evaluated in *Room1* (Fig. 14, 15) where *Window1* was open, Radiator1 was turned off, *Door1* was open, and the temperature measured by *Thermometer1* was low (12.5 degrees). The system monitored only one alarm (*Alarm1*), of which type was *lowTemperature*. The rules allowed the reasoner embedded in Protégé to deduce that the three desks (*Desk1*, *Desk2*, *Desk3*) were obstacles and to trigger *Alarm1* (Fig. 14). Moreover, the two possible causes of *Alarm1* (*Window1* open and *Radiator1* turned-off) have also been correctly inferred by the reasoner (Fig. 15). Many other scenarios could have been evaluated, but their presentation was beyond the scope of this paper.

IV. DISCUSSION

A. ERROR CHECKING AND DETECTION

Apart from providing environment spatial and non-spatial perception support, SOs can be leveraged for error detection and correction. Indeed, context information extracted from both occurrences and the data generated by SOs can be used to recheck information generated by other components of the system (e.g. embedded sensors). All information generated by the system is prone to error, particularly that of both essential functions: navigation and obstacle signaling. Firstly, during navigation, the system can be mistaken about the
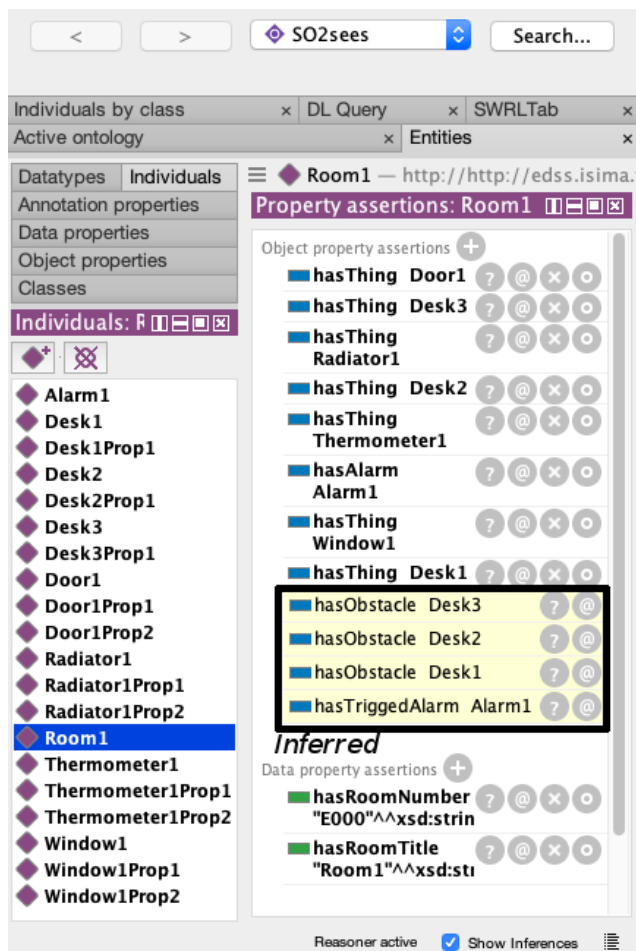

FIGURE 14. Obstacle presence and temperature alarm inference in Protégé.
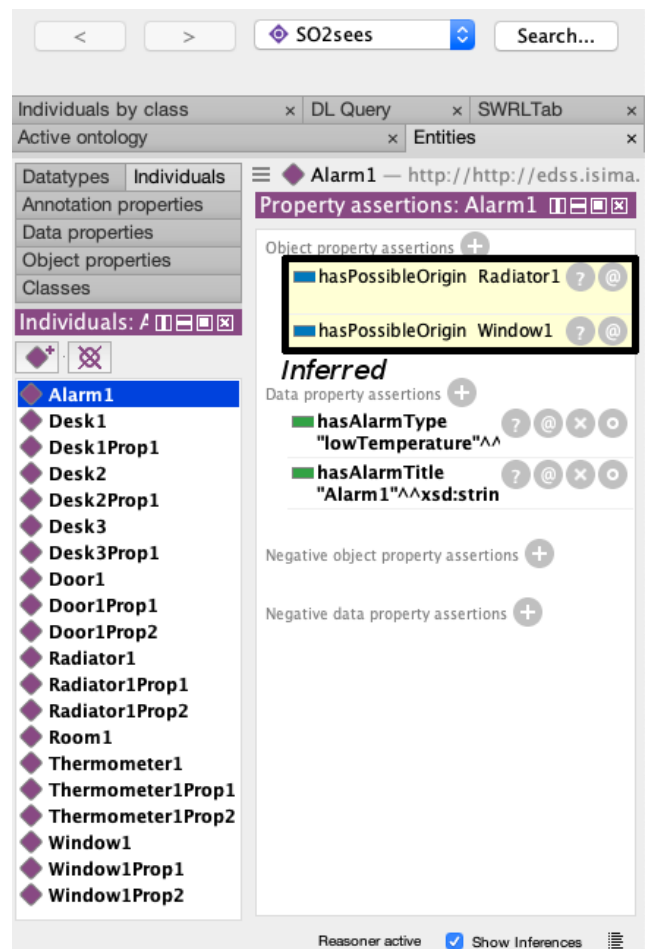

FIGURE 15. Possible causes of a low temperature in the room.

VIP's location, by a "small" distance. Indeed, some error is essentially unavoidable because of the uncertainty of sensory data and of processing algorithms, but it should be kept within limits to avoid downgrading of the system functionality. The system can also be deeply mistaken about the VIP's location. For example, it could estimate that the user is in another building than the actual one, or, worse, in a city instead of another one. As unlikely as this type of error may appear at first sight, it can occur for at least two reasons: if there is an error in a local sub-map (e.g. one building door links to the right city while another links to a wrong city), or if the system loses localization information, and in the absence of an absolute localization (e.g. after a hardware reboot with previous state loss in a building). The system can also give wrong navigation directions, because of outdated map information, for instance. Secondly, in obstacle detection, the system can miss some obstacles because of their shape and position (e.g. a hanging wire), their material characteristics (e.g. an automatic glass door), their variable position (e.g. a door can be closed or open), or their movement (e.g. a vehicle). Hopefully, the information provided by surrounding SOs can mitigate some of these issues. During navigation, mitigation of "small" errors is difficult because it requires SOs to have accurate location data about both themselves and the user. As a first approach, radio-based indoor positioning techniques could be used to compute the user's position in a building, but this issue has not been addressed in this paper. Smart doors can also be used to fix these errors: since they are stationary and precisely indicated in building floor plans, they can be easily located and they can give an accurate estimation of the user's location when they are being used (opened and crossed). However, as other people can interact with the door, the system should be able to discriminate usage from its user and outsiders. Mitigation of "large" errors is easier. Indeed the mere detection of a SO will often lead to the discovery of its location; the system will then be faced with a discrepancy between its internal state and external information. To conclude, navigation errors are difficult to mitigate as this requires knowledge about maps or even paths from SOs, which they are unlikely to hold a priori. It is difficult to find real-world examples of obstacles that are undetected by sensors because of their shape and position, which, when equipped with communication abilities, would provide enough spatial information to be reliably avoided by VIPs. This would require SOs to hold accurate information about both their location and shape (i.e. be mapped), and about the localization and trajectory course of the user, which is, again, a priori unlikely. The only cases standing out as intuitively possible are doors and windows, but there might be others. For objects that could hardly be detected because of their characteristics (e.g. glass doors), it is possible for the system to activate or to give more trust to specific sensors (e.g. ultrasound instead of laser sensors) if it can identify these objects through network communication in the first place. Obstacles that are difficult to detect because of their movement constitute an interesting case. Intelligent transportation systems (ITS) are quickly developing. One compelling component of such systems is the knowledge of the position and velocity of on-road vehicles, which are either measured by the vehicles themselves or by road infrastructure. These data benefit from being shared along the road in order to increase the efficiency and the safety of the transport network. They could thus be picked up by the assistive system from dedicated emitters at crossings, or from available ITS infrastructure (e.g. traffic signals, bus, tramways and taxi stations) in order to detect incoming vehicles.

B. INTEROPERABILITY ISSUES

The above corrections of navigation and obstacle detection raise interoperability issues. Indeed, data produced by SOs need to be encoded in a format that the 2SEES can understand at the semantic level, and the possible interactions with the SOs should be described in an equally coherent semantic format. In that aim, the SO2SEES functionality relies on several assumptions:

- All SOs have an associated IRI;
- The 2SEES can access a formal description of SOs from their IRI;
- This IRI can lead to the data of the SO, either directly, if it represents an accessible location, or indirectly, (e.g. the resource name) and a mean to access the SO from its name;
- The SOs periodically broadcast locally their IRI. Alternatively, any type of discovery protocol can be employed, as long as it is standardized across devices (e.g. answering identity queries broadcasted by the 2SEES).

The performance of the SO2SEES is tightly linked to its ability to interact with SOs and cloud platforms. The wider the range of connectable external platforms, the better the SO perception function will work: more diverse SOs can be encountered in more locations, allowing the exploitation of synergies between SOs and enhancing the ubiquity of the query service. However, such interactions with externally managed entities would only be possible if stakeholders reach a consensus on standards – preferably, open ones – defining mechanisms at all level of interactions, from physical connectivity to semantic information exchange and interfacing. But it is unlikely that disability-oriented issues become a prime driver for setting IoT interoperability standards as they weigh less than general consumer applications. The 2SEES system will thus have to follow standards of the IoT industry and to integrate them in order to work with SOs. The system architecture itself has to follow constraints arising from interoperability issues. Indeed, different architectures will put different constraints on the shape of the IoT environment. The system should then be designed in a way that minimizes constraints, or, more specifically, in a way that maximizes the likelihood of constraints to be fulfilled. Several constraints have driven our design choices. The first and chiefly important one is the question of the access to SOs, which underlies several

other questions like physical-layer networking, high-level addressing and routing. The key choice is whether the system must communicate with SOs directly - using connectivity on the lower layers of the OSI or TCP/IP model - or indirectly, using network layer or higher. Direct wireless communication allows for fast interactions and avoids the need of having Internet access. However, it requires interoperability, from the physical level, to be built into SOs. On the other hand, indirect access puts few constraints on SOs themselves but requires a heavy infrastructure: Internet access and back-ends. It thus appears that the constraints of indirect access are less demanding than those of direct access as they are shifted from smart objects themselves to higher elements in the hierarchy (i.e. edge routers, cloud back-ends). In many applications of IoT, SOs are very heavily constrained, particularly in terms of energy and computing power. Consequently, it would be counterproductive for a standard to impose the implementation of additional functions to objects such as self-advertising for discovery or getting out of sleep mode to answer requests. On a physical level, different interfaces and protocols are adapted to different applications, and the 2SEES is not likely to have hardware and software for all of them. Furthermore, wired networks, which are commonly used in building automation systems, for instance, do not allow direct connection. Indirect access is more flexible: Internet access is to be expected in areas with a rich SO environment and access to SO data through cloud back-ends is also very likely. Another important choice is the access method, of which a first aspect is the discovery of objects. We have chosen to use local broadcast from the edge router, a middle ground between discovery broadcast/answering from objects and a mechanism purely independent from local communication where objects identifiers would be taken from a geographical database. The latter possibility requires an accessible and up to date set of databases (each vendor probably having its own) and accurate localization. The second aspect is the addressing and the access itself. In the OSI or TCP/IP network model, Layer 3/network addressing could be possible, thanks to the large address space of IPv6, but some of the most constrained SOs do not have enough resources to be directly connected. Indirect or higher level addressing solves this issue: the location pointed by the address can lead to an intermediate service representing the node (i.e. a virtual node). We have chosen to use dereferencable IRIs as identifiers and access methods for objects. IRIs are standard [35] and widely used, in particular, they are the chosen method to identify objects in the W3C standard RDF [36].

V. CONCLUSION

In this article, we have presented several applications of smart objects (SOs) for the perceptual assistance of visually impaired people (VIP). In both fields of spatial and non-spatial perception, mechanisms enabling SOs to provide perceptual information have been discussed. Furthermore, insights into potential and practical implementations have been given. In particular, we have designed a new system,

called SO2SEES, that allows VIP to get information from distributed knowledge bases belonging to various actors of the Internet of Things ecosystem (manufacturers, vendors, distributors). This system is based on semantic web standards of the W3C (i.e. RDF, RDFS, OWL, SWRL) and allows VIP to choose among lists of pre-defined queries selected on the basis of their relevance with respect to context. To validate our concept, we have implemented a simple version of the SO2SEES that works in controlled conditions. We assume that this new approach, combined with current assistive technologies (e.g. GNSS, obstacle sensing), will enhance VIP's perceptual ability, both indoors and outdoors, due to the widespread deployment of SOs in the coming years (e.g., smart care, smart transportation, smart homes and buildings) However, our work is still at an early stage and thorough testing should be performed: both synthetic performance testing and subjective feedback are required. Experiments with VIP will provide critical reviews of the concept and its implementation. Formal comparisons with other assistive devices should also be conducted. Beyond testing, the SO2SEES system also requires additional development. Coupling it with external localization mechanisms could enhance its functionality: the local broadcast of IRI for the discovery phase could be replaced by geographical knowledge, effectively transferring one interoperability issue from the hardware of the local platform to another component of the knowledge base. More broadly, a substantial amount of work and research still has to be done on spatial perception.

REFERENCES

[1] C. Thinus-Blanc and F. Gaunet, "Representation of space in blind persons: Vision as a spatial sense?" *Psychol. Bull.*, vol. 121, no. 1, pp. 20–42, 1997.

[2] R. R. A. Bourne *et al.*, "Magnitude, temporal trends, and projections of the global prevalence of blindness and distance and near vision impairment: A systematic review and meta-analysis," *Lancet Global Health*, vol. 5, no. 9, pp. e888–e897, Sep. 2017, doi: 10.1016/S2214-109X(17)30293-0.

[3] R. Efron, "What is perception?" in *Proc. Boston Colloq. Philosophy Sci.*, R. S. Cohen and M. W. Wartofsky, Eds. Dordrecht, The Netherlands: Springer, 1969, pp. 137–173.

[4] D. L. Schacter, D. T. Gilbert, D. M. Wegner, and M. K. Nock, *Psychology*, 3rd ed. New York, NY, USA: Worth Publishers, 2014.

[5] E. Ahissar and E. Assa, "Perception as a closed-loop convergence process," *eLife*, vol. 5, p. e12830, May 2016, doi: 10.7554/eLife.12830.

[6] C. Teufel and B. Nanay, "How to (and how not to) think about top-down influences on visual perception," *Consciousness Cognition*, vol. 47, pp. 17–25, Jan. 2017, doi: 10.1016/j.concog.2016.05.008.

[7] E. Pissaloux and R. Velázquez, "On spatial cognition and mobility strategies," in *Mobility of Visually Impaired People: Fundamentals and ICT Assistive Technologies*, E. Pissaloux and R. Velázquez, Eds. Cham, Switzerland: Springer, 2018, pp. 137–166, doi: 10.1007/978-3-319-54446-5_5.

[8] J. M. Loomis, R. L. Klatzky, and N. A. Giudice, "Sensory substitution of vision: Importance of perceptual and cognitive processing," in *Assistive Technology for Blindness and Low Vision*, R. Manduchi and S. Kurniawan, Eds. Boca Raton, FL, USA: CRC Press, 2012.

[9] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. New York, NY, USA: W.H. Freeman and Company, 1982.

[10] P. Bach-y-Rita, C. C. Collins, F. A. Saunders, B. White, and L. Scadden, "Vision substitution by tactile image projection," *Nature*, vol. 221, no. 5184, pp. 963–964, Mar. 1969.

[11] C. Lenay and G. Declerck, "Technologies to access space without vision. Some empirical facts and guiding theoretical principles," in *Mobility of Visually Impaired People: Fundamentals and ICT Assistive Technologies*, E. Pissaloux and R. Velázquez, Eds. Cham, Switzerland: Springer, 2018, pp. 53–75, doi: 10.1007/978-3-319-54446-5_2.

[12] B. Hsu, C.-H. Hsieh, S.-N. Yu, E. Ahissar, A. Arieli, and Y. Zilbershtain-Kra, "A tactile vision substitution system for the study of active sensing," in *Proc. 35th Annu. Int. Conf. IEEE EMBS*, Osaka, Japan, Jul. 2013, pp. 3206–3209.

[13] H. Kajimoto, M. Suzuki, and Y. Kanno, "HamsaTouch: Tactile vision substitution with smartphone and electro-tactile display," in *Proc. CHI*, Toronto, ON, Canada, Apr./May 2014, pp. 1273–1278, doi: 10.1145/2559206.2581164.

[14] S. Maidenbaum, S. Hanassy, S. Abboud, G. Buchs, D.-R. Chebat, S. Levy-Tzedek, and A. Amedi, "The 'EyeCane', a new electronic travel aid for the blind: Technology, behavior & swift learning," *Restorative Neurol. Neurosci.*, vol. 32, no. 6, pp. 813–824, 2014, doi: 10.3233/RNN-130351.

[15] J. Borenstein and I. Ulrich, "The GuideCane—A computerized travel aid for the active guidance of blind pedestrians," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 2, 1997, pp. 1283–1288, doi: 10.1109/ROBOT.1997.614314.

[16] W. Bischof, E. Krajnc, M. Dornhofer, and M. Ulm, "NAVCOM–WLAN communication between public transport vehicles and smart phones to support visually impaired and blind people," in *Computers Helping People with Special Needs*, K. Miesenberger, A. Karshmer, P. Penaz, and W. Zagler, Eds. Berlin, Germany: Springer, 2012, pp. 91–98, doi: 10.1007/978-3-642-31534-3_14.

[17] Handisco. *Sherpa Par Handisco*. Accessed: Aug. 1, 2019. [Online]. Available: https://handisco.com

[18] Microsoft. *Seeing AI*. Accessed: Aug. 1, 2019. [Online]. Available: https://www.microsoft.com/ en-us/seeing-ai/

[19] D. Lymberopoulos, J. Liu, X. Yang, R. R. Choudhury, V. Handziski, and S. Sen, "A realistic evaluation and comparison of indoor location technologies: Experiences and lessons learned," in *Proc. 14th Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, Washington, DC, USA, 2015, pp. 178–189, doi: 10.1145/2737095.2737726.

[20] L. A. Brabyn and J. A. Brabyn, "An evaluation of 'talking signs' for the blind," *Hum. Factors*, vol. 25, no. 1, pp. 49–53, 1983.

[21] J. R. Marston and R. G. Golledge, "Improving transit access for the blind and vision impaired," *Res. Updates Intell. Transp. Syst.*, vol. 7, no. 2, pp. 4–5, and 11, 1998.

[22] Y. Tian, C. Yi, and A. Arditi, "Improving computer vision-based indoor wayfinding for blind persons with context information," in *Computers Helping People with Special Needs*, K. Miesenberger, J. Klaus, W. Zagler, and A. Karshmer, Eds. Berlin, Germany: Springer, 2010, pp. 255–262, doi: 10.1007/978-3-642-14100-3_38.

[23] M. Yusro, K. M. Hou, E. Pissaloux, H. L. Shi, K. Ramli, and D. Sudiana, "SEES: Concept and design of a smart environment explorer stick," in *Proc. 6th Int. Conf. Hum. Syst. Interact. (HSI)*, Jun. 2013, pp. 70–77, doi: 10.1109/HSI.2013.6577804.

[24] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010, doi: 10.1016/j.comnet.2010.05.010.

[25] J. Ibarra-Esquer, F. González-Navarro, B. Flores-Rios, L. Burtseva, and M. Astorga-Vargas, "Tracking the evolution of the Internet of Things concept across different application domains," *Sensors*, vol. 17, no. 6, p. 1379, Jun. 2017, doi: 10.3390/s17061379.

[26] G. Fortino, A. Guerrieri, and W. Russo, "Agent-oriented smart objects development," in *Proc. IEEE 16th Int. Conf. Comput. Supported Cooperat. Work Des. (CSCWD)*, May 2012, pp. 907–912, doi: 10.1109/CSCWD.2012.6221929.

[27] S. Kaebisch, T. Kamiya, M. McCool, and V. Charpenay, *Web of Things (WoT) Thing Description*, W3C, W3C Candidate Rec., May 2019. Accessed: Aug. 1, 2019. [Online]. Available: https://www.w3.org/TR/2019/CR-wot-thing-description-20190516/

[28] G. Declerck and C. Lenay, "Living in space. A phenomenological account," in *Mobility of Visually Impaired People*, E. Pissaloux and R. Velázquez, Eds. Cham, Switzerland: Springer, 2018, pp. 3–51, doi: 10.1007/978-3-319-54446-5.

[29] G. Janzen and M. van Turennout, "Selective neural representation of objects relevant for navigation," *Nature Neurosci.*, vol. 7, no. 6, pp. 673–677, May 2004, doi: 10.1038/nn1257.

[30] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "SWRL: A semantic Web rule language combining OWL and RuleML," in *Proc. WC*, May 2004, pp. 1–31. Accessed: Aug. 1, 2019. [Online]. Available: https://www.w3.org/Submission/SWRL/

[31] S. Rodrigo-Mu noz, O. Trullols, J. M. Barcelo-Ordinas, J. Garcia-Vidal, K. M. Hou, H. Shi, and X. Diao, "Collective awareness platform for tropospheric ozone pollution," Univ. Politècnica de Catalunya and Univ. Clermont Auvergne, Clermont-Ferrand, France, Tech. Rep., 2017. Accessed: Aug. 1, 2019. [Online]. Available: https://www.captor-project.eu/wp-content/uploads/2018/02/D4.5_final.pdf

[32] J. Connier, S. Xu, P. Vaslin, C. de Vaulx, H. Shi, K. M. Hou, and J. Li, "The 2SEES smart stick: Concept and experiments," in *Proc. 11th Int. Conf. Hum. Syst. Interact. (HSI)*, 2018, pp. 226–232, doi: 10.1109/HSI.2018.8431361.

[33] M. J. O'Connor and A. Das, "The SWRLTab: An extensible environment for working with SWRL rules in Protégé-OWL," Tech. Rep., Jan. 2006. [Online]. Available: https://www.researchgate.net/publication/253384943_The_SWRLTab_An_Extensible_Environment_for_working_with_SWRL_Rules_in_Protege-OWL

[34] M. O'Connor and A. Das, "SQWRL: A query language for OWL," in *Proc. 6th Int. Conf. OWL, Exper. Directions*, Aachen, Germany, vol. 529, 2009, pp. 208–215.

[35] M. Duerst and M. Suignard, *Internationalized Resource Identifiers (IRIs)*, document RFC 3987, Jan. 2005. [Online]. Available: https://www.ietf.org/rfc/rfc3987.txt

[36] D. Wood, M. Lanthaler, and R. Cyganiak, "RDF 1.1 concepts and abstract syntax," document, W3C, W3C Rec., Feb. 2014. Accessed: Aug. 1, 2019. [Online]. Available: http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/

JEAN CONNIER was born in 1992. He received the double M.E. degree in computer science from the ISIMA School, University Blaise Pascal, Clermont-Ferrand, France, in 2015, and the Harbin Institute of Technology, China, in 2017. He is currently pursuing the Ph.D. degree with the LIMOS Laboratory. He is working on an assistive device for mobility of visually impaired people encompassing navigation assistance, obstacle avoidance assistance, and perceptual enhancement through the usage of the ever-denser environment of smart objects.

HAIYING ZHOU was born in 1975. He received the B.E. and M.E. degrees in electronic engineering from Wuhan University, China, in 1997 and 1999, respectively, and the Ph.D. degree in computer science from Blaise Pascal University, France, in 2004. He was a Postdoctoral Researcher with the Institute of Cemagref (UR TSCF/UMR TETIS) and the LIMOS Laboratory (CNRS UMR 6158), from 2005 to 2007. He is currently a Full Professor with the Hubei University of Automotive Technology (HUAT), China. His research interests include the areas of new technologies and smart applicants of the Internet of Things (IoT), electric vehicle and intelligent driving, embedded computing, mobile computing, operation systems, and wireless sensor networks.

CHRISTOPHE DE VAULX was born in France in 1976. He received the Ph.D. degree in computer science from University Blaise Pascal, France, in 2003. From 2004 to 2005, he was a Postdoctoral Researcher with the Laboratory of Computing, Modelling and Optimization of the Systems (LIMOS) a Mixed Unit of Research (UMR 6158) in computing, University Blaise Pascal. Since 2005, he has been an Associate Professor with Polytech Clermont-Ferrand and LIMOS, University Blaise Pascal, from 2005 to 2016, and has been with University Clermont Auvergne, since 2017. He has authored more than 40 articles and two inventions. His research interests include cyber-physical systems, the Internet of Things, wireless sensors networks, embedded systems, and wireless communication protocols.
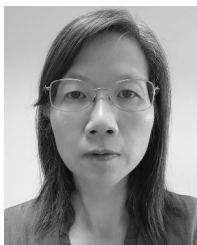
JIAN-JIN LI received the bachelor's degree in mathematics from Wuhan University, China, in 1987, the master's degree in computer science from the INSA de Lyon, in 1989, the Ph.D. degree in computer science from University Lyon 1, in 1992, and the Ph.D. degree from the LIP Laboratory, Ecole Normale Supérieure de Lyon, France, under the supervision of Pr. Yves Robert. In 1990s, she worked in parallel programming, developing parallel algorithms in 3D images processing, and communication algorithms for massively parallel processing architectures. Since 2000, she has been working in the communication protocols for intelligent transport systems, wireless sensor networks, and the Internet of Things. She is currently an Associate Professor with the Institute of Computer Science, University of Clermont Auvergne.

HONGLING SHI was born in China in 1979. He received the B.S. and M.S. degrees from the Electronic Information School, Wuhan University, China, in 2000 and 2004, respectively, and the Ph.D. degree in computer science from LIMOS— UMR 6158—UBP/CNRS, France, in 2014. From July 2005 to July 2008, he worked on Feature Phone Software in Honsyn, Wuhan, China. From July 2004 to June 2005, he worked on MIB and Embedded Software in Melit, Wuhan. From July 2008 to November 2009, he worked on ARM Linux-Based Embedded Software in Arkage, Wuhan. From December 2009 to March 2019, he was a Research and Development Engineer with the SMIR Team, LIMOS— UMR 6158. Since November 2009, he has been focusing on researching and developing of energy efficient and fault tolerant wireless sensor networks. He is currently an Associate Professor with the College of Electronics and Information Engineering, South-Central University for Nationalities, Wuhan. His major research interests include wireless sensor networks, cyberphysical systems, the Internet of Things, and web of thing.

PHILIPPE VASLIN received the Ph.D. degree in sport sciences (specialty: biomechanics) from Bordeaux University, France, in 1993. He joined Université Clermont Auvergne (UCA), Clermont-Ferrand, France, in 1994 as an Assistant Professor. Since 1994, he has mainly been working on biomechanics of manual wheelchair locomotion at LIMOS, UCA. This research has been financially supported by two French National Projects (ANAES 1999 and ANR TecSan from 2006 to 2011). Beside this main topic, he has also been involved in research projects on sport biomechanics and on wheeled robotics, jointly with Pprime Institute, Poitiers University, France, and the Pascal Institute, UCA, France, respectively. Since 2014, he has been involved in a project focused on localization and navigation of Visually Impaired People (VIP), which led to the design and build of an original electronic white cane (2SEES). This project has been funded by the French Ministry of Higher Education and by the Region Auvergne-Rhône-Alpes (AURA). He has authored or coauthored about one hundred of national and international scientific articles, book chapters, and communications. He is a member of the French Society of Biomechanics (SB) and of the European Society of Biomechanics (ESB).
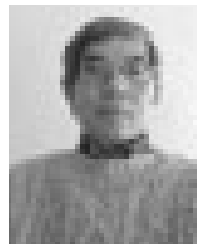
KUN MEAN HOU was born in Cambodia, in 1956. He received the Ph.D. degree and the HDR degree in computer science from the University of Technology of Compiègne (UTC), France, in 1984 and 1996, respectively. He worked as an Associate Professor with UTC, from 1984 to 1986. In 1986, he joined IN2 as a Research and Development Engineer Group Leader to develop fault-tolerant super-minicomputer. From 1989 to 1996, he created a research group, which investigated parallel architecture dedicated to real-time image processing at laboratory HEUDIASYC UMR 6599 CNRS (UTC). In 1997, he joined the College of Engineering School, Institut Supérieur d'Informatique de Modélisation et de leurs Applications (ISIMA) as a Full Professor, where he created the Systèmes Multisensoriels Intelligents integrés et Répartis (SMIR) Team of the laboratory LIMOS UMR 6158 CNRS (ten researchers). SMIR team focuses on the development of core technologies: basic hardware and software dedicated to wireless sensor networks or the IoT. Different IoT nodes (Bluetooth, Wi-Fi, and ZigBee), embedded wireless communication and embedded real-time kernel (SDREAM, LIMOS, and HEROS) are implemented and deployed in different applications such as telemedicine, intelligent transportation systems, and precision agriculture. Since October 2018, he has been an Emeritus Professor with University Clermont-Ferrand, France, and a Visiting Professor with the Hubei University of Automotive Technology (HUAT), China. He holds six patents and two APPs. He evolved in three EU projects and 20 technology transfers. He published more than 140 articles in the journals and internal conferences. He is involved in several scientific committees and boards. He is an ICT-AGRI EU Expert. He is currently an Associate Editor of IEEE Access and Associate Editor of IEEE Access for a special issue on smart farming 4.0.

• • •